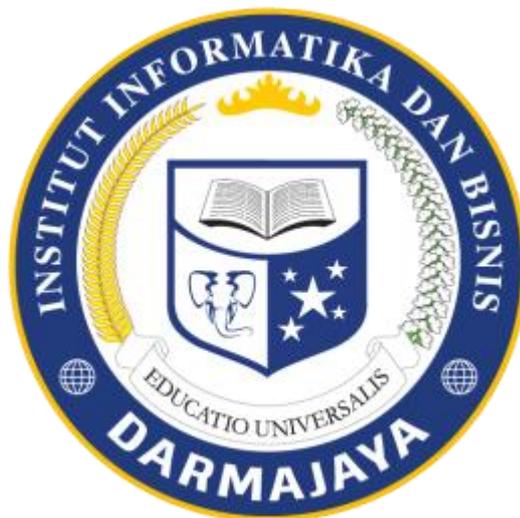


**PENGONTROL SUHU DAN PEMBERI PAKAN AYAM OTOMATIS
PADA PETERNAKAN AYAM DENGAN OUTPUT SMS BERBASIS
MIKROKONTROLER**

SKRIPSI

**Diajukan Sebagai Salah Satu Syarat Untuk Mencapai Gelar
SARJANA
Pada Program Studi Sistem Komputer
IIB Darmajaya Bandar Lampung**



Oleh

Sunaryo

1411060025

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

PERNYATAAN ORISINILITAS PENELITIAN



Saya yang bertanda tangan di bawah ini, menyatakan bahwa Skripsi yang saya ajukan 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 dan diterbitkan orang lain kecuali yang secara tertulis diacu dalam naskah ini disebutkan dalam daftar pustaka. Karya ini adalah milik saya dan pertanggungjawaban sepenuhnya berada dipundak saya.

Bandar Lampung, 16 September 2019



Sunaryo
1411060025

PERSETUJUAN

Judul Skripsi : **Pengontrol Suhu Dan Pemberi pakan Ayam Otomatis Pada Peternakan Ayam Dengan Output Sms Berbasis Mikrokontroler.**

Nama Mahasiswa : **Sunaryo**

No. Pokok Mahasiswa : **1411060025**

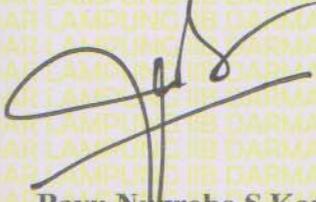
Jurusan : **S1 Sistem Komputer**

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

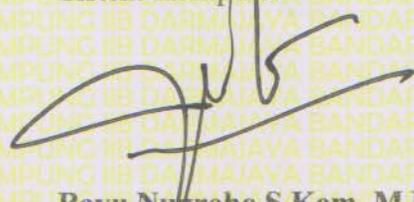
Disetujui oleh :

Dosen Pembimbing

Ketua Jurusan,
Sistem Komputer


Bayu Nugroho S.Kom.,M.Eng

NIK 00200700


Bayu Nugroho S.Kom.,M.Eng

NIK 00200700

PENGESAHAN

Telah diuji dan dipertahankan didepan tim penguji Skripsi
prodi Sistem Komputer Institut Informatika dan Bisnis Darmajaya
Bandar Lampung dan dinyatakan diterima untuk
memenuhi syarat guna memperoleh Gelar
Sarjana

Mengesahkan

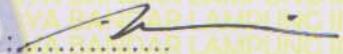
1. Tim Penguji

Tanda Tangan

Ketua :Novi Herawadi Sudibyo,S.Kom.,M.T.I



Anggota : Ari Widiantoko,S.Kom.,M.Tech



2. Dekan Fakultas Ilmu Komputer

Sriyanto,S.Kom.,M.M

NIK 00210800

Tanggal Lulus Ujian Skripsi : 10 maret 2019

HALAMAN PERSEMBAHAN

Bismillahirrahmanirrahiim

Assalamu'alaikum warahmatullahi wabarakatauh

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

1. Terimakasih untuk Ayahanda Wiyono Asnan tercinta yang telah memberikan saya semangat tanpa henti dan membawa saya sampai ke jenjang perkuliahan.
2. Terimakasih untuk Ibunda Teti Resnawati tercinta yang selalu memberikan saya masukan untuk menjalankannya dengan tanpa menyerah..
3. Terimakasih untuk kakak Supriatna dan Adik saya Ardianti Fauziah yang selalu memberikan doa dan dukungannya.
4. Terimakasih untuk paman saya Bambang Santoso yang tidak pernah lelah untuk membantu, menyemangati dan memberi masukan.
5. Seluruh dosen-dosen IIB Darmajaya terimakasih semua, khususnya dosen-dosen Jurusan Sistem Komputer dan Teknik Komputer.
6. Terimakasih buat Almamaterku tercinta IIB Darmajaya.

Wassalamu'alaikum warahmatullahi wabarakatauh.

MOTTO

“Aku menemukan bahwa kehidupan memiliki cara tersendiri untuk memberikan apa yang kita butuhkan. Namun bukan dalam bentuk yang kau inginkan ”

(Orang tua Tercinta)

“Setiap suatu usaha yang kita kerjakan tentunya akan menghasilkan suatu yang memusaskan, karna usaha tidak akan pernah menghinati hasil ketika kita selipkan do'a disetiap usaha yang kita lakukan”

(Sunaryo)

ABSTRAK

PENGONTROL SUHU DAN PEMBERI PAKAN AYAM OTOMATIS DENGAN OUTPUT SMS BERBASIS MIKROKONTROLER

Oleh

Sunaryo

Ayam broiler merupakan jenis ayam hasil dari budidaya teknologi peternakan yang memiliki ciri khas pertumbuhan yang cepat, sebagai penghasil daging dengan konversi pakan yang rendah dan siap dipotong pada usia ayam 28-45 hari dengan berat badan ayam 1,2-1,9 kg / ekor. Umumnya para peternak mereka menggunakan tangan untuk menaburkan pakan pada tilang pakan dan berjalan sepanjang kandang yang mana kandang ayam pedaging yang di ternakan sangatlah luas. Sehingga kegiatan seperti itu bagi peternak ayam sangatlah akan menyita waktu dan tenaga. Serta hal kedua yang harus diperhatikan oleh peternak ayam adalah suhu. Maka peneliti akan merancang suatu alat dengan judul Pengontrol Suhu Dan Pemberi Pakan Ayam Otomatis Dengan Output SMS Berbasis Mikrokontroler. Dalam merancang alat ini peneliti menggunakan beberapa inputan yaitu pertama sensor DHT11, sensor ultrasonik, keypad, RTC yang akan diproses oleh arduino sehingga akan menghasilkan output berupa motor servo, motor DC, GSM Shield dan tampilan LCD 20x4. Hasil dari uji coba sistem keseluruhan dapat diketahui jika sensor ultrasonic 1 dan 2 akan mengukur ketinggian pakan <8 cm yang artinya pakan dalam kondisi penuh sehingga motor servo akan tertutup dan motor DC akan berhenti. Gsm shield akan mengirimkan SMS jika sensor ultrasonic ke 3 yang digunakan sebagai pengukur ketinggian tandon pakan >12 CM Sensor DHT 11 yang digunakan sebagai pengukur suhu kandang akan berkerja menyalakan kipas jika suhu kandang >34° dan jika suhu <34° maka lampu akan menyala.

Kata Kunci : DHT11, Ultrasonik, Ardino dan Ayam

ABSTRACT

AUTOMATIC TEMPERATURE CONTROLLER AND CHICKEN FEEDER USING MICROCONTROLLERBASED SMS OUTPUT

By:
Sunaryo

Broiler chicken is a type of chicken resulting from the cultivation of livestock technology that has characteristic of rapid growth, as a producer of meat with feed conversion low and ready to be cut at the age of 28-45 day chickens with chicken weight 1.2-1.9 kg / tail. Generally, their farmers use their hands to sprinkle feed on speeding ticket and goes along the cage where the broiler cages are on the breed is very wide. Therefore, activities like that for chicken farmers are very consuming time and energy. And the second thing that must be considered by chicken farmers is temperature. Then the researcher designed an instrument with the title Temperature Controller Dan Automatic Chicken Feeder Using Microcontroller-Based SMS Output. In designing this tool the researcher used several inputs namely the first DHT11 sensor, ultrasonic sensor, keypad, RTC which will be processed by Arduino so that it produced serpent motorized output, DC motor, GSM Shield and 20x4 LCD display. The results of the test tried the whole system was known if ultrasonic sensors 1 and 2 measured the height of feed < 8 cm which means the feed is in full condition so the servo motor will closed and the DC motor will stop. The GSM shield will send an SMS if it is censored ultrasonic 3 which is used as a height gauge for feed tendons >12 CM sensor DHT 11 which is used as a cage temperature gauge will work to turn on the fan if the temperature of the enclosure is 34 ° and if the temperature is 34 ° the light will turn on.

Keywords: DHT11, Ultrasonic, Arduino and Chicken



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 “Pengontrol Suhu Dan Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Dengan Output Sms Berbasis Mikrokontroler” 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 khususnya 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 Dr. RZ. Abdul Aziz.,S.T., M.T. Selaku Wakil Rektor Institut Informatika dan Bisnis Darmajaya.
4. Bapak Bayu Nugroho S.Kom.,M.Eng. Selaku Ketua Jurusan Sistem Komputer dan Teknik Komputer, terimakasih atas waktu dan saran yang telah bapak berikan kepada saya.
5. Bapak Novi Herawadi Sudibyo,S.Kom., M.T.I selaku Sekertaris Jurusan Teknik Komputer dan Sistem Komputer, terimakasih atas waktu dan saran yang telah bapak berikan kepada saya.
6. Bapak Bayu Nugroho S.Kom.,M.Eng. dosen pengajar sekaligus sebagai pembimbing saya dalam menyelesaikan laporan Skripsi ini, terima kasih banyak saya ucapkan kepada semoga jasa beliau mendapatkan balasan oleh Allah SWT.

7. Dosen – dosen pengajar khususnya di Jurusan Sistem Komputer dan Teknik Komputer.
8. Ayah dan Ibu tercinta yang selalu memberikan dukungan, semangat dan doa kepada saya.
9. 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 tugas akhir 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, 16 September 2019

Sunaryo
1411060025

DAFTAR ISI

PERNYATAAN ORISINILITAS PENELITIAN	Error! Bookmark not defined.
PERSETUJUAN	Error! Bookmark not defined.
PENGESAHAN	Error! Bookmark not defined.
HALAMAN PERSEMBAHAN	Error! Bookmark not defined.
MOTTO	Error! Bookmark not defined.
ABSTRAK	Error! Bookmark not defined.
ABSTRACT	Error! Bookmark not defined.
KATA PENGANTAR	Error! Bookmark not defined.
DAFTAR ISI	1
DAFTAR TABEL	5
DAFTAR GAMBAR	6
DAFTAR LAMPIRAN	8
BAB I PENDAHULUAN	Error! Bookmark not defined.
1.1 Latar Belakang	Error! Bookmark not defined.
1.2 Rumusan Masalah	Error! Bookmark not defined.
1.3 Ruang Lingkup Penelitian	Error! Bookmark not defined.
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	Error! Bookmark not defined.
1.5 Sistematika Penulisan	Error! Bookmark not defined.
BAB II TINJAUAN PUSTAKA	Error! Bookmark not defined.
2.1 <i>Studi Literatur</i>	Error! Bookmark not defined.
2.2.1 Ayam Broiler	Error! Bookmark not defined.
2.2.2 Pakan Ternak	Error! Bookmark not defined.
2.3 Perangkat Keras Yang Digunakan	Error! Bookmark not defined.
2.3.1 Sensor Suhu DHT 11	Error! Bookmark not defined.
2.3.2 Spesifikasi sensor DHT11	Error! Bookmark not defined.
2.3.3 <i>Ultrasonik HC-SR04</i>	Error! Bookmark not defined.
2.3.4 Real Time Clock (RTC DS1307)	Error! Bookmark not defined.
2.3.5 <i>Keypad</i>	Error! Bookmark not defined.
2.3.6 Relay	Error! Bookmark not defined.

2.3.6.1	Prinsip Kerja Relay	Error! Bookmark not defined.
2.3.6.2	Jenis-jenis Relay	Error! Bookmark not defined.
2.3.6.3	Fungsi-fungsi <i>Relay</i>	Error! Bookmark not defined.
2.3.6.4	Driver Relay.....	Error! Bookmark not defined.
2.3.6.5	<i>Interface Driver Relay</i>	Error! Bookmark not defined.
2.3.7	Motor Servo	Error! Bookmark not defined.
2.3.8	Driver Motor L298N	Error! Bookmark not defined.
2.3.9	Motor DC	Error! Bookmark not defined.
2.3.9.1	Konstruksi Motor DC	Error! Bookmark not defined.
2.3.9.2	Stator Motor DC	Error! Bookmark not defined.
2.3.9.3	Rotor Motor DC.....	Error! Bookmark not defined.
2.3.9.4	Komutator Motor DC.....	Error! Bookmark not defined.
2.3.9.5	Prinsip Kerja Motor DC.....	Error! Bookmark not defined.
2.3.10	<i>Internet Shield SIM900</i>	Error! Bookmark not defined.
2.3.11	<i>SMS (Short Massage Service)</i>	Error! Bookmark not defined.
2.3.12	LCD (Liquid Crystal Display)	Error! Bookmark not defined.
2.3.13	<i>Mikrokontroller</i>	Error! Bookmark not defined.
2.3.14	Modul Arduino AT <i>Mega2560</i>	Error! Bookmark not defined.
2.3.15	Arsitektur Arduino Mega 2560	Error! Bookmark not defined.
2.3.16	Konfigurasi Pin Arduino Mega.....	Error! Bookmark not defined.
2.4	Perangkat Lunak Yang Digunakan.....	Error! Bookmark not defined.
2.4.1	<i>Software</i> Mikrokontroller Arduino Uno.....	Error! Bookmark not defined.
2.4.1.1	Program Arduino Ide	Error! Bookmark not defined.
2.4.1.2	<i>Header</i>	Error! Bookmark not defined.
2.4.1.3	<i>Setup</i>	Error! Bookmark not defined.
2.4.1.4	<i>Loop</i>	Error! Bookmark not defined.
2.4.2	<i>Software</i> ISIS & ARES Proteus 7.0	Error! Bookmark not defined.
BAB III	METODOLOGI PENELITIAN	Error! Bookmark not defined.
3.1	Pengumpulan Data	Error! Bookmark not defined.
3.2	Perancangan Sistem.....	Error! Bookmark not defined.

3.2.1	Alat.....	Error! Bookmark not defined.
3.2.2	Bahan.....	Error! Bookmark not defined.
3.2.3	Perancangan Perangkat Keras	Error! Bookmark not defined.
3.2.3.1	Rangkaian <i>Power Supply</i>	Error! Bookmark not defined.
3.2.3.2	Rangkaian Sensor DHT 11	Error! Bookmark not defined.
3.2.3.3	Rangkaian Sensor Ultrasonik....	Error! Bookmark not defined.
3.2.3.4	Rangkaian RTC DS1307	Error! Bookmark not defined.
3.2.3.5	Rangkaian Keypad.....	Error! Bookmark not defined.
3.2.3.6	Rangkaian <i>Relay</i>	Error! Bookmark not defined.
3.2.3.7	Rangkaian Motor Servo	Error! Bookmark not defined.
3.2.3.8	Rangkaian <i>GSM Shield</i>	Error! Bookmark not defined.
3.2.3.9	Rangkaian <i>LCD (Liquid Crystal Display)</i>	Error! Bookmark not defined.
3.2.3.10	Rangkaian Keseluruhan	Error! Bookmark not defined.
3.2.4	Perancangan Perangkat Lunak	Error! Bookmark not defined.
3.3	Pengujian sistem.....	Error! Bookmark not defined.
3.3.1	Pengujian Program Arduino.....	Error! Bookmark not defined.
3.3.2	Rancangan Pengujian Sensor DHT11	Error! Bookmark not defined.
3.3.3	Rancangan Pengujian Ultrasonik	Error! Bookmark not defined.
3.3.4	Rancangan Pengujian RTC	Error! Bookmark not defined.
3.3.5	Rancangan Pengujian Keypad.....	Error! Bookmark not defined.
3.3.6	Rancangan Pengujian Relay.....	Error! Bookmark not defined.
3.3.7	Rancangan Pengujian Motor Servo...	Error! Bookmark not defined.
3.3.8	Rancangan Pengujian <i>Gsm Shield</i>	Error! Bookmark not defined.
3.3.9	Rancangan Pengujian LCD 20x4	Error! Bookmark not defined.
3.3.10	Pengujian Sistem Keseluruhan.....	Error! Bookmark not defined.
3.4	Analisis Kerja	Error! Bookmark not defined.
BAB IV HASIL DAN PEMBAHASAN		Error! Bookmark not defined.
4.1	Hasil Uji Coba	Error! Bookmark not defined.
4.1.1	Hasil Pengujian dan Pembahasan.....	Error! Bookmark not defined.

4.1.2	Pengujian Sensor DHT 11.....	Error! Bookmark not defined.
4.1.3	Pengujian Sensor Ultrasonik.....	Error! Bookmark not defined.
4.1.4	Pengujian Servo	Error! Bookmark not defined.
4.1.5	Pengujian RTC.....	Error! Bookmark not defined.
4.1.6	Pengujian Setting Keypad.....	Error! Bookmark not defined.
4.1.7	Pengujian <i>GSM Shield</i>	Error! Bookmark not defined.
4.2	Pengujian Sistem Secara Keseluruhan ...	Error! Bookmark not defined.
BAB V KESIMPULAN DAN SARAN.....		Error! Bookmark not defined.
5.1	Kesimpulan	Error! Bookmark not defined.
5.2	Saran.....	Error! Bookmark not defined.
DAFTAR PUSTAKA.....		Error! Bookmark not defined.

DAFTAR TABEL

Tabel 2.1 Kebenaran Untuk 2 Motor	Error! Bookmark not defined.
Tabel 3.1. Alat Yang Dibutuhkan	Error! Bookmark not defined.
Tabel 3.2. Komponen Yang Dibutuhkan	Error! Bookmark not defined.
Tabel 4.1 Pengujian Sensor DHT 11	Error! Bookmark not defined.
Tabel 4.2 Pengujian Sensor Ultrasonik	Error! Bookmark not defined.
Tabel 4.3 Pengujian Motor Servo	Error! Bookmark not defined.
Tabel 4.4. Hasil Pengujian RTC	Error! Bookmark not defined.
Tabel 4.5. Pengujian <i>Setting keypad</i>	Error! Bookmark not defined.
Tabel 4.6. Hasil Pengujian <i>Gsm Shield</i>	Error! Bookmark not defined.
Tabel 4.7. Hasil Pengujian Sistem Keseluruhan	Error! Bookmark not defined.

DAFTAR GAMBAR

Gambar 2.1 Sensor DHT11	Error! Bookmark not defined.
Gambar 2.2 Ultrasonik HC-SR04	Error! Bookmark not defined.
Gambar 2.3. <i>Real Time Clock</i> (RTC DS1307).....	Error! Bookmark not defined.
Gambar 2.4 (a) Bentuk Fisik (b) Rangkaian dasar <i>keypad 4x4</i>	Error! Bookmark not defined.
defined.	
Gambar 2.5 Gambar dan Simbol Relay	Error! Bookmark not defined.
Gambar 2.6 Relay	Error! Bookmark not defined.
Gambar 2.7 Struktur Sederhana Relay	Error! Bookmark not defined.
Gambar 2.8 Jenis Relay berdasarkan <i>Pole</i> dan <i>Throw</i>	Error! Bookmark not defined.
Gambar 2.9 Rangkaian <i>Driver Relay</i>	Error! Bookmark not defined.
Gambar 2.10 Rangkaian <i>Interface Driver Relay</i>	Error! Bookmark not defined.
Gambar 2.11 Motor Servo	Error! Bookmark not defined.
Gambar 2.12. Rangkaian Driver Motor L298.....	Error! Bookmark not defined.
Gambar 2.13 Motor DC Sederhana.....	Error! Bookmark not defined.
Gambar 2.14. Konstruksi Motor DC.....	Error! Bookmark not defined.
Gambar 2.15. Stator Mesin DC.....	Error! Bookmark not defined.
Gambar 2.16 <i>Internet Shield (GSM)</i>	Error! Bookmark not defined.
Gambar 2.17 Bentuk Fisik LCD	Error! Bookmark not defined.
Gambar 2.18 Arduino Mega 2560	Error! Bookmark not defined.
<i>Sumber https://www.arcaelectronica.com</i>).....	Error! Bookmark not defined.
Gambar 2.19 ATMega 2560 pada Arduino Mega 2560	Error! Bookmark not defined.
Gambar 2.20 Tampilan Program <i>Arduino Uno</i>	Error! Bookmark not defined.
Gambar 2.21. Tampilan Software ISIS & ARES Proteus.	Error! Bookmark not defined.
Gambar 3.1 Diagram Alur Penelitian	Error! Bookmark not defined.
Gambar 3.2. Blok Diagram Sistem	Error! Bookmark not defined.
Gambar 3.4 Rangkaian <i>Sensor DHT11</i>	Error! Bookmark not defined.
Gambar 3.5 Script Program <i>Sensor DHT 11</i>	Error! Bookmark not defined.
Gambar 3.6. Rangkaian Sensor Ultrasonik.....	Error! Bookmark not defined.
Gambar 3.7 Script Program <i>Sensor Ultrasonik</i>	Error! Bookmark not defined.
Gambar 3.8 Rangkaian RTC DS1307	Error! Bookmark not defined.
Gambar 3.9 Script Program <i>RTC DS1307</i>	Error! Bookmark not defined.
Gambar 3.10 Rangkaian <i>Keypad</i>	Error! Bookmark not defined.

Gambar 3.11 Script Program <i>Keypad</i>	Error! Bookmark not defined.
Gambar 3.12 Rangkaian <i>Relay</i>	Error! Bookmark not defined.
Gambar 3.13 Script Program <i>Keypad</i>	Error! Bookmark not defined.
Gambar 3.14 Rangkaian Motor Servo	Error! Bookmark not defined.
Gambar 3.15 Script Program <i>Motor Servo</i>	Error! Bookmark not defined.
Gambar 3.16 Rangkaian <i>GSM Shield</i>	Error! Bookmark not defined.
Gambar 3.17 Script Program <i>GSM Shield</i>	Error! Bookmark not defined.
Gambar 3.18. Rangkaian <i>Liquid Crystal Display 20 X 4</i> ..	Error! Bookmark not defined.
Gambar 3.19 Script Program <i>Liquid Crystal Display 20X4</i>	Error! Bookmark not defined.
defined.	
Gambar 3.20 Rangkaian Keseluruhan	Error! Bookmark not defined.
Gambar 3.21 <i>Flowcart</i> Suhu	Error! Bookmark not defined.
Gambar 3.22. <i>Flowcart</i> Pengisian Pakan.....	Error! Bookmark not defined.
Gambar 3.23 Simulai Suhu Dan Pemberi Pakan	Error! Bookmark not defined.
Gambar 3.24 Simulasi Tempat Kandang Ayam	Error! Bookmark not defined.
Gambar. 4.1. Bentuk Fisik Alat	Error! Bookmark not defined.

DAFTAR LAMPIRAN

Lampiran 1. Program	
Lampiran 2. Datasheet Arduino	
Lampiran 3. Datasheet Sensor Ultrasonik.....	
Lampiran 4. Datasheet LCD	
Lampiran 5. Datasheet Motor Servo	
Lampiran 5. Datasheet LM lengkap.....	
Lampiran 6. Datasheet Motor Driver	

BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan Negara Agraris yang sangat subur, Mayoritas penduduknya hidup dari sektor pertanian dan bekerja sebagai petani, pekebun, peternak dan nelayan, salah satu peternakan di Indonesia adalah peternakan ayam pedaging (*Broiler*). Ayam broiler merupakan jenis ayam hasil dari budidaya teknologi peternakan yang memiliki ciri khas pertumbuhan yang cepat, sebagai penghasil daging dengan konversi pakan yang rendah dan siap dipotong pada usia ayam 28-45 hari dengan berat badan ayam 1,2-1,9 kg / ekor, Ayam broiler merupakan hasil dari persilangan antara bangsa ayam *Cornish* yang berasal dari negara inggris dengan ayam *white play mounth rock* dari negara Amerika (M.Iqbal Suriansyah 2016).

Bagi peternak yang memiliki sejumlah besar populasi ayam, dapat menjadi tugas yang sulit untuk menjaga mereka makan sepanjang waktu. Umumnya para peternak ayam masih menggunakan sistim konvensional untuk memberi makan ayam-ayam yang dipelihara. Mereka menggunakan tangan untuk menaburkan pakan pada tilang pakan dan berjalan sepanjang kandang yang mana kandang ayam pedaging yang di ternakan sangatlah luas. Sehingga kegiatan seperti itu bagi peternak ayam sangatlah akan menyita waktu dan tenaga. Serta hal kedua yang harus diperhatikan oleh peternak ayam adalah suhu.

Suhu panas sangatlah penting bagi lingkungan peternakan ayam yang telah menjadi salah satu perhatian utama karena akan dapat menyebabkan kerugian ekonomi akibat peningkatan kematian dan penurunan produktivitas. Keadaan suhu yang *relatif* tinggi pada suatu lingkungan pemeliharaan akan menyebabkan terjadinya cekaman panas. Cekaman panas akan menyebabkan gangguan terhadap pertumbuhan ayam broiler. pertumbuhan ayam akan terganggu yang disebabkan

dengan adanya penurunan konsumsi pakan dan peningkatan konsumsi air minum selama ayam mengalami cekaman panas (Desita Damayanti, 2011).

Banyak penelitian yang sudah melakukan penelitian mengatur suhu dan pemberi pakan secara otomatis, salah satunya yaitu: (Sebayang, 2015). Perancangan Sistem Pengaturan Suhu Kandang Ayam Berbasis Mikrokontroler sebagai pengendali utama, DHT11 sebagai sensor suhu pada kandang dan IC L293D sebagai *driver* motor DC. Mikrokontroler akan memerintahkan motor DC untuk bekerja, apabila suhu yang terukur diatas dari batasan suhu yang ditetapkan dan akan memerintahkan relay untuk menyalakan atau mematikan lampu pijar apabila suhu yang terukur dibawah dari batasan suhu yang telah ditentukan.

(Nur Komala Sarig, 2015) Dengan judul Rancang Bangun Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Berbasis Mikrokontroller System ini merupakan alat kontrol yang mampu memberikan pakan ayam secara otomatis sesuai jadwal. Alat otomasi ini hanya dapat bekerja pada ayam petelur karena ayam tipe ini memiliki kandang *battery individual*. Pengendali utama *system* ini menggunakan Mikrokontroler yang dihubungkan dengan sebuah RTC (*Real Time Clock*) sebagai penyesuai waktu pemberian pakan ayam dengan *real time*.

(Adi Sapto Raharjo,2018) Dengan judul Rancang Bangun Pengendali Dan Pengawasan Gas Amonia Pada Peternakan Ayam Berbasis Arduino 2560 R3 System ini dapat mendeteksi dan mengendalikan kadar gas amonia agar dibawah ambang batas kadar serta memonitoring dan mengendalikan suhu ruangan di peternakan ayam.Pengendali utama System ini menggunakan Arduino sebagai Mikrokontroler,MQ-135 sebagai sensor yang akan mendeteksi kadar gas amonia dan sensor DHT11 yang akan menjadi sensor mendeteksi suhu ruangan kandang ayam. Data yang terdeteksi pada sensor akan disimpan didalam database pada aplikasi monitoring dengan durasi per 1 menit.

Dari permasalahan diatas, maka peneliti ingin membuat sistem “ Pengontrol Suhu Dan Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Dengan Output SMS Berbasis Mikrokontroler”

1.2 Rumusan Masalah

Berdasarkan dari latar belakang yang telah dikemukakan, maka rumusan masalah dalam penelitian ini, yaitu:

1. Bagaimana cara menjaga suhu kandang ayam broiler agar tetap stabil?
2. Bagaimana cara meringankan beban pekerja dalam pemberian pakan ayam broiler yang dapat menyita waktu dan tenaga?
3. Bagaimana cara agar pekerja dapat dengan mudah mengontrol suhu dalam ruangan kandang, dan jadwal waktu pemberian pakan ayam broiler?
4. Bagaimana cara agar pekerja kandang dapat mengetahui informasi mengenai pemberitahuan ketika tandon pakan telah habis?

1.3 Ruang Lingkup Penelitian

Agar Penelitian ini tidak meluas, maka ruang lingkup penelitian yang diatur yakni:

1. Mikrokontroler yang digunakan dalam system ini yaitu menggunakan Arduino Atmega 2560 sebagai kendali utama sistem otomasi.
2. Sensor jarak yang digunakan dalam sistem ini yaitu sensor ultrasonic dan sensor suhu yang digunakan dalam system ini yaitu sensor DHT11.
3. Heater digunakan sebagai penghangat suhu ruangan kandang dan blower digunakan sebagai sirkulasi keluar masuknya udara agar suhu tetap terjaga.
4. Dalam penjadwalan pemberi pakan menggunakan RTC DSI307 dan Penjadwalan pemberi pakan ayam hanya dua kali dalam sehari.
5. Dalam penelitian ini hanya membuat pengontrol suhu dan pemberi pakan otomatis untuk peteranakan ayam broiler.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini yang hendak dicapai adalah sebagai berikut:

1. Untuk merancang system yang dapat Mengontrol suhu dan pemberi pakan ternak.
2. Dapat merancang system yang dapat memberikan informasi pemberitahuan ketika wadah tandon pakan telah habis.

3. Dapat membuat alat untuk pengatur suhu dalam ruangan kandang ayam, serta dapat membuat alat untuk pengatur jadwal pemberian pakan ayam pada wadah pakan yang telah habis.

1.4 Manfaat Penelitian

1. Manfaat dari penelitian ini adalah dapat memberikan masukan kepada peternak ayam, tentang teknologi yang lebih menguntungkan bagi dunia peternakan ayam *broiler*.
2. Menghemat waktu dan dapat meningkatkan hasil produksi dalam pemeliharaan ayam *broiler*..
3. Dapat menjadi bahan acuan bagi peternak ayam *broiler*, dalam melakukan penelitian tentang pengontrol suhu dan pemberi pakan ayam secara otomatis.

1.5 Sistematika Penulisan

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

1. BAB I PENDAHULUAN

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

2. BAB II TINJAUAN PUSTAKA

Bab ini berisikan tentang teori – teori yang berkaitan dengan “Pengontrol Suhu Dan Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Dengan Output SMS Berbasis Mikrokontroler”.

3. 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.

4. BAB IV HASIL DAN PEMBAHASAN

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

5. BAB V KESIMPULAN DAN SARAN

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

6. DAFTAR PUSTAKA

7. LAMPIRAN

BAB II

TINJAUAN PUSTAKA

2.1 Studi Literatur

Penelitian tentang pengontrol suhu dan pemberi pakan secara otomatis sudah pernah dilakukan oleh beberapa peneliti. Ringkasan *Studi Literatur* yang dilakukan untuk mengetahui sejauh mana penelitian yang sudah ada.

Penelitian yang dilakukan oleh Rahmawanto Rahmawano, R. Arif Tri pada tahun 2014 berjudul Rancang Bangun Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Berbasis Mikrokontroler dengan Pengendali utama system ini menggunakan Mikrokontroler yang dihubungkan dengan sebuah RTC (*Real Time Clock*) sebagai penyesuai waktu pemberian pakan ayam dengan real time. Kekurangan alat ini Tidak ada waktu yang tepat dalam menentukan putaran roda Motor Servo dan DC, sehingga ditambahkan limit switch pada alat otomasi ini.

Selanjutnya penelitian dengan judul Rancang Bangun Sistem Pemberian Pakan Ternak (Sapi) Dan Pengadukannya Secara Otomatis Berbasis Mikrokontroler dilakukan pada tahun 2018 oleh Ardiansyah dengan tujuan merancang sistem pemberi pakan ternak ini dapat meringankan pekerjaan peternak masalah manajemen waktu. Sistem ini juga dapat membantu agar bahan pakan yang digunakan bisa seefisien. Hasil dari penelitian ini yaitu memiliki nilai keakuratan alat dalam pemberian pakan setiap harinya mencapai 90,86% sedangkan standar pemberian pakan untuk mencapai berat badan ideal sapi adalah 80% keatas.

Selanjutnya penelitian dengan judul Rancang bangun alat pemberi pakan ikan Otomatis berbasis mikrokontroler dilakukan pada tahun 2015 oleh Hendra dengan dengan sistem kerja Sensor photodiode, Keypad dan RTC input dari alat ini, sedangkan pemrosesnya adalah Mikrokontroler ATmega16, Motor servo sebagai outputnya. Kelebihannya adalah alat ini akan otomatis.

mampu mengirimkan sms pemberitahuan ketika pakan telah diberikan dan ketika tampungan dalam keadaan kosong atau habis.

Selanjutnya peneliti dengan judul Perancangan dan implementasi alat pemberi makan ikan otomatis berbasis mikrokontroler at89s52 oleh Haryanto pada tahun 2016 dengan memanfaatkan IC ULN2003 untuk inputan mikrokontroler AT89S52 untuk pemroses serta outputnya penaburan makanan ke kolam dan LCD. alat pemberi makan ikan otomatis dirancang untuk selalu menutup lubang saluran makanan saat *motor* berhenti berputar, sehingga kondisi terbukanya saluran makanan saat *motor* berhenti berputar tidak akan terjadi dari kelebihan.

Selanjutnya peneliti dengan judul Perangkat pemberi pakan otomatis Pada kolam budidaya oleh Yenni pada tahun 2016 dengan sistem kerja RTC untuk inputan Mikrokontroler Atmega 2560 untuk memproses serta Motor dc, Lcd. Kelebihan dari alat ini adalah mampu bertahan dengan kondisi cuaca panas dan hujan karena peralatan sudah dibuat menggunakan bahan yang tidak mudah rapuh dan komponen elektroniknya dimuat dalam wadah yang tertutup rapat sehingga tidak terkena air ketika pada musim hujan dan panas.

2.2 Landasan Teori

2.2.1 Ayam Broiler

Ayam broiler atau ayam pedaging adalah ayam yang mempunyai kemampuan hidup yang tinggi dan mampu mengubah pakan menjadi daging secara efisien. Pada umumnya ayam ini siap panen pada usia 28-45 hari dengan berat badan 1,2-2 kg/ekor. Suhu lingkungan mempengaruhi pertumbuhan ayam. Pada prinsipnya pertumbuhan dan efisiensi penggunaan makanan yang maksimum tidak dapat dicapai, bila ayam dipelihara di bawah atau di atas suhu lingkungan yang tidak sesuai. Pada suhu 34°C, ayam mengalami kesulitan dalam membuang panas, terutama jika diikuti dengan kelembaban yang tinggi dalam keadaan demikian ayam tidak dapat lagi membuang panasnya, sehingga suhu tubuh cenderung melambung. Pada saat hewan sudah tidak mampu lagi mempertahankan *homeoterm*, hewan akan mereduksi produksi panas dengan menggunakan

mekanisme fisiologis internal untuk mengupayakan pengaturan keseimbangan panas menjadi lebih baik kembali. Konsumsi pakan dan *sekresi hormone termogenik* akan mengalami penurunan, untuk mengurangi *metabolisme* basal yang akan diikuti dengan adanya penurunan produktivitas. Jika semua mekanisme *fisiologis* tersebut gagal untuk memperbaiki atau mengembalikan keseimbangan muatan panas tubuh maka suhu tubuh hewan akan meningkat dan hewan tersebut memasuki fase akut

2.2.2 Pakan Ternak

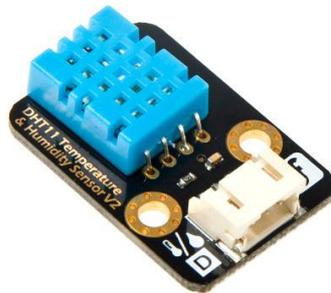
Pakan Ternak adalah semua bahan pakan yang bisa diberikan dan bermanfaat bagi ternak serta tidak menimbulkan pengaruh negatif terhadap tubuh ternak. Pakan yang diberikan harus berkualitas tinggi, yaitu mengandung zat-zat yang diperlukan oleh tubuh ternak dalam hidupnya seperti air, karbohidrat, lemak, protein. Pakan sendiri merupakan komoditi yang sangat penting bagi ternak. Zat-zat nutrisi yang terkandung dalam pakan dimanfaatkan oleh ternak untuk memenuhi kebutuhan hidup pokok dan produksi ternak itu sendiri. Selain itu, pakan juga merupakan dasar bagi kehidupan yang secara terus menerus berhubungan dengan kimiawi tubuh dan kesehatan. Dalam pemberiannya pakan harus sesuai dengan kebutuhan tubuh ternak tersebut. Bahan pakan merupakan bahan makanan ternak yang terdiri dari bahan kering dan air yang harus diberikan kepada ternak untuk memenuhi kebutuhan hidup pokok dan produksinya. Setiawan (2005).

2.3 Perangkat Keras Yang Digunakan

2.3.1 Sensor Suhu DHT 11

Sensor DHT11 adalah salah satu sensor yang dapat mengukur dua parameter lingkungan sekaligus, yakni suhu dan kelembaban udara (humidity). Dalam sensor ini terdapat sebuah thermistor tipe NTC (*Negative Temperature Coefficient*) untuk mengukur suhu, sebuah sensor kelembaban tipe resistif dan sebuah mikrokontroler 8 bit yang mengelola kedua sensor tersebut dan mengirim hasilnya ke pin *output* dengan format *single-wire bi-directional* (kabel tunggal dua arah).

(Ajie, 2016). Berikut adalah bentuk fisik dari sensor suhu dan kelembaban DHT11.



Gambar 2.1 Sensor DHT11

(Sumber: <http://encrypted-tbn.gstatic.com/>)

2.3.2 Spesifikasi sensor DHT11

1. Tegangan: 5 V
2. Rentang temperatur :0-50 ° C kesalahan ± 2 ° C
3. Kelembaban :20-90% RH ± 5 % RH error
4. Interface: Digital

2.3.3 Ultrasonik HC-SR04



Gambar 2.2 Ultrasonik HC-SR04

(Sumber <https://www.elektronka.com>,2015)

Sensor HC-SR04 adalah sensor pengukur jarak berbasis gelombang ultrasonik. Prinsip kerja sensor ini mirip dengan radar ultrasonik. Gelombang ultrasonik di pancarkan kemudian di terima balik oleh receiver ultrasonik. Jarak antara waktu pancar dan waktu terima adalah representasi dari jarak objek. Sensor ini cocok untuk aplikasi elektronik yang memerlukan deteksi jarak termasuk untuk sensor

pada robot. Sensor HC-SR04 adalah versi low cost dari sensor ultrasonic PING buatan parallax. Perbedaannya terletak pada pin yang digunakan. HC-SR04 menggunakan 4 pin sedangkan PING buatan parallax menggunakan 3 pin. Pada Sensor HC-SR04 pin trigger dan output diletakkan terpisah. Sedangkan jika menggunakan PING dari Parallax pin trigger dan output telah diset default menjadi satu jalur. Tidak ada perbedaan signifikan dalam pengimplementasiannya. Jangkauan karak sensor lebih jauh dari PING buatan parallax, dimana jika ping buatan parallax hanya mempunyai jarak jangkauan maksimal 350 cm sedangkan sensor HC-SR04 mempunyai kisaran jangkauan maksimal 400-500cm.

1. Jangkauan deteksi: 2cm sampai kisaran 400 -500cm
2. Sudut deteksi terbaik adalah 15 derajat
3. Tegangan kerja 5V DC
4. Resolusi 1cm
5. Frekuensi Ultrasonik 40 kHz
6. Dapat dihubungkan langsung ke kaki mikrokontroler

2.3.4 Real Time Clock (RTC DS1307)

Real Time Clock merupakan suatu chip (IC) yang memiliki fungsi sebagai penyimpan waktu dan tanggal. DS1307 merupakan Real-time clock (RTC) yang dapat menyimpan data-data detik, menit, jam, tanggal, bulan, hari dalam seminggu, dan tahun valid hingga 2100. 56-byte, battery-backed, RAM *nonvolatile* (NV) RAM untuk penyimpanan. DS1307 merupakan Real-time clock (RTC) dengan jalur data parallel yang memiliki Antarmuka serial Two-wire (I2C), Sinyal luaran gelombang-kotak terprogram (*Programmable squarewave*), Deteksi otomatis kegagalan-daya (*power-fail*) dan rangkaian *switch*, Konsumsi daya kurang dari 500nA menggunakan mode baterai cadangan dengan operasional osilator. Tersedia fitur industri dengan ketahanan suhu: -40°C hingga +85°C. Tersedia dalam kemasan 8-pin DIP atau SOIC.



Gambar 2.3. Real Time Clock (RTC DS1307)

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

Berikut Penjelasan Pin-Pin Pada IC DS1307.

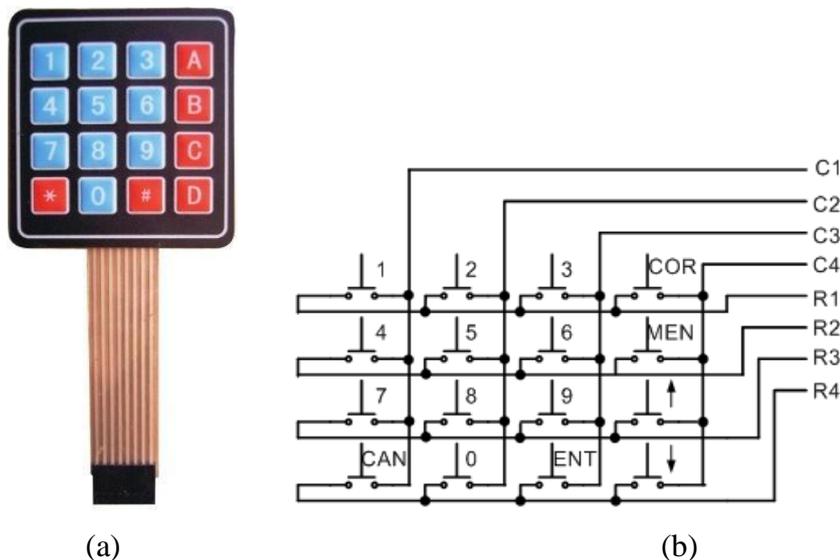
1. X1 Merupakan pin yang digunakan untuk dihubungkan dengan kristal sebagai pembangkit clock.
2. X2 Berfungsi sebagai keluaran / output dari crystal yang digunakan. Trhubung juga dengan X1.
3. VBAT Merupakan backup supply untuk RTC dalam menjalankan fungsi waktu dan tanggal. Besarnya adalah 3V dengan menggunakan jenis Lithium Cell atau sumber energy lain. Jika pin ini tidak di gunakan maka harus terhubung dengan Ground. Sumber tegangan dengan 48mAH atau lebih besar dapat digunakan sebagai cadangan energy sampai lebih dar 10 tahun, namun dengan persyaratan untuk pengoprasian dalam suhu 25°C.
4. GND Berfungsi sebagai *Ground*.
5. SDA Barfungsi sebagai masukan / keluaran (I/O) untuk I2C serial interface. Pin ini bersifat open drain, oleh sebab itu membutuhkan eksternal pull up resistor.
6. SCL Berfungsi sebagai clock untuk input ke I2C dan digunakan untuk mensinkronisasi pergerakan data dalam serial *interface*. bersifat open drain, oleh sebab itu membutuhkan *eksternal pull up resistor*.
7. SWQ/OUT Sebagai square wafe / Output Driver . jika di aktifkan, maka akan menjadi 4 *frekuensi* gelombang kotak yaitu 1 Hz, 4kHz, 8kHz, 32kHz sifat dari pin ini sama dengan sifat pin SDA dan SCL sehingga membutuhkan

eksternal pull up resistor. Dapat dioperasikan dengan VCC maupun dengan VBAT.

8. VCC Merupakan sumber tegangan utama. Jika sumber tegangan terhubung dengan baik, maka pengaksesan data dan pembacaan data dapat dilakukan dengan baik. Namun jika *backup supply* terhubung juga dengan VCC, namun besar VCC di bawah VTP, maka pengaksesan data tidak dapat dilakukan.

2.3.5 Keypad

Keypad sering digunakan sebagai suatu input pada beberapa peralatan yang berbasis mikroprocessor atau mikrokontroler. *Keypad* terdiri dari sejumlah saklar, yang terhubung sebagai baris dan kolom dengan susunan seperti yang ditunjukkan pada gambar 2.4. Agar mikrokontroler dapat melakukan *scan keypad*, maka port mengeluarkan salah satu *bit* dari 4 *bit* yang terhubung pada kolom dengan logika *low* “0” dan selanjutnya membaca 4 bit pada baris untuk menguji jika ada tombol yang ditekan pada kolom tersebut. Sebagai *konsekuensi*, selama tidak ada tombol yang ditekan, maka mikrokontroler akan melihat sebagai logika *high* “1” pada setiap pin yang terhubung ke baris.



Gambar 2.4 (a) Bentuk Fisik (b) Rangkaian dasar keypad 4x4

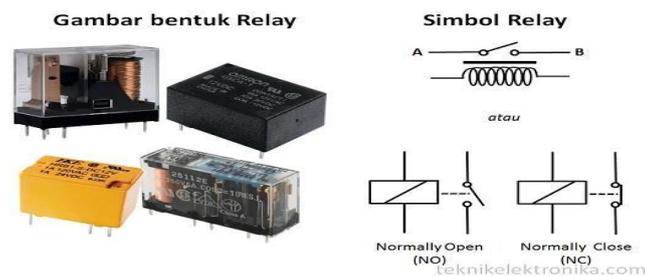
(Sumber <https://digitalmeans.co.uk/shop/image/cache/catalog/dfrobot/>)

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).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 2.5 Gambar dan Simbol Relay

[\(http://teknikelektronika.com/pengertian-relay-fungsi-relay/\)](http://teknikelektronika.com/pengertian-relay-fungsi-relay/)



Gambar 2.6 Relay

(Sumber : Kilian, Christopher T, Modern Control Technology 2016)

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 : menguatkan arus atau tegangan.
3. Pengatur logika kontrol suatu sistem. Susunan kontak pada *relay* adalah:
4. *Normally Open* : *Relay* akan menutup bila dialiri arus listrik.
5. *Normally Close* : *Relay* akan membuka bila dialiri arus listrik.
6. *Changeover* : *Relay* ini memiliki kontak tengah yang akan melepaskan diri dan membuat kontak lainnya berhubungan.

2.3.6.1 Prinsip Kerja Relay

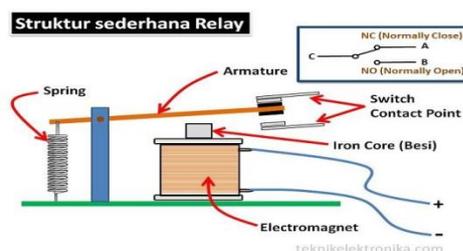
Pada dasarnya, *Relay* terdiri dari 4 komponen dasar yaitu :

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 2.7 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 lainnya 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.

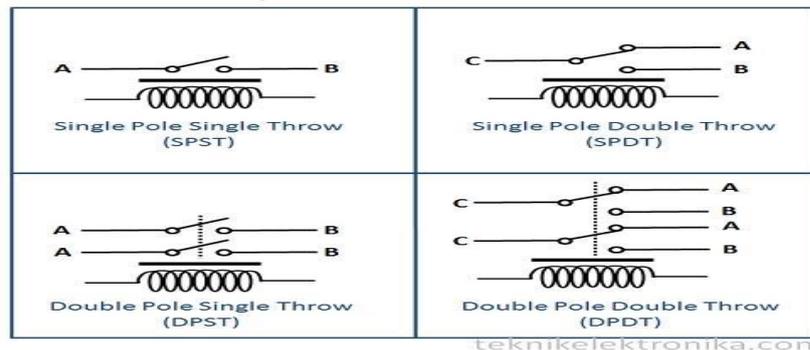
2.3.6.2 Jenis-jenis Relay

Berikut ini penggolongan relay berdasar jumlah *pole dan throw* :

1. DPST (*Double Pole Single Throw*), relay golongan ini memiliki 6 terminal, diantaranya 4 terminal yang terdiri dari 2 pasang terminal saklar sedangkan
2. terminal lainnya untuk *coil*. Relay DPST dapat dijadikan 2 saklar yang dikendalikan oleh 1 *coil*.
 - A. SPST (*Single Pole Single Throw*), relay golongan ini memiliki 4 terminal, 2 terminal untuk saklar dan 2 terminalnya lagi untuk *coil*.
 - B. SPDT (*Single Pole Double Throw*), relay golongan ini memiliki 5 terminal, 3 terminal untuk saklar dan 2 terminalnya lagi untuk *coil*.
 - C. DPDT (*Double Pole Double Throw*), relay golongan ini memiliki terminal sebanyak 8 terminal, diantaranya terminal yang merupakan 2 pasang relay SPDT yang dikendalikan oleh 1 (*single*) *coil*. Sedangkan 2 terminal lainnya untuk *coil*.

Selain golongan *relay* diatas, terdapat juga *relay-relay* yang *Pole* dan *Throw*-nya melebihi dari dua. Misal-nya 3PDT (*Triple Pole Double Throw*) ataupun 4PDT (*Four Pole Double Throw*) dan lain sebagainya.

Berikut ini merupakan gambar dari jenis *Relay* berdasarkan *Pole* dan *Throw*-nya :



Gambar 2.8 Jenis Relay berdasarkan Pole dan Throw

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

2.3.6.3 Fungsi-fungsi Relay

Beberapa fungsi *relay* yang telah umum diaplikasikan kedalam peralatan elektronika diantara-nya adalah :

1. *Relay* digunakan untuk menjalankan fungsi logika (*logic function*).
2. *Relay* digunakan untuk memberikan fungsi penundaan waktu (*time delay function*).
3. *Relay* digunakan untuk mengendalikan sirkuit tegangan tinggi dengan bantuan dari signal tegangan rendah.
4. Ada juga *relay* yang berfungsi untuk melindungi motor ataupun komponen lainnya dari kelebihan tegangan ataupun hubung singkat (*short*).

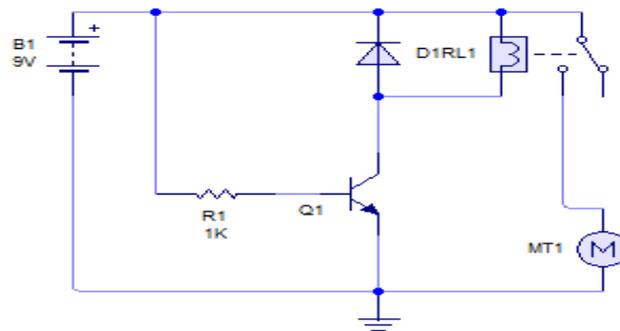
2.3.6.4 Driver Relay

Rangkaian *driver relay* berfungsi untuk mengendalikan motor arus searah (dc) yang dihasilkan dari *port* paralel I/O. Sinyal dari keluaran port biasanya berupa sinyal-sinyal yang kecil, sehingga tidak mampu untuk menggerakkan sistem daya berupa motor arus searah. Untuk dapat dimanfaatkan sinyal keluaran port, diperlukan suatu rangkaian *driver relay* agar sinyal yang kecil dapat

dipergunakan untuk penggerak objek yang akan dikendalikan dari jarak jauh. Rangkaian *driver relay* ini dibangun oleh suatu komponen utama yaitu transistor dan *relay*. Transistor di rangkain *driver relay* difungsikan sebagai penguat sinyal dan *switching*, serta *relay* sebagai penggerak motor dc. *Driver relay* ini selain sebagai penguat dan *switching*, sekaligus difungsikan untuk mengendalikan motor dc dalam sistem pembalik putaran. Jadi, *driver relay* ini dapat mengatur arah putaran motor *forward* dan *reverse*. Semua *driver relay* pada sistem ini memiliki rangkaian dan karakteristik yang sama. Saat *relay* 1 bekerja maka posisi positif motor akan mendapat sumber tegangan positif dan posisi negatif motor terhubung dengan kutub negatif sumber tegangan. Sehingga, motor akan berputar dengan arah putaran searah jarum jam (*clockwise*). Dengan cara yang sama untuk menggerakkan kontak *relay* 2, maka terjadi kondisi yang berkebalikan yaitu motor akan berputar dengan arah putaran yang berlawanan arah jarum jam (*counter clockwise*).

Penggunaan *driver relay* ini menjadi pilihan karena *driver relay* mudah dikontrol, dapat diberi beban yang besar baik beban AC maupun DC serta sebagai isolator yang baik antara rangkaian beban dengan rangkaian kendali. Rangkaian *driver relay* dapat dibangun menggunakan konsep transistor sebagai saklar. Teknik antara *relay* dengan rangkaian digital atau mikrocontroller adalah rangkaian *driver relay* dengan menggunakan transistor sebagai penguat.

Berikut merupakan contoh dari gambar rangkaian *Driver Relay* :



Gambar 2.9 Rangkaian Driver Relay

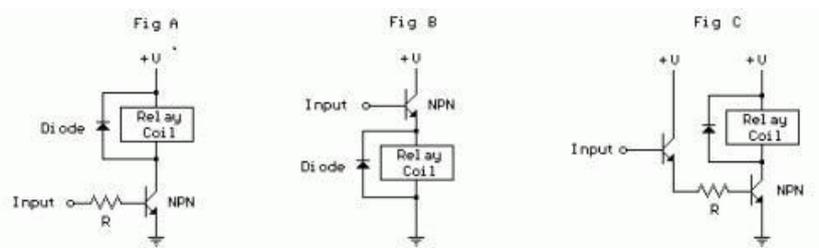
(Sumber <https://rangkaian-driver-dengan-relay-robot.com>)

Pada rangkaian menyerupai sirkuit diatas, dapat dilihat untuk mengoperasikan transistor sebagai saklar transistor dalam keadaan sepenuhnya "OFF" (*cut-off*) atau dalam keadaan "ON" (saturasi). Namun, ketika dinyalakan dalam kondisi ON (saturasi) , maka aliran arus maksimum. Dalam prakteknya ketika transistor diaktifkan "OFF", arus kebocoran akan kecil ketika mengalir melalui transistor dan ketika diaktifkan "ON" maka rangkaian tersebut akan memiliki tegangan saturasi kecil (V_{CE}) Meskipun transistor tidak dalam saklar yang sempurna, baik di *cut-off* dan daerah saturasi. Agar arus Basis mengalir, terminal input Basis harus dibuat lebih positif daripada Emitter dengan meningkatkan itu di atas 0,7 volt yang dibutuhkan untuk perangkat silikon. Dengan memvariasikan *Base-Emitter* ini tegangan V_{BE} arus basis juga mengontrol jumlah arus kolektor yang mengalir melalui transistor.

Ketika arus kolektor maksimum mengalir maka transistor dikatakan saturasi. Nilai dari resistor Basis menentukan berapa banyak masukan tegangan yang diperlukan dan sesuai saat Basis untuk beralih transistor sepenuhnya "ON". Transistor BC108 adalah transistor umum NPN *bipolar junction* (BJT) digunakan untuk memperkuat daya rendah atau aplikasi *switching*. Hal ini dirancang untuk arus yang rendah sampai medium, daya yang rendah , tegangan medium, dan dapat beroperasi pada kecepatan yang cukup tinggi.

2.3.6.5 Interface Driver Relay

Penggunaan *relay* sering menjadi pilihan karena *relay* mudah dikontrol, *relay* dapat diberi beban yang besar baik beban AC maupun DC, dan sebagai isolator yang baik antara rangkaian beban dengan rangkaian kendali. Rangkaian *interface relay* dapat dibangun menggunakan konsep transistor sebagai saklar. Transistor yang digunakan untuk *driver relay* dapat dikonfigurasi dengan *common emitor*, *emitor follower* atau transistor *darlington*. Teknik *interface* antara *relay* dengan rangkaian digital atau rangkaian *microcontroller* dapat dilihat pada gambar dibawah ini :



Gambar 2.10 Rangkaian Interface Driver Relay

(<http://elektronika-dasar.web.id/teori-elektronika/interface-relay-ke-rangkaian-digital>)

Rangkaian *interface* antar *relay* dengan rangkaian digital pada gambar diatas ada 3 jenis interface yang dapat digunakan. Bagian dan fungsi komponen dari rangkaian *interface relay* diatas sebagai berikut :

1. Rangkaian pada gambar A, rangkaian pada gambar A tersebut menggunakan mode *common emitor*, apabila basis mendapat sinyal *input* logika 1 (sumber tegangan positif) maka transistor pada gambar A akan mendapat bias maju, sehingga transistor ON dan memberikan sumber tegangan ke *relay* dan *relay* menjadi ON.
2. Rangkaian pada gambar B adalah *interface relay* yang menggunakan transistor teknik *emior folower* dimana *relay* diletakan pada kaki emitor trnasistor. Fungsi dioda yang dipasangkan pada rangkaian *interface* tersebut digunakan untuk menyerap tegangan induksi yang dihasilkan oleh *relay*.

3. Rangkaian pada gambar C merupakan teknik *inteface relay* ke rangkaian digital menggunakan transistor yang dirangkai secara *darlington*.

2.3.7 Motor Servo

Motor servo adalah sebuah motor DC yang dilengkapi rangkaian kendali dengan sistem closed feedback yang terintegrasi dalam motor tersebut. Pada motor servo posisi putaran sumbu (axis) dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Contoh motor servo, teori motor servo, definisi motor servo, bentuk motor servo, dasar teori motor servo, pengertian motor servo, analisa motor servo. Motor servo disusun dari sebuah motor DC, gearbox, variabel resistor (VR) atau potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas maksimum putaran sumbu (axis) motor servo. Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang pada pin kontrol motor servo.

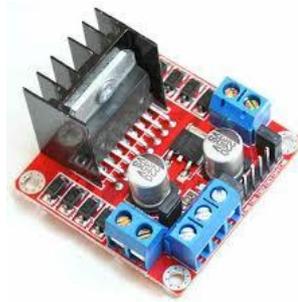


Gambar 2.11 Motor Servo

(Sumber <http://sainsdanteknologiku.blogspot.co.id/2017>)

2.3.8 Driver Motor L298N

L298 adalah driver motor berbasis H-Bridge, mampu menangani beban hingga 4A pada tegangan 6V – 46V. Dalam chip terdapat dua rangkaian H-Bridge. Selain itu driver ini mampu mengendalikan 2 motor sekaligus dengan arus beban 2 A. berikut gambar rangkaian driver motor L298.



Gambar 2.12. Rangkaian Driver Motor L298

(Sumber <http://elektronika.blogspot.co.id/2017>)

Rangkaian driver motor, untuk output motor DC digunakan dioda, hal ini ditujukan agar driver motor dapat menahan arus balik yang datang dari motor DC. Input driver motor berasal dari mikrokontroler utama, untuk MOT 1A dan MOT 1B untuk menggerakkan motor 1, ENABLE 1 untuk mengatur kecepatan motor 1 menggunakan PWM, selanjutnya untuk MOT 2A dan MOT 2B untuk menggerakkan motor 2, ENABLE 2 untuk mengatur kecepatan motor 2 menggunakan PWM.

Tabel 2.1 Kebenaran Untuk 2 Motor

MOT 1A	MOT 1B	ENB 1	MOT 2A	MOT 2B	ENB 2	GERAK
H	L	H	H	L	H	Maju
L	H	H	L	H	H	Mundur
H	L	H	L	L	H	Belok kanan
L	L	H	H	L	H	Belok kiri

Nah dari rangkaian driver motor DC selanjutnya kita ke rangkaian driver motor H-Bridge MOSFET. Tapi sebelum itu saya akan menjelaskan sedikit tentang H-Bridge. Secara konsep rangkaian ini terdiri dari 4 saklar yang tersusun sedemikian rupa sehingga memungkinkan motor dapat teraliri arus dengan arah yang berkebalikan. Yaitu searah jarum jam dan berlawanan arah jarum jam. Pada rangkaian driver motor ini, saklar-saklar tersebut digantikan oleh transistor atau

MOSFET yang dikerjakan pada daerah saturasi dan cut-off (Switch). Berikut cara kerja dari H-Bridge motor.

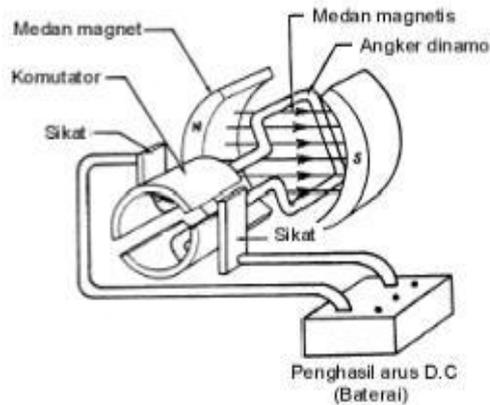
Dari Gambar diatas berikut H-Bridge bekerja:

1. Ketika S1 dan S4 tertutup (diagonal) dan lainnya terbuka maka arus akan mengalir dari battery ke kutub positif motor kemudian keluar ke kutub negatif motor, maka motor akan berputar kearah kanan.
2. Ketika S2 dan S3 tertutup (diagonal) dan lainnya terbuka, maka arus akan mengalir sebaliknya, motor juga akan berputar kearah sebaliknya.
3. Jika semua saklar tertutup, maka motor akan berhenti, dan jika ini diteruskan maka akan menyebabkan rangkaian menjadi "short circuit".

Dari Rangkaian diatas hanya menggunakan 1 pin direction untuk memutar motor yaitu jika diberi logika low (0) maka arahnya CCW dan sebaliknya jika logika high (1) maka arahnya CW.

2.3.9 Motor DC

Motor listrik merupakan perangkat elektromagnetis yang mengubah energi listrik menjadi energi mekanik. Energi mekanik ini digunakan untuk, misalnya memutar *impeller* pompa, *fan*, atau blower, menggerakkan kompresor, mengangkat bahan, dll. Motor listrik kadangkala disebut "kuda kerja" nya industri sebab diperkirakan bahwa motor – motor menggunakan sekitar 70% beban listrik total di industri. Gerakan motor didasarkan pada prinsip bahwa pada saat penghantar berarus ditempatkan di dalam medan magnet, gaya mekanik muncul pada penghantar. Arahnya ditentukan oleh kaidah tangan Fleming sehingga penghantar bergerak pada arah gaya. Jika motor dihubungkan dengan sumber arus searah, arus searah mengalir melalui sikat dan komutator menuju lilitan jangkar. Saat arus melewati komutator, arus diubah menjadi tegangan bolak – balik sehingga kelompok penghantar pada kutub medan yang berturutan dialiri arus pada arah yang berlawanan. Bentuk motor paling sederhana memiliki kumparan satu lilitan yang bisa berputar bebas di antara kutub – kutub magnet permanen.



Gambar 2.13 Motor DC Sederhana

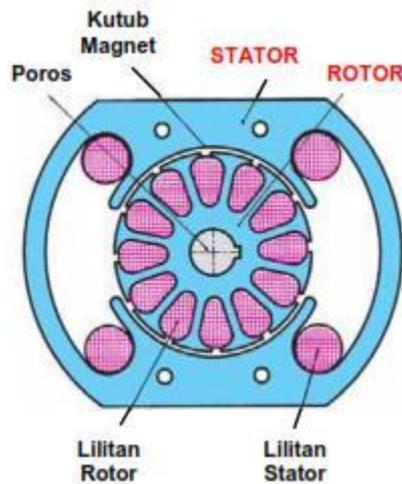
(Sumber <http://sainsdanteknologiku.blogspot.co.id/2017>)

Mekanisme kerja untuk seluruh jenis motor secara umum :

1. Arus listrik dalam medan magnet akan memberikan gaya.
2. Jika kawat yang membawa arus dibengkokkan menjadi sebuah lingkaran/ *loop*, maka kedua sisi loop, yaitu pada sudut kanan medan magnet, akan mendapatkan gaya pada arah yang berlawanan.
3. Pasangan gaya menghasilkan tenaga putar/torque untuk memutar kumparan.
4. Motor – motor memiliki beberapa loop pada dinamonya untuk memberikan tenaga putaran yang lebih seragam dan medan magnetnya dihasilkan oleh susunan elektromagnetik yang disebut kumparan medan.

2.3.9.1 Konstruksi Motor DC

Bagian utama mesin listrik terdiri dari dua bagian: yaitu bagian bergerak disebut Rotor, dan bagian diam yang disebut Stator. Masing – masing bagian mempunyai lilitan kawat. Pada Stator, lilitan kawat berfungsi sebagai pembangkit medan magnet, sedangkan pada Rotor, lilitan berfungsi sebagai pembangkit gaya gerak listrik.

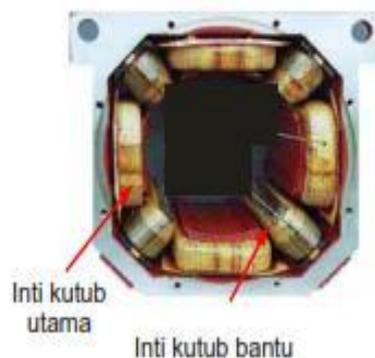


Gambar 2.14. Konstruksi Motor DC

(Sumber <http://sainsdanteknologiku.blogspot.co.id/2017>)

2.3.9.2 Stator Motor DC

Sebuah mesin DC terdiri dari bagian stator, yang terdiri dari set-magnet dengan cincin baja dan lilitan kawat yang menonjol dengan inti kutub utama, sepatu kutub yang terbuat dari lempeng-elektro serta lilitan kawat penguat eksitasi dan inti-kutub bantu. Kontruksi ini biasanya pada mesin DC berdaya maksimum 20 kW. Mesin jenis ini akan bekerja sepanjang ada magnetisasi. Untuk mesin dengan daya hingga 1 kW, terdiri dari sebuah komutator berkutub utama, yang terbuat dari baja atau lempeng elektro dengan lilitan kawat. Sepatu-sepatu kutub dari kutub utama terdapat lilitan kompensasi.



Gambar 2.15. Stator Mesin DC

(Sumber <http://sainsdanteknologiku.blogspot.co.id/2017>)

2.3.9.3 Rotor Motor DC

Bagian rotor (pada mesin DC seringkali disebut jangkar) terbuat dari poros baja beralur dan lilitan kawat pada alur-alur tersebut. sikat arang (*carbon brush*) adalah bagian dari stator. Sikat ini ditahan oleh pemegang sikat (*brush holder*).

2.3.9.4 Komutator Motor DC

Sebuah komutator terdiri dari segmen-segmen tembaga, dimana setiap ujungnya disambungkan dengan ujung lilitan rotor. Komutator adalah bagian mesin listrik yang perlu sering dirawat dan dibersihkan. Bagian ini bersinggungan dengan sikat arang untuk memasukkan arus dari jala - jala ke rotor.

2.3.9.5 Prinsip Kerja Motor DC

Motor – motor DC pada awalnya membutuhkan momen gerak (gaya torsi) yang besar dan tidak memerlukan kontrol kecepatan putar. Kecepatan putar motor selanjutnya akan dikontrol oleh medan magnet. Pada motor DC dengan penguat terpisah, sumber eksitasi didapat dari luar, misalnya dari aki. Terjadinya gaya torsi pada jangkar disebabkan oleh hasil interaksi dua garis medan magnet. Kutub magnet menghasilkan garis medan magnet dari utara-selatan melewati jangkar. Lilitan jangkar yang dialiri arus listrik DC menghasilkan magnet dengan arah ke kiri.

2.3.10 *Internet Shield SIM900*



Gambar 2.16 *Internet Shield (GSM)*
(Sumber [https:// www.elektronika.cc.com](https://www.elektronika.cc.com))

Arduino *GSM Shield V2* menghubungkan Arduino ke internet menggunakan jaringan nirkabel GPRS. Cukup colokkan modul ini ke papan Arduino Anda, pasang kartu SIM dari operator yang menawarkan jangkauan GPRS dan ikuti beberapa petunjuk sederhana untuk mulai mengendalikan dunia Anda melalui internet. Anda juga dapat membuat / menerima panggilan suara menggunakan jack audio / mikrofon on-board dan mengirim / menerima pesan SMS. Arduino *GSM Shield 2* memungkinkan dewan Arduino terhubung ke internet, membuat / menerima panggilan suara dan mengirim / menerima pesan SMS. Perisai menggunakan modem radio M10 oleh Quectel. Hal ini dimungkinkan untuk berkomunikasi dengan board menggunakan perintah AT. Perpustakaan *GSM* memiliki sejumlah besar metode untuk komunikasi dengan perisai.

Perisai menggunakan pin digital 2 dan 3 untuk komunikasi serial perangkat lunak dengan M10. Pin 2 terhubung ke pin TX M10 dan pin 3 ke pin RX-nya. Lihat catatan ini untuk bekerja dengan Arduino Uno, Uno ADK, atau Leonardo. Pin PWRKEY modem terhubung ke pin Arduino 7.

M10 adalah *modem Quad-band GSM / GPRS* yang bekerja pada frekuensi GSM850MHz, GSM900MHz, DCS1800MHz dan PCS1900MHz. Ini mendukung protokol TCP / UDP dan HTTP melalui koneksi GPRS. Kecepatan downlink data GPRS dan kecepatan transfer uplink maksimal adalah 85,6 kbps.

Untuk antarmuka dengan jaringan selular, board memerlukan kartu SIM yang disediakan oleh operator jaringan. Lihat halaman awal untuk informasi tambahan tentang penggunaan SIM.

2.3.11 SMS (Short Message Service)

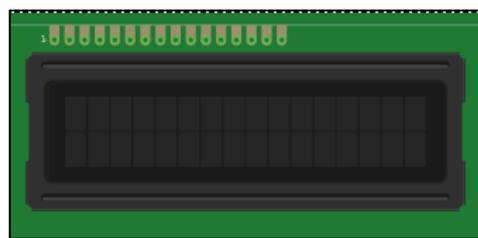
Teknologi telekomunikasi pada saat ini semakin berkembang, salah satu teknologi telekomunikasi yang sedang berkembang yaitu Short Message Service atau biasanya disebut dengan SMS. SMS adalah kemampuan untuk mengirim dan menerima pesan singkat dalam bentuk teks dari sebuah perangkat nirkabel, yaitu perangkat telekomunikasi telpon seluler, dalam hal ini perangkat nirkabel yang digunakan adalah telpon *seluler*. Teks tersebut bisa terdiri dari kata-kata atau nomor ataupun kombinasi *alphanumeric*. Pendapat lain mengenai pengertian SMS

di utarakan oleh Romzi Imron (Romzi Imron 2004:1) yang mengungkapkan tentang pengertian SMS adalah sebagai berikut:

“Layanan yang banyak di aplikasikan pada jaringan komunikasi tanpa kabel yang memungkinkan dilakukannya pengiriman pesan dalam bentuk *alphanumeric* antar terminal pelanggan (Ponsel) atau antara terminal pelanggan dengan sistem eksternal seperti *e-mail*, *paging*, *voice mail* dan sebagainya” (Imron,2004:1)

2.3.12 LCD (Liquid Crystal Display)

Display LCD (*Liquid Crystal Display*) adalah penampil kristal cair yang terdiri atas tumpukan tipis atau sel dari dua lembar kaca yang sampingnya tertutup rapat. Permukaan luar dari masing-masing keping kaca mempunyai lapisan penghantar tembus cahaya. Sel mempunyai ketebalan sekitar 1×10^{-5} meter dan diisi dengan kristal cair. Beberapa hal yang perlu diperhatikan untuk pengaksesan LCD yaitu LCD selalu berada pada kondisi tulis (Write) yaitu dengan menghubungkan kaki R/W ke ground. Hal ini dimaksudkan agar LCD tersebut tidak pernah mengeluarkan data (pada kondisi baca) yang mengakibatkan tabrakan data dengan komponen lain di jalur bus. Penampil kristal cair memerlukan catu daya dari power supply sebesar +5 volt. Bentuk LCD seperti pada gambar 2.17.



Gambar 2.17 Bentuk Fisik LCD

(Sumber <http://sainsdanteknologiku.blogspot.co.id/2017>)

2.3.13 Mikrokontroller

Mikrokontroler adalah sebers chip yang berfungsi sebagai pengontrol rangkaian elektronik dan umumnya dapat menyimpan program pada umumnya terdiri dari CPU (*Central Processing Unit*), memori, I/O tertentu dan unit pendukung seperti *Analog-to-Digital Converter* (ADC) yang sudah terintegrasi di dalamnya. Kelebihan utama dari *Mikrokontroler* ialah tersedianya RAM dan peralatan I/O

pendukung sehingga ukuran board *Mikrokontroller* menjadi sangat ringkas. (Arduino, 2016).

2.3.14 Modul Arduino AT Mega2560

Arduino Mega 2560 adalah sebuah papan mikrokontroler berbasis Atmega 2560 (*datasheet*). Mempunyai 54 pin digital *input/output* (dimana 14 pun dapat digunakan sebagai keluaran PWM), 16 pin input analog, 2 UARTs (*Hardware serial ports*), sebuah *crystal oscillator* 16 MHz, sebuah penghubung USB, sebuah colokan listrik, ICSP *header*, dan tombol kembali. Setiap isi dari Arduino Mega 2560 membutuhkan dukungan mikrokontroler; koneksi mudah antara Arduino mega 2560 ke komputer dengan sebuah kabel USB atau daya dengan AC to DC adaptor atau baterai untuk memulai. Arduino mega cocok sebagai rancangan pelindung untuk Arduino *Deumilanove* atau *Diecimila*.



Gambar 2.18 Arduino Mega 2560

Sumber <https://www.arcaelectronica.com>)

2.3.15 Arsitektur Arduino Mega 2560

Arduino Mega 2560 terbentuk dari prosessor yang dikenal dengan Mikrokontroler ATmega 2560. Mikrokontroler ATmega 2560 memiliki beberapa fitur / spesifikasi yang menjadikannya sebagai solusi pengendali yang efektif untuk berbagai keperluan. Fitur-fitur tersebut antara lain :

1. Tegangan Operasi sebesar 5 V

2. Tegangan input sebesar 6 – 20 V tetapi yang direkomendasikan untuk ATmega 2560 sebesar 7 – 12 V.
3. Pin digital I/O sebanyak 54 pin dimana 14 pin merupakan keluaran dari PWM.
4. Pin input analog sebanyak 16 pin
5. Arus DC pin I/O sebesar 40 mA sedangkan Arus DC untuk pin 3.3V sebesar 50 mA
6. Flash memory 156 Kb yang mana 8 Kb digunakan oleh bootloader.
7. SRAM 8 Kbyte
8. EEPROM 4 Kbyte
9. Serta mempunyai 2 Port UARTs untuk komunikasi serial.



Gambar 2.19 ATmega 2560 pada Arduino Mega 2560
(Atmel Corporation, 2014)

2.3.16 Konfigurasi Pin Arduino Mega

1. VCC adalah tegangan catu digital
2. GND adalah *Ground*
3. Port A (PA7..PA0)

Port A adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port A memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port A eksternal *pulled low* sumber arus jika resistor pull-up aktif. Pin port A dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan. Port A juga menyajikan fungsi dari berbagai fitur spesial dari Atmega640/1280/1281/2560/2561.

4. Port B (PB7..PB0)

Port B adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port B memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port A eksternal *pulled low* sumber arus jika resistor pull-up aktif. Pin port A dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan. Port B mempunyai kemampuan bergerak lebih baik daripada port lainnya.

5. Port C (PC7..PC0)

Port C adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port C memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port C eksternal *pulled low* sumber arus jika resistor pull-up aktif. Pin port C dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

6. Port D (PD7..PD0)

Port D adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port D memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port D eksternal *pulled low* sumber arus jika resistor *pull-up* aktif. Pin port D dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

7. Port E (PE7..PE0)

Port E adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port E memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port E eksternal *pulled low* sumber arus jika resistor *pull-up* aktif. Pin port E dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

8. Port F (PF7..PF0)

Port F disajikan sebagai masukan analog ke *A/D converter*. Port F juga menyajikan sebuah port I/O 8 bit dua arah, jika *A/D Converter* tidak

digunakan. Pin port dapat menyediakan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port F memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port F *eksternal pulled low* sumber arus jika resistor pull-up aktif. Pin port F dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan. Jika antarmuka JTAG mengizinkan, *pull-up resistor* pada pin PF7(TDI), PF5(TMS), dan PF4(TCK) akan iaktifkan bahkan jika terjadi reset. Port F juga menyajikan fungsi dari antarmuka JTAG.

9. Port G (PG7..PG0)

Port G adalah sebuah port I/O 6 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port G memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port G *eksternal pulled low* sumber arus jika resistor *pull-up* aktif. Pin port G dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

10. Port H (PH7..PH0)

Port H adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port H memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port H *eksternal pulled low* sumber arus jika *resistor pull-up* aktif. Pin port H dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

11. Port J (PJ7..PJ0)

Port J adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port J memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port J *eksternal pulled low* sumber arus jika *resistor pull-up* aktif. Pin port J dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

12. Port K (PK7..PK0)

Port K disajikan sebagai masukan analog ke *A/D converter*. Port K adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port K memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port K eksternal *pulled low* sumber arus jika *resistor pull-up* aktif. Pin port K dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

13. Port L (PL7..PL0)

Port L adalah sebuah port I/O 8 bit dua arah dengan *internal pull-up resistor* (dipilih untuk masing-masing bit). Penyangga output Port L memiliki karakter penggerak karakteristik dengan kedua sink tinggi dan kemampuan sumber. Sebagai input, pin Port L eksternal *pulled low* sumber arus jika *resistor pull-up* aktif. Pin port L dinyatakan tri ketika sebuah kondisi reset menjadi aktif, bahkan jika waktu tidak berjalan.

14. Reset

Input reset. Sebuah level rendah pada pin ini untuk lebih panjang dari pada panjang minimum pulsa akan menghasilkan sebuah reset, bahkan jika waktu tidak berjalan. Panjang minimum pulsa dijelaskan pada “Sistem dan karakter reset” pada halaman 360. Pulsa terpendek tidak dijamin menghasilkan sebuah reset .

15. XTAL1

Input ke *inverting amplifier oscillator* dan input ke internal jalur operasi waktu.

16. XTAL2

Keluaran dari *inverting oscillator amplifier*

17. AVCC

AVCC merupakan pin tegangan catu untuk port F dan *A/D Converter*. AVCC dapat terhubung secara eksternal ke VCC, bahkan jika ADC tidak digunakan

jika ADC digunakan, ADC akan terhubung ke VCC melalui sebuah *low pass filter*.

18. AREF

AREF adalah pin referensi analog untuk *A/D Converter* (Atmel Corporation, 2014).

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 Mikrokontroler 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 :

1. *Editor Program*

Sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *processing*.

2. *Compiler*

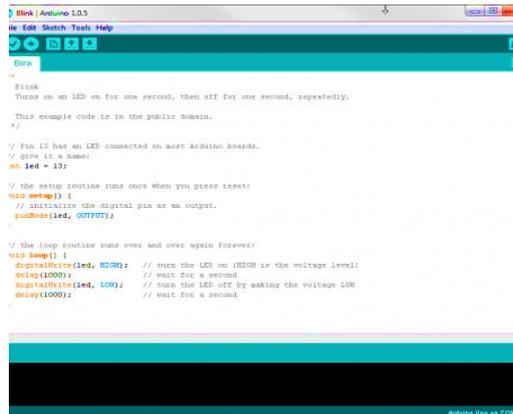
Berfungsi untuk kompilasi *sketch* tanpa unggah ke *board* bisa dipakai untuk pengecekan kesalahan kode *sintaks sketch*. Sebuah modul yang mengubah kode program menjadi kode *biner* bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *processing*.

3. *Uploader*

Berfungsi untuk mengunggah hasil kompilasi *sketch* ke *board* target. Pesan *error* akan terlihat jika *board* belum terpasang atau alamat *port* COM belum

terkonfigurasi dengan benar. Sebuah modul yang memuat kode *biner* dari komputer ke dalam *memory* didalam papan *arduino*.(Sumber: B.Gustomo, 2015).

2.4.1.1 Program Arduino Ide



```
Arduino IDE v1.0.5
File Edit Sketch Tools Help
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 2.20 Tampilan Program Arduino Uno

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*

2.4.1.2 Header

Pada bagian ini biasanya ditulis definisi-definisi penting yang akan digunakan selanjutnya dalam program, misalnya penggunaan *library* dan pendefinisian *variable*. *Code* dalam blok ini dijalankan hanya sekali pada waktu *compile*. Di bawah ini contoh *code* untuk mendeklarasikan *variable led* (integer) dan sekaligus di isi dengan angka 13.

```
int led = 13;
```

2.4.1.3 Setup

Di sinilah awal program *Arduino* berjalan, yaitu di saat awal, atau ketika *power on Arduino board*. Biasanya di blok ini diisi penentuan apakah suatu pin digunakan sebagai *input* atau *output*, menggunakan perintah *pinMode*. Inisialisasi *variable* juga bisa dilakukan di blok ini // the setup routine runs once when you press reset: void setup() { // initialize the digital pin as an output. pinMode(led, OUTPUT); }

OUTPUT adalah suatu makro yang sudah didefinisikan *Arduino* yang berarti = 1. Jadi perintah di atas sama dengan pinMode(led, 1);

Suatu pin bisa difungsikan sebagai *OUTPUT* atau *INPUT*. Jika difungsikan sebagai output, dia siap mengirimkan arus listrik (maksimum 100 mA) kepada beban yang disambungkannya. Jika difungsikan sebagai INPUT, pin tersebut memiliki *impedance* yang tinggi dan siap menerima arus yang dikirimkan kepadanya.

2.4.1.4 Loop

Blok ini akan dieksekusi secara terus menerus. Apabila program sudah sampai akhir blok, maka akan dilanjutkan dengan mengulang eksekusi dari awal blok. Program akan berhenti apabila tombol *power Arduino* di matikan. Di sinilah fungsi utama program *Arduino* kita berada.

```
void loop() {
```

```
digitalWrite(led, HIGH); // nyalakan LED delay(1000); // tunggu 1000 milidetik  
digitalWrite(led, LOW); // matikan LED delay(1000); // tunggu 1000 milidetik }
```

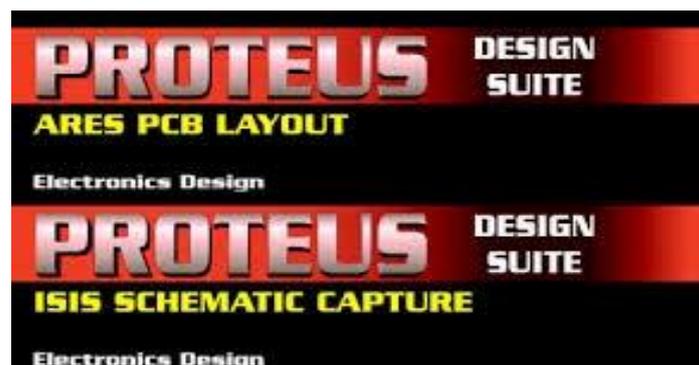
Perintah *digitalWrite*(pinNumber,nilai) akan memerintahkan *arduino* untuk menyalakan atau mematikan tegangan di *pinNumber* tergantung nilainya. Jadi perintah di atas *digitalWrite*(led,HIGH) akan membuat pin nomor 13 (karena di header dideklarasikan led = 13) memiliki tegangan = 5V (HIGH). Hanya ada dua kemungkinan nilai *digitalWrite* yaitu *HIGH* atau *LOW* yang sebetulnya adalah nilai integer 1 atau 0. Kalau sudah dibuat program di atas, selanjutnya kita ambil kabel USB yang diikutsertakan pada saat membeli *Arduino*, pasang ke komputer dan *board arduino*, dan *upload* programnya. Lampu LED yg ada di

Arduino board kita akan kelap-kelip. Sekedar informasi, sebuah LED telah disediakan di *board Arduino Uno* dan disambungkan ke pin 13.

Selain blok *setup()* dan *loop()* di atas kita bisa mendefinisikan sendiri blok fungsi sesuai kebutuhan. Kita akan jumpai nanti pada saat pembahasan proyek. (Sumber: Septa Ajjie, 2016).

2.4.2 Software ISIS & ARES Proteus 7.0

Proteus adalah sebuah software untuk mendesain PCB yang juga dilengkapi dengan simulasi PSpice pada level skematik sebelum rangkaian skematik di-upgrade ke PCB sehingga sebelum PCBnya di cetak kita akan tahu apakah PCB yang akan kita cetak apakah sudah benar atau tidak. Proteus mampu mengkombinasikan program ISIS untuk membuat skematik desain rangkaian dengan program ARES untuk membuat layout PCB dari skematik yang kita buat. Software Proteus ini bagus digunakan untuk desain rangkaian mikrokontroller.



Gambar 2.21. Tampilan Software ISIS & ARES Proteus

(Sumber <https://www.Anakkendali.com,2018>)

Proteus juga bagus untuk belajar elektronika seperti dasar-dasar elektronika sampai pada aplikasi pada mikrokontroller. Software Proteus ini menyediakan banyak contoh aplikasi desain yang disertakan pada instalasinya. Sehingga

memungkinkan bisa belajar dari contoh-contoh yang sudah ada. Fitur- fitur yang terdapat dalam Proteus adalah sebagai berikut :

1. Memiliki kemampuan untuk mensimulasikan hasil rancangan baik digital maupun analog maupun gabungan keduanya.
2. Mendukung simulasi yang menarik dan simulasi secara grafis.
3. Mendukung simulasi berbagai jenis microcontroller seperti PIC 8051 series
4. Memiliki model-model peripheral yang interactive seperti LED, tampilan LCD, RS232, dan berbagai jenis library lainnya.
5. Mendukung instrument-instrument virtual seperti voltmeter, ammeter, oscilloscope, logic analyser, dan lain-lainnya.
6. Memiliki kemampuan menampilkan berbagai jenis analisis secara grafis seperti transient, frekuensi, noise, distorsi, AC dan DC, dan lain-lainnya.
7. Mendukung berbagai jenis komponen-komponen analog.
8. Mendukung open architecture sehingga kita bisa memasukkan program seperti C++
9. untuk keperluan simulasi.
10. Mendukung pembuatan PCB yang di-update secara langsung dari program ISIS ke program pembuat PCB-ARES.

ISIS dipergunakan untuk keperluan pendidikan dan perancangan. Beberapa fitur umum dari ISIS adalah sebagai berikut :

1. Windows dapat dioperasikan pada Windows 98/Me/2k/XP dan Windows terbaru.
2. Routing secara otomatis dan memiliki fasilitas penempatan dan penghapusan dot.
3. Sangat powerful untuk pemilihan komponen dan pemberian properties-nya.
4. Mendukung untuk perancangan berbagai jenis bus dan komponen-komponen pin, pormodul dan jalur.
5. Memiliki fasilitas report terhadap kesalahan-kesalahan perancangan dan simulasi elektrik.
6. Mendukung fasilitas interkoneksi dengan program pembuat PCB-ARES.

7. Memiliki fasilitas untuk menambahkan package dari komponen yang belum didukung.

ARES (Advanced Routing and Editing Software) digunakan untuk membuat modul layout PCB. Adapun fitur-fitur dari ARES adalah sebagai berikut :

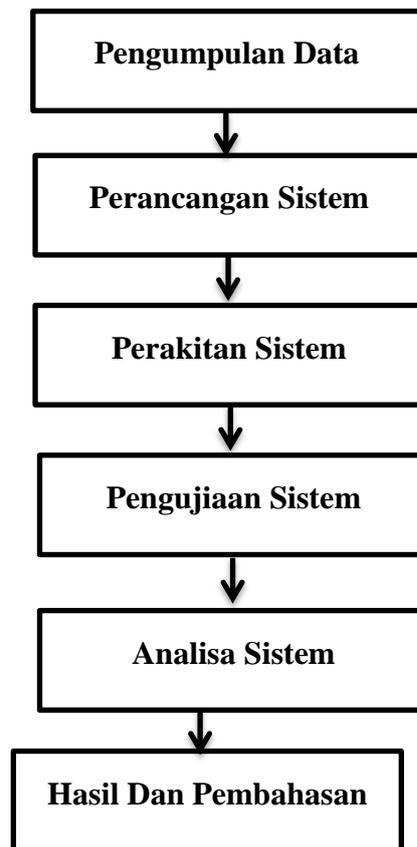
1. Memiliki database dengan tingkat keakuratan 32-bit dan memberikan resolusi sampai 10 nm, resolusi angular 0,1 derajat dan ukuran maksimum board sampai 10 m.
2. ARES mendukung sampai 16 layer.
3. Terintegrasi dengan program pembuat skematik ISIS, dengan kemampuan untuk menentukan informasi routing pada skematik.
4. Visualisasi board 3-Dimensi.
5. Penggambaran 2-Dimensi dengan simbol library.

Proteus lebih memiliki kelebihan pada desainnya yang sederhana, sangat mudah dan bagus digunakan untuk perancangan rangkaian mikrokontroller yang akan sangat membantu digunakan oleh mahasiswa yang mengambil mata kuliah berhubungan dengan mikrokontroller. Kelebihannya yang lain adalah sebelum PCB dicetak skematiknya bisa disimulasikan dulu.

BAB III

METODOLOGI PENELITIAN

Pada bab ini akan menjelaskan metode penelitian yang akan digunakan dan mendapatkan dari referensi, jurnal-jurnal dan skripsi terdahulu, dan akan menjelaskan bagaimana merancang Implementasi Pengontrol Suhu Dan Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Dengan Output SMS Berbasis Mikrokontroler, menjelaskan alat dan bahan yang akan digunakan, diagram blok dari alat dan cara kerja alat tersebut, diagram alur penelitian ini dapat dilihat pada gambar 3.1.



Gambar 3.1 Diagram Alur Penelitian

Penjelasan diagram alur penelitian:

1. Pengumpulan Data

Mencari referensi dari berbagai sumber seperti halaman situs, jurnal, buku, dan lain sebagainya yang terkait dengan penelitian yang akan dilakukan, guna untuk menambah pengetahuan penelitian dan informasi yang dapat digunakan, serta untuk membantu proses penelitian “Pengontrol Suhu Dan Pemberi Pakan Ayam Otonom Ternakan Ayam Dengan Output SMS berbasis mikrokontroler”. Pengumpulan Data ini juga di peroleh dari survei lokasi dan wawancara yang dilakukan pada pemilik kandang ternak ayam broiler.

2. Perancangan Sistem

Perancangan sistem berisi dari menentukan alat, bahan yang akan digunakan sampai merancang dan simulasi dari setiap komponen yang digunakan, dalam penelitian ini. Perancangan terbagi menjadi 2 macam, yaitu perancangan perangkat keras dan perancangan perangkat lunak.

3. Perakitan Sistem

Perakitan pada tahap ini, semua perancangan yang telah di buat kemudian dirakit dan di-*compile* ke dalam sistem yang utuh dengan sumber tegangan, sensor, aktuator, dan mikrokontroler-nya.

4. Pengujian sistem

Setelah perakitan selesai, penulis melakukan pengujian sistem untuk menguji apakah sistem telah berjalan dengan baik dan sesuai tujuan awal sistem, serta untuk menetapkan hasil dan menemukan kesalahan-kesalahan yang mengganggu sistem untuk mencapai tujuan penelitian.

5. Penerapan

Tahapan ini berisi tentang implementasi atau penerapan dari sistem yang telah dirancang, Sistem ini akan dibawa ke lokasi penelitian untuk dilakukan pembahasan dan pengambilan data berdasarkan hasil penerapan yang telah dilakukan

1.1 Pengumpulan Data

Pengumpulan data skripsi ini didapat dari jurnal, buku, tesis, *proceeding*, dan artikel yang telah diunduh dari internet sebagai tolak ukur dan penambah materi

mengenai sistem yang akan di buat. Selain mengunduh dari internet untuk *referensi* penulis juga melakukan survei lokasi dan wawancara agar mendapatkan *informasi* yang lebih *spesifik*.

1.2 Perancangan Sistem

Setelah mendapatkan data yang diperlukan untuk melakukan penelitian ini, alat dan bahan yang dibutuhkan untuk mencapai tujuan sistem akan diidentifikasi dan didapatkan.

1.2.1 Alat

Sebelum membuat Pengontrol Suhu Dan Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Dengan Output SMS berbasis mikrokontroler ada beberapa peralatan yang harus disiapkan. Daftar peralatan yang digunakan dalam penelitian ini akan dituliskan pada Tabel 3.1.

Tabel 3.1. Alat Yang Dibutuhkan

No	Nama Alat	Spesifikasi	Fungsi	Jumlah
1	Komputer/ laptop	Window 7- 10 32/64bit	Untuk membuat seberas aplikasi yang akan di pakai di perangkat keras dan pernangkat lunak	1 unit
2	IDE Arduino	Arduino 1.6.3	Membuat program yang akan di-download perangkat arduino	
3	Proteus	7.1 Profesional	Merancang rangkaian yang akan digunakan untuk membuat alat	1 beras
4	Kit Arduino	Atmega2560	Sebagai proses perintah yang akan di jalankan	1
5	GSM Shiell	-	Sebagai otputan untuk memberi informasi berupa SMS	1
6	<i>Sensor DHT11</i>	-	Digunakan untuk mmengukur suhu pada ruangan ternak ayam	1
7	<i>RTC Ds1307</i>		Digunakan sebagai penjadwalan pakan ayam	1

8	Motor DC		Digunakan sebagai penggerak pakan dari tempat ke 1 ke yang lain	1
9	Motor Servo		Digunakan sebagai pembuka dan penutup pakan ayam	1
10	Lcd		Digunakan sebagai tampilan dari sensor	1
11	Jumper		Digunakan sebagai penghubung/menjumper seluruh komponen	30
12	Keypad		Digunakan sebagai pengatur jadwal pakan	1

1.2.2 Bahan

Sebelum membuat Pengontrol Suhu Dan Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Dengan Output SMS berbasis mikrokontroler ada beberapa peralatan yang harus disiapkan. Daftar komponen yang digunakan dalam penelitian ini akan dituliskan pada Tabel 3.2.

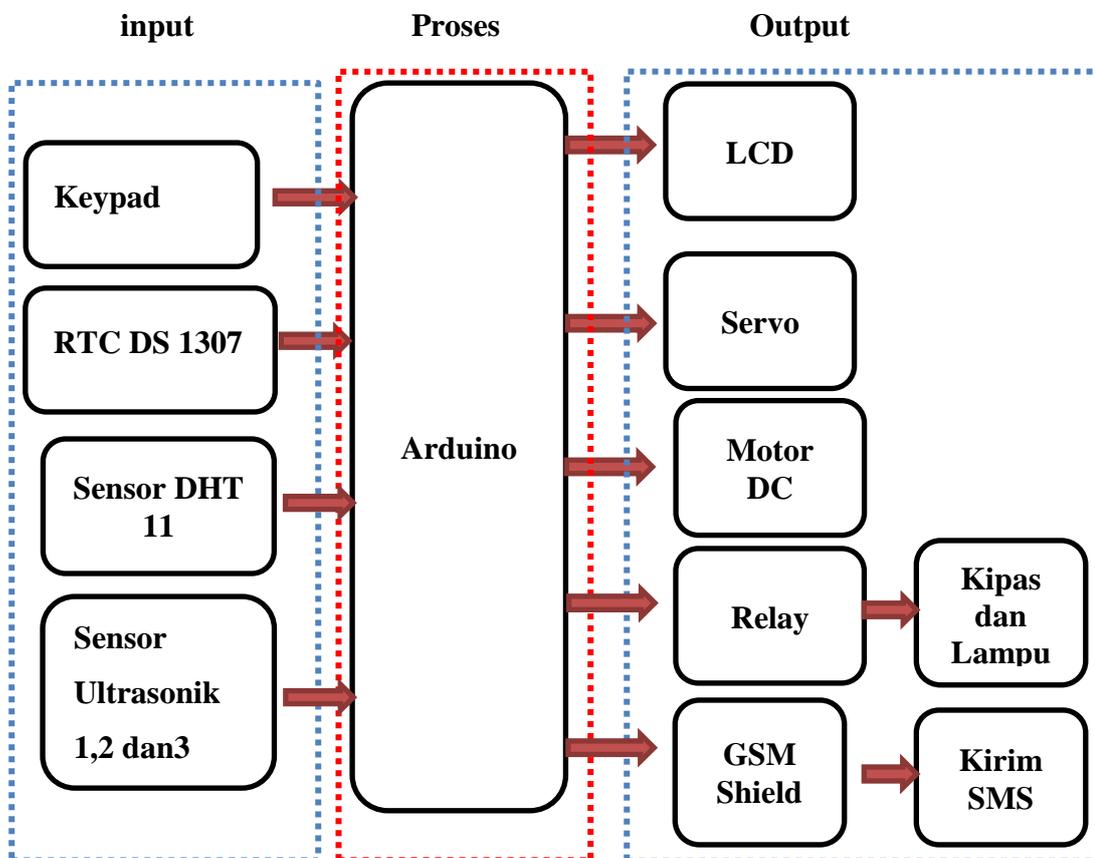
Tabel 3.2. Komponen Yang Dibutuhkan

No	Nama Alat	Sepesifikasi	Fungsi	Jumlah
1	Multitester	Analog/Digital	digunakan untuk mengukur tegangan (ACV-DCV), dan kuat arus (mA- μ A)	1 buah
2	Obeng	Obeng + dan -	Untuk merangkai alat	2 buah
3	Solder	-	Untuk menempelkan timah ke komponen	1 buah
4	Bor pcb	-	Untuk membuat lobang baut atau komponen	1 buah
5	Tang Potong	-	Untuk memotong kabel dan kaki komponen	1 buah
6	Lem Cina	-	Digunakan Sebagai plekat Akralik	1 buah
7	Akalik	-	Digunakan Sebagai Penutup Kandang	1 meter

8	Triplek	-	Digunakan Sebagai Penutup Kandang	3 meeter
9	Paralon dan T	-	Digunakan Sebagai tempat airan pemberi pakan	2 meter
10	Corong	-	Digunakan sebagai wadah pakan	1 buah
11	Solasi Hitam	-	Membungkus sambungan kabel	1 buah
12	Baut	-	Digunakan sebagai pengencang alat	20 buah
13	Kabel Power	-	Digunakan sebagai aliran listrik pada tafo	1buah
14	Papan PCB	-	Digunakan untuk merangkai alat	5 buah

1.2.3 Perancangan Perangkat Keras

Perancangan sistem ini terbagi menjadi 2, yaitu perancangan perangkat keras dan perangkat lunak. Blok diagram dari sistem yang akan dibangun dapat dilihat pada gambar 3.2.

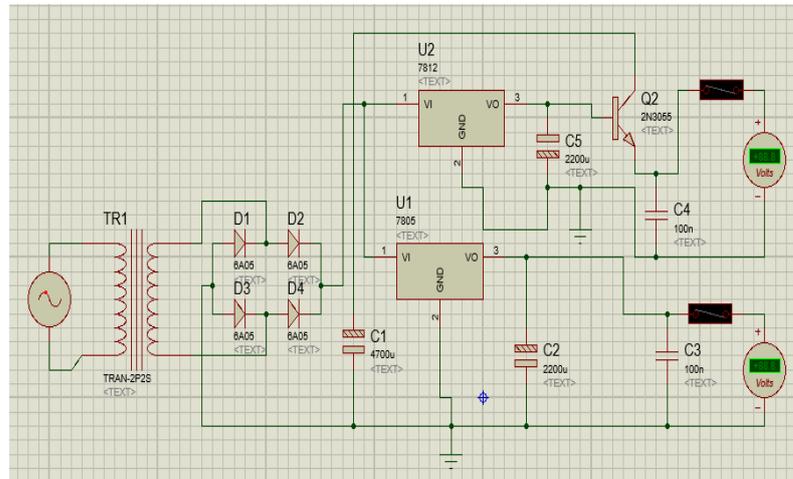


Gambar 3.2. Blok Diagram Sistem

Dari gambar blok diagram sistem dapat diketahui sistem kerja dari alat yaitu jika. Sensor suhu melakukan pengukuran suhu mencapai $>40^{\circ}$ maka *relay* akan aktif menyalakan kipas, jika suhu $<40^{\circ}$ maka relay akan mengaktifkan lampu agar panas kembali normal hasil pengukuran dari suhu akan ditampilkan pada LCD 20x4. Sedangkan dalam melakukan pemberian pakan maka peneliti melakukan penjadwalan pada RTC yang akan diatur melalui keypad sehingga jika jam sudah menunjukkan waktu yang telah diatur tadi maka arduino akan memproses motor servo terbuka dan motor DC akan aktif sehingga pakan akan turun mengisi setiap-setiap wadah pakan ayam melalui selah-selah paralon. Sedangkan sensor ultrasonik digunakan sebagai pengukur jarak pakan saat sedang melakukan pengisian sehingga jika pakan sudah mencapai dengan batas jarak yang telah dibuat pada sensor ultrasonik, maka motor servo akan tertutup dan motor DC akan berhenti berputar. Jika sensor ultrasonik yang ada pada tandon penampung pakan telah habis maka akan memerintahkan *GSM Shield* agar mengirimkan SMS kepada pemilik sehingga pemilik akan melakukan pengisian tandon penampung pakan tersebut.

1.2.3.1 Rangkaian Power Supply

Rangkaian power supply digunakan untuk merubah tegangan AC 220V menjadi DC 12V dalam pembuat *power supply* 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 3.4.



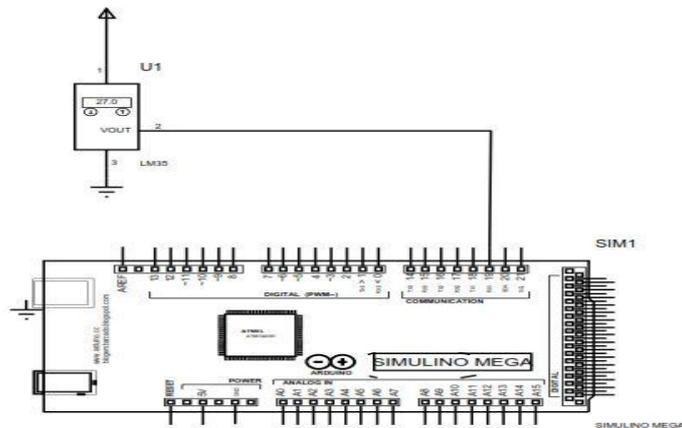
Gambar 3.4 Rangkaian Power Supply

Penjelasan:

TR1 adalah *Transformer Centre Tap* dengan 2 kaki input 220V AC dan 2 kaki output 12V yang berfungsi untuk menurunkan tegangan listrik sesuai dengan kebutuhan komponen elektronika, **D1-D4** adalah Dioda 6A05 yang dirangkai secara bridge berfungsi sebagai *Rectifier* (penyearah gelombang), sedangkan **U1** dan **U2** adalah IC yang berfungsi sebagai Pengatur Tegangan dimana **U1** adalah IC *Regulator* 7805 dengan 1 kaki tegangan masukan, 1 kaki *ground*, dan 1 kaki tegangan keluaran. **U2** adalah IC *Regulator* 7812 dengan 1 kaki tegangan masukan, 1 kaki *ground*, dan 1 kaki tegangan keluaran. **C1** adalah kapasitor dengan besar kapasitansi $4700\mu F$, **C2** dan **C5** adalah kapasitor penyangkapan dengan besar kapasitansi $2200\mu F$, **C3** dan **C4** adalah kapasitor penyangkapan dengan besar kapasitansi $100nF$, **Q2** adalah transistor 2N3055 sebagai *Over Voltage Protection* penguat tegangan.

1.2.3.2 Rangkaian Sensor DHT 11

Rangkaian *sensor DHT11* digunakan sebagai *input* dalam mengukur suhu pada kandang ayam yang akan diproses oleh arduino mega, sehingga akan menghasilkan output menyalakan lampu atau kipas sebagai pemanas dan penurunan suhu pada kandang ayam. Gambar rangkaian *Sensor DHT11* dan tata letak dapat dilihat seperti pada gambar 3.4.



Gambar 3.4 Rangkaian *Sensor DHT11*

Pada rangkaian *sensor DHT11* hanya beberapa kaki yang dihubungkan ke pin digital *arduino* agar hasil proses pada *arduino* dapat menyalakan lampu sebagai pemanas dari ruangan dan menyalakan kipas sebagai penurun suhu dalam kandang ayam.

Penjelasan pin sensor DHT 11 sebagai berikut:

Sensor DHT 11 mendapat tegangan input sebesar +5.0V dari sumber tegangan. Kaki GND pada sensor DHT 11 mendapat *Ground* dari sumber tegangan dan Kaki Data sensor DHT 11 mendapat pin D19 dari mikrokontroler.

Potongan program (*script*) *Sensor DHT 11*:

```

void suhu(){ //PERINTAH UNTUK SENSOR DHT11
  float kelembapan=dht.readHumidity();
  float suhu=dht.readTemperature();

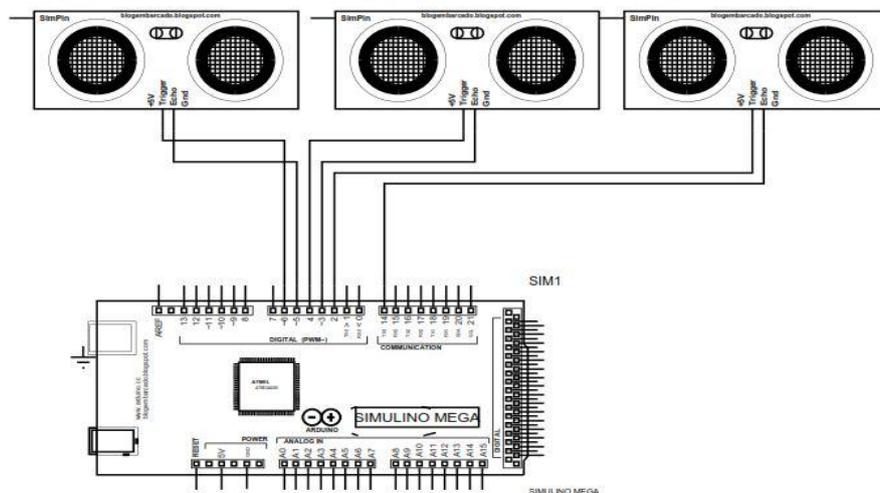
  Serial.print("Kelembapan:"); Serial.print(kelembapan); Serial.print(" --><<-- "); Serial.print("Suhu:"); Serial.println(suhu);
  lcd.setCursor(4,3); lcd.print("Suhu: "); lcd.print(suhu);
  if(suhu <= 34){
    digitalWrite(lampu,LOW); //LAMPU NYALA DAN KIPAS MATI
    digitalWrite(kipas,HIGH);
  }
  else{
    digitalWrite(lampu,HIGH); //LAMPU MATI DAN KIPAS NYALA
    digitalWrite(kipas,LOW);
  }
}

```

Gambar 3.5 Script Program *Sensor DHT 11*

1.2.3.3 Rangkaian Sensor Ultrasonik

Rangkaian *sensor ultrasonik* digunakan sebagai *input* yang akan diproses oleh arduino, sehingga menghasilkan output berupa motor servo akan bergerak buka tutup pakan ternak ayam. Gambar rangkaian *Sensor ultrasonik* dan tata letak dapat dilihat seperti pada gambar 3.6.



Gambar 3.6. Rangkaian Sensor Ultrasonik

Pada rangkaian *sensor ultrasonik* hanya beberapa kaki yang dihubungkan ke pin digital *arduino* agar hasil proses pada arduino dapat menggerakkan motor servo untuk melakukan buka tutup pakan ayam. Penjelasan penggunaan PIN arduino ATmega2560 dan *Sensor Ultrasonik* sebagai berikut:

Sensor Ultrasonik 1 digunakan sebagai pengukur tinggi tandon

Sensor Ultrasonik 1 mendapat tegangan input sebesar +5.0V dari sumber tegangan, Kaki GND sensor mendapat *Ground* dari sumber tegangan, Kaki Data PIN *Trigger* pada sensor 1 mendapatkan pin D6 dari mikrokontroler untuk kluarnya sinyal, Kaki Data *Echo* pada sensor 1 mendapat pin D5 dari mikrokontroler.

Sensor ultrasonik 2 dan 3 digunakan sebagai pengukur tinggi tempat pakan penjelasan pin sensor ultrasonik 2 dan 3 sebagai berikut:

Sensor Ultrasonik 2 dan 3 mendapat tegangan input sebesar +5.0V dari sumber tegangan, Kaki GND pada sensor mendapat *Ground* dari sumber tegangan, Kaki Data *Trigger* pada sensor 2 dan 3 mendapat pin D4 dan D2 pada mikrokontroler untuk keluarnya sinyal dari sensor, dan Kaki Data *Echo* pada sensor ultrasonik 2 dan 3 berfungsi untuk menangkap sinyal pantul dari jarak pakan, kaki data *Echo* pada sensor ultrasonik 2 dan 3 mendapat pin D3 dan 14 dari mikrokontroler.

Potongan program (*script*) *Sensor Ultrasonik*:

```
void setup() {
  pinMode(trigPin1, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin1, INPUT); // Sets the echoPin as an Input
  pinMode(trigPin2, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin2, INPUT); // Sets the echoPin as an Input
  pinMode(trigPin3, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin3, INPUT); // Sets the echoPin as an Input
  Serial.begin(9600); // Starts the serial communication
}

void loop() {
  distance1 = getDistance(trigPin1, echoPin1);
  printDistance(1, distance1);
  delay(150);

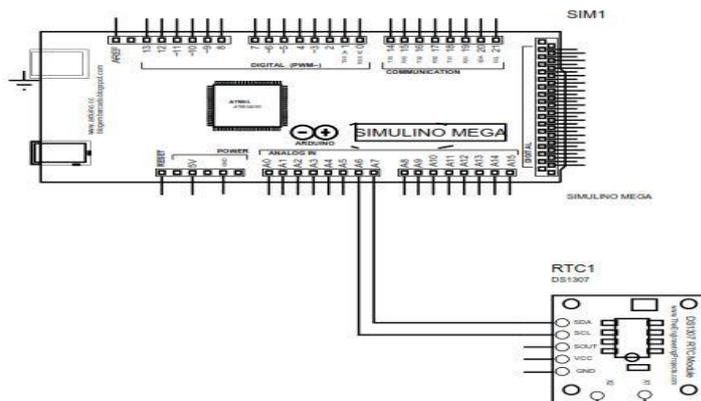
  distance2 = getDistance(trigPin2, echoPin2);
  printDistance(2, distance2);
  delay(150);

  distance3 = getDistance(trigPin3, echoPin3);
  printDistance(3, distance3);
  delay(150);
  Serial.println("");
  delay(500);
}
```

Gambar 3.7 Script Program *Sensor Ultrasonik*

1.2.3.4 Rangkaian RTC DS1307

Rangkaian RTC DS1307 yang digunakan sebagai input 3 untuk menyimpan jadwal pemberi pakan yang telah diolah oleh Arduino. gambar rangkaian RTC DS1307 dan tata letak dapat dilihat seperti pada gambar 3.8.



Gambar 3.6 Rangkaian RTC DS1307

Pada rangkaian RTC DS1307 hanya beberapa kaki yang dihubungkan ke pin digital arduino agar hasil proses pada arduino dapat menghasilkan tampilan jam yang akurat sehingga dapat digunakan sebagai penjadwalan pemberian pakan. Penjelasan penggunaan PIN arduino dan *RTC DS1307* sebagai berikut:

kaki pin Vcc Pada RTC DS1307 mendapat tegangan input sebesar +5.0V dari sumber tegangan, Kaki GND pada RTC DS 1307 mendapat Ground dari sumber tegangan, Kaki Data SDA pada RTC DS1307 mendapat pin A4 dari mikrokontroler dan Kaki Data SCL pada RTC DS1307 mendapat pin A5 dari mikrokontroler.

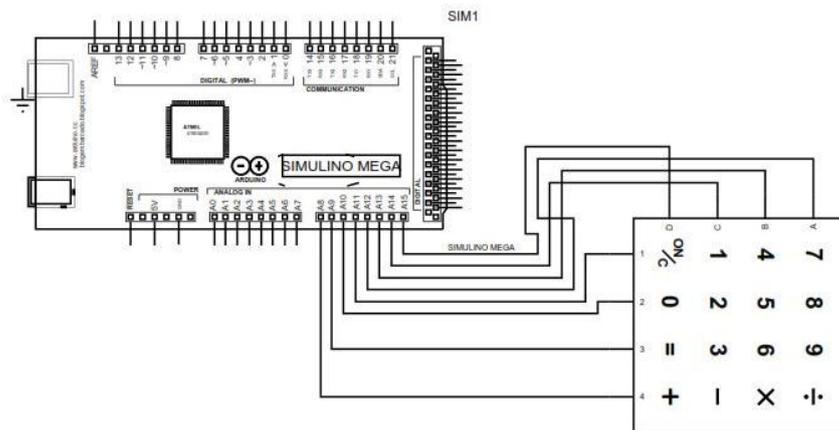
program (*script*) *RTC DS 1307*:

```
void loop() {
  rtcDS3231();
  if((dataJam == 8 && dataMenit >= 0) && (dataJam == 8 && dataMenit <= 1) && makan1 == true){
    Serial.println("Pemberian Pakan Pertama");
    pakanPertama();
  }
  else if((dataJam == 8 && dataMenit >= 3) && (dataJam == 8 && dataMenit <= 4) && makan2 == true){
    Serial.println("Pemberian Pakan Kedua");
    pakanKedua();
  }
  else if((dataJam == 8 && dataMenit >= 6) && (dataJam == 8 && dataMenit <= 7) && makan1 == true){
    Serial.println("Pemberian Pakan Pertama LAGII!!!");
    pakanPertama();
  }
}
```

Gambar 3.9 Script Program *RTC DS1307*

1.2.3.5 Rangkaian Keypad

Rangkaian *keypad* digunakan sebagai *input* yang akan diproses oleh Arduino untuk melakukan setting penjadwalan pada pakan ternak ayam secara otomatis. Gambar rangkaian *keypad* dan tata letak dapat dilihat seperti pada gambar 3.10.



Gambar 3.10 Rangkaian Keypad

Pada rangkaian *keypad* hanya beberapa kaki yang dihubungkan ke pin analog *arduino* agar hasil proses pada *arduino* agar pengguna dapat lebih mudah dalam melakukan setting waktu memberi pakan secara otomatis. Penjelasan penggunaan PIN *arduino* dan *keypad* sebagai berikut:

Kaki Data Output keypad A mendapat pin A12 dari mikrokontroler, Kaki Data Output keypad B mendapat pin A13 dari mikrokontroler, Kaki Data Output keypad C mendapat pin A14 dari mikrokontroler, Kaki Data Output keypad D mendapat pin A15 dari mikrokontroler, Kaki Data Output keypad 1 mendapat pin A11 dari mikrokontroler, Kaki Data Output keypad 2 mendapat pin A10 dari mikrokontroler, Kaki Data Output keypad 3 mendapat pin A9 dari mikrokontroler, Kaki Data Output keypad 4 mendapat pin A8 dari mikrokontroler.

Potongan program (*script*) Keypad:

```

void loop() {
  //baca();
  key = myKeypad.getKey();
  if(key) Serial.println(key);
}

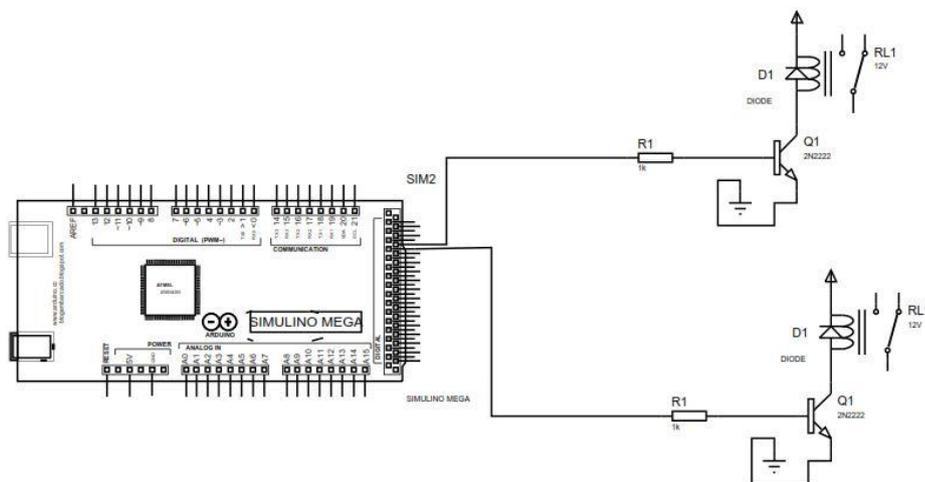
void baca() {
  key = myKeypad.getKey();
  if(key=='*') {
    cocok=false;
    loopMenu=0;
    tulisMenu();
  }
  if(key=='A') {
    cocok=true;
  }
}

```

Gambar 3.11 Script Program Keypad

1.2.3.6 Rangkaian Relay

Rangkaian *relay* digunakan sebagai *output* yang akan diproses oleh arduino sehingga akan menyalakan lampu dan kipas yang digunakan sebagai pendingin dan pemanas pada kandang. Gambar rangkaian *relay output* dan tata letak dapat dilihat seperti pada gambar 3.12.



Gambar 3.12 Ragkaian Relay

Pada rangkaian *relay* hanya beberapa kaki yang dihubungkan ke pin analog arduino agar hasil proses pada arduino dapat membuka dan mengunci pintu pada ruangan perkuliahan menggunakan *solenoid doorlock*. Penjelan penggunaan PIN arduino dan *relay* ditampilkan sebagai berikut:

Pin D21 dan D22 mikrokontroler mendapat resistor dengan tahanan sebesar 100Ω, Resistor mendapat kaki basis dari transistor agar dapat mengetahui ada atau tidaknya arus pemicuan pada kaki basisnya, saat kaki basis menerima arus pemicuan, maka transistor akan berubah ke keadaan *saturasi* dan mengantarkan arus. Pada kaki kolektor transistor dengan kaki coil *relay* dan kaki anoda dari dioda 1N4001 agar dapat menyimpan arus, sehingga Kaki katoda dari dioda 1N4001 mendapat tegangan masukan sebesar +12V jadi jika relay tidak di aliri listrik lagi, dia masih memiliki arus. kaki coil relay, Kaki NO Relay terhubung ke NO kontaktor dan Kaki COM Relay terhubung ke coil kontaktor.

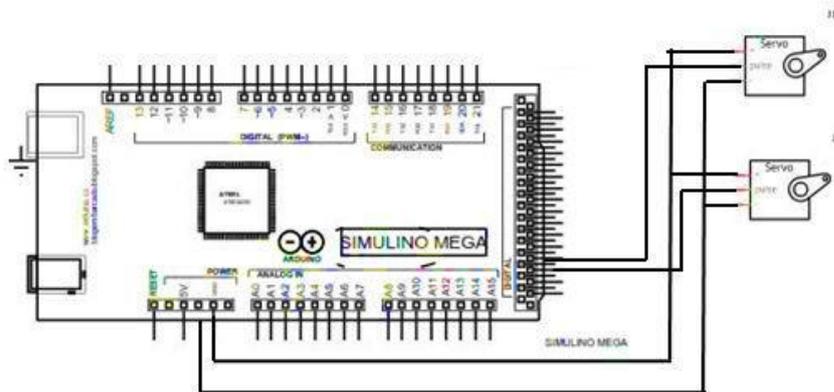
Potongan program (*script*) *Relay*:

```
digitalWrite(lampu, LOW); //LAMPU NYALA DAN KIPAS MATI
digitalWrite(kipas, HIGH);
}
else{
digitalWrite(lampu, HIGH); //LAMPU MATI DAN KIPAS NYALA
digitalWrite(kipas, LOW);
}
```

Gambar 3.13 Script Program Keypad

1.2.3.7 Rangkaian Motor Servo

Rangkaian motor servo 1 dan 2 digunakan sebagai *output* dalam membuka penutup pakan saat kondisi penuh dan membuka saat kondisi habis. Gambar rangkaian *motor servo output 2* dan tata letak dapat dilihat seperti pada gambar 3.14.



Gambar 3.14 Rangkaian Motor Servo

Pada rangkaian *motor servo* hanya beberapa kaki yang dihubungkan ke pin digital *arduino* agar hasil proses pada *arduino* dapat membuka dan menutup pakan.

Penjelasan penggunaan PIN *arduino* dan *servo* output sebagai berikut:

Motor Servo 1 mendapat tegangan input sebesar +5.0V dari sumber tegangan, Kaki GND mendapat Ground dari sumber tegangan, Kaki Data Output Servo 1 mendapat pin D51 dari mikrokontroler dan Kaki Data output servo 2 mendapat pin A52 dari mikrokontroler.

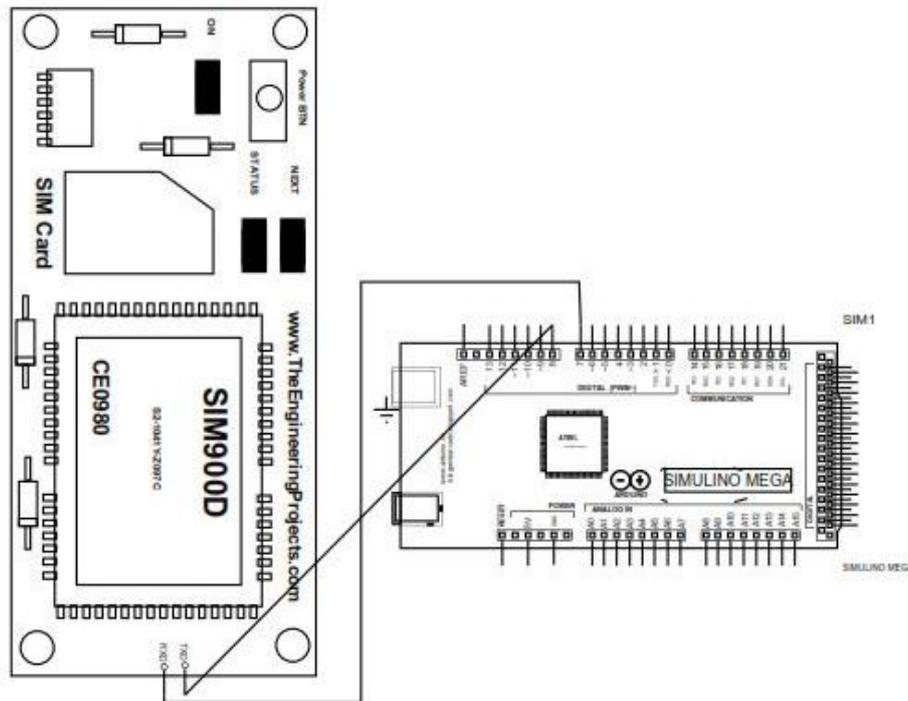
Potongan program (*script*) *Motor Servo*:

```
servo1.write(90); //Servo menuju arah 90 derajat
if(distance1 <= 10){
  lcd.clear();
  lcd.setCursor(3,2); lcd.print("Pakan 1 Telah");
  lcd.setCursor(5,3); lcd.print("Terpenuhi");
  Serial.println("Pakan 1 Telah Terpenuhi");
  servo1.write(0); //Servo menuju arah 0 derajat
  delay(2000);
  .....
```

Gambar 3.15 Script Program Motor Servo

1.2.3.8 Rangkaian *GSM Shield*

Rangkaian *GSM Shield* digunakan sebagai *output* berupa informasi yang akan diolah oleh arduino sehingga akan menghasilkan output berupa pengiriman SMS. Gambar rangkaian *gsm shield* dan tata letak dapat dilihat seperti pada gambar 3.16.



Gambar 3.16 Rangkaian *GSM Shield*

Pada rangkaian *gsm shield* hanya beberapa kaki yang dihubungkan ke pin digital arduino agar hasil proses pada arduino dapat mengirimkan SMS. Penggunaan PIN arduino dan *gsm shield* sebagai berikut:

GSM Shield mendapat tegangan *input* sebesar +5.0V dari sumber tegangan, Kaki GND mendapat *Ground* dari sumber tegangan, Kaki Data TX (*TRANSMITER*) mendapat pin D7 dari mikrokontroler, yang berfungsi sebagai mengirim data, Kaki Data RX (*RECEIVER*) mendapat pin D8 dari mikrokontroler, Sebagai jalur penerimaan data.

Potongan program (*script*) *GSM Shield*

```

void kirimPesan(){
  Serial.println("SIM900 Mengirim Pesan");
  // AT command to set SIM900 to SMS mode
  Serial1.print("AT+CMGF=1\r");
  delay(100);

  // REPLACE THE X's WITH THE RECIPIENT'S MOBILE NUMBER
  // USE INTERNATIONAL FORMAT CODE FOR MOBILE NUMBERS
  Serial.println("SIM900 Set SMS Number");
  Serial1.println("AT + CMGS = \"+6289653430648\"");
  delay(100);

  // REPLACE WITH YOUR OWN SMS MESSAGE CONTENT
  Serial.println("SIM900 SMS Konteks");
  Serial1.println("Message example from Arduino Uno.");
  delay(100);

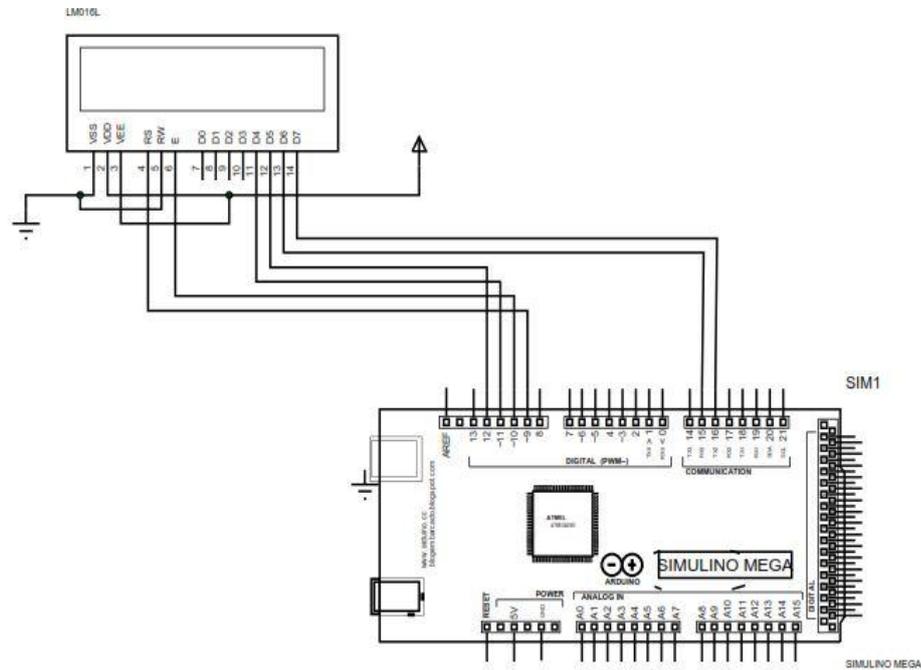
  // End AT command with a ^Z, ASCII code 26
  Serial.println("SIM900 Finish");
}

```

Gambar 3.17 Script Program GSM Shield

1.2.3.9 Rangkaian LCD (*Liquid Crystal Display*)

Rangkaian LCD Digunakan sebagai *output* untuk menampilkan informasi berbentuk data yang berasal dari sumber masukan dari *sensor DHT11 dan RTC* yang telah diolah oleh Arduino . Gambar rangkaian LCD dan tata letak dapat dilihat seperti pada gambar 3.18.



Gambar 3.18. Rangkaian *Liquid Crystal Display 20 X 4*

Pada rangkaian LCD hanya beberapa kaki yang dihubungkan ke pin digital arduino uno agar hasil proses pada arduino dapat ditampilkan kedalam LCD. Penjelasan penggunaan PIN arduino dan LCD 20x4 sebagai berikut:

Kaki VSS LCD mendapat *Ground* dari sumber tegangan, Kaki VCC LCD mendapat tegangan *input +5.0V* dari sumber tegangan sebagai *suplay*, Kaki VEE LCD terhubung dengan Pin *Potensio* sebagai pengatur kontras *display*, Kaki Data D4 LCD terhubung dengan Pin 10 mikrokontroler, Kaki Data D5 LCD terhubung dengan Pin 11 mikrokontroler, Kaki Data D6 LCD terhubung dengan Pin 15 mikrokontroler, Kaki Data D7 LCD terhubung dengan Pin 16 mikrokontroler, Kaki Data D14 LCD terhubung dengan Pin 16 mikrokontroler, Kaki Data D15 LCD terhubung dengan Pin 12 mikrokontroler, Kaki A0-A2 mendapat *Ground* dari sumber tegangan.

Potongan program (*script*) *Liquid Crystal Display 20X4* :

```

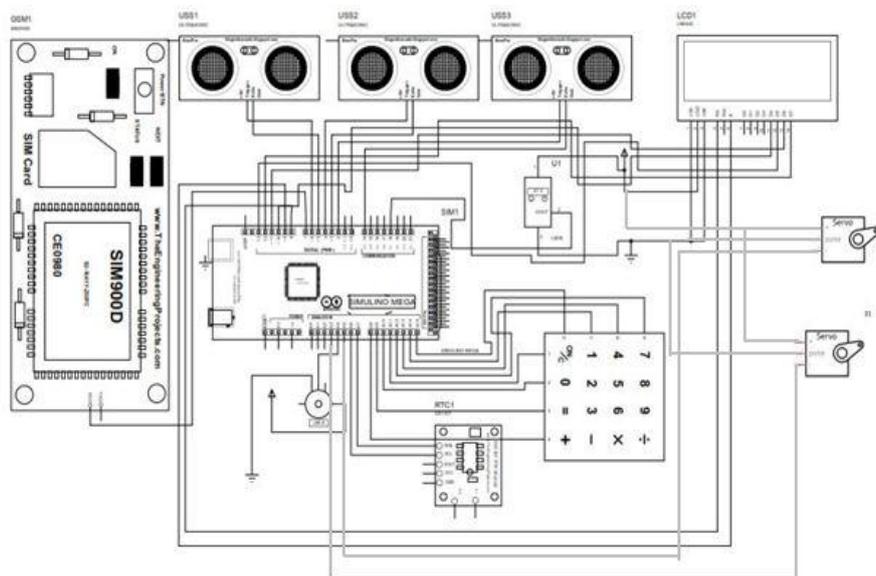
lcd.setCursor(0,0); printAngka (dataTanggal);
lcd.print ("/"); lcd.print (dataBulan);
lcd.print ("/"); lcd.print (dataTahun);
lcd.setCursor(12,0); printAngka (dataJam);
lcd.print (":"); printAngka (dataMenit);
lcd.print (":"); printAngka (dataDetik);
delay(500);
}

```

Gambar 3.19 Script Program *Liquid Crystal Display 20X4*

1.2.3.10 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 3.20



Gambar 3.20 Rangkaian Keseluruhan

Dari rangkaian keseluruhan dapat diketahui sistem kerja alat yaitu: Pertama pemilik harus menyetting jadwal pemberi pakan melalui keypad, jika sudah disetting maka arduino akan melakukan penyimpanan data jadwal pemberi pakan tersebut. Agar jadwal dapat berkerja dengan baik maka perlu digunakan RTC

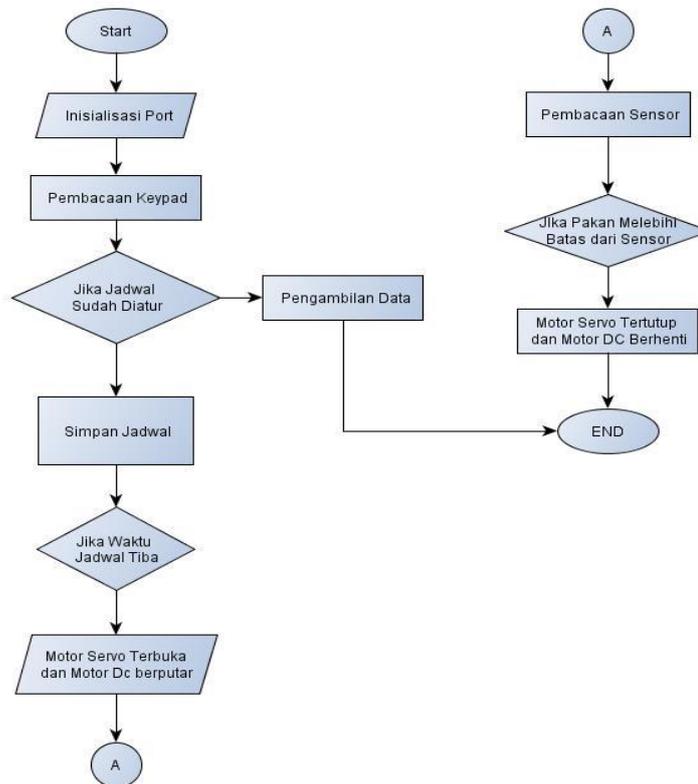
sebagai penghitung jam pemberi pakan pada kandang ayam. Sedangkan sensor ultrasonik digunakan sebagai pengukur isi pakan yang ada di dalam tandon penampung dan mengukur batas pemberian pakan pada tempat pakan ayam, motor servo digunakan sebagai pembuka dan penutup tandon pakan ayam sehingga tidak terjadinya pengisian terus menerus dan motor DC digunakan sebagai pendorong pakan agar dapat mengalir melalui paralon yang digunakan sebagai pengisi tempat wadah pakan ayam. Sedangkan *GSM Shield* digunakan sebagai output dalam memberi informasi jika isi pakan pada tandon sudah akan habis. Sensor DHT11 digunakan sebagai pengukur suhu ruangan didalam kandang ayam yang akan ditampilkan pada LCD 20x4.

1.2.4 Perancangan Perangkat Lunak

Perancangan perangkat lunak dibuat dari pembuatan *flowchart* untuk pembuatan pada *hardware*. Pada gambar 3.21. akan ditampilkan *flowchart* dari program yang akan dibuat dalam penelitian ini.

7. *End* adalah Akhir dari system

Dibawah ini merupakan flowchat kerja sistem dalam melakukan pengisian pakan ayam. Rancangan flowchat pengisian pakan dapat dilihat pada gambar 3.22.

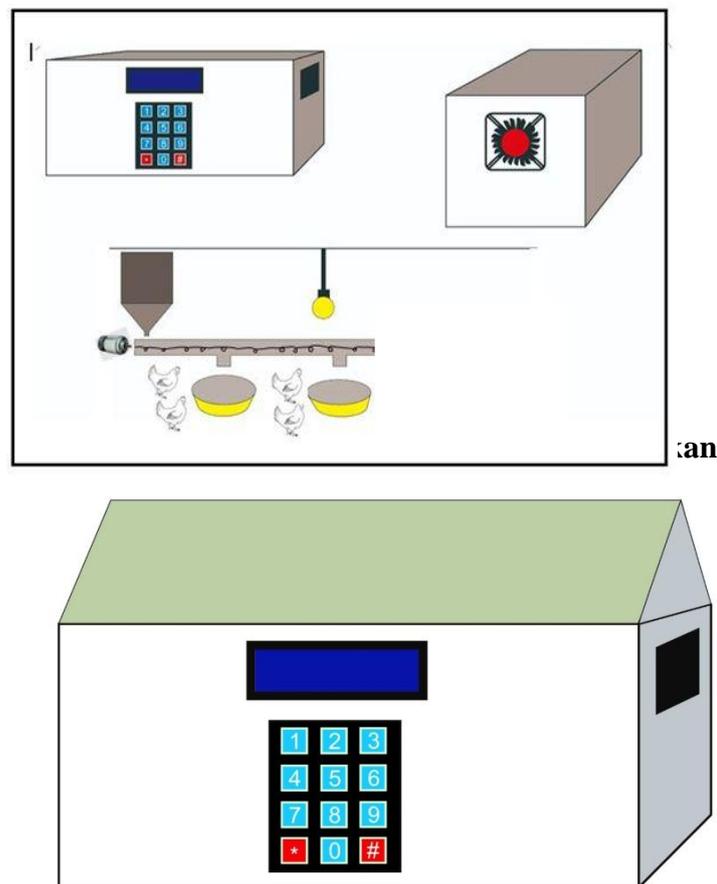


Gambar 3.22. Flowcart Pengisian Pakan

1. Start Memulai sistem yang akan bekerja.
2. Inisialisasi Merupakan proses pengolahan data dari pin input ataupun output dari memori.
3. Periksa keypad Mengecek apakah keypad yang akan digunakan berjalan dengan sebagaimana mestinya.
4. Jika jadwal pemberi pakan sudah diatur maka akan disimpan dalam memori.
5. Sedangkan jika waktu sudah sesuai dengan jadwal yang dibuat maka motor servo dan motor DC akan bergerak yang digunakan sebagai pengisian pada tempat-tempat pakan ayam yang ada dikandang.
6. Jika dalam melakukan pengisian pakan telah mendekati sensor maka motor servo dan motor DC akan berhenti.

7. Jika jarak pada sensor ultrasonik yang ada di atas tandon pengisian menunjukan pakan telah habis maka akan mengirimkan SMS kepemilik untuk melakukan pengisian tandon pengisian pakan.
8. *End* adalah Akhir dari sistem.

Perancangan simulasi sistem yang sangat penting dilakukan dalam pembuatan alat sehingga dengan adanya gambar simulasi dari alat akan mudah dalam melakukan perakitan. *Simulasi* sistem pengontrol suhu dan pemberi pakan ayam otomatis dapat dilihat mulai dari gambar 3.15 sampai dengan 3.23.



Gambar 3.24 Simulasi Tempat Kandang Ayam

1.3 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 di lakukan pada bagian-bagian seperti pengujian respon, jangkauan sistem, catu daya dan rangkaian keseluruhan pada sistem ini.

1.3.1 Pengujian Program Arduino

Pada *Software* Arduino program ditulis kemudian *dicompile*, tujuannya adalah untuk mengetahui apakah program yang dibuat sudah benar atau belum. Langkah terakhir yaitu meng-*upload* program kedalam modul *mikrokontroller*. Pada penelitian ini program yang dibuat, dirancang untuk mendeteksi suhu dan pemberi pakan ayam secara otomatis 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 di lakukan pada bagian-bagian seperti pengujian *respon* dan rangkaian keseluruhan pada sistem.

1.3.2 Rancangan Pengujian Sensor DHT11

Pengujian sensor DHT11 bertujuan untuk mengetahui keakuratan sensor DHT11 dalam mengukur suhu pada ternak ayam. Dalam melakukan perbandingan disini peneliti akan menggunakan pengukur suhu digital.

1.3.3 Rancangan Pengujian Ultrasonik

Pengujian ultrasonik bertujuan untuk mengetahui apakah pembacaan jarak yang dihasilkan oleh sensor ultrasonik telah baik, sensor ultrasonik akan digunakan sebagai pembuka dan penutup pakan jika tempat pakan telah habis dan penuh. Serta peneliti akan mengukur jarak yang dapat dijangkau oleh sensor ultrasonik.

1.3.4 Rancangan Pengujian RTC

Pengujian RTC bertujuan untuk mengetahui respon dan keakuratan RTC DS1307 dalam melakukan penjadwalan, apakah RTC dapat berkerja dengan baik yaitu dapat melakukan penjadwalan sesuai dengan yang ada pada perogram arduino.

1.3.5 Rancangan Pengujian Keypad

Pengujian keypad ini bertujuan untuk apakah keypad dapat di gunakan dengan baik sebagai setting penjadwalan pakan sehingga perlu dilakukan ujicoba agara

peneliti mengetahui apakah sistem penyetingan telah sesuai dengan program yang ada pada arduino.

1.3.6 Rancangan Pengujian Relay

Pengujian Relay ini bertujuan untuk mengetahui apakah lampu dan kipas dapat bekerja dengan baik sebagai pendingin dan pemanas pada kandang, sehingga perlu dilakukan ujicoba agar peneliti mengetahui apakah penyetingan system sesuai dengan program arduino yang telah dibuat.

1.3.7 Rancangan Pengujian Motor Servo

Pengujian rangkaian motor servo bertujuan untuk mengetahui apakah motor servo dapat bekerja memutar dari sudut 0° sampai 75° dan kebalikannya dari 75° ke 0° . Agar mengetahui apakah rangkaian motor servo telah berkerja sesuai dengan program arduino yang telah dibuat.

1.3.8 Rancangan Pengujian *Gsm Shield*

Rancangan pengujian *GSM Shield* bertujuan untuk mengetahui ketika *GSM Shield* mendapat perintah dari input apakah dapat mengirimkan SMS kepada pemilik ternak dan mengetahui waktu respon yang dibutuhkan untuk pengiriman SMS, dalam melakukan ujicoba *GSM Shield* peneliti menggunakan timer untuk mengukur waktu respon.

1.3.9 Rancangan Pengujian LCD 20x4

Pengujian LCD bertujuan sebagai *output* untuk menampilkan informasi berbentuk data yang berasal dari sumber masukan dari *sensor DHT11 dan RTC* yang telah diolah oleh Arduino.

1.3.10 Pengujian Sistem Keseluruhan

Pengujian sistem secara keseluruhan bertujuan untuk memastikan semua komponen dapat berjalan dengan sempurna. Mulai dari power supply, sensor DHT11, *Motor Servo, motor DC, GSM Shield RTC, LCD, Keypad* blok sistem arduino *At mega* dan program yang mengatur jalannya sistem keseluruhan.

1.4 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 Pengontrol Suhu Dan Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Dengan Output SMS berbasis mikrokontroler. 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

Bab ini berisi tentang hasil uji coba dan analisis terhadap sistem. Pengujian dimulai dengan memastikan setiap komponen (arduino, sensor DHT 11, motor servo, keypad, sensor ultrasonik dan motor dc) apakah alat yang telah dirangkai dalam kondisi bagus dapat bekerja dengan baik sesuai dengan program yang telah dibuat, kemudian mengecek setiap jalur yang terhubung dengan komponen yang digunakan telah terkoneksi, dimana rangkaiannya disesuaikan dengan gambar skematiknya. Pengujian yang dilakukan meliputi pengujian sensor DHT 11, motor servo, keypad, sensor ultrasonik dan motor dc dan pengujian sistem keseluruhan.

4.1 Hasil Uji Coba

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

4.1.1 Hasil Pengujian dan Pembahasan

Pada pengujian ini meliputi pengujian Sensor warna, Sensor *ultrasonik*, *RTC DS1307*, *keypad*, *motor servo*, *GSM Shield* motor DC dan rangkaian keseluruhan. Pengujian ini dilakukan agar peneliti dapat mengetahui kelebihan dan kekurangan sistem yang telah di buat hasil pengujian sebagai berikut:

4.1.2 Pengujian Sensor DHT 11

Pada pengujian sensor *DHT 11* dilakukan untuk mengetahui apakah sensor suhu dapat berfungsi dengan baik dalam membaca suhu yang ada pada kandang ayam, dan apakah sensor dapat dengan baik dalam melakukan penyalan relay. Hasil ujicoba dapat dilihat seperti pada tabel 4.1.

Tabel 4.1 Pengujian Sensor DHT 11

Uji Coba	Statu Sensor DHT 11	Status Relay		Keterangan
		1	2	
1	>34°	High		Kipas Menyala
2	<34°		High	Lampu Menyala

Dari hasil ujicoba dapat diketahui jika suhu > 34° maka arduino akan merintahkan *relay* agar menyalakan *relay* 1 yang digunakan sebagai menyalakan kipas yang artinya suhu dalam keadaan panas, serta jika suhu pada kandang ayam <34 °maka kandang ayam dinyatakan sangat dingin sehingga *relay* 2 akan berstatus high yang akan menyalakan lampu. Hasil pengukuran akan ditampilkan pada LCD 20x4

4.1.3 Pengujian Sensor Ultrasonik

Pada pengujian sensor *ultrasonik* dilakukan untuk mengetahui apakah sensor jarak dapat berfungsi dengan baik dalam membaca jarak pada tempat pakan dan tandon pengisian pakan, dalam penelitian ini peneliti akan melakukan ujicoba mulai dari jarak 2 cm hingga jarak 16 cm pada tempat pakan dan tandon pengisian dalam melakukan ujicoba jarak sensor ultrasonik peneliti menggunakan penggaris yang digunakan sebagai pengukur jarak seperti pada tabel 4.2.

Tabel 4.2 Pengujian Sensor Ultrasonik

Status Sensor Ultrasonik	Jarak (Cm) Sensor Ultrasonik	Keterangan
Sensor Ultrasonik 1	< 8	Pakan 1 Penuh
	>8	Pakan 1 sedang mengisi (pakan habis).
Sensor Ultrasonik 2	< 8	Pakan 2 Penuh
	>8	Pakan 2 sedang mengisi (pakan habis).
Sensor Ultrasonik 3	14	Tandon pengisian pakan dalam keadaan habis
	8	Tandon dalam keadaan hampir habis
	4	Tandon dalam keadaan penuh

4.1.4 Pengujian Servo

Pengujian *Servo* yaitu bertujuan untuk mengukur respon ketika motor servo membuka dan menutup tempat pengisian pakan. Dari hasil pengujian dari *motor servo* yang telah dilakukan dilihat pada tabel 4.3 berikut.

Tabel 4.3 Pengujian Motor Servo

Sudut yang diinginkan	Pembacaan busur derajat	Error (%)	Keteranngan
0°	0°	0	Nutup
45°	50°	11,11	Buka setengah
90°	90°	0	Buka Full

Dalam ujicoba *motor servo* peneliti melakukan ujicoba mulai dari 0° sampai dengan 90°. Peneliti mendapatkan hasil ujicoba yaitu dalam pengukuran ujicoba pertama dengan sudut yang diinginkan 0° dan pada pembacaan menggunakan busur hasil yang didapat tidak mengalami *error*. Sedangkan pada ujicoba kedua peneliti melakukan ujicoba pada sudut 45° yang dimana hasil pembacaan pada busur mengalami *error* sebanyak 11,11% (50°) dan pada ujicoba ke3 peneliti melakukan ujicoba dengan sudut 90° yang dimana pembacaan pada penggaris busur tidak mengalami *error*. Dalam ujicoba *motor servo* peneliti menggunakan penggaris busur sebagai perbandingan derajat *motor servo*.

4.1.5 Pengujian RTC

Pada pengujian ini dibutuhkan RTC 1307 yang akan digunakan sebagai penjadwalan waktu pemberi pakan ayam yang akan dilakukan 2 kali penjadwalan yaitu pagi dan sore hari. Hasil dari penjadwalan RTC akan ditampilkan pada LCD 20x4.

Tabel 4.4. Hasil Pengujian RTC

Uji Coba ke	Tampilan LCD Waktu	Status Servo		Motor DC	Jarak Sensor Ultrasonik CM		Keterangan
		1	2		1	2	
1	Jam 7.00-7.15	Tertutup		Berputar	<8		Pakan 1 Dalam Kondisi Penuh
		Terbuka		Berputar	>8		Sedang Mengisi pakan
			Tertutup	Berhenti		<8	Pakan 2 Dalam Kondisi Penuh
			Terbuka	Terbuka		>8	Sedang 2 Mengisi pakan
2	Jam 16.15-16.30	Tertutup		Berputar	<8		Pakan 1 Dalam Kondisi Penuh
		Terbuka		Berputar	>8		Sedang Mengisi pakan
			Tertutup	Berhenti		<8	Pakan 2 Dalam Kondisi Penuh
			Terbuka	Terbuka		>8	Sedang 2 Mengisi pakan

4.1.6 Pengujian Setting Keypad

Pengujian sensor *keypad* yaitu bertujuan untuk mengetahui kode dalam setting melalui keypad yang akan digunakan sebagai pengatur pemberi pakan ayam. Dari hasil pengujian setting *keypad* yang telah dilakukan dapat dilihat pada tabel 4.5 berikut.

Tabel 4.5. Pengujian *Setting keypad*

Tekan keypad	Tampilan LCD	Keteangan
*	Pilih waktu pakan 1 dan 2	
1	Setting waktu pakan 1 07000715	Jadwal pakan 1 jam 07.00 WIB - 07.15
2	Setting waktu pakan 2 16151630	Jadwal pakan 2 jam 16.15 WIB - 16.30

Berdasarkan hasil pengujian *setting keypad* dapat diketahui jika untuk masuk ke menu atur waktu pakan, tekan tombol * pada *Keypad* dan nanti akan menampilkan 2 pilihan waktu yang dipilih. Untuk memilih waktu pemberian pakan pertama tekan *Keypad* 1 dan untuk memilih waktu pemberian pakan kedua tekan *Keypad* 2. Ketika tombol *Keypad* 1 atau 2 ditekan, selanjutnya dilakukan penekanan tombol antara 0 – 9 untuk menentukan waktunya pemberian pakan. Tampilan pada LCD ketika pengetikan waktu seperti ini “07000715”, dimana artinya itu pemberian pakan kesatu dilakukan dari jam 07.00 sampai 07.15 sedangkan pada pakan kedua yaitu “16151630”. artinya itu pemberian pakan kedua dilakukan dari jam 16.15 sampai 16.30.

4.1.7 Pengujian *GSM Shield*

pengujian ini bertujuan untuk memastikan bahwa jika *sensor ultrasonik* 1 mendeteksi, jika pakan pada tandon pengisian telah habis apakah *GSM Shield* dapat dengan baik untuk mengirimkan SMS serta akan meenguji coba berapa lama respon yang dibutuhkan *GSM Shield* untuk mengirim SMS hasil pengujian pada tahap ini akan ditampilkan pada gambar berikut :

Jarak sensor ultrasonik 3	Status <i>Gsm Shield</i>	Waktu respon pengiriman SMS
2	<i>Gsm Shield</i> tidak Mengirim SMS	-
4	<i>Gsm Shield</i> tidak Mengirim SMS	-
6	<i>Gsm Shield</i> tidak Mengirim SMS	-
8	<i>Gsm Shield</i> tidak Mengirim SMS	-
10	<i>Gsm Shield</i> tidak Mengirim SMS	-
12	<i>Gsm Shield</i> tidak Mengirim SMS	-
14	<i>Gsm Shield</i> Mengirim SMS	5 detik
16	<i>Gsm Shield</i> Mengirim SMS	5 detik

Tabel 4.6. Hasil Pengujian Gsm Shield

Dari hasil tabel diatas dapat diketahui yaitu jika jarak *sensor ultrasonik* >2 CM dan <14 Cm maka *sensor uultrasonik* tidak akan mengirimkan SMS kepada pemilik sedangkan Jika jarak *sensor ultrasonik* >14 CM maka sensor mendeteksi bahwa pakan pada tandon pengisian telah habis sehingga *GSM Shield* akan mngirimkan SMS kepemilik agar segera mengisi Tadon pakan, waktu yng dibutuhkan dalam pengriminan SMS kepada pemilik yaitu 5 detik.

4.2 Pengujian Sistem Secara Keseluruhan

Pengujian sistem secara keseluruhan dilakukan untuk menguji kinerja Sistem pengontrol suhu dan pemberi pakan ayam otomatis pada perternakan ayam dengan output SMS berbasis mikrokontroler , dilakukan ujicoba sistem agar peneliti dapat mengetahui apakah sistem yang telah dibuat dapat berkerja dengan baik. Dari hasil ujicoba sistem dapat di ketahui bahwa sistem dapat berkerja dengan baik sesuai perintah pada program yang telah dibuat dapat dilihat seperti pada tabel 4.7. berikut hasil pengujian sistem keseluruhan.

Tabel 4.7. Hasil Pengujian Sistem Keseluruhan

Status Pakan	Jarak Sensor CM			Status Servo		Status Motor DC	GSM Shield	Status Suhu	Status Relay	
	1	2	3	1	2				1	2
Pakan 1	<8	-	2	Tertutup	-	Berputar	Tidak Mengirim SMS	34°	High	Low
	>8		4	Terbuka	-	Berputar	Tidak Mengirim SMS	35°	High	Low
Pakan 2		<8	8		Tertutup	Berhenti	Tidak Mengirim SMS	33°	Low	High
		>8	10		Terbuka	Berputar	Tidak Mengirim SMS	30°	Low	High
Tandon Pakan	6	7	14	Tertutup	Tertutup	Berhenti	Mengirim SMS	30°	Low	High
	9	12	16	Terbuka	Terbuka	Berputar	Mengirim SMS	34°	High	Low

Dari hasil uji coba sistem keseluruhan dapat diketahui yaitu jika pakan 1 dengan jarak *sensor ultrasonik* 1 < 8 maka motor *servo* tertutup, motor dc berputar, sedangkan jika jarak ultrasonic 1 dan 2 > 8 maka motor servo akan terbuka dan jika jarak *sensor ultrasonik* 2 $<$ maka motor *servo* tertutup serta motor DC akan berhenti dan jika ultrasonic 3 berjarak 2 sampai 12 cm maka *GSM Shield* tidak akan mengirimkan SMS yang artinya tandon pengisian pakan masih dalam keadaan penuh, jika hasil pengukuran sensor ultrasonic 3 > 12 maka *GSM Shield* akan mengirim SMS yang artinya tandon pengisian pakan dalam kondisi habis . Pada ujicoba pengukuran suhu pada kandang ayam dapat diketahui jika suhu < 34 maka *relay* 1 (lampu) akan menyala serta kipas akan mati sedangkan jika suhu kandang ayam > 34 maka *relay* 2 (kipas) akan menyala dan lampu akan mati. Hasil dari pembacaan system akan ditampilkan pada LCD 20x4.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

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

1. Dari hasil uji coba sistem keseluruhan dapat diketahui jika sensor ultrasonik 1 dan 2 akan mengukur ketinggian pakan <8 cm yang artinya pakan dalam kondisi penuh sehingga motor servo akan tertutup dan motor DC akan berhenti.
2. *Gsm shield* akan mengirimkan SMS jika *sensor ultrasonik 3* yang digunakan sebagai pengukur ketinggian tendon pakan >12 CM.
3. Sensor DHT 11 yang digunakan sebagai pengukur suhu kandang akan berkerja menyalakan kipas jika suhu kandang $>34^{\circ}$ dan jika suhu $<34^{\circ}$ maka lampu akan menyala.

5.2 Saran

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

1. Untuk penelitian selanjutnya disarankan agar menggunakan sensor lain yang lebih *efektif* dalam mengukur ketinggian pakan.
2. Diharapkan peneliti selanjutnya dapat mengembangkan dengan ditambahkan *convayer* yang akan digunakan sebagai pengisian jika tandon pada pakan telah dalam keadaan habis.

DAFTAR PUSTAKA

Ardiansyah. (2018). *Rancang Bangun Sistem Pemberian Pakan Ternak (Sapi) Dan Pengadukannya Secara Otomatis Berbasis Mikrokontroler*. Universitas Islam Negeri Alauddin Makasar.

Arduino. (2016). *Arduino Uno & Geniuno Uno*. Dipetik Mei 6, 2016, dari Arduino Website: <https://www.arduino.cc/en/main/arduinoBoardUno>

Haryanto, E. (2016). *Perancangan dan implementasi alat pemberi makan ikan otomatis berbasis mikrokontroler at89s52*. Universitas Padang.

Hendra. (2015). *Rancang bangun alat pemberi pakan ikan Otomatis berbasis mikrokontroler*. *Teknik Elektro*, vol 2.

Nur Komala Sari. (2011). *Rancang Bangun Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Berbasis Mikrokontroller*. *Teknik Telekomunikasi*, Vol 1.

Nur Komala Sarig. (2015). *Rancang Bangun Pemberi Pakan Ayam Otomatis Pada Peternakan Ayam Berbasis Mikrokontroller*. *Teknologi Elektro*.

Sebayang, R. K. (2015). *Perancangan Sistem Pengaturan Suhu Kandang Ayam Berbasis Mikrokontroler*. *Teknik Elektro*.

Yenni, H. (2016). *Perangkat pemberi pakan otomatis Pada kolam budidaya*. *Sistem Komputer*.

Saragih, Astriani Romaria (2016). *pengembangan alat rancang bangun perangkat pemberi pakan ikan otomatis pda kolam pembenih ikan berbasis arduino*, *Teknik Elektro*, Kepulauan Riau.

Riza, Faishol. 2011. *Perancangan Sistem Pengendali Suhu dan Memonitoring Kelembaban Berbasis Atmega8535 pada Plant Inkubator*. Universitas Diponegoro. Semarang.

Badan Pusat Statistik, 2017. Populasi Ayam Ras Pedaging Menurut Provinsi 2009-2016. [online]

Available at: <https://www.bps.go.id/linkTableDinamis/view/id1034>

CANDRA, R., 2015. *Rancang Bangun Sistem Pengendalian dan Monitoring Peralatan Listrik Jarak Jauh Berbasis Android*. Bandar Lampung: Institut Informatika dan Bisnis Darmajaya.

Yoga Alif Kurnia Utama, S. M., 2016. Perbandingan Kualitas Antara Sensor Suhu dengan Menggunakan Arduino Pro Mini. *e – Jurnal NARODROID*, 2(2), P. 147.

Wibowo, F. Y. H., 2017. *Pembuatan Sistem Kontrol Suhu Berbasis Mikrokontroler Arduino*, Bogor: Institut Pertanian Bogor.

LAMPIRAN

Lampiran1. Program lengkap

```
#include "DHT.h"

#define DHTPIN A1 // what pin we're connected to
#define DHTTYPE DHT11 // DHT 11
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println("DHT test!");

  dht.begin();
}
void loop() {
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  float f = dht.readTemperature(true);
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }

  float hif = dht.computeHeatIndex(f, h);
  float hic = dht.computeHeatIndex(t, h, false);
  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.print(" %\t");
  Serial.print("Temperature: ");
  Serial.print(t);
  Serial.print(" *C ");
  Serial.print(f);
  Serial.print(" *F\t");
  Serial.print("Heat index: ");
  Serial.print(hic);
  Serial.print(" *C ");
  Serial.print(hif);
  Serial.println(" *F");
}
}
```

```

// defines pins numbers
const int trigPin1 = 2;
const int echoPin1 = 3;
const int trigPin2 = 4;
const int echoPin2 = 5;
const int trigPin3 = 6;
const int echoPin3 = 14;

// defines variables
int distance1,distance2,distance3;
int looppertama,loopkedua;
boolean hasil1 = false;
boolean hasil2 = false;
boolean hasil3 = false;
boolean hasil4 = false;
boolean hasil5 = false;
boolean hasil6 = false;
boolean makan1 = false;
boolean makan2 = false;

void setup() {

  Serial.begin(9600); // Starts the serial communication
  pinMode(trigPin1, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin1, INPUT); // Sets the echoPin as an Input
  pinMode(trigPin2, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin2, INPUT); // Sets the echoPin as an Input
  pinMode(trigPin3, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin3, INPUT); // Sets the echoPin as an Input
  pinMode(lampu, OUTPUT); // Sets the trigPin as an Output
  hasil1=true;
  hasil4=true;
  makan1=true;

}

void loop() {

int getDistance (int trigPin, int echoPin){
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  unsigned long pulseTime = pulseIn(echoPin, HIGH);
  int distance = pulseTime*0.034/2;
  return distance;
}

void printDistance(int id, int dist){

```

```

Serial.print(id);
Serial.print("----->");
Serial.print(dist, DEC);
Serial.println(" cm");

}

void pakanPertama(){
  if(hasil1 == true){
    digitalWrite(lampu,HIGH);
    distance1 = getDistance(trigPin1, echoPin1);
    printDistance(1, distance1);
    delay(150);
    if(distance1 <= 7){
      Serial.println("Pakan 1 Telah Terpenuhi");
      delay(2000);
      digitalWrite(lampu,LOW);
      hasil2=true; hasil1=false;
    }
  }
  else if(hasil2 == true){
    digitalWrite(lampu,HIGH);
    distance2 = getDistance(trigPin2, echoPin2);
    printDistance(2, distance2);
    delay(150);
    if(distance2 <= 7){
      Serial.println("Pakan 2 Telah Terpenuhi");
      delay(2000);
      digitalWrite(lampu,LOW);
      hasil3=true; hasil2=false;
    }
  }

  else if(hasil3 == true){
    digitalWrite(lampu,HIGH);
    distance3 = getDistance(trigPin3, echoPin3);
    printDistance(3, distance3);
    delay(150);
    if(distance3 <= 7){

      delay(2000);

      digitalWrite(lampu,LOW);
      hasil1 = true; makan2=true; makan1=false; hasil3=false;
    }
  }
}
}

```

```

void pakanKedua(){
  if(hasil4 == true){
    Serial.println("baca PAKAN KEDUA DIMULAI");
    distance1 = getDistance(trigPin1, echoPin1);
    printDistance(1, distance1);
    delay(150);
    if(distance1 <= 7){

      Serial.println("Pakan 1 Telah Terpenuhi");
      delay(2000);
      hasil5=true; hasil4=false;
    }
  }
  else if(hasil5 == true){
    distance2 = getDistance(trigPin2, echoPin2);
    printDistance(2, distance2);
    delay(150);
    if(distance2 <= 7){
      Serial.println("Pakan 2 Telah Terpenuhi");
      delay(2000);

      hasil6=true; hasil5=false;
    }
  }

  else if(hasil6 == true){
    distance3 = getDistance(trigPin3, echoPin3);
    printDistance(3, distance3);
    delay(150);
    if(distance3 <= 7){

      Serial.println("Pakan 3 Telah Terpenuhi");
      delay(2000);

      hasil4 = true; makan1=true; makan2=false; hasil6=false;
    }
  }
}

```

```

void setup() {
  rtc.begin();
  Serial.begin(9600); // Starts the serial communication
  rtc.setDOW(4);      //Setting hari RTC
  rtc.setDate(15,02,2019); //Setting tanggal RTC (tanggal,bulan,tahun)
  rtc.setTime(7,59,56); //Setting waktu RTC (jam,menit,detik)
}

```

```

void loop() {
  rtcDS3231();
  if((dataJam == 8 && dataMenit >= 0) && (dataJam == 8 && dataMenit
<= 1) && makan1 == true){
    Serial.println("Pemberian Pakan Pertama");
    pakanPertama();
  }
  else if((dataJam == 8 && dataMenit >= 3) && (dataJam == 8 &&
dataMenit <= 4) && makan2 == true){
    Serial.println("Pemberian Pakan Kedua");
    pakanKedua();
  }
  else if((dataJam == 8 && dataMenit >= 6) && (dataJam == 8 &&
dataMenit <= 7) && makan1 == true){
    Serial.println("Pemberian Pakan Pertama LAGII!!!");
    pakanPertama();
  }
}

int getDistance (int trigPin, int echoPin){
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  unsigned long pulseTime = pulseIn(echoPin, HIGH);
  int distance = pulseTime*0.034/2;
  return distance;
}

void rtcDS3231(){
  waktu = rtc.getTime();
  dataJam = waktu.hour;
  dataMenit = waktu.min;
  dataDetik = waktu.sec;
  dataTanggal = waktu.date;
  dataBulan = rtc.getMonthStr(1);
  dataTahun = waktu.year;
  Hari = rtc.getDOWStr();
  Serial.print(Hari); Serial.print(',');
  Serial.print(dataJam); Serial.print(':');
  Serial.print(dataMenit); Serial.print(':');
  Serial.print(dataDetik); Serial.print(" -- ");
  Serial.print(dataTanggal); Serial.print('/');
  Serial.print(dataBulan); Serial.print('/');
  Serial.println(dataTahun);
}
}

```

```

#include <Keypad.h>
#include <EEPROM.h>

const byte numRows= 4;
const byte numCols= 4;;
char key;

char keymap[numRows][numCols]=
{
{'1', '2', '3', 'A'},
{'4', '5', '6', 'B'},
{'7', '8', '9', 'C'},
{'*', '0', '#', 'D'}
};

byte rowPins[numRows] = {35,34,33,32};
byte colPins[numCols]= {31,30,29,28};
unsigned int loopMenu=0;
char a[3],b[3],c[3],d[3],e[3],f[3],g[3],h[3];
int alamat=1,j=0;
int add1=1,add2=3,add3=5,add4=7,add5=9,add6=11,add7=13,add8=15;
boolean cocok=false;

Keypad myKeypad= Keypad(makeKeypad(keymap), rowPins, colPins,
numRows, numCols);

void setup(){
  Serial.begin(9600);
}

void loop(){
  //baca();
  key = myKeypad.getKey();
  if(key) Serial.println(key);
}

void baca(){
  key = myKeypad.getKey();
  if(key=='*'){
    cocok=false;
    loopMenu=0;
    tulisMenu();
  }
  if(key=='A'){
    cocok=true;
  }
}

```

```

if(cocok == true){
  rtcDS3231();
  for(int x=0;x<2;x++){
    a[x]=EEPROM.read(add1);
    add1++;
    if(add1>=3)add1=1;
    b[x]=EEPROM.read(add2);
    add2++;
    if(add2>=5)add2=3;
    c[x]=EEPROM.read(add3);
    add3++;
    if(add3>=7)add3=5;
    d[x]=EEPROM.read(add4);
    add4++;
    if(add4>=9)add4=7;
    e[x]=EEPROM.read(add5);
    add5++;
    if(add5>=11)add5=9;
    f[x]=EEPROM.read(add6);
    add6++;
    if(add6>=13)add6=11;
    g[x]=EEPROM.read(add7);
    add7++;
    if(add7>=15)add7=13;
    h[x]=EEPROM.read(add8);
    add8++;
    if(add8>=17)add8=15;
  }
  int jam1=atoi(a); int menit1=atoi(b);
  int jam2=atoi(c); int menit2=atoi(d);
  Serial.print(jam1); Serial.print(" // "); Serial.print(menit1); Serial.print("
<<>> "); Serial.print(jam2); Serial.print(" // "); Serial.println(menit2);
  int jam3=atoi(e); int menit3=atoi(f);
  int jam4=atoi(g); int menit4=atoi(h);
  Serial.print(jam3); Serial.print(" // "); Serial.print(menit3); Serial.print("
<<>> "); Serial.print(jam4); Serial.print(" // "); Serial.println(menit4);

  if((dataJam == jam1 && dataMenit >= menit1) && (dataJam == jam2
&& dataMenit <= menit2)){
    Serial.println("Pemberian Pakan Pertama");
    //pakanPertama();
  }
  else if((dataJam == jam3 && dataMenit >= menit3) && (dataJam ==
jam4 && dataMenit <= menit4)){
    Serial.println("Pemberian Pakan Kedua");
    cocok=false;
    //pakanKedua();
  }
}

```

```

    }
  }
}

void tulisMenu(){
  while(loopMenu<1){
    key = myKeypad.getKey();
    if(key=='#'){
      loopMenu=1;
      baca();
    }

    if(key=='1'){
      Serial.println("Pilih Atur Waktu Pertama");
      alamat=1;
      loopMenu=1;
      j=0;
      tulisWaktu();
    }
    if(key=='2'){
      Serial.println("Pilih Atur Waktu Kedua");
      alamat=9;
      loopMenu=1;
      j=0;
      tulisWaktu();
    }
  }
}

void tulisWaktu(){
  while(j<8){
    key = myKeypad.getKey();

    if(key=='1' || key=='2' || key=='3' || key=='4' || key=='5' || key=='6' || key=='7' || key=='8' || key=='9' || key=='0'){
      j++;
      EEPROM.write(alamat,key);
      alamat++;
      Serial.print(key); Serial.print(" >>> "); Serial.println(alamat);
      if(j==8){
        Serial.println("Jam Baru Selesai");
        baca();
        j=8;
      }
    }
  }
}
}

```

```

#include <Servo.h>
Servo myservo;
int pos = 0;
void setup()
{
myservo.attach(9);
}
void loop()
{
for(pos = 0; pos < 90; pos += 1)
{
myservo.write(pos);
delay(15);
}
for(pos = 90; pos >= 1; pos -= 1)
{
myservo.write(pos);
delay(15);
}
}
#include <SoftwareSerial.h>
//SoftwareSerial gsm(7,8); //RX | TX
char c = ' ';

void setup() {
// put your setup code here, to run once:
Serial.begin(9600);
while(!Serial);
Serial.println("Arduino with SIM900 is ready");
delay(1000);
Serial1.begin(9600);
Serial.println("SIM900 started at 9600");
delay(1000);
Serial.println("Setup Complete! SIM900 is Ready");
 kirimPesan();
}

void loop() {
// put your main code here, to run repeatedly:
/*if(Serial1.available()){
c=Serial1.read();
Serial.write(c);
}

if(Serial.available()){
c=Serial.read();
Serial1.write(c);
}*/

```

```

}

void kirimPesan(){
  Serial.println("SIM900 Mengirim Pesan");
  // AT command to set SIM900 to SMS mode
  Serial1.print("AT+CMGF=1\r");
  delay(100);

  // REPLACE THE X's WITH THE RECIPIENT'S MOBILE NUMBER
  // USE INTERNATIONAL FORMAT CODE FOR MOBILE NUMBERS
  Serial.println("SIM900 Set SMS Number");
  Serial1.println("AT + CMGS = \"+6289653430648\");
  delay(100);

  // REPLACE WITH YOUR OWN SMS MESSAGE CONTENT
  Serial.println("SIM900 SMS Konteks");
  Serial1.println("Message example from Arduino Uno.");
  delay(100);

  // End AT command with a ^Z, ASCII code 26
  Serial.println("SIM900 Finish");
  Serial1.println((char)26);
  delay(100);
  Serial1.println();
  // Give module time to send SMS
  delay(5000);
  Serial.println("SIM900 -> Selesai Mengirim Pesan");
}

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd (2,3,4,5,6,7);
void setup() {
  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);
  // Print a message to the LCD.
  lcd.print("hello, world!");
}
void loop() {
  // Turn off the display:
  lcd.noDisplay();
  delay(500);
  // Turn on the display:
  lcd.display();
  delay(500);
}

```