Perancangan Sistem *Monitoring* dan Pengendali pada Mesin Pembuat Pupuk Cair Berbasis NodeMCU dan Android

SKRIPSI

untuk memenuhi salah satu persyaratan
mencapai derajat Sarjana S1

Disusun oleh:

Arigo Fahreza Putratama

16524021

Jurusan Teknik Elektro

Fakultas Teknologi Industri

Universitas Islam Indonesia

Yogyakarta

2020

LEMBAR PENGESAHAN

Perancangan Sistem *Monitoring* dan Pengendali pada Mesin

Pembuat Pupuk Cair Berbasis NodeMCU dan Android

TUGAS AKHIR

Diajukan sebagai Salah Satu Syarat untuk Memperoleh
Gelar Sarjana Teknik
pada Program Studi Teknik Elektro
Fakultas Teknologi Industri
Universitas Islam Indonesia

Disusun oleh:

Arigo Fahreza P
16524021

Yogyakarta, 18 Juli 2020

Menyetujui,

Pembimbing 1



Dwi Ana Ratna Wati, S.T.,M.Eng.
035240102

LEMBAR PENGESAHAN

SKRIPSI

Perancangan Sistem *Monitoring* dan Pengendali pada Mesin Pembuat Pupuk Cair Berbasis NodeMCU dan Android

Dipersiapkan dan disusun oleh:

Arigo Fahreza Putratama

16524021

Telah dipertahankan di depan dewan penguji

Pada tanggal: 13 Agustus 2020

Susunan dewan penguji

Ketua Penguji : Dwi Ana Ratna Wati, S.T., M.Eng. _____

Anggota Penguji 1: Alvin Sahroni, S.T., M.Eng., Ph.D. _____

Anggota Penguji 2: Elvira Sukma Wahyuni, S.Pd., M.Eng. _____

Skripsi ini telah diterima sebagai salah satu persyaratan untuk memperoleh gelar Sarjana

Tanggal: 13 Agustus 2020

Ketua Program Studi Teknik Elektro

Yusuf Aziz Amrullah, S.T., M.Sc., Ph.D.

045240101

PERNYATAAN

Dengan ini Saya menyatakan bahwa:

1. Skripsi ini tidak mengandung karya yang diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan Saya juga tidak mengandung karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.
2. Informasi dan materi Skripsi yang terkait hak milik, hak intelektual, dan paten merupakan milik bersama antara tiga pihak yaitu penulis, dosen pembimbing, dan Universitas Islam Indonesia. Dalam hal penggunaan informasi dan materi Skripsi terkait paten maka akan diskusikan lebih lanjut untuk mendapatkan persetujuan dari ketiga pihak tersebut diatas.

Yogyakarta, 18 Juli 2020



Arigo Fahreza P

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Assalamu'alaikum Wr.Wb

Puji syukur penulis panjatkan kehadirat Allah SWT, yang telah memberikan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan Laporan Skripsi dengan judul "Perancangan Sistem *Monitoring* dan Pengendali pada Mesin Pembuat Pupuk Cair Berbasis NodeMCU dan Android " dengan baik dan benar. Laporan Skripsi ini wajib dibuat oleh mahasiswa Teknik Elektro, Fakultas Teknologi Industri, Universitas Islam Indonesia untuk memenuhi syarat dalam menempuh jenjang Strata 1.

Selama pembuatan Laporan Skripsi ini, penulis banyak menerima bantuan dan dukungan sehingga dapat menyelesaikan Laporan Skripsi ini. Oleh karena itu, penulis ingin mengucapkan terimakasih kepada :

1. Ibu Dwi Ana Ratna Wati S.T., M,Eng. selaku dosen pembimbing yang telah memberikan arahan dan bantuan selama melakukan penelitian hingga penyelesaian penulisan Laporan Skripsi ini,
2. Bapak Yusuf Aziz Amrulloh, S.T., M.Eng., Ph.D. selaku Ketua Jurusan Teknik Elektro, Fakultas Teknologi Industri, Universitas Islam Indonesia.
3. Bapak Medilla Kusriyanto, S.T., M.Eng. selaku Sekretaris Jurusan Teknik Elektro, Fakultas Teknologi Industri, Universitas Islam Indonesia.
4. Kedua orang tua dan keluarga yang selalu mendukung dan memberikan do'a dan usaha untuk kelancaran penelitian skripsi.
5. Teman-teman mahasiswa Jurusan Teknik Elektro Fakultas Teknologi Industri Universitas Islam Indonesia yang membantu penyelesaian masalah terkait penelitian skripsi.
6. Seluruh pihak yang telah memberikan bantuan selama proses penyelesaian penelitian hingga penulisan Laporan Skripsi.

Penulis menyadari bahwa Laporan Skripsi ini masih jauh dari sempurna karena ada keterbatasan ilmu dan pengalaman yang dimiliki. Oleh karena itu, kritik dan saran yang membangun akan diterima dengan baik. Semoga Laporan Skripsi ini bermanfaat bagi seluruh pihak yang memerlukan.

Wassalamu'alaikum Wr. Wb

Yogyakarta, 28 April 2020

Arigo Fahreza P

ARTI LAMBANG DAN SINGKATAN

kg	Kilogram
TPA	Tempat Pembuangan Akhir
AC	*Alternating Current*
ESP	*Espressif*
pH	*Potential Hydrogen*
USB	*Universal Serial Bus*
GND	*Ground*
GPIO	*General Purpose Input Output*
ADC	*Analog to Digital Converter*
NC	*Normally Closed*
NO	*Normally Open*
DC	*Direct Current*
PWM	*Pulse Width Modulation*
API	*Application Programming Interface*
AWS	*Amazon Web Service*
LCD	*Liquid Crystal Display*
VCC	*Voltage*
WiFi	*Wireless Fidelity*
RAM	*Random Access Memory*
OS	*Operating System*
CPU	*Central Processing Unit*
RTC	*Real Time Clock*
mm	Milimeter
MB	*Megabyte*
GPRS	*General Packet Radio Service*

ABSTRAK

Limbah organik merupakan salah satu limbah yang dapat mencemari lingkungan sekitar. Oleh karena itu, untuk mengatasi permasalahan tersebut dibutuhkan pengolahan limbah organik terlebih dahulu sebelum dibuang ke tempat pembuangan akhir (TPA). Pengolahan paling banyak untuk mengatasi permasalahan limbah organik adalah dibuat menjadi pupuk cair. Pembuatan pupuk cair organik diawali dari pencampuran bahan limbah organik dengan bakteri, lalu terjadi penguraian. Dalam pembuatan pupuk tersebut ditambahkan gula pasir dan diaduk setiap 3 hari untuk mempertahankan proses penguraian. Kendala selama pembuatan pupuk cair secara manual adalah kondisi temperatur yang harus dipantau karena akan berdampak pada bakteri yang bekerja dan juga apabila pembuat tidak berada di tempat, maka tidak dapat memantau kondisi pupuk cair setiap saat. Untuk memudahkan dua proses tersebut dibuat sistem *monitoring* dan sistem pengendali untuk memudahkan pembuat pupuk. Mesin mekanik pupuk cair telah ada yang terdiri dari motor *alternating current* (AC) dan juga alat pengaduk. Sistem *monitoring* dibuat dengan menggunakan NodeMCU dan juga termokopel tipe K untuk membaca temperatur selama proses pembuatan pupuk berlangsung. Kemudian melakukan komunikasi *wireless* dengan *server* untuk menyimpan data. Setelah data tersimpan akan langsung dikirim ke aplikasi android agar pengguna dapat melihat temperatur secara *real time*. Sistem pengendali menggunakan *relay* dan *timer* untuk mengatur kerja motor pengaduk AC satu *phase* dan modul pemberi gula otomatis berupa motor servo. Pengujian dilakukan untuk sistem *monitoring* yang dipasang pada mesin manual diuji untuk membuat pupuk cair dan juga sensor diuji dengan menggunakan air yang temperaturnya divariasikan kemudian dibandingkan dengan termometer alkohol. Untuk sistem pengendali diuji dengan cara mengatur waktu secara bervariasi dan kemudian dibandingkan dengan *stopwatch* untuk mengetahui apabila adanya keterlambatan waktu. Hasil dari penelitian ini, sistem *monitoring* dapat membaca temperatur dengan akurasi sebesar 93,12% dan toleransi pengukuran $\pm 0{,}3^0$C dan juga seluruh data yang terbaca telah tersimpan di dalam *server* yang dibuat. Sistem pengendali motor servo dan motor AC juga bekerja tanpa adanya keterlambatan waktu. Aplikasi android juga dapat menampilkan temperatur secara *real time* untuk setiap 3 detik seperti program yang diberikan. Pembacaan temperatur pada aplikasi android juga akan tertampil pada grafik untuk menunjukkan adanya perubahan temperatur. Secara umum penelitian ini dapat memberikan informasi tentang sistem otomatis pembuat pupuk cair organik dengan tepat dan akurat.

Kata kunci : Limbah Organik, Pupuk Cair Organik, NodeMCU, *Relay,* Android

DAFTAR ISI

LEMBAR PENGESAHAN 2
LEMBAR PENGESAHAN ..... **Error! Bookmark not defined.**
PERNYATAAN 3
KATA PENGANTAR ii
ARTI LAMBANG DAN SINGKATAN iii
ABSTRAK iv
DAFTAR ISI v
DAFTAR GAMBAR vii
DAFTAR TABEL viii
BAB 1 PENDAHULUAN 1
1.1 Latar Belakang Masalah 1
1.2 Rumusan Masalah 2
1.3 Batasan Masalah 2
1.4 Tujuan Penelitian 2
1.5 Manfaat Penelitian 2
BAB 2 TINJAUAN PUSTAKA 3
2.1 Studi Literatur 3
2.2 Tinjauan Teori 4
2.2.1 Pupuk Cair 4
2.2.2 NodeMCU 5
2.2.3 Termokopel 6
2.2.4 MAX6675 8
2.2.5 XY-LJ02 *Relay* 8
2.2.6 Motor Servo DC 9
2.2.7 Cloud Computing 10

BAB 3 METODOLOGI 12
3.1 Alur Penelitian 12
3.2 Perancangan Sistem 12
3.2.1 Proses Pembuatan Pupuk 12
3.2.2 Perangkat Keras Sistem *Monitoring* 14
3.2.3 Kalibrasi Sensor Termokopel Tipe K 15
3.2.4 Perangkat Keras Sistem Kendali Pemberian Gula 17
3.2.5 Perangkat Keras Sistem Pengaduk 18
3.2.6 *Flowchart* Sistem *Monitoring* dan Sistem Pengendali 20
3.2.7 Perangkat Lunak Aplikasi *Smartphone* 21
3.3 Cara Analisis 22
BAB 4 HASIL DAN PEMBAHASAN 23
4.1 Mesin Pembuat Pupuk Cair 23
4.2 Pengujian Sistem *Monitoring* dan Perangkat Lunak *Monitoring* 24
4.2.1 Hasil Kalibrasi 24
4.2.2 Hasil Pengujian Sistem *Monitoring* 25
4.2.3 Hasil Pengujian Aplikasi *Smartphone* 27
4.3 Pengujian Sistem Pengendali Pemberian Gula 27
4.4 Pengujian Sistem Pengendali Motor AC 30
BAB 5 KESIMPULAN DAN SARAN 34
5.1 Kesimpulan 34
5.2 Saran 34
DAFTAR PUSTAKA 35
LAMPIRAN 1

DAFTAR GAMBAR

Gambar 2.1 *Pinout* NodeMCU 6
Gambar 2.2 Cara Kerja Termokopel 7
Gambar 2.3 MAX6675 [12] 8
Gambar 2.4 *Relay* dan *Timer* XY LJ02 [13] 9
Gambar 2.5 Motor Servo DC 10
Gambar 3.1 Alur Penelitian 12
Gambar 3.2 Alur Proses Pembuatan Pupuk Cair 13
Gambar 3.3 Desain Sistem Mekanik Mesin Pupuk 13
Gambar 3.4 Blok Diagram Sistem *Monitoring* 14
Gambar 3.5 Rangkaian Perangkat Keras Sistem *Monitoring* 15
Gambar 3.6 Alur Pembacaan Temperatur Setelah Kalibrasi 17
Gambar 3.7 Program *Timer* Sistem Kendali Pemberian Gula 17
Gambar 3.8 Rangkaian Perangkat Keras Sistem Kendali Pemberian Gula 18
Gambar 3.9 Rangkaian Perangkat Keras Sistem Pengaduk 19
Gambar 3.10 Program *Trigger* pada Modul *Timer* XY LJ02 19
Gambar 3.11 *Flowchart* Sistem 20
Gambar 3.12 Perancangan Perangkat Lunak 22
Gambar 4.1 Mesin Mekanik Pembuat Pupuk Cair 23
Gambar 4.2 Sistem *Monitoring* (Terpasang) 25
Gambar 4.3 Perubahan Temperatur Proses Pembuatan Pupuk Cair 26
Gambar 4.4 Aplikasi Android (*running*) 27
Gambar 4.5 Perangkat Pemberian Gula 28
Gambar 4.6 Perangkat Pemberian Gula (Terpasang) 30
Gambar 4.7 Rangkaian Terpasang Pengendali Motor AC 31
Gambar 4.8 Fungsi *Timer* dan *Relay* 32

DAFTAR TABEL

Tabel 2.1 *Pinout* NodeMCU 6
Tabel 2.2 Jenis-Jenis Termokopel 8
Tabel 3.1 Pengambilan Data Temperatur 16
Tabel 4.1 Pembacaan Temperatur Setelah Kalibrasi 24
Tabel 4.2 Perbandingan Waktu Antara *Delay* Terhadap *Real Time* Motor Servo 29
Tabel 4.3 Perbandingan Waktu Antara *Delay* Terhadap *Real Time* Motor AC 33

BAB 1
PENDAHULUAN

1.1 Latar Belakang Masalah

Limbah merupakan bahan yang mengandung racun yang dapat menyebabkan beberapa polusi. Jika limbah tersebut berasal dari rumah tangga maka dapat menyebabkan polusi tanah dan air. Dari limbah tersebut dapat menyebabkan kerugian bagi lingkungan sekitar. Tidak hanya pencemaran lingkungan, tapi limbah yang menumpuk juga menyebabkan kondisi lingkungan kotor sehingga dapat menimbulkan penyakit. Berdasarkan data dari Badan Lingkungan Hidup Kota Yogyakarta dalam satu hari dengan lima orang anggota keluarga dapat menghasilkan limbah 4.500 gram sehingga jika diakumulasikan satu tahun akan menghasilkan 1.620 kilogram (kg) [1]. Perilaku masyarakat juga mempengaruhi penganggulangan limbah tersebut. Limbah tersebut adalah limbah yang akan masuk ke dalam TPA. Pada TPA tersebut terdapat limbah organik yang masih memiliki banyak nutrisi yang dapat dikembangkan menjadi pupuk untuk meningkatkan pertumbuhan dan perkembangan pertanian [2].

Salah satu pengolahan untuk limbah organik adalah dengan dibuatnya menjadi pupuk. Pupuk terdiri dari pupuk cair dan padat. Metode untuk membuat pupuk mempunyai beragam cara. Proses pembuatan pupuk cair dapat dilakukan dengan cara menggunakan bantuan bakteri untuk mempercepat dan membantu proses penguraian. Pupuk bisa jadi selama satu minggu hingga paling lama satu bulan. Pada proses pembuatan pupuk menggunakan metode anaerob, diperlukan pengamatan temperatur dan tidak dapat terkena sinar matahari langsung. Dalam proses tersebut terdapat juga pemberian gula dan juga proses pengadukan untuk setiap tiga hari sekali. Kendala dalam pembuatan pupuk secara manual adalah kondisi temperatur yang tidak selalu terpantau oleh pembuat dan pembuat tidak selalu berada di tempat pembuatan pupuk. Hal tersebut dapat menyebabkan bakteri pengurai tidak bekerja maksimal. Karena inti dari proses pembuatan pupuk ada pada bakterinya dan bakteri juga tidak bisa hidup di tempat yang panas, biasanya bakteri pengurai hidup ditemperatur 25^0C – 40^0C [3]. Untuk mempermudah seluruh proses tersebut diperlukan alat yang dapat berjalan secara otomatis, sehingga pengguna hanya memasukkan bahan-bahan yang diperlukan dan menunggu proses selesai. Kemudian juga ditambahkan pengamatan temperatur dengan menggunakan aplikasi untuk *smartphone* sehingga pengguna hanya perlu mengamati melalui *smartphone* masing-masing.

Penelitian ini akan merancang alat yang dapat digunakan pengguna agar dapat melakukan pembuatan pupuk secara otomatis. Sehingga pengguna tidak perlu membuang waktu untuk mengamati dan menunggu proses selesai. *Server* juga disiapkan untuk menyimpan seluruh hasil

pembacaan temperatur dan waktu secara *real time* dengan harapan apabila data tersebut ingin digunakan untuk proses analisis lebih lanjut [4].

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah :

1. Bagaimana cara merancang sistem *monitoring* dan sistem pengendali pada mesin pembuat pupuk cair berbasis NodeMCU dan android.
2. Bagaimana unjuk kerja dari sistem *monitoring* dalam hal ini yaitu akurasi pembacaan temperatur dan juga pengiriman data ke *server*.
3. Bagaimana unjuk kerja dari sistem pengendali tentang ketepatan waktu proses yang sudah ditentukan oleh pengguna.

1.3 Batasan Masalah

1. Mesin pupuk cair sudah ada yang terdiri dari motor induksi AC *single phase* dengan 1400 rotasi tiap menit dan juga alat pengaduk.
2. Aplikasi android dibuat dengan menggunakan Android Studio versi 3.5.0.0.
3. Alat yang digunakan diciptakan dengan skala rumah tangga.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk merancang sistem *monitoring* dan sistem pengendali pada mesin pembuat pupuk cair yang berbasis android dan NodeMCU beserta hasil dari unjuk kerja alat tersebut meliputi akurasi dari pembacaan sensor dan juga ketepatan waktu proses dari sistem pengendali motor servo dan motor AC.

1.5 Manfaat Penelitian

Penelitian ini bermanfaat untuk mengolah limbah yang menjadi permasalahan lingkungan agar dapat diubah menjadi pupuk cair. Pupuk cair yang dihasilkan juga mempunyai kadar *nitrogen, fosfor*, dan *kalium* yang bagus sehingga mempunyai efek yang maksimal. Penelitian ini juga digunakan untuk mempermudah proses agar pengguna dapat mudah menghasilkan pupuk cair dengan sistem otomatis.

BAB 2
TINJAUAN PUSTAKA

2.1 Studi Literatur

Penelitian yang terkait dengan pembahasan pupuk organik telah banyak dilakukan, karena perkembangan pertanian dan teknologi yang semakin pesat. Penelitian pernah dilakukan oleh H. Soewardi dan R. Laksmita yang meneliti tentang cara menentukan komposisi yang optimum untuk pupuk cair. Penelitian ini dilakukan karena banyak dari pupuk cair yang dijual di pasaran tidak sesuai dengan standar yang ada di Indonesia. Sehingga pupuk tersebut membahayakan tanaman dengan cara menghambat pertumbuhannya. Sehingga penelitian ini bertujuan untuk menciptakan pupuk yang sesuai dengan standar di pertanian Indonesia dengan menggunakan metode *Taguchi* . Penelitian tersebut dapat menghasilkan pupuk cair organik dengan *potential hydrogen* (pH) 7 [5].

Penelitian lain juga dilakukan oleh Jordão, Marcelo D. L. yang meneliti tentang membuat alat yang terjangkau untuk *monitoring* temperatur dalam proses pembuatan pupuk kompos. Penelitian ini bertujuan untuk membuat alat *monitoring* dengan harga tidak lebih dari 2 juta rupiah. Sehingga untuk mendapatkan tujuan tersebut komponen yang digunakan harus terjangkau dan mudah didapat. Alat tersebut menggunakan Arduino Uno sebagai komponen utama dan sensor temperatur dengan jenis DS18B20. Data-data yang terbaca oleh sensor akan disimpan ke dalam *Micro SD* untuk mengamati unjuk kerja dari alat yang telah dibuat. Penelitian tersebut menghasilkan alat yang sudah teruji dengan memberikan hasil dari regresi dengan akurasi $\pm 1^0$C untuk kalibrasi dari sensor temperatur [6].

Penelitian yang dilakukan oleh Manu, M.K dan Kumar, R yang meneliti tentang pembuatan pupuk kompos dengan menggunakan drum dari limbah makanan dengan menganalisis energi kinetiknya. Penelitian ini dilakukan pada tahun 2016, penelitian ini menggunakan drum sebagai media utama dan memberikan sirkulasi udara secara alami dengan cara melubangi pinggiran drum dengan diameter 10 milimeter (mm). Bahan yang digunakan untuk membuat pupuk tidak hanya *food waste* tetapi juga menggunakan *yard waste* sebagai tambahan. Penelitian ini menghasilkan temuan *reaction rate* sebanyak 0,015 per hari [7].

Penelitan lain juga dilakukan oleh Vrettos G dan Kazamias G yang telah dipublikasikan pada tahun 2017 tentang pengembangan teknologi *smart composting system monitoring*. Penelitian ini bertujuan untuk membuat sistem *monitoring* untuk membantu proses pembuatan pupuk. Sistem yang digunakan menggunakan *wireless sensor* dan Arduino. Beberapa sensor yang digunakan adalah temperatur DHT11, metana CNG , dan kelembaban DHT11. Pembacaan data dari sensor tersebut akan dikirimkan ke *server* dan pengguna dapat melihat angka tersebut pada aplikasi

website. Pengukuran akan dilakukan setiap satu jam dan setiap data yang telah disimpan dapat di *download* oleh pengguna [8].

Penelitian juga pernah dikembangkan oleh Oscar Casas, tentang pengembangan *wireless sensor network* untuk membantu dan mengatur proses selama pembuatan pupuk. Penelitian tersebut bertujuan untuk membuat alat dengan hasil pembacaan sensor yang terbungkus harus tetap akurat dan teruji dari segi mekanik dan elektrik. Alat tersebut menggunakan NodeMCU yang digunakan sebagai komponen utama dan *web interface* sebagai penunjuk untuk mengetahui hasil pembacaan dari sensor temperatur dan kelembaban. Penelitian tersebut menghasilkan sensor yang teruji dan menghasilkan akurasi $0{,}1^0$C untuk rentang dari 0 – 80^0C. Sensor kelembaban juga mempunyai akurasi 5% dengan rentang 20 – 80% [4].

Berdasarkan studi literatur yang telah dilakukan, penelitian tersebut saling berkaitan karena memberikan informasi tentang bagaimana komposisi yang baik untuk pupuk cair dan juga metode pembuatan pupuk secara manual menggunakan drum. Terdapat pembahasan mengenai teknologi seperti sistem *monitoring* yang digunakan untuk memantau kondisi temperatur dengan menggunakan berbagai sensor seperti DHT11 dan DS18B20. Data tersebut dapat ditampilkan dalam *website* sehingga pengguna dapat melihat menggunakan *browser* dan komputer. Literatur tersebut akan berhubungan dengan penelitian yang akan dilakukan. Penelitian ini akan menggunakan teknologi sensor yang akan terkena langsung oleh zat pupuk cair. Sehingga akan mengurangi kesalahan pembacaan dan untuk mengurangi kesalahan akan dikalibrasi dengan regresi linear. Seluruh pembacaan temperatur akan dikirim ke *server* yang telah dibuat untuk menyimpan seluruh data. Untuk mempermudah pengamatan oleh pengguna maka dibutuhkan sejenis *mobile app*, karena apabila menggunakan halaman *website* harus dilihat melalui *browser* ataupun komputer sehingga diperlukan aplikasi yang portabel pada *smartphone* seperti aplikasi android.

2.2 Tinjauan Teori

Bagian tinjauan teori merupakan bagian yang menjelaskan tentang dasar-dasar ilmu yang digunakan dan juga penjelasan teori tentang piranti yang digunakan. Teori-teori tersebut akan dijelaskan pada bagian selanjutnya.

2.2.1 Pupuk Cair

Limbah sayur dan sisa buah termasuk ke dalam limbah yang *open dumping* atau biasanya tanpa pengolahan sebelum dibuang sehingga dapat menimbulkan pencemaran lingkungan. Penanganan paling baik adalah pengolahan dengan membuat pupuk karena dari gizi yang

diberikan oleh limbah tersebut dan dengan bantuan mikroorganisme dapat menciptakan fermentasi. Hasil tersebut dapat meningkatkan keragaman mikroba tanah dan juga peran dari mikroorganisme dapat mempercepat proses pembuatan pupuk. Pupuk organik dapat menjadi pilihan lain pengganti dari pupuk kimia. Pembuatan pupuk tidak hanya harus cepat tetapi juga harus memperhatikan kualitas yang dihasilkan yaitu nilai pH dan juga kandungan *Nitrogen*, *Kalium*, dan *Fosfor*. Faktor eksternal seperti temperatur dan kelembaban juga dapat mempengaruhi proses tersebut. Karena jika kondisi temperatur terlalu panas maka mikroorganisme akan mati, sedangkan jika terlalu dingin maka mikroorganisme belum dapat bekerja. Oleh karena itu biasanya pembuatan pupuk disimpan ditempat yang terlindung dari sinar matahari secara langsung [9].

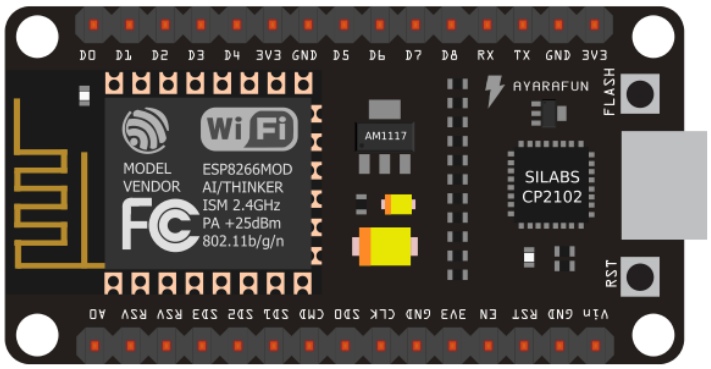
2.2.2 NodeMCU

NodeMCU merupakan piranti yang umum digunakan dalam pengembangan *Internet of Things*. NodeMCU berasal dari singkatan *node* dan *Microcontroller unit* (MCU). NodeMCU berfungsi sebagai alat yang dapat menghubungkan ke jaringan internet dengan menggunakan *Wi-Fi*. NodeMCU juga dapat berfungsi sebagai titik akses yang menjadi penghubung ke perangkat lain, sehingga tidak hanya menyambungkan koneksi tetapi dapat membuat koneksi dasar sendiri. NodeMCU merupakan perangkat keras yang menggunakan modul dasar *Espressif*-12 (ESP-12). Karena ESP-12 tidak mempunyai *microcontroller* sendiri maka harus menggunakan piranti tambahan seperti Arduino ataupun jenis lainnya. Terdapat tiga generasi dari seri NodeMCU, generasi pertama sering disebut juga V1 adalah seri yang menggunakan inti ESP-12 dengan ukuran memori sebesar 4 *Megabyte* (MB). Kemudian seri kedua yaitu dilakukan peningkatan dengan mengubah inti yang digunakan dengan ESP-12E dan komunikasi serial yang sebelumnya terdapat pada seri pertama yaitu CHG340 sedangkan pada seri kedua yaitu menggunakan CP2102. Kemudian generasi yang paling banyak digunakan sekarang yaitu seri ketiga dengan perbedaan penambaan jumlah pin untuk *ground* (GND). Suplai tegangan untuk menyalakan NodeMCU yaitu sebesar 5V. Oleh karena itu NodeMCU menyediakan suplai menggunakan *universal serial bus* (USB) karena tegangan yang diberikan komputer/laptop sebesar 5V. NodeMCU menyediakan 22 pinout yang dapat digunakan sesuai dengan kebutuhan. Berikut ini adalah Tabel 2.1 yang menyebutkan tentang *pinout* NodeMCU [10].

Tabel 2.1 *Pinout* NodeMCU

Pinout	Nama
1	*External Reset*
2	*Converter Analog Digital*
3	*Chip Enable*
4	GPIO16
5	GPIO14
6	GPIO12
7	GPIO13
8	GND
9	GPIO11
10	GPIO07
11	GPIO09
12	GPIO10
13	GPIO08
14	GPIO06
15	*Voltage* (VCC)
16	GPIO15
17	GPIO02
18	GPIO00
19	GPIO04
20	GPIO05
21	GPIO03
22	GPIO01

Pinout NodeMCU paling banyak digunakan sebagai *general purpose input output* (GPIO). GPIO merupakan pin yang dapat digunakan sebagai keluaran untuk keperluan tertentu. *Pinout* NodeMCU akan ditunjukkan pada Gambar 2.1.

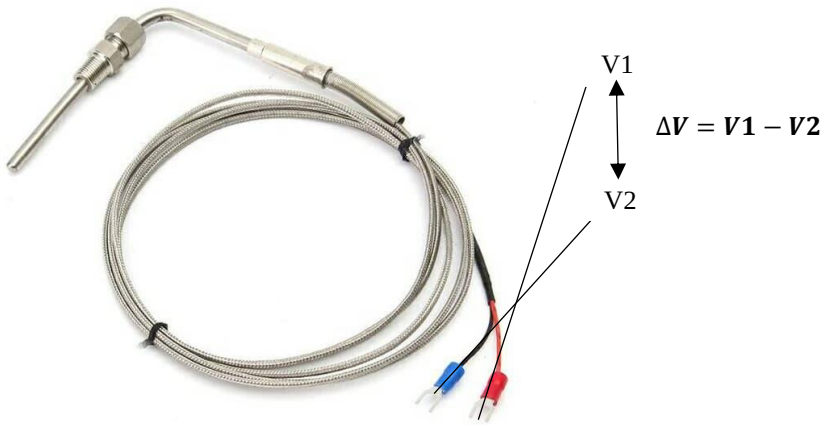


Gambar 2.1 *Pinout* NodeMCU

2.2.3 Termokopel

Termokopel adalah sensor temperatur yang berfungsi dengan cara mendeteksi adanya perbedaan panas dengan menggunakan dua sambungan berbeda dan mengubahnya menjadi suatu tegangan listrik. Termokopel juga termasuk ke dalam jenis termometer elektronik. Jangkauan

temperatur yang dapat diukur oleh termokopel berbeda-beda sesuai dengan jenisnya. Gambar 2.2 akan menjelaskan cara kerja termokopel.



Gambar 2.2 Cara Kerja Termokopel

Karena pada dasarnya termokopel terdiri dari dua kawat konduktor logam yang berbeda kemudian digabungkan pada ujungnya, sehingga satu logam akan menjadi referensi dengan temperatur konstan sedangkan logam satunya akan mendeteksi temperatur yang diukur. Perbedaan termperatur tersebut akan menghasilkan beda potensial sesuai dengan rumus berikut.

$$\Delta V = V1 - V2 \tag{2.1}$$

Berdasarkan persamaan (2.1) jika kedua konduktor logam membaca temperatur yang sama maka beda potensial yang dihasilkan sama dengan 0. Akan tetapi, jika ada temperatur yang panas/dingin diberikan ke sensor maka akan terjadi beda potensial. Biasanya tegangan yang dihasilkan mulai dari 1µV – 70 µV tiap derajat Celcius. Jika pembacaan tersebut ingin dibaca oleh suatu komputer maka perlu ditambahkan pengondisi sinyal atau yang biasa disebut *analog to digital converter* (ADC) untuk mengubah tegangan yang dihasilkan oleh sensor menjadi nilai digital yang dapat dibaca komputer. Kemudian berikut ini adalah Tabel 2.2 yang menyebutkan jenis-jenis termokopel [11].

Tabel 2.2 Jenis-Jenis Termokopel

No	Tipe	Bahan		Rentang Temperatur (^{0}C)	Output (μV^0C)
		Positif	Negatif		
1.	E	Nikel - Kromium	Nikel - Tembaga	-200 – 900	68
2.	J	Besi	Nikel - Tembaga	0 – 750	52
3.	K	Nikel - Kromium	Nikel - Alumunium	-200 – 1250	41
4.	N	Nikel - Kromium - Silikon	Nikel -Silikon - Magnesium	0 – 1250	39
5.	T	Tembaga	Tembaga - Nikel	-200 – 350	43
6.	B	Rhodium - Platinum	Platinum	>1800	10
7.	R	Rhodium - Platinum	Platinum	>1600	10
8.	S	Rhodium - Platinum	Nikel - Tembaga	>1600	10

2.2.4 MAX6675

MAX6675 merupakan komponen yang dapat mengubah sinyal termokopel yang awalnya analog menjadi digital dengan masukan data 12 bit ADC. MAX6675 diatur untuk dapat bekerja dengan *microcontroller* atau aplikasi *monitoring* lainnya. Alat ini dapat mengurangi *noise* yang diberikan oleh termokopel yang terhubung dengan bagian positif dan negatif dari termokopel. Unjuk kerja dari MAX6675 akan berjalan secara optimal apabila temperatur pada acuan termokopel dan *device* berada pada temperatur yang sama. Gambar MAX6675 akan ditunjukkan pada Gambar 2.3.

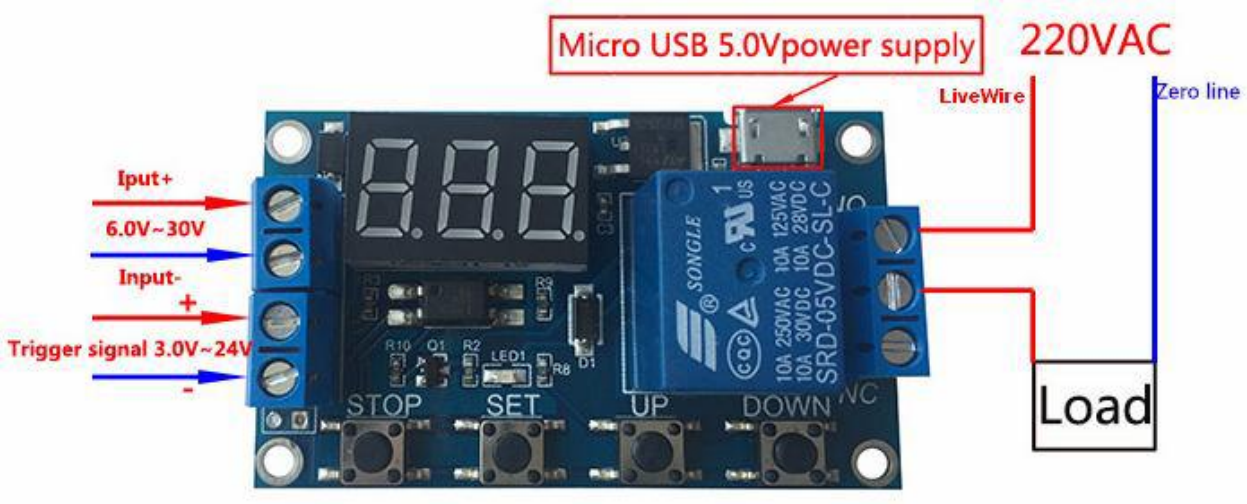


Gambar 2.3 MAX6675 [12]

2.2.5 XY-LJ02 *Relay*

Relay merupakan saklar otomatis yang dioperasikan secara elektrik yang terdiri dari dua bagian yaitu *coil* dan kontak saklar. Cara kerja dari *relay* menggunakan prinsip elektromagnetik, sehingga apabila *coil* diberi arus listrik maka *armature* akan tertarik sehingga menjadi berubah kondisi. Ada dua jenis kondisi pada *relay* yaitu *normally closed* (NC) dan *normally open* (NO). Kondisi NC adalah kondisi dimana kondisi awal saklar sudah dalam keadaan tertutup dan jika kondisi NO adalah kondisi dimana kondisi awal dari saklar dalam keadaan terbuka. Sehingga kondisi NC dan NO dapat dimanfaatkan untuk situasi yang berbeda jika ingin begitu saklar diberi arus ingin menghidupkan sesuatu maka pasang saklar dalam keadaan NO, begitu juga untuk kondisi sebaliknya. *Relay* biasanya dikendalikan dengan menggunakan *trigger*. *Trigger* sendiri

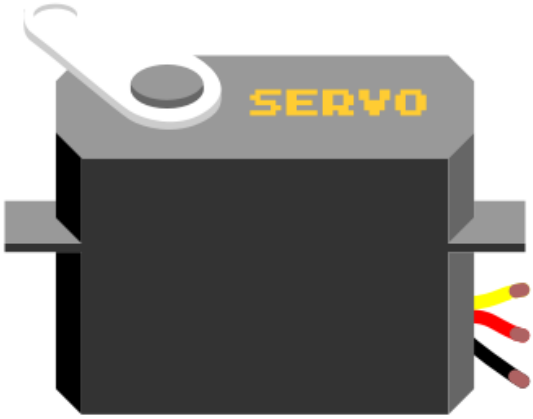
dapat diatur keluaranya dengan menggunakan *timer* atau juga program yang dibuat. XY LJ02 merupakan *relay* dan *timer* dalam satu modul. Sehingga tidak perlu komponen tambahan untuk mengatur kerja *relay*, cukup dengan menggunakan *timer* tersebut untuk mengatur *trigger*. Gambar *relay* akan ditunjukkan pada Gambar 2.4.



Gambar 2.4 *Relay* dan *Timer* XY LJ02 [13]

2.2.6 Motor Servo DC

Motor servo *direct current* (DC) adalah motor listrik yang digunakan untuk mengendalikan kecepatan pada pada listrik DC. Motor servo dapat bekerja dengan cara memberikan masukan sebuah *pulse width modulation* (PWM) sebagai target dan motor servo akan bergerak ke suatu posisi. Motor DC terdapat beberapa jenis seperti posisi rotasi, rotasi 360, linear, dan *brushless*. Motor DC yang paling banyak digunakan yaitu posisi rotasi. Contoh dalam penggunaanya adalah suatu sistem yang membutuhkan pergerakan secara akurat untuk mencapai suatu posisi seperti lengan robot, mobil *remote control*, dan pesawat. Motor DC jenis ini hanya dapat bekerja dengan radius 180^0. Untuk mengatur target dari motor DC dapat menggunakan program dengan *microcontroller* Arduino. Motor servo DC akan ditunjukkan pada Gambar 2.5.



Gambar 2.5 Motor Servo DC

2.2.7 Cloud Computing

Cloud Computing merupakan komputasi yang dilakukan melalui *cloud server*. Proses pengolahan tersebut meliputi (*Central Processing Unit* (CPU), *Random Access Memory* (RAM), *Operating Service* (OS), dan juga *Software*). *Cloud computing* biasanya menggunakan beberapa komponen seperti *virtual computer*, *server*, dan *user interface*. *Virtual computer* bisa didapatkan dari penyedia layanan *cloud* dan *database* seperti *Amazon Web Service* (AWS). AWS merupakan penyedia layanan *cloud* yang aman. Selain menyediakan *cloud computing*, AWS juga menawarkan ruang penyimpanan *database*. Sebagai analogi jika lokal *server* memerlukan komputer yang harus selalu menyala agar *server* tetap berjalan, sedangkan AWS menjadi komputer virtual yang disewa dan akan selalu berjalan 24 jam sehingga *server* akan selalu berjalan. Komputer virtual tersebut akan menjalankan *database* yang sudah dibuat. *Database* merupakan kumpulan data yang diorganisir oleh sistem agar dapat menyimpan data. Penyedia sistem *database* juga berbagai macam, salah satunya merupakan Sqlite *Database*. Sqlite *Database* merupakan mesin *database* yang dapat ditanam pada sistem yang ingin digunakan. Komunikasi utama yang digunakan yaitu melalui pemanggilan *application programming interface* (API) secara langsung melalui bahasa pemrograman yang digunakan. *Database* tersebut dapat ditulis dalam berbagai bahasa dengan fungsi tertentu seperti *read, write, delete*, dan masih banyak fungsi lainya. Sqlite merupakan *database* lokal, jika ingin membuatnya terhubung dengan internet maka diperlukan *web-service* sebagai penyedia *server* untuk membuat *database* selalu menyala. Sebagai contoh penyedia *web-service* yang paling umum yaitu AWS. Kemudian bagian *user interface* yang paling mudah digunakan adalah halaman *website.* Tetapi kelemahan dalam penggunaan *website* adalah hanya

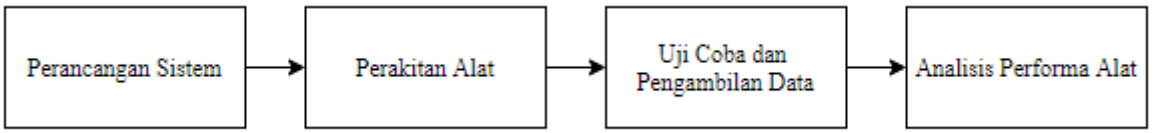
dapat dibuka menggunakan *browser* sehingga apabila menggunakan aplikasi android diharapkan akan lebih ringkas, karena perangkat pembuka aplikasi dapat dibawa – bawa lebih mudah.

Android merupakan sistem operasi yang telah dirancang oleh Google dan bersifat *open source*. Oleh karena itu *smartphone* paling banyak menggunakan sistem operasi ini dan juga banyak pengembang aplikasi yang mengembangkan aplikasi dengan sistem operasi tersebut. Untuk mempermudah hal tersebut pihak pengembang Android membuat perangkat lunak Android Studio untuk membuat aplikasi Android. Bahasa pemrograman yang digunakan juga beragam yaitu terdapat java, kotlin, dan juga antarmuka flutter. Sudah banyak fungsi-fungsi yang diberikan oleh pengembang aplikasi di internet secara gratis, sehingga setiap orang tidak perlu susah untuk mencari *source code* yang ingin digunakan. Untuk menghubungkan antara aplikasi android dengan database pihak Google juga memberikan fitur *Firebase Realtime Database*. *Firebase* digunakan untuk menyimpan dan menyinkronkan data secara *real-time* terhadap *database* SQL [14].

BAB 3
METODOLOGI

3.1 Alur Penelitian

Pembuatan sistem *monitoring* dan sistem pengendali pada mesin pupuk cair memerlukan alur agar dapat menghasilkan suatu alat sesuai dengan tujuan penelitian. Tahapan tersebut terdiri dari perancangan rangkaian pada alat, perakitan alat, melakukan uji coba untuk pengambilan data, dan menganalisis unjuk kerja alat tersebut. Untuk lebih jelas skema alur penelitian akan ditunjukkan pada gambar Gambar 3.1.



Gambar 3.1 Alur Penelitian

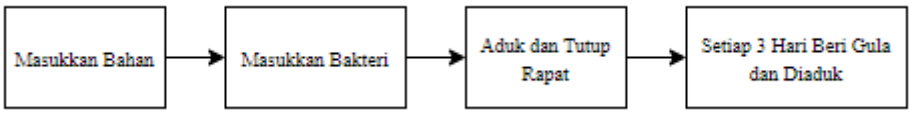
Alur pertama adalah perancangan rangkaian sistem, perancangan tersebut dilakukan untuk mendapatkan *wiring* dan daftar komponen yang dibutuhkan. *Wiring* dilakukan dengan menggunakan perangkat lunak Proteus dan Fritzing. Kemudian alur selanjutnya adalah perakitan alat, setelah *wiring* selesai didesain maka proses perakitan akan lebih mudah. Kemudian uji coba alat dilakukan untuk mendapatkan data apakah alat tersebut sudah berjalan baik atau belum. Parameter baik atau belum disesuaikan dengan target yang ingin dicapai. Kemudian alur terakhir adalah menganalisis apa saja unjuk kerja yang kurang dan memperbaiki alat jika dibutuhkan untuk memperbaiki unjuk kerja.

3.2 Perancangan Sistem

Perancangan sistem akan menjelaskan bagaimana seluruh skema metode pembuatan pupuk, perangkat keras, dan perangkat lunak yang digunakan. Metode pembuatan pupuk akan dijelaskan karena akan berhubungan dengan cara kerja dari sistem perangkat keras. Pembuatan perangkat lunak akan dijelaskan karena akan digunakan untuk sistem *monitoring.*

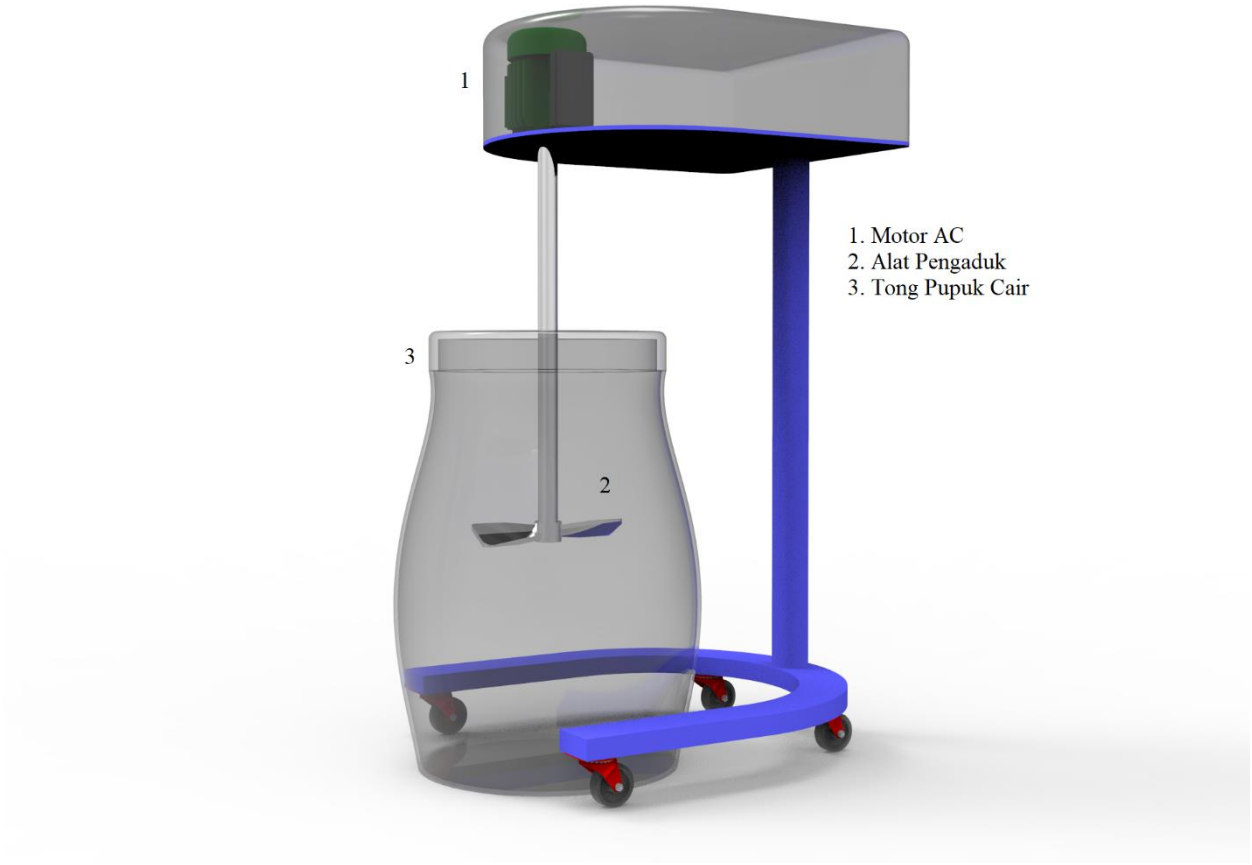
3.2.1 Proses Pembuatan Pupuk

Proses pembuatan pupuk akan dilakukan selama satu minggu dan akan terdapat aksi untuk setiap 3 hari sekali yaitu pemberian gula dan pengadukan. Untuk detail proses akan ditunjukkan pada Gambar 3.2



Gambar 3.2 Alur Proses Pembuatan Pupuk Cair

Pupuk cair dibuat dengan menggunakan bahan-bahan organik dan bantuan bakteri. Proses tersebut dapat dilakukan secara manual, tetapi untuk melakukan proses tersebut secara manual harus terus memantau kondisi pupuk. Setelah bahan – bahan dimasukkan bakteri akan bekerja untuk memproses pupuk tersebut yang mengakibatkan penambahan temperatur. Bakteri tersebut harus tetap disimpan dengan cara setiap 3 hari sekali harus diberi nutrisi untuk sumber makanan bakteri. Untuk membuat seluruh area mendapatkan nutrisi yang sama dilakukan pengadukan pada saat yang sama setelah pemberian nutrisi. Sehingga untuk proses selama satu minggu, aksi tersebut akan dilakukan dua kali. Sehingga untuk mempermudah hal tersebut seperti pemantauan temperatur dan juga sistem pengadukan dan pemberian gula dapat dibuat secara otomatis.



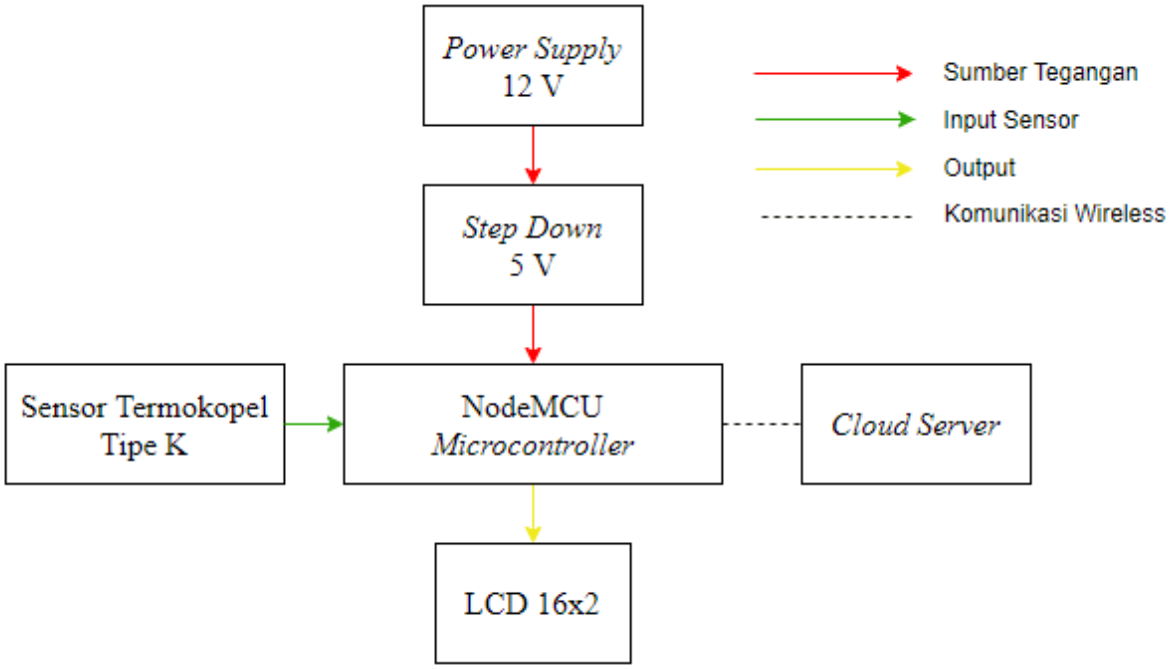
Gambar 3.3 Desain Sistem Mekanik Mesin Pupuk

Berdasarkan Gambar 3.3 untuk melakukan proses pengadukan, digunakan motor induksi *single phase* karena dalam pupuk cair terdapat bahan padat sehingga perlu motor yang mempunyai torsi yang besar dari pada kecepatan putar yang tinggi. Agar dapat berjalan otomatis ditambahkan *relay*

dan *timer* yang digunakan untuk menyalakan dan mematikan motor induksi tersebut. Waktu tersebut diatur sesuai dengan metode pembuatan pupuk. Karena sasaran alat ini adalah rumah tangga sehingga saat pengguna hanya dapat mengakomodasi sistem *monitoring*, sistem lainya dapat dijalankan secara manual.

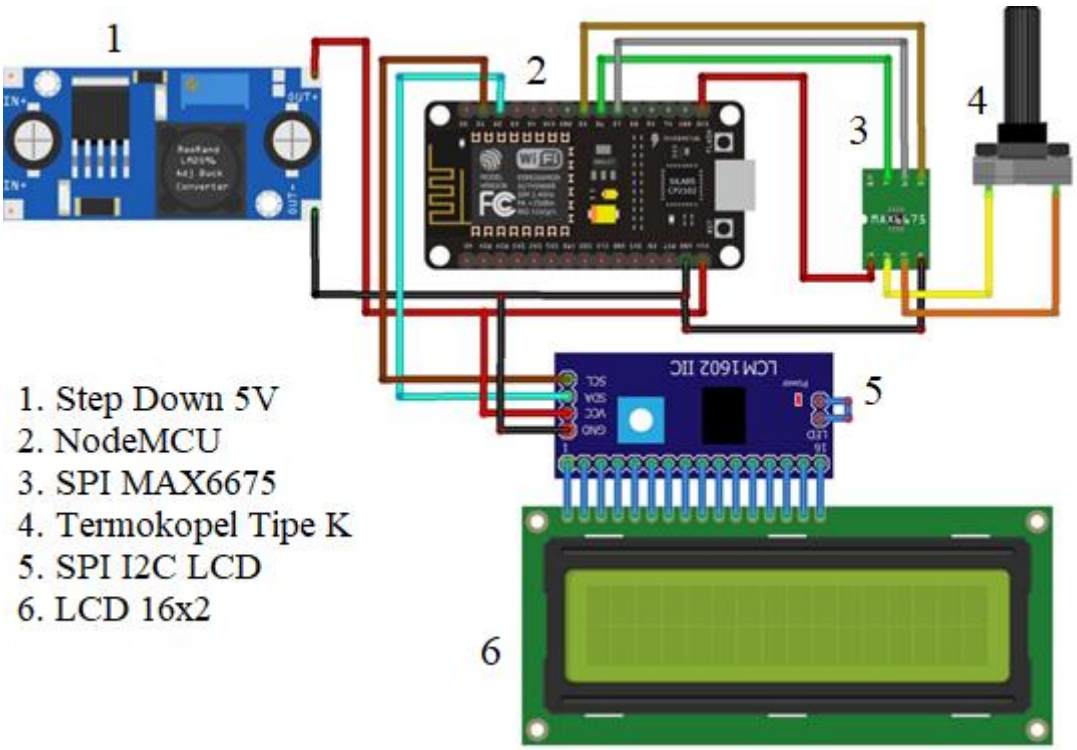
3.2.2 Perangkat Keras Sistem *Monitoring*

Sistem *monitoring* digunakan untuk memantau kondisi temperatur selama proses pembuatan pupuk. Oleh karena itu sistem akan menggunakan sensor temperatur dan *microcontroller* sebagai komponen utama dalam sistem ini. Blok diagram sistem akan menunjukkan gambaran secara umum bagaimana sistem berjalan. Blok diagram akan ditunjukkan pada Gambar 3.4.



Gambar 3.4 Blok Diagram Sistem *Monitoring*

Rangkaian perangkat keras sistem *monitoring* akan ditunjukkan pada Gambar 3.5



Gambar 3.5 Rangkaian Perangkat Keras Sistem *Monitoring*

Rangkaian terdiri dari komponen NodeMCU, DC *Step Down*, *Liquid Crystal Display* (LCD) dan I2C 16x2, dan Sensor temperatur. *Microcontroller* yang digunakan sudah terdapat modul yang dapat menghubungkan ke jaringan *wirelles fidelity* (WiFi). Penggunaan *microcontroller* yang sudah termasuk dengan modul WiFi dapat menghemat penggunaan komponen yang digunakan. Sensor temperatur yang digunakan adalah termokopel tipe K, karena rentang temperatur yang akan diukur dengan maksimal 50^0C sesuai dengan temperatur kehidupan bakteri pada proses biogas. Ketelitian dari pembacaan termokopel tipe K adalah 1^0C yang dapat membaca di dalam media cairan. *Step Down* digunakan untuk menurunkan tegangan dari *power supply* 12V menjadi 5V untuk menyalakan NodeMCU, sehingga NodeMCU tidak perlu selalu terhubung dengan komputer. Setiap jalur yang berwarna merah menandakan tegangan VCC dan jalur hitam menandakan GND. LCD 16x2 digunakan untuk menunjukkan temperatur selain pada perangkat lunak. Komponen ini digunakan jika pengguna sedang atau tidak membawa *smartphone* maka tetap dapat melihat temperatur pada LCD.

3.2.3 Kalibrasi Sensor Termokopel Tipe K

Kalibrasi sensor akan dilakukan dengan menggunakan metode regresi linear. Metode regresi liner dapat digunakan untuk mendekati pemodelan hubungan antara dua variabel. Regresi linear menggunakan dua variabel x dan y. Variabel x merupakan variabel yang tidak terikat, sedangkan variabel y merupakan variabel terikat. Variabel x merupakan data pengukuran temperatur oleh

sensor dan variabel y merupakan data pengukuran temperatur oleh temometer alkohol. Termometer alkohol dijadikan sebagai acuan pembacaan temperatur yang akan dibandingkan dengan pembacaan temperatur sensor. Pemilihan regresi linear untuk kalibrasi ini berdasarkan studi literatur yang telah dilakukan karena dapat mengurangi nilai *error* yang dihasilkan sensor pada saat setelah proses kalibrasi. Metode regresi merupakan metode sederhana yang dapat memprediksi hasil keluaran sensor dengan mengetahui hubungan antara temperatur termokopel dan termometer alkohol. Variabel x dan y akan menunjukkan persamaan regresi linear seperti pada persamaan (3.1).

$$\boldsymbol{Y = a + bX} \tag{3.1}$$

Persamaan (3.1) membutuhkan variabel a dan b yang didapat melalui perhitungan. Rumus tersebut akan ditunjukkan pada persamaan (3.2) dan (3.3).

$$\boldsymbol{a = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum xy)}{n(\sum x^2) - (\sum x)^2}} \tag{3.2}$$

$$\boldsymbol{b = \frac{n(\sum xy) - (\sum x)(\sum y)}{n(\sum x^2) - (\sum x)^2}} \tag{3.3}$$

Pada penelitian kali ini kalibrasi sensor dilakukan dengan cara membaca temperatur dengan menggunakan acuan termometer alkohol pada air yang diatur pada temperatur yang berbeda-beda dengan sampel sebanyak 70 pembacaan untuk setiap temperatur yang ditentukan dan hasil dari pembacaan akan dirata – rata. Tabel data pembacaan akan ditunjukkan pada Tabel 3.1.

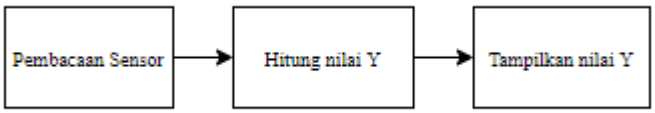
Tabel 3.1 Pengambilan Data Temperatur

Pembacaan Temperatur (^{0}C)	
Sensor	**Termometer Alkohol**
30,8	30
40,8	40
51,3	50
61,7	60
71,3	70
81,3	80

Pengambilan data dilakukan pada temperatur 30^{0}C hingga 80^{0}C untuk setiap kelipatan 10^{0}C. Kemudian perhitungan regresi linear dilakukan menggunakan program Matlab dengan rumus pada persamaan (3.2) dan (3.3) .Sehingga persamaan regresi yang dihasilkan seperti pada persamaan (3.4).

$$\boldsymbol{Y = -0.4889 + 0.9871X} \tag{3.4}$$

Hasil dari regresi tersebut akan dimasukkan ke dalam program yang digunakan untuk membaca temperatur. Sehingga hasil pembacaan temperatur akan dikalkukasi ke dalam persamaan (3.4) sebagai variabel x dan akan menghasilkan variabel y yang akan ditampilkan ke LCD dan dikirim ke *server*. Sehingga alur pembacaan temperatur akan ditunjukkan pada Gambar 3.6.



Gambar 3.6 Alur Pembacaan Temperatur Setelah Kalibrasi

3.2.4 Perangkat Keras Sistem Kendali Pemberian Gula

Sistem kendali pemberian gula terdiri dari motor servo yang digunakan untuk mengatur buka dan tutup dari corong wadah. Pemilihan motor servo dikarenakan motor servo dapat menggerakkan suatu benda untuk mendapatkan posisi yang diinginkan. Motor servo akan membuka dan menutup corong setelah seluruh proses berjalan selama tiga hari.

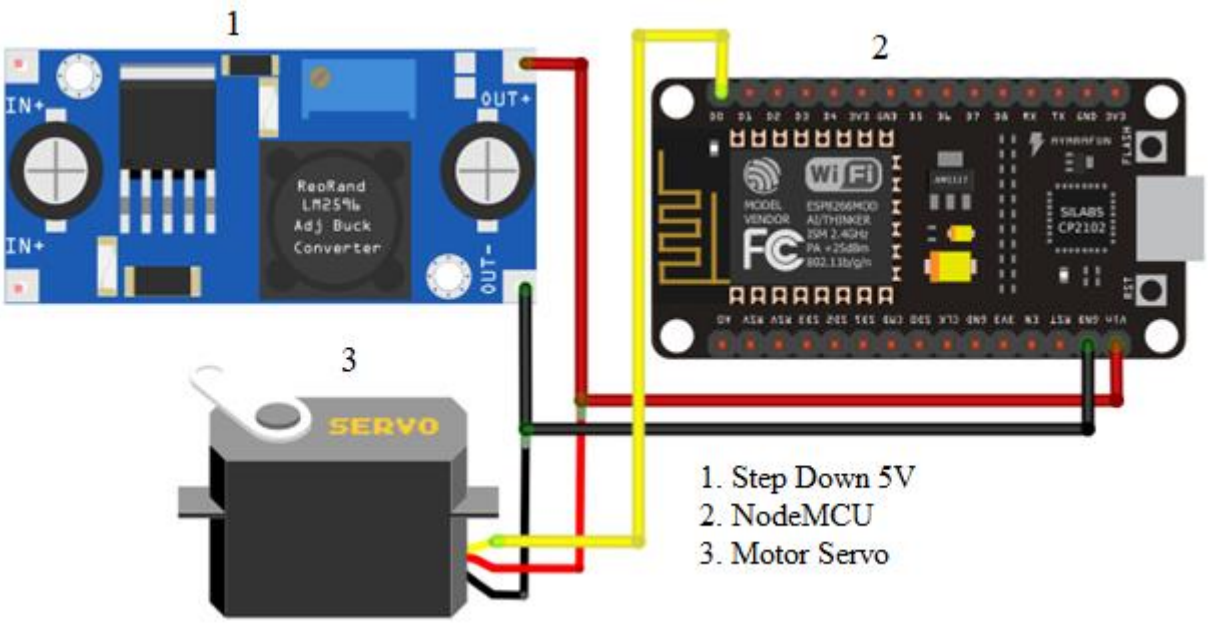
```
#define INTERVAL_MESSAGE1 600000
#include <Servo.h>
Servo servoku;

unsigned long time_1 = 0;
```

```
if(millis() == time_1 + INTERVAL_MESSAGE1){
    time_1 +=INTERVAL_MESSAGE1;
    print_time(time_1);
    servoku.write(45);
    delay(500);
    servoku.write(90);
    delay(500);
    servoku.write(45);
    delay(500);
```

Gambar 3.7 Program *Timer* Sistem Kendali Pemberian Gula

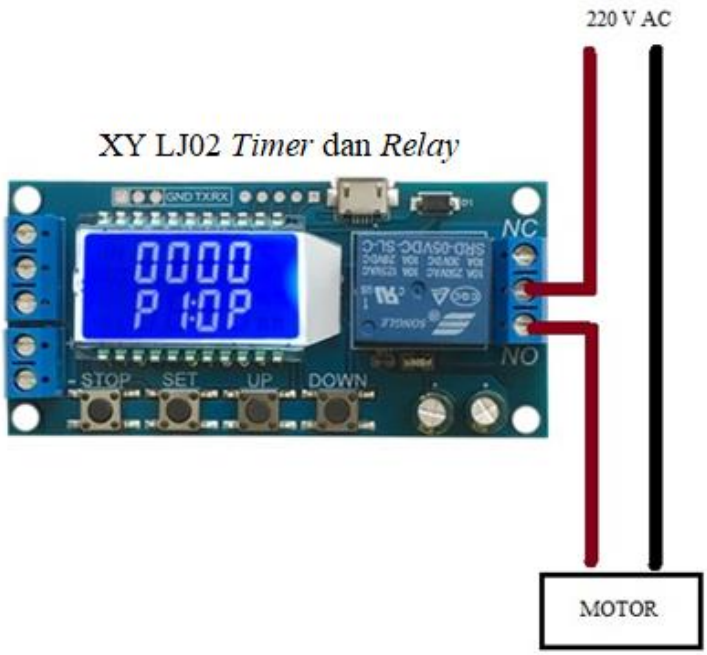
Berdasarkan pada Gambar 3.7 sistem tersebut dirancang untuk menyala setelah waktu sama dengan 3 hari atau 259200000 *mili* detik. Pembacaan waktu pada motor servo menggunakan fungsi *millis* yaitu pencacahan waktu pada *microcontroller* yang setara dengan waktu setiap *mili* detik saat setelah *microcontroller* dinyalakan dan kemudian akan kembali (*overflow*) setelah melewati 50 hari. Setelah waktu pembacaan antara fungsi *millis* sama dengan waktu 3 hari, maka aktuator berupa motor servo akan menyala pada posisi 45^0 untuk kondisi menutup dan 90^0 untuk kondisi membuka. Rangkaian NodeMCU dan motor servo akan ditunjukkan pada Gambar 3.8.



Gambar 3.8 Rangkaian Perangkat Keras Sistem Kendali Pemberian Gula

3.2.5 Perangkat Keras Sistem Pengaduk

Sistem pengaduk akan berjalan berkesinambungan dengan sistem pemberian gula. Tetapi dalam komponen yang berbeda karena menggunakan motor yang berbeda. Untuk sistem pengaduk motor yang digunakan adalah motor AC induksi *single phase* yang diatur *on/off* nya dengan menggunakan *relay*. Pada dasarnya alur kerja dari pengendalian sistem pengaduk sama seperti motor servo, yang membedakan hanya komponen elektriknya. *Relay* dan *timer* yang digunakan adalah XY LJ02 karena pada modul tersebut sudah terdapat *timer* sehingga tidak memerlukan *microcontroller* lagi. Sistem ini akan menyala setiap 3 hari sekali. Rangkaian sistem akan ditunjukkan pada Gambar 3.9.



Gambar 3.9 Rangkaian Perangkat Keras Sistem Pengaduk

Timer XYLJ02 membutuhkan *trigger* yang digunakan sebagai pemicu awal untuk mencacah waktu. *Trigger* tersebut didapat dari NodeMCU yang sama dengan pengendali motor servo. *Trigger* tersebut dimunculkan saat sistem pertama kali mulai dan saat setelah sistem berjalan selama 3 hari. Sehingga motor AC akan bergerak lagi pada hari ke 6. Alur tersebut akan diprogram pada NodeMCU sesuai pada gambar Gambar 3.10.

```
void setup() {
    Serial.begin(115200);
    servoku.attach(16);
    pinMode(led, OUTPUT);
    servoku.write(45);
    delay(500);
    digitalWrite (trig, HIGH);
    delay(500);
    digitalWrite (trig, LOW);
    delay(500);

}
```

```
void loop() {

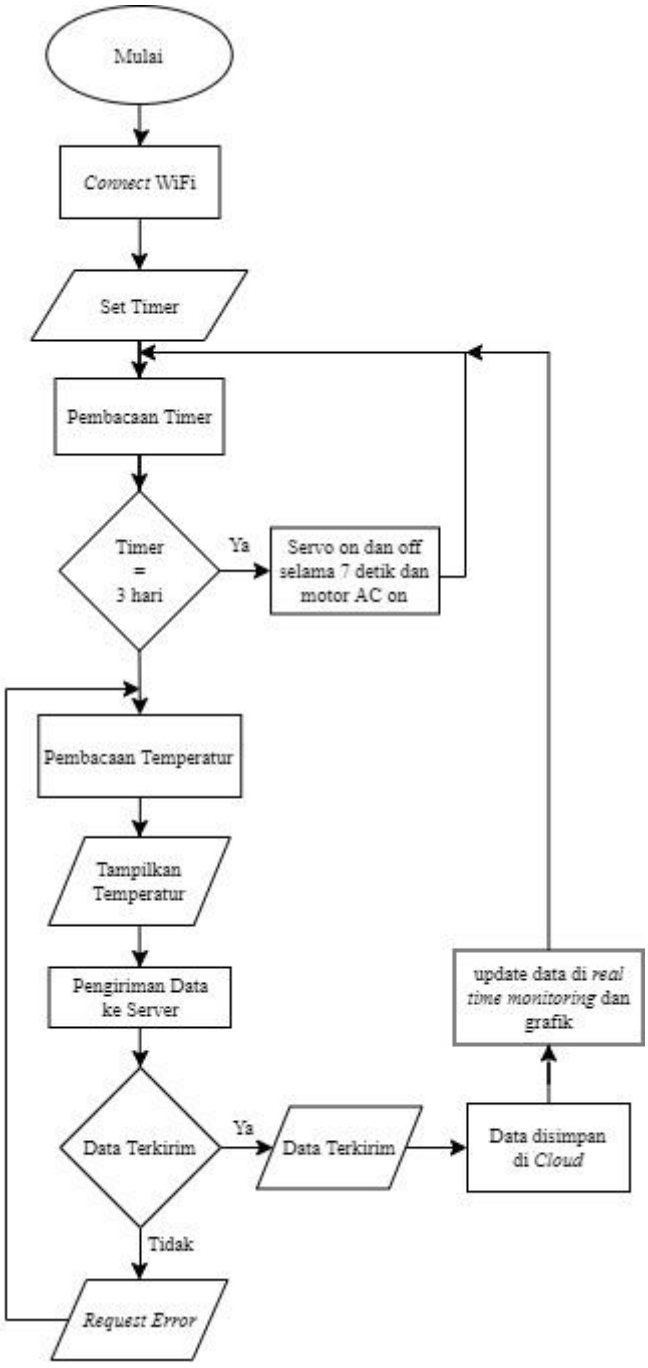
    if(millis() == time_1 + INTERVAL_MESSAGE1){
        time_1 +=INTERVAL_MESSAGE1;
        print_time(time_1);
        servoku.write(45);
        delay(500);
        servoku.write(90);
        delay(500);
        servoku.write(45);
        delay(500);
        servoku.write(90);
        digitalWrite (led, HIGH);
        delay(500);
        digitalWrite (led, LOW);
        delay(500);
        Serial.println("Perintah 1");
    }

}
```

Gambar 3.10 Program *Trigger* pada Modul *Timer* XY LJ02

3.2.6 *Flowchart* Sistem *Monitoring* dan Sistem Pengendali

Alur proses dari seluruh sistem akan dijelaskan pada bagian ini. Perancangan perangkat keras yang telah dilakukan dengan cara mendesain rangkaian elektrik. Perangkat lunak akan berupa aplikasi *smartphone* yang dibuat dengan menggunakan perangkat lunak Android Studio. Berikut ini adalah penjelasan tentang *flowchart* mekanisme kerja seluruh sistem pada Gambar 3.11.



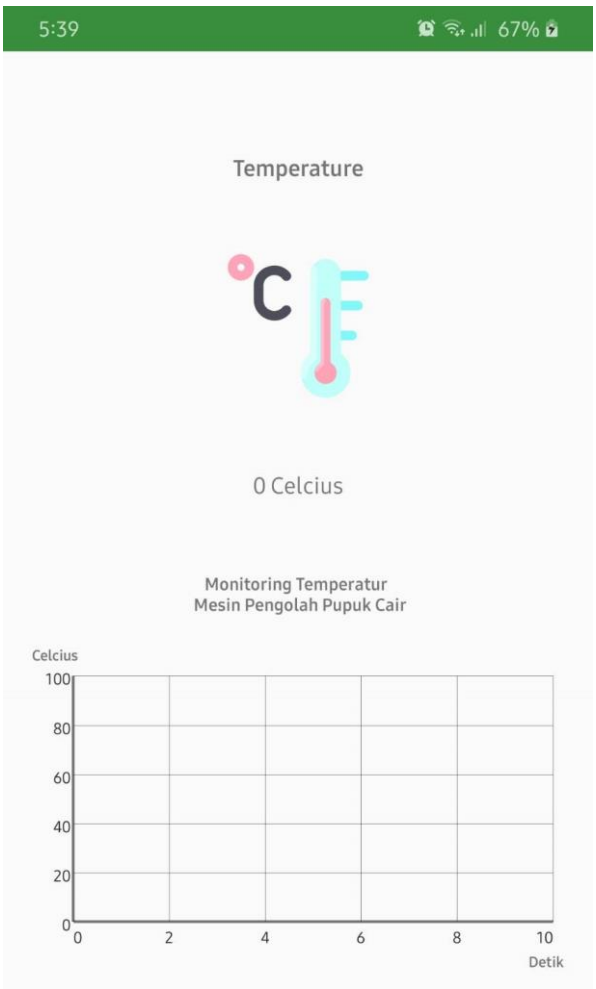
Gambar 3.11 *Flowchart* Sistem

Sistem *monitoring* dan sistem pengendali akan berjalan berkesinambungan tapi berbeda perangkat. Mula-mula sistem *monitoring* akan berjalan dimulai dengan melakukan koneksi dengan

WiFi. Sistem pengendali dimulai dengan melakukan *setting timer* dan jika *timer* sudah sesuai maka akan menyalakan servo dan motor induksi kemudian akan membaca temperatur. Setelah pembacaan tersebut diterima oleh *microcontroller* akan dikirimkan ke *server* yang telah dibuat dan disimpan didalam *Cloud*. Setiap ada perubahan data pada *server*, tampilan pada aplikasi *smartphone* juga akan dirubah pada waktu yang sama. Sistem akan berjalan terus-menerus dan untuk menghentikan sistem hanya perlu mematikan *power supply* yang digunakan.

3.2.7 Perangkat Lunak Aplikasi *Smartphone*

Perancangan sistem perangkat lunak dibuat sebagai aplikasi android yang digunakan untuk menampilkan temperatur secara *real time* dan grafik temperatur pada waktu tertentu. Sistem aplikasi akan mengambil data dari *server* yang telah dibuat. Sehingga saat *server* telah menyimpan data, data tersebut akan langsung tertampil di aplikasi android. Pembuatan aplikasi ini dipilih karena dapat membuat *interface* yang sesuai dengan kebutuhan. Untuk mempermudah pengguna dalam pemantauan proses pembuatan pupuk cair, hal yang paling utama dipantau adalah temperatur. Karena temperatur akan menujukkan adanya kerja bakteri pada proses pembuatan pupuk. Sehingga akan terjadi perubahan temperatur pada waktu tertentu. Untuk mendapatkan tren temperatur dibuatlah grafik sehingga saat sistem menyala maka temperatur akan terbaca dan ditampilkan ke dalam grafik. Perancangan perangkat lunak akan ditunjukkan pada Gambar 3.12.



Gambar 3.12 Perancangan Perangkat Lunak

3.3 Cara Analisis

Analisis akan dilakukan dengan cara uji coba seluruh alat selama proses pembuatan pupuk untuk melihat apakah alat tersebut dapat berjalan dengan baik selama proses tersebut. Kemudian analisis akan dilakukan untuk setiap alat yaitu sistem *monitoring* dan juga sistem pengendali motor. Analisis sistem *monitoring* berupa keakuratan temperatur yang akan diuji pada air yang mempunyai temperatur 30^0C – 80^0C setiap kelipatan 10^0C dan kemudian dibandingkan dengan menggunakan temometer alkohol dengan toleransi pembacaan $\pm1^0$C dan juga pembacaan temperatur harus tersimpan juga di dalam *server*. Analisis sistem pengendali akan dilakukan dengan cara uji coba apakah dengan *timer* yang telah diberikan dan dibandingkan menggunakan *stopwatch*, sistem *on*/*off* akan berjalan atau tidak dan jika terdapat waktu keterlambatan maka akan dianalisis menjadi waktu *delay* sistem.

BAB 4
HASIL DAN PEMBAHASAN

4.1 Mesin Pembuat Pupuk Cair

Mesin pupuk cair yang sudah ada digunakan untuk memudahkan pengguna dalam sistem pengadukan pupuk. Mesin tersebut terdiri dari motor AC dan kipas pengaduk. Mesin tersebut akan dipadukan dengan sistem elektrik pada penelitian ini agar seluruh proses menjadi otomatis.



Gambar 4.1 Mesin Mekanik Pembuat Pupuk Cair

Berdasarkan Gambar 4.1 mesin mekanik tersebut terdapat sedikit perbedaan dari desain sistem pada Gambar 3.3. Pada desain tidak terdapat katrol, tujuan katrol diberikan untuk memperkecil

gaya yang digunakan oleh motor AC. Motor AC pada mesin ini akan disambungkan dengan perangkat elektrik berupa *relay* sehingga kerja *on* dan *off* dapat diatur secara otomatis.

4.2 Pengujian Sistem *Monitoring* dan Perangkat Lunak *Monitoring*

4.2.1 Hasil Kalibrasi

Sebelum sensor terpasang pada mesin pembuat pupuk cair, terlebih dahulu dikalibrasi untuk mengurangi nilai *error* yang dihasilkan. Berdasarkan persamaan (3.4), persamaan regresi linear yang dihasilkan telah dimasukkan ke dalam program yang menjadi pengali sebelum nilai temperatur ditampilkan. Pembacaan temperatur dianalisis dengan cara menguji pembacaan yang sudah terkalibrasi selama sepuluh detik di air yang temperaturnya berubah-ubah. Hasil pembacaan temperatur akan ditunjukkan pada Tabel 4.1.

Tabel 4.1 Pembacaan Temperatur Setelah Kalibrasi

Pembacaan Temperatur (^{0}C)			
Sensor	**Termometer Alkohol**	***Error***	**Rata – rata *error***
30,7	30	0,7	0,55
39,3	40	0,7	
49,9	50	0,1	
59,2	60	0,8	
69,9	70	0,1	
79,1	80	0,9	

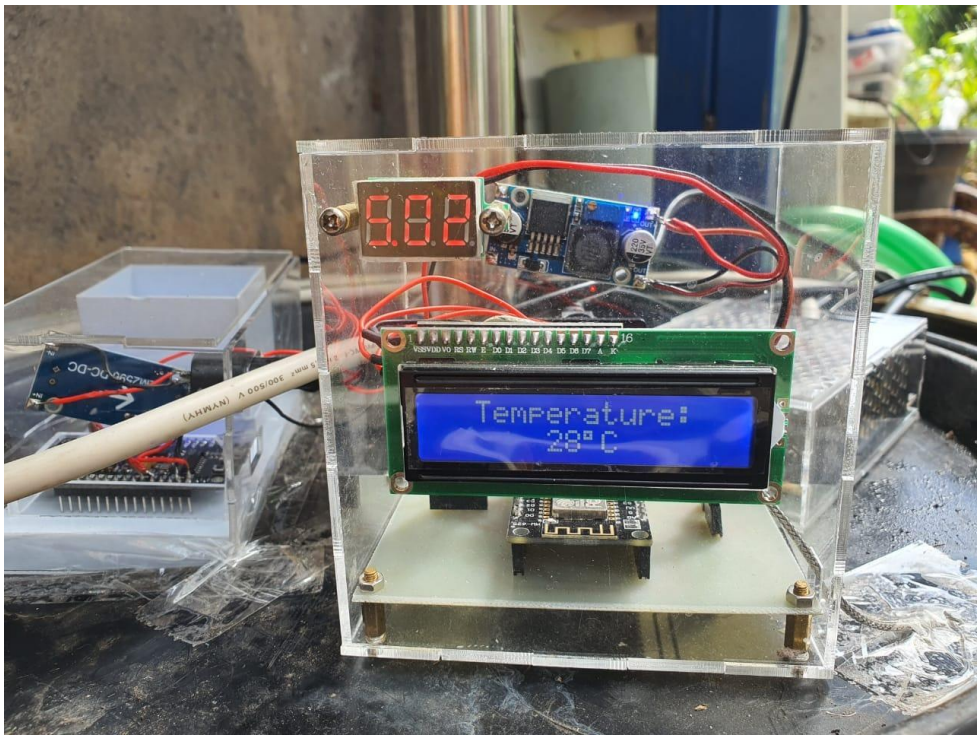
Pengambilan data dilakukan dengan rentang temperatur antara 30 – 80 ^{0}C dan setiap temperatur diambil sampel sebanyak 10 data dalam waktu 10 detik dan dilakukan sekali. Perhitungan *error* dilakukan dengan rumus (4.1).

$$\boldsymbol{error = |aktual - pembacaan|} \tag{4.1}$$

Nilai *error* setelah dirata – rata menghasilkan 0,55. Sehingga akurasi pembacaan yang dihasilkan oleh sensor tersebut adalah 93,12%. Sehingga saat pembacaan sensor mencapai suatu nilai, nilai tersebut mempunyai interval *error*. Interval *error* didapat melalui perhitungan standar deviasi menggunakan Matlab. Hasil standar deviasi yang dihasilkan menggunakan program Matlab adalah 0,3. Sehingga untuk interval *error* yang dihasilkan saat sensor melakukan pembacaan adalah $\pm 0{,}3^0$C . Sensor yang digunakan berupa termokopel tipe K yang mempunyai perubahan pembacaan 1^0C. Oleh karena itu pembacaan temperatur oleh sensor mempunyai interval yang sudah lebih kecil dari $\pm 1^0$C [12].

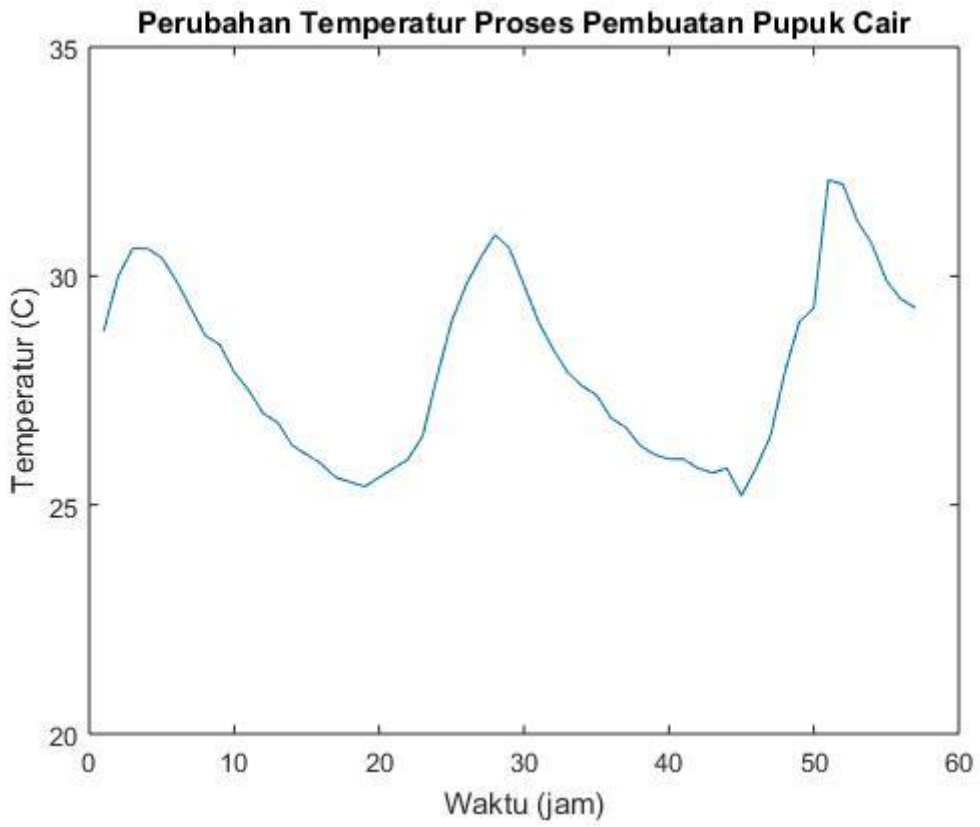
4.2.2 Hasil Pengujian Sistem *Monitoring*

Pada bagian ini, sistem *monitoring* akan diuji apakah sistem dapat berjalan ketika dipasang pada mesin pembuat pupuk cair dan apakah data yang terbaca oleh sensor dapat dikirim ke *server*. Sistem *monitoring* dipasang pada tutup tangki pupuk cair dengan sensor tercelup pada tangki. Sistem akan berjalan selama satu minggu seperti sistem pengendali motor AC dan motor servo. Berikut ini adalah gambar sistem *monitoring* yang sudah terpasang pada tangki pupuk cair



Gambar 4.2 Sistem *Monitoring* (Terpasang)

Berdasarkan Gambar 4.2 nilai temperatur ditampilkan pada LCD 16x2 dan juga akan dikirim ke *server* yang sudah dibuat. Terdapat indikator tegangan masukan untuk NodeMCU karena apabila tegangan melebihi kapasitas masukan dari *microcontroller* akan menyebabkan kerusakan. Data yang dikirimkan akan diterima oleh *server*. Apabila data telah diterima oleh *server* akan dilanjutkan ke aplikasi *monitoring* yang sudah terpasang pada *smartphone* pengguna. Setelah melakukan pengujian pada proses pembuatan pupuk cair, sistem *monitoring* telah membaca sensor selama 4 hari dengan interval setiap 3 detik. Perubahan tersebut akan digambar kan dalam bentuk grafik pada Gambar 4.3.

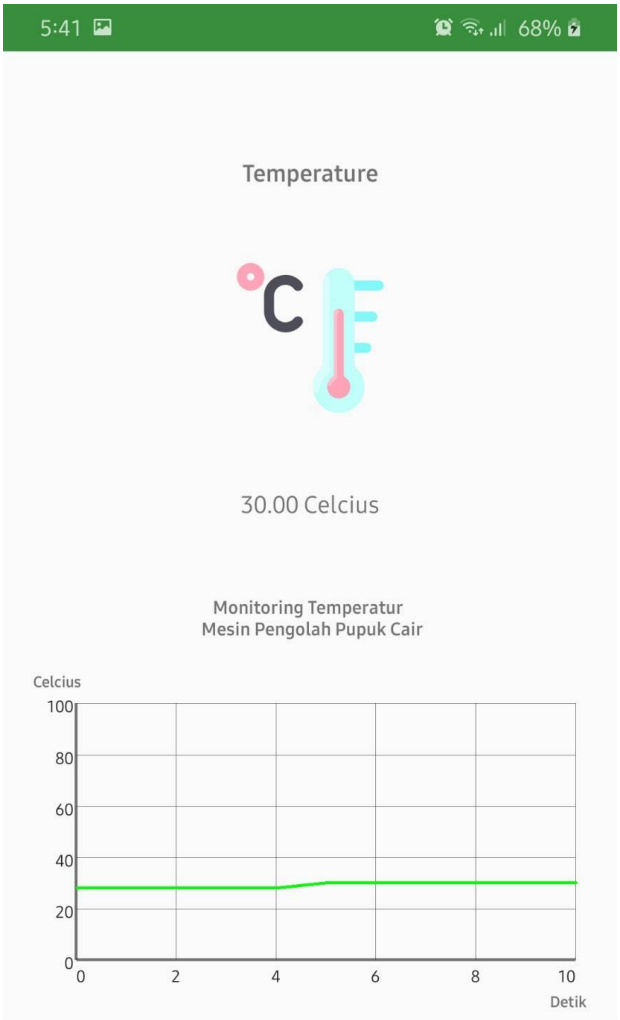


Gambar 4.3 Perubahan Temperatur Proses Pembuatan Pupuk Cair

Data pada Gambar 4.3 diambil dari *server* dan diolah menggunakan komputer. Pengambilan data dimulai pada jam 11 dan data tersebut merupakan rata – rata dari setiap jam pengambilan data. Tetapi data yang telah disimpan dalam bentuk setiap 3 detik. Data pada jam ke-1 hingga ke-24 merupakan data temperatur pada tanggal 1 Juli 2020. Kemudian jam ke-25 hingga ke-49 data pada tanggal 2 Juli 2020 dan jam ke-50 hingga jam ke-60 data pada tanggal 3 Juli 2020. Perubahan temperatur akan meningkat pada jam 8 pagi hingga jam 3 sore setiap harinya dengan maksimal temperatur 32^0C yang merupakan jam ke-52 yaitu jam 12 hingga jam 1 siang pada tanggal 3 Juli 2020. Hal tersebut menandakan adanya aktivitas bakteri pengurai yang bekerja. Kemudian temperatur akan menurun pada jam 4 sore hingga 7 pagi dengan temperatur terendah berada pada 25^0C yang merupakan jam ke-45 yaitu jam 6 hingga 7 pagi pada tanggal 2 Juli 2020. Turunnya temperatur menandakan bahwa bakteri sedang tidak aktif. Dari perubahan tersebut menunjukkan adanya perubahan aktivitas bakteri pada siang hingga ke malam hari. Dengan adanya Gambar 4.3 dapat diketahui perubahan temperatur yang menandakan aktivitas bakteri pada pembuatan pupuk cair.

4.2.3 Hasil Pengujian Aplikasi *Smartphone*

Aplikasi yang digunakan untuk memonitor temperatur secara *real time* dan juga terdapat grafik yang dapat menunjukkan perubahan temperatur tetapi tidak secara utuh. Setiap ada perubahan data yang berhasil dikirim ke *server*, aplikasi *smartphone* akan langsung melakukan *update* dan juga menambahkan data yang baru ke dalam grafik. Pada aplikasi android, apabila data yang lama ingin dilihat kembali, grafik dapat di *zoom out* dan digeser ke data tertentu. Gambar 4.4 menunjukkan aplikasi saat sedang berjalan.



Gambar 4.4 Aplikasi Android (*running*)

4.3 Pengujian Sistem Pengendali Pemberian Gula

Pengujian sistem pengendali dilakukan dengan cara mecocokkan antara *timer* yang diberikan dengan waktu *real-time* apabila sistem sudah dijalankan. NodeMCU yang sudah disambungkan dengan motor servo SG-90 yang digunakan untuk membuka dan menutup saluran untuk mengeluarkan gula. Alat tersebut akan ditunjukkan pada Gambar 4.5.



Gambar 4.5 Perangkat Pemberian Gula

NodeMCU pada sistem ini terpisah dengan sistem *monitoring*. Walaupun pin yang berada pada sistem *monitoring* masih cukup, tetapi sistem pewaktuan yang ada pada NodeMCU sistem *monitoring* akan mengganggu sistem pewaktuan pada sistem pengendali gula. Jika gabungan sistem tersebut dijalankan sistem pewaktuan akan selalu *reset* ke 0 detik sehingga sistem pengendali tidak akan mencapai waktu 259200000 ms. NodeMCU sistem pengendali diprogram

dengan menggunakan konsep *millis* yang mempunyai perhitungan angka tersendiri pada *microcontroller* sebagai penanda waktu. Sehingga *millis* akan diatur sebagai pewaktuan dengan satuan *mili second* yang akan menjadi interval untuk melakukan suatu proses dengan *if-else*, kemudian mengasumsikan sebuah waktu awalan untuk menandakan bahwa sistem tersebut mulai dari keadaan mati. Sehingga saat nilai dari asumsi waktu dan *millis* sama maka akan menjalankan proses buka tutup saluran dengan motor servo. Pada pin Vin NodeMCU dipasang *Step Down* agar sistem tersebut dapat menyala dari *power supply* yang digunakan. Sebelum masuk ke Vin tegangan diturunkan menjadi 5V dengan menggunakan *Step Down* LM2569. Setelah program dijalankan pengujian dilakukan untuk mendapatkan akurasi dari waktu saat sistem menyala dengan waktu saat menjalankan proses terhadap *real-time* akan dilakukan dengan variasi pada interval *millis*. Pada proses sesungguhnya waktu yang digunakan adalah 259200000 *ms* atau 3 hari setelah sistem berjalan. Oleh karena itu pengukuran waktu akan dilakukan dengan *stopwatch*.

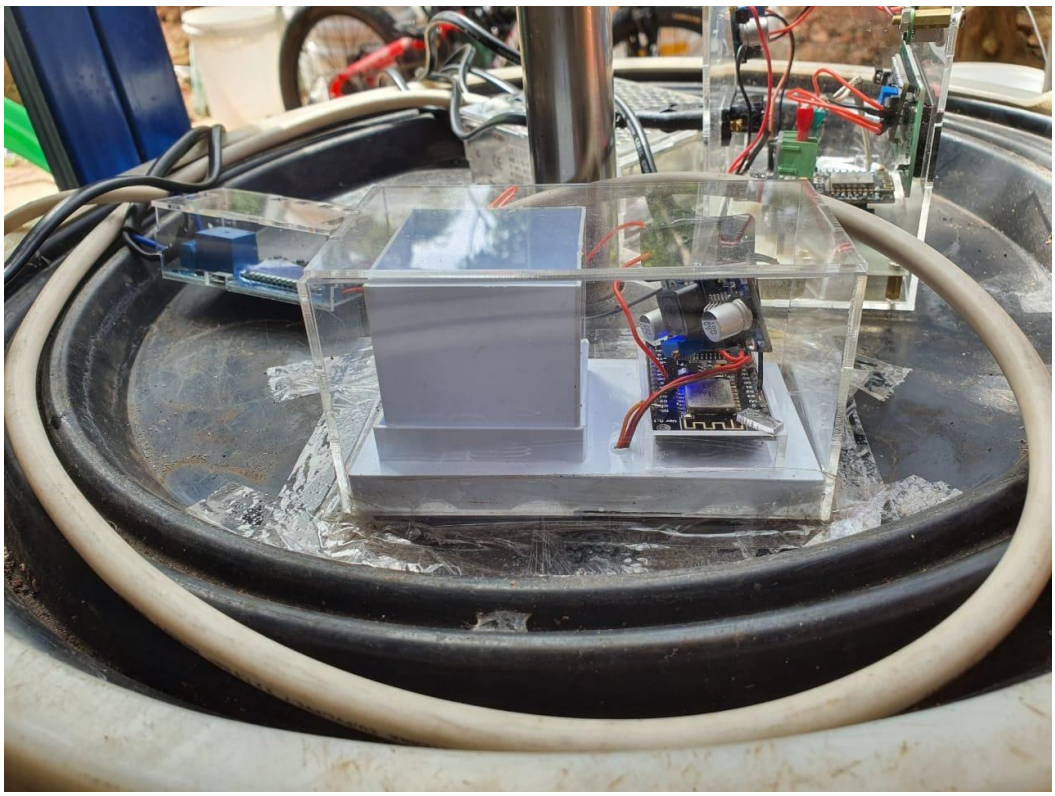
Tabel 4.2 Perbandingan Waktu Antara *Delay* Terhadap *Real Time* Motor Servo

***Delay* (s)**	***Real Time* (s)**			**Rata - rata**	**Waktu Proses (s)**			**Rata - rata**	***Error* (%)**
	1	**2**	**3**		**1**	**2**	**3**		
60	60	60	60	60	7	7	7	7	0
300	300	300	300	300	7	7	7	7	0
600	600	600	600	600	7	7	7	7	0
1800	1800	1800	1800	1800	7	7	7	7	0
3600	3600	3600	3600	3600	7	7	7	7	0

Data *real time* didapatkan dengan menggunakan *stopwatch* dan setiap pengujian dilakukan pengualan sebanyak 3 kali. Berdasarkan Tabel 4.2 waktu *delay* divariasikan untuk mendapatkan akurasi *error* dari berbagai macam waktu *delay*. *Error* dihitung dengan menggunakan rumus (4.2).

$$\%\boldsymbol{Error} = \left|\frac{\boldsymbol{real\ time - delay}}{\boldsymbol{delay}}\right| \times \mathbf{100}\% \qquad (4.2)$$

Untuk setiap waktu pengujian dihasilkan nilai *error* 0%. Hasil tersebut didapat karena penggunaan pewaktuan menggunakan *microcontroller* NodeMCU dengan konfigurasi *millis*. Sistem tersebut akan mencacah waktu setiap *mili* detik saat sistem dinyalakan hingga waktu yang telah ditentukan. Sehingga saat waktu *real time* sama dengan *delay* proses akan langsung berjalan selama 7 detik. Waktu proses akan berlangsung selama 7 detik karena hanya akan mengeluarkan gula sebanyak 3 sendok makan. 7 detik tersebut sudah dapat mengeluarkan gula sebanyak 3 sendok makan. Kemudian sistem tersebut dipasang pada mesin mekanik pupuk agar dapat memberikan gula secara otomatis. Alat tersebut dipasang pada tutup tangki pupuk cair. Gambar perangkat tersebut akan ditunjukkan pada Gambar 4.6.



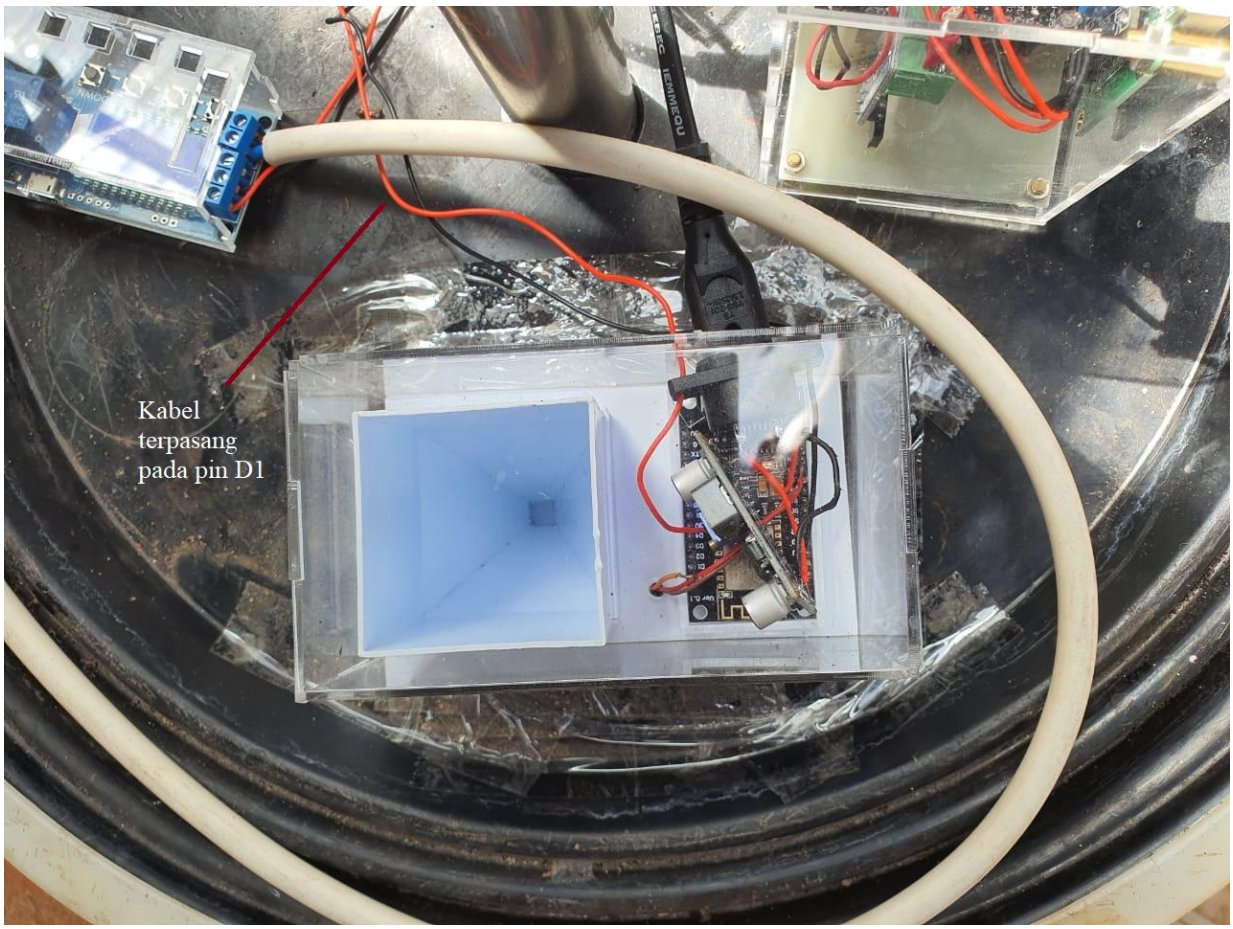
Gambar 4.6 Perangkat Pemberian Gula (Terpasang)

Berdasarkan Gambar 4.6 tutup pada tangki pupuk diberi lubang untuk meletakkan perangkat pemberi gula. Wadah tersebut ditutup akrilik agar gula tidak menjadi basah karena temperatur dan udara dari luar. Perangkat tersebut telah diatur dengan waktu 259200000 *ms* atau tiga hari saat sistem diberikan *supply* tegangan. Kemudian *timer* akan melakukan pengulangan untuk tiga hari kedepan.

Setelah perangkat pemberian gula dipasang pada tangki pupuk cair, selama tiga hari gula yang tersimpan pada wadah tersebut mengkristal karena udara dan temperatur luar, sehingga gula tidak mau turun ke dalam pupuk cair. Untuk mengatasi permasalahan tersebut diperlukan wadah yang tertutup rapat dan juga tidak terkena sinar matahari secara langsung untuk menghindari pengkristalan.

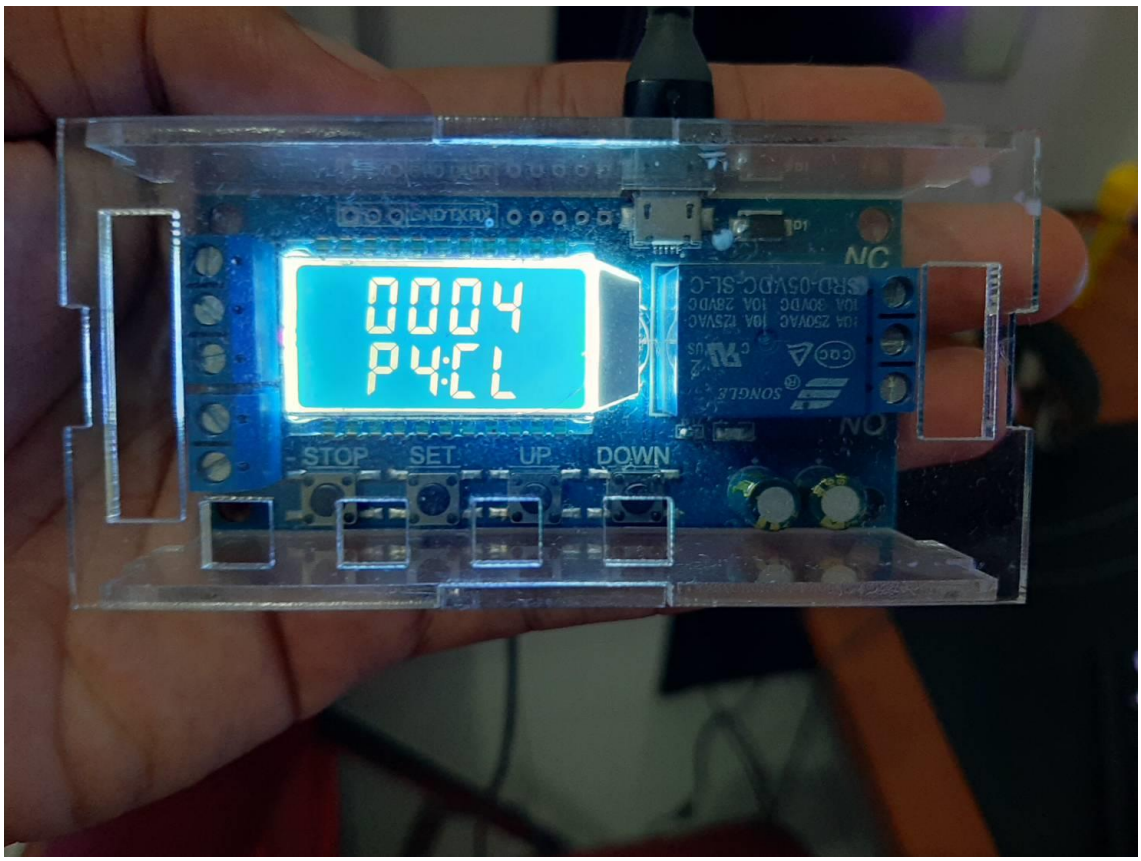
4.4 Pengujian Sistem Pengendali Motor AC

Sistem pengendali motor AC menggunakan *relay* dan *timer* yang dipasang juga pada tutup tangki pupuk cair. Motor AC dikendalikan dengan mengatur *on* dan *off* dari keluaran *relay*. Untuk mengatur kerja *relay* terdapat *timer* yang sudah terpasang pada modul. Tetapi untuk memulai perhitungan waktu dibutuhkan *trigger* sebagai pacuan tegangan awal. Tegangan tersebut didapat dari pin digital pada NodeMCU. Sehingga rangkaian akan berubah seperti Gambar 4.7.



Gambar 4.7 Rangkaian Terpasang Pengendali Motor AC

Pada saat perangkat pengendali motor AC mendapatkan *power supply*, program pada NodeMCU akan langsung memberikan *trigger* untuk menghitung waktu 3 hari. Pada perangkat ini pengaturan waktu dilakukan dalam satuan menit. Ketika *power supply* disambungkan ke perangkat pengendali motor AC akan terdapat pengaturan untuk mengatur fungsi *timer* yang akan digunakan. Untuk setiap waktu yang telah diatur akan selalu tersimpan apabila perangkat mati sehingga tidak perlu mengatur ulang waktu yang digunakan.



Gambar 4.8 Fungsi *Timer* dan *Relay*

Pemilihan perangkat ini juga cocok untuk motor AC yang digunakan karena motor AC yang digunakan menggunakan arus sebesar 3,5 A dan kapasitas dari *relay* yang digunakan sebesar 10 A. Berdasarkan Gambar 4.8 pada panel baris kedua terdapat fungsi P4 yang digunakan untuk menyalakan dan mematikan motor AC. P4 bekerja dengan cara apabila *timer* mendapat *trigger* berupa tegangan, *timer* akan menghitung mundur untuk menyalakan motor AC. Waktu CL diatur untuk mendapatkan *off time* dari *relay* dan kemudian setelah waktu habis akan langsung menyalakan *relay*. Pengujian waktu *relay* dilakukan sama seperti sistem pengendali pemberian gula dengan cara mengatur waktu dengan *delay* tertentu dan dibandingkan dengan acuan *stopwatch*.

Tabel 4.3 Perbandingan Waktu Antara *Delay* Terhadap *Real Time* Motor AC

***Delay* (s)**	***Real Time* (s)**			**Rata - rata**	**Waktu Proses (s)**			**Rata - rata**	***Error* (%)**
	1	**2**	**3**		**1**	**2**	**3**		
4	4	4	4	4	7	7	7	7	0
10	10	10	10	10	7	7	7	7	0
30	30	30	30	30	7	7	7	7	0
60	60	60	60	60	7	7	7	7	0
120	120	120	120	120	7	7	7	7	0

Berdasarkan Tabel 4.3 nilai *error* yang didapat sebesar 0%. Nilai *error* tersebut didapat melalui perhitungan sama dengan persamaan (4.2). Hasil tersebut didapat karena sistem pewaktuan pada XY LJ02 menggunakan *real time clock* (RTC) sehingga perhitungan waktu akan sama dengan hitungan detik pada umumnya. Setelah modul RTC diatur waktu menyala dan matinya maka sistem akan berjalan sesuai dengan waktu yang ditentukan [13].

BAB 5
KESIMPULAN DAN SARAN

5.1 Kesimpulan

1. Sistem *monitoring* yang telah dirancang dan dibuat berhasil melakukan pembacaan data dengan sensor termokopel tipe K dengan akurasi 93,12%. Seluruh pembacaan dikirim ke *server* dan terbaca menggunakan aplikasi android.
2. Sistem pengendali pemberian gula berhasil bekerja tanpa adanya keterlambatan waktu. Hasil tersebut didapat karena penggunakan NodeMCU sebagai sistem pewaktuan dengan fungsi *millis*. Sehingga seluruh waktu yang telah divariasikan berjalan tepat waktu.
3. Sistem pengendali motor AC juga dapat bekerja tanpa adanya keterlambatan karena modul yang digunakan mempunyai RTC tersendiri yang bekerja seperti sistem pewaktuan pada umumnya.

5.2 Saran

1. Sistem pengiriman data menggunakan tambahan modul *General Packet Radio Service* (GPRS), sehingga tidak perlu menggunakan *smartphone* yang harus memberikan koneksi *thetering* dengan jangkauan tertentu.
2. Grafik yang dihasilkan pada aplikasi android dapat disimpan dan juga dapat mencakup semua proses pembuatan pupuk dari awal sampai akhir.
3. Wadah gula didesain lebih tahan panas dan juga dipastikan tidak ada udara yang masuk sehingga tidak menyebabkan gula mengkristal.

DAFTAR PUSTAKA

[1] A. Mulasari, A. H. Husodo, and N. Muhajir, "Situation Analysis of Waste Problem in Yogyakarta," *J. Kesehat. Lingkung. Indones. KEMAS*, vol. 11, no. 2, pp. 98–106, 2016.

[2] A. Gao, Z. Tian, Z. Wang, R. Wennersten, and Q. Sun, "Comparison between the Technologies for Food Waste Treatment," *Energy Procedia*, vol. 105, pp. 3915–3921, 2017.

[3] Y. Nikoloudakis, S. Panagiotakis, T. Manios, E. Markakis, and E. Pallis, "Composting as a Service: A Real-World IoT Implementation," *Futur. Internet*, vol. 10, no. 11, p. 107, 2018.

[4] O. Casas *et al.*, "Wireless sensor network for smart composting monitoring and control," *Meas. J. Int. Meas. Confed.*, vol. 47, no. 1, pp. 483–495, 2014.

[5] H. Soewardi and R. Laksmita, "Development of the Optimum Composition of Organic Liquid Fertilizer," *Int. J. Chem. Eng. Appl.*, vol. 9, no. 3, pp. 88–92, 2018.

[6] M. D. L. Jordão, K. de Paiva, H. T. Firmo, C. T. Inácio, O. C. Rotunno Filho, and T. de A. e Lima, "Low-cost automatic station for compost temperature monitoring," *Rev. Bras. Eng. Agrícola e Ambient.*, vol. 21, no. 11, pp. 809–813, 2017.

[7] M. K. Manu, R. Kumar, and A. Garg, "Drum Composting of Food Waste: A Kinetic Study," *Procedia Environ. Sci.*, vol. 35, pp. 456–463, 2016.

[8] Vrettos G, Kazamias G, and Lekkas DF, "Smart Compost Monitoring System using Open Source Technologies," *15th Int. Conf. Environ. Sci. Techonlogy*, no. September, pp. 1–6, 2017.

[9] Jalaluddin, Z. A. Nasrul, and R. Syafrina, "Pengolahan Sampah Organik Buah-Buahan menjadi Pupuk dengan Menggunakan Effektive Mikroorganisme," *Teknol. Kim. Unimal*, vol. 1, pp. 30–44, 2016.

[10] F. Koyanangi, "NodeMCU ESP8266: Details and Pinout," *Www.Instructables.Com*, 2018. [Online]. Available: https://www.instructables.com/id/NodeMCU-ESP8266-Details-and-Pinout/. [Accessed: 29-Apr-2020].

[11] A. Prasetiyo, "Jenis - Jenis Tipe Termokopel (Thermocouple)," *https://www.duniapembangkitlistrik.com/*, 2018. [Online]. Available: https://www.duniapembangkitlistrik.com/2018/02/jenis-jenis-tipe-termokopel-thermocouple.html. [Accessed: 29-Apr-2020].

[12] MAXIM, "MAX6675 - Cold junction compensated K-Thermocouple to digital converter," *Maxim Dallas*, p. 8, 2002.

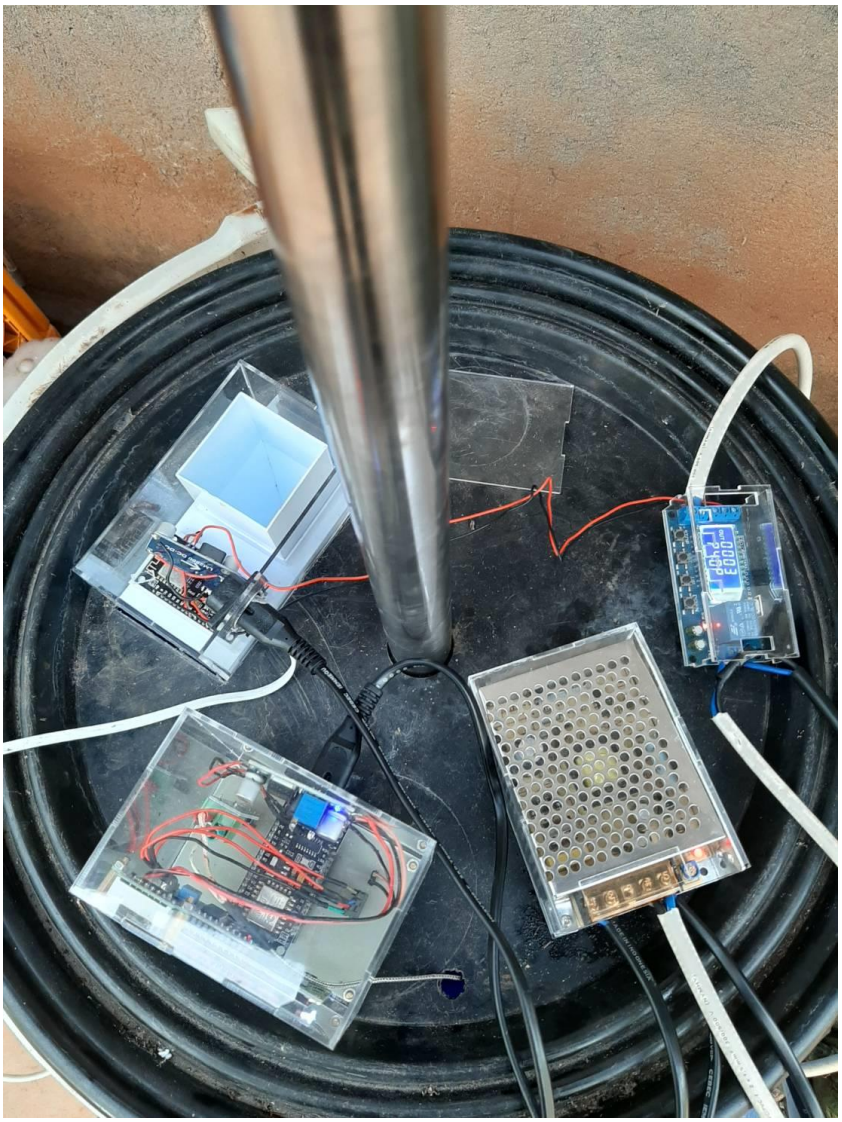
[13] Droking, "XY-LJ02 Manual," *www.droking.com*, 2018. [Online]. Available: https://images-na.ssl-images-amazon.com/images/I/813pHbtLQiL.pdf. [Accessed: 24-

Apr-2020].

[14] M. Chiregi and N. Jafari Navimipour, “Cloud computing and trust evaluation: A systematic literature review of the state-of-the-art mechanisms,” *J. Electr. Syst. Inf. Technol.*, vol. 5, no. 3, pp. 608–622, 2018.

LAMPIRAN

1. Tampilan Alat (tampak atas)



2. Modul gula saat penggunaan



3. Pengujian dan Pengambilan data Regresi

