

**RANCANG BANGUN ALAT PEMBERI PAKAN UDANG
VANAME (*SEMI AUTOMATIC FEEDER*) BERBASIS IOT
(*INTERNET OF THINGS*)**

PROYEK AKHIR

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat
Kelulusan Diploma III Politeknik Manufaktur Negeri Bangka Belitung



Disusun Oleh :

Muhammad Anzullah NIRM : 0031645

Rachmah Suci Saputri NIRM : 0031652

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2019**

LEMBAR PENGESAHAN

RANCANG BANGUN ALAT PEMBERI PAKAN UDANG VANAME (SEMI AUTOMATIC FEEDER) BERBASIS IOT (INTERNET OF THINGS)

Oleh :

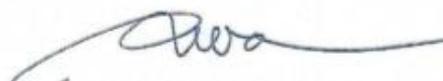
Muhammad Anzullah NIRM : 0031645

Rachmah Suci Saputri NIRM : 0031652

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan
Program Diploma III Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1



Irwan, M.Sc., Ph.D

Pembimbing 2



Muhammad Iqbal Nugraha, M.Eng

Penguji 1



Ocsirendi, M.T.

Penguji 2



Zanu Saputra, M.T.

Penguji 3



Dr. Parulian Silalahi, M.Pd.

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini :

Nama Mahasiswa 1 : Muhammad Anzullah NIRM : 0031645

Nama Mahasiswa 2 : Rachmah Suci Saputri NIRM : 0031652

Dengan Judul : RANCANG BANGUN ALAT PEMBERI PAKAN UDANG
VANAMEI (*SEMI AUTOMATIC FEEDER*) BERBASIS IOT
(*INTERNET OF THINGS*)

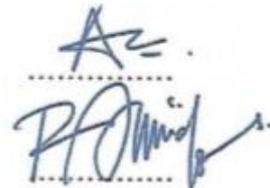
Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

Sungailiat, 29 Agustus 2019

Nama Mahasiswa

1. Muhammad Anzullah
2. Rachmah Suci Saputri

Tanda Tangan



The image shows two handwritten signatures in blue ink. The first signature is 'Az.' and the second is 'Rachmah Suci Saputri'. Both signatures are written over a dotted line.

ABSTRAK

Pada saat ini telah banyak dibuat dan dikembangkan tambak udang vaname khususnya di Kepulauan Bangka Belitung. Salah satu masalah yang saat ini dihadapi bagi para pembudidaya udang yaitu proses pemberian pakan yang dilakukan secara manual dinilai kurang produktif karena memerlukan banyak tenaga kerja dan waktu yang cukup banyak. Selain itu pemberian pakan secara manual dapat menyebabkan sisa pakan menumpuk pada satu titik, hal ini dapat menyebabkan kualitas air pada tambak menurun. Jika hal ini terjadi maka dapat membuat pertumbuhan udang menjadi kurang baik. Dari persoalan tersebut, maka dibuatlah sebuah alat yang bernama Automatic Feeder berbasis Internet of Things (IoT). Tujuan dari penelitian ini adalah untuk mempermudah petani dalam proses pemberian pakan di tambak udang. Sistem ini menggunakan Arduino Mega 2560 dan NodeMCU sebagai komponen utama dalam hal mengendalikan motor DC 12 volt dan 24 volt. Sementara itu untuk memonitoring proses makan udang di tambak serta mengetahui indikator jumlah pakan udang menggunakan RTC dan sensor Ultrasonic. Alat ini dapat berfungsi secara otomatis yang dikontrol dan dimonitor melalui sebuah Android. Automatic Feeder ini dapat dikontrol secara otomatis menggunakan Smartphone dan manual dengan keypad, dengan ketinggian alat mencapai 160 cm lontaran pakan dapat mencapai 15 meter, alat ini dapat memonitoring jumlah pakan yang masih tersisa serta waktu dalam proses pemberian pakan yang sedang dilakukan, dan dengan penggunaan PWM pada driver motor dapat membuat pakan terbagi secara merata.

Kata Kunci : *Udang Vaname, Automatic Feeder, Internet of Things*

ABSTRACT

At this time many vaname shrimp ponds have been made and developed especially in the Bangka Belitung Islands. One of the problems currently faced by shrimp farmers is that the manual feeding process is considered to be less productive because it requires a lot of labor and considerable time. Besides manual feeding can cause the rest of the feed to accumulate at one point, this can cause water quality in the pond to decline. If this happens it can make shrimp growth to be less good. From this problem, then made a tool called Automatic Feeder based on Internet of Things (IoT). The purpose of this research is to facilitate farmers in the process of feeding on shrimp ponds. This system uses the Arduino Mega 2560 and NodeMCU as the main components in terms of controlling 12 volt and 24 volt DC motors. Meanwhile, to monitor the process of eating shrimp in ponds and to find out the indicator of the amount of shrimp feed using RTC and Ultrasonic sensors. This tool can function automatically which is controlled and monitored via an Android. This Automatic Feeder can be controlled automatically using a Smartphone and manually with a keypad, with the height of the tool reaching 160 cm the feed can reach 15 meters, this tool can monitor the amount of feed remaining and the time in the feeding process that is being done, and with the use of PWM the motor driver can make the feed evenly distributed.

Keyword : *Vaname Shrimp, Automatic Feeder, Internet of Things*

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena berkat rahmat dan karunia-Nya penulis dapat menyelesaikan laporan Proyek Akhir ini dengan baik dan tepat waktu. Laporan ini disusun sebagai sebagai salah satu persyaratan mahasiswa dalam menyelesaikan Proyek Akhir ini.

Dalam Proyek Akhir ini penulis membuat sebuah alat yang bernama *Automatic Feeder*. Dimana alat ini berfungsi untuk memberikan pakan udang di tambak. Sistem kontrol yang digunakan pada alat ini yaitu penggunaan Arduino yang dikoneksikan melalui aplikasi pada Android menggunakan komunikasi internet. Dalam pembuatan Proyek Akhir ini penulis mengakui bahwa selesainya Proyek Akhir ini tidak lepas dari bantuan dan dukungan dari pihak-pihak tertentu. Oleh karena itu penulis mengucapkan terima kasih kepada :

1. Bapak Irwan, Ph.D, selaku pembimbing 1 dalam Proyek Akhir ini
2. Bapak Muhammad Iqbal Nugraha, M.Eng, selaku pembimbing 2 dalam Proyek Akhir ini.
3. Dedhe Jumriladin Putra Susila Mahasiswa tingkat 2 TMM A Politeknik Manufaktur Negeri Bangka Belitung, yang telah membantu dalam proses pengerjaan mekanik pada alat ini.
4. Bapak Eko Sulisty, M.T, selaku Kepala Prodi Teknik Elektronika dan Informatika Politeknik Manufaktur Negeri Bangka Belitung.
5. Bapak Aan Febriansyah, M.T, selaku Kepala Jurusan Teknik Elektronika dan Informatika Politeknik Manufaktur Negeri Bangka Beliung.
6. Bapak Sugeng Ariyono, M.Eng, Ph.D selaku Direktur Politeknik Manufaktur Bangka Belitung.
7. Bapak Surojo, M.T, selaku wali mahasiswa.
8. Seluruh staf pengajar dan karyawan di Politeknik Manufaktur Negeri Bangka Belitung.

9. Orang tua beserta keluarga yang lainnya, yang selalu memberikan kasih sayang, doa serta dukungan kepada penulis.
10. Teman-teman yang telah ikut mendukung dan memberikan bantuan serta masukan dalam pembuatan Proyek Akhir ini.
11. Pihak-pihak lain yang telah memberikan bantuan secara langsung maupun tidak langsung yang tidak dapat disebutkan satu per satu.

Dalam laporan Proyek Akhir ini penulis menyadari masih terdapat kekurangan, oleh karena itu penulis sangat mengharapkan saran, kritik dan masukan yang bersifat membangun dalam rangka perbaikan laporan ini. Demikianlah laporan ini dibuat dan semoga laporan ini dapat bermanfaat dan menambah wawasan bagi para pembaca. Akhir kata penulis mengucapkan terima kasih.

Sungailiat, 29 Agustus 2019

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN.....	ii
PERNYATAAN BUKAN PLAGIAT.....	iii
ABSTRAK.....	iv
ABSTRACT.....	v
KATA PENGANTAR.....	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR.....	x
DAFTAR LAMPIRAN.....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Proyek Akhir.....	3
BAB II DASAR TEORI.....	4
2.1 Udang Vaname.....	4
2.2 <i>Automatic Feeder</i>	5
2.2.1 IoT (<i>Internet of Things</i>).....	6
2.2.2 Cara Kerja <i>Automatic Feeder</i>	7
BAB III METODE PELAKSANAAN.....	9
3.1 Tahap Pertama.....	9
3.2 Tahap Kedua.....	9
3.3 Tahap Ketiga.....	10
3.4 Tahap Keempat.....	11
3.5 Tahap Kelima.....	11
BAB IV PEMBAHASAN.....	12
4.1 Deskripsi Alat.....	12
4.2 Sistem Mekanik <i>Automatic Feeder</i>	13
4.2.1 Wadah Penampung Pakan.....	15

4.2.2 Corong Penyalur Pakan.....	15
4.2.3 Penampung dan Penyalur Pakan.....	16
4.2.4 Motor DC 12V.....	17
4.2.5 Pelontar Pakan.....	18
4.2.6 Motor DC 24 volt.....	18
4.2.7 Tiang Penyangga dan Tiang Pengunci.....	19
4.3 Rangkaian Kontrol dan Komunikasi.....	21
4.3.1 Program Penyalan motor DC 24V dan 12V.....	23
4.3.2 Program Pengiriman Data Serial Arduino Mega ke NodeMCU.....	23
4.3.3 Program NodeMCU.....	24
4.3.4 Seting Aplikasi Monitoring pada <i>Smartphone</i> (Blynk).....	24
4.4 Uji Coba <i>Hardware</i>	27
4.4.1 Pengujian Motor DC 12 volt & 24 volt.....	27
4.4.2 Pengujian Jarak Lontaran Pakan.....	28
4.4.3 Pengujian Keluaran Pakan.....	28
4.5 Uji Coba <i>Software</i>	29
4.5.1 Pengujian Koneksi ON/OFF Melalui <i>Software</i> Blynk.....	29
4.5.2 Pengujian Arduino, <i>Real Time Clock</i> (RTC) dan NodeMCU.....	30
4.5.3 Pengujian Arduino, Sensor Ultrasonik dan NodeMCU.....	31
4.5.4 Koneksi Arduino, <i>Keypad</i> dan <i>Liquid Crystal Display</i> (LCD).....	31
BAB V KESIMPULAN DAN SARAN.....	33
5.1 Kesimpulan.....	33
5.2 Saran.....	33
DAFTAR PUSTAKA.....	344

DAFTAR TABEL

Tabel 4. 1 Jarak lontaran pakan.....	28
Tabel 4. 2 Uji coba keluaran pakan per menit.	29

DAFTAR GAMBAR

Gambar 2. 1 Udang Vaname	4
Gambar 2. 2 <i>Automatic Feeder</i>	8
Gambar 3. 1 Diagram blok <i>Automatic Feeder</i> secara umum.....	10
Gambar 4. 1 Diagram blok <i>Automatic Feeder</i>	13
Gambar 4. 2 <i>Automatic Feeder</i> berbasis <i>Internet of Things</i>	14
Gambar 4. 3 Wadah penampung pakan	15
Gambar 4. 4 Corong penyalur pakan	16
Gambar 4. 5 Penguat sisi corong.....	16
Gambar 4. 6 Penampung dan penyalur pakan.....	17
Gambar 4. 7 Peletakkan motor DC 12 volt	17
Gambar 4. 8 Pelontar pakan	18
Gambar 4. 9 Penempatan motor DC 24 volt	18
Gambar 4. 10 Pengunci wadah pakan	19
Gambar 4. 11 Tiang penyangga <i>Automatic Feeder</i>	20
Gambar 4. 12 <i>Automatic Feeder</i> berbasis <i>Internet of Things</i>	20
Gambar 4. 13 Rangkaian kontrol	21
Gambar 4. 14 <i>Flowchart</i> sistem kontrol	22
Gambar 4. 15 Blok diagram NodeMCU	23
Gambar 4. 16 Pengaturan komponen ultrasonik	25
Gambar 4. 17 Pengaturan komponen <i>push button</i>	26
Gambar 4. 18 Pengaturan komponen notifikasi.....	26
Gambar 4. 19 Blok diagram uji coba penyalaan motor DC 24 volt dan 12 volt...	27
Gambar 4. 20 Tampilan <i>Push Button</i>	30
Gambar 4. 21 Monitoring RTC pada Blynk.....	31
Gambar 4. 22 Indikator pakan menggunakan Ultrasonik	31
Gambar 4. 23 Tampilan LCD dengan menggunakan <i>keypad</i>	32
Gambar 4. 24 Tampilan setingan waktu pada LCD.....	32

DAFTAR LAMPIRAN

1. Lampiran : Daftar Riwayat Hidup
2. Lampiran : Desain *Automatic Feeder*
3. Lampiran : Program Arduino Mega 2560
4. Lampiran : Program NodeMCU
5. Lampiran : SOP Penggunaan *Automatic Feeder*

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan usaha budi daya udang saat ini telah banyak dilakukan di Indonesia khususnya di Pulau Bangka sendiri. Untuk mendapatkan hasil udang yang baik, diperlukan proses budidaya yang baik serta lingkungan yang baik. Lingkungan yang dimaksud yaitu air yang terdapat pada tambak tersebut. Semakin bagus kualitas air semakin baik pula lingkungan yang dihasilkan [1]. Tetapi kualitas air juga bergantung dari jumlah pakan yang ditebar, semakin banyak pakan yang ditebar maka penurunan kualitas air akan semakin cepat. Salah satu upaya yang dapat dilakukan agar kualitas air tetap terjaga yaitu dengan menebar pakan secara merata dengan menggunakan mesin *Automatic Feeder*. Menurut penelitian yang telah dilakukan oleh DJPB dengan menggunakan *Automatic Feeder* secara nyata mampu mengurangi masa pemeliharaan udang *Vanamei* secara intensif, yaitu dapat dipersingkat hingga 25% dibandingkan dengan cara manual. Sebagai gambaran, untuk mencapai ukuran udang size 35 gram biasanya diperlukan waktu 120 hari, namun dengan menggunakan *Automatic Feeder* bisa dipersingkat sekitar 90 hari. Alat ini sungguh sangat membantu sekali [1].

Sama halnya dengan tambak udang yang ada di Kabupaten Bangka, Kecamatan Belinyu Dusun Pejem, tambak ini juga telah menggunakan *Automatic Feeder*. Saat melakukan survei ke tambak tersebut, dapat dilihat bagaimana bentuk serta cara kerja dari mesin *Automatic Feeder* yang telah ada sebelumnya. Setelah diperhatikan, kelemahan *Automatic Feeder* ini yaitu pakan yang dialirkan dari wadah langsung ke pelontar, hal ini sering membuat pakan menumpuk sebelum sempat terlontarkan ke luar, kemudian jarak akses antara alat dengan *Smartphone* hanya sejauh 100 meter. Karena tidak adanya kaki penyanggah maka pengguna harus menyediakan dudukan untuk alat di dalam tambak. Kemudian bagian alat yang lainnya terdiri dari wadah penampung pakan berbentuk pelet. Wadah yang

berbentuk tong ini memiliki kapasitas tampung ± 50 kg. Pemutaran wadah dilakukan dengan sebuah motor AC 3 *phase* dengan putaran sebanyak ± 1500 RPM. Dengan putaran tersebut, pakan dapat menyebar ke titik 10 m dari posisi mesin dan ketinggian *Automatic Feeder* tidak dapat diatur. Dari survei yang dilakukan ini, inovasi yang akan dilakukan untuk mengembangkan *Automatic Feeder* ini dengan penambahan sistem kontrol berbasis IoT (*Internet of Things*), menambahkan pengatur ketinggian pada alat dan penyebaran pakan dapat terbagi dari jarak dekat hingga jauh.

Dengan adanya alat ini dapat mempersingkat waktu para pekerja tambak dan menebar pakan udang secara merata agar pertumbuhan udang dapat berjalan dengan semestinya sehingga meminimalisir kerugian bagi para pembudidaya udang.

1.2 Rumusan Masalah

Berikut adalah permasalahan yang terjadi di tambak udang yang menyebabkan masalah bagi para pembudidaya udang yaitu :

1. Bagaimana proses agar dapat membantu penebaran pakan tidak menumpuk pada satu titik tertentu.
2. Dengan penggunaan *Automatic Feeder* bagaimana cara agar dapat mengefisiensi waktu para pekerja tambak.
3. Bagaimana agar jauhnya lontaran pakan yang dikeluarkan dapat diatur sesuai keinginan pengguna.

1.3 Batasan Masalah

Proyek Akhir ini dibatasi oleh beberapa ruang lingkup permasalahan yang merupakan batasan dari kemampuan *Automatic Feeder* ini yaitu :

1. Lontaran pakan terbatas yaitu maksimal 15 meter sehingga tidak seluruh bagian tambak yang dapat terbagi pakan.
2. Pemberian pakan dengan *Automatic Feeder* ini untuk usia udang di atas 30 hari.

3. Kapasitas wadah hanya dapat menampung pakan sebanyak 50 kg (1 kali makan sebanyak 25 kg).

1.4 Tujuan Proyek Akhir

Tujuan dari dibuatnya *Automatic Feeder* pada Proyek Akhir ini yaitu :

1. Menghasilkan alat dengan spesifikasi dapat melontarkan pakan mulai dari 0.2 meter sampai dengan 15 meter serta dapat diatur ketinggiannya.
2. Menghasilkan alat yang dapat dimonitor waktu saat pemberian pakan dilakukan dan mengetahui sisa pakan yang tersedia menggunakan aplikasi Blynk pada *Smartphone*.

BAB II

DASAR TEORI

2.1 Udang Vaname

Udang Vaname (*Litopenaeus Vanname*) merupakan jenis udang introduksi yang berasal dari Amerika Selatan yang banyak dibudidayakan di Indonesia sejak akhir tahun 90-an untuk menggantikan udang Windu (*Penaeus Monodon*) yang sudah sulit dibudidayakan karena rentan akan penyakit bintik putih (*White Spot*) [2]. Perkembangan budidaya udang jenis vaname ini pun mulai berkembang di Bangka Belitung. Udang ini bisa hidup di air asin. Pertumbuhan udang vaname sangat bergantung pada kualitas air dan proses pemberian pakan. Untuk menjaga kualitas air pada tambak digunakan suatu kincir yang berfungsi untuk menjaga sirkulasi air agar tetap stabil, menghasilkan oksigen dan bisa membuat sisa pakan yang menumpuk pada dasar tambak mengumpul pada titik pembuangan sisa pakan yang terletak pada tengah-tengah tambak. Menurut riset yang telah diteliti pada tambak udang yang ada di Sulawesi Selatan, aspek-aspek yang perlu diperhatikan dalam usaha budi daya udang vaname pola tradisional plus antara lain: persiapan tambak, kualitas benih, teknik penebaran, padat penebaran, manajemen pakan, pemeliharaan kualitas air, dan teknik panen [3]. Di bawah ini adalah gambar udang vaname yang banyak dibudidayakan terdapat pada gambar 2.1 berikut ini :



Gambar 2. 1 Udang Vaname [4]

Usia panen untuk udang vanamei sekitar umur 100-110 hari (tergantung permintaan konsumen). Pada usia ini berat udang bisa mencapai rata-rata ± 35 gram per ekor. Udang ini termasuk jenis udang pemakan yang sangat agresif. Setiap hari terjadi penambahan dosis pakan yang diberikan sesuai dengan usia pertumbuhan udang. Biasanya penambahan pakan yang diberikan per 0,3% setiap hari nya. Untuk pemasaran hasil panen udang, pembudidaya menjual hasil panen ke luar daerah Bangka Belitung dan juga kemudian di ekspor ke luar negeri. Hal ini karena untuk pengelolaan udang vaname sendiri belum ada yang dilakukan di Pulau Bangka, baik itu untuk produksi sebuah produk ataupun yang lainnya. Selain itu dengan dilakukan penjualan ke luar negeri tentunya akan mendapatkan pendapatan yang lebih besar bagi para pembudidaya udang vaname.

Pembudidayaan udang vaname ini bergantung pada kualitas air pada tambak. Oleh karena itu, pemberian pakan secara manual dapat dikatakan kurang baik, karena dapat membuat pakan menumpuk pada titik tertentu, apabila terlalu banyak nya pakan pada titik tersebut, dapat membuat sisa-sisa pakan yang tidak dimakan oleh udang mengendap ke dasar tambak. Semakin banyak sisa pakan yang mengendap dapat menjadi limbah pada air tambak sehingga membuat kualitas air pada tambak menjadi kurang baik. Untuk menghindari hal ini terjadi dibutuhkan nya sebuah *Automatic Feeder* yang dapat membantu penebaran pakan udang agar tidak menumpuk pada titik tertentu saja.

2.2 Automatic Feeder

Automatic Feeder adalah alat pemberi pakan udang yang bekerja secara otomatis dalam hal pemberian jadwal pakan, yang dilakukan selama 24 jam secara terus menerus [5]. Alat ini diciptakan untuk memudahkan para penambak udang dalam mengatasi masalah pemberian pakan terhadap udang itu sendiri. Diambil lah contoh pada *Automatic Feeder* yang telah ada sebelumnya yaitu merk e-Fishery, alat ini dilengkapi dengan sebuah wadah penampung pakan yang dapat menampung kapasitas pakan sebesar 50 kg. kemudian disalurkan langsung ke pelontar pakan, alat ini memiliki 2 bagian pelontar yaitu kiri dan kanan. Pemutaran pelontar pakan

dilakukan dengan menggunakan motor listrik AC. Pada *Automatic Feeder* ini memiliki kaki yang tetap sehingga petambak harus menyediakan kaki untuk tempat peletakan di dalam tambak. Pada sistem lontaran *Automatic Feeder* yang telah ada ini lontaran langsung mencapai titik 10 meter.

Alat yang menjadi tugas akhir ini merupakan pengembangan dari alat yang telah ada sebelumnya. Pada *Automatic Feeder* sebelumnya masih terdapat kekurangan yaitu menggunakan motor AC 3 *phase*, pengaturan waktu makan secara manual, jalur keluar pakan menuju pelontar sering tersumbat, dudukan *Automatic Feeder* masih kurang efektif, pemberian dosis pakan masih dilakukan secara manual dan tidak ada sensor pendeteksi jumlah pakan pada wadah penampung pakan.

Dari kekurangan-kekurangan tersebut maka dibuatlah sebuah *Automatic Feeder* yang sistem kerjanya lebih dikembangkan dari alat sebelumnya. *Automatic Feeder* ini akan menggunakan sistem otomatis berbasis *Internet of Things (IoT)* dengan jaringan GSM. *Automatic Feeder* yang telah dikembangkan ini dapat membantu para pembudidaya udang agar lebih mudah dan praktis dalam proses pemberian pakan. Semakin mudahnya proses pemberian pakan yang dilakukan, dapat membantu meningkatkan hasil produksi serta waktu dan tenaga pembudidaya menjadi lebih efektif.

2.2.1 IoT (*Internet of Things*)

Internet of Things merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dan konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuannya seperti berbagi data, remot kontrol, dan sebagainya. Pada dasarnya, *Internet of Things* mengacu pada benda yang dapat diidentifikasi secara unik sebagai representasi virtual dalam struktur berbasis internet [6]. IoT yang digunakan pada *Automatic Feeder* kali ini memiliki sistem kerja menggunakan kartu GSM untuk mengaktifkan jaringan pada NodeMCU. Jika NodeMCU telah tersambung ke jaringan internet, maka aplikasi Blynk yang digunakan pada *Smartphone* akan terkoneksi ke NodeMCU. Jika Blynk telah

terkoneksi maka user dapat mengaktifkan dan mematikan *Automatic Feeder* dari jarak jauh.

IoT merupakan teknologi yang memungkinkan benda-benda di sekitar kita terhubung dengan jaringan internet. Teknologi ini ditemukan oleh Kevin Ashton pada tahun 1999. Hingga saat ini, teknologi IoT sudah dikembangkan dan diaplikasikan. Cara kerjanya yaitu setiap benda yang terhubung dengan internet kapan saja dimana saja [7]. Sehingga dapat disimpulkan bahwa IoT adalah sebuah teknologi yang dapat membantu mempermudah pekerjaan manusia dengan memanfaatkan jaringan internet. Contoh user dapat mengaktifkan atau mematikan suatu benda dari jarak jauh dengan memanfaatkan *Smartphone* yang tersambung ke jaringan internet yang ada.

Cara kerja *Internet of Things* yaitu dengan memanfaatkan sebuah argumentasi pemrograman yang dimana tiap-tiap perintah argumennya itu menghasilkan sebuah interaksi antara sesama mesin yang terhubung secara otomatis tanpa campur tangan manusia dan dalam jarak berapa pun. Internet lah yang menjadi penghubung di antara kedua interaksi mesin tersebut, sementara manusia hanya bertugas sebagai pengatur dan pengawas bekerjanya alat tersebut secara langsung. Tantangan terbesar dalam mengkonfigurasi *Internet of Things* ialah menyusun jaringan komunikasinya sendiri, yang dimana jaringan tersebut sangatlah kompleks, dan memerlukan sistem keamanan yang ketat [8].

2.2.2 Cara Kerja *Automatic Feeder*

Automatic Feeder merupakan suatu alat yang di fungsikan untuk membantu atau mempermudah para petani atau pembudidaya udang dalam pemberian pakan yang menggunakan pelet agar pemberian pakan dapat dilakukan tepat waktu dan sesuai dengan dosis makan yang diinginkan. Alat ini bekerja secara otomatis dalam proses pengeluaran dan penyebaran pakan dengan kontrol yang bisa dilakukan dari jarak jauh menggunakan Android. Untuk proses keluarnya pakan di atur sesuai dengan keinginan pengguna dengan setingan waktu yang dapat di atur selama 24 jam. Alat ini dikendalikan dengan menggunakan Arduino. Arduino ini berfungsi

sebagai media kontrol alat seperti *On/Off* alat, jalur keluar pakan, setingan waktu makan, jarak lontaran pakan dan banyaknya dosis pakan yang dikeluarkan.

Jalur keluarnya pakan secara otomatis di atur menggunakan motor DC yang dikontrol melalui Arduino. Sistem kontrol Arduino Mega 2560 ini diprogram dengan menggunakan komputer yang sudah dilengkapi dengan *software* Arduino. Dari Arduino ini nantinya akan terhubung ke Android pengguna. Media penghubung komunikasi antara Android dan Arduino menggunakan jaringan GSM yang terhubung ke mode *wifi nodeMCU* ESP8266. Jadi dari komunikasi ini nantinya akan terjadi proses timbal balik pengirim dan penerima perintah antara kontrol Arduino dan Android pengguna. Sehingga proses timbal balik inilah yang akan memberikan notifikasi apakah perintah yang diberikan sudah dijalankan atau belum.

Untuk kontruksi mekaniknya, menggunakan tiang penyangga tetap yang dilengkapi pengunci naik turun agar alat bisa menyesuaikan sesuai keinginan pembudidaya. Kontruksi naik turun ini berfungsi untuk mengatur jarak lontaran pakan yang diinginkan. Untuk penggunaan motor DC, kecepatan motor DC ini akan diatur menggunakan PWM agar kecepatannya berubah-ubah dari pelan hingga cepat kemudian pelan kembali sampai akhirnya berhenti, hal ini bertujuan agar pemberian pakan terjadi secara merata. Gambar 2.2 berikut ini adalah gambar *Automatic Feeder* yang telah ada.



Gambar 2. 2 *Automatic Feeder* [9]

BAB III

METODE PELAKSANAAN

Dalam proses pembuatan Proyek Akhir yang berjudul “*Rancang Bangun Automatic Feeder berbasis IoT*” terdiri dari beberapa tahapan sebagai berikut:

1. *Studi literature* dan pengumpulan data
2. Desain sistem *Automatic Feeder*
3. Pembuatan sistem mekanik dan elektronik.
4. Desain dan pembuatan perangkat lunak.
5. Pengujian.

Penjelasan lebih lanjut dari tahapan di atas dapat dilihat pada langkah-langkah berikut ini:

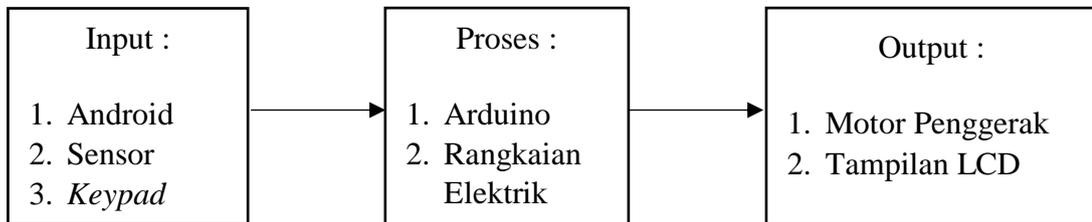
3.1 Tahap Pertama

Melakukan *studi literature* dan survei ke lapangan tentang *Automatic Feeder* pada pakan udang. *Studi literature* dan pengumpulan data ini bertujuan untuk mengetahui secara garis besar tentang *Automatic Feeder* yang digunakan ditambah udang, serta untuk mengatasi permasalahan yang terjadi. Data yang telah terkumpul dijadikan acuan untuk tahapan proses pembuatan *Automatic Feeder* selanjutnya.

3.2 Tahap Kedua

Mendesain Proyek Akhir “*Rancang Bangun Alat Automatic Feeder Udang Vanamei Berbasis IoT*” baik secara *hardware* dan *software*. Tujuannya agar mempermudah dalam proses pembuatan tugas akhir ini. Perancangan *hardware* ini berupa perancangan mekanik dengan menggunakan *software Corel Draw X6*, serta perancangan elektrik yang dibutuhkan dalam *Automatic Feeder* itu sendiri. Untuk perancangan *software* kami menggunakan aplikasi yang terdapat di Android yaitu Blynk, serta *software* Arduino sebagai sistem kontrol nya.

Berikut ini adalah diagram blok secara umum dari *Automatic Feeder* yang akan dibuat :



Gambar 3. 1 Diagram blok *Automatic Feeder* secara umum

3.3 Tahap Ketiga

Pembuatan *hardware* dan *software*. Untuk pembuatan *hardware* berupa pembuatan konstruksi mekanik yang berupa penentuan bahan yang digunakan, bagian-bagian alat serta ukuran yang digunakan. Perencanaan ini harus dilakukan dan difikirkan secara matang-matang hal ini bertujuan untuk mengurangi kesalahan pada saat pembuatan alat nantinya.

Berikut adalah proses pembuatan alat pada bagian mekanik yang meliputi:

1. Penentuan dan pembuatan wadah penampung pakan dengan menggunakan plat dengan ketebalan 0.6 mm.
2. Pembuatan corong penyalur pakan dengan menggunakan plat setebal 0.8 mm.
3. Pembuatan pipa penampung dan penyalur pakan, dengan menggunakan pipa PVC.
4. Pemasangan tata letak motor DC 12 volt sebagai media pemutar penampung pakan.
5. Pembuatan pelontar pakan dengan menggunakan *Stainless Steel 207*.
6. Pemasangan tata letak motor DC 24 volt sebagai penggerak pelontar.
7. Pembuatan kaki atau tiang penyangga *Automatic Feeder* beserta pengunci nya.
8. Perakitan bagian-bagian mekanik *Automatic Feeder*.

Sedangkan proses pembuatan alat pada bagian sistem kontrol meliputi :

1. Pemrograman ON/OFF *Automatic Feeder*

2. Pengaturan putaran motor DC 12 volt dan 24 volt
3. Pemrograman kontrol monitoring jumlah pakan menggunakan sensor *Ultrasonik*
4. Pemrograman monitoring waktu pemberian pakan

3.4 Tahap Keempat

Setelah proses pembuatan *hardware* dan *software* selesai, langkah selanjutnya yaitu proses uji coba alat. Uji coba ini bertujuan untuk mengetahui kerja alat apakah bekerja secara optimal dan berfungsi sesuai dengan fungsi yang diinginkan. Uji coba ini terdapat dua tahapan :

1. Uji Coba *Hardware*

Uji coba *hardware* yang dilakukan meliputi:

- Uji coba jarak lontaran pakan
- Uji coba keluaran pakan
- Uji coba naik turun kaki *Automatic Feeder*

2. Uji Coba *Software*

Uji coba *software* yang dilakukan meliputi:

- Uji coba koneksi antara Arduino, Android, dan *Mode Wifi Node MCU*
- Uji coba koneksi antara Arduino dan *Real Time Clock (RTC)*
- Uji coba koneksi antara Arduino dan Sensor Ultrasonik
- Uji coba koneksi antara Arduino, *Keypad* dan *Liquid Crystal Display (LCD)*

3. Uji Coba Keseluruhan

Uji coba secara keseluruhan ini dilakukan untuk mengetahui hasil pengujian dari keseluruhan alat yang telah dibuat.

3.5 Tahap Kelima

Pada tahap ini merupakan tahap analisis data pada hasil uji coba yang telah dilakukan. Hal ini bertujuan untuk melihat kekurangan terhadap alat yang telah dibuat baik dari segi konstruksi, rangkaian kontrol serta program yang dibuat.

BAB IV

PEMBAHASAN

Bab pembahasan ini menguraikan proses pengerjaan proyek akhir ini berdasarkan metode yang telah dipaparkan pada bab sebelumnya. Secara umum bab ini menguraikan tentang:

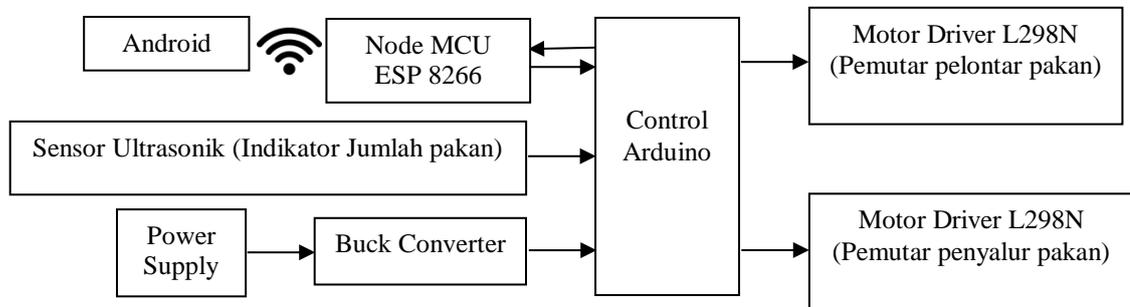
1. Deskripsi alat
2. Sistem mekanik dalam pembuatan *Automatic Feeder*
3. Rangkaian kontrol dan komunikasi
4. Pengujian *hardware*
5. Pengujian *software*

4.1 Deskripsi Alat

Automatic Feeder berbasis *Internet of Things* (IoT) adalah sebuah alat yang digunakan untuk memberikan pakan udang di tambak udang. Alat ini akan beroperasi secara otomatis dengan diaktifkan melalui sebuah aplikasi pada Android yaitu Blynk. Sementara untuk pengontrolan dan pemonitorannya menggunakan sebuah Arduino Mega 2560 yang telah terprogram beserta sebuah NodeMCU sebagai pengirim data yang akan ditampilkan pada Android. Pada sistem ini menggunakan sebuah sensor Ultrasonik yang di letakkan di dalam penutup wadah pakan yang berfungsi untuk memonitoring jumlah pakan.

Output pada alat ini akan mengaktifkan sebuah motor DC 24 volt yang berfungsi sebagai pelontar pakan dan sebuah motor DC 12 volt yang berfungsi sebagai pemutar penampung pakan. Aplikasi Blynk pada Android akan berfungsi sebagai *ON/OFF* alat disertai dengan notifikasi pemberitahuan. Pemberitahuan ini berupa tulisan yang otomatis akan tampil di layar *Smartphone* saat alat mulai beroperasi. Perbedaan dengan *Automatic Feeder* sebelumnya yaitu pada *Automatic Feeder* yang diperbaharui ini memiliki tiang pengunci yang bisa di atur ketinggiannya sesuai keinginan petambak itu sendiri.

Berikut adalah gambar diagram blok *Automatic Feeder* berbasis *Internet of Things* yang terdapat pada gambar 4.1.



Gambar 4. 1 Diagram blok *Automatic Feeder*

Untuk instruksi perintah di Arduino menggunakan aplikasi Blynk yang di download di *Play Store*, setelah itu diproses oleh Arduino Mega 2560, selanjutnya untuk pengkoneksian antara Arduino dan *Automatic Feeder* menggunakan NodeMCU ESP8266.

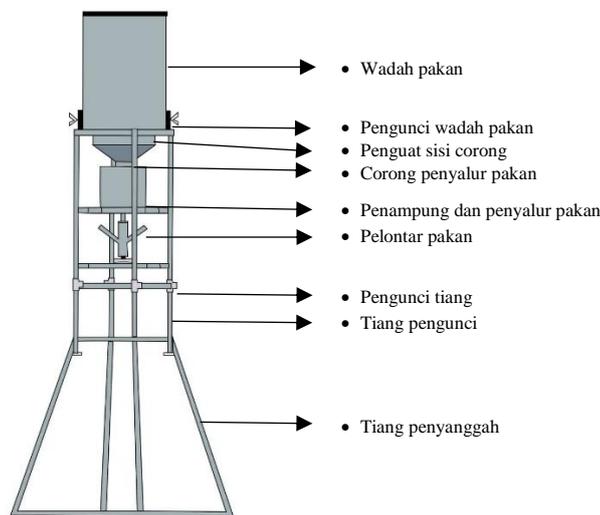
Power Supply pada rangkaian kontrol ini berfungsi sebagai penurun dan pengubah tegangan dari 220 volt AC menjadi 24 volt DC. Keluaran *power supply* 24 volt DC kemudian terhubung ke rangkaian *Buck Converter*. *Output Buck Converter* sebesar 12 volt masuk ke dalam rangkaian motor driver dan Arduino Mega 2560. Kemudian dari Arduino Mega 2560 terjadilah komunikasi dua arah atau proses pengirim dan penerima berupa notifikasi antara Arduino Mega 2560 dengan Android melalui Mode Wifi NodeMCU ESP8266. Tegangan *output* dari Arduino Mega 2560 sebesar 5V terhubung ke driver motor untuk mengaktifkan motor pengatur pakan dan motor pelontar pakan sesuai dengan waktu yang telah ditentukan. *Output 5V* Arduino juga masuk ke sensor ultrasonic, RTC dan LCD.

4.2 Sistem Mekanik *Automatic Feeder*

Untuk perancangan mekanik pada *Automatic Feeder* ini kami menggunakan *software Corel Draw X6*. Rancangan ini menggambarkan secara garis besar bagaimana kontruksi yang akan dibuat. Mekanisme kerja alat ini adalah saat *Automatic Feeder* telah diaktifkan pakan pada wadah akan mengalir ke penyalur

pakan melewati corong, dengan motor DC 12 volt penyalur pakan akan berputar ke bawah menyalurkan pakan menuju pelontar, setelah itu pelontar akan melontarkan pakan mulai dari titik terdekat alat yaitu 0.2 meter sampai dengan titik terjauh 15 meter. Untuk titik terjauh lontaran pakan dapat diatur oleh petambak itu sendiri dengan mengatur ketinggian pada tiang pengunci.

Perancangan sistem mekanik ini dimulai dari menentukan bagian-bagian pada alat serta menentukan bentuk yang sesuai dengan bagian alat yang ingin dibuat. Gambar 4.2 di bawah ini menunjukkan gambar secara keseluruhan sistem mekanik *Automatic Feeder* pada proyek akhir ini.



Gambar 4. 2 *Automatic Feeder* berbasis *Internet of Things*

Pembuatan sistem mekanik dilakukan dalam beberapa tahapan. Tahapan ini berfungsi sebagai acuan dalam proses pembuatan proyek akhir. Berikut adalah tahapan-tahapan dalam proses pembuatan mekanik *Automatic Feeder*.

1. Perancangan mekanik *Automatic Feeder*
2. Penentuan wadah penampung pakan
3. Pembuatan corong penyalur pakan
4. Pembuatan penampung dan penyalur pakan
5. Peletakan motor DC 12 volt
6. Pembuatan pelontar pakan

7. Peletakan motor DC 24 volt
8. Pembuatan penguat sisi keliling corong
9. Pembuatan pengunci wadah
10. Pembuatan tiang penyanggah dan tiang pengunci
11. Pembuatan pengaman motor DC
12. Perakitan bagian-bagian mekanik

Untuk penjelasan lebih lanjut tentang pembuatan mekanik *Automatic Feeder* dapat dilihat pada pembahasan di bawah ini:

4.2.1 Wadah Penampung Pakan

Wadah penampung pakan merupakan tempat untuk meletakkan pakan udang yang akan ditebar di kolam tambak. Untuk bahan material wadah penampung pakan ini menggunakan material plat dengan ketebalan 0.6 mm. Diameter wadah ini sebesar 50 cm dan ketinggian wadah 40 cm. Wadah ini dapat menampung pakan sebanyak 50 kg. Gambar 4.3 menunjukkan gambar wadah penampung pakan.



Gambar 4. 3 Wadah penampung pakan

4.2.2 Corong Penyalur Pakan

Corong penyalur pakan ini merupakan corong yang berfungsi sebagai tempat penyalur pakan setelah wadah. Pembuatan corong penyalur pakan ini meliputi pemilihan bahan material yang tepat dan sesuai. Oleh karena itu dipilihlah tutup dari dandang nasi yang berbentuk krucut dengan ukuran diameter 40 cm dan plat

dengan ketebalan 0.8 mm, yang kemudian digabungkan hingga menjadi corong penyalur pakan yang sesuai dengan kebutuhan yang diinginkan. Gambar 4.4 di bawah ini merupakan gambar dari corong penyalur pakan.



Gambar 4. 4 Corong penyalur pakan

Setelah corong selesai, langkah selanjutnya yaitu membuat penguat sisi keliling corong. Penguat ini menggunakan material berupa plat besi yang diletakkan di sekeliling bagian atas dan samping corong. Penguat ini berfungsi untuk melindungi corong agar tidak mudah penyok saat tertekan beban pakan. Proses pembuatan penguat ini menggunakan bahan material plat dengan ketebalan 2 mm. Gambar penguat pada sisi keliling corong dapat dilihat pada gambar 4.5 di bawah ini.



Gambar 4. 5 Penguat sisi corong

4.2.3 Penampung dan Penyalur Pakan

Penampung dan penyalur pakan merupakan bagian yang berfungsi sebagai penyalur pakan dari corong, kemudian menyalurkan pakan tersebut ke dalam pelontar pakan. Untuk itu kami menggunakan bahan material pipa PVC,

dikarenakan bahan ini terbilang ringan dan telah memiliki bentuk sebagaimana yang diinginkan. Kemudian di sisi pipa dibuat lubang dengan ukuran 11x2 cm yang berfungsi sebagai pengatur keluaran dan penampung pakan. Gambar 4.6 di bawah ini merupakan gambar penampung dan penyalur pakan.



Gambar 4. 6 Penampung dan penyalur pakan

4.2.4 Motor DC 12V

Motor DC 12 volt ini berfungsi sebagai penggerak penampung dan penyalur pakan, jadi pakan yang disalurkan dari corong akan masuk kedalam penampung pakan, setelah itu motor DC 12 volt ini akan memutar penampung pakan sehingga pakan di dalam penampung akan tersalurkan menuju pelontar pakan. Motor DC ini diletakkan di samping penampung pakan. Kecepatan putaran motor DC ini sebesar 100 RPM. Motor DC ini dapat menahan beban maksimal 3 kg. Gambar 4.7 di bawah ini merupakan penempatan motor DC 12 volt.



Gambar 4. 7 Peletakkan motor DC 12 volt

4.2.5 Pelontar Pakan

Pelontar pakan merupakan bagian alat yang berfungsi sebagai pelontar pakan udang. Proses pembuatan pelontar pakan ini terdiri dari bagian badan penampung pakan dan pelontar. Pada bagian badan penampung menggunakan bahan material penutup pipa PVC dengan ukuran diameter 3 cm dan tinggi 4 cm. Kemudian di sisi kiri dan kanan pipa dipasang pelontar dengan ukuran diameter awal 1.7 cm dan diameter ujung 1.4 cm dengan panjang nya 10.7 cm, Pemasangan mata pelontar ini dibuat miring sebesar 45° agar dapat membantu melontarkan pakan dengan jarak jauh. Pembuatan pelontar ini menggunakan bahan material *Stainless Steel 207*. Gambar 4.8 di bawah ini adalah gambar pelontar pakan.



Gambar 4. 8 Pelontar pakan

4.2.6 Motor DC 24 volt

Motor DC 24 volt pada proyek akhir ini berfungsi sebagai penggerak pelontar. Motor DC 24 volt ini diletakkan di bagian bawah pelontar pakan. Kecepatan motor DC ini diatur menggunakan PWM. Kecepatan motor dimulai dari lemah, sedang, hingga cepat dan begitupula sebaliknya. Hal ini bertujuan agar pakan yang dilontarkan terbagi secara merata. Gambar 4.9 di bawah ini adalah gambar peletakan motor DC 24 volt pada *Automatic Feeder*.



Gambar 4. 9 Penempatan motor DC 24 volt

4.2.7 Tiang Penyangga dan Tiang Pengunci

Tiang penyangga ini merupakan tiang yang berada paling bawah yang memiliki ketinggian 140 cm, sedangkan tiang pengunci merupakan tiang yang diletakkan di atas tiang penyangga. Pada bagian sisi atas tiang pengunci terdapat pengunci wadah. Pengunci wadah pakan adalah pengunci yang berfungsi sebagai penahan wadah, pengunci ini diletakkan pada sisi wadah, tujuan dibuatnya pengunci ini agar wadah tidak mudah goyang dan lebih tahan terhadap getaran yang ditimbulkan oleh motor pelontar. Pengunci ini bersifat tidak permanen atau tidak tetap. Gambar 4.10 di bawah ini merupakan gambar pengunci untuk wadah.



Gambar 4. 10 Pengunci wadah pakan

Sedangkan untuk proses pembuatan tiang penyangga ini terbagi dalam 3 bagian. Bagian paling bawah sebagai penyanggah menggunakan bahan material *Stainless Steel* dengan panjang 148.9 cm, bagian atas penyanggah dipasang tiang lagi sebagai penghubung ke tiang pengunci menggunakan bahan material *Stainless Steel* dengan diameter $1\frac{1}{4}$ dengan panjang nya 49 cm, kemudian bagian paling atas yang berfungsi sebagai tiang pengunci memiliki diameter 1 cm dengan panjang nya 98 cm.

Ketinggian tiang pengunci ini dapat menjadi tolak ukur dari jarak lontaran alat. Semakin tinggi alat maka semakin jauh lontaran pakan yang dikeluarkan oleh pelontar, begitu pula sebaliknya semakin pendek alat maka semakin pendek jarak lontaran pakan yang dikeluarkan oleh pelontar pakan, oleh karena itu *Automatic Feeder* ini dapat menyesuaikan bagaimana kebutuhan lontaran yang diinginkan

oleh tambak itu sendiri. Gambar 4.11 di bawah ini merupakan gambar dari tiang penyangga beserta tiang pengunci.



Gambar 4. 11 Tiang penyangga *Automatic Feeder*

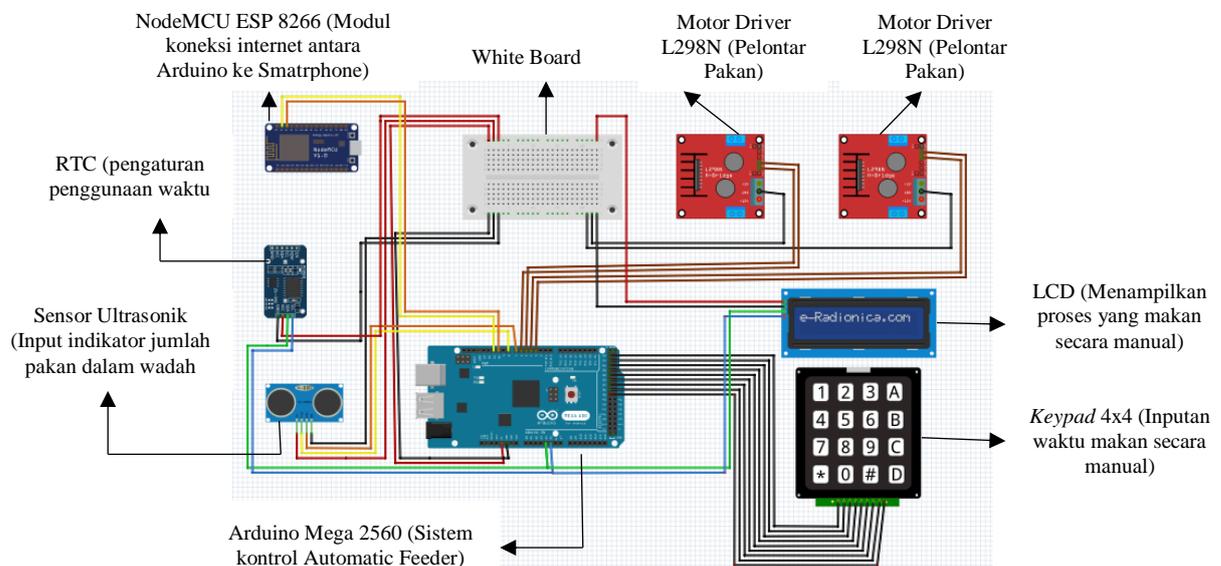
Setelah bagian-bagian pada alat mekanik telah selesai dibuat, maka langkah selanjutnya yaitu menggabungkan bagian mekanik tersebut. Pada tahap ini bagian-bagian yang telah dibuat di awal disatukan menjadi satu kesatuan. Gambar secara keseluruhan *Automatic Feeder* ini dapat dilihat pada gambar 4.12 berikut ini.



Gambar 4. 12 *Automatic Feeder* berbasis *Internet of Things*

4.3 Rangkaian Kontrol dan Komunikasi

Rangkaian kontrol mesin *Automatic Feeder* berbasis IoT ini dirangkai di dalam sebuah box panel dengan ukuran 30 x 40 cm. Di dalam rangkaian ini terdapat sebuah Arduino Mega untuk mengolah data nilai sensor, RTC, LCD, motor driver, sensor ultrasonic, *keypad* dan NodeMCU sebagai prantara komunikasi. Komponen-komponen tersebut disambungkan ke Arduino Mega 2560 sebagai sistem kontrol pada alat ini. Sehingga setiap komponen memiliki pin-pin tersendiri untuk disambungkan ke Arduino Mega 2560, pin-pin yang disambungkan ke Arduino ini harus sama ditulis persis seperti di dalam program Arduino. Jika pin di dalam program tidak sama antara sambungan komponen-komponen ke Arduino Mega 2560, maka alat juga tidak bisa berfungsi. Oleh karena itu diperlukannya ketelitian dan konsentrasi yang baik dalam perangkaian komponen serta pembuatan program Arduino untuk sistem kontrol *Automatic Feeder* ini. Untuk rangkaian kontrol dan komunikasi dapat dilihat pada gambar 4.13 di bawah ini.

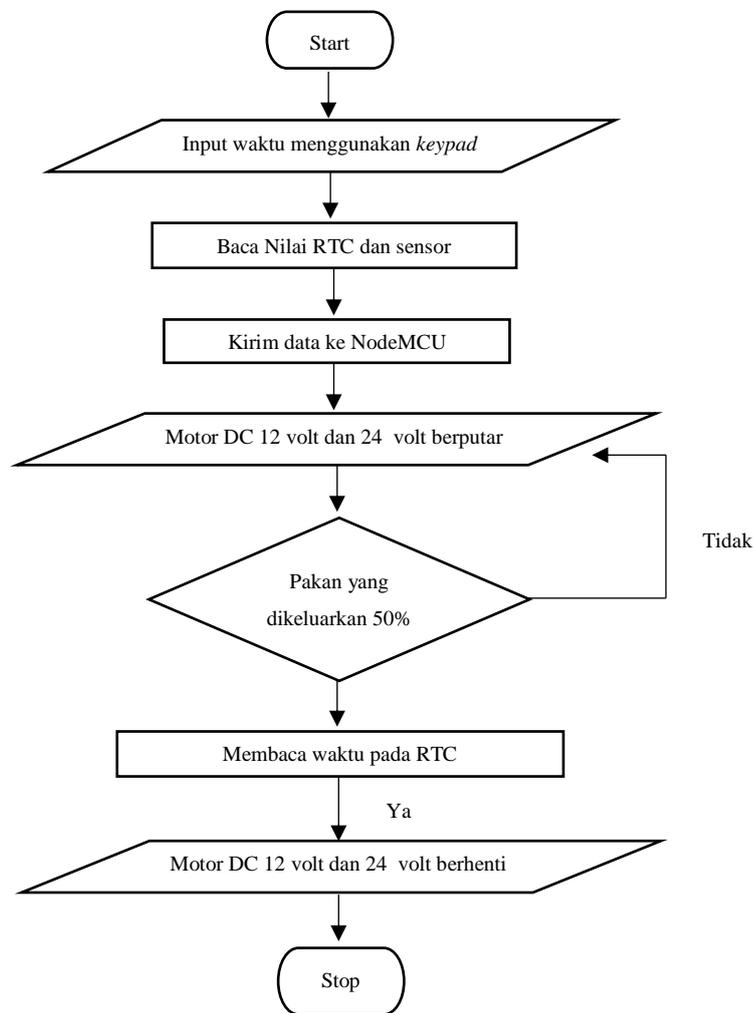


Gambar 4. 13 Rangkaian kontrol

Skematik di atas menjelaskan rangkaian kontrol *hardware* yang digunakan pada alat *Automatic Feeder* berbasis *Internet of Things*. Sistem kontrol disini menggunakan Arduino Mega 2560. Untuk penggerak motor DC 12 volt dan 24 volt

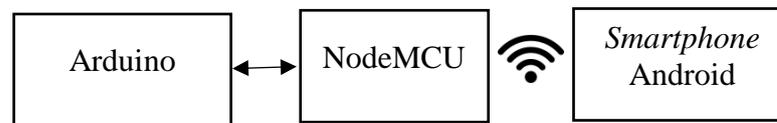
menggunakan motor Driver L298N, kemudian untuk koneksi antara Arduino dengan *Smartphone* digunakanlah sebuah modul wifi yaitu NodeMCU ESP 8266. Setelah itu ada sensor Ultrasonik yang berfungsi sebagai input untuk mengetahui sisa pakan yang masih ada di dalam wadah, kemudian untuk pengaturan waktu digunakanlah 1 buah RTC, dan untuk pengontrolan alat secara manual menggunakan *keypad* 4x4 yang ditampilkan langsung proses nya pada sebuah LCD yang berukuran 16x2.

Berikut merupakan *flowchart* pemrograman Arduino terdapat pada gambar 4.14 di bawah ini.



Gambar 4. 14 *Flowchart* sistem kontrol

Berikut merupakan blok diagram pemrograman NodeMCU terdapat pada gambar 4.15 berikut ini.



Gambar 4. 15 Blok diagram NodeMCU

Pada *flowchart* NodeMCU data didapatkan melalui pengiriman data serial dari Arduino. Kemudian jika NodeMCU terhubung dengan internet maka NodeMCU akan mengirim data ke Android yang akan tampil pada *software* yang telah didesain.

4.3.1 Program Penyalan motor DC 24V dan 12V

Pemrograman utama ini terdapat pada Arduino yaitu menangkap sensor ultrasonik, pembacaan RTC dan pengontrolan motor. Pembacaan nilai sensor ini digunakan sebagai media untuk melihat indikator jumlah pakan yang tersedia serta memenuhi kriteria untuk mematikan motor DC saat pemberian pakan telah terpenuhi. Dalam pemrograman ini pun terdapat fungsi lain seperti konversi nilai sensor ultrasonik ke persen dan lain-lain. Untuk pemrograman penyalan motor DC dapat dilihat pada lampiran.

Pada pemrograman penyalan motor DC di atas dapat dijelaskan bahwa waktu untuk pemberian pakan secara otomatis telah terprogram di dalam RTC. Sehingga secara otomatis motor DC akan aktif saat waktu yang telah ditentukan terbaca. Jika sensor ultrasonik telah menunjukkan sisa pakan tinggal 1%, maka secara otomatis motor DC akan berhenti berputar, dan akan kembali menyala saat waktu yang telah ditentukan kembali terbaca.

4.3.2 Program Pengiriman Data Serial Arduino Mega ke NodeMCU

Pemrograman ini digunakan untuk mengirimkan data yang akan dibaca oleh NodeMCU kemudian akan dikirimkan ke *Smartphone*. Program ini bertujuan untuk menampilkan nilai apa saja yang ingin ditampilkan. Kemudian dari program tersebut data yang dikirimkan adalah nilai dari sensor ultasonik yaitu jarak jumlah

pakan yang juga dikonversikan ke dalam bentuk persen. serta waktu dan tanggal yang terprogram oleh RTC. Untuk pemrograman Arduino secara keseluruhan maka dapat dengan menggabungkan program pengiriman data serial oleh Arduino berserta program penyalaan motor DC 24 volt dan 12 volt.

4.3.3 Program NodeMCU

Data yang dikirimkan oleh Arduino nantinya akan diterima oleh NodeMCU yang selanjutnya akan diolah oleh NodeMCU untuk dikirimkan ke aplikasi Blynk pada *Smartphone* agar dapat dimonitoring oleh user. Proses penerima data dilakukan oleh Arduino dapat dilihat di dalam lampiran.

Program tersebut merupakan program sampling penerimaan data serial dari Arduino. Saat data diterima oleh NodeMCU maka data tersebut akan dijadikan acuan dalam menentukan data yang akan dikirimkan ke *Smartphone*. Pada program juga kita harus mengisi token, dimana token ini didapatkan dari aplikasi Blynk yang dikirimkan ke Gmail yang kita daftar saat membuat akun di Blynk. Selain itu kita juga harus mengisi SSID serta *password* WIFI yang akan kita sambungkan ke NodeMCU.

4.3.4 Seting Aplikasi Monitoring pada *Smartphone* (Blynk)

Aplikasi Blynk ini dapat diunduh di *Play Store* yang ada pada *Smartphone*. Untuk menggunakan aplikasi ini user diwajibkan untuk mendaftar terlebih dahulu. Untuk pendaftaran disini lebih baik menggunakan email. Setelah itu user akan mendapatkan 2000 *power* secara gratis. *Power* ini berfungsi untuk membeli komponen-komponen yang akan digunakan. Pada *Automatic Feeder* ini komponen yang digunakan yaitu ultrasonik satu buah yang digunakan sebagai untuk menampilkan sisa pakan yang terdapat di dalam wadah, kemudian *push button* sebanyak dua buah yang berfungsi sebagai pengaturan otomatis atau manual alat serta untuk mengaktifkan dan mematikan *Automatic Feeder*. Terakhir yaitu terminal satu buah yang berfungsi untuk memonitoring waktu pemberian pakan, terminal ini akan menampilkan hari, jam, menit, serta detik kemudian saat

Automatic Feeder telah aktif maka terminal akan menampilkan tulisan “waktu makan”, hal ini bertujuan agar user mengetahui bahwa alat sedang dalam proses pemberian pakan. Berikut ini akan dibahas satu persatu tentang pengaturan komponen-komponen yang digunakan.

1. Seting komponen ultrasonik

Pada setingan komponen ultrasonik ini kita dapat mengisi identitas tittle dan input serta reading rate. Disini data yang dimonitor adalah jarak jumlah pakan yang terdapat di dalam wadah. Sehingga user dapat melihat sisa pakan yang terdapat dalam wadah tersebut. Untuk setingan komponen ultrasonik dapat dilihat pada gambar 4.16 di bawah ini.

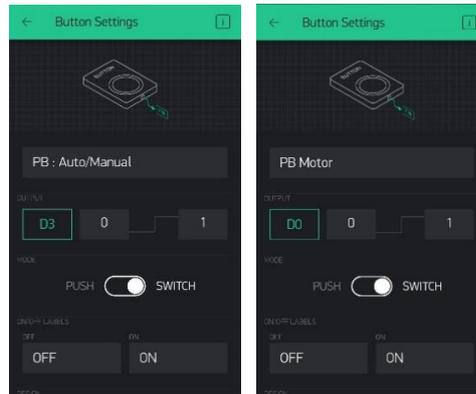


Gambar 4. 16 Pengaturan komponen ultrasonik

2. Seting komponen *Button*

Penggunaan *button* pada alat ini menggunakan 2 *push button*. *Push button* ini berfungsi untuk mrnghidupkan motor DC 12 volt dan 24 volt secara otomatis melalui *Smartphone*. *Push button* pertama untuk auto/manual dan *push button* kedua untuk penyalaan motor DC. Sehingga untuk penggunaannya kedua *push button* ini harus diaktifkan kedua-duanya agar motor DC dapat berfungsi, jika *push button* auto/manual dalam keadaan *OFF* maka *push button* motor tidak akan bisa

digunakan. Oleh karena itu *push button* auto/manual harus dalam keadaan *ON* agar *push button* motor dapat digunakan. Setingan *push button* dapat dilihat pada gambar 4.17 di bawah ini.



Gambar 4. 17 Pengaturan komponen *push button*

3. Setingan notifikasi

Untuk notifikasi yang digunakan pada Blynk ini, setingan nya tidak dirubah sama sekali. Karena untuk setingan nya telah tepat seperti apa yang diinginkan. Notifikasi ini berfungsi untuk menampilkan pemberitahuan dari *Automatic Feeder* berupa tulisan dan bunyi yang akan dengan otomatis muncul saat alat aktif dan saat alat telah selesai beroperasi Untuk gambar setingan notifikasi dapat dilihat dari gambar 4.18 di bawah ini.



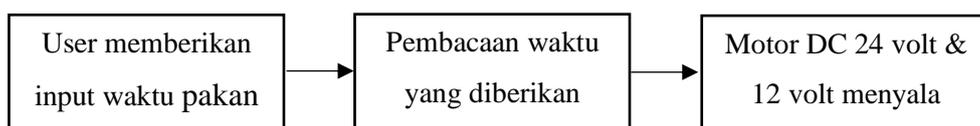
Gambar 4. 18 Pengaturan komponen notifikasi

4.4 Uji Coba *Hardware*

Pada bagian pengujian alat ini, dapat dilakukan hanya jika kontruksi nya telah selesai dibuat. Pengujian yang dilakukan adalah sebagai berikut.

4.4.1 Pengujian Motor DC 12 volt & 24 volt

Uji coba yang dilakukan pada motor DC 24 volt ini yaitu apakah motor dapat menyala saat telah diberikan tegangan dari power supply. Setelah itu saat telah dimasukkan ke dalam program, apakah motor dapat berputar seperti perintah yang telah terprogram. Kemudian saat waktu yang telah ditentukan tercapai, apakah motor akan berhenti. Kedua motor DC ini memiliki RPM yang berbeda. Motor DC 12 volt 100 RPM dan motor DC 24 volt sebesar 7000 RPM. Tahapan untuk uji coba motor DC ini dimulai dari pemberian waktu makan oleh user, saat waktu makan tercapai motor DC 24 volt akan berputar mulai dari kecepatan lemah sampai dengan paling cepat, saat putaran telah mencapai kecepatan maksimum maka putaran motor akan kembali melemah, siklus ini akan seperti ini terus menerus secara berulang-ulang, hal ini dikarenakan pengaturan PWM yang diatur di dalam program. Kemudian motor DC 12 volt akan mulai berputar saat motor DC 24 volt telah berputar sebanyak 3 siklus. Jika telah memenuhi kriteria yang diinginkan ini maka motor DC 12 dan 24 volt telah berfungsi dengan baik. Tahap terakhirnya yaitu saat waktu yang diberikan telah habis atau saat alat dimatikan secara otomatis menggunakan *Smartphone* maka motor DC 12 volt dan 24 volt akan berhenti secara bersamaan, jika motor DC berhenti secara bersamaan maka pengujian motor DC 12 volt dan 24 volt ini telah berhasil. Gambar 4.19 di bawah ini merupakan blok diagram uji coba penyalan motor DC 12 volt & 24 volt.



Gambar 4. 19 Blok diagram uji coba penyalan motor DC 24 volt dan 12 volt

4.4.2 Pengujian Jarak Lontaran Pakan

Uji coba jarak lontaran pakan ini bertujuan untuk mencapai jarak lontaran pakan maksimum. Karena semakin jauh jarak lontaran maksimum nya, maka semakin baik. Selain itu juga lontaran maksimum ini terdapat di dalam tuntutan proyek akhir ini. Untuk hasil pengujian jarak lontaran pakan dapat dilihat di tabel 4.1 berikut ini.

Tabel 4. 1 Jarak lontaran pakan

No	RPM	Tegangan	Ketinggian	Jarak Lontaran
1	400	17 V	10 cm	4 m
2	400	19 V	130 cm	5 m
3	7000	19 V	70 cm	7 m
4	7000	19 V	100 cm	9 m
5	7000	24 V	100 cm	12 m
6	7000	24 V	130 cm	13.5 m
7	7000	24 V	160 cm	15 m

4.4.3 Pengujian Keluaran Pakan

Keluaran pakan ini diuji untuk mengetahui apakah alat penampung dan penyalur pakan dapat berfungsi seperti apa yang diharapkan. Pengujian ini untuk melihat apakah pakan dapat tersalurkan menuju pelontar dengan lancar atau mampet pada satu keadaan. Hal lain yang dilihat dari pengujian ini yaitu untuk mengetahui apakah pakan udang saat keluar dari wadah ada yang berceceran/terjatuh ke tanah yang akan membuat alat ini menjadi tidak produktif.

Pada pengujian keluaran pakan ini, pengujian lain yang dilakukan yaitu penghitungan berapa banyak pakan yang dikeluarkan oleh motor pelontar pada satu waktu. Tujuan nya yaitu untuk mengetahui garis besar berapa lama waktu yang terpakai untuk menghabiskan pakan sebesar 25 kg (1 kali makan). Pengujian ini dilakukan beberapa kali sampai ditemukannya hasil akhir yang pas untuk keluaran pakan ini. Pakan yang digunakan untuk uji coba ini sebanyak 1 kg, pengujian yang dilakukan ini untuk mencari seberapa lama waktu yang diperlukan *Automatic*

feeder dalam menghabiskan pakan sebanyak 1 kg. Pengujian ini dilakukan dengan merubah program pada putaran motor DC 12 volt sebagai pemutar penyalur pakan. Sehingga untuk memperkecil banyak nya pakan yang dikeluarkan penyalur pakan harus lebih diperlambat putarannya. Untuk hasil pengujian keluaran pakan per menit dapat dilihat pada tabel 4.2 berikut ini.

Tabel 4. 2 Uji coba keluaran pakan per menit.

No	Banyak Pakan	Putaran penyalur pakan	Waktu yang dihabiskan
1	1 kg	Berputar mengikuti pelontar	1 menit
2	1 kg	Berputar setelah pelontar	2 menit
3	1 kg	Berputar setelah pelontar berputar dalam 2 fase	3 menit
4	1 kg	Berputar setelah pelontar berputar dalam 3 fase	5 menit
5	1 kg	Berputar setelah pelontar berputar dalam 3 fase	5 menit
6	1 kg	Berputar setelah pelontar berputar dalam 3 fase	5 menit

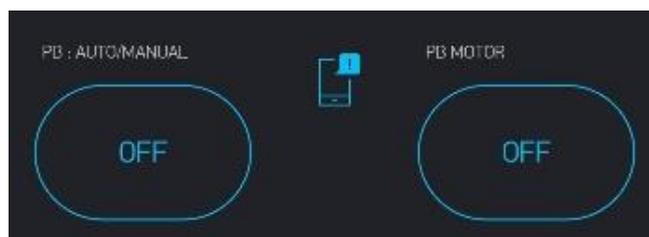
4.5 Uji Coba Software

Pengujian *software* ini dilakukan untuk mengetahui apakah alat yang telah dibuat sebelumnya dapat berfungsi jika telah digabungkan menggunakan sistem kontrol. Ada beberapa macam sistem kontrol yang terdapat pada alat ini. Sistem kontrol ini nantinya berfungsi sebagai kontrol untuk menggerakkan *Automatic Feeder*. Berikut adalah pengujian software pada *Automatic Feeder* berbasis *Internet of Things*.

4.5.1 Pengujian Koneksi ON/OFF Melalui Software Blynk

Agar *Automatic Feeder* dapat diaktifkan dari jarak jauh alat ini menggunakan NodeMCU ESP8266. NodeMCU disini berfungsi sebagai kontrol Arduino ke aplikasi Blynk. Koneksi antara Arduino dan NodeMCU ini akan tersambung jika

ada sebuah jaringan internet. Jaringan internet ini menggunakan sebuah modem merk BOLT dengan menggunakan kartu GSM Telkomsel. Untuk proses tahapan pengujian ini yang pertama kali dilakukan yaitu pengecekan apakah program NodeMCU telah memiliki SSID dan *password* yang sama dengan modem yang digunakan. Jika telah sama, maka secara otomatis Blynk akan terkoneksi terhadap NodeMCU. Langkah selanjutnya yaitu pada PB: Auto/Manual dibuat dalam keadaan *ON*, sama halnya dengan PB: Motor. Jika kedua PB ini telah dalam keadaan *ON* maka motor DC 12 volt dan 24 volt secara otomatis akan berputar. Jika kedua PB ini dibuat dalam keadaan *OFF* maka secara otomatis motor DC 12 volt dan 24 volt akan berhenti berputar. Program *ON/OFF* ini dapat dilihat pada lampiran. Untuk tampilan *ON/OFF* dapat dilihat pada gambar 4.20 berikut ini.

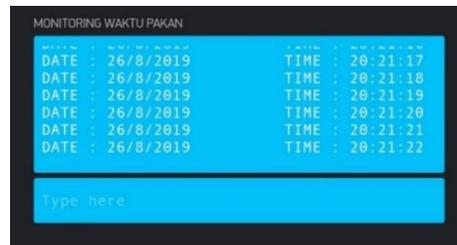


Gambar 4. 20 Tampilan *Push Button*

4.5.2 Pengujian Arduino, *Real Time Clock (RTC)* dan NodeMCU

Selain tampilan *ON/OFF*, RTC juga ditampilkan di aplikasi Blynk. Tampilan RTC ini berfungsi sebagai monitoring waktu pemberian pakan. Hal ini bertujuan agar si user dapat melihat kapan waktu pemberian pakan di tambak udang tersebut. Oleh karena itu harus adanya program pengkoneksian antara Arduino, RTC dan NodeMCU. Untuk tahapan proses pengujian ini yang pertama yaitu pengecekan apakah Blynk telah terkoneksi terhadap NodeMCU, jika telah terkoneksi secara otomatis maka pada wedget terminal dapat dilihat tampilan hari, tanggal, jam, menit, serta detik. Kemudian saat *Automatic Feeder* sedang dijalankan maka tampilan pada wedget terminal bertuliskan “waktu makan”, dan saat *Automatic Feeder* berhenti maka tampilan akan kembali seperti semula. RTC juga menampilkan hari, tanggal, jam, menit, serta detik pada LCD. Program ini dapat

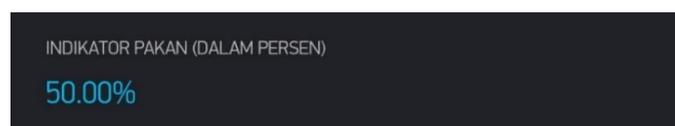
dilihat pada lampiran. Untuk gambar monitoring RTC dapat dilihat pada gambar 4.21 berikut ini.



Gambar 4. 21 Monitoring RTC pada Blynk

4.5.3 Pengujian Arduino, Sensor Ultrasonik dan NodeMCU

Sensor Ultrasonik ini merupakan salah satu tampilan di dalam Blynk yang digunakan untuk melihat indikator pakan di dalam wadah. Untuk menampilkan hasil dari sensor ultrasonik ke dalam Blynk diperlukan program pengkoneksian antara Arduino dan NodeMCU. Untuk tahapan pengujian ini yang pertama kali dilakukan yaitu pengecekan apakah Blynk telah terkoneksi terhadap NodeMCU, jika telah terkoneksi maka secara otomatis tampilan sensor ultrasonik akan menampilkan sisa pakan yang tersedia di dalam wadah dalam bentuk persen. Program ini dapat dilihat pada lampiran. Gambar 4.22 di bawah ini merupakan tampilan sensor Ultrasonik di dalam Blynk.



Gambar 4. 22 Indikator pakan menggunakan Ultrasonik

4.5.4 Koneksi Arduino, Keypad dan Liquid Crystal Display (LCD)

Keypad dan LCD ini berfungsi sebagai masukan waktu pemberian pakan secara manual dengan menggunakan Arduino. *Keypad* disini disediakan bagi user untuk memasukkan waktu pemberian pakan secara manual. Inputan waktu yang harus dimasukkan oleh user ini sebanyak 2 kali, yaitu waktu mulai pemberian pakan sampai dengan waktu berakhirnya pemberian pakan dilakukan. LCD yang digunakan ini berukuran 16x2, fungsinya yaitu untuk menampilkan inputan yang

dilakukan di dalam *keypad*. Gambar 4.23 di bawah ini menampilkan tampilan LCD setelah terprogram dan kondisi saat user ingin memasukkan waktu saat pemberian pakan.



Gambar 4. 23 Tampilan LCD dengan menggunakan *keypad*

Setelah tampilan awal pada LCD seperti gambar 4.24 di atas. Langkah selanjutnya yang dilakukan oleh user yaitu menekan tombol “C” pada *keypad*. Setelah tombol C ditekan tampilan LCD akan berubah menjadi menu jam dan menit, pada keadaan ini user dapat menginput waktu makan sesuai keinginan. Pada tahap ini user diminta untuk menginput waktu makan sebanyak 2 kali, yaitu waktu mulai makan dan waktu berhenti. Setelah user telah selesai menginput waktu, maka user harus menekan tombol “#” agar tampilan LCD kembali seperti semula. Gambar 4.24 di bawah ini merupakan tampilan LCD saat penginputan waktu oleh *keypad*.



Gambar 4. 24 Tampilan setingan waktu pada LCD

BAB V

KESIMPULAN DAN SARAN

Setelah dilakukannya pembuatan dan pengujian sampai selesainya alat ini dibuat, kesimpulan dan saran yang dapat diambil yaitu :

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari pembuatan Proyek Akhir ini adalah :

1. *Automatic Feeder* ini dapat dikendalikan dan dikontrol dari jarak jauh menggunakan *Smartphone* berbasis *Internet of Things*.
2. *Automatic Feeder* dapat melontarkan pakan dengan jarak sesuai keinginan pengguna sampai batas terjauh yaitu 15 meter.
3. Dengan menggunakan *Smartphone* user dapat melihat sisa pakan yang tersedia serta waktu pemberian pakan yang sedang berlangsung.

5.2 Saran

Jika Proyek Akhir ini akan dikembangkan saran yang diberikan yaitu :

1. Pembaharuan mekanik pada tiang penyanggah lebih dipendekkan, dan diganti dengan diperpanjang nya tiang pengunci pada alat. Hal ini bertujuan agar lebih mudah dalam proses pengangkutan alat.
2. Tampilan pada Blynk saat ini hanya dapat mengaktifkan, memonitor serta mengetahui jumlah pakan. Jika alat ini akan dikembangkan diharapkan alat dapat mengukur PH air pada tambak serta menampilkan alarm saat pakan telah habis
3. *Automatic Feeder* yang telah dibuat saat ini tidak bisa melontarkan pakan yang bersifat basah oleh karena itu jika alat ini dikembangkan alat dapat melontarkan pakan yang bersifat basah.

DAFTAR PUSTAKA

- [1] DJPB, "Penggunaan Automatic Feeder Tingkatkan Efisiensi Budidaya Udang Di Tambak," Rabu April 2017 : <http://djpb.go.id>. [Diakses Sabtu Maret 2019].
- [2] Growpd, "Pembesaran Udang Vaname Unit 13," PT Say Grup Indonesia, Selasa Desember 2017 : <https://www.growpal.co.id/>. [Diakses Senin April 2019].
- [3] E. A. Hendajat, "Budi Daya Udang Vaname (*Litopenaeus vaname*) Pola Tradisional Plus Di Kabupaten Sulawesi Selatan," *Media Akuakultur*, vol. 2, no. 11, p. 1, 2019.
- [4] Panca, "Mengintip Potensi Bisnis Udang Vaname," Rabu April 2019 : <https://www.google.com/amp/s/chnnel8.id>. [Diakses Senin Juni 2019].
- [5] TEK, "Automatic Feeder," Selasa September 2016 : <http://teknologi-kelautan.com/automatic-feeder-metrotv/>. [Dakses Selasa April 2019].
- [6] Azmiathallah, "Internet of Things and Possibilities," Wikipedia, Jumat November 2013. Available: <https://id.m.wikipedia.org>. [Diakses Senin Juni 2019].
- [7] C. N. Tanuwijaya, "Apakah Itu Internet of Things," Binus University, Kamis Maret 2018. Available: <https://sis.binus.ac.id>. [Diakses Jumat Juli 2019].
- [8] D. Wong, "Bagaimana cara kerja IoT," PT. Progresstek Abdibhakti Lokakarya, Selasa Maret 2016 : <https://www.progresstech.co.id>. [Dakses Senin Juni 2019].
- [9] efishery, "The Smartest Shrimp Autofeeder," PT. Multidaya Teknologi Nusantara, Rabu September 2017 : www.efishery.com. [Diakses Senin Juli 2019].

LAMPIRAN 1 : DAFTAR RIWAYAT HIDUP

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Muhammad Anzullah
Tempat dan tanggal lahir : Kemuja, 21 Januari 1998
Alamat Rumah : Desa Kemuja
Telp : -
Hp : 03175634477
Email : muhammadanzullah@gmail.com
Jenis Kelamin : Laki-laki.
Agama : Islam



2. Riwayat Pendidikan

SD N 8 Kemuja 2004-2009
SMP N 1 Mendo Barat 2009-2011
SMK N 2 Pangkalpinang 2011-2016

Pendidikan Non Formal

-

Sungailiat, 29 Agustus 2019

Muhammad Anzullah

DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Rachmah Suci Saputri
Tempat dan tanggal lahir : Belinyu, 18 Januari 1998
Alamat Rumah : Jln. Pahlawan XII Belinyu,
Bangka
Telp : -
Hp : 0822-8089-1622
Email : rachmahsucisaputri@gmail.com
Jenis Kelamin : Perempuan
Agama : Islam



2. Riwayat Pendidikan

SDI Belinyu 2004-2009
SMP N 1 Belinyu 2009-2011
SMK YPN Belinyu 2011-2016

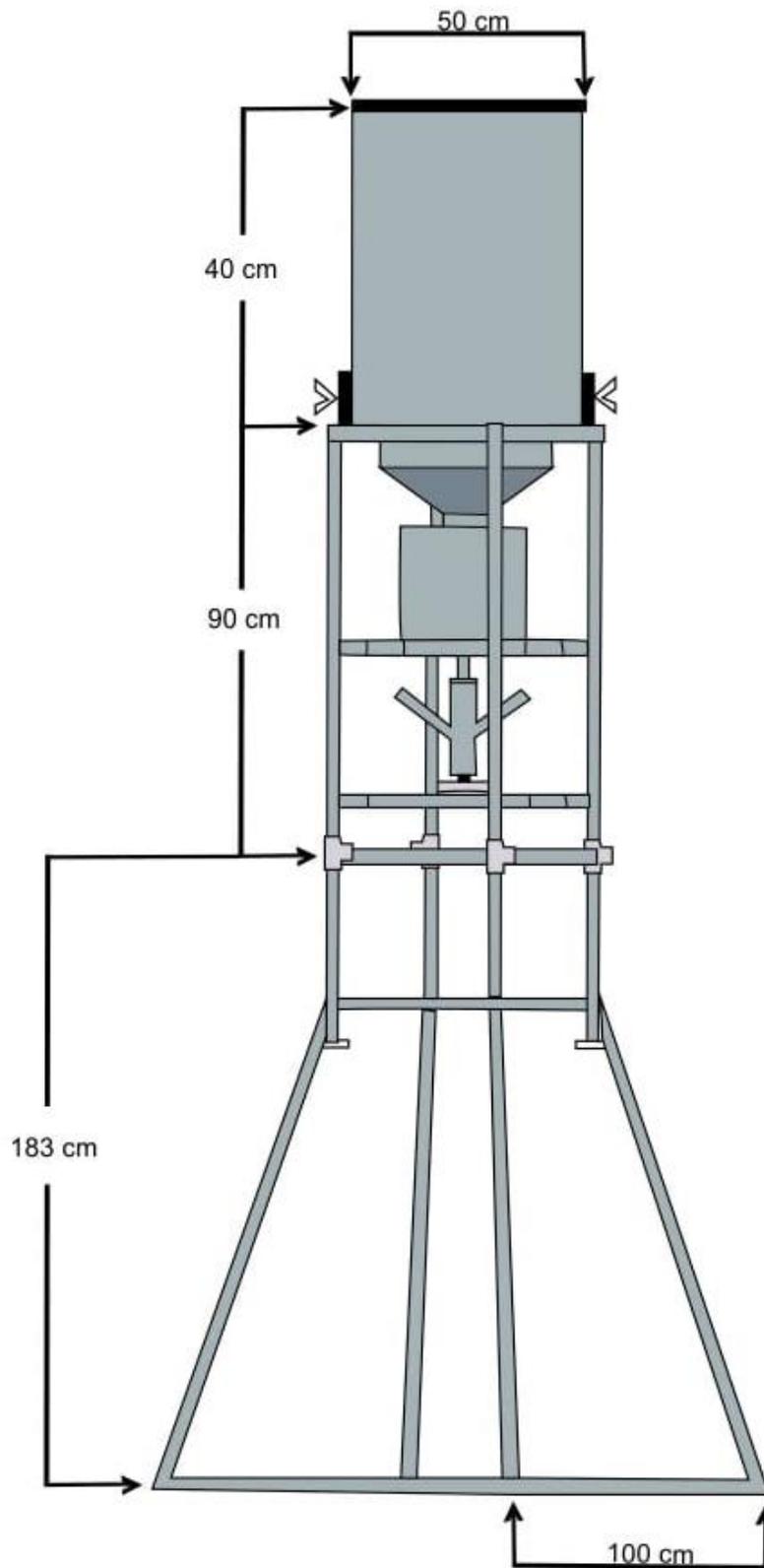
Pendidikan Non Formal

-

Sungailiat, 29 Agustus 2019

Rachmah Suci Saputri

LAMPIRAN 2 : DESAIN *AUTOMATIC FEEDER*



Automatic Feeder

Skala :

Digambar		
Diperiksa		
Judul		

POLMAN NEGERI BABEL

LAMPIRAN 3 : PROGRAM ARDUINO MEGA 2560

```

//keypad
#include<Keypad.h>
#include<LiquidCrystal_I2C.h>
#include<Wire.h>
#include <SoftwareSerial.h>
SoftwareSerial mySerial(10, 11);
#define echoPin 12 //Echo Pin
#define trigPin 13 //Trigger Pin
#define d0 8
#define d3 9

int maximumRange = 200; //kebutuhan akan maksimal range
int minimumRange = 00; //kebutuhan akan minimal range
long duration, distance; //waktu untuk kalkulasi jarakc
float persentase;
const byte ROWS = 4;
const byte COLS = 4;
char keys[ROWS][COLS] = {
  {'1', '2', '3', '+'},
  {'4', '5', '6', '-'},
  {'7', '8', '9', 'C'},
  {'*', '0', '=', '/'}}
};

byte rowPins[ROWS] = {22, 24, 26, 28};
byte colPins[COLS] = {30, 32, 34, 36};
// Created instances
Keypad myKeypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS,
COLS);
LiquidCrystal_I2C lcd(0x27, 16, 2);
boolean firstNumState = true;
String firstNum = "";
String secondNum = "";
int a=0,b=0,c=0,d=0,t=0;
float result = 0.0;
char operatr = ' ';
int z=0;
int k=0;
int l=0;
int g=0,h=0;
int keadaan=0;
int m,n;

//mendefinisikan pin yang digunakan untuk control pin
int IN_1 = 5;
int IN_2 = 6;

```

```

int IN_3 = 3;
int IN_4 = 4;
int p=0,q=0;

// Date and time functions using a DS3231 RTC connected via I2C and Wire lib
#include "RTClib.h"
RTC_DS3231 rtc;
const int relay=1;
int x=0,y=0,w=0,r,s;
unsigned long waktu;
char daysOfTheWeek[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis",
"Jumat", "Sabtu"};
void ultrasonik()
{
    digitalWrite(trigPin, LOW);delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);

    //perhitungan untuk dijadikan jarak
    distance = duration/58.2;
    persentase=map(distance,26,65,50,0);
    Serial.print(persentase);
    Serial.print("%");
    Serial.print("\t");
    Serial.println(distance);
    r=(distance-65)*(-1);
    s=map(r,0,39,0,50);
    Serial.println(s);

}

void setup () {
mySerial.begin(9600);
    //motor
    pinMode(IN_1, OUTPUT);
    pinMode(IN_2, OUTPUT);
    pinMode(IN_3, OUTPUT);
    pinMode(IN_4, OUTPUT);
    pinMode(relay,OUTPUT);
    pinMode(42, OUTPUT);
    pinMode(44, OUTPUT);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(d3,INPUT );
    pinMode(d0, INPUT);

```

```

#ifndef ESP8266
  while (!Serial); // for Leonardo/Micro/Zero
#endif
lcd.init();
  // Print a message to the LCD.
  lcd.backlight();
  Serial.begin(9600);
  lcd.setCursor(0, 0);
  lcd.print("Proyek Akhir");
  lcd.setCursor(0, 1);
  lcd.print("2019 Bisa");
  delay(1000);
// rtc.begin();
if (! rtc.begin()) {
  Serial.println("Couldn't find RTC");
  while (1);
  lcd.println("Couldn't find RTC");
  while (1);
}

if (rtc.lostPower()) {
  Serial.println("RTC lost power, lets set the time!");
  lcd.println("RTC lost power, lets set the time!");
  // following line sets the RTC to the date & time this sketch was compiled
  rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  // This line sets the RTC with an explicit date & time, for example to set
  // January 21, 2014 at 3am you would call:
  // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
}
clr();
lcd.clear();
}
void rtcku()
{
  DateTime now = rtc.now();
  Serial.print(now.year(), DEC);
  Serial.print('/');
  Serial.print(now.month(), DEC);
  Serial.print('/');
  Serial.print(now.day(), DEC);
  Serial.print(" ");
  Serial.print(daysOfTheWeek[now.dayOfTheWeek()]);
  Serial.print(" ");
  Serial.print(now.hour(), DEC);
  Serial.print(':');
  Serial.print(now.minute(), DEC);

```

```

Serial.print(':');
Serial.print(now.second(), DEC);
Serial.println();
lcd.clear();
lcd.print(now.year(), DEC);
lcd.print('/');
lcd.print(now.month(), DEC);
lcd.print('/');
lcd.print(now.day(), DEC);
lcd.setCursor(0,1);
lcd.print(daysOfTheWeek[now.dayOfTheWeek()]);
lcd.print(" ");
lcd.print(now.hour(), DEC);
lcd.print(':');
lcd.print(now.minute(), DEC);
lcd.print(':');
lcd.print(now.second(), DEC);

}
void loop () {
while(keadaan==5){
int Auto=digitalRead(d3);
if(Auto==1){
keadaan=5;
int pb_motor=digitalRead(d0);
if(pb_motor==1){
Serial.println("motor hidup");
}else{
Serial.println("motor mati");
}
}else{
keadaan=0;
}
}
}

DateTime now = rtc.now();
Serial.print(now.year(), DEC);
Serial.print("/");
Serial.print(now.month(), DEC);
Serial.print("/");
Serial.print(now.day(), DEC);
Serial.print(" ");
Serial.print(daysOfTheWeek[now.dayOfTheWeek()]);
Serial.print(" ");
Serial.print(now.hour(), DEC);
Serial.print(':');

```

```

Serial.print(now.minute(), DEC);
Serial.print(':');
Serial.print(now.second(), DEC);
Serial.println();
ultrasonik();
mySerial.print (now.year(),DEC);
mySerial.print ("a");
mySerial.print (now.month(),DEC);
mySerial.print ("b");
mySerial.print (now.day(),DEC);
mySerial.print ("c");
mySerial.print (now.hour(), DEC);
mySerial.print ("d");
mySerial.print (now.minute(), DEC);
mySerial.print ("e");
mySerial.print (now.second(), DEC);
mySerial.print ("f");
mySerial.print(s); // Menampilkan nilai jarak yang akan dikirim
mySerial.print("g");
while(keadaan==0){
  int Auto=digitalRead(d3);
  if(Auto==1){
    keadaan=5;
    int pb_motor=digitalRead(d0);
    if(pb_motor==1){
      Serial.print("motor hidup");
    }else{
      Serial.print("mootor mati");
    }
  }else{
    keadaan=0;
  }
}
rtcku();
DateTime now = rtc.now();
ultrasonik();
mySerial.print (now.year(),DEC);
mySerial.print ("a");
mySerial.print (now.month(),DEC);
mySerial.print ("b");
mySerial.print (now.day(),DEC);
mySerial.print ("c");
mySerial.print (now.hour(), DEC);
mySerial.print ("d");
mySerial.print (now.minute(), DEC);
mySerial.print ("e");
mySerial.print (now.second(), DEC);

```

```

mySerial.print ("f");
mySerial.print(s); // Menampilkan nilai jarak yang akan dikirim
mySerial.print("g");

tombol();
Serial.print(a);
Serial.print(b);
Serial.print(c);
Serial.println(d);
Serial.print(keadaan);
delay (1000);
}
while(keadaan==1){
  tombol();
  Serial.print(a);
  Serial.print(b);
  Serial.print(c);
  Serial.println(d);
  Serial.print(keadaan);
  DateTime now = rtc.now();
  ultrasonik();
  mySerial.print (now.year(),DEC);
  mySerial.print ("a");
  mySerial.print (now.month(),DEC);
  mySerial.print ("b");
  mySerial.print (now.day(),DEC);
  mySerial.print ("c");
  mySerial.print (now.hour(), DEC);
  mySerial.print ("d");
  mySerial.print (now.minute(), DEC);
  mySerial.print ("e");
  mySerial.print (now.second(), DEC);
  mySerial.print ("f");
  mySerial.print(s); // Menampilkan nilai jarak yang akan dikirim
  mySerial.print("g");
  //rtcku();
  if(t==1){
    keadaan=2;
  }
  delay(1000);
}
while(keadaan==2){
  t=0;
  DateTime now = rtc.now();
  if(a==now.hour() && b==now.minute()){
    keadaan=3;
  }
}

```

```

}
rtcku();
Serial.print(keadaan);
ultrasonik();
mySerial.print (now.year(),DEC);
mySerial.print ("a");
mySerial.print (now.month(),DEC);
mySerial.print ("b");
mySerial.print (now.day(),DEC);
mySerial.print ("c");
mySerial.print (now.hour(), DEC);
mySerial.print ("d");
mySerial.print (now.minute(), DEC);
mySerial.print ("e");
mySerial.print (now.second(), DEC);
mySerial.print ("f");
mySerial.print(s); // Menampilkan nilai jarak yang akan dikirim
mySerial.print("g");
delay(1000);
}
while(keadaan==3){
    ultrasonik();
    mySerial.print ("h");
    mySerial.print(s); // Menampilkan nilai jarak yang akan dikirim
    mySerial.print("g");
    digitalWrite(44,HIGH);
    digitalWrite(42,LOW);
    Serial.print(keadaan);
    motor();
    //k=b+5;
    //if(k=now.minute())
    //{
    //keadaan=4;
    //}
}

while(keadaan==4){
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("istirahat");
    digitalWrite(44,LOW);
    digitalWrite(42,HIGH);
    DateTime now = rtc.now();
    ultrasonik();
    mySerial.print ("i");
    mySerial.print(s); // Menampilkan nilai jarak yang akan dikirim

```

```

mySerial.print("g");
n=((now.hour()-a)*60)+(now.minute()-b);
Serial.println(n);
m=n%8;
Serial.println(m);
if(m==5){
keadaan=4;
}else{
keadaan=3;
}

if(c==now.hour() && d==now.minute()){
digitalWrite(44,LOW);
digitalWrite(42,HIGH);
analogWrite(IN_1,0);
analogWrite(IN_2,0);
analogWrite(IN_3,0);
analogWrite(IN_4,0);
keadaan=0;
}

//motorlontar();
analogWrite(IN_1,0);
analogWrite(IN_2,0);
analogWrite(IN_3,0);
analogWrite(IN_4,0);
delay(100);
}
}

void tombol()
{
char newKey = myKeypad.getKey();
if (newKey != NO_KEY && (newKey == '1' || newKey == '2' || newKey == '3' ||
newKey == '4' || newKey == '5' || newKey == '6' || newKey == '7' || newKey == '8'
|| newKey == '9' || newKey == '0'))
{
if (firstNumState == true)
{
firstNum = firstNum + newKey;

lcd.print(newKey);
}
else
{
secondNum = secondNum + newKey;
}
}
}

```

```

        lcd.print(newKey);
    }
}
if (newKey != NO_KEY && newKey == '/')
{
    if (firstNumState == true)
    {
        operatr = newKey;
        firstNumState = false;
        lcd.setCursor(15, 0);
        lcd.print(operatr);
        lcd.setCursor(8, 1);
    }
}

```

```

if (newKey != NO_KEY && newKey == '=')
{
    z++;
    if(z==1)
    {
        a=firstNum.toInt();
        b=secondNum.toInt();
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("jam  : ");
        lcd.setCursor(12, 0);
        lcd.print("op ");
        lcd.setCursor(0, 1);
        lcd.print("menit : ");
        lcd.setCursor(8, 0);
        firstNumState = true;
        firstNum = "";
        secondNum = "";
    }
    if(z==2)
    {
        a=a;
        b=b;
        c=firstNum.toInt();
        d=secondNum.toInt();
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Awal = ");
        lcd.print(a);
        lcd.print(":");
    }
}

```

```

    lcd.print(b);
    lcd.setCursor(0, 1);
    lcd.print("Aakhir = ");
    lcd.print(c);
    lcd.print(":");
    lcd.print(d);
}
}

if (newKey != NO_KEY && newKey == 'C')
{

    keadaan=1;
    clr();
}
if (newKey != NO_KEY && newKey == '+')
{
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("selesai");

    t=1;
}
}
void clr()
{
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("jam  :");
    lcd.setCursor(12, 0);
    lcd.print("op ");
    lcd.setCursor(0, 1);
    lcd.print("menit :");
    lcd.setCursor(8, 0);
    firstNum = "";
    secondNum = "";
    operatr = '';
    result=0.0;
    z=0;
    a=0;b=0;c=0;d=0;
}

void motor()
{
    for(p=0; p<=250; p=p+10)
    {

```

```

Serial.print(keadaan);
DateTime now = rtc.now();
n=((now.hour()-a)*60)+(now.minute()-b);
Serial.println(n);
m=n%8;
Serial.println(m);
if(m==5){
keadaan=4;
Serial.print(keadaan);
DateTime now = rtc.now();
n=((now.hour()-a)*60)+(now.minute()-b);
Serial.println(n);
m=n%8;
Serial.println(m);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("istirahat");
analogWrite(IN_1,0);
analogWrite(IN_2,0);
analogWrite(IN_3,0);
analogWrite(IN_4,0);
ultrasonik();
mySerial.print ("i");
mySerial.print(s); // Menampilkan nilai jarak yang akan dikirim
mySerial.print("g");
delay(100);
q=255;
}else{
keadaan=3;
}

//motorlontar();
//Putar Mesin searah jarum jam
if(c==now.hour() && d==now.minute()){
digitalWrite(44,LOW);
digitalWrite(42,HIGH);
analogWrite(IN_1,0);
analogWrite(IN_2,0);
analogWrite(IN_3,0);
analogWrite(IN_4,0);
keadaan=0;
}
if(m==0||m==1||m==2||m==3||m==4){
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("waktu makan");
}

```

```

//sensorpakan();
analogWrite(IN_1,0);
analogWrite(IN_2,p);
analogWrite(IN_3,0);
analogWrite(IN_4,0);
h++;
if(h<=75 && h>50)
{
  analogWrite(IN_3,(250-p));
  analogWrite(IN_4,0);
  if(h>=75)
  {
    h=0;
  }
}
delay(100);
if(p==255)
{
  q=255;
}
}
}

for(q=250; q>=0; q=q-10)
{
  Serial.print(keadaan);
  DateTime now = rtc.now();
  n=((now.hour()-a)*60)+(now.minute()-b);
  Serial.println(n);
  m=n%8;
  Serial.println(m);
  if(m==5){
  keadaan=4;
  Serial.print(keadaan);
  DateTime now = rtc.now();
  n=((now.hour()-a)*60)+(now.minute()-b);
  Serial.println(n);
  m=n%8;
  Serial.println(m);
  lcd.clear();
  ultrasonik();
  mySerial.print ("i");
  mySerial.print(s); // Menampilkan nilai jarak yang akan dikirim
  mySerial.print("g");
  lcd.setCursor(0, 0);
  lcd.print("istirahat");
}
}
}

```

```

analogWrite(IN_1,0);
analogWrite(IN_2,0);
analogWrite(IN_3,0);
analogWrite(IN_4,0);
delay(100);
p=0;
}else{
    keadaan=3;
}
if(c==now.hour() && d==now.minute()){
digitalWrite(44,LOW);
digitalWrite(42,HIGH);
analogWrite(IN_1,0);
analogWrite(IN_2,0);
analogWrite(IN_3,0);
analogWrite(IN_4,0);
keadaan=0;
}
if(m==0||m==1||m==2||m==3||m==4){
    lcd.clear();
    lcd.setCursor(0, 0);
//    motorlontar();
    lcd.print("waktu makan");
//    sensorpakan();
    analogWrite(IN_1,0);
    analogWrite(IN_2,q);
    analogWrite(IN_3,0);
    analogWrite(IN_4,0);
    g++;
    if(g<=75 && g>50)
    {
        analogWrite(IN_3,(q-250));
        analogWrite(IN_4,0);
        if(g>=75)
        {
            g=0;
        }
    }
    delay(100);
    if(q==0)
    {
        p=0;
    }
}
}
}

```

LAMPIRAN 4 : PROGRAM NodeMCU

```

#include <BlynkSimpleEsp8266.h>=
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
int x=0,y=0;
BlynkTimer timer;
#include <SoftwareSerial.h>
SoftwareSerial nodemcu(D5, D6); // RX, TX
float a,ult, ultrasonik;
int tombol=0;
int thn,bln,hr,jm,mnt,dtk,tahun, bulan, hari,jam, menit, detik;
// Go to the Project Settings (nut icon).
char auth[] = "TzKzIHTqywy2F80iqOGL_me_eTgg_J6M";
// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "BOLTSuper4G-C871";
char pass[] = "9rifm2mg";
WidgetTerminal terminal (V2);
// BLYNK_WRITE(V0);
/*{
  int pb = param.asInt();
  if(pb==HIGH)
  {
    tombol=1;
    Serial.println(tombol);//
  }
  else{ tombol=0; Serial.println(tombol); }
}*/
void notify1()
{
  int rahat=digitalRead(5);
  if(rahat==0 && x==0){
    Blynk.notify("istirahat");
    Blynk.virtualWrite(V2,"istirahat");
    x=1;
  }
  if(rahat==1){
    x=0;
  }
  int makan=digitalRead(4);
  if(makan==0 && y==0){
    Blynk.notify("waktu makan");
    Blynk.virtualWrite(V2,"waktu makan");
    y=1;
  }
  if(makan==1){
    y=0;
  }
}

```

```

}
}
void setup()
{
  pinMode(5,INPUT_PULLUP);
  pinMode(4,INPUT_PULLUP);
  timer.setInterval(100L, notify1);
  Serial.println("aa");
  // Open serial communications and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo only
  }
  // ultrasonik = nodemcu.parseFloat();
  // Serial.println(ultrasonik);
  // set the data rate for the SoftwareSerial port
  nodemcu.begin(9600);
  Blynk.begin (auth, ssid, pass);
}
void loop()
{
  Blynk.run();
  timer.run();
  nodemcu.print(tombol);
  nodemcu.print('k');
  while (nodemcu.available()>0)
  {
    thn=nodemcu.parseFloat();
    if(nodemcu.read()=='a'){
      tahun=thn;
    }
    bln=nodemcu.parseFloat();
    if(nodemcu.read()=='b'){
      bulan=bln;
    }
    hr=nodemcu.parseFloat();
    if(nodemcu.read()=='c'){
      hari=hr;
    }
    jm=nodemcu.parseFloat();
    if(nodemcu.read()=='d'){
      jam=jm;
    }
    mnt=nodemcu.parseFloat();
    if(nodemcu.read()=='e'){
      menit=mnt;
    }
  }
}

```

```

}
dtk=nodemcu.parseFloat();
if(nodemcu.read()=='f'){
    detik=dtk;
}
ult=nodemcu.parseFloat();
if(nodemcu.read()=='g'){
    ultrasonik=ult;
}
String currentTime = String("TIME : ") + jam + ":" + menit + ":" + detik + "\n";
String currentDate = String("DATE : ") + hari + "/" + bulan + "/" + tahun+";
String currentultrasonik = String(ultrasonik) + "%";
Serial.print("date:");
Serial.print(tahun);
Serial.print("/");
Serial.print(bulan);
Serial.print("/");
Serial.print(hari);
Serial.print("\n");
Serial.print("time:");
Serial.print(jam);
Serial.print(":");
Serial.print(menit);
Serial.print(":");
Serial.print(detik);
Serial.print("\n");
Serial.print("ultrasonik: ");
Serial.print(ultrasonik);
Serial.print("\n");
Blynk.virtualWrite(V0, currentultrasonik);
Blynk.virtualWrite(V2, currentDate);
Blynk.virtualWrite(V2, currentTime);
}
}

```

LAMPIRAN 5 : Standar Operasional Prosedur

Standard Operation Procedure (SOP)

Penggunaan Automatic Feeder

A. Penggunaan Secara Otomatis

1. Download aplikasi “Blynk” di Play Store pada Smartphone Android
2. Daftar akun pada Blynk dengan menggunakan email.
3. Buat projek baru pada Blynk, tulis nama projek sesuai user inginkan.
4. Tambahkan penggunaan komponen pada Widget Box.

Berikut adalah komponen-komponen yang digunakan pada Automatic Feeder :

a) Button 1 :

- Berikan nama pada Push B (PB : Auto/Manual)
- Output gunakan pin digital 3 (D3 PWM)
- Mode : SWITCH
- Pada label off ketikkan : OFF
- Pada label on ketikkan : ON
- Pemilihan warna dan ukuran teks dapat menyesuaikan pada keinginan user.

b) Button 2 :

- Berikan nama pada PB (PB : Motor)
- Output gunakan pin digital 0 (D0 PWM)
- Mode : SWITCH
- Pada label off ketikkan : OFF
- Pada label on ketikkan : ON
- Pemilihan warna dan ukuran teks dapat menyesuaikan pada keinginan user.

c) Notification

- Notify When Hardware Goes Offline : OFF
- Priority : Normal

- Pemilihan warna notifikasi dapat menyesuaikan pada keinginan user.

d) Value Display Settings :

- Pada title (optional) ketikkan : Indikator Pakan (Dalam Persentase)
- Input gunakan pin virtual 0 (V0)
- Reading Rate : PUSH
- Pemilihan warna dan ukuran teks dapat menyesuaikan pada keinginan user.

e) Terminal Settings

- Pada kolom terminal ketikkan : Monitoring Waktu Pakan
- Input gunakan pin virtual 2 (V2)
- Add Newline : NO
- Input Line : ON
- Autoscroll : ON
- Pemilihan warna dan ukuran teks dapat menyesuaikan pada keinginan user.

5. Instal software Arduino pada laptop/Notebook.
6. Buka program Arduino dan program NodeMCU
7. Isikan token yang dikirimkan Blynk ke email pada program NodeMCU dibagian “char auth”.
Contoh : TzKzIHTqywy2F80iqOGL_me_eTgg_J6M.
8. Sambungkan Arduino Mega 2560 ke laptop/Notebook menggunakan kabel donloader Arduino.
9. Sambungkan NodeMCU ke laptop/Notebook menggunakan kabel USB.
10. Pastikan modem data pada Box Panel hidup.
11. Klik “Tools” di dalam program Arduino :
 - Pada Board pilih : NodeMCU 1.0 (ESP-12E Module)
 - Pada Port : Pilih sesuai COM yang digunakan pada laptop/Notebook
12. Upload program NodeMCU di Software Arduino.

13. Buka program Arduino
14. Klik “Tools” di dalam program Arduino :
 - Pada Board pilih : Arduino/Genuino Mega or Mega 2560
 - Pada Port : Pilih sesuai COM yang digunakan pada laptop/Notebook
15. Upload program Arduino
16. Pastikan Blynk pada Smartphone telah terkoneksi. Contoh : “Online since 10:00 Aug 20, 2019”.
17. Kelik PB : Auto/Manual dari keadaan OFF menjadi ON
18. Kelik PB : Motor dari keadaan OFF menjadi ON
19. Jika waktu makan telah tercapai, klik tombol PB : Motor kembali keadaan OFF.

B. Penggunaan Secara Manual

1. Buka Program Arduino
2. Sambungkan Arduino Mega 2560 ke laptop/Notebook menggunakan kabel donloader Arduino.
3. Klik “Tools” di dalam program Arduino :
 - Pada Board pilih : Arduino/Genuino Mega or Mega 2560
 - Pada Port : Pilih sesuai COM yang digunakan pada laptop/Notebook
4. Upload program Arduino.
5. Pastikan LCD menyala
6. Tekan tombol “C” agar tampilan LCD menunjukkan tampilan menu jam dan menit.
7. Masukkan jam pemberian pakan yang kita inginkan.
8. Tekan tombol “D” agar kursor berpindah ke menit.
9. Masukkan waktu menit yang diinginkan.
10. Tekan tombol “#” agar tampilan LCD menunjukkan tampilan menu jam dan menit kembali.
11. Ulangi seperti proses 7-9.
12. Tekan tombol “A” agar tampilan LCD kembali seperti tampilan awal.