

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu merupakan penelitian yang diambil sebagai acuan dalam membuat laporan akhir untuk menghindari terjadinya kesamaan pada penelitian yang dilakukan. Penelitian terdahulu juga berguna sebagai bahan perbandingan pada penelitian dan memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Berikut merupakan penelitian terdahulu berupa beberapa jurnal yang terkait dengan judul laporan akhir.

Pada penelitian yang dilakukan oleh Amaliah dan Khofifah Sintya pada tahun 2020 dengan judul “**Rancang Bangun Alat Pendeteksi Kebakaran Rumah Penduduk Menggunakan Esp8266 Berbasis Iot**”, peneliti menggunakan ESP8266 sebagai basis Mikrokontroler lalu disambungkan kepada sensor api, sensor asap, sensor suhu, sensor air, *buzzer*, monitor, *relay*, dan *alarm*. Peneliti menggunakan aplikasi Telegram yang berfungsi sebagai pengirim notifikasi dari ESP8266 kepada pengguna. Peneliti membuat alat yang dapat dipasang pada rumah penduduk yang dapat mendeteksi perubahan suhu pada rumah, mendeteksi api, mendeteksi asap pada saat api timbul dan dapat memadamkan kebakaran secara otomatis. Ketika ada api dan asap yang terdeteksi oleh sensor, maka ESP8266 akan membunyikan *buzzer* yang terpasang dan mengirimkan notifikasi kepada pengguna bahwa kebakaran terdeteksi di dalam rumah lalu akan langsung memadamkan api menggunakan *nozzle* yang tersambung ke mikrokontroler. Kesimpulan yang didapat oleh peneliti berdasarkan penelitian yang dilakukan adalah penggunaan telegram sebagai pengirim notifikasi yang disambungkan ke ESP8266 berhasil mengirim pesan kepada pengguna dan alat dapat memadamkan kebakaran pada titik tertentu didalam rumah yang telah dipasang alat buatan peneliti tersebut. (Amaliah 2020).

Pada penelitian yang dilakukan oleh Dalimunte dan Alfisar Pikri pada tahun 2020 dengan judul “**Mengendalikan Robot *Mobile* Pemadam Kebakaran Menggunakan *Joystick* PS2**”, peneliti menggunakan Arduino Mega sebagai basis

mikrokontroler. Peneliti juga menggunakan modul *bluetooth* yang berguna untuk menyambungkan antara robot dan *joystick* ps 2 yang digunakan sebagai kontrol utama dalam pergerakan robot yang dilakukan oleh operator. Peneliti melakukan pengujian terhadap jarak maksimal yang dapat digunakan untuk mengendalikan robot dengan hasil jarak 1 – 3 meter robot masih dapat dikendalikan dengan sempurna dan pada jarak > 4 meter robot mulai mengalami *packet lost* yang menyebabkan pergerakan robot kurang memuaskan. Peneliti berhasil menggerakkan robot lalu memadamkan api secara manual menggunakan kontroler tersebut dan mendapat hasil yang memuaskan dengan jarak 1-3 meter dari pengguna. (Dalimunte, 2022).

Pada penelitian yang dilakukan oleh Febrian dan Redo pada tahun 2020 dengan judul “**Robot Hexapod Pemadam Api Menggunakan Sensor Uvtron Berbasis Mikrokontroler Atmega 2560**” peneliti menggunakan Mikrokontroler Arduino Mega 2560 sebagai basis dari robot tersebut yang kemudian dihubungkan dengan sensor mic kondesor, sensor *ultrasonik*, sensor *uvtron*, *photodiode* dan sensor kompas. Peneliti menggunakan logika yang mana ketika suara terdeteksi oleh sensor mic kondesor, robot akan mulai berpatroli dengan mengelilingi ruangan menggunakan sensor kompas sebagai penunjuk sudut ruangan, robot akan menghindari objek yang menghalangi dan terdeteksi oleh sensor ultrasonik, lalu ketika robot mendeteksi kebakaran robot menganalisa menggunakan sensor *uvtron*, dan sensor api. Peneliti melakukan pengujian terhadap robot sebanyak 4 kali dengan menggunakan arena sebesar 244cm x 244cm yang terdiri dari 4 ruangan yang masing-masing ditandai dengan garis putih, ketika robot diletakkan robot akan mendeteksi garis putih sebagai ruangan yang berbeda lalu mendeteksi kebakaran yang terjadi pada ruangan tersebut dan memadamkannya, notifikasi kebakaran dapat terlihat pada lcd yang menampilkan detail sudut ruangan, arah kebakaran dan jarak ukur yang terdeteksi oleh sensor. (Febrian 2020).

Pada penelitian yang dilakukan oleh Ilmi dan Kemas Ahmad Miftahul pada tahun 2020 dengan judul “**Mengendalikan Robot Mobile Pemadam Kebakaran Menggunakan Smartphone Android**”, peneliti menggunakan Arduino Mega dan ESP8266 sebagai basis Mikrokontroler yang kemudian disambungkan kepada

Handphone Android menggunakan aplikasi Blynk IoT. Peneliti menggunakan Blynk sebagai kontroler dari smartphone lalu mengirimkan perintah ke ESP8266 yang kemudian disalurkan ke Arduino Mega agar motor dc dapat bergerak sesuai dengan arah yang diinginkan oleh peneliti. Peneliti melakukan pengujian sebanyak 1x pada masing-masing motor dc untuk menguji pengendalian arah gerak pada robot. Peneliti memperoleh kesimpulan bahwa pergerakan dari robot dengan jarak tertentu akan menyebabkan robot delay dalam menerima perintah dari *smartphone* peneliti. (Ilmi 2020).

Pada penelitian yang dilakukan oleh Agustinus Yeheskiel pada tahun 2019 dengan judul “**Merancang Robot Lomba Pemadam Api Menggunakan Arduino Uno**” peneliti menggunakan Arduino Uno sebagai basis Mikrokontroler yang kemudian disambungkan ke sensor *ultrasonik*, sensor garis dan sensor api 5 *channel*. Peneliti menggunakan arena yang berukuran 90cm x 60cm sebagai arena pengujian robot yang diuji sebanyak 30 kali pengujian untuk menguji instruksi tidak menabrak objek yang kemudian mendapatkan hasil rata-rata 90%. Pengujian lalu melakukan uji coba pemadaman robot menggunakan lilin sebagai sumber api yang diuji pada 2 ruangan yang berbeda sebanyak 30 kali dengan tingkat keberhasilan pemadaman api sebesar 100% dan rata-rata waktu yang dibutuhkan untuk memadamkan api yaitu 4 detik. Adapun kesimpulan yang didapatkan dari penelitian ini ialah sensor dapat bekerja dengan baik dengan tingkat keberhasilan tinggi dapat menghindari objek-objek yang menghalangi robot, lalu dapat memadamkan api dengan cepat. Untuk kekurangan yang disampaikan peneliti yaitu sensor api masih mengalami permasalahan dalam mendeteksi cahaya api dan cahaya matahari sehingga dibutuhkan perbaikan untuk kedepannya dan untuk gerak robot peneliti berharap agar dapat menambah roda agar gerak robot tidak menjadi zig- zag. (Agustinus Yeheskiel, 2019).

Pada penelitian yang dilakukan oleh Amir Dhiman, dll. pada tahun 2022 yang sudah diterjemahkan dengan judul “**Fire fighter robot with deep learning and machine vision**” peneliti menggunakan Raspberry pi 4 sebagai basis Mikrokontroler yang dihubungkan ke sensor ultrasonik, modul kamera, modul GSM. Peneliti menggunakan servo sebagai aktuator dalam memadamkan api

menggunakan pemadam api *portable*. Peneliti mengimplementasikan *deep learning and machine vision* yaitu sebuah implementasi robot yang mengandalkan modul kamera dalam mendeteksi objek. Peneliti melakukan pengujian terhadap 6 jenis kebakaran yang berbeda dengan tingkat keberhasilan secara total rata-rata yang dicapai 91,833%. Cara kerja robot yang dijelaskan oleh peneliti ialah menggunakan modul kamera dengan implementasi *machine vision* untuk mendeteksi cahaya api jadi ketika api terdeteksi maka robot akan melakukan pencocokan dengan 6 klasifikasi kebakaran, lalu robot akan bergerak menuju api dan stop pada jarak 15cm dari api. Jika api tidak kunjung padam maka robot akan mengirimkan notifikasi melalui sms kepada *user* menggunakan GSM modul karena api yang dideteksi sudah tidak dapat dikendalikan oleh robot. (Dhiman dkk., 2022).

Pada penelitian yang dilakukan oleh Sarita C. dan B. P. Harish pada tahun 2020 yang sudah diterjemahkan dengan judul “***Firefighting Robot with Face Detection System***” peneliti menggunakan Arduino sebagai basis Mikrokontroler yang kemudian disambungkan kepada sensor *infrared*, sensor api, sensor asap. Peneliti menambahkan lcd pada robot yang berfungsi menampilkan pesan ketika robot berhasil memadamkan api. Peneliti menggunakan modul kamera dalam pembuatan robot yang berfungsi mendeteksi wajah manusia, ini dimaksudkan agar robot dapat mendeteksi keberadaan manusia yang terperangkap dalam api lalu mengirimkan notifikasi kepada *operator* menggunakan aplikasi yang dibuat pada *software* MATLAB. Peneliti membuat robot agar dapat memadamkan api di dalam ruangan di area yang sulit terjangkau oleh manusia lalu robot akan kembali bergerak ke sekitarnya untuk mendeteksi kebakaran, kebocoran maupun mendeteksi wajah korban kebakaran pada bangunan. (Harish, 2020).

Pada penelitian yang dilakukan oleh Md Abdullah Al Rakib, dll. pada tahun 2022 yang sudah diterjemahkan dengan judul “***Fire Detection and Water Discharge Activity for Fire Fighting Robots using IoT***” peneliti mengangkat permasalahan bencana kebakaran yang terjadi pada Bangladesh dengan rentang tahun 2006 sampai 2018 yang terus meningkat. Kejadian ini terus meningkat seiring dengan pertumbuhan penduduk dan pemukiman yang padat sehingga pihak setempat kesulitan dalam memadamkan kebakaran. Peneliti kemudian membuat

robot pemadam api dengan tujuan dapat memadamkan kebakaran pada tempat yang sulit terjangkau oleh manusia. Peneliti menggunakan Arduino Mega sebagai basis mikrokontroler pada robot dan menggunakan sensor api untuk mendeteksi cahaya dari api, sensor ultrasonik untuk mendeteksi objek yang menghalangi gerak robot, dan pompa air yang digunakan untuk memadamkan kebakaran. Dalam membuat robot prototype ini, peneliti berhasil menguji robot menghindari objek yang terdeteksi sensor ultrasonik, lalu dengan otomatis menuju ke lokasi kebakaran yang terdeteksi oleh sensor api dengan menetapkan jarak sekitar 20 cm dari api robot dapat menyemburkan air langsung ke api yang terdeteksi. (Al Rakib dkk., 2022).

Pada penelitian yang dilakukan oleh Ionel-Alexandru GAL, dll. pada tahun 2018 yang sudah diterjemahkan dengan judul "*Fuzzy Decision Algorithm For Hybrid Fire Fighting Robot*", peneliti membuat robot hybrid menggunakan Arduino Mega sebagai basis mikrokontrollernya. Peneliti menggunakan logika fuzzy ke dalam robot yang dibuat guna mendeteksi jenis landasan tempat tujuan kebakaran yang dideteksi oleh robot. Logika yang digunakan oleh peneliti ialah ketika robot mendeteksi kebakaran menggunakan sensor api, robot akan menuju ke tempat kebakaran dengan menggunakan sensor ultrasonik yang berguna mendeteksi halangan robot. Ketika robot mencapai jenis dataran tertentu, robot akan mengkalkulasikan kemungkinan dan menggunakan logika fuzzy yang ditetapkan untuk menggunakan jenis kaki tambahan pada robot dan pengaturan pergerakan pada robot. Percobaan dilakukan oleh peneliti terhadap 2 jenis dataran yaitu, dataran rata dan tidak rata dengan hasil dataran rata robot berjalan menggunakan roda karena permukaan yang mulus dan hasil dataran tidak rata robot berjalan menggunakan kaki yang terpasang dengan 2 jenis pergerakan. (Gal dkk., 2018).

Pada penelitian yang dilakukan oleh Md. Mahmudul Hasan dan Nayeem Al-Tamzid Bhuiyan pada tahun 2020 yang sudah diterjemahkan dengan judul "*A Multi-Sensor Based Fire Fighting Robot With Wireless Control And Visual System*", peneliti menggunakan gabungan antara Arduino Mega dan NodeMCU sebagai mikrokontroler. Peneliti juga menggunakan 4 sensor api, 2 sensor ultrasonik, dan pompa air didalam robot guna mendeteksi dan memadamkan

kebakaran yang terjadi. Peneliti membuat cara kerja robot menjadi 2, yaitu penggunaan NodeMCU dan Blynk lalu Arduino Mega (otomatisasi). Penggunaan NodeMCU dan Blynk digunakan oleh peneliti agar dapat mengontrol robot secara manual dengan mendeteksi api lalu mengirimkan notifikasi kebakaran yang terdeteksi kepada *smartphone user*. Lalu ketika *user* mengaktifkan mode otomatis, robot akan menggunakan Arduino Mega sebagai mikrokontroler utama dengan mendeteksi api lalu menuju ke daerah kebakaran untuk memadamkan api, lalu mengirimkan sms notifikasi kepada *user* ketika terdeteksi tanda-tanda makhluk hidup. (Md. Mahmudul Hasan & Nayeem Al-Tamzid Bhuiyan, 2020).

2.2 Mikrokontroler

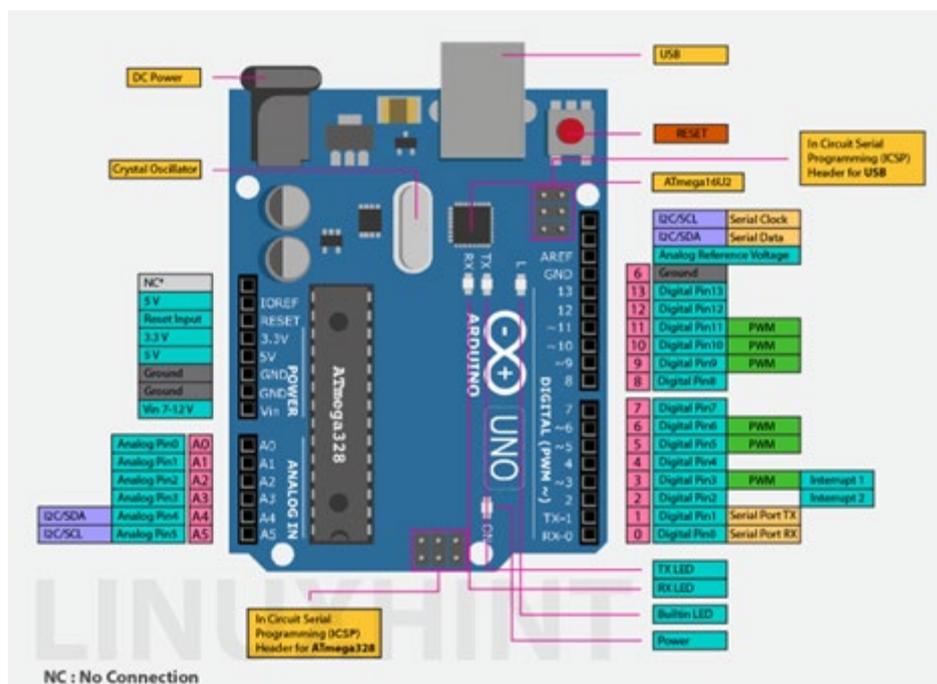
Mikrokontroler atau kadang dinamakan pengontrol tertanam (*embedded controller*) adalah suatu sistem yang mengandung masukan atau keluaran, memori, dan prosesor yang digunakan pada produk seperti mesin cuci, pemutar video, mobil dan telepon. Pada prinsipnya, mikrokontroler adalah sebuah komputer berukuran kecil yang dapat digunakan untuk mengambil keputusan, melakukan hal-hal bersifat berulang dan dapat berinteraksi dengan peranti-peranti eksternal, seperti sensor ultrasonik untuk mengukur jarak terhadap suatu objek, penerima *Global Positioning System* (GPS) untuk memperoleh data posisi kebumian dari satelit dan motor untuk mengontrol gerak pada robot. Sebagai komputer yang berukuran kecil, Mikrokontroler cocok diaplikasikan pada benda-benda yang berukuran kecil, misalnya sebagai pengendali pada robot. (Suhaeb dkk., 2018).

2.3 Arduino

Menurut Abdul Kadir (2019) dalam bukunya yang berjudul “Simulasi Arduino” berpendapat bahwa Arduino merupakan perangkat keras sekaligus sebagai perangkat lunak yang memungkinkan siapa pun untuk membuat *prototype* rangkaian elektronik berbasis mikrokontroler dengan mudah dan cepat.

Arduino adalah mikrokontroller *single-board* yang bersifat *open-source*, diturunkan dari *Wiring platform*, mempunyai fleksibilitas yang tinggi baik dari segi *software* maupun *hardware* untuk memudahkan rancang bangun elektronik dalam berbagai bidang.

Adapun jenis arduino ada beberapa macam, yang digunakan peneliti dalam melakukan penelitian ini adalah Arduino Uno. Arduino Uno adalah papan mikrokontroler berbasis ATmega328 (*datasheet*). Memiliki 14 pin *input* dari *output* digital dimana 6 pin *input* tersebut dapat digunakan sebagai *output* PWM dan 6 pin *input* analog, 16 MHz osilator kristal, koneksi USB, *jack power*, ICSP *header*, dan tombol *reset*. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan papan Arduino Uno ke komputer dengan menggunakan kabel USB atau listrik dengan AC yang ke *adaptor* DC atau baterai untuk menjalankannya.



Gambar 2. 1 Arduino Uno

2.4 Sensor Jarak Ultrasonik HC-SR04

Sensor *ultrasonik* adalah alat elektronika yang kemampuannya bisa mengubah dari energi listrik menjadi energi mekanik dalam bentuk gelombang suara *ultrasonik*. Sensor HC-SR04 merupakan salah satu sensor ultrasonik yang sering digunakan untuk memantau jarak benda (objek) dengan sensor. Sensor ini terdiri dari rangkaian pemancar ultrasonik yang dinamakan *transmitter* dan penerima *ultrasonik* yang disebut *receiver*. Jarak yang bisa ditangani berkisar antara 2 cm hingga 400 cm, dengan tingkat presisi sebesar 0,3 cm. sudut deteksi yang bisa ditangani tidak lebih dari 15°. Arus yang dibutuhkan tidak lebih dari 2mA dan

tegangan yang dibutuhkan sebesar +5V. Jumlah pin adalah 4. (Purwanto dkk., 2019).

Untuk mengetahui konfigurasi masing-masing pin pada sensor HC-SR04 dapat dilihat pada Gambar berikut.



Gambar 2. 2 Sensor Ultrasonik HC-SR04

Pada Tabel dibawah menunjukkan keterangan masing-masing pin untuk sensor ultrasonic HC-SR04.

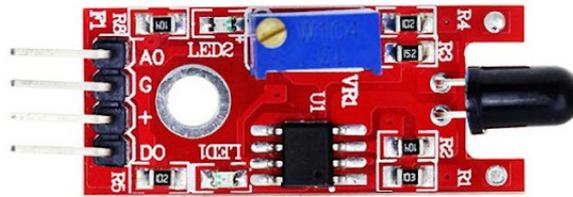
Tabel 2. 1 Tabel Pin Sensor Ultrasonik HC-SR-04

Pin	Keterangan
Pin 1	VCC (dihubungkan ke tegangan +5V)
Pin 2	Trig (untuk mengirimkan gelombang suara)
Pin 3	Echo (untuk menerima pantulan gelombang suara)
Pin 4	GND (pin ini dihubungkan ke <i>ground</i>)

2.5 Sensor Api

Sensor api digunakan untuk mendeteksi api atau radiasi. Sensor ini juga dapat mendeteksi sumber cahaya yang memiliki panjang gelombang antara 760 nm hingga 1100 nm. Infra merah merupakan warna dari cahaya tampak dengan panjang gelombang 700 nm sampai 1 mm. (Maspiyanti & Hadiyanti, 2021).

Sedangkan cahaya *ultraviolet* memancarkan cahaya dengan panjang gelombang sekitar 300 nm – 400 nm. Sensor ini bisa mendeteksi cahaya tampak, sinar infra merah dan sinar ultraviolet. Prinsip kerja sensor api adalah dimulai dari bahwa api akan bisa dideteksi oleh keberadaan spectrum cahaya *infrared* maupun *ultraviolet*, dan dari situ semacam sensor dalam sensor api akan bekerja untuk membedakan spektrum cahaya yang terdapat pada api yang terdeteksi tersebut.



Gambar 2. 3 Sensor Api

Fungsi dari pin sensor api bisa dilihat pada tabel berikut.

Tabel 2. 2 Tabel Fungsi Pin Sensor Api

Pin	Keterangan
Pin 1	VCC (dihubungkan ke tegangan +5V)
Pin 2	GND (pin ini dihubungkan ke <i>ground</i>)
Pin 3	DO (pin ini dihubungkan ke pin digital, dan memberikan keluaran berbentuk digital (<i>LOW</i> atau <i>HIGH</i>))
Pin 4	A0 (yang dihubungkan ke pin analog <i>input</i> , dan memberikan nilai <i>integer</i> antar 0 dan 1023.)

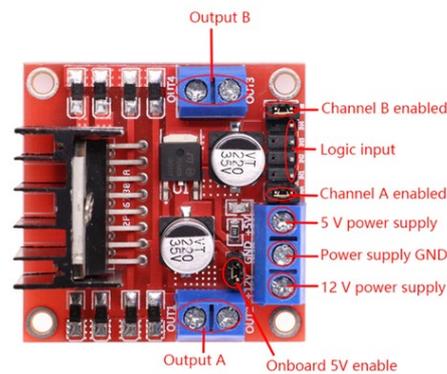
2.6 *Module Motor Driver L298N*

Driver motor L298N merupakan modul driver motor DC yang paling banyak digunakan atau dipakai di dunia elektronika yang difungsikan untuk mengontrol kecepatan serta arah perputaran motor DC. IC L298N merupakan sebuah IC tipe H-bridge yang mampu mengendalikan beban-beban induktif seperti relay, solenoid, motor DC dan motor stepper. Kelebihan akan modul driver motor L298N ini yaitu dalam hal kepresisian dalam mengontrol motor sehingga motor lebih mudah untuk dikontrol. (Imam Rama Muttaqin & Dian Budhi Santoso, 2021).

IC L298 merupakan sebuah IC tipe H-bridge yang mampu mengendalikan beban-beban induktif seperti relay, solenoid, motor DC dan motor stepper. Pada ICL298n terdiri dari transistor-transistor logic (TTL) dengan gerband nandyang berfungsi untuk memudahkan dalam menentukan arah putaran suatu motor DC maupun motor stepper. Adapun Spesifikasi modul motor Driver Motor L298N yaitu sebagai berikut :

1. Menggunakan IC L298N (DoubleH bridge Drive Chip)
2. Tegangan minimal untuk masukan power antara 5V-35V
3. Tegangan operasional 5Vd

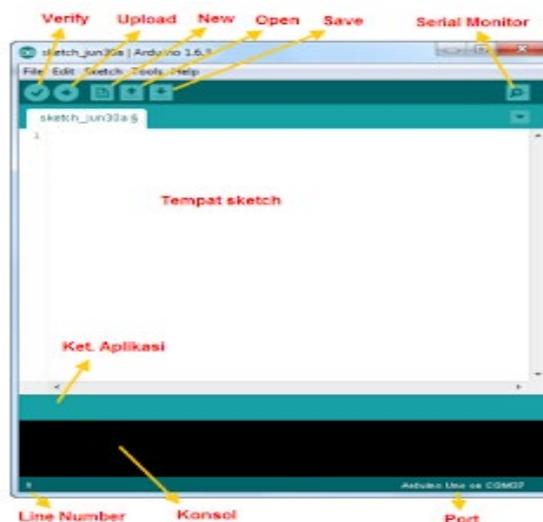
4. Arus untuk masukan antara 0-36mA
5. Arus maksimal untuk keluaran per output A maupun B yaitu 2A
6. Daya maksimal yaitu 25W
7. Dimensi modul yaitu 43 x 43 x 26mm
8. Berat : 26g (Yulio dkk, 2021).



Gambar 2. 4 Module Motor Driver L298N

2.7 IDE Arduino

Sketch merupakan kode program Arduino yang dibuat menggunakan bahasa pemrograman C. Program atau *sketch* yang sudah selesai ditulis di Arduino IDE (*Integrated Development Environment*) bisa langsung *dicompile* dan *diupload* ke *Arduino Board*. (Abdul Kadir, 2019).



Gambar 2. 5 Program Arduino IDE

Berikut adalah penjabaran dari bagian-bagian yang ada pada menu aplikasi Arduino IDE:

1. **Verify** pada versi sebelumnya dikenal dengan istilah *Compile*. Sebelum aplikasi di-*upload* ke *board* Arduino, biasakan untuk memverifikasi terlebih dahulu *sketch* yang dibuat. Jika ada kesalahan pada *sketch*, nanti akan muncul *error*. Proses *Verify / Compile* mengubah *sketch* ke *binary code* untuk di-*upload* ke mikrokontroler.
2. **Upload** tombol ini berfungsi untuk mengupload *sketch* ke *board* Arduino. Walaupun kita tidak mengklik tombol *verify*, maka *sketch* akan di-*compile*, kemudian langsung di-*upload* ke board. Berbeda dengan tombol *verify* yang hanya berfungsi untuk memverifikasi *source code* saja.
3. **New Sketch** Membuka *window* dan membuat *sketch* baru.
4. **Open Sketch** Membuka *sketch* yang sudah pernah dibuat. *Sketch* yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi file *.ino*
5. **Save Sketch** menyimpan *sketch*, tapi tidak disertai dengan *mengcompile*.
6. **Serial Monitor** Membuka *interface* untuk komunikasi *serial*, nanti akan kita diskusikan lebih lanjut pada bagian selanjutnya.
7. **Keterangan Aplikasi** pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal “*Compiling*” dan “*Done Uploading*” ketika kita *mengcompile* dan *mengupload sketch* ke *board* Arduino
8. **Konsol log** Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang *sketch* akan muncul pada bagian ini. Misal, ketika aplikasi *mengcompile* atau ketika ada kesalahan pada *sketch* yang kita buat, maka informasi *error* dan baris akan diinformasikan di bagian ini.
9. **Baris Sketch** bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada *sketch*.
10. **Informasi Board dan Port** Bagian ini menginformasikan *port* yang dipakai oleh *board* Arduino.

Sketch dalam Arduino secara sederhana dikelompokkan menjadi 3 blok, seperti contoh gambar yang ada diatas :

1. *Header*

Merupakan bagian deklarasi awal yang berisi definisi penting yang akan digunakan dalam program, misal contoh dalam pendefinisian variable dalam menambahkan file dan penggunaan *library yang akan digunakan*. Kode pada blok hanya dijalankan sekali pada waktu menyusun program.

2. *Setup*

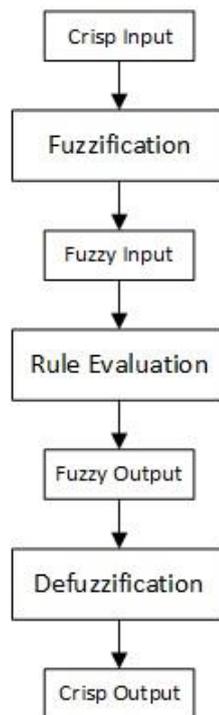
Suatu pin dapat difungsikan sebagai *output* ataupun *input*. Jika pin tersebut memiliki *impedance* yang tinggi dan siap menerima arus yang dikirimkan maka fungsinya sebagai *input*,. Sedangkan jika siap mengirimkan arus listrik maksimum 100 mA (mili ampere) kepada beban yang disambungkan maka difungsikan sebagai *output*.

3. *Loop*

Pada bagian ini akan dilakukan looping (dieksekusi secara terus menerus). Ketika program sudah sampai akhir, program akan mengulang dari awal eksekusi blok. Apabila tombol power Arduino di matikan maka program akan terhenti. Disini fungsi utama program Arduino berada

2.8 Logika Fuzzy

Fuzzy logic pertama kali dikembangkan oleh Prof. Lotfi A. Zadeh, seorang peneliti dari Universitas California, pada tahun 1960-an. *Fuzzy logic* dikembangkan dari teori himpunan fuzzy. Himpunan fuzzy adalah pengelompokan sesuatu berdasarkan variabel bahasa (linguistik variable), yang dinyatakan dengan fungsi keanggotaan, dalam semesta U. Keanggotaan suatu nilai pada himpunan dinyatakan dengan derajat keanggotaan yang nilainya antara 0.0 sampai 1.0. (Endra dkk., 2020).



Gambar 2. 6 Proses Fuzzy Logic

Pada Gambar 2.5 merupakan langkah-langkah yang dilakukan pada metode *Fuzzy Logic*.

1. *Crisp input* → *input* awal berupa jarak dengan satuan cm, yang diperlukan untuk melakukan proses.
2. *Fuzzification* → proses untuk mengubah variabel non fuzzy (satuan cm) menjadi variabel fuzzy.
3. *Fuzzy input* → nilai fuzzy yang merupakan hasil dari proses *fuzzification*.
4. *Rule evaluation* → proses evaluasi nilai yang dihasilkan terhadap *rules* yang telah ditentukan. Dalam penelitian ini melakukan evaluasi *rules* dengan operator dasar UNION, dengan penggunaan operator AND. Terjadi ketika salah satu maupun kedua elemen himpunan terpenuhi.
5. *Fuzzy Output* → variabel fuzzy yang merupakan hasil keluaran dari proses *rules evaluation*.
6. *Defuzzification* → proses pengubahan besaran fuzzy dalam bentuk himpunan fuzzy untuk mendapat kembali bentuk tegasnya (*crisp*). Pada penelitian ini metode yang digunakan pada perhitungan *velocity* Robot pada penelitian ini adalah metode *weighted average* Fuzzy Tsukamoto, yakni

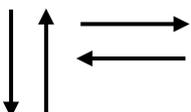
sebuah metode yang mengambil nilai rata-rata dari ketetapan *velocity* dan waktu *delay* yang telah dibuat sebelumnya untuk pemrograman perintah Robot Pemadam Api.

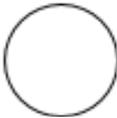
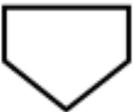
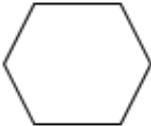
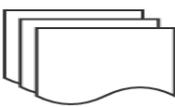
7. *Crisp output* → hasil keluaran proses *defuzzification* dalam bentuk nilai tegasnya.

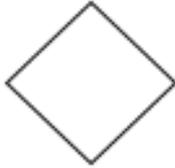
2.9 Flowchart

Menurut Santoso,dkk (2019) *Flowchart* adalah suatu diagram yang mendeskripsikan suatu algoritma atau prosedur dalam menyelesaikan masalah tertentu, dengan menggunakan *flowchart* akan memudahkan pengguna dalam melakukan pengecekan atau analisis suatu masalah, fungsi lain dari *flowchart* adalah sebagai fasilitas untuk memudahkan dalam memahami program yang dibuat antara pemrogram satu dengan yang lainnya dalam tim suatu proyek sehingga dapat memahami beberapa urutan logika yang rumit dan panjang. Selain itu, *flowchart* juga dapat membantu mengkomunikasikan jalannya sistem atau program yang dibuat kepada orang lain yang bukan seorang pemrogram.

Tabel 2. 3 Simbol-Simbol *Flowchart*

NO	Nama	Simbol	Keterangan
1.	Flow Direction Symbol		Untuk menghubungkan antara simbol yang satu dengan simbol yang lain atau menyatakan jalannya arus dalam suatu proses.
2.	Terminal (mulai atau berhenti)		Simbol ini digunakan untuk menunjukkan awal kegiatan (<i>start</i>) atau akhir dari suatu kegiatan (<i>stop</i>).
3.	Input dan Output		Untuk menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.
4.	Proses (Pengolahan)		Untuk menunjukkan pengolahan yang dilakukan oleh komputer atau PC.

NO	Nama	Simbol	Keterangan
5.	Connector		Simbol suatu keluaran atau masukan prosedur atau proses dalam lembar atau halaman yang sama.
6.	Offline Connector		Simbol untuk keluaran atau masukan prosedur atau proses dalam lembar atau halaman yang berbeda.
7.	Document		Untuk menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak ke kertas.
8.	Manual Input		Berfungsi untuk pemasukan data secara manual <i>on-line keyboard</i> .
9.	Preparation		Berfungsi untuk mempersiapkan penyimpanan yang sedang/ akan digunakan sebagai tempat pengolahan didalam <i>storage</i> .
10.	Manual Operation		Berfungsi untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer atau PC.
11.	Multiple Document		Sama seperti symbol document, hanya saja dokumen yang digunakan lebih dari satu dalam simbol ini.
12.	Predefined		Untuk pelaksanaan suatu bagian (subprogram) / prosedur.
13.	Display		Simbol yang digunakan untuk menyatakan perangkat output yang digunakan.

NO	Nama	Simbol	Keterangan
14.	Decision (Keputusan)		Menunjukkan suatu perbandingan yang harus dibuat bila hasilnya “ya”, maka alir data menunjukkan ke suatu tempat, bila “tidak” maka akan menuju ke tempat lain.
15.	Database		Simbol yang digunakan untuk menyatakan data disimpan dalam database.