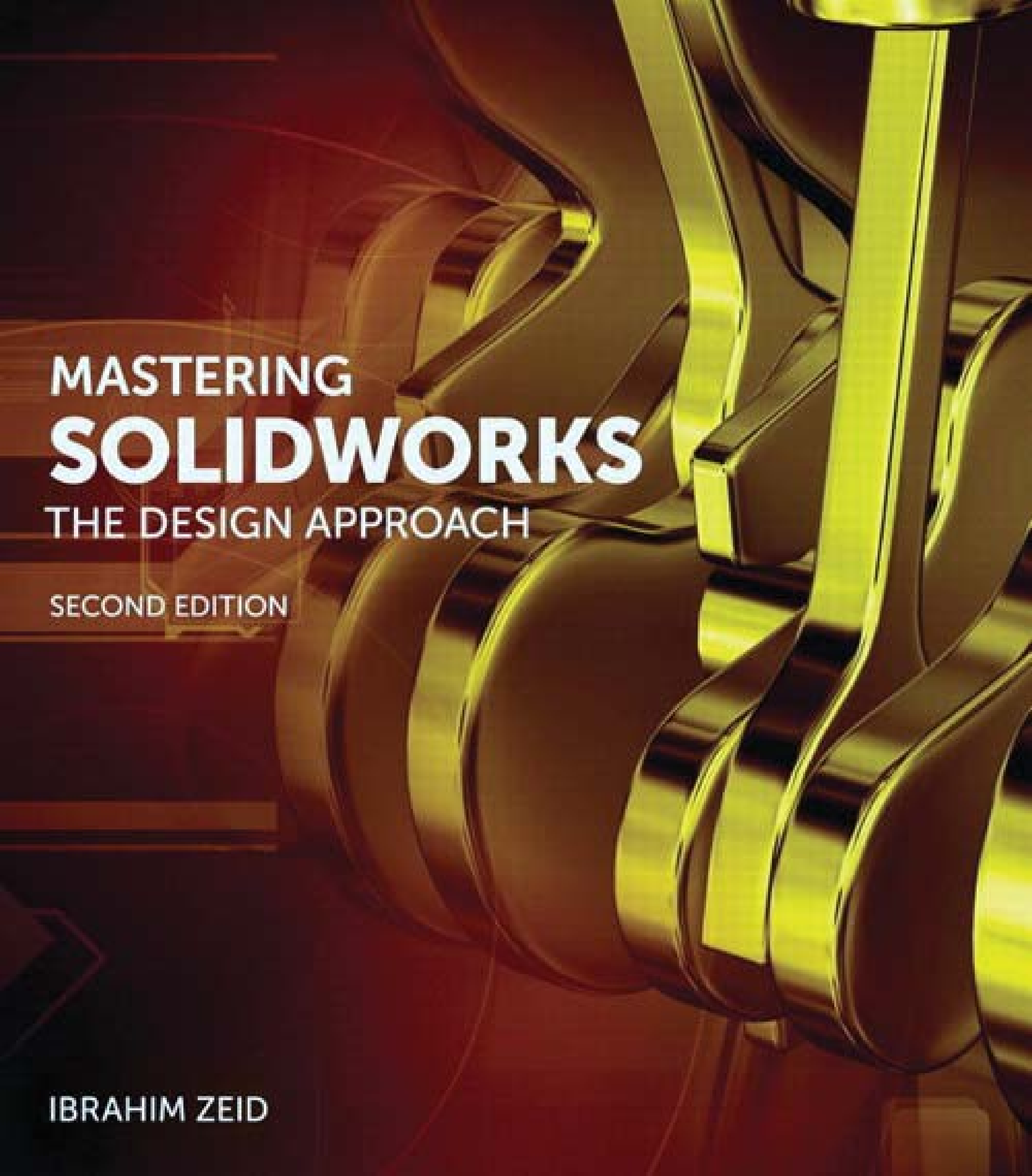

MASTERING
SOLIDWORKS
THE DESIGN APPROACH
SECOND EDITION
IBRAHIM ZEID

Mastering SolidWorks®

The Design Approach

Second Edition

Ibrahim Zeid

Northeastern University

PEARSON

Boston Columbus Indianapolis New York San Francisco Upper Saddle River
Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montreal Toronto
Delhi Mexico City São Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo

Executive Editor: Lisa McClain
Production Editor: Katerina Malone
Cover Designer: Mimi Heft
Full-Service Project Management: Mohinder Singh/Aptara®, Inc.
Composition: Aptara®, Inc.
Printer/Binder: Edward Brothers Malloy
Cover Printer: Edward Brothers Malloy
Text Font: Berkeley Book

Credits and acknowledgments borrowed from other sources and reproduced, with permission, in this textbook appear on the appropriate page within the text.

Library of Congress Cataloging-in-Publication Data

Zeid, Ibrahim.
Mastering SolidWorks®: the design approach / Ibrahim Zeid, Northeastern University. — Second edition.
pages cm.
ISBN 978-0-13-388594-1 — ISBN 0-13-388594-1 1. SolidWorks. 2. Computer graphics.
3. Computer-aided design. I. Title.
T385.Z44 2015
620'.0042028553—dc23

2014016374

10 9 8 7 6 5 4 3 2 1

PEARSON

ISBN-13: 978-0-13-388594-1
ISBN-10: 0-13-388594-1

Features of *Mastering SolidWorks*®: *The Design Approach*

Tutorials

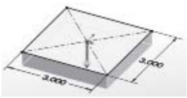
Tutorial 4–6: Create Features

This tutorial covers the creation of these features: chamfer, fillet, slot, shell, draft, and rib. All dimensions are inches. Here are useful observations:

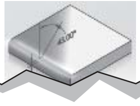
1. Make sure to pay attention to the visual clues shown in the left pane while creating these features.
2. For example, the box symbol under **Chamfer parameters** indicates that you can chamfer a face, an edge, or a vertex (corner point). As expected, chamfering a face chamfers all its edges. Chamfering a corner chamfers the three edges that meet there.
3. A rib requires a profile sketch (e.g., a line or stepwise line) and a thickness.

Step-by-Step Instructions

Step 1: Create *Sketch1* and *Block* feature: **File** > **New** > **Part** > **OK** > **Top Plane** > **Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > reverse extrusion direction > enter 0.5 for thickness **D1** > ✔ > **File** > **Save As** > *tutorial4.6* > **Save.**



Step 2: Chamfer an edge of *Block* feature: **Fillet** dropdown on **Features** tab > **Chamfer** > select *Block* edge shown



Step 5: Shell *Block* feature: **Shell** on **Features** tab > select top face of *Block* > enter 0.1 for wall thickness **D1** > ✔.

Step 6: Draft *Block* feature: Delete the chamfer, fillet, and shell features > **Draft** on **Features** tab > enter 10 degrees for **Draft Angle** > select top face of *Block* as **Neutral Plane** > select *Block* four side faces to draft > ✔.

Hands-on for Tutorials

HANDS-ON FOR TUTORIAL 12–1. Edit the title block to add a tolerance general note in the **Comments** box of the title block. The note should read:

GENERAL TOLERANCE
.X ± .030
.XX ± .010
.XXX ± .005
.XXXX ± .0005

Examples and Solutions

Example 12.1 Calculate the limits and tolerance zones for the following three fits: clearance of RC3, transition of LT4, and interference of FN2. Use a basic size of 5.0000 in.

Solution Following the above four steps, Table 12.2 shows the results.

TABLE 12.2 Limits for Basic Size d = 5.000 in.

Fit	d_{hmin}	d_{hmax}	d_{smin}	d_{smax}	h	s	A
RC3 (H7/f6)	5.0000	5.0016	4.9974	4.9984	0.0016	0.0010	0.0016
LT4 (H8/k7)	5.0000	5.0025	5.0001	5.0017	0.0025	0.0016	−0.0017
FN2 (H7/s6)	5.0000	5.0016	5.0035	5.0045	0.0016	0.0010	−0.0045

Problems

problems

1. List the sources of variability in manufacturing.
2. Why do we need tolerances?
3. List the two types of tolerances. What does each type control?
4. Inspection gauges are used to check whether a part is within its tolerance limits. Perform an in-depth research study on inspection gauges including their types, their design, and how they are used during part inspection.
5. What is the difference between a nominal and a basic size? Give an example.
6. Three types of tolerances exist: unilateral, bilateral, and symmetric. Describe each type. Give a numerical example of each type.

Preface

The target audience for this book is college students in courses that use SolidWorks to learn and master CAD/CAM for design, visualization, prototyping, and manufacturing. The book's primary market is four-year colleges and two-year community colleges. Freshman Engineering Design courses should find this book useful, refreshing, and interesting. Other important markets include high schools, professionals, and training courses. We have written the book with the target audience in mind. Page iii highlights some of the book's features.

The book includes just the right amount of math in Chapter 8 (Curves), Chapter 9 (Surfaces), and Chapter 13 (Analysis Tools). The math is concentrated in one or two sections in each of these three chapters. We include the math for two reasons. First, it shows students who are curious how CAD/CAM systems work "under the hood." Second, it broadens the book's appeal to many students, professors, and readers. This math may be ignored without affecting the continuity of the coverage of the material in any of these three chapters.

The philosophy behind the book is original, unique, and effective. We cover and present SolidWorks as a design system rather than a software program. Thus, instead of focusing on describing SolidWorks menus and syntax, we describe design approaches, methodologies, and techniques to help CAD designers/engineers and draftspersons achieve their engineering tasks in the fastest, easiest, and most effective way.

Based on this philosophy, the book approach uses design, modeling, and drafting concepts as the building blocks, instead of menus and commands. Thus, we present command sequences to achieve CAD and modeling tasks. Of course, we provide SolidWorks syntax and details, but do so in keeping with the proposed philosophy of the book. We start with a CAD task to accomplish (what to do) and then go about accomplishing it (showing how to use SolidWorks to do it). This philosophy is more motivating to student learning than simply going through layers of menus and commands.

The book approach is designed to bring the real power of SolidWorks as a powerful modeling and design system instead of only a software program. We include challenging modeling and design examples and problems in the book. As part of the book's unique approach, we cover the theoretical concepts behind the various functions of SolidWorks. This should provide information to curious minds about why things work the way they do, as well as explain their limitations and use.

The book provides plenty of illustrations, step-by-step instructions, and rich and challenging end-of-chapter problems. The book is suitable for use at various levels, from freshman to senior to graduate courses. Instructors can choose the chapters and topics that suit their teaching needs and courses. They can also choose the level of depth. The book includes both examples and tutorials. An example covers one concept whereas a tutorial is more comprehensive by covering a full design task. Each example and tutorial has a hands-on exercise at the end that serves two purposes. First, it ensures that the student has gone through the example or tutorial, because it builds on it. Second, it both challenges the student's understanding and extends it.

The book is organized into parts and chapters. Instructors may cover the chapters in any order and select as they see fit their course and student needs. However, we recommend covering Chapters 1 and 2 first to build a sound background in 3D CAD/CAM modeling concepts. Chapter 1 is designed to provide a quick grasp of basic functionalities: create parts, create assemblies, and create drawings. These three functionalities correspond to the three modes of SolidWorks: part, assembly, and drawing. The idea here is that students can start designing basic and simple products after using only one chapter of the book; they do not have to wait until later chapters to learn how to design and document simple parts and assemblies. Then they can dig deeper in the later chapters to learn more. Thus, Chapter 1 provides breadth and the remainder of the book provides depth. As for why we need Chapter 2, it covers all the essential concepts required for a sound understanding of the 3D modeling concepts and efficient use of today's parametric features-based solid modeling CAD/CAM systems such as SolidWorks.

I would like to thank many people who contributed to this book including my students, the book reviewers, the Prentice Hall team, the editorial team, and my family. Many of my students have shaped how I should present and teach concepts to help them understand better. They have also contributed directly and indirectly to the book. This book is the outcome of their influence.

I would also like to thank Boston Gear for granting permission to download and use some of their gears in the book tutorials. I thank Jake Hustad for providing the Universal Joint assembly. I also thank Ivette Rodriguez of ASME for granting permission to use ASME Y14.5M-1994 (R2004) material.

Many thanks are due DS SolidWorks Corporation for its technical support throughout the writing process and using SolidWorks. My sincere thanks go to my friends Marie Planchard, Christian Blanc, and Christine Morse.

I owe thanks to the many reviewers who helped to shape this book (first edition). They are:

Charles Coleman, *Argosy University*
Paige Davis, *Louisiana State University*
Joe Fitzpatrick, *VIC Inc., Boston, Massachusetts*
Max P. Gassman, *Iowa State University*
Julia Jones, *University of Washington*
Dean Kerste, *Monroe County Community College*
Julie Korfhage, formerly of *Clackamas Community College*
Paul Lienard, *Northeastern University College of Professional Studies*
Payam H. Matin, *University of Maryland Eastern Shore*
Jianbiao (John) Pan, *California Polytechnic State University*
Lisa Richter, *Macomb Community College*
Nishit Shah, *NyproMold Inc., Massachusetts*
David W. Ward, *Clackamas Community College*

Special thanks are due Lisa McClain of Peachpit/Pearson for her help and support in handling all the book logistics.

Last, but not least, my family and friends deserve many thanks for their support, and apologies to them for hiding out to finish this book project. Their love and unconditional support is priceless.

The author always looks forward to and values feedback. Please contact him at zeid@coe.neu.edu with any ideas, corrections, typos, or parts/assemblies that you may want to donate. Full credit will be given to you.

Abe Zeid
Boston, MA

Instructor Resources

The **Online Instructor's Manual** provides answers to chapter exercises and tests and solutions to end-of-chapter problems; drawing files to get users started; and lecture-supported PowerPoint slides.

To access supplementary materials online, instructors need to request an instructor access code. Go to **www.pearsonhigherred.com/irc**, where you can register for an instructor access code. Within 48 hours after registering you will receive a confirming email including an instructor access code. Once you have received your code, go to the site and log on for full instructions on downloading the materials you wish to use.

Brief Contents

Part I **Computer Aided Design (CAD) Basics 1**

1 Getting Started 3
2 Modeling Management 41
3 Design Intent 85

Part II **Basic Part Modeling 99**

4 Features and Macros 101
5 Drawings 137
6 Assemblies 159
7 Rendering and Animation 201

Part III **Advanced Part Modeling 221**

8 Curves 223
9 Surfaces 255
10 Sheet Metal and Weldments 295
11 Sustainable Design 325

Part IV **Part Development and Analysis 341**

12 Tolerances 343
13 Analysis Tools 383

Part V

Part Manufacturing 415

14 Rapid Prototyping 417
15 Numerical Control Machining 431
16 Injection Molding 483

A ANSI and ISO Tolerance Tables 509
B SolidWorks Certification 523

Index 535

Contents

Part I Computer Aided Design (CAD) Basics 1

1 Getting Started 3

1.1 Introduction 3
1.2 Engineering Design Process 4
1.3 CAD Process 4
1.4 Manufacturing Process 5
1.5 CAM Process 6
1.6 SolidWorks Installation and Resources 6
1.7 SolidWorks Overview 8
1.8 Customize SolidWorks 12
1.9 Modeling Plan 13
1.10 Part Creation 14
1.11 Examples 15
1.12 Tutorials 23
Tutorial 1–1 Create the Flap 23
Tutorial 1–2 Create the Pin and Bushing Bearing 25
Tutorial 1–3 Create the Pillow Block 26
Tutorial 1–4 Create Drawings 29
Tutorial 1–5 Create Assembly 31
PROBLEMS 34

2 Modeling Management 41

2.1 Overview 41
2.2 Types of CAD Models 41
2.3 Planning Part Creation 43
2.4 Part Topology 44
2.5 Parametric Modeling 44
2.6 Customizing SolidWorks 46
2.7 Productivity Tools 47
2.8 Coordinate Systems 47
2.9 Sketch Planes 48
2.10 Sketch Status 50
2.11 Part Features Tree 51
2.12 Construction Geometry 52
2.13 Reference Geometry 52
2.14 Sketch Entities 54
2.15 Sketch Relations 55
2.16 Equations and Link Values 55
2.17 Geometric Modifiers 57
2.18 Grids 58
2.19 Patterns 58
2.20 Selecting, Editing, and Measuring Entities 62
2.21 Boolean Operations 63
2.22 Templates 65
2.23 Viewing 65
2.24 Model Communication 66
2.25 Tutorials 67
Tutorial 2–1 Create a Coil Spring 67
Tutorial 2–2 Create Mount Plate 69
Tutorial 2–3 Create Bracket 70
Tutorial 2–4 Create Wheel 71
Tutorial 2–5 Create Tire and Pin 73
Tutorial 2–6 Create Caster Assembly 74
PROBLEMS 77

3 Design Intent 85

3.1 Introduction 85
3.2 Capturing Design Intent 86
3.3 Documenting Design Intent 87

3.4 Comments 87
3.5 Design Binder 88
3.6 Equations 88
3.7 Design Tables and Configurations 89
3.8 Dimension Names 89
3.9 Feature Names 89
3.10 Folders 90
3.11 Tutorials 90
Tutorial 3–1 Design Intent via Two Modeling Plans 90
Tutorial 3–2 Design Intent via Three Modeling Plans 92
Tutorial 3–3 Design Intent via Design Specifications 94
Tutorial 3–4 Design Intent via Mating Conditions 96
PROBLEMS 98

Part II Basic Part Modeling 99

4 Features and Macros 101

4.1 Introduction 101
4.2 Features 102
4.3 Spur Gears 105
4.4 Design Library and Library Features 110
4.5 Configurations and Design Tables 111
4.6 Macros 112
4.7 Tutorials 115
Tutorial 4–1 Create Sweep Features 115
Tutorial 4–2 Create Loft Features 117
Tutorial 4–3 Use the Hole Wizard 118
Tutorial 4–4 Create Compression Spring 120
Tutorial 4–5 Create Spiral 121
Tutorial 4–6 Create Features 122
Tutorial 4–7 Use the Smart Fasteners Wizard 123
Tutorial 4–8 Create a Bolt 124
PROBLEMS 127

5 Drawings 137

5.1 Introduction 137
5.2 Engineering Drafting and Graphics Communication 138
5.3 ASME Abbreviation Rules 139
5.4 ASME Drafting Rules 140
5.5 ASME Dimensioning Rules 140
5.6 Dimensions 143
5.7 Drawing Content and Layout 146
5.8 Angle of Projection 147
5.9 Views 147
5.10 Sheets 149
5.11 Title Block 150
5.12 Drafting Control 150
5.13 Tolerances 151
5.14 Bill of Materials 151
5.15 Model and Drawing Associativity 152
5.16 Design Checker 152
5.17 Tutorials 152
Tutorial 5–1 Create Drawing Views 152
Tutorial 5–2 Insert Annotations 153
Tutorial 5–3 Fill Title Block 154
Tutorial 5–4 Create Assembly Drawing with Bill of Materials 155
Tutorial 5–5 Use Model-Drawing Associativity 156
PROBLEMS 158

6 Assemblies 159

6.1 Introduction 159
6.2 Assembly Mates 160
6.3 Bottom-Up Assembly Modeling 160
6.4 Top-Down Assembly Modeling 161
6.5 Assembly Tree 165
6.6 Assembly Drawing 166
6.7 Assembly Exploded View and Animation 166
6.8 Assembly Motion Study 167
6.9 Interference and Collision Detection 167
6.10 Assembly Design Tables 168
6.11 Tutorials 168
Tutorial 6–1 Create Cam and Follower Assembly 168
Tutorial 6–2 Create Working Hinge Assembly 170
Tutorial 6–3 Mate Two Gears with Gear Mate 171
Tutorial 6–4 Create Functional Rack and Pinion 172
Tutorial 6–5 Create a Functional Ball Screw 175

Tutorial 6–6 Study Universal Joint Motion 176
Tutorial 6–7 Create Motion Study 178
Tutorial 6–8 Detect Collision and Interference 180
Tutorial 6–9 Create Design Table 181
Tutorial 6–10 Create Part in Context of Assembly 183
PROBLEMS 186

7 Rendering and Animation 201

7.1 Introduction 201
7.2 Scenes and Lighting 201
7.3 Rendering Models 202
7.4 Decals 203
7.5 Textures 205
7.6 Materials 205
7.7 Appearance and Transparency 206
7.8 Background and Scenes 206
7.9 Cameras and Camera Sleds 207
7.10 Animation 207
7.11 Tutorials 207
Tutorial 7–1 Apply Colors to Objects 208
Tutorial 7–2 Apply Background and Scene 208
Tutorial 7–3 Apply Lights to Scene 209
Tutorial 7–4 Add Material and Transparency 210
Tutorial 7–5 Add Camera to Scene 211
Tutorial 7–6 Create Motion Study 212
Tutorial 7–7 Create Camera-Sled Based Animation 215
PROBLEMS 219

Part III Advanced Part Modeling 221

8 Curves 223

8.1 Introduction 223
8.2 Curve Representation 224
8.3 Line Parametric Equation 225
8.4 Circle Parametric Equation 226
8.5 Spline Parametric Equation 227
8.6 Two-Dimensional Curves 228
8.7 Three-Dimensional Curves 229
8.8 Curve Management 230
8.9 Tutorials 230
Tutorial 8–1 Create a 2D Curve Using Explicit Equation 230
Tutorial 8–2 Create a 2D Curve Using Parametric Equation 231
Tutorial 8–3 Create a 3D Curve Using Parametric Equation 231
Tutorial 8–4 Create a 3D Curve Using 3D Points 233
Tutorial 8–5 Create a 3D Curve Using 3D Sketching 234
Tutorial 8–6 Create a 3D Curve Using Composite Curves 235
Tutorial 8–7 Create a 3D Curve by Projecting a Sketch onto a Curved Face 237
Tutorial 8–8 Create a 3D Curve Using Projected Curves 238
Tutorial 8–9 Create a Stethoscope Model 247
PROBLEMS 250

9 Surfaces 255

9.1 Introduction 255
9.2 Surfaces 255
9.3 Using Surfaces in Solid Modeling 258
9.4 Surface Representation 259
9.5 Plane Parametric Equation 260
9.6 Ruled Surface Parametric Equation 262
9.7 Surface Visualization 265
9.8 Surface Management 266
9.9 Tutorials 266
Tutorial 9–1 Create Basic Surfaces: Extrude, Revolve, Loft, Sweep, Knit, and Radiate 266
Tutorial 9–2 Create Basic Surfaces: Planar, Filled, Boundary, and Offset 268
Tutorial 9–3 Visualize Surfaces 269
Tutorial 9–4 Create an Artistic Bowl 270
Tutorial 9–5 Use Surface Intersections 273
Tutorial 9–6 Create a Tablespoon 275

Tutorial 9–7 Create a Computer Mouse 276
Tutorial 9–8 Create a Baseball Hat 279
Tutorial 9–9 Create a Hair Dryer 282
Tutorial 9–10 Create an Oil Container 285
PROBLEMS 288

10 Sheet Metal and Weldments 295

10.1 Introduction 295
10.2 Sheet Metal 295
10.3 Sheet Metal Features 297
10.4 Sheet Metal FeatureManager Design Tree 302
10.5 Sheet Metal Methods 303
10.6 Weldments 305
10.7 Weldment Features 307
10.8 Weld Symbols 310
10.9 Tutorials 311
Tutorial 10–1 Create Sheet Metal 312
Tutorial 10–2 Create Sheet Metal Drawing 313
Tutorial 10–3 Create Sheet Metal Part from Solid Body 315
Tutorial 10–4 Create Sheet Metal Part from Flattened State 316
Tutorial 10–5 Create Weldment 317
Tutorial 10–6 Create Weldment Drawing 319
PROBLEMS 322

11 Sustainable Design 325

11.1 Introduction 325
11.2 Design and Society 326
11.3 Guidelines and Principles 327
11.4 Life Cycle Assessment 329
11.5 Impact Metric 330
11.6 Implementation 331
11.7 Design Activities 332
11.8 Sustainable Design Tools 333
11.9 SolidWorks SustainabilityXpress 333
11.10 Tutorials 336
Tutorial 11–1 Redesign a Steel Washer 336
PROBLEMS 339

Part IV Part Development and Analysis 341

12 Tolerances 343

12.1 Introduction 343
12.2 Types 344
12.3 Concepts 344
12.4 ASME Tolerance Rules 346
12.5 Tolerancing Tapers 349
12.6 Limits of Dimensions 351
12.7 Tolerance Accumulation 355
12.8 Statistical Tolerancing 356
12.9 True Position 358
12.10 Geometric Tolerances 358
12.11 Datum Target Symbols 361
12.12 Tolerance Interpretation 362
12.13 Tolerance Analysis 363
12.14 SolidWorks Tolerance Analysis 365
12.15 Tutorials 368
Tutorial 12–1 Create Conventional Tolerances 368
Tutorial 12–2 Create Geometric Tolerances 369
Tutorial 12–3 Define Datum Targets 371
Tutorial 12–4 Tolerance a Taper 372
Tutorial 12–5 Perform Tolerance Stack-up Analysis 373
PROBLEMS 375

13 Analysis Tools 383

13.1 Introduction 383
13.2 Data Exchange 383
13.3 Mass Properties 386
13.4 Animation and Motion Analysis 390
13.5 Flow Simulation 391
13.6 Finite Element Method 391
13.7 Finite Element Analysis 394
13.8 SolidWorks Simulation 395
13.9 Von Mises Stress 396
13.10 Tutorials 400
Tutorial 13–1 Export Native SolidWorks Files 400
Tutorial 13–2 Import IGES and STEP Files into SolidWorks 401
Tutorial 13–3 Calculate Mass Properties of a Solid 402

Tutorial 13–4 Perform Motion Analysis Using a Motor 402
Tutorial 13–5 Perform Static Linear FEA on a Part 407
Tutorial 13–6 Perform Thermal FEA on a Part 409
Tutorial 13–7 Perform Flow Analysis on a Hose 410
PROBLEMS 413

Part V **Part Manufacturing** **415**

14 Rapid Prototyping **417**
14.1 Introduction 417
14.2 Applications 418
14.3 Overview 419
14.4 Concepts 420
14.5 SolidWorks Triangulation 422
14.6 Steps 424
14.7 Building Techniques 424
14.8 Bottle Prototype 425
14.9 Tutorials 426
Tutorial 14–1 Generate Part Prototype File 426
Tutorial 14–2 Generate Assembly Prototype File 427
Tutorial 14–3 Read back an STL File 428
PROBLEMS 430

15 Numerical Control Machining **431**
15.1 Introduction 431
15.2 Basics of Machine Tools 432
15.3 Basics of Machining 434
15.4 Turning 441
15.5 Drilling 442
15.6 Milling 442
15.7 Electrical Discharge Machining 442
15.8 Manufacturing of Design 445
15.9 SolidWorks DFMXpress 445
15.10 Basics of NC Machining 448
15.11 G-Code and M-Code Programming 450
15.12 CAM Add-Ins Software 452
15.13 Tutorials 452
Tutorial 15–1 Turn a Stepped Shaft 453
Tutorial 15–2 Drill Holes 455
Tutorial 15–3 Mill Faces 461
Tutorial 15–4 Mill Pockets 466
Tutorial 15–5 Mill Slots 471
Tutorial 15–6 Wire EDM a Spline Shaft 477
PROBLEMS 480

16 Injection Molding **483**
16.1 Introduction 483
16.2 Basics of Injection Molding Machines 484
16.3 Basics of Injection Molding 485
16.4 Basics of Mold Design 487
16.5 Basics of Part Design 490
16.6 Phases of Mold Design 490
16.7 SolidWorks Mold Design 491
16.8 Tutorials 492
Tutorial 16–1 Create a Block Mold 493
Tutorial 16–2 Create a Sandbox Mold 496
Tutorial 16–3 Create a Hemisphere Mold 500
Tutorial 16–4 Create an Easter Egg Mold 502
Tutorial 16–5 Generate a Mold Drawing 505
PROBLEMS 507

A ANSI and ISO Tolerance Tables **509**
B SolidWorks Certification **523**

Index **535**

PART I

Computer Aided Design (CAD) Basics

The primary goal of Part I is to learn how to use SolidWorks fairly quickly to create parts, assemble them, document them, and visualize them. The core use of SolidWorks in industry is to create CAD parts (SolidWorks Part mode), assemble the parts to create products (SolidWorks Assembly mode), and create drawings of the parts and assemblies for production and manufacturing (SolidWorks Drawing mode).

Chapter 1 (Getting Started) is an overview of SolidWorks, its philosophy, how to configure it, and how to administer it. Chapter 2 (Modeling Management) is about learning the 3D CAD modeling concepts to enable the completion of design tasks in the shortest time possible. Chapter 3 (Design Intent) covers both how to embed design intelligence into CAD design and how the way you create a CAD part influences its future edits and manufacturing.

Now that we are ready to start, note that this book advocates an active learning style. This means you learn as you use SolidWorks to do the book tutorials. Each chapter in the book covers the basic concepts of the chapter subject independent of the SolidWorks syntax (subject semantics), followed by tutorials that show detailed instructions of SolidWorks syntax (subject syntax) to use the concepts to accomplish design tasks. This approach should maximize the value of learning, as the reader can relate syntax to semantics.

This page intentionally left blank

CHAPTER

1

Getting Started

1.1 Introduction

As the preface indicates, this book is written with a focus on achieving CAD design and modeling, and CAM modeling and manufacturing tasks instead of on SolidWorks, its menu/tree structure, and syntax. Of course, we do explain SolidWorks in detail and show how to use it to accomplish these tasks. In other words, we cover SolidWorks as a powerful design and manufacturing system, rather than a software program. This approach makes the reader engaged and eager to learn SolidWorks.

This book does not require prior knowledge of SolidWorks or the CAD and CAM subject matters. Contrary to its name, the book takes a student with no background in CAD and CAM modeling and gradually and systematically builds their background to an advanced and mastery level, thus the name of the book. If you do have prior knowledge of CAD and CAM, you will still benefit from the unique presentation of the material and solidify your prior knowledge.

Another benefit of using this book, in addition to learning and mastering CAD and CAM from scratch, is the ability to pass the SolidWorks CSWA (Certified SolidWorks Associate) and CSWP (Certified SolidWorks Professional) certification exams and become a SolidWorks certified CAD designer. As a matter of fact, the book prepares you far beyond what the CSWA and CSWP exams require. Refer to Appendix B for more details on SolidWorks exams.

This book is written with one goal in mind: to help you become a better CAD designer by (1) understanding the intricacy of three-dimensional (3D) modeling and (2) learning SolidWorks. By understanding 3D modeling concepts, you become more efficient, perform CAD tasks quicker, and eliminate the time-consuming and frustrating trial-and-error approach. By learning SolidWorks, you gain the skills required by commercial CAD/CAM systems. These skills are transferable from one CAD/CAM system to another because all the systems are built on the same theory and concepts. Even though the syntax (user interface) of these systems is different, their semantics (concepts) are the same.

The book uses the tutorial approach for learning, supported by explaining the concepts behind the tutorials. We cover the concepts of each chapter independent of SolidWorks and use the majority of the chapter to cover tutorials related to the concepts. We follow a meaningful numbering system for the tutorials and examples. Tutorials are numbered as Tutorial x–y, where x is the chapter number and y is the tutorial number within the chapter. For example, Tutorials 1–2 and Tutorial 1–3 are, respectively, Tutorial 2 and Tutorial 3 in Chapter 1.

In addition to using tutorials, the book uses examples also, numbered in the same way as the tutorials. Unlike a tutorial, which uses multiple concepts to achieve a design task, an example focuses on only one concept and is always used in its respective section that introduces that concept.

1.2 Engineering Design Process

Unlike science, in which the goal is to understand physical phenomena, engineering is all about making products that work even though it involves doing elaborate, complex theoretical investigations. There are good reasons for that. Engineered products and systems are too complex to fit closed-form solutions. Thus, engineers and designers resort to computational methods and techniques, and design tools (such as CAD) to achieve their part and product design. Designers typically begin with a rough part design of the problem to solve and then continue to refine and test it until the part design meets all the design requirements.

The well-known *engineering design process* (EDP) reflects this very nature of engineering design. Figure 1.1 shows the steps of the EDP. The input to EDP is an idea or a problem to solve. The output is a design to implement the idea or solve the problem.

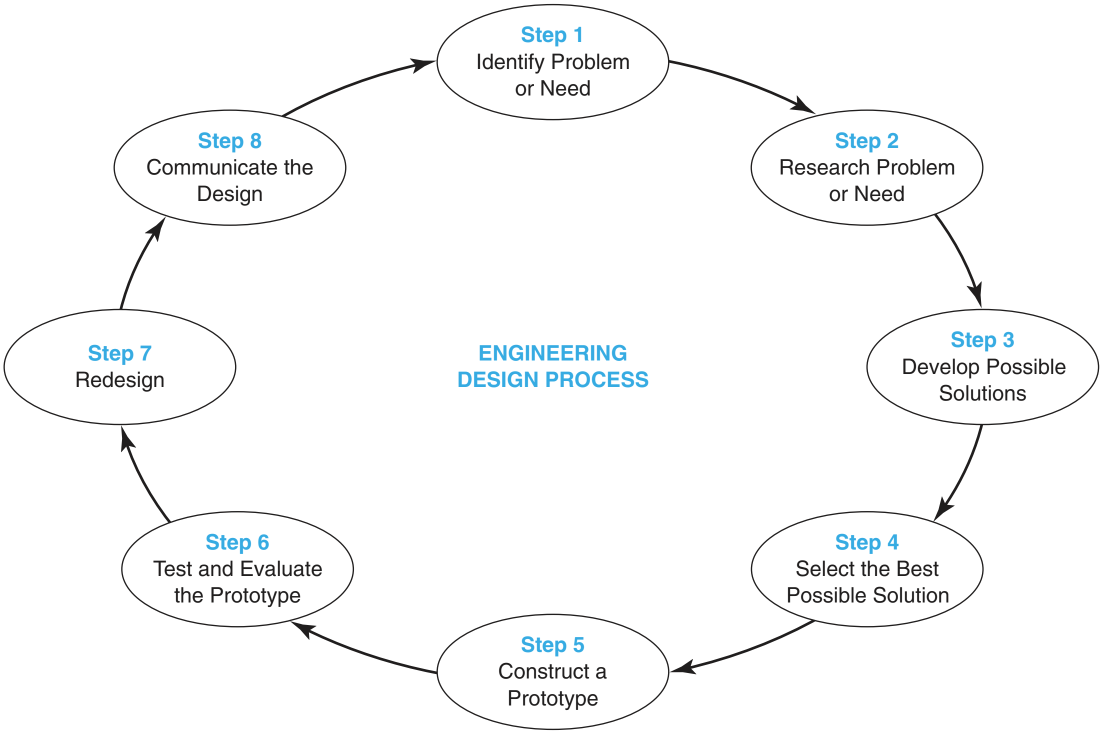


FIGURE 1.1
Engineering design process (EDP)

1.3 CAD Process

The CAD process is a subprocess of the EDP. We use it to implement Steps 5 through 8 of the EDP shown in Figure 1.1. The CAD process is carried out on a CAD/CAM system. In a general sense, use the CAD software to create 3D models of the part design, conduct analysis on the models, redesign the part, if needed, and document the final design. Figure 1.2 shows the CAD process.

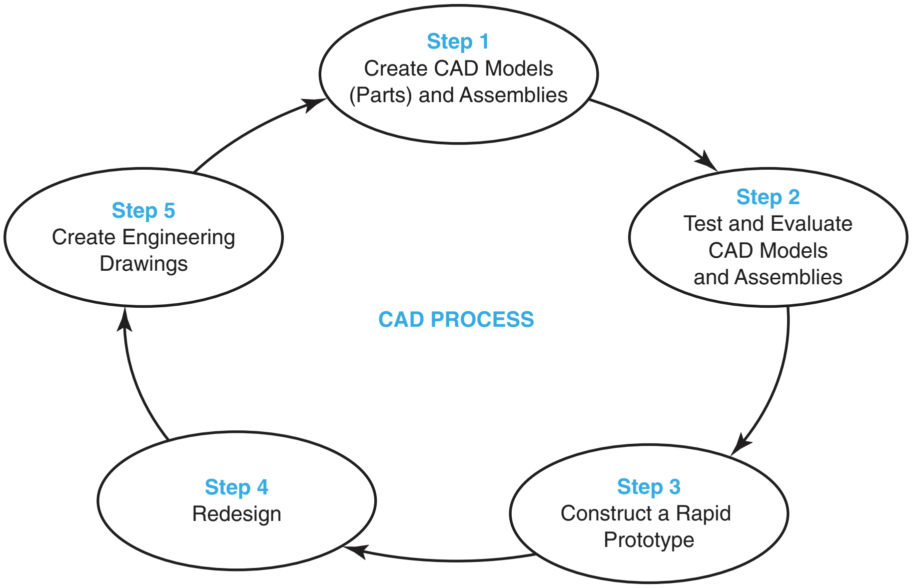


FIGURE 1.2
CAD process

1.4 Manufacturing Process

With the design complete, we manufacture it to produce the part or product (a product is an assembly of individual parts or components). The input to the manufacturing process is a design, and the output is the actual part or product that the design represents. Figure 1.3 shows the manufacturing process. It picks up where the design process left off:

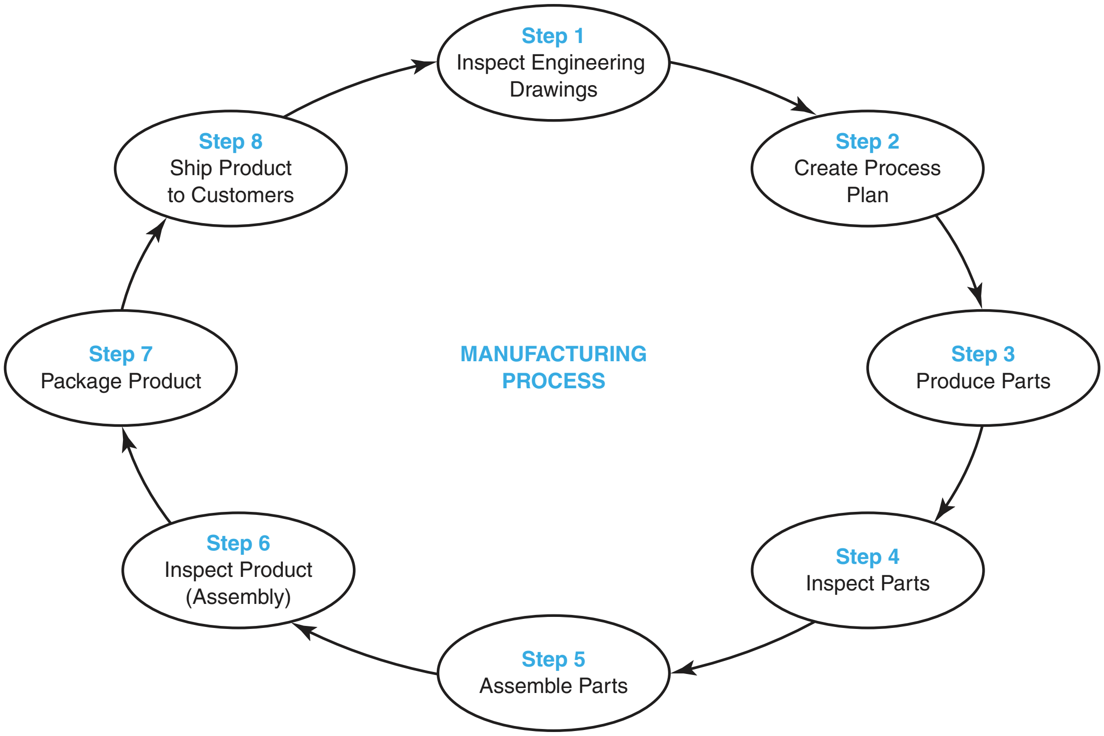


FIGURE 1.3
Manufacturing process

engineering drawings. A manufacturing engineer (known as a *process planner*) inspects the engineering drawings (Step 1 in Figure 1.3) for manufacturing purposes to ensure that all dimensions make sense, are not contradictory, and that all specified tolerances are producible in a cost-effective way. In Step 2, the manufacturing engineer creates the process plan to produce the part. This plan includes and coordinates all the details of production that the factory (shop floor) supervisors, foremen, and workers need to make the parts and the products. The output from the process plan includes lists of NC programs, production machines, tools, materials, routing sheets (that specify manufacturing sequences), cost estimate sheets, time estimates, and production floor schedule. The other manufacturing steps in Figure 1.3 are self-explanatory, and we all can relate to them.

1.5 CAM Process

The *computer aided manufacturing* (CAM) process is a subprocess of the manufacturing process. Like the CAD process, the CAM process is carried out on a CAD/CAM system. In a general sense, use the CAM software to create process plans, NC programs, and part inspection. Other manufacturing software exists, but it may not be part of the CAM software of a typical CAD/CAM system. Figure 1.4 shows the CAM process.

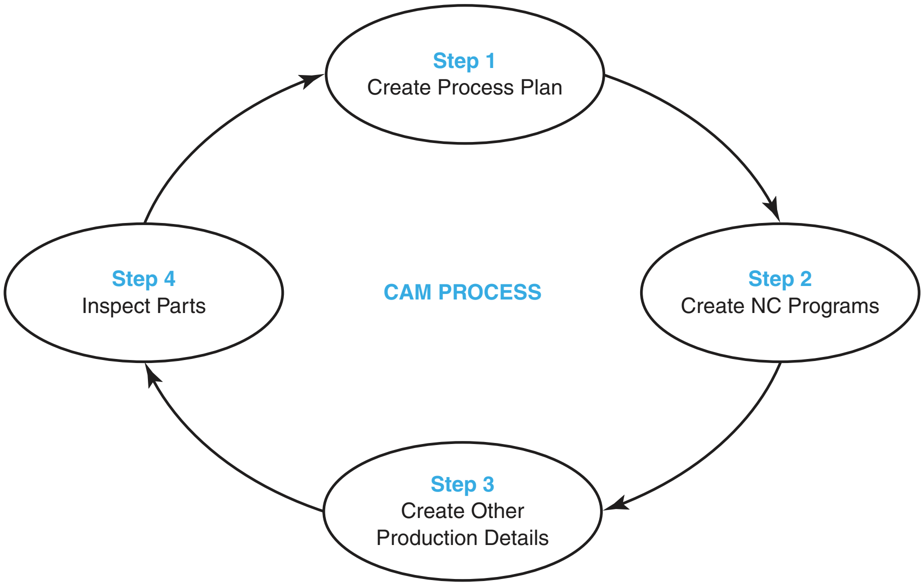


FIGURE 1.4
CAM process

1.6 SolidWorks Installation and Resources

While the book is written for SolidWorks 2014, it should work with other versions (e.g., 2012, 2013, or 2015). SolidWorks version affects, if at all, only the tutorials, not the concepts. Unless you are running a 32-bit version of the Windows operating system (OS), you should install the 64-bit version of SolidWorks. Click the following sequence to find the Windows version you are running: Right click **My Computer** or **Computer** (varies depending on the system) > **Properties.** If you do not see “x64 Edition” or “64-bit Operating System” under the **System** section, you are running the 32-bit version. If you install SolidWorks on your personal computer, you should make sure that the computer has enough memory (RAM), enough hard disk space, and the proper graphics (video) card for display. Check the SolidWorks website (www.solidworks.com/sw/support/SystemRequirements.html) for these requirements. Starting with SolidWorks 2015, SolidWorks will not install on a 32-bit operating system anymore. When you install SolidWorks, it creates shortcuts on your desktop. The ones we are interested in are SolidWorks (starts the program) and eDrawings (starts eDrawings).

SolidWorks is a native Windows application, meaning it does not run under other OSs such as Macintosh. If you need to run SolidWorks on a Mac, you must run Windows OS on the Mac computer. You have two options: Use BootCamp to partition your computer (for dual boot) to install Windows OS on one partition, or run virtualization software (e.g., Parallels or VMWare) that allows you to run a virtual machine (VM) copy of Windows from within the Mac OS. The latter option is not good because of the speed reduction of running SolidWorks, but it allows you to run Mac OS at the same time. BootCamp does not allow running both OSs at the same time.

SolidWorks comes with many resources to help you get started, troubleshoot, and learn. The **Help*** menu of SolidWorks is excellent and provides a wealth of information. To access this menu, click **Help** on the menu bar after you start SolidWorks. Other resources are available on SolidWorks website, www.solidworks.com. Visit the website frequently to find the latest news. Also, visit https://forum.solidworks.com to find answers to your questions. If you are interested in SolidWorks certifications, visit www.solidworks.com/cswa.

SolidWorks has useful software that is free. The two products of interest to us are eDrawings and SolidWorks Viewer. The eDrawings is an e-mail tool that allows you to communicate SolidWorks designs to anyone without them having to install SolidWorks software. It also allows you to mark up drawings you have been sent and then send them back. The SolidWorks Viewer allows you to view part and assembly files without having to install SolidWorks. It also allows you to pan (move), zoom in/out, and rotate the parts and assembly models for better visualization. Download both eDrawings and SolidWorks Viewer software from www.solidworks.com/sw/support/downloads.htm (once there, click FREE CAD TOOLS tab). Other useful SolidWorks resources for educators and students are listed here.

	Resource	Description and URL
Curriculum	SolidWorks Teacher Guides	Tutorials and projects: www.solidworks.com/curriculum
	SolidWorks Student Guides	Various resources: www.solidworks.com/curriculum
	Teacher Blog	Lessons developed by teachers for teachers: http://blogs.solidworks.com/teacher
	SolidProfessor	Videos and CDs of how to use SolidWorks: www.solidprofessor.com
Community	3D ContentCentral	Library of parts, assemblies, and macros: www.3dcontentcentral.com
	SolidWorks User Group Network	SolidWorks users groups and blogs: www.swugn.org
	SolidWorks Contests and Sponsorships	Various student competitions: www.solidworks.com/sponsoreddesigncontest
	SolidWorks Discussion Forum	Resource on specific product areas: http://forum.solidworks.com
	SolidWorks Blog Community	Network of SolidWorks users with their own blogs about SolidWorks: www.solidworks.com/sw/communities/read-solidworks-blogs.htm

(*continued*)

*Names of SolidWorks menus, menu items, tabs, toolbars, windows, and keyboard keys are shown in bold with first letter in cap. Sketch names, features names, and file names are shown in italic.

	Resource	Description and URL
Others	CADJunkie	Various tutorials and blogs: www.cadjunkie.com
	YouTube Video	www.youtube.com/results?search_query=solidworks
	CB Model Pro	Free-form surface modeling tool: www.cbmodelpro.com
	SolidWorks Labs	Labs: http://www.solidworks-lab.com
	SolidWorks Design Gallery	Model gallery: www.solidworks.com/pages/successes/gallery/model_gallery.html
	SolidWorks White Papers	Papers: www.solidworks.com/pages/services/WhitePaper.html

1.7 SolidWorks Overview

Let us run SolidWorks and open a new part file to provide an overview of SolidWorks window (user interface). To start SolidWorks, double-click its shortcut icon (or follow your CAD lab instruction). Then click this sequence to open a new part file: **File > New > Part > OK**. Figure 1.5 shows SolidWorks main window after you open a new part. Hover over any part of the window with the mouse, and a tooltip with information will pop up. If you want to see the hover effect on the menu bar, click anywhere on the bar

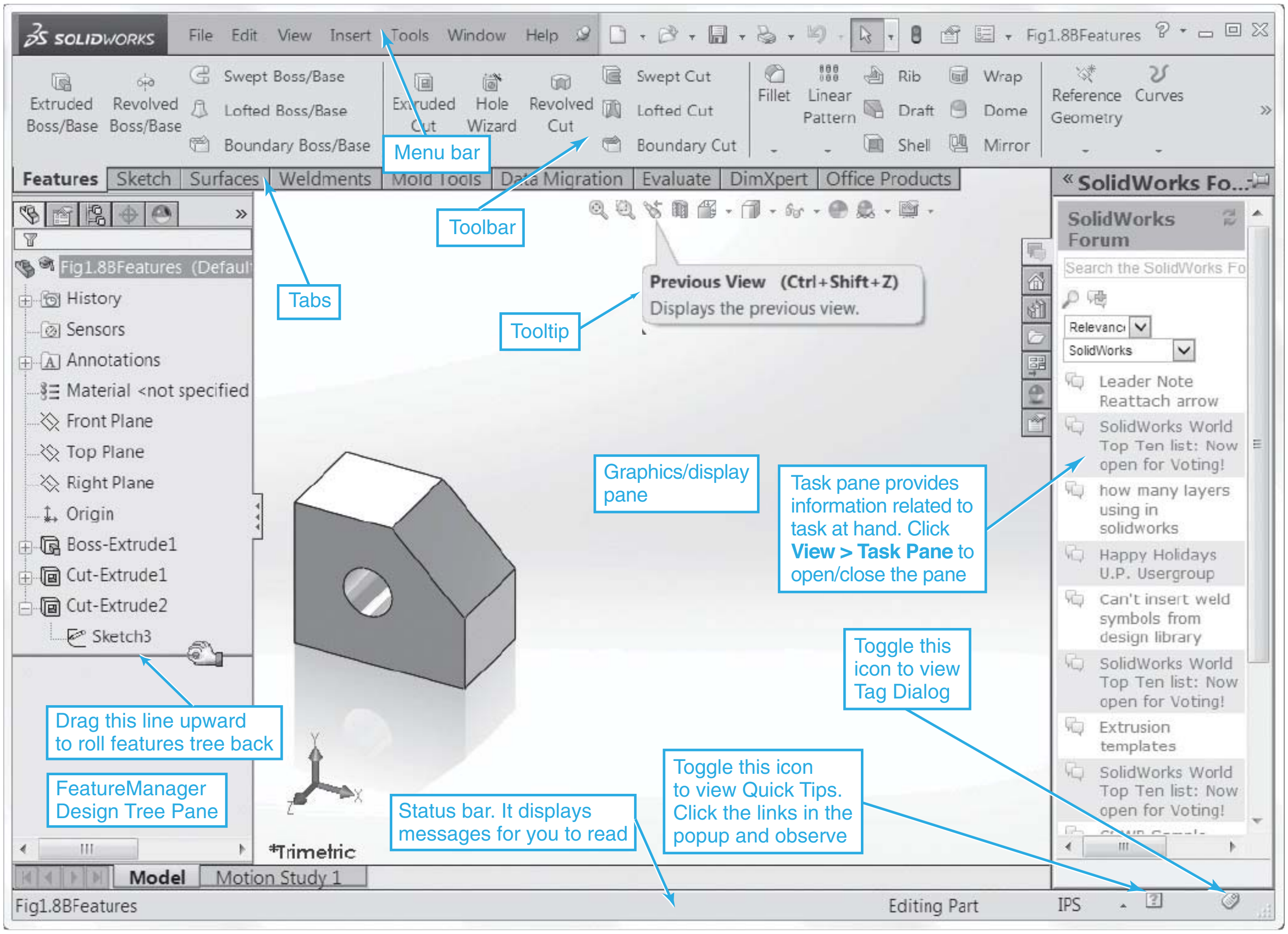


FIGURE 1.5
SolidWorks main window

first to add the focus there. The main window has three panes (areas), as shown in Figure 1.5. Familiarize yourself with the window and the panes. Although most of the information in Figure 1.5 is self-explanatory, the status bar at the bottom of the window displays messages related to your current activity. Make a habit of watching this bar.

SolidWorks offers multiple modes, depending on what you would like to do. The three basic modes are Part (create parts or components), Assembly (create assemblies), and Drawing (create drawings). Other modes include Simulation, Animation, Analysis, and Machining. SolidWorks displays the commands for each mode when you activate it.

One important goal of using a CAD system is to be efficient and quick in achieving design tasks. We offer the following general tips for the basic repetitive SolidWorks tasks:

Task	Instructions or Command sequence to click to achieve task
Creation steps	We use the verbose approach to describe a SolidWorks procedure or skill the first time we use it. After that we use a shorter version.
Using examples and tutorials	We encourage you to use the book with SolidWorks software open at the same time, so you can follow the sequence of commands and also explore more on your own.
Mouse buttons	Use the left button for selections as you always do. Use the right button to open a context (selection) menu. *Click* means click the left mouse button. If we need a mouse right click, we will write so explicitly. We use *select* and *click* interchangeably.
Mouse wheel	Rotate the wheel to zoom in and out of part. Press and hold it and drag (move) the mouse to rotate the part on screen.
Multiple entity selection	Click the left mouse button to select the first entity, then press and hold the **Ctrl** key on keyboard to select other entities, or select by window. Press and hold the mouse left button and drag (move) the mouse to define window. Any entities fully enclosed by the window are selected. Any partially enclosed entities are not selected.
Exit current creation mode	Hit the **Esc** key on keyboard to terminate sketching or any current mode.
Save, abort, undo symbols	These three symbols  mean save (**OK**), abort (**Close**), and **Undo** respectively. When you see them, use the one you need.
 and 	These sketch symbols are shown on the upper right corner of the sketch screen. The first symbol means save changes and exit the sketch. The second symbol means do not save changes made to the sketch and exit it.
Delete entities	Use the **Delete** key on the keyboard. Do not use the **Backspace** key or the **Delete** key on the Num pad. They do not work.
Enable/disable snap	If you lose snap to endpoints or midpoints of entities, click this sequence to enable it: **Tools** (on menu bar) **> Options > System Options** (tab) > **Relations/Snaps** > **Enable Snapping** > **OK**
Capture graphics screen	Press **Alt + Print Screen** keys on keyboard. This sequence captures the entire SolidWorks window; or, click this sequence to capture the content of the graphics pane (Figure 1.5): **View** > **Screen Capture** > **Image Capture**
Turn on/off Task Pane	**View** > **Task Pane** (Figure 1.5)
Features tree	SolidWorks calls it **FeatureManager Design Tree**. We call it features tree for simplicity. It is shown in the left pane of SolidWorks window as shown in Figure 1.5. This tree is also known as the part history tree. It stores a sequential record of the part creation steps. It has nodes; each node represents a feature. Read the nodes from the top down. If you expand a node (click its "+" sign), you see the sketches and operations needed to create the feature. Also, you can drag the bottom blue line of the tree (called rollback bar) upward to roll it back to hide features displayed temporarily in the graphics/display window. This is useful to trace back (reverse engineer), investigate, and debug part creation.

(*continued*)

Task	Instructions or Command sequence to click to achieve task
Deleting a feature	To delete a feature, right click its node in the features tree > **Delete**. Deleting a feature does not delete its sketch. Conversely, you cannot delete a sketch before deleting its feature. SolidWorks would not allow it.
Deleting sketch entities	To delete one entity (e.g., a line), select it with the left mouse button and hit the **Delete** key on the keyboard. To delete multiple entities, select them by pressing the left mouse button and dragging the mouse to highlight the entities, then hit the **Delete** key. If entities are scattered, press the **Ctrl** key on the keyboard and select the entities, one by one, using the left mouse button, then hit the **Delete** key.
Handle multiple open windows of SolidWorks	If any of these open windows does not stay up when you hover over it, it means that an edit feature operation is open in one of the windows. Locate that window and close the edit operation.
Display sketch entities	Sometimes a sketch becomes hidden after deleting a feature. If this happens click **View** > **Hide All Types**. This is a toggle (on/off). Or, click sketch node in features tree > **Show** (eyeglasses symbol) from context toolbar that pops up.
Control font size of dimensions in a drawing	If font size for dimensions in drawing views is too small or too big: **Tools** > **Options** > **Document Properties** tab > **Dimensions** (click it; do not expand) > **Font** > **Points** > Select size > **OK** > **OK**
Geometric relation symbols	While sketching entities in a sketch, the mouse cursor may show a symbol to indicate one of the relations shown in Figure 1.6.
Rendering and viewing parts	You have different options to render the part as shown in Figure 1.7A. Hover over any rendering icon to read it. You can also view the part in different views while modeling. Figure 1.7B shows SolidWorks **Heads-up View** toolbar. Hover over any view icon to read it. Also, click the **Zoom to Area** icon, select an area of the part by pressing and dragging the mouse left button to enlarge it to inspect overlapping entities, and check other modeling details. This is useful in debugging modeling errors. Once finished, click the **Zoom to Fit** icon to fit the entire part back to the graphics/display screen. You can also split the modeling screen to view different views of the part as shown in Figure 1.7C.

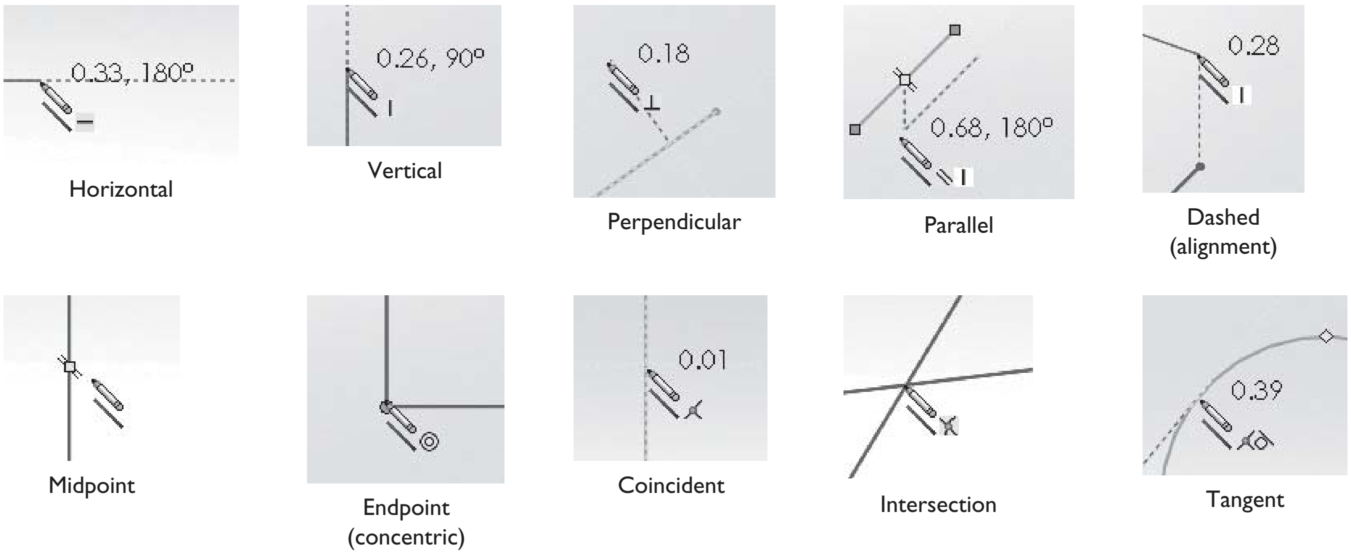
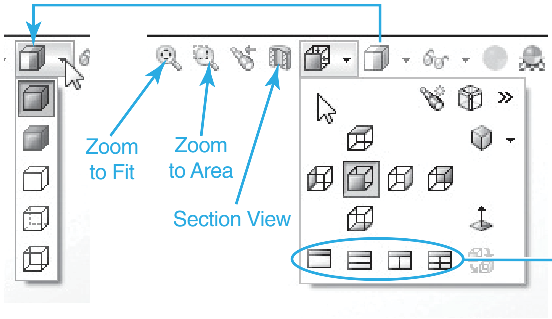
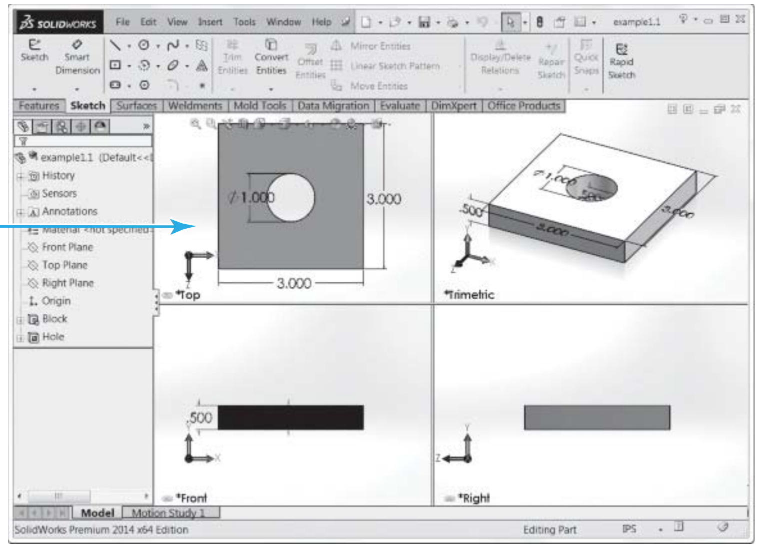


FIGURE 1.6
Geometric relation symbols



(A) Part rendering options

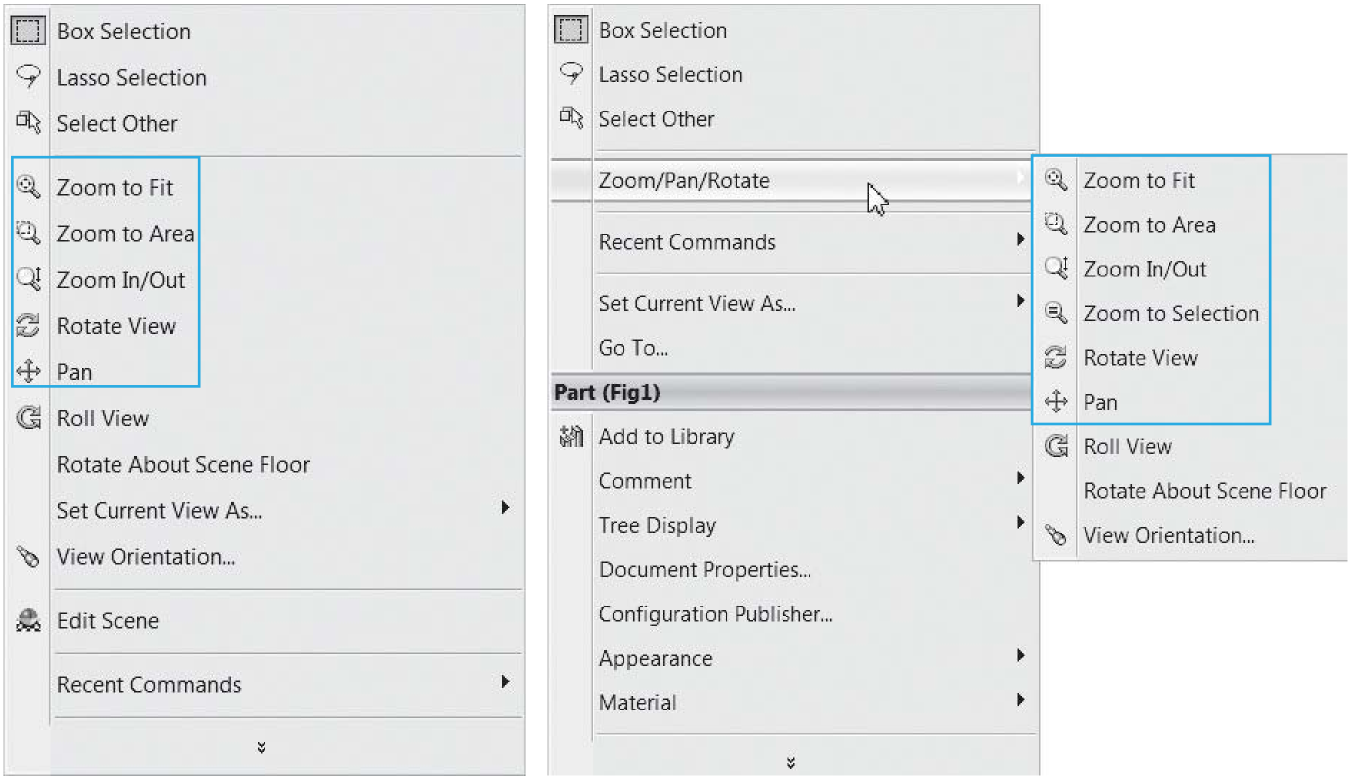
(B) Part standard views



(C) Split modeling window

FIGURE 1.7
Rendering and viewing parts

Task	Instructions or Command sequence to click to achieve task
Zoom/Pan/Rotate	Use these functions during modeling to inspect and view the part or move it on the screen. Figure 1.8 shows the two options of the pop-up window that has these functions. Figure 1.8A shows the window when no nodes in the features tree are selected. Figure 1.8B shows the window when any node is selected or you are in the edit mode of a sketch or a feature. Right click on the graphics/display pane to open either pop-up. Then click the function you need to use. To zoom, select the zoom type as shown in Figure 1.8. To rotate, press and hold the mouse wheel and drag (move) the mouse to rotate the part on screen. To pan, drag the part around the screen to position as needed.
Toggle multiple open parts	If you open multiple part files in one session, click **Window** (menu) > select the part you want to display from the bottom of the popup. Alternatively, click **Window > Cascade** to display all open parts at the same time, each in its own window. Maximize any window to exit the cascade mode.
Place toolbars	If you double click any menu bar, it gets unlocked and floats on screen. To place it back at the top of the screen, drag it, place it over the arrow that shows up on top middle of the screen, and drop it there.
Toggle displaying radius or diameter dimension of a circle or arc	After inserting the dimension of circle or arc, right click it > **Display Options** from popup > **Display As Radius** (if a diameter is displayed) or **Display As Diameter** (if radius is displayed).
Task Pane has moved to the bottom right corner of the screen. How to get it back to default position (right side of screen)?	Drag it and drop it onto the SolidWorks screen until it snaps to the right side of the screen.



Option A Option B

FIGURE 1.8
Zoom/Pan/Rotate functions

1.8 Customize SolidWorks

SolidWorks can be customized in many ways, but the basic customization needs are available when you click **Tools > Options** to access a window with two tabs: **System Options** and **Document Properties**. Customizing at the system level (using **System Options)** affects current and future documents (i.e., global effect). Customizing at the document level (using **Document Properties**) affects the current document only (i.e., local effect). Familiarize yourself with the choices available for each option.

Another, but related, customization way is to open a new blank part. Then, use the **Document Properties** tab to customize annotations, dimensions, part units, and so on. After you are done, save the part file as *master.sldprt* for future use. When you want to create a new part (CAD model), open *master.sldprt*, create a new design and save it under a new name. Repeat for future parts.

A third way to customize is to change the part, assembly, and drawing file templates of SolidWorks itself, instead of using the *master.sldprt* idea. SolidWorks comes with template files with the names *Part.prtdot*, *Assembly.asmdot*, and *Drawing.drwdot*. If you open any of these files and customize them using the **System Options or Document Properties** tab and save them, the changes apply to future usages. For example, if you change the units from IPS to MMGS in the *Part.prtdot* file, each time you open a new part, it uses these units. To edit any template click this sequence: **File** > **Open** > **Template (*.prtdot, *.asmdot, *.drw.dot)** under **Quick Filter** > Select the file template you like to edit > **Open** > **Tools** > **Options** > Use the **System Options** or **Document Properties** tab to customize as needed > **OK** > **File** > **Save**.

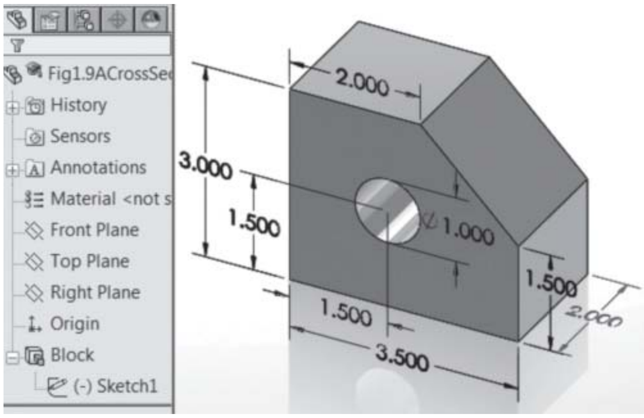
1.9 Modeling Plan

Quite often, CAD users (designers) tend to rush to create CAD models (parts) without some critical thinking and strategizing about the best modeling plan to create the models. (We use the terms *model* and *part* interchangeably in the book; the two words are synonymous.) We measure "best" by multiple factors:

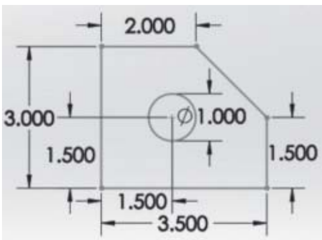
1. Fastest time to create the part
2. Most efficient way to edit the part after creation
3. How the part will be manufactured

Generally speaking, the two common modeling approaches are the cross-section or the features approach. The cross-section approach is generally faster than the features approach, but may be inefficient in editing parts later on. Within each approach, some modeling plans are more efficient than others. To help grasp the importance of a modeling approach and a modeling plan within the approach, think of traveling from point A to point B and ask yourself: What is the best and fastest way to get from point A to point B?

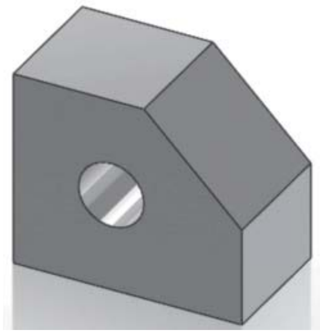
Consider the chamfered block with a hole shown in Figure 1.9 to discuss these concepts further. Figure 1.9A shows the cross-section modeling plan to create this part. It is easy, consisting of two steps: Create the cross section sketch, and extrude it to create the part. Figure 1.9B shows the features modeling plan. It has three steps, creating a feature in each step: Create the block, create the chamfer, and create the hole. The order of creating the chamfer or the hole is irrelevant.



- Part and its modeling plan (shown by features tree)
- We renamed the feature node to *Block* from its default name (*Boss-Extrude1*) for ease of tree read.
- The *Block* node is shown in its expanded form. Click the "-" to collapse it.



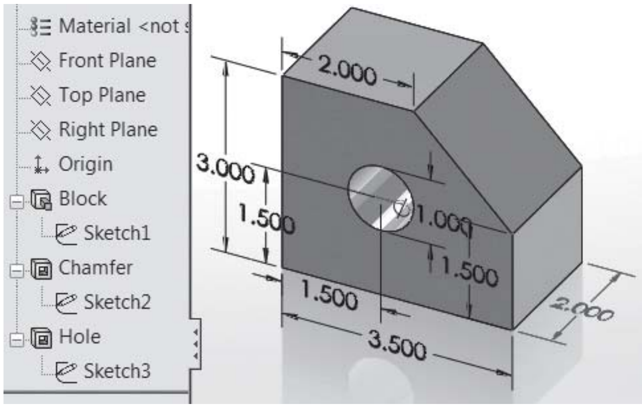
Step 1: Create *Sketch1* of *Block* cross section.



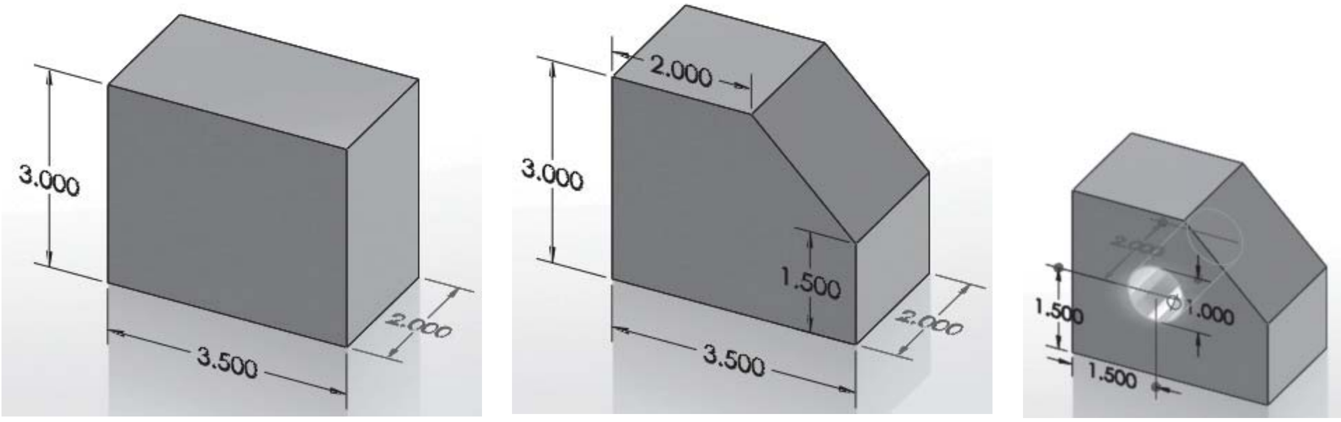
Step 2: Create *Block* feature: Extrude *Sketch1* 2.0 inches.

(A) Cross-section modeling plan

FIGURE 1.9
Modeling plan of a chamfered block



- Part modeling plan is shown in the features tree.
- Create three features in the order shown.
- We renamed the features nodes in tree as shown.



Step 1: Create *Sketch1* and *Block* feature: Sketch the 3 in. × 3.5 in. rectangle on **Front Plane** sketch plane and extrude 2 in.

Step 2: Create *Sketch2* and *Chamfer* feature: Use **Chamfer** feature on **Features** tab. Use 1.5 in. for chamfer distance.

Step 3: Create *Sketch3* and *Hole* feature: Sketch circle with diameter = 1.0 in. on *Block* front face and extrude cut 2 in.

(B) Features modeling plan

FIGURE 1.9
(*continued*)

Which is a more efficient plan to create this part? That is the heart of the debate. Considering the part creation only, the cross-section plan is generally faster. The features plan is always more modular and mimics the manufacturing steps in the machine shop. The machinist starts with block stock, mills out the chamfer, and drills the hole.

This debate is always settled based on heuristics in practice and personal preferences. Some CAD designers, especially those who have manufacturing background, prefer the features modeling approach. In general the cross-section approach is faster than the features approach. We use both approaches in this book depending on the modeling task at hand. In this chapter, though, we use only the features approach for learning purposes by creating features repetitively.

1.10 Part Creation

After deciding on a modeling plan of a part, you execute the plan on a CAD/CAM system to create the CAD part. A CAD part is made up of features that are combined in the order of creating them. A CAD/CAM system stores this order in a features tree that it creates.

Each time you create a new feature, the system adds or subtracts it from the features that are already created. Therefore, at any one time you have only one combinatory feature representing the CAD part. Figure 1.9B shows three features that make up the part. The tree shows that the user created the *Block* (originally named *Boss-Extrude1* by SolidWorks), followed by creating the *Chamfer* (*Cut-Extrude1*), followed by creating the *Hole* feature (*Cut-Extrude2*).

A **feature** is a 3-dimensional (3D) CAD part such as a block, cylinder, hole, and so on. The two basic features are extrusion and revolve. An **extrusion** is a prismatic feature that has a constant cross section and a uniform thickness perpendicular to the cross section plane. The *Block* shown in Figure 1.9B is an extrusion. A **revolve** is an axisymmetric feature that also has a constant cross section and an axis of revolution to revolve the cross section around.

The creation of a feature begins with a 2D sketch to draw its cross section. You draw the geometry of the cross section (sketch) on a sketch plane. A **sketch plane** is a predefined plane that you select or a plane that you create**.** Each of the three steps of Figure 1.9B shows the sketch of each feature. After creating the sketch, you use it to create the feature. SolidWorks provides a **Sketch** tab and a **Features** tab as shown to the far left of Figure 1.5**.** You can start creating a feature using either tab. If you start with the **Sketch** tab, you would need to select a feature from the **Features** tab after finishing and exiting the sketch because SolidWorks would not know which feature you want to create using the sketch. If you start with the **Features** tab, you select the feature you want to create. SolidWorks automatically activates the **Sketch** tab because the next logical step is to create the feature sketch. After you are done with the sketch, SolidWorks goes back to the features mode and lets us finish the feature creation. Either way to create a feature is fine. We prefer the **Features** tab, as we feel it is more efficient. Thus, we start with the **Features** tab in this book.

1.11 Examples

This chapter covers examples and tutorials of the basic tasks of SolidWorks: creating parts, assemblies, and drawings in a quick fashion for those readers who prefer to do it all quickly. More in-depth coverage of these design activities is provided throughout the book. The examples cover creating basic simple parts, assemblies, and drawings. The tutorials cover more involved parts.

Example 1.1 Create the plate shown in Figure 1.10. All dimensions are in inches.

Solution Figure 1.10 shows the plate modeling plan. The part is an extrusion with a hole in its center.

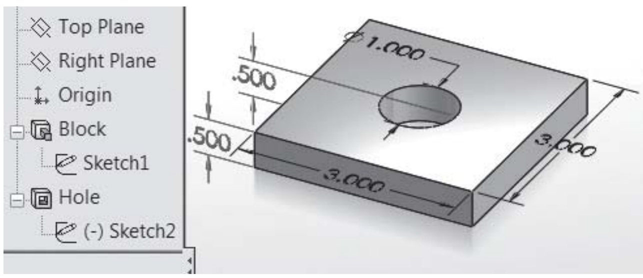
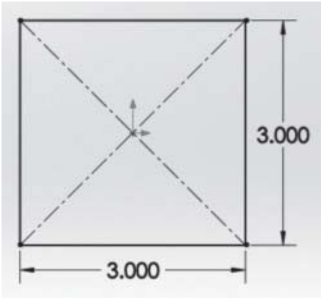
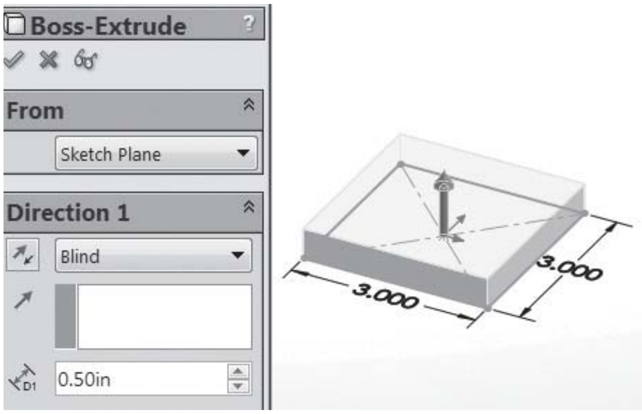


FIGURE 1.10
Plate modeling plan shown in its features tree

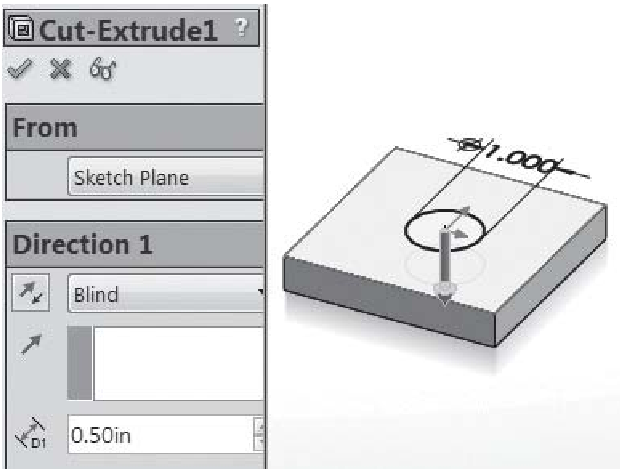
Step 1: Create *Sketch1*: **File** > **New** > **Part** > **OK** > **Top Plane** (sketch plane) > **Extruded Boss/ Base** on the **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and drag to sketch > **Smart Dimension** on **Sketch** tab > click a side and move tool (mouse) to place dimension > click to release > enter 3.0 in the box that pops up > hit **Enter** on keyboard or click ✔ > repeat for other side > click the **Exit Sketch** icon on far top right screen to save and exit the sketch.



Step 2: Create *Block* feature and save part: When you exit the sketch, you see screenshot below. Enter 0.5 for thickness (**D1**) > click ✔ to finish. SolidWorks creates a node in the features tree called *Boss-Extrude1*. Click it once and rename it *Block*. Save the part as *plate:* **File** > **Save As** > *plate* > **Save.**



Step 3: Create *Sketch2* and *Hole* feature: Click the top face of *Block* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > click origin and drag to create circle > **Smart Dimension** > click circle > move mouse to place dimension and click to release > enter 1 in the pop-up box > hit **Enter** on keyboard > exit sketch > Enter 0.5 for thickness (**D1**) > click ✔ to finish. SolidWorks creates a node in the features tree called *Cut-Extrude1*. Click it once and rename it *Hole*.



Example 1.2 Create the pin shown in Figure 1.11. All dimensions are in inches.

Solution Figure 1.11 shows the pin modeling plan. The part is a revolve feature because it is axisymmetric.

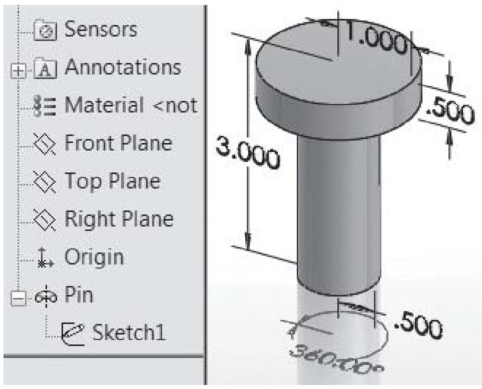
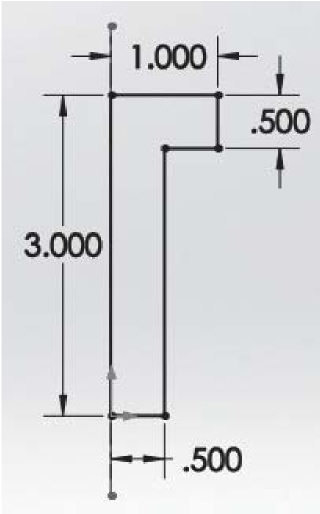


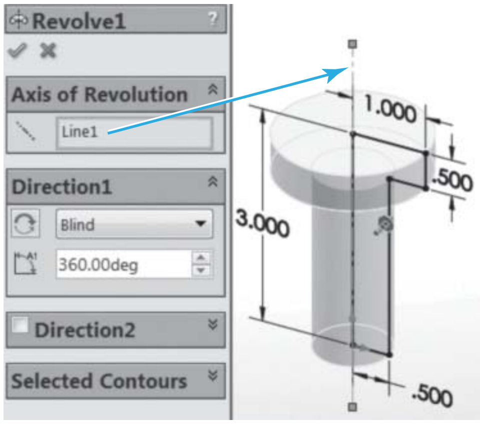
FIGURE 1.11
Pin modeling plan shown in its features tree

Step 1: Create *Sketch1*: **File** > **New** > **Part** > **OK** > **Front Plane** (sketch plane) > **Revolved Boss/ Base** on **Features** tab > **Line** dropdown on **Sketch** tab > Centerline > sketch a vertical centerline passing through origin as shown to the right > **Line** on **Sketch** tab > sketch the cross shown to the right (follow horizontal and vertical directions) > dimension the cross section as shown to the right (see Example 1.1 for detailed steps to dimension a sketch) > click the **Exit Sketch** icon on far top right of screen to save and exit the sketch.



Step 2: Create *Pin* feature and save part: When you exit the sketch, you see screenshot to the right. Click ✔ to finish. SolidWorks creates a node in the features tree called *Revolve1*. Click it once and rename it *Pin*. Save the part as *pin:* **File** > **Save As** > *pin* > **Save.**

Note that you need an axis of revolution and a closed cross section to create a revolve. You rotate the cross section about the axis of revolution an angle as shown below.



Example 1.3 Create the base plate shown in Figure 1.12. All dimensions are in inches.

Solution Figure 1.12 shows the base plate modeling plan. The part is an extrusion with a step cut and a hole.

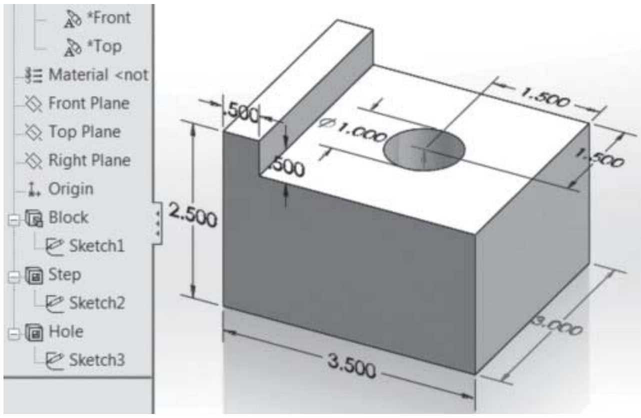
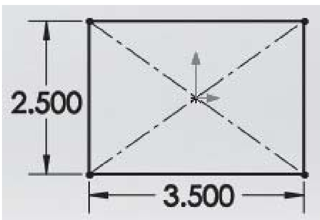


FIGURE 1.12
Base plate modeling plan shown in its features tree

Step 1: Create *Sketch1*: **File** > **New** > **Part** > **OK** > **Front Plane** (sketch plane) > **Extruded Boss/ Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and drag to sketch > dimension the cross section as shown to the left > exit sketch.

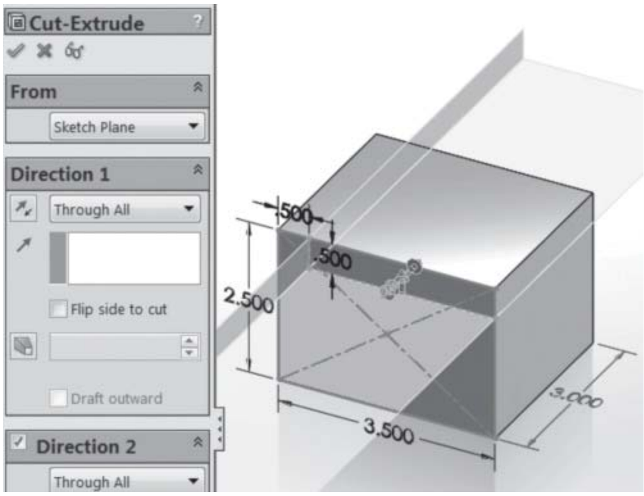


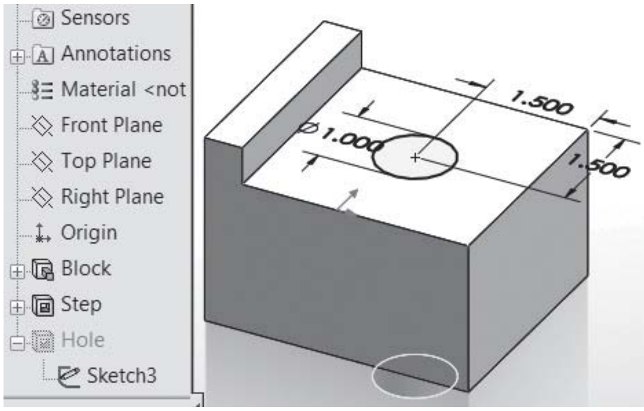
Step 2: Create *Block* feature and save part: Exit the sketch, enter 3.0 for thickness (**D1**) > reverse the extrusion direction (drag extrusion arrow in the graphics pane) > click ✔ to finish. SolidWorks creates

a node in the features tree called *Boss-Extrude1*. Click it once and rename it *Block*. Save the part as *basePlate:* **File > Save As >** *basePlate* **> Save.**

Step 3: Create *Sketch1* and *Step* feature: Click the front face of *Block* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Line** on **Sketch** tab > sketch a vertical line and a horizontal line as shown below (make sure line ends snap to edges of existing face of *Block* feature (Step 1) > dimension sketch as shown below > Exit sketch > click ✔ to finish. SolidWorks creates a node in the features tree called *Cut-Extrude1*. Click it once and rename it *Step*. This extrusion is **Through All**, so it cuts the Block feature.

Step 4: Create *Sketch3* and *Hole* feature: Click the top face of *Step* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > sketch a circle as shown below > dimension sketch as shown below > exit sketch > enter 2.5 for thickness (**D1**) > click ✔ to finish. SolidWorks creates a node in the features tree called *Cut-Extrude1*. Click it once and rename it *Hole*.





Example 1.4 Create engineering drawings (see Figure 1.13) of the parts created in Examples 1.1–1.3.

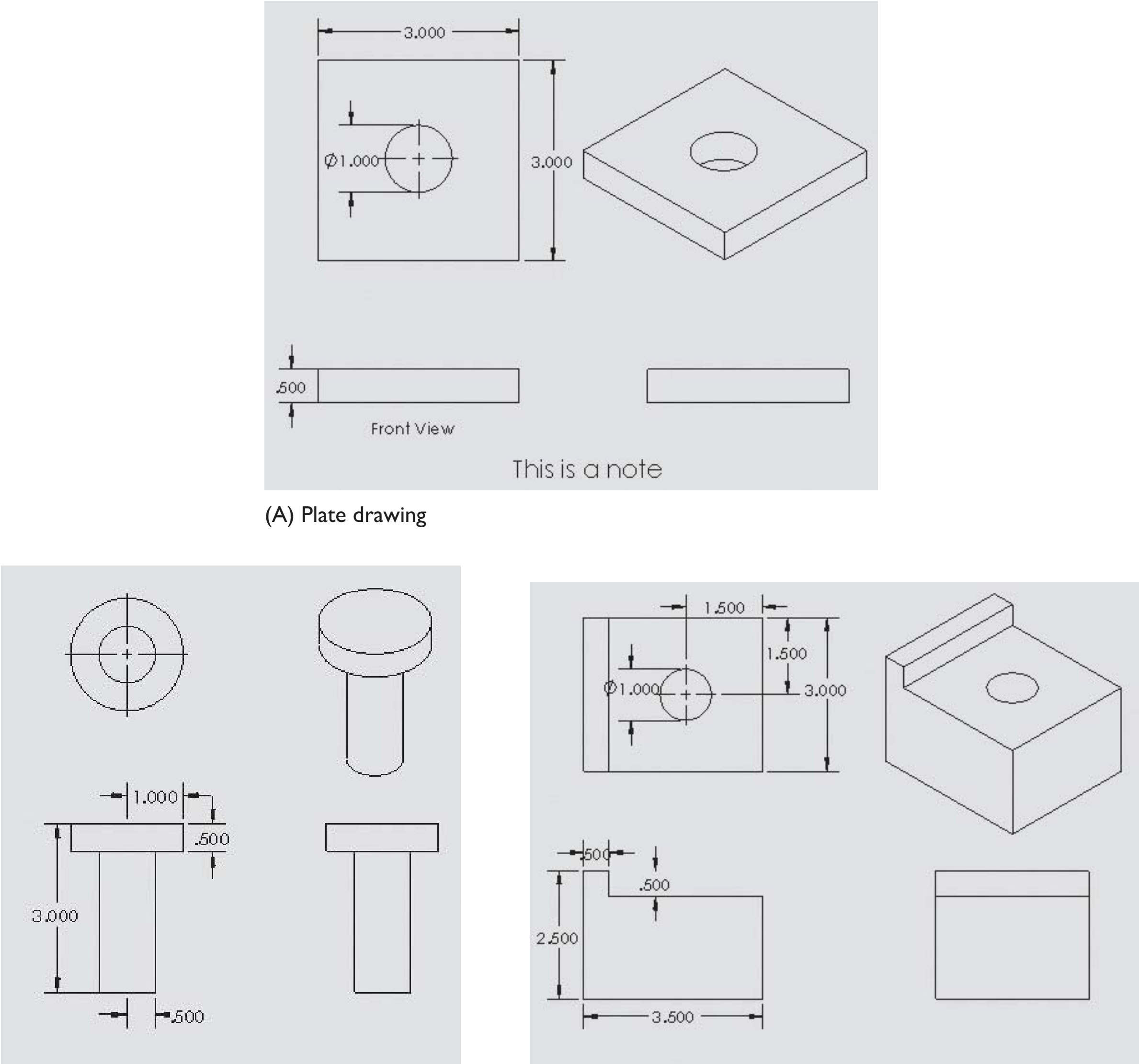
Solution We create three engineering drawings of the three parts. We insert the three typical views (front, top, and right) into each drawing and dimension each view. We also insert an isometric view in each drawing. In practice, some companies use the ISO view; others do not.

There is a two-way associativity between parts and their drawings. Changing a part is reflected in its drawing views automatically, and, vice versa, changing the part (model) items in a drawing is reflected automatically in the part.

Open SolidWorks in the Drawing mode to create a drawing: Click **Open > New > Drawing > OK > OK** (accept the default **Standard sheet size**). SolidWorks drawings use the concept of sheets. You can create multiple sheets within a drawing. We use only one sheet per drawing. Once the drawing is open, SolidWorks offers three methods to dimension a drawing: individual, auto, and model items. The first two methods are available under the **Smart Dimension** icon in the **Annotation** tab. When you click the **Smart Dimension** icon, two tabs show in the left pane on the screen: **DimExpert** and **Autodimension**. Use **DimExpert** to insert individual dimensions, and use **Autodimension** to dimension an entire view at once.

The model items method is available as the **Model Items** icon in the **Annotation** tab. We prefer using this method because it preserves the model-drawing associativity concept. SoildWorks displays the layout of the sketch dimensions in a drawing view. For example, if you have created the sketch in the **Front Plane**, the sketch becomes the front view and will be dimensioned the same way you dimensioned the sketch.

SolidWorks requires an active part to feed into the drawing you are creating. You may open the part before or after you open a new drawing. With this quick overview of drawings, we create three drawings for the three parts as shown in Figure 1.13.



(A) Plate drawing

(B) Pin drawing

(C) Base plate drawing

FIGURE 1.13
Engineering drawings

Step 1: Create the drawing and views: **File** > **New** > **Drawing** > **OK** > **OK** (accept the default **Standard sheet size**) > **Browse** (locate and select the *Pin* part) > click the mouse near the bottom left corner of the drawing sheet to insert the front view > move the mouse up and click to place the top view > move the mouse near the bottom right corner and click to place the right view > move the mouse near the top right corner and click to place the ISO view > click the mouse right button to finish.

Note: Each view has a bounding box (hover near the view until you see it). You can drag any edge of the box to adjust the view placement in the drawing sheet. SolidWorks maintains the orthogonality between the three views (front, top, and right). If you move one of them, the other follows accordingly.

Note: Click this sequence to change angle of projection if you get the wrong views: Right click anywhere on the drawing sheet > **Properties** > **Third angle** from window that pops up > **OK**

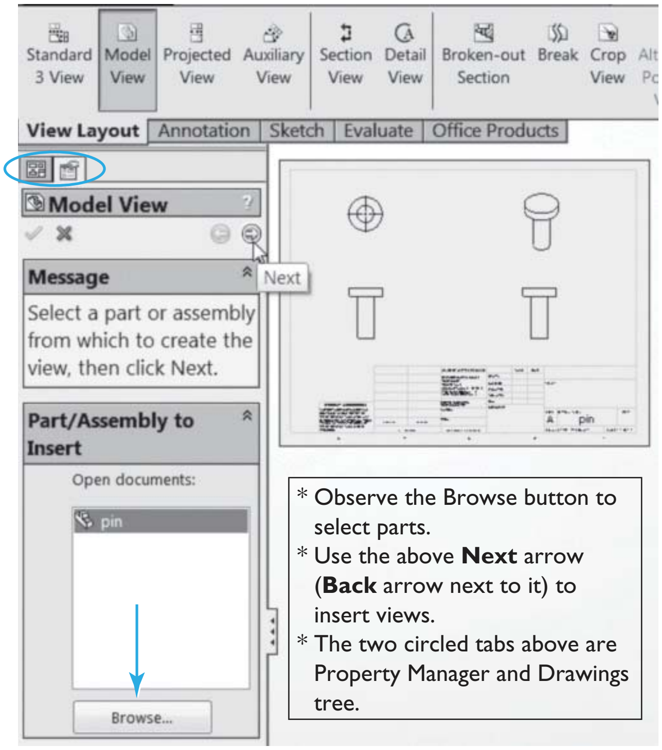
Step 2: Create dimensions, notes and save drawing: To insert dimensions click **Annotation** tab shown to the right **> Model Items >** ✔. To insert a note like the one shown in Figure 1.13A, click **Annotation** tab > **Note** > click in drawing sheet to place note > type note text > **Esc** > **Esc**

Save the drawing as *pinDrawing*: **File > Save As** > *pinDrawing* > **Save**

Note: Repeat Steps 1 and 2 to create drawings of other parts.

Note: SolidWorks opens the **Model Items** pane to the left of the screen when you click **Model Items** icon. Investigate its offerings to learn more.

Note: If needed, drag dimensions or notes to reposition them relative to views.



Example 1.5

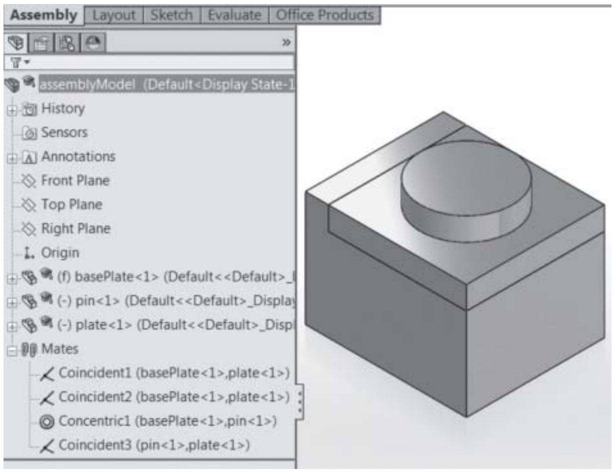
Create the assembly model (see Figure 1.14) of the parts created in Examples 1.1–1.3. Also, create an assembly drawing, an animated exploded view, and a rendered view of the assembly.

Solution You create an assembly by inserting the individual parts (components) of the assembly into the assembly model. (We use the terms *part* and *component* interchangeably.) The first part you insert is considered the base part onto which other parts are assembled. To open SolidWorks in the Assembly mode, click **File > New > Assembly > OK**. When you insert a part into the assembly model, SolidWorks actually makes a copy of the part and inserts it. This copy is known as an *instance*. Thus, you can insert as many instances of a part as the assembly needs.

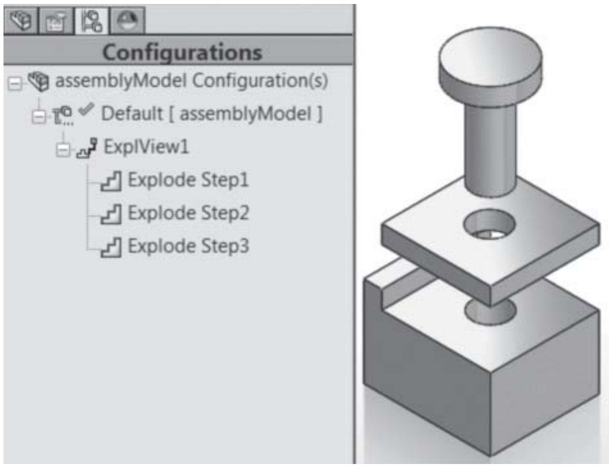
After you insert the parts, you assemble them to create the assembly. Assembling a part in an assembly entails applying mating conditions (mates) to constrain it in the assembly. A mate positions and orients two components relative to one another. Conceptually, the mates resemble assembling parts of a product in real life. A mate applies to two entities of the parts being mated, be it a face, edge, point, axis, origin, sketch plane, or plane. For example, a mate may require two faces to be coincident, two cylindrical faces to be concentric, or two planes to be parallel.

The outcome of mating two parts is to prevent them from moving relative to each other in space, thus mimicking real life. An assembled part must be fixed (anchored) in space unless it is free to move for functional requirements; for example, a motor shaft must rotate. In general, a component needs three mates to lock it in 3D space. There is a total of six degrees of freedom (DOF) in the assembly modeling space: three translations along the three axes (*X*, *Y*, and *Z*) and three rotations around these axes. Three mates are what it takes to constrain the six DOF because locking one locks another.

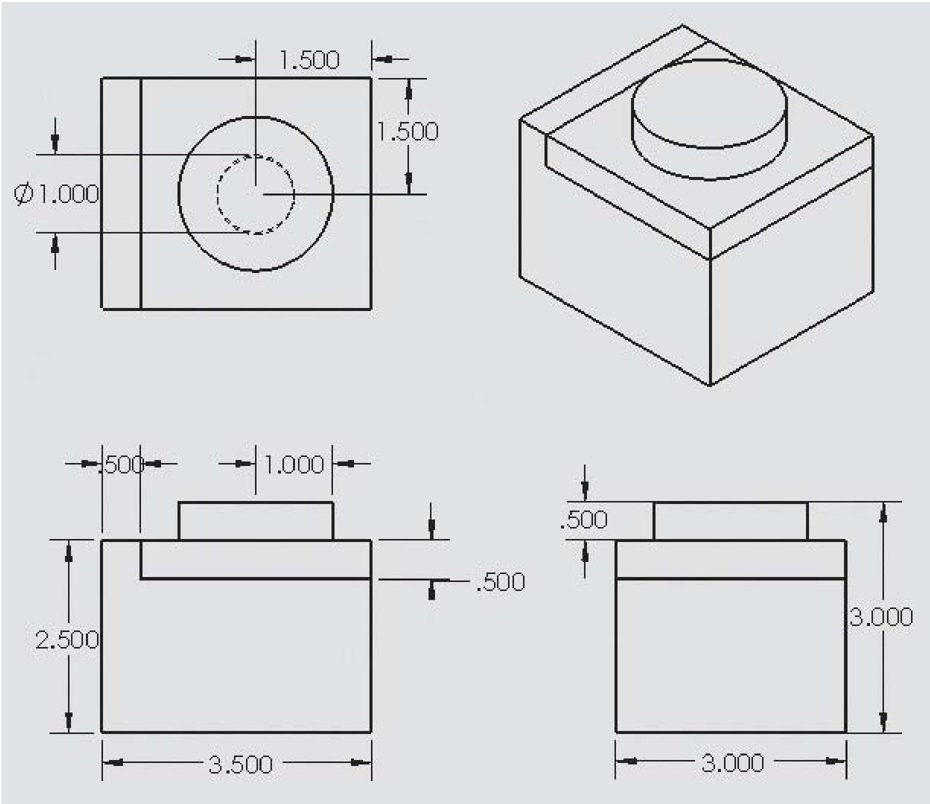
There is a two-way associativity between parts and their assemblies. Changing a part model is reflected in all its instances automatically, and vice versa, changing the instance in an assembly is reflected automatically in the part model.



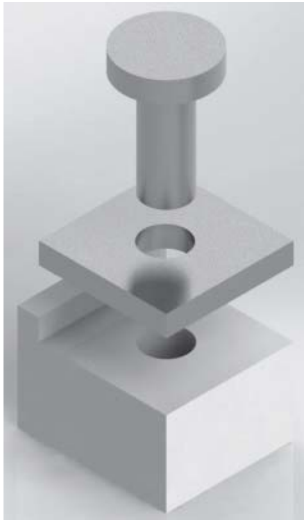
(A) Collapsed assembly view



(B) Exploded assembly view



(C) Assembly drawing



(D) Rendered assembly model

Note: Assign material to each part before rendering. Rendering is intensive computationally; it requires fast processors, large RAM, and a high-end graphics card (processor). SolidWorks offers **PhotoView 360** to render parts and assemblies. To access it, click **Office Products** tab > **PhotoWorks 360**

FIGURE 1.14
Assembly model of three parts

Step 1: Create assembly model and save assembly: **File > New > Assembly > OK**

Browse (button on left pane of screen and shown here to the right) > locate each part file and double click it to add it > when done, go to graphics window and click in the middle to place the *basePlate* part > click again above *basePlate* to place *pin* > click one last time somewhere between *basePlate* and *pin* to place *plate*. Place all as shown in Figure 1.14B. You can change the insertion order by selecting a part from **Browse** window.

Note: The first part you insert (*basePlate*) is fixed in space. Features tree shows an **(f)** before its name as shown in Figure 1.14A. Other parts that follow are floating (–), meaning you can move them by dragging them along their degrees of freedom. Right click a part in the tree > **Float** (or **Fix**) to convert it from fixed (or float).

Step 2: Mate the parts and save assembly: Mate the *plate* and *basePlate*: **Mate** on **Assembly** tab > **Coincident** > select two corresponding edges of the two components > click right mouse button (✔). Repeat for two other corresponding edges: **Coincident** > select two corresponding edges of the two components > ✔.

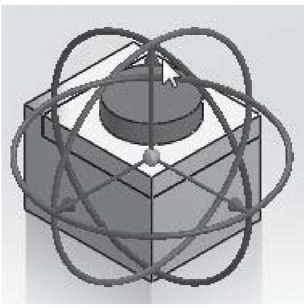
Mate *pin* with *basePlate*: **Concentric** > select cylindrical faces of *pin* and *basePlate* hole > ✔. Rotate parts with mouse to have clear view of faces to select and mate.

Mate *pin* and *plate*: **Coincident** > select bottom face of *pin* head and top face of *plate* > click right mouse button > ✔ on left pane to finish.

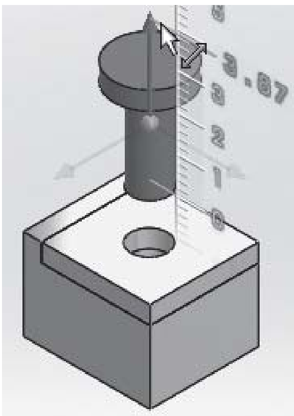
Save assembly as *assemblyModel*: **File > Save As >** *assemblyModel* > **Save.**

Note: Expand the **Mates** node in features tree on left pane to see mates you created (see Figure 1.14A). A total of four mates (two for each pair of components) are sufficient to create the assembly. The pin still has the rotational degree of freedom free. Drag the pin with the mouse left button to spin it (although you cannot visually see the pin rotates). Try to drag any of the other two parts to move or spin them; you cannot. They are fully constrained in space.

Step 3: Create exploded view: We start with the assembly collapsed state (Figure 1.14A) and create the exploded view state (Figure 1.14B). Click **Exploded**

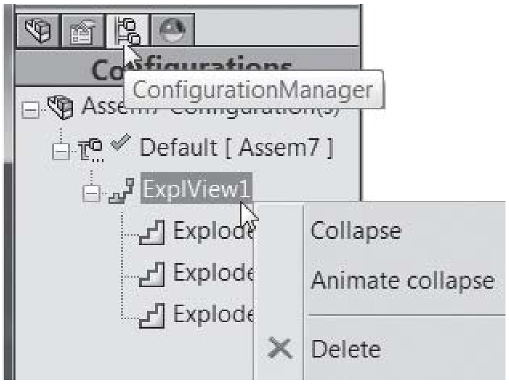


These axes show when you select *pin*. As shown to the right, drag *pin* up and away.



View (**Assembly** tab) > select *pin* > grab the tip (arrow) of the Y-axis and move *pin* up and away from *plate* > select *plate* > grab the tip of the Y-axis and move *plate* up and away from *basePlate* > select *basePlate* > grab the tip of the Y-axis and move *basePlate* down and away from *plate* > ✔

Note: Click the **ConfigurationManager** tab (hover to read it) on left pane to manage the exploded view. Expand the **ExpView1** node. Right click it and select an item from the menu below to collapse the view or animate it. The animation opens **Animation controller** window. Close it to stop animation.



Step 4: Create assembly drawing: Follow steps of Example 1.4. Show only the overall dimensions in assembly drawings as shown in Figure 1.14C.

Step 5: Render the assembly model: We assign material to each part, then render: Right click *pin* > **Material** > **Edit Material** > **Aluminium Alloys** > **6061 Alloy** > **Apply** > **Close.** Repeat and assign *plate* and *basePlate* same material.

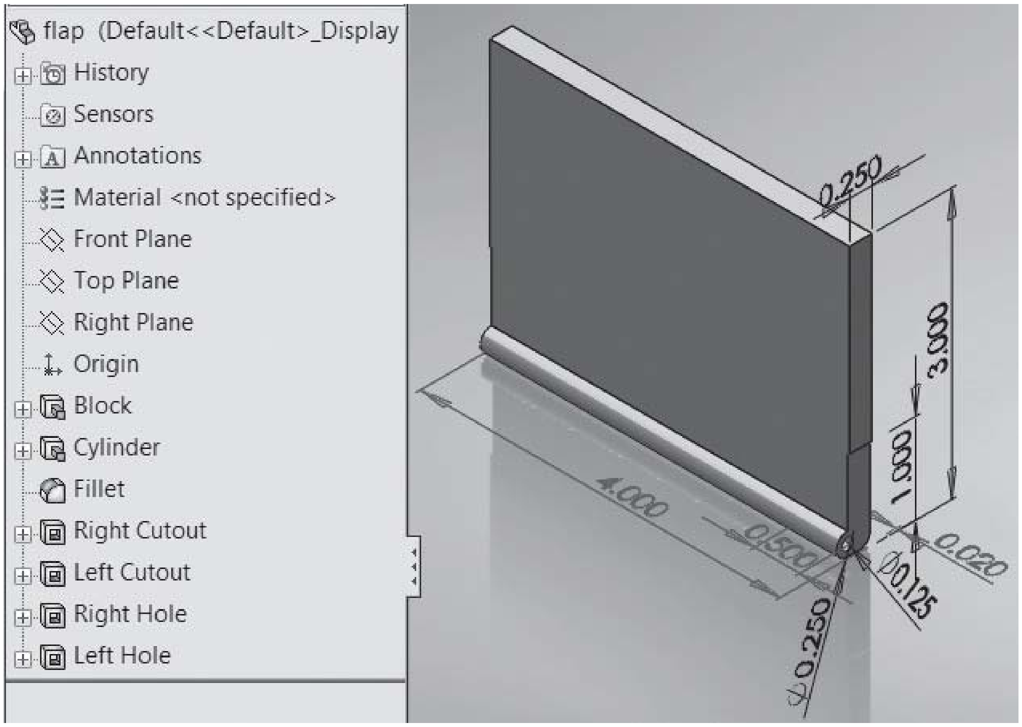
Render the assembly: **Office Products** tab > **PhotoView 360** (to open **Render Tools** tab) > **Render Tools** tab > **Integrated Render**

Note: Unlike **Integrated Render**, **Final Render** icon on **Render Tools** tab opens its own window.

1.12 Tutorials

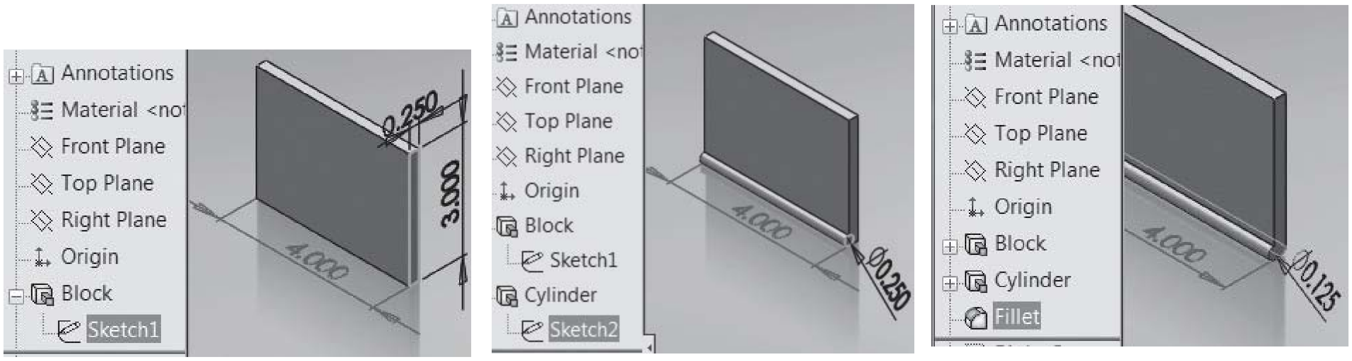
Tutorial 1–1: Create the Flap

This tutorial and the next two create four parts for a door valve assembly of a mass spectrometer machine (not shown here). Figure 1.15 shows the flap modeling plan and Figure 1.16 shows the execution of the plan. All dimensions are in inches.



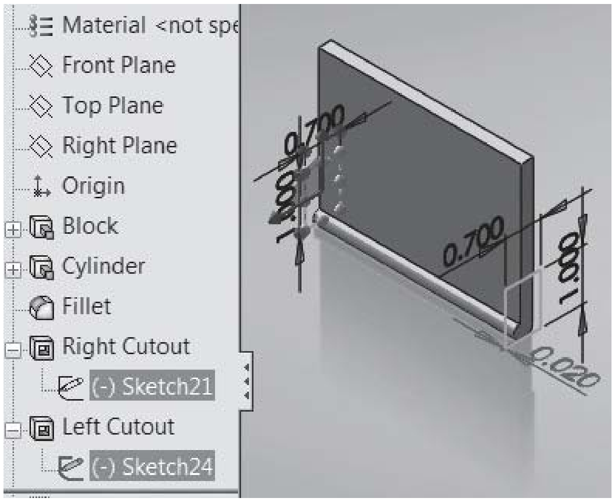
The flap is symmetric with respect to a right plane passing through its center. Thus we show the dimensions for half of it only. The features tree to the left shows the modeling plan and its steps to create the flap. Be sure to remember the symbols for Extruded Boss/Base and Extruded Cut to follow the tree.

FIGURE 1.15
Flap and its modeling plan

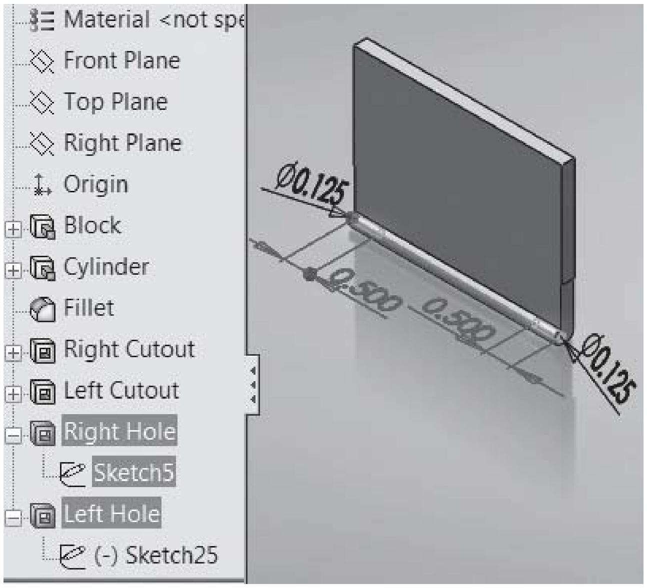


Step 1: Create *Block* extrusion. **Step 2:** Create *Cylinder* extrusion. **Step 3:** Create *Fillet* Feature.

FIGURE 1.16
Execution of the flap modeling plan



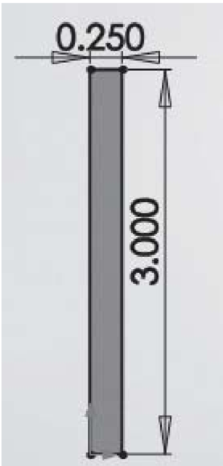
Step 4: Create *Right Cutout* and *Left Cutout* extruded cuts.



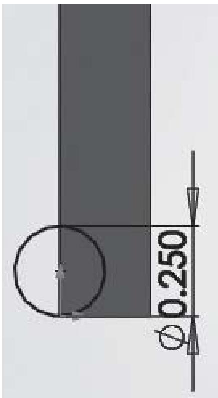
Step 5: Create *Right Hole* and *Left Hole* extruded cuts.

FIGURE 1.16
(continued)

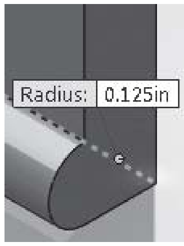
Step 1: Create *Sketch1* and *Block* feature: **File** > **New** > **Part** > **OK** > **Right Plane** > **Extruded Boss/ Base** on **Features** tab > **Rectangle** on **Sketch** tab > click origin and drag to sketch > dimension the cross section as shown to the right > exit sketch > enter 4.0 for thickness (**D1**) > reverse extrusion direction (drag extrusion arrow) > click ✔ to finish > click tree node and rename it *Block*.



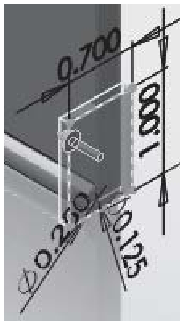
Step 2: Create *Sketch2* and *Cylinder* feature: Click the right face of *Block* feature to select it as a sketch plane > **Extruded Boss/ Base** on **Features** tab > **Circle** on **Sketch** tab > sketch a circle (center on vertical edge and tangent to bottom edge) and dimension it as shown > **Exit Sketch** > select **Through All** from dropdown of **Direction 1** menu for extrusion > ✔ > click tree node and rename it *Cylinder*.



Step 3: Create fillet feature at *Block* corner: **Features** tab > **Fillet** > select bottom back edge of *Block* as shown to the right > click **Radius** value and change to 0.125 as shown > ✔ > click tree node and rename it *Flap Fillet*.



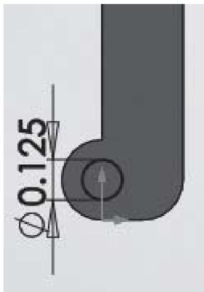
Step 4: Create *flap* right and left cutouts: Click right face of *flap* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Rectangle** on **Sketch** tab > sketch a rectangle and dimension as shown > exit sketch > enter 0.02 for **D1** > ✔ > click tree node and rename it *Right Cutout*.



Note: The size of rectangle should be big enough to cut *flap* right face.

Repeat to create *Left Cutout*. Rotate model with mouse to access left face.

Step 5: Create *flap* right and left holes: Click right face of *flap* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > sketch a concentric circle to *Cylinder* circle (click at center point) and dimension as shown > exit sketch > enter 0.5 for **D1** > ✔ > click tree node and rename it *Right Hole*.

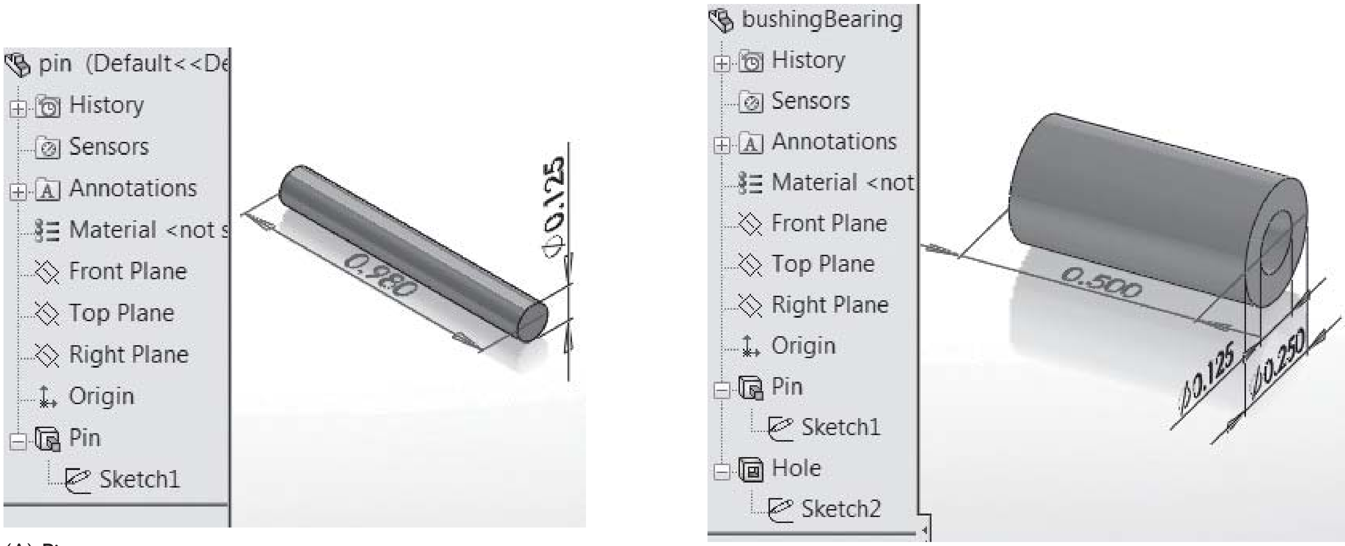


Repeat to create *Left Hole*. Rotate model with mouse to access left face.

HANDS-ON FOR TUTORIAL 1–1. Create two additional side cutouts at the right and left faces of the flap. Each cutout starts from the top edge of the face with a size of 0.02 × 1.0 in., similar to the bottom cutouts shown in Figure 1.15.

Tutorial 1–2: Create the Pin and Bushing Bearing

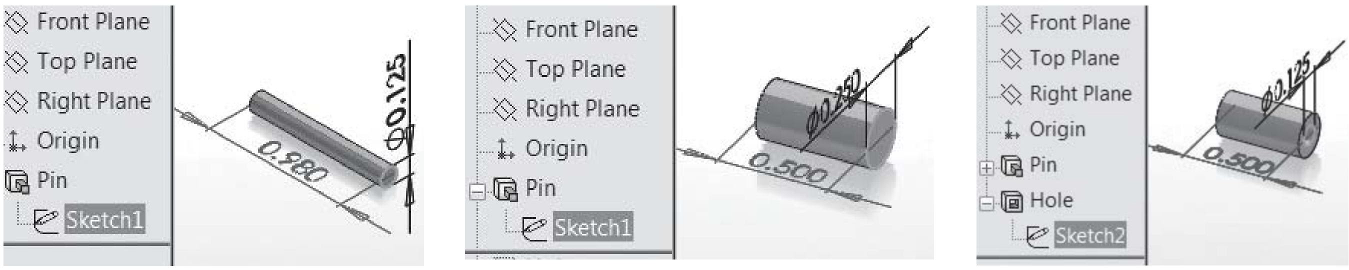
Figure 1.17 shows the modeling plan for both parts, and Figure 1.18 shows the execution of the plan. All dimensions are in inches.



(A) Pin

(B) Bushing bearing

FIGURE 1.17
Pin and bushing bearing and their modeling plans



Step 1: Create pin extrusion.

Step 2: Create bushing (pin) extrusion.

Step 2: Continued: Create hole extruded cut.

FIGURE 1.18
Execution of the pin and bushing bearing modeling plans

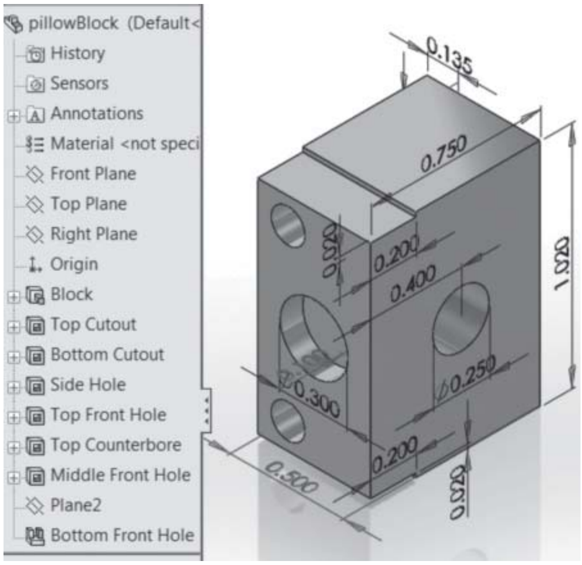
Step 1: Create *Sketch1* and *Pin* feature: **File** > **New** > **Part** > **OK** > **Right Plane** > **Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and drag to sketch > dimension the circle diameter as shown in Figure 1.17A > exit sketch > enter 0.98 for thickness (**D1**) > reverse extrusion direction (drag extrusion arrow) > ✔ to finish > click tree node and rename it *Pin*.

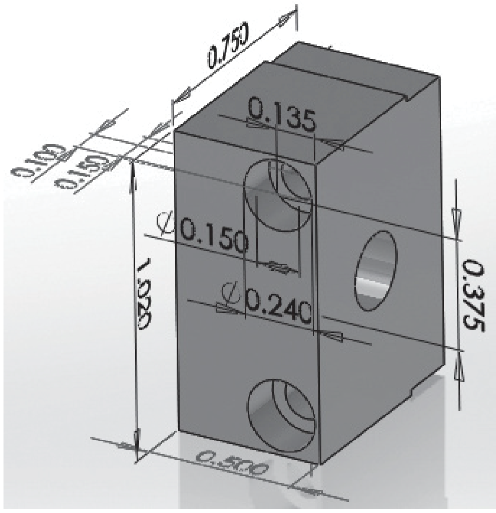
Step 2: Create *bushingBearing* feature: **File** > **New** > **Part** > **OK** > **Right Plane** > **Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and drag to sketch > dimension the circle (Ø0.25) as shown in Figure 1.17B > exit sketch > enter 0.5 for thickness (**D1**) > reverse extrusion direction (drag extrusion arrow) > ✔ to finish > select right face of feature just created to sketch on it > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > sketch a concentric circle and dimension as shown (Ø0.125) > exit sketch > enter 0.5 for D**1** > ✔

HANDS-ON FOR TUTORIAL 1–2. Chamfer both ends of the pin. Use chamfer distances **D** of 0.01, 0.02, 0.05, and 0.1 in. What happens to the chamfer feature? Repeat, but for a fillet feature. What happens? What is your conclusion? **Help:** To create a fillet or a chamfer feature, click this sequence: **Features** tab > **Fillet** drop-down > **Fillet** or **Chamfer**.

Tutorial 1–3: Create the Pillow Block

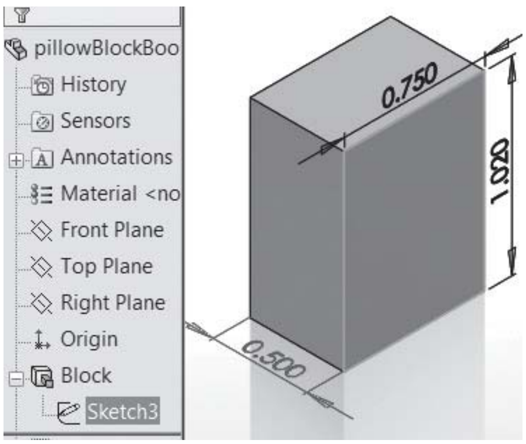
Figure 1.19 shows the pillow block modeling plan and Figure 1.20 shows the execution of the plan. All dimensions are in inches.



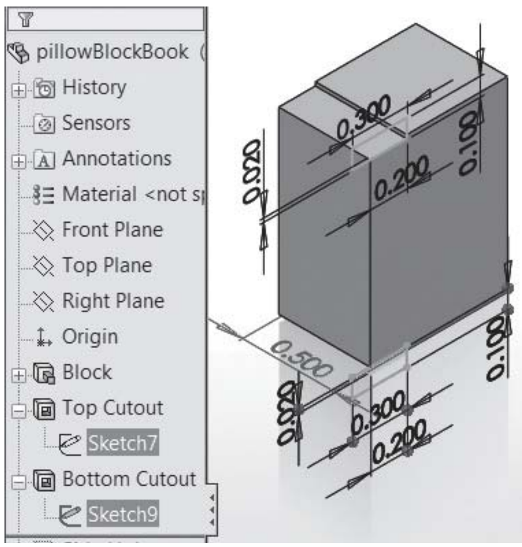


Rear view to show rear dimensions

FIGURE 1.19
Pillow block and its modeling plan

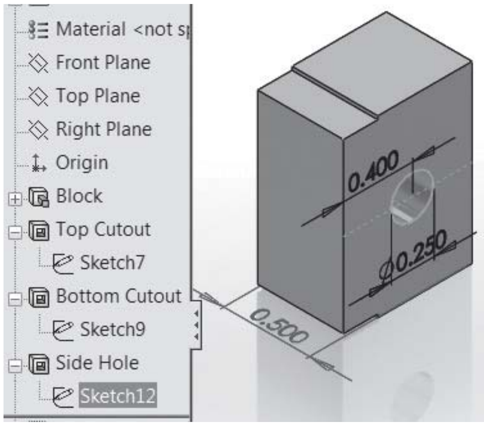


Step 1: Create *Block* extrusion.

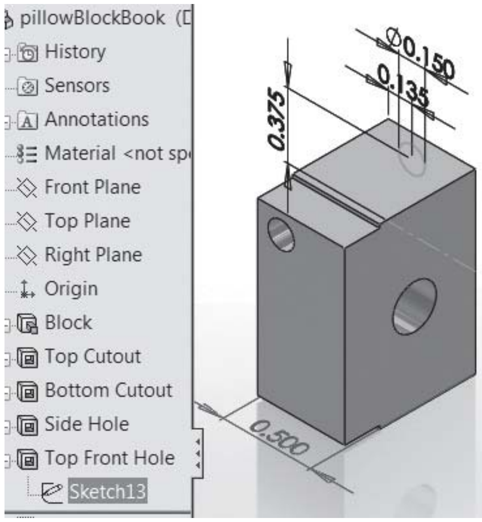


Step 2: Create *Top Cutout* and *Bottom Cutout* extruded cuts.

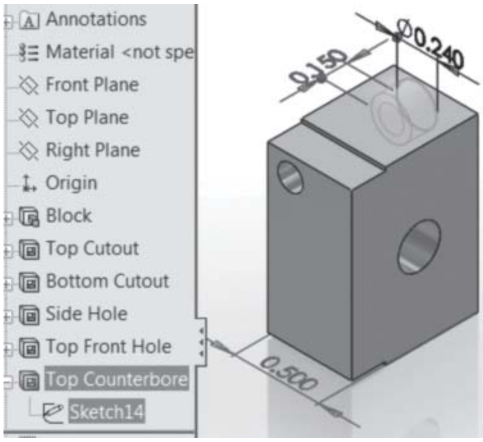
FIGURE 1.20
Execution of the pillow block modeling plan



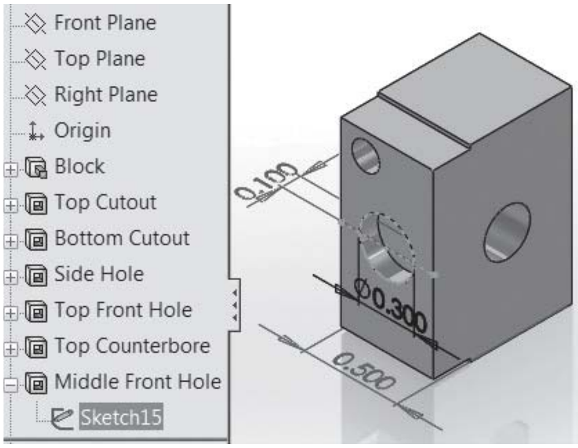
Step 3: Create *Side Hole* extruded cut.



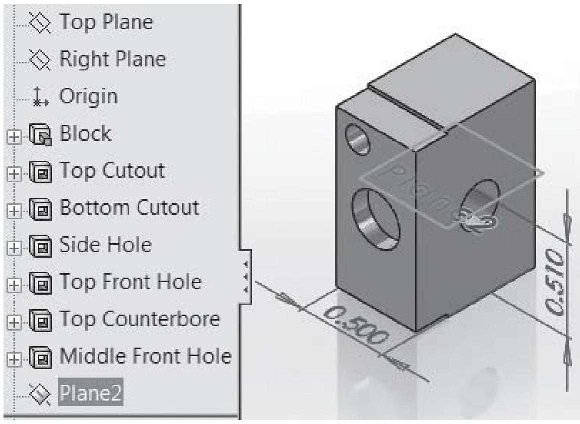
Step 4: Create *Top Front Hole* extruded cut.



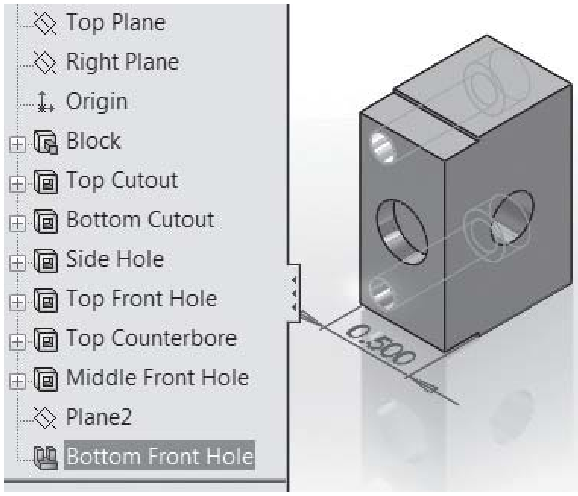
Step 5: Create *Top Counterbore* extruded cut.



Step 6: Create *Middle Front Hole* extruded cut.



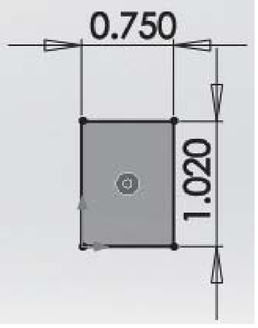
Step 7: Create *Plane2* Reference Geometry feature.



Step 8: Create *Bottom Front Hole* feature.

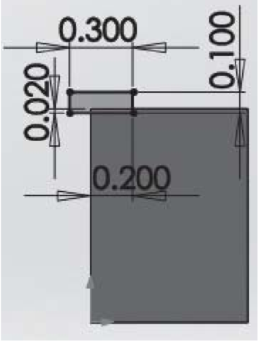
FIGURE 1.20
(continued)

Step 1: Create *Block* extrusion: **File > New > Part > OK > Right Plane > Extruded Boss/Base** on **Features** tab **> Rectangle** on **Sketch** tab **>** click origin and drag to sketch **>** dimension the cross section as shown below > exit sketch > enter 0.5 for thickness (**D1**) > reverse extrusion direction (drag extrusion arrow) > click ✔ to finish > click tree node and rename it *Block*.

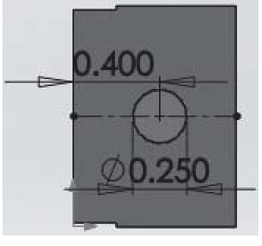


Step 2: Create *Top Cutout* and *Bottom Cutout* Extruded cuts: Click right face of *Block* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Rectangle** on **Sketch** tab > sketch a rectangle and dimension as shown > exit sketch > select **Through All** from dropdown of **Direction 1** menu for extrusion > ✔ > click tree node and rename it *Top Cutout*.

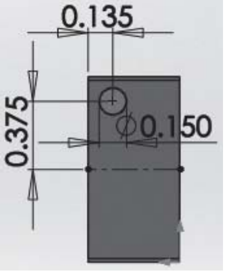
Repeat to create the *Bottom Cutout* feature.



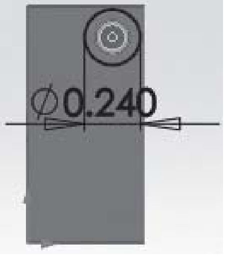
Step 3: Create *Side Hole* extruded cut: Click right face of *Block* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Centerline** from the **Line** menu on **Sketch** tab > select the midpoints of the two vertical lines of the right face as shown > **Circle** on **Sketch** tab > sketch a circle with center on the centerline and dimension as shown > exit sketch > select **Through All** from dropdown of **Direction 1** menu for extrusion > ✔ > click tree node and rename it *Side Hole*.



Step 4: Create *Top Front Hole* extruded cut: Click front face of *Block* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > select the midpoints of the two vertical lines of the right face as shown > **Circle** on **Sketch** tab > click on front face and drag to sketch **>** dimension the circle center and diameter as shown > exit sketch > select **Through All** from dropdown of **Direction 1** menu for extrusion > ✔ > click tree node and rename it *Top Front Hole*.

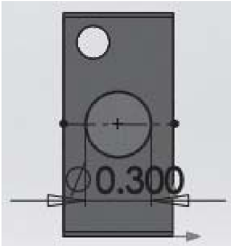


Step 5: Create *Top Counterbore* extruded cut: Rotate the model and click the back face of *Block* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > click the center of existing hole and drag to sketch **>** dimension the circle as shown > exit sketch > select **Blind** from dropdown of **Direction 1** menu for extrusion > enter 0.15 for **D1** > ✔ > click tree node and rename it *Top Counterbore*.



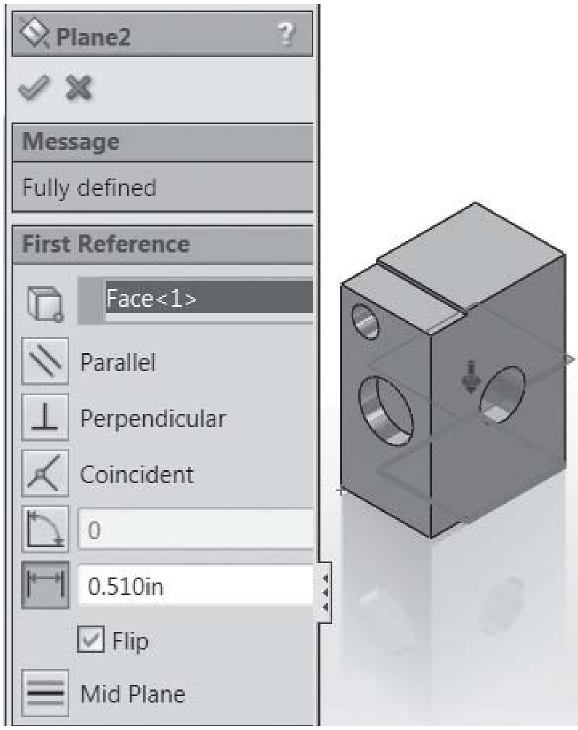
Step 6: Create *Middle Front Hole* extruded cut: Click the front face of *Block* feature to select it as a sketch plane > **Extruded Cut** on **Features** tab > **Centerline** from the **Line** menu on **Sketch** tab > select the midpoints of the two vertical lines of the front face as

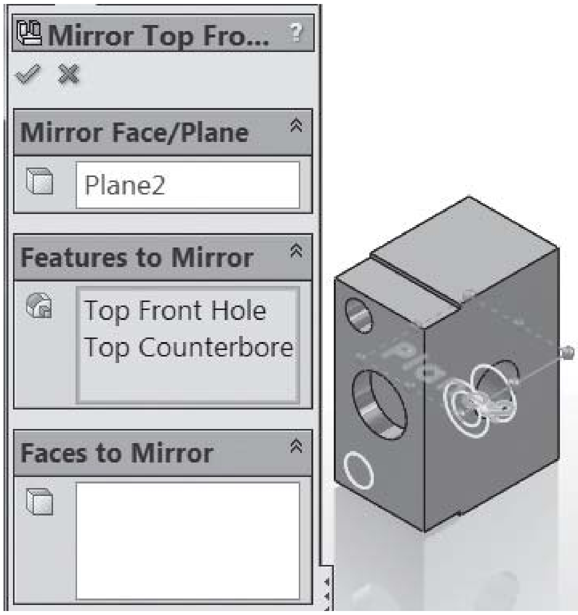
shown > **Circle** on **Sketch** tab > sketch a circle with center on the midpoint of centerline and dimension as shown > exit sketch > select **Blind** from dropdown of **Direction 1** menu for extrusion > enter 0.1 for **D1** > ✔ > click tree node and rename it *Middle Front Hole*.



Step 7: Create *Plane2* Reference Geometry feature: **Reference Geometry** on **Features** tab > **Plane** > rotate model and click bottom face as **First Reference** as shown > enter 0.51 in the **Distance** field as shown > click **Flip** checkbox as shown > ✔ > click tree node and rename it *Plane2*.

Step 8: Create *Bottom Front Hole* feature: **Mirror** on **Features** tab > click *Plane2* as **Mirror Face/Plane** as shown > select top hole and counterbore features from the features tree (not shown here) as the **Features to Mirror** > ✔ > click tree node and rename it *Bottom Front Hole*.





HANDS-ON FOR TUTORIAL 1–3. Redo this tutorial using the cross section modeling approach. How many steps did it take you to create the pillow block? Compare the pros and cons of both the features modeling approach used here and the cross section approach you used.

Tutorial 1–4: Create Drawings

We follow Example 1.4 to create the engineering drawings for the flap and the pillow block. Figures 1.21 and 1.22 show the two drawings. We repeat the steps covered in Example 1.4 here for convenience.

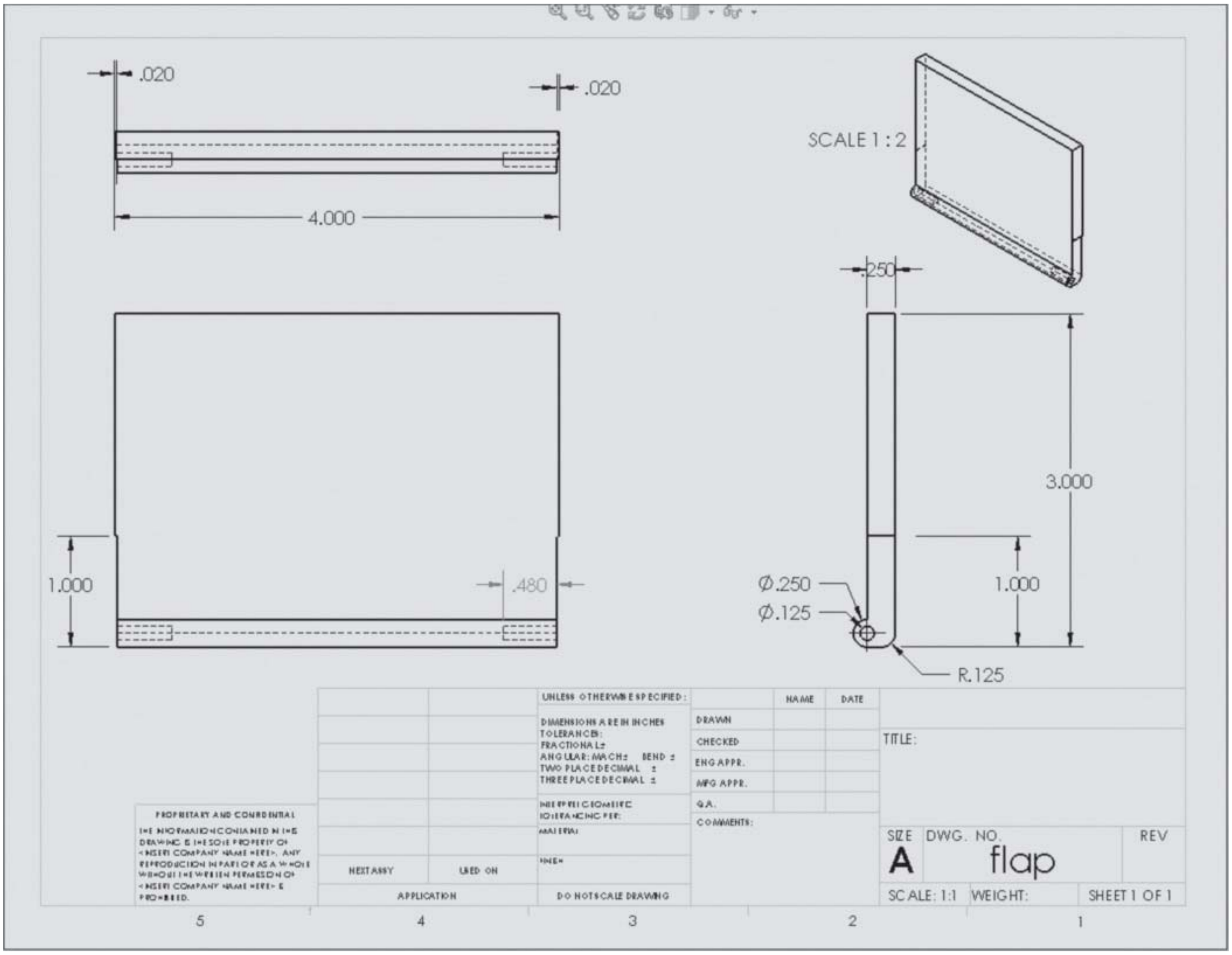


FIGURE 1.21
Flap drawing

Step 1: Create the drawing and views: **Open** > **New** > **Drawing** > **OK** > **OK** (accept the default **Standard sheet size**) **> Browse** (locate and select the *flap* part) > click the mouse near the bottom left corner of the drawing sheet to insert the front view > move the mouse up and click to place the top view > move the mouse near the bottom right corner and click to place the right view > move the mouse near the top right corner and click to place the ISO view > click the mouse right button to finish.

Note: Click this sequence to change angle of projection if you get the wrong views: Right click anywhere on the drawing sheet > **Properties** > **Third angle** from window that pops up > **OK.**

Step 2: Create dimensions, notes, and save drawing: To insert dimensions click **Annotation** (tab) **> Model Items >** ✔. Save the drawing: **File > Save As** > *flap* > **Save**

Step 3: Create the drawing and views of pillow block: Repeat Steps 1 and 2 to generate the drawing shown in Figure 1.22.

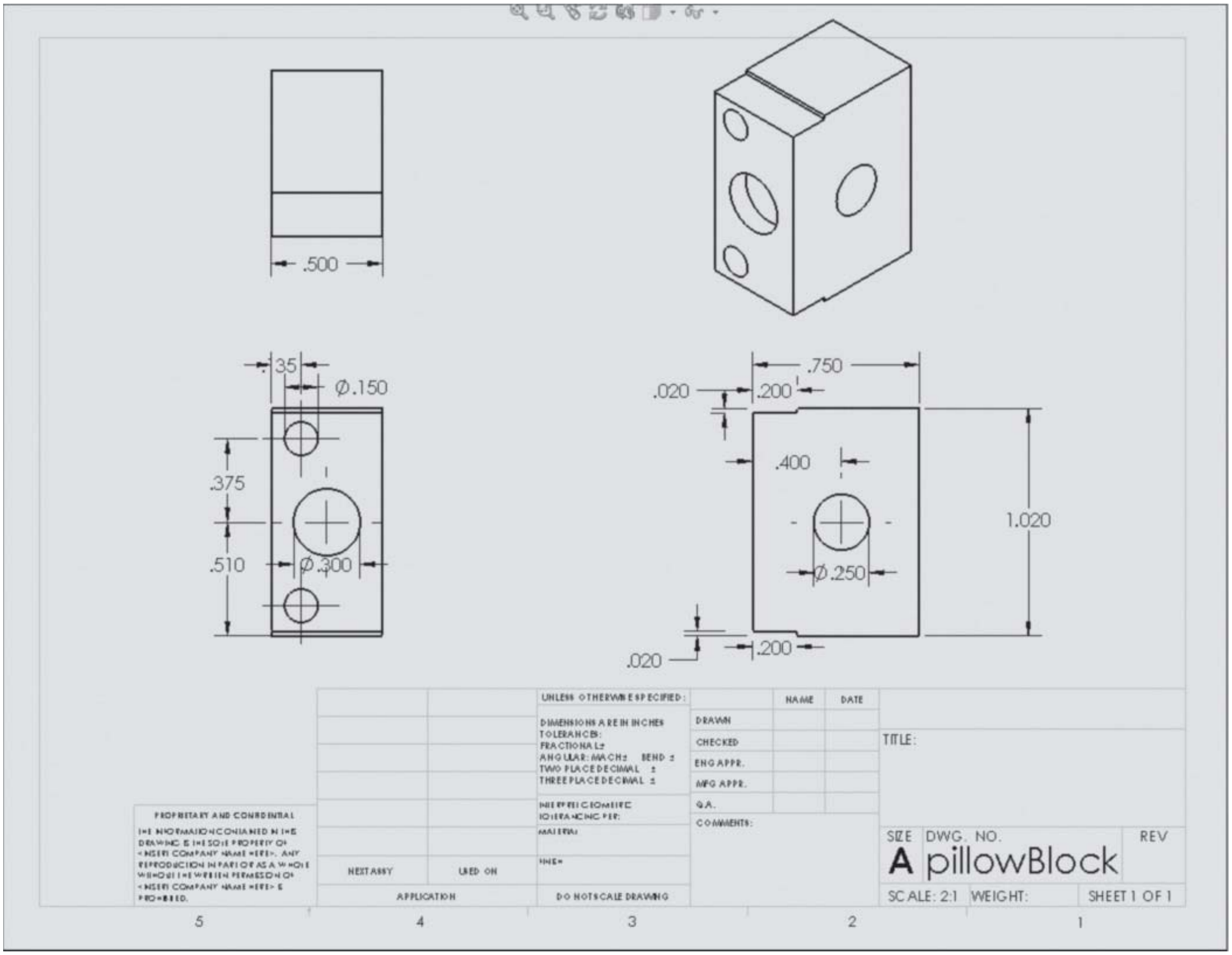


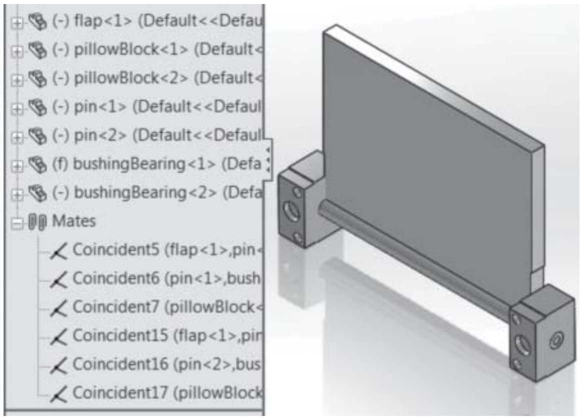
FIGURE 1.22
Pillow block drawing

> ***HANDS-ON FOR TUTORIAL 1–4.*** Create the drawings for the pin and the bushing bearing.

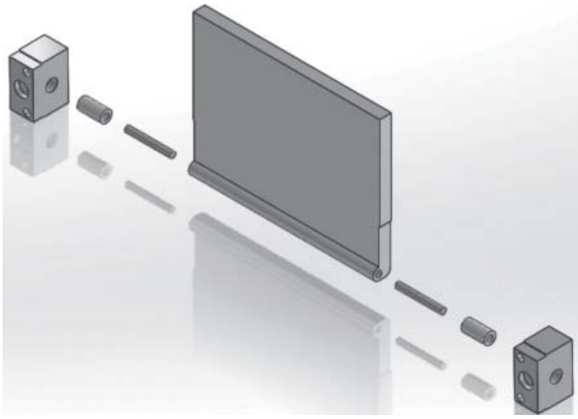
Tutorial 1–5: Create Assembly

Create the assembly model of the mass spectrometer parts. Also, create an assembly drawing, an animated exploded view, and a rendered view of the assembly.

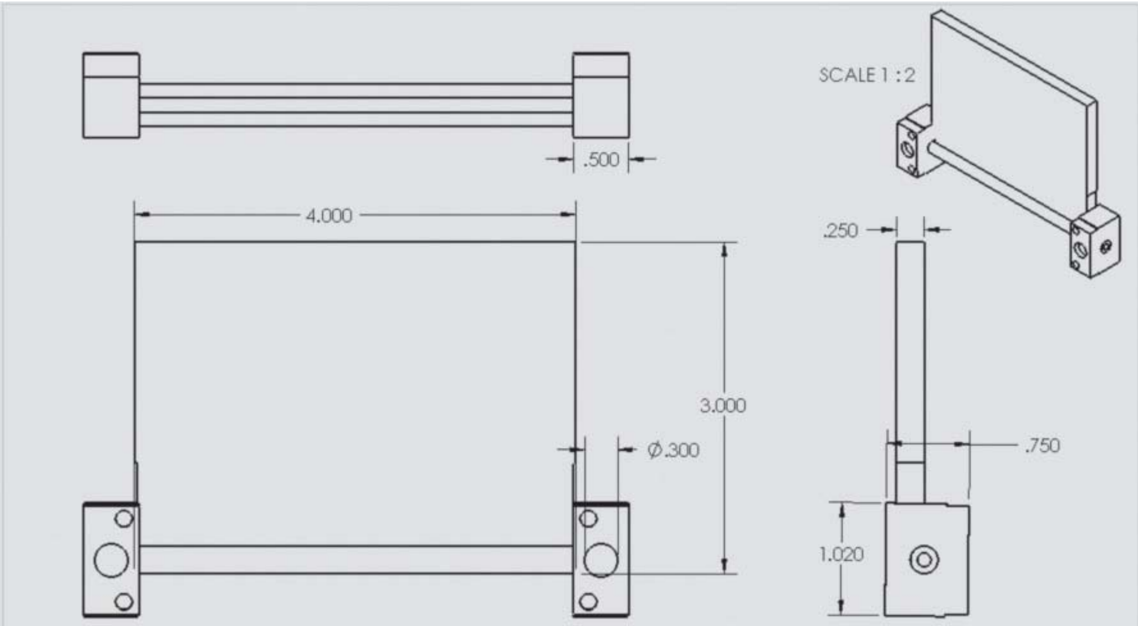
We follow Example 1.5 to create the assembly. Figure 1.23 shows the assembly. We repeat the steps covered in Example 1.5 here for convenience.



(A) Collapsed assembly view



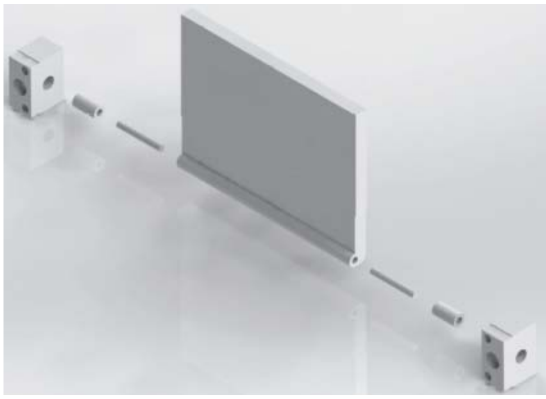
(B) Exploded assembly view



(C) Assembly drawing

Note: To insert dimensions click **Annotation** (tab) **> Model Items > Selected feature** for **Source** on left pane on screen to avoid cluttering the drawing if you use **Entire model**. We selected *flap* and *pillowBlock* features to show dimensions above.

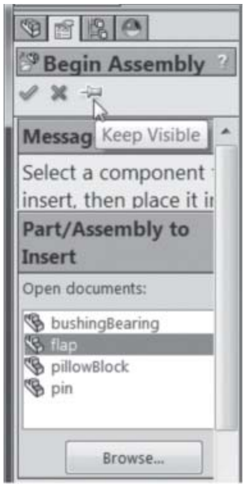
Note: Click **View** (menu) > **Display** > **Shaded with Edges** to remove rendering.



(D) Rendered assembly model

FIGURE 1.23
Assembly model of the mass spectrometer parts

Step 1: Create assembly model and save assembly: **File > New > Assembly > OK**



Browse (button on left pane of screen and shown here to the right) > locate each part file and double click it to add it > when done, click the push pin shown (to insert multiple instances of one part) > select *flap* as shown > go to graphics window and click in the middle to place it > select *pillowBlock* > click again to the right of *flap* > click again to the left of *flap* to place a second instance of *pillowBlock* > repeat to place two instances of *pin* in the graphics window > repeat to place two instances of *bushingBearing* in graphics window. (Place all as shown in Figure 1.23B.)

Step 2: Mate the parts and save assembly: Mate *pin* and *flap*: **Mate** on **Assembly** tab > **Coincident** > select two corresponding edges of the two components (edge circle of *pin* and inside edge circle of blind hole in *flap*) > click mouse right button (✔).

Mate *pin* with *bushingBearing*: **Coincident** > select two corresponding edges of the two components (edge circle of *pin* and edge of *bushingBearing* hole circle > click mouse right button (✔).

Mate *bushingBearing* and *pillowBlock*: **Coincident** > select two corresponding edges of the two components (edge circle of *pin* and edge of *bushingBearing* hole circle) > click mouse right button (✔) > ✔ on left pane to finish.

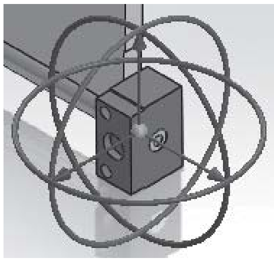
Save assembly: **File > Save As** > *doorValve* > **Save.**

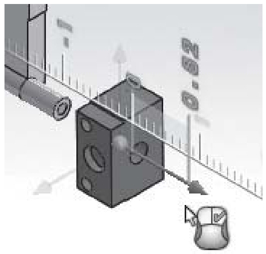
Note: Repeat to assemble the other instances on the other side. We mate edges instead of faces because it enables us to use the minimum number of mates.

Note: Right click *flap* > **Float**. Then right click each *bushingBearing* instance > **Fix**. Now, rotate *flap* with the mouse and observe the door moves.

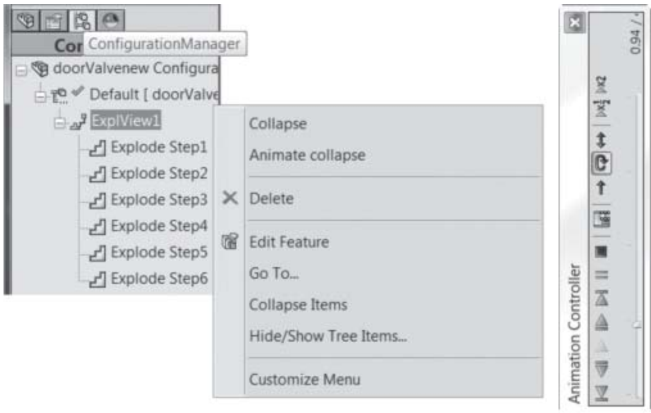
Step 3: Create exploded view: We start with the assembly collapsed state (Figure 1.23A) and create the exploded view state (Figure 1.23B). Click **Exploded View** (**Assembly** tab) > select *pillowBlock* > grab the tip of the X-axis and move *pillowBlock* to the right > select *bushingBearing* > grab the tip of the X-axis and move *bushingBearing* to the right > select *pin* > grab the tip of the X-axis and move *pin* to the right > ✔.

Repeat to explode the left side of *doorValve*.





Note: Click the **ConfigurationManager** tab (hover to read it) on left pane to manage the exploded view. Expand the **ExpView1** node. Right click it and select an item from the menu below to collapse the view or animate it. The animation opens **Animation controller** window. Close it to stop animation.



Step 4: Create assembly drawing: Follow Step 1 of Tutorial 1.4 **>** to insert dimensions click **Annotation** (tab) **> Model Items > Selected feature** for **Source >** select *flap* and *pillowBlock* features in drawing sheet **>** ✔

Step 5: Render the assembly model: We assign material to each part then render: Right click *pin* > **Material** > **Edit Material** > **Aluminum Alloys** > **6061 Alloy** > **Apply** > **Close.** Repeat and assign *flap*, *bushingBearing*, and *pillowBlock* same material.

Render the assembly: **Office Products** tab > **PhotoView 360** (to open the **Render Tools** tab) > **Render Tools** tab > **Integrated Render**

Note: Unlike **Integrated Render**, **Final Render** icon on **Render Tools** tab opens its own window.

HANDS-ON FOR TUTORIAL 1–5. Figure 1.23 A shows that we needed only six mates to assemble the *doorValve* parts properly. Re-create the assembly using faces of parts instead of edges. How many mates did you need to constrain the assembly properly? Explain your answer.

problems

1. Reverse engineering is a good first step to learn good design methodologies. Similarly, reverse CAD modeling is a good first step to learn good CAD modeling techniques. Search 3DcontentCentral (www.3dcontentcentral.com) or grabcad.com, find a CAD model of one simple part of interest to you, download its SolidWorks files, open them in SolidWorks, and study its features tree (roll back the tree to follow model creation). After understanding how the model is created, analyze the modeling steps and evaluate how effective is the modeling plan. Can you find other modeling plans that are quite different? If yes, do you think they are more or less efficient?
2. Same as Problem 1, but for a more complex part.
3. Same as Problem 1, but for a simple assembly model. For this problem, reverse engineer the mates of the assembly and analyze them.
4. Same as Problem 3, but for a complex assembly.
5. SolidWorks allows you to reorder the nodes of a features tree by dragging a node up or down the tree and dropping it before or after another node. SolidWorks allows you to shuffle the tree nodes as long as you do not violate the model topology (i.e., you cannot create an extruded cut if you do not have an extruded boss/base, or material, to cut from). Follow Tutorial 1.1 and create the flap part. Try to reorder the features tree nodes by moving them around. Which nodes can you move around and which can you not? Explain why or why not.
6. Same as Problem 5 above, but for Tutorial 1.3, pillow block part.
7. The flap part of Tutorial 1.1 is symmetric with respect to the right sketch plane, passing through its midsection. Re-create the CAD model by constructing half of the flap, then mirror it with respect to the right plane. Hint: Study how we used the mirror feature in Tutorial 1.3, pillow block part.
8. Editing existing CAD models offers a quick, powerful way of changing a design during the conceptual design phase (Step 3 of EDP shown in Figure 1.1) to explore different design ideas. Create the CAD model of the plate part of Example 1.1. Then edit it and make it round (i.e., convert the square plate into a cylindrical part of diameter 3.0 inches). You must edit the tree node of the part to convert the part by editing its sketch. Figure 1.24 shows the modified part.

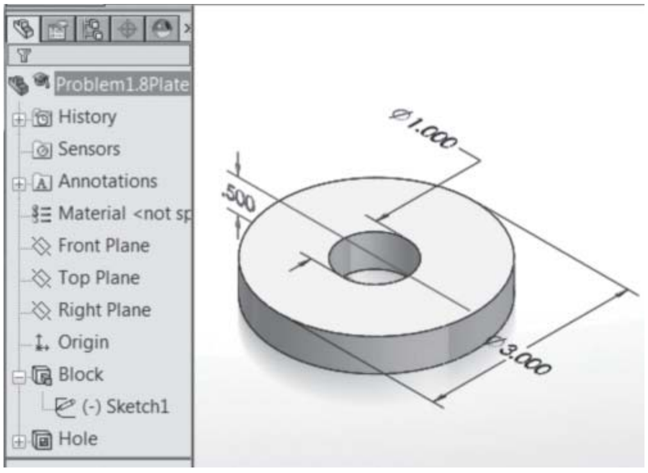
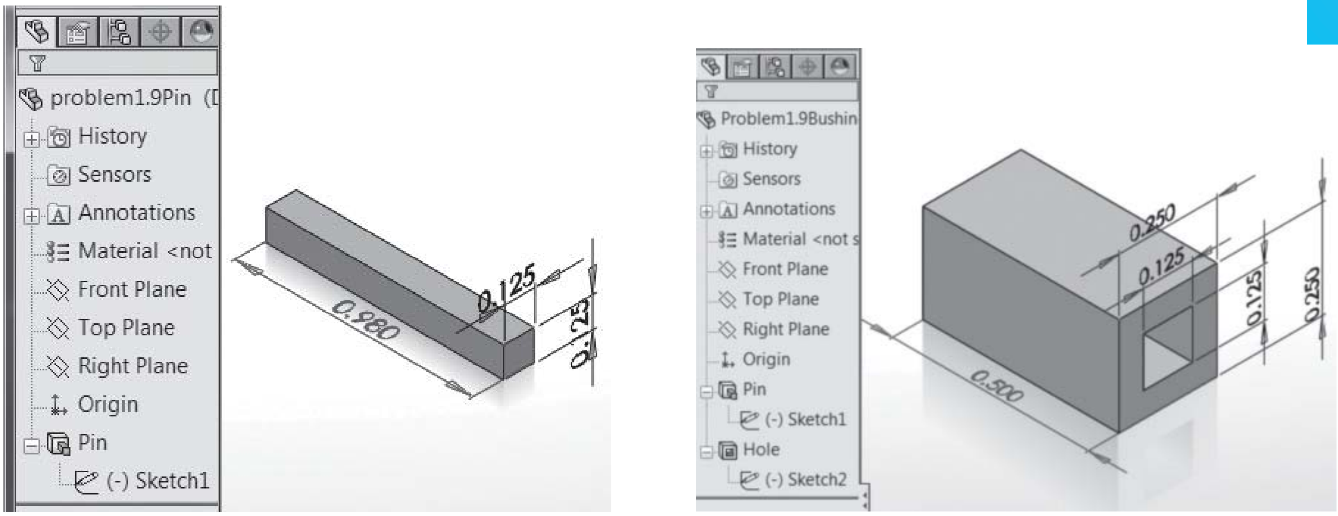


FIGURE 1.24
Round plate

9. Same as Problem 8, but for the pin and bushing bearing of Tutorial 1.2. Edit the tree nodes of each tree to create a square pin and a square bushing bearing as shown in Figure 1.25. Create the assembly model of the two parts.



(A) Square pin

(B) Bushing

FIGURE 1.25
Square pin and bushing

10. Create the CAD part of the chamfered block model shown in Figure 1.26. Also, create the drawing of the part.

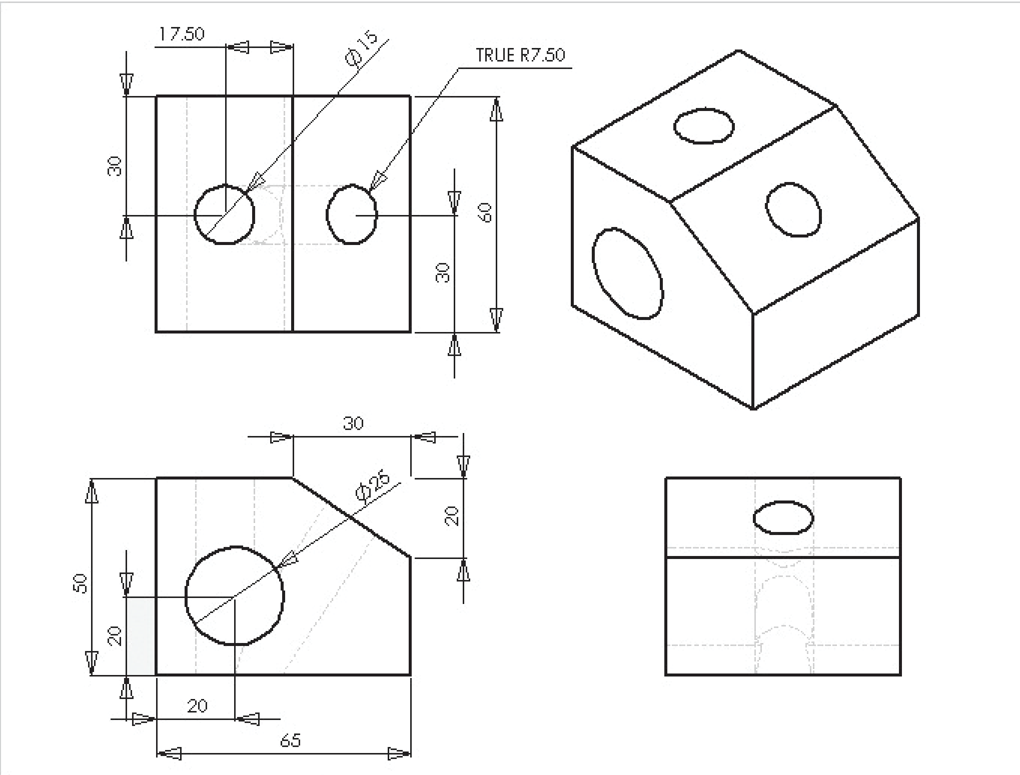


FIGURE 1.26
A chamfered block CAD model. All dimensions are in mm.

11. Create the CAD part of the slotted block model shown in Figure 1.27. Also, create the drawing of the part.

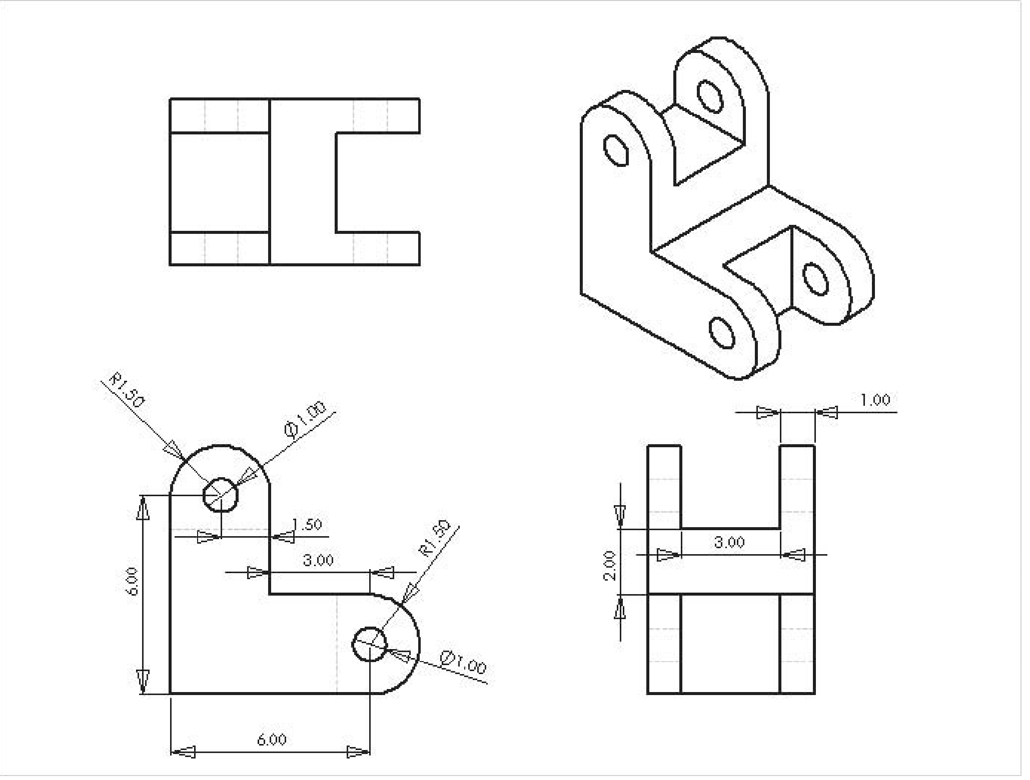


FIGURE 1.27
A slotted block CAD model. All dimensions are in inches.

12. Create the SolidWorks part of the slotted block model shown in Figure 1.28. Use an extrusion depth of 100 mm. Also, create the drawing of the part.

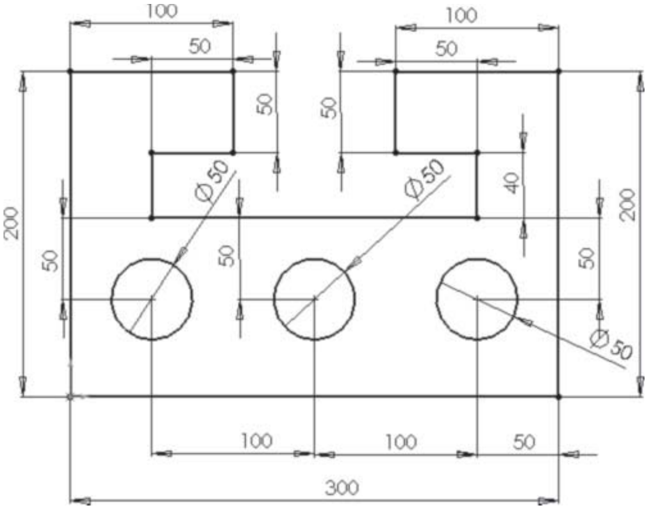


FIGURE 1.28
A slotted block CAD model. All dimensions are in mm.

13. Figure 1.29 shows a center piece of a rock climbing ascender tool. Create the CAD model of the part. Also, create the drawing of the part. Use a fillet radius of 0.2 cm for all edges of the part except the top hole as shown.

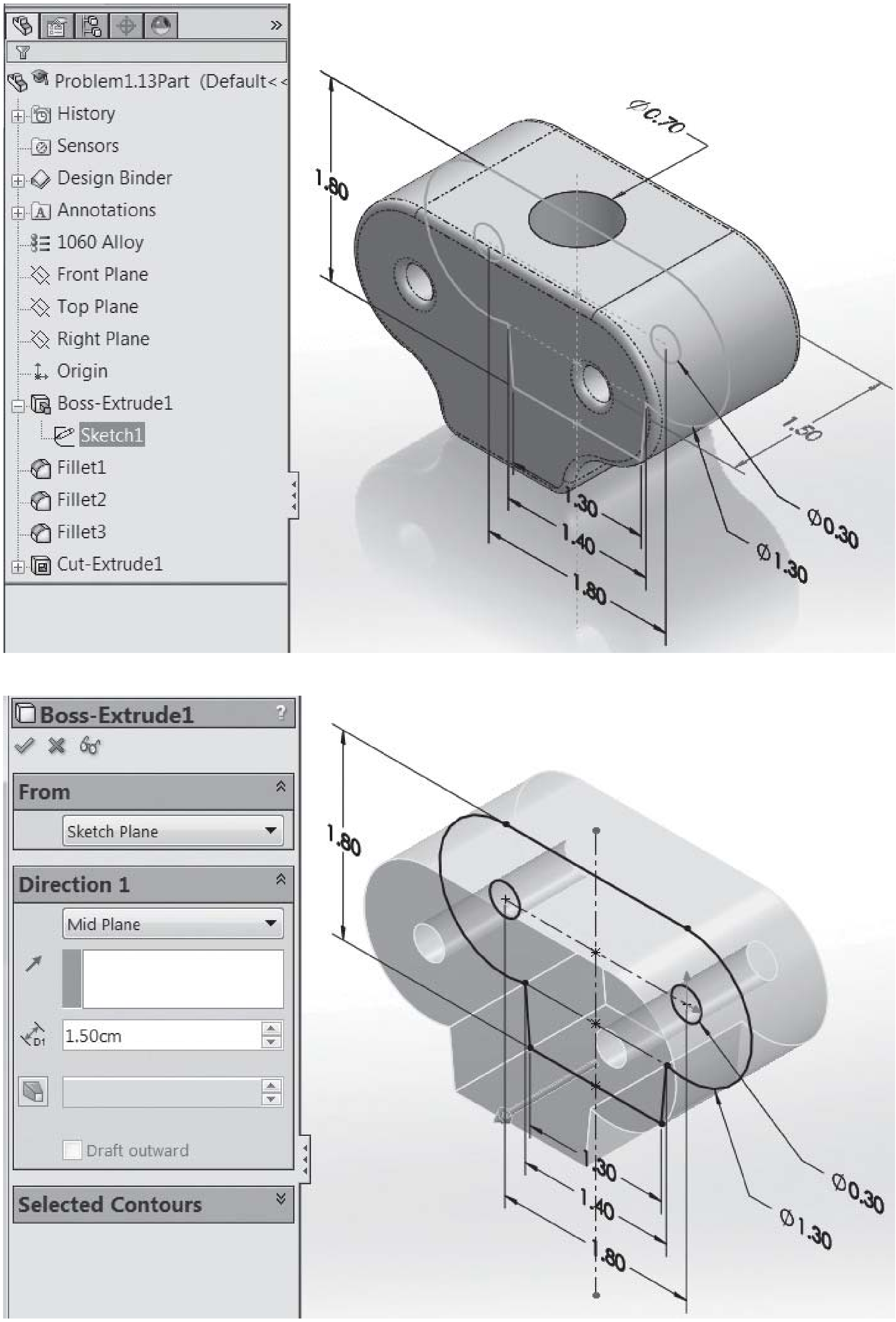


FIGURE 1.29

Center piece of a rock climbing ascender tool. All dimensions are in cm.

14. Figure 1.30 shows the pusher (pushes staples forward with the aid of a spring) of an office stapler. Create the CAD model of the part. Also, create the drawing of the part.

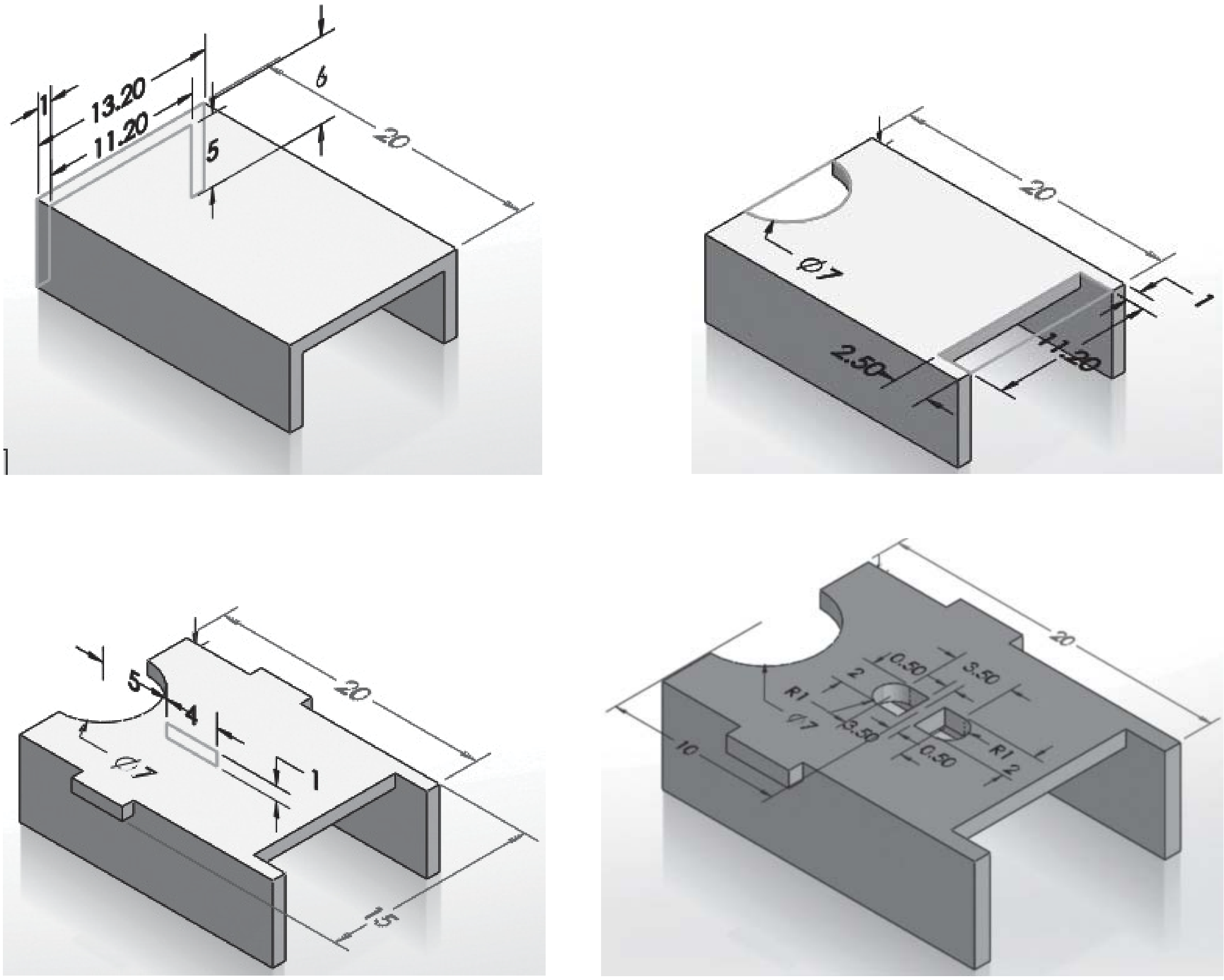


FIGURE 1.30
Pusher part of an office stapler. All dimensions are in mm.

15. Figure 1.31 shows the crankshaft of an engine piston. Create the CAD model of the part. Also, create the drawing of the part.

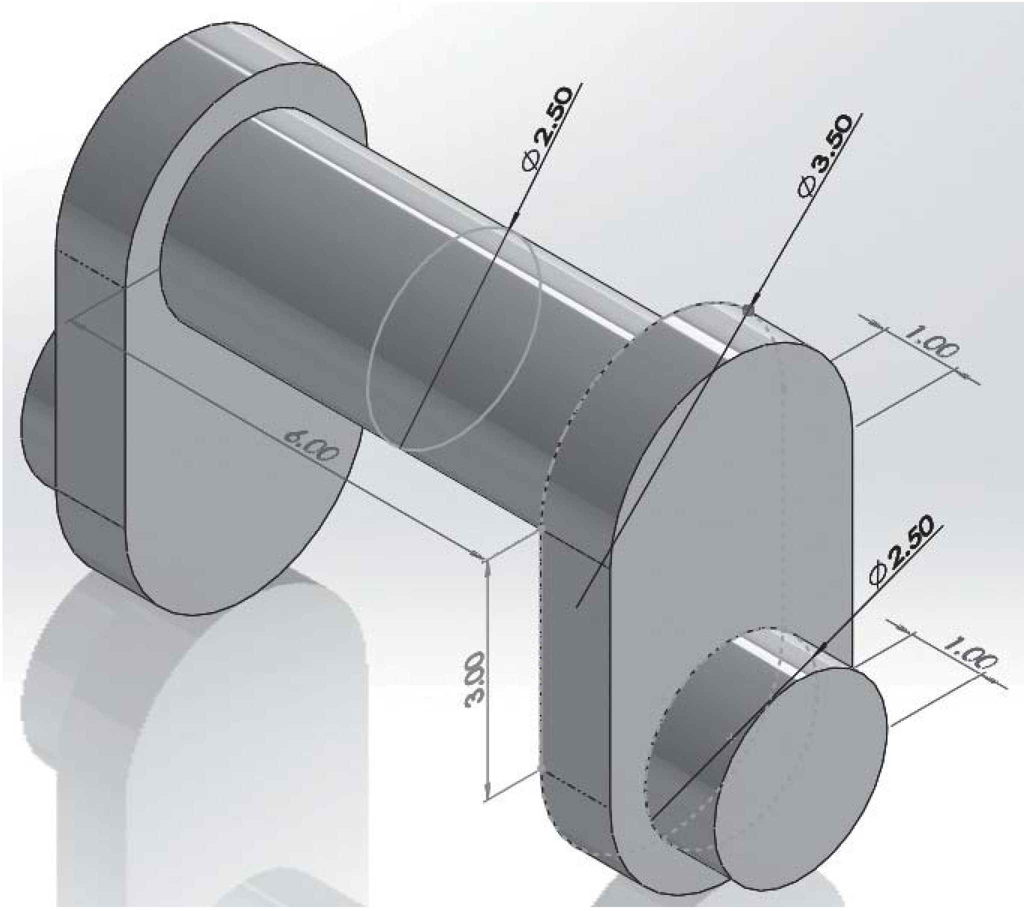


FIGURE 1.31
Crankshaft of an engine piston. All dimensions are in inches.

This page intentionally left blank

CHAPTER

2

Modeling Management

2.1 Overview

We covered part modeling planning and creation in Sections 1.9 and 1.10 respectively in Chapter 1. We extend these concepts in Chapter 2 and cover them in more depth. CAD modeling requires forward thinking and planning ahead. The way the CAD model (part) is created affects its downstream activities such as using **Model Items** in dimensioning the views of a drawing. The goal of this chapter is to cover the many concepts necessary to understand and control 3D CAD part modeling and creation.

2.2 Types of CAD Models

CAD models can be classified into four types from a creation point of view:

- Extrusion: This is a part with a constant cross section along a given axis, with a uniform thickness along this axis.
- Revolve: This is an axisymmetric part with a constant cross section through an angle of revolution about a given axis of revolution.
- Composite: This is a part that combines both extrusions and/or revolves. One or more of its subparts may be extrusions or revolves.
- Free form: This is a part that does not exhibit any uniform shape. An auto body or a computer mouse is an example. Modeling of this class of parts requires surfaces and other techniques as we discuss later in the book.

Figure 2.1 shows the different types of CAD models. The extrusions and revolves require one sketch in one sketching plane to sketch the cross section. For example, Figure 2.1A extrusion requires the top sketch plane to sketch the cross section, followed by extruding it up or down in the perpendicular direction. For the Figure 2.1B extrusion, we create the cross section in the front sketch plane. For the revolve of Figure 2.1C, we create the glass cross section and the axis of revolution in the front sketch plane. The phone model shown in Figure 2.1D is a composite part. It consists of two extrusions (the base and the receiver holder). The two cylinders can be created as additional extrusions. Finally, we can add the phone buttons as a pattern. The two models shown in Figures 2.1E and 2.1F are free-form, requiring more than one sketch in different sketch planes. We use advanced operations such as lofts to create them.

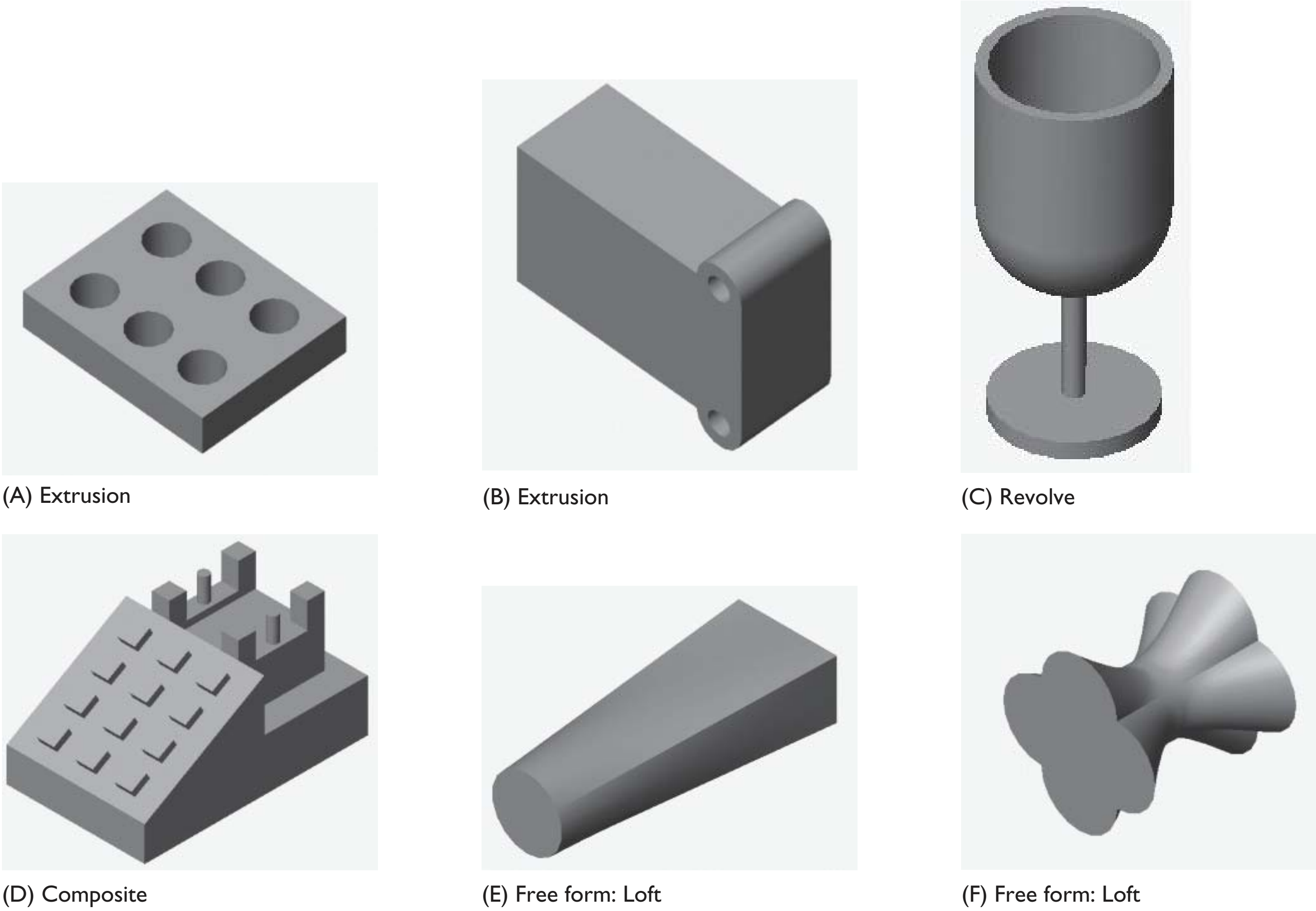


FIGURE 2.1
Types of CAD models

TABLE 2.1 Planning Part Creation on a CAD/CAM System	
Part Clue	**How to Use in Part Creation in a CAD System**
Part type	• Extrusion: create a sketch and extrude it. • Revolve: create a sketch and revolve it. • Composite: use a combination of extrusions and/or revolves. • Free form: use lofts, sweeps, surfaces, etc.
Part symmetry	• Split the part into two halves at the symmetry plane. • Construct one half and mirror it about the symmetry plane to finish the creation of the CAD model.
Part patterns	If they exist (e.g., holes in a flange or phone buttons), use CAD patterns to create them.
Designated points	Make use of end-, mid-, and/or intersection points of entities during creation to avoid unnecessary calculations. SolidWorks snaps to these points when you move the mouse close to them; or hover the mouse over them during construction.
Sketch relations	They include horizontal, vertical, and perpendicular. You must be in an active sketch to view them. You must also activate them: **View** (menu) > **Sketch Relations**. This sequence is a toggle. Do it once to show relation symbols on sketch; do it again to turn them off.
Equations	An example is $D_1 = 2D_2$.
Center planes for features	Always build features off center (i.e., use mid planes). When you extrude a sketch, extrude it on both sides of the sketch plane. This strategy is good for mirror; use the sketch plane to mirror.
Miscellaneous	• If you do not have dimensions, sketch freely. • Trim entities to clean up creation and avoid calculations. • Use transformations (move, rotate, copy, scale, offset) to speed up creation.

2.3 Planning Part Creation

Planning the creation of a part usually comes with experience; the more you know about what a CAD/CAM system can do, the faster and more efficiently you can create models. When it comes to planning part creation on a CAD system, the best approach is to examine the part for geometric clues that may lead you to the easiest and fastest steps to create the CAD model. Table 2.1 shows some of these clues.

Example 2.1 What is the best modeling plan to use to create the CAD model shown in Figure 2.2? Why?

Solution The key modeling concept here is how to create the four holes in the block so that they are always equally placed from the four corners of the front face of the block, as shown in Figure 2.2; that is, the holes are always equally placed diagonally from the center of the front face. Such placement should hold even after you change the size of the block (dimensions 1.25 and 1.00) or the spacing (dimensions 0.80 and 0.40) between the holes.

We have two modeling plans to choose from: cross section or features. We use the cross section here. Figure 2.2 shows the modeling plan, which has two steps: create the block cross section, and then create the block feature.

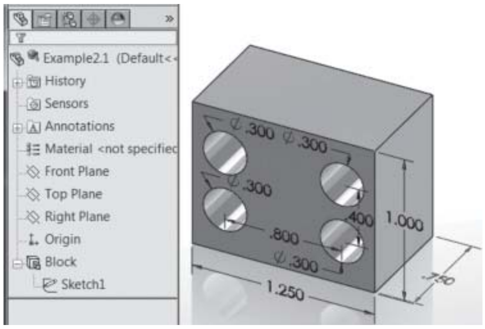
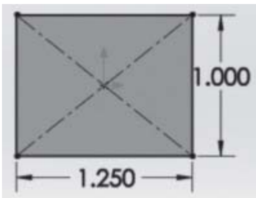
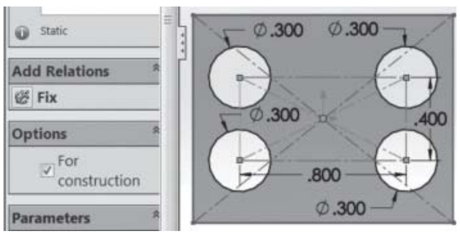


FIGURE 2.2
Plate modeling plan shown in its features tree

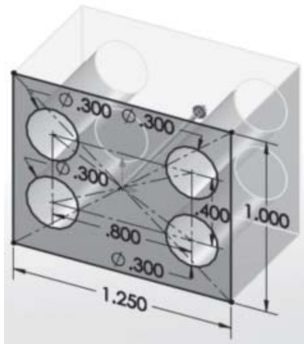
Step 1: Create *Sketch1*: **File > New > Part > OK > Front Plane > Extruded Boss/Base** on **Features** tab **> Center Rectangle** on **Sketch** tab **>** click origin and drag to sketch **> Smart Dimension** on **Sketch** tab > dimension as shown.



Step 2: Create *the four holes*: While *sketch1* is still open, click **Center Rectangle** on **Sketch** tab > click origin and drag to sketch **>** select **For Construction** checkbox **> Circle** on **Sketch** tab > click at each of the four corners as shown to create four circles > **Smart Dimension** on **Sketch** tab > dimension as shown > ✔.



Step 3: Create *Block* feature: **Exit** the sketch > enter 0.75 for thickness (**D1**) > reverse extrusion direction > ✔. SolidWorks creates a node in the features tree called *Boss-Extrude1*. Click it once and rename it *Block*.



Save the part as *plate*: **File > Save As** > *example2.1* > **Save.**

2.4 Part Topology

The CAD models (parts) we create using CAD/CAM systems are solid models. A **solid model** is a geometric model of a part and represents the most complete definition of the part. Figure 2.3 shows the solid model of a block. A solid model is also known as a body (B). Topologically, a solid model consists of faces (F), edges (E), vertices (V), loops (L), and genus (G). A **face** is a surface and may be planar or nonplanar. An **edge** is a curve that may be a line. A **vertex** is a point (corner). Faces meet (intersect) at edges and edges meet (intersect) at vertices. A **loop** is a set of contiguous edges in a face. A loop is viewed as a hole in a face. For example, the front face shown in Figure 2.3 has four loops. A **genus** is a through hole in a solid. A genus is viewed as a 3D hole. For example, the solid model shown in Figure 2.3 has four genus.

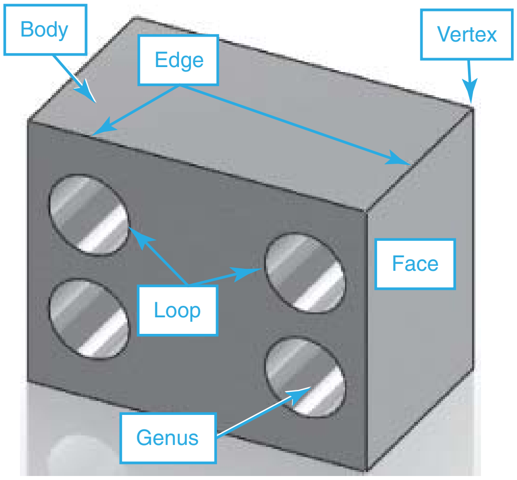


FIGURE 2.3
Topology of a block

The above model description is known as the model topology. The topology of a valid (correct) solid model must satisfy the Euler equation given by:

$$F - E + V - L = 2(B - G) \tag{2.1}$$

The solid model shown in Figure 2.3 has the following topology: $F = 6$, $E = 12$, $V = 8$, $L = 8$, $B = 1$, and $G = 4$. This topology satisfies Eq. (2.1). Therefore the model is a valid model. Obviously, CAD/CAM systems create only valid solid models.

2.5 Parametric Modeling

CAD modeling is based on the concept of parametric modeling (parametrics). CAD/CAM systems use parameters to define the model, instead of specific explicit dimensions. When we assign values to the parameters of a model, these values define the specific size of the model and become the model dimensions that a CAD system uses to generate the model drawings. You may change the values of the parameters later, and the CAD system will update the model to reflect the modifications.

The great virtue of parametric modeling is ease of editing. Figure 2.4 shows the difference between parameters and dimensions for a simple rectangle sketch. A **parameter** is a variable that can assume any value. A **dimension** is a specific numerical value for a parameter. Parametric modeling offers three benefits. First, you sketch freely in a sketch

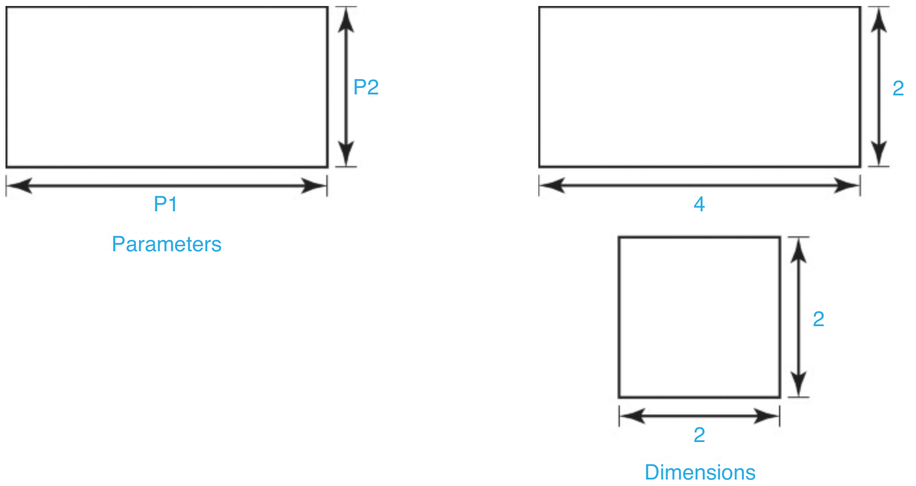


FIGURE 2.4
Parameters and dimensions of a sketch

plane. As you sketch, the CAD system creates parameters and assigns them values (dimensions). When done sketching, you may edit the dimensions as needed. You can display the names of the parameters and edit their names. The CAD system stores and manages the parameters and their values. Second, as Figure 2.4 shows, parametric modeling provides the ability to create a family of parts in which all the parts have identical shapes (topology) with different dimensions. Third, you can create relations between parameters to control the geometry of the sketch. These relations add design intent or intelligence to the model design.

The concept of parametric modeling is liberating during conceptual design; it gives you, the designer, incredible flexibility and power. All you need to do is sketch the model without worrying about dimensions. After you are done sketching, you assign dimensions to the sketch entities. If you do not like the way the sketch looks, you change the dimensions. Thus, we can start sketching and modeling without a need for dimensions, or with just minimal dimensions up front. You can actually use the sketch to define the appropriate values for the missing dimensions.

Example 2.2 Use parametric modeling. Figure 2.5 shows a sketch.

1. Create the sketch in SolidWorks.
2. Modify the dimensions of the sketch.
3. Change the dimension names.

Solution This example illustrates showing and editing dimension names. Figure 2.5 shows the default and the edited dimension names.

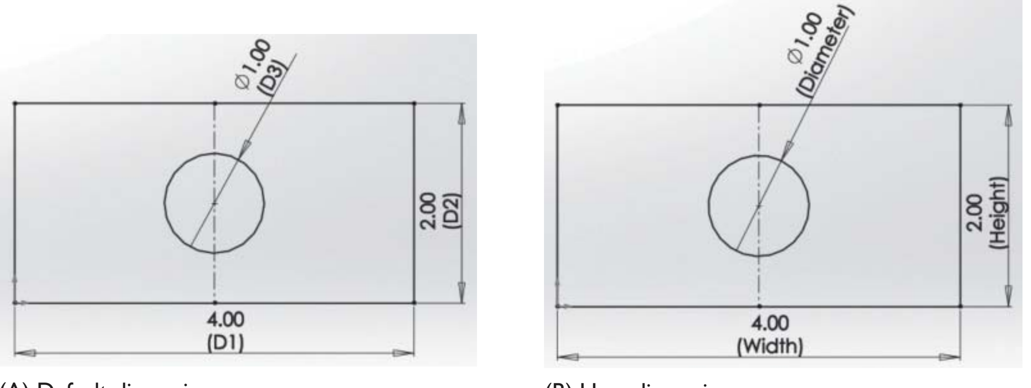


FIGURE 2.5
Dimension names

(A) Default dimension names

(B) User dimension names

Step 1: Create *Sketch1*: **File** > **New** > **Part** > **OK** > **Front Plane** > **Sketch** tab > **Rectangle** on **Sketch** tab > click somewhere and drag to sketch > **Line** on **Sketch** tab > create a vertical line passing through the midpoints of the rectangle sides as shown above > check **For Construction** box > **Circle** on **Sketch** tab > click the midpoint of the construction line and drag to draw a circle > **Smart Dimension** on **Sketch** tab > dimension as shown in Figure 2.5A. Save sketch by clicking **File** > **Save As** > *example2.2* > **Save.**

Step 2: View and edit dimension names: **View** > **Dimension Names** to display the default names of the dimensions. To change a name, double click it > type the desired name (Figure 2.5B) > ✔.

Note: **View** > **Dimension Names** is a toggle; click once to display names, click again to hide them.

Note: The default names are sequential starting with **D1** and created in the order you dimension the sketch. Figure 2.5A above indicates that the user dimensioned the rectangle width first, followed by the height, and finally by the circle. Also, the sketch name is appended to the names. You see it when you edit a dimension.

2.6 Customizing SolidWorks

SolidWorks allows you to customize just about anything in its main window. Click **Tools** > **Customize** to open the SolidWorks **Customize** window shown in Figure 2.6. We encourage you to explore the many options and tabs of the window. In addition, the

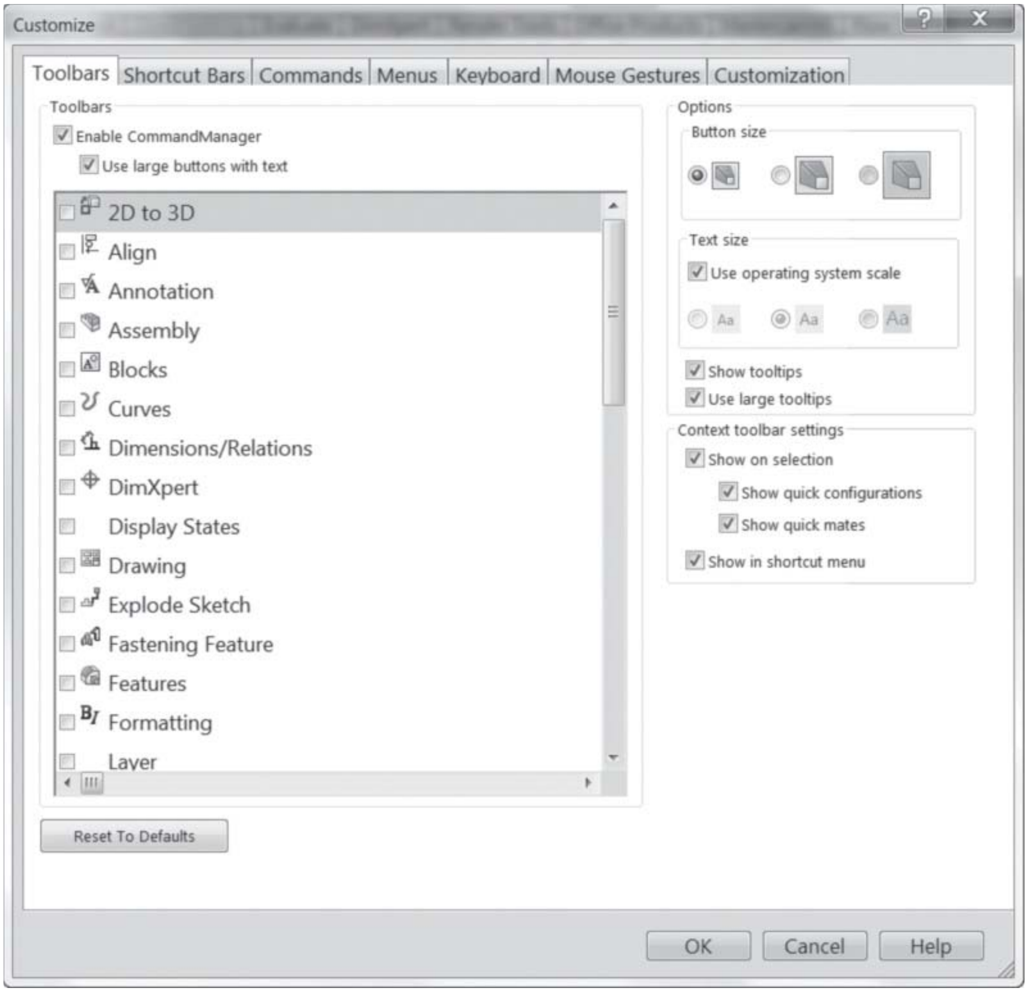


FIGURE 2.6
Customize window (click **Tools** > **Customize** to open)

last menu item on each menu is **Customize Menu**. When you click this item, the items of the menu you are in appear again, this time with the ability to turn them on/off. Now, turn on/off any menu items. When done, click anywhere on the screen to exit. Now, click on the menu again to see the results of the changes you just made.

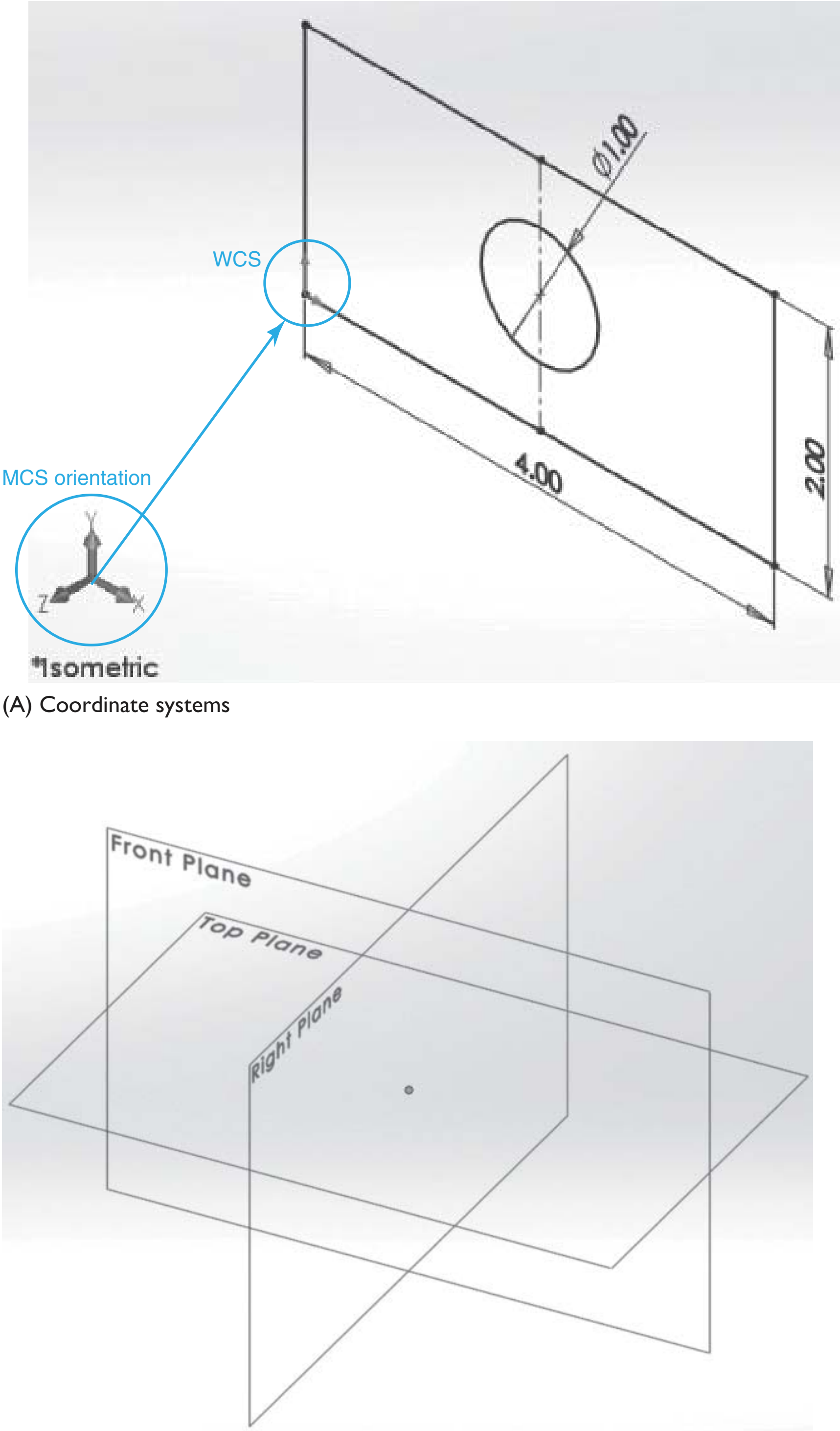
2.7 Productivity Tools

We define a **productivity tool** as an operation or function that speeds accomplishing your design and other tasks in SolidWorks. These tools are diverse and include customizing SolidWorks, hotkeys, programmable mouse, templates, layers, visualization, macros, family of parts, and libraries. The **hotkeys** are the key strokes that correspond to commands. For example, like Word, **Ctrl+S** saves the document and **Ctrl+P** prints the document. You see these hotkeys next to the commands of any menu. As you learn the commands, you learn the hotkeys. A **programmable mouse** is a mouse with multiple buttons where a CAD designer may program each button to do a CAD task. A button may perform one command or a sequence of commands. Clicking a mouse button may be faster for some people than using the hotkeys on a keyboard, or clicking items from menus or icons on toolbars.

2.8 Coordinate Systems

CAD/CAM systems use two coordinate systems to make part modeling easy and convenient for CAD designers. The first system is the model coordinate system (MCS). The **MCS** is the reference system that the CAD/CAM system uses to store the model geometry in the part file. The MCS is defined by the CAD/CAM system, and its location and orientation in space cannot be changed by the user. A CAD/CAM system uses its MCS to define its default (predefined) sketch planes (top, front, and right) and its default (predefined) views (top, front, right, and others). The MCS is an orthogonal coordinate system defined by three orthogonal planes that intersect at the system axes: *X*, *Y*, and *Z*. An MCS assumes one of two orientations in a CAD/CAM system. In one, the XY plane of the MCS defines the horizontal plane (top sketch plane) and the top view; in the other, the XY plane defines the vertical plane (front sketch plane) and the front view. SolidWorks uses the latter orientation for its MCS, as shown in Figure 2.7A.

The second system is the working coordinate system (WCS). The **WCS** is a coordinate system that facilitates model creation. It is a user-defined system. The user may define a WCS anytime during modeling. However, only one WCS is active at any one time. The sketch plane is the XY plane of the current WCS. As the user sketches on the sketch plane, the CAD/CAM system keeps track of the transformation back and forth between the WCS and MCS coordinates. It converts the WCS coordinates into MCS coordinates and stores them in the part file. The WCS is related to the MCS via the location of its origin and its orientation, as shown in Figure 2.7A. Note that the origins of the MCS and WCS shown in Figure 2.7A are actually coincident and are located at the origin of the WCS. We use the shown MCS for illustrative purposes only. The CAD/CAM system displays the *X*- and *Y*-axes of the WCS only. The *Z*-axis is perpendicular to the XY plane with a positive direction defined by the right-hand rule. In terms of the day-to-day jargon of CAD/CAM systems, we hardly use the word *WCS*. We always use sketch planes instead. The default WCS of a CAD/CAM system has the same orientation and location as its MCS. In the case of SolidWorks, the default WCS is the one that uses the front sketch plane shown in Figure 2.7B.



(A) Coordinate systems

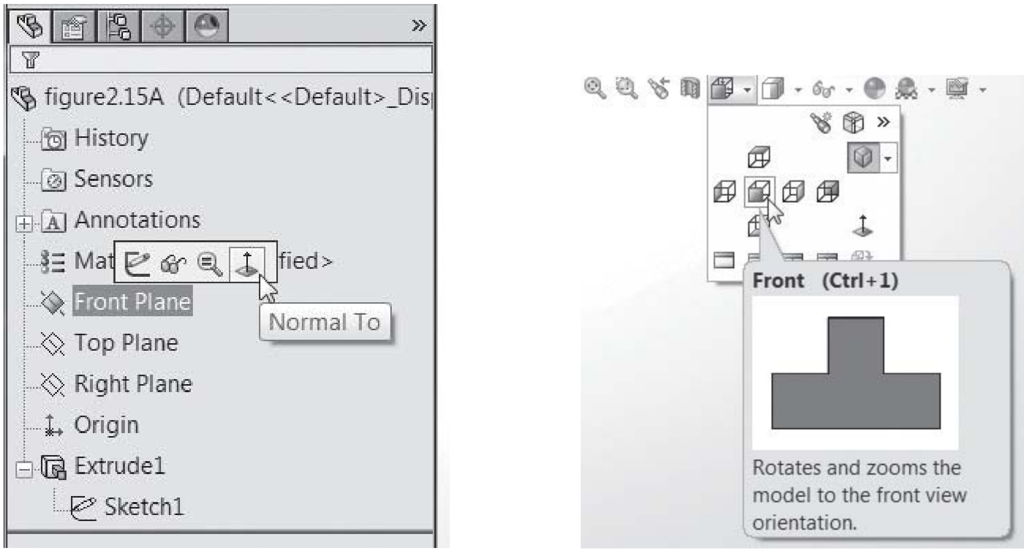
(B) Default sketch planes

FIGURE 2.7
SolidWorks coordinate systems and default sketch planes

2.9 Sketch Planes

Sketch planes control the part creation in the CAD/CAM 3D modeling space. You always begin by selecting a sketch plane to create the cross section (sketch) of the feature you wish to create. The sketch is the basis of 3D modeling. If you need multiple sketch planes to create a 3D model, you select one at a time, create the sketch, and finish the feature before selecting another sketch plane to create another feature. The three default sketch planes are **Front Plane**, **Top Plane**, and **Right Plane** (see Figure 2.7B). If you need a sketch plane in a different orientation, you must create it. You can also select any face of any feature as a sketch plane. Simply click the face and open it as a sketch to begin sketching.

When you sketch, you want the sketch plane to be perpendicular to the line of sight; that is, the sketch plane aligns with the screen of your computer monitor to enable you to see the sketch entities you create. This would require you to align the view with the sketch plane. For example, if you want to sketch on the front sketch plane, select **Front Plane** from the features tree > click the **Normal To** icon (hover until you read it) from the context toolbar that pops up, shown in Figure 2.8A. Alternatively, you can select the front view from the **Heads-up View** toolbar as shown in Figure 2.8B.



(A) Select **Normal To** from the context menu.

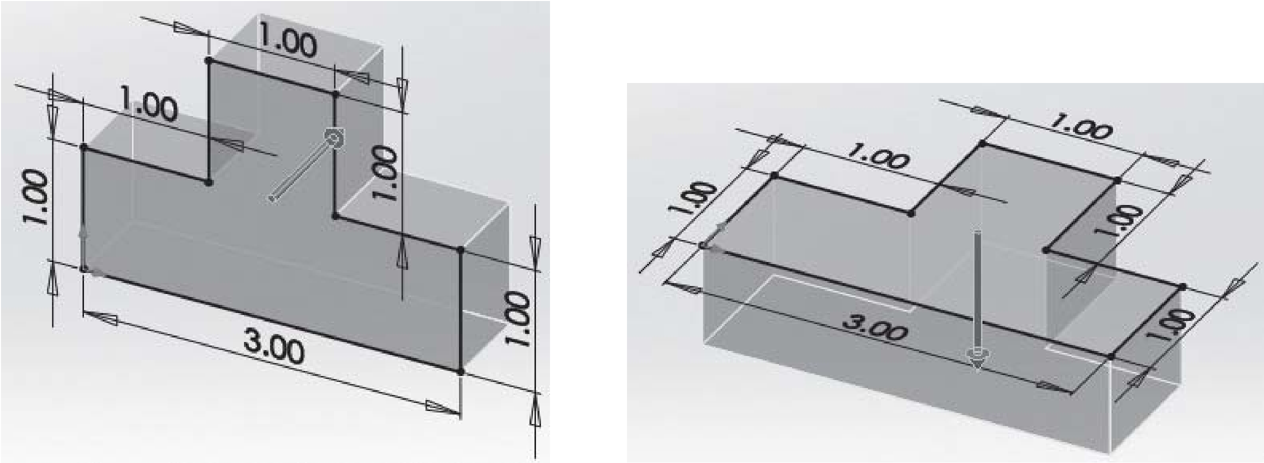
(B) Select view from the screen.

FIGURE 2.8
Orient sketch plane with model viewing

A sketch plane could be an imaginary plane, like the ones SolidWorks provides (Front, Top, or Right), or it could be an actual face of a feature. In either case, select and orient it to begin sketching on it. If a feature's face is not directly accessible to select it, change the view orientation or rotate the model until you can select the face.

The very first sketch plane that you use to create a feature determines the model orientation in the 3D modeling space. This, in turn, affects the views of the model that you create in engineering drawings. You need to align the orientation, in the 3D modeling space, of the 3D model of a part with the physical or perceived orientation of the actual part. You simply use the corresponding sketch plane to create the cross section of the part; that is, you use the front sketch plane to create the part front cross section, the top sketch plane to create the top cross section, and so on. If you use the front sketch plane to create the top cross section, you effectively have rotated the model 90 degrees in the 3D modeling space. Thus, the top view of the physical part becomes the front view of the model. Figure 2.9 shows an example.

There are ways to get around misorientation of the model in the 3D modeling space due to starting with the incorrect sketch plane. One (inefficient) way is to delete the model and re-create it using the correct sketch plane. A better way is to reorient the

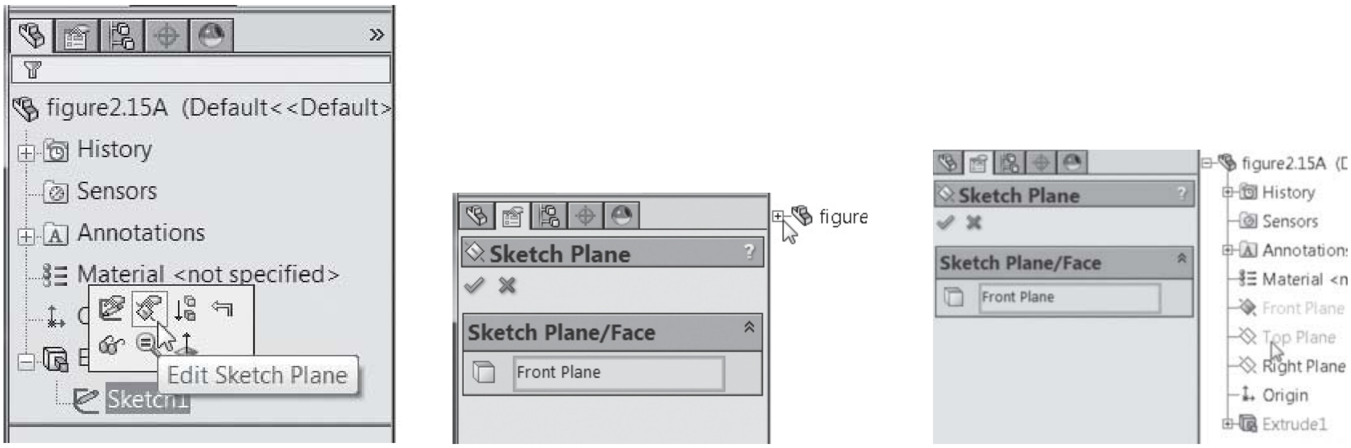


(A) Part orientation using **Front Plane**

(B) Part orientation using **Top Plane**

FIGURE 2.9
Effect of first sketch plane on part orientation in 3D modeling space

model in the space as follows and shown in Figure 2.10. Click the sketch in the features tree (Figure 2.10A) > select **Edit Sketch Plane** from the context menu that pops up (Figure 2.10A) > expand the features tree (click the "+" symbol shown in Figure 2.10B) > select the plane you want such as **Top Plane** (Figure 2.10C) > ✔. The model view on the screen confirms the sketch plane reorientation. Drawing views also confirm the results.



(A) Select sketch plane to edit.

(B) Expand the features tree.

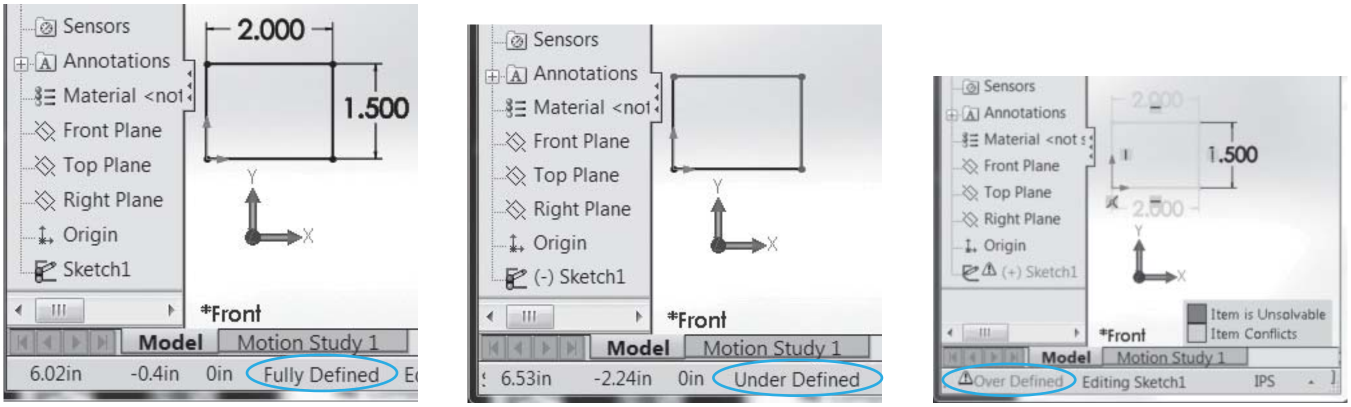
(C) Select the new desired sketch plane.

FIGURE 2.10
Editing the sketch plane of a sketch to change its orientation

2.10 Sketch Status

A **sketch status** is defined as the geometric state of a sketch. After you create a sketch, you may add dimensions to its parameters before you create the feature that uses the sketch. The sketch status assumes one of three states. A **fully defined** sketch means the sketch is dimensioned and constrained correctly. This is the ideal status. An **under defined** sketch lacks necessary dimensions and/or geometric constraints. An **over defined** sketch has more dimensions/constraints than what it needs. SolidWorks allows you to create a feature using an under or over defined sketch. You should not create a feature without a fully defined sketch, although you could.

SolidWorks uses color and text to convey the sketch status to the designer. It also displays a symbol in the features tree next to the sketch name if the sketch is under (–) or over (+) defined as shown in Figure 2.11B and Figure 2.11C respectively. For the over defined sketch, SolidWorks also displays a yellow triangle (Figure 2.11C) next to the (+) symbol. If all sketch entities are displayed in black on the screen, the sketch is fully defined and no symbol appears next to its name in the features tree as shown in Figure 2.11A. Entities that are under defined are displayed in blue. Yellow areas show where the sketch is over defined. In the case of Figure 2.11B, the blue entities (right and top lines of the rectangle) are under defined because they are not dimensioned. In the case of Figure 2.11C, the bottom and top horizontal lines are displayed in yellow, indicating that one of their dimensions must be removed because it is driven by the other (i.e., redundant).



(A) Fully defined sketch (B) Under defined sketch (C) Over defined sketch

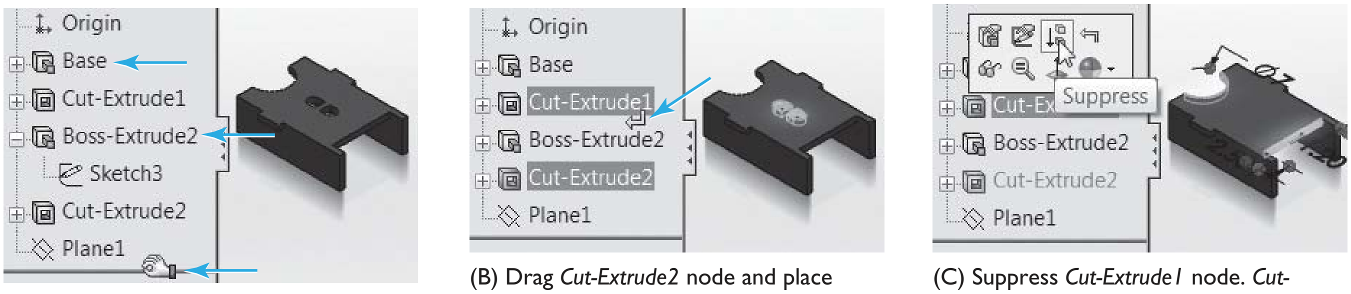
FIGURE 2.11
Sketch status

2.11 Part Features Tree

A part features tree (SolidWorks calls it **FeatureManager Design Tree**) is an important modeling tool. The tree is an uprooted tree. The top node (root) is the part (CAD model) under which the other nodes come. Each node has leaves, which are the sketches. The tree shows the steps (history) of creating the part. You can step through the tree to reverse engineer the part design by rolling back the tree.

If you right-click any tree node, you can perform useful functions. Click any feature to display its dimensions. View the parent/child relationships by right-clicking a feature > **Parent/Child** from the pop-up. You can delete a sketch or feature by right-clicking its node in the tree > **Delete** from the pop-up. Deleting a feature does not delete its sketch. Conversely, you cannot delete a sketch unless you delete its feature first. Investigate the many possibilities of using the features tree when you need them. Here are some more thoughts on how to use the tree (Figure 2.12):

1. Expand or collapse nodes to have access to or to hide sketches of features. A node that is collapsible has a "+" (collapsed state) or "–" (expanded state) symbol, as shown in Figure 2.12. Click a node symbol to toggle its state.
2. Rename a node (Figure 2.12A) for ease of following the part creation steps.
3. Rearrange nodes (Figure 2.12B): drag a node and move it up or down the tree in case you ever want to change the order of the creation steps. You can do that as long as you do not violate the part topology.
4. Roll back the tree. Drag the bottom blue line (rollback bar) of the tree up to hide features or down to show them again. Place the mouse over the rollback bar (hand



(A) Rename/Expand node, roll back tree.

(B) Drag *Cut-Extrude2* node and place below *Cut-Extrude1* node.

(C) Suppress *Cut-Extrude1* node. *Cut-Extrude2* is suppressed.

FIGURE 2.12
Using part features tree

shown in Figure 2.12A) and drag it up or down. When you roll back the tree, you hide the features from the bottom of the tree up to the rollback position. This is beneficial because it allows you to insert features at any location in the tree. When done with the insertion, drag the rollback bar all the way down the tree.

5. Suppress a node (Figure 2.12C): you can hide (suppress) a feature (node) without having to delete it and re-create it. This could be useful in design studies. Unlike rolling back a tree, suppressing a node enables you to hide a node selectively without having to hide all the tree nodes that follow underneath it. Suppressed nodes are displayed gray in the features tree. To suppress a node: click it in the tree > **Suppress** from the context toolbar (hover over until you read it). To unsuppress it, click it again > **Unsuppress** from the context toolbar.

2.12 Construction Geometry

When you create a sketch, you may need auxiliary geometry to assist in creating the sketch entities. Such geometry is known as *construction* geometry. Any entity you sketch can become construction geometry if you check off the **For construction** box (option) during sketching as shown in Example 2.1. Select the option after you create the entity. If you check the option again (uncheck the box), the entity reverts back to a sketch entity. When you sketch a line, you have the option to create a centerline under the line icon on the **Sketch** tab; alternatively, you can sketch a line and then check off its box. If required, you can edit a sketch and convert any of its entities to **For construction** by selecting first.

Points and centerlines are always construction entities. Construction geometry uses the same line style as centerlines. Examples of using construction geometry include an axis of revolution to create a revolve feature, a circular axis to place holes (circular pattern) in a flange, and so on. Example 2.1 shows a construction rectangle to place the four holes of the block.

2.13 Reference Geometry

Reference geometry, as construction geometry, is part of a feature definition, but not part of the feature geometry. Whereas construction geometry belongs to sketch creation, reference geometry belongs to feature creation. As such, construction geometry (e.g., centerline) is available on the **Sketch** tab, whereas reference geometry is available on the **Features** tab under the **Reference Geometry** drop down shown in Figure 2.13. The figure shows the reference geometry that you can create and use. **Plane** and **Point** are the most commonly used reference geometry. We already used a plane in Tutorial 1.3. Reference points are useful to use to create a curve.

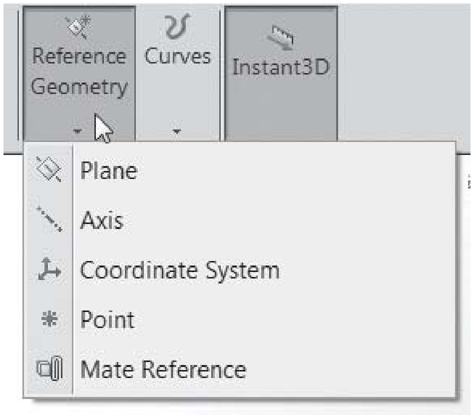


FIGURE 2.13
Reference geometry

Example 2.3 Create the part shown in Figure 2.14

Solution The goal of this example is to show how to create inclined sketch plans and use them in modeling. The part has five features: block, cylinder, slot, cylinder fillet, and slot fillet. To create a plane at an angle from a given face, you need to identify an edge on the face to measure the angle from. Also, when you click a plane node in the features tree, you can use the eyeglasses symbol from the context toolbar that pops up (and shown in Step 3 below) to hide/show the plane. It is a toggle.

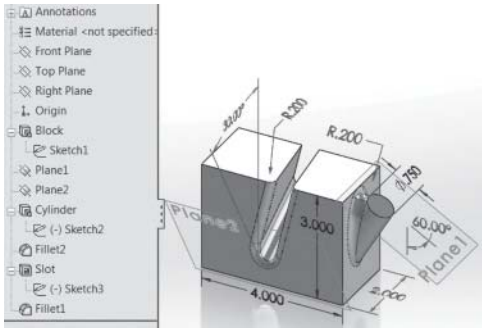
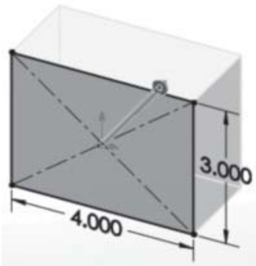


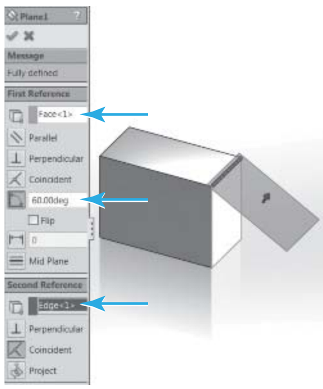
FIGURE 2.14
Block with inclined planes

Step 1: Create *Sketch1* and *Block* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and drag to sketch > dimensions cross section as shown > exit sketch > enter 2.0 for thickness (**D1**) > reverse extrusion direction (drag extrusion arrow) > ✔ > click tree node and rename it *Block*.



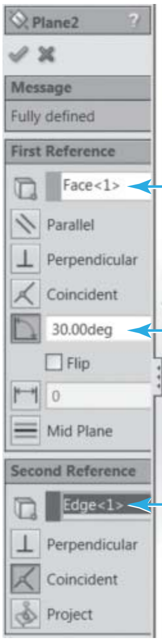
Save the part as *example2.3:* **File** > **Save As** > *example2.3* > **Save.**

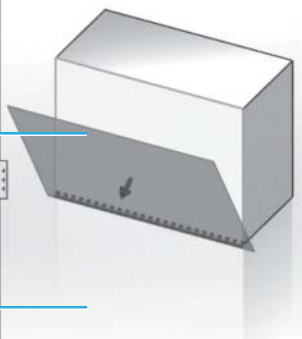
Step 2: Create *Plane1*: **Features** tab > **Reference Geometry** > **Plane** > click right face > click top edge of right face > enter 60 for angle > ✔.

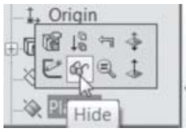


Step 3: Create *Plane2*: **Features** tab > **Reference Geometry** > **Plane** > click front face > click bottom edge of front face > enter 30 for angle > ✔.

Note: Hide/show *plane2* node.

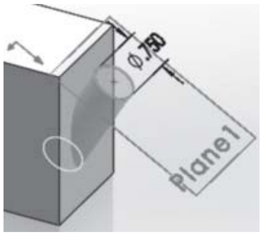




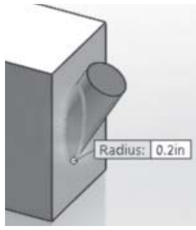


Step 4: Create *Sketch2*, *Cylinder* and *Fillet1*:

Cylinder: Plane1 > **Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click somewhere on *Plane1* > sketch a circle > dimension as shown > exit sketch > enter 2.0 for thickness (**D1**) > ✔ > click tree node and rename it *Cylinder*.



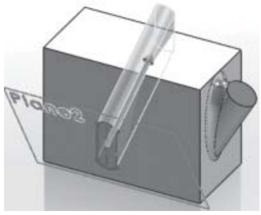
Fillet1: **Features** tab > **Fillet** > click circle shown > enter 0.2 for radius > ✔.



Step 5: Create *Sketch3*, *Slot* and *Fillet2*:

Slot: Plane2 > **Extruded Cut** on the **Features** tab > **Slot** on **Sketch** tab > click somewhere on *Plane2* > drag in one direction, then in another to sketch slot > exit sketch > **Through All** > ✔ > click tree node and rename it *Slot*

Fillet2: **Features** tab > **Fillet** > click slot edge on *Block* front face > enter 0.2 for radius > ✔.



HANDS-ON FOR EXAMPLE 2.3. Sketch2 and Sketch3 are under dimensioned as shown in Figure 2.14. Why? How can you solve the problem?

2.14 Sketch Entities

The SolidWorks **Sketch** tab provides all the entities you can use to create parts. Figure 2.15 shows the entities. Hover over any entity or click a drop down list to investigate further. Most of the entities are self-explanatory. We just offer a few remarks. The **Ellipse** menu has a **Parabola** entity that you can use to create parabolas. Use the **Spline** menu to create any shaped curve. A **spline** curve is defined as a general-shaped curve that provides freehand sketching ability. To create a spline, click the **Spline** icon, and then click on the screen at multiple locations. When done, hit **Esc** on the keyboard. Use the **Text** icon (shown as A in Figure 2.15) to create text on faces, curves, edges, and sketch entities. This text is treated as modeling entities and can be, for example, extruded. It is unlike text you create in drawings.

FIGURE 2.15
Sketch entities

2.15 Sketch Relations

A sketch **relation** is a geometric constraint (condition) between two sketch entities such as lines or arcs. You can add relations while creating the entities or after. If you want to add relations after creating entities, you must edit them. These relations include horizontal, vertical, perpendicular, coincident, midpoint (of a line), and so on. As you sketch, SolidWorks guides you by showing a symbol for the most logical relation you can use (see Figure 1.6). If you hover over a relation symbol after sketching, SolidWorks displays its name. You can turn on/off the display of relation symbols in a sketch by clicking: **View** > **Sketch Relations**. It is a toggle.

2.16 Equations and Link Values

An **equation** is a mathematical relationship between sketch entities, feature dimensions, or other model properties. Parameters form the basis for creating equations. An equation could relate two or more parameters. For example, we can write an equation to relate the two parameters $P1$ and $P2$ (e.g., $P1 = 2P2$). For an equation to evaluate correctly, the values for all the parameters on the right side of the equation must be known ($P2$ in this case). $P1$ is the evaluated parameter. $P1$ is also known as the dependent or driven parameter, and $P2$ is the independent or driving parameter (dimension). Note that you cannot edit driven parameters to change their values because they are controlled by their equations. You can only change the values of the driving parameters.

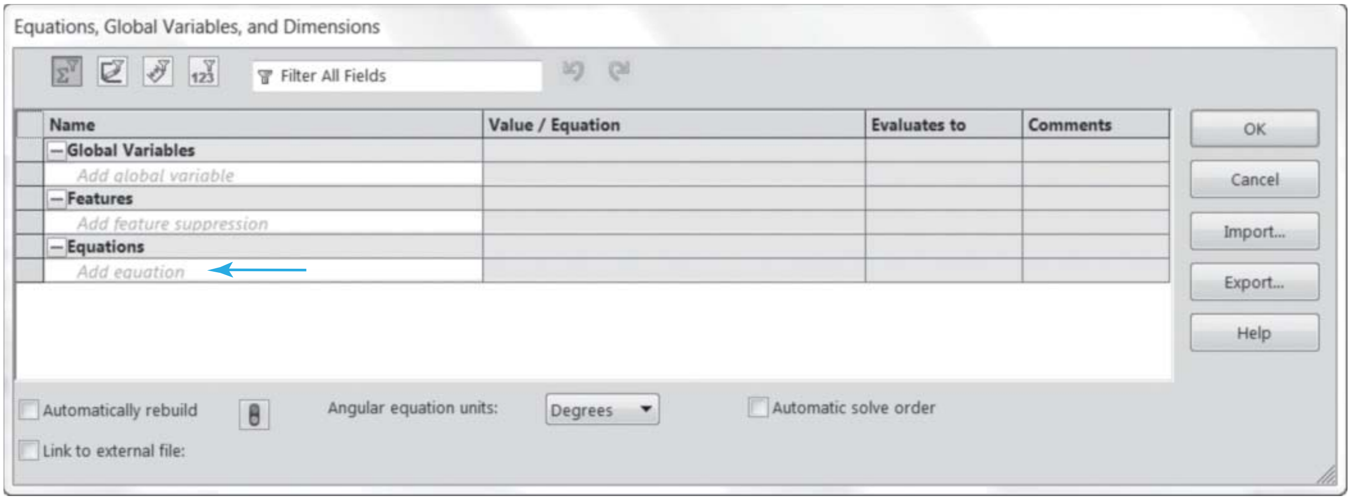
Link values work in a similar, but simpler way as compared to equations. They allow us to link two parameters together to have the same value, effectively creating an equality equation (e.g., $D1 = D2$). Changing the value of one parameter changes the other. In link values, we can change either parameter. Unlike equations, there is no driver or driven parameter. Note that we cannot link parameters that are defined by equations to prevent conflicts. You link parameters by assigning them a shared name (e.g., length). Right click any dimension > **Link values** > enter shared name > **OK**. To unlink a parameter, right click it > **Unlink Value**.

Example 2.4 Use equations and link values to control modeling. This example builds on Example 2.2. We create the sketch in a different way by using equations. We need to control the rectangle shape so that the value of *D1* is always twice that of *D2* and the hole center is always coincident with the rectangle center. Create the equations to implement these relations. Prove that the equations work by changing the dimensions.

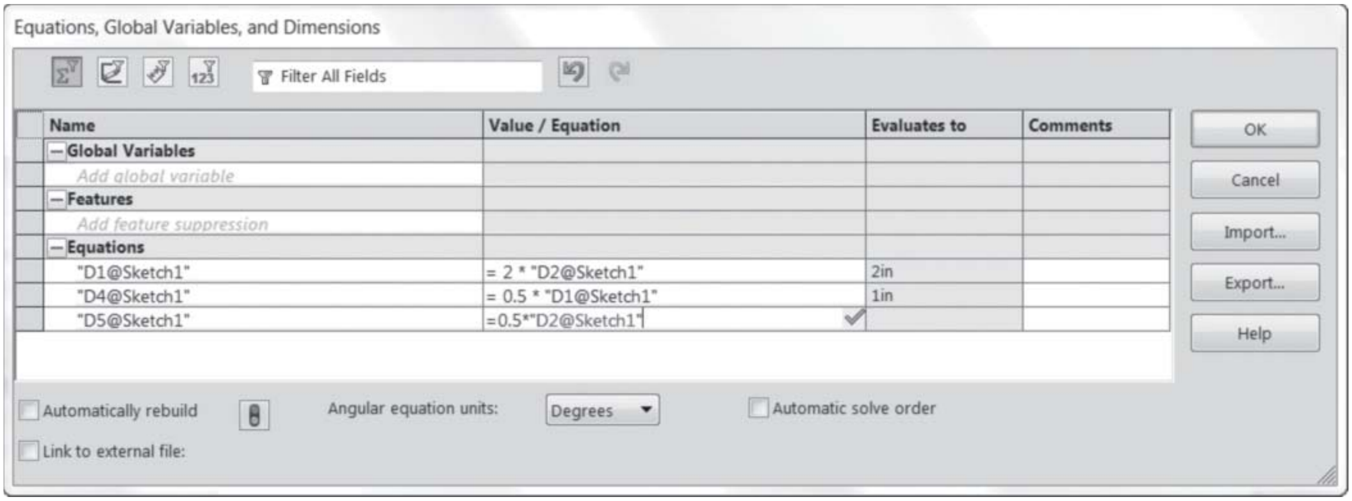
Solution The coincidence of the hole and the rectangle centers can be forced by dimensioning the hole center (x_{hole}, y_{hole}) from the rectangle's two edges. The equations we need can be written as follows:

$$D1 = 2D2$$
$$x_{hole} = 0.5D1$$
$$y_{hole} = 0.5D2$$

Figure 2.16 shows the equation editor. x_{hole} and y_{hole} coordinates of the hole (circle) center are shown as **D4** and **D5** respectively in the equation editor and on the sketch. To delete an equation, right click its row > **Delete Equation** from popup. To edit it, click the column (Figure 2.16B) you want to edit.

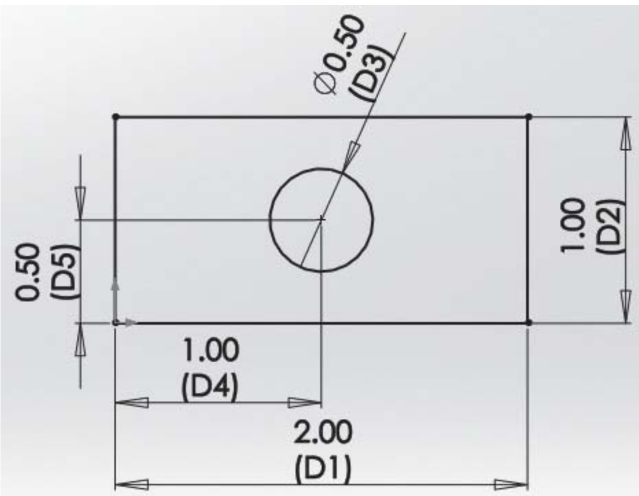


(A) No equations



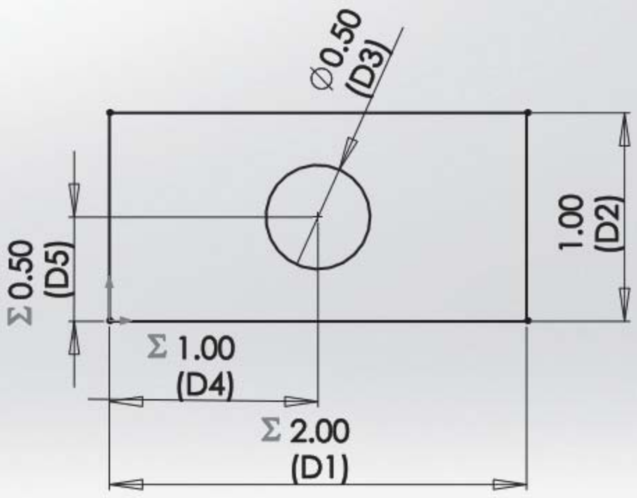
(B) Sample equations

FIGURE 2.16
Equations, Global Variables, and Dimensions window



Step 1: Create *Sketch1* and view dimension names: **File** > **New** > **Part** > **OK** > **Front Plane** > **Sketch** tab > **Rectangle** on **Sketch** tab > sketch rectangle > **Circle** on **Sketch** tab > sketch circle > **Smart Dimension** on **Sketch** tab > dimension as shown. Save sketch by clicking **File** > **Save As** > *example2.4* > **Save.**

Step 2: Create equations: **Tools** > **Equations** (to open **Equations** window shown in Figure 2.16A) > place cursor in the field indicated by the arrow shown in Figure 2.16A > click **D1** (cursor moves to next column, **Value / Equation**) > type 0.5* > hit **Enter** on keyboard > click **D2** from sketch > ✔ repeat for other

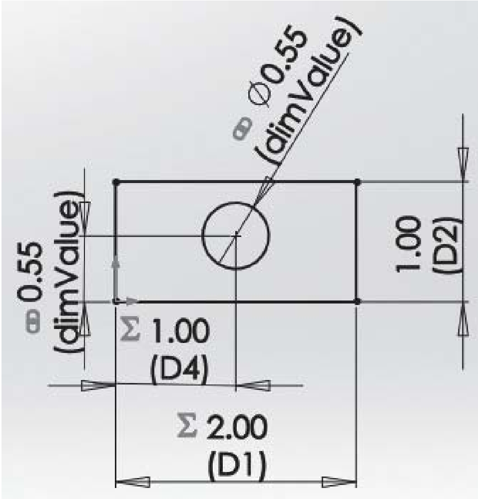


equations > **OK** to finish. You should see equation symbols (summation sign) as shown above.

To test the equations, double click **D2** and change its value to 2, and observe **D1**, **D4,** and **D5** change and that the hole center stays centered.

Step 3: Link *D3* and *D5*: Double click *D5* parameter shown in Step 2 > delete the equation from the **Modify** box that opens up > right click *D5* > **Link Values** from the popup > type *dimValue* in the box that opens up > **OK** > right click *D3* > **Link Values** from the popup > type *dimValue* in the box that opens up > **OK**. Observe the different symbols for equations (sum sign) and linked values (chain sign) shown below.

To test the link, double click *D3* and change its value to 0.55 and observe *D5*. Also, double click *D5* and change it back to 0.5, and observe *D3*.



HANDS-ON FOR EXAMPLE 2.4. The sketch of this example has a problem. A user can make the hole diameter, *D3*, larger than the rectangle height, *D2*. Add this equation to the sketch to solve this problem: $D3 = 0.5D2$.

2.17 Geometric Modifiers

A **geometric modifier** is a qualifier to select designated points, such as end- or midpoints, on an existing (already created) sketch entity. The benefit of using these modifiers is to speed up entity creation by accessing these points without having to calculate their coordinates explicitly by hand. Geometric modifiers are part of what SolidWorks calls *quick snaps*. Geometric modifiers are **end**, **center**, and **intersection**. The **end** modifier identifies the endpoints of an entity. If the entity is an open curve, like a line, it has two separate endpoints. If the entity is closed, like a circle, it has two coincident endpoints. The **center** modifier identifies the center or midpoint of an entity. For a line, it is the midpoint, and for a circle it is the point located on its circumference at 180° (not its center). The **intersection** modifier identifies where two entities intersect. If more than one intersection point exists, the CAD software picks up the one closest to the user selection. As you sketch, SolidWorks anticipates your next move and displays the expected intent (modifier) if you hover over an existing entity.

2.18 Grids

A **grid** is an equally spaced set of points in either a rectangular or radial pattern. Rectangular grids are more commonly used. A grid has three parameters: spacing along two axes (*X* and *Y*), an origin, and an orientation. Grids are used to speed up construction. Grids are useful in creating sketches with repetitive shapes. Click this sequence to display grids: **Tools > Options > Document Properties** tab > **Grid/Snap** > Check (turn on or off) **Display grid** and set the other properties shown in Figure 2.17A > **OK**. Figure 2.17B shows an example grid. Major grid lines are shown darker than the minor grid lines. The "+" symbol indicates a grid point.

Displaying a grid does not mean that you can snap to its points during construction. To turn on the snap, click **Tools > Options > System Properties** tab > **Relations/Snaps** (under **Sketch** item) > check (turn on or off) **Grid > OK**. Grid lines are visible on the screen only when you are in a sketch. You know the snap is on when the mouse moves in a snappy unexpected way and a "+" symbol shows up when the mouse moves closer to a grid point.

(A) Enable grid display

(B) Grid lines

FIGURE 2.17
Grids

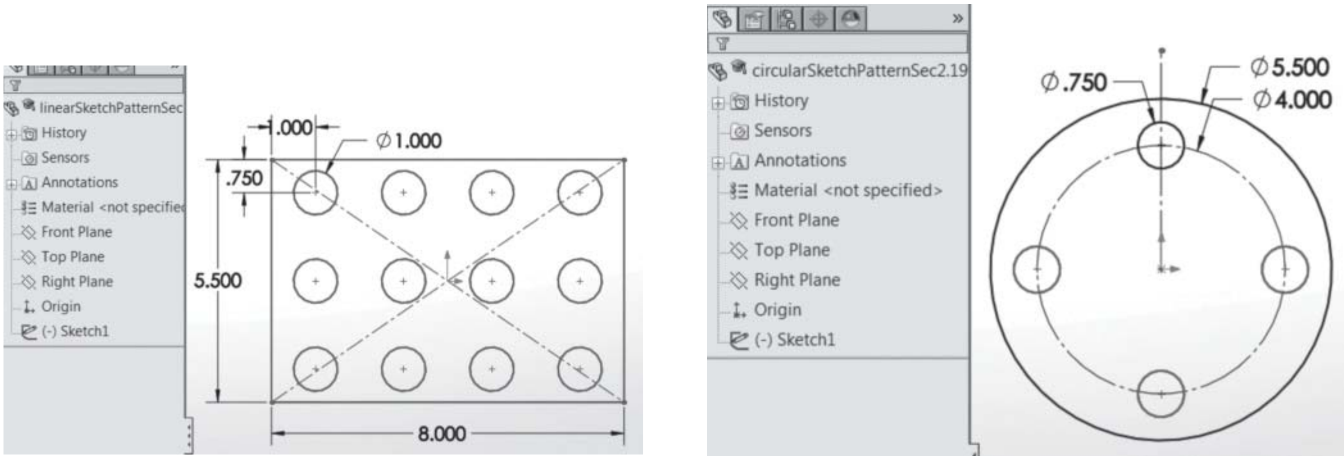
2.19 Patterns

Patterns are also known as *geometric arrays*. A **pattern** is a uniform layout of a sketch entity or a feature in specific directions. Examples include the keys of a phone keypad and the holes in a flange. You may pattern an entity along a curve (known as a *curve-driven pattern*). Two types of patterns exist: rectangular (Cartesian) and circular (angular). A rectangular pattern is also known as a *linear pattern*. The instances of a linear pattern are separated by increments in the X and/or Y directions. The instances of a circular pattern are separated by increments in the angular and/or radial directions.

You may pattern sketch entities or features. You need to be in a sketch to pattern sketch entities and out of a sketch to pattern features.

Example 2.5 Create sketch-based patterns

Solution We show how to create linear and circular sketch patterns that are shown in Figure 2.18. All dimensions are in inches.

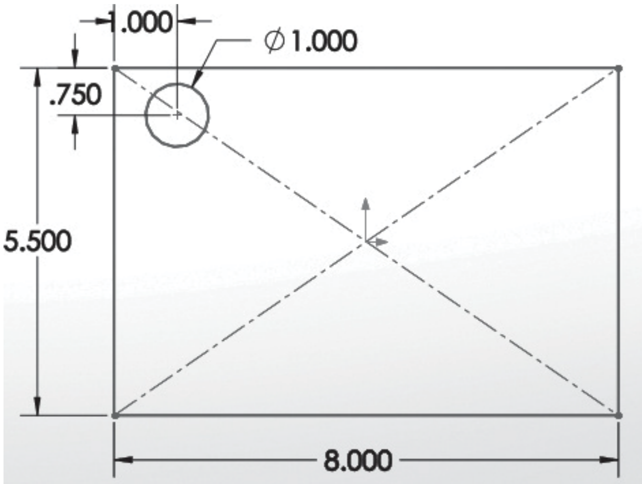


(A) Linear pattern

(B) Circular pattern

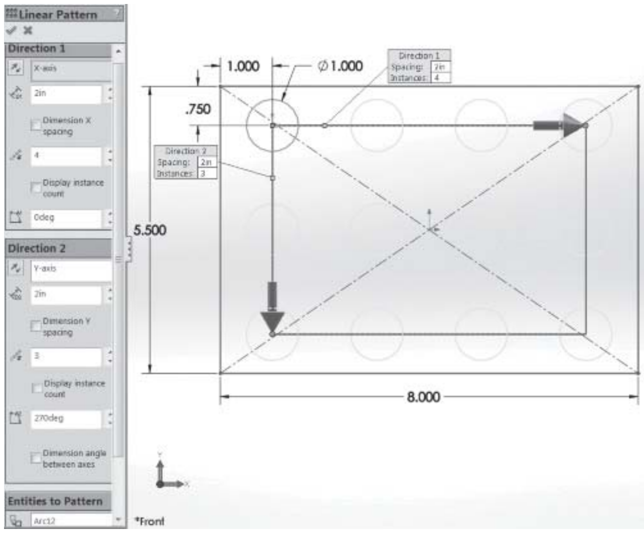
FIGURE 2.18
Sketch patterns

Step 1: Create *Sketch1*: **File** > **New** > **Part** > **OK** > **Front Plane** > **Sketch** tab > **Center Rectangle** on **Sketch** tab > sketch rectangle at origin > **Circle** on **Sketch** tab > sketch circle anywhere > **Smart Dimension** on **Sketch** tab > dimension as shown below. Save sketch by clicking **File** > **Save As** > *example2.7A* > **Save.**

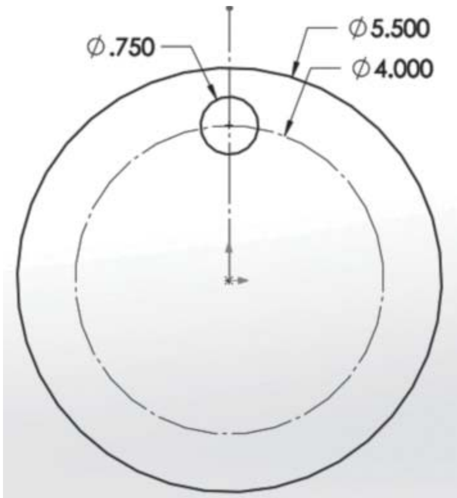


Step 2: Create linear pattern: Select circle > **Linear Sketch Pattern** on **Sketch** tab > input pattern data on left pane as follows > for **Direction 1**: type 2 for **D1** (second box), 4 in third box > for **Direction 2**: type 2 for **D2** (second box), 3 in third box > ✔.

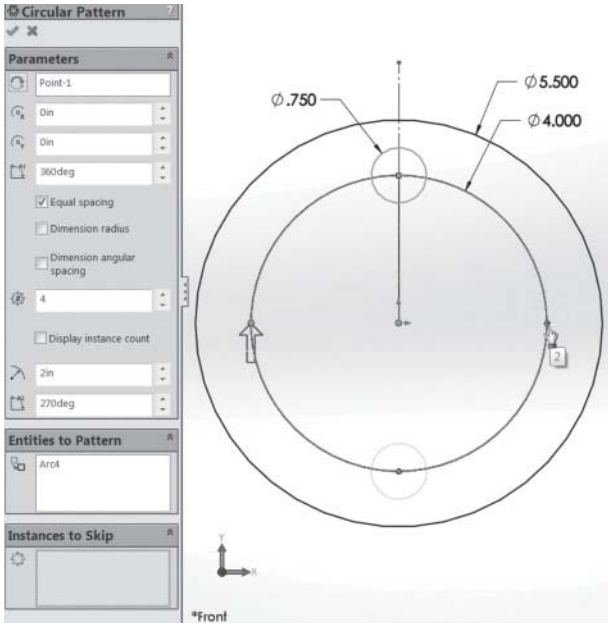
Note: Click the arrows shown to reverse the pattern directions. Follow Step 4 Note to skip instances.



Step 3: Create *Sketch1*: **File** > **New** > **Part** > **OK** > **Front Plane** > **Sketch** tab > **Circle** on **Sketch** tab > sketch outer circle at origin > sketch inner circle at origin and



check **For Construction > Line** on **Sketch** tab > sketch a vertical centerline as shown **> Smart Dimension** on **Sketch** tab > dimension as shown above. Save sketch by clicking **File > Save As** > *example2.7B* > **Save.**

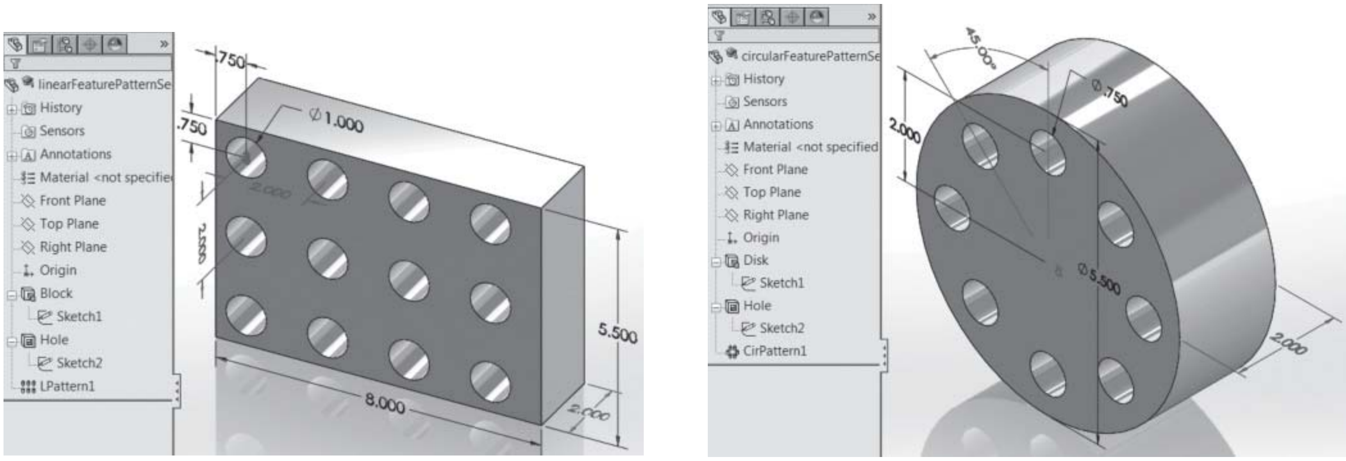


Step 4: Create circular pattern: Select small circle > **Circular Sketch Pattern** from **Linear Sketch Pattern** dropdown on **Sketch** tab > accept pattern default data on left pane > ✔.

Note: Click the arrow shown in center to reverse pattern direction. The pattern instances are numbered. Expand (then click inside) **Instances to Skip** box shown to the right > hover over an instance and click it to skip it.

Example 2.6 Create feature-based patterns

Solution We re-create Example 2.5 linear and circular sketch-based patterns using feature-based patterns. Figure 2.19 shows the patterns. All dimensions are in inches.

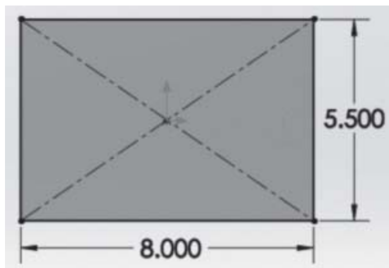


(A) Linear pattern

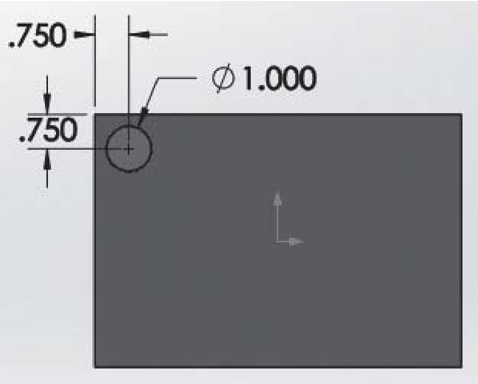
(B) Circular pattern

FIGURE 2.19
Feature patterns

Step 1: Create *Sketch1* and *Block* feature: **File** > **New > Part > OK > Front Plane > Extruded Boss/Base** on **Features** tab **> Center Rectangle** on **Sketch** tab **>** click origin and drag to sketch **>** dimension the cross section as shown to the right > exit sketch > enter 2.0 for thickness (**D1**) > reverse extrusion direction (drag extrusion arrow) > ✔ > click tree node and rename it *Block* **> File > Save As** > *example2.8A* > **Save.**

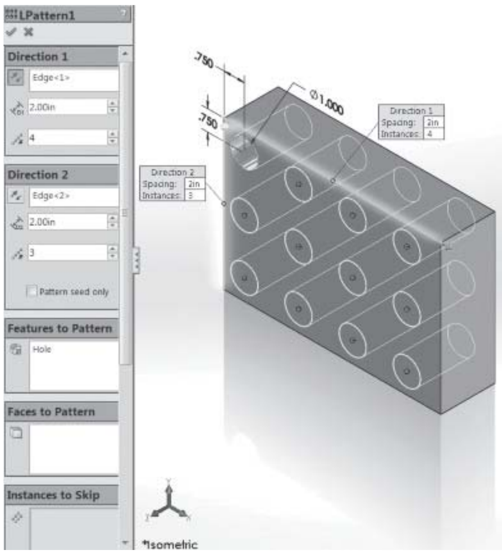


Step 2: Create *Sketch2* and *Hole* feature: **Extruded Cut** on **Features** tab **>** **Circle** on **Sketch** tab **>** sketch circle **>** dimension as shown > exit sketch > **Through All** under **Direction 1** > ✔ > click tree node and rename it *Hole*.

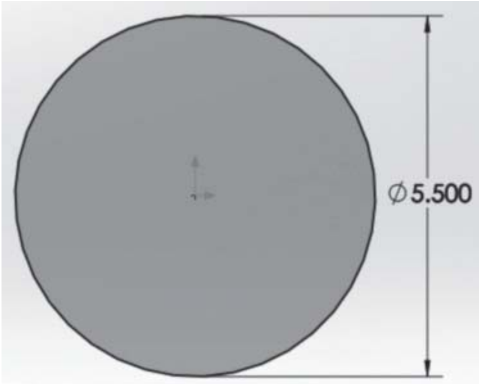


Step 3: Create *LPattern1*: Select *Hole* feature > **Linear Pattern** on **Features** tab > select top edge of *Block* for **X-axis** > input pattern data on left pane as follows > for **Direction 1**: type 2 for **D1** (first box), 4 in second box > select left edge of *Block* for **Y-axis** > input pattern data on left pane as follows > for **Direction 2**: type 2 for **D2** (second box), 3 in third box > ✔.

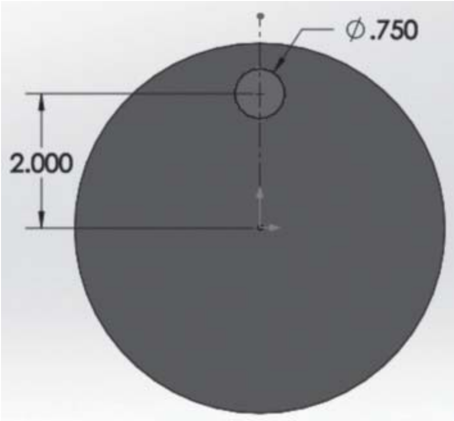
Note: Click the arrows shown to reverse the pattern directions. Follow the Step 6 Note to skip instances.



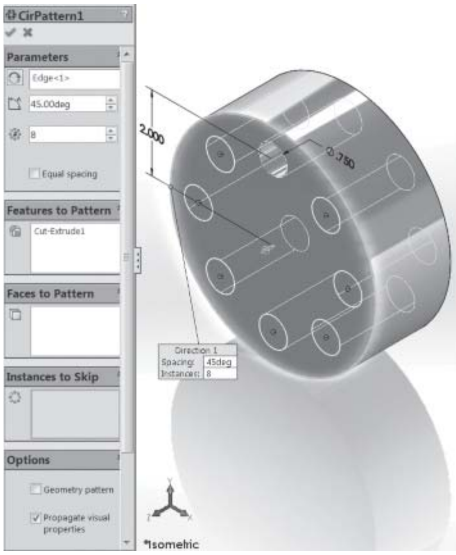
Step 4: Create *Sketch1* and *Disk* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and drag to sketch > dimension circle > exit sketch > enter 2.0 for thickness (**D1**) > reverse extrusion direction (drag extrusion arrow) > ✔ > click tree node and rename it *Disk* > **File** > **Save As** > *example2.8B* > **Save.**



Step 5: Create *Sketch2* and *Hole* feature: **Extruded Cut** on **Features** tab **> Line** on **Sketch** tab > create vertical construction line passing through origin and click **For Construction > Circle** on **Sketch** tab **>** sketch circle > dimension as shown > exit sketch > **Through All** under **Direction 1** > ✔ > click tree node and rename it *Hole*.



Step 6: Create *CirPattern1*: Select *Hole* feature > **Circular Pattern** from **Linear Pattern** dropdown on **Features** tab > select edge of *Disk* > input pattern data on left pane as follows > for **Parameters**: type 8 for number of instance (third box) > ✔.



Note: The pattern instances are numbered. Expand (then click inside) **Instances to Skip** box shown above > hover over an instance and click it to skip it.

2.20 Selecting, Editing, and Measuring Entities

There are multiple reasons why you would want to select existing entities (e.g., to edit them, use them for new construction, delete them). You may select entities in a sketch or in a feature. The available selection methods are:

- ☐ Click an entity to select it.
- ☐ To select multiple entities at the same time, press the **Ctrl** key on the keyboard followed by clicking the entities.
- ☐ To select many entities, define a select window around them by dragging the mouse from left to right or right to left to enclose the entities.
- ☐ To select entities in a chain (e.g., a rectangle or polygon), right-click any entity in the chain and choose **Select Chain** from the menu that pops up. You must have a chain; otherwise you will not see the **Select Chain** option.
- ☐ To select entities in a loop, follow the same sequence for chains. Loops and chains are similar; we use loops when we select from feature faces, and chains when we select from sketches.
- ☐ You can select any nodes (features, components, planes, drawing views) from the features tree on a part.
- ☐ As you move the mouse over entities to select them, SolidWorks attaches a symbol to the mouse indicating the entity type it senses: line for edge, square for face, and dot for vertex (corner).
- ☐ If multiple entities overlap where the mouse is, you can right-click any entity and choose **Select Other** from the menu that pops up to step through all the entities under the mouse.

Editing an entity includes offsetting, trimming, and transforming. Offsetting an entity means copying it in a new location defined by an offset distance. Click **Offset Entities** on the **Sketch** tab to investigate. Trimming an entity entails shortening or extending it. Mostly we trim entities to shorten them. Click the **Trim Entities** dropdown on the **Sketch** tab > **Trim Entities** and investigate the available different trimming options. One of the very interesting and powerful trimming options is the **Power trim**. If you select this option and move the mouse over an entity, it gets trimmed.

Transforming an entity includes translating (moving), copying, rotating, scaling, mirroring, or stretching it. You can transform sketch entities or features. To transform sketch entities, click the **Move Entities** dropdown on the **Sketch** tab and investigate. A translation (**Move Entities**) requires a translation vector (i.e., a distance and direction). A rotation (**Rotate Entities**) requires a center of rotation and an angle. Copying (**Copy Entities**) requires a copying distance. Scaling (**Scale Entities**) requires a scale factor and a point to scale about. Stretching (**Stretch Entities**) requires a point to stretch with respect to. Mirroring (**Mirror Entities** on the **Sketch** tab) requires a mirror axis. It has its own icon on the **Sketch** tab.

Transforming features occurs in 3D space, unlike transforming sketch entities, which occurs in 2D (the sketch plane). Thus, a translation vector is defined by three components: Δx, Δy, and Δz; or it may be defined by an edge (to define direction) and a distance. A rotation requires a rotation axis and an angle. You can translate, rotate, scale, or mirror features. You can also make a copy of the feature you want to transform and keep the original feature in its place. Click this sequence to access translate/rotate functions: **Insert** (menu) **> Features > Move/Copy.** The pane that opens up to the left of the screen allows you to translate or rotate.

To scale a feature, click **Insert** (menu) **> Features > Scale**. You can perform uniform or nonuniform scaling. Uniform scaling preserves the feature aspect ratio (i.e., it does not distort its look). Uniform scaling uses the same scale factor for the three axes. Nonuniform scaling uses different scale factors. To mirror a feature, click **Mirror** on the **Features** tab. To mirror, you need a mirror face or plane and the features to mirror.

After editing or transforming entities, we may need to measure their geometric properties. We can measure sketch entities or features. We can measure coordinates of a point or length of an entity if we are in a sketch. For a feature, we can measure the coordinates of a vertex, the length of an edge, or the area and the perimeter of a face. Click this sequence to measure: **Tools > Measure** > select any entity to measure. Measurements could be useful during design activities.

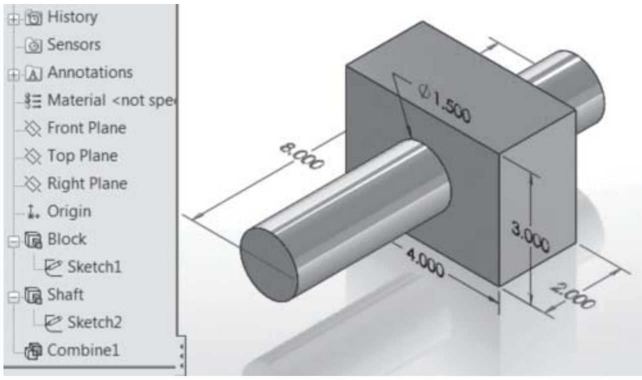
2.21 Boolean Operations

Boolean operations are the mathematical functions that SolidWorks uses to combine features during model creations. CAD/CAM systems utilize set theory, surface-to-surface intersection, curve-to-curve intersection, and curve-to-surface intersection to implement Boolean operations. A Boolean operation combines two features (we call them *bodies*) at a time. The most common Boolean operations are union, intersection, and subtraction. The union of two features is their sum. The intersection of two features is the overlapping volume between them. The subtraction of two features is the opposite of intersection; that is, it is the volume left in a feature after subtracting another feature from it. While the order of selecting the two features is not important for both union and intersection, it is for subtraction. Let us assume we have two features A and B. if you subtract A from B, the result is what is left from B. Conversely, if you subtract B from A, the result is what is left from A. The feature you are subtracting from is sometimes called the *target*. The feature that is used to subtract is called the *tool*, analogous to machining cutting tools.

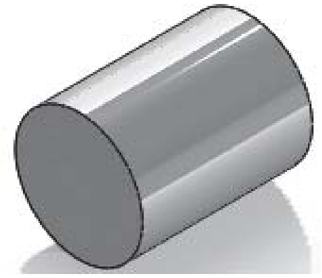
When you create a feature in SolidWorks, the default is to combine it with the previous feature (**Merge Result**), thus resulting in only one feature (body) at any time. If you check off the **Merge Result** box in the left pane of feature creation, you get multiple bodies that you can apply Boolean operations to. To access Boolean operations, click **Insert > Features > Combine**. While the use of Boolean operations is not obvious, there are cases where you may need it. For example, you can use the subtraction operation to create molding, casting, or forming tools.

Example 2.7 Investigate Boolean operations

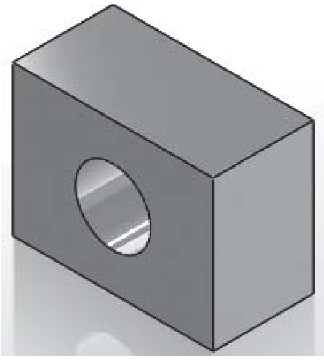
Solution We create a block and a shaft as separate features. We refer to them as B and S respectively. We combine them via Boolean operations. Figure 2.20 shows B and S and the results. All dimensions are in inches.



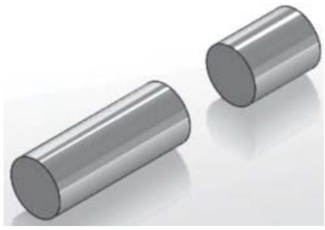
(A) Union (B + S)



(B) Intersect B & S



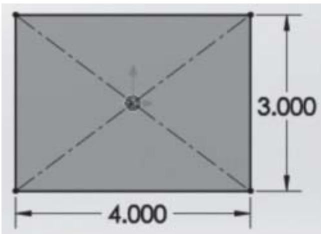
(C) Subtract (B – S)



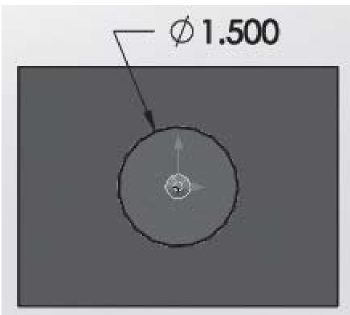
(D) Subtract (S – B)

FIGURE 2.20
Feature patterns

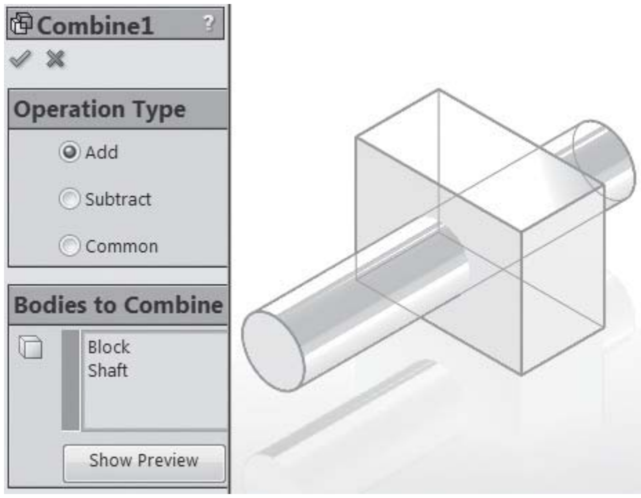
Step 1: Create *Sketch1* and *Block* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and drag to sketch > dimension as shown > exit sketch > enter 2.0 for thickness (**D1**) > reverse extrusion direction (drag extrusion arrow) > ✔ > click tree node and rename it *Block* > **File** > **Save As** > *example2.9* > **Save.**



Step 2: Create *Sketch2* and *Shaft* feature: Click front face of *Block* > **Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and drag to sketch > dimension as shown > exit sketch > select **Mid Plane** > enter 8.0 for thickness (**D1**) > uncheck **Merge Result** checkbox > ✔ > click tree node and rename it *Shaft*.



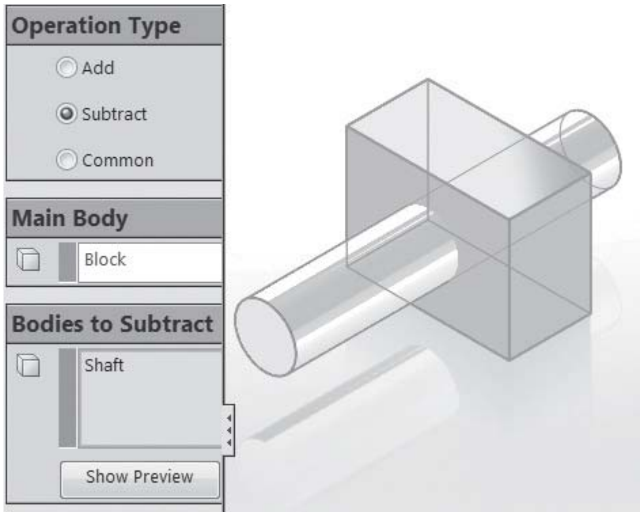
Step 3: Union *Block* and *Shaft* features: **Insert** > **Features** > **Combine** > select **Add** for **Operation Type** > click *Block* and *Shaft* > ✔. Figure 2.20A shows the result.



Step 4: Intersect *Block* and *Shaft* features: **Insert** > **Features > Combine >** select **Common** for **Operation Type >** click **Block** and **Shaft** > ✔. Figure 2.20B shows the result.

Step 5: Subtract *Shaft* from *Block*: **Insert** > **Features > Combine >** select **Subtract** for **Operation Type >** click *Block* for **Main Body** and *Shaft* for **Bodies to Subtract** > ✔. Figure 2.20C shows the result. Reverse the two feature selections to subtract *Block* from *Shaft* (Figure 2.20D).

Note: The two shaft pieces are one "disjoint" body; deleting one piece deletes the other.



2.22 Templates

Templates are also known as *start* or *master parts*. You customize the template to include your preferred settings in tools, options, and document properties. You may also include your favorite base stock (geometry and features), dimensions (preferred units, decimals, tolerances), annotations favorites, commonly used reference geometry, preset drawing views, drawing notes, layers, or materials. You can create different templates: one for parts, one for assemblies, and one for drawings. You create any template as you would create any file and save as template, as you do in Word. For example, to save a part document (file) as a template, click **File > Save As >** select **Part Templates (*.prtdot)** from the **Save as type** dropdown > type file name > **Save**. The document is saved in the SolidWorks **Templates** folder and will become available when you start a new part. SolidWorks templates are saved in this folder: **C:*Program Data\SolidWorks\SolidWorks 2014\templates*.** Note that the ***Program Data*** folder is a hidden file. You need to make hidden files visible on your computer to see it (for Windows 7, open ***Windows Explorer* > *Organize* > *Folder and Search Options* > *View*** tab **>** ***Show hidden files, folders, and drives*** **>** ***OK***). Click this sequence to use the template: **File > New** > **Advanced** > Select the template from the **Templates** tab > **OK**. You can create multiple templates of the same type (part, assembly, drawing) for multiple designs or applications.

2.23 Viewing

You can view a model from different angles during construction. The typical standard views are the front, top, right, and isometric views. Other views exist (e.g., back, bottom, left). Viewing and creating a model are two different activities. Viewing is controlled by the view orientation you select, while construction is controlled by the active sketch plane (i.e., the WCS). While we can view the model from any view during creation, we usually view the model perpendicular to the active sketch plane to get the best view of the results of our creation activities. Figure 1.7 shows the various ways to view geometry on the screen.

2.24 Model Communication

CAD models are communicated and shared by two distinct groups of people: technical and professional. Technical people include CAD designers, design engineers, and manufacturing engineers. Professional people include marketing, sales, and other personnel. Although engineering drawings are the formal method of documenting and communicating a design among technical people, professional people have other communication needs, such as generating screenshots of the model for inclusion in reports, proposals to customers, presentations, and e-mail messages. These needs do not require SolidWorks software, unlike engineering drawings that do require access to SolidWorks software to view them. Also, technical people know how to use SolidWorks, whereas professional people do not.

Recognizing these needs and the inability or disinterest of professional people in learning how to use SolidWorks, SolidWorks offers some effective communication tools. Designers use these tools and generate files for use by professional people. The designers themselves can use these files also to liberate themselves from needing SolidWorks wherever they go. For example, they can use them to communicate with customers (during the design phase), subcontractors, suppliers, and others. To generate screenshots of SolidWorks graphics/geometry pane, click: **View > Screen Capture > Image Capture**. SolidWorks stores the screen capture in Windows

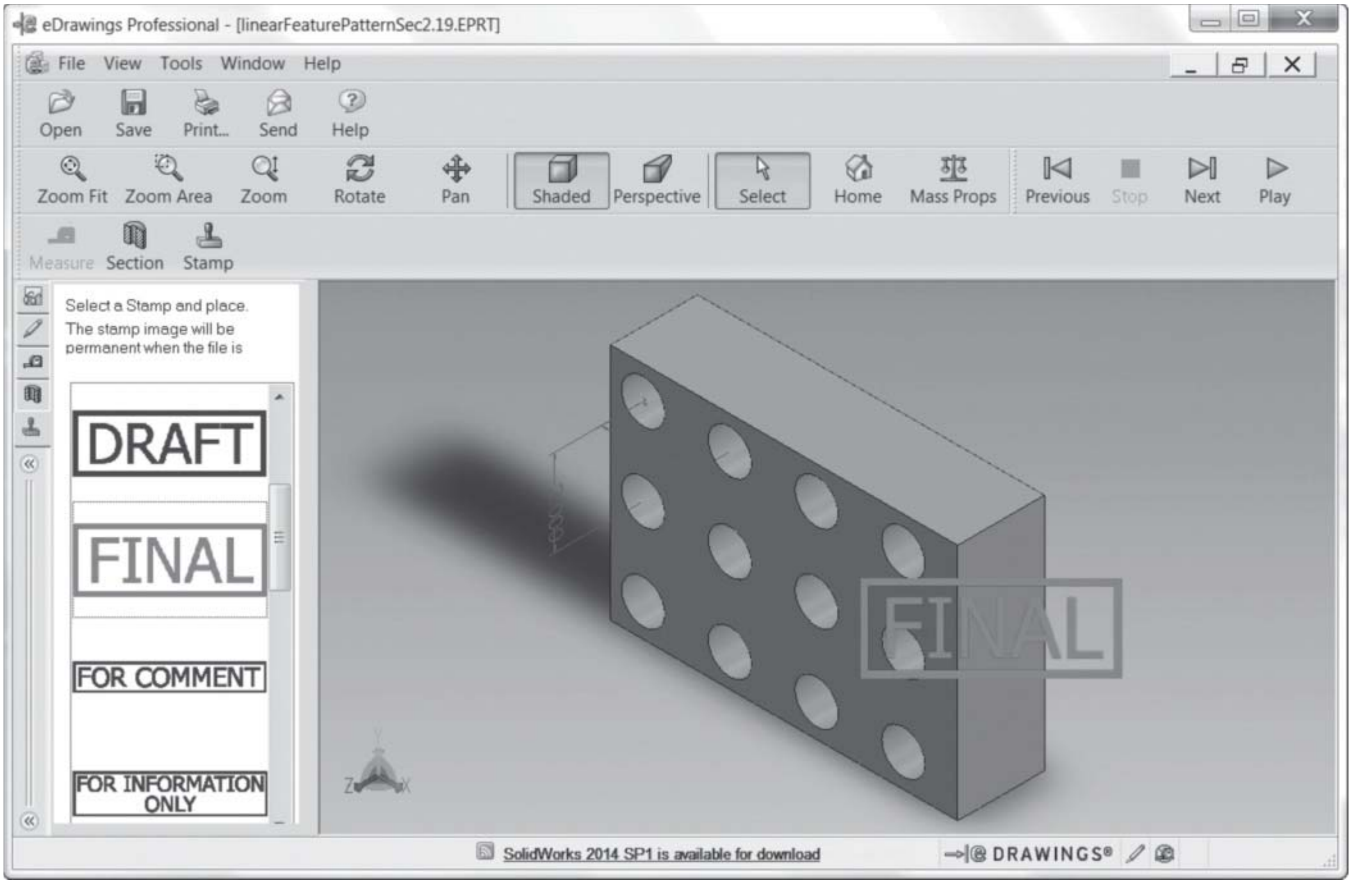


FIGURE 2.21
SolidWorks eDrawings

buffer, waiting for you to use it. Simply open the application such as Word or PowerPoint and paste the capture (use **Ctrl + v** combination on the keyboard). Alternatively, when you save a part, assembly, or a drawing, you can save it as an image. Click: **File** > **Save As** > select **JPEG (*.jpg)** or **Tif (*.tif)** from the **Save as type** dropdown menu.

SolidWorks offers two methods to view SolidWorks files (parts, assemblies, drawings) without needing SolidWorks software. You can download and install the SolidWorks viewer for free. Visit www.solidworks.com/sw/downloads.htm, then click the **FREE CAD TOOLS** tab. Follow the instructions to download and install the viewer version you need. The viewer version must be compatible with the SolidWorks files you intend to open. The viewer, like SolidWorks itself, is backward compatible but not forward compatible (i.e., the 2014 version can open 2013 and older files, but not 2015 files). We encourage you to use and investigate the viewer menus and options. You can use these to transform (translate, rotate, or scale) or add lighting to the model.

The best of all options is the eDrawings tool. eDrawings is a free application from SolidWorks that allows you to view SolidWorks part, drawing, and assembly files. You can save any SolidWorks file as an *eDrawings* file. SolidWorks offers eDrawings software, a very interesting free tool to open and manipulate eDrawings files. After the file is open, you can rotate the CAD model. Click the mouse and rotate the model to view. eDrawings is installed automatically with SolidWorks software. An interesting feature is stamping a model or a drawing. Figure 2.21 shows a stamped model.

2.25 Tutorials

Tutorial 2–1: Create a Coil Spring

Create the coil spring shown in Figure 2.22. This tutorial shows how to create a spring (for learning purposes) although you can download a SolidWorks CAD model as an off-the-shelf component. For example, you can download a spring from McMaster-Carr (e.g., part #9271K182) or use SolidWorks Toolbox to find one. Figure 2.22 shows the coil spring modeling plan. All dimensions are in inches.

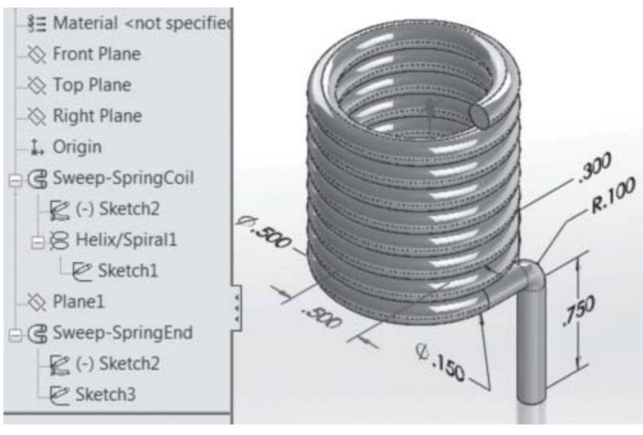
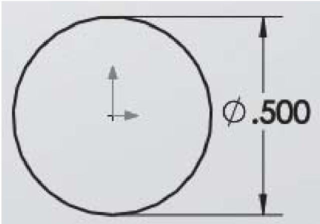
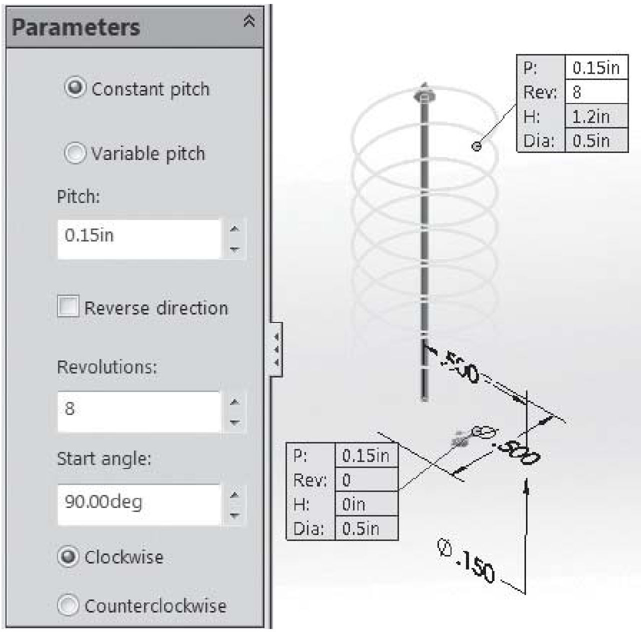


FIGURE 2.22
Coil spring modeling plan

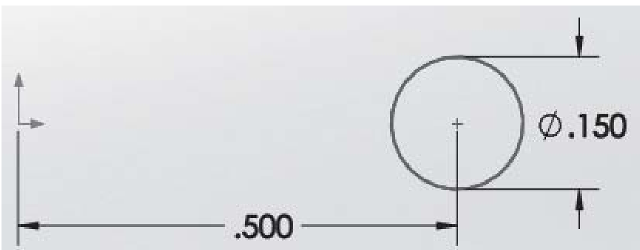
Step 1: Create *Sketch1*: **File > New > Part > OK > Top Plane > Circle** on **Sketch** tab **>** click origin and drag to sketch **>** dimension as shown **>** exit sketch **> File > Save As >** *tutorial2.1* **> Save.**



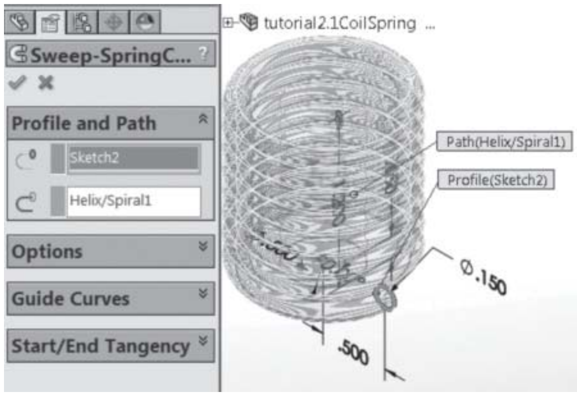
Step 2: Create *Helix/Spiral1*: Select *Sketch1* **> Insert** (menu) **> Curve > Helix/Spiral >** enter helix parameters shown > ✔.



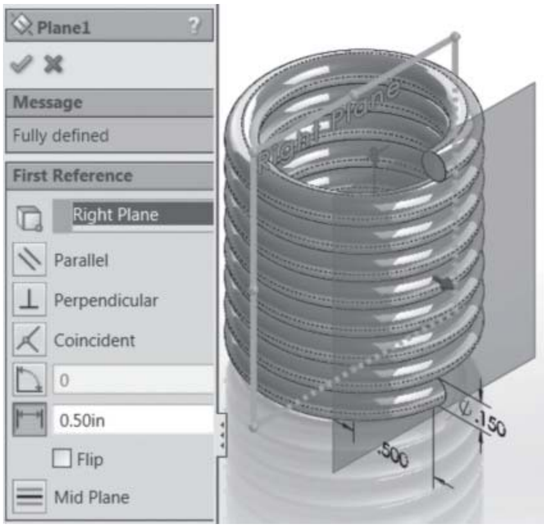
Step 3: Create *Sketch2*: **Front Plane > Circle** on **Sketch** tab **>** click somewhere and drag to sketch **>** dimension as shown **>** exit sketch.



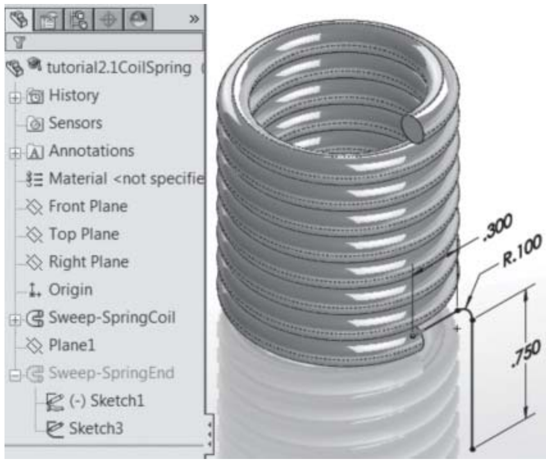
Step 4: Create *Sweep-SpringCoil* feature: **Swept Boss/ Base** on **Features** tab **>** select *sketch2* as the profile and *HelixSpiral1* as the path **>** ✔.



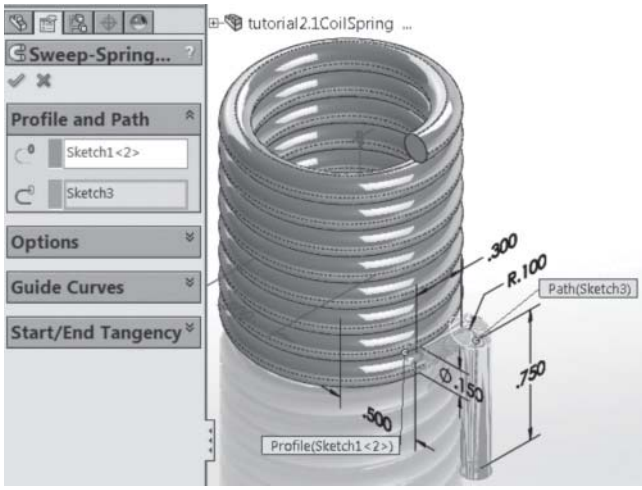
Step 5: Create *Plane1*: **Features** tab **> Reference Geometry > Plane > Right Plane** from **Features** tab **>** enter 0.5 for distance as shown **>** ✔.



Step 6: Create *Sketch3*: *Plane1* **>** hover over context toolbar and select **Sketch > Line** on **Sketch** tab **>** sketch horizontal and vertical lines and dimension as shown **> Fillet** on **Sketch** tab **>** fillet with radius of 1.0 as shown **>** exit sketch.



Step 7: Create *Sweep-SpringEnd* feature: **Swept Boss/Base** on **Features** tab > select *sketch2* as the profile and *Sketch3* as the path > ✔.

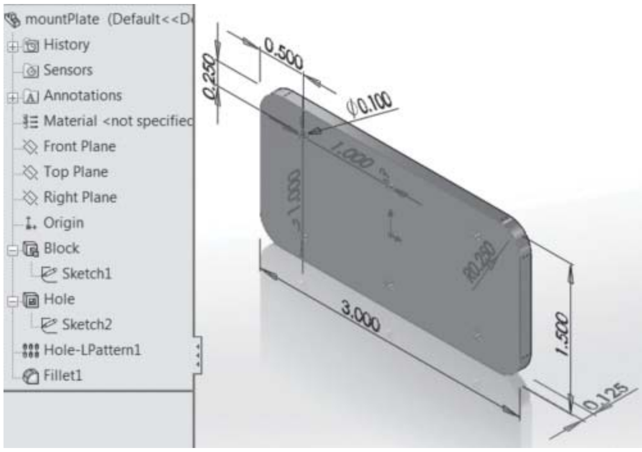


HANDS-ON FOR TUTORIAL 2–1. Create the other end of the spring identically to the one created here.

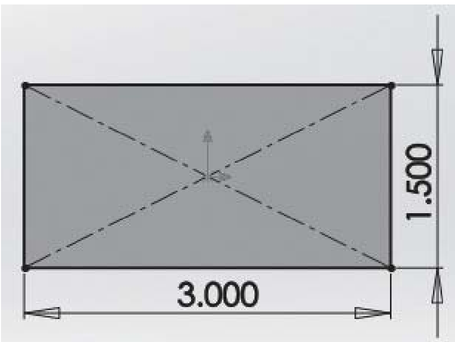
Tutorial 2–2: Create Mount Plate

Figure 2.23 shows the mount plate and its modeling plan. All dimensions are in inches.

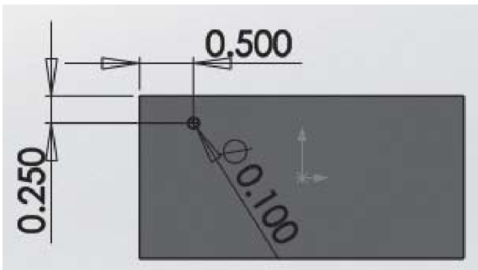
FIGURE 2.23
Mount plate modeling plan



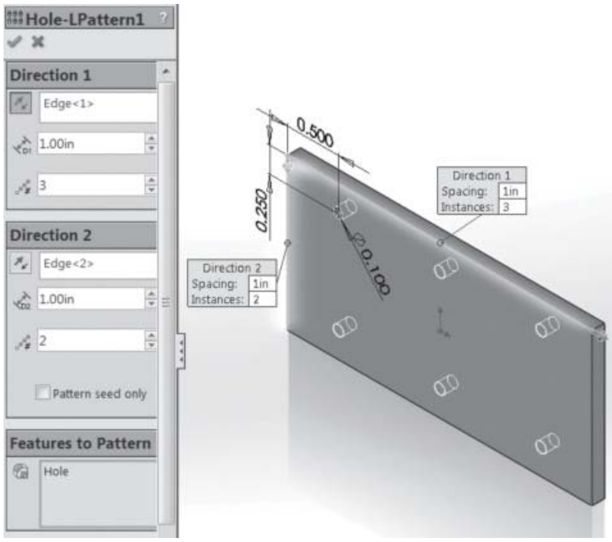
Step 1: Create *Sketch1* and *Block* feature: **File > New > Part > OK > Top Plane > Features** tab **> Extruded Boss/base > Center Rectangle** on **Sketch** tab > click origin and drag to sketch > dimension as shown > exit sketch > enter **0.125** for thickness **(D1)** > reverse extrusion direction **> File > Save As >** *mountPlate* > **Save.**



Step 2: Create *Sketch2* and *Hole* feature: Select *Block* front face as a sketch plane > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab> sketch a circle and dimension as shown > exit sketch > **Through All** from **Direction 1** dropdown > ✔.

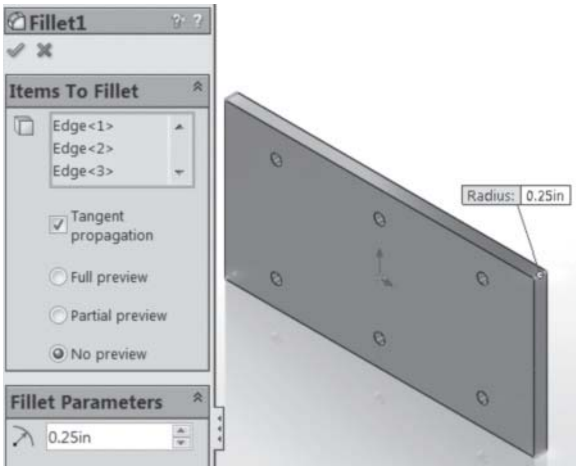


Step 3: Create *Hole-LPattern1* feature: Select *Hole* feature > **Linear Pattern** on **Features** tab > select top edge of *Block* for **X-axis** > input pattern data on left pane as follows > for **Direction 1**: type 1 for **D1** (first box), 3 in second box > select left edge of *Block* for **Y-axis** > input pattern data on left pane as follows > for **Direction 2**: type 1 for **D2** (second box), 3 in third box > ✔.



Note: We use the features approach here to create the six holes in the mount plate, instead of the cross section approach. We create one hole (the top left hole) feature and pattern it. This strategy is good because if you edit the original hole, the pattern holes update automatically.

Step 4: Create *Fillet1* feature: **Features** tab > **Fillet** > select the four edges of *Block* along its depth > enter 0.25 for fillet radius > ✔.



HANDS-ON FOR TUTORIAL 2–2. Create the engineering drawing of the mount plate.

Tutorial 2–3: Create Bracket

Figure 2.24 shows the bracket and its modeling plan. All dimensions are in inches.

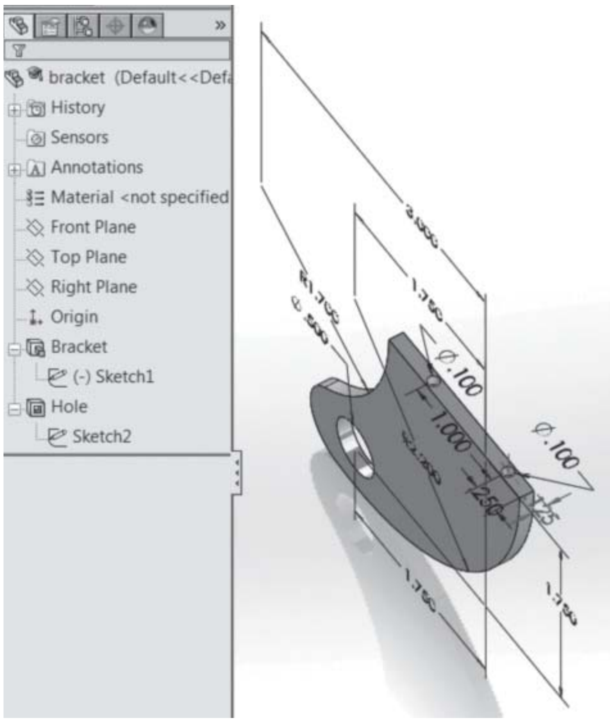
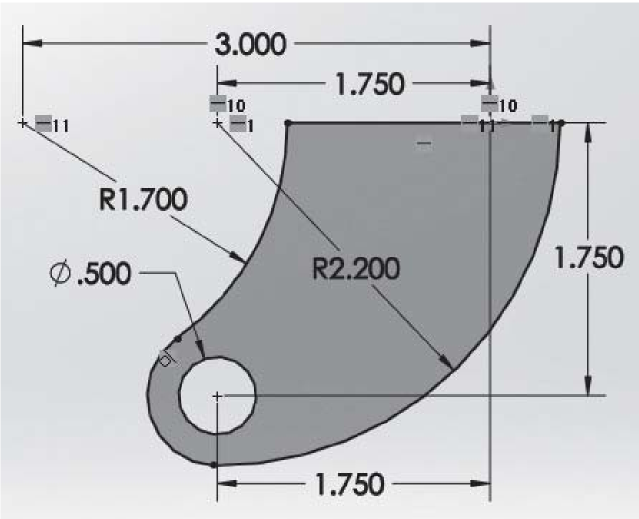
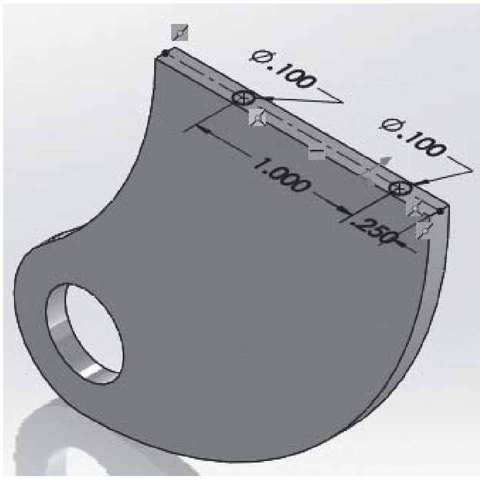


FIGURE 2.24
Bracket modeling plan

Step 1: Create *Sketch1* and *Bracket* feature: **File > New > Part > OK > Front Plane > Features** tab **> Extruded Boss/Base > Centerpoint Arc** on **Sketch** tab **>** click somewhere for arc center, then click twice anywhere to define a clockwise direction (CW) **>** select arc center and origin (use **Ctrl** on keyboard) **> Horizontal >** dimension as shown > repeat to create a second arc > dimension as shown > **Circle** on **Sketch** tab > sketch a circle > dimension as shown > **Line** on **Sketch** tab > sketch a horizontal line passing through the origin as shown > **Centerpoint Arc** on **Sketch** tab > click origin of circle just created, then click twice anywhere to define a CW > **Trim Entities** on **Sketch** tab > with **Power trim** option selected, hover over the line and arcs to trim to their intersections > ✔ > select the two end points of the middle and left arcs (use **Ctrl** on keyboard) **> Merge** > repeat for the other two end points of the middle and right arcs **>** select the middle and left arcs (use **Ctrl** on keyboard) **> Tangent >** exit sketch **>** enter **0.125** for thickness **(D1) >** reverse extrusion direction **> File > Save As >** *bracket* **> Save.**



Step 2: Create *Sketch2* and *Hole* feature: Select top face of *Bracket* feature as sketch plane **> Features** tab **> Extruded Cut > Centerline** on **Sketch** tab **>** connect the two midpoints of the top face right and left edges as shown > **Circle** on **Sketch** tab > create two circles with centers on the centerline just created > dimension as shown > exit sketch > enter 0.1 for thickness (**D1**) > ✔.



HANDS-ON FOR TUTORIAL 2–3. Create the engineering drawing of the bracket.

Tutorial 2–4: Create Wheel

Figure 2.25 shows the wheel and its modeling plan. All dimensions are in inches.

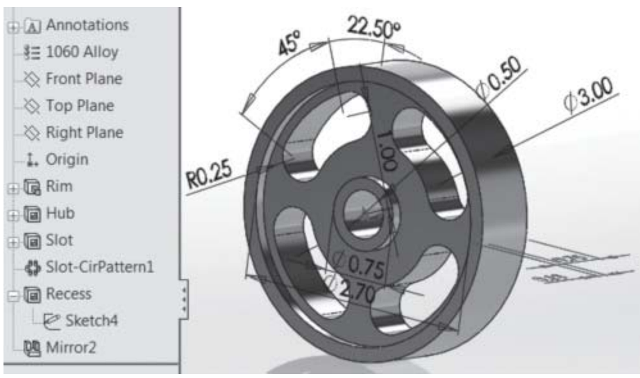
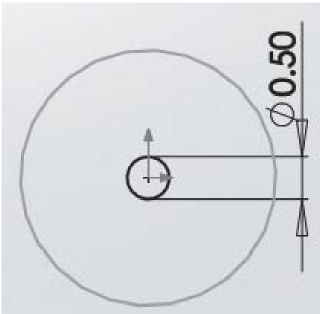


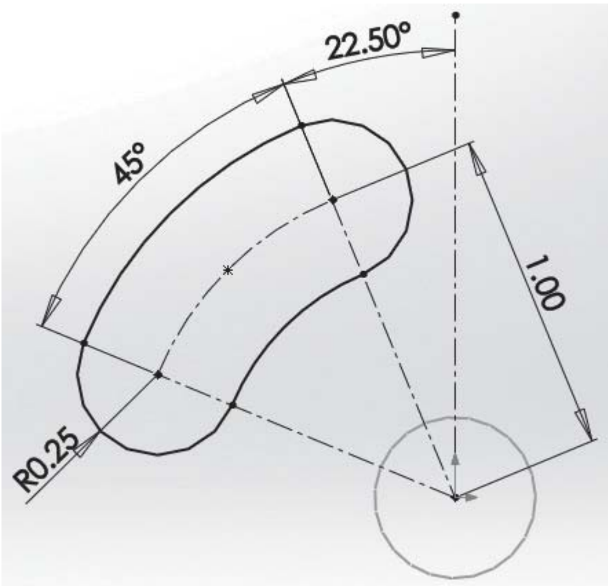
FIGURE 2.25
Wheel modeling plan

Step 1: Create *Sketch1* and *Rim* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Features** tab > **Extruded Boss/Base** > **Circle** on **Sketch** tab > click origin and drag to sketch > dimensions as shown > exit sketch > select **Mid Plane** from **Direction 1** dropdown and enter 0.75 for **D1** > ✔ > **File** > **Save As** > *wheel.*

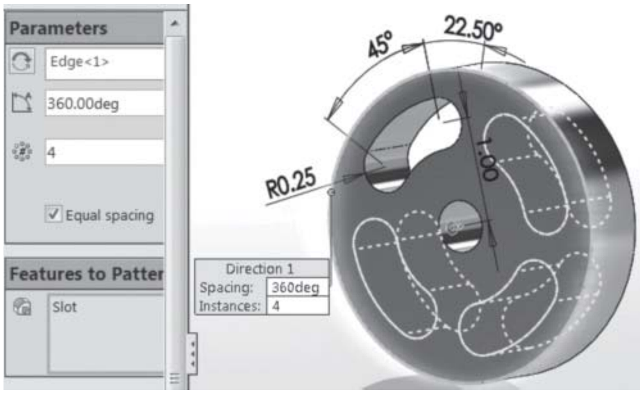
Step 2: Create *Sketch2* and *Hub* feature: Select front face of *Rim* feature as sketch plane > **Features** tab > **Extruded Cut** > **Circle** on **Sketch** tab > click origin and drag to sketch > dimension as shown > exit sketch > **Through All** > ✔.



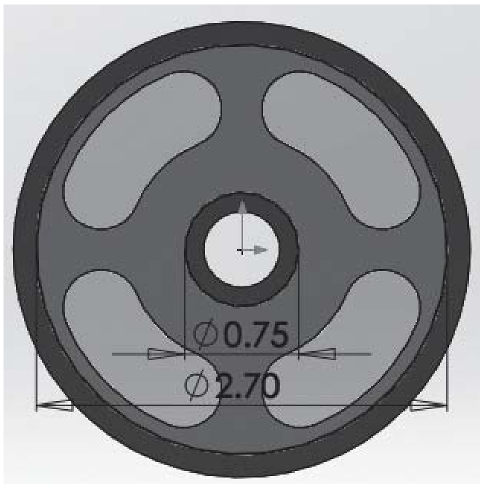
Step 3: Create *Sketch3* and *Slot* feature: Select front face of *Rim* feature > **Features** tab > **Extruded Cut** > **Centerline** on **Sketch** tab > click origin and drag vertically to sketch centerline as shown > **Centerpoint Arc Slot** on **Sketch** tab > click somewhere on screen for arc center, then click twice anywhere to define a CW direction, then drag perpendicular to arc to create slot > dimension as shown > exit sketch > **Through All** > ✔.



Step 4: Create *Slot-CirPattern1* feature: Select *Slot* feature > **Circular Pattern** from **Linear Pattern** dropdown on **Features** tab > select edge of *Rim* feature > input pattern data on left pane as follows > for **Parameters**: type 4 for number of instance (third box) > ✔.

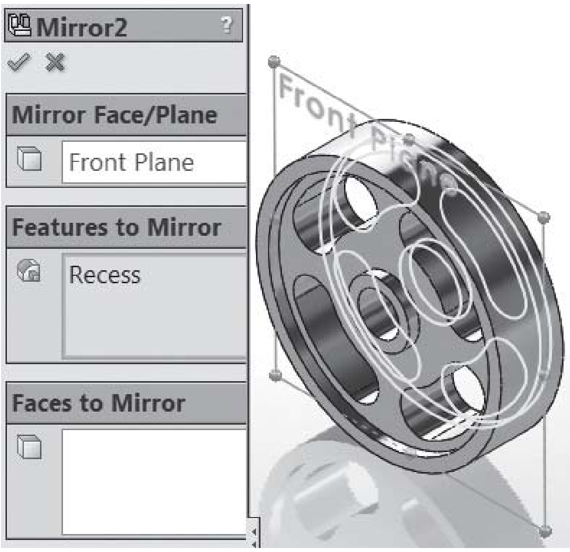


Step 5: Create *Sketch4* and *Recess* feature: Select front face of *Rim* feature > **Features** tab > **Extruded Cut** > **Circle** on **Sketch** tab > click origin and drag to sketch big circle > repeat to create small circle > dimensions as shown > exit sketch > enter **0.125** for thickness **(D1)** > ✔.



Step 6: Create *Mirror2* feature: Select *Recess* feature > **Mirror** on **Features** tab > expand features tree > select **Front Plane** > ✔.

Note: We were able to select **Front Plane** as the mirror plane because we created the *Rim* extrusion using **Mid Plane**. Had not we done that, we would have to create a mirror plane.



HANDS-ON FOR TUTORIAL 2–4. Create the engineering drawing of the wheel.

Tutorial 2–5: Create Tire and Pin

Figures 2.26 and 2.27 show the tire and pin and their modeling plans. All dimensions are in inches.

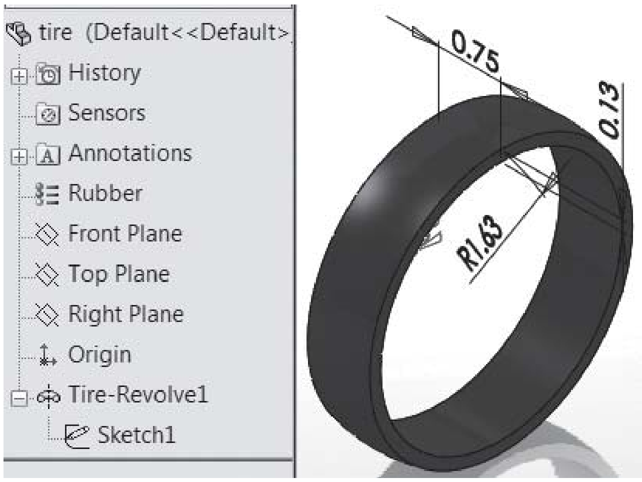


FIGURE 2.26
Tire modeling plan

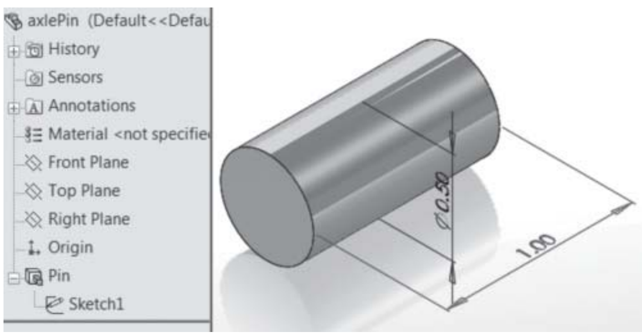
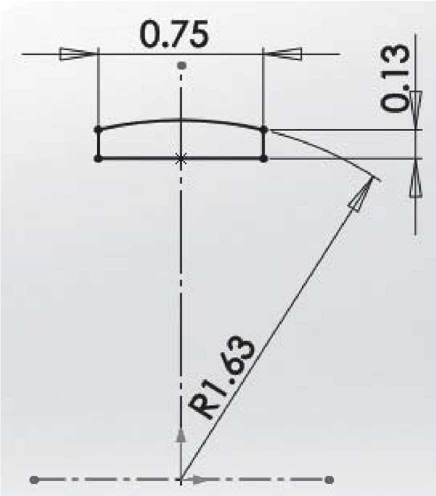


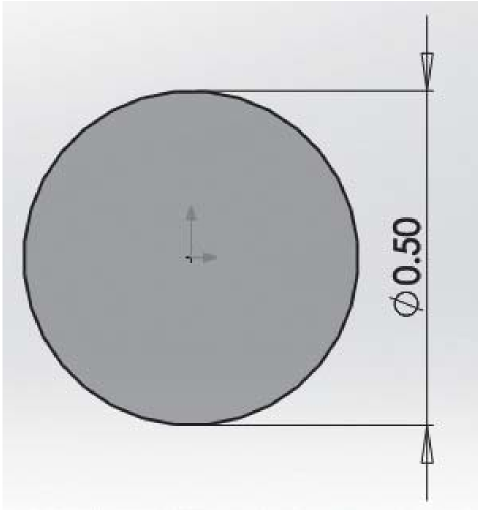
FIGURE 2.27
Axle pin

Creation steps for tire shown in Figure 2.26



Step 1: Create *Sketch1* and *Tire-Revolve1* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Features** tab > **Revolved Boss/Base** > **Centerline** on **Sketch** tab > create a horizontal and vertical line passing through origin as shown > **Centerpoint Arc** on **Sketch** tab > click origin for arc center, then click twice anywhere to define a counterclockwise (CCW) direction > **Line** on **Sketch** tab > sketch a horizontal line > **Point** on **Sketch** tab > insert a point at midpoint of line you just created > use **Ctrl** on keyboard to multi-select the point and vertical centerline > select **Coincident** > **Line** on **Sketch** tab > create two vertical lines passing by the two ends of the horizontal line > **Trim Entities** on **Sketch** tab > **Power trim** all > dimension as shown > exit sketch > select horizontal centerline as **Axis of Revolution** > ✔ > **File** > **Save As** > *tire*.

Creation steps for axle pin shown in Figure 2.27

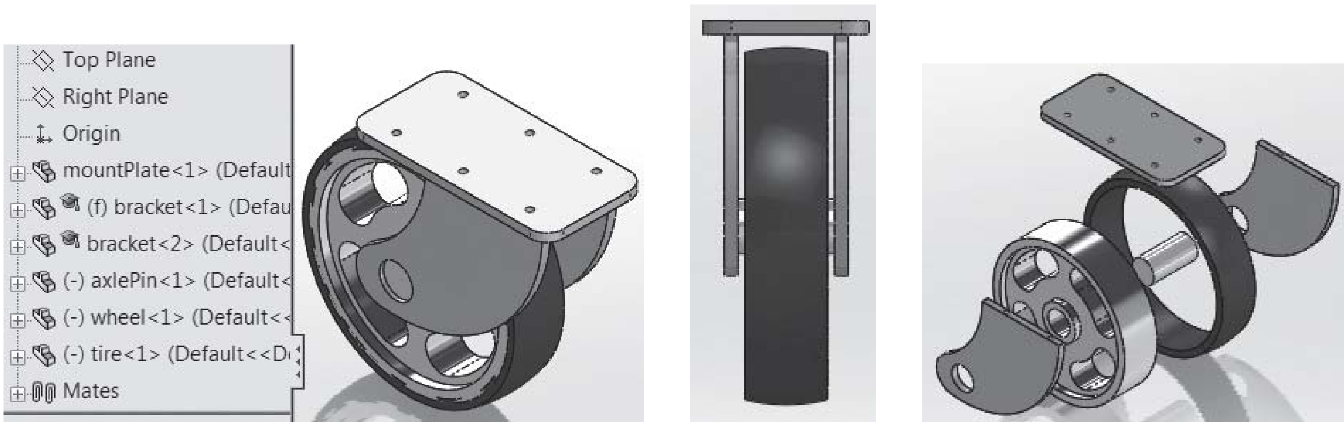


Step 1: Create *Sketch1* and *Pin* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and drag to sketch > dimension circle diameter as shown > exit sketch > enter 1.0 for thickness (**D1**) > select **Mid Plane** from **Direction 1** dropdown > ✔ to finish > **File** > **Save As** > *axlePin* > **Save**.

HANDS-ON FOR TUTORIAL 2–5. Create the engineering drawing of the tire and the axle pin.

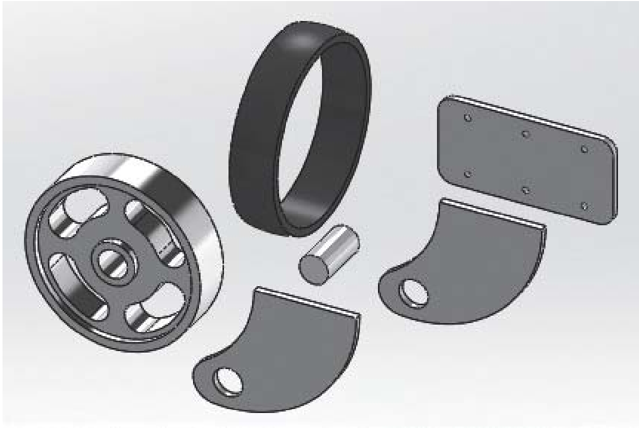
Tutorial 2–6: Create Caster Assembly

Figure 2.28 shows the assembly and its components. We inserted all the components at once into the assembly as shown in Figure 2.28B and assembled each pair. Alternatively, you can insert one component at a time, assemble the first two, insert another component,



(A) Collapsed/exploded views

FIGURE 2.28
Assembly model of the caster

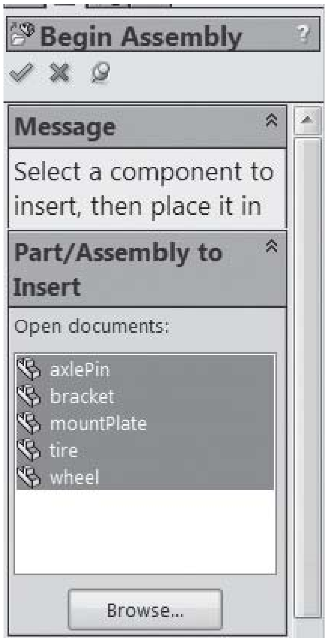


(B) Caster components to be assembled

FIGURE 2.28
(*continued*)

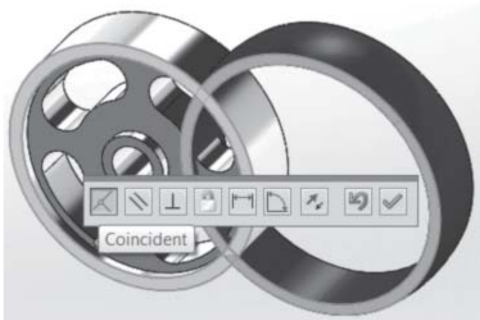
assemble it to the already assembled components, and so forth. This latter approach works better for large assemblies that may have many components. Also, keep in mind that the first component you insert into the assembly becomes fixed in the assembly modeling space (graphics window). An (**f**) shows next to its node in the assembly tree. You can always float the component if needed: right click it > **Float** from the popup that opens; or, you can always fix a component: right click it > **Fix** from the popup that opens.

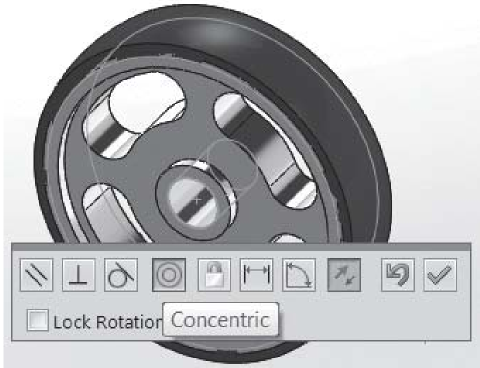
Step 1: Create assembly model and save assembly: **File > New > Assembly > OK** > **Browse** button > use **Ctrl** on keyboard to multi-select the caster components at once > **Open** > click the push pin shown > select *bracket* and click twice to place two instances > select *mountPlate* and click once in graphics window > select each of the remaining *axlePin*, *tire*, and *wheel* and click once to place an instance of each > place instances as shown in Figure 2.28B. Drag any component to adjust its placement as needed.



Step 2: Mate *tire* and *wheel*:

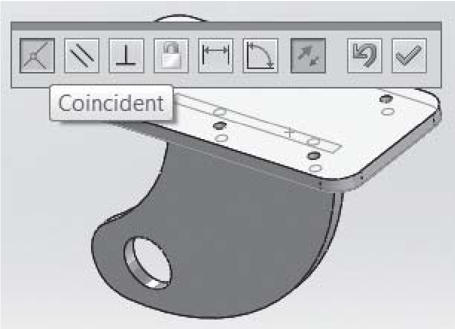
Mate on **Assembly** tab > **Coincident** > select two shown side faces > click mouse right button (✔) > **Concentric** > select the two inside cylindrical faces > ✔ > **File > Save As** > *caster* > **Save.**

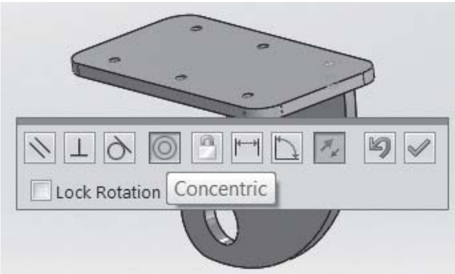




Step 3: Mate *bracket* instances and *mountPlate*: We need two coincident and four concentric mates in total.

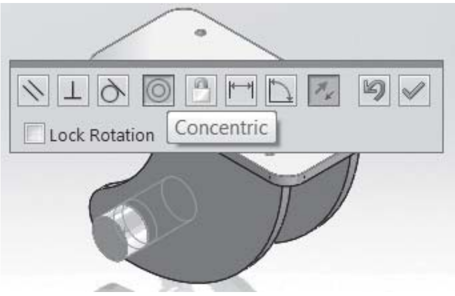
Mate on **Assembly** tab > **Coincident** > select large face of *mountPlate* and small top face of *bracket* > ✔ > **Concentric** > select cylindrical faces of two corresponding holes on *mountPlate* and *bracket* as shown > ✔ > repeat the concentric mate for two other corresponding holes; otherwise, the two components can rotate relative to each other > repeat to mate other *bracket* instance and *mountPlate*.



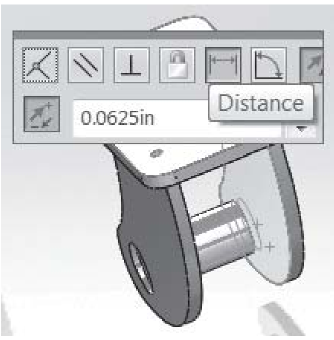


Note: If it is too hard to select hole faces, right click near any face > **Select Other** from popup that opens > click each choice until you find the desired face.

Step 4: Mate *axlePin* and *bracket* instances: We need two concentric and one distance mates.



Mate on **Assembly** tab > **Concentric** > select cylindrical faces of *axlePin* and one *bracket* lower hole > ✔ > repeat the concentric mate for *axlePin* and the lower hole of other *bracket* instance > **Distance** mate > select right side face of *axlePin* > select inside face of right *bracket* instance as shown > enter 0.0625 for distance > ✔.

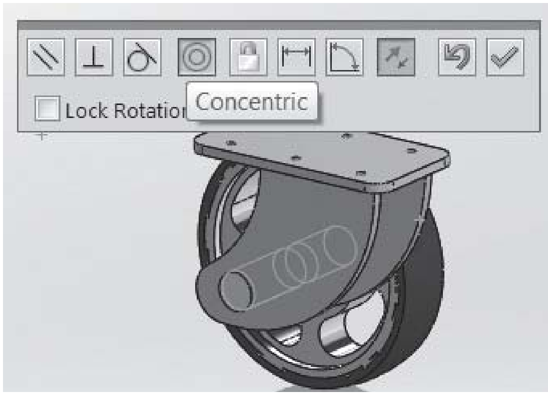


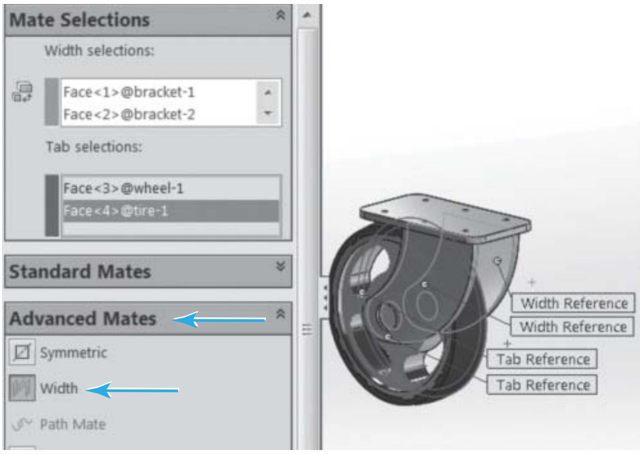
Note: Make sure that the pin right side face is inside the bracket hole for system to get your intent; otherwise, it uses the distance as a gap between the pin right side face and the inside face.

Step 5: Mate *tire/wheel* and *axlePin* instances: We need one concentric and one width mate.

Mate on **Assembly** tab > **Concentric** > select cylindrical faces of *wheel* hub and *axlePin* > ✔ > expand **Advanced Mates** section > **Width** > select the inside faces of *bracket* two instances > select the two side faces of *wheel* > ✔

Note: We need the **Width** mate to center *wheel* between the two *bracket* instances.

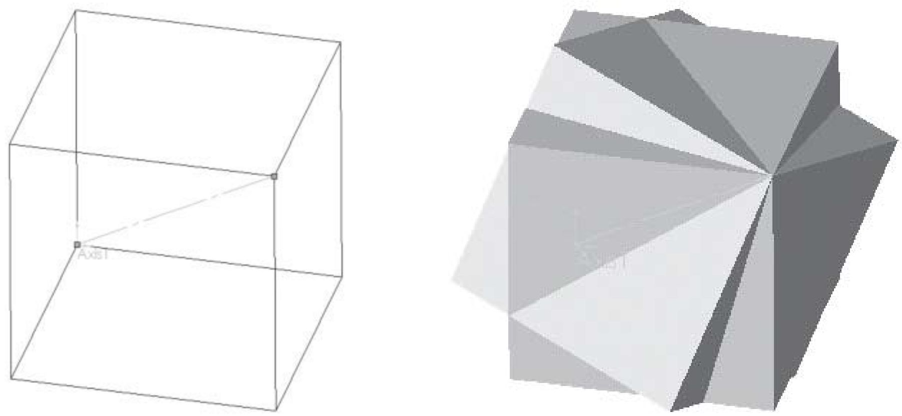




HANDS-ON FOR TUTORIAL 2–6. Create the assembly drawing, the exploded view, and the animation file of the caster assembly.

problems

1. How many faces, edges, vertices, loops, and genus does each part in Figure 2.1 have? Verify the Euler equation for each part.
2. Create the part shown in Figure 2.1B. Assume all dimensions. Create two equations so that the length of the front face of the extrusion is equal to twice the depth of the extrusion and the height of the front face is equal to 1.25 the depth of the extrusion.
3. Create a $60 \times 60 \times 60$ mm cube. Create one of its diagonals as **Axis** from the **Reference Geometry** menu. Figure 2.29 shows the cube and the axis. Rotate a copy of the cube 30 degrees around the axis to get the result shown.

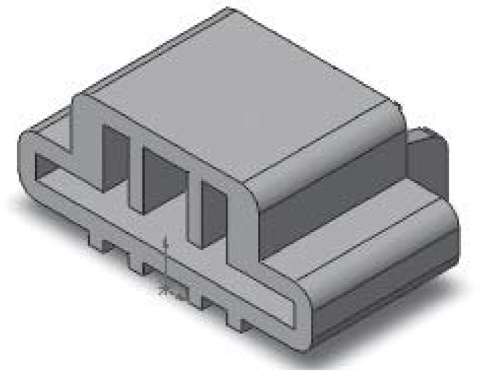


(A) Original cube (B) Rotated cube

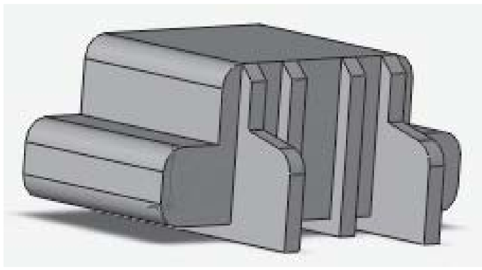
FIGURE 2.29
Model rotation

4. You need to verify and get the geometric information of the vertices, edges, and faces of the cube shown in Figure 2.29A. Use SolidWorks to verify ONLY the top face of the cube, two of its edges, and its four vertices. Create the front face of the cube in the **Front** sketch plane such that the WCS origin is at the center of the front face (use a **Center Rectangle** on the **Sketch** tab). Extrude the sketch to the back, away from the viewing eye. Verify SolidWorks results manually. Do they match your hand calculations? *Hint:* Use **Tools > Measure**. When you select an entity, SolidWorks displays its geometric information.
5. Translate a copy of the cube shown in Figure 2.29A along the following vector: $d_x = -50$, $d_y = 50$, and $d_z = 100$.
6. Scale the cube shown in Figure 2.29A uniformly by a scale of 2, and nonuniformly by scales of 1, 2, and 3 in the X, Y, and Z directions respectively.

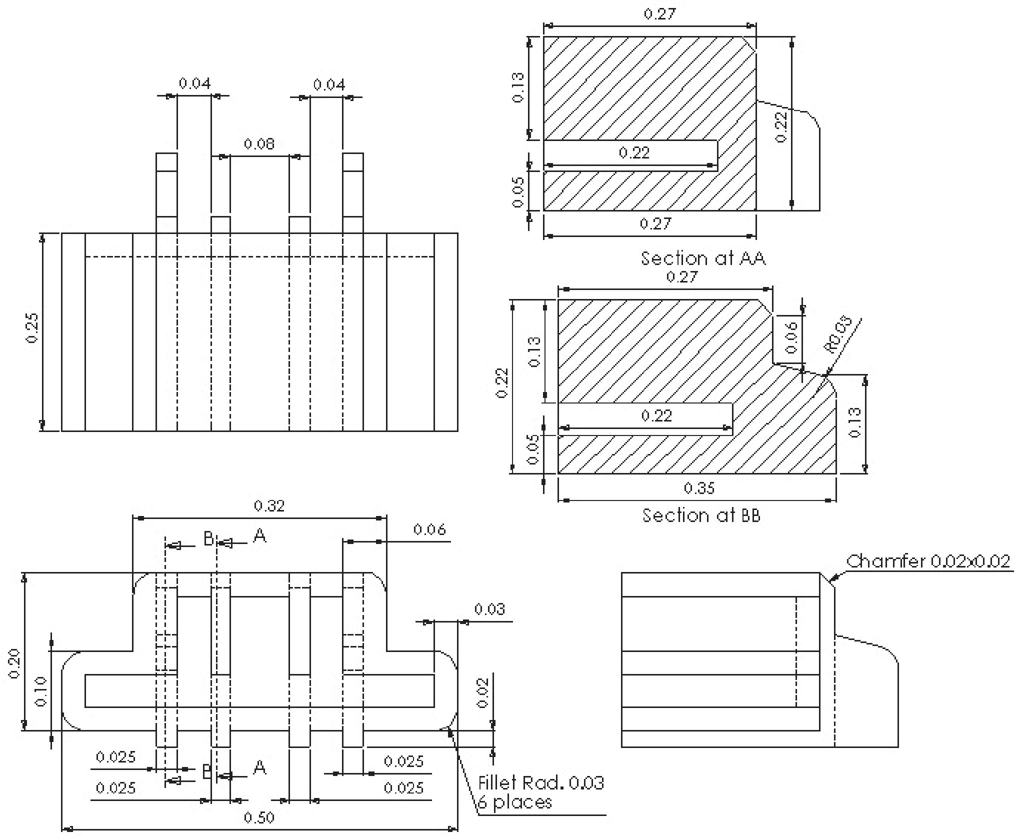
7. Create the AMP connector model shown in Figure 2.30. Also, create the drawing of the part. All dimensions are in inches.



(A) Front view



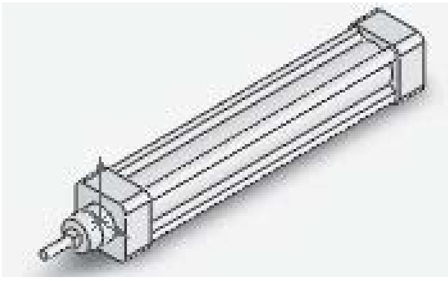
(B) Rear view



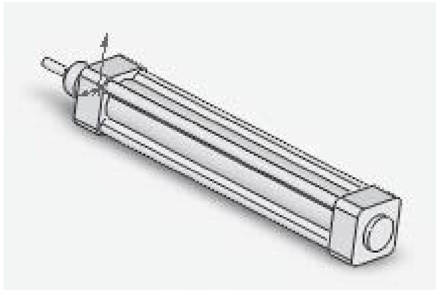
(C) Dimensioned views

FIGURE 2.30
AMP connector

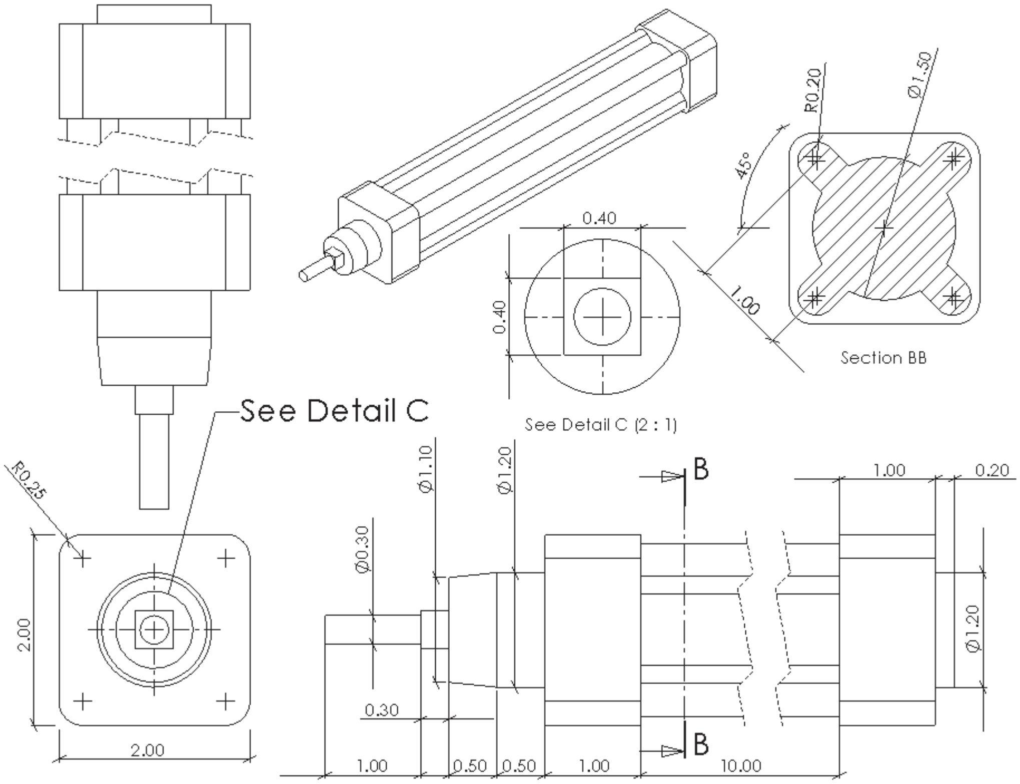
8. Create the air cylinder model shown in Figure 2.31. Also, create the drawing of the part. All dimensions are in inches.



(A) Front view



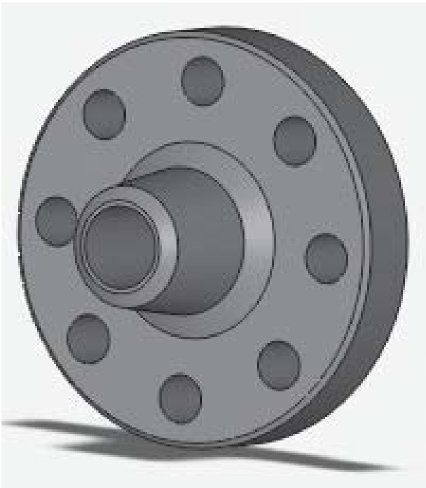
(B) Rear view



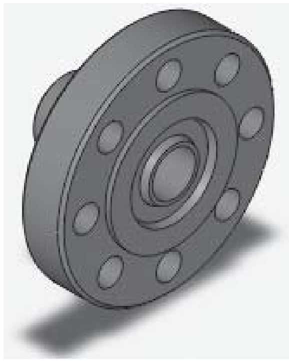
(C) Dimensioned views

FIGURE 2.31
Air cylinder

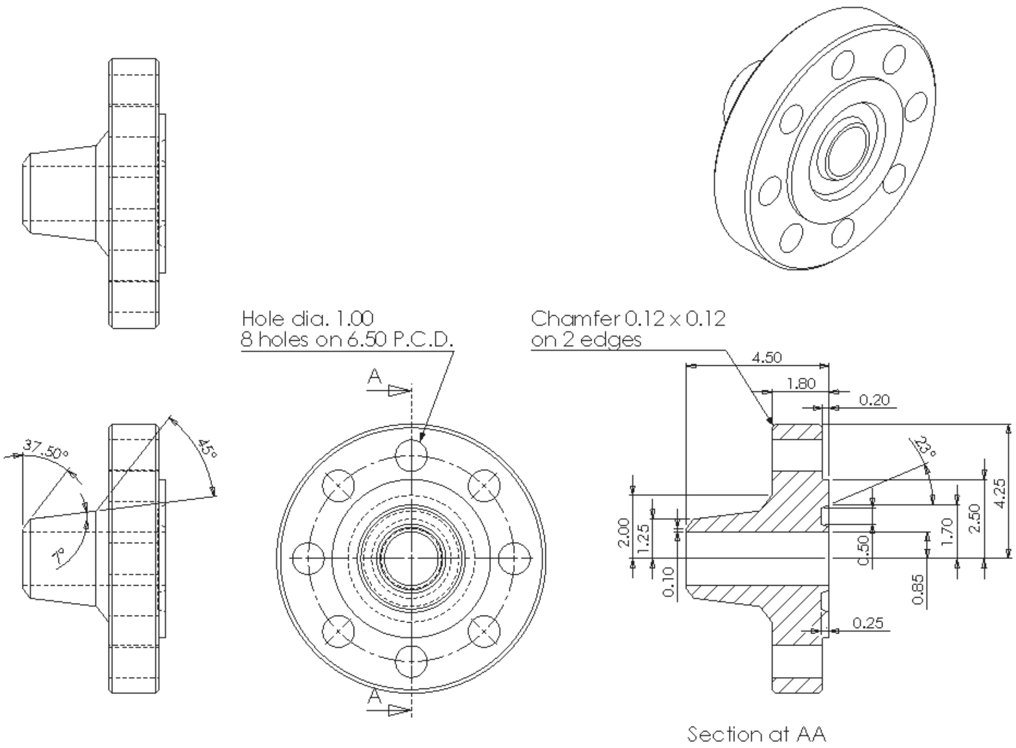
9. Create the flange model shown in Figure 2.32. Also, create the drawing of the part. All dimensions are in inches.
10. Modify the flange model shown in Figure 2.32 as follows:
    a. Change the circular flange to a square flange. The length of the square side is equal to the diameter of the flange outside circle, i.e. 8.5 inch as shown in Section AA in Figure 2.32.
    b. Change the hub to be cylindrical instead of conical. Use the larger diameter of the current conical hub as the diameter for the new cylindrical hub.
    c. Use 8 holes as the current hub, spaced equally along the edges of the square. Create the drawing of the new square flange.



(A) Front view



(B) Rear view



(C) Dimensioned views

FIGURE 2.32
Flange

11. Create the laser mount model shown in Figure 2.33. Also, create the drawing of the part. All dimensions are in inches.

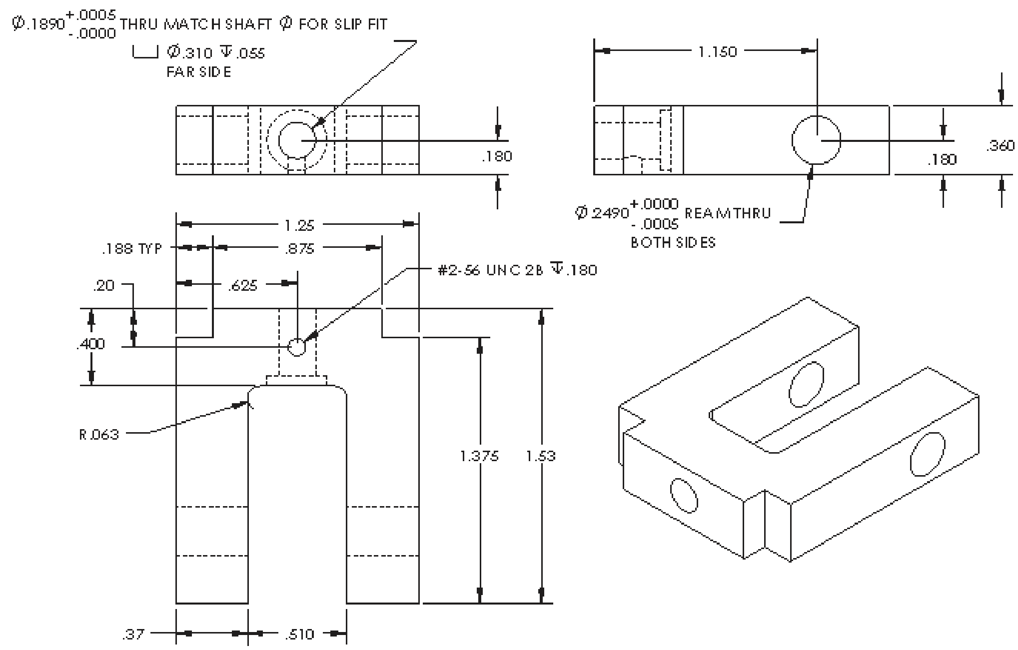


FIGURE 2.33
Laser mount
(Courtesy of VIC)

12. Create the switch activator lever model shown in Figure 2.34. Also, create the drawing of the part. All dimensions are in inches.

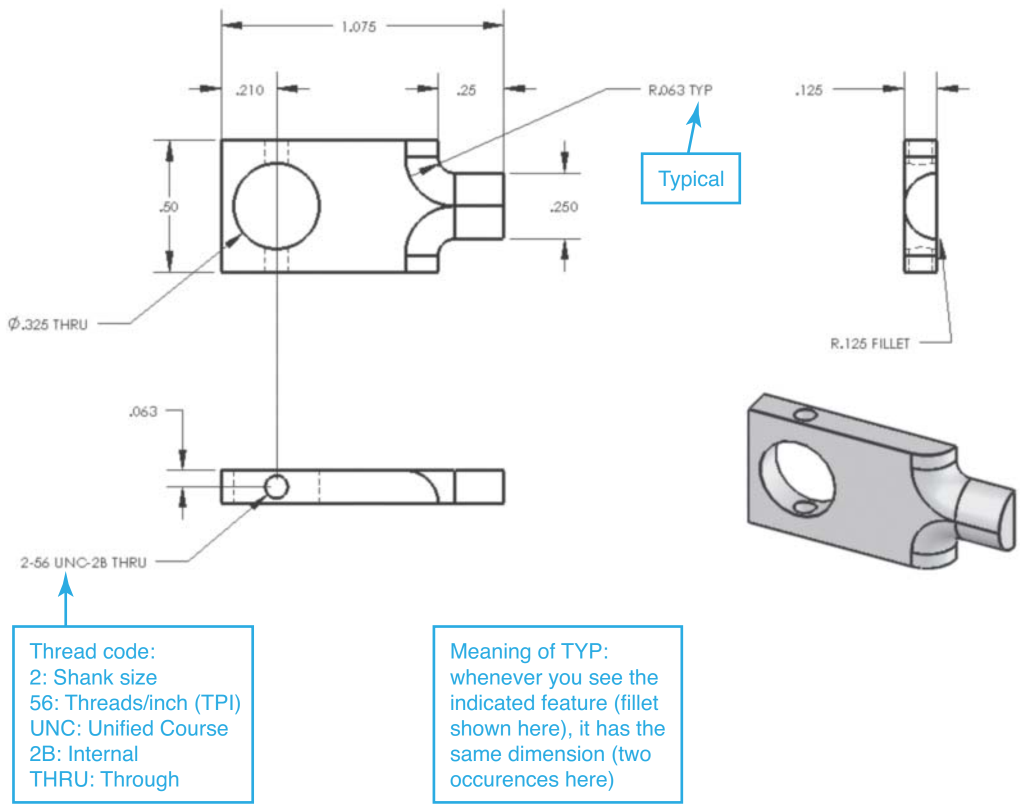
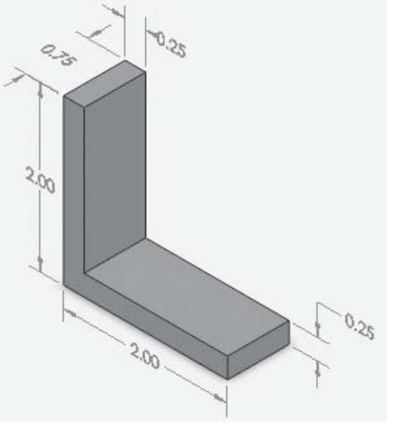
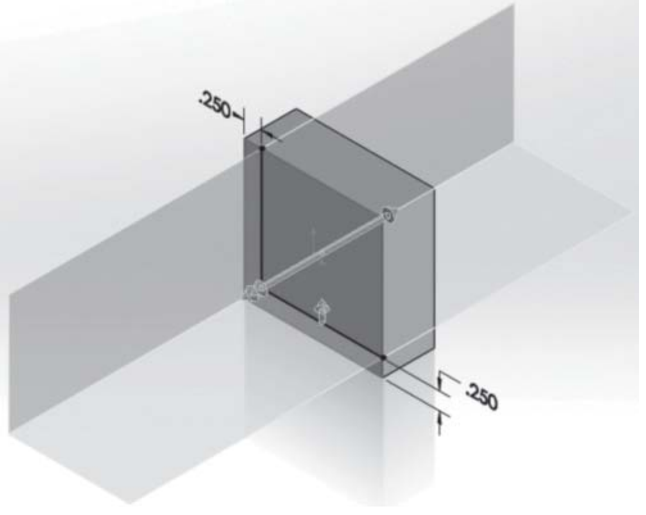


FIGURE 2.34
Switch activator lever
(Courtesy of VIC)

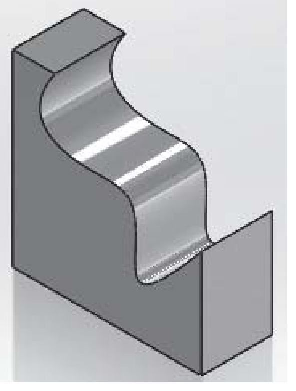
13. Create the L bracket shown in Figure 2.35. Also, create the drawing of the part. Create the bracket using the cross section and features approaches. Which approach do you think is faster to create and why? Explain your answer. ***Note:*** To create the extruded cut, select the front face of the block, sketch the two lines that form the L as shown in Figure 2.35B, and exit the sketch. There is no need to create a closed rectangle in the sketch. SolidWorks understands the intent, closes the sketch with the face edges, and creates the cut. Using this idea, you can create extrude cuts with elaborate profiles in the block as shown in Figure 2.35C.



(A) L bracket



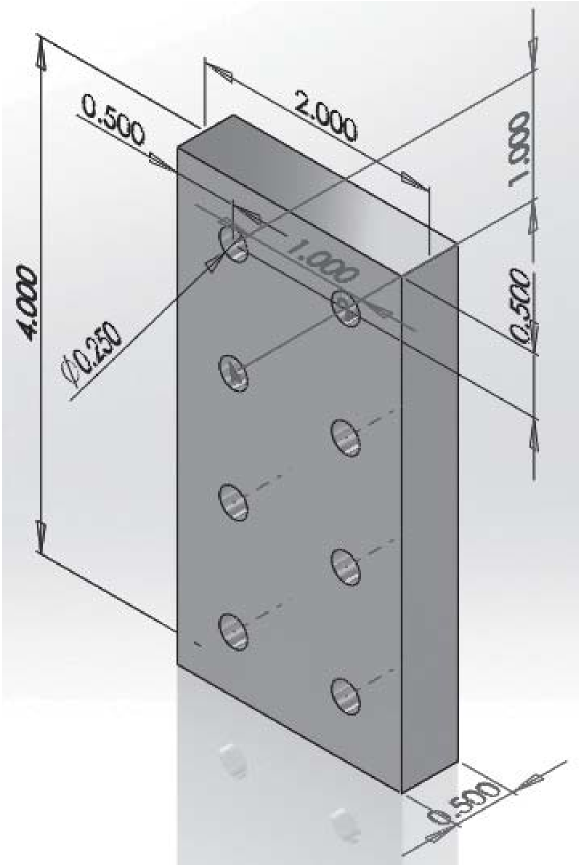
(B) Two lines define extruded cut sketch



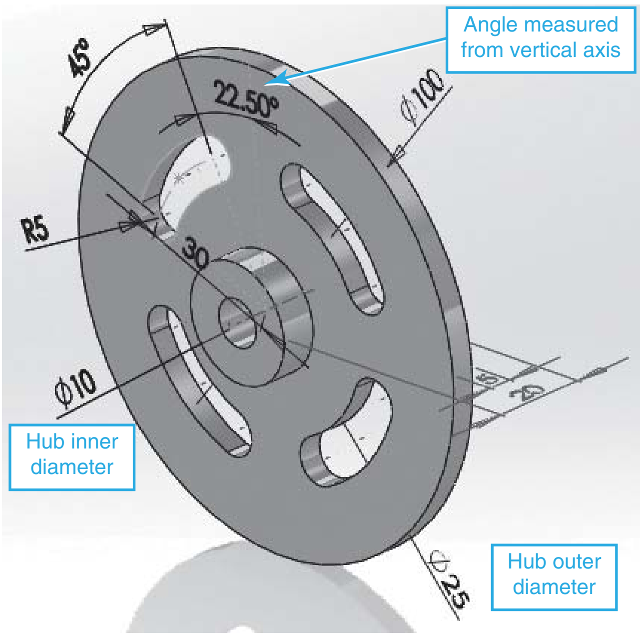
(C) Elaborate cut

FIGURE 2.35
L bracket (all dims in inches)

14. Figure 2.36A shows a part with a linear pattern and Figure 2.36B shows a part with a circular pattern. Create both parts.



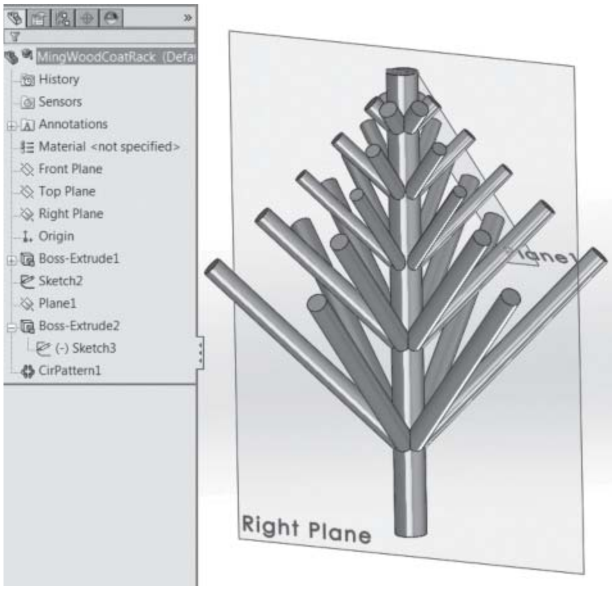
(A) Block with linear pattern (all dimensions in inches)



(B) Wheel with circular pattern (all dimensions in mm)

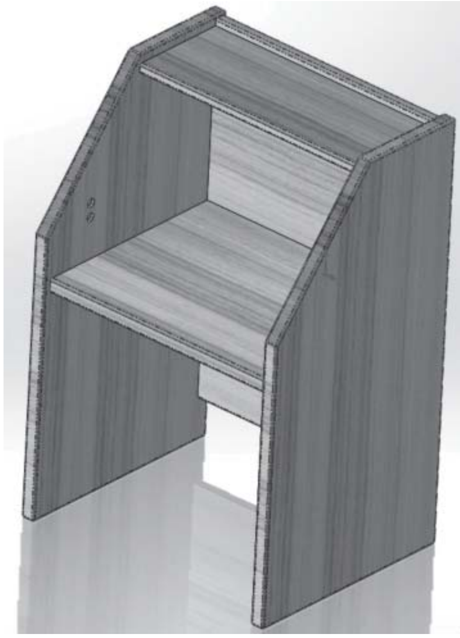
FIGURE 2.36
Parts with patterns

15. Figure 2.37 shows some useful CAD parts. Each part model is shown with its features tree. Each part represents an object we use in our daily lives. The parts are shown without dimensions. These parts challenge your creativity and abilities. Create the CAD model of each part. If you prefer, find equivalent and similar parts around you and model them.



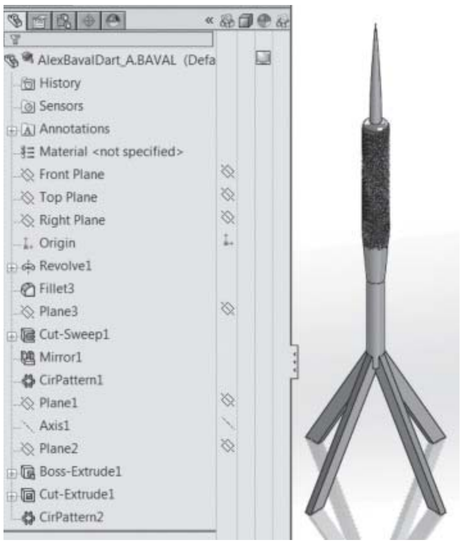
(A) Coat rack

Modeling highlights: Create **Plane 1** at 45 degrees > create five circles on **Plane 1** > **Extruded Boss/Base** > circular pattern the five extrusions.



(B) Library desk

Modeling highlights: Create different extrudes. Note, the two holes on the left side of the desk are the electric outlets to charge your laptop while studying in the library.



(C) A dart

Modeling highlights: *CirPattern1* shown in the features tree creates the knurling effect shown below the needle at the top. To create it, you create a cut in the revolve part > circular pattern it 30 times along 360 degrees.

(D) A cheese wedge

Modeling highlights: Start with extruding a top sketch (wedge shape) > fillet the top and bottom edges as shown > create different extruded cuts. To create the FINLAND text, click this sequence: **Front Plane** > **Text** (shown as letter **A**) on **Sketch** tab > type FINLAND > exit sketch > **Wrap** on **Features** tab > select the face to wrap text on > type 0.05 for **D1** > ✔.

FIGURE 2.37
CAD Challenges

CHAPTER

3

Design Intent

3.1 Introduction

Design intent is important in CAD design. It influences the way we think about creating CAD models. Generally, we define **design intent** as the rationale behind the decision-making process during design. There is a difference between design intent and design functionality. Intent justifies the design whereas functionality describes what the design does. The effect of design intent is observed in design and manufacturing tasks, for example, editing/modifying the design or manufacturing the part. Design intent is not a well-defined topic; it is open for different interpretations and may take various forms. Some view design intent as equations, geometric relations, mating conditions in assemblies, and/or the sequence of creating a part. Others believe design intent is to create the CAD part in the same way it will be manufactured (machined). Some even talk about manufacturing intent.

Although it is universally acknowledged that knowing and using design intent are important, there is a lack of both support for framework for design intent and widespread use in engineering. Yet, interest in design intent systems has grown. We define a **design intent system** as a tool for capturing design intent and making it easily accessible. These systems are important tools because they include not only the reasons behind a design decision but also the justification for it, the other alternatives considered, the trade-offs evaluated, and the arguments that led to the decision. Design intent systems improve dependency management, collaboration, reuse, maintenance, learning, and documentation.

Because the design and manufacturing processes evolve around the geometric shape of the product, the current CAD/CAM systems are based on geometric modeling techniques. Although these techniques are powerful modeling tools, they are deficient in recording the embodiment details of the product such as design intent, manufacturing specifications, and other constraints. CAD/CAM systems cannot easily answer design-related questions such as “How is it supposed to work?” “Why is it done this way?” “What alternatives were considered?” “What can be changed?” and “What will be affected?”

3.2 Capturing Design Intent

In terms of geometric modeling, the part modeling plan is the best expression of design intent. We also follow the notion of designing the part "as you would manufacture it." Tutorial 3–1 shows an example. We can capture design intent in various ways during geometric modeling:

1. **Modeling plan:** A plan involves careful selection of a modeling sequence and evaluating how it affects future editing, modifications, and changes. In general, modeling plans depend on personal preferences and experiences, level of design knowledge, level of manufacturing knowledge, and heuristics and rules among designers in a company.
2. **Sketch relations:** Specify relations among sketch entities to capture the design intent among them. For example, two entities may be parallel or perpendicular to each other.
3. **Equations:** Define equations to capture design intent.
4. **Assembly mating conditions:** These conditions specify how components of an assembly are connected to each other.

Keep in mind that with good design intent, CAD models can be updated almost effortlessly. Changes made to one aspect of a model propagate appropriately through the model, assembly, and drawing. With poor design intent, features may update inappropriately or fail. When selecting references, creation sequence, or other modeling activities, always ask yourself how the model might change. Should a hole stay centered on the part? Should it remain a fixed distance from some face or edge? Should it move with an associated feature, such as a boss? Considering the full range of possible changes, will a drastic change in a dimension make a sketch impossible to resolve, for example, turning it inside out?

The CAD designer should carefully consider the consequences of his or her design methodology. The following is a list of typical questions that the designer may address to facilitate capturing the design intent without violating the design concepts:

- ☐ Which feature to create first, that is, what is the base feature
- ☐ In what order other features should follow (holes, cutouts, chamfer, fillets)
- ☐ What tool to use to create a feature (some geometries such as holes can be created using many different tools/commands)
- ☐ What sketch plane to use (a default sketch plane or a feature face)
- ☐ How to dimension the feature sketch (should an arc use a radius or diameter?)
- ☐ What relations to use to lock the sketch shape (horizontal, vertical, parallel, etc.)
- ☐ What equations to use that ensure valid sketch updates (length = 2*width)
- ☐ How the feature will be dimensioned in the drawing
- ☐ Can the components be assembled in reality?
- ☐ What is the ease of assembly?
- ☐ What assembly constraints to use (screw in a hole)
- ☐ Do these constraints reflect/mimic reality?
- ☐ What entities to select to apply assembly constraints (shaft in bearing using inner face)
- ☐ Can the feature be machined/produced? If yes, is it cost effective?

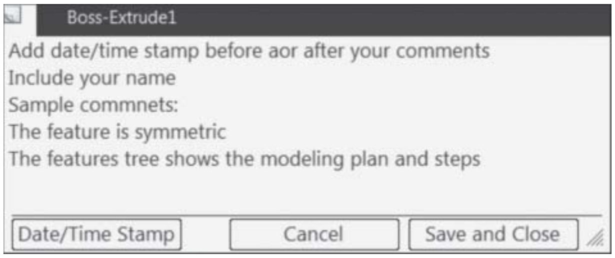
How much planning and consideration should a designer give to design intent? The answer is difficult. We are certain that paying too much attention to design intent prohibits creativity and innovation in design, and slows finishing modeling tasks considerably. Our advice is that design intent and intuition come with experience. Sometimes it is hard to foresee all possible cases down the design road. Thus, start with the obvious design intent in your part, and discover other design intent during modeling.

3.3 Documenting Design Intent

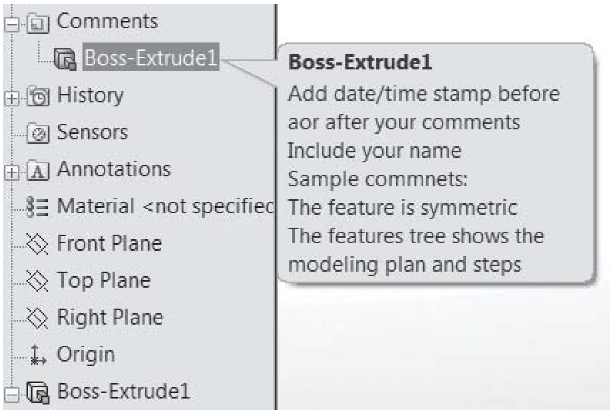
Documenting and capturing design intent make your design more reusable and easier to modify in the future, by both you and other designers. Documenting design intent facilitates communicating and enforces any agreed-upon design principles and rules. The key to effective use of design intent is to be sure all members of the design team consistently document it all the time. The design intent can be shared via HTML or Word documents, as shown in this chapter. SolidWorks provides multiple methods to document design intent: comments, design binder, equations, design tables, configurations, dimension names, feature names, and organizing the features tree into folders. These methods are discussed in the sections to follow.

3.4 Comments

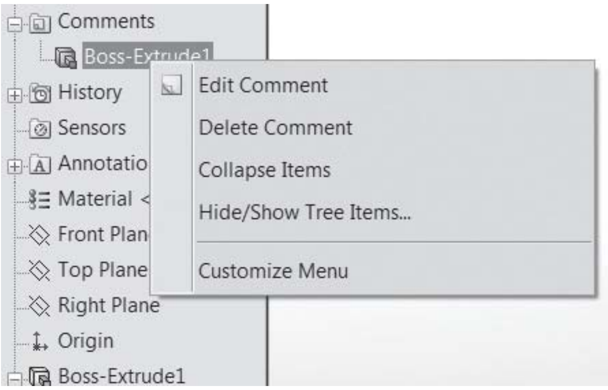
A designer may add comments to features. The designer documents the decisions made during the design process to enable other design team members to understand the design intent of the features design, thus enabling them to modify the design more easily. Click this sequence to add a comment to a feature: Right-click a feature from the features tree > **Comment** (from the window that pops up) > **Add Comment** > type your comments in the window that opens up > **Save and Close**. This sequence opens the sticky note shown in Figure 3.1A. We recommend that your note begins with a date/time stamp and your name followed by your comment. Your comments should be succinct, short, and explain your decisions. When you hover over the feature, your note appears in a balloon, as shown in Figure 3.1B. The features tree has a folder called **Comments** that appears in the tree only when you add comments to features. Features that have comments appear in the **Comments** folder. To edit or delete comments of a feature, right-click it in the **Comments** folder, and select from the menu that pops up shown in Figure 3.1C.



(A) Add comments



(B) View comments

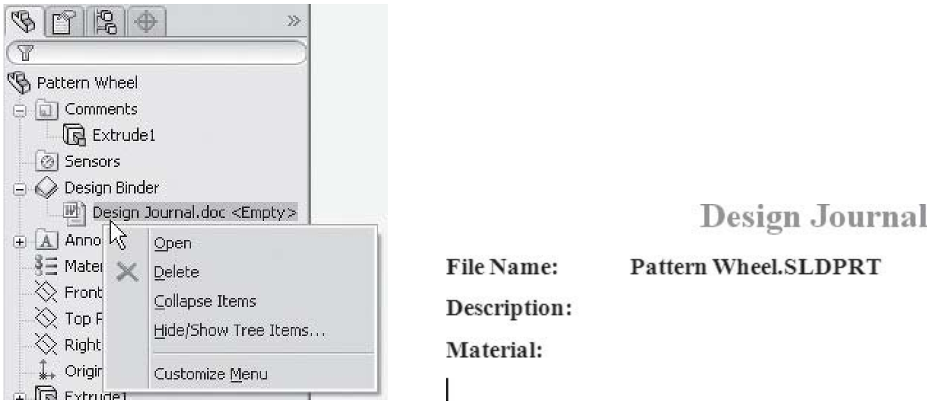


(C) Edit/delete comments

FIGURE 3.1
Comments

3.5 Design Binder

The design binder is an embedded Microsoft Word document that allows you to add more elaborate information about your design than the comments you add in sticky notes. You can add text, screenshots, or any content just like a typical Word document. Follow this sequence to display the **Design Binder** in a features tree: **Tools** > **Options** > **System Options** tab > **FeatureManager** (from pane on left) > select **Show** (for **Design Binder**) from dropdown list > **OK**. This sequence shows the Designer Binder node in the features tree. Note that it shows in every part features tree you open because it is a system-level option. You follow the above sequence to turn the node off by selecting **Hide** from the dropdown list. When you expand the **Design Binder** in the features tree, you see a **Design Journal** Word document (Figure 3.2A). It is an empty document. Double-click it to open it and start documenting (Figure 3.2B) or right click it to see a menu. The document has three lines; the first one is the part name (see Figure 3.2B). You can include screenshots of the design in addition to writing. The journal is saved automatically; there is no need to save the document upon exiting it. It is saved with the part; you would not see it. However, you can save a copy of it if you click Word **File** > **Save Copy As**. Right-click the **DesignJournal.doc** item in the features tree to delete it (Figure 3.2A). If you delete it, an empty document replaces it.



(A) **Design Journal** options (B) **Design Journal** document

FIGURE 3.2
Design Binder

Another important observation is that **Design Binder** enables you to load attachments; use this sequence: right click the **Design Binder** node in features tree > **Add Attachment** > browse to locate and select attachment > **OK**. The attachment is added as a node under the default **Design Journal.doc** node. Right click it to open or delete it. The attachment idea is very useful, as you can upload documents related to the design such as special calculations, group brainstorming sessions, and hand-drawn conceptual sketches. Effectively, the **Design Binder** becomes the central depository for the design.

3.6 Equations

Section 2.16 in Chapter 2 covers how to create and use equations. When you use equations, make sure you use meaningful names for the dimensions variables used in the equation for better readability.

3.7 Design Tables and Configurations

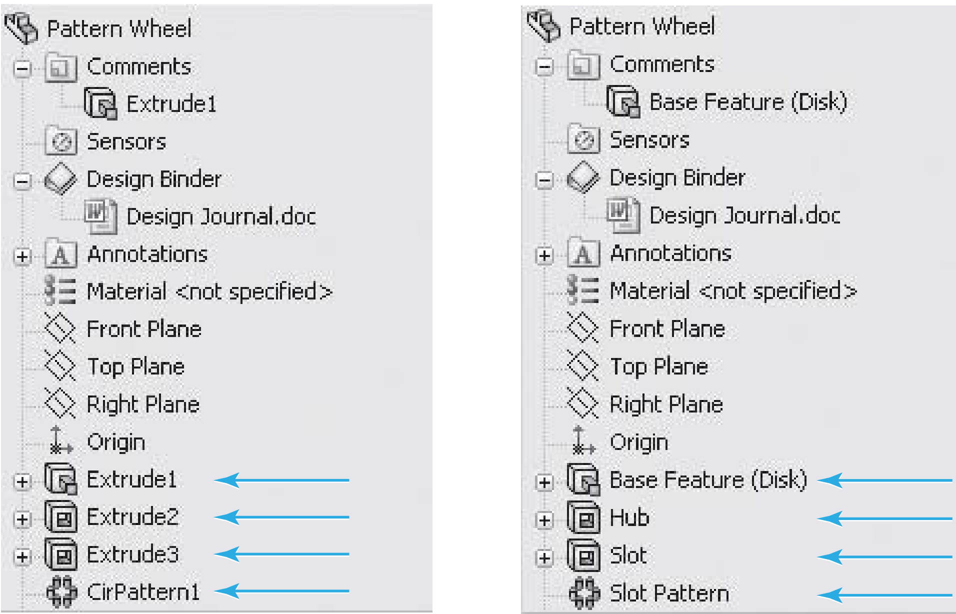
Design tables are a design tool that allows you to change dimensions and create a new instance of the part. These instances form a family of parts (configurations). We cover design tables and configurations later in the book.

3.8 Dimension Names

Changing the default dimension names to more meaningful names enhances design intent. Chapter 2 shows how to change dimension names. These names make equations or design tables more readable.

3.9 Feature Names

Changing the default names of features in the features tree to more meaningful names makes it much easier to follow the tree and makes the design intent clearer for someone who wants to reverse engineer the part creation steps. The default names are usually the feature type followed by a number (e.g., **Extrude1**, **Extrude2**, **Revolve1**, **Cut-Extrude1**). To change a feature's name, click the name twice slowly (do not double-click) in the features tree and type the new name. Figure 3.3 shows the same features tree with default names and better names. To make naming features easier, you can enable the **Name** option (check box) by clicking this sequence: **Tools** > **System Options** > **FeatureManager** > **Name feature on creation**. When you click the green check mark to finish the feature creation, SolidWorks highlights the feature name and waits for you to input a new name.



(A) Default names (B) User-created names

FIGURE 3.3
Naming features

3.10 Folders

You can organize the features tree into folders to help the user follow the design. You can create folders in a part or assembly tree. You can rename new folders and drag features into them, thus reducing the length of the tree. When you select a folder in the features tree, all the parts of the folder are highlighted in the graphics pane on the screen. Conversely, when you select a feature in the graphics pane, its corresponding folder is highlighted and expanded in the features tree. You can drag and drop features into tree folders as long as you do not violate the model topology. SolidWorks stops you.

You must have at least one feature (to click it) to create features tree folders. To create a new folder, right-click any feature on the tree, and select **Add to New Folder** or **Create New Folder** from the popup window that opens. To delete a folder, right click it and select **Delete** from the popup windows that opens. Deleting a folder does not delete its content (features inside it). It only removes the folder from the features tree and leaves its features in their respective locations in the tree.

3.11 Tutorials

The theme for the tutorials in this chapter is to get you to practice creating and documenting design intent.

Tutorial 3–1: Design Intent via Two Modeling Plans

This tutorial applies the concept that design intent is to create the CAD part such that it will lend itself to quick and painless modification in the future. It also applies the concept that a part should be designed the same way it will be manufactured (machined). The part is slider block shown in Figures 3.4 and 3.5. It is an extrusion with two holes, a slot, and a chamfer. There are two modeling plans to create the part: cross section or features. Figures 3.4 and 3.5 show each plan and its steps. The first plan uses a total of two operations, whereas the second plan uses five operations. Both result in the same part; however, later you will see why the features plan is better when it comes to editing the part later.

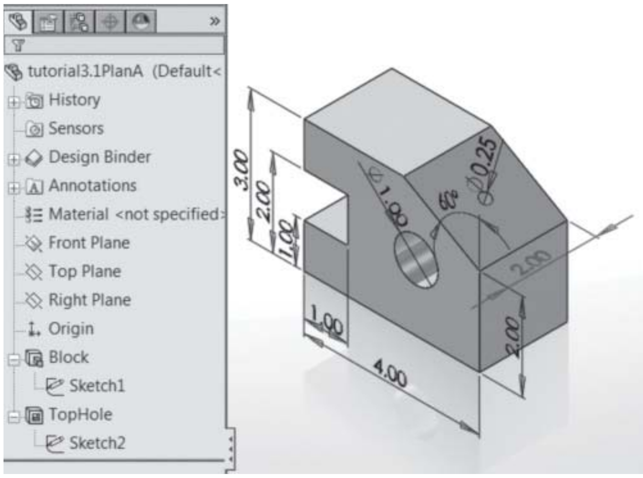


FIGURE 3.4
Cross section modeling plan

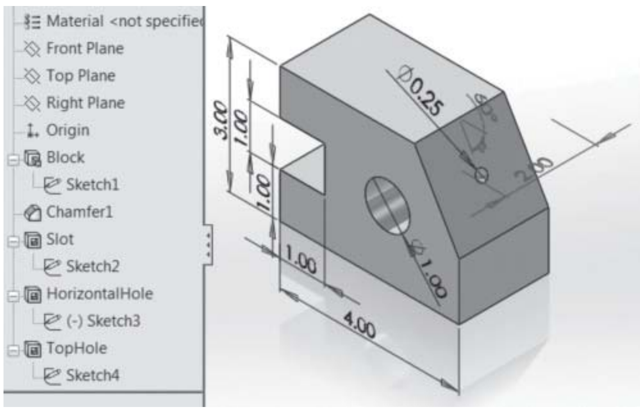
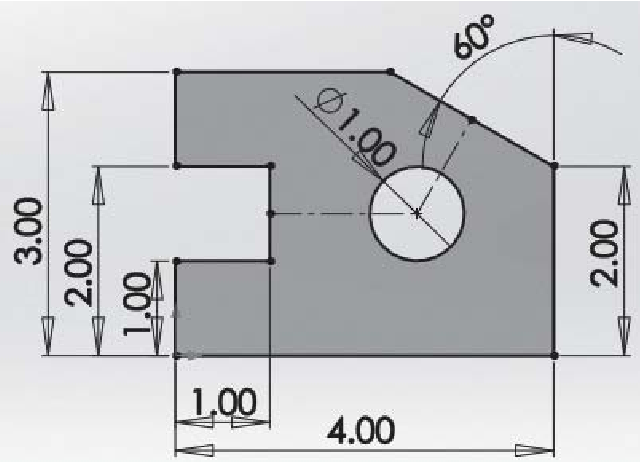


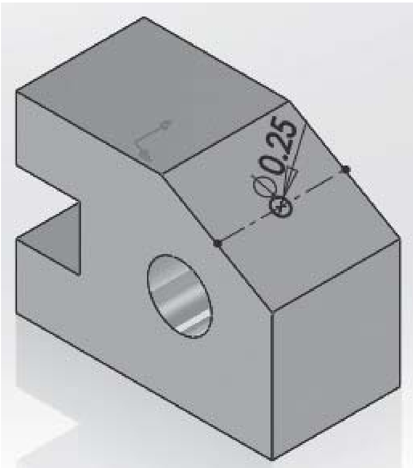
FIGURE 3.5
Features modeling plan

Slider block cross section modeling steps:

Step 1: Create *Sketch1* and *Block* feature of Figure 3.4: **File** > **New** > **Part** > **OK** > **Front Plane** > **Line** on **Sketch** tab > sketch outside lines as shown > dimension as shown (to dimension angle, click both lines) > **Centerline** on **Sketch** tab > sketch two lines passing through the midpoints of the respective lines and perpendicular to them > **Trim Entities** on **Sketch** tab > trim the two centerlines to their intersection point > **Circle** on **Sketch** tab > sketch a circle at this intersection point > dimension as shown > exit sketch > enter 2.0 for thickness **D1** > reverse extrusion direction (drag extrusion arrow in graphics window) > ✔ > **File** > **Save As** > *tutorial3.1PlanA* > **Save**.

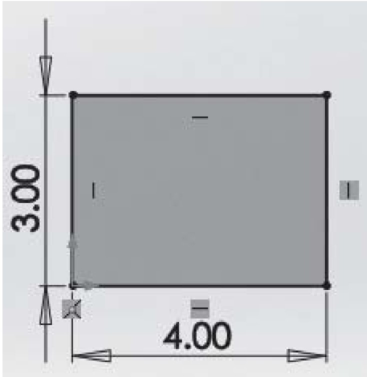


Step 2: Create *Sketch2* and *TopHole* feature: Select inclined face > **Extruded Cut** on **Features** tab > **Centerline** on **Sketch** tab > snap to the two midpoints of the face edges as shown to sketch the centerline > **Circle** on **Sketch** tab > snap to midpoint of the centerline just created and sketch a circle> dimension circle as shown > exit sketch > **Up To Surface** from **Direction 1** dropdown > select face of horizontal hole > ✔.

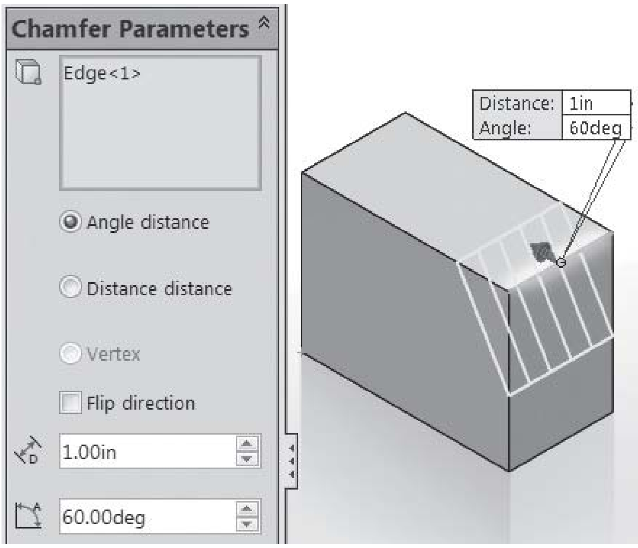


Slider block features modeling steps:

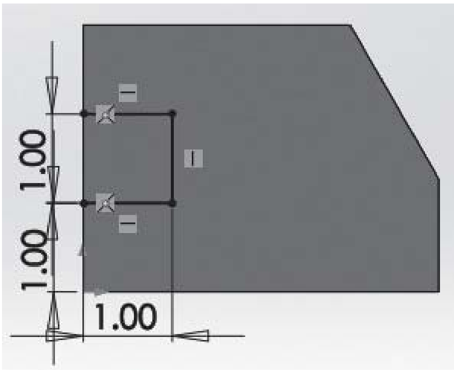
Step 1: Create *Sketch1* and *Block* feature of Figure 3.5: **File** > **New** > **Part** > **OK** > **Front Plane** > **Corner Rectangle** on **Sketch** tab > sketch rectangle > dimension as shown > exit sketch > enter 2.0 for thickness **D1** > reverse extrusion direction (drag extrusion arrow in graphics window) > ✔ > **File** > **Save As** > *tutorial3.1PlanB* > **Save**.



Step 2: Create *Chamfer1* feature: **Fillet** on **Features** tab > **Chamfer** from dropdown that opens > select top edge of inclined face as shown > enter 1 for **D** and 60 for angle > ✔.

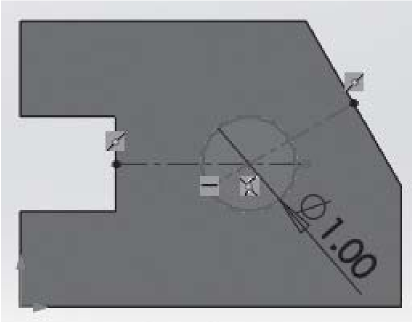


Step 3: Create *Sketch2* and *Slot* feature: Select *Block* front face > **Features** tab > **Extruded Cut** > **Line** on **Sketch** tab > sketch three lines as shown > dimension as shown > exit sketch > **Through All** > ✔.

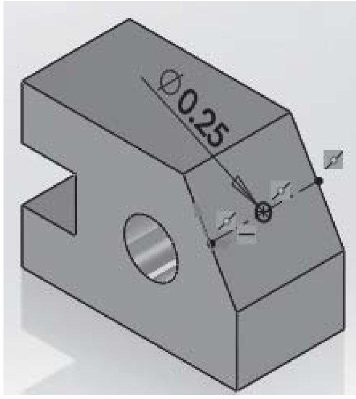


Note: Ensure the two horizontal lines snap to the *Block* left edge as shown by the relation symbols. Also, work in the front view when sketching. If you try to sketch in the ISO view, SolidWorks does not know your intent.

Step 4: Create *Sketch3* and *HorizontalHole* feature: Select *Block* front face > **Features** tab > **Extruded Cut** > **Centerline** on **Sketch** tab > sketch one line passing through midpoint of inclined edge and perpendicular to it as shown > repeat to create the horizontal centerline > **Circle** on **Sketch** tab > click the intersection point of two lines just created and drag to sketch circle > dimensions as shown > exit sketch > **Through All** > ✔.



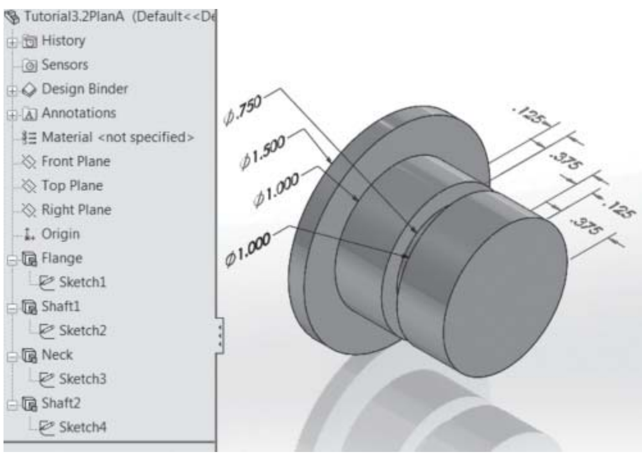
Step 5: Create *Sketch4* and *TopHole* feature: Select inclined face > **Extruded Cut** on **Features** tab > **Centerline** on **Sketch** tab > snap to the two midpoints of the face edges as shown to sketch the centerline > **Circle** on the **Sketch** tab > snap to midpoint of the centerline just created and sketch a circle > dimension circle as shown > exit sketch > **Up To Surface** from **Direction 1** dropdown > select face of horizontal hole > ✔.



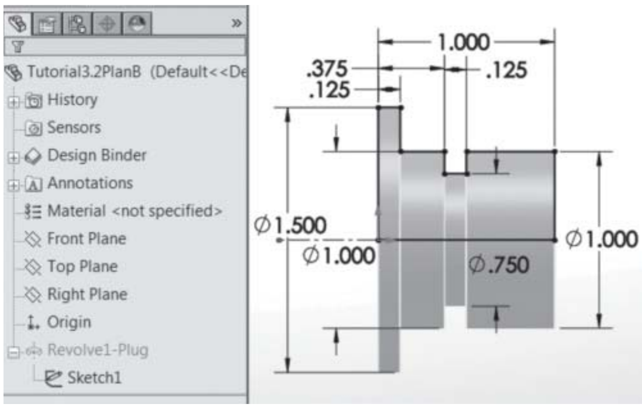
HANDS-ON FOR TUTORIAL 3–1. A revision of the design calls for the horizontal hole in the front face to be blind with depth of 1.0″. Modify the block of each plan accordingly. Which plan is easier to modify and why?

Tutorial 3–2: Design Intent via Three Modeling Plans

This tutorial applies the concept that design intent is to create the CAD part in the same way it will be manufactured (machined). Figure 3.6 shows a drain plug. We can think of at least three modeling plans to create the plug. In the first plan (Figure 3.6A), called *Additive modeling plan*, we continuously add features to previous features. For the second plan (Figure 3.6B), *Cross-section modeling plan*, we revolve the cross section to

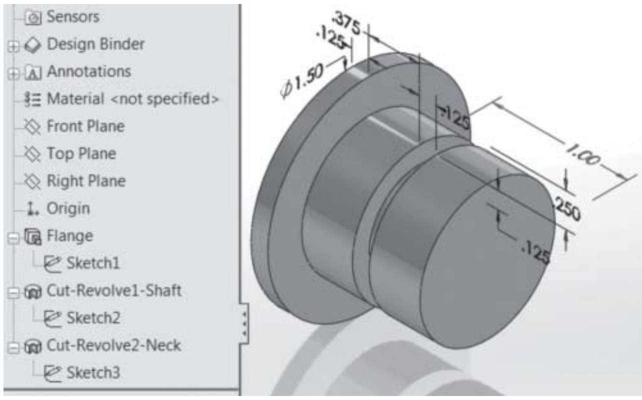


(A) Additive modeling plan



(B) Cross-section modeling plan

FIGURE 3.6
Drain plug



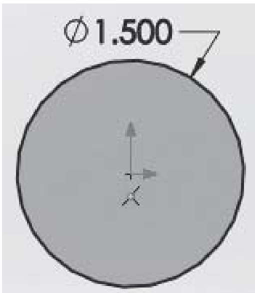
(C) Subtractive (manufacturing) modeling plan

FIGURE 3.6
(*continued*)

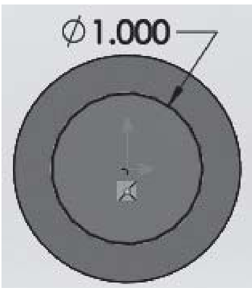
create the entire part in one feature. Finally in the third plan (Figure 3.6C), *Subtractive (manufacturing) modeling plan*, we create the part using an extrusion and two revolve cuts to mimic how the part would be manufactured on a lathe.

Figure 3.6A Additive modeling plan steps:

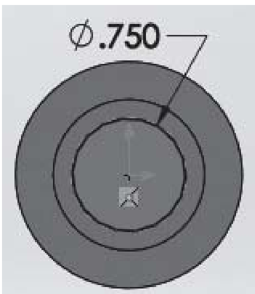
Step 1: Create *Sketch1* and *Flange* feature: **File** > **New** > **Part** > **OK** > **Right Plane** > **Features** tab > **Extruded Boss/ Base** > **Circle** on **Sketch** tab > click origin and sketch circle > dimension as shown > exit sketch > enter 0.125 for thickness **D1** > ✔ > **File** > **Save As** > *tutorial3.2PlanA* > **Save.**



Step 2: Create *Sketch2* and *Shaft1* feature: Select right face of *Flange* feature > **Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch circle > dimension as shown > exit sketch > enter 0.375 for thickness **D1** > ✔.



Step 3: Create *Sketch3* and *Neck* feature: Select *Shaft1* right face > **Extruded Boss/ Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch circle > dimension as shown > exit sketch > enter 0.125 for thickness **D1** > ✔.



Step 4: Create *Sketch4* and *Shaft2* feature: Select *Neck* right face > **Extruded Boss/ Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch circle > dimension as shown > exit sketch > enter 0.375 for thickness **D1** > ✔.

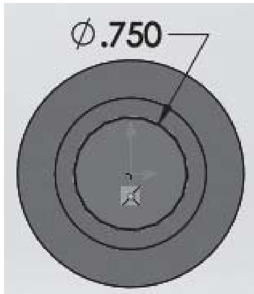


Figure 3.6B Cross-section modeling plan steps:

Step 1: Create *Sketch1* and *Revolve1-Plug* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Features** tab > **Extruded Boss/Base** > **line** on **Sketch** tab > sketch lines shown > dimension as shown > **Centerline** on **Sketch** tab > sketch a horizontal line as shown > exit sketch > select centerline as **Axis of Revolution** > ✔ > **File** > **Save As** > *tutorial3.2PlanB* > **Save.**

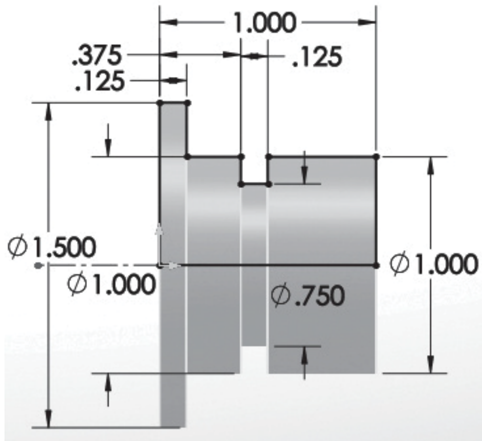
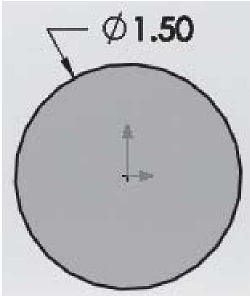
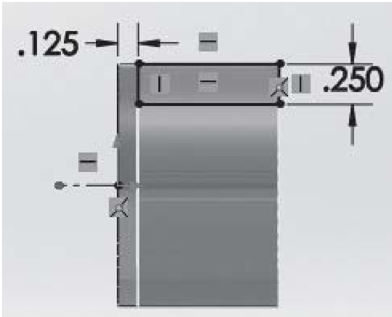


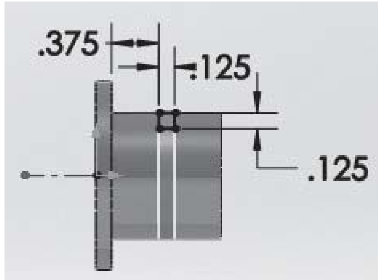
Figure 3.6C Subtractive modeling plan steps:



Step 1: Create *Sketch1* and *Flange* feature: **File > New > Part > OK > Right Plane > Features** tab > **Extruded Boss/Base > Circle** on **Sketch** tab **>** click origin and sketch circle > dimension as shown > exit sketch **>** enter 1.0 for thickness **D1 >** ✔ > **File > Save As** > *tutorial3.2PlanC* > **Save.**



Step 2: Create *Sketch2* and *Cut-Revolve1-Shaft* feature: **Front Plane > Revolved Cut** on **Features** tab **> Line** on **Sketch** tab **>** sketch lines shown > dimension as shown > **Centerline** on **Sketch** tab > sketch centerline shown > exit sketch **>** select centerline as **Axis of Revolution** > ✔.



Step 3: Create *Sketch3* and *Cut-Revolve2-Neck* feature: **Front Plane > Revolved Cut** on **Features** tab **> Line** on **Sketch** tab **>** sketch lines shown > dimension as shown > **Centerline** on **Sketch** tab **>** sketch centerline shown > exit sketch **>** select centerline as **Axis of Revolution** > ✔.

> ***HANDS-ON FOR TUTORIAL 3–2.*** Use each of the three plans to modify the part by deleting the groove (neck). Notice how easy it is to work with Plan C. What are the difficulties you encountered during modifying Plan A and Plan B?

Tutorial 3–3: Design Intent via Design Specifications

Figure 3.7 shows a hand wheel. The design intent is embedded in the wheel design specs as follows. The hub shall remain at the center of the wheel. The spokes will connect the wheel to the hub. The part shall be designed such that we can easily modify the wheel and hub diameter.

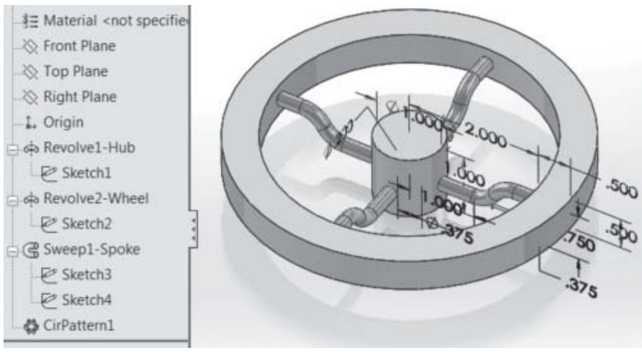
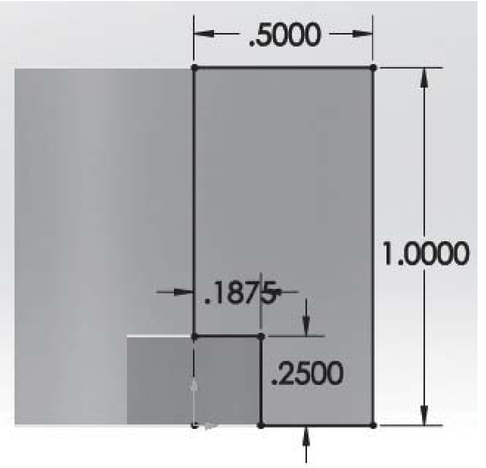
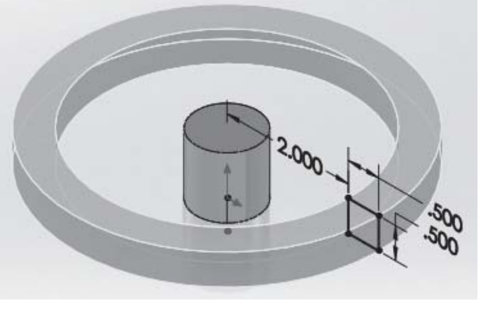


FIGURE 3.7
Hand wheel

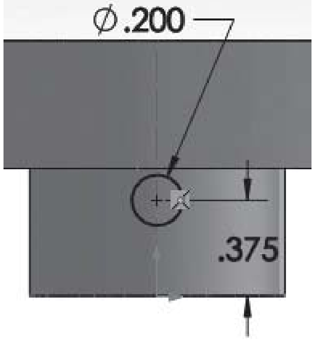
Step 1: Create *Sketch1* and *Revolve1-Hub* feature: **File > New > Part > OK > Front Plane > Features** tab > **Revolved Boss/Base > Line** on **Sketch** tab **>** sketch horizontal and vertical lines and dimension as shown > **Centerline** on **Sketch** tab > sketch vertical centerline passing through origin > exit sketch > select centerline as **Axis of Revolution** > ✔ > **File > Save As** > *tutorial3.3* > **Save**.



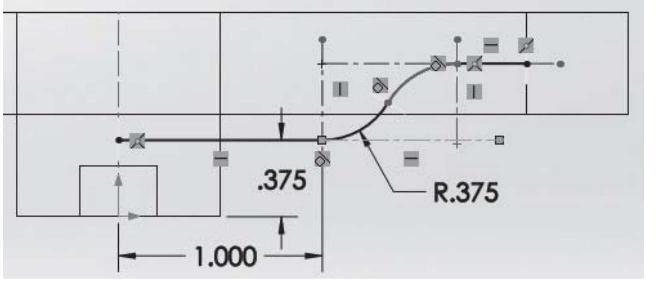
Step 2: Create *Sketch2* and *Revolve2-Wheel* feature: **Front Plane > Features** tab > **Revolved Boss/Base > Rectangle** on **Sketch** tab **>** sketch rectangle and dimension as shown > select the top line of rectangle + **Ctrl** on keyboard + select the top edge of *Revolve1-Hub* feature created in Step 1 > select **Collinear** from **Add Relations** options in left pane > ✔ **> Centerline** on **Sketch** tab > sketch center-line shown > exit sketch > select centerline as **Axis of Revolution** > ✔.



Step 3: Create *Sketch3*: This sketch and the next one (*Sketch4*) are needed to create the sweep feature of the wheel spoke. **Right Plane** > **Circle** on **Sketch** tab > click vertical centerline from Step 1 and drag to sketch circle > dimension as shown > exit sketch.

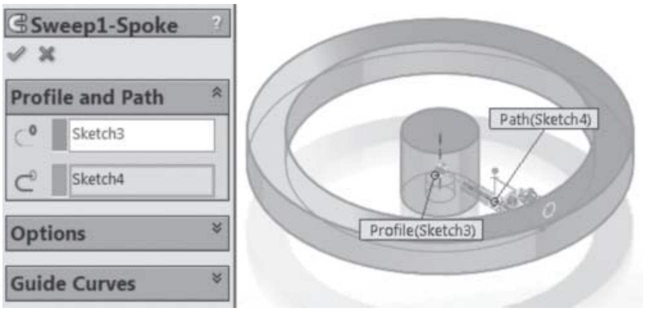


Step 4: Create *Sketch4*:



Front Plane > select **Wireframe** from **Display Style** dropdown (shown on **Heads-up View** toolbar; hover over icons until you find it) > **Line** on **Sketch** tab > sketch a horizontal line (make sure it does not snap to internal edge of wheel yet) > create another horizontal line snapping to centerline as shown > **3 Point Arc** on **Sketch** tab > sketch two arcs as shown (we use extra centerline to show how the arc centers align with the two horizontal lines) > make sure the left arc endpoint snaps to the left line and the right arc endpoint snaps to the left arc > select left line + **Ctrl** key + select left arc > **Tangent** from **Add Relations** options > select one arc + **Ctrl** key + select other arc > **Tangent** from **Add Relations** options > select right arc right end-point + **Ctrl** key + select left endpoint of right line > **Merge** from **Add Relations** options > trim right arc and right line if needed > dimension all as shown > exit sketch.

Step 5: Create *Sweep1-Spoke* feature:



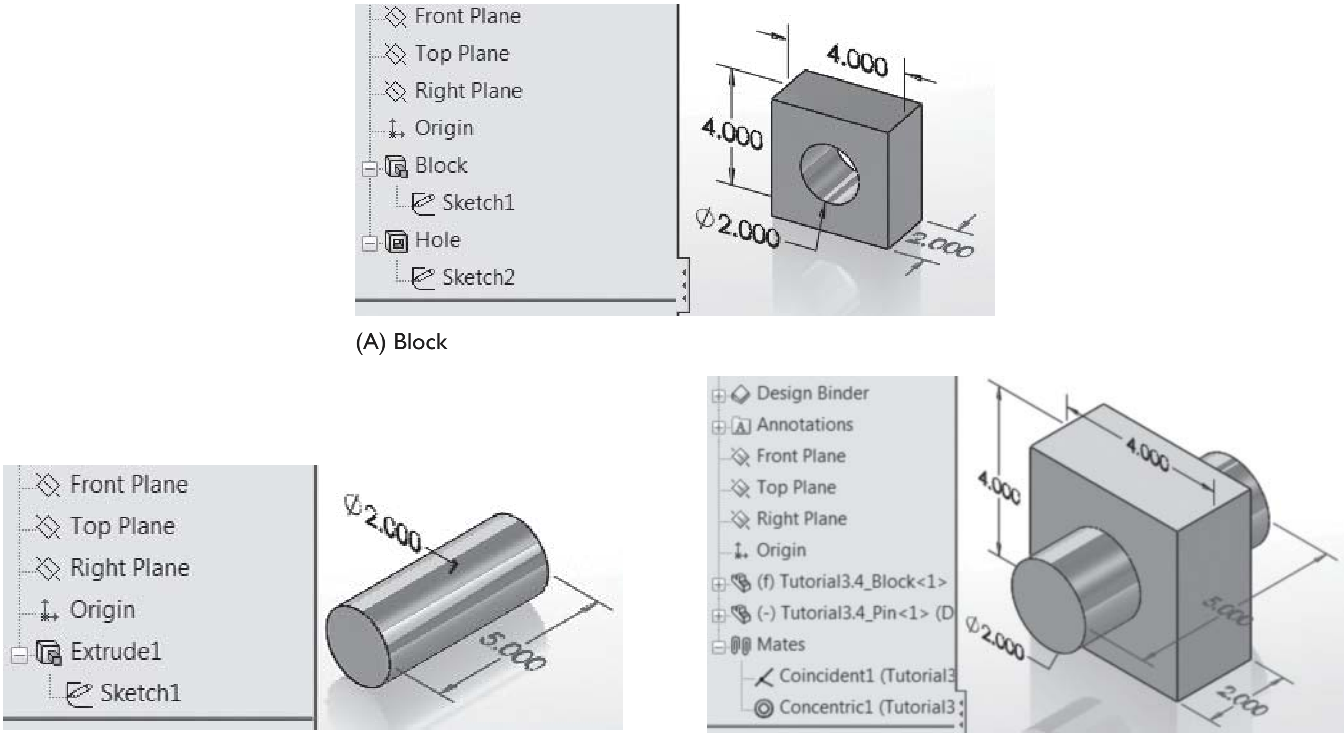
Feature tab > **Swept Boss/Base >** select *Sketch3* as the **Profile >** select *Sketch4* as the **Path >** ✔.

Step 6: Create *CirPattern1* feature: **Features** tab > select **Circular Pattern** from **Linear Pattern** drop-down > expand features tree > select *Sweep1-Spoke* from tree for **Features to Pattern** (on left pane) > click first box under **Parameters** on left pane > select the center axis > type 4 for number of instances > ✔.

HANDS-ON FOR TUTORIAL 3–3. We enforce the design intent in our modeling plan by using the following relation: **Coincident** relation between the right end of the sweep path and the inner edge of the wheel. Remove this relation and modify the wheel radius to 1.00″. What happens to the spokes?

Tutorial 3–4: Design Intent via Mating Conditions

We create the Pin Block assembly shown in Figure 3.8. This tutorial applies the concept that design intent is to create the CAD part in the same way it will be used in an assembly. The pin shall remain centered on the block regardless of changes. We will create the pin and the block about a center plane (**Front Plane**). We will then use this center plane to mate the two parts.



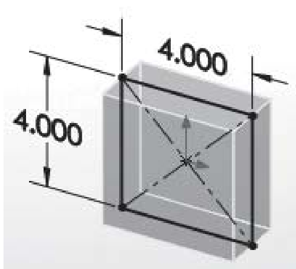
(A) Block

(B) Pin

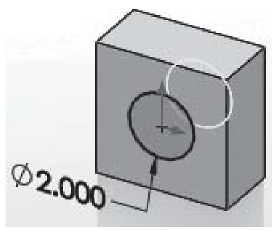
(C) Assembly

FIGURE 3.8
Pin block assembly

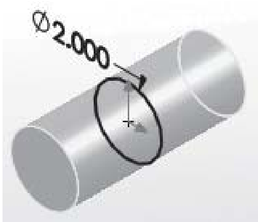
Step 1: Create *Sketch1* and *Block* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Features** tab > **Extruded Boss/Base** > **Center Rectangle** on **Sketch** tab > click origin and drag to sketch > dimension as shown > exit sketch > select **Mid Plane** from **Direction 1** dropdown > enter 2 for thickness **D1** > ✔ > **File** > **Save As** > *tutorial3.4Block* > **Save**.



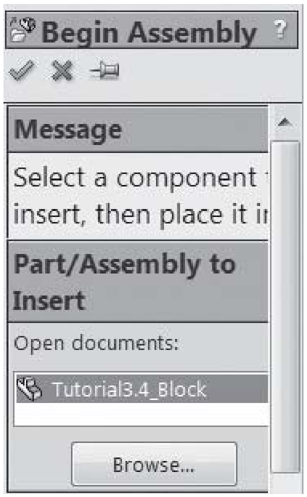
Step 2: Create *Sketch2* and *Hole* feature: Front face of *Block* > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch a circle> dimension circle as shown > exit sketch > **Through All** from **Direction 1** dropdown > ✔ .



Step 3: Create *Sketch1* and *Extrude1*: **File > New > Part > OK > Front Plane > Features** tab > **Extruded Boss/Base > Circle** on **Sketch** tab **>** click origin and drag to sketch > dimension as shown > exit sketch > select **Mid Plane** from **Direction 1** dropdown > enter 5 for thickness **D1** > ✔ > **File > Save As** > *tutorial3.4Pin* > **Save**.



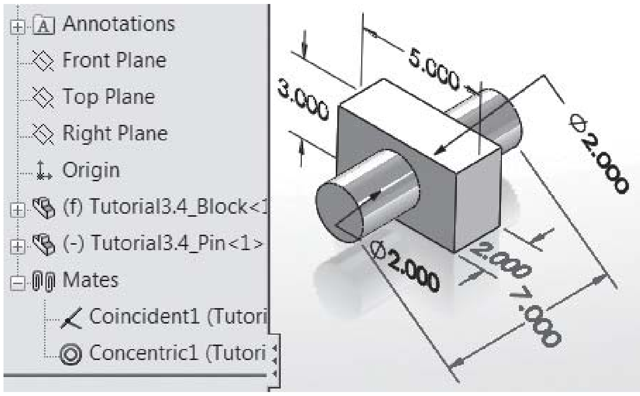
Step 4: Create assembly: **File > New > Assembly > OK**



Browse (button on left pane of screen and shown here to the right) > locate *tutorial3.4Block* part file and double click it to add it > ✔ without clicking in graphics pane (this fixes the part origin to assembly origin). This is a good practice to use when inserting first part of an assembly > **Insert Component** on **Assembly** tab > locate *tutorial3.4Pin* part file and double click it to add it > click in graphics pane to insert it > **File > Save As** > *tutorial3.4BlockPin* > **Save**.

Step 5: Mate the parts: **Mate** on **Assembly** tab > **Coincident** > expand assembly tree > expand each part features tree > select **Front Plane** of each part > ✔ > **Concentric** > select *Hole* face and *Pin* cylindrical face > ✔.

Step 6: Modify parts geometry: Open the two parts files > change *Block* size to 5 (width) × 3 (height) and *Pin* extrusion **D1** to 7 > save both files > open assembly file > update it and observe **Coincident** mate is still intact.



HANDS-ON FOR TUTORIAL 3–4. Currently, the hole and the shaft diameters are not related. Add a new design intent in the form of an equation that forces the two diameters to be equal. Change the value from 2.000″ to 3.000″.

problems

1. Why is design intent important in CAD design?
2. List and discuss briefly the different methods to capture design intent.
3. List and discuss briefly the different methods to document design intent.
4. Use the **Comment** method to document (add comment to any sketch and/or feature) the design intent for any of the Chapter 1 or 2 problems.
5. Use the **Design Binder** method to document (add a binder to part) the design intent for any of the Chapter 1 or 2 problems. Also add an attachment to the **Design Binder**.
6. Use the **Folder** method to streamline the features tree of any of the Chapter 1 or 2 problems. Combine/group related features into one folder.

PART

II

Basic Part Modeling

The primary goal of this part is to explore and cover the details of engineering drawings, assemblies, and rendering. We have covered these topics briefly in Part I. Each topic has its own chapter in this Part II to understand it in depth. Part II also covers the full set of features that can be used in CAD modeling. In Part I, we purposely have limited our models to the basic features of extrusions and revolves.

Chapter 4 (Features and Macros) is all about learning when and how to use the full set of features available to enable you to design any parts and complex geometry you may run into. Chapter 5 (Drawings) covers the details of drawings including the creation and control of the title block. Chapter 6 (Assemblies) covers assembly details including the bottom-up and top-down approaches. Chapter 7 (Rendering and Animation) closes Part II by showing how to create realistic rendering of parts and assemblies, including showing material and texture. CAD visualization is important to convey and present designs efficiently.

CHAPTER

4

Features and Macros

4.1 Introduction

We have been using a limited set of features thus far in the book. We have used extrusions and revolves. In terms of SolidWorks, we have used **Extruded Boss/Base** and **Revolved Boss/Base**, and their subtracting counterparts, **Extruded Cut** and **Revolved Cut**. These four features create one class of parts: the ones with constant cross sections. We use extruded boss or cut to create uniform thickness parts. We use the revolved boss or cut to create axisymmetric parts. Amazingly, these four features can create about 80% or more of the mechanical parts.

Other classes of parts exist that these four features cannot create. These are the parts whose cross sections are variable, or parts that have nonplanar faces or other geometric shapes. The features that allow us to create these types of parts are loft, sweep, the hole wizard, rib, draft, shell, and dome as shown in Figure 4.1. We cover all these features in this chapter. You can also access more features by clicking this sequence: **Insert** (menu) > **Features**.

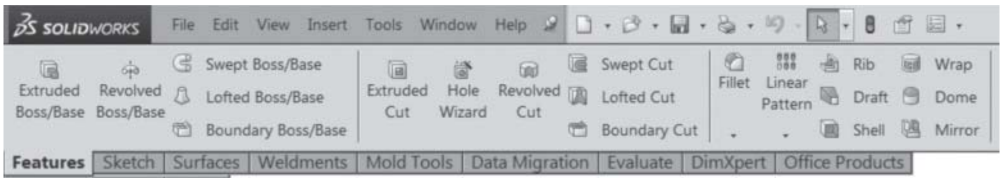


FIGURE 4.1
Available features

A **feature** is defined as a solid that when combined with other features (solids) creates parts. Conversely, a CAD part consists of a set of features created in a certain sequence stored in its features tree. Some features, such as bosses and cuts, originate as sketches while others, such as shells and fillets, modify other features. Features are always listed in the features tree (**FeatureManager Design Tree** as SolidWorks calls it) of the part.

Today's modeling is referred to as feature-based modeling, and the resulting models are known as feature-based models. The first feature you create in a part is known as the base feature. You use the base feature as the basis to create other features. A base feature obviously cannot be negative (i.e., a cut). SolidWorks would not allow you to do that. When you begin feature creation, **Extruded Boss/Base** and **Revolved Boss/Base** are the only selectable feature types. While you would also expect **Swept Boss/Base** and **Lofted Boss/Base** to be selectable when you begin feature creation, they are not. The **Lofted Boss/Base** becomes selectable only after you create a sketch (profile), and the **Swept Boss/Base** becomes selectable after you create a cross section and a path (sweep direction).

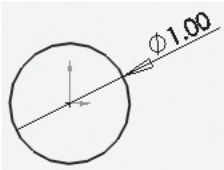
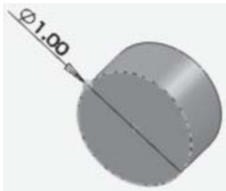
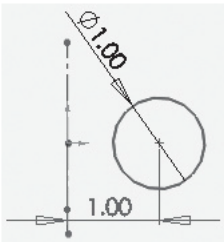
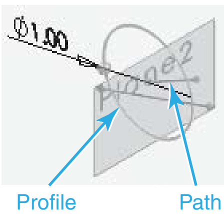
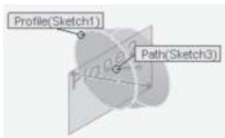
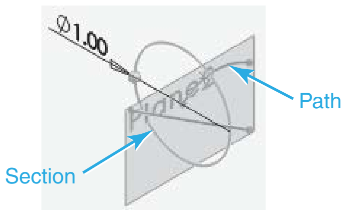
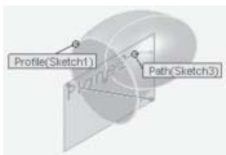
4.2 Features

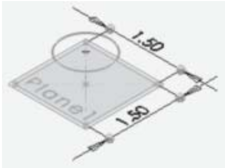
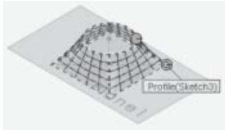
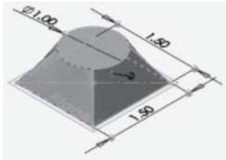
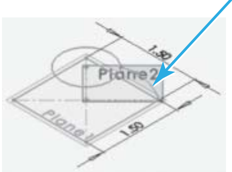
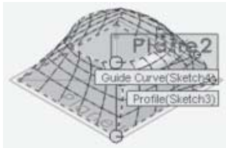
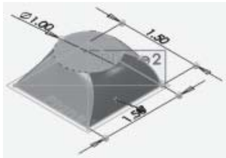
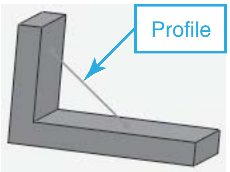
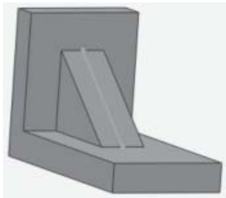
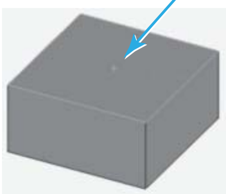
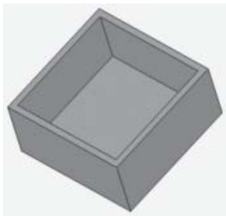
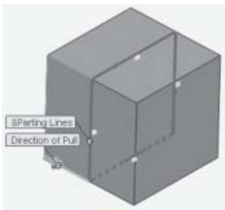
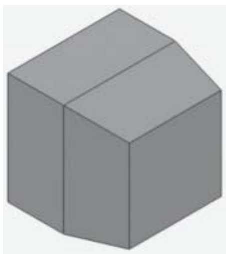
If we want to master feature-based modeling, we should be able to answer three fundamental questions:

1. What are the available features that a CAD/CAM system offers for modeling parts?
2. What is the input required to create each feature?
3. Which feature should we use for a given modeling problem?

Section 4.1 and Figure 4.1 provide the answer to the first question. Table 4.1 answers the other two questions. It shows a simple basic example of each feature. Keep in mind that the third question may have multiple answers; one of them is always the best answer. For example, we may use a loft or a sweep. However, if the part has a constant cross section along a curve, sweep is better to use because it requires fewer steps to create the part. If the part has a variable cross section, a loft is better to use. The tutorials in this chapter provide some modeling examples.

TABLE 4.1 Available features

No.	Feature	Input (sketch)	Resulting feature	When to use in modeling?
1	Extrusion	Cross section and a thickness 		• Use for parts with constant cross section (CS) and uniform thickness (UT). • If needed, break part into subparts, each with a constant CS and UT.
2	Revolve	Cross section, an axis of revolution, and an angle of revolution 		• Use for parts that are axisymmetric. • If needed, break part into subparts, each of which is axisymmetric.
3	Sweep	Linear sweep: cross section and a line as a path 		• Use for parts with constant cross section (CS) along a linear direction (path) that may or may not be perpendicular to the cross section. • If the path is perpendicular to the cross section, the linear sweep becomes an extrusion.
		Nonlinear sweep: cross section and a curve as a path 		• Use for parts with constant cross section (CS) along a nonlinear direction that may or may not be perpendicular to the cross section.

No.	Feature	Input (sketch)	Resulting feature	When to use in modeling?
4	Loft	Linear loft: at least two cross sections (profiles) 	 	• Use for parts with variable cross section along a given direction. • The cross sections are blended linearly from one section to the other.
		Nonlinear loft: at least two cross sections (profiles), and a curve as a guide curve Guide Curve 	 	• Use for parts with variable cross section along a given direction. • The cross sections are blended nonlinearly from one section to the other, along the guide curve.
5	Rib	Rib profile (e.g., line or stepwise line) 		• Use when a stiffener between angled walls (faces) of a part is required to increase part structural strength.
6	Shell	Shell face and shell wall thickness Shell face 		• Use when you need to remove material from an existing part. • The material removal (shelling) occurs in a direction perpendicular to the selected shelling face. • While you can achieve same result using an extrude cut for simple shells, a shell operation is faster to use.
7	Draft	Direction of pull, parting lines, and a draft angle. The direction of pull must be perpendicular to the parting lines. 		• Use when you need to draft faces at an angle; usually used for injection molding to allow pulling the molded part from the mold cavity.

The other features shown in Figure 4.1 and not covered in Table 4.1 are covered in the tutorials of the chapter.

Example 4.1 Create the free-form torus shown in Figure 4.2.

Solution The torus shown in Figure 4.2 is a variation of the torus (donut shape) feature (No. 2) shown in Table 4.1. While that feature of Table 4.1 is a revolve, the free-form torus shown in Figure 4.2 can only be created as a sweep. The key modeling concept here is to use pierce relations to force the torus cross section (small circle) to conform to the sweep path (large circle) and the guide curve (closed spline), as we show in Figure 4.2. We pierce the small circle to the spline and pierce the center of the small circle to the large circle. These two pierce conditions force the small circle to become "elastic," that is, it expands and shrinks, as it must always touch the spline and the big circle. Pierce condition is only available to pierce a point to a curve. You cannot pierce two curves. SolidWorks enables the pierce condition within the right context.

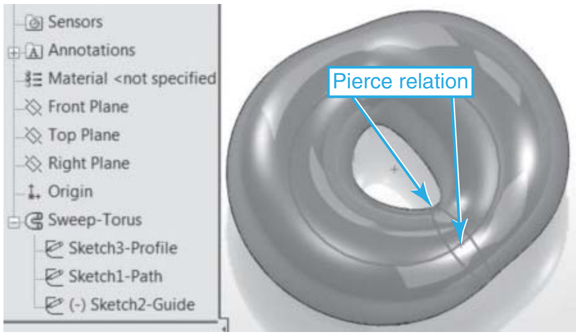
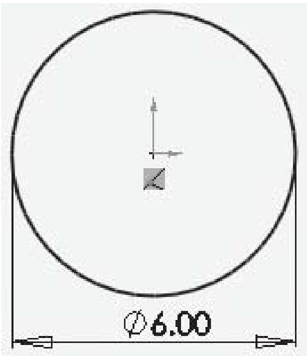
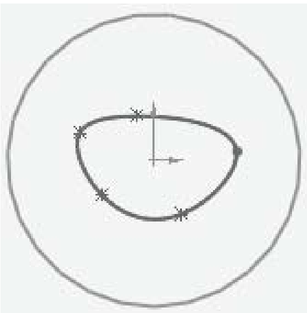


FIGURE 4.2
Free-form torus

Step 1: Create *Sketch1-Path* (sweep path): **File** > **New** > **Part** > **OK** > **Top Plane** > **Sketch** tab > **Circle** on **Sketch** tab > click origin to sketch and dimension as shown > exit sketch > **File** > **Save As** > *example4.1* > **Save.**



Step 2: Create *Sketch2-Guide* (sweep guide curve): **Top Plane** > **Sketch** tab > **Spline** on **Sketch** tab > sketch free spline as shown > exit sketch.

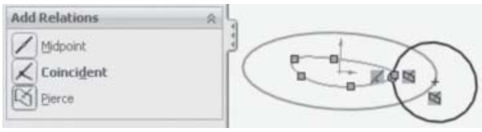


Note: *Sketch1-Path* and *Sketch2-Guide* are two separate sketches, each using **Top Plane**.

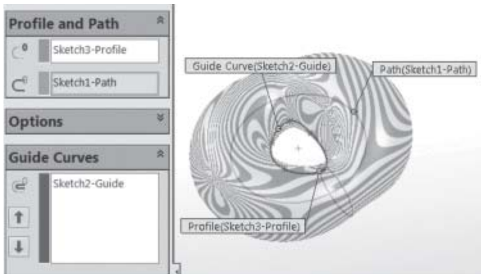
Step 3: Create *Sketch3-Profile* (sweep profile): **Front Plane** > **Sketch** tab > **Circle** on **Sketch** tab > sketch a circle anywhere > **Point** on **Sketch** tab > click circle anywhere.

Note: Do not dimension circle as it over-constrains it when we apply the pierce relation.

Step 4: Create pierce relations: While *Sketch3-Profile* is still open from Step 3 > click small circle center + **Ctrl** + select large circle > **Pierce** relation > ✔ > select point created on circle + **Ctrl** + spline > **Pierce** relation > ✔ > exit sketch.



Step 5: Create *Sweep-Torus* feature: **Features** tab > **Swept Boss/base** > select *Sketch3-Profile* as the **Profile** > select *Sketch1-Path* as the **Path** > select *Sketch2-Guide* as the **Guide Curve** > ✔.



HANDS-ON FOR EXAMPLE 4.1. Re-create the free-form torus by replacing the spline by a circle that is not centric with the large circle.

4.3 Spur Gears

Gears are an important and essential mechanical element in mechanical design. A wide range of products and applications uses gears. There are various types of gears: spur, helical, bevel, spiral, worm, planetary, and rack and pinion, to name a few. Spur gear is the simplest type of gear, which we cover here. Typical mechanical design courses in colleges cover the principles and design of gears. In this section, we cover spur gears from a CAD point of view (i.e., how we construct a gear once it is designed). While gears are standard elements that can be purchased off the shelf (they can also be inserted from SolidWorks Toolbox into a part or assembly file), it is important to learn how to create a gear feature in a CAD/CAM system.

A gear tooth is the intricate part of a gear. Figure 4.3 shows two meshing gears. Figure 4.4A shows the conjugate line and pressure angle. Figure 4.4B shows the involute

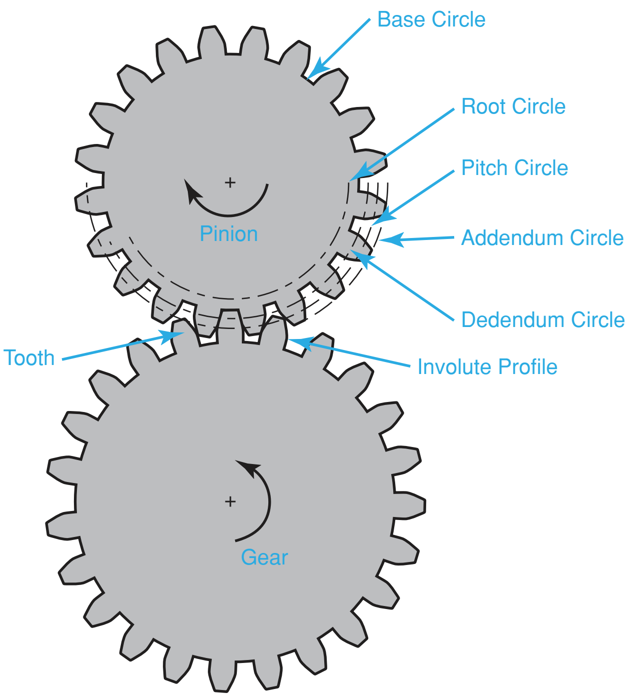


FIGURE 4.3
Meshing gears

profile. Gearing and gear meshing ensure that two disks (the two gears) in contact roll against one another without slipping. Moreover, the gear teeth should not interfere with the uniform rotation that one gear would induce in the other, a requirement known as the conjugate action. The conjugate action also ensures that the perpendicular line to a tooth profile at its point of contact with a tooth from the other gear always passes through a fixed point on the centerline connecting the centers of the two meshing gears. Figure 4.4A shows the conjugate line. The conjugate line is also known as the line of force because the driving force from the driving gear (driver) is transmitted in the direction of this line to the other gear (driven). The angle between the perpendicular radius to the conjugate line and the centerline is always constant for two meshing gears. This angle is known as the *pressure angle* and is shown as the angle Ø in Figure 4.4A.

The key to successful functional gears is the conjugate action. While various profiles can produce conjugate action, the involute profile is the best because it allows for imperfections in gear manufacturing and yet maintains the conjugate action. The imperfection may produce a slightly different distance between the two shafts of the gears from the designed value. Figure 4.4B shows how the shape of the involute profile is generated. An **involute** is defined as the path of the end point of a cord when it is pulled straight (held taut) and unwrapped from a circular disk as shown in Figure 4.4B. The involute geometry ensures that a constant rotational speed of the driving gear produces a constant rotational speed in the driven gear. For spur gears, the teeth are cut perpendicular to the plane of the gear, where the involute profile resides.

The creation of a gear CAD model requires two basic concepts: knowledge of the gear geometry and the involute equation. The geometry is shown in Figure 4.3. The

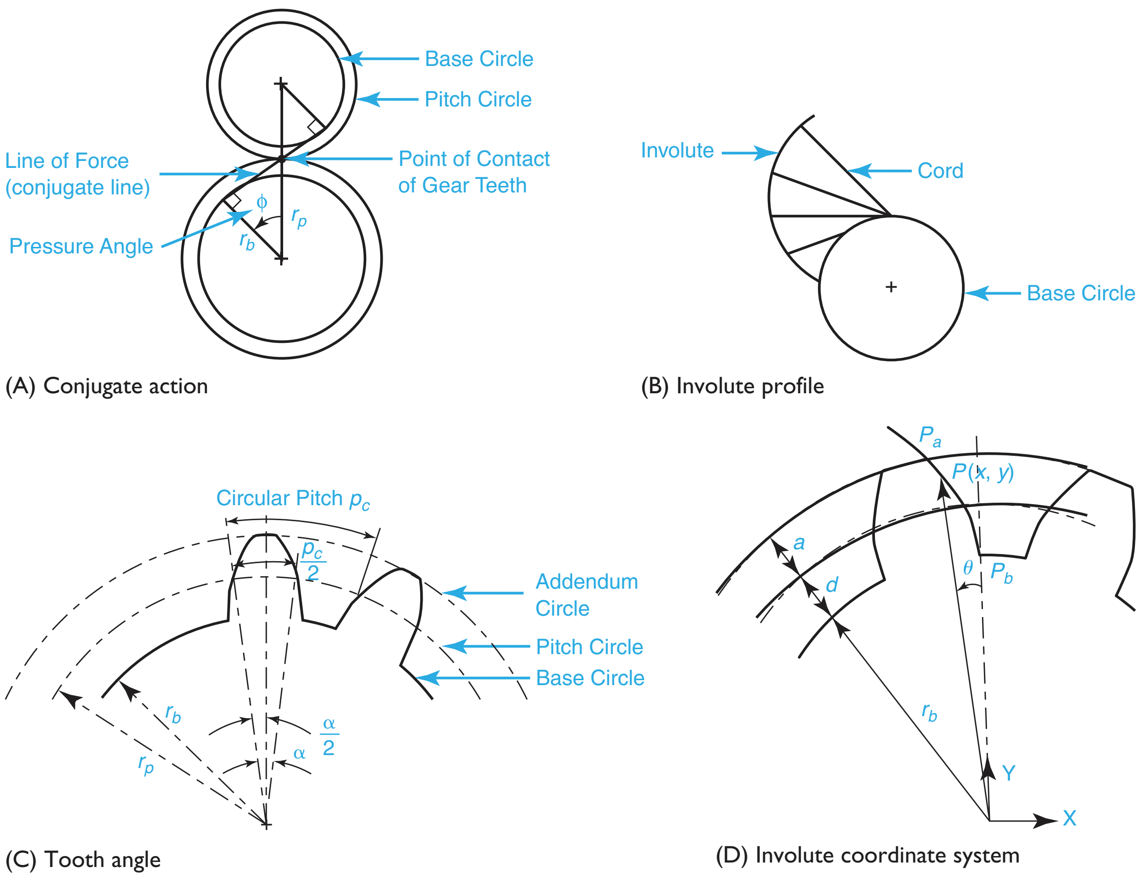


FIGURE 4.4
Details of a gear tooth

base circle is the circle where the involute profile begins. The **pitch circle** defines the contact (pitch) point between the two gears (see Figure 4.4A). The **dedendum circle** is usually the same as the base circle, as can be concluded from Figure 4.3A (dedendum $d = r_p - r_b$). The **addendum circle** is the circle that defines the top of the tooth as shown in Figure 4.4C (addendum $a = r_a - r_p$, where r_a is the addendum circle radius). Typically, the addendum and the dedendum are equal. In such case, the pitch and base circle sizes determine the values for both. The **root circle** is smaller than the base circle to allow cutting the tooth during manufacturing. The tooth profile between the base and root circles is not an involute. It could be any geometry such as line.

The creation of a gear CAD model requires two steps: calculate the tooth angle α and the tooth involute profile. While many books on mechanical engineering design offer extensive in-depth coverage of gear analysis, we offer a simplified, but accurate, version to enable us to create a CAD model of the gear. We begin with the definition of circular pitch. As shown in Figure 4.4C, the **circular pitch, p_c,** is defined as the distance along the pitch circle between corresponding points on adjacent teeth. As shown in Figure 4.4C, we use p_c as the circular pitch of the gear, r_p as the pitch circle radius, and α as the tooth angle. Using these variables, we can write:

$$p_c = \frac{\pi d_p}{N} \tag{4.1}$$

Where $d_p = 2r_p$ is the pitch circle diameter, and N is the number of gear teeth. From the tooth geometry shown in Figure 4.4C, we can write:

$$\frac{p_c}{2} = r_p \alpha \tag{4.2}$$

Substituting p_c from Eq. (4.2) into Eq. (4.1) and reducing gives:

$$\alpha = \frac{\pi}{N} \text{ radians or } \alpha = \frac{180}{N} \text{ degrees} \tag{4.3}$$

The derivation of the involute equation is more complex and is not covered here. We align the involute of one tooth with the XY coordinate system as shown in Figure 4.4D where the lowest point P_b on the involute lies on the Y axis. This orientation does not represent a limitation, but rather simplifies the form of the involute equation, which is therefore given by:

$$\begin{aligned} x &= -r_b(\sin\theta - \theta\cos\theta) \\ y &= r_b(\cos\theta + \theta\sin\theta) \end{aligned} \quad 0 \le \theta \le \theta_{\max} \tag{4.4}$$

Where r_b (the base circle radius) is given by (see Figure 4.4A):

$$r_b = r_p \cos\phi \tag{4.5}$$

and (x, y) are the coordinates of any point P on the involute at an angle θ as shown in Figure 4.4D. The lowest point P_b on the involute corresponds to the value of $\theta = 0$ and lies on the base circle. Point P_a lies on the addendum circle and does not necessarily correspond to the value of $\theta = \theta_{\max}$. We can arbitrarily select a large enough value for $\theta_{\max}$ so that the involute crosses the addendum circle and then trim it to that circle. Therefore, we create the involute profile by generating points on it using Eq. (4.4) and connecting them with a spline curve, or we input Eq. (4.4) into a CAD/CAM system.

The root circle is always less than the base circle. For simplicity, we have the root circle radius, r_r, be 0.98 of the base circle radius (there are other formulas that do not give consistent results). Thus, we write:

$$r_r = 0.98 r_b \tag{4.6}$$

The following steps summarize the calculations we need to create a gear CAD model:

1. The input parameters we need are the pitch circle radius r_p, the pressure angle $\varnothing$, and the gear number of teeth N.
2. Calculate r_b using Eq. (4.5).
3. Calculate r_r using Eq. (4.6).
4. Calculate the gear dedendum $d = r_p - r_b$.
5. Assuming that the addendum and dedendum are equal, calculate the addendum circle radius as $r_a = r_p + a = r_p + d$ (see Figures 4.4C and 4.4D).
6. Use Eq. (4.3) to calculate the tooth angle α.
7. Enter the involute parametric equation given by Eq. (4.4) into a CAD/CAM system to sketch the involute curve as a spline.
8. Create one gear tooth and use sketch circular pattern to pattern it to create all gear teeth.

Example 4.2 Create the CAD model of a spur gear with $r_p = 60$ mm, $\varnothing = 20°$, and $N = 20$.

Solution Using the above calculation steps, we get $r_b = 56.382$ mm, $d = a = 3.618$ mm, $r_a = 63.618$ mm, $r_r = 55.254$ mm, and $\alpha = 9°$. There are two methods to create the tooth involute curve.

In the first method, we use Eq. (4.4) with $\Delta\theta = 5°$. We generate 11 points on the involute, for a $\theta_{max} = 50°$. We generate the points on the involute curve. We then use **Insert** > **Curve** > **Curve Through XYZ Points**. A better method is to input Eq. (4.4) to SolidWorks and let it generate the curve. We need to use radians for the angle θ. We use 1 radian for θ_{max}. This value is arbitrary to ensure that the involute curve intersects and crosses the addendum circle to be able to trim it to the intersection point. Also, SolidWorks uses the parameter t, requiring us to replace θ by t when we input the equation. Figure 4.5 shows the spur gear. We create half a tooth, mirror it to create a full tooth, then circular pattern the full tooth to create all teeth of the gear. Here are the detailed steps.

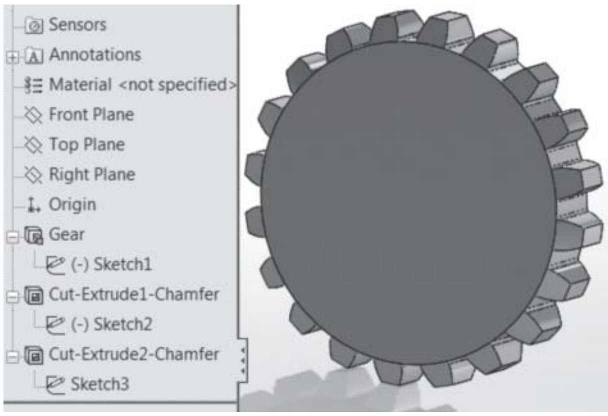
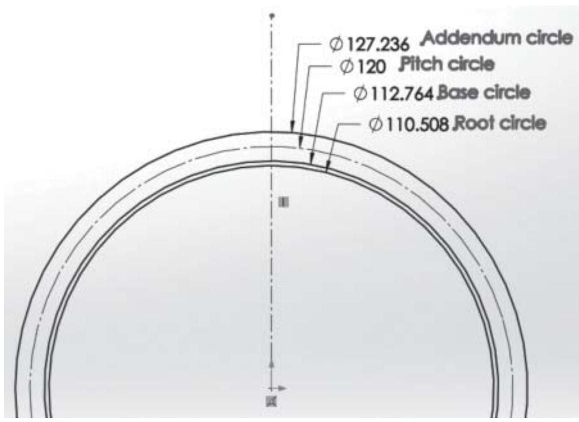


FIGURE 4.5
Spur gear



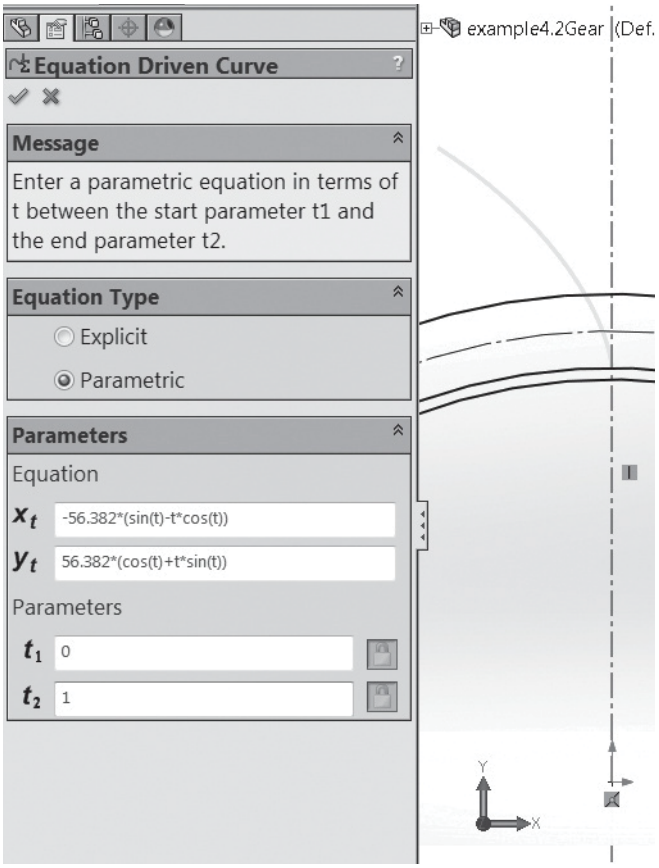
Step 1: Create *Sketch1* circles and axes:

File > **New** > **Part** > **OK** > **Front Plane** > **Circle** on **Sketch** tab > sketch four circles and dimension as shown > **Centerline** on the **Sketch** tab > sketch vertical line > **File** > **Save As** > *example4.2* > **Save.**

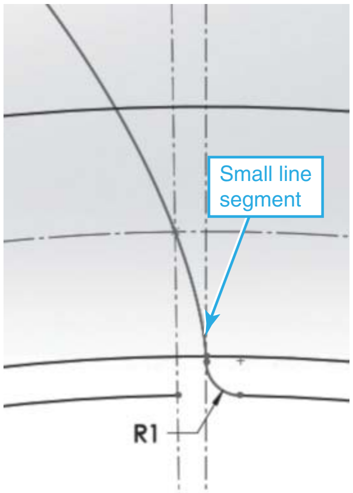
Note: Set the part units to mm before you start. The vertical centerline serves as a validation that the involute bottom endpoint passes through it when we create it in Step 2. Also, you will not close the sketch until you finish Step 5.

Step 2: Create *Sketch1*-tooth involute:

Sketch tab **>** **Spline** dropdown on the **Sketch** tab **>** **Equation Driven Curve** > **Parametric** > enter x and y equations and limits as shown > ✔.

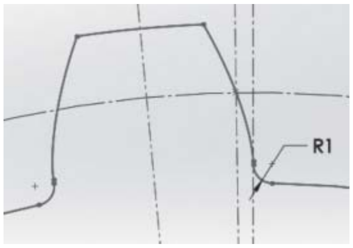


Step 3: Create *Sketch1*-tooth bottom:

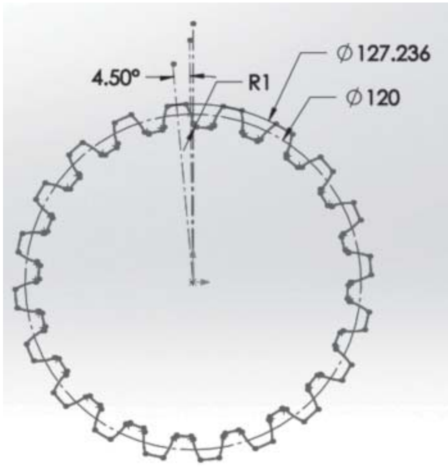


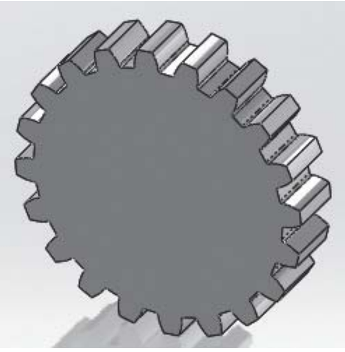
Line on **Sketch** tab **>** sketch a line passing through bottom end of involute curve and crossing the root circle > **Esc** on keyboard > select the line + **Ctrl** on keyboard + involute curve > **Tangent** from **Add Relations** options on left pane > ✔ > **Point** on **Sketch** tab > create a point at intersection of involute and pitch circle (turn relations on: **View** > **Sketch Relations** to see all) > **Center line** on **Sketch** tab > sketch a line passing through origin and crossing involute at any point > **Esc** key > select centerline just created + **Ctrl** + point > **Coincident** from **Add Relations** options on left pane > ✔ > **Trim Entities** on **Sketch** tab > **Trim to closest** > select line below root circle and select root circle between two centerlines > ✔ > **Fillet** on **Sketch** tab > enter 1 mm for radius > select line and root circle > **Yes** to continue > ✔ > select base circle > **Delete** key on keyboard.

Step 4: Create *Sketch1*-tooth other half: **Trim Entities** on **Sketch** tab > **Trim to closest** > select involute top part > **Centerline** on **Sketch** tab > sketch a line passing through origin and to left of involute > **Smart Dimension** on **Sketch** tab > select the centerline just created and the other centerline to the right of it > enter 4.5 > ✔ > **Mirror Entities** on **Sketch** tab > select involute + **Ctrl** key + line segment connected to involute + fillet created in Step 3 > click **Mirror about** box on left of screen > select the far left centerline > ✔ > **Trim Entities** on **Sketch** tab > **Trim to closest** > click addendum circle outside tooth > click root circle inside tooth twice to delete its two segments inside the tooth > ✔.



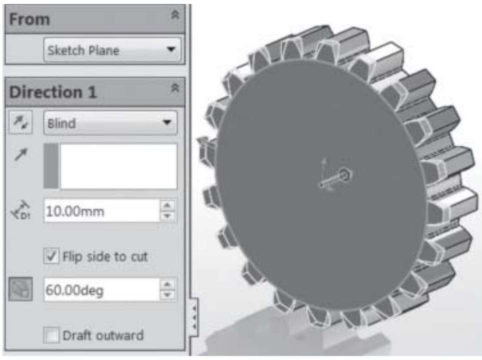
Step 5: Create *Sketch1*-all gear teeth: **Linear Sketch Pattern** dropdown on **Sketch** tab > **Circular Sketch Pattern** > click first box under **Parameters** on left pane **>** select origin to define axis of pattern > click **Entities to Pattern** box **>** select the tooth profile 7 entities **>** enter 20 for the number of instances to create **>** ✔ > the sketch becomes over defined when you pattern the tooth because of the profile mirror of first tooth. Click this sequence to resolve it: **Over Defined** (shown red in status bar) > **Diagnose** > **Accept** > **Trim Entities** > **Trim to closest** > trim all excess form root circle (segments inside teeth) > ✔ > exit sketch.





Step 6: Create *Gear* feature: Select *Sketch1* **> Features** tab **> Extruded Boss/Base** > Enter 25 for thickness (**D1**) > reverse extrusion direction > ✔.

Step 7: Create *Sketch2* and *Cut-Extrude1-Chamfer*: Select *Gear* front face **> Features tab > Extruded Cut** > **Circle** on **Sketch** tab > click origin and snap to teeth root circle > exit sketch > enter 10 for thickness (**D1**) > check **Flip side cut** as shown above > click **Draft** icon as shown above > enter 60 for draft angle > ✔.



Step 8: Create *Sketch3* and *Cut-Extrude2-Chamfer*: Repeat Step 7, but use the back face of *Gear*.

> ***HANDS-ON FOR EXAMPLE 4.2.*** Add a hole and a keyway to the gear blank. Use a diameter of 50 mm for the hole and 10 × 10 mm keyway.

4.4 Design Library and Library Features

Design reuse and using off-the-shelf standard components are important concepts that speed up the design, and thus make it less expensive. The field of mechanical design has many standard parts that designers use every day in their designs (e.g., fasteners [nuts and bolts], gears, bearings). While these parts are universally standard, a company may have some parts that are unique and reusable only in that company's designs. SolidWorks provides the concepts of design library and library features to enable designers to reuse off-the-shelf components.

A **library feature** is a part that you create once and save in a library for reuse in the future. This library is known as the *design library*. You may save a library feature with the *.sldprt* or *.sldlfp* extension. Most of the time library features are inserted into assemblies as components or inserted into new empty (blank) parts. Commonly used library features include holes, slots, and many others. You can use several library features to construct a single part. Not only does this save time, but it also ensures consistency in your CAD models.

Using library features is easy; you drag a library feature from the design library and drop it onto the open part or assembly. SolidWorks asks you if you want to insert a copy or instance (**derived part**) of the part as shown in Figure 4.6. If you select **Yes**, it inserts the copy in the open part. If you answer **No**, it opens a blank part and inserts it there.

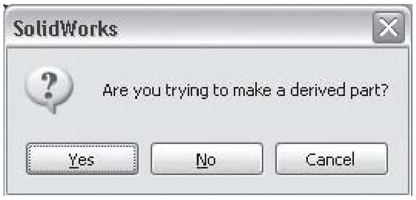
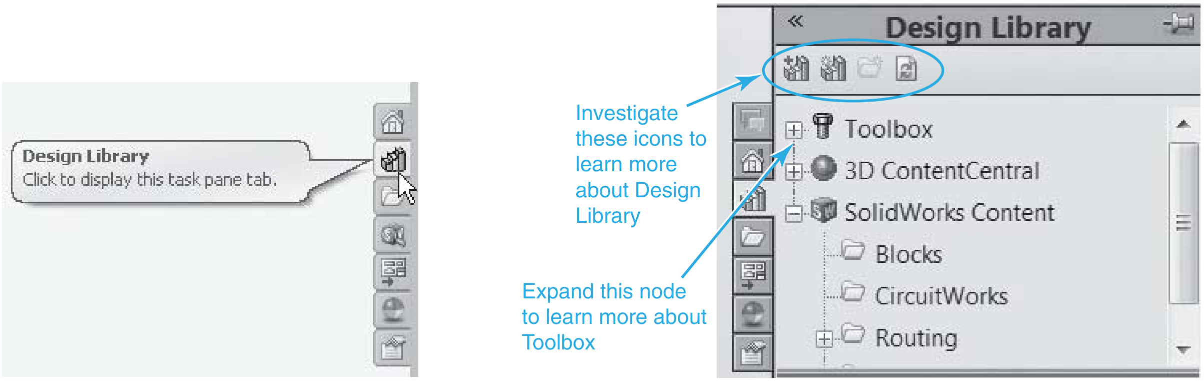


FIGURE 4.6
Using a library feature

You save library features in a design library. You can organize the library into folders. SolidWorks comes with a Design Library. The path to this library folder is *C:\Program Data\SolidWorks\SolidWorks version\design library*. (Make sure that hidden files are visible in Windows to see the *Program Data* folder.) Click the **Design Library** tab in the SolidWorks Task Pane (shown on the right of the screen) as shown in Figure 4.7A, to open the design library. The library is organized into folders as shown in Figure 4.7B. You can also add



(A) Open Design Library

(B) Design Library folders

FIGURE 4.7
SolidWorks Design Library

your new custom folders to the library. You should save your library features into the SolidWorks **Design Library** to have them accessible as shown in Figure 4.7B. If you do not, then you would have to navigate to the folder where you saved them. The most commonly used SolidWorks library is **Toolbox** shown in Figure 4.7B. Expand the Toolbox node and investigate its content.

4.5 Configurations and Design Tables

Family of parts is a natural outcome of the parametrics concept of solid modeling. Defining a solid (part) in a sketch by parameters (dimensions are values for the parameters) enables us to modify the dimensions and create a different-size clone of the part with a click of a button. These clones are what we refer to as family of parts. SolidWorks calls them *configurations*. We can also create clones of assemblies. The clones are created by changing the dimensions of some key parameters of the part or assembly. The clones have the same topology as the original, but different geometry. For example, consider a two-feature part: a base block and a shaft boss. You clone the part into a square block and a skinny long boss, or a rectangle block and short fat boss.

FIGURE 4.8
Design table

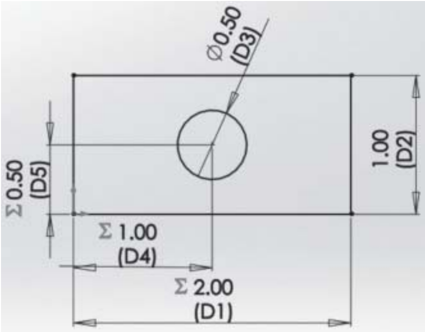
We use design tables to help create and control configurations in three ways: change sizes, change configurations of components, and/or suppress/unsuppress features. SolidWorks uses Microsoft Excel sheet as its design table. You can insert a design table into an open part or assembly by clicking this sequence: **Insert** (menu) **> Tables > Design Table.** This opens the **Design Table** pane under the **PropertyManager** tab on the left pane, as shown in Figure 4.8. You can specify one of the three sources shown to create the design table. When you click the green check mark to finish, SolidWorks acts accordingly. For example, if you select the **Auto-create** (default) option, SolidWorks displays a list of the open part dimensions and asks you to select some to include in the design table. The table is created with the current values of the part dimensions as the default set (configuration). The set shows as a row in the table. You can add other rows with different values for dimensions. Each row is a configuration. The **Design Table** is saved under the **ConfigurationManager** tab.

Example 4.3 Create design tables.

Solution This example builds on Example 2.4. We create a design table where we change the value of parameter (demission name) *D2* to create four configurations of the sketch. The steps are as follows.

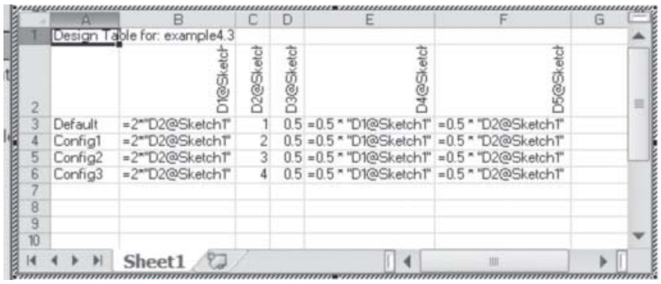
Step 1: Open *example2.4* part:

File > Open > locate and select the file > **Open**.



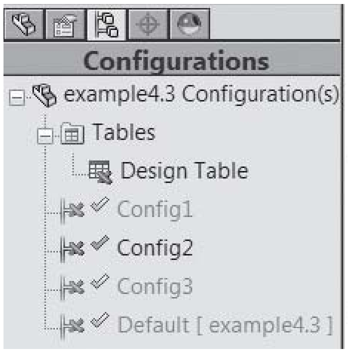
Step 2: Create design table with *Default* config:

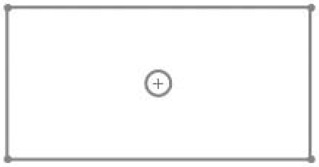
Insert > Tables > Design Table > ✔ > click *D1* (shown here) + **Shift** key on keyboard + *D5* (shown here) to select all dims > **OK.**



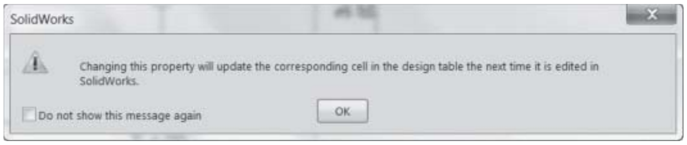
	A	B	C	D	E	F	G
1	Design Table for: example4.3						
2		D1@Sketch1	D2@Sketch1	D3@Sketch1		D4@Sketch1	D5@Sketch1
3	Default	=2*"D2@Sketch1"	1	0.5	=0.5 * "D1@Sketch1"	=0.5 * "D2@Sketch1"	
4	Config1	=2*"D2@Sketch1"	2	0.5	=0.5 * "D1@Sketch1"	=0.5 * "D2@Sketch1"	
5	Config2	=2*"D2@Sketch1"	3	0.5	=0.5 * "D1@Sketch1"	=0.5 * "D2@Sketch1"	
6	Config3	=2*"D2@Sketch1"	4	0.5	=0.5 * "D1@Sketch1"	=0.5 * "D2@Sketch1"	

Step 3: Create three other configs: Step 2 opens the Excel sheet shown > copy the *Default* config row and paste three times > edit the name cell and change config names as shown above > edit the *D2* column and change as shown above > click anywhere on the screen away from Excel sheet to make it disappear > **OK** to accept the popup window.





Step 4: Review the design configs: Step 3 creates a tree with four nodes (as shown above) under the **ConfigurationManager** tab > double click any config to display the corresponding sketch > the current config is displayed in dark black in the tree.



Step 5: Investigate effect of design table on sketch parameters: Click features tree > *Sketch1* > observe that dimensions are displayed in a pink color and each time you click or move one, you get a window (shown above) warning you.

HANDS-ON FOR EXAMPLE 4.3. Edit the sketch and change the value of *D2* to 6. What happens to the design table? Explain the result.

4.6 Macros

Design automation offers two benefits. First, it enhances productivity. Second, it helps with repetitive tasks that are mundane. For example, if we follow the same design process over and over, automating it would be the logical thing to do.

Macros aid in design automation. Macros are also viewed as a way to customize your CAD/CAM system. A **macro** is a short computer program that is used to repeat commonly performed operations. That computer program is generated automatically by a CAD/CAM system in the background during use of the system, from the time we turn the macro on until we turn it off. We can play the macro back after creation over and over, with different input values (e.g., new dimensions). After we create a macro, we can use it for recording, edit it, run it, pause it, stop it, and assign it to a shortcut (hot) key or to a menu item. When we assign a macro to a shortcut key or to a menu item, we can specify which method (function) of the macro to run. Click **Tools** > **Macro** to access the macro menu. SolidWorks saves the macro file in the same part folder and uses *.swp* as the file extension, with default names of *Macro1.swp*, *Macro2.swp*, unless you change them.

A higher level of automation than using macros is to use Visual Basic (VB) or another programming language to perform full automation and have better control of the automation. As a matter of fact, VB is the programming engine behind macros. Instead of writing the macro VB code, the SolidWorks macro interface enables us to generate the code automatically while we perform the design tasks as usual. It is this VB code that we save in a file when we save the macro. We can use the VB editor to edit and tweak the VB code.

If you take a closer look at the macro VB code, it uses what we call the SolidWorks API (Application Programming Interface). The code makes calls to API functions. Think of API as the gateway between the application we want to write and SolidWorks code that has been already written. In other words, the API provides access to SolidWorks geometric engine.

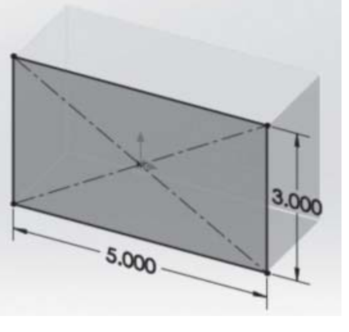
You can learn VB programming by creating multiple macros, study their generated VB code, and expand on it. That is what we call the "brute force" approach. Just keep in mind that VB is an object-oriented programming (OOP) language that requires knowledge and understanding of the object-oriented design and how objects are defined and implemented.

The programming approach could be useful to create an entire assembly from a few parameters. We can write a program to define some variables, storing them in a row in a design table. Each row represents a new version of the assembly. We delete all rows from the table and keep the last row, which is the new assembly.

Example 4.4 Develop a macro to create an extrusion.

Solution This example illustrates creating and using macros. We create a block extrusion and record the steps of creating it in a macro.

Step 1: Turn macro on and create *Sketch1* and *Block* feature:



File > **New** > **Part** > **OK** > **Tools** > **Macro** > **Record** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Sketch** tab > **Center Rectangle** on **Sketch** tab > sketch a rectangle and dimension as shown > exit sketch > enter 2 for thickness (**D1**) > reverse extrusion direction > ✔ > **File** > **Save As** > *example4.4* > **Save.**

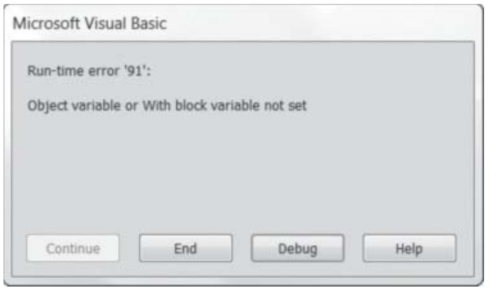
Note: **Macro** > **Record** opens the **Macro** recording window shown. Hover over its buttons to read them.

Step 2: Stop and save macro:

Tools > **Macro** > **Stop** (or click black square on **Macro** window shown in Step 1) > type *BlockShaft* for macro name in the **Save As** window that opens > **Save**.

Note: full macro file name is *BlockShaft.swp*. The default folder for the file is the part file folder from Step 1.

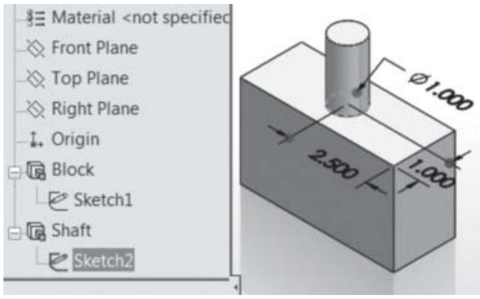
Step 3: Run and debug the macro: **Tools** > **Macro** > **Run** > select *BlockShaft.swp* > **Open.** The macro will generate the error shown here as soon as it displays the dimension box and you click ✔ or hit **Enter**.



We do not know why this error occurs but we know how to fix it. Hit the **Debug** button shown above to open the macro code in the VB editor. The editor highlights the line of code that causes the error in yellow. Comment every yellow statement and save the macro file. Figure 4.9 shows the entire macro VB code and the fix as indicated by two arrows.

Step 4: Run the macro again after the fix: Open a new part or delete the feature from the current part > **Tools** > **Macro** > **Run** > select *BlockShaft.swp* **> Open** > enter 8 for rectangle width when dimension box appears > enter 6 for rectangle height when dimension box appears.

HANDS-ON FOR EXAMPLE 4.4. Modify the part as shown to the right and re-create the macro. Rerun the macro twice to investigate locating the shaft to the left or to the right side from the center. This simulates the idea of using a macro to investigate "what if" design scenarios such as sizes or feature locations.



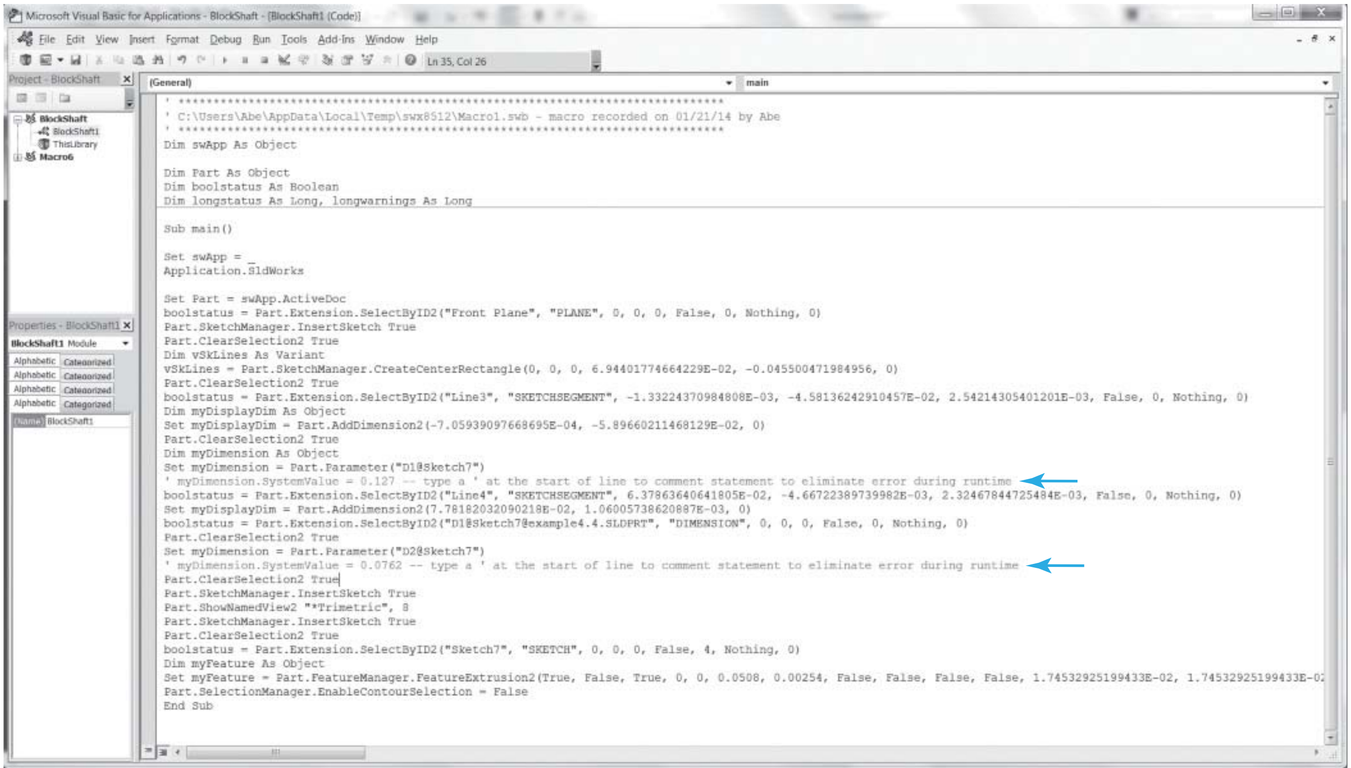


FIGURE 4.9
VB Editor Window

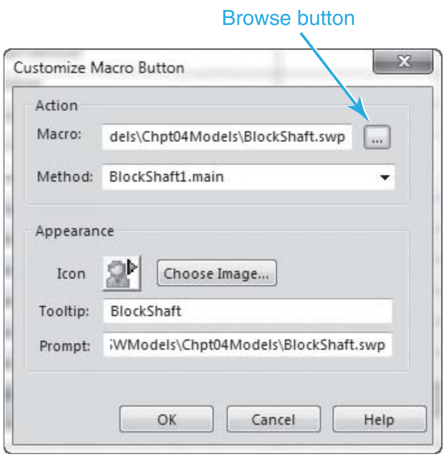
Example 4.5 Create a hot key for a macro.

Solution This example assigns the macro of Example 4.4 to a hot key on the keyboard, namely **Shift** + **X**. When the user hits this combination on the keyboard, the macro runs.

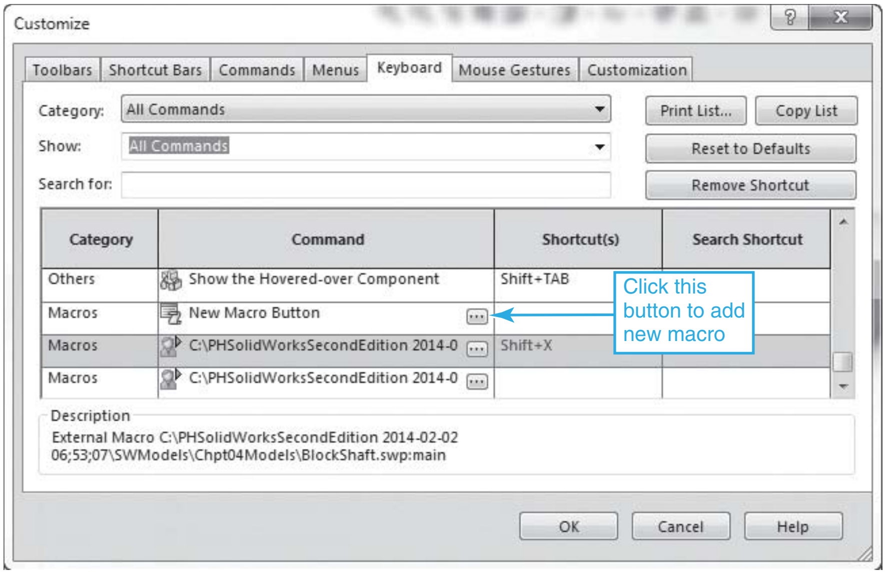
Step 1: Create macro hot key:

File > **Open** > locate *example4.4* > **Open.**

Step 2: Locate macro file: **Tools** > **Customize** > **Keyboard** tab > scroll to bottom and locate **Macros** row > click button in this row to open window shown > click the browse button > locate macro file (*.swp* extension) > **Open** > **OK.**



Step 3: Create hot key: Click **Shortcut(s)** column in **Macros** row as shown > **Shift** key + **X** key to add shortcut as shown below > **OK.**



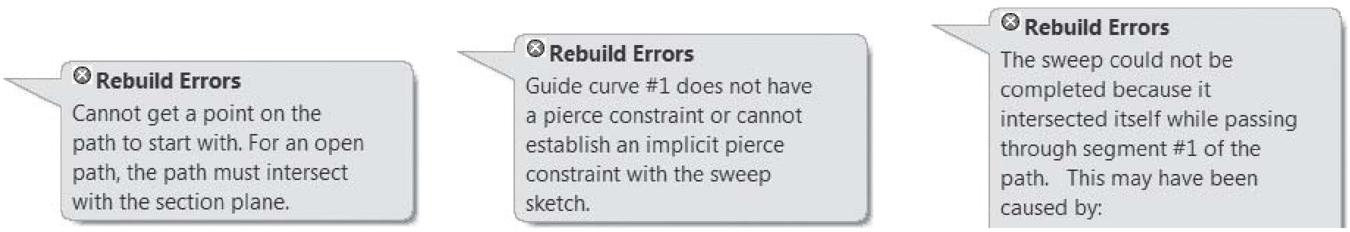
Step 4: Use hot key to run macro: Hit **Shift** + **X** keys in an open part.

4.7 Tutorials

Tutorial 4–1: Create Sweep Features

A sweep feature requires, at minimum, a profile (cross section) to sweep and a path (curve) to sweep along. We can use a guide curve to control the sweep further. If we do not use a guide curve, the sweep cross section stays constant.

Sweep operations may fail for different reasons. Figure 4.10 shows three of them. As a general rule, the sweep path and guide must intersect the cross section plane, and the cross section must not intersect itself as it traverses the path and/or the guide curve.

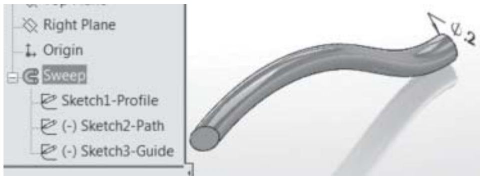


(A) Path does not intersect CS plane. (B) Guide curve does not intersect CS plane. (C) CS intersects itself.

FIGURE 4.10
Some possible sweep operation errors

Create the sweep features shown in Figure 4.11. All dimensions are in inches.

(A) No guide curve

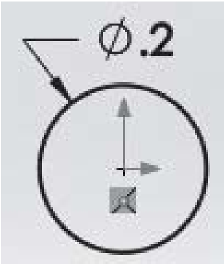


(B) With guide curve

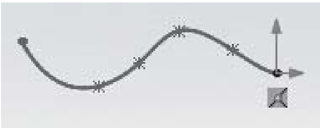
FIGURE 4.11
Sweep features

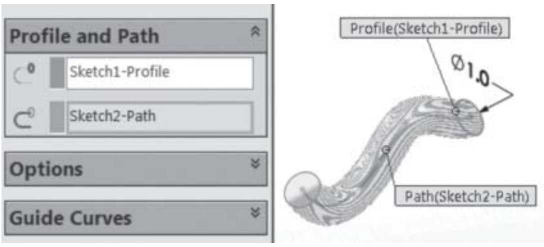
No-guide-curve sweep (Figure 4.11A) modeling steps:

Step 1: Create *Sketch1-Profile*: **File > New > Part > OK > Front Plane > Circle** on **Sketch** tab > click origin and sketch circle and dimension as shown > exit sketch > **File > Save As** > *tutorial4.1A* > **Save.**



Step 2: Create *Sketch2-Path*: **Right Plane > Spline** on **Sketch** tab > sketch spline as shown; make sure spline snaps to origin > exit sketch.

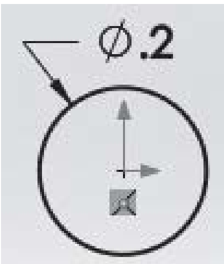




Step 3: Create *Sweep1* feature: **Sweep Boss/Base** on **Features** tab > select circle sketch as profile as shown above > select spline sketch as path as shown above > ✔.

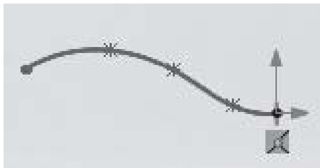
With-guide-curve sweep (Figure 4.11B) modeling steps:

Step 1: Create *Sketch1-Profile*: **File > New > Part > OK > Front Plane > Circle** on **Sketch** tab > click origin and sketch circle and dimension as shown > exit sketch > **File > Save As** > *tutorial4.1B* > **Save.**

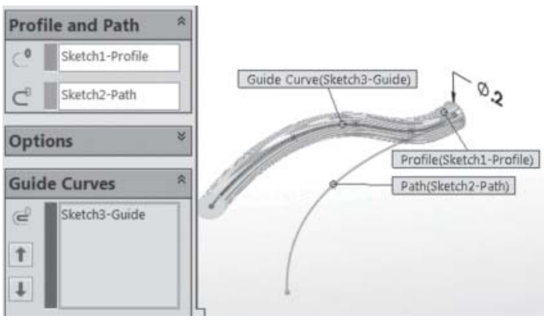


Step 2: Create *Sketch2-Path*: **Top Plane > Spline** on **Sketch** tab > sketch spline as shown; make sure spline snaps to origin > exit sketch.

Step 3: Create *Sketch3-Guide*: **Right Plane > Spline** on **Sketch** tab > sketch spline as shown; make sure spline snaps to origin > exit sketch.



Step 4: Create *Sweep* feature: **Sweep Boss/Base** on **Features** tab > select circle sketch as profile as shown > select **Step 2** spline sketch as path > select **Step 3** spline sketch as **Guide** > ✔.



HANDS-ON FOR TUTORIAL 4–1. Change the cross sections of both sweeps to a 2 × 2 inch square. What happens? Resolve the error(s) and explain your solution.

Tutorial 4–2: Create Loft Features

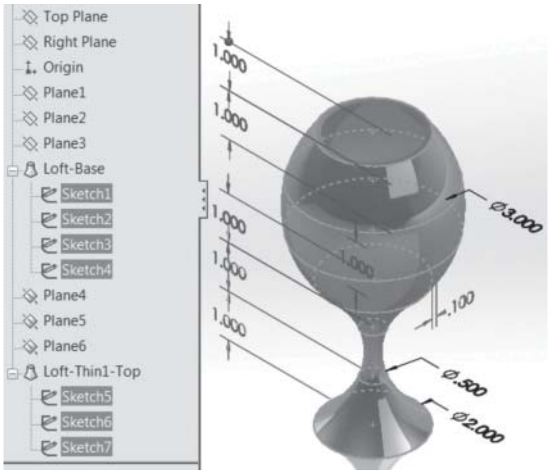
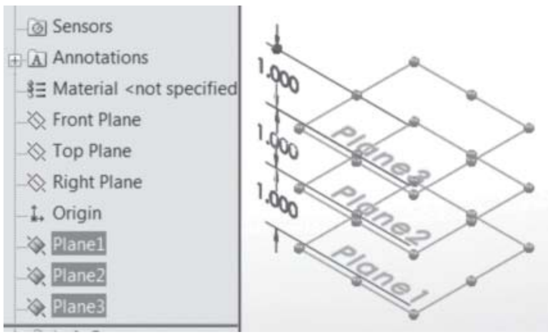


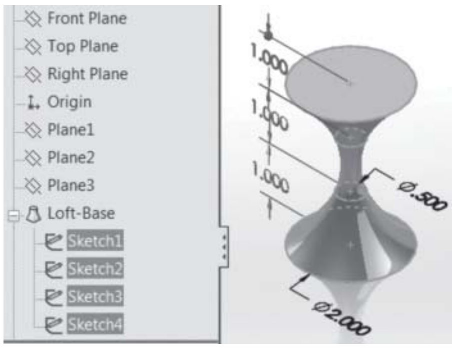
FIGURE 4.12
Loft feature

Create the wine glass shown in Figure 4.12. All dimensions are in inches. We introduce the concept of **Convert Entities** on the **Sketch** tab in this tutorial. We copy one circle in one sketch to another sketch. This concept enables us to copy entities from one sketch to another. While we could have easily created a new circle and dimensioned it, the **Convert Entities** method is faster (no need to sketch a circle and dimension it). SolidWorks creates an **On Edge** relation between the two circles and shows a small green cube on the copied entity to indicate the relationship. When you click the copied circle while editing the sketch, SolidWorks displays the **On Edge** relation in the relations pane to the left of the screen.

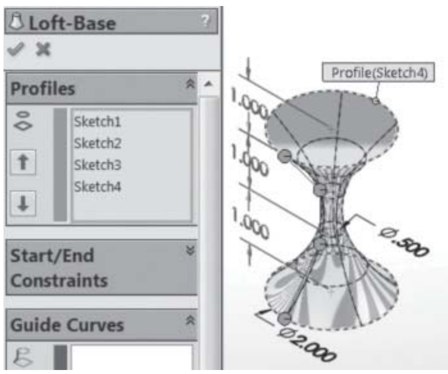
Step 1: Create *Plane1–Plane3*: **File** > **New** > **Part** > **OK** > **Reference Geometry** on **Features** tab > **Plane** > expand features tree and select **Top Plane** > enter 1.0 for distance > ✔ > repeat for *Plane2* and *Plane3*, but select the previously created plane > **File** > **Save As** > *tutorial4.2* > **Save**.



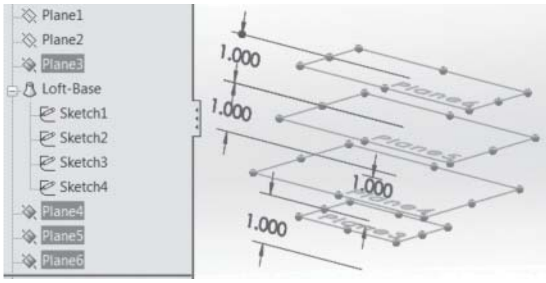
Step 2: Create *Sketch1–Sketch4*: **Top Plane** > **Circle** on **Sketch** tab > click origin and drag to sketch and dimension a 2.0 inch diameter circle > exit sketch > select *Plane1* as sketch plane, and create a 0.5 inch diameter circle > exit sketch > select *Plane2* as sketch plane > **Convert Entities** on **Sketch** tab > click circle on *Plane1* > ✔ > ✔ > exit sketch > select *Plane3* as sketch plane > **Convert Entities** on **Sketch** tab > click circle on **Top Plane** > ✔ > ✔ > exit sketch.



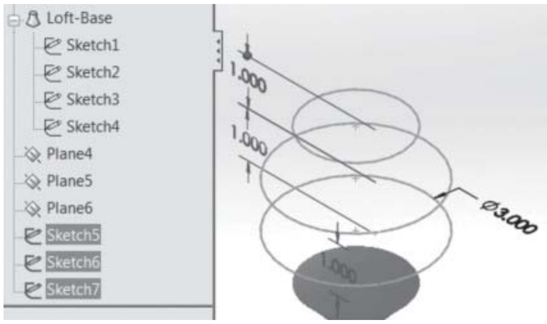
Step 3: Create *Loft-Base* feature: **Lofted Boss/Base** on **Features** tab > select *Sketch1–Sketch4* > ✔.



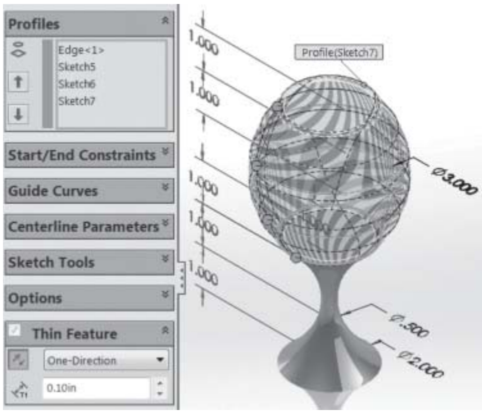
Step 4: Create *Plane4–Plane6*: **Reference Geometry** on **Features** tab **>** **Plane** **>** expand features tree and select *Plane3* > enter 1.0 for distance > ✔ > repeat for *Plane5* and *Plane6*, but select the previously created plane.



Step 5: Create *Sketch5–Sketch7*: Select *Plane4* as sketch plane **>** **Circle** on **Sketch** tab > click origin and drag to sketch and dimension a 3.0 inch diameter circle > exit sketch > select *Plane5* as sketch plane > **Convert Entities** on **Sketch** tab > click *Sketch5* just created > ✔ > ✔ > exit sketch > select *Plane6* as sketch plane > **Convert Entities** on **Sketch** tab > click circle on *Plane3* > ✔ > ✔ > exit sketch.



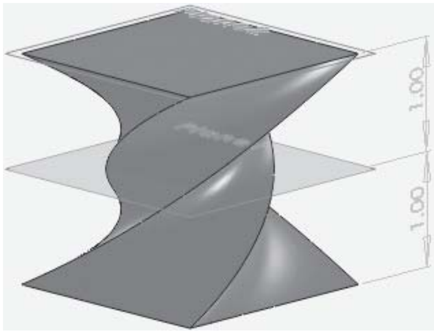
Step 6: Create *Loft-Thin1-Top* feature: **Lofted Boss/Base** on **Features** tab > select circle of *Sketch4*, then select *Sketch5–Sketch7* > check **Thin Feature** box > enter 0.1 for **T1** > if needed, click direction box to toggle direction of thickness > ✔.



Note: Make sure you select the circle and sketches in a way so that the interpolation points (green circles shown here) line up; otherwise, you twist the loft.

Note: The thickness of the thin feature has a direction: inside or outside the profile. Reverse the direction of the double arrows shown to toggle.

HANDS ON FOR TUTORIAL 4–2. Created a loft using three squares of different sizes as cross sections separated by one inch. The square sizes are 2 × 2, 1 × 1, and 2 × 2 respectively. Create the loft connecting the three sections such that the loft is twisted incorrectly as shown to the right.



Tutorial 4–3: Use the Hole Wizard

The hole wizard provides two advantages. First, it helps us create standard hole sizes and types, so off-the-shelf bolts will fit perfectly in the holes. Second, it speeds up creating these holes greatly. This tutorial shows how to create sample holes: counterbore, countersink, and tapped holes. Figure 4.13 shows these holes. We create an extrusion and add holes to it. We also create *Sketch2* on the top face of the *Block* feature with a center rectangle (construction rectangle) that we use to place holes at its corners.

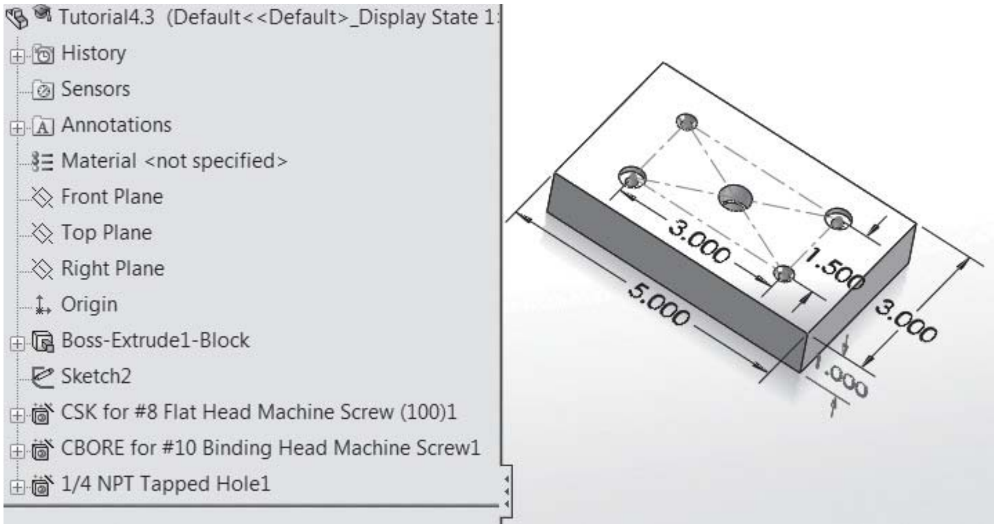
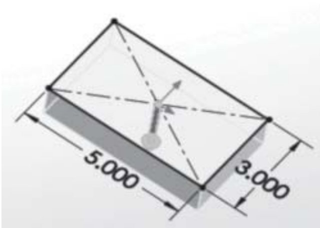


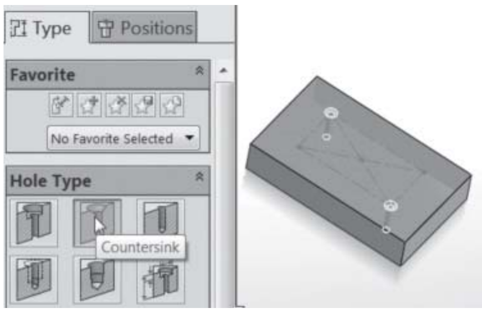
FIGURE 4.13
Wizard holes

Step 1: Create *Sketch1* and *Block* feature: **File** > **New** > **Part** > **OK** > **Top Plane** > **Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and sketch and dimension rectangle as shown > exit sketch > enter 1 for thickness **D1** > reverse extrusion direction > **File** > **Save As** > *tutorial4.3* > **Save**.

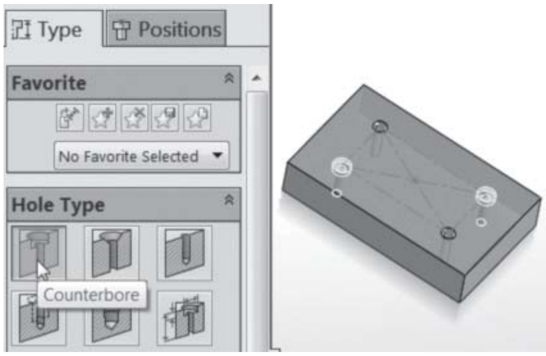


Step 2: Create *Sketch2*: top face of *Block* > **Center Rectangle** on **Sketch** tab > click origin and sketch a 3.0 × 1.5 rectangle as shown in Figure 4.13 > click **For Construction** box > exit sketch.

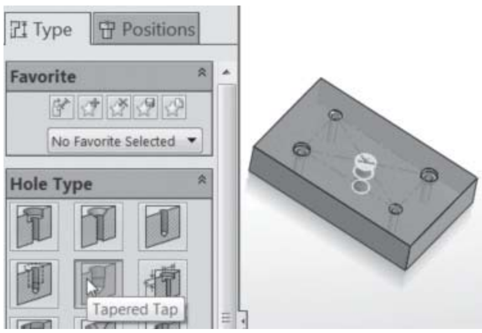
Step 3: Create two diagonal countersink hole features (*CSK…* node in features tree): **Hole Wizard** on **Features** tab > select **Countersink** under **Hole Type** (hover over types until you read it) > select #8 for **Size** under **Hole Specifications** > **Positions** tab > click top face of *Block*, then click two corners of construction rectangles as shown > ✔.



Step 4: Create two diagonal counterbore hole features (*CBORE…* node in features tree): **Hole Wizard** on **Features** tab > select **Counterbore** under **Hole Type** (hover over types until you read it) > select #10 for **Size** under **Hole Specifications** > **Positions** tab > click top face of *Block*, then click two other corners of construction rectangles as shown > ✔.



Step 5: Create *Tapped Hole1* feature: **Hole Wizard** on **Features** tab > select **Tapered Tap** (hover over types until you read it) > select 1/4 for **Size** under **Hole Specifications** > **Positions** tab > click top face of *Block*, then click rectangle center as shown > ✔.



HANDS-ON FOR TUTORIAL 4–3. Create a 1/16 tapered pipe tap through all located 0.5 from the top edge and 1.5 from the left edge of the block.

Tutorial 4–4: Create Compression Spring

Figure 4.14 shows the constant length compression spring we create in this tutorial.

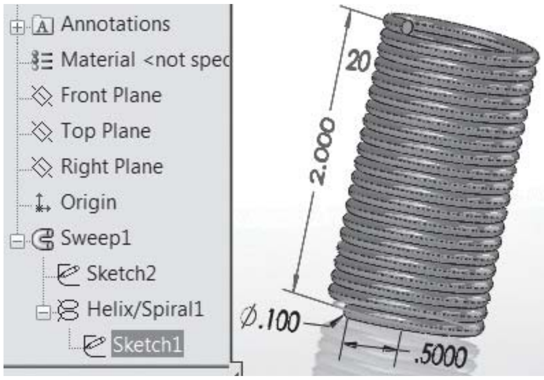
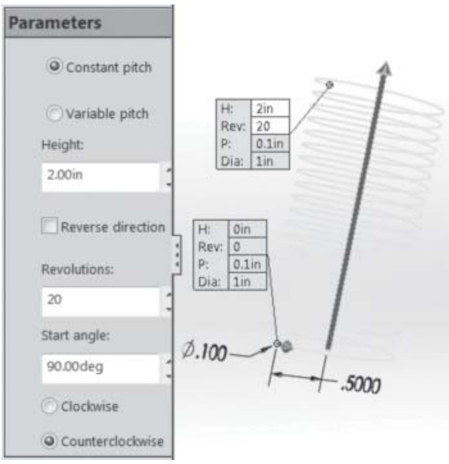
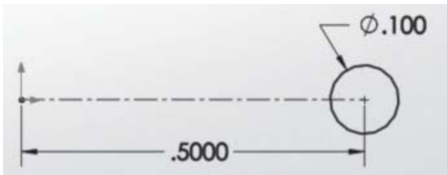


FIGURE 4.14
Compression spring

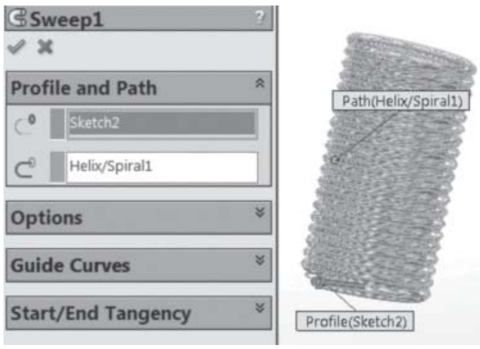
Step 1: Create *Sketch1* and *Helix/Spiral1* curve: **File** > **New** > **Part** > **OK** > **Top Plane** > **Circle** on **Sketch** tab > click origin and sketch and dimension circle with 1.0 inch diameter > exit sketch > **Insert** (menu) > **Curve** > **Helix/Spiral** > enter 2.0 for **Height** and 20 for **Revolutions** as shown > ✔ > **File** > **Save As** > *tutorial4.4* > **Save**.



Step 2: Create *Sketch2*: **Front Plane** > **Circle** on **Sketch** tab > sketch and dimension circle as shown (align center with X-axis as shown) > exit sketch.



Step 3: Create *Sweep1* feature (spring): **Swept Boss/Base** on **Features** tab > select *Sketch2* as **Profile** and *Helix/Spiral1* as **Path** > ✔.



HANDS-ON FOR TUTORIAL 4–4. Edit the spring helix to have a variable pitch. Use a pitch of 0.2 at midheight point.

Tutorial 4–5: Create Spiral

Figure 4.15 shows the spiral spring we create in this tutorial.

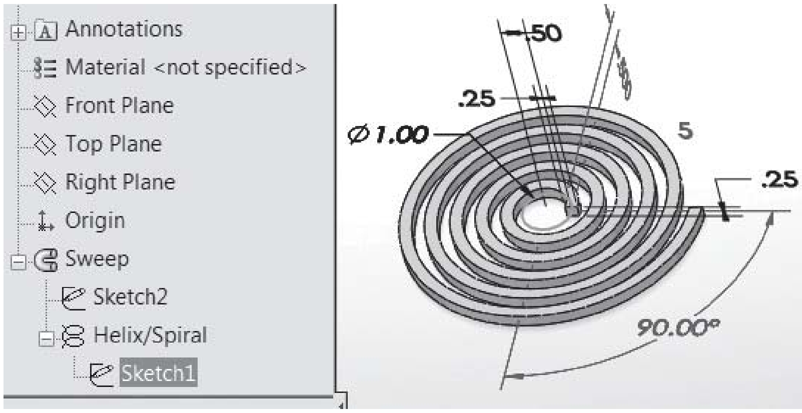
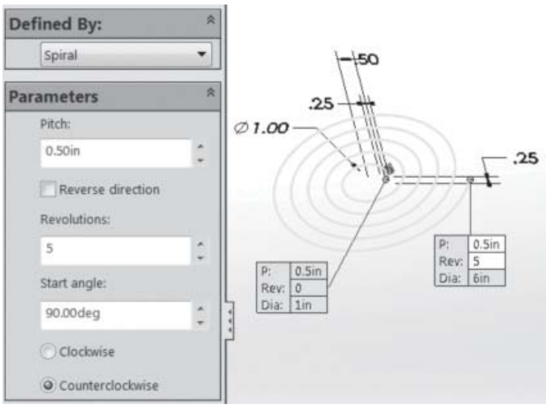
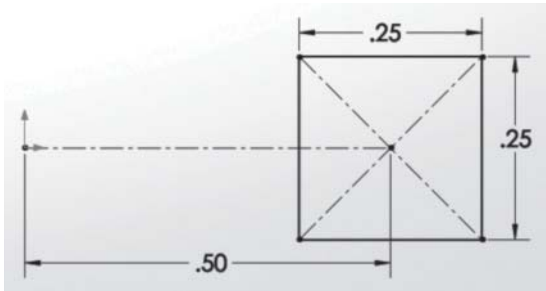


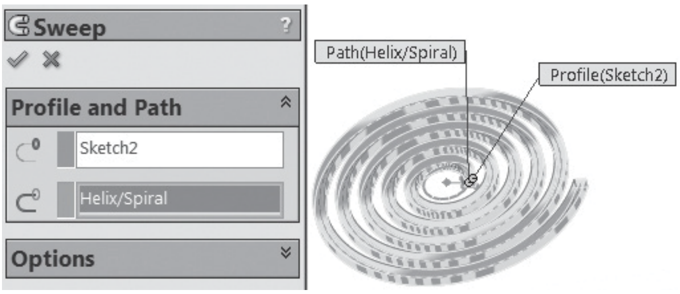
FIGURE 4.15
Spiral spring

Step 1: Create *Sketch1* and *Helix/Spiral* curve: **File** > **New** > **Part** > **OK** > **Top Plane** > **Circle** on the **Sketch** tab > click origin and sketch and dimension circle with 1.0 inch diameter > exit sketch > **Insert** (menu) > **Curve** > **Helix/Spiral** > **Spiral** from **Defined By** dropdown shown > enter 0.5 for **Pitch** and 5 for **Revolutions** as shown > ✔ > **File** > **Save As** > *tutorial4.5* > **Save**.



Step 2: Create *Sketch2*: **Front Plane** > **Central Rectangle** on **Sketch** tab > sketch and dimension rectangle as shown (align center with X-axis as shown) > exit sketch.





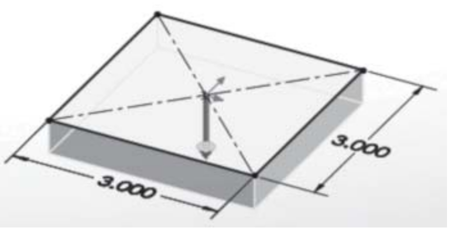
Step 3: Create *Sweep* feature (spiral): **Swept Boss/Base** on **Features** tab > select *Sketch2* as **Profile** and *Helix/Spiral* as **Path** > ✔.

HANDS-ON FOR TUTORIAL 4–5. Change the spiral cross section to a circle with 2.0″ diameter. Can you generate the spiral? Why or why not? Explain.

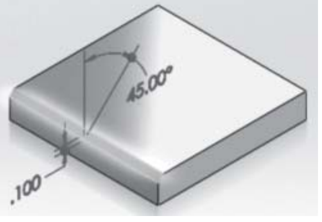
Tutorial 4–6: Create Features

This tutorial covers the creation of these features: chamfer, fillet, slot, shell, draft, and rib. All dimensions are inches. Here are useful observations:

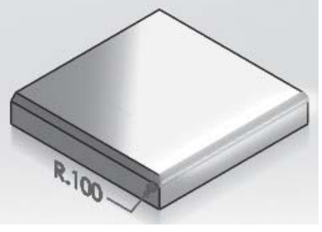
1. Make sure to pay attention to the visual clues shown in the left pane while creating these features.
2. For example, the box symbol under **Chamfer parameters** indicates that you can chamfer a face, an edge, or a vertex (corner point). As expected, chamfering a face chamfers all its edges. Chamfering a corner chamfers the three edges that meet there.
3. A rib requires a profile sketch (e.g., a line or stepwise line) and a thickness.



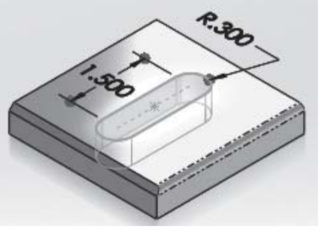
Step 1: Create *Sketch1* and *Block* feature: **File** > **New** > **Part** > **OK** > **Top Plane** > **Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > reverse extrusion direction > enter 0.5 for thickness **D1** > ✔ > **File** > **Save As** > *tutorial4.6* > **Save.**



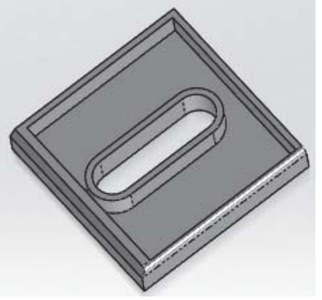
Step 2: Chamfer an edge of *Block* feature: **Fillet** dropdown on **Features** tab > **Chamfer** > select *Block* edge shown > use 0.1 for **D** and 45 degrees for angle > ✔.



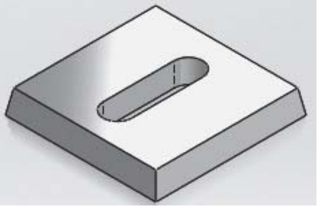
Step 3: Fillet an edge of *Block* feature: **Fillet** on **Features** tab > select *Block* edge shown > use 0.1 for **fillet radius** > ✔.



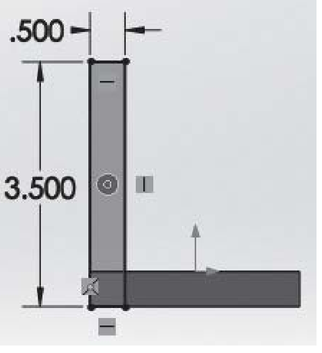
Step 4: Create a straight slot in *Block* feature: Select *Block* top face as a sketch plane > **Extruded Cut** on **Features** tab > **Straight Slot** on **Sketch** tab > sketch and dimension slot as shown > exit sketch > **Through All** > ✔.

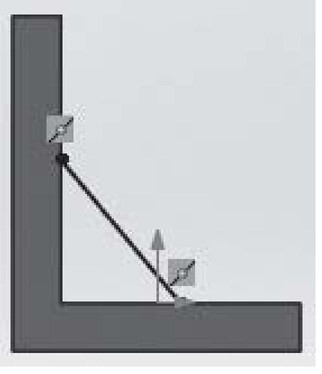


Step 5: Shell *Block* feature: **Shell** on **Features** tab > select top face of *Block* > enter 0.1 for wall thickness **D1** > ✔.



Step 6: Draft *Block* feature: Delete the chamfer, fillet, and shell features > **Draft** on **Features** tab > enter 10 degrees for **Draft Angle** > select top face of *Block* as **Neutral Plane** > select *Block* four side faces to draft > ✔.





Step 7: Create a rib feature: Delete the slot and draft features > select front face of *Block* > **Extruded Boss/ Base** on **Features** tab > **Rectangle** on **Sketch** tab > sketch and dimension rectangle as shown above > exit sketch > reverse extrusion direction > enter 3.0 for thickness **D1** > ✔ > **Front Plane** > **Rib** on **Features** tab > **Line** on **Sketch** tab > sketch a line using the midpoints of the two edges as shown above > exit sketch > enter 0.5 for thickness **T1** > ✔.

HANDS-ON FOR TUTORIAL 4–6. Create the following features:

1. Distance-distance chamfer
2. Vertex chamfer
3. Variable size fillet
4. Face fillet
5. Full round fillet
6. 3-point arc slot
7. 3-stepped rib using 3-stepped line as the rib profile

Tutorial 4–7: Use the Smart Fasteners Wizard

We use the Smart Fasteners wizard to insert the correct fastener based on the hole we select. It is a form of automation using off-the-shelf components. SolidWorks has its own standard library of fasteners. We can only use the wizard at the assembly level. We need to activate the wizard in order for the **Smart Fasteners** icon on the **Assembly** tab to work. If we click the icon before activation, we get this error: **Smart Fasteners requires SolidWorks Toolbox, which is not present.**

In this tutorial, we create an assembly of a block and plate. We create a counterbore hole in the plate and a blind hole in the block, assemble them, and fasten them together with a smart fastener. Figure 4.16 shows the assembly and its tree. All dimensions are in inches.

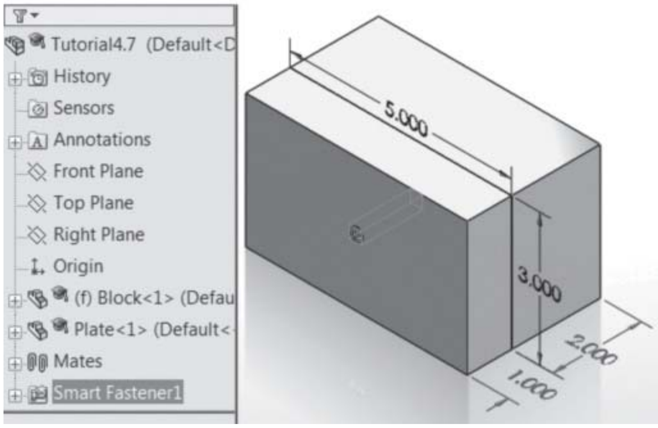
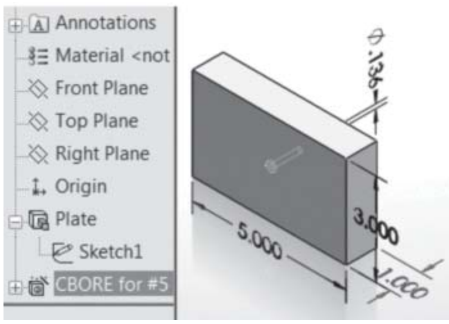


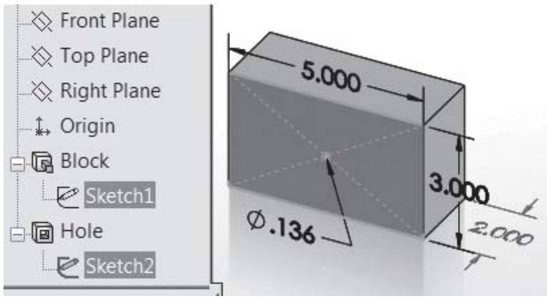
FIGURE 4.16
Assembly using smart fastener

Step 1: Create *Plate* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and sketch and dimension rectangle as shown > exit sketch > enter 1 for thickness **D1** > reverse extrusion direction > ✔ > **Hole Wizard** on **Features** tab > select **Counterbore** under **Hole Type** (hover over types until you read it) > select #5 for **Size** under **Hole Specifications** > **Positions** tab > click front face of *Plate*, then click origin > ✔ > **File** > **Save As** > *Plate* > **Save.**



Note: The diameter of the counterbore hole shown above corresponds to #5 size. We need it to create the corresponding hole in the block in Step 2.

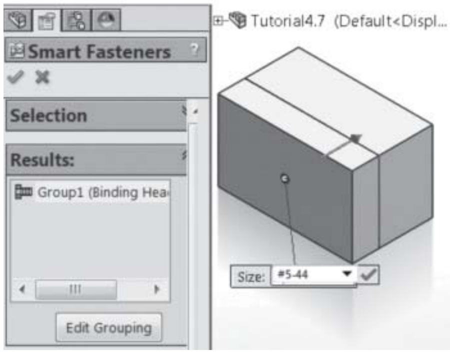
Step 2: Create *Block* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and sketch and dimension rectangle as shown > exit sketch > enter 2 for thickness **D1** > reverse extrusion direction > ✔ > front face of *Block* > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > enter 1 for thickness **D1** > ✔ > **File** > **Save As** > *Block* > **Save**.



Step 3: Create assembly: **File** > **New** > **Assembly** > **OK** > **Browse** > locate *Block* and *Plate* parts > select **Block** + **Ctrl** + **Plate** > **Open** > click ✔ to place *Block* instance in assembly origin > click anywhere in graphics pane to place *Plate* instance > **Mate** on **Assembly** tab > **Coincident** > select the corresponding top edges of *Block* and *Plate* > ✔ > select the corresponding right edges of *Block* and *Plate* > ✔ > ✔.

Step 4: Activate Smart Fasteners wizard: **Tools** > **Add-Ins** > **SolidWorks Toolbox Browser** > **OK**. This adds the **Toolbox** menu to the menu bar to the right of the **Tools** menu. You may deactivate the **Toolbox** by using the same sequence but unchecking the **Toolbox Browser** from the **Add-Ins** window.

Step 5: Add a fastener: **Smart Fasteners** on **Assembly** tab > **OK** (to accept that it may take extra time) > expand features tree > expand *Plate* instance tree node > select *CBORE for #5* node > **Add** > ✔.



HANDS-ON FOR TUTORIAL 4–7. Modify *Block* and *Plate* parts to create four corner countersink holes. Re-create the assembly and use four smart fasteners.

Tutorial 4–8: Create a Bolt

Bolts, like gears, are an important and essential mechanical element. While bolts are standard off-the-shelf components, this tutorial shows how to create the CAD model of one due to its learning value. Figure 4.17 shows the bolt and its features tree. All dimensions are in inches.

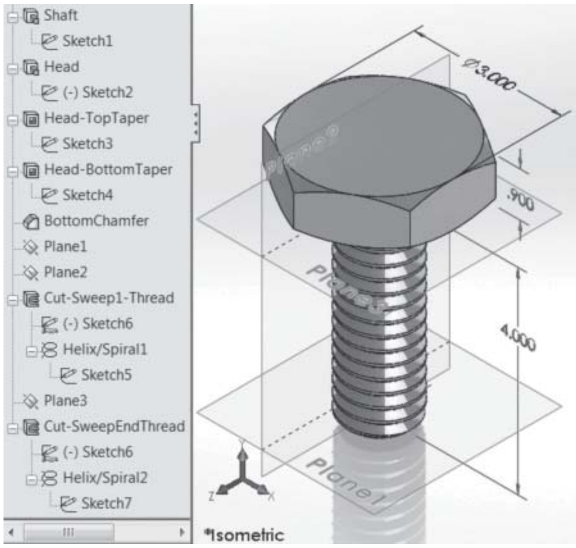
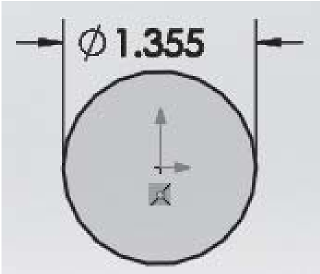
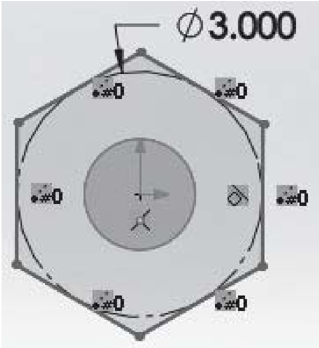


FIGURE 4.17
A bolt

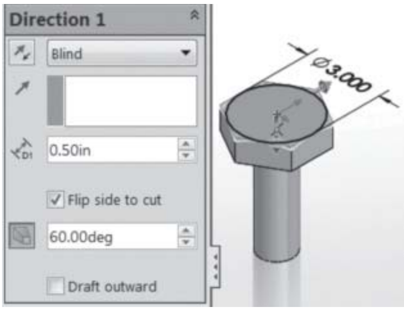
Step 1: Create *Sketch1* and *Shaft* feature: **File** > **New** > **Part** > **OK** > **Top Plane** > **Extruded Boss/ Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > enter 4 for thickness **D1** > reverse extrusion direction > ✔ > **File** > **Save As** > *Bolt* > **Save**.



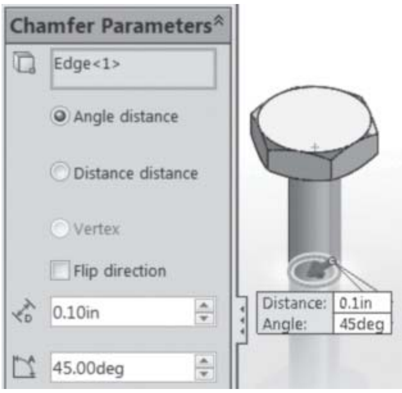
Step 2: Create *Sketch2* and *Head* feature: Top face of *Shaft* feature > **Extruded Boss/Base** on **Features** tab > **Polygon** (hexagon icon) on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > enter 0.9 for thickness **D1** > ✔.



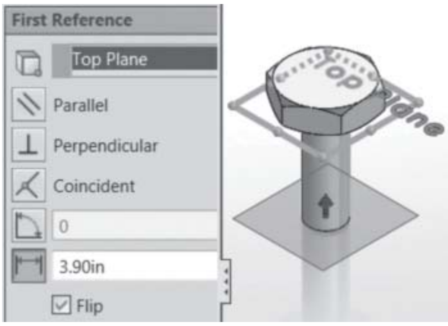
Step 3: Create *Sketch3* and *Head-TopTaper* feature: Top face of *Head* feature > **Extruded Cut** > **Circle** on **Sketch** tab > click origin and sketch (make circle tangent to hexagon sides) and dimension as shown > exit sketch > enter 0.5 for thickness **D1** > click checkbox as shown > enter 60 for angle > ✔ > repeat to create *Head-BottomTaper* to chamfer the bottom of the head.



Step 4: Create *BottomChamfer* feature: **Fillet** dropdown on **Features** tab > **Chamfer** > select bottom edge of *Shaft* feature > enter 0.1 for radius **D** > enter 45 for angle > ✔.

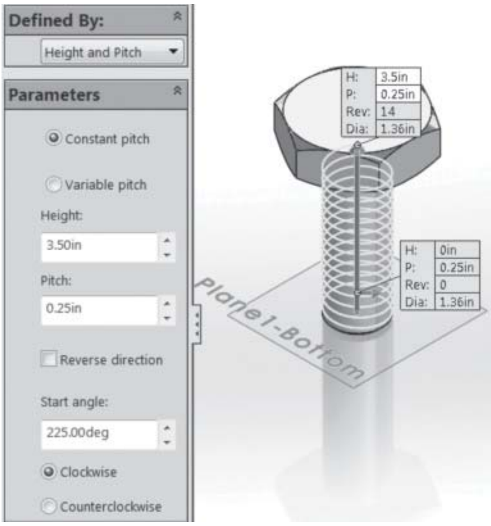


Step 5: Create *Plane1*: **Reference Geometry** on **Features** tab > **Plane** > expand features tree > select **Top Plane** > enter 3.9 for distance as shown above > click **Flip** checkbox > ✔.



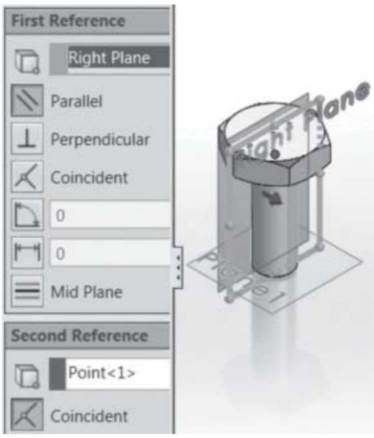
Note: We use a distance of 3.9 for *Plane1*, not 4.0, because the chamfer is 0.1 high.

Step 6: Create *Sketch5* and *Helix/ Spiral1*: select *Plane1* > **Convert Entities** on **Sketch** tab > expand features tree > select *Sketch1* > ✔ > select *Sketch5* > **Insert** > **Curve** > **Helix/Spiral** > enter 3.5 for Height > ✔

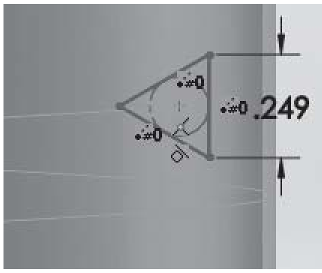


Note: The helix 3.5 height is arbitrary. That leaves 0.4 (out of 3.9). We use 0.25 for the other end thread helix and 0.15 underneath the bottom of bolt head as a length with no threads.

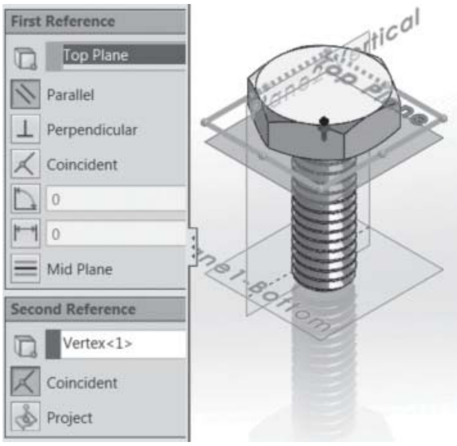
Step 7: Create *Plane2*: **Reference Geometry** on **Features** tab > **Plane** > expand features tree > select > **Right Plane Parallel** > click **Second Reference** box > select top endpoint of *Helix/ Spiral1* > ✔.



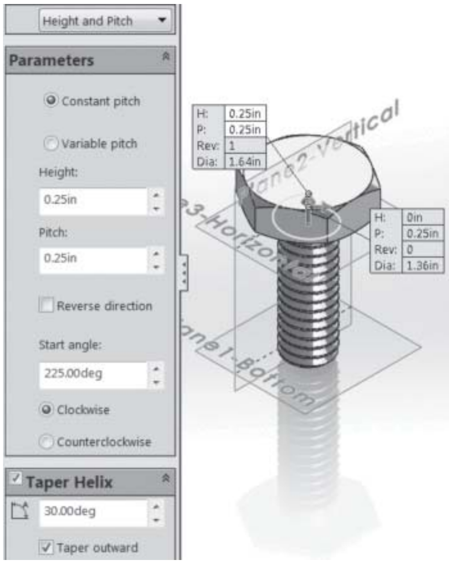
Step 8: Create *Sketch6* and *Cut-Sweep1-Thread* feature: *Plane2* > **Swept Cut** on **Features** tab > **Polygon** (hexagon icon) on **Sketch** tab > click near end of helix and sketch and dimension as shown > exit sketch > select *Sketch6* as profile and *Helix/Spiral1* as path > ✔.



Step 9: Create *Plane3*: **Reference Geometry** on **Features** tab **> Plane > expand** features tree > select > **Top Plane Parallel** > click **Second Reference** box > select top endpoint of *Helix/Spiral1* > ✔.



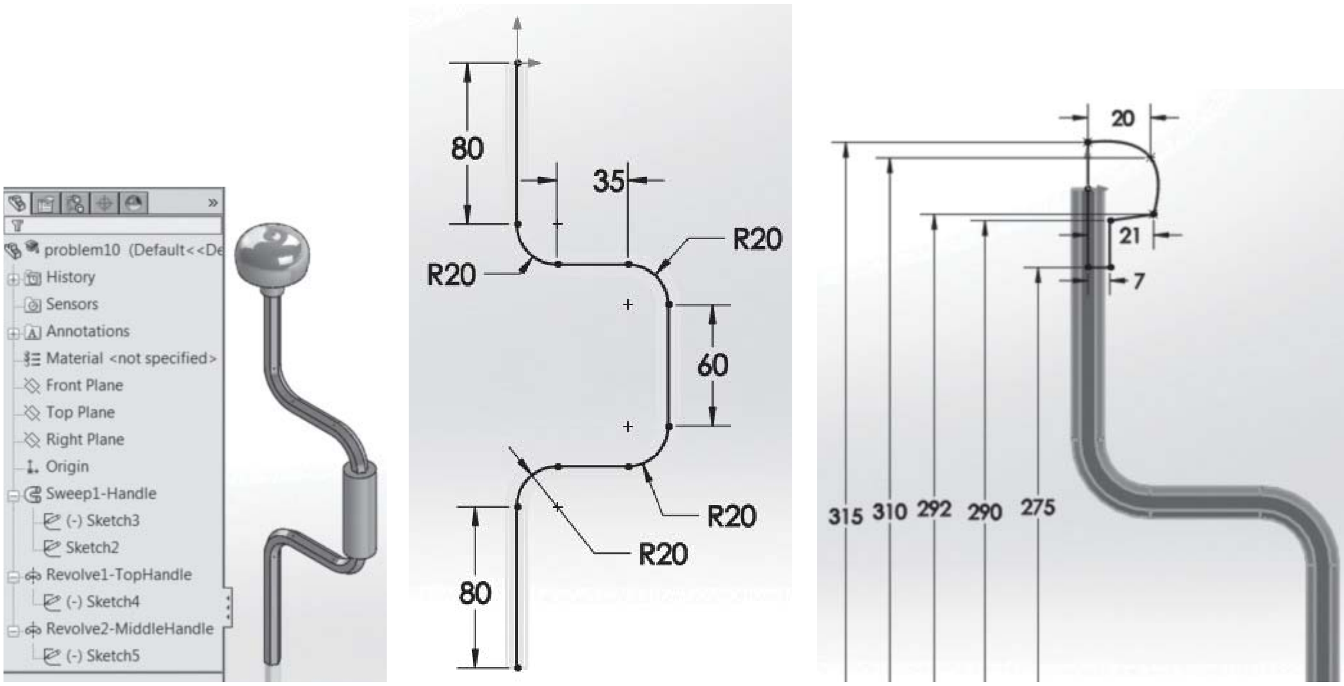
Step 10: Create *Sketch7* and *Helix/Spiral2*: Select *Plane3* > **Convert Entities** on **Sketch** tab > expand features tree > select *Sketch1* > ✔ > select *Sketch7* > **Insert > Curve > Helix/Spiral** > enter 0.25 for **Height** > click **Taper Helix** check-box > enter 30 for angle > click **Taper outward** checkbox > ✔.



Step 11: Create *Cut-SweepEndThread* feature: **Swept Cut** on **Features** tab > select *Sketch6* as profile and *Helix/Spiral2* as path > ✔.

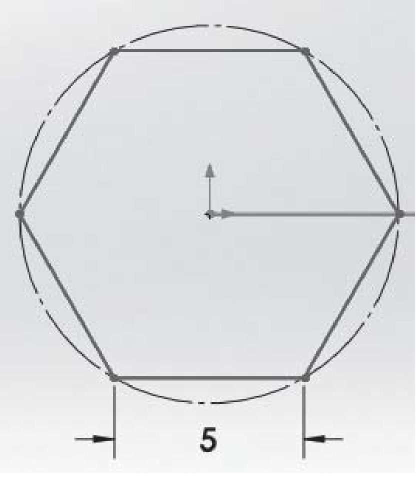
problems

1. What is a feature? Give two examples of features.
2. Why can an extruded cut or a revolved cut NOT be a base feature?
3. What is the required input needed to create a sweep? What is the optional input?
4. What is the required input needed to create a loft? What is the optional input?
5. Table 4.1 shows a rib feature. Which is the better way to create it: using a rib or an extrusion? Explain your answer.
6. Table 4.1 shows a block that is shelled. Which is the better way to create it: shelling or extrusion cut? Explain your answer.
7. A spur gear has a pitch circle radius of 3 inches, a pressure angle of 14.5°, and a number of teeth of 20. Calculate all the parameters required to create the gear CAD model. Create the CAD model.
8. Same as Problem 7, but for a pitch circle radius of 100 mm, pressure angle of 14.5°, and a number of teeth of 30.
9. Create a macro to automate the creation of a donut revolve.
10. Create the brace drill handle shown in Figure 4.18. All dimensions are in millimeters.

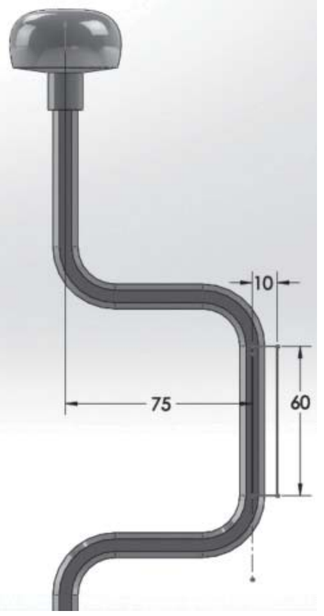


All vertical dimensions are measured from the bottom end of the drill handle. The top curve is a spline connecting three points.

FIGURE 4.18
Brace drill handle



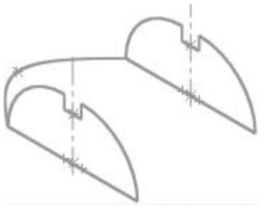
Drill handle cross section



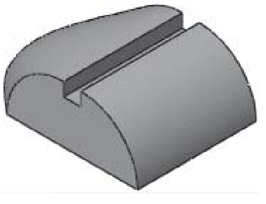
Cross section of middle handle

FIGURE 4.18
(*continued*)

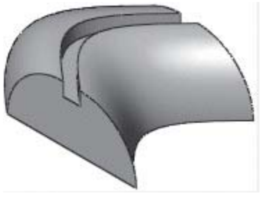
11. Create the loft feature shown in Figure 4.19. Assume dimensions. Hint #1: This loft shows you the local and global influence of the guide curve. Hint #2: Use **Guide curves influence types** under the **Guide Curves** section shown on the left pane of the screen.



Two profiles and guide curve



Local influence of guide curve



Global influence of guide curve

FIGURE 4.19
Influence of guide curve

12. Create the CAD model of the helical spring shown in Figure 4.20. All dimensions are in centimeters.

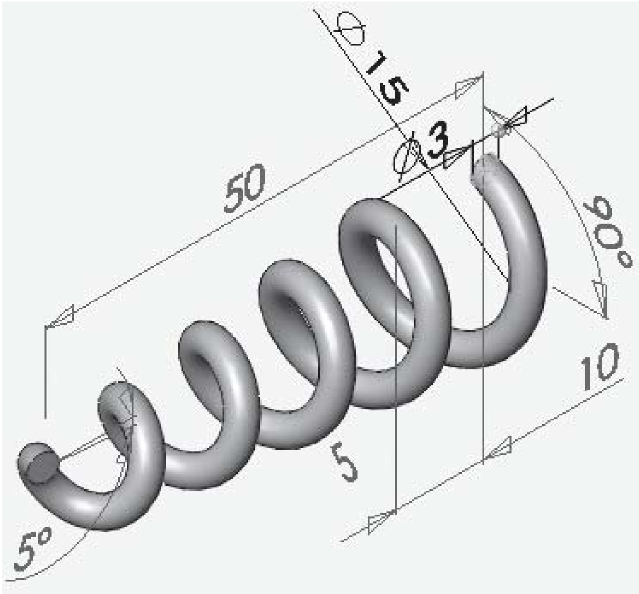
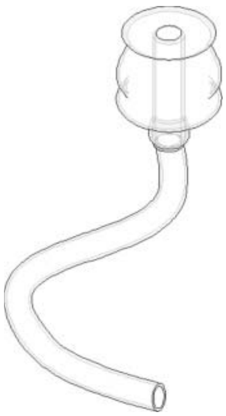


FIGURE 4.20
Helical spring

13. Create the CAD model of the 3D probe shown in Figure 4.21. All dimensions are in millimeters.



x	*y*	*z*
0	2.4	0
0	–50	0
0	–60	150
15	–65	160
150	–65	160

The probe uses a 3D curve to define its profile. The probe is hollow. The outer diameter is 20 mm and thickness is 2 mm

FIGURE 4.21
3D probe

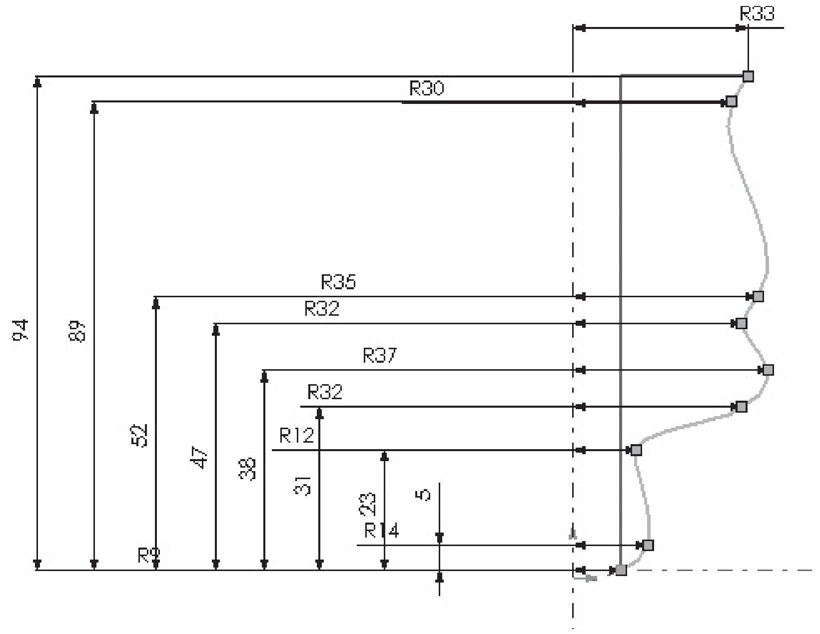
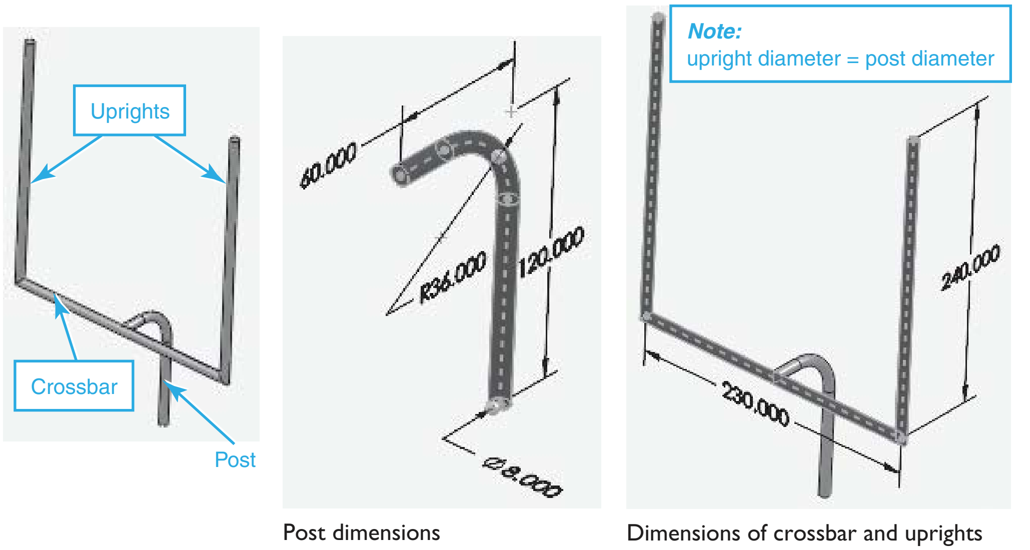


FIGURE 4.21
(*continued*)

14. Create the CAD model of the football goal post shown in Figure 4.22. All dimensions are in inches. Hint: The dimensions of the post are per NFL specs: the post is 10 feet (120 in.) high, the crossbar is 18.5 feet (222 in.) wide from the inner edges of the uprights, and the uprights are 20 feet (240 in.) high. Check www.sportsknowhow.com/pops/football-field-pro.html. The diameter of the post tubes is arbitrary, so we use 8 in. here.

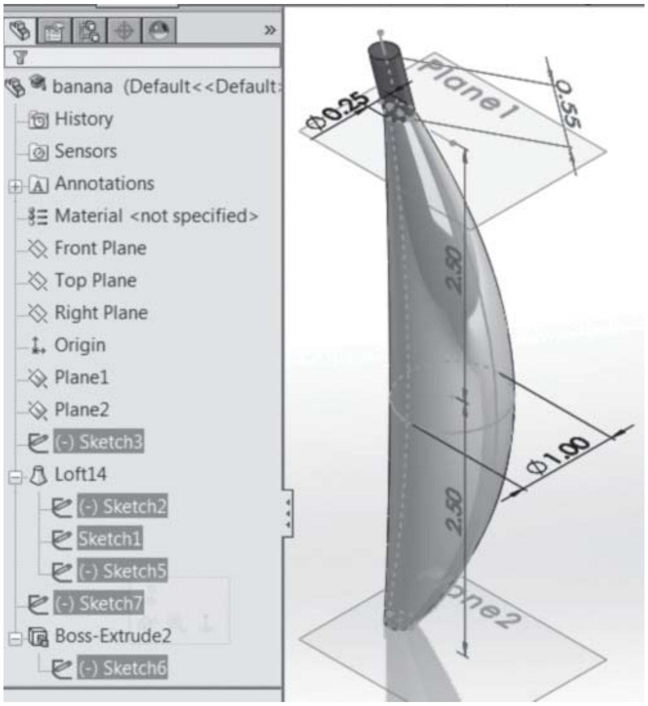


Post dimensions

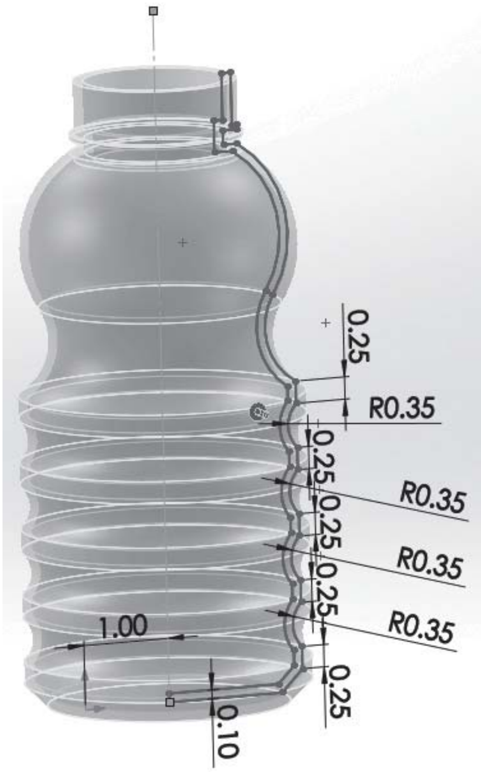
Dimensions of crossbar and uprights

FIGURE 4.22
Football goal post

15. Create the CAD models shown in Figure 4.23. All dimensions are in inches.



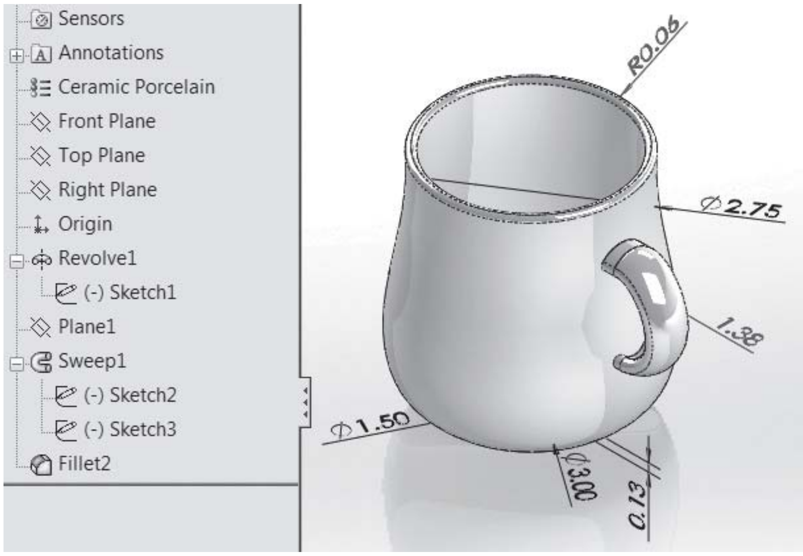
(A) Banana



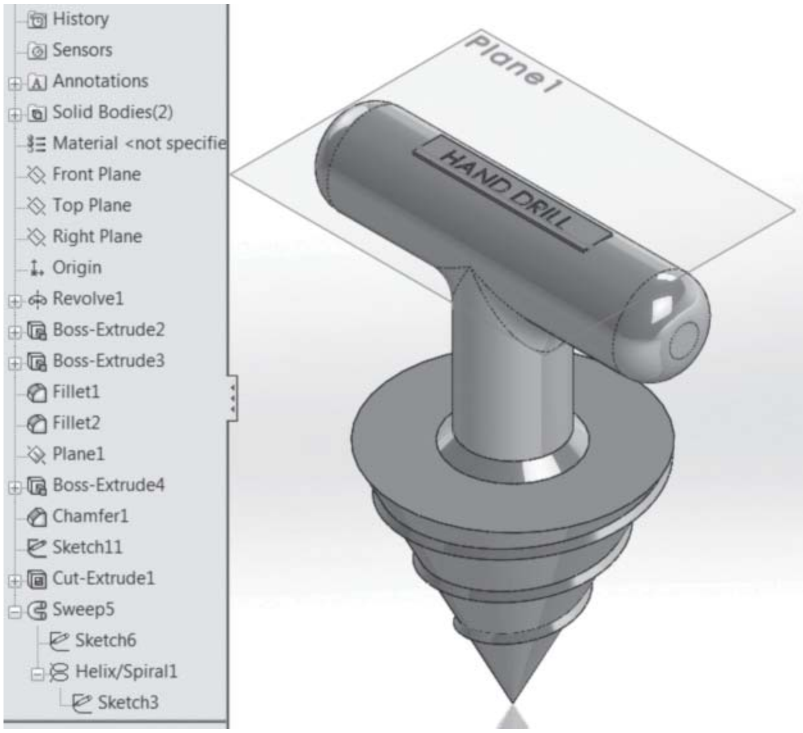
(B) Water bottle

FIGURE 4.23
CAD models

16. Create the CAD models shown in Figure 4.24. All dimensions are in inches.



(A) Coffee mug



(B) Hand Drill

FIGURE 4.24
CAD models

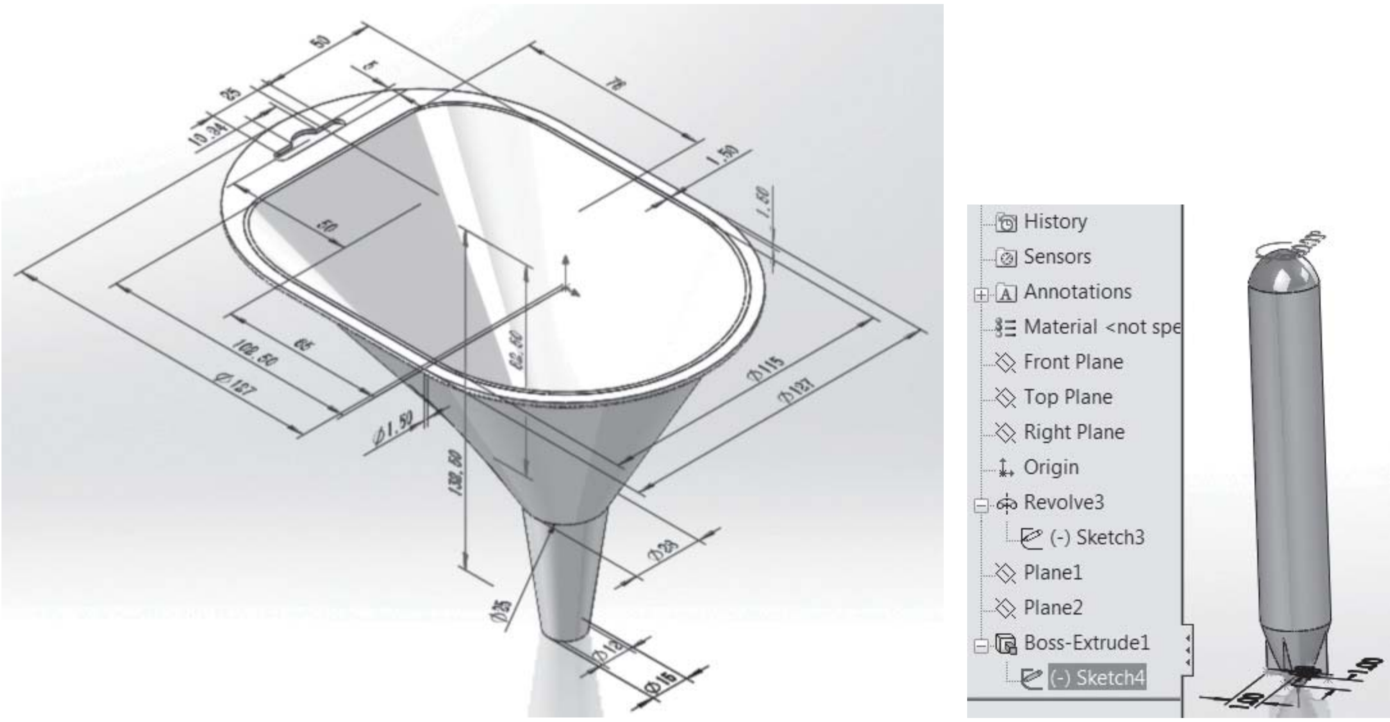
17. Create the CAD models shown in Figure 4.25. All dimensions are in inches.

(A) Goblet

(B) Baseball bat

FIGURE 4.25
CAD models

18. Create the CAD models shown in Figure 4.26.

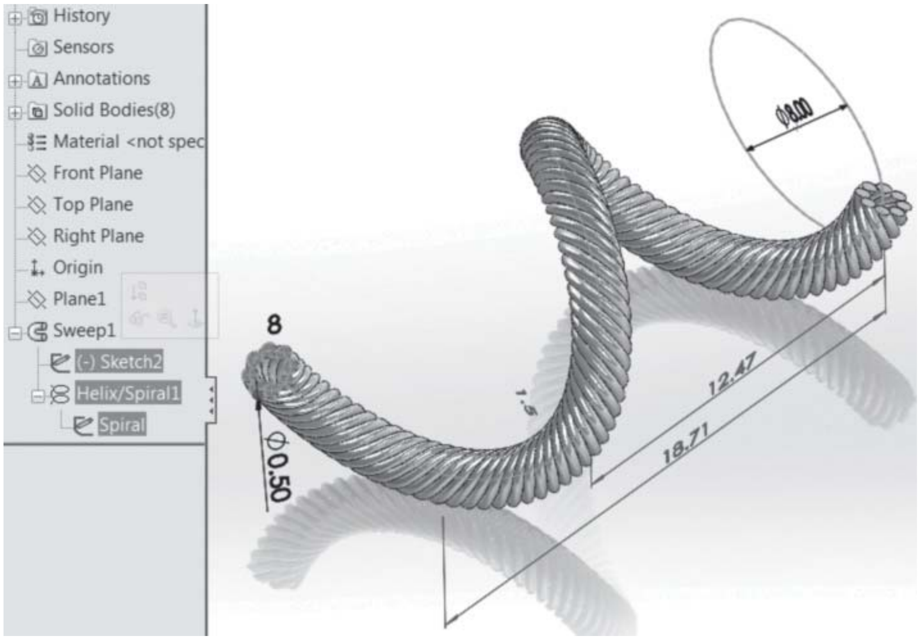


(A) Funnel (dimensions in mm)

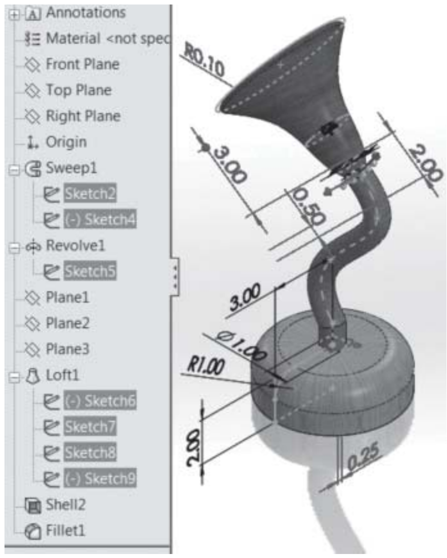
(B) Torpedo (dimensions in inches)

FIGURE 4.26
CAD models

19. Create the CAD models shown in Figure 4.27. All dimensions are in inches.



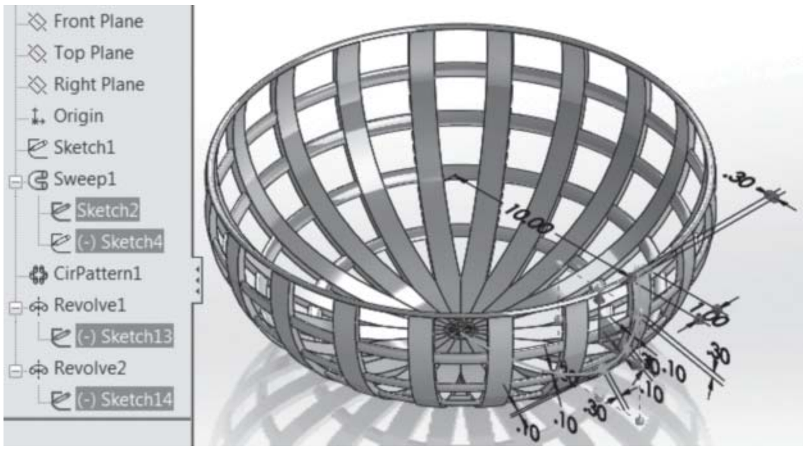
(A) Fiber optic wire



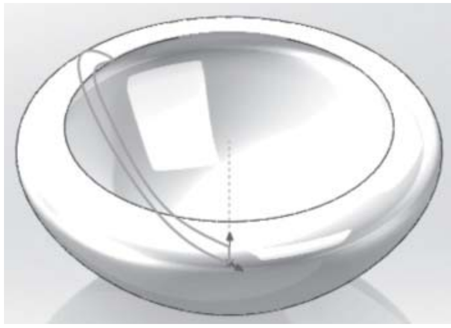
(B) Desk lamp

FIGURE 4.27
CAD models

20. Create the CAD models shown in Figure 4.28. All dimensions are in inches.



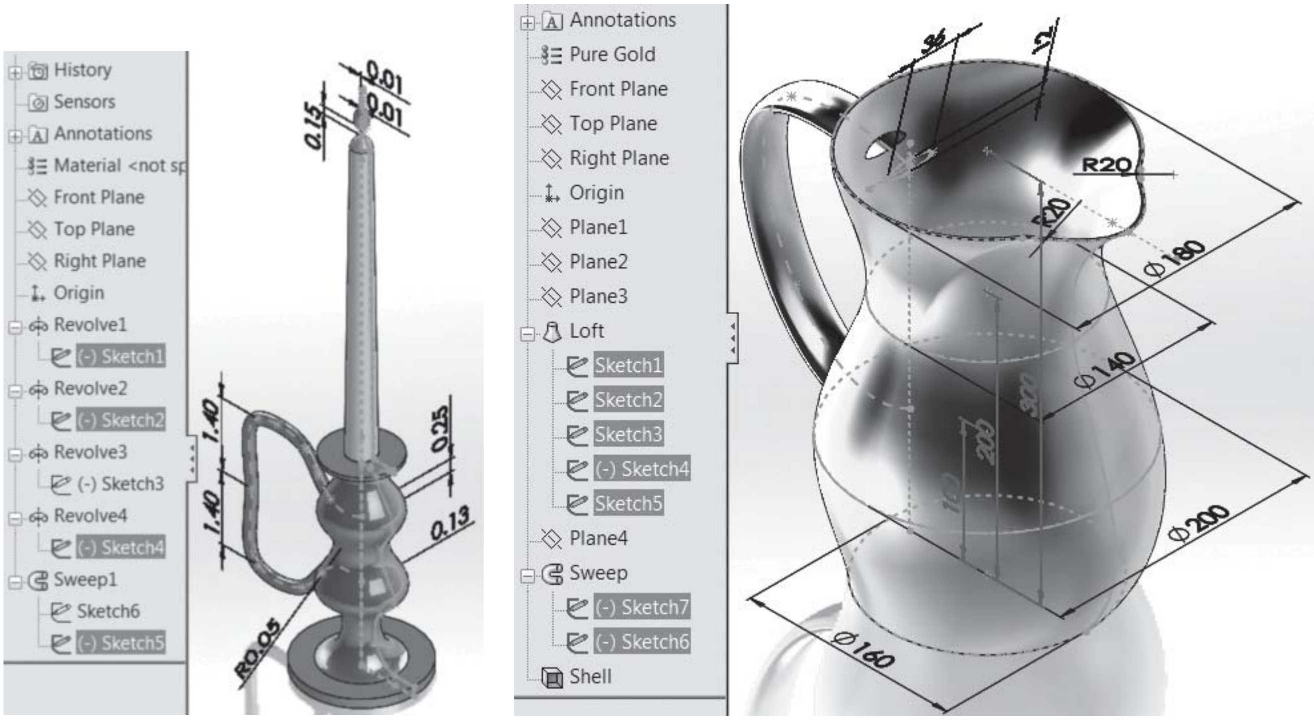
(A) Fruit basket



(B) Bowl

FIGURE 4.28
CAD models

21. Create the CAD models shown in Figure 4.29.

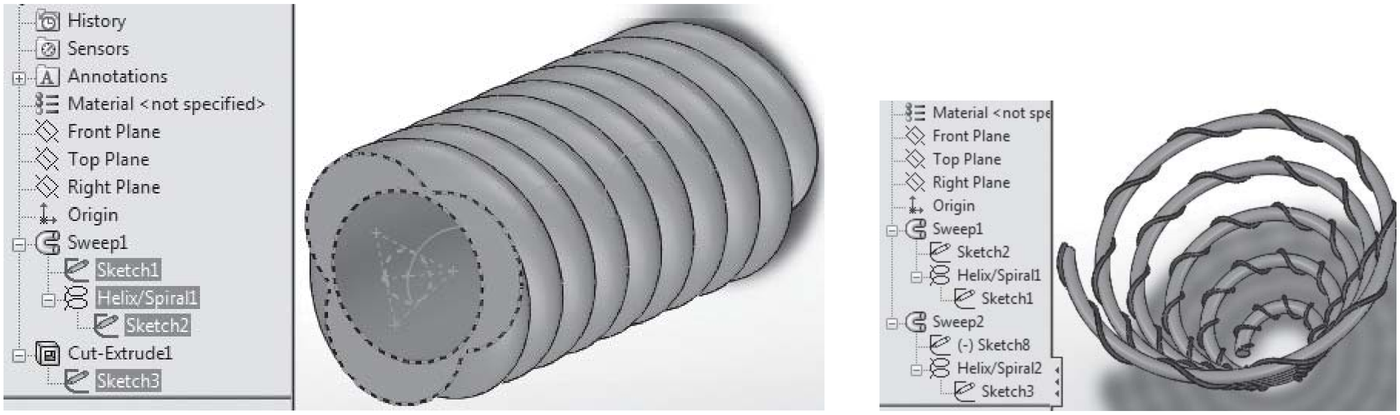


(A) Candle and holder (dimensions in inches)

(B) Pitcher (dimensions in mm)

FIGURE 4.29
CAD models

22. Create the CAD models shown in Figure 4.30.

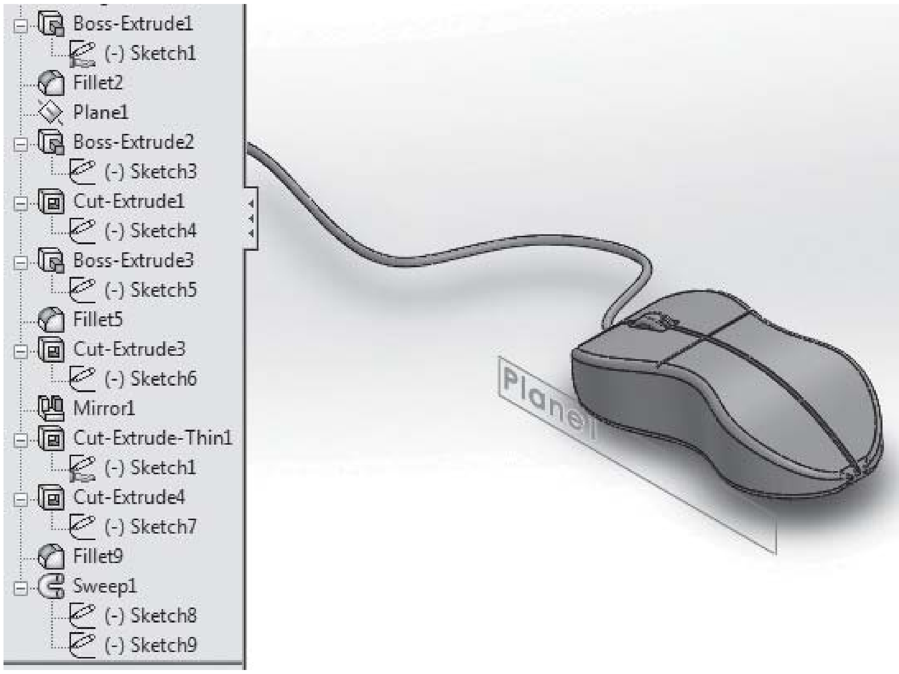


(A) A flex

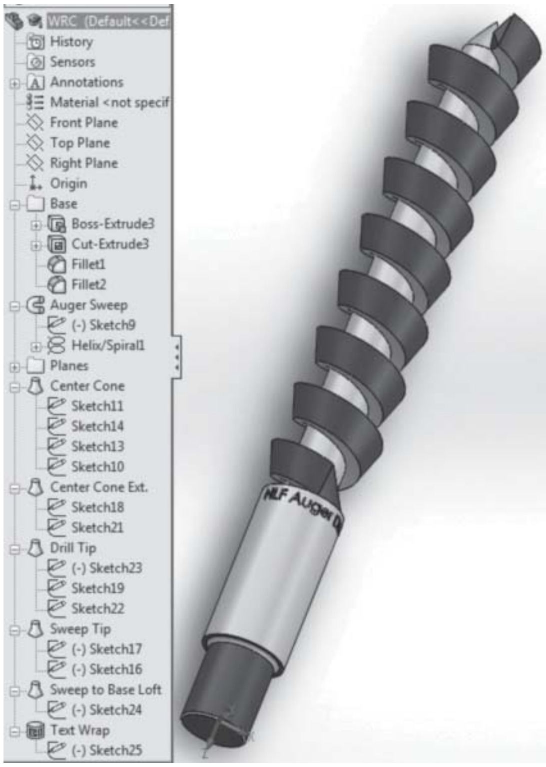
(B) Steel cone

FIGURE 4.30
CAD models

23. Create the CAD models shown in Figure 4.31.



(A) Computer mouse



(B) Auger drill

FIGURE 4.31
CAD models

CHAPTER

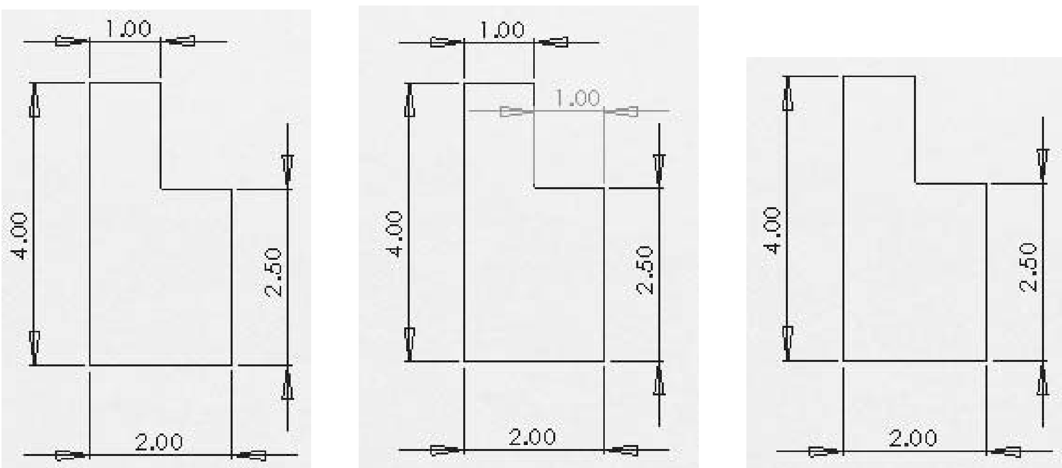
5

Drawings

5.1 Introduction

Engineering drawings are important in engineering for two reasons. First, designers use them to document their designs. Second, they are the *de facto* standard in design communications among the various groups and departments involved in the product design and manufacturing (e.g., manufacturing, tooling, production, inspection, assembly). The typical (standard) sizes of engineering drawings are A, B, C, D, and E.

Generating engineering drawings requires knowledge of drafting and communication rules. In general, it is not a good idea to over-dimension or under-dimension views, just as it is not a good idea to over-define or under-define sketches. The drawing should be fully defined, just like a sketch. Figure 5.1 shows the three possible scenarios of dimensioning a drawing. If you must show a redundant dimension, use the text "REF" next to it or include it in parentheses. Consequently, you would display the extra dimension shown in Figure 5.1B as 1.00REF or (1.00).



(A) Fully dimensioned (B) Over-dimensioned (C) Under-dimensioned

FIGURE 5.1
Dimensioning a drawing

Why is dimensioning a drawing so important? The way you display dimensions on the views implies how to manufacture the part. Let us limit our discussion to machining a part—that is, milling and/or drilling it. Dimensioning drawings requires manufacturing knowledge. The CAD designer needs to know the common manufacturing processes and how they work. The designer also must be familiar with the common stock shapes available off the shelf. The way you dimension a drawing tells the machinist the way you want to produce the part, even if you do not do so intentionally. More importantly, the dimensioning scheme you use will influence the machining cost significantly.

The dimensioning scheme shown in Figure 5.1A is ideal from a machining point of view. The scheme tells the machinist to use the left and bottom faces as references (datums) to measure from. The machinist begins with a rectangular block as a stock, cuts the left and bottom faces, and squares them, that is, makes them perpendicular to each other. Next, the machinist measures and cuts the two horizontal dimensions (2.00 and 1.00) using the left face. Finally, the machinist measures and cuts the two vertical dimensions (4.00 and 2.50), using the bottom face to finish the part machining. Note: One problem with the part dimensions as shown here is that the machinist cannot produce them because they require zero tolerances. We must specify tolerances. This is the topic of Chapter 12. Without tolerances, the machining cost is prohibitively expensive.

The dimensioning scheme also conveys which dimensions are most important for the part functional requirements. In other words, you are telling the machinist that the 1.00 and 2.00 horizontal dimensions are more important than any other combination (e.g., 1.00 and 1.00) or the other 1.00 and 2.00. Similarly, the 4.00 and 2.50 vertical dimensions are more important than the 4.00 and 1.50 or the 2.50 and 1.50 combinations. The horizontal 1.00 and 1.00 combination and the vertical 2.50 and 1.50 combination are the worst dimensioning scheme because they make the machining cost expensive for this part. This scheme forces the machinist to machine the step faces (horizontal and vertical) very accurately to make them datums to measure from.

The best way to dimension a drawing in SolidWorks is to use the concept of model items we covered in Tutorial 1–4 in Chapter 1. Review the tutorial as it covers the basic important concepts in creating drawings.

5.2 Engineering Drafting and Graphics Communication

Section 5.1 provides a glimpse of how important dimensioning and tolerancing a part are. Some CAD designers, especially beginners or those without manufacturing experience or knowledge, do not see the value of dimensioning for manufacturing, but it is crucial. If unable to read the design intent embedded in your dimensioning scheme, the machinist will use his or her own interpretations to machine the part, resulting in a high rejection (scarp) rate during inspection and making the production very inefficient and expensive. You may want to consult the following books for more details on dimensioning and tolerancing:

- *Dimensioning and Tolerancing Handbook* by Bruce A. Wilson, Genium Publishing Corporation (ISBN 0-931690-80-3)
- *Technical Drawing* by Frederick E. Giesecke et al., Pearson Publishing Company (ISBN 0-13-008183-3)

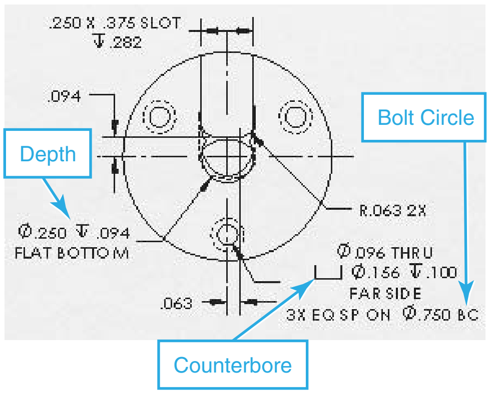
The guidelines to follow for engineering drafting and graphics communication are standard. The American National Standards Institute (ANSI) and the International Organization for Standardization (ISO) committees maintain these standards in coordination and consultation with companies. There are abbreviation rules for use on drawings and in text, dimensioning rules, and tolerancing rules. The American Society of Mechanical Engineers (ASME) publishes two documents on these subjects. ASME Y1.1-1989, “Abbreviations for Use on Drawings and in Text,” covers the abbreviation rules. ASME Y14.5M-1994, “Dimensioning and Tolerancing,” covers the rules on how to display dimensions and tolerances on a drawing. ASME has published a revision of its well-known ASME Y14.5M-1994; the new revision is ASME Y14.5-2009.

5.3 ASME Abbreviation Rules

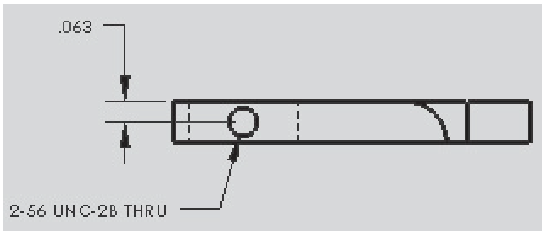
ASME indicates where and how abbreviations should be used as well as some basic rules. ASME publishes the ASME Y1.1-1989 book that lists the abbreviations in alphabetical order. Here are some abbreviation rules:

1. Minimize using abbreviations: Using abbreviations on drawings should be minimized for clarity purposes. ASME's abbreviation book begins by stating that "the purpose is to establish standard abbreviations rather than to promote the use of abbreviations." The reason is that abbreviations are language-dependent.
2. Pay attention to foreign use: Abbreviations are conventional representations of words or names and may differ from one language to another. Thus, companies that are multinational should not use abbreviations.
3. Pay attention to clarity: Abbreviations should be used only where necessary to save space and time. Also, use only the obvious abbreviations that are easy to understand; otherwise, spell out the word(s).
4. Be aware of duplicates: Duplicate abbreviations exist for some words because of the established practice. Thus, be careful.
5. Define when extensively used: Books and large publications that use abbreviations should define them in one convenient place for readers to find.
6. Take advantage of single use: Single organizations may use abbreviations more freely than do others because it is easier to set communication standards within one company.
7. Avoid using nonstandard abbreviations: Refrain from making up and using your own abbreviations. Nobody but you will understand them.
8. Follow military rules: Military agencies set and publish their own abbreviations for contractors to use.
9. Use all capital letters: All abbreviations should be shown in capital letters on drawings and in lowercase in text.
10. Avoid using subscripts: Subscripts should not be used in abbreviations.

Figure 5.2 shows sample abbreviations.



(A) Holes



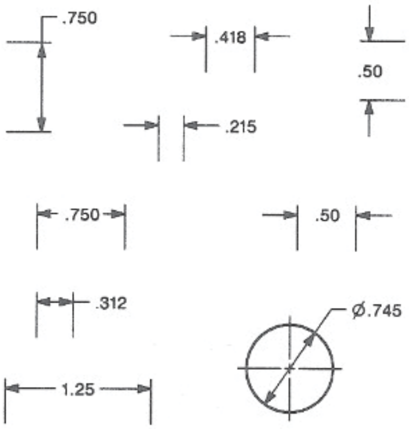
(B) Thread

This is a thread code:
2: shank size; 56: threads/inch (TPI); UNC: Unified Coarse; 2B: Internal; THRU: Through

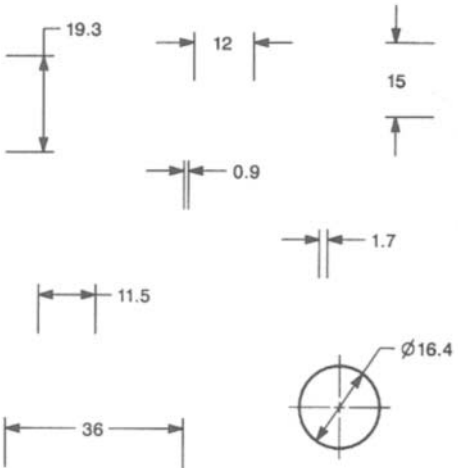
FIGURE 5.2
Sample abbreviations

5.4 ASME Drafting Rules

Some of the ASME drafting rules are as follows (Figure 5.3 shows some of these):



(A) Millimeter-based rules



(B) Inch-based rules

FIGURE 5.3
Sample drafting rules
(Reprinted from ASME Y14.5M-1994 [R2004], by permission of The American Society of Mechanical Engineers. All rights reserved.)

1. Use tolerances: Each dimension should have a tolerance except those dimensions that are labeled as reference, as discussed in Section 5.1.
2. Provide full feature definition: Each feature should be fully dimensioned so that it is fully understood.
3. Show only what is needed: Show necessary dimensions for a complete definition of the part. Do not give more dimensions than necessary. Minimize the use of reference dimensions.
4. Follow functional requirements: Select and arrange dimensions to meet the functional and mating requirements of a part. See Section 5.1, for example, on whether to use a 4.00 and 2.50 or 2.50 and 1.50 combination. Also, there should be one and only one way to interpret the dimensions.
5. Do not specify manufacturing methods: Define the drawing without specifying manufacturing methods. For example, show a hole diameter on the drawing without indicating whether it is to be drilled, milled, punched, or reamed.
6. Show processing dimensions: If needed, you may show processing dimensions to indicate, for example, finish or shrink allowance. In such a case, label the dimension as NON MANDATORY (MFG DATA), all in caps as shown here.
7. Show dimensions clearly: Arrange dimensions so that they are easy to read, provide required information, are shown in true profile, and refer to visible outlines.
8. Use linear dimensions for specific parts: Use linear dimensions for wires, cables, sheets, and rods that are produced according to gage or code numbers. These dimensions usually show the diameter or the thickness.
9. You may not specify a 90° angle: If you have a pattern in which the centerlines are shown at right angles on the drawing, there is no need to show the angle; a 90° angle applies.
10. Keep temperature in mind: All dimensions shown on a drawing are assumed to be measured at room temperature (20°C or 68°F). If measurements are made at other temperatures, provide appropriate compensation.
11. Understand geometric tolerances: When you apply geometric tolerances, they apply to the full depth, length, and width of the feature. We cover geometric tolerances later in the book.

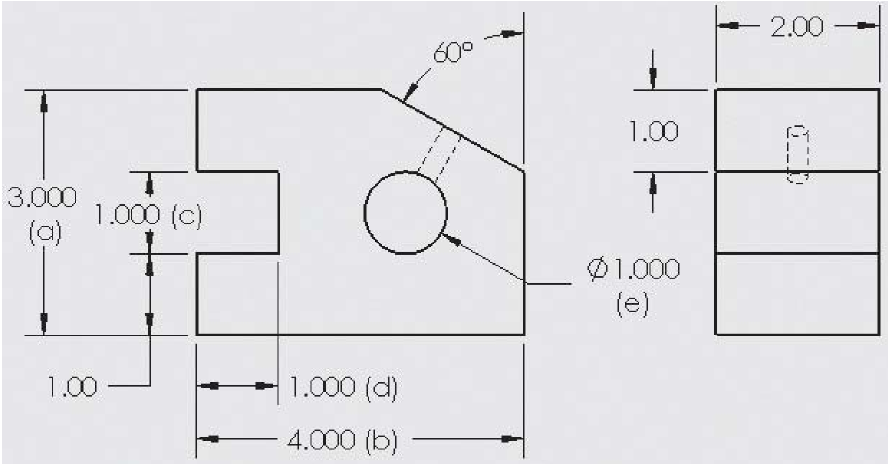
5.5 ASME Dimensioning Rules

We need to be familiar with the types of dimensions in order to understand their rules. These types are Cartesian, radial, angular, true length, and ordinate or baseline. Cartesian dimensions are specified along the horizontal and vertical directions of the drawing views as we have done in Chapter 1's tutorials. Radial dimensions specify a circle's or arc's radius or diameter. Angular dimensions specify an angle. True length dimensions specify a dimension along a line that is not horizontal or vertical. Ordinate or baseline dimensions specify all dimensions in one direction from a reference face or datum.

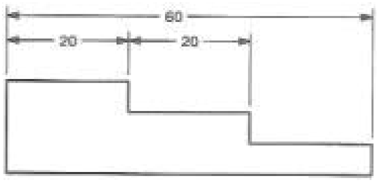
As a general rule for specifying ordinate or baseline dimensions, use fewer datums to reduce the machining cost of the part by requiring fewer part surfaces (faces) that must be machined with high accuracy to measure other dimensions from. Figure 5.1A shows the Cartesian and ordinate dimensioning types. The other dimensioning types are more obvious. The horizontal and vertical dimensions shown are for both the Cartesian and ordinate types. The horizontal dimensions (2.00 and 1.00) use the left face of the part as the datum. The vertical dimensions (4.00 and 2.50) use the bottom face of the part as the datum. The optimum ordinate dimensioning strategy (scheme) is to use only one datum per direction. This strategy guarantees the least machining cost to manufacture the part.

Some of the ASME dimensioning rules are as follows:

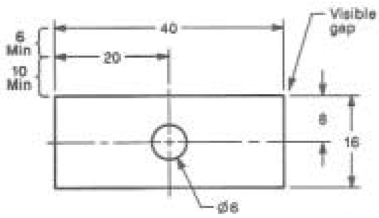
1. Be careful when to use a zero before the decimal point: If the dimension is in millimeters and less than 1, use a zero before the decimal point; for example, use 0.5, not .5. If the dimension is in inches, do not use a zero; for example, use .5, not 0.5. See Figure 5.3.
2. Do not use a zero or the decimal point for a whole number millimeter dimension: For example, use 35, not 35.0.
3. Do not add a zero to a decimal millimeter dimension: For example, use 35.6, not 35.60.
4. Use the same number of decimal places as its tolerance for an inch dimension: For example, use 4.500, not 4.5 if the dimension tolerance requires three decimal places.
5. Show decimal points clearly: The decimal points must be uniform, dense, and large enough to be clearly visible.
6. Use dimension lines correctly: ASME rules prefer that dimension lines be broken for the insertion of the dimension value (number). You may use unbroken dimension lines, as shown in Figure 5.1. In such a case, the dimension value should be shown above the dimension line for horizontal lines. Figure 5.1 follows this rule because the CAD/CAM system uses and enforces ASME rules.
7. Group dimension lines: If possible, have all related dimension lines shown next to each other in one group to make it easier for users to read the engineering drawing. In Figure 5.1, we could move the 4.00 dimension line to the right next to the 2.50 dimension line to make them as one group.
8. Space dimension lines: The space between the first dimension line and the part outline should not be less than 10 mm; and the space between succeeding parallel dimension lines should not be less than 6 mm.
9. Do not cross dimension lines: Avoid crossing dimension lines. When unavoidable, the dimension lines are unbroken. Figures 5.1 and 5.4 follow this rule where all dimension lines (lines with arrows) are not crossing.
10. Do not cross extension (projection) lines: An **extension line** is a line that is perpendicular to the part outline to allow placing the dimension line. For example, in Figure 5.1A, the two vertical lines extending down to place the 2.00 dimension line are extension lines. There is always a gap between the extension line and the part outline as shown. Extension lines should neither cross each other nor cross dimension lines. Where unavoidable, break the extension line.
11. Use leaders (leader lines) if needed: A **leader** is a line used to place a dimension (e.g., a circle radius), a note (text on a drawing), or a symbol (special character) on a drawing. A leader terminates in an arrowhead if it ends on a part outline. If it ends inside the part outline, terminate it with a dot. Also, a leader should be an inclined straight line with a short horizontal portion extending to the first or last letter of the note, or the first or last digit of the dimension. Also, make adjacent leaders parallel to each other for ease of display.
12. Use reference dimensions if needed. When an overall dimension is specified, one intermediate dimension is omitted or identified as a reference dimension.



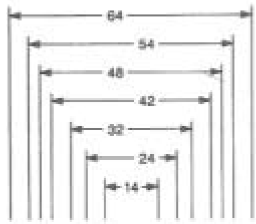
ASME Rule 6 (break dimension line to insert dimension value)



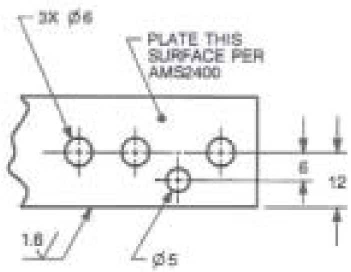
ASME Rule 7 (Group dim lines)



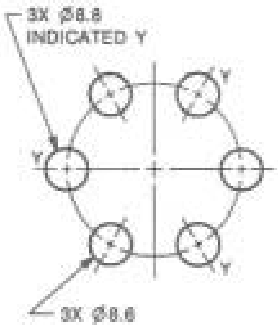
ASME Rule 8 (Space out dim lines)



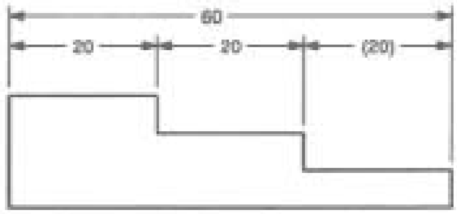
ASME Rule 9 (no dim lines crossing)



ASME Rule 11 (use leaders)



ASME Rule 11 (minimize leaders)



ASME Rule 12 (use reference dimension)

FIGURE 5.4

Sample dimensioning rules (inches)

(Reprinted from ASME Y14.5M-1994 [R2004], by permission of The American Society of Mechanical Engineers. All rights reserved.)

Figure 5.4 shows sample dimensioning (we refer to the rule numbers just listed).

The drawing illustrating ASME Rule 6 and shown in Figure 5.4 displays a combination of baseline and chain dimensioning. This drawing will communicate the importance (accuracy) of dimensions (a), (b), (c), (d), and (e). Although the dimensions with fewer decimal places are less important to the functionality of the part, not enough accuracy will create a nonfunctional part; too much accuracy will be complicated and too expensive to manufacture. The outer shape of this part is fully defined. However, the two hole features have not been identified or located.

The drawing shown in Figure 5.5 is dimensioned in metric units of millimeters. Dimension (a) reads as follows: Four times, radius 5 mm. Slots are spaced equally around the part. As in ASME dimensioning rule 2, there are no trailing zeros to designate accuracy or tolerance when dimensioning in metric units.

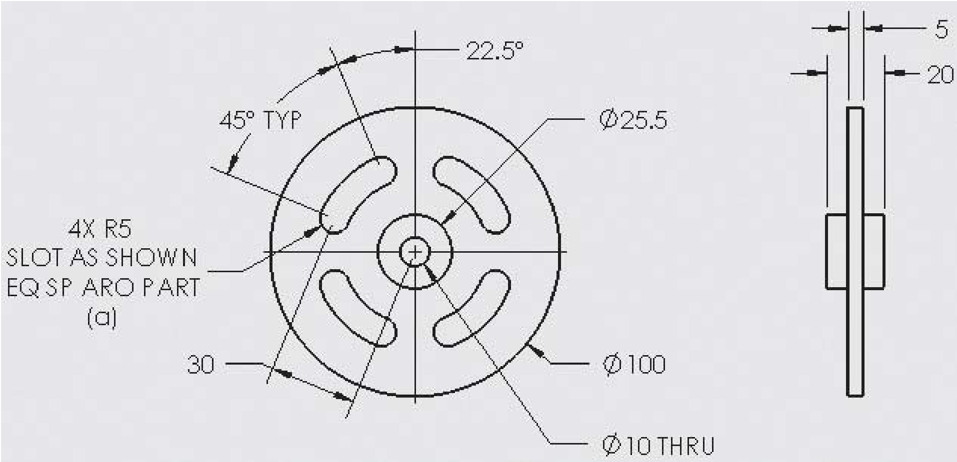


FIGURE 5.5
Sample dimensioning rules (millimeters)

5.6 Dimensions

We extend the dimensioning rules coverage here by showing how SolidWorks implements these rules, thus helping you to follow them. Figure 5.6 shows the SolidWorks **Dimensions/Relations** toolbar. Click this sequence to show/hide (toggle) the toolbar:

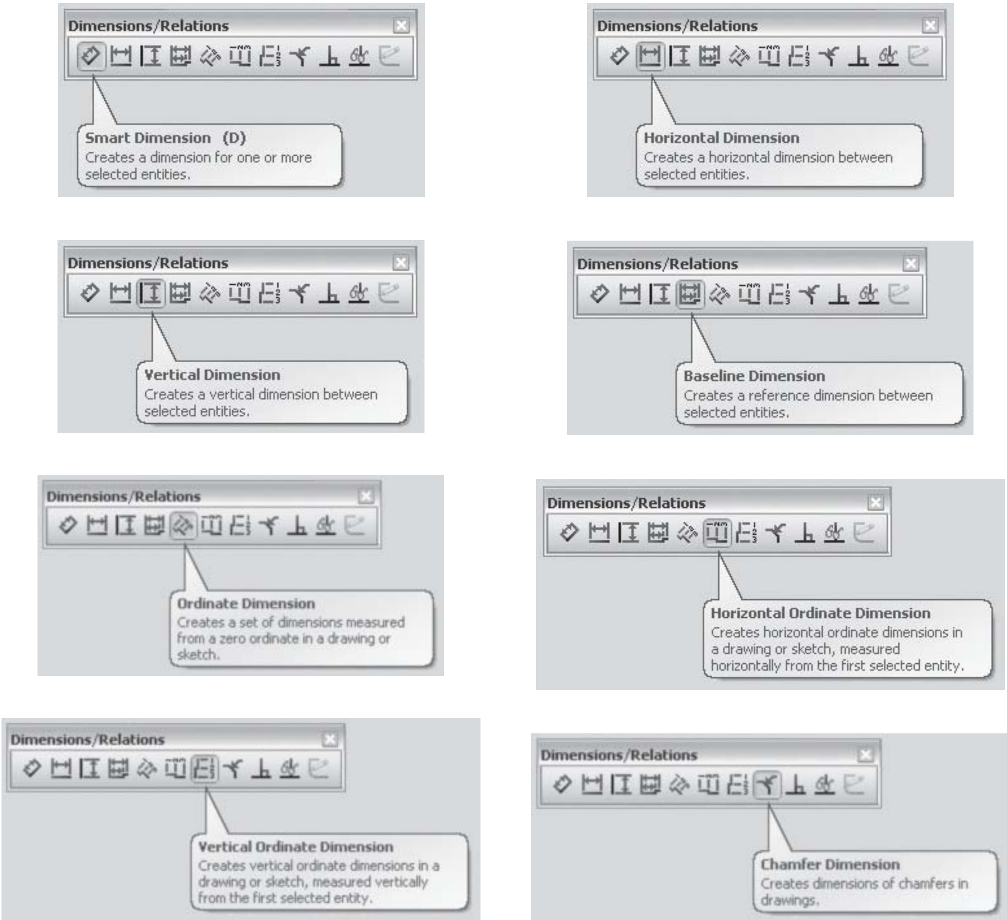


FIGURE 5.6
Types of dimensions

View > **Toolbars** > **Dimensions/Relations**, or **Annotation** tab (must be in a drawing to see it) > **Smart Dimension** dropdown. You can insert a dimension type by clicking as follows: Select icon from the **Dimensions/Relations** toolbar or the **Smart Dimension** dropdown > Select entity to dimension > Place dimension > ✔.

When you insert a dimension, SolidWorks allows you to add symbols (abbreviations) to it. While the dimension mode is active, you see the **Dimension Text** block on the bottom left of the screen.

Example 5.1 Create the model shown in Figure 5.7, and use it to show baseline, ordinate, and chamfer dimensions in a drawing.

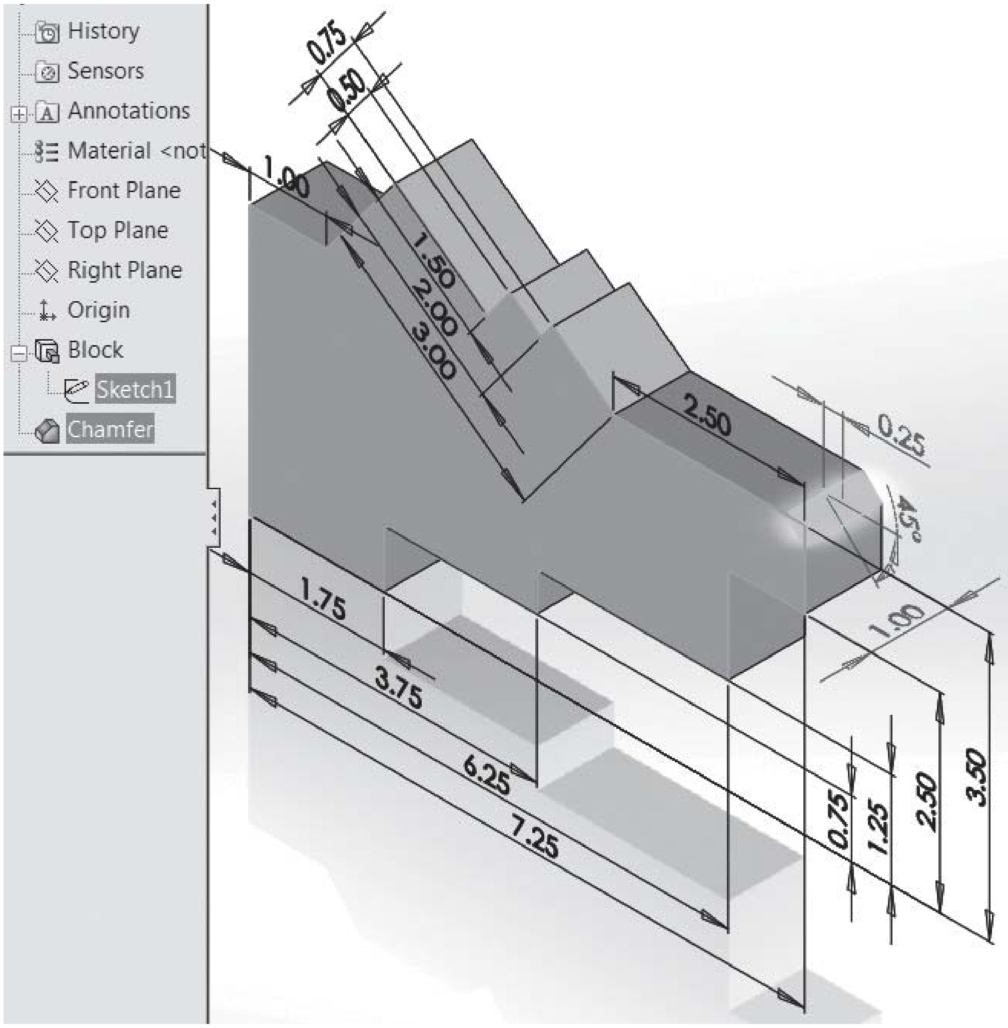


FIGURE 5.7
A stepped block

Solution The block shown in Figure 5.7 is an extrusion. The dimensions shown are designed to illustrate the use of baseline and ordinate dimension types in a drawing. The cross section shown in Figure 5.8 is anchored at the origin shown. It also uses three types of relations: horizontal, vertical, and collinear. SolidWorks creates the horizontal and vertical lines implicitly by predicting your intention as you sketch. The dimensioning scheme of the cross section is selected to minimize the number of reference faces (datums) and to lend itself to the baseline and ordinate dimension schemes we use in the drawing. The baseline dimensions are identical to those shown in Figure 5.8.

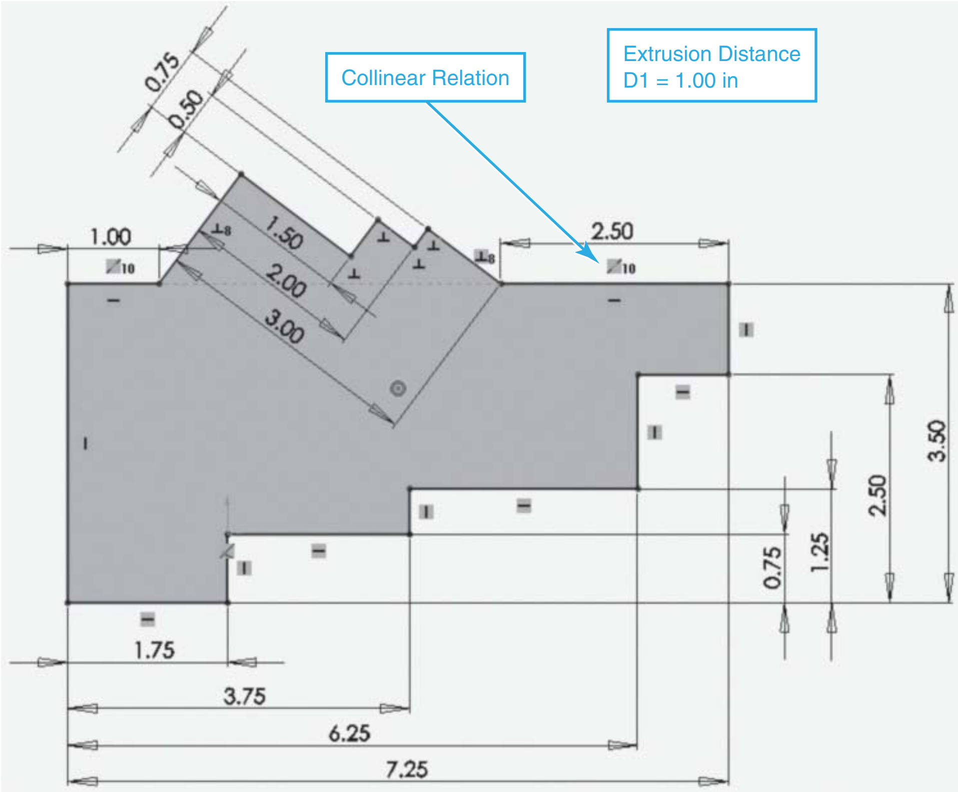
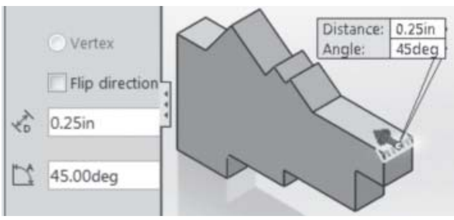


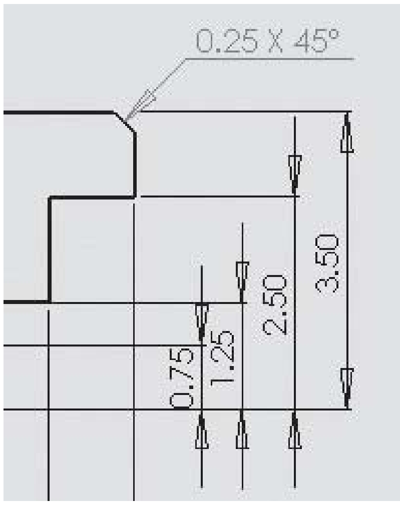
FIGURE 5.8
A stepped block cross section

Step 1: Create *Sketch1* and *Block* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Line** on **Sketch** tab > click origin and sketch and dimension as shown in Figure 5.8 > select one of the top horizontal line segments + **Ctrl** + select other top horizontal line segment > **Collinear** from **Add Relations** options > exit sketch > reverse extrusion direction > enter 1.0 for thickness **D1** > ✔ > **File** > **Save As** > *example5.1* > **Save**.

Step 2: Chamfer the corner edge of *Block* feature: **Fillet** dropdown on **Features** tab > **Chamfer** > select *Block* edge shown > use 0.25 for **D** and 45 degrees for angle > ✔.

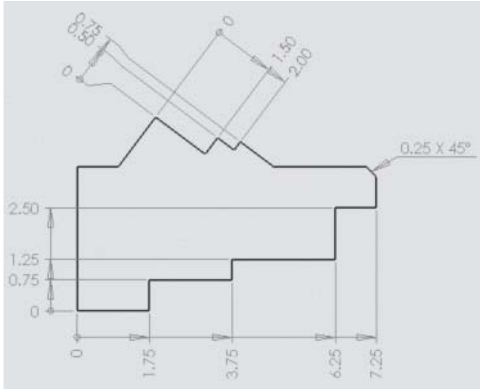


Step 3: Create drawing: **File** > **New** > **Drawing** > **OK** > **Next (arrow)** > place a front view in drawing > ✔ > **Annotation** tab > **Model Items** > **Entire model** for **Source/Destination** > ✔ > delete the chamfer dimension and



re-create > **Annotation** tab > **Smart Dimension** dropdown > **Chamfer Dimension** > click chamfer > click one of its adjacent edges > click to position the chamfer dimension as shown > ✔.

Step 4: Create ordinate dimensions: Delete the dimensions created in Step 3 except chamfer dimension > **Ordinate dimension** > click far left edge of model > click other edges, one at a time > repeat for inclined edges > ✔.



HANDS-ON FOR EXAMPLE 5.1. Create two 1-inch diameter through holes in the front face of the block. The centers of the two holes are (1.5, 2) and (3.5, 2) from the left bottom corner of the block. Use both baseline and ordinate dimensioning schemes to dimension them in a drawing.

5.7 Drawing Content and Layout

FIGURE 5.9
Annotations

As we covered in Chapter 1, you can create a part or an assembly drawing. Figures 1.21 and 1.22 show part drawings, and Figure 1.23C shows an assembly drawing. As demonstrated in these figures, drawing content includes the views, dimensions, bill of materials, and a title block. Typically, a drawing includes three views: front, top, and right. You may include the isometric view as a fourth view. Other views, such as sectioned views, may be added if they are needed to clarify the drawing.

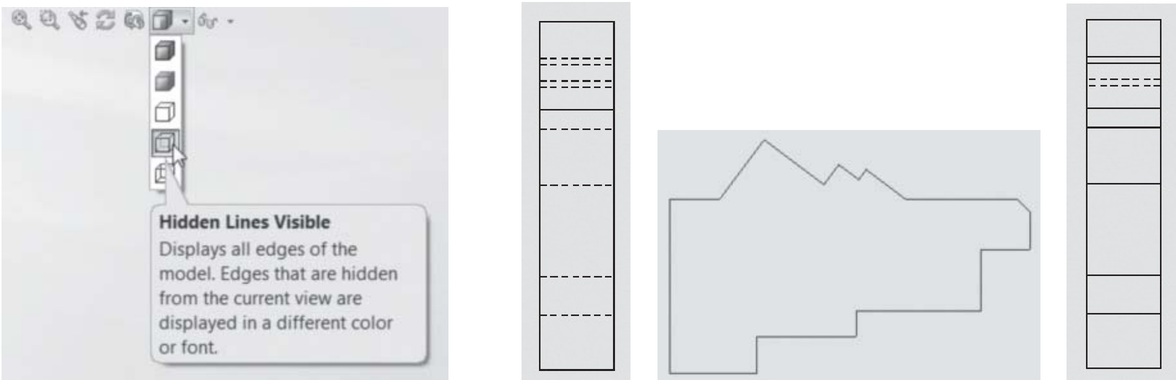
In addition, a drawing may include tolerances, notes, balloons, hole callouts, weld symbols, surface finish symbols, surface roughness values, and a bill of materials (BOM). A BOM is usually used in assembly drawings. It typically shows the part number, how many instances of the part are used in the assembly, and what its material is. If a BOM is used in a part drawing, it shows the same information as the assembly BOM except for the number of instances. Companies may create special BOMs to fit their needs.

SolidWorks groups most of the drawing content under the **Annotation** tab or **Annotations** menu. While in a drawing, click **Annotation** tab to view its icons, or **Insert** (menu) > **Annotations** to access all the possible annotations (see Figure 5.9) you can add to a drawing. Figure 5.9 also shows a note (the text that reads as such) and a balloon (the circle that points to the note). Balloons can be added to notes, as shown in Figure 5.9, or they can be used to label parts of an assembly and relate them to item numbers in the assembly BOM.

5.8 Angle of Projection

A CAD/CAM system uses projection to generate 2D views from 3D models. The system does this by projecting the model onto a projection plane. SolidWorks offers two angles of projections: first angle and third angle. Both angles of projections produce orthographic views, but they are opposite to each other. First angle produces back, bottom, and left views. Third angle produces front, top, and right views. You may ask why there are two different sets of views. The United States and Canada use third-angle projection whereas Europe and other countries use first-angle projection.

To set the angle of projection, right-click anywhere on the drawing sheet > **Properties** > Select **Third angle** (or **First angle**) > **OK**. You can tell which angle of projection a drawing is using by displaying the hidden lines of the drawing views as shown in Figure 5.10.



(A) **Heads-up View** toolbar

(B) Front view (C)

B: First angle (left view)

C: Third angle (right view)

FIGURE 5.10

Angles of projection

5.9 Views

CAD/CAM systems offer different types of views to convey a model design. SolidWorks offers the following types (see Figure 5.11):

1. Named (orthographic) views: These are the standard orthographic views: front, top, right, back, bottom, and left. These are the views we have created so far in the book.
2. Section view: This view type allows you to look at the inside of the model by cutting a section in it. The cut could be straight (use one section line) or stepped (use stepped section line). The CAD/CAM system requires you to define a direction (section line) for the sectional view. To create a section view, click **View Layout** tab > **Section View** (or **Insert** menu **> Drawing view > Section**) > Sketch a line > Move (do not drag) the mouse to where to place the section view and click there.
3. Projected view: You may project an existing view in a given direction. This speeds up and standardizes view creation. For example, you may select a front view and click to the right or left of it to create a right or left view, respectively. Click **View Layout** tab > **Projected View** (or **Insert** menu **> Drawing view > Projected**) >

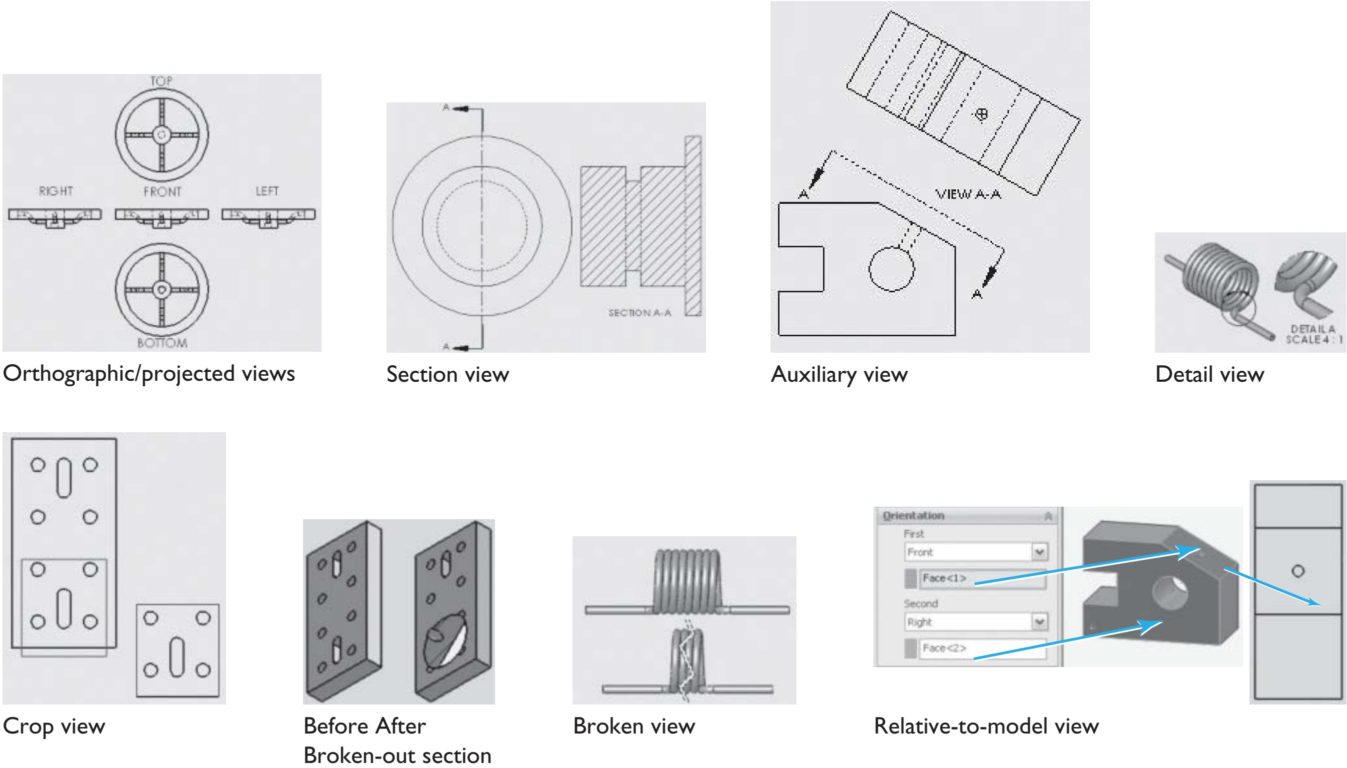


FIGURE 5.11
Types of views

Select a view > Move the mouse in any direction (left, right, top, bottom) of the selected view to insert the new view.

4. Auxiliary view: This view is similar to the projected view, but it is unfolded normal to a selected edge in an existing view. To create an auxiliary view, select an edge, then click **View Layout** tab > **Auxiliary View** (or **Insert** menu **> Drawing view > Auxiliary**) > Click in the screen to place the view.
5. Detail view: A CAD designer uses this view to zoom in to a portion of a view (orthographic, ISO, sectional, assembly, or another detail view) to show more details, usually at an enlarged scale. To create a detail view, click **View Layout** tab > **Detail View** (or **Insert** menu > **Drawing view > Detail**) > Circle the desired portion of the drawing with the mouse > Move (do not drag) the mouse to where to place the detail view and click there.
6. Crop view: You can crop any drawing view except a detail view. A crop view allows you to cut a piece of an existing view. The crop view deletes the view you crop. The creation of a crop view is a two-step process. First, you need to define a closed crop profile such as a circle. Second, create the crop view by clicking **View Layout** tab > **Crop View** (or **Insert > Drawing view > Crop**).
7. Broken-out section: Using a broken-out section, you can carve/remove material from a region of an existing view for a given depth to show inner details. A closed profile, usually a spline, defines the region (broken-out section). Follow this sequence to create a broken-out section: **View Layout** tab > **Broken-out Section** (or **Insert > Drawing view > Broken-out Section**) > Sketch a closed spline > Green check mark to finish. The section will be removed from the drawing view.

8. Broken view: A broken view makes it possible to display a drawing view in a large scale on a smaller size drawing sheet. The view still shows the actual dimension values. We usually break the view in areas where view details are not important to that view. To create a broken view, click **View Layout** tab > **Break** (or **Insert > Drawing view > Break**) > Select a drawing to break > Select the broken view settings (vertical or horizontal break line, gap size, and break line style) from the **Broken View** panel that shows to the left > ✔.
9. Relative-to-model view: This view allows you to create a true view of angled faces in a model. Consider a block with a chamfered corner that has a hole in its center. If you need to create a true view of the chamfer (where the observer is looking perpendicular at the chamfer plane), you need to create a relative view. The relative view requires two orthogonal faces or planes: the chamfer face and the front face of the block. SolidWorks creates a view of the chamfer face perpendicular to the front face of the block. Click this sequence: **Insert > Drawing view > Relative To Model** > Switch to the model that is open or right-click in the graphics pane and open the model > Select the chamfer face > Select the front face > ✔ > Click to place the view in the desired location.

5.10 Sheets

SolidWorks uses the concept of sheets in a drawing. A drawing may consist of one or more sheets. The use of multiple sheets allows you to include more views in a drawing than what one sheet can hold, instead of creating a whole new drawing file. For example, you may use one sheet to show the three standard orthographic and the ISO views, then use another sheet to show a detail view and a section view. In the case of an assembly, you may use one sheet to show the standard views and one sheet to show an exploded view.

To create a new drawing, click **File > New > Drawing > OK** > Select the sheet (drawing) size and format from the window that pops up > **OK**. This opens what is called a *drawing template*. The template consists of both the sheet itself and the sheet format. At the sheet level of the template, the user will find the views and all associated information. After the drawing template is open, the system will prompt you to insert views. After you insert the views, you see a node called *Sheet1* in the features tree shown in the left pane of the screen. You also see a tab at the bottom left of the screen called *Sheet1*. You can rename the sheet. To add another sheet, right-click in the graphics pane > **Add Sheet** from the menu that pops up, or right click the *Sheet1* tab at the bottom left corner of the screen > **Add Sheet**. This adds a new sheet and displays it on the screen, ready to add content to it. SolidWorks creates tabs, one per sheet at the bottom of the screen. You can toggle the sheets by selecting a tab. A very useful feature at the sheet level is to lay out predefined views. Predefined views will automatically determine the orientation, display style, location, and scale of a view in which a part/assembly reference will be added later.

Each drawing sheet has a sheet format. Sheet format determines the size of the drawing, the borders and title block, the default sheet scale, the type of view projection (**First** or **Third Angle**), and the next letter of the alphabet to be used in view datum labels. These can be changed by right-clicking anywhere on the sheet in the graphics pane and selecting **Properties**. The dialog box opens, as shown in Figure 5.12.

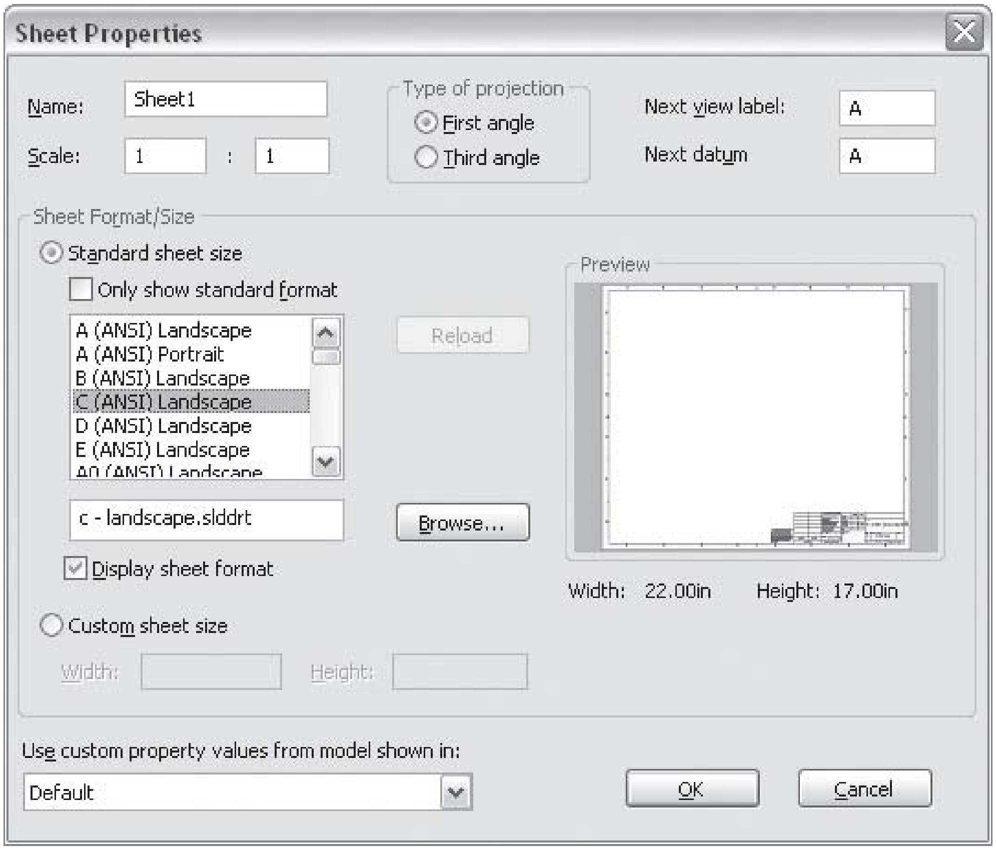


FIGURE 5.12
Sheet properties

5.11 Title Block

A drawing has a generic title block that can also be filled and/or customized. Typically the title block contains **Notes** that are linked to **Custom Properties**. Custom Properties can be linked from the part/assembly level, the sheet format level, or, in the case of special properties, the sheet itself. You can edit the title block and customize it to your needs. To edit the title block, right-click a sheet in the features tree > **Edit Sheet Format** > Double click any box in title block to edit > When done editing, exit the edit mode by clicking the symbol (similar to exit sketch symbol) shown on the top right of the screen to return to the content of the sheet. Tutorial 5-3 offers another method to edit the title block.

Sheet formats have a file extension of *.slddrt* and can be saved. Any custom properties in the title block or changes in the sheet properties will be saved at this level. Sheet formats do not save views. Formats are automatically saved to *SolidWorks install directory\data*, but can be saved anywhere. If the default location is chosen, the new format will appear in the **Templates** window after the user clicks **File > New**; otherwise, the user can browse to find a format.

5.12 Drafting Control

SolidWorks offers many other menus and options for drawings and drafting to control drawing activities. We list some here:

1. Control display styles and hatch: Click **Tools > Options > System Options** tab > **Display Style** or **Area Hatch/Fill** to explore them.

2. Control dimensions display: Click **Tools > Options > Document Properties** tab > **Dimensions** > Select any option to investigate/set.
3. Control font size: Click **Tools > Options > Document Properties** tab > **Dimensions** > **Font** > **Points** > select font size (12, 13) from dropdown > **OK** > **OK.**
4. Control leaders type: Click **Tools > Options > Document Properties** tab > **Annotation** > click **Use bent leaders** checkbox under the **Bent leaders** section.
5. Control arrow size: Click **Tools > Options > Document Properties** tab > Expand **View**s node on left > **Section** > click **Scale with section view arrow letter height** checkbox > **OK.**
6. Control arrow style: Click **Tools > Options > Document Properties** tab > select **ISO** or **ANSI** from the **Base section view standard** dropdown on the right.
7. Control ordinate dimensions display: Click **Tools > Options > Document Properties** tab > expand **Dimensions** node > **Ordinate** > select **ISO** or **ANSI** from the **Base ordinate dimension standard** dropdown on the right.
8. Control dimension line style (bent or straight for a circle radius or diameter): Click dimension in drawing > **Leaders** tab from left pane > select a style from **Custom Text Position** section on left pane > ✔.

5.13 Tolerances

Tolerancing is an important concept to manufacturing. Without tolerances, modern production would not be possible. The ASME tolerancing rules and concepts set the standards for engineering practice. CAD/CAM systems use these rules and concepts in their tolerancing software modules, and your CAD/CAM system should help you follow the ASME dimensions and tolerancing rules. Although we cover tolerances in full detail later in the book, here are some basics. Tolerances are our way of dealing with manufacturing imperfections. We cannot produce what is known as a perfect form of a part. For example, we cannot create a box of size 3 × 2 × 4 inches. Manufacturing imperfections are due to many reasons, such as the skill level of the machinist, accuracy and age of the machine tool, environmental conditions, and cutting conditions.

Designers assign tolerances to dimensions based on the functional requirements of the part and its assembly. A designer does not need to assign tolerances to all the dimensions in a drawing, but only to the important ones. Typically, the designer may designate a general tolerance in the drawing (as a note) to indicate the value of the tolerance that should be applied to any dimension on the drawing that does not have a tolerance associated with it. Conversely, the designer may leave out the general tolerance altogether, and the machinist uses his or her experience.

5.14 Bill of Materials

The bill of materials (BOM) is typically displayed as a table. This can be imported from an Excel file or created at the drawing level by SolidWorks. SolidWorks will automatically create and populate the cells of the table with all of the information relating to a specific assembly. The default BOM has four columns—**ITEM NO.**, **PART NUMBER**, **DESCRIPTION** and **QTY.**—that are each linked to properties found in the assembly itself. Any changes at the assembly level will affect the BOM. The user can add more columns with other linked properties or replace any of the default columns with different information. The order of item numbers reflects the order of appearance in the assembly.

The BOM behaves much like a sheet in an Excel file. The right-click is a very powerful option to find many of the typical table functions such as delete column, insert row, and others. Also, there is an option to split the table in order to fit it according to other items in the drawing. The BOM created can also be saved as an Excel file and then modified using the Excel program.

5.15 Model and Drawing Associativity

The associativity between a model and its drawing links the changes (edits) between the two. If you modify the model (part), its views in a drawing are updated automatically, and vice versa; if you change the model item dimensions in a drawing, the part is updated automatically. Tutorial 5–5 shows an example. The associativity property works only for model item dimensions.

SolidWorks allows you to control this associativity. When you right-click a dimension in a sketch, you can select **Mark For Drawing** to turn on/off this property. This property is on by default. If you turn it off for a dimension, the dimension will not show as a model item in the drawing.

5.16 Design Checker

In design and drafting practice, some companies designate a special person to check engineering drawings; this person is known as a *design checker*. Think of the checker as an intermediary between the design/engineering department and the manufacturing department. The checker typically has both design and manufacturing experience. The checker's tasks are to (1) ensure that the design as documented in the drawing can be produced with the least manufacturing cost, (2) look for inconsistencies in the part dimensioning, (3) check the tolerances and make sure that manufacturing processes can produce them, (4) check the dimensioning scheme according to the concepts and principles covered in this chapter, and (5) ensure that materials are specified. After the design checker reviews the drawing, the checker may either send it back to the design department for corrections, or sign off on it and send it to the manufacturing department for processing.

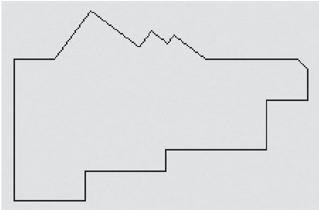
SolidWorks provides an "electronic" design checker for design standards compliance checking. For this to work, you need to capture all the tasks of the "human" checker and codify them in an electronic document as rules that SolidWorks can use to check; effectively you create a checking template with predefined design checks (standards) that SolidWorks uses to assess the drawing. Click **Tools > Design Checker** > **Check Active Document** to activate it. A **Design Checker** tab opens in the task pane to the right of the graphics pane on the screen (**View** > **Task Pane** to see it). Open it and follow the instructions. Close the tab when done to close the checker.

5.17 Tutorials

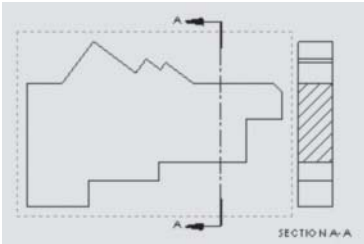
Tutorial 5–1: Create Drawing Views

This tutorial shows how to create the available different views. Drawing views are an important subject in SolidWorks CSWA and CSWP certification exams (see Appendix B).

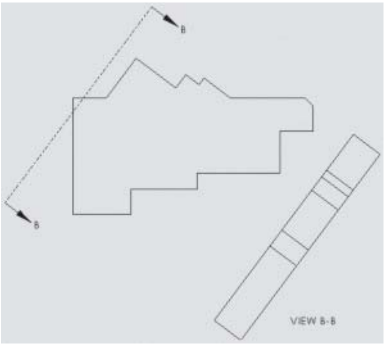
Step 1: Create drawing and insert front view: **Open** > **New** > **Drawing** > **OK** > **OK** (accept the default **Standard sheet size**) > **Browse** (locate and select *example5.1* part) > click the mouse near the bottom left corner of the drawing sheet to insert the front view > ✔.



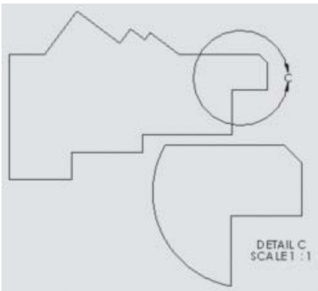
Step 2: Insert section view: **Section View** on **View Layout** tab > move mouse to place cutting line as shown > click mouse to place > ✔ > move mouse to place the section view > click to place the view > ✔.



Step 3: Insert auxiliary view: **Auxiliary View** on **View Layout** tab > select the far left inclined edge > move mouse to place the section view > click to place the view > ✔.

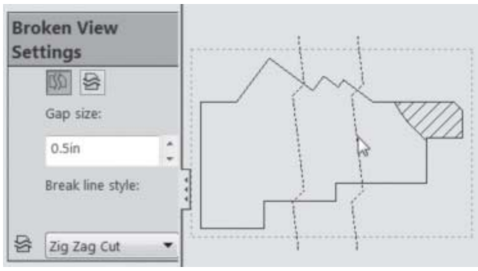


Step 4: Insert detail view: **Detail View** on **View Layout** tab > sketch a circle around the area you want to detail > move mouse to place the section view > click to place the view > ✔.

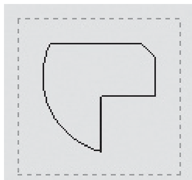


Step 5: Create broken-out section: **Broken-out Section** on **View Layout** tab > the cursor shows a spline symbol waiting for you to sketch > sketch a closed spline around the area of the view you would like to break out to view inside model > enter 0.2 for **Depth D1** > ✔.

Step 6: Create broken view: **Break** on **View Layout** tab > click in view to select it > click mouse to define a break line > move mouse a little and click again to define another break line > ✔.



Step 7: Create crop view: **Circle** on **Sketch** tab > sketch circle around view area you would like to crop > ✔ > **Crop View** on **View Layout** tab.



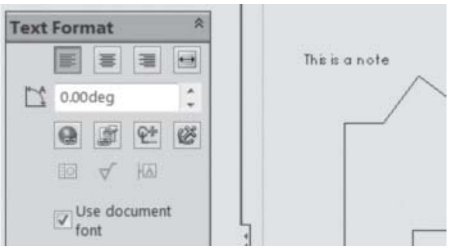
Note: Both the circle and the view must be active (shown in blue on screen) for the crop to occur.

> ***HANDS-ON FOR TUTORIAL 5–1.*** Create a **Relative To Model** view. Use your own parameters to define the view.

Tutorial 5–2: Insert Annotations

Two of the drawing common annotations are notes and labels. A note is a text block that is added to a drawing to clarify an aspect of a drawing such as tolerances, or it could be a set of instructions to manufacture or assemble the part. A label is a note that is attached to a drawing geometry such as edges or holes.

Step 1: Open drawing of Example 5.1: **File** > **Open** > locate Example 5.1 drawing file > **Open.**



Step 2: Add a note: **Insert** menu > **Annotations** > **Note** > click somewhere as shown above > type note text shown above > if needed, use the **formatting** bar that pops up > ✔ > if needed, drag the note around for better placement.

Step 3: Add a label: **Insert** menu > **Annotations** > **Note** > select leader type shown > select edge shown above > click somewhere to place note > type note text shown above > if needed, use the **formatting** bar that pops up > ✔.

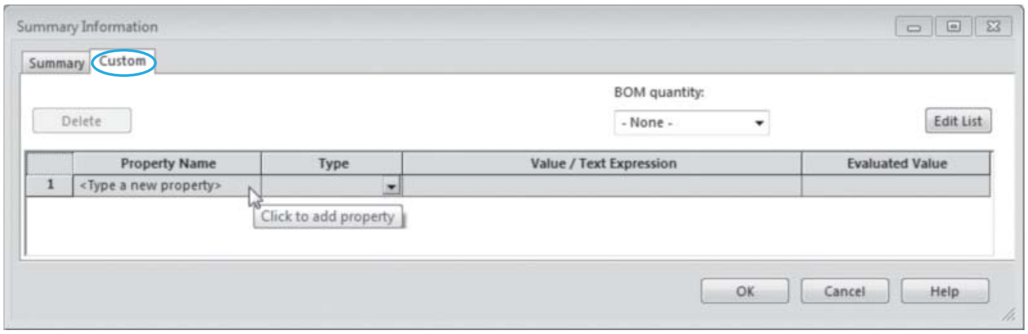
HANDS-ON FOR TUTORIAL 5–2. Add one more note and one more label.

Tutorial 5–3: Fill Title Block

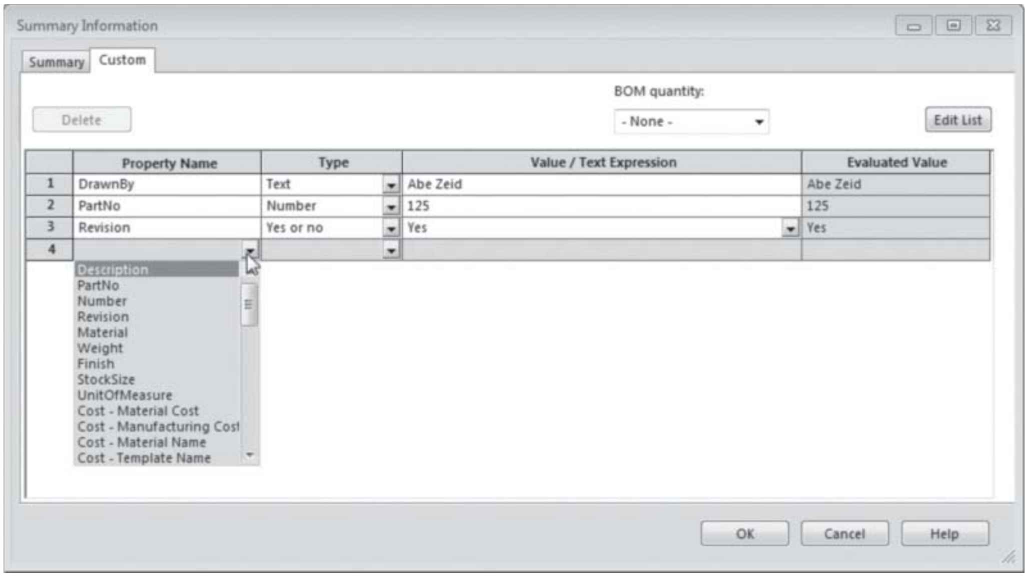
A CAD designer can fill a title block of a drawing once, save the drawing as a master, and use it in the future by updating it. A drawing must be open in order to access the title block fields and fill them.

The title block uses a text file named *properties.txt* that resides in this default location: **C:\Program Data\SolidWorks\SolidWorks2014\lang\english**. If SolidWorks points to another location, the dropdown list of the title block shown in Step 2 below will be missing. To correct this problem, click as follows: **Tools** > **Options** > **System Options** tab > **File Locations** on left > **Custom Property Files** from "**Show folders for:**" dropdown > select the current folder in the **Folders** box > **Delete** > **Add** > navigate to the above folder > **OK**.

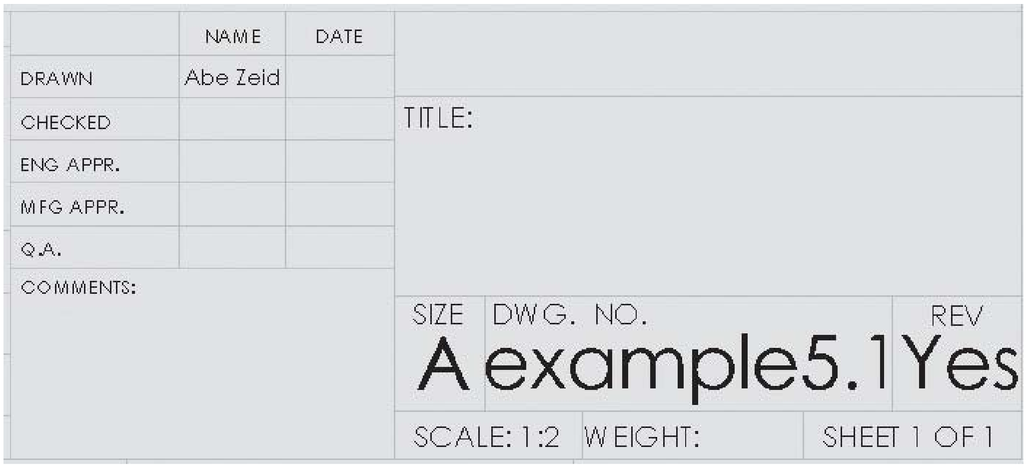
Step 1: Open drawing and access title block: **File** > **Open** > locate Example 5.1 drawing file > **Open** > **File** > **Properties** > click **Custom** tab shown > click as shown to the right to access dropdown list.



Step 2: Fill title block fields: Select a field from dropdown > select its **Type** from third column > type its value in fourth column as shown > repeat by selecting another field from dropdown > **OK.**



Step 3: Review filled title block: The title block should show the input entered in Step 2.



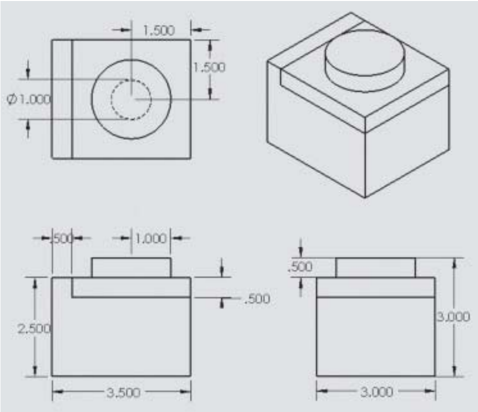
HANDS-ON FOR TUTORIAL 5–3. Modify the title block to add the other data.

Tutorial 5–4: Create Assembly Drawing with Bill of Materials

Assembly drawings, like part drawings, are very important for conveying the information necessary to create the assembly. Assembly drawings are different from part drawings in two aspects:

- ☐ When dimensioning an assembly drawing, we show only the overall dimensions of the assembly, not detailed dimensions of each part in the assembly. Detailed part dimensions can be found in part drawings.
- ☐ An assembly drawing usually has a BOM (bill of materials).

Step 1: Create or open the assembly drawing of Example 1.5 from Chapter 1: **File** > **Open** > locate Example 1.5 assembly drawing file shown below > **Open** > drag views in drawing to move and place them toward the bottom left corner of the drawing to create room to insert BOM.



Step 2: Insert BOM: Click ISO view (or any other view so system knows the assembly) to select it > **Insert** > **Tables** > **Bill of Materials** > ✔ > move mouse to place the BOM table (snap it to top right corner of screen).

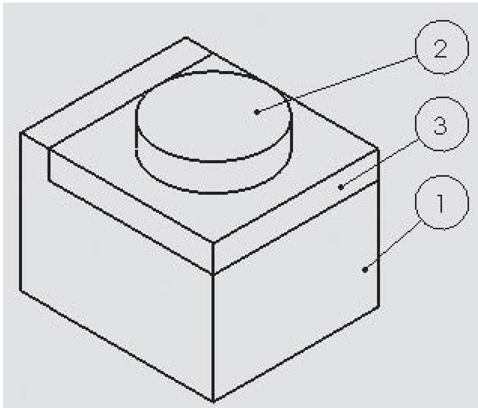
ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	basePlate		1
2	pin		1
3	plate		1

Step 3: Edit BOM: Double click any field under **DESCRIPTION** column > **Keep Link** > fill as shown below > click away from field to close it > repeat to fill other fields > repeat again to change fields under **PART NUMBER** column as shown.

Note: When you hover over the BOM table, you see a handle at top left corner (cross shown below). You can drag the table from the handle and position it where needed.

	A	B	C	D
1	ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
2	1	1	Base Plate	1
3	2	2	Plate	1
4	3	3	Pin	1

Step 4: Insert balloons to number parts on views: **Insert** > **Annotations** > **Balloon** > click a part > click, move mouse, and click again to place balloon > repeat for other parts until done as shown below > ✔.



Note: A balloon uses the **ITEM NO.**, not the **PART NUMBER.**

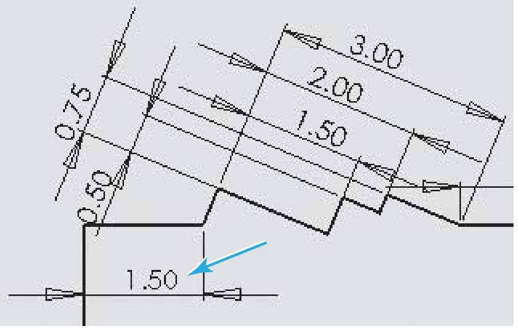
> ***HANDS-ON FOR TUTORIAL 5–4.*** Change the property of the balloon to show the file name of the part instead of the item number.

Tutorial 5–5: Use Model-Drawing Associativity

There are two main types of dimensions: Model Items and Reference dimensions. Model Items are the dimensions used in the part modeling process to create and define the geometry. This type of dimension can modify the geometry at the part level and also at the drawing level. Reference dimensions, however, are typically inserted at the drawing level and have no power to modify the model geometry in any way.

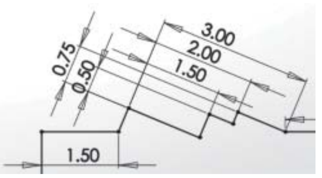
Step 1: Open drawing of Example 5.1: **File** > **Open** > locate Example 5.1 drawing file > **Open**.

Step 2: Change dimensions in drawing: Double click the 1.00 dimension > enter 1.5 in the dimension box that opens up > ✔ > right click view > **Update View** > **File** > **Save** > **Save All** to allow updating the model.

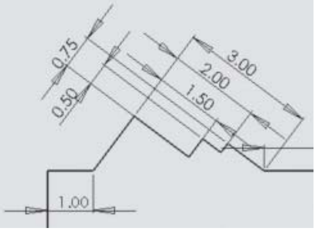


Note: If the new dimension is too large to break the model topology, SolidWorks ignores it.

Step 3: Open model to confirm dimension change: **File** > **Open** > *example5.1*.



Step 4: Change dimension in model: Click *Sketch1* of *Block* **> Edit Sketch >** double click the 1.50 dimension > enter 1.0 in the dimension box that opens up > ✔ > exit sketch > **File** > **Save** > open drawing to see dimension change already there.



problems

1. What are the dimensions of the different drawing sizes (A–E)? You may use SolidWorks to answer this question; open a new drawing to find out.
2. Why is dimensioning a drawing important?
3. Are abbreviations good to use in drawings? Why or why not?
4. What content does a drawing typically hold?
5. What is the problem when your CAD/CAM system gives the opposite view in a drawing than what you expected (e.g., you get the back view instead of the front view)? How do you fix this problem?
6. List the types of views. Describe briefly what each view provides.
7. Why does SolidWorks provide sheets to use in a drawing? List the sheet properties.
8. What are the benefits of creating a drawing template?
9. When do you use a BOM? What does a typical BOM show?
10. How useful is the associativity between a model and its drawings?
11. Select a drawing from the Chapter 2 Problems section, and add some abbreviations to it.
12. Select a drawing from the Chapter 2 Problems section, and change the type of angle of projection from third to first. What happens to the drawing views? Make sure to use the hidden lines display mode to observe the difference.
13. Select a drawing from the Chapter 2 Problems section, and insert two additional views: an auxiliary view and a detail view.
14. Select a drawing from Chapter 2 Problems section, and change its dimensioning to baseline dimension type.
15. Select a drawing from Chapter 2 Problems section, and change its dimensioning to ordinate dimension type.
16. Select a drawing from Chapter 2 Problems section, and add some symbols (abbreviations) to it.
17. Select a drawing from Chapter 2 Problems section, and add some notes, labels, and balloons to it.
18. Select a drawing from Chapter 2 Problems section, and add your custom title block.
19. Add a bill of materials to the Door Valve assembly of Tutorial 1–5 in Chapter 1.
20. Select a drawing from Chapter 2 Problems section, and investigate the associativity between model and drawing.

CHAPTER

6

Assemblies

6.1 Introduction

Products are made up of components (parts) that are assembled together. We manufacture the parts first, and then we assemble them. Similarly, in CAD, we create the parts individually and assemble them to create an assembly model. An **assembly** is a collection of parts positioned and oriented correctly relative to each other. Assemblies can be created by assembling parts, subassemblies, or both. For example, a universal joint has four parts: center block, yoke, pin, and bushing. A car has many parts (such as tires and rims) and subassemblies (such as steering system and engine).

Assembly modeling is different from part modeling in multiple ways. The most distinctive differences are creating mates and the assembly tree. A **mate** is a geometric condition between faces (or other entities) of two parts that allows you to position or orient them correctly in the assembly model. A part features tree lists the features of the part. An assembly tree lists the parts (and subassemblies) and the assembly mates. If you expand the **Mates** node of the tree, you see all the mates you created.

When you create an assembly model, you need to have an assembly plan, similar to having a modeling plan to create a part model. Think of a CAD assembly plan along the lines of a real-life assembly plan. We typically begin creating an assembly model by opening an assembly file. From there we have two alternatives to create the assembly model. We either insert all the parts first, then mate two at a time; or we insert first two parts, assemble them, then insert one part at a time thereafter and assemble it into what has been already assembled. As done with manual assembly, the first part we insert is usually the base component onto which we assemble other components. For example, in a car assembly the base part is the chassis. Note that we use the terms *part* and *component* interchangeably in this chapter. We also use *assembly file* and *assembly model* interchangeably.

Inserting parts into an assembly model uses the concept of instance. An **instance** is a copy of a part that is linked to the part. This linkage allows the CAD/CAM system to maintain the associativity between the instance and its part, much like the associativity between a model and its drawing, which we discussed in Chapter 5. If you change the part, its instance is updated automatically in each assembly that uses it, and vice versa; if you edit or modify the instance in an assembly, its part is updated automatically. You can use one or more instances of the same part in the same assembly. For example, we could use multiple instances of a bolt in a mechanical assembly. Each instance is assembled correctly using the correct mates.

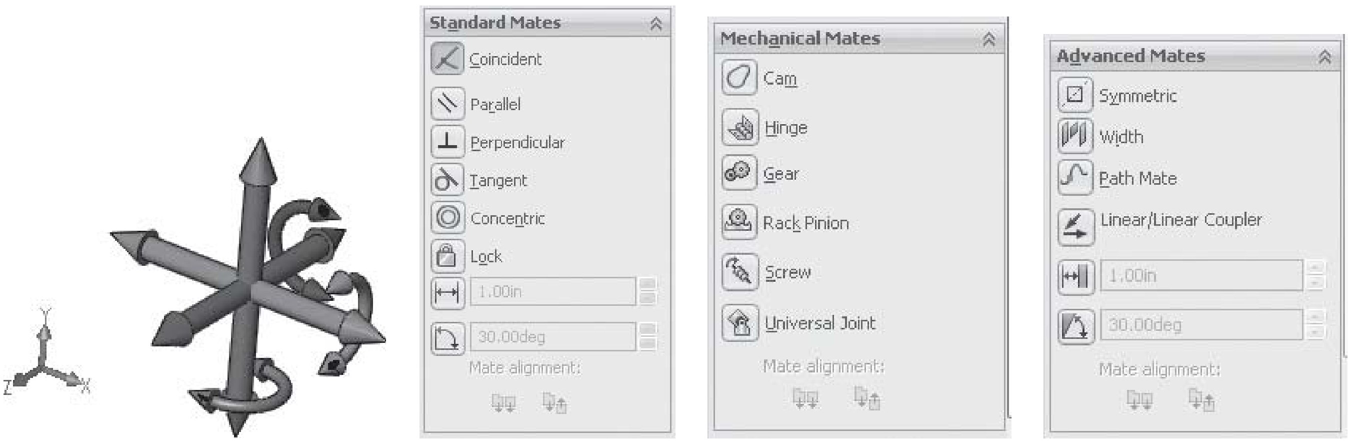
An assembly model is always under defined as shown in the window status bar. That is acceptable for assemblies because not all the parts of the assembly are fixed. The only way to get a fully defined assembly is to fix all the parts of the assembly.

6.2 Assembly Mates

After you insert a component in the assembly model, you assemble it. Assembling it entails applying mates to constrain it in the assembly. The outcome of mating two components is to prevent them from moving relative to each other in the assembly space, thus mimicking real life. An assembled component must be fixed (anchored) in space unless it is free to move for functional requirements (e.g., a motor shaft must rotate). In general, a component needs three mates to lock it in space completely. There is a total of six degrees of freedom (DOF) in the assembly modeling space (as shown in Figure 6.1A): three translations along the three axes (X, Y, and Z) and three rotations around these axes. Three mates are what it takes to constrain the six DOF because locking one locks another.

SolidWorks offers three groups of mates: standard, mechanical, and advanced as shown in Figure 6.1B–6.1D. The important mating concepts are as follows:

- ☐ Why to mate: We mate components to position and orient them correctly relative to each other in their assembly.
- ☐ What to mate: Because any component (part) is a feature, it consists of faces, edges, and vertices. Thus, you mate these elements of the two components you want to assemble. In addition, we can mate nonfeature elements such as center-lines, sketch planes, origins, and planes.
- ☐ How to mate: Mating is about restricting motions of components relative to each other. For example, a concentric mate between a shaft and a hole eliminates two translational DOF and two rotational DOF. A coincident mate between two faces eliminates one translational DOF and two rotational DOF.



(A) Degrees of freedom (B) Standard mates (C) Mechanical mates (D) Advanced mates

FIGURE 6.1
Types of mates

After we assemble components, we need to test them to see whether they are fully constrained in the 3D assembly space. Although we could use the **Move Component** menu, simply drag a component with the mouse and try to move or rotate it. If there is a permissible DOF, the component moves or rotates. If the component is fully fixed, it will not move.

6.3 Bottom-Up Assembly Modeling

Two approaches exist to create assemblies: bottom-up or top-down. The bottom-up approach is more intuitive and therefore more commonly used. This is the traditional approach we used in Chapter 1, and we use it in the tutorials in this chapter. We follow three steps to create assemblies using the bottom-up approach: (1) create the parts, (2) insert them into an assembly model, and (3) use mates to assemble the parts.

6.4 Top-Down Assembly Modeling

The top-down approach is also known as *in-context approach*. It is preferred for conceptual design when a design team is trying to conceive a new product and the layout of its components relative to each other in the assembly. It is a first step to define the design intent of the product (assembly) being designed. The top-down approach is also more efficient to use than the bottom-up for large and complicated assemblies because it reduces errors within the assembly. The top-down approach uses a layout sketch (also known as a *skeleton* or *napkin sketch*) to relate the assembly parts together. Some view the layout sketch as a way to claim space for the components in the assembly because it shows how the components are laid relative to each other. Others view it as a block diagram or a reference sketch of the assembly, establishing relationships and parameters for the parts of the assembly and their dimensions to facilitate their automatic placement in the assembly.

The top-down approach begins with a layout sketch where the designer sketches the skeleton of the parts. The main point about the layout sketch is that it is just that: a sketch that shows both the assembly layout and the main dimensions of the assembly. Use the layout sketch to define the component size, shape, and location within the assembly; make sure that each component references the geometry in the layout sketch. We cannot create more than one layout sketch in the assembly. The plane of the layout sketch is the front plane by default, and we cannot change it. If you select another sketch plane (e.g., right plane) while in the layout sketch, SolidWorks will not permit you to create parts from the geometry you created on this sketch plane. You get the error message shown in Figure 6.2. See Example 6.1 about how to use the layout sketch.

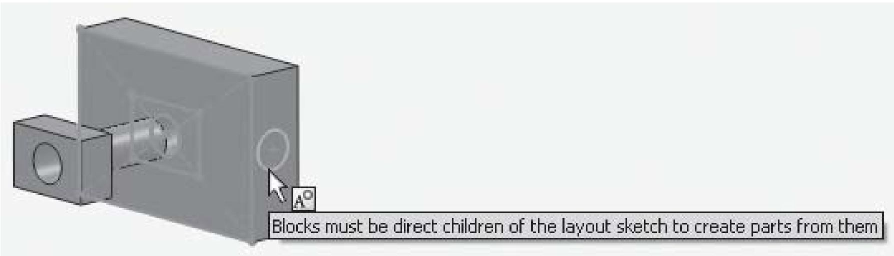


FIGURE 6.2
Error message due to violation of layout sketch plane

You can create and modify the assembly before and after you create components. In addition, you can use the layout sketch to make changes in the assembly at any time. The major advantage of designing an assembly using a layout sketch is that if you change the layout sketch (especially changing locations of components), the assembly and its parts are automatically updated. You can make changes quickly and in just one place. Another advantage to using the layout sketch is that it reduces the parent–child relationships; thus parts are more robust and easier to change and edit.

Creating assemblies using the top-down approach requires the use of the concept of blocks. After you create the sketch entities in the layout sketch, you group them into blocks. Each block defines the sketch entities of a component (part) of the assembly. A block behaves as a single entity. The layout sketch must be active in order to create and edit blocks. After defining a block, use it to make a part. We can save the part either within the assembly file or in its own external file. If we save it within the assembly file, it is referred to as a virtual component (part). Later, we can save virtual components in external files or delete them. The name of a virtual component is the name of its block shown inside square brackets (indicating that it is a virtual component). We may edit the component name to rename it. See Example 6.1 for details.

In the early stages of design when we are still experimenting with it, we recommend using virtual components because of the advantages they offer. We can easily rename a virtual component by changing its name in the assembly features tree (no need to change the name of the external file). Also, we do not clutter the assembly folder with many unused components. Finally, we can make an instance of a virtual component in one step.

How can we use the top-down assembly approach to manage design teams, so that all team members can work concurrently on their parts of the project? We can make parts from the blocks of the layout assembly and save these parts externally. Each team member works on his or her respective part. When team members open the assembly at any time, they see all the latest work.

The layout sketch method just described is used to create entire assemblies via the top-down approach. Another method is the in-context method. Sometimes, we may create new parts/features (known as *in-context features*) in an existing assembly. In this case, the part (component) we create is mated to another existing component in the assembly, and the geometry for the component we build is based upon the existing component. Creating features in context is useful for parts whose geometry depends completely on other parts, for example, brackets and fixtures. Tutorial 6–10 shows how to use this method.

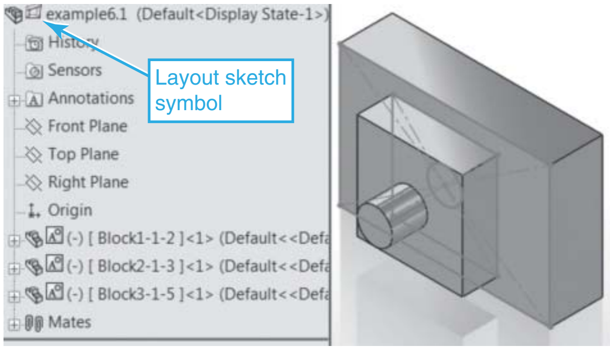
Example 6.1 Create the three-part assembly shown in Figure 6.3 using the top-down assembly approach. All dimensions are in inches.

Solution We create the assembly in three steps: (1) create a layout sketch, and make blocks from the sketch, (2) make parts from the blocks, and (3) create the parts. The details of these steps are shown below. The concept of blocks enables us to separate the layout sketch entities into buckets (blocks); each one defines a sketch for a part.

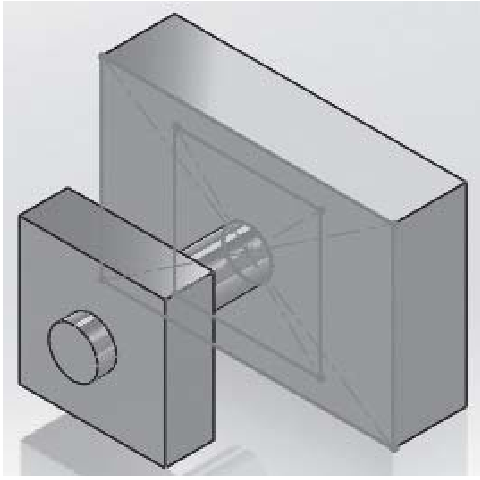
You need to use the following strategy when you create the layout sketch. Create the cross section of each component of the assembly as you would if you were to create it individually as a part. That sketch would be the front cross section of the part. If the component has features that require other sketch planes, you would create them in the component's individual external part file. If you attempt to create them while the layout sketch is active, you get the error message shown in Figure 6.2.

There are some important concepts about a layout sketch that you need to know:

- ☐ Sketching in the layout sketch applies the **On Plane** relation automatically to each entity. Otherwise, the entities are displayed in gray color and are not accessible to dimension. Click an entity and observe the **On Plane** relation under the **Add Relations** section in the left pane.
- ☐ The layout sketch symbol is shown at the top of the tree next to the assembly name as shown in Figure 6.3A. If you close the layout sketch and need to edit it later, right click that symbol > **Layout** from the popup menu. Also, notice that there is no **Delete** in that popup menu. If we have a need to delete layout entities, we must open it first (right click it > **Layout**) then delete.
- ☐ There is one and only one layout sketch in any top-down assembly model (file).
- ☐ Select the sketch layout units before beginning sketching: **Tools** > **Options** > **Document Properties** tab > **Units** > select an option under **Unit system** > **OK**.



(A) Collapsed view

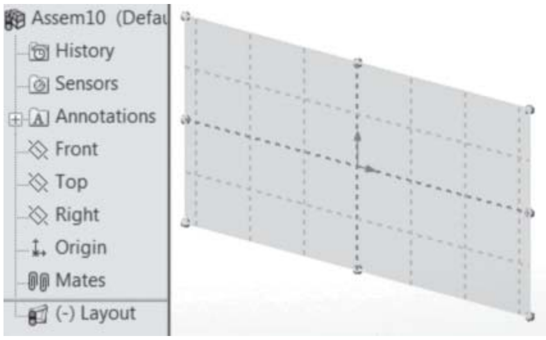


(B) Exploded view

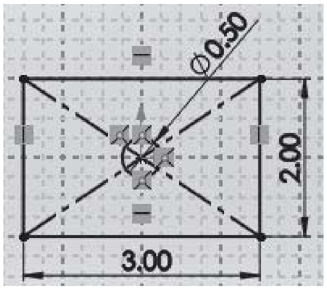
FIGURE 6.3
Three-part assembly

Step 1: Create a new assembly file and set units: **File** > **New** > **Assembly** > **OK** > **Create Layout** button

Note: This step creates the *Layout* node shown below and opens the **Layout** tab.

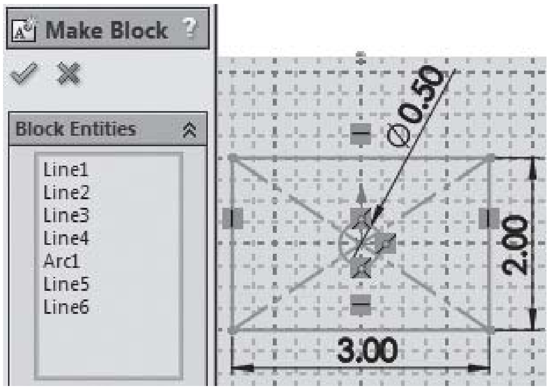


Step 2: Create layout sketch of big rectangle with circle and dimension all: **Center Rectangle** and **Circle** on **Layout** tab > click origin and sketch rectangle and circle and dimension as shown > ✔ > **File** > **Save As** > **Yes** to rebuild assembly > *example6.1* > **Save > Layout** icon on **Layout** tab to open layout sketch to edit it



Note: Click the **Layout** icon to open or close the layout sketch.

Step 3: Make a block from entities created in Step 2: **Make Block** on **Layout** tab > select the rectangle (including its diagonals) and circle created in Step 2 > ✔. This creates *Block1-1* in assembly tree.



Notes:

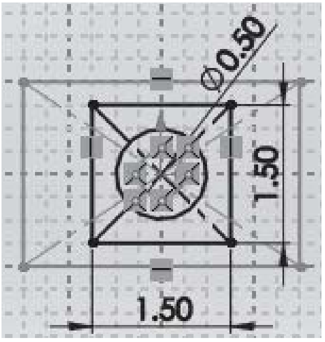
(1) You must select the two diagonal lines because they are part of the center rectangle definition. If you do not select them, the layout sketch becomes over defined and SolidWorks cannot interpret it.

(2) After you make the block, the dimensions disappear from the sketch because they are now part of the block definition. To see and/or edit them, you must edit the block.

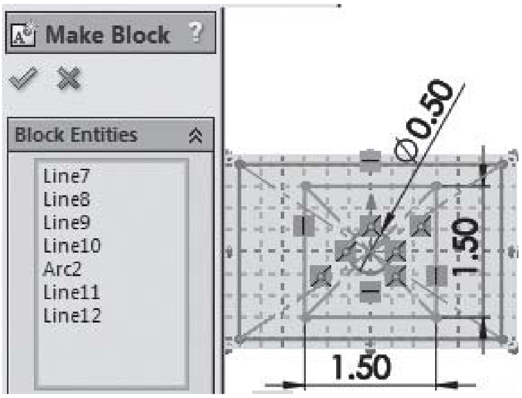
(3) When you edit the block (right click it > **Edit Block**), you are in the block mode. You must exit this mode (as you exit a sketch). To delete the block, right click it > **Delete** from the popup menu. If you delete the block, the block together with all its entities are deleted. Use **Explode Block** to release its entities, but yet delete it.

(4) When you make a block, its entities are displayed faded in a gray color to help separate them from entities that do not belong to a block yet.

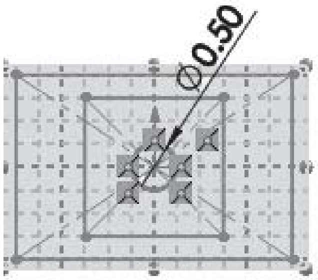
Step 4: Create layout sketch of small rectangle with circle and dimension all: Repeat Step 2 instructions to create geometry shown.



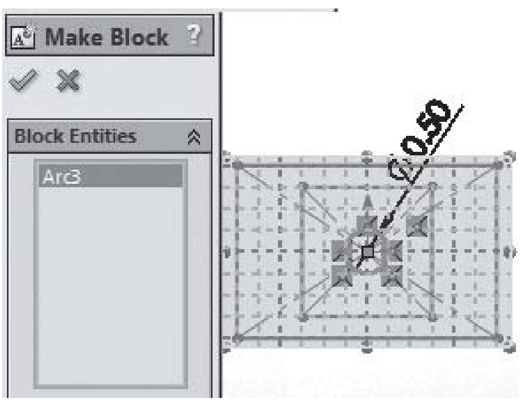
Step 5: Make a block from entities created in Step 4: **Make Block** on **Layout** tab > select all entities created in Step 4 including origin and center rectangle diagonals > ✔. This creates *Block2-1* in assembly tree.



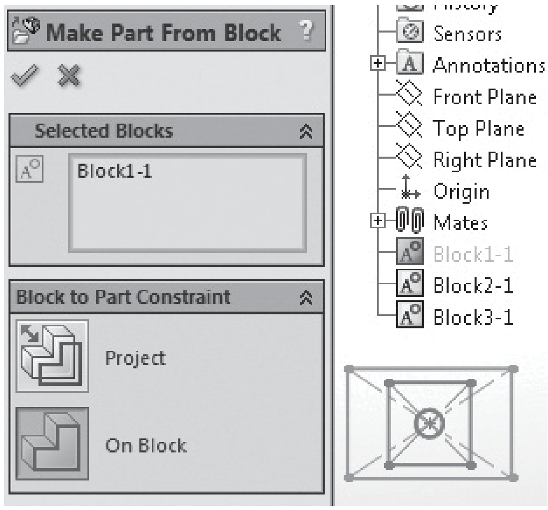
Step 6: Create layout sketch of shaft and dimension all: **Circle** on **Layout** tab > click origin and sketch and dimension circle as shown > ✔.



Step 7: Make a block from circle created in Step 6: **Make Block** on **Layout** tab > select circle created in Step 6 > ✔. This creates *Block3-1* in assembly tree > exit layout sketch.

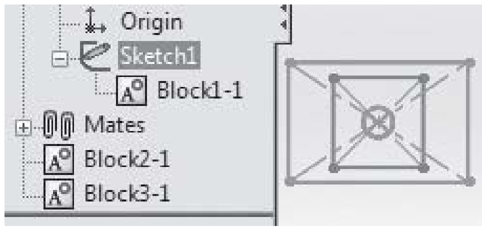


Step 8: Make extruded part from *Block1-1* created in Step 3: **Make Part from Block** on **Layout** tab **>** expand assembly tree > select *Block1-1* node **>** ✔.



Notes:

(1) *Block1-1* disappears from the features tree and a virtual component appears in the tree. If you expand the component node, you see *Block1-1* as a child of *Sketch1* of the component as shown here.



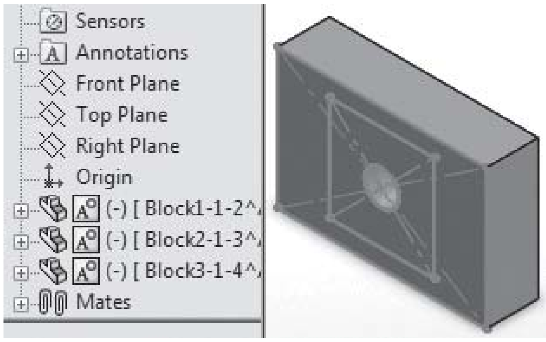
(2) *Block1-1* defines the component (part) sketch. Note that the part name resembles the block name and is shown inside square brackets to indicate that it is a virtual part. At this point, you do not see any difference in the visual appearance of the assembly on the screen.

(3) Click the virtual part node in the features tree to invoke its context toolbar shown here. Hover over each item of the menu and investigate it to learn more. The first icon (**Open Part**) of the menu enables you to open the part and create it. When you open the part, you see a sketch of it that you use to create the part.

(4) Right click the part node in the features tree and investigate the items of the popup menu. The **Save Part (in External File)** does just that. It converts the virtual part into a real part.

Step 9: Make extruded parts from *Block2-1* and *Block3-1*: Repeat Step 8.

Step 10: Create big block part: Virtual part *Block1-1* node from the assembly features tree > **Open Part** (from context toolbar that pops up) > select *Sketch1* from part features tree > **Extruded Boss/Base** on **Features** tab > reverse the extrusion direction > enter 0.75 for distance **D1** > ✔ > **File** > **Save** > **File** > **Close** > **Yes** to update the assembly.

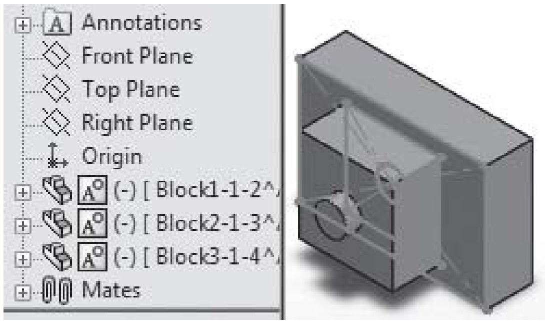


Notes:

(1) You have created the big block virtual part. When you save it, you save it inside the assembly file. You do not see a file name unless you save it externally as discussed in Step 8.

(2) When you return to the assembly file (still open), you see the newly created big block part. Change the view to ISO for a better view (shown here). As shown, the other blocks are visible and will be used to create the other parts.

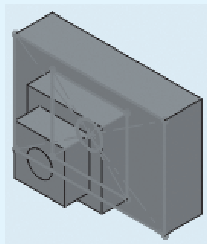
Step 11: Create small block and shaft parts: Repeat Step 10. Use 0.75 extrusion distance **D1** for small block and 2.0 for shaft. Use **Mid Plane** extrusion for shaft.



Note: Do not reverse extrusion direct for small block; otherwise, big block swallows small block. The final three-part assembly is shown here.

HANDS-ON FOR EXAMPLE 6.1. Add a 1.0 × 1.0 × 0.5 inch extrusion with 0.5 inch hole to the assembly, as shown.

Hint: Can you create the part in the layout sketch? If you cannot, create the part separately, and then use the bottom-up approach to assemble it.



6.5 Assembly Tree

Like a part model, an assembly model has an assembly tree. SolidWorks calls it **Features-Manager Design Tree**, the same name as for individual parts. We call it *assembly tree*. An assembly tree is different from a features tree in two ways. **First**, the nodes of an assembly tree are parts/subassemblies and mates. There are as many part nodes in the tree as there are assembly parts (instances). However, there is only one **Mates** node. Each parts node shows a symbol to the left of its name indicating whether it is fixed in space (f) or floating (-). To change a part's state in the assembly, right click it and select **Fix** or **Float.** Typically, we have one (f) node in the assembly, indicating that the corresponding part is the base. **Second**, we cannot roll back an assembly tree. We can suppress parts in the assembly if needed. Right click a part > **Suppress** from the menu that appears.

The top node (root) is the assembly model under which the other nodes come. Each part node has leaves, which are the part mates and features. The tree shows the steps (history) of inserting parts into the assembly. The tree also shows the steps (history) of creating mates in the assembly.

If we right-click any assembly tree node, we can perform useful functions. We can delete a part or mate by right-clicking its node in the tree > **Delete** from the popup. Investigate the many possibilities of using the assembly tree. Here are some more thoughts on how to use the tree:

1. Expand or collapse nodes to have access to or hide sketches of features. A node that is collapsible has a "+" (collapsed state) or "–" (expanded state) symbol.
2. Rename instance nodes. This is not as useful as in parts trees. SolidWorks appends the new name to the part name because it must maintain the relationship between the parent part and its instance in the assembly.
3. Rearrange its nodes: drag a node and move it up or down the tree in case you want to change the order of inserting parts.
4. Suppress a node: you can hide (suppress) a part (node) without having to delete it and re-insert and mate it again. This could be useful in debugging assemblies. Suppressed nodes are displayed gray in the assembly tree. To suppress a node: click it in the tree > **Suppress** from the context toolbar (hover over until you read it). To unsuppress it, click it again > **Unsuppress** from context toolbar.

6.6 Assembly Drawing

Assembly drawings are similar to part drawings in content. However, there are some differences. Assembly drawings use BOMs and balloons to fully document the assembly. When we dimension a drawing, we show the overall dimensions instead of every dimension (which we can access from a part drawing). Consult Chapter 1 on creating an assembly drawing.

6.7 Assembly Exploded View and Animation

An exploded view of an assembly is a view in which the components of the assembly are moved along the axes of the assembly modeling space. We usually explode the ISO view. SolidWorks creates display states for the assembly. You can show the assembly ISO view in a collapsed state where all the components are in assembled positions or in an exploded state where components are shown in moved positions. SolidWorks shows the exploded view in its **ConfigurationManager** tab in the left pane; hover over it to display its name.

After we create the exploded view, we can animate it as shown in Chapter 1 tutorials. The animation basically displays the creation steps of the exploded view in a continuous (animation) fashion on the screen. SolidWorks provides the **Animation Controller** window to help control the animation while it is running. Consult Chapter 1 tutorials.

Chapter 1 tutorials cover the basics of creating an assembly exploded view, and we extend this coverage here by discussing more on assembly configurations. An assembly configuration holds an assembly state. For example, we can define different settings (such as hide/show, display mode, color, texture, transparency, etc.) for assembly components and save them in different display states (configurations).

We can create new configurations or edit existing ones. Click this sequence to create one: **ConfigurationManager** tab > Right-click anywhere in the tab pane and select **Add**

Configuration. To edit an existing configuration of an exploded view, edit its components one at a time. Click this sequence to edit a component: Right-click the **Exploded Step** x (x is the step number) node of the component from the **ConfigurationManager** tab > **Edit Feature**.

6.8 Assembly Motion Study

Assembly motion study enables us to check the assembly motion. SolidWorks provides two types of motion for an assembly: Animation and Basic Motion. Animation is a motion driven by key points and constrained by assembly mates. Basic Motion is a more realistic simulation using assembly mates, springs, gravity, and motors. We cover the Animation method in this chapter.

When we open an assembly file, there is a default **Motion Study 1** tab shown in the bottom left corner of the screen, as shown in Figure 6.4. The tab shows the two types of motion. If you need more motion studies, right click **Motion Study 1** tab > Create **New Motion Study**. We can delete the motion tabs we create (Right click any tab > **Delete** from the popup menu), but we cannot delete the default **Motion Study 1** tab (although we can rename it; double click it or right click it).

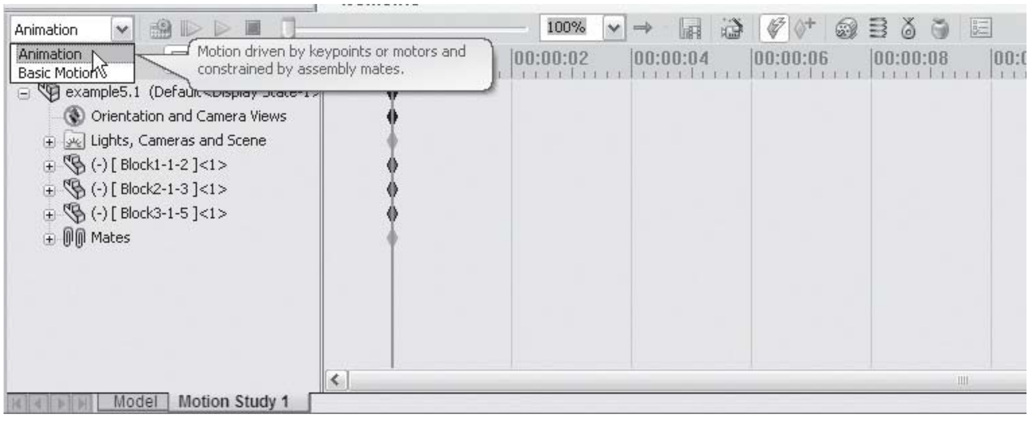


FIGURE 6.4
Assembly motion study

We create motion study–specific mates to perform a motion study. The motion study–specific mates are different from and independent of the assembly model mates that we use to assemble components in the assembly. We can easily see that the motion study–specific mates are different from the assembly mates by viewing the **Mates** node in the assembly tree in the assembly model. Using motion study–specific mates allows us to create multiple motion studies to analyze the assembly motion with different mates without changing the assembly model.

The motion study–specific mates allow us to restrict motion between components for motion studies. For example, we can create distance and angle mates, and change their values in an animation.

6.9 Interference and Collision Detection

Interference and collision detections help check whether assemblies are created correctly and whether the dimensions of their parts are correct relative to each other. Interference checks whether the parts overlap each other (static interference), whereas collision checks whether parts collide while they move (dynamic interference). Thus, interference analyzes static (still) parts, whereas collision analyzes dynamic (moving) parts.

We use interference detection to find static interference between assembly components. This detection option takes a list of components and finds interferences between them. It shows graphically the paired components with their interference (overlapping) regions shaded in a designated color, usually red. We may also change the display settings of interfering and noninterfering components to see the interference better. We may also select to ignore intentional interferences such as press fits and threads. We can choose to distinguish between coincidence (from coincidence mate) and true interferences. Correcting interferences could be achieved by changing the dimensions of the interfering components or by filleting or chamfering some of their edges.

We use collision detection to check when faces of components clash or collide during motion. If they do, we hear an audible sound, and the components stop moving, thus preventing parts from penetrating each other. We move the parts with a mouse by dragging them. We have the options of stopping the motion upon collision, highlighting the colliding faces, and generating a system sound.

6.10 Assembly Design Tables

Design tables are used in assemblies in a similar fashion to their use in parts. A **design table** is an Excel spreadsheet that is used to create multiple configurations in a part or assembly. We can use design tables to control configurations of parts, mates, as well as distance and angle relationships between parts. Follow these steps to deal with design tables: create them, modify them, and test them.

6.11 Tutorials

The theme for the tutorials in this chapter is to practice creating and using assemblies. Some tutorials use off-the-shelf components to mimic what CAD designers do in practice. We use Web resources to download the parts we need for each tutorial.

Tutorial 6–1: Create Cam and Follower Assembly

We create an assembly that uses the Pin Block assembly from Tutorial 3–4 in Chapter 3 to follow a cam created in this tutorial. Using the Pin Block assembly illustrates the concept of using subassemblies in an assembly, a common practice in industry. The cam must be extruded from a fully closed loop consisting of tangent arcs (or splines). Figure 6.5 shows the assembly and the cam. All dimensions are in inches.

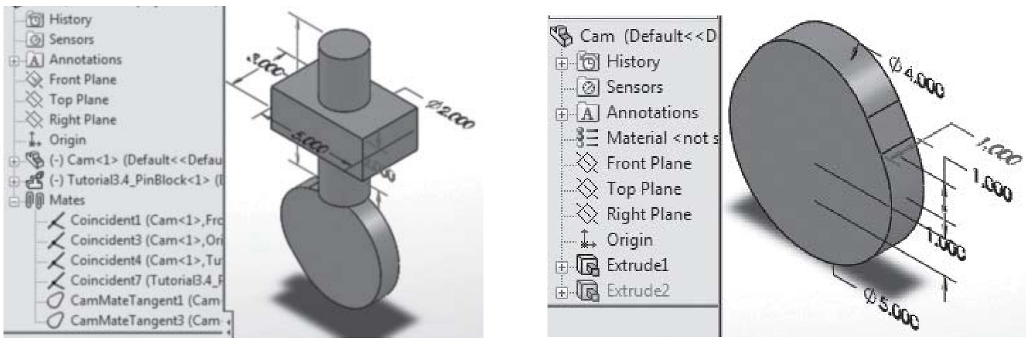
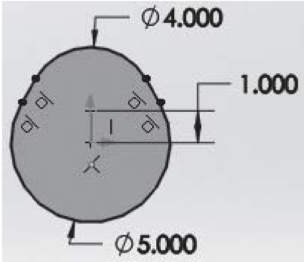


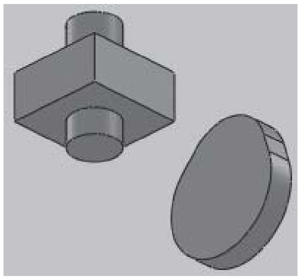
FIGURE 6.5
Cam-follower assembly

(A) Assembly (B) Cam

Step 1: Create cam sketch and part: **File > New > Part > OK > Front Plane > Extruded Boss/Base** on **Features** tab > **3 Point Arc** on **Sketch** tab > sketch top and bottom arcs as shown > **Line** on **Sketch** tab > sketch two lines to connect the arcs > select one line + **Ctrl** key + corresponding arc end > tangent from **Add Relations** section > ✔ > repeat three times for other ends > dimensions as shown > exit sketch > reverse extrusion direction > enter 1 for thickness **D1 > Mid Plane > File > Save As >** *cam* **> Save.**



Step 2: Create assembly: **File > New > Assembly > OK > Browse** (button on left pane of screen) > locate cam file and double click it to insert it > locate Pin Block assembly file and double click it to insert > **File > Save As >** *tutorial6.1* **> Save.**



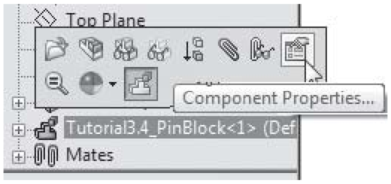
Note: Float both components to permit all DOF.

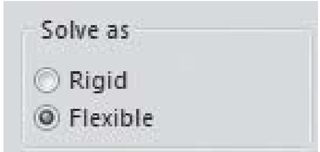
Step 3: Assemble cam: **View > Temporary axes** (to show cam axes of its two arcs as we need them) > **Mate** on **Assembly** tab > **Coincident** mate between the cam front plane and the assembly front plane (expand assembly tree and select from it) > **Coincident** mate again between the assembly origin and the axis of the larger end (arc) of cam (expand assembly tree and select from it).

Note: Select the cam axis at about its midpoint to place assembly origin there.

Step 4: Assemble pin block subassembly: > **Mate** on **Assembly** tab > **Coincident** mate between the axis of the hole in the block and the origin of the assembly > **Coincident** mate between the top plane of the Pin Block assembly and the front plane of the cam.

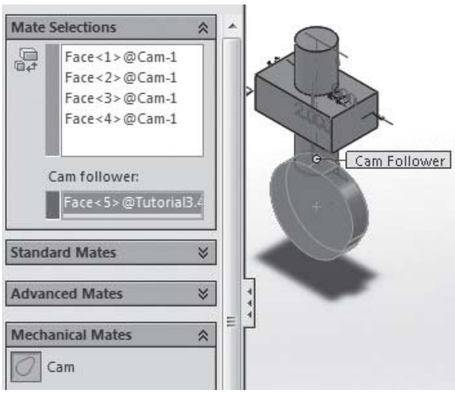
Step 5: Make subassembly flexible: Click pin block node in assembly tree > **ComponentProperties** from context toolbar that opens as shown > select **Flexible** as shown > **OK.**





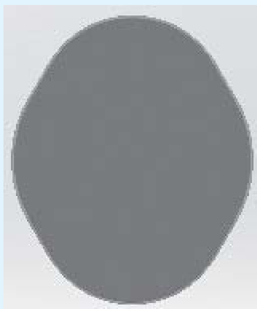
Note: Making subassembly flexible breaks it up into its individual components, allowing us to move the pin independent of the block. This makes the pin (follower) follow cam profile during cam rotation.

Step 6: Use cam mate between cam and follower: **Mate** on the **Assembly** tab > **Mechanical Mates > Cam >** Right-click one of cam side faces > **Select Tangency**. This will automatically select all cam side faces and fill in the top box as shown > click the **Cam follower** box > select the bottom of the pin as shown.



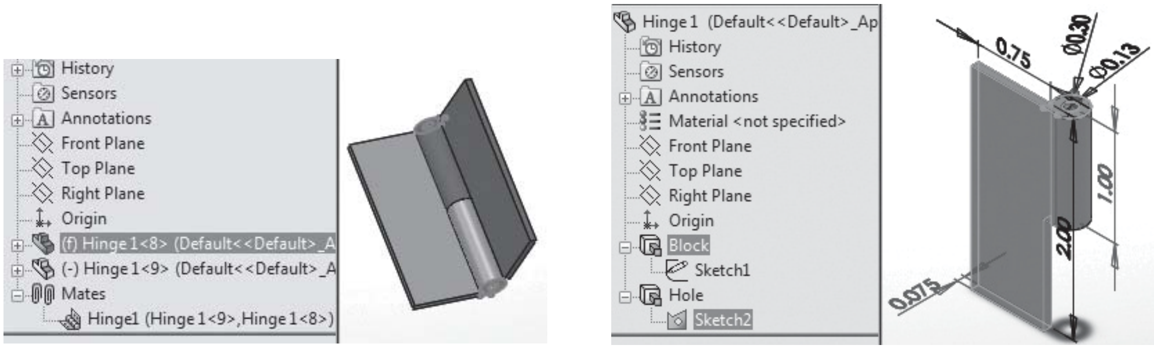
Step 7: Test the cam follower motion: Drag the cam with the mouse and rotate it. Notice that the follower oscillates as the cam rotates.

HANDS-ON FOR TUTORIAL 6–1. Create a second configuration of the cam that has two lobes that are 180° out of phase from each other as shown. Test the cam in the assembly by dragging and rotating it. The follower must follow the cam profile. Note: Create an axis (centerline) at the center point of the new cam and mate it with the assembly origin to allow the cam to rotate correctly a full 360 degrees.



Tutorial 6–2: Create Working Hinge Assembly

Create a working hinge that opens and closes with limits to simulate real life. Figure 6.6 shows the assembly. The hinge mate we use in this tutorial is equivalent to using a concentric mate and a coincident mate. The advantage, however, of using this mechanical mate is that it attaches limits to the travel of the hinge as well as using only one mate as opposed to two.

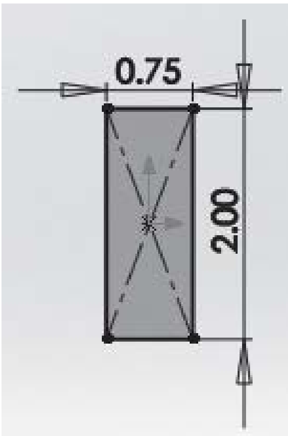


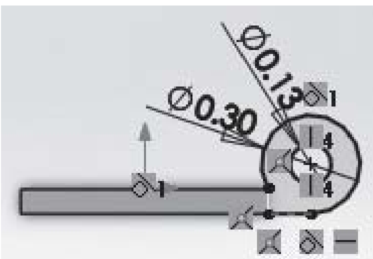
(A) Assembly

(B) Hinge

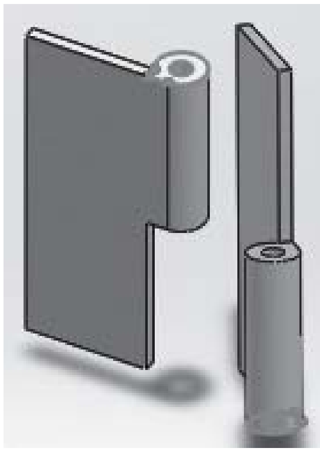
FIGURE 6.6
Hinge assembly

Step 1: Create hinge *Sketch1* and *Sketch2* and features: **File > New > Part > OK > Front Plane > Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > enter 0.075 for thickness **D1** > reverse extrusion direction > ✔ > select top face of extrusion as sketch plane > **Extruded Boss/Base** on **Features** tab > sketch and dimension as shown exit sketch > enter 1 for thickness **D1** > ✔ > **File > Save As** > *tutorial6.2* > **Save.**



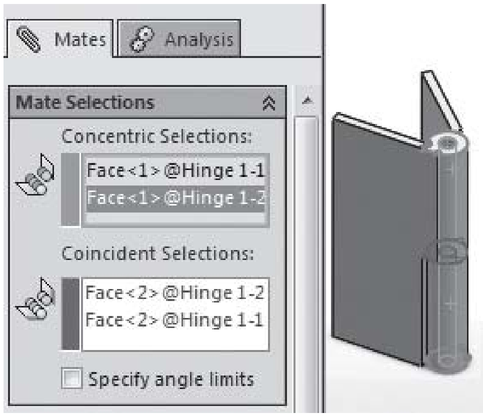


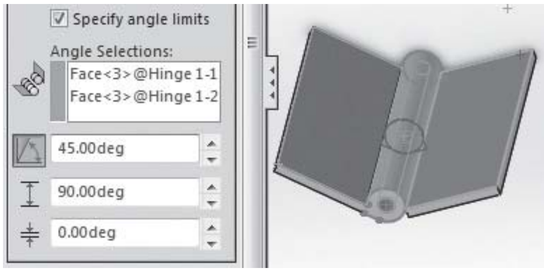
Step 2: Insert and orient instances: **File > New > Assembly > OK > Browse** (button on left pane of screen) > locate *hinge* file and double click it to insert two instances of it > **File > Save As >** *tutorial6.2* > **Save > Rotate Component** from **Move Component** dropdown on **Assembly** tab > rotate and orient one instance as shown.



Note: When you insert the two instances, they both are vertical as the left instance shown here. When you assemble, the two instances are displayed on top of each other. This is why we rotate them as shown.

Step 3: Assemble hinge: **Mate** on **Assembly** tab > **Mechanical Mates > Hinge** > select two holes (or two outer cylindrical faces) for **Concentric Selections** as shown > select two end planar faces of the cylindrical parts that should butt each other for **Coincident Selections** as shown > ✔.





Note: The first value, 45°, is the current opening shown. The second value, 90°, is the maximum opening. The third value, 0°, is the minimum opening.

Step 4: Set the limits of the hinge mate: Check off **Specify angle limits** > select the two internal flat faces of the two hinge instances as shown > enter 90° for angle limit as shown.

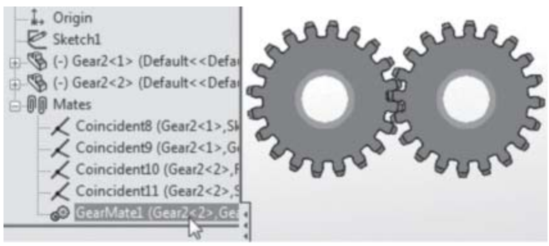
Step 5: Test the hinge motion: Drag the unfixed plate of the hinge with the mouse and rotate it. Observe that the hinge obeys the angle limits set in Step 4.

HANDS-ON FOR TUTORIAL 6–2. Create and assemble the hinge pin, and change the limits of the hinge to allow a range of motion of zero to 270°.

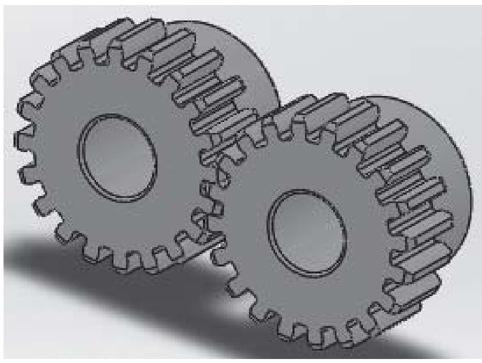
Tutorial 6–3: Mate Two Gears with Gear Mate

The gear mate can be used to mate any circular shapes such that they will rotate relative to one another. For this tutorial, a gear is downloaded from the Boston Gear website and used in an assembly. However, the gear mate does not require teeth to use it. We use a gear mate to relate the rotational ratio of two downloaded gears. Figure 6.7 shows the assembly. The gears must spin in the correct direction and with the correct ratio so as to simulate real life.

The necessary parts for this tutorial will be downloaded from the Boston Gear website (Courtesy of Boston Gear LLC, a subsidiary of Altra Industrial Motion, Inc.). The necessary mates will be added to the assembly to create a functional relationship between the gears.



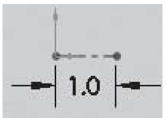
(A) Assembly



(B) Gear

FIGURE 6.7
Gear assembly

Step 1: Create a sketch with two anchor points to locate the gears: **File > New > Assembly > OK > Front Plane > Centerline** on **Sketch** tab > click origin and sketch and dimension a horizontal line as shown > exit sketch > **File > Save As** > *tutorial6.3* > **Save**



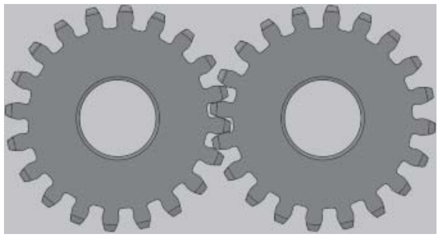
Note: For simplicity, we will place the gears at the endpoints of the line, thus avoiding creating shafts to mount the gears on.

Step 2: Download the gear: Visit www.bostongear.com to register for free and download part# *na20b* from the 3D CAD library as shown here.

Step 3: Insert the gear instances: **Insert Component** on **Assembly** tab > **Browse** (button on left pane of screen) > locate the gear part file and double click it to insert two instances of it.

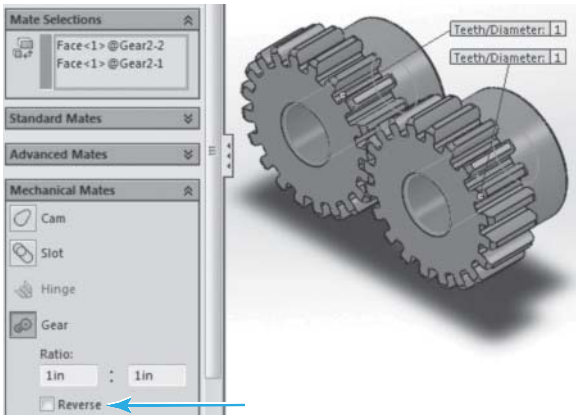
Note: If you open the downloaded gear, SolidWorks asks you if you would like to recognize its features. Ignore it (select **No**) unless you plan to edit the gear. In such a case, the gear part has only one node named *Imported1* in the assembly tree. Expand the nodes of the two gear instances in the assembly tree to observe this fact.

Step 4: Assemble the gears: **Mate** on **Assembly** tab > **Coincident** > front plane of assembly > front face of one gear instance > **Concentric** > hole face of one gear instance > one endpoint of centerline > repeat to mate other gear instance > ✔.



Note: SolidWorks does not use the teeth to mesh the gears. Thus you need to align the teeth manually before creating the gear mate to properly visualize the gear meshing.

Step 5: Create the gear mate: **Mates** on **Assembly** tab **> Mechanical Mates > Gear** > select the faces of the holes of the two gear instances > ✔.



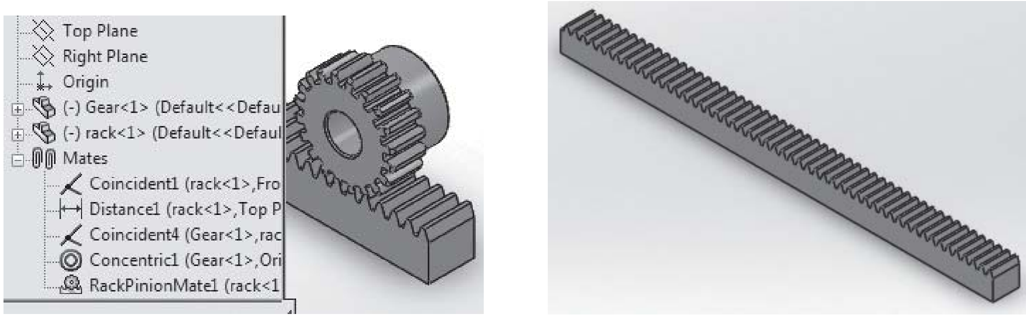
Note: SolidWorks creates the gear ratio automatically based on the diameters of the gears.

Step 6: Test the gear motion: Drag the face of one of the gears with the mouse and rotate it. If the rotations of the gears are opposite to each other, edit the gear mate (expand **Mates** node > click *GearMate1* > **Edit Feature** from context toolbar) and click **Reverse** checkbox shown.

HANDS-ON FOR TUTORIAL 6–3. Change the gear ratio from 0.38:0.38 to 1.00:0.38. Do the teeth mesh correctly when you rotate the gears with the mouse?

Tutorial 6–4: Create Functional Rack and Pinion

We use a rack and pinion mate to relate the rotation of a spur gear to the travel on a linear gear rack. Figure 6.8 shows the assembly. The rack pinion mate can be used to mate any cylindrical face to any linear face such that they move relative to one another. They



(A) Assembly

(B) Rack

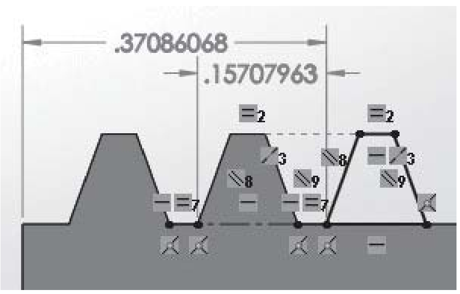
FIGURE 6.8
Rack and pinion assembly

do not need teeth for a mate. The necessary parts for this tutorial are downloaded from the Boston Gear website (Courtesy of Boston Gear LLC, a subsidiary of Altra Industrial Motion, Inc.). The necessary mates are added to the assembly to create a functional relationship between the rack and pinion. The spur gear (pinion) must spin in the correct direction to relate to the travel of the rack gear (rack) so as to simulate real life.

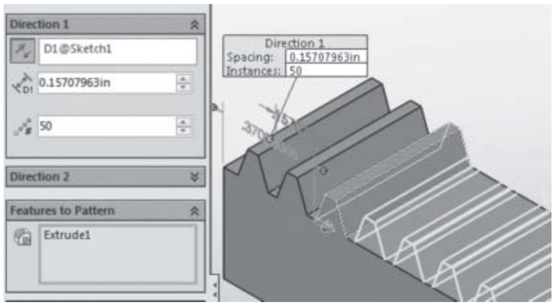
The downloaded rack is long and has only two teeth. When we open its file, we do not use features recognition, so we import it as one body resulting in one node, *Imported1*, in the features tree. We shorten the rack length and create one additional tooth, and pattern it to create a functional rack.

Step 1: Download the rack: Visit www.bostongear.com to register for free and download part# *L2020–4* from the 3D CAD library as shown here.

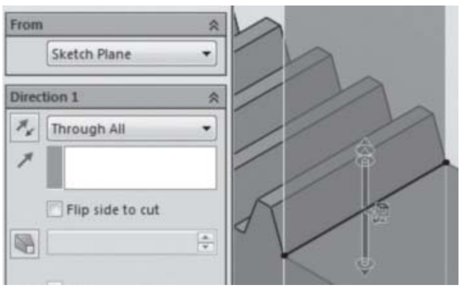
Step 2: Add one tooth to rack: **File > Open >** *rack* **> Front Plane > Extruded Boss/Base** on **Features** tab > sketch tooth profile as shown > exit sketch > **Up to Surface** from dropdown > select back face of rack > ✔.



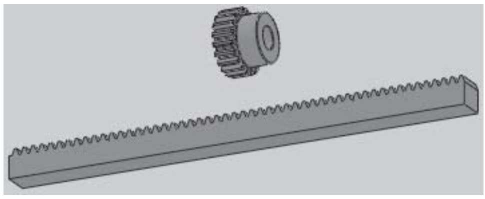
Step 3: Create teeth pattern: Select extrusion from Step 2 **> Linear Pattern** on **Features** tab > select rack top edge for **Direction 1** > enter 0.15707963 for **D1** > enter 50 for number of teeth > ✔.



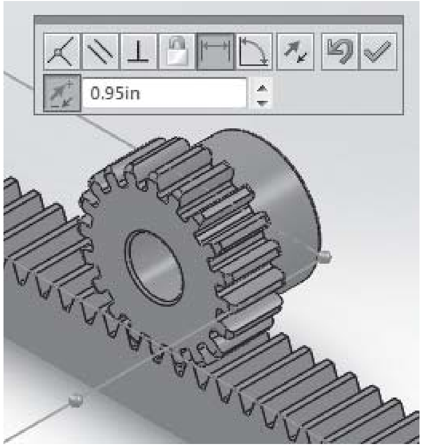
Step 4: Shorten rack length: Select top face of rack beyond the teeth created in Step 3 **> Convert Entities** on **Sketch** tab > bottom edge of last tooth of the pattern created in Step 3 > exit sketch > **Extruded Cut** on **Features** tab > **Through All** > ✔ > **File > Save**.



Step 5: Insert the spur gear and rack: **File > New > Assembly > OK > Browse** (button on left pane of screen) > locate and insert the L2020–4 gear rack along with the na20b spur gear from Tutorial 6.3 as shown > **File > Save As >** *tutorial6.4*.

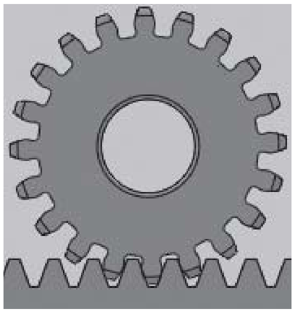


Step 6: Assemble rack: **Mate** on **Assembly** tab > **Coincident** > front plane of assembly > front plane of rack > ✔ > distance mate between top plane of assembly and bottom face of rack > enter 0.95 for distance as shown > ✔.

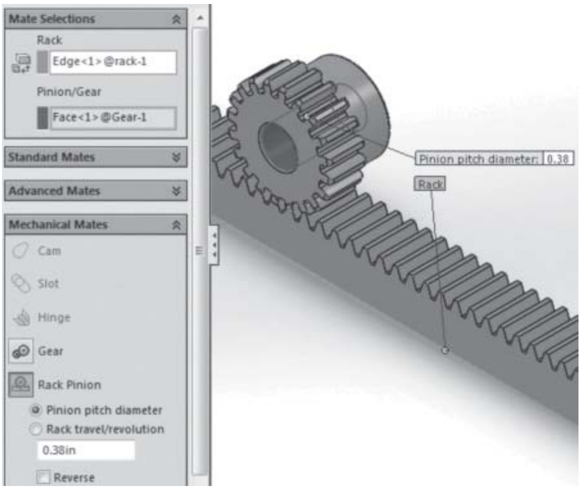


Step 7: Assemble pinion: **Mate** on **Assembly** tab > **Coincident >** front face of pinion > front face of rack > **Concentric** > pinion hole (bore) > assembly origin > ✔.

Step 8: Set rack and pinion for correct visualization: Rotate the pinion with mouse until it looks like the teeth are correctly meshed as shown. The teeth of the gear have no impact on how the parts behave relative to each other in the assembly. Therefore, we manually set the pinion for visual demonstration only.

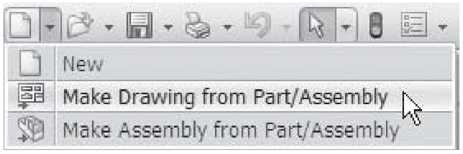


Step 9: Create rack pinion mate: **Mates** on **Assembly** tab **> Mechanical Mates > Rack Pinion** > select rack bottom edge as shown > select pinion hole face as shown > ✔.

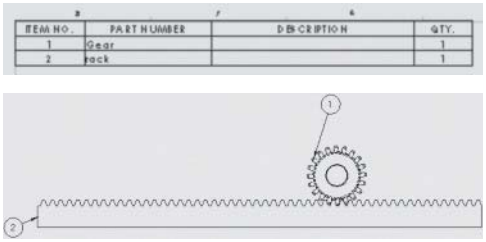


Note: SolidWorks automatically creates **Rack travel/revolution** shown based on the dimensions of the assembly components. You can change the ratio.

Step 10: Create assembly drawing: **New** dropdown from menu bar as shown **> Make Drawing from Part/Assembly.**



Step 11: Add BOM and add balloons: Select the front view from drawing > **Insert > Tables > Bill of Materials >** ✔ > position BOM at the top right corner of the drawing as shown > select **front** view again > **Annotation > Auto Balloon >** ✔.

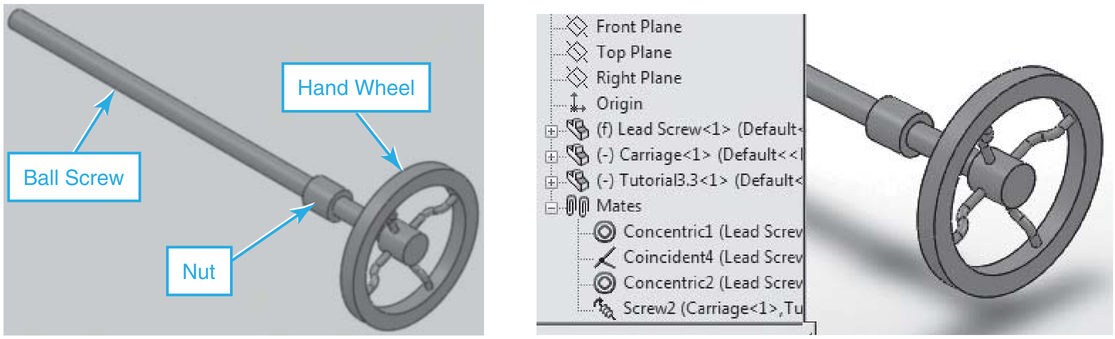


ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	Gear		1
2	rack		1

> ***HANDS-ON FOR TUTORIAL 6–4.*** If you observe this rack and pinion system, you will notice that the teeth do not mesh correctly while moving the parts with the mouse. Why do they not mesh correctly? Correct the pinion pitch diameter (**Rack travel/revolution** shown in Step 9) such that the teeth mesh correctly.
>
> **Hint:** Visit the website to find the correct pitch diameter value.

Tutorial 6–5: Create a Functional Ball Screw

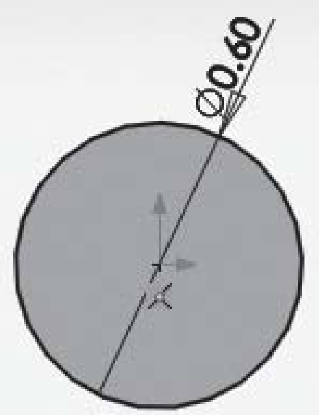
We use a screw mate to relate a rotating shaft (ball screw) to the transverse travel of a bearing. The screw mate can be used to relate the linear travel of a bearing or a nut to the rotation of a ball screw. This tutorial shows how to use the screw mate. First, we create the parts. The wheel has been created in Tutorial 3-3. Second, we add the necessary assembly mates to create a functional relationship between the ball screw and the nut. The bearing must travel away from the handle as it is rotated counterclockwise. Figure 6.9 shows the assembly.



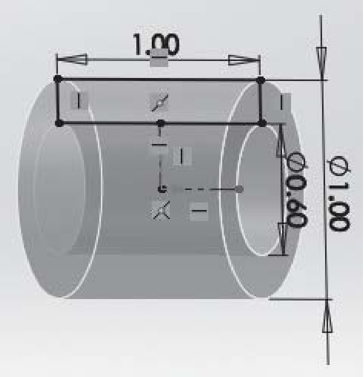
(A) Assembly (B) Assembly tree

FIGURE 6.9
Ball screw assembly

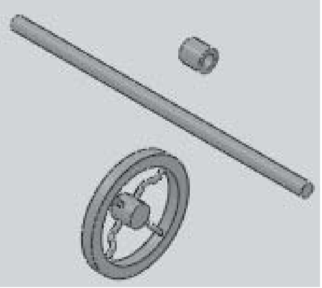
Step 1: Create the ball screw sketch and feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > enter 1.5 for thickness **D1** > ✔ > **File** > **Save As** > *screw* > **Save**.



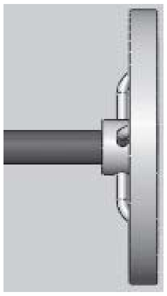
Step 2: Create the nut sketch and feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Revolved Boss/Base** on **Features** tab > **Line** on **Sketch** tab > sketch and dimension cross section as shown > **Centerline** on **Sketch** tab > sketch as shown > exit sketch > centerline as Axis of Revolution > ✔ > **File** > **Save As** > *nut* > **Save**.



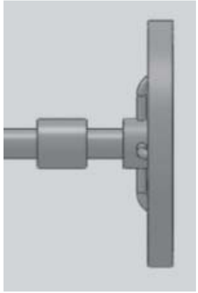
Step 3: Insert the three parts: **File** > **New** > **Assembly** > **OK** > **Browse** (button on left pane of screen) > locate the three parts and insert them (fix the screw in space) > **File** > **Save As** > *tutorial6.5*.



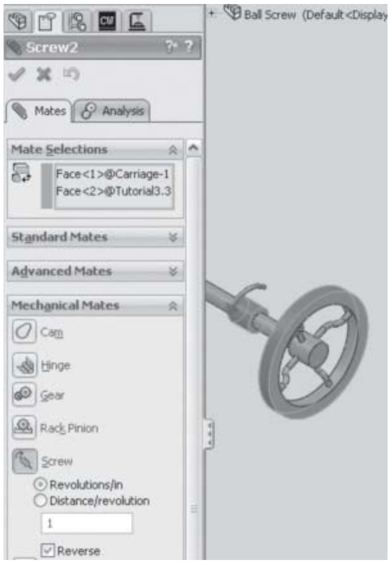
Step 4: Assemble the hand wheel: Align the axis of the hand wheel to the axis of the lead screw. Mate one end of the lead screw to the bottom face of the hand wheel as shown > ✔.



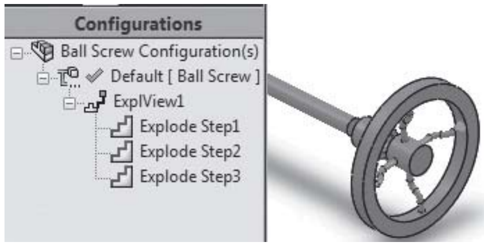
Step 5: Align the bearing: Slide the bearing toward the end near the hand wheel as shown > ✔.



Step 6: Create the screw mate: **Mate** on the **Assembly** tab > **Mechanical Mates > Screw >** select the faces and settings as shown > ✔.



Step 7: Test the ball screw: Drag with mouse and rotate the hand wheel counterclockwise. The nut will travel away from the wheel end.



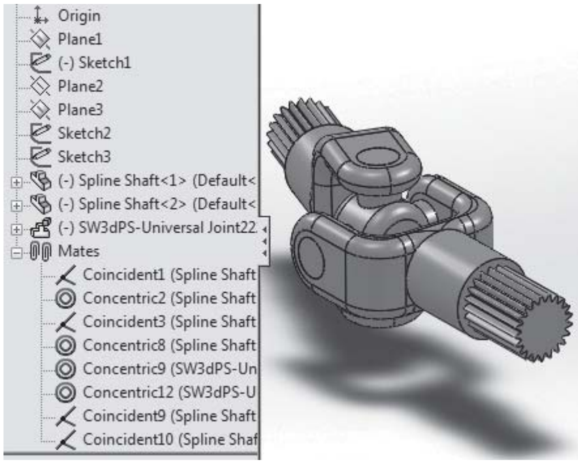
Step 8: Create exploded view: **Insert > Exploded View** > move components > ✔.

HANDS-ON FOR TUTORIAL 6–5. Re-create the assembly such that the bearing is fixed and the ball screw and hand wheel will travel when rotated.

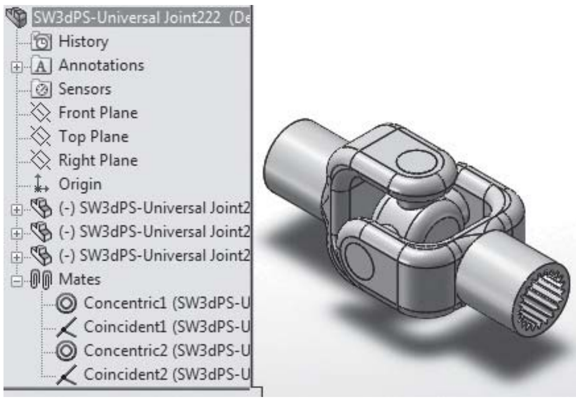
Hint: Use the right-click menu to set the bearing to **Fix** and the ball screw to **Float**. Test the motion by rotating the hand wheel.

Tutorial 6–6: Study Universal Joint Motion

We create an assembly that will simulate a real-life interaction of a universal joint and spline assembly. A universal joint is used to transmit torque between two shafts that are not aligned. Here the universal joint is downloaded along with the spline from 3D ContentCentral. The universal joint and spline are inserted into an assembly and, using the correct mates and reference planes, we demonstrate a working universal joint. Using a system of reference planes and mates, the universal joint will rotate along with the splines as in real life. Figure 6.10 shows the universal joint assembly.



(A) Assembly

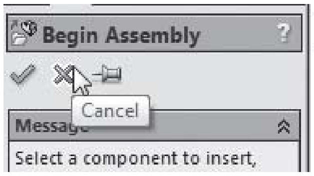


(B) Universal joint subassembly

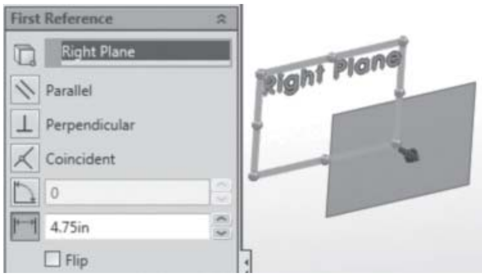
FIGURE 6.10
Universal-joint spline assembly

Step 1: Download universal joint and spline: Visit www3dcontentcentral.com. In the search bar, type "universal joint222" to download and save the universal joint subassembly. Then type "spline shaft" to download and save the first item in the search results.

Step 2: Create an assembly file: **File** > **New** > **Assembly** > **OK** > cancel as shown > **File** > **Save As** > *tutorial6.6* > **Save**.

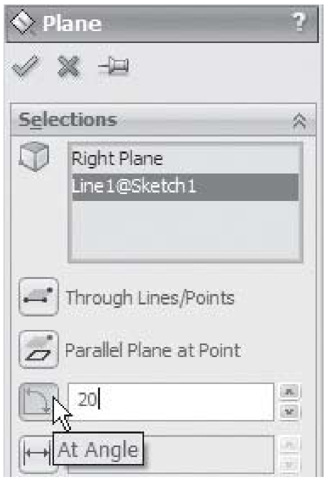


Step 3: Create reference plane *Plane1* to locate the spline: **Reference Geometry** on **Assembly** tab > **Plane** > **Right Plane** > enter 4.75 as the offset distance as shown > ✔.

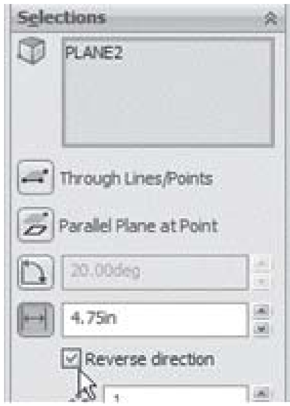


Step 4: Create *Sketch1* to use to define *Plane2* in Step 5: **Right Plane** as a sketch plane > sketch a vertical line as shown. The length of the line is not important > exit sketch.

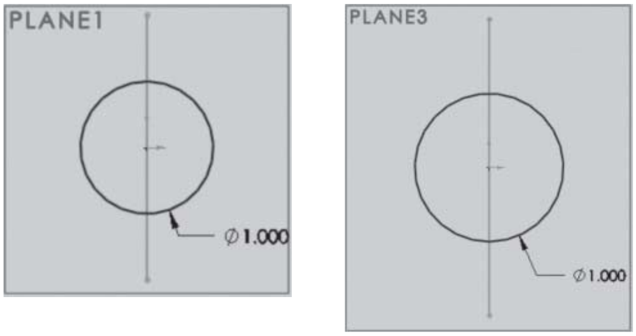
Step 5: Create reference plane *Plane2*: **Reference Geometry** on **Assembly** tab > **Plane** > **Right Plane** > select line created in Step 4 > enter 20° as shown > ✔.



Step 6: Create reference plane *Plane3*: **Reference Geometry** on **Assembly** tab > **Plane** > select *Plane2* created in Step 5 > create a plane that is offset by 4.75 in. > check off **Reverse direction** checkbox as shown > ✔.

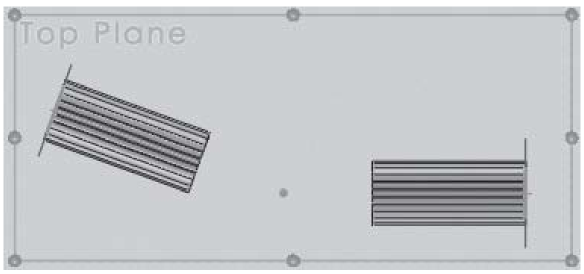


Step 7: Create *Sketch2* and *Sketch3*: *Plane1* as sketch plane > **Circle** on **Sketch** tab > click origin and sketch and dimension circle as shown > exit sketch > *Plane3* as sketch plane > **Circle** on **Sketch** tab > click origin and sketch and dimension circle as shown > exit sketch.

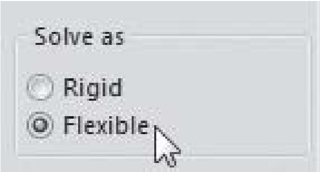


Step 8: Insert U-joint and spline shaft: **Insert Components** on **Assembly** tab > insert two instances of spline shaft and one instance of U-joint > ✔.

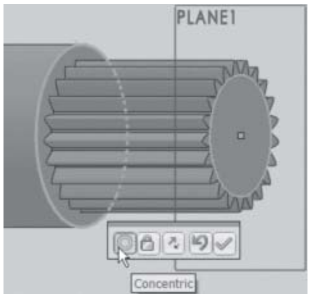
Step 9: Position two spline shafts: Mate one end of the first spline **Coincident** to *Plane1* and **Concentric** to the circle on *Plane1* > mate one end of the second spline **Coincident** to *Plane3* and **Concentric** to circle on *Plane3*. The result should look as shown looking down on the top plane.

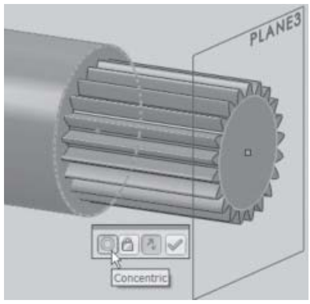


Step 10: Break up U-joint subassembly into components: Click U-joint node in assembly tree > **Component Properties** from context toolbar that pops up **>** **Flexible** as shown.



Step 11: Position U-joint: Use **Concentric** mate between one claw's edges and the sketch as shown here > repeat for the other claw and the other sketch > use **Coincident** mate between one spline's right plane and the corresponding claw's front plane > repeat for the other side.





Step 12: Test assembly: Drag either of the splines with mouse to make them rotate, and watch the behavior of the U-joint as it rotates.

HANDS-ON FOR TUTORIAL 6–6. Create a motion study for the U-joint.

Hint: Follow the steps from Tutorial 6–7.

Tutorial 6–7: Create Motion Study

We use the Cam assembly created in Tutorial 6–1 to create an animation motion study. An animation motion study is a visual demonstration of how an assembly moves. We use the mates of the assembly to restrict movements of parts in the assembly. The **Motion Manager** is a timeline interface from which we can use one of two types of studies: animation and basic motion. Figure 6.11 shows the motion timeline.

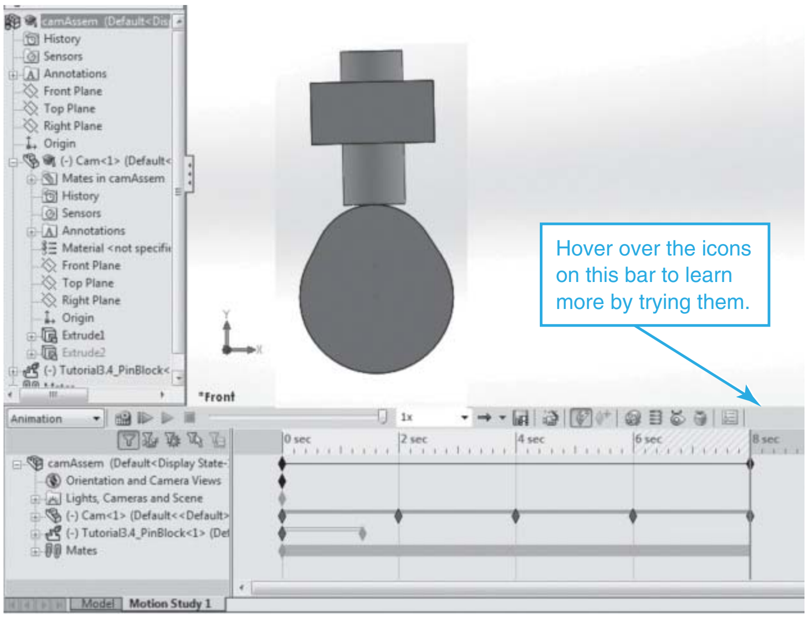
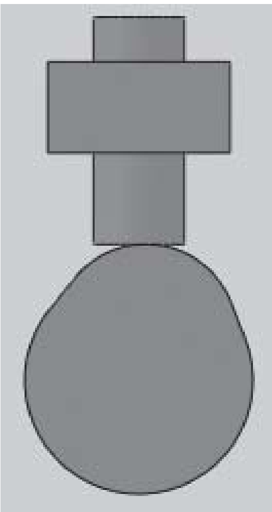
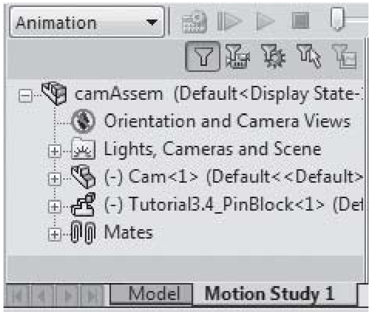


FIGURE 6.11
Motion timeline

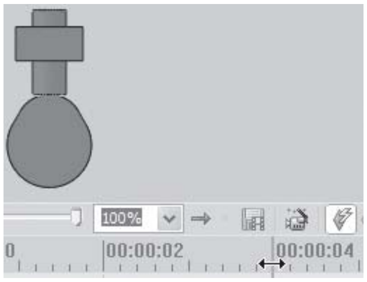
Step 1: Open Tutorial 6–1 cam assembly and prepare it for animation: **File > Open** > locate the assembly > **Open** > drag the cam until it is orientated vertically as shown.



Step 2: Open the **Motion Study** tab: Open the **Motion Manager** by clicking on the **Motion Study 1** tab at lower left of screen as shown.

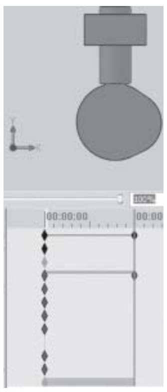


Step 3: Set the timeline at the first point: With the cam vertical, drag the timeline to 2 seconds as shown.



Note: You can create multiple motion studies. Right click the **Motion Study 1** tab > **Create New Motion Study** from the menu that pops up. You can delete any tabs you create but you cannot delete the **Motion Study 1** tab.

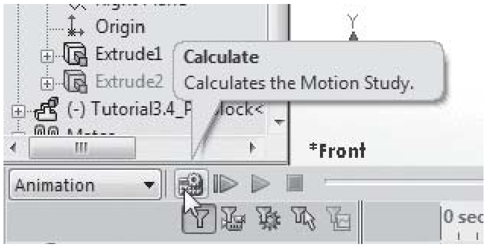
Step 4: Set cam for 2-second animation key: Drag the cam in a clockwise direction until it is horizontal (90° degree position) as shown.



Note: The timeline records the motion of the cam and follower. SolidWorks will insert what is called a *key* (diamond shape) after each motion drag is recorded. Right click any key to delete it if needed.

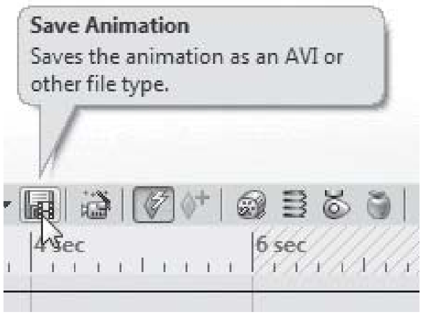
Step 5: Set cam for 4-, 6-, and 8-second animation keys: Repeat Steps 3 and 4 for the 4-, 6-, and 8-second keys, each time moving the key line by 2 seconds and turning the cam 90° from its current position.

Step 6: Calculate cam motion: Click the **Calculate** icon shown here to play the animation just recorded.



Step 7: Watch the playback: Click the **Play** icon to review the animation. If you need to make corrections or delete the animation, right click any key and delete as discussed in Step 4.

Step 8: Save the motion study (animation): Click the **Save** icon > accept default name (camAssem.avi) > **Save > OK.**



Note: The animation file is an external file that can be played back anytime independent of the assembly file being open or not. You also need an *avi* player (**Windows Media Player**), which the Windows OS provides. To play the file, simply double click it.

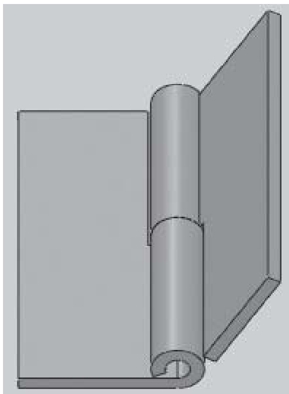
HANDS-ON FOR TUTORIAL 6–7. Create a second motion study that will rotate the cam in the opposite direction twice as fast.

Hint: Use a time interval of 1 second instead of 2 to span the 90° interval.

Tutorial 6–8: Detect Collision and Interference

The interference and collision detection tools can be very useful in analyzing an assembly. This tutorial makes use of the hinge created in Tutorial 6–2 to demonstrate these tools. Figure 6.12 shows the hinge assembly to test.

FIGURE 6.12
Hinge assembly

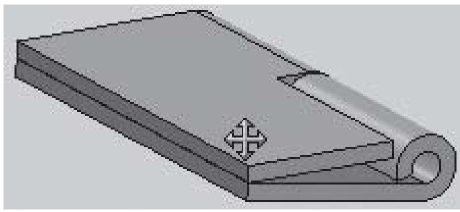


Step 1: Open Tutorial 6–1 hinge assembly and suppress hinge mate: **File > Open** > locate the assembly > **Open** > expand **Mates** node on assembly > click hinge mate > **Suppress** from the context toolbar that shows up (hover over icons until you read it as shown).

Note: The hinge mate has built-in limits of travel. For this tutorial the limits of travel will be a collision between the parts.

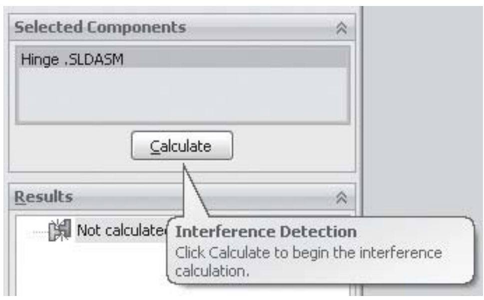
Step 2: Remate the parts: **Mates** on the **Assembly** tab > **Concentric** > click cylindrical faces of both hinge instances > ✔ > **Coincident** > click the flat faces that butt each other > ✔.

Step 3: Detect collision: **Move Component** on **Assembly** tab > select **Collision Detection, All components, Stop at collision** from the **Options** section of the menu that opens up > move one hinge plate until it cannot move as shown.

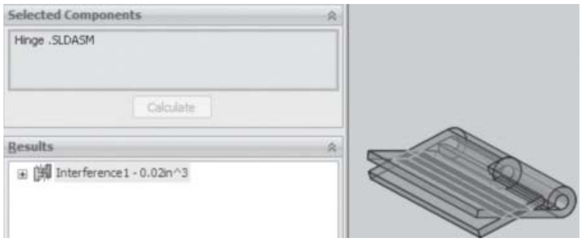


Note: SolidWorks will detect the collision of the two hinge plates and prevent the parts from moving any farther as well as sound a tone and highlight the faces in collision.

Step 4: Check interference: **Interference Detection** on **Evaluate** tab **> Calculate** button shown > review results (no interference is detected).



Step 5: Rotate hinge parts and recheck for interferences: Rotate the parts such that they are meshed together as shown > repeat Step 4 to recalculate > review results (interference is detected) as shown.



> ***HANDS-ON FOR TUTORIAL 6–8.*** Create a pin that connects the two hinge plates. Assemble the pin into the hinge assembly. Increase the diameter of the pin by 10% purposely above the hinge hole diameter, and check the collision and the interference of the assembly.

Tutorial 6–9: Create Design Table

A design table is an Excel sheet. A design table in an assembly can be used to control the suppression or configuration of parts, mates, and assembly features and also the numeric distance or angle relationships between parts. We create and set up a design table such that the configurations control the location of the pin in the Pin Block assembly of Tutorial 3–4 in Chapter 3. We assign a variable and value to each configuration in the design table. Figure 6.13 shows the three resulting configurations.

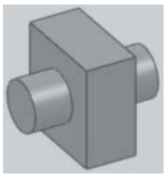
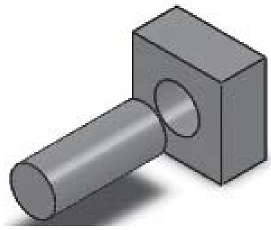
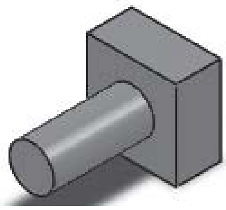
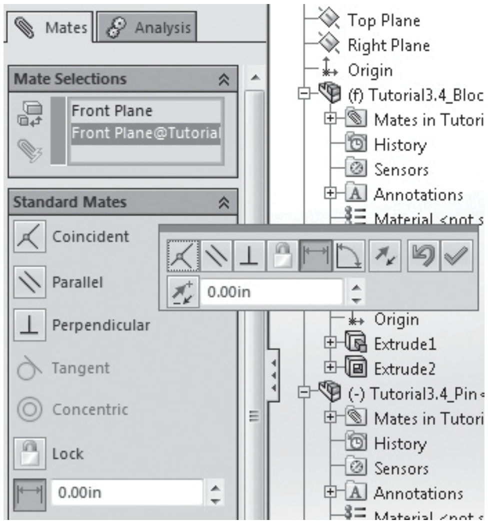
Configuration	Pin location
Default	
2	
3	

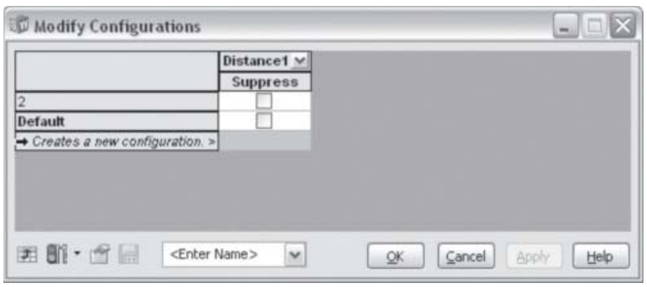
FIGURE 6.13
Design table

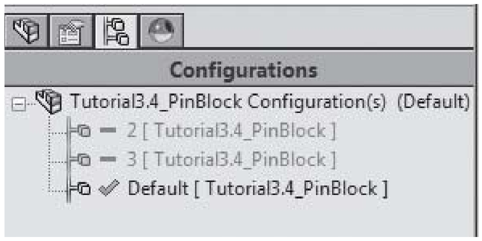
Step 1: Open Tutorial 3–4 Pin Block assembly and convert the coincident mate to a distance mate: **File > Open** > locate the assembly > **Open** > convert the **Coincident** mate between the **Front Plane** of the pin and the **Front Plane** of the assembly to a **Distance** mate as follows: click the mate in the assembly tree > **Edit Feature** from context toolbar that pops up > select the **Distance** mate as shown > right click each top plane shown under **Mate Selections**, delete, and replace them by the assembly **Front Plane** and the pin **Front Plane** (expand the trees to access them) > leave the distance value at 0.0 > ✔.



Step 2: Configure the distance mate: Right click the **Distance** mate in the assembly tree > **Configure feature**. This opens up a design table (Excel sheet) shown in Step 3.

Step 3: Add configurations: Add *Configuration* 2 by clicking in the cell that reads **Creates a new configuration >** enter 2 for **Configuration Name** > enter 3.00 for **D1** > > repeat to create *Configuration 3* with **D1** of 5 > **OK.**

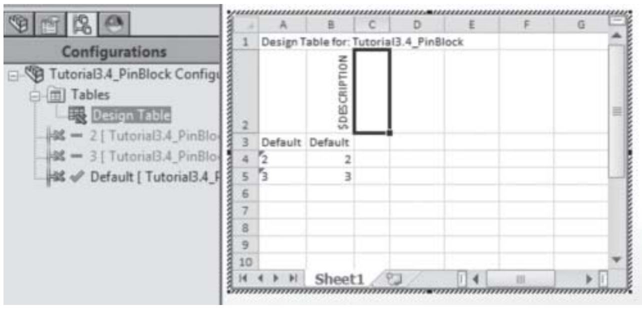




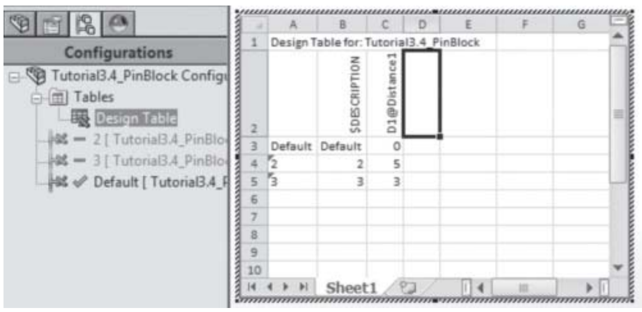
Note: The **ConfigurationManager** has been populated by the configurations named “2” and “3” as shown. The default configuration has 0 value for **D1.**

Step 4: Create the design table: **Insert > Tables > Design Table** (use the default **Auto-create**) > ✔.

Note: This automatically populates the Excel worksheet with the three configurations. The first column is the name of the configurations. The second column is the description of each configuration as shown.

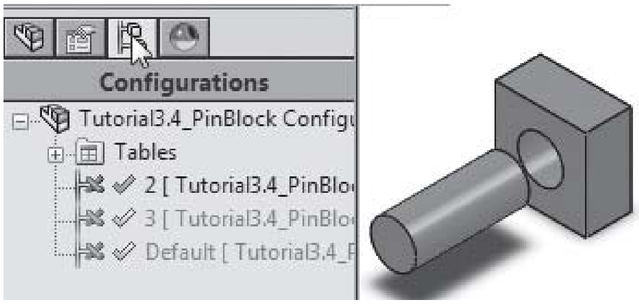


Step 5: Edit the design table: While the design table is open from Step 4, click assembly tree tab > expand **Mates** node > click **Distance1** mate to show dimension value in graphics pane > double click the value (should be 0.0) to add **D1** and its current value to design table as shown > click **D1** cells for other configurations and enter 3 and 5 for **D1** as shown > click anywhere on graphics pane away from table to close and save it.



Step 6: Test the design table: **ConfigurationManager** tab > double click any configuration (Default, 2, or 3) shown > observe the position of the pin as shown to the left.

Note: The active configuration is shown in **ConfigurationManager** pane with ✔.



> ***HANDS-ON FOR TUTORIAL 6–9.*** Add a "4" configuration such that the distance mate is suppressed.
>
> **Hint:** Right click **Distance1** mate in **Mates** node in assembly tree > **Configure Feature** from the menu that pops up > add configuration "4" with distance of 7 in. > check off the **Suppress** checkbox > **OK**. Double click configuration "4", and drag the pin with the mouse to move it. It will move, therefore not obeying the distance of the 7 inch requirement because the mate is suppressed. Try moving the pin in the other configurations. Suppressing a mate may be useful in answering "what if" design questions.

Tutorial 6–10: Create Part in Context of Assembly

We create a part in the context of an assembly such that the part's dimensions reference other parts and assembly features. Figure 6.14 shows the assembly with the part in context (cross bar). Creating a part in the context of an assembly can be useful when the size of the part is completely dependent on other parts or assembly features. The part (cross bar) created in context must always contact the two blocks to act as a bracket. First, three parts are created and then inserted into an assembly. Second, a part is created in the context of the assembly that references the sizes and locations of the original three parts.

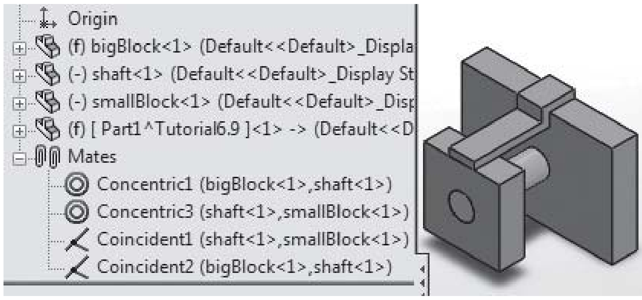
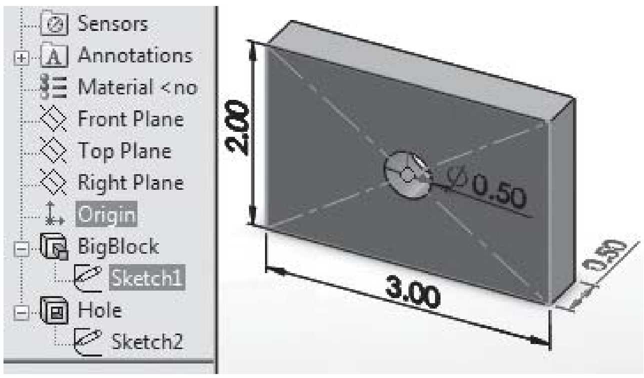
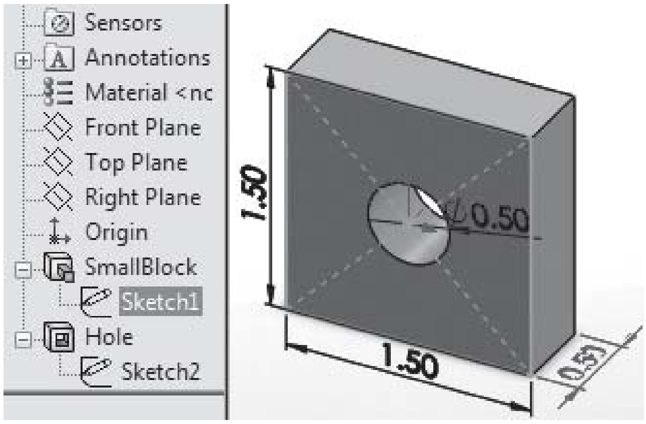


FIGURE 6.14
Assembly with cross bar as part in context

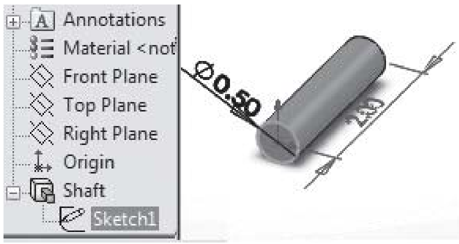
Step 1: Create *BigBlock* feature: **File** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and sketch and dimension as shown above > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > front face of extrusion just created > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch and dimension as shown above > exit sketch > **Through All** > ✔ **File** > **Save As** > *bigBlock* > **Save.**



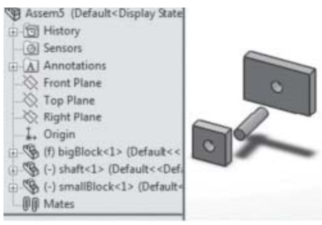
Step 2: Create *SmallBlock* feature: **File > New > Part > OK > Front Plane > Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > front face of extrusion just created > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > **Through All** > ✔ **File > Save As** > *smallBlock* > **Save.**



Step 3: Create *Shaft* feature: **File > New > Part > OK > Front Plane > Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > enter 2.0 for thickness **D1** > ✔ **File > Save As** > *shaft* > **Save.**

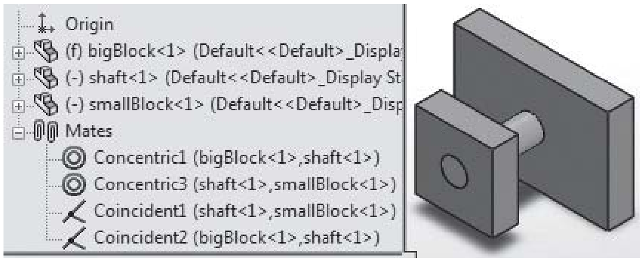


Step 4: Create assembly: **File > New > Assembly > OK** > **Browse** (button on left pane of screen) > locate the files of the three parts just created and insert them > **File > Save As** > *tutorial6.10* > **Save**.

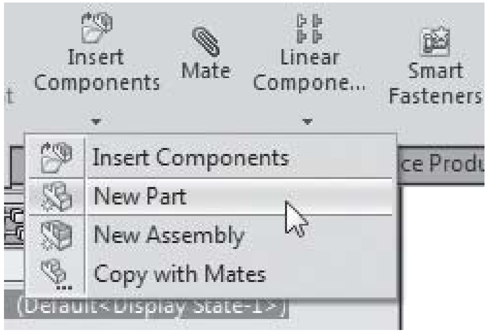


Note: Fix *BigBlock* as shown.

Step 5: Assemble parts: **Mate** on **Assembly** tab > **Concentric** mate between the hole of *BigBlock* and *Shaft* > **Concentric** mate between the hole of *SmallBlock* and *Shaft* > **Coincident** mate between the outside face of *BigBlock* and one end face of *Shaft* > **Coincident** mate between the outside face of *SmallBlock* and the other end face of *Shaft*.

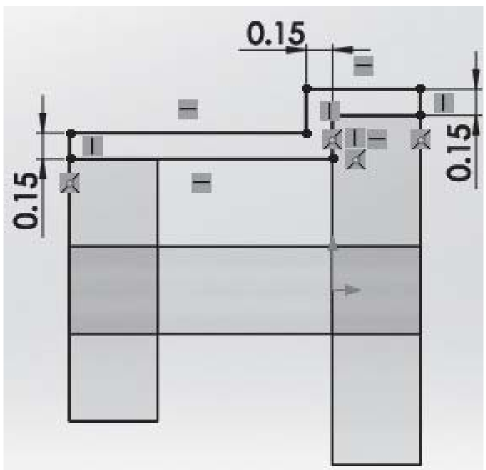


Step 6: Insert blank cross bar part (part in context): **Insert > Component > New Part** as shown > (click somewhere in graphics pane to exit).



Note: The new part exists as a node in the assembly tree (look for it). Expand it. The part has no geometry, but its sketch planes exist and are automatically aligned with the planes of the assembly.

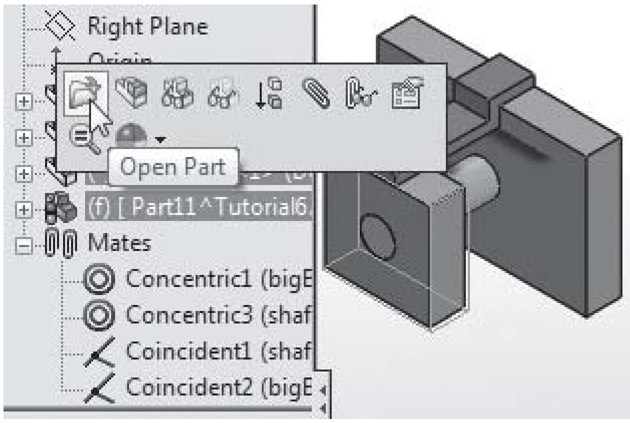
Step 7: Create cross bar sketch and feature: Click the part node > **Edit Part** from the context toolbar that shows (hover over icons until you read it) > expand



part node > **Right plane** > **Extruded Boss/Base** on the **Features** tab > **Line** from **Sketch** tab > sketch and dimension as shown > do NOT exit the sketch > **Features** tab > **Extruded Boss/Base** > enter 0.5 for thickness **D1** > **Mid Plane** > > ✔ .

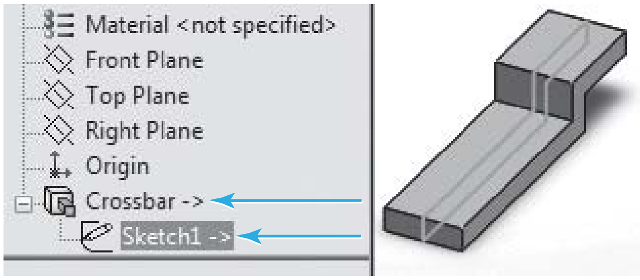
Note:
(1) The far left vertical line does not become vertical as you sketch it, so click it > **Vertical** from the **Add Relations** section.
(2) Second to last symbol shown in context is the **Edit Part** mode. Click it to exit the **Edit Part** mode and return to typical assembly operations.
(3) To edit a part or assembly in the context of another assembly, click it > **Edit Part.**

Step 8: Open context part: Click the part node > **Open Part** from the context toolbar that shows (hover over icons until you read it).



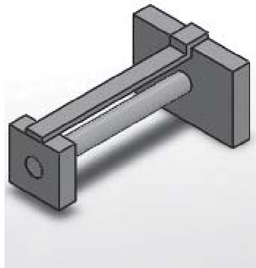
Step 9: Save part: **File** > **Save** > **File** > **Close** to return to assembly.

Note:
(1) Do not use **Save As**



(2) You cannot name the part yourself. Its name becomes Partxx^assemblyName (*Part1^Tutorial6.10* for this tutorial). But you can rename the features tree nodes.
(3) A context symbol (->) is attached to part tree nodes as shown.
(4) Assembly tree shows the cross bar part as a new node.
(5) Saving the cross bar part in context of assembly does not create a new file for the part. If you wish to do so, follow Step 7 > **File** > **Save As** > **Yes** to break the link between context part and assembly.

Step 10: Edit assembly: Click part node > **Open Part** from the context toolbar that shows (hover over icons until you read it) > click shaft node > **Edit Feature** from context toolbar > enter 5 for **D1**, replacing current value of 2 > ✔ > **File** > **Save** > **File** > **Close** > **Yes** to rebuild assembly > exit part by clicking .



Note: Observe that the cross bar length increases to maintain sketch relations as shown.

HANDS-ON FOR TUTORIAL 6–10. Edit the shaft diameter such that it references the diameter of the hole in the *BigBlock* part. Also edit the hole in *SmallBlock* such that it references the shaft diameter. Then change the diameter of the hole in the block to 1.0 in.

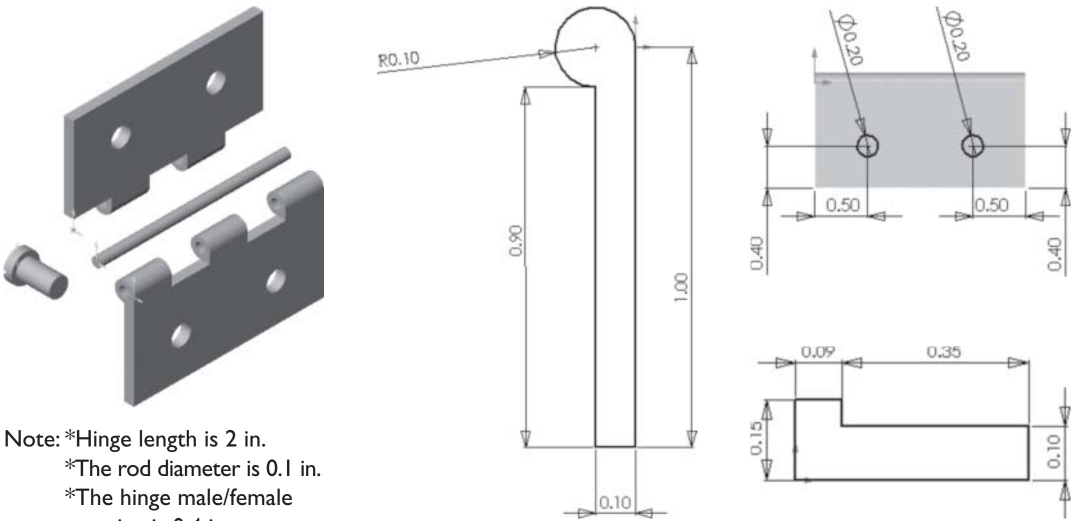
Note: You will have to delete the 0.25 in. dimension in the cross section sketch of the shaft.

problems

1. What is the difference between part modeling and assembly modeling?
2. What is an instance? Why does a CAD/CAM system use instances in assemblies?
3. What does an assembly mate do?
4. List the part entities that you can use for mating.
5. What are the differences between bottom-up and top-down assembly modeling?
6. What is the advantage of using the layout sketch in the top-down assembly approach?
7. What is the difference between interference detection and collision detection between assembly components? Give an example.
8. What does an assembly design table enable you to do for an assembly?

For the remaining problems, use the bottom-up approach to create the assembly models shown. Also, create an assembly drawing, exploded view, and animation (follow Tutorial 1–5 in Chapter 1). Check for interference and collision detections. We sometimes do not show all dimensions if they make the figure cluttered and hard to read. So, approximate them.

9. Door hinge.

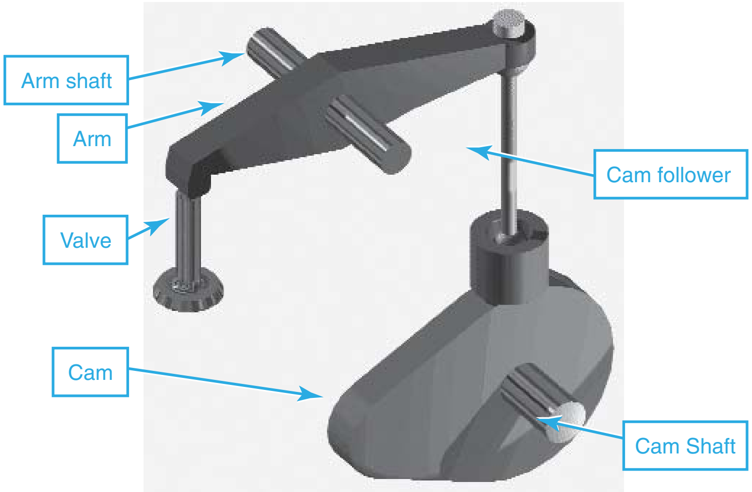


Note: *Hinge length is 2 in.
*The rod diameter is 0.1 in.
*The hinge male/female spacing is 0.4 in.
*Other dimensions are shown to the right.

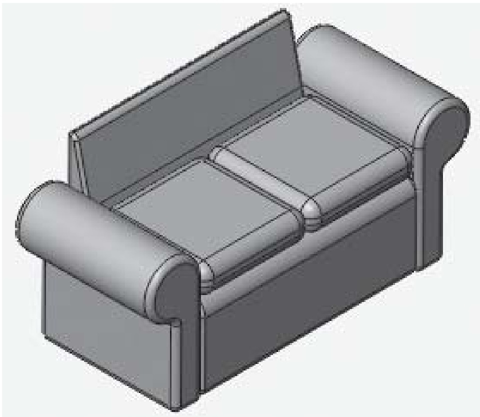
FIGURE 6.15
A door hinge (all dimensions are in inches)

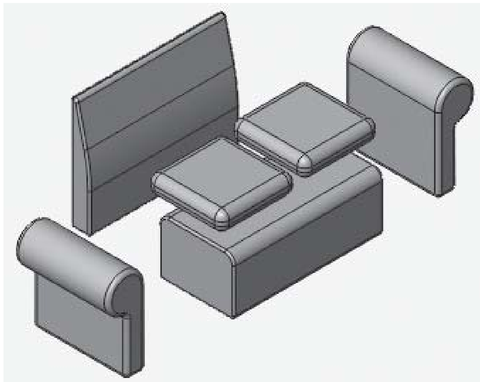
10. Rocker arm.

FIGURE 6.16
A rocker arm (make up your dimensions in inches or mm)

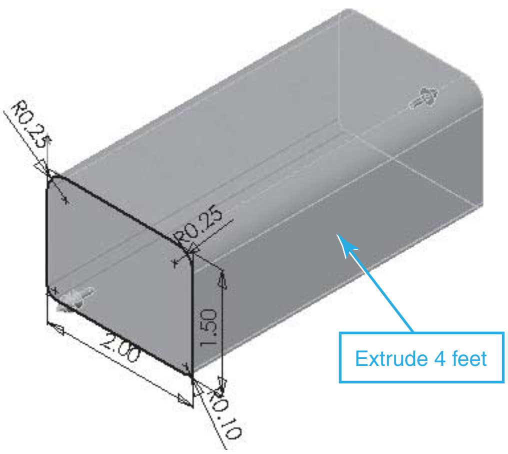


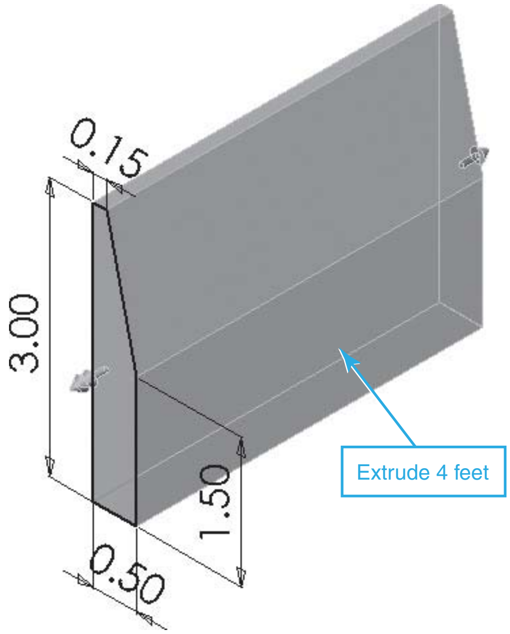
11. A couch.





Note: All fillets have a diameter of 0.1 in., except cushions (use 0.2 in. diameter fillets).

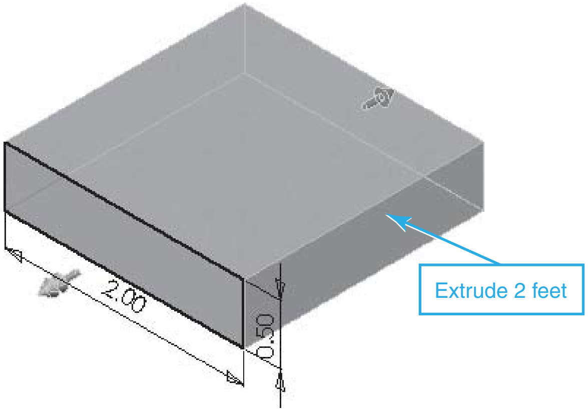




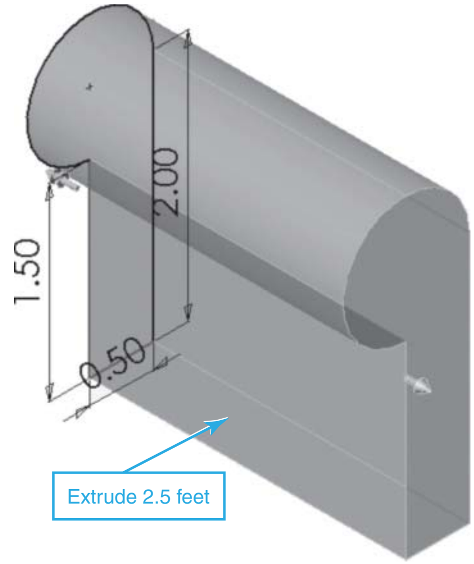
Couch base

Couch back-rest

FIGURE 6.17
A couch (all dimensions are in inches)



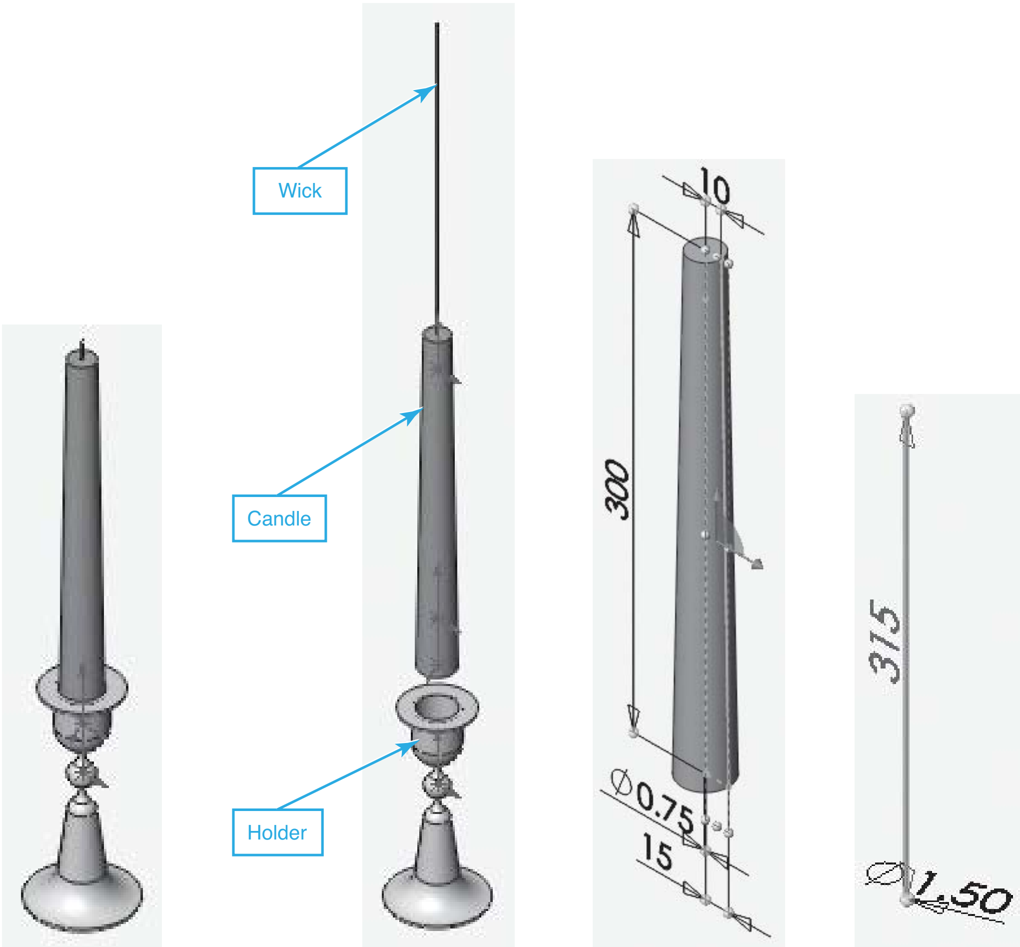
Couch cushion



Couch arm rest

FIGURE 6.17
(*continued*)

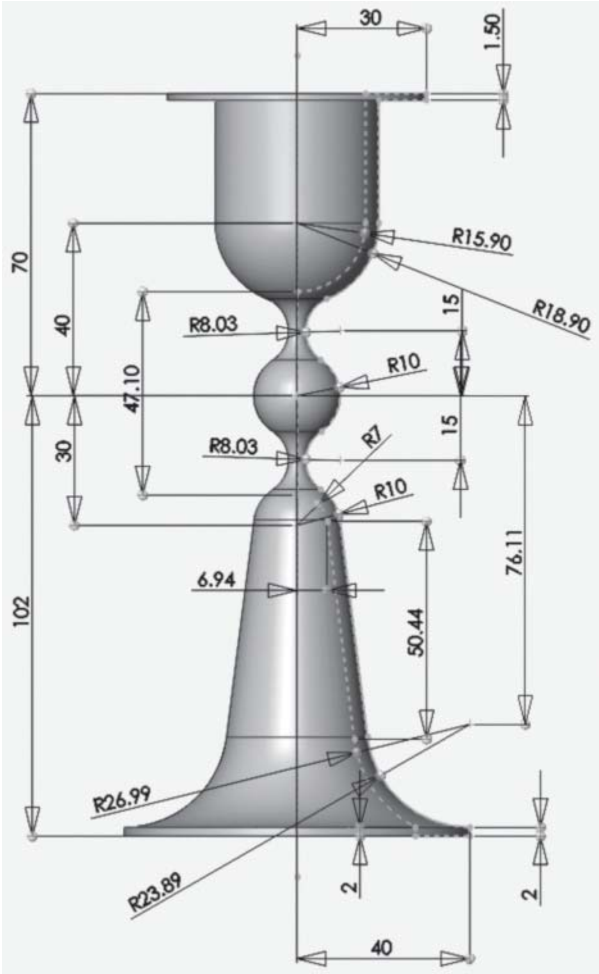
12. A candle holder.



Candle

Wick

FIGURE 6.18
A candle holder (all dimensions are in mm)



Holder

FIGURE 6.18
(*continued*)

13. A ballpoint pen.

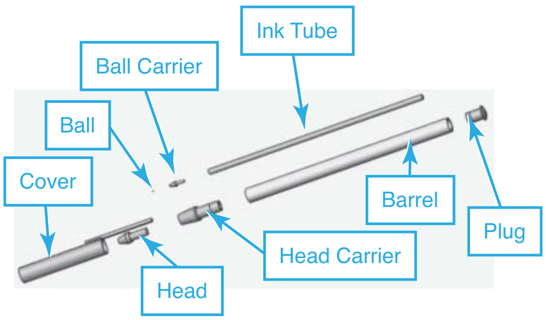
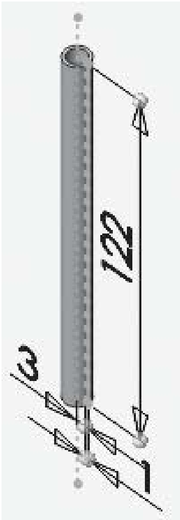
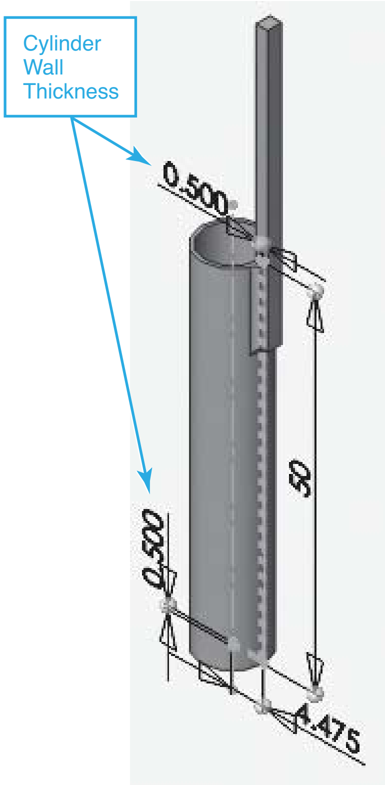


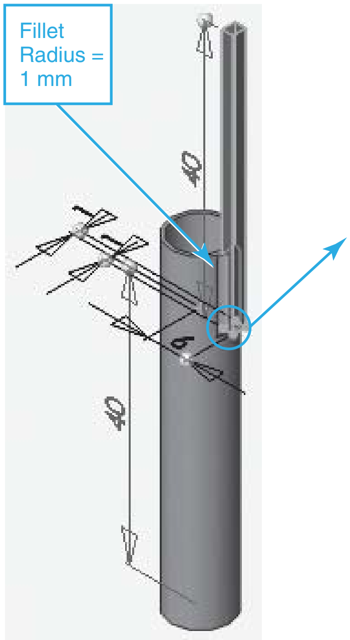
FIGURE 6.19
A ball point pen (all dimensions are in mm)



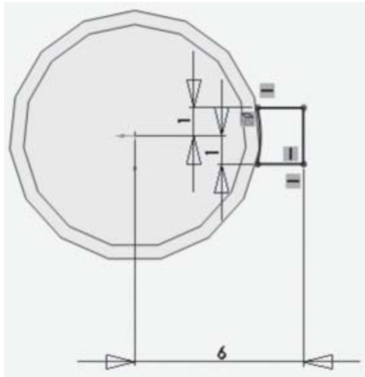
Barrel



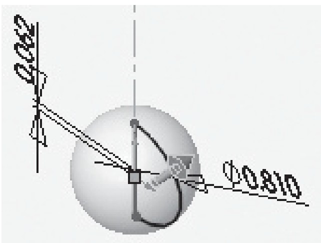
Cover cylinder



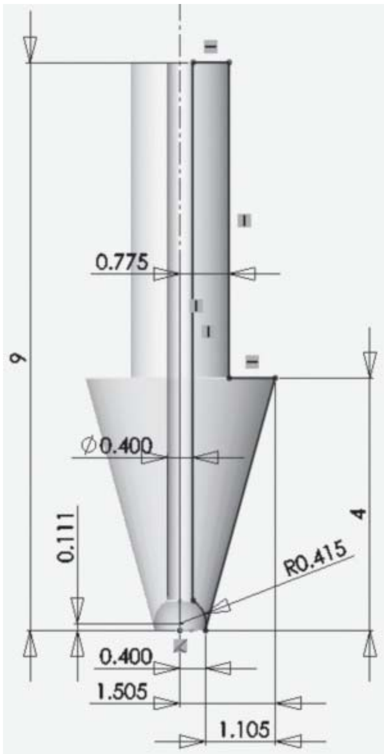
Cover clip



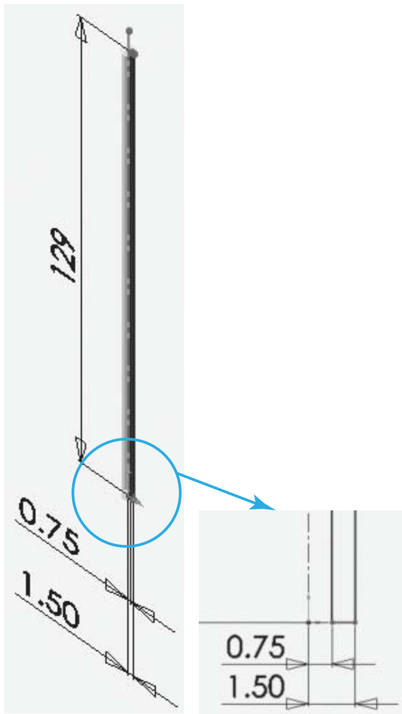
Top view of the clip sketch (cross section)
Note: The sketch consists of three lines and an arc whose radius is 4.475 mm. There is a fillet between the clip and the cylinder of radius = 1 mm as shown to the left.



Ball

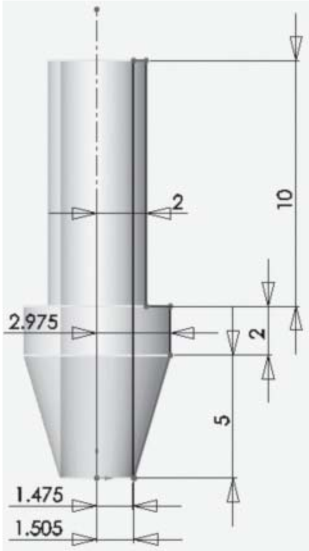


Ball carrier

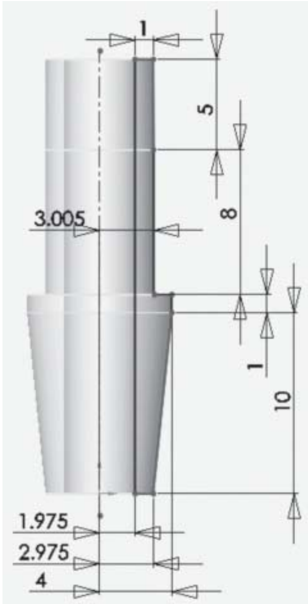


Ink tube

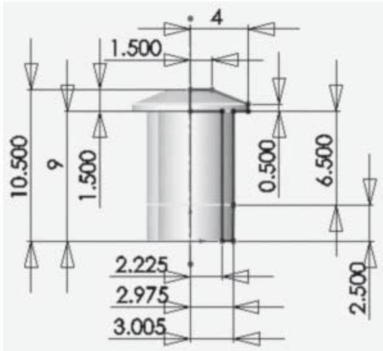
FIGURE 6.19
(continued)



Head



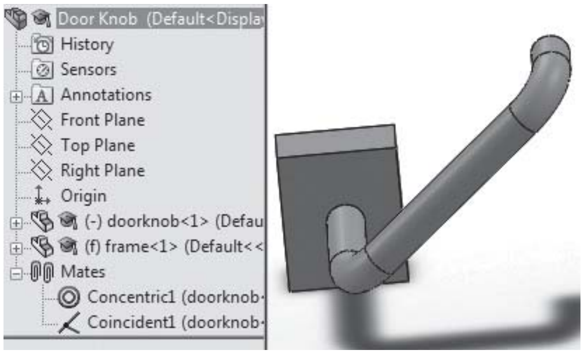
Head carrier



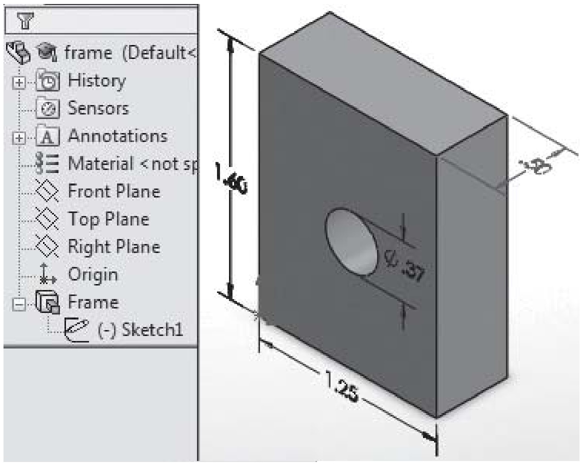
Plug

FIGURE 6.19
(*continued*)

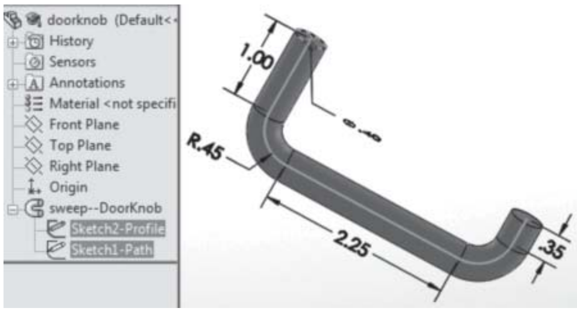
14. A door knob.



(A) Assembly



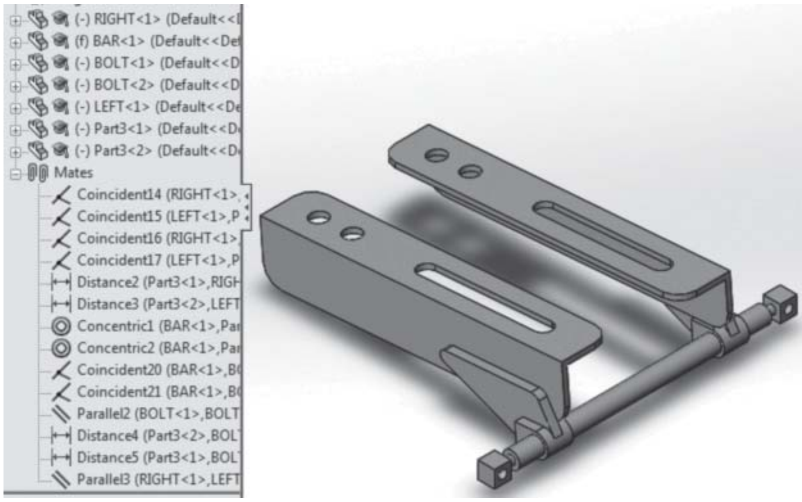
(B) Frame



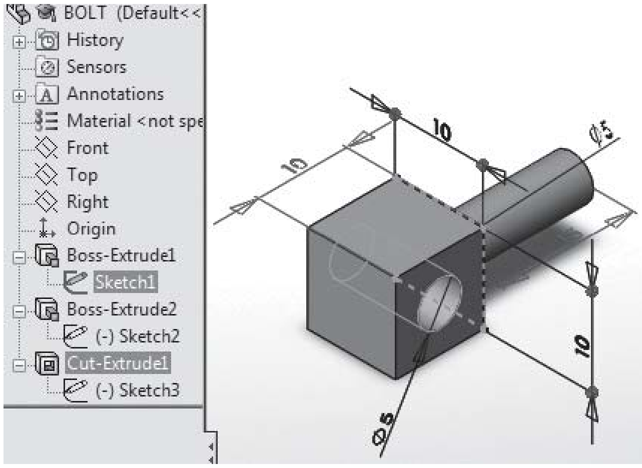
(C) Knob

FIGURE 6.20
A door knob (all dimensions are in inches)

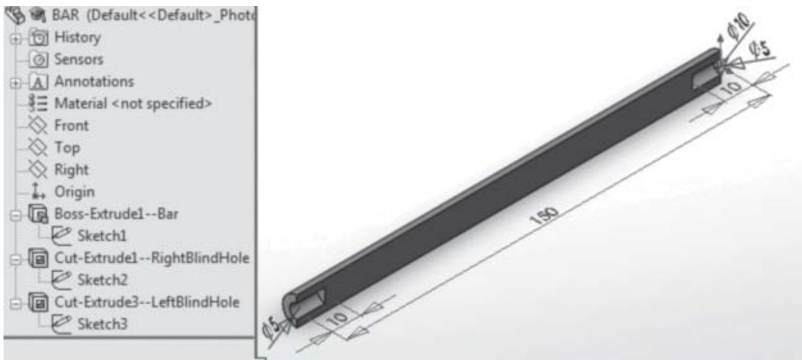
15. A bracket.



(A) Assembly



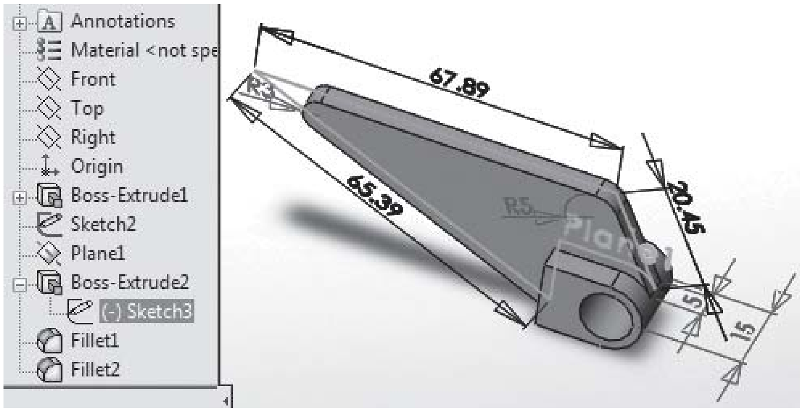
(B) Bolt



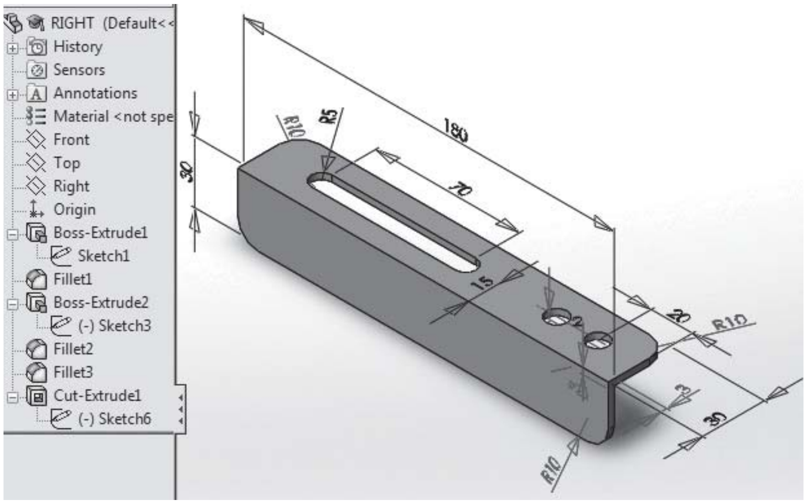
(C) Bar

FIGURE 6.21

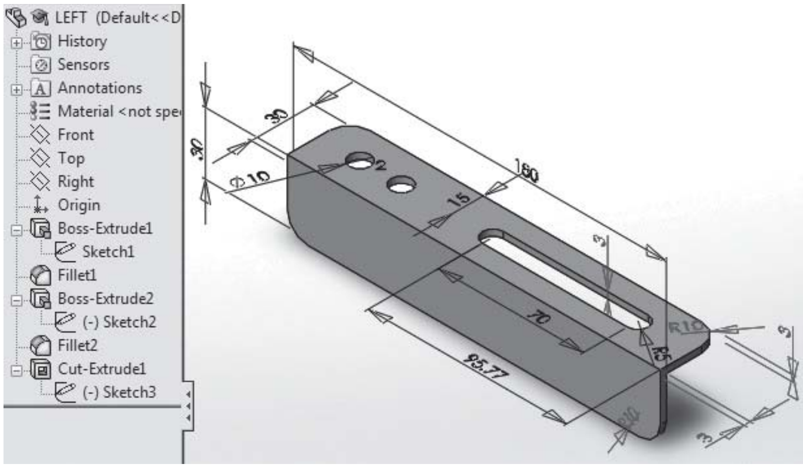
A bracket (all dimensions are in cm)



(D) Bracket



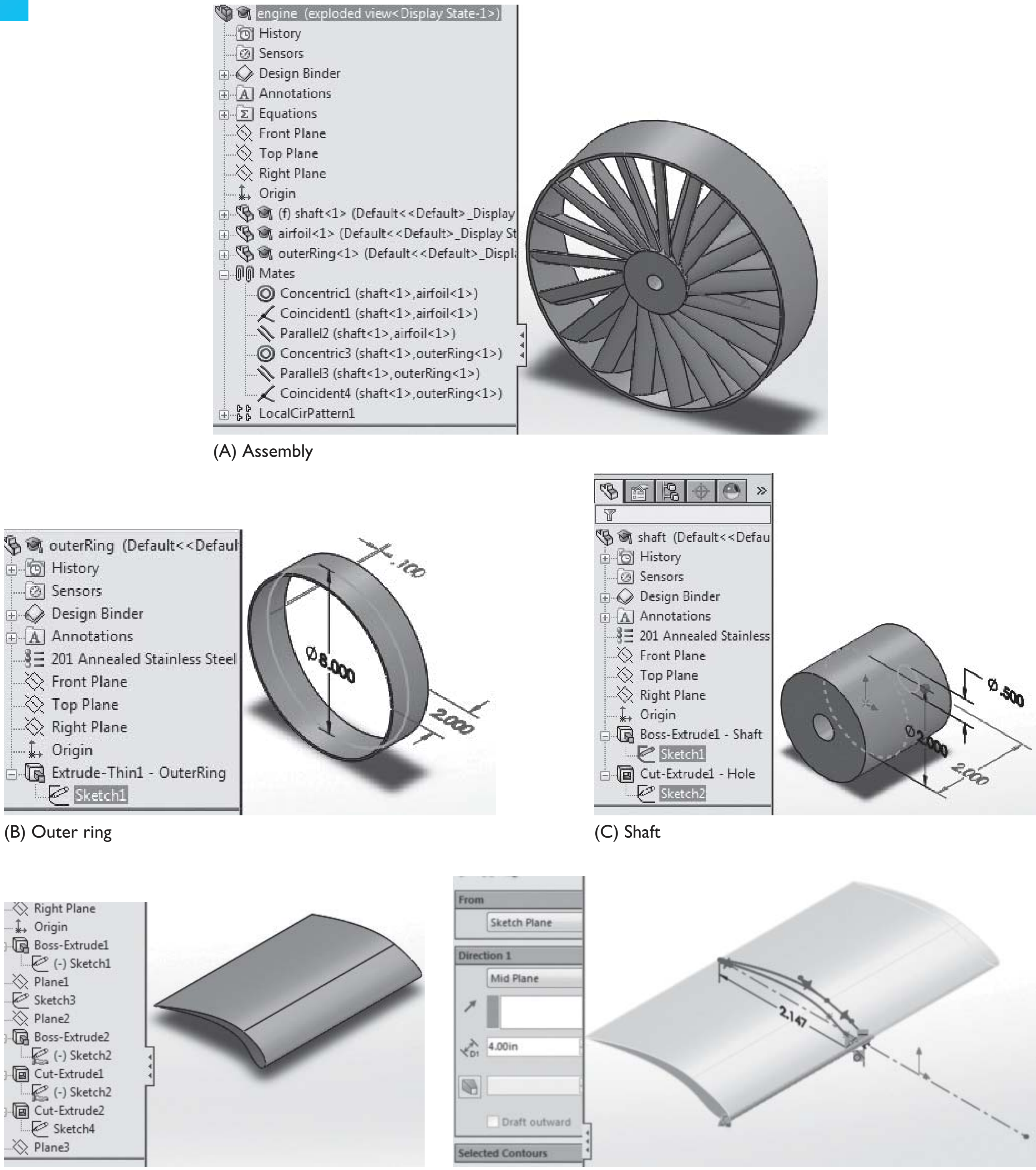
(E) Right plate



(F) Left plate

FIGURE 6.21
(*continued*)

16. A simplified jet engine blade assembly that requires creating an airfoil and a circular pattern of it. An interesting concept here is that you create the circular pattern in the assembly after you assemble the airfoil. Another interesting concept is the animation of the assembly. It requires you to move all the airfoil instances.



(A) Assembly

(B) Outer ring

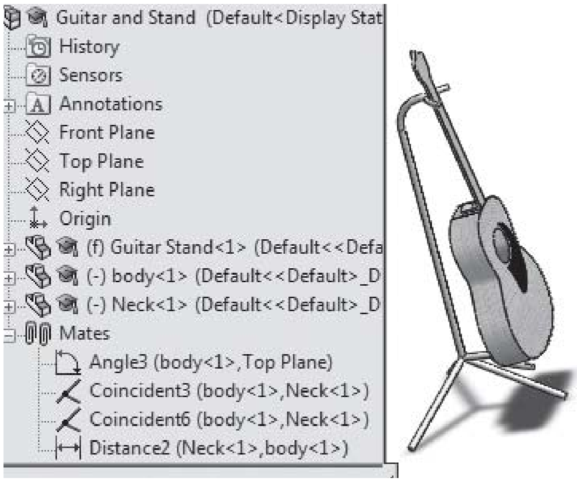
(C) Shaft

(D) Airfoil

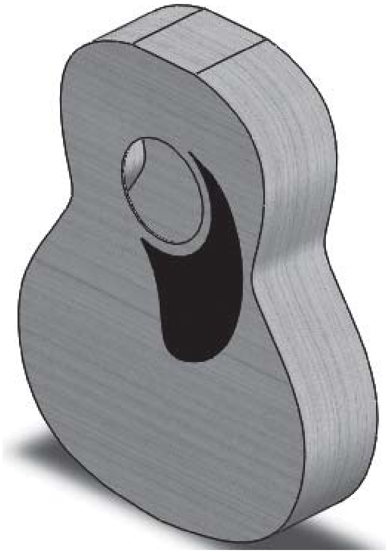
(E) Airfoil cross section

FIGURE 6.22
Simplified jet engine blade assembly (all dimensions are in inches)

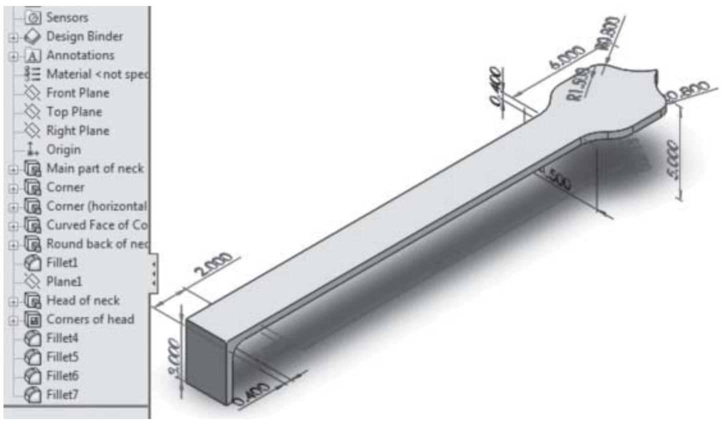
17. A guitar and stand assembly.



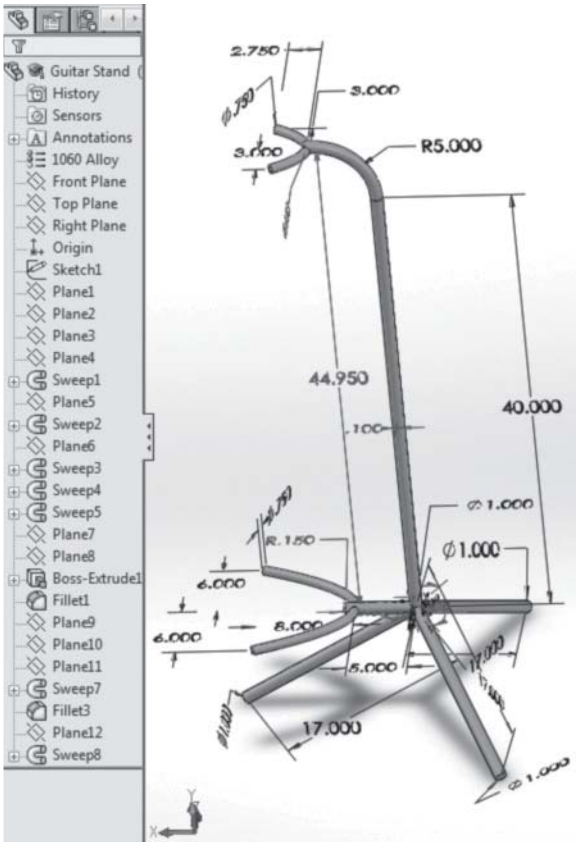
(A) Assembly



(B) Body



(C) Neck



(D) Stand

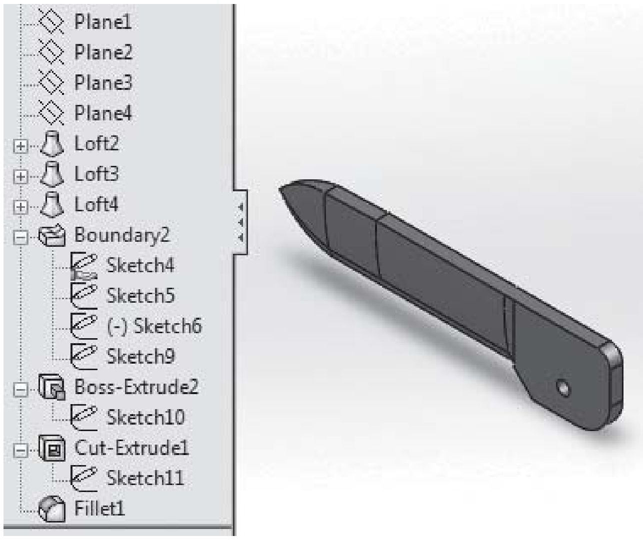
FIGURE 6.23

Guitar and stand assembly (all dimensions are in inches)

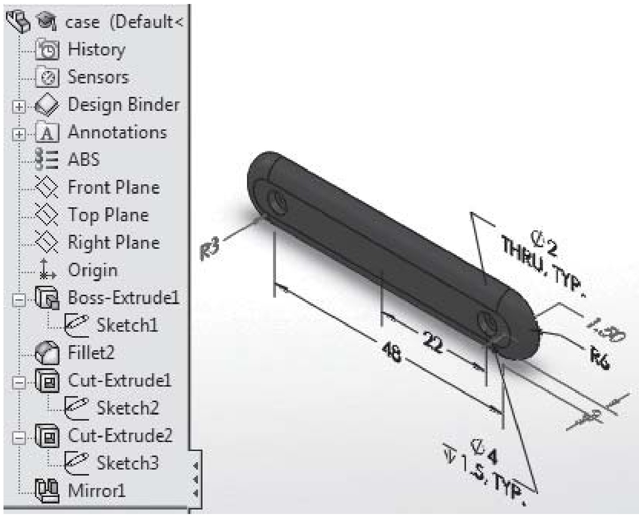
18. A Swiss Army knife.

(A) Assembly

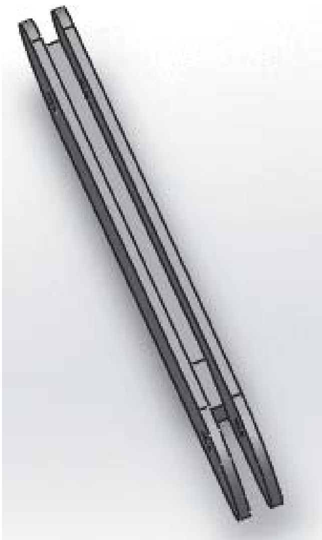
(B) Exploded view



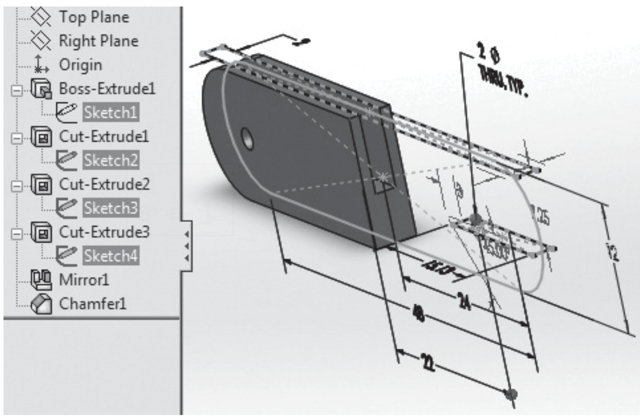
(C) Blade



(D) Case

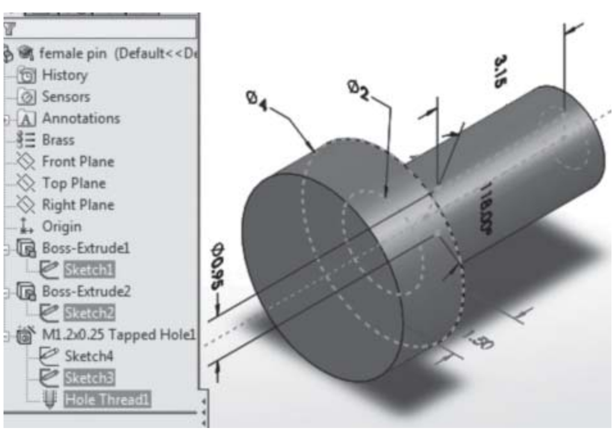


(E) Blade stopper

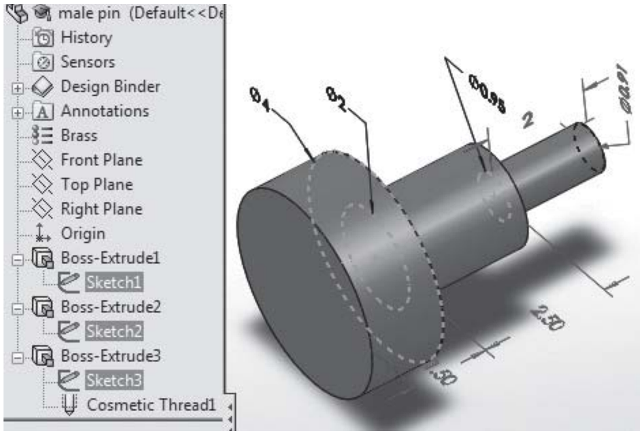


(F) Stopper section view

FIGURE 6.24
Swiss Army knife



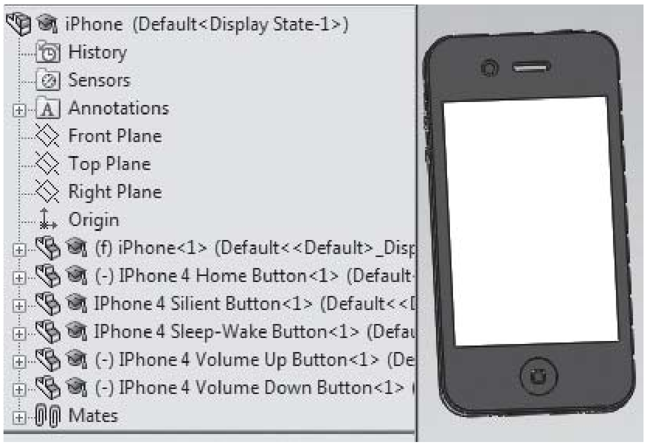
(G) Female pin



(H) Male pin

FIGURE 6.24
(*continued*)

19. iPhone. Use your own phone and measure dimensions.



(A) Assembly

(B) Main body

FIGURE 6.25
iPhone

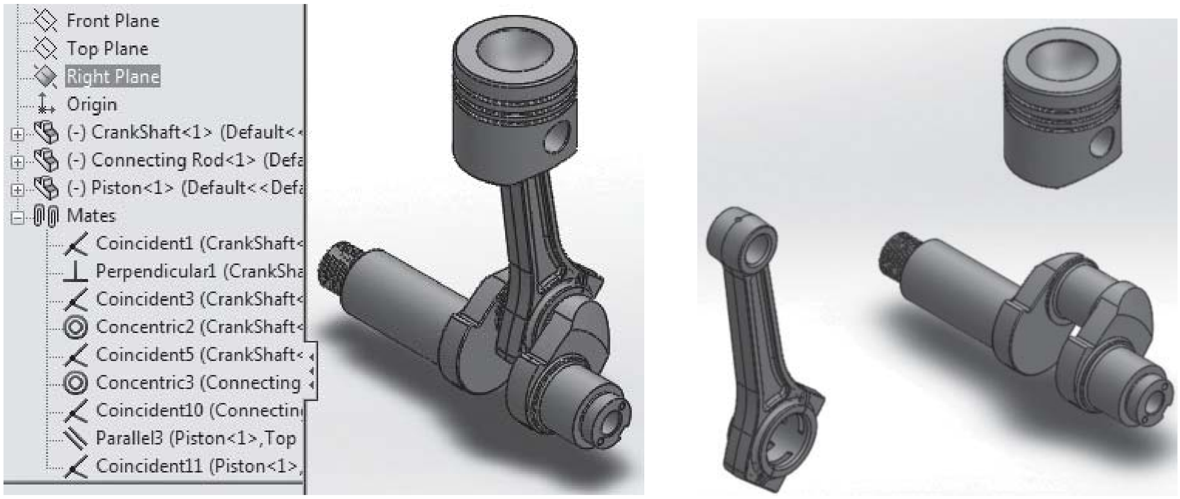
20. A laptop. Use your own laptop and measure dimensions.

(A) Assembly

(B) Exploded view
Note: The circle with letters inside is a decal reading “UL” that is pasted on the laptop screen.

FIGURE 6.26
A laptop

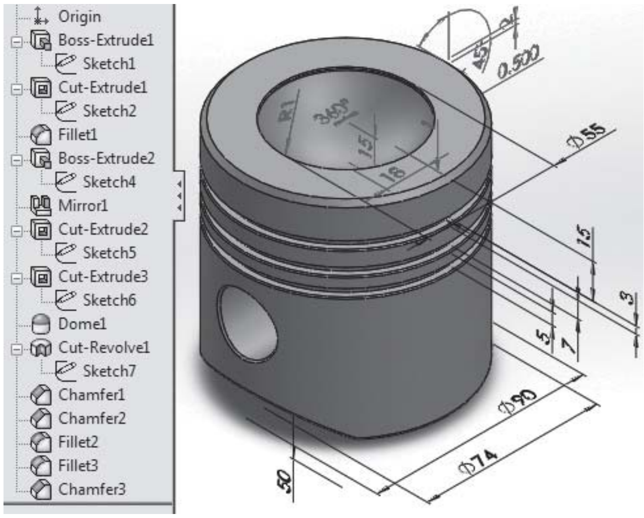
21. An engine piston assembly.



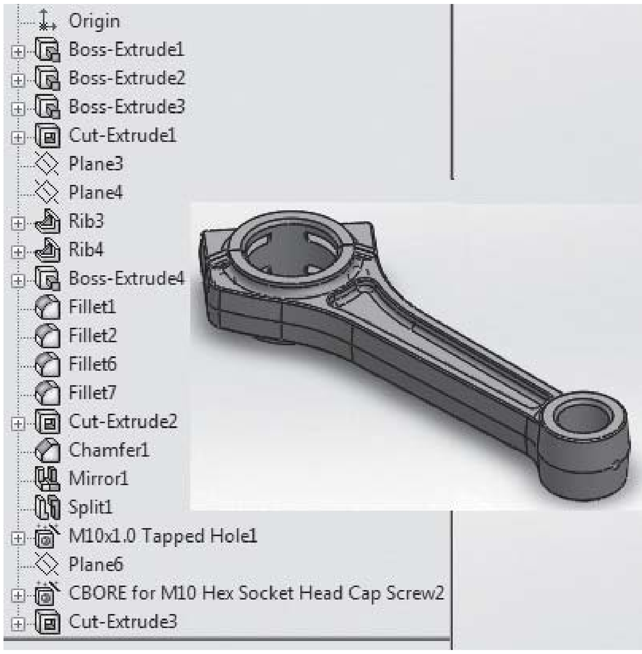
(A) Assembly

(B) Exploded view

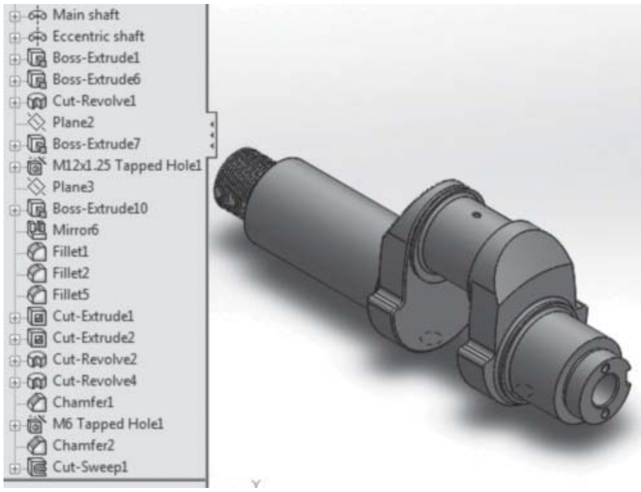
FIGURE 6.27
An engine piston assembly (all dimensions are in mm)



(C) Piston



(D) Connecting rod



(E) Crank shaft

FIGURE 6.27
(*continued*)

This page intentionally left blank

CHAPTER

7

Rendering and Animation

7.1 Introduction

Visualization has long been recognized as an effective tool of communication, and more so for engineering. CAD/CAM software offers very rich menus for visualization, including 3D views, hidden line removal, wireframe, and shaded views. Click the **Display Style** icon of the **Heads-up View** toolbar at the top of the graphics pane on the screen to experience these styles. Engineering visualization tools include rendering, photo realism, and prototyping. We cover rendering in this chapter and prototyping later in the book. Rendered photos are used both for sales and marketing (sales brochures) and for conveying design ideas to teams.

Rendering generates realistic images, almost like photographic scenes taken with a digital camera. We can render the screen graphics back onto the graphics pane of SolidWorks, to a printer, or to a file. To increase its rendering speed, CAD/CAM software uses the graphics card of the computer to support real-time rendering, so as we rotate or move a CAD model on the screen, it is rendered quickly.

Creating scenes is a trial-and-error process because you want to evaluate various effects on the scene until you find the best rendering. One major issue that is detrimental to rendering time and image size is the rendering resolution, which is much like graphics resolution. The higher the resolution, the more time the image takes to render and the larger its size is. The resolution is known as the *anti-aliasing effect*. Aliasing is the jaggedness (known as *staircase effect*) of the model edges due to the screen pixels. Higher anti-aliasing produces high-quality images, but it takes a long time to compute (render) the scene.

Rendering is a complex topic. We need to understand rendering models, how to create a scene, what texture to use, and what types of lights are available and their effect. Without such understanding, rendering remains a mystery and the use of rendering software becomes inefficient. SolidWorks has **PhotoView 360** as an Add-Ins. The tutorials cover how to use it effectively.

7.2 Scenes and Lighting

Digital rendering uses a rendering scene and a rendering model. The rendering scene describes the lighting environment including the types of lights in the scene. Figure 7.1 shows the elements of a scene, which includes the CAD model to render and the lights. SolidWorks uses a room with four walls to define the boundaries of its scene. These are the top (ceiling), the bottom (floor), the left wall, and the right wall. The CAD model includes its material and texture. We can assign a texture map to a model. The scene may have one type or multiple types of lights. The outcome of the rendering process is a rendered model.

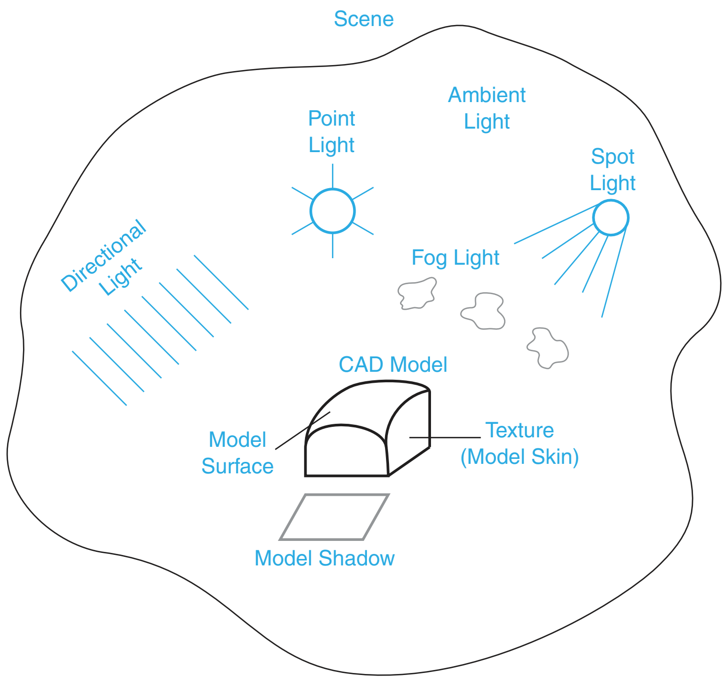


FIGURE 7.1
A rendering scene

SolidWorks offers different types of lights. You can turn any or all of the following lights on or off for a given scene:

1. Ambient light: This illuminates the model evenly from all directions. Ambient light has intensity and color that you can modify. You cannot delete or add additional ambient lights.
2. Directional light: This light is placed at infinity from the scene. Thus, it shines parallel rays from a given direction onto the CAD model. You can modify the intensity, color, and position of this light. The directional light simulates sunlight.
3. Point light: This is a concentrated light that comes from a source located at a specific location in the model space. This light source emits light in all directions. You can modify the intensity, color, and position of this light.
4. Spot light: Think of this as a beamed light source. The light comes from a restricted, focused light with a cone-shaped beam that is brightest at its center. You can shoot (aim) the spot light at a specific area of the model. You can modify the intensity, color, and position of the light source. You can also adjust the angle through which the beam spreads.
5. Fog light: This simulates scattering of the light by fog. It enables the effect of a fog light. You can specify fog lights in point or spot light. You can control the light intensity, which controls the brightness of the fog light effect.

7.3 Rendering Models

Rendering is a digital method to simulate how CAD models respond to different types of lighting around them. The response depends on the types of existing lights in the model scene (environment) and the type of the model material. The model material affects how the model reflects the lights that are shined on its surfaces. The goal of rendering is to display a CAD model on the screen as it would look in real life.

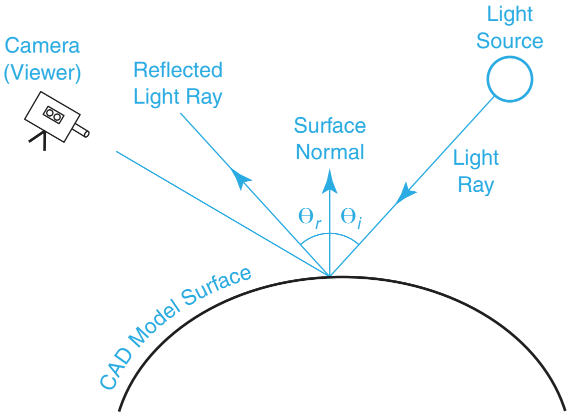
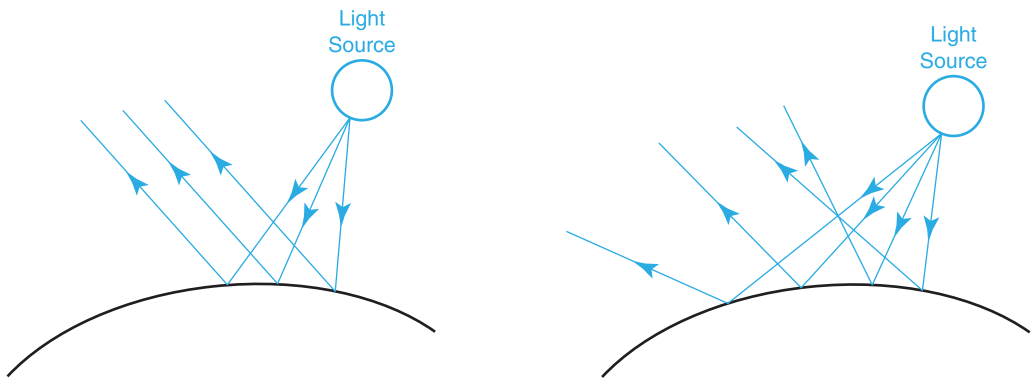


FIGURE 7.2
A rendering model

We use rendering models to render scenes after we create them. Figure 7.2 shows a typical rendering model. The light reflection occurs with respect to the surface normal vector at the point of contact between the surface and the light ray. The angles θ_i and θ_r are, respectively, the angle of incidence and the angle of reflection. A light source in the scene sends light rays to the CAD model surfaces (faces). When a light ray hits a surface, it is reflected off it. If the light is reflected evenly in one direction (Figure 7.3A, i.e., $\theta_i = \theta_r$), we have a shiny surface (known as a *specular surface*) like a mirror. If the light is reflected unevenly in many directions (Figure 7.3B, i.e., $\theta_i \neq \theta_r$), we have a dull surface (known as a *diffuse surface*) like wood and paper. The scene camera (viewer's eye) receives the reflected rays from any direction that is not necessarily coincident with the reflected rays. All these angles, scene lights, and variables are used to calculate the rendered image. These calculations are beyond the scope of this book and are not covered here.



(A) Shiny (specular) surface

(B) Dull (diffuse) surface

FIGURE 7.3
Types of reflection

The computational algorithm that performs these calculations is known as *ray tracing* and is not covered here. The algorithm requires intensive memory and computations. It is even more expensive if it uses anti-aliasing techniques to increase the quality of the rendered scene.

Caustic effects are the result of combined reflective and diffuse reflections. These effects result from interactions between different objects in the scene or from cascading reflections from surfaces of one object in the scene. For example, a light emitted from a light source goes through one or more specular and/or diffuse reflections before reaching the viewer's eye. Consider the sun shining on a swimming pool. By the time sunlight reaches the bottom of the pool, it gets reflected more than once going through the water, hits the pool floor, and goes through diffuse reflection before reaching the viewer's eye. What the viewer sees are the caustic effects on the pool floor.

7.4 Decals

A **decal** is an image that you would display on a face(s) of a CAD model. You use decals for marketing purposes. SolidWorks provides a decal library that you can use or add to. You can drag and drop decals from the decal library onto the current CAD model

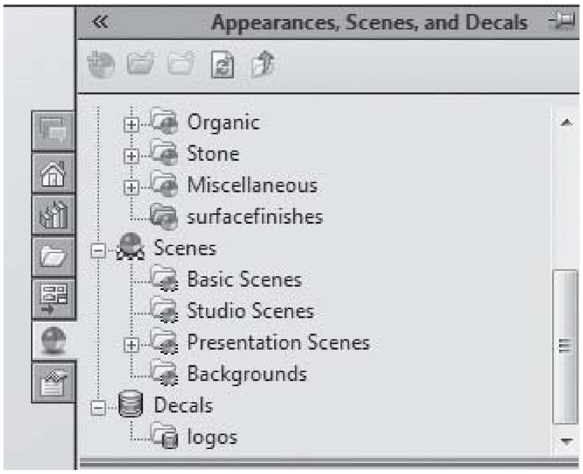


FIGURE 7.4
Decals library

displayed in the graphics pane and manipulate them. To access the library, click the **Appearances, Scenes, and Decals** tab in the task pane shown in Figure 7.4. The figure also shows the **Decals** library, which resides in this folder: *C:\Program Files\SolidWorks Corp\SolidWorks\data\graphics\Decals\Logos*. We can also access our own locally saved decals in any local folder by clicking this sequence: **PhotoView 360** on **Office Products** tab > **Edit Decal** on the **Render Tools** tab that opens up > **Browse** button.

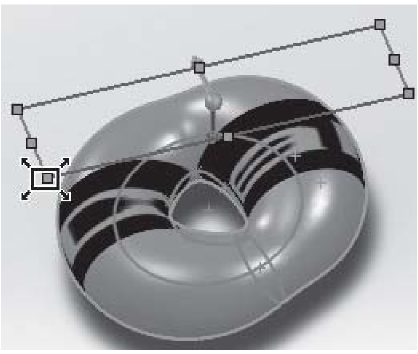
Decals have an order that controls their visibility. If multiple decals are applied, they are rendered in the order applied. If a decal is applied on top of another, it appears on top. You can use this fact to display overlapping decals to create fancy visual effects.

Example 7.1 Create a decal on the free-form torus shown in Figure 4.2.

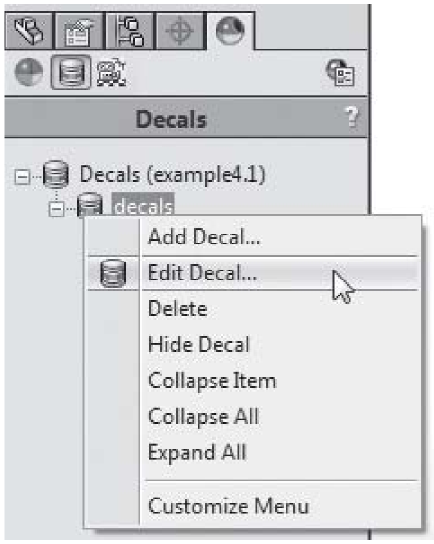
Solution

Step 1: Open *example4.2* part file: **File > Open >** locate the part file **> Open.**

Step 2: Apply decal to model: **Appearances, Scenes, and Decals** tab shown in Figure 7.4 > **Decals** > **logos >** drag a decal and drop onto torus face > grab a handle (small square) to resize the decal or the axes to adjust decal location > ✔.



Step 3: Edit decals: **DisplayManager** tab (rightmost tab on left pane) > Expand **Decals** node > Right click the **decal** node > **Edit** as shown.



Note: You can save you own logo locally. Use this sequence to add it to a model face: **PhotoView 360** on **Office Products** tab > **Edit Decal** on the **Render Tools** tab that opens up > **Browse** button.

HANDS-ON FOR EXAMPLE 7.1. Download your institution's logo, save it locally, and use it to replace the example decal.

7.5 Textures

Texture is also known as *surface finish* in SolidWorks. You can control the appearance of CAD models at three levels: scene lighting, textures, and material. We may assign material to a model, apply texture to it, and/or apply scene around it. The visualization effects of scene lights are limited to using different types of light and colors. Scene lighting cannot display the material look (e.g., metal, plastic, brick). We can assign a texture map to a model.

Texture maps are based on image files; that is, a texture map is an image of the texture, stored in an image file. A texture file is shrink-wrapped around a CAD model (similar to a decal), just like wallpaper or gift wrap. We need to select the location and orientation of the texture relative to the model before applying the texture. This is analogous to wrapping something in gift wrap. We locate and orient the gift relative to the wrap.

We access texture through **PhotoView 360**: **PhotoView 360** on the **Office Products** tab **>** **Render Tools** tab that opens up **>** **Edit Appearance** to open the **PropertyManager** tab > **Advanced** tab. The **Advanced** tab has four tabs that we use in examples and tutorials. When you select appearance, you are able to specify a material such as steel or plastic.

7.6 Materials

Materials are used in CAD/CAM for multiple purposes: CAD, CAM, and appearance. For CAD, we use them to calculate mass properties, perform finite element analysis, do dynamic analysis, and accomplish other tasks. For CAM, we use materials to select machining parameters such as cutting speed, feed rate, or cutting conditions (coolant on/off). For rendering, materials affect the final look of the CAD model. Specifying texture along with material makes the rendered CAD model look almost real.

SolidWorks comes with a materials library. We can also create our own new material types or edit existing ones by clicking this sequence: Right-click **Material <not specified>** (node in features tree) > **Edit Material**. This sequence opens the **Material** window shown in Figure 7.5. We can assign a material to the part by selecting from the

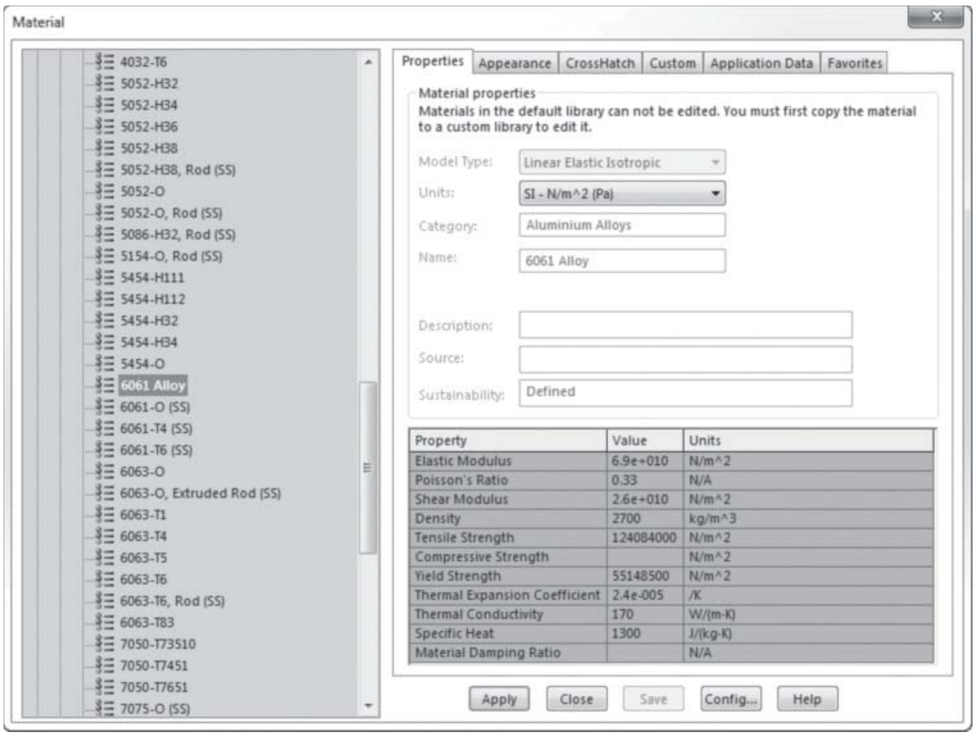


FIGURE 7.5
Materials library

menu on the left. We can also add our own material by adding a custom material: right click any material node > **New Library**.

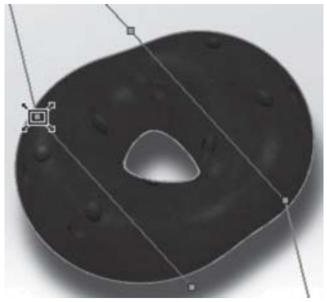
Example 7.2 Add material and texture (surface finish) to the free-form torus shown in Figure 4.2. Use your own custom texture map file, Stamped.jpg.

Solution We add material and texture as follows.

Step 1: Open *example4.2* part file: **File > Open >** locate the part file **> Open.**

Step 2: Apply material to model: Right click the **Material** node in features tree > **Edit Material** > select **BUTYL** (under **Rubber** node in the **Material** window that opens up) > **Apply** > **Close.**

Step 3: Apply texture (surface finish) to model: **PhotoView 360** on **Office Products** tab **> Render Tools** tab that opens up **> Edit Appearance** to open the **PropertyManager** tab > **Advanced** tab > **Surface Finish** tab > **From File** from the dropdown > locate and select *Stamped.jpg* image > grab the handle to resize the image > ✔.



Note: You may use your own surface finish local image file.

HANDS-ON FOR EXAMPLE 7.2. Change the torus material to ABS plastic. Also use the *knurl.jpg* texture (search for the file).

7.7 Appearance and Transparency

Using appearance, we can add material look and feel to models without adding the physical properties of materials; that is, we add visual characteristics to the models. We can assign appearance to parts, features, faces, or surfaces. As part of appearance, we can specify how transparent the model is (available under the **Illumination** tab under the **Advanced** tab). Transparency allows us to see through the model. Zero transparency means the model is like a brick wall; you cannot see through it. Full or maximum transparency (with a setting of 1) means the model is like a glass wall; you can see through it.

7.8 Background and Scenes

We are able to set the background as part of **PhotoView 360**. (We can set the background color as part of the system options by clicking **Tools > Options > System Options > Colors**.) We can drag and drop scenes from the scenes library onto the current CAD model displayed in the graphics pane. To access the library, click the **Appearances, Scenes, and Decals** tab in the task pane shown in Figure 7.4. The figure also shows the **Scenes** library, which resides in this folder: *C:\Program Files\SolidWorks Corp\SolidWorks\data\graphics\Scenes*. The left pane of the screen also has the **DisplayManager** tab (the rightmost tab).

7.9 Cameras and Camera Sleds

Thus far, we are able to create scenes with different types of lights and backgrounds. As shown in Figure 7.2, we can add cameras to the scene. We add a camera in a given location in the scene. The main objects in the scene are the CAD model and the camera, which can move relative to each other. The camera can rotate (float) around the CAD model while it is fixed in the 3D modeling space; or the CAD model can rotate (float) around the camera being fixed. Obviously, moving (floating) the camera around the CAD model is beneficial because it allows us to view the model from different angles in space.

The floating camera model requires guiding the camera motion in space by creating target points or paths (curves) for the camera's travel. SolidWorks calls these paths *camera sleds*. Think of it as though you mount the camera on the sled and let it ride it. As the sled moves, the camera moves with it. Sleds are usually used to create animation.

7.10 Animation

Animation is a useful visualization concept whereby we observe a continuous motion of an object in an attempt to understand its dynamic behavior with time. There are two types of animation: real time or playback. In real-time animation, we physically follow the motion of the physical object in space as it happens and record its motion. Playback animation is what we use. (Hollywood also uses it to create motions of cartoon characters and movies.) In playback animation, we create key frames at key points in the time domain. We assemble the key frames and display them in a continuous fashion to create the illusion of motion. We can interpolate between the key frames linearly or nonlinearly.

The camera sled moves along a defined path (linear or nonlinear) between the key frames (target points). As the camera moves, it takes shots of the model from different angles and displays them to the viewer in a continuous fashion. This is done by creating a sequence of camera views of the model from different angles (points on the sled path) and then displaying them. The camera may also move through the model to reveal its inside. A camera sled is a "dummy" component we create in the CAD model. We attach the camera to a sketch entity on the camera sled. We can hide the sled (the dummy component) from the sketch to view only the camera movement during the animation.

We refer to this type of animation as *camera-based animation*, as opposed to kinematics- or dynamic-based animation. We use the animation in motion studies. For example, we may use kinematics to create key frames of the model and then use the camera to combine them and show the animation.

7.11 Tutorials

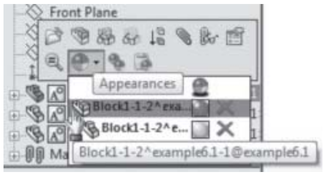
The theme for the tutorials in this chapter is to practice the rendering concepts covered in this chapter.

Tutorial 7–1: Apply Colors to Objects

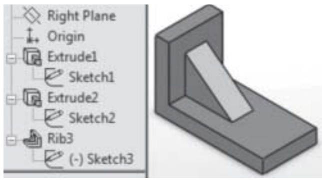
Thus far, we have not used colors in the CAD models we create (parts and assemblies). This tutorial shows how to apply colors to assemblies, parts, features, and faces. Note: If **PhotoView 360** does not show in the **Office Products** tab, add it in as follows: **Tools > Add-Ins** > check off **PhotoView 360** checkbox > **OK**.

Step 1: Open *example6.1* assembly file: **File > Open >** locate the part file **> Open.**

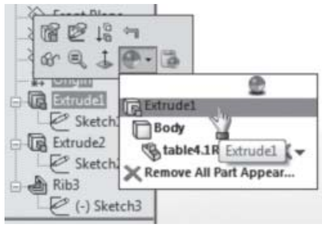
Step 2: Apply colors to assembly model: Click a part in the assembly tree > dropdown list of **Appearances** from context toolbar that opens up > first feature on menu as shown > select blue color from palette > ✔ > repeat for small block, but use a yellow color > ✔ > repeat for shaft, but use a green color > ✔.



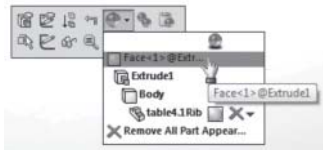
Step 3: Create part to apply colors: Create the L bracket shown as three features: horizontal extrude, vertical extrude, and a rib.



Step 4: Apply colors to part features: Click the first feature in the features tree > dropdown list of **Appearances** from context toolbar that opens up > first feature (*Extrude1*) on menu as shown > select blue color from pallet > ✔ > repeat for second feature (vertical extrude), but use a yellow color > ✔ > repeat for third feature (rib), but use a green color > ✔.



Step 5: Apply colors to part faces: Click top face of horizontal extrude > dropdown list of **Appearances** from context toolbar that opens up > first feature (*Face<1>*) on menu as shown > select blue color from palette > ✔.



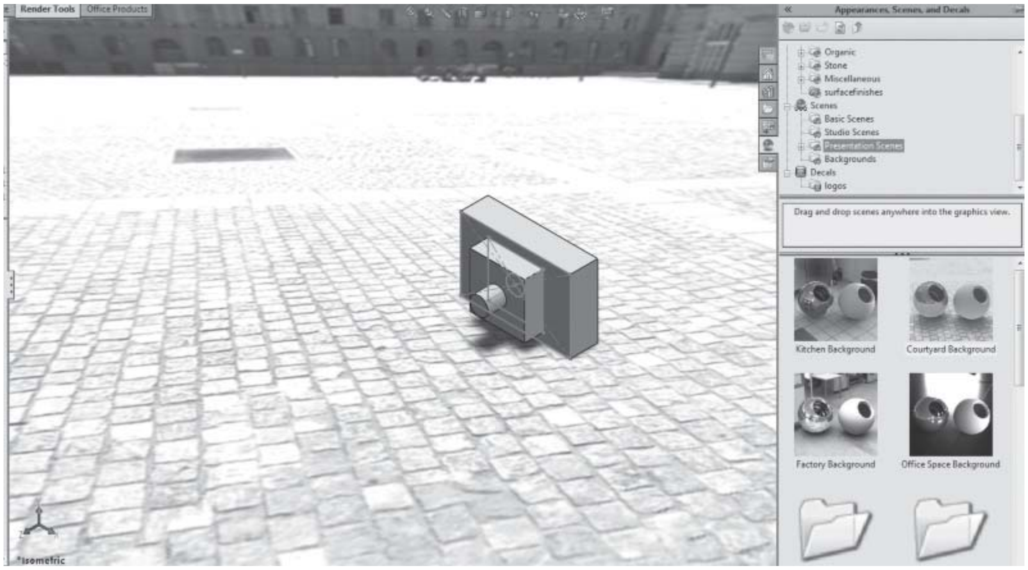
Note: The first colors you apply override subsequent colors.

> ***HANDS-ON FOR TUTORIAL 7.1.*** Create a hole in the horizontal extrude. Add a blue color to the hole face. Add a yellow color to the hole feature. Which color do you see and why? Hint: Reverse the order of applying colors and observe.

Tutorial 7–2: Apply Background and Scene

Three types of scenes are offered under the **Scenes** node in the task pane as shown in Figure 7.4: basic, studio, and presentation. The difference is that some scenes are more elaborate than others. When we add scenes, they override existing scenes; they do not remove them. We must delete existing scenes before applying new ones if we need to investigate effects of different scenes.

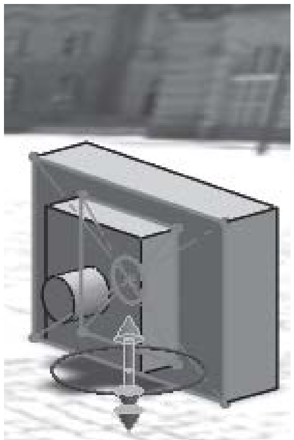
Step 1: Open *example6.1* assembly file: **File > Open >** locate the part file **> Open.**



Step 2: Apply scene: Click the **Appearances, Scenes, and Decals** tab of the task pane > expand **Scenes** node > **Presentation Scenes** > drag one of the scenes shown to the right of the screen and drop onto the graphics pane to add scene as shown above.

Step 3: Manipulate scene: Step 2 has added the scene to the **Scene** node of the **Display-Manager** tab (third from features tree tab) of the model > click the tab to open it > right click the **Scene** node > select any action from the menu shown here.

Step 4: Edit scene: Select **Edit Scene** from menu above > drag handle that shows up to position object (assembly) in scene > drag the blue circle that shows up and rotate the mouse around it to view the scene from different angles.



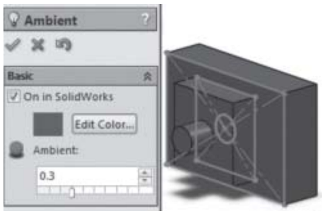
HANDS-ON FOR TUTORIAL 7.2. Turn on/off model shadow and use perspective view. Use the **View Settings** icon in the **Heads-up View** toolbar of graphics pane shown here.

Tutorial 7–3: Apply Lights to Scene

The different types of light shown in Figure 7.1 are available in SolidWorks under the **DisplayManager** tab. Expanding the **Lights** node shows ambient and directional lights. Right click **Scene Illumination** to access the other types of lights.

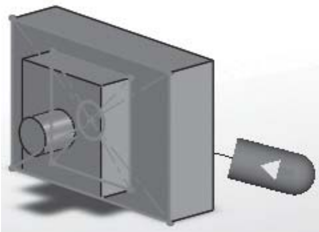
Step 1: Open *example6.1* assembly file: **File > Open >** locate the part file **> Open.**

Step 2: Add ambient light: **DisplayManager** tab > **Scene, Lights, and Cameras** icon (hover to read it) > expand **Lights** node > double click **Ambient > Edit Color** button > select blue from the **Color** window that opens up > **OK** > ✔.

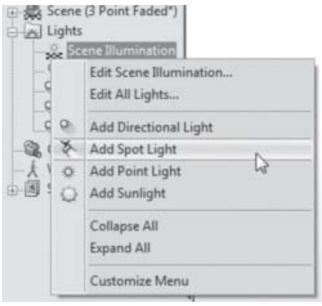


Note: There is no delete ambient light per se. To remove ambient light, right click **Ambient** > **Edit Light** > **Edit Color** button to assign white color (i.e., no color).

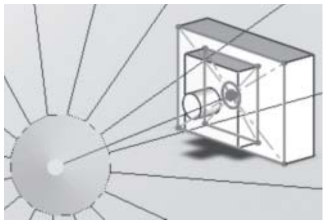
Step 3: Add directional light: Double click one of the directional light nodes under **Lights** node > drag and move the handle that shows to change light direction > adjust light settings shown in the left pane.



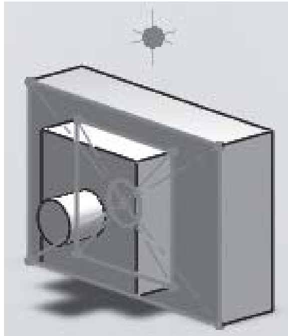
Step 4: Open spot light editor: Right click **Scene Illumination** node > **Add Spot Light** from menu shown.



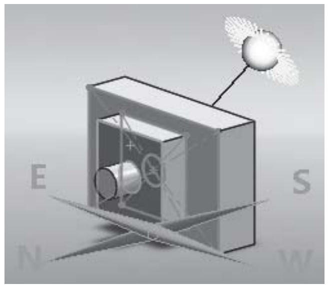
Step 5: Add spot light: **>** Step 4 adds a spot light to scene > drag and move the handle that shows to change light direction > adjust light settings shown in the left pane > ✔.



Step 6: Add point light: Repeat Step 4 and Step 5 but use point light.



Step 7: Add sunlight: Right click **Scene Illumination** node > **Add Sunlight** > select a face to change North direction > ✔.

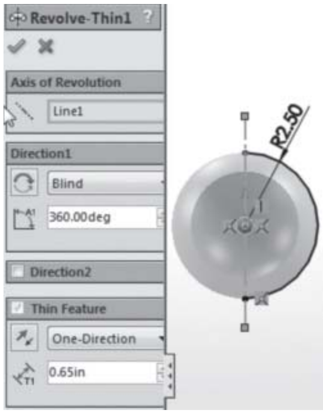


HANDS-ON FOR TUTORIAL 7.3. Combine all lights of the tutorial to create a comprehensive scene.

Tutorial 7–4: Add Material and Transparency

We use the material of a part to add appearance. SolidWorks has built-in information about many common materials found in engineering and in nature. This tutorial covers the basics of how to change the material and make it transparent. Transparency enables us to see through an object, so we can look inside it. All dimensions are in inches.

Step 1: Create a thin revolve: **File > New > Part > OK > Front Plane > Revolved Boss/ Base** on **Features** tab > **Center Point Arc** on **Sketch** tab > click origin and sketch and dimension as shown **> Centerline** on **Sketch** tab > sketch as shown > exit sketch **> No** (to keep cross section open to create thin revolve) > enter 0.65 for thickness **> ✔ >** **File > Save As >** *tutorial7.4* **> Save**.



Step 2: Add mahogany material**:** Right click **Material** node in features tree **> Edit Material > Wood > Mahogany > Apply > Close**.

Step 3: Change transparency: **Edit Appearance** on **Render Tools** tab **> Advanced** tab **> Illumination** tab > locate **Transparent amount** slider and drag it to 0.70 (min value is zero and max value is 1) **> ✔ >** select **Shaded** mode (to make model looks smoother) from **Display Style** dropdown in the **Heads-up View** toolbar on top of graphics pane.

Step 4: Change material to water: Right click **Material** node in features tree **> Edit Material > Other Non-Metals > Water > Apply > Close >** select **Shaded** mode (to make model looks smoother) from **Display Style** dropdown in the **Heads-up View** toolbar on top of graphics pane.

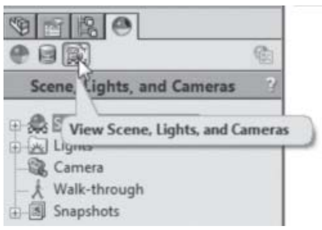
HANDS-ON FOR TUTORIAL 7.4. Change the transparency setting of water to zero and 1 and observe the difference.

Tutorial 7–5: Add Camera to Scene

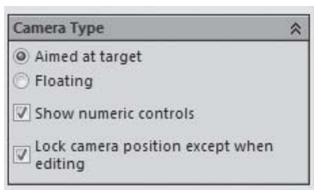
We add a camera to view the model. A camera can be added to look at the part or assembly in a very specific way. Multiple cameras can be used as well as moved around the part to create an animation effect of the model. When using the Camera wizard, there are many position, aim, and focal options. When the camera is created, it will be accessed through the view orientations menu (e.g., the front, top, and isometric views) located on the **Heads-up View** toolbar.

Step 1: Open *example6.1* assembly file: **File > Open >** locate the part file **> Open.**

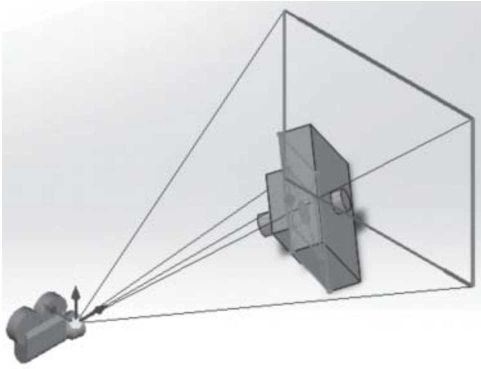
Step 2: Open scene editor to access Camera: **DisplayManager** tab (last tab as shown) **> View Scene, Lights, and Cameras** icon.



Step 3: Add a camera: Right click **Camera** from menu shown in Step 2 > **Add Camera >** check off **Show numeric controls** checkbox shown.



Step 4: Set camera position: Observe that **Camera Position** defaults to **Spherical** (you may set it to



Cartesian) > click **Zoom to Fit** icon in **Heads-up View** toolbar to see camera > drag any camera axis (similar to exploded view axes) and move to position > ✔.

Step 5: Show camera in graphics pane: Right click **Camera** node > **Show Cameras** to view the camera just added in the graphics pane (this is a toggle; right click again > **Show Cameras** to take it away).

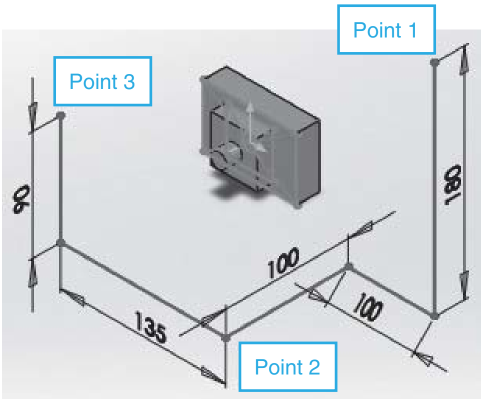
HANDS-ON FOR TUTORIAL 7.5. Use camera **Cartesian Position** and change the Cartesian coordinates of the camera to the following three locations: (0, 0, 60), (60, 60, 0), (60, 60, 60). Then change the type of camera to **135mm Telephoto** (under **Field of View** dropdown). Observe the effect of camera location and type on model rendering.

Tutorial 7–6: Create Motion Study

We add a camera to motion study the model. Cameras can be used to view the model from different perspectives as opposed to just from the graphics pane. This tutorial shows how to view the model from the graphics pane as well as with the camera created in the previous tutorial. The camera travels between multiple points over time. All dimensions are in mm.

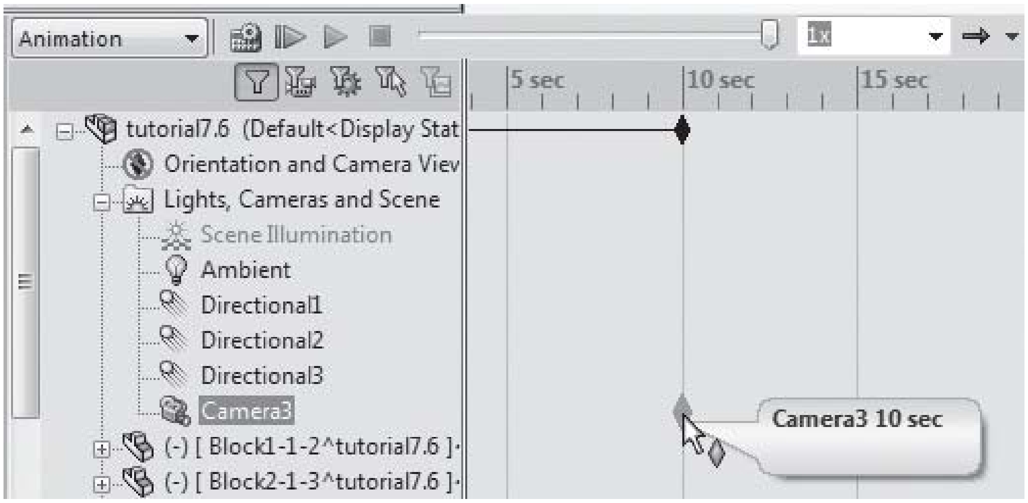
Step 1: Open *example6.1* assembly file: **File > Open >** locate the part file **> Open.**

Step 2: Create 3D sketch: Sketch dropdown on **Sketch** tab > **3D Sketch > Line** on **Sketch** tab > sketch lines shown

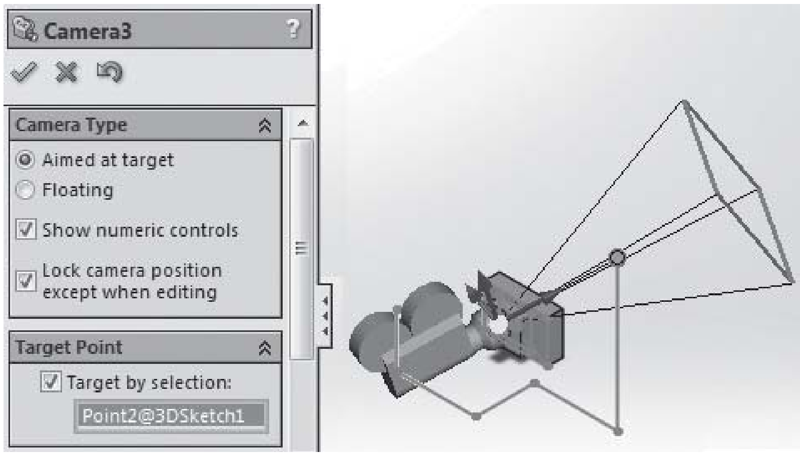


Note: All lines are along the *X*-, *Y*-, or *Z*-axis. Use the **Tab** key to change between sketch planes. The three points shown will be used to locate the camera at various times of the motion study.

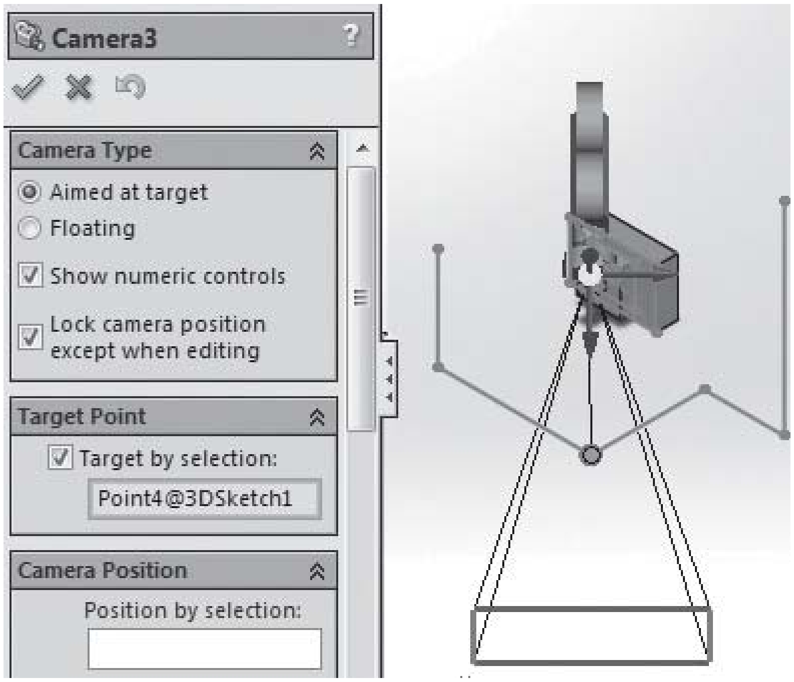
Step 3: Create first key point in motion study: Click **Motion Study 1** tab at bottom of screen **>** expand **Lights, Cameras and Scene** node in motion study **>** drag the **Key** (diamond shape) next to *Camera3* node (created in Tutorial 7–5) out to 10 sec on the timeline as shown.



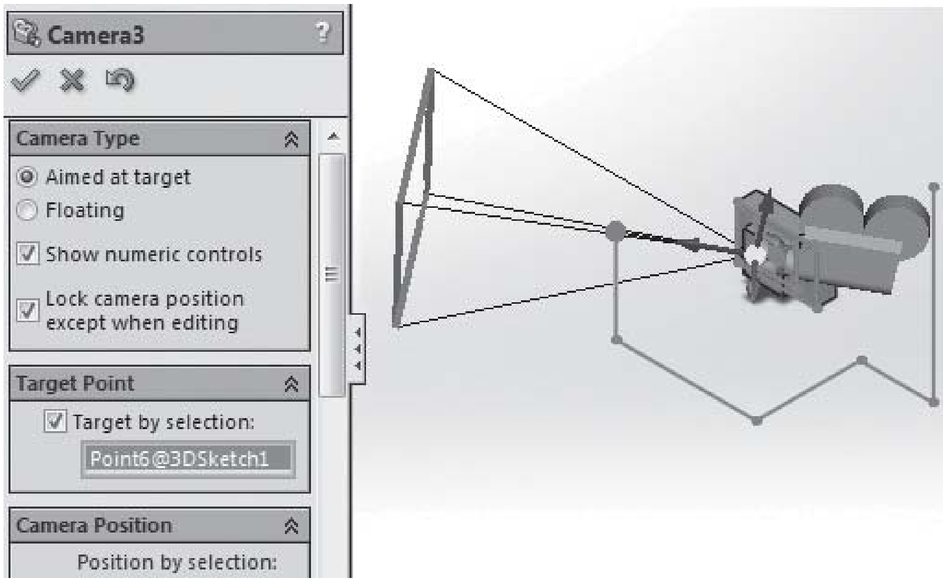
Step 4: Set camera position at key point 1 created in Step 3: Double click the key pointing to the camera (at 10 sec) > this opens up the camera pane under **Target Point**, click **Target by Position** box shown and select **Point 1** from the 3D sketch shown on graphics pane > ✔.



Step 5: Create second key point and set camera position: Repeat Steps 3 and 4. This time, drag the camera key to 15 sec mark > ✔.

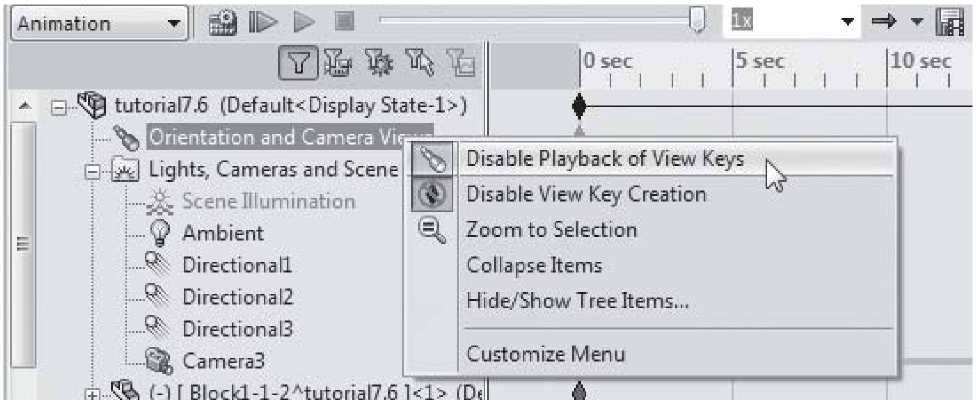


Step 6: Create third key point and set camera position: repeat Steps 3 and 4. This time drag the camera key to 20 sec mark > ✔.

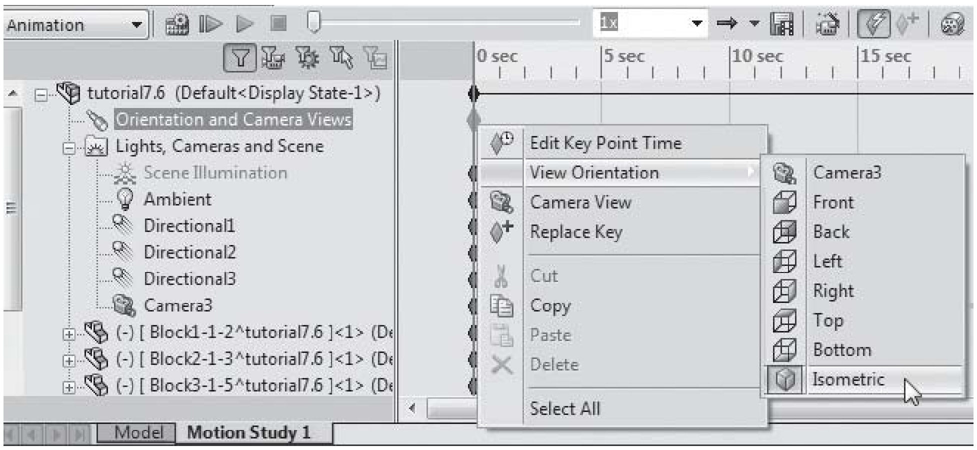


Step 7: Show camera and hide 3D sketch: Right click **Camera3** > Show Cameras if camera is not shown in graphics pane > click features tree tab > click *3DSketch1* > **Hide** from context toolbar (turn off sketch to see a cleaner motion study).

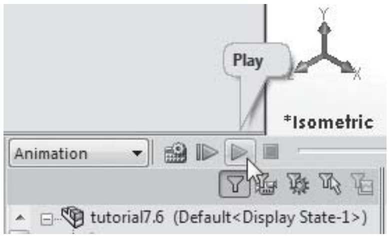
Step 8: Set key properties: Right click **Orientation and Camera Views** as shown > select first item as shown (**Disable Playback of View Keys**).



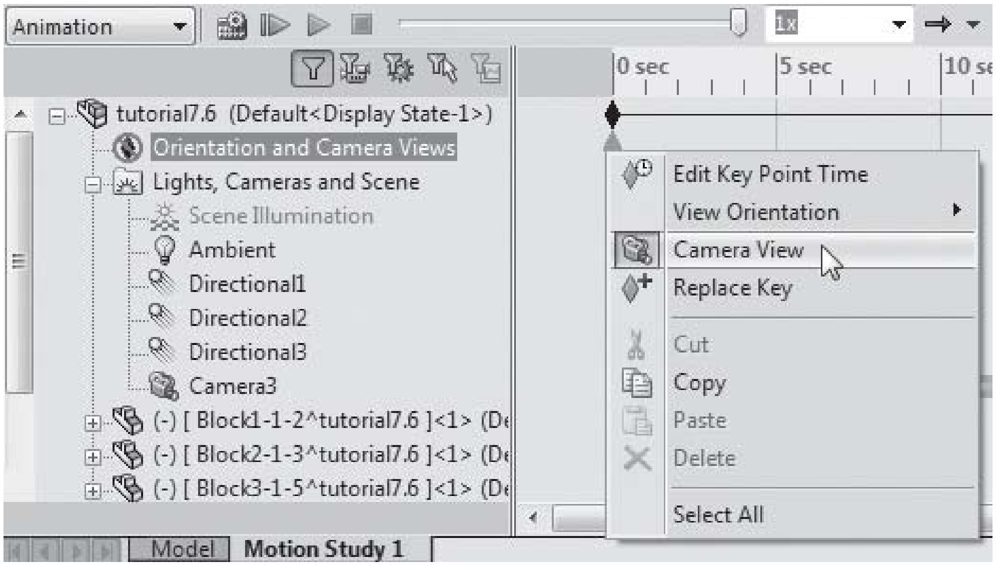
Step 9: Set view to Isometric: Right click on the key to the right of the **Orientation and Camera View**. Set the orientation to **Isometric** as shown.



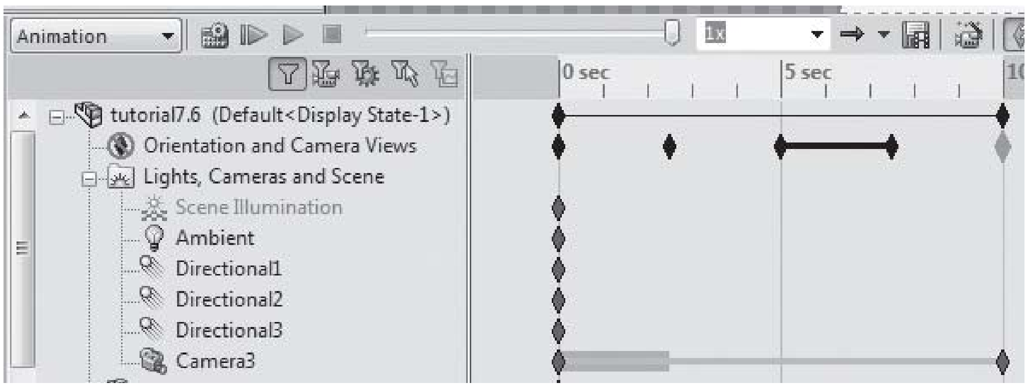
Step 10: Review the motion study: Click the green **Play** button in the motion study to watch the camera orbit the part.



Step 11: View the motion study from camera: Right-click the **Key** to the right of the **Orientation and Camera Views.** Select **Camera View** as shown to view the model from the camera as it orbits the part > **Play** icon shown in Step 10 to play the study.



Step 12: View the motion study both from isometric position and from camera: Drag the **Key** to the right of the **Orientation and Camera Views** out four times to create a key every 2.5 s as shown > set the first two keys to view from the isometric position. Set the other two keys to view from the camera view > click the **Play** icon.



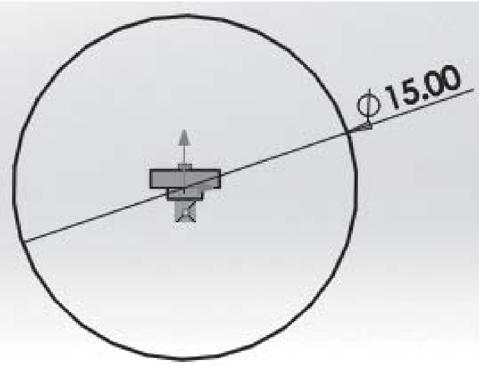
HANDS-ON FOR TUTORIAL 7.6. Extend the motion study to 20 sec.

Tutorial 7–7: Create Camera-Sled Based Animation

We add a camera sled to the model and use it to guide the camera through a motion study. The camera sled is a dummy object used to guide a camera through a model. The sled is forced to follow a path to which the camera is attached. All dimensions are in inches.

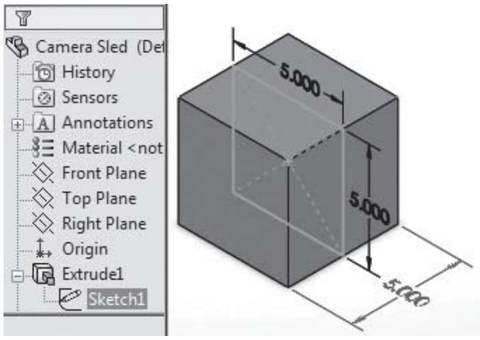
Step 1: Open *example6.1* assembly file: **File > Open >** locate the part file **> Open.**

Step 2: Sketch camera sled path: **Top Plane > Circle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > **Save.**



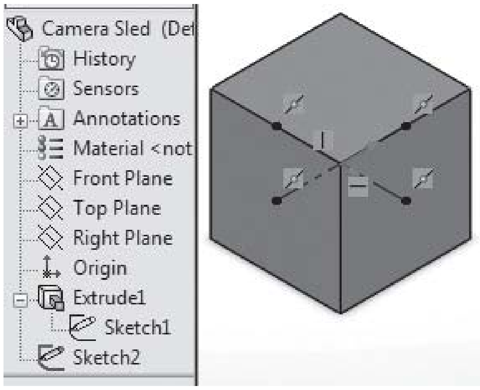
Note: The size of the circle is irrelevant.

Step 3: Create camera sled part: **File > New > Part > OK > Front Plane > Extruded Boss/Base** on **Features** tab > **Center Rectangle** on **Sketch** tab > click origin and sketch and dimension as shown > exit sketch > enter 5 for thickness **D1** > **Mid Plane** for extrusion direction > ✔.



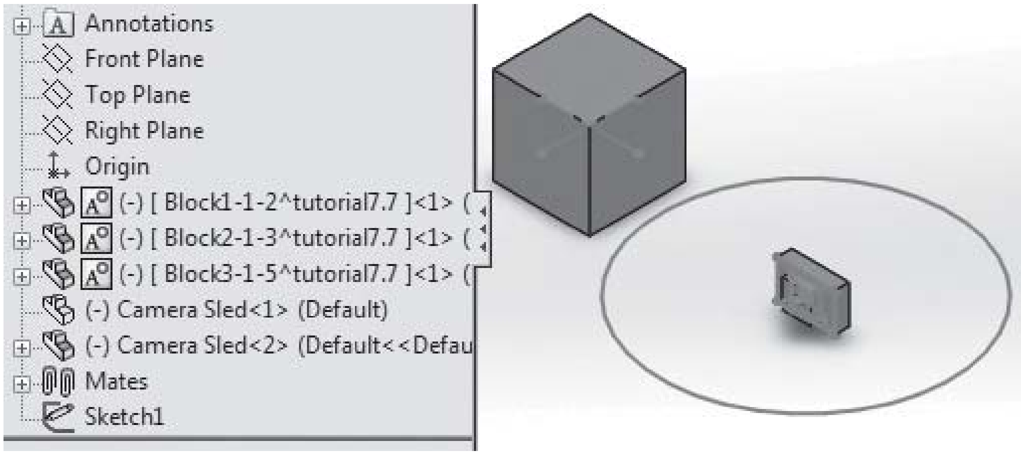
Note: The size and shape of the camera sled part are irrelevant.

Step 4: Create *Sketch2* in camera sled part: Top face of feature created in Step 3 > **Center Line** on **Sketch** tab > sketch two lines as shown > exit sketch > **File > Save As** > *cameraSled* > **Save.**

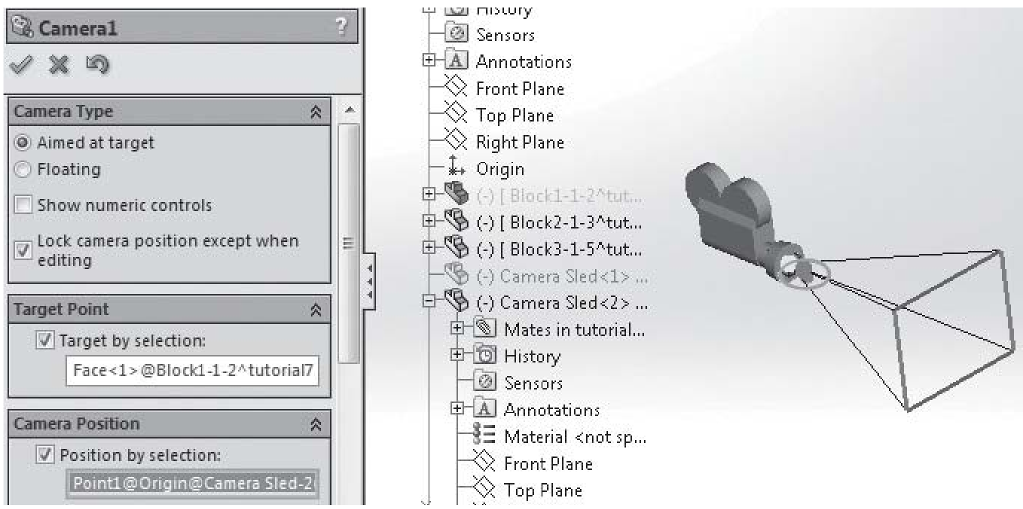


Note: Sketch2 is used to locate the sled and the camera.

Step 5: Insert camera sled into assembly: **Insert Components** on **Assembly** tab > locate and insert *cameraSled* part as shown > **Mate** on **Assembly** tab > **Coincident** between **Top Plane** of assembly and **Top Plane** of *cameraSled* > ✔ > **Coincident** between the origin of *cameraSled* and circle created in Step 2 > ✔ > **Parallel** between **Right Plane** of assembly and **Right Plane** of *cameraSled* > ✔ > ✔.

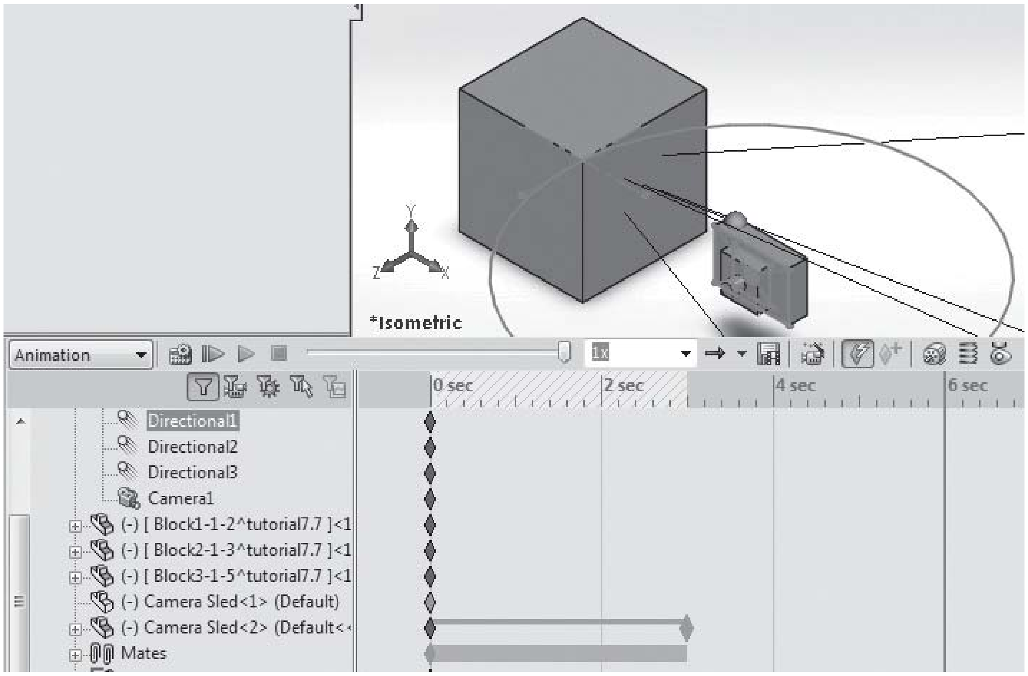


Step 6: Add camera to scene: **Scene, Lights, and Cameras** tab from left pane > right click **Camera > Add Camera > click Target by selection** box > top face of big block of assembly > click **Position by selection >** origin of *cameraSled* part > ✔ > right click Camera1 > **Show Cameras**.



Step 7: Create first key for motion study: **Motion Study 1** tab > drag the *cameraSled* key out to 3 sec as shown > drag *cameraSled* block in graphics pane 1/4 around the circle created in Step 2.

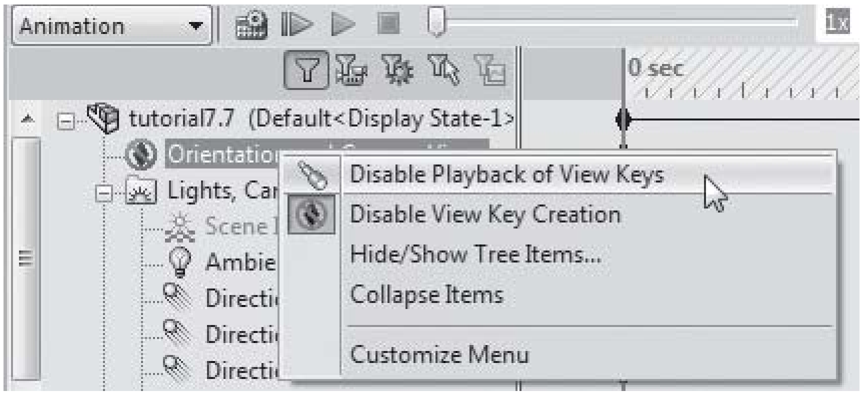
Note: The timeline automatically fills in a time. For this tutorial, drag it back to the 3 sec mark.



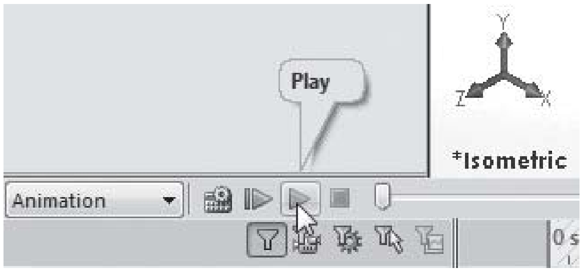
Step 8: Create second key for motion study: **Motion Study 1** tab > drag the *cameraSled* key out to 6 sec > drag *cameraSled* block in graphics pane 1/4 around the circle created in Step 2.

Note: The timeline automatically fills in a time. For this tutorial, drag it back to the 6 sec mark.

Step 9: Set camera view: Right click **Orientation and Camera Views** as shown > select first item as shown (**Disable Playback of View Keys**).



Step 10: Display camera: Right click **Camera1** in motion study tree **> Camera View**.



Step 11: Review motion study: play green arrow.

HANDS-ON FOR TUTORIAL 7.10. Create a motion study such that the assembly moves in a straight line as the camera moves around the circle. *Hint:* Set the part to float and then lock it to two out of the three assembly planes. Use the third degree of freedom to move the part along a straight line.

problems

1. Describe the elements of a digital rendering scene.
2. List the five types of lights that SolidWorks uses. What type of light rays does each type provide?
3. Sketch a typical rendering model. What type of reflections do a shiny surface and a dull surface reflect?
4. Search and find a cool texture image. Save it to your desktop. Use it to render a CAD model.
5. Repeat Problem 4, but use transparency to see through the CAD model.
6. Create a camera and camera sled to animate any of the assembly models shown in Chapter 6 problems.

This page intentionally left blank

PART

III

Advanced Part Modeling

The primary goal of Part III is to explore and cover in detail the topics of curves, surfaces, sheet metal, weldments, and sustainable design. This part shows the real power of geometric modeling as well as how to model and design complex parts such as computer mice, fan blades, and hair dryers. Two of the important modeling concepts covered here are 3D curves and surfaces. Combining these concepts enables us to model any complex shape we come across.

Chapter 8 (Curves) covers the details of curves, their parametric representation, and their types of 2D and 3D representation. Chapter 9 (Surfaces) extends the curve theory and covers the parametric representation of surfaces, the available types of surfaces we can use in modeling, and how to use surfaces to create solid models. Chapter 10 (Sheet Metal and Weldments) shows how we can create sheet metal parts and weldments. Chapter 11 (Sustainable Design) closes Part III by covering the important topic of environmentally conscious design, sustainability and life cycle assessment (LCA), and analysis.

This page intentionally left blank

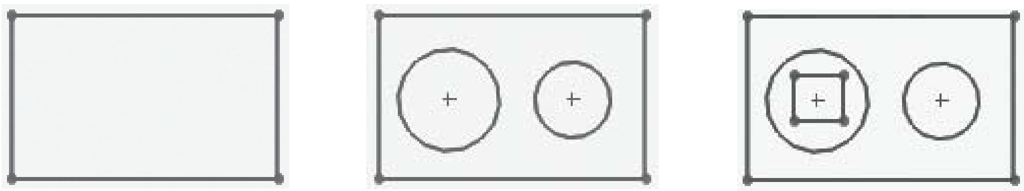
CHAPTER

8

Curves

8.1 Introduction

Curves form the backbone of geometric modeling and creating solid models. A sketch consists of multiple curves that are connected to form closed contours (loops). Loops may be nested. A sketch that consists of only one contour generates a solid without holes, whereas a sketch with multiple contours generates a solid with holes in it. Figure 8.1 shows examples. A one-contour sketch is demonstrated in Figure 8.1A. You can create a sketch with only one-level nesting; that is, an outside contour with one or more disconnected contours inside it, as shown in Figure 8.1B. However, if you attempt to use more than one-level nesting as shown in Figure 8.1C, the solid creation operation fails.



(A) One-contour sketch

(B) One-level nesting sketch with two nested contours

(C) Two-level nesting sketch: rectangle inside left circle

FIGURE 8.1
Nesting contours to create solids

Mathematically, two families of curves exist. The first family is the analytic curves. These curves have closed-form equations defining them. Examples of analytic curves include lines, circles, ellipses, parabolas, and hyperbolas. This family is also known as *conics* because the curves result from intersecting a cone with a plane. For example, intersecting a cone with a plane passing through its axis produces a line. Intersecting a cone with a plane perpendicular to its axis produces a circle; and, if the plane is not perpendicular to the cone, the intersection results in an ellipse, parabola, or hyperbola.

The second family of curves is the synthetic curves. A **synthetic curve** is defined by a polynomial that uses a set of data points. These data points control the curve shape. Examples include a cubic curve and a B-spline (or a *spline,* as SolidWorks calls it). Figure 8.2 shows sample curves with SolidWorks menus. These curves are accessible from the SolidWorks **Sketch** tab. The ellipse is defined by two radii (each connecting two opposite points, as shown in Figure 8.2A), and the spline is defined by points P_0, P_1, …, P_n.

Synthetic curves offer more flexibility in modeling than analytic curves. They are efficient to use for creating free-form shapes. All we need to do is to define the curve points (Figure 8.2B) by either clicking in the sketch area or entering (x, y, z) coordinates. After the curve is created, we can easily modify its shape by editing its points to change their locations.

Line

Ellipse

(B) Spline

(A) Analytic curves

(B) Synthetic curves

FIGURE 8.2
Families of curves

8.2 Curve Representation

Curves and surfaces can be described mathematically by nonparametric (explicit) or parametric equations. A point on a curve has (x, y) coordinates for planar (sketch) curves, and (x, y, z) coordinates for nonplanar curves. For a planar curve, an **explicit equation** relates the y coordinate to the x coordinate as follows:

$$y = f(x) \quad x_{\min} \le x \le x_{\max} \tag{8.1}$$

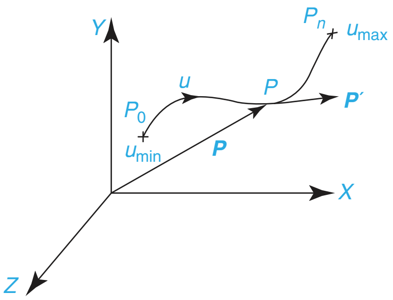


FIGURE 8.3
Parametric curve

Explicit equations for nonplanar curves are complex and are not supported by CAD/CAM systems because they do not lend themselves well to the CAD design environment.

A **parametric equation** is an equation that uses a parameter (e.g., u) to describe the (x, y) coordinates of a point on a planar curve or the (x, y, z) coordinates of a point on a nonplanar curve. Figure 8.3 shows the parametric representation of a curve. The parameter u starts with a minimum value, $u_{\min}$, at one end of the curve and finishes with a maximum value, $u_{\max}$, at the other end of the curve. The parameter increases in value from $u_{\min}$ to $u_{\max}$, thus defining the parameterization direction of the curve as indicated by the arrow on the curve. Any point P on the curve is defined by its position vector, $\boldsymbol{P}$, that is a function of u,

$$\boldsymbol{P} = \boldsymbol{P}(u) \qquad u_{\min} \le u \le u_{\max} \tag{8.2}$$

Alternatively, point P is defined by its (x, y, z) coordinates. Thus,

$$\boldsymbol{P} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix} \qquad u_{\min} \le u \le u_{\max} \tag{8.3}$$

The tangent vector $\boldsymbol{P}'$ at any point on the curve (Figure 8.3) is given by:

$$\boldsymbol{P}' = \frac{d\boldsymbol{P}}{du} = \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \qquad u_{\min} \le u \le u_{\max} \tag{8.4}$$

The tangent vector is an important concept in CAD/CAM applications such as mass property calculations and NC (numerical control) programming. For these two applications, we use the tangent vector at any point on the curve to calculate the normal vector to the curve at the same point. For mass properties, the direction of the normal vector is used to determine the inside (where material is) and the outside (where holes exist) of the solid. For NC programming, we move the cutting tool along the direction of the normal vector until it makes contact with the part surface to be machined. This minimizes the lateral (shear) forces on the cutting tool, which in turn reduces the chance of breaking the tool upon contact with the surface to be machined.

SolidWorks supports both explicit and parametric equations for curves. It allows the user to enter explicit equations for planar (sketch) curves. It does not support explicit equations for nonplanar curves. However, it does support parametric equations for both planar and nonplanar curves. It uses the parameter t instead of u. This chapter shows how to enter and use equations in SolidWorks.

8.3 Line Parametric Equation

Figure 8.4 shows the parametric representation of a straight line defined by two endpoints, P_0 and P_1. The parameterization direction of the line shown in Figure 8.4 goes from P_0 to P_1, indicating that we start sketching the line at P_0 and finish at P_1. The parametric equation of this line is given (in vector form) by:

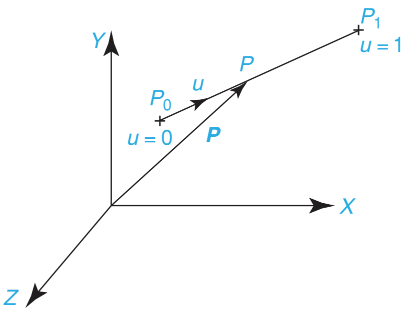


FIGURE 8.4
Parametric line

$$\boldsymbol{P} = \boldsymbol{P}(u) = \boldsymbol{P}_0 + u(\boldsymbol{P}_1 - \boldsymbol{P}_0) \quad 0 \leq u \leq 1 \tag{8.5}$$

or (in scalar form),

$$\boldsymbol{P} = \boldsymbol{P}(u) = \begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix} = \begin{bmatrix} x_0 + u(x_1 - x_0) \\ y_0 + u(y_1 - y_0) \\ z_0 + u(z_1 - z_0) \end{bmatrix} \quad 0 \leq u \leq 1 \tag{8.6}$$

Using Eq. (8.4), the tangent vector of the line is given by:

$$\boldsymbol{P}' = \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} x_1 - x_0 \\ y_1 - y_0 \\ z_1 - z_0 \end{bmatrix} \tag{8.7}$$

Equation (8.7) shows that the tangent vector is constant (independent of u), as expected. We can easily derive the line slope from the tangent vector. For example, the line slope in the XY plane is given by:

$$\frac{dy}{dx} = \frac{dy/du}{dx/du} = \frac{y'}{x'} \tag{8.8}$$

The elegance of the parametric representation is that it is independent of the dimensionality of the modeling space whether it is 2D (x and y coordinates only) or 3D (x, y, and z coordinates). In other words, use $z = 0$ in the 3D equations and we get 2D modeling. As a matter of fact, when we sketch in a sketch plane, the z value is set to zero; when we are done sketching, SolidWorks transforms the sketch "2D" WCS coordinates to "3D" MCS coordinates.

Example 8.1 A designer created a line connecting point (−3, 2, 1) to point (0, −4, 2). Find the line parametric equation, its midpoint, its endpoints, and its tangent vector.

Solution Using Eq. (8.6), the line parametric equation is

$$\boldsymbol{P}(u) = \begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix} = \begin{bmatrix} -3 \\ 2 \\ 1 \end{bmatrix} + u\begin{bmatrix} 3 \\ -6 \\ 1 \end{bmatrix} = \begin{bmatrix} 3u - 3 \\ -6u + 2 \\ u + 1 \end{bmatrix} \quad 0 \leq u \leq 1 \tag{8.9}$$

Differentiating Eq. (8.9) with respect to u, the tangent vector is:

$$\boldsymbol{P}' = \begin{bmatrix} x'(u) \\ y'(u) \\ z'(u) \end{bmatrix} = \begin{bmatrix} 3 \\ -6 \\ 1 \end{bmatrix} \tag{8.10}$$

The midpoint of the line occurs at $u = 0.5$. Thus, it is

$$\begin{bmatrix} -1.5 \\ -1 \\ 1.5 \end{bmatrix}$$

The endpoints of the line occur at $u = 0$ and $u = 1$ and are the same as the points given in the example. Thus, we can check whether Eq. (8.9) is correct by substituting the values $u = 0$ and $u = 1$.

HANDS-ON FOR EXAMPLE 8.1. Find the line equation if the designer reverses the order of the endpoints to sketch the line.

8.4 Circle Parametric Equation

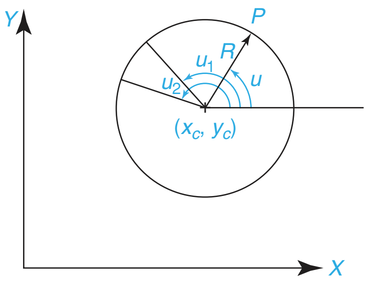


FIGURE 8.5
Parametric circle

We consider the case of a circle defined by a center point (x_c, y_c) and a radius R only to simplify the formulation. When we sketch a circle, we must define a sketch plane because a center and a radius define an infinite number of circles. Figure 8.5 shows this definition of a parametric circle. The parameter u is the angle, measured in counterclockwise direction. The circle equation is given by:

$$\boldsymbol{P}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} x_c + R\cos u \\ y_c + R\sin u \end{bmatrix} \quad 0 \le u \le 2\pi \tag{8.11}$$

We can normalize the u limits in Eq. (8.11) to (0, 1) instead of (0, 2π). This gives:

$$\boldsymbol{P}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} x_c + R\cos 2\pi u \\ y_c + R\sin 2\pi u \end{bmatrix} \quad 0 \le u \le 1 \tag{8.12}$$

Note that the circle, although closed, has two coincident endpoints, one for each u limit.

The tangent vector of the circle is given by:

$$\boldsymbol{P}'(u) = \begin{bmatrix} x'(u) \\ y'(u) \end{bmatrix} = \begin{bmatrix} -R\sin u \\ R\cos u \end{bmatrix} \quad 0 \le u \le 2\pi \tag{8.13}$$

Or,

$$\boldsymbol{P}'(u) = \begin{bmatrix} x'(u) \\ y'(u) \end{bmatrix} = \begin{bmatrix} -2\pi R\sin 2\pi u \\ 2\pi R\cos 2\pi u \end{bmatrix} \quad 0 \le u \le 1 \tag{8.14}$$

Equations (8.11) through (8.14) can also be used to define arcs. The only difference is the u limits, that is, $u_1 \le u \le u_2$. Figure 8.5 shows this arc segment.

Example 8.2 Find the equation of a circle with a radius of 2.0 in. and a center at the origin. Find the midpoint of the circle. What is the tangent vector and slope at this point?

Solution Using Eq. (8.11) with the given data, we get:

$$\boldsymbol{P}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} 2\cos u \\ 2\sin u \end{bmatrix} \qquad 0 \le u \le 2\pi \tag{8.15}$$

The midpoint of the circle occurs in the mid-range of the u parameter, that is, at $u = \pi$. Equation (8.15) gives:

$$\boldsymbol{P}(\pi) = \begin{bmatrix} x(\pi) \\ y(\pi) \end{bmatrix} = \begin{bmatrix} -2 \\ 0 \end{bmatrix} \tag{8.16}$$

These coordinates agree with the visual inspection of Figure 8.5; the point is the leftmost point on the circle circumference. Note that the midpoint of the circle is not its center; the center does not lie on the circle circumference. The tangent vector at the midpoint is obtained by substituting $u = \pi$ in Eq. (8.13) to get:

$$\boldsymbol{P}'(\pi) = \begin{bmatrix} x'(\pi) \\ y'(\pi) \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \end{bmatrix} \tag{8.17}$$

The slope at this point is given by $\dfrac{y'}{x'} = \infty$ as expected because the tangent vector is vertical at this point; that is, it makes a 90° angle measured from the horizontal axis counterclockwise.

HANDS-ON FOR EXAMPLE 8.2. Find the parametric equation of the quarter circle in the second quadrant. What is the midpoint of this arc? What is the tangent vector at this point?

8.5 Spline Parametric Equation

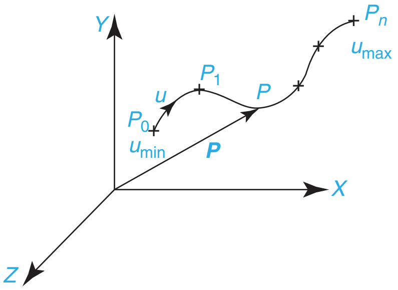


FIGURE 8.6
Parametric spline

Different types of splines exist. The most commonly used one by CAD/CAM systems is the cubic B-spline curve, or spline for short. Figure 8.6 shows a spline connecting $n + 1$ data points. The spline equation takes the following form:

$$\boldsymbol{P}(u) = f_0(u)\boldsymbol{P}_0 + f_1(u)\boldsymbol{P}_1 + \cdots + f_n(u)\boldsymbol{P}_n \qquad u_{\min} \le u \le u_{\max} \tag{8.18}$$

where the highest degree of any of these $f(u)$ functions is cubic.

Example 8.3 A spline is given by:

$$\boldsymbol{P}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} 2 + 3u^2 - 2u^3 \\ 2 + 3u - 3u^2 \end{bmatrix} \qquad 0 \le u \le 1 \tag{8.19}$$

Find the endpoints and the midpoint of the spline. Also, find the tangent vector and the slope at the midpoint. Using your CAD/CAM system, create the spline defined by the above equation.

Solution The endpoints occur at $u = 0$ and $u = 1$. The midpoint of the spline occurs at $u = 0.5$. Substituting these values in Eq. (8.19), we get:

$$\boldsymbol{P}(0) = \begin{bmatrix} 2 \\ 2 \end{bmatrix}, \quad \boldsymbol{P}(0.5) = \begin{bmatrix} 2.5 \\ 2.75 \end{bmatrix}, \text{ and } \boldsymbol{P}(1) = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

Differentiating Eq. (8.19) gives the tangent vector to the spline at any point on it as:

$$\boldsymbol{P}'(u) = \begin{bmatrix} 6u - 6u^2 \\ 3 - 6u \end{bmatrix} \quad 0 \le u \le 1$$

The tangent vector at the midpoint is $\boldsymbol{P}'(0.5) = \begin{bmatrix} 1.5 \\ 0 \end{bmatrix}$, and the slope is $\dfrac{y'}{x'} = \dfrac{0}{1.5} = 0$; that is, the tangent vector is horizontal at the midpoint.

The following steps show how to create the spline (Figure 8.7) using SolidWorks.

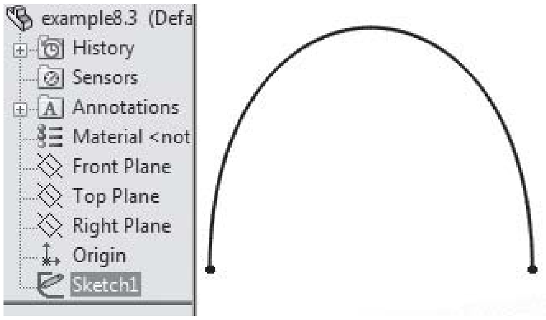
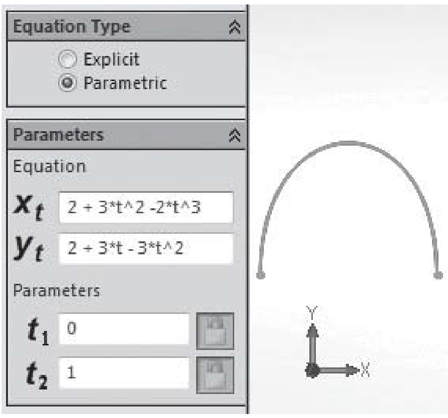


FIGURE 8.7
Spline

Step 1: Create *spline*: **File** > **New** > **Part** > **OK** > **Front Plane** > **Spline** dropdown on **Sketch** tab > **Equation Driven Curve** > **Parametric** > enter *x* and *y* equations and limits given by Equation (8.19) as shown > ✔ > **File** > **Save As** > *example8.3* > **Save**.



HANDS-ON FOR EXAMPLE 8.3. Find the coordinates of the points on the spline where $u = 0.25$ and $u = 0.75$.

8.6 Two-Dimensional Curves

SolidWorks implements the curve parametric theory presented here and provides versatile ways for designers to create curves. Curves may be planar (2D curves) or nonplanar (3D curves). The 2D curves are sketch entities; that is, they lie in the sketch plane regardless of how complex they may look. To create 2D curves from equations, we may use this sequence (or the sequence shown in Step 1 of Example 8.3): **Tools > Sketch Entities > Equation Driven Curve**. We must be in a sketch to access it.

8.7 Three-Dimensional Curves

3D curves, unlike 2D curves, do not belong to only one sketch plane. A segment of a 3D curve may belong to one sketch plane while another segment may belong to another sketch plane. 3D curves are valuable to use in some designs, and they simplify feature creation significantly as shown in the tutorials in this chapter. The 3D curves become more powerful and elegant when we combine them with surfaces, as shown later in the book.

There are multiple methods to create 3D curves (refer to the tutorials of the chapter on how to use each method):

1. Curve explicit equation: This method takes a curve explicit equation in the form given by Equation (8.1).
2. Curve parametric equation: This method takes a curve parametric equation in the form given by Equation (8.3).
3. 3D points: This method requires a list of (x, y, z) coordinates of the 3D points. The user may input the coordinates while creating the curve or store them in a text file and read it into SolidWorks.
4. 3D sketching: This method allows us to sketch entities in different sketch planes.
5. Composite curve: We can create composite curves by combining curves, sketch geometry, and model edges into a single curve. The individual curves may belong to one sketch or different sketches. If they belong to the same sketch, the resulting composite curve is 2D. If they belong to multiple sketches, the resulting curve is 3D.
6. Project a curve onto a model face: This method takes a 2D curve created on a sketch plane and converts it into a 3D curve by projecting it onto a model face. The model face needs to be nonplanar to create a 3D curve; otherwise, the projected curve remains 2D.
7. Projected curve: This is a very powerful method to create 3D curves. You can create a 3D curve from two 2D curves. These 2D curves are two projections of the 3D curve onto two intersecting sketch planes. The 3D curve represents the surface/surface intersection of two extruded surfaces generated by the two 2D curves, as shown in Figure 8.8. A common practice is to sketch the two best projections of the 3D curve on two different sketches (e.g., front and top or front and right) and then use the projected curve method to create the 3D curve. If the resulting curve needs tweaking, delete it, modify the two sketches, and then re-create it. We continue this iterative process until we are satisfied with the resulting 3D curve.

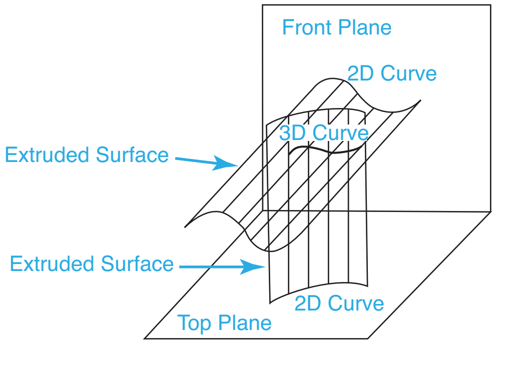


FIGURE 8.8
Creating a 3D projected curve

8.8 Curve Management

After we create curves, we can manage and manipulate them in different ways. We can modify, edit, trim, split (divide), and/or intersect them. These manipulations are easily done as an outcome of the curve parametric formulation presented in this chapter. We have already done all these manipulations except breaking a curve. Follow this sequence to split a curve: Right click it > **Split Entities** (from the pop-up window) > click the curve where you want to split it > **Esc** key on the keyboard to finish. To verify the split, hover over the entities and observe the curve segments. You may also verify the split by right clicking a segment > **Delete** from the popup menu.

8.9 Tutorials

The theme for the tutorials in this chapter is to practice creating and using curves. The tutorials show how to use all the methods for creating 3D curves that we have covered.

Tutorial 8–1: Create a 2D Curve Using Explicit Equation

We create a curve defined by the following explicit equation:

$$y = 0.5x^2 \quad 0 \leq x \leq 3 \tag{8.20}$$

Equation (8.20) defines half a parabola that is symmetric with respect to the Y axis. We revolve the parabola to create a feature (Figure 8.9) for better visualization.

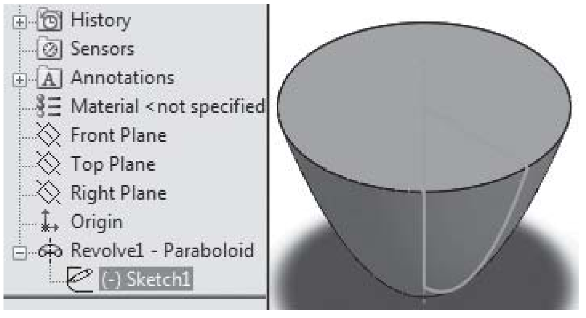
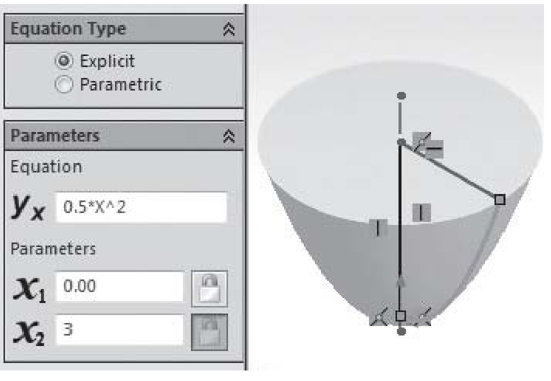


FIGURE 8.9
A revolve

Step 1: Create *Sketch1* and *Revolve1-Paraboloid*: **File** > **New** > **Part** > **OK** > **Front Plane** > **Spline** dropdown on **Sketch** tab > **Equation Driven Curve** > **Explicit** > enter equation and limits given by Eq. (8.20) as shown > ✔ > **Line** on **Sketch** tab > sketch horizontal and vertical lines shown to close cross section > **Centerline** on **Sketch** tab > sketch vertical line shown passing through origin > exit sketch > **Revolved Boss/Base** on **Features** tab > ✔ > **File** > **Save As** > *tutorial8.1* > **Save**.



HANDS-ON FOR TUTORIAL 8–1. Find the coordinates of the end points of the half parabola using SolidWorks. Verify the results by using Equation (8.20).

Hint: Use the **Measure** tool on the **Evaluate** tab.

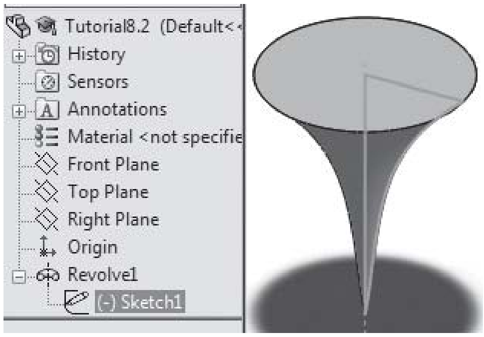
Tutorial 8–2: Create a 2D Curve Using Parametric Equation

We create a curve defined by the following parametric equation:

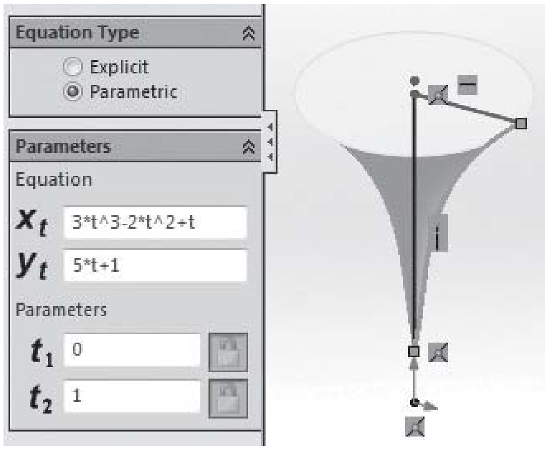
$$P(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} 3u^3 - 2u^2 + u \\ 5u + 1 \end{bmatrix} \qquad 0 \leq u \leq 1 \tag{8.21}$$

We revolve the resulting curve to create a feature (Figure 8.10) for better visualization.

FIGURE 8.10
A revolve



Step 1: Create *Sketch1* and *Revolve1*: **File** > **New** > **Part** > **OK** > **Front Plane** > **Spline** dropdown on **Sketch** tab > **Equation Driven Curve** > **Parametric** > enter equation and limits given by Eq. (8.21) as shown > ✔ > **Line** on **Sketch** tab > sketch horizontal and vertical lines shown to close cross section > **Centerline** on **Sketch** tab > sketch vertical line shown passing through origin > exit sketch > **Revolved Boss/Base** on **Features** tab > ✔ > **File** > **Save As** > *tutorial8.2* > **Save**.



> ***HANDS-ON FOR TUTORIAL 8–2.*** Find the coordinates of the end points of the parametric curve using SolidWorks. Verify the results by using Eq. (8.21).
>
> **Hint:** Use the **Measure** tool on the **Evaluate** tab.

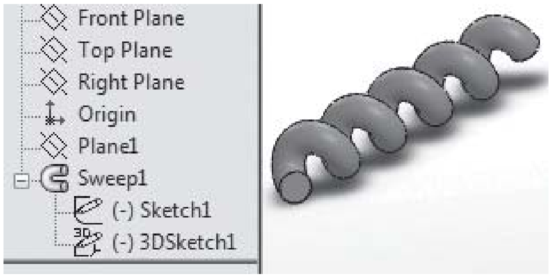
Tutorial 8–3: Create a 3D Curve Using Parametric Equation

We create a helix defined by the following parametric equation:

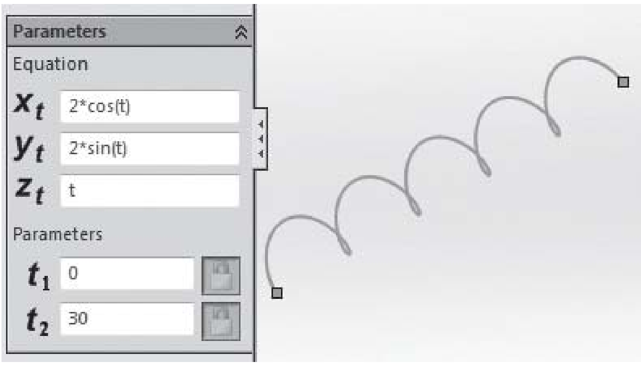
$$P(u) = \begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix} = \begin{bmatrix} 2\cos u \\ 2\sin \\ u \end{bmatrix} \qquad 0 \leq u \leq 30 \tag{8.22}$$

We sweep a circle along the resulting helix to create a feature (Figure 8.11) for better visualization.

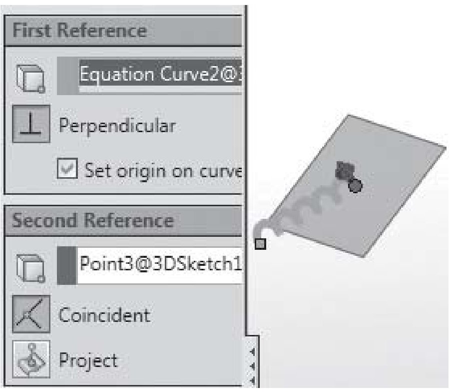
FIGURE 8.11
A sweep



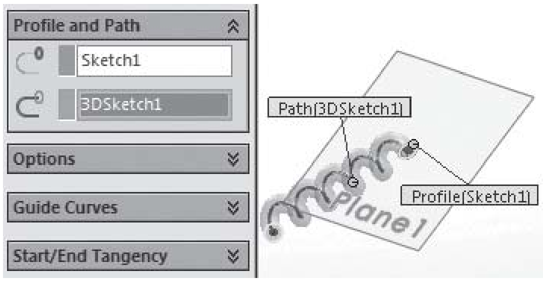
Step 1: Create 3D curve: **File** > **New** > **Part** > **OK** > **Sketch** dropdown on **Sketch** tab > **3D Sketch** > **Equation Driven Curve** > enter equation and limits given by Eq. (8.22) as shown > ✔.



Step 2: Create a plane perpendicular to curve endpoint: **Reference Geometry** dropdown on **Features** tab > **Plane** > helix endpoint shown > ✔.



Step 3: Create sweep: *Plane1* created in Step 2 as sketch plane > **Circle** on **Sketch** tab > click helix endpoint and drag to sketch a circle with diameter of 3.0 inches > exit sketch > **Swept Boss/Base** > circle sketch as profile and helix sketch as path as shown > ✔.



Note: We use 3D sketch in Step 1 to have access to the z coordinate of the parametric equation as shown. If we use a 2D sketch as in Tutorial 8.2, the z coordinate does not show.

HANDS-ON FOR TUTORIAL 8–3. Replace 2 by 4 and 30 by 60 in Step 1. What happens to the helix?

Tutorial 8–4: Create a 3D Curve Using 3D Points

We create a set of 3D points in 3D modeling space and connect them with a spline. We use the resulting curve to create a sweep (Figure 8.12) for better visualization. All dimensions are in mm.

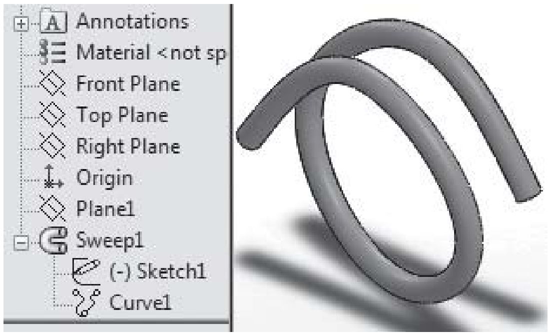
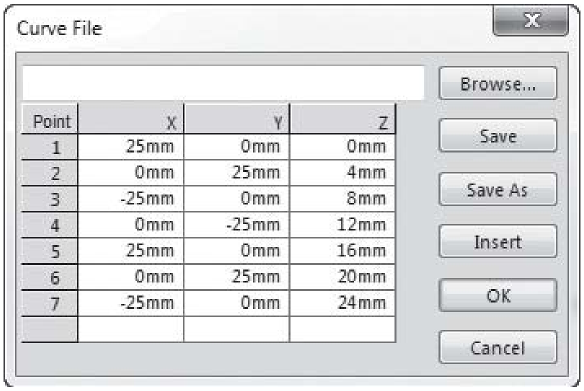


FIGURE 8.12
A sweep

Step 1: Create 3D curve, *Curve1*, by entering 3D points: **File > New > Part > Insert > Curve > Curve Through XYZ Points >** double click any cell to enter value > enter all values shown > **OK.**

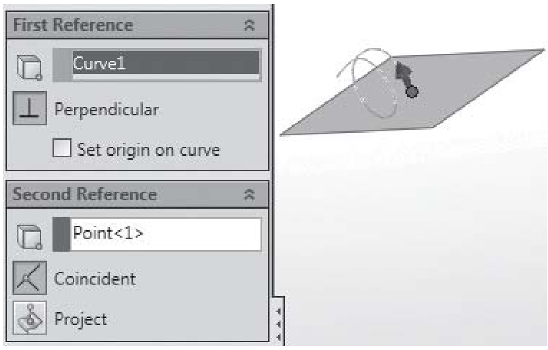


Point	X	Y	Z
1	25mm	0mm	0mm
2	0mm	25mm	4mm
3	-25mm	0mm	8mm
4	0mm	-25mm	12mm
5	25mm	0mm	16mm
6	0mm	25mm	20mm
7	-25mm	0mm	24mm

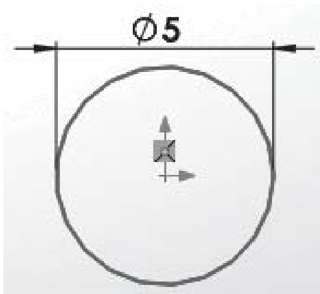
Note: You can save the points' coordinates in a text file and read them in instead of typing as follows: **Browse** button shown > locate the file > **Open > OK.** The file format is as follows: one line per point. Point coordinates (x, y, z) are separated by spaces, e.g.,

25 0 16
0 25 4
–25 0 8

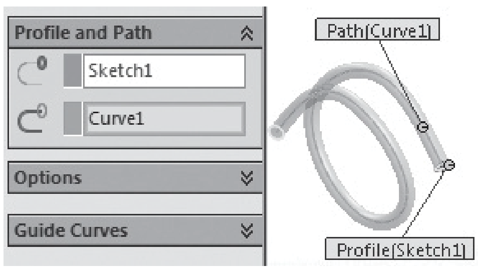
Step 2: Create a plane perpendicular to curve end-point: **Reference Geometry** dropdown on **Features** tab > **Plane >** select curve as **First Reference** > select curve endpoint shown as **Second Reference** > ✔.



Step 3: Create *Sketch1*: Select plane created in Step 2 as a sketch plane > **Circle** on **Sketch** tab > click curve end and drag and sketch and dimension circle as shown > exit sketch.



Step 4: Create sweep for better visualization: **Swept Boss/Base** on **Features** tab > *Sketch1* as profile > *Curve1* as path > ✔ > **File > Save As >** *tutorial8.3* > **Save**.



HANDS-ON FOR TUTORIAL 8–3. Save the points in a text file, add five more points to the file, and re-create the curve by reading the file into SolidWorks.

Tutorial 8–5: Create a 3D Curve Using 3D Sketching

We build a bicycle handle bar using a 3D sketch. Enable sketching in 3D space by toggling (changing) sketch planes. We toggle using the **Tab** key on keyboard. The available sketch planes are XY (front), XZ (top), and YZ (right). The current plane designation is attached to the mouse as it moves. The bicycle handle bar requires the XY plane and the YZ plane. Figure 8.13 shows the handlebar. All dimensions are in inches.

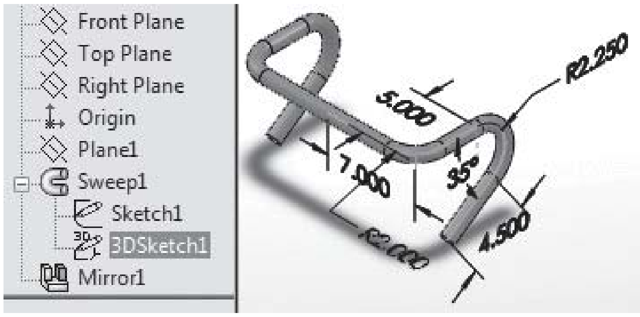
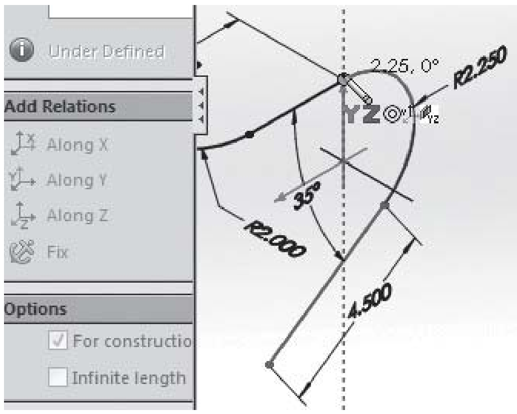
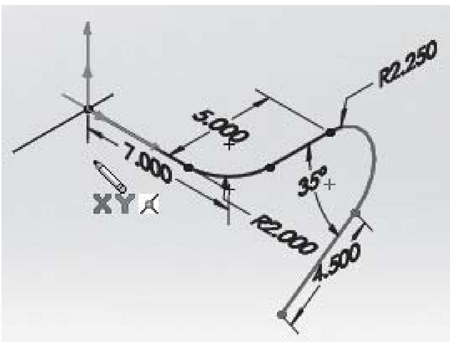


FIGURE 8.13
Bicycle handlebar

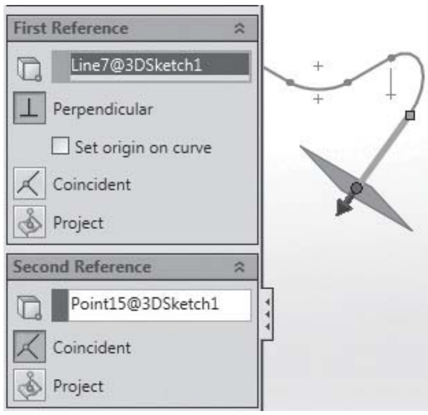
Step 1: Create *3DSketch1*: **File > New > Part > OK > Sketch** dropdown on **Sketch** tab > **3D Sketch > Line** on **Sketch** tab > click origin and drag along the X axis to sketch a line in the **XY** plane as shown **> Tab** key to switch to **YZ** plane > **Line** on **Sketch** tab > Sketch line > **Arc** dropdown on **Sketch** tab > **Tangent Arc** > sketch (in **YZ** plane) an arc passing through endpoint of line as shown **> Line** on **Sketch** tab > sketch (in **YZ** plane) a line passing through endpoint of arc as shown > **Esc** > line **+ Ctrl** key + arc > **Tangent** from **Add Relations** box > ✔ > fillet two lines as shown > **Smart Dimensions** on **Sketch** tab > dimension all as shown.



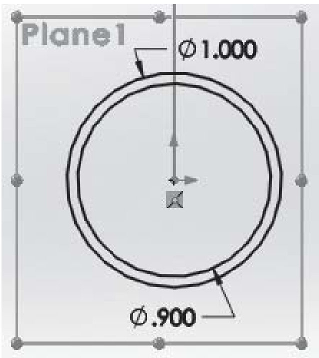


Step 2: Fully define *3DSketch1*: **Centerline** on **Sketch** tab > sketch line connecting the arc center and line endpoint as shown> right click > select **Make Along Y** relation from menu shown > exit sketch.

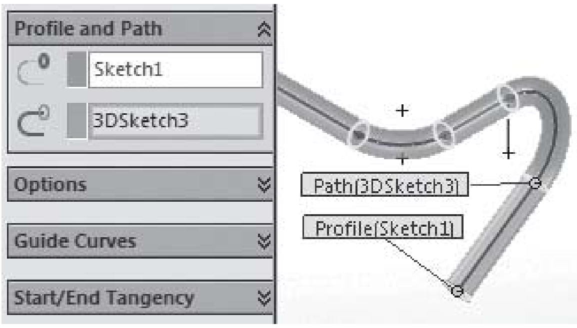
Step 3: Create a *Plane1*: **Reference Geometry** dropdown on **Features** tab > **Plane** > **Perpendicular** > line segment shown > line endpoint shown > ✔.



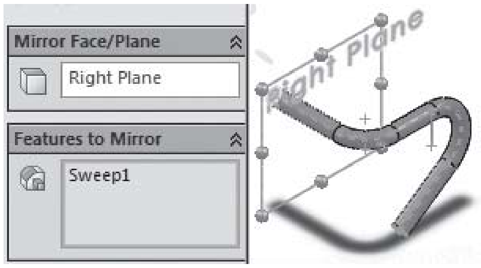
Step 4: Create *Sketch1*: Select *Plane1* created in Step 3 as a sketch plane > **Circle** on **Sketch** tab > click line end and drag and sketch and dimension circles as shown > exit sketch.



Step 5: Create *Sweep1* for better visualization of curve: **Swept Boss/Base** on **Features** tab > *Sketch1* as profile > *3DSketch3* as path > ✔.



Step 6: Create the other half of bike handlebar: **Mirror** on **Features** tab > expand features tree > **Right Plane** as **Mirror Face/Plane** as shown > *Sweep1* as **Features to Mirror** as shown > ✔.

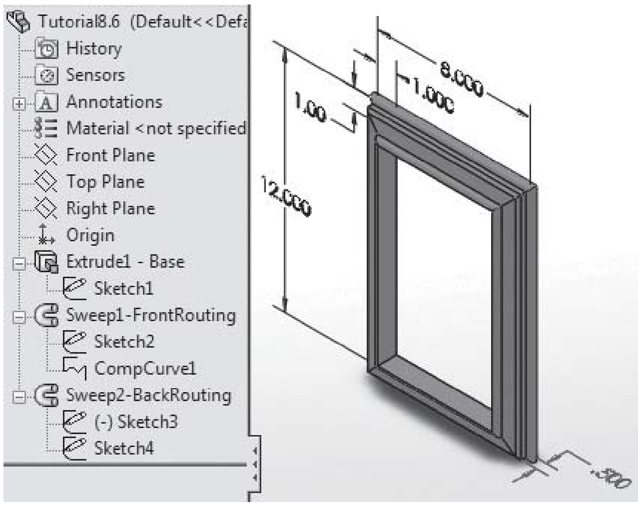


HANDS-ON FOR TUTORIAL 8–5. Redo the tutorial but without using the mirror feature of Step 6. Instead, use 3D sketch to create the entire skeleton of the handlebar. Which approach is easier and faster? Why?

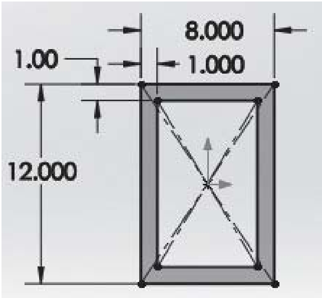
Tutorial 8–6: Create a 3D Curve Using Composite Curves

We create a picture frame using a composite curve consisting of all the outside edges of the frame. We use the composite curve to create a sweep to model the decorative routing of the frame as shown in Figure 8.14. All dimensions are in inches.

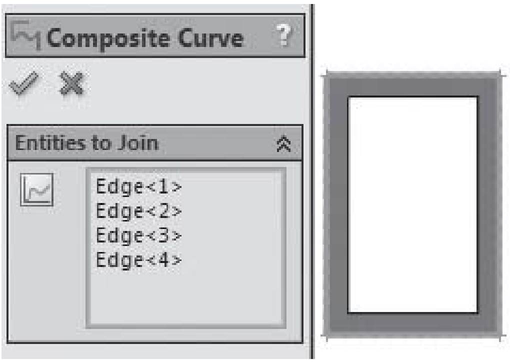
FIGURE 8.14
Picture frame with decorative routing



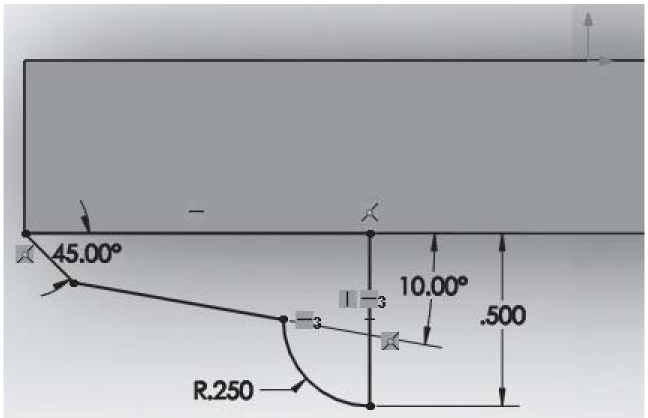
Step 1: Create *Sketch1* and *Extrude1-Base* feature: **File > New > Part > Front Plane > Extruded Boss/Base** on **Features** tab **> Center Rectangle** on **Sketch** tab > sketch and dimension two center rectangles shown > exit sketch > enter **0.5** for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As** > *tutorial8.6* > **Save.**



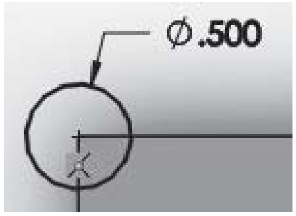
Step 2: Create *CompCurve1*: **Insert** menu **> Curve > Composite** > select the four edges of the outer rectangle created in Step 1 in counterclockwise direction **>** ✔.



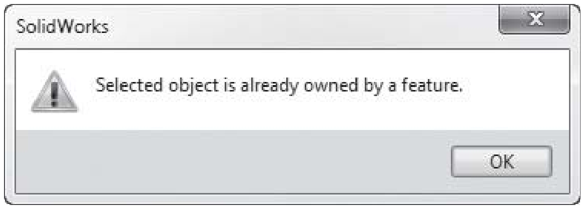
Step 3: Create *Sketch2* and *Sweep1-FrontRouting* feature: **Top Plane** > **Line** on **Sketch** tab > sketch and dimension lines shown > **Arc** dropdown on **Sketch** tab > **3 Point Arc** from dropdown > sketch and dimension arc shown > exit sketch > **Swept Boss/Base** on **Features** tab > *Sketch2* for profile and *CompCurve1* for path > ✔.



Step 4: Create *Sketch3*: **Top Plane** > **Circle** on **Sketch** tab > sketch and dimension circle shown (make sure you snap to intersection point as shown) > exit sketch.



Step 5: Create *Sketch4* and *Sweep2-BackRouting* feature: > **Front Plane** as sketch plane > **Convert Entities** on **Sketch** tab > expand features tree > *CompCurve1* > ✔ > **Swept Boss/Base** on **Features** tab > *Sketch3* for profile and *Sketch4* for path > ✔.



Note: *Sketch4* is a copy of *Sketch3* using the **Convert Entities** concept because SolidWorks would not allow sharing *CompCurve1* (composite curve) between two features. It gives the error message shown.

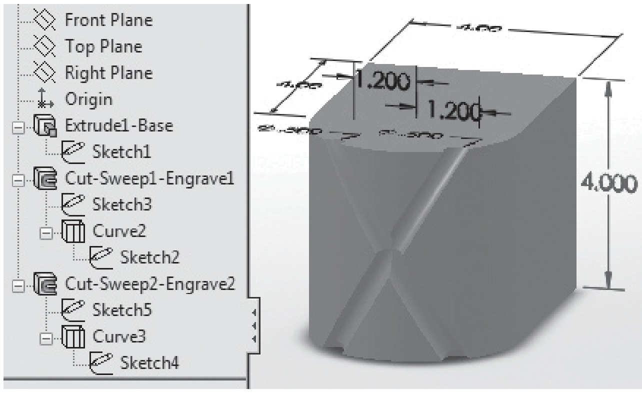
Step 6: Change material to teak wood: Right-click **Material** node > **Edit Material > Woods > Teak > Apply > Close**.

HANDS-ON FOR TUTORIAL 8–6. Redo Tutorial 8–5 using the composite curve method instead of the 3D sketch method. Which method is easier and faster?

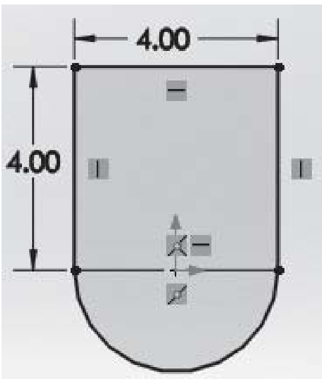
Tutorial 8–7: Create a 3D Curve by Projecting a Sketch onto a Curved Face

We use a curve projected onto a cylindrical face to engrave a part as shown in Figure 8.15. All dimensions are in inches.

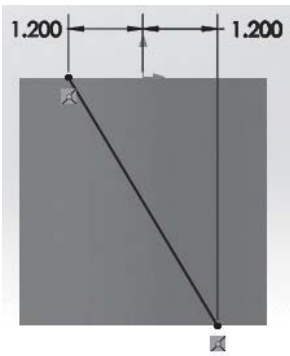
FIGURE 8.15
Engraving the letter X on a curved face



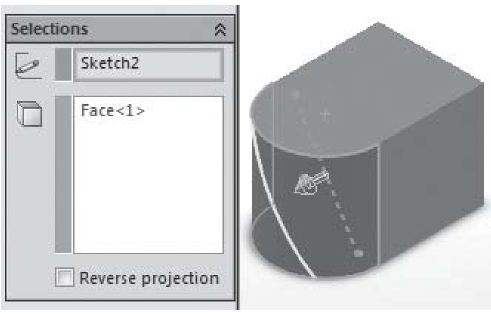
Step 1: Create *Sketch1* and *Extrude1-Base* feature: **File** > **New** > **Part** > **Top Plane** > **Extruded Boss/Base** on **Features** tab > **Line** on **Sketch** tab > sketch and dimension lines shown > **Arc** dropdown on **Sketch** tab > **3 Point Arc** > sketch and dimension arc shown > exit sketch > enter **4.0** for thickness **D1** > reverse extrusion direction > ✔ > **File** > **Save As** > *tutorial8.7* > **Save**.



Sketch1

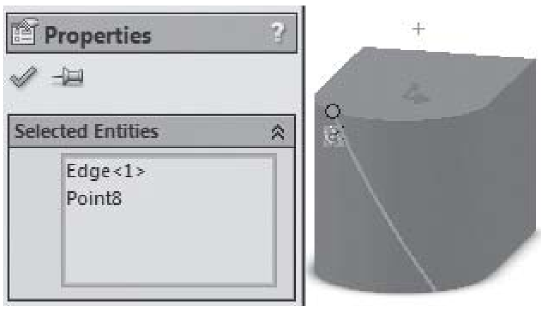


Sketch2

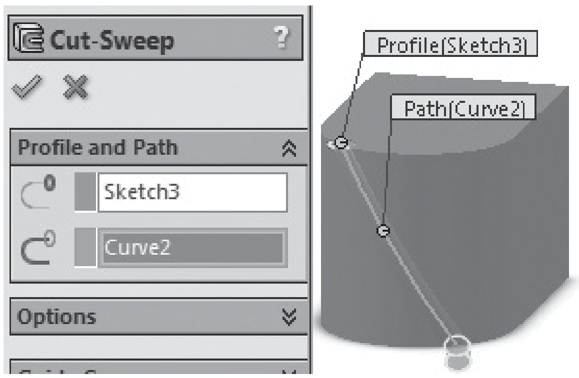


Step 2: Create *Sketch2* and *Curve2*: **Front Plane** > **Line** on the **Sketch** tab > sketch and dimension line shown above > exit sketch > **Insert Curve** > **Projected** > expand features tree > *Sketch2* > curved face > ✔.

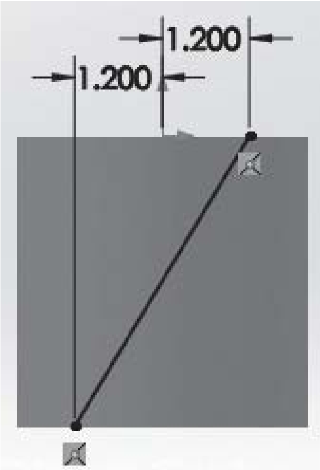
Step 3: Create *Sketch3*: **Top Plane** > **Point** on **Sketch** tab > click somewhere near top face of *Extrude1-Base* feature > **Esc** key > select point just created + **Ctrl** key + *Curve2* > **Pierce** from **Add Relations** section > ✔ > **Circle** on **Sketch** tab > click point and drag to sketch and dimension circle with a 0.5 inch diameter > exit sketch.



Step 4: Create *Cut-Sweep1-Engraved1* feature: **Swept Cut** on **Features** tab > *Sketch3* for profile and *Curve2* for path > ✔.

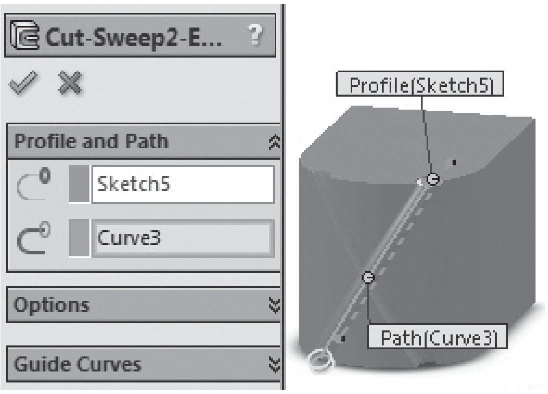


Step 5: Create *Sketch4* and *Curve3*: Repeat Step 2 to create the other part of the X letter. Shown here are *Sketch4* dimensions.



Step 6: Create *Sketch5*: Repeat Step 3 to create the circle. Use same dimension.

Step 7: Create *Cut-Sweep2-Engraved2* feature: **Swept Cut** on **Features** tab > *Sketch5* for profile and *Curve3* for path > ✔.



> ***HANDS-ON FOR TUTORIAL 8–7.*** Create a swept cut of the first letter of your last name using the same steps in the tutorial.

Tutorial 8–8: Create a 3D Curve Using Projected Curves

We create a 3D curve from projected curves (2D sketches). Projecting two 2D curves from multiple sketch planes creates a 3D curve. This tutorial shows examples of creating various solids using different projected sketches and a sweep profile. Case 1 (Figure 8.16) is shown in detail. The remaining cases are illustrated but are not shown in detail; the same steps can be followed as in case 1 in order to accomplish the goal. We create 3D curves using the following 2D curves:

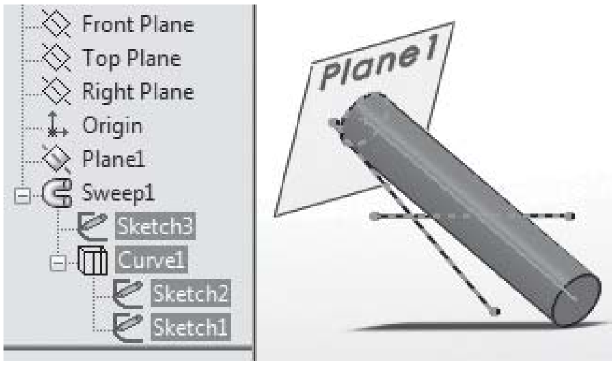
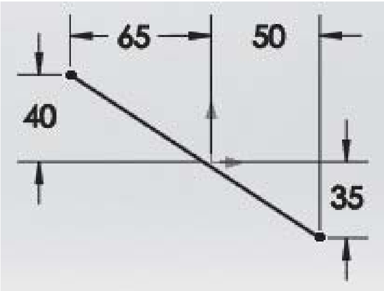


FIGURE 8.16
Case 1: Use two lines

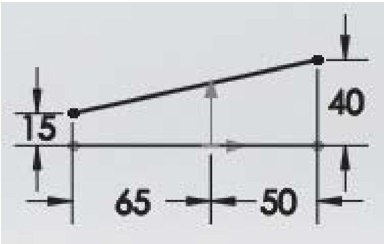
Case 1: Use two lines
Case 2: Use two circles
Case 3: Use two arcs
Case 4: Use a line and a circle
Case 5: Use a line and an arc
Case 6: Use two ellipses
Case 7: Use an ellipse and a line
Case 8: Use an ellipse and a circle

Case 9: Use an ellipse and an arc
Case 10: Use two splines
Case 11: Use a spline and a line
Case 12: Use a spline and a circle
Case 13: Use a spline and an ellipse

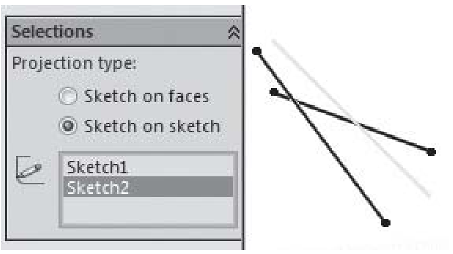
Step 1: Sketch *Sketch1*: **File > New > Part > Tools > Options > Document Properties** tab **> Units > MMGS > OK > Front Plane > Line** on **Sketch** tab > sketch and dimension line shown > exit sketch > **File > Save As** > *tutorial8.8* > **Save**.



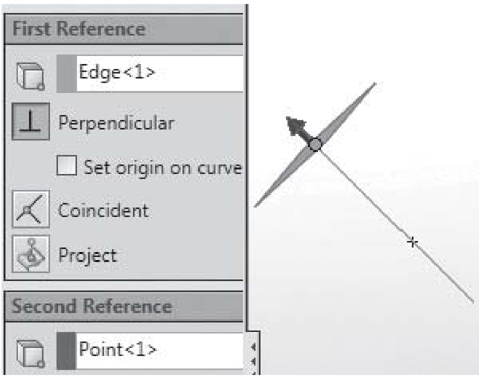
Step 2: Create *Sketch2*: Repeat Step 1 to create sketch shown. Use **Top Plane** to create sketch.



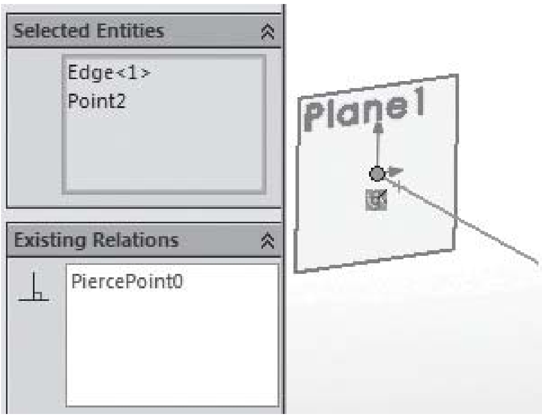
Step 3: Create projected curve: **Insert > Curve > Projected** > expand features tree > *Sketch1* > *Sketch 2* > ✔.



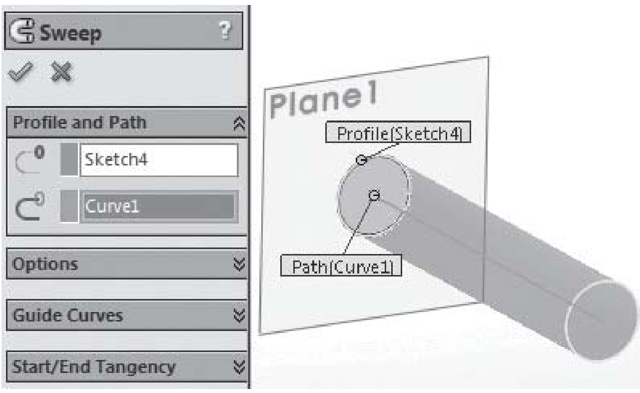
Step 4: Create *Plane1*: **Reference Geometry** on **Features** tab > **Plane** > *Curve1* > **Perpendicular** > endpoint shown > ✔.

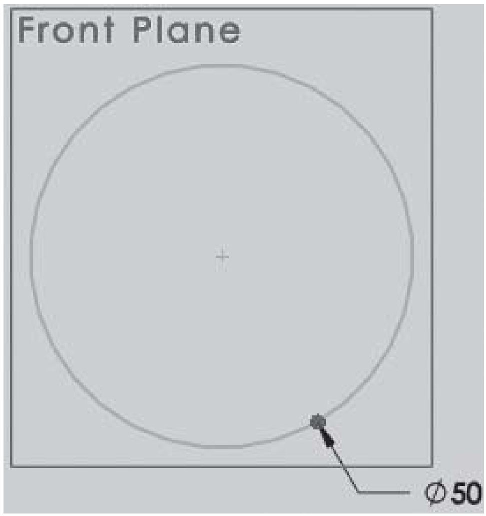


Step 5: Create *Sketch3*: *Plane1* as sketch plane > **Point** on **Sketch** tab > sketch point shown > **Esc** key > point just created + **Ctrl** key + *Curve1* > **Pierce** from **Add Relations** section > ✔ > **Circle** on **Sketch** tab > click point and drag to sketch and dimension circle with 25 mm diameter > exit sketch.

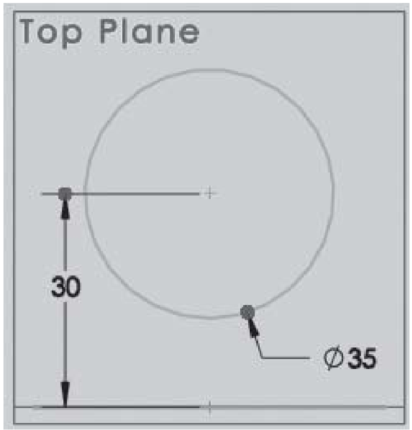


Step 6: Create *Sweep1*: **Swept Boss/Base** on **Features** tab > *Sketch3* for profile and *Curve1* for path > ✔.

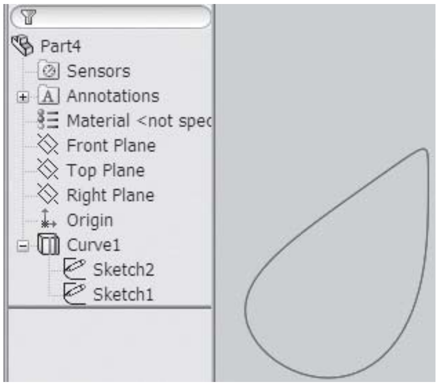




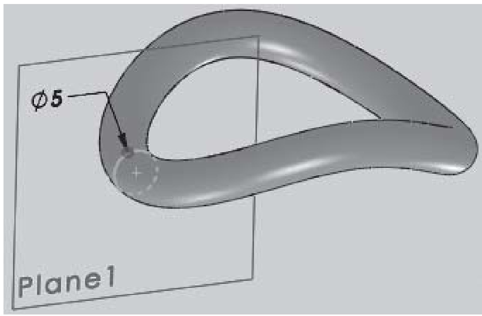
(A) Sketch 1



(B) Sketch 2

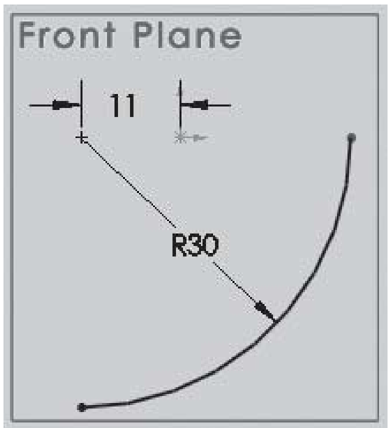


(C) Projected curve

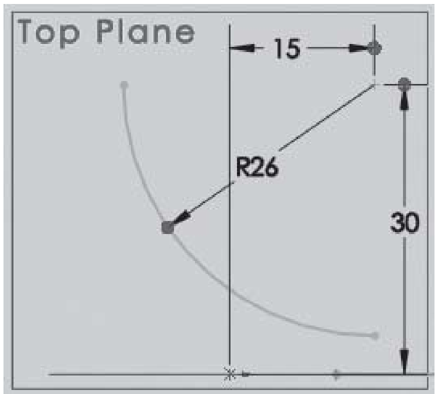


(D) Feature
Note: To create a plane on a closed curve, insert a reference point on curve first.

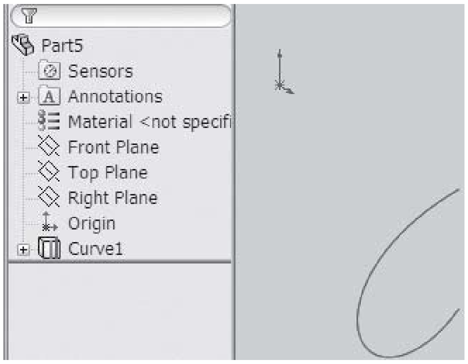
Case 2: Use two circles.



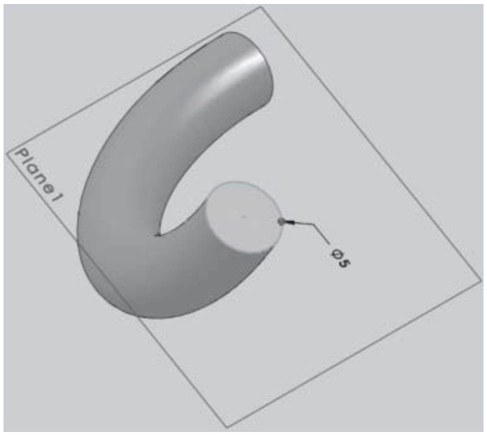
(A) Sketch 1



(B) Sketch 2

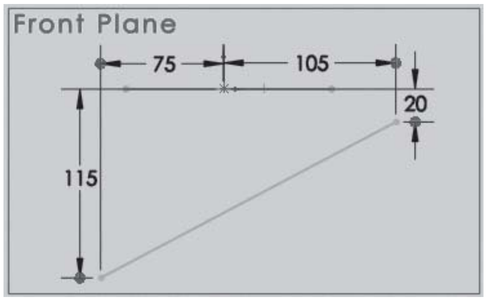


(C) Projected curve

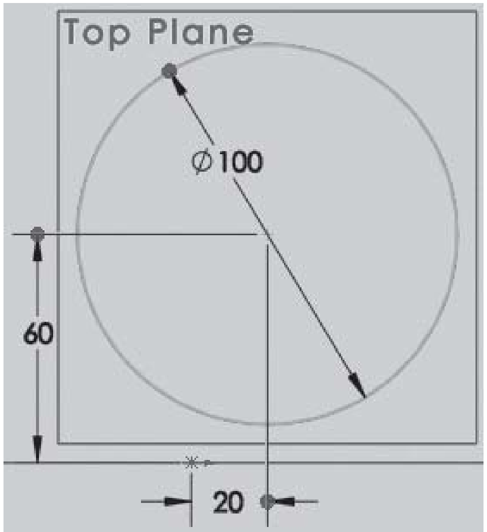


(D) Feature

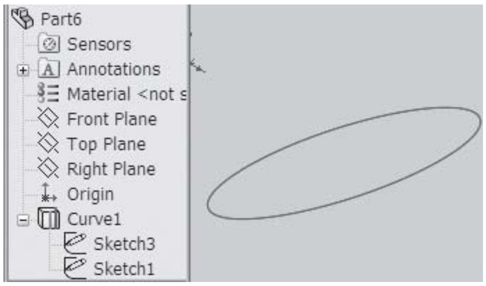
Case 3: Use two arcs.



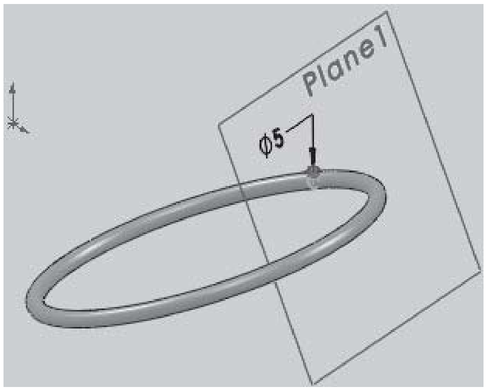
(A) Sketch 1



(B) Sketch 2

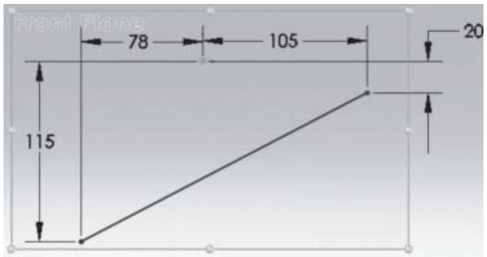


(C) Projected curve

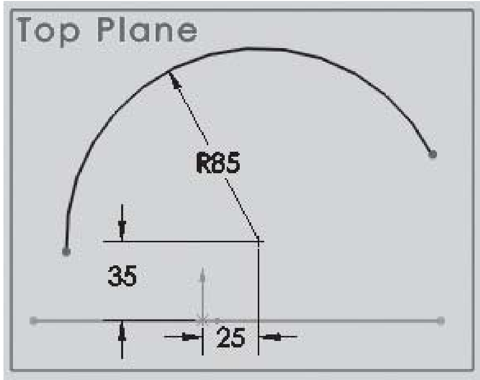


(D) Feature

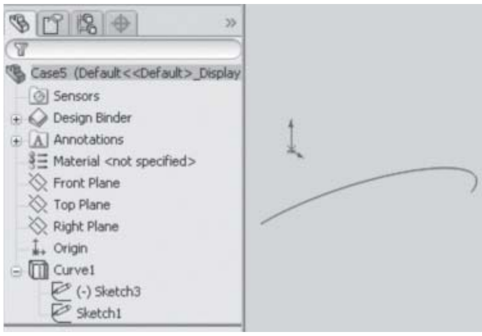
Case 4: Use a line and a circle.



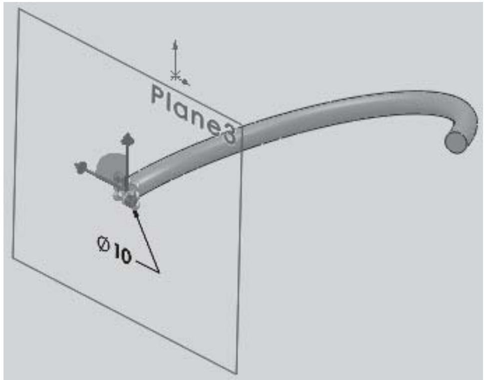
(A) Sketch 1



(B) Sketch 2

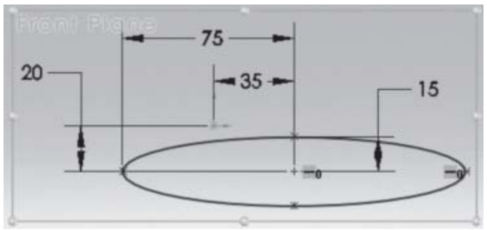


(C) Projected curve

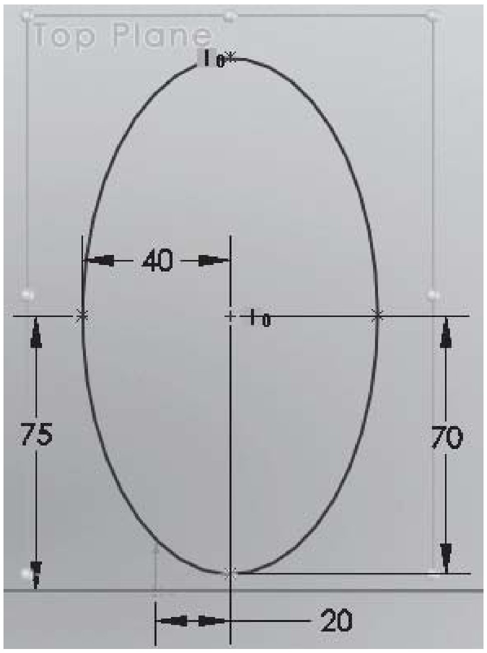


(D) Feature

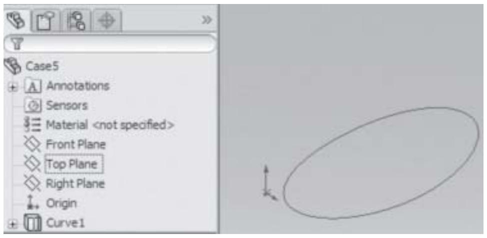
Case 5: Use a line and an arc.



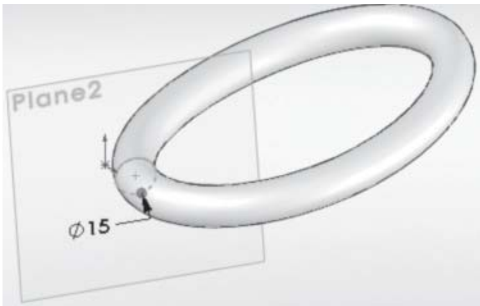
(A) Sketch 1



(B) Sketch 2

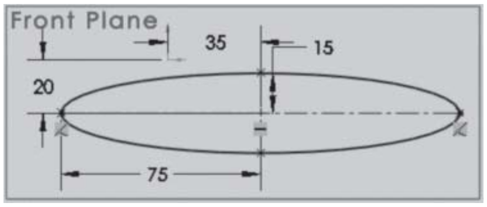


(C) Projected curve

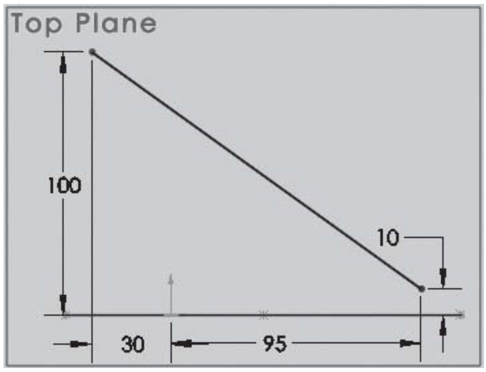


(D) Feature

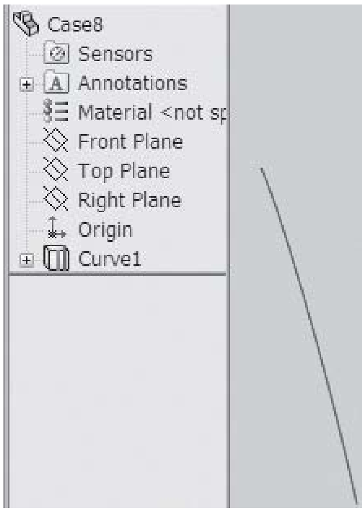
Case 6: Use two ellipses.



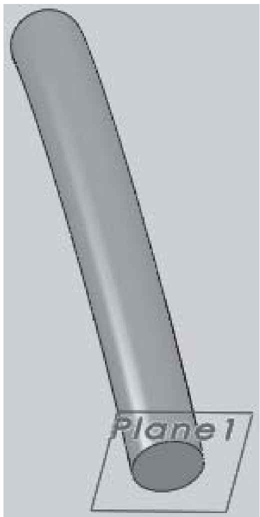
(A) Sketch 1



(B) Sketch 2

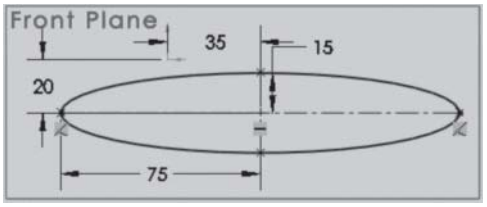


(C) Projected curve

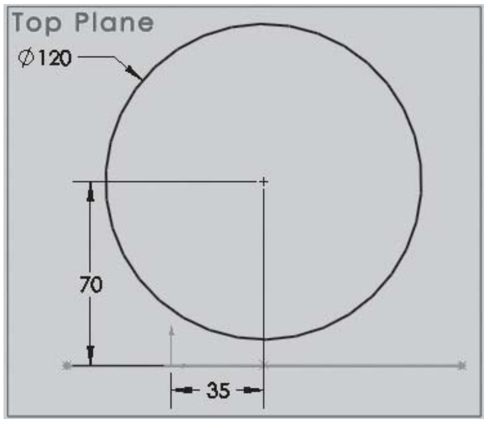


(D) Feature

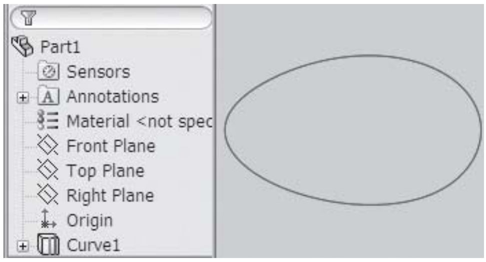
Case 7: Use an ellipse and a line.



(A) Sketch 1



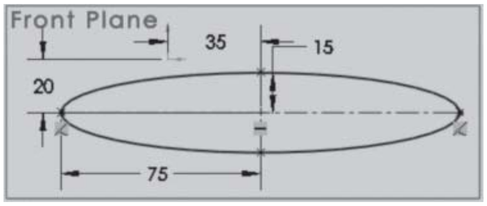
(B) Sketch 2



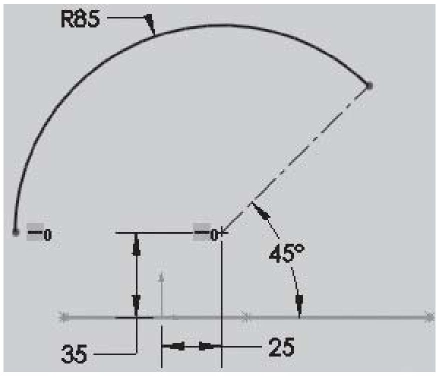
(C) Projected curve

(D) Feature

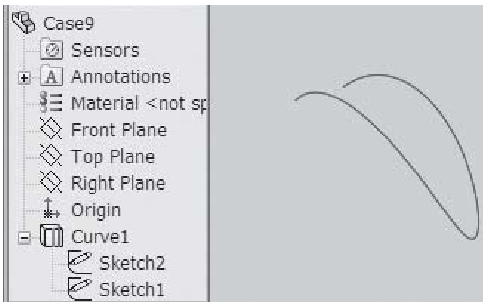
Case 8: Use an ellipse and a circle.



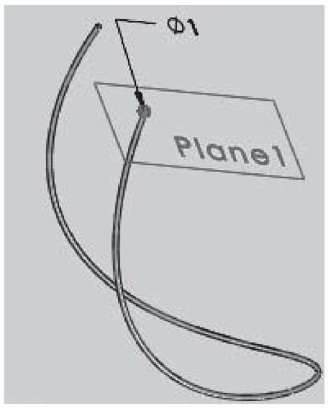
(A) Sketch 1



(B) Sketch 2

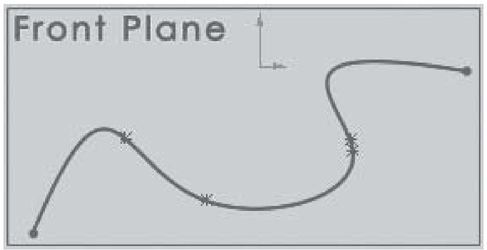


(C) Projected curve

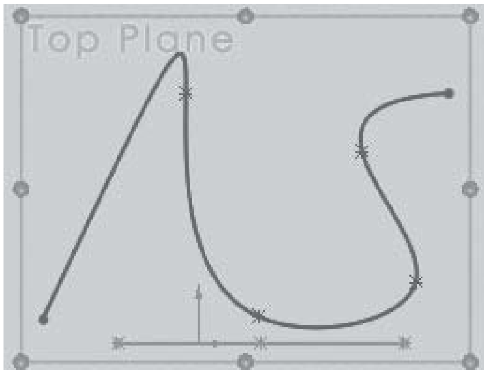


(D) Feature

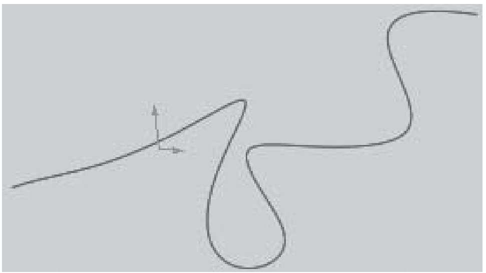
Case 9: Use an ellipse and an arc.



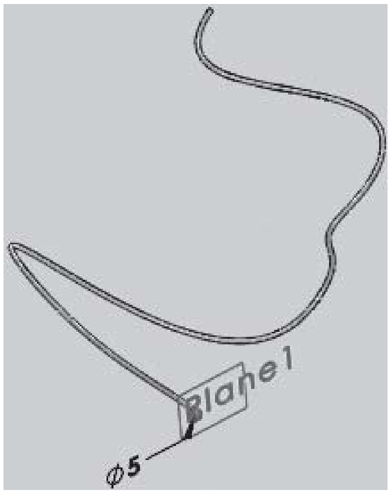
(A) Sketch 1



(B) Sketch 2

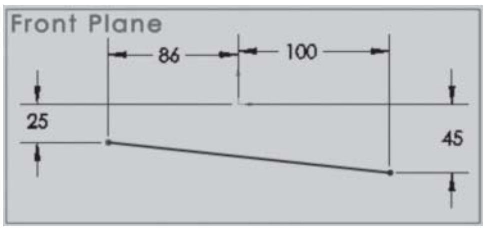


(C) Projected curve

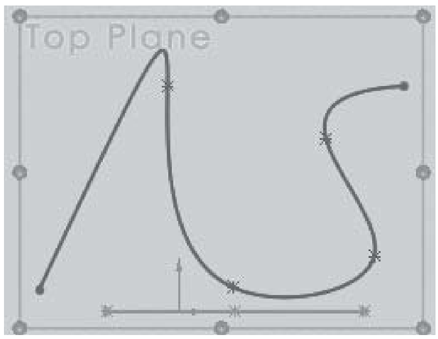


(D) Feature

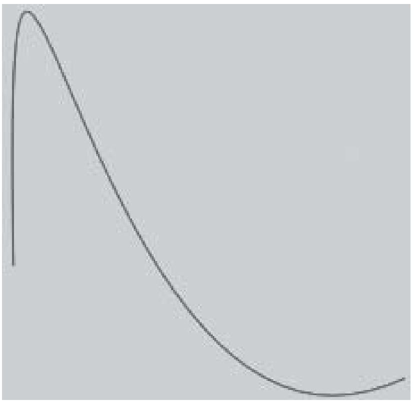
Case 10: Use two splines.



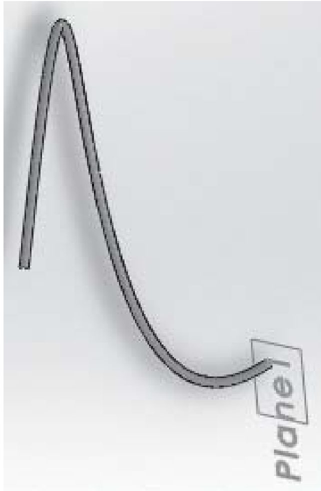
(A) Sketch 1



(B) Sketch 2

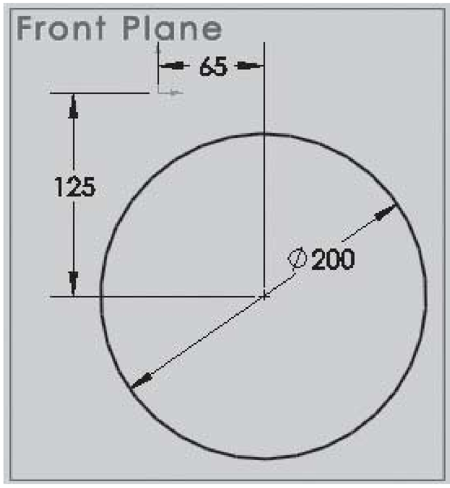


(C) Projected curve

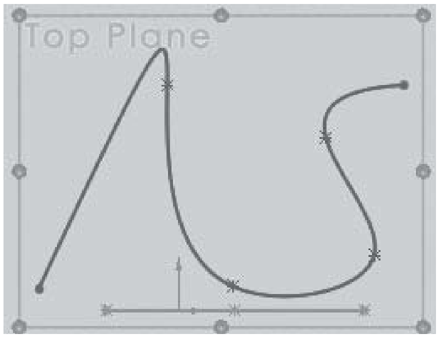


(D) Feature

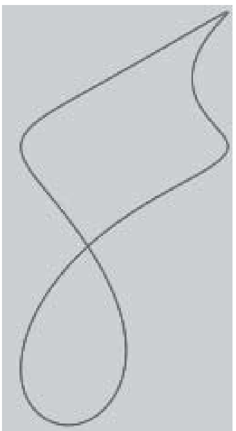
Case 11: Use a spline and a line.



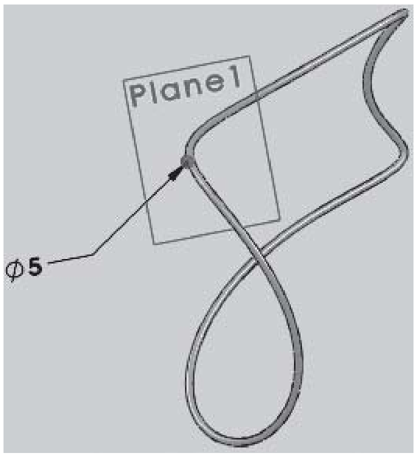
(A) Sketch 1



(B) Sketch 2

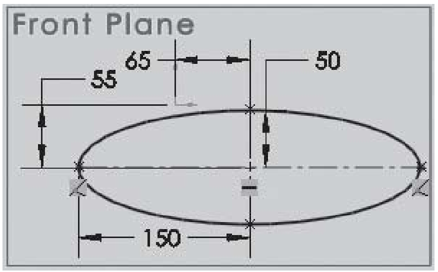


(C) Projected curve

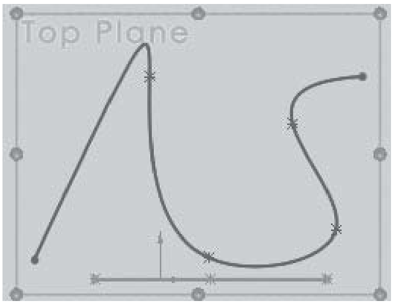


(D) Feature

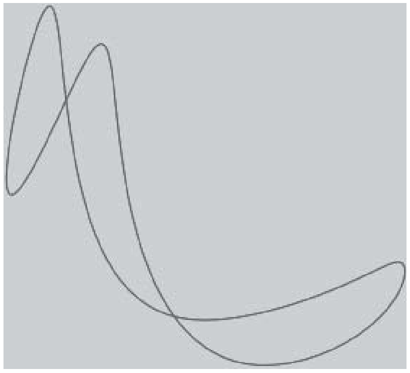
Case 12: Use a spline and a circle.



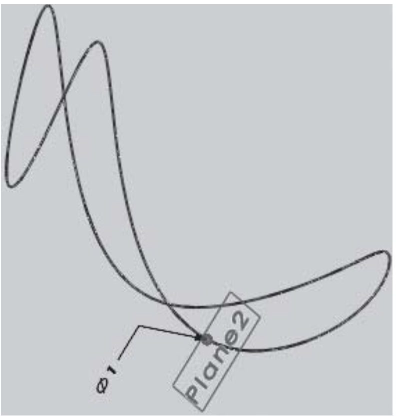
(A) Sketch 1



(B) Sketch 2



(C) Projected curve



(D) Feature

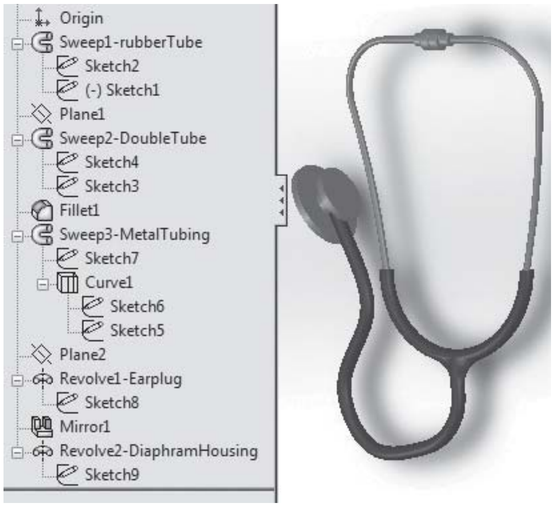
Case 13: Use a spline and an ellipse.

HANDS-ON FOR TUTORIAL 8–8. Create a 3D curve using an ellipse and a parabola. Sweep a circle along the 3D curve for better visualization.

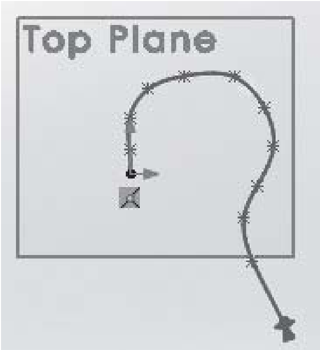
Tutorial 8–9: Create a Stethoscope Model

We use 2D and 3D (projected) curves to create the stethoscope model shown in Figure 8.17. All dimensions are in inches.

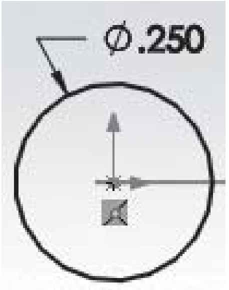
FIGURE 8.17
A stethoscope



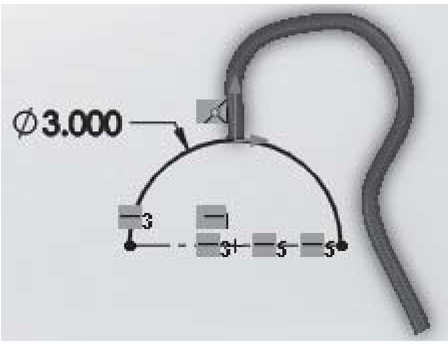
Step 1: Create *Sketch1*: **File > New > Part > Top Plane > Spline** on **Sketch** tab > starting at origin, sketch spline shown > exit sketch > **File > Save As >** *tutorial8.9* > **Save**.



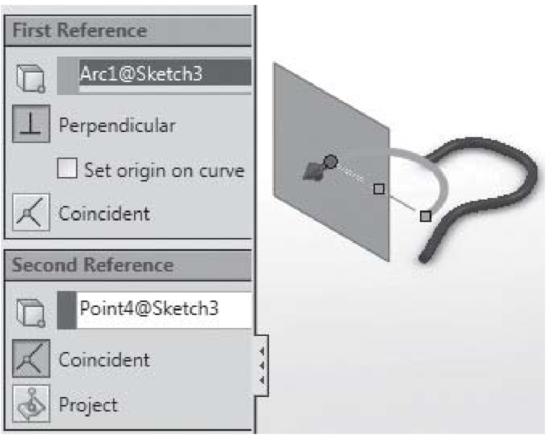
Step 2: Create *Sketch2* and *Sweep1-RubberTube*: **Front Plane > Circle** on **Sketch** tab > sketch and dimension circle shown > exit sketch **> Swept Boss/Base** on **Features** tab **>** *Sketch2* as profile > *Sketch1* as path **>** ✔.



Step 3: Create *Sketch3*: **Top Plane > Arc dropdown** on **Sketch** tab > **3 Point Arc** > sketch and dimension half circle shown > exit sketch.

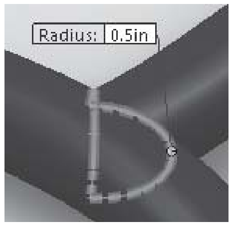


Step 4: Create *Plane1*: **Reference Geometry** drop-down on the **Features** tab **> Plane** > arc created in Step 3 > **Perpendicular** > arch endpoint shown > ✔.

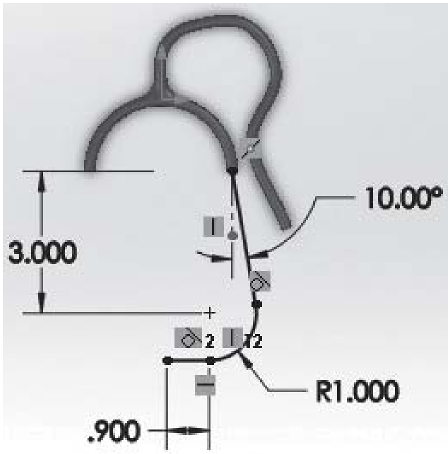


Step 5: Create *Sketch4* and *Sweep2-DoubleTube: Plane1* as a sketch plane > **Circle** on **Sketch** tab > click endpoint of arc shown **> Esc** key > circle just created + **Ctrl** key + circle of **Sketch2** > **Equal** from **Add Relations** section > ✔ > **Swept Boss/Base** on **Features** tab > *Sketch4* as profile > *Sketch3* as path > ✔.

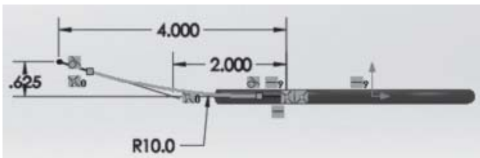
Step 6: Fillet the tubing: **Fillet** on **Features** tab > hover over intersection area and select as shown > use 0.5 in. for fillet radius > ✔.



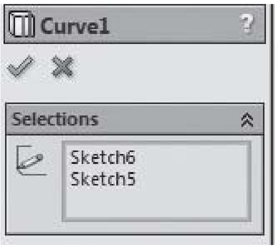
Step 7: Create *Sketch5*: **Top Plane** > **Line** on **Sketch** tab > sketch lines shown > **Arc** dropdown on **Sketch** tab > **3 Point Arc** > sketch arc shown > dimension as shown > exit sketch.



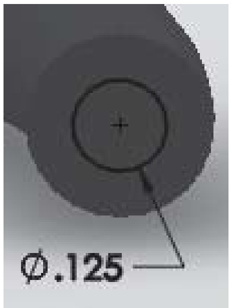
Step 8: Create *Sketch6*: **Right Plane** > **Line** on **Sketch** tab > sketch and dimension two lines shown > **Arc** dropdown on **Sketch** tab > **3 Point Arc** > sketch and dimension arc shown > exit sketch.



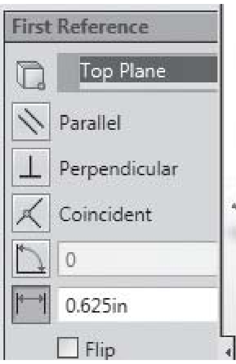
Step 9: Create *Curve1*: **Insert** > **Curve** > **Projected** > *Sketch5* > *Sketch6* > ✔.

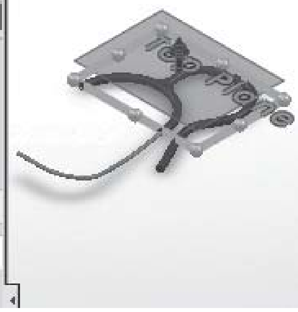


Step 10: Create *Sketch7* and *Sweep3-MetalTubing*: Face shown of *Sweep2-DoubleTube* > **Circle** from **Sketch** tab > hover over face until you see intersection symbol > click and drag to sketch and dimension circle shown > exit sketch > **Swept Boss/Base** from **Features** tab > *Sketch7* for profile and *Curve1* as path > ✔.

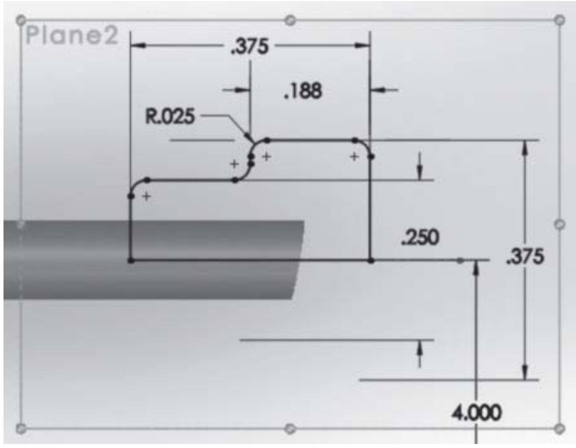


Step 11: Create *Plane 2*: **Reference Geometry** dropdown on **Features** tab **>** **Plane** > expand features tree > **Top Plane** > enter **0.625** for distance > ✔.





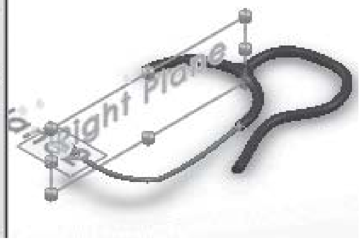
Step 12: Create *Sketch8*: *Plane2* as sketch plane > **Line** on **Sketch** tab > sketch and dimension lines shown > **Sketch Fillet** on **Sketch** tab > fillet and dimension corners shown > **Line** on **Sketch** tab > **Centerline** > sketch horizontal line shown > exit sketch.



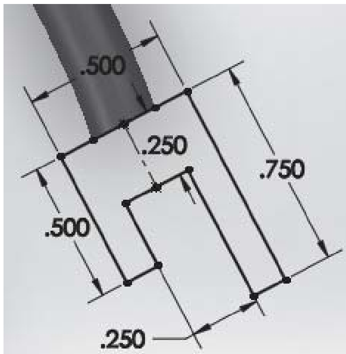
Note: The 4.0 in. dimension is measured from the end face of the *Sweep2-DoubleTube* feature created in Step 5.

Step 13: Create *Revolve1-Earplug*: *Sketch08* > **Revolved Boss/Base** on **Features** tab > centerline created in Step 12 as **Axis of Revolution** > ✔.

Step 14: Create other half: **Mirror** on **Features** tab > expand features tree > **Right Plane** as **Mirror Face/Plane** > *Sweep3MetalTubing* > *Revolve1-Earplug* > ✔.



Step 15: Create *Sketch9* and *Revolve2-DiaphramHousing*: **Top Plane** > **Revolved Boss/Base** on **Features** tab > **Line** on **Sketch** tab > sketch and dimension lines shown > exit sketch > line touching *Sweep2-DoubleTube* as **Axis of Revolution** > ✔.



HANDS-ON FOR TUTORIAL 8–9. Redesign the stethoscope 3D curve to make it look more realistic, as shown to the right.

problems

1. What is the difference between analytic and synthetic curves? Which family is better? Why?
2. If a parametric curve has u limits of u_{min} and u_{max}, what is the u value at its midpoint?
3. Use the tangent vector Eq. (8.4) to find the curve slopes in the XY, XZ, and YZ planes.
4. Find the parametric equation of the line connecting point (2, 1, 0) to point (–2, –5, 0). Find the line midpoint and its tangent vector. Sketch the line, and show the endpoints and the parameterization direction on the sketch.
5. For the line of Problem 4, find the coordinates of the points located at $u = 0.25$ from both ends of the line.
6. Reverse the parameterization direction of the line of Problem 4, and re-solve the problem.
7. Repeat Problems 4 and 5 but for points (1, 3, 7) and (–2, –4, –6).
8. Find the line slopes of Problem 7 in the XY, XZ, and YZ planes.
9. Find the equation of a circle with a diameter of 3.0 in. and a center at (1, –2). Find the four quarter points on the circle. Use Eq. (8.12). What are the tangent vectors and slopes at these points?
10. Repeat Problem 9 but for a circle with a radius of 1 in. and a center at the origin.
11. Use both Eqs. (8.11) and (8.12) to write the equation of an arc whose diameter is 1.5 in. and that is located in the third quadrant with a center at the origin.
12. A spline is given by:

$$\boldsymbol{P}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} 1 + 4u - u^2 \\ 2 - u + u^2 \end{bmatrix} \qquad 0 \leq u \leq 1$$

Find the endpoints and the midpoint of the spline. Also, find the tangent vector and the slope at the midpoint. Using your CAD/CAM system, create the spline defined by the above equation.

13. A spline is given by:

$$\boldsymbol{P}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} -2 - 2u^2 \\ -1 + u - 2u^2 \end{bmatrix} \qquad 0 \leq u \leq 1$$

Find the endpoints and the midpoint of the spline. Also, find the tangent vector and the slope at the midpoint. Using your CAD/CAM system, create the spline defined by the above equation.

14. A spline is given by:

$$\boldsymbol{P}(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} 2u \\ 2 - u \end{bmatrix} \qquad 0 \leq u \leq 1$$

What shape is this spline? Why? Verify your answer by creating it on your CAD/CAM system.

15. Use your CDA/CAM system to create a revolve defined by the following explicit equation of an ellipse:

$$\frac{x^2}{(6.5)^2} + \frac{y^2}{(3)^2} = 1 \quad -6.5 \leq x \leq 6.5$$

16. Use your CDA/CAM system to create a revolve defined by the following parametric equation of a spline:

$$x(u) = 3u^2 - 2u + 2 \qquad 0 \leq u \leq 1$$
$$y(u) = -3u^2 + 3u + 2 \qquad 0 \leq u \leq 1$$

17. Table 8.1 shows the (x, y, z) coordinates of 3D points that define the two edges (profiles) of a laboratory chair. Create each edge. Sweep a circle with radius of 1/2 in. along each curve. All dimensions are in inches.

–7	0	0
–7	1	0.5
–7	2	0.5
–7	5	0
–7	9	1
–5.5	9	3
–5.5	8.5	5
–5.5	9	10

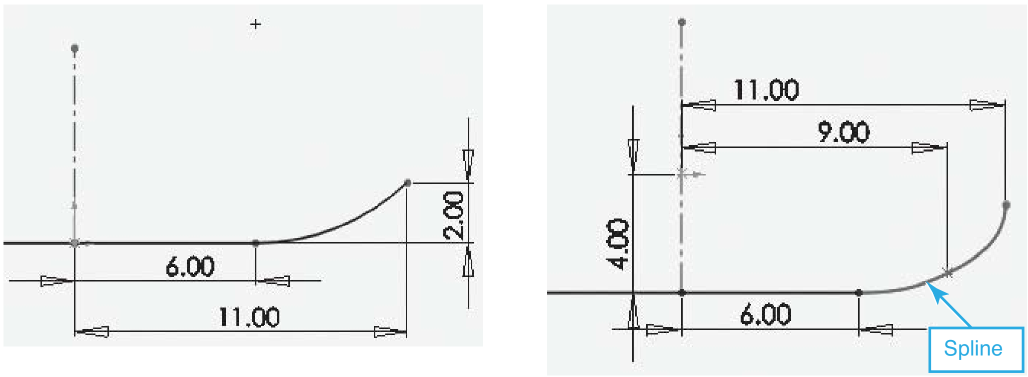
(A) Curve for Edge 1

7	0	0
7	1	0.5
7	2	0.5
7	5	0
7	9	1
5.5	9	3
5.5	8.5	5
5.5	9	10

(B) Curve for Edge 2

TABLE 8.1
3D points of the edges of a laboratory chair

18. Figure 8.18 shows the front and top sketches of half the profile of a skateboard. Use your CAD/CAM system to create the 3D curve that represents the skateboard profile. Sweep a circle with radius of 1/2 in. along the curve. All dimensions are in inches.

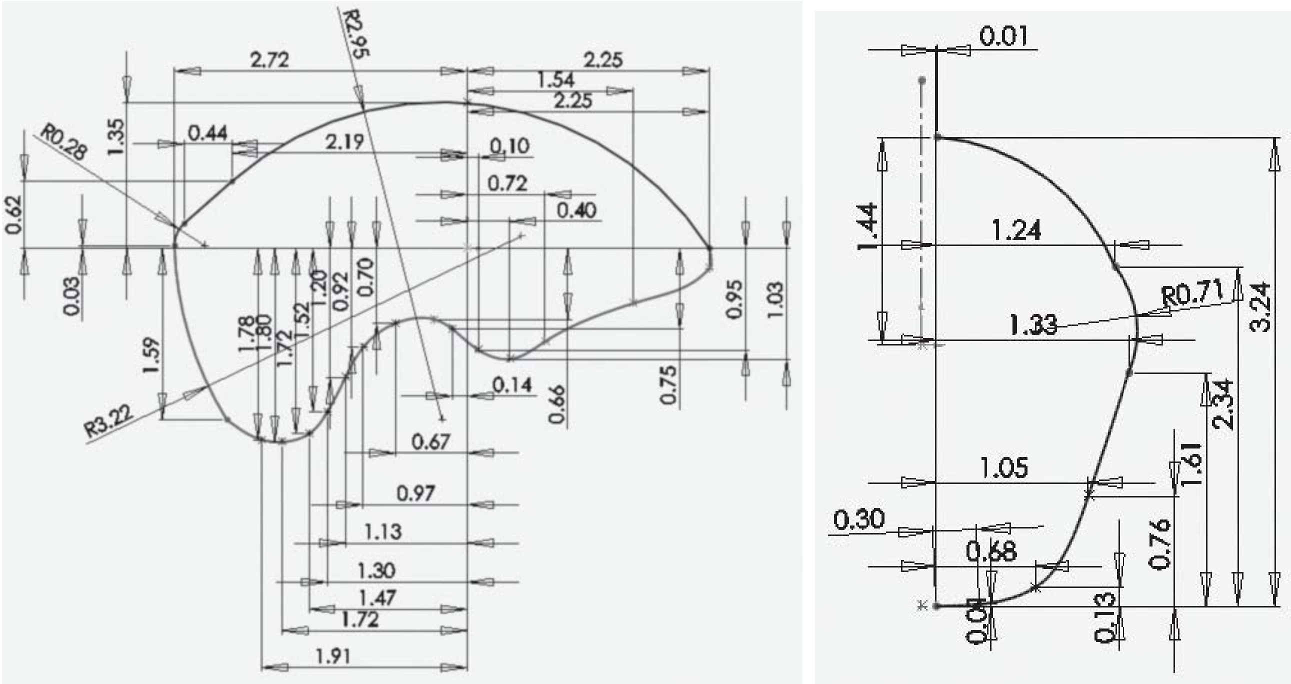


(A) Front plane sketch (symmetric about vertical axis)

(B) Top plane sketch (symmetric about vertical axis)

FIGURE 8.18
2D projections of a skateboard profile

19. Figure 8.19 shows the front and right sketches of a bicycle helmet. Use your CAD/CAM system to create the 3D curve that represents the helmet profile. Sweep a circle with radius of 1/2 in. along the curve. All dimensions are in inches.



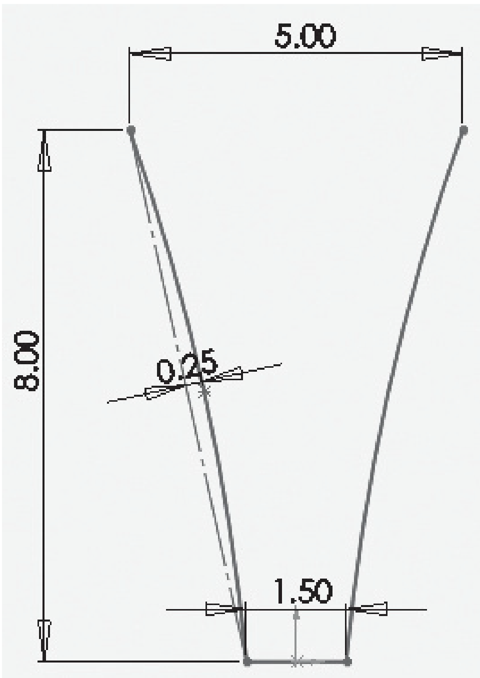
(A) Front plane sketch

(B) Right plane sketch

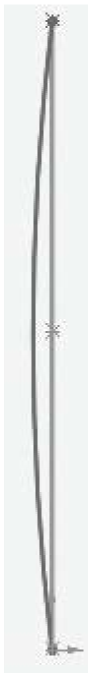
FIGURE 8.19
2D projections of a bicycle helmet profile

20. Figure 8.20 shows the front and right sketches of the profile of a weed whacker debris shield. Use your CAD/CAM system to create the 3D curve that represents the profile. Sweep a circle with radius of 1/2 in. along the curve. All dimensions are in inches.

FIGURE 8.20
2D projections of the profile of a weed whacker debris shield

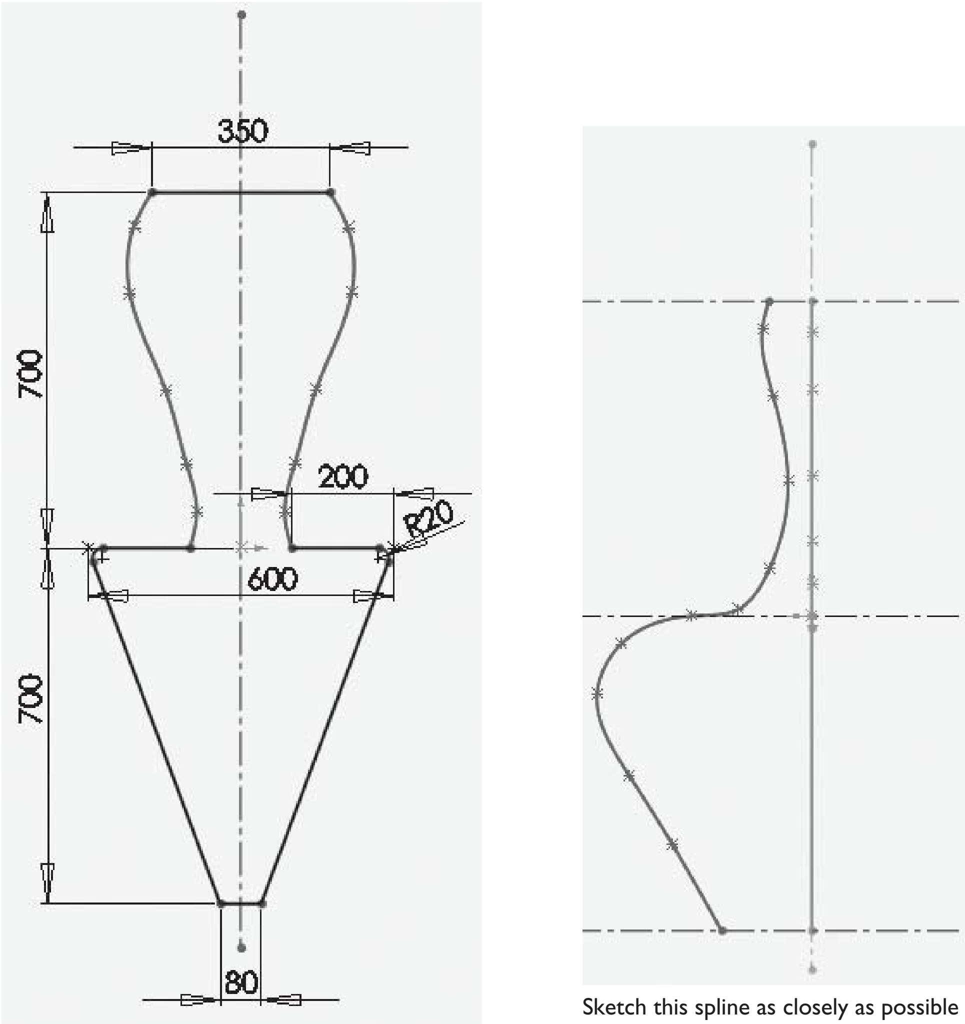


(A) Front plane sketch



Note: This arc passes through (matches) the two endpoints of the front plane sketch and has the same distance of 0.25 in. from the vertical line.
(B) Right plane sketch

21. Figure 8.21 shows the front and left sketches of the profile of an S-shaped chair. Use your CAD/CAM system to create the 3D curve that represents the profile. Sweep a circle with radius of 1/2 in. along the curve. All dimensions are in mm.

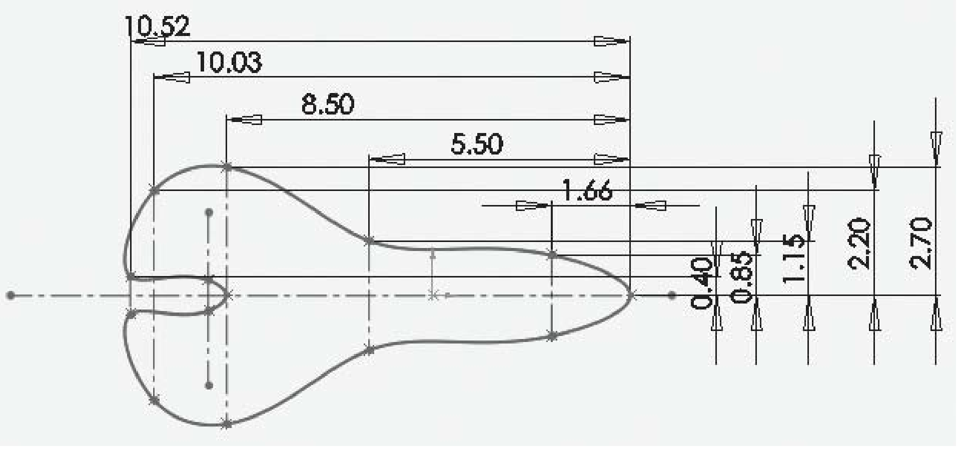


(A) Front plane sketch

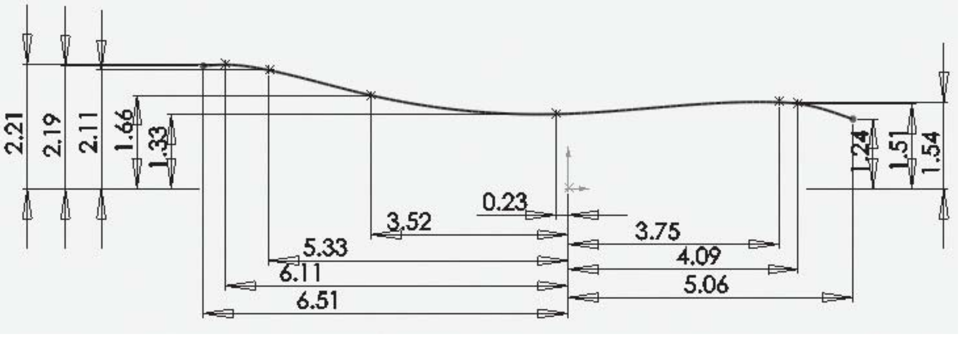
(B) Right plane sketch

FIGURE 8.21
2D projections of the profile of an S-shaped chair

22. Figure 8.22 shows the top and front sketches of the profile of a bike seat. Use your CAD/CAM system to create the 3D curve that represents the profile. Sweep a circle with radius of 1/2 in. along the curve. All dimensions are in inches.



(A) Top plane sketch



(B) Front plane sketch

FIGURE 8.22
2D projections of the profile of a bike seat

CHAPTER

9

Surfaces

9.1 Introduction

Even with the modeling power that the features and curves provide, they are still incapable of handling all the modeling problems and scenarios we encounter. Consider, for example, modeling an intricate computer mouse, a table spoon, a hair dryer, a shoe, or a sports car body. This class of objects, called *free-form parts*, is characterized by having free-form surfaces of intricate shapes. Surfaces, then, are the best modeling technique to use to create the solid models of these parts.

Surfaces build on curves, from both the theory and the modeling aspects. Creating surfaces requires the creation of curves first, as illustrated in the tutorials of this chapter. We view surfaces as extensions of curves. Surface equations, as we cover in this chapter, extend the parametric representation of curves. Thus, surfaces are classified in the same way as are curves: analytic and synthetic. Examples of analytic surfaces are plane, ruled surface, surface of revolution, and sweep. An example of synthetic surfaces is a spline surface.

Combining 3D curves and surfaces provides the most sophisticated and advanced modeling technique, which enables us to create a solid model of any part we may imagine. However, before we get carried away and let our design imagination run wild, we must always ask ourselves this basic question: Can the part be manufactured? A “yes” answer to this question is not enough. A follow-up question is, at what cost? These two questions explain why about 80% of the parts in practice can be modeled using extrusions and revolves, which we covered in Chapter 1.

9.2 Surfaces

Figure 9.1 shows all the surfaces we can create in SolidWorks. A **surface** is defined as a thin planar or nonplanar sheet that does not have a thickness. Creating a surface in itself is not useful. Surfaces are viewed as an intermediate step between creating curves and solids. We use them to create solids (features) that curves cannot create. Like features, surfaces are always listed in the features tree (**FeatureManager Design Tree**). If we were to expand a **Surface** node in the tree, we would see the sketch that defines the surface.

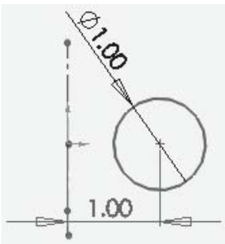
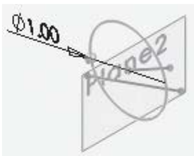
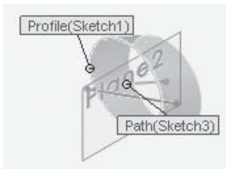
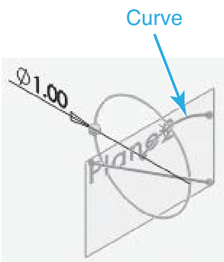
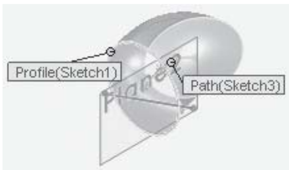
FIGURE 9.1
Available surfaces

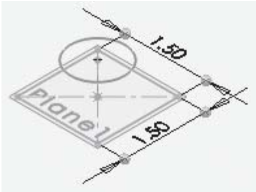
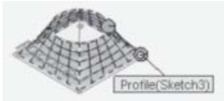
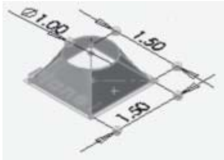
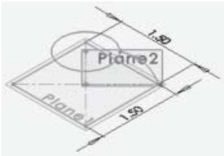
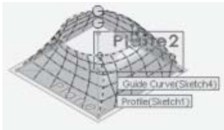
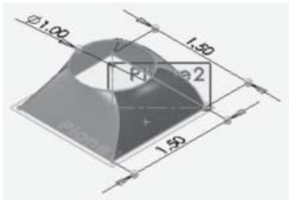
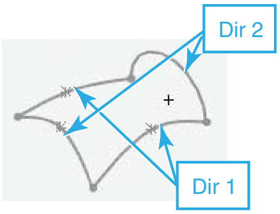
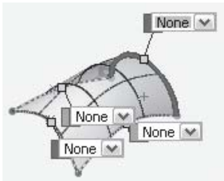
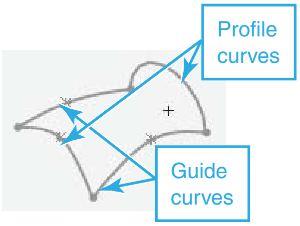
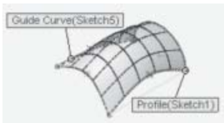
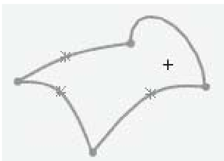
If we would like to master surface creation, we should be able to answer three fundamental questions:

1. What are the available surfaces that a CAD/CAM system offers for modeling parts?
2. What is the input required to create each surface?
3. Which surface should we use for a given modeling problem?

Figure 9.1 provides the answer to the first question. Table 9.1 answers the other two questions. It shows a simple, basic example of each surface. Keep in mind that the third question may have multiple answers; one of them is always the best answer. For example, we may use a loft or a sweep. However, if the surface has a constant cross section along a curve, a sweep is better to use because it requires fewer steps to create the surface. If the surface has a variable cross section, a loft is better to use.

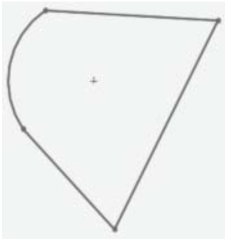
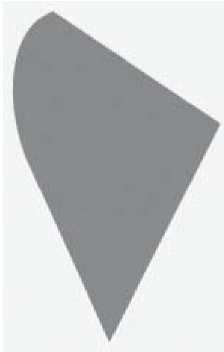
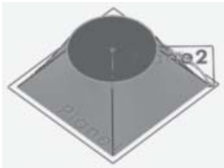
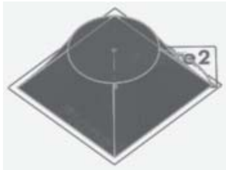
TABLE 9.1 Available Surfaces

<table>
<tr><th>No.</th><th>Surface</th><th>Input (sketch)</th><th>Resulting surface</th><th>When to use in modeling?</th></tr>
<tr><td>1</td><td>Extrusion</td><td>Cross section and a thickness
</td><td></td><td>• Use for surfaces with constant cross section.
• If needed, break a surface into subsurfaces, each with a constant cross section.</td></tr>
<tr><td>2</td><td>Revolve</td><td>Cross section, an axis of revolution, and an angle of revolution

</td><td>
</td><td>• Use for surfaces that are axisymmetric.
• If needed, break surface into subsurfaces; each is axisymmetric.</td></tr>
<tr><td rowspan="2">3</td><td rowspan="2">Sweep</td><td>Linear sweep: Cross section and a line as a path

</td><td>
</td><td>• Use for surfaces with constant cross section (CS) along a linear direction (path shown to the left) that may or may not be perpendicular to the cross section.
• If the path is perpendicular to the cross section, the linear sweep surface becomes an extrusion.</td></tr>
<tr><td>Nonlinear sweep: Cross section and a curve as a path

</td><td>
</td><td>• Use for surfaces with constant cross section (CS) along a nonlinear direction (path shown to the left) that may or may not be perpendicular to the cross section.</td></tr>
</table>

No.	Surface	Input (sketch)	Resulting surface	When to use in modeling?
4	Loft	Linear loft: At least two cross sections (profiles)  	  	• Use for surfaces with variable cross section along a given direction. • The cross sections are blended linearly from one end to the other.
		Nonlinear loft: At least two cross sections (profiles), and a curve as a guide curve  	  	• Use for surfaces with variable cross section along a given direction. • The cross sections are blended nonlinearly from one end to the other, along the guide curve.
5	Boundary	A set of connected curves in two directions  	 	• Use when you need to blend a surface between connected curves. It requires curves in two directions.
6	Loft (same as in No. 4, but different case)	Two or more curves, with or without guide curves  	 	• Use when you need to blend a surface between connected curves. It may or may not use guide curves.
7	Filled	A set of planar or nonplanar curves defining a closed boundary 	 	• Use when you need to create a surface patch defined by a closed boundary defined by multiple curves.

(*continued*)

TABLE 9.1 (*continued*)

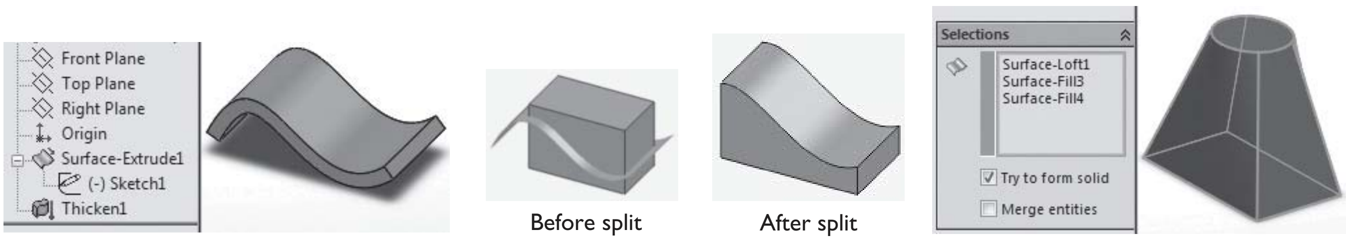
No.	Surface	Input (sketch)	Resulting surface	When to use in modeling?
8	Planar	A sketch consisting of curves defining a closed boundary (i.e., the curves are planar) 		• Use when you need to create planar surfaces bounded by a certain boundary.
9	Knit	A set of surfaces to knit (here we knit a surface fill on the top with a loft surface on the sides) 		• Use when you need to combine (knit) a set of surface patches into one surface. The resulting surface may or may not be closed. • Use the knit surface to cut a solid or convert the knit surface to a solid.

To learn surfaces quickly, keep in mind that the majority of them parallel features, with the difference that they do not have thickness. For example, the first four surfaces shown in Figure 9.1 parallel extruded boss, revolved boss, swept boss, and lofted boss, respectively. The tutorials in this chapter provide modeling examples. The other surfaces, shown in Figure 9.1 and not covered in Table 9.1, are dealt with in the tutorials of the chapter.

9.3 Using Surfaces in Solid Modeling

The goal of creating surfaces is not to create surfaces, but to use them to create complex features and solids. How can we do that? There are three methods as shown in Figure 9.2.

The first method is to thicken a surface to create a solid and is shown as the **Thicken** icon on the far right of the **Surface** tab in Figure 9.1. The user specifies a thickness. Depending on the surface shape and the thickness value that the user specifies, the thicken operation may fail and the solid may not be created. Understanding

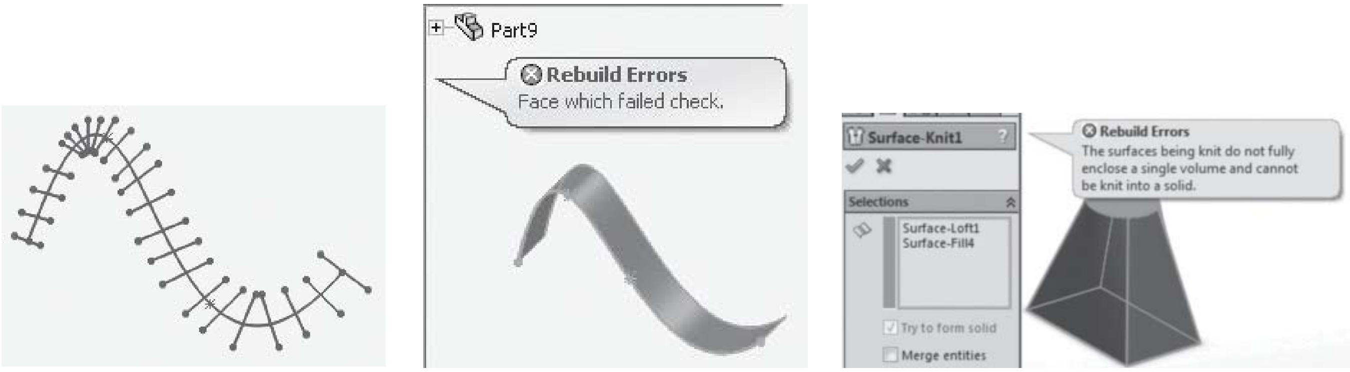


(A) Thicken a surface

(B) Cut solid with a surface

(C) Convert a surface to solid (surface must be closed)

FIGURE 9.2
Methods of using surfaces to create solids



(A) Surface with sharp changes　(B) Failure of thicken surface operation　(C) Failure of converting a surface to a solid

FIGURE 9.3
Errors of converting surfaces to solids

how the thicken operation works explains its potential failure. When we thicken the surface, conceptually the CAD/CAM system creates a copy of the surface and displaces it by the thickness amount. The cross sections of the thickened surface (solid) must stay perpendicular to the surface profile and nonintersecting, as shown in Figure 9.3A. If the thicken operation results in intersecting cross sections, the operation fails. Intersecting cross sections usually happen if the surface has abrupt sharp changes, as shown in Figure 9.3A. Figure 9.3B depicts the SolidWorks error message when the thicken operation fails.

The second method of using surfaces to create solids is to have a surface cut a solid (feature) to carve out the solid we want. If the surface is open (like a sheet), it splits the solid into two; we keep the part we want and throw away the other part. In this case, make sure that the surface extends beyond the solid faces from all directions. If the surface is closed (like a sphere) and enclosed inside a solid, it carves out part of the solid.

The third method is to convert the surface itself into a solid. In this case, the surface must be closed; otherwise, the conversion process fails with the error message shown in Figure 9.3C. Typically, we combine multiple surfaces (called *surface patches*) to create a fully closed surface by "knitting" them together using a knit surface, shown at the end of Table 9.1.

9.4 Surface Representation

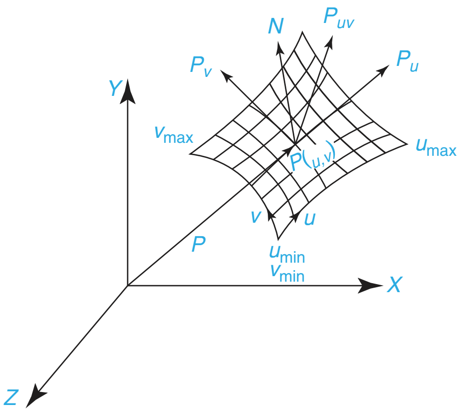


FIGURE 9.4
Parametric surface patch

CAD/CAM software uses parametric equations to represent surfaces. A surface equation uses two parameters (e.g., u and v) to describe the (x, y, z) coordinates of a point. Any point P on the surface is located (defined) by two values of the u and v parameters. Figure 9.4 shows the parametric representation of a surface patch. Each of the parameters u and v starts with a minimum value at one corner of the surface and finishes with a maximum value at the opposite corner. The parameter increases in value from the minimum value to the maximum one, thus defining the parameterization direction of the surface, in both u and v directions, as indicated by the arrows shown on the surface in Figure 9.4. The parameters have their lowest values typically at the bottom left corner of the surface patch, as shown in Figure 9.4.

Any point P on the surface is defined by its position vector $\boldsymbol{P}$ that is a function of u and v, that is,

$$\boldsymbol{P} = \boldsymbol{P}(u, v) \quad u_{\min} < u < u_{\max}, v_{\min} < v < v_{\max} \tag{9.1}$$

Alternatively, point P is defined by its (x, y, z) coordinates. Thus,

$$\boldsymbol{P} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix} \quad u_{\min} < u < u_{\max}, v_{\min} < v < v_{\max} \tag{9.2}$$

A point on a surface has four vectors: two tangent vectors, one in each direction ($\boldsymbol{P_u}$ and $\boldsymbol{P_v}$), a normal vector $\boldsymbol{N}$, and a twist vector $\boldsymbol{P_{uv}}$. The tangent vector $\boldsymbol{P_u}$ in the u direction at any point on the surface (Figure 9.4) is given by:

$$\boldsymbol{P}_u = \frac{\partial \boldsymbol{P}}{\partial u} = \begin{bmatrix} x_u \\ y_u \\ z_u \end{bmatrix} \quad u_{\min} < u < u_{\max}, v_{\min} < v < v_{\max} \tag{9.3}$$

Similarly, the tangent vector $\boldsymbol{P}_v$ in the v direction is given by:

$$\boldsymbol{P_v} = \frac{\partial \boldsymbol{P}}{\partial v} = \begin{bmatrix} x_v \\ y_v \\ z_v \end{bmatrix} \quad u_{\min} < u < u_{\max}, v_{\min} < v < v_{\max} \tag{9.4}$$

We explain the twist vector concept as follows. A tangent vector measures how a surface changes in the u or the v direction, that is, if we follow the surface in one direction only. For example, $\boldsymbol{P}_u$ shows how the surface $v =$ constant curves change in the u direction. Similarly, $\boldsymbol{P}_v$ shows how the surface $u =$ constant curves change in the v direction. The twist vector measures the change of the tangent vector; that is, $\boldsymbol{P}_{uv}$ measures the change of $\boldsymbol{P}_u$ in the v direction, and $\boldsymbol{P}_{vu}$ measures the change of $\boldsymbol{P}_v$ in the u direction. Mathematically, we express this behavior by this equation:

$$\boldsymbol{P_{uv}} = \boldsymbol{P_{vu}} = \frac{\partial^2 \boldsymbol{P}}{\partial u \partial v} = \begin{bmatrix} x_{uv} \\ y_{uv} \\ z_{uv} \end{bmatrix} \quad u_{\min} < u < u_{\max}, v_{\min} < v < v_{\max} \tag{9.5}$$

The normal vector $\boldsymbol{N}$ to the surface at a point is perpendicular to the plane formed by the two surface tangent vectors at the point. Thus, we can write:

$$\boldsymbol{N} = \boldsymbol{P_u} \times \boldsymbol{P_v} \quad u_{\min} < u < u_{\max}, v_{\min} < v < v_{\max} \tag{9.6}$$

We can also use $\boldsymbol{N} = \boldsymbol{P_v} \times \boldsymbol{P_u}$. The difference is a minus sign. CAD/CAM software uses one form consistently.

These four surface vectors provide ample insight into the surface behavior. Other surface calculations include the radius of curvature and the curvature, which we do not cover here. The relation between the curvature, χ, and the radius of curvature, ρ, is given by:

$$\chi = \frac{1}{\rho} \tag{9.7}$$

9.5 Plane Parametric Equation

Figure 9.5 shows the parametric representation of a plane defined by three points, P_0, P_1, and P_2. The parameterization directions of the plane are shown in the figure. The parametric equation of this plane is given (in vector form) by:

$$P = \boldsymbol{P}(u, v) = \boldsymbol{P}_0 + u(\boldsymbol{P}_1 - \boldsymbol{P}_0) + v(\boldsymbol{P}_2 - \boldsymbol{P}_0)$$
$$0 \leq u \leq 1, 0 \leq v \leq 1 \tag{9.8}$$

or (in scalar form),

$$\boldsymbol{P} = \boldsymbol{P}(u, v) = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix} = \begin{bmatrix} x_0 + u(x_1 - x_0) + v(x_2 - x_0) \\ y_0 + u(y_1 - y_0) + v(y_2 - y_0) \\ z_0 + u(z_1 - z_0) + v(z_2 - z_0) \end{bmatrix}$$
$$0 \leq u \leq 1, 0 \leq v \leq 1 \tag{9.9}$$

Using Eqs. (9.3) and (9.4), the tangent vectors of the plane are given by:

$$\boldsymbol{P}_u = \begin{bmatrix} x_u \\ y_u \\ z_u \end{bmatrix} = \begin{bmatrix} x_1 - x_0 \\ y_1 - y_0 \\ z_1 - z_0 \end{bmatrix} \tag{9.10}$$

$$\boldsymbol{P}_v = \begin{bmatrix} x_v \\ y_v \\ z_v \end{bmatrix} = \begin{bmatrix} x_2 - x_0 \\ y_2 - y_0 \\ z_2 - z_0 \end{bmatrix} \tag{9.11}$$

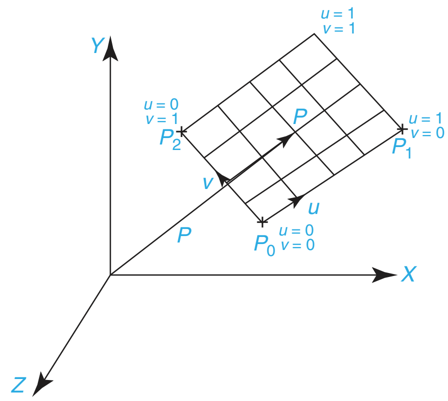


FIGURE 9.5
Parametric plane

Equations (9.10) and (9.11) show that the tangent vectors are constant (independent of u and v), as expected.

Using Eq. (9.5), the twist vector of the surface is zero because both tangent vectors of the plane as given by Eqs. (9.10) and (9.11) are independent of u *or* v. Thus,

$$\boldsymbol{P}_{uv} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{9.12}$$

Using Eq. (9.6), the normal vector to the plane is given by:

$$\mathbf{N} = \begin{vmatrix} i & j & k \\ x_1 - x_0 & y_1 - y_0 & z_1 - z_0 \\ x_2 - x_0 & y_2 - y_0 & z_2 - z_0 \end{vmatrix} \tag{9.13}$$

The determinant shown in Eq. (9.13) is the result of the cross product shown in Eq. (9.6). Eq. (9.13) produces a vector with a constant direction because it is independent of u and v.

The elegance of the parametric representation is that it is independent of the dimensionality of the modeling space whether it is 2D (x and y coordinates only) or 3D (x, y, and z coordinates). In other words, use $z = 0$ in the 3D equations and we get 2D modeling.

Example 9.1 **Find the parametric equation of a plane passing through the three points P_0 (−3, 2, 1), P_1 (0, −4, 2), and P_2 (4, 0, 7). Find the plane midpoint and all its vectors.**

Solution Using Eq. (9.9), the plane parametric equation is:

$$P = \boldsymbol{P}(u, v) = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix} = \begin{bmatrix} -3 + 3u + 7v \\ 2 - 6u - 2v \\ 1 + u + 6v \end{bmatrix} \quad 0 \leq u \leq 1, 0 \leq v \leq 1 \tag{9.14}$$

Differentiating Eq. (9.14) with respect to u and v, the tangent vectors are:

$$\boldsymbol{P}_u = \begin{bmatrix} x_u \\ y_u \\ z_u \end{bmatrix} = \begin{bmatrix} 3 \\ -6 \\ 1 \end{bmatrix} \tag{9.15}$$

$$P_v = \begin{bmatrix} x_v \\ y_v \\ z_v \end{bmatrix} = \begin{bmatrix} 7 \\ -2 \\ 6 \end{bmatrix} \quad (9.16)$$

The twist vector is given by Eq. (9.12). Using Eq. (9.13), the normal vector is:

$$N = \begin{vmatrix} i & j & k \\ 3 & -6 & 1 \\ 7 & -2 & 6 \end{vmatrix} = -34i - 11j + 36k \quad (9.17)$$

The midpoint of the line occurs at $u = 0.5$ and $v = 0.5$. Thus, substituting in Eq. (9.14), it is:

$$\begin{bmatrix} 2.0 \\ -2 \\ 4.5 \end{bmatrix}$$

HANDS-ON FOR EXAMPLE 9.1. Find the plane equation if P_2 becomes P_0.

9.6 Ruled Surface Parametric Equation

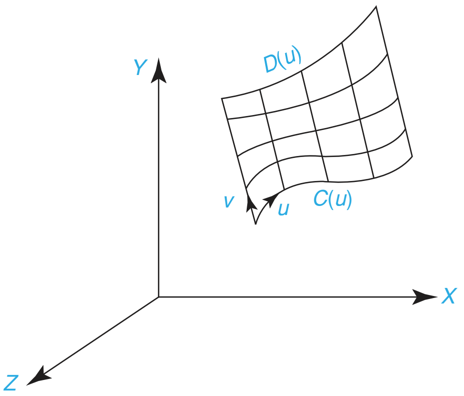


FIGURE 9.6
Parametric ruled surface

A ruled surface interpolates two planar curves linearly in the v direction, as shown in Figure 9.6. The two curves are known as the *rails* of the surface. The parameterization direction of the two curves should be the same. It defines the u direction of the ruled surface. The u limits of the two curves should be the same. The parametric equation of the ruled surface is given by:

$$P = P(u, v) = (1 - v)C(u) + vD(u)$$
$$u_{min} \leq u \leq u_{max}, \; 0 \leq v \leq 1 \quad (9.18)$$

Alternatively, we can express the coordinates of a point on the surface as follows:

$$P(u, v) = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix} = \begin{bmatrix} (1 - v)x_C(u) + vx_D(u) \\ (1 - v)y_C(u) + vy_D(u) \\ (1 - v)z_C(u) + vz_D(u) \end{bmatrix}$$
$$0 \leq u \leq 1, \; 0 \leq v \leq 1 \quad (9.19)$$

Where $x_C(u)$, $y_C(u)$, and $z_C(u)$ are the components of the curve $C(u)$ vector equation in the X, Y, and Z directions. Similarly, $x_D(u)$, $y_D(u)$, and $z_D(u)$ are the components of the curve $D(u)$ vector equation in the X, Y, and Z directions. The surface vectors can easily be determined once Eq. (9.19) has a specific form.

SolidWorks has a **Ruled Surface** on its **Surfaces** tab as shown in Figure 9.1. It is not the same as the ruled surface covered here. The ruled surface covered here corresponds to **Lofted Surface** on the SolidWorks **Surfaces** tab.

Example 9.2 Find the equation of the ruled surface connecting a spline and a line shown in Figure 9.7. Use Eq. (8.19) from Chapter 8 as the spline equation. The line connects the two points $P_0(1, 1)$ and $P_1(4, 1.5)$. Find the midpoint of the surface.

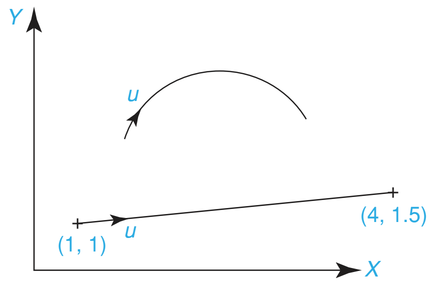


FIGURE 9.7
Ruled surface rails

Solution Using the two points, the line equation is:

$$P(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} 1 + 3u \\ 1 + 0.5u \end{bmatrix} \quad 0 \le u \le 1 \qquad (9.20)$$

Note that both rails have the same parameterization direction and the same u limits. Let us use the line as rail $\boldsymbol{C}(u)$ and the spline as rail $\boldsymbol{D}(u)$. Thus, Eq. (9.19) gives:

$$P(u, v) = \begin{bmatrix} x(u, v) \\ y(u, v) \end{bmatrix} = \begin{bmatrix} 1 + 3u + v - 3uv + 3u^2v - 2u^3v \\ 1 + 0.5u + v + 2.5uv - 3u^2v \end{bmatrix}$$

$$0 \le u \le 1, 0 \le v \le 1 \qquad (9.21)$$

The midpoint of the surface is located at (0.5, 0.5). Substituting these u and v values in Eq. (9.21) gives:

$$P(0.5, 0.5) = \begin{bmatrix} x(0.5, 0.5) \\ y(0.5, 0.5) \end{bmatrix} = \begin{bmatrix} 2.5 \\ 2 \end{bmatrix}$$

HANDS-ON FOR EXAMPLE 9.2. Find the surface vectors. Evaluate them at the surface midpoint.

Example 9.3 Implement Example 9.2 in SolidWorks. Use the **Measure** tool.

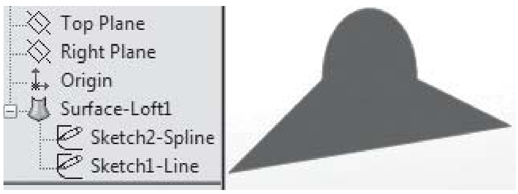
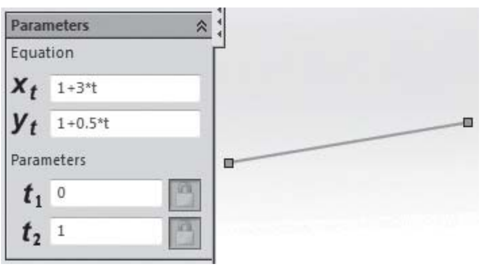


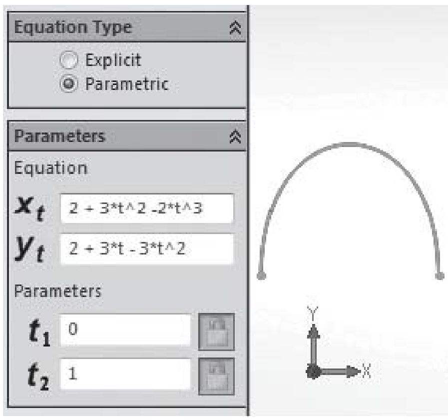
FIGURE 9.8
Lofted surface connecting a line and a spline

Solution Unlike parametric curves, we cannot enter surface parametric equations in SolidWorks. Thus, we enter the parametric equations of the two rails (curves), using them to create a boundary surface as shown in Figure 9.8. Note: To activate the **Surfaces** tab, right click **Features** tab (or any tab) > **Surfaces** from the list that pops up.

Step 1: Create *Sketch1-Line*: **File** > **New** > **Part** > **OK** > **Front Plane** > **Spline** dropdown on **Sketch** tab > **Equation Driven Curve** > **Parametric** > enter x and y equations and limits given by Eq. (9.20) as shown > ✔ > **File** > **Save As** > *example9.3* > **Save.**

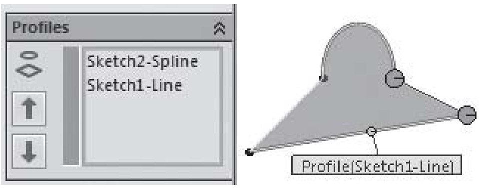


Step 2: Create *Sketch2-Spline*: **Front Plane** > **Spline** dropdown on **Sketch** tab > **Equation Driven Curve** > **Parametric** > enter x and y equations and limits given by Eq. (8.19) as shown > ✔.



Step 3: Create *Surface-Loft1*: **Lofted Surface** on **Surfaces** tab > *Sketch1-Line* > *Sketch2-Spline* > ✔.

Note: Make sure to pick curves at corresponding ends as shown; otherwise, you twist the surface.



> ***HANDS-ON FOR EXAMPLE 9.3.*** Verify the coordinates of the surface midpoint calculated manually in Example 9.2. Hint: Use the **Measure** tool on the **Evaluate** tab. Create a line connecting the midpoint of the line (hover until you see the midpoint symbol) and spline (hover until you see the point [1.5 and 90°]). Then, insert a point at the midpoint of the line just created. Last, use **Measure** to display point coordinates.

Example 9.4 Create a surface-to-surface intersection curve.

Solution Surface-to-surface intersection is a powerful modeling concept to create complex 3D curves. In this example we intersect two cylindrical surfaces as shown in Figure 9.9 and create a sweep along the intersection curve for visualization purposes (see the example hands-on). All dimensions are in inches.

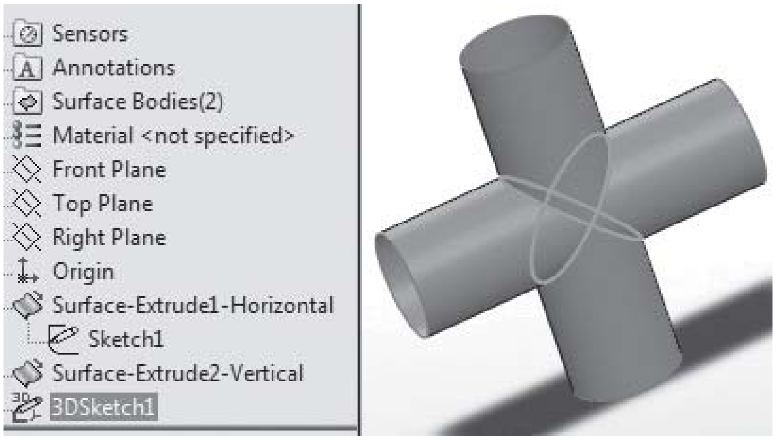
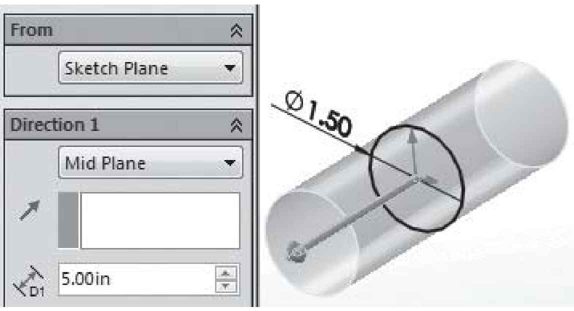
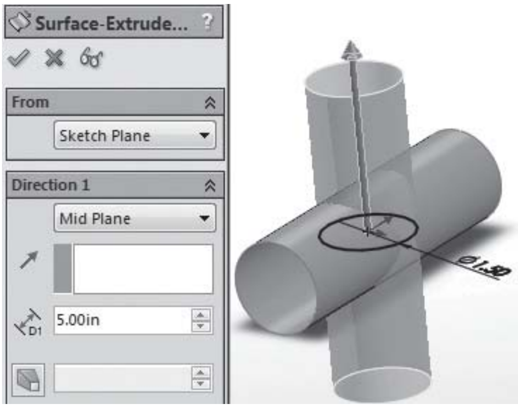


FIGURE 9.9
Surface-to-surface intersection curve

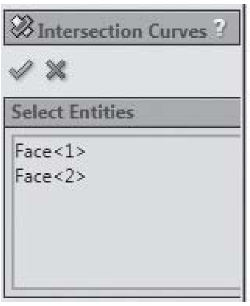
Step 1: Create *Surface-Extrude1-Horizontal*: **File > New > Part > OK > Front Plane > Circle** on **Sketch** tab > sketch a circle with 1.5 in. diameter > exit sketch > **Extruded Surface** on **Surfaces** tab > **Mid Plane** from dropdown > enter **5** for **D1** > ✔ > **File > Save As** > *example9.4* > **Save.**



Step 2: Create *Surface-Extrude2-Vertical*: **Top Plane > Circle** on **Sketch** tab > sketch and dimension circle shown > exit sketch > **Extruded Surface** on **Surfaces** tab > **Mid Plane** from dropdown > enter **5** for **D1** > ✔.



Step 3: Create intersection curve (*3DSketch1*): **Tools > Sketch Tools > Intersection Curve >**



Surface-Extrude1-Horizontal > *Surface-Extrude1-Vertical* > ✔ > ✔ > exit sketch.

Note: The surface-to-surface intersection is almost always a 3D curve; this is why SolidWorks creates *3DSketch1* node in features tree.

HANDS-ON FOR EXAMPLE 9.4. Sweep a circle along the intersection curve for better visualization purposes. Hint: Create a reference point on the curve using **Reference Geometry** on the **Features** tab > create a plane passing through the point and perpendicular to the curve. Note: If you do not see curves after creating or deleting surfaces, click: **View > Hide All Types** (it is a toggle).

9.7 Surface Visualization

Surfaces are harder to visualize than curves because of their curvature in space. As such, CAD/CAM systems provide surface visualization aids. These aids are as follows:

1. Surface mesh: The mesh (u-v grid) consists of u = constant and v = constant curves in a grid fashion as depicted in Figure 9.4. The higher the mesh size (u-v gridlines), the more visual the surface. However, keep in mind that the surface mesh size has no implication on surface accuracy because the accuracy is controlled by the surface equation. As a matter of fact, a CAD/CAM system uses the surface equation to generate the mesh gridlines. SolidWorks does not provide an easy way to display surface mesh except during creating a **Boundary Surface**.
2. Surface curvature: Each surface has a radius of curvature and curvature at any of its points. Let us assume that the surface has a radius of curvature of ρ at any of its points. The surface curvature at this point is the inverse of the radius of curvature and is given by Eq. (9.7).

   SolidWorks displays a curvature contour map superimposed on the surface. As the user moves the mouse over the surface, ρ and curvature are displayed. The user must invoke the display mode: **View** > **Display** > **Curvature** (this sequence is a toggle; do it again to turn the display mode off). The display of a surface curvature is a topological map shown in color to help demonstrate how the surface curves and twists in space.
3. Curve combs: This option applies to curves, not surfaces. It displays the curve curvature as the teeth of a comb. A tooth length at a curve point is proportional to the curvature amount. A circle has a constant curvature; thus, comb teeth lengths are equal.
4. Surface zebra stripes: These stripes indicate how smooth the tangency between adjacent surface patches are; that is, they indicate how smooth the transition is between patches across the shared edges. View Zebra stripes as follows: **View** > **Display** > **Zebra Stripes** (this sequence is a toggle; do it again to turn the display mode off).

9.8 Surface Management

After we create surfaces, we manage and manipulate them in different ways. We can investigate their curvatures and change their appearances by assigning them materials. Surfaces are harder than curves to manipulate during modeling.

9.9 Tutorials

The theme for the tutorials in this chapter is to practice creating and using surfaces. The tutorials show how to create surfaces and how to use them to create solids.

Tutorial 9–1: Create Basic Surfaces: Extrude, Revolve, Loft, Sweep, Knit, and Radiate

This tutorial covers some basic surface types. Figure 9.10 shows the resulting surface model. We also cover surface trim. All dimensions are in inches.

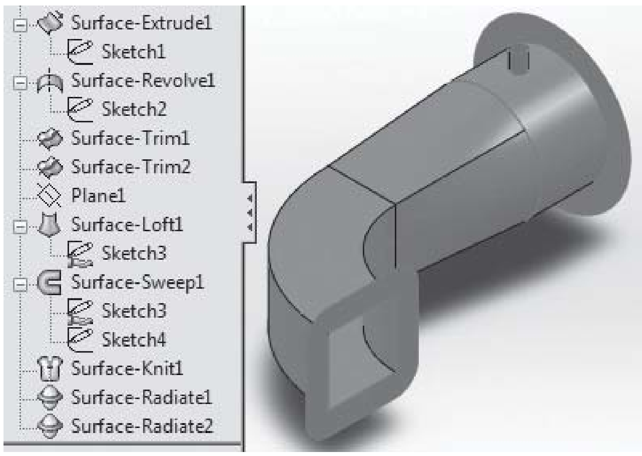
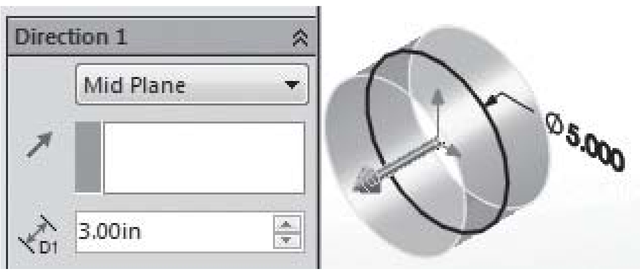
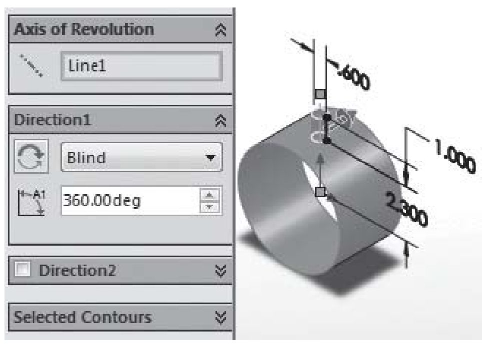


FIGURE 9.10
Air duct surfaces

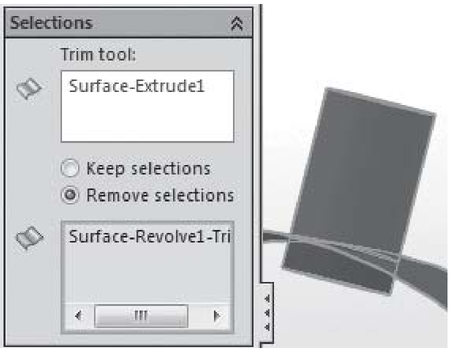
Step 1: Create *Surface-Extrude1*: **File** > **New** > **Part** > **OK** > **Front Plane** > sketch a 5.0″ circle centered on the origin > exit sketch > **Extruded Surface** on **Surfaces** tab > **Mid Plane** > enter **3.0** for **D1** > ✔ > **File** > **Save As** > *tutorial9.1* > **Save.**



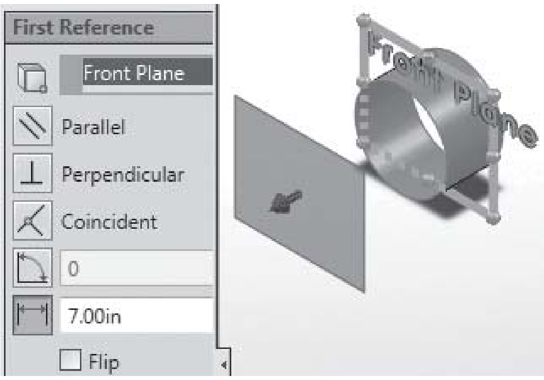
Step 2: Create *Surface-Revolve1*: **Front Plane** > sketch vertical line shown > sketch a centerline passing through Y axis > dimension line as shown > exit sketch > **Revolved Surface** on **Surfaces** tab > ✔.



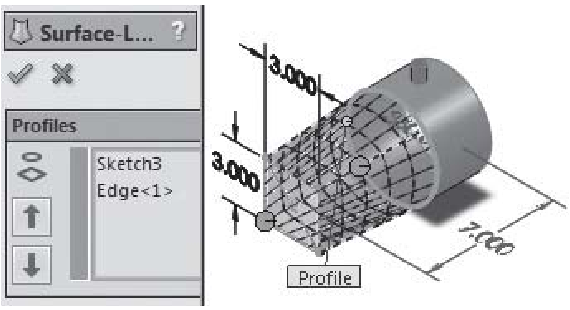
Step 3: Trim two surfaces: **Trim Surface** on **Surfaces** tab > *Surface-Extrude1* as trim tool > *Surface-Revolve1* as surface to trim (click revolved surface inside the extruded surface to remove this side) > ✔ > repeat to trim extruded surface (click extruded surface inside the revolved surface) to remove the surface part inside the surface revolve > ✔.



Step 4: Create *Plane1*: Create plane offset from **Front Plane** by 7″ > ✔.



Step 5: Create *Surface-Loft1*: Sketch 3 × 3 center rectangle shown on *Plane1* > **Lofted Surface** on **Surfaces** tab > *Sketch3* > edge of *Surface-Extrude1* > ✔.

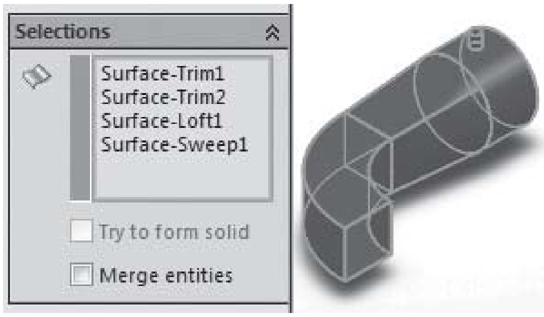


Note: You must select *Sketch3* first, then *Edge<1>* of *Surface-Extrude1*; otherwise, SolidWorks closes both ends (caps) of surface loft.

Note: The two big solid green circles shown indicate the corresponding points that SolidWorks uses to interpolate the two rails (curves). If they are not matching, the surface you create is twisted.

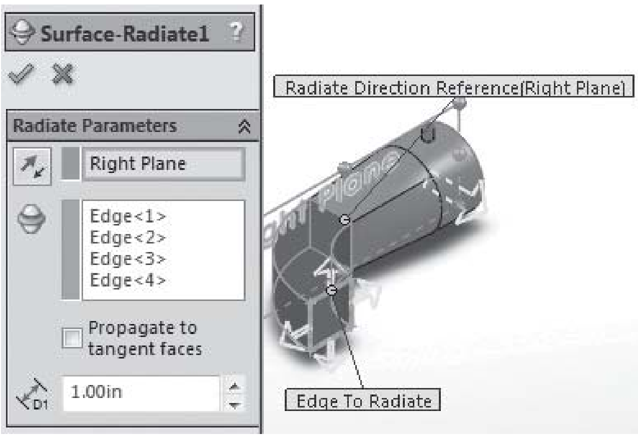
Step 6: Create *Surface-Sweep1*: Use **3 Point Arc** to create sketch shown (*Sketch4*) in the top plane > exit sketch > **Swept Surface** on **Surfaces** tab > *Sketch3* as profile and *Sketch4* as path > ✔.

Step 7: Create *Surface-Knit1*: **Knit Surface** on **Surfaces** tab > select all surfaces created thus far > ✔.



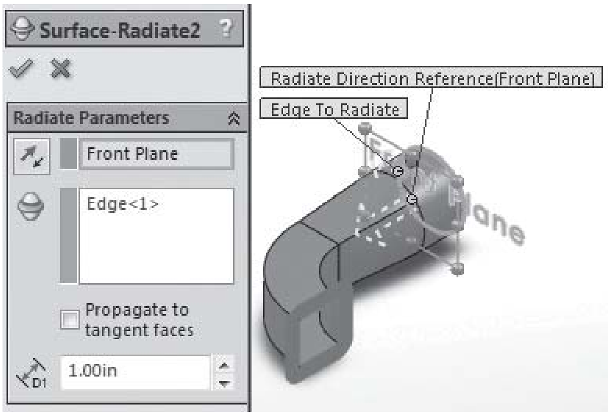
Note: There is no visible difference when the knit surface is created. Knit surface in itself has no modeling value for this tutorial other than practicing surface creation. Also, notice that the **Try to form solid** option is shown as inactive because the knot surface is not closed.

Step 8: Create *Surface-Radiate1*: **Insert** > **Surface** > **Radiate** > expand features tree > **Right Plane** > four edges of *Surface-Sweep1* > enter **1.0** for **D1** > ✔



Note: The **Radiate** surface is not readily available on the **Surfaces** tab.

Step 9: Create *Surface-Radiate2*: **Insert** > **Surface** > **Radiate** > expand features tree > **Front Plane** > edge of *Surface-Extrude1* > enter **1.0** for **D1** > ✔.



HANDS-ON FOR TUTORIAL 9–1. Suppress the two radiate surfaces. Modify the surface model to close the knit surface (create a closed volume) of Step 7 to convert it to a solid.

Tutorial 9–2: Create Basic Surfaces: Planar, Filled, Boundary, and Offset

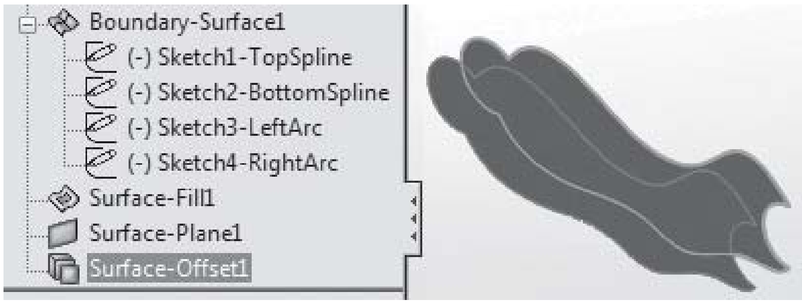
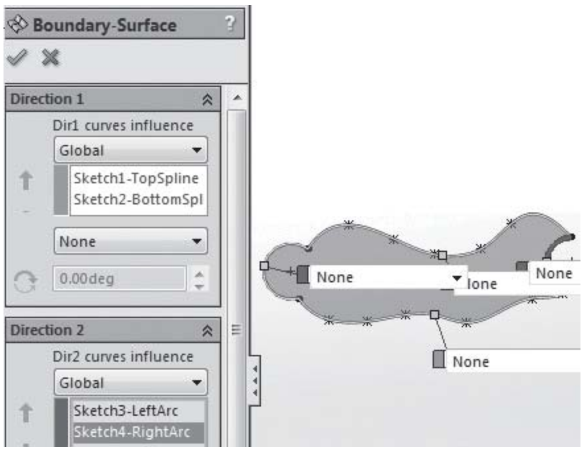


FIGURE 9.11
Surface patches

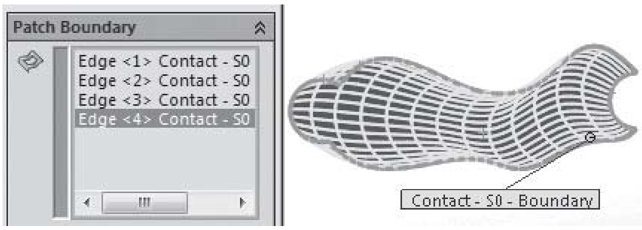
This tutorial covers other basic surface types (shown in Figure 9.11) not covered in Tutorial 9.1. Note: The first three surfaces (boundary, fill, and plane) of the features tree shown in Figure 9.11 overlap—not a good practice. However, they illustrate the idea that you can use the same boundary curves to create multiple types of surfaces.

Step 1: Create *Sketch1–Sketch4*: **File** > **New** > **Part** > **OK** > **Front Plane** > sketch freely to create the four separate sketches shown (each sketch holds one entity only) > use two splines (top and bottom) and two **3 Point Arc** (left and right) > exit each sketch > **File** > **Save As** > *tutorial9.2* > **Save.**

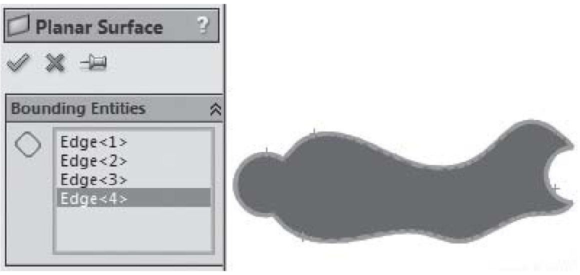


Step 2: Create *Boundary-Surface1*: **Boundary Surface** on **Surfaces** tab > two splines for **Direction 1** > click **Direction 2** box > select two arcs > ✔.

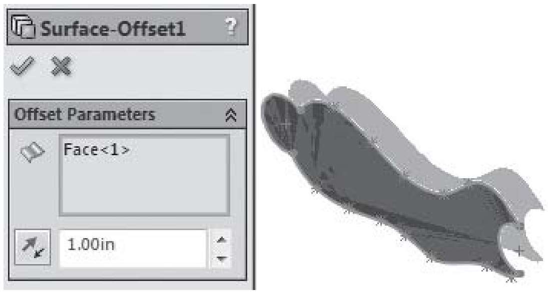
Step 3: Create *Surface-Fill1*: **Filled Surface** on **Surfaces** tab > select four entities in order (clockwise- or counterclockwise direction) > ✔.



Step 4: Create *Surface-Plane1*: **Planar Surface** on **Surfaces** tab > select four entities in order (clockwise- or counterclockwise direction) > ✔.



Step 5: Create *Surface-Offset1*: **Offset Surface** on **Surfaces** tab > select existing surface > enter **1.0** for offset distance > ✔.



HANDS-ON FOR TUTORIAL 9–2. Delete the above surfaces except the boundary surface (you need it to create a ruled surface). Use the edges of the boundary surface to create a SolidWorks **Ruled Surface**. How many edges does the system allow you to select? Why?

Tutorial 9–3: Visualize Surfaces

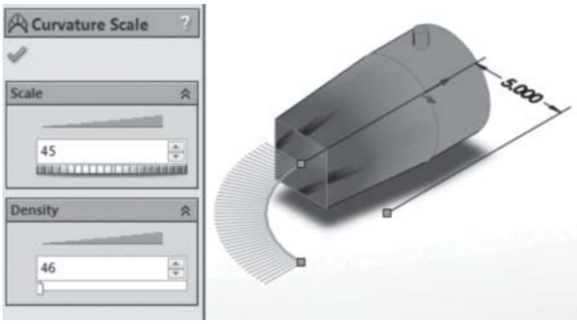
We use the three methods for surface visualization, curvature, combs, and zebra stripes, to analyze surface quality. Figure 9.12 shows a surface curvature map of the air duct shown in Figure 9.10. Surface curvature maps use the color codes—black, blue, green, red—in this order, indicating lowest (no) curvature (value of 0 for flat surfaces) to highest curvature (value of 1).

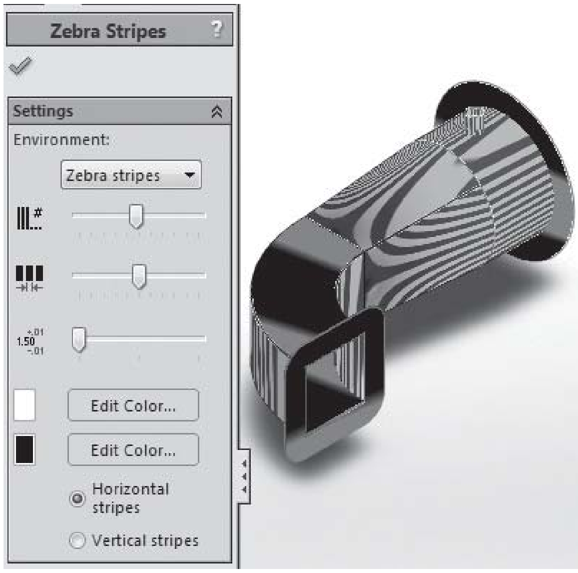
FIGURE 9.12
Surface curvature map

Step 1: Display surface curvature map: **View > Display > Curvature** (this is a toggle) > hover mouse over any point on a surface to read values > **File > Save As** > *tutorial9.3* > **Save.**

Note: Hovering over the circle gives a ρ of 3.5, which is the circle radius expected. Curvature $= 1/\rho = 0.285714285$

Step 2: Display curve combs: Edit *Sketch4* of *Surface-Sweep1* shown in Figure 9.10 > right click curve shown in Figure 9.12 > **Show Curvature Combs** (this is a toggle) from list that pops up > set **Scale** to 45 and **Density** to 46 (minimum) > ✔.





Step 3: Display zebra stripes: **View** > **Display** > **Zebra Stripes** (this is a toggle) > set values > ✔.

HANDS-ON FOR TUTORIAL 9–3. Display the curvature and the curvature combs for the circle of *Surface-Extrude1* shown in Figure 9.10.

Tutorial 9–4: Create an Artistic Bowl

This tutorial shows a combination of advanced modeling tools used to create an artistic bowl shown in Figure 9.13. We use 3D curves, surfaces, and features to create the bowl. This bowl can only be produced by injection molding. All dimensions are not included to simplify model presentation. All dimensions are in inches.

The main modeling idea here is to create 3D curves that we use to create two surfaces, a sweep and a loft. We then knit the two surfaces and thicken them to create a solid. Last, we create two extrude cuts to make the wholes.

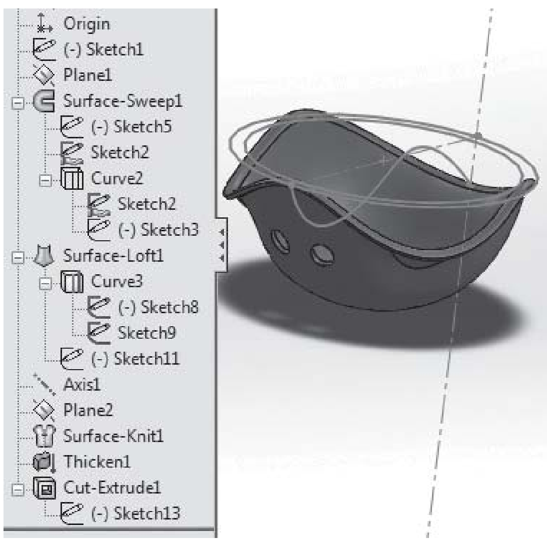
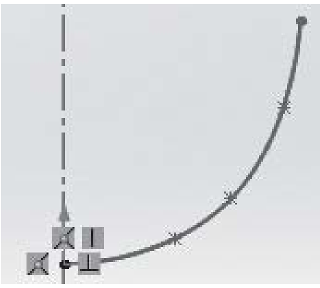


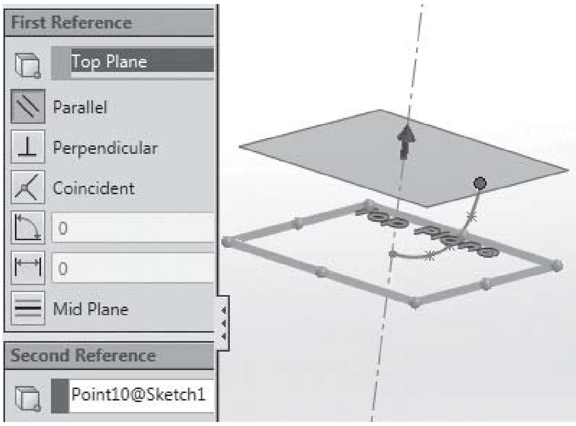
FIGURE 9.13
Surface-based solid

Step 1: Create *Sketch1*: **File > New > Part > OK > Front Plane** > sketch freely to create spline shown > exit sketch > **File > Save As** > *tutorial9.4* > **Save.**

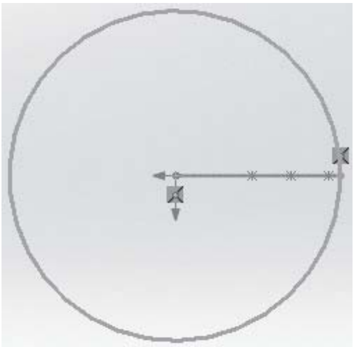


Note: We use **Convert Entities** later to copy this sketch to create *Sketch5*.

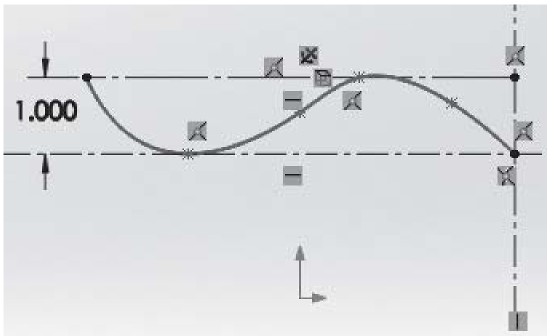
Step 2: Create *Plane1*: Create *Plane1* as shown. We use *Plane1* to create *Sketch2* in Step 3.



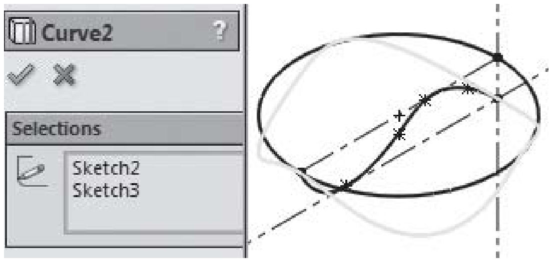
Step 3: Create *Sketch2:* Use *Plane1* to create a circle with center at origin and its perimeter coincident with top endpoint of spline of *Sketch1*.



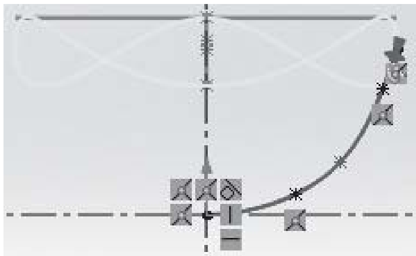
Step 4: Create *Sketch3:* Use **Right Plane** to sketch spline shown.



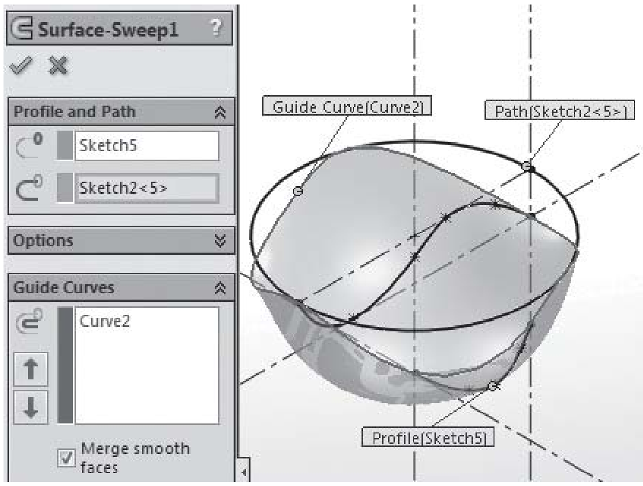
Step 5: Create *Curve2*: Use *Sketch2* and *Sketch3* to create the 3D curve.



Step 6: Create *Sketch5*: **Front Plane > Convert Entities** on **Sketch** tab > *Sketch1* > ✔.

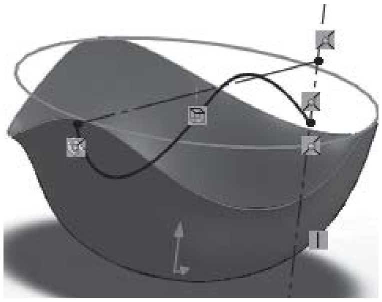


Step 7: Create *Surface-Sweep1*: **Swept Surface** on **Surfaces** tab > *Sketch5* as profile > *Sketch2* as path > *Curve2* as **Guide Curves** > ✔.

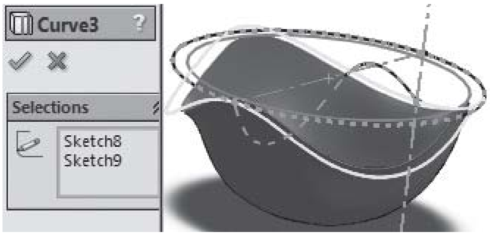


Step 8: Create *Sketch8:* Use *Plane1* to create a slightly larger circle than that of *Sketch2* of Step 3, as shown.

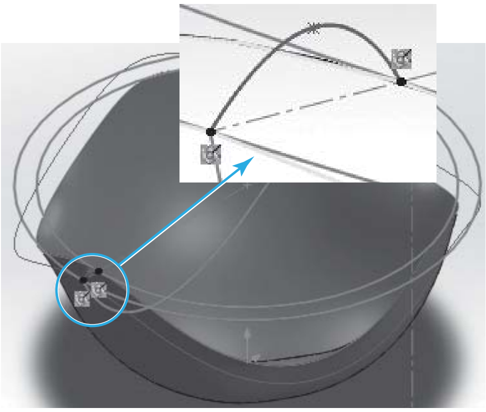
Step 9: Create *Sketch9*: **Right Plane** > **Convert Entities** on **Sketch** tab > *Sketch3* > ✔.



Step 10: Create *Curve3*: Use *Sketch8* and *Sketch9* to create the 3D curve.

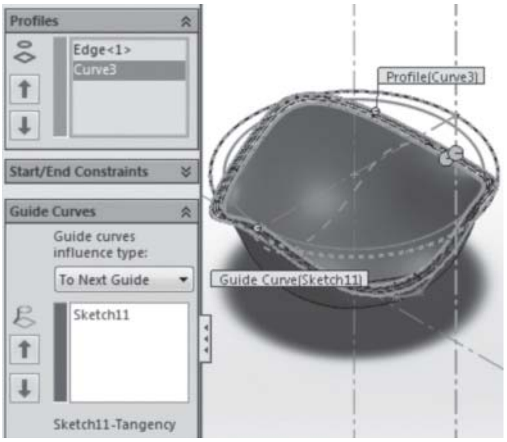


Step 11: Create *Sketch11*: **Right Plane** > sketch spline connecting the two circles as shown > exit sketch.

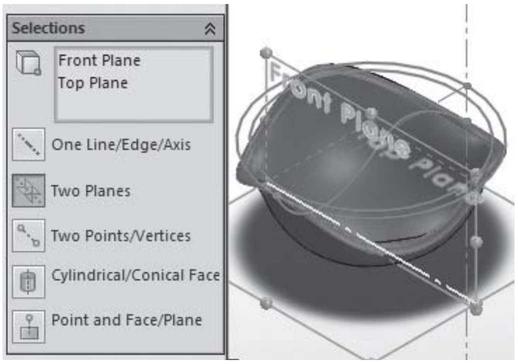


Note: This spline creates the curved lip of the bowl.

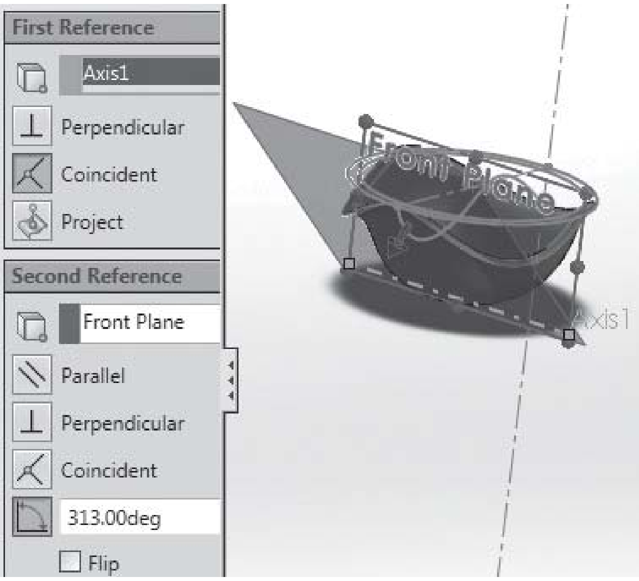
Step 12: Create *Surface-Loft1*: **Lofted Surface** on **Surfaces** tab > *Curve2* and *Curve3* as profile > *Sketch11* as **Guide Curves** > ✔.



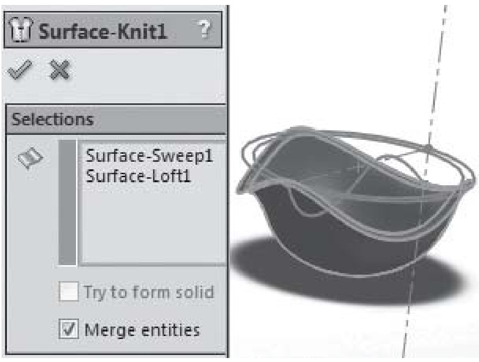
Step 13: Create *Axis1* to use to create *Plane2* in Step 14: **Reference Geometry** on **Features** tab > **Axis** > **Two Planes** > expand features tree > **Front Plane** > **Top Plane** > ✔.



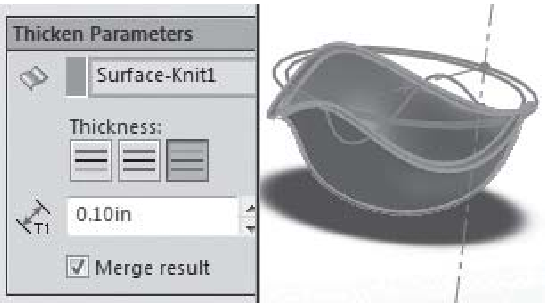
Step 14: Create *Plane2* passing by *Axis1* and at 313 degrees from **Front Plane**: **Reference Geometry** on **Features** tab > **Plane** > *Axis1* > **Front Plane** > 313 > ✔.



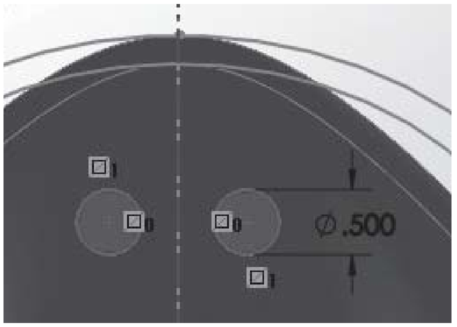
Step 15: Create *Surface-Knit1*: **Knit Surface** on **Surfaces** tab > *Surface-Sweep1* > *Surface-Loft1* > ✔.



Step 16: Convert *Surface-Knit1* to solid: Select *Surface-Knit1* node in features tree > **Thicken** on **Surfaces** tab > enter **0.1** for **T1** > ✔.



Step 17: Create *Sketch13* and *Cut-Extrude1* to make the bowl holes: *Plane2* > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > sketch two circles shown > exit sketch > **Through All** > ✔.



Tutorial 9–5: Use Surface Intersections

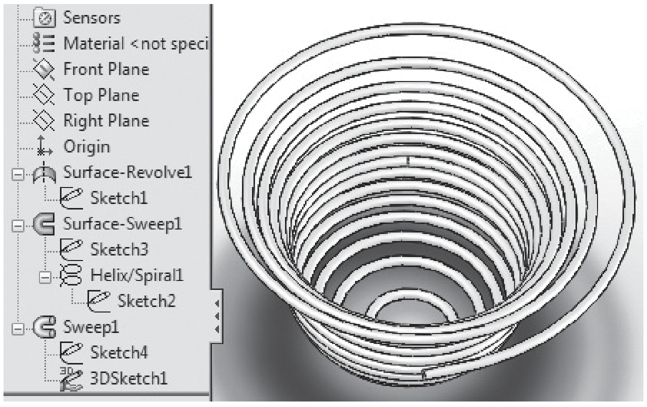
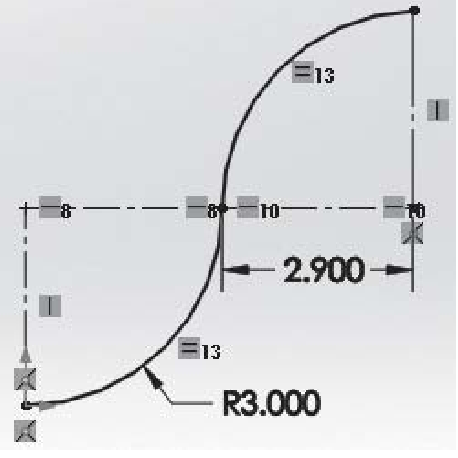


FIGURE 9.14
Surface-intersection-based solid

We use surface intersection to create intricate 3D curves. The idea here is to intersect two surfaces to create intricate intersection curves. If one of the intersecting surfaces is a helix surface, we generate visually pleasing and creative intersection curves, as this tutorial shows. These curves may be used to create sweeps of different shapes. We may use taper helixes to add to the visual complexity of the intersection curve.

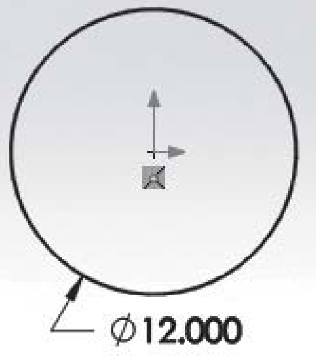
We create a surface revolve and a surface sweep. The surface revolve uses two arcs. The surface sweep uses a line as the sweep profile and a helix as the sweep path. Intersect the two surfaces to create a 3D curve. Sweep a circle (sweep profile) along the intersection curve (sweep path) for better visualization. The resulting swept feature is shown in Figure 9.14. All dimensions are in inches.

Step 1: Create *Sketch1* and *Surface-Revolve1*: **File** > **New** > **Part** > **OK** > **Front Plane** > sketch and dimension arcs shown > sketch a centerline passing through origin as shown > exit sketch > **Revolved Surface** on **Surfaces** tab > ✔ > **File** > **Save As** > *tutorial9.5* > **Save.**

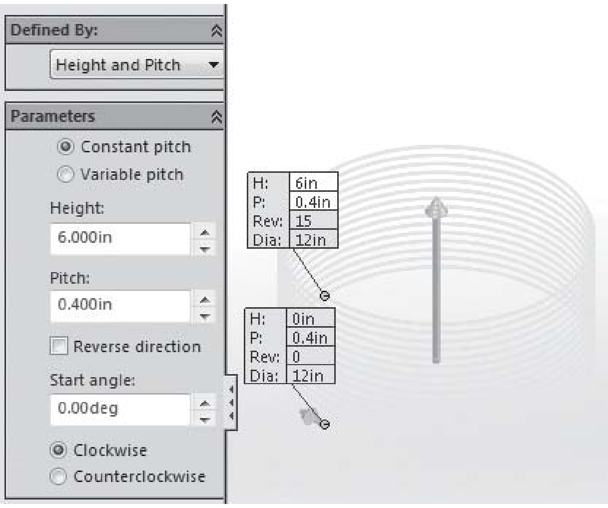


Note: The two arcs are of equal length and radius as shown by the "=13" relation symbol.

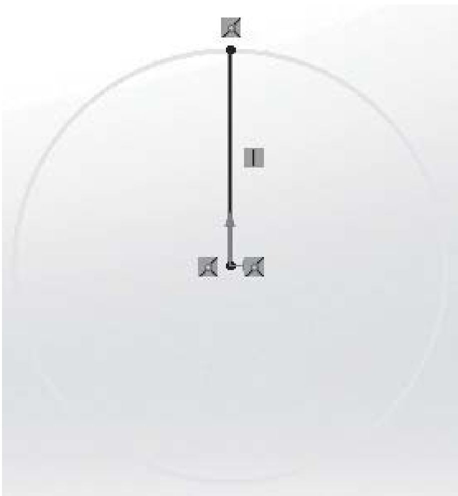
Step 2: Create *Sketch2*: **Top Plane** as sketch plane > create helix circle and dimension as shown > exit sketch.



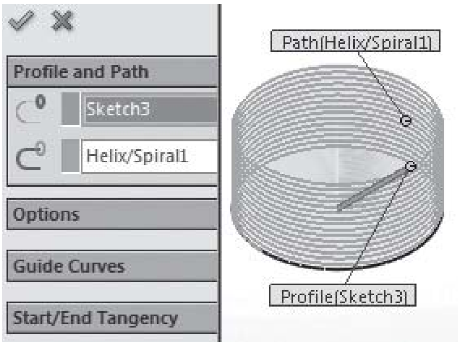
Step 3: Create *Helix/Spiral1*: **Insert** > **Curve** > **Helix/Spiral** > use the helix parameters shown (6 for **Height**, 0.4 for **Pitch**) > ✔.

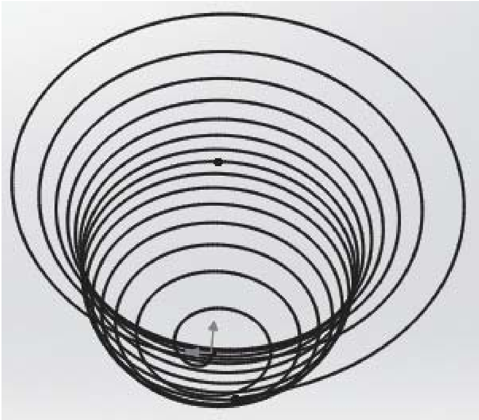


Step 4: Create *Sketch3*: **Top Plane** as sketch plane > create a vertical line from origin to perimeter of helix circle created in Step 2 > exit sketch.



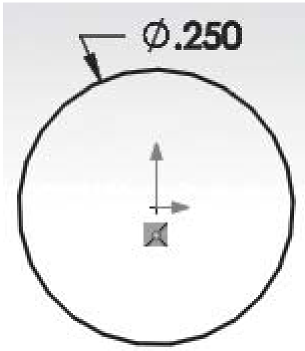
Step 5: Create *Surface-Sweep1*: **Swept Surface** on **Surfaces** tab > *Sketch3* for profile > *Helix/Spiral1* for path > ✔.



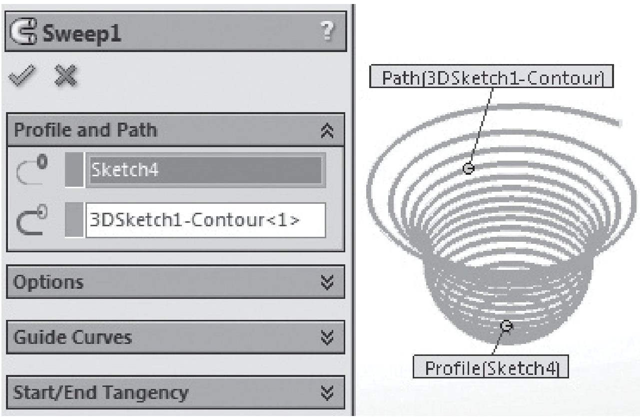


Step 6: Create intersection curve *3DSketch1*: **Tools** > **Sketch Tools** > **Intersection Curves** > *Surface-Revolve1* > *Surface-Sweep1* > ✔.

Step 7: Create *Sketch4*: **Front Plane** > sketch circle at origin and dimension as shown > exit sketch.



Step 8: Create *Sweep4*: **Swept Boss/Base** on **Features** tab > *Sketch4* for profile > *3DSketch1* for path > ✔.



HANDS-ON FOR TUTORIAL 9–5. Change the shape of the 3D curve by modifying either the surface revolve or the helix. Generate four cases: (1) change *Sketch1* of Step 1 into an arc, (2) change it to a circle, (3) repeat 1 for a taper helix (Step 3), (4) repeat 2 for a taper helix.

Tutorial 9–6: Create a Tablespoon

We use 3D curves and surfaces to create the tablespoon model shown in Figure 9.15. We create the sketches necessary to form the 3D projected curves to outline the spoon. We use a lofted surface to create the spoon surface and then thicken it. All dimensions are in mm.

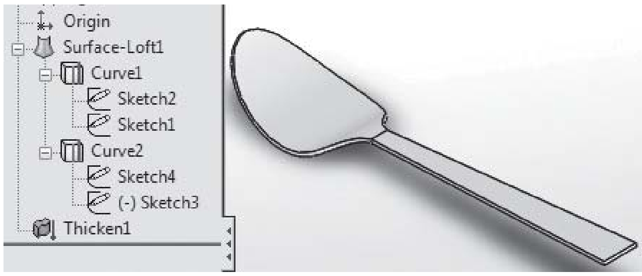
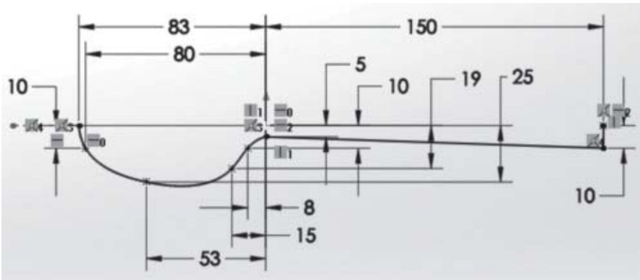
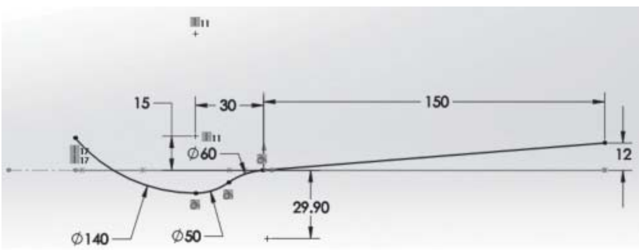


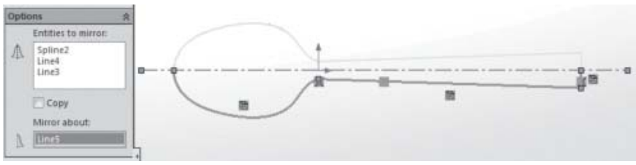
FIGURE 9.15
Tablespoon

Step 1: Create *Sketch1*: **File** > **New** > **Part** > **OK** > **Top Plane** > sketch and dimension spline and two lines as shown > exit sketch > **File** > **Save As** > *tutorial9.6* > **Save.**

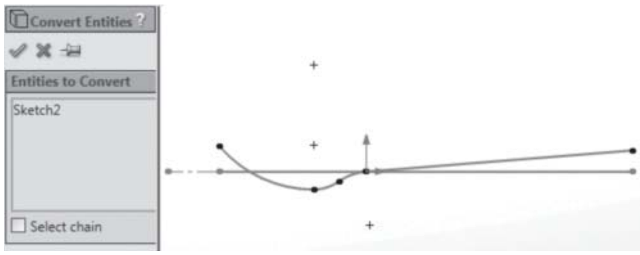




Step 2: Create *Sketch2*: Sketch and dimension three arcs and line shown above > exit sketch.

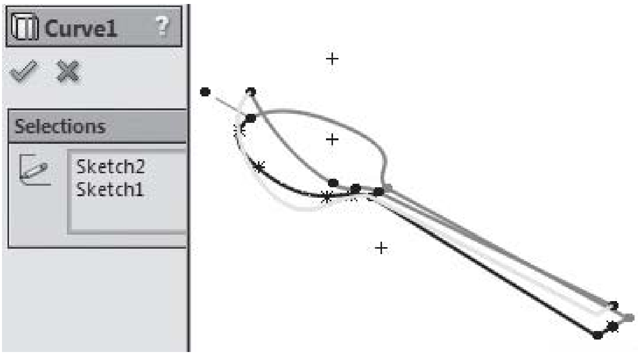


Step 3: Create *Sketch3*: **Top Plane** > **Convert Entities** on **Sketch** tab > *Sketch1* > ✔ > sketch horizontal centerline shown > **Mirror Entitie**s on **Sketch** tab > select spline and two lines > click **Mirror About** box > centerline just created > uncheck **Copy** checkbox shown > ✔ > exit sketch.

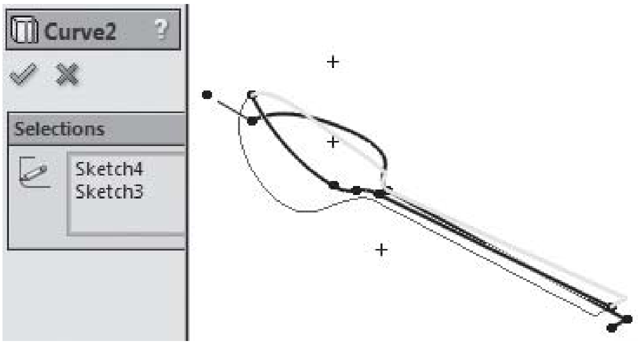


Step 4: Create *Sketch4*: **Front Plane** > **Convert Entities** on **Sketch** tab > expand features tree > *Sketch2* as shown above > ✔ > exit sketch.

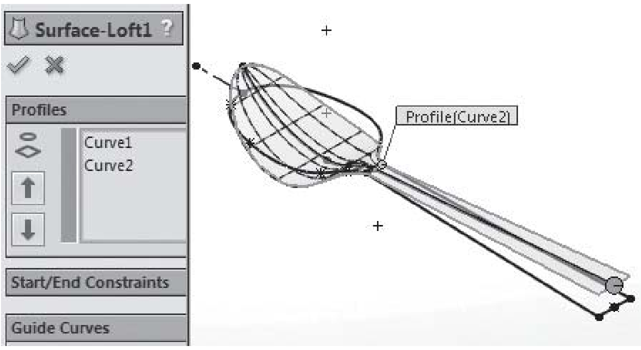
Step 5: Create *Curve1*: **Insert > Curve > Projected** > *Sketch1* > *Sketch2* > ✔.



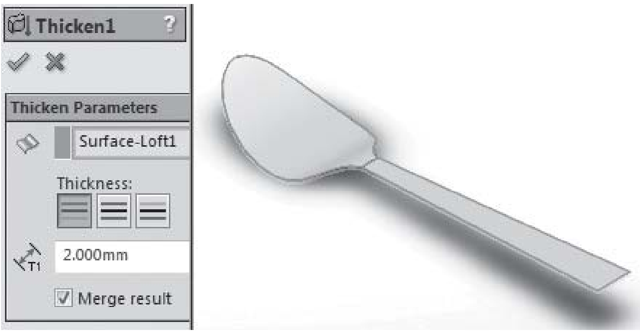
Step 6: Create *Curve2*: **Insert > Curve > Projected** > *Sketch3* > *Sketch4* > ✔.



Step 7: Create *Surface-Loft1*: **Lofted Surface** on **Surfaces** tab > *Curve1* > *Curve2* > ✔.



Step 8: Create *Thicken1* solid: **Thicken** on **Surfaces** tab > enter **2.0** for thickness **T1** > ✔.



Note: The **Thicken** operation provides the **Thickness** directions shown. Try them to see the effect.

HANDS-ON FOR TUTORIAL 9–6. The current spoon is shallow. Add two guide curves for the surface loft operation between the two 3D curves to deepen the spoon holding area. For the new spoon design, find the threshold thickness at which the thicken operation fails. Use the three **Thickness** options: **Thicken Side 1**, **Thicken Side 2**, or **Thicken Both Sides**. Explain why the thicken operation fails.

Tutorial 9–7: Create a Computer Mouse

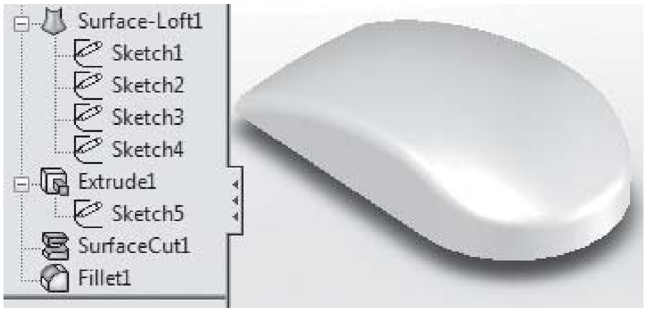
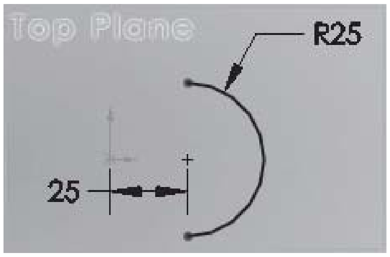


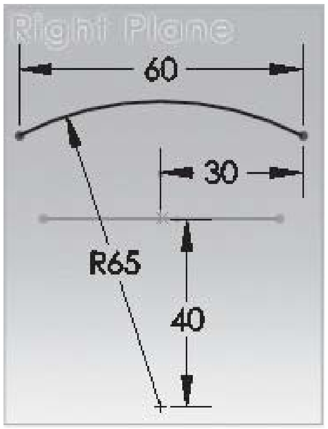
FIGURE 9.16
Computer mouse model

We create the mouse shown in Figure 9.16 using two different methods. One method cuts a solid with a surface while the other method thickens a surface. We create a lofted surface to use in both methods. The lofted surface is defined by four curves created in four separate sketches, as shown in Figure 9.16. All dimensions are in mm.

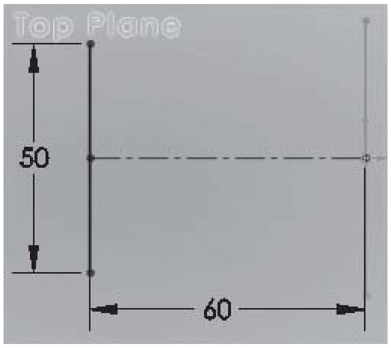
Step 1: Create *Sketch1*: **File > New > Part > OK > Top Plane >** sketch and dimension half circle shown > exit sketch.



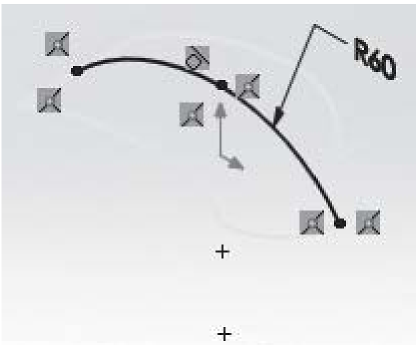
Step 2: Create *Sketch2*: **Right Plane** > sketch and dimension arc shown > exit sketch.



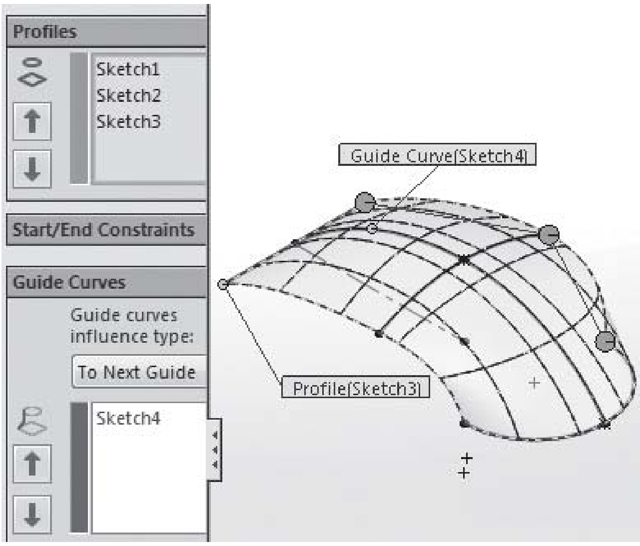
Step 3: Create *Sketch3*: **Top Plane** > sketch and dimension line shown > exit sketch.



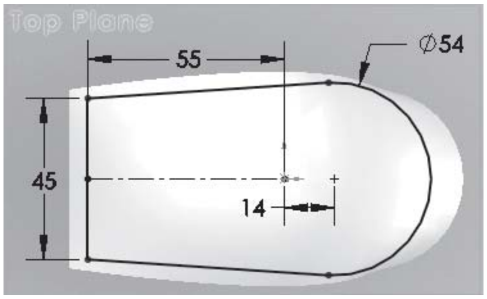
Step 4: Create *Sketch4*: **Front Plane** > sketch two arcs shown > exit sketch. **Note:** The right arc ends on the Y axis and the left arc is tangent to the right arc as shown. Arcs snap to right and left boundaries.



Step 5: Create *Surface-Loft1*: **Lofted Surface** on **Surfaces** tab > select *Sketch1–Sketch3* for **Profiles** > select *Sketch4* for **Guide Curves** > ✔.



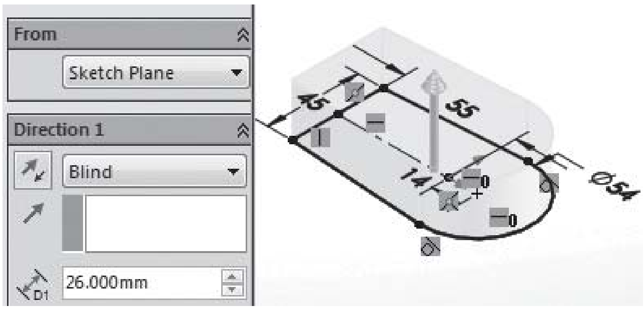
Step 6: Create *Sketch5*: **Top Plane** > sketch and dimension half circle and lines shown > exit sketch.



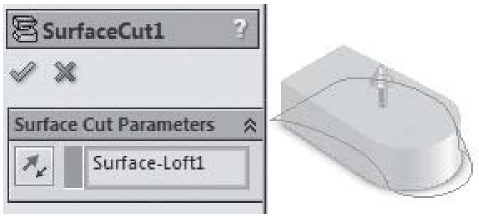
Step 7: Save model under two different names: *tutorial9.6Mouse1* and *tutorial9.6Mouse2*.

Method 1: Cut a solid with a surface

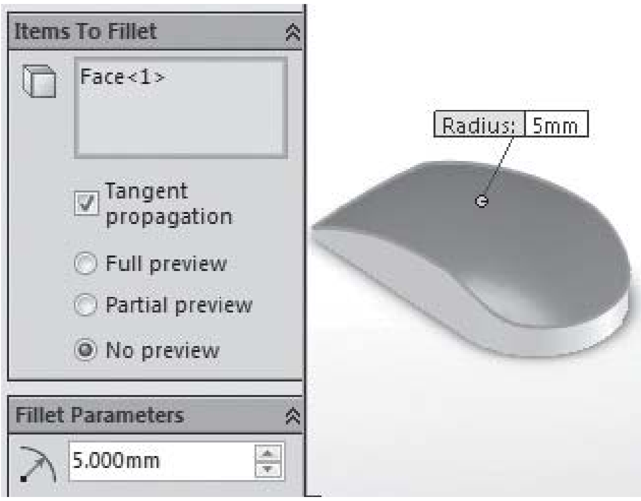
Step 8: Working with *tutorial9.6Mouse1*, create *Extrude1*: *Sketch5* > **Extruded Boss/Base** on **Features** tab > enter **26** for **D1** > ✔.



Step 9: Create *SurfaceCut1*: **Insert > Cut > With Surface** as shown and then hide the loft surface.

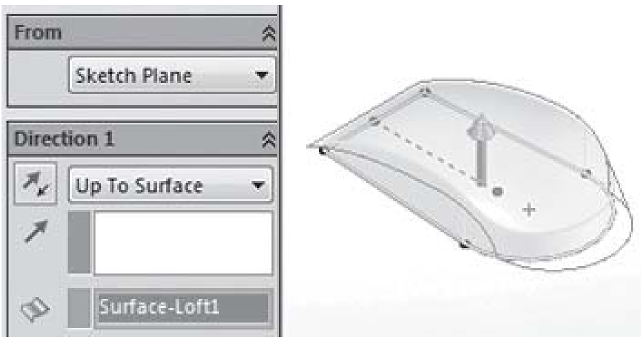


Step 10: Create *Fillet1*: **Fillet** on **Features** tab > top face > enter 5 for **Fillet Parameters** > ✔.

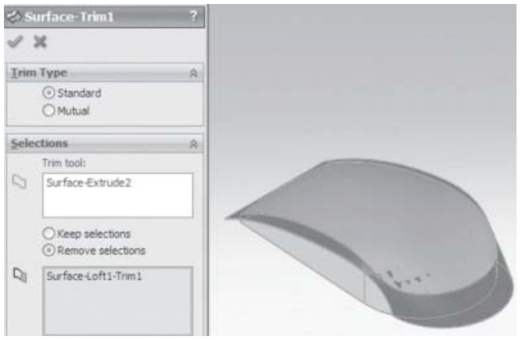


Method 2: Thickness surface

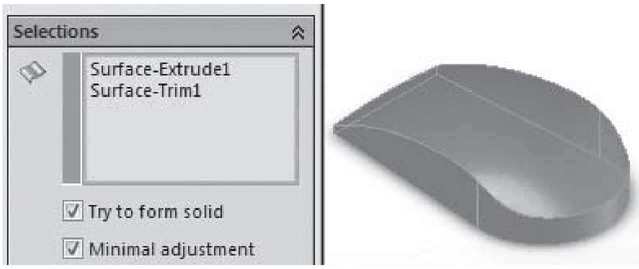
Step 11: Working with *tutorial9.6Mouse2*, create *Surface-Extrude1: Sketch5* > **Extruded Boss/Base** on **Features** tab > **Up To Surface** from **Direction 1** dropdown > expand features tree > *Surface-Loft1* > ✔.



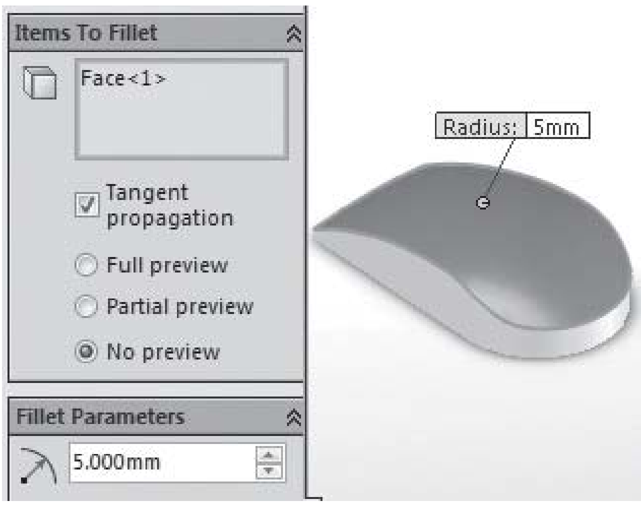
Step 12: Create *Surface-Trim1*: **Insert > Surface > Trim** > *Surface-Extrude2* for **Trim tool** > *Surface-Loft1* > ✔.



Step 13: Create *Surface-knit* and solid: **Knit Surface** on **Surfaces** tab > *Surface-Extrude1* > *Surface-Trim1* > check off **Try to form solid** box shown > ✔.



Step 14: Create *Fillet1*: **Fillet** on **Features** tab > top face > enter **5** for **Fillet Parameters** > ✔.



HANDS-ON FOR TUTORIAL 9–7. Shell *tutorial9.7_Mouse1* using a thickness of 2 mm. Cut a 20 mm hole at the bottom face to see the shelling result.

Help: When you shell the model, you will not see any visual difference. It is only visible when you create the hole that you can see the hollow space inside the shell.

Tutorial 9–8: Create a Baseball Hat

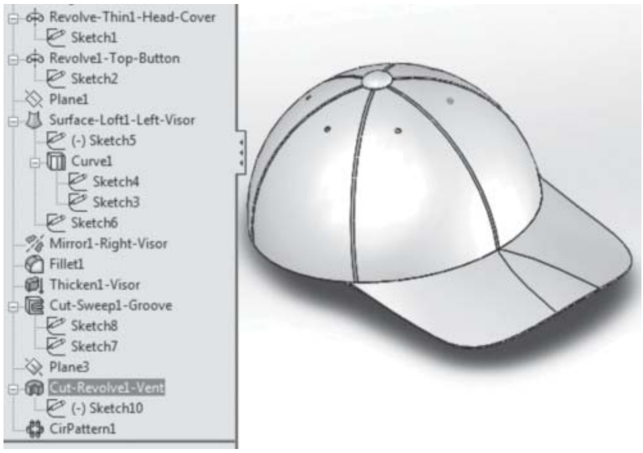
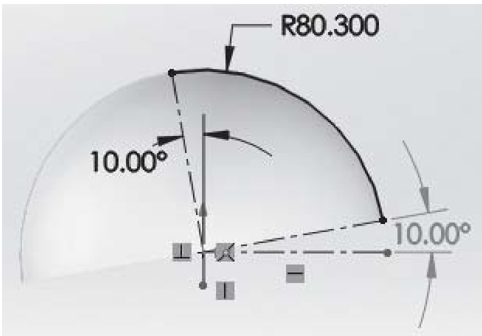


FIGURE 9.17
Baseball hat

We use 3D curves and surfaces to create the baseball hat shown in Figure 9.17. The hardest part of the baseball hat to create is the visor because it requires nonplanar (3D) curves to define. We use the projected curves modeling technique together with surfaces to create the visor. When we sketch the 2D curves to define a 3D curve, we sketch freely; that is, no prior coordinates are known. We evaluate the shape of the visor after we create it. If we do not like it, we go back and tweak the curve data. We repeat this process until we are satisfied with the visor shape and look.

The modeling plan is as follows. The head cover and the button are created using simple revolve features. The visor is created using 3D curves and surfaces. The groove lines are created using swept cuts and circular patterns. All dimensions are in mm.

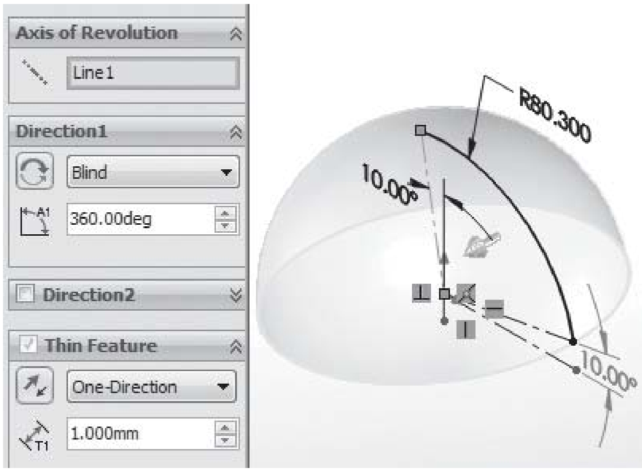
Step 1: Create *Sketch1*: **File** > **New** > **Part** > **OK** > **Front Plane** > sketch and dimension arc shown > exit sketch.



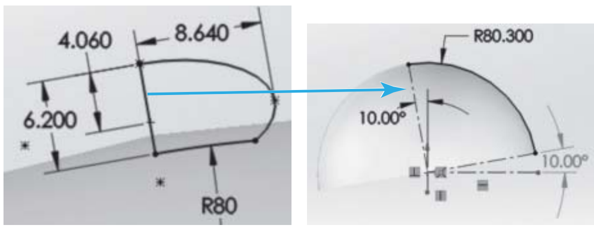
Note: The arc is tilted 10° from the horizontal; the arc angle is 90°.

Note: The revolve axis is the line tilted 10° from the vertical direction.

Step 2: Create *Revolve-Thin1-Head-Cover*: *Sketch1* > **Revolved Boss/Base** from **Features** tab > **No** (to closing the sketch) > check box (to turn on **Thin Feature**) > use 360° and 1 mm for **T1** as shown > **Centerline** (for axis of revolve; use the line that is 10° from vertical) > ✔.

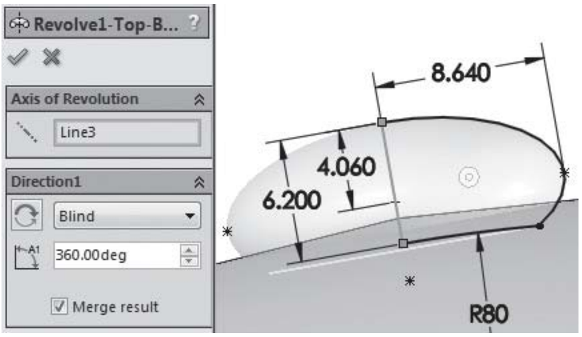


Step 3: Create *Sketch2*: **Front Plane** > sketch and dimension ellipse, circular arc, and line shown > exit sketch.

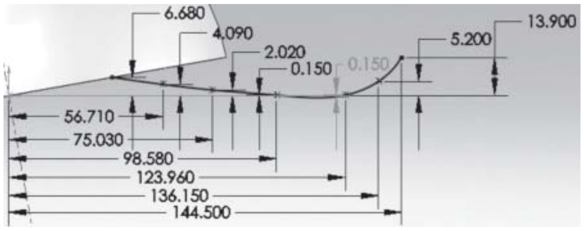


Note: The R80 arc has a center at origin.

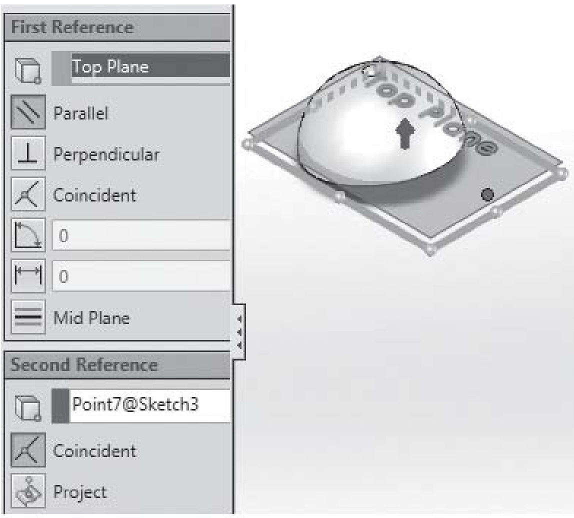
Step 4: Create *Revolve1-Top-Button*: *Sketch2* > **Revolved Boss/Base** on **Features** tab > centerline shown > ✔.



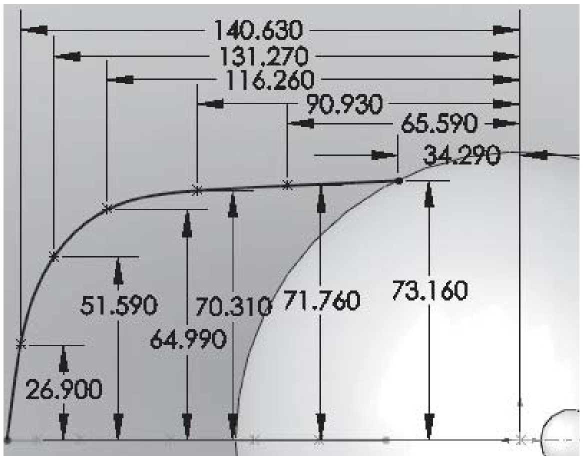
Step 5: Create visor *sketch3*: **Front Plane** > sketch spline with seven points as shown > exit sketch.



Step 6: Create *Plane1*: **Reference Geometry** > **Plane** > expand features tree > **Top Plane** > **Parallel** > endpoint of spline created in Step 5 > exit sketch.

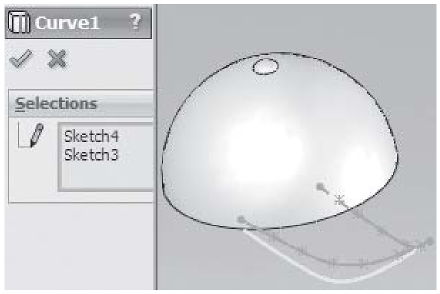


Step 7: Create visor *Sketch4*: *Plane1* > sketch spline with seven points as shown below > exit sketch.

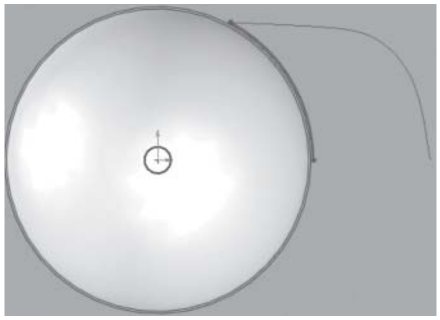


Note: The endpoint (bottom left) that shows no coordinates is coincident to the endpoint of the spline created in Step 5.

Step 8: Create visor 3D curve *Curve1*: **Insert > Curve > Projected** > *Sketch3* > *Sketch4* > ✔.

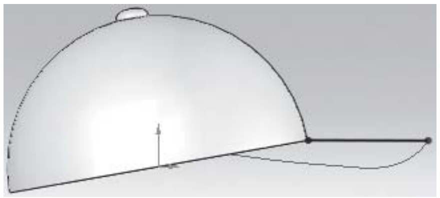


Step 9: Create visor *Sketch5*: Using the bottom face of the head cover as sketch plane, sketch arc shown > exit sketch.

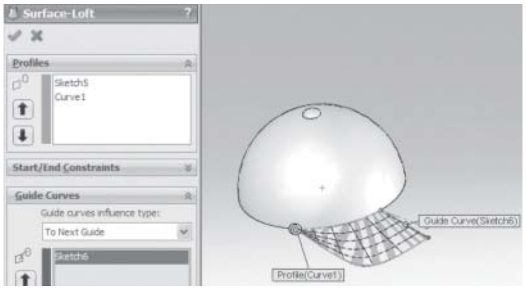


Note: Make sure that one endpoint of the arc is coincident with one endpoint of the 3D curve, and the other endpoint is horizontal to the origin.

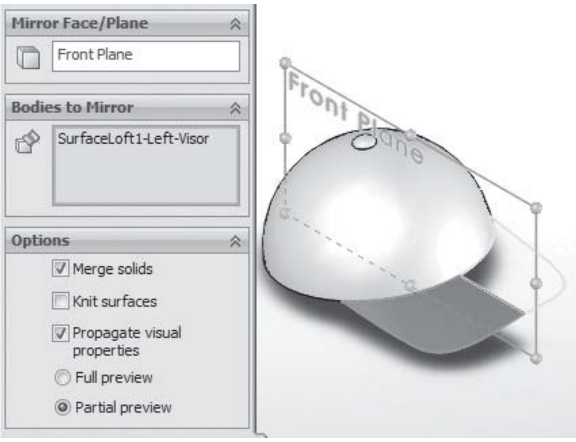
Step 10: Create visor *Sketch6*: **Front Plane** > sketch line shown, passing through the endpoints of the arc and the 3D curve > exit sketch.



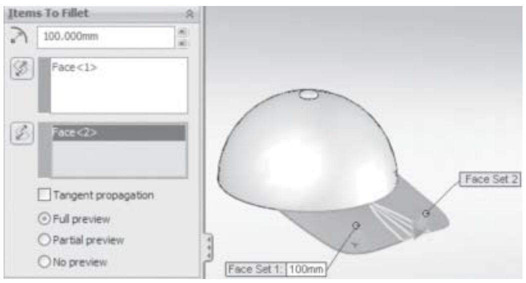
Step 11: Create *SurfaceLoft1-Left-Visor* patch: **Lofted Surface** on **Surfaces** tab > *Curve1* and *Sketch5* as **Profiles** > *Sketch6* as **Guide Curves** > ✔.



Step 12: Create *Mirror1-Right-Visor*: **Mirror** on **Features** tab > **Front Plane** as **Mirror Face/Plane** > *SurfaceLoft1-Left-Visor* as **Bodies to Mirror** > ✔.

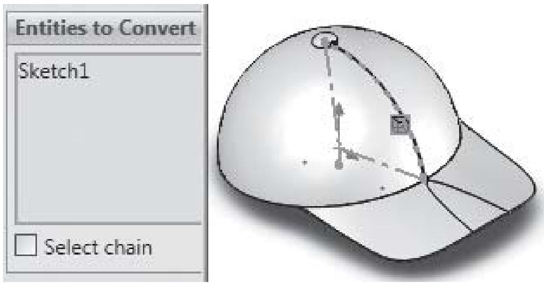


Step 13: Fillet visor two halves: **Fillet** on **Features** tab > select visor halves as faces to fillet > enter 100 for radius > ✔.

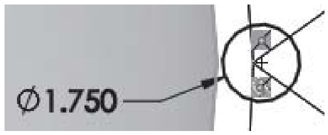


Step 14: Create *Thicken1-Visor* solid: **Thicken** on **Surfaces** tab > *Fillet1* > enter 1 for **T1** > ✔.

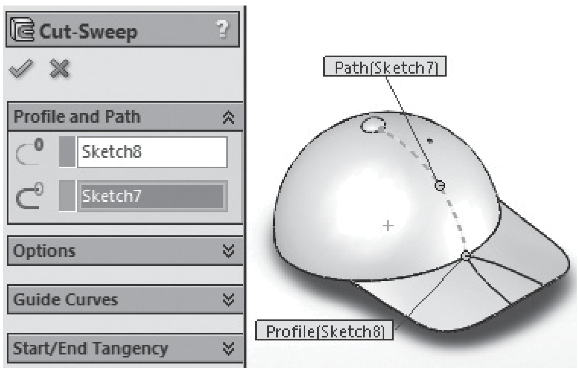
Step 15: Create *Sketch7*: **Front Plane > Convert Entities** on **Sketch** tab > expand features tree > *Sketch1* > ✔.



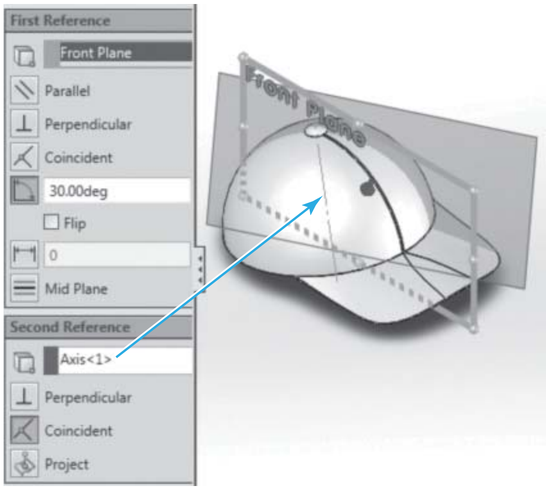
Step 16: Create *Sketch8*: Bottom face of *Revolve-Thin1-Head-Cover* feature > sketch and dimension circle shown > create an intersection relation between circle center and arc of *Sketch7* as shown > create a pierce relation between circle center and arc of *Sketch7* as shown > exit sketch.

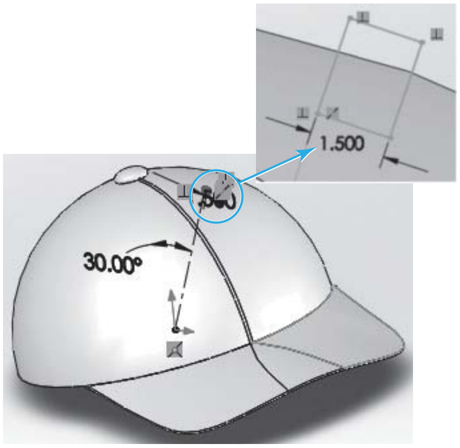


Step 17: Create *Cut-Sweep1-Groove*: Create a swept cut as shown.

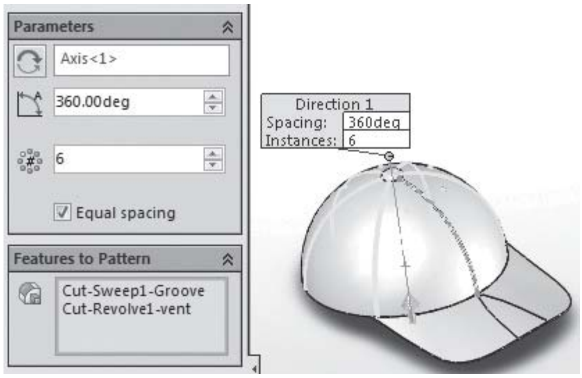


Step 18: Create *Plane3*: *Plane3* is at 30° from **Front Plane** and passes through axis of arc created in Step 1 and shown here. We use this plane to create vent sketch.





Step 19: Create *Sketch10* and *Cut-Revolve1-vent* feature: S`ketch and dimension rectangle shown above on *Plane3* > revolve cut *Sketch10* to create feature.



Step 20: Create the other grooves and vents: Use circular pattern on **Features** tab to create circular pattern groove feature (Step 17) and vent feature (Step 19) six times evenly around the central axis of the head cover.

HANDS-ON FOR TUTORIAL 9–8. Create another row of vents in the middle of the head cover

Tutorial 9–9: Create a Hair Dryer

We use 2D curves and surfaces to create the hair dryer shown in Figure 9.18. All dimensions are in inches.

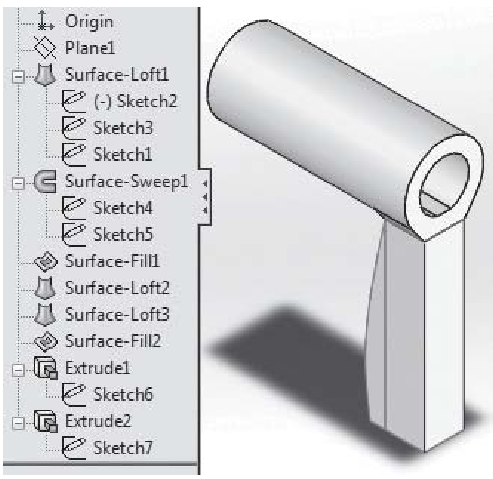
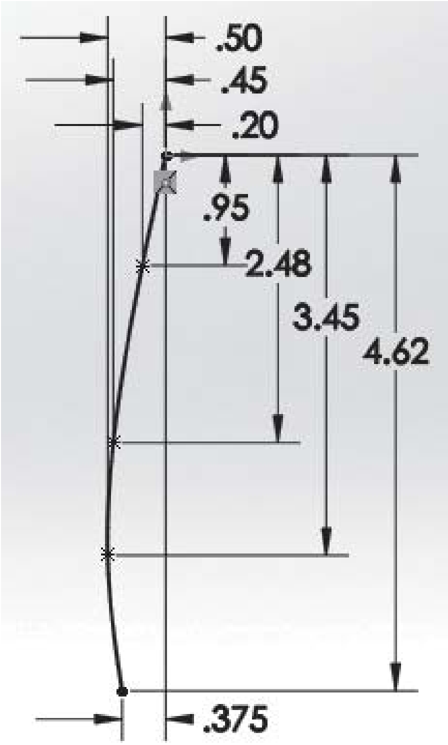


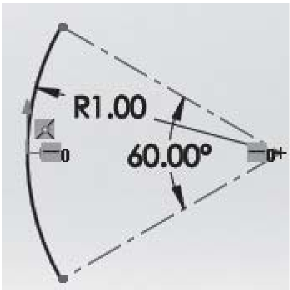
FIGURE 9.18
Hair dryer

Step 1: Create *Sketch1*: **File > New > Part > OK > Front Plane >** sketch and dimension spline shown > exit sketch > **File > Save As** > *Tutorial9.9* > **Save.**



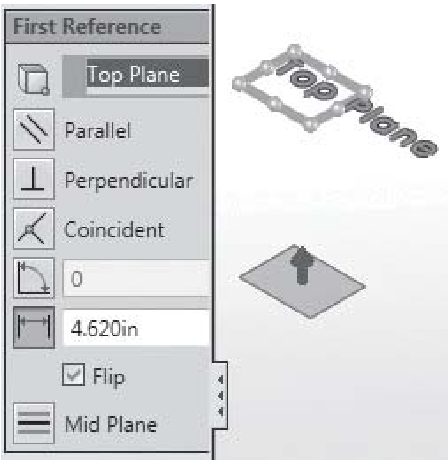
Note: Spline top endpoint snaps to origin.

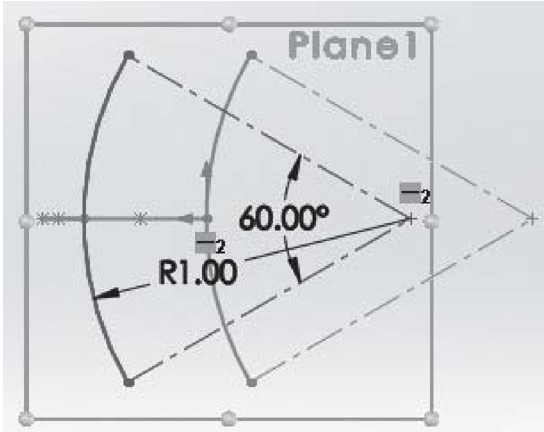
Step 2: Create *Sketch2*: **Top Plane** > sketch and dimension arc shown > exit sketch.

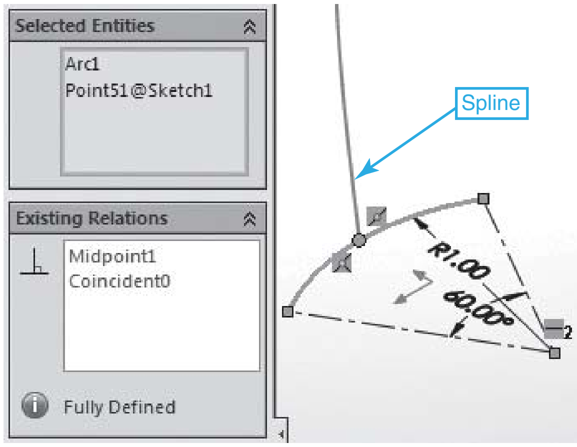


Step 3: Create *Plane1*: **Reference Geometry > Plane** > create plane defined as shown > ✔.

Note: The 4.62 distance is the largest Y coordinate of the spline shown in Step 1, so *Plane1* passes by the endpoint of the spline.

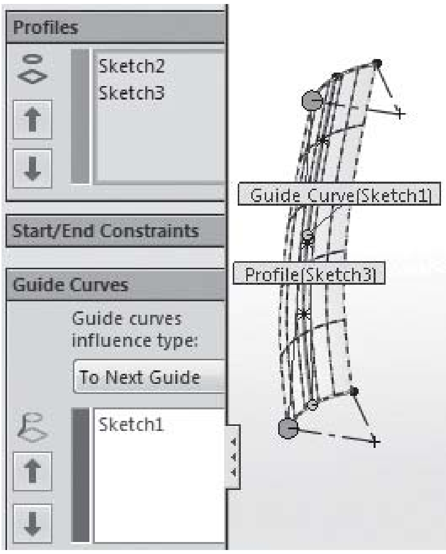




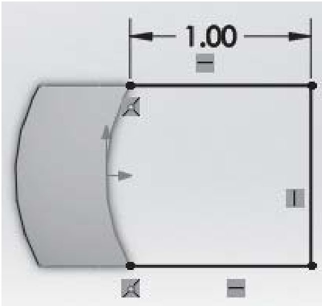


Step 4: Create *Sketch3*: *Plane1* > sketch and dimension arc shown above > select arc + **Ctrl** + spline endpoint shown > **Coincident** and **Midpoint** relations as shown above > exit sketch.

Step 5: Create *Surface-Loft1*: **Lofted Surface** on **Surfaces** tab > *Sketch2* and *Sketch3* as **Profiles** and *Sketch1* as **Guide Curves** > ✔.

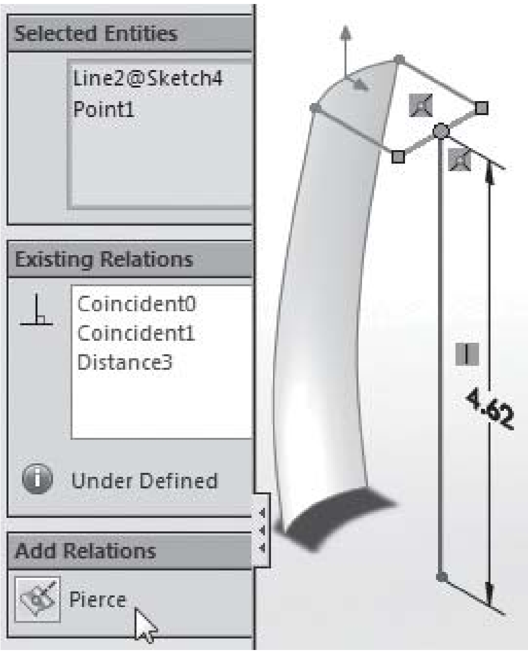


Step 6: Create *Sketch4*: **Top Plane** > sketch and dimension three lines shown > exit sketch.

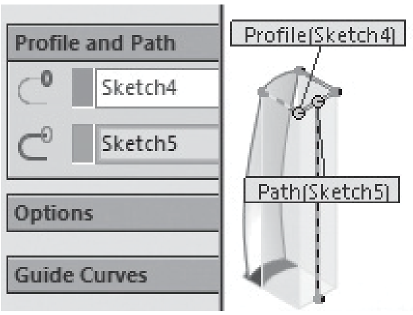


Note: Horizontal line snap to the arc ends.

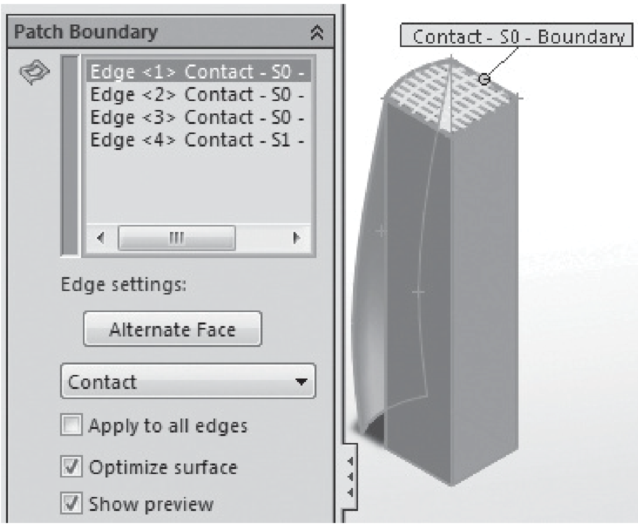
Step 7: Create *Sketch5*: **Front Plane** > sketch and dimension line shown > select top endpoint of line + **Ctrl** + top line > **Pierce** relation shown > ✔ > exit sketch.



Step 8: Create *Surface-Sweep1*: **Swept Surface** on **Surfaces** tab > *Sketch4* as **Profile** > *Sketch5* as **Path** > ✔.

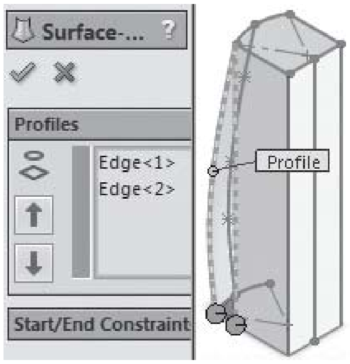


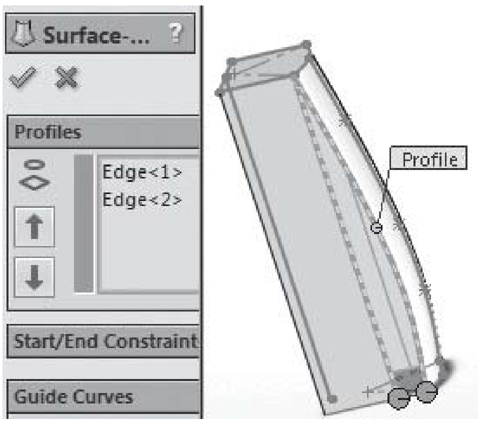
Step 9: Create handle top *Surface-Fill1*: **Filled Surface** on **Surfaces** tab > four edges shown > ✔.



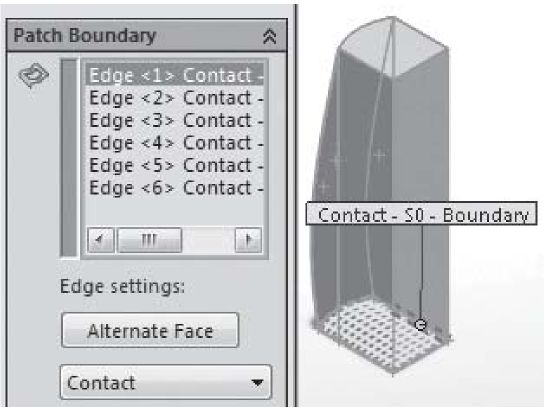
Note: The spike shown is caused by the viewing angle.

Step 10: Create *Surface-Loft2* and *Surface-Loft3*: **Lofted Surface** on **Surfaces** tab > select two edges shown to create one surface> ✔ > repeat as shown to create other surface > ✔.



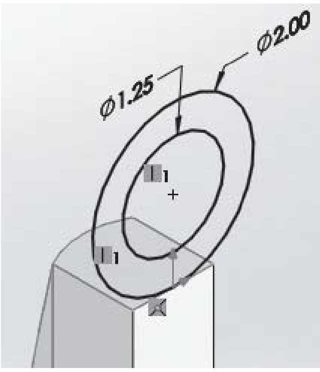


Step 11: Create *Surface-Fill2*: **Filled Surface** on **Surfaces** tab > select six edges shown in order > ✔

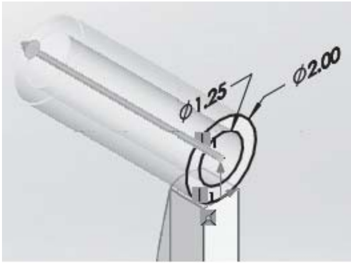


Note: There are 6 edges, not 4, because two sides have 2 edges each, not 1, based on how we created surfaces.

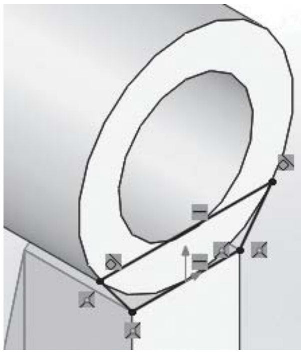
Step 12: Create *Sketch6* and *Extrude1* feature: back face of handle as sketch plane > sketch and dimension two circles shown here > exit sketch > **Extruded**



Boss/Base on **Features** tab > enter thickness of **5.0** in. for **D1** > ✔.



Step 13: Create *Sketch7* and *Extrude2* feature: Back face of handle as sketch plane > sketch lines with all relations shown > exit sketch > **Extruded Boss/Base** on **Features** tab > enter thickness of **1.0** in. for **D1** as shown > ✔.



HANDS-ON FOR TUTORIAL 9–9. Create a solid using the surfaces created in this tutorial. Fillet the edges of the resulting solid. Also, create a grill for the back opening of the barrel where the handle is.

Tutorial 9–10: Create an Oil Container

We use surfaces to create a model of the motor oil container shown in Figure 9.19. The motor oil container is physically made out of one piece of plastic like many other containers. The modeling plan is to create two surface extrudes and then a lofted surface between them to create the container neck. Close the bottom of the bottle with a filled surface. Finally knit the surfaces together and use one thicken function to create a solid container. All dimensions are in inches.

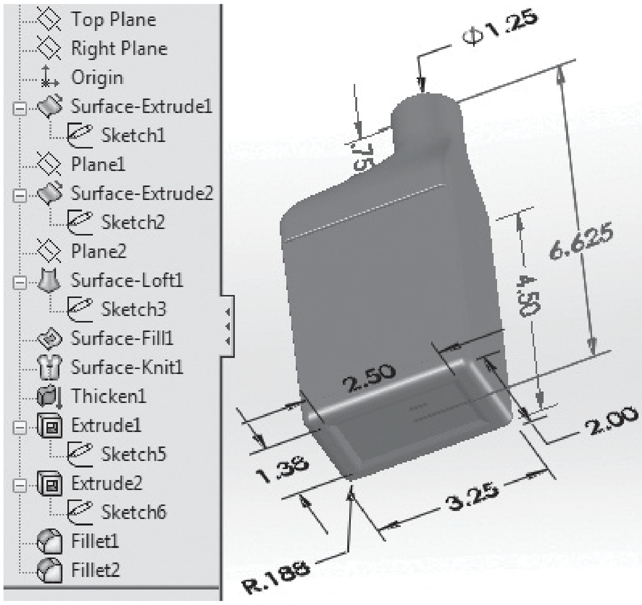
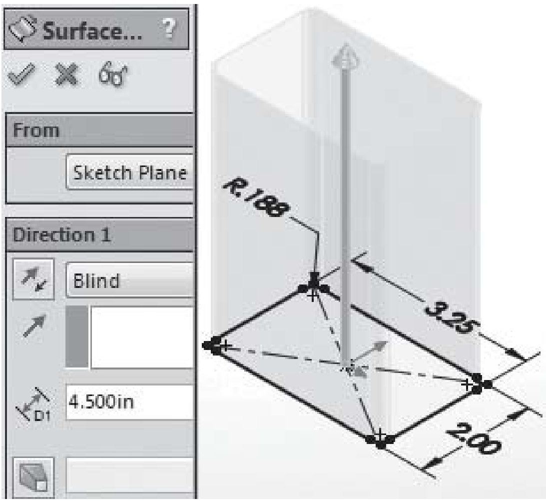
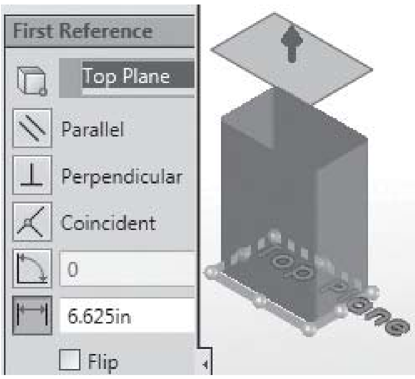


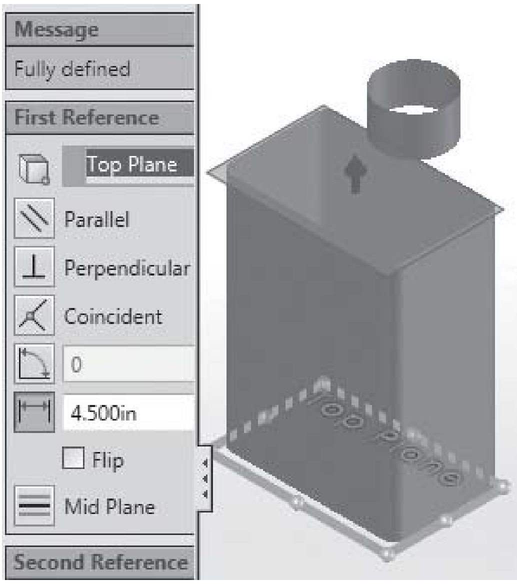
FIGURE 9.19
Oil container

Step 1: Create *Sketch1* and *Surface-Extrude1*: **File** > **New** > **Part** > **OK** > **Top Plane** > sketch a center rectangle at origin, fillet and dimension it > exit sketch > **Extruded Surface** on **Surfaces** tab > enter 4.5 for thickness **D1** > ✔.

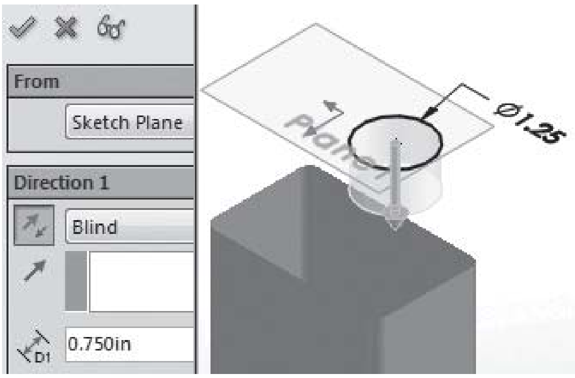


Step 2: Create *Plane1* and *Plane2*: Create a reference plane 6.625 in. away from **Top Plane** for *Plane1* as shown to the right and 4.5 in. away for *Plane2* as shown at the top of the next columns.

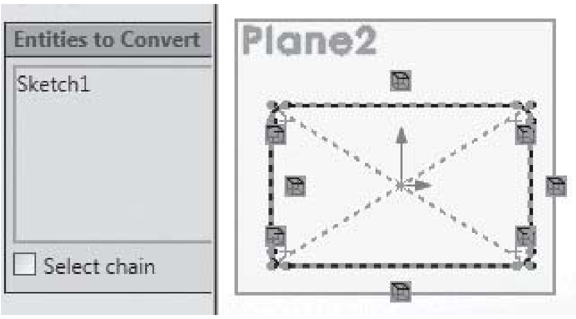




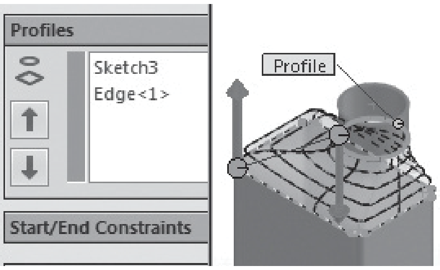
Step 3: Create spout cross section *Sketch2* and *Surface-Extrude2*: *Plane1* as sketch plane > sketch and dimension circle shown > exit sketch > **Extruded Surface** on **Surfaces** tab > enter **0.75** for thickness **D1** > ✔.



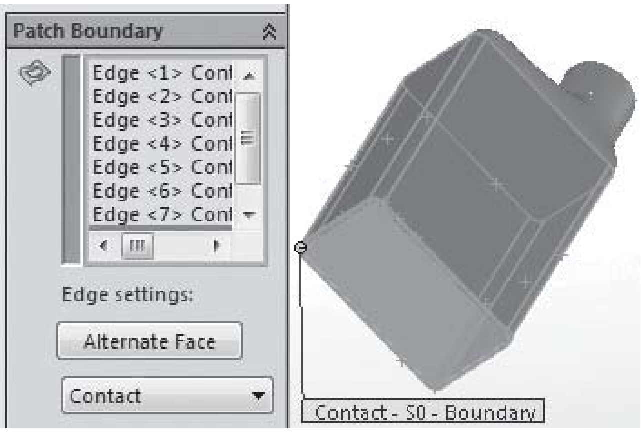
Step 4: Create *Sketch3*: *Plane2* as sketch plane > **Convert Entities** on **Sketch** tab > expand features tree > *Sketch1* > ✔.



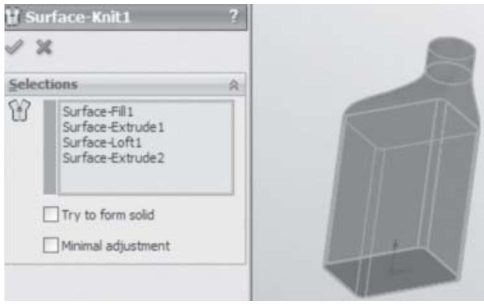
Step 5: Create neck *Surface-Loft1*: **Lofted Surface** on **Surfaces** tab > *Sketch3* > lower circular edge shown > ✔.



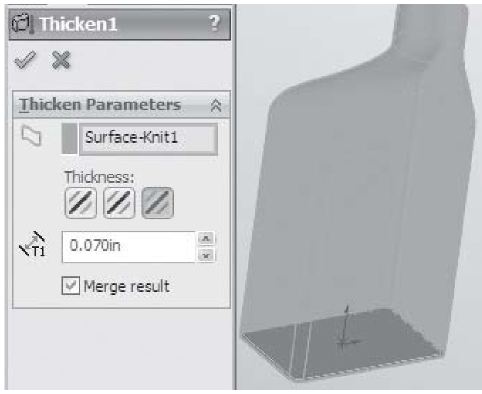
Step 6: Create bottom *Surface-Fill1*: **Filled Surface** on **Surfaces** tab > select 8 bottom edges (4 lines and 4 fillets) in order > ✔.



Step 7: Create knit surface *Surface-Knit1*: **Knit Surface** on **Surfaces** tab > select all surfaces created so far and shown > ✔.



Step 8: Create *Thicken1* solid: **Thicken** on **Surfaces** tab > *Surface-Knit1* > enter 0.07 for thickness **T1** > ✔.



Note: **Thicken** has a direction relative to the surface sheet: inside of surface, outside of surface, or on both sides of surface. The side selection affects the volume of the container.

HANDS-ON FOR TUTORIAL 9–10.

(a) Create the two side cuts and fillets in the container to make it look more realistic as shown in Figure 9.19. Also, use colors to render the model.

(b) Create the threads around the spout of the container. Assume necessary dimensions to model the threads.

problems

1. If a parametric surface has u limits of u_{min} and u_{max} and v limits of v_{min} and v_{max}, what are the u and v values of its midpoint?
2. Find the parametric equation of the plane connecting point (2, 1, 0) to point (−2, −5, 0), and point (2, 1, 0) to point (0, 3, −2). Find the plane midpoint and its vectors. Manually on a piece of paper, sketch the plane and show the data points and the parameterization direction on the sketch.
3. For the plane of Problem 2, find the coordinates of the point located at $u = 0.25$ from the $u = 1$ end and at $v = 0.25$ from the $v = 1$ end.
4. Find the parametric equation of the plane connecting point (1, 3, 7) to point (−2, −4, −6), and point (1, 3, 7) to point (5, 0, −8). Find the plane midpoint and its vectors. Manually on a piece of paper, sketch the plane, and show the data points and the parameterization direction on the sketch.
5. Two splines are given by:

$$P(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} 1 + 4u - u^2 \\ 2 - u + u^2 \end{bmatrix} \quad 0 \leq u \leq 1$$

and

$$P(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} -2 - 2u^2 \\ -1 + u - 2u^2 \end{bmatrix} \quad 0 \leq u \leq 1$$

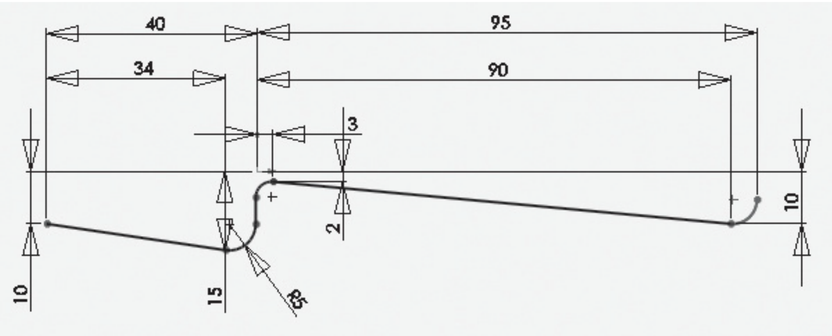
Find the equation of the ruled surface connecting the two splines. Find the corner points and the midpoint of the surface. Also, find the surface vectors at the midpoint.
6. Use your CAD system to create the lofted surface connecting the two splines of Problem 5. Find the coordinates of the four corner points of the surface using the **Measure** tool of the CAD system. Verify the results using the parametric equations of the splines.
7. A spline is given by:

$$P(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix} = \begin{bmatrix} 2u \\ 2 - u \end{bmatrix} \quad 0 \leq u \leq 1$$

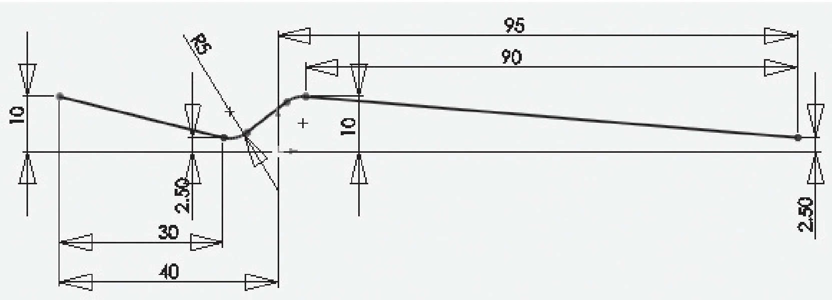
Use this spline together with a line connecting points (2, 5) and (4, 6), and find the equation of the ruled surface that connects the spline and the line.
8. Using Problem 17 in Chapter 8, create a surface that connects the two edges of the laboratory chair. Thicken the surface by a value of 15 mm to create the chair solid.
9. Using Problem 18 in Chapter 8, create the skateboard surface whose boundary is defined by the 3D curve. Thicken the surface by 0.5 in. to create the skateboard solid.
10. Using Problem 19 in Chapter 8, create the bicycle surface whose boundary is defined by the 3D curve. Thicken the surface by 0.5 in. to create the bicycle helmet solid.
11. Using Problem 20 in Chapter 8, create the surface whose boundary is defined by the 3D curve. Thicken the surface by 0.5 in. to create the weed whacker debris shield.

12. Using Problem 21 in Chapter 8, the surface whose boundary is defined by the 3D curve. Thicken the surface by 0.5 in. to create the S-shaped chair.
13. Using Problem 22 in Chapter 8, the surface whose boundary is defined by the 3D curve. Thicken the surface by 0.5 in. to create the bike seat.
14. Figure 9.20 shows the model of a table fork. The model requires 3D curves and surfaces. The figure shows the two projections of the 3D curve that defines the fork profile. Thicken the surface to create the fork. All dimensions are in mm.

(A) Fork



(B) **Top Plane** sketch of fork half profile



(C) **Front Plane** sketch of fork half profile

FIGURE 9.20
Table fork model

15. Figure 9.21 shows an elliptic paraboloid surface. Its equation is given by:

$$\frac{x^2}{A^2} + \frac{y^2}{B^2} = \frac{z}{h}$$

Use $A = 2$, $B = 3$, and $h = 4$. The surface becomes a curve for a fixed z value. Using $z = 4$, use the explicit equation to create the curve in your CAD system. Then, revolve it to create the surface. Cut the surface by sections at $z = 0, 2, 4$. Insert key points at each section and measure their coordinates. Verify CAD/CAM system results against the equation. Is there an error? If yes, what is the error percentage?

Note: This problem introduces an important concept in CAD design. Sometimes, a designer may run an engineering analysis that produces surface equations that must be included in part design for better performances (e.g., lift and drag, or aerodynamics of car surfaces).

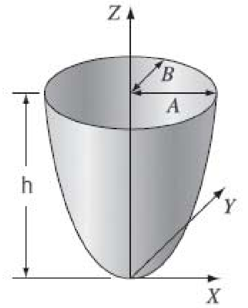


FIGURE 9.21
Elliptic paraboloid

16. Figure 9.22 shows the model of a soda bottle. Create the model. The bottle bottom has five identical legs. A leg would require creating different splines on different sketch planes (at angles from each other) and then creating a loft surface between them. Create one and pattern it. After patterning, fill the gaps with loft surfaces. Create the other surfaces and then thicken all the surfaces to create the bottle model. All dimensions are in mm.

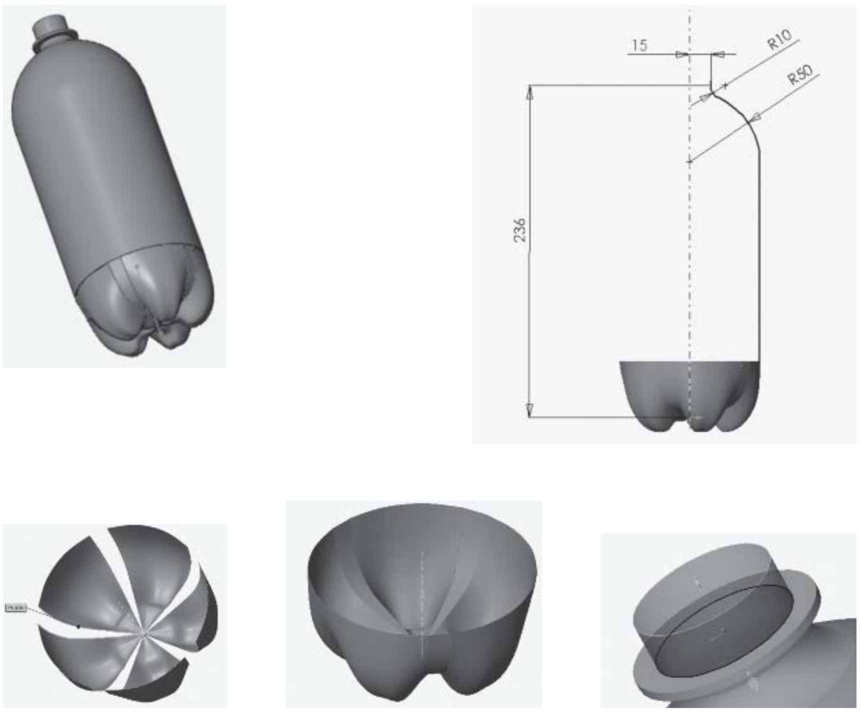


FIGURE 9.22
Soda bottle model

17. Find your favorite rock or other object that requires 3D curves and surfaces to model as a solid. Create the solid model of the rock or the object. Figure 9.23 shows an example.

FIGURE 9.23
A rock

18. Create the earphone model shown in Figure 9.24. Use your own earphone to find dimensions. Figure 9.24 shows the features tree for some guidance. This is one part, not an assembly.

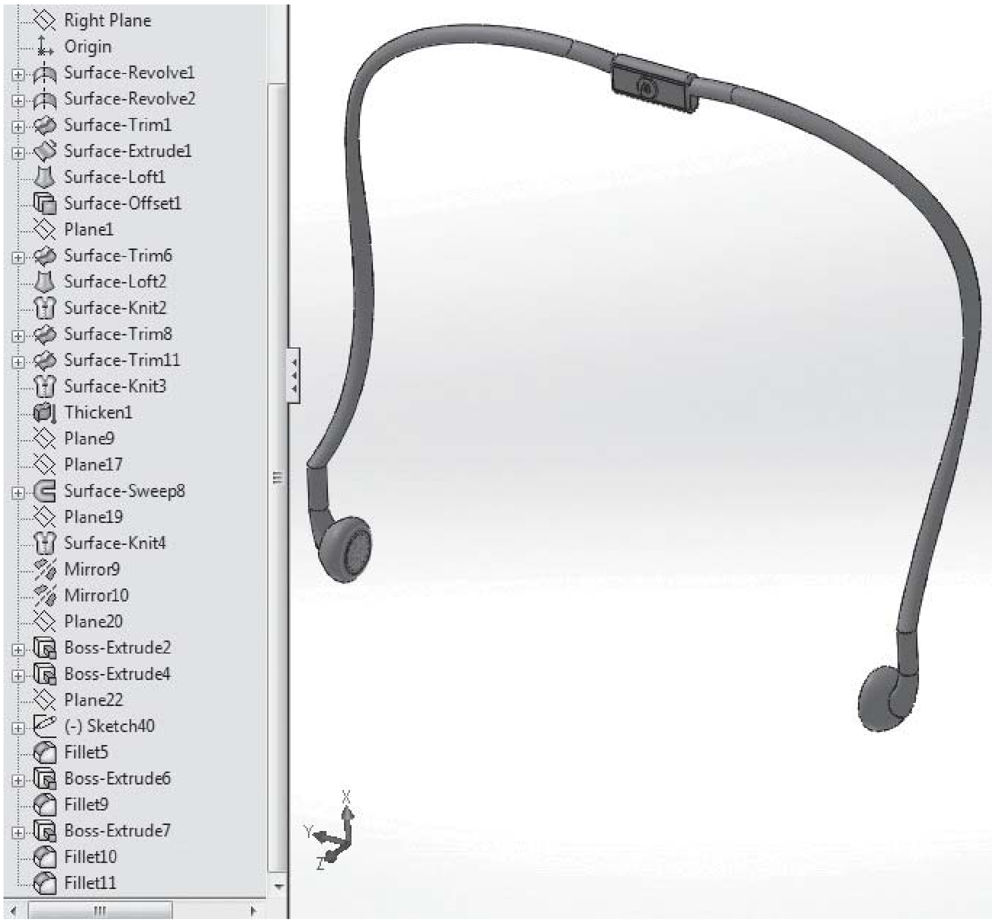


FIGURE 9.24
Earphones

19. Create the grinder model shown in Figure 9.25. Assume all dimensions. Figure 9.25 shows the features tree for some guidance. This is one part, not an assembly.

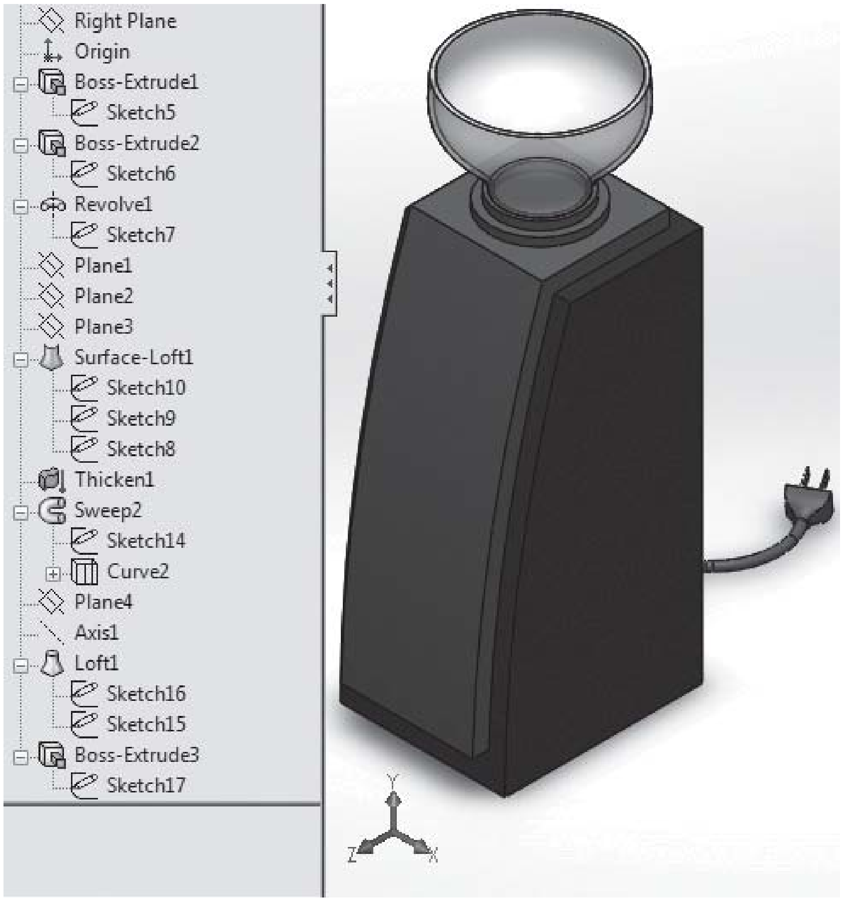


FIGURE 9.25
Grinder

20. Create the vase model shown in Figure 9.26. Use your own vase to find dimensions. Figure 9.26 shows the features tree for some guidance. This is one part, not an assembly.

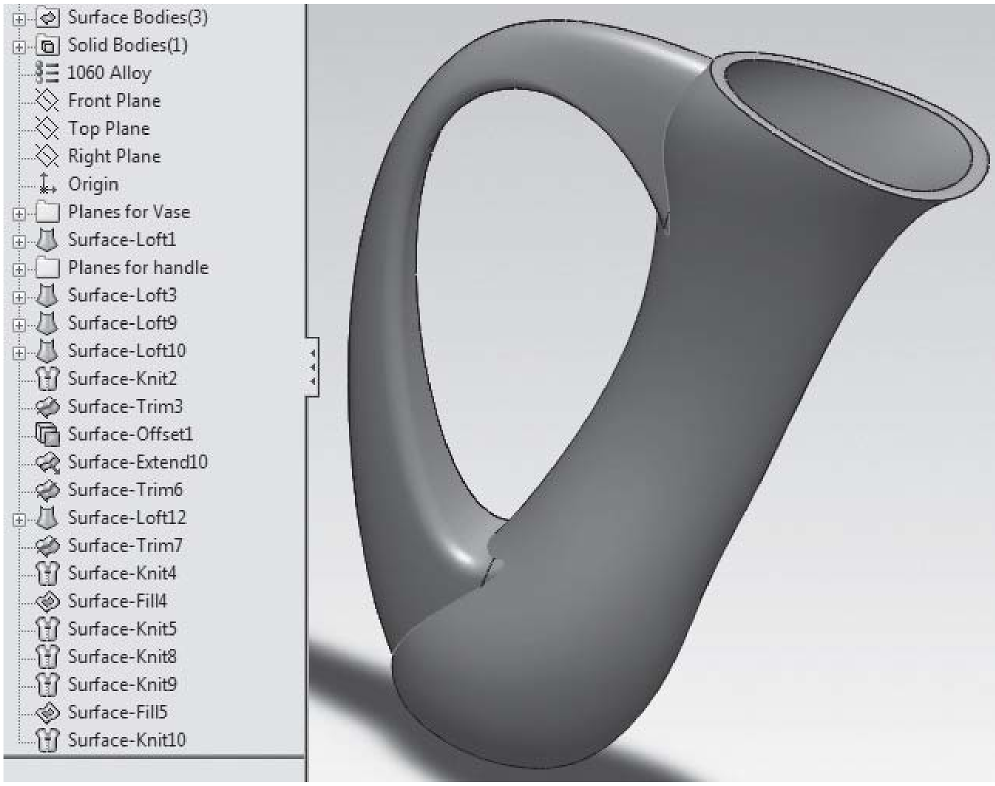


FIGURE 9.26
Vase

21. Create the racing wheel model shown in Figure 9.27. Assume dimensions. Figure 9.27 shows the features tree for some guidance. This is one part, not an assembly.

FIGURE 9.27
Racing wheel

22. Create the Crocs™ shoe model shown in Figure 9.28. Assume dimensions. Figure 9.28 shows the features tree for some guidance. This is one part, not an assembly.

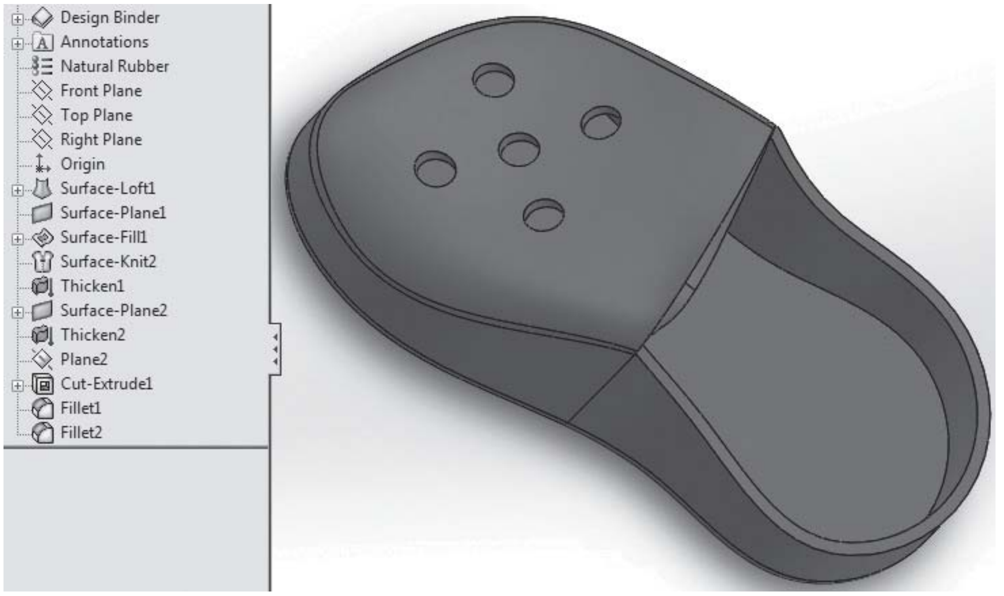


FIGURE 9.28
Crocs shoe

23. Create the jawbone model shown in Figure 9.29. Assume dimensions. Figure 9.29 shows the features tree for some guidance. This is one part, not an assembly.

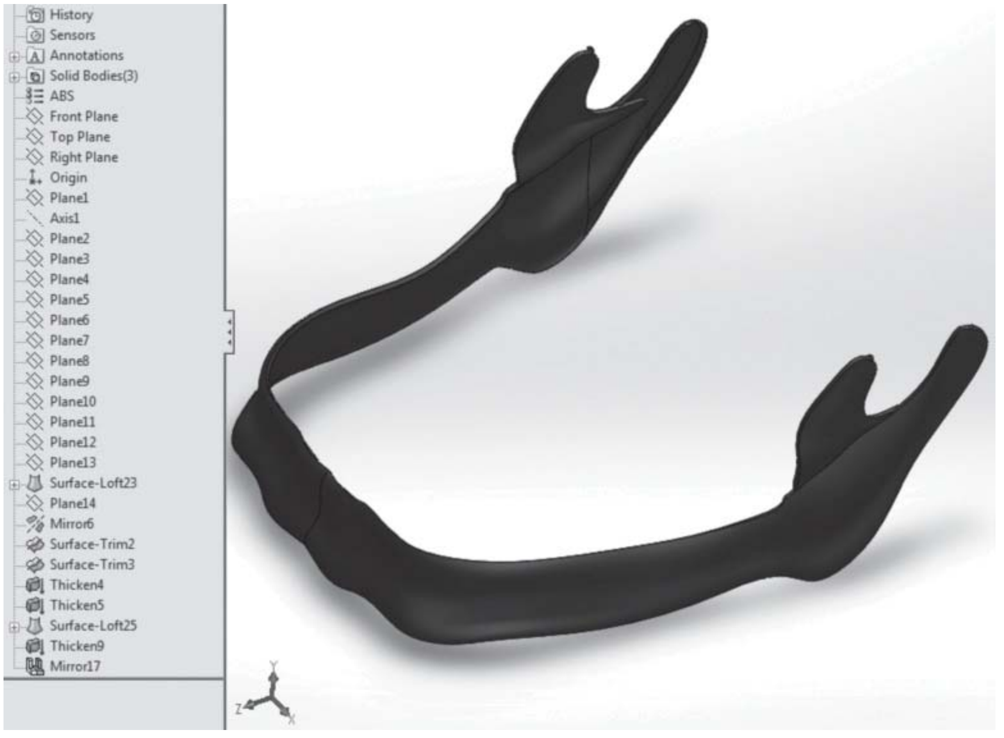


FIGURE 9.29
Jawbone

CHAPTER

10

Sheet Metal and Weldments

10.1 Introduction

The previous chapters have focused heavily on presenting and harnessing the power of features, curves (especially 3D curves), and surfaces to create solids. As a matter of fact, there are no parts that cannot be created using these modeling tools. However, this chapter covers the basics of one class of parts that, although it can be modeled using these tools, is created more efficiently via other specialized modeling techniques. This class is sheet metal parts and weldments (welded parts). These parts are used heavily in industry, and CAD/CAM systems provide special modeling techniques to create them efficiently. For example, a weldment is created as a steel or welded frame or structure instead of as an assembly. Even though they are not related, we cover both sheet metal and weldments in one chapter because they belong to this special class.

10.2 Sheet Metal

Sheet metal is thin, flat pieces of metal that come in different sizes that can be cut and bent into a variety of shapes. The thickness of sheet metals can vary significantly, from 1 mm to 6 mm (0.25 inch). An extremely thin piece of metal is considered a foil or leaf (e.g., kitchen aluminum foil), and pieces thicker than 6 mm are considered plates. Sheet metal is a fundamental form used in metalworking. Many parts of objects around us are made from sheet metal. Examples include airplane wings, car bodies, medical tables, roofs for buildings, brackets, chases, enclosures, panels, channels, hinges, razor blades, and signage. Industries that use sheet metal include audio-video, electronics, fiber optics, medical, security, semiconductors, and telecommunications.

Sheet metal processes and equipment include punching (via punch press), rolling, embossing, stamping, breaking, notching, shearing, spot welding, insertion, and finishing. Machining tolerance could be as small as 0.0005 in. Commonly used materials include aluminum, steel, brass, copper, silver, gold, nickel, platinum, and titanium. NC (numerical control) programming may be used to program sheet metal fabrication.

Sheet metal comes as flat sheets or coiled strips. The coils are formed by running flat sheets through a roll slitter. The thickness of a sheet metal is known as its gauge. Gauges range from 30 gauge to 8 gauge. The higher the gauge, the thinner the sheet.

If you need a sheet metal part as an enclosure, you can design it in two ways: either on its own without any references to the parts it will enclose, or in the context of the assembly that contains the enclosed components.

One common operation of sheet metal is bending a sheet. Sheet metal bending is also known as a brake. The machine that bends sheet metal is known as a bending brake or bending machine. The machine can create simple bends and creases, or create box and pan shapes. Bending changes the shape of the metal by plastically deforming it. When a sheet metal is bent, the inside surface of the bend is compressed, and the outer surface of the bend is stretched. Somewhere within the metal thickness lies its neutral axis (NA). An NA is a line or layer in the metal that is free from any forces; that is, it is neither compressed nor stretched. Thus, the length of the metal remains the same along the NA, before and after bending.

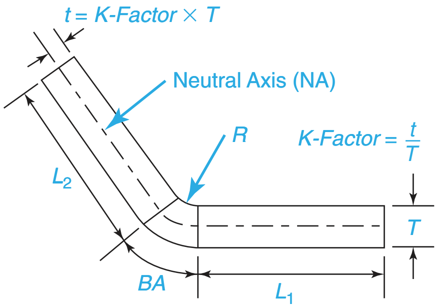


FIGURE 10.1
Bending a sheet metal

The location of the NA is referred to as the K-Factor of the sheet metal; that is, the K-Factor is used to calculate the location of the neutral axis. The K-Factor is important in determining the flat length of the sheet metal that is required to produce certain bends. Consider the example shown in Figure 10.1. We want a sheet metal workpiece with a bend angle A and a bend radius R, in which one leg measures L_1, and the other measures L_2. The NA is located at thickness t from the inner surface, and the metal total thickness is T. It is obvious from Figure 10.1 that the total length of the flat piece that we should bend is not $L_1 + L_2$, as we might first assume. We need to add the bend part, BA, for the bend angle A. This part is known as the bend allowance (BA), which we must add to the leg lengths to get the final bend shape that we want. Thus, the flat length, L, of the sheet metal is given by:

$$L = L_1 + L_2 + BA \tag{10.1}$$

The location of the NA varies depending on the material itself, the bend radius, R, the ambient temperature, the direction of the material grain, and the method used for bending. The **K-Factor** is the ratio of the location of the NA as measured by the thickness, t, shown in Figure 10.1 with respect to the thickness, T, of the sheet metal part; that is,

$$\text{K-Factor} = \frac{t}{T} \tag{10.2}$$

The K-Factor is usually calculated by trial and error. The K-Factor is typically between 0.3 and 0.5. For most types of steel it is between 0.33 and 0.4.

The bend allowance, BA, is given by the following arc equation:

$$BA = r\theta = (R + t)\frac{\pi A}{180} = (R + KT)\frac{\pi A}{180} = \frac{\pi A(R + KT)}{180} \tag{10.3}$$

where A is in degrees.

Substituting Eq. (10.3) into Eq. (10.1), we get the sheet metal flat length as:

$$L = L_1 + L_2 + \frac{\pi A(R + KT)}{180} \tag{10.4}$$

Sometimes, we let the lengths of the two legs, L_1 and L_2, meet (intersect), and overtake the arc length BA. In this case, the total length, $L_1 + L_2$, exceeds the length of the flat sheet by an amount called the *bend deduction* (BD); that is,

$$BD = L_1 + L_2 - \text{flat length} \tag{10.5}$$

While it may seem logical that BA and BD are equal, they are not. We will not get into any more details. Interested readers should consult the literature. There are BA tables that are separate and different from BD tables. CAD/CAM systems provide tables for K-Factor, BA, and BD for most commonly used materials. SolidWorks has these tables in this folder: *C:\Program Files\SolidWorks Corp\SolidWorks\lang\english\Sheetmetal Bend Tables*. It also has gauge tables in another folder: *C:\Program Files\SolidWorks\SolidWorks\lang\english\Sheet Metal Gauge Tables*.

10.3 Sheet Metal Features

The best way to understand sheet metal modeling and design is to ask a simple question: How different are sheet metal parts from ordinary parts? From the outset one could trivialize modeling of sheet metal parts by recognizing them as simple extrusions connected together. (This would obviously be the wrong view of modeling and designing sheet metal parts.) We now answer the question. The modeling of sheet metal parts has two unique aspects. First, the sheet metal parameters covered in Section 10.2 must be modeled correctly to ensure proper manufacturing (the gauge, the K-Factor, and the bend angles.). Second, we need to calculate the length of the flat sheet that we must use to create a final shape; we must be able to flatten the sheet metal part that we model (final model) using the K-Factor.

FIGURE 10.2
Sheet metal features

In addition to these two aspects, sheet metal modeling should provide features that are unique to sheet metal such as bends, flanges, corners, and others. These features must be easy to create and incorporate the concepts covered in Section 10.2. SolidWorks provides a comprehensive and easy-to-use sheet metal module with ample features. Figure 10.2 shows these features.

The main sheet metal feature is a flange. Think of a flange as an extrusion. Three types of flanges exist: base, edge, and miter. A **base flange** is the first flange you create in a sheet metal part. An **edge flange** is one that is built off an existing edge on a base flange. A **miter flange** is a flange that connects two panels (sides) of a sheet metal cut at a 45° angle.

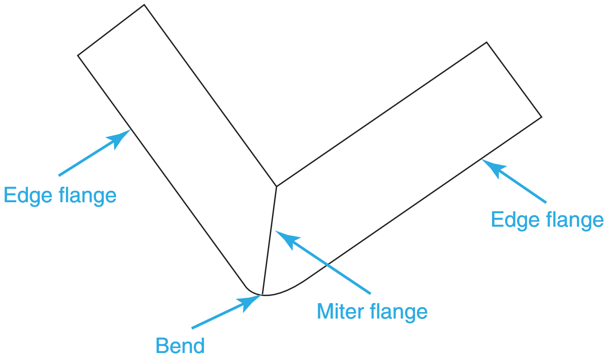


FIGURE 10.3
Sheet metal flanges

Figure 10.3 shows the types of sheet metal flanges. Other sheet metal features include a tab, a bend, a jog, a corner, a rip, a hem, and a lofted bend. A **tab** is a subflange that you can bend. For example, you can use an edge flange to build a tab and then bend it. The first time you click the **Base Flange/Tab** icon shown in Figure 10.2, you create a base flange. For subsequent clicks, you create a tab each time you click it because you can have only one base feature in a part.

A **bend** is used to bend an existing flange about a bend line. The bend is labeled **Sketched Bend** in Figure 10.2. A **jog** is used to add two bends to an existing flange about a bend line. Another way to think about it is that a part (defined by the bend line) of an existing flange is offset (jogged) up or down, thus creating two bends. A **corner** is used to treat a bend area, that is, weld at the bend corner to stiffen the sheet metal or break/trim the corner. A **rip** creates a gap between two edges. A **hem** curls an edge on itself. A **lofted-bend** is a transition from one shape to another. Think of it as a bend between different shapes. Table 10.1 shows all these sheet metal features and how to create them.

In addition to the sheet metal features, the interface shown in Figure 10.2 offers two groups. One group has two typical features: **Extruded Cut** and **Simple Hole**. They are offered for convenience; otherwise, you have to go to the **Features** tab (Figure 10.2) to access them. Why only these two features? Because they are the most commonly used in the context of sheet metal design. The most common tasks you do for sheet metal is cut it, bend it, make cuts in it, and/or make holes in it. The other group has functions related to manufacturing: **Unfold**, **Fold**, and **Flatten**. Flattening is particularly useful because it allows you to know the exact dimensions of a sheet metal stock that you need to manufacture the final product. You need the flattening because of the K-Factor effect.

TABLE 10.1 Available Sheet Metal Features

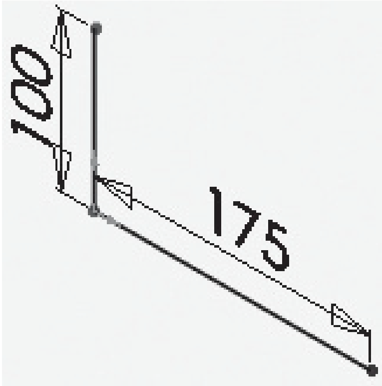
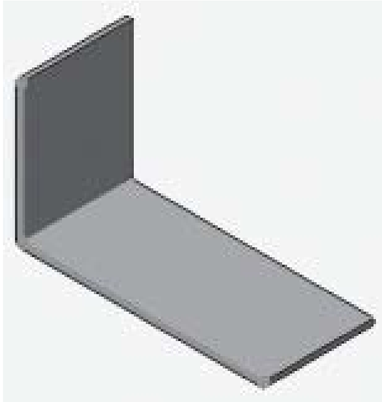
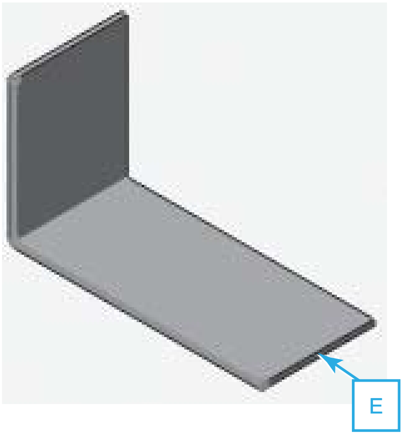
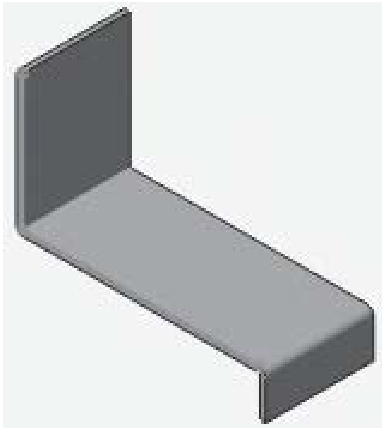
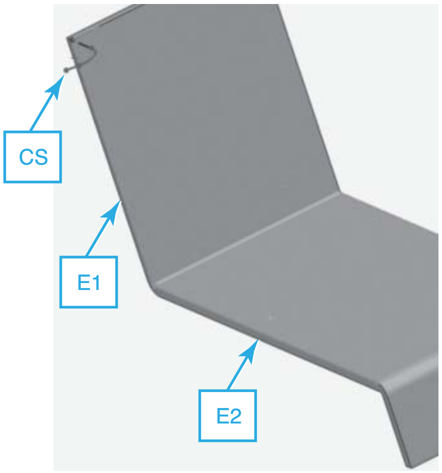
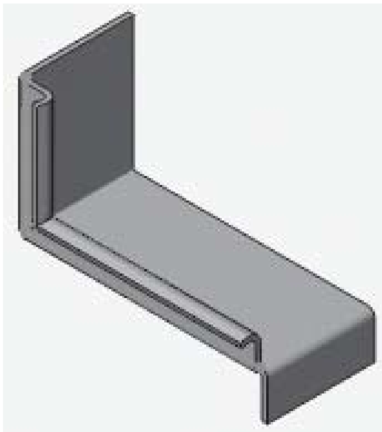
Feature	Input	Before feature creation	After feature creation
Base flange	Flange sketch		
Edge flange	Edge to flange (E), flange angle (90°), flange length (30), flange position (bend outside)		
Miter flange	Flange cross section (CS) and edges to flange. CS must be perpendicular to edges and can only be lines and/or arcs, no splines.		

TABLE 10.1 Available Sheet Metal Features (*continued*)

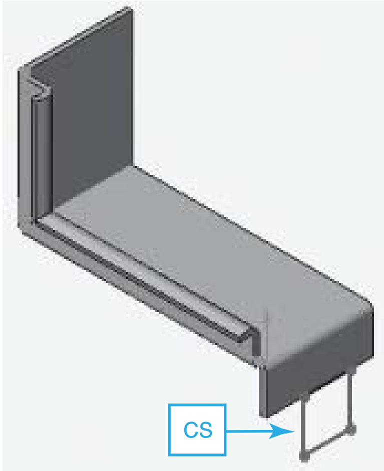
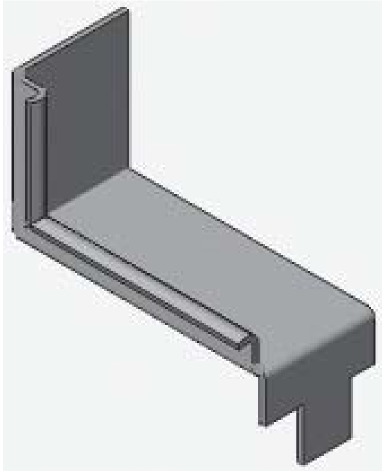
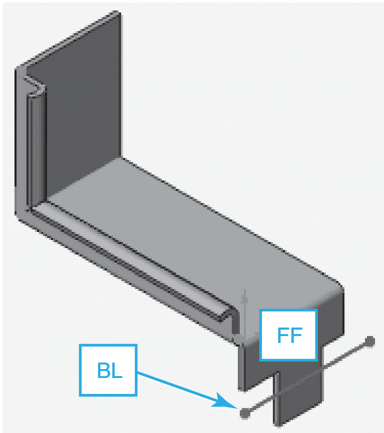
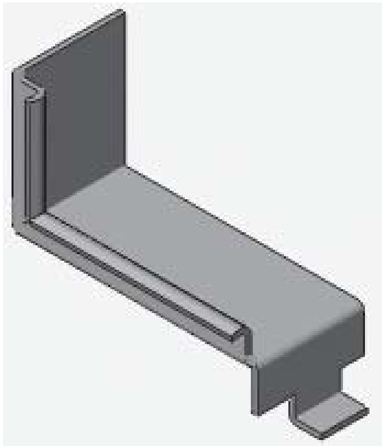
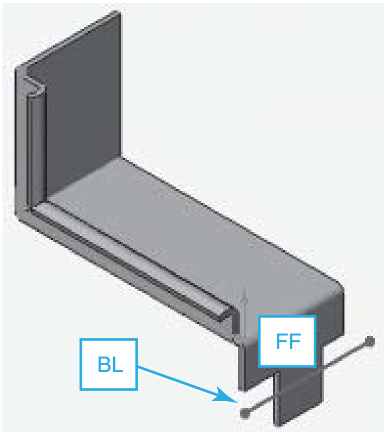
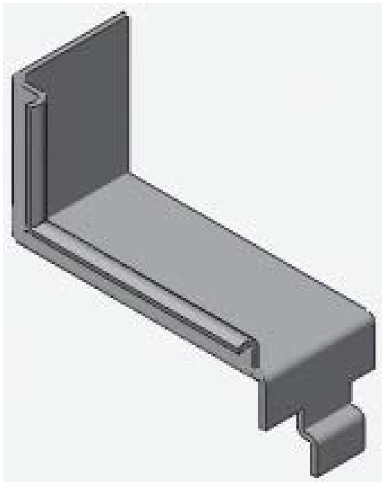
Feature	Input	Before feature creation	After feature creation
Tab	A sketch of the cross section (CS) of the tab		
Bend (sketched Bend)	A bend line (BL) and fixed face (FF) to bend with respect to		
Jog	A bend line (BL) and fixed face (FF) to bend with respect to **Hint:** The difference between bend and jog is that the latter creates two bends instead of one, thus the effect of jog.		

TABLE 10.1 Available Sheet Metal Features (*continued*)

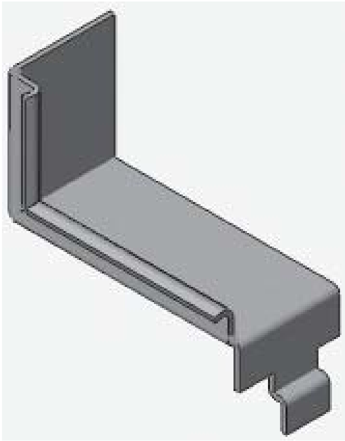
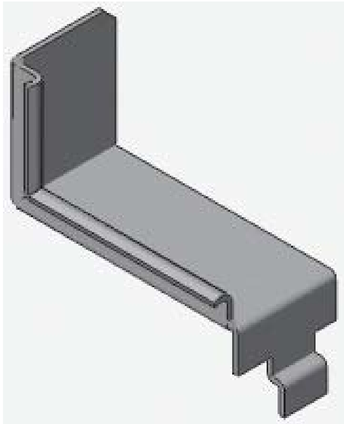
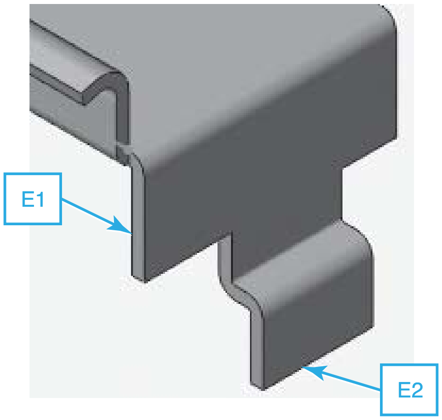
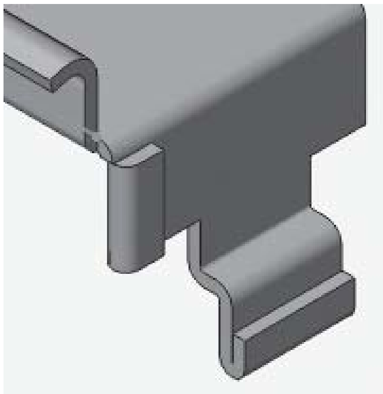
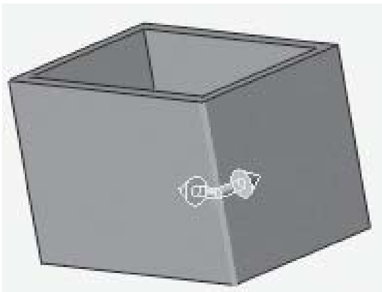
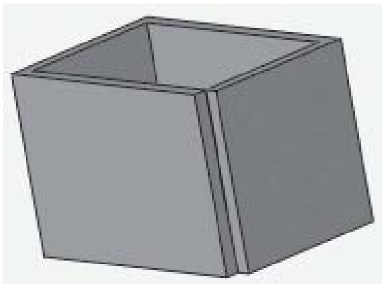
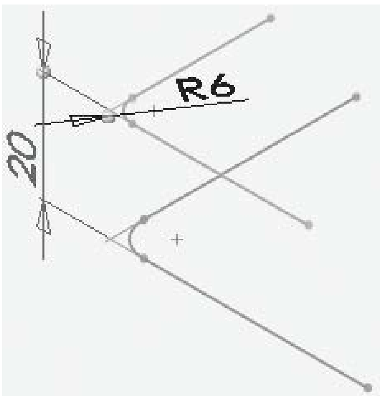
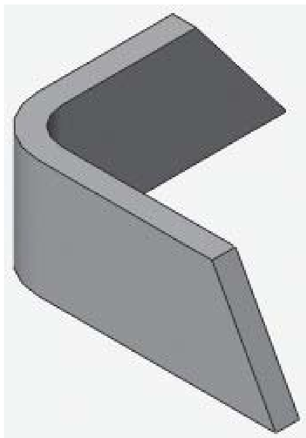
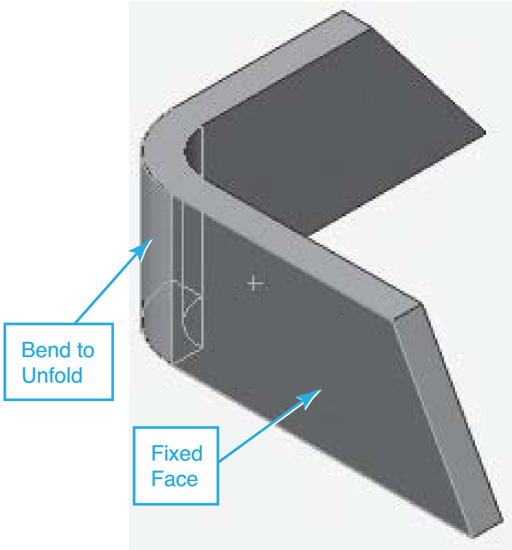
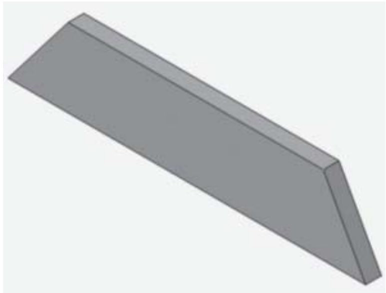
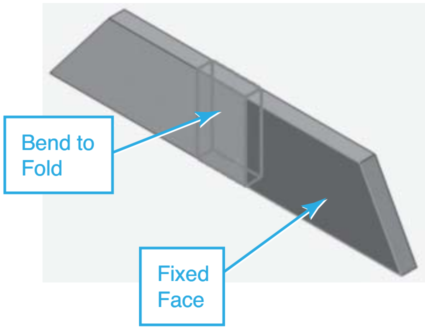
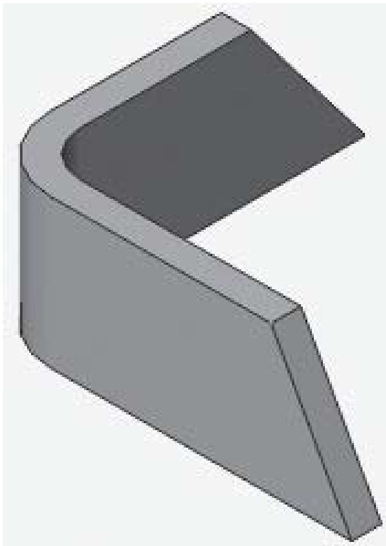
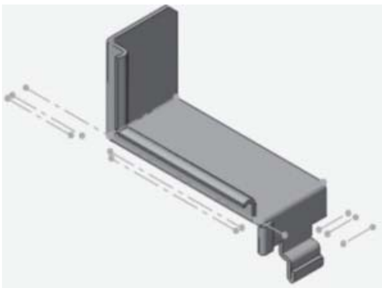
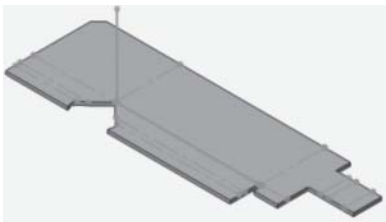
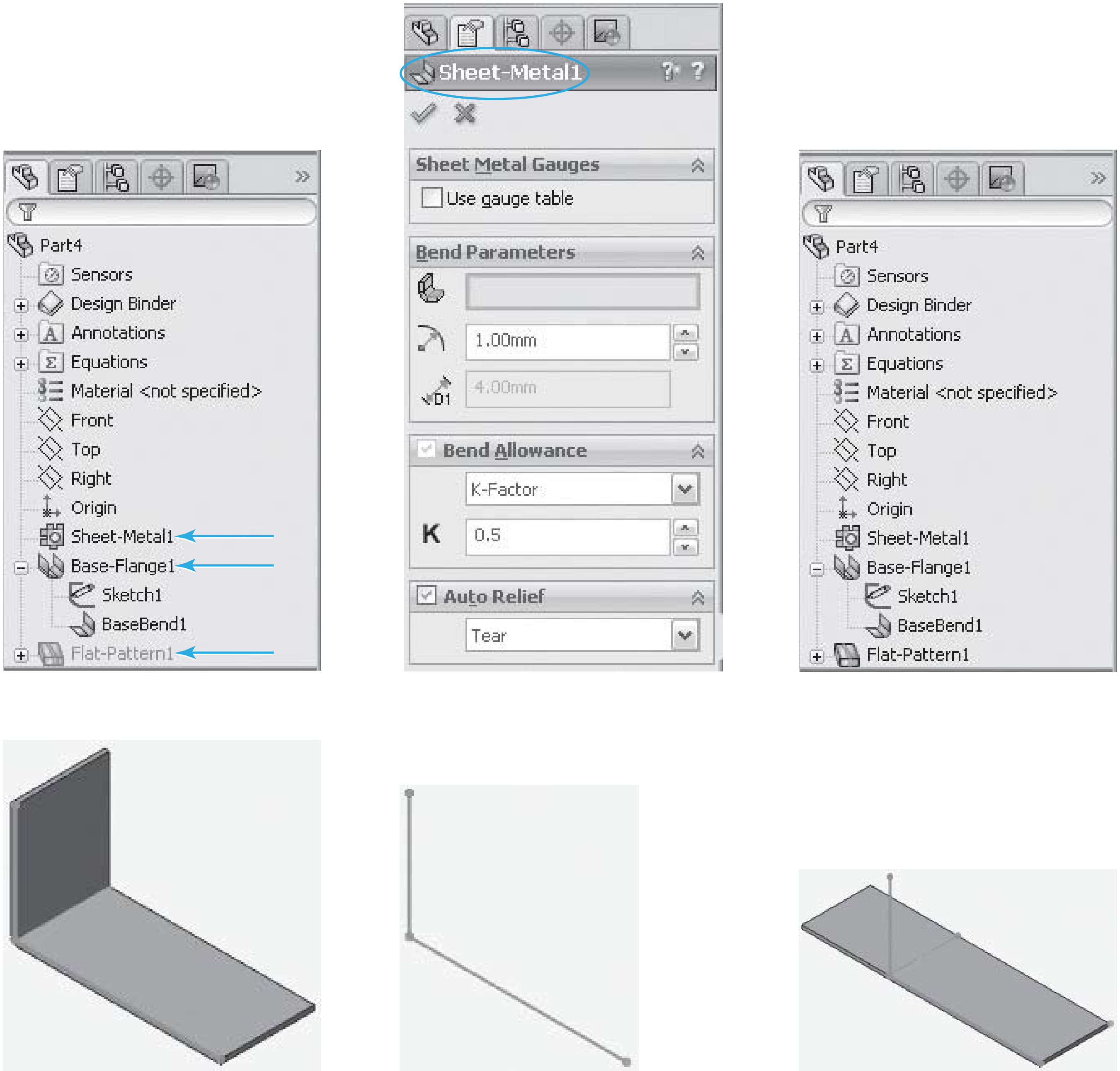
Feature	Input	Before feature creation	After feature creation
Corner (we use break corner)	Corner to break, break type (use chamfer), and distance **Hint:** We break (chamfer) the top right corner.		
Hem	Edge(s) to hem **Hint:** Hemming an edge is like bending (folding) it on itself as shown.		
Rip	Edge(s) to rip, and a rip direction(s) (arrows shown to the right)		
Lofted bend	Two sketches with no sharp corners **Hint:** Corners must be filleted to allow creating a bend for sheet metal.		

TABLE 10.1 Available Sheet Metal Features (*continued*)			
Feature	**Input**	**Before feature creation**	**After feature creation**
Unfold	Fixed face and bends to unfold		
Fold	Fixed face and bends to fold (of unfolded part)		
Flatten	Unflattened part **Hint:** Unfolding and flattening a part are not the same.		

10.4 Sheet Metal FeatureManager Design Tree

Thus far in the book, when we create a base feature in a part, only the feature is created as we expect and shows as one node in the part features tree. Sheet metal modeling is different. When we create a base flange feature in a sheet metal part, SolidWorks creates a total of three features: the base flange and two supporting features, as shown in Figure 10.4. **Sheet-Metal1** contains the bend parameters including the bend radius, bend allowance, and relief type. Right-click it to edit the default values if needed.



(A) **Base-Flange1:** Bent state

(B) Bend parameters

(C) **Base-Flange1:** Flatten state

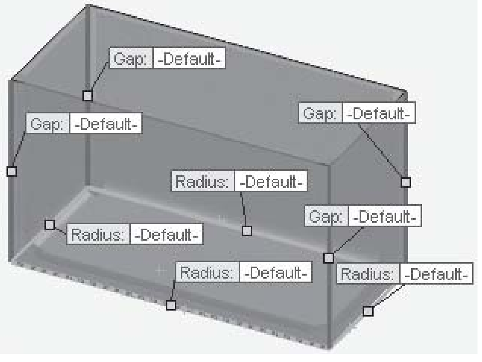
FIGURE 10.4
Sheet metal supporting features

Base-Flange1 is the base feature that represents the sheet metal part we want. This tree node has two subnodes: the feature sketch and the bend radius. **Flat-Pattern1** enables you to flatten/unflatten the part. The flat state is suppressed by default; SolidWorks shows the part in its bent state. To flatten the part, click the **Flat-Pattern1** node and select the **Unsuppress** icon from the context toolbar that pops up (hover on icons to read them). When you exit the flat state (click the exit icon shown on the top right of screen), it puts the part back in its bent state. You can achieve the same result by clicking the **Flatten** icon shown in Figure 10.2, once to flatten and again to bend (the icon is a toggle) . Note that when you flatten the part, the new features you create are inserted below the **Flat-Pattern1** node in the features tree; otherwise, they are inserted above it.

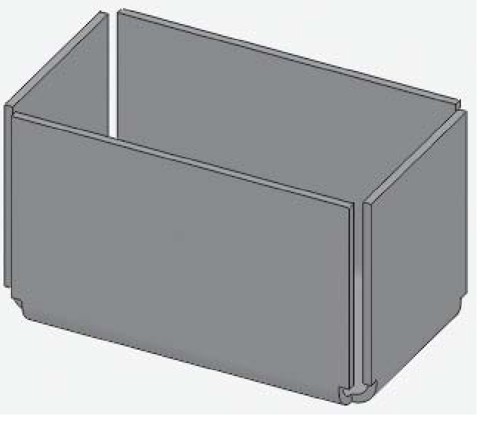
10.5 Sheet Metal Methods

There are four methods to create sheet metal parts:

1. Create a sheet metal part using the sheet metal features covered in Section 10.4 and shown in Table 10.1. Here, you start by creating a base flange and add the needed sheet metal features until the part is complete.
2. Create a sheet metal part by converting a solid body: After we create a solid as usual, use **Insert** (menu) > **Sheet Metal > Convert To Sheet Metal**. This requires a fixed face and edges that are planar with the fixed face. These are the bend edges that are used to create the sheet metal part. Figure 10.5 shows an example. We select the bottom face as the fixed face and its four edges as the bend edges. When you select the face and its edges, SolidWorks selects the four vertical edges as the rip edges to create the sheet metal part.



(A) Solid part and input

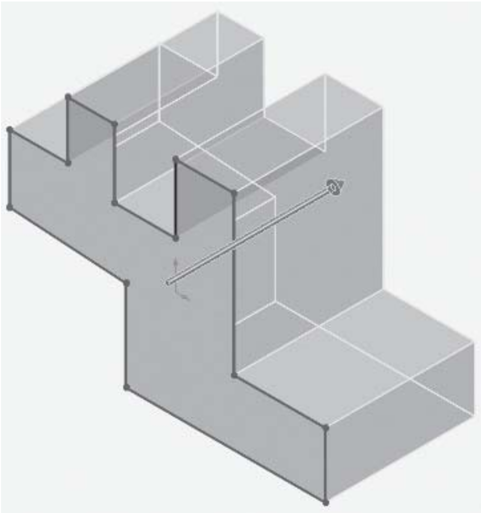


(B) Resulting sheet metal part

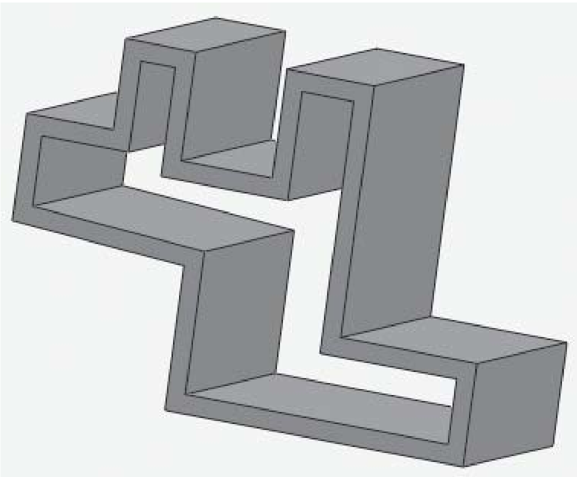
FIGURE 10.5
Converting a solid part to a sheet metal part

3. Convert a shelled solid body to a sheet metal part: This method requires a shelled solid. After we have the shelled solid, we use **Insert Bends** to create the sheet metal part. Following is the sequence of using this method.

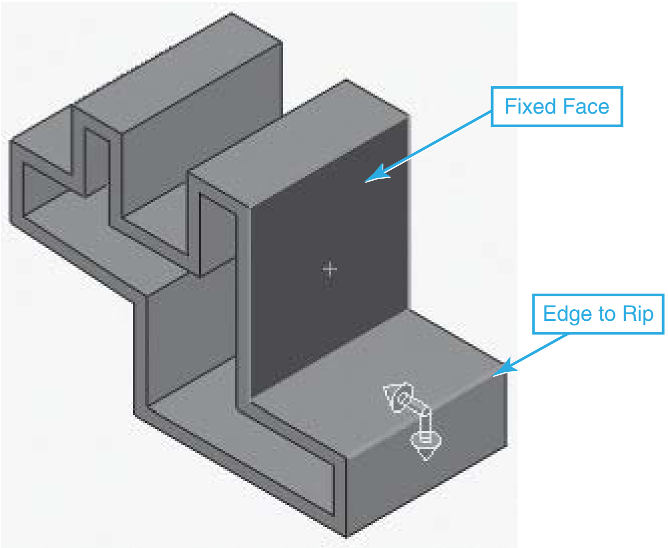
Step 1: Create an extrusion: Create the sketch shown on **Front Plane** and extrude it.

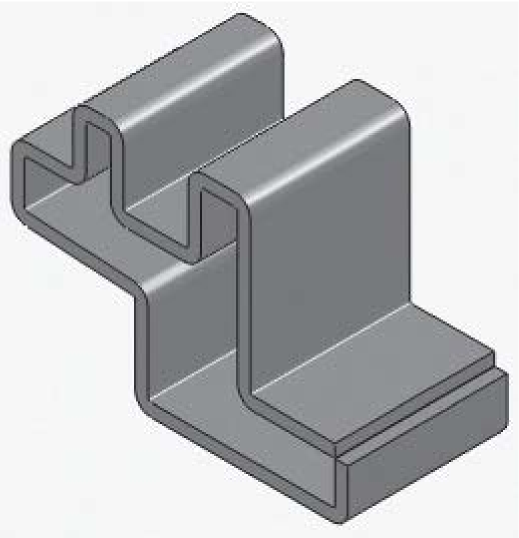


Step 2: Shell the extrusion: **Shell** on **Features** tab > select front and back faces of extrusion > enter 4 (mm) for shell thickness.



Step 3: Create the sheet metal part: **Insert Bends** on **Sheet Metal** tab > select shown face as a fixed face > select shown edge to rip.

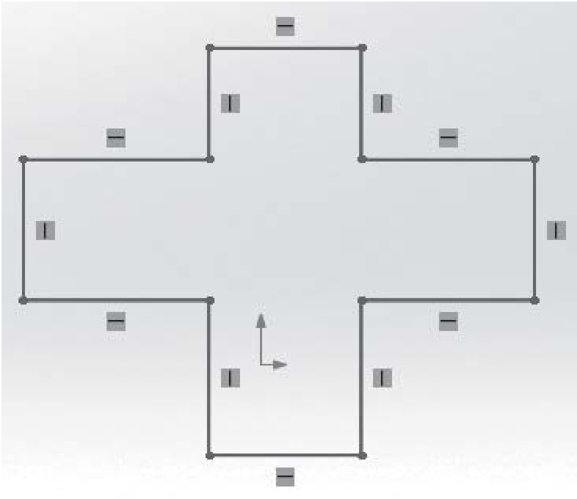




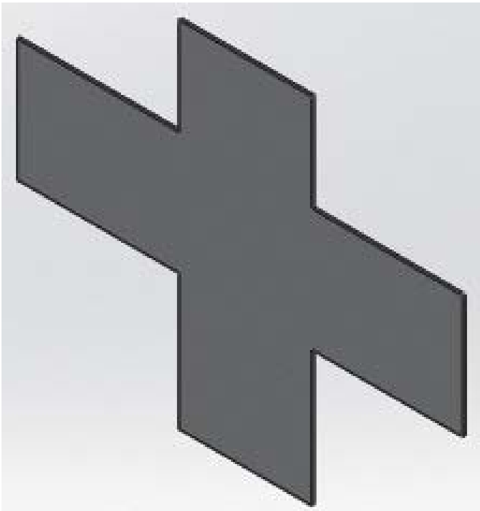
Resulting Sheet Metal

4. Design a sheet metal part from the flattened state: In this method, you create the sheet metal as one sketch and then create bend lines (in another sketch) where you want to bend (fold) the flat sheet (sketch) to create the part. The following steps show how to use this method.

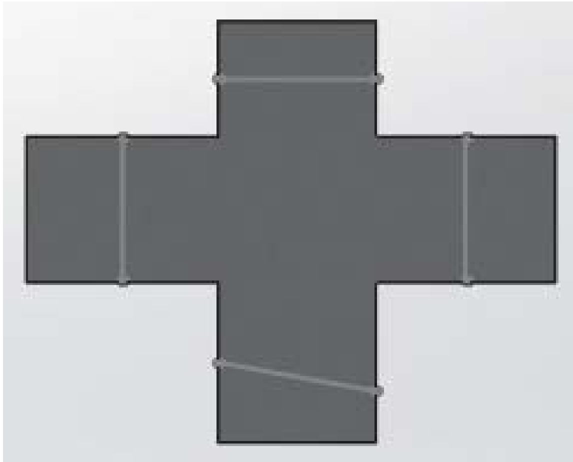
Step 1: Create the sketch of the sheet metal part: **File > New > Part > OK > Front Plane** > sketch as shown > exit sketch > **File** > **Save As** > *method4* > **Save.**



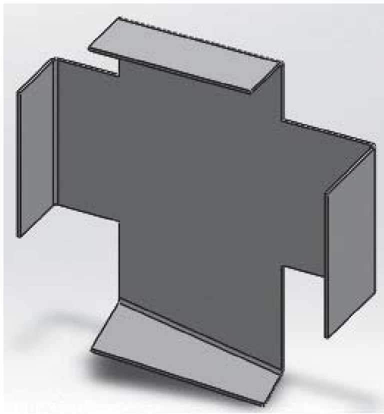
Step 2: Create the sheet metal part without bends: **Base Flange/Tab** on **Sheet Metal** tab > accept default values > ✔.



Step 3: Create the bend sketch of the sheet metal part: **Front Plane** as sketch plane (or front face of base flange created in Step 2 > **Line** on **Sketch** tab > sketch as shown > exit sketch.



Step 4: Create the sheet metal bend: Select *Sketch2* > **Sketched Bend** on **Sheet Metal** tab > click front face of unbent sheet metal created in Step 2 > ✔.



10.6 Weldments

Welding is a fabrication process that joins metal (steel, aluminum) or thermoplastic components together by melting the joints (workpieces) and adding a filler material to mix with the joints' materials. When the joints are cooled down, a strong joint connection is created. Example products that use welds are metal frames used in building construction, exhaust system and pipes of a car, campfire grill, trailer dolly, rainwater system (house gutters), heat exchangers, and condensers.

Welding processes include gas welding, arc welding, spot welding, resistance welding, solid state welding, and others. All these processes have one thing in common: They all require an energy source to heat up and melt the weld joints and the filler material.

They all differ in the source of generating the energy (e.g., gas, electric arc, resistance). For example, gas welding burns gas to generate very hot flames to melt and fuse metal joints with the filler material. You probably have seen or used one of those small portable cylinders that burns propane and produces a flame torch that you can use for welding.

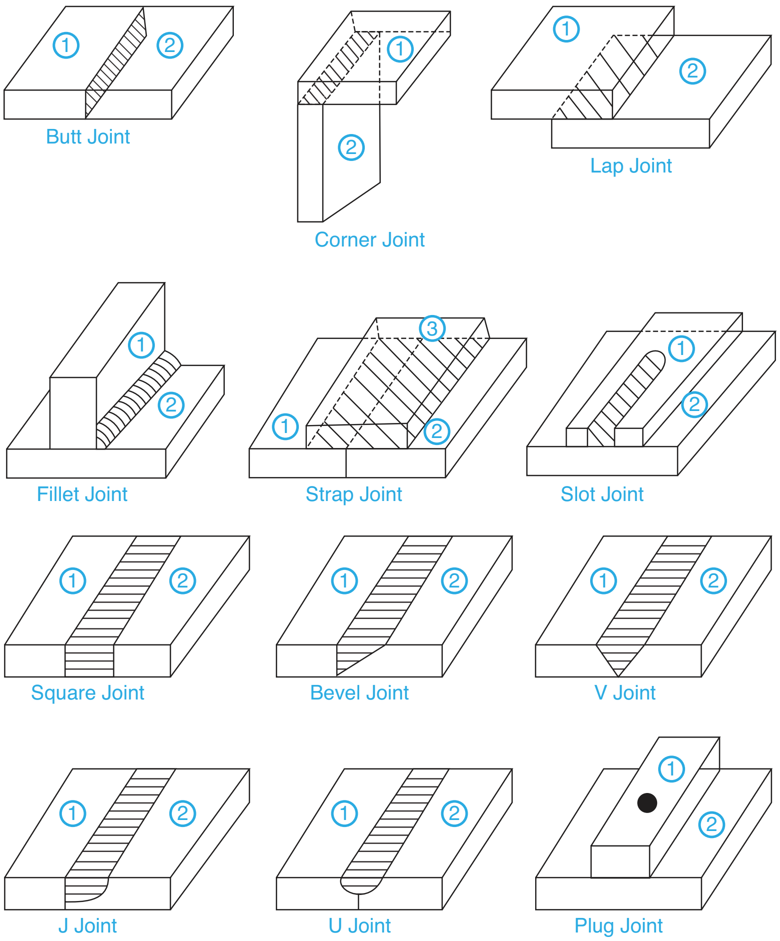
Welding equipment and supplies include welding machines, cylinders, spot welders, cutters, torches, welding hoods (to contain fumes), grinders, filler metals, air compressors, cutting guides, clamps and holders, and safety equipment. Gas welding uses gas cylinders that are filled with gas (propane) under high pressure. These cylinders are regulated and follow strict standards and codes.

Weld defects can occur, and they include inclusions, segregation, and porosity. **Inclusions** are impurities of foreign substances that get into the weld puddle (area of molten metal) during welding and become embedded in the weld joints after they solidify. These inclusions have the same effect as cracks because they are typically much weaker than the weld joint material. **Segregation** is a condition in which the weld puddle does not have uniform metal mix. For example, there may not be enough filler material flowing uniformly in the puddle. **Porosity** is the formation of voids or tiny pinholes that result from trapping air bubbles in the puddle.

Weld joints can be inspected to assess joint quality. Nondestructive examination (NDE) methods are used to inspect weld joints. These methods include visual, liquid penetration, magnetic particle, ultrasonic, and X-ray (radiographic). These methods allow us to "see through" weld joints to look for inclusions, segregation, and porosity.

If welding is all about connecting weld joints together, what kind of weld joints can we design and use? There are many types of weld joints, and Figure 10.6 shows the

FIGURE 10.6
Types of weld joints



common types. The parts that are welded are shown with a number in a circle. The weld puddle (area) is shown as hatched.

Which joint type is best suited for a particular application? The answer depends on many factors. Weld joints are designed primarily to meet strength and safety requirements. Other considerations in selecting the type of weld joint design are the joint load and the ratio of the joint strength to the base metal strength. The type (tension or compression) and amount of the load that the joint carries determine the joint type. For example, if we have a vertical compression load at the joint, a butt joint would be a weak joint, whereas a corner or a lap joint would be better and stronger. The strength ratio could be used to determine the safer mode of failure of the weld joint. For example, we could use a butt joint that shears off first before the welded parts themselves fail.

We refer to the welded parts as *weldments*. A **weldment** is a part we design using CAD/CAM systems. Weldments can take various shapes such as structural members and blocks. You can use 2D and 3D sketches to define the basic framework of a weldment structure. After you define the frame, you create the structural members and add gussets, and end gaps. The following section provides more details.

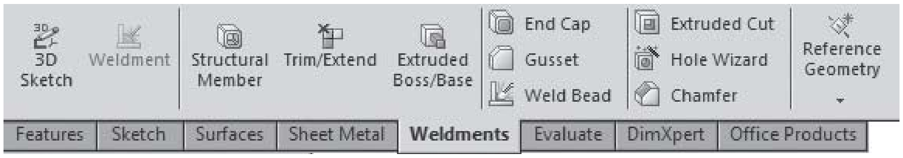
10.7 Weldment Features

Welded parts are mostly structures consisting of frames or trusses, that is, skeletons. Each frame consists of members that are welded (connected) together. The frames usually form a 3D sketch. After creating the frame (skeleton), we add cross sections to it to create the structural members, followed by creating the weld joints, and closing the members' ends (if needed). Thus, the modeling steps of creating welded parts (weldments) are as follows:

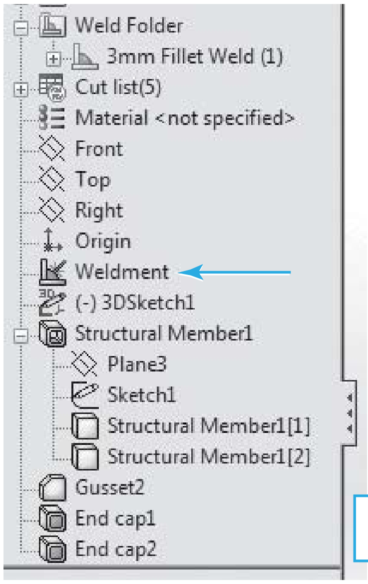
1. Create the weld frame: Use a 2D or 3D sketch depending on the frame design at hand.
2. Create the structural members: Each member uses a cross section. These members trace the frame shape. Think of it as adding meat (members) to the bones (frame skeleton).
3. Create the weld joints: Add the welded joints based on their types (Figure 10.6). Also, stiffen the joints (if needed) by adding ribs.
4. Close the ends of the members if needed: Use end caps to close the open ends of the weld structure. These ends are welded onto the open ends of the structure.

As with sheet metal modeling, modeling welded parts requires unique features such as weld joints, stiffeners, and end caps. These features must be easy to use. SolidWorks provides a comprehensive and easy-to-use **Weldments** module with ample features. Figure 10.7A shows these features. The features come in three groups: base, weld-specific, and generic. The base feature is the Weldment feature shown in Figure 10.7B. It is not a feature in the common sense that we know such as extrusion or revolve. You cannot edit the Weldment feature, but you can delete it (right-click it in a features tree to see what you can do with it). Think of the Weldment feature as a container that sets up the weldment modeling environment in SolidWorks and designates the part as a weldment. All the features of a welded part (see Figure 10.7C) you create become children of the Weldment (parent) feature in the features tree. If you delete the parent, all its children are deleted with it.

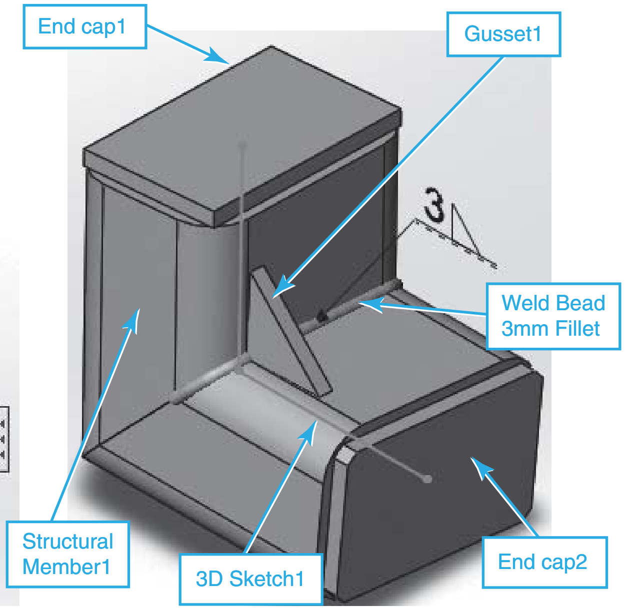
The second group of Weldment features shown in Figure 10.7A is the weld-specific features. They are Structural Member, Trim/Extend, End Cap, Gusset, and Weld Bead. A **Structural Member** sweeps a predefined (by SolidWorks) cross section (profile) along user-defined paths created using sketching. The profiles follow industry standards.



(A) **Weldments** tab



(B) Weldment feature



(C) Welded part whose tree is shown to the left

FIGURE 10.7
Weldment features

SolidWorks provides drop-down lists to select from to ensure adherence to these standards. You would need to select the type of standard (ANSI or ISO), the profile type (C channel, pipe, square tube), and the profile size. You can group structural members as you create them. When you delete one member of the group, the entire group is deleted.

A **Trim/Extend** feature closes the structural members by trimming or extending them relative to each other to create a coherent, integrated welded part. An **End Cap** feature closes the open-end faces of structural members. This is useful if you want to prevent any foreign materials (water, debris) from going inside the structural members of a welded part. A **Gusset** is a rib that you add between two adjoining planar faces to stiffen a weld joint. Adding gussets between faces has strict requirements. You cannot add gussets between nonplanar faces or faces that do not share edges. We leave the details of creating gussets to the tutorials. A **Weld Bead** creates a fillet joint between two structural members.

The third and final group of Weldment features shown in Figure 10.7A is the generic features. These are features we have already used before in modeling any parts. They are included as part of the **Weldments** tab as a convenience to the designer to circumvent going to the **Features** tab to grab them from there. These features are the ones most commonly needed in the context of designing welded parts. As shown in Figure 10.7A, they are **Extruded Boss/Base**, **Extruded Cut**, **Hole Wizard**, **Chamfer**, and **Reference Geometry**. Another important item of the generic group is the **3D Sketch** icon that we use to create the frame of a welded part. Table 10.2 shows the SolidWorks Weldment features and how to create them.

TABLE 10.2 Available Weldment Features

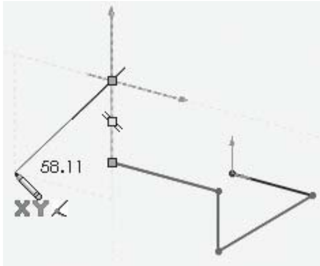
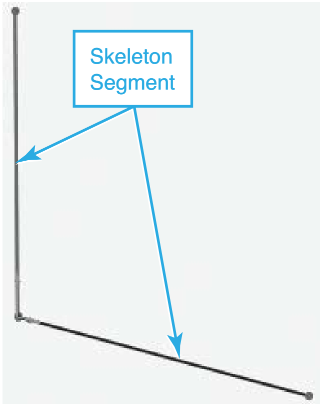
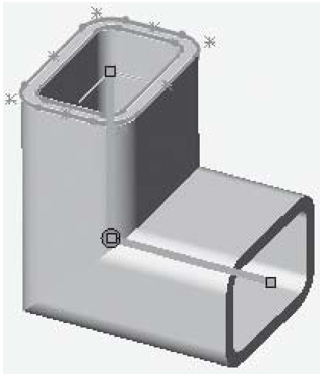
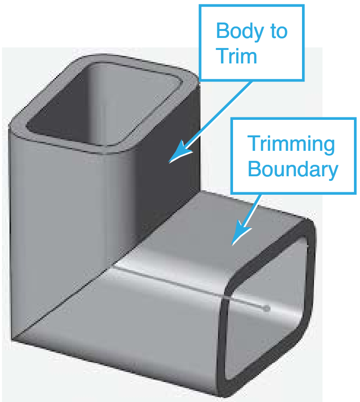
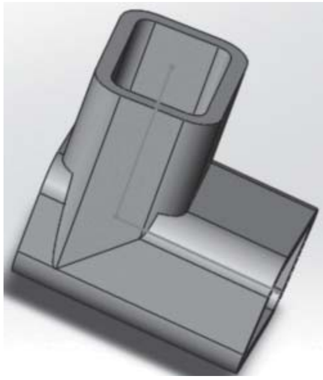
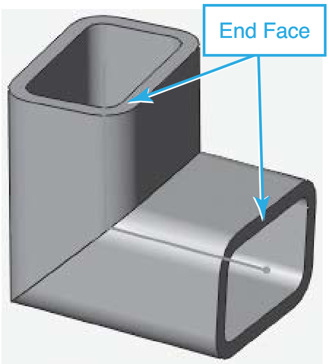
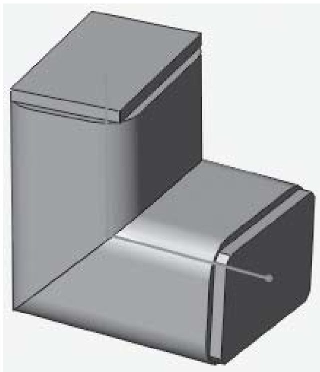
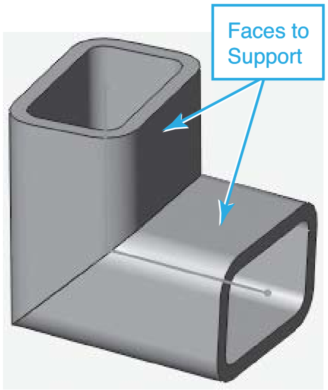
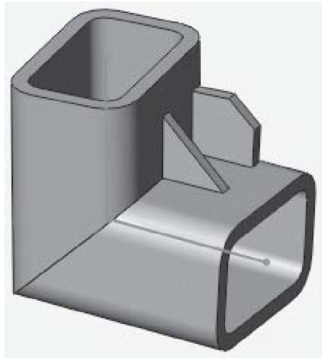
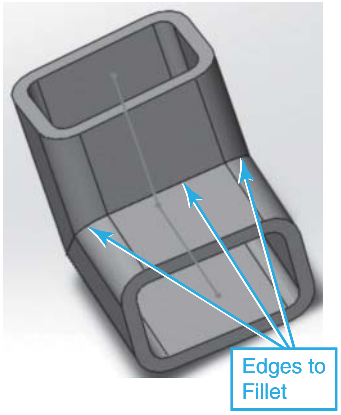
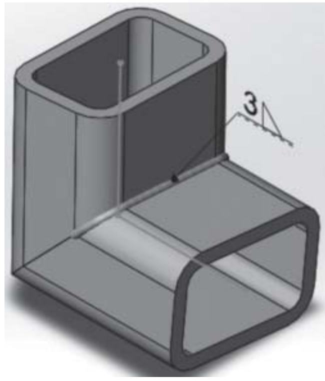
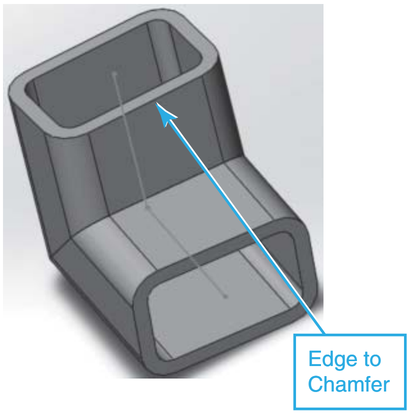
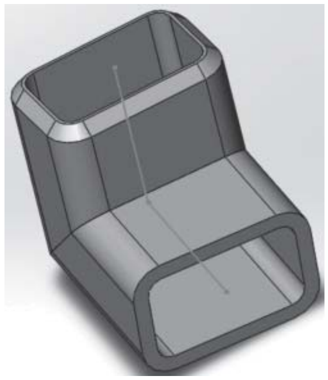
Feature	Input	Before feature creation	After feature creation
3D Sketch	Sketch entities along the 3D axes	None	
Structural Member	Sketch (skeleton) segment(s), standard (ANSI or ISO), profile type, and profile size		
Trim/Extend	Body to trim, trimming boundary, and others (see left side of SolidWorks screen for other inputs)		
End Cap	End face of structural member, cap thickness, thickness direction, and type of chamfer corners		

TABLE 10.2 Available Weldment Features (*continued*)

Feature	Input	Before feature creation	After feature creation
Gusset	Faces to support, gusset profile type, thickness amount and direction, and gusset location		
Weld Bead	Edges to fillet		
Chamfer	Edge to chamfer		

10.8 Weld Symbols

The Weld Bead feature in Table 10.2 shows a weld symbol with the fillet size (3 mm). We need to understand these weld symbols because we use them in welding engineering drawings to specify the weld joint type and parameters. There are ISO and ANSI weld symbols. Table 10.3 shows SolidWorks ISO weld symbols. Table 10.4 shows SolidWorks ANSI weld symbols. Note that all these symbols can also be displayed in an upside-down fashion (have identification line on top as shown for the first symbol in Table 10.3) to accommodate the other style and preference of displaying.

TABLE 10.3 ISO Weld Symbols

Or Butt	Square butt	Single V butt	Single V butt with root	Single bevel butt	Single bevel butt with root
Single U butt	Single j butt	Backing run	Fillet	Plug or slot	Spot
Spot (centered)	JIS spot	Seam	Seam (centered)	JIS seam	Flat
Convex	Concave	Fillet with flat	Fillet with Convex	Fillet with Concave	

TABLE 10.4 ANSI Weld Symbols

Square	Scarf	V Groove	Bevel	U groove	J Groove
Flare-V	Flat-bevel	Fillet	Plug or Slot	Stud	Spot or Projection
Spot or Projection (Centered)	Seam	Seam (Centered)	Back Fill	Surfacing	Flange-Edge
Flange-Corner	Consumable-Insert	Melt Through	Flat	Convex	Concave
Fillet with Flat	Fillet with Convex	Fillet with Concave			

10.9 Tutorials

The theme for the tutorials in this chapter is to practice creating and using sheet metal and weldments. The tutorials show how to create sheet metal and weld parts, as well as create sheet metal and welding drawings.

Tutorial 10–1: Create Sheet Metal

This tutorial covers the basic sheet metal operations. We use the sheet metal features to create the part shown in Figure 10.8. All dimensions are in inches.

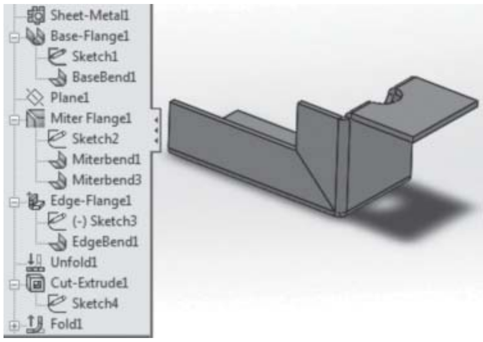
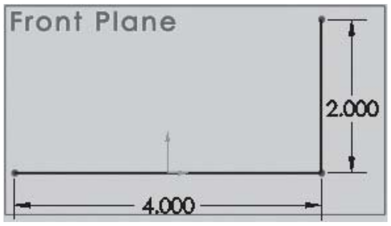
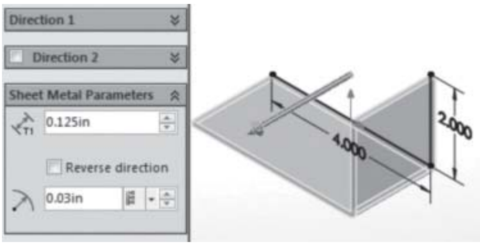


FIGURE 10.8
A sheet metal part

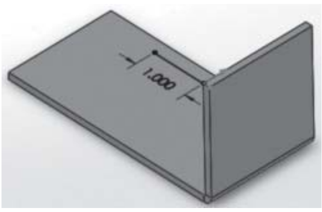
Step 1: Create *Sketch1*: **File > New > Part > OK > Front Plane** on **Sketch** tab > sketch and dimension lines shown > exit sketch > **File > Save As** > *tutorial10.1* > **Save.**



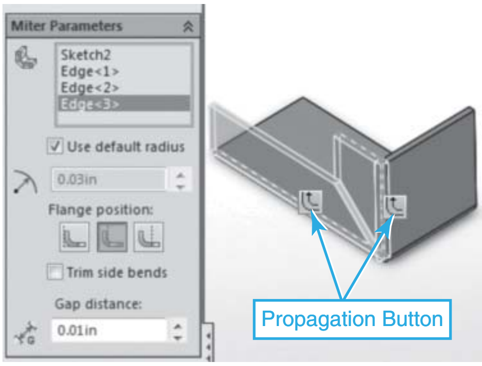
Step 2: Create *Base-Flange1*: **Base Flange/Tab** on **Sheet Metal** tab > enter 0.125 for **T1** and 0.03 for bend radius > ✔.



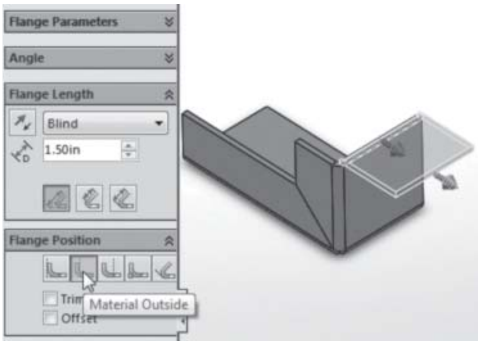
Step 3: Create *Sketch2*: top face of *Base-Flange1* > sketch and dimension line shown > exit sketch.



Step 4: Create *Miter Flange1*: *Sketch2* > **Miter Flange** on **Sheet Metal** tab > select edges shown > ✔.

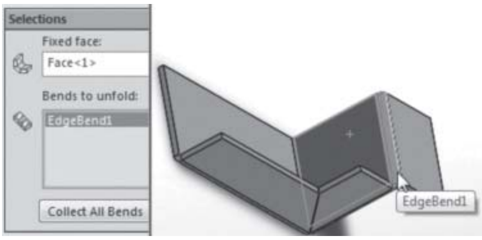


Step 5: Create *Edge-Flange1*: **Edge Flange** on **Sheet Metal** tab > outside edge of top face as shown > enter 1.5 for **Flange Length** > select **Material Outside** > ✔.

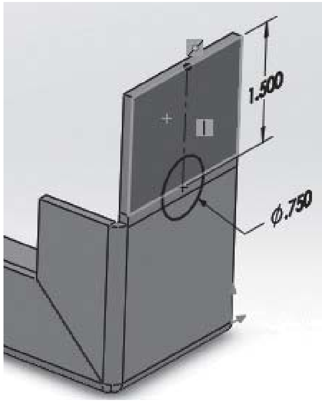


Note: *Sketch3* shown in features tree is created automatically as part of edge flange definition.

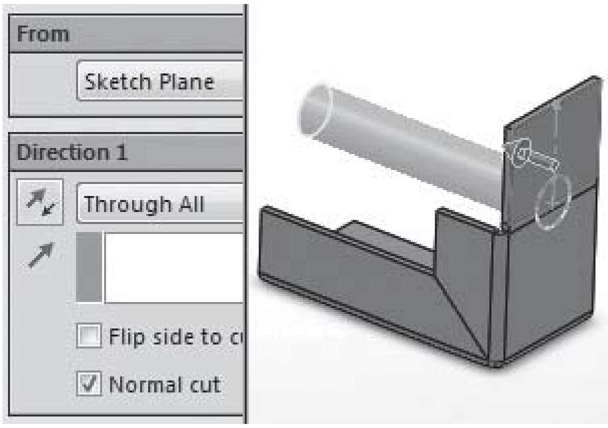
Step 6: Create *Unfold1* to unfold *Edge-Flange1*: **Unfold** on **Sheet Metal** tab > face shown as **Fixed face** > bend shown as **Bends to unfold** > ✔.



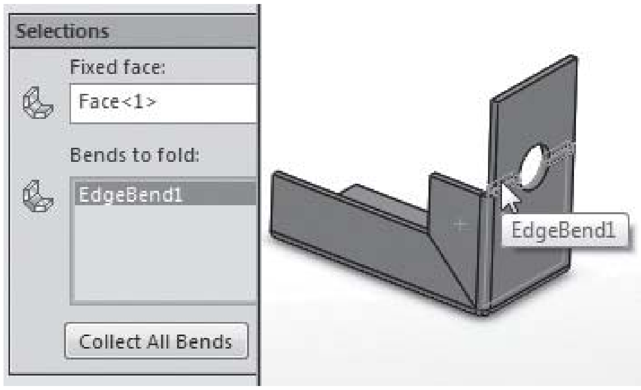
Step 7: Create *Sketch4*: Face shown as sketch plane > **Centerline** on **Sketch** tab > sketch and dimension vertical line shown to pass through midpoint of top edge > **Circle** on **Sketch** tab > sketch and dimension circle shown > exit sketch.



Step 8: Create *Cut-Extrude1*: *Sketch4* > **Extruded Cut** on the **Sketch** tab > **Through All** > ✔.



Step 9: Create *Fold1* to fold edge back: **Fold** on the **Sheet Metal** tab > hover to select *EdgeBend1* > ✔.



HANDS-ON FOR TUTORIAL 10–1. Mirror the miter flange to make the model symmetric.

Tutorial 10–2: Create Sheet Metal Drawing

This tutorial covers the basic sheet metal drawing options. We create the sheet metal drawing shown in Figure 10.9. A sheet metal drawing is like any typical drawing, but with one difference: We can create a flat pattern view in the drawing that has special notes that tell how the sheet metal is created from the flat pattern. For example, Figure 10.9 shows three "UP 90° R 0.03" indicating that the corresponding flat pattern is bent upward a 90° angle at the three edges shown, with a bend radius of 0.03 inch. Similarly, the "DOWN 90° R 0.03" indicates that the side wall is bent down a 90° angle at the edge shown, with a bend radius of 0.03 inch.

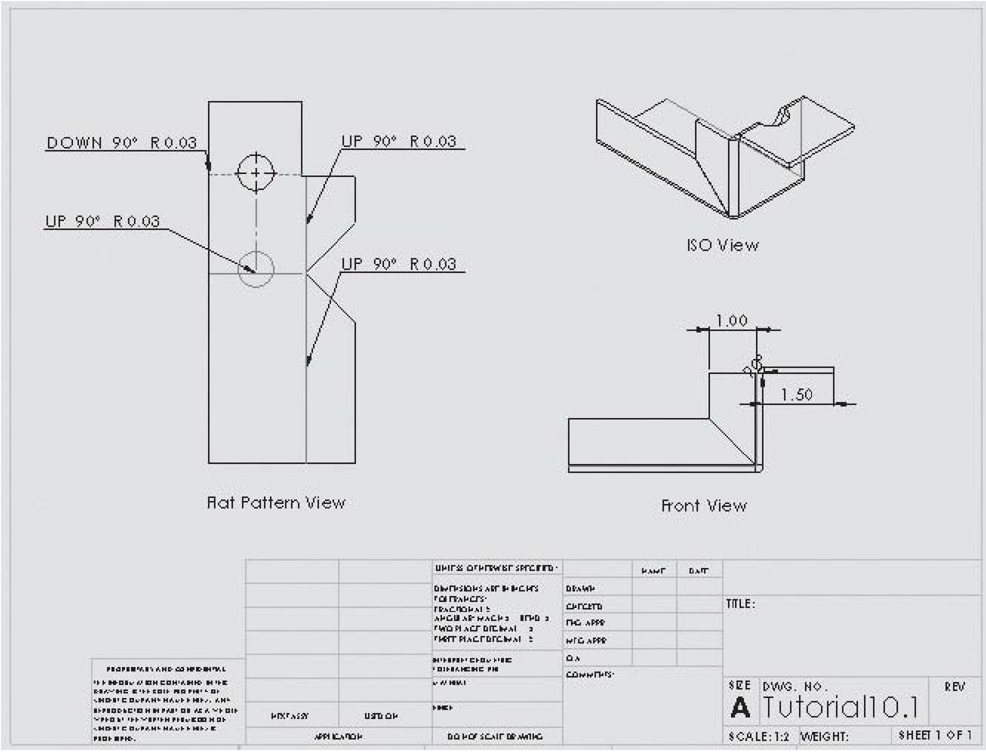
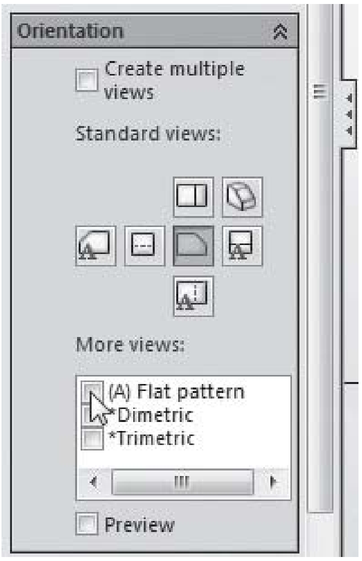
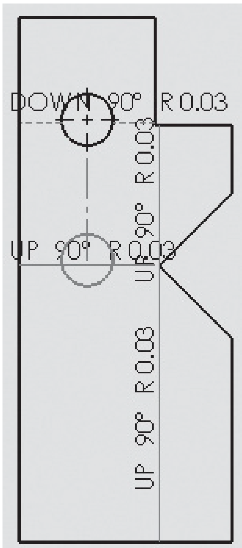


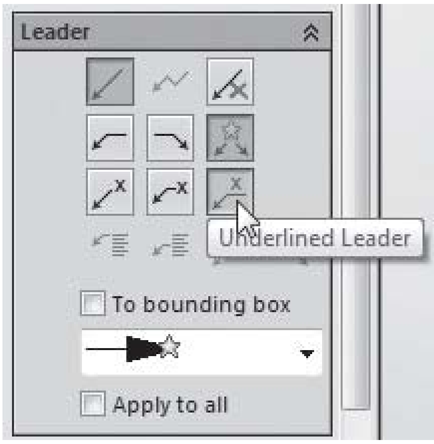
FIGURE 10.9
A sheet metal drawing

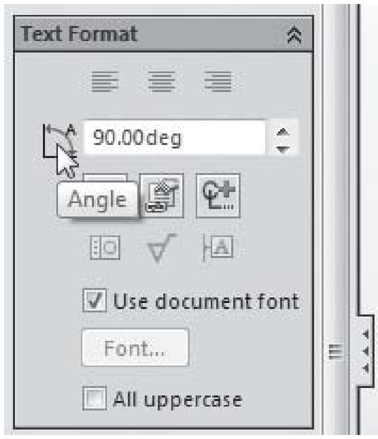
Step 1: Create *drawing*: **File** > **New** > **Drawing** > **OK** > **OK** > **File** > **Save As** > *tutorial10.2* > **Save.**





Step 2: Insert **Flat Pattern** view: **Browse** button > locate *tutorial10.1* part file > **Open** > check off **(A) Flat pattern** box under **Orientation** section shown above > click in drawing sheet area to place view shown above > ✔.





Step 3: Adjust the display of notes shown in Step 2: Click a horizontal note > select leader box under **Leader** section shown > select the bottommost right box as shown > drag the note selected to position > repeat for other horizontal note > repeat for each vertical note but change 90° to 0° under **Text Format** section to change note orientation as shown in Figure 10.9 > ✔.

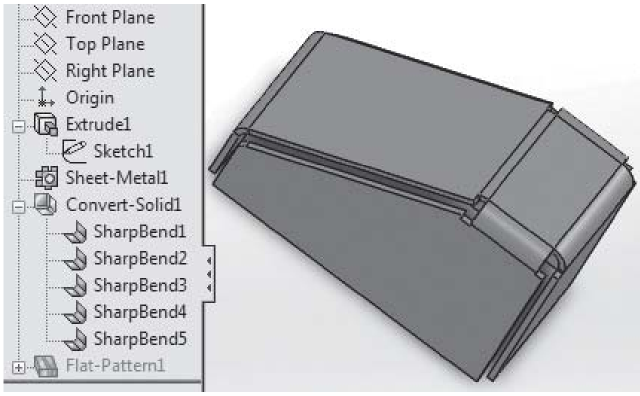
Step 4: Add more views and dimensions: **Model View** on **View Layout** tab > **Next** (little arrow on top right) > insert **ISO** and **Front** views > ✔ > **Smart Dimension** on **Sketch** tab > dimension as shown in Figure 10.9 > ✔.

> ***HANDS-ON FOR TUTORIAL 10–2.*** Generate the drawing (similar to Figure 10.9 drawing) for the sheet metal part of the hands-on of Tutorial 10.1.

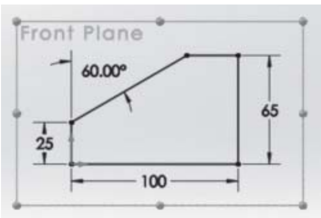
Tutorial 10–3: Create Sheet Metal Part from Solid Body

We create a sheet metal part from a solid part, as shown in Figure 10.10. All dimensions are in mm.

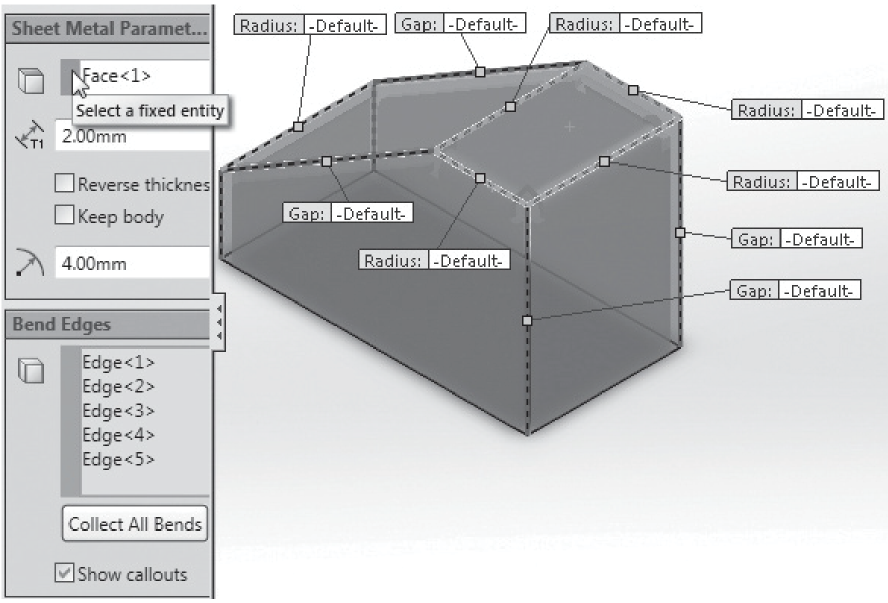
FIGURE 10.10
A sheet metal part converted from a solid body



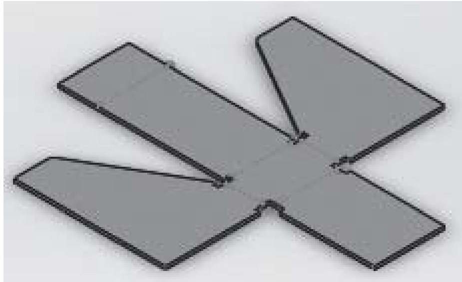
Step 1: Create *Sketch1* and *Extrude1* feature: **File > New > Part > OK > Front Plane > Extruded Boss/Base** > sketch and dimension as shown > exit sketch > **Mid Plane** > enter 50 for **D1** > ✔ > **File > Save As** > *tutorial10.3* > **Save.**



Step 2: Create *Convert-Solid1*: **Convert to Sheet Metal** on **Sheet Metal** tab > top face as the fixed entity > all edges of top face and far edge of angled face as **Bend Edges** as shown > enter 2 for **T1** and 4 for radius > ✔.



Note: The system decides on rip edges to use to break open the solid. Unsuppress the *Flat-Pattern1* node to view sheet metal as shown.



HANDS-ON FOR TUTORIAL 10–3. Create a drawing for the part created in Tutorial 10–3. Insert both a folded isometric view and a flat pattern view.

Tutorial 10–4: Create Sheet Metal Part from Flattened State

This tutorial covers the ability to create a sheet metal part, shown in Figure 10.11, by sketching a base flange in flat state and bending it. The model is created by sketching a base flange on one sketch, sketching bend lines on a separate sketch, and folding the flange using the bend lines. All dimensions are in inches.

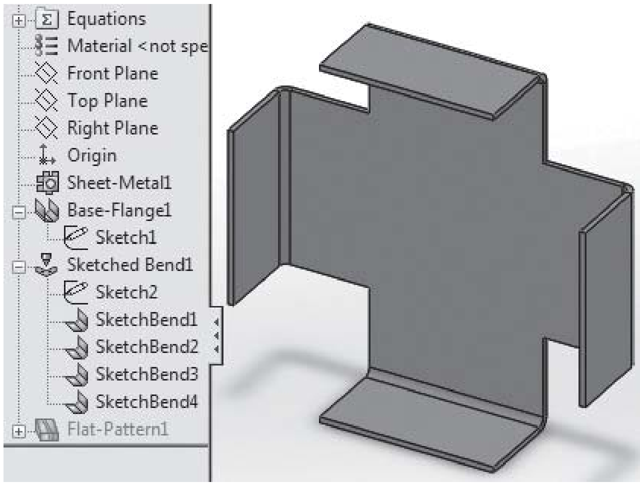
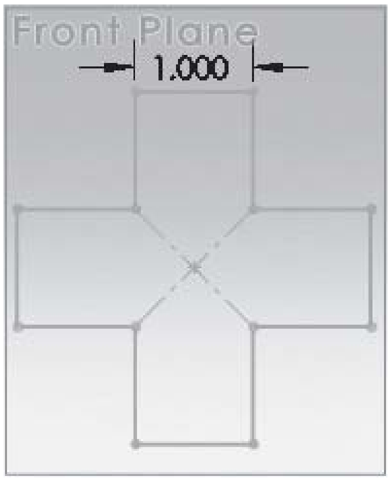
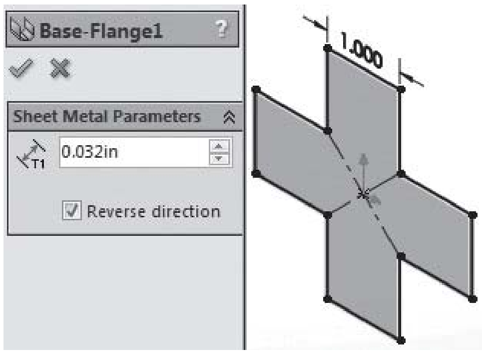


FIGURE 10.11
A sheet metal part created from a flattened state

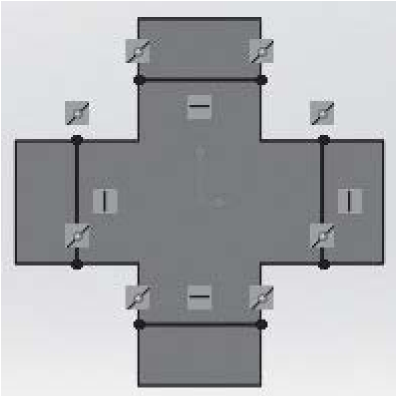
Step 1: Create *Sketch1*: **File > New > Part > OK > Front Plane >** sketch and dimension equal-length horizontal and vertical lines shown > exit sketch > **File > Save As >** *tutorial10.4* > **Save.**



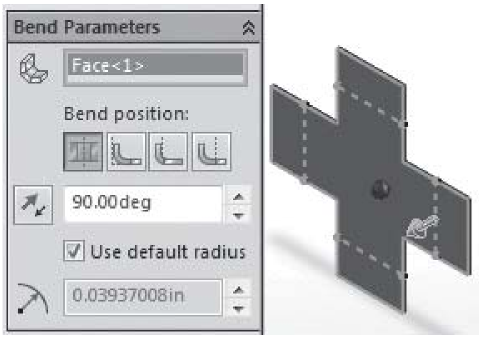
Step 2: Create *Base-Flange1*: **Base Flange/Tab** on **Sheet Metal** tab > type 0.032 for **T1** > ✔.



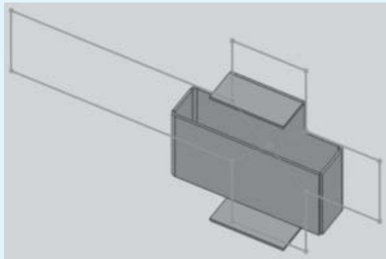
Step 3: Create bend lines in *Sketch2*: Front face of *Base-Flange1* > snap to midpoints of edges to sketch and dimension lines shown > exit sketch.



Step 4: Create *Sketched-Bend1*: *Sketch2* > **Sketched Bend** on **Sheet Metal** tab > click front face of *Base-Flange1* somewhere inside of the four lines to define fixed face > ✔.



HANDS-ON FOR TUTORIAL 10–4. Modify *Sketch1* of Step 1 to create the sheet metal part shown to the right.



Tutorial 10–5: Create Weldment

This tutorial covers the basic weldment operations. We use the weldment operations to create the weldment structure shown in Figure 10.12. We create the 2D sketches first and then add structural members along them. All dimensions are in inches.

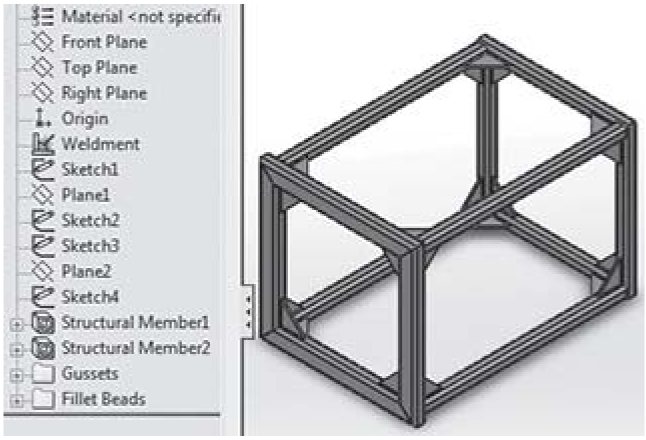
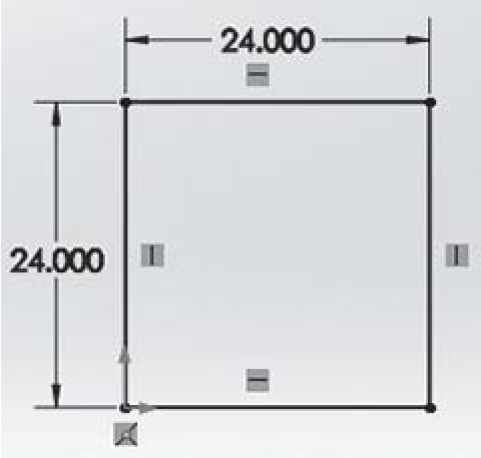
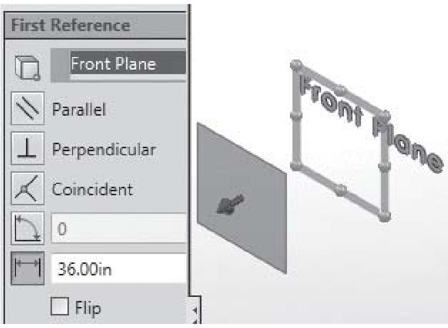


FIGURE 10.12
A weldment structure

Step 1: Create *Sketch1*: **File** > **New** > **Part** > **OK** > **Front Plane** > sketch and dimension lines shown > exit sketch > **File** > **Save As** > *tutorial10.4* > **Save.**

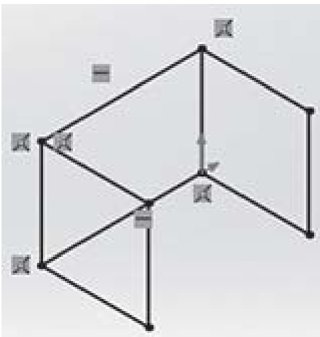


Step 2: Create *Plane1*: **Reference Geometry** on **Features** tab > **Plane** > expand features tree > **Front Plane** > enter 36 for distance > ✔.

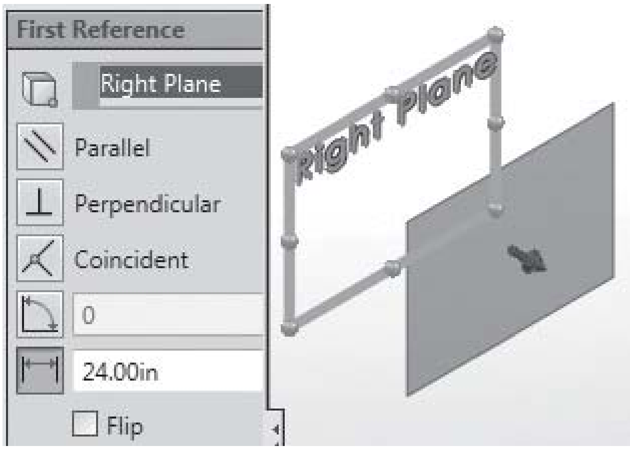


Step 3: Create *Sketch2*: *Plane1* > **Convert Entities** on **Sketch** tab > expand features tree > *Sketch1* > ✔.

Step 4: Create *Sketch3*: **Right Plane** > sketch two lines shown to bridge *Sketch1* and *Sketch2* > exit sketch.

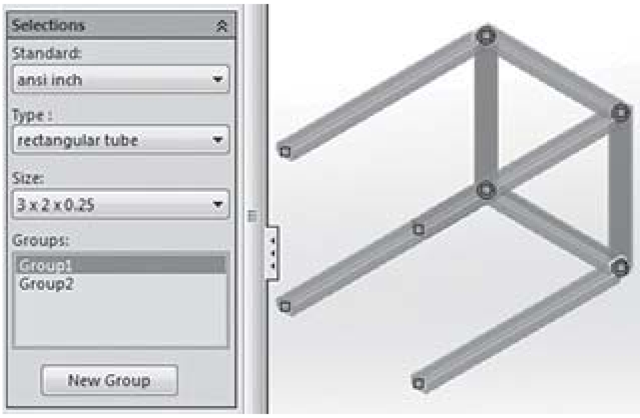


Step 5: Create *Plane2*: **Reference Geometry** on **Features** tab > **Plane** > expand features tree > **Right Plane** > enter 24 for distance > ✔.

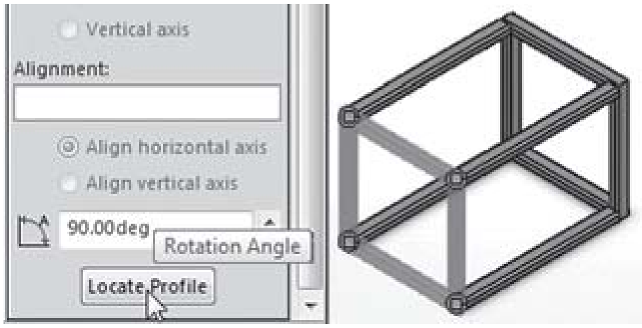


Step 6: Create *Sketch4*: *Plane2* > **Convert Entities** on **Sketch** tab > expand features tree > *Sketch3* > ✔.

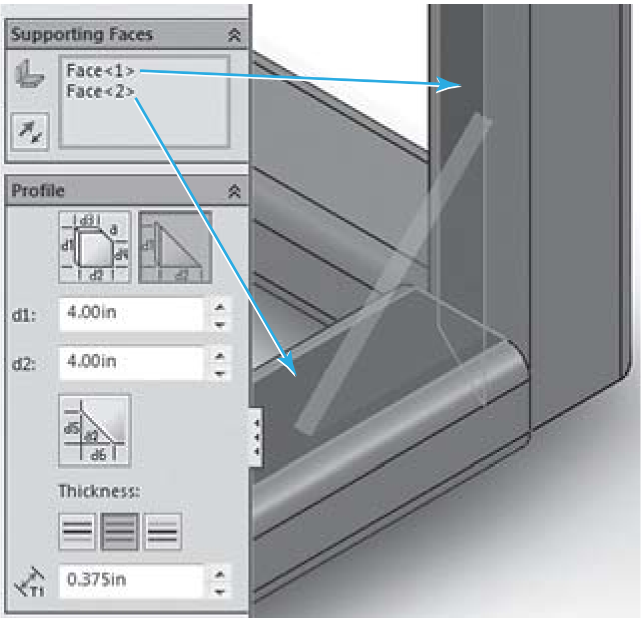
Step 7: Create *Structural Member1*: **Structural Member** on **Weldments** tab > set **Selections** shown > expand features tree > *Sketch1* > **New Group** button > expand features tree > *Sketch3* > *Sketch4* > ✔.



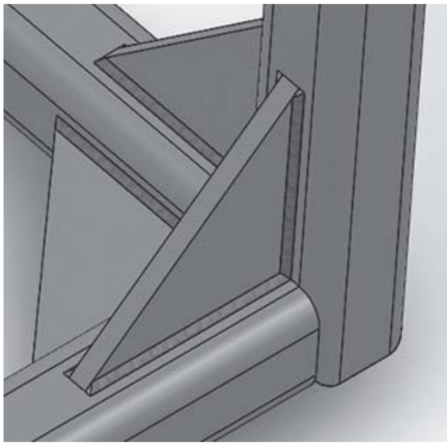
Step 8: Create *Structural Member2*: **Structural Member** on **Weldments** tab > set **Selections** as in Step 7 > expand features tree > *Sketch2* > set **Apply corner treatment** to **End Miter** (hover to read it) > enter 90 for **Rotation Angle** to orient profile with respect to other structural members > **Locate Profile** button > ✔.



Step 9: Insert gussets: **Gusset** on **Weldments** tab > select two faces shown > enter 4 for **d1** and **d2** > select **Both Sides** (hover to read it) for **Thickness** > enter 0.375 for **T1** > ✔ > repeat to create a total of 24 gussets (3 at each of 8 corners).



Step 10: Create weld beads: **Weld Bead** on **Weldments** tab > select a gusset edge > enter 0.125 for radius > ✔ > repeat to create other beads.

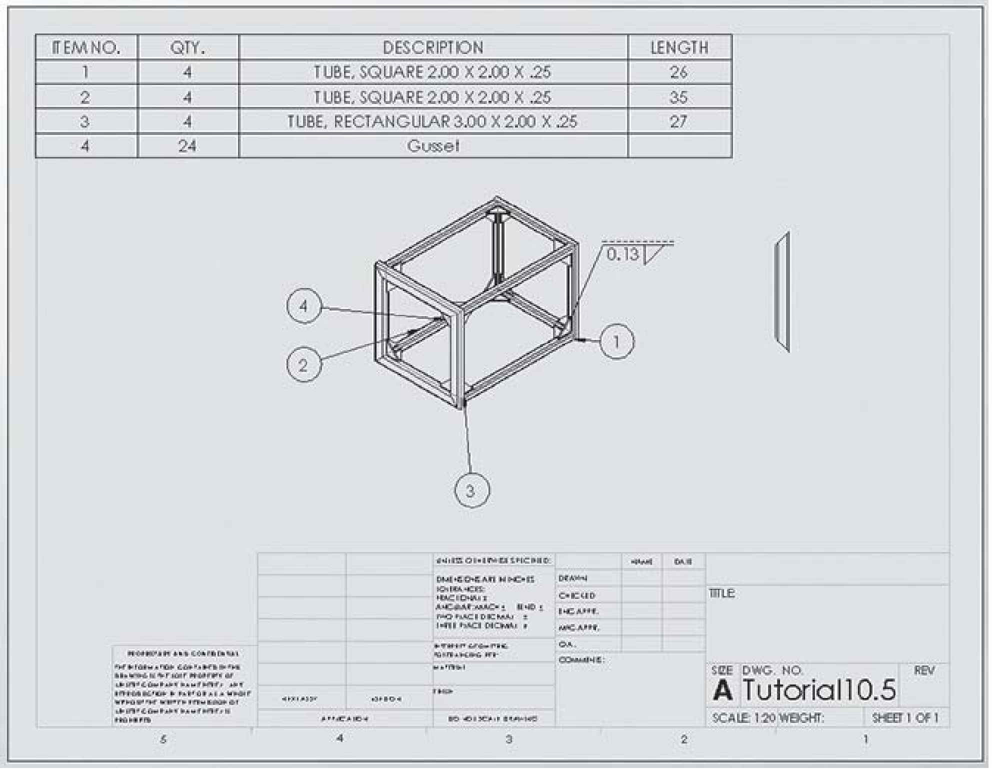


> ***HANDS-ON FOR TUTORIAL 10–5.*** Using ANSI Standard, change the member's cross section **Type** to **pipe** and the **Size** to **1.5 sch 40**. What happens to the model? Fix the problems.

Tutorial 10–6: Create Weldment Drawing

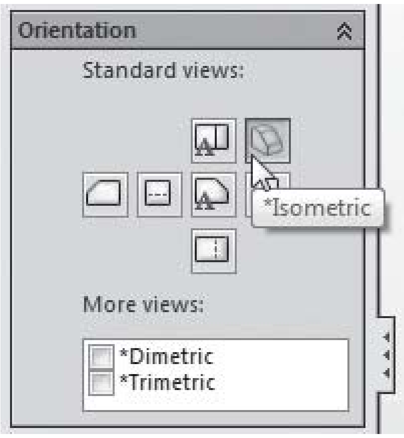
This tutorial covers the basic weldment drawing operations. We create the drawing shown in Figure 10.13 for the weldment created in Tutorial 10–5. The drawing shows a **Relative To Model** view to the right of the ISO view. We need two orientations to fully define the new view relative to the existing model. The first face we select in Step 8 defines the front (by default) of the new view. The second face we select in Step 8 defines the right (by default) of the new view.

FIGURE 10.13
A weldment drawing

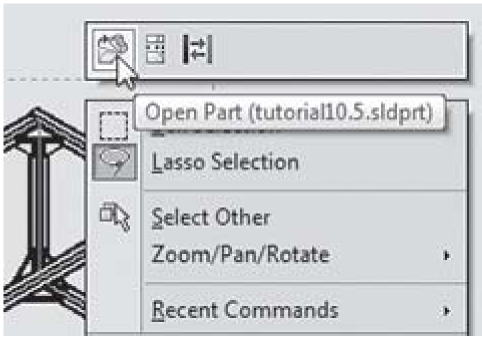


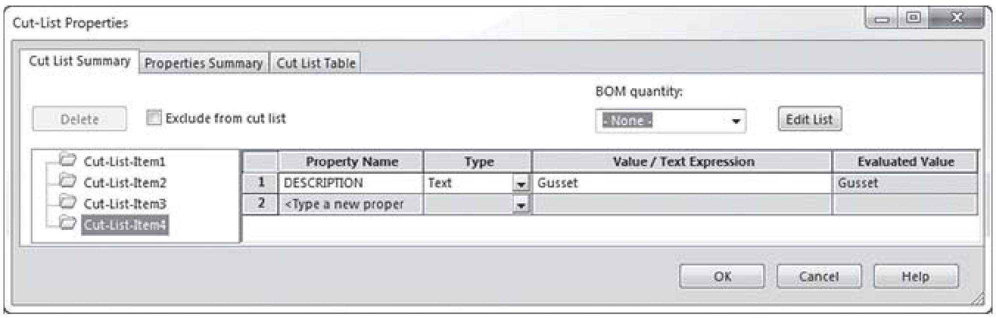
Step 1: Create *drawing*: **File > New > Drawing > OK > OK > File > Save As >** *tutorial10.6* **> Save.**

Step 2: Insert ISO view: **Browse** button > locate *tutorial10.5* part file > **Open** > click **Isometric** icon in **Orientation** section in left pane > click in drawing sheet area to place view > ✔.



Step 3: Return to model temporarily: Right-click ISO view in sheet area > select **Open Part** from the context toolbar as shown here:



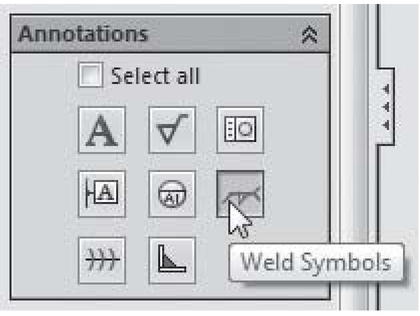


Step 4: Open **Properties** box to modify gusset name: Expand **Cut list (84)** in the features tree (part is open from Step 3) > right-click **Cut-List-Item4(24) > Properties** to open **Cut-List Properties** window shown > select **Text** in **Type** column > type "Gusset" in **Value / Test Expression** column as shown > **OK > File > Close** to return to drawing.

Step 5: Insert weldment cut list table: Click ISO view in drawing sheet >[ ] **Insert** > **Tables** > **Weldment Cut List** > ✔ > move mouse to drawing sheet and snap to top left corner to place the list as shown in Figure 10.13.

Note: You now see the word "Gusset" that you typed in Step 4, the last row of the table.

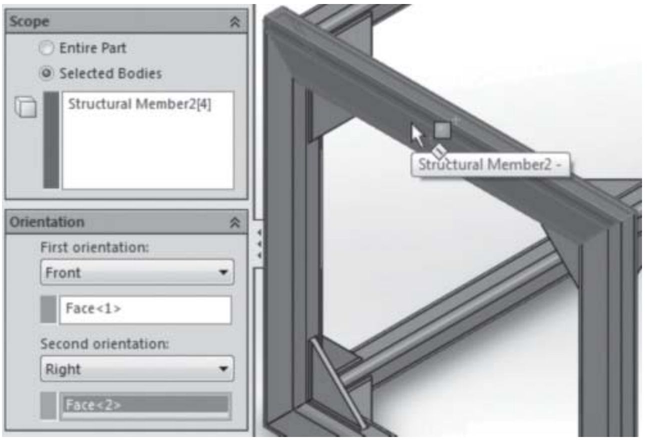
Step 6: Add weld callouts (symbols): **Model Items** on **Annotations** tab > **Selected feature** from dropdown of **Sourc**e on left pane > deselect all options except **Weld Symbols** shown > delete all 0.13 symbols except the one shown on right in Figure 10.13 > ✔.



Step 7: Add balloons: **Auto Balloon** on the **Annotations** tab > click on ISO view in drawing sheet to place balloon as shown in Figure 10.13.

Note: The number inside a balloon refers to the **Item NO.** in the cut list table.

Step 8: Add view of structure member: **Insert** > **Drawing View** > **Relative To Model** > **Selected Bodies** option under **Scope** as shown > top far left structure member > left face of this member as **Face<1>** > top face of this member as **Face<2>** > ✔ > click in sheet to place view as shown in Figure 10.13.



> ***HANDS-ON FOR TUTORIAL 10–6.*** First, manually add the dimensions to the relative view. Then change the properties of the balloons so that the balloons associated with the structural members display the length of each member, and the balloons associated with the gussets display the quantity of gussets.

problems

1. What is so special about sheet metal parts that they cannot be modeled using the conventional features modeling approach?
2. Why should we not use the assembly modeling approach to create welded parts as assemblies?
3. Define sheet metal. What distinguishes it from foil or plates?
4. Select one of the sheet metal processes (punching, rolling, stamping, etc.), and describe how the process works. Provide a report of the process details with sketches.
5. Find out the standard gauges of sheet metal for different materials and their corresponding thicknesses. For example, Gauge 10 for Aluminum is 3 mm.
6. Find typical values for K-Factor, bend allowance (BA), and bend deduction (BD) for sheet metal materials.
7. Calculate the flat length of the sheet metal shown in Figure 10.1. Use the following parameters: material is aluminum, $L_1 = L_2 = 100$ mm, $T = 5$ mm, $R = 10$ mm, $A = 90°$, and the K-Factor for aluminum $= 0.485$.
8. What is the difference between an edge flange and a miter flange?
9. List and describe the four methods to create sheet metal parts.
10. Use the flattened state method, and create the sheet metal parts shown in Figure 10.14 using one flat base flange for both parts (i.e., bend the base in opposite directions as shown).

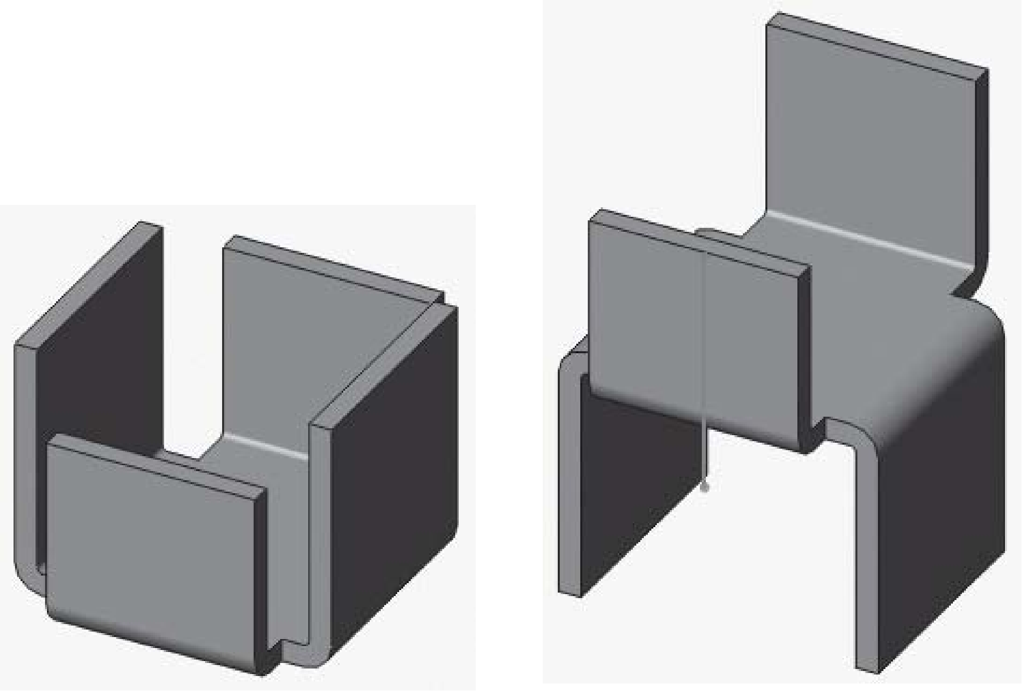


FIGURE 10.14
Sheet metal parts

11. Select one of the welding processes (gas welding, arc welding, resistance welding, etc.), and describe how the process works. Provide a report of the process details with sketches.

12. List and describe three weld defects and why they happen.
13. Many weld joints exist. List four of them. How do you decide which weld joint to use?
14. List and describe the steps to create welded parts.
15. Create the sheet metal frame shown in Figure 10.15. Assume all dimensions.
16. Create the bicycle frame weldment shown in Figure 10.16. All dimensions are in inches.

FIGURE 10.15
Sheet metal frame

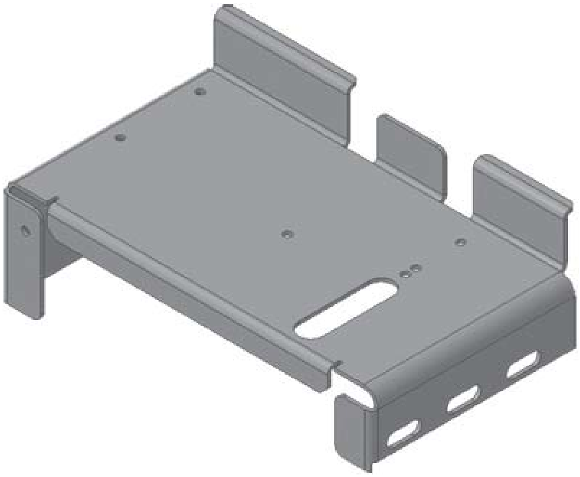
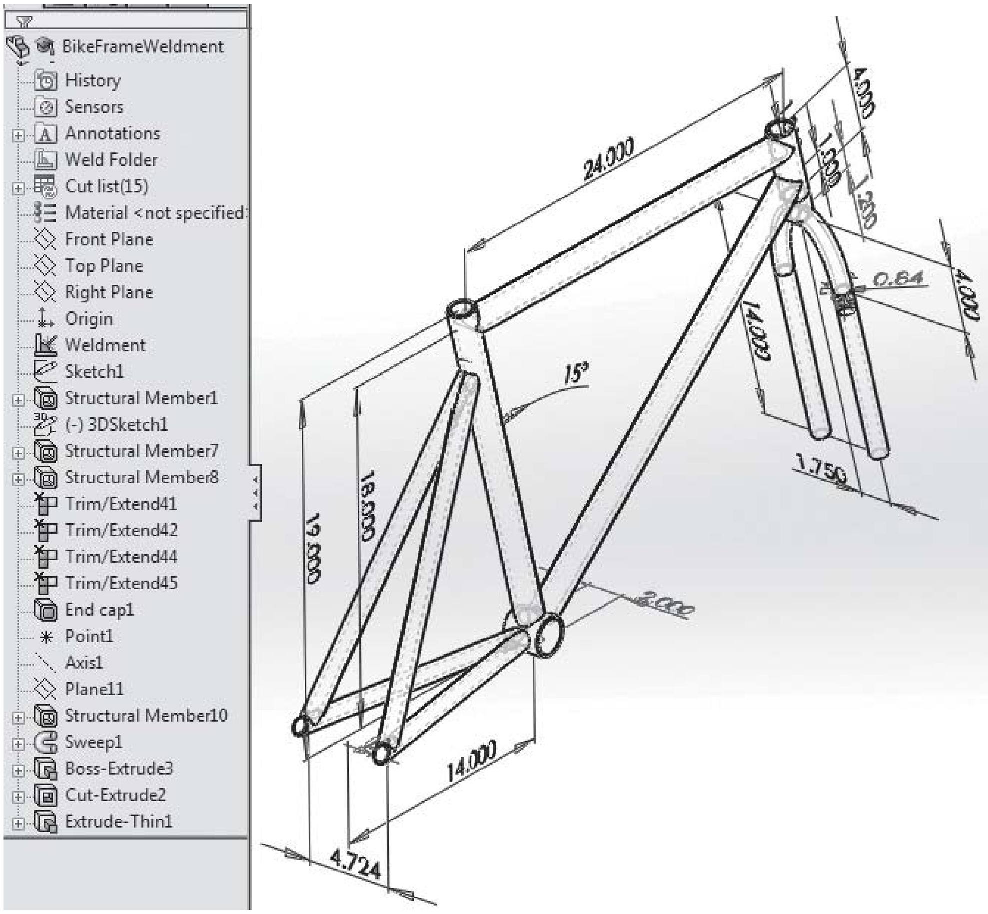
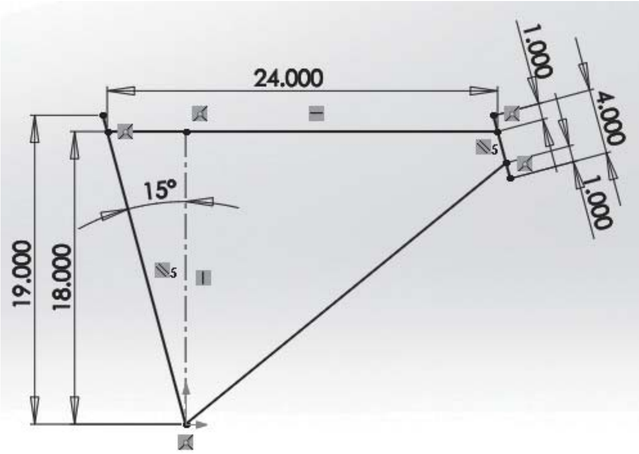


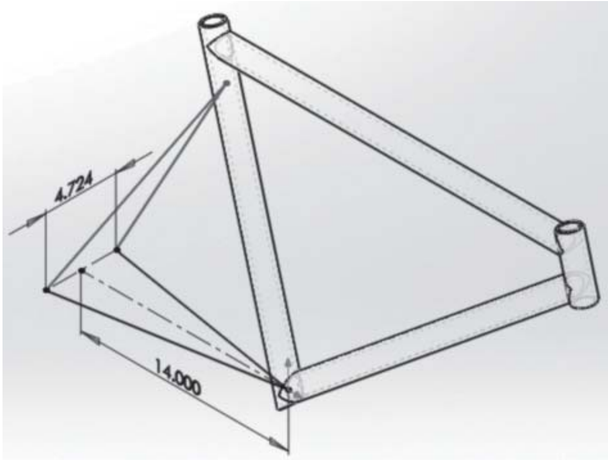
FIGURE 10.16
A bicycle frame



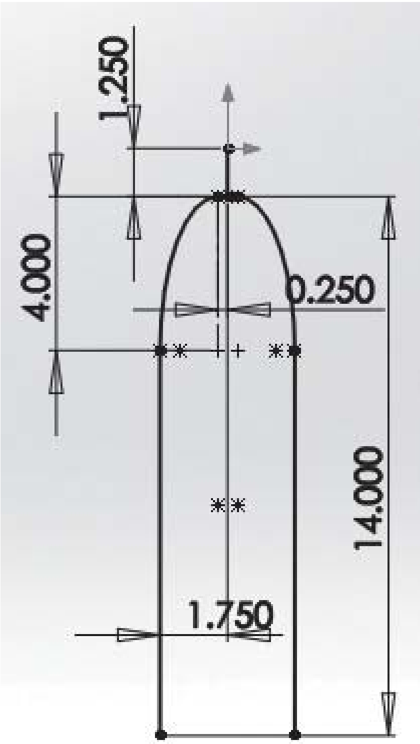
(A) Full frame



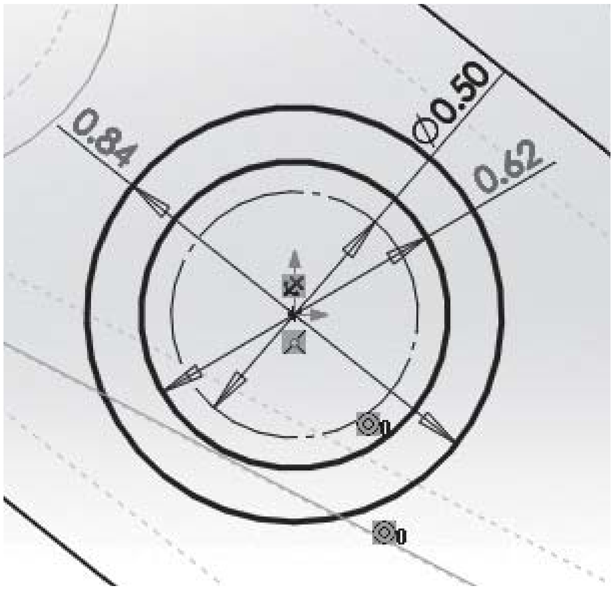
(B) **Front Plane** sketch (skeleton) of main frame



(C) 3D sketch (skeleton) of back wheels



(D) 2D sketch (skeleton) of front wheels; create an inclined plane for it



(E) Frame pipe cross section

FIGURE 10.16
(*continued*)

CHAPTER

11

Sustainable Design

11.1 Introduction

Not too long ago engineering design would focus on in-service requirements such as functional and strength requirements. Today, design must also consider stringent environmental requirements and the handling of products after their "death." The central question is, What do you do with products after they reach the end of their useful life? Even during their useful life, products must conform to limited natural resources and be evaluated especially in terms of energy consumption and pollution.

Sustainable design represents a paradigm shift in how we design products. It is a direct outcome of strict regulations on the environment, the use of natural resources, limited landfills for waste collection, hazardous material, health concerns, and other factors. Designers must think hard about product use and disposal including health effects and environmental impact. A good example is electronics and computer devices. Designers are trying to phase out components that represent health hazards by using new materials that are more environmentally friendly.

Sustainable design addresses two product design concerns: during life (in service) and after life (end of service). During life, a product must meet a host of environmental and energy constraints. For the environment, the product must minimize pollution and the greenhouse effect as much as possible. Some of the pollution measures include carbon dioxide emission, water pollution, and air pollution. Energy consumed by products must also be minimized by making products as efficient as possible.

Designing and producing sustainable products require sustainable design, sustainable manufacturing, and sustainable waste. We cover sustainable design in detail in this chapter. Sustainable design determines to a large extent sustainable waste (fewer emissions and less pollution of the environment). Sustainable manufacturing is controlled by materials used and the manufacturing processes themselves. In practice, the common sustainability factor is energy use in manufacturing and transporting products, as well as the energy the products use while in service.

After life, a product must equally meet a host of disposal constraints. Components of discarded products must be easy to disassemble and sort for refurbishing, recycling, and disposal. We are all familiar with the many products labeled "recycled" or with the recycle symbol. Examples include paper, printer cartridges, and many plastic products. Refurbished (also known as *rebuilt*) components are common in the automotive industry where we may buy a refurbished alternator, water pump, or brake calipers for a car. As for the disposed components, we try to minimize them in product design as much as possible.

Sustainable engineering has been used as a broader context that includes sustainable design. Other terms have been used for sustainable design such as *ecodesign*, *environmentally sustainable design*, *environmentally conscious design*, and *green design*. The latter two terms are borrowed from the manufacturing field where the terms *environmentally conscious manufacturing* and *green manufacturing* have been used to promote less polluting manufacturing methods and products that are easy to disassemble and reuse.

Minimizing energy consumption and using renewable energy (such as wind and solar energy) are important to sustainable design. The less we depend on nonrenewable energy such as fossil fuels (oil and gas), the better we preserve our natural resources. The difference between renewable and nonrenewable energy is the time required to generate the energy. Fossil fuels take millions of years to form because we must wait for the decomposition of buried dead organisms. Fossil fuels contain a high percentage of carbon and hydrocarbon, two excellent sources of energy.

In order for us to learn and perform sustainable design tasks, we need to be aware of and learn the many sustainable aspects including technologies, materials, manufacturing methods, and energy resources. Designers can think only in terms of what they know. For example, the more materials we know (e.g., metals, plastics, ceramics), the better material we can select.

Successful and effective sustainable design tools, like many other engineering tools, are the ones that quantify the sustainability concepts. Designers would like to have design tools that allow them to compare different design alternatives to find the best one. The SolidWorks **Sustainability** module fits this bill. It is a great start in the right direction.

We may ask ourselves this question before we continue: What is sustainable engineering or design? According to the WEPSD (World Engineering Partnership for Sustainable Development), **sustainable engineering** is the melting pot (integration) of three disciplines (fields): environmental, economical, and social. The use of these three fields in design defines **sustainable design**, which makes designers environmentally conscious about their designs. **Environmental sustainability** is defined as meeting the needs of the present without compromising the ability of future generations to meet their needs.

An important source when it comes to sustainability is the Brundtland Report published by the Brundtland Commission, formerly known as the World Commission on Environment and Development (WCED). The commission, created by the United Nations in 1983, began as a response to the growing concern about the deterioration of the global environment and the need for immediate action. The Brundtland Report, also known as *Our Common Future*, was published by Oxford University Press in 1987. The commission report addressed both the need for promoting sustainable development and the need for the change of policies to achieve sustainable development at the world level. For more information, visit en.wikipedia.org/wiki/Brundtland_Commission and www.un-documents.net/a42r187.htm.

11.2 Design and Society

We always practice engineering (in a broader sense) and design (in a narrower sense) within societal contexts. Engineering is inherently tied to society and human needs. Engineering design evolves over time and reflects societal needs and concerns. As society evolves, so does engineering design. Societal impact, in positive ways, is always the main drive behind engineering design.

Over time, societal needs and priorities shift and pull design along with them. Let us consider a few scenarios. When manufacturing cost became a concern in the 1980s and 1990s, there was a push for better ways to make high-quality products that are less expensive. Thus, concepts such as DFA (design for assembly), DFM (design for manufacturing), and concurrent engineering evolved, flourished, and were embraced by the engineering profession. Similarly, the concept of six sigma was conceived and adopted to improve product quality.

When societies across the globe began to push for environmental awareness and concerns, the "green" movement made its way to design. We may use the term *DFS* (design for sustainability) for sustainable design because it extends the broader concept behind DFA, DFM, concurrent engineering, and six sigma; that is, to adapt the engineering design process and its activities to address current societal concerns.

11.3 Guidelines and Principles

Guidelines for sustainable design have been evolving and developing and will continue to do so over time. They are mostly qualitative. The following guidelines are commonly used:

A. Minimize energy consumption: Design products that require less energy to operate. For example, hybrid cars offer high gas mileage. Another example is the prolonging of battery life in many products, and a third example is Energy Star (energy-efficient) household appliances.

B. Use safe materials: Use materials that have low impact on the environment during processing or after discarding products. For example, use nontoxic, recyclable materials that require little energy to process and manufacture. Also, use materials coming from recyclable resources as much as possible such as metal, aluminum, or paper.

C. Use efficient manufacturing processes: Use processes that are fast to complete and for which the equipment consumes less energy. Also, these processes should have minimum pollution and other adverse environmental impacts.

D. Minimize or reduce product carbon footprint to protect outdoor air quality: A **carbon footprint** of a product is defined as the total set of greenhouse gas (GHG) emissions caused by the product. It is expressed in terms of the amount of emitted carbon dioxide equivalent. The word *equivalent* means including methane, carbon monoxide, and other greenhouse gases. Carbon footprint can also be defined for a company, an organization, or a nation. Strategies to reduce carbon footprint of a product include using better process and product management, better energy consumption strategies, and alternative renewable energies. The mitigation of carbon footprints is often known as carbon offsetting.

Carbon footprints exist today for many products. They have been calculated by different organizations and countries. For example, the U.S. EPA (Environmental Protection Agency) has addressed paper, plastic wrappers for candy, glass, cans, computers, carpet, and tires. The U.S. Postal Service has addressed mailing letters and packages. Australia has addressed lumber and other building materials.

E. Use life cycle assessment (LCA): **LCA** is defined as a method to quantitatively assess the environmental impact of a product throughout its entire life cycle, from the procurement of raw materials to production, distribution, use, disposal, and recycling of the product. Design impact measures for LCA for any product use are increasingly required and available. LCA includes many product facets including energy consumption and environmental impact.

F. Reuse and recycling: Products, processes, and systems should be designed for recycling and reuse, that is, design for reuse and recycle (DFR2). We use many products today that are recyclable. Today, recycling is an important part of our personal daily routine and habits. Good examples are soda can and printer cartridge recycling programs.

G. Increase quality and durability: The less frequently we have to replace and repair products, the more energy and resources we save.
H. Share resources: We should design products for public ownership (sharing) instead of private ownership (e.g., carpooling or car rental).
I. Create healthy buildings: Although not a mechanical product, buildings and houses form an important sustainable design problem. We need to design healthy buildings that are harmful neither to their occupants nor to the environment. For example, indoor air quality and energy-efficient buildings are important design goals.

To promote the preceding design guidelines necessary for sustainability, the "Hannover Principles" (also known as the "Bill of Rights for the Planet") were adopted in the EXPO 2000 held in Hannover, Germany. We cast these principles in terms of engineering design as follows:

1. Humanity and nature must coexist in a healthy, sustainable way.
2. Engineering design interacts with and depends on the natural world and, as such, must consider implications on the environment.
3. Engineering design (matter and human elements) must respect and coexist with nature (spirit). As such, engineers must protect and conserve air and water quality.
4. Engineering design is responsible for the consequences of its decisions on human well-being, the viability of natural resources, and their right to coexist.
5. Engineering design should be held accountable for future damage to the environment. Engineers should not burden future generations with potential damage and danger due to careless creation of products, processes, and standards.
6. Eliminate waste. Optimize product life cycle as much as possible to eliminate the concept of waste.
7. Use renewable energy sources as much as possible. Rely on natural energy flow such as solar and wind energies. Minimize the consumption of nonrenewable fossil energy.
8. Engineering design should respect nature as a superpower. No matter how brilliant human design is, it is still limited and does not solve all problems. As such, we must treat nature as a role model and mentor, not as an enemy to conquer and control.
9. Promote communication and knowledge sharing. All involved in LCA and product development should communicate openly to integrate sustainable considerations and integrate natural processes with human design and activity.

In addition to this bill of rights for the planet, in 1993 the World Congress of the International Union of Architects (UIA) and the American Institute of Architects (AIA) signed a "Declaration of Interdependence for a Sustainable Future." Visit server.uia-architectes.org/texte/england/2aaf1.html. The declaration states that today's society is degrading its environment and that the UIA, AIA, and their members are committed to the following:

1. Professionals are responsible for environmental and social sustainability.
2. Promote sustainable design via developing and improving practices, products, services, and standards.
3. Disseminate the value of sustainable design via educating the general public.
4. Help support sustainable design practices by working to change policies, regulations, and standards in government and businesses.
5. Bring the existing built environment and codes to sustainable design standards.

11.4 Life Cycle Assessment

Successful products are developed by integrating LCA directly into the engineering design process. LCA is commonly known as "cradle to grave" analysis. Although this extent of assessment is desirable to achieve, some products may be assessed based on "cradle to gate" or some other variation. Figure 11.1 shows a typical product life cycle. The main stages of the cycle are the procurement of the raw materials through production, distribution, use, disposal, and recycling. The details of these stages are described as follows:

1. **Raw material extraction:** Different procurement and extraction methods exist, depending on the type of material needed. For wood, we plant, grow, and harvest (cut) the trees. For metals, we mine raw ore. For plastics, we drill and pump oil.
2. **Material processing:** This phase converts raw materials into engineering materials ready for use in manufacturing. Wood requires cleaning and slicing the trees into wood slabs. Metals require, for example, converting ore to steel and bauxite into aluminum. Plastics require converting oil to plastics.
3. **Part manufacturing:** Manufacturing processes materials into finished goods or parts. Wood processing makes wooden products such as furniture. Metal processing converts metal to steel columns and aluminum to soda cans. Injection molding and stamping convert plastics to toys.
4. **Assembly:** Finished parts are assembled to create the final product.

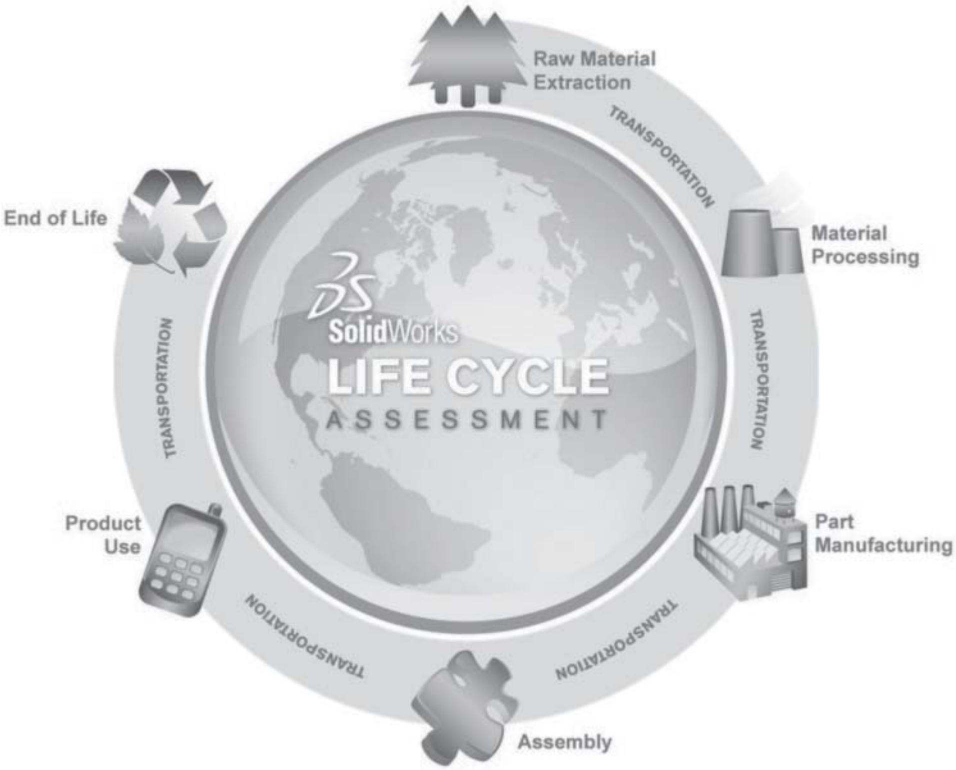


FIGURE 11.1
Product life cycle
(Courtesy of DS SolidWorks Corp.)

5. Product use: Intended end consumers use the products for their intended life span. This use may involve gasoline, electricity, and other resources.
6. End of life (EOL): When a product is discarded (deceased), it should be processed (buried) in an environmentally conscious way. Ideally, we should disassemble the product and decide which parts (components) of the product should be recycled, thrown into landfills for decomposition over time, or incinerated for energy recovery to minimize damage to the environment.

A product LCA evaluates the effects that a product has on the environment over its entire life period, thus increasing resource-use efficiency and decreasing environment liabilities. The LCA has an IQ (Identify and Quantify) and an impact (Evaluate and Assess). Thus, the LCA key elements are as follows:

1. Identify: Find the environmental loads that we need to assess and evaluate. These loads could include raw materials, energy, emissions, and generated waste needed and consumed by the product throughout its life cycle.
2. Quantify: Find and use the appropriate formulas, charts, and other tools to calculate the environmental loads. For example, quantification of carbon footprint for many products already exists, as we have covered in Section 11.3.
3. Evaluate: With quantification numbers at hand, we are able to evaluate the impact of the environmental loads.
4. Assess: Repeat Steps 1 through 3 for available options, and assess these options to reduce the environmental impact.

Unlike human IQ, the lower the IQ for LCA, the more sustainable and environmentally friendly the LCA is.

11.5 Impact Metric

The impact on the environment of making a product comes from three categories: materials, methods, and systems. We associate materials with product design, methods with product manufacturing processes, and systems with product transportation and use during its useful life and EOL. Figure 11.1 clearly shows these three categories. We assert here that material selection is a key factor that determines the level of environmental impact of making a product and therefore determines its degree of sustainability. This is because material selection affects the product making, transportation, use, and EOL fate. For example, material selection (metal or plastic) affects the product manufacturing process (machining or injection molding). Material selection also affects the product logistics and transportation through its different phases from transporting materials, to plants, to shipping product, to end consumers. Some materials may be easier and faster to ship to manufacturing, thus reducing the carbon footprint. Finally, material selection controls the EOL fate of the product.

If the material selection is crucial, how can we then assess the impact of different materials on the environment? In other words, how can we measure the sustainability of a design based on material selection? We define an impact metric with the following five impact factors:

1. Carbon footprint: CO_2 and other gases result from burning fossil fuels. These carbon gases are trapped in the atmosphere, resulting in air pollution and in raising the earth's average temperature. Carbon footprint is measured in kilograms of CO_2 equivalent. Carbon footprint has very serious environmental repercussions and is responsible for the global warming effect that causes loss of glaciers, extinction of species, and extreme weather swings.

2. Energy consumption: This impact factor is measured by the amount of nonrenewable (bad) energy (e.g., petroleum, fossil fuel, coal) that a product consumes throughout its life cycle. This consumption is measured in MJ (megajoules). This impact factor includes all the energies that a product requires to be manufactured, transported, and used during its life cycle. The calculation of the product energy consumption takes into account the efficiencies in energy conversion.
3. Air acidification: In addition to releasing CO_2 into the air we breathe, burning of nonrenewable fuels produces other harmful gases including sulfur dioxide (SO_2) and nitrous oxide (NO_X). These and other acidic emissions to the air result in acid rain. Acid rain has serious consequences on the environment including making land and water toxic for human and animal uses as well as for aquatic life and wildlife. Acid rain also eats into and dissolves building materials such as concrete. The unit of measuring the air acidification is kilograms of sulfur dioxide equivalent (SO_2e) for the CML methodology. Other units of measurement exist for other impact assessment methodologies. The two most commonly used life cycle impact methodologies are CML and TRACI. CML stands for Centrum voor Milieukunde Leiden (a Dutch university). For more information about CML, visit cml.leiden.edu/software/data-cmlia.html. TRACI stands for Tool for the Reduction and Assessment of Chemical and Other Environmental Impacts. For more information about TRACI, visit www.epa.gov/nrmrl/std/traci/traci.html.
4. Water eutrophication: This impact factor measures water pollution that is not related to acid rain, but to dumping societal waste into the water ecosystem. For example, water waste from manufacturing plants and agricultural fertilizers contains high levels of nitrogen (N) and phosphate (also known as phosphorus) (PO_4). Both cause more algae to grow in water, which depletes oxygen from water, causing the death of plants and fish. The unit of measuring water eutrophication is kilograms of phosphate equivalent (PO_4e) for the CML methodology. Other units of measurement exist for other impact assessment methodologies such as TRACI.
5. Water footprint: Worldwide consumption of water has put pressure on the aquatic systems and resulted in dramatic shortage and deterioration of the global water supply, just as the nonrenewable fuels have done to the earth and the atmosphere. Water consumed by companies to make products is an important factor in both company and product sustainability. An analysis of the water footprint of a product can be calculated. The unit of measuring water footprint is cubic meters per year. You can calculate national, corporate, or individual (personal) water footprint.

Interested readers may consult these references for more details about LCA:

A. PE International, a leading company in LCA technology: www.pe-international.com
B. EPA LCA resources: www.epa.gov/nrmrl/std/lca/lca.html
C. German IFEU Research Institute: www.ifeu.de/english
D. German UBA Federal Environmental Protection Agency: www.umweltbundesamt.de
E. Water footprint calculations: www.waterfootprint.org

11.6 Implementation

The implementation of the LCA is guided by ISO: International Organization for Standardization (www.iso.org). ISO has multiple sustainable standards. ISO 14040/44 provides the LCA principles and framework. ISO 14062 is concerned with greenhouse assessment. ISO 14025 deals with Type III eco-label and IPCC national inventories.

The implementation of LCA provides a systematic approach for identifying, quantifying, and assessing environmental impacts through the life cycle of a product, process, or activity. This implementation considers material and energy uses and releases to the environment from "cradle to grave." The goal of LCA use is to identify "hot spots" of potential environmental impact, compare one or more aspects of a product or process, and establish a baseline for further research and comparisons.

LCA is only one of the tools used in environmental decision making. We generally use it in conjunction with other tools such as risk assessment. There are also other life cycle approaches besides LCA. LCA does not necessarily embody every approach called for.

The application and use of LCA require substantial efforts. First, we need to understand the principles and the concepts. Next, we need to quantify the LCA impact factors as we have discussed. Third, we need to collect lots of data about many materials, processes, systems, transportations, regulations in different countries, and other issues. Then, we need to build the LCI (life cycle inventory) database or library that houses all that data. Finally, we write software tools that designers can use during their design activities.

Although LCA can contribute to improved environmental decision making to produce better sustainable designs and products, we offer a word of caution. LCA comes with its own pitfalls. LCA users need to maximize the potential benefits of LCA. Designers and their companies need to be clear on the objectives of their LCA and the measures of success. Designers and their companies need to recognize that LCA provides only one component of a more comprehensive decision-making process, and that understanding the trade-offs between required resources and the LCA demands is important. After all, our goal as engineers and designers is to make products that are both affordable and sustainable.

11.7 Design Activities

Thus far, we have a set of sustainability concepts that are ready for use in our design activities. These concepts are as follows:

A. We have identified materials, processes, and systems as key elements that affect sustainable design.

B. We have concluded that materials are the decisive and driving element in sustainable design because they directly control the choices of processes and systems.

C. We have also identified the impact factors that affect the environment significantly.

D. We have quantified the impact factors, meaning we can compute their environmental impact.

We now need to put these concepts in a systematic and logical order to provide designers with tools to enable designers to perform LCA, and create and evaluate sustainable designs. The following steps embody this order:

1. Create part design: Create a part as we have covered so far in the book.
2. Select material: Assign the part a material.
3. Select manufacturing processes: Based on the material selection, decide on the appropriate manufacturing processes.
4. Select system: This includes the part transportation and the intended region of the world where the part is to be used.
5. Set baseline: Calculate the impact factors covered in Section 11.5. This sets a baseline for sustainable design. We use this baseline to compare the selections of different materials.

6. Repeat Steps 2 through 5: Calculate the impact factors for the new material selection.
7. Repeat Step 6 as many times as needed: Do this until a satisfactory sustainable design is achieved. If needed, redesign the part (Step 1).

11.8 Sustainable Design Tools

Tools for sustainable design are particularly important for both academia and industry. These tools are most useful when they are quantitative to enable designers in practice to evaluate their sustainable design alternatives and options. These tools are also useful for educating students, future engineers, of the importance of sustainability and making them aware of the environmental impact of engineering design.

Two of these tools are available in the form of software. They are offered by SolidWorks and PE International. SolidWorks offers two sustainability modules: **SustainabilityXpress** and **Sustainability**. They both evaluate the environmental impact of a design throughout the life cycle of a product. You can compare results from different designs to ensure a sustainable solution for the product and the environment. **SustainabilityXpress** analyzes parts only and is a standard option with SolidWorks core software. **Sustainability** provides more comprehensive in-depth analysis (more impact factors and expanding reports) and provides designers with more control and the ability to analyze both parts and assemblies (including configurations). These two modules are similar to SolidWorks FEM/FEA modules: **SimulationXpress** and **Simulation**.

PE International offers comprehensive sustainability solutions. Developed in-house over the course of thousands of sustainability consulting projects, PE International's comprehensive sustainability software solutions are designed to support cost-effective corporate and product sustainability projects. PE's GaBi sustainability software offers practicing engineers effective tools to aid them in their sustainable design tasks. Interested readers may visit www.gabi-software.com/america/solutions/life-cycle-assessment/?gclid=CPiowIzC4rwCFTIV7Aodsg0Apg

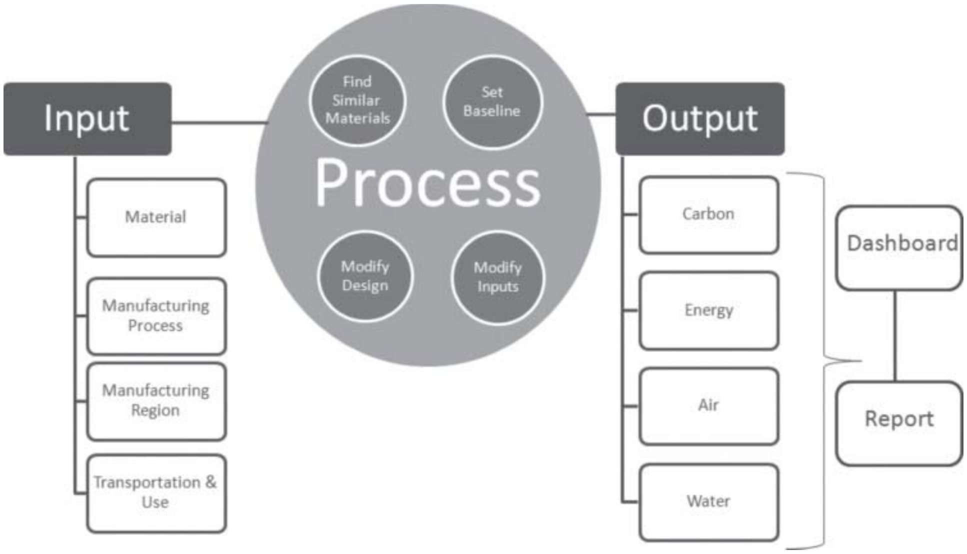
11.9 SolidWorks SustainabilityXpress

SolidWorks implements the concepts of sustainable design in its **SustainabilityXpress** module. It uses only the first four impact factors (carbon footprint, energy consumption, air acidification, and water eutrophication) of Section 11.5. SolidWorks generates a report for the impact factors. SolidWorks **SustainabilityXpress** measures the environmental impact based on material used, manufacturing process and region, use region, and EOL.

SolidWorks **SustainabilityXpress** has incorporated extensive libraries of materials, manufacturing processes, and transportations from PE International (www.pe-international.com). Designers select from these libraries to investigate various sustainable scenarios. Moreover, SolidWorks has a library of manufacturing regions to select a region from. Different regions have different environmental regulations and laws. For example, designers may select a manufacturing region and/or transportation/use region.

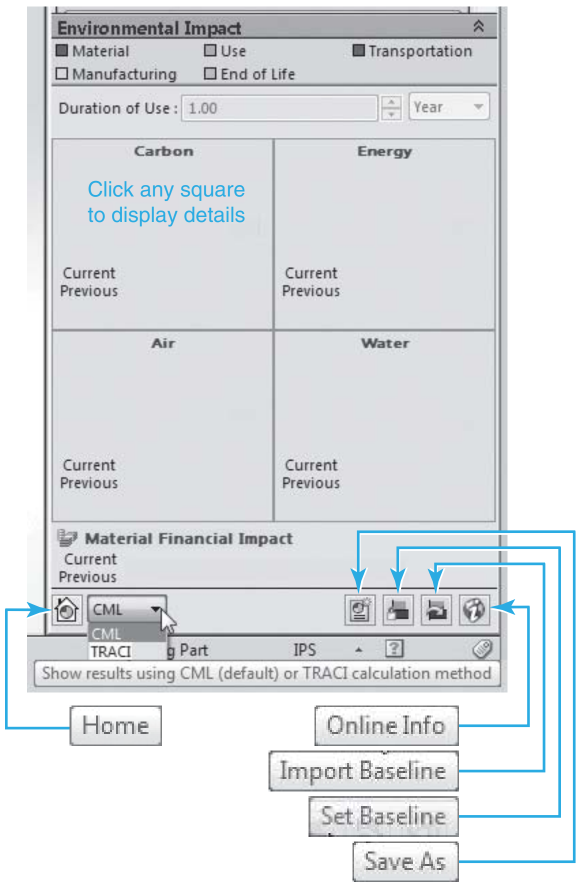
SolidWorks **SustainabilityXpress** follows the design steps covered in Section 11.7. In addition, SolidWorks **SustainabilityXpress** provides a dashboard to display and compare results. It also enables the designer to print reports of the impact factor calculations and the sustainable design. Figure 11.2 shows SolidWorks sustainability methodology.

FIGURE 11.2
SolidWorks sustainability methodology
(Courtesy of DS SolidWorks Corp.)



Click this sequence to access the **SustainabilityXpress** module: **Tools > SustainabilityXpress** (see Figure 11.3). Figure 11.3A shows the three key elements that affect sustainable design: material, manufacturing, and transportation and use and End of Life (or *systems*, as we call it). Figure 11.3B shows the dashboard that displays the calculation results for the four impact factors (the impact metric as we call it). Both Figures 11.3A

(A) Sustainability categories



(B) Impact factor dashboard

FIGURE 11.3
SolidWorks **SustainabilityXpress** user interface

and 11.3B are blank; yet after a designer specifies the three key elements (Figure 11.3A), the dashboard areas (Figure 11.3B) populate with the results. Use the **Save As** icon shown in Figure 11.3B to generate sustainability reports.

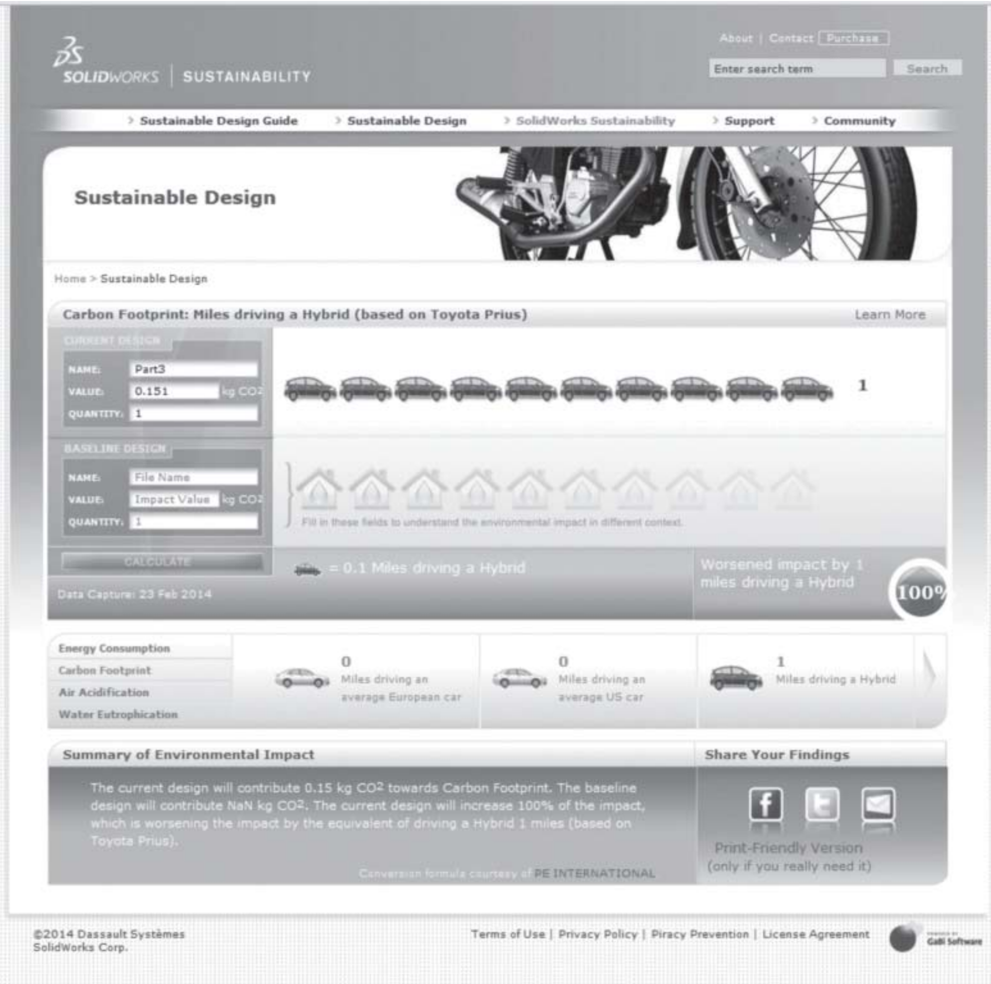
One interesting aspect of SolidWorks sustainability calculations is that they take into consideration the region where the product is made (manufacturing and transportation) and used (use). Both of these areas are affected by a host of regional issues such as cost, environmental regulations, natural resources, type of energy used (fossil, nuclear, hydroelectric, etc.), transportation infrastructure, traffic patterns, and cultural habits. Sample regions that SolidWorks has implemented are North America, Europe, Asia, Japan, South America, Australia, and India.

SolidWorks sustainability analysis begins with selecting the key elements and ends with the values of the environmental impact factors. The designer may perform parametric studies ("what if" scenarios) to find the "best" sustainable design. The steps of this sustainability analysis are as follows:

1. Create the CAD model of the part to be designed.
2. Select the material class and name. SolidWorks supports a wide variety of material classes (e.g., steel, iron, aluminum, copper, titanium, zinc, plastics, fibers, silicon, and wood). Within each class are multiple materials. For example, the class of plastics includes ABS, PVC, nylon, and many others. Check **SustainabilityXpress** for a full list of classes and materials.
3. Select a manufacturing process. For each material class, possible manufacturing processes that can manufacture this class are supported by SolidWorks. For example, injection molding and extrusion processes exist for plastics. For the aluminum class, many processes exist including die casting, extrusion, milling, turning, forging, and stamping. Check **SustainabilityXpress** for a full list of available manufacturing processes.
4. Select the manufacturing region. The region selection determines the type of energy used (how much clean renewable energy and how much nonrenewable energy) and the resources consumed by the manufacturing processes.
5. Select the transportation and use region. A region selection here determines the environmental impact associated with transporting the product from its manufacturing location to its use location. It also affects the energy resources needed by the product while in use and its EOL destiny (landfill, incineration, and/or recycling).
6. Calculate environmental impact. SolidWorks **SustainabilityXpress** uses the input from Steps 2 through 5 and calculates the four impact factors: carbon footprint, energy consumption, air acidification, and water eutrophication.
7. Set baseline. You need to explicitly identify the results from Step 6 as baseline. SolidWorks **SustainabilityXpress** uses this baseline as a reference to compare multiple sustainable designs. If you want to use different results for baseline, you may import them as shown in Figure 11.3B.
8. Redesign. If the environmental results in Step 6 are high, select a new material (Step 2) and repeat Steps 3 through 6. Make design changes as necessary.
9. Evaluate the results. Use the baseline established in Step 7 to evaluate the results from Step 8.

SolidWorks **SustainabilityXpress** provides an online calculator that enables you to convert the environmental impact, as measured by the values of the four impact factors, into human measurable parameters. For example, convert carbon footprint into miles driven by a car. Click the **Online Info** icon shown in Figure 11.3B to access the online calculator shown in Figure 11.4.

FIGURE 11.4
SolidWorks Sustainability online calculator



11.10 Tutorials

The theme for the tutorials in this chapter is to practice using sustainable design concepts implemented by SolidWorks **SustainabilityXpress**. The following tutorial helps assess the impact of a design on the environment.

Tutorial 11–1: Redesign a Steel Washer

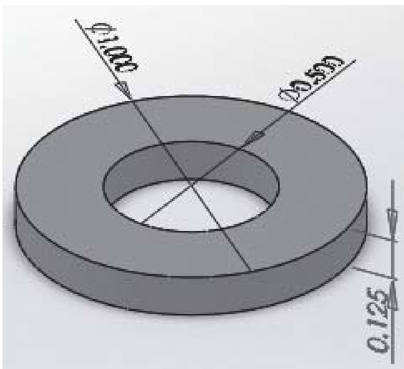


FIGURE 11.5
A typical washer

Figure 11.5 shows the dimensions of a washer (OD, ID, and thickness in inches) made out of carbon steel. The washer must be replaced by another material such as aluminum or plastic to meet new EOL environmental constraints of recycling. Investigate which design is better for the environment. The washer is used in North America. Also, currently the steel washer is manufactured in Asia. The new washer could be manufactured in Asia, Europe, or Japan. Which of these regions is the best to produce the new washer?

We need to perform sustainable design evaluation on the washer to find out which new material is better (aluminum or plastic) and which manufacturing region is better. The combination of the two materials and three regions produces a total of six design scenarios to consider. We can reduce the scenarios by half as follows. First, we decide which material is better while using the current manufacturing region (Asia). Then, we use this material to compare the other two regions (Europe and Japan).

We follow Steps 1 through 9 of Section 11.9 to evaluate the design. We use the steel washer to establish the baseline for the sustainability analysis.

Step 1: Create washer feature: **File > New > Part > OK > Top Plane > Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > click origin and drag to sketch and dimension two concentric circles shown in Figure 11.5 > exit sketch > enter 0.125 for thickness **D1** > reverse extrusion direction to extrude down, not up > **File > Save As >** *tutorial11.1* > **Save.**

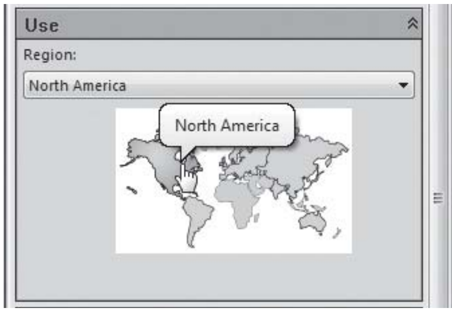
Step 2: Select material: **Tools > Sustainability Xpress** to open **Sustainability Xpress** task pane to right of screen > **Steel** from **Class** dropdown > **1023 Carbon Steel Sheet (SS)** from **Name** dropdown.

Step 3: Select manufacturing process: **Stamped/ Formed Sheetmetal** from **Process** dropdown menu.

Note: Metal washers can be made with stamping, sheet metal, or other processes.

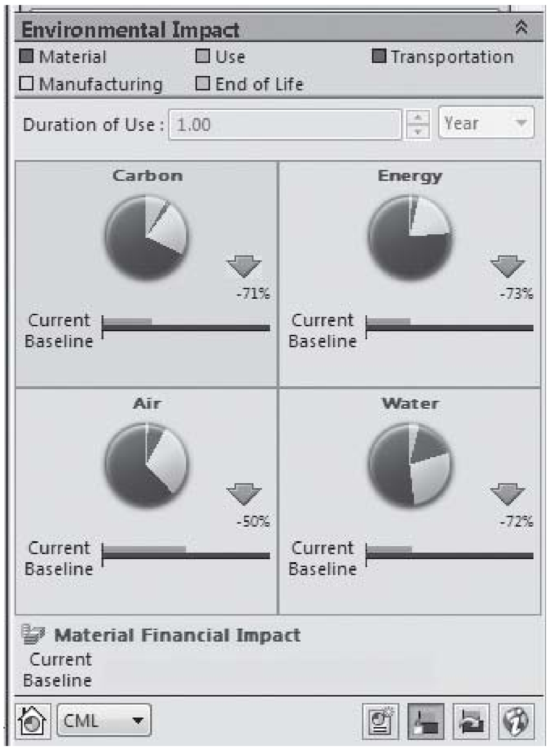
Step 4: Select manufacturing region as **Asia**: Hover over the map and click **Asia** as the manufacturing region.

Step 5: Select use region as **North America**: Hover over the map and click **North America** as the use region of the washer.



Step 6: Set baseline of the study: After Step 5, the four environmental impact factors are calculated and displayed > click **Set Baseline** icon (see Figure 11.3B) to establish the baseline.

Step 7: Evaluate the aluminum washer: Repeat Steps 2 and 3, using **Aluminum Alloys** as Class, **Alumina** as **Name**, and **Stamped/Formed Sheetmetal** as **Process**. Keep Steps 4 and 5 the same. The screenshot shows that aluminum is better than steel.



Step 8: Evaluate the plastic washer: Repeat Steps 2 and 3, using **Plastics** as Class**, Epoxy unfilled** as **Name**, and **Injection Molded** as **Process**. Keep Steps 4 and 5 the same. Knowing that aluminum is better than steel, we set aluminum as baseline and compare plastic to it. The environmental impact (not shown here) shows that aluminum is better than plastic.

Step 9: Evaluate impact of manufacturing region: Now that we have decided on aluminum material, change manufacturing region to Japan, then Europe and compare. The results (not shown here) indicate insignificant difference.

HANDS-ON FOR TUTORIAL 11–1. Investigate the washer design with two new materials: copper and rubber. Which design is better for the environment? Is either of the new materials better than aluminum?

problems

1. What is the core idea behind sustainable engineering?
2. List the three possible venues for disassembled components of a product at its end of life.
3. What is the difference between renewable and nonrenewable energy? Give examples of each.
4. Explain the core concept behind each of the following design paradigms: DFA, DFM, DFS, and concurrent engineering.
5. List and describe five of the sustainable design guidelines.
6. List and describe the Hannover Principles.
7. List and describe the elements of the “Declaration of Interdependence for a Sustainable Future.”
8. Define product LCA.
9. List and describe the stages of an LCA.
10. List and describe the key elements of an LCA.
11. List and describe the impact metric (factors) of an LCA. How is each factor measured?
12. List and describe the design steps (activities) that a designer needs to follow to create and evaluate sustainable designs.
13. What are the input parameters that SolidWorks requires to calculate the environmental impact factors?
14. What are the four environmental impact factors that SolidWorks uses to evaluate sustainable designs?
15. Use SolidWorks to calculate the environmental impact of the flange model shown in Figure 2.32 of Chapter 2. Use steel for material, turning to produce the flange, Asia for production region, and Asia for use. Submit the values for the four impact values and the sustainability report.
16. Redo Problem 15 but for the L bracket shown in Figure 2.35 of Chapter 2. Use aluminum for material, stamping to produce the bracket, Japan for production region, and Japan for use.
17. Redo Problem 15 but for the pattern block shown in Figure 2.36A of Chapter 2. Use steel for material, milling to produce the block, Europe for production region, and Europe for use.
18. Redo Problem 15 but for the pattern wheel shown in Figure 2.36B of Chapter 2. Use steel for material, milling to produce the hub, Asia for production region, and North America for use.
19. Perform a sustainability analysis on the bolt shown in Figure 4.17 (Tutorial 4–8) of Chapter 4. Run the following parametric studies (for all the studies, US is the use region):
    - **A.** Use two different materials: steel and aluminum (fix the manufacturing region to Asia).
    - **B.** Use Asia and US as two manufacturing regions (fix the material to steel).

    Which design of A and B is a more environmentally friendly design? Why?

This page intentionally left blank

PART

IV

Part Development and Analysis

The primary goal of Part IV is to explore and cover the design activities that follow the completion of CAD (geometric) models. During these activities, we use various engineering analyses and simulations to ensure that designs are safe and meet strength requirements.

Chapter 12 (Tolerances) covers the details of tolerances, their types, and their impact on manufacturing. Chapter 13 (Analysis Tools) covers the most widely used analysis and simulation tools. This includes mass property calculations, animation and motion analysis, flow simulation, and finite element modeling and analysis. Chapter 13 also covers data exchange between CAD/CAM systems.

This page intentionally left blank

CHAPTER 12

Tolerances

12.1 Introduction

We all have heard the saying "life is never perfect," and so it is with manufacturing. We can never manufacture a perfect form. An example of a perfect form is a block with $5 \times 3 \times 4$ inch dimensions. Instead, variability is an inherent characteristic of manufacturing. Sources of variability are abundant and include the skills of the machine operator (machinist), the accuracy and age of the machine, the ambient conditions, and the condition and age of the cutting tool. Therefore, producing parts with perfect dimensions would be impossible and prohibitively expensive. Thus, we attempt to control manufacturing variability rather than entirely eliminate it.

In addition to controlling physical attributes of manufacturing such as training operators well and maintaining the machines and cutting tools, we also specify and control the acceptable range of variability of part dimensions. We use the concept of tolerances during part design to specify the acceptable variations for part dimensions that still allow the part to perform its intended function. A **tolerance** is an amount of deviation (variability) in a part dimension from the perfect form. For example, we may specify a tolerance amount of ± 0.010 inch on the $5 \times 3 \times 4$ inch dimensions of the block. This means that any of the block dimensions is allowed to change within this tolerance. For example, the 5 inch dimension may assume any values between 4.090 and 5.010 inches inclusive.

Tolerances play a crucial role in design and manufacturing. Designers use them to meet functional requirements; for example, they specify clearance between two moving parts. For manufacturing, tolerances control the manufacturing cost of parts. The smaller (tighter) the tolerances, the more expensive it is to make the part. Tolerances also have an effect on the part surface finish. The tighter the tolerances, the better the surface finish. This is because tighter tolerances require more precise and better manufacturing processes.

Tolerances are the key behind the concept of spare (replacement) parts or interchangeability. Without tolerances, we would not be sure that a spare part (e.g., a car part) would fit perfectly. We may ask the following question: When we replace a part, why does the new part sometimes feel a little tighter or looser than the old one? Also, different copies of the same new product may feel different; some are tighter than others. This is because the tolerance we specify on a part dimension generates a normally distributed population of the part. The mean of the normal distribution is the perfect form (i.e., the dimension without tolerances on it), for example, the 5 inch dimension of the block. Thus, depending on where a product copy falls in the distribution population, we get tighter or looser products when we assemble the parts.

Geometric dimensioning and tolerancing (GD&T) provide many other benefits in addition to what has been mentioned. These benefits are as follows:

1. Having standardized design language for communication among designers.
2. Clearly communicating the design intent to designers, customers, suppliers, manufacturing, inspection, and the entire product supply chain.
3. Calculating the worst-case assembly scenario by using tolerance limits.
4. Making production and inspection repeatable via using datums (covered later in the chapter).
5. Assembling parts to create products is guaranteed because all the parts belong to their respective normally distributed populations.

12.2 Types

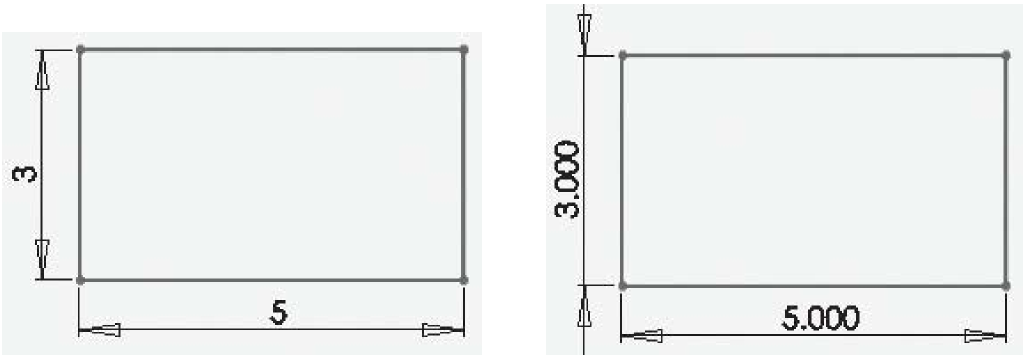
Tolerances are divided into two types: conventional and geometric. Conventional tolerances are more widely used than geometric tolerances because they are the first to develop and they are easier to apply and understand. Conventional tolerances control the part size (dimensions). Geometric tolerances (GTOL) control the part form (i.e., its shape). Consider a rectangle. Controlling the width and height is achieved via conventional tolerances, whereas controlling the angles of the rectangle and the parallelism of its sides is achieved via geometric tolerances.

Both types of tolerances are used for part inspection after manufacturing. There exist off-the-shelf inspection gauges that are used to check whether the parts meet the tolerance requirements, that is, whether their dimensions are within the specified values (limits). A commonly used inspection gauge is the Go-NoGo gauge. If the part fails inspection, it is rejected by the inspection team and becomes scrap. A high percentage of scrap makes the part manufacturing more expensive and calls for changes to better control the manufacturing processes. Inspection is an element of the part quality control (QC) and quality assurance (QA). Other elements include visual inspection and surface finish (roughness) measurements.

12.3 Concepts

Both conventional and geometric tolerances introduce concepts and definitions that we must understand to use tolerances correctly in our designs and to be able to use the tolerance tools offered by CAD/CAM systems. The concepts are covered in the ASME Y14.5M-1994 publication titled "Dimensioning and Tolerancing"; the new revision is ASME Y14.5-2009. We cover the following concepts here:

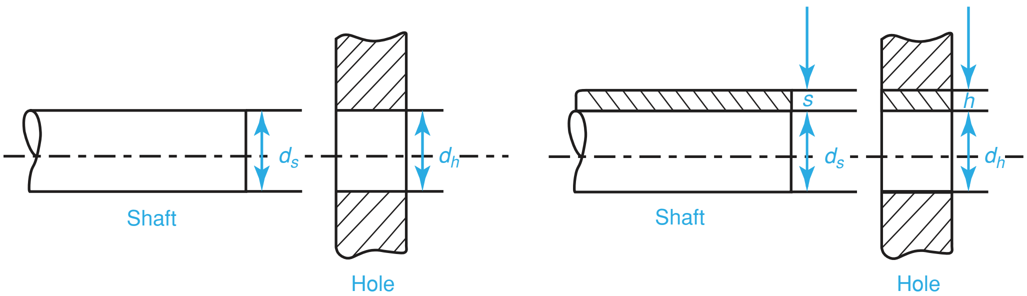
1. Size: A **size** refers to a dimension of a feature, for example, length of an edge, or radius/diameter of a shaft or a hole. Three types of sizes are used in conjunction with tolerances: nominal, basic, and actual. A **nominal size** is a size (value) without decimals (e.g., 5). A **basic size** is a nominal size with decimals to indicate its accuracy (e.g., 5.0, 5.00, or 5.000). Note that these three basic sizes are not the same although their nominal size (5) is the same. They require dimensional accuracy of one, two, and three decimal places, respectively. An **actual size** is the measured length on the manufactured part of the basic size. The actual size must be a value within the specified tolerances of the basic size; otherwise, the part is rejected during inspection. Figure 12.1 shows the nominal and basic sizes.
2. Shaft and hole: The tolerance concepts are based on a shaft fitting into a hole. Each has a basic size of d_s for the shaft and d_h for the hole. Although $d_s = d_h$, each has a different tolerance. Figure 12.2A shows a shaft and a hole.



(A) Nominal size (B) Basic size

FIGURE 12.1
Nominal and basic sizes

3. Tolerance zone: The amount of change in d_s and d_h is known as the **tolerance zone** s and h, respectively. This amount is related to the specified tolerance on the shaft or the hole and can be determined from that tolerance. Figure 12.2B shows the two tolerance zones.
4. Hole and shaft systems: Both the dimensions of a shaft or a hole are subject to variability during manufacturing. In order to specify and calculate tolerances, we need to fix one relative to the other. We may use either as a reference. If we use the hole as a reference, we have a hole-based system for specifying and calculating tolerances. A shaft-based system uses the shaft as a reference. The hole-based system is more widely used in practice over the shaft-based system because it is believed that manufacturing a hole is more difficult than manufacturing a shaft. Thus, it is beneficial to fix the hole size and change the shaft size to achieve the desired tolerance. An example application that uses the shaft-based system is the textile industry.
5. Limit dimensions: A **limit dimension** is the maximum or minimum value of a basic size. For example, the limit dimensions of a basic size of 3.000 and tolerance of ±0.005 are 3.005 (maximum) and 2.995 (minimum).
6. Unilateral tolerances: A dimension may change in one direction from the basic size or in both directions. A **unilateral tolerance** is a tolerance in which variation is permitted in only one direction from the basic size. For example, the limit dimensions of a basic size of 3.000 and a unilateral tolerance of +0.005 are 3.005 (maximum) and 3.000 (minimum). And, the limit dimensions of a basic size of 3.000 and a unilateral tolerance of −0.005 are 3.000 (maximum) and 2.995 (minimum). In both cases, we may think that the other tolerance is 0.000.



(A) Shaft and a hole (B) Tolerance zones s and h

FIGURE 12.2
Shaft and hole tolerance zones

7. Bilateral tolerance: A **bilateral tolerance** is a tolerance in which variation is permitted in both directions from the basic size. For example, the limit dimensions of a basic size of 3.000 and a bilateral tolerance of +0.005 and −0.003 are 3.005 (maximum) and 2.997 (minimum).
8. Symmetric tolerance: A **symmetric tolerance** is a bilateral tolerance with equal variations in both directions from the basic size. For example, a basic size of 3.000 with a symmetric tolerance of ±0.005 has the limits of 3.005 (maximum) and 2.995 (minimum).
9. Material condition: The **material condition** is a condition we attach to a geometric tolerance to indicate to the inspection department how the part feature should be inspected. The material condition is an important concept when designing design inspection gauges. The design ensures that a part will not pass inspection unless its dimensions meet the specified tolerances under the specified material condition. Three types of material conditions exist: maximum, least, and regardless of feature size. A maximum material condition (MMC) is designated by the symbol of the letter M inside a circle. The MMC ensures that the maximum amount of material stays in the part after manufacturing, that is, minimum hole size and maximum shaft size. A least material condition (LMC) is the opposite of MMC. It is designated by the symbol of the letter L inside a circle. The LMC ensures that the least amount of material stays in the part after manufacturing, that is, maximum hole size and minimum shaft size. A regardless of feature size material condition (RFS) is designated by the symbol of the letter S inside a circle. The RFS indicates, to the part inspector, that the tolerance applies regardless of the actual produced (manufactured) size of the feature. Applicability of MMC, LMC, and RFS is limited to features subject to variations in size. The MMC is the one most commonly used in practice because it minimizes the amount of manufacturing scrap, thus saving material and keeping the manufacturing cost down.
10. Datum: Datums are used in conjunction with geometric tolerances. A **datum** is a part feature (plane) that is used as a reference to specify geometric tolerances. The part inspector uses the datums during inspection to check whether the manufactured part is within the specified tolerances. The datum concept is required to be able to control the form (shape) of a part. For example, we need a datum against which we can measure the perpendicularity or parallelism of a part face. A datum is usually an actual face of the part. To use a part face as a datum requires that the machinist uses a highly accurate machining process to machine the face. In addition to controlling form, the datum is also used to measure part linear dimensions from.
11. Datum target: Datum targets define how we define a target. As a datum is a part plane, how do define a plane? We define a datum via three noncollinear points, or via a line and a point not lying on it. Datum targets have physical significance during inspection. The inspection department uses the targets to design and produce the gauge that is used to inspect the part. When the part inspector places the part on the inspection gauge, its faces must make contact with the inspection gauge at the target points and/or lines to pass inspection; otherwise, it does not pass and becomes scrap.

12.4 ASME Tolerance Rules

We briefly discussed ASME tolerances in Chapter 5. Here are some ASME tolerance rules:

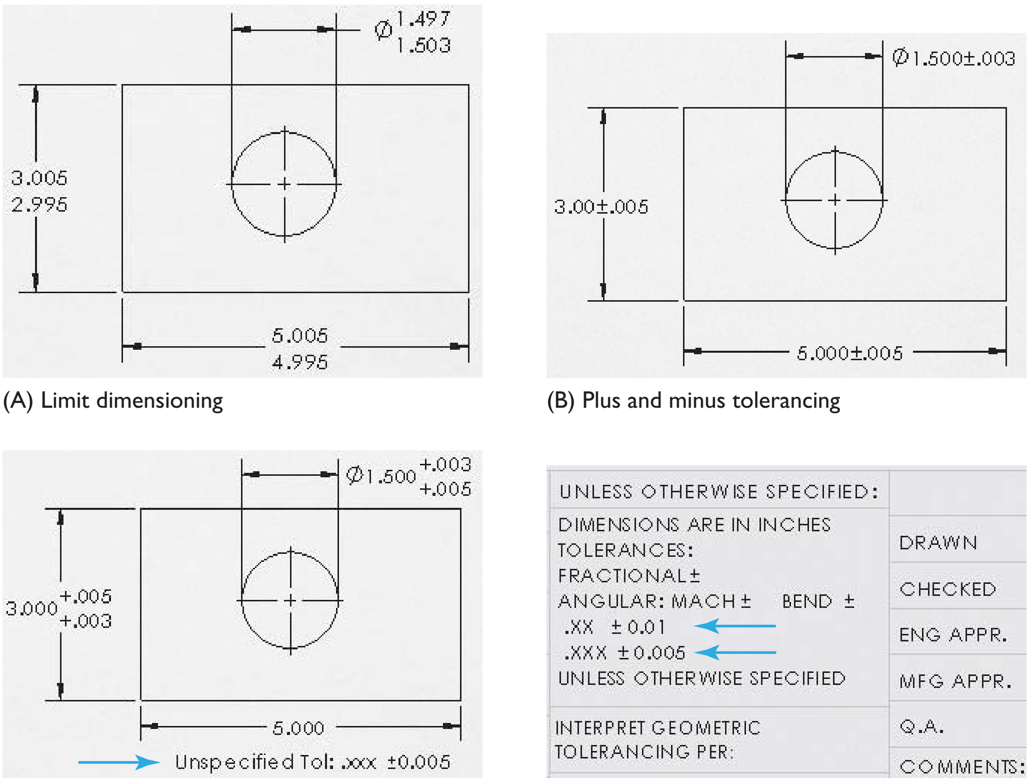
1. Express tolerances correctly: There are four methods to specify tolerance in an engineering drawing:
   A. Limit dimensioning: Show the minimum and maximum values of a dimension. Use the MMC to show the limits, as shown in Figure 12.3A. Show the

maximum value above the minimum value for a shaft and vice versa for a hole. This minimizes the amount of scrap. Keep in mind that the machinist can only remove material, but the material cannot be added once it has been removed. Also, keep in mind that the machinist will attempt, naturally, to meet the dimension value above the dimension line first.

B. Plus and minus tolerancing: Show the basic size followed by a ± tolerance value, as shown in Figure 12.3B. A plus and minus tolerance may be bilateral or symmetric. This method of tolerancing is practiced most widely because machinists prefer it. A machinist typically aims to meet the basic size (dimension) value during machining.

C. As a note referring to specific dimension: Include a note in the drawing to specify tolerances on specific dimensions, as shown in Figure 12.3C.

D. In the title block: Specify general tolerances in the title block of the engineering drawing, as shown in Figure 12.3D.



(A) Limit dimensioning

(B) Plus and minus tolerancing

(C) Tolerance note

(D) Title block

FIGURE 12.3
Methods of specifying tolerances in an engineering drawing

2. Ensure tolerance accuracy matches the dimension decimal: The number of decimals in the basic size dictates the accuracy of the tolerance specified for that size. For example, a basic size of 3.500 should have a tolerance of ±0.005. And a basic size of 3.50 should have a tolerance of ±0.05.

3. Follow these rules for millimeter (metric) tolerances:

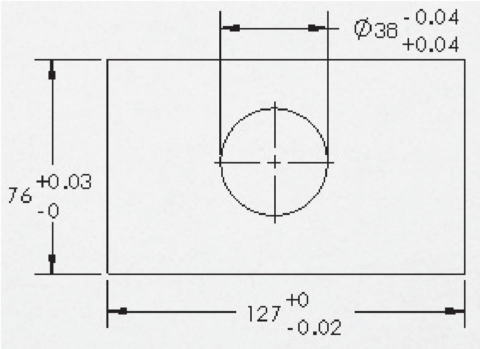
A. For unilateral tolerancing, use a single zero as shown in Figure 12.4A. Note that SolidWorks demonstrates this rule as shown in Figure 12.4A.

B. For bilateral tolerancing, you should use the same number of decimal places, using zeros when necessary, as shown in Figure 12.4B.

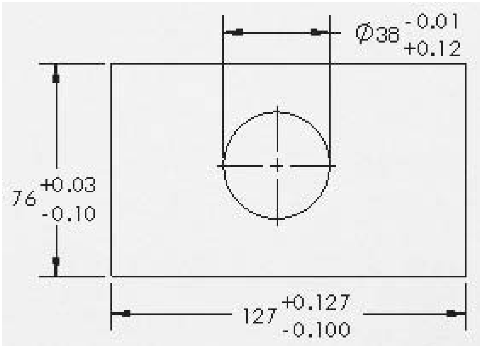
C. Limit dimensions should have the same number of decimal places by adding zeros, if necessary, as shown in Figure 12.4C.

D. Drop the zeros in a basic size if you do not specify conventional (size) tolerance on it, as shown in Figure 12.4D.

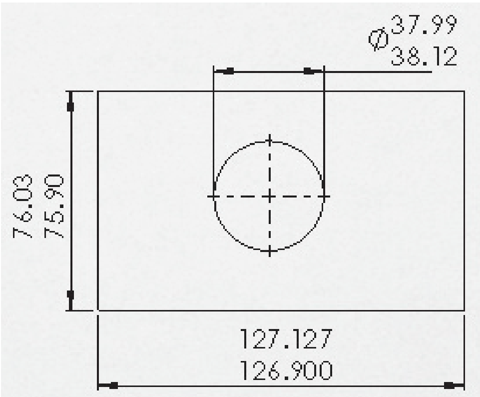
Note that SolidWorks enforces these rules as shown in Figure 12.4 when using the MMGS (millimeter, gram, second) Units system. Also, make sure you select **ISO** drafting standard; use this sequence: **Tools** (SolidWorks menu) > **Options** > **Document Properties** tab > **Drafting Standard** > **ISO** (from drop-down list) > **OK**.



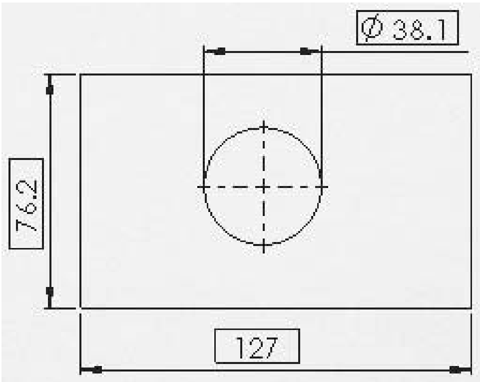
(A) Use a single zero in zero-tolerance value



(B) Use same number of decimals for both tolerances



(C) Use same number of decimals for limits



(D) Drop zeros in basic size

FIGURE 12.4
Rules of millimeter tolerances

4. Follow these rules for inch (English) tolerances:

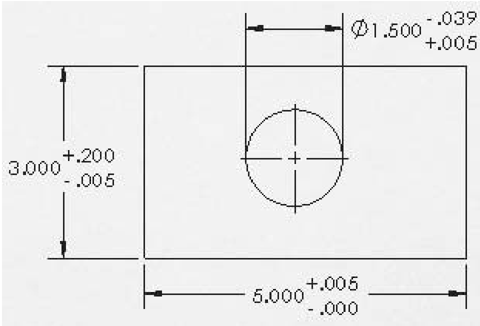
A. For unilateral tolerancing, use the same number of zeros for the zero value as the nonzero tolerance value with the proper sign, as shown in Figure 12.5A.

B. For bilateral tolerancing, you should use the same number of decimal places in the tolerance values as used in the basic size, using zeros when necessary, as shown in Figure 12.5B.

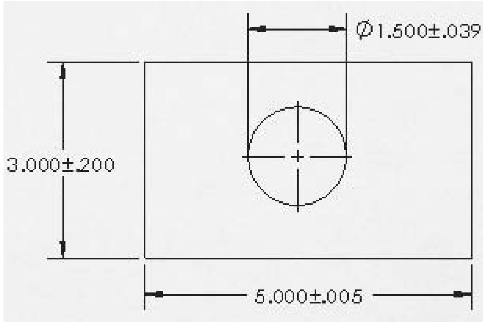
C. Limit dimensions should have the same number of decimal places by adding zeros, if necessary, as shown in Figure 12.5C.

D. Keep the zeros in a basic size if you do not specify conventional (size) tolerance on it as shown in Figure 12.5D.

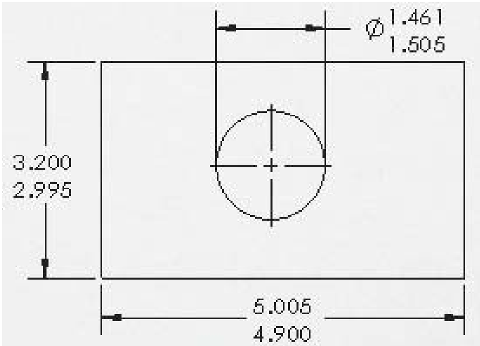
Note that SolidWorks enforces these rules as shown in Figure 12.5 when using the IPS (inch, pound, second) Units system. Also, make sure you select **ANSI** drafting standard; use this sequence: **Tools** (SolidWorks menu) > **Options** > **Document Properties** tab > **Drafting Standard** > **ANSI** (from drop-down list) > **OK**.



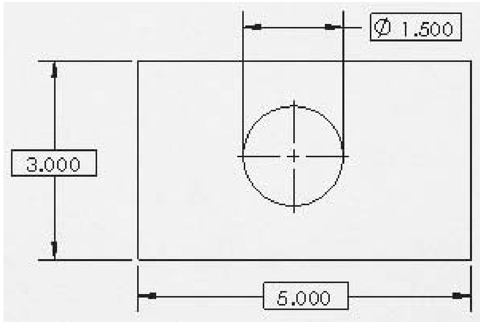
(A) Use many zeros in zero-tolerance value



(B) Use same number of decimals as basic size



(C) Use same number of decimals for limits



(D) Keep zeros in basic size

FIGURE 12.5
Rules of inch tolerances

5. When you use tolerances on an angle dimension, use the same number of decimals in the tolerance value as in the basic size, that is, 45.00° ± 0.10°, or 45.0° ± 0° 30′ (which can also be written as 45.0° ± 0.5°): Note that a degree has 60 minutes and a minute has 60 seconds. The symbols of degree, minute, and second are, respectively, °, ′, and ″.
6. Interpret tolerances during inspection: All limits are absolute regardless of the accuracy (number of decimal places) of the specified tolerances; that is, 5.002 means 5.0020000 … 0. However, the number of decimals informs the inspection personnel what to do. They compare the measured value with the specified value, and any deviation outside the specified limiting value is ignored.
7. Use single limit appropriately: You may use MIN or MAX to specify a tolerance on a dimension where other elements of the design clearly determine the other unspecified limit. That is, the design intent is clear and the unspecified limit can be zero or approach infinity and will not result in design problems. Use a single limit for features such as the depth of a hole, length of a thread, corner radii, and chamfers. SolidWorks uses this format to specify a MIN or MAX: 5.000 MIN or 5.000 MAX.

12.5 Tolerancing Tapers

There exist two types of tapers: conical and flat, as shown in Figure 12.6. An example of a conical taper is a conical shaft, and an example of a flat taper is a wedge. Figure 12.6 also shows how to specify (dimension) a taper. The dimensioning and tolerancing of a taper depend heavily on its functional requirements. Figure 12.6A shows the four common schemes to dimension a conical taper. Scheme 1 implies that the taper cross section at distance L from the left end with a diameter d is important and should be toleranced.

For example, the conical taper may plug a conical hole at the specified location to prevent a leak. Thus, the location and the diameter are crucial to the taper functionality.

Scheme 2 implies that the taper end diameters, D and d, and its overall length L are important and should be toleranced. For example, the taper may be a floating core of a valve that must block the gas or fluid flow from one end of the valve to the other. The dimensioning and tolerancing of a flat taper follow similar rules and are shown in Figure 12.6B.

Scheme 3 shows the use of an angle and scheme 4 shows the use of a slope. Note the use of the slope symbol. SolidWorks does not have the slope symbol for the conical taper, shown in Figure 12.6A4, but it does have the symbol for the flat taper shown in Figure 12.6B4.

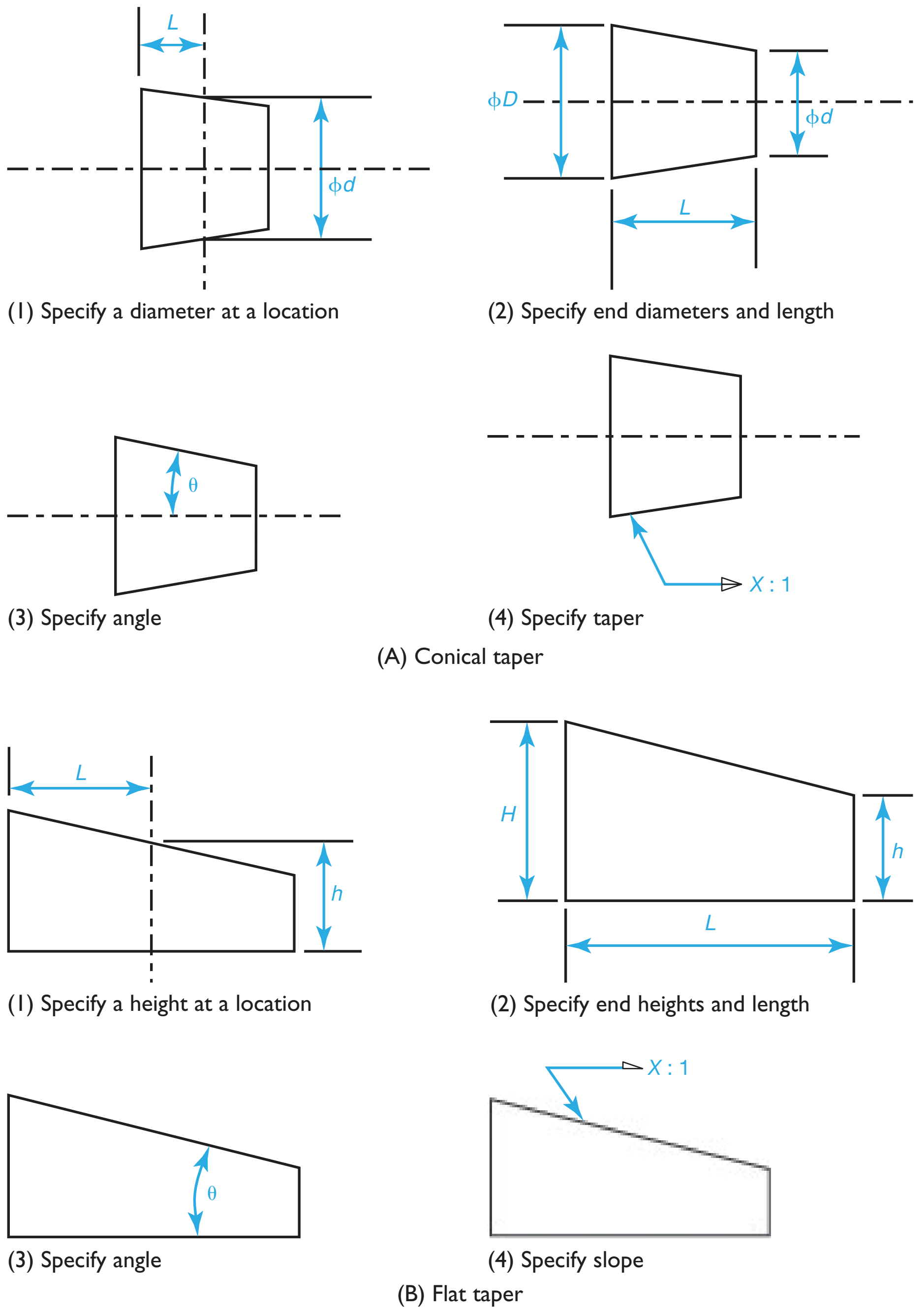


FIGURE 12.6
Types and dimensioning of tapers

The taper of the conical taper shown in Figure 12.6A4 is defined by the following equation:

$$Taper = (D - d)/L \tag{12.1}$$

And the slope of the flat taper shown in Figure 12.6B4 is given by:

$$Slope = (H - h)/L \tag{12.2}$$

Figures 12.6A4 and 12.6B4 show the taper and slope as normalized values of x:1, where x is the right-hand side of each equation. Observe that the definitions of the taper and the slope, given by Eqs. (12.1) and (12.2), are a little different from each other. These definitions are set by the ASME Y14.5M-1994 standard; the new revision is ASME Y14.5-2009. The tutorials of this chapter apply the taper concepts discussed here using SolidWorks.

12.6 Limits of Dimensions

The two questions we answer in this section are: Why and how do we standardize tolerances? And how do we use them to calculate the limits of a dimension? We need to standardize tolerances to standardize part manufacturing and inspection. This in turn ensures parts' interchangeability, which guarantees that replacement (spare) parts fit well into products as intended. Both ANSI and ISO standards have developed two equivalent systems of standard classes of fits that designers can use. Each fit has standard tolerances associated with it depending on the basic size that the designer wants to assign tolerances to. Table 12.1 shows the ANSI fits and their ISO equivalents. These fits are divided into three classes: clearance, transition, and interference. Clearance fit occurs

TABLE 12.1 ANSI and ISO Fits

		ISO Symbol				ISO Symbol	
Class	**ANSI Symbol**	**Hole**	**Shaft**	**Class**	**ANSI Symbol**	**Hole**	**Shaft**
Clearance fits	RC1	H5	g4	Transition fits	LT1	H7	js6
	RC2	H6	g5		LT2	H8	js7
	RC3	H7	f6		LT3	H7	k6
	RC4	H8	f7		LT4	H8	k7
	RC5	H8	e7		LT5	H7	n6
	RC6	H9	e8		LT6	H7	n7
	RC7	H9	d8	Interference fits	LN1	H6	n5
	RC8	H10	c9		LN2	H7	p6
	RC9	H11	c11		LN3	H7	r6
	LC1	H6	h5		FN1	H6	n6
	LC2	H7	h6		FN2	H7	s6
	LC3	H8	h7		FN3	H7	t6
	LC4	H10	h9		FN4	H7	u6
	LC5	H7	g6		FN5	H8	x7
	LC6	H9	f8				
	LC7	H10	e9				
	LC8	H10	d9				
	LC9	H11	c10				
	LC10	H12	c12				
	LC11	H13	c13				

when the shaft moves loosely inside the hole it fits into. Transition fit occurs when the shaft is in transition from clearance to interference; that is, the shaft is snug fit into the hole. Interference fit occurs when the shaft is pressed into the hole and cannot move inside it. There are different grades (levels) within each tolerance class: very loose, moderate, or tight clearance or interference. Think of the fits as a sliding scale from very loose clearance to very tight interference, and anything in between.

Table 12.1 shows that each ANSI fit designation has two ISO symbols. For example, the ANSI fit designation of RC1 is equivalent to the ISO fit designation of H5/g4. H5 and g4 designate the hole and shaft, respectively. All holes use the letter H to indicate that we use a hole-based system, whereas shafts use different letters. Each letter represents a fit class. The number next to each letter is the tolerance grade number or the IT grade. Our example of H5/g4 implies that the hole has a class and IT grade of H and 5, respectively, and the shaft has g and 4, respectively. The hole IT grade is always 1 higher than the shaft IT grade because manufacturing the hole is harder than manufacturing the shaft. There exist ANSI tables (not shown here) that relate the tolerance IT grades to the different manufacturing processes. Each process is capable of producing a range of IT grades.

Having answered the first question, we now show how to use the fits in Table 12.1 to calculate the limits on a dimension. These calculations require the introduction of the concept of allowance a. While the shaft and hole tolerance zones s and h shown in Figure 12.2B control the dimensions of each of the shafts and the holes individually, they do not have an effect on how loose or tight the assembly of the two is; that is, s and h cannot control the type of fit between the shaft and the hole. The concept of allowance a gives us this control. We can produce the three classes of fits (clearance, transition, and interference) by controlling the allowance a. There are variations within each class. For example, we have loose and tight clearances, tight and shrink-fit interferences. The **allowance a** is an abstract algebraic value; it is positive for clearance and negative for interference.

The three values of the three parameters s, h, and a determine the shaft and hole limits as well as the type of fit of their assembly. These parameters provide the designer with full control over the functional requirements of an assembly. They also ensure that manufacturing meets these requirements. Figure 12.7 shows the three classes of fit in terms of the three tolerance zones s, h, and a. As Figure 12.7 shows, we use the basic size d of the shaft or the hole as a reference (X-axis). We place the tolerance zones as shown for each fit. The allowance a is defined as the difference between the maximum shaft and the minimum hole dimensions. This definition is consistent with the MMC. A transition fit is the changeover fit from clearance to interference, as shown in Figure 12.7D. Also, notice that the hole tolerance zone h always butts the basic size reference line, thus ensuring that $d_{h\min}$ is always equal to the basic size d. This is a result of using the hole-based system. Also, notice that a transition fit may be a clearance (represented by tolerance zone s below the basic size line) or interference (represented by tolerance zone s above the basic size line). Finally, notice that the allowance a may be zero for a transition fit or not.

We use Figure 12.7 to calculate the dimension limits for both the hole and the shaft. The following equations are easily derived:

$$d_{h\min} = d \tag{12.3}$$

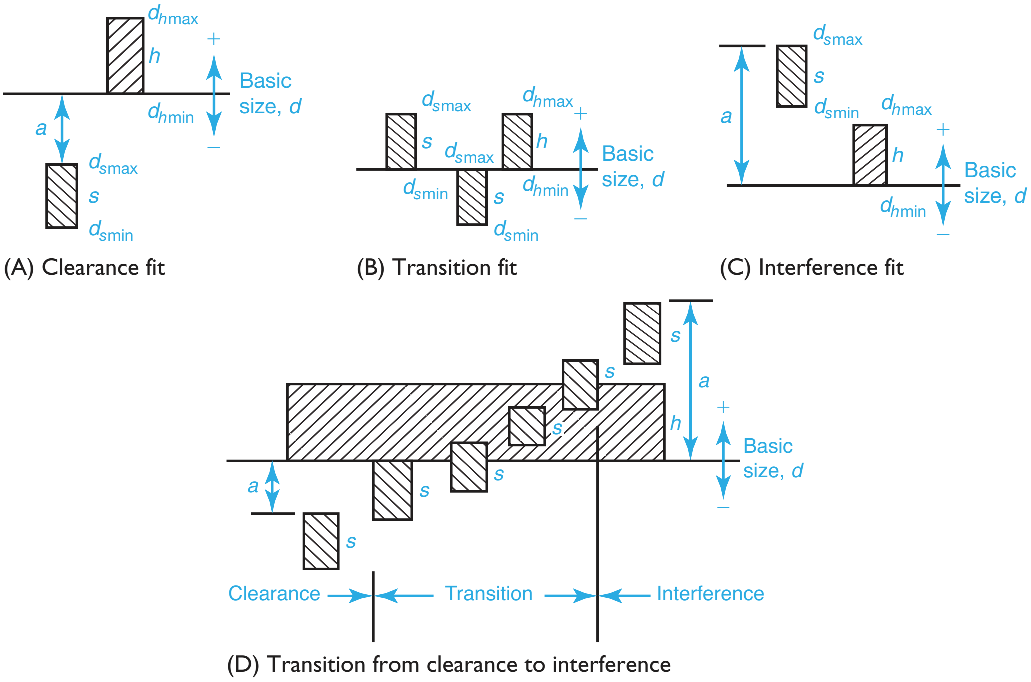
$$d_{h\max} = d + h \tag{12.4}$$

$$d_{s\max} = d - a \tag{12.5}$$

$$d_{s\min} = d - a - s = d_{s\max} - s \tag{12.6}$$

Equations (12.3)–(12.6) are applicable to any fit type because the allowance a is algebraic. We use the tolerance tables listed in Appendix A to calculate the limits for all types of fits. Follow these steps to calculate the dimension limits:

FIGURE 12.7
Representing classes of fit using the hole-based system



1. Select a fit class from Table 12.1 based on the desired fit type between the shaft and the hole under consideration. This type is determined by the functional requirements of the shaft/hole assembly.
2. Use the ANSI tolerance tables listed in Appendix A to calculate the limits on both the shaft and the hole.

While Step 1 is straightforward, we elaborate Step 2. The tolerance tables in Appendix A implement the fits shown in Table 12.1 for different basic sizes. They also implement Eqs. (12.3) through (12.6). This facilitates the job of the designer to calculate the dimension limits. The ANSI tables in Appendix A have three columns for each fit to help you calculate the limits that correspond to the fit. Use the **Hole** and **Shaft** columns of the appendix Tables A.1 - A3 to determine the limits. Simply add the values algebraically to the basic size, *d,* as shown in Eqs. (12.7)–(12.10). SolidWorks has implemented the tables of Appendix A. Keep in mind that all the values listed in the appendix tables must be multiplied by 10^{-3} as the table footnote indicates.

Here is how to use the appendix to calculate the hole and shaft limits for a given basic size *d* and a given fit:

Step 1: Use the type of fit (clearance, transition, or interference) to select the appropriate table (Table A.1 for clearance, Table A.2 for transition, or Table A.3 for interference).
Step 2: Within the selected table, scan the **Basic Size** column (at the far left of the table) using the value of the basic size *d* to locate the row whose range includes that *d* value.
Step 3: Go across using the row to locate the **Hole** and **Shaft** columns that correspond to the given fit.
Step 4: Once there, use the four values that intersect the row and are listed under the two columns as follows:

$$d_{hmin} = d + \text{small value in } \textbf{Hole} \text{ column (always 0 for hole-based system)} \quad (12.7)$$

$$d_{hmax} = d + \text{large value in } \textbf{Hole} \text{ column} \quad (12.8)$$

$$d_{smax} = d + \text{large algebraic value in } \textbf{Shaft} \text{ column} \quad (12.9)$$

$$d_{smin} = d + \text{small algebraic value in } \textbf{Shaft} \text{ column} \quad (12.10)$$

Substitute the values calculated from Eqs. (12.8)–(12.10) into Eqs. (12.4)–(12.6) to calculate the tolerance zones h, s, and a. Keep in mind that a is an algebraic value—positive for clearance, negative for interference, and zero for transition (Figure 12.7). Equation (12.5) should confirm that observation ($a = d - d_{smax}$).

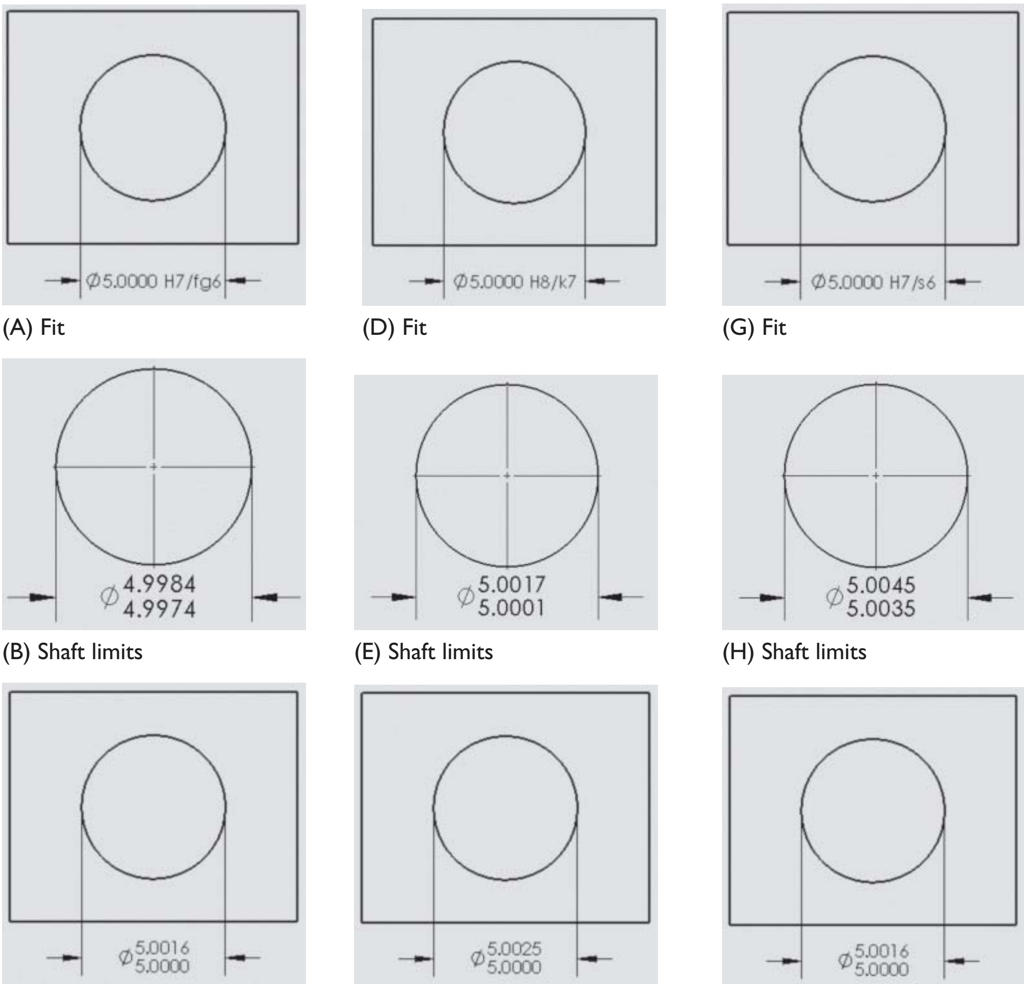
Example 12.1 Calculate the limits and tolerance zones for the following three fits: clearance of RC3, transition of LT4, and interference of FN2. Use a basic size of 5.0000 in.

Solution Following the above four steps, Table 12.2 shows the results.

TABLE 12.2 Limits for Basic Size d = 5.000 in.

Fit	d_{hmin}	d_{hmax}	d_{smin}	d_{smax}	h	s	A
RC3 (H7/f6)	5.0000	5.0016	4.9974	4.9984	0.0016	0.0010	0.0016
LT4 (H8/k7)	5.0000	5.0025	5.0001	5.0017	0.0025	0.0016	−0.0017
FN2 (H7/s6)	5.0000	5.0016	5.0035	5.0045	0.0016	0.0010	−0.0045

These limits shown in Table 12.2 agree with the SolidWorks results shown in Figure 12.8.



(A) Fit (D) Fit (G) Fit

(B) Shaft limits (E) Shaft limits (H) Shaft limits

(C) Hole limits Clearance fit RC3 (H7/f6)

(F) Hole limits Transition fit LT4 (H8/k7)

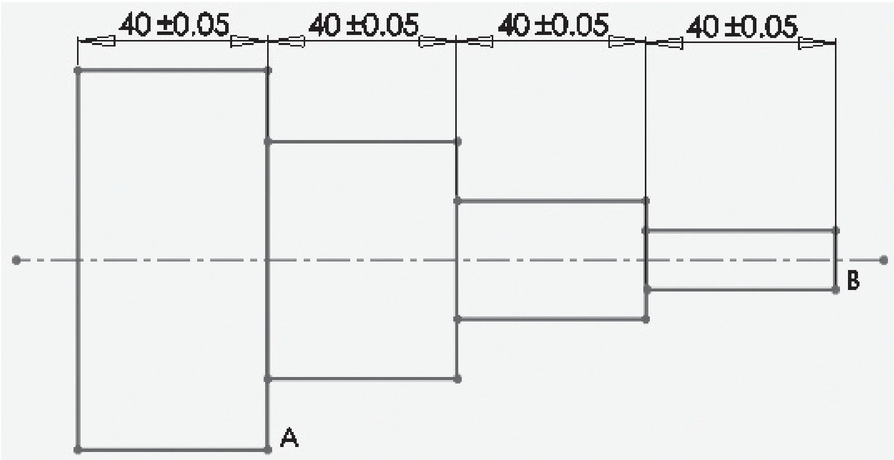
(I) Hole limits Interference fit (H7/s6)

FIGURE 12.8 Limits verification using SolidWorks

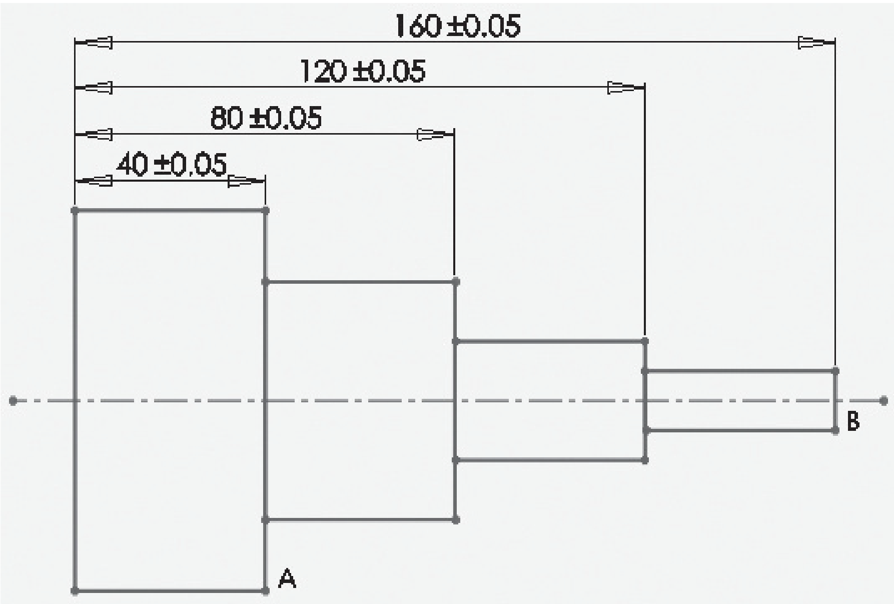
HANDS-ON FOR EXAMPLE 12.1. Find the dimension limits for RC2, LT2, and FN2. Use a basic size of 6.0000 in.

12.7 Tolerance Accumulation

Tolerance accumulation is the chain effect of a series of tolerances. Tolerance accumulation is controlled by the dimensioning scheme. Tolerance accumulation is also known as tolerance stackup. For example, a horizontal or vertical chain of toleranced dimensions results in tolerance accumulations. You should avoid tolerance accumulation as much as possible by changing dimensioning schemes. Figure 12.9 shows a toleranced stepped shaft. Our goal is to dimension the shaft in such a way as to minimize tolerance

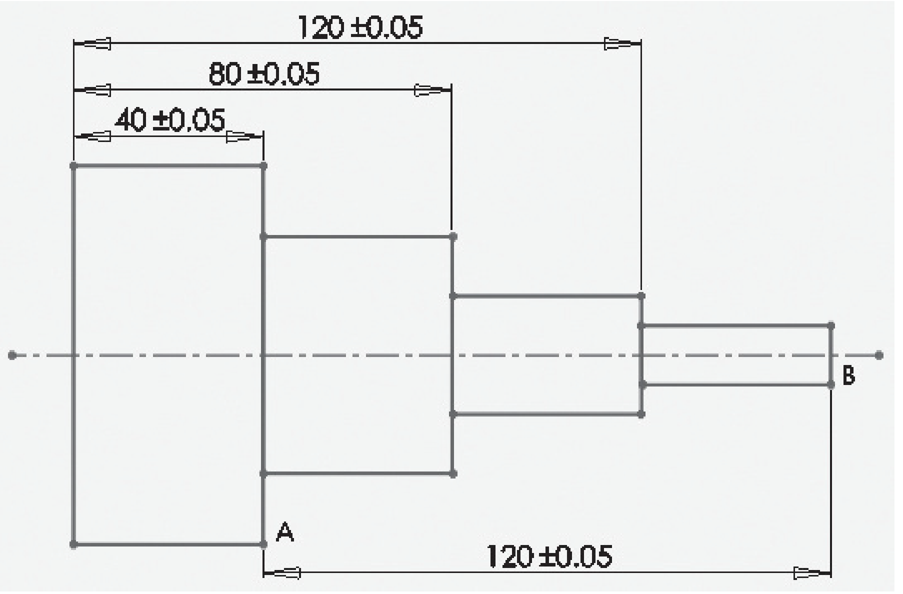


(A) Chain dimensioning



(B) Baseline dimensioning

FIGURE 12.9
Tolerance accumulation



(C) Direct dimensioning

FIGURE 12.9
(*continued*)

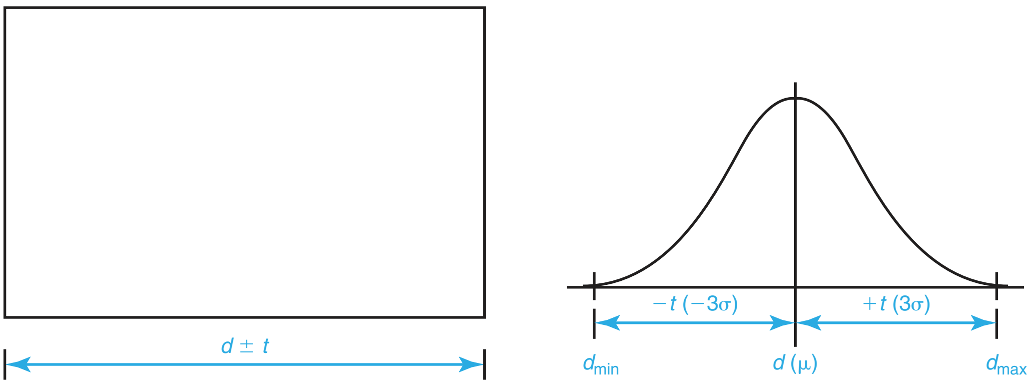
accumulation between planes A and B to meet part functional requirements. We investigate the effect of the following three dimensioning schemes on tolerance accumulation:

1. Chain dimensioning: Figure 12.9A shows this dimensioning scheme, which is a horizontal dimension chain. This is the worst dimensioning scheme of the three schemes shown in Figure 12.9 because it produces the maximum tolerance accumulation. The maximum variation between planes A and B is equal to the sum of the tolerances between the two planes (i.e., ± 0.15).
2. Baseline dimensioning: Figure 12.9B shows this dimensioning scheme. The maximum variation between the two planes is equal to the sum of the two dimensions from the farmost left plane (origin) to the two planes (i.e., ± 0.10). This results in a reduction of the tolerance accumulation from the preceding scheme.
3. Direct dimensioning: Figure 12.9C shows this dimensioning scheme. It is the best of the three schemes shown in Figure 12.9 because it does not produce any tolerance accumulation. The maximum variation between planes A and B is equal to the tolerance of the dimension between the two planes (i.e., ± 0.05).

12.8 Statistical Tolerancing

When we assign tolerances to parts (components), our goal is to control the assembly of these parts into a functional assembly. As we discussed in Section 12.1, specifying a tolerance on a part dimension (Figure 12.10A) generates a normally distributed population of the part (Figure 12.10B). The horizontal axis represents the actual dimension produced, and the vertical axis represents the probability of producing a given dimension. The mean, μ, of the normal distribution is the perfect form (i.e., the basic size d). The two extremes of the curve are the maximum and minimum sizes, d_{max} and d_{min}, respectively, as shown in Figure 12.10B, and represent the maximum and minimum material conditions. The distance between the mean and any of the two extremes is known as 3σ, where σ is the standard deviation. The area under the bell curve that is

FIGURE 12.10
Population of a toleranced dimension



(A) Toleranced dimension
(B) Normal distribution of parts

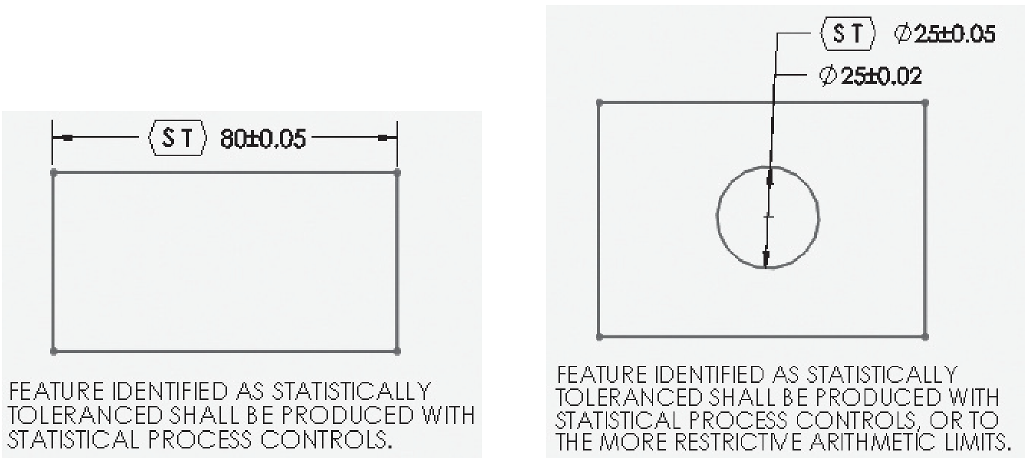
bounded by $\pm 3\sigma$ includes 99.73% of all the possible parts produced (manufactured) with the specified tolerance.

All manufacturing processes subject to common cause variation produce parts that vary in size around the basic size, *d*. If the manufacturing process is "perfectly" capable, then very few parts are produced at the extremes of the bell curve (Figure 12.10B), that is, at maximum or minimum conditions. In other words, the probability of getting parts on both tails of the curve is very small as the curve shows, whereas the probability of getting parts around the basic size is very high. Thus, we can use this observation to relax tolerances on dimensions in a calculated manner. In such a case, we use the concept of statistical tolerancing. **Statistical tolerancing** is a method of tolerancing that is based on the fact that it is highly unlikely that we would get mating parts (in an assembly) at both maximum and minimum material conditions.

Unlike statistical tolerancing, arithmetic tolerancing assumes all parts are produced at the maximum or minimum material conditions. When using arithmetic tolerances is restrictive, we may use statistical tolerancing to increase tolerances on a dimension. The increased tolerance may reduce manufacturing cost, but should be employed where the appropriate process control is used to control the manufacturing process.

Specifying statistical tolerances on dimensions is designated as shown in Figure 12.11. Designers can use SolidWorks to specify statistical tolerances as shown in the figure. To access the **<ST>** symbol, click the **More** button while inserting a dimension to access the Symbol library. The designations shown in Figure 12.11 follow the ASME Y14.5M-1994 standard. Figure 12.11A shows the use of the statistical tolerance symbol. The standard mandates that the statement shown be placed on the drawing. Figure 12.11B shows an

FIGURE 12.11
Specifying statistical and arithmetic tolerances



(A) Statistical tolerance
(B) Statistical tolerance with arithmetic limits

interesting example where it may be necessary to designate both the statistical and arithmetic limits when the dimension has the possibility of being produced without statistical process control (SPC). In such a case, the shown note must be placed on the drawing. Note that the arithmetic tolerance is tighter than the statistical tolerance, thus making it more expensive to manufacture the part.

12.9 True Position

The concept of true position is part of geometric tolerancing. True position applies to locating and tolerancing a hole in a part. A hole has a size (radius or diameter) and a location (center). Tolerancing the hole size is straightforward and uses conventional tolerances. Tolerancing the hole center depends on how we locate it. We locate the center by (x, y) coordinates measured from two planes. Thus, we specify a tolerance on each coordinate (dimension), that is, $\pm\Delta x$ and $\pm\Delta y$, as shown in Figure 12.12A. Although this logic seems to make sense, it is not good because it "under-tolerances" the part; that is, the hole location has more tolerance than we intend. The tolerances produce a rectangular tolerance zone that is $2\Delta x \times 2\Delta y$, as shown in Figure 12.12A, thus allowing more tolerance on the hole location than specified. The length of the diagonal of the tolerance area is given by:

$$L = 2\sqrt{(\Delta x)^2 + (\Delta y)^2} \tag{12.11}$$

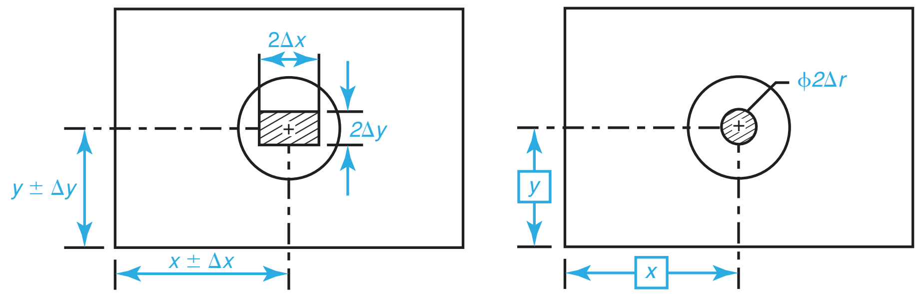


FIGURE 12.12
True position of a hole

The hole center can move along the diagonal whose length is larger than the specified $2\Delta x$ or $2\Delta y$. This allows the part to pass inspection when in reality it should not.

The preceding problem can be easily solved by decoupling the tolerance on the hole center from its (x, y) coordinates. In other words, remove the $\pm\Delta x$ and $\pm\Delta y$. The concept of true position achieves that, as shown in Figure 12.12B. **True position** is defined as specifying the hole (x, y) coordinates without any tolerances. The true position allows us to define a circular tolerance zone on the hole with a diameter of $2\Delta r$, as shown in Figure 12.12B. We customarily display the hole coordinates in boxes, as shown in Figure 12.12B. How do we specify tolerances on the hole location then? We define a position tolerance, as we show when we cover geometric tolerances.

12.10 Geometric Tolerances

Geometric tolerances complement conventional tolerances. They control the location, shape (form), and profile of individual features. Table 12.3 shows the different geometric tolerances and their symbols as specified by the ASME Y14.5M-1994 standard.

TABLE 12.3 Symbols for Geometric Tolerances

Feature Type	Tolerance Type	Description	Symbol
Individual feature	Form	Straightness	—
		Flatness	▱
		Circularity (roundness)	○
		Cylindricity	⌭
Individual or related features	Profile	Line profile	⌒
		Surface profile	⌓
Related features	Orientation	Angularity	∠
		Perpendicularity	⊥
		Parallelism	//
	Location	Position	⌖
		Concentricity	◎
	Runout	Circular runout	↗
		Total runout	⌰

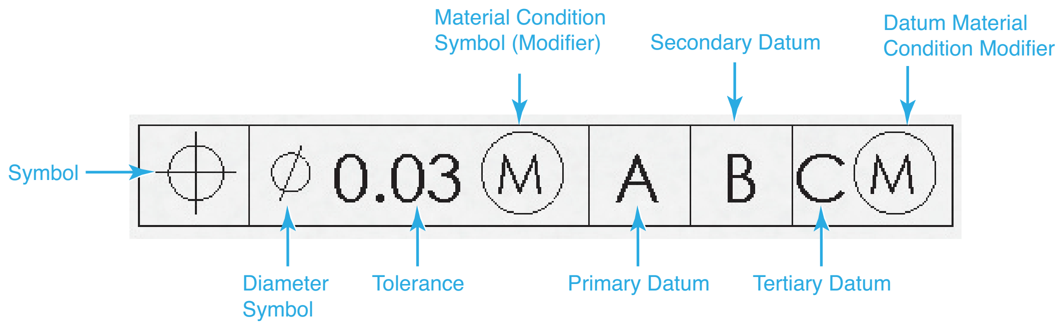
To master geometric tolerances, we need to know how to assign and interpret them. Assigning a geometric tolerance requires the following steps:

1. Specify needed datums. These datums are used to measure dimensions from.
2. Decide on the type of geometric tolerance you need to apply to a given feature. Select the desired tolerance from Table 12.3.
3. Create the geometric tolerance details.

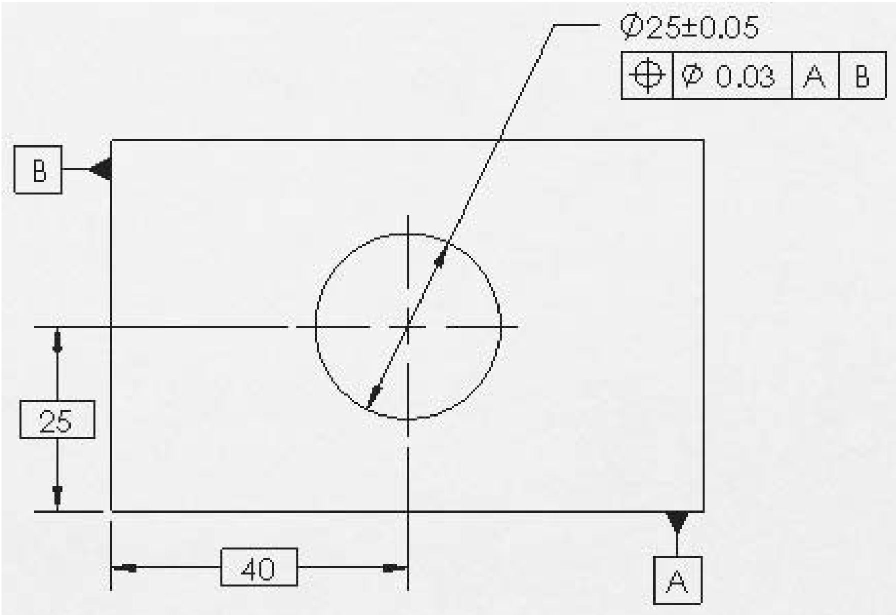
Figure 12.13 shows an example. Figure 12.13A indicates the elements of a geometric tolerance, which we create one by one in SolidWorks. The symbol comes from Table 12.3, depending on the tolerance we need to create (position tolerance in our example). The tolerance amount (zone) follows the symbol. The diameter symbol is optional and used when it makes sense. In this example, we need it because we specify a tolerance zone for the center of the hole. The material condition symbol (modifier) is optional. The next three letters indicate the datums. The first datum is the primary datum, followed by the secondary datum, followed by the tertiary datum. The order shown indicates to the part inspector how to inspect the part. There are three planes of the inspection gauge that correspond to these datums. The part inspector must first align the part surface A with the primary datum (A) of the gauge, followed by the others. If the part surface A does not mate with the gauge plane at the datum target points and/or lines, the part fails inspection. Last, we may optionally specify a material condition on any datum.

Figure 12.13B shows two tolerances on the hole. One is a size tolerance on the hole diameter, which is a conventional tolerance. The other is a location tolerance on the hole center. It reads as follows: "There is a tolerance zone with a diameter of 0.03 mm on the hole center measured with respect to datums A (Primary) and B (Secondary)." In other words, the hole center is free to move from its true position within a circle whose diameter is 0.03 mm and which is centered at the true position point. If the hole center is located outside this circle, the part is rejected during inspection. Why do we use two datums in Figure 12.13B? It is because the hole location is determined by the (x, y) coordinates, each requiring a datum.

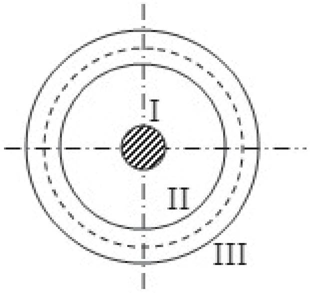
Using and interpreting a material condition symbol (modifier) is the hardest to do and can be well understood only in the context of designing an inspection gauge. For a Go-NoGo gauge, the location of the gauge pin or hole is located where the condition is



(A) Anatomy of a geometric tolerance specification



(B) True position tolerance



(C) Interpret position tolerance

FIGURE 12.13
Sample geometric tolerance

met, thus forcing the part to meet the specified condition as well. Typically, it takes time and experience to use material condition modifiers in geometric tolerances. The required depth of coverage needed to fully understand the use and interpretation of material conditions is beyond the scope of this book.

Interpreting geometric tolerances is important to learn because it enhances our understanding and therefore enables us to assign geometric tolerances correctly. When we interpret geometric tolerance, we always look for the tolerance zone that results from the specified tolerance, the part feature (surface) that is affected by it, how it moves within the tolerance zone, and how the datum planes influence the interpretation. Let us apply these four observations to the hole tolerances shown in Figure 12.13B. The resulting tolerance zones are shown in Figure 12.13C. The true position tolerance on the hole location (center) produces circle I. This is a circular tolerance zone with a diameter of 0.03 mm as specified by the geometric tolerance. Any part produced with a hole center within this circle is acceptable and meets the design requirements.

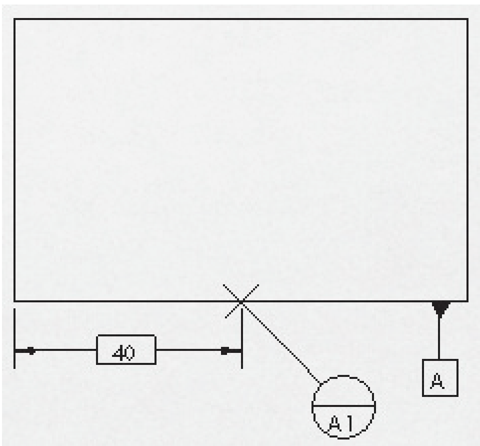
The size tolerance on the hole indicates that the hole diameter is acceptable anywhere between circles II and III shown in Figure 12.13C. These two circles represent the maximum material condition (circle II with a diameter of 24.95 mm) and minimum material condition (circle III with diameter of 25.05 mm). The dashed circle shown in Figure 12.13C represents the basic size of the hole (Ø25 mm). When we combine the effect of both the size and location tolerances, we can think of infinite combinations for the resulting hole. All these holes fall under the bell curve shown in Figure 12.10B. Each part with a hole is a point under the curve. The correct inspection gauge is capable of checking for all these infinite combinations and passing only the ones that fit under the curve.

12.11 Datum Target Symbols

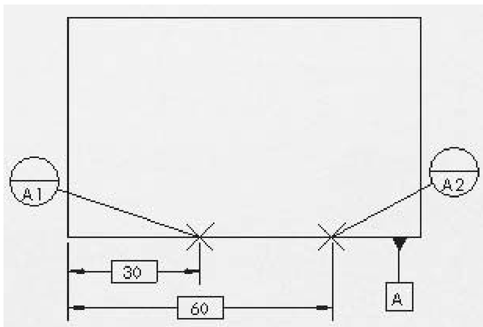
As discussed at the end of Section 12.3, datum targets are designated points, lines, or areas of contact on a part to define a datum (feature of reference). These designated targets are used during inspection. We use datum targets to define datums because an entire feature (planar or cylindrical face) cannot be used to establish a datum due to the irregularity of the part face that serves as a datum. A datum is always an actual face of a part.

Points, lines, and areas on datum features are designated on a drawing via datum target symbols. Figure 12.14 shows different ways to define datum targets. These targets are created in SolidWorks. Figure 12.14A shows how to indicate a datum target point. In this case, we specify (define) datum A by the point located at 40 mm from the left face of the part. When we want to specify a datum (A) by a line, we specify two points, A1 and A2, on the line, as shown in Figure 12.14B. Figure 12.14C shows how to specify a datum (C) by an area. Datum C is the front plane. The area is indicated by a diameter (Ø10). Figure 12.14C shows the two options to specify the datum: with or without a shaded area. The area indicates that an area of contact is necessary to establish a datum. The shape of the target area may be circular or rectangular. Consult the SolidWorks menu; click **Insert** > **Annotations** > **Datum Target**.

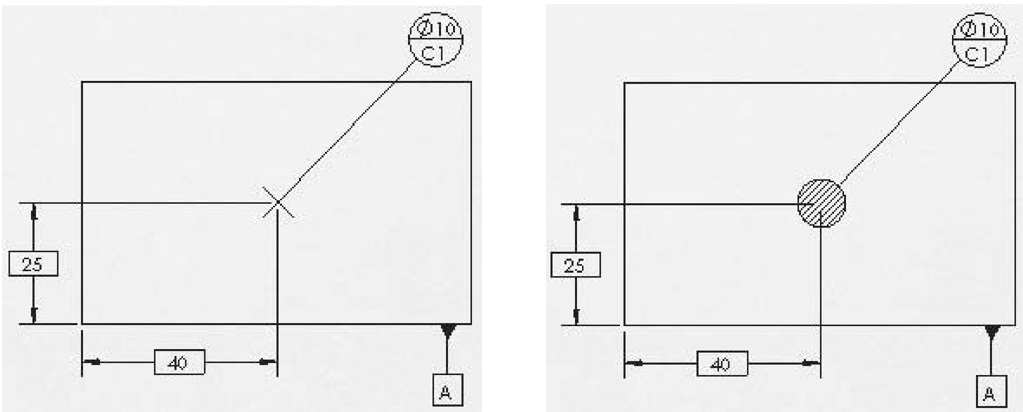
FIGURE 12.14
Datum targets



(A) Datum target point



(B) Datum target line



(C) Datum target area

12.12 Tolerance Interpretation

Interpreting tolerances is important to understanding them. Conventional tolerances are usually easier to interpret than geometric tolerances. The underlying principle is that a tolerance defines a zone within which the toleranced feature resides. The toleranced feature assumes any location and orientation within the tolerance zone. Table 12.4 shows sample tolerances with their graphical interpretations. These tolerances (except for the size tolerance) correspond to those shown in Table 12.3.

TABLE 12.4 Tolerance Interpretation

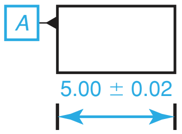
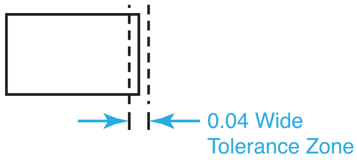
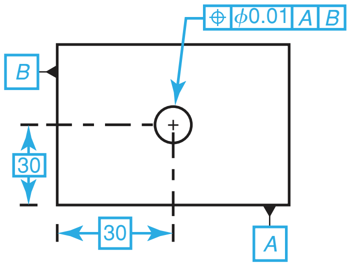
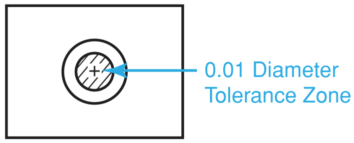
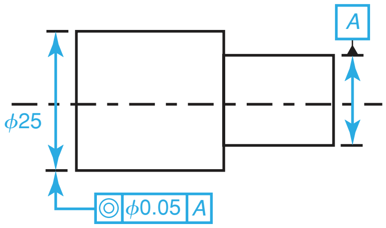
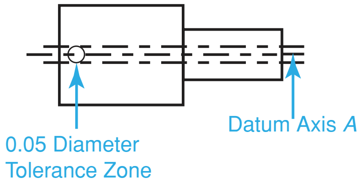
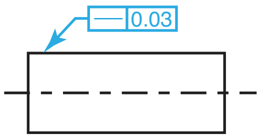
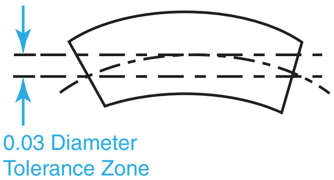
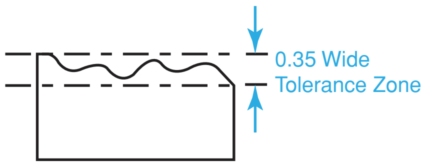
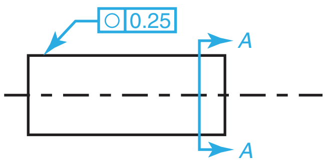
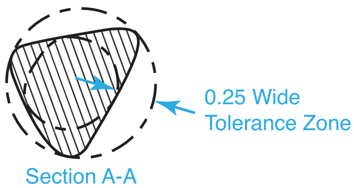
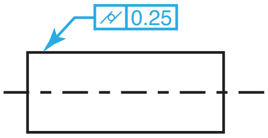
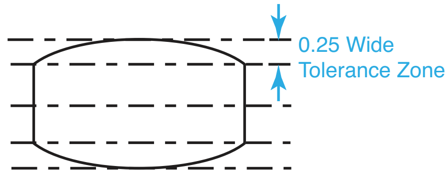
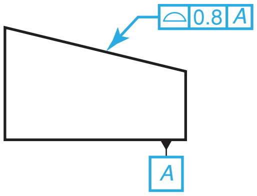
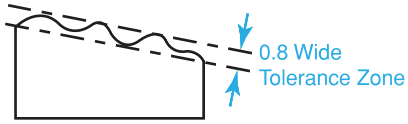
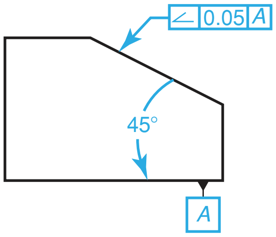
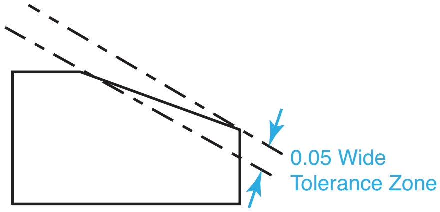
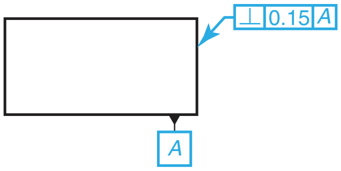
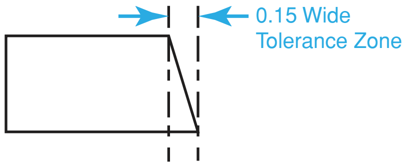
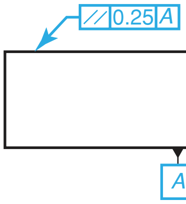
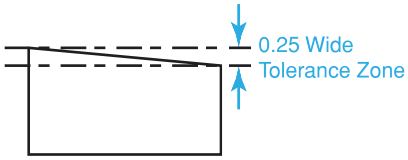
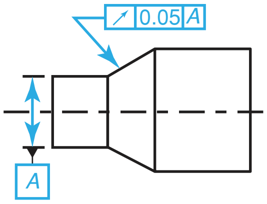
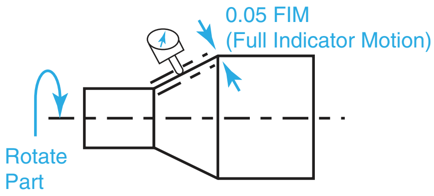
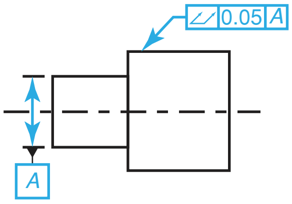
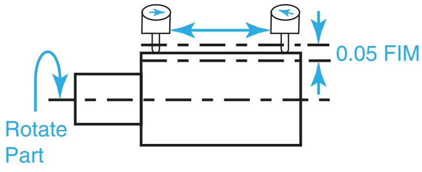
Tolerance Type	This on the Drawing	Means This
Size		
Position		
Concentricity		
Straightness		
Flatness		
Circularity		

TABLE 12.4 (*Continued*)

Tolerance Type	This on the Drawing	Means This
Cylindricity		
Surface Profile		
Angularity		
Perpendicularity		
Parallelism		
Circular runout		
Total runout		

12.13 Tolerance Analysis

There are two types of tolerance studies: analysis and synthesis. **Tolerance analysis** (also known as *tolerance stack-up analysis*) is the study of how tolerances and assembly methods affect dimensional stack-up between two features of an assembly. In other words, we assign tolerances to individual components and study the effect of these tolerances on the assembly of the components. Thus, component tolerances are known and we calculate the resulting assembly tolerance.

Tolerance synthesis is also known as *tolerance allocations*. It is the reverse of tolerance analysis. The assembly tolerance is known from design requirements, while component tolerances are not known. **Tolerance synthesis** is the method of how we allocate (distribute) the assembly tolerance among its individual components in a reasonable way. We cover tolerance analysis only in this chapter.

The two most commonly used methods for tolerance analysis are worst case and statistical. The worst-case method assumes that the dimension of each component in the assembly is at its worst, that is, maximum or minimum. This method produces the worst possible assembly limits. In the worst-case analysis, the assembly tolerance is given by the linear summation of the component tolerances. For one-dimensional assemblies with N components, the assembly tolerance is given by:

$$T = \sum_{i=1}^{i=N} T_i \tag{12.12}$$

where T and T_i are, respectively, the assembly tolerance and the component tolerance. One-dimensional assembly means an assembly in which toleranced dimensions form a chain in one direction (horizontal, vertical, or any other direction).

For multidimensional assemblies, Eq. (12.12) becomes:

$$T = \sum_{i=1}^{i=N} \left| \frac{\partial f}{\partial x_i} \right| T_i \tag{12.13}$$

where $|\cdot|$ is the absolute value, x_i is the nominal component dimensions, and $f(x_i)$ is an assembly function describing the resulting dimension of the assembly, such as clearance or interference. The partial derivatives represent the sensitivity of the assembly tolerance to variations in individual component dimensions.

Equations (12.12) and (12.13) assume that the tolerances are symmetric (i.e., the negative and the positive limits are equal). If the tolerances are bilateral (i.e., the limits are not equal), Eqs. (12.12) and (12.13) can be applied to each limit to give:

$$T_{\text{min}} = \sum_{i=1}^{i=N} T_{i\text{min}}, \qquad T_{\text{max}} = \sum_{i=1}^{i=N} T_{i\text{max}} \tag{12.14}$$

and

$$T_{\text{min}} = \sum_{i=1}^{i=N} \left| \frac{\partial f}{\partial x_i} \right| T_{i\text{min}}, \qquad T_{\text{max}} = \sum_{i=1}^{i=N} \left| \frac{\partial f}{\partial x_i} \right| T_{i\text{max}} \tag{12.15}$$

The statistical method of tolerance analysis is based on the statistical concepts covered in Section 12.8. This method uses the RSS (root sum squared) to evaluate the statistical tolerance stack-up. The method assumes a normal (Gaussian) distribution for the component variations. In this case, the one-dimensional assembly tolerance is given by:

$$T = \sqrt{\sum_{i=1}^{i=N} T_i^2} \tag{12.16}$$

For a multidimensional assembly, the assembly tolerance is given by:

$$T = \sqrt{\sum_{i=1}^{i=N} \left(\frac{\partial f}{\partial x_i} \right)^2 T_i^2} \tag{12.17}$$

For cases of non-normal distributions, the Monte Carlo simulation method is used to evaluate the tolerance stack-up. The coverage of this method is beyond the scope of this book.

Equations (12.16) and (12.17) assume that the tolerances are symmetric (i.e., the negative and the positive limits are equal). If the tolerances are bilateral (i.e., the limits are not equal), Eqs. (12.16) and (12.17) can be applied to each limit to give:

$$T_{\min} = \sqrt{\sum_{i=1}^{i=N} T_{i\min}^2}, \qquad T_{\max} = \sqrt{\sum_{i=1}^{i=N} T_{i\max}^2} \tag{12.18}$$

and

$$T_{\min} = \sqrt{\sum_{i=1}^{i=N} \left(\frac{\partial f}{\partial x_i}\right)^2 T_{i\min}^2}, \qquad T_{\max} = \sqrt{\sum_{i=1}^{i=N} \left(\frac{\partial f}{\partial x_i}\right)^2 T_{i\max}^2} \tag{12.19}$$

Example 12.2 **An assembly consists of four identical boxes that are stacked on top of each other. We need to control the assembly overall height, *h*, to be between 3.9 and 4.1 in. The height of each box is 1.00 ± 0.03 in. Does the tolerance specification on each box height meet the assembly tolerance requirement?**

Solution The assembly height is 4.00 ± 0.10 in. (3.9, 4.1). Thus, the assembly tolerance is ±0.10 in. Using the worst-case method, Eq. (12.12) gives:

$$T = 0.03 + 0.03 + 0.03 + 0.03 = 0.12 \text{ in.}$$

The tolerance stack-up (0.12 in.) is more than the assembly allowable tolerance limit of 0.10 in. Thus, the tolerance of the box height must be tightened.

Using the statistical method, Eq. (12.16) gives:

$$T = \sqrt{(0.03)^2 + (0.03)^2 + (0.03)^2 + (0.03)^2} = 0.06$$

The resulting statistical assembly tolerance is ±0.06, thus meeting the assembly tolerance requirement. The tolerance calculated by the worst-case method is double that calculated by the statistical method. The two methods of tolerance analysis produce different results and can lead to opposite conclusions. Selecting which method of tolerancing to use is important. We would use the statistical method if the manufacturing methods were well controlled, as covered in Section 12.8. However, the worst-case method is the safer approach. If the inputs (component tolerances) are within their respective tolerances, the output (assembly tolerance) is guaranteed to be within its worst-case tolerance. This is especially important for products like heart valves or critical components of airplanes. However, this guarantee comes at high cost. As discussed in Section 12.8, the worst-case scenario is highly unlikely, if not impossible, in most cases. In this example, all the four boxes must have a height of either 0.97 or 1.03. This could occur only if the manufacturing process producing the boxes had zero variation.

12.14 SolidWorks Tolerance Analysis

SolidWorks implements the tolerance analysis concepts covered here in its **TolAnalyst** module that helps us study how tolerances and assembly methods affect dimensional stack-up between two features of an assembly. In other words, **TolAnalyst** is used to perform stack-up analysis of an assembly. It uses both the worst-case and the statistical methods covered in Section 12.13. The result is a minimum and maximum tolerance stack (Eq. 12.14), a minimum and maximum root sum squared (RSS) tolerance stack (Eq. 12.18), and a list of contributing features and tolerances.

It is advisable to know how to use the SolidWorks **DimXpert** dimensioning module before using **TolAnalyst**. **DimXpert** is used with parts. It prepares models for use with **TolAnalyst**. **DimXpert** works by inserting dimensions and tolerances, automatically or manually, into manufacturing features such as holes and slots. **TolAnalyst** automatically recognizes tolerances and dimensions created in **DimXpert**.

TolAnalyst performs a tolerance analysis called a *study*, which we create using four steps:

1. Measurement: Establish the measurement, which is a linear distance between two features, created using **DimXpert**
2. Assembly sequence: Select the ordered set of parts to establish a tolerance chain between the two measurement features. The selected parts form the "simplified assembly"
3. Assembly constraints: Define how each part is placed or constrained into the simplified assembly
4. Analysis results: Evaluate and review the minimum and maximum tolerance stacks

To create a **TolAnalyst** study:

1. Use **DimXpert** for parts to add tolerances and dimensions to parts of an assembly.
2. Open the assembly.
3. Perform the **TolAnalyst** study: Click **Tools** > **DimXpert** > **TolAnalyst Study**.
4. Follow the above four steps needed to perform the tolerance analysis (study).

To edit a **TolAnalyst** study:

1. Open the assembly containing the study.
2. In the **DimXpertManager** tab, right-click the study and select **Edit Feature**.

Example 12.3 Redo Example 12.2 using SolidWorks TolAnalyst. Compare the results.

Solution This example shows how to use SolidWorks **TolAnalyst** and compares the results with the equations covered in Section 12.13. Figure 12.15 shows the results of the analysis. These results are identical to those of Example 12.2. The **Min** and **Max** values shown in the figure result from the worst-case method. These values are calculated using Eq. (12.14). Click the **MAX** or **MIN** radio button shown in the figure to toggle to view both extremes. The **RSS Min** and **RSS Max** values shown result from the statistical method. These values are calculated using Eq. (12.18). The steps to create the **TolAnalyst** study are as follows:

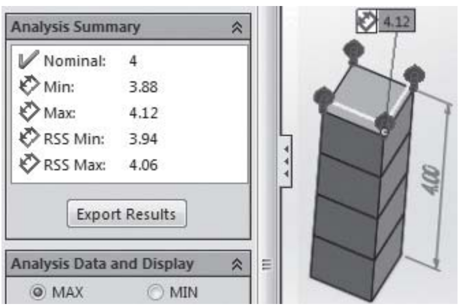
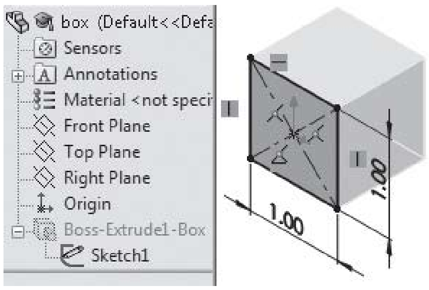


FIGURE 12.15
TolAnalyst study

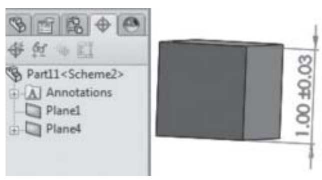
1. Create a $1 \times 1 \times 1$ box (dimensions are in inches)
2. Create the assembly
3. Select the assembly dimension to analyze (the assembly height in this example)
4. Define the assembly sequence. This sequence relates the dimension selected in Step 3 to the dimensions of the assembly components (parts) created in Step 1
5. Set the assembly constraints
6. View the analysis results

The details of each step are shown below. All dimensions are in inches.

Step 1: Create *Sketch1* and *Boss-Extrude1-Box*: **File > New > Part > OK > Front Plane** > sketch and dimension rectangle shown > exit sketch > enter 1 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As >** *box* **> Save.**



Step 2: Add **DimXpert** dimension: **Tools > DimXpert > Auto Dimension Scheme >** bottom face of box as **Primary Datum >** ✔.

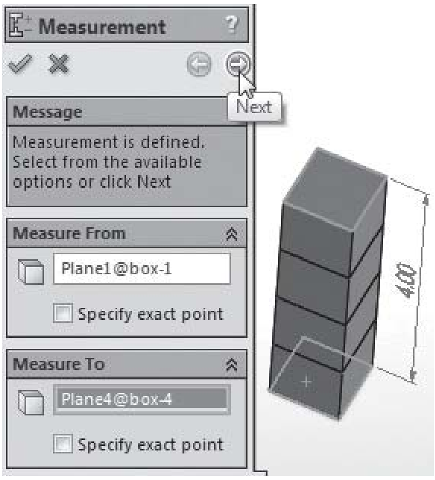


Note: **DimXpert** adds the height (tolerance dimension) shown automatically. Click the dimension to edit the tolerance, if needed. **DimXpert** dimensions are shown in pink to distinguish them from other dimensions.

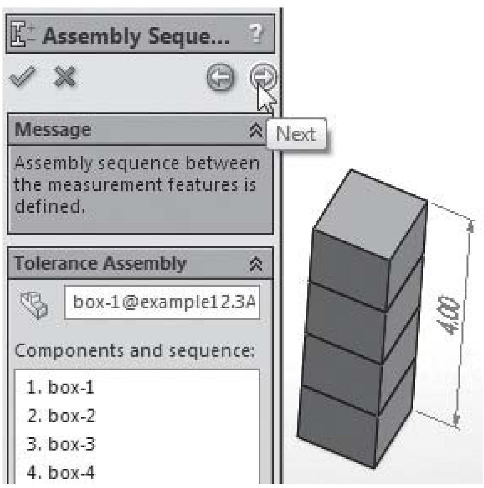
Note: We use the height as the desired dimension because it controls the assembly as discussed in Example 12.2.

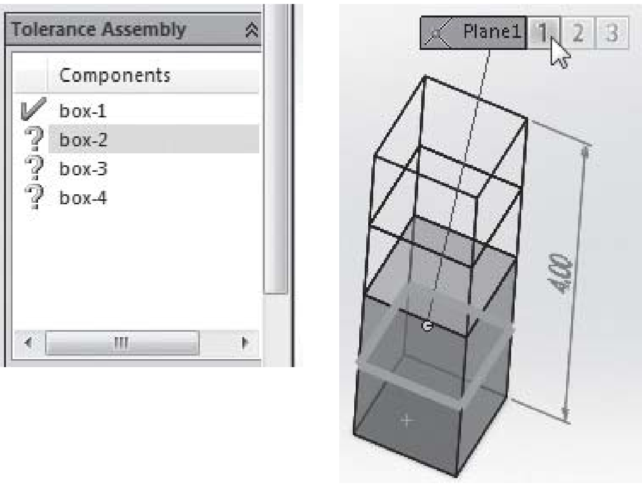
Step 3: Create assembly: **File** > **New** > **Assembly** > insert four instances of box > use two coincident mates between edges of instances to assemble and stack them up as shown in Figure 12.15.

Step 4: Select the assembly dimension to analyze: **Tools** > **Add-ins** > **TolAnalyst** > **OK** > **Tools** > **DimXpert** > **TolAnalyst Study** > bottom face of assembly > top face of assembly > click on screen to place dimension > ✔.

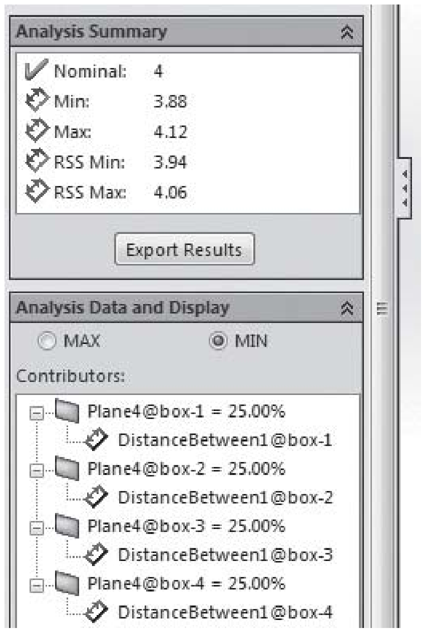


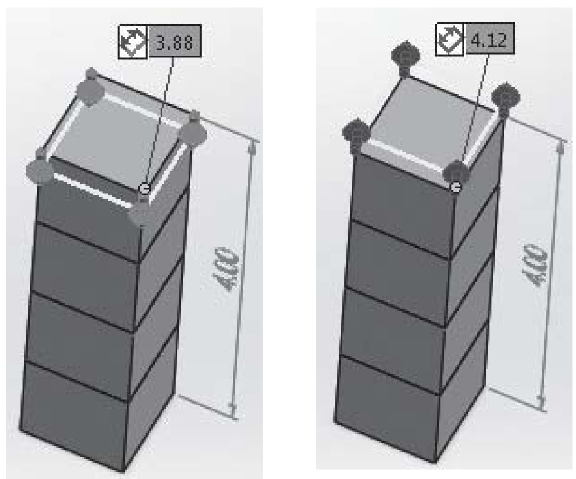
Step 5: Define the assembly sequence: **Next** (arrow shown) > select four components (from assembly tree or graphics pane), in order, from bottom to top.





Step 6: Set the assembly constraints: **Next** (arrow shown in Step 5) > **box-2** shown above > **1** from the callout (shown above) that pops up to set constraint between **box-2** and **box-1** > repeat for **box-3** > and **box-4** > ✔.





Note: We apply constraints between mating faces of assembly components. These constraints determine how the dimensions of the assembly components change relative to the assembly dimension.

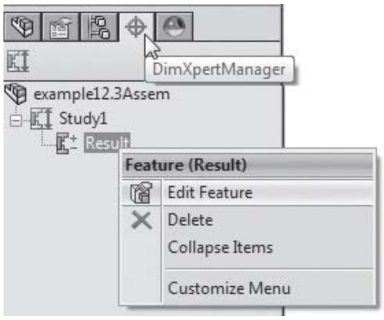
Note: Each constraint relates two planes, one from each instance.

Note: Constraints are different from the assembly mates. They control how the dimensions change in the tolerance stack-up analysis.

Step 7: View and analyze the tolerance analysis results: **Next** (arrow shown in Step 5) > view results shown > ✔.

Note: Each box plane contributes equally (25%) to tolerance stack-up analysis.

Step 8: Edit study: **DimExpertManager** tab shown > expand **Study1** node > right-click **Result** > **Edit Feature** > ✔ when done.



HANDS-ON FOR EXAMPLE 12.3. Double the box size to 2 × 2 × 2 and redo the tolerance stack-up analysis. How do the results change? Is it a linear change? Explain your answer.

12.15 Tutorials

The theme for the tutorials in this chapter is to practice specifying both conventional and geometric tolerances as well as performing tolerance stack-up analysis.

Tutorial 12–1: Create Conventional Tolerances

There are three methods to add conventional tolerances to a drawing as shown in Figure 12.16:

1. Tolerance selected dimensions
2. Add tolerance note in drawing title block
3. Add tolerance note in drawing

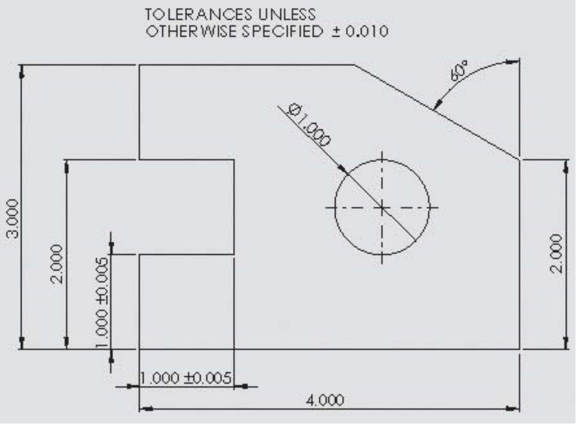
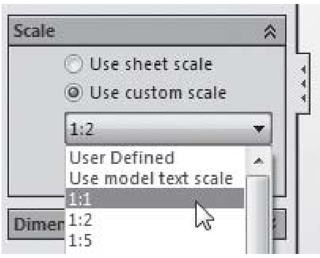
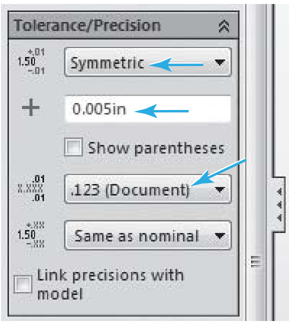


FIGURE 12.16
Conventional tolerances

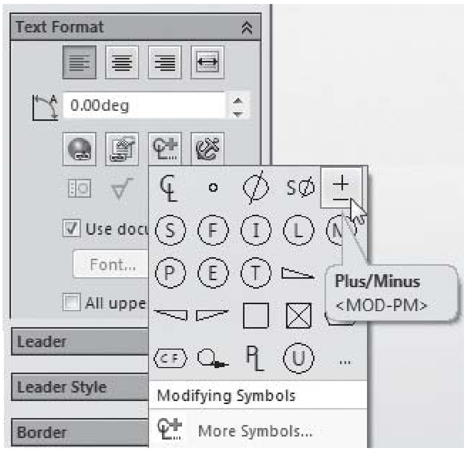
Step 1: Create drawing: **File** > **New** > **Drawing** > **OK** > **OK** (accept the default **Standard sheet size**) > **Browse** (locate and select *tutorial3.1PlanA* part from Chapter 3) > click in drawing sheet to insert front view > **Esc** key > click view just added > **Scale** section shown > **Use custom scale** > 1:1 from dropdown > ✔ > click *SCALE 1:1* note that shows up underneath view > **Delete** key on keyboard > **File** > **Save As** > *tutorial12.1* > **Save.**



Step 2: Add dimensions with tolerances: **Model Items** on **Annotation** tab > **Selected Feature** under **Source/Destination** > click view > ✔ > horizontal dim **1.000** shown in Figure 12.16 > set the parameters for **Tolerance/Precision** section as shown > repeat for vertical 1.00 dim > ✔.



Step 3: Add tolerance note: **Note** on **Annotation** tab > click in sheet to place note > type text shown in Figure 12.16 in note box > **Add Symbol** icon in **Text Format** section shown on left pane > symbol icon shown > **Plus/Minus** icon shown to add symbol to note shown in Figure 12.16 > ✔.



> ***HANDS-ON FOR TUTORIAL 12–1.*** Edit the title block to add a tolerance general note in the **Comments** box of the title block. The note should read:
>
> GENERAL TOLERANCE
> .X ± .030
> .XX ± .010
> .XXX ± .005
> .XXXX ± .0005

Tutorial 12–2: Create Geometric Tolerances

We add geometric tolerances to the drawing shown in Figure 12.17. All dimensions are in mm.

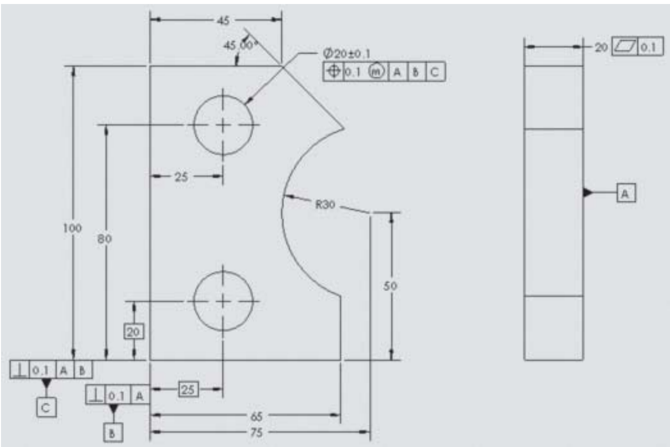
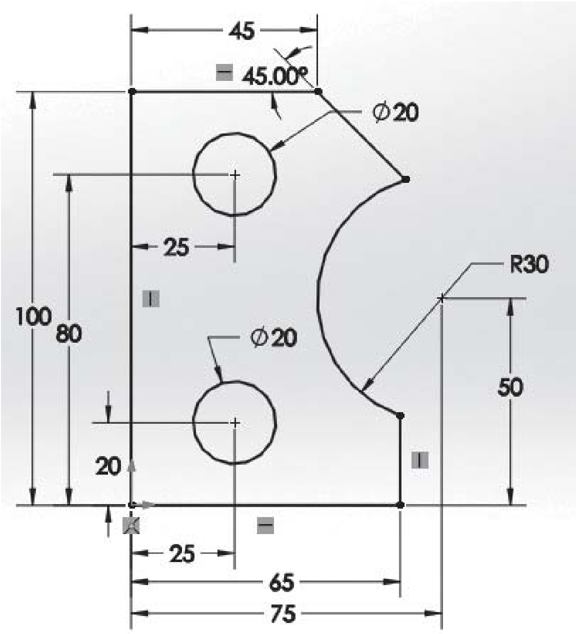
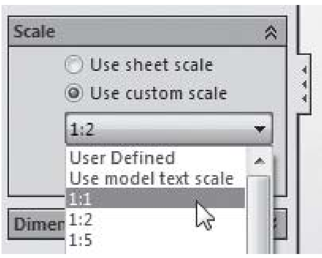


FIGURE 12.17
Geometric tolerances

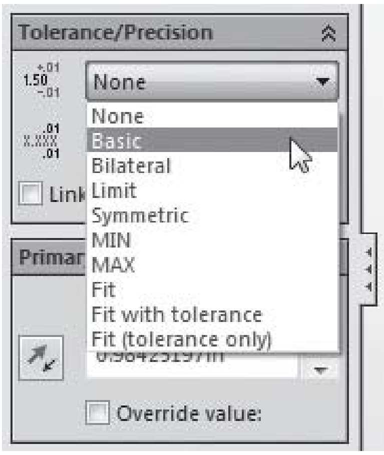
Step 1: Create part: **Open** > **New** > **Part** > **OK** > **Front Plane** > **Extruded Boss/Base** on **Features** tab > sketch as shown > exit sketch > enter 0.75 for thickness **D1** > reverse extrusion direction > ✔ > **File** > **Save As** > *tutorial12.2* > **Save.**



Step 2: Create drawing: **File** > **New** > **Drawing** > **OK** > **OK** (accept the default **Standard sheet size**) > **Browse** (locate and select *tutorial12.2* part from Step1) > click in drawing sheet to insert front and right views > **Esc** key > click view just added > **Scale** section shown > **Use custom scale** > 1:1 from dropdown > ✔ > click *SCALE 1:1* note that shows up underneath each view > **Delete** key on keyboard > **File** > **Save As** > *tutorial12.2* > **Save.**

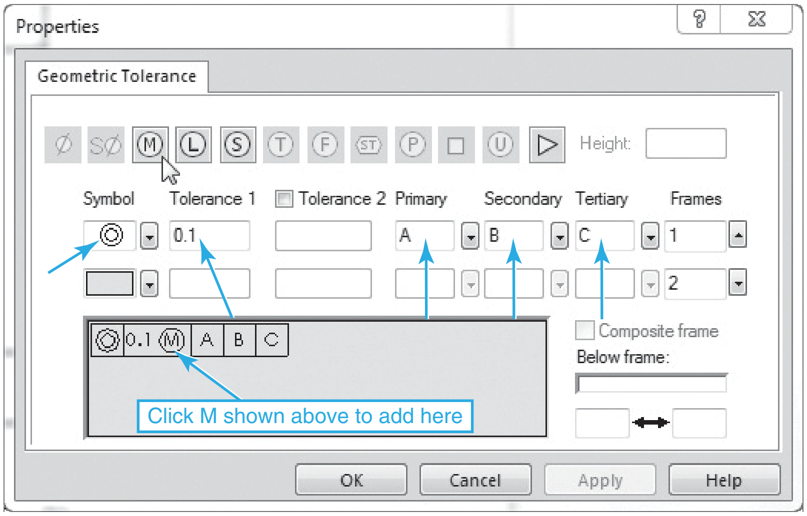


Step 3: Add dimensions and create basic sizes: **Model Items** on **Annotation** tab > **Selected Feature** under **Source/Destination** > click front view > ✔ > horizontal dim **25** shown in Figure 12.17 > **Basic** from

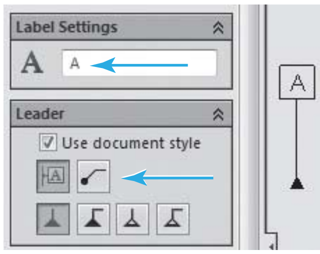


Tolerance/Precision dropdown shown > repeat for vertical 20 dim > ✔.

Step 4: Add geometric tolerances: Click in sheet to place tolerance > **Geometric Tolerance** on **Annotation** tab > select symbol shown > type 0.1, A, B, C shown > **OK** > drag and place tolerance block as shown in Figure 12.17 > repeat for other geometric tolerances > ✔.



Step 5: Add datum features: **Datum Feature** on **Annotation** tab > edit **Leader** attributes if needed > click once on sheet to place (anchor) feature > drag mouse and click again to position features > ✔ > drag feature and attach to intended part feature as shown in Figure 12.17 > repeat for other datums.

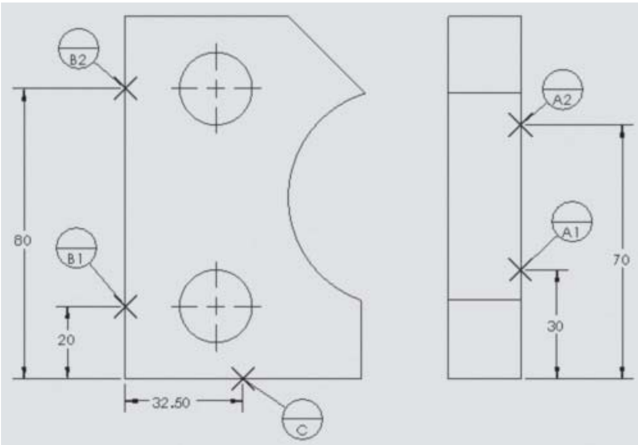


HANDS-ON FOR TUTORIAL 12–2. The top face of the block must be parallel to its bottom face and perpendicular to its left face within 0.1 for functional requirements. Add two geometric tolerances to the top face to specify these two requirements.

Tutorial 12–3: Define Datum Targets

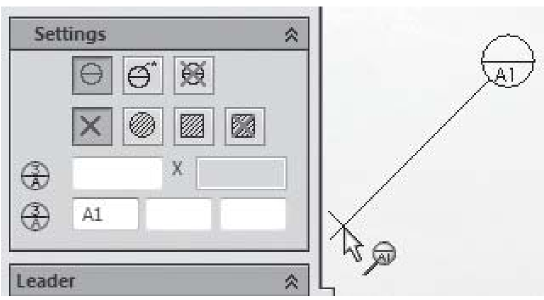
We create datum targets to define datums A, B, and C shown in Figure 12.17. We define datums A and B by two target points each and define datum C by one target point. Figure 12.18 shows these targets. These targets are used to build inspection gauges and are used during the part inspection. The inspector slides the part onto the inspection gauge and pushes the part against the faces of the gauge and observes the points of contact between the two. If the part is in complete contact with the gauge at the locations specified by the target points, the part passes inspection; otherwise, it is a reject and becomes scrap.

FIGURE 12.18
Datum targets



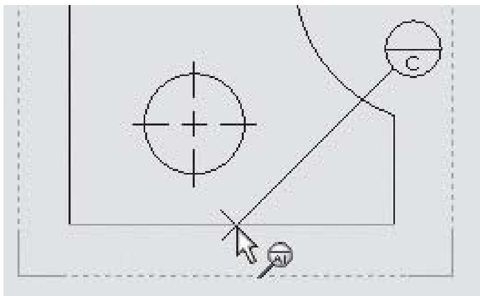
Step 1: Create drawing: **File** > **Open** > locate *tutorial2.2* file > **Open** > delete all dimensions > **File** > **Save As** > *tutorial12.3* > **Save.**

Step 2: Add datum targets for datum A: **Datum Target** on **Annotation** tab > type A1 shown > click on right edge of block to anchor target (Figure 12.18) > click again to place (Figure 12.18) > repeat to create A2 target > ✔ > **Smart Dimension** on **Annotation** tab > dimension target locations as shown in Figure 12.18 > ✔.



Step 3: Add datum targets for datum B: Repeat Step 2, but type B1 and B2 instead of A1 and A2.

Step 4: Add datum target for datum C: **Datum Target** on **Annotation** tab > type C for target name > move mouse close to bottom edge until its color changes, indicating that you snapped to it > click edge to anchor target there > click again to place (Figure 12.18) > ✔ > **Smart Dimension** on **Annotation** tab > dimension target location as shown inFigure 12.18 > ✔.



HANDS-ON FOR TUTORIAL 12–3. Change the datum targets for datum A to use a circular area of contact with diameter of 2 mm. Change the datum targets for datum B to use a square area of contact with a side length of 2 mm.

Tutorial 12–4: Tolerance a Taper

We use the tolerance techniques discussed in this chapter to tolerance a flat taper and a conical taper, as shown in Figure 12.19, per the ASME Y14.5M-1994 standard; the new revision is ASME Y14.5-2009.

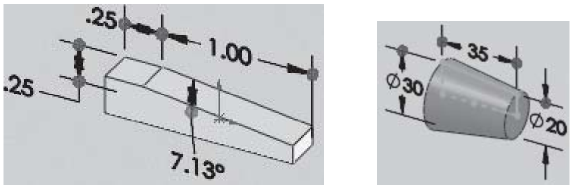
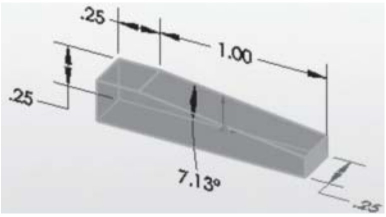
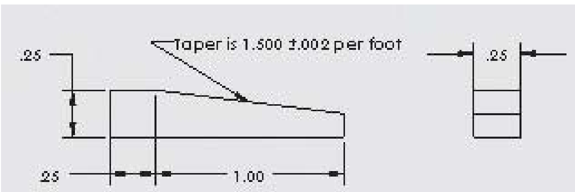


FIGURE 12.19
Flat and round tapers

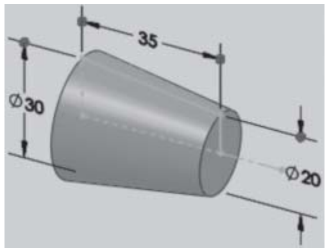
Step 1: Create flat taper: **File** > **New** > **Part** > **OK** > create part shown using inches units > **File** > **Save As** > *tutorial12.4A* > **Save.**



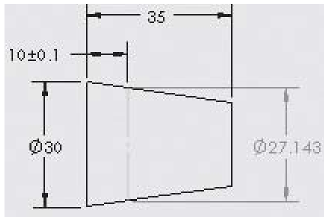
Step 2: Tolerance taper per ANSI standard: **File** > **New** > **Drawing** > **OK** > **OK** (accept the default **Standard sheet size**) > **Browse** (locate and select *tutorial2.4A* part) > add tolerance label shown > ✔.



Step 3: Create conical taper: **File** > **New** > **Part** > **OK** > create part shown using mm units > **File** > **Save As** > *tutorial12.4B* > **Save.**



Step 4: Use scheme 1 to tolerance taper: **File** > **New** > **Drawing** > **OK** > **OK** (accept the default **Standard sheet size**) > **Browse** (locate and select *tutorial2.4B* part**)** > add tolerances shown > ✔.



Note: Sketch a vertical line on the view from top to bottom. Dimension the line as shown. One of the dimensions is driven.

Note: This method of tolerance applies a critical accuracy to a specific location of taper.

HANDS-ON FOR TUTORIAL 12–4. The tutorial uses scheme 1 shown in Figure 12.6A to tolerance the cross section located by *L* and *d*. Use Tolerancing scheme 2 shown in Figure 12.6A to tolerance the taper (i.e., add tolerances to *L*, *D*, and *d*). Use a symmetric tolerance amount of 1 mm for the three dimensions.

Tutorial 12–5: Perform Tolerance Stack-up Analysis

We use **TolAnalyst** to perform tolerance stack-up of the assembly shown in Figure 12.20. All dimensions are in inches. We analyze the effect of stacking a tolerance amount of 0.02 inch on the block depth when we assemble two instances as shown in Figure 12.20. **TolAnalyst** creates a study to store the results of the stack-up analysis.

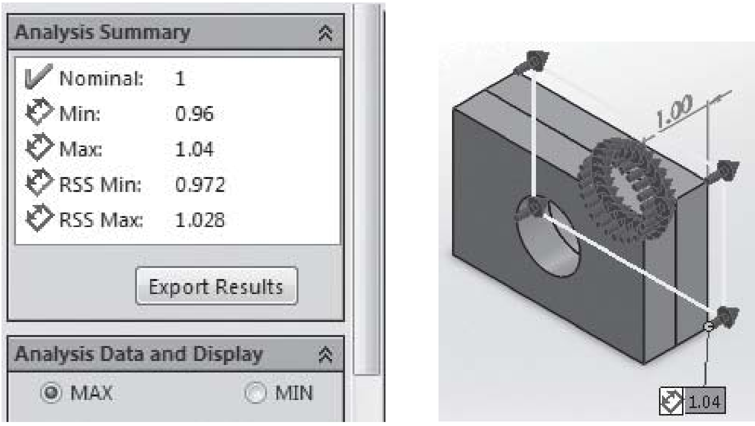
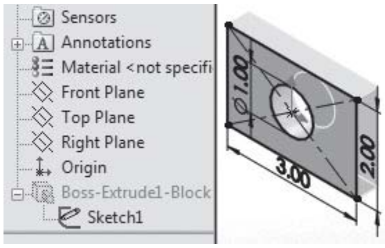
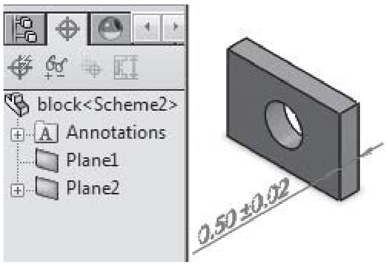


FIGURE 12.20
TolAnalyst stack-up results

Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK > Front Plane > Extruded Boss/Base** on **Features** tab > sketch and dimension as shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > **File** > **Save As** > *block* > **Save.**



Step 2: Add **DimXpert** dimension: **Tools > DimXpert > Auto Dimension Scheme** > front face of block as **Primary Datum** > ✔.



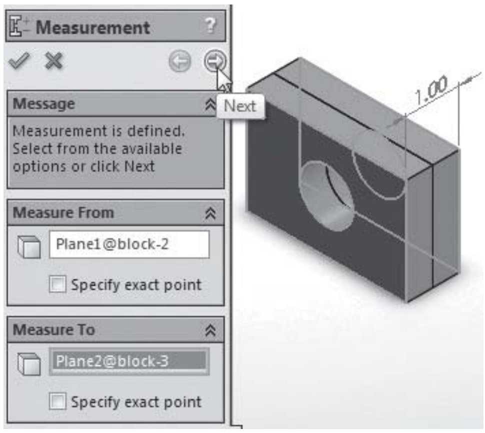
Note: **DimXpert** adds the thickness (toleranced dimension) shown automatically. Click the dimension to edit the tolerance, if needed.

Note: We use the block depth as the desired dimension because it controls the assembly.

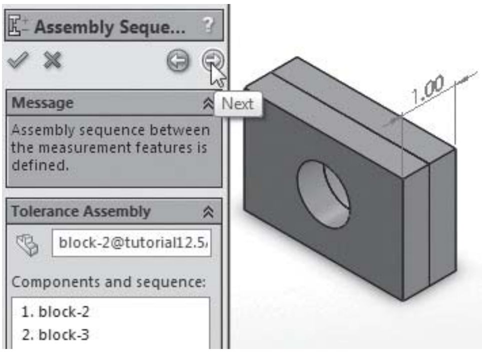
Note: **DimXpert** includes hole diameter in the graphics pane and feature under *Plane2* shown. Delete both of them.

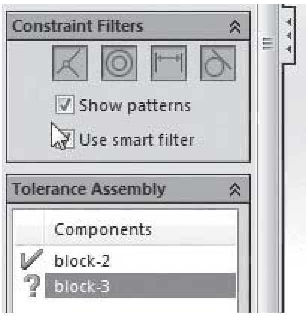
Step 3: Create assembly: **File** > **New** > **Assembly** > insert two instances of block > use two coincident mates between edges to assemble and stack them up as shown in Figure 12.20.

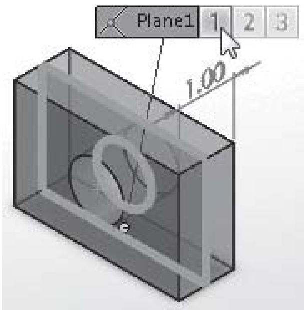
Step 4: Select the assembly dimension to analyze: **Tools** > **Add-ins** > **TolAnalyst > OK > Tools** > **DimXpert** > **TolAnalyst > Study** > front face of assembly > back face of assembly > click on screen to place dimension > ✔.



Step 5: Define the assembly sequence: **Next** (arrow shown) > select two components (from assembly tree or graphics pane), in order, from front to back.





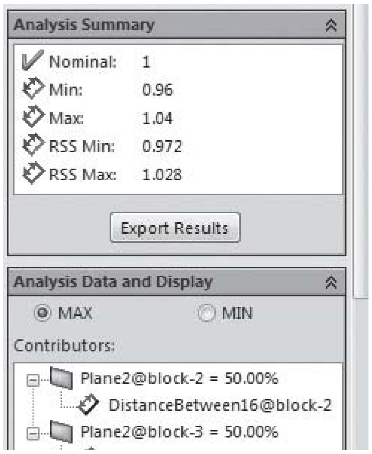


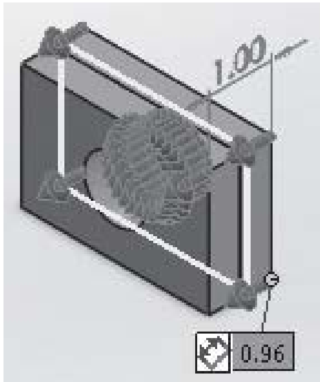
Step 6: Set the assembly constraints: **Next** (arrow shown in Step 5) > **block-3** shown above > **1** from the callout (shown above) that pops up to set constraints between **block-2** and **block-3** > ✔.

Note: We apply constraints between mating faces of assembly components. These constraints determine how the dimensions of the assembly components change relative to the assembly dimension.

Note: Each constraint relates two planes, one from each part.

Note: Constraints are different from the assembly mates. They control how the dimensions change in the tolerance stack-up analysis.

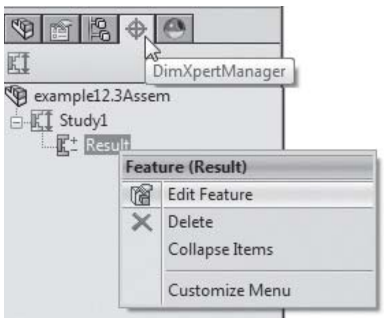




Step 7: View and analyze the tolerance analysis results: **Next** (arrow shown in Step 5) > view results shown above > ✔.

Note: Each box plane contributes equally (50%) to tolerance stack-up analysis.

Step 8: Edit study: **DimExpertManager** tab > expand **Study1** node > right-click **Result** > **Edit Feature** > ✔ when done.



HANDS-ON FOR TUTORIAL 12–5. Redo the tutorial without deleting the hole feature. Explain the results. Also, verify the results using Eqs. (12.12) and (12.16).

problems

1. List the sources of variability in manufacturing.
2. Why do we need tolerances?
3. List the two types of tolerances. What does each type control?
4. Inspection gauges are used to check whether a part is within its tolerance limits. Perform an in-depth research study on inspection gauges including their types, their design, and how they are used during part inspection.
5. What is the difference between a nominal and a basic size? Give an example.
6. Three types of tolerances exist: unilateral, bilateral, and symmetric. Describe each type. Give a numerical example for each type.
7. List the three types of material conditions. Explain each type.
8. What is a datum? Why do we need them? How is a datum defined?
9. Which ASME tolerance rule should you use to show dimension limits on a drawing? Sketch an example showing a shaft and hole dimensioning.
10. The basic size of a dimension and its tolerance are specified as 3.500 in. and ±0.05, respectively. Is this specification correct? Explain your answer.
11. How do you fully define a conical or flat taper? How do you tolerance it?
12. List and describe the three types of fit.
13. The three parameters (tolerance zones) that determine a fit are *s*, *h*, and *a*. Sketch the combination of the three for clearance, transition, and interference fits.
14. Calculate manually the limits and tolerance zones for the following fits:
    a. Clearance fit of RC5 for a basic size of 5.0000 in.
    b. Transition fit of LT3 for a basic size of 3.0000 in.
    c. Interference fit of FN4 for a basic size of 6.0000 in.
    Use SolidWorks to verify the manual results.
15. Figure 12.21 shows a stepped and grooved block sketch with a depth of 2 in. Apply the three dimensioning schemes of Section 12.7 (shown in Figure 12.9) to the sketch. Calculate, manually, the coordinates of point *P* for each scheme. Which scheme has the worst tolerance accumulation on the point location? Calculate the accumulation for each scheme. *Note:* All missing dimensions are either 1 in. or 0.5 in.

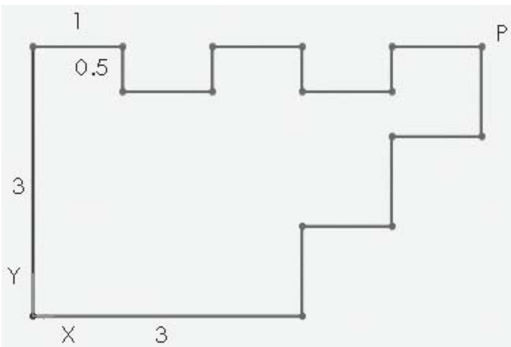


FIGURE 12.21
Stepped and grooved block

16. What is the difference between arithmetic and statistical tolerancing? Which one is more restrictive and why?

17. Figure 12.22 shows a toleranced hole. What is the problem with this tolerancing method? How do you solve it? All dimensions are in mm.

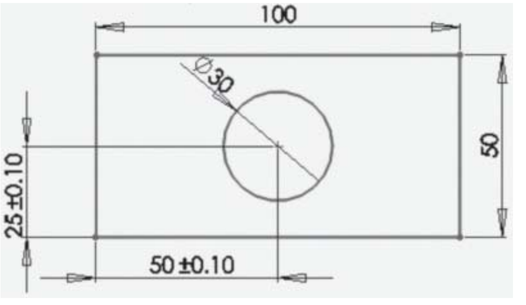


FIGURE 12.22
Specifying tolerances on a hole position

18. What type of geometric tolerance should you use to control the following variability in a part?
    - **a.** The orientation of one of its planes
    - **b.** How straight a surface is
    - **c.** How flat a surface is
    - **d.** How parallel two surfaces are, and
    - **e.** How perpendicular two surfaces are

    Illustrate your answer by sketching, manually, the tolerance symbol for each on a block with a rectangular cross section. Also, sketch the interpretation of each tolerance. Follow the format shown in Table 12.4.

19. Define and sketch, manually, datum targets for the datums you use in Problem 18.

20. Figure 12.23 shows a four-bar mechanism assembly. Each bar of the assembly has the tolerance shown. The mechanism is fixed at point *A*, and its free end is *D* and can only slide horizontally. Calculate the tolerance on *D* when the mechanism is in the horizontal position. How would you calculate the tolerance in a nonhorizontal position, for example, when rod *AB* is at 45° from the horizontal? Use both the worst-case and statistical methods.

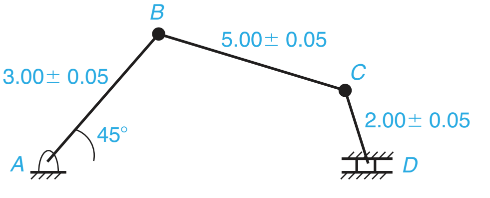


FIGURE 12.23
Four-bar mechanism

21. Figure 12.24 shows a hinge assembly. The functional requirement of the hinge requires a clearance fit, that is, $D > A + B + C$ for any assembly. The dimensions of the assembly components (parts) are 2.00 ± 0.05, 2.00 ± 0.05, 30.00 ± 0.05, and 35.00 ± 0.10 for *A*, *B*, *C*, and *D*, respectively. Does the tolerance specification on a component meet the assembly tolerance requirement? Use both the worst-case and statistical methods.

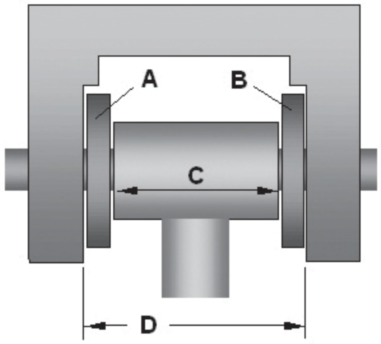
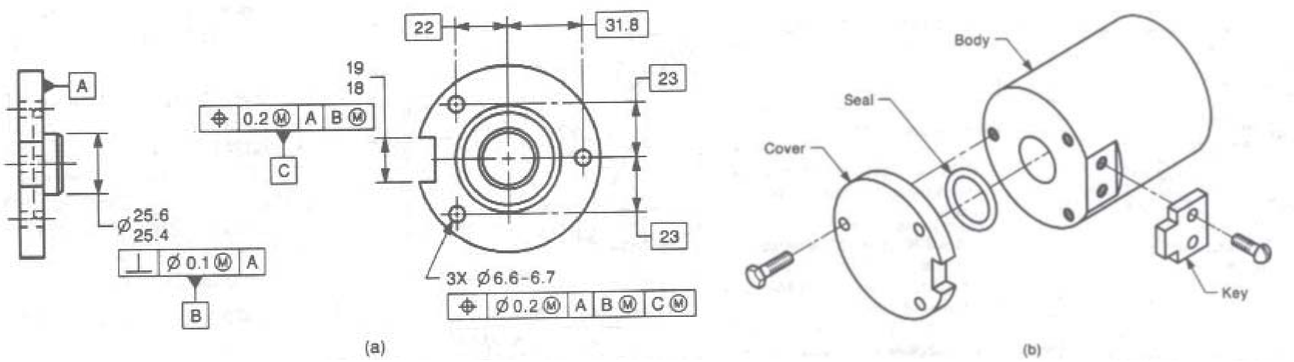


FIGURE 12.24
A hinge assembly

22. Solve Problem 15 using SolidWorks.
23. Solve Problem 20 using SolidWorks.
24. Solve Problem 21 using SolidWorks.
25. Use SolidWorks to create the geometric tolerances in Figure 12.25. Assume any missing dimensions. All dimensions are in mm.



(A) Drawing

(B) Model

FIGURE 12.25
Part where angular orientation is important

26. Use SolidWorks to create the geometric tolerances in Figure 12.26. Assume any missing dimensions. All dimensions are in mm.

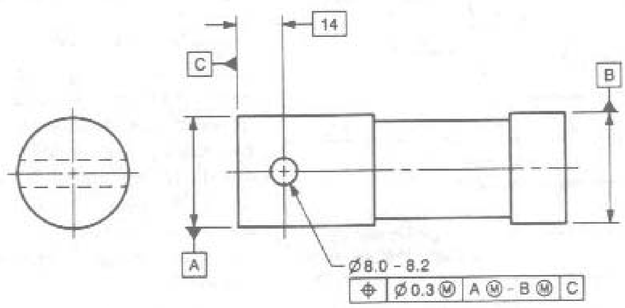
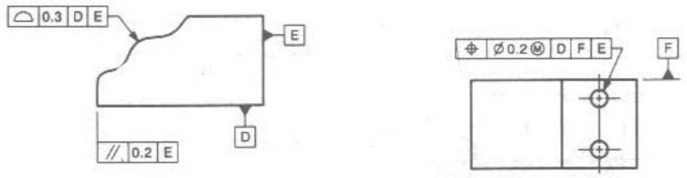


FIGURE 12.26
Two datum features, single datum axis

27. Use SolidWorks to create the geometric tolerances in Figure 12.27. Assume any missing dimensions. All dimensions are in mm.

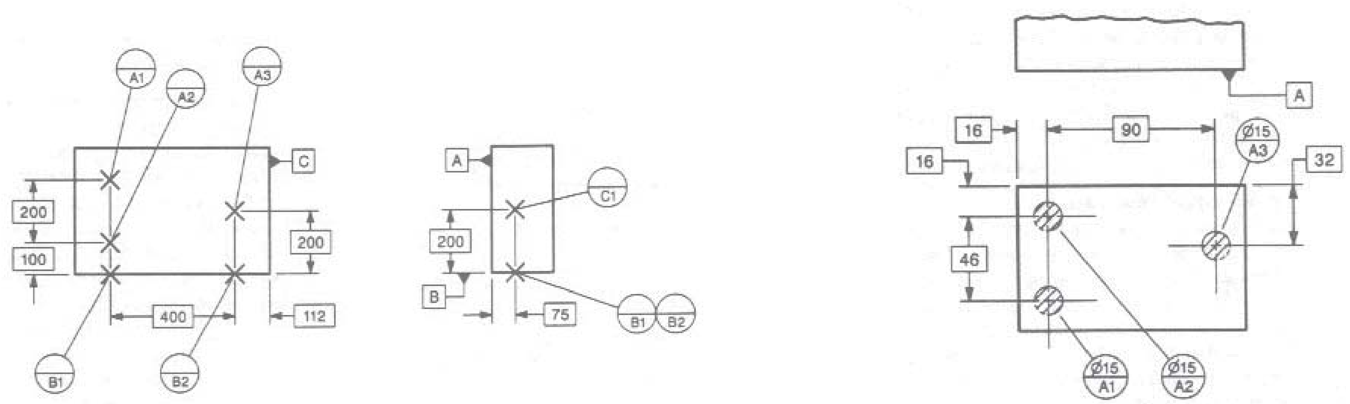


(A) Front view (B) Top view

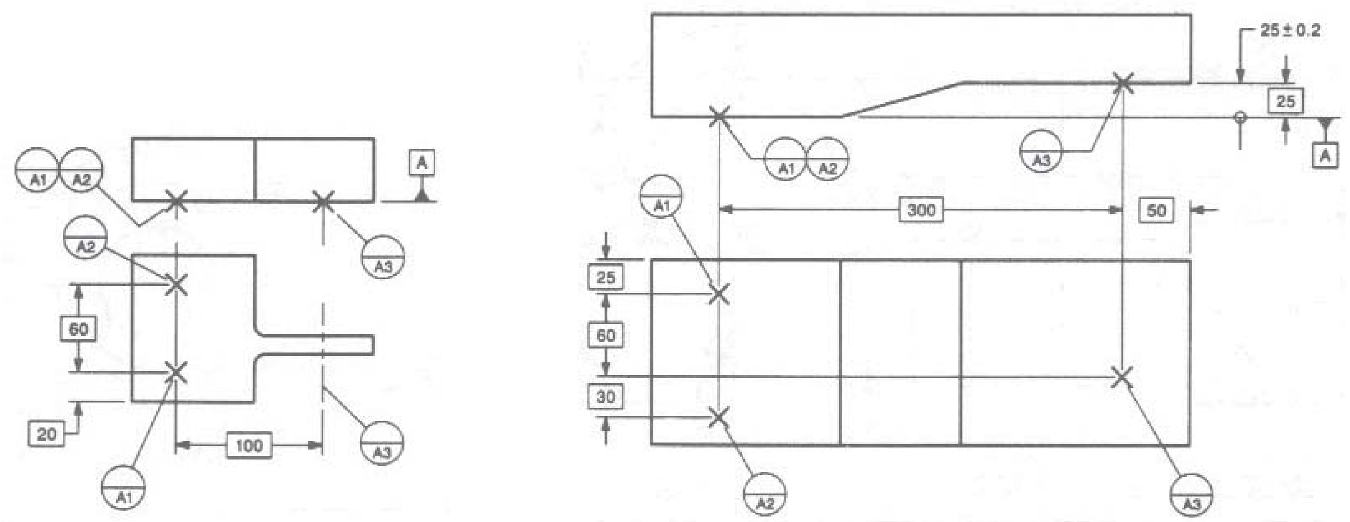
FIGURE 12.27

Using multiple datums

28. Use SolidWorks to create the datum targets in Figure 12.28. Assume any missing dimensions. All dimensions are in mm.



(A) Use target points (B) Use target areas



(C) Use target points and a target line (D) Use step datum feature

FIGURE 12.28

Defining datum targets

29. Use SolidWorks to create the geometric tolerances in Figure 12.29. Assume any missing dimensions. All dimensions are in mm.

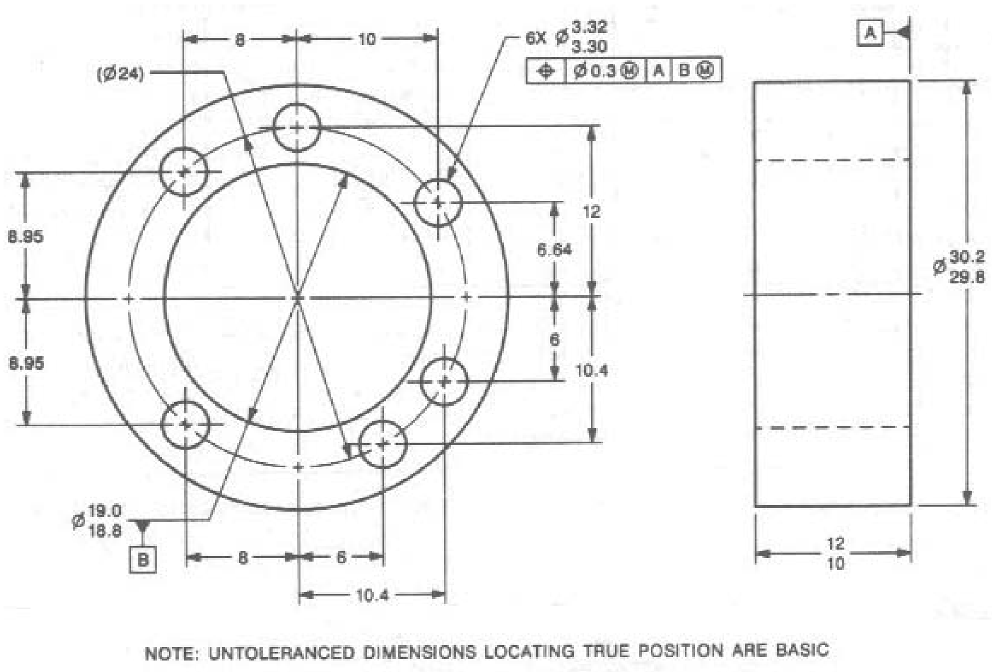


FIGURE 12.29

Position tolerance using two datums

30. Use SolidWorks to create the geometric tolerances in Figure 12.30. Assume any missing dimensions. All dimensions are in mm.

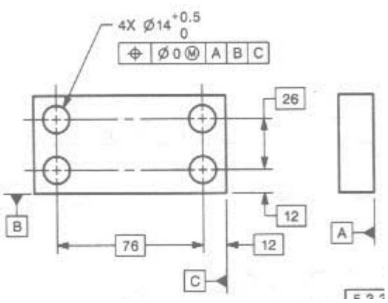


FIGURE 12.30

Zero position tolerance

31. Use SolidWorks to create the geometric tolerances in Figure 12.31. Assume any missing dimensions. All dimensions are in mm.

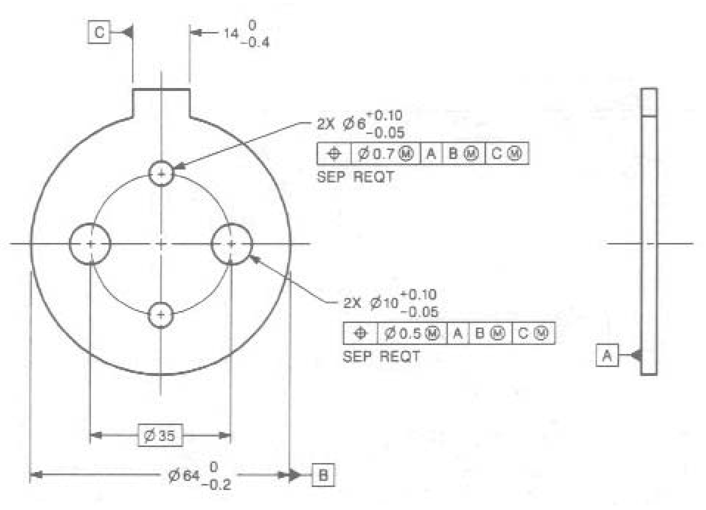


FIGURE 12.31

Multiple features, separate requirements

32. Use SolidWorks to create the geometric tolerances in Figure 12.32. Assume any missing dimensions. All dimensions are in mm.

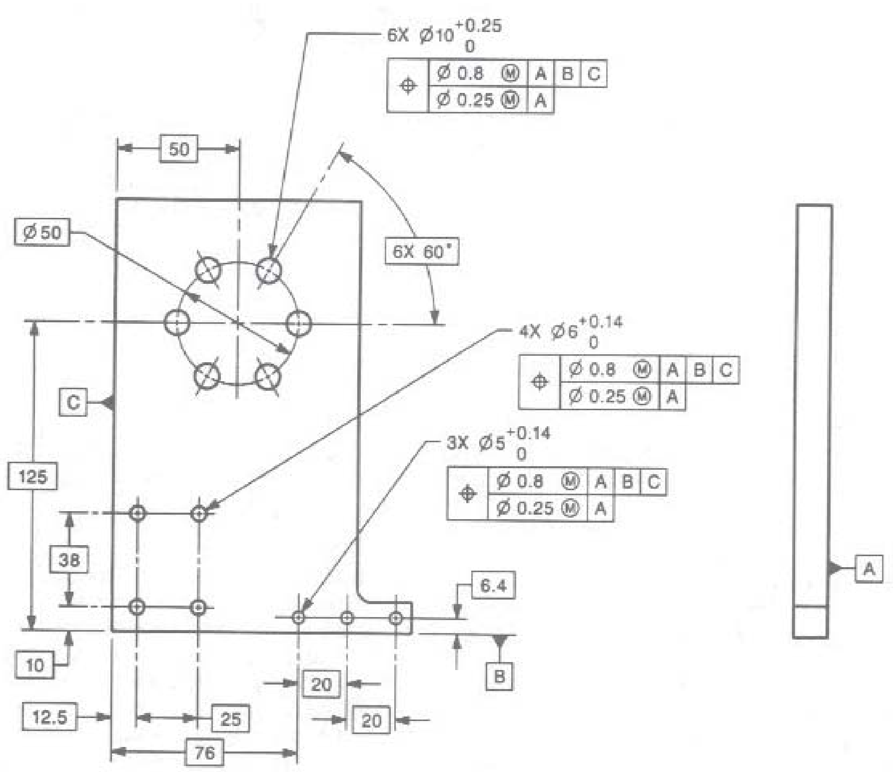


FIGURE 12.32

Hole patterns located by composite positional tolerances

33. Use SolidWorks to create the geometric tolerances in Figure 12.33. Assume any missing dimensions. All dimensions are in mm.

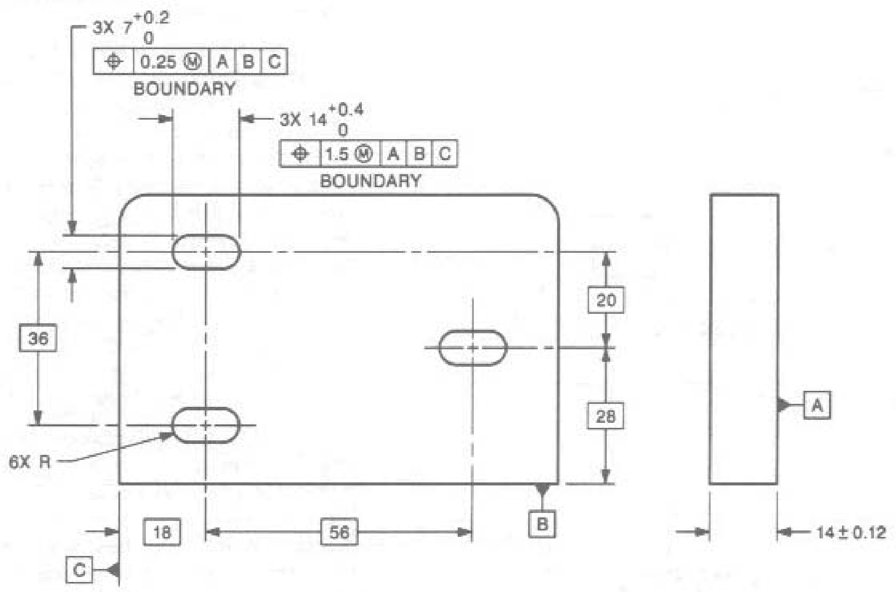


FIGURE 12.33

Using the BOUNDARY concept

(Reprinted from ASME Y14.5M-1994 [R2004], by permission of The American Society of Mechanical Engineers.

34. Use SolidWorks to create the geometric tolerances in Figure 12.34. Assume any missing dimensions. All dimensions are in mm.

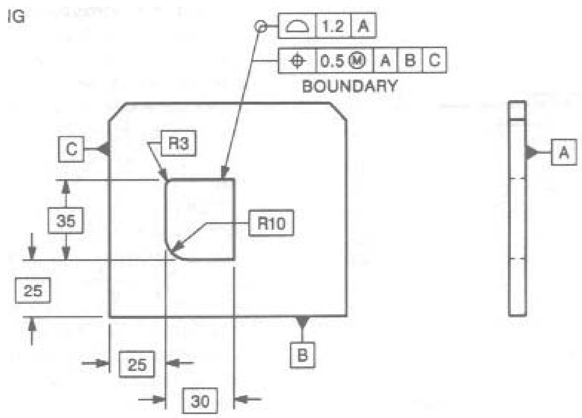
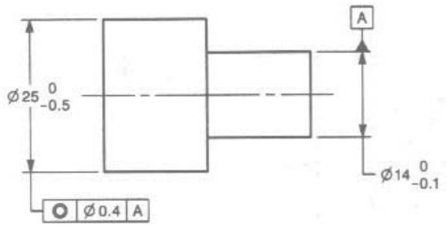


FIGURE 12.34

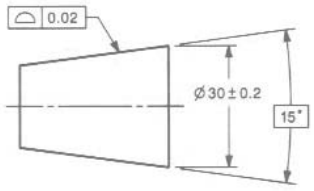
Specifying profile tolerance

(Reprinted from ASME Y14.5M-1994 [R2004], by permission of The American Society of Mechanical Engineers.

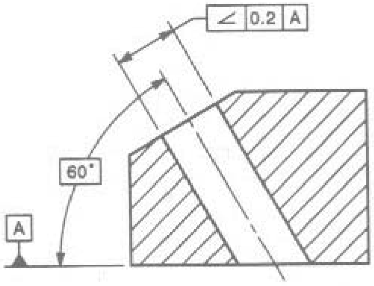
35. Use SolidWorks to create the geometric tolerances in Figure 12.35. Assume any missing dimensions. All dimensions are in mm.



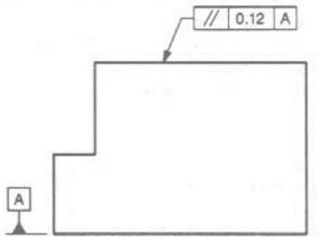
(A) Concentricity tolerance



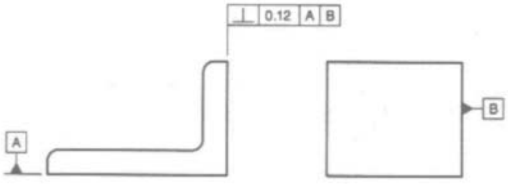
(B) Profile tolerance



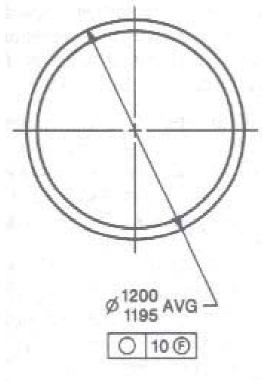
(C) Angularity tolerance



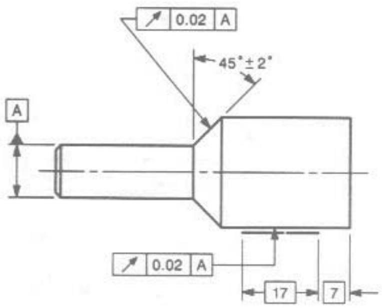
(D) Parallelism tolerance



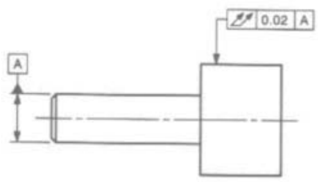
(E) Perpendicularity tolerance



(F) Circularity tolerance



(G) Runout tolerance



(H) Total runout tolerance

FIGURE 12.35

Specifying geometric tolerances

(Reprinted from ASME Y14.5M-1994 [R2004], by permission of The American Society of Mechanical Engineers. All rights reserved)

36. Interpret the geometric tolerances of Problem 35. Sketch the tolerance zone and the feature shape for each case of A through H. Use Table 12.4 style.

CHAPTER

13

Analysis Tools

13.1 Introduction

The creation of a geometric model is not an end in itself, but rather a means to study the part or assembly design to ensure that it meets the functional requirements of the design, be it stress analysis, fatigue analysis, thermal analysis, buckling, or any other. CAD/CAM systems offer a multitude of analysis tools that the designer can use to verify his or her design.

The crucial factor in using these analysis tools is understanding how to set up the problem correctly, use the right input parameters, and interpret the results. Each tool is based on and implements a mathematical model. The effective use of these tools requires that the designer be versed in the area of the analysis that the tools implement. For example, the designer must possess a sound understanding of the finite element method to effectively use the stress analysis tool of a CAD/CAM system.

Analysis tools include such applications as mass property calculations, finite element method, thermal analysis, fatigue analysis, frequency analysis, mold flow analysis, buckling analysis, simulation, and fluid flow analysis. We cover some of these analysis tools in this chapter. Although data exchange is not exactly considered an analysis tool, we cover it here because we consider it part of model activities after the model creation.

CAD/CAM systems offer analysis tools in three ways: built-ins, add-ins, or separate packages. *Built-in* tools come with the CAD/CAM system as part of the CAD/CAM software installation (e.g., mass properties is a built-in application). The user simply uses them. *Add-ins* also come with the CAD/CAM system, but the user needs to activate them. **Simulation** in SolidWorks is an example. *Separate packages* require separate installations and interface with CAD/CAM systems via file share. For example, a specialized analysis package may read a CAD/CAM file and perform its analysis on it.

13.2 Data Exchange

Data exchange is routinely done in practice to transfer CAD/CAM files from one system to another; for example, a company that outsources its manufacturing needs to send its CAD models to its subcontractors who may not use the same CAD/CAM system. Data exchange solves the following problem. How do you transfer CAD/CAM models from one CAD/CAM system to another? Each CAD/CAM system, like any application, has its own proprietary native file format; that is, the system saves the model (part) in its own format. Such a file is not transferable to another system; other systems cannot open or read it.

The common solution to solving this data exchange problem is to save the files in "neutral" formats that are commonly read by all CAD/CAM systems. Some of these formats are standard whereas others are *de facto* standards. Examples of standard neutral file formats are STEP and IGES. Examples of *de facto* formats are AutoCAD DXF and ACIS. SolidWorks supports all these formats and more. Another group is image and Web formats. Image format includes saving a model as a JPEG image. Web format includes saving a model as a VRML (virtual reality markup language) model.

Data exchange occurs between two CAD/CAM systems. The user exports or imports a CAD file from one system to another. Consider a file exchanged from system A to system B. We say the user exports the file from system A when the user saves it in system A in a neutral file format. Conversely, we say the user imports the file to system B when the user reads the neutral file into system B, thus converting it to the native format of system B. In summary, we say the user exports the file from system A and imports it to system B.

Data exchange comes with potential problems. Each CAD/CAM system has two translators: one to export its native format to neutral format, and an opposite one to import neutral format to its native format. Translators for standards such as IGES and STEP are developed and written by interpreting sets of ISO and ANSI standards. Such interpretations open the door for translation errors. Thus, translating model data from their native format to neutral format, and vice versa, has the potential of data loss. Thus, the exchanged files need to be carefully inspected, including visual inspection, to ensure no data loss or no corruption occurred during the exchange process. CAD/CAM systems developers utilize multiple tests to check the accuracy of their translators.

Figure 13.1 shows all the possible file formats that SolidWorks supports. Table 13.1 presents a brief description of some of the formats and when to use them. Click **File** (menu) > **Save As** to access these formats.

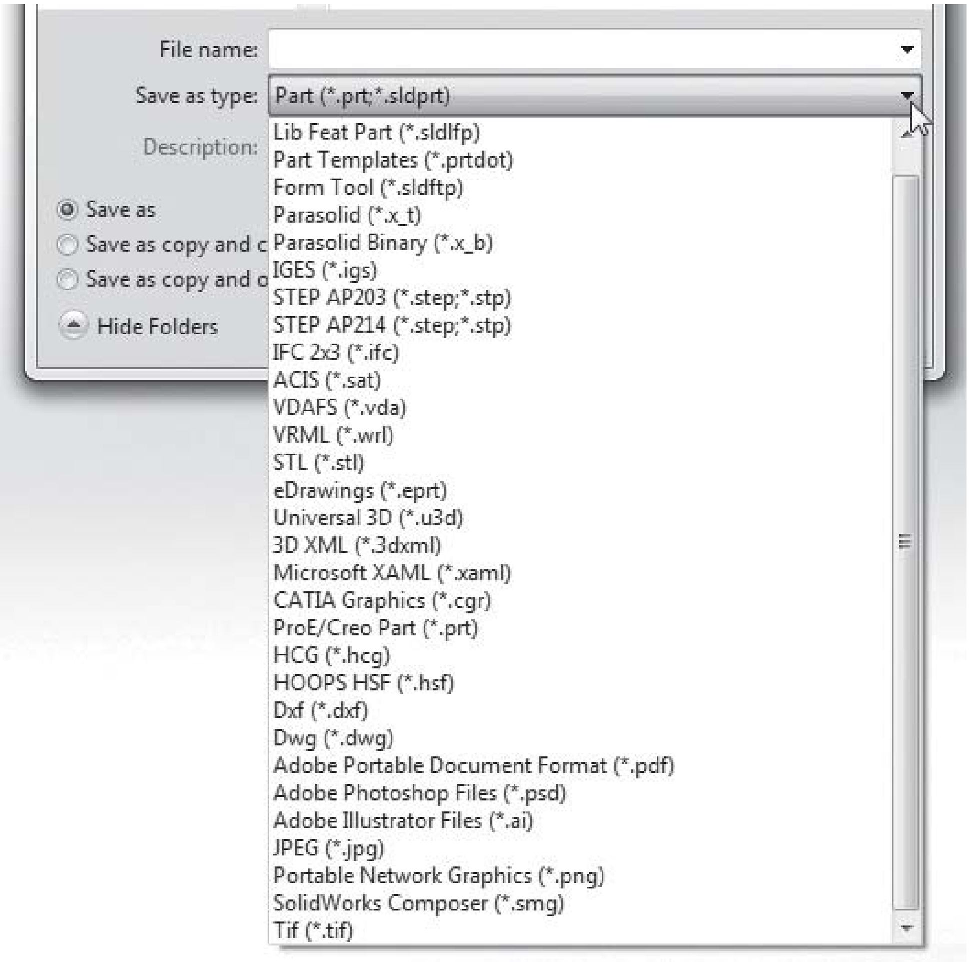


FIGURE 13.1
SolidWorks file formats

TABLE 13.1 SolidWorks File Exchange Formats

File Type	Description
.sldprt	Part file. This is the most commonly used format.
.sldfp	Library feature for reuse. The feature becomes usable in the future.
.prtdot	Template or master for reuse, similar to saving a Word template.
.sldftp	Form tool part that is used in sheet metal work. SolidWorks saves the tool in the Design Library. Use the tool to deform a sheet metal (e.g., make complex dents).
.x_t	Parasolid text file. Parasolid is a format created by Unigraphics, similar to AutoCAD DXF format.
.x_b	Parasolid binary file. Parasolid is a format created by Unigraphics, similar to AutoCAD DXF format.
.igs	IGES file. IGES is a standard neutral file format supported by all CAD/CAM systems.
.step	STEP file. STEP is a standard neutral file format supported by all CAD/CAM systems. STEP is more comprehensive and encompassing than IGES.
.sat	ACIS file. ACIS is a *de facto* standard. ACIS is a powerful geometric engine that is used and/or supported by multiple commercial CAD/CAM systems.
.vda	VDAFS file. This neutral file format is used to exchange surface geometry. STEP replaces both IGES and VDAFS.
.wrl	VRML file. This is a Web format that enables viewing CAD models in a Web browser. You need to download and install a VRML plug-in for the browser.
.stl	STL file. This is the standard prototyping format. A prototyping machine reads the STL file to build the prototype.
.eprt	eDrawing file. You can view the file in a Web browser. Chapter 2 has more details.
.pdf	PDF file. Convert the model display into a PDF file for ease of communication.
.u3d	Universal 3D file. You can open a *.u3d* file in Adobe Acrobat 3D software; or you can insert *.u3d* files into a PDF document and view them in Acrobat Reader.
.3dxml	3DXML file. 3DXML is a proprietary 3D file format developed by Dassault Systemes, the parent company of SolidWorks, to enable file transfer between SolidWorks and Dassault CATIA CAD/CAM systems.
.psd	Photoshop file. Convert a SolidWorks file to a Photoshop file for use in media.
.ai	Adobe Illustrator file. Use this file in company marketing material, brochures, manuals, and other documents.
.xaml	Microsoft XML file. XAML is a modified version of the standard XML format. Use an XML parser to process (parse) the file.
.cgr	CATIA graphics file. CGR is a proprietary file format developed by Dassault Systemes, the parent company of SolidWorks, to enable file transfer between SolidWorks and Dassault CATIA CAD/CAM systems. CGR is a triangulation form of the 3D model.
.prt	Pro/E file. This is a direct translation between Pro/E and SolidWorks.
.jpg	JPEG file. Converts 3D model to an image.
.hcg	HCG file. HCG (highly compressed graphics) is a proprietary file format developed by Dassault Systemes, the parent company of SolidWorks, to enable file transfer between SolidWorks and Dassault CATIA CAD/CAM systems in CATWeb.
.hsf	HOOPS file. HOOPS is a format that allows streaming graphics files (i.e., downloads the file piece by piece to a Web page). It is helpful for displaying large files.
.dxf	DXF file. This format is an AutoCAD *de facto* standard. It is a direct translation between SolidWorks and AutoCAD.
.dwg	DXF/DWG file. This is an AutoCAD file format.
.tif	TIF file. Another image format similar to JPEG.

Out of the formats listed here, we use only a handful. We recommend using STEP to exchange CAD models, eDrawings to read files in a Web browser, and JPEG to generate images of CAD models to include in reports. SolidWorks offers two versions of STEP: AP203 and AP214. AP203 is used in any mechanical CAD sector, whereas AP214 is used specifically in the automotive industry. Other formats may be used on an as-needed basis; for example, use DXF/DWG formats if you deal with AutoCAD files.

SolidWorks performs checks when it imports a neutral file. It asks the user a series of questions to validate its translation of the file. We can perform the following simple test to experience this translation process. Create a box. Save it as STEP AP203 format. Then read it back into SolidWorks. SolidWorks asks whether to run diagnostics on the part. If we confirm, SolidWorks displays any faulty faces and/or any gaps between faces.

Finally, it asks whether we need to proceed with feature recognition. If we confirm, it maps the features stored in the neutral files to its native format and creates a SolidWorks native features tree. We refer to this imported solid as a "smart" solid. If we decline the feature recognition step, SolidWorks imports what we refer to as a "dumb" solid. Such a solid has only one node in the features tree named *Importedx* (e.g., *Imported1*, *Imported2*); thus its features and sketches are not available to edit.

The understanding of the CAD file translation process requires knowledge of the boundary representation (B-rep) of a CAD model. A **B-rep** defines a CAD model by its boundary elements, that is, vertices (points), edges (curves), and faces (surfaces). Each model feature we create consists of this B-rep. A B-rep also maintains the CAD model topology. This understanding of B-rep explains the SolidWorks translation process.

The diagnostics test that SolidWorks performs during translation detects faulty faces that may have resulted from the translation from other CAD/CAM systems or from the importing process. It also detects any gaps between faces. Both of these problems occur due to the numerical accuracy used by each system to represent the geometry. They may also occur due to the internal representation (equations) used by each CAD/CAM system to define the geometry. If SolidWorks does not find any problems, it converts the B-rep into features and displays the model and its features tree. The model is now good and in native SolidWorks format, and so it is ready to use in design and other activities.

13.3 Mass Properties

Mass properties calculations are the oldest application offered by CAD/CAM systems. The foundation behind this application comes from basic engineering courses (calculus and statistics). These calculations include volume, mass, center of mass, first and second moments of inertia, and principal moments and axes of inertia. The following equations provide these properties:

Volume:

$$V = \iiint_V dx\,dy\,dz \tag{13.1}$$

Mass:

$$M = \rho V \tag{13.2}$$

Center of mass (centroid):

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \frac{1}{V}\begin{bmatrix} \iiint_V x dV \\ \iiint_V y dV \\ \iiint_V z dV \end{bmatrix} \tag{13.3}$$

Second moments of inertia:

$$I_{xx} = \int_M (y^2 + z^2)dm \tag{13.4}$$

$$I_{yy} = \int_M (x^2 + z^2)dm \tag{13.5}$$

$$I_{zz} = \int_M (x^2 + y^2)dm \tag{13.6}$$

First moments of inertia:

$$I_{xy} = \int_M (xy)dm \tag{13.7}$$

$$I_{xz} = \int_M (xz)dm \tag{13.8}$$

$$I_{yz} = \int_M (yz)dm \tag{13.9}$$

Inertia tensor:

$$[I] = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \tag{13.10}$$

Principal inertia tensor:

$$[I] = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \tag{13.11}$$

The evaluation of the integrals shown in Eqs. (13.1)–(13.9) is done numerically because they are evaluated over the boundaries of a 3D CAD model. This makes the exact integration of these equations impossible. The most efficient and accurate numerical integration method is Gauss quadrature. (Matlab has Gauss quadrature built into it.) CAD/CAM systems use this method to calculate the mass properties. Gauss quadrature converts an integral equation into a summation that is easily calculated over the integral limits. The integral could be single or double. These integrals are functions of u or (u, v) parameters that we have covered in Chapters 8 and 9. These two types of integrals take the following general forms:

$$I = \int_a^b f(u)du \tag{13.12}$$

$$I = \int_c^d \int_a^b f(u, v)du\, dv \tag{13.13}$$

The single integral given by Eq. (13.12) is used to integrate functions over a curve, whereas the double integral given by Eq. (13.13) is used to integrate functions over a surface.

The calculations of mass properties proceed as follows. We know that CAD/CAM systems use parametric equations to represent and store the curves and surfaces that

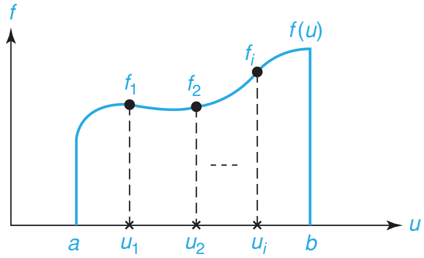


FIGURE 13.2
Gauss quadrature

make up the features of CAD models. Thus, we recast Eqs. (13.1) through (13.9) in terms of the parameters u and (u, v) and convert them to the forms given by Eqs. (13.12) and (13.13). This conversion is beyond the scope of this book. After we have Eqs. (13.12) and (13.13), we integrate them numerically using Gauss quadrature to calculate the mass properties.

Gauss quadrature is a simple, but accurate, numerical integration method. We know from calculus that the area under a bounded curve is the integral of the curve function. Figure 13.2 shows the graphical representation of Eq. (13.12). Gauss quadrature samples the function at selected sampling points (Figure 13.2 shows three sampling points), evaluates $f(u)$ at these points, and evaluates the integral. Using this procedure, Eq. (13.12) becomes:

$$I = \int_a^b f(u)du = \frac{b-a}{2}\sum_{i=1}^{n} W_i f_i = W_1 f_1 + W_2 f_2 + W_3 f_3 + \cdots + W_n f_n \quad (13.14)$$

where n is the number of sampling points, and W_i are Gauss weights that are assigned to the functions f_i. The functions f_i are calculated at the sampling points u_i given by:

$$u_i = \frac{b-a}{2}C_i + \frac{b+a}{2} \quad (13.15)$$

where C_i is the Gaussian location of the sampling points. Table 13.2 shows the values of C_i and W_i for different numbers of sampling points.

TABLE 13.2 Gauss Quadrature Data

Number of Sampling Points, n	Location, C_i	Weight, W_i
1	0.0000000000000000	2.000000000000000
2	+0.577350269189626	1.000000000000000
	−0.577350269189626	1.000000000000000
3	+0.774596669241483	0.555555555555556
	0.0000000000000000	0.888888888888889
	−0.774596669241483	0.555555555555556

How many sampling points should a CAD/CAM system use in mass property calculations? Before we answer this question, keep in mind that $f(u)$ is a polynomial in the CAD/CAM field because it comes from the parametric representation that uses only polynomials. Gauss quadrature produces exact results (equal to the results we could obtain from evaluating the integral in exact form, if that were possible) for integrating polynomials according to this rule: polynomials degree $\leq 2n - 1$, where n is the number of sampling points. Thus, three sampling points ($n = 3$) should produce exact results for a polynomial of degree of 5. This is more than adequate for CAD/CAM because the highest order polynomial we use is 3. In engineering and CAD/CAM applications, 3 sampling points is all we need.

The evaluation of the double integral of Eq. (13.3) using Gauss quadrature is the same as for the single integral and yields the following equation:

$$I = \int_c^d \int_a^b f(u,v)du\, dv = \frac{(b-a)}{2}\frac{(d-c)}{2}\sum_{i=1}^{n}\sum_{j=1}^{m} W_i W_j f_{ij} \quad (13.16)$$

The calculations of W_i, W_j, and f_{ij} are the same as for the single integral. Think of extending the calculations from 1D to 2D. f_{ij} is evaluated at the sampling point located at (u_i, v_j).

Example 13.1 Use Gauss quadrature to evaluate this integral:

$$I = \int_1^3 (u^4 - u)du \tag{13.17}$$

Solution The polynomial degree is 4. Using the rule of exact integration, we need $2n - 1 = 4$, or $n = 2.5$ sampling points for Gauss quadrature to yield the exact results. Thus, we use $n = 3$. Here are the steps to evaluate the integral:

1. Using the integral, we identify the following parameters: $a = 1$, $b = 3$, $f(u) = u^4 - u$
2. Using Eq. (13.14) for 3 sampling points, we write Eq. (13.17) as:

$$I = \frac{3 - 1}{2}(W_1 f_1 + W_2 f_2 + W_3 f_3) \tag{13.18}$$

3. Using Table 13.1 for $n = 3$, calculate f_i at the three sampling points; we designate them by their Gauss data (C_i, W_i). Let us designate f_1, f_2, and f_3, respectively, by (0.7745967, 0.5555556), (0.0, 0.8888889), and (–0.7745967, 0.5555556). Calculating these functions requires calculating the sampling coordinate value u_i at each point using Eq. (13.15). This gives $u_1 = 2.7745967$, $u_2 = 2$, and $u_3 = 1.2254033$. Thus,

$$f_1 = (2.7745967)^4 - 2.7745967 = 56.4905634$$

   Similarly, $f_2 = 14$, and $f_3 = 1.0294390$
4. Substituting into Eq. (13.18) and reducing give $I = 44.40000405$

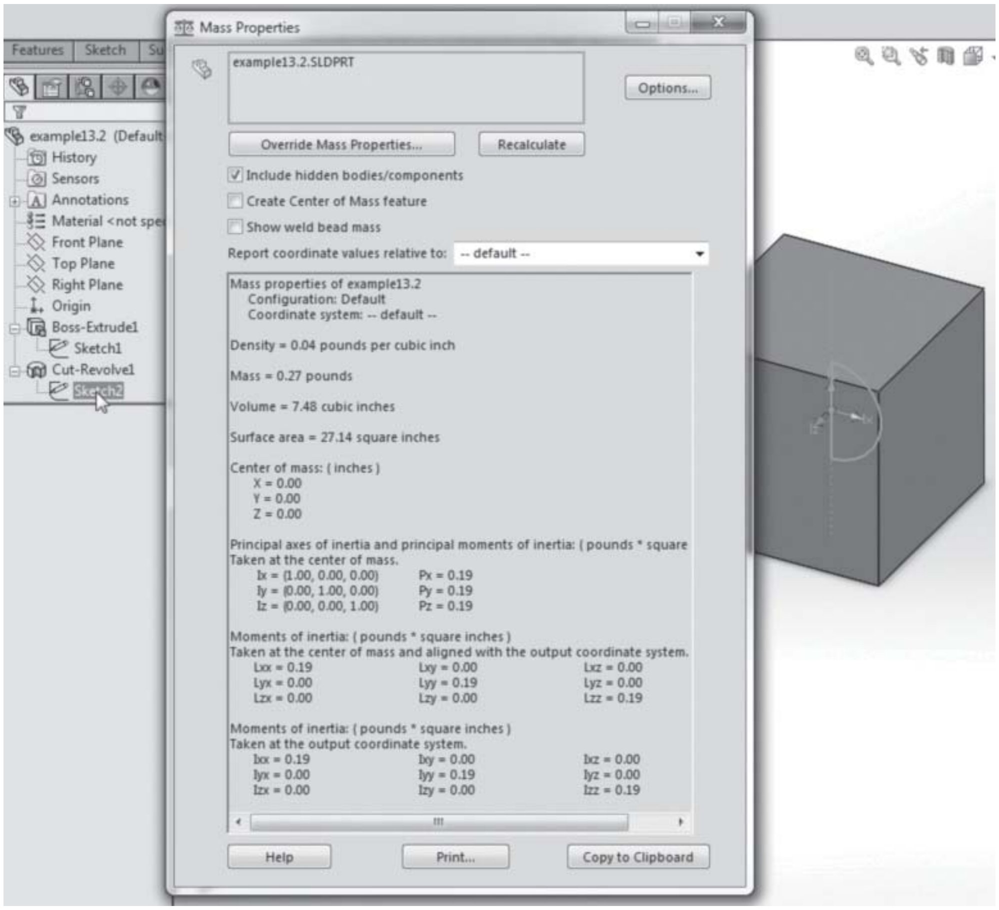
The exact integration of the integral gives 44.4. Gauss quadrature yields the exact value. The small error (0.00000405) is due to truncation errors. We used only 7 decimal places in this example. If we use the 15 decimal places shown in Table 13.2, we will get better results.

HANDS-ON FOR EXAMPLE 13.1. Use 1 and 2 sampling points. Compare the accuracy of the results with the 3 point result.

Example 13.2 Use SolidWorks to calculate the centroid and volume of a 2 × 2 × 2 in. box with a spherical void (hole) at its center. The sphere diameter is 1 in. Create the box so that its MCS is at its center.

Solution This part is somewhat odd with a void inside it. But it illustrates the accuracy of the mass properties calculations of SolidWorks. The box sketch is a square at the front plane with a center at the origin. Extrude the sketch using the mid plane direction, so we can use the front plane to sketch a circle for the sphere (**Revolved Cut**) and centerline (axis of revolution). Trim the circle, as shown in Figure 13.3, before revolving it around the centerline; otherwise, SolidWorks produces an error (self-intersecting) and the revolve cut fails.

FIGURE 13.3
SolidWorks mass properties



Run SolidWorks mass properties by clicking **Mass Properties** icon on the **Evaluate** tab. The results in Figure 13.3 show that the center of mass is located at (0, 0, 0), as we would expect, because the model is symmetric with respect to the origin. The volume is listed as 7.48 in^3. That is easily verifiable. The box volume is given by $2^3 = 8$ in^3. The sphere volume is $(4/3)\ \pi R^3 = (4/3)\ \pi 0.5^3 = 0.52$ in^3. Thus, the net volume $= 8 - 0.52 =$ 7.48 in^3. That is identical to the SolidWorks result.

HANDS-ON FOR EXAMPLE 13.2. Replace the sphere by an ellipsoid and calculate the mass properties. Compare with the exact results. *Note:* Ellipsoid volume is (4/3) πabc, where a, b, and c are the ellipsoid radii. When you sketch the ellipse to revolve cut, use $a = 0.75$ in. and $b = 0.5$ in. Depending on the axis of revolution you use, c could be 0.5 or 0.75. If you revolve around the longer axis ($a = 0.75$), $c = 0.5$.

Note: We have not assigned a material to the model. SolidWorks assumes a default density.

13.4 Animation and Motion Analysis

SolidWorks provides animation and motion analysis of an assembly. Motion studies can be time based or event based. Time-based motion studies describe the response to time-based changes in motion elements on the assembly motion. Event-based motion studies are defined with a set of motion actions resulting from triggering events. We can obtain the time sequence for element changes by calculating an event-based motion study.

We use event-based motion analysis to specify motion from some combination of sensors, times, or previous events. Event-based motion requires a set of tasks. The tasks can be sequential or can overlap in time. Each task is defined by a triggering event and its associated task action that controls or defines motion during the task.

13.5 Flow Simulation

With SolidWorks **Flow Simulation**, you are able to analyze the flow of up to 10 fluids of different types (liquids, gases/steam, real gases, non-Newtonian liquids, and compressible liquids). SolidWorks contains numerous fluids with predefined properties. In addition to preexisting fluids, SolidWorks allows you to define your own fluids. The flow analysis includes mutual dissolution of fluids. Mixing fluids must be of the same type. A simple example of flow simulation is to study the flow of water inside a hose and animate the streamlines to visualize the water movement.

SolidWorks requires a closed control volume to perform its flow simulation. The flow model it uses requires an inlet and outlet for the flow to enter and exit the control volume. The inlet and outlet need to be closed to run the simulation. The flow parameters include temperature, pressure, flow velocity, and flow rate. We need to set the fluid initial settings (boundary conditions, B.C.) at the inlet and outlet before we start the simulation. The B.C. must ensure the creation of a gradient or drop to force a fluid flow (e.g., inlet pressure must be greater than outlet pressure to create a pressure drop to move the flow in the control volume). Flow parameters depend on each other. For example, if we specify a pressure drop, the simulation calculates the flow velocity and rate. If we specify a flow rate or velocity, pressure drop is calculated. Tutorial 13.7 provides more details.

13.6 Finite Element Method

The **finite element method** is a numerical method that is capable of solving almost any problem with any level of complexity. The method consists of finite element modeling (FEM) and analysis (FEA), or FEM/FEA. Many engineering design problems are too complex to find closed-form solutions for their governing equilibrium equations. The elegance of the finite element method is that it converts the equilibrium (differential) equation of a continuum into an integral equation. Then, instead of solving the equilibrium equation over the entire domain of the continuum, it applies the integral equation over many small regions called *elements* and, finally, assembles the elements' equations into a set of simultaneous algebraic equations whose solutions generate the domain solution.

Thus, the method divides a complex shape (domain) into smaller elements. Each element has designated points called *nodes*. The elements are adjacent to each other, with no gaps between their sides, and are connected at the nodes. Each node has a set of degrees of freedom (DOF) that depend on the problem at hand. The selection of element type, the number of nodes per element, and the DOF at each node are among the important decisions to make to create and run an accurate FEA for a problem.

The finite element method begins with modeling (FEM) and ends with analysis (FEA). The method is complex to use and requires good understanding of the problem we need to solve. CAD/CAM systems automate the method to the point that many designers use it as a black box. This is both good and bad. It is good to speed up the

FEM/FEA so the designer can get answers quickly to the "what if" design scenarios and questions. It is bad because it requires that the designer have good understanding of using FEM/FEA to be able to model the problem correctly and interpret/analyze/verify the analysis results.

FEM sets up the problem for FEA. FEM has the following steps:

1. Generate the mesh: This includes breaking down the problem domain (the geometric model) into nodes and elements (mesh). The general rule is that the more elements and the more nodes per element we use (i.e., finer mesh), the better the accuracy of the FEA, but the more expensive it is to perform because it requires more computational time. Keep in mind that once we create the mesh, the geometric model is only defined by the nodes of the mesh. No other points that are not nodes are recognized by the FEA. Thus, we need to make sure that points of importance are made nodes; for example, points where external load are applied. While the CAD/CAM FEM/FEA module takes care of all these details, we need to be aware of them.
2. Select the element type and the number of nodes per element: This selection depends on the type of analysis at hand and the desired accuracy of the solution. For example, we can use a quadrilateral element with a minimum of four (corner) nodes and a maximum of eight nodes (four corner nodes and four mid-side nodes) and any number in between; or we can use a triangular element with a minimum of three (corner) nodes and a maximum of six nodes (three corner and three mid-side nodes).
3. Assign the DOF at the nodes: Depending on the element type and the problem to solve, assign the DOF at the nodes. For example, there is a maximum of 6 DOF per node (three translations and three rotations) for 3D problems. For a 3D stress/strain problem, we use three DOF: displacements along the three axes (X, Y, and Z). For a beam bending problem, we use two DOF: lateral displacement and rotation/slope.
4. Assign boundary conditions (B.C.): We use these conditions to reflect how the model (part) is fixed in real life. We need boundary conditions; otherwise, the model will float in space and the FEA will give wrong results (usually very large values). Example boundary conditions include fixing an end face of a cantilever beam, an edge of a face, or a vertex. Boundary conditions apply to nodes by suppressing all their DOF.
5. Apply material: We need to assign a material to the model. Some materials are linear, such as steel and aluminum, while others are nonlinear, such as rubber and plastics. Linear materials may use linear (for small deformation problems) or nonlinear (for large deformation problems) FEA. Nonlinear materials require nonlinear FEA.
6. Apply external loads: These loads cause the model to deform and create the FEA results. Loads could be forces at nodes, pressures at faces, moments, heat flux, or others. When we apply a load, we need to specify its type (force, moment), direction (along X-, Y-, or Z-axis), sense (up or down), and value (amount).

The FEM/FEA modules of CAD/CAM systems automate the preceding six steps to a great extent. The designer does not see Steps 1 through 3. The CAD/CAM system generates the mesh automatically. The most commonly used type of element is tetrahedral element with 10 nodes (4 corner and 6 mid-side). The system would require the designer to supply data for Steps 4 through 6. Although all CAD/CAM systems automate the mesh generation, setting up the finite element model is very crucial to FEA in order to get both the desired and accurate results. Ideally, the designer must have a good theoretical background of FEM/FEA to be efficient in setting up the model and selecting

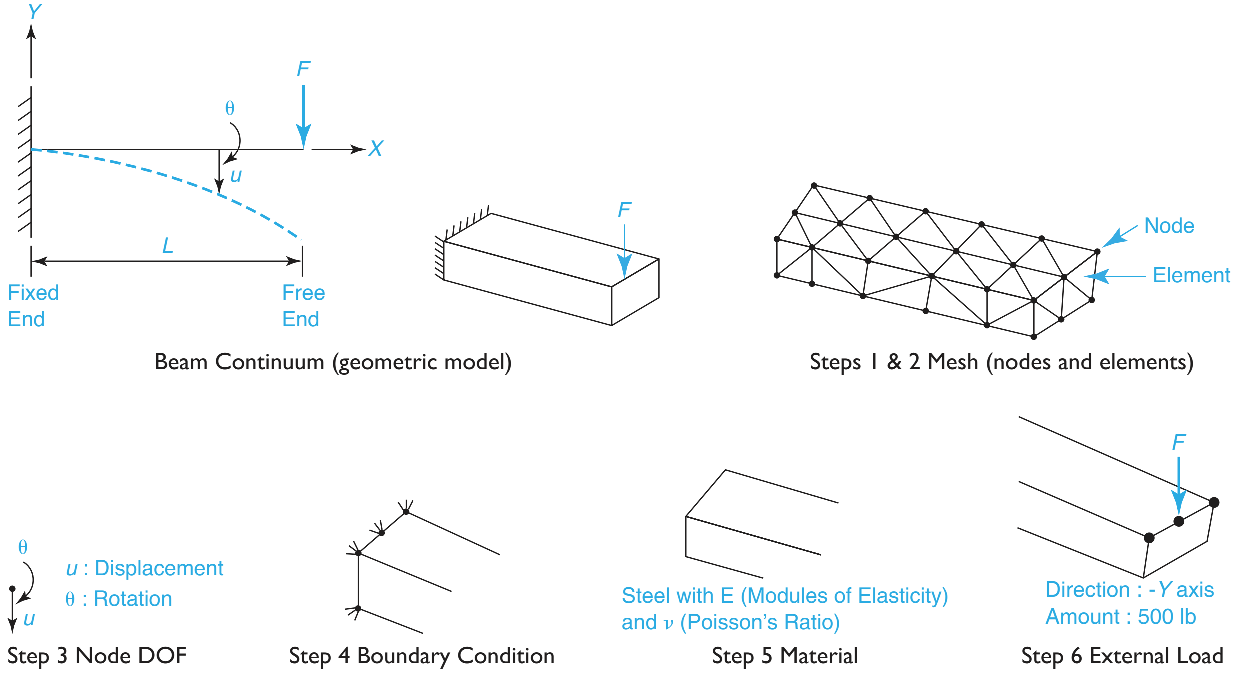


FIGURE 13.4
FEM Steps

the type of analysis (linear, nonlinear, static, dynamic, etc.). Figure 13.4 shows an overview of the preceding six steps as applied to a cantilever beam. Figure 13.5 shows the different types of elements. 2D elements (Figure 13.5B) include triangular (3 and 6 nodes) and quadrilateral (4 and 8 nodes) elements. 3D elements (Figure 13.5C) include tetrahedral (4 and 10 nodes) and hexahedral (8 and 20 nodes) elements.

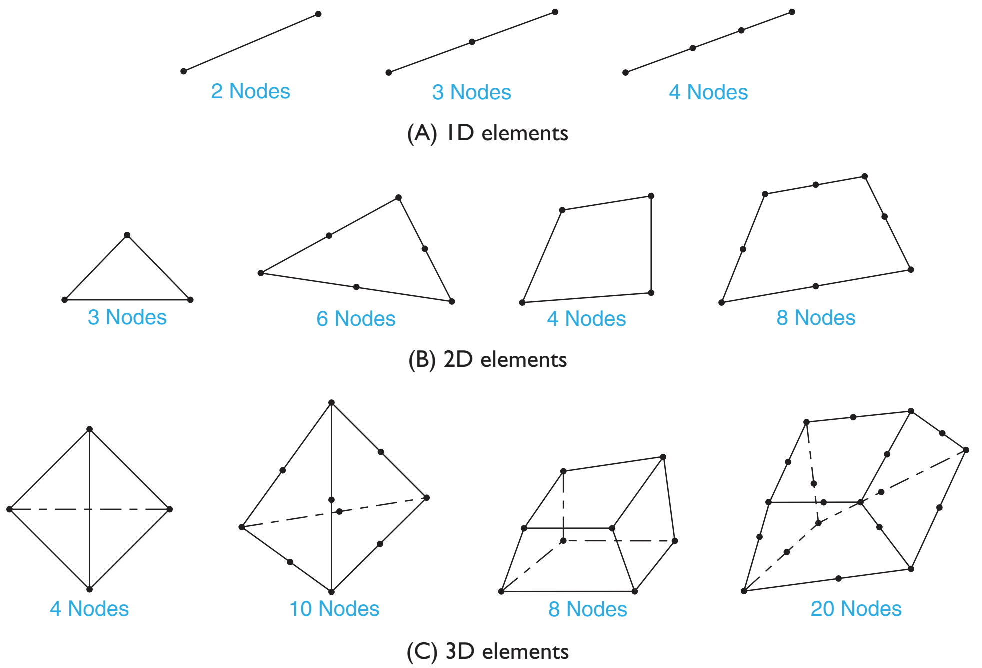


FIGURE 13.5
Sample finite elements

After we complete these six steps of FEM, we are ready to perform FEA. The FEA includes running the analysis to solve the finite element equations, display the results as curves and/or contours, and analyze the results. Analyzing and making sense of the results requires good understanding of the problem we are solving. After analyzing the results, the designer has two options: accept the design as is, or iterate it (change the part design, regenerate FEM, redo FEA) until satisfactory results are achieved.

The most commonly used and simplest type of FEA is static linear analysis. FEA could be static or dynamic. Within each type, the analysis could be linear or nonlinear. Nonlinearity could be due to large deformation or nonlinear material (such as rubber or plastic). In addition, we can perform FEM/FEA at the part or assembly level. The difference comes in the modeling of the mating between the assembly parts. Parts are not rigidly connected in an assembly. Thus, we usually need clever ways to model these mating regions. Experience with modeling of mechanical systems would help greatly. Typically, we may add springs, dampers, or other mechanical elements that closely describe the behavior of the mating region.

13.7 Finite Element Analysis

Although it is out of the scope of this book to provide full coverage of FEA, we offer a quick overview of it to help us better understand the FEM steps and activities. Let us consider the cantilever beam shown in Figure 13.4. The beam bending equation (also known as the *Euler-Bernoulli beam equation*) is:

$$EI\frac{d^4u}{dx^4} = 0 \tag{13.19}$$

where E is the modulus of elasticity of the beam material, and I is the moment of inertia of the beam cross section. The B.C. are that the beam has a fixed end (does not move). The external load is the point load (force) F at its free end.

Equation (13.19) is the equilibrium equation of the beam deflection under the force F. It is a differential equation. After processing Eq. (13.19) through the theoretical formulation of the finite element method, it becomes the following integral equation:

$$I = \frac{1}{2}\int_0^L EI\left(\frac{d^2u}{dx^2}\right)^2 dx - Fu_L \tag{13.20}$$

where u_L is the beam deflection at its free end where F is applied. This integral equation applies to the entire beam, that is, the continuum.

Equation (13.20) is difficult to solve (in closed form) because it still applies to the entire beam. We need to break the beam into nodes and elements, apply Eq. (13.20) to each element domain, solve it there, and then assemble the solution over the entire beam. This process results in converting the integral Eq. (13.20) into the following matrix equation:

$$[K]\boldsymbol{\delta} = \boldsymbol{P} \tag{13.21}$$

where $[K]$, $\boldsymbol{\delta}$, and $\boldsymbol{P}$ are, respectively, the beam stiffness matrix, the beam displacements or deflections (DOF) at the nodes, and the externally applied loads at the beam nodes. Both $\boldsymbol{\delta}$ and $\boldsymbol{P}$ are vectors. The stiffness matrix $[K]$ has in it the material effect, that is, the modulus of elasticity, E. It also has the beam geometry and cross section (I) effects in it. It finally reflects the B.C. of the beam. That is, it does not include the fixed nodes at the fixed end because they do not move; they are eliminated from the $\boldsymbol{\delta}$ vector.

FEA solves Eq. (13.21). This is a matrix equation. It consists of a set of algebraic equations that must be solved simultaneously (using the Gauss elimination method) for the $\boldsymbol{\delta}$ vector. CAD/CAM systems use this vector to display the deflected shape of the beam, the stress contours, and the animation of the beam deflection.

Other types of FEA follow the same formulation. Equation (13.21) is for static linear analysis. It is linear because $[K]$ is not a function of $\boldsymbol{\delta}$ and the material is linear (such as steel or aluminum). Otherwise, we have nonlinear static analysis. The sources of nonlinearity could be large displacement or nonlinear material. Dynamic analysis occurs when the problem changes with time. If the beam force, F, changes with time or the beam vibrates, we have dynamic FEA. Equation (13.21) takes the following general form for dynamic FEA:

$$[M]\ddot{\boldsymbol{\delta}} + [C]\dot{\boldsymbol{\delta}} + [K]\boldsymbol{\delta} = \boldsymbol{P} \tag{13.22}$$

where $[M]$, $[C]$, $\ddot{\boldsymbol{\delta}}$, and $\dot{\boldsymbol{\delta}}$ are, respectively, the mass matrix, the damping matrix, the acceleration vector, and the velocity vector. If $[C]$ and/or $[K]$ depend on $\boldsymbol{\delta}$ or its derivatives, we have nonlinear dynamic analysis; otherwise, we have linear dynamic analysis.

Once we solve Eq. (13.21) or (13.22) for the displacement, we can calculate the strain using the following equation:

$$\varepsilon = \frac{\Delta}{L} \tag{13.23}$$

where L is the element length that went through displacement Δ. Using the material relationship, the stress is:

$$\boldsymbol{\sigma} = E\varepsilon \tag{13.24}$$

Let us relate FEM Steps 1 through 6 to this FEA to make sense of these steps so we can use them effectively in SolidWorks **Simulation** (FEM/FEA). Steps 1 through 5 create the matrices $[M]$, $[C]$, and $[K]$. Step 6 creates the $\boldsymbol{P}$ vector. This enables FEA to solve Eq. (13.21) or (13.22) and generate the results.

13.8 SolidWorks Simulation

SolidWorks implements the FEM/FEA concepts discussed in this chapter. It offers two FEM/FEA modules: **SimulationXpress** and **Simulation**. The former comes with standard SolidWorks licensing whereas the latter is an Add-In. SolidWorks **SimulationXpress** is a built-in first-pass design analysis and verification tool for testing part designs quickly and easily. It allows us to test a design and investigate different design alternatives without having to manufacture and test the design. SolidWorks **SimulationXpress** is suitable for linear stress analysis of simple parts. It generates HTML reports and creates SolidWorks eDrawings files to document and communicate the results of the analysis.

SolidWorks **SimulationXpress** provides only part analysis (geometry type); stress and displacement (analysis type); uniform pressure and forces (load type); stress, displacement, and factor of safety plots (visualization); and HTML report, eDrawings, and AVI animation files (reports). If we need additional functionality or more in-depth FEM/FEA, we use SolidWorks **Simulation**. SolidWorks **Simulation** offers all that **SimulationXpress** does and more. It provides assembly (including contact and friction) analysis, more external load types (nonuniform pressure, torques, heat loading), assembly connectors (springs, elastic foundation, pins, bolts, spot welds), more visualization (ISO plots), and more reports (customization and image [bitmap, JPEG, or VRML]).

SolidWorks reports the stresses as Von Mises stress. Simply stated, Von Mises stress is based on the fact that the material starts to yield when the maximum stress, $\boldsymbol{\sigma}_{\max}$, due to externally applied load reaches or exceeds its yield stress, $\boldsymbol{\sigma}_y$; that is, when $\boldsymbol{\sigma}_{\max} \geq \boldsymbol{\sigma}_y$.

If $\boldsymbol{\sigma}_{max} \geq \boldsymbol{\sigma}_y$, the material fails (yields) and the part fails, causing the assembly (product) to fail and triggering failure consequences. Materials should always operate within their elastic zone. Thus, as a designer, your FEM/FEA should ensure that the design is safe within a factor of safety (*FS*), defined as:

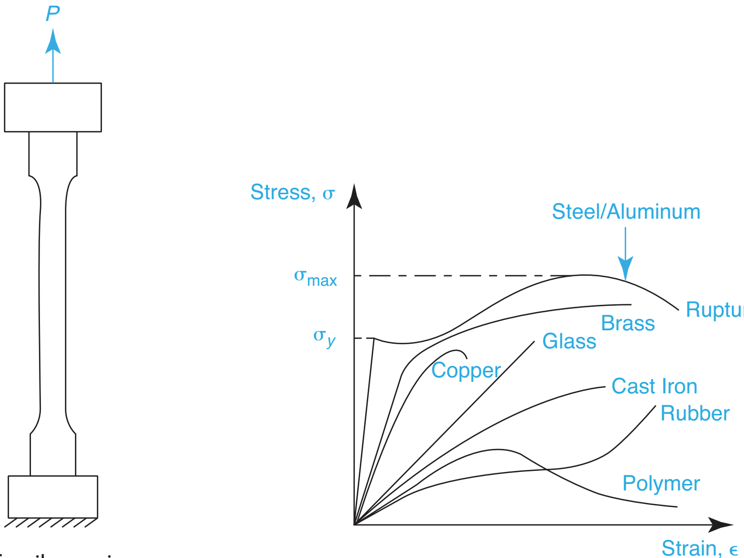
$$FS = \frac{\sigma_y}{\sigma_{max}} \geq 1 \qquad (13.25)$$

The factor of safety, *FS*, should always be greater than 1, for example, 1.5 or 2. Depending on how catastrophic the failure consequences of a part are, the *FS* should be determined. The worse the consequences of failure, the higher the *FS* should be. However, there is a silver lining here. Although higher *FS* seems logical, it makes the design more expensive and unnecessarily heavy because we use more material to increase the part strength. Thus, experience, heuristics, and field testing play an important role in determining a reasonable *FS* for a part design.

13.9 Von Mises Stress

We need to understand the stress-strain curve of a material to understand Von Mises stress. Von Mises stress is a criterion for ductile material failure. Ductile material is a material that has yield stress, $\boldsymbol{\sigma}_y$, as shown in Figure 13.6. Ductile materials are also known as isotropic materials that obey Hook's law. Table 13.3 shows the strength values

FIGURE 13.6
Stress-strain curves



(A) Tensile specimen with load *P*

(B) Curves for sample engineering materials

TABLE 13.3 Material Properties for Some Materials

Material	Modulus of Elasticity, E (GPa)	Poisson's Ratio, ν	Yield Stress, σ_y (MPa)	Max Stress, σ_{max} (MPa)
Tungsten	407	0.28	760	960
Steel (1020)	207	0.30	180	380
Titanium	107	0.34	450	520
Brass (70–30)	97	0.34	75	300
Aluminum	69	0.33	35	90

for some materials. The units shown in the table are gigapascal (GPa) and megapascal (MPa). Giga is 1 billion (10^9) and mega is 1 million (10^6), and a Pascal is 1 Newton per square meter (N/m^2). Also, 1 N/m^2 is equivalent to 0.0001450377 psi.

Steel and aluminum are sample ductile materials. The stress-strain curve is linear up to the yield stress, $\boldsymbol{\sigma}_y$. The slope of the line is the material modulus of elasticity, E. The material is elastic in this linear zone, meaning it fully recovers when the externally applied load is removed. After the yield point, the material becomes plastic, meaning that it experiences permanent deformation. Such deformation renders the part dysfunctional in its assembly; that is, the part fails its functional requirements and must be replaced.

The stress-strain curves shown in Figure 13.6 are usually derived from what is known in the materials field as a *uniaxial stress test*, also known as a *simple tensile test*. Scientists use a tensile test machine, design and produce a tensile specimen (rod with circular cross section and square ends, as shown in Figure 13.6A), mount the specimen in the machine, and apply a progressive load, $\boldsymbol{P}$, to the specimen until it necks and breaks. While the load value is increasing, the specimen is elongating under the load. And while it is elongating, it is necking at its midpoint along its length. Scientists mount strain gages on the specimen (at the necking region) to measure its displacements. These measurements are used to plot the stress-strain curves shown in Figure 13.6B and calculate the yield stress, $\boldsymbol{\sigma}_y$. There are all sorts of standards and requirements the scientist must follow to create the specimen (with the required dimensions and shape) and conduct the test.

Although the simple tensile test conducted in materials laboratories generates the material properties, it seldom represents loading of the material in real life. Such loading is almost always multiaxial. Multiaxial loading introduces two safety concerns for designers. First, multiaxial loading is more severe than uniaxial loading and would cause the material to fail sooner. Second, the material is loaded along all axes of loading. In such a case, which stress, along which axis, should the designer use to determine the design factor of safety?

The remedy for these two concerns is the Von Mises stress, $\boldsymbol{\sigma}_v$. **Von Mises stress** is defined as an equivalent tensile stress that is used to predict yielding of materials under multiaxial loading conditions using results from the simple uniaxial stress test. In other words, the Von Mises criterion is a formula for calculating whether the stress combination at a given point will cause failure. We have at most three directions or axes (X, Y, and Z) along which we can apply loads. (These axes are also considered to be the principal axes of loading.) A simple example of multiaxial loading occurs when we apply a force that is not along one of the axes. For example, applying a force F in the XY plane produces two force components: F_x (along the X-axis) and F_y (along the Y-axis), thus producing multiaxial loading.

Let us assume that loading a part produces three stresses (at a point) of $\boldsymbol{\sigma}_1$, $\boldsymbol{\sigma}_2$, and $\boldsymbol{\sigma}_3$ along the three axes X, Y, and Z, respectively. Von Mises stress at this point is given by:

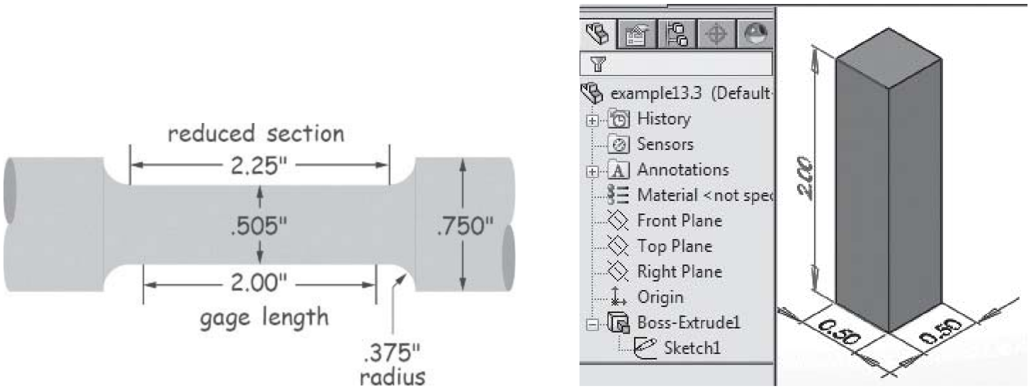
$$\boldsymbol{\sigma}_v = \sqrt{\frac{(\boldsymbol{\sigma}_1 - \boldsymbol{\sigma}_2)^2 + (\boldsymbol{\sigma}_2 - \boldsymbol{\sigma}_3)^2 + (\boldsymbol{\sigma}_1 - \boldsymbol{\sigma}_3)^2}{2}} \tag{13.26}$$

In this case, yielding occurs when the equivalent stress, $\boldsymbol{\sigma}_v$, reaches the yield stress of the material in the simple tension test, $\boldsymbol{\sigma}_y$. In the case of uniaxial stress, $\boldsymbol{\sigma}_1 \neq 0, \boldsymbol{\sigma}_2 = \boldsymbol{\sigma}_3 = 0$, and Eq. (13.26) reduces to $\boldsymbol{\sigma}_v = \boldsymbol{\sigma}_1$. If $\boldsymbol{\sigma}_1$ reaches $\boldsymbol{\sigma}_y$, the material fails.

SolidWorks **Simulation** calculates $\boldsymbol{\sigma}_v$ via FEA as follows. It calculates the displacements at the nodes (points) by solving Eq. (13.21) for static loading. These displacements are due to multiaxial loading that is reflected in the load vector $\boldsymbol{P}$. It then uses Eqs. (13.23) and (13.24) to calculate the strains and stresses at the nodes, respectively. These equations provide three strains (ε_1, ε_2, ε_3) and three stresses (σ_1, σ_2, σ_3) at each node. Finally, SolidWorks substitutes these stresses into Eq. (13.26) to calculate the Von Mises stress, σ_v, at the center point of each element. It also uses Eq. (3.25) to calculate the factor of safety, *FS*. It displays the stress contour and *FS* for the designer to evaluate.

Example 13.3 **Figure 13.7 shows a steel column. The column has a square cross section that is 0.5 × 0.5 in. The column has a height of 2 in. It is subjected to a tensile force of 100 lb at its top face and is fixed at its bottom face. Use FEM/FEA to calculate the stresses in the column. Verify the FEA results against the exact solution.**

Solution The idea behind this example is to verify the accuracy of the finite element method against the exact solution of a simple problem. The example simplifies the geometry of a tensile specimen, creates a geometric model (part) of the simplified geometry, runs the FEM/FEA, and verifies the results. We use SolidWorks **SimulationXpress** instead of **Simulation**.



(A) Specimen standard dimensions (B) Geometric model of specimen

FIGURE 13.7
Finite element modeling of a tensile test specimen model

A standard tensile specimen has standard dimensions. Figure 13.7A shows a sample. Our geometric model of the specimen shown in Figure 13.7B uses the dimensions of the middle part of the specimen with simplifications. It uses a square cross section instead of a circular one. It also uses 0.5 in. instead of 0.505 in. for simplicity. We apply a force of value 100 lb at the center of the top face of the geometric model while we fix its bottom face.

The general steps to perform FEM/FEA using **SimulationXpress** are:

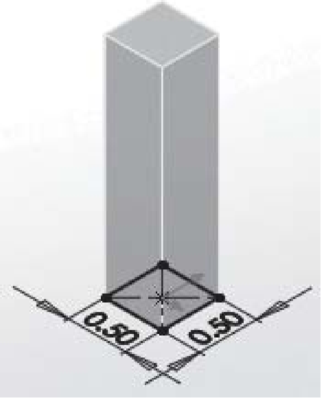
1. Create the column model
2. Start **SimulationXpress** and set units
3. Apply B.C
4. Apply external loads
5. Apply material
6. Run FEA
7. Display the results
8. Generate reports

Step 7 below shows that the minimum and maximum Von Mises stresses in this part are approximately 215 and 649 psi, respectively. If we interpolate to the midpoint of the column (at $L = 1.0$ in.) by taking the average of the two values, we get a stress value of $\sigma = 432$ psi. The closed-form solution to this problem is:

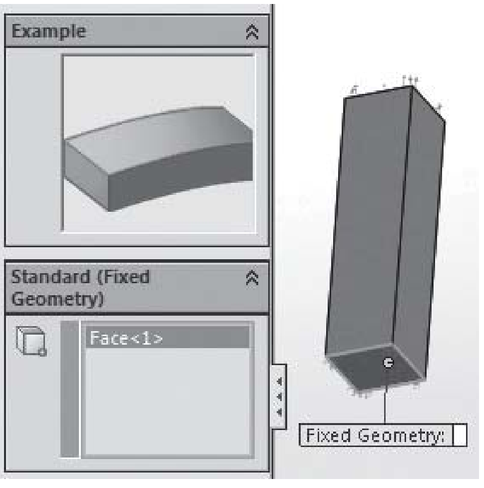
$$\sigma = \frac{P}{A} = \frac{100}{0.5 \times 0.5} = 400 \text{ psi} \tag{13.27}$$

Thus, we have an error of 8% (32/400).

Step 1: Create column model: **File** > **New** > **Part** > **OK** > create sketch shown on **Top Plane** and extrude 2.00 in up > **File** > **Save As** > *example13.3* > **Save.**



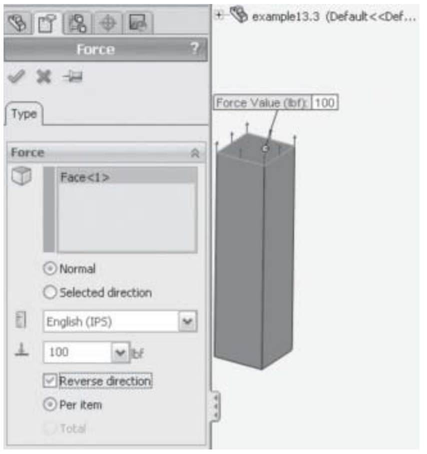
Step 2: Start **SimulationXpress** and set options: **Tools** > **SimulationXpress** > **Options** > select **English units** from dropdown > check off checkbox shown > **OK.**



Step 3: Apply B.C.: **Next** > **Add a fixture** > bottom face shown > ✔.

Note: Fixing the bottom face mimics the tensile test.

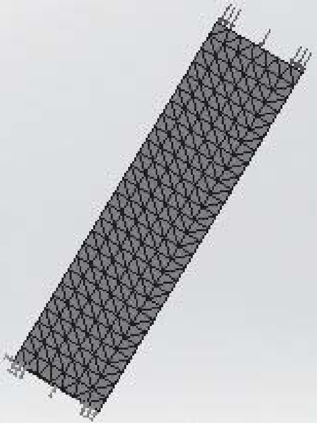
Step 4: Apply external loads: **Next** > **Add a force** > top face shown > Type 100 for force > reverse direction > ✔.



Note: SimulationXpress does not allow applying a point force; rather, only on a face.

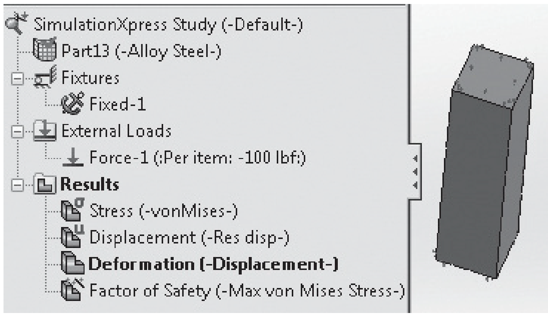
Note: We turn on **Reverse direction** because we want the force to be tensile, not compression.

Step 5: Apply material: **Next** > **Choose Material** > **Alloy Steel** > **Units as English (IPS)** > **Apply** > **Close.**



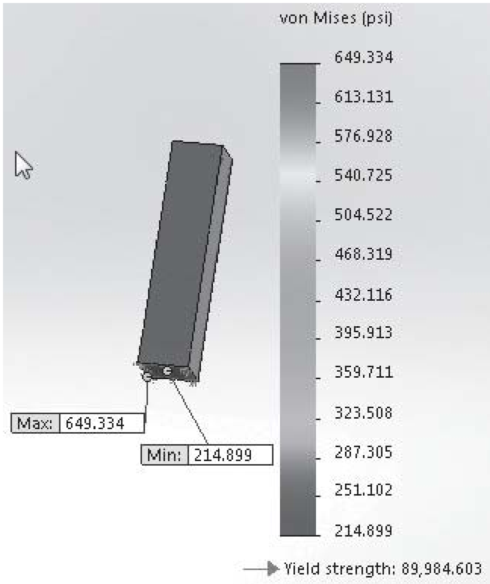
Note: SimulationXpress shows the mesh (right-click part node shown below in Step 6 > **Show Mesh**), but does not allow you to control it. If you need such control and more, you must use SolidWorks **Simulation.**

Step 6: Run FEA: **Next > Run Simulation > Stop animation** (or **Play animation**)



Step 7: Display results: **Yes, Continue** > double-click any result from the list shown above in Step 6 to display and view.

Step 8: Generate reports: **Done viewing results** > **Generate HTML report** (or **Generate eDrawings file**).



Note: If you select the HTML report, SolidWorks displays a Web page in the browser. This is your report. The report is also stored in the folder specified in Step 1.

HANDS-ON FOR EXAMPLE 13.3. Change the cross section of the column to a circular shape with a diameter = 0.5 in. Also apply a uniform pressure that is equivalent to 100 lb force to the top face. Calculate the exact stress by hand and compare with the FEM/FEA results. What is the error?

13.10 Tutorials

The theme for the tutorials in this chapter is to practice the different analysis tools covered in this chapter including exporting CAD models, performing mass property calculations, motion analysis, flow analysis, stress FEM/FEA, and thermal FEM/FEA.

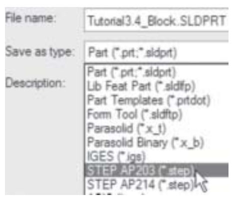
Tutorial 13–1: Export Native SolidWorks Files

We export a native SolidWorks file in both IGES and STEP formats.

Step 1: Open *Tutorial 3.4 _Block* part: **File** > **Open** > locate part file > **Open.**

Step 2: Save part as IGES file: **File > Save as** > *block* > **IGES** from the **Save as type** drop-down > **Save.**

Step 3: Save part as STEP file: **File > Save as** > *block* > **STEP AP203** from the **Save as type** drop-down > **Save.**



HANDS-ON FOR TUTORIAL 13–1. Investigate the options of saving the IGES and STEP files.

Tutorial 13–2: Import IGES and STEP Files into SolidWorks

We import the IGES file created in Tutorial 13–1 into SolidWorks, and use the **Compare Geometry** to verify the imported model. We run a diagnostic on the file to verify that no changes occurred due to the conversion.

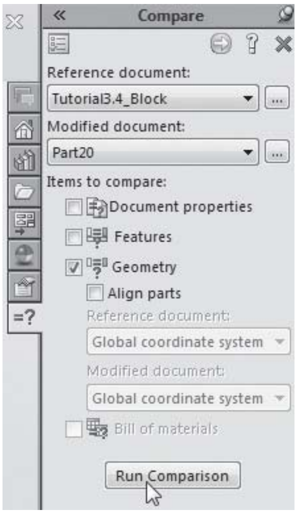
Step 1: Open IGES file: **File** > **Open** > locate *block.igs* part file > **Open.**

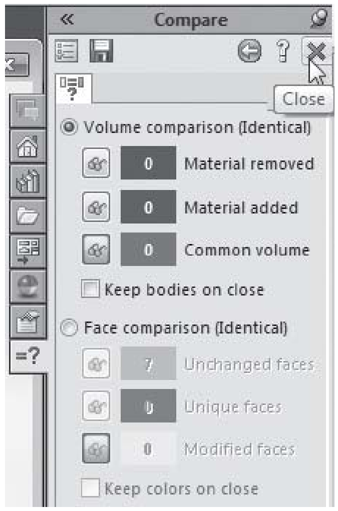
Step 2: Run import diagnostics: **Yes** to run **Import Diagnostics** shown above > ✔.

Step 3: Run feature recognition: **Yes** to feature recognition shown above > **Next** (arrow on top right corner of left pane) > ✔.

Note: If you say **No**, a dumb feature (cannot be edited) is created with one node in the features tree named *Imported1*.

Step 4: Compare geometry: **Tools > Compare > Geometry** > IGES part as **Reference Document** > part resulting from feature recognition as **Modified Document** > **Run Comparison** button shown > **Compare** window opens up with results as shown > close **Compare** window when done.





Note: Click a checkbox above to view results.

Note: Screen splits to show parts. Click window **Maximize** symbol to return to full screen mode.

Note: When you access **Compare > Geometry** tool, SolidWorks automatically checks off **SolidWorks Utilities** add-in. Verify using this sequence: **Tools > Add-Ins.**

HANDS-ON FOR TUTORIAL 13–2. Run the same import and compare geometry steps on the STEP AP203 part created in Step 3 of Tutorial 13–1.

Tutorial 13–3: Calculate Mass Properties of a Solid

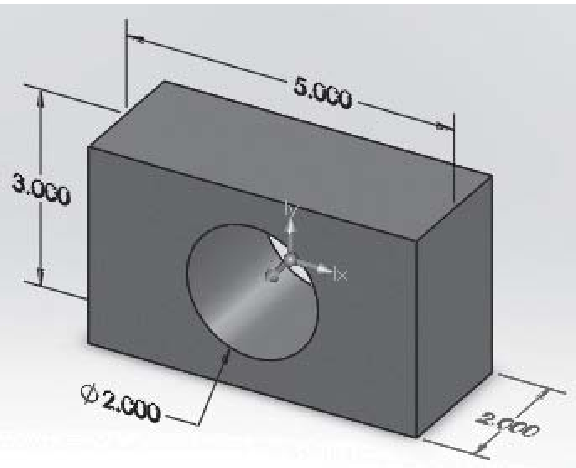
We calculate the mass properties of a part. SolidWorks calculates the mass, the center of mass, and moments of inertia.

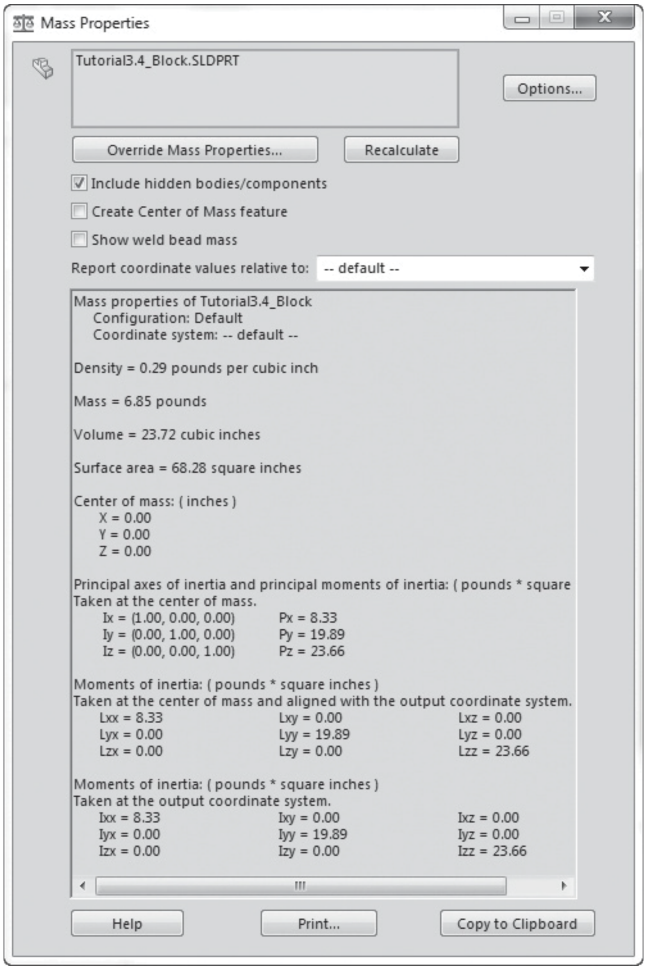
Step 1: Open part file: **File** > **Open** > locate *tutorial3.4_Block* part file > **Open.**

Step 2: Assign material properties to block: Right-click **Material <not specified>** node in features tree > **Edit Material** > expand **SolidWorks Materials** node > expand **Steel** node > **AISI 304** > **Apply** > **Close.**

Step 3: Calculate mass properties: **Evaluate** tab > **Mass Properties** > view results shown to right and center of mass shown below.

Note: Block is extruded using **Mid Plane.**





> ***HANDS-ON FOR TUTORIAL 13–3.*** (A) Calculate the volume, mass, and surface area of the block manually and compare with SolidWorks results. (B) Edit the block and use **Blind** extrusion instead of **Mid Plane**. Use SolidWorks to calculate the new mass properties. Compare the results with the results shown here. Do the new coordinates of the center of mass make sense?

Tutorial 13–4: Perform Motion Analysis Using a Motor

We create a motion analysis of the assembly of a two-axle steel wheel drum that is mounted on two blocks at its ends as shown in Figures 13.8 and 13.9. We add a motor to drive the wheel drum. We study the motor functionality using SolidWorks **Motion**

FIGURE 13.8
Wheel Drum Assembly (all dimensions are in inches)

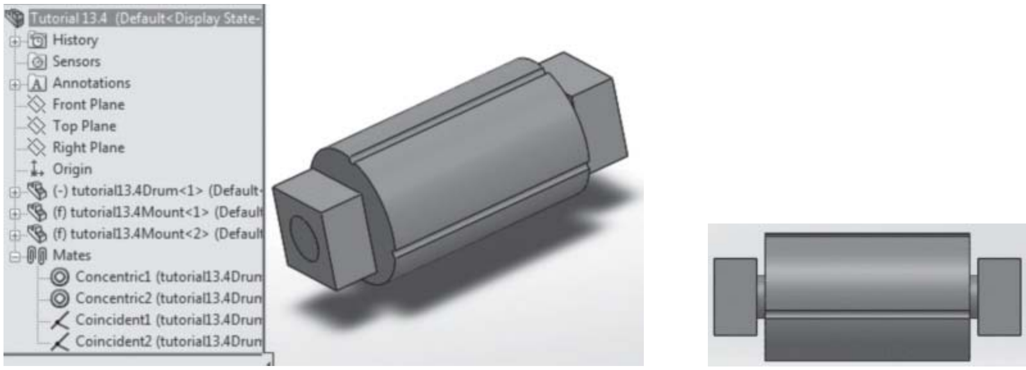
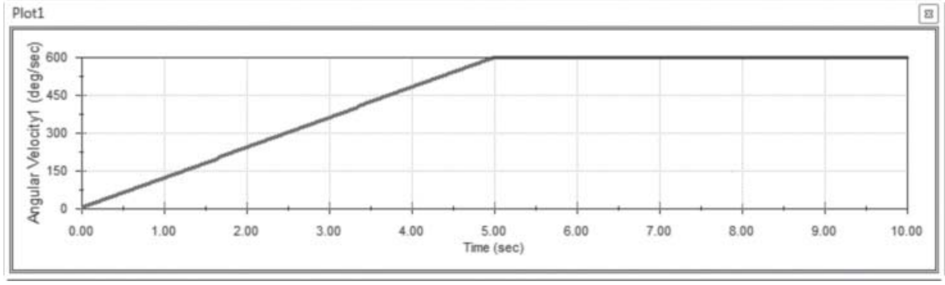
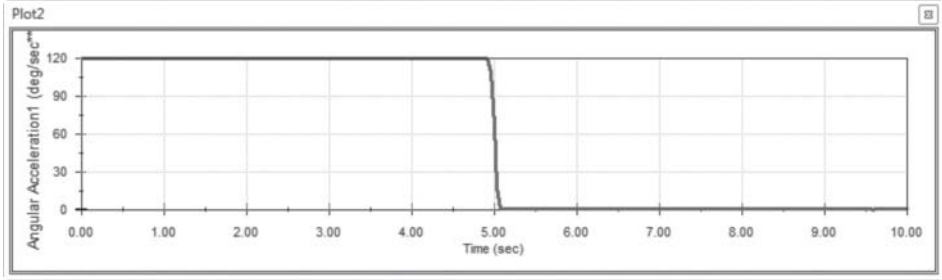


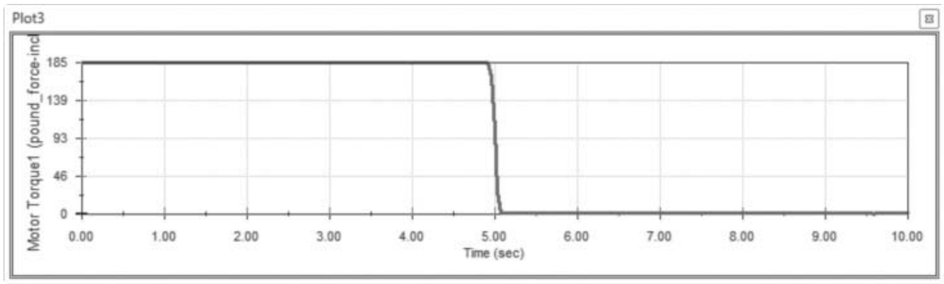
FIGURE 13.9
Motion Analysis Graphs and Plots



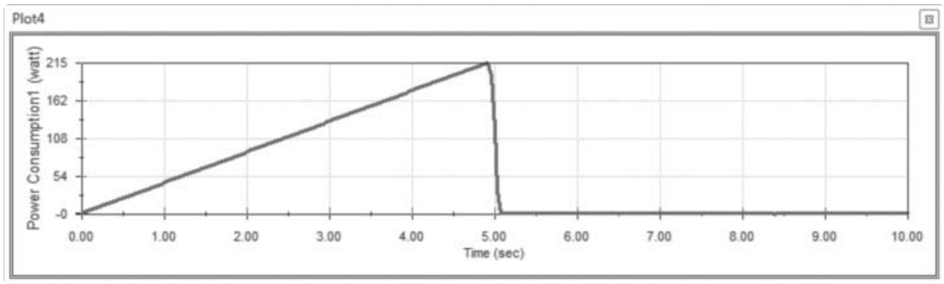
(A) Motor angular velocity



(B) Motor angular acceleration



(C) Motor torque



(D) Motor power consumption

Analysis and compare to hand (manual) calculations. The motion analysis includes calculating speeds, accelerations, forces, and torques. We plot these variables to study their behaviors.

Let us use the following motion specifications:

- ☐ Wheel drum is made out of Steel AISI 304
- ☐ Motor accelerates from 0 to 100 RPM (revolution per minute) in 5 sec
- ☐ After 5 sec, the motion is steady for another 5 sec (i.e., constant 100 RPM and zero acceleration)
- ☐ Ignore bearing friction in motor

We need to calculate the motor torque and power to provide this motion. Based on well-known kinematics and dynamic analysis, we can write the following equations for the angular motion of the motor shaft:

$$T = I\alpha \tag{13.28}$$

$$P = T\omega \tag{13.29}$$

$$\omega^2 = \omega_0^2 + 2\alpha\theta \tag{13.30}$$

$$\omega = \omega_0 + \alpha t \tag{13.31}$$

where I, ω, ω_0, α, θ, and t are respectively the inertia, angular velocity, initial angular velocity, angular acceleration, angle of rotation, and time of the wheel drum motion (rotation). T and P are respectively the motor torque and power.

Based on the motion specifications and knowing that one revolution is 360 degrees or 2π radians and one minute is 60 sec, we write $\omega_0 = 0$, $\omega = [100(2\pi)/60] = 10.472$ rad/sec = 600 deg/s, and $t = 5$ sec. We use SolidWorks to calculate the mass properties of the wheel drum to find its I_{zz} as 34116.70 lb.in^2. We use I_{zz} because the wheel drum rotates about that axis based on how we created the wheel drum model. This is the Z-axis of the model MCS.

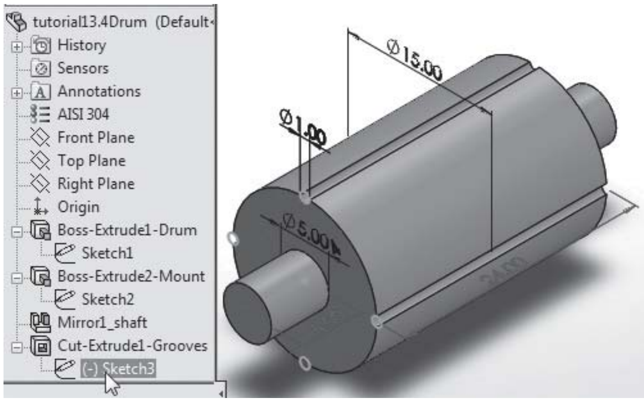
- ☐ Substituting these values into Eq. (13.31), we get $\alpha = \omega/t = 10.472/5 = 2.094$ rad/s^2 = 119.977 deg/s^2
- ☐ Using Eq. (13.30), we get $\theta = \omega^2/2\alpha = (10.472)^2/(2*2.094) = 26.185$ radians
- ☐ Using Eq. (13.28), we get $T = 71440.370$ lb_mass.in = 5953.364 lb_mass.ft = 185.036 lb_force.ft, where 1 lb_force = 1g lb_mass, where g is the gravity acceleration = 32.174 ft/s^2
- ☐ Using Eq. (13.29), we get $P = (185.036*10.472) = 1937.700$ lb_force.ft/s = 1937.700 *1.3558 = 2627.134 Watts = 2627.134 /745.700 HP = 3.523 HP

Summary of hand calculations to use in SolidWorks motion analysis:

- ☐ Angular velocities: $\omega = 0$ at $t = 0$, 600 deg/s at $t = 5$ s, and 600 deg/s at $t = 10$ s
- ☐ Angular acceleration: $\alpha = 119.977$ deg/s^2, constant for 5 sec
- ☐ Mass moment of inertia: $I_{zz} = 34116.70$ lb_mass.in^2
- ☐ Torque: $T = 185.036$ lb_force.ft
- ☐ Power: $P = 2627.134$ Watts

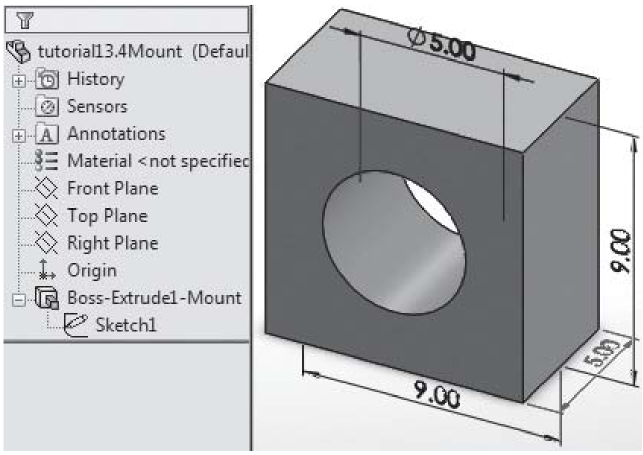
Figure 13.9 shows the graphs generated for the study. As shown in the figure, the hand calculations match the SolidWorks results except for the power consumption. We calculated a power peak of 2627.134 Watts while SolidWorks calculated a power peak of 215 Watts.

Step 1: Create wheel drum: **File** > **New** > **Part** > **OK** > create part shown > extrude drum using **Mid Plane** as shown > apply **Steel AISI 304** material to drum > **File > Save As** > *tutorial13.4Drum* > **Save.**



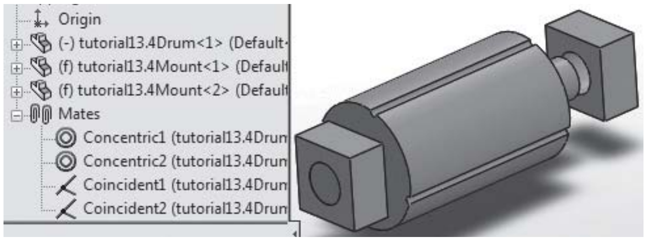
Note: We create four grooves in the wheel drum to help us visualize the drum's rotation during its motion analysis.

Note: We mirror the left shaft about the **Front Plane** to create the right shaft. Use a circular pattern to create *Sketch3*.



Step 2: Create wheel drum mount: **File** > **New** > **Part** > **OK** > create part shown above > extrude block using **Mid Plane** as shown > **File > Save As** > *tutorial13.4Mount* > **Save.**

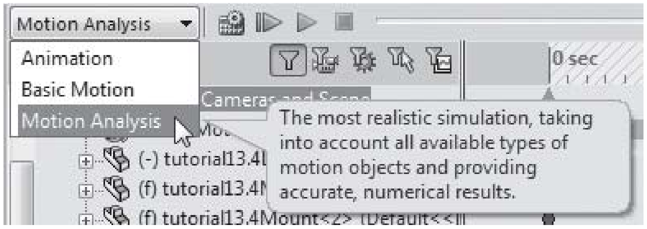
Step 3: Create wheel drum assembly: **File** > **New** > **Assembly > OK** > assemble one instance of wheel drum and two instances of mount > use concentric mate between drum cylindrical face and mount hole face > use coincident mate between drum left shaft flat face and mount flat left face > repeat for drum right shaft and other mount > fix the two mount instances and float the wheel drum as shown > **File > Save As** > *tutorial13.4* > **Save.**



Step 4: Evaluate mass properties: **Evaluate** tab > **Mass Properties.**

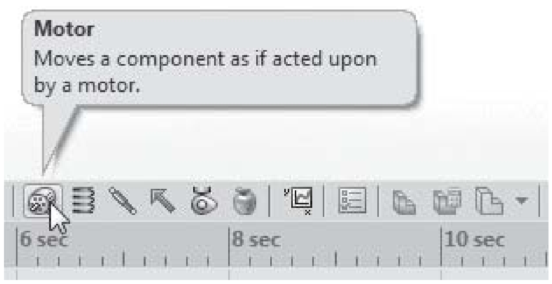
Note: This step is not required by **Motion Analysis**. It provides us with I_{zz} for hand calculations. SolidWorks implicitly calculates it to perform the analysis.

Step 5: Add motion analysis: **Tools** > **Add-Ins** > **SolidWorks Motion.**

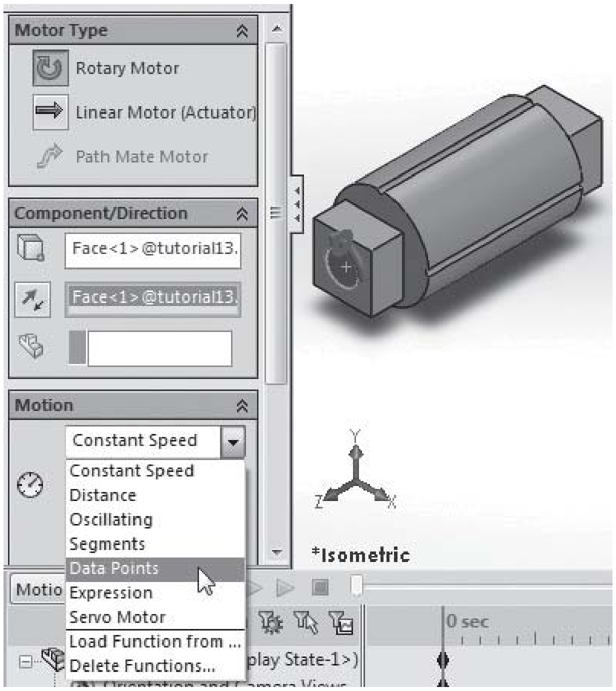


Note: Without this **Add-Ins**, you would not see **Motion Analysis** in the dropdown list of the **Motion Study 1** tab, and shown above.

Step 6: Add a motor: **Motor** on **Motion Study 1** tab tool bar shown.



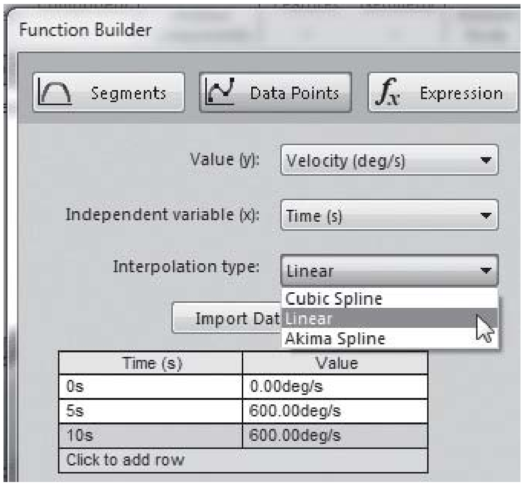
Step 7: Set motor parameters: **Rotary Motor** for **Motor Type** as shown > flat face of wheel drum for **Component/Direction** > **Data points** from **Motion** dropdown as shown to open **Function Builder** window shown in Step 8.

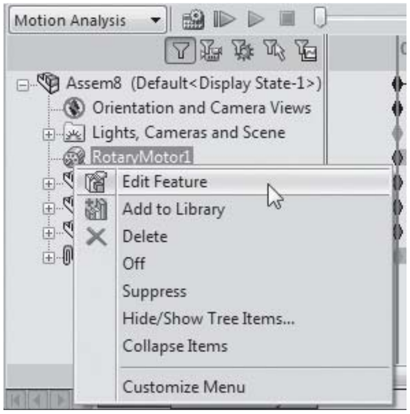


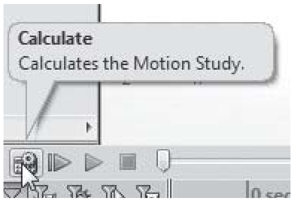
Step 8: Define velocity curve: **Linear** from **Interpolation Type** dropdown shown > click in **Time – Value** table shown to enter data > enter **Time** and **Value** shown > **OK** to exit **Function Builder** > ✔ to exit **Motor** definition phase.

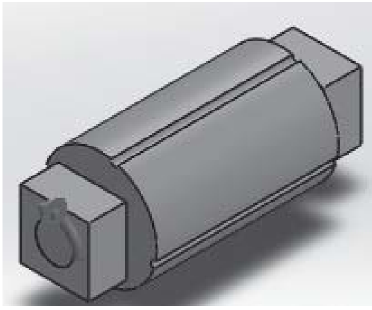
Note: A **Rotary Motor1** node is created in **Motion Study 1** > right-click it to edit it if needed.

Note: Graphs are created after data input.









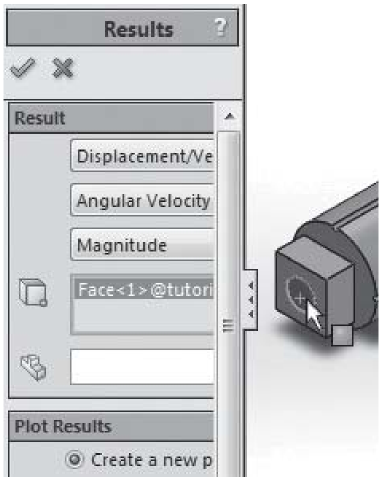
Step 9: Run motion study: Extend the 5 sec key (diamond) created to 10 sec by dragging it > **Calculate** shown above to run analysis.

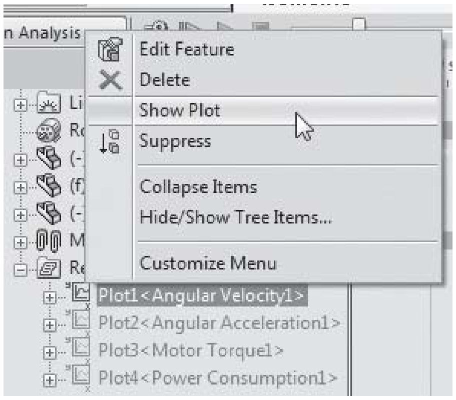
Note: Observe that the wheel drum starts slowly and accelerates for the first 5 sec, then maintains a constant angular speed for the next 5 sec.

Note: The drum rotates in the direction set in Step 7.

Step 10: Create graphs to analyze motion: **Results and Plots** as shown > set parameters as shown for angular velocity > ✔ > repeat for Angular Acceleration, torque, and power consumption.

Note: Expand **Results** node > right-click any plot to edit/show.





Note: This step generates the four plots shown in Figure 13.9. The first two use the **Displacement/Velocity/Acceleration** category. The third plot uses the **Forces** category. The fourth uses **Momentum/Power/Energy**. Select the **Rotary Motor**1 node for this plot. Hover over boxes to read what is required.

HANDS-ON FOR TUTORIAL 13–4. Replace the linear velocity curve of Step 7 by a cubic spline. Rerun the analysis and generate the plots shown in Figure 13.9. How does the cubic spline distribution of angular velocity affect the results? Does it make sense?

Tutorial 13–5: Perform Static Linear FEA on a Part

We run a static linear FEA on a cantilever beam using SolidWorks **Simulation**. We can use different types of finite elements to model the beam. We can use a 3-node 1D beam element, a 10-node 3D tetrahedral element, or a 20-node 3D hexahedral element. SolidWorks **Simulation** does not provide a choice and uses a tetrahedral element. Figure 13.10 shows the stress contours. All dimensions are in mm.

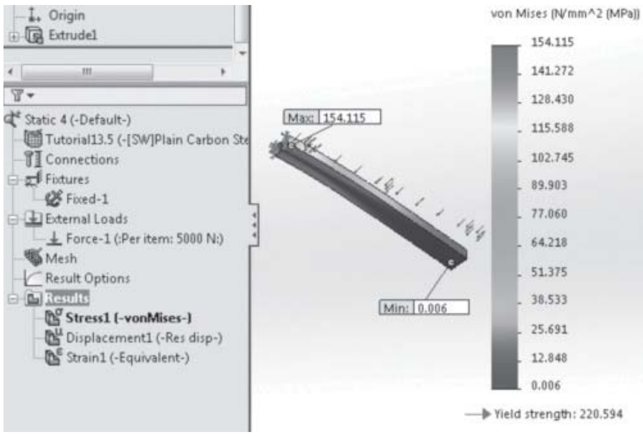
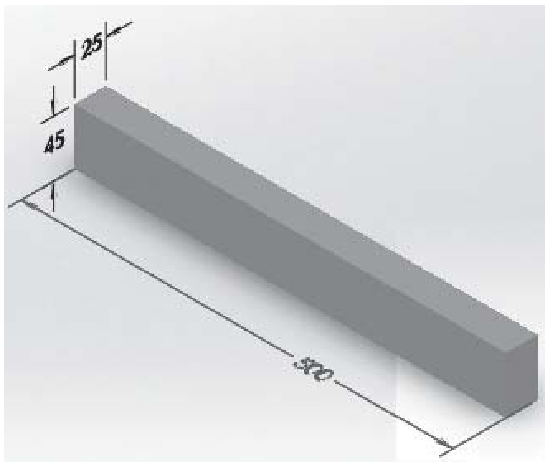


FIGURE 13.10
Stress contours of a cantilever beam

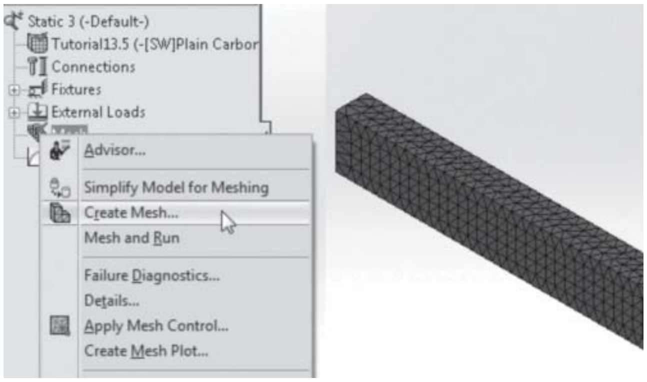
Step 1: Create beam model: **File** > **New** > **Part > OK** > create sketch on Right Plane and extrude 500 mm to right > apply **Plain Carbon Steel** material to beam > **File > Save As** > *tutorial13.5* > **Save.**



Step 2: Start **Simulation**: **Tools > Add-ins** > Solid-Works **Simulation > OK > Study Advisor** on **Simulation** tab that opens up > **New Study** from dropdown > **Static** > ✔.

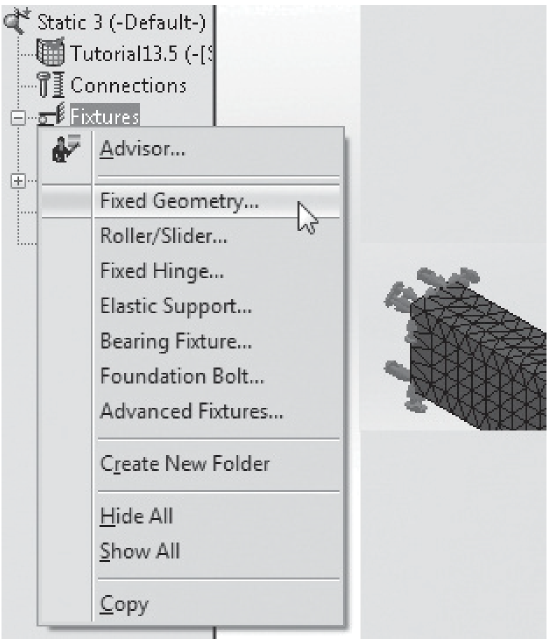
Note: After you add the **Simulation** module, you cannot run **SimulationXpress**. You must remove **Simulation** first. Repeat the sequence above to remove it.

Step 3: Create mesh: Right-click **Mesh** on left pane as shown > **Create Mesh** from popup > ✔.



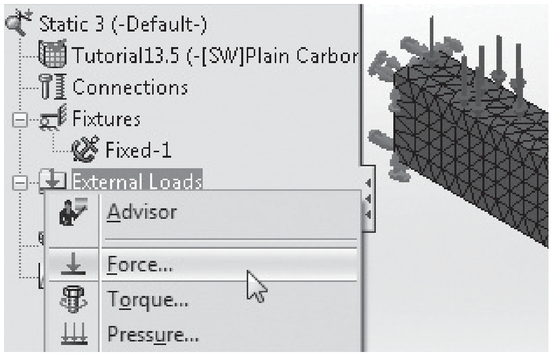
Note: Accept all mesh defaults. You may investigate mesh control later.

Step 4: Apply B.C.: Right-click **Fixtures** node shown > **Fixed Geometry** from popup > back face of beam > ✔.



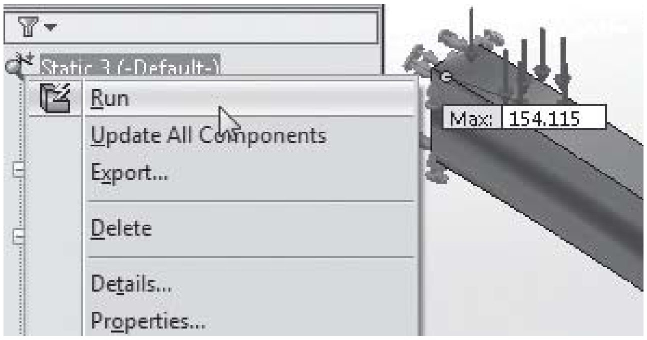
Note: The **Fixed**-1 node is created under the **Fixtures** node on the left pane.

Step 5: Apply external loads: Right-click **External Loads** node shown > **Force** from popup > top face of beam > SI > 5000 > ✔.

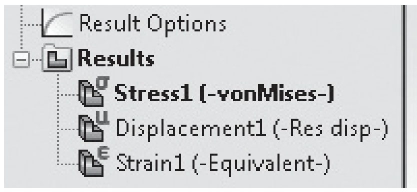


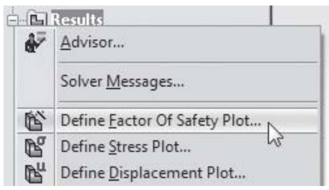
Note: The **Force**-1 node is created under the **External Loads** node on the left pane.

Step 6: Run FEA: Right-click **Static** node shown > **Run**.



Note: FEA runs and displays the stress contours shown in Figure 13.10 and adds the **Results** node to the left pane.





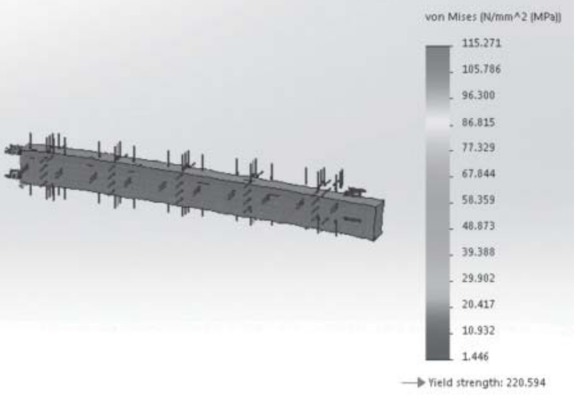
Step 7: Display results: Double-click any result from the list shown above to display and view > right-click **Results** node > **Define Factor Of Safety Plot** to create one.

> ***HANDS-ON FOR TUTORIAL 13–5.*** The **Simulation** tab has many useful analysis tools: (A) perform a design insight: **Results Advisor** on **Simulation** tab > **New Plot** > **Design Insight**; (B) create animation file: **Plot Tools** on **Simulation** tab > **Animate**; (C) generate reports: **Report** on **Simulation** tab; (D) use **Probe** on **Simulation** tab to find stresses at midpoint of beam. Compare the value with beam theory.

Tutorial 13–6: Perform Thermal FEA on a Part

We run a thermal analysis on the beam of Tutorial 13–5. Figure 13.11 shows the stress contours due to thermal loading. All dimensions are in mm.

FIGURE 13.11
Thermal stress contours of a cantilever beam



Step 1: Open beam model: **File** > **Open** > locate part file created in Tutorial 13–5 > **Open.**

Step 2: Start **Simulation**: **Tools > Add-ins** > SolidWorks **Simulation > OK > Study Advisor** on **Simulation** tab that opens up > **New Study** from dropdown > **Static** > ✔.

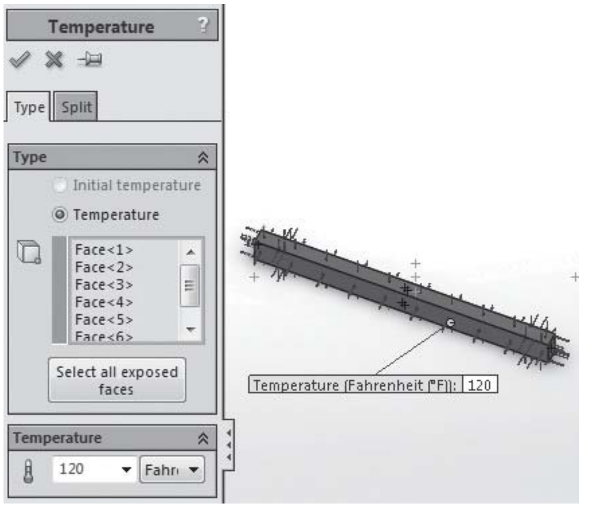
Step 3: Create mesh: Right-click **Mesh** on left pane > **Create Mesh** from popup > ✔.

Note: Accept all mesh defaults. You may investigate mesh control later.

Step 4: Apply B.C.: Right-click **Fixtures** node on left pane on screen > **Fixed Geometry** from popup > back face of beam > ✔.

Note: The **Fixed**-1 node is created under the **Fixtures** node on the left pane.

Step 5: Apply external loads: Right-click **External Loads** node shown > **Temperature** from popup > **Select all exposed faces** button shown > **Fahrenheit** > 120 > ✔.



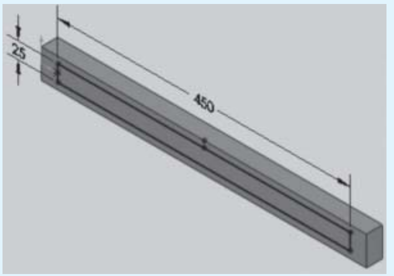
Note: The **Temp**-1 node is created under the **External Loads** node on the left pane.

Step 6: Run FEA: **Run** on the **Simulation** tab.

Note: FEA runs and displays the stress contours shown in Figure 13.11 and adds the **Results** node.

Step 7: Display results: Double-click any result from the list shown above to display and view > right-click **Results** node > **Define Factor Of Safety Plot** to create one.

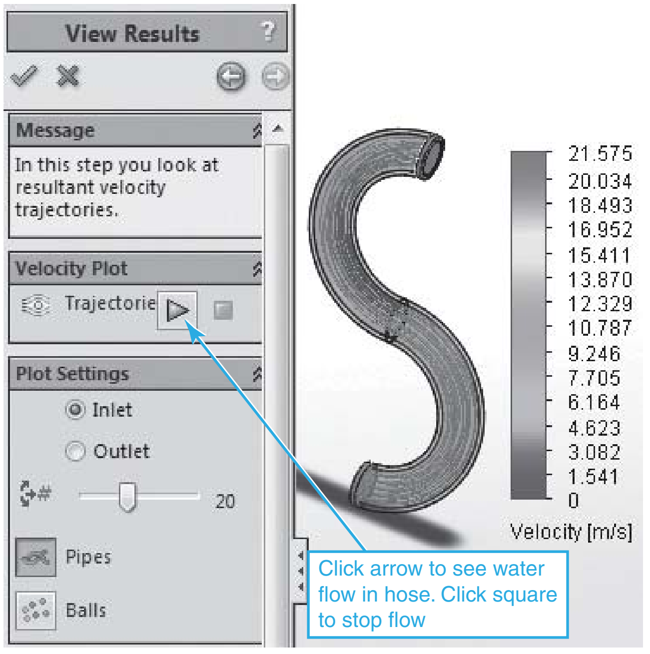
HANDS-ON FOR TUTORIAL 13–6. Create a cut through the center of the part as shown to the right. Rerun the simulation, and report the difference in the thermal stress and displacement values.



Tutorial 13–7: Perform Flow Analysis on a Hose

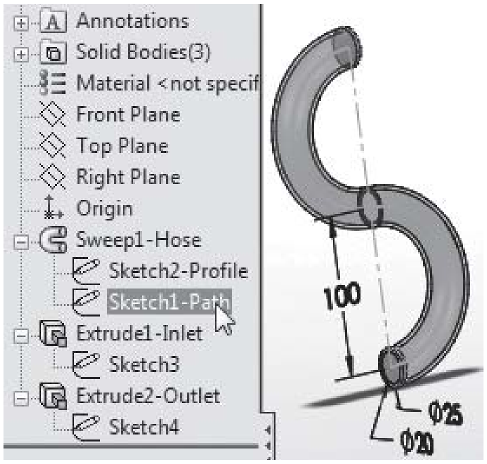
We run a flow analysis on a hose using SolidWorks **FloXpress**. The SolidWorks **FloXpress** tool gives a basic insight into how water or air will flow through a part or an assembly. SolidWorks also has a **Flow Simulation Add-Ins** module that is more comprehensive than **FloXpress**. This tutorial studies the flow of water through a hose using **FloXpress**. Figure 13.12 shows the flow simulation. All dimensions are in mm.

FIGURE 13.12
Flow simulation of a hose



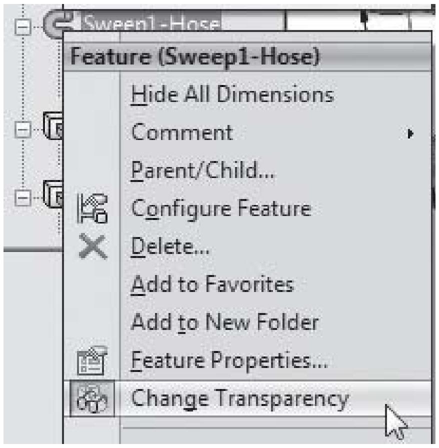
Flow simulation requires a closed control volume that is mandated by the theoretical mathematical formulation it uses. We also need to set up the simulation model correctly. We need to provide B.C. (boundary conditions) at the inlet and outlet of the control volume. There are three variables available: pressure, flow velocity, and flow rate. We specify one and calculate the other two. Our B.C. must create a gradient drop from inlet to outlet to generate flow motion. For example, the inlet pressure must be higher than the outlet pressure to push the flow in the control volume, just as in real life. If the inlet and outlet pressures were equal, water would not flow. Specifying an outlet pressure higher than inlet pressure does not make sense and SolidWorks tells us so.

Step 1: Create hose model: **File** > **New** > **Part** > **OK** > create *Sketch1-Path* (two equal half circles) on **Front Plane** > create *Sketch2-Profile* (two concentric circles shown) on **Right Plane** > create *Sketch3* (circle with diameter of 25 mm) on bottom circular face of sweep > create *Sketch4* (circle with diameter of 25 mm) on top circular face of sweep > **File** > **Save As** > *tutorial13.7* > **Save**



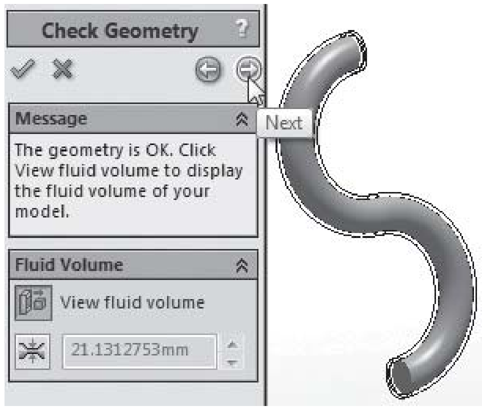
Note: Use 3 mm extrusion thickness for inlet and outlet extrusions (*Sketch3* and *Sketch4*).

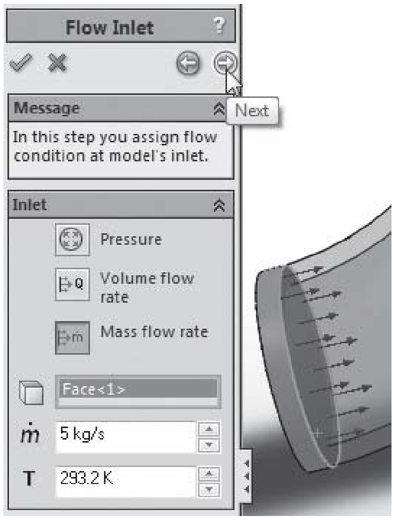
Step 2: Set hose transparency: Right-click *Sweep1-Hose* > **Change Transparency** from popup.

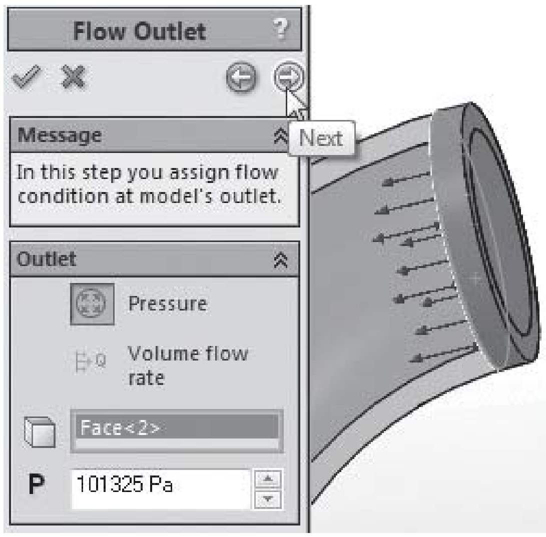


Note: The transparency allows us to see the flow simulation inside the hose.

Step 3: Start flow simulation: **Tools > FloXpress** > **Next** (top right arrow on left pane) > **View fluid volume** to preview the volume that will be studied > **Next.**

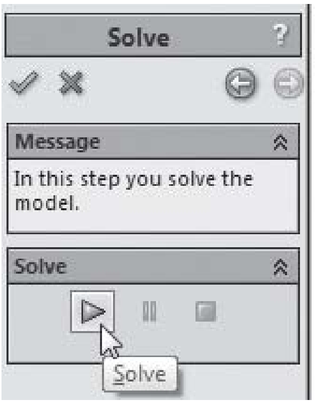


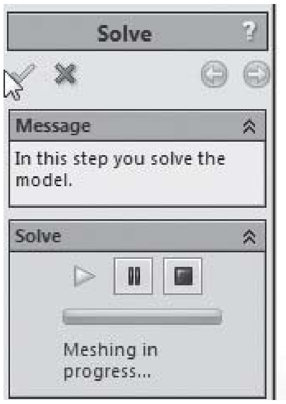


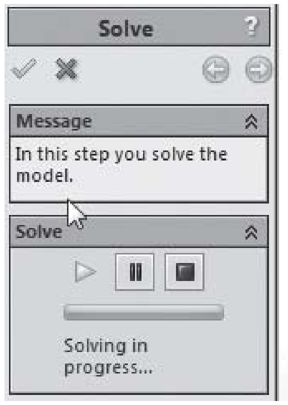


Step 4: Select fluid and set inlet and outlet B.C.: **Water** as fluid to simulate > **Next** > select inside face of *Extrude1-Inlet* > enter inlet B.C. as shown > **Next** > select inside face of *Extrude2-Outlet* > enter outlet B.C. as shown > **Next.**

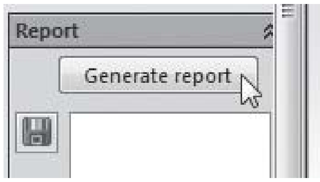
Step 5: Run flow simulation: **Solve** shown to the right > SolidWorks meshes model as shown right > then it solves as shown right > view results as shown in Figure 13.12.







Step 6: Generate report: **Generate report** button that shows up in left pane > a Word document pops up showing the results > save report if needed.



HANDS-ON FOR TUTORIAL 13–7. Reduce the diameters of the hose by half. Run the same **FloXpress** study (i.e., keep flow rate the same). Was there a difference in maximum velocity? Why?

problems

1. Why do we create CAD/CAM geometric models?
2. How do CAD/CAM systems exchange model data among themselves?
3. What are the potential problems that come with data exchange? How are they solved?
4. Use Gauss quadrature to evaluate these integrals. Justify the number of sampling points you use for each problem. Compare with the exact solution:

   a. $\int_1^3 (u^5 - u^3 + 2u)du$

   b. $\int_1^3 (6u^{1/2} - u^{1/6})du$

   c. $\int_2^5 (u^3 - 2u^2 + 6u - 10)du$

   d. $\int_0^1 du$

   e. $\int_3^7 3udu$

   f. $\int_0^1 (5u^2 - 3)du$

5. List and describe the FEM steps.
6. Use the finite element method to check the strength of the brace drill handle shown in Figure 4.18 in Chapter 4. Use a force of 200 N (~45 lb; 1 lb = 4.4482 N) at the drill bottom face. The load applies compression on the drill. Think of it as the environment pushing on the drill. Fix the top round face of the drill for the B.C. Use Alloy Steel for material. What is the factor of safety? Is the model safe?
7. Redo Problem 6, but use a torque of 100 N.m as an external load instead of the force. Apply the torque at the center of the middle part where you grip the drill. Fix both the top and the bottom faces. What is the factor of safety? Is the model safe?
8. Redo Problem 6, but for the football goal post shown in Figure 4.22 in Chapter 4. Apply a uniform pressure of 40 psi on the crossbar and the uprights (this simulates gusting winds pushing on them). Fix the bottom face of the post where it meets the ground. What is the factor of safety? Is the model safe?
9. Redo Problem 6, but for the L bracket shown in Figure 2.35A in Chapter 2. Apply a uniform temperature of 200°F at the vertical front face. What is the factor of safety? Is the model safe?

This page intentionally left blank

PART

V

Part Manufacturing

The primary goal of Part V is to explore and cover part manufacturing. Manufacturing is a very large field. We focus our attention here on basic manufacturing including prototyping, machining, and injection molding. The goal is to provide the reader with some sense of what happens to designs after they leave the design department and how they become products.

Chapter 14 (Rapid Prototyping) covers the concepts of how to produce prototypes to evaluate designs. Chapter 15 (Numerical Control Machining) covers the concepts of machine tools and basic machining processes such as turning, drilling, and milling. Chapter 16 (Injection Molding) describes injection molding as another basic manufacturing process that is different from machining. Machining applies to metals whereas injection molding applies to plastics.

This page intentionally left blank

CHAPTER

14

Rapid Prototyping

14.1 Introduction

Design prototyping has long been a common practice in engineering design. Design and manufacturing engineers always want to ensure that a product design as perceived and visualized on paper (or on a CAD/CAM screen) is what it is. The advantage of verification is that it eliminates any hidden mistakes or surprises that may be discovered during product manufacture. These surprises are costly. Even if there are no design mistakes, holding a prototype in hand may prompt some design changes such as changes in dimensions, aspect ratio, or relocation of features for ease of access by the intended users and maintenance personnel.

Prototyping is not only found in engineering; it is also part of life. Many of us have built prototypes of amateur designs such as building a racing car for the car derby race in Boy Scouts, or building the tallest structure using balsa wood sticks. Engineering students must design and build a prototype of a device as part of their mandatory capstone design course in their senior year.

There exist multiple ways to create prototypes in industry, depending on the industry and the product. For example, the automotive industry has been using prototyping for years to model concept cars. Designers use clay (plasticine) to build a full-size model of a concept design before, during, or after a design is completed. They typically build the car body out of clay. Once finalized, they may use the model to measure the coordinates of points on the body using a CMM (coordinate measuring machine). They feed these coordinates to a CAD/CAM system, and use them to create free-form surfaces of the car CAD model. Alternatively, designers may create the design in a CAD/CAM system and then build a clay model. Almost all companies still produce full-size clay models toward the end of the design process. These models can produce surprises that manufacturers want to avoid before a vehicle enters the tooling and production phases.

Given the value of prototyping and advances in many fields, rapid prototyping (RP) has been in existence for some time and is a common practice today (*3D printing* is a common name for RP). Two factors contribute to the popularity of RP. First, CAD models can easily be converted for prototyping. Second, RP hardware prices have been falling while RP quality has been rising, thus enabling many companies of all sizes and academic institutions to acquire RP machines. The most common machine is a 3D printer that is a desktop machine.

Although RP started as a method of verifying and prototyping designs, it is now used to produce parts that are included in actual products. Also, prototypes can be used as master patterns for injection molding core and cavity inserts, thermoforming, blow molding, and various metal casting processes. That is due to the advancements in prototyping material strength, the accuracy of the building process, and the ability to build prototypes very fast.

Considered as a manufacturing process, RP is distinctively different from traditional manufacturing. RP is a process that is typically referred to as *additive manufacturing* because the process involves adding successive layers of material to build the prototype. In contrast, traditional manufacturing processes such as milling, drilling, and turning are considered to be *subtractive manufacturing* because each process involves starting with a blank (stock) and removing or carving out material until we obtain the final shape.

The primary advantage of additive manufacturing is its ability to create any shape or geometric feature. Another advantage is that RP is a WYSIWYG (what you see is what you get) process whereby the physical (manufactured) model and the RP virtual model are identical, unlike traditional manufacturing where the stock and the final product are very different from each other.

There are key benefits to RP, including time and cost savings as well as accuracy. A prototype can always be produced in several hours to several days, depending on the type of RP machine used and the size and complexity of the model. The accuracy of building a prototype is comparable to traditional manufacturing. Example tolerances are ±0.005 and 0.0015 in. Other benefits of RP are as follows:

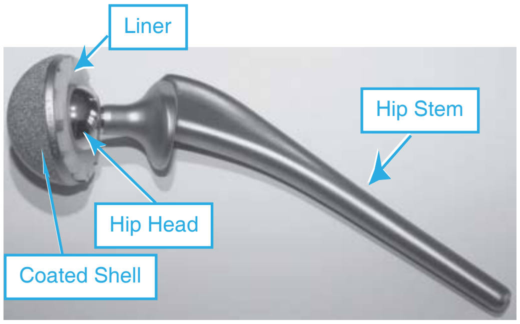
1. Objects of any geometric complexity or intricacy can be produced
2. There is no need for an elaborate machine setup or final assembly
3. The construction is manageable, straightforward, and relatively fast
4. It reduces time to market
5. RP helps you to better understand and communicate product designs
6. It allows us to make rapid tooling for injection molding and investment casting
7. RP may be less expensive than traditional manufacturing

14.2 Applications

By industry sector, the automotive industry is by far the largest user of RP due to the complexity of the CAD models common to automotive designs such as engine blocks, intake manifolds, or exhaust systems. RP has been utilized in many applications beyond its original intended one (prototyping). Here is a list of some common applications:

1. Prototyping (visualization): Holding a prototype in hand, turning it around, and looking at it from all sides is the best way to visualize it; and, yes, it is better than visualizing a 3D CAD model on the screen. No matter how experienced a designer is at reading blueprints and CAD images of a complex object, it is still very difficult to visualize exactly the actual part. Blind holes, complex interior details, and complex free-form surfaces often lead to wrong interpretations. Analogous to the "a picture is worth a thousand words" saying, we have "a prototype is worth a thousand pictures."
2. Verification: Due to the fast build speed of RP, designers have ample time to modify and remodify a design and verify it at a reasonable cost and well within the production and release deadlines. More often than not, especially under deadline pressures, we tend to forgo minor errors although we catch them and know they are there. We tend to fix them in the next revisions. RP helps eliminate this attitude and provide timely visual verification of designs, with reasonable cost.
3. Testing: A designer may need to produce a prototype to test the design before finalizing. An example is testing an airfoil shape by placing it in a wind tunnel. The designer may redesign the airfoil based on the test results.

4. Optimization: This can be thought of as an extension of iterating a design. Design iteration may produce multiple designs to choose from. Optimization criteria could range from size, to weight, to aesthetic appearance.
5. Fabrication: This is an important application of RP that was not intended initially. The use of prototypes as final parts for use is very beneficial because it allows us to speed up production. RP is an essential concept to rapid manufacturing. Sample RP parts that are used as end products, or as components in products, include knee and hip replacements (Figure 14.1), dentures, and ankle bracelets. A common theme among these parts is customizations. Typically, a medical imaging of the patient part (knee, hip, mouth, or ankle) is generated via imaging software. The image is then converted to a CAD model that is used to generate the RP parts. Such a process guarantees a perfect fit of the RP body part in its final placement.



(A) Hip replacement

(B) Knee replacement

FIGURE 14.1
Hip and knee replacements

14.3 Overview

RP is known by various other names: 3D printing, solid free-form fabrication, desktop manufacturing, and layered manufacturing. **Rapid prototyping** is a process that creates physical parts/products from CAD models. The underlying concept of RP is to build a prototype in layers (slices) one on top of the other, from the bottom up. The RP software slices the model into horizontal slices, and the RP hardware (machine) builds the prototype one slice at a time. Stacking the slices up builds the prototype.

Figure 14.2 shows a schematic of the RP process. This schematic is used by any RP process regardless of its details. The steps of the process are as follows:

1. Create the CAD model of the part or assembly: RP is capable of creating one single part or an assembly. Any gaps or loose parts in the assembly are built as such. An assembly created via RP is a functional one.

FIGURE 14.2
RP process

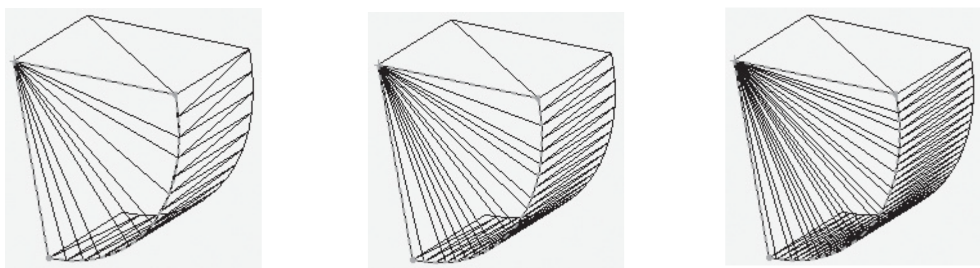
2. Perform pre-processing: The CAD/CAM software triangulates the CAD model and saves the related data in a file in stereolithography (STL) format. This file is transferred to the RP software that reads it, slices the CAD triangulated data, and saves the slice data in a new "build" file.
3. Build the prototype: The RP machine reads the build file and builds the prototype, one slice at a time.
4. Perform post-processing: The prototype from Step 3 requires cleanup at minimum. In some instances, it may need further processing including baking, polishing, and finishing.

14.4 Concepts

An RP process uses a set of concepts regardless of the different commercial implementations. Even though different RP vendors use different RP techniques and build different machines, they all use the same concepts, which are triangulation (tessellation), build orientation, layering (slicing) thickness, and support structure.

A. Triangulation (tessellation): Triangulation represents the first step in the RP process. A CAD/CAM software accesses its B-rep of a CAD model and uses it to triangulate the model. Triangulating a model means converting its B-rep (all faces) to triangles (representing and preserving the model topology), as shown in Figure 14.3. The designer can control the accuracy of triangulating the model. Figure 14.3 shows three different accuracy (resolution) levels generated using SolidWorks. The accuracy directly affects the quality of the resulting physical prototype: the higher the triangulation resolution, the better the quality and accuracy of the prototype. The designer should use higher resolution if the resulting prototype is used as a final product or for experimental testing. Low or medium resolution should suffice if the model is used for visualization or verification.

   Figure 14.3 shows that the resolution affects only the nonplanar faces of a CAD model. The top face shown in Figure 14.3 is triangulated into two faces regardless of the resolution we use. The front planar face of the model has many

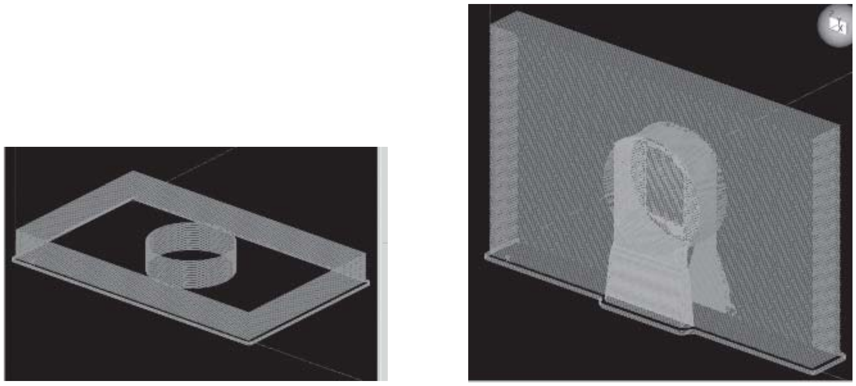


(A) Low (coarse) resolution (60 triangles) (B) Medium (fine) resolution (92 triangles) (C) High (custom) resolution (136 elements)

FIGURE 14.3
Triangulating a CAD model

triangles because it is affected by the triangulation of the cylindrical face. It is this face that is affected significantly by the resolution.

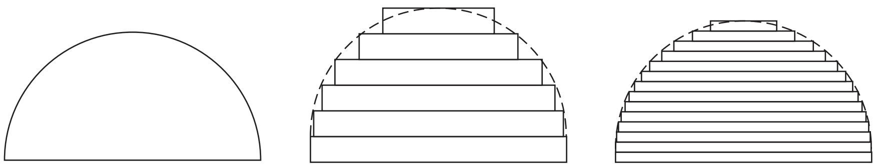
B. Build orientation: The build orientation of a prototype affects its build time and its support structure. Figure 14.4 shows what we mean by build orientation. The figure shows a block with a hole in the center and demonstrates two build orientations: horizontal and vertical. The horizontal orientation shown in Figure 14.4A is better to use for this part because it needs no support structure, and therefore less build time (We have found out that the resolution of the triangulation has an insignificant impact, if any at all, on the build time). We can control the triangulated (faceted) model orientation in the RP software regardless of the model orientation in the CAD/CAM software. Also, the RP software allows us to manipulate (rotate, translate, and/or scale up or down) the triangulated model.



(A) Horizontal orientation (B) Vertical orientation

FIGURE 14.4
Build orientation of a model

C. Layering (slicing): The layer (slice) thickness affects RP in two ways: model accuracy and build time. The user controls the layer thickness in the RP software during the slicing step. The larger the number of slices, the smaller the slice thickness, and therefore the higher the build time and the more accurate the model. Figure 14.5 shows a schematic to illustrate the effect of the slice thickness.



(A) Model (B) Thick layers (slices) (C) Thin layers (slices)

FIGURE 14.5
Effect of layer thickness

D. Support structure: Support structure is needed to build the prototype slice by slice. The structure serves as scaffolding. We need a support structure any time the model boundary starts curving into nonmaterial zones. Consider building the prototype of the model shown in Figure 14.4. Building the prototype in the horizontal orientation (Figure 14.4A) does not require any support structure because each new slice is fully supported by the previously built slice. However, building the model in the vertical orientation (Figure 14.4B) requires a support structure immediately after we reach half the hole. After the half mark, the hole slices start hanging inward into the air, thus requiring support; otherwise, they fall to the hole bottom. In anticipation of this need, the RP software begins building the support structure from the base up along with building the prototype slices as shown in Figure 14.4B.

Keep in mind that the RP build material is usually in a liquid or semiliquid state during build time. Thus, it does not have the strength to hold its own weight. Note that the support structure for the hole, shown in Figure 14.4B, extends all the way down to the bottom of the RP platform to transfer the weight to the ground. RP software figures out the needed support structure automatically based on the orientation of the sliced model. The only intervention that the user does is to orient the model properly to avoid triggering the need for support structure as much as possible, as Figure 14.4 shows.

If a support structure cannot be avoided, the user must use the orientation that produces the least support structure. Support structures should be avoided as much as possible because they increase both the build time and the post-processing (cleanup) time, as well as consuming more material to build the prototype. A support structure is usually built from a separate material that is weaker than the build material itself. The support structure is also built as a weak structure so that it is easily broken and removed during the post-processing of the prototype. Typical shapes of support structures are honeycomb or thin strips.

A special type of support structure is the base support structure. Unlike a support structure, a base support structure cannot be avoided at all. It is a structure that the RP machine inserts between its platform and the prototype it builds. This structure is needed to facilitate removing the prototype from the machine platform. Figure 14.6 shows an example of a base support structure. After the RP process is complete, the prototype is broken (sheared) off the platform by hand or with the help of a knife.

FIGURE 14.6
Base support structure

14.5 SolidWorks Triangulation

Generating an STL (triangulated) file from a CAD model is a simple one-step activity. When we save the CAD model, we save it in STL format, thus generating a file with the *.stl* extension. We transfer the file to a RP machine via any common method (USB, network, e-mail, CD, or DVD). We open the file using the RP software and process it to generate the build (slices) file. Thus, we view the STL file as the interface between the CAD/CAM system and the RP machine.

While a CAD part or assembly is open in SolidWorks, click this sequence: **File** (menu) > **Save As** > select **STL (*.stl)** from the **Save as type** drop-down list > **Save**. We can save the parts of an assembly into one file by checking off the box shown in Figure 14.7. In such a case, the assembly is built as one prototype; otherwise, its parts are built individually and must be assembled manually. In some assemblies, this is not possible.

SolidWorks allows us to control the resolution of the model triangulation (see Figure 14.3) before saving the file. Click the **Options** button on the **Save As** window

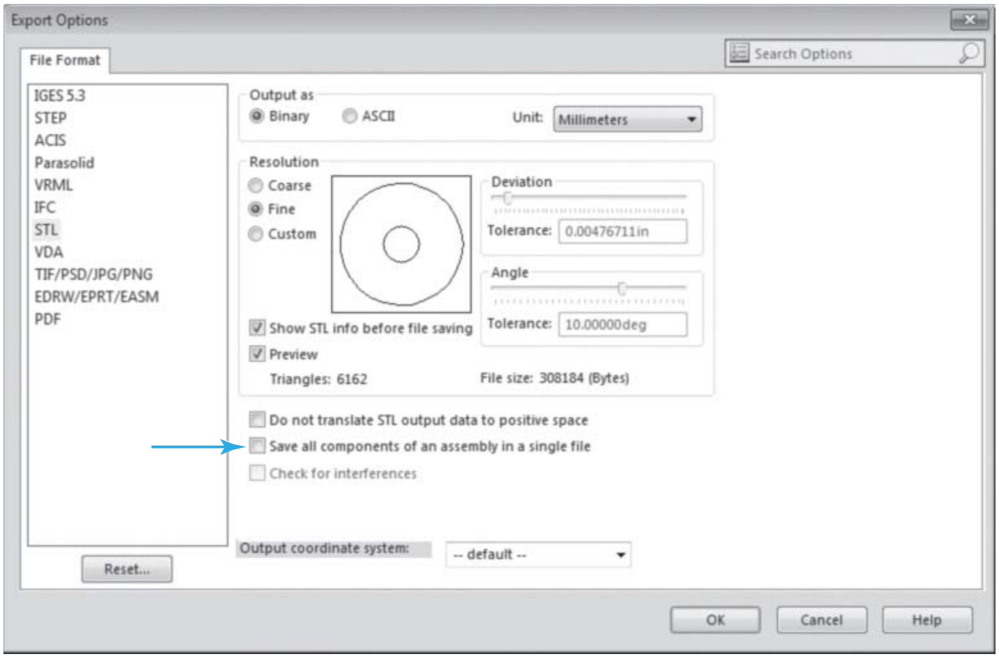
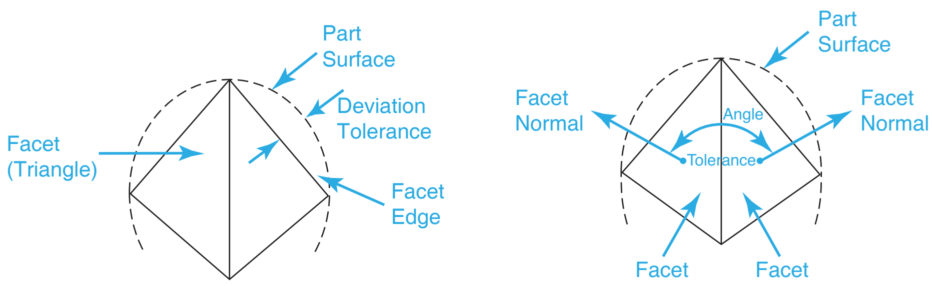


FIGURE 14.7
STL export options

(not shown here) to access the **Export Options** window shown in Figure 14.7. As also shown, we can generate a binary or ASCII file. We usually use the binary format because the size of the resulting file is smaller than the ASCII file.

There are three resolutions to select from as shown in Figure 14.7: **Coarse**, **Fine**, and **Custom**. In addition, there are two tolerances that affect quality of the triangulation: **Deviation** and **Angle**. Triangulation replaces the CAD model with a faceted model that consists of facets (triangles). A **facet** is a small planar (flat) face that is a triangle. These facets need to be as close as possible to the model face (surface) they replace. Such closeness is a measure of the quality of the model tessellation. It is controlled by specifying the **Deviation** and **Angle** tolerances shown in Figure 14.7 and explained in Figure 14.8. The smaller these tolerances, the more accurate the STL representation of the CAD model is. Deviation tolerance controls the maximum distance allowed between the edge of a facet and the edge of the corresponding face (surface) of the part, as shown in Figure 14.8A. Deviation is also known as the *chord distance* or the *surface tolerance*. Angle tolerance controls the change of orientation between two adjacent facets, as measured by the angle between their normal vectors as shown in Figure 14.8B.



(A) Deviation tolerance

(B) Angle tolerance

FIGURE 14.8
Definitions of STL tolerances

14.6 Steps

Section 14.3 provides an overview of the RP steps that start with a CAD model and end with a prototype of the model. We elaborate on these steps here.

1. Create the CAD model of the part or assembly: This step does not require any special attention because of RP. RP is just another application like mass properties and FEM/FEA.
2. Perform pre-processing: The pre-processing starts in the CAD/CAM software and ends in the RP software. Save the CAD part as an STL file. The file contains the coordinates of the vertices of the triangles (facets) and the outward normal vectors of these triangles.

   Model orientation used for building the part is important for several reasons. First, properties of prototypes vary from one coordinate direction to another. For example, prototypes are usually weaker and less accurate in the Z direction than in the XY plane. Second, part orientation partially determines the amount of time required to build the model. Placing the shortest dimension in the Z direction reduces the number of layers or slices, thereby shortening the build time.

   The RP software uses a layer thickness ranging from 0.01 mm to 0.7 mm, depending on the RP technique in use. The program may also generate a support (auxiliary) structure to support the model during the build. The sliced model and its support structure are stored in a build file.
3. Build the prototype: In this step the RP machine actually begins constructing the prototype using the build file. The RP machine builds one layer (slice) at a time using RP build material such as polymer (resin), paper, or powdered metal. Most machines are fairly autonomous, needing little human intervention.
4. Perform post-processing: This is the final step. It involves removing the prototype from the machine and detaching the support structures that have been built. Some photosensitive materials need to be fully cured before use. Prototypes may also require minor cleaning and surface treatment. Sanding, sealing, and/or painting the model usually improve its appearance and durability, and may be done if needed.

14.7 Building Techniques

Stereolithography apparatus (SLA) is commonly considered to have been the first RP technique (machine). It was developed by 3D Systems. Since then, a number of different RP techniques have become available. The following RP techniques are used by commercial RP systems:

- ☐ Stereolithography (SLA)
- ☐ Laminated object manufacturing (LOM)
- ☐ Selective laser sintering (SLS)
- ☐ Fused deposition modeling (FDM)
- ☐ Solid ground curing (SGC)
- ☐ 3D printing (3DP)

Each RP technique uses a certain build material and a corresponding process to build and cure the material. For example, SLA, LOM, SLS, and FDM use, respectively, photopolymer (resin), paper, thermoplastic/metal powder, and thermoplastic/eutectic metals or wax.

As for the curing process, SLA uses a laser to solidify the resin layers. LOM glues the layers together and cuts them to shape with a knife or laser cutter. SLS curing is similar to SLA. FDM deposits molten material as beads following the layer cross-section boundary. SGC uses a laser to harden resin, similar to SLA.

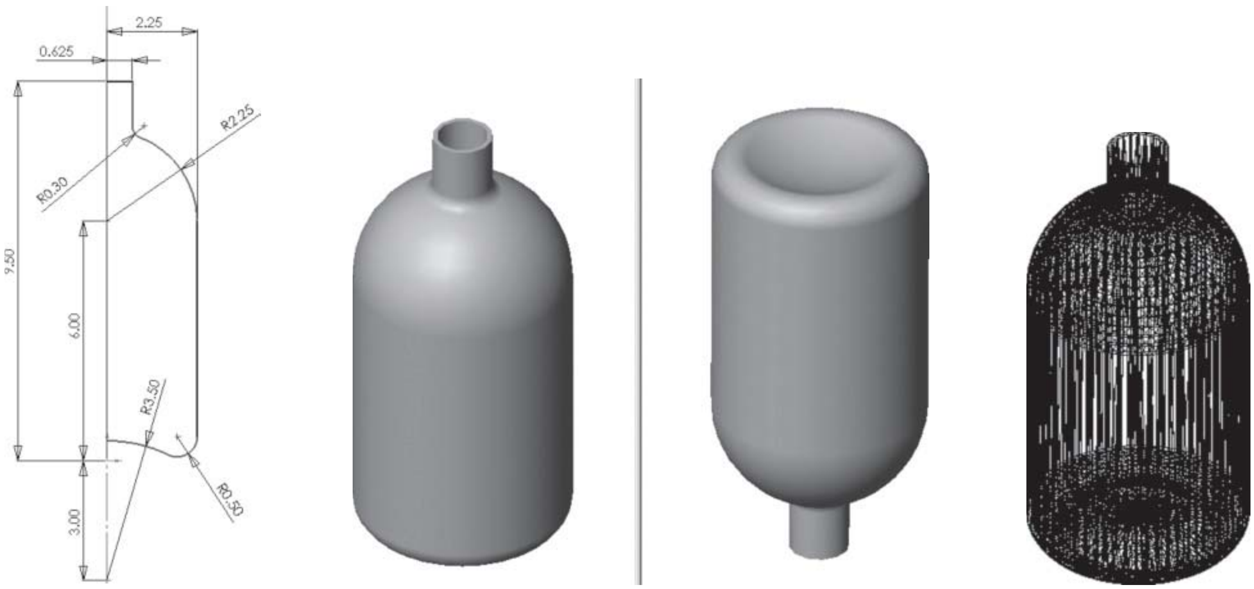
3D printing uses many materials, for example, powdered ceramics, powdered plastics, ABS, thermoplastics, and Ultem. 3D printing uses layering like the other techniques, but it is generally faster, more affordable, and easier to use. 3D printing has the ability to use several materials in the same build process, which is useful for prototyping an assembly made of numerous materials. 3D printing is used in jewelry, footwear, dental, medical, automotive, aerospace, and other industries.

14.8 Bottle Prototype

This section provides a complete example from slicing an STL file to building and cleaning the prototype. We create a prototype of the bottle shown in Figure 14.9. We select this bottle because of the interesting challenges it offers. First, what is the best orientation to build it: horizontal or vertical? Second, what is the effect of its curved bottom on the base support structure? Here are the generic steps to create the prototype:

1. Create the CAD model: Figures 14.9A and 14.9B show the bottle model. All dimensions are in inches. It is a revolve.
2. Generate the STL file: Use **Fine** resolution (Figure 14.7), and export the CAD model as an STL file: **File > Save As > STL** format > **Options > Fine** > enter bottle for file name > **Save**. Figure 14.9C shows the STL model.
3. Generate the build file: The RP machine software reads the STL file and displays the model. It also provides the designer with functions to manipulate the model: pan, rotate, and scale. This model is best built in the vertical orientation shown in Figure 14.9C.

   After setting up the model orientation, we select the slicing thickness (layer resolution). The RP software allows us to select from a recommended resolution list because the slice thickness is determined based on the RP machine build accuracy, the build material, and other factors.



(A) **Front Plane** sketch (B) CAD axisymmetric model (revolve) (C) STL model

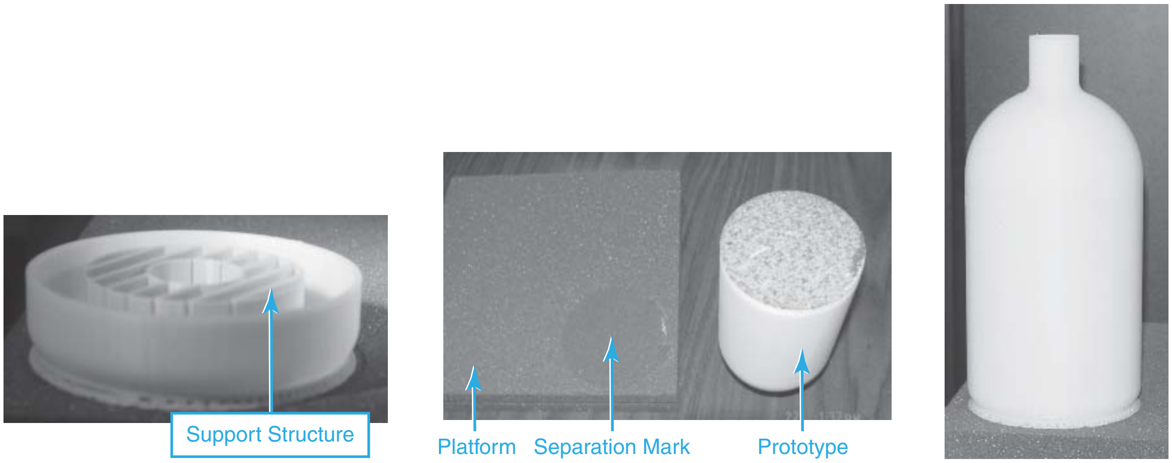
FIGURE 14.9
A bottle (all dimensions are in inches)

We then select the style of the support structure (break-away, sparse, or other type). The base support structure fills the curved gap at the bottom of the bottle to support the layer build as it goes up in the Z direction. The support structure is created inside the bottle and extends from the curved bottom of the bottle all the way up to the bottom of its neck (Figure 14.9C).

RP machines are designed to build more than one prototype in one build to alleviate the long build time. If we can fit multiple models on the machine platform, the machine builds them concurrently.

The output from the software is a proprietary build file that the RP machine reads to build the prototype.

4. **Build the bottle prototype:** Start the machine to build. Make sure there are enough build and support materials. The machine builds and does not have to be attended during build. Figure 14.10 shows the build progress, the machine platform on which the model is built, and the final model.



(A) Build progress (B) Build platform (C) Prototype model

FIGURE 14.10
Prototype model progression during build time

5. **Remove and clean up the prototype:** This is the post-processing of the prototype. Use a putty knife to separate the prototype from the machine platform. Once separated, clean up the bottom of the bottle by scraping it. Removing the support structure from inside the bottle prototype requires more careful handling or a dissolving solution in which we submerge the model for a period of time.

14.9 Tutorials

The theme for the tutorials in this chapter is to practice the triangulation concepts of SolidWorks faceting. Other important concepts such as investigating orientation, slicing, and support structure are not covered because they are machine dependent.

Tutorial 14–1: Generate Part Prototype File

We create STL prototype files for a number of parts of varying complexity as shown in Figure 14.11.

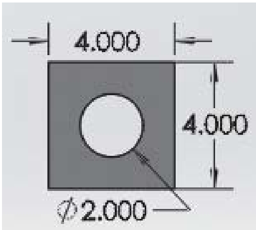
Model STL
(A) Block with hole

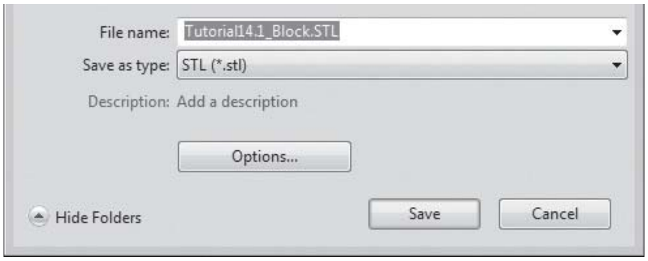
Model STL
(B) Sphere

See Fig 9.10
(C) Air duct

FIGURE 14.11
Prototype models

Step 1: Create block: **File** > **New** > **Part** > **OK** > create sketch on **Front Plane** and extrude 2.0 in. using **Mid Plane** > **File** > **Save As** > *block* > **Save.**

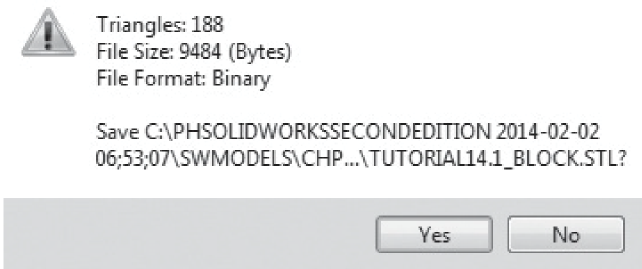




Step 2: Access STL file parameters: **File** > **Save As** > **STL (*.stl)** from **Save as type** dropdown shown above > **Options** shown above.

Note: This step allows you to control the tesellation parameters as shown in the next step.

Step 3: Set STL file parameters: **Export Options** window opens up (Figure 14.7) > **Binary** for **Output** > **Fine** for **Resolution** > check off **Show STL info before file saving** box > check off **Preview** box to read data > **OK.**



Step 4: Review STL mesh (triangles): **Save** button shown in Step 2 to open window shown above > **Yes** when done reviewing to close.

Note: This step also displays the STL model shown in Figure 14.11A.

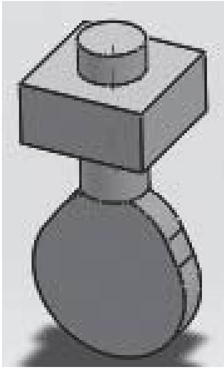
Step 5: Repeat Steps 1–4 for sphere and air duct.

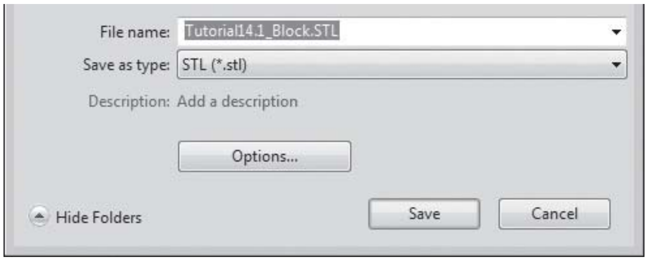
HANDS-ON FOR TUTORIAL 14–1. Report the number of triangles and the file size for the sphere and the air duct.

Tutorial 14–2: Generate Assembly Prototype File

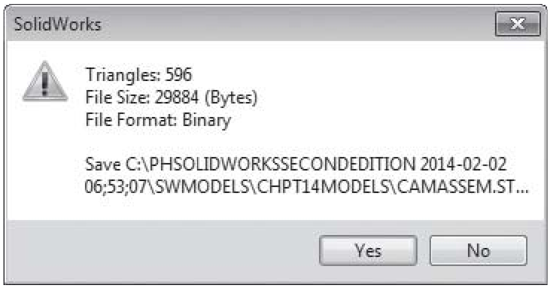
We create one STL prototype file for an assembly, enabling us to build the entire assembly as one functional prototype.

Step 1: Open cam assembly and access STL file parameters: **File** > **Open** > locate cam assembly file from Chapter 6 > **Open** > **File** > **Save As** > **STL (*.stl)** from **Save as type** dropdown shown > **Options** shown.



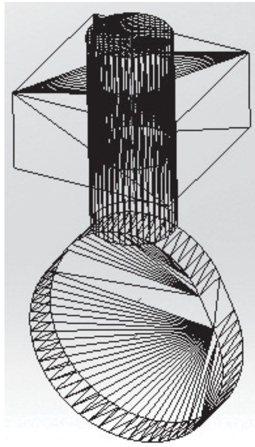


Step 2: Set STL parameters: **Export Options** window opens up (Figure 14.7) > **Binary** for **Output** > **Fine** for **Resolution** > check off **Show STL info before file saving** box > check off **Save all components of an assembly in a single file** > check off **Preview** box to read data > **OK.**



Step 3: Review STL mesh (triangles): **Save** button shown in Step 1 to open window shown above > **Yes** when done reviewing to close > review STL model shown.

Note: The assembly prototype is functional with all components moving as intended.

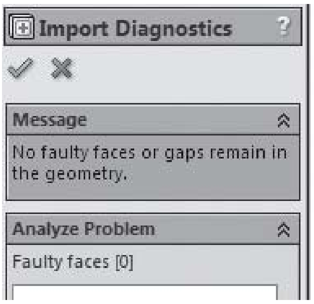


HANDS-ON FOR TUTORIAL 14–2. Regenerate the STL file using Coarse and Custom resolutions. Compare the number of triangles and the file size for each resolution to the number shown here.

Tutorial 14–3: Read Back an STL File

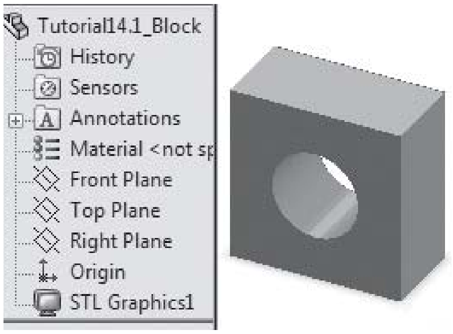
We read (open) the STL files we created in Tutorials 14–1 and 14–2 back in SolidWorks. Note that SolidWorks converts the STL to a “dumb” solid because it cannot recognize its features or create a features tree for it. Thus, we cannot edit any of the original features of the part or edit the assembly mates of the assembly. Unlike reading IGES of STEP files, SolidWorks (or any CAD/CAM system) has no way of knowing part features or assembly components from the STL file because the file is only a B-rep; all it has are the facets (triangles) defining the model faces.

Step 1: Open Tutorial 14–1 STL file of block part: **File** > **Open** > locate Tutorial 14–1 STL file of block > **Open > Yes shown above** to run diagnostics > ✔.

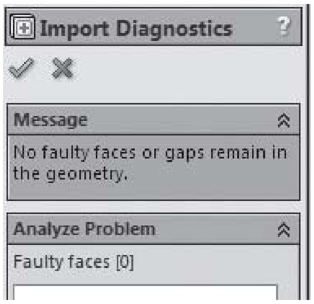


Note: No problems are reported as shown.

Step 2: Review dumb part: Features tree shown displays only one node (*STL Graphics1*) for the imported STL part, thus rendering the model useless to edit or modify. A good reason to read STL files in a CAD system is if they are the only available files to deal with.



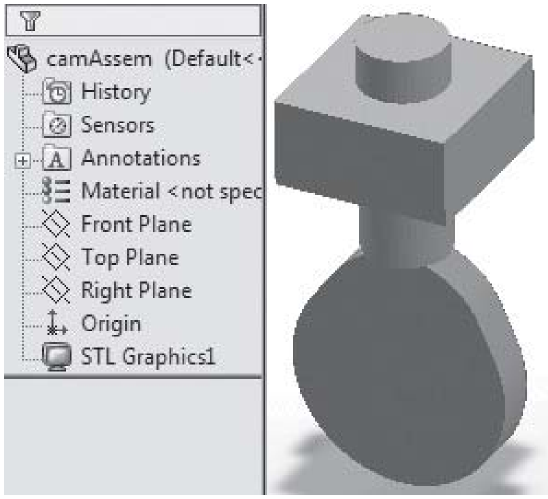
Step 3: Open Tutorial 14–2 STL file of cam assembly: **File** > **Open** > locate Tutorial 14–2 STL file of cam assembly > **Open > Yes** shown above to run diagnostics > ✔.



Note: No problems are reported as shown.

Step 4: Review dumb assembly: Features tree shown displays only one node (*STL Graphics1*) for the imported STL assembly.

Note: If you were to inspect closely the pin of the assembly or the hole of the block, you would recognize the facets and effect of tessellation.



HANDS-ON FOR TUTORIAL 14–3. Save the dumb part and assembly back as STL files. What happens? Explain your answer.

problems

1. What is the difference between rapid prototyping and traditional manufacturing?
2. List the benefits of rapid prototyping.
3. List and describe some of the applications of rapid prototyping.
4. List and describe the steps of the rapid prototyping process.
5. How many triangles does the faceted model of a block (cube with no holes) have? Why? What is the effect of the triangulation resolution?
6. Why is orienting cylindrical holes vertically better than doing it horizontally for building a rapid prototype?
7. How does the slicing (layering) thickness affect the quality of rapid prototyping?
8. Describe the two types of support structures for prototyping.
9. List and describe the concepts of rapid prototyping.
10. Explain and sketch what is meant by deviation tolerance in triangulation. How does it affect the triangulation accuracy?
11. Explain and sketch what is meant by angle tolerance in triangulation. How does it affect the triangulation accuracy?
12. List and describe the steps of the rapid prototyping process.
13. Research the stereolithography (SLA) technique. Submit a detailed report about how the technique works, the build and support structure material, the model slicing and orientation, the machine operation and how it builds the prototype, the pre-process operations, the post-process operations, and the cleanup and painting. Include a schematic of the SLA machine to show its operation mechanism including the laser operation, the vat, and the photopolymer.
14. Redo Problem 13 but for laminated object manufacturing (LOM).
15. Redo Problem 13 but for selective laser sintering (SLS).
16. Redo Problem 13 but for fused deposition modeling (FDM).
17. Redo Problem 13 but for 3D printing (3DP).
18. Use SolidWorks to generate the faceted model of the holed block shown in Figure 2.2 of Chapter 2. What is the best orientation to build the prototype? Why?
19. Redo Problem 18 but for the model shown in Figure 2.9 of Chapter 2.
20. Redo Problem 18 but for the AMP connector shown in Figure 2.30 of Chapter 2.
21. Redo Problem 18 but for the flange model shown in Figure 2.32 of Chapter 2.
22. Redo Problem 18 but for the L bracket shown in Figure 2.35 of Chapter 2.
23. Redo Problem 18 but for the pattern block shown in Figure 2.36A of Chapter 2.
24. Redo Problem 18 but for the pattern wheel shown in Figure 2.36B of Chapter 2.
25. Redo Problem 18 but for the loft feature shown in Figure 4.12 of Chapter 4.
26. Redo Problem 18 but for the helical spring shown in Figure 4.20 of Chapter 4.
27. Redo Problem 18 but for the assembly shown in Figure 6.14 of Chapter 6.
28. Redo Problem 18 but for the Couch assembly shown in Figure 6.17 of Chapter 6.
29. Redo Problem 18 but for the Candle Holder assembly of Problem 6.18 of Chapter 6.
30. Redo Problem 18 but for the Ballpoint Pen assembly of Problem 6.19 of Chapter 6.

CHAPTER 15

Numerical Control Machining

15.1 Introduction

Manufacturing a part is the culmination of product design. We discussed the manufacturing and CAM processes in Chapter 1. A variety of manufacturing processes exists. The manufacturing engineer (process planner) selects a suitable manufacturing process depending on the part material and design tolerances. The process planner typically has a strong background in and knowledge of manufacturing processes. With this knowledge, the planner may request design changes from the design engineers to simplify part manufacturing and, as a result, reduce manufacturing cost. For example, machining a flat blind hole is harder and more expensive than machining a conical blind hole. The former requires a milling operation with a flat end mill, whereas the latter requires a drilling operation with a drill bit.

Different types of manufacturing also exist. Mass production is the type of manufacturing in which hundreds or thousands of the same part or product are produced on a mass scale. Examples include consumer products such as cars, cell phones, and computers, to name a few. The advantage of mass production is the cost reduction realized from setting up a production operation once to produce many units. Another type of manufacturing is job shop, which is the other extreme. In job shop, we produce either one-of-a-kind pieces or a small number (tens) of units. There is also the concept of mass customization whereby companies try to combine the benefits of scale (less cost) with the benefits of meeting customer requirements.

The existing manufacturing processes (methods or operations) can be categorized based on the material they can process. The oldest traditional methods are cutting metals (steel and aluminum). These processes include turning, drilling, milling, grinding, boring (used for large holes), reaming (used for holes up to 1-inch diameter), lapping and honing, broaching, casting, forging, and EDM (electrical discharge machining). Manufacturing methods for polymers and plastics include injection molding.

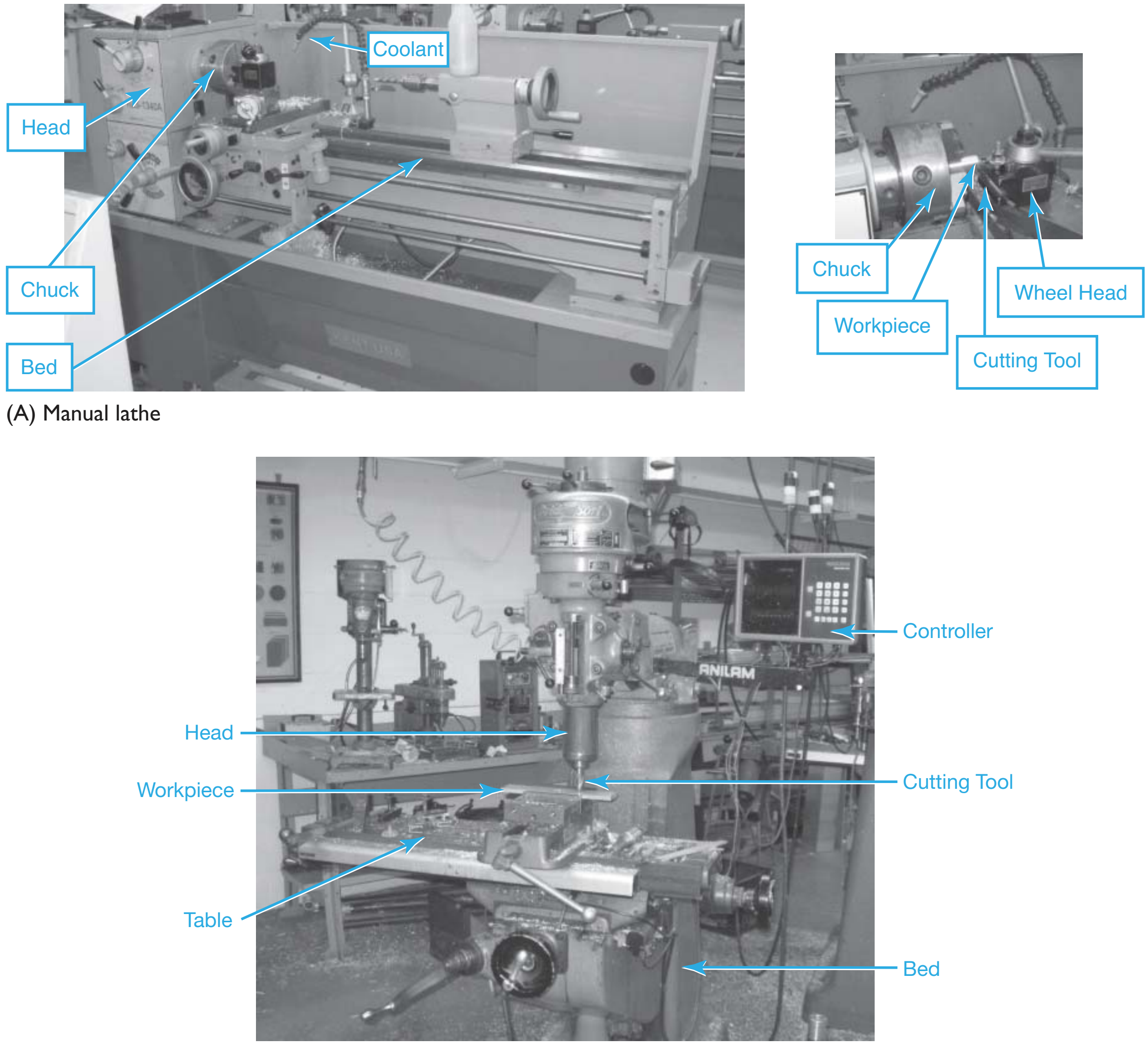
A machining process directly influences the part accuracy and surface quality (roughness). For example, a turning process produces less surface quality than do honing and lapping. Milling is much better than casting and forging.

We cover only four basic machining processes in this chapter: turning, drilling, milling, and EDM. The goal is to provide a basic understanding of machining so we can become better designers and be able to connect design and manufacturing. Thus, we will understand the production and cost implications of specifying tighter or looser tolerances.

Machining processes can be done manually or programmatically. Although manual machining is not used too often, it still serves as a good foundation to learn machining. Programmed machines are more common. These computer-controlled machines are known as *NC* (numerical control or numerically controlled) *machines*. Machine shops may have both manual and NC machines. We cover NC machines here and how to program them.

15.2 Basics of Machine Tools

Machining machines are known as *machine tools*. A **machine tool** is a manufacturing machine that performs one or more machining operations. For example, we can perform milling, drilling, reaming, boring, tapping, and threading using a milling machine. We need to understand both how machine tools work and what the basics of NC machining are. This section covers the basics of machine tools. Subsequent sections cover the basics of NC machining. Figure 15.1 shows sample machines. (We use the terms *machine tools* and *machines* interchangeably throughout the chapter.)

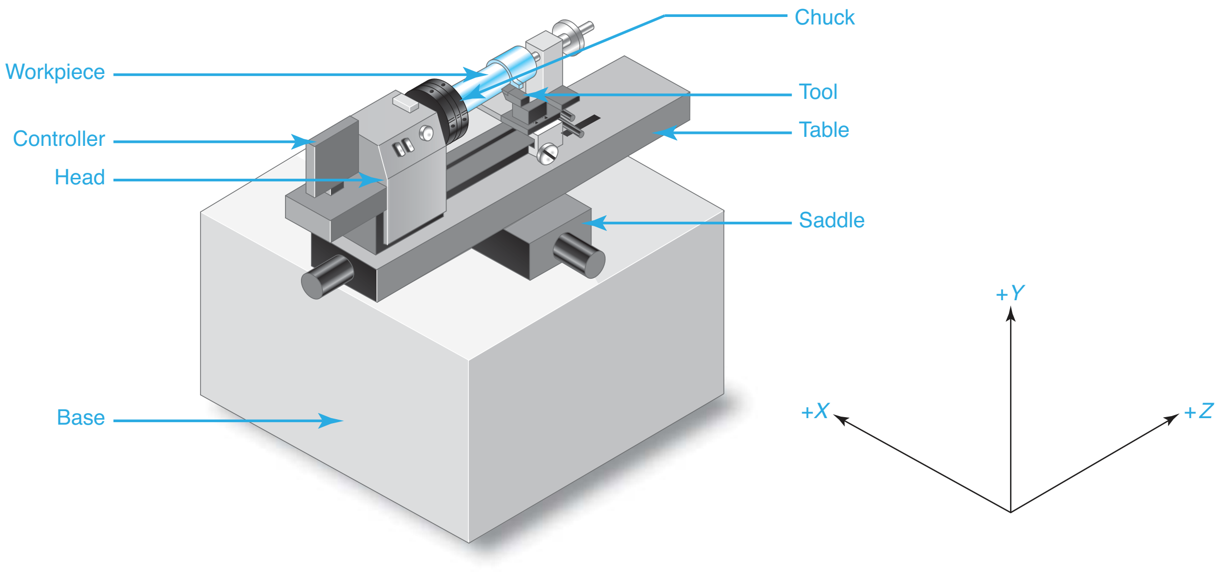


(A) Manual lathe

(B) NC milling machine

FIGURE 15.1
Machine tools

Figure 15.2 shows abstractions (schematics) of the machine tools shown in Figure 15.1. The abstract machine tool helps us understand its basics and its setup and operations. A machine tool has a solid heavy frame typically made out of cast iron. The bed, column (part that holds the head), and head shown in Figure 15.2 are part of the



(A) Lathe

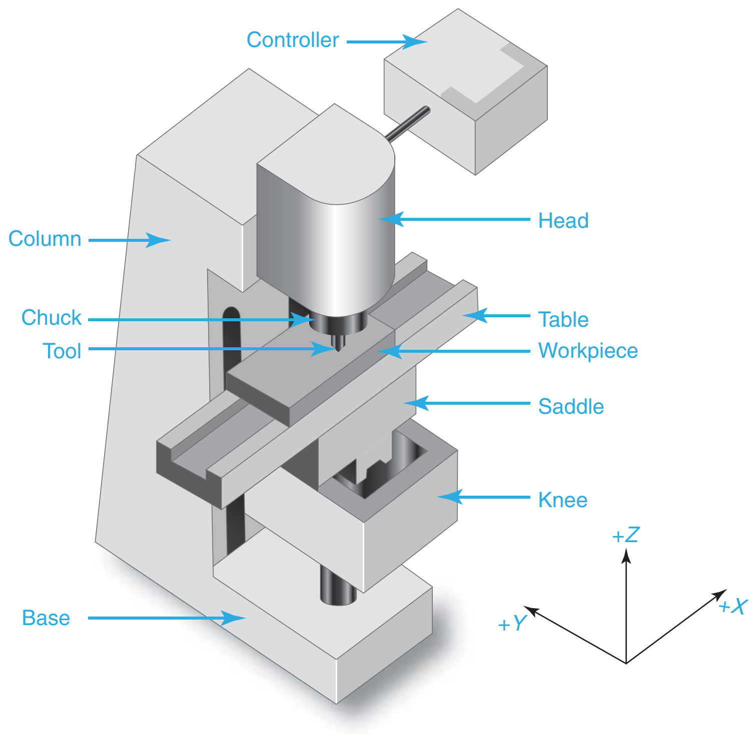


FIGURE 15.2
Schematics of machine tools

(B) Milling machine

machine frame. The machine also has a table that moves in two orthogonal directions controlled by two lead screws, one for each direction. Each machine has its own Cartesian coordinate system. The table of a milling machine moves in the X and Y directions shown in Figure 15.2B, whereas the table of a lathe moves in the X and Z directions shown in Figure 15.2A. The cutting tool always moves in the Z direction as shown. The positive Z direction is always pointing away from the workpiece by convention.

The workpiece is fixed securely to the machine table via jigs and fixtures that also ensure aligning the workpiece relative to the machine table and the cutting tool for correct machining. The lathe has a chuck (instead of jigs and fixtures for a milling machine) to hold the workpiece. If the machine is numerically controlled, it has an NC controller. Manual machines do not have controllers. The machine head is where the power is transferred from the machine motor to rotate the cutting tool at high cutting speed.

We describe the operations of the two machine tools shown in Figure 15.2. A turning machine (lathe) is typically used to machine axisymmetric cylindrical parts such as shafts and cylinders. Those are the revolves we create using CAD/CAM software. A milling machine can perform many operations including drilling holes, tapping holes, and milling part surfaces of any shape. Extrusions we create using CAD/CAM software are manufactured on milling machines.

The machining operation of a part begins by setting up the machine. The setup starts by mounting and securely fixing the workpiece (known as *stock*) on the machine table by the machine operator (machinist). Next, the machinist establishes the machine reference point (known as the *machine zero*) and the workpiece reference point (known as the *workpiece zero*). The machinist must align these two zeros to machine the workpiece correctly. When the machine setup is complete, the machinist turns on the machine power and the machine begins cutting (removing material from) the workpiece. If the machine is manual, the machinist moves the machine table manually. If it is an NC machine, the machinist loads the NC program (code) to the machine controller and starts the controller to execute the program.

The machining operation could be wet or dry. Each machine tool has a coolant system that cycles cutting fluid or coolant (oil or water-miscible fluid) (see Figure 15.1A). The machinist points the coolant nozzle to pour the coolant continuously on the cutting tool and the workpiece. The coolant is caught in a tub under the machine bed, gets filtered from the machining chips, and then is recycled again. Dry machining does not use coolant. The advantages of wet machining are twofold: it produces better surface finish than does dry machining, and it elongates the cutting tool life (does not get dull quickly). NC programming provides commands to control the coolant, that is, turn it on or off.

The skills of the machinist determine the accuracy and the quality of the finished parts. The machinist's skills are more crucial in manual machining than in NC machining. After machining is finished, the machinist removes the finished part. Further machining or processing on other machines may be required. After all machining is complete, the parts move to inspection and then are assembled into their intended products (assemblies).

15.3 Basics of Machining

Controlling the position and the motion of the cutting tool while it is removing (chipping) material (chips) away from a workpiece is essential and follows well-established concepts that include motion axes, cutting parameters, home position, toolpaths, and other topics that we cover here.

1. Motion axes: Motion axes of a machine tool determine the versatility of the machine and what type of surfaces (faces) it can cut. An **axis** defines a degree of freedom (DOF) along which a cutting tool can move. In 3D space, there is a maximum of six DOF: three translations along the X-, Y-, Z-axes and three rotations around these axes. However, a machine tool may have more than six DOF. There exist 2-axis, 3-axis, and multi-axis machine tools. A multi-axis machine tool has many DOF, sometimes up to 10.

   A 2-axis machine tool means that its cutting tool can move along two axes only and simultaneously. An example is a lathe in which the tool moves along the Z-axis (length of workpiece being turned) and the X-axis (into the workpiece to remove material), as shown in Figure 15.2A. In a 3-axis machine, the tool can move along three axes (*X*, *Y*, *Z*) simultaneously. A 3-axis milling machine provides an example. Most milling machines are 3-axis because you can machine any complex surface by controlling the tool motion along three axes. Additional DOF could be added either as rotations about *X*-, *Y*-, *Z*-axes or as translations of other parts of the machine (in addition to the tool translation). These additional DOF enable machining very complex parts as found typically in the aerospace industry. Milling machines with more than three DOF are typically referred to as machining centers. For example, a 4-axis machine has *X*, *Y*, and *Z*, and a fully rotating table that can rotate simultaneously while the tool is moving and cutting.

Some machine tools (specifically milling machines) are 2½-axis or 3½-axis machines. The ½ axis designation means that the motion along this axis is limited. For example, a 2½-axis milling machine means that its cutting tool can mill any shape (cross section) in a plane that is not parallel to the XY plane of the machine (if the plane is parallel to the table, we have a 2-axis machine tool). A 3½-axis machine has *X*, *Y*, and *Z* as its primary axes of motion, plus an indexing table designated as axis *A*. The indexing table is used for positioning; it cannot rotate simultaneously with the motion of the primary axes.

2. Cutting tools: Cutting tools are made of special materials and have special shapes. These materials must be harder than any material the tool is intended to cut; otherwise, the tool will not be able to cut it. The harder the cutting tool material, the more resistant it is to wear and thus the longer its life. Cutting tool materials include HSS (high speed steel), cobalt, carbide, titanium, and diamond. HSS is the softest material and diamond is the hardest material. The harder the material, the more expensive the cutting tool is, the heavier (more weight) it is, and the greater speed and feedrate it can run at.

   Another important characteristic of cutting tools is the number of flutes. A **flute** is defined as a helical groove in the tool along its length (shank) to remove chips (cut material) away from the tool cutting edges (teeth) as the tool rotates. As the tool cuts into the material, the chips need a path to get out. A tool has multiple flutes depending on how many cutting edges it has. Figure 15.3 shows 2-flute, 3-flute, 4-flute, and 6-flute cutting tools. An *n*-flute tool has *n* cutting teeth; each tooth has its flute.

   A third important characteristic of cutting tools is what their cutting ends look like. For example, a drill always has a conical end; this is why blind holes always have conical ends. There exist flat- or ball-end mills. Flat-end mill (Figure 15.3D, E, and F) performs flat surface milling whereas ball-end mill (Figure 15.3C) performs pocket milling. Counterbore and countersink tools are shaped as shown in Figure 15.3G and H. Counterbores and countersinks are used to create grooves to hide heads of bolts and screws and make them flush with intended part surfaces, usually for esthetic and safety purposes.

   Many types of cutting tools exist to support the varied cutting processes. For example, we have tools for turning, drilling, milling, and threading. Many tools exist for each process. Figure 15.3 shows some tools for some processes. Tools are usually designated by various parameters, depending on the tool. For example, drills and mills are designated by the tool diameter (not used here), number of flutes, and the end type; for example, 2-flute drill, 3-flute flat-end rough mill, or 3-flute ball-end mill. A tapping tool is used to make threads in a hole. Think of it as thread male. An example of tapping tool designation is 9/16 • 18NF (Figure 15.3J), referring to the thread outer diameter (9/16 inch); number of teeth per inch, or tpi (18); and the thread type (NF). Two thread types exist. One type is unified fine (UNF), which is referred to as NF (national fine) in the retail industry. The other type is unified coarse (UNC), which is known as NC (national coarse) in retail.

(A) One-point cutting-edge lathe carbide tool
(B) 2-flute drill
(C) 2-flute ball-end mill
(D) Rough 3-flute flat-end mill
(E) Rough 4-flute flat-end mill (cobalt)
(F) 4-flute flat-end mill
(G) Counterbore tool
(H) 6-flute countersink tool
(I) Reamer
(J) 9/16 · 18NF tapping cutter

FIGURE 15.3
Cutting tools

3. Stock: A **stock** is a large material from which workpieces can be cut. The standard stock shapes are shown in Figure 15.4. Companies typically buy stocks of different materials. When needed, they cut workpieces from stocks and machine them to make parts. The terms *stock* and *workpiece* are usually used interchangeably. Workpiece dimensions are always larger than the dimensions of the finished part to allow for machining down (removing material) the stock. Stock dimensions may be ⅛ to ¼ inch larger than those of a finished part. The rule is that we always select a stock shape and dimensions as close as we can to the finished part to minimize material waste and avoid longer machining time.

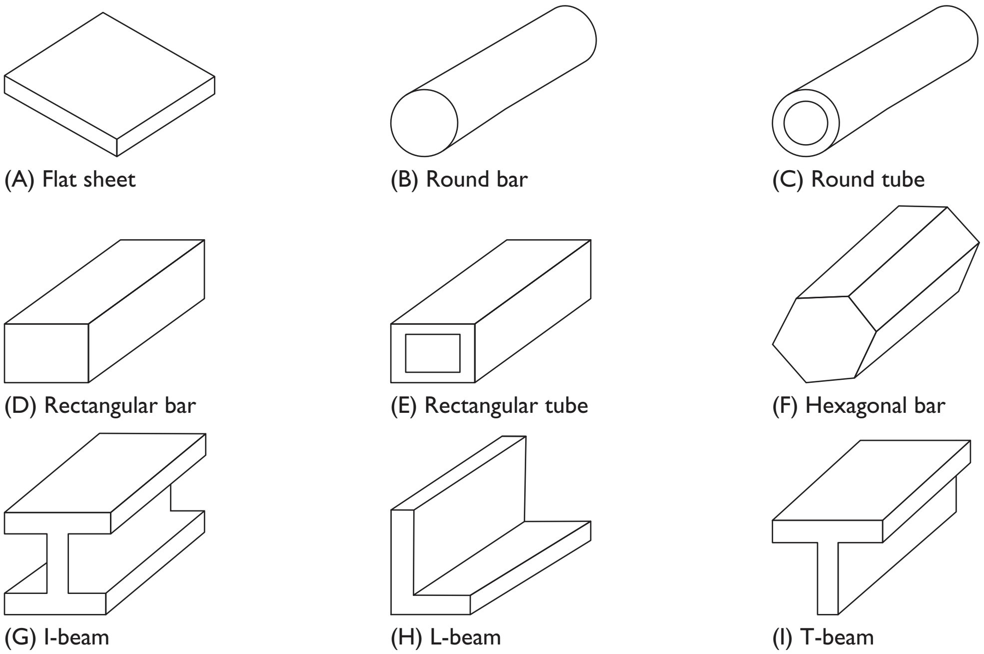


FIGURE 15.4
Stock shapes

4. Machining parameters: These parameters include the spindle (rotation) speed (also known as *cutting speed*), feedrate, and depth of cut, as shown in Figure 15.5. Notice that the workpiece rotates in place and the cutting tool moves in turning. The opposite is true for milling: The workpiece moves and the cutting tool rotates in place. A turning, drilling, milling, tapping, or other similar cutting tool is connected to the machine head via a chuck. The chuck grips the tool and keeps it securely in place. A tool rotates as it penetrates the workpiece to cut it. This rotation, known as the *spindle speed*, is measured in rpm (revolution per minute).

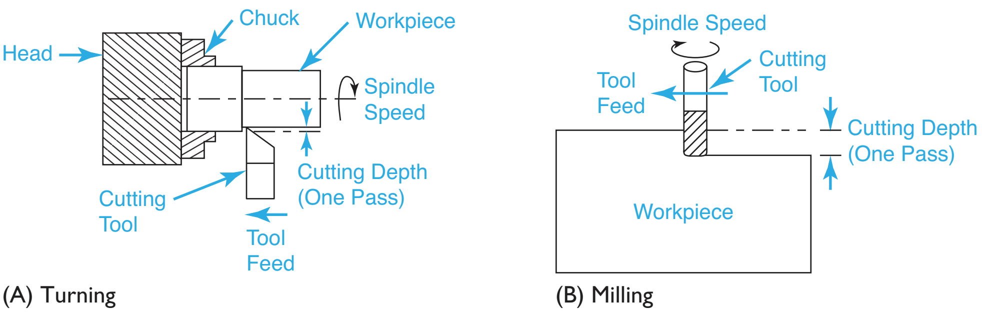


FIGURE 15.5
Machining parameters

The tool chuck is connected to the machine motor spindle that transmits the rotation from the machine motor to the tool.

The feedrate is how fast the cutting tool is removing material from the workpiece; that is, it is the linear speed of the cutting tool, measured in feet or inches per minute. Units are typically feet per minute (fpm). Fast feedrates result in rough surface finish. The value of a feedrate depends on the materials of both the workpiece and the cutting tool. Excessive feedrate or cutting depth could result in breaking the tool due to the extreme force exerted on the tool from the workpiece material.

The depth of cut (cutting depth) is the thickness of the material that the cutting tool removes from the workpiece as it travels in contact with the workpiece. This thickness represents one "pass" of the cutting tool over the workpiece. If we need to remove more material than the depth of the cut allows, we need multiple passes. Smaller depths result in a smoother surface finish.

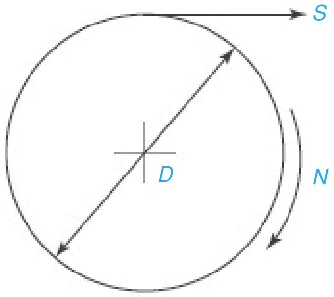


FIGURE 15.6
Cutter motion parameters

Most machining operations are conducted on machine tools that have rotating spindles. Thus, we relate the machining parameters to the spindle speed. The cutting tool rotates with the rotational speed, N, of the spindle and has a diameter, D. Thus, any point on the outer surface of the tool has a linear speed, S, known as the *cutting speed*. Figure 15.6 shows the view looking down at the cutting tool along its axis. We relate the spindle rotational (angular) speed and the tool surface (linear) speed as follows:

$$N = \frac{12S}{\pi D} \tag{15.1}$$

where N is in rpm (revolution per minute), D is in inches, and S is in feet per minute. Equation (15.1) is based on the simple equation of angular motion, $v = \omega r$. Sometimes we write S as SFM (surface feet per minute). Thus, Eq. (15.1) becomes:

$$N = \frac{12SFM}{\pi D} \tag{15.2}$$

In turning, D is the diameter of the workpiece being turned; in milling, drilling, reaming, and other operations that use a rotating tool, D is the cutter diameter.

The calculation of the feedrate requires knowing the tooth load during cutting and the number of teeth (flutes) of the tool. The load is the force that the workpiece exerts on the tool during cutting. The feedrate is given by the following equation:

$$F = F_t T N = F_{rev} N \tag{15.3}$$

where F is the feedrate (inches per minute), F_t is the tooth load (feed/tooth) in inches per tooth, T is the number of teeth, and N is the spindle speed (rpm). F_t is also known as the *chip load*, which is the size or amount (measured in inches per tooth, ipt) of chip that each tooth of the cutter can remove in one revolution. F is the distance that the cutting tool advances per minute. F_{rev} is the distance that the cutting tool advances during one revolution of the spindle measured in in./rev, and is given by:

$$F_{rev} = F_t T \tag{15.4}$$

F_{rev} is sometimes known as *cutting feed*. The time it takes to machine a length L of a workpiece is given by L/F in minutes. This is the time the tool needs to travel the distance L.

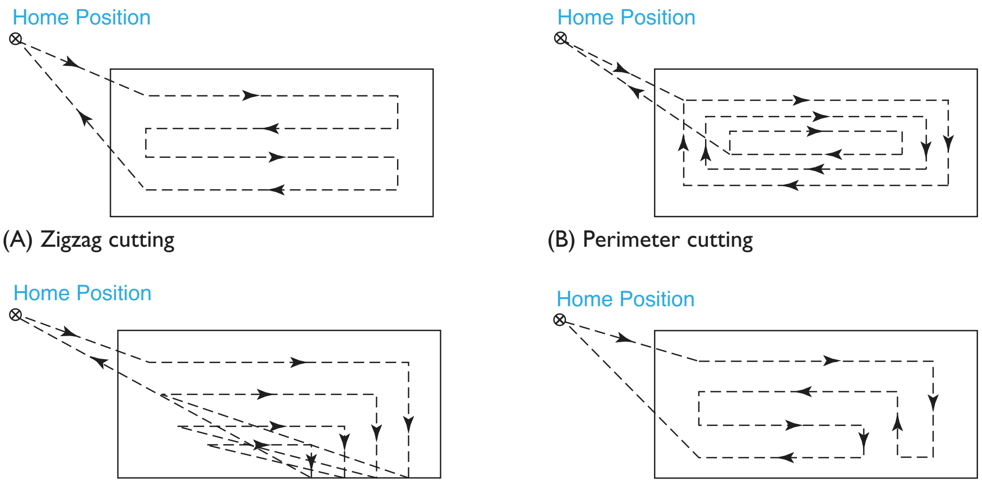
Toolmakers publish tables of spindle speed, SFM, and feedrate. Also, machinery handbooks publish similar tables. SFM tables are typically given for a tool diameter, D, of 1 inch. The most common number of tool teeth is 2, 3, or 4. Here are some example values of machining parameters. Use spindle speed and feedrate of 5000 rpm and 1 in./min, respectively, if we are roughing steel workpiece with carbide cutting tool, and 2000 rpm and 1 in./min if we are finishing it. If we are cutting aluminum with a carbide tool, we use 10,000 to 15,000 rpm for roughing and 5000 rpm for finishing, and 2 in./min for feedrate for both.

There are so many factors and variables that influence the selection of spindle speed and feedrate. These variables include materials and coatings of cutting tools, methods of holding the tools, different material properties of materials being cut, toolpath software, and cutting technique. Realizing these ever-changing values, NC machining software provides its users the ability to populate their cutting libraries and databases with new cutting values as they become available. However, NC software may provide what is called "out-of-the-box" values for machining parameters. These values are usually approximate. Interested readers should consult with their toolmakers and machine makers for the latest and most accurate values for machining parameters.

5. Machining quality: When we remove material away from a workpiece, we keep two goals in mind: removal speed and quality of surface finish. We need to cut (chip) away material as fast as possible while producing a high-quality surface finish. To achieve this compromise, we adopt a simple but effective strategy: roughing followed by finishing. In a roughing operation, we cut away large amounts of material, leaving a poor surface finish. In a finishing operation, we remove a smaller amount of material, leaving a good surface finish. There exist cutting tools for roughing and finishing. Figure 15.3D and Figure 15.3E show two roughing end mill cutters. Figure 15.3C and Figure 15.3F show two finishing mill cutters.
6. Squaring stock: A machinist typically cuts a piece of stock to start a machining operation, which then becomes the workpiece. Stocks are typically produced by low-quality manufacturing processes to keep their cost down. The machinist would need to prepare the stock before machining it to make the final part. Stock preparation is what we call *squaring*. Squaring means just that: machine the stock so that its faces are perfectly normal to each other, within the tolerance limits. Squaring begins by milling one side and then gripping the stock correctly with a vise to square the other sides, typically by placing the milled side on the machine bed; that is, making it horizontal in a vertical milling machine. Mill the opposite horizontal side of the stock. When done, ungrip the stock, rotate it, and repeat the squaring process to square another side. We continue until we finish squaring the stock's six sides, assuming the stock is a block.
7. Home position: The machinist needs to establish the machine zero before cutting. This is the home position of the cutting tool. The machine tool indexes this location and uses it to measure the tool coordinates during its movements. The home position could be a point on or near the workpiece. In an NC machine tool, the machine controller uses the home position to interpret the coordinates used in an NC program. Typically, the machinist sets the home position to align the tool axis with an edge of the workpiece.
8. Toolpath: A toolpath represents a machining strategy. Once we decide on a machining operation and a cutting tool, we need to decide on the best way to perform the cutting. By "best," we mean finding the shortest and fastest path to cut the workpiece. For example, Figure 15.7 shows four strategies and their toolpaths to mill the top face of a workpiece. Which toolpath offers the best strategy? Why? The selection of a toolpath depends on machining the part correctly to avoid damaging it or requiring repair. For our example, the toolpath that creates the least burr in the part is the best. A **burr** is a raised edge or small piece of material

remaining attached to the workpiece after machining. A burr must be removed via another machining process called *deburring*. Drilling burrs are common and happen in all materials. Deburring adds to and therefore increases the part machining cost.

Let us investigate the four toolpaths shown in Figure 15.7. The zigzag toolpath is the best and is typically used to mill a face because it produces the least burr. Also, the tool ends up at the end of the part all the time. The perimeter toolpath creates more burr all around the edges if you cut from the center of the face and out to the edges (called *in-out*). If the tool cuts its way in (called *out-in*) to reduce the burr, you create another problem because the tool leaves a spot (mark) at the center of the part because it has to spin there while not moving to mill the last area. The L and L-R-L (L Reverse L) toolpaths are a waste of time because we would waste time moving the tool to the beginning of the cut (Figure 15.7C), or we would need two moves to reverse the tool motion (Figure 15.7D). CAM software always defaults to the best toolpath to machine a feature. It also provides different types of toolpaths and allows the machinist to select a toolpath based on his or her experience.



(A) Zigzag cutting

(B) Perimeter cutting

(C) L cutting

(D) L-R-L cutting

FIGURE 15.7
Toolpaths to mill a face

Toolpaths come in two types depending on the machining operation: PTP (also called *positioning*) or continuous (also called *contouring*). In PTP (point-to-point) machining, the tool is not in constant contact with the workpiece during its motion. Drilling is an example of PTP machining. The cutting tool is in contact with the workpiece only momentarily to drill the hole at the desired point (location). The tool does not contact the workpiece during its travel from one point to another. Figure 15.8A shows a PTP toolpath to drill holes. In continuous path machining, the tool is in constant contact with the workpiece, as in the case of face or side (contour) milling, as shown in Figure 15.8B.

While the tool is moving without being in contact with the workpiece, how high should it "fly" above the workpiece top face to clear it? In practice, machinists typically use 0.1 inch (read as 100 thousandths of an inch). Some may use half this height (i.e., 50 thousandths of an inch). We need smaller heights to save time in moving the tool toward and away from the workpiece.

The coordinates of key points on a toolpath are measured with respect to the part coordinate system, which is typically the MCS of the part CAD model. CAM

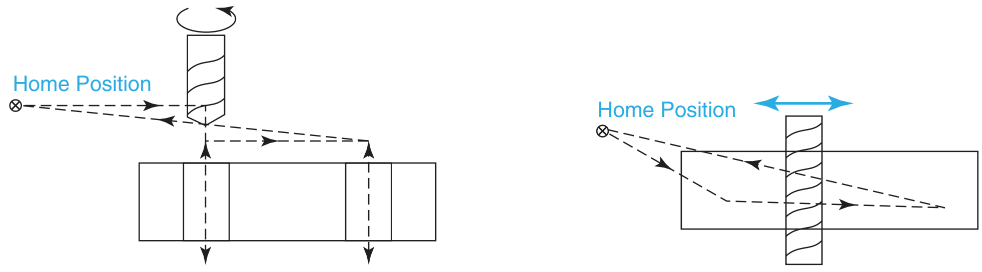


FIGURE 15.8
Types of toolpaths

software generates the toolpath using the MCS coordinates. The correct machining of the part requires the machinist to align the workpiece (part) coordinate system with that of the machine. Also, the machinist should align the zero (origin) of the workpiece (MCS) with the machine zero. Without the correct alignment between the workpiece and machine coordinate systems, the machine controller interprets the toolpath incorrectly, rendering the workpiece as scrap.

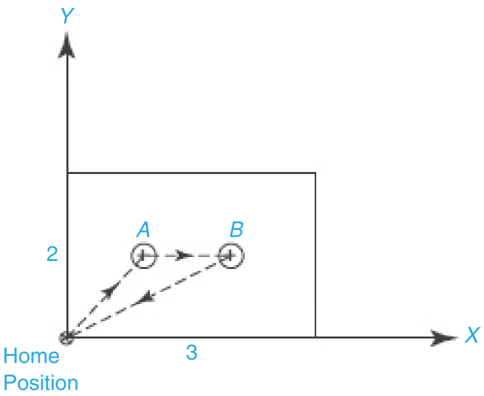


FIGURE 15.9
Absolute and relative coordinate measurements

The coordinates of a point on the toolpath may be expressed in absolute or relative coordinates. In the absolute system, the coordinates are always measured with respect to the machine zero (origin of the MCS). In the relative system, the coordinates are measured with respect to the last location of the tool. For example, consider Figure 15.9. Assuming that the two holes, *A* and *B*, are centered in the part, the absolute coordinates of their centers are (1, 1) and (2, 1), respectively. The relative coordinates of center *B* relative to *A* are (1, 0), and of center *A* relative to *B* are (–1, 0). In reaching these values, we move the XY coordinate system shown and attach it to the reference point. We prefer the absolute coordinate system in practice. The major drawback of the relative coordinate system is that any toolpath point change renders all the coordinates wrong.

9. Rapid positioning: We can move the cutting tool at two speeds: rapid and slow. When the tool is not in contact with the workpiece while moving, we use rapid motion (also known as *rapid positioning*) because we do not have to worry about the tool breaking or impacting the workpiece surface finish. For example, we use rapid positioning to move a drill from one hole location to another, or to send a milling cutter from home position to a workpiece or vice versa.

15.4 Turning

We provide brief descriptions of some basic machining processes (operations): turning, drilling, milling, and EDM. We start with turning, which is done on a lathe (Figures 15.1A and 15.2A). Turning is used for producing axisymmetric (e.g., cylindrical or conical) parts, the revolves we create on a CAD/CAM system. The workpiece is rotated while the cutting tool moves parallel to the axis of rotation of the workpiece, as shown in Figure 15.2A. Although turning is the machining operation most widely done on a lathe, a lathe may also be used for drilling, boring, tapping, facing, threading, polishing, grooving, knurling, and trepanning. Each operation uses a different-shaped cutting tool. A workpiece may be held in a 3-, 4-, or 6-jaw chuck, with collets, or it may also be held between centers. A single-point tool (Figure 15.3A) is used for turning.

15.5 Drilling

Drilling is a machining operation that creates cylindrical holes in parts. Drilled blind holes have a conical end, the same shape as the drill cutter (see Figure 15.3B). When we drill through (non-blind) holes, we make sure to feed the drill bit (tool) enough so that the drill conical end clears the bottom face of the workpiece to ensure the correct size of the hole diameter throughout the entire hole length. This would require gripping the workpiece appropriately with the jigs and fixtures to ensure that the drill end would not hit any of the holding faces.

A drilling operation could be done on a turning or milling machine, depending on the shape of the workpiece. For example, holes in a flange are drilled on a lathe whereas holes in a block are drilled on a milling machine. Sometimes, we may spot a hole before drilling it. This means that we use a "spotting drill" bit to make a dent in the workpiece first before drilling to locate (establish) the hole center. Other times, we use a "center drill" bit to make a conical starting indentation for a larger-sized drill bit used in a lathe drilling operation.

In some cases, drilling provides the base operation for other machining operations to follow. For example, we may use a reaming operation after drilling if we need a very smooth surface finish for the hole. Figure 15.3I shows a reamer. Or, we may use a tapping operation to create a threaded hole. Figure 15.3J shows a tapping cutter. Or, we may add counterbores or countersinks to holes. Figure 15.3G and Figure 15.3H show the corresponding cutting tools. Counterbores and countersinks are used to hide a bolt head or a nut. We use them either to flush the bolt head or nut with the hole top or bottom face, or to hide them a little below the face. This hiding is done for either aesthetic or safety reasons.

15.6 Milling

Milling is the most common and versatile machining operation. It is done on a milling machine or center, and allows us to machine almost any shape imaginable. Milling machines may be horizontal or vertical (Figure 15.1B and 15.2B); vertical milling machines are more popular than horizontal ones.

Milling machines are typically 3- or 3½-axis machines. A 3-axis milling machine provides 3 translations on the cutting tool simultaneously (see Figure 15.2B). A 3½-axis provides an indexing axis of the machine turret. A **turret** is a tool holder (changer) that holds multiple tools and changes them automatically between operations. The advantage of a turret is to eliminate changing the cutting tool manually between operations, thus saving tool-changing time. Milling centers can have more axes, providing rotational DOF and/or auxiliary translational DOF. Some vertical milling centers have flexible heads. This means that a machine head can be tilted from side to side and from front to back to allow for machining flexibility.

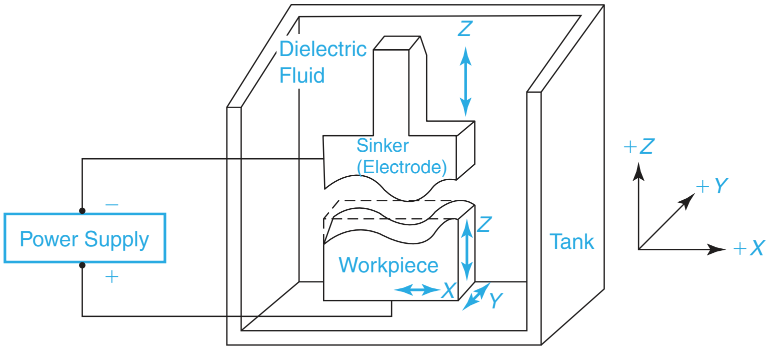
A variety of milling operations and tools exists. Examples include slot, face, pocket, and side (contour) milling. In slot milling, the cutter is a flat-end mill that cuts on its end face and periphery (sides). In face milling, the tool cuts with its end face. Milling a pocket is similar to milling a slot, but the tool performs multiple passes. In side milling, the tool cuts with its periphery and goes around the perimeter of the workpiece.

15.7 Electrical Discharge Machining

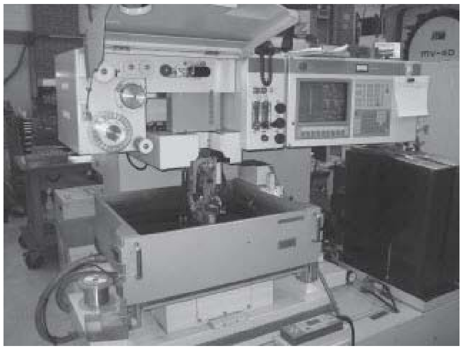
Electrical discharge machining (EDM) is a process that removes material from a workpiece via an electric spark between two conducting surfaces. (EDM removes material via spark erosion.) One surface is an electrode and the other is the workpiece itself. The workpiece material must be conductive to electricity. There are two types of EDM: sinker (plunge) and wire. Figure 15.10 shows the two types. What distinguishes the

FIGURE 15.10
EDM machines

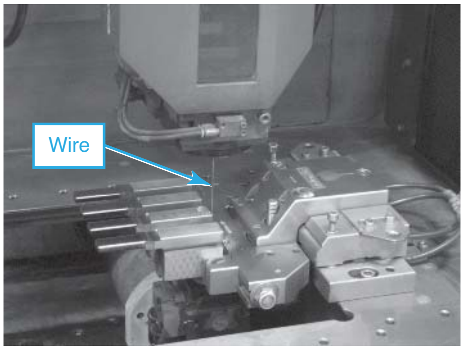
(A) Sinker EDM (Courtesy of NyproMold Inc.)



(B) Schematic of a sinker EDM machine

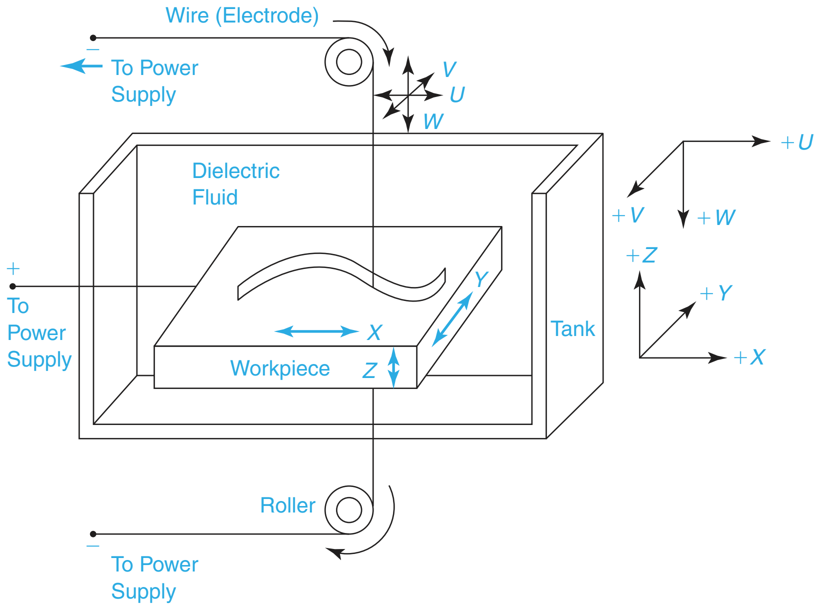


Idle wire EDM machine



Wire EDM machine in action (creating holes)

(C) Wire EDM (Courtesy of NyproMold Inc.)



(D) Schematic of a wire EDM machine

Part of a mold

A cam (a mold ejector component)

(E) Sample wire EDM parts (Courtesy of NyproMold Inc.)

two types is the shape of the electrode. If the electrode is a block, we have sinker EDM, and if it is a wire, we have wire EDM. Sinker EDM is used to make mold (die) cavities and stamping dies. Wire EDM is primarily used for through hole machining; that is, to make through holes of any shape as shown in Figure 15.10D. Depending on the piece we want to keep after machining, we can have a hole or the cutout piece as the final product.

Figure 15.10B shows a schematic of a sinker EDM machine. The electrode is made from copper or graphite and is shaped as the mold or the stamp we need to create. The sinker moves up and down as a milling cutting tool. The workpiece is submerged in a tank containing dielectric (does not conduct electricity) fluid, usually oil or synthetic fluid. The sinker moves up and down only, similar to other cutting tools. The workpiece moves along the three axes as shown. When the sinker gets close to the workpiece, an electric spark is generated. The spark is strong enough to melt (erode) the workpiece to cut it to shape.

The dielectric fluid is continuously flowing in the tank. The molten particles from the spark drop in the dielectric fluid, which sweeps them away to be filtered out. The dielectric fluid is recycled after filtering. In addition to flushing the material away, the fluid serves as a coolant to minimize the heat-affected zone, thus preventing potential damage to the workpiece.

The spark is generated at a controlled frequency. Spark frequency is defined as the number of times per second that the electric current is switched on and off. Low frequency is used for roughing the workpiece, and high frequency is used for finishing. The material removal is done only during the on time. The longer the on time, the more material is removed in each sparking cycle. Roughing operations use extended on time for high material removal rates. The resulting craters are broader and deeper.

Figure 15.10D shows a schematic of a wire EDM machine. The operating principles of a wire EDM are almost the same as those of the sinker EDM. The main difference is the electrode shape. The wire material could be brass, copper, tungsten, or zinc, which all have excellent electric and thermal conductivity. The wire diameter ranges from 0.004 to 0.012 in., in increments of 0.002 in. The wire moves over the roller like a belt and goes through the workpiece. Unlike the sinker electrode, the wire can move in three directions, as shown in Figure 15.10D. However, the wire motion in the W direction (along its length) is unidirectional. The workpiece also moves in three directions. The wire cuts the workpiece and is used only once to ensure accuracy and precise cylindrical cuts. Wire EDM machines use deionized water as the dielectric fluid.

Wire EDM is used primarily for shapes cut through a part or assembly. If a cutout needs to be created, an initial hole must be drilled first (using EDM also). Then, the wire can be fed through the hole to complete the machining. Keep in mind that the wire (negative charge) must never touch the workpiece (positive charge); otherwise, we create a short circuit. It is the spark that goes from the wire to the workpiece and cuts it.

EDM produces parts with high accuracy (tight tolerances of ± 5 microns; 1 micron = 1 millionth of a meter) and quality, and it can cut any material that is conductive. EDM has many advantages over conventional machining. It is great to use to produce one or a few of a complex part (low-volume production) where traditional machining may be expensive. Wire EDM produces a burr-free, superior edge finish. It is also useful for custom parts such as jewelry or human implants. Wire EDM can cut diamonds or make orthopedic implants, aerospace and automotive parts, extrusion dies for rubber and plastics, and gears. Cutting with a wire allows us to cut any complex shapes and intricate contours. Here the wire is the tool. The wire can take any shape because it is flexible to bend and twist along its guides.

EDM can cut through a stack of workpieces (sheet materials), thus cutting several parts at the same time. EDM can also be used to create prototypes, similar to rapid prototyping (e.g., trial stamping of parts). EDM can cut parts weighing up to 10,000 pounds.

15.8 Manufacturing of Design

We now have a good background about manufacturing. We see how our designs are manufactured and produced. It is logical to ask this question: How can we make our designs better from a manufacturing point of view? This question addresses a broader issue. That is, designers should not only concern themselves with functional requirements and force analysis, but they should also think broader and consider their design implications on the entire product life cycle from design (product birth), to manufacturing, to assembly, to disassembly, to disposal (product death). We have covered design sustainability already in this book. Many design foci (concepts) exist to address product life cycle concerns. These include DFA (design for assembly), DFM (design for manufacturing), DFX (design for anything), concurrent engineering, PDM (product data management), and PLM (product lifecycle management).

We cannot explain each of these design foci in depth in this book, but we offer a quick overview here. Interested readers should consult other books and resources for more in-depth coverage. All these foci try to change product design during the design phase where design changes are simply a change on paper or in CAD software. Changes at the design phase are not detrimental or costly as are changes during manufacturing on the production floor.

The focus of DFA is assembly; change the design to make it easy to assemble. For example, reduce the number of parts of an assembly as much as possible to make it quicker to assemble. Having fewer parts per assembly also reduces its maintenance and repair cost. The focus of DFM is manufacturing. DFM tries to address this question: How can we change the design to make it easier and less expensive to make? For example, do not use blind holes that have flat bottoms. The "X" in DFX extends the concepts of DFA and DFM to any design concern we may have and should address.

Each of these design philosophies (DFA, DFM, DFX) has a narrow focus. As a matter of fact, these philosophies may and often do produce conflicting results. Thus, we use more encompassing approaches. Concurrent engineering approach, for example, suggests that all the teams (design, materials, manufacturing, marketing, sales, and others) concerned with a new product should meet and work concurrently during the product design phase to resolve any disagreements and conflicts so that the final design is acceptable to all, including customers (users of the product).

PDM and PLM are similar to concurrent engineering, but from a management and software point of view. PDM and PLM software are extensions of CAD/CAM software. PDM concerns itself with managing product design activities only. PLM goes beyond design and looks at disassembly and product end of life. SolidWorks offers **DFMXpress** for DFM and **PDMWorks** for PDM.

15.9 SolidWorks DFMXpress

SolidWorks implements the concept of DFM in **DFMXpress**. **DFMXpress** validates the manufacturability of SolidWorks parts. It identifies design areas where manufacturing may have a problem or production cost may be excessive. SolidWorks uses three DFM categories: rule description, configuration rules, and validating parts. Rule description provides rules for turning, drilling, milling, sheet metal, and injection molding as well as standard hole sizes.

Turning rules discuss corner radii and reliefs as follows:

- ☐ When designing stepped shafts, have large enough diameters at the changeover edges to allow for using a tool with large nose radius.
- ☐ Provide tool relief for the bottoms of blind bored holes.

Drilling rules discuss hole diameters, holes with flat bottoms, hole entry and exit surfaces, holes intersecting cavities, partial holes, and linear and angular tolerances. Here are the drilling rules:

- ☐ Avoid holes with small diameters (< 3 mm) or large length-to-diameter ratio (> 2.75) because these holes are difficult to machine. For example, deep holes make chip removal difficult, especially if the hole is blind.
- ☐ Avoid flat-bottom holes. Use conical bottoms with angles that conform to standard drills. Flat-bottom holes have to be milled and/or reamed, two expensive operations.
- ☐ Entry and exit surfaces of holes should be perpendicular to the hole axis to minimize the shear forces on the drill bit causing it to wander or break on impact when it meets the surface.
- ☐ If a hole must intersect a cavity, the drill axis should be outside the cavity to minimize the impact of contact.
- ☐ Holes drilled on edges of features should have 75% of the hole area within the feature material.
- ☐ Tolerances should not be tighter than necessary.

Milling rules discuss deep pockets and slots, inaccessible features, sharp internal corners, and fillets on outside edges. Here are the milling rules:

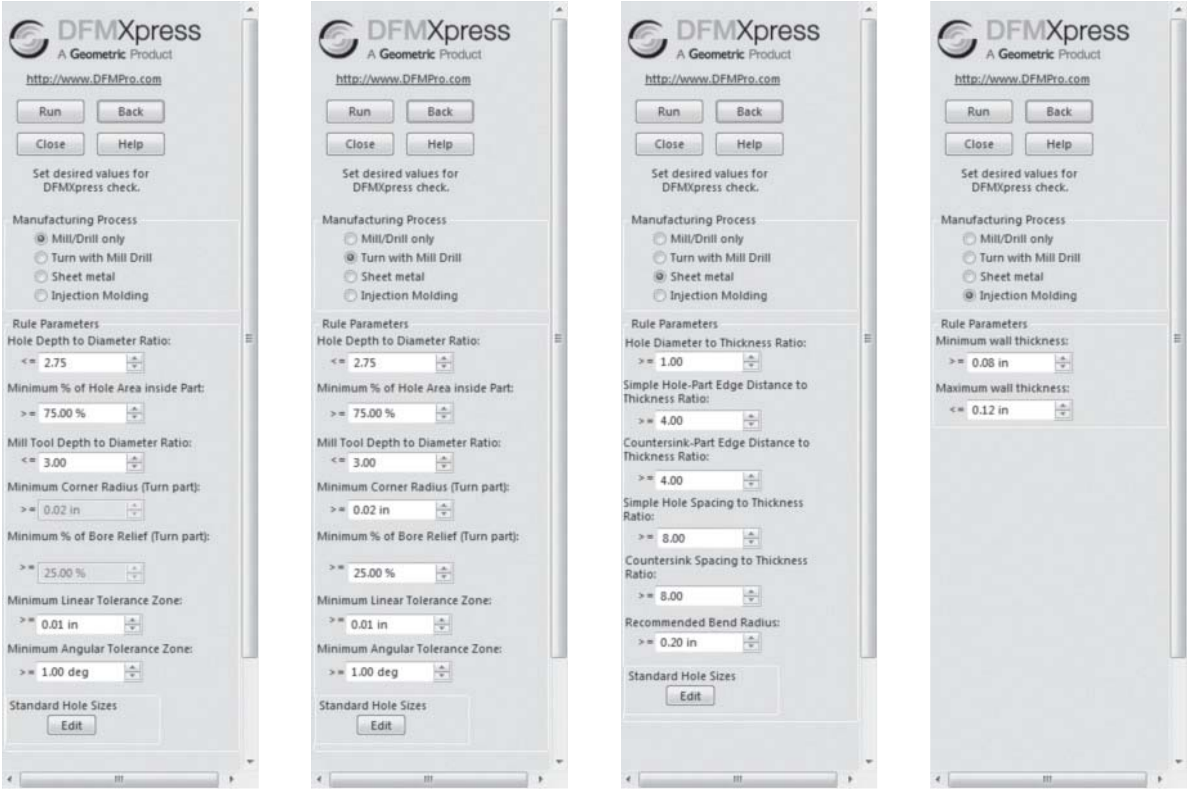
- ☐ Avoid narrow slots because they are difficult to machine. Long, slender end mills required to machine them cannot meet the tolerance requirements because the tools are prone to chatter. Deep slots also make chip removal difficult.
- ☐ Avoid long corners with long radii.
- ☐ Design milled areas so that the end mill length-to-diameter ratio is ≤3.
- ☐ Avoid inaccessible features because they require special tools and machining techniques.
- ☐ Avoid sharp inside corners. For example, a pocket with sharp corners cannot be milled. We have to use EDM to cut it. A three-edge inside corner must have one of its corners with a radius equal to the radius of the end mill. If a sharp corner cannot be avoided, first drill a relief hole at the corner and then mill.
- ☐ Always chamfer outside edges instead of filleting them. An outside fillet is expensive to machine because it requires a form-relieved cutter and a precise setup. Also, blending fillets into existing surfaces is expensive, even with a ball-end mill.

Sheet metal rules discuss hole diameters, hole-to-edge distances, hole spacing, and bend radii. Here are the rules:

- ☐ Avoid designing parts with very small holes. Small drill bits can break easily.
- ☐ Have holes far enough from an edge or bend to avoid distorting the edge.
- ☐ Space holes well apart so the material between them does not become weak and get distorted.
- ☐ Do not use excessive bend; otherwise, the material might crack. A rule of thumb is to have the bend radius larger than the material thickness.

The standard hole sizes rule is simple to follow. Hole sizes should be based on the sizes of standard drill bits.

Injection molding rules is only one rule specifying the minimum and maximum wall thickness as we discuss in Chapter 16.



(A) Milling/drilling rules (B) Turning rules (C) Sheet metal rules (D) Injection molding rules

FIGURE 15.11
DFMXpress rules

DFMXpress implements all these rules and helps the designer use them. To find all these rules, click this sequence: **Tools > DFMXpress** > select a manufacturing process to access its rules as discussed above. Also, **DFMXpress** can validate these rules for a given part design. After selecting a process from the above sequence, click **Run** and investigate the results. Figure 15.11 shows the **DFMXpress** interface. Figure 15.12 shows the standard hole sizes. To access these sizes, click the **Edit** button at the bottom of the **DFMXpress** pane.

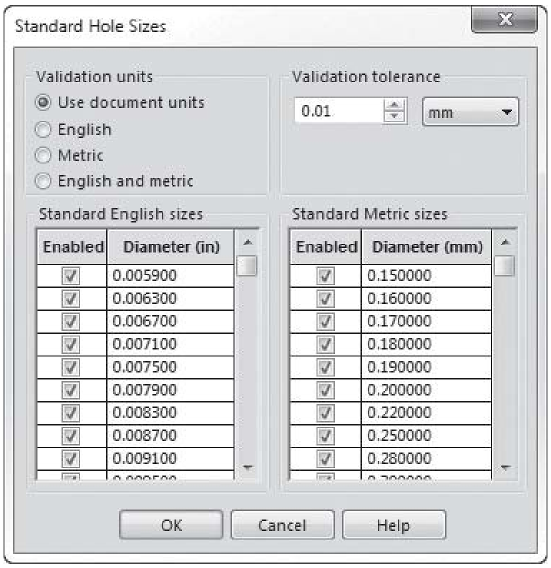


FIGURE 15.12
Standard hole sizes

Example 15.1 Figure 15.13 shows a part. Check the part manufacturability. All dimensions are in inches.

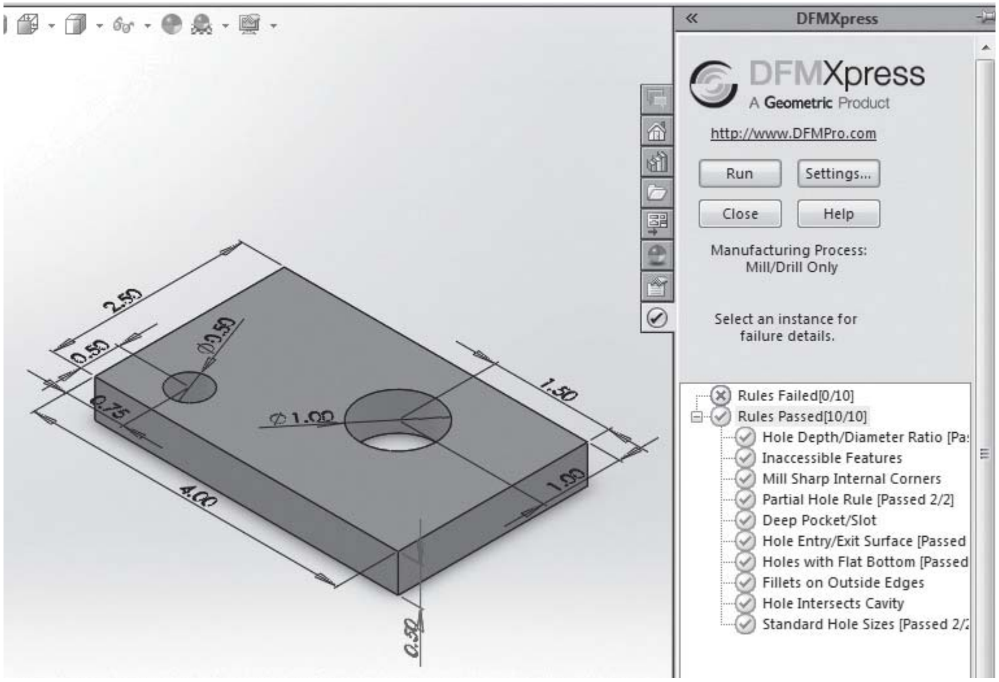


FIGURE 15.13
Check part manufacturability

Solution While the part is open, click **Tools > DFMXpress > Mill/Drill only > Run**. The results shown to the right of Figure 15.13 indicate that the part passed DFM analysis.

Note: Turn on task pain first (**View > Task Pane**)

HANDS-ON FOR EXAMPLE 15.1. Edit the part to move the small hole very close to the left bottom corner. Does the part pass DFM analysis? What errors do you get?

15.10 Basics of NC Machining

The basics of machining we have covered earlier in this chapter apply here. This section extends these concepts with additional ones that apply to NC machining, commonly known as *NC programming*. NC programming applies to NC machine tools. Each one of these machines has an NC controller, as shown in Figure 15.2. The controller reads an NC program, executes it, and uses it to control the motion of both the machine table and the cutting tool. NC machining is more accurate and faster than manual machining.

Machinists or designers use CAD models and NC software to write NC programs. NC software could be a software module in CAD/CAM software or completely separate software. For example, Mastercam and CAMWorks are **Add-Ins** to SolidWorks.

NC programs are written using G-code and M-code. NC controllers read G-code and M-code instructions. While G-code and M-code are ANSI/EIA standard language, some post-processing may be needed to accommodate proprietary features and different commercial interpretations of the code. The benefit of standard programming language is easy transfer of NC programs from one machine controller to another.

An **NC program** is a list of instructions (commands or statements) that orders an NC controller to carry out a sequence of operations on the machine tool to achieve a given machining process. After an NC program is written, it is downloaded to the machine controller. While we can write NC programs directly on NC controllers, this is not a common practice. It would be very cumbersome and difficult to program the machining of an intricate part this way. Instead, we use CAD/CAM software to generate the NC toolpath, verify it, edit it, and finally generate the NC program that can be downloaded to the NC controller via a network or a storage device such as a USB. **Toolpath verification** is a visual simulation of the cutting tool traversing the toolpath. The NC programmer may spot errors and correct them before real manufacturing, thus saving money and time.

We cover G-code and M-code programming. Before we can write NC programs, we offer the following NC programming concepts:

1. Zero-radius programming: The definition of a toolpath geometry requires the (x, y, z) coordinates of its key points. Knowing that the cutting tool follows the toolpath, should the coordinates somehow reflect the size (diameter) of the tool? No. When we write NC programs, we assume the diameter of the tool to be zero. Such an NC programming approach is known as *zero-radius programming*, or programming the part. We specify the diameter of the tool at the beginning of the program, and the NC controller compensates for the diameter while executing the NC program. The advantage of this approach is that we can use the same NC program with multiple tools of different diameters.
2. Tool offset: We can use tool offset to cut more than one part using the same program in one setup. Setup cost before machining usually increases part manufacturing cost. Consider drilling a hole in a part. Why could we not put multiple parts in one or two rows on the machine table all at once and set them up? Then we apply an *X*-, *Y*-, or *X*- and *Y*-offset to the NC program to move the table to the right location (under the drill bit) to drill the hole. This increases machining productivity and reduces machining cost by reducing the number of setups, which in turn reduces the setup cost.
3. Driving the tool: During actual machining, the tool does not move (except small movements in the vertical, depth of cut, direction); the workpiece does move because it is fixed to the machine table that moves in the X, Y, and Z directions. However, in NC programming, we visualize the tool motion (tool moving around the workpiece), not the machine motion. We move the tool and hold the workpiece fixed during writing or generating NC programs. It is easier this way. The NC controller reverses the motion during executing the NC programs.
4. Interpolations and canned cycles: NC controllers provide linear and circular interpolations to machine linear and circular profiles. For example, we define a line by two endpoints and a circle by a center and radius. From there, the NC controller knows how to cut these profiles in small linear increments (movements of the machine table). The controller may also have other canned cycles such as tapping holes. A **canned cycle** is a prewritten function that the NC programmer can call in an NC program.
5. CL data: The cutter location (CL) data define the data of the tool centerline as described by the G-code and M-code. CL data files are G-code and M-code files.

15.11 G-Code and M-Code Programming

Writing an NC program requires the context of how we perform a machining operation. First, we need to define the toolpath, and then move the tool along this path. Accordingly, an NC program defines the toolpath geometry and instructs the tool to move along it. Other machining instructions may be needed such as rapid positioning, turning coolant on and off, and others.

An **NC program** consists of a sequence of instructions (statements), one per line. In the G-code and M-code programming, a statement is known as a *block*. Each block (line) begins with a block (line) number (N) followed by a code word (e.g., G00). Each code word begins with a letter followed by numerical digits as shown. Following the code word (*code* or *word* is also used) is the required data. A block may contain more than one word. An NC controller executes one block at a time. Here is an example:

```
N01 G00 X1.0 Y0.0 Z0.0
N02 M13
N03 G01 X1.0 Y0.0 Z-1.0
```

The first block is Number 1 (N01) and uses the G00 code followed by (1, 0, 0) coordinates of a point, say P_1. The G00 code is rapid positioning code. It instructs the NC controller to move the cutting tool from its current position rapidly to P_1 in preparation for the next instruction (block). The M13 code in the second block (N02) instructs the NC controller to turn on the machine spindle rotation in the clockwise direction and also turn on the coolant in preparation for drilling. Finally, the last block (N03) performs the drilling operation by moving the drill bit a -1 unit in the Z direction using the G01 code.

Prior to executing the above NC program, the machinist turns the machine power on and prepares for machining. Then, the machinist clamps the workpiece to the machine table, and lines up the machine and workpiece zeros with the zero used in the NC program. Finally, the machinist loads the drill bit into the NC machine chuck and runs (executes) the program. After drilling, the machinist stops the spindle rotation, clears the workpiece from the spindle area, and unclamps and removes the workpiece, which is the finished part. The part may be inspected later to ensure that it is within the specified tolerances.

There is a finite set of codes that is available for G-code and M-code programming. The codes use all the alphabet letters, from A to Z, as follows. A to E codes define angular dimensions. F is the feed code. G is the preparatory (geometry) code. H defines tool length offset. L specifies fixed cycle loop count. I, J, and K are used for threading. M is the miscellaneous code. N is the sequence (number) code. O is the secondary sequence number. P, Q, and R are used for rapid-traverse dimensions. S is the spindle speed code. T is the tool code. U, V, and W are the codes for secondary axes of motion of the machine table. X, Y, and Z are the codes for the primary axes of motion of the machine table.

Table 15.1 shows a subset of the available G-codes. Interested readers should consult other resources for more codes. Some G-codes not listed here mean different things to different NC controllers. This is why we always need to post-process an NC program generated on a CAD/CAM to a particular NC controller to ensure its correct execution.

Table 15.2 shows a subset of the available M-codes. Interested readers should consult other resources for more codes. Some M-codes not listed here mean different things to different NC controllers.

TABLE 15.1 Subset of the Available G-Codes			
Code	**Data**	**Description**	**Example**
O	Number	Program number	O0001
G00	*x*, *y*, *z*	Tool rapid position to the point specified by data	G00 X1.0 Y0.0 Z0.0
G001	*x*, *y*, *z*	Move tool linearly to point specified by data	G01 X2.0 Y0.0 Z0.0
G02	(x_c, y_c), *R*	Circular interpolation clockwise (CW)	G02 X4.0 Y3.0 R1.5
G03		Circular interpolation counterclockwise (CCW)	G03 X4.0 Y3.0 R1.5
G04	Number in seconds or milliseconds	Dwell is an intentional time delay where the cutting tool remains rotating and in contact with the workpiece. Use dwell to stop the NC program execution to remove chips, improve machining quality, etc.	G X2.5 in seconds, ORG P2500 in milliseconds **Help:** X or P specifies sec or millisec. P cannot take decimals.
G10	*x*, *y*, *z*	Tool offset distance in one or all *X*-, *Y*-, *Z*-axes	G10 X1.0 Y2.5 Z0.0
G17	None	Select XY plane of machine tool	G17
G18	None	Select XZ plane of machine tool	G18
G19	None	Select YZ plane of machine tool	G19
G20	None	Input in inches (inch mode)	G20
G21	None	Input in mm (mm mode)	G21
G28	*x*, *y*, *z*	Return to reference point specified by data	G28 X3.0 Y2.0 Z0.5
G90	None	Absolute programming	G90
G91	None	Incremental programming	G91
G92	*x*, *y*, *z*	Set program zero to point specified by data	G92 X0.0 Y1.0 Z3.5
G94	F code	Feedrate (in./min or mm/min)	G94 F3.75
G95	F code	Feed per revolution (in./rev or mm/rev)	G95 F0.003

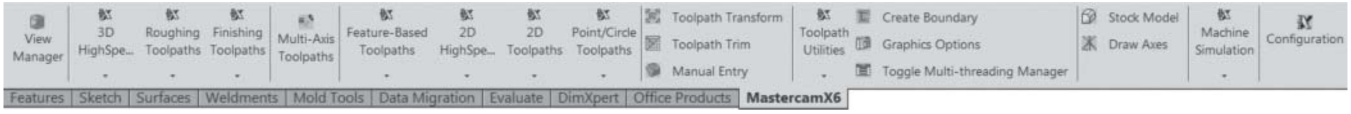
TABLE 15.2 Subset of the Available M-Codes			
Code	**Data**	**Description**	**Example**
M00	None	Unconditional automatic stop the machine. Machinist must push a button to continue with the remainder of the program	M00
M01	None	Conditional (optional) stop. Triggered only if the machinist pushes any button on the NC controller	M01
M02	None	End of program. NC machine stops	M02
M03	None	Start spindle rotation in forward (CW) direction	M03
M04	None	Start spindle rotation in reverse (CCW) direction	M04
M05	None	Stop spindle, i.e., spindle off	M05
M06	None	Tool change	M06
M07	None	Turn on coolant in mist mode	M07
M08	None	Turn on coolant in flood mode	M08
M09	None	Turn off coolant	M09
M10	None	Automatic clamping of workpiece, fixtures, spindle	M10
M11	None	Automatic unclamping of workpiece, fixtures, spindle	M11
M13	None	Start spindle rotation in forward (CW) direction and turn coolant on at the same time	M13
M14	None	Start spindle rotation in reverse (CCW) direction and turn coolant on at the same time	M14
M19	None	Oriented spindle stop. Spindle stops at predetermined angle	M19

15.12 CAM Add-Ins Software

There are many CAM software packages as there are many CAD/CAM packages. We cover two of the systems that SolidWorks uses as **Add-Ins**: CAMWorks 2014 and Mastercam X7. We need to install the software first. Once installed, we can access it within SolidWorks. We also need to be aware these two packages run in standalone mode. We need to install the version that integrates with SolidWorks. Figure 15.14 shows the toolbar for each package. CAMWorks has two tabs. The functionality we need resides on both tabs. We use both packages in the tutorials. The reader can use the tutorials that match his or her local CAM installation.

(A) CAMWorks tab1

(B) CAMWorks tab2



(C) Mastercam tab

FIGURE 15.14
CAM software user interface

Follow the following steps to generate toolpaths in CAMWorks:

1. Create the part in SolidWorks
2. Define the machine and the controller
3. Define the stock (shape and size)
4. Define (extract) the machinable features
5. Generate the operation plan and adjust the machining parameters if needed
6. Generate the toolpath
7. Verify the toolpath
8. Post-process the toolpath to generate the CL (G-code and M-code) file

Follow the following steps to generate toolpaths in Mastercam:

1. Create the part in SolidWorks
2. Define the stock (shape and size)
3. Define the machine and the controller
4. Define the machining operation and parameters
5. Generate the toolpath
6. Verify the toolpath
7. Post-process the toolpath to generate the CL (G-code and M-code) file

15.13 Tutorials

The theme for the tutorials in this chapter is to practice using the manufacturing and NC programming concepts by using both CAMWorks and Mastercam. Each tutorial is done twice: once using CAMWorks and once using Mastercam. Readers can use either one depending on the CAM package available to them.

Tutorial 15–1: Turn a Stepped Shaft

We generate the toolpath and CL file to turn the stepped shaft shown in Figure 15.15. All dimensions are in inches. The cutting parameters are spindle speed = 1500 rpm and feedrate = 2.50 in./min. Coolant is on for wet cutting.

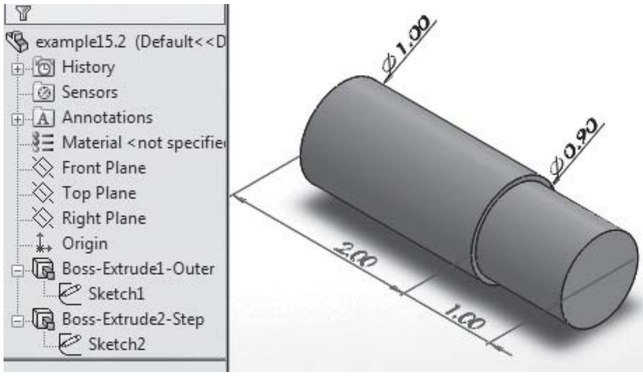
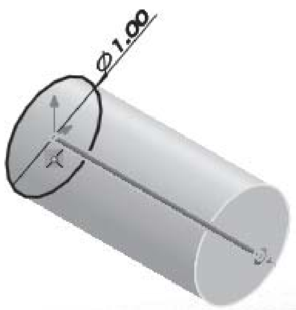


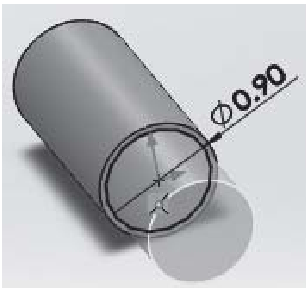
FIGURE 15.15
A stepped shaft

Generate turning toolpath and CL file using CAMWorks

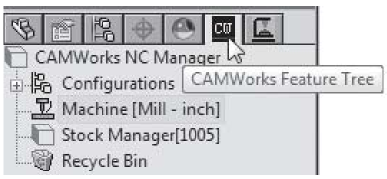
Step 1: Create *Sketch1* and *Boss-Extrude1-Outer*: **File > New > Part > OK > Right Plane > Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > sketch and dimension circle shown > exit sketch > enter 2 for thickness **D1** > ✔ > **File > Save As** > *Tutorial15.1* > **Save.**



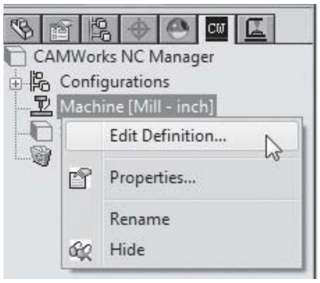
Step 2: Create *Sketch2* and *Boss-Extrude2-Step*: right flat face of *Boss-Extrude1-Outer* > **Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > sketch and dimension circle shown > exit sketch > enter 1 for thickness **D1** > ✔.

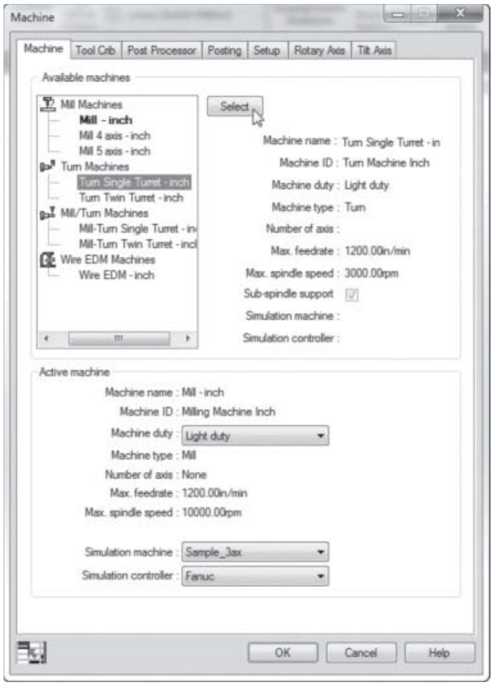


Step 3: Add CAMWorks: **Tools > Add-Ins** > check off **CAMWorks 2014** box in window that pops up > **OK > CAMWorks Feature Tree** icon on left pane shown here.



Step 4: Edit default machine: Right-click **Machine [Mill – inch] > Edit Definition** to open **Machine** window shown.



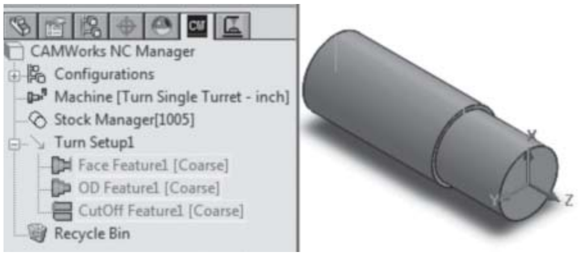


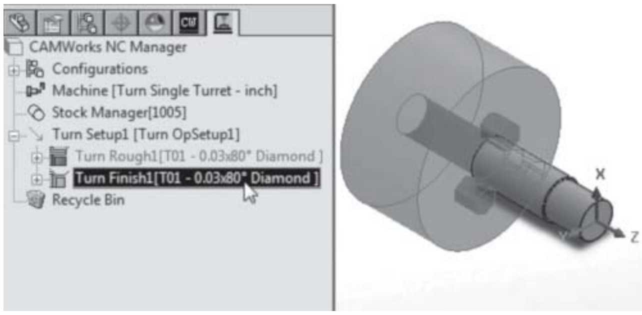
Note: We need to change the default milling machine to a lathe to turn the shaft.

Step 5: Select turn machine: **Turn Single Turret – inch** shown > **Select** button shown > **OK.**

Note: Select button sets **Turn Single Turret – inch** as the Active machine.

Step 6: Extract machinable features: **Extract Machinable Features** on **CAMWorks 2014** tab > right-click **Face Feature1 [Coarse]** shown > **Delete** from popup menu > **Yes** > right-click **CutOff Feature1 [Coarse]** shown > **Delete** from popup menu > **Yes.**





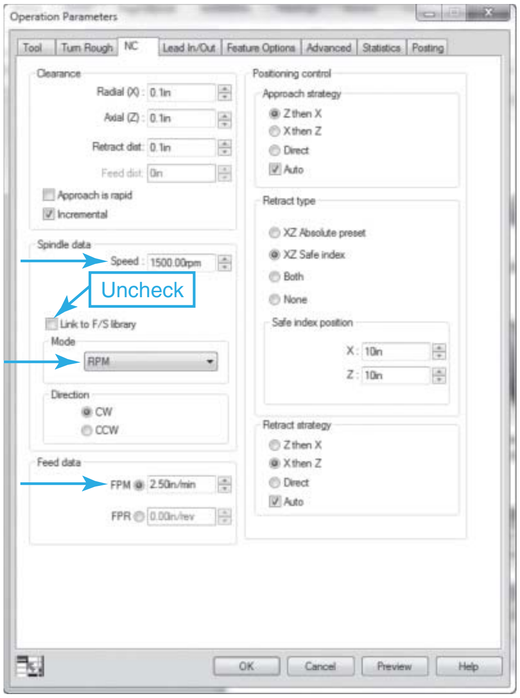
Note: Three features are extracted, facing, turn, and cut off. We delete facing and cut off for simplicity. We only need to create a step in the shaft in this example.

Note: Deleted features go under the **Recycle Bin** node shown. Right-click feature in **Recycle Bin** > **Remove** to delete permanently, or **Restore** to use if needed.

Step 7: Generate operation plan: **Generate Operation Plan** on **CAMWorks 2014** tab.

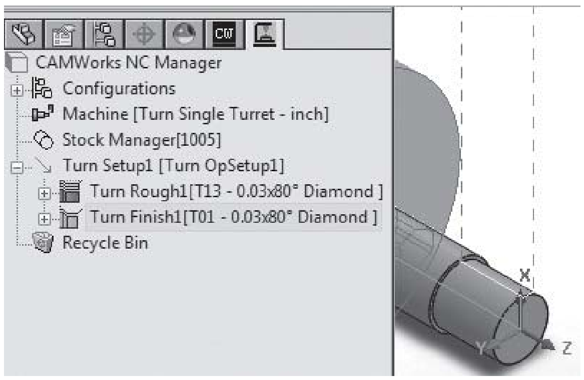
Note: This step generates two nodes shown above, one for rough turn and one for finish turn. It also displays the machine chuck shown with two grips gripping the shaft.

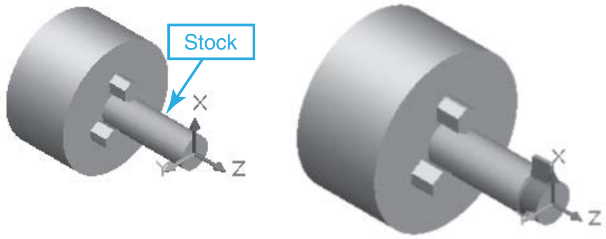
Note: CAMWorks displays tool specs and material for each operation.

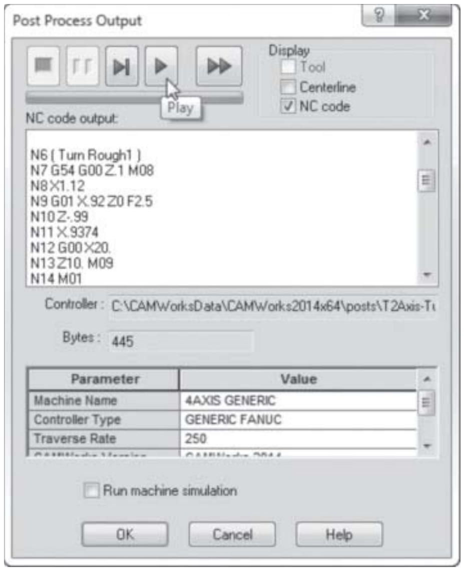


Step 8: Define machining parameters: Right-click **Turn Rough1[…]** > **Edit Definition > Tool** tab in **Operation Parameters** window that opens up and shown above > change **Shank length** to 2 > **NC** tab > make changes shown > **Posting** tab > **On** from **Coolant** dropdown > **OK** > repeat for **Turn Finish[…]** and use same spindle speed of 5000 RPM and same feedrate of 2.5 in./min > **OK.**

Step 9: Generate toolpath: **Generate toolpath** on **CAMWorks 2014** tab > click a Turn node in **CAMWorks Feature Tree** to view its path.







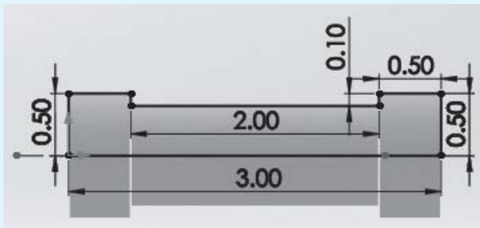
Step 10: Verify toolpath: **Simulate toolpath** on **CAMWorks 2014** tab to open **Toolpath Simulation** window shown above > **Run.**

Step 11: Post-process toolpath: **Post Process** on **CAMWorks** tab > accept file (*tutorial15.1.txt*) shown in window that opens up > **Save > Play** button shown to display code > **OK.**

Step 12: Save CL file: **Save CL File** on **CAMWorks** tab > accept file (*tutorial15.1.clt*) shown in window that opens up > **Save** > open file in Notepad to view.

Note: The CL file is not G-code and M-code. It is some form of a higher-level language, similar to APT (not covered here). The *tutorial15.1.txt* file (Step 11) is G-code and M-code.

HANDS-ON FOR TUTORIAL 15–1. Redo the tutorial to turn the middle of the shaft (i.e., turn the middle 2 in. length of the shaft as shown to the right).



Tutorial 15–2: Drill Holes

We generate the toolpath and CL file to drill the two holes shown in Figure 15.16. All dimensions are in inches. The cutting parameters are spindle speed = 5000 rpm and feedrate = 2.50 in./min. Coolant is on in flood mode.

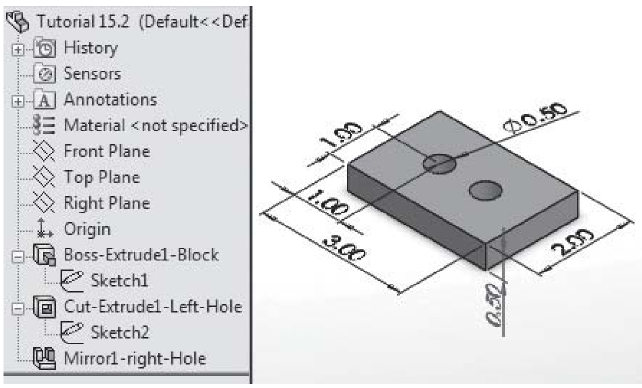
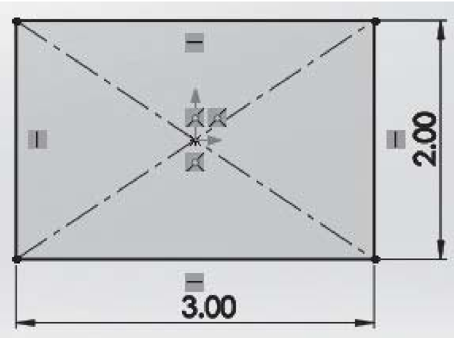
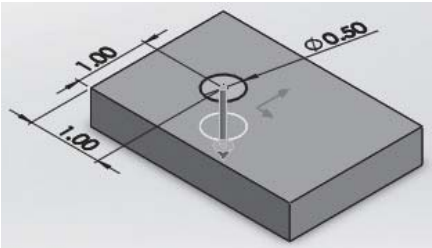


FIGURE 15.16
Block with two holes

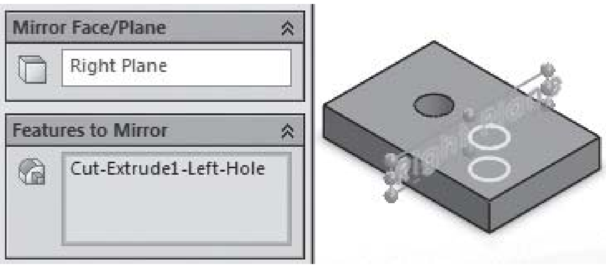
Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK > Top Plane > Extruded Boss/Base** on **Features** tab > **Center rectangle** on **Sketch** tab > sketch and dimension shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As** > *Tutorial15.2* > **Save.**



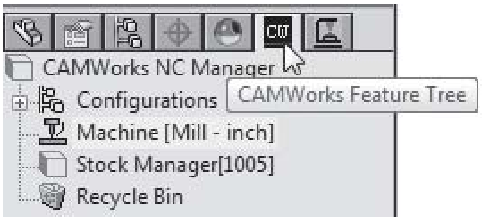
Step 2: Create *Sketch2* and *Cut-Extrude1-Left-Hole*: Top face of block > **Extruded Cut** on **Features** tab > **Circle** on **Sketch** tab > sketch and dimension circle shown > exit sketch > **Through All** > ✔.



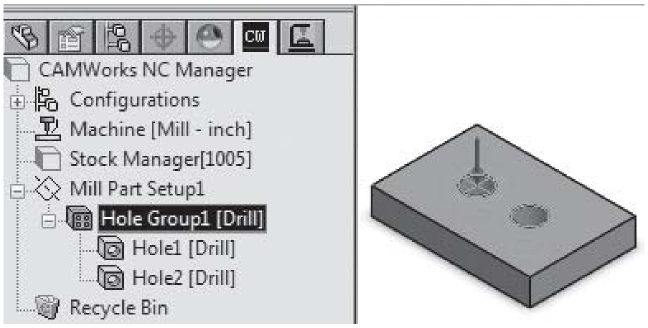
Step 3: Create *Mirror1-Right-Hole*: **Mirror** on **Features** tab > **Right Plane** from features tree > *Cut-Extrude1-Left-Hole* from features tree > ✔.



Step 4: Add CAMWorks: **Tools > Add-Ins >** check off **CAMWorks 2014** box in window that pops up > **OK** > **CAMWorks Feature Tree** on left pane.



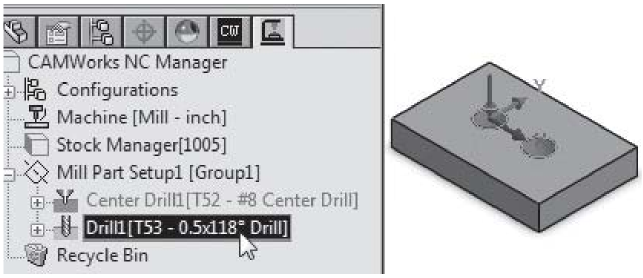
Step 5: Extract machinable features: **Extract Machinable Features** on **CAMWorks 2014** tab.



Note: CAMWorks recognizes the two hole features, extracts them, and adds the **Hole Group1 [Drill]** node shown.

Note: No need to edit machine definition as we can drill using a milling machine.

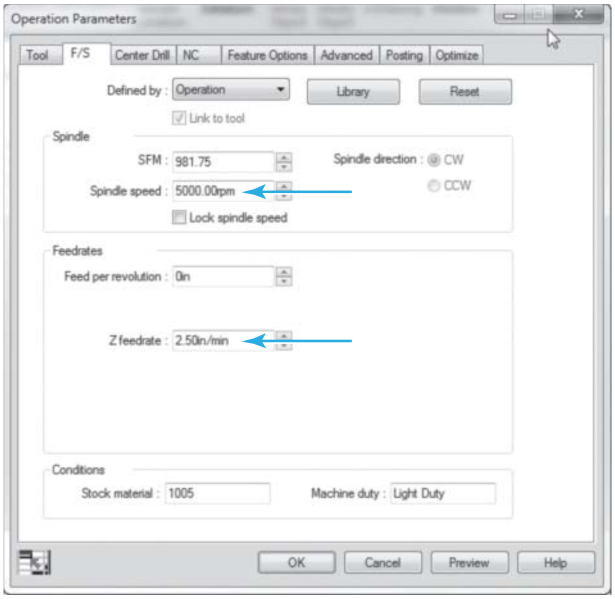
Step 6: Generate operation plan: **Generate Operation Plan** on **CAMWorks 2014** tab.

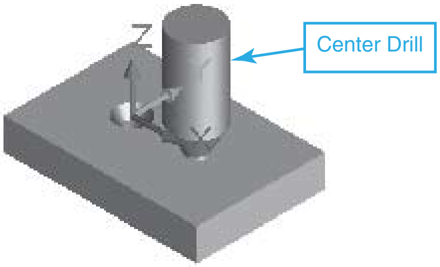


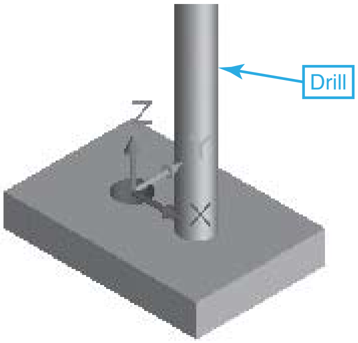
Note: This step generates two nodes shown, one for each hole.

Note: CAMWorks displays tool specs and drill type for each operation.

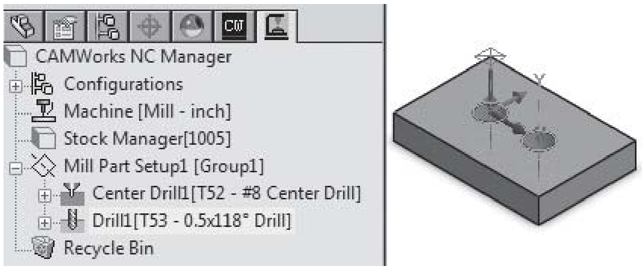
Step 7: Define machining parameters: Right-click **Center Drill1[...]** > **Edit Definition > Tool** tab in **Operation Parameters** window that opens up as shown > change **Overall length** to 2 > **F/S** tab > make changes shown > **Posting** tab > **Flood** from **Coolant** dropdown > **OK** > repeat for **Drill[...]** and use same spindle speed of 5000 RPM and same feedrate of 2.5 in./min > **OK.**



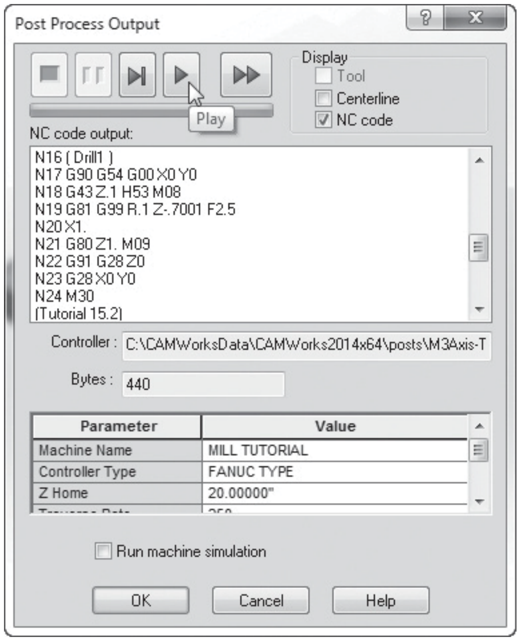




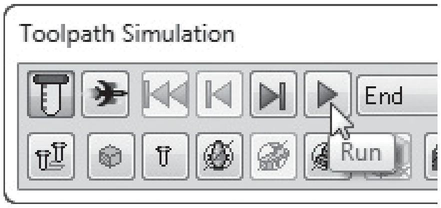
Step 8: Generate toolpath: **Generate toolpath** on **CAMWorks 2014** tab > click a **Drill** node in **CAMWorks Feature Tree** to view its path.

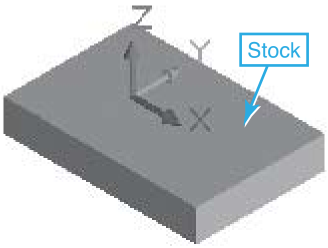


Step 10: Post-process toolpath: **Post Process** on **CAMWorks** tab > accept file (*tutorial15.2.txt*) shown in window that opens up > **Save > Play** button shown to display code > **OK.**



Step 9: Verify toolpath: **Simulate toolpath** on **CAMWorks 2014** tab to open **Toolpath Simulation** window shown below > **Run.**

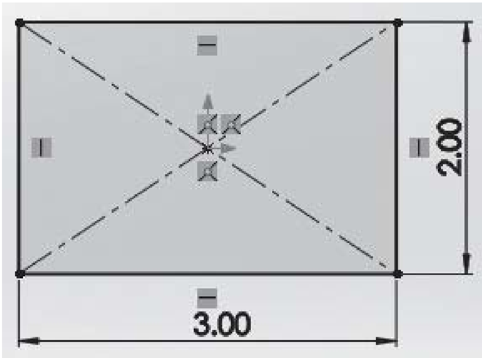




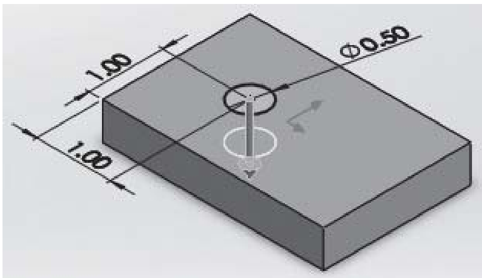
Step 11: Save CL file: **Save CL File** on **CAMWorks** tab > accept file (*tutorial15.2.clt*) shown in window that opens up > **Save** > open file in Notepad to view.

Note: The CL file is not G-code and M-code. It is some form of a higher-level language, similar to APT (not covered here). The *tutorial15.2.txt* file (Step 10) is G-code and M-code.

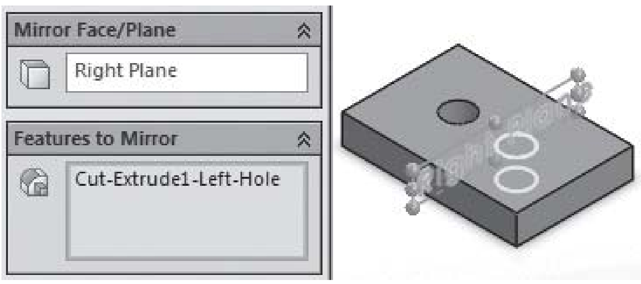
Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK > Top Plane > Extruded Boss/Base** on **Features** tab > **Center rectangle** on **Sketch** tab > sketch and dimension shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As** > *Tutorial15.2* > **Save.**



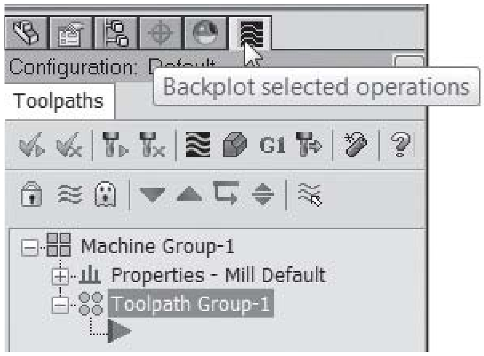
Step 2: Create *Sketch2* and *Cut-Extrude1-Left-Hole*: Top face of block > **Extruded Cut Features** tab > **Circle** on **Sketch** tab > sketch and dimension circle shown > exit sketch > **Through All** > ✔.

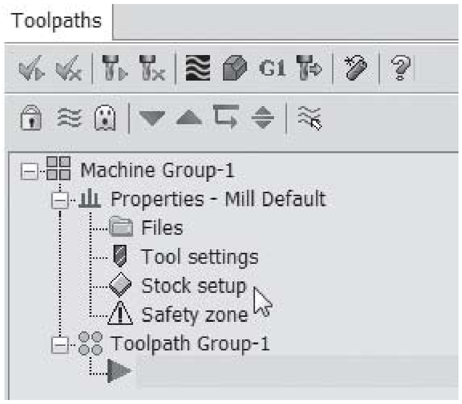


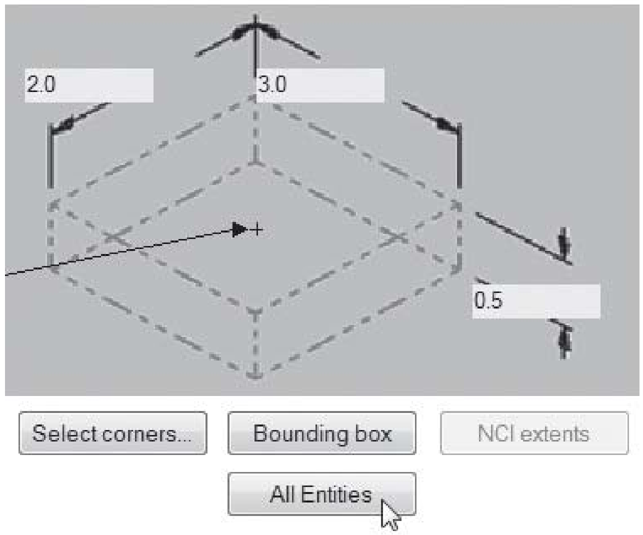
Step 3: Create *Mirror1-Right-Hole*: **Mirror** on **Features** tab > **Right Plane** from features tree > *Cut-Extrude1-Left-Hole* from features tree > ✔.



Step 4: Add Mastercam: **Tools > Add-Ins** > check off **Master X7 for SolidWorks** box in window that pops up > **OK** > **Backplot selected operations** icon shown.



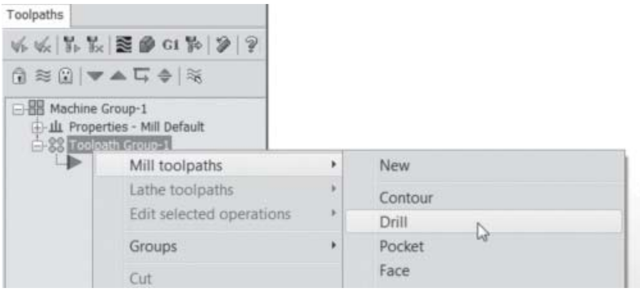




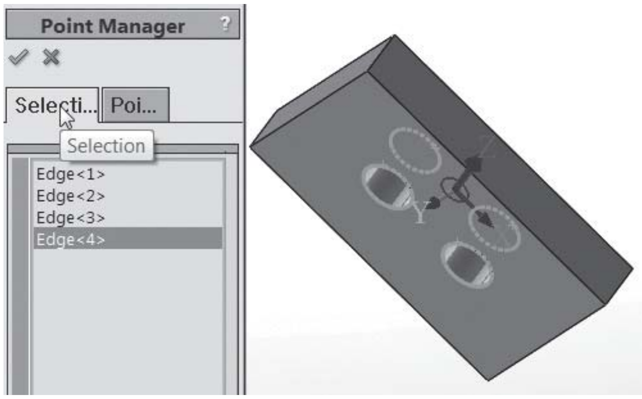
Step 5: Define stock size: Expand **Properties – Mill Default** node shown above > **Stock setup** > **All Entities** button > ✔.

Note: Without this step the workpiece display in Step 10 will look like a sheet.

Step 6: Select drill operation: Right-click **Toolpath Group-1** shown > **Mill toolpaths > Drill.**



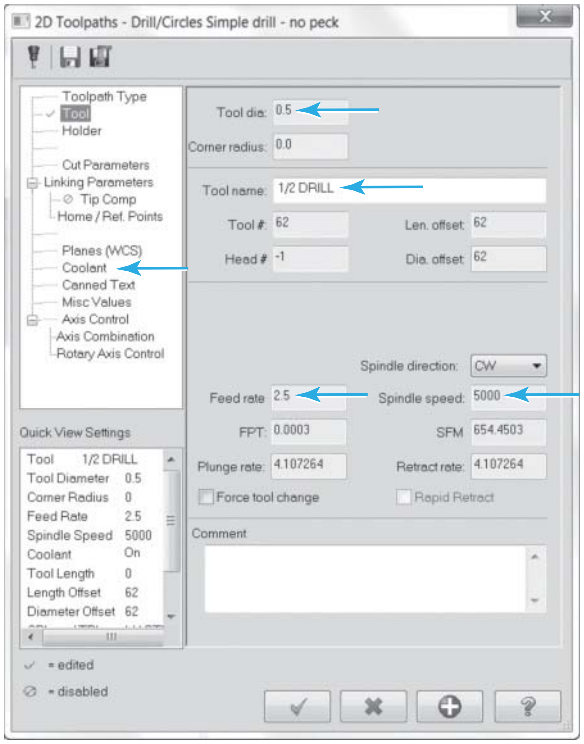
Note: We drill on a milling machine or a lathe. There are no drilling-only machines.



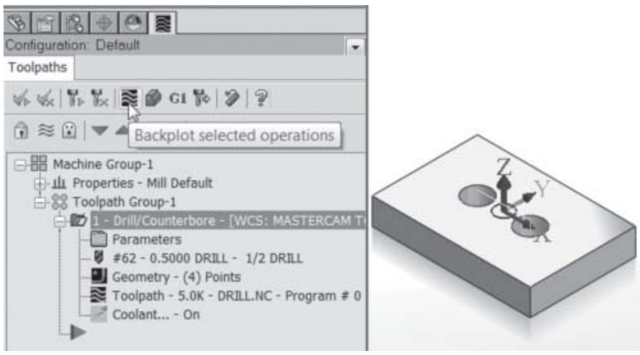
Step 7: Define holes to drill: Top and bottom edges of left hole as shown above > top and bottom edges of right hole > ✔ > type *drill* for **NC name** as shown above > **OK.**

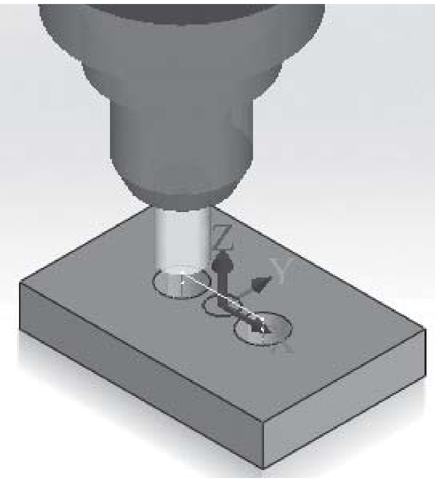
Note: Instead of selecting top and bottom edges of the holes, you may select the top edges only and specify the hole thickness as a negative depth in the *Linking Parameters* link shown (click it) in Step 8.

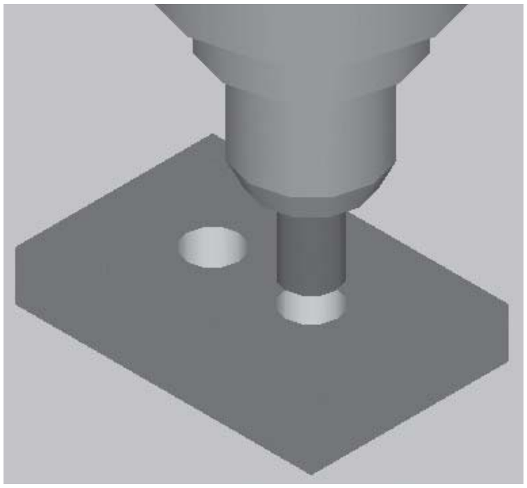
Step 8: Define machining parameters: **Tool** in the **2D Toolpaths** window that opens as shown below > change **Tool dia** to 0.5 (hole size in this tutorial) > change **Tool name** to *½ DRILL* > change **Feed rate** to 2.5 > change **Spindle speed** to 5000 > **Coolant** > **On** from the **Flood** dropdown > ✔.

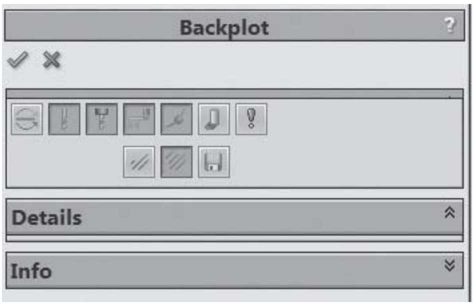


Note: Toolpath is generated at end of this step as shown below. Yellow path indicates that tool is not in contact with workpiece during motion. Blue path indicates contact.



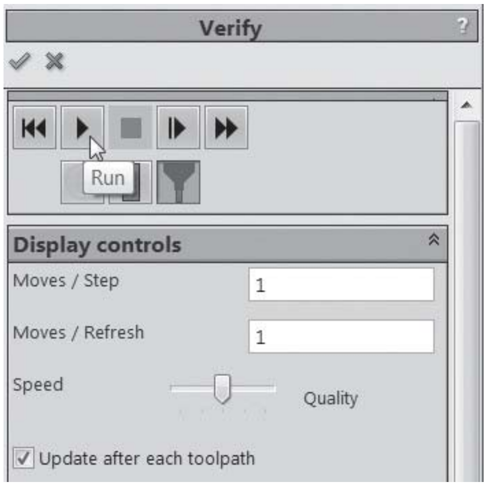






Step 9: Verify toolpath using **Backplot**: **Backplot** icon shown in Step 8 > **Play** button shown above.

Note: Use **Backplot** control bar (shown above) that opens up to control display.



Step 10: Verify toolpath using **Verify**: **Verify** icon (next to **Backplot** icon shown in Step 8; hover until you read it) > **Run** shown at bottom left.

Note: Investigate the pane shown above to learn more.

Step 11: Post-process and save CL file: **G1** icon (shown in Step 8) > ✔ to accept **Post Processing** parameters in window that opens up > type *drill* for NC file name in window that opens up > **Save.**

```
N100 G20
N102 G0 G17 G40 G49 G80 G90
N104 T62 M6
N106 G0 G90 G54 X-.5 Y0. A0. S5000 M3
N108 G43 H62 Z.1 M8
N110 G99 G81 Z0. R.1 F2.5
N112 Z-.5
N114 X.5 Z0.
N116 Z-.5
N118 G80
N120 M5
N122 G91 G28 Z0. M9
N124 G28 X0. Y0. A0.
N126 M30
%
```

Note: View code generated and shown here.

HANDS-ON FOR TUTORIAL 15–2. Redo the tutorial to drill the two holes with a new size of 0.75 in. diameter. Also, define the home position of the toolpath at the bottom left corner of the block top face. Define the toolpath to start and end at home position.

Tutorial 15–3: Mill Faces

This tutorial illustrates 2½-axis milling. We generate the toolpath and CL file to mill the top face of the block shown in Figure 15.17. All dimensions are in inches. The cutting parameters are spindle speed = 1500 rpm and feedrate = 2.50 in./min. Coolant is on in flood mode.

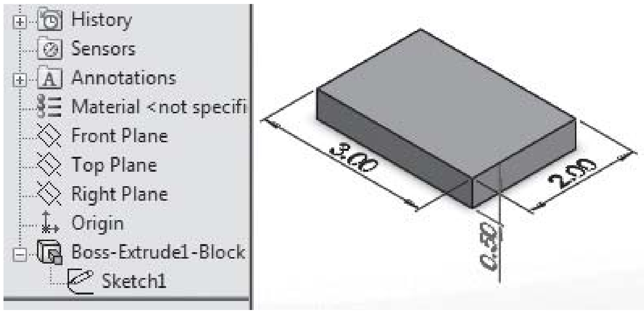
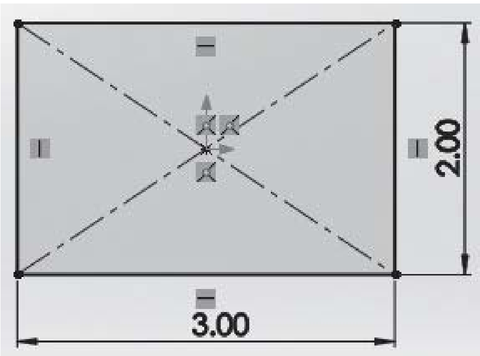


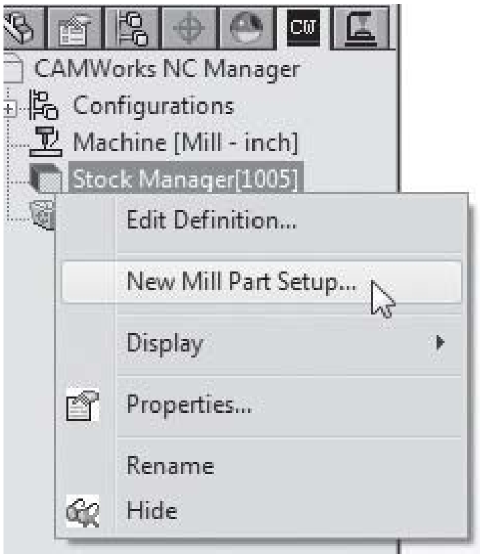
FIGURE 15.17
A block

We use 2½-axis milling operation in CAMWorks to mill the top face. We use face milling in Mastercam.

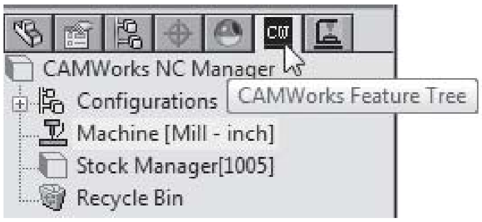
Generate face milling toolpath and CL file using CAMWorks

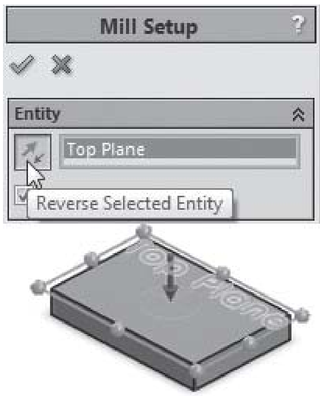
Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK > Top Plane > Extruded Boss/Base** on **Features** tab > **Center rectangle** on **Sketch** tab > sketch and dimension shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As >** *Tutorial15.3* **> Save.**





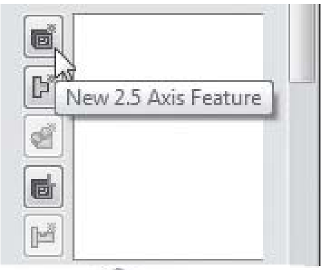
Step 2: Add CAMWorks: **Tools > Add-Ins >** check off **CAMWorks 2014** box in window that pops up > **OK > CAMWorks Feature Tree** on left pane.

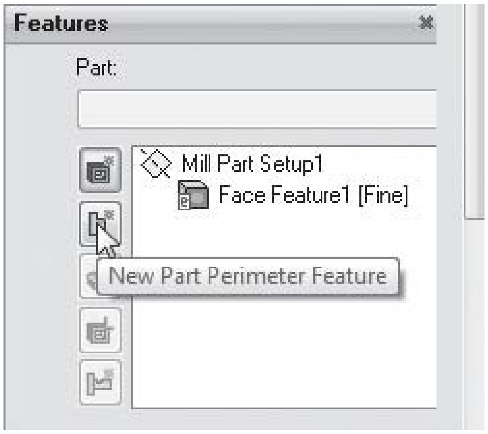




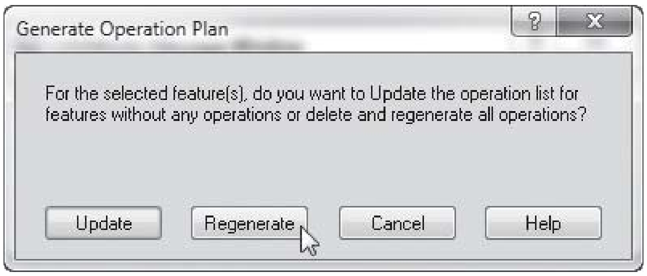
Step 3: Define a new part setup: Right-click **Stock-Manager[…]** > **New Mill Part Setup** shown > **Top Plane** from features tree > click icon shown to reverse direction > ✔.

Step 4: Define new 2.5 axis milling operation: **New 2.5 Axis Mill Operation** dropdown on **CAMWorks 2014** tab > **Face Mill** > top face of block > **New 2.5 Axis Feature** icon shown > **Next** button in **2.5 Axis Feature Wizard** window that opens up > **Fine** from **Strategy** dropdown > **Next** > **Finish** > **New Part Perimeter Feature** icon shown below > **Fine** from dropdown that shows up > ✔ > ✔ > this opens up **Operations Parameters** window > **F/S** tab > **Operation** from **Defined by** dropdown > set **Spindle Speed** to 1500 (this sets the feedrate automatically) > **OK.**



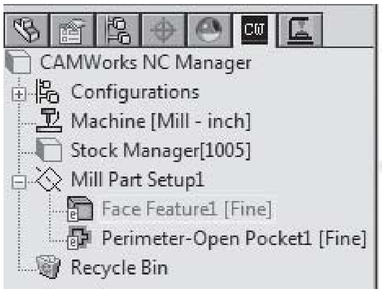


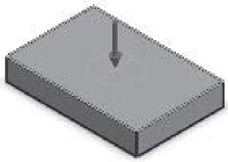
Note: The milling operation picks up the **Mill Part Setup1** created in Step 3 (see CW features tree in left pane on screen) automatically.



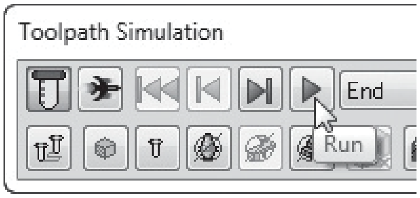
Step 5: Generate operation plan: **Generate Operation Plan** on **CAMWorks 2014** tab > **Regenerate** button shown.

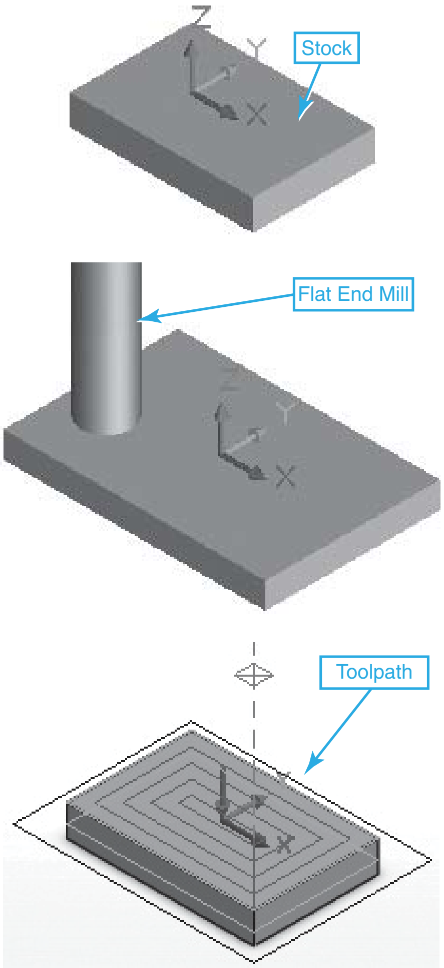
Step 6: Generate toolpath: **Generate toolpath** on **CAMWorks 2014** tab.



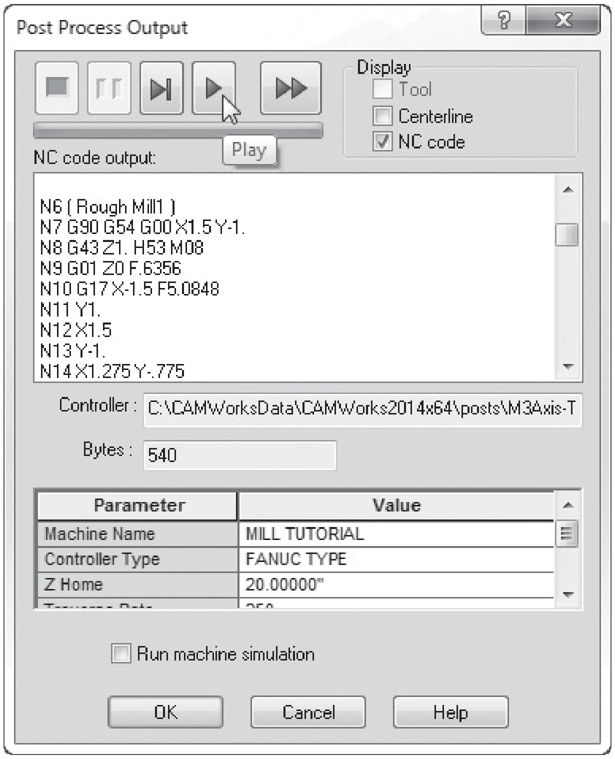


Step 7: Verify toolpath: **Simulate toolpath** on the **CAMWorks 2014** tab to open **Toolpath Simulation** window shown > **Run.**



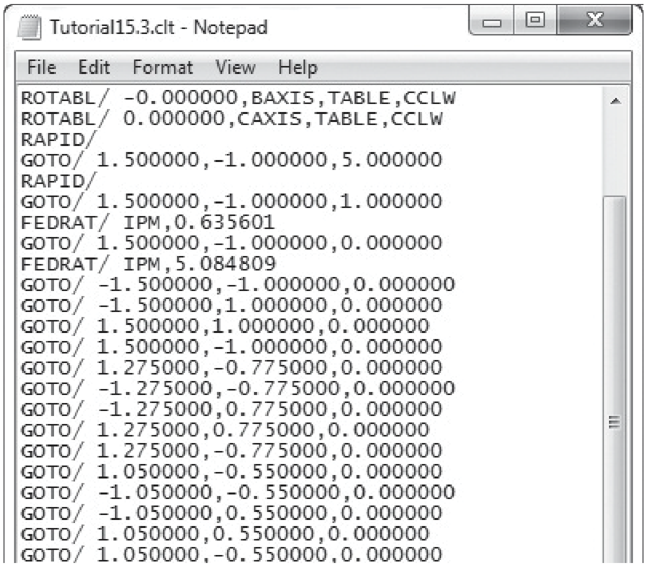


Step 8: Post-process toolpath: **Post Process** on **CAMWorks** tab > accept file (*tutorial15.3.txt*) shown in window that opens up > **Save > Play** button shown to display code > **OK.**



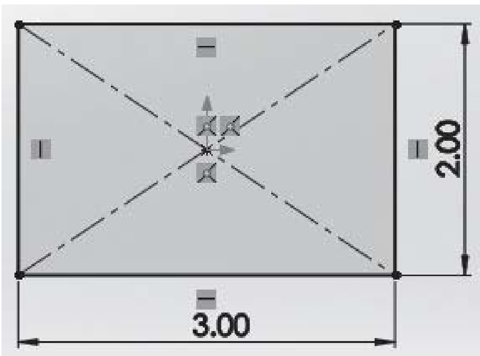
Step 9: Save CL file: **Save CL File** on **CAMWorks** tab > accept file (*tutorial15.3.clt*) shown in window that opens up > **Save** > open file in Notepad to view.

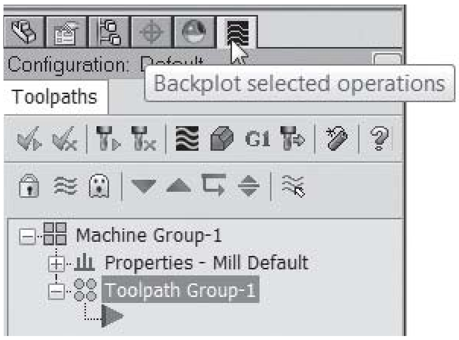
Note: The CL file (see below) is not G-code and M-code. It is some form of a higher-level language, similar to APT (not covered here). The *tutorial15.3.txt* file (Step 8) is G-code and M-code.



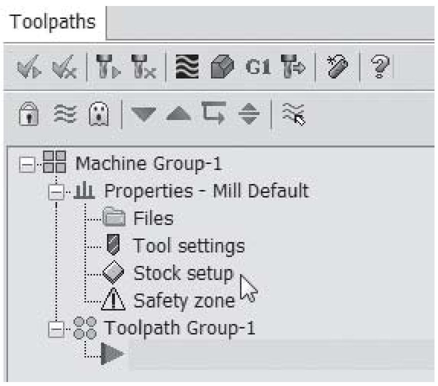
Generate face milling toolpath and CL file using Mastercam

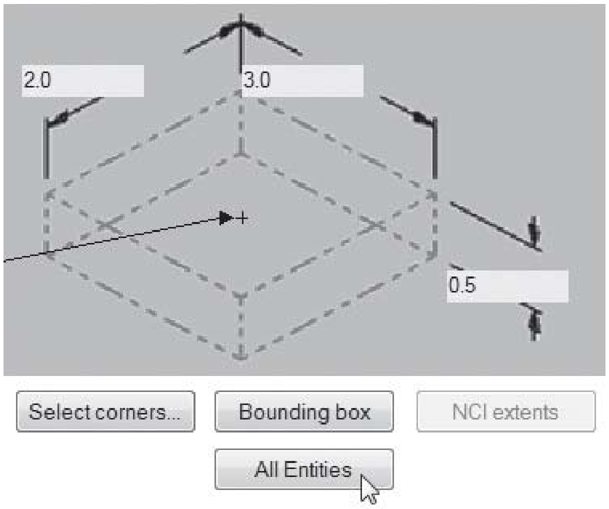
Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK > Top Plane > Extruded Boss/Base** on **Features** tab > **Center rectangle** on **Sketch** tab > sketch and dimension shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As** > *Tutorial15.3* > **Save.**





Step 2: Add Mastercam: **Tools > Add-Ins** > check off **Master X7 for SolidWorks** box in window that pops up > **OK > Backplot selected operations** on left pane.

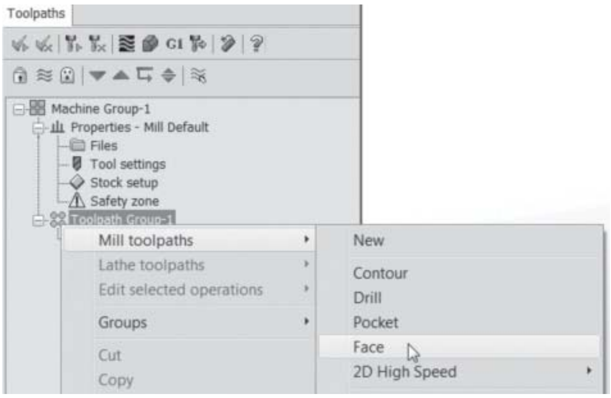


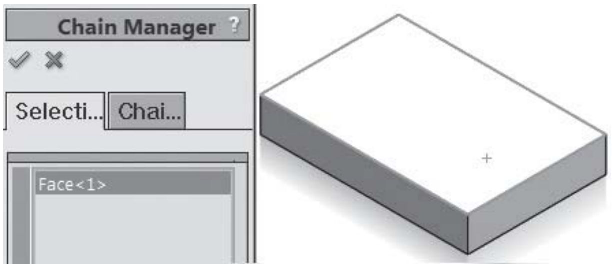


Step 3: Define stock size: Expand **Properties - Mill Default** node shown > **Stock setup** > **All Entities** button > ✔.

Note: Without this step the workpiece display in Step 8 will look like a sheet.

Step 4: Select face operation: Right-click **Toolpath Group-1** shown > **Mill toolpaths** > **Face.**

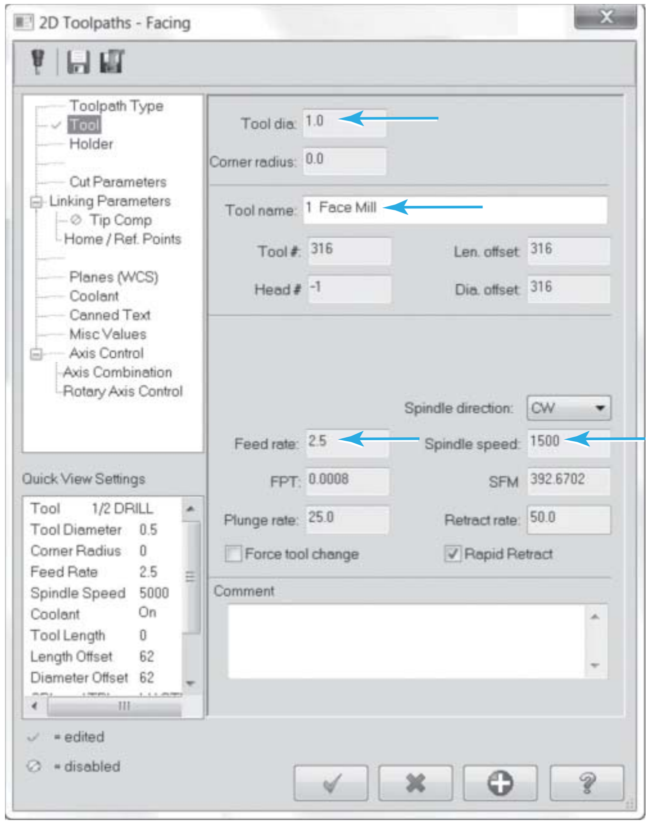


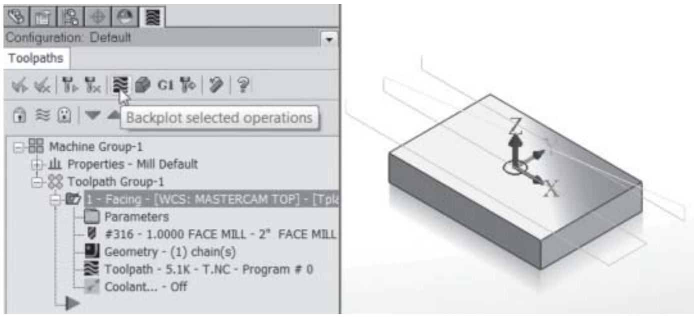


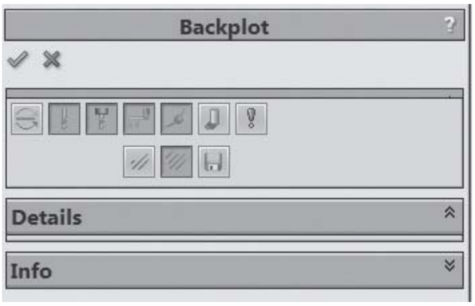
Step 5: Define face to mill: Top face shown at left below > ✔ > type *facemill* for **NC name** as shown > **OK.**

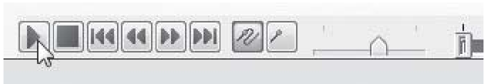
Step 6: Define machining parameters: **Tool** in the **2D Toolpaths** window that opens as shown here > change **Tool dia** to 1.0 > change **Tool name** to *1 Face Mill* > change **Feed rate** to 2.5 > change **Spindle speed** to 1500 > **Coolant** on left pane > **On** from the **Flood** dropdown > ✔.

Note: Toolpath is generated at end of this step as shown below. Yellow path indicates that tool is not in contact with workpiece during motion. Blue path indicates contact.



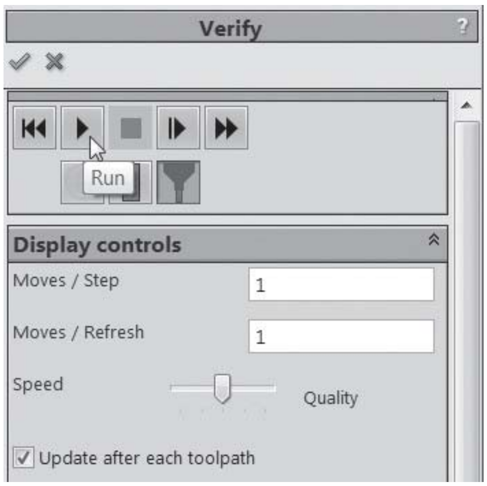


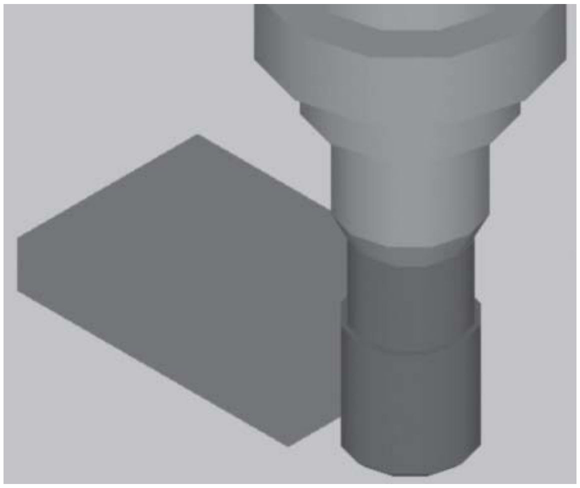




Step 7: Verify toolpath using **Backplot**: **Backplot** icon shown in Step 6 > **Play** shown above > ✔.

Note: Use **Backplot** pane (shown above) that opens up to control display.





Step 8: Verify toolpath using **Verify**: **Verify** icon (next to Backplot icon shown in Step 6; hover until you read it) > **Run** shown to the left below > ✔.

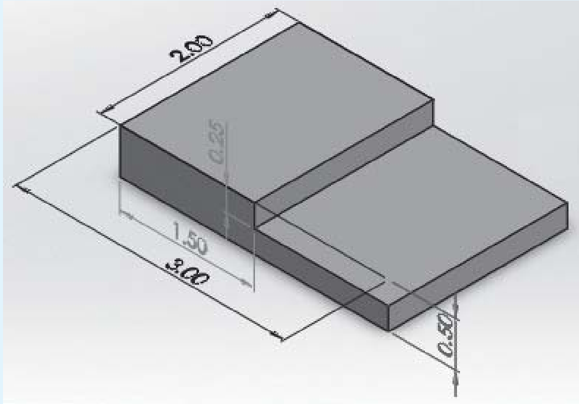
Note: Investigate pane shown to learn more.

Step 9: Post-process and save CL file: **G1** icon (shown in Step 6) > ✔ to accept **Post Processing** parameters in window that opens up > type *facemill* for NC file name in window that opens up > **Save.**

```
N100 G20
N102 G0 G17 G40 G49 G80 G90
N104 T316 M6
N106 G0G90G54X-2.6Y.7499A0.S150CM3
N108 G43 H316 Z.25 M8
N110 Z.2
N112 G1 Z0. F25.
N114 X2.6 F2.5
N116 G0 Z.25
N118 X-2.6 Y0.
N120 Z.2
N122 G1 Z0. F25.
N124 X2.6 F2.5
N126 Y-.7499
N128 X-2.6
N130 G0 Z.25
N132 M5
N134 G91 G28 Z0. M9
N136 G28 X0. Y0. A0.
N138 M30
%
```

Note: View code generated and shown here.

HANDS-ON FOR TUTORIAL 15–3. Edit the part to create a ¼ in. step in top face in the middle as shown to the right. Mill the new face. Use a Zigzag toolpath. Also, define the toolpath so that the mill tool cuts (zigzags) along the 2.00 inch side.



Tutorial 15–4: Mill Pockets

This tutorial illustrates pocket milling. We generate the toolpath and CL file to mill the pocket shown in Figure 15.18. All dimensions are in inches. The cutting parameters are spindle speed = 5000 rpm and feedrate = 2.50 in./min. Coolant is on.

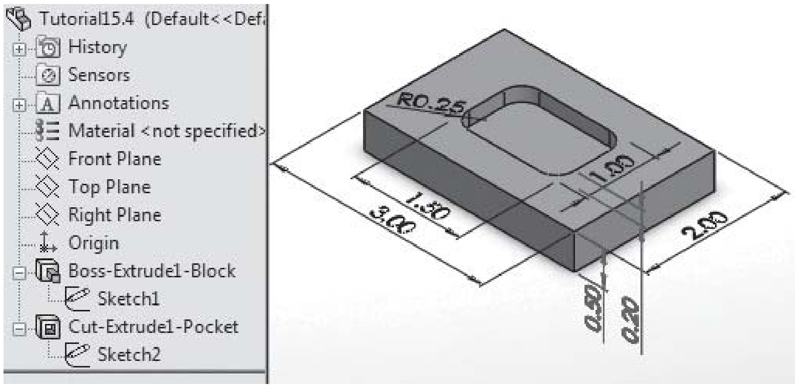
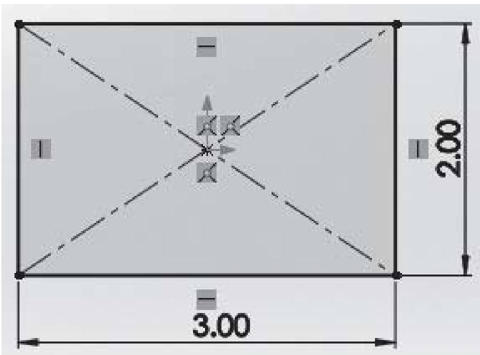


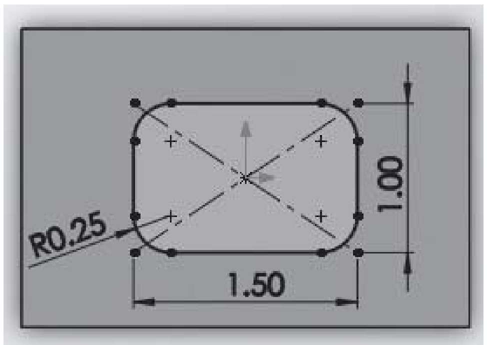
FIGURE 15.18
A block with a pocket

Generate pocket milling toolpath and CL file using CAMWorks

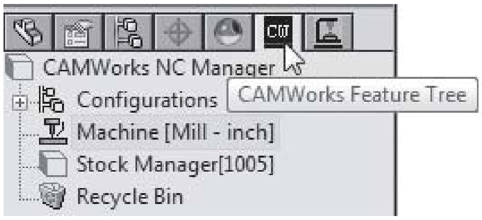
Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK > Top Plane > Extruded Boss/Base** on **Features** tab > **Center rectangle** on **Sketch** tab > sketch and dimension shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As** > *Tutorial15.4* > **Save.**



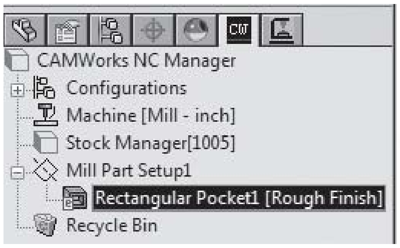
Step 2: Create *Sketch2* and *Cut-Extrude1-Pocket*: Top face of block > **Extruded Cut** on **Features** tab > **Center rectangle** on **Sketch** tab > sketch as shown > **Fillet** on **Sketch** tab > fillet and dimension as shown > exit sketch > enter **0.2** for thickness **D1** > ✔.

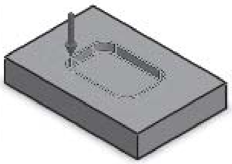


Step 3: Add CAMWorks: **Tools > Add-Ins >** check off **CAMWorks 2014** box in window that pops up > **OK** > **CAMWorks Feature Tree** shown.



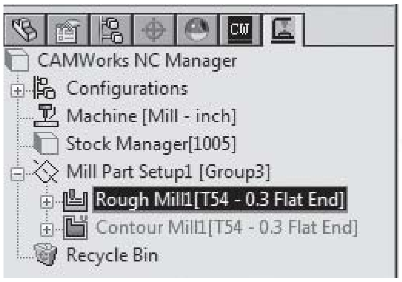
Step 4: Extract machinable features: **Extract Machinable Features** on **CAMWorks 2014** tab.

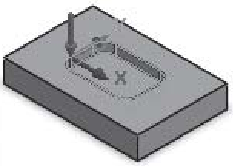




Note: CAMWorks recognizes the pocket feature, extracts it, and adds **Rectangular Pocket1 [Rough Finish]** node shown.

Step 5: Generate operation plan: **Generate Operation Plan** on **CAMWorks 2014** tab.

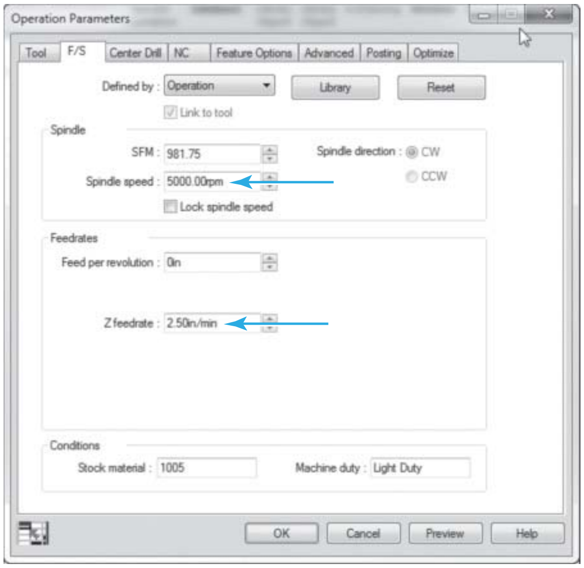




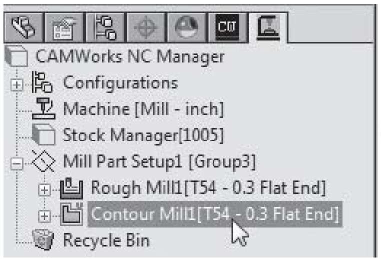
Note: This step generates the two nodes shown above.

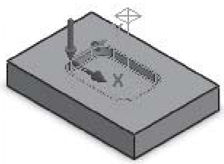
Note: CAMWorks displays tool specs and tool type for each operation.

Step 6: Define machining parameters: Right-click **Rough Mill1[…] > Edit Definition > Tool** tab in **Operation Parameters** window that opens up as shown > change **Overall length** to 2 > **F/S** tab > make changes shown > **Posting** tab > **Flood** from **Coolant** dropdown > **OK** > repeat for **Contour Mill1[. . .]** and use same spindle speed of 5000 RPM and same feedrate of 2.5 in./min > **OK.**

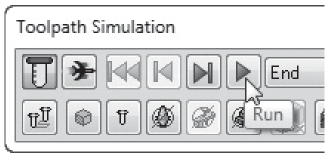


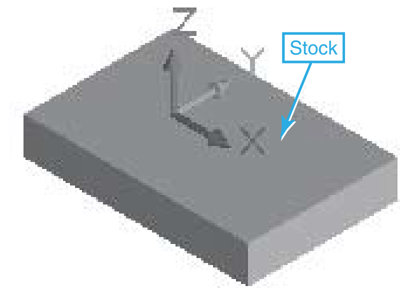
Step 7: Generate toolpath: **Generate toolpath** on **CAMWorks 2014** tab > click a **Mill** node in **CAMWorks Feature Tree** to view its path.

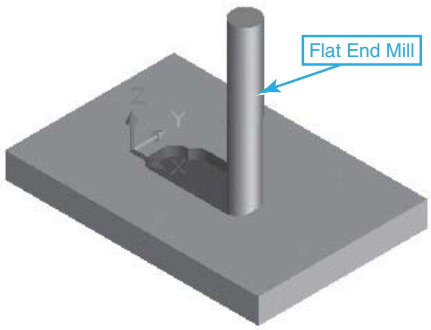


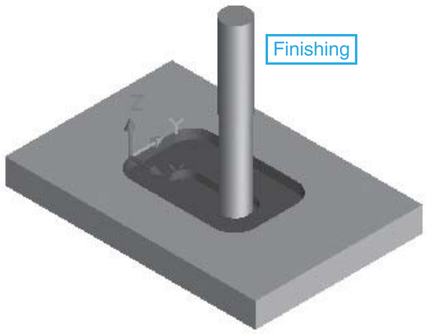


Step 8: Verify toolpath: **Simulate toolpath** on **CAMWorks 2014** tab



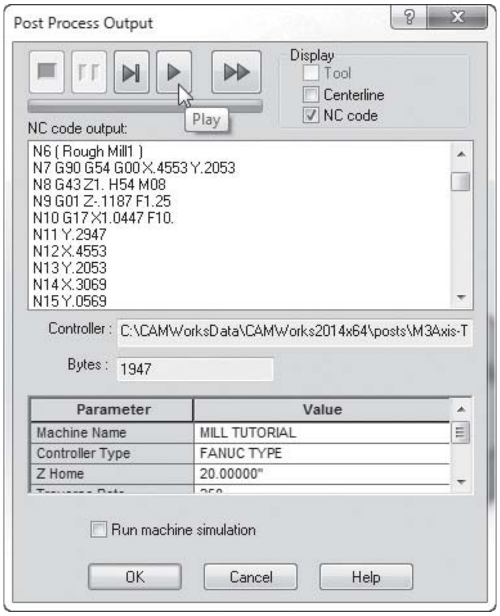






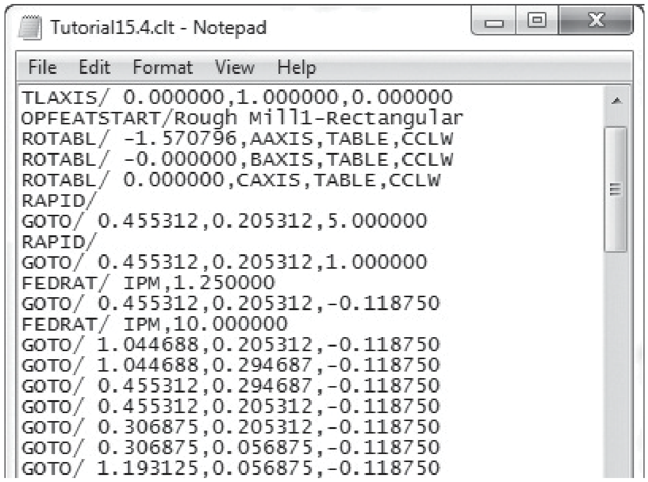
to open **Toolpath Simulation** window shown > **Run.**

Step 9: Post-process toolpath: **Post Process** on **CAMWorks** tab > accept file (*tutorial15.4.txt*) shown



in window that opens up > **Save > Play** button shown to display code > **OK.**

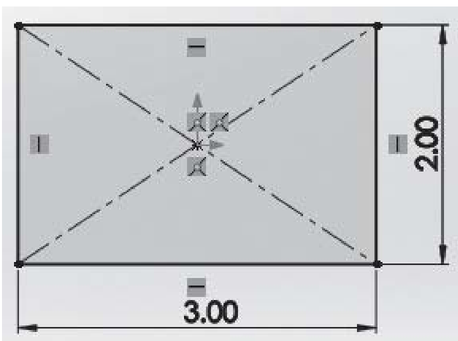
Step 10: Save CL file: **Save CL File** on **CAMWorks** tab > accept file (*tutorial15.4.clt*) shown in window that opens up > **Save** > open file in Notepad to view.



Note: The CL file is not G-code and M-code. It is some form of a higher-level language, similar to APT (not covered here). The *tutorial15.4.txt* file (Step 9) is G-code and M-code.

Generate pocket milling toolpath and CL file using Mastercam

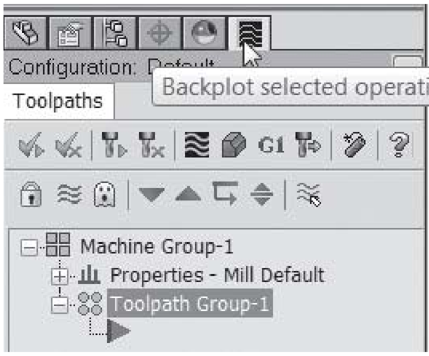
Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK > Top Plane > Extruded Boss/Base** on **Features** tab > **Center rectangle** on **Sketch** tab >

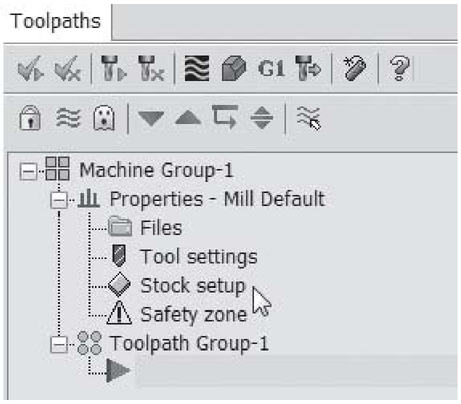


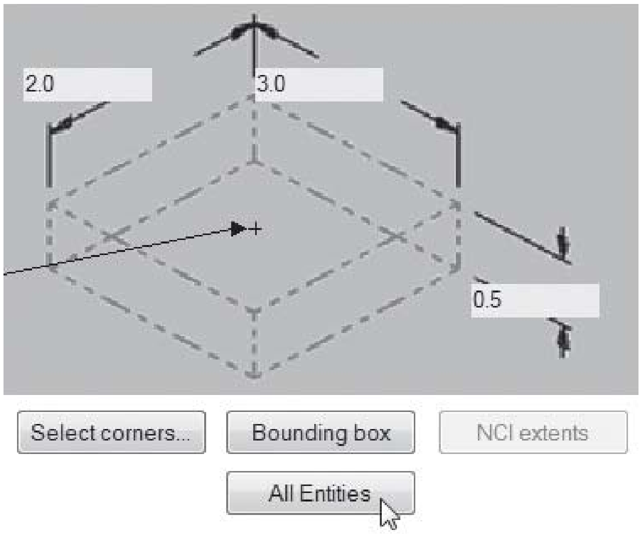
sketch and dimension shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As** > *Tutorial15.4* > **Save.**

Step 2: Create *Sketch2* and *Cut-Extrude1-Pocket*: Top face of block > **Extruded Cut** on **Features** tab > **Center rectangle** on **Sketch** tab > sketch as shown > **Fillet** on **Sketch** tab > sketch as shown > dimensions as shown > exit sketch > enter 0.2 for thickness **D1** > ✔.

Step 3: Add Mastercam: **Tools > Add-Ins** > check off **Master X7 for SolidWorks** box in window that pops up > **OK** > **Backplot selected operations** icon shown.



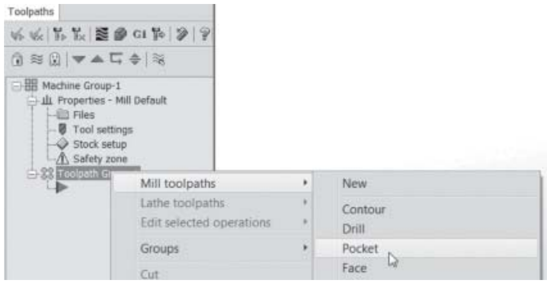


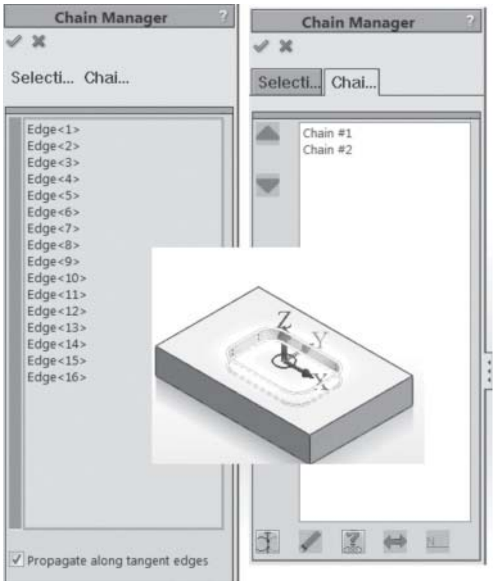


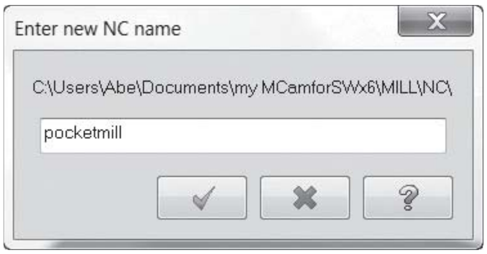
Step 4: Define stock size: Expand **Properties - Mill Default** node shown above > **Stock setup** > **All Entities** button > ✔.

Note: Without this step the workpiece display in Step 9 will look like a sheet.

Step 5: Select pocket operation: Right-click **Toolpath Group-1** shown to the right above > **Mill toolpaths** > **Pocket.**





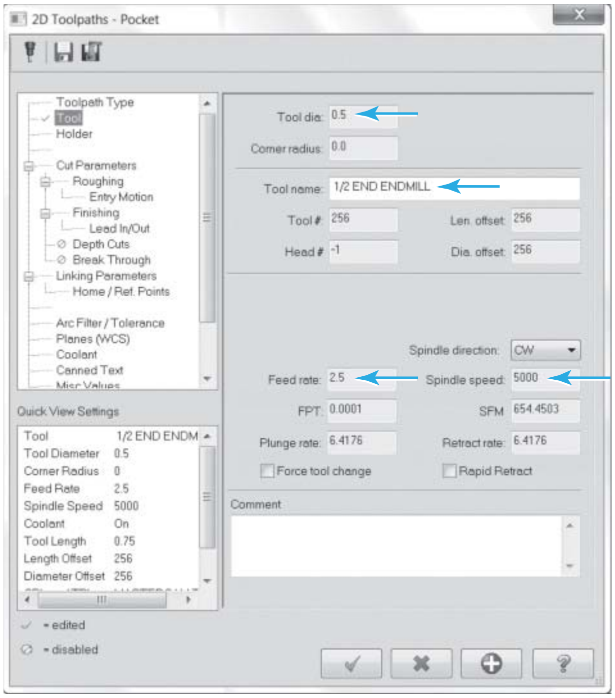


Step 6: Define pocket to mill: Edges as shown above > ✔ > type *pocketmill* for **NC name** as shown above > **OK.**

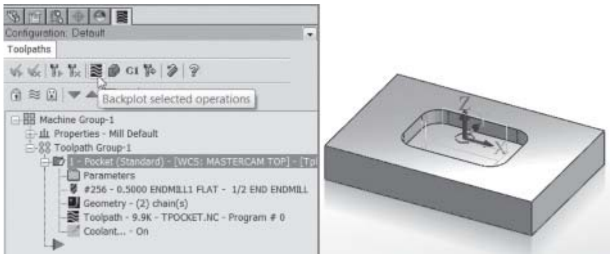
Note: Check off **Propagate along tangent edges** box shown to ease edge selection.

Note: Instead of selecting top and bottom edges of the pocket, you may select the top edges only and specify the pocket thickness as a negative depth in the *Linking Parameters* link shown (click it) in Step 7.

Step 7: Define machining parameters: **Tool** in the **2D Toolpaths** window that opens as shown > change **Tool dia** to 0.5 > change **Tool name** to *½ END MILL* > change **Feed rate** to 2.5 > change **Spindle speed** to 5000 > **Coolant** in window shown > **On** from the **Flood** dropdown > ✔.

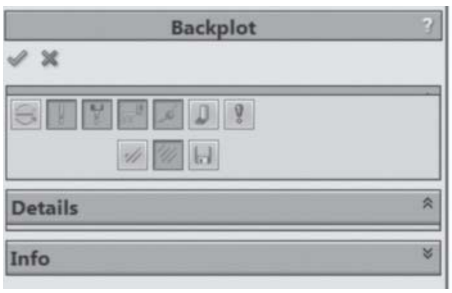


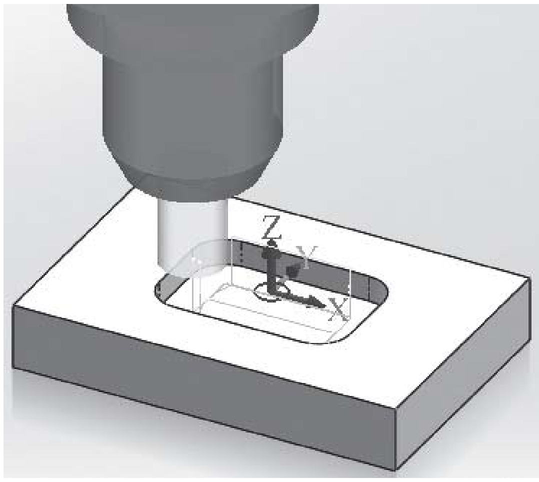
Note: The toolpath is generated at end of this step as shown here. A yellow path indicates that tool is not in contact with workpiece during motion. A blue path indicates contact.

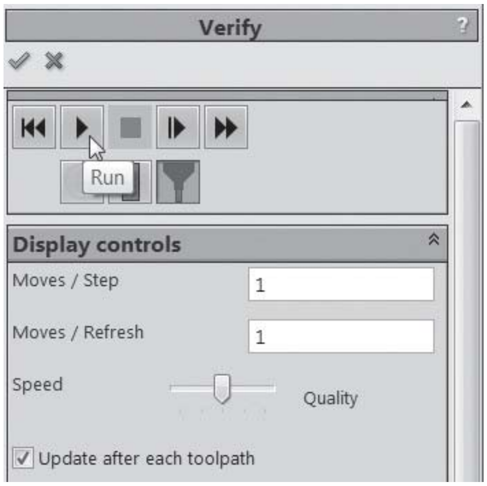


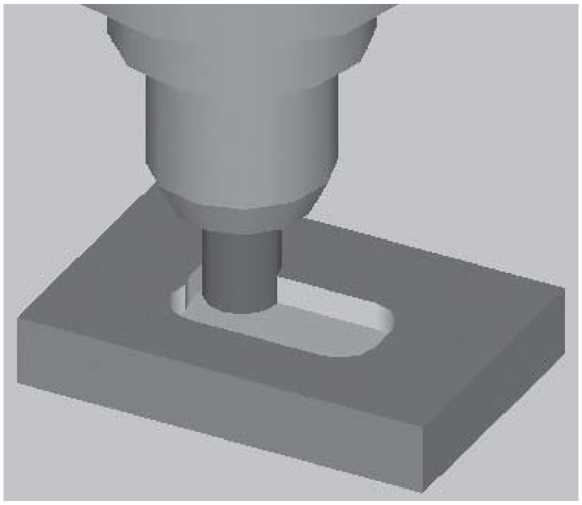
Step 8: Verify toolpath using **Backplot**: **Backplot** icon shown in Step 7 > **Play** shown to the left > ✔.

Note: Use **Backplot** control bar (shown below) that opens up to control display.









Step 9: Verify toolpath using **Verify**: **Verify** icon (next to Backplot icon shown in Step 7; hover until you read it) > **Run** shown above > ✔.

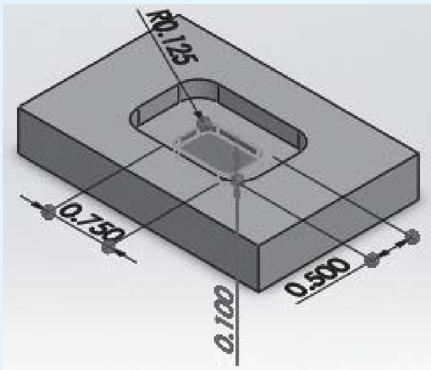
Note: Investigate the pane shown to learn more.

Step 10: Post-process and save CL file: **G1** icon (shown in Step 7) > ✔ to accept **Post Processing** parameters in window that opens up > type *pocketmill* for NC file name in window that opens up > **Save.**

Note: View the code generated and shown.

```
(MATERIAL - ALUMINUM INCH - 2024)
( T256 |  1/2 END ENDMILL | H256 )
N100 G20
N102 G0 G17 G40 G49 G80 G90
N104 T256 M6
N106 G0 G90 G54 X-.49 Y-.2399 A0. S1069 M3
N108 G43 H256 Z.25 M8
N110 Z.2
N112 G1 Z-.2 F6.42
N114 X.49
N116 Y0.
N118 X-.49
N120 Y.2399
N122 X.49
N124 Z0.
N126 G0 Z.05
N128 Z.2
N130 X-.4386 Y.24
N132 G1 Z-.2
N134 G3 X-.5 Y0. I.4386 J-.24
N136 G1 Y-.25
N138 X.5
N140 Y.25
N142 X-.5
N144 Y0.
N146 G3 X-.4386 Y-.24 I.5 J0.
N148 G1 Z0.
N150 G0 Z.25
N152 M5
N154 G91 G28 Z0. M9
N156 G28 X0. Y0. A0.
N158 M30
%
```

HANDS-ON FOR TUTORIAL 15–4. Edit the part to have another pocket (0.75 × 0.5 × 0.1) as shown. Mill the new pocket.



Tutorial 15–5: Mill Slots

This tutorial illustrates slot milling. We generate the toolpath and CL file to mill the slot shown in Figure 15.19. All dimensions are in inches. The cutting parameters are spindle speed = 5000 rpm and feedrate = 2.50 in./min. Coolant is on.

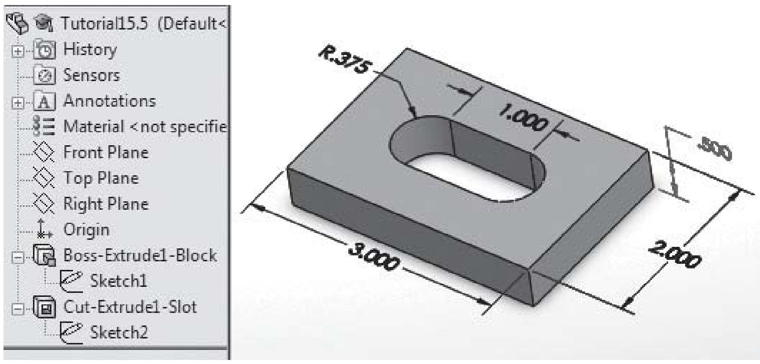
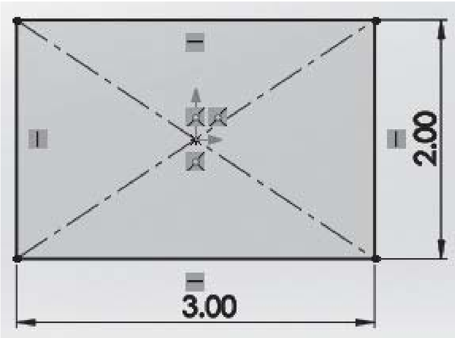
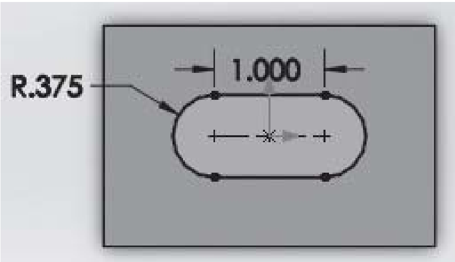


FIGURE 15.19
A block with a slot

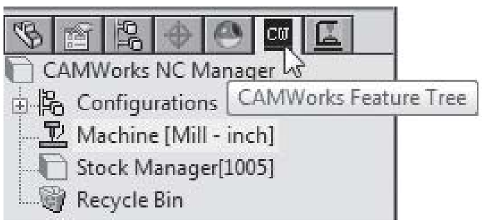
Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK > Top Plane > Extruded Boss/Base** on **Features** tab > **Center rectangle** on **Sketch** tab > sketch and dimension shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As** > *Tutorial15.5* > **Save.**



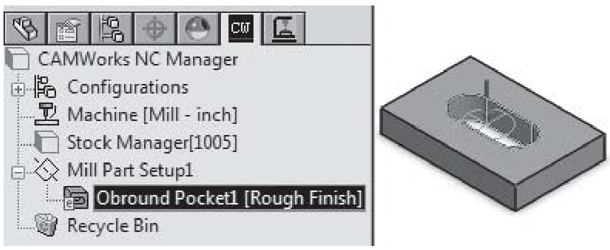
Step 2: Create *Sketch2* and *Cut-Extrude1-Slot*: Top face of block > **Extruded Cut** on **Features** tab > **Centerpoint Straight Slot** on **Sketch** tab > click origin, drag horizontally, click, drag vertically, click > dimensions as shown > exit sketch > **Through All** > ✔.



Step 3: Add CAMWorks: **Tools > Add-Ins** > check off **CAMWorks 2014** box in window that pops up > **OK** > **CAMWorks Feature Tree** icon shown.

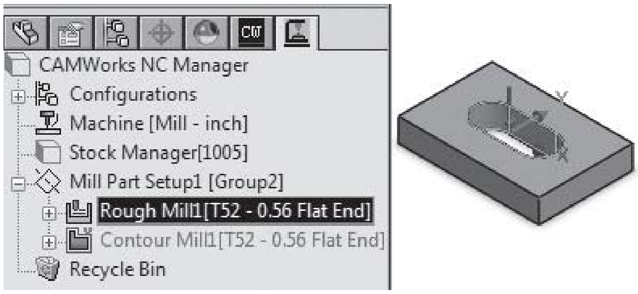


Step 4: Extract machinable features: **Extract Machinable Features** on **CAMWorks 2014** tab.



Note: CAMWorks recognizes the slot feature, extracts it, and adds the **Obround Pocket1 [Rough Finish]** node shown.

Note: Right-click **Mill Part Setup1** shown > check off **Reverse** box that shows up > ✔.

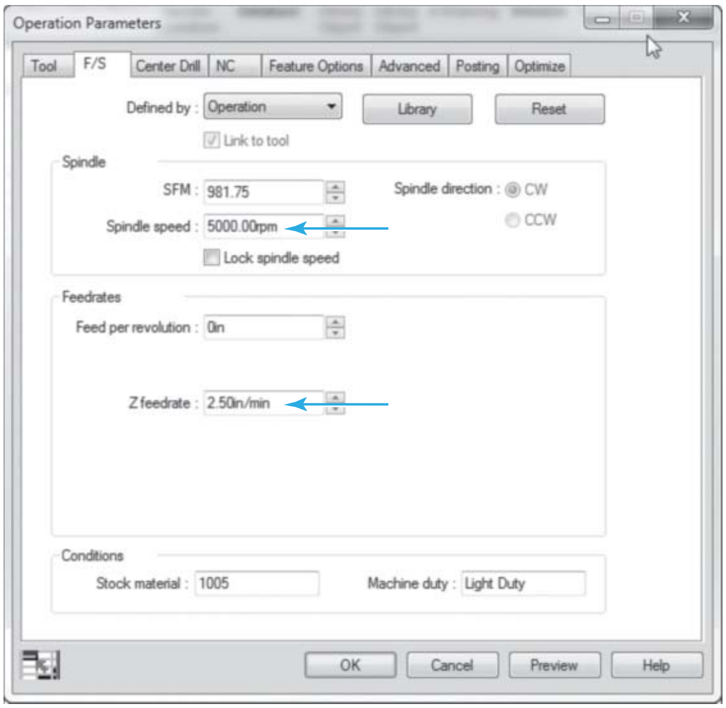


Step 5: Generate operation plan: **Generate Operation Plan** on **CAMWorks 2014** tab.

Note: This step generates the two nodes shown above.

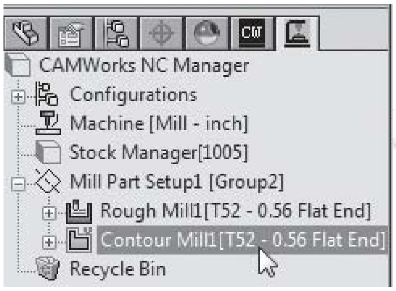
Note: CAMWorks displays tool specs and tool type for each operation.

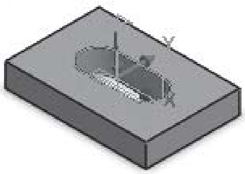
Step 6: Define machining parameters: Right-click **Rough Mill1[...]** > **Edit Definition > Tool** tab in



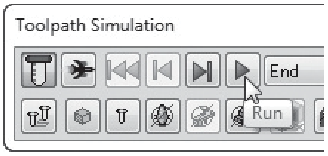
Operation Parameters window that opens up and shown > change **Overall length** to 2 > **F/S** tab > make changes shown > **Posting** tab > **Flood** from **Coolant** dropdown > **OK** > repeat for **Contour Mill1[…]** and use same spindle speed of 5000 rpm and same feedrate of 2.5 in./min > **OK.**

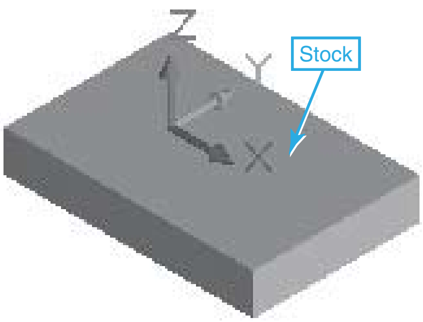
Step 7: Generate toolpath: **Generate toolpath** on **CAMWorks 2014** tab > click a **Mill** node in **CAMWorks Feature Tree** to view its path.

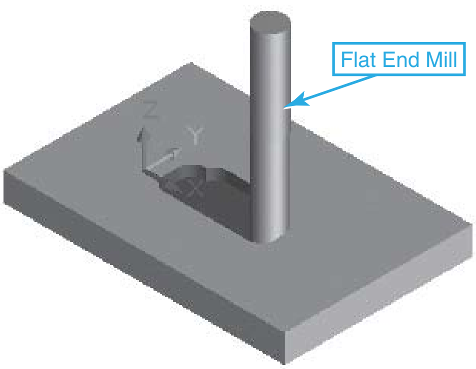


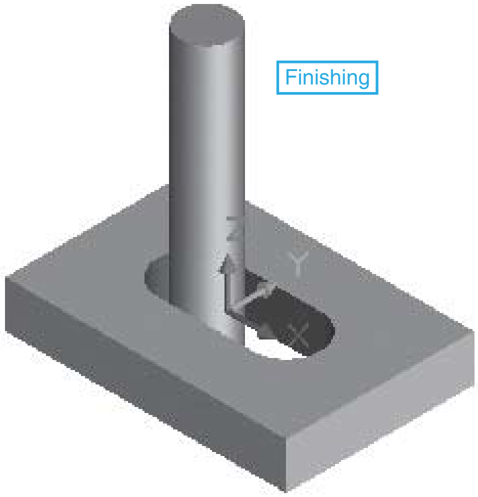


Step 8: Verify toolpath: **Simulate toolpath** on **CAMWorks 2014** tab to open **Toolpath Simulation** window shown below > **Run.**

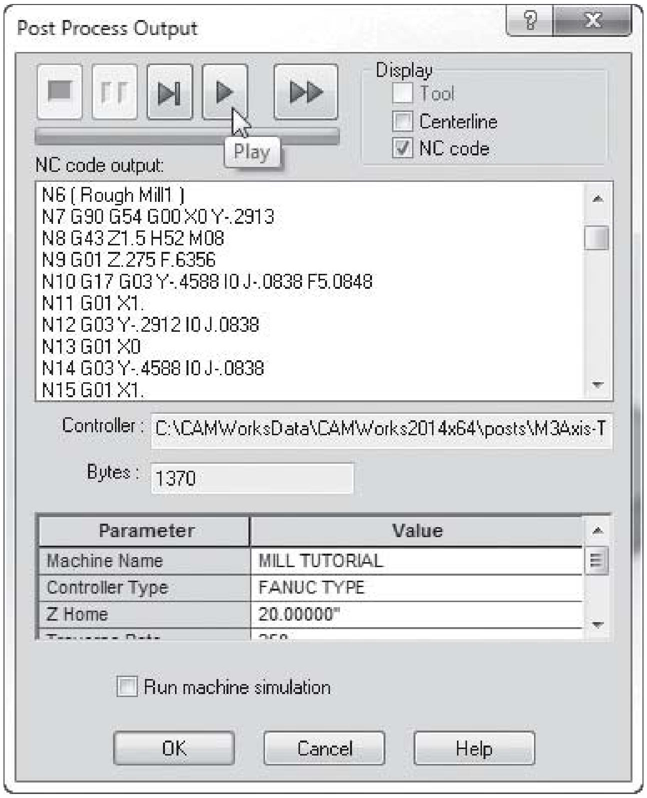








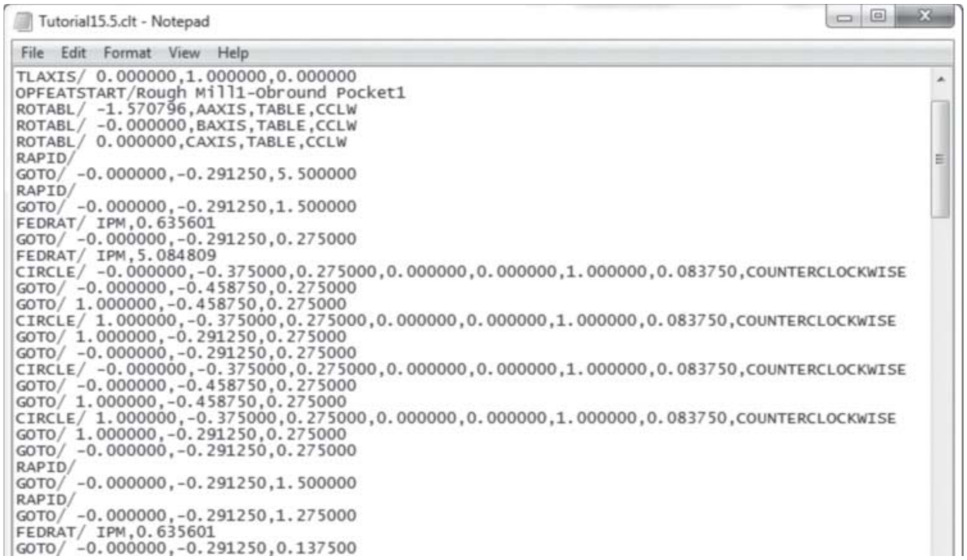
Step 9: Post-process toolpath: **Post Process** on **CAMWorks** tab > accept file (*tutorial15.5.txt*) shown in window that opens up > **Save > Play** button shown to display code > **OK.**



Step 10: Save CL file: **Save CL File** on **CAMWorks** tab > accept file (*tutorial15.5.clt*) shown in window that

opens up > **Save** > open file in Notepad to view.

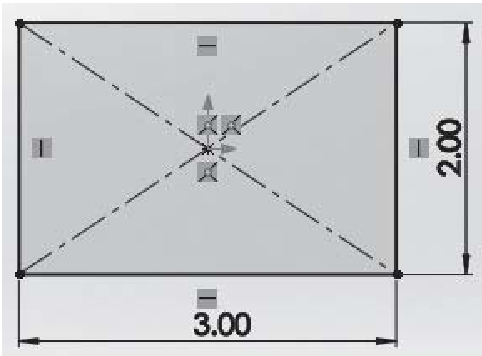
Note: The CL file is not G-code and M-code. It is some form of a higher-level language, similar to APT (not covered here). The *tutorial15.5.txt* file (Step 9) is G-code and M-code.



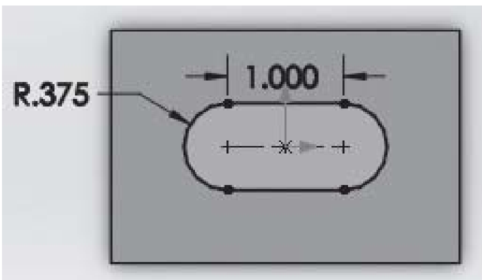
```
TLAXIS/ 0.000000,1.000000,0.000000
OPFEATSTART/Rough Mill-obround Pocket1
ROTABL/ -1.570796,AAXIS,TABLE,CCLW
ROTABL/ -0.000000,BAXIS,TABLE,CCLW
ROTABL/ 0.000000,CAXIS,TABLE,CCLW
RAPID/
GOTO/ -0.000000,-0.291250,5.500000
RAPID/
GOTO/ -0.000000,-0.291250,1.500000
FEDRAT/ IPM,0.635601
GOTO/ -0.000000,-0.291250,0.275000
FEDRAT/ IPM,5.084809
CIRCLE/ -0.000000,-0.375000,0.275000,0.000000,0.000000,1.000000,0.083750,COUNTERCLOCKWISE
GOTO/ -0.000000,-0.458750,0.275000
GOTO/ 1.000000,-0.458750,0.275000
CIRCLE/ 1.000000,-0.375000,0.275000,0.000000,0.000000,1.000000,0.083750,COUNTERCLOCKWISE
GOTO/ 1.000000,-0.291250,0.275000
GOTO/ -0.000000,-0.291250,0.275000
CIRCLE/ -0.000000,-0.375000,0.275000,0.000000,0.000000,1.000000,0.083750,COUNTERCLOCKWISE
GOTO/ -0.000000,-0.458750,0.275000
GOTO/ 1.000000,-0.458750,0.275000
CIRCLE/ 1.000000,-0.375000,0.275000,0.000000,0.000000,1.000000,0.083750,COUNTERCLOCKWISE
GOTO/ 1.000000,-0.291250,0.275000
GOTO/ -0.000000,-0.291250,0.275000
RAPID/
GOTO/ -0.000000,-0.291250,1.500000
RAPID/
GOTO/ -0.000000,-0.291250,1.275000
FEDRAT/ IPM,0.635601
GOTO/ -0.000000,-0.291250,0.137500
```

Generate slot milling toolpath and CL file using Mastercam

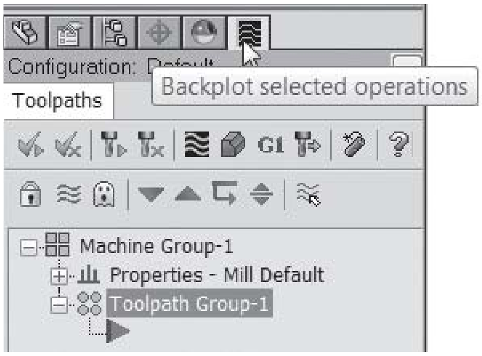
Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK > Top Plane > Extruded Boss/Base** on **Features** tab > **Center rectangle** on **Sketch** tab > sketch and dimension shown > exit sketch > enter 0.5 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As** > *Tutorial15.5* > **Save.**

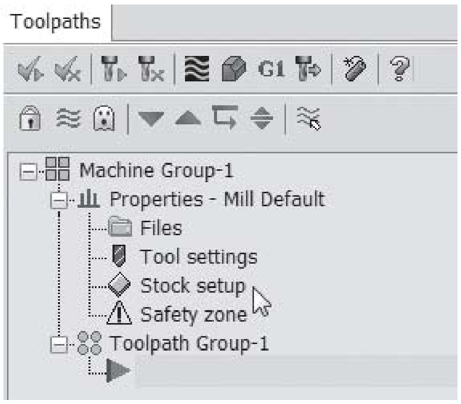


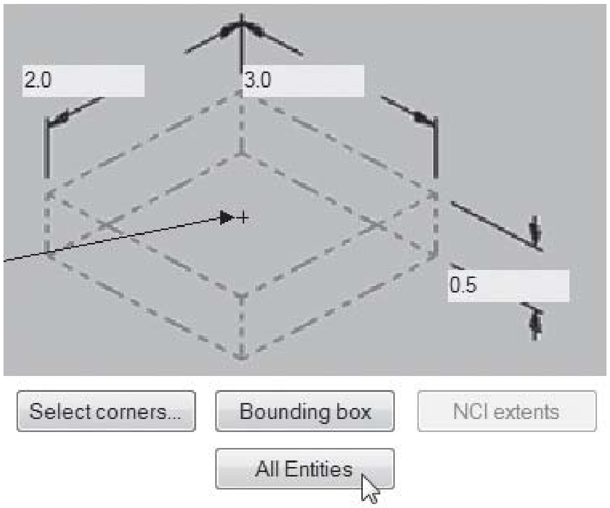
Step 2: Create *Sketch2* and *Cut-Extrude1-Slot*: Top face of block > **Extruded Cut** on **Features** tab > **Centerpoint Straight Slot** on **Sketch** tab > click origin, drag horizontally, click, drag vertically, click > dimension as shown > exit sketch > **Through All** > ✔.



Step 3: Add Mastercam: **Tools > Add-Ins** > check off **Master X7 for SolidWorks** box in window that pops up > **OK > Backplot selected operations** on left pane.



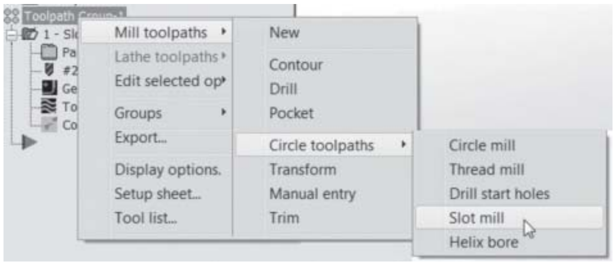


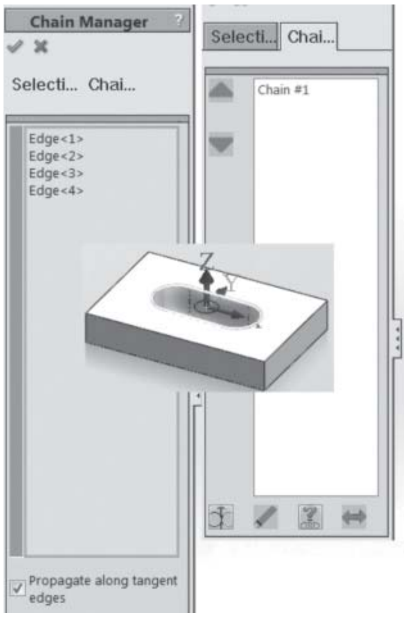


Step 4: Define stock size: Expand **Properties Mill Default** node shown > **Stock setup** > **All Entities** button above > ✔.

Note: Without this step the workpiece display in Step 9 will look like a sheet.

Step 5: Select slot milling operation: Right-click **Toolpath Group-1** shown > **Mill toolpaths > Circle toolpaths > Slot mill.**

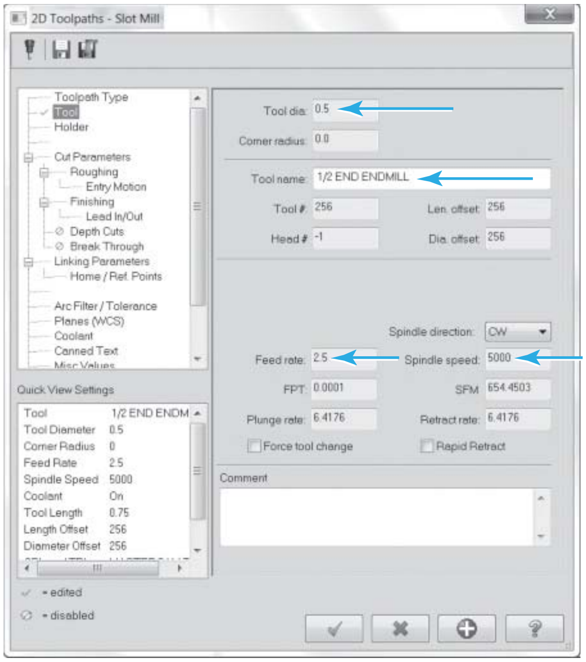




Step 6: Define slot to mill: Edges as shown > ✔ > type *slotmill* for **NC name** as shown above > **OK.**

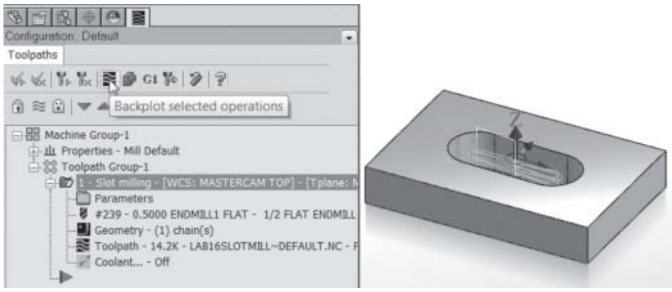
Note: We only select top edges. We specify slot depth vial *Linking Parameters* link shown (click it) in Step 7.

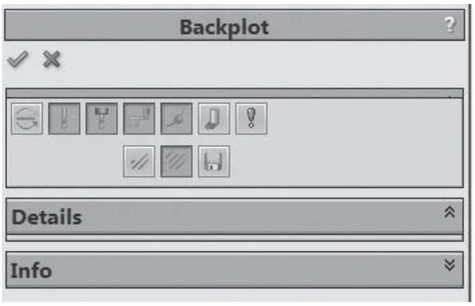
Step 7: Define machining parameters: **Tool** in the **2D Toolpaths** window that opens as shown below > change **Tool dia** to 0.5 > change **Tool name** to *½ END MILL* > change **Feed rate** to 2.5 > change **Spindle speed** to 5000 > **Coolant** > **On** from the **Flood** dropdown > **Linking Parameters** > enter –0.6 for **Depth** > ✔.

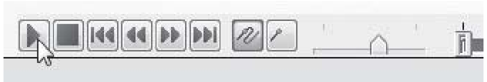


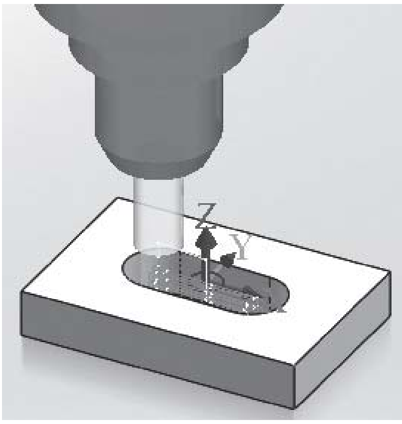
Note: We use –0.6 depth because the slot is 0.5 deep.

Note: The toolpath is generated at the end of this step as shown. A yellow path indicates that the tool is not in contact with the workpiece during motion. A blue path indicates contact.



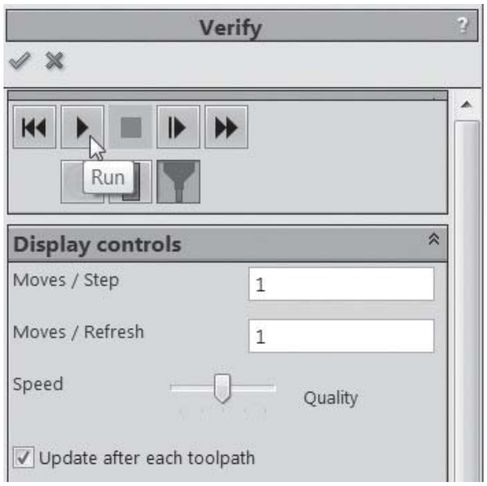


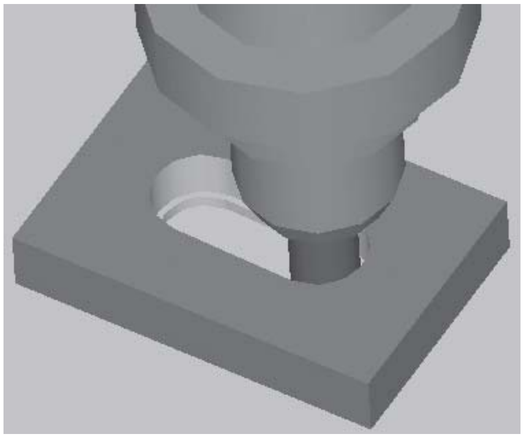




Step 8: Verify toolpath using **Backplot**: **Backplot** icon shown in Step 7 > **Play** shown above > ✔.

Note: Use **Backplot** control bar (shown above) that opens up to control display.





Step 9: Verify toolpath using **Verify**: **Verify** icon (next to Backplot icon shown in Step 7; hover until you read it) > **Run** shown to the left below > ✔.

Note: Investigate pane shown above to learn more.

Step 10: Post-process and save CL file: **G1** icon (shown in Step 7) > ✔ to accept **Post Processing** parameters in window that opens up > type *slotmill* for NC file name in window that opens up > **Save.**

Note: View code generated and shown.

```
N100 G20
N102 G0 G17 G40 G49 G80 G90
N104 T239 M6
N106 G0 G90 G54 X-.5549 Y.0656 A0. S1069 M3
N108 G43 H239 Z.25
N110 Z.1
N112 G1 Z0. F15.
N114 X-.5635 Y.0545 Z-.0007 F25.
N116 X-.5698 Y.042 Z-.0014
N118 X-.5737 Y.0285 Z-.0022
N120 X-.575 Y.0145 Z-.0029
N122 X-.5736 Y.0001 Z-.0037
N124 X-.5695 Y-.0137 Z-.0044
N126 X-.5628 Y-.0265 Z-.0052
N128 X-.5538 Y-.0378 Z-.0059
N130 X-.5428 Y-.0471 Z-.0067
N132 X-.5302 Y-.0541 Z-.0075
N134 X-.5165 Y-.0587 Z-.0082
N136 X-.5022 Y-.0605 Z-.009
N138 X.4978 Y-.0895 Z-.0614
N140 X.5 Z-.0615
N142 X.5146 Y-.0881 Z-.0623
N144 X.5287 Y-.0838 Z-.063
N146 X.5417 Y-.0769 Z-.0638
N148 X.5531 Y-.0676 Z-.0646
N150 X.5624 Y-.0562 Z-.0654
N152 X.5693 Y-.0432 Z-.0662
N154 X.5736 Y-.0291 Z-.0669
N156 X.575 Y-.0145 Z-.0677
N158 X.5736 Y-.0001 Z-.0685
N160 X.5695 Y.0137 Z-.0692
N162 X.5628 Y.0265 Z-.07
```

HANDS-ON FOR TUTORIAL 15–5. Figure 15.19 shows that we used the features approach to create the slot. Create the part using the cross section approach (create the top face of the part shown in Figure 15.19 as one sketch using **Top Plane**). Generate the toolpath. Is there a difference in the tutorial steps or the NC generated code? If yes, explain why.

Tutorial 15–6: Wire EDM a Spline Shaft

This tutorial illustrates using wire EDM machining. We generate the toolpath and CL file to wire EDM the spline shaft shown in Figure 15.20. All dimensions are in inches. Coolant is on.

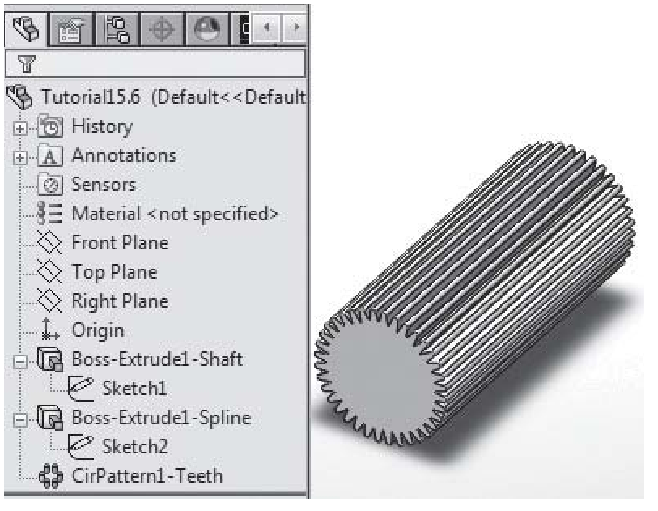
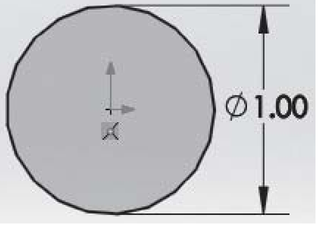
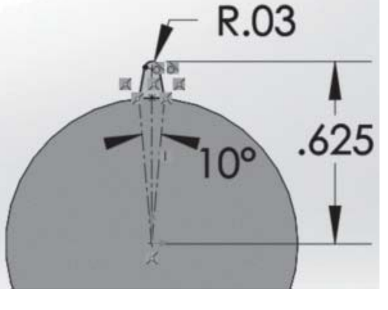


FIGURE 15.20
Spline shaft

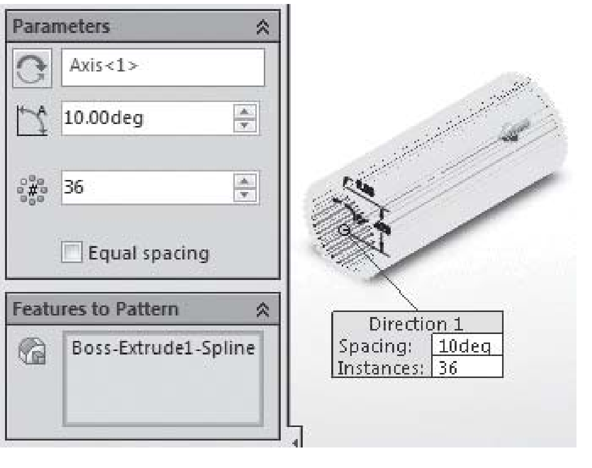
Step 1: Create *Sketch1* and *Boss-Extrude1-Shaft*: **File > New > Part > OK > Front Plane > Extruded Boss/Base** on **Features** tab > **Circle** on **Sketch** tab > sketch and dimension shown > exit sketch > enter 3.0 for thickness **D1** > reverse extrusion direction > ✔ > **File > Save As >** *Tutorial15.6* > **Save.**



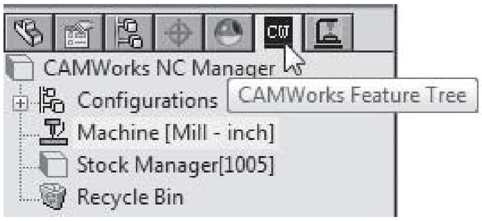
Step 2: Create *Sketch2* and *Boss-Extrude1-Spline*: Front face of shaft > **Extruded Boss/Base** on **Features** tab > sketch and dimension shown > exit sketch > enter 3 for thickness **D1** > reverse extrusion direction > ✔.



Step 3: Create *CirPattern1-Teeth*: **Circular Pattern** on **Features** tab > circle center for axis > *Boss-Extrude1-Spline* for feature to pattern > 36 > ✔.

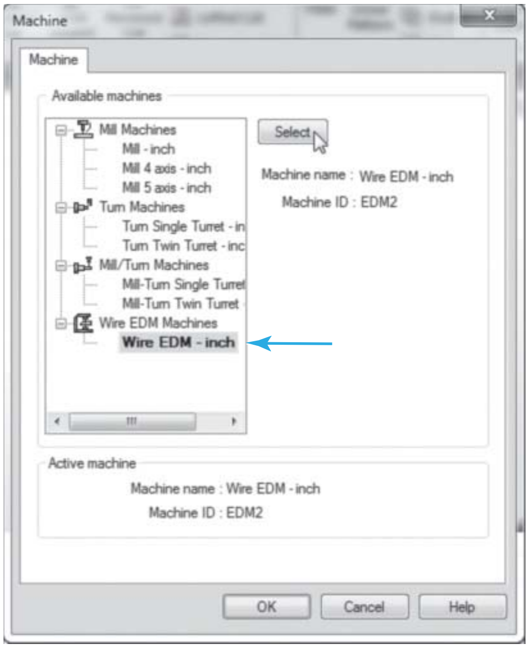


Step 4: Add CAMWorks: **Tools > Add-Ins** > check off **CAMWorks 2014** box in window that pops up > **OK** > **CAMWorks Feature Tree** shown.



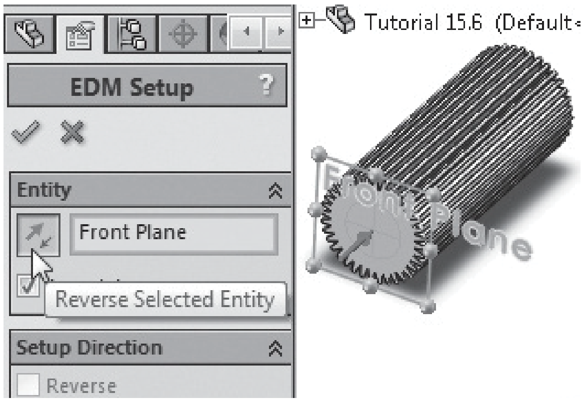
Step 5: Edit default machine: Right-click **Machine [Mill – inch] > Edit Definition.**

Note: We need to change the default milling machine to a wire EDM to machine the spline.



Step 6: Select wire EDM machine: > **Wire EDM – inch** shown above > **OK.**

Note: The **Select** button sets **Wire EDM – inch** as the **Active machine**.

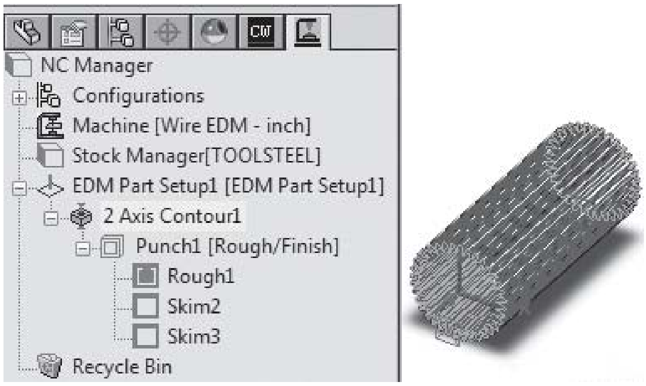


Step 7: Define a new part setup: Right-click **StockManager[TOOLSTEEL]** in CW feature tree > **New Part Setup > Front Plane** from part features tree > click icon shown above to reverse direction > ✔.

Note: This adds **Punch1 [Rough/Finish]** node shown.

Step 8: Extract machinable features: **Extract Machinable Features** on **CAMWorks 2014** tab.

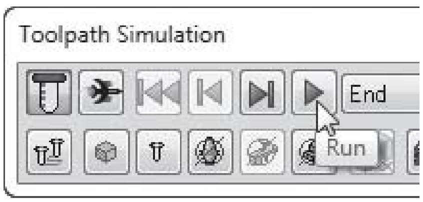
Step 9: Generate operation plan: **Generate Operation Plan** on **CAMWorks 2014** tab.

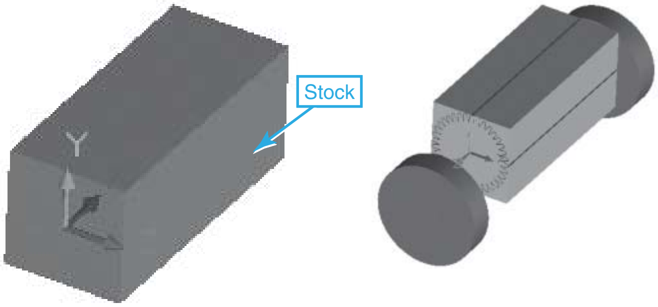


Step 10: Generate toolpath: **Generate toolpath** on **CAMWorks 2014** tab.

Note: This step generates the nodes shown above.

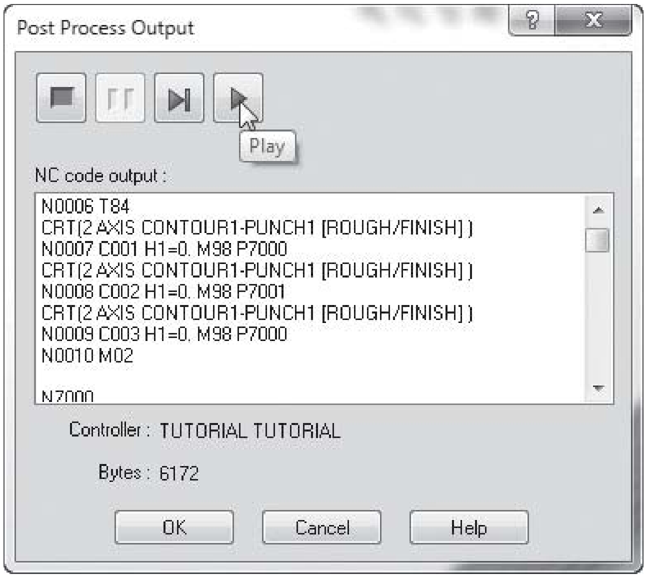
Step 11: Verify toolpath: **Simulate toolpath** on **CAMWorks 2014**





tab to open **Toolpath Simulation** window shown > **Run.**

Step 12: Post-process toolpath: **Post Process** on **CAMWorks** tab > accept file (*tutorial15.6.txt*) shown in window that opens up > **Save > Play** button shown to display code > **OK.**



problems

1. What is the difference between mass production and mass customization?
2. What types of parts can turning and milling operate on, that is, machine?
3. What does the ½-axis designation mean for a machine tool? Give an example.
4. List the materials of machining cutting tools. Which is the softest and which is the hardest material? What is the effect of material strength on the tool operating parameters?
5. Flutes (teeth) and end types characterize cutting tools. Provide examples for drilling and milling tools.
6. What does the 3/8 • 16UNF thread designation mean?
7. List and sketch six of the common stock shapes.
8. What are the machining parameters that we typically use in machining? Show them on sketches of turning and milling operations.
9. What is the difference between roughing and finishing machining operations?
10. Explain the concept of workpiece squaring. Why is squaring important?
11. What is machine zero? Why is it important?
12. What is a burr? Why does it occur?
13. Sketch PTP and continuous machining. What is the main difference between the two types?
14. Research the turning operations. Submit a detailed report about turning machines, turning centers, how many axes are available, turning operations (boring, reaming, threading, and others), turning cutting tools (their materials, machining parameters, etc.), speeds, feedrates, cutting depth, jigs, and fixtures used to clamp workpieces, and sample parts.
15. Redo Problem 14 but for drilling.
16. Redo Problem 14 but for milling.
17. Redo Problem 14 but for sinker EDM.
18. Redo Problem 14 but for wire EDM.
19. Explain the main difference between DFA and DFM.
20. What does zero-radius NC programming mean? What is its benefit?
21. What does tool offset in NC programming mean? What is its benefit?
22. Use SolidWorks **DFMXpress** to check the manufacturability of the part shown in Figure 15.21A. All dimensions are in inches. Note: This is a turning operation.

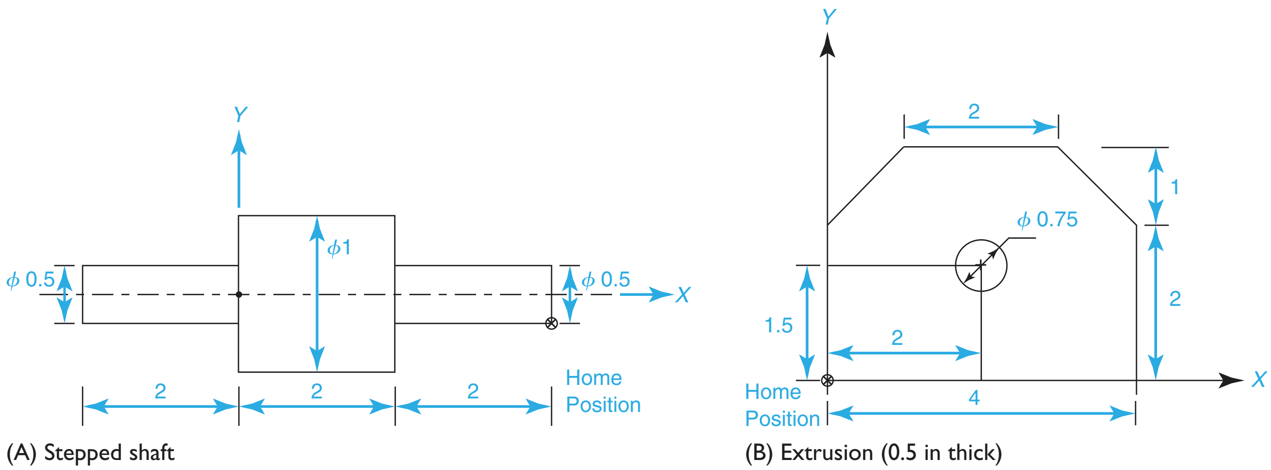


FIGURE 15.21
Two parts

23. Redo Problem 22 but for the part shown in Figure 15.21B. All dimensions are in inches. Note: This is a drilling operation.
24. Redo Problem 22 but for the AMP connector model shown in Figure 2.30 of Chapter 2.
25. Redo Problem 22 but for the flange model shown in Figure 2.32 of Chapter 2.
26. Redo Problem 22 but for the L bracket shown in Figure 2.35 of Chapter 2.
27. Redo Problem 22 but for the pattern block shown in Figure 2.36A of Chapter 2.
28. Redo Problem 22 but for the pattern wheel shown in Figure 2.36B of Chapter 2.
29. Redo Problem 22 but for the loft feature shown in Figure 4.12 of Chapter 4.
30. Generate the G-code, M-code, and CL file for turning the shaft shown in Figure 15.21A.
31. Generate the G-code, M-code, and CL file to machine the part shown in Figure 15.21B.
32. Generate the G-code, M-code, and CL file to machine the AMP connector model shown in Figure 2.30 of Chapter 2.
33. Generate the G-code, M-code, and CL file to machine the flange model shown in Figure 2.32 of Chapter 2.
34. Generate the G-code, M-code, and CL file to machine the L bracket shown in Figure 2.35 of Chapter 2.
35. Generate the G-code, M-code, and CL file to machine the pattern block shown in Figure 2.36A of Chapter 2.
36. Generate the G-code, M-code, and CL file to machine the pattern wheel shown in Figure 2.36B of 2.
37. Generate the G-code, M-code, and CL file to machine the loft feature shown in Figure 4.12 of Chapter 4.

38. Generate the G-code, M-code, and CL file to machine the part shown in Figure 15.22. All dimensions are in inches.

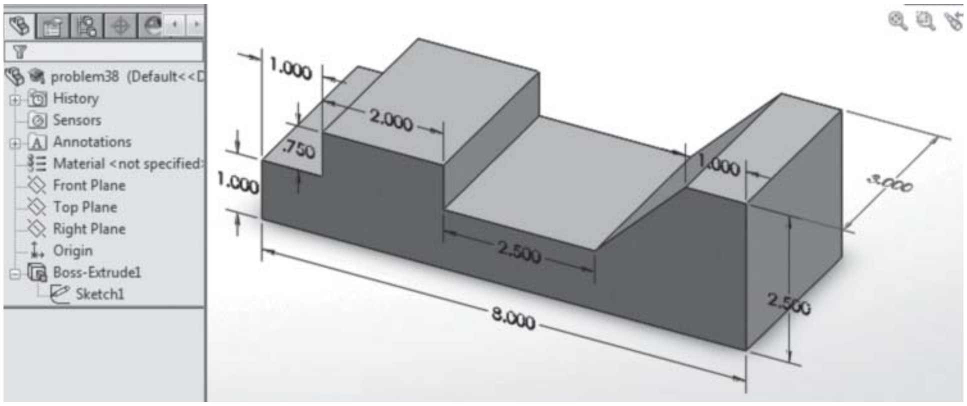


FIGURE 15.22
An extrusion

39. Generate the G-code, M-code, and CL file to drill the two holes shown in Figure 15.23. All dimensions are in inches.

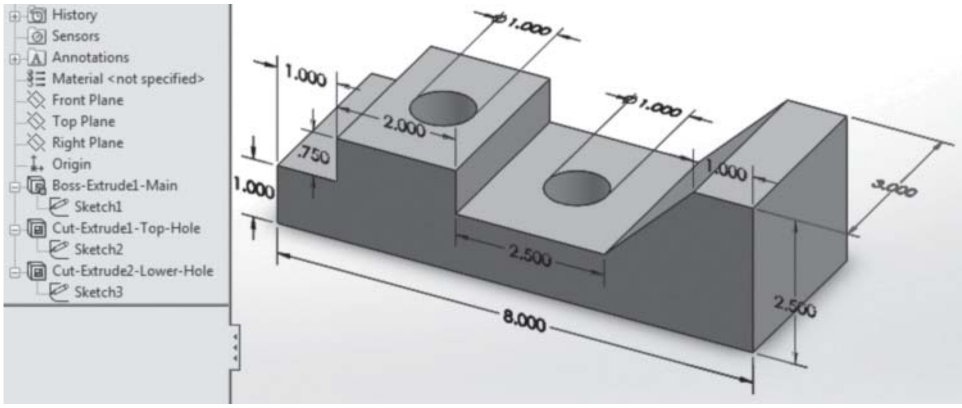


FIGURE 15.23
An extrusion with holes

40. Generate the G-code, M-code, and CL file to machine the pocket and slot shown in Figure 15.24. All dimensions are in inches.

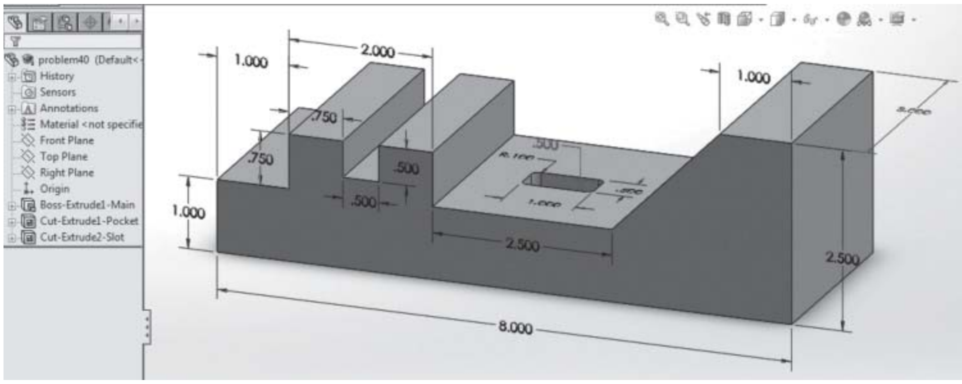


FIGURE 15.24
An extrusion with pocket and slot

CHAPTER

16

Injection Molding

16.1 Introduction

Machining can only produce products that have surfaces that are accessible by machine tools. These are surfaces that do not have complex shapes and/or do not include geometry that is hard to reach on machine tools. Injection molding is a manufacturing process that is capable of producing any shape made of materials such as plastics or polymers. Many plastic parts such as bottles, toothbrushes, bottle caps, car parts, and wires are injection molded. Injection molding is widely used in various industries including medical, consumer, automotive, health care, and toys. Injection molding is ideal for mass production due to the fast production cycles, and as a result the cost per item is very inexpensive.

Injection molding is a cyclic process whereby molten plastic is formed into a desired shape by forcing the plastic into a cavity under pressure. Sizes of molded parts vary from the smallest size components used in medical devices, to entire body panels of cars. Injection molding produces excellent dimensional tolerances. It also requires minimal or no finishing or assembly operations, making it ideal for fast delivery of finished products from the time an order is received.

Thermoplastic materials are the most commonly used for molded parts. These materials (also known as *resins*) include polystyrene, polyamide, polypropylene, polyethylene, polyvinyl chloride (PVC), and acrylonitrile butadiene styrene (ABS). Material selection depends on the end application and the desired strength/ductility of the resin. Another group of materials includes ceramics, metals, and hard metals. They are used in powder injection molding to produce large volumes of small, complex parts. Powder injection molding combines two manufacturing processes: injection molding and sintering.

The injection molding process requires three elements: an injection molding machine, plastic material, and a mold. The plastic is melted in the machine and injected into the mold where it cools and solidifies into the final part. The process is characterized by very large forces and very short processing time. Injection pressure that presses the molten material into the mold is very high, ranging from 3000 to 40,000 psi, with a typical value of 15,000 psi. The clamping force that holds the mold together and the ejection force that ejects the part from the mold are usually very high. The processing (cycle) time to produce one molded part is very short, typically in the seconds.

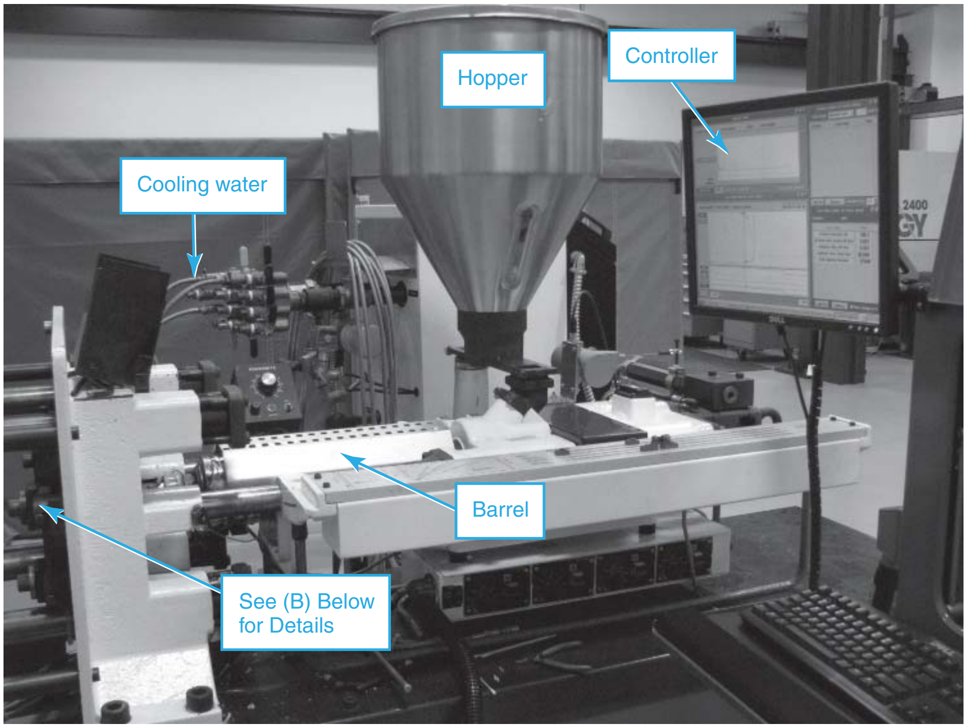
Injection molding has advantages and disadvantages. Its advantages are (1) reduced material handling, as the hopper holds the plastic pellets; (2) creating parts with metal inserts easily; (3) producing parts with tight tolerances; and (4) lower cost per part, resulting from highly automating the injection molding process. The disadvantages of injection molding are (1) warpage of the parts could result if the mold design and the molding process are not optimized; (2) weak parts at connection lines (known as knit lines) where they are likely to break; and (3) parts could become scrap if the mold is not designed properly. Mold design is usually an intricate and expensive task.

16.2 Basics of Injection Molding Machines

Injection molding requires injection molding (IM) machines, just as machining requires machine tools. With an appropriately designed mold that can be accommodated in the IM machine, it can make a variety of parts similar to a machine tool. We need to understand both how IM machines work and what the basics of IM are. This section covers the basics of IM machines. Subsequent sections cover the basics of IM. Figure 16.1 shows a sample IM machine. IM machines come in either a horizontal or vertical configuration. Horizontal configurations are common (Figure 16.1).

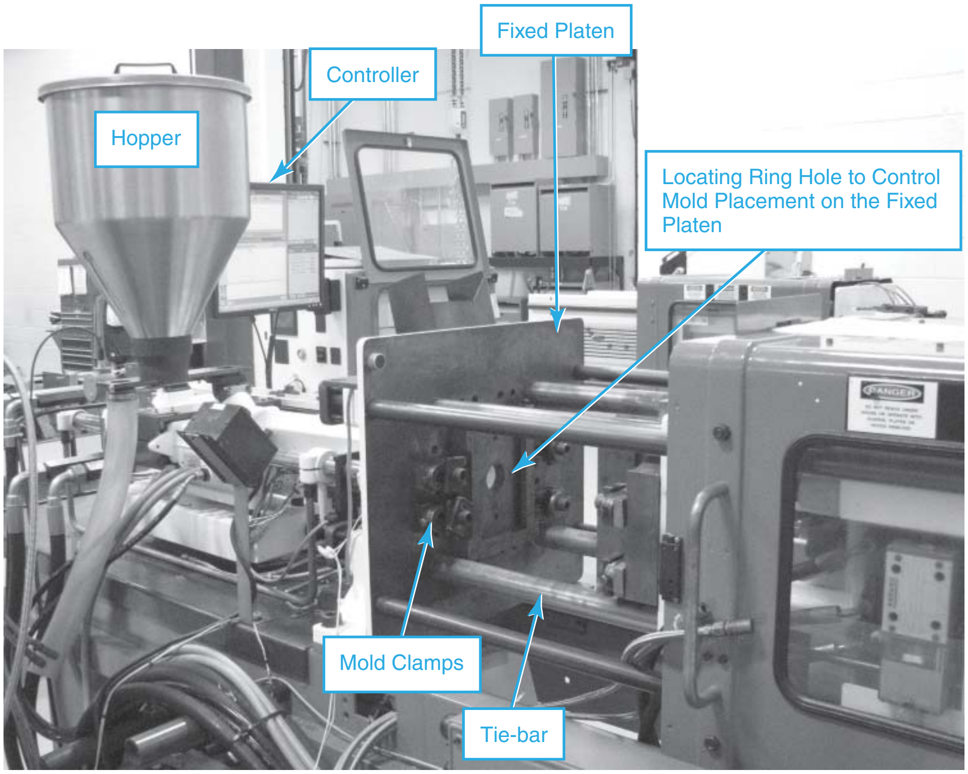
Figure 16.2 shows an abstraction (schematic) of the IM machine shown in Figure 16.1. The abstract machine helps us understand the machine operations. An IM machine has a solid frame that is usually long and skinny, as shown in Figure 16.1. The machine has a controller to operate it. The machine does not have too many externally visible moving parts.

An IM machine has three main sections as shown in Figure 16.2: injection unit, mold assembly, and clamping unit. The injection unit receives the plastic pellets from the hopper, heats up the plastic (plastic becomes plasticized, i.e., changes from a solid to a semi-viscous state), and delivers it to the mold assembly. The mold assembly section contains the mold that makes the part, which has two halves: the cavity and the core. The clamping unit holds the mold tightly closed during the injection of the plastic into the mold.



(A) Close-up view of the injection unit

FIGURE 16.1
An injection molding machine
(Courtesy of NyproMold Inc.)



(B) Close-up view of the mold space

FIGURE 16.1
(*continued*)

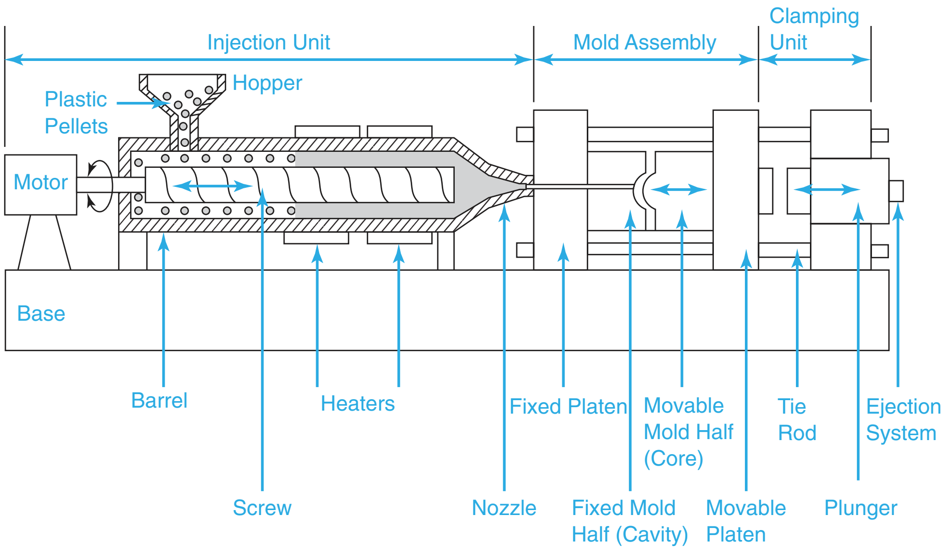


FIGURE 16.2
Schematic of an injection molding machine

16.3 Basics of Injection Molding

The IM process to make one part is called a *cycle*. The cycle has four stages in this order: clamping, injection, cooling, and ejecting. The cycle begins by the clamping unit securing the mold halves in place before plastic injection begins. The injection stage begins with the barrel receiving plastic pellets from the hopper (see Figure 16.2).

The screw inside the barrel rotates and slides axially via the motor and pushes the pellets forward inside the barrel. When they reach the heaters area, they melt and become molten plastic.

The screw continues pushing the plastic into the nozzle. The nozzle feeds into the mold cavity and fills it with plastic that takes the shape of the mold cavity. The clamping unit holds the mold halves tightly and presses the core into the cavity firmly to prevent any leak of the molten plastic. As soon as the plastic touches the mold, it cools down immediately. Once the cooling down is complete after a preset cooling period, the screw retracts, the clamping unit unclamps, and the ejectors eject the final part, releasing it from the core. The part is dropped to the bottom of the IM machine into a bucket or container. Figure 16.3 shows the IM cycle.

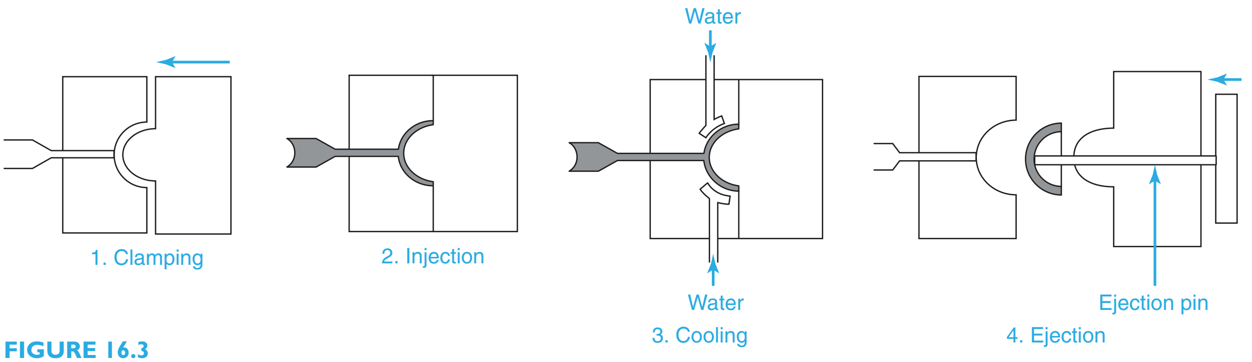


FIGURE 16.3
Injection molding cycle

Each cycle produces one or more parts and is repeated at a high rate. The screw is reciprocating: It pushes the plastic at the beginning of the cycle and retracts at the end of the cycle to allow more plastic to flow into the barrel for the next cycle. The cooling is typically done with water. The IM machine is connected to water pipes that circulate the water.

The mold halves are fixed to large plates called *platens*. The fixed (front) half of the mold has the cavity (also called the *mold cavity* or *A half*) and is mounted to the stationary (fixed) platen. The movable (rear) half of the mold has the core (also called the *mold core* or *B half*) and is mounted to the movable platen. The plunger (Figure 16.2) of the clamping unit holds the mold securely closed while the material is injected. As soon as the material hits the mold, it cools down. After the required cooling time, the mold is opened and the ejection system pushes the solidified (final) part out of the mold. The ejection system is usually attached to the movable (rear) half of the mold.

The plastic parts come out of the IM machine ready for use. However, some minor cleaning (post-processing) may be needed to break off extra plastic pieces. For example, if the mold is used to make multiple parts, these parts become connected by runners (branches) of plastic that must be broken off the parts (via using a cutter). These runners represent the “chips.” They can be recycled by melting and converting them into pellets for reuse. The recycling includes placing the chips in plastic grinders and grinding them into pellets (known as the *regrind*). Due to the change in material properties after injection, the regrind is always mixed with raw material in the proper ratio. The new mix is then used to make new molded parts (when permissible).

Defects are an expected part of manufacturing processes, and IM is no exception. Molding defects are caused by the mold itself (due to poor design) or the molding process. For example, defects may be caused by a nonuniform cooling rate due to mold nonuniform wall thickness or temperature. Possible defects are:

1. Sink marks: These marks are caused by low injection pressure and/or mold nonuniform wall thickness, causing some sections of the mold to solidify quicker than others, leading to uneven shrinkage and leaving temporary voids in the mold.
2. Bubbles: These are an indication that the injection temperature is too high or there is too much moisture in the plastic. These bubbles are visible on the part's outer surfaces.
3. Short shot (partial filling): Some sections of the cavity may be unfilled because the shot size is insufficient or the plastic flow in the mold is slow, causing it to solidify before reaching the far ends of the mold, away from the gates.
4. Ejector marks: These are small indentations made in the part where the ejection system pushed the part out of the mold. They are an indication of too short a cooling time or excessive ejection force.
5. Warpage: Some sections of the mold may cool faster than others, resulting in warping of the part. Warpage is an indication that the cooling rate is nonuniform inside the mold.
6. Flashing: The molten plastic may seep out of the mold cavity between the mold halves and solidify along the parting line as a thin layer of plastic. This layer is visible after ejecting the part. It is an indication that the clamping force is too low or the injection pressure is too high.
7. Burn marks: These are black or brown burned areas on the part's surface. They are an indication of improper venting of the tool (mold) or that the injection temperature is too high.
8. Flow marks: These are directional "off tone" wavy lines or patterns. They are an indication of low injection pressure.
9. Voids: These are hollow spots (air pockets) inside the part, indicating a lack of holding the injection pressure at the proper level during the entire injection (shot) time.
10. Weld lines: These are discolored lines where two flow fronts meet. They indicate low temperatures of the mold and the melt when the two fronts meet, so they do not bond well into each other.

16.4 Basics of Mold Design

Mold designers design molds, and tool (mold) makers make them. Mold designers use mold design software to create molds and toolmakers use precision-machining techniques to produce the molds. Mold machining combines several machining techniques including turning, milling, grinding, EDM, and polishing. We focus only on mold design here. The following concepts enable us to understand mold design and effectively use mold design software.

1. Types of molds: There exist many types of injection molds. We can classify them in multiple ways as follows:
   A. Based on the number of cavities, we have a single- or a multi-cavity mold.
   B. Based on construction, we have 2-plate, 3-plate, side-action, or stack molds.
   C. Based on ejection, we have pin-eject or stripper-eject molds. Pin ejection occurs with pins, as shown in Figure 16.3. Stripper ejection uses a stripper that pushes the part along its perimeter (think of the rim of a cup).
   D. Based on the type of runner (feed) system, we have hot runner mold, cold runner mold, or hot to cold runner mold.

2. Core and cavity: These are the two halves of the mold. The cavity represents the "imprint" of the outside of the part, and the core represents the "imprint" of the inside of the part. The cavity half (mold cavity) is always attached to the fixed (stationary) platen of the IM machine, and the core (mold core) is attached to the movable platen. We use the concept of cavity and core in mold design to be able to extract the molded (final) part easily out of the mold. Because the mold needs to be pulled apart to remove the part, the mold design should ensure this separation.

   Industry uses jargon to refer to mold halves. The A (also known as *fixed* or *cavity*) half is the mold half that does not move (Figure 16.2) and the B (also known as *movable* or *core*) half is the mold half that moves before and after injection (Figure 16.2). The B half moves before injection to clamp the mold and moves after injection to allow the removal (extraction) of the molded part.
3. Tool splitting: Tool splitting is synonymous with core and cavity. Tool splitting is the process by which we split the mold into two halves: core and cavity. After the split, designer typically adds the gates, runner system, ejection pins, and fasteners.
4. Mold components and assembly: A mold consists of its core, cavity, and housing. The cavity and the core are the two main components. The mold **housing** is its base that houses all the mold components. These components are alignment components, feed system, cooling system, and ejection system. The housing consists of plates to affix the mold components to. The mold assembly consists of two subassemblies. The core and cavity subassembly has the core and cavity only. The housing (base) subassembly has all the other components of the mold.
5. Shot: A shot is defined as the amount of plastic that is injected to make one part. Each IM cycle requires one shot.
6. Shrinkage: Plastic expands when it is hot and shrinks when it cools down. After a shot is delivered to the mold cavity, it cools down immediately. If its size does not account for shrinkage, the molded part will be smaller than desired. Thus, the mold design should account for shrinkage. Mold analysis can help estimate the amount of shrinkage (shrink factor) and calculates the correct shot size accordingly. The shrink factor is based on the type of plastic and the mold conditions.
7. Cooling time: This is the correct amount of time needed to solidify plastic inside a mold cavity before opening. Inaccurate calculation of cooling time results in defective parts. Mold analysis provides equations to calculate this time. Cooling channels are usually built into the mold design to allow water to flow through the mold walls near the cavity to cool down the molten plastic.
8. Ejection: Ejection is the last activity in the IM cycle. We use an ejection system to push the part out of the mold core when we open the mold. When the plastic cools down and solidifies at the end of the cooling time, it adheres to the mold core, in which case a force must be applied to eject the part. An ejection system is part of the mold design.
9. Tooling: Tooling (or custom tooling) refers to molds. A mold is typically made out of steel or aluminum.
10. Parting line, surface, and axis: A **parting line** is the line of demarcation along the part outside. It indicates where the mold halves separate. It separates the core from the cavity. The parting line can take different shapes ranging from a simple line to a piecewise line, to a curve. The parting line is a closed entity regardless of its shape.

    A **parting surface** is the surface where the two mold halves contact each other. Parting surfaces may be flat, stepwise planar, angled, or curved (free-form). Molds with flat parting surfaces are less expensive to make than others.

    A **parting axis** (direction) is the axis along which the two halves of a mold will separate to allow the part to be ejected.

11. Mold base: These are the plates that the mold cavity and core are fixed to. The cavity and the sprue are attached to the stationary (front) plate. The core and the ejection system are attached to the movable (rear) plates, as shown in Figure 16.2. The ejection system has ejector pins that are actuated to push the part off the core when the mold opens up.
12. Gate and runner system: This is the system that distributes the molten plastic from the nozzle shown in Figure 16.2 to the mold. The gate represents the entry point of injection of the plastic into the mold. The runners are the channels that carry the plastic from the sprue to the gates of the mold cavity. An important issue is to make sure that enough molten plastic can be pushed through the gates of the mold to avoid defective parts. Also, if the plastic flow is not fast enough, the leading edge (front) of the plastic could solidify, preventing the plastic from reaching the entire mold cavity and causing defects as well. The design of the gates (how many, best locations, and the sizes of the orifices) and the runners (how many and how long) is important to successful injection molding.

    Figure 16.4 shows a schematic of a gate and runner system. The main channel that feeds plastic to the runners from the barrel is known as the *sprue*. If we use only one gate, there is no need for a runner system. The sprue feeds directly to that gate, as shown in Figure 16.4A. Figure 16.4B shows a runner system where

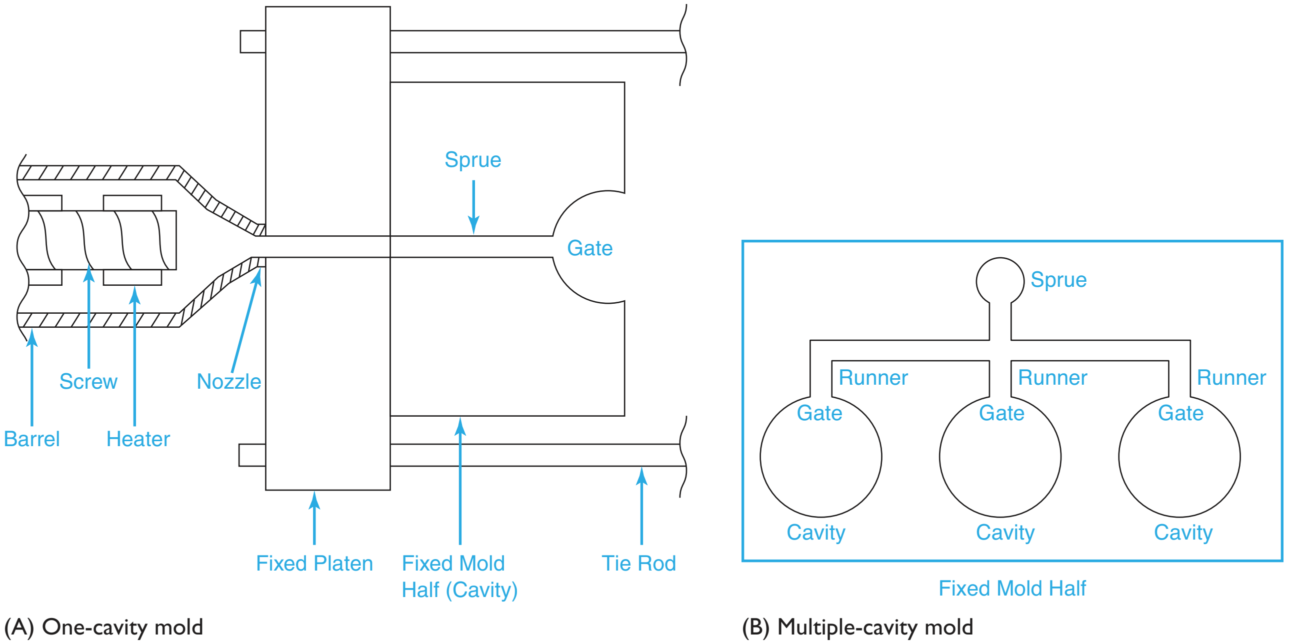


FIGURE 16.4
Schematic of a gate and runner system

the mold has three cavities to make three parts in one injection cycle. In such a case, the sprue feeds to the runners and the runners feed to the gates. The sprue and the runners are always in perpendicular planes; the sprue is always vertical (in the middle of the mold A half) and the runners are always horizontal. In this configuration, the runners feed to the cavities from the side (horizontal). Think of Figure 16.4B as a right view of Figure 16.4A. The molten plastic inside the sprue and the runners solidifies and must be ejected along with the part.
13. Tooling cost: The tooling cost is the total cost of the two mold halves and the mold base. The part size determines the mold size, and the mold size, in turn, determines the size of the base. The larger the mold base, the more expensive it is. As

for the mold itself, its cost depends primarily on the size and shape of its cavity or if multiple cavities are needed. The cavity machining depends on the complexity of the part's geometry. Additional machining cost could accrue from machining parting surfaces, high tolerances, and high surface finish.

14. Draft: **Draft** is a slight taper on selected model faces (walls) that facilitates removal (ejection) of the part from the mold without distorting or damaging it. The draft angle is specified such that the opening of the mold cavity is wider than its base. The draft taper is specified by an angle value, typically 1 to 5 degrees.
15. Undercut: An **undercut** is an optional feature of a molded part to help keep the part secured to the core side, that is, the moving half (where the ejection system is).
16. Insert: Sometimes designing a one-piece core or one-piece cavity is not feasible if the mold is too large. Thus, we split the core and cavity into pieces, each called an insert. Inserts simplify the mold cost, maintenance, and replacement.
17. Venting: Venting provides a path for the air to escape from the cavity as the melt displaces it. Inadequate venting of the cavity can seriously reduce the melt flow to it. Venting affects the quality of injection molded parts. A vent should always be located opposite to a gate, as this will be the point of final fill. Venting prevents "burn mark" defects.

16.5 Basics of Part Design

Successful production of injection molded parts has two requirements: good mold design and good part design. Without both of them, molded parts may be defective. Section 16.4 covers the basics of mold design. The following guidelines serve as the basics for good part design:

1. Part wall thickness: Keep it uniform and to a minimum to reduce the part volume and shorten the injection time.
2. Part edges and corners: Avoid sharp edges and corners. Fillet the edges and round the corners.
3. Drafts: Apply drafts to the part walls that are parallel to the parting axis (direction) to facilitate the part's removal from the mold.
4. Ribs: Add ribs to strengthen the part where needed instead of increasing the part wall thickness. Do not violate the first rule listed above. Also, always add ribs so that they are perpendicular to the axis about which bending may occur.
5. Bosses: Bosses should be supported by ribs that connect to the nearest part wall.
6. Undercuts: Minimize the number of undercuts as much as possible. For example, relocating the parting line and/or redesigning a part feature may eliminate some undercuts.
7. Threads: Orient part features with external threads to be perpendicular to the parting axis.

16.6 Phases of Mold Design

According to industry practice, the three major phases of a mold design are in this sequence:

1. Part design: This is the part design that we are accustomed to and is covered in this book.
2. Prepare part for mold design: The designs of parts that are to be produced by injection molding must go through thorough evaluation by mold designers before

they are approved for mold design. This is because mold designers must ensure that the corresponding molds are possible to make and that they would work correctly. The evaluation process includes draft analysis, undercut analysis, and parting line analysis. The draft analysis ensures that the final part can be ejected from the mold. The undercut analysis is needed to ensure that the part can be safely ejected. The parting line analysis allows the designer to investigate multiple pull directions and select the best one.

Another important mold preparation activity is the flow and temperature analyses of the plastic inside the mold core and cavity. Commercial software tools such as Moldflow exist to aid designers in performing the calculations. Flow and temperature analyses inside a mold are a highly nonlinear complex problem that is solved using nonlinear FEM/FEA. The designer must have strong theoretical background to be able to model the flow problem and understand and interpret the analysis results generated by the software. Designers may display these results as contours or gradients to analyze them both visually and numerically.

3. Mold design: After the part design is tested, modified, and approved, a mold designer begins the mold design process. Designers follow the design concepts covered in Section 16.4 and use mold design software such as SolidWorks **Mold Tools**. They begin by scaling the model up using the plastic shrink factor to accommodate for the cooling effect of the plastic. Following that, the designer creates the parting line, shut-off surfaces, and parting surfaces. Once finished, the designer splits the tool (mold) to generate the two halves (core and cavity). The details of these activities and their explanation are covered in Sections 16.7 and 16.8 as applied to SolidWorks.

   Although the core and cavity are important components of a mold, other components are just as important or more important. These are the mold plates, alignment and clamping components, runner system, gates, ejection system, cooling system, and electrical systems.

16.7 SolidWorks Mold Design

SolidWorks implements the concepts of mold design in its mold design module. In addition, it uses the following concepts:

1. Tooling split: This technique is used to split a block that encloses the CAD model to create the core and cavity. Tooling split uses the parting line, parting surfaces, and shut-off surfaces to split the block.
2. Shut-off surfaces: These surfaces are needed to close any open surfaces with holes in them in order to be able to separate the core and cavity. Surfaces that do not have holes do not require shut-off surfaces. For example, a block with a center hole requires a shut-off surface.

SolidWorks helps designers design and validate molds. SolidWorks supports the three phases of mold design covered in Section 16.6. However, SolidWorks does not provide any mold analysis tools such as Moldflow. Interested readers need to use STEP to exchange models between SolidWorks and Moldflow or other mold analysis software. SolidWorks offers integrated mold design tools that control the mold creation.

Figure 16.5 shows the toolbar, which has four groups: surfaces, preparation, analysis, and tooling. The surfaces group provides different surfaces to help create parting surfaces and shut-off surfaces. We have covered all these surfaces in the book. The preparation group includes all the work needed before using the tooling split. The icons

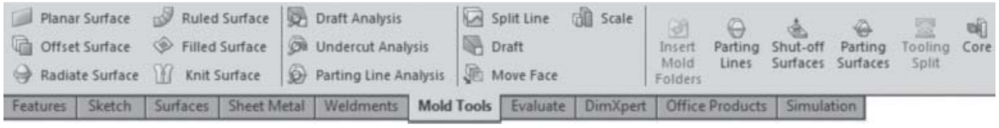


FIGURE 16.5
SolidWorks Mold Tools toolbar

of this group are split line, scale, parting lines, shut-off surfaces, and parting surfaces. The analysis group includes the draft, undercut, and parting line analyses. The tooling group includes the tooling split and core.

The mold design process in SolidWorks begins with the CAD model and ends with the mold. This mold design process uses the following concepts:

1. Enclose and center the CAD model inside a block (tooling block).
2. Remove (subtract or carve cut) the CAD model from inside the block.
3. Split the block open into two halves (core and cavity). We split the block at the parting surface perpendicular to the parting axis. The parting surface is defined by the parting lines and the shut-off surfaces.

The steps of this mold process are as follows:

1. Create the CAD model of the part to be molded
2. Insert mounting bosses if needed
3. Apply shrink factor (scaling)
4. Add draft
5. Create parting line
6. Create shut-off surfaces, if needed
7. Create parting surfaces
8. Apply tooling split
9. Separate the core and cavity
10. Create the tooling assembly

The draft in Step 3 ensures model ejection from the mold. The requirements of such a draft are:

- All faces must draft away from the parting line, which divides the core from the cavity.
- Design specifications must include a minimum draft angle to test against.
- Cavity side surfaces must display a positive draft.
- Core side surfaces must display a negative draft.
- All surfaces must display a draft angle greater than the minimum specified by the design specifications.
- No straddle faces must exist.

Steps 1 through 9 create a multibody part file, which maintains the mold design in one location. Changes to the CAD model are automatically reflected in the tooling bodies. Now, we need to create an assembly where we can add other supporting mold components such as gates, runner system, and ejecting pins. Step 10 creates a mold assembly file.

16.8 Tutorials

The theme for the tutorials in this chapter is to practice using mold design concepts implemented by SolidWorks **Mold Tools**. The tutorials show how to design and document molds.

Tutorial 16–1: Create a Block Mold

We create the injection mold for the plastic block shown in Figure 16.6. All dimensions are in inches.

There are multiple issues to consider when designing a mold. We consider the following issues for this tutorial:

- ☐ Orientation of mold halves: horizontal or vertical. We use the vertical orientation here, meaning the mold halves move up and down, not left to right, in the CAD model, as shown in Figure 16.6. In practice, we use a horizontal injection mold machine to produce the part.

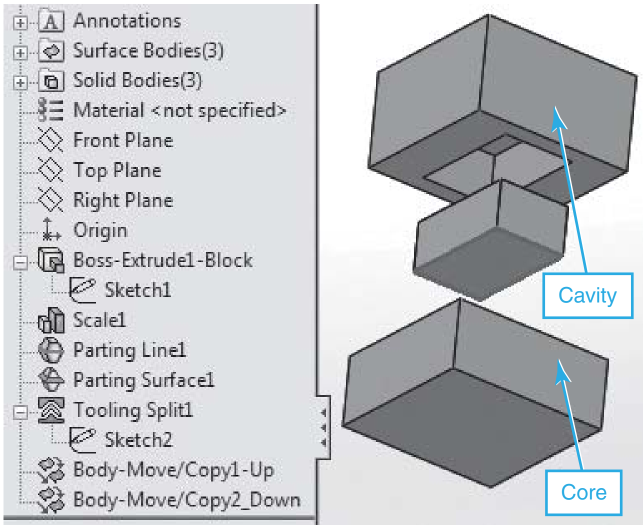


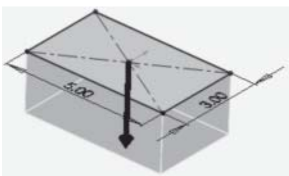
FIGURE 16.6
Core and cavity of a block mold

- ☐ Location of parting line: top or bottom face of the block. We use the bottom face here. The difference comes in placing the mold cavity and core in the CAD model. It is customary to place the cavity above the core in the CAD model, as shown in Figure 16.6.
- ☐ The pulling direction (parting axis) is the direction along which we pull the core away from the cavity. This direction determines the direction of the draft of the part faces.
- ☐ We could create core and cavity parts from scratch, using the part and assembly modeling concepts. As a matter of fact, as parts become complex, mold designers in practice design molds using these concepts instead of using mold software. To create mold halves, simply create two blocks and subtract the part from each using an extrude cut operation. The difference between the halves comes in what we add to them later. The half where we add the gate becomes the cavity and the half where we add the ejection system becomes the core.

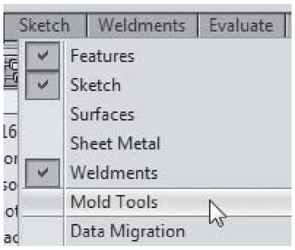
We create the mold in seven steps:

1. Create the block CAD model
2. Apply shrink factor
3. Create parting line
4. Create shut-off surface
5. Create parting surface
6. Apply tooling split to create the mold core and cavity
7. Create the tooling assembly

Step 1: Create *Sketch1* and *Boss-Extrude1-Block*: **File > New > Part > OK** > create 5×3 sketch on **Top Plane** and extrude down 2.0 in. > **File > Save As** > *tutorial16.1* > **Save.**

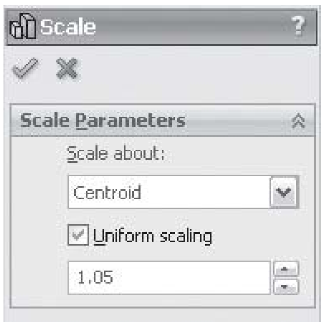


Step 2: Activate **Mold Tools** tab: Right-click **Sketch** tab (or any tab) > **Mold Tools** from drop-down shown.



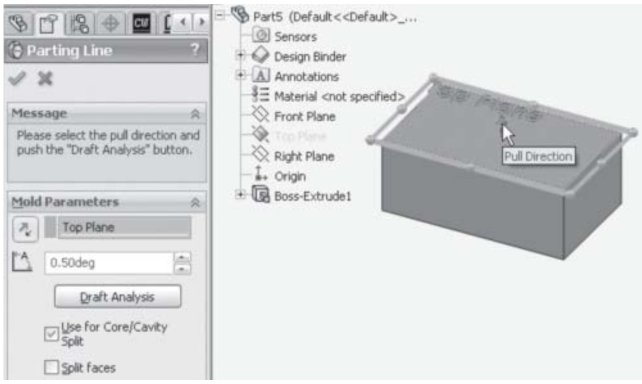
Note: This is a toggle; repeat to hide the tab.

Step 3: Create *Scale1* by applying a shrink factor of 5%: **Scale** on **Mold Tools** tab > enter 1.05 > ✔.



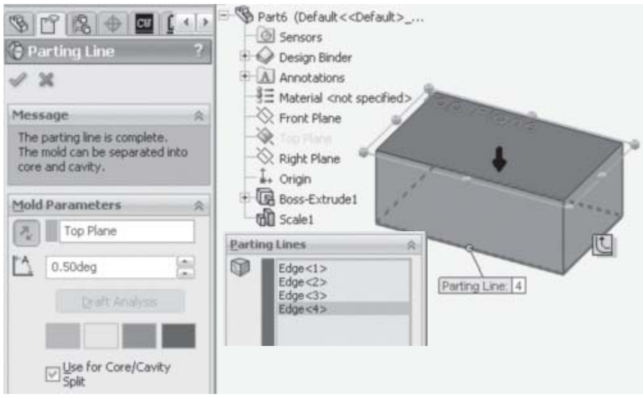
Note: Scale does change dimensions of sketches.

Step 4: Define pull direction for parting line: **Parting Line** on **Mold Tools** tab > expand features tree > **Top Plane** > click arrow direction to reverse the pull direction shown.



Note: We reverse pulling direction because we place cavity on top of core.

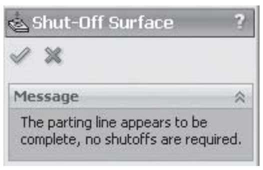
Note: As a rule, pulling direction must always pull away from cavity.



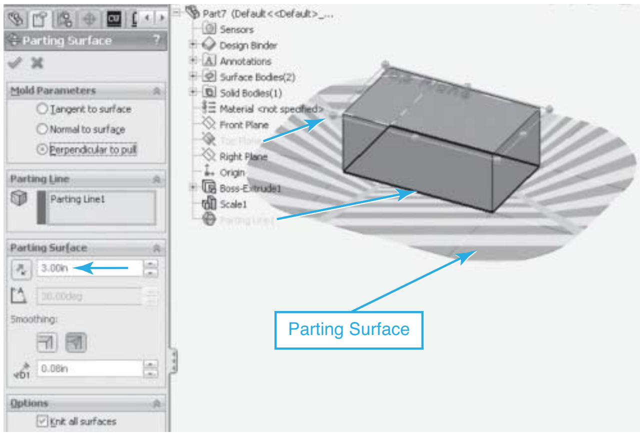
Step 5: Create *Parting Line1*: Enter 0.5 > **Draft Analysis** (button shown in Step 4) > four edges on bottom face (these edges form parting line) > ✔.

Note: The parting line must form a closed contour; otherwise, the tooling split step fails.

Step 6: Create shut-off surface: **Shut-Off Surface** on **Mold Tools** tab > ✖.



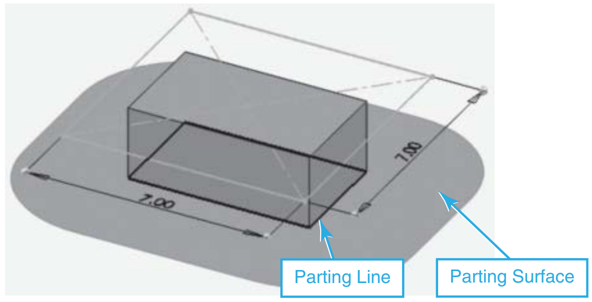
Note: The mold does not have holes to shut off.



Step 7: Create *Parting Surface1*: **Parting Surface** on **Mold Tools** tab > enter 3 for parting surface thickness shown above > ✔.

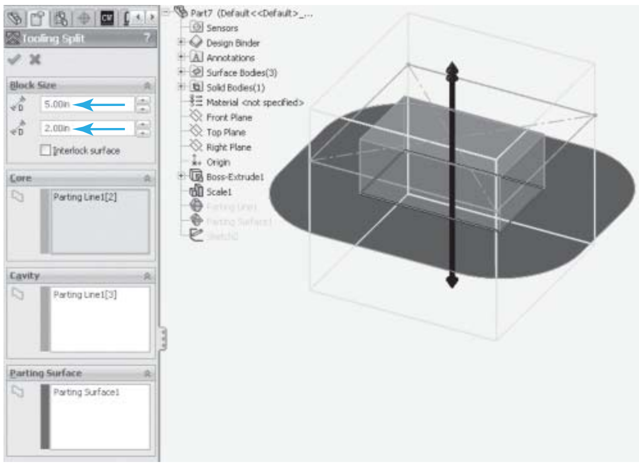
Note: We use a value of 3.0 in. for parting surface thickness. This thickness must be large enough to extend beyond the size of the tooling split block size defined later.

Step 8: Create *Sketch2 of Tooling Split1*: **Tooling Split** on **Mold Tools** tab > **Top Plane** > sketch 7 × 7 in. center rectangle shown > exit sketch.



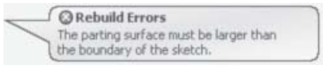
Note: The part center rectangle is 5 × 3 in. Thus, 7 × 7 encloses it.

Step 9: Create *Tooling Split1*: Enter 5 for **D** and 2 for **D** > ✔.



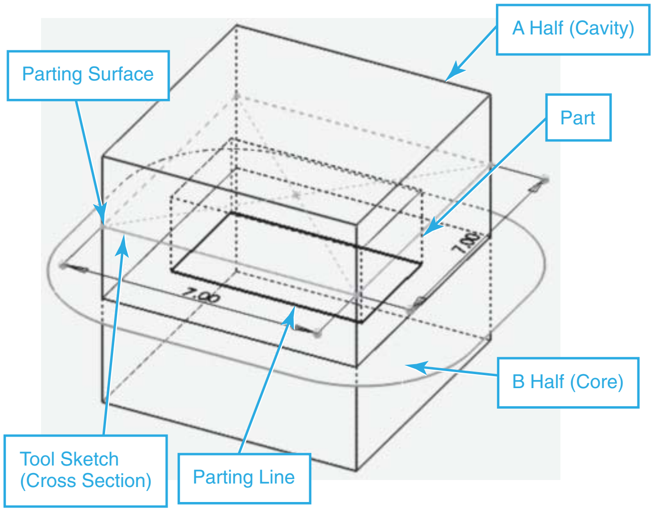
Note: You may check off the **Interlock surface** box. The interlock surface surrounds the perimeter of the parting surfaces in a 5° taper. The interlock surface drafts away from the parting line. Interlock surfaces are used to seal the mold properly and minimize the cost of machining the mold plates across the split because the area surrounding the interlock is planar.

Note: The block heights **D** of 5 and 2 in. shown are arbitrary and indicate the heights of the mold cavity and core, respectively.



Note: The error message shown is generated if the size of the **Parting Surface** is smaller than that of the tooling split size.

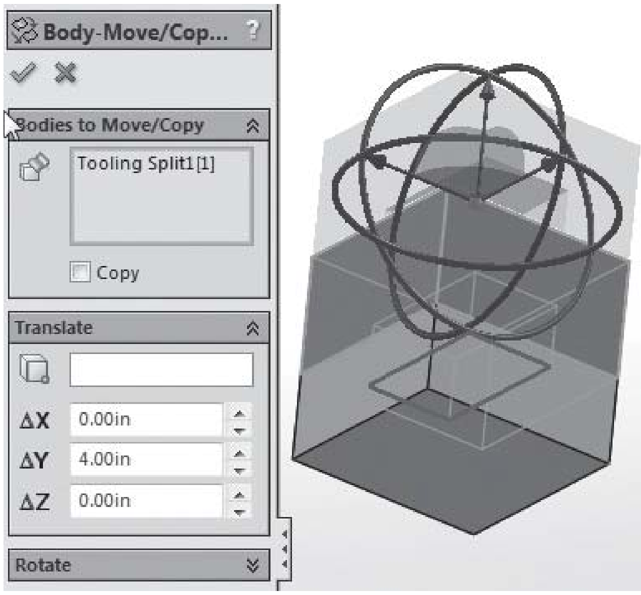
Note: The tool is created as shown. What is left is to move the two halves away to visualize the mold halves.

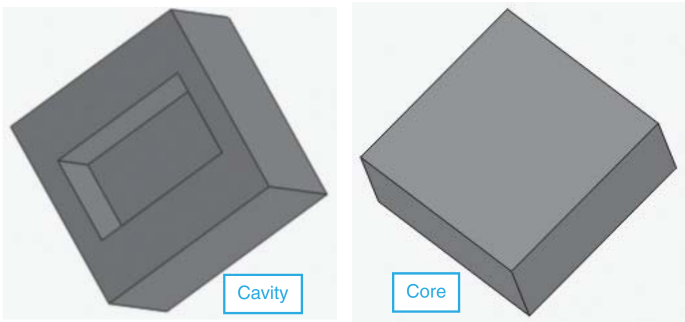


Step 10: Create *Body-Move/Copy1-Up* and *Body-Move/Copy1-Down*: Right-click **Sketch** tab (or any tab) > **Data Migration** tab from dropdown > **Move/Copy Bodies** on the **Data Migration** tab > **Translate/Rotate** button > cavity body (upper half) in graphics pane > enter 4 for **ΔY** field under **Translate** > ✔ > repeat for core (bottom half), but use –4 for **ΔY.**

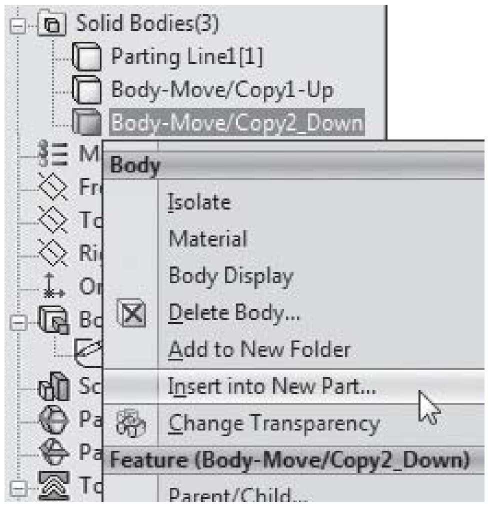
Note: The translation distance of 4 is arbitrary, just enough to provide good visualization of the mold.

Note: Hide the parting surface and rotate the mold halves shown below to view as shown. Also, use transparency to view the inside.



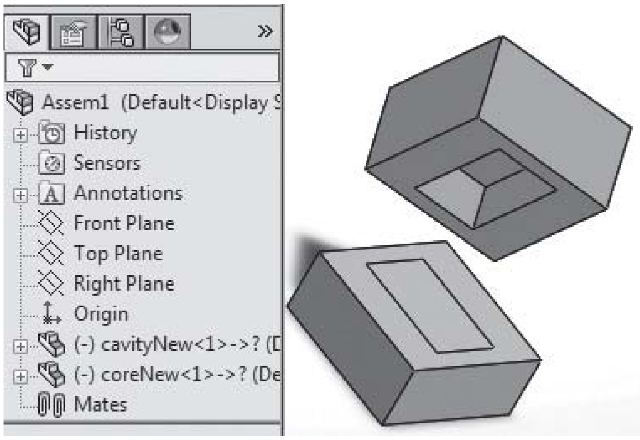


Step 11: Create tooling parts: Expand **Solid Bodies** node on features tree > right-click *Body-Move/Copy1-Up* > **Insert into New Part** from pop-up menu > type *cavity* for part name > **Save**. Repeat for *Body-Move/Copy2-Down* and save as *core*.



Step 12: Create tooling assembly: **File > New > Assembly > OK** > insert and assemble the core and cavity parts. Save the assembly *as tutorial16.1*.

Note: The two tooling parts are now components of the assembly, with external references to *tutorial16.1.sldprt*. You can add other mold features, create mates, and so on. Changes to the block model (in the *tutorial16.1* part file) are automatically reflected in the tooling parts in the assembly.



HANDS-ON FOR TUTORIAL 16–1. Add a vertical 1.0 in. diameter through hole in the block. The hole is centered in the top face of the block. Create the new mold.

Hint: You need a peg in the mold to create the hole in the block. The core must have the peg, not the cavity; otherwise, you cannot eject the part. The shut-off surface you create determines where the peg goes, to the core or the cavity. Experiment until you understand it.

Tutorial 16–2: Create a Sandbox Mold

We create the injection mold for the plastic sandbox shown in Figure 16.7. All dimensions are in inches. The difference between this tutorial and Tutorial 16–1 is that the sandbox is a shell with a wall thickness. Following Tutorial 16–1, we use the vertical orientation for the mold and we place the parting line at the top face of the shell. If we create the mold for the part in its present orientation, the core is placed above the cavity, which is against the customary convention. Thus, we flip the sandbox upside down to force the cavity above the core. We create the mold in seven steps:

1. Create the sandbox CAD model and flip it
2. Apply shrink factor and flip the part
3. Create parting line
4. Create shut-off surface
5. Create parting surface
6. Apply tooling split to create the mold core and cavity
7. Create the tooling assembly

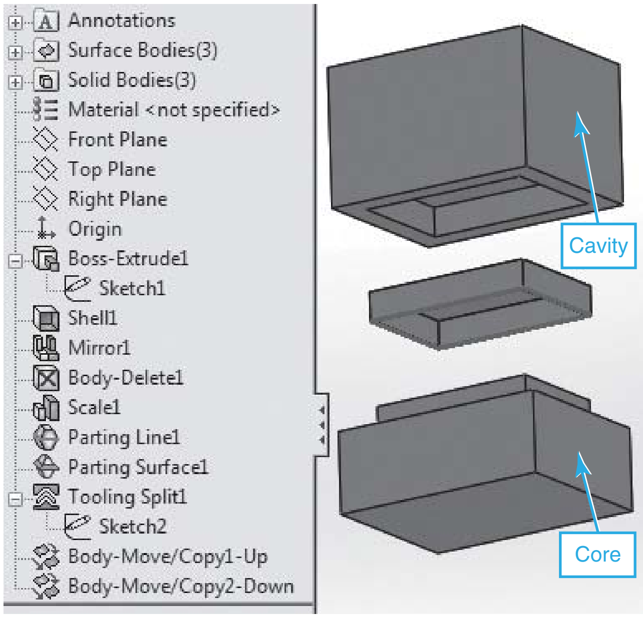
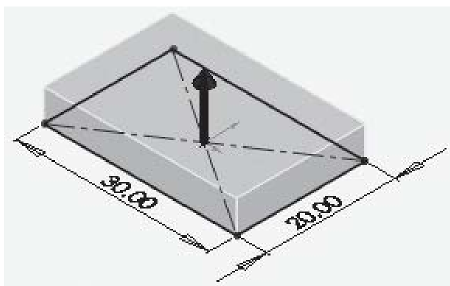
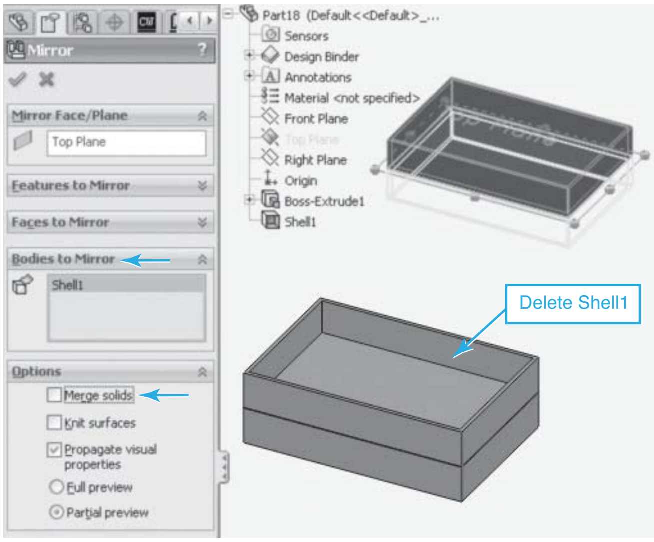


FIGURE 16.7
Core and cavity of a sandbox mold

Step 1: Create *Sketch1, Boss-Extrude1, and Shell1*: **File > New > Part > OK** > create 30 × 20 sketch on **Top Plane** and extrude 5.0 in. > shell top face with 0.5 in. thickness > **File > Save As** > *tutorial16.2* > **Save.**



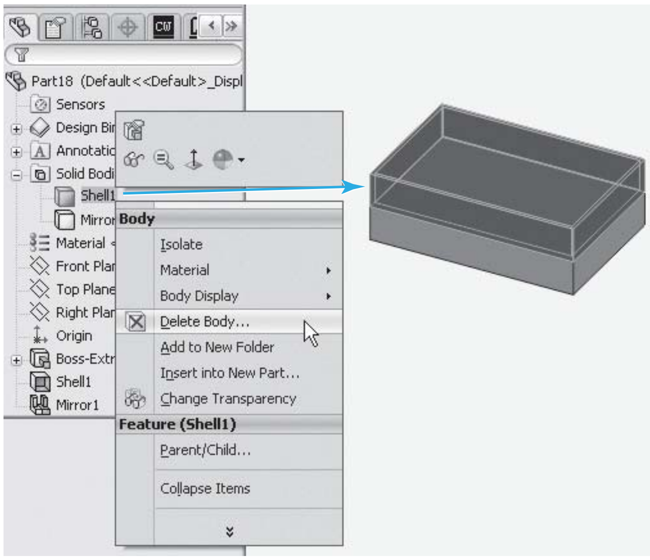
Step 2: Flip the sandbox upside down: **Mirror** on **Features** tab > **Top Plane** from features tree as mirror plane > expand **Bodies to Mirror** pane shown > select shell from graphics pane > uncheck **Merge solids** check box > ✔.



Note: If we do not uncheck the **Merge solids** check box, SolidWorks will combine the original shell with the mirrored one.

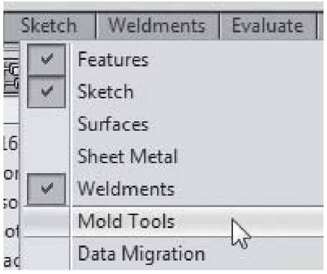
Step 3: Delete the original shell (*Shell1*): Expand **Solid Bodies(2)** on features tree > right-click *Shell1* > **Delete Body** > ✔.

Note: We did Steps 2 and 3 because we want the core to be below the cavity as shown in Figure 16.7.



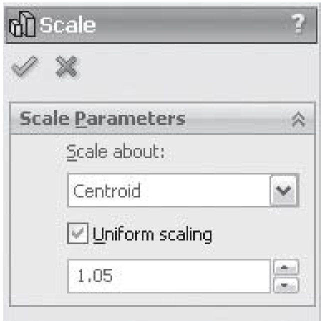
Step 4: Activate **Mold Tools** tab: Right-click **Sketch** tab (or any tab) > **Mold Tools** from dropdown that opens.

Note: This is a toggle; repeat to hide tab.



Step 5: Create *Scale1* by applying shrink factor of 5%: **Scale** on **Mold Tools** tab > enter 1.05 > ✔.

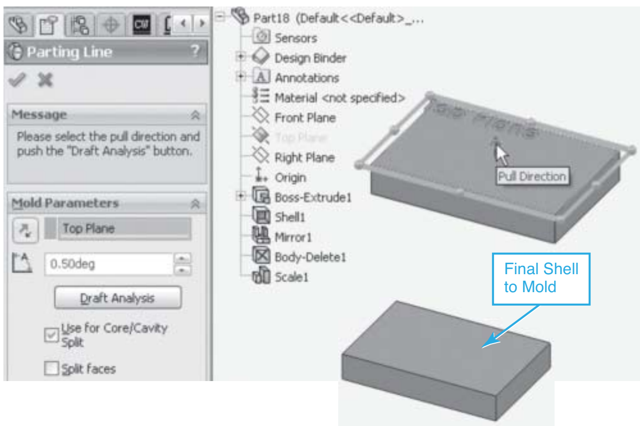
Note: **Scale** does change dimensions of sketches.



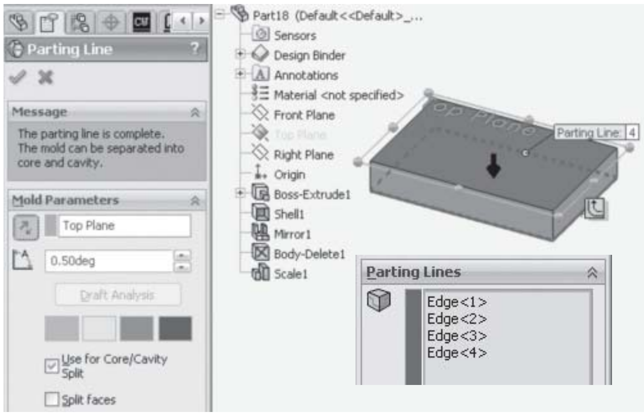
Step 6: Define pull direction for parting line: **Parting Line** on **Mold Tools** tab > expand features tree > **Top Plane** > click arrow direction to reverse the pull direction shown.

Note: We reverse pulling direction because we place cavity on top of core.

Note: As a rule, pulling direction must always pull away from cavity.



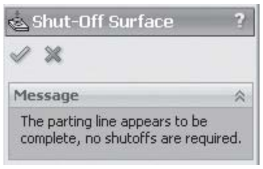
Step 7: Create *Parting Line1*: Enter 0.5 > **Draft Analysis** (button shown in Step 6) > four edges on top face of shell (these edges form parting line) > ✔.

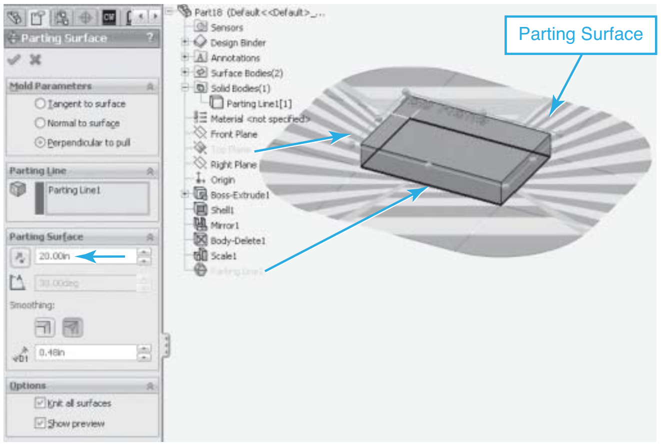


Note: The parting line must form a closed contour; otherwise, the tooling split step fails.

Step 8: Create shut-off surface: **Shut-Off Surface** on the **Mold Tools** tab > ✖.

Note: The mold does not have holes to shut off.



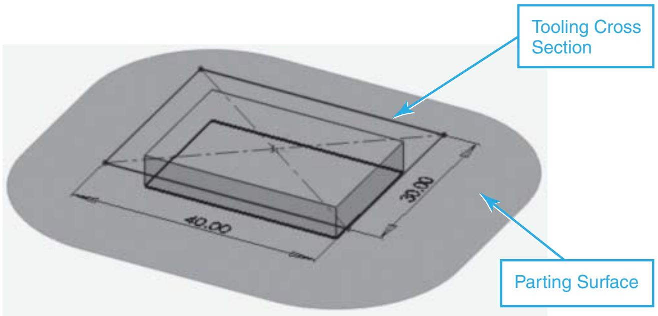


Step 9: Create *Parting Surface1*: **Parting Surface** on the **Mold Tools** tab > enter 20 for parting surface thickness shown above > ✔.

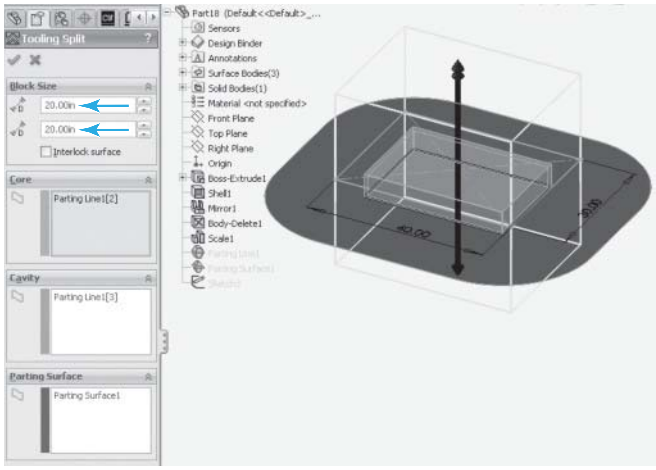
Note: We use a value of 20.0 in. for parting surface thickness. This thickness must be large enough to extend beyond the size of the tooling split block size defined in Step 10.

Step 10: Create *Sketch2 of Tooling Split1*: **Tooling Split** on **Mold Tools** tab > **Top Plane** > sketch 40 × 30 in center rectangle shown > exit sketch.

Note: The part center rectangle is 30 × 20 in. Thus, 40 × 30 encloses it.



Step 11: Create *Tooling Split1*: Enter 20 for **D** and 20 for **D** > ✔.

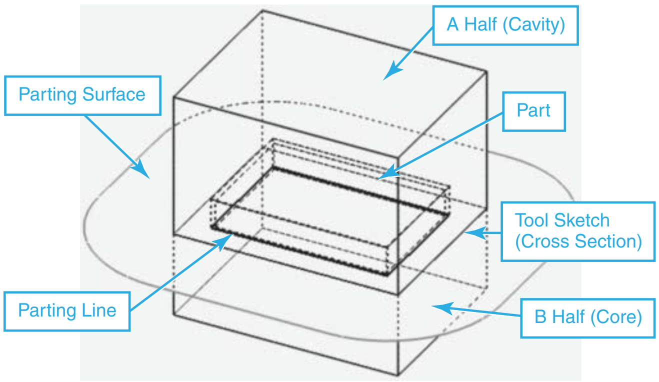


Note: You may check off the **Interlock surface** box. The interlock surface surrounds the perimeter of the parting surfaces in a 5° taper. The interlock surface drafts away from the parting line. Interlock surfaces are used to seal the mold properly and minimize the cost of machining the mold plates across the split because the area surrounding the interlock is planar.

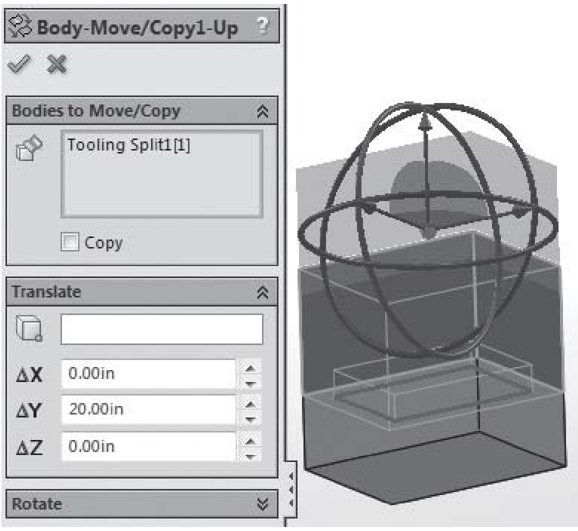
Note: The block heights **D** of 20 and 20 in. shown are arbitrary and indicate the heights of the mold cavity and core, respectively.

Note: The error message shown is generated if the size of the **Parting Surface** is smaller than that of the tooling split.

Note: The tool is created. What is left is to move the two halves away to visualize the mold halves.

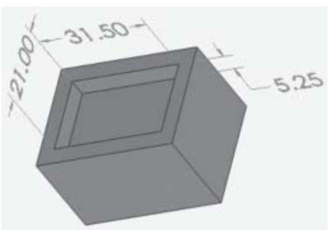


Step 12: Create *Body-Move/Copy1-Up* and *Body-Move/Copy1-Down*: Right-click any tab > **Data Migration** tab from dropdown > **Move/Copy Bodies** on **Data Migration** tab > **Translate/Rotate** button > cavity body (upper half) in graphics pane > enter 20 for **ΔY** field under **Translate** > ✔ > repeat for core (bottom half), but use –20 for **ΔY.**

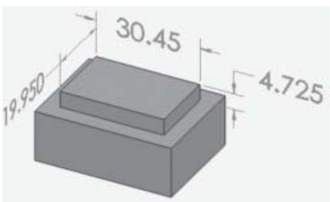


Note: The translation distance of 20 is arbitrary, just enough to provide good visualization of the mold.

Note: Hide the parting surface and rotate the mold halves to view as shown below. Also, use transparency to view the inside.



Cavity

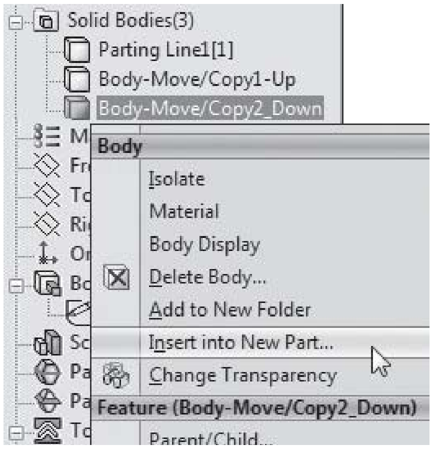


Core

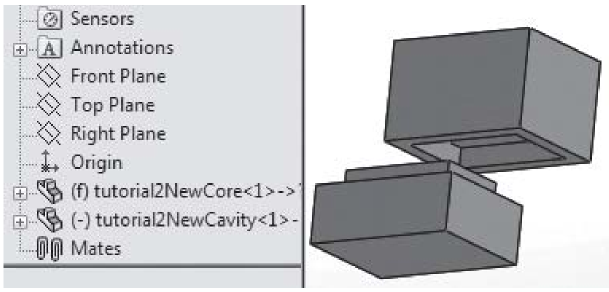
Note: We display the core and cavity dimensions to verify that our steps are correct. Study these dimensions and make sense of them. Keep in mind that the values shown above reflect the 5% scaling. Note that the effect of scaling does not show in the part dimensions, only the mold halves.

Note: SolidWorks does not allow you to edit the mold halves in their respective parts. There are no sketches or geometry in the features tree to edit. You must edit in the CAD model.

Step 13: Create tooling parts: Expand **Solid Bodies** node on features tree > right-click *Body-Move/Copy1-Up* > **Insert into New Part** from pop-up menu > type *cavity* for part name > **Save**. Repeat for *Body-Move/Copy2-Down* and save as *core*.



Step 14: Create tooling assembly: **File > New > Assembly > OK** > insert and assemble the core and cavity parts. Save the assembly as *tutorial16.2*.



Note: The two tooling parts are now components of the assembly, with external references to *tutorial16.2.sldprt*. You can add other mold features, create mates, and so on. Changes to the block model (in the *tutorial16.2* part file) are automatically reflected in the tooling parts in the assembly.

HANDS-ON FOR TUTORIAL 16–2. Redo the tutorial without Steps 2 and 3. What happens to the mold halves?

Tutorial 16–3: Create a Hemisphere Mold

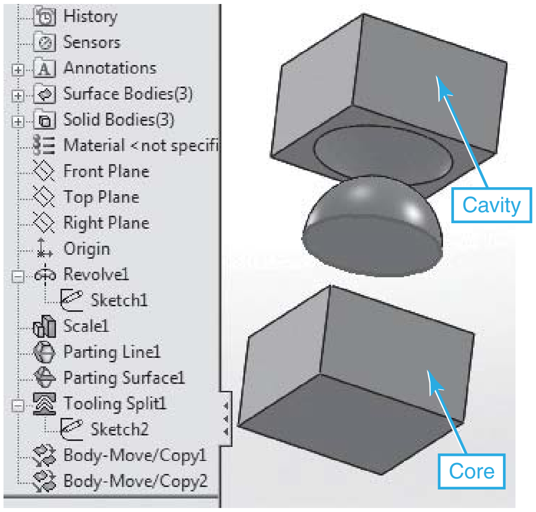
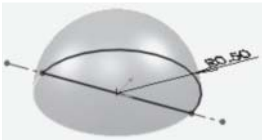


FIGURE 16.8
Core and cavity of a hemisphere mold

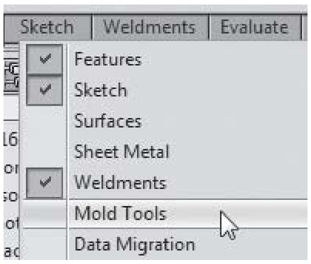
We create the injection mold for the plastic hemisphere shown in Figure 16.8. All dimensions are in inches.

We create the mold in seven steps:

1. Create the hemisphere CAD model
2. Apply shrink factor
3. Create parting line
4. Create shut-off surface
5. Create parting surface
6. Apply tooling split to create the mold core and cavity
7. Create the tooling assembly

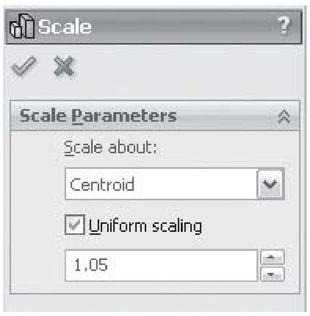


Step 1: Create *Sketch1* and *Revolve1*: **File > New > Part > OK** > create 0.5 in. radius half circle sketch shown on **Top Plane** and revolve sketch 180 degrees > **File > Save As** > *tutorial16.3* > **Save.**



Step 2: Activate **Mold Tools** tab: Right-click **Sketch** tab (or any tab) > **Mold Tools** from drop-down shown.

Note: This is a toggle; repeat to hide the tab.

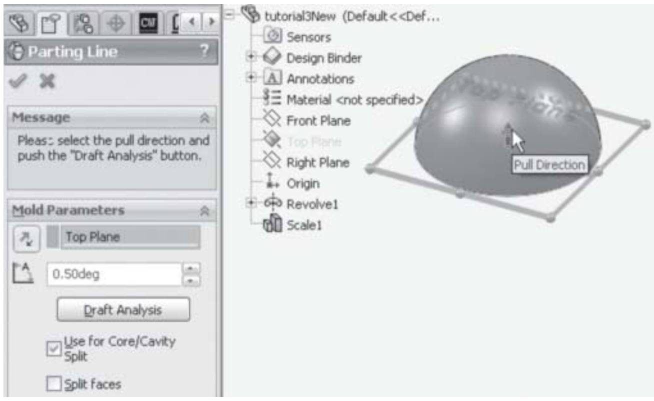


Step 3: Create *Scale1* by applying shrink factor of 5%: **Scale** on **Mold Tools** tab > enter 1.05 > ✔.

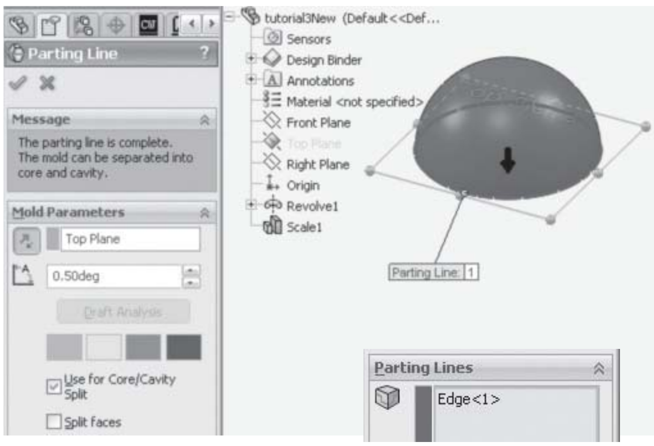
Note: Scale does change dimensions of sketches.

Step 4: Define pull direction for parting line: **Parting Line** on **Mold Tools** tab > expand features tree > **Top Plane** > click arrow direction to reverse the pull direction shown.

Note: We reverse pulling direction because we place cavity on top of core.



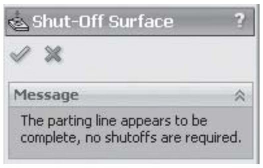
Note: As a rule, pulling direction must always pull away from cavity.



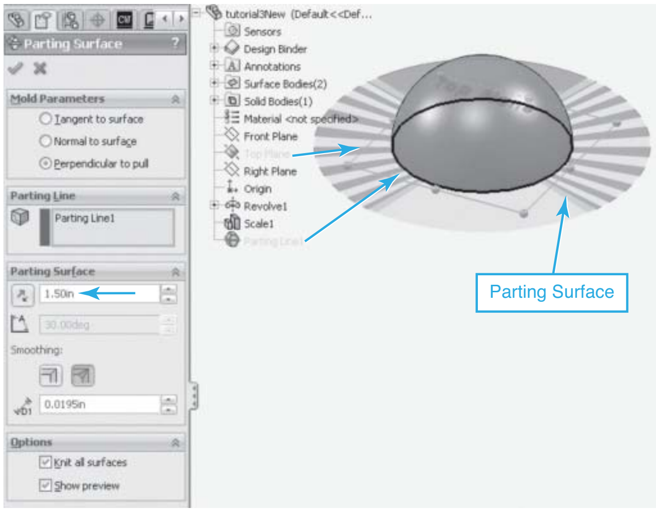
Step 5: Create *Parting Line1*: Enter 0.5 > **Draft Analysis** (button shown in Step 4) > edge on bottom face (this edge forms parting line) > ✔.

Note: The parting line must form a closed contour; otherwise the tooling split step fails.

Step 6: Create shut-off surface: **Shut-Off Surface** on the **Mold Tools** tab > ✖.



Note: The mold does not have holes to shut off.



Step 7: Create *Parting Surface1*: **Parting Surface** on **Mold Tools** tab > enter 1.5 for parting surface thickness shown above > ✔.

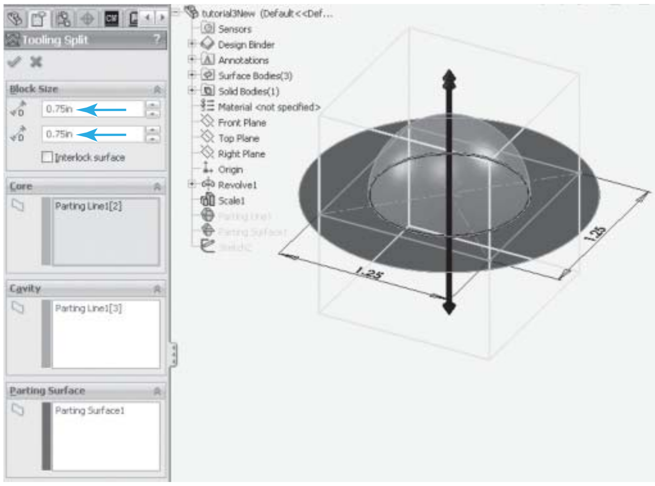
Note: We use a value of 1.5 in. for parting surface thickness. This thickness must be large enough to extend beyond the size of the tooling split block size defined later.

Step 8: Create *Sketch2 of Tooling Split1*: **Tooling Split** on **Mold Tools** tab > **Top Plane** > sketch 1.25 × 1.25 in. center rectangle shown > exit sketch.

Note: The part circle is 0.5 in. radius. Thus, the 1.25 × 1.25 rectangle encloses it.

Step 9: Create *Tooling Split1*: Enter 0.75 for **D** and 0.75 for **D** > ✔.

Note: You may check off the **Interlock surface** box. The interlock surface surrounds the perimeter of the parting surfaces in a

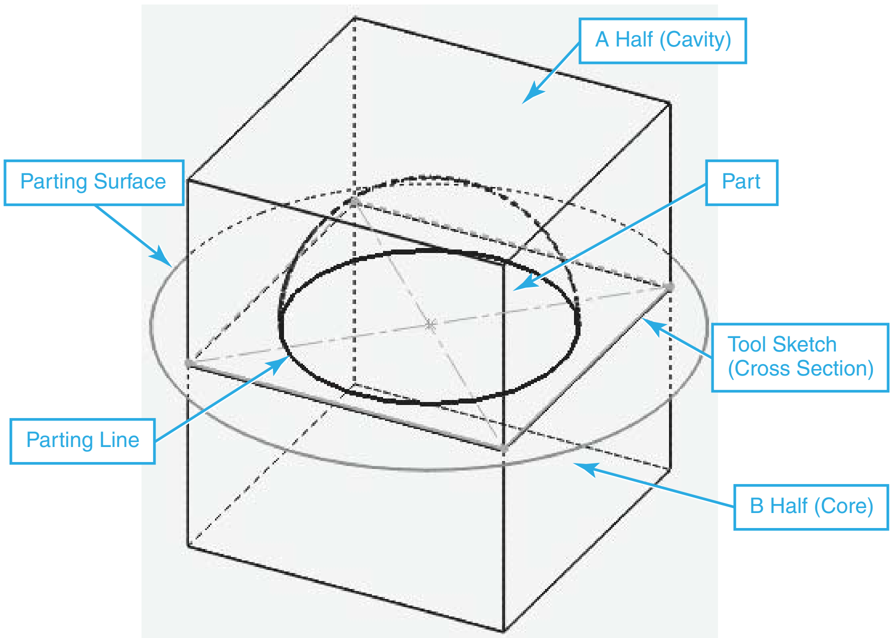


5° taper. The interlock surface drafts away from the parting line. Interlock surfaces are used to seal the mold properly and minimize the cost of machining the mold plates across the split because the area surrounding the interlock is planar.

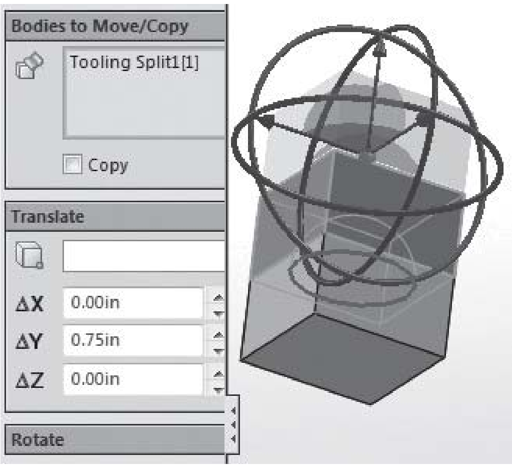
Note: The block heights **D** of 0.75 and 0.75 in. shown are arbitrary and indicate the heights of the mold cavity and core, respectively.

Note: The error message shown is generated if the size of the **Parting Surface** is smaller than that of the tooling split.

Note: The tool is created. What is left is to move the two halves away to visualize the mold halves.

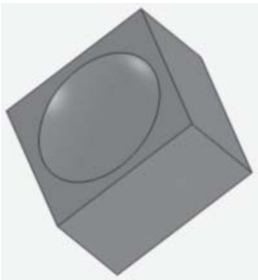


Step 10: Create *Body-Move/Copy1-Up* and *Body-Move/Copy1-Down*: Right-click **Sketch** tab > **Data Migration** tab from dropdown > **Move/Copy Bodies** on **Data Migration** tab > **Translate/Rotate** button > cavity body (upper half) in graphics pane > enter 0.75 for **ΔY** field under **Translate** > ✔ > repeat for core (bottom half), but use –0.75 for **ΔY.**

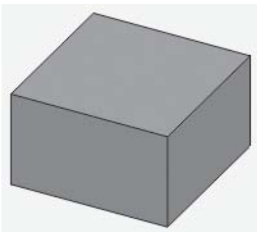


Note: The translation distance of 0.75 is arbitrary, just enough to provide good visualization of the mold.

Note: Hide the parting surface and rotate the mold halves to view as shown below. Also, use transparency to view the inside.

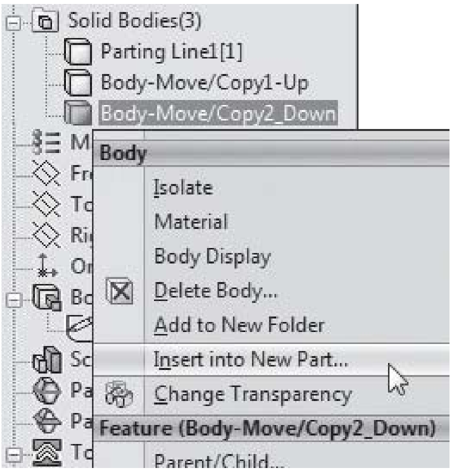


Cavity



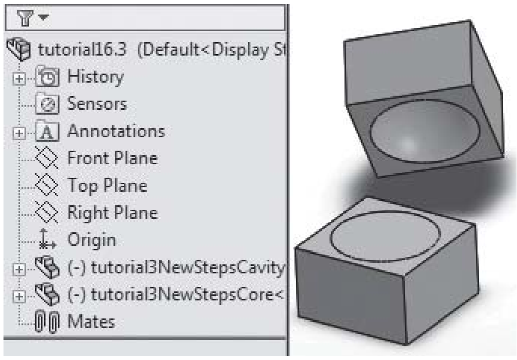
Core

Step 11: Create tooling parts: Expand **Solid Bodies** node on features tree > Right-click *Body-Move/Copy1-Up* > **Insert into New Part** from pop-up menu > type *cavity* for part name > **Save**. Repeat for *Body-Move/Copy2-Down* and save as *core*.



Step 12: Create tooling assembly: **File** > **New** > **Assembly** > **OK** > insert and assemble the core and cavity parts. Save the assembly *as tutorial16.3*.

Note: The two tooling parts are now components of the assembly, with external references to *tutorial16.3.sldprt*. You can add other mold features, create mates, and so on. Changes to the block model (in the *tutorial16.3* part file) are automatically reflected in the tooling parts in the assembly.



HANDS-ON FOR TUTORIAL 16–3. Create a 0.5 inch diameter hemisphere hole at the center of the hemisphere. Create the new mold.

Tutorial 16–4: Create an Easter Egg Mold

We create the injection mold for the plastic Easter egg shown in Figure 16.9. All dimensions are in inches. We show how to split the Easter egg face in the middle to create a symmetric mold (i.e., core and cavity are identical). The key is to check off the **Split faces** box while creating the parting line.

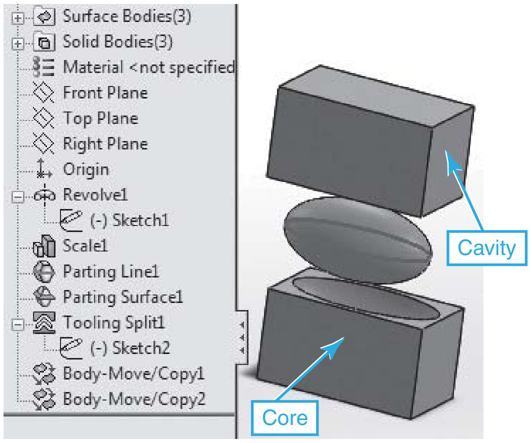
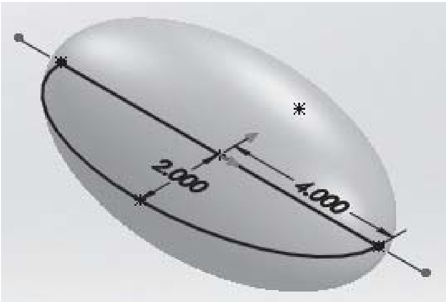


FIGURE 16.9
Core and cavity of an Easter egg mold

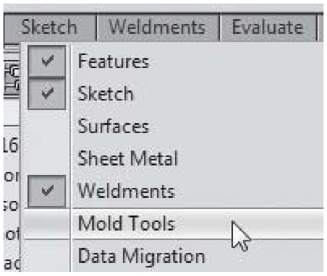
We create the mold in seven steps:

1. Create the hemisphere CAD model
2. Apply shrink factor
3. Create parting line
4. Create shut-off surface
5. Create parting surface
6. Apply tooling split to create the mold core and cavity
7. Create the tooling assembly

Step 1: Create *Sketch1* and *Revolve1*: **File > New > Part > OK >** create half an ellipse with major and minor radii of 4 and 2 in. respectively in sketch on **Top Plane** and revolve sketch 360 degrees > **File > Save As >** *tutorial16.4* > **Save.**

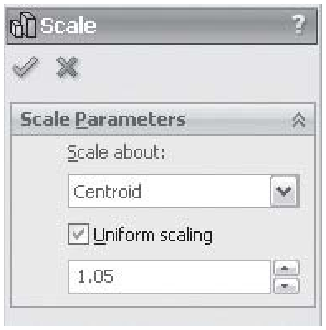


Step 2: Activate **Mold Tools** tab: Right-click **Sketch** tab (or any tab) > **Mold Tools** from dropdown shown.



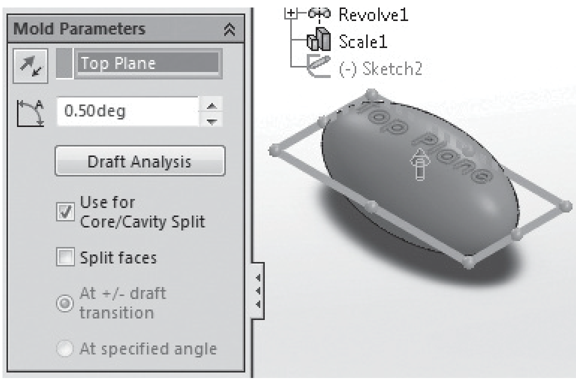
Note: This is a toggle; repeat to hide the tab.

Step 3: Create *Scale1* by applying shrink factor of 5%: **Scale** on the **Mold Tools** tab > enter 1.05 > ✔.



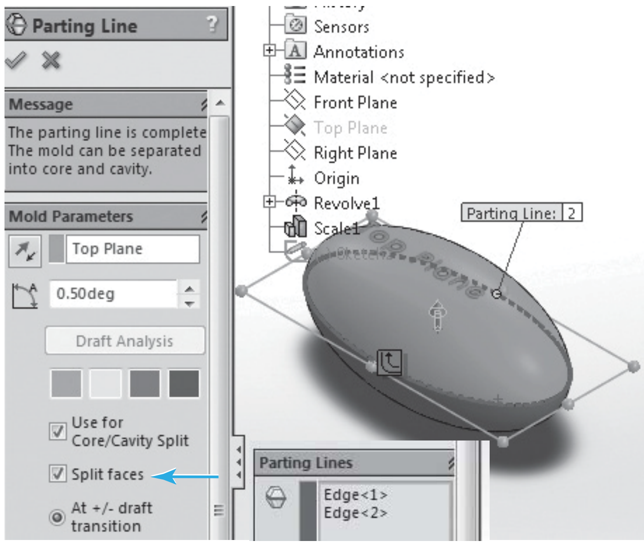
Note: Scale does change dimensions of sketches.

Step 4: Define pull direction for parting line: **Parting Line** on **Mold Tools** tab > expand features tree > **Top Plane** > click arrow direction to reverse the pull direction shown.



Note: We reverse pulling direction because we place cavity on top of core.

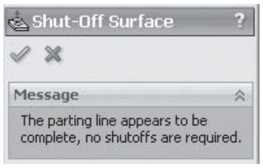
Note: As a rule, pulling direction must always pull away from cavity.



Step 5: Create *Parting Line1*: Enter 0.5 > **Draft Analysis** (button shown in Step 4) > **Split faces** checkbox shown > two middle edges shown (these edges form the parting line) > ✔.

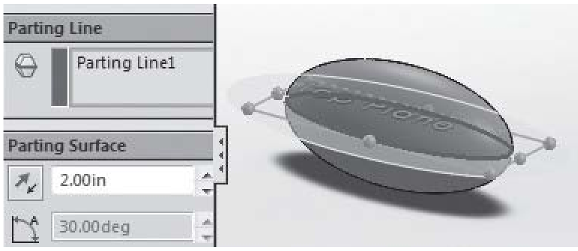
Note: The parting line must form a closed contour; otherwise, the tooling split step fails.

Step 6: Create shut-off surface: **Shut-Off Surface** on **Mold Tools** tab > ✖.



Note: The mold does not have holes to shut off.

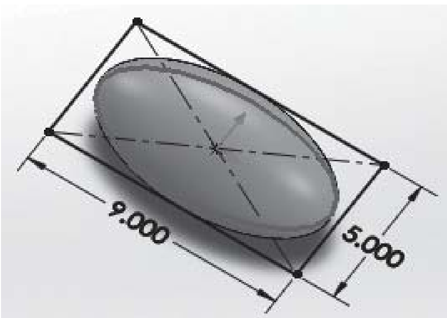
Step 7: Create *Parting Surface1*: **Parting Surface** on **Mold Tools** tab > enter 2 for parting surface thickness shown above > ✔.



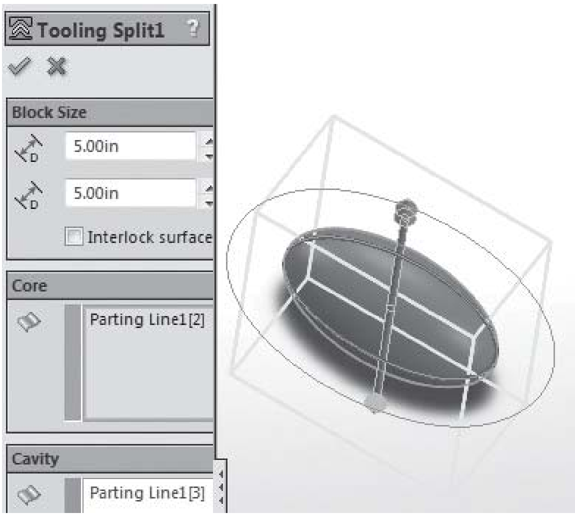
Note: We use a value of 2.0 in. for parting surface thickness. This thickness must be large enough to extend beyond the size of the tooling split block size.

Step 8: Create *Sketch2 of Tooling Split1*: **Tooling Split** on **Mold Tools** tab > **Top Plane** > sketch 9 × 5 in. center rectangle shown > exit sketch.

Note: The part ellipse has 4 × 2 in. radii. Thus, the 9 × 5 rectangle encloses it.



Step 9: Create *Tooling Split1*: Enter 5.0 for **D** and 5.0 for **D** > ✔.

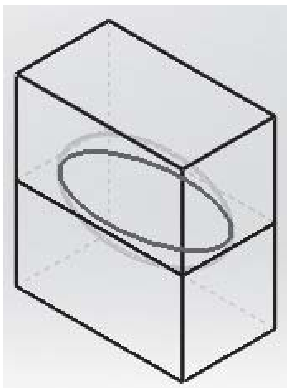


Note: You may check off the **Interlock surface** box. The interlock surface surrounds the perimeter of the parting surfaces in a 5° taper. The interlock surface drafts away from the parting line. Interlock surfaces are used to seal the mold properly and minimize the cost of machining the mold plates across the split because the area surrounding the interlock is planar.

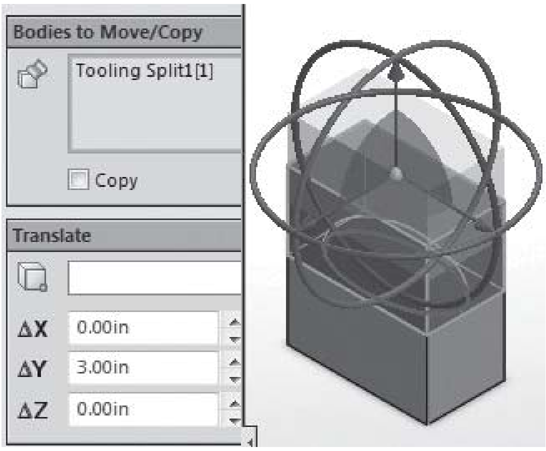
Note: The block heights **D** of 5.0 and 5.0 in. shown are arbitrary and indicate the heights of the mold cavity and core, respectively.

Note: The error message shown is generated if the size of the **Parting Surface** is smaller than that of the tooling split.

Note: The tool is created. What is left is to move the two halves away to visualize the mold halves.

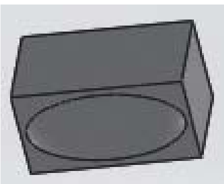


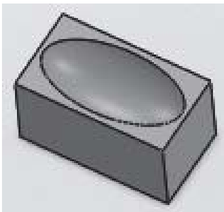
Step 10: Create *Body-Move/Copy1-Up* and *Body-Move/Copy1-Down*: Right-click **Sketch** tab > **Data Migration** tab from dropdown > **Move/Copy Bodies** on **Data Migration** tab > **Translate/Rotate** button > cavity body (upper half) in graphics pane > enter 3.0 for **ΔY** field under **Translate** > ✔ > repeat for core (bottom half), but use –3.0 for **ΔY.**



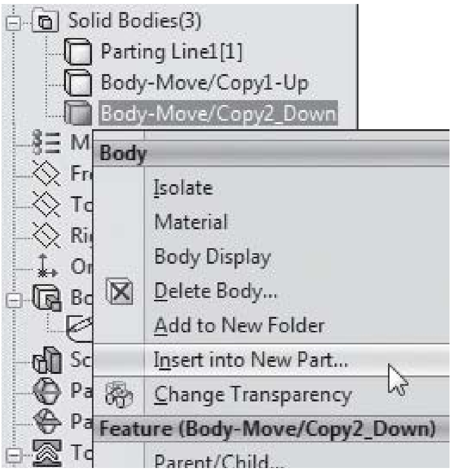
Note: The translation distance of 3.0 is arbitrary, just enough to provide good visualization of the mold.

Note: Hide the parting surface and rotate the mold halves to view as shown below. Also, use transparency to view the inside.





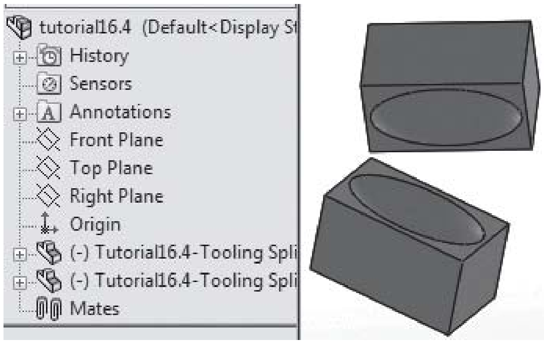
Step 11: Create tooling parts: Expand **Solid Bodies** node on features tree > Right-click *Body-Move/Copy1-Up* > **Insert into New Part** from pop-up menu > type



cavity for part name > **Save**. Repeat for *Body-Move/Copy2-Down* and save as *core*.

Step 12: Create tooling assembly: **File** > **New** > **Assembly** > **OK** > insert and assemble the core and cavity parts. Save the assembly as *tutorial16.4*.

Note: The two tooling parts are now components of the assembly, with external references to *tutorial16.4.sldprt*. You can add other mold features, create mates, and so on. Changes to the block model (in the *tutorial16.4* part file) are automatically reflected in the tooling parts in the assembly.



HANDS-ON FOR TUTORIAL 16–4. Create a mold that will make two Easter eggs in one injection cycle.

Tutorial 16–5: Generate a Mold Drawing

We create the engineering drawing of the sandbox mold created in Tutorial 16–2.

Step 1: Create cavity drawing: **Open** > **New** > **Drawing** > **OK** > **OK** (accept the default **Standard sheet size**) > **Browse** (locate and select cavity part) > insert views and dimension as shown in Figure 16.10 > ✔.

Note: The dimensions shown reflect the 5% scaling (shrink) factor.

Note: Study and verify that these dimensions are what you would expect.

Step 2: Create core drawing: Repeat Step 1 to create core drawing shown in Figure 16.11.

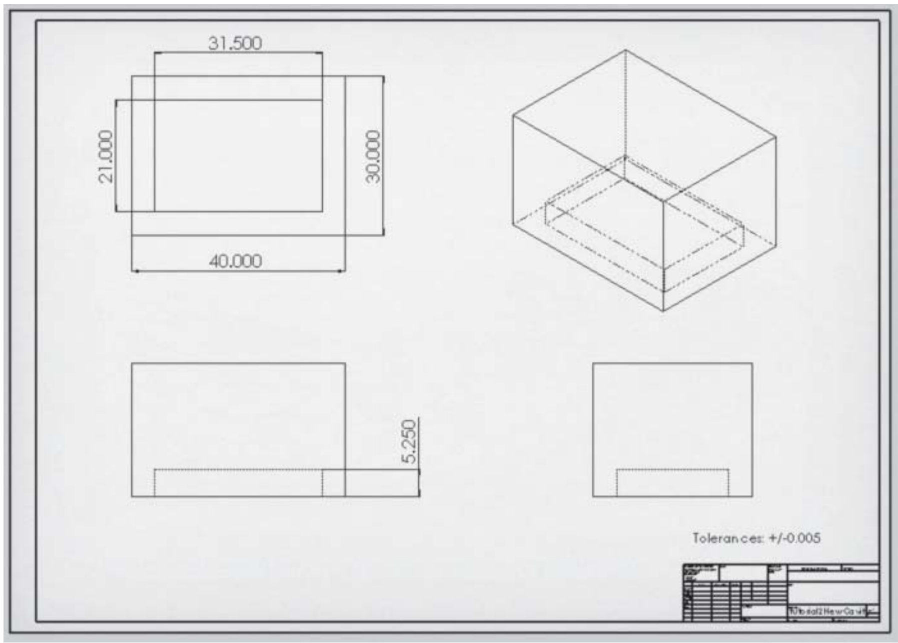


FIGURE 16.10
Mold cavity drawing

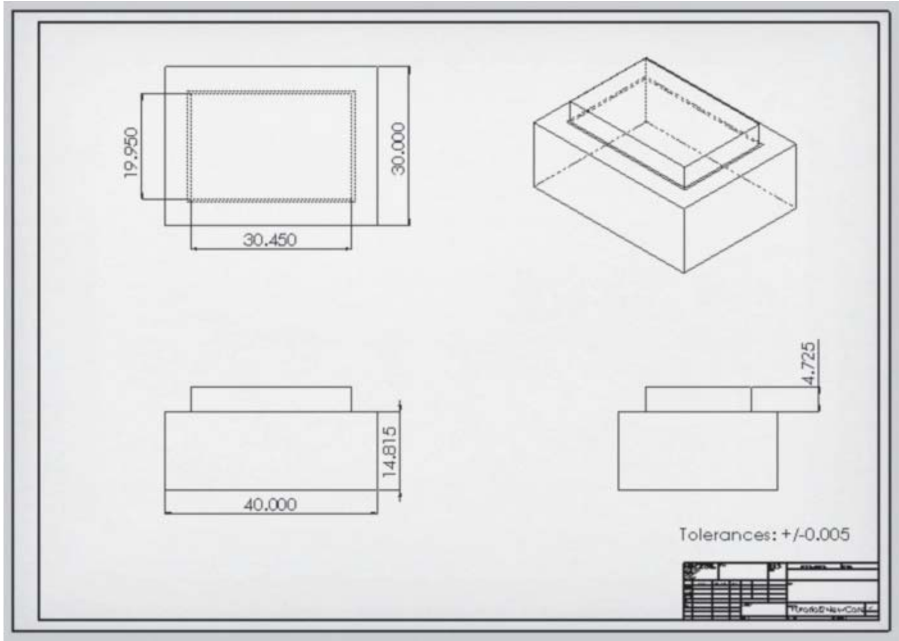


FIGURE 16.11
Mold core drawing

HANDS-ON FOR TUTORIAL 16–5. Create an assembly drawing for the mold. Add dimensions and tolerances.

problems

1. What are the differences between injection molding and machining?
2. Describe the injection molding process and its phases (elements).
3. Perform research on the type of products that injection molding can produce. Submit a detailed report about products, their applications, materials, size, and so on.
4. Research the injection molding materials. Submit a detailed report about the material properties, applications (type of parts it makes), melting temperature, molding pressure, and so on.
5. What are the advantages and disadvantages of injection molding?
6. Describe the main sections of an injection molding machine. What does each section do? Sketch a machine.
7. Describe the phases of an injection molding cycle. Sketch these phases.
8. Classify the mold types. Describe these types.
9. What does a shrink factor do? Why do we need it in mold design?
10. Why do we need a parting line and a parting surface in mold design?
11. What is the result of a poor design of a runner system?
12. What is the difference between a sprue and other cooling channels?
13. Describe and explain five defects of injection molding. What causes them?
14. What is the result of a poorly vented mold?
15. List and discuss some of the mold design rules.
16. What does a tooling split do?
17. What is a shut-off surface? Why do we need it?
18. Use SolidWorks to design a mold for the flange model shown in Figure 2.32 of Chapter 2.
19. Use SolidWorks to design a mold for the L bracket shown in Figure 2.35 of Chapter 2.
20. Use SolidWorks to design a mold for the pattern block shown in Figure 2.36A of Chapter 2.
21. Use SolidWorks to design a mold for the pattern wheel shown in Figure 2.36B of Chapter 2.

22. Figure 16.12 shows the mold of a funnel. Create the mold as shown. Assume all dimensions.

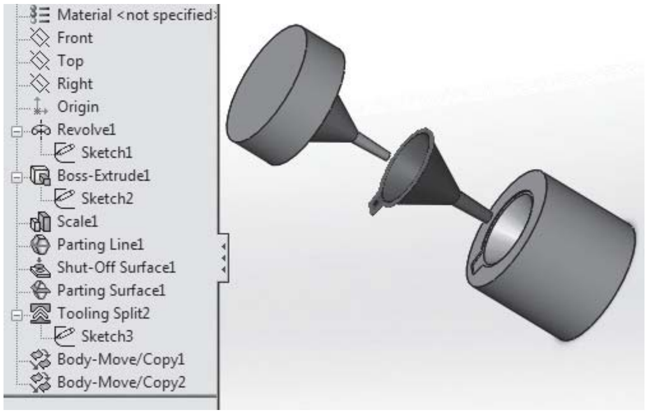


FIGURE 16.12
Core and cavity of a funnel mold

23. Figure 16.13 shows the mold of a solid bottle. Create the mold as shown. Assume all dimensions.

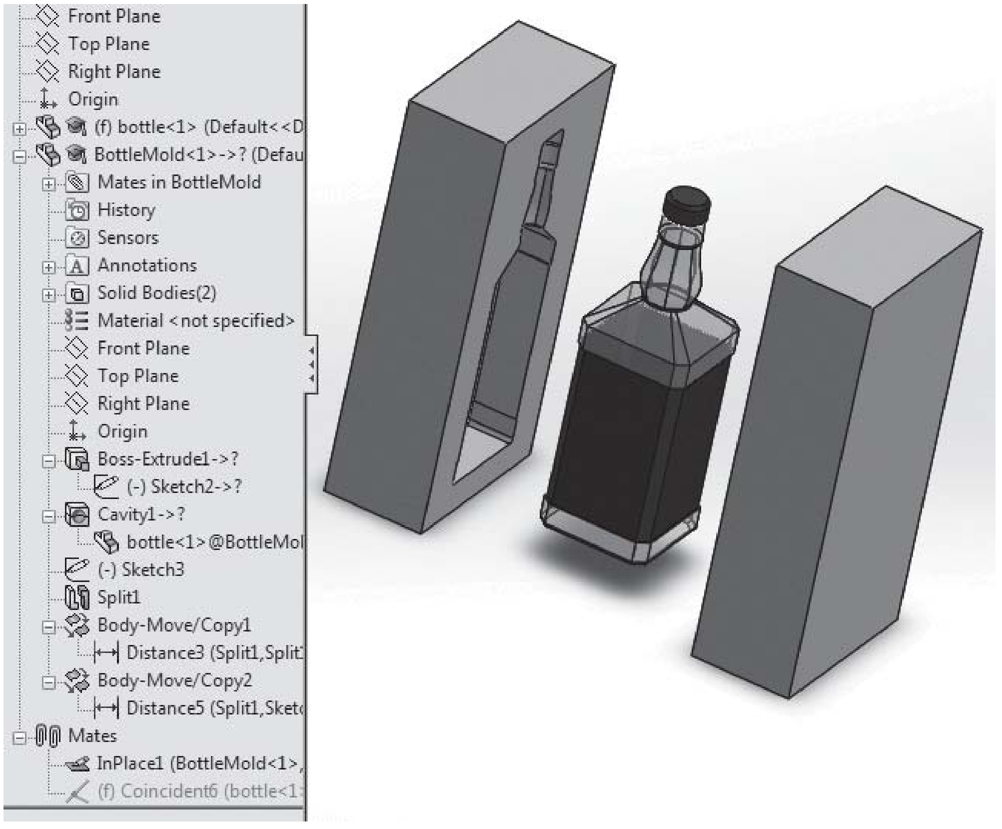


FIGURE 16.13
Core and cavity of a bottle mold

APPENDIX

A

ANSI and ISO Tolerance Tables

This appendix lists the ANSI and ISO tolerance tables. The tables show the ANSI fits and their equivalent ISO symbols. The tolerance values are given for ranges of the basic sizes. All numbers in the tables are in inches. The tolerance values shown in the tables must be multiplied by 10^{-3}.

TABLE A.1 Clearance Fits[1]

	RC1			RC2			RC3		
Basic Size	Limits of Clearance	Standard Limits		Limits of Clearance	Standard Limits		Limits of Clearance	Standard Limits	
Over–To		Hole H5	Shaft g4		Hole H6	Shaft g5		Hole H7	Shaft f6
0–0.12	0.1 0.45	+0.2 0	−0.1 −0.25	0.1 0.55	+0.25 0	−0.1 −0.3	0.3 0.95	+0.4 0	−0.3 −0.55
0.12–0.24	0.15 0.5	+0.2 0	−0.15 −0.3	0.15 0.65	+0.3 0	−0.15 −0.35	0.4 1.12	+0.5 0	−0.4 −0.7
0.24–0.40	0.2 0.6	+0.25 0	−0.2 −0.35	0.2 0.85	+0.4 0	−0.2 −0.45	0.5 1.5	+0.6 0	−0.5 −0.9
0.40–0.71	0.25 0.75	+0.3 0	−0.25 −0.45	0.25 0.95	+0.4 0	−0.25 −0.55	0.6 1.7	+0.7 0	−0.6 −1.0
0.71–1.19	0.3 0.95	+0.4 0	−0.3 −0.55	0.3 1.2	+0.5 0	−0.3 −0.7	0.8 2.1	+0.8 0	−0.8 −1.3
1.19–1.97	0.4 1.1	+0.4 0	−0.4 −0.7	0.4 1.4	+0.6 0	−0.4 −0.8	1.0 2.6	+1.0 0	−1.0 −1.6
1.97–3.15	0.4 1.2	+0.5 0	−0.4 −0.7	0.4 1.6	+0.7 0	−0.4 −0.9	1.2 3.1	+1.2 0	−1.2 −1.9
3.15–4.73	0.5 1.5	+0.6 0	−0.5 −0.9	0.5 2.0	+0.9 0	−0.5 −1.1	1.4 3.7	+1.4 0	−1.4 −2.3
4.73–7.09	0.6 1.8	+0.7 0	−0.6 −1.1	0.6 2.3	+1.0 0	−0.6 −1.3	1.6 4.2	+1.6 0	−1.6 −2.6
7.09–9.85	0.6 2.0	+0.8 0	−0.6 −1.2	0.6 2.6	+1.2 0	−0.6 −1.4	2.0 5.0	+1.8 0	−2.0 −3.2
9.85–12.41	0.8 2.3	+0.9 0	−0.8 −1.4	0.8 2.9	+1.2 0	−0.8 −1.7	2.5 5.7	+2.0 0	−2.5 −3.7

[1]Table entries are in inches. Multiply the entries in *Limits of clearance* and *Standard limits* columns by 10^{-3}.

TABLE A.1 Clearance Fits[2] (*continued*)

	RC4			RC5			RC6		
Basic Size	**Limits of Clearance**	**Standard Limits**		**Limits of Clearance**	**Standard Limits**		**Limits of Clearance**	**Standard Limits**	
Over–To		**Hole H8**	**Shaft f7**		**Hole H8**	**Shaft e7**		**Hole H9**	**Shaft e8**
0–0.12	0.3 1.3	+0.6 0	−0.3 −0.7	0.6 1.6	+0.6 0	−0.6 −1.0	0.6 2.2	+1.0 0	−0.6 −1.2
0.12–0.24	0.4 1.6	+0.7 0	−0.4 −0.9	0.8 2.0	+0.7 0	−0.8 −1.3	0.8 2.7	+1.2 0	−0.8 −1.5
0.24–0.40	0.5 2.0	+0.9 0	−0.5 −1.1	1.0 2.5	+0.9 0	−1.0 −1.6	1.0 3.3	+1.4 0	−1.0 −1.9
0.40–0.71	0.6 2.3	+1.0 0	−0.6 −1.3	1.2 2.9	+1.0 0	−1.2 −1.9	1.2 3.8	+1.6 0	−1.2 −2.2
0.71–1.19	0.8 2.8	+1.2 0	−0.8 −1.6	1.6 3.6	+1.2 0	−1.6 −2.4	1.6 4.8	+2.0 0	−1.6 −2.8
1.19–1.97	1.0 3.6	+1.6 0	−1.0 −2.0	2.0 4.6	+1.6 0	−2.0 −3.0	2.0 6.1	+2.5 0	−2.0 −3.6
1.97–3.15	1.2 4.2	+1.8 0	−1.2 −2.4	2.5 5.5	+1.8 0	−2.5 −3.7	2.5 7.3	+3.0 0	−2.5 −4.3
3.15–4.73	1.4 5.0	+2.2 0	−1.4 −2.8	3.0 6.6	+2.2 0	−3.0 −4.4	3.0 8.7	+3.5 0	−3.0 −5.2
4.73–7.09	1.6 5.7	+2.5 0	−1.6 −3.2	3.5 7.6	+2.5 0	−2.5 −5.1	3.5 10.0	+4.0 0	−3.5 −6.0
7.09–9.85	2.0 6.6	+2.8 0	−2.0 −3.8	4.0 8.6	+2.8 0	−4.0 −5.8	4.0 11.3	+4.5 0	−4.0 −6.8
9.85–12.41	2.5 7.5	+3.0 0	−2.5 −4.5	5.0 10.0	+3.0 0	−5.0 −7.0	5.0 13.0	+5.0 0	−5.0 −8.0

[2]Table entries are in inches. Multiply the entries in *Limits of clearance* and *Standard limits* columns by 10^{-3}.

TABLE A.1 Clearance Fits[3] (*continued*)

	RC7			RC8			RC9		
Basic Size	**Limits of Clearance**	**Standard Limits**		**Limits of Clearance**	**Standard Limits**		**Limits of Clearance**	**Standard Limits**	
Over–To		**Hole H9**	**Shaft d8**		**Hole H10**	**Shaft g9**		**Hole H11**	**Shaft c11**
0–0.12	1.0 2.6	+1.0 0	−1.0 −1.6	2.5 5.1	+1.6 0	−2.5 −3.5	4.0 8.1	+2.5 0	−4.0 −5.6
0.12–0.24	1.2 3.1	+1.2 0	−1.2 −1.9	2.8 5.8	+1.8 0	−2.8 −4.0	4.5 9.0	+3.0 0	−4.5 −6.0
0.24–0.40	1.6 3.9	+1.4 0	−1.6 −2.5	3.0 6.6	+2.2 0	−3.0 −4.4	5.0 10.7	+3.5 0	−5.0 −7.2
0.40–0.71	2.0 4.6	+1.6 0	−2.0 −3.0	3.5 7.9	+2.8 0	−3.5 −5.1	6.0 12.8	+4.0 0	−6.0 −8.8
0.71–1.19	2.5 5.7	+2.0 0	−2.5 −3.7	4.5 10.0	+3.5 0	−4.5 −6.5	7.0 15.5	+5.0 0	−7.0 −10.5
1.19–1.97	3.0 7.1	+2.5 0	−3.0 −4.6	5.0 11.5	+4.0 0	−5.0 −7.5	8.0 18.0	+6.0 0	−8.0 −12.0
1.97–3.15	4.0 8.8	+3.0 0	−4.0 −5.8	6.0 13.5	+4.5 0	−6.0 −9.0	9.0 20.5	+7.0 0	−9.0 −13.5
3.15–4.73	5.0 10.7	+3.5 0	−5.0 −7.2	7.0 15.5	+5.0 0	−7.0 −10.5	10.0 24.0	+9.0 0	−10.0 −15.0
4.73–7.09	6.0 12.5	+4.0 0	−6.0 −8.5	8.0 18.0	+6.0 0	−8.0 −12.0	12.0 28.0	+10.0 0	−12.0 −18.0
7.09–9.85	7.0 14.3	+4.5 0	−7.0 −9.8	10.0 21.5	+7.0 0	−10.0 −14.5	15.0 34.0	+12.0 0	−15.0 −22.0
9.85–12.41	8.0 16.0	+5.0 0	−8.0 −11.0	12.0 25.0	+8.0 0	−12.0 −17.0	18.0 38.0	+12.0 0	−18.0 −26.0

[3]Table entries are in inches. Multiply the entries in *Limits of clearance* and *Standard limits* columns by 10^{-3}.

TABLE A.1 Clearance Fits[4] (*continued*)

	LC1			LC2			LC3		
Basic Size	Limits of Clearance	Standard Limits		Limits of Clearance	Standard Limits		Limits of Clearance	Standard Limits	
Over–To		Hole H6	Shaft h5		Hole H7	Shaft h6		Hole H8	Shaft h7
0–0.12	0 0.45	+0.25 0	0 −0.2	0 0.65	+0.4 0	0 −0.25	0 1	+0.6 0	0 −0.4
0.12–0.24	0 0.5	+0.3 0	0 −0.2	0 0.8	+0.5 0	0 −0.3	0 1.2	+0.7 0	0 −0.5
0.24–0.40	0 0.65	+0.4 0	0 −0.25	0 1.0	+0.6 0	0 −0.4	0 1.5	+0.9 0	0 −0.6
0.40–0.71	0 0.7	+0.4 0	0 −0.3	0 1.1	+0.7 0	0 −0.4	0 1.7	+1.0 0	0 −0.7
0.71–1.19	0 0.9	+0.5 0	0 −0.4	0 1.3	+0.8 0	0 −0.5	0 2	+1.2 0	0 −0.8
1.19–1.97	0 1.0	+0.6 0	0 −0.4	0 1.6	+1.0 0	0 −0.6	0 2.6	+1.6 0	0 −1
1.97–3.15	0 1.2	+0.7 0	0 −0.5	0 1.9	+1.2 0	0 −0.7	0 3	+1.8 0	0 −1.2
3.15–4.73	0 1.5	+0.9 0	0 −0.6	0 2.3	+1.4 0	0 −0.9	0 3.6	+2.2 0	0 −1.4
4.73–7.09	0 1.7	+1.0 0	0 −0.7	0 2.6	+1.6 0	0 −1.0	0 4.1	+2.5 0	0 −1.6
7.09–9.85	0 2.0	+1.2 0	0 −0.8	0 3.0	+1.8 0	0 −1.2	0 4.6	+2.8 0	0 −1.8
9.85–12.41	0 2.1	+1.2 0	0 −0.9	0 3.2	+2.0 0	0 −1.2	0 5	+3.0 0	0 −2.0

[4]Table entries are in inches. Multiply the entries in *Limits of clearance* and *Standard limits* columns by 10^{-3}.

TABLE A.1 Clearance Fits[5] (*continued*)

	LC4			LC5			LC6		
Basic Size	Limits of Clearance	Standard Limits		Limits of Clearance	Standard Limits		Limits of Clearance	Standard Limits	
Over–To		Hole H10	Shaft h9		Hole H7	Shaft g6		Hole H9	Shaft f8
0–0.12	0 2.6	+1.6 0	0 −1.0	0.1 0.75	+0.4 0	−0.1 −0.35	0.3 1.9	+1.0 0	−0.3 −0.9
0.12–0.24	0 3.0	+1.8 0	0 −1.2	0.15 0.95	+0.5 0	−0.15 −0.45	0.4 2.3	+1.2 0	−0.4 −1.1
0.24–0.40	0 3.6	+2.2 0	0 −1.4	0.2 1.2	+0.6 0	−0.2 −0.6	0.5 2.8	+1.4 0	−0.5 −1.4
0.40–0.71	0 4.4	+2.8 0	0 −1.6	0.25 1.35	+0.7 0	−0.25 −0.65	0.6 3.2	+1.6 0	−0.6 −1.6
0.71–1.19	0 5.5	+3.5 0	0 −2.0	0.3 1.6	+0.8 0	−0.3 −0.8	0.8 4.0	+2.0 0	−0.8 −2.0
1.19–1.97	0 6.5	+4.0 0	0 −2.5	0.4 2.0	+1.0 0	−0.4 −1.0	1.0 5.1	+2.5 0	−1.0 −2.6
1.97–3.15	0 7.5	+4.5 −0	0 −3	0.4 2.3	+1.2 0	−0.4 −1.1	1.2 6.0	+3.0 0	−1.2 −3.0
3.15–4.73	0 8.5	+5.0 0	0 −3.5	0.5 2.8	+1.4 0	−0.5 −1.4	1.4 7.1	+3.5 0	−1.4 −3.6
4.73–7.09	0 10	+6.0 0	0 −4	0.6 3.2	+1.6 0	−0.6 −1.6	1.6 8.1	+4.0 0	−1.6 −4.1
7.09–9.85	0 11.5	+7.0 0	0 −4.5	0.6 3.6	+1.8 0	−0.6 −1.8	2.0 9.3	+4.5 0	−2.0 −4.8
9.85–12.41	0 13	+8.0 0	0 −5	0.7 3.9	+2.0 0	−0.7 −1.9	2.2 10.2	+5.0 0	−2.2 −5.2

[5]Table entries are in inches. Multiply the entries in *Limits of clearance* and *Standard limits* columns by 10^{-3}.

TABLE A.1 Clearance Fits[6] (*continued*)									
	LC7			LC8			LC9		
Basic Size	**Limits of Clearance**	**Standard Limits**		**Limits of Clearance**	**Standard Limits**		**Limits of Clearance**	**Standard Limits**	
Over–To		**Hole H10**	**Shaft e9**		**Hole H10**	**Shaft d9**		**Hole H11**	**Shaft c10**
0–0.12	0.6 3.2	+1.6 0	−0.6 −1.6	1.0 3.6	+1.6 0	−1.0 −2.0	2.5 6.6	+2.5 0	−2.5 −4.1
0.12–0.24	0.8 3.8	+1.8 0	−0.8 −2.0	1.2 4.2	+1.8 0	−1.2 −2.4	2.8 7.6	+3.0 0	−2.8 −4.6
0.24–0.40	1.0 4.6	+2.2 0	−1.0 −2.4	1.6 5.2	+2.2 0	−1.6 −3.0	3.0 8.7	+3.5 0	−3.0 −5.2
0.40–0.71	1.2 5.6	+2.8 0	−1.2 −2.8	2.0 6.4	+2.8 0	−2.0 −3.6	3.5 10.3	+4.0 0	−3.5 −6.3
0.71–1.19	1.6 7.1	+3.5 0	−1.6 −3.6	2.5 8.0	+3.5 0	−2.5 −4.5	4.5 13.0	+5.0 0	−4.5 −8.0
1.19–1.97	2.0 8.5	+4.0 0	−2.0 −4.5	3.0 9.5	+4.0 0	−3.0 −5.5	5.0 15.0	+6.0 0	−5.0 −9.0
1.97–3.15	2.5 10.0	+4.5 0	−2.5 −5.5	4.0 11.5	+4.5 0	−4.0 −7.0	6.0 17.5	+7.0 0	−6.0 −10.5
3.15–4.73	3.0 11.5	+5.0 0	−3.0 −6.5	5.0 13.5	+5.0 0	−5.0 −8.5	7.0 21.0	+9.0 0	−7.0 −12.0
4.73–7.09	3.5 13.5	+6.0 0	−3.5 −7.5	6.0 16.0	+6.0 0	−6.0 −10.0	8.0 24.0	+10.0 0	−8.0 −14.0
7.09–9.85	4.0 15.5	+7.0 0	−4.0 −8.5	7.0 18.5	+7.0 0	−7.0 −11.5	10.0 29.0	+12.0 0	−10.0 −17.0
9.85–12.41	4.5 17.5	+8.0 0	−4.5 −9.5	7.0 20.0	+8.0 0	−7.0 −12.0	12.0 32.0	+12.0 0	−12.0 −20.0

[6]Table entries are in inches. Multiply the entries in *Limits of clearance* and *Standard limits* columns by 10^{-3}.

TABLE A.1 Clearance Fits[7] (*continued*)

	LC10			LC11		
Basic Size	Limits of Clearance	Standard Limits		Limits of Clearance	Standard Limits	
Over–To		Hole H12	Shaft c12		Hole H13	Shaft c13
0–0.12	4.0 12.0	+4.0 0	−4.0 −8.0	5.0 17.0	+6.0 0	−5.0 −11.0
0.12–0.24	4.5 14.5	+5.0 0	−4.5 −9.5	6.0 20.0	+7.0 0	−6.0 −13.0
0.24–0.40	5.0 17.0	+6.0 0	−5.0 −11.0	7.0 25.0	+9.0 0	−7.0 −16.0
0.40–0.71	6.0 20.0	+7.0 0	−6.0 −13.0	8.0 28.0	+10.0 0	−8.0 −18.0
0.71–1.19	7.0 23.0	+8.0 0	−7.0 −15.0	10.0 34.0	+12.0 0	−10.0 −22.0
1.19–1.97	8.0 28.0	+10.0 0	−8.0 −18.0	12.0 44.0	+16.0 0	−12.0 −28.0
1.97–3.15	10.0 34.0	+12.0 0	−10.0 −22.0	14.0 50.0	+18.0 0	−14.0 −32.0
3.15–4.73	11.0 39.0	+14.0 0	−11.0 −25.0	16.0 60.0	+22.0 0	−16.0 −38.0
4.73–7.09	12.0 44.0	+16.0 0	−12.0 −28.0	18.0 68.0	+25.0 0	−18.0 −43.0
7.09–9.85	16.0 52.0	+18.0 0	−16.0 −34.0	22.0 78.0	+28.0 0	−22.0 −50.0
9.85–12.41	20.0 60.0	+20.0 0	−20.0 −40.0	28.0 88.0	+30.0 0	−28.0 −58.0

[7]Table entries are in inches. Multiply the entries in *Limits of clearance* and *Standard limits* columns by 10^{-3}.

TABLE A.2 Transition Fits[8]

	LT1			LT2			LT3		
Basic Size	Limits of Fit	Standard Limits		Limits of Fit	Standard Limits		Limits of Fit	Standard Limits	
Over–To		Hole H7	Shaft js6		Hole H8	Shaft js7		Hole H7	Shaft k6
0–0.12	+0.50	0	−0.10	+0.8	0	−0.2			
0.12–0.24	−0.15 +0.65	+0.5 0	+0.15 −0.15	−0.25 +0.95	+0.7 0	+0.25 −0.25			
0.24–0.40	−0.2 +0.8	+0.6 0	+0.2 −0.2	−0.3 +1.2	+0.9 0	+0.3 −0.3	−0.5 +0.5	+0.6 0	+0.5 +0.1
0.40–0.71	−0.2 +0.9	+0.7 0	+0.2 −0.2	−0.35 +1.35	+1.0 0	+0.35 −0.35	−0.5 +0.6	+0.7 0	+0.5 +0.1
0.71–1.19	−0.25 +1.05	+0.8 0	+0.25 −0.25	−0.4 +1.6	+1.2 0	+0.4 −0.4	−0.6 +0.7	+0.8 0	+0.6 +0.1
1.19–1.97	−0.3 +1.3	+1.0 0	+0.3 −0.3	−0.5 +2.1	+1.6 0	+0.5 −0.5	−0.7 +0.9	+1.0 0	+0.7 +0.1
1.97–3.15	−0.3 +1.5	+1.2 0	+0.3 −0.3	−0.6 +2.4	+1.8 0	+0.6 −0.6	−0.8 +1.1	+1.2 0	+0.8 +0.1
3.15–4.73	−0.4 +1.8	+1.4 0	+0.4 −0.4	−0.7 +2.9	+2.2 0	+0.7 −0.7	−1.0 +1.3	+1.4 0	+1.0 +0.1
4.73–7.09	−0.5 +2.1	+1.6 0	+0.5 −0.5	−0.8 +3.3	+2.5 0	+0.8 −0.8	−1.1 +1.5	+1.6 0	+1.1 +0.1
7.09–9.85	−0.6 +2.4	+1.8 0	+0.6 −0.6	−0.9 +3.7	+2.8 0	+0.9 −0.9	−1.4 +1.6	+1.8 0	+1.4 +0.2
9.85–12.41	−0.6 +2.6	+2.0 0	+0.6 −0.6	−1.0 +4.0	+3.0 0	+1.0 −1.0	−1.4 +1.8	+2.0 0	+1.4 +0.2
12.41–15.75	−0.7 +2.9	+2.2 0	+0.7 −0.7	−1.0 +4.5	+3.5 0	+1.0 −1.0	−1.6 +2.0	+2.2 0	+1.6 +0.2
15.75–19.69	−0.8 +3.3	+2.5 0	+0.8 −0.8	−1.2 +5.2	+4.0 0	+1.2 −1.2	−1.8 +2.3	+2.5 0	+1.8 +0.2

[8]Table entries are in inches. Multiply the entries in *Limits of fit* and *Standard limits* columns by 10^{-3}.

TABLE A.2 Transition Fits[9] (*continued*)									
	LT4			LT5			LT6		
Basic Size	Limits of Fit	Standard Limits		Limits of Fit	Standard Limits		Limits of Fit	Standard Limits	
Over–To		Hole H8	Shaft k7		Hole H7	Shaft n6		Hole H7	Shaft n7
0–0.12				−0.5 +0.15	+0.4 0	+0.5 +0.25	−0.65 +0.15	+0.4 0	+0.65 +0.25
0.12–0.24				−0.6 +0.2	+0.5 0	+0.6 +0.3	−0.8 +0.2	+0.5 0	+0.8 +0.3
0.24–0.40	−0.7 +0.8	+0.9 0	+0.7 +0.1	−0.8 +0.2	+0.6 0	+0.8 +0.4	−1.0 +0.2	+0.6 0	+1.0 +0.4
0.40–0.71	−0.8 +0.9	+1.0 0	+0.8 +0.1	−0.9 +0.2	+0.7 0	+0.9 +0.5	−1.2 +0.2	+0.7 0	+1.2 +0.5
0.71–1.19	−0.9 +1.1	+1.2 0	+0.9 +0.1	−1.1 +0.2	+0.8 0	+1.1 +0.6	−1.4 +0.2	+0.8 0	+1.4 +0.6
1.19–1.97	−1.1 +1.5	+1.6 0	+1.1 +0.1	−1.3 +0.3	+1.0 0	+1.3 +0.7	−1.7 +0.3	+1.0 0	+1.7 +0.7
1.97–3.15	−1.3 +1.7	+1.8 0	+1.3 +0.1	−1.5 +0.4	+1.2 0	+1.5 +0.8	−2.0 +0.4	+1.2 0	+2.0 +0.8
3.15–4.73	−1.5 +2.1	+2.2 0	+1.5 +0.1	−1.9 +0.4	+1.4 0	+1.9 +1.0	−2.4 +0.4	+1.4 0	+2.4 +1.0
4.73–7.09	−1.7 +2.4	+2.5 0	+1.7 +0.1	−2.2 +0.4	+1.6 0	+2.2 +1.2	−2.8 +0.4	+1.6 0	+2.8 +1.2
7.09–9.85	−2.0 +2.6	+2.8 0	+2.0 +0.2	−2.6 +0.4	+1.8 0	+2.6 +1.4	−3.2 +0.4	+1.8 0	+3.2 +1.4
9.85–12.41	−2.2 +2.8	+3.0 0	+2.2 +0.2	−2.6 +0.6	+2.0 0	+2.6 +1.4	−3.4 +0.6	+2.0 0	+3.4 +1.4
12.41–15.75	−2.4 +3.3	+3.5 0	+2.4 +0.2	−3.0 +0.6	+2.2 0	+3.0 +1.6	−3.8 +0.6	+2.2 0	+3.8 +1.6
15.75–19.69	−2.7 +3.8	+4.0 0	+2.7 +0.2	−3.4 +0.6	+2.5 0	+3.4 +1.8	−4.3 +0.7	+2.5 0	+4.3 +1.8

[9]Table entries are in inches. Multiply the entries in *Limits of fit* and *Standard limits* columns by 10^{-3}.

TABLE A.3 Interference Fits[10]

	LN1			LN2			LN3		
Basic Size	Limits of Interference	Standard Limits		Limits of Interference	Standard Limits		Limits of Interference	Standard Limits	
Over–To		Hole H6	Shaft n5		Hole H7	Shaft p6		Hole H7	Shaft r6
0–0.12	0 0.45	+0.25 0	+0.45 +0.25	0 0.65	+0.4 0	+0.65 +0.4	0.1 0.75	+0.4 0	+0.75 +0.5
0.12–0.24	0 0.5	+0.3 0	+0.5 +0.3	0 0.8	+0.5 0	+0.8 +0.5	0.1 0.9	+0.5 0	+0.9 +0.6
0.24–0.40	0 0.65	+0.4 0	+0.65 +0.4	0 1.0	+0.6 0	+1.0 +0.6	0.2 1.2	+0.6 0	+1.2 +0.8
0.40–0.71	0 0.8	+0.4 0	+0.8 +0.4	0 1.1	+0.7 0	+1.1 +0.7	0.3 1.4	+0.7 0	+1.4 +1.0
0.71–1.19	0 1.0	+0.5 0	+1.0 +0.5	0 1.3	+0.8 0	+1.3 +0.8	0.4 1.7	+0.8 0	+1.7 +1.2
1.19–1.97	0 1.1	+0.6 0	+1.1 +0.6	0 1.6	+1.0 0	+1.6 +1.0	0.4 2.0	+1.0 0	+2.0 +1.4
1.97–3.15	0.1 1.3	+0.7 0	+1.3 +0.7	0.2 2.1	+1.2 0	+2.1 +1.4	0.4 2.3	+1.2 0	+2.3 +1.6
3.15–4.73	0.1 1.6	+0.9 0	+1.6 +1.0	0.2 2.5	+1.4 0	+2.4 +1.6	0.6 2.9	+1.4 0	+2.9 +2.0
4.73–7.09	0.2 1.9	+1.0 0	+1.9 +1.2	0.2 2.8	+1.6 0	+2.8 +1.8	0.9 3.5	+1.6 0	+3.5 +2.5
7.09–9.85	0.2 2.2	+1.2 0	+2.2 +1.4	0.2 3.2	+1.8 0	+3.2 +2.0	1.2 4.2	+1.8 0	+4.2 +3.0
9.85–12.41	0.2 2.3	+1.2 0	+2.3 +1.4	0.2 3.4	+2.0 0	+3.4 +2.2	1.5 4.7	+2.0 0	+4.7 +3.5

[10]Table entries are in inches. Multiply the entries in *Limits of interference* and *Standard limits* columns by 10^{-3}.

TABLE A.3 Interference Fits[11] (*continued*)

	FN1			FN2			FN3		
Basic Size	Limits of Interference	Standard Limits		Limits of Interference	Standard Limits		Limits of Interference	Standard Limits	
Over–To		Hole H6	Shaft n6		Hole H7	Shaft s6		Hole H7	Shaft t6
0–0.12	0.05 0.5	+0.25 0	+0.5 +0.3	0.2 0.85	+0.4 0	+0.85 +0.6			
0.12–0.24	0.1 0.6	+0.3 0	+0.6 +0.4	0.2 1.0	+0.5 0	+1.0 +0.7			
0.24–0.40	0.1 0.75	+0.4 0	+0.75 +0.5	0.4 1.4	+0.6 0	+1.4 +1.0			
0.40–0.56	0.1 0.8	+0.4 0	+0.8 +0.5	0.5 1.6	+0.7 0	+1.6 +1.2			
0.56–0.71	0.2 0.9	+0.4 0	+0.9 +0.6	0.5 1.6	+0.7 0	+1.6 +1.2			
0.71–0.95	0.2 1.1	+0.5 0	+1.1 +0.7	0.6 1.9	+0.8 0	+1.9 +1.4			
0.95–1.19	0.3 1.2	+0.5 0	+1.2 +0.8	0.6 1.9	+0.8 0	+1.9 +1.4	0.8 2.1	+0.8 0	+2.1 +1.6
1.19–1.58	0.3 1.3	+0.6 0	+1.3 +0.9	0.8 2.4	+1.0 0	+2.4 +1.8	1.0 2.6	+1.0 0	+2.6 +2.0
1.58–1.97	0.4 1.4	+0.6 0	+1.4 +1.0	0.8 2.4	+1.0 0	+2.4 +1.8	1.2 2.8	+1.0 0	+2.8 +2.2
1.97–2.56	0.6 1.8	+0.7 0	+1.8 +1.3	0.8 2.7	+1.2 0	+2.7 +2.0	1.3 3.2	+1.2 0	+3.2 +2.5
2.56–3.15	0.7 1.9	+0.7 0	+1.9 +1.4	1.0 2.9	+1.2 0	+2.9 +2.2	1.8 3.7	+1.2 0	+3.7 +3.8
3.15–3.94	0.9 2.4	+0.9 0	+2.4 +1.8	1.4 3.7	+1.4 0	+3.7 +2.8	2.1 4.4	+1.4 0	+4.4 +3.5
3.94–4.73	1.1 2.6	+0.9 0	+2.6 +2.0	1.6 3.9	+1.4 0	+3.9 +3.0	2.6 4.9	+1.4 0	+4.9 +4.0
4.73–5.52	1.2 2.9	+1.0 0	+2.9 +2.2	1.9 4.5	+1.6 0	+4.5 +3.5	3.4 6.0	+1.6 0	+6.0 +5.0
5.52–6.30	1.5 3.2	+1.0 0	+3.2 +2.5	2.4 5.0	+1.6 0	+5.0 +4.0	3.4 6.0	+1.6 0	+6.0 +5.0
6.30–7.09	1.8 3.5	+1.0 0	+3.5 +2.8	2.9 5.5	+1.6 0	+5.5 +4.5	4.4 7.0	+1.6 0	+7.0 +6.0
7.09–7.88	1.8 3.8	+1.2 0	+3.8 +3.0	3.2 6.2	+1.8 0	+6.2 +5.0	5.2 8.2	+1.8 0	+8.2 +7.0
7.88–8.86	2.3 4.3	+1.2 0	+4.3 +3.5	3.2 6.2	+1.8 0	+6.2 +5.0	5.2 8.2	+1.8 0	+8.2 +7.0
8.86–9.85	2.3 4.3	+1.2 0	+4.3 +3.5	4.2 7.2	+1.8 0	+7.2 +6.0	6.2 9.2	+1.8 0	+9.2 +8.0
9.85–11.03	2.8 4.9	+1.2 0	+4.9 +4.0	4.0 7.2	+2.0 0	+7.2 +6.0	7.0 10.2	+2.0 0	+10.2 +9.0
11.03–12.41	2.8 4.9	+1.2 0	+4.9 +4.0	5.0 8.2	+2.0 0	+8.2 +7.0	7.0 10.2	+2.0 0	+10.2 +9.0

[11]Table entries are in inches. Multiply the entries in *Limits of interference* and *Standard limits* columns by 10^{-3}.

TABLE A.3 Interference Fits[12] (*continued*)

	FN4			FN5		
Basic Size	**Limits of Interference**	**Standard Limits**		**Limits of Interference**	**Standard Limits**	
Over–To		**Hole H7**	**Shaft u6**		**Hole H8**	**Shaft x7**
0–0.12	0.3 0.95	+0.4 0	+0.95 +0.7	0.3 1.3	+0.6 0	+1.3 +0.9
0.12–0.24	0.4 1.2	+0.5 0	+1.2 +0.9	0.5 1.7	+0.7 0	+1.7 +1.2
0.24–0.40	0.6 1.6	+0.6 0	+1.6 +1.2	0.5 2.0	+0.9 0	+2.0 +1.4
0.40–0.56	0.7 1.8	+0.7 0	+1.8 +1.4	0.6 2.3	+1.0 0	+2.3 +1.6
0.56–0.71	0.7 1.8	+0.7 0	+1.8 +1.4	0.8 2.5	+1.0 0	+2.5 +1.8
0.71–0.95	0.8 2.1	+0.8 0	+2.1 +1.6	1.0 3.0	+1.2 0	+3.0 +2.2
0.95–1.19	1.0 2.3	+0.8 0	+2.3 +1.8	1.3 3.3	+1.2 0	+3.3 +2.5
1.19–1.58	1.5 3.1	+1.0 0	+3.1 +2.5	1.4 4.0	+1.6 0	+4.0 +3.0
1.58–1.97	1.8 3.4	+1.0 0	+3.4 +2.8	2.4 5.0	+1.6 0	+5.0 +4.0
1.97–2.56	2.3 4.2	+1.2 0	+4.2 +3.5	3.2 6.2	+1.8 0	+6.2 +5.0
2.56–3.15	2.8 4.7	+1.2 0	+4.7 +4.0	4.2 7.2	+1.8 0	+7.2 +6.0
3.15–3.94	3.6 5.9	+1.4 0	+5.9 +5.0	4.8 8.4	+2.2 0	+8.4 +7.0
3.94–4.73	4.6 6.9	+1.4 0	+6.9 +6.0	5.8 9.4	+2.2 0	+9.4 +8.0
4.73–5.52	5.4 8.0	+1.6 0	+8.0 +7.0	7.5 11.6	+2.5 0	+11.6 +10.0
5.52–6.30	5.4 8.0	+1.6 0	+8.0 +7.0	9.5 13.6	+2.5 0	+13.6 +12.0
6.30–7.09	6.4 9.0	+1.6 0	+9.0 +8.0	9.5 13.6	+2.5 0	+13.6 +12.0
7.09–7.88	7.2 10.2	+1.8 0	+10.2 +9.0	11.2 15.8	+2.8 0	+15.8 +14.0
7.88–8.86	8.2 11.2	+1.8 0	+11.2 +10.0	13.2 17.8	+2.8 0	+17.8 +16.0
8.86–9.85	10.2 13.2	+1.8 0	+13.2 +12.0	13.2 17.8	+2.8 0	+17.8 +16.0
9.85–11.03	10.2 13.2	+2.0 0	+13.2 +12.0	15.0 20.0	+3.0 0	+20.0 +18.0
11.03–12.41	12.0 15.2	+2.0 0	+15.2 +14.0	17.0 22.0	+3.0 0	+22.0 +20.0

[12]Table entries are in inches. Multiply the entries in *Limits of interference* and *Standard limits* columns by 10^{-3}.

TABLE A.4 Tolerance Grades—U.S. (English) Units[13]						
Basic Size	**Tolerance Grade**					
Over–To	**IT6**	**IT7**	**IT8**	**IT9**	**IT10**	**IT11**
0–0.12	0.2	0.4	0.6	1.0	1.6	2.4
0.12–0.24	0.3	0.5	0.7	1.2	1.9	3.0
0.24–0.40	0.4	0.6	0.9	1.4	2.3	3.5
0.40–0.72	0.4	0.7	1.1	1.7	2.8	4.3
0.72–1.20	0.5	0.8	1.3	2.0	3.3	5.1
1.20–2.00	0.6	1.0	1.5	2.4	3.9	6.3
2.00–3.20	0.7	1.2	1.8	2.9	4.7	7.5
3.20–4.80	0.9	1.4	2.1	3.4	5.5	8.7
4.80–7.20	1.0	1.6	2.5	3.9	6.3	9.8
7.20–10.00	1.1	1.8	2.8	4.5	7.3	11.4
10.00–12.60	1.3	2.0	3.2	5.1	8.3	12.6
12.60–16.00	1.4	2.2	3.5	5.5	9.1	14.2

[13]Table entries are in inches. Multiply the entries in *Tolerance grade* columns by 10^{-3}.

TABLE A.5 Fundamental Deviation—US (English) Units[14]										
Basic Size	**Shaft Tolerance Symbol**									
Over–To	**c**	**d**	**f**	**g**	**h**	**k**	**n**	**p**	**s**	**u**
0–0.12	−2.4	−0.8	−0.2	−0.1	0	0	+0.2	+0.2	+0.6	+0.7
0.12–0.24	−2.8	−1.2	−0.4	−0.2	0	0	+0.3	+0.5	+0.7	+0.9
0.24–0.40	−3.1	−1.6	−0.5	−0.2	0	0	+0.4	+0.6	+0.9	+1.1
0.40–0.72	−3.7	−2.0	−0.6	−0.2	0	0	+0.5	+0.7	+1.1	+1.3
0.72–0.96	−4.3	−2.6	−0.8	−0.3	0	+0.1	+0.6	+0.9	+1.4	+1.6
0.96–1.20	−4.3	−2.6	−0.8	−0.3	0	+0.1	+0.6	+0.9	+1.4	+1.9
1.20–1.60	−4.7	−3.1	−1.0	−0.4	0	+0.1	+0.7	+1.0	+1.7	+2.4
1.60–2.00	−5.1	−3.1	−1.0	−0.4	0	+0.1	+0.7	+1.0	+1.7	+2.8
2.00–2.60	−5.5	−3.9	−1.2	−0.4	0	+0.1	+0.8	+1.3	+2.1	+3.4
2.60–3.20	−5.9	−3.9	−1.2	−0.4	0	+0.1	+0.8	+1.3	+2.3	+4.0
3.20–4.00	−6.7	−4.7	−1.4	−0.5	0	+0.1	+0.9	+1.5	+2.8	+4.9
4.00–4.80	−7.1	−4.7	−1.4	−0.5	0	+0.1	+0.9	+1.5	+3.1	+5.7
4.80–5.60	−7.9	−5.7	−1.7	−0.6	0	+0.1	+1.1	+1.7	+3.6	+6.7
5.60–6.40	−8.3	−5.7	−1.7	−0.6	0	+0.1	+1.1	+1.7	+3.9	+7.5
6.40–7.20	−9.1	−5.7	−1.7	−0.6	0	+0.1	+1.1	+1.7	+4.3	+8.3
7.20–8.00	−9.4	−6.7	−2.0	−0.6	0	+0.2	+1.2	+2.0	+4.8	+9.3
8.00–9.00	−10.2	−6.7	−2.0	−0.6	0	+0.2	+1.2	+2.0	+5.1	+10.2
9.00–10.00	−11.0	−6.7	−2.0	−0.6	0	+0.2	+1.2	+2.0	+5.5	+11.2
10.00–11.20	−11.8	−7.5	−2.2	−0.7	0	+0.2	+1.3	+2.2	+6.2	+12.4
11.20–12.60	−13.0	−7.5	−2.2	−0.7	0	+0.2	+1.3	+2.2	+6.7	+13.0
12.60–14.20	−14.2	−8.3	−2.4	−0.7	0	+0.2	+1.5	+2.4	+7.5	+15.4
14.20–16.00	−15.7	−8.3	−2.4	−0.7	0	+0.2	+1.5	+2.4	+8.2	+17.1

[14]Table entries are in inches. Multiply the entries in *Shaft tolerance symbol* columns by 10^{-3}.

TABLE A.6 Tolerance Grades—Metric (SI) Units[15]						
Basic Size	Tolerance Grade					
Over–To	IT6	IT7	IT8	IT9	IT10	IT11
0–3	6	10	14	25	40	60
3–6	8	12	18	30	48	75
6–10	9	15	22	36	58	90
10–18	11	18	27	43	70	110
18–30	13	21	33	52	84	130
30–50	16	25	39	62	100	160
50–80	19	30	46	74	120	190
80–120	22	35	54	87	140	220
120–180	25	40	63	100	160	250
180–250	29	46	72	115	185	290
250–315	32	52	81	130	210	320
315–400	36	57	89	140	230	360

[15]Table entries are in millimeters. Multiply the entries in *Tolerance grade* columns by 10^{-3}.

TABLE A.7 Fundamental Deviation—Metric (SI) Units[16]										
Basic Size	Shaft Tolerance Symbol									
Over–To	c	d	f	g	h	k	n	p	s	u
0–3	−60	−20	−6	−2	0	0	+4	+6	+14	+18
3–6	−70	−30	−10	−4	0	+1	+8	+12	+19	+23
6–10	−80	−40	−13	−5	0	+1	+10	+15	+23	+28
10–14	−95	−50	−16	−6	0	+1	+12	+18	+28	+33
14–18	−95	−50	−16	−6	0	+1	+12	+18	+28	+33
18–24	−110	−65	−20	−7	0	+2	+15	+22	+35	+41
24–30	−110	−65	−20	−7	0	+2	+15	+22	+35	+48
30–40	−120	−80	−25	−9	0	+2	+17	+26	+43	+60
40–50	−130	−80	−25	−9	0	+2	+17	+26	+43	+70
50–65	−140	−100	−30	−10	0	+2	+20	+32	+53	+87
65–80	−150	−100	−30	−10	0	+2	+20	+32	+59	+102
80–100	−170	−120	−36	−12	0	+3	+23	+37	+71	+124
100–120	−180	−120	−36	−12	0	+3	+23	+37	+79	+144
120–140	−200	−145	−43	−14	0	+3	+27	+43	+92	+170
140–160	−210	−145	−43	−14	0	+3	+27	+43	+100	+190
160–180	−230	−145	−43	−14	0	+3	+27	+43	+108	+210
180–200	−240	−170	−50	−15	0	+4	+31	+50	+122	+236
200–225	−260	−170	−50	−15	0	+4	+31	+50	+130	+258
225–250	−280	−170	−50	−15	0	+4	+31	+50	+140	+284
250–280	−300	−190	−56	−17	0	+4	+34	+56	+158	+315
280–315	−330	−190	−56	−17	0	+4	+34	+56	+170	+350
315–355	−360	−210	−62	−18	0	+4	+37	+62	+190	+390
355–400	−400	−210	−62	−18	0	+4	+37	+62	+208	+435

[16]Table entries are in millimeters. Multiply the entries in *Shaft tolerance symbol* columns by 10^{-3}.

APPENDIX

B

SolidWorks Certification

SolidWorks offers certification exams to become a SolidWorks certified designer. Certification exams are common in the professional world and they exist in various professions and trades. In the software field, for example, you may become a Microsoft certified Windows programmer or a certified Oracle database programmer. In the CAD/CAM field, you may become a SolidWorks, Pro/E, or AutoCAD certified designer.

The first certification exam of SolidWorks focuses on the CAD/CAM fundamentals we have covered in this book. A typical SolidWorks certification exam consists of two or more sections. One section asks about the fundamentals. Other sections are related to modeling skills. These sections require knowing and mastering SolidWorks software, reading a drawing, and visualizing a model. The better SolidWorks user you are, the faster you can finish these sections and finish them correctly. Exams usually test basic modeling skills as well as advanced skills.

The value of a SolidWorks certificate is to increase your competitiveness in the job market and help your career plans. Some companies only hire certified SolidWorks designers. Different certification levels exist. There is a license for each SolidWorks software module that we have covered in this book. Visit www.solidworks.com/sw/mcad-certification-programs.htm for more details. All license exams are online, and you can take them in test centers around the world. The website covers the details of the online testing process. You must earn a minimum score to pass any exam and become certified in the test-related subject. The available SolidWorks exams are listed below.

1. Certified SolidWorks Associate (CSWA)

This is the most basic certificate you can earn. The exam is 3 hours long and the minimum passing grade is 70%. It covers the basic modeling concepts. The exam has a written part and a modeling part. The following table shows a mapping of the exam topics to the book chapters to prepare and study for the certification exam:

Exam Topic	Book Chapter and Section to Study
Sketch entities (lines, rectangles, circles, arcs, ellipses, centerlines)	Chapter 1 Chapter 2
Sketch tools (offset, convert, trim) and sketch relations, and reference geometry (plane, axis, mate reference)	Chapter 1 Chapter 2
Boss and cut features (extrudes, revolves, sweeps, lofts), fillets and chamfers, patterns (linear, circular, fill), and feature conditions (start and end)	Chapter 2 Chapter 4 (Sections 4.1 and 4.2, and Tutorials 4–1 and 4–2)
Drawing sheets and views, dimensions and model items, and annotations	Chapter 5
Mass properties, materials	Chapter 13 (Sections 13.1 and 13.3, and Tutorial 13–3)
Assembly operations: insert components, standard mates (coincident, parallel, perpendicular, tangent, concentric, distance, angle)	Chapter 6

2. Certified SolidWorks Professional (CSWP)

This is the next level of certification above the CSWA. The exam is 3 hours and 40 minutes long and the minimum passing grade is 75%. It covers the design and analysis of parametric parts and assemblies with complex features. The exam has three separate segments that you can take separately. The following table shows a mapping of the exam topics to the book chapters to prepare and study for the certification exam:

Exam Topic	Book Chapter and Section to Study
Segment 1 Topics (90 minutes)	
Create part from drawing	Chapter 5
Associative dimensions and equations	Chapter 5
Mass property analysis	Chapter 13 (Sections 13.1 and 13.3)
Editing existing parts	Chapter 2 Chapter 3
Segment 2 Topics (40 minutes)	
Configurations	Chapter 3 Chapter 5
Mass properties	Chapter 13 (Sections 13.1 and 13.3)
Edit part features	Chapter 2 Chapter 5
Segment 3 Topics (80 minutes)	
Assembly operations: add parts, collision detection, create coordinate system, mass properties	Chapter 6 Chapter 13 (Sections 13.1 and 13.3)

3. CSWP – Surfacing

CSWP – Core tests competency in parameteric modeling in general. Other exams focus on testing competencies in specific areas. CSWP – Surfacing exam and the ones that follow are area-specific exams. CSWP – Surfacing tests proficiency in surfaces. The exam is 90 minutes long and the minimum passing grade is 75%. The following table shows a mapping of the exam topics to the book chapters to prepare and study for the certification exam:

Exam Topic	Book Chapter and Section to Study
Create splines	Chapter 8
Create 3D curves	Chapter 8
Create surfaces: boundary, filled, swept, planar, knit, trim, fillet, thicken	Chapter 9
Edit surfaces: trim, move	Chapter 9

4. CSWP – Sheet Metal

CSWP – Sheet Metal tests proficiency in sheet metal. The exam is 2 hours long and the minimum passing grade is 75%. The following table shows a mapping of the exam topics to the book chapters to prepare and study for the certification exam:

Exam Topic	Book Chapter and Section to Study
Create flanges: linear, curved, miter	Chapter 10
Create closed corners and gauge tables	Chapter 10
Perform bending calculations: bend allowance, bend deduction, K-Factor	Chapter 10
Create sheet metal features: hem, jog, bend,	Chapter 10
Perform sheet metal operations: forming tool, unfold and fold, flatten	Chapter 10

5. CSWP – Weldments

CSWP – Weldments tests proficiency in weldments. The exam is 2 hours long and the minimum passing grade is 75%. The following table shows a mapping of the exam topics to the book chapters to prepare and study for the certification exam:

Exam Topic	Book Chapter and Section to Study
Create weldment profiles	Chapter 10
Create and modify weldment parts	Chapter 10
Create and modify weldment corners	Chapter 10
Create weldment features: end caps, gussets	Chapter 10
Manage cut lists in part and drawing	Chapter 10

6. CSWP – FEA

CSWP – FEA tests proficiency in FEA. The exam is 90 minutes long and the minimum passing grade is 75%. The following table shows a mapping of the exam topics to the book chapters to prepare and study for the certification exam:

Exam Topic	Book Chapter and Section to Study
Static, frequency, and thermal studies	Chapter 13 (Sections 13.7, 13.8, 13.9)
Working with solid and shell elements	Chapter 13 (Sections 13.7, 13.8, 13.9)
Working with beams	Chapter 13 (Sections 13.7, 13.8, 13.9)
Applying loads and restraints	Chapter 13 (Sections 13.7, 13.8, 13.9)
Planes of symmetry	Chapter 13 (Sections 13.7, 13.8, 13.9)
Nonuniform loads	Chapter 13 (Sections 13.7, 13.8, 13.9)
Mesh controls	Chapter 13 (Sections 13.7, 13.8, 13.9)
Use of configurations to compare designs	Chapter 13 (Sections 13.7, 13.8, 13.9)
Convergence of results	Chapter 13 (Sections 13.7, 13.8, 13.9)
Result forces	Chapter 13 (Sections 13.7, 13.8, 13.9)

7. CSWP – Mold Tools

CSWP – Mold Tools tests proficiency in mold design. The exam is 90 minutes long and the minimum passing grade is 80%. The following table shows a mapping of the exam topics to the book chapters to prepare and study for the certification exam:

Exam Topic	Book Chapter and Section to Study
Create mold features: parting line, parting surface, shut-off surface	Chapter 16
Create tooling split	Chapter 16
Perform draft analysis	Chapter 16

Multiple academic institutions around the world use SolidWorlds certification exams to test the proficiency of their students in their CAD/CAM courses. Some instructors use these exams as supplements to their traditional measures of evaluation such as homework and projects. In this case, the instructor's institution becomes a center (provider) of the exam. If you are interested, check these resources:

1. To apply to become a CSWA provider, visit www.solidworks.com/CSWAProvider.
2. For information on CSWA, visit www.solidworks.com/cswa.
3. Virtual Testing Center (where SolidWorks keeps track of test takers over their careers and employers can verify certificates), visit www.virtualtester.com/solidworks.

The remainder of the appendix shows sample test questions.

Certified SolidWorks Associate (CSWA)

Website: www.solidworks.com/cswa
Exam Length: 3 hours (180 minutes)
Passing Grade: 165/240 (68.75%)
Total number of questions: 14 questions as follows:

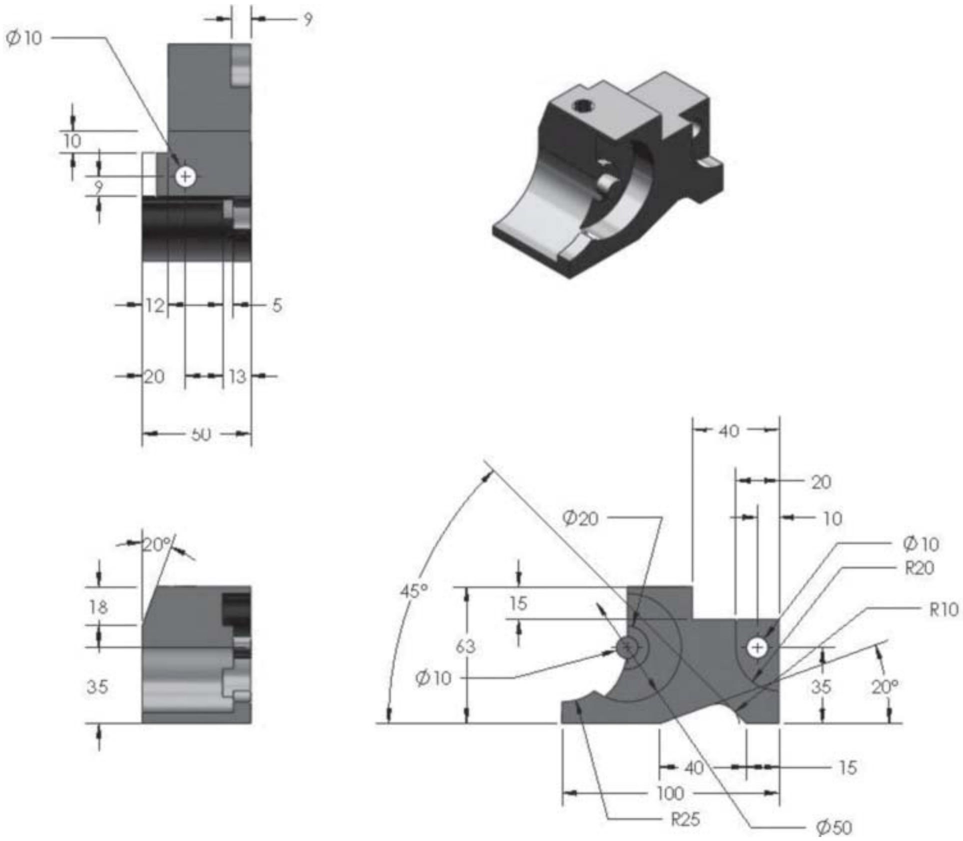
Topic	No. of questions	Points/question	Total Points
Drawing Competencies	3	5	15
Basic Part Modeling	2	15	30
Intermediate Part Modeling	2	15	30
Advanced Part Modeling	3	15	45
Assembly Modeling	4	30	120
Total	**14**	**Total**	**240**

Sample Exam Questions

The questions below represent sample CSWA Exam questions.

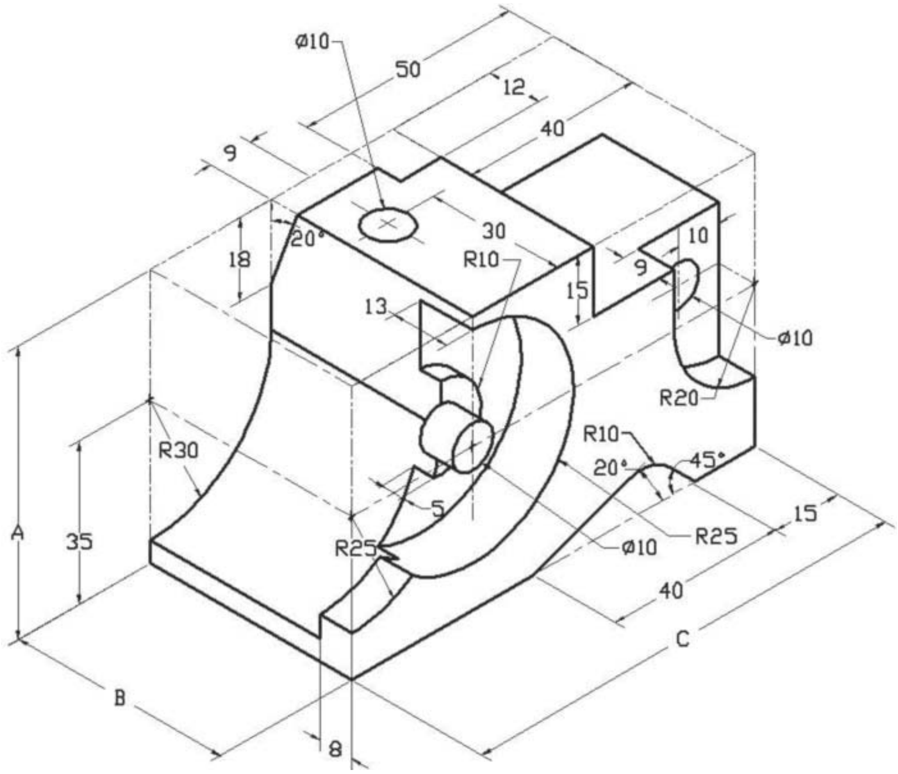
Question 1

Build this part in SolidWorks. Unit system: MMGS (millimeter, gram, second). Decimal places: 2. Part origin: Arbitrary, A = 63 mm, B = 50 mm, C = 100 mm. All holes through all. Part material: Copper. Density = 0.0089 g/mm^3



What is the overall mass of the part in grams?

a. 1205
b. 1280
c. 144
d. 1108



Question 2

COSMOSXPress allows changes to mesh settings. Which of the following statements is not true?

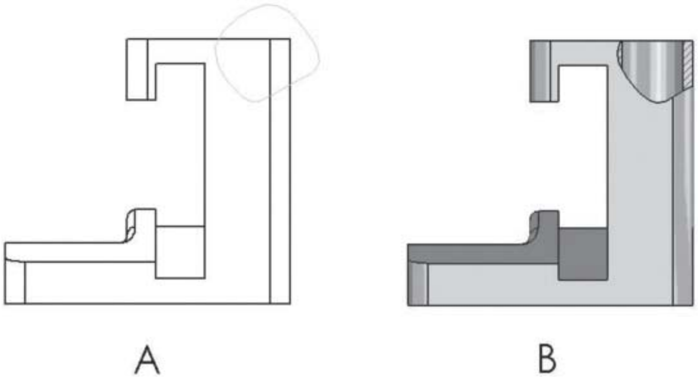
a. A fine mesh setting produces more accurate results than a coarse mesh.
b. A coarse mesh setting produces less accurate results than a fine mesh.
c. A fine mesh setting can be applied to a specific face instead of the entire model.
d. All of the above

Question 3

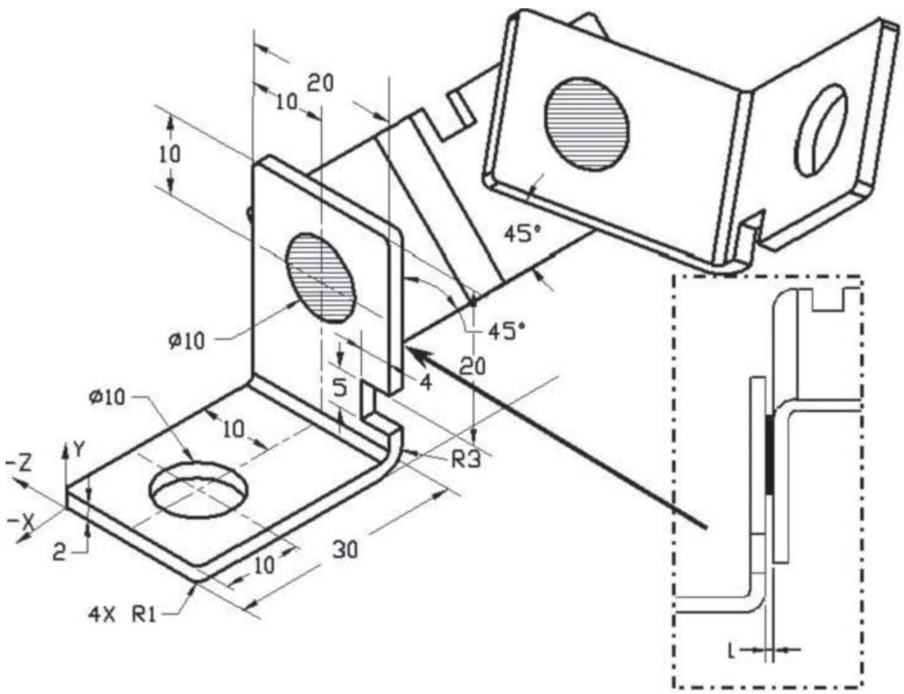
To create drawing view B, it is necessary to sketch a spline (as shown) on drawing view A and insert which SolidWorks view type?

a. Broken-out section
b. Aligned section
c. Section
d. Detail

Build this assembly in SolidWorks.



Question 4



Build this assembly in SolidWorks.
It contains 3 machined brackets and 2 pins.

Brackets: 2 mm thickness, and equal size (holes through-all). Materials: 6061 Alloy, Density = 0.0027g/mm^3. The top edge of the notch is located 20 mm from the top edge of the machined bracket.

Pins: 5 mm length and equal in diameter, Material: Titanium, Density = 0.0046g/mm^3. Pins are mated concentric to bracket holes (no clearance). Pin end faces are coincident to bracket outer faces. There is a 1 mm gap between the brackets. Brackets are positioned with equal angle mates (45 degrees).

Unit system: MMGS (millimeter, gram, second). Decimal places. 2. Assembly origin: As shown. What is the center of mass of the assembly?

a. X = −11.05, Y = 24.08, Z = −40.19

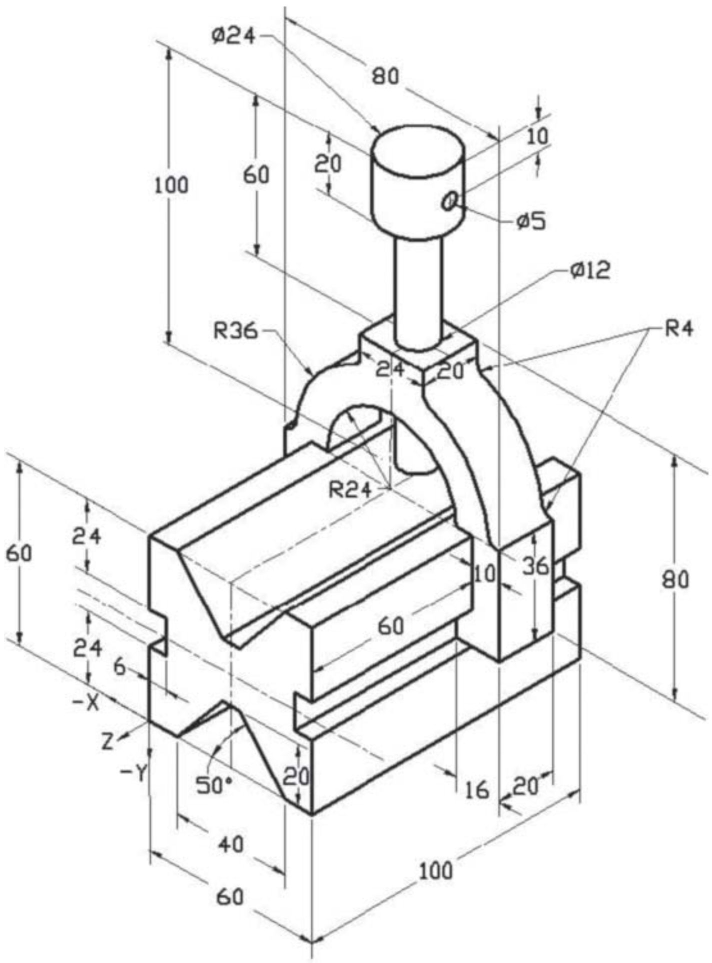
b. X = −11.05, Y = −24.08, Z = 40.19

c. X = 40.24, Y = 24.33, Z = 20.75

d. X = 20.75, Y = 24.33, Z = 40.24

Question 5

Build this assembly in SolidWorks. It contains 3 components: base, yoke, adjusting pin. Apply the MMGS unit system.



Material: 1060 Alloy for all components. Density = 0.0027g/mm^3

Base: The distance between the front face of the base and the front face of the yoke = 60 mm.

Yoke: The yoke fits inside the left and right square channels of the base component (no clearance). The top face of the yoke contains a Ø12 mm through-all hole.

Adjusting Pin: The bottom face of the adjusting pin head is located 40 mm from the top face of the yoke component. The adjusting pin component contains a Ø5 mm through-all hole.

What is the center of mass of the assembly with respect to the illustrated coordinate system?

a. X = −30.00, Y = −40.16, Z = −40.16

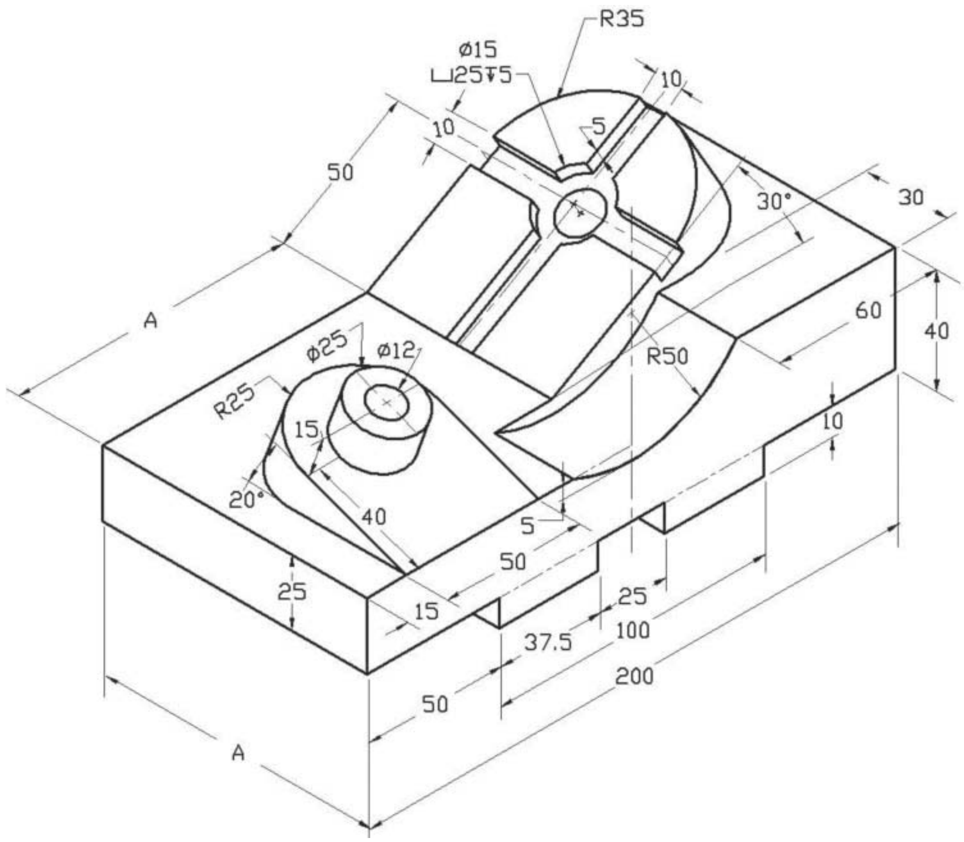
b. X = 30.00, Y = 40.16, Z = −43.82

c. X = −30.00, Y = −40.16, Z = 50.20

d. X = 30.00, Y = 40.16, Z = −53.82

Question 6

Build this part in SolidWorks.



Material: 6061 Alloy. Density = 0.0027g/mm^3

Unit system: MMGS (millimeter, gram, second)

Decimal places: 2.

Part origin: Arbitrary **A = 100**. All holes through

What is the overall mass of the part in grams?

a. 2040.57 **b.** 2004.57

c. 102.63 **d.** 1561.23

Correct Answers:

1. b **2.** c **3.** a

4. c **5.** d **6.** a

Certified SolidWorks Professional (CSWP)

Sample Exam Questions

The questions below represent sample CSWP Exam questions.

Question 1

A cut-extrude assembly feature is an example of a time-dependent feature.

a. True **b.** False

Question 2

You cannot reorder components within the FeatureManager design tree of an assembly.

a. True **b.** False

Question 3

When creating sketches, there are two types of inferencing lines to help you work more efficiently. Blue inference lines indicate that a reference is added automatically; brown inference lines indicate that no relation is added.

a. True **b.** False

Question 4

Which of the following is **not** a valid header for a column or row in an assembly design table?

a. $COMMENT

b. $PROPERTIES

c. $PARTNUMBER

d. $USER_NOTES

Question 5

It is necessary to have write access to a SolidWorks part file before you can mate to it in an assembly.

a. True **b.** False

Question 6

If a bill of materials is too long to fit on a drawing sheet, it can be split into sections.

a. True **b.** False

Question 7

What should you select when adding a <u>concentric</u> mate between two cylinders in an assembly?

a. The circular edges on the ends of the cylinders

b. Either the cylinder faces or circular edges

c. The temporary axes that run through the centers of the cylinders

d. Any of the above

Question 8

When editing a sketch, the status bar displays the status of the sketch. Choose the answer that describes three states of a sketch:

a. Under-defined, fully defined, conflicting
b. Self-intersecting, open, closed
c. Derived, defined in context, copied
d. Dangling, not solved, open
e. Under-defined, fully defined, over-defined

Question 9

The > symbol following a feature or a component in the FeatureManager design tree means:

a. The feature or component has an external reference
b. The feature or component is disjoint
c. The feature or component has a rebuilt error
d. The feature or component is under-defined

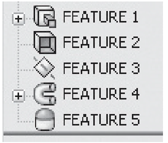
Question 10

To process a part as a sheet metal part, it must have uniform thickness.

a. True **b.** False

Question 11

The illustration to the right depicts a typical FeatureManager design tree. Look at the icons displayed alongside of FEATURE 1 through FEATURE 5. Identify the type of feature, and write the letter of its name (a–j) in the space provided below.



FEATURE 1:____________	**a.** Dome	**f.** Weld bead
FEATURE 2:____________	**b.** Tube	**g.** Loft
FEATURE 3:____________	**c.** Sweep	**h.** Rib
FEATURE 4:____________	**d.** Plane	**i.** Extrude
FEATURE 5:____________	**e.** Revolve	**j.** Shell

This page intentionally left blank

Index

Pages numbers followed by t indicate table; those followed by f indicate figure.

Abbreviation rules (ASME), 139, 139f
Actual size, 344
Addendum circle, 105f, 106f, 107
Additive manufacturing, 418
Additive modeling plan, 92
Advanced mates, 160, 160f
A half, 486
Air acidification, 331
Aliasing, 201
Allowance α, 352
Ambient light, 202
American Institute of Architects (AIA), 328
American National Standards Institute (ANSI), 138
  fits, 351t
  tolerance tables, 509t–522t
  weld symbols, 311t
American Society of Mechanical Engineers (ASME), 138
  abbreviation rules, 139, 139f
  dimensioning rules, 140–143, 142f, 143f
  drafting rules, 140, 140f
  tolerance rules, 346–349, 347f–349f
Analysis tools, 383–412
  animation and motion analysis, 390–391
  data exchange, 383–386, 384f, 385t
  finite element analysis, 394–395
  finite element method, 391–394, 393f
  Flow Simulation, 391
  mass properties, 386–390, 388f, 388t
  tutorials, 400–412
  Von Mises stress, 396–400, 396f, 396t, 398f
Analytic curves, 223
Analytic surfaces, 255
Angle tolerances, 423, 423f
Angular dimensions, 140
Angular patterns, 58
Animation
  as analysis tool, 390–391
  assembly exploded view and, 166–167
  camera-based, 207
  camera-sled based, 215–218
  playback, 207
  real-time, 207
Animation controller, 22
Annotations
  annotation tab, 18
  drawing content/layout, 146, 146f
  inserting, 153–154
ANSI. *See* American National Standards Institute (ANSI)
Anti-aliasing effect, 201
Arithmetic tolerancing, 357, 357f
ASME. *See* American Society of Mechanical Engineers (ASME)
Assembly/assemblies, 159–185
  assembly tree, 165–166
  bottom-up modeling, 160–161
  defined, 159
  design tables, 168
  drawings, 155–156, 166
  exploded view and animation, 166–167
  interference and collision detection, 167–168
  mates, 159, 160, 160f
  mating conditions, 86
  motion study, 167, 167f
  top-down modeling, 161–164, 161f
  tutorials, 168–185
Assembly model creation, 20–22, 21f, 22f
Assembly prototype file generation, 427–428
Autodimension, 18
Automation, macros and, 112–115, 114f
Automobile design, 417
Auxiliary view, 148, 148f
Axis, 434

Ball screw assembly, 175–176, 175f
Baseball hat, creating, 279–282, 279f
Base circle, 105f, 106f, 107
Base flange, 297, 298t
Baseline dimensions, 141, 355f, 356
Base plate modeling plan, 17–18, 17f–18f
Base support structure, 422
Basic size, 344
Beam bending equation, 394
Bend (sketched bend), 297, 299t
Bicycle handle bar, 234–235
Bilateral tolerance, 346, 348
Bill of materials, 151–152
  creating assembly drawing with, 155–156
"Bill of Rights for the Planet," 328
Block mold creation, 493–496, 493f
Block statement, 450
Bolts, creating, 124–126, 124f
Boston Gear, 171, 173
Bottle prototype, 425–426, 425f
Bottom-up assembly modeling, 160–161
Boundary, 257t, 268
Boundary representation (B-rep), 386
Bowl, creating, 270–273, 270f
Bracket tutorial, 70–71, 70f
B-rep, 386
Broken-out section, 148, 148f
Broken view, 148f, 149
Brundtland Report, 326
B-spline, 223, 227
Bubbles (defects), 487
Build orientation, 421, 421f
Burn marks, 487
Burr, 439–440

CAD models
  communication, 66–67, 66f
  types of, 41–42, 42f
CAD process, 4, 5f, 86
CAM Add-Ins software, 452, 452f
Cam and follower assembly, 168–169, 168f
Camera-based animation, 207
Cameras, 207
Camera sleds, 207
CAM process, 6, 6f, 42t, 43
CAMWorks, 448
  generating toolpaths in, 452
  tutorials using, 453, 456, 461,466, 472, 478
Canned cycle, 449
Carbon footprint, 327, 330
Car design, 417
Cartesian dimensions, 140, 141
Cartesian patterns, 58
Caster assembly, 74–76, 74f
Caustic effects, 203
Center modifier, 57
Certified SolidWorks Associate (CSWA), 3, 523–524
  sample exam questions and answers, 527–531

Certified SolidWorks Professional (CSWP), 3, 524
FEA, 526
mold tools, 526
sample exam questions, 532–533
sheet metal, 525
Surfacing, 524–525
weldments, 525
Chain dimensioning, 355f, 356
Chamfer, 308, 310t
Chamfer parameters, 122
Chip load, 438
Chord distance, 423
Circle parametric equation, 226–227, 226f
Circular patterns, 58, 59f
Circular pitch, 106f, 107
Clearance fits, 351–352, 351t, 353f
ANSI and ISO tolerance tables, 509t–515t
Coil spring tutorial, 67–69, 67f
Coincident mating, 22
Collision detection, 167–168, 180–181, 180f
Color application, 208
Combs, 265
Composite, 41, 42f
Compression spring tutorial, 120, 120f
Computer aided design (CAD) process, 6, 6f, 42t, 43
capturing design intent, 86
Computer aided manufacturing (CAM) process, 6
part creation planning, 42t, 43
Computer mouse, creating, 276–278, 276f
Concentric mating, 22
Concept cars, 417
Concurrent engineering, 445
Configuration Manager, 22
Configurations, 111
Conical taper, 349–351, 350f
Conics, 223
Conjugate action, 106, 106f
Construction geometry, 52
Convert Entities method, 118
Coordinate systems, 47, 48f
Corner feature, 297, 300t
Crop view, 148, 148f
Cross-section modeling approach, 13–14, 13f, 43, 92–93
CSWA. *See* Certified SolidWorks Associate (CSWA)
CSWP. *See* Certified SolidWorks Professional (CSWP)
Cubic curve, 223
Curvature, of surfaces, 265
Curve combs, 265, 270
Curve-driven patterns, 58
Curves, 223–249
circle parametric equation, 226–227, 226f
creation of (tutorials), 230–249
curve management, 230
families of, 224f
line parametric equation, 225–226, 225f
representation, 224–225, 224f
spline parametric equation, 227–228, 227f
three-dimensional, 229, 229f
two-dimensional, 228–229
Cutter location (CL) data, 449
Cutting feed, 438
Cutting speed, 437, 438
Cutting tools, 435, 435f

Datum, 346
Datum targets, 346
symbols, 361, 361f
Deburring, 440
Decals, 203–204, 204f
"Declaration of Interdependence for a Sustainable Future," 328
Dedendum circle, 105f, 107
Derived part, 110
Design, and society, 326–327
Design Binder, 88
Design checkers, 152
Design for anything (DFX), 445
Design for assembly (DFA), 327, 445
Design for manufacturing (DFM), 327, 445
Design for sustainability (DFS), 327
Design intent, 85–97
capturing, 86
comments, 87
defined, 85
design binder, 88
Design Binder, 88
dimension names, 89
documenting, 87
equations, 88
feature names, 89, 89f
folders, 90
tables and configurations, 89
tutorials, 90–97
via design specifications, 94–95, 94f
via mating conditions, 96–97, 96f
Design intent system, 85
Design library, 110–111, 111f
Design prototyping. *See* Rapid prototyping
Design tables, 111–112, 111f
assembly, 168
creation of, 181–183, 181f
Desktop manufacturing. *See* Rapid prototyping
Detail view, 148, 148f
Deviation tolerances, 423, 423f
Diffuse surface, 203
Dimensioning and Tolerancing Handbook (Wilson), 138
Dimensioning rules (ASME), 140–143, 142f, 143f
Dimensions
defined, 44–45, 45f
dimensioning rules, 140–144, 142f, 143f
DimXpert, 18, 365–366
Direct dimensioning, 356, 356f
Directional light, 202
Disposal constraints, 325
Draft, 490
Drafting rules (ASME), 140, 140f
Drawings, 137–157
angle of projection, 147, 147f
ASME abbreviation rules, 139, 139f
bill of materials, 151–152
content and layout, 146, 146f
design checkers, 152
dimensions, 137–138, 143–146, 143f–145f
drafting control, 150–151
drafting rules, 140, 140f
engineering drafting and graphics communication, 138
model and drawing associativity, 152
mold drawings, 505–506, 506f
sheets, 149, 150f
title block, 150
tolerances, 151
tutorials, 29–31, 30f, 31f, 153–157
views, 147–149, 148f
Drawing template, 149
Drill holes, 455–460, 455f
Drilling, 431, 442, 446
Ductile materials, 396–397
Durability, of product, 328
DXF, 384
Dynamic interference, 168

Easter egg mold, 502–505, 503f
Ecodesign. *See* Sustainable design
Edge, 44
Edge flange, 297, 298t
eDrawings, 7, 67
Ejector marks, 487
Electrical discharge machining, 431, 442–445, 443f, 477–479, 477f
End cap, 308, 309t
End modifier, 57
End of life (EOL), 330
Energy constraints, 325
Energy consumption, 327, 331
Engineering design process, 4, 4f
Engineering drawing examples, 18–20, 19f
Environmental constraints, 325
Environmentally conscious design. *See* Sustainable design
Environmentally sustainable design. *See* Sustainable design
Environmental sustainability, 326
Equations, 55, 86. *See also* specific equation
Euler-Bernoulli beam equation, 394
Explicit equation, 224, 230
Exploded view, 22, 166–167
Extension line, 141
Extruded Boss/Base, 30, 101
Extruded cut, 297, 308
Extrusion, 15, 41, 42f, 256t, 266, 266f
creating with a macro, 113–114

Fabrication, rapid prototyping, 419
Face, 44
Face milling, 442
Facet, 423
Feature-based modeling, 102–105, 102t–103t
Feature-based patterns, 58, 60
Feature Manager Design Tree, 51
Feature recognition, 386
Features
available, 102t–103t
defined, 15, 101
transforming, 62–63
tutorial on creation of, 122–123
Features Manager Design Tree, 165
Features modeling plan, 13–14, 14f
Features tree, 101
Feedrate, 438
FEM/FEA, 391–394, 393f
Filled surface, 257t, 268
Fillet bead, 308
Fill title block tutorial, 154–155
Finite element analysis (FEA), 391, 394–395, 407–408, 407f
static linear, 407–408, 407f
thermal, 409, 409f
Flap creation tutorial, 23–25, 23f
Flashing, 487
Flat taper, 349–351, 350f
Flatten, 297, 301t
Flow analysis, 383, 391, 410–412, 410f
Flow marks, 487
FloXpress, 410
Flute, 435, 436t
Fog light, 202
Fold, 297, 301t
Free form model, 41, 42f
Free form parts, 255
Front plane, 49, 50f
Fully defined sketch, 50, 51f
Fundamental deviation (ANSI and ISO tolerance)
metric (SI) units, 522t
U.S. (English) units, 521t
Fused deposition modeling (FDM), 424, 425

Gate and runner system, 489, 489f
Gauss quadrature, 387–388, 388f, 388t
G-Code programming, 450, 451t
Gears. *See also* Spur gears
  mate two gears with gear mate, 171–172, 171f
Gear tooth, 105–106, 106f
Genus, 44
Geometric arrays. *See* Patterns
Geometric dimensioning and tolerancing (GD&T), 344
Geometric modeling. *See* Curves
Geometric modifiers, 57
Geometric relation symbols, 10f
Geometric tolerances, 140, 344, 358–361, 359t, 360f, 369–370, 369f
  assigning, 359
  interpreting, 360, 362, 362t–363t
  symbols for, 359t
  true position, 358, 358f
Go-NoGo gauge, 344, 359
Graphics communication, 138
Green design. *See* Sustainable design
Grids, 58, 58f
Gusset, 308, 310t

Hair dryer, creating, 282–285, 282f
"Hanover Principles," 328
Healthy buildings, 328
Help menu, 7
Hem, 297, 300t
Hemisphere mold, 500–502, 500f
Hidden line removal, 201
Hole and shaft systems, 345
Hole Wizard, 118–119, 119f, 308
Home position, 439
Hook's law, 396
Hotkeys, 47, 114–115
Housing (mold housing), 488

IGES, 384, 385t, 400, 401, 428
Inclusions, 306
In-context approach, 161
In-context features, 162
Injection molding, 483–506
  advantages and disadvantages of, 483–484
  basics, 485–487, 485f, 486f
  block mold creation, 493–496, 493f
  cycle, 485–486, 485f
  defects, 486–487
  defined, 483
  drawings, 505–506, 506f
  hemisphere mold, 500–502, 500f
  machines, 484, 484f, 485f
  mold design, 487–490, 489f, 490–491
  part design, 490
  sandbox mold creation, 496f
  SolidWorks Mold Tools, 491–492, 492f
  tutorials, 492–506
In-out cutting, 440
Instance, 159
Interference detection, 167–168, 180–181, 180f
Interference fits, 351t, 352, 353f
  ANSI and ISO tolerance tables, 518t–520t
International Organization for Standardization (ISO), 138
  fits, 351t
  sustainability standards, 331–332
  tolerance tables, 509t–522t
  weld symbols, 311t
Intersection modifier, 57
Involute profile, 105–106, 106f
ISO. *See* International Organization for Standardization (ISO)
Isotropic materials, 396

Jog, 297, 299t

K-Factor, 296
Knit surface, 258t, 266f, 267

Laminated object manufacturing (LOM), 424
Layered manufacturing. *See* Rapid prototyping
Layering (slicing), 421, 421f
Layout sketch, 161–162
Leader line, 141
Library feature, 110–111, 110f
Life cycle assessment, 327, 329–330, 329f
Limit dimension, 34, 345
Linear patterns, 58, 59f
Line parametric equation, 225–226, 225f
Link values, 55–57, 56f
Loft, 257t, 266f, 267
Lofted bend, 297, 300t
Lofted Boss/Base, 101
Loft features tutorial, 117–118, 117f
Loop, 44

Machine tools, 432–434, 432f, 433f
Machine zero, 434
Machining. *See also* Numerical control machining
  basics, 434–441, 436f–438f, 440f, 441f
  electrical discharge, 442–445, 443f
  parameters, 437
Machining parameters, 437–439, 437f
Machining process, 431
Machining quality, 439
Macros, 112–115, 114f
Manufacturing process, 5–6, 5f
  categorization of, 431
  efficiency of, 327
  life cycle assessment and, 329
  variability, 343
Mass customization, 431
Mass production, 431
Mass properties, 386–390, 388f, 388t, 402
Mastercam, 488
  generating toolpaths in, 452
  tutorials using, 458, 463, 468
Master parts, 65
Material condition, 346
  symbol (modifier), 359–360
Materials, 205–206, 205f
  ductile, 496–497
  isotropic, 396
  library, 205, 205f
  material processing, 329
  rendering, 205–206, 205f, 210–211
  selection, environmental impact of, 327, 330
  stock, 437, 437f, 439
  thermoplastic, 483
Mates, 22, 159, 160, 160f
M-Code programming, 450, 451t
Mechanical mates, 160, 160f
Mesh, 265
Mill faces, 461–465, 461f
Milling, 431, 433f, 442, 446
Mill pockets, 466–471, 466f
Mill slots, 471–476, 471f
Miter flange, 297, 298t
Model coordinate system (MCS), 47
Model-drawing associativity, 156–157
Modeling management, 41–76
  Boolean operations, 63–65, 64f
  CAD model types, 41–42, 42f
  construction geometry, 52
  coordinate systems, 47, 48f
  customizing SolidWorks, 46–47, 46f
  equations and link values, 55–57, 56f
  geometric modifiers, 57
  grids, 58, 58f
  model communication, 66–67, 66f
  parametric modeling, 44–46, 45f
  part creation planning, 42t, 43
  part features tree, 51–52, 52f
  part topology, 44, 44f
  patterns, 58–61, 59f, 60f
  productivity tools, 47
  reference geometry, 52, 53f
  selecting, editing, measuring entities, 62–63
  sketch entities, 54, 54f
  sketch planes, 48–50, 49f
  sketch relations, 55
  sketch status, 50–51, 51f
  templates, 65
  tutorials, 67–76
  viewing, 65
Modeling plan, 13–14, 13f–14f, 86
Model Items, 18, 41
Mold base, 489
Mold cavity, 486
Mold design, 490–491
Motion analysis, 390–391, 402–407, 403f
Motion axes, 434–435
Motion Manager, 178–179
Motion study
  assembly, 167, 167f
  creating, 212–215
  tutorial, 178–180, 178f
Motor torque, 404, 403f
Mount plate tutorial, 69–70, 69f

Named (orthograhic) views, 147
Napkin sketch, 161
Nominal size, 344
Nonplanar curves, 228
Nonuniform scaling, 63
Numerical control machining, 431–479
  basics, 448–449
  CAM Add-Ins, 452, 452f
  design manufacturing, 445
  drilling, 442
  electrical discharge machining, 442–445, 443f
  G-Code and M-Code programming, 450, 451t
  machine tool basics, 432–434, 432f, 433f
  machining basics, 434–441, 436f–438f, 440f, 441f
  milling, 442
  NC program, 449, 450
  SolidWorks DFMXpress, 445–448, 447f, 448f
  turning, 441
  tutorials, 452–464

Offset surface, 268
Oil container, creating, 285–287, 286f
On Edge relation, 117
Optimization, rapid prototyping and, 419
Ordinate dimensions, 141
Orthographic views, 147, 148f
Our Common Future, 326
Out-in cutting, 440
Over defined sketch, 50, 51f

Parameter, 44, 45f
Parametric circle, 226f
Parametric equations, 224, 259–260
Parametric modeling, 44–46, 45f
Parametric spline, 227–228, 227f
Part creation, 14–15
  in context of assembly, 183–185, 183f
  planning, 42t, 43
Part design, 490
Part features tree, 51–52, 52f
Parting axis, 488

Parting line, 488
Parting surface, 488
Part prototype file generation, 426–427, 427f
Part topology, 44, 44f
Patterns, 58–61, 59f, 60f
PE International, 331
Perimeter cutting, 440, 440f
Photo realism, 201
PhotoView 360, 201
Picture frame, 235–236
Pillow block tutorial, 26–29, 26f–27f, 31f
Pin and bushing bearing tutorial, 25–26, 25f
Pin block assembly, 96, 168
Pin modeling plan, 16–17, 16f–17f
Pitch circle, 106f, 107
Planar curves, 228
Planar surface, 258t, 268–269
Plane parametric equation, 260–262, 261f
Plate modeling plan, 15–16, 15f–16f
Platens, 486
Playback animation, 207
Pocket milling, 442
Point light, 202
Point-to-point (PTP) machining, 440
Porosity, 306
Powder injection molding, 483
Pressure angle, 106
Process planner, 6, 431
Product data management (PDM), 445
Productivity tools, 47
Product life cycle, 327, 329–330, 329f
Product life cycle management (PLM), 445
Programmable mouse, 47
Projected view, 147
Prototyping, 201, 417. *See also* Rapid prototyping

Quality, of product, 328
Quality assurance, 344
Quality control, 344
Quick snaps, 57

Rack and pinion assembly, 172–174, 173f
Radial dimensions, 140
Radiate surface, 266f, 267
Rapid positioning, 441
Rapid prototyping, 417–429
  applications, 418–419, 419f
  bottle prototype, 425–426, 425f
  building techniques, 424–425
  concepts, 420–422, 420f–422f
  defined, 419
  overview, 419–420
  steps, 424
  triangulation (tessellation), 420–423, 420f
  tutorials, 426–429
Raw material extraction, 329
Ray tracing, 203
Real-time animation, 207
Rebuilt components, 325
Rectangular patterns, 58
Recycling, 325, 327
Reference geometry, 52, 53f, 308
Refurbished components, 325
Relative-to-model view, 148f, 149
Rendering, 201–207
  appearance, 206
  background and scenes, 206–209
  cameras and camera sled, 207, 211–212, 215–218
  decals, 203–204, 204f
  materials, 205–206, 205f, 210–211
  models, 202–203, 203f
  motion study creation, 212–215
  scenes and lighting, 201–202, 202f, 209–210
  textures, 205
  transparency, 206, 210–211
  tutorials, 207–218
Resins, 483
Resources, sharing, 328
Reuse of products, processes, systems, 327
Revolve, 15, 41, 42f, 256t, 266, 266f
Revolved Boss/Base, 101
Right plane, 49
Rip, 297, 300t
Root circle, 105f, 107
Ruled surface parametric equation, 262–265, 262f–264f

Sandbox mold, 496–499, 496f
Section view, 147
Segregation, 306
Selective laser sintering (SLS), 424
SFM (surface feet per minute), 438–439
Shaded views, 201
Shaft and hole, 344
Sheet metal, 296–305
  creating parts from flattened state, 316–317, 316f
  creating parts from solid body, 303, 315–316, 315f
  defined, 296
  drawings, 313–315, 314f
  FeatureManager design tree, 302–303, 302f
  features, 297, 297f, 298t–301t
  K-Factor, 296
  methods, 303–305, 303f
  modeling and operations, 312–313, 312f
  rules, 446
  tutorials, 313–317
Sheets, 149, 150f
Short shot (partial filling), 487
Side milling, 442
Simple hole, 297
Simple tensile test, 397
SimulationXpress, 333, 395–396, 398
Sink marks, 487
Size, 344
Skeleton sketch, 161
Sketched bend, 297, 299t
Sketch entities, 54, 54f
  editing, 62
  equations, 55, 56f
  geometric modifiers, 57
  measuring, 63
  patterns, 58–61, 59f, 60f
  relations, 55
  selecting, 62
Sketch planes, 15, 48–50, 49f
Sketch relations, 55, 86
Slot milling, 442
Smart Dimension, 18
Smart Fasteners Wizard tutorial, 123–124, 123f
Society, and design, 326–327
Solid free-form fabrication. *See* Rapid prototyping
Solid ground curing (SGC), 424
Solid model, 44
SolidWorks
  Compare Geometry, 401
  core use of, 1
  customizing, 12, 46–47, 46f
  DFMXpress, 445–448, 447f, 448f
  exams, 3
  exporting native files, 400
  file formats, 384f, 385t
  Flow Simulation, 391
  FloXpress, 410
  importing IGES and STEP files into, 401
  installation and resources, 6–8
  Mass Properties, 390
  materials library, 205, 205f
  Mold Tools, 491–506, 492f
  overview, 8–12, 8f, 10f–12f
  PDMWorks, 445
  PhotoView 360, 201
  repetitive tasks of, 9f–10f
  SimulationXpress and Simulation, 383, 395–396, 398–399
  SustainabilityXpress, 333–336, 336f
  tolerance analysis, 365–368, 366f
  triangulation, 422–423, 423f
SolidWorks certification, 523–533
  Certified SolidWorks Associate (CSWA), 523–524, 527–531
  Certified SolidWorks Professional (CSWP), 524–526
SolidWorks Viewer, 7
Spark frequency, 444
Specular surface, 203
Spindle (rotation) speed, 437
Spiral spring tutorial, 121, 121f
Spline curve, 54, 223
Spline parametric equation, 227–228, 227f
Spot light, 202
Spur gears, 105–110, 105f, 106f, 108f
Staircase effect, 201
Standard mates, 160, 160f
Start parts, 65
Static interference, 167
Statistical tolerancing, 356–358, 357f
Steel washer redesign, 336–338, 336f
STEP , 384, 385t, 400, 401, 428, 491
Stepped shaft tutorial, 453–455, 453f
Stereolithography (SLA), 424–425
Stereolithography (STL) file format, 422–423, 423f, 428–429
Stethoscope model, 247–249, 247f
STL file format, 422–423, 423f, 428–429
Stock material, 437, 437f, 439
Structural member, 309t
Subtractive manufacturing, 93, 418
Support structure, 421–422
Surface feet per minute (SFM), 438–439
Surface finish, 205
Surface patches, 259
Surfaces, 255–287
  available, 256t–258t
  creating basic, 266–268, 266f
  defined, 255
  intersections of, using, 273–274, 273f
  management of, 266
  plane parametric equation, 260–262, 261f
  representation, 259–260, 259f
  ruled surface parametric equation, 262–265, 262f–264f
  in solid modeling, 258–259, 258f, 259f
  tutorials, 266–287
  visualization of, 265, 269–270, 269f
Surface tolerance, 423
SustainabilityXpress, 333–336, 334f, 336f
Sustainable design, 325–338
  activities, 332–333
  design and society, 326–327
  guidelines and principles, 327–328
  impact metric, 330–331
  implementation, 331–332
  life cycle assessment, 329–330, 329f
  SustainabilityXpress, 333–336, 334f, 336f
  tools, 333
  tutorials, 336–338
Sustainable engineering, 326
Sweep, 256t, 266f, 267
Sweep feature tutorial, 115–116, 115f, 116f
Swept Boss/Base, 101
Symmetric tolerance, 346
Synthetic curve, 223
Synthetic surfaces, 255

Tablespoon, creating, 275–276, 275f
Tab subflange, 297, 299t
Tangent vector, 224
Tapping tool, 435
Technical Drawing (Giesecke et al.), 138
Templates, 65
Testing, rapid prototyping and, 418
Textures, 205
Thead type, 435
Thermoplastic materials, 483
3D curves, 229, 229f, 231–247
3D modeling concepts, 3
  curves, 221
  surfaces, 221
3D printing, 424, 425. *See also* Rapid prototyping
3D sketch, 309t
3D views, 201
Tire and pin tutorial, 73–74, 73f
Title block, 150
TolAnalyst, 365–366, 373–374, 373f
Tolerance allocations, 364
Tolerance grades (ANSI and ISO)
  metric (SI) units, 522t
  U.S. (English) units, 521t
Tolerances, 151, 343–374
  analysis, 363–365, 373–374
  angle, 423, 423f
  ANSI and IOS tolerance tables, 509t–522t
  ASME rules, 346–349, 347f–349f
  bilateral, 346, 348
  concepts, 344–346
  conventional, 344, 368–369, 368f
  datum targets, 361, 361f, 371, 371f
  defined, 343
  deviation, 423, 423f
  in drafting, 140
  geometric, 140, 344, 358–361, 359t, 360f, 369–370, 369f
  inch (English), 348, 349f
  interpretation, 362, 362t–363t
  limits of dimensions, 351–354, 351t, 353f, 354f, 354t
  SolidWorks analysis, 365–368, 366f
  statistical, 356–358, 357f
  symmetric, 346
  tapers, 349–351, 350f, 372, 372f
  tolerance accumulation, 355–356, 355f–356f
  true position, 358, 358f
  tutorials, 368–374
  types of, 344
  unilateral, 345, 348
Tolerance stack-up analysis, 363
Tolerance synthesis, 365
Tolerance zone, 345
Tool chuck, 437–438
Tool offset, 449
Toolpath, 439–441, 440f, 452
Toolpath verification, 449
Tool splitting, 488
Top-down assembly modeling, 161–164, 161f
Top plane, 49, 50f
Transition fits, 351t, 352, 353f
  ANSI and ISO tolerance tables, 516t–517t
Triangulation (tessellation), 420–421, 420f
  SolidWorks, 422–423, 423f
Trim/extend features, 308, 309t
True length dimensions, 140
True position, 358, 358f
Turning, 431, 441, 446
Turret, 442
2D curves, 229, 229f, 230, 230f, 231, 231f

Undercut, 490
Under defined sketch, 50, 51f
Unfold, 297, 301t
Uniaxial stress test, 397
Uniform scaling, 63
Unilateral tolerance, 345, 348
Universal joint motion study, 176–178, 176f

Variability, in manufacturing, 343
Verification, rapid prototyping and, 418
Vertex, 44
Visual Basic, 113
Visualization. *See also* Animation; Rendering
  rapid prototyping and, 418
  of surfaces, 265, 269–270, 269f
Voids, 487
Von Mises stress, 396–400, 396f, 396t, 398f

Warpage, 487
Water eutrophication, 331
Water footprint, 331
Weld bead, 310t
Welding, 305
Weld lines, 487
Weldments, 305–311, 306f
  creating, 317–319, 317f
  defined, 307
  drawings, 319–321, 320f
  features, 307–308, 309t–310t
  tutorials, 317–321
  types of weld joints, 306–307, 306f
  weld joints, types of, 306f
Weld symbols, 310, 311t
Wheel tutorial, 71–73, 71f
Wire EDM machine, 443f, 444
Wireframe, 201
Working coordinate system (WCS), 37
Working hinge assembly, 170–171, 170f
Workpiece, 437
Workpiece zero, 434
World Commission on Environment and Development (WCED), 326
World Congress of the International Union of Architects (UIA), 328
World Engineering Partnership for Sustainable Development (WEPSD), 326

Zebra stripes, 265
Zero-radius programming, 449
Zigzag cutting, 440, 440f