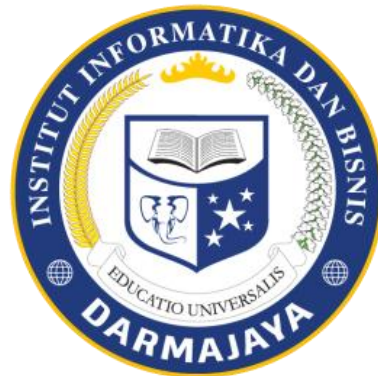


**IMPLEMENTASI FUZZY LOGIC CONTROL PADA SISTEM
PERINGATAN DINI KEBAKARAN BERBASIS IOT**

SKRIPSI

**Diajukan Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA**

**Pada Program Studi Sistem Komputer
IIB Darmajaya Bandar Lampung**



Oleh

M.Arif Dharmawan

1411060022

**FAKULTAS ILMU KOMPUTER
PROGRAM STUDI SISTEM KOMPUTER
INSTITUT INFORMATIKA DAN BISNIS DARMAJAYA
BANDAR LAMPUNG
2019**

PERNYATAAN ORISINILITAS PENELITIAN



PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa Skripsi yang diajukan ini adalah hasil karya saya sendiri, tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi atau karya yang pernah ditulis atau diterbitkan orang lain kecuali yang secara tertulis diacu dalam naskah ini disebutkan dalam daftar pustaka. Karya ini adalah milik saya dan pertanggung jawaban sepenuhnya berada dipundak saya.

Bandar Lampung, 9 September 2019

METERAI
TEMPEL
TGL. 20
358B3AHF6512/237
6000
ENAM RIBU RUPIAH

M. Ari Duai mawan
1411060022

PERSETUJUAN

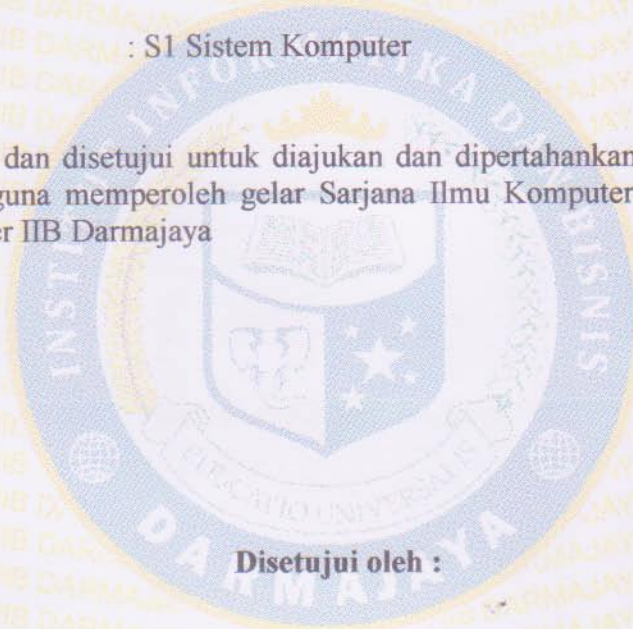
Judul Skripsi : **IMPLEMENTASI FUZZY LOGIC CONTROL
PADA SISTEM PERINGATAN DINI KEBAKARAN
BERBASIS IOT**

Nama Mahasiswa : **M. Arif Dharmawan**

No. Pokok Mahasiswa : 1411060022

Program Studi : S1 Sistem Komputer


Telah diperiksa dan disetujui untuk diajukan dan dipertahankan dalam sidang Tugas Penutup Studi guna memperoleh gelar Sarjana Ilmu Komputer pada Program Studi Sistem Komputer IIB Darmajaya

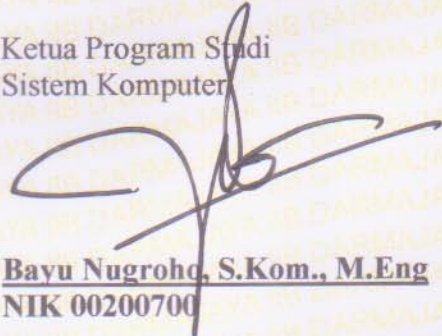


Disetujui oleh :

Dosen Pembimbing

Ketua Program Studi
Sistem Komputer


Dodi Yudho Setyawan, S.Si., M.T.I
NIK 11340809


Bayu Nugroho, S.Kom., M.Eng
NIK 00200700

PENGESAHAN

Telah diuji dan dipertahankan di depan Tim Penguji Skripsi
Program Studi Sistem Komputer Institut Informatika dan Bisnis Darmajaya
Bandar Lampung dan dinyatakan diterima untuk memenuhi syarat guna
memperoleh gelar Sarjana

Mengesahkan

1. Tim Penguji

Ketua : **Nurfiana, S.Kom., M.Kom**

Anggota : **Ari Widiyantoko, S.Kom., M.Tech**

2. Dekan Fakultas Ilmu Komputer



Zaidir Jamal, S.T., M.Eng

NIK 00590203

Tanggal Lulus Ujian Skripsi : **27 September 2019**

HALAMAN PERSEMBAHAN

Bismillahirrahmanirrahiim

Assalamu 'alaikum warahmatullahi wabarakatuh

Seiring Syukur Atas Ridho Allah SWT Saya sebagai penulis dapat menyelesaikan Skripsi yang saya persembahkan kepada :

1. Ayahanda tercinta Hermawan yang telah memberikan saya semangat tanpa henti dan membawa saya sampai ke jenjang perkuliahan.
2. Ibunda tercinta Warsini yang selalu memberikan saya masukan untuk menjalankan perkuliahan tanpa menyerah serta selalu menyemangati saya untuk tidak mengeluh.
3. Adek Ninis Leadystha yang selalu memberikan doa dan dukungannya dan Alm. Nidha Ameylia (wafat 5 agustus 2019) yang selalu memberikan doa dan dukungannya sampai akhir hayatnya serta Nindya Agustin yg selalu mendoakan dan memberikan semangat.
4. Seluruh keluarga besarku yang selama ini mendukungku selamaku menuntut ilmu diperguruan tinggi IIB Darmajaya.
5. Untuk Teman teman terdekatku yang telah memberikan saya semangat dan dukungannya dalam mengerjakan skripsi..
6. Seluruh dosen-dosen IIB Darmajaya terimakasih semua, khususnya dosen-dosen Program Studi Sistem Komputer dan Teknik Komputer.
7. Almamaterku tercinta IIB Darmajaya.

Wassalamu 'alaikum warahmatullahi wabarakatuh

MOTTO

“Ubah cara berpikirmu dan dekatkan diri pada tuhanmu maka kau dapat mengubah hidupmu”

(M.ARIF DHARMAWAN)

ABSTRAK

IMPLEMENTASI FUZZY LOGIC CONTROL PADA SISTEM PERINGATAN DINI KEBAKARAN BERBASIS IOT

Oleh

M.ARIF DHARMAWAN

Sistem peringatan kebakaran merupakan sistem yang dirancang sebagai alat peringatan jika terjadinya kebakaran pada suatu gedung perkantoran, rumah, atau sektor perindustrian. Contohnya alarm kebakaran yang terdapat pada gedung pusat perbelanjaan ataupun gedung pusat industri. Pada saat terjadi kebakaran kebanyakan alarm yang digunakan masih menggunakan sistem manual untuk memadamkan api, contohnya pada Zet Water sehingga petugas harus membuka kran air terlebih dahulu sehingga proses pemadaman membutuhkan waktu yang lama dan berdampak kepada kerugian material dan nyawa seseorang. Maka dari itu tujuan peneliti yaitu membuat sistem pemadam kebakaran secara otomatis menggunakan metode Fuzzy, Arduino, serta Node MCU sebagai pengendali sekaligus sebagai alat untuk mengirimkan pesan jika terjadinya kebakaran melalui aplikasi Telegram sehingga jika terjadinya kebakaran dapat cepat terdeteksi besarnya api melalui sensor MQ2 dan Flame serta alat akan mengirimkan pesan melalui aplikasi Telegram yang menampilkan kondisi kebakaran menggunakan perhitungan fuzzy mamdani dan dapat langsung dipadamkan menggunakan alat ini secara otomatis. Hasil dari penelitian sistem peringatan dini kebakaran ini dapat meminimalisir terjadinya kebakaran yang dapat meluas serta cepat dalam memberikan informasi kondisi kebakaran yang terjadi melalui aplikasi Telegram.

Kata Kunci : fuzzy, Arduino ,MCU

ABSTRACT

IMPLEMENTATION OF FUZZY LOGIC CONTROL ON IOT-BASED FIRE EARLY WARNING SYSTEM

By:

M.ARIF DHARMAWAN

A fire warning system is a system designed as a warning device in the event of a fire in an office building, house, or industrial sector such as fire alarms in shopping center buildings or industrial center buildings. When a fire occurs, most of the alarms used still use a manual system to extinguish the fire, for example in Zet Water, in which the officers have to open the water tap first so that the extinguishing process takes a long time and has an impact on material loss and someone's life. Therefore, the purpose of the research was to create an automatic fire extinguishing system using the Fuzzy, Arduino, and MCU Node as a controller as well as a tool to send messages when a fire occurred via the Telegram application. Therefore, if a fire occurs, the size of the fire can be quickly detected through the MQ2 and Flame sensors and the tool will send a message via the Telegram application that displays fire conditions using fuzzy mamdani calculations and can be immediately extinguished using this tool automatically. The results of this research showed that the fire early warning systems was able to minimize the spread of fire and to provide information on fire conditions instantly through the Telegram application.

Keywords: fuzzy, Arduino, MCU

KATA PENGANTAR

Assalamu'alaikum Wr.Wb

Puji syukur saya ucapkan kehadirat Allah SWT yang telah melimpahkan segenap rahmat dan hidayah-nya sehingga saya dapat menyelesaikan laporan skripsi yang berjudul “Implementasi Fuzzy Logic Control Pada Sistem Peringatan Dini Kebakaran Berbasis Iot” Skripsi ini disusun sebagai persyaratan untuk memperoleh Gelar Sarjana Komputer (S.Kom) Sistem Komputer, IIB Darmajaya.

Saya mengucapkan terima kasih kepada pihak – pihak yang telah memberikan bantuan dan dukungan selama pengerjaan Skripsi ini. Ucapan terima kasih khusus saya sampaikan kepada :

1. Bapak Dr. Hi.,Andi Desfiandi, Se, Ma. Selaku ketua yayasan Institut Informatika dan Bisnis Darmajaya.
2. Bapak Ir. Hi.,Firmansyah Y.Alfian Mba., M.Sc Selaku Rektor Institut Informatika dan Bisnis Darmajaya.
3. Bapak Zaidir Jamal S.T.,M.Eng Selaku Dekan Ilmu Komputer Institut Informatika dan Bisnis Darmajaya.
4. Bapak Bayu Nugroho, S.Kom., M.Eng Selaku Ketua Program Studi Sistem Komputer dan Teknik Komputer, terimakasih atas waktu dan saran yang telah bapak berikan kepada saya.
5. Bapak Novi Herawadi Sudiby, S.Kom., M.T.I selaku Sekertaris Program Studi Teknik Komputer dan Sistem Komputer,terimakasih atas waktu dan saran yang telah bapak berikan kepada saya.
6. Bapak Dodi Yudho Setyawan, S.Si.,M.T.I selaku dosen pengajar sekaligus sebagai pembimbing saya dalam menyelesaikan laporan Skripsi ini. terimakasih atas waktu dan saran yang telah bapak berikan kepada saya.
7. Ayah dan Ibu tercinta yang selalu memberikan dukungan, semangat dan doa kepada saya.

8. Seluruh teman – teman Teknik Komputer dan Sistem Komputer Angkatan 2014, semoga kebersamaan kita selama ini terus terjalin.

Dengan segala keterbatasan saya menyadari bahwa masih banyak kekurangan dalam penyusunan laporan skripsi ini. Untuk itu saran dan kritik yang *konstruktif* dan *solutif* dari semua pihak sangat saya harapkan demi perbaikan dan peningkatan skripsi ini.

Akhirnya, saya hanya bisa mendoakan semoga Allah SWT. Membalas semua kebaikan – kebaikan mereka selama ini. Amin.

Wassalamualaikum Wr.Wb

Bandar Lampung, 9 September 2019

M.Arif Dharmawan
1411060022

DAFTAR ISI

PERNYATAAN ORISINILITAS PENELITIAN.....	ii
PERSETUJUAN	iii
PENGESAHAN	iv
HALAMAN PERSEMBAHAN.....	v
MOTTO.....	vi
ABSTRAK	vii
ABSTRACT.....	Error! Bookmark not defined.
KATA PENGANTAR	x
DAFTAR ISI.....	xii
DAFTAR TABEL.....	xv
DAFTAR GAMBAR	xvi
DAFTAR LAMPIRAN	xviii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Ruang Lingkup Penelitian	3
1.3 Rumusan Masalah.....	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	4
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA.....	5
2.1 <i>Studi Literatur</i>	5
2.2 Dasar Teori	6
2.2.1 Kebakaran.....	6
2.2.2 Asap.....	7
2.3 Perangkat Keras Yang Digunakan.....	7
2.3.1 Sensor Asap MQ2.....	7
2.3.2 Konfigurasi Sensor MQ2.....	9
2.3.3 Prinsip Kerja Sensor MQ2.....	10
2.3.4 <i>Flame Sensor</i>	11
2.3.5 Modul Arduino Uno	12
2.3.6 Relay.....	14

2.3.6.1	Prinsip Kerja Relay.....	15
2.3.7	Logika Fuzzy (<i>Fuzzy Logic</i>).....	16
2.3.7.1	Klasifikasi Fuzzy	16
2.3.7.2	Himpunan Crisp Dan Himpunan Fuzzy	18
2.3.7.3	Fungsi Keanggotaan	19
2.3.8	Sistem Interferensi Fuzzy.....	23
2.3.8.1	Metode Penalaran Monoton.....	23
2.3.8.3	Mendefinisikan karakteristik model secara fungsional dan operasional	23
2.3.8.4	Melakukan dekomposisi variabel model menjadi himpunan fuzzy	23
2.3.8.5	Membuat aturan fuzzy	23
2.3.8.6	Menentukan metode defuzzy untuk tiap-tiap variabel solusi ..	24
2.3.9	<i>Internet of Things</i>	24
2.3.10	Telegram dengan Telegram Bot dan API.....	25
2.3.11	NodeMCU ESP8266	26
2.4	Perangkat Lunak Yang Digunakan.....	28
2.4.1	<i>Software</i> Mikrokontroler Arduino Uno	28
2.4.1.1	Program Arduino IDE.....	29
BAB III METODOLOGI PENELITIAN		31
3.1	<i>Studi Literatur</i>	31
3.2	Analisa Perancangan Sistem.....	31
3.2.1	Perancangan Perangkat Keras	33
3.2.1.1	Rangkaian <i>Power Supply</i>	33
3.2.1.2	Rancangan Rangkaian Sensor MQ2	34
3.2.1.3	Rancangan Rangkaian Sensor <i>Flame</i>	35
3.2.3.4	Rangkaian <i>Relay</i>	36
3.2.1.4	Rangkaian <i>Keseluruhan</i>	36
3.2.2	Perancangan Perangkat Lunak	37
3.3	Analisa Kebutuhan	43
3.3.1	Alat.....	43
3.3.2	Komponen.....	44

3.3.3	Software.....	44
3.4	Implementasi.....	44
3.4.1	Implementasi Perangkat Keras	45
3.4.2	Implementasi Perangkat Lunak	45
3.4.3	Pembuatan Bot Via Telegram.....	46
3.5	Pengujian Sistem.....	48
3.5.1	Pengujian Rangkaian Sensor MQ2.....	48
3.5.2	Pengujian Rangkaian Sensor Api	49
3.5.3	Pengujian <i>fuzzifikasi</i>	50
3.5.4	Pengujian Rangkaian <i>Telegram</i>	51
3.5.5	Pengujian Sistem Keseluruhan	52
3.6	Analisis Kerja	52
BAB IV HASIL DAN PEMBAHASAN		53
4.1	Hasil Perakitan	53
4.1.1	Hasil Pengujian dan Pembahasan	54
4.1.2	Pengujian Sensor MQ2.....	54
4.1.3	Pengujian Flame	57
4.1.4	Hasil Pengujian Pengiriman Telegram.....	58
4.1.5	Hasil Pengujian Sistem Keseluruhan.....	58
BAB V KESIMPULAN DAN SARAN.....		61
5.1	Kesimpulan.....	61
5.2	Saran	61
DAFTAR PUSTAKA		63

DAFTAR TABEL

Tabel 3.1 Tabel Aturan Dasar Fuzzy	40
Tabel 3.2 Alat Yang Dibutuhkan	43
Tabel 3.3 Komponen Yang Dibutuhkan	44
Tabel 3.4. Daftar <i>Software</i> Yang Digunakan	44
Tabel 4.1. Hasil Pengujian Sensor MQ2.....	56
Tabel 4.3 Hasil Pegujian Sensor Flame	57
Tabel 4.3 Hasil Ujicoba Keseluruhan	60

DAFTAR GAMBAR

Gambar 2.1 Sensor MQ-2	8
Gambar 2.2 Konstruksi Sensor MQ2	8
Gambar 2.3 Internal Sensor MQ2	9
Gambar 2.4. Konfigurasi Sensor MQ2.....	10
Gambar 2.5 Prinsip Kerja Sensor MQ2	10
Gambar 2.6 Flame Sensor	11
Gambar 2.7 Arduino Uno.....	12
Gambar 2.8 Gambar dan Simbol <i>Relay</i>	14
Gambar 2.9 <i>Relay</i>	15
Gambar 2.10 Struktur Sederhana <i>Relay</i>	16
Gambar 2.11 Sistem Kendali Fuzzy.....	17
Gambar 2.12 Sistem Klasifikasi Fuzzy	17
Gambar 2.13 Sistem Diagnosis Fuzzy	18
Gambar 2.14 Representasi Linear Naik	20
Gambar 2.15 Representasi Linear Turun	21
Gambar 2.16 Kurva Segitiga.....	21
Gambar 2.17 Kurva Trapesium.....	22
Gambar 2.18 Kurva Bentuk Bahu	23
Gambar 2.19. Ilustrasi dari <i>Internet Of Things</i>	25
Gambar 2.20, GPIO NodeMCU ESP8266 v3	27
Gambar 2.21 Tampilan Program <i>Arduino Uno</i>	29
Gambar 3.1. Alur Penelitian.....	31
Gambar 3.2. Blok Diagram Sistem	32
Gambar 3.3. Blok Diagram Sistem Pengiriman Telegram.....	32
Gambar 3.3 Rangkaian <i>Power Supply</i>	33
Gambar 3.4 Rangkaian Sensor MQ2.....	34
Gambar 3.5. Perancangan Rangkaian Sensor flame.....	35
Gambar 3.5 Rangkaian <i>Relay</i>	36
Gambar 3.7 Rangkaian Keseluruhan.....	37
Gambar 3.8 Diagram keanggotaan asap.....	38
Gambar 3.9 Diagram keanggotaan api.....	39
Gambar 3.10 Diagram Output fuzzy	40
Gambar 3.12 Prangkat Lunak Arduino	45

Gambar 3.13 Pilih <i>Start</i>	46
Gambar 3.14 Pilih <i>Getid</i>	46
Gambar 3.15 Pilih <i>Start</i>	47
Gambar 3.16 Pilih <i>New Bot</i>	47
Gambar 3.17 Masukkan Alamat Id Bot	48
Gambar. 4.1. Bentuk Fisik Alat	53
Gambar 4.2 Diagram Keanggotaan Asap Renggang	54
Gambar 4.3 Diagram Keanggotaan Asap Sedang.....	55
Gambar 4.4 Diagram Keanggotaan Asap Pekat.....	55
Gambar 4.5 Diagram Keanggotaan Api.....	57
Gambar 4.2 Hasil Pengiriman Telegram.....	58
Gambar 4.3 Kadar Gas Pekat Dan Kadar Panas Dekat.....	59
Gambar 4.4 Kadar Gas Sedang Dan Kadar Panas Jauh.....	59
Gambar 4.5 Kondisi Kadar Gas Renggang Dan Kadar Panas Aman	59
Gambar 4.6 Kondisi Kadar Gas Renggang Dan Kadar Panas	59

DAFTAR LAMPIRAN

Lampiran 1. Program	48
Lampiran 2. Datasheet <i>Arduino</i> Uno.....	52
Lampiran 3. Datasheet Sensor <i>MQ2</i>	62
Lampiran 4. Datasheet Sensr <i>Flame</i>	65
Lampiran 5. Datasheet <i>Node MCU</i>	68

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kebakaran merupakan salah satu musibah yang sering terjadi baik di beberapa kota besar maupun di pedesaan yang selalu merenggut korban jiwa. Hampir setiap hari kita mendengar terjadinya kebakaran melalui media surat kabar atau siaran televisi tentang musibah kebakaran yang terjadi baik di kawasan pemukiman penduduk, gedung perkantoran, hotel, pertokoan, gudang, atau pasar. Bencana kebakaran sangatlah berbahaya dikarenakan dapat selalu menyebabkan korban jiwa. Selain itu kebakaran yang terjadi di kawasan penghunian ataupun perdagangan akan menyebabkan kerugian mulai dari material dan ekonomi yang dapat menghambat segalanya.

Ada beberapa hal yang perlu diperhatikan saat terjadinya kebakaran, yaitu manusia, harta, material dan bangunan lainnya yang berada disekitar bangunan itu sendiri. Berdasarkan data dari Badan Nasional Penanggulangan Bencana (BNPB), terdapat 399 kebakaran pemukiman di tahun 2017, sedangkan pada tahun berikutnya sampai dengan tanggal 6 Oktober, pemukiman warga yang telah terbakar mencapai 406 kebakaran (BNPB, 2018). Berdasarkan hasil dari BNPD dapat diketahui telah terjadi peningkatan frekuensi kebakaran pemukiman warga dari tahun 2017 ke tahun 2018. Keselamatan merupakan suatu hal utama yang perlu diperhatikan ketika terjadinya kebakaran pada sebuah bangunan. Berdasarkan data dari BNPB mengenai kebakaran pemukiman warga yang terjadi di Indonesia dari tahun 2017 ada 79 korban jiwa meninggal dunia, 347 jiwa terluka, 11284 jiwa yang harus mengungsi diakibatkan oleh kehilangan tempat tinggal dan 8706 menderita penyakit ISPA. Kerugian yang dialami mencapai sekitar 10606 bangunan dan 36 unit kendaraan bermotor. (Zain, 2018)

Suatu lokasi kawasan perdagangan seperti pusat pertokoan akan menyebabkan terjadinya kebakaran yang lebih besar jika tidak segera ditangani karena bangunan

pada pertokoan umumnya jaraknya sangatlah berdekatan antara satu toko dengan toko yang lain. Sehingga api dapat dengan cepat meluas menjadi lebih besar pada sore atau malam hari, apabila bangunan tidak diawasi secara terus menerus. Permasalahan yang sering terjadi pada kebakaran adalah lambatnya petugas dalam memadamkan api sehingga api akan cepat menyebar ke sekitarnya, hal tersebut dikarenakan sistem pemadaman api yang dilakan oleh petugas masih menggunakan penyemprotan manual sehingga perlu adanya alat yang dapat mendeteksi suatu kebakaran dan melakukan penyemprotan secara otomatis agar api dapat segera dipadamkan. Selain itu minimnya penggunaan sensor pada beberapa sistem pendeteksi kebakaran mengakibatkan kurang maksimalnya pembacaan sensor sehingga kesensitifan alat pendeteksi kebakaran tersebut berkurang, oleh sebab itu penggunaan sensor pada alat pendeteksi kebakaran tersebut harus ditambah dan menggunakan perhitungan *fuzzy*.

Peneliti menggunakan aplikasi *Telegram* yang telah *terkoneksi* dengan *NodeMCU* sebagai media untuk menampilkan *notifikasi* dan nilai dari pembacaan sensor *flame* dan sensor *MQ2*. Peneliti menggunakan aplikasi telegram karena telegram lebih mudah dalam pengkoneksian ke *nodemcu* dibandingkan *sms gateway*, serta cepat dalam pengiriman notifikasi peringatannya dan tentunya lebih canggih karena sudah berbentuk aplikasi modern. Selain itu peneliti menggunakan *Telegram* sebagai penampil notifikasi dan pembacaan nilai dari sensor *flame* dan sensor *MQ2* untuk memanfaatkan penerapan IOT dalam penelitian ini.

Sistem akan bekerja ketika terdapat input-an yang diperoleh oleh dua sensor tersebut. Proses akan dilakukan berdasarkan logika *fuzzy* yang telah diterapkan. Pada alat ini terdapat keluaran, yaitu pompa air dan pesan via *Telegram*. Logika *fuzzy* digunakan sebagai dasar pemikiran untuk menentukan keluaran dari sistem ini, yaitu notifikasi ke *Telegram* dan *Relay* untuk menghidupkan pompa air. Alasan menggunakan *Fuzzy* karna logika *Fuzzy* sangat *fleksibel*, memiliki toleransi, logika *Fuzzy* mampu memodelkan fungsi – fungsi non linier yang sangat kompleks, logika *Fuzzy* dapat bekerjasama dengan teknik – teknik kendali secara

konvensional, serta didasarkan pada bahasa alami. Di dalam fuzzy ini dianalisa semua data-data masukan yang didapatkan dari sensor asap dan sensor flame. Metode inferensi *fuzzy* yang digunakan yaitu metode mamadani.

Dari pembahasan tersebut maka peneliti akan merancang alat penelitian skripsi dengan judul **“IMPLEMENTASI FUZZY LOGIC CONTROL PADA SISTEM PERINGATAN DINI KEBAKARAN BERBASIS IOT”** Sistem kerja dari alat ini yaitu, Jika sensor *MQ2* mendeteksi adanya asap dan sensor *Flame* mendeteksi adanya api maka mengirimkan sinyal ke *Arduino*, *Arduino* menghidupkan *Relay* yang tersambung ke pompa air dan mengirim perintah ke *NodeMCU* untuk mengirimkan informasi melalui *wifi* ke aplikasi *Telegram* yang telah terhubung dengan *NodeMCU* hasil pembacaan sensor akan ditampilkan pada *chatting* aplikasi *Telegram*.

1.2 Ruang Lingkup Penelitian

Berdasarkan dari penelitian yang akan dilakukan, maka ruang lingkup dalam penelitian ini, yaitu;

1. Alat ini hanya berbentuk prototype.
2. Peneliti hanya membahas tentang alat yang digunakan yaitu sensor *MQ2*, Sensor *Flame*, *NodeMCU*, dan *Arduino*.
3. *NodeMCU* dan *Arduino* yang digunakan proses dari seluruh sistem.
4. Hanya mengirimkan notifikasi yang berisi informasi nilai pembacaan sensor jika terjadi kebakaran melalui aplikasi *Telegram*.
5. Peneliti hanya menggunakan 2 sensor *MQ2* dan 2 sensor *Flame*
6. Bagaimana mengimplementasikan metode inferensi *fuzzy* pada alat deteksi asap dan api.

1.3 Rumusan Masalah

Berdasarkan dari latar belakang diatas dapat diuraikan rumusan permasalahan dalam penelitian ini, bagaimana cara membuat dan merancang suatu peringatan dini kebakaran fuzzy logic control yang terkoneksi ke telegram.

1.4 Tujuan Penelitian

Merancang dan membuat alat yang dapat memberikan informasi jika terjadinya kebakaran melalui telegram menggunakan *Arduino*, *NodeMCU*, sensor *MQ2* dan sensor *Flame*.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah

1. Dapat meminimalisir terjadinya korban kebakaran
2. Dapat menerima informasi jika terjadinya kebakaran melalui aplikasi telegram.
3. Dapat diperoleh suatu alat pemantau dan pemadam kebakaran otomatis dalam suatu ruangan.
4. Bagi penulis, sebagai bentuk Tri Dharma Perguruan Tinggi dalam bentuk penelitian untuk mencapai gelar Sarjana.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam tugas akhir ini terbagi dalam beberapa pokok bahasan, yaitu :

BAB I PENDAHULUAN

Dalam bab ini berisikan latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian dan manfaat penelitian.

BAB II TINJAUAN PUSTAKA

Bab ini berisikan tentang teori – teori yang berkaitan dengan “Implementasi Fuzzy Logic Control Pada Sistem Peringatan Dini Kebakaran Berbasis Iot”

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan apa yang akan digunakan dalam uji coba pembuatan alat, tahapan perancangan dari alat, diagram blok dari alat, dan cara kerja alat tersebut.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang implementasi alur, analisis dan pembahasan dari alur yang dirancang.

BAB V SIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dari pengujian sistem serta saran apakah rangkaian ini dapat digunakan secara tepat dan dikembangkan perakitannya.

BAB II

TINJAUAN PUSTAKA

2.1 Studi Literatur

Penelitian tentang proteksi kebakaran ini sudah pernah dilakukan oleh beberapa peneliti. Beberapa ringkasan *Studi Literatur* digunakan untuk mengetahui sejauh mana penelitian tersebut sudah dilakukan.

Salah satu jurnal yang menjadi referensi peneliti yaitu yang dilakukan oleh Salah satu jurnal yang menjadi referensi peneliti yaitu yang dilakukan oleh (Zain, 2018) dengan judul Rancang Bangun Sistem Proteksi Kebakaran Menggunakan Smoke dan Heat Detector. Penelitian bertujuan membuat alat yang dapat mencegah terjadinya kebakaran dengan mendeteksi suatu asap dan panas yang berlebihan di sekitar rumah. Hasil dari penelitian adalah. Apabila terjadi bahaya asap atau panas, maka sistem akan berkerja untuk memberikan peringatan dengan bunyi alarm, yang digunakan sebagai penunjuk ke jalur *emergency*, serta akan memadamkan api tersebut sebelum api akan membesar.

Peneliti selanjutnya dengan judul Prototipe Sistem Pemadam Kebakaran Otomatis Berbasis Mikrokontroler Atmega328p karya Mega Apriyaningsih, Abdul Muid, Nurhasanah (2017) sistem ini dapat memberikan peringatan dini berupa layanan SMS (*Short Message Service*) dan menyalakan pompa air saat terdeteksi asap atau suhu $>40^{\circ}\text{C}$. Perancangan Sistem Pendeteksi Kebakaran Rumah Penduduk Pada Daerah Perkotaan Berbasis Mikrokontroler karya Dodon Yendri, Wildian, Amalia Tiffany (2017) sistem pendeteksi kebakaran ini yang bekerja secara *realtime* dan sekaligus mengetahui lokasi kebakaran terjadi.

Sistem ini menggunakan sensor suhu (LM35) dan sensor asap (MQ-9) berbasis mikrokontroler untuk mengukur suhu dan asap kebakaran. Realisasi Sistem Peringatan Kebakaran Melalui Layanan SMS dan MMS karya Hendri Andrianto, M.D. Awaludin Hakim (2011) sistem deteksi ini ketika suhu ruangan lebih dari

35°C dan kadar asap rangan melebihi nilai 200, maka *buzzer*, pompa, *sprinkle* akan aktif serta alat dapat mengirimkan SMS dan MMS.

Rancang Bangun Sistem Detektor Kebakaran Via Handphone Berbasis Mikrokontroler karya (Apriyandi, 2013) sistem ini akan mengirim SMS “Bahaya Kebakaran” ke pemilik jika mendeteksi adanya sumber api dan asap dan sekalian menghidupkan *buzzer*. Sistem Peringatan Dini Akan Bahaya Kebakaran karya Wiweko, Hang Suharto (2008) sistem ini akan memberikan peringatan dini kepada lingkungan sekitar bila terjadi kebakaran dengan cara mengaktifkan alarm, penyemprot air dan mengirimkan sms kepada pemilik alat.

Dan yang terakhir jurnal tentang model peringatan kebakaran dengan fuzzy mamdani karya (Riri Irawati, 2017) Model peringatan kebakaran secara ini dirancang agar dapat mengetahui ada tidaknya kebakaran dengan memberikan peringatan dini berupa suara alarm buzzer dan pengiriman. Alat akan mendeteksi adanya api dan kebocoran gas sehingga dapat mencegah kebakaran lebih awal. Untuk dapat mengetahui keberadaan api dan kebocoran gas digunakan sensor api dan sensor gas LPG. Mikrokontroler Arduino berfungsi untuk mengolah sinyal yang dikirimkan oleh sensor kemudian memberikan perintah untuk mendeteksi api, menyalakan buzzer alarm, menghidupkan display dan memberikan peringatan awal berupa sms gateway ke divisi pemadam kebakaran dan sms ke lima nomor mobile phone. Pada penelitian ini akan menggunakan metode logika fuzzy Mamdani dimana metode tersebut memproses aturan-aturan yang dibuat user yang memerintah system control target. Hasil yang didapat dari penelitian ini adalah keakuratan hasil pembacaan sensor gas dan sensor api pada skala 70% - 100%.

2.2 Dasar Teori

2.2.1 Kebakaran

Kebakaran adalah peristiwa yang sangat sering dan cepat terjadi secara tidak dikehendaki. Akibat dari kebakaran dapat menimbulkan kerusakan atau kerugian yang sangat fatal, hal ini disebabkan ketidakdisiplinan dalam menggunakan bahan-bahan, peralatan instalasi listrik, atau kelalaian manusia Dalam

menggunakan alat dan bahan yang berhubungan dengan api. Sebab lain dari kebakaran adalah karena bencana alam.

Api merupakan reaksi kimia dari suatu bahan eksotermik yang disertai dengan timbulnya panas/kalor, cahaya (nyala), asap, gas, dan hasil dari bahan yang terbakar. Proses ini dapat dinamakan dengan reaksi pembakaran. Pencegahan kebakaran merupakan usaha mencegah akan faktor-faktor yang menjadi sebab munculnya atau terjadinya kebakaran (Anizar, 2009).

2.2.2 Asap

Asap merupakan suspensi dari hasil partikel kecil yang dihasil oleh udara (aerosol) yang berasal dari pembakaran tak sempurna suatu bahan bakar. Asap merupakan produk lain yang tidak diinginkan dari suatu pembakaran (termasuk kompor dan lampu) yang selalu ada karena pembakaran yang kurang sempurna, tapi dapat juga digunakan untuk pembasmian hama serangga (fumigasi), komunikasi darurat (sinyal asap), pertahanan (layar asap, smoke-screen) atau penghirupan tembakau atau obat bius. Asap kadang digunakan sebagai perisa agen pemberi rasa (flavoring agent), pengawet untuk berbagai bahan makanan agar kadar air dalam bahan makanan tersebut berkurang sehingga memperlambat proses pembusukan, dan bahan baku asap cair (liquid smoke) (Wardhana,2001).

2.3 Perangkat Keras Yang Digunakan

2.3.1 Sensor Asap MQ2

Sensor MQ2 yaitu salah satu sensor yang sensitif terhadap asap rokok. Bahan utama sensor MQ2 adalah SnO₂ dengan konduktifitas rendah pada udara bersih. Jika terjadi kebocoran gas konduktifitas sensor menjadi lebih tinggi, setiap kenaikan konsentrasi gas maka konduktifitas sensor juga naik. MQ2 sensitif terhadap gas LPG, Hidrogen, Propana, Karbon Monoksida, Alkohol dan Metana serta gas mudah terbakar diudara lainnya.

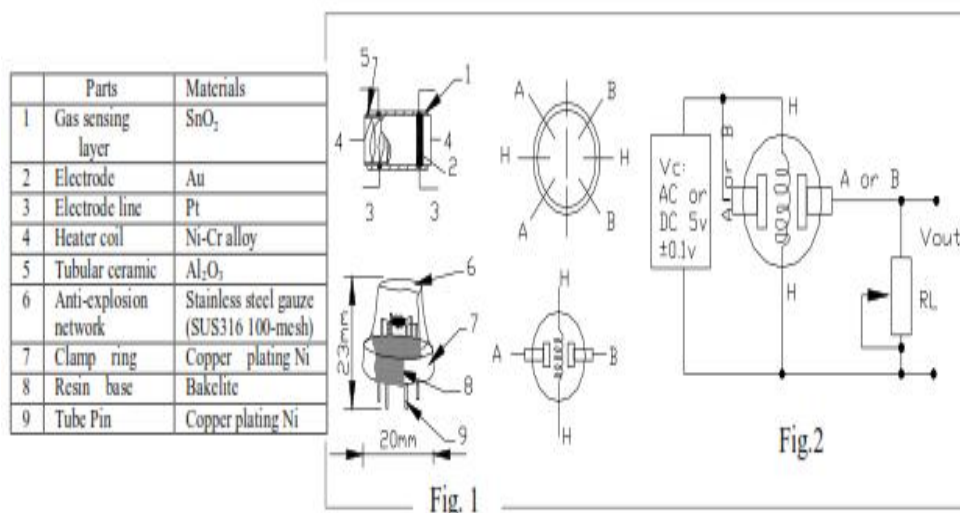


Gambar 2.1 Sensor MQ-2

(Sumber : <https://www.components101.com>)

Sensor MQ2 terdapat 2 masukan tegangan yaitu VH dan VC. VH berguna untuk tegangan pada pemanas (Heater) internal dan VC merupakan tegangan sumber. Catu daya yang dibutuhkan pada sensor MQ2 yaitu VC <24VDC dan VH = 5V ±0.2V tegangan AC atau DC.

Sensor ini dapat mengukur konsentrasi gas mudah terbakar dari 300 sampai 10.000 sensor ppm. Dapat beroperasi pada suhu dari -20° C sampai 50 ° C dan mengkonsumsi kurang dari 150 mA pada 5V. Sensor gas dan asap ini mendeteksi konsentrasi gas yang mudah terbakar di udara serta asap dan output membaca sebagai tegangan analog. Dibawah ini merupakan gambar bentuk, internal sensor MQ2.

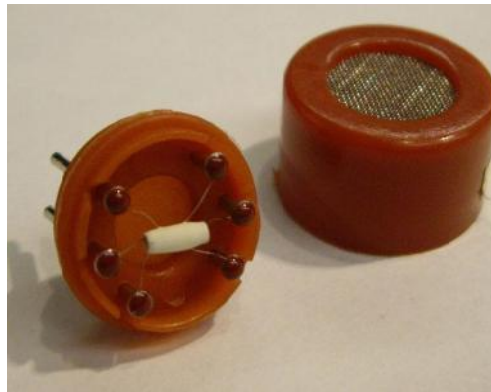


Gambar 2.2 Konstruksi Sensor MQ2

(Sumber : <https://www.components101.com>)

Internal sensor MQ2 ini terdapat 6 buah pin :

1. Empat pin yang lain digunakan untuk memberikan masukan atau mengambil output.
2. Dua pin digunakan untuk sistem pemanas dalam tabung.



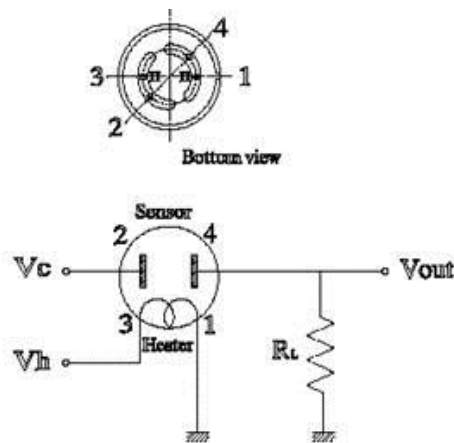
Gambar 2.3 Internal Sensor MQ2

(Sumber : <https://www.components101.com>)

2.3.2 Konfigurasi Sensor MQ2

Sensor MQ2 terdapat 2 masukan tegangan yaitu VH dan VC. VH berguna untuk tegangan pada pemanas (Heater) internal dan VC merupakan tegangan sumber dan memiliki keluaran yang menghasilkan tegangan berupa tegangan analog. Berikut konfigurasi dari sensor MQ2 :

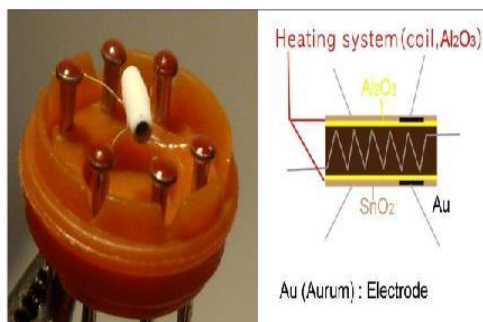
1. Pin 1, heater internal yang terhubung dengan ground.
2. Pin 2, tegangan sumber (VC) yaitu $VC < 24 \text{ VDC}$.
3. Pin 3 (VH) berguna untuk tegangan pada pemanas (heater internal) yaitu $VH = 5\text{VDC}$.
4. Pin 4, output yang akan menghasilkan tegangan analog.



Gambar 2.4. Konfigurasi Sensor MQ2
 (Sumber : <https://www.components101.com>)

2.3.3 Prinsip Kerja Sensor MQ2

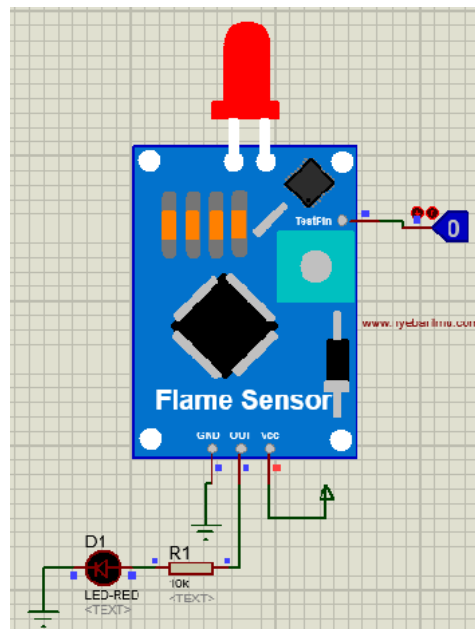
Sensor Asap MQ2 yang berfungsi untuk mendeteksi keberadaan asap yang berasal dari gas mudah terbakar di udara. Pada dasarnya sensor ini terdiri dari tabung aluminium yang dikelilingi oleh silikon dan dipusatnya ada elektroda yang terbuat dari aurum di mana ada elemen pemanasnya. Ketika terjadi proses pemanasan, kumparan akan dipanaskan sehingga SnO₂ keramik menjadi semikonduktor atau sebagai penghantar sehingga melepaskan elektron kemudian ketika asap dideteksi oleh sensor dan mencapai aurum elektroda maka output sensor MQ2 akan menghasilkan tegangan analog. Sensor MQ2 ini memiliki 6 buah masukan yang terdiri dari tiga buah power supply (Vcc) sebesar +5 volt untuk mengaktifkan heater dan sensor, Vss (Ground dan pin keluaran dari sensor tersebut).



Gambar 2.5 Prinsip Kerja Sensor MQ2
 (Sumber : <https://www.components101.com>)

2.3.4 Flame Sensor

Flame sensor merupakan sensor yang mempunyai fungsi sebagai pendeteksi nyala api yang dimana api tersebut memiliki panjang gelombang antara 760nm – 1100nm. Sensor ini menggunakan infrared sebagai transduser dalam mensensing kondisi nyala api. Cara kerja sensor ini yaitu dengan mengidentifikasi atau mendeteksi nyala api dengan menggunakan metode optik. Pada sensor ini menggunakan transduser yang berupa infrared (IR) sebagai sensing sensor. Transduser ini digunakan untuk mendeteksi akan penyerapan cahaya pada panjang gelombang tertentu.



Gambar 2.6 Flame Sensor
(Sumber : <https://www.components101.com>)

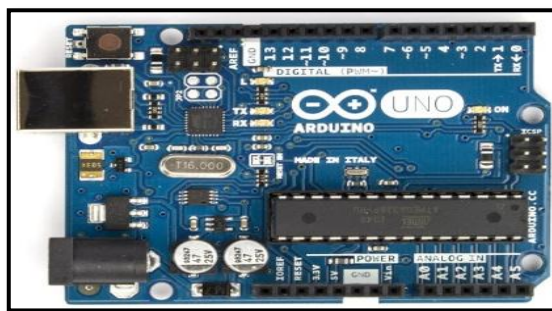
Fitur dari flame sensor

- Tegangan operasi antara 3,3 – 5 Vdc
- Terdapat 2 output yaitu digital output dan analog output yang berupa tegangan
- Sudah terpackage dalam bentuk modul

- Terdapat potensiometer sebagai pengaturan sensitivitas sensor dalam mensensing

2.3.5 Modul Arduino Uno

Modul Arduino Uno adalah papan sirkuit berbasis Mikrokontroler AT Mega 328. *IC (integrated circuit)* ini memiliki 14 masukan/keluaran digital (6 keluaran untuk PWM), 6 analog masukan, resonator kristal keramik 16 MHz, Koneksi *USB (Universal Serial Bus)*, soket adaptor, pin *header ICSP*, dan tombol *reset*. Hal inilah yang dibutuhkan untuk *mensupport*. Mikrokontroler secara mudah terhubung dengan kabel *power USB* atau kabel *power supply* adaptor AC ke DC atau juga *battery* (Arduino, 2016). Bentuk fisik Arduino Uno seperti pada gambar 2.7.



Gambar 2.7 Arduino Uno
(Sumber <https://www.arduino.com>,2016)

- 3 *UART (Universal Asynchronous Receiver/Transmitter)* adalah antar muka yang digunakan untuk komunikasi serial seperti pada RS-232, RS-422 dan RS-485.
- 4 2 KB RAM pada *memory* kerja bersifat *volatile* (hilang saat daya dimatikan), digunakan oleh variable-variabel di dalam program.
- 5 32 KB RAM *flash memory* bersifat *non-volatile*, digunakan untuk menyimpan program yang dimuat dari komputer. Selain program, *flash memory* juga menyimpan *bootloader*. *Bootloader* adalah program inisiasi yang ukurannya kecil, dijalankan oleh CPU saat daya dihidupkan. Setelah *bootloader* selesai dijalankan, berikutnya program di dalam RAM akan dieksekusi.

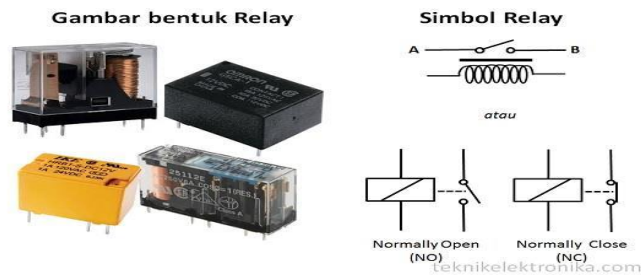
- 6 1 Kb *eeprom* bersifat *non-volatile*, digunakan untuk menyimpan data yang tidak boleh hilang saat daya dimatikan. Tidak digunakan pada papan Arduino.
- 7 CPU, bagian dari Mikrokontroler untuk menjalankan setiap instruksi dari program.
- 8 *Port* masukan/keluaran, pin-pin untuk menerima data digital atau analog, dan mengeluarkan data digital atau analog.
- 9 14 pin masukan/keluaran digital (0-13).
Berfungsi sebagai masukan atau keluaran, dapat diatur oleh program. Khusus untuk 6 buah pin 3, 5, 6, 9, 10 dan 11, dapat juga berfungsi sebagai pin analog keluaran dimana tegangan keluaran-nya dapat diatur. Nilai sebuah pin keluaran analog dapat diprogram antara 0 – 255, dimana hal itu mewakili nilai tegangan 0 – 5V.
- 10 USB berfungsi untuk memuat program dari komputer ke dalam papan, memberi daya listrik kepada papan dan komunikasi serial antara papan dan komputer.
- 11 Sambungan SV1 atau jumper untuk memilih sumber daya papan, apakah dari sumber eksternal atau menggunakan USB. Sambungan ini tidak diperlukan lagi pada papan Arduino versi terakhir karena pemilihan sumber daya eksternal atau USB dilakukan secara otomatis.
- 12 Q1 – Kristal (*quartz crystal oscillator*) Jika Mikrokontroler dianggap sebagai sebuah otak, maka kristal adalah jantung-nya karena komponen ini menghasilkan detak-detak yang dikirim kepada Mikrokontroler agar melakukan sebuah operasi untuk setiap detaknya. Kristal ini dipilih yang berdetak 16 juta kali per detik (16MHz).
- 13 Tombol *Reset* S1 Untuk mereset papan sehingga program akan mulai lagi dari awal. Perhatikan bahwa tombol *reset* ini bukan untuk menghapus program atau mengosongkan Mikrokontroler.
- 14 *In Circuit Serial Programming (ICSP) Port ICSP* memungkinkan pengguna untuk memprogram Mikrokontroler secara langsung, tanpa melalui *bootloader*. Umumnya pengguna Arduino tidak melakukan ini sehingga *ICSP* tidak terlalu dipakai walaupun disediakan.

- 15 IC 1 – Mikrokontroler Atmega Komponen utama dari papan Arduino, di dalamnya terdapat CPU, ROM dan RAM.
- 16 X1 – sumber daya eksternal, jika hendak disuplai dengan sumber daya eksternal, papan Arduino dapat diberikan tegangan DC antara 9-12V.
- 17 6 pin masukan analog (0-5). Pin ini sangat berguna untuk membaca tegangan yang dihasilkan oleh sensor analog, seperti sensor suhu. Program dapat membaca nilai sebuah pin masukan antara 0 – 1023, dimana hal itu mewakili nilai tegangan 0 – 5V.

2.3.6 Relay

Relay merupakan bentuk hambatan terdiri atas titik-titik kontak bawah dengan gulungan *spool*-nya tidak bergerak dan titik kontak bagian atas yang bergerak. Prinsip kerja hambatan adalah menghubungkan titik-titik kontak bagian bawah dengan titik bagian atas yaitu terletak gulungan *spool* dialiri arus listrik yang timbul elektromagnet. (Handy Wicaksono,1996,1-12). *Relay* merupakan bentuk hambatan terdiri atas titik-titik kontak bawah dengan gulungan *spool*-nya tidak bergerak dan titik kontak bagian atas yang bergerak. Prinsip kerja hambatan adalah menghubungkan titik-titik kontak bagian bawah dengan titik bagian atas yaitu terletak gulungan *spool* dialiri arus listrik yang timbul elektromagnet. (Handy Wicaksono,1996,1-12). Secara sederhana *relay* elektromekanis ini didefinisikan sebagai berikut :

1. Alat yang menggunakan gaya elektromagnetik untuk menutup (atau membuka) kontak saklar.
 2. Saklar yang digerakkan (secara mekanis) oleh daya/energi listrik.
- Dibawah ini adalah gambar fisik, bentuk dan Simbol *Relay* yang sering ditemukan di Rangkaian Elektronika.



Gambar 2.8 Gambar dan Simbol Relay

(<http://teknikelektronika.com/pengertian-relay-fungsi-relay/>)



Gambar 2.9 Relay

(Sumber : Kilian, Christopher T, Modern Control Technology, (West Published Co : 1996)

Bagian titik kontak dibagi menjadi 2 bagian yaitu bagian kontak utama dan kontak bantu yaitu : Bagian kontak utama gunanya untuk menghubungkan dan memutuskan arus listrik bagian yang menuju beban/pemakai. Bagian kontak bantu gunanya untuk menghubungkan dan memutuskan arus listrik ke bagian yang menuju bagian pengendali. Kontak bantu mempunyai 2 kontak yaitu kontak hubung (NC) dan kontak putus (NO) menandakan masing-masing kontak dan gulungan *spool*. Secara umum, *relay* digunakan untuk memenuhi fungsi–fungsi berikut :

1. *Remote control* : Dapat menyalakan atau mematikan alat dari jarak jauh.
2. Penguatan daya : Memperkuat arus atau tegangan.
3. Pengatur logika kontrol suatu sistem. Susunan kontak pada *relay* adalah :
 - *Normally Open* : *Relay* akan menutup bila dialiri arus listrik.
 - *Normally Close* : *Relay* akan membuka bila dialiri arus listrik.

2.3.6.1 Prinsip Kerja Relay

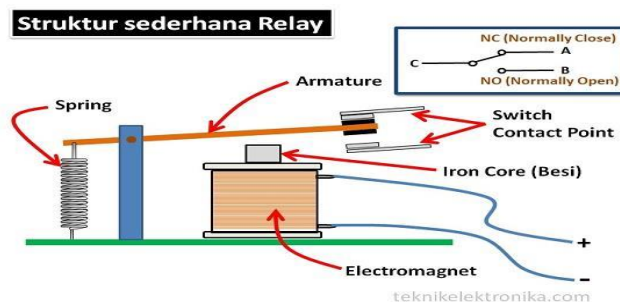
Pada dasarnya, *Relay* terdiri dari 4 komponen dasar yaitu :

1. *Electromagnet (Coil)*
2. *Armature*
3. *Switch Contact Point (Saklar)*
4. *Spring*

Seperti saklar, *relay* juga dibedakan berdasar *pole* dan *throw* yang dimilikinya.

1. *Pole* : Banyaknya *contact* yang dimiliki oleh *relay*.
2. *Throw* : Banyaknya kondisi (*state*) yang mungkin dimiliki *contact*.

Berikut ini merupakan gambar dari bagian-bagian *Relay* :



Gambar 2.10 Struktur Sederhana Relay

(<http://teknikelektronika.com/pengertian-relay-fungsi-relay/>)

Kontak *normally open* akan membuka ketika tidak ada arus mengalir pada kumparan, tetapi tertutup secepatnya setelah kumparan menghantarkan arus atau diberi tenaga. Kontak *normally close* akan tertutup apabila kumparan tidak diberi tenaga dan membuka ketika kumparan diberi daya. Masing-masing kontak biasanya digambarkan sebagai kontak yang tampak dengan kumparan tidak diberi tenaga atau daya. *Relay* terdiri dari 2 terminal *trigger*, 1 terminal *input* dan 1 terminal *output*.

1. Terminal *trigger* : Yaitu terminal yang akan mengaktifkan relay, seperti alat elektronik lainya relay akan aktif apabila di aliri arus (+) dan arus (-). Pada contoh *relay* yang kita gunakan terminal *trigger* ini adalah 85 dan 86.
2. Terminal *input* : Yaitu terminal tempat kita memberikan masukan, pada contoh adalah terminal 30.

3. Terminal *output* : Yaitu tempat keluarnya *output* pada contoh adalah terminal 87.

2.3.7 Logika Fuzzy (*Fuzzy Logic*)

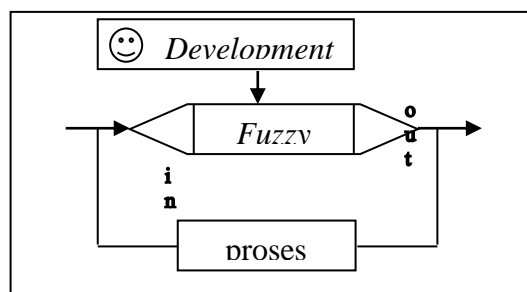
Logika fuzzy adalah suatu cara pandang yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output. Ada beberapa cara untuk memetakan input ke output diantaranya,

1. Sistem fuzzy
2. Sistem linear
3. Sistem pakar
4. Jaringan syaraf
5. Persamaan differensial
6. Tabel interpolasi multi-dimensi
7. Dll

2.3.7.1 Klasifikasi Fuzzy

Sistem fuzzy memiliki beberapa jenis klasifikasi, yaitu : sistem kendali fuzzy, sistem klasifikasi fuzzy dan sistem diagnosis fuzzy. Pada ruang lingkup yang lebih luas lagi, masih ada sistem lainnya yang dapat digunakan dan terbukti cukup sukses diantaranya, sistem pakar fuzzy, sistem analisa data fuzzy, sistem pengolahan citra fuzzy dan berbagai ragam aplikasi sistem fuzzy yang sudah ada.

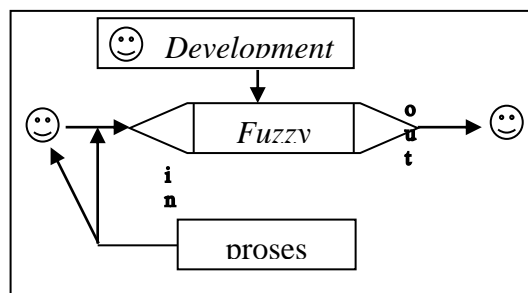
1. Sistem Kendali Fuzzy



Gambar 2.11 Sistem Kendali Fuzzy

Sistem kendali fuzzy yang digambarkan, gambar 2.8 merupakan suatu sistem lingkaran tertutup, dimana tidak terdapat operator yang menjadi bagian dari sistem lingkaran kendali (*control loop*). Contoh dari sistem ini adalah *vacuum cleaner*, dimana sistem pada alat ini mengatur daya motor penghisap tergantung dari banyaknya debu.

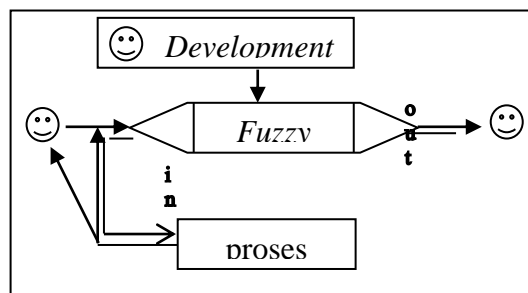
2. Sistem Klasifikasi Fuzzy



Gambar 2.12 Sistem Klasifikasi Fuzzy

Sistem klasifikasi fuzzy yang digambarkan, gambar 2.9 bukan merupakan suatu sistem lingkaran tertutup, sistem ini hanya menerima masukan dan memberi keluaran dari proses untuk selanjutnya memberikan informasi berupa kondisi (*state*), dari proses tadi informasi kondisi ini digunakan untuk mengendalikan sistem atau memberikan tanggung jawab kendali kepada operator. Contoh dari sistem ini adalah : mesin cuci fuzzy

3. Sistem Diagnosis Fuzzy



Gambar 2.13 Sistem Diagnosis Fuzzy

Pada sistem ini, gambar 2.10 peranan manusia atau operator lebih dominan. Ketika sistem memerlukan data tambahan maka, pengiriman data akan dilaksanakan oleh operator, selain itu operator dapat meminta atau menanyakan informasi dari sistem diagnosis berupa hasil koklusi diagnosis atau prosedur detail hasil diagnosis oleh sistem. Dilihat dari sifatnya, sistem diagnosis fuzzy dapat digolongkan ke dalam sistem pakar fuzzy. Sistem pakar fuzzy menggunakan notasi fuzzy pada aturan-aturan dan proses inferensi (logika keputusan).

2.3.7.2 Himpunan Crisp Dan Himpunan Fuzzy

Himpunan Crisp didefinisikan oleh item-item yang ada pada himpunan itu. Jika a anggota dari A , maka nilai yang berhubungan dengan a adalah 1. Namun, jika a bukan anggota dari A , maka nilai yang berhubungan dengan a adalah 0. Notasi $A = \{x \mid P(x)\}$ menunjukkan bahwa A berisi item x dengan $P(x)$ benar. Jika X_A merupakan fungsi karakteristik A dan properti P , maka dapat dikatakan bahwa $P(x)$ benar, jika dan hanya jika $X_A(x) = 1$.

Himpunan fuzzy didasarkan pada gagasan untuk memperluas jangkauan fungsi karakteristik sedemikian sehingga fungsi tersebut akan mencakup bilangan real pada interval $[0,1]$. Nilai keanggotaannya menunjukkan bahwa suatu item dalam semesta pembicaraan tidak hanya berada pada 0 atau 1, namun juga nilai yang terletak diantaranya. Dengan kata lain, nilai kebenaran suatu item tidak hanya bernilai benar atau salah. Nilai 0 menunjukkan salah, nilai 1 menunjukkan benar dan masih ada nilai-nilai yang terletak antara benar dan salah.

Beberapa hal yang perlu diketahui dalam memahami sistem fuzzy :

1. Variabel Fuzzy

Merupakan variabel yang hendak dibahas dalam suatu sistem fuzzy, contoh : umur, temperature, permintaan dan sebagainya.

2. Himpunan Fuzzy

Merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel fuzzy, contoh :

- Variabel umur dibagi menjadi 3 himpunan fuzzy : muda, parobaya, tua
- Variabel temperature ibagi menjadi 5 himpunan fuzzy : dingin, sejuk, normal, hangat dan panas.

3. Semesta Pembicaraan

Keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel fuzzy, contoh : semesta pembicaraan untuk variabel temperature : [0 40]

4. Domain

Keseluruhan nilai yang diinginkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan fuzzy.

5. Nilai Ambang *Alfa-Cut*

Merupakan nilai ambang batas domain yang didasarkan pada nilai keanggotaan untuk tiap-tiap domain, dimana α - cut memiliki 2 kondisi.

- α -cut lemah dapat dinyatakan sebagai : $\mu (x) \geq \alpha$ (2.1)
- α - cut kuat dapat dinyatakan sebagai : $\mu (x) > \alpha$ (2.2) [3]

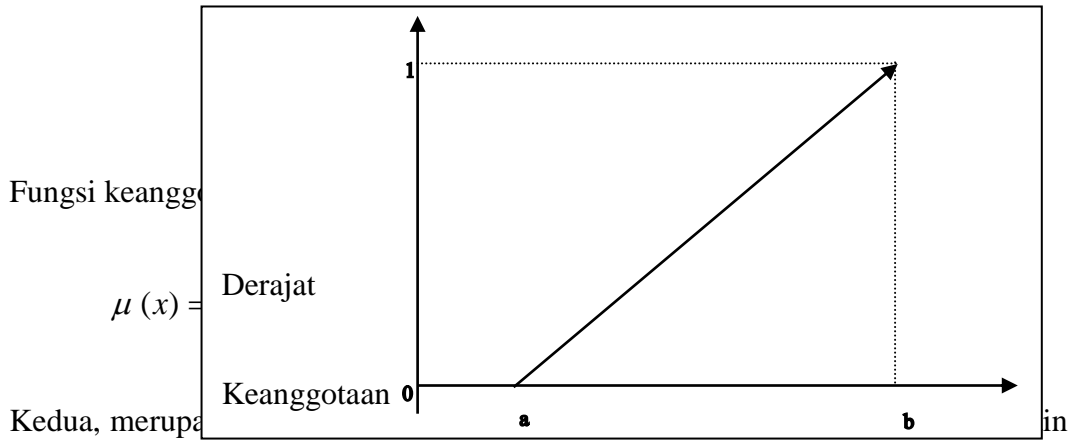
2.3.7.3 Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data kedalam nilai keanggotaannya (sering disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1.

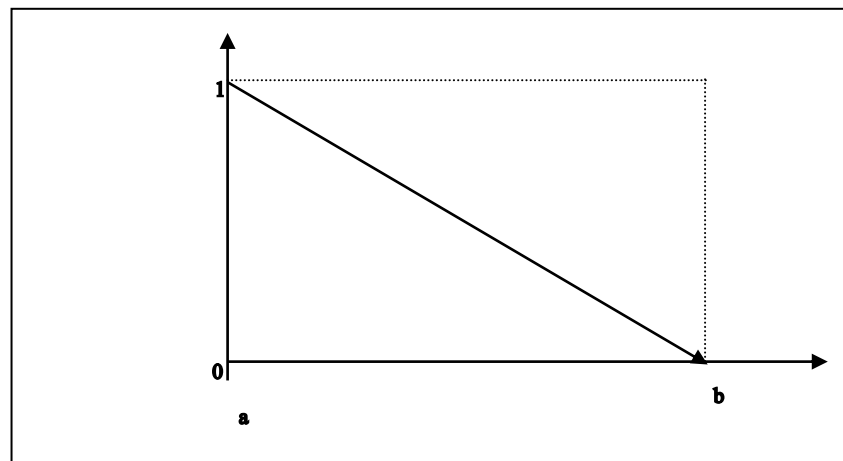
Fungsi keanggotaan dapat dibuat kedalam beberapa bentuk kurva diantaranya,

1. Representasi Linier

Pada representasi linier, permukaan digambarkan sebagai suatu garis lurus. Bentuk ini paling sederhana dan menjadi pilihan yang baik untuk mendekati suatu konsep yang kurang jelas. Ada 2 kemungkinan keadaan himpunan fuzzy yang linier. Pertama, kenaikan himpunan dimulai pada nilai dominan yang memiliki derajat keanggotaan nol [0] bergerak kekanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi seperti yang tergambar pada gambar 2.11.



Kedua, merupakan dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah pada gambar 2.12.



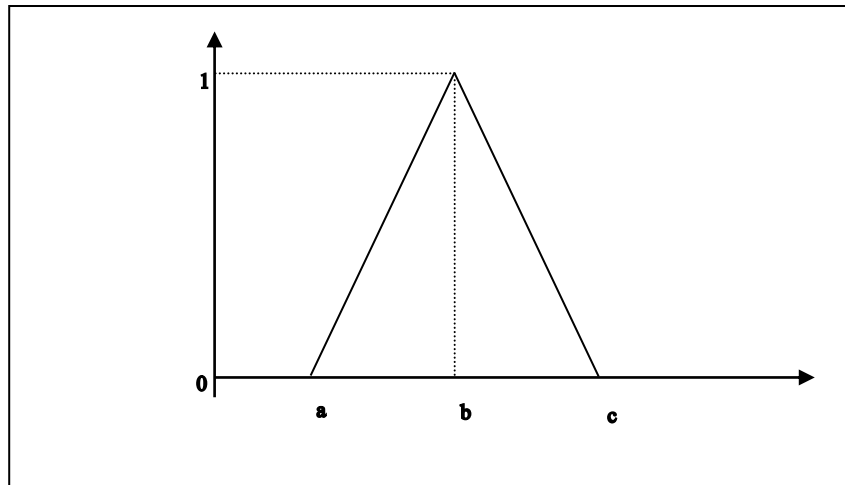
Gambar 2.15 Representasi Linear Turun

Fungsi keanggotaan :

$$\mu(x) = \begin{cases} (b-x)/(b-a); & a \leq x \leq b \\ 0; & x \geq b \end{cases} \dots\dots\dots(2.4)$$

2. Representasi Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linier seperti terlihat pada gambar 2.13).



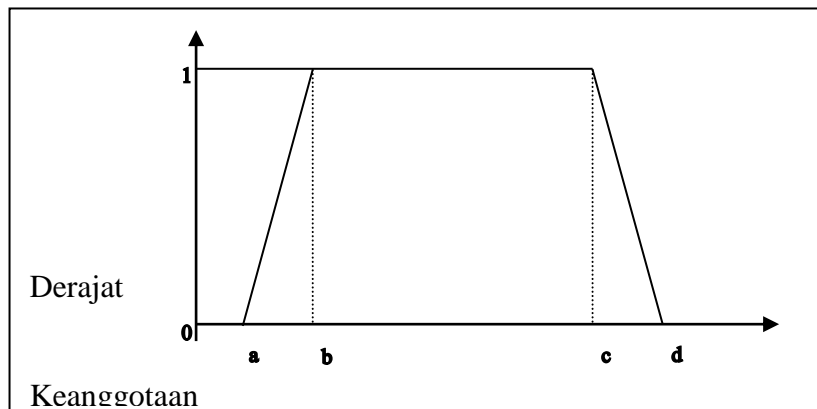
Gambar 2.16 Kurva Segitiga

Fungsi keanggotaan :

$$\mu(x) = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ (x-a)/(b-a); & a \leq x \leq b \\ (b-x)/(c-b); & b \leq x \leq c \end{cases} \dots\dots\dots(2.5)$$

3. Representasi Kurva Trapesium

Kurva trapesium pada dasarnya seperti bentuk kurva segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1 pada gambar 2.17.



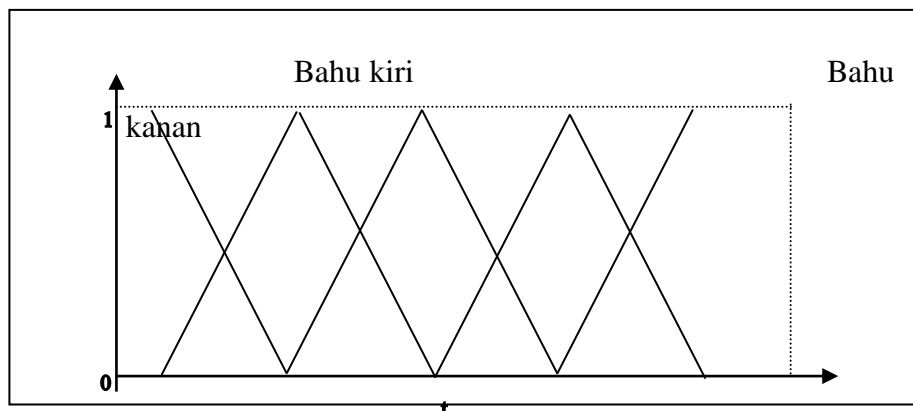
Gambar 2.17 Kurva Trapesium

Fungsi keanggotaan :

$$\mu(x) = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ (x-a)/(b-a); & a \leq x \leq b \\ 1; & b \leq x \leq c \\ (d-x)/(d-c); & c \leq x \leq d \end{cases} \dots\dots\dots(2.6)$$

4. Representasi Kurva Bentuk Bahu

Daerah yang terletak di tengah-tengah suatu variabel yang direpresentasikan dalam bentuk segitiga, pada sisi kanan dan kirinya akan naik dan turun. Tetapi terkadang salah satu sisi dari variabel tersebut tidak mengalami perubahan. Himpunan fuzzy ‘bahu’, bukan segitiga, digunakan untuk mengakhiri variabel suatu daerah fuzzy. Bahu kiri bergerak dari benar ke salah demikian juga bahu kanan bergerak dari salah ke benar pada gambar 2.18



Gambar 2.18 Kurva Bentuk Bahu

2.3.8 Sistem Interferensi Fuzzy

2.3.8.1 Metode Penalaran Monoton

Metode penalaran monoton digunakan sebagai dasar untuk teknik implikasi fuzzy. Meskipun penalaran dengan menggunakan teknik ini sudah jarang sekali digunakan, namun terkadang masih digunakan untuk penskalaan fuzzy. Jika 2 daerah direlasikan dengan implikasi sederhana sebagai berikut:

IF x is A THEN y is B

Transfer fungsi:

$$y = f ((x , A) , B)$$

Maka sistem fuzzy dapat berjalan tanpa harus melalui komposisi dan dekomposisi fuzzy. Nilai output dapat diestimasi secara langsung dari derajat keanggotaan yang berhubungan dengan antesedennya.

2.3.8.2 Metodologi Desain Sistem Fuzzy

Untuk melakukan perancangan suatu sistem fuzzy perlu dilakukan beberapa tahapan berikut ini :

2.3.8.3 Mendefinisikan karakteristik model secara fungsional dan operasional

Pada bagian ini perlu diperhatikan karakteristik apa saja yang dimiliki oleh sistem yang ada, kemudian dirumuskan karakteristik operasi-operasi yang akan digunakan pada model fuzzy.

2.3.8.4 Melakukan dekomposisi variabel model menjadi himpunan fuzzy

Dari variabel-variabel yang telah dirumuskan, dibentuk himpunan-himpunan fuzzy yang berkaitan tanpa mengesampingkan domainnya.

2.3.8.5 Membuat aturan fuzzy

Aturan pada fuzzy menunjukkan bagaimana suatu sistem beroperasi. Cara penulisan aturan secara umum adalah : *If (X1 is A1) (Xn is An) Then Y is B* dengan (.) adalah operator (OR atau AND), X adalah scalar dan A adalah variabel linguistik. Hal yang perlu diperhatikan dalam membuat aturan adalah :

- Kelompokkan semua aturan yang memiliki solusi pada variabel yang sama.
- Urutkan aturan sehingga mudah dibaca.
- Gunakan identitas untuk memperlihatkan struktur aturan.
- Gunakan penamaan yang umum untuk mengidentifikasi variabel-variabel pada kelas yang berbeda.
- Gunakan komentar untuk mendeskripsikan tujuan dari suatu atau sekelompok aturan.
- Berikan spasi antar aturan.

- Tulis variabel dengan huruf-huruf besar-kecil, himpunan fuzzy dengan huruf besar dan elemen-elemen bahasa lainnya dengan huruf kecil.

2.3.8.6 Menentukan metode defuzzy untuk tiap-tiap variabel solusi

Pada tahap defuzzy akan dipilih suatu nilai dari suatu variabel solusi yang merupakan konsekuen dari daerah fuzzy. Metode yang paling sering digunakan adalah metode centroid, metode ini memiliki konsistensi yang tinggi, memiliki tinggi dan lebar total daerah fuzzy yang sensitif.

2.3.8.7 Menjalankan simulasi sistem

Pada tahap ini dibuat simulasi sistem secara lengkap. Simulasi harus memberikan kemudahan bagi user. Perlu dipertimbangkan juga kemungkinan pengembangannya. Untuk itu, pada program simulasi perlu disertakan editor untuk :

- Perbaiki variabel model input / output;
- Perbaiki himpunan fuzzy;
- Perbaiki aturan;
- Pilihan metode defuzzifikasi;

2.3.9 *Internet of Things*

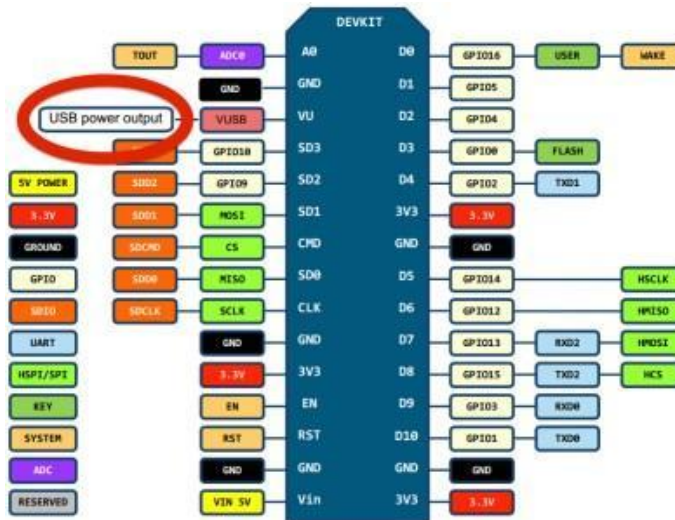
Internet of Things, atau dikenal juga dengan singkatan **IOT**, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, remote control, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif. Pada dasarnya, Internet of Things mengacu pada benda yang dapat diidentifikasi secara unik sebagai representasi virtual dalam struktur berbasis Internet. Istilah Internet of Things awalnya disarankan oleh Kevin Ashton pada tahun 1999 dan mulai terkenal melalui Auto-ID Center di MIT. Dan kini IoT menjadi salah satu tugas bagi seorang mahasiswa di sebuah perguruan tinggi.

2.3.11 NodeMCU ESP8266

NodeMCU merupakan sebuah open source platform IOT dan pengembangan kit yang menggunakan bahasa pemrograman LUA untuk membantu dalam membuat prototype produk IoT atau bisa dengan memakai sketch dengan adruino IDE. Pengembangan kit ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (*Pulse Width Modulation*), IIC, 1-Wire dan ADC (*Analog to Digital Converter*) semua dalam satu board. GPIO NodeMCU ESP8266 seperti Gambar 2.1. NodeMCU berukuran panjang 4.83cm, lebar 2.54cm, dan berat 7 gram. Board ini sudah dilengkapi dengan fitur WiFi dan Firmwarena yang bersifat opensource.

Spesifikasi yang dimiliki oleh NodeMCU sebagai berikut :

1. Board ini berbasis ESP8266 serial WiFi SoC (Single on Chip) dengan onboard USB to TTL. Wireless yang digunakan adalah IEEE 802.11b/g/n.
2. 2 tantalum capacitor 100 micro farad dan 10 micro farad.
3. 3.3v LDO regulator.
4. Blue led sebagai indikator.
5. Cp2102 usb to UART bridge.
6. Tombol reset, port usb, dan tombol flash.
7. Terdapat 9 GPIO yang di dalamnya ada 3 pin PWM, 1 x ADC Channel, dan pin RX TX
8. 3 pin ground.
9. S3 dan S2 sebagai pin GPIO 4
10. S1 MOSI (Master Output Slave Input) yaitu jalur data dari master dan masuk ke dalam slave, sc cmd/sc.
11. S0 MISO (Master Input Slave Input) yaitu jalur data keluar dari slave dan masuk ke dalam master.
12. SK yang merupakan SCLK dari master ke slave yang berfungsi sebagai clock.
13. Pin Vin sebagai masukan tegangan.
14. Built in 32-bit MCU.



Gambar 2.20, GPIO NodeMCU ESP8266 v3

(Sumber : <https://www.meccanismocomplexo.org/en/iot-internet-of-things/>.)

1. RST : berfungsi mereset modul
2. ADC: Analog Digital Converter. Rentang tegangan masukan 0-1v, dengan skup nilai digital 0-1024
3. EN: Chip Enable, Active High
4. IO16 :GPIO16, dapat digunakan untuk membangunkan chipset dari mode deep sleep
5. IO14 : GPIO14; HSPI_CLK
6. IO12 : GPIO12: HSPI_MISO
7. IO13: GPIO13; HSPI_MOSI; UART0_CTS 5
8. VCC: Catu daya 3.3V (VDD)
9. CS0 :Chip selection
10. MISO : Slave output, Main input
11. IO9 : GPIO9
12. IO10 GBIO10
13. MOSI: Main output slave input
14. SCLK: Clock
15. GND: Ground
16. IO15: GPIO15; MTDO; HSPICS; UART0_RTS

17. IO2 : GPIO2;UART1_TXD
18. IO0 : GPIO0
19. IO4 : GPIO4
20. IO5 : GPIO5
21. RXD : UART0_RXD; GPIO3
22. TXD : UART0_TXD; GPIO

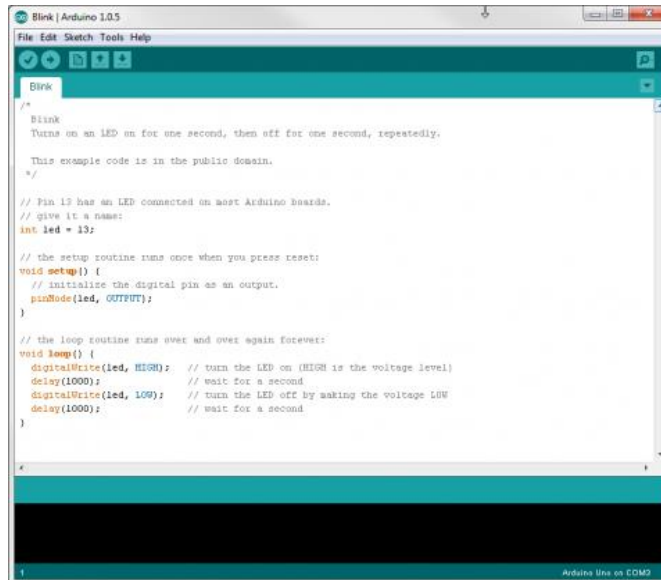
2.4 Perangkat Lunak Yang Digunakan

Pengertian perangkat lunak atau biasa disebut software adalah sekumpulan data elektronik yang sengaja disimpan dan diatur oleh komputer berupa program ataupun instruksi yang akan menjalankan sebuah perintah. Perangkat lunak atau software disebut juga sebagai penerjemah perintah-perintah yang dijalankan oleh user untuk diteruskan dan diproses oleh perangkat keras (hardware). Dengan adanya perangkat lunak inilah sebuah sistem mampu menjalankan perintah.

2.4.1 Software Mikrokontroller Arduino Uno

Software arduino yang digunakan adalah *driver* dan IDE, walaupun masih ada beberapa *software* lain yang sangat berguna selama pengembangan arduino. *Integrated Development Environment (IDE)*, suatu program khusus untuk suatu komputer agar dapat membuat suatu rancangan atau *sketsa* program untuk papan *Arduino*. IDE *arduino* merupakan *software* yang sangat canggih ditulis dengan menggunakan *java*. IDE arduino terdiri dari

2.4.1.1 Program Arduino IDE

The image shows a screenshot of the Arduino IDE interface. The window title is "Blink | Arduino 1.0.5". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The main text area contains the following code:

```
Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 * This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

The status bar at the bottom right indicates "Arduino Uno on COM3".

Gambar 2.21 Tampilan Program *Arduino Uno*

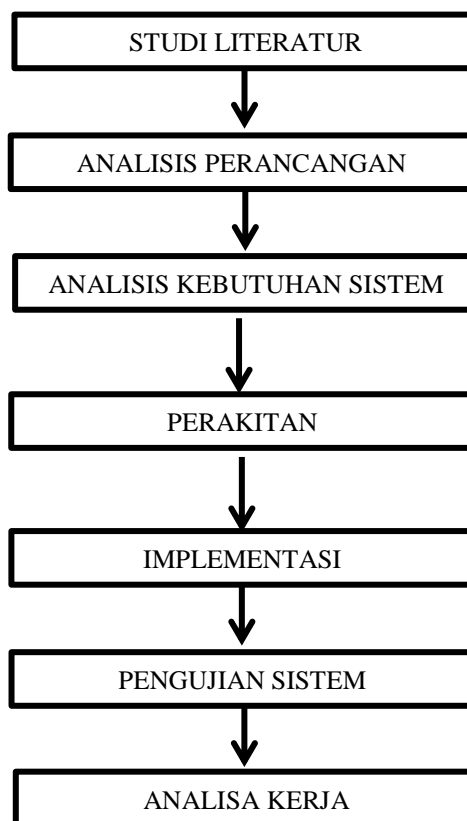
(Sumber : <https://www.meccanismocomplessio.org/en/iot-internet-of-things/>.)

Kode Program *Arduino* biasa disebut *sketch* dan dibuat menggunakan bahasa pemrograman C. Program atau *sketch* yang sudah selesai ditulis di *Arduino IDE* bisa langsung *dicompile* dan *diupload* ke *Arduino Board*. Secara sederhana, *sketch* dalam *Arduino* dikelompokkan menjadi 3 blok (lihat gambar di atas):

1. Header
2. Setup
3. Loop

BAB III METODOLOGI PENELITIAN

Bab ini akan menjelaskan langkah-langkah penelitian yang akan dilakukan dalam Implementasi Fuzzy Logic Control Pada Sistem Peringatan Dini Kebakaran Berbasis Iot. Alur penelitian yang digunakan seperti pada gambar 3.1.



Gambar 3.1. Alur Penelitian

3.1 Studi Literatur

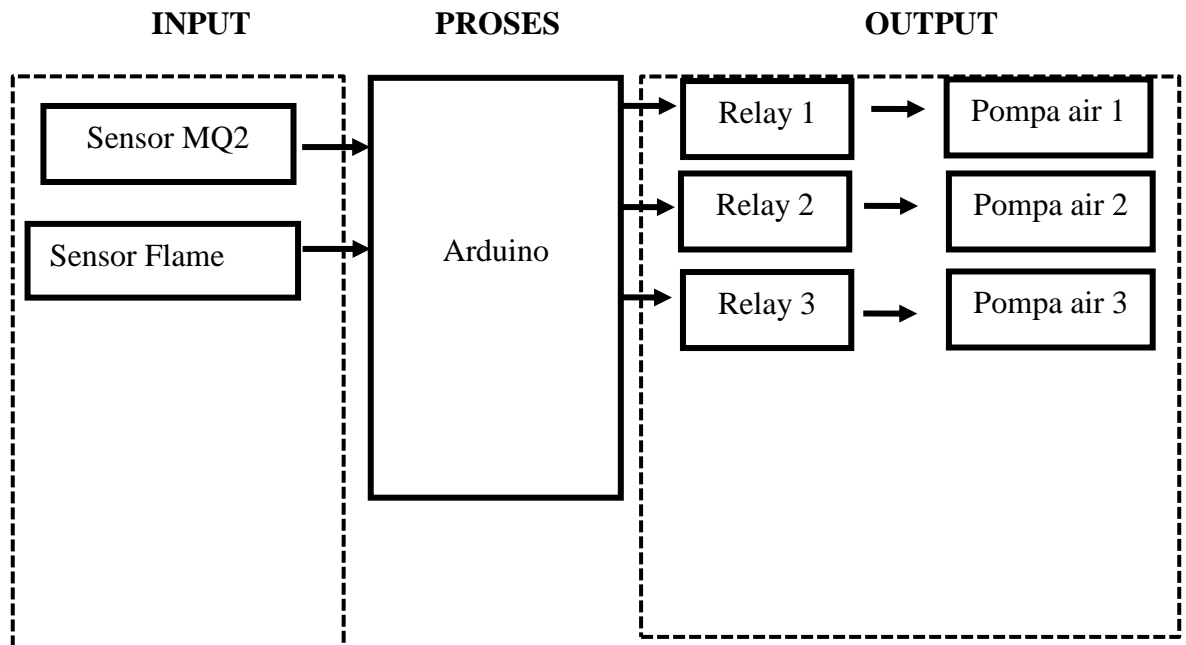
Pada metode ini penulis mencari bahan penulisan tugas akhir yang diperoleh dari buku, jurnal dan website yang terkait dengan pembuatan “Implementasi Fuzzy Logic Control Pada Sistem Peringatan Dini Kebakaran Berbasis Iot”.

3.2 Analisa Perancangan Sistem

Perancangan sistem merupakan suatu hal yang dilakukan untuk mempermudah proses pembuatan alat. Konsep “Implementasi Fuzzy Logic Control Pada Sistem Peringatan Dini Kebakaran Berbasis IOT” digambarkan pada diagram blok dapat

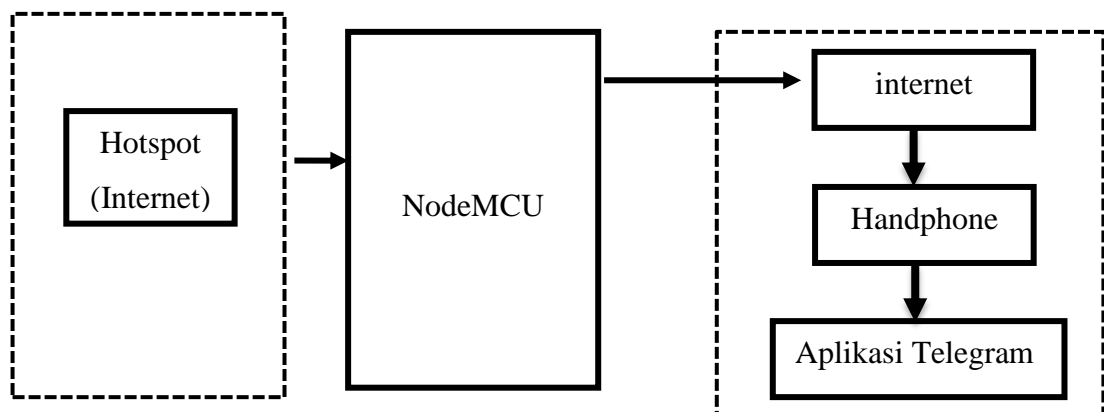
dilihat pada gambar 3.2 Blok diagram menjelaskan gambaran umum mengenai cara kerja dari sistem yang akan dibuat.

Blok Diagram Sistem Kerja Peringatan Dini Kebakaran



Gambar 3.2. Blok Diagram Sistem

Blok Diagram Sistem Kerja Pengiriman Via Telegram



Gambar 3.3. Blok Diagram Sistem Pengiriman Telegram

Sistem kerja dari alat ini yaitu, Jika sensor *MQ2* mendeteksi adanya asap dan sensor *flame* mendeteksi adanya api maka mengirimkan sinyal ke *Arduino*, *Arduino* menghidupkan *relay* yang tersambung ke pompa air dan mengirimkan perintah ke *NodeMCU* untuk mengirimkan informasi melalui *wifi* ke aplikasi

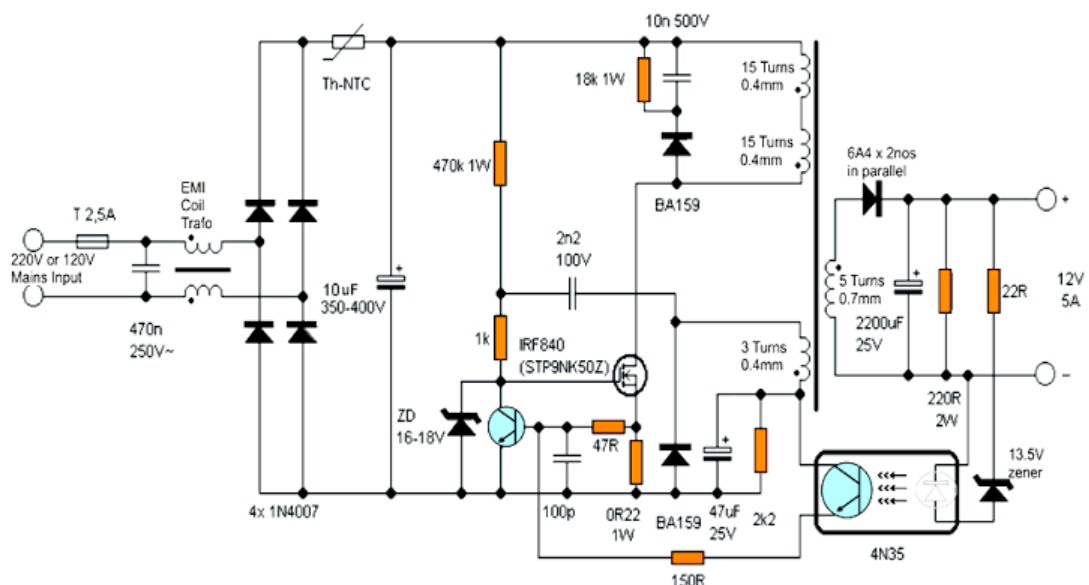
telegram yang telah terhubung dengan *nodemcu* hasil pembacaan sensor akan ditampilkan pada *chatting* aplikasi *Telegram*.

3.2.1 Perancangan Perangkat Keras

Perancangan menjadi bagian yang sangat penting dilakukan dalam pembuatan suatu alat karena dengan merancang terlebih dahulu dengan komponen yang tepat akan mengurangi berlebihnya pembelian komponen yang kurang dibutuhkan dan diharapkan kerja alat dapat optimal sesuai dengan yang diinginkan. Untuk menghindari kerusakan komponen perlu dipahami juga akan karakteristik dari komponen-komponen tersebut.

3.2.1.1 Rangkaian *Power Supply*

Rangkaian power supply digunakan untuk merubah tegangan AC 220V menjadi DC 12V dalam pembuat power suplay 12 volt dan 5 volt peneliti menggunakan IC LM7812 dan LM7805 menyalurkan sumber tegangan ke semua komponen elektronika yang ada pada suatu rangkaian agar rangkaian tersebut dapat bekerja seperti pada gambar 3.3.



Gambar 3.3 Rangkaian *Power Supply*

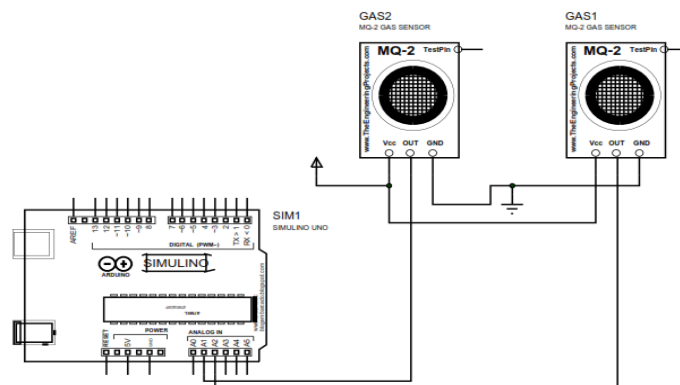
(Sumber : <https://www.homemade-circuits.com>)

Penjelasan:

- EMI Coil Travo kaki input 120 - 220V AC dan 1 kaki output 12V
- 4x 1N4007 adalah dioda 6A05 yang dirangkai bridge dan Beberapa dioda BA159 yang dirangkai paralel
- Kapasitor dengan besaran kapasistansi 470nf-250v, 10mf-400v, 2200mf-25v yang berfungsi sebagai (penyaring) tegangan.
- Terdapat beberapa resistor dengan besaran 470k, 1k, 18k, 22R, dan 150R yang berfungsi sebagai (tahanan) tegangan
- ZD 16-18v dan Zener 13,5v adalah dioda Zener yang berfungsi untuk menstabilkan tegangan pada rangkaian
- Ada 2 Transistor yang berfungsi sebagai penguat sekaligus sebagai sirkuit pemutus dan penyambung (switching) pada rangkaian
- STP9NK50Z adalah IC pada rangkaian

3.2.1.2 Rancangan Rangkaian Sensor MQ2

Rangkaian *MQ2* digunakan sebagai *inputan* dalam mendeteksi adanya asap yang akan diproses oleh *Arduino* sehingga akan menghasilkan outputan untuk mengaktifkan pompa air dan pengiriman pesan via telegram sebagai tanda jika terjadinya kebakaran. Gambar rangkaian *MQ2*, layout dan tata letak dapat dilihat seperti pada gambar 3.4.



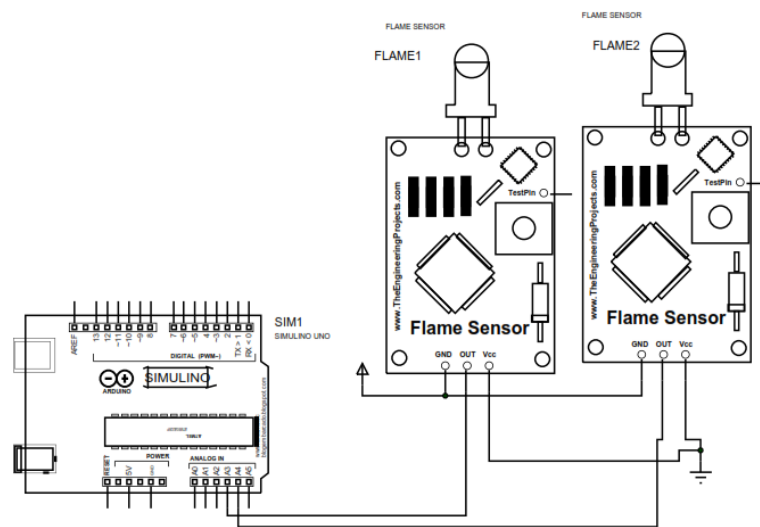
Gambar 3.4 Rangkaian Sensor MQ2

Pada rangkaian *MQ2* hanya beberapa kaki yang dihubungkan ke pin digital *Arduino* agar hasil proses pada *Arduino* dapat mengaktifkan *Relay* dan mengirimkan perintah ke *NodeMCU* untuk mengirimkan pesan via telegram. Penjelasan penggunaan PIN *Arduino* dan *Sensor MQ2* ditampilkan sebagai berikut:

- *Sensor MQ2* mendapat tegangan input sebesar +5.0V dari sumber tegangan
- Kaki GND mendapat Ground dari sumber tegangan
- Kaki Data Out2 mendapat pin D3 dari *Arduino*

3.2.1.3 Rancangan Rangkaian Sensor *Flame*

Rangkaian Sensor flame digunakan sebagai *inputan* dalam mendeteksi adanya api atau kebakaran pada yang telah diolah oleh *Arduino* sehingga akan menghasilkan outputan berupa pengaktifan *Relay* dan pengiriman informasi via telegram melalui *NodeMCU*. Gambar rangkaian Sensor *Flame* dan tata letak dapat dilihat seperti pada gambar 3.5.



Gambar 3.5. Perancangan Rangkaian Sensor flame

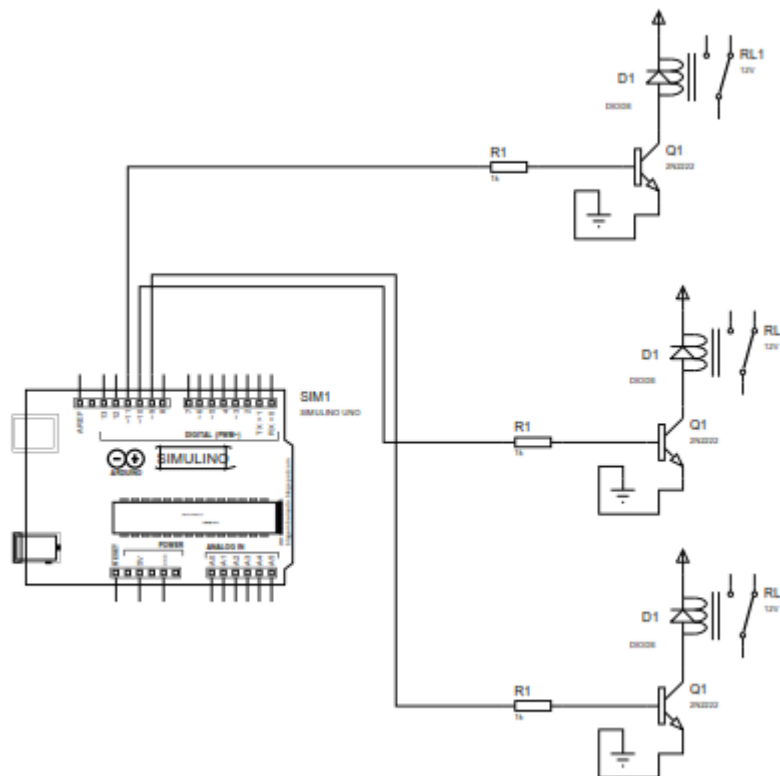
Pada rangkaian Sensor *Flame* hanya beberapa kaki yang dihubungkan ke pin analog *Arduino* agar hasil proses pada *Arduino* dapat mengaktifkan *relay* dan pengiriman informasi via telegram. Penjelasan penggunaan PIN *Arduino* dan sensor *Flame* ditampilkan sebagai berikut:

- *Sensor Flame* mendapat tegangan input sebesar +5.0V dari sumber tegangan

- Kaki GND mendapat Ground dari sumber tegangan
- Kaki Data Out1 mendapat pin A0 dari *Arduino*

2.2.3.4 Rangkaian *Relay*

Rangkaian *relay* digunakan sebagai *output* yang akan diproses oleh *Arduino* sehingga akan menyalakan pompa air untuk memadamkan api yang ada diruangan. Gambar rangkaian *relay output* dan tata letak dapat dilihat seperti pada gambar 3.5.



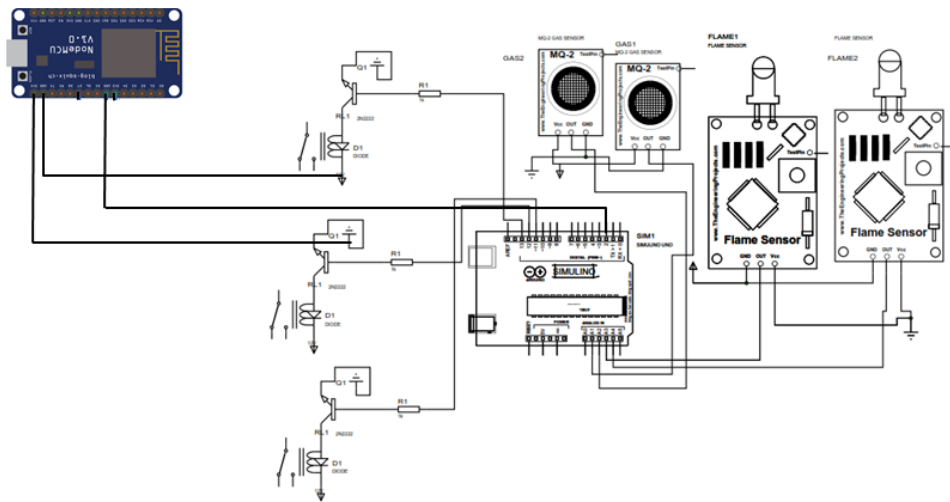
Gambar 3.5 Rangkaian *Relay*

Pada rangkaian *Relay* hanya beberapa kaki yang dihubungkan ke pin Digital *Arduino* agar hasil proses pada *Arduino* dapat menghidupkan *Relay*. Penjelasan penggunaan PIN *Arduino* dan *Relay* ialah Pin D13 Mikrokontroler (*Arduino Uno*) mendapat resistor dengan tahanan sebesar 100Ω, Resistor mendapat kaki basis dari transistor BC548, Kaki kolektor transistor BC548 terhubung dengan kaki *Coil Relay* dan kaki anoda dari dioda 1N4001, Kaki katoda dari dioda 1N4001

mendapat tegangan masukan sebesar +12V dan kaki *Coil Relay*, Kaki NO *Relay* terhubung ke NO kontaktor, Kaki COM *Relay* terhubung ke *coil* kontaktor.

3.2.1.4 Rangkaian Keseluruhan

Rangkaian keseluruhan merupakan tahap terakhir dari perancangan yang telah dilakukan. Dalam tahap ini seluruh komponen dipasang sesuai dengan sistem yang telah dibuat, Adapun rangkaian keseluruhan dapat dilihat pada gambar 3.7



Gambar 3.7 Rangkaian Keseluruhan

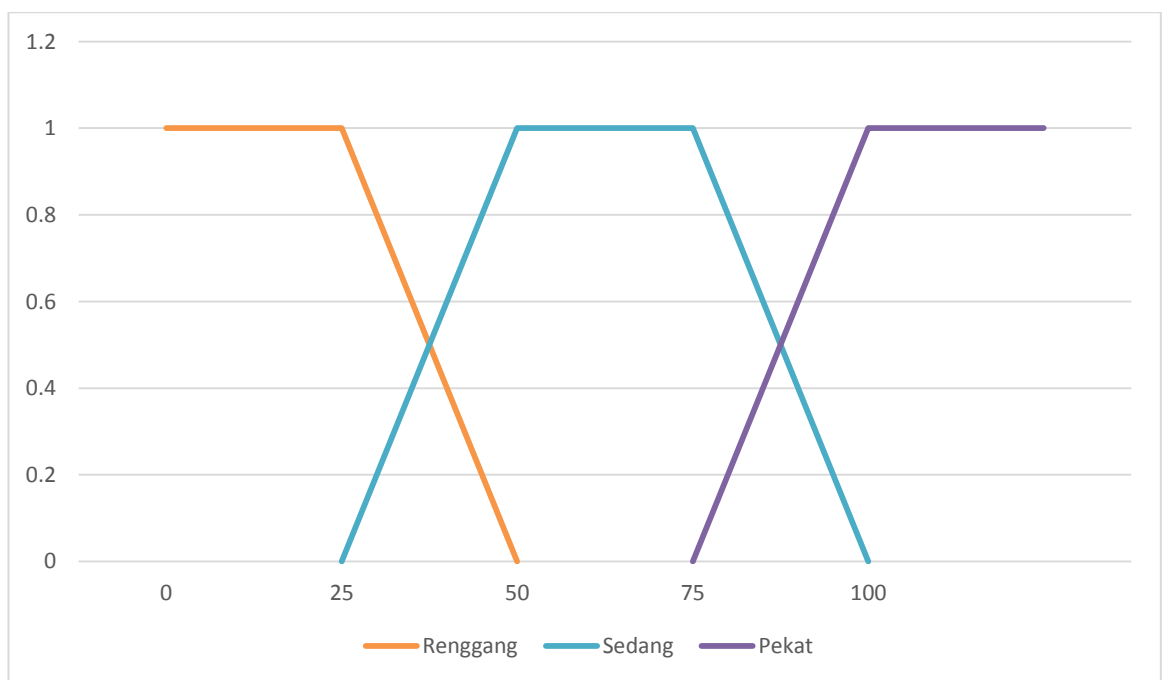
3.2.2 Perancangan Perangkat Lunak

Perancangan sistem dengan menggunakan metode fuzzy membutuhkan beberapa proses sehingga terbentuknya suatu keputusan output dari sistem sesuai dengan perhitungan fuzzy. Proses tersebut antara lain fuzzyfikasi, implikasi, komposisi, dan defuzzyfikasi.

Sistem inferensi yang dipakai pada alat ini menggunakan metode Mamdani. Secara garis besar metode Mamdani dibagi menjadi tiga tahapan yaitu Fuzzyfikasi, Inferensi, dan Defuzzyfikasi. Fuzzyfikasi yaitu suatu proses untuk mengubah suatu masukan dari bentuk tegas (crisp) menjadi fuzzy (variabel linguistik) yang biasa disajikan dalam bentuk himpunan-himpunan fuzzy dengan

suatu fungsi keanggotaannya masing-masing. Inferensi adalah sebuah kerangka kerja perhitungan yang berdasarkan pada konsep teori himpunan fuzzy, aturan fuzzy if –then, dan pemikiran fuzzy. Defuzzyfikasi adalah langkah terakhir dalam suatu sistem logika fuzzy dimana tujuannya adalah mengkonversi setiap hasil dari inference engine yang diekspresikan dalam bentuk fuzzy set kesatuan bilangan real.

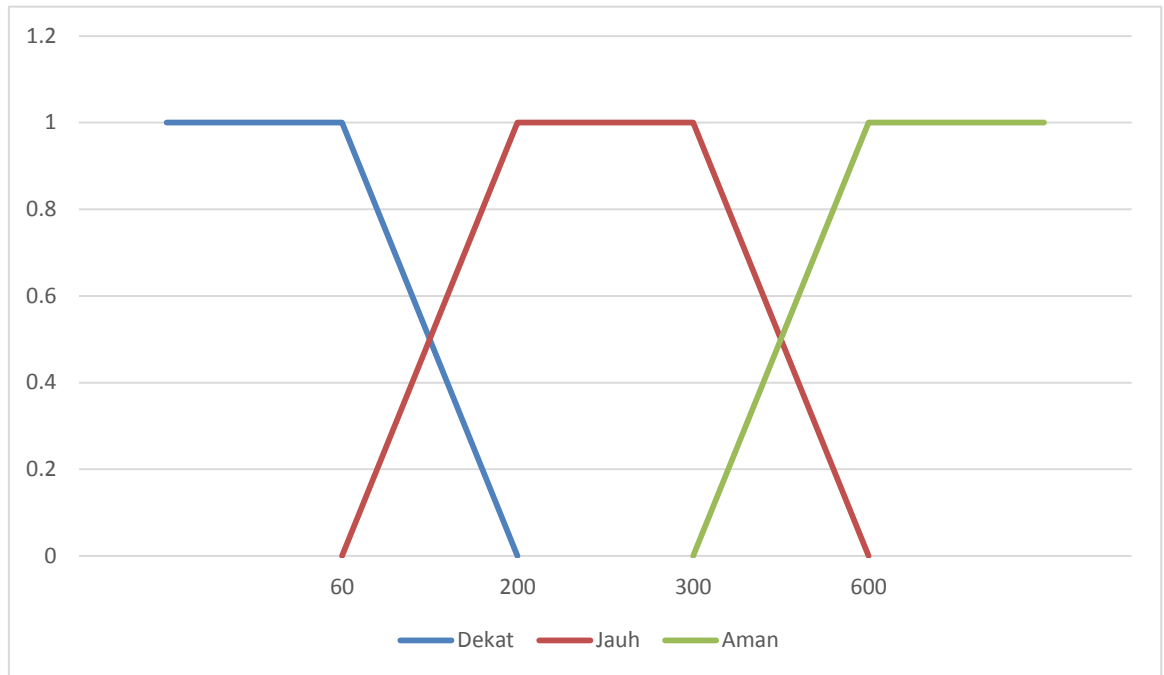
Sistem yang dibangun ini menggunakan 2 jenis input yaitu data kadar asap dan data api. Pada data kadar asap terdapat 3 keanggotaan yaitu Renggang, Sedang, dan Pekat. Perancangan himpunan fuzzy kadar asap dapat dilihat pada gambar 3.9 berikut.



Gambar 3.8 Diagram keanggotaan asap

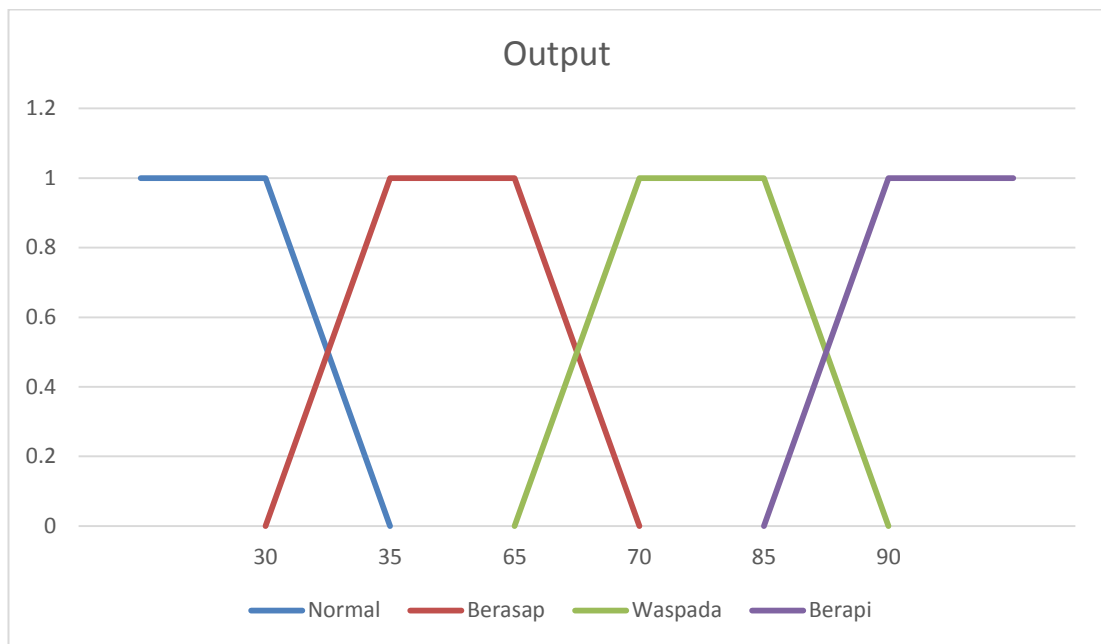
Dari gambar diatas, akan dilakukan pencarian derajat keanggotaan (crisp input) yaitu 30 ppm dan 87 ppm. Sehingga crisp input 30 ppm berada pada himpunan fuzzy Sedang, dan crisp input 87 ppm berada pada himpunan fuzzy Pekat.

Untuk data api digolongkan menjadi 3 kriteria yaitu Dekat, Jauh, Aman. Perancangan himpunan fuzzy api dapat dilihat gambar berikut.



Gambar 3.9 Diagram keanggotaan api

Pada diagram keanggotaan api akan dilakukan pencarian derajat keanggotaan (crisp input) yaitu 70, 315. Untuk crisp input 70 berada pada himpunan fuzzy antara Dekat dan jauh, sedangkan crisp input 315 berada pada himpunan fuzzy antara Jauh dan Aman. Sedangkan untuk keluaran system berupa tampilan notifikasi pada *Telegram* dengan menggunakan besara fuzzy yaitu Normal (N), Berasap (B), Waspada (W), Berapi (BA). Fungsi keanggotaan output dapat dilihat pada gambar. 3.10



Gambar 3.10 Diagram Output fuzzy

Setelah proses Fuzzyfikasi selesai, dilanjutkan dengan proses Inferensi. Inferensi adalah penggabungan banyak aturan berdasarkan data yang tersedia. Dari uraian diatas telah terbentuk sebanyak 6 himpunan fuzzy sebagai input, yaitu renggang, sedang, pekat, dekat, jauh, dan Aman. Ditambah dengan 4 himpunan kondisi sebagai output, yaitu normal, berasap, waspada, dan berapi. Pada aturan fuzzy ini akan memberikan aturan-aturan dalam fuzzy sistem yang akan dibuat dengan menggunakan perintah “IF” dan “AND” dan menghasikan perintah “THEN”. Aturan dasar fuzzy yang digunakan untuk menentukan kondisi kebakaran didalam

Flame	MQ2	Renggang	Sedang	Pekat
Dekat		W	BA	BA
Jauh		N	B	W
Tidak Ada Api		N	B	B

ruangan dapat dilihat pada Tabel 3.1.

Tabel 3.1 Tabel Aturan Dasar Fuzzy

R1. IF gas = Renggang AND api = Dekat THEN tampilan IS W (Waspada)

R2. IF gas = Renggang AND api = Jauh THEN tampilan IS N (Normal)

R3. IF gas = Renggang AND api = Aman THEN tampilan IS N (Normal)

R4. IF gas = Sedang AND api = Dekat THEN tampilan IS BA (Berapi)

R5. IF gas = Sedang AND api = Jauh THEN tampilan IS B (Berasap)

R6. IF gas = Sedang AND api = Aman THEN tampilan IS B (Berasap)

R7. IF gas = Pekat AND api = Dekat THEN tampilan IS BA (Berapi)

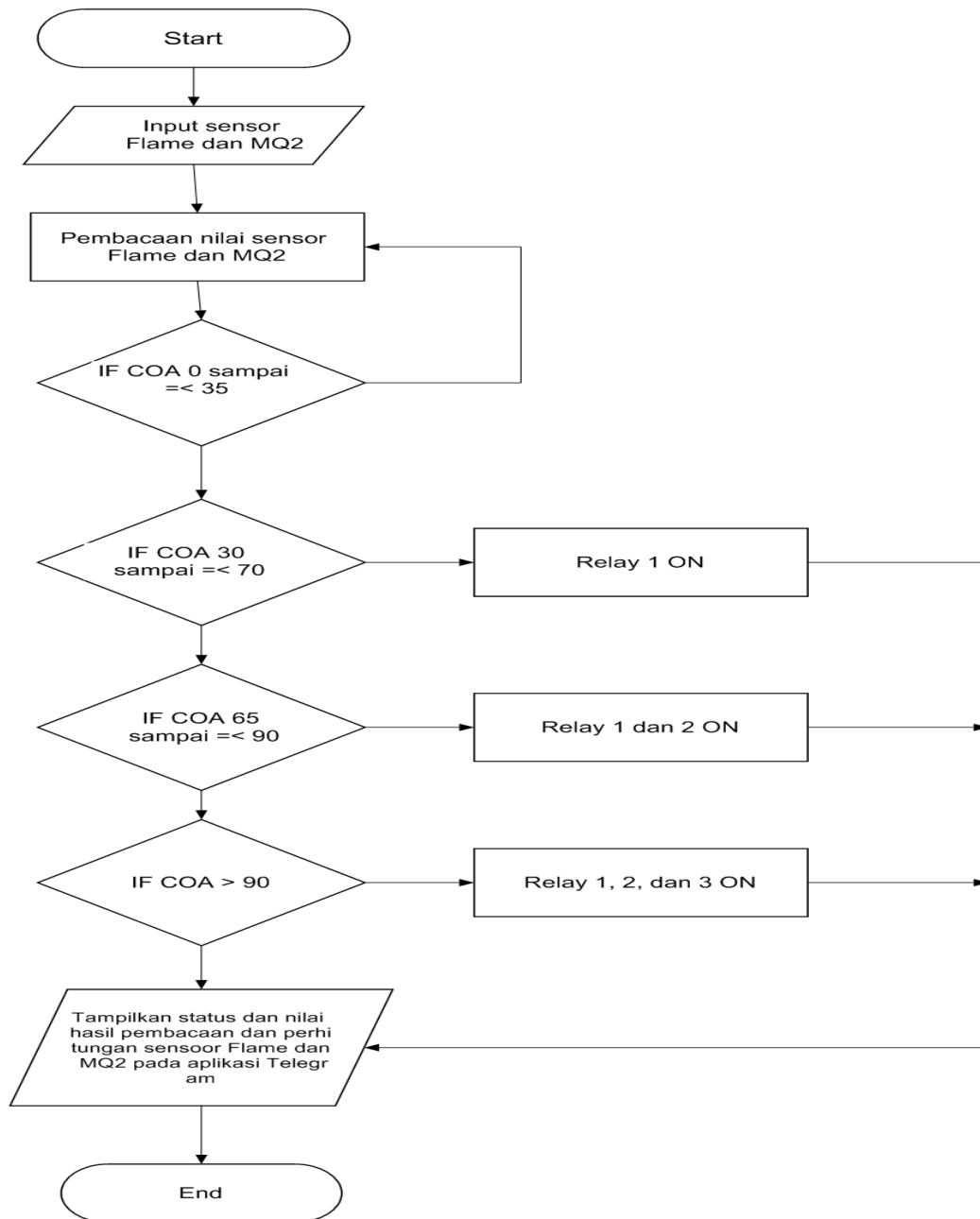
R8. IF gas = Pekat AND api = Jauh THEN tampilan IS W (Waspada)

R9. IF gas = Pekat AND api = Aman THEN tampilan IS B (Berasap)

Berdasarkan 9 aturan fuzzy tersebut, akan ditentukan nilai α untuk masing-masing aturan. α adalah nilai keanggotaan anteseden dari setiap aturan. Berikut ini adalah langkah-langkah untuk mengkonversi sembilan aturan fuzzy tersebut sehingga diperoleh nilai α dari setiap aturan. Aturan yang digunakan adalah aturan MIN pada fungsi implikasinya. Setelah diketahui nilai α pada masing masing aturan. Menurut metode MIN-MAX selanjutnya tiap variabel kondisi akan mengevaluasi masing-masing rule yang terkait dengan kondisi tersebut untuk dicari nilai terbesarnya (MAX), seperti:

- Normal = max (rule 2, dan rule 3)
- Berasap = max (rule 5, (max (rule 6, rule 9))
- Waspada = max (rule 1, rule 8)
- Berapi = max (rule 4, rule 7)

Pada tahap akhir dari pemrograman sistem ini, yaitu pembuatan program *defuzzyfikasi* untuk mengubah dari variable fuzzy hasil keputusan menggunakan basis aturan menjadi nilai digital (crisp). Program *defuzzyfikasi* mengikuti metode *Central Of Area* (COA). Tahap terakhir yaitu *defuzzyfikasi* dimana perubahan nilai output *logica fuzzy* menjadi nilai output sesungguhnya. Metode *defuzzyfikasi* yang digunakan adalah *centroid method* disebut juga sebagai *Center Area* atau *Center Of Gravity*.



Gambar 3.11 *Flowchart* Sistem

Di bawah ini merupakan penjelasan dari *flowchart* sistem pada gambar 3.11. Star adalah proses penyalaan alat dan Inisialisasi port adalah proses membaca port pada arduino. Jika sensor *MQ2* mendeteksi adanya asap dan sensor *Flame* mendeteksi adanya api dengan nilai yang telah ditentukan maka arduino

memproses perintah dan akan mengirimkan sinyal ke *relay* 1,2,dan 3(on/off) dan arduino memberikan perintah ke *NodeMCU* untuk mengirimkan informasi melalui *wifi* ke aplikasi telegram untuk menampilkan hasil nilai dan status dari pembacaan sensor *Flame* dan *MQ2*.

3.3 Analisa Kebutuhan

Tahapan selanjutnya setelah membuat rancangan perangkat keras dan perangkat lunak yaitu membuat analisa kebutuhan sistem. Analisa kebutuhan sistem dilakukan untuk mengetahui alat dan komponen serta perangkat lunak apa saja yang akan digunakan untuk mengimplementasikan sistem.

3.3.1 Alat

Sebelum membuat “Implementasi Fuzzy Logic Control Pada Sistem Peringatan Dini Kebakaran Berbasis Iot” ada beberapa peralatan yang harus disiapkan. Daftar peralatan yang digunakan dalam penelitian ini akan dituliskan pada Tabel 3.2.

Tabel 3.2 Alat Yang Dibutuhkan

No	Nama Alat	Spesifikasi	Fungsi	Jumlah
1	Komputer/ laptop	Window 7-10 32/64bit	Untuk membuat sebuah aplikasi yang akan di pakai di perangkat keras dan perangkat lunak	1 unit
2	Multitester	Analog/Digital	digunakan untuk mengukur tegangan (ACV-DCV), dan kuat arus (mA- μ A)	1 buah
3	Obeng	Obeng + dan -	Untuk merangkai alat	1 buah
4	Solder	-	Untuk menempelkan timah ke komponen	1 buah
5	Bor pcb	-	Untuk membuat lobang baut atau komponen	1 buah
6	Tang Potong	-	Untuk memotong kabel dan kaki komponen	1 buah
7	Kit Arduino	-	Komponen Komplit arduino UNO	1 buah
8	Lem Tembak	-	-	1 unit
9	Kabel Jumper	-	-	-

3.3.2 Komponen

Sebelum membuat “Implementasi Fuzzy Logic Control Pada Sistem Peringatan Dini Kebakaran Berbasis Iot” ada beberapa koomponen yang harus disiapkan. Daftar komponen yang digunakan dalam penelitian ini akan dituliskan pada Tabel 3.3.

Tabel 3.3 Komponen Yang Dibutuhkan

No	Nama Alat	Sepesifikasi	Fungsi	Jumlah
1	Nodemcu	ESP8266	Sebagai penghubung ke aplikasi telegram	1
2	Sensor MQ2	-	Digunakan sebagai inputan untuk membaca adanya asap	1
3	Sensor flame	-	Digunakan sebagai pendeteksi api	1
4	Arduino	R3	Untuk pemroses perintah yang akan dijalankan	1
5	Relay	3 slot	Digunakan sebagai penyambung dan pemutus arus listrik	1

3.3.3 Software

Sebelum membuat “Implementasi Fuzzy Logic Control Pada Sistem Peringatan Dini Kebakaran Berbasis Iot ada beberapa *Software* yang harus disiapkan. Daftar *Software* yang digunakan dalam penelitian ini akan dituliskan pada Tabel 3.4.

Tabel 3.4. Daftar Software Yang Digunakan

No	Nama	Spesifikasi	Fungsi
1	IDE Arduino	Arduino 1.6.3	Membuat program yang akan di-download perangkat arduino
2	Proteus	7.1 Profesional	Merancang rangkaian yang akan digunakan untuk membuat alat

3.4 Implementasi

Setelah mengumpulkan alat dan bahan, langkah selanjutnya adalah melakukan implementasi rancangan alat yang telah dibuat. Pada tahap ini hasil rancangan yang telah dibuat akan diimplementasikan untuk menjadi sistem yang sesungguhnya. Implementasi pada penelitian ini dibagi menjadi dua bagian, yaitu:

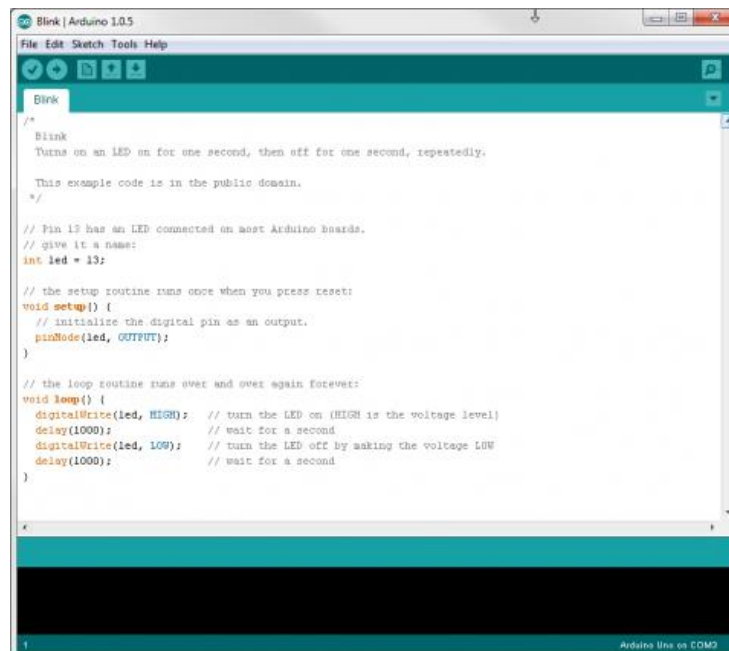
Implementasi perangkat keras dan Implementasi perangkat lunak. Implementasi perangkat keras merupakan tahap terakhir dari perancangan sistem yang dilakukan dalam tahap ini seluruh komponen dipasang sesuai dengan sistem yang telah dibuat.

3.4.1 Implementasi Perangkat Keras

Realisasi perangkat keras merupakan tahap terakhir dari perancangan yang telah dilakukan. Dalam tahap ini seluruh komponen dipasang sesuai dengan sistem yang telah dibuat.

3.4.2 Implementasi Perangkat Lunak

Penerapan perangkat lunak merupakan suatu tahap dimana program yang telah dirancang akan disimpan kedalam modul *mikrokontroler* melalui *downloader* dan menggunakan *software* tertentu sesuai dengan bahasa pemrograman yang akan digunakan. Disini peneliti menggunakan bahasa C dan menggunakan *software Arduino*. Pada *Software Arduino* program ditulis kemudian *dcompile*, tujuannya adalah untuk mengetahui apakah program yang dibuat sudah benar atau belum. Langkah terakhir yaitu meng-*upload* program kedalam modul *mikrokontroler*.

The image shows a screenshot of the Arduino IDE interface. The title bar reads "Blink | Arduino 1.0.5". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". The main text area contains the following C++ code for a Blink sketch:

```
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

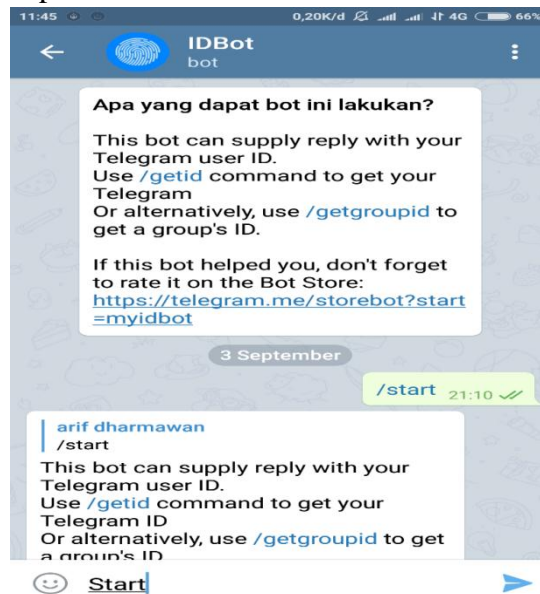
// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

Gambar 3.12 Prangkat Lunak Arduino

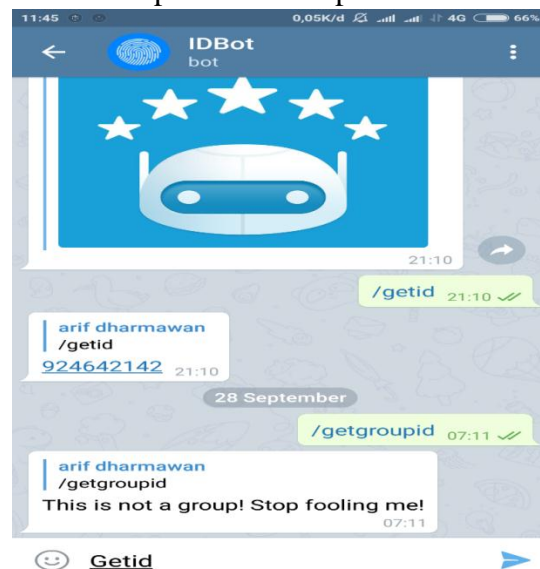
3.4.3 Pembuatan Bot Via Telegram

- Pertama *download* aplikasi telegram di google *playstore*
- Selanjutnya bukan aplikasi *Telegram* dan pada kolom search cari IDBot dan masuk *chat* obrolan
- Selanjutnya mulai pilih */start*



Gambar 3.13 Pilih Start

- Lalu pilih */getid* dan IDBot pun telah didapatkan



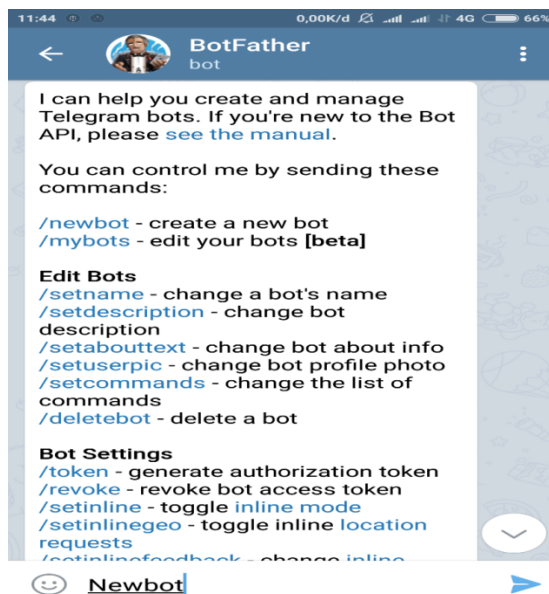
Gambar 3.14 Pilih Getid

- Selanjutnya untuk mendapatkan alamat *token BotFather* pada kolom *search* cari *BotFather* dan masuk chat obrolan
- Selanjutnya pilih */start*



Gambar 3.15 Pilih Start

- Lalu pilih */newbot* untuk membuat bot baru



Gambar 3.16 Pilih New Bot

- Lalu kita diminta untuk memasukkan nama bot, maka ketikkan nama bot yang kita inginkan
- Selanjutnya kita diminta memasukkan ulang nama bot yang telah kita buat namun kita tambahkan tanda “_bot “



Gambar 3.17 Masukan Alamat Id Bot

- Setelah itu alamat *token* bot pun kita dapatkan

3.5 Pengujian Sistem

Setelah perancangan *hardware* dan *software* selesai, maka yang dilakukan adalah *running* program, pengujian tiap-tiap rangkaian apakah sudah sesuai dengan yang diinginkan atau belum. Pengujian dilakukan pada bagian-bagian seperti pengujian respon, jangkauan sistem, catu daya dan rangkaian keseluruhan pada sistem ini.

3.5.1 Pengujian Rangkaian Sensor MQ2

Pengujian Sensor *MQ2* bertujuan untuk mengetahui ketika adanya asap pada rumah atau kantor apakah sensor dapat dengan baik dalam mendeteksinya dan peneliti melakukan ujicoba berapa lama delay yang dibutuhkan sensor *MQ2* dalam membaca adanya asap dan jarak *MQ2* dalam mendeteksi asap. Dalam mengukur respon, peneliti menggunakan timer sedangkan dalam mengukur jarak peneliti menggunakan meteran. Rangkaian alat sesuai dengan gambar rangkaian, selanjutnya *upload* Potongan *script* program sensor *MQ2* ke dalam *Arduino* dapat dilihat di bawah:

```

void gasMQ2(){
  smoke=MQGetGasPercentage(MQRead(MQ_PIN)/Ro,GAS_SMOKE);
  //=====Renggang=====//
  if(smoke==0){
    miu_renggang=1;
    renggang=true;
    Hkadar="Renggang";
  }
  else if(smoke<=25){
    miu_renggang=1;
    renggang=true;
    Hkadar="Renggang";
  }
  else if(smoke>37.5 && smoke<50){
    miu_renggang=(50 - smoke)/(50 - 25);
    renggang=true;
    Hkadar="Renggang";
  }
  else if(smoke>=50){
    miu_renggang=0;
    renggang=false;
  }
}

```

3.5.2 Pengujian Rangkaian Sensor Flame

Pengujian Sensor Api bertujuan untuk mengetahui ketika apakah sensor dapat dengan baik dalam membaca adanya api, peneliti akan melakukan uji coba yaitu respon dan jarak sensor dalam mendeteksi adanya api. Dalam mengukur respon peneliti menggunakan timer sedangkan dalam mengukur jarak peneliti menggunakan meteran. Rangkai alat sesuai dengan gambar rangkaian, selanjutnya *upload* Potongan *script* program sensor *Flame* ke dalam *Arduino* dapat dilihat di bawah:

```

void SensorApi(){
  apiSens=analogRead(flame);
  //=====Dekat=====//
  if(apiSens>=0 && apiSens<=350){
    miu_dekat=1;
    dekat=true;
    Hpanas="Dekat";
  }
  else if(apiSens>400 && apiSens<450){
    miu_dekat=(450 - apiSens)/(450 - 350);
    dekat=true;
    Hpanas="Dekat";
  }
  else if(apiSens>=450){
    miu_dekat=0;
    dekat=false;
  }
}

```

3.5.3 Pengujian fuzzyfikasi

Pada proses *fuzzyfikasi* dilakukan perhitungan derajat keanggotaan dari nilai-nilai input yang ada yaitu sensor asap *MQ2* dan sensor *Flame* yang akan diproses ke dalam aturan-aturan fuzzy yang dirancang. Potongan *script* program *Fuzzyfikasi* dapat dilihat di bawah:

```

void defuzzy(){
  nilaiMax();
  Dnormal=(5+10+15+20+25+30+35);    //total 7
  Dberasap=(30+35+40+45+50+55+60+65+70);//total 9
  Dwaspada=(65+70+75+80+85+90);    //total 6
  Dberapi=(85+90+95+100);          //total 4
  hasil_atas=((Dnormal)*(MAX_normal))+((Dberasap)*(MAX_berasap))+((Dwaspada)*(MAX_waspada))+((Dberapi)*(MAX_berapi));
}

```

```

hasil_bawah=((MAX_normal)*7)+((MAX_berasap)*9)+((MAX_waspada)*6)+((
MAX_berapi)*4);
COA=((hasil_atas)/(hasil_bawah));
if(COA>=0 && COA<=25){
    keadaan="normal";
}
else if(COA>30 && COA<35){
    keadaan="normal";
}
else if(COA==30){
    keadaan="antara normal dan berasap";
}
else if(COA>25 && COA<30){
    keadaan="berasap";
}
else if(COA>=35 && COA<=60){
    keadaan="berasap";
}

```

3.5.4 Pengujian Rangkaian Telegram

Pengujian telegram bertujuan untuk mengetahui ketika sensor *MQ2* dan sensor *Flame* mendeteksi adanya api apakah dapat mengirimkan informasi pembacaan nilai sensor via telegram. Serta mengukur waktu respon program menggunakan timer. Rangkai alat sesuai dengan gambar rangkaian selanjutnya *upload* Potongan *script* program sensor Telegram ke dalam *NodeMCU* sertakan *IDBot* dan alamat *token* yang telah didapatkan, dapat dilihat di bawah:

```

void loop() {
    while(node.available(>0){
        teks=node.readStringUntil('\n');
        //teks+=baca;
    }
    if(teks.length(>0){

```



```

Serial.println(teks);
index1 = teks.indexOf('|');
index2 = teks.indexOf('|', index1+1);
index3 = teks.indexOf('|',index2+1);
index4 = teks.indexOf('|',index3+1);
index5 = teks.indexOf('|',index4+1);
index6 = teks.indexOf('|',index5+1);
index7 = teks.indexOf('|',index6+1);
A = teks.substring(index1+1, index2);
B = teks.substring(index2+1, index3);
C = teks.substring(index3+1, index4);
D = teks.substring(index4+1, index5);
E = teks.substring(index5+1, index6);
F = teks.substring(index6+1, index7);

Serial.println(A); Serial.println(B);
Serial.println(C); Serial.println(D);
Serial.println(E); Serial.println(F);
gas="MQ2:"+A+" *** Flame:"+B;
kadar="Kadar_gas:"+C+" *** Kadar_panas:"+D;
coa="Nilai_COA:"+E+" *** Kondisi:"+F;

```

3.5.5 Pengujian Sistem Keseluruhan

Pengujian sistem secara keseluruhan bertujuan untuk memastikan semua komponen dapat berjalan dengan sempurna. Mulai dari *Sensor MQ2*, sensor *falme* blok sistem *NodeMCU* dan program yang mengatur jalannya sistem keseluruhan.

3.6 Analisis Kerja

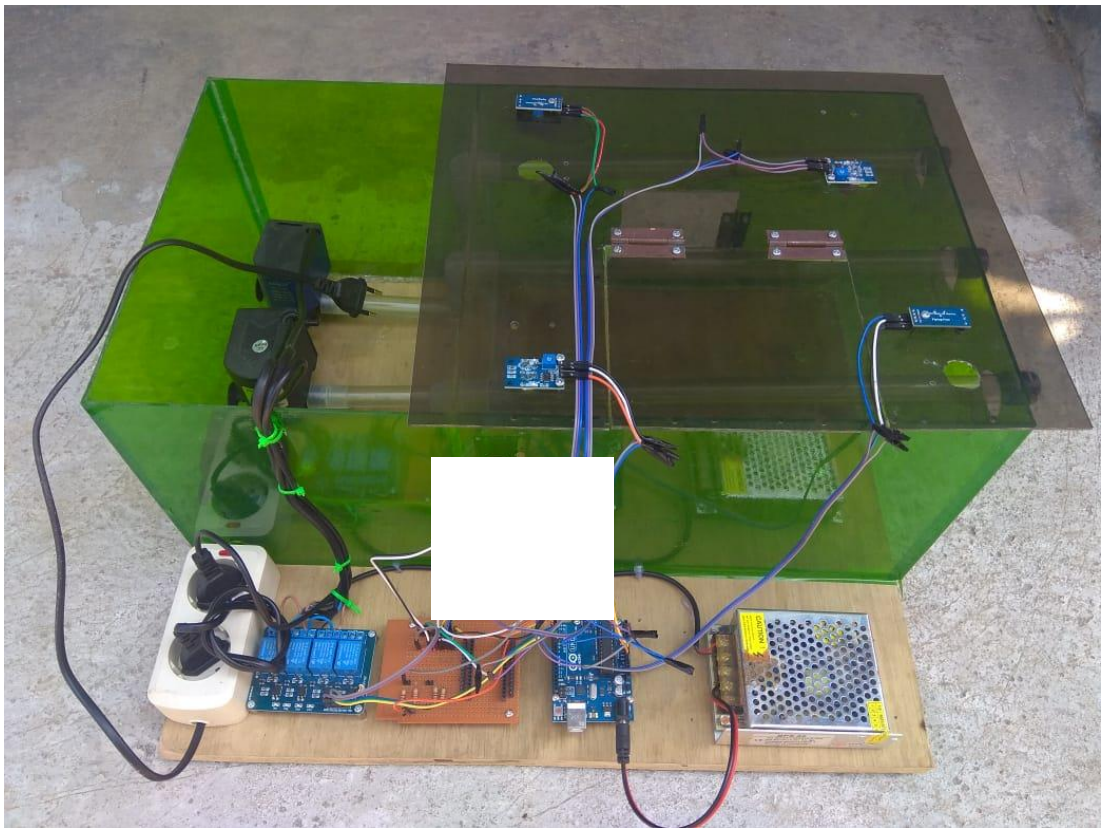
Untuk analisa kerja, dilakukan bersama pada saat melakukan uji coba alat yang bertujuan untuk mengetahui kerja alat tersebut. Selain itu yang akan dianalisa adalah jarak, respon dalam untuk inputan pada sistem “Implementasi *Fuzzy Logic Control* Pada Sistem Peringatan Dini Kebakaran Berbasis Iot”. Berdasarkan hasil

pengujian sistem yang telah di dapat akan dianalisis untuk memastikan bahwa sistem yang telah dibuat sesuai dengan harapan.

BAB IV HASIL DAN PEMBAHASAN

4.1 Hasil Perakitan

Uji coba dilakukan untuk memastikan rangkaian yang dihasilkan mampu bekerja sesuai dengan yang diharapkan. maka terlebih dahulu dilakukan langkah pengujian dan mengamati langsung rangkaian serta komponen. Hasil pengukuran ini dapat diketahui rangkaian telah bekerja dengan baik atau tidak, sehingga apabila terdapat kesalahan dan kekurangan akan terdeteksi. Gambar 4.1 berikut ini merupakan gambar dari bentuk fisik alat yang telah dibuat.



Gambar. 4.1. Bentuk Fisik Alat

Sistem kerja dari alat ini yaitu, Jika sensor *MQ2* mendeteksi adanya asap dan sensor *Flame* mendeteksi adanya api maka mengirimkan sinyal ke *Arduino*, *Arduino* menghidupkan relay yang tersambung ke pompa air dan mengirim perintah ke *NodeMCU* untuk mengirimkan informasi melalui *wifi* ke aplikasi

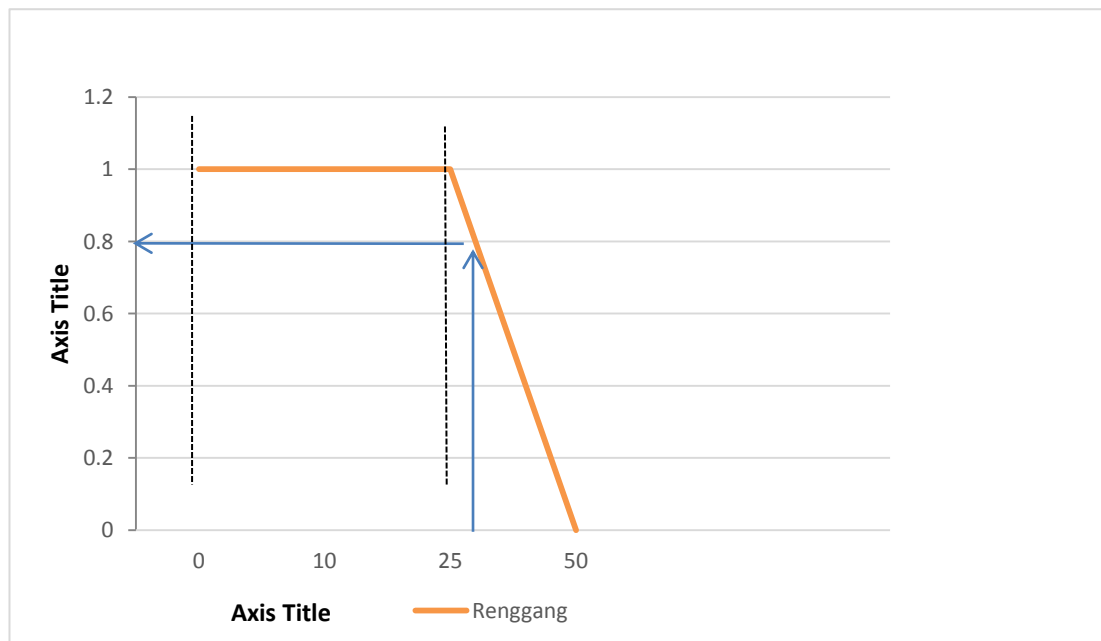
telegram yang telah terhubung dengan *nodemcu* hasil pembacaan sensor akan ditampilkan pada *chatting* aplikasi *Telegram*.

4.1.1 Hasil Pengujian dan Pembahasan

Pada pengujian ini meliputi pengujian Sensor *MQ2*, Sensor *Flame*, dan pengujian sistem keseluruhan. Pengujian ini dilakukan agar peneliti dapat mengetahui kelebihan dan kekurangan sistem yang telah dibuat hasil pengujian sebagai berikut:

4.1.2 Pengujian Sensor MQ2

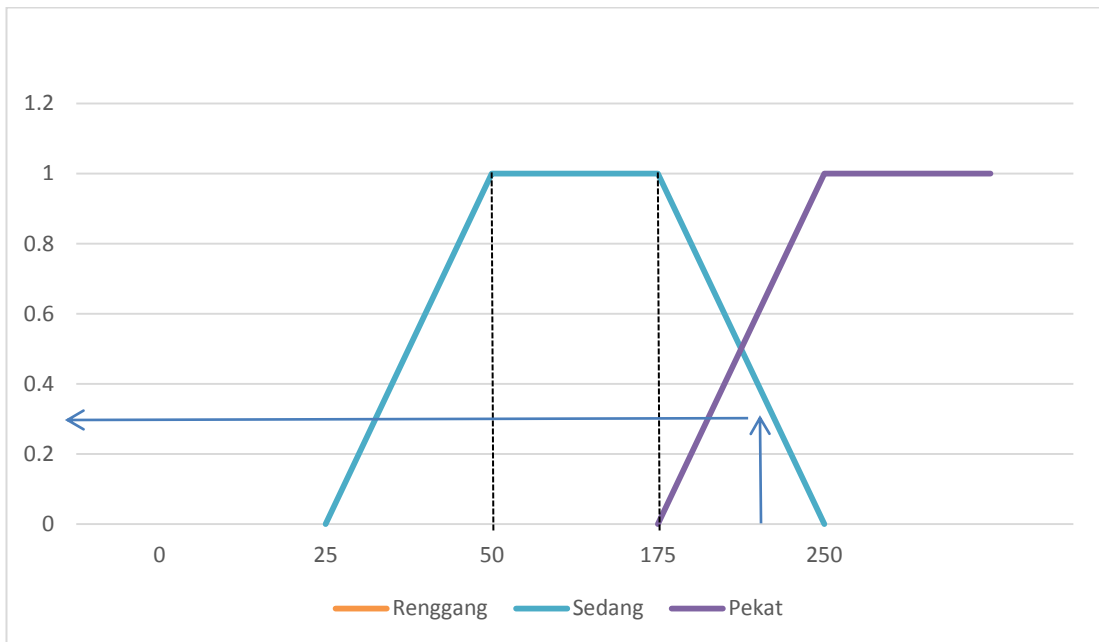
Pada pengujian akuisisi data dari sensor *MQ2* adalah untuk mengetahui kemampuan sensor *MQ2* dalam membaca kadar asap. Output dari sensor *MQ2* telah dikalibrasi kedalam bentuk ppm (*part per million*). Pengujian dilakukan dengan membuat asap hasil pembakaran dari gas dan rokok, Hasil pembacaan sensor seperti pada table 4.1.



Gambar 4.2 Diagram Keanggotaan Asap Renggang

Fungsi keanggotaan untuk himpunan renggang pada sensor *MQ2* seperti terlihat pada gambar 4.2

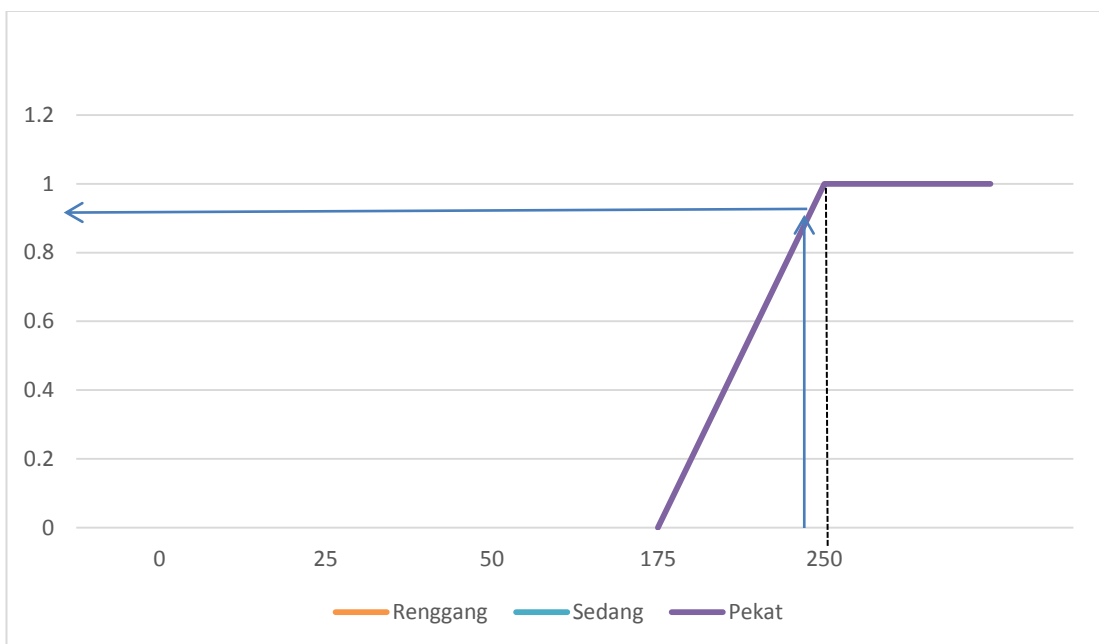
$$\begin{aligned}\mu_{\text{Renggang}}[30] &= (50-30)/(50-25) \\ &= 20/25 = 0,8\end{aligned}$$



Gambar 4.3 Diagram Keanggotaan Asap Sedang

Fungsi keanggotaan untuk himpunan sedang pada sensor *MQ2* seperti terlihat pada gambar 4.3

$$\begin{aligned} \mu_{\text{Sedang}}[200] &= (175-200)/(250-175) \\ &= 25/75 = 0,3 \end{aligned}$$



Gambar 4.4 Diagram Keanggotaan Asap Pekat

$$\begin{aligned} \mu_{\text{Pekat}}[245] &= (245-175)/(250-175) \\ &= 70/75 = 0,9 \end{aligned}$$

Tabel 4.1. Hasil Pengujian Sensor MQ2

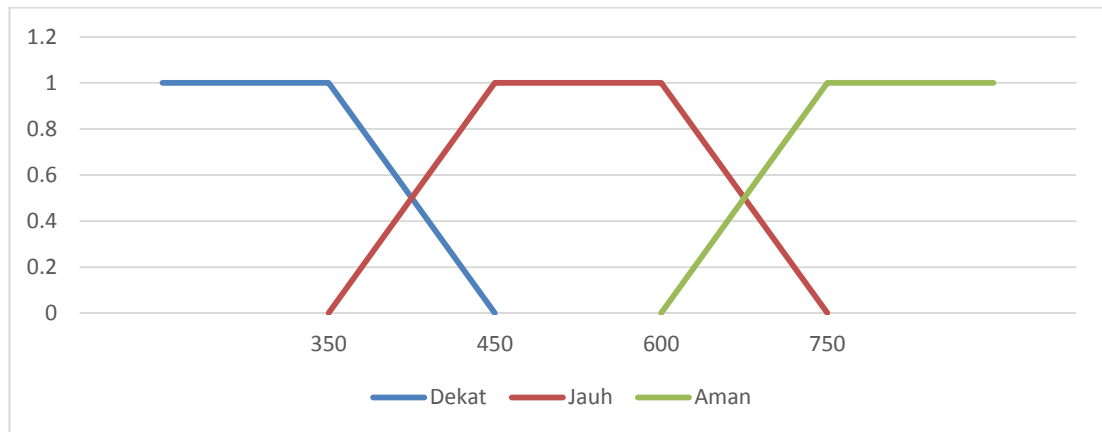
Uji Coba Ke-	Kondisi MQ2	Nilai Pembacaan MQ2	Kondisi COA	Nilai Pembacaan COA Pada Sensor MQ2
1	Renggang	0	Belum ada asap (Normal)	COA0 ppm
		10		COA10 ppm
		20		COA15 ppm
		40		COA20 ppm
		65		COA30 ppm
		70		COA35 ppm
2	Sedang	71	Berasap	COA 40 ppm
		75		COA 42 ppm
		85		COA 45 ppm
		90		COA 55 ppm
		95		COA 60 ppm
		100		COA 75 ppm
		233		
3	Pekat	255	Sedang	COA 77 ppm
		283		COA 80 ppm
		290		COA 85 ppm
		300		COA 90 ppm
		350		
4	Pekat	505	Berapi	COA 92ppm
		700		COA 95ppm
		1000		COA 100ppm

Dari hasil ujicoba Sensor *MQ2* dapat diketahui bahwa: sensor *MQ2* pada ujicoba pertama nilai sensor *MQ2* = 0 sampai dengan 70 maka hasil perhitungan nilai fuzzy maka dikategorikan (renggang), serta kondisi COA = 0 sampai dengan 35ppm (*part per million*) maka terdeteksi belum ada nya asap. Pada ujicoba ke dua nilai sensor *MQ2* = 71 sampai dengan 233 maka hasil perhitungan nilai fuzzy maka dikategorikan (Sedang) serta kondisi COA = 40 sampai dengan 75ppm (*part per million*) maka terdeteksi berasap, pada ujicba ketiga dan keempat nilai sensor *MQ2* = 255 sampai dengan 1000 maka hasil perhitungan nilai fuzzy maka dikategorikan (Pekat) serta pada ujicoba ketiga nilai kondisi COA = 77 sampai

dengan 90ppm (*part per million*) maka terdeteksi sedang, sedangkan pada ujicoba keempat nilai kondisi COA = 92 sampai dengan 100ppm (*part per million*) maka dikategorikan berapi.

4.1.3 Pengujian Flame

Pada pengujian sensor *Flame* adalah untuk mengetahui nilai ADC yang dihasilkan oleh sensor ketika mendeteksi adanya infrared yang dihasilkan oleh api.. Hasil dan gambar pengujian dapat dilihat seperti pada tabel diawah ini, seperti pada tabel 4.3



Gambar 4.5 Diagram Keanggotaan Api

Tabel 4.3 Hasil Pegujian Sensor Flame

Uji Coba Ke-	Jarak Cm	Nilai ADC Hasil Pembacaan Sensor	Keterangan
1	5	59	Dekat
2	10	90	Deket
3	20	161	Dekat
4	40	359	Dekat
5	60	536	Jauh
6	80	681	Jauh
7	100	743	Jauh
8	120	840	Aman
9	140	923	Aman
10	160	923	Aman

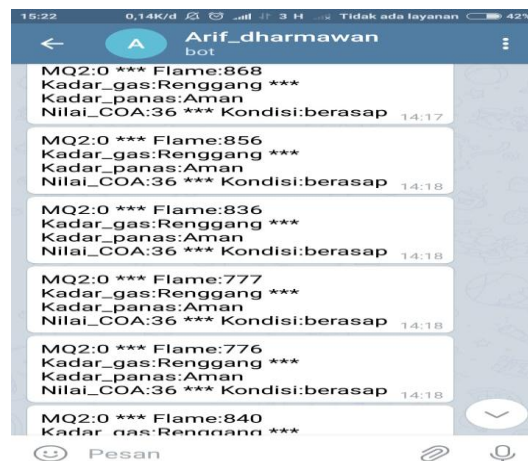
Dari hasil ujicoba dapat diketahui jika pada ujicoba kesatu dengan jarak api 5 sampai dengan 40 cm dengan nilai perhitungan ADC 0 sampai dengan 359 maka

dapat dikategorikan api “Dekat” sedangkan pada jarak 40 sampai dengan 100cm dengan nilai ADC 536 sampai dengan 743 dapat dikategorikan api “Jauh”, sedangkan dengan jarak 120 lebih dengan nilai perhitungan ADC 840 lebih dikategorikan “Aman”

4.1.4 Hasil Pengujian Pengiriman Telegram

Pengujian pengiriman *telegram* dilakukan agar mengetahui apakah aplikasi telegram dapat dengan baik mengirimkan informasi berupa nilai sensor *MQ2*, sensor *flame*, kadar gas, kadar panas, nilai COA dan kondisi. Gambar hasil pengujian dapat dilihat seperti

4.2.



pada gambar

Gambar 4.2 Hasil Pengiriman Telegram

4.1.5 Hasil Pengujian Sistem Keseluruhan

Pengujian sistem secara keseluruhan dilakukan untuk menguji kinerja Implementasi Fuzzy Logic Control Pada Sistem Peringatan Dini Kebakaran Berbasis Iot. Peneliti akan menguji coba sistem mulai dari kerja sensor *MQ2*, sensor *Flame* dan *relay*, dilakukan uji coba sistem agar peneliti dapat mengetahui apakah sistem yang telah dibuat dapat berkerja dengan baik. Dari hasil ujicoba sistem menggunakan 2 sensor *MQ2* dan 2 sensor *Flame* dapat diketahui bahwa sistem dapat berkerja dengan baik sesuai perintah pada program yang telah dibuat hasil dari ujicoba sistem keseluruhan dapat dilihat seperti gambar dan pada tabel dibawah ini:



Gambar 4.3 Kadar Gas Pekat Dan Kadar Panas Dekat



Gambar 4.4 Kadar Gas Sedang Dan Kadar Panas Jauh



Gambar 4.5 Kondisi Kadar Gas Renggang Dan Kadar Panas Aman



Gambar 4.6 Kondisi Kadar Gas Renggang Dan Kadar Panas

Nilai Fuzzyfikasi

NilaiMax ();

Normal = (5+10+15+20+25+30+35);

Berasap = (30+35+40+45+50+55+60+65+70);

Waspada = (65+70+75+80+85+90);

Berapi = (85+90+95+100);

$$\text{hasil_atas} = ((\text{normal}) * (\text{MAX_normal})) + ((\text{berasap}) * (\text{MAX_berasap})) + ((\text{waspada}) * (\text{MAX_waspada})) + ((\text{berapi}) * (\text{MAX_berapi}))$$

$$\text{hasil_bawah} = ((\text{MAX_normal}) * 7) + ((\text{MAX_berasap}) * 9) + ((\text{MAX_waspada}) * 6) + ((\text{MAX_berapi}) * 4)$$

$$\text{COA} = \frac{(\text{hasil_atas})}{(\text{hasil_bawah})}$$

Tabel 4.3 Hasil Ujicoba Keseluruhan

Nilai MQ2	Nilai Flame	Nilai COA	Status relay			Keterangan
			1	2	3	
320	665	65,535	High	High	Low	Waspada
234	536	36	High	Low	Low	Berasap
0	868	18	Low	Low	Low	Normal

Dari hasil ujicoba sistem keseluruhan dapat diketahui jika nilai sensor *MQ2* = 320 dan nilai sensor *Flame* = 46 sedangkan nilai COA = 59 maka pompa 1 dan 2 akan aktif artinya ruangan dala kondisi waspada sedangkan jika hasil pembacaan sensor *MQ2* = 234 dan *Flame* = 536 maka hanya pompa 1 yang akan menyala

mendakan hanya berasap sedang jika sensor *MQ2* = 1000 dan sensor *Flame* = 30 serta COA mendapatkan nilai 80 maka pompa 1, 2 dan 3 akan menyala artinya api berbahaya.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan pengujian dan analisa sistem yang telah dilakukan, dapat disimpulkan sebagai berikut:

Dari hasil ujicoba sistem keseluruhan menggunakan 2 sensor *MQ2* dan 2 sensor *Flame* dapat diketahui jika nilai sensor *MQ2* = 320 dan nilai sensor *flame* = 46 sedangkan nilai COA = 59 maka pompa 1 dan 2 akan aktif artinya ruangan dalam kondisi waspada sedangkan jika hasil pembacaan sensor *MQ2* = 234 dan *Flame* = 536 maka hanya pompa 1 yang akan menyala mendakan hanya berasap sedangkan jika sensor *MQ2* dan sensor *Flame* membaca nilai 100 dan 30 serta COA mendapatkan nilai 80 maka *Relay* 1, 2 dan 3 akan menyala artinya api dekat. Serta jika terjadinya pembacaan waspada, berasap dan berapi maka *NodeMCU* akan mengirim pesan via *Telegram* untuk memberikan informasi nilai pembacaan sensor *MQ2* dan sensor *Flame*.

5.2 Saran

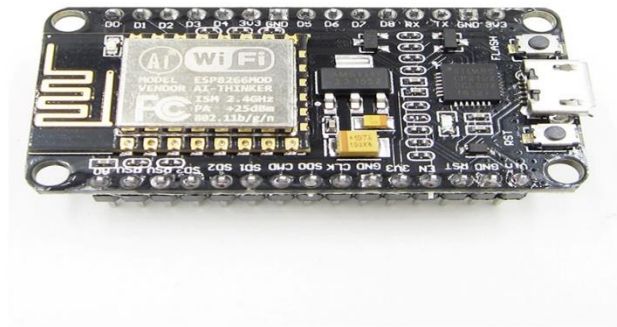
Alat ini masih terdapat kekurangan sehingga perlu diadakanya pengembangan. Berikut saran untuk pengembangan penelitian :

1. Akan lebih baik jika peneliti selanjutnya menggunakan lebih dari 2 sensor baik sensor *MQ2* dan Sensor *Flame*.
2. Perlu dilakukan penambahan camera agar dapat lebih baik dalam memonitoring kondisi ruangan.

DAFTAR PUSTAKA

- Apriyandi, S. (2013). Rancang Bangun Sistem Detektor Kebakaran Via Handphone Berbasis Mikrokontroler. *Prosiding SNATIF Ke-4*, 1.
- Dodon Yendri, W. (2017). Perancangan Sistem Pendeteksi Kebakaran Rumah Penduduk Pada Daerah Perkotaan Berbasis Mikrokontroler. *Teknik Elektro*.
- Hendri Andrianto, M. A. (2015). Relisasi Sistem Peringatan Kebakaran Melalui Layanan SMS dan MMS. *Electrical Enggenering*, Vol 1.
- Mega Apriyaningsih, A. M. (2017). Prototipe Sstem Pemadaman Kebakaran Otomatis Berbasis Mikrokonroler Atmega328. *Prisma Fisika*, Vol 3, Hal 106-110.
- Nulhakim, L. (2014). Alat Pemberi Makan Ikan Di Akuarium Otomatis Berbasis Mikrokontroler Atmega16. *Teknik Elektro*.
- Ratna Susana, S. . (2015). Implemantasi Wireless Sensor Network Prototype Sebagai Fire Detector Menggunakan Arduino Uno. *Jurnal Elektro Komunikasi Terapan*, 1.
- Rinanda, M. R. (2014). Rancang Bangun Smart Aquarium Berbasis Android Dan Short Message Service . *Citee journall*, vol 1.
- Siti Fatima, A. T. (2017). Prototye Kebakaran Hutan Menggunakan Model Warna CMY Dan CMYK. *Prosiding SNATIF Ke-4*, 1.
- Zain, A. (2016). Rancang Bangun Sistem Proteksi Kebakaran Menggunakan Smoke dan Heat Detector. *INTEK*, Vol 3.

NODEMCU LOLIN V3



Lolin NodeMCU V3 is an open source IoT platform. It uses the Lua scripting language. The eLua project is the basis of board, and built on the ESP8266 SDK 1.4. NodeMCU uses many open source projects, such as lua-cjson, and spiffs. The NodeMCU runs on the ESP8266 Wi-Fi SoC, and hardware which based on the ESP-12 module.

The Lolin NodeMCU V3 board ads USB/UART converter chip as well as decoupled LDO power supply. Also the board adds 2 miniature push buttons. The most important feature is that it breaks out all ESP8266 pins to board headers. The board headers are breadboard compatible 2.54 mm pitch headers.

The Lolin NodeMCU board uses the CH340G USB/UART converter chip. You will need to download and install the proper driver to get going with the development. You can find the drivers here:

NodeMCU CH340/CH340G Driver Download page (if not automatically recognised): [Click Here](#)

For MAC users please check this link .

Once you install the driver you can use the control panel to get the assigned serial COM port number. Connect to the LUA interpreter running on the ESP8266 via your favorite terminal emulator. The baud rate for most of the boards is 9600 baud (1 start bit, 8 data bits, no parity).

An easy way to communicate with the LUA interpreter on the ESP8266 is by using the ESP8266 LuaLoader. It allows you to perform simple tasks. For example, you can set the SSID and password for your wireless router so it can connect to your network via GUI. You can also read out or set the status of its GPIO ports. Get information like the IP address or the chipID, or upload files. Yet, you can as well try ESPlorer which has a more contemporary user interface. It also has syntax highlighting on LUA code.

You can find some usage examples here to start your development activities.

Lolin NodeMCU Features

- Arduino-Like Hardware IO
- Code like Arduino, but interactively in Lua script
- Event-driven API for network applications, which facilitates developers writing code
- Integrates GPIO, PWM, IIC, 1-Wire and ADC all in one board
- 10 GPIO, every GPIO can be PWM, I2C, 1-wire
- 4M Flash Memory
- Built-in WiFi Antenna

SONGLE RELAY

	RELAY ISO9002	SRD
---	---------------	-----



1. MAIN FEATURES

- Switching capacity available by 10A in spite of small size design for highdensity P.C. board mounting technique.
- UL,CUL,TUV recognized.
- Selection of plastic material for high temperature and better chemical solution performance.
 - Sealed types available.
- Simple relay magnetic circuit to meet low cost of mass production.

2. APPLICATIONS

- Domestic appliance, office machine, audio, equipment, automobile, etc.
(Remote control TV receiver, monitor display, audio equipment high rushing current use application.)

3. ORDERING INFORMATION

SRD	XX VDC	S	L	C
Model of relay	Nominal coil voltage	Structure	Coil	Contact form
SRD	03 05 06 09 12 24 48VDC	S:Sealed type F:Flux free type	L:0.36W D:0.45W	A:1 form A B:1 form B C:1 form C

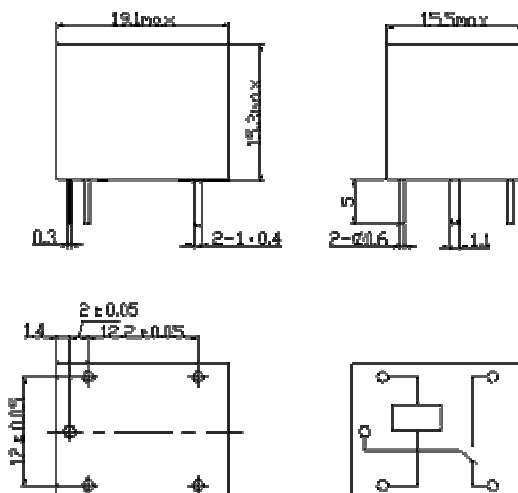
4. RATING

CCC	FILE NUMBER:CQC03001003729	7A/240VDC
CCC	FILE NUMBER:CQC03001003731	10A/250VDC
UL/CUL	FILE NUMBER: E167996	10A/125VAC 28VDC
TUV	FILE NUMBER: R50056114	10A/250VAC 30VDC

5. DIMENSION(unit:mm)

DRILLING(unit:mm)

WIRING DIAGRAM



6. COIL DATA CHART (AT20 ° C)

Coil Sensitivity	Coil Voltage Code	Nominal Voltage (VDC)	Nominal Current (mA)	Coil Resistance (Ω) □	Power Consumption (W)	Pull-In Voltage (VDC)	Drop-Out Voltage (VDC)	Max-Allowable Voltage (VDC)
SRD (High Sensitivity)	03	03	120	25	abt. 0.36W	75%Max.	10% Min.	120%
	05	05	71.4	70				
	06	06	60	100				
	09	09	40	225				
	12	12	30	400				
	24	24	15	1600				
SRD (Standard)	03	03	150	20	abt. 0.45W	75% Max.	10% Min.	110%
	05	05	89.3	55				
	06	06	75	80				
	09	09	50	180				
	12	12	37.5	320				
	24	24	18.7	1280				
	48	48	10	4500	abt. 0.51W			

7. CONTACT RATING

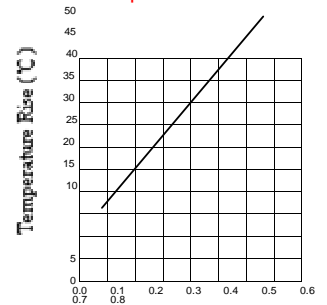
Item	Type	SRD
	FORM C	FORM A
Contact Capacity	7A	10A 30VDC
Resistive Load (cosΦ=1)	30VDC	10A 240VAC
Inductive Load (cosΦ=0.4 L/R=7msec)	10A 125VAC	5A 120VAC
	10A 250VAC	5A 28VDC
	3A 120VAC	
	3A 28VDC	
Max. Allowable Voltage	250VAC/110VDC	250VAC/110VDC
Max. Allowable Power Force	800VAC/240W	1200VA/300W
Contact Material	AgCdO	AgCdO

8. PERFORMANCE (at initial value)

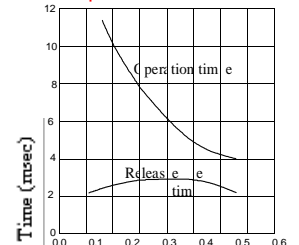
Item	Type	SRD
Contact Resistance		100mΩ Max.
Operation Time		10msec Max.
Release Time		5msec Max.
Dielectric Strength	Between coil & contact	1500VAC 50/60HZ (1 minute)
	Between contacts	1000VAC 50/60HZ (1 minute)
Insulation Resistance		100 MΩ Min. (500VDC)
Max. ON/OFF Switching	Mechanically	300 operation/min
	Electrically	30 operation/min
Ambient Temperature		-25°C to +70 C
Operating Humidity		45 to 85% RH
Vibration	Endurance	10 to 55Hz Double Amplitude 1.5mm
	Error Operation	10 to 55Hz Double Amplitude 1.5mm
Shock	Endurance	100G Min.
	Error Operation	10G Min.
Life Expectancy	Mechanically	10 ⁷ operations. Min. (no load)
	Electrically	10 ⁵ operations. Min. (at rated coil voltage)
Weight		abt. 10grs.

9. REFERENCE DATA

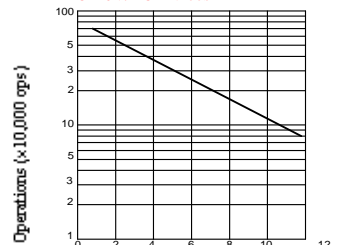
Coil Temperature Rise



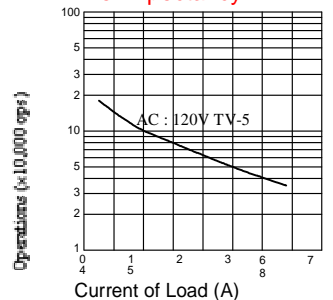
Operation Time



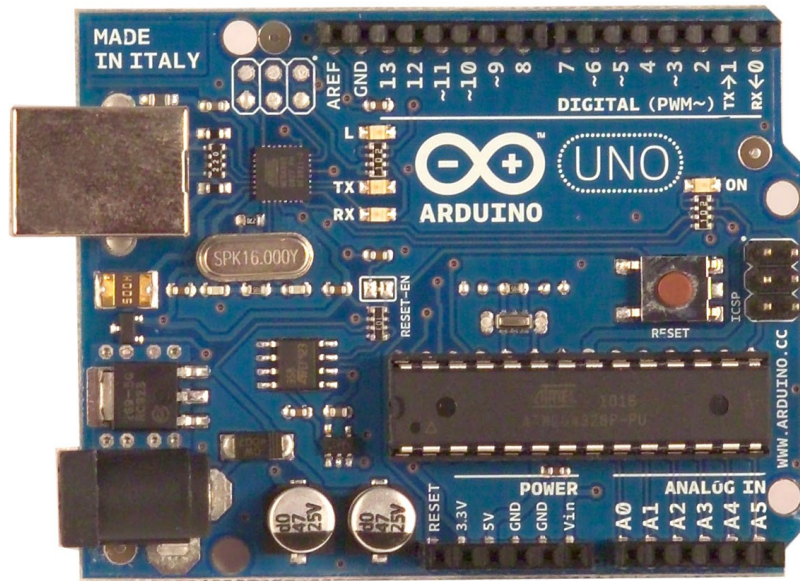
Life Expectancy



Life Expectancy



Arduino UNO



Product Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

Index

Technical Specifications

Page 2

How to use Arduino
Programming Environment, Basic Tutorials

Page 6

Terms & Conditions

Page 7

Environmental Policies
half sqm of green via Impatto Zero®

Page 7



radiospares

RADIONICS



Technical Specification

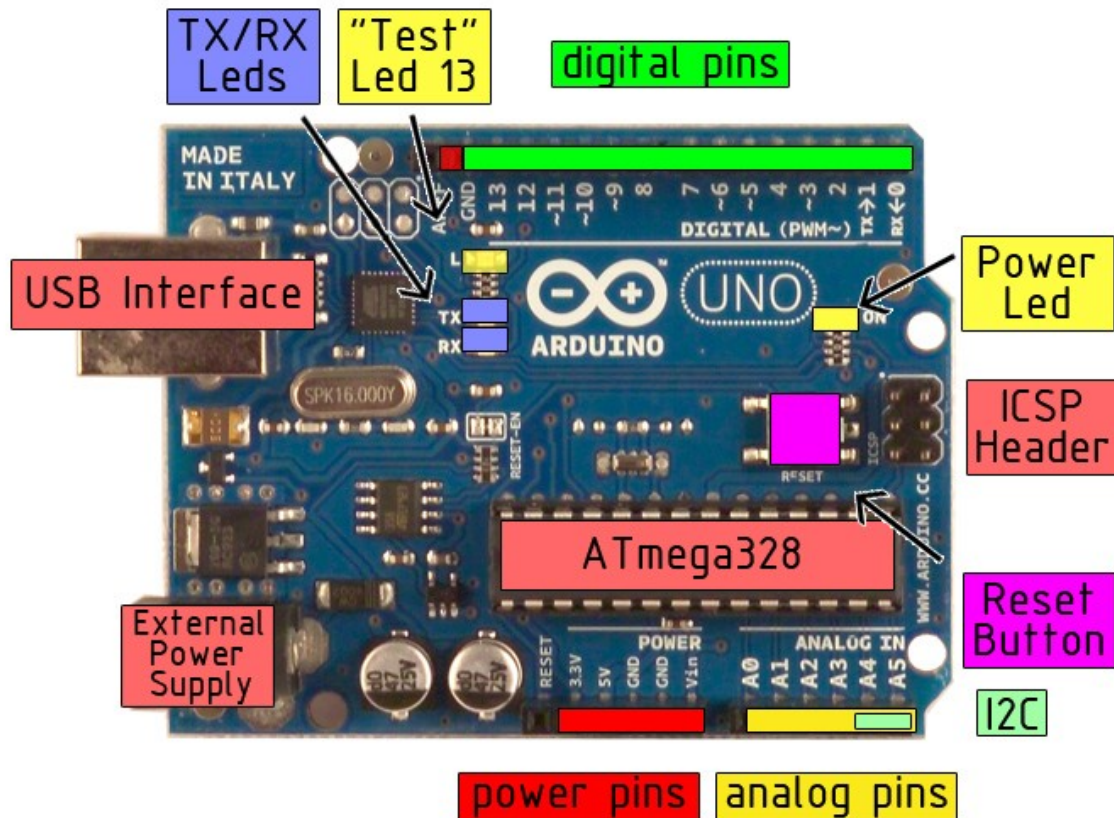


EAGLE files: [arduino-duemilanove-uno-design.zip](#) Schematic: [arduino-uno-schematic.pdf](#)

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

the board



radiospares

RADIONICS



Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The Atmega328 has 32 KB of flash memory for storing code (of which 0,5 KB is used for the bootloader); It has also 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.



radiospares

RADIONICS



The Uno has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- **I²C: 4 (SDA) and 5 (SCL).** Support I²C (TWI) communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and Atmega328 ports](#).

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega8U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '8U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an *.inf file is required..

The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. To use the SPI communication, please see the ATmega328 datasheet.

Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno w/ ATmega328" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega8U2 firmware source code is available . The ATmega8U2 is loaded with a DFU bootloader, which can be activated by connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader).



RADIOSPARES

RADIONICS



Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

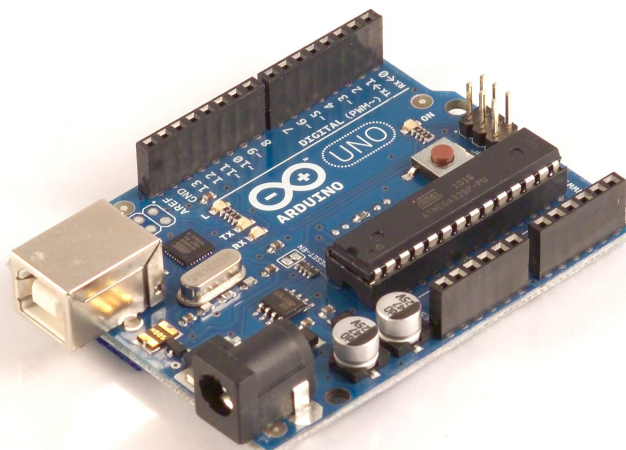
The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.



RADIOSPARES

RADIONICS



How to use Arduino



Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the [Arduino programming language](#) (based on [Wiring](#)) and the Arduino development environment (based on [Processing](#)). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platform program. You'll have to follow different instructions for your personal OS. Check on the [Arduino site](http://arduino.cc/en/Guide/HomePage) for the latest instructions. <http://arduino.cc/en/Guide/HomePage>

Linux Install

Windows Install

Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

**File>Sketchbook>
Arduino-0017>Examples>
Digital>Blink**

Once you have your sketch you'll see something very close to the screenshot on the right.

In **Tools>Board** select

Now you have to go to **Tools>SerialPort** and select the right serial port, the one arduino is attached to.

```
Blink | Arduino 0017
File Edit Sketch Tools Help
Blink $
int ledPin = 13; // LED connected to digital pin 13

// The setup() method runs once, when the sketch starts

void setup() {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power

void loop()
{
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(1000); // wait for a second
  digitalWrite(ledPin, LOW); // set the LED off
  delay(1000); // wait for a second
}
```



Done compiling.

Press Compile button
(to check for errors)



Upload



TX RX Flashing



Blinking Led!

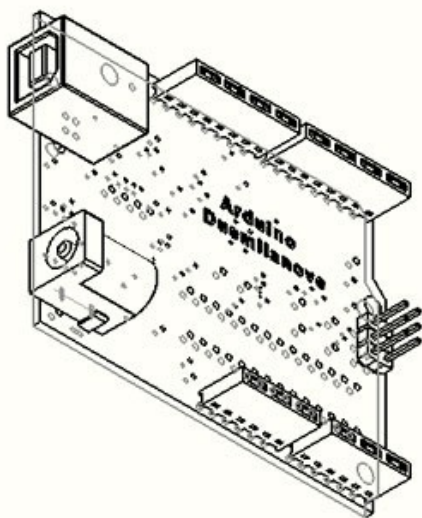
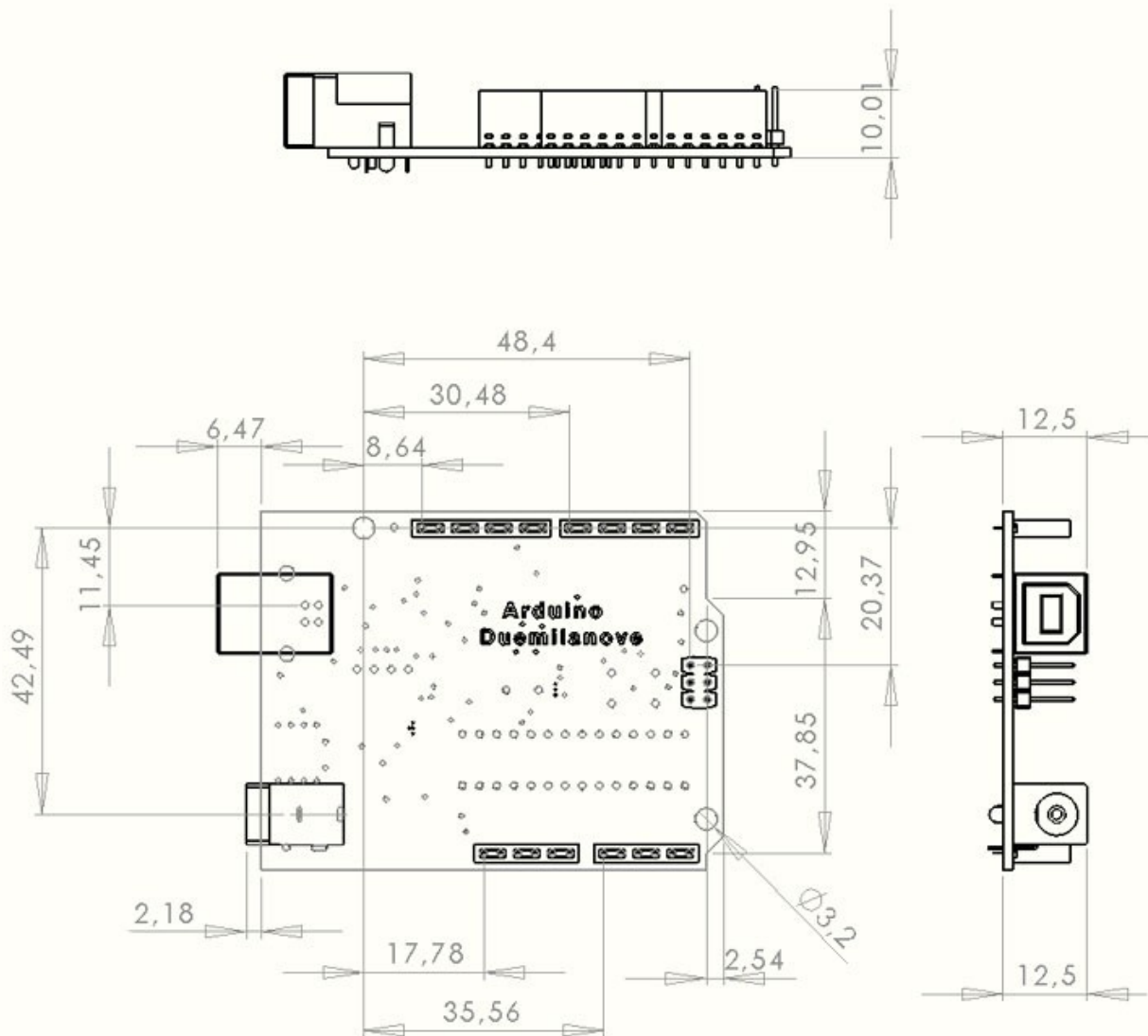


radiospares

RADIONICS



Dimensioned Drawing



radiospares **RADIONICS**



Terms & Conditions



1. Warranties

1.1 The producer warrants that its products will conform to the Specifications. This warranty lasts for one (1) years from the date of the sale. The producer shall not be liable for any defects that are caused by neglect, misuse or mistreatment by the Customer, including improper installation or testing, or for any products that have been altered or modified in any way by a Customer. Moreover, The producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.

1.2 If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are determined by the producer not to conform to such warranty. If the producer elects to replace such products, the producer shall have a reasonable time to replacements. Replaced products shall be warranted for a new full warranty period.

1.3 EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE

1.4 Customer agrees that prior to using any systems that include the producer products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical, applications or design advice, quality characterization, reliability data or other services. Customer acknowledges and agrees that providing these services shall not expand or otherwise alter the producer's warranties, as set forth above, and no additional obligations or liabilities shall arise from the producer providing such services.

1.5 The Arduino™ products are not authorized for use in safety-critical applications where a failure of the product would reasonably be expected to cause severe personal injury or death. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino™ products are neither designed nor intended for use in military or aerospace applications or environments and for automotive applications or environment. Customer acknowledges and agrees that any such use of Arduino™ products which is solely at the Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

1.6 Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino™ products in Customer's applications, notwithstanding any applications-related information or support that may be provided by the producer.

2. Indemnification

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third-party losses, damages, liabilities and expenses it incurs to the extent directly caused by: (i) an actual breach by a Customer of the representation and warranties made under this terms and conditions or (ii) the gross negligence or willful misconduct by the Customer.

3. Consequential Damages Waiver

In no event the producer shall be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.

4. Changes to specifications

The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." The producer reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.



Environmental Policies



The producer of Arduino™ has joined the Impatto Zero® policy of LifeGate.it. For each Arduino board produced is created / looked after half squared Km of Costa Rica's forest's.



radiospares

RADIONICS



Flame Sensor Module



Introduction

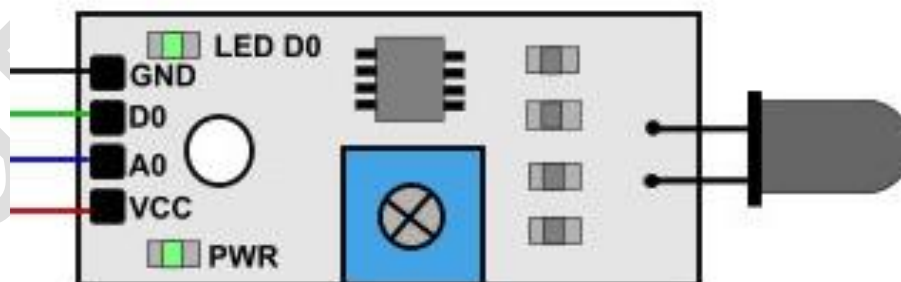
This module is sensitive to the flame and radiation. It also can detect ordinary light source in the range of of a wavelength 760nm-1100 nm. The detection distance is up to 100 cm. The Flame sensor can output digital or analog signal. It can be used as a flame alarm or in fire fighting robots.

Description

- Detects a flame or a light source of a wavelength in the range of 760nm-1100 nm
- Detection distance: 20cm (4.8V) ~ 100cm (1V)
- Detection angle about 60 degrees, it is sensitive to the flame spectrum.
- Comparator chip LM393 makes module readings stable.
- Adjustable detection range.
- Operating voltage 3.3V-5V
- Digital and Analog Output
 - " DO digital switch outputs (0 and 1)
 - " AO analog voltage output
- Power indicator and digital switch output indicator

Interface Description (4-wire)

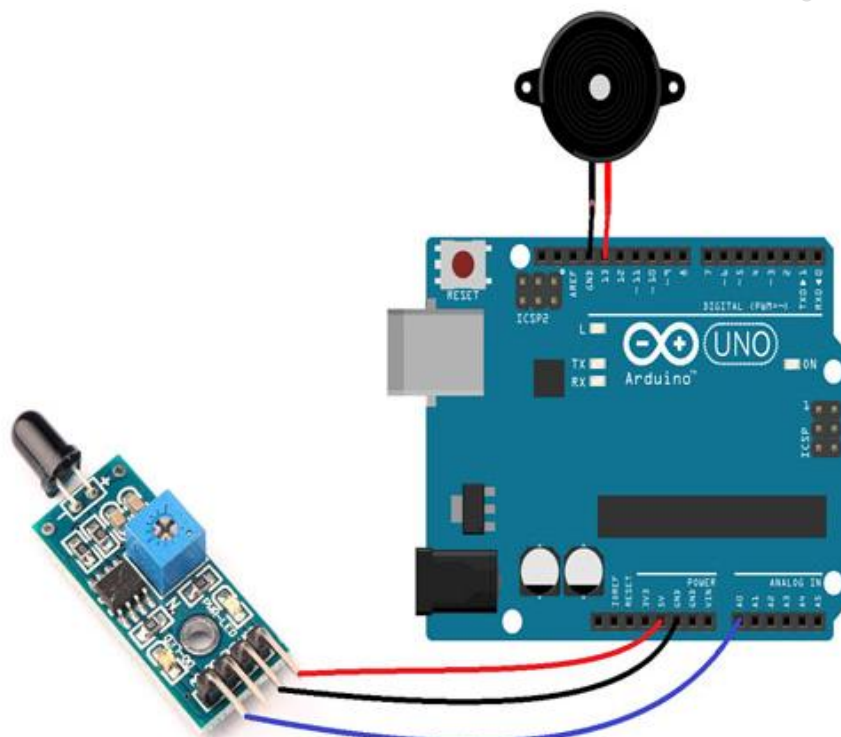
- 1) VCC -- 3.3V-5V voltage
- 2) GND -- GND
- 3) DO -- board digital output interface (0 and 1)
- 4) AO -- board analog output interface



Arduino Example

Here is sample code and connection to Arduino board. The analog output can be connected to any analog input pin on Arduino.

AnalogReadSerial (A0)



Reads an analog input on pin 0, prints the result to the serial monitor.
Attach the center pin of a potentiometer to pin A0, and the outside pins to +5V and ground.

This example code is in the public domain.

*/

// the setup routine runs once when you press reset:

```
const int analogPin = A0; // Flame Sensor (A0) to Arduino analog input pin A0
```

```
const int BuzzerPin = 13; // Buzzer output pin
```

```
const int threshold = 400; // Flame level threshold (You can vary the value depends on your need)
```

```
void setup() {
```

```
pinMode(BuzzerPin, OUTPUT);
```

```
// initialize serial communications:
```

```
Serial.begin(9600);

}

void loop() {

  // read the value of the Flame Sensor:

  int analogValue = analogRead(analogPin);

  Serial.println(analogValue); //serial print the FLAME sensor value

  if (analogValue > threshold) {

    digitalWrite(BuzzerPin, HIGH);

    Serial.print("High FLAME");

  }

  else if (analogValue = threshold){

    Serial.print("Low FLAME");

    digitalWrite(BuzzerPin, HIGH);

    delay(400);

    digitalWrite(BuzzerPin, LOW);

  }

  else {

    digitalWrite(BuzzerPin, LOW);

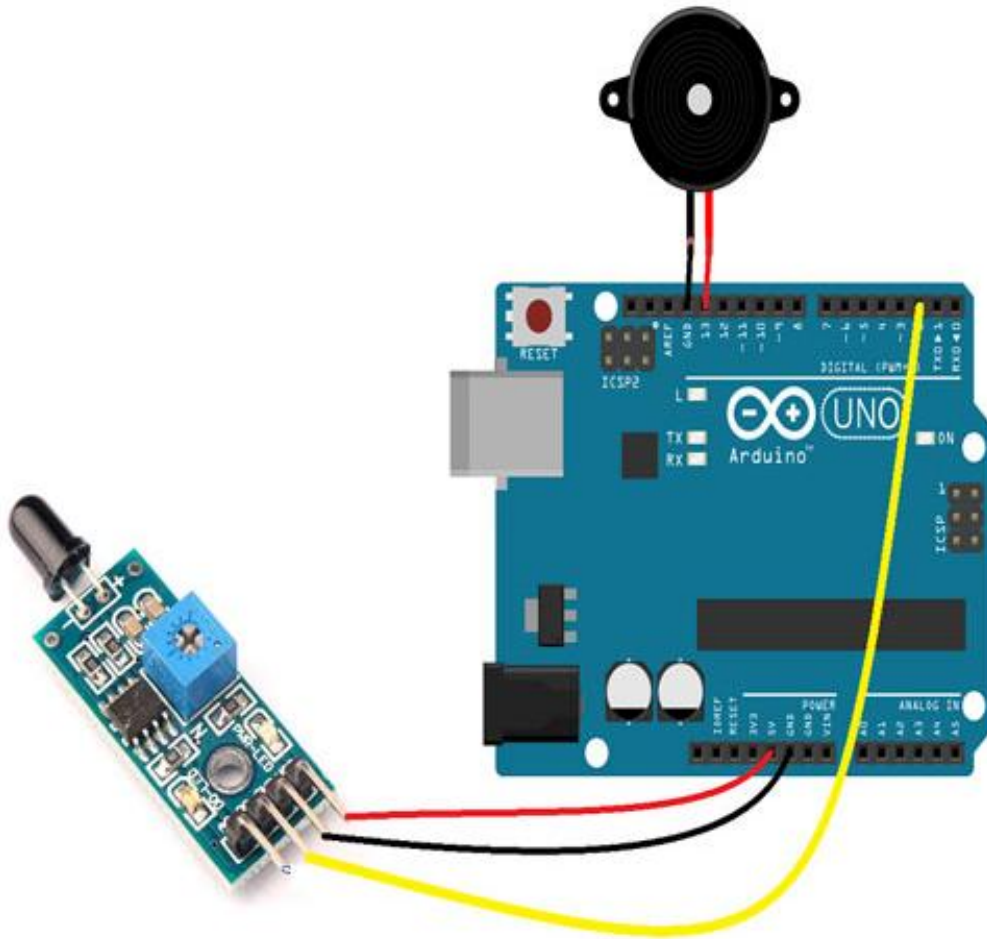
    Serial.print("No flame");

  }

  delay(1);

}
```

Digital (D0) Output



```
int Buzzer = 13; // Use buzzer for alert

int FlamePin = 2; // This is for input pin

int Flame = HIGH; // HIGH when FLAME Exposed

void setup() {

  pinMode(Buzzer, OUTPUT);

  pinMode(FlamePin, INPUT);

  Serial.begin(9600);
```

```
}  
  
void loop() {  
  
  Flame = digitalRead(FlamePin);  
  
  if (Flame== HIGH)  
  
  {  
  
    Serial.println("HIGH FLAME");  
  
    digitalWrite(Buzzer, HIGH);  
  
  }  
  
  else  
  
  {  
  
    Serial.println("No flame");  
  
    digitalWrite(Buzzer, LOW);  
  
  }  
  
}
```


MQ-2 Semiconductor Sensor for Combustible Gas

Sensitive material of MQ-2 gas sensor is SnO₂, which with lower conductivity in clean air. When the target combustible gas exist, The sensor's conductivity is more higher along with the gas concentration rising. Please use simple electrocircuit, Convert change of conductivity to correspond output signal of gas concentration.

MQ-2 gas sensor has high sensitivity to LPG, Propane and Hydrogen, also could be used to Methane and other combustible steam, it is with low cost and suitable for different application.

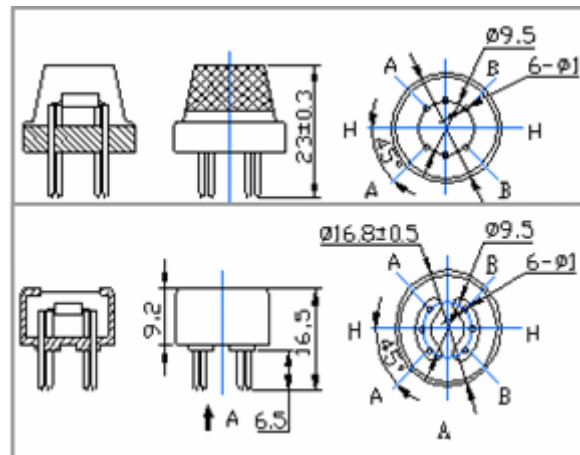
Character

- * Good sensitivity to Combustible gas in wide range
- * High sensitivity to LPG, Propane and Hydrogen
- * Long life and low cost
- * Simple drive circuit

Application

- * Domestic gas leakage detector
- * Industrial Combustible gas detector
- * Portable gas detector

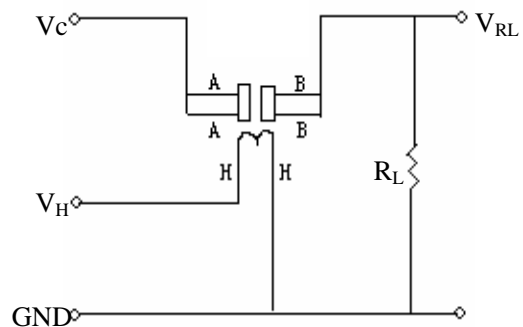
Configuration



Technical Data

Model No.		MQ-2	
Sensor Type		Semiconductor	
Standard Encapsulation		Bakelite (Black Bakelite)	
Detection Gas		Combustible gas and smoke	
Concentration		300-10000ppm (Combustible gas)	
Circuit	Loop Voltage	V _c	≤24V DC
	Heater Voltage	V _H	5.0V±0.2V AC or DC
	Load Resistance	R _L	Adjustable
Character	Heater Resistance	R _H	31Ω±3Ω (Room Tem.)
	Heater consumption	P _H	≤900mW
	Sensing Resistance	R _s	2KΩ-20KΩ(in 2000ppm C ₃ H ₈)
	Sensitivity	S	R _s (in air)/R _s (1000ppm isobutane) ≥ 5
	Slope	α	≤ 0.6(R _{5000ppm} /R _{3000ppm} CH ₄)
Condition	Tem. Humidity	20°C±2°C; 65%±5%RH	
	Standard test circuit	V _c : 5.0V±0.1V; V _H : 5.0V±0.1V	
	Preheat time	Over 48 hours	

Basic test loop



The above is basic test circuit of the sensor. The sensor need to be put 2 voltage, heater voltage (V_H) and test voltage (V_C). V_H used to supply certified working temperature to the sensor, while V_C used to detect voltage (V_{RL}) on load resistance (R_L) whom is in series with sensor. The sensor has light polarity, V_c need DC power. V_C and V_H could use same power circuit with precondition to assure performance of sensor. In order to make the sensor with better performance, suitable R_L value is needed:
Power of Sensitivity body (P_s):
$$P_s = V_c^2 \times R_s / (R_s + R_L)^2$$

Resistance of sensor(R_s): $R_s=(V_c/V_{RL}-1)\times R_L$

Sensitivity Characteristics

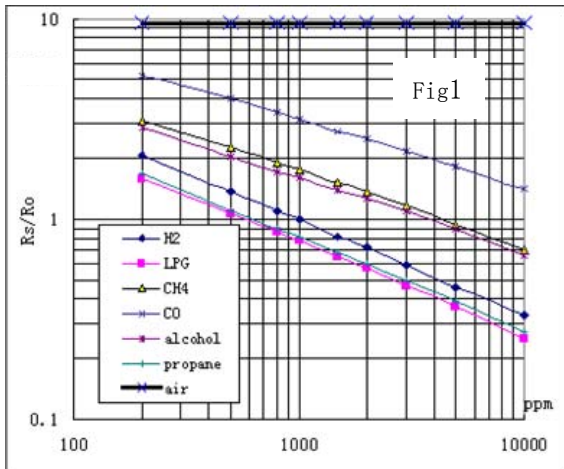


Fig.1 shows the typical sensitivity characteristics of the MQ-2, ordinate means resistance ratio of the sensor (R_s/R_o), abscissa is concentration of gases. R_s means resistance in different gases, R_o means resistance of sensor in 1000ppm Hydrogen. All test are under standard test conditions.

Influence of Temperature/Humidity

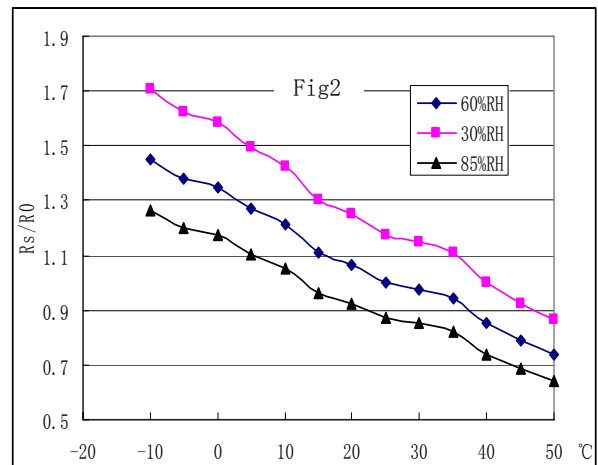
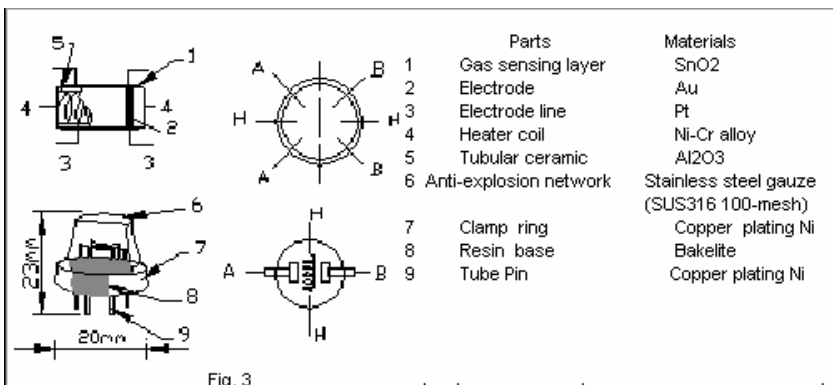


Fig.2 shows the typical temperature and humidity characteristics. Ordinate means resistance ratio of the sensor (R_s/R_o), R_s means resistance of sensor in 1000ppm Butane under different tem. and humidity. R_o means resistance of the sensor in environment of 1000ppm Methane, 20°C/65%RH

Structure and configuration



Structure and configuration of MQ-2 gas sensor is shown as Fig. 3, sensor composed by micro Al₂O₃ ceramic tube, Tin Dioxide (SnO₂) sensitive layer, measuring electrode and heater are fixed into a crust made by plastic and stainless steel net. The heater provides necessary work conditions for work of sensitive components. The enveloped MQ-2 have 6 pin, 4 of them are used to fetch signals, and other 2 are used for providing heating current.

Notification

1 Following conditions must be prohibited

1.1 Exposed to organic silicon steam

Organic silicon steam cause sensors invalid, sensors must be avoid exposing to silicon bond, fixture, silicon latex, putty or plastic contain silicon environment

1.2 High Corrosive gas

If the sensors exposed to high concentration corrosive gas (such as H_2S_z , SO_x , Cl_2 , HCl etc), it will not only result in corrosion of sensors structure, also it cause sincere sensitivity attenuation.

1.3 Alkali, Alkali metals salt, halogen pollution

The sensors performance will be changed badly if sensors be sprayed polluted by alkali metals salt especially brine, or be exposed to halogen such as fluorin.

1.4 Touch water

Sensitivity of the sensors will be reduced when splattered or dipped in water.

1.5 Freezing

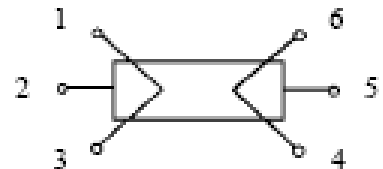
Do avoid icing on sensor's surface, otherwise sensor would lose sensitivity.

1.6 Applied voltage higher

Applied voltage on sensor should not be higher than stipulated value, otherwise it cause down-line or heater damaged, and bring on sensors' sensitivity characteristic changed badly.

1.7 Voltage on wrong pins

For 6 pins sensor, if apply voltage on 1、3 pins or 4、6 pins, it will make lead broken, and without signal when apply on 2、4 pins



2 Following conditions must be avoided

2.1 Water Condensation

Indoor conditions, slight water condensation will effect sensors performance lightly. However, if water condensation on sensors surface and keep a certain period, sensor' sensitivity will be decreased.

2.2 Used in high gas concentration

No matter the sensor is electrified or not, if long time placed in high gas concentration, if will affect sensors characteristic.

2.3 Long time storage

The sensors resistance produce reversible drift if it's stored for long time without electrify, this drift is related with storage conditions. Sensors should be stored in airproof without silicon gel bag with clean air. For the sensors with long time storage but no electrify, they need long aging time for stbility before using.

2.4 Long time exposed to adverse environment

No matter the sensors electrified or not, if exposed to adverse environment for long time, such as high humidity, high temperature, or high pollution etc, it will effect the sensors performance badly.

2.5 Vibration

Continual vibration will result in sensors down-lead response then repture. In transportation or assembling line, pneumatic screwdriver/ultrasonic welding machine can lead this vibration.

2.6 Concussion

If sensors meet strong concussion, it may lead its lead wire disconnected.

2.7 Usage

For sensor, handmade welding is optimal way. If use wave crest welding should meet the following conditions:

2.7.1 Soldering flux: Rosin soldering flux contains least chlorine

2.7.2 Speed: 1-2 Meter/ Minute

2.7.3 Warm-up temperature: $100\pm 20^{\circ}C$

2.7.4 Welding temperature: $250\pm 10^{\circ}C$

2.7.5 1 time pass wave crest welding machine

If disobey the above using terms, sensors sensitivity will be reduced.