

**RANCANG BANGUN ALAT CERDAS PEMBERI PAKAN
IKAN DENGAN SISTEM TIGA RADIUS TEMBAKAN
BERBASIS ARDUINO UNO**

SKRIPSI

OLEH :

**ABDUL RAHMAN HARAHAHAP
148120014**



**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS MEDAN AREA
MEDAN
2020**

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah
3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area

Document Accepted 15/12/21

Access From (repository.uma.ac.id)15/12/21

**RANCANG BANGUN ALAT CERDAS PEMBERI PAKAN
IKAN DENGAN SISTEM TIGA RADIUS TEMBAKAN
BERBASIS ARDUINO UNO**

SKRIPSI

Diajukan sebagai Salah Satu Syarat untuk Memperoleh
Gelar Sarjana di Fakultas Teknik
Universitas Medan Area

Oleh :

**ABDUL RAHMAN HARAHAHAP
148120014**



**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS MEDAN AREA
MEDAN
2020**

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah
3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area

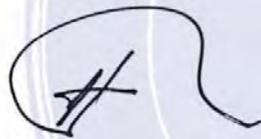
Document Accepted 15/12/21

Access From (repository.uma.ac.id)15/12/21

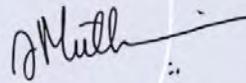
HALAMAN PENGESAHAN

Judul Skripsi : Rancang Bangun Alat Cerdas Pemberi Pakan Ikan Dengan Sistem Tiga Radius Tembakan Berbasis Arduino Uno
Nama : Abdul Rahman Harahap
NPM : 14.812.0014
Fakultas : Teknik

Disetujui oleh
Komisi pembimbing



Prof. Dr. Dadan Ramdan, M.Eng, M.Sc
Pembimbing I



Syarifah Muthia Putri, ST, MT.
Pembimbing II



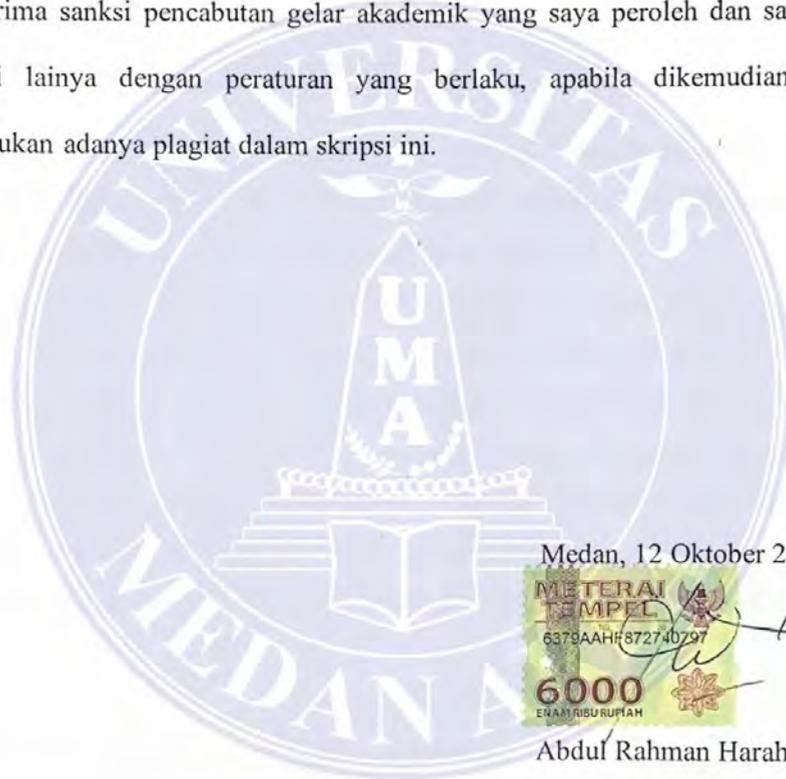
Dr. Ir. Dina Maizana, MT
Dekan



Syarifah Muthia Putri, ST, MT.
Ketua Program Studi

LEMBAR PERNYATAAN

Saya menyatakan bahwa skripsi yang saya susun, sebagai syarat memperoleh gelar sarjana merupakan hasil karya tulis saya sendiri dan bantuan orang lain. Adapun bagian – bagian tertentu dalam penulisan skripsi ini yang saya kutip dari hasil karya orang lain telah dituliskan sumbernya secara jelas sesuai dengan norma, kaidah dan etika penulisan ilmiah. Saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi – sanksi lainnya dengan peraturan yang berlaku, apabila dikemudian hari ditemukan adanya plagiat dalam skripsi ini.



Medan, 12 Oktober 2020



Abdul Rahman Harahap

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR/SKRIPSI/TESIS UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Medan Area, saya yang bertanda tangan di bawah ini:

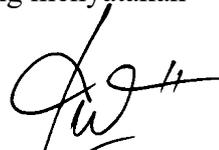
Nama : Abdul Rahman Harahap
NPM : 14.812.0014
Program Studi : Teknik Elektro
Fakultas : Teknik
Jenis karya : Tugas Akhir/Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Medan Area **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

“RANCANG BANGUN ALAT CERDAS PEMBERI PAKAN IKAN DENGAN SISTEM TIGA RADIUS TEMBAKAN BERBASIS ARDUINO UNO”
berserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Medan Area berhak menyimpan, mengalihmedia/format-kan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir/skripsi/tesis saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Medan
Pada tanggal : 12 Oktober 2020
Yang menyatakan

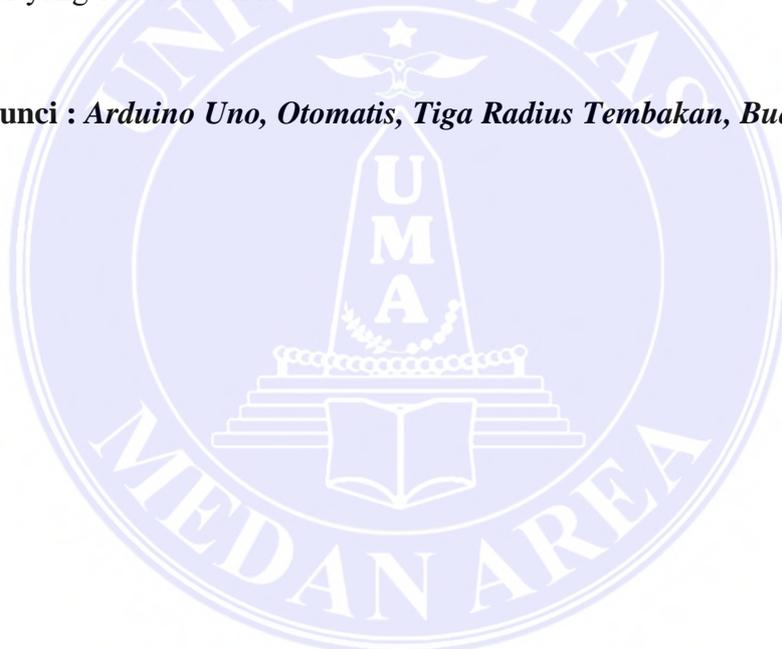


(Abdul Rahman Harahap)

ABSTRAK

Pembudidayaan ikan merupakan suatu kegiatan maupun sebuah usaha yang banyak digemari oleh masyarakat. Namun pemberian pakan yang masih manual oleh pembudidaya/peternak ikan sering mengalami kerugian dikarenakan pemberian pakan yang tidak terjadwal dan tidak merata ke seluruh kolam. Berdasarkan penjelasan tersebut dibutuhkan suatu alat pemberi pakan ikan yang membantu pembudidaya/peternak ikan dalam pemberian pakan secara teratur dan terjadwal serta merata ke seluruh kolam. Pada penelitian ini dibuat suatu alat pemberi pakan ikan otomatis yang dapat diatur waktu/jadwal pemberian pakan dengan sistem tiga radius tembakan berbasis arduino uno yang memungkinkan pakan dapat disebar ke seluruh kolam sehingga tidak menumpuk disatu titik. Metode pelaksanaan kegiatan ini terdiri dari beberapa tahapan yaitu, observasi awal, perancangan flowchart, perancangan rangkaian, proses pembuatan alat, dan pengujian alat. Kinerja sistem akan diuji fungsinya sesuai dengan flowchart yang telah didesain. Hasil pengujian disajikan berupa data jarak pelemparan pakan dengan massa dan diameter yang berbeda-beda.

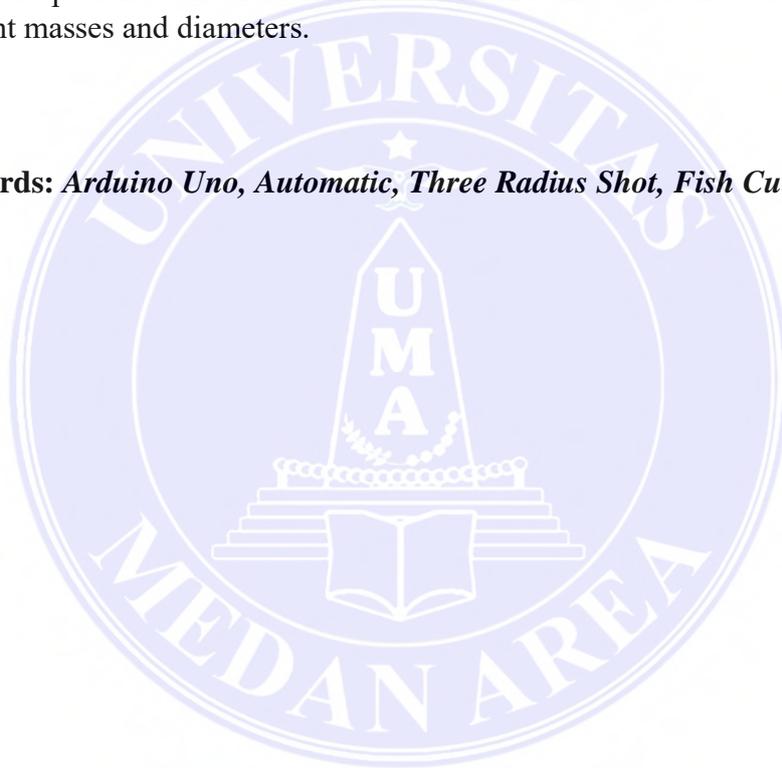
Kata kunci : *Arduino Uno, Otomatis, Tiga Radius Tembakan, Budidaya Ikan*



ABSTRACT

Fish farming is an activity or a business that is popular with the community. However, manual feeding by fish farmers often experiences a loss due to unscheduled and uneven feeding throughout the pond. Based on this explanation, a fish feeder is needed that helps fish farmers in providing regular and scheduled feeds and evenly throughout the pond. In this research, an automatic fish feeder that can be adjusted for the time / schedule of feeding with an Arduino Uno-based three radius shot system which allows feed to be distributed throughout the pond so that it does not accumulate at one point is made. The method of implementing this activity consists of several stages, that is initial observation, flowchart design, circuit design, tool making process, and tool testing. The function of the system will be tested in accordance with the flowchart that has been designed. The test results are presented in the form of data on the throw distance of the feed with different masses and diameters.

Keywords: *Arduino Uno, Automatic, Three Radius Shot, Fish Cultivation*



KATA PENGANTAR

Ucapan syukur kepada Tuhan Yang Maha Esa karena berkat rahmat-Nya penulis bisa menyelesaikan Skripsi yang berjudul Rancang Bangun Alat Cerdas Pemberi Pakan Ikan Dengan Sistem Tiga Radius Tembakan Berbasis Arduino Uno.

Tugas Akhir ini dibuat guna melengkapi persyaratan kurikulum pada Program Studi Teknik Elektro Strata Satu, Universitas Medan Area. Semoga hasil dari Skripsi ini ada manfaatnya bagi pihak yang berkepentingan. Pada Kesempatan ini, Penulis ingin menyampaikan ucapan terima kasih kepada :

1. Bapak Prof. Dr. Dadan Ramdan, M.Eng, M.Sc selaku Rektor Universitas Medan Area sekaligus Pembimbing I yang telah banyak meluangkan waktu, tenaga serta fikiran dalam membimbing penulis selama mengerjakan Skripsi ini.
2. Ibu Dr. Grace Yuswita Harahap, ST, MT selaku Dekan Fakultas Teknik.
3. Ibu Syarifah Muthia Putri, ST, MT selaku Ketua Program Studi Teknik Elektro sekaligus Pembimbing II yang telah banyak meluangkan waktu, tenaga serta fikiran dalam membimbing penulis selama mengerjakan Skripsi ini.
4. Seluruh Staff pengajar Universitas Medan Area khususnya Program studi Teknik Elektro.
5. Orang tua yang selalu memberi dukungan doa, motivasi dan materi selama penulis menyelesaikan Skripsi ini.
6. Sahabat penulis yang selalu memberikan dukungan, doa, hiburan dan semangat kepada penulis dalam menyelesaikan Skripsi ini.
7. Rekan-Rekan kelas saya terkhususnya buat mahasiswa teknik elektro angkatan yang banyak memberikan semangat serta doa kepada penulis dalam menyelesaikan Skripsi ini.
8. Semua Pihak yang memberikan dukungan dan semangat kepada penulis sehingga dapat menyelesaikan Skripsi ini.

9. Saudara – saudari penulis, Haritsah Hammamah Harahap, Harits al-Hammam Harahap, Abdah Rohimah Harahap, dan Rahmi Harahap yang banyak memberikan semangat serta doa kepada penulis dalam menyelesaikan Skripsi ini.

Medan, 12 Oktober 2020

Penulis



DAFTAR ISI

ABSTRAK	iii
ABSTRACT	v
KATA PENGANTAR	vi
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah	2
1.3. Tujuan dan Manfaat.....	2
1.4. Batasan Masalah.....	3
1.5. Sistematika Pembahasan.....	3
BAB II TEORI PENUNJANG	5
2.1. Arduino	5
2.2. Arsitektur Mikrokontroler <i>ATMega</i>	7
2.3. Definisi Program	9
2.4. Software Arduino IDE.....	11
2.5. Driver L298.....	12
2.6. Motor <i>DC</i>	14
2.7. Catu Daya	16
2.8. LCD 2x16	18
2.9. Keypad 3x4 Matrix.....	19
2.10. Soket <i>USB</i>	21
2.11. Rangkaian Tiny <i>RTC (Real Time Clock)</i>	21
BAB III METODOLOGI PENELITIAN	23
3.1. Tempat dan Waktu Penelitian	23
3.1.1. Tempat Penelitian	23
3.1.2. Waktu Penelitian.....	23
3.2. Metoda Penelitian.....	23
3.3. Alat dan Bahan.....	25
3.4. Rancangan Sistem Elektrikal	25
3.4.1. Sistem <i>AC-DC</i> Adaptor.....	26

3.4.2. Sistem Minimum Arduino Uno	27
3.4.3. Sistem Driver L298 dengan Aktuator	28
3.4.4. Sistem Arduino dengan Keypad 4x4 Matrix	29
3.4.5. Sistem Arduino dengan Motor Servo.....	30
3.4.6. Sistem Arduino dengan Penampil Data (LCD 2x16).....	31
3.4.7. Sistem secara Keseluruhan.....	32
3.5. Flowchart Sistem Kerja Alat.....	34
BAB IV HASIL DAN PEMBAHASAN.....	35
4.1. Hasil Pembuatan Alat.....	35
4.2. Pengujian Alat dan Analisa.....	36
4.2.1. Pengujian Modul Arduino Uno	37
4.2.2. Pengujian RTC	39
4.2.3. Pengujian <i>LCD 2x16</i>	42
4.2.4. Pengujian Driver L298 dengan Motor DC Gear Box.....	44
4.2.5. Pengujian Motor Servo	47
4.2.6. Pengujian Alat secara Keseluruhan	50
BAB V KESIMPULAN DAN SARAN.....	51
5.1. Kesimpulan	51
5.2. Saran	51
DAFTAR PUSTAKA.....	52
LAMPIRAN	53

DAFTAR GAMBAR

Gambar 2. 1	Arduino Uno	5
Gambar 2. 2	Mikrokontroler ATmega 328	7
Gambar 2. 3	Blok Diagram Mikrokontroler ATmega 328	8
Gambar 2. 4	Blok Diagram Mikrokontroler ATmega 1280	9
Gambar 2. 5	Tampilan Arduino IDE Versi 1.6.4.....	11
Gambar 2. 6	Modul Driver L298	12
Gambar 2. 7	Skema Sistem Kerja PWM.....	14
Gambar 2. 8	Bentuk Fisik dan Spesifikasi Motor DC Gearbox	15
Gambar 2. 9	Bentuk Fisik LCD	18
Gambar 2. 10	Keypad 3x4 Matrix	19
Gambar 2. 11	Rangkaian Dasar Keypad 3x4.....	20
Gambar 2. 12	Bentuk Fisik Soket USB.....	21
Gambar 2. 13	Bentuk Fisik Tiny RTC (Real Time Clock)	22
Gambar 3. 1	Flowchart Kerangka Berfikir.....	24
Gambar 3. 2	AC-DC Adaptor	26
Gambar 3. 3	Skema Rangkaian AC-DC Adaptor	26
Gambar 3. 4	Sistem Minimum Arduino Uno	27
Gambar 3. 5	Instalasi Driver L298 dengan Dua Motor DC Gearbox	28
Gambar 3. 6	Instalasi Driver L298 dengan Dua Motor DC Gearbox	29
Gambar 3. 7	Instalasi Arduino dengan Keypad.....	30
Gambar 3. 8	Rangkaian Instalasi Arduino dengan Motor Servo	31
Gambar 3. 9	Pola Instalasi LCD 2x16 pada Arduino Uno	32
Gambar 3. 10	Skema Rangkaian Seluruh Sistem	33
Gambar 3. 11	Flowchart Sistem Kerja Alat	34
Gambar 4. 1	Alat Cerdas Pemberi Pakan Ikan dengan 3 Radius Tembakan.....	35
Gambar 4. 2	Rangkaian Pengujian Modul Arduino dengan Indikator Led.....	38
Gambar 4. 3	Listing Program Pengujian Lampu Led	38
Gambar 4. 4	Listing Program Pengujian Lampu Led	39
Gambar 4. 5	Bentuk Pengujian RTC.....	40
Gambar 4. 6	Hasil Pengujian RTC.....	42
Gambar 4. 7	Pengujian Rangkaian LCD 2x16 Karakter	43
Gambar 4. 8	Listing Program Pengujian Rangkaian LCD 2x16	44
Gambar 4. 9	Rangkaian Pengujian Motor Servo Menggunakan Arduino	48

DAFTAR TABEL

Tabel 2. 1	Operasi Dasar LCD.....	19
Tabel 3. 1	Penetapan Komponen (Bahan).....	25
Tabel 4. 1	Lokasi Pin RTC	40
Tabel 4. 2	Tabel Pengujian 1	50
Tabel 4. 3	Tabel Pengujian 2	50
Tabel 4. 4	Tabel Pengujian 3	50



BAB I PENDAHULUAN

1.1. Latar Belakang Masalah

Budidaya ikan adalah salah satu kegiatan maupun sebuah usaha yang sangat menjanjikan para peternak ikan untuk mendapatkan keuntungan yang meningkat apabila hasil ternak berhasil dibudidayakan oleh peternak ikan. Berdasarkan pengamatan, ikan ternak pada umumnya adalah ikan air tawar, dimana mengkonsumsi ikan ini sangat dianjurkan bagi kesehatan manusia karena mengandung protein dan vitamin lainnya yang sangat berguna. Jika dilihat dari sisi bisnis, ikan air tawar termasuk ikan yang banyak dikonsumsi oleh manusia karena sumber gizinya dan kesehatan ikannya, sehingga volume kebutuhan pasar akan ikan air tawar ini memuncak apalagi pada hari-hari besar sehingga sulit didapatkan, berkaitan hal tersebut harganya pun makin melonjak dan para peternak hingga pebisnis ikan pun makin banyak untung.

Namun untuk mendapatkan kesuksesan tersebut tidak semuanya para peternak ataupun pebisnis meraihnya, dikarenakan banyak hal yang perlu diperhatikan oleh si peternak ikan terhadap kolam tempat ikan ditanamkan ataupun dibudidayakan. Berkaitan dengan masalah tersebutlah para peternak ikan sangat sulit melakukan hal tersebut akibat waktu dan energi digunakan untuk kegiatan lain yang terkadang lebih krusial pada saat itu.

1.2. Rumusan Masalah

Salah satu syarat yang sangat penting dilakukan dalam budidaya ikan dalam kolam adalah pada sisi jadwal pemberian pakan ikan. Sebab jadwal pemberian pakan ikan sangat berpengaruh terhadap cepat dan lambatnya pertumbuhan ikan dalam kolam. Selain itu pola pemberian pakanpun harus merata pada seluruh ikan. Terkait hal tersebutlah para peternak ikan merasa kesulitan untuk melakukannya karena kesibukan lain yang sedang dilakukan dan bahkan terkadang kesibukan tersebut tidak bisa ditinggalkan.

Oleh sebab itu timbul ide peneliti untuk membuat sebuah solusi dengan merancang sebuah alat cerdas yang dikendalikan oleh sebuah sistem minimum Arduino Uno.

1.3. Tujuan dan Manfaat

Tujuan penelitian ini adalah sebagai berikut:

- a. Membuat alat yang mampu memberi pakan ikan sesuai kendali yang dirancang.
- b. Memudahkan pemilik ikan dalam memberikan pakan ikan di kolam sesuai jadwal
- c. Membantu pemilik ikan menjaga kesehatan/pertumbuhan ikan peliharaan.
- d. Membuat pemerataan dalam pemberian pakan ikan dengan 3 (tiga) radius tembakan pakan.

Manfaat yang dapat dirasakan adalah sebagai berikut:

- a. Menjamin ikan mendapat pakan sesuai jadwal.
- b. Kesehatan dan pertumbuhan ikan menjadi lebih baik.
- c. Memberikan kemudahan kepada peternak ikan.

1.4. Batasan Masalah

Dalam perancangan dan pembuatan alat ini, terdapat beberapa hal sebagai batasan masalah dalam pembahasan, antara lain:

- a. Sistem pengendali yang digunakan adalah sistem minimum Arduino Uno
- b. Bahasa program yang digunakan adalah “C”
- c. Motor yang digunakan adalah motor *DC* (magnet permanen) dengan *gear box*.
- d. Sistem yang dirancang adalah ukuran skala kecil atau *prototype*.

1.5. Sistematika Pembahasan

Untuk memperoleh hasil yang maksimal dalam penyusunan Skripsi ini maka, peneliti membuat urutan pembahasan sebagai berikut:

BAB I PENDAHULUAN

Dalam bab ini akan dibahas latar belakang, rumusan masalah, tujuan dan manfaat, batasan masalah, dan sistematika pembahasan.

BAB II TEORI PENUNJANG

Dalam bab ini berisikan tentang penjelasan beberapa teori dasar *hardware* dan *software* yang digunakan dalam penyelesaian pembuatan sistem rangkaian alat cerdas “Pemberi Pakan Ikan dengan Sistem Tiga Radius Tembakan”.

BAB III METODOLOGI PENELITIAN

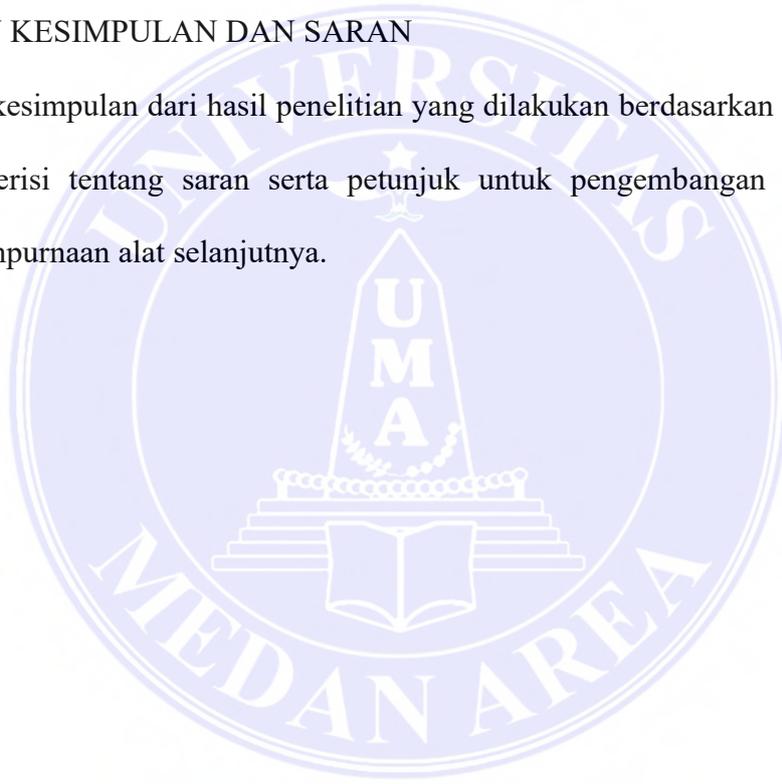
Bab ini membahas tentang metoda yang dilakukan dalam perancangan dan pembuatan alat secara *hardware* maupun *software*, serta pengujiannya secara detail.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini merupakan bagian yang menyajikan hasil penelitian yang dilakukan, serta menganalisa akurasi sistem yang telah dibuat.

BAB V KESIMPULAN DAN SARAN

Berisi kesimpulan dari hasil penelitian yang dilakukan berdasarkan data yang ada, juga berisi tentang saran serta petunjuk untuk pengembangan dalam rangka penyempurnaan alat selanjutnya.



BAB II TEORI PENUNJANG

2.1. Arduino

Arduino adalah sebuah perangkat prototype elektronik berbasis mikrokontroler yang fleksibel dan open-source, perangkat keras dan perangkat lunaknya mudah digunakan. Perangkat ini diperuntukkan bagi siapapun yang tertarik memanfaatkan mikrokontroler secara mudah dan efisien. Penggunaan Board ini bagi pemula akan memudahkan mempelajari pengendalian dengan mikrokontroler, bagi desainer pengontrol memudahkan dalam membuat prototype maupun implementasi, demikian juga bagi yang hobi mengembangkan mikrokontroler. Arduino dapat digunakan mendeteksi lingkungan dengan menerima masukan dari berbagai sensor (misalnya: suhu, inframerah, cahaya, tekanan, ultrasonik, jarak, kelembaban) dan dapat mengendalikan peralatan sekitarnya (misal: lampu, berbagai jenis motor, dan aktuator lainnya). (Andrianto, 2015)

Berikut ini adalah Gambar 2.1 yang menampilkan bentuk fisik Arduino Uno:



Gambar 2. 1 Arduino Uno

(Sumber: <http://store.linksprite.com/arduino-usb-board-uno-r3/>)

Adapun data teknis *board* Arduino Uno adalah sebagai berikut:

- a. Mikrokontroler : ATmega328
- b. Tegangan Operasi : 5V
- c. Tegangan Input (recommended) : 7 - 12 V
- d. Tegangan Input (limit) : 6-20 V
- e. Pin digital I/O : 14 (6 diantaranya pin PWM)
- f. Pin Analog input : 6
- g. Arus DC per pin I/O : 40 mA
- h. Arus DC untuk pin 3.3 V : 150 mA
- i. Flash Memory : 32 KB dengan 0.5 KB digunakan untuk bootloader
- j. SRAM : 2 KB
- k. EEPROM : 1 KB
- l. Kecepatan Pewaktuan : 16 Mhz

Papan *Arduino* merupakan elektronis yang berisi mikrokontroler berbasis *ATMega*, sedangkan mikrokontroler adalah sebuah keping (*integrated circuit*) yang mengandung prosesor alias sang pemroses dan sekaligus memori yang berguna untuk menyimpan data. (Kadir, 2016)

Berikut ini adalah Gambar 2.2 yang memperlihatkan contoh bentuk fisik sebuah mikrokontroler:



Gambar 2. 2 Mikrokontroler ATmega 328
(Sumber: <https://lapakrobotika.com/atmega-328A>)

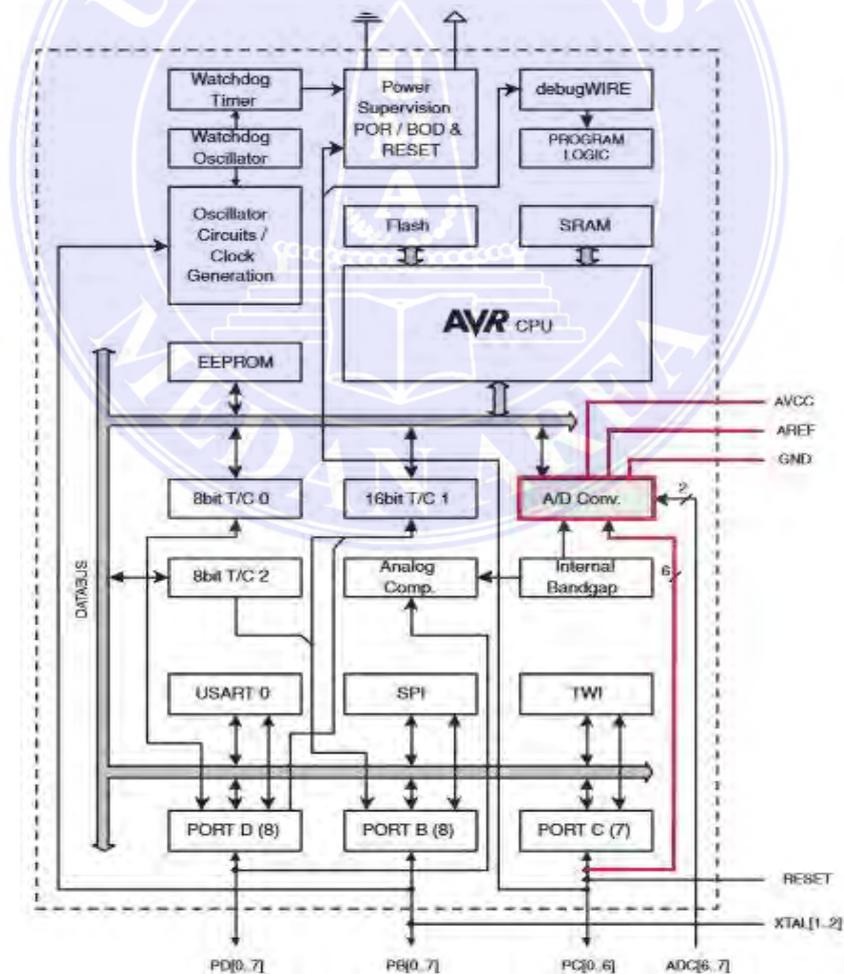
Kelebihan Arduino, antara lain:

- a. Sudah memiliki sarana komunikasi USB, sehingga pengguna laptop yang tidak memiliki port serial/RS323 bisa menggunakannya.
- b. Memiliki modul siap pakai (*shield*) yang bisa ditancapkan pada *board* arduino. Contohnya *shield* GPS, *Ethernet*, dan lain-lain.
- c. Tidak perlu perangkat *chip programmer* karena di dalamnya sudah ada *bootloader* yang akan menangani *upload* program dari komputer.

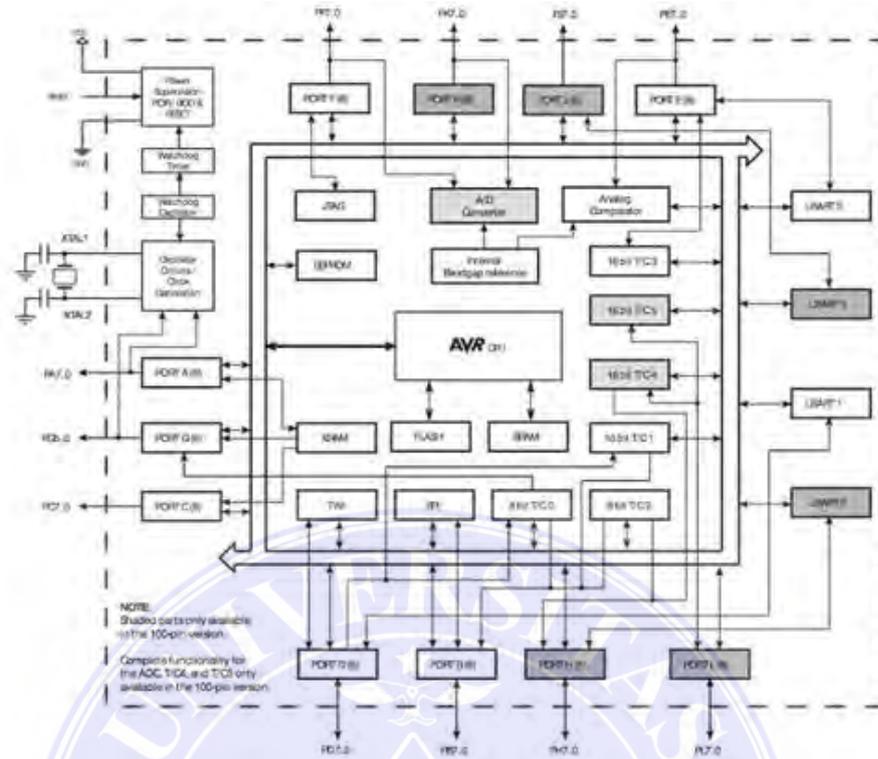
2.2. Arsitektur Mikrokontroler ATmega

Seluruh mikrokontroler yang diimplementasikan pada produk *Arduino* menggunakan *ATmega* keluarga *AVR*. Salah satunya seri *ATmega 328* (Gambar 2.3) dengan sejumlah fitur diantaranya *ON-Chip System Debug*, 5 mode tidur (*Sleep Mode*), 6 saluran *ADC* yang mendukung reduksi derau, mode hemat daya (*Power Save Mode, Power Down*), dan mode siaga (*Standby Mode*). (Istiyanto, 2014)

Mikrokontroler *ATMega 328* umumnya digunakan pada *board Arduino UNO*, *Nano*, dan lain-lain. Sedangkan *ATMega 1280* (Gambar 2.4) dipakai pada tipe *Arduino Mega*, baik *ATMega 328* maupun *ATMega 1280*, keduanya memakai kristal 16 MHz sebagai pembangkit clock. Keduanya juga memiliki blok memori *Flash* untuk menyimpan instruksi program, *SRAM* untuk menyimpan variabel data sementara, dan *EEPROM* sebagai media penyimpanan data yang tetap tersimpan saat mikrokontroler pada kondisi tidak dicatu. Fitur mikrokontroler AVR seri lainnya, seperti *ATMega 168* atau *2560* tidak jauh berbeda dengan 328 atau 1280, kecuali pada ukuran kapasitas blok memori *EEPROM*, *Flash*, dan *SRAM*. (Istiyanto, 2014)



Gambar 2. 3 Blok Diagram Mikrokontroler ATMega 328



Gambar 2. 4 Blok Diagram Mikrokontroler ATmega 1280

(Sumber:

<https://www.allaboutcircuits.com/technical-articles/understanding-arduino-uno-hardware-design/>
<http://readingrat.net/arduino-mega-2560-circuit-diagram/arduino-mega-2560-circuit-diagram-the-wiring-diagram/>)

2.3. Definisi Program

Menurut Kadir dan Heriyanto (2005:2) “suatu program ditulis dengan mengikuti kaidah bahasa pemrograman tertentu. Bahasa pemrograman dapat dianalogikan dengan bahasa yang digunakan manusia (bahasa manusia)”.

Sebagaimana diketahui, ada bermacam-macam bahasa manusia, seperti bahasa Rusia, Inggris, Indonesia, maupun bahasa daerah. Kumpulan instruksi dalam bahasa manusia yang berupa sejumlah kalimat dapat dianalogikan dengan suatu program. Manusia dapat mengerjakan suatu intruksi berdasarkan kalimat dan komputer dapat menjalankan suatu instruksi program.

Dalam pemrograman, terdapat beberapa bahasa pemrograman, seperti BASIC, C, Pascal, dan C++. Pada dasarnya bahasa pemrograman dikelompokkan menjadi:

- a. High Level Language
- b. Low Level Language

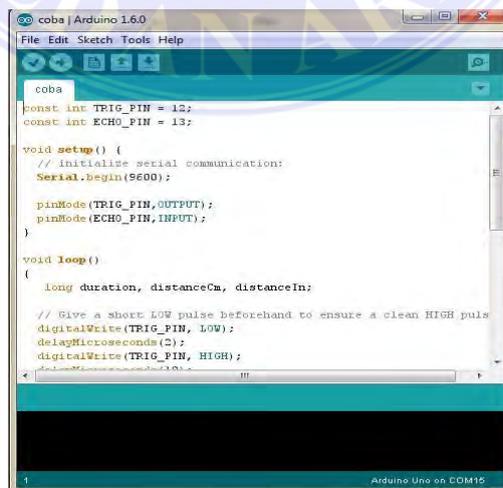
High Level Language adalah bahasa pemrograman yang berorientasi kepada bahasa manusia. Program dibuat dengan menggunakan bahasa pemrograman yang mudah dipahami oleh manusia, biasanya menggunakan kata-kata dalam bahasa Inggris; misalnya IF untuk menyatakan “jika” dan AND untuk menyatakan “dan”. Yang termasuk dalam kelompok bahasa ini adalah bahasa BASIC, C, Pascal, dan C++.

Low Level Language adalah bahasa pemrograman yang berorientasi kepada mesin. Bahasa ini menggunakan kode biner (kode 0 dan 1) atau suatu kode sederhana untuk menggantikan kode tertentu dalam sistem biner. Yang termasuk dalam kelompok bahasa ini adalah bahasa mesin dan bahasa rakitan. Bahasa-bahasa ini sulit dipahami oleh orang awam dan sangat membosankan bagi programmer yang biasa dengan High Level Language. Programmer harus menguasai operasi komputer secara teknis. Namun bahasa pemrograman ini memberikan eksekusi program yang sangat cepat. Selain itu bahasa mesin tergantung pada mesin (machine dependent); maksudnya bahasa satu mesin dengan mesin yang lainnya jauh berbeda.

2.4. Software Arduino IDE

IDE (Integrated Development Environment) adalah sebuah program spesial yang berjalan di komputer yang mengizinkan user menulis sketch untuk board Arduino dalam bentuk bahasa pemrograman yang menggunakan Bahasa Processing. Software Arduino ini dapat diinstal di berbagai OS (operating system) seperti: Windows, Mac OS, LINUX. Software Arduino IDE terdiri dari 3 bagian yaitu :

- a. Editor program, untuk mengedit dan menulis program dalam bahasa Processing. Listing program Arduino disebut sketch.
- b. Compiler, modul yang berfungsi untuk mengubah bahasa Processing ke dalam kode biner, karena kode biner merupakan bahasa program yang dipahami mikrokontroler.
- c. Uploader, modul yang berfungsi mengupload kode biner ke dalam memori mikrokontroler. (Feri, 2011). Gambar 2.5 di bawah ini adalah tampilan Arduino IDE:



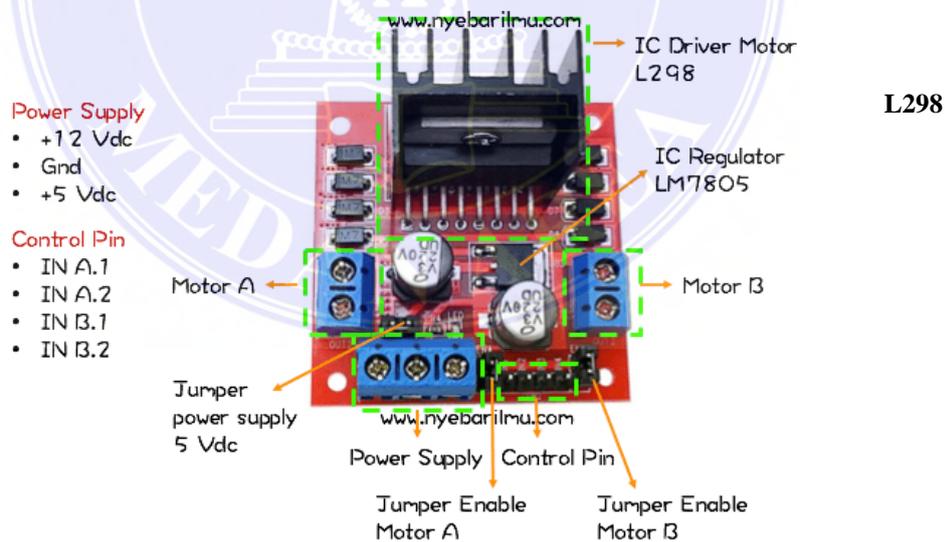
Gambar 2. 5 Tampilan Arduino IDE Versi 1.6.4

(Sumber: <https://learn.adafruit.com/add-boards-arduino-v164/overview>)

2.5. Driver L298

L298 adalah komponen elektronik yang berfungsi untuk mengontrol arah putaran motor DC. Sebuah L298 bisa digunakan untuk mengontrol dua motor DC. Selain untuk mengontrol arah putaran motor DC, L298 juga bisa digunakan sebagai driver motor Stepper Bipolar. IC driver L298 dapat menggerakkan motor DC sampai arus 2A dan tegangan maksimal 40 volt DC untuk satu kanalnya. Pin enable A dan B untuk mengatur kecepatan motor, pin input 1 sampai 4 digunakan untuk mengendalikan arah putaran. Pin output pada IC L298 13 dihubungkan ke motor DC yang sebelumnya melalui dioda yang disusun secara H-bridge. Kecepatan motor diatur menggunakan teknik PWM (Pulse Width Modulation). (Adriansyah, 2013)

Berikut ini adalah Gambar 2.6 yang menampilkan bentuk fisik dari Driver L298:



Gambar 2. 6 Modul Driver L298

(Sumber: <https://www.nyebarilmu.com/tutorial-arduino-mengakses-driver-motor-l298n/>)

Keterangan :

- a. Enable A : untuk mengaktifkan bagian output motor A
- b. Enable B : untuk mengaktifkan bagian output motor B
- c. Jumper 5 vdc : mode pemilihan sumber tegangan 5 Vdc, jika tidak di jumper maka akan ke mode sumber tegangan 12 Vdc
- d. Control Pin : kendali perputaran dan kecepatan motor yang dihubungkan ke Mikrokontroler

Jika kita perhatikan pada Gambar 2.5 di atas maka dapat dijelaskan spesifikasi dari modul driver L298 sebagai berikut:

- a. Menggunakan L298 Dual Full Bridge.
- b. Tegangan suplai operasi maksimal 46V.
- c. Total arus DC sampai dengan 4A.
- d. Terdiri dari 2 bagian yang independen.
- e. Memiliki 4 output.
- f. Dapat dihubungkan langsung pada mikrokontroler Arduino, ARM, AVR, MCS-51, dan sebagainya.
- g. Dimensi : 4,3 cm x 4,3 cm x 2,6 cm.
- h. Berat : 27 gr

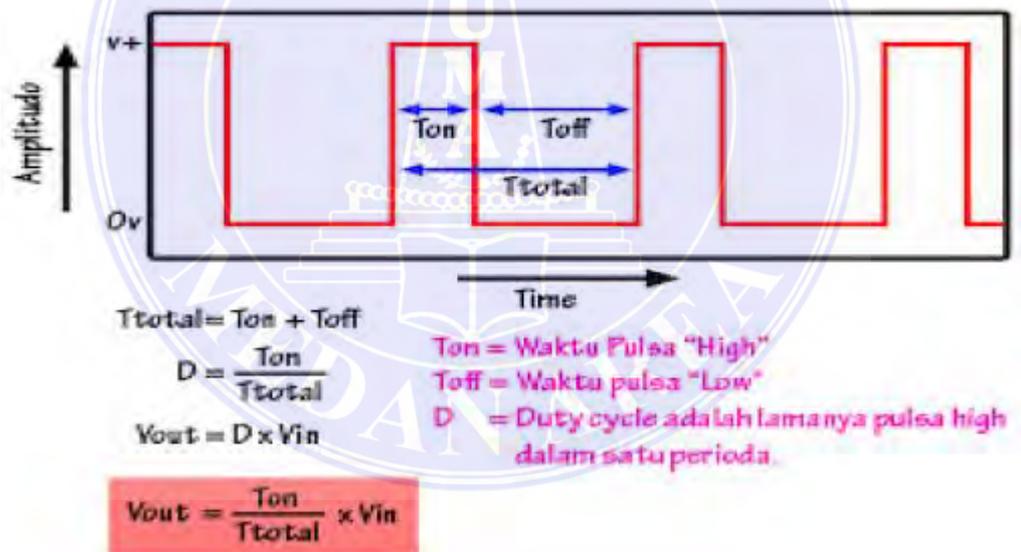
Kelebihan L298N motor driver:

- a. Lebih presisi dalam mengontrol motor
- b. Dapat mengendalikan motor yang besar (maksimal 2A)
- c. Dilengkapi dengan heatsink sehingga lebih tahan panas
- d. Pemasangan mudah

Pemasangan driver L298N membutuhkan 6 buah pin mikrokontroler. 2 pin Enable (satu untuk tiap motor DC). 4 pin untuk mengatur kecepatan motor DC. Pada dasarnya rangkaian driver motor L298N dapat mengatur arus dan tegangan untuk mengatur kecepatan dan arah motor.

2.6. Motor DC

Motor DC dapat berputar searah jarum jam (CW) maupun berlawanan arah jarum jam (CCW). Selain itu kecepatan putarannya dapat diatur menggunakan PWM. (Andrianto, 2015). Berikut ini Gambar 2.7 yang menampilkan skema sistem PWM dan Gambar 2.8 yang menampilkan bentuk fisik dari motor DC :



Gambar 2. 7 Skema Sistem Kerja PWM
(Sumber: Google.com)



Gambar 2. 8 Bentuk Fisik dan Spesifikasi Motor DC Gearbox
(Sumber: Google.com)

Motor DC Gearbox 6V 200 Rpm
Tegangan : 3v - 6v
Rpm : 100-200
Dimensi motor : 2.76 in x 1.46 in x 0.87 in (7 cm x 3.7 cm x 2.2 cm)

Strong magnetic with anti-interference
Double axis gear motor
Reduction ratio: 1:48
Working voltage: 3V ~ 6V
Unloads current: 200mA @ 6V- 150mA @ 3V
Unloads speed: 200 10%RPM @ 6V- 90 10%RPM @ 3V

Untuk memutar roda ada dua cara, yaitu :

1. Motor dihubungkan langsung dengan roda
2. Motor dan roda menggunakan *gear*

Tujuan penggunaan gear yaitu menghasilkan kecepatan dan power (torsi yang besar). Berdasarkan tujuan tersebut maka ada dua konfigurasi pemasangan *gear*.

- a. Kecepatan (Motor Gear Besar; Roda Gear Kecil)

Motor DC menggunakan gear yang lebih besar daripada gear yang digunakan pada roda, sehingga satu kali putaran gear motor DC akan menghasilkan beberapa kali putaran gear roda, sehingga menghasilkan kecepatan yang tinggi.

b. Power (Kecil ke Besar)

Motor DC menggunakan *gear* yang lebih kecil daripada gear yang digunakan pada roda, sehingga menghasilkan putaran roda yang memiliki kecepatan yang rendah, namun memiliki power (torsi) yang besar.

2.7. Catu Daya

Pin catu daya adalah pin yang memberikan tegangan untuk komponen atau rangkaian yang dihubungkan dengan arduino. Pada bagian catu daya ini terdapat pin V input dan Reset. V input digunakan untuk memberikan tegangan langsung ke arduino tanpa melalui tegangan pada USB atau adaptor, sedangkan Reset adalah pin untuk memberikan sinyal reset melalui tombol atau rangkaian eksternal. (Syukur, 2017)

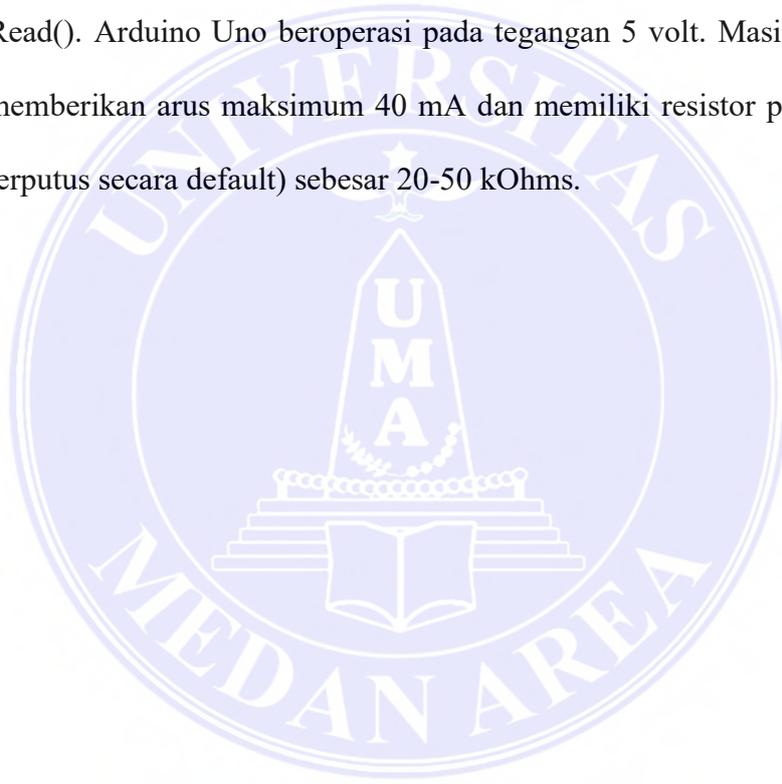
Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya eksternal (non - USB) berasal dari adaptor AC-DC atau baterai. Adaptor dapat dihubungkan dengan menghubungkan steker 2,1 mm ke jack sumber tegangan pada papan. Jika tegangan berasal dari baterai dapat dihubungkan langsung melalui header pin Gnd dan pin Vin dari konektor power.

Pin tegangan yang tersedia pada papan Arduino antara lain:

- a. Vin : Input tegangan papan Arduino ketika menggunakan sumber daya eksternal.
- b. 5V : Pin yang mengeluarkan tegangan 5 Volt, dari pin ini tegangan sudah diatur dari regulator yang tersedia (buitin) pada papan. Arduino dapat diaktifkan dari jack power DC (7-12 V), konektor USB (5V), atau pin Vin pada board (7-12 V).

- c. 3V3 : Pin yang mengeluarkan tegangan 3,3 Volt. Arus maksimum yang dihasilkan adalah 50 mA.
- d. GND : Pin Ground atau Massa.
- e. IOREF : Pin ini berfungsi memberikan referensi tegangan yang beroperasi pada mikrokontroler.

Masing-masing dari jumlah digital pin pada Arduino Uno dapat digunakan sebagai input atau output, menggunakan fungsi `pinMode()` , `digitalWrite()` , dan `digitalRead()`. Arduino Uno beroperasi pada tegangan 5 volt. Masing-masing pin dapat memberikan arus maksimum 40 mA dan memiliki resistor pull-up internal (yang terputus secara default) sebesar 20-50 kOhms.



2.8. LCD 2x16

LCD merupakan perangkat penampil yang banyak digunakan. Penampil LCD (Gambar 2.9) menggantikan penampil CRT (Cathode Ray Tube), yang sebelumnya digunakan manusia sebagai penampil gambar/text baik monokrom (hitam dan putih), maupun berwarna. LCD memberikan keuntungan lebih dibandingkan CRT, karena CRT adalah tabung triode yang digunakan sebelum transistor ditemukan. (Syukur, 2017)

Keuntungan LCD dibandingkan dengan CRT adalah konsumsi daya yang relatif kecil, lebih ringan, dan tampilan yang bagus.



Gambar 2. 9 Bentuk Fisik LCD
(Sumber: Google.com)

LCD memanfaatkan silicon atau gallium dalam bentuk Kristal cair sebagai pemender cahaya. Keunggulan LCD hanya menarik arus kecil (beberapa mikro ampere), sehingga alat atau sistem menjadi portable karena dapat menggunakan catu daya yang kecil. Selain itu keunggulan LCD dapat dibaca dengan mudah di bawah sinar matahari. Di bawah sinar matahari yang redup atau dalam kondisi gelap, sebuah lampu (berupa led) harus dipasang dibelakang layar tampilan. (Izar, 2012).

Pada dasarnya operasi LCD terdiri dari empat, yaitu instruksi mengakses proses internal, instruksi menulis data, instruksi membaca kondisi sibuk, dan instruksi membaca data yang dapat dilihat pada tabel 2.1 berikut.

Tabel 2. 1 Operasi Dasar LCD

RS	R/W	Operasi
0	0	Input Interuksi ke LCD
0	1	Membaca status flag (DB7) dan alamat counter (DB0 ke DB6)
1	0	Menulis Data
1	1	Membaca Data

2.9. Keypad 3x4 Matrix

Keypad merupakan salah satu bagian penting dari suatu perangkat elektronik yang butuh interaksi manusia. Keypad berfungsi sebagai interface (lantar muka) antara perangkat (mesin) elektronik dengan manusia atau dikenal dengan istilah HMI (Human Machine Interface).

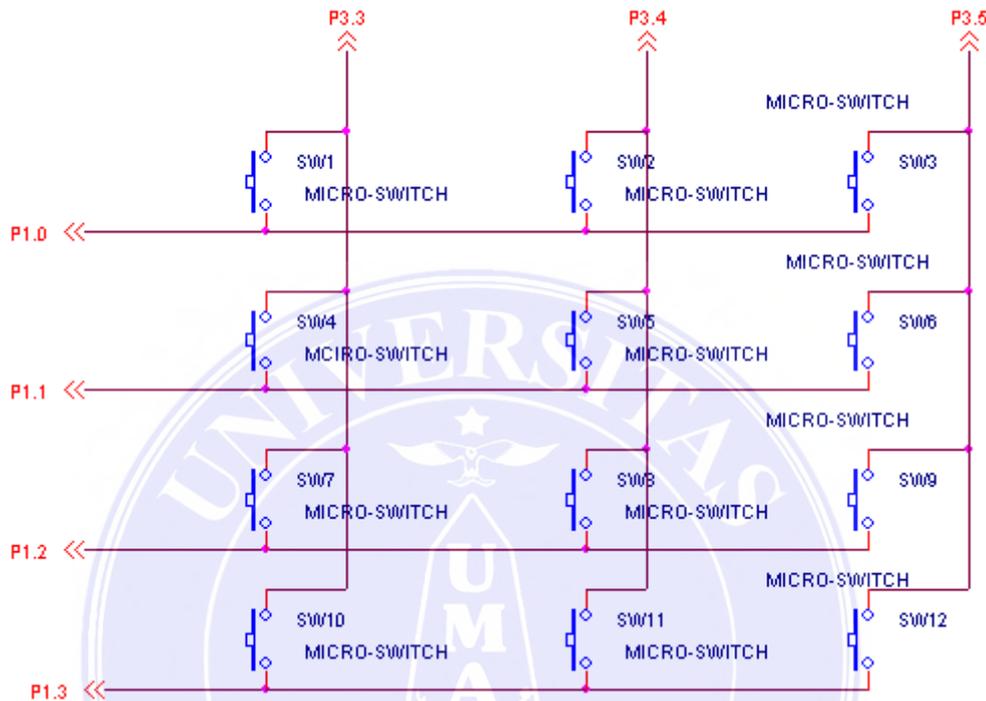
Keypad Matriks adalah tombol-tombol yang disusun secara maktriks (baris x kolom) sehingga dapat mengurangi penggunaan pin input. Sebagai contoh, Keypad Matriks 3×4 (Gambar 2.10) cukup menggunakan 7 pin untuk 12 tombol. Hal tersebut memungkinkan karena tombol-tombol disusun secara horizontal membentuk baris dan secara vertikal membentuk kolom.



Gambar 2. 10 Keypad 3x4 Matrix
(Sumber : google.com)

Matrix keypad 3×4 memiliki konstruksi atau susunan yang simple dalam penggunaan port mikrokontroler. Konfigurasi keypad dengan bentuk matrix ini bertujuan untuk menghemat port mikrokontroler karena jumlah tombol yang

dibutuhkan banyak pada suatu sistem mikrokontroler. Konstruksi matrix keypad 3×4 untuk mikrokontroler dapat dibuat seperti pada Gambar 2.11 berikut:



Gambar 2. 11 Rangkaian Dasar Keypad 3x4
(Sumber: google.com)

Proses scanning untuk membaca penekanan tombol pada matrix keypad 3×4 di mikrokontroler dilakukan secara bertahap kolom demi kolom. Program untuk scanning matrix keypad 3×4 dapat bermacam-macam. Misalkan keypad aktif LOW (semua line kolom dan baris dipasang resistor pull-up) dan dihubungkan ke port mikrokontroler dengan jalur kolom adalah jalur input dan jalur baris adalah jalur output.

Mengirimkan logika Low untuk kolom 2 dan logika HIGH untuk kolom yang lain kemudian membaca data baris, misal tombol 2 ditekan maka data baris pertama Row 1 akan LOW sehingga data baris yang dibaca adalah 0111, atau tombol yang ditekan tombol 5 maka data pada baris ke 2 (Row 2) akan LOW

sehingga data yang terbaca 1011, atau tombol 8 yang ditekan sehingga data yang terbaca 1101, atau tombol 0 yang ditekan maka data yang dibaca adalah 1110 dan atau tidak ada tombol pada kolom pertama yang di tekan maka data pembacaan baris akan 1111. (Febri, 2015)

2.10. Soket *USB*

Soket USB (Gambar 2.12) adalah soket kabel USB yang disambungkan ke laptop atau komputer, yang fungsinya untuk mengirim program ke arduino dan juga sebagai port komunikasi serial. Input/Output Digital atau digital pin adalah pin-pin untuk menghubungkan arduino dengan komponen atau rangkaian digital, contohnya, jika ingin membuat LED berkedip, LED tersebut bisa dipasang pada salah satu pin input atau output digital dan ground komponen lain yang menghasilkan output digital atau menerima input digital bisa disambungkan ke pin ini. (Syukur, 2017).



Gambar 2. 12 Bentuk Fisik Soket USB
(Sumber: Google.com)

2.11. Rangkaian *Tiny RTC (Real Time Clock)*

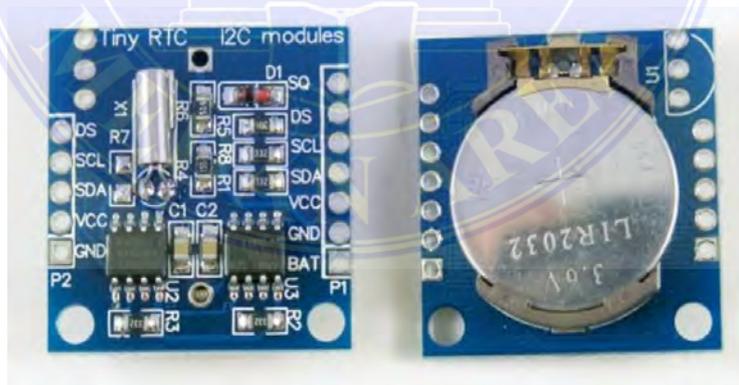
RTC adalah jenis penunjuk waktu yang bekerja berdasarkan waktu yang sebenarnya atau dengan kata lain berdasarkan waktu yang ada pada jam. Dalam prosesnya, penunjuk waktu ini memerlukan dua parameter utama yang

harus ditentukan, yaitu saat start (mulai) dan saat stop (berhenti). RTC pada dasarnya berupa sirkuit yang fungsinya untuk menjaga waktu tetap pada kondisi yang sebenarnya.

Rangkaian ini berfungsi sebagai penunjuk waktu dengan fitur-fitur sebagai berikut :

- a. Menghitung waktu dari detik, menit, jam, hari, tanggal, bulan dan tahun, dengan kompensasi tahun kabisat sampai tahun 2100.
- b. Memory / RAM sebesar 32 byte.
- c. Akses single byte atau burst.
- d. Support baterai Litium atau Ni-Cd untuk backup suplai.
- e. Kemampuan Trickle Charge untuk pengisian baterai jenis Ni-Cd.

Modul ini telah dilengkapi baterai backup jenis Litium untuk menjaga agar data pada DS1307 tetap update. Berikut ini adalah Gambar 2.13 yang menampilkan bentuk fisik dari *RTC* :



Gambar 2. 13 Bentuk Fisik Tiny RTC (Real Time Clock)
(Sumber: Google.com)

BAB III METODOLOGI PENELITIAN

3.1. Tempat dan Waktu Penelitian

3.1.1. Tempat Penelitian

Pembuatan dan pengujian alat cerdas pemberi pakan ikan dengan sistem tiga radius tembakan berbasis arduino uno ini dilakukan di :

- a. Nama Tempat : Laboratorium Sistem Digital dan PLC
Universitas Medan Area
- b. Alamat : Jalan Kolam No. 1 Medan Estate

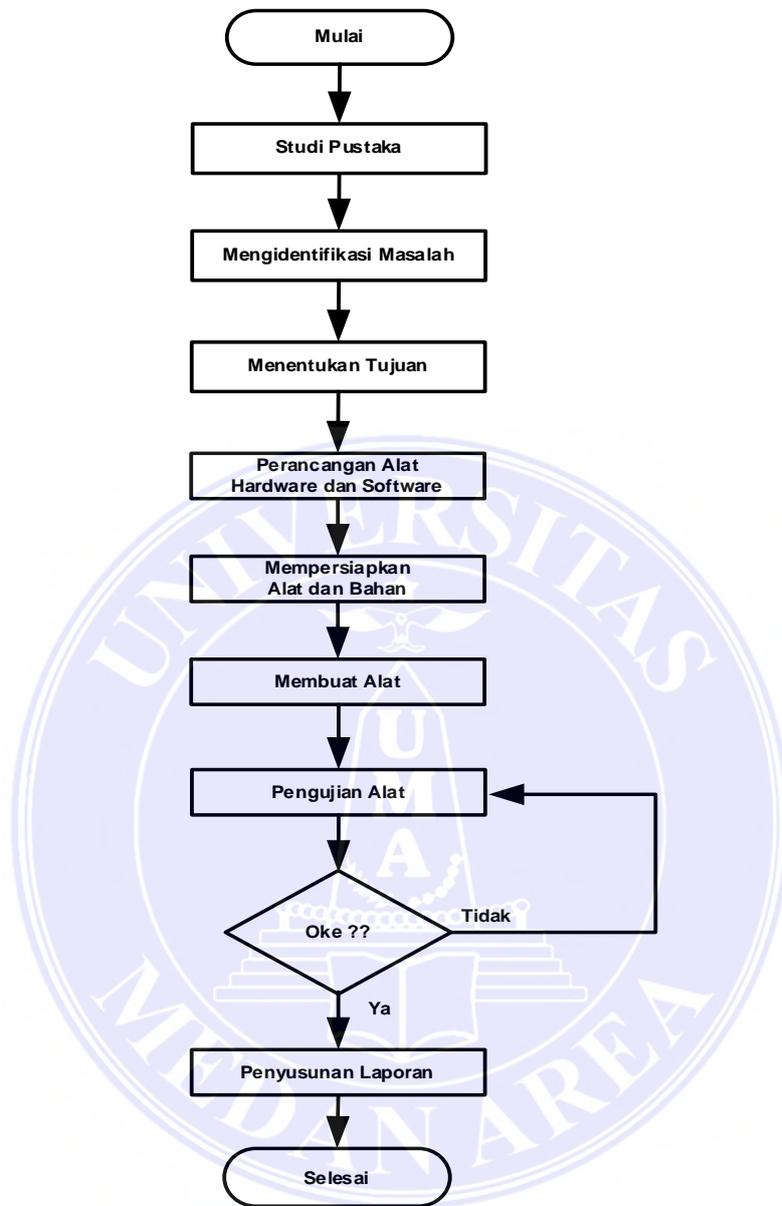
3.1.2. Waktu Penelitian

Pembuatan dan pengujian sistem ini membutuhkan waktu dengan rincian sebagai berikut :

- a. Penyediaan bahan dan alat : 1 minggu
- b. Perancangan dan pembuatan seluruh sistem : 1 bulan
- c. Pengujian sistem : 1 minggu
- d. Penyusunan laporan Skripsi : 1 minggu

3.2. Metoda Penelitian

Pelaksanaan penelitian ini dilakukan dalam beberapa tahap untuk mempermudah dan memperjelas arah penelitian. Berikut ini adalah Gambar 3.1, yaitu *flowchart* kerangka berfikir dalam penelitian, dimana berdasarkan *flowchart* inilah sebagai tahapan-tahapan yang dilakukan oleh peneliti dalam melakukan proses penelitian rancang bangun alat cerdas pemberi pakan ikan dengan sistem tiga radius tembakan berbasis Arduino Uno:



Gambar 3. 1 Flowchart Kerangka Berfikir
(Sumber : dokumentasi penulis)

3.3. Alat dan Bahan

Alat-alat yang digunakan pada pembuatan sistem ini adalah : 1 set *tools mechanic*, gerinda, bor listrik, solder listrik, mistar, alat tulis. Alat-alat yang digunakan pada uji kinerja alat rancangan antara lain : multimeter, dan testpen.

Bahan elektrik maupun mekanik yang digunakan dalam pembuatan sistem ini secara umum adalah seperti pada Tabel 3.1 berikut :

Tabel 3. 1 Penetapan Komponen (Bahan)

No.	Komponen	No.	Komponen
1	Modul Arduino Uno	10	Kabel pelangi
2	Motor dc gear box	11	Modul driver L298
3	Aqua bekas	12	Motor Servo
4	Acrelic	13	Modul RTC
5	IC LM7805	14	Sekrup
6	PCB bolong	15	LCD 2x16
7	Pilox warna biru	16	Papan triplek
8	AC-DC Adaptor	17	Spicer plastik
9	Keypad 4x4 matrix	18	Kabel USB Downloader

3.4. Rancangan Sistem Elektrikal

Rancangan sistem elektrikal yang dimaksud adalah meliputi tujuh rancangan yaitu:

- a. Sistem AC-DC adaptor
- b. Sistem minimum Arduino Uno
- c. Sistem driver L298 dengan aktuator
- d. Sistem Arduino dengan Keypad 4x4 Matrix
- e. Sistem Arduino dengan Motor Servo
- f. Sistem Arduino dengan LCD 2x16
- g. Sistem secara keseluruhan

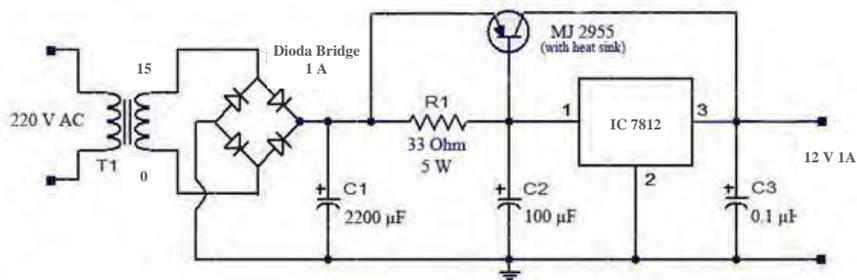
3.4.1. Sistem AC-DC Adaptor

AC-DC adaptor yang dirancang adalah cukup menggunakan AC-DC adaptor yang sudah jadi dan lebih simpel yaitu dengan membelinya di toko-toko penjual komponen elektronika, namun hal yang harus diperhatikan adalah spesifikasinya harus sesuai dengan kebutuhan sistem yang dirancang secara keseluruhan. Berikut adalah Gambar 3.2 yang memperlihatkan bentuk fisik AC-DC adaptor yang digunakan beserta spesifikasinya yang tertera di badan adaptor :



Gambar 3.2 AC-DC Adaptor
(Sumber : www.google.com)

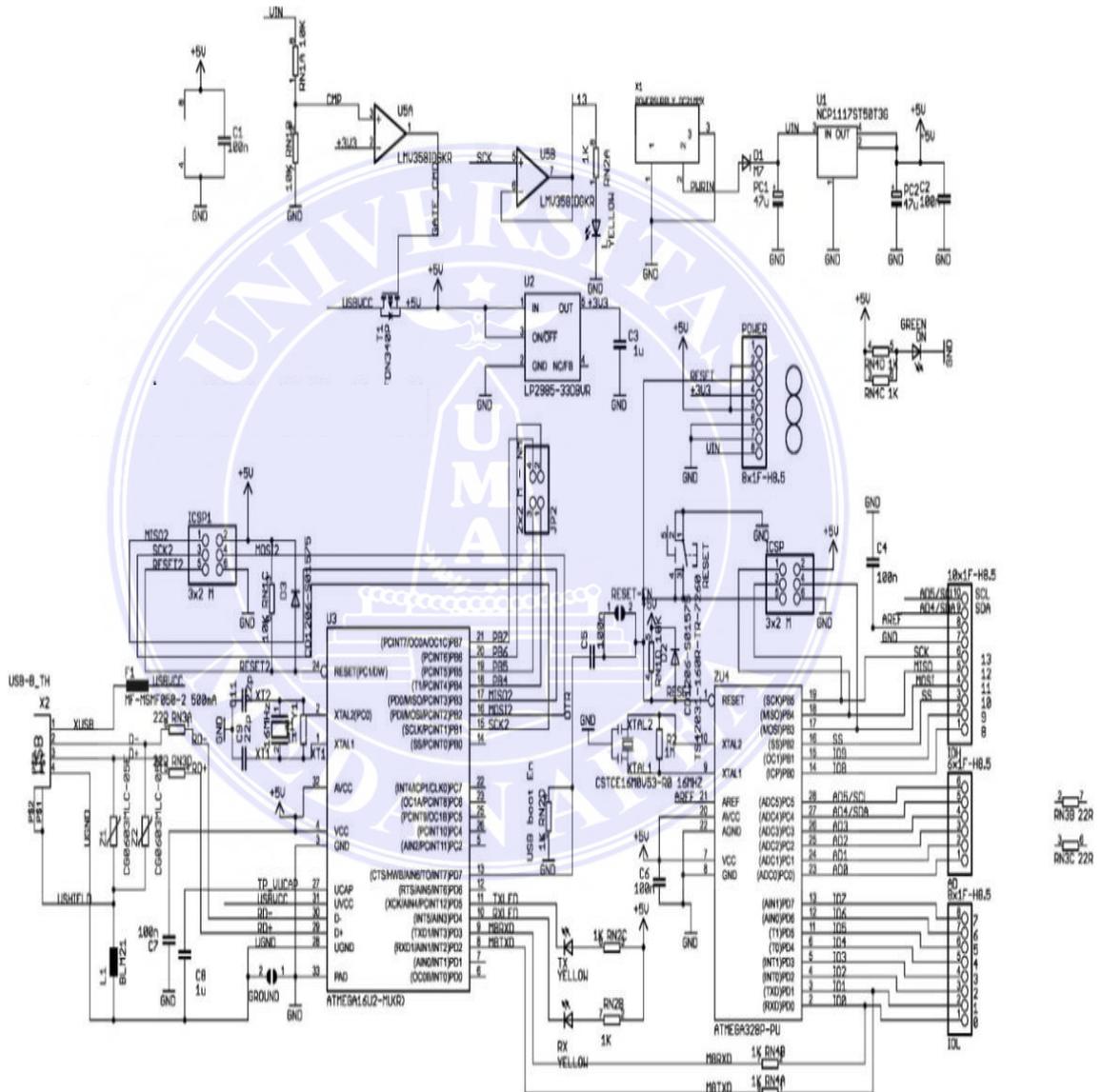
Pada Gambar 3.2 di atas yang memperlihatkan bentuk fisik AC-DC adaptor dapat juga dilihat skema rangkaian didalamnya seperti Gambar 3.3 di bawah ini :



Gambar 3.3 Skema Rangkaian AC-DC Adaptor
(Sumber : www.google.com)

3.4.2. Sistem Minimum Arduino Uno

Sistem minimum arduino Gambar 3.4 di bawah ini adalah gambar yang menunjukkan skema rangkaian dari sistem minimum Arduino Uno beserta mikrokontroler ATmega 328 :



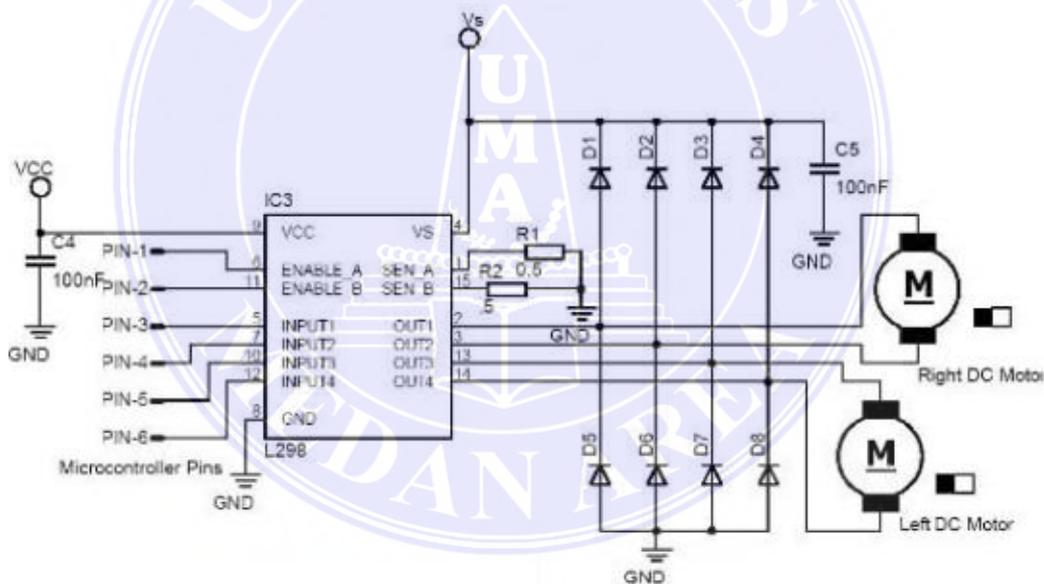
Gambar 3. 4 Sistem Minimum Arduino Uno
(Sumber : www.google.com)

3.4.3. Sistem Driver L298 dengan Aktuator

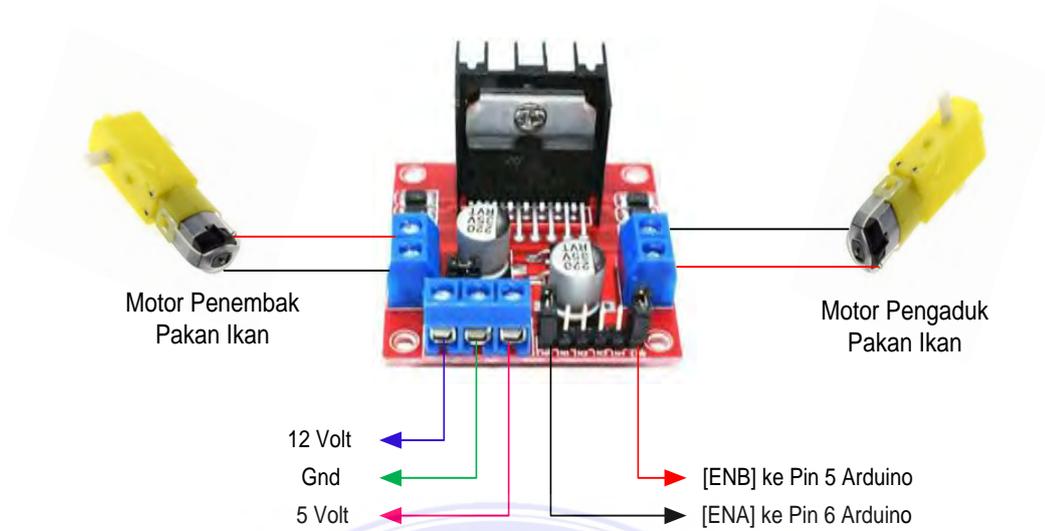
Sistem ini dirancang untuk mengendalikan arah putaran ataupun kecepatan motor DC gear box yang dalam penelitian ini ada dua buah, yaitu yang bertindak sebagai aktuator antara lain :

- a. Aktuator penembak pakan ikan
- b. Aktuator pengaduk pakan ikan

Sistem ini dirancang untuk mampu mengeluarkan output tegangan dan mengatur *PWM (Pulse Width Modulation)* kepada motor DC gear box hingga motor dapat bergerak dengan putaran kecepatan lambat, sedang dan cepat. Berikut adalah Gambar 3.5 yang memperlihatkan rangkaian instalasinya:



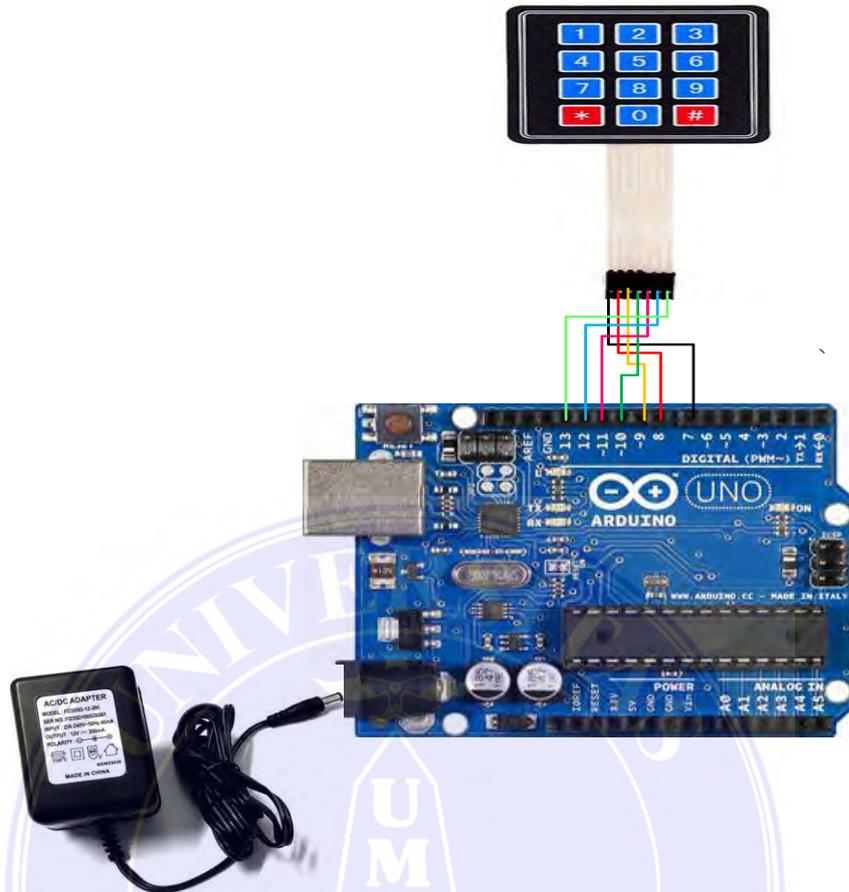
Gambar 3.5 Instalasi Driver L298 dengan Dua Motor DC Gearbox
(Sumber : www.google.com)



Gambar 3. 6 Instalasi Driver L298 dengan Dua Motor DC Gearbox
(Sumber : www.google.com)

3.4.4. Sistem Arduino dengan Keypad 4x4 Matrix

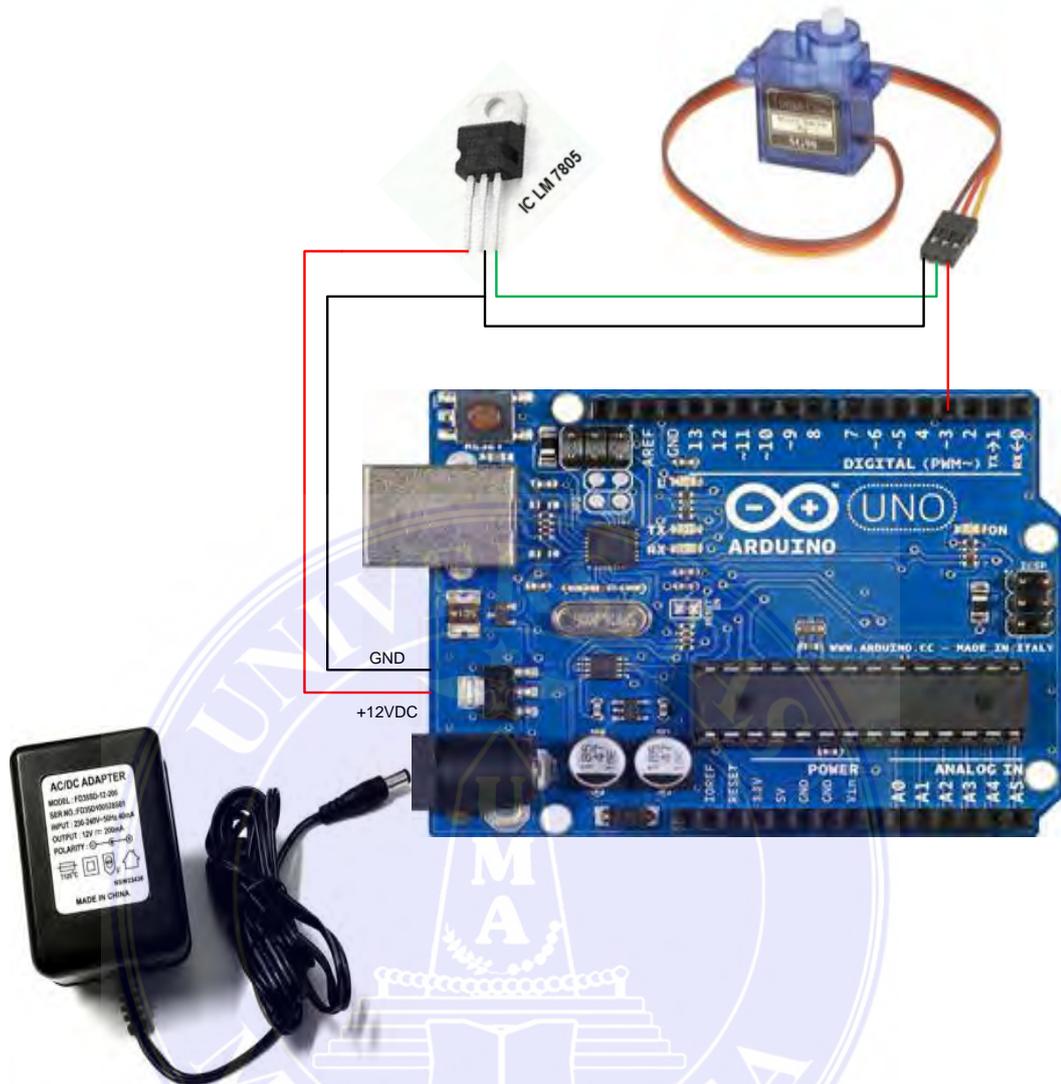
Keypad dalam hal ini berfungsi sebagai *interface* antara perangkat (mesin) elektronik dengan manusia atau dikenal dengan istilah HMI (*Human Machine Interface*). Dalam penelitian ini *keypad* ini digunakan sebagai *interface* untuk memasukkan sejumlah perintah atau data yang diinginkan oleh pengguna sehingga data tersebut akan diolah oleh perangkat pengendali (arduino uno) untuk melakukan proses berikutnya sesuai dengan program yang disetting. Berikut ini adalah Gambar 3.7 yang menampilkan rangkaian instalasi dari *Arduino* dengan *keypad*:



Gambar 3. 7 Instalasi Arduino dengan Keypad
(Sumber : dokumentasi penulis)

3.4.5. Sistem Arduino dengan Motor Servo

Rangkaian ini berfungsi sebagai pengendali motor servo yang digunakan sebagai motor penggerak untuk buka dan tutup katup dari tabung wadah pakan ikan. Adapun rangkaian instalasinya dengan Arduino uno adalah dapat dilihat seperti Gambar 3.8 di bawah ini:



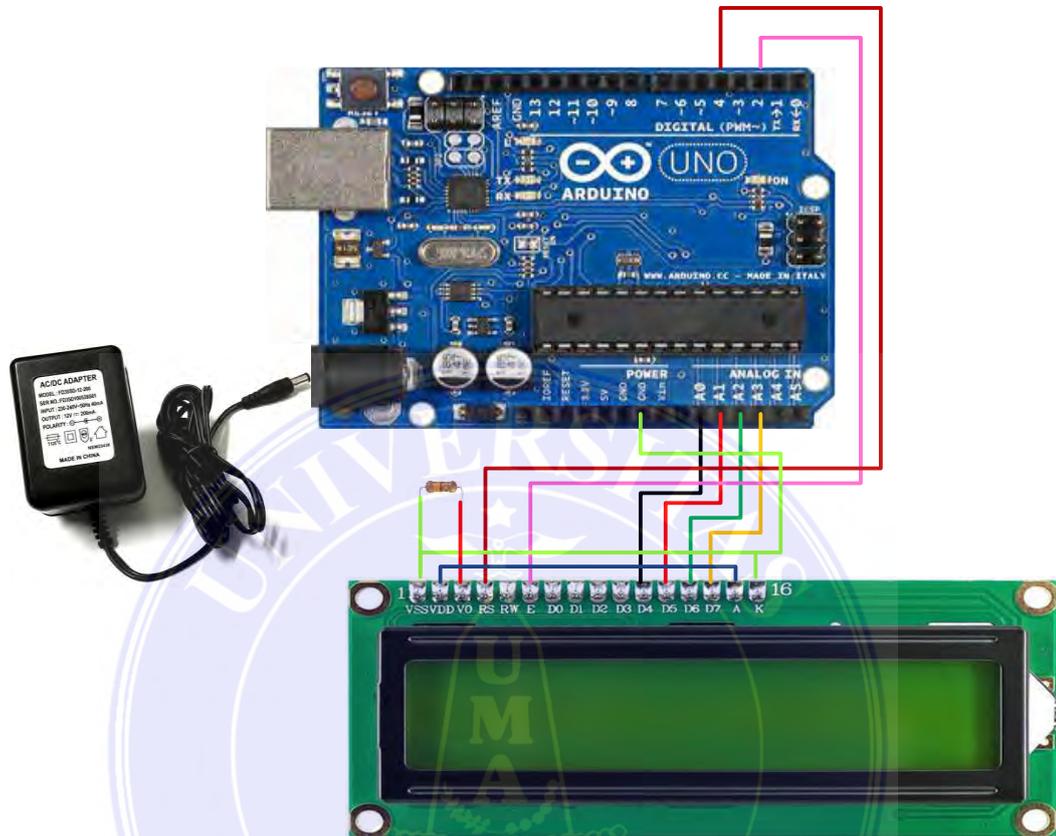
Gambar 3. 8 Rangkaian Instalasi Arduino dengan Motor Servo
(Sumber : dokumentasi penulis)

3.4.6. Sistem Arduino dengan Penampil Data (LCD 2x16)

Fungsi dari sistem ini adalah sebagai penampil data berupa tulisan atau kalimat-kalimat yang tampak sesuai settingan dalam program dan bukanlah yang tampak dalam bentuk sinyal tegangan atau arus namun telah dirubah olehnya data tegangan menjadi suatu tulisan yang tertampil dilayarnya. Sedangkan tulisan yang tampil tergantung dari tulisan yang dibuat pada program yang dibuat.

Dalam penelitian ini perlu diketahui bagaimana pola penginstalasian *LCD* 2x16 terhadap pin Arduino Uno agar dapat bekerja sebagai fungsi *LCD* sebagai

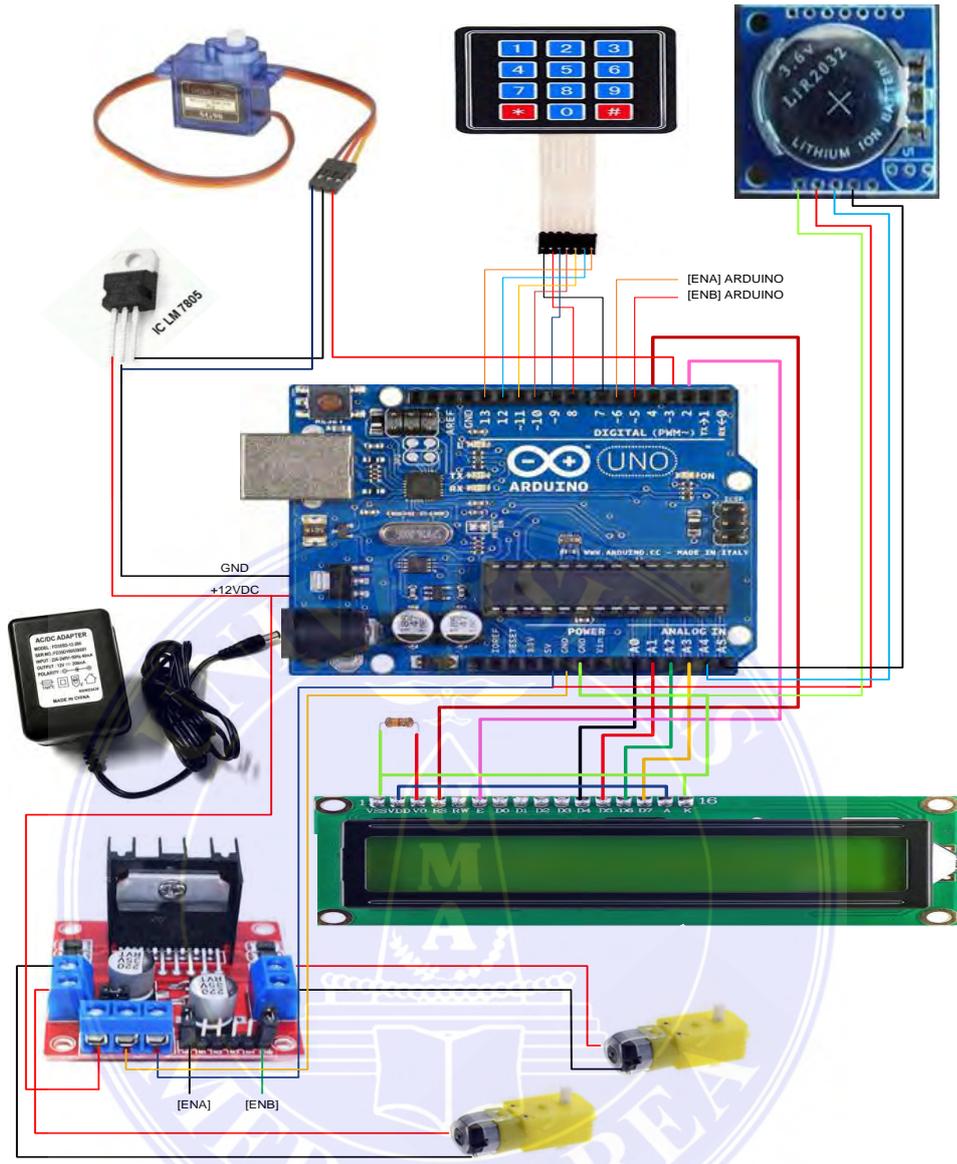
penampil data. Berikut adalah Gambar 3.9 yaitu pola penginstalasian *LCD 2x16* terhadap Arduino Uno :



Gambar 3. 9 Pola Instalasi LCD 2x16 pada Arduino Uno
(Sumber : dokumentasi penulis)

3.4.7. Sistem secara Keseluruhan

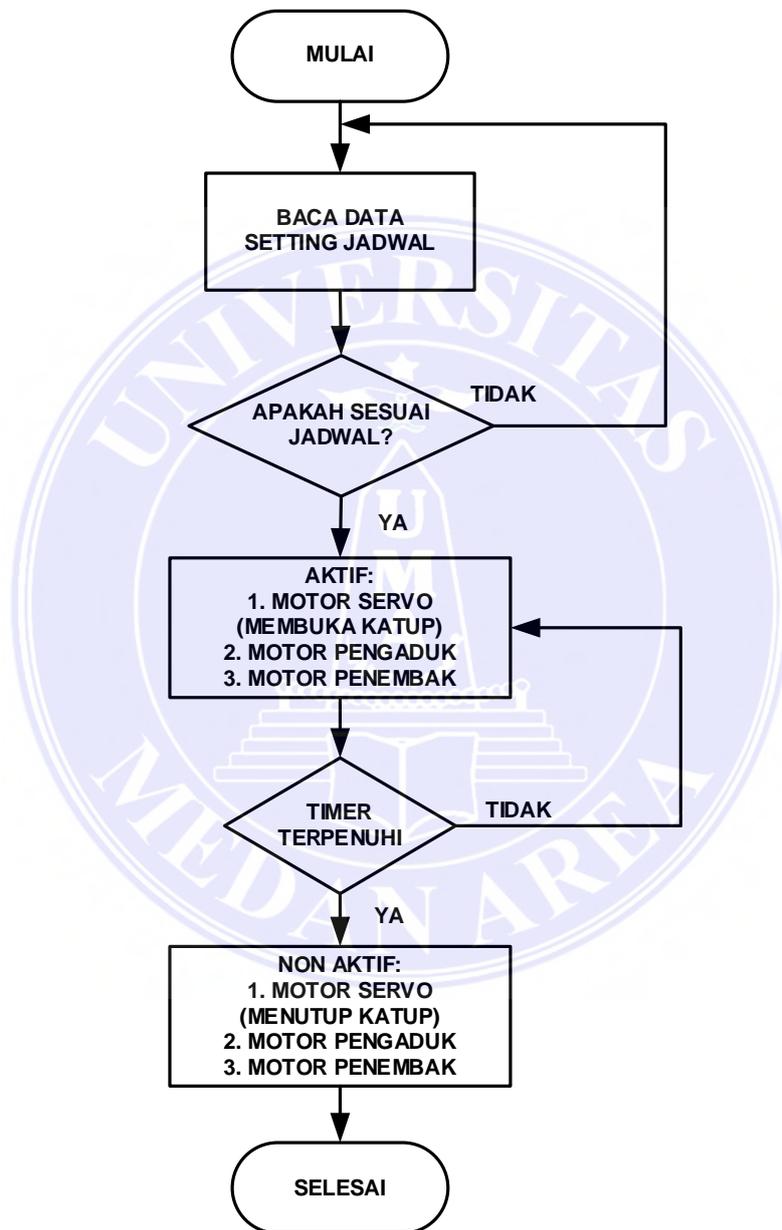
Dalam perancangan dan pembuatan sistem secara keseluruhan berarti seluruh komponen pembentuk sistem pemantauan ketinggian cairan infus akan dilakukan penggabungan seluruhnya baik dari segi mekanik maupun instalasi listriknya. Berikut Gambar 3.10 yang menampilkan skema rangkaian seluruh sistem.



Gambar 3. 10 Skema Rangkaian Seluruh Sistem
(Sumber : dokumentasi penulis)

3.5. Flowchart Sistem Kerja Alat

Berikut adalah Gambar 3.11 yang memperlihatkan alur kerja alat cerdas pemberi pakan ikan dengan sistem tiga radius tembakan berbasis arduino uno:



Gambar 3. 11 Flowchart Sistem Kerja Alat
(Sumber : dokumentasi penulis)

BAB V KESIMPULAN DAN SARAN

5.1. Kesimpulan

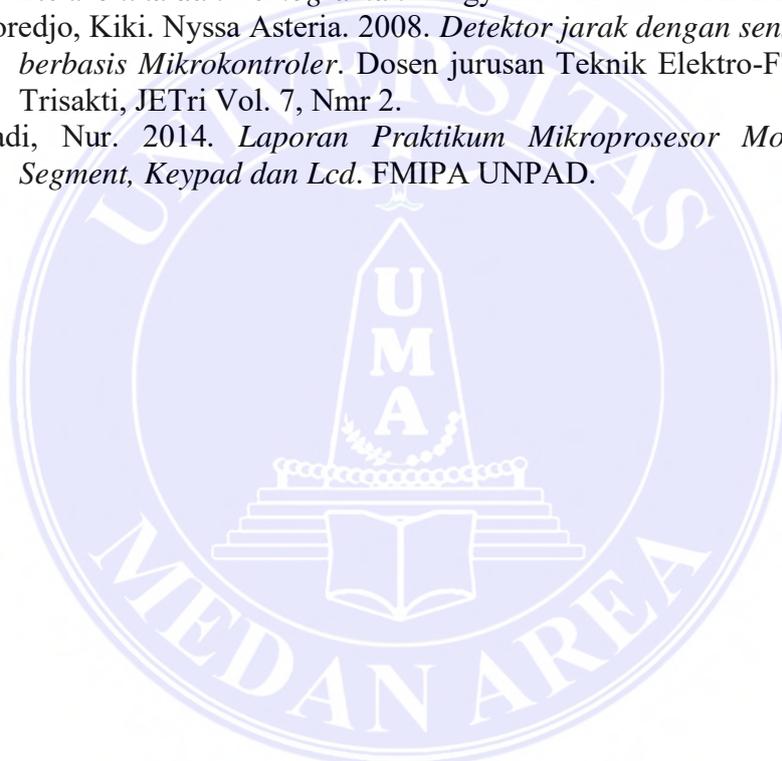
- a. Alat pemberi pakan ikan secara otomatis dapat meringankan pekerjaan peternak ikan dalam hal pemberian pakan secara teratur sehingga diharapkan produksi lebih meningkat.
- b. Alat pemberi pakan ikan ini dapat diatur waktu pemberian pakan dan jumlah/volume pakan secara teratur dan terjadwal sehingga dapat menjaga kesehatan ikan.
- c. Alat pemberi pakan ikan ini dapat memberikan/menyebarkan pakan dengan 3 radius tembakan sehingga pemberian pakan merata ke seluruh kolam.

5.2. Saran

- a. Mengoptimasi motor yang digunakan untuk pemutaran yang lebih kuat agar pakan dapat menyebar lebih luas.
- b. Dibuat alat pelindung dari air hujan agar alat tidak mudah rusak.

DAFTAR PUSTAKA

- Andrianto, Heri. 2015. *Arduino Belajar Cepat dan Pemrograman*. Bandung :INFORMATIKA
- Budiharto,Widodo dan Firmansyah, Sigit. 2005. *Elektronika Digital Dan Mikroprosesor*. Yogyakarta : Penerbit ANDI.
- Djuandi, Feri. 2011. *Pengenalan arduino*, <http://www.tobuku.com/docs/Arduino-Pengenalan.pdf>. Diakses tanggal 21 Agustus 2017.
- Istiyanto, Eko Jazi. 2014. *Pengantar Elektronika dan Instrumentasi*. Yogyakarta: Penerbit ANDI.
- Kadir, Abdul dan Heriyanto. 2005. *Algoritma Pemrograman Menggunakan C++*. Yogyakarta: Andi Offset.
- Kadir, Abdul. 2016. *Scratch for Arduino (S4A)-Panduan Mempelajari Elektronika dan Pemograman*. Yogyakarta. Penerbit ANDI.
- Prawiroredjo, Kiki. Nyssa Asteria. 2008. *Detektor jarak dengan sensor Ultrasonik berbasis Mikrokontroler*. Dosen jurusan Teknik Elektro-FTI Universitas Trisakti, JETri Vol. 7, Nmr 2.
- Supriyadi, Nur. 2014. *Laporan Praktikum Mikroprosesor Modul II Seven Segment, Keypad dan Lcd*. FMIPA UNPAD.



LAMPIRAN

LISTING PROGRAM ARDUINO

```
#include <Wire.h>

#include <LiquidCrystal.h>

#include <Servo.h>

#include <Keypad.h>

#define pengaduk 6

#define penyebar 5

Servo myservo;

LiquidCrystal lcd(4, 2, A0, A1, A2, A3);

unsigned long processingTime = 5000; // set limit proses in miliSecond

const int DS1307 = 0x68; // Address of DS1307 see data sheets

const byte ROWS = 4; //four rows

const byte COLS = 3; //four columns

//define the symbols on the buttons of the keypads

char hexaKeys[ROWS][COLS] = {

  {'1','2','3'},

  {'4','5','6'},

  {'7','8','9'},

  {'*','0','#'}};

};

const char* days[7] =

{"Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"};

const char* months[12] =
```

```
 {"Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug","Sep", "Oct", "Nov",  
 "Dec"};  
  
 byte rowPins[ROWS] = {7, 8, 9, 10}; //connect to the row pinouts of the keypad  
  
 byte colPins[COLS] = {11, 12, 13 }; //connect to the column pinouts of the  
 keypad  
  
 byte second = 0;  
  
 byte minute = 0;  
  
 byte hour = 0;  
  
 byte weekday = 0;  
  
 byte monthday = 0;  
  
 byte month = 0;  
  
 byte year = 0;  
  
 byte dataPengerjaan = 0;  
  
 long waktu = 10000;  
  
 String Text;  
  
 String _data1,_data2;  
  
 String dataSetPoint_ONE = "06:30";  
  
 String dataSetPoint_TWO = "17:00";  
  
 String dataTimeNow,dataTimeLow;  
  
 Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins,  
 ROWS, COLS);  
  
 ///////////////////////////////////////  
  
 ///////////////////////////////////////  
  
 void setup(){
```

```
myservo.attach(3);

Wire.begin();

Serial.begin(9600);

delay(10);

lcd.begin(16, 2);

myservo.write(0);

lcd.clear();

lcd.print("Init Success");

delay(1000);

pinMode(pengaduk,OUTPUT);
pinMode(penyebar,OUTPUT);
digitalWrite(pengaduk, LOW);
digitalWrite(penyebar, LOW);
readTimeNow();
menuReady();
}

unsigned long interval,interval1;

////////////////////////////////////

////////////////////////////////////

void menuReady(){

    if(millis(>)interval1+9000)interval1=millis();

    else if(millis(>)interval1+5000 && millis(<)interval1+7000){

        lcd.setCursor(0,0); lcd.print("SetPoint :      ");
```

```
lcd.setCursor(0,1); lcd.print(" "); lcd.print(dataSetPoint_ONE); lcd.print(" ");  
lcd.print(dataSetPoint_TWO); lcd.print(" ");  
}  
else if(millis()>interval1+7000 && millis()<interval1+9000){  
    lcd.setCursor(0,0); lcd.print("==SISTEM READY==");  
    lcd.setCursor(0,1); lcd.print("Press # to SET ");  
}  
else if(millis()>interval1 && millis()<interval1+5000){  
    lcd.setCursor(0,1);  
    lcd.print(String(days[weekday-1]) + " " + String(months[month-1]) + " " +  
String(monthday) + ", 20" + String(year));  
    lcd.setCursor(0,0);  
    lcd.print(" ");  
    if(hour < 10) lcd.print("0");  
    lcd.print(hour);  
    lcd.print(":");  
    if(minute < 10) lcd.print("0");  
    lcd.print( minute);  
    lcd.print(":");  
    if(second < 10) lcd.print("0");  
    lcd.print(second);  
    lcd.print(" ");  
}  
}
```

```
//////////////////////////////////  
//////////////////////////////////  
  
void loop(){  
  
    menuReady();  
  
    check_Keypad();  
  
    if(millis()>interval + 1000){  
  
        readTimeNow();  
  
        check_SetPoint();  
  
        interval = millis();  
  
    }  
  
    }  
  
//////////////////////////////////  
//////////////////////////////////  
  
void check_Keypad(){  
  
    char customKey = customKeypad.getKey();  
  
  
    if(customKey != NO_KEY){  
  
        //    pressing();  
  
        if(customKey == '#'){  
  
            Setting();  
  
        }  
  
    }  
  
    }  
  
}
```



```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////  
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////  
  
void SettingSetPOINT_ONE(){  
  
    dataSetPoint_ONE = _takePOINT();  
  
    dataPengerjaan = 0;  
  
}  
  
void SettingSetPOINT_TWO(){  
  
    dataSetPoint_TWO = _takePOINT();  
  
    dataPengerjaan = 0;  
  
}  
  
String _takePOINT(){  
  
    lcd.clear();  
  
    lcd.setCursor(0,0); lcd.print("Setting SetPoint");  
  
    lcd.setCursor(0,1); lcd.print("Loading...");  
  
    delay(1500);  
  
    String _data, _dat1, _dat2;  
  
    lcd.clear();lcd.setCursor(0,0);  
  
    lcd.print("Hour, 0-23. ");  
  
    lcd.setCursor(0,1); hour = readByte();  
  
    lcd.setCursor(12,1);lcd.print(hour);delay(1000);  
  
    lcd.clear();lcd.setCursor(0,0);  
  
    lcd.print("Minute, 0-59. ");  
  
    lcd.setCursor(0,1); minute = readByte();
```

```
lcd.setCursor(12,1);lcd.print(minute);delay(1000);

lcd.clear();

lcd.print("SetPoint Entered");

lcd.setCursor(5,1);

if(hour<10){lcd.print("0");_dat1 = "0";}

lcd.print(hour);

lcd.print(":");

if(minute<10){lcd.print("0");_dat2 = "0";}

lcd.print(minute);

_data = "";

_data = _data + _dat1 + String(hour) + ":" + _dat2 + String(minute);

Serial.println(_data);

delay(1000);

return _data;

}

////////////////////////////////////

////////////////////////////////////

void SettingTimmer(){

lcd.clear();

lcd.setCursor(0,0); lcd.print("Setting Timmer");

lcd.setCursor(0,1); lcd.print("Loading...");

delay(2000);

byte a = 1;
```

```
while(a == 1){  
  
  lcd.begin(16,2);  
  
  lcd.clear();lcd.setCursor(0,0);  
  
  lcd.print("Year, 00-99. ");  
  
  lcd.setCursor(0,1); year = readByte();  
  
  lcd.setCursor(12,1);lcd.print("20" + String(year));delay(1000);  
  
  lcd.clear();lcd.setCursor(0,0);  
  
  lcd.print("Month, 1-12. ");  
  
  lcd.setCursor(0,1); month = readByte();  
  
  lcd.setCursor(12,1);lcd.print(months[month-1]);delay(1000);  
  
  lcd.clear();lcd.setCursor(0,0);  
  
  lcd.print("Date, 1-31. ");  
  
  lcd.setCursor(0,1);monthday = readByte();  
  
  lcd.setCursor(12,1);lcd.print(monthday);delay(1000);  
  
  lcd.clear();lcd.setCursor(0,0);  
  
  lcd.print("Day, 0-7. ");  
  
  lcd.setCursor(0,1); weekday = readByte();  
  
  lcd.setCursor(12,1);lcd.print(days[weekday-1]);delay(1000);  
  
  lcd.clear();lcd.setCursor(0,0);  
  
  lcd.print("Hour, 0-23. ");  
  
  lcd.setCursor(0,1); hour = readByte();  
  
  lcd.setCursor(12,1);lcd.print(hour);delay(1000);  
  
  lcd.clear();lcd.setCursor(0,0);  
  
  lcd.print("Minute, 0-59. ");
```

```
lcd.setCursor(0,1); minute = readByte();

lcd.setCursor(12,1);lcd.print(minute);delay(1000);

second = 0;

// The following codes transmits the data to the RTC

Wire.beginTransmission(DS1307);

Wire.write(byte(0));

Wire.write(decToBcd(second));

Wire.write(decToBcd(minute));

Wire.write(decToBcd(hour));

Wire.write(decToBcd(weekday));

Wire.write(decToBcd(monthday));

Wire.write(decToBcd(month));

Wire.write(decToBcd(year));

Wire.write(byte(0));

Wire.endTransmission();

// Ends transmission of data

readTimeNow();

menuReady();

a=0;

}

}
```

```
////////////////////////////////////  
////////////////////////////////////  
  
void check_SetPoint(){  
    if(dataSetPoint_TWO == dataTimeNow){  
        if(dataPengerjaan == 0){  
            dataPengerjaan = 1;  
            proses();  
        }  
    }  
  
    else if(dataSetPoint_ONE == dataTimeNow){  
        if(dataPengerjaan == 0){  
            dataPengerjaan = 1;  
            proses();  
        }  
    }  
}  
  
////////////////////////////////////  
////////////////////////////////////  
  
void proses(){  
    lcd.clear();  
  
    lcd.setCursor(0,0);  
  
    lcd.print("PAKAN IKAN AKTIF");  
  
    myservo.write(90); //buka  
  
    delay(1000);
```

```
    analogWrite(pengaduk, 80); //80-255

    analogWrite(penyebar, 70); // min

    delay(waktu);

    analogWrite(penyebar, 120); // med

    delay(waktu);

    analogWrite(penyebar, 255); //max

    delay(waktu);

    analogWrite(penyebar, 0);

    myservo.write(0); //tutup

    analogWrite(pengaduk, 0);

    delay(500);

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print(" Proses Selesai ");

    delay(1500);

}

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

////////////////////////////////////

void readTimeNow() {

    Wire.beginTransmission(DS1307);

    Wire.write(byte(0));

    Wire.endTransmission();

}
```

```
Wire.requestFrom(DS1307, 7);

second = bcdToDec(Wire.read());

minute = bcdToDec(Wire.read());

hour = bcdToDec(Wire.read());

weekday = bcdToDec(Wire.read());

monthday = bcdToDec(Wire.read());

month = bcdToDec(Wire.read());

year = bcdToDec(Wire.read());

dataTimeNow = "";

_data1 = "";

_data2 = "";

if(hour<10){_data1 = "0";}

if(minute<10){_data2 = "0";}

dataTimeNow = dataTimeNow + _data1 + String(hour) + ":" + _data2 +

String(minute);

if(dataTimeNow != dataTimeLow){

    dataPengerjaan = 0;

}

dataTimeLow = dataTimeNow;

Serial.println(String("time now  :") +dataTimeNow);

Serial.println(String("time low  :") +dataTimeLow);

Serial.println(String("data Kerja :") +dataPengerjaan);

Serial.println(String("setPoint1  :") +dataSetPoint_ONE);

Serial.println(String("setPoint2  :") +dataSetPoint_TWO);
```

```
Serial.println();Serial.println();

}

////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////

byte decToBcd(byte val) {
    return ((val/10*16) + (val%10));
}

byte bcdToDec(byte val) {
    return ((val/16*10) + (val%16));
}

byte readByte() {
    byte a = 1;
    byte z = 0;
    byte b[2] = {};
    while (a==1){
        char customKey = customKeypad.getKey();

        if (customKey != NO_KEY){

//    pressing();

            if(customKey != '#' && customKey != '*') {

                if(z < 2){

                    b[z] = customKey;

                    lcd.print(customKey);
```

```
z++;  
  
}  
  
}  
  
else if (customKey == '*'){  
  
    lcd.setCursor(0,1);lcd.print("      ");  
  
    lcd.setCursor(0,1);  
  
    z = 0; b[1] = NULL;  
  
    a = 1;  
  
}  
  
else if (customKey == '#'){  
  
    a = 0;  
  
}  
  
}  
  
}  
  
z = 0;  
  
byte reading = 0;  
  
byte incomingByte = b[0];  
  
lcd.setCursor(7,1);lcd.print("Set, ");  
  
while (incomingByte != NULL ) {  
  
    //Serial.println(incomingByte);  
  
    if (incomingByte >= '0' && incomingByte <= '9' && reading < 10){  
  
        reading = reading * 10 + (incomingByte - '0');  
  
    }  
  
    else;
```

```
incomingByte = b[1];  
  
if(z > 1) incomingByte = NULL;  
  
z++;  
  
}  
  
return reading;  
  
}
```

