

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pada saat ini perkembangan restoran cepat saji berkembang sangat pesat. Hal ini dibuktikan dengan mudahnya masyarakat untuk mencari restoran cepat saji dan selalu ramai dikunjungi oleh pengunjung setiap hari. Penyediaan makanan dan minuman dengan waktu yang relatif singkat merupakan salah satu alasan restoran cepat saji digemari oleh konsumen.

Dilansir dari halaman situs Wonderopoli, restoran cepat saji pertama kali berdiri pada tahun 1912 oleh Walter Anderson dan Edgar W. “Billy” Ingram di Wichita, Kansas dalam bentuk restoran White Castle. Berawal dari inovasi Anderson dan Billy, hingga saat ini restoran cepat saji berkembang dan bertambah setiap tahunnya. Di Indonesia, perkembangan restoran cepat saji bertumbuh secara stabil hingga 10% hingga 15% per tahun (Ricard, 2019). Beberapa restoran cepat saji yang paling diminati oleh masyarakat Indonesia menurut survey yang dilakukan oleh Dihni (2022) yaitu KFC, mcdonalds, Hoka-Hoka Bento, A&W dan Richeese Factory.

Restoran cepat saji dikenal dengan menu yang beragam, dengan berbagai pilihan hidangan seperti burger, ayam goreng, kentang goreng, dan lainnya. Saus berperan penting dalam memberikan variasi pada hidangan ini. Misalnya, saus tomat, mayones, saus sambal, atau saus keju sering digunakan untuk menyesuaikan rasa dan preferensi konsumen. Kualitas rasa makanan cepat saji sering dikaitkan

dengan saus yang menyertainya. Saat saus yang sesuai digunakan dengan benar, ini dapat meningkatkan pengalaman makan pelanggan.

Ketersediaan elemen kecil seperti saus tomat pada restoran cepat saji dapat meningkatkan kepuasan konsumen, keinginan untuk datang kembali dan merekomendasikan restoran kepada orang lain (chun & Nyam-Ochir, 2020 ; ong dkk, 2022). Sebagai upaya dalam menjaga ketersediaan saus pada restoran cepat saji, restoran dapat menggunakan alat untuk mendeteksi kapasitas atau ketersediaan saus di dalam tempat saus. Penggunaan teknologi berbasis IoT dengan mikrokontroler seperti ESP8266 dapat membantu pemilik restoran atau karyawan cepat saji memonitoring kapasitas saus secara real-time dan juga dapat meningkatkan efisiensi dalam bekerja.

Penelitian yang dilakukan pada tahun 2021 oleh Priyandoko, Siswanto & Kurniawan menunjukkan bahwa penggunaan perangkat monitoring infus berbasis Internet of Things (IoT) dapat membantu perawat dengan tingkat akurasi sebesar 98,89% untuk mendeteksi ketersediaan air infus pada pasien sesuai dengan volume yang ditetapkan. Penelitian lainnya dengan menggunakan nodemcu ESP8266 efektif untuk memantau level cairan dan memberikan notifikasi melalui aplikasi smartphone secara real-time (darsi dkk, 2023) dan mendeteksi kapasitas cairan secara otomatis, mengurangi kesalahan manusia, dan meningkatkan efisiensi (azhar dkk, 2024).

Perkembangan teknologi dan kebutuhan akan efisiensi dalam mengelola restoran cepat saji, pengembangan alat monitoring berbasis IoT seperti nodemcu ESP8266 menjadi solusi yang sangat relevan untuk menjaga ketersediaan saus secara real-time. Hal ini tidak hanya membantu meningkatkan efisiensi kerja, tetapi

juga memastikan kepuasan pelanggan tetap terjaga, sehingga restoran dapat terus bersaing di industri yang berkembang pesat ini. Berdasarkan uraian diatas, penulis ingin mengembangkan projek mengenai monitoring kapasitas saus pada restoran cepat saji yang berjudul **“Prototype Alat Monitoring Kapasitas Saus Tomat Pada Restoran Cepat Saji Menggunakan Mikrokontroler ESP8266”**.

## **1.2 Perumusan Masalah**

Perumusan masalah merupakan proses merumuskan pertanyaan atau pernyataan yang jelas dan terfokus untuk mengidentifikasi inti dari masalah yang akan dianalisis atau diselesaikan. Berdasarkan konteks yang telah dijelaskan di latar belakang sebelumnya, maka dapat dirumuskan masalah dalam penelitian monitoring kapasitas saus tomat sebagai berikut :

1. Bagaimana Mikrokontroller Arduino Mega 2560 dapat mengontrol sistem monitoring kapasitas saus tomat pada restoran cepat saji secara keseluruhan?
2. Bagaimana NodeMCU ESP8266 dapat menghubungkan sistem monitoring kapasitas saus tomat dengan aplikasi telegram ?
3. Bagaimana sistem monitoring kapasitas saus tomat dapat mendeteksi kapasitas saus menggunakan sensor ultrasonic ?
4. Bagaimana Sensor load cell dapat mengukur berat kapasitas saus yang akan di ambil ?
5. Bagaimana sensor water pump dapat menyedot dan mengalirkan saus pada tempat penampung saus ?

## **1.3 Batasan Masalah**

Batasan masalah merupakan batasan yang menunjukkan ruang lingkup

dari permasalahan agar pembahasan yang disusun tidak terlampaui jauh dan melebar. Agar menghindari terlalu luasnya permasalahan dan pemecahan masalah yang dilakukan dari tujuan yang akan dicapai, maka perlu dibatasi sistem yang dirancang. Batasan-batasan yang diberikan adalah :

1. Mikrokontroler Arduino Mega 2560 digunakan sebagai pengontrol sistem alat monitoring kapasitas saus tomat pada restoran cepat saji.
2. Sensor Ultrasonic digunakan sebagai pendeteksi jarak pada saus tomat untuk menentukan ketersediaan saus tomat.
3. NodeMCU ESP8266 digunakan untuk menghubungkan sistem monitoring kapasitas saus tomat dengan aplikasi telegram.
4. Sensor Load Cell digunakan sebagai pengukur berat saus tomat yang akan di ambil.
5. Sensor ultrasonic digunakan sebagai pendeteksi kapasitas saus tomat pada sistem monitoring kapasitas saus tomat pada restoran cepat saji.

#### **1.4 Hipotesa**

Hipotesis adalah dugaan sementara yang dibuat berdasarkan data atau fakta awal sebagai jawaban atas suatu masalah, yang perlu diuji kebenarannya melalui penelitian. Berdasarkan pada perumusan masalah diatas, penulis dapat megambil beberapa hipotesa, yaitu :

1. Diharapkan Mikrokontroler Arduino Mega 2560 dapat mengontrol sistem monitoring kapasitas saus tomat pada restoran cepat saji secara keseluruhan.
2. Diharapkan NodeMCU ESP8266 dapat menghubungkan sistem monitoring kapasitas saus tomat dengan aplikasi telegram.

3. Diharapkan sistem monitoring kapasitas saus tomat dapat mendeteksi kapasitas saus menggunakan sensor ultrasonic.
4. Diharapkan Sensor load cell dapat mengukur berat kapasitas saus yang akan di ambil.
5. Diharapkan sensor water pump dapat menyedot dan mengalirkan saus pada tempat penampung saus.

### **1.5 Tujuan Penelitian**

Tujuan penelitian adalah hasil atau pencapaian yang ingin diperoleh dari suatu penelitian, seperti menjawab pertanyaan, menguji hipotesis, atau memberikan solusi atas masalah tertentu. Adapun tujuan yang diinginkan dalam pembuatan alat monitoring kapasitas saus tomat pada restoran cepat saji ini adalah sebagai berikut:

1. Menggunakan Arduino Mega 2560 untuk mengontrol sistem monitoring kapasitas saus tomat pada restoran cepat saji bekerja dengan baik.
2. Mengimplementasikan NodeMCU ESP8266 untuk dapat menghubungkan sistem monitoring kapasitas saus tomat dengan aplikasi telegram bekerja dengan baik.
3. Menggunakan sensor ultrasonic untuk mendeteksi kapasitas saus tomat pada restoran cepat saji bekerja dengan baik.
4. Menggunakan Sensor load cell untuk mengukur berat kapasitas saus yang akan diambil berjalan dengan baik.
5. Menggunakan water pump untuk menyedot dan mengalirkan saus pada tempat penampung saus dapat bekerja dengan baik.

## 1.6 Manfaat Penelitian

Manfaat penelitian merupakan serangkaian atau kumpulan kegunaan hasil penelitian, baik bagi kepentingan untuk pengembangan program maupun kepentingan ilmu pengetahuan yang dianggap penting untuk dilakukan. Berdasarkan manfaat penelitian diatas, maka ditentukan manfaat penelitian sebagai berikut :

### A. Bagi Penulis

1. Sebagai syarat bagi penulis untuk mendapatkan gelar sarjana sekaligus untuk dapat menambah pengetahuan di bidang elektronika, komputer dan robotika.
2. Dapat memperluas wawasan dan meningkatkan pengetahuan dalam pemanfaatan komponen-komponen yang digunakan.

### B. Bagi Program Studi

1. Menambah referensi dalam memperbanyak literatur bagi mahasiswa yang berhubungan dengan ESP8266.
2. Meningkatkan reputasi program studi dengan hasil penelitian yang relevan dan inovatif.

### C. Bagi Masyarakat

1. Meningkatkan kepuasan konsumen dan terhindar dari keluhan habisnya saus di tempat saus pada restoran cepat saji.
2. Memudahkan konsumen untuk mengambil saus tomat dengan jumlah sesuai yang diinginkan oleh konsumen.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Konsep Dasar Sistem**

Sistem adalah suatu kerangka dari prosedur-prosedur yang saling berhubungan, yang disusun sesuai dengan skema yang menyeluruh untuk melaksanakan suatu kegiatan atau fungsi utama dari perusahaan yang dihasilkan oleh suatu proses tertentu yang bertujuan untuk menyediakan informasi untuk membantu mengambil keputusan manajemen operasi perusahaan dari hari ke hari serta menyediakan informasi yang layak untuk pihak di luar Perusahaan (M. Arfa Andika Candra, dkk, 2021).

##### **2.1.1 Pengertian Sistem**

Menurut Marakas & O'Brien menyatakan bahwa "Sistem didefinisikan sebagai seperangkat komponen yang saling terkait, dengan batasan yang jelas, yang bekerja sama untuk mencapai tujuan tertentu dengan menerima masukan dan menghasilkan keluaran dalam proses transformasi yang terorganisasi" (Januari, 2021).

##### **2.1.2 Karakteristik Sistem**

Sistem yang mempunyai komponen-komponen, batas sistem, lingkungan sistem, penghubung, masukan, keluaran, pengolah dan sasaran, secara umum sebuah sistem terdiri dari input dan output. Ketiga hal tersebut merupakan konsep

sebuah sistem yang paling sederhana. Suatu sistem dapat dikatakan sebagai sistem yang baik

apabila memiliki karakteristik-karakteristik tertentu. Sistem mempunyai karakteristik sebagai berikut :

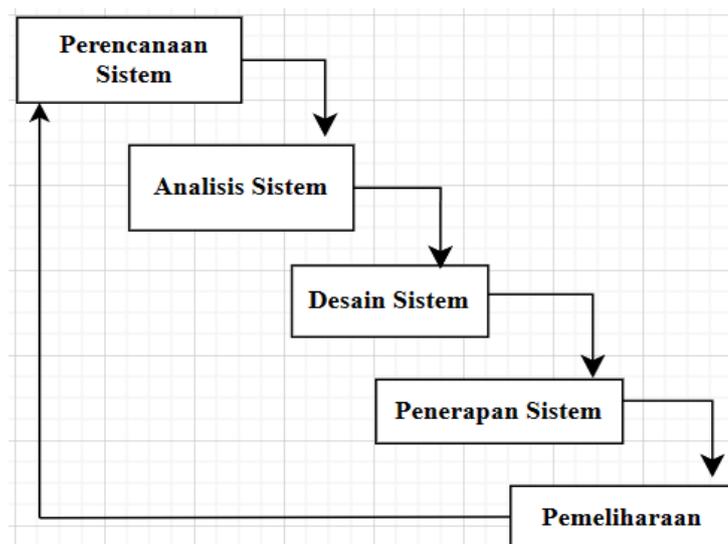
1. Komponen atau elemen (*Components*) suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Jadi berjalan/berfungsi sebagaimana mestinya. Tentunya sistem tersebut tidak akan berjalan mulus atau mungkin juga sistem tersebut rusak sehingga dengan sendirinya tujuan sistem tersebut tidak tercapai.
2. Batas sistem (*Boundary*) merupakan daerah yang membatasi antara suatu system dengan system yang lainya atau dengan lingkungan luarnya. Batas system ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu system menunjukkan ruanglingkup (*scope*) dari system tersebut.
3. Lingkungan Luar Sistem (*Environments*) adalah apapun di luar dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menggantungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menggantungkan merupakan energi dari sistem dan dengan demikian harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung (*Interface*) Sistem merupakan media penghubung antara suatu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke yang lainnya. Keluaran (output) dari satu subsistem akan menjadi masukan (input) untuk subsistem lainnya dengan melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.
5. Masukan (*Input*) Sistem adalah energi yang dimasukkan kedalam sistem. Masukan dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signalinput). Maintenance input adalah energy yang dimasukkan supaya system tersebut dapat beroperasi. Signal input adalah energy yang diproses untuk didapatkan keluaran. Sebagai contoh didalam system komputer, program adalah maintenance input yang digunakan untuk mengoperasikan komputernya dan data adalah signal input untuk di olah menjadi informasi.
6. Keluaran (*Output*) Sistem adalah hasil dari energy yang di olah dan diklasifikasikan menjadi keluaran yang berguna dan pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada super sistem. Misalnya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.
7. Pengolah (*Process*) Suatu system dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. Suatu system produksi akan menolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi. Sistem akuntansi akan mengolah data-data transaksi menjadi laporan- laporan lain yang di butuhkan oleh manajemen.

8. Sasaran (*Objektives*) atau Tujuan (*Goal*) Suatu sistem pasti mempunyai tujuan atau sasaran. Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

### 2.1.3 Siklus Hidup Pengembangan Sistem (System Development Life Cycle)

*System Development Life* (SDLC) merupakan siklus hidup pengembangan sistem yang terdiri dari beberapa tahapan dalam membangun perangkat lunak, dapat dilihat pada Gambar 2.1. Berikut tahapan-tahapan SDLC yang dilakukan pada penelitian ini :



**Gambar 2. 1 Tahapan-Tahapan Dalam SDLC**

*Sumber: Darmanta Sukrianto, dkk, 2020*

#### a. Perencanaan Sistem (System Planning)

Langkah-langkah perencanaan yaitu menyadari adanya masalah, mengidentifikasi masalah, dan menentukan tujuan sistem. Selanjutnya mendefinisikan kebutuhan informasi seperti dengan melakukan pengamatan

secara langsung bagaimana sistem kinerja Monitoring Ketersediaan Saus Tomat dapat berguna dengan baik.

b. Analisis sistem

Tahapan pada proses pengumpulan data-data kebutuhan sistem dari pemecahan identifikasi masalah yang timbul untuk di analisa dalam pembuatan aplikasi yang diinginkan dengan melihat bagaimana prosedur pada Monitoring Ketersediaan Saus Tomat yang tepat untuk digunakan.

c. Desain Sistem

Malakukan penggunaan bahasa pemograman C dengan program Arduino pada Monitoring Ketersediaan Saus Tomat.

d. Penerapan

Dalam tahap ini dilakukan dengan menerapkan penggunaan alat Monitoring Ketersediaan Saus Tomat dalam kinerja untuk memantau ketersediaan saus tomat.

e. Pemeliharaan

Pemeriksaan secara berkala/periodik, memperkaya atau mengembangkan sistem dengan penambahan fitur-fitur baru yang dapat meningkatkan kinerja sistem (Darmanta Sukrianto, dkk, 2020).

## 2.2 Konsep Dasar Monitoring

monitoring merupakan suatu langkah untuk mendapatkan data secara berkala yang berguna untuk mendeteksi progres dan peningkatan dari suatu projek. hal ini dilakukan dari awal pelaksanaan program agar memastikan segala sesuatu yang diinginkan sesuai dengan rencana (LPKN,2023).

### 2.2.1 Pengertian Monitoring

Monitoring alat mengacu pada kegiatan pengawasan dan pengukuran kinerja suatu alat atau perangkat secara terus-menerus untuk memastikan bahwa alat tersebut berfungsi sesuai dengan standar atau spesifikasi yang ditentukan. Tujuan dari monitoring alat adalah untuk mendeteksi anomali atau masalah kinerja dengan cepat, memastikan keandalan, dan mencegah kegagalan alat yang dapat berdampak pada operasi atau hasil yang diinginkan.

### 2.2.2 Saus Tomat

*Ketchup* adalah sebutan saus tomat dalam berbahsa Inggris, sedangkan di Indonesia dikenal dengan nama saus tomat. Saus tomat telah dikenal dari lama oleh masyarakat Asia yang dikenal dengan berbagai sebutan, seperti *koe-chiap* dalam dialek Amoy di Tiongkok dan *kechap* dalam bahasa Melayu. Racikan saus tomat tersebar di sepanjang jalur perdagangan Indonesia Filipina dan disukai oleh para pedagang dari Inggris (Butler, 2012).

Pada abad-19 merupakan tahun keemasan bagi saus tomat. Pada masa ini resep pembuatan saus tomat terdiri dari beberapa komponen, diantaranya tiram, kerang, jamur, kenari, lemon, seledri dan juga terdiri dari beberapa buah-buahan seperti plum dan persik. Semua komponen direbus hingga berubah tekstur menjadi sirup dan selanjutnya didiamkan bersama garam hingga tercipta rasa pedas dan asin. Resep pembuatan saus tomat pertama kali di temukan pada sebuah buku yang berjudul *Sugar House Book* pada tahun 1801 di Amerika. Selanjutnya, pada tahun 1812, resep saus tomat pertama kali diperkenalkan oleh James Mease (Rozin, 1994). Seiring dengan berjalannya waktu, saus tomat menjadi populer di Amerika.

Di lansir dari situs Tomato Gardening Guru saus tomat populer lebih lama daripada salad tomat.

Jonas Yerkes dianggap sebagai orang Amerika pertama yang mengenalkan saus tomat dalam kemasan botol ke masyarakat (Skrabec, 2019). Pada tahun 1837, Jonas Yerkes membuat dan menjual saus tomat kepada masyarakat dengan bumbu tradisional. Sehingga, tak lama kemudian beberapa perusahaan mengikuti ide tersebut, salah satunya yaitu F. &J. Heinz. Dilansir dari situs HPSauce.co.uk perusahaan F.&J. Heinz meluncurkan saus tomat pertamanya pada tahun 1876. Saus tomat yang diluncurkan oleh perusahaan F.&J. Heinz terus berkembang mulai di ekspor ke UK hingga berbagai negara.

### **2.2.3 Restoran Cepat Saji**

Restoran cepat saji merupakan restoran dengan pelayanan yang cepat atau quick service restaurant, dengan sedikitnya pelayanan pada meja pelanggan. Dilansir dari halaman situs Wonderopoli, restoran cepat saji pertama kali berdiri pada tahun 1912 oleh Walter Anderson dan Edgar W. “Billy” Ingram di Wichita, Kansas dalam bentuk restoran White Castle. Berawal dari inovasi Anderson dan Billy, hingga saat ini restoran cepat saji berkembang dan bertambah setiap tahunnya. Di Indonesia, perkembangan restoran cepat saji bertumbuh secara stabil hingga 10% hingga 15% per tahun (Ricard, 2019). Beberapa restoran cepat saji yang paling diminati oleh masyarakat Indonesia menurut survey yang dilakukan oleh Dihni (2022) yaitu KFC, McDonalds, Hoka-Hoka Bento, A&W dan Richeese Factory.

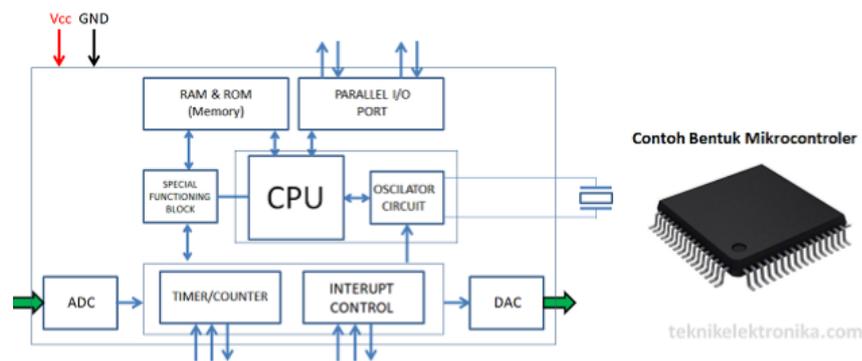
Pada saat ini perkembangan restoran cepat saji berkembang sangat pesat. Hal ini dibuktikan dengan mudahnya masyarakat untuk mencari restoran cepat saji dan selalu ramai di kunjungi oleh konsumen setiap hari. Alasan restoran cepat saji digemari oleh beberapa masyarakat Indonesia adalah karena dapat menyediakan makanan atau minuman dengan waktu yang relatif singkat. Selain itu, hal lainnya yang mempengaruhi perkembangan restoran cepat saji berkembang pesat juga karena makanan yang digemari oleh semua golongan masyarakat dan memiliki variasi menu yang beragam, mulai dari menu sarapan, makan berat, cemilan, makanan penutup dan minuman.

Restoran cepat saji dikenal dengan menu yang beragam, dengan berbagai pilihan hidangan seperti burger, ayam goreng, kentang goreng, dan lainnya. Saus berperan penting dalam memberikan variasi pada hidangan ini. Misalnya, saus tomat, mayones, saus sambal, atau saus keju sering digunakan untuk menyesuaikan rasa dan preferensi konsumen. Kualitas rasa makanan cepat saji sering dikaitkan dengan saus yang menyertainya. Saat saus yang sesuai digunakan dengan benar, ini dapat meningkatkan pengalaman makan pelanggan.

#### **2.2.4 Mikrokontroler**

Mikrokontroler adalah komputer mini yang terintegrasi dalam bentuk chip IC (Integrated Circuit) dan dirancang untuk melaksanakan tugas atau operasi spesifik. Pada dasarnya, sebuah IC Mikrokontroler terdiri dari satu atau lebih inti prosesor (CPU), memori (RAM dan ROM), serta perangkat input dan output yang dapat diprogram. Dalam aplikasinya, mikrokontroler digunakan dalam produk atau perangkat yang dikendalikan secara otomatis, seperti sistem kontrol mesin mobil dan perangkat medis, pengendali jarak jauh, mesin, peralatan listrik, mainan dan

perangkat-perangkat yang menggunakan sistem tertanam lainnya (Junaedi, Puspitasari, & Maulidina, 2021). Diagram blok dan struktur mikrokontroler dapat dilihat pada Gambar 2.2.



**Gambar 2. 2 Diagram Blok dan Struktur Mikrokontroler**

*Sumber : Junaedi, Puspitasari, & Maulidina, 2021.*

## 2.3 Alat Bantu Dalam Perancangan Sistem Dan Logika Program

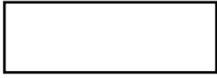
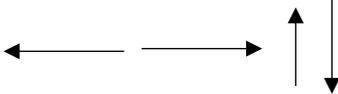
Sebagai upaya untuk melakukan suatu proses penganalisaan terhadap suatu sistem, maka sebagai aturan dasar yang harus dilakukan adalah mendefinisikan secara menyeluruh sistem yang akan dirancang. Hal ini mengandung arti bahwa harus ada gambaran yang jelas mengenai ruang lingkup tentang sistem yang dibahas. Media yang digunakan untuk menggambarkan sistem tersebut adalah Context Diagram, Data Flow Diagram (DFD), dan Flowchart.

### 2.3.1 Context Diagram

CD (Context Diagram) menampilkan gambaran keseluruhan dari sistem yang dirancang, di mana semua entitas eksternal harus digambarkan dengan jelas sehingga aliran data pada input, proses, dan output terlihat. CD menggunakan tiga

simbol: satu untuk entitas eksternal, satu untuk aliran data, dan satu untuk proses. CD hanya boleh memiliki satu proses saja dan tidak boleh lebih dari itu, serta tidak menggambarkan data store. Proses dalam CD biasanya tidak diberi nomor. Simbol context diagram dapat dilihat pada Tabel 2.1.

**Tabel 2. 1 Simbol Context Diagram**

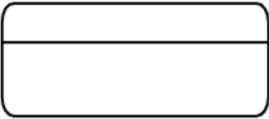
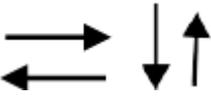
Simbol Context Diagram	Arti Khusus
	Sistem
	Entitas luar
	Arah Aliran

*Sumber: Mrizky, 2020.*

### 2.3.2 Data Flow Diagram (DFD)

Data Flow Diagram (DFD), juga dikenal sebagai Diagram Arus Data (DAD), adalah model logis yang menggambarkan aliran data atau proses dalam sistem. DFD menunjukkan asal data, tujuan data yang keluar dari sistem, lokasi penyimpanan data, proses yang menghasilkan data, serta interaksi antara data yang disimpan dan proses yang diterapkan pada data tersebut. Simbol data flow diagram dapat dilihat pada Tabel 2.2.

Tabel 2. 2 Simbol Data Flow Diagram

Simbol	Nama Simbol	Keterangan
	Kesatuan luar ( <i>eksternal entity</i> )	Simbol ini merupakan sumber yang berupa data bagian atau orang yang berada di luar sistem, tapi berhubungan dengan sistem lainnya, baik itu memasukkan data maupun mengambil data dari sistem
	Proses ( <i>Process</i> )	Simbol ini digunakan untuk suatu proses pengolahan data untuk menunjukkan sebuah kegiatan yang mengubah aliran data masuk ( <i>input</i> ) menjadi aliran data keluar ( <i>output</i> ).
	Penyimpanan Data ( <i>data storage</i> )	Simbol ini berfungsi sebagai tempat penyimpanan dokumen atau file yang dibutuhkan dalam suatu sistem informasi.
	Aliran data ( <i>data flow</i> )	Simbol ini menunjukkan arus dalam proses dimana simbol aliran data ini mempunyai nama sendiri

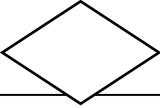
Sumber : Cintya Tresna Walidin, 2022.

### 2.3.3 Flowchart

Flowchart adalah representasi grafis dari langkah-langkah dan urutan prosedur dalam suatu program. Flowchart membantu analis dan programmer untuk memecah masalah menjadi segmen-segmen yang lebih kecil serta membantu dalam mengevaluasi alternatif-alternatif operasional lainnya. Flowchart biasanya mempermudah penyelesaian masalah, terutama yang memerlukan kajian dan

evaluasi lebih lanjut. Diagram ini menggambarkan alur dengan satu atau dua arah secara berurutan. Flowchart digunakan untuk merepresentasikan atau merancang program. Simbol flowchar dapat dilihat pada Tabel 2.3.

**Tabel 2. 3 Simbol Flowchart**

Simbol	Nama	Fungsi
	Terminal ( <i>terminal</i> )	Menunjukkan awal dan akhir program
	Garis alir ( <i>flow line</i> )	Digunakan untuk menunjukkan arah aliran data
	Persiapan ( <i>preperation</i> )	Memberikan nilai awal pada suatu variable atau counter
	Pemroses ( <i>processing</i> )	Pengolahan aritmatika dan pemindahan data
	Keputusan ( <i>decision</i> )	Mewakili operasi perbandingan logika.
	Proses terdefenisi ( <i>predefined process</i> )	Simbol ini digunakan untuk proses yang ditailnya dijelaskan terpisah
	Penghubung terputus ( <i>offpage connector</i> )	Menunjukkan penghubung arus proses yang terputus dengan sambungannya ada di halaman lain.

Sumber : Bunaya Arthavia Sitorus, 2023.

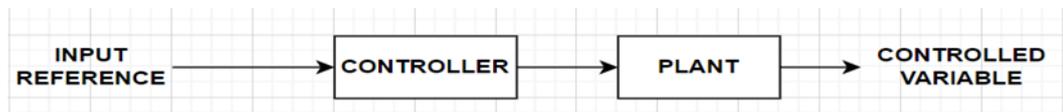
## 2.4 Sistem Kontrol

Sistem kontrol adalah sistem yang dikendalikan secara otomatis menggunakan komputer digital dan operasinya diawasi oleh manusia. Berdasarkan fungsi pengontrolannya secara menyeluruh, komponen sistem kontrol dapat dibagi menjadi empat bagian yaitu: sensor transduser, pemroses, aktuator, dan penguat.

Berdasarkan cara kerjanya, sistem kontrol dapat dibagi menjadi dua jenis, yaitu sistem kontrol loop terbuka dan sistem kontrol loop tertutup.

#### 2.4.1 Sistem Loop Terbuka

Dalam sistem kendali loop terbuka, keluaran tidak mempengaruhi aksi kontrol. Dengan kata lain, pada sistem kendali ini, keluaran tidak dapat digunakan sebagai perbandingan atau umpan balik untuk masukan sistem. Gangguan yang ada merupakan sinyal yang cenderung merugikan dan akan memberikan pengaruh buruk pada nilai keluaran sistem tersebut, baik gangguan yang berasal dari dalam sistem maupun dari luar sistem. Diagram blok sistem pengendalian loop terbuka dapat dilihat pada Gambar 2.3.

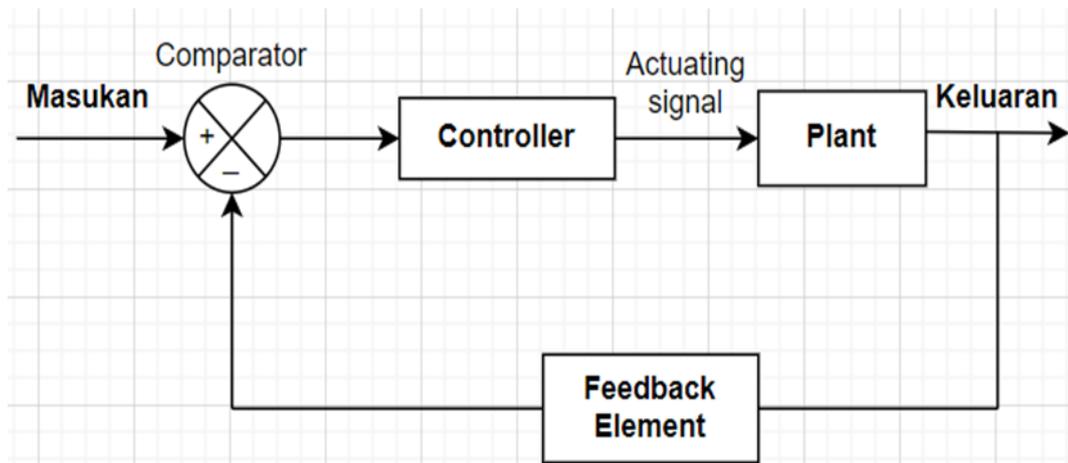


**Gambar 2. 3 Sistem Loop Terbuka**

*Sumber : totorialspoint.*

#### 2.4.2 Sistem Loop Tertutup

Sistem yang menggunakan keluaran sebagai referensi untuk masukan disebut sistem kendali loop tertutup, atau sering disebut sebagai sistem kontrol umpan balik. Secara sederhana, sistem kontrol umpan balik dan sistem kendali loop tertutup selalu melibatkan penggunaan aksi kontrol umpan balik untuk mengurangi kesalahan sistem. Diagram blok sistem pengendalian loop tertutup dapat dilihat pada Gambar 2.4.



**Gambar 2. 4 Sistem Loop Tertutup**

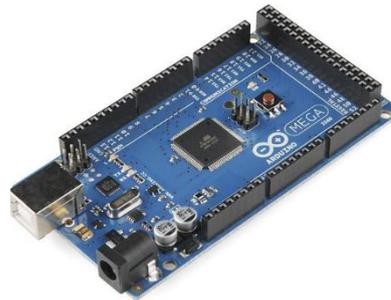
*Sumber : tutorialspoint.*

## 2.5 Komponen Utama

Pada bagian ini akan dijelaskan komponen utama yang akan digunakan pada sistem monitoring kapasitas saus ini adalah:

### 2.5.1 Arduino Mega 2560

Arduino adalah sebuah papan elektronik atau kit yang didukung perangkat lunak open source, menggunakan keluarga mikrokontroler ATmega dan berfungsi sebagai pengendali mikro single-board. Alat ini dirancang untuk memudahkan aplikasi elektronik di berbagai bidang dan diproduksi oleh Atmel. Perangkat kerasnya menggunakan prosesor Atmel AVR, sementara perangkat lunaknya memiliki bahasa pemrograman tersendiri. Bentuk fisik dari Arduino Mega 2560 dapat dilihat pada Gambar 2.5.



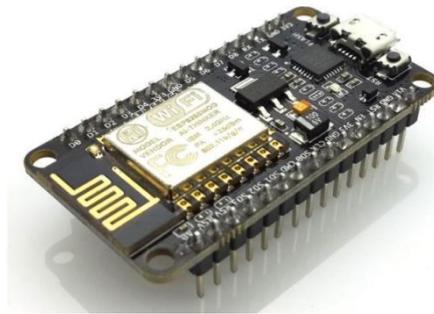
**Gambar 2. 5 Arduino Mega 2560**

*Sumber : Iskandar, A., Muhajirin, & Lisah, 2017.*

Arduino Mega memiliki input/output dengan 15 pin yang bisa digunakan sebagai output PWM, 16 pin untuk input analog, dan 14 pin sebagai UART (port serial perangkat keras). Selain itu, papan ini dilengkapi dengan osilator kristal 16 MHz, tombol reset, header ICSP, koneksi USB, dan jack daya. Semua fitur ini mendukung mikrokontroler untuk beragam tugas. Untuk mulai mengoperasikannya, cukup hubungkan perangkat ke komputer melalui kabel USB, sumber daya, atau baterai (Iskandar, A., Muhajirin, & Lisah, 2017).

### 2.5.2 NodeMCU ESP8266

NodeMCU adalah papan elektronik berbasis chip ESP8266 yang berfungsi sebagai mikrokontroler dan menyediakan koneksi WiFi. Dengan beberapa pin I/O, NodeMCU dapat digunakan untuk aplikasi pemantauan dan pengendalian dalam proyek IoT. Papan ini dapat diprogram menggunakan Arduino IDE dan dilengkapi dengan port mini USB untuk memudahkan pemrograman. NodeMCU ESP8266, yang dikembangkan dari modul platform IoT ESP-12, dirancang khusus untuk koneksi internet (Sulistiyorini, Sofi, & Sova, 2022). Bentuk fisik dari NodeMCU ESP8366 dapat dilihat pada Gambar 2.6.



**Gambar 2. 6 NodeMCU ESP8266**

*Sumber : Sulistyorini, Sofi, & Sova, 2022.*

### 2.5.3 Load Cell

Sensor loadcell adalah transduser yang mengubah berat suatu benda menjadi sinyal listrik, melalui perubahan resistansi pada strain gauge. Sebuah sensor loadcell biasanya memiliki empat strain gauge yang tersusun dalam sebuah konfigurasi. Nilai konduktansi sensor ini berbanding lurus dengan gaya atau beban yang diterima, bersifat resistif. Ketika tidak ada beban, resistansi di setiap sisi loadcell bernilai sama, namun saat beban diterapkan, resistansinya menjadi tidak seimbang. Perubahan inilah yang digunakan untuk mengukur berat suatu benda (Wibowo & Supriyono, 2019). Bentuk fisik dari sensor Load Cell dapat dilihat pada Gambar 2.7.



**Gambar 2. 7 Load Cell**

*Sumber : Wibowo & Supriyono, 2019.*

#### 2.5.4 Modul HX711

HX711 adalah komponen dari "AVIA SEMICONDUCTOR", berupa ADC 24-bit presisi yang dirancang untuk sensor timbangan digital dalam aplikasi kontrol industri. Modul ini mengubah perubahan resistansi menjadi tegangan dan berkomunikasi dengan komputer atau mikrokontroler melalui TTL232. Dengan struktur sederhana, mudah digunakan, stabil, andal, sangat sensitif, dan mampu mengukur perubahan dengan cepat (Mukhammad, Santika, & Haryuni, 2022). Bentuk fisik dari Modul HX711 dapat dilihat pada Gambar 2.8.



**Gambar 2. 8 Modul HX711**

*Sumber : Mukhammad, Santika, & Haryuni, 2022.*

#### 2.5.5 Keypad 4x4

Modul keypad 4x4 adalah sebuah modul keypad yang memiliki ukuran 4 kolom x 4 baris. Modul ini dapat digunakan sebagai perangkat input dalam berbagai aplikasi seperti sistem keamanan digital, pencatatan data, absensi, pengendalian kecepatan motor, robotika, dan lain-lain. Keypad matriks 4x4 hanya memerlukan 8 pin untuk mengoperasikan 16 tombol yang ada (Hendra, Ngemba, & Mulyono, 2017). Bentuk fisik dari Keypad 4x4 dapat dilihat pada Gambar 2.9.



**Gambar 2. 9 KeyPad 4x4**

*Sumber : Kamolan, A., & Sampebatu, L., 2021.*

### 2.5.6 Water Pump

Pompa air merupakan alat yang secara cepat dan efisien mengeluarkan air dalam jumlah lebih besar dibandingkan jika dilakukan secara manual dengan ember. Alat ini menggunakan tenaga untuk memompa air dari sumbernya, berbeda dengan metode mengangkat ember dengan tangan. Tingkat kandungan air sangat mempengaruhi jumlah energi yang diperlukan untuk menjalankan pompa air, tanpa mempengaruhi hasil yang diperoleh (Shofi, A., A., dkk, 2023). Bentuk fisik dari Water Pump dapat dilihat pada Gambar 2.10.



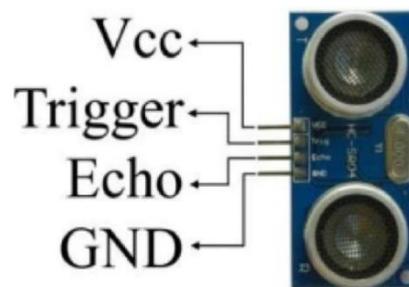
**Gambar 2. 10 Water Pump**

*Sumber : Shofi, A., A., dkk, 2023.*

### 2.5.7 Ultrasonic Sensor

Sensor ultrasonic HC-SR04 adalah sensor yang digunakan untuk mengukur jarak antara objek dan sensor tersebut. HC-SR04 memiliki empat pin:

Vcc, Trigger, Echo, dan Ground. Spesifikasi sensor ultrasonic adalah sebagai berikut: sensor ini beroperasi pada tegangan DC 5V dengan arus kerja sebesar 15mA, frekuensi kerja 40Hz, jarak pengukuran maksimal 4 meter, dan jarak pengukuran minimal 2cm (Arifin, Pratiwi, & Janrafsasih, 2022). Bentuk fisik dari sensor Ultrasonic dapat dilihat pada Gambar 2.11.



**Gambar 2. 11 Sensor Ultrasonic**

*Sumber : Arifin, Pratiwi, & Janrafsasih, 2022.*

### 2.5.8 LCD 16x2

Liquid Crystal Display merupakan komponen elektronik yang digunakan untuk menampilkan data, baik berupa karakter, huruf, maupun grafik. LCD (Liquid Crystal Display) adalah salah satu jenis display elektronik yang memanfaatkan teknologi CMOS logic. LCD ini tidak memproduksi cahaya sendiri, melainkan memantulkan cahaya dari sekitar (front-lit) atau mentransmisikan cahaya dari belakang (back-lit). Fungsi LCD adalah untuk menampilkan data dalam bentuk karakter, huruf, angka, atau grafik (Ardi Prianto, 2021). Bentuk fisik dari LCD 16x2 dapat dilihat pada Gambar 2.12.



**Gambar 2. 12 LCD 16x2**

*Sumber : Mindasari, As'ad, & Meilantika, 2022.*

### 2.5.9 Modul I2C

I2C LCD adalah modul LCD yang dikontrol secara serial sinkron menggunakan protokol I2C/IIC (Inter Integrated Circuit) atau TWI (Two Wire Interface). Biasanya, modul LCD dikendalikan secara paralel, baik untuk jalur data maupun kontrol. Namun, pengaturan jalur paralel membutuhkan banyak pin pada pengendali (seperti Arduino, komputer, dll). Setidaknya, diperlukan 6 atau 7 pin untuk mengoperasikan sebuah modul LCD. Oleh karena itu, bagi pengendali yang harus mengelola banyak I/O, menggunakan jalur paralel bukanlah solusi yang efektif (Suryantoro, & Budiyanto, 2019). Bentuk fisik dari Modul I2C dapat dilihat pada Gambar 2.13.



**Gambar 2. 13 Modul I2C**

*Sumber : Suryantoro, & Budiyanto, 2019.*





**Gambar 2. 15 Modul Step Down XL4005E1**

*Sumber : Arif, M., 2024.*

#### 2.5.12 LED

Led adalah komponen elektronika yang dapat memancarkan cahaya monokromatik ketika diberikan tegangan maju. LED merupakan keluarga Dioda yang terbuat dari bahan semikonduktor (Roghib muh, 2018). Bentuk fisik dari LED dapat dilihat pada Gambar 2.16.



**Gambar 2. 16 LED**

*Sumber : Roghib Muh, 2017.*

#### 2.5.13 Bot Telegram

Bot adalah layanan di aplikasi Telegram yang menjalankan tugas otomatis, sering digunakan untuk menggantikan manusia. Bentuk paling umum adalah chatbot yang mengenali teks atau ucapan dan merespons. Setelah Telegram

meluncurkan API bot pada 2015, pihak ketiga dapat membuat bot sebagai antarmuka utama. Bot dapat dibuat melalui channel Telegram atau BotFather.



**Gambar 2. 17 Bot Telegram**

*Sumber : Marshal, G., W., & Soelistijadi, R., 2023.*

Pada penelitian ini, pembuatan Telegram bot dilakukan menggunakan BotFather. BotFather merupakan satu-satunya metode untuk membuat bot Telegram tanpa perlu coding untuk mengelola bot yang sudah ada (Pardede & dkk, 2022). Ikon bot telegram dapat dilihat pada Gambar 2.17.

## **2.6 Komponen Pendukung**

Komponen pendukung atau komponen pendukung sistem merujuk pada berbagai bagian, perangkat, atau elemen yang berperan dalam mendukung operasi dan fungsi suatu sistem.

### **2.6.1 Kabel Jumper**

Kabel jumper adalah kabel listrik dengan pin konektor di setiap ujungnya, yang memungkinkan kamu untuk menghubungkan dua komponen yang melibatkan Arduino tanpa perlu solder. Pada dasarnya, fungsi kabel jumper adalah sebagai konduktor listrik untuk menyambungkan rangkaian listrik. Biasanya, kabel jumper digunakan pada breadboard atau alat prototyping lainnya untuk memudahkan dalam mengatur rangkaian. Konektor di ujung kabel terdiri dari dua jenis, yaitu konektor

jantan (*male connector*) dan konektor betina (*female connector*). Bentuk fisik dari kabel jumper dapat dilihat pada Gambar 2.18.



**Gambar 2. 18 Kabel Jumper**

*Sumber : Alfan & Ramadhan, 2022.*

#### 2.6.2 Kabel Pelangi

Kabel Pelangi adalah ini biasa digunakan pada rangkaian elektronik yang membutuhkan banyak kawat konduktor sebagai alat untuk penghubung (Elga Aris Prasetyo, 2017). Bentuk fisik dari kabel pelangi dapat dilihat pada Gambar 2.19.



**Gambar 2. 19 Kabel Pelangi**

*Sumber : Elga Aris Prasetyo, 2017.*

### 2.6.3 Resistor

Resistor merupakan komponen elektronika yang mempunyai sifat dapat menghambat arus listrik. Termasuk komponen pasif karena tidak memerlukan arus listrik agar dapat berfungsi. Komponen ini dibuat dari bahan karbon dan keramik serta memiliki bentuk tabung (Ashar Arifin, 2020). Bentuk fisik dari resistor dapat dilihat pada Gambar 2.20.



**Gambar 2. 20 Resistor**

*Sumber : Ashar Arifin, 2020*

### 2.6.4 Kapasitor

Kapasitor merupakan salah satu komponen elektronika yang sangat penting fungsinya. Pengertian kapasitor adalah suatu komponen elektronika yang dapat menyimpan dan melepaskan muatan listrik atau energi listrik. Selain itu, kapasitor juga dapat berfungsi sebagai penyaring frekuensi. Kapasitor memiliki berbagai macam ukuran dan bentuk tergantung dari kapasitas, tegangan kerja dan faktor lainnya yang berpengaruh. Kapasitor sering disebut juga dengan kondensator. Fungsi kapasitor untuk menyimpan muatan listrik disebut dengan kapasitansi atau kapasitas. Kapasitor memiliki simbol  $C$  (*Capasitor*) sedangkan fungsi kapasitor dalam menyimpanmuatan listrik disimbolkan oleh  $F$  (*Farad*) (Sitohang, E., dkk, 2018). Bentuk fisik dari kapasitor dapat dilihat pada Gambar 2.21.



**Gambar 2. 21 Kapasitor**

*Sitohang, E., dkk, 2018*

## **2.7 Bahasa Pemrograman Arduino**

Bahasa pemrograman Arduino yang dipakai adalah Arduino. Bahasa C adalah Bahasa yang lazim dipakai sejak awal komputer diciptakan dan sangat berperan dalam perkembangan software. Karena Arduino menggunakan Bahasa C yang multi- platform, software Arduino juga bisa dijalankan pada sistem operasi yang umum, misalnya Windows, Linux, dan MacOS.

### **2.7.1 Struktur Arduino**

Dasar struktur dari bahasa pemrograman Arduino sangatlah mudah dan sederhana. Agar program dapat berjalan dengan baik, maka perlu setidaknya dua bagian atau fungsi, yaitu `setup ( )` yang di panggil hanya satu kali, biasanya untuk inialisasi program (seting input atau seting serial, dan lain-lain). Dan `loop ( )` tempat untuk mengeksekusi program secara berulang, biasanya untuk membaca input, atau men-tingger ouput.

a. Setup ( )

Setup ( ) function hanya dipanggil satu kali saja saat program mulai berjalan. Fungsi setup ( ) berguna untuk melakukan inisialisasi mode pin atau untuk memulai komunikasi serial. Setup ( ) ini harus ada meskipun tidak ada program yang akan dieksekusi.

b. Loop ( )

Fungsi loop ( ), sesuai dengan namanya fungsi ini akan mengulang program yang akan secara terus-menerus, sehingga program akan berubah dan merespon sesuai inputan. Fungsi loop ( ) ini akan secara aktif mengontrol board Arduino.

c. Functions ( )

Sebuah fungsi berupa sekumpulan program yang diberi nama khusus, yang dapat dieksekusi dengan cara memanggil fungsi tersebut. Sebuah fungsi bisa dibuat sesuai dengan kebutuhannya untuk melakukan suatu pekerjaan yang berulang ataupun bisa membuat program jadi lebih sederhana.

d. Curly Braces { }

Tanda kurung kurawal { } selalu digunakan secara seimbang pada pemrograman di Arduino. Apabila tanda kurung kurawal { } tidak seimbang maka saat program di compile akan eror. Tanda tersebut selalu digunakan pada awal dan akhir suatu blok program. Kurung kurawal terdapat pada fungsi, perulangan, dan kondisi statement.

e. Semicolon ;

Titik koma digunakan pada akhir statement ataupun memisahkan element didalam looping atau perulangan. Titik koma juga digunakan untuk memberi tahu komputer saat akan mengeksekusi instruksi.

f. Block Comment `/*...*/`

Blok comment merupakan area text yang tidak akan berpengaruh oleh program. blok comment digunakan untuk memberikan komentar atau catatan tentang program yang dibuat, sehingga pembaca lain bias memahami maksud program tersebut.

### 2.7.2 Syntax Pemrograman

Berikut ini adalah elemen Bahasa C yang dibutuhkan untuk format penulisan:

a. (Komentar satu baris) `//`

Pemberian komentar setelah penulisan kode. Apapun yang ditulis dibelakang symbol ini akan diabaikan oleh program.

b. (Komentar banyak baris) `/*...*/`

Pemberian komentar pada kode yang ditulis jika komentar lebih dari satu baris program.

c. (Kurung kurawal) `{ }`

Digunakan untuk mendefinisikan kapan blok program mulai dan berakhir.

d. (Titik koma) `;`

Setiap baris kode harus diakhiri dengan titik koma, jika tidak maka program tidak bisa dijalankan.

### 2.7.3 Variabel

Sebuah program secara garis besar dapat didefinisikan sebagai instruksi untuk memindahkan angka dengan cerdas. Variabel inilah yang digunakan untuk memindahkannya.

1. Int (Integer)

Digunakan untuk menyimpan angka dalam 2 byte (16 bit). tidak mempunyai angka decimal dan menyimpan nilai dari -32.76 sampai 32.767.

2. Long (long)

Digunakan Ketika integer tidak mencukupi lagi. Memakai 4 byte (32 bit) Dari memori RAM dan mempunyai rentang dari -2.147.483.648 sampai 2.147.483.648.

3. Boolean (Boolean)

Variabel sederhana yang digunakan untuk menyimpan nilai benar atau salah. Sangat berguna karena hanya menggunakan 1 bit dari RAM.

4. Float (float)

Digunakan untuk angka decimal. Memakai 4 bit (32 bit) dari RAM dan mempunyai rentang dari -3,4028235E+38 sampai 3,4028235E+38.

5. Char (character)

Menyimpan 1 karakter menggunakan kode ASCII. Hanya memakai 1 byte (8 bit) dari RAM.

6. Byte

Angka antara 0 sampai 255. Sama halnya dengan char, namun byte hanya menggunakan 1 byte memori.

7. Unsigned int

Sama dengan integer, menggunakan 2 byte tetapi tidak dapat menyimpan nilai negatif nilai 0 sampai 65,35.

## 8. Unsigned long

Sama dengan long, namun tidak dapat menyimpan angka negative, batasannya dari 0 sampai 4.294.967.295.30.

## 9. Double

Angka ganda dengan presisi maksimum  $1.7976931348623157 \times 10^{308}$

## 10. String

Digunakan untuk informasi teks, dengan karakter ASCII.

## 11. Array

Kumpulan variable bertipe sama. Setiap variable dalam kumpulan variable tersebut terdapat elemen, dapat diakses melalui indeks.

## 2.7.4 Struktur Pengaturan

Program sangat tergantung pada pengaturan apa yang akan dijalankan berikutnya, berikut ini adalah elemen dasar pengaturan.

## 1. If...else, dengan format seperti berikut ini:

If (kondisi) { } Else { } Dengan struktur seperti diatas program akan menjalankan kode yang ada didalam kurung kurawal jika kondisinya TRUE, dan jika tidak (FALSE) maka akan diperiksa apakah kondisi pada Else if dan jika kondisinya FALSE maka kode pada Else yang akan dijalankan.

## 2. For, dengan format seperti berikut ini:

For (int=0;I <#pengulangan; ++ ) { } Digunakan bila anda ingin melakukan perulangan didalam kurung kurawal. Beberapa kali, ganti #pengulangan dengan jumlah perulangan yang diinginkan. Malekukan perhitungan keatas dengan i++ atau kebawah i--.

### 2.7.5 Operator

Operator adalah simbol atau kata kunci yang digunakan untuk melakukan operasi pada satu atau lebih operand (nilai, variabel, atau ekspresi). Operator memungkinkan anda untuk melakukan tugas-tugas seperti perhitungan matematika, manipulasi data, dan logika dalam program. Berikut jenis jenis operator dalam pemrograman :

#### a. Operator Aritmatika

Operator aritmatika merupakan operator yang digunakan untuk operasi aritmatika seperti penjumlahan, pengurangan, perkalian, atau pembagian. Tabel 2.4 akan menunjukkan operator aritmatika.

**Tabel 2. 4 Operator Aritmatika**

<b>Operator</b>	<b>Keterangan</b>
+	Penjumlahan
-	Pengurangan
*	Perkalian
/	Pembagian
%	Sisa Bagi
**	Pemangkatan

#### b. Operator Pembandingan

Operator pembandingan biasanya digunakan dalam pemilihan kondisi dan perulangan. Operator ini akan membandingkan dua angka atau variabel. Tabel 2.5 akan menunjukkan operator pembandingan.

**Tabel 2. 5 Operator Pembandingan**

<b>Operator</b>	<b>Keterangan</b>
>	Lebih Besar
<	Lebih Kecil
==	Sama Dengan
!=	Tidak Sama Dengan
>=	Lebih Besar Sama Dengan
<=	Lebih Kecil Sama dengan

c. Operator Logika

Operator logika biasanya digunakan untuk membuat keputusan atau pernyataan logika dalam program, terutama dalam kondisional (seperti if, while) dan struktur kontrol lainnya. Operator ini memeriksa kondisi tertentu dan mengembalikan nilai benar (True) atau salah (False). Tabel 2.6 akan menunjukkan operator logika.

**Tabel 2. 6 Operator Logika**

<b>Operator</b>	<b>Keterangan</b>
AND	menghasilkan nilai logika "True" jika kedua operand bernilai benar.
OR	menghasilkan nilai "True" jika salah satu operand bernilai benar.
XOR	menghasilkan nilai "TRUE" jika salah satu operand bernilai benar, namun tidak keduanya.

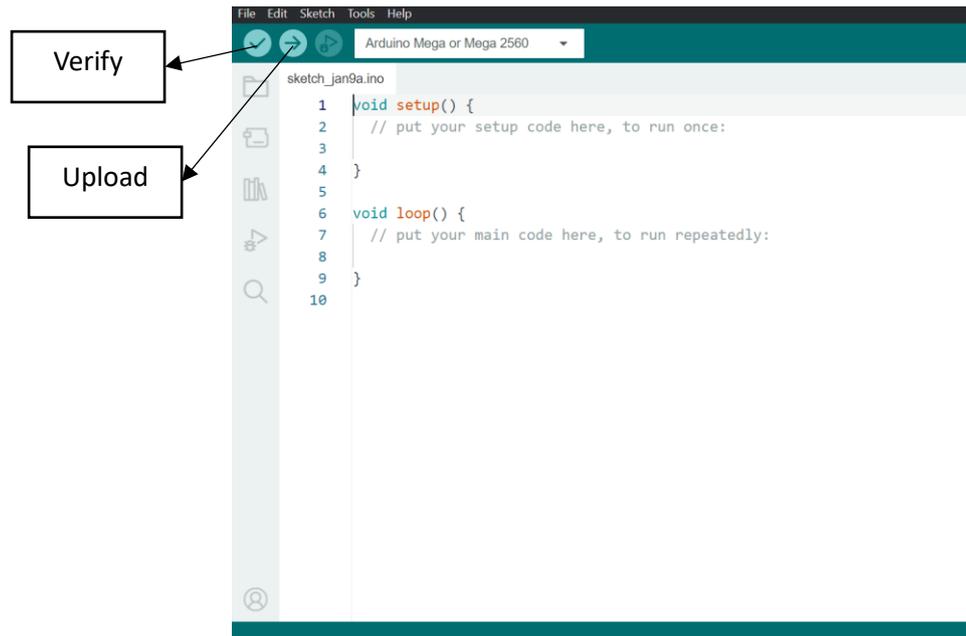
Operator	Keterangan
NOT	menghasilkan nilai "True" jika salah satu operand bernilai "False".

## 2.8 Software Arduino IDE

IDE merupakan singkatan dari *Integrated Development Environment*. IDE merupakan program yang digunakan untuk membuat program pada Arduino Mega dan ESP8266 *NodeMcu*. Program yang ditulis dengan 'menggunakan Software Arduino IDE disebut sebagai sketch. Sketch ditulis dalam suatu editor teks dan disimpan dalam file dengan ekstensi .ino. Pada Software Arduino IDE, terdapat semacam message box berwarna hitam yang berfungsi menampilkan status, seperti pesan error, compile, dan upload program. Di bagian bawah paling kanan Software Arduino IDE, menunjukkan board yang terkonfigurasi beserta COM Ports yang digunakan.

1. Verify/Compile, berfungsi untuk mengecek apakah sketch yang dibuat ada kekeliruan dari segi sintaks atau tidak. Jika tidak ada kesalahan, maka sintaks yang dibuat akan dicompile kedalam bahasa mesin.
2. Upload, berfungsi mengirimkan program yang sudah dikompilasi ke Arduino Board.

Tampilan Software IDE Arduino dapat dilihat pada Gambar 2.22.



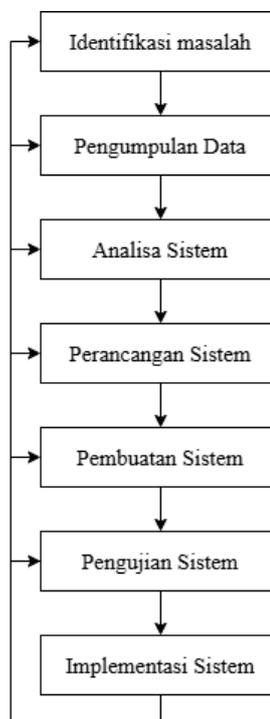
**Gambar 2. 22 Arduino IDE**

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Kerangka Kerja Penelitian

Kerangka kerja penelitian merupakan tahap awal dari pembuatan sebuah produk. Tahap ini yang akan menentukan hasil akhir dari sebuah produk yang akan dibuat hampir secara keseluruhan. Didalam kerangka kerja penelitian ini akan digambarkan tahap-tahap dari sebuah penelitian yang dilakukan untuk mempermudah langkah- langkah dari pembuatan alat yang akan dibuat, sehingga dapat dijadikan pedoman dalam menyelesaikan permasalahan yang ada. Maka diuraikan pada Gambar 3.1.



**Gambar 3. 1 Kerangka Kerja Penelitian**

## **3.2 Uraian Kerangka Kerja Penelitian**

Berdasarkan kerangka kerja penulisan yang telah digambarkan, maka dapat diuraikan pembahasan masing-masing tahap dalam penulisan adalah sebagai berikut :

### **3.2.1 Identifikasi masalah**

Identifikasi masalah dilakukan menggunakan pendekatan terhadap objek penulisan. Tujuan dari tahap ini yaitu untuk mengetahui permasalahan yang terjadi secara tepat, sehingga diharapkan penulisan dapat memberikan solusi yang paling optimal terhadap pemecahan permasalahan tersebut.

Pada tahapan ini, penulis menentukan objek penulisan yang akan diteliti lalu menetapkan masalah yang ingin dianalisis. Pada restoran cepat saji, ketersediaan saus tomat sering kali sulit dipantau secara efisien karena masih menggunakan metode manual. Hal ini menyebabkan waktu respons untuk pengisian ulang menjadi lambat, terutama pada jam-jam sibuk.

### **3.2.2 Pengumpulan Data**

Berdasarkan metode yang dilakukan pada saat melakukan proses penulisan untuk memperoleh data yang akurat dan juga dalam pembuatan alat dan pengerjaan laporan ini. metode yang digunakan untuk mengumpulkan data ada beberapa metode, diantaranya :

#### **3.2.2.1 Metode Penelitian**

Dalam melakukan penelitian agar mendapatkan hasil seperti yang diinginkan, maka sekiranya diperlukan suatu metodologi penelitian yang umum

dilakukan. Metode yang digunakan untuk mengumpulkan data ada beberapa metode, diantaranya :

a. Observasi

Observasi dilakukan dengan mengumpulkan data melalui pengamatan terhadap minat masyarakat terhadap kemajuan teknologi yang pesat saat ini. Data tersebut kemudian dapat diterapkan pada prototype alat monitoring kapasitas saus tomat pada restoran cepat saji menggunakan mikrokontroler ESP8266.

b. Studi pustaka

Data dikumpulkan dengan mempelajari buku dan literatur lainnya yang bisa dijadikan referensi untuk mengembangkan rototype alat monitoring kapasitas saus tomat pada restoran cepat saji menggunakan mikrokontroler ESP8266.

c. Studi Laboratorium

Metode ini dilakukan untuk menguji konsep-konsep yang ada dengan menggunakan peralatan yang dipakai dan sesuai. Adapun objek yang diuji spesifikasi hardware dan software yang digunakan dalam penulisan ini dapat dilihat pada Table 3.1.

**Tabel 3. 1 Spesifikasi Hardware dan Software**

<b>Hardware</b>	<b>Softwaree</b>
Satu Unit laptop Asus, Intel (R)	Sistem Operasi Windows 11
Core(TM) i7-10510U CPU @	Home Single Language
1.80GHz 2.30 GHz, RAM 8,00 GB, SSD	Microsoft Office 2016
Tera, Modul Mikrocontroller	Bahasa Pemrograman Arduino
Arduino Mega 2560	Fritzing2

### 3.2.3 Analisis Sistem

Berdasarkan identifikasi masalah diatas, penulis melakukan analisa sistem terlebih dahulu. Hal ini bertujuan agar pemecahan masalah dapat menghasilkan sebuah solusi baru.

Tahap analisis merupakan langkah krusial dalam pengembangan sebuah sistem, karena pada tahap ini dilakukan evaluasi sistem, perancangan sistem, dan penentuan langkah-langkah yang dibutuhkan untuk merancang alat sampai mendapatkan analisis yang diinginkan. Peneliti melakukan analisis data untuk memecahkan masalah guna menghasilkan solusi baru sebelum merancang alat. Dalam penelitian ini, peneliti bertujuan menjelaskan input, output, dan proses dari sistem yang akan dibuat. Input adalah sebuah masukan yang bisa diproses menjadi output. salah satunya pada alat ini adalah Sensor load cell berfungsi untuk pembacaan berat pada tempat penampung saus dan modul HX711 sebagai modul timbangannya. LCD akan menampilkan berapa banyak saus yang ingin di ambil. Led berfungsi untuk indikator banyak nya saus. Telegram berfungsi untuk memonitoring dari jarak jauh. Serta akan memberikan notifikasi ketika saus pada tempat tempat penampung saus akan habis. Dan konsumen bisa mengetahui berapa saus yang tersedia pada tempat penampung saus dengan memberikan pesan ke telegram. Keypad berfungsi untuk menentukan banyak saus yang akan di ambil. Sensor ultrasonic sebagai pendeteksi banyak nya saus pada tempat penampung saus. Water pump berfungsi untuk memompa saus.

### 3.2.4 Perancangan Sistem

Perancangan dalam Penelitian berupa gambaran dan bentuk awal dari alat secara keseluruhan, dengan adanya desain ini maka prinsip kerja dari alat serta

komponen-komponen dari sistem yang digunakan akan dapat dilihat dengan jelas sebagai berikut :

1. *Context Diagram*

*Context diagram* menggambarkan rancangan keseluruhan, *entity external* harus digambarkan sedemikian rupa, sehingga terlihat data yang mengalir pada *input-proses-output*.

2. *Data Flow Diagram (DFD)*

*Data Flow Diagram (DFD)* merupakan suatu diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sistem ke entitas. DFD juga dapat diartikan sebagai teknik grafis yang menggambarkan alir data dan transformasi yang digunakan sebagai perjalanan data dari input atau masukan menuju keluaran output.

3. *Block Diagram*

*Block Diagram* yang dibuat untuk mempetakan proses kerja pada suatu alat, hal ini bertujuan untuk memudahkan seseorang dalam mengenal komponen-komponen elektronika dari alat dan memahami alur kerja didalamnya.

4. *Flowchart*

*Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program. *Flowchart* menolong analis dan programmer untuk memecahkan masalah dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif lain dalam pengoperasian, khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.

## 5. *Software Design*

*Software Design* dari alat yang dibuat berupa gambaran dan bentuk awal dari alat secara keseluruhan. Dengan adanya desain ini maka prinsip kerja dari alat serta komponen-komponen dari sistem yang digunakan akan dapat dilihat dengan jelas. Software yang digunakan untuk perancangan alat adalah *google sketchup*.

### 3.2.5 Pembuatan alat

Pada proses pembuatan alat, semua data yang telah dikumpulkan dan riset yang telah dilakukan akan di olah dan semua komponen utama dan pendukung akan disatukan, dirangkai berdasarkan konsep-konsep yang dibuat dan alat bantu (*context diagram, flowchart, dan data flow diagram*) pada proses kegiatan perancangan sistem, setelah semua di rangkai maka lanjut proses pengimputan program keseluruhan menggunakan aplikasi Arduino IDE agar kondisi alat dapat berjalan sesuai dengan yang diharapkan.

### 3.2.6 Pengujian sistem

Pengujian alat dibutuhkan agar alat yang dibuat sesuai dengan kebutuhan sehingga dapat digunakan oleh restoran cepat saji. Tujuan dari pengujian alat yaitu untuk melihat seberapa tepat alat yang dibuat sehingga sesuai dengan kebutuhan dan mengidentifikasi masalah yang terjadi pada sistem monitoring kapasitas saus pada restoran cepat saji. Tahapan pengujian :

1. Sambungkan alat monitoring kapasitas saus tomat pada restoran cepat saji ke sumber listrik.
2. LCD akan menampilkan jumlah persesntase saus tomat.

3. Hidupkan hotspot yang terdapat di handphone untuk mengkoneksikannya dengan NodeMCu ESP8266.
4. Letakan tempat saus di atas sensor loadcell untuk di timbang.
5. Tekan tombol C pada keypad untuk tare berat nol.
6. Tentukan jumlah saus yang akan di ambil dengan cara menekan angka pada keypad.
7. Apabila sudah menginputkan jumlah saus yang akan di ambil, tekan tombol # pada keypad untuk mengeluarkan saus tomat.
8. Jika kapasitas saus tomat di tempat penampung saus tomat kurang dari 20% maka telegram akan mengirimkan notifikasi “stok saus habis”.
9. Jika mengirimkan pesan “CEK” pada telegram maka telegram akan mengirimkan jumlah saus yang tersisa pada tempat penampung saus.

### 3.2.7 Implementasi sistem

Implementasi sistem merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan, sehingga pengguna dapat memberi masukan kepada gerai-gerai restoran cepat saji.

## BAB IV

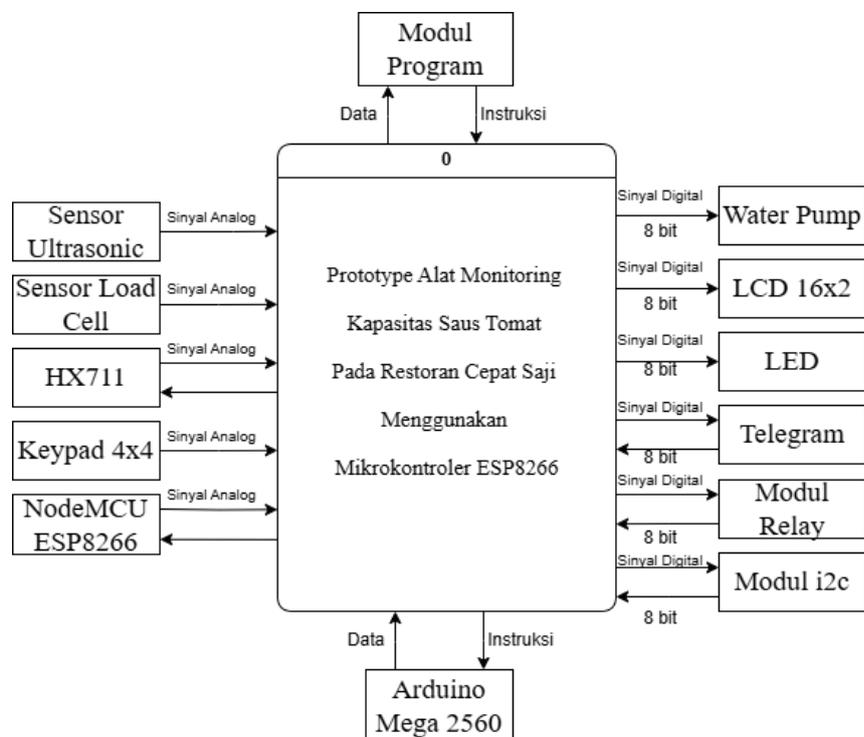
### ANALISA DAN HASIL

#### 4.1 Desain Sistem Secara Umum

Rancangan dari sistem yang dibuat merupakan representasi keseluruhan sistem. Dengan adanya rancangan ini, prinsip kerja sistem serta komponen-komponen yang digunakan dapat terlihat secara jelas.

##### 4.1.1 Context Diagram

*Context diagram* adalah gambaran umum tentang sistem yang akan dirancang. Diagram ini berfungsi untuk memudahkan dalam proses analisis sistem secara menyeluruh. *Context diagram* yang dimaksud dapat dilihat pada Gambar 4.1.



**Gambar 4. 1 Context Diagram**

Berdasarkan Gambar 4.1 yang menggambarkan sistem secara umum entiti yang digunakan saling berhubungan. Sesuai dengan penamaannya maka proses ini akan mengolah data *input* menjadi *output*. Proses ini akan berinteraksi dengan beberapa entiti yaitu :

1. Arduino Mega 2560

Arduino Mega 2560 berfungsi sebagai pusat sistem pengolahan seluruh data dan instruksi.

2. Modul Program

Modul program berfungsi untuk melakukan pembacaan terhadap pin-pin Arduino, baik pembacaan terhadap sinyal-sinyal input, memberikan instruksi-instruksi untuk mengaktifkan pin-pin output. Modul program mengontrol semua proses yang akan terjadi pada sistem dan program yang digunakan adalah bahasa pemrograman Arduino.

3. NodeMCU ESP826

NodeMCU ESP8266 berfungsi sebagai penghubung antara sistem dan telegram dengan cara menghidupkan hotspot pada sistem dan android/laptop.

4. Load Cell

Load Cell berfungsi sebagai penimbang berat saus yang akan di ambil, setelah mengetikan pada keypad 4x4 jumlah saus tomat yang akan diambil.

5. Modul HX711

Modul HX711 berfungsi untuk memperkuat sinyal dari load cell dan mengubahnya menjadi data digital untuk dibaca mikrokontroler.

#### 6. Keypad 4x4

Keypad 4x4 berfungsi untuk mengatur jumlah saus tomat yang akan diambil dengan menyetikikan pada papan keypad 4x4.

#### 7. Water Pump

Water pump berfungsi mengalirkan saus tomat dari tempat penampung saus ke wadah yang telah disiapkan diatas loadcell.

#### 8. Ultrasonic Sensor

Ultrasonic Sensor berfungsi sebagai pendeteksi jarak pada saus tomat untuk menentukan ketersediaan saus tomat.

#### 9. LCD 16x2

LCD 16x2 berfungsi untuk menampilkan berapa ketersediaan saus tomat, dan juga menampilkan berapa banyak saus yang ingin di ambil.

#### 10. Modul I2C

Modul I2C berfungsi sebagai komunikasi antar perangkat dengan dua kabel (SDA,SCL), menghemat pin dan menyederhanakan koneksi.

#### 11. Modul Relay

Modul Relat berfungsi untuk mengontrol *on/off* water pump.

#### 12. LED

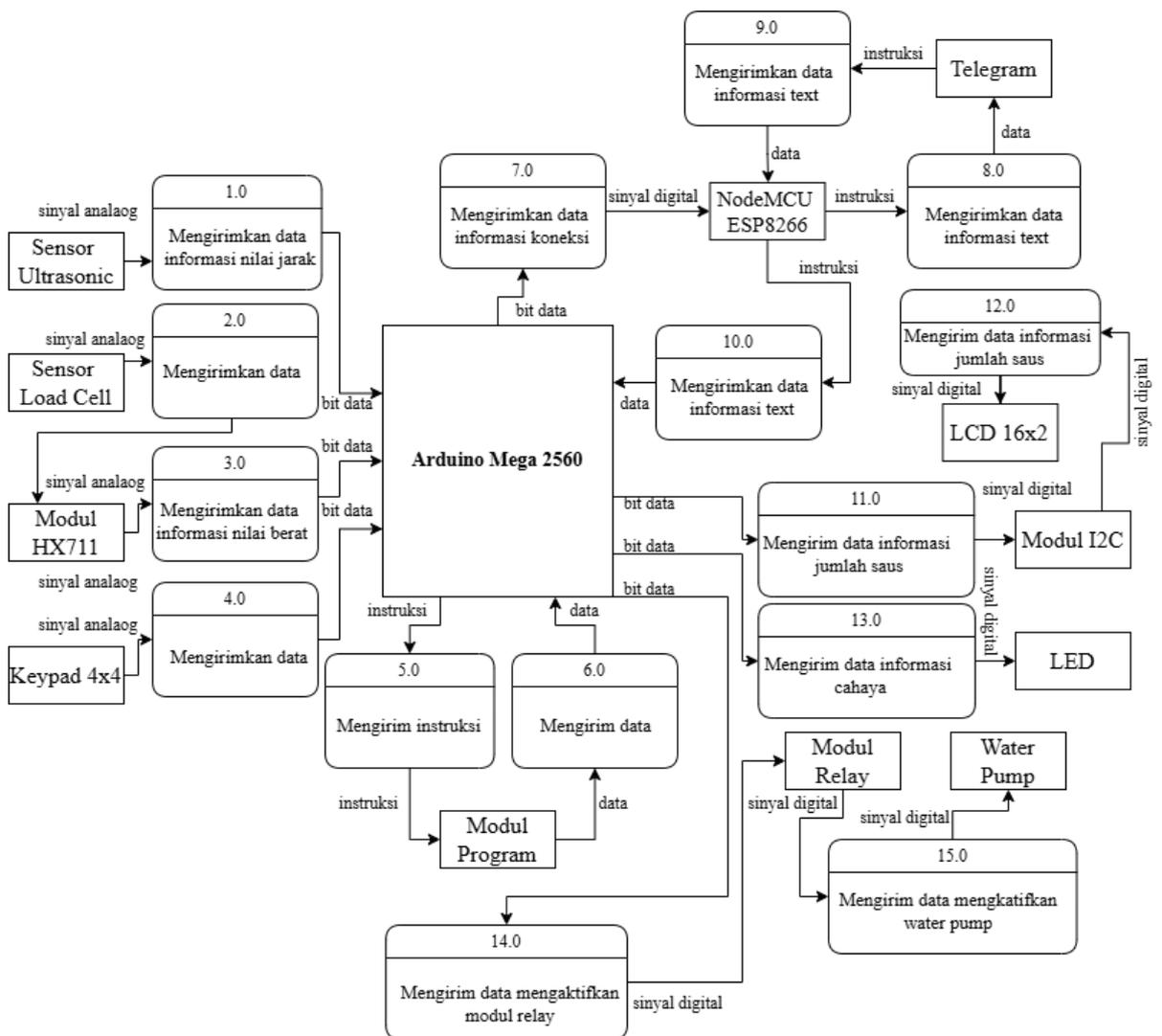
Led berfungsi seabagai indikator banyak saus tomat yang tersedia.

#### 13. Telegram

Telegram berfungsi untuk menampilkan notifikasi pesan berapa jumlah saus tomat yang tersisa.

#### 4.1.2 Data Flow Diagram

Data Flow Diagram (DFD) adalah gambaran yang lebih rinci dari alat yang dirancang untuk merepresentasikan alur data dalam sistem. Diagram ini diuraikan berdasarkan Context Diagram yang telah dijabarkan sebelumnya. Data Flow Diagram menunjukkan alur data dan proses secara terstruktur dalam suatu sistem. Gambar data flow diagram dapat dilihat pada Gambar 4.2.



**Gambar 4. 2 Data Flow Diagram**

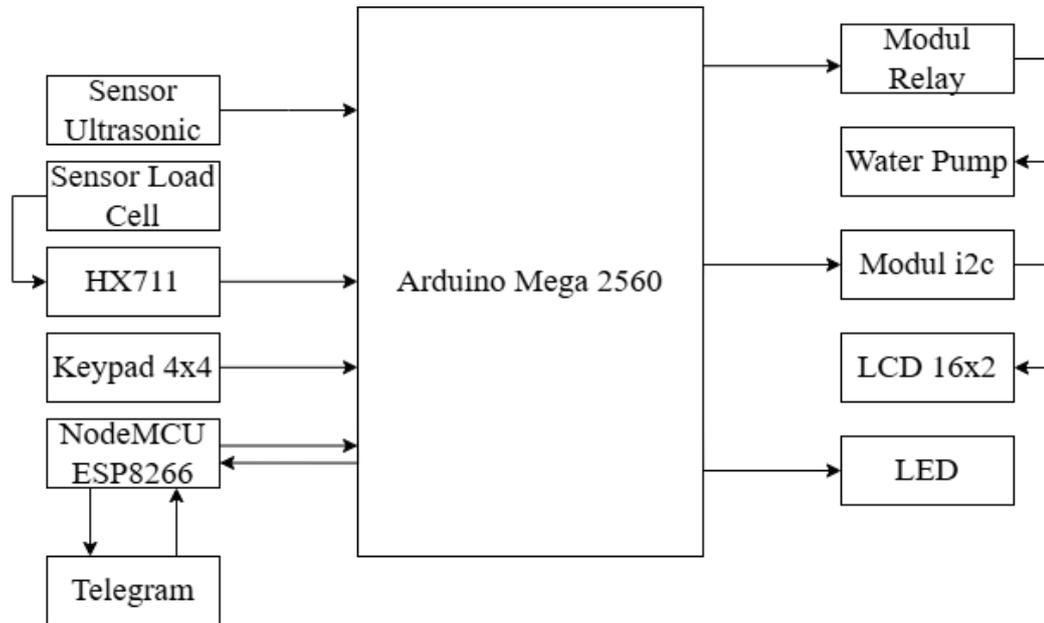
Berdasarkan Gambar 4.2 dapat dijelaskan bahwa sistem kendali dari alat memiliki beberapa urutan proses instruksi yaitu :

1. Sensor Ultrasonic mengirim data berupa sinyal analog yang kemudian akan diproses menjadi sinyal digital lalu dikirim ke Arduino Mega 2560 (1.0).
2. Sensor Load Cell mengirim data berupa sinyal analog yang kemudian akan diproses menjadi sinyal digital lalu dikirim ke modul HX711 (2.0).
3. Modul HX711 mengirim data informasi nilai berat berupa sinyal analog yang kemudian akan diproses menjadi sinyal digital lalu dikirim ke Arduino Mega 2560 (3.0).
4. Keypad 4x4 mengirim data berupa sinyal analog yang kemudian akan diproses menjadi sinyal digital lalu dikirim ke Arduino Mega 2560 (4.0).
5. Arduino Mega 2560 akan mengirimkan data ke modul program untuk diproses (5.0).
6. Modul program akan melakukan pembacaan terhadap pin-pin Arduino, baik pembacaan terhadap sinyal-sinyal input, memberikan instruksi-instruksi untuk mengaktifkan pin-pin output. Modul program mengontrol semua proses yang akan terjadi pada sistem dan program yang digunakan adalah bahasa pemrograman Arduino IDE, kemudian hasilnya dikirim ke Arduino Mega 2560 (6.0).
7. Arduino Mega 2560 akan mengirim sinyal digital ke NodeMCU ESP8266, yang berfungsi sebagai penghubung antara sistem dan telegram dengan cara menghidupkan *hotspot* pada sistem android. (7.0).
8. NodeMCU ESP8266 akan mengirimkan pembacaan sensor ultrasonic yang akan di teruskan ke telegram (8.0).

9. Ketika data pesan pembacaan sensor ultrasonic sudah tersampaikan ke telegram lalu instruksi akan diteruskan dengan mengirim data informasi ke NodeMCU ESP8266 (9.0).
10. Setelah data diteruskan ke NodeMCU ESP8266, lalu instruksi akan diteruskan dengan mengirimkan data pembacaan sensor ultrasonic ke Arduino Mega 2560 untuk diproses (10.0).
11. Arduino Mega 2560 akan mengirimkan data berupa sinyal digital ke modul i2c untuk diteruskan ke LCD 16x2 (11.0).
12. Selanjutnya modul i2c akan mengirimkan data berupa sinyal digital ke modul LCD 16x2 yang berfungsi sebagai indikator menampilkan nilai persentase jumlah saus tomat (12.0).
13. Arduino Mega 2560 akan mengirimkan data informasi cahaya berupa sinyal digital untuk mengaktifkan LED (13.0).
14. Arduino Mega 2560 akan mengirimkan data berupa sinyal digital untuk mengaktifkan modul relay (14.0).
15. Modul relay akan mengirimkan data berupa sinyal digital untuk mengaktifkan water pump (15.0).

#### 4.1.3 Blok Diagram

Diagram blok merupakan representasi visual yang menunjukkan hubungan antar komponen dalam suatu sistem. Biasanya, diagram ini digunakan untuk menggambarkan sistem secara keseluruhan tanpa menyertakan rincian teknis. Diagram blok sering dimanfaatkan dalam berbagai keperluan, seperti merancang sistem, menganalisis sistem, serta mendokumentasikan sistem. Ilustrasi diagram blok dapat dilihat pada Gambar 4.3.



**Gambar 4. 3 Blok Diagram**

Pada blok diagram di atas dapat dilihat hubungan atau proses yang dilakukan oleh sistem. Perancangan sistem sebagai pengontrolan terhadap semua komponen dengan menggunakan sebuah media pemrosesan yaitu Arduino Mega 2560. Arduino Mega 2560 memiliki lima input yaitu Sensor Ultrasonic, Sensor Load Cell, Modul HX711, dan Keypad 4x4. Sistem ini memiliki beberapa output diantaranya Water Pump, LCD 16x2, LED, Telegram, Modul Relay dan Modul i2c.

#### **4.2 Prinsip Kerja Sistem**

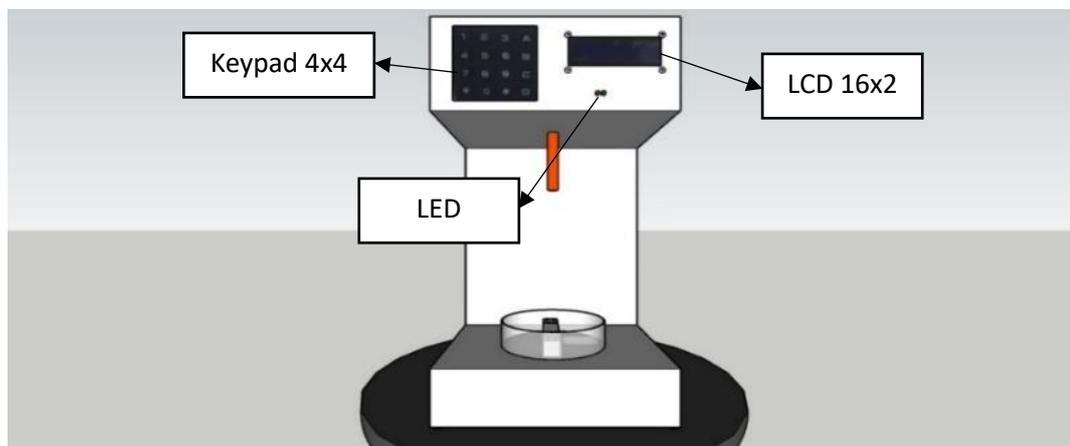
Prinsip kerja sistem merupakan mekanisme bagaimana suatu sistem berfungsi untuk mencapai tujuan tertentu. Berikut dibawah ini akan dijelaskan prinsip kerja alat secara keseluruhan dari keadaan alat mati hingga keadaan alat selesai digunakan. Ketika alat dihubungkan ke sumber tegangan menggunakan kabel steker, alat langsung aktif dan menampilkan persentase ketersediaan saus

tomat pada layar LCD. Setelah itu, nyalakan hotspot pada ponsel atau laptop untuk menghubungkan perangkat.

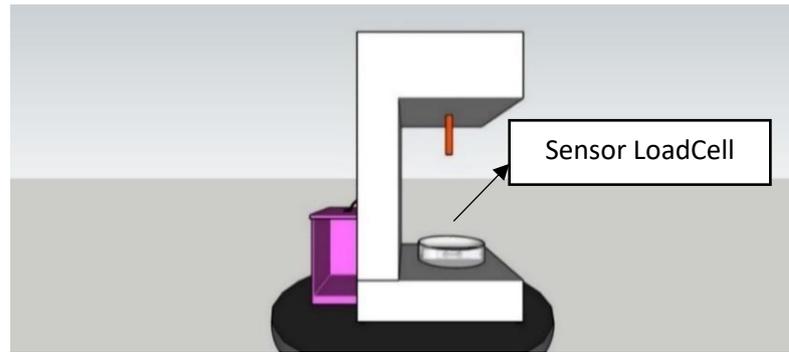
Letakkan wadah di atas sensor loadcell, setelah itu tekan tombol C pada keypad untuk tare, lalu masukkan jumlah saus tomat yang diinginkan dengan menekan jumlah yang diinginkan pada keypad, jumlah tersebut akan ditampilkan pada layar LCD, setelah itu, tekan tombol # untuk mengaktifkan water pump, water pump akan berhenti secara otomatis ketika berat saus yang terdeteksi oleh sensor loadcell mencapai jumlah yang diinputkan, dan jika ingin mengganti jumlah saus tomat, tekan tombol D pada keypad untuk menghapus input sebelumnya lalu inputkan kembali jumlah saus yang ingin diganti.

### 4.3 Rancangan Fisik Sistem

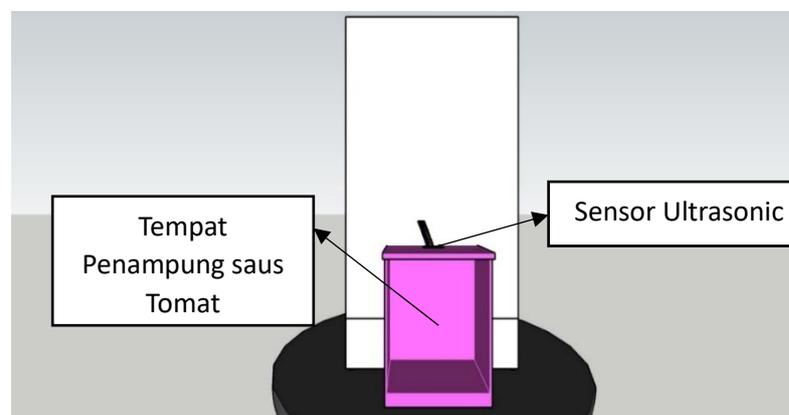
Rancangan fisik alat merupakan tahap awal dari pemasangan dan menganalisa permasalahan yang dihadapi berdasarkan literatur yang menunjang perancangan alat. Rancangan fisik alat dapat dilihat pada Gambar 4.4.



(a) Tampak Depan



(b) Tampak Samping



(c) Tampak Belakang

#### Gambar 4. 4 Rancangan Fisik Sistem

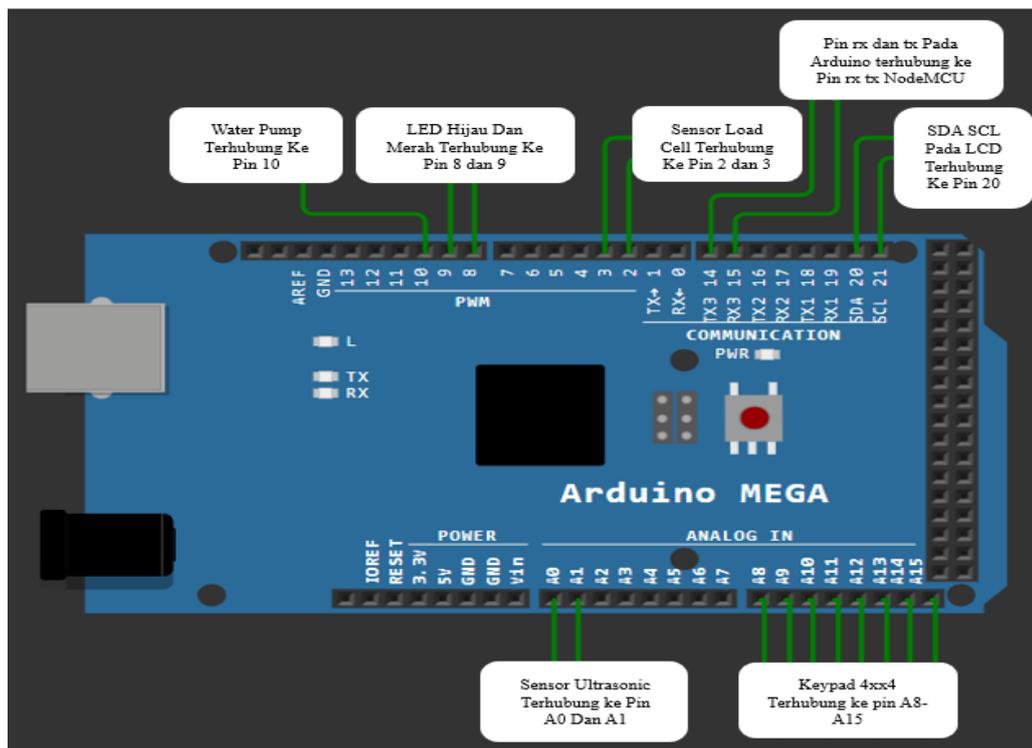
Berdasarkan gambaran rancangan fisik sistem, terlihat bahwa sistem yang akan dibuat telah dirancang sedemikian rupa agar sistem dapat bekerja sebagaimana seharusnya sesuai dengan rancangan awal sistem tanpa menyisakan *entity-entity* penyusun sistem, Pemroses atau pengendali dari sistem yang dirancang menggunakan mikrokontroler Arduino Mega 2560 NodeMCU ESP8266. Sistem yang dirancang memiliki *output*, yaitu Water Pump, LCD 16x2, LED dan NodeMCU ESP8266 sebagai penghubung antara telegram dengan alat.

## 4.4 Desain Terperinci

Desain sistem yang dibuat merupakan representasi menyeluruh dari sistem. Dengan adanya desain ini, prinsip kerja sistem serta komponen-komponen yang terlibat dapat dilihat dengan jelas.

### 4.4.1 Sistem Minimum Arduino Mega 2560

Rangkaian sistem minimum berfungsi untuk menjalankan Arduino Mega 2560 agar dapat bekerja sesuai dengan yang dibutuhkan dimana perencanaannya bertujuan untuk mempermudah pengguna alat. Rangkaian sistem minimum Arduino Mega 2560 dapat dilihat pada Gambar 4.5.



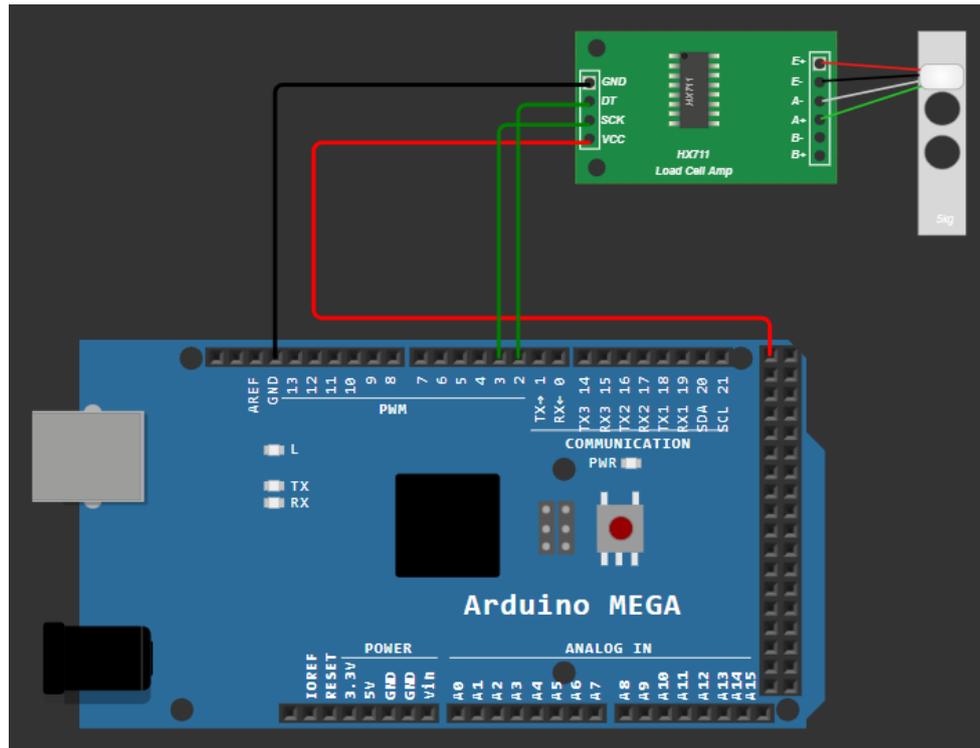
Gambar 4. 5 Rangkaian Sistem Minimum

Berdasarkan pada rangkaian sistem minimum dapat dilihat susunan pin-pin entiti yang terhubung ke Arduino Mega 2560 yaitu sebagai berikut :

1. Pin 10 pada Arduino Mega 2560 terhubung ke anoda pada water pump dan pin GND pada Arduino Mega 2560 terhubung ke katoda pada water pump.
2. Pin 9 dan 8 pada Arduino Mega 2560 terhubung ke anoda pada LED hijau dan anoda pada LED merah, sedangkan pin GND pada Arduino Mega 2560 terhubung ke katoda pada LED hijau dan LED merah.
3. Pin 2 dan 3 pada Arduino Mega 2560 terhubung ke pin DT dan SCK pada sensor Load Cell.
4. Pin 15 dan 14 atau rx,tx pada Arduino Mega 2560 terhubung ke pin rx,tx pada NodeMCU ESP8266.
5. Pin 20 dan 21 atau SDA,SCL pada Arduino Mega 2560 terhubung ke LCD 16x2.
6. Pin A0 dan A1 pada Arduino Mega 2560 terhubung ke pin ECHO dan TRIG pada sensor *ultrasonic*.
7. Pin A8-A15 pada Arduino Mega 2560 terhubung ke keypad 4x4.

#### 4.4.2 Rangkaian Sensor Load Cell Dan Modul HX711

Sensor Load Cell dan modul HX711 berfungsi untuk menimbang berat saus tomat yang akan di ambil. Rangkaian Sensor Load Cell dan modul HX711 dapat dilihat pada Gambar 4.6.



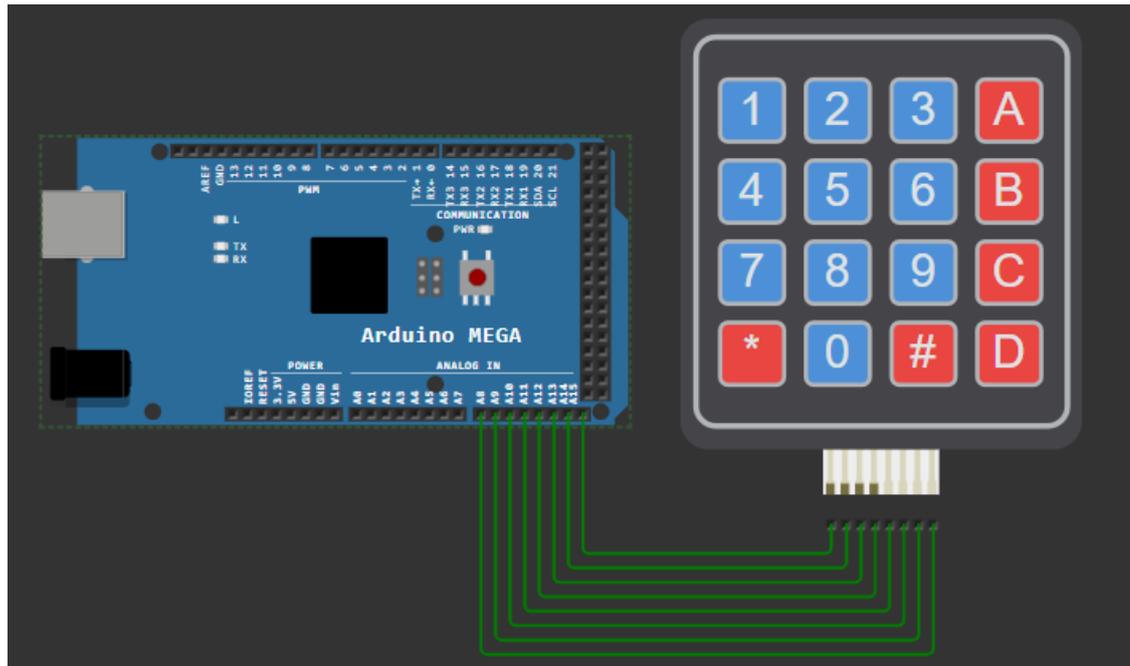
**Gambar 4. 6 Rangkaian Sensor Load Cell Dan Modul HX711**

Berdasarkan pada rangkaian sensor Load Cell dan modul HX711 dapat dilihat susunan pin pada modul HX711 yaitu sebagai berikut :

1. Vcc pada Modul HX711 k terhubung ke vcc Ardino Mega 2560
2. Gnd Modul HX711 terhubung ke Gnd Arduino Mega 2560.
3. Dt pada modul HX711 terhubung ke pin 2 Arduino Mega 2560.
4. Sck pada modul HX711 terhubung ke pin 3 Arduino Mega 2560.

#### 4.4.3 Rangkaian Keypad 4x4

Keypad 4x4 berfungsi untuk menginput jumlah saus tomat yang akan di ambil. Rangkaian Keypad 4x4 dapat dilihat pada Gambar 4.7.



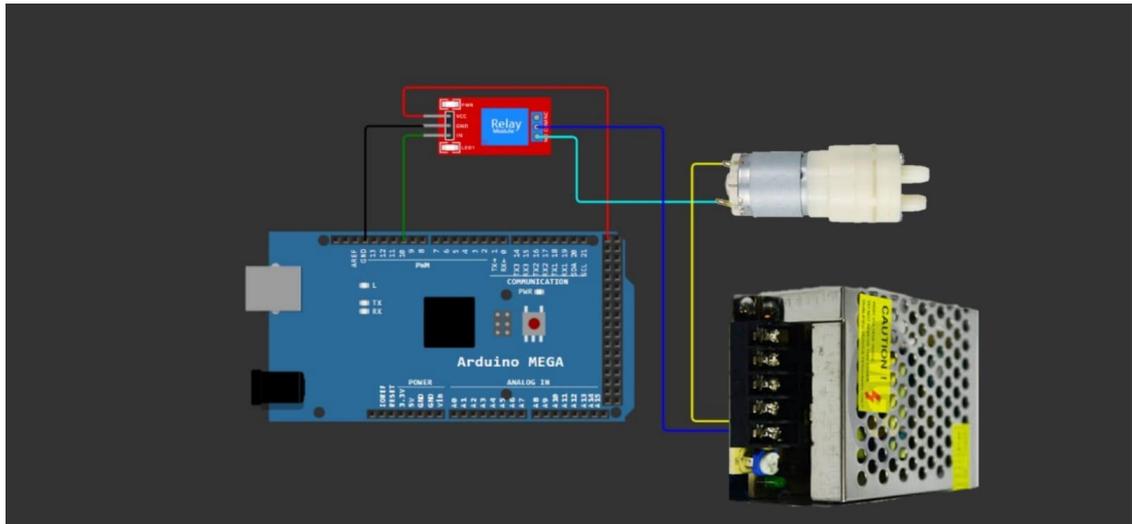
**Gambar 4. 7 Rangkaian Keypad 4x4**

Berdasarkan pada rangkaian keypad 4x4 dapat dilihat susunan pin pada keypad 4x4 yaitu sebagai berikut :

1. R1 pada Keypad 4x4 terhubung ke pin A8 Ardino Mega 2560.
2. R2 pada Keypad 4x4 terhubung ke pin A9 Ardino Mega 2560.
3. R3 pada Keypad 4x4 terhubung ke pin A10 Ardino Mega 2560.
4. R4 pada Keypad 4x4 terhubung ke pin A11 Ardino Mega 2560.
5. C1 pada Keypad 4x4 terhubung ke pin A12 Ardino Mega 2560.
6. C2 pada Keypad 4x4 terhubung ke pin A13 Ardino Mega 2560.
7. C3 pada Keypad 4x4 terhubung ke pin A14 Ardino Mega 2560.
8. C4 pada Keypad 4x4 terhubung ke pin A15 Ardino Mega 2560.

#### 4.4.4 Rangkaian Water Pump Dan Modul Relay

Water Pump Dan Modul Relay berfungsi untuk mengalirkan saus tomat ke dari wadah ke wadah pada sensor load cell untuk ditimbang . Rangkaian Water Pump Dan Modul Relay dapat dilihat pada Gambar 4.8.



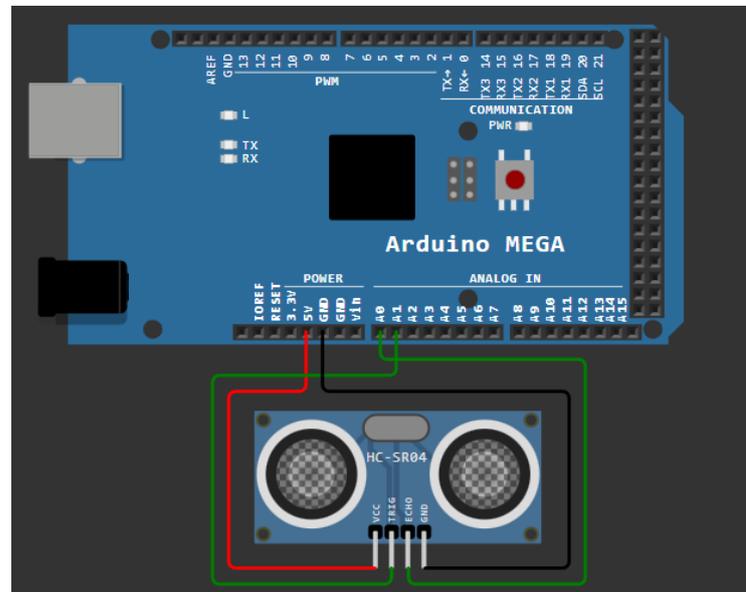
**Gambar 4. 8 Rangkaian Water Pump Dan Modul Relay**

Berdasarkan pada rangkaian *water pump* dan modul relay dapat dilihat susunan pin pada water pump dan modul relay yaitu sebagai berikut :

1. Terminal 1 pada water pump terhubung ke power supply.
2. Terminal 2 pada water pump terhubung ke NO pada modul relay.
3. Com pada modul relay terhubung ke power supply.
4. Vcc pada modul relay terhubung ke vcc pada Arduino Mega 2560.
5. Gnd pada modul relay terhubung ke gnd pada Arduino Mega 2560.
6. In pada modul relay terhubung ke pin 10 Arduino Mega 2560.

#### 4.4.5 Rangkaian Ultrasonic Sensor

Sensor Ultrasonik berfungsi untuk ketersediaan saus tomat pada wadah penampung. Rangkaian Sensor Ultrasonik dapat dilihat pada Gambar 4.9.



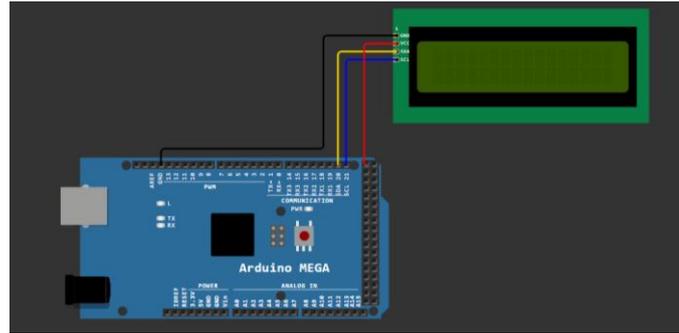
**Gambar 4. 9 Rangkaian Ultrasonic Sensor**

Berdasarkan pada rangkaian sensor ultrasonic dapat dilihat susunan pin pada sensor ultrasonik yaitu sebagai berikut :

1. Vcc pada Sensor Ultrasonic terhubung ke vcc Ardino Mega 2560.
2. Gnd Ultrasonic terhubung ke Gnd Arduino Mega 2560.
3. Trig pada sensor ultrasonic terhubung ke pin A1 pada Arduino Mega 2560.
2. Echo pada sensor ultrasonic terhubung ke pin A0 pada Arduino Mega 2560.

#### 4.4.6 Rangkaian LCD 16x2 Dan Modul I2C

LCD 16x2 berfungsi untuk menampilkan jumlah persentase ketersediaan saus tomat. Rangkaian LCD 16x2 dapat dilihat pada Gambar 4.10.



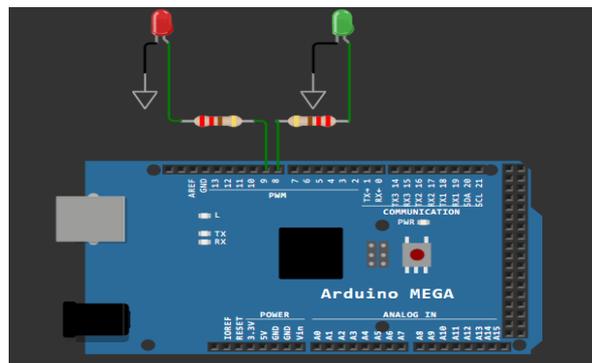
**Gambar 4. 10 Rangkaian LCD 16x2 Dan Modul I2C**

Berdasarkan pada rangkaian LCD 16x2 dan modul i2c dapat dilihat susunan pin pada LCD 16x2 yaitu sebagai berikut :

1. Vcc pada LCD 16x2 terhubung ke vcc Arduino Mega 2560.
2. Gnd LCD16x2 terhubung ke Gnd Arduino Mega 2560.
3. SDA LCD 16x2 terhubung ke pin 20 Arduino Mega 2560.
4. SCL LCD 16x2 terhubung ke pin 21 Arduino Mega 2560.

#### 4.4.7 Rangkaian LED

LED berfungsi sebagai indikator ketersediaan saus tomat. Rangkaian LED dapat dilihat pada Gambar 4.11.



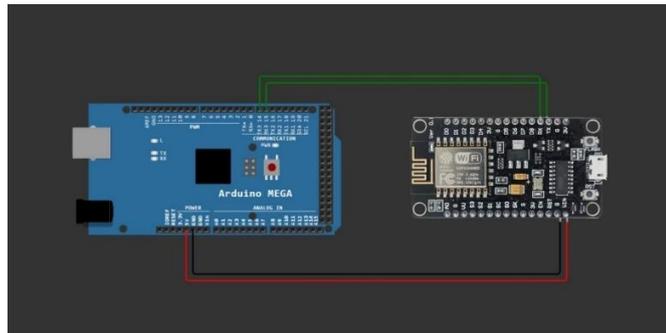
**Gambar 4. 11 Rangkaian LED**

Berdasarkan pada rangkaian LED dapat dilihat susunan pin pada LED yaitu sebagai berikut :

1. Pin anoda LED hijau dan LED merah terhubung ke pin 8 dan 9 pada Arduino Mega 2560.
2. Pin Katoda LED hijau dan LED merah terhubung ke pin gnd pada Arduino Mega 2560.

#### 4.4.8 Rangkaian Node MCU ESP8266

NodeMCU ESP8266 pada perancangan sistem ini berfungsi sebagai penghubung mikrocontroller dan juga koneksi internet (WIFI). Rangkaian Node MCU ESP8266 dapat dilihat pada Gambar 4.12.



**Gambar 4. 12 Rangkaian Node MCU ESP8266**

Berdasarkan pada rangkaian NodeMCU ESP8266 dapat dilihat susunan pin pada NodeMCU ESP8266 yaitu sebagai berikut :

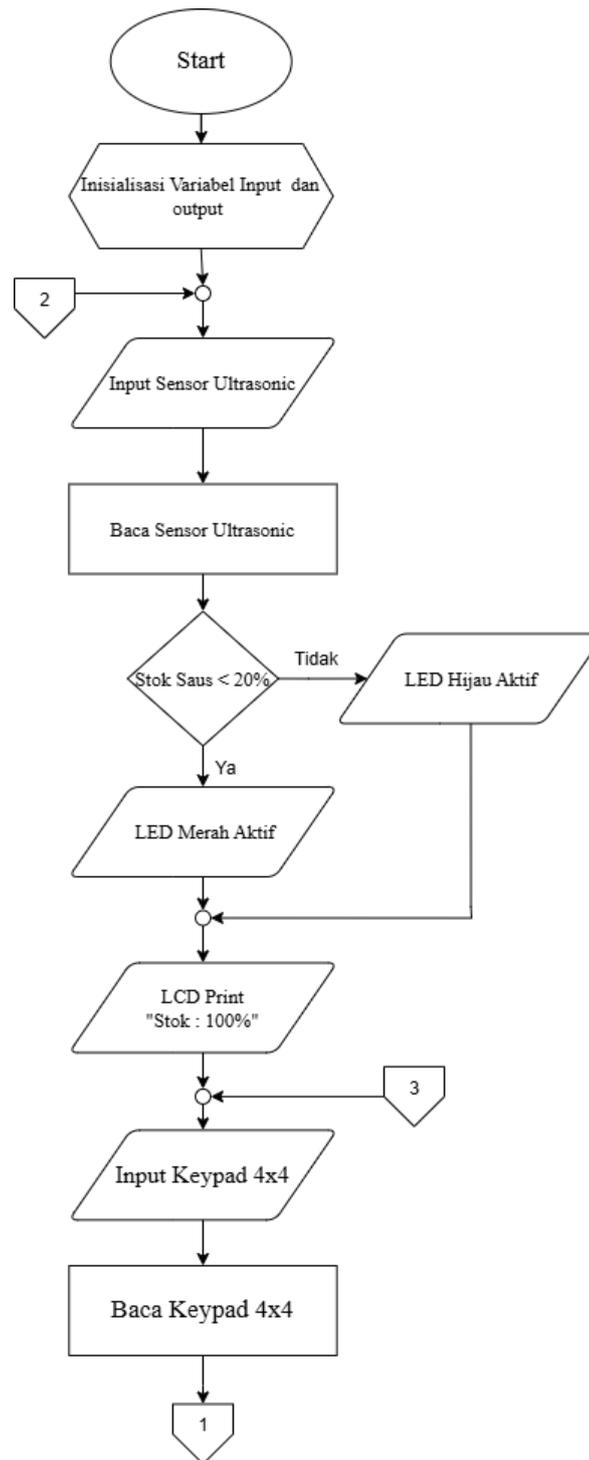
1. Pin rx dan tx NodeMCU terhubung ke pin tx dan rx pada Arduino Mega 2560.
2. Vcc pada NodeMCU terhubung ke pin vcc Arduino Mega 2560.
3. Gnd pada NodeMCU terhubung ke pin gnd Arduino Mega 2560.

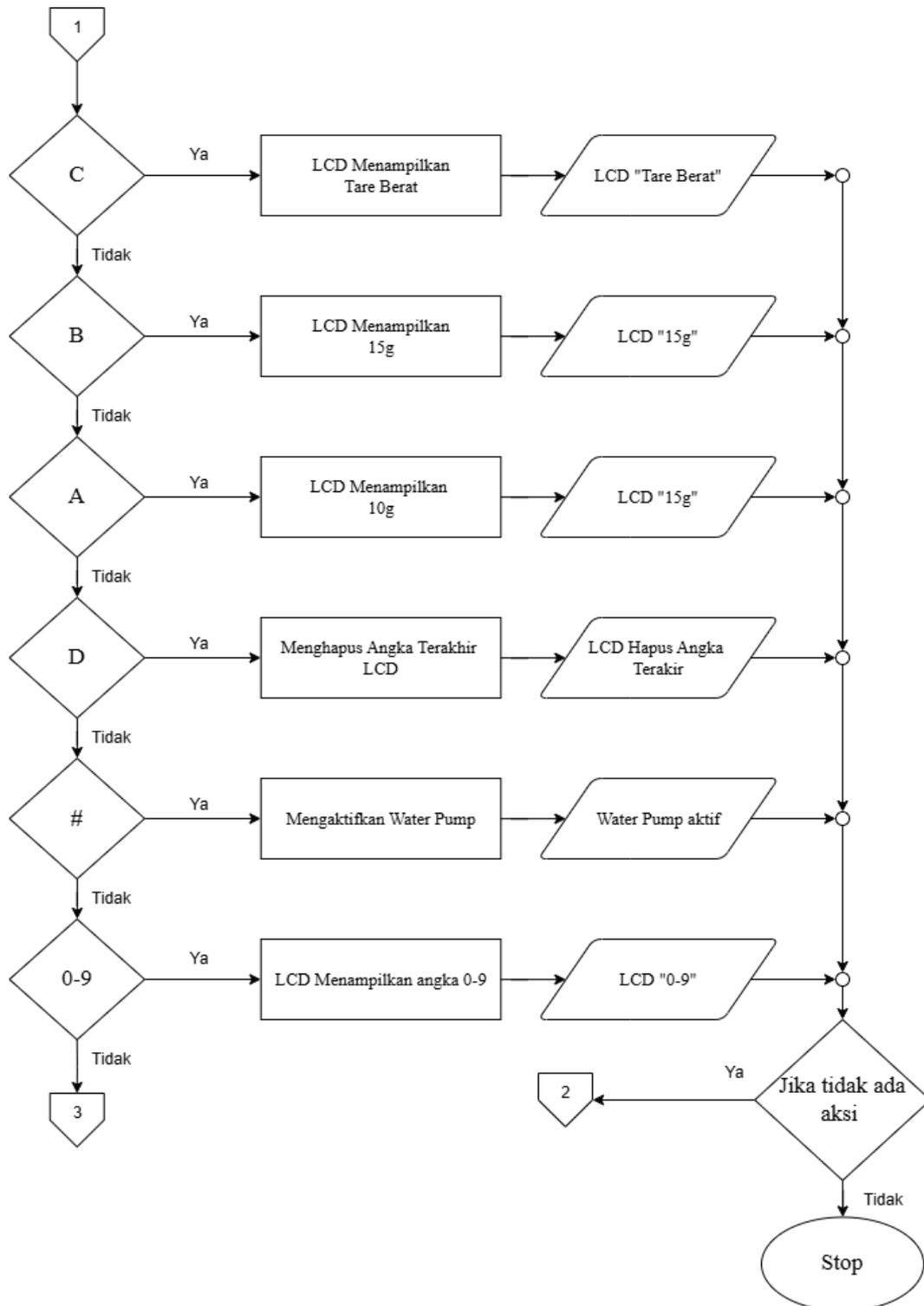
## 4.5 Rancangan Modul Program

Sub bab ini menjelaskan tentang modul program yang digunakan untuk mengontrol kerja dari sistem ini. Untuk lebih mudah dimengerti rancangan modul dapat dibagi menjadi dua bagian yaitu, flowchart dan modul program.

### 4.5.1 Flowchart Arduino Mega 2560

Sebagai langkah awal perancangan program diwujudkan dalam bentuk *flowchart*. Adapun *flowchart* alat yang dimaksud dapat dilihat pada Gambar 4.13.



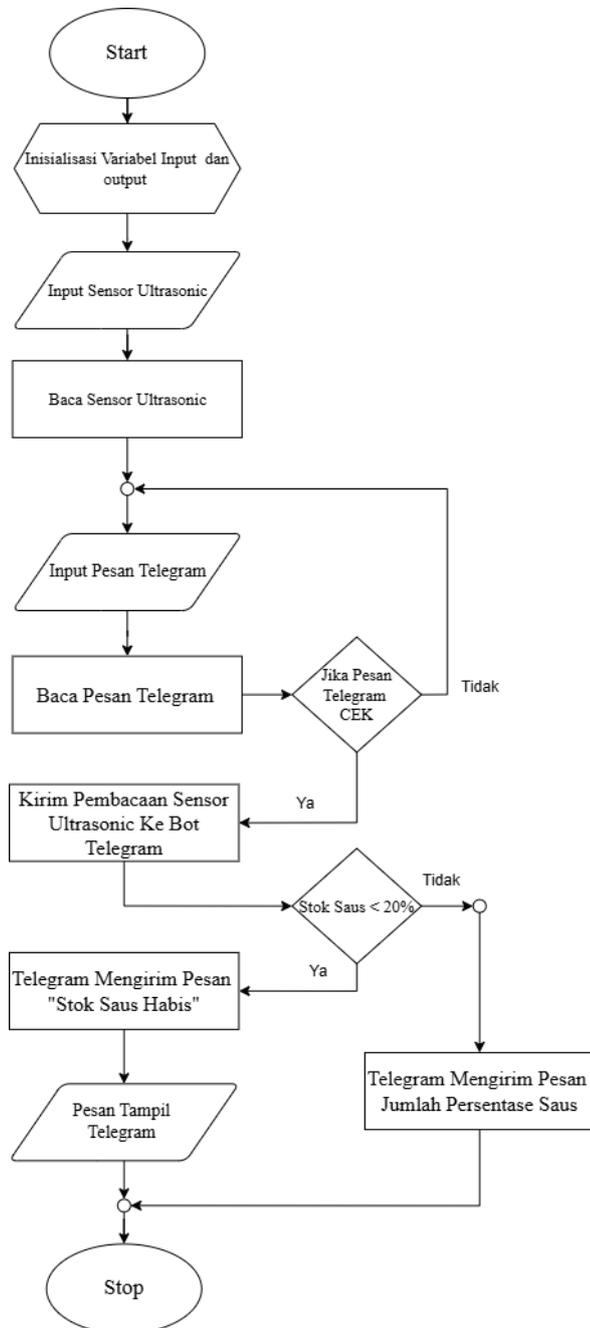


**Gambar 4. 13 Flowchart Arduino Mega**

#### 4.5.2 Flowchart NodeMCU ESP8266

Flowchart NodeMCU ESP8266 berisi langkah-langkah program.

Flowchart NodeMCU ESP8266 dapat dilihat pada Gambar 4.14.



**Gambar 4. 14 Flowchart NodeMCU ESP8266**

### 4.5.3 Modul Program

Berikut ini diuraikan tentang program utama dari sistem yang dirancang adapun modul program adalah sebagai berikut :

#### a. Program NodeMCU ESP8266

```
#include "CTBot.h"
CTBot myBot;
String ssid = "hayolo";
String pass = "1234567890";
String token = "7427694994:AAG20LDmiSuYdhOimwcJWyFSM8siEBsq-7k";
uint8_t led = 2;
int jumlah=0;
int war=4;
int warr=0;
const int id = 684231649;
String data;
char c;
void setup() {
  Serial.begin(115200);
  Serial.println("Starting TelegramBot...");
  myBot.wifiConnect(ssid, pass);
  myBot.setTelegramToken(token);
  pinMode(war, INPUT_PULLUP);
  if (myBot.testConnection())
    Serial.println("\ntestConnection OK");
  else
    Serial.println("\ntestConnection NOK");
}
void loop() {
  warr=digitalRead(war);
  TBMessage msg;
  if (warr==LOW)
  {
    myBot.sendMessage(msg.sender.id, "Stok Saus Habis");
    Serial.println("mengirim notifikasi habis");
  }
  if (myBot.getNewMessage(msg))
```

```

{
  if (warr==LOW)
  {
    myBot.sendMessage(msg.sender.id, "Stok Saus Habis");
    Serial.println("mengirim notifikasi habis");
  }

  if (msg.text.equalsIgnoreCase("CEK"))
  {
    myBot.sendMessage(msg.sender.id, "OK");
    Serial.print("CEK");
  }
  else
  {
    String reply;
    reply = (String)"Welcome " + msg.sender.username + (String)".
Command:CEK";
    myBot.sendMessage(msg.sender.id, reply);
  }
}
while(Serial.available()>0){
  delay(10);
  c = Serial.read();
  data += c;

}
if (data.length()>0) {
  myBot.sendMessage(msg.sender.id, "Stok Saus " + data + " %");
  delay(10);
  data = "";
}
}

```

#### b. Program Load Cell

```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Keypad.h>
#include <HX711.h>
#include <EEPROM.h>
String data;

```

```

String suhu;
char c;
int war=22;
// Inisialisasi LCD 16x2 dengan I2C
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Load cell HX711
HX711 scale;
const int DOUT_PIN = 2; // Ganti nama variabel
const int SCK_PIN = 3; // Ganti nama variabel

// Ultrasonik
const int trigPin = A1;
const int echoPin = A0;

// LED dan pompa
const int ledMerah = 9;
const int ledHijau = 8;
const int pumpPin = 10;

// Keypad 4x4
const byte ROWS = 4;
const byte COLS = 4;
char keys[ROWS][COLS] = {
  { '1', '2', '3', 'A' },
  { '4', '5', '6', 'B' },
  { '7', '8', '9', 'C' },
  { '*', '0', '#', 'D' }
};
byte rowPins[ROWS] = { A15, A14, A13, A12 }; // Connect keypad ROW0,
ROW1, ROW2 and ROW3 to these Arduino pins.
byte colPins[COLS] = { A11, A10, A9, A8 };
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS,
COLS);

// Variabel global
float beratTarget = 0;
float faktorKalibrasi;
long duration;
float distance;

// Alamat EEPROM untuk faktor kalibrasi
const int EEPROM_ADDR = 0;

void setup() {
  lcd.init();
  lcd.backlight();
  Serial.begin(9600);

```

```

Serial3.begin(115200);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
pinMode(ledMerah, OUTPUT);
pinMode(ledHijau, OUTPUT);
pinMode(pumpPin, OUTPUT);
pinMode(war, OUTPUT);

digitalWrite(war, HIGH);
digitalWrite(ledMerah, HIGH);
digitalWrite(ledHijau, HIGH);
digitalWrite(pumpPin, HIGH); // Pompa mati

scale.begin(DOUT_PIN, SCK_PIN); // Menggunakan nama variabel baru
loadKalibrasi();
scale.set_scale(faktorKalibrasi);
scale.tare();

lcd.setCursor(0, 0);
lcd.print("Masukkan berat:");
}

void loop() {
while (Serial3.available() > 0) {
  delay(10);
  c = Serial3.read();
  data += c;
}
if (data.length() > 0) {
  Serial.println(data);
  if (data == "CEK") {
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distance = duration * 0.034 / 2;

    float levelStok = (17 - distance) * 7;
    if (levelStok > 100) {
      levelStok = 100;
    }
    if (levelStok < 0) {
      levelStok = 0;
    }

    Serial.println(levelStok);

```

```

    Serial3.print(levelStok);
  }
  data = "";
}
delay(10);
char key = keypad.getKey();
if (key) {
  switch (key) {
    case 'A':
      beratTarget = 10;
      tampilBeratTarget();
      break;
    case 'B':
      beratTarget = 15;
      tampilBeratTarget();
      break;
    case 'C': // Tare
      scale.tare();
      lcd.clear();
      lcd.print("Tare berhasil");
      delay(1000);
      lcd.clear();
      lcd.print("Masukkan berat:");
      break;
    case 'D': // Hapus angka terakhir
      beratTarget = beratTarget / 10;
      tampilBeratTarget();
      break;
    case '#': // Mulai proses pengisian
      mulaiPengisian();
      break;
    case '*': // Masuk ke mode kalibrasi
      aturKalibrasi();
      break;
    default:
      if (key >= '0' && key <= '9') {
        beratTarget = beratTarget * 10 + (key - '0');
        tampilBeratTarget();
      }
      break;
  }
}
}
/*float beratBaca = scale.get_units(10); // Baca berat dari load cell
Serial.print("Berat: ");
Serial.print(beratBaca);
Serial.println(" g");
*/
bacaStokSaus();

```

```

kontrolLED();
delay(5);
}

```

### c. Program Keypad 4x4

```

char key = keypad.getKey();
if (key) {
  switch (key) {
    case 'A':
      beratTarget = 10;
      tampilBeratTarget();
      break;
    case 'B':
      beratTarget = 15;
      tampilBeratTarget();
      break;
    case 'C': // Tare
      scale.tare();
      lcd.clear();
      lcd.print("Tare berhasil");
      delay(1000);
      lcd.clear();
      lcd.print("Masukkan berat:");
      break;
    case 'D': // Hapus angka terakhir
      beratTarget = beratTarget / 10;
      tampilBeratTarget();
      break;
    case '#': // Mulai proses pengisian
      mulaiPengisian();
      break;
    case '*': // Masuk ke mode kalibrasi
      aturKalibrasi();
      break;
    default:
      if (key >= '0' && key <= '9') {
        beratTarget = beratTarget * 10 + (key - '0');
        tampilBeratTarget();
      }
      break;
  }
}
/*float beratBaca = scale.get_units(10); // Baca berat dari load cell
Serial.print("Berat: ");
Serial.print(beratBaca);
Serial.println(" g");

```

```

*/
  bacaStokSaus();
  kontrolLED();
  delay(5);
}

```

#### d. Program LED

```

void kontrolLED() {
  float levelStok = (20 - distance ) * 5;
  if (levelStok > 20) {
    digitalWrite(ledMerah, HIGH);
    digitalWrite(ledHijau, LOW);
  } else
  {

    digitalWrite(ledMerah, LOW);
    digitalWrite(ledHijau, HIGH);
  }
}

```

#### e. Program LCD 16x2

```

void tampilBeratTarget() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Berat target:");
  lcd.setCursor(0, 1);
  lcd.print(beratTarget);
  lcd.print(" g");}

```

#### f. Program Water Pump

```

void mulaiPengisian() {
  lcd.clear();
  lcd.print("Mengisi...");

  while (scale.get_units() < beratTarget) {
    digitalWrite(pumpPin, LOW); // Pompa aktif (aktif low)
    delay(100);
  }

  digitalWrite(pumpPin, HIGH); // Pompa mati
  lcd.clear();
  lcd.print("Pengisian selesai");
  delay(2000);
  lcd.clear();
}

```

```

    lcd.print("Masukkan berat:");
}

```

#### g. Program Sensor Ultrasonic

```

void bacaStokSaus() {
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distance = duration * 0.034 / 2;

    float levelStok = (17 - distance ) * 7;
    if (levelStok>100)
    {
        levelStok=100;
    }
    if (levelStok<0)
    {
        levelStok=0;
    }
    if (levelStok < 20) {
        digitalWrite(war, LOW);
        Serial.println("habis");
    }
    if (levelStok > 19) {
        digitalWrite(war, HIGH);
    }
    lcd.setCursor(0, 0);
    lcd.print("Stok: ");
    lcd.print(levelStok);
    lcd.print("%  ");}

void aturKalibrasi() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Faktor Kalibrasi");
    delay(2000);

    float nilaiKalibrasi = 0;
    lcd.clear();
    lcd.print("Masukkan faktor:");

    while (true) {
        char key = keypad.getKey();

```

```

if (key >= '0' && key <= '9') {
    nilaiKalibrasi = nilaiKalibrasi * 10 + (key - '0');
    lcd.setCursor(0, 1);
    lcd.print(nilaiKalibrasi);
    lcd.print(" "); // Menghapus karakter lama
} else if (key == 'D') { // Hapus angka terakhir
    nilaiKalibrasi = nilaiKalibrasi / 10;
    lcd.setCursor(0, 1);
    lcd.print(" ");
    lcd.setCursor(0, 1);
    lcd.print(nilaiKalibrasi);
    lcd.print(" "); // Menghapus karakter lama
} else if (key == '#') { // Selesai input faktor kalibrasi
    break;
}
}

faktorKalibrasi = nilaiKalibrasi; // Set faktor kalibrasi dari input
scale.set_scale(faktorKalibrasi);
simpanKalibrasi();

lcd.clear();
lcd.print("Kalibrasi selesai");
delay(1000);
lcd.clear();
lcd.print("Faktor:");
lcd.print(faktorKalibrasi);
delay(2000);
lcd.clear();
lcd.print("Masukkan berat:");
}

void loadKalibrasi() {
    EEPROM.get(EEPROM_ADDR, faktorKalibrasi);
    if (isnan(faktorKalibrasi) || faktorKalibrasi == 0) {
        faktorKalibrasi = 2280.0; // Nilai default
    }
}
}

```

