



**PERANCANGAN ALGORITMA PEMILAH SAMPAH OTOMATIS  
MENGUNAKAN *INDUCTIVE PROXIMITY SENSOR*,  
*CAPACITIVE PROXIMITY SENSOR*, DAN *INFRARED SENSOR***

**SKRIPSI**

**Fajar Ajis Saputra**

**NPM 16650076**

**PROGRAM STUDI TEKNIK MESIN  
FAKULTAS TEKNIK DAN INFORMATIKA  
UNIVERSITAS PGRI SEMARANG**

**2022**



**PERANCANGAN ALGORITMA PEMILAH SAMPAH OTOMATIS  
MENGUNAKAN *INDUCTIVE PROXIMITY SENSOR*,  
*CAPACITIVE PROXIMITY SENSOR*, DAN *INFRARED SENSOR***

**SKRIPSI**

**Diajukan kepada Fakultas Teknik dan Informatika  
Universitas PGRI Semarang untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Teknik**

**Fajar Ajis Saputra  
NPM 16650076**

**PROGRAM STUDI TEKNIK MESIN  
FAKULTAS TEKNIK DAN INFORMATIKA  
UNIVERSITAS PGRI SEMARANG**

**2022**

**SKRIPSI**

**PERANCANGAN ALGORITMA PEMILAH SAMPAH OTOMATIS  
MENGUNAKAN *INDUCTIVE PROXIMITY SENSOR*,  
*CAPACITIVE PROXIMITY SENSOR*, DAN *INFRARED SENSOR***

Disusun dan diajukan oleh

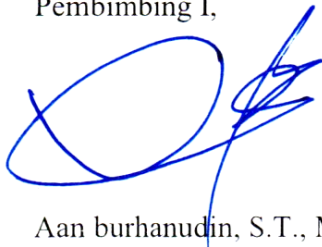
FAJAR AJIS SAPUTRA

NPM 16650076

Telah disetujui oleh pembimbing untuk dilanjutkan di  
hadapan Dewan Penguji

Semarang, 15 Mei 2022

Pembimbing I,



Aan burhanudin, S.T., M.T

NIP/NPP 148301458

Pembimbing II,



Agus Mukhtar, S. Pd., M. T.

NIP/NPP 148101429

**SKRIPSI**

**PERANCANGAN ALGORITMA PEMILAH SAMPAH OTOMATIS  
MENGUNAKAN *INDUCTIVE PROXIMITY SENSOR*,  
*CAPACITIVE PROXIMITY SENSOR*, DAN *INFRARED SENSOR***

**Disusun dan diajukan oleh**

**Fajar Ajis Saputra**

**NPM 16650076**

**telah dipertahankan di depan Dewan Penguji**

**pada tanggal 09 Agustus 2022**


**dan dinyatakan telah memenuhi syarat**

**Dewan Penguji**



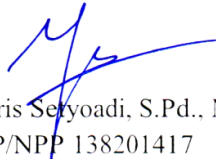
**Dr. Slamet Supriadi, M.Env., S.T.**  
NIP/NPP 195912281986031003

Sekretaris,



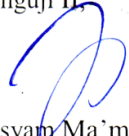
**Aan Burhanudin, S.T., M.T.**  
NIP/NPP 148301458

Penguji I,



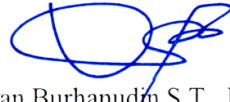
**Yuris Setyoadi, S.Pd., M.T.**  
NIP/NPP 138201417

Penguji II,



**Hisyam Ma'mun, S.T., M.T.**  
NIP/NPP 148101430

Penguji III,



**Aan Burhanudin S.T., M.T.**  
NIP/NPP 148301458

## **MOTO DAN PERSEMBAHAN**

### **Motto:**

1. Jangan pernah menunda sampai besok apa yang bisa anda lakukan hari ini. (Thomas Jefferson)
2. Salah satu kunci penting kesuksesan adalah percaya diri. Kunci penting percaya diri adalah persiapan. (Arthur Ashe)
3. Jangan takut gagal. Jangan sia-siakan energi untuk menutupi kegagalan. Pelajari kegagalan dan lanjut ke tantangan berikutnya. Tidak apa-apa gagal. Jika anda tidak gagal. Anda tidak tumbuh. (H. Stanley Judd)
4. Saat kamu sedang bermalas-malasan, ribuan pesaingmu sedang sibuk belajar mati-matian.

### **Persembahan:**

Skripsi ini saya persembahkan untuk:

1. Bapak dan Ibu tercinta.
2. Teman-teman satu angkatan.
3. Almamaterku Universitas PGRI Semarang.

## **PERYATAAN KEASLIAN TULISAN**

Saya yang bertanda tangan dibawah ini:

Nama : Fajar Ajis Saputra

NPM : 16650076

Progdi : Teknik Mesin

Fakultas : Teknik dan Informatika

Menyatakan dengan sebenarnya bahwa skripsi yang saya buat ini benar-benar merupakan hasil karya saya sendiri, bukan plagiarisme.

Apabila pada kemudian hari skripsi ini terbukti hasil plagiarisme, saya bersedia menerima sanksi atas perbuatan tersebut.

Semarang, 09 Agustus 2022

Yang membuat pernyataan

Fajar Ajis Saputra

NPM 16650076

## ABSTRAK

Kurangnya kesadaran dan kepedulian masyarakat terhadap kebersihan lingkungan menyebabkan permasalahan sampah masih menjadi persoalan serius. Untuk mengatasi permasalahan sampah, selain tanggung jawab pemerintah untuk menyediakan tempat pengolahan sampah. Masyarakat juga harus bisa membantu dengan mengelompokkan sampah sesuai dengan jenis sampah tersebut, agar mudah untuk didaur ulang ataupun untuk dijadikan kompos. Penelitian ini adalah membuat algoritma yang dapat digunakan pada alat pemilah sampah yang menggunakan *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor*. Serta cara mengkonfigurasi sensor agar dapat digunakan sesuai dengan harapan yang di inginkan.

Hasil dari penelitian ini *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor* dapat digunakan untuk memilah beberapa jenis sampah dengan mengkonfigurasi nilai hasil pembacaan setiap sensor. Jika *inductive proximity sensor* bernilai 0, *capacitive proximity sensor* bernilai 0, dan *infrared sensor* bernilai 0 berarti sensor mendeteksi sampah berjenis logam. Selanjutnya, Jika *inductive proximity sensor* bernilai 1, *capacitive proximity sensor* bernilai 1, dan *infrared sensor* bernilai 0 berarti sensor mendeteksi sampah kertas, kain atau botol plastik. Kemudian, apabila *inductive proximity sensor* bernilai 1, *capacitive proximity sensor* bernilai 0, dan *infrared sensor* bernilai 0 berarti sensor mendeteksi sampah kayu, kemasan *snack* atau sampah basah. Algoritma pemilah sampah telah berhasil dibuat dengan menggunakan bahasa C selanjutnya diimplementasikan pada *prototype* alat pemilah sampah yang telah dibuat. Algoritma yang telah dibuat dibagi menjadi beberapa kondisi dari kondisi 0 (nol) sampai kondisi 6 (enam) dimana setiap kondisi memiliki fungsi kerja masing-masing.

**Kata kunci:** Sampah, *Proximity Sensor*, Algoritma, Bahasa C

## **PRAKATA**

Assalamu Alaikum Warahmatullahi Wabarakatuh.

Syukur Alhamdulillah penulis panjatkan kehadiran Allah SWT atas rahmat dan hidayah-Nya sehingga skripsi ini dapat terselesaikan, Shalawat dan salam penulis kirimkan kepada Nabi Muhammad SAW, beserta para sahabat dan keluarga beliau yang telah memberikan tauladan menjalani kehidupan didunia dan diakhirat.

Selama proses penyusunan dan penyelesaian proposal skripsi ini tidak terlepas dari hambatan dan kesulian-kesulitan. Namun berkat bimbingan, bantuan, nasehat, dan dorongan serta saran-saran dari berbagai pihak, khususnya pembimbing, segala halangan yang menghambat serta kesulitan tersebut dapat diselesaikan dengan baik. Oleh karena itu penulis mengucapkan banyak terimakasih setulus hati kepada :

1. Dr. Sri Suciati, M. Hum. Rektor Universitas PGRI Semarang yang telah memberi kesempatan kepada penulis untuk menimba ilmu di Universitas PGRI Semarang.
2. Dr. Slamet Supriyadi, M. Env., S.T. Dekan Fakultas Teknik dan Informatika yang telah memberikan izin penulis untuk melakukan penelitian.
3. Aan Burhanudin, S.T., M.T. Selaku Pembimbing I dan Ketua Program Studi Teknik Mesin Universitas PGRI Semarang yang telah menyetujui topik skripsi penulis
4. Agus Mukhtar, S.Pd., M.T. Selaku pembimbing II yang telah membimbing penulis dengan penuh dedikasi yang tinggi
5. Bapak dan Ibu Dosen Program Studi Teknik Mesin yang telah memberi bekal ilmu kepada penulis selama belajar di Universitas PGRI Semarang.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna oleh karena itu, penulis mengharapkan segala bentuk saran serta masukan bahkan kritik yang membangun dari berbagai pihak.



Semarang, Juli 2022

Penulis,

(Fajar Ajis Saputra)

## DAFTAR ISI

<b>HALAMAN SAMBUTAN</b> .....	i
<b>HALAMAN PERSETUJUAN</b> .....	ii
<b>HALAMAN PENGESAHAN</b> .....	iii
<b>MOTO DAN PERSEMBAHAN</b> .....	iv
<b>PERYATAAN KEASLIAN TULISAN</b> .....	v
<b>ABSTRAK</b> .....	vi
<b>PRAKATA</b> .....	vii
<b>DAFTAR ISI</b> .....	viii
<b>DAFTAR TABEL</b> .....	xi
<b>DAFTAR GAMBAR</b> .....	xii
<b>DAFTAR LAMPIRAN</b> .....	xiii
<b>BAB I PENDAHULUAN</b> .....	1
A. Latar Belakang .....	1
B. Identifikasi Masalah .....	2
C. Pembatasan Masalah .....	2
D. Perumusan Masalah .....	3
E. Tujuan Penelitian .....	3
F. Manfaat Penelitian .....	3
1. Manfaat Teoritis .....	3
2. Manfaat Praktis.....	4
<b>BAB II KAJIAN PUSTAKA</b> .....	5
A. Tinjauan Pustaka .....	5
B. Landasan Teori.....	6

1. Sampah .....	6
2. ESP32 <i>DevKit</i> .....	6
3. LCD ( <i>Liquid Crystal Display</i> ).....	8
4. I2C/TWI LCD 1602 .....	10
5. Arduino IDE .....	10
6. <i>Relay</i> .....	11
7. Sensor Ultrasonik .....	14
8. Sensor <i>Proximity</i> .....	15
9. Konsep Dasar Algoritma .....	18
10. Himpunan .....	22
11. Konsep Dasar Probabilitas.....	24
C. Kerangka Berpikir.....	26
<b>BAB III METODE PENELITIAN</b> .....	27
A. Jenis Penelitian.....	27
B. Lokasi/Fokus Penelitian.....	27
C. Variabel Penelitian.....	27
D. Prosedur Penelitian .....	28
1. Diagram Alir Penelitian.....	28
2. Tahap Studi Pendahuluan.....	28
a. Analisis Permasalahan dan Potensi.....	29
b. Perancangan Algoritma Alat Pemilah Sampah.....	29
3. Tahap Studi Pengembangan .....	29
a. Uji Coba Produk .....	29
b. Revisi Produk.....	31
E. Teknik Analisis Data.....	31

F. Jadwal.....	31
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>32</b>
A. Data Hasil Pengujian Sensor.....	32
B. Himpunan Probabilitas Sensor.....	34
C. Perancangan Algoritma.....	35
D. Implementasi.....	36
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>41</b>
A. Kesimpulan .....	42
B. Saran.....	42
<b>DAFTAR PUSTAKA .....</b>	<b>42</b>

## **DAFTAR TABEL**

Tabel 2. 1 Perbandingan ESP8266 dengan ESP32 .....	7
Tabel 2. 2 Spesifikasi LCD 16x2 .....	9
Tabel 3. 1 Pengujian Nilai Sensor .....	30
Tabel 3. 2 Jadwal Penelitian.....	32
Tabel 4. 1 Data Pengujian Nilai Sensor .....	32
Tabel 4. 1 Pengelompokan jenis sampah sesuai nilai sensor .....	35

## DAFTAR GAMBAR

Gambar 2. 1 ESP32 DevKit .....	7
Gambar 2. 2 LCD 16x2 .....	9
Gambar 2. 3 Modul I2C LCD 16x2 .....	10
Gambar 2. 4 Tampilan Arduino IDE.....	11
Gambar 2. 5 Relay.....	12
Gambar 2. 6 Struktur Sederhana Relay .....	13
Gambar 2. 7 Sensor Ultrasonik SR04 .....	14
Gambar 2. 8 <i>Infrared Proximity Sensor</i> .....	17
Gambar 3. 1 Diagram Alir Penelitian .....	28
Gambar 4. 1 Pengujian Sampah Kaleng .....	32
Gambar 4. 2 Pengujian Sampah Kemasan <i>Snack</i> .....	33
Gambar 4. 3 Rancangan Algoritma Pemilah Sampah.....	36
Gambar 4. 4 Program Kondisi 0 .....	37
Gambar 4. 5 Program Kondisi 1 .....	38
Gambar 4. 6 Program Kondisi 2 .....	39
Gambar 4. 7 Program Kondisi 5 .....	40
Gambar 4. 8 Program Kondisi 6 .....	41

## DAFTAR LAMPIRAN

Lampiran 1 Gambar Uji Coba Algoritma Deteksi Sampah .....	44
Lampiran 2 Gambar Penampung Sampah Sementara .....	46
Lampiran 3 Program Algoritma dalam Bahasa C ( <i>coding</i> ) .....	46

# **BAB I**

## **PENDAHULUAN**

### **A. Latar Belakang**

Sampah menjadi masalah yang tidak bisa dihindari dan terus berkembang. Sampah adalah material sisa yang dibuang sebagai hasil dari proses produksi, baik itu industri maupun rumah tangga yang merupakan sesuatu yang tidak diinginkan oleh manusia setelah proses atau penggunaannya berakhir (Akbar, Anjasmara, & Wardhani, 2021). Tata kelola sampah yang kurang baik menyebabkan sampah menumpuk dan mengeluarkan bau busuk serta dapat menjadi sumber penularan penyakit. Sampah juga bisa mengakibatkan penyumbatan pada saluran drainase dan sungai. Masih kurangnya kesadaran dan kepedulian masyarakat terhadap kebersihan lingkungan menyebabkan permasalahan sampah masih menjadi persoalan serius bagi pemerintah.

Untuk mengatasi permasalahan sampah, selain tanggung jawab pemerintah untuk menyediakan tempat pengolahan sampah. Masyarakat juga harus bisa membantu dengan mengelompokkan sampah sesuai dengan jenis sampah tersebut, agar mudah untuk didaur ulang ataupun untuk dijadikan kompos. Namun, kenyataanya di tempat pembuangan sampah berbagai jenis sampah bercampur menjadi satu baik sampah logam maupun nonlogam (Widodo & Suleman, 2020). Hal ini dapat dilihat saat pengangkutan sampah yang masih bercampur antara sampah organik dan anorganik. Banyak sampah organik yang masuk ke dalam tempat sampah anorganik, begitu juga sebaliknya. Salah satu penyebabnya adalah kurangnya pengetahuan masyarakat untuk membedakan kedua jenis sampah ini.

Teknologi yang berkembang sekarang ini yang memanfaatkan piranti-piranti digital sehingga dapat membantu dalam mengerjakan hal-hal yang rumit sekaligus. Pemanfaatan teknologi ini misalnya dalam mengatur buka tutup tempat sampah, pengenalan terhadap objek yang menggunakan sensor dan dikontrol melalui mikrokontroler. Dari permasalahan membuang sampah

dan potensi teknologi yang ada saat ini, penulis ingin membuat alat yang dapat membantu masyarakat dalam memilah jenis sampah.

Tujuan dari penelitian ini adalah membuat algoritma yang dapat digunakan pada alat pemilah sampah yang menggunakan *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor*. Serta cara mengkonfigurasi sensor agar dapat digunakan sesuai dengan harapan yang diinginkan.

#### B. Identifikasi Masalah

Berdasarkan uraian yang ada pada latar belakang, maka dapat diidentifikasi masalah-masalah sebagai berikut:

1. Saat ini masih banyak orang yang belum menyadari pentingnya memilah dan membuang sampah sesuai dengan jenis sampah tersebut agar mudah dalam pengolahan sampah lebih lanjut.
2. Tempat sampah yang umum digunakan saat ini hanya berfungsi sebagai penampungan tanpa ada monitoring volume sampah dan tidak bisa memilah jenis sampah.
3. Bisakah *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor* digunakan untuk memilah sampah.
4. Bagaimana pengaturan sensor paling optimal untuk dapat memilah sampah.

#### C. Pembatasan Masalah

Pemilahan sampah secara otomatis dapat membantu mempermudah dalam mengelola sampah untuk di daur ulang maupun dijadikan kompos, namun dalam penerapannya ada beberapa kendala yang dihadapi diantaranya:

1. Penelitian ini hanya sebatas pembuatan algoritma alat pemilah sampah yang menggunakan *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor*.
2. Pengelompokan jenis sampah hanya terbatas pada material sampah logam, non logam, plastik dan material-material yang dapat dideteksi sensor.



3. Penelitian ini hanya terbatas pada pembuatan algoritma, desain dan konstruksi tempat sampah tidak termasuk dalam kajian yang dibahas oleh peneliti.

#### D. Perumusan Masalah

Berdasarkan uraian yang telah dikemukakan di latar belakang dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana cara konfigurasi *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor* agar dapat mendeteksi berbagai jenis sampah.
2. Bagaimana rancangan algoritma pemilah sampah yang menggunakan *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor*.

#### E. Tujuan Penelitian

Tujuan dari penelitian kali ini, peneliti memiliki beberapa tujuan yang ingin dicapai diantaranya:

1. Mengetahui cara konfigurasi *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor* agar dapat digunakan untuk memilah sampah.
2. Terciptanya Algoritma yang dapat diaplikasikan pada alat pemilah sampah yang menggunakan *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor*.

#### F. Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat baik kepada akademisi maupun masyarakat secara umum. Adapun manfaat yang dapat diambil dari penelitian ini diantaranya adalah:

1. Manfaat Teoritis
  - a. Penelitian ini dapat menambah wawasan serta pengetahuan bagi akademisi ataupun masyarakat secara umum.
  - b. Dapat digunakan sebagai acuan dan bahan pertimbangan untuk penelitian-penelitian selanjutnya, khususnya bagi penelitian yang berkaitan tentang mekatronika.

## 2. Manfaat Praktis

- a. Bagi penulis, dengan melakukan penelitian ini diharapkan penulis dapat menerapkan ilmu pengetahuan yang dipelajari dan dapat ikut serta berkontribusi bagi kemajuan ilmu pengetahuan.
- b. Bagi industri, *prototype* yang dihasilkan dalam penelitian ini diharapkan dapat menjadi salah satu alternatif alat pengatur barang yang lebih praktis dan efisien.

## BAB II

### KAJIAN PUSTAKA

#### A. Tinjauan Pustaka

Beberapa peneliti telah meneliti tentang tempat sampah pintar ataupun alat pemilah sampah menggunakan *inductive proximity sensor* dan *capacitive proximity sensor*. Ismail dkk pada Januari 2021 telah membuat “Tempat Sampah Pintar Berbasis *Internet of Things (IoT)* dengan Sistem Teknologi Informasi”. Mereka membuat sistem tempat sampah pintar menggunakan *raspberry pi* dan *internet of things (iot)* dengan sistem teknologi informasi, yang mengirim informasi data ke *web server* dan diolah di aplikasi *thingspeak* yang dapat di privat atau bisa di publik, sehingga dapat mengidentifikasi persoalan sampah yang berserakan. Penyatuan sistem dengan teknologi informasi, dapat memungkinkan tempat sampah bisa di monitoring dari jarak jauh. Dengan adanya otomatis motor penggerak pintu tong sampah, dapat memudahkan masyarakat membuang sampah (Ismail, Abdullah, & Abdussamad, 2021).

Pada Maret 2020 Wafi dkk telah mempublikasikan hasil penelitian mereka yang berjudul “Prototipe Sistem *Smart Trash* Berbasis IOT (*Internet of Things*) dengan Aplikasi Android”. Alat yang mereka buat menggunakan 2 sensor untuk memilah sampah, yaitu *inductive proximity sensor* dan *capacitive proximity sensor*. Mereka mengelompokkan sampah menjadi sampah organik dan anorganik. Hasil dari pembuatan alat mereka didapatkan keberhasilan deteksi sampah organik 95% dan deteksi sampah anorganik 97.5% (Wafi, Setyawan, & Ariyani, 2020).

Sebelum itu pada tahun 2019 Dewi dkk juga telah mempublikasikan hasil penelitian mereka yang berjudul “Perancangan dan Implementasi *Smart Trash Bin* Menggunakan Metode Logika *Fuzzy*”. Pada penelitian ini Dewi dkk menggunakan sensor *proximity* induktif dan sensor warna TCS230 sebagai sensor yang berfungsi untuk membedakan jenis sampah. Jenis sampah yang mereka gunakan terbatas pada sampah kaleng, sampah botol plastik dan sampah kertas.

*Smart Trash Bin* yang mereka buat mampu membedakan sampah kaleng, kertas dan plastik secara otomatis. *Smart Trash Bin* memiliki nilai akurasi, presisi dan sensitifitas yang cukup baik. Dengan keseluruhan nilai akurasi sebesar 84.4%, presisi sebesar 77.7% dan sensitifitas sebesar 81.6%. Sampah plastik dan kertas terkadang terdeteksi memiliki nilai yang tertukar, hal ini dikarenakan nilai RGB untuk sampah botol plastik mendekati nilai RGB untuk sampah kertas.

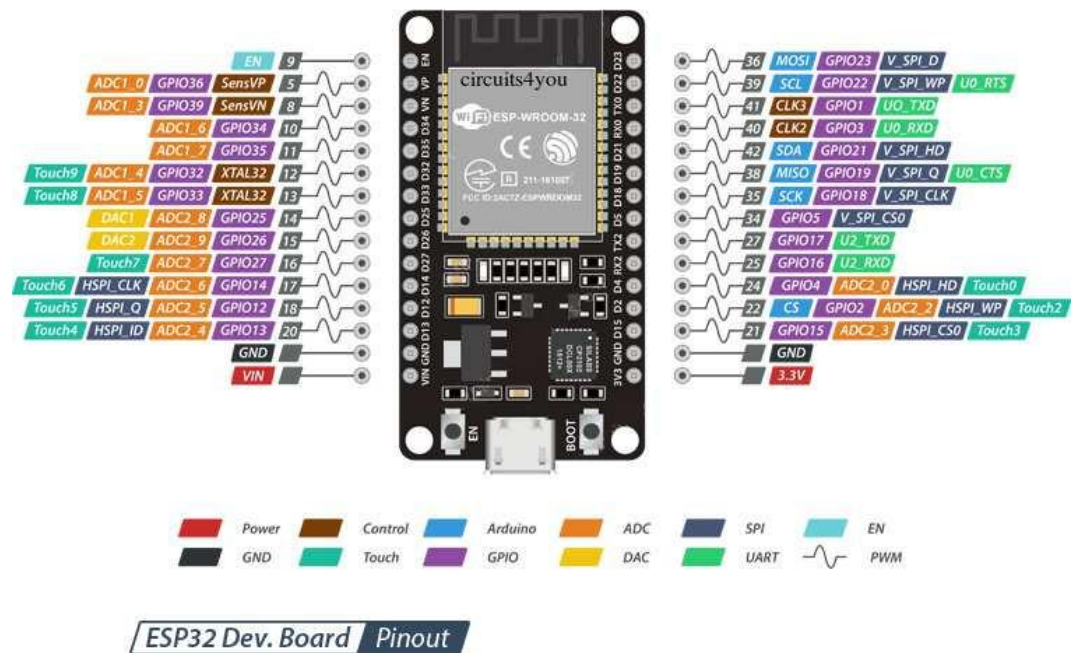
## B. Landasan Teori

### 1. Sampah

Sampah adalah penamaan dari bahan sisa manusia yang tidak diinginkan lagi sesudah berakhirnya suatu proses (Puadi & Hambali, 2022). Sampah juga merupakan sesuatu yang sudah tidak terpakai dan tidak diinginkan lagi yang pada akhirnya dibuang pada tempat sampah. Sampah dianggap sebagai sesuatu hal yang sangat tidak bernilai, namun bagi orang yang paham, sampah dapat diolah kembali dan menghasilkan sesuatu yang sangat bermanfaat. Sampah dikelompokkan menjadi dua bagian yaitu sampah padat dan sampah cair. Sampah padat terbentuk akibat sisa asal manusia berupa zat padat dan sebaliknya.

### 2. ESP32 *DevKit*

ESP32 *DevKit* merupakan salah satu mikrokontroler keluaran *espressif* dan merupakan penerus dari ESP8266. ESP32 ini memiliki keunggulan yang tidak dimiliki oleh arduino, diantaranya yaitu memiliki fitur *Wi-Fi* dan *bluetooth* 4.2 yang sudah tertanam di dalam board itu sendiri. Kemudian ESP32 ini memiliki kecepatan prosesor yang cukup cepat yang sudah *dual-core* 32-bit dengan kecepatan 160/240MHz.



Gambar 2. 1 ESP32 DevKit

(Sumber: *circuits4you.com*, 2018)

ESP32 DevKit sendiri telah banyak digunakan untuk pemrograman berbasis *IoT* karena memiliki konektivitas yang sudah ada di dalam *board* ESP32 tersebut sehingga tidak perlu modul tambahan lagi untuk penggunaan *Wi-Fi* ataupun *Bluetooth*. Selain itu terlihat pada Gambar 2.1 ESP32 memiliki GPIO sebanyak 36 pin, GPIO sendiri merupakan *General Purpose Input Output* yang berfungsi sebagai pin *input* dan *output* analog maupun digital. Berikut pada Tabel 2.1 terlihat perbandingan ESP8266 dan ESP32 secara fitur dan spesifikasi lengkap.

Tabel 2. 1 Perbandingan ESP8266 dengan ESP32

Spesifikasi	Board	
	ESP8266	ESP32
MCU	Xtensa Single-core 32-bit L106	Xtensa Dual-Core 32-bit LX6 with 600DMIPS
Wi-Fi	802.11 b/g/n tipe HT20	802.11 b/g/n tipe HT40

<i>Bluetooth</i>	Tidak Ada	<i>Bluetooth 4.2 dan BLE</i>
Frekuensi	80 MHz	160 MHz
SRAM	Tidak Ada	Ada
Total GPIO	17 pin	36 pin
Total ADC pin	1 pin	15 pin
Total Digital pin	9 pin	2 pin
Tegangan <i>Output</i>	3.3 – 5 Volt	3.3 – 5 Volt
Total SPI-UART-I2C-I2S	2-2-1-2	4-2-2-2
Resolusi ADC	10 bit	12 bit
Suhu operasional kerja	-40°C hingga 125°C	-40°C hingga 125°C
Sensor dalam modul	Tidak ada	<i>Touch Sensor, Temperature Sensor, Hall Effect Sensor</i>
Harga di pasaran	Rp. 30.000 – 350.000	Rp. 70.000 – 650.000

Seperti yang terlihat pada Tabel 2.1, sudah sangat jelas ESP32 lebih unggul dan memiliki *processor* yang lebih tinggi sehingga pengolahan data akan lebih cepat. Selain itu pin ADC yang terdapat pada ESP32 lebih banyak dibandingkan dengan ESP8266. Sehingga dapat melakukan pemrograman yang lebih kompleks.

### 3. LCD (*Liquid Crystal Display*)

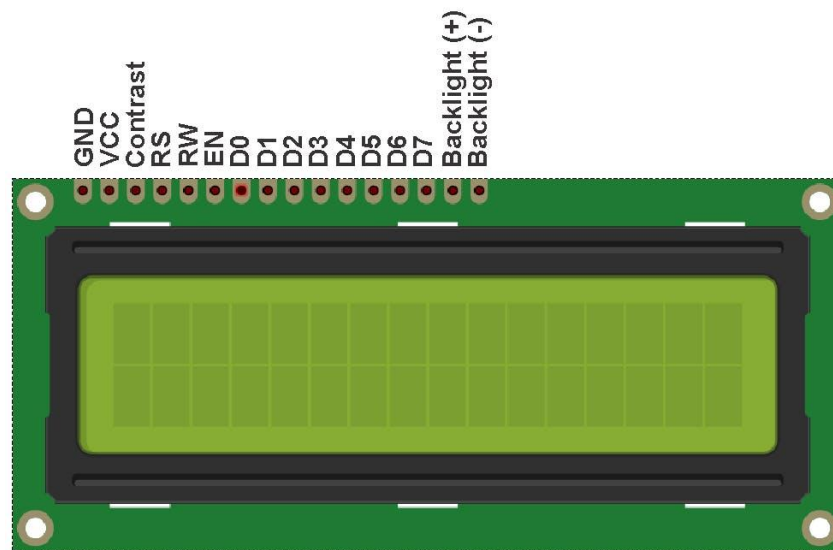
*Liquid Crystal Display* merupakan media yang digunakan untuk menampilkan hasil dari keluaran pada sebuah rangkaian elektronika. Fitur yang terdapat dalam LCD ini adalah

- a. 16 karakter dan 2 baris atau biasa disebut LCD 16x2
- b. Memiliki 192 karakter.
- c. Memiliki karakter generator yang terprogram.
- d. Dapat digunakan melalui mode 4-bit dan 8-bit.
- e. Dapat digunakan secara *back light*.

Definisi pin lcd 16x2 dapat dilihat ditabel 2. 2 dan gambar 2. 2 adalah *device* LCD.

Tabel 2. 2 Spesifikasi LCD 16x2

Pin	Deskripsi
1	<i>Ground ( - )</i>
2	VCC ( + )
3	Mengatur kontras atau pencahayaan
4	<i>Register Select</i>
5	<i>Read/Write LCD Register</i>
6	<i>Enable</i>
7-14	Data I/O ( <i>input/output</i> )
15	VCC ( + ) LED
16	<i>Ground ( - ) LED</i>



Gambar 2. 2 LCD 16x2

(Sumber: Joseph, 2020)

#### 4. I2C/TWI LCD 1602

Merupakan modul yang dipakai untuk mengurangi penggunaan kaki di LCD 1602. Modul ini memiliki 4 Pin yang akan dihubungkan ke Arduino Wilmshurst, 2009 dalam (Saputra dkk. 2020).

- a. GND: dihubungkan ke GND Arduino
- b. VCC: dihubungkan ke 5V Arduino
- c. SDA: Merupakan I2C data dan dihubungkan ke pin analog pada Arduino
- d. SCL: Merupakan I2C *clock* dan dihubungkan ke pin analog pada arduino.



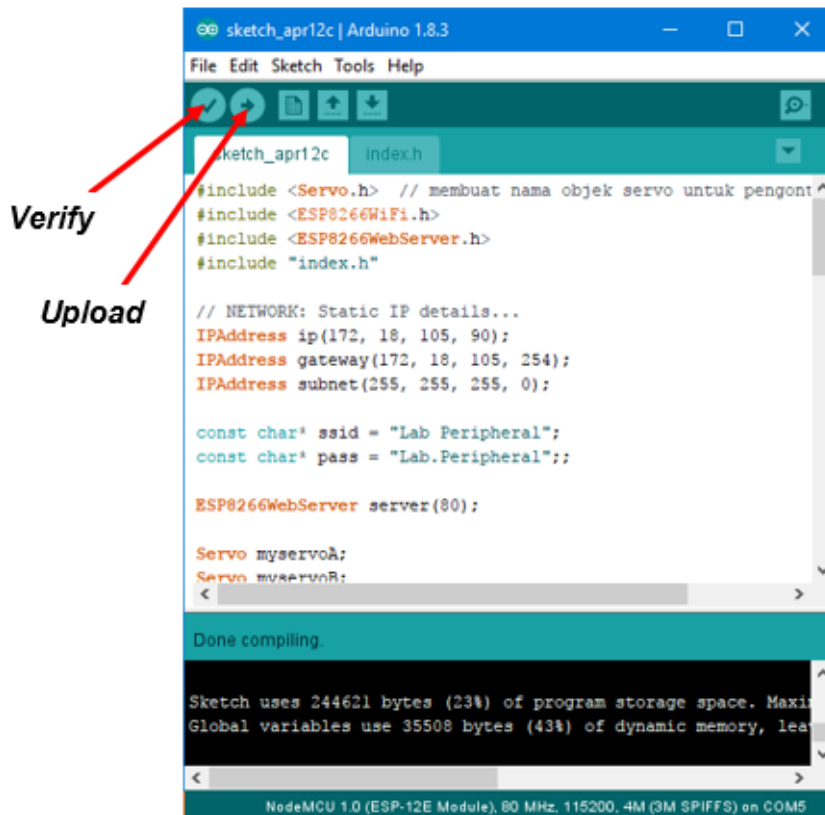
Gambar 2. 3 Modul I2C LCD 16x2

(Sumber: Saputra dkk. 2020)

#### 5. Arduino IDE

*Software* yang digunakan dalam pemrograman nodeMCU esp32 pada penelitian ini adalah Arduino IDE. Arduino IDE (*Integrated Development Environment*) merupakan sebuah perangkat lunak yang berfungsi sebagai wadah untuk melakukan pemrograman dalam bentuk bahasa C. Arduino IDE ini dibuat dari bahasa pemrograman Java yang sudah dilengkapi dengan *library C/C++* yang biasa disebut *wiring* yang membuat operasi *input* atau *output* menjadi lebih mudah. Arduino IDE juga dapat disebut sebagai *software* untuk mendesain sebuah fungsi-fungsi yang akan dituangkan kedalam perangkat keras. Arduino IDE sendiri sangat populer dan banyak digunakan oleh pengembang untuk melakukan perancangan sederhana hingga kompleks sekalipun. Berikut adalah tampilan Arduino IDE yang dapat dilihat pada Gambar 2.4.





Gambar 2. 4 Tampilan Arduino IDE

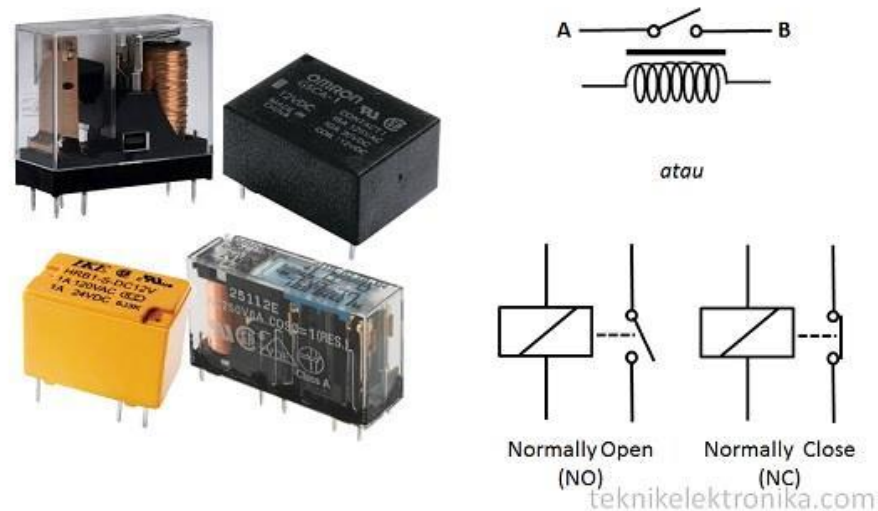
(Sumber: Endra dkk, 2019)

Pada Gambar 2.4 terlihat tampilan Arduino IDE yang isinya terdapat 2 buah fungsi yang sudah tersedia yaitu, *void setup* dan *void loop*. *Void setup* merupakan sebuah fungsi untuk meng-inisialisasi node-node yang akan digunakan. Sedangkan *void loop* merupakan sebuah paragraf untuk mengatur program agar melakukan aksi terhadap *node-node* yang digunakan atau memberi perintah untuk melakukan fungsi tertentu.

## 6. Relay

*Relay* adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen *electromechanical* (elektromekanikal) yang terdiri dari 2 bagian utama yakni elektromagnet (*coil*) dan mekanikal (Saklar/*Switch*). *Relay* menggunakan prinsip elektromagnetik untuk menggerakkan Kontak Saklar sehingga dengan arus listrik yang kecil (*low*

*power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Sebagai contoh, dengan *relay* yang menggunakan elektromagnet 5V dan 50 mA mampu menggerakkan *Armature Relay* (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A.



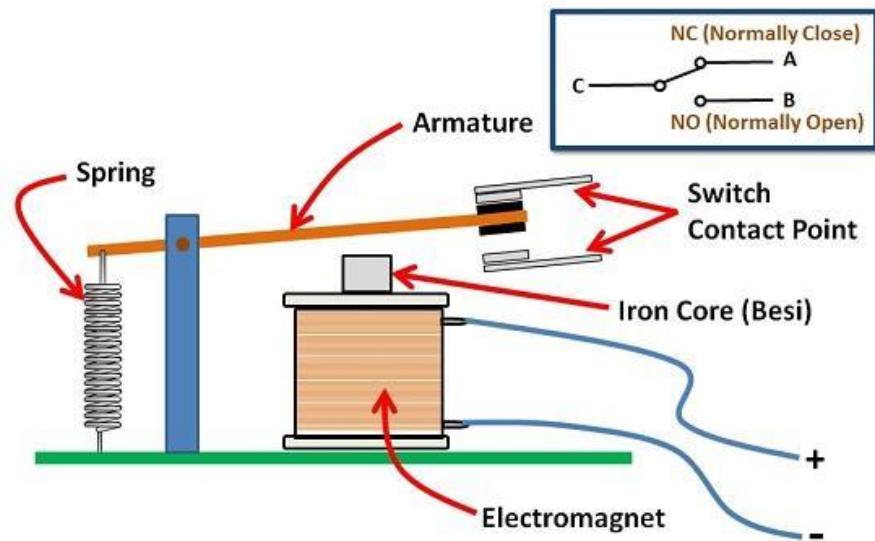
Gambar 2. 5 Relay

(Sumber: Kho, 2020)

Pada dasarnya, *relay* terdiri dari 4 komponen dasar yaitu:

- a) *Electromagnet* (Coil)
- b) *Armature*
- c) *Switch Contact Point* (Saklar)
- d) *Spring*

Gambar 2.6 merupakan gambar dari bagian-bagian relay: Struktur dasar Relay



Gambar 2. 6 Struktur Sederhana *Relay*

(Sumber: Kho, 2020)

Kontak Poin (*Contact Point*) *Relay* terdiri dari 2 jenis yaitu:

- a) *Normally Close (NC)* yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *CLOSE* (tertutup)
- b) *Normally Open (NO)* yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi *OPEN* (terbuka)

Berdasarkan gambar diatas, sebuah Besi (*Iron Core*) yang dililit oleh sebuah kumparan *coil* yang berfungsi untuk mengendalikan besi tersebut. Apabila kumparan *coil* diberikan arus listrik, maka akan timbul gaya elektromagnet yang kemudian menarik *armature* untuk berpindah dari posisi sebelumnya (NC) ke posisi baru (NO) sehingga menjadi saklar yang dapat menghantarkan arus listrik di posisi barunya (NO). Posisi dimana *armature* tersebut berada sebelumnya (NC) akan menjadi *OPEN* atau tidak terhubung. Pada saat tidak dialiri arus listrik, *armature* akan kembali lagi ke posisi awal (NC). *Coil* yang digunakan oleh *relay* untuk menarik *contact poin* ke posisi *Close* pada umumnya hanya membutuhkan arus listrik yang relatif kecil.

## 7. Sensor Ultrasonik

Sensor ultrasonik adalah sensor yang mendeteksi objek yang berada disekitar sensor (Suherman, Mardeni, Irawan, & Sugiati, 2020). Sensor Ultrasonik berfungsi mengubah besaran fisik (bunyi) menjadi besaran listrik atau sebaliknya. Sensor ultrasonik terdiri dari dua bagian, yaitu rangkaian pemancar gelombang ultrasonik (*transmitter*) dan rangkaian penerima gelombang ultrasonik (*Receiver*). Sensor ultrasonik mentransmisi gelombang ultrasonik dengan kecepatan diatas jangkauan pendengaran manusia dan mengeluarkan pulsa yang sesuai dengan waktu yang dibutuhkan gelombang untuk kembali ke sensor. Sensor ultrasonik SR-04 terdiri dari sebuah pemancar gelombang ultrasonik dan sebuah rangkaian penerima gelombang ultrasonik.



Gambar 2. 7 Sensor Ultrasonik SR04

(Sumber: Saputra dkk. 2020)

Pemancar gelombang ultrasonik mengubah gelombang ultrasonik dengan frekuensi 40 KHz menjadi suara, sedangkan rangkaian penerima gelombang ultrasonik berfungsi untuk mendeteksi pantulan gelombang ultrasonik (Harmaji & Khairullah, 2019). Sensor ini mendeteksi jarak objek dengan cara memancarkan gelombang ultrasonik dan mendeteksi

pantulannya (*echo*), kemudian menghitung jarak benda dengan perhitungan waktu tempuh gelombang antara pemancar hingga sampai ke penerima.

#### 8. Sensor *Proximity*

*Proximity Sensor* (Sensor Proksimitas) atau dalam bahasa Indonesia disebut dengan sensor jarak adalah sensor elektronik yang mampu mendeteksi keberadaan objek di sekitarnya tanpa adanya sentuhan fisik. Dapat juga dikatakan bahwa sensor *proximity* adalah perangkat yang dapat mengubah informasi tentang gerakan atau keberadaan objek menjadi sinyal listrik.

Sensor *proximity* dapat diklasifikasikan menjadi 4 jenis, yaitu *inductive proximity sensor*, *capacitive proximity sensor*, *ultrasonic proximity sensor* dan *photoelectric sensor*.

##### a) *Inductive Proximity Sensor* (Sensor Jarak Induktif)

Sensor Jarak Induktif atau *Inductive Proximity Sensor* adalah sensor jarak yang digunakan untuk sensor jarak yang digunakan untuk mendeteksi keberadaan logam baik logam jenis *ferrous* maupun logam jenis *non-ferrous*. Sensor ini dapat digunakan untuk mendeteksi keberadaan (ada atau tidak adanya objek logam), menghitung objek logam dan aplikasi pemosisian. Sensor induktif sering digunakan sebagai pengganti saklar mekanis karena kemampuannya yang dapat beroperasi pada kecepatan yang lebih tinggi dari sakelar mekanis biasa. Sensor jarak induktif ini juga lebih andal dan lebih kuat.

Sensor *Proximity* Induktif pada umumnya terbuat dari kumparan/koil dengan inti ferit sehingga dapat menghasilkan medan elektromagnetik frekuensi tinggi. *Output* dari sensor jarak jenis induktif ini dapat berupa analog maupun digital. Versi analog dapat berupa tegangan (biasanya sekitar 0–10VDC) atau arus (4–20mA). Jarak pengukurannya bisa mencapai hingga 2 inci. Sedangkan versi digital biasanya digunakan pada rangkaian DC saja ataupun rangkaian AC/DC. Sebagian besar sensor induktif digital dikonfigurasi dengan *output* “*NORMALLY-OPEN*” namun ada juga yang dikonfigurasi dengan *output*

“*NORMALLY-CLOSE*”. Sensor induktif ini sangat cocok untuk mendeteksi benda-benda logam di mesin dan di peralatan otomatisasi.

*Inductive proximity sensor* ini pada dasarnya terdiri dari sebuah osilator, sebuah koil dengan inti ferit, rangkaian detektor, rangkaian output, kabel dan konektor. Osilator pada sensor jarak ini akan membangkitkan gelombang sinus dengan frekuensi yang tetap. Sinyal ini digunakan untuk menggerakkan kumparan atau koil. Koil dengan inti ferit ini akan menginduksi medan elektromagnetik. Ketika garis-garis medan elektromagnetik ini ter-interupsi oleh objek logam, tegangan osilator akan berkurang sebanding dengan ukuran dan jarak objek dari kumparan/koil. Dengan demikian, sensor proksimitas ini dapat mendeteksi adanya objek yang sedang mendekatnya. Pengurangan tegangan osilator ini disebabkan oleh arus Eddy yang diinduksi pada logam yang meng-interupsi garis-garis logam.

b) *Capacitive Proximity Sensor* (Sensor Jarak Kapasitif)

Sensor Jarak Kapasitif atau *Capacitive Proximity Sensor* adalah sensor jarak yang dapat mendeteksi gerakan, komposisi kimia, tingkat dan komposisi cairan maupun tekanan. Sensor Jarak Kapasitif dapat mendeteksi bahan-bahan dielektrik rendah seperti plastik atau kaca dan bahan-bahan dielektrik yang lebih tinggi seperti cairan sehingga memungkinkan sensor jenis ini untuk mendeteksi tingkat banyak bahan melalui kaca, plastik maupun komposisi kontainer lainnya.

Sensor Jarak Kapasitif ini pada dasarnya mirip dengan Sensor Jarak Induktif, perbedaannya adalah sensor kapasitif menghasilkan medan elektrostatik sedangkan sensor induktif menghasilkan medan elektromagnetik. Sensor Jarak Kapasitif ini dapat digerakan oleh bahan konduktif dan bahan non-konduktif. Elemen aktif Sensor Jarak Kapasitif dibentuk oleh dua elektroda logam yang diposisikan untuk membentuk ekuivalen (sama dengan) dengan kapasitor terbuka. Elektroda ini ditempatkan di rangkaian osilasi yang berfrekuensi tinggi. Ketika objek mendekati permukaan sensor jarak kapasitif ini, medan elektrostatik pelat

logam akan terinterupsi sehingga mengubah kapasitansi sensor jarak. Perubahan ini akan mengubah kondisi dalam pengoperasian sensor jarak sehingga dapat mendeteksi keberadaan objek tersebut.

c) *Ultrasonic Proximity Sensor* (Sensor Jarak Ultrasonik)

Sensor Jarak Ultrasonik atau *Ultrasonic Proximity Sensor* adalah sensor jarak yang menggunakan prinsip operasi yang mirip dengan radar atau sonar yaitu dengan menghasilkan gelombang frekuensi tinggi untuk menganalisis gema yang diterima setelah terpantul dari objek yang mendekatinya. Sensor *Proximity* Ultrasonik ini akan menghitung waktu antara pengiriman sinyal dengan penerimaan sinyal untuk menentukan jarak objek yang bersangkutan. Sering digunakan untuk mendeteksi keberadaan objek dan mengukur jarak objek di proses otomasi pabrik.

d) *Photoelectric Proximity Sensor* (Sensor Jarak Fotolistrik)

Sensor Jarak Fotolistrik atau *Photoelectric Proximity Sensor* adalah sensor jarak yang menggunakan elemen peka cahaya untuk mendeteksi obyek. Sensor *Proximity* Fotolistrik terdiri sumber cahaya (atau disebut dengan *Emitor*) dan Penerima (*Receiver*).



Gambar 2. 8 *Infrared Proximity Sensor*  
(Sumber: Singgeta & Manembu, 2021)

Terdapat 3 jenis Sensor Jarak Fotolistrik, yaitu:

- 1) *Direct Reflection–Emitor* dan *Receiver* yang ditempatkan bersama, menggunakan cahaya yang dipantulkan langsung dari obyek untuk dideteksi.
- 2) *Refleksi* dengan Reflektor–*Emitor* dan *Receiver* yang disimpan bersama dan membutuhkan Reflektor, Sebuah Obyek dideteksi ketika obyek tersebut mengganggu berkas cahaya antara sensor dan reflektor.
- 3) *Thru Beam–Emitor* dan *Receiver* ditempatkan secara terpisah, mendeteksi suatu obyek ketika obyek tersebut mengganggu berkas cahaya antara pemancar dan penerima.

## 9. Konsep Dasar Algoritma

### a) Pengertian Algoritma

Algoritma berarti solusi. Ketika orang berbicara mengenai algoritma di bidang pemrograman, maka yang dimaksud adalah solusi dari suatu masalah yang harus dipecahkan dengan menggunakan komputer. Algoritma harus dibuat secara runtut agar komputer mengerti dan mampu mengeksekusinya. Analisis kasus sangat dibutuhkan dalam membuat sebuah algoritma, misalnya proses apa saja yang sekiranya dibutuhkan untuk menyelesaikan masalah yang harus diselesaikan.

Algoritma berasal dari kata *algoris* dan *ritmis* yang pertama kali diungkapkan oleh Abu Ja'far Mohammad Ibn Musa Al Khowarizmi (825M) dalam buku *Al-Jabr Wa-al Muqobla*. Dalam beberapa buku terdapat beberapa definisi algoritma. Tetapi kalau kita cermati dengan baik, buku-buku tersebut mempunyai definisi yang sama. Definisi Algoritma adalah susunan langkah penyelesaian suatu masalah secara sistematis dan logis (Sitorus, 2015). Terdapat dua kata yang menjadi perhatian dalam definisi ini, yaitu *sistematis* dan *logis*.

Membangun sebuah program pada dasarnya adalah membuat alat bantu untuk menyelesaikan suatu masalah. Sebelum kita dapat



menghasilkan program yang mampu membantu kita menyelesaikan masalah tersebut, kita dihadapkan pada 3 (tiga) tahapan pokok, yaitu:

- 1) Memahami permasalahan dan tujuan sebuah program dibuat. Pada tahap ini kita harus mampu mengidentifikasi jenis, bentuk dan karakteristik input serta output yang diharapkan. Tetapi untuk skala permasalahan yang besar, selain jenis, bentuk dan karakteristik, kita juga perlu mengetahui dengan pasti asal, frekuensi dan volume data output yang diharapkan.
- 2) Mampu menyusun konsep/rancangan/desain penyelesaian dari masalah yang akan kita selesaikan. Dari hasil pemahaman kita terhadap permasalahan di atas, kita harus mampu merancang sebuah alur proses untuk mengolah data input dan menghasilkan data output dengan jenis, bentuk dan karakteristik seperti yang diharapkan.
- 3) Mampu mengimplementasikan hasil rancangan kita dalam bentuk program yang terstruktur. Program tersebut dapat kita buat dengan menggunakan sembarang bahasa pemrograman. Untuk itulah kita diharuskan memahami dan menguasai komponen bahasa pemrograman dan teknik pemrograman dengan baik.

b) Kriteria Algoritma

Menurut Donald E. Knuth, algoritma yang baik memiliki kriteria sebagai berikut:

1) *Input*

Suatu algoritma harus memiliki 0 (nol) atau lebih masukan (*input*). Artinya, suatu algoritma itu dimungkinkan tidak memiliki masukan secara langsung dari pengguna, maka semua data dapat diinisialisasikan atau dibangkitkan dalam algoritma.

2) *Output*

Suatu algoritma harus memiliki satu atau lebih keluaran (*output*). Suatu algoritma yang tidak memiliki keluaran (*output*) adalah suatu algoritma yang sia-sia, yang tidak perlu dilakukan.

Algoritma dibuat untuk tujuan menghasilkan suatu yang diinginkan, yaitu berupa hasil keluaran.

3) *Finiteness*

Setiap pekerjaan yang dikerjakan pasti berhenti. Demikian juga algoritma harus dijamin akan berhenti setelah melakukan sejumlah langkah proses.

4) *Definiteness*

Tidak menimbulkan makna ganda (*ambiguous*). Setiap baris aksi/pernyataan dalam suatu algoritma harus pasti, artinya tidak menimbulkan penafsiran lain bagi setiap pembaca algoritma, sehingga memberikan output yang sesuai dengan yang diharapkan oleh pengguna.

5) *Effectiveness*

Langkah-langkah algoritma dikerjakan dalam waktu yang wajar. Suatu algoritma tidak terdapat suatu aksi yang tidak perlu dilakukan. Setiap aksi akan memerlukan waktu eksekusi. Apabila aksi yang dilakukan tidak diperlukan maka aksi tersebut hanya akan menambah waktu proses. Misalnya aksi  $X + 0$ , aksi ini jelas tidak ada pengaruh dan tidak berguna karena hasil dari  $X + 0$  adalah  $X$ . Jadi tidak perlu dilakukan karena sia-sia.

c) Penyajian Algoritma

Algoritma adalah independen terhadap bahasa pemrograman tertentu, artinya algoritma yang telah dibuat tidak boleh hanya dapat diterapkan pada bahasa pemrograman tertentu. Penulisan algoritma tidak terikat pada suatu aturan tertentu, tetapi harus jelas maksudnya untuk tiap langkah algoritmanya. Namun pada dasarnya algoritma dibagi menjadi beberapa macam berdasarkan format penulisannya, yaitu:

1) Deskriptif

Algoritma bertipe deskriptif maksudnya adalah algoritma yang ditulis dalam bahasa manusia sehari-hari (misalnya bahasa Indonesia atau bahasa Inggris) dan dalam bentuk kalimat. Setiap langkah algoritmanya diterangkan dalam satu atau beberapa kalimat.

Sebagai contoh misalnya algoritma menentukan bilangan terbesar dari 3 bilangan berikut ini:

Algoritma Menentukan\_bilangan\_terbesar\_dari\_3\_bilangan:

- a. Meminta input 3 bilangan dari *user*, misalkan bilangan a, b, dan c.
- b. Apabila bilangan a lebih besar dari b maupun c, maka bilangan a merupakan bilangan terbesar
- c. Jika tidak (bilangan a tidak lebih besar dari b atau c) berarti bilangan a sudah pasti bukan bilangan terbesar. Kemungkinanannya tinggal bilangan b atau c. Apabila bilangan b lebih besar dari c, maka b merupakan bilangan terbesar. Sebaliknya apabila bilangan b tidak lebih besar dari c, maka bilangan c merupakan yang terbesar.
- d. Selesai

## 2) *Pseudocode*

***Pseudo*** berarti imitasi dan ***code*** berarti kode yang dihubungkan dengan instruksi yang ditulis dalam bahasa komputer (kode bahasa pemrograman). Apabila diterjemahkan secara bebas, maka *pseudocode* berarti tiruan atau imitasi dari kode bahasa pemrograman. Pada dasarnya, *pseudocode* merupakan suatu bahasa yang memungkinkan programmer untuk berpikir terhadap permasalahan yang harus dipecahkan tanpa harus memikirkan *syntax* dari bahasa pemrograman yang tertentu. Tidak ada aturan penulisan *syntax* di dalam *pseudocode*. Jadi *pseudocode* digunakan untuk menggambarkan logika urutan dari program tanpa memandang bagaimana bahasa pemrogramannya.

## 3) *Flowchart*

Dalam *structure English* / struktur Indonesia digambarkan tahap-tahap penyelesaian masalah dengan menggunakan kata-kata (teks). Kelemahan cara ini adalah dalam penyusunan algoritma sangat dipengaruhi oleh tata bahasa pembuatnya, sehingga kadang-

kadang orang lain sulit memahaminya. Oleh sebab itu kemudian dikembangkan metode yang menggambarkan tahap-tahap pemecahan masalah dengan merepresentasikan simbol-simbol tertentu yang mudah dimengerti, mudah digunakan dan standar.

## 10. Himpunan

Himpunan adalah konsep dasar dari semua cabang matematika. Konsep tentang himpunan pertama kali dikemukakan oleh seorang ahli matematika berkebangsaan Jerman, yaitu Gerorge Cantor (1918), akhir abad ke-19. Konsep himpunan pada saat itu masih menjadi bahan perdebatan dan baru pada tahun 1920, konsep ini mulai digunakan sebagai landasan matematika.

Himpunan adalah sekumpulan objek yang didefinisikan dengan jelas dan dapat dibeda-bedakan (Suryanti & Zawawi, 2020). Objek yang dimaksud dapat berupa bilangan, manusia, hewan, tumbuhan, negara dan sebagainya. Objek ini selanjutnya dinamakan anggota atau elemen dari himpunan itu. Syarat tertentu dan jelas dalam menentukan anggota suatu himpunan ini sangat penting karena untuk membedakan mana yang menjadi anggota himpunan dan mana yang bukan merupakan anggota himpunan. Inilah yang kemudian dinamakan himpunan yang terdefinisi dengan baik (*well-defined set*).

Contoh:

- a. Himpunan semua huruf hidup dari abjad, yaitu a, i, u, e, o
- b. Himpunan semua bilangan genap, yaitu 0, 2, 4, 6, ...
- c. Himpunan semua bilangan riil  $x$  yang memenuhi  $x^2 - 3x - 4 = 0$

### a) Penyajian Cara Himpunan

#### 1) Enumerasi

Contoh.

- Himpunan empat bilangan asli pertama:  $A = \{1, 2, 3, 4\}$ .
- Himpunan lima bilangan genap positif pertama:  $B = \{4, 6, 8, 10\}$ .
- $C = \{\text{kucing, a, Amir, 10, paku}\}$
- $R = \{a, b, \{a, b, c\}, \{a, c\}\}$

- $C = \{a, \{a\}, \{\{a\}\}\}$
- $K = \{\{\}\}$
- Himpunan 100 buah bilangan asli pertama:  $\{1, 2, \dots, 100\}$
- Himpunan bilangan bulat ditulis sebagai  $\{\dots, -2, -1, 0, 1, 2, \dots\}$ .

#### Keanggotaan

$x \in A$  :  $x$  merupakan anggota himpunan  $A$ ;

$x \notin A$  :  $x$  bukan merupakan anggota himpunan  $A$ .

#### 2) Simbol-simbol Baku

$P$  = himpunan bilangan bulat positif =  $\{1, 2, 3, \dots\}$

$N$  = himpunan bilangan alami (natural) =  $\{1, 2, \dots\}$

$Z$  = himpunan bilangan bulat =  $\{\dots, -2, -1, 0, 1, 2, \dots\}$

$Q$  = himpunan bilangan rasional

$R$  = himpunan bilangan riil

$C$  = himpunan bilangan kompleks

#### 3) Notasi Pembentuk Himpunan

Notasi:  $\{x \mid \text{syarat yang harus dipenuhi oleh } x\}$

Contoh.

- a.  $A$  adalah himpunan bilangan bulat positif yang kecil dari 5

$$A = \{x \mid x \text{ adalah bilangan bulat positif lebih kecil dari } 5\}$$

atau

$$A = \{x \mid x \in P, x < 5\}$$

yang ekuivalen dengan  $A = \{1, 2, 3, 4\}$

- b.  $M = \{x \mid x \text{ adalah mahasiswa yang mengambil kuliah IF2151}\}$

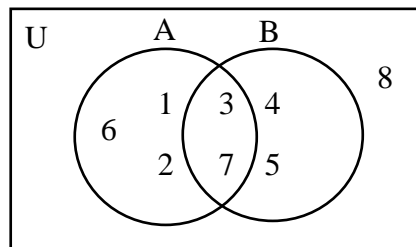
#### 4) Diagram Venn

Contoh

Misalkan  $S = \{1, 2, 3, 4, 5, 6, 7, 8\}$ ,  $A = \{1, 2, 3, 6, 7\}$ ,

dan  $B = \{3, 4, 5, 7\}$ .

Diagram Venn:



Gambar 2.8 Diagram Venn

(Sumber: Suryanti &amp; Zawawi, 2020)

## 11. Konsep Dasar Probabilitas

### a) Pengetahuan Probabilitas

Secara langsung atau tidak langsung, dasar-dasar *probability* memegang peranan penting dalam semua problema ilmu pengetahuan, *business*, dan segi-segi kehidupan lainnya yang mengandung unsur-unsur “ketidakpastian”, juga di dalam statistika. Inti dari pengetahuan *probability* adalah agar kita dapat menentukan “Pada suatu keadaan tertentu, apa yang mungkin terjadi” (Sumargo, 2021). Di dalam *study* “Apa yang mungkin terjadi?”, ada dua problema dasar yang dapat dijumpai, yaitu:

- 1) Problema membuat daftar dari semua yang terjadi pada suatu keadaan tertentu.
- 2) Problema penentuan ada berapa macam kejadian dapat berlangsung (tanpa membuat daftar terlebih dahulu). Hal ini penting karena kita tidak selamanya dapat membuat daftar semua kemungkinan yang dapat terjadi pada suatu keadaan tertentu.

### b) Permutasi

Suatu permutasi adalah penyusunan (*arrangement*) dari seluruh maupun sebagian di antara sejumlah  $n$  objek (Sumargo, 2021). Suatu penyusunan objek sebanyak  $r$  yang nilainya berapa saja, asal  $\leq n$  ke dalam suatu susunan tertentu dan dengan memperhatikan tata susunannya, dinamakan suatu permutasi  $n$  atau permutasi dari objek sebanyak  $n$  dengan setiap pengambilan sejumlah  $r$  objek.

Definisi lain dari permutasi adalah Banyaknya permutasi dari  $m$  elemen adalah jumlah maksimum dari cara-cara yang berbeda di dalam mengatur atau membuat urutan dari  $m$  elemen tersebut. (Permutasi dapat dengan mengatur setiap kalinya sejumlah elemen yang kurang dari  $m$ ).

Contoh:

- i. 3 5 2 4 1 adalah salah satu permutasi yang mungkin di antara angka 1, 2, 3, 4, 5. Demikian pula 1 3 2 4 5 dan 2 4 5 1 3 dan lain sebagainya.
- ii. Seluruh permutasi di 2 huruf hidup yang didapat dari 5 huruf hidup a, e, i, o, u ialah ae ai ao au ei eo eu io iu ea ia oa ua ie oe ue oi ui uo

Jumlah seluruh permutasi di antara  $n$  objek yang setiap pengambilannya sebanyak  $r$  objek, biasanya diberi simbol  $P(n, r)$  atau  ${}^n P_r$  dan lain sebagainya.

Permutasi  $r$  dari objek yang seluruhnya sebanyak  $n$  objek yang berlainan, hasil dari pilihan ke-1 =  $n$  cara, pilihan ke-2 =  $(n - 1)$  cara, pilihan ke-3 =  $(n - 2)$  cara dan seterusnya dan terakhir ialah pilihan ke  $r$  (ditentukan permutasi  $r$ ), ialah sebanyak  $(n - r + 1)$  cara. Jadi, seluruh jumlah permutasi  $r$  dari  $n$  objek yang berlainan, didapat cara permutasi sebanyak  $n(n - 1)(n - 2) \dots (n - r + 1)$  cara:

$$\frac{n(n - 1)(n - 2) \dots (n - r + 1)(n - r)!}{(n - r)!} = \frac{n!}{(n - r)!}$$

$$P(n - 1) = {}^n P_r \frac{n!}{(n - r)!}$$

Keterangan:

$P$  : Permutasi

$n$  : jumlah seluruh objek

$r$  : jumlah anggota permutasi

Contoh:

1. Seluruh jumlah yang mungkin untuk permutasi 3 huruf dari seluruh 4 huruf a, b, c, d

Penyelesaian:

$$n = 4$$

$$r = 3$$

$${}^n P_r = \frac{n!}{(n-r)!} = \frac{4!}{(4-3)!} = 4! = 24$$

Seluruh kemungkinan susunan 3 huruf dari permutasi
abc, acb, bac, bca, cab, cba
abd, adb, bad, bda, dab, dba
acd, adc, cad, cda, dac, dca
bcd, bdc, cbd, cdb, dbc, dcb

### C. Kerangka Berpikir

Permasalahan sampah adalah permasalahan yang sampai saat ini masih perlu perhatian khusus di Indonesia. Dalam UU No 18 Tahun 2008, sampah merupakan semua sisa kegiatan sehari-hari manusia atau proses alam yang berbentuk padat atau semi padat berupa zat organik atau anorganik bersifat dapat terurai atau tidak dapat terurai yang dianggap sudah tidak berguna lagi dan dibuang ke lingkungan. Saat ini masih banyak orang yang kurang peduli dengan pengelompokan sampah sesuai dengan jenisnya. Sampah yang dibuang sembarangan tanpa dikelompokkan sesuai dengan jenisnya menyebabkan pengolahan sampah menjadi tidak efisien.



### **BAB III**

## **METODE PENELITIAN**

#### A. Jenis Penelitian

Penelitian ini menggunakan metode penelitian dan pengembangan. Metode penelitian dibagi menjadi 3 tahapan utama, yaitu Studi pendahuluan, Pancangan algoritma, dan Pengujian.

#### B. Lokasi/Fokus Penelitian

Lokasi penelitian ini dilaksanakan di Laboratorium Robotika Universitas PGRI Semarang Universitas PGRI Semarang yang beralamat di Jln. Lontar No.1, Karangtempel, Semarang Timur, Kota Semarang, Jawa Tengah 50232.

#### C. Variabel Penelitian

Variabel penelitian pada dasarnya adalah segala sesuatu yang berbentuk apa saja yang ditetapkan oleh peneliti untuk dipelajari sehingga diperoleh informasi tentang hal tersebut, kemudian ditarik kesimpulannya (Sugiyono, 2004). Pada penelitian ini variabel yang akan digunakan adalah jarak dan jenis sampah.

##### 1. Variabel Independen( bebas )

Variabel independen atau sering juga disebut sebagai variabel *stimulus, prediktor, antecedent* dan dalam bahasa Indonesia sering juga disebut sebagai variabel bebas. Variabel bebas adalah variabel yang mempengaruhi atau yang menjadi sebab perubahannya atau timbulnya variabel dependen (terikat). Variabel bebas yang ditentukan oleh peneliti dalam penelitian ini adalah:

###### a) Jenis sampah

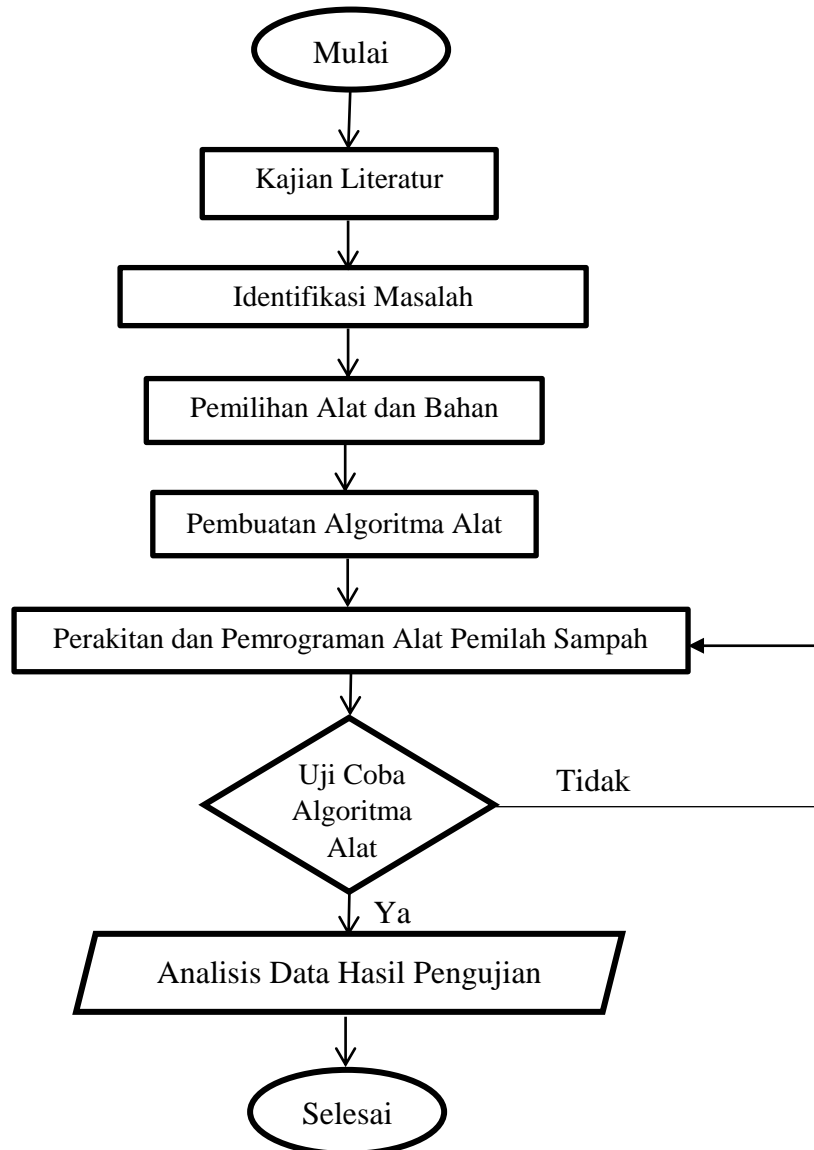
##### 2. Variabel Dependen (terikat)

Merupakan variabel yang dipengaruhi atau yang menjadi akibat karena adanya variabel bebas. Variabel terikat pada penelitian ini

adalah nilai *Inductive Proximity sensor*, *capacitive Proximity sensor*, dan *infrared sensor*.

#### D. Prosedur Penelitian

##### 1. Diagram Alir Penelitian



Gambar 3. 1 Diagram Alir Penelitian

Sumber: (Dokumentasi Pribadi, 2022)

##### 2. Tahap Studi Pendahuluan

Pada tahap studi pendahuluan ini ada dua tahap yang dilakukan oleh peneliti. Tahap awal peneliti melakukan analisis permasalahan dan potensi yang ada. Kemudian tahap selanjutnya adalah tahap perancangan algoritma program. Berikut adalah penjelasan dari tahap-tahap yang akan peneliti lakukan.

a. Analisis Permasalahan dan Potensi

Analisis permasalahan merupakan tahapan dimana dilakukan observasi untuk mengetahui permasalahan dasar yang ada di masyarakat tentang pemilahan dan pengelolaan sampah. Analisis permasalahan dilaksanakan berdasarkan observasi kondisi dan potensi yang ada di lingkungan sekitar. Melalui analisis permasalahan tersebut akan diperoleh data awal mengenai permasalahan yang ada masyarakat sekitar. Berdasarkan observasi permasalahan dan potensi dapat dicari solusi dari permasalahan tersebut dengan melihat potensi teknologi yang ada saat ini.

b. Perancangan Algoritma Alat Pemilah Sampah

Pada tahap ini mulai dikumpulkan komponen terhadap alat yang akan dirancang. Pengumpulan komponen didapatkan dengan studi literatur. Setelah itu, mulai dirancang alat yang akan dikembangkan. Alat yang dikembangkan merupakan perangkat pendeteksi jenis sampah yang menggunakan mikrokontroler dan beberapa sensor pendukung. Untuk ini, diperlukan rancangan rangkaian semua perangkat keras yang digunakan. Kemudian, agar alat ini dapat berfungsi secara maksimal, perlu dilakukan konfigurasi pada sensor dan penerapan algoritma agar alat dapat membedakan jenis sampah menggunakan beberapa sensor yang di konfigurasi menjadi satu kesatuan. Setelah rancangan selesai dibuat kemudian alat dikembangkan dengan menggunakan teknik *prototyping* dan metode uji coba *trial and error*.

3. Tahap Studi Pengembangan

a. Uji Coba Produk

Uji coba yang akan dilakukan pada alat pemilah sampah dibagi menjadi 3. Setelah *prototype* alat selesai dibuat, kemudian dilakukan pengujian terhadap *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor*. Hasil dari pembacaan sensor kemudian di analisa dan dibandingkan dengan perhitungan atau perkiraan probabilitas pembacaan sensor yang telah dibuat. Kemudian dari hasil pembacaan sensor tersebut juga dibuat algoritma yang dapat memilah sampah sesuai dengan pembacaan dari *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor*.

Tabel 3. 1 Pengujian Nilai Sensor

<b>No</b>	<b>Jenis Sampah</b>	<b>Pengambilan Data</b>	<b>Capacitive Proximity Sensor</b>	<b>Inductive Proximity Sensor</b>	<b>Infrared Proximity Sensor</b>
<b>1</b>		I			
		II			
		III			
<b>2</b>		I			
		II			
		III			
<b>3</b>		I			
		II			
		III			
<b>4</b>		I			
		II			
		III			

Setelah algoritma selesai dibuat lalu algoritma tersebut diubah menjadi bahasa pemrograman yang sesuai dengan mikrokontroler yang digunakan. Pada penelitian ini peneliti menggunakan bahasa c untuk membuat algoritma program. Program yang telah dibuat kemudian

diaplikasikan ke *prototype* alat pemisah sampah dan dilakukan uji coba fungsi alat.

#### b. Revisi Produk

Setelah melalui uji coba, maka dapat dilihat sejauh mana keefektifan dan kehandalan algoritma alat pemilah sampah yang telah dibuat dalam membedakan jenis sampah. Kemudian dilakukan perbaikan-perbaikan terhadap kekurangan yang ada pada algoritma program sehingga alat dapat berfungsi sesuai dengan harapan yang diinginkan.

#### E. Teknik Analisis Data

Teknik analisis data bertujuan menguraikan dan memecahkan masalah yang berdasarkan data yang diperoleh. Teknik analisis data yang digunakan menggunakan Metode Analisis Deskriptif. Analisis Deskriptif adalah sebuah metode yang dilakukan untuk mengetahui keberadaan variabel mandiri, baik hanya pada satu variabel atau lebih tanpa membuat perbandingan dan mencari hubungan variabel itu dengan variabel yang lain.

#### F. Jadwal

No	Bulan Ke-	1	2	3	4	5
1	Analisis potensi dan masalah					
2	Pengumpulan data dan informasi					
3	Perancangan Alat Pemilah Sampah Otomatis					
4	Pembuatan Alat Pemilah Sampah Otomatis					
5	Uji coba Alat Pemilah Sampah Otomatis					
6	Revisi Alat Pemilah Sampah Otomatis					
7	Pengujian dan Pengambilan Data Alat					

	Pemilah Sampah Otomatis					
8	Analisis Data					

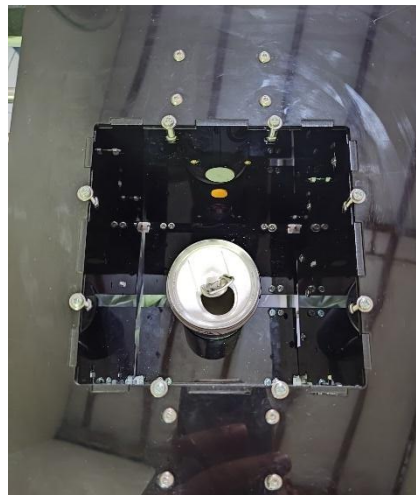
Tabel 3. 2 Jadwal Penelitian

## BAB IV

### HASIL DAN PEMBAHASAN

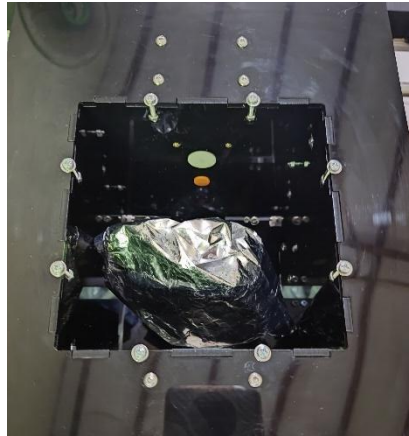
#### A. Data Hasil Pengujian Sensor

Pengujian dilakukan dengan menempatkan beberapa sampah di tempat penampung sampah sementara. Agar data yang didapat lebih akurat pengujian masing-masing jenis sampah dilakukan sebanyak tiga kali. Setiap hasil pembacaan sensor kemudian dicatat pada tabel yang sudah disiapkan. Pengujian dilakukan seperti yang terlihat pada gambar 4. 1 dan gambar 4. 2.



Gambar 4. 1 Pengujian Sampah Kaleng

Sumber: (Dokumentasi Pribadi, 2022)



Gambar 4. 2 Pengujian Sampah Kemasan *Snack*

Sumber: (Dokumentasi Pribadi, 2022)

Tabel 4. 1 Data Pengujian Nilai Sensor

<b>No</b>	<b>Jenis Sampah</b>	<b>Pengambilan Data</b>	<b><i>Capacitive Proximity Sensor</i></b>	<b><i>Inductive Proximity Sensor</i></b>	<b><i>Infrared Proximity Sensor</i></b>
<b>1</b>	Kayu	I	0	1	0
		II	0	1	0
		III	0	1	0
<b>2</b>	Kaleng	I	0	0	0
		II	0	0	0
		III	0	0	0
<b>3</b>	Botol Plastik	I	1	1	0
		II	1	1	0
		III	1	1	0
<b>4</b>	Kain	I	1	1	0
		II	1	1	0
		III	1	1	0
<b>5</b>	Kertas	I	1	1	0
		II	1	1	0
		III	1	1	0
<b>6</b>	Daun	I	0	1	0

		II	0	1	0
		III	0	1	0
		I	0	1	0
7	Gelas Kaca	II	0	1	0
		III	0	1	0
		I	0	1	0
8	Botol Kaca	II	0	1	0
		III	0	1	0
		I	0	1	0

## B. Himpunan Probabilitas Sensor

Dalam teori probabilitas biasanya kita mempelajari gejala acak (*random*) sebagai lawan dari gejala yang tertentu atau deterministik. Dalam hal ini kita ingin mempelajari hasil percobaan dan percobaan ini tidak selalu menghasilkan hasil yang sama. Persoalan kita adalah mengumpulkan semua hasil yang mungkin dari percobaan ini, dan ini berfungsi sebagai himpunan semesta  $S$ . Himpunan bagian  $A$ ,  $B$ ,  $C$  ... dari  $S$  menyatakan kejadian yang mungkin muncul dan ingin diketahui probabilitas atau peluangnya untuk terjadi. Dalam menghitung probabilitas tersebut biasanya kita menggunakan manipulasi teori himpunan yang telah kita bahas di landasan teori.

Pada penelitian kali ini kita menggunakan 3 sensor *proximity* (*capacitive proximity sensor*, *inductive proximity sensor*, dan *infrared sensor*) sebagai input untuk menentukan jenis sampah. Penyusunan probabilitas nilai pembacaan sensor urut sesuai dengan penyebutan ketiga sensor di atas. Hasil yang didapat dari tiap sensor adalah angka 0 atau 1. Jadi, dalam hal ini himpunan semesta  $S$  adalah

$$S = \{(0, 0, 0), (0, 0, 1), (0, 1, 0), (1, 0, 0), (0, 1, 1), (1, 0, 1), (1, 1, 0), (1, 1, 1)\}$$

Dari penjabaran himpunan semesta di atas, ada 8 kemungkinan kombinasi pembacaan 3 sensor yang digunakan. Apabila kita melihat dari data pengujian sensor dengan beberapa jenis sampah maka akan ada beberapa kombinasi yang



tidak muncul pada data pengujian sensor. Kombinasi yang tidak muncul adalah (0, 0, 1), (1, 0, 0), (0, 1, 1), (1, 0, 1) dan kombinasi yang muncul adalah (0, 0, 0), (0, 1, 0), (1, 1, 0), (1, 1, 1).

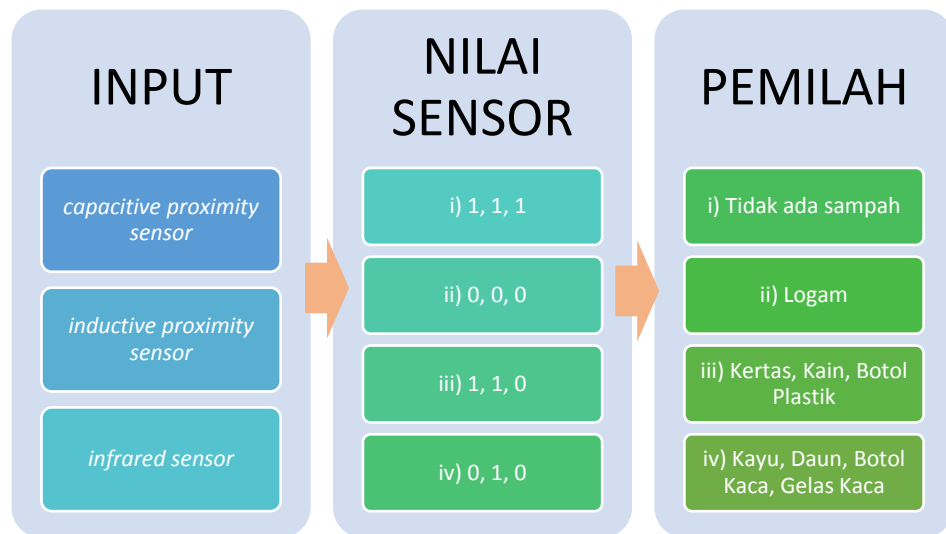
### C. Perancangan Algoritma

Sebuah algoritma pada dasarnya adalah membuat alat bantu untuk menyelesaikan suatu masalah. Algoritma yang akan dibuat kali bertujuan untuk dapat mengkombinasikan nilai pembacaan *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor* agar dapat mengidentifikasi jenis sampah. Pada sub bab sebelumnya sudah dijabarkan bahwa ada 8 kemungkinan kombinasi yang dihasilkan dari nilai sensor. Apabila kita kelompokkan sesuai dengan nilai sensor maka pengelompokan jenis sampah dapat dilihat pada table 4. 2 dibawah ini.

Tabel 4. 2 Pengelompokan jenis sampah sesuai nilai sensor

No	Nilai Sensor	Sampah
1	1, 1, 1	Tidak ada sampah
2	0, 0, 0	Logam (Kaleng)
3	1, 1, 0	Kertas, Kain, Botol Plastik
4	0, 1, 0	Kayu, Daun, Botol Kaca, Gelas Kaca

Setelah data pengelompokan sampah tersedia maka langkah selanjutnya adalah menyusun konsep/rancangan/desain penyelesaian yang akan di aplikasikan pada sampah. Pada algoritma ini langkah selanjutnya adalah menyortir atau memisahkan sampah sesuai dengan pengelompokan yang ada pada tabel diatas. Langkah selanjutnya adalah mengimplementasikan hasil rancangan algoritma yang telah dibuat kedalam bentuk program yang terstruktur. Pada penelitian ini program di buat dengan menggunakan bahasa pemrograman C.

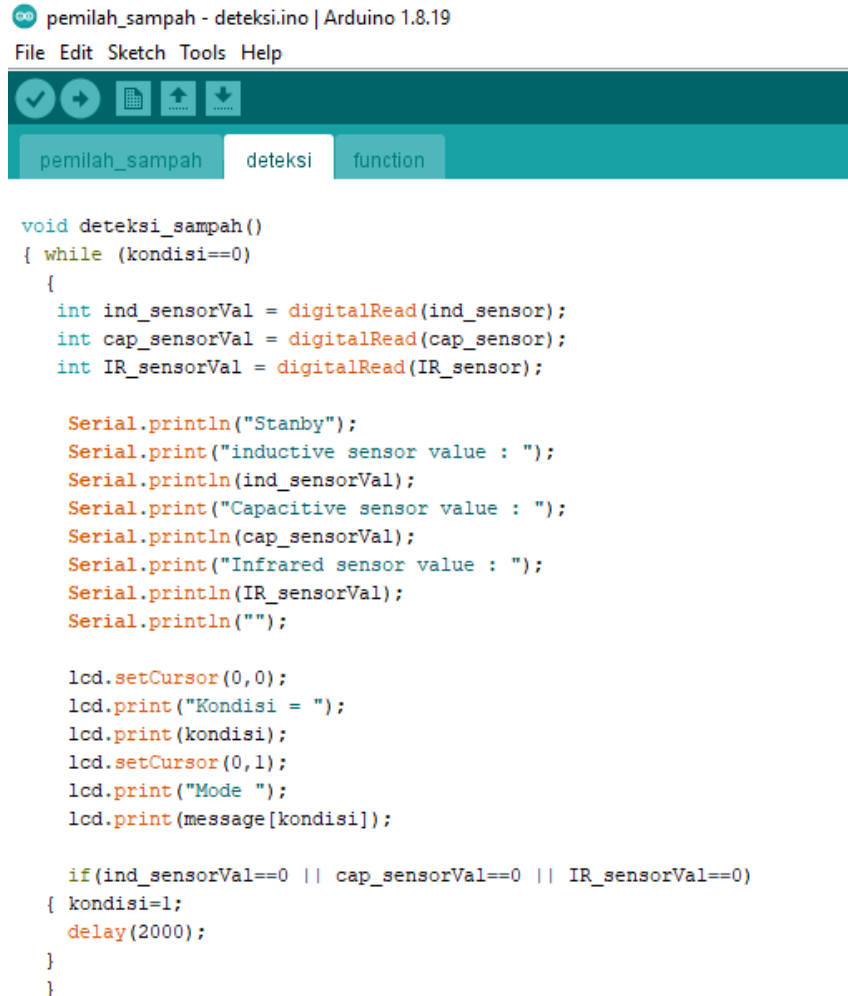


Gambar 4. 3 Rancangan Algoritma Pemilah Sampah

Sumber: (Dokumentasi Pribadi, 2022)

#### D. Implementasi

Algoritma pemilah sampah yang telah diubah menjadi *coding* atau program dengan menggunakan bahasa C selanjutnya di implementasikan pada *prototype* alat pemilah sampah yang telah dibuat. Algoritma yang telah dibuat dibagi menjadi beberapa kondisi dari kondisi 0 (nol) sampai kondisi 6 (enam). Program untuk kondisi 0 dapat dilihat pada gambar 4. 4.



```

pemilah_sampah - deteksi.ino | Arduino 1.8.19
File Edit Sketch Tools Help

pemilah_sampah deteksi function

void deteksi_sampah()
{ while (kondisi==0)
  {
    int ind_sensorVal = digitalRead(ind_sensor);
    int cap_sensorVal = digitalRead(cap_sensor);
    int IR_sensorVal = digitalRead(IR_sensor);

    Serial.println("Stanby");
    Serial.print("inductive sensor value : ");
    Serial.println(ind_sensorVal);
    Serial.print("Capacitive sensor value : ");
    Serial.println(cap_sensorVal);
    Serial.print("Infrared sensor value : ");
    Serial.println(IR_sensorVal);
    Serial.println("");

    lcd.setCursor(0,0);
    lcd.print("Kondisi = ");
    lcd.print(kondisi);
    lcd.setCursor(0,1);
    lcd.print("Mode ");
    lcd.print(message[kondisi]);

    if(ind_sensorVal==0 || cap_sensorVal==0 || IR_sensorVal==0)
    { kondisi=1;
      delay(2000);
    }
  }
}

```

Gambar 4. 4 Program Kondisi 0

Sumber: (Dokumentasi Pribadi, 2022)

Pada kondisi 0, alat diposisikan dalam keadaan *stanby* dan sensor siap untuk membaca jenis sampah yang dimasukkan. Selama kondisi pada program masih bernilai 0, maka program akan selalu membaca nilai dari ketiga sensor yang ada. Nilai dari tiap sensor akan ditampilkan pada *serial monitor*. Apabila salah satu sensor bernilai 0 (mendeteksi benda) maka kondisi akan bernilai 1 dan program akan terjeda selama 2 detik, kemudian berlanjut ke algoritma selanjutnya.



```

//////////      KONDISI 1      //////////
while(kondisi==1)
{
  int ind_sensorVal = digitalRead(ind_sensor);
  int cap_sensorVal = digitalRead(cap_sensor);
  int IR_sensorVal = digitalRead(IR_sensor);

  if(ind_sensorVal==0 && cap_sensorVal==0 && IR_sensorVal==0)
  {
    //logam          // posisi 1
    kondisi = 2;
    Serial.println("Kondisi 1");
  }

  if(ind_sensorVal==1 && cap_sensorVal==1 && IR_sensorVal==0)
  {
    //botol plastik, kain, kertas          // posisi 2
    kondisi = 3;
    Serial.println("Kondisi 2");
  }

  if(ind_sensorVal==1 && cap_sensorVal==0 && IR_sensorVal==0)
  {
    //kayu & sampah basah          // posisi 3
    kondisi = 4;

    Serial.println("Kondisi 3");
  }
  delay(50);
}

```

Gambar 4. 5 Program Kondisi 1

Sumber: (Dokumentasi Pribadi, 2022)

Saat kondisi pada program bernilai 1, maka program akan membaca nilai dari *inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor*. Jika *inductive proximity sensor* bernilai 0, *capacitive proximity sensor* bernilai 0, dan *infrared sensor* bernilai 0 maka program akan masuk ke kondisi 2. Kondisi 2 ini berarti sensor mendeteksi sampah berjenis logam. Selanjutnya, Jika *inductive proximity sensor* bernilai 1, *capacitive proximity sensor* bernilai 1, dan *infrared sensor* bernilai 0 maka program akan masuk ke kondisi 3.

Kondisi 3 ini berarti sensor mendeteksi sampah kertas, kain atau botol plastik. Kemudian, apabila *inductive proximity sensor* bernilai 1, *capacitive proximity sensor* bernilai 0, dan *infrared sensor* bernilai 0 maka program akan masuk ke kondisi 4. Kondisi 4 ini berarti sensor mendeteksi sampah kayu, kemasan *snack* atau sampah basah.



The image shows a screenshot of the Arduino IDE interface. The title bar reads 'pemilah\_sampah - deteksi.ino | Arduino 1.8.19'. The menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Below the menu bar is a toolbar with icons for a checkmark, a right arrow, a document, an up arrow, and a down arrow. The main editor area shows the following code:

```

//////////////////////////////// KONDISI 2 //////////////////////////////////
while(kondisi==2)
{
  int limit_1 = digitalRead(lim_1);
  int limit_2 = digitalRead(lim_2);
  int limit_3 = digitalRead(lim_3);
  if (limit_1==1)
  {
    stepperKR(); // kiri pengguna
  }

  if (limit_1==0)
  {
    kondisi=5;
  }
}

```

Gambar 4. 6 Program Kondisi 2

Sumber: (Dokumentasi Pribadi, 2022)

Pada saat kondisi pada program bernilai 2, 3, atau 4, program akan memerintahkan alat untuk membaca nilai dari *limit switch*. Nilai dari *limit switch* ini digunakan untuk mengetahui posisi dari penampung sampah sementara. Kemudian setelah posisi dari penampung sampah sementara diketahui, selanjutnya motor stepper akan menggerakkan penampung sampah sementara pada posisi yang sesuai dengan jenis sampah yang di deteksi. Apabila penampung sampah sementara telah berada pada posisi yang sesuai maka nilai kondisi akan berubah menjadi 5.

```

//////////      KONDISI 5      //////////
while(kondisi==5)
{
    baca_sensor_sr04();
    if(cm > 10)
    {
        servo_buka();
        delay(1000);
        servo_tutup();
        delay(500);
        kondisi=6;
    }
    if(cm <= 10)
    {
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("SAMPAH PENUH");
        delay(3000);
        lcd.clear();
        kondisi=6;
    }
}

```

Gambar 4. 7 Program Kondisi 5

Sumber: (Dokumentasi Pribadi, 2022)

Pada gambar 4.7 saat kondisi pada program bernilai 5, sensor ultrasonik sr04 akan membaca volume dari tempat sampah. Apabila tempat sampah masih belum terisi penuh maka servo akan membuka penampung sampah sementara dan kondisi pada program akan bernilai 6. Apabila tempat sampah masih telah terisi penuh maka servo tidak akan membuka penampung sampah sementara dan kemudian kondisi pada program akan bernilai 6.

Selanjutnya saat kondisi bernilai 6, alat akan membaca nilai dari ketiga *limit switch* yang ada. Nilai dari *limit switch* ini digunakan untuk mengetahui posisi dari penampung sampah sementara. Kemudian setelah posisi dari penampung sampah sementara diketahui, selanjutnya motor stepper akan menggerakkan penampung sampah sementara kembali ke posisi tengah.

Gambar 4. 8 Program Kondisi 6  
 Sumber: (Dokumentasi Pribadi, 2022)

## BAB V

### KESIMPULAN DAN SARAN

## A. Kesimpulan

Berdasarkan hasil perancangan dan implementasi yang telah dilakukan maka dapat diambil beberapa kesimpulan sebagai berikut:

1. *Inductive proximity sensor*, *capacitive proximity sensor*, dan *infrared sensor* dapat digunakan untuk memilah beberapa jenis sampah dengan mengkonfigurasi nilai hasil pembacaan setiap sensor. Jika *inductive proximity sensor* bernilai 0, *capacitive proximity sensor* bernilai 0, dan *infrared sensor* bernilai 0 berarti sensor mendeteksi sampah berjenis logam. Selanjutnya, Jika *inductive proximity sensor* bernilai 1, *capacitive proximity sensor* bernilai 1, dan *infrared sensor* bernilai 0 berarti sensor mendeteksi sampah kertas, kain atau botol plastik. Kemudian, apabila *inductive proximity sensor* bernilai 1, *capacitive proximity sensor* bernilai 0, dan *infrared sensor* bernilai 0 berarti sensor mendeteksi sampah kayu, kemasan *snack* atau sampah basah.
2. Algoritma pemilah sampah telah berhasil dibuat dengan menggunakan bahasa C selanjutnya di implementasikan pada *prototype* alat pemilah sampah yang telah dibuat. Algoritma yang telah dibuat dibagi menjadi beberapa kondisi dari kondisi 0 (nol) sampai kondisi 6 (enam) dimana setiap kondisi memiliki fungsi kerja masing-masing.

## B. Saran

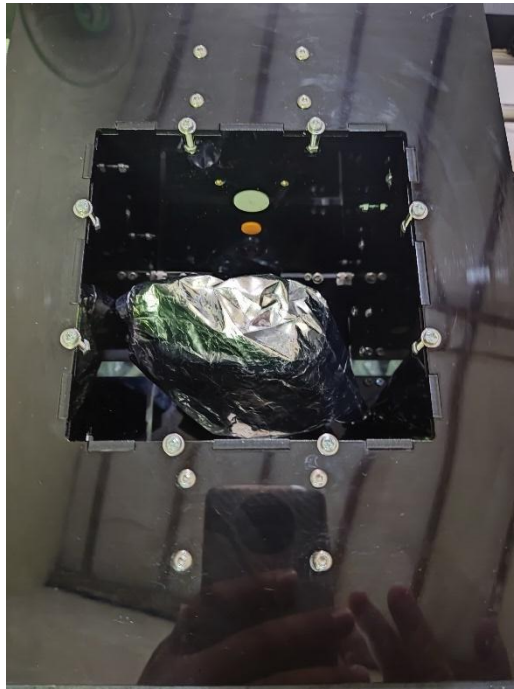
Algoritma pemilah sampah yang dibuat masih memiliki ruang pengembangan yang lebih luas lagi. Seperti penggunaan sensor atau perangkat lain yang dapat lebih akurat dan spesifik dalam mendeteksi jenis sampah. Untuk pengembangan selanjutnya alat juga dapat ditambahkan kamera dan menggunakan *image processing* ataupun teknologi-teknologi lain yang sesuai. Untuk pengaplikasian alat secara langsung di masyarakat juga masih perlu dilakukan penelitian dan pengembangan agar alat lebih ekonomis dan mudah digunakan

## DAFTAR PUSTAKA

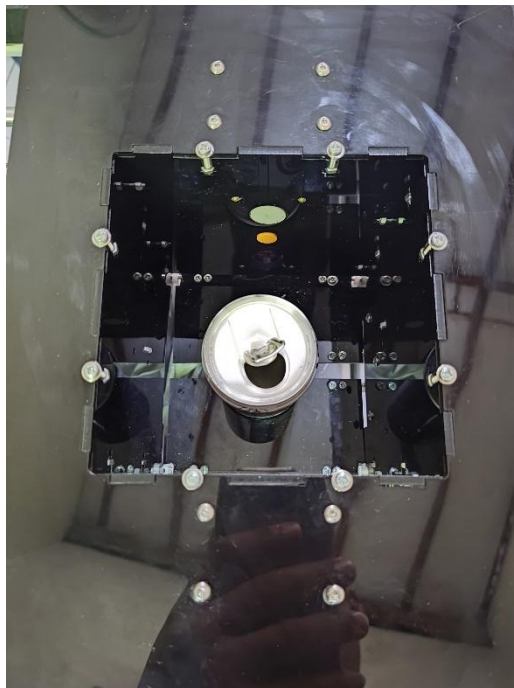


- Akbar, M., Anjasmara, S. D., & Wardhani, K. D. (2021). Rancang Bangun Alat Pendeteksi Sampah Organik dan Anorganik Menggunakan Sensor Proximity dan NodeMCU ESP8266. *Jurnal Politeknik Caltex Riau*, 290-299.
- circuits4you.com. (2018, Desember 31). *ESP32 DevKit ESP32-WROOM GPIO Pinout*. Retrieved from circuits4you.com: <https://circuits4you.com/2018/12/31/esp32-devkit-esp32-wroom-gpio-pinout/>
- Endra, R. Y., Cucus, A., Affandi, F. N., & Syahputra, M. B. (2019). Model Smart Room Dengan Menggunakan Mikrokontroler Arduino Untuk Efisiensi Sumber Daya. *Explore*, 1-9.
- Harmaji, L., & Khairullah. (2019). Rancang Bangun Tempat Pemilahan Sampah Logam dan Nonlogam Otomatis Berbasis Mikrokontroler. *Jurnal Ilmiah Komputer*, 73-82.
- Ismail, M. A., Abdullah, R. K., & Abdussamad, S. (2021). Tempat Sampah Pintar Berbasis Internet of Things (IoT) Dengan Sistem Teknologi Informasi. *Jambura Journal of Electrical and Electronics Engineering*, 7-12.
- Joseph, A. (2020, Mei 2). *Interface 16x2 LCD (parallel interface) with Arduino Uno*. Retrieved from create.arduino.cc: <https://create.arduino.cc/projecthub/akshayjoseph666/interface-16x2-lcd-parallel-interface-with-arduino-uno-2e87e2>
- Kho, D. (2020). *Pengertian Relay dan Fungsinya*. Retrieved from Teknik Elektronika: <https://teknikelektronika.com/pengertian-relay-fungsi-relay/>
- Puadi, O., & Hambali. (2022). Perancangan Alat Pemilahan Sampah Otomatis. *Jurnal Teknik Elektro Indonesia*, 1-14.
- Ridha, N. (2017). Proses Penelitian, Masalah, Variabel, dan Paradigma Penelitian. *Jurnal Hikmah*, 62-70.

- Saputra, D. A., Amarudin, Utami, N., & Setiawan, R. (2020). Rancang Bangun Alat Pemberi Pakan Ikan Menggunakan Mikrokontroler. *Jurnal ICTEE*, 15-19.
- Singgeta, R. L., & Manembu, P. D. (2021). Implementasi Sistem Monitoring Penggunaan Air Minum Pada Multiple Dispenser Berbasis IoT. *Rang Teknik Journal*, 127-133.
- Sitorus, L. (2015). *Algoritma dan Pemrograman*. Yogyakarta: ANDI.
- Sugiyono. (2004). *Metode Penelitian Administratif*. Bandung: CV Alfabeta.
- Suherman, Mardeni, Irawan, Y., & Sugiati. (2020). Rancang Bangun Tempat Sampah Otomatis Menggunakan Mikrokontroler dan Sensor Ultrasonik dengan Notifikasi Telegram. *Jurnal Ilmu Komputer*, 154-160.
- Sumargo, B. (2021). *Probabilitas Untuk Statistik*. Jakarta: UNJ PRESS.
- Suryanti, S., & Zawawi, I. (2020). *Pengantar Dasar Matematika*. Yogyakarta: Deepublish.
- Wafi, A., Setyawan, H., & Ariyani, S. (2020). Prototipe Sistem Smart Trash Berbasis IOT (Internet Of Things) dengan Aplikasi Android. *Jurnal Teknik Elektro dan Komputasi*, 20-29.
- Widodo, A. E., & Suleman. (2020). Otomatisasi Pemilah Sampah Berbasis Arduino Uno. *Indonesian Journal on Software Engineering*, 12-18.

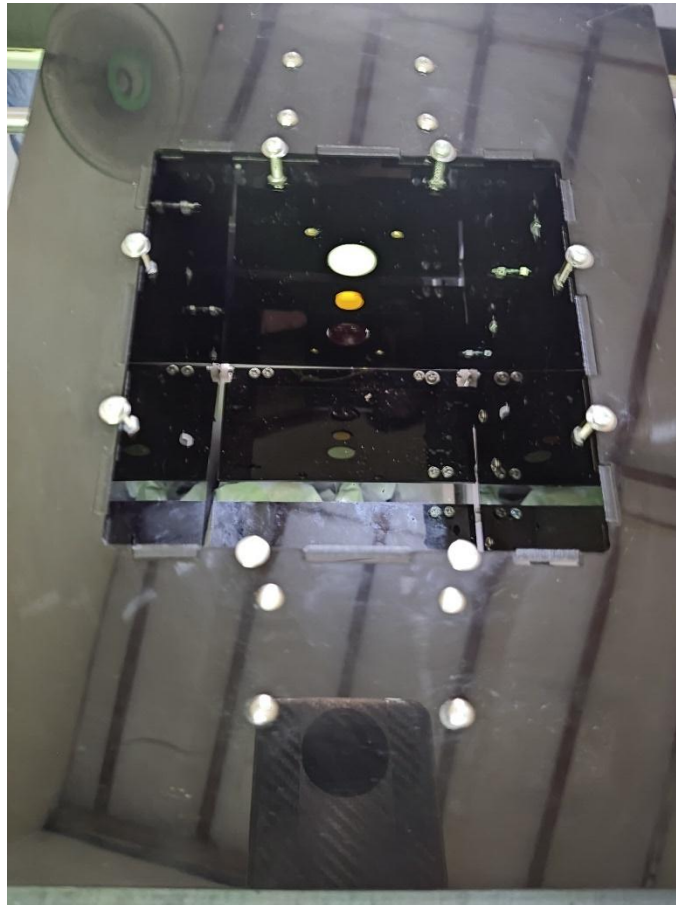


Gambar Pengujian Sampah Kemasan *Snack*



Gambar Pengujian Sampah Kaleng

Lampiran 2 Gambar Penampung Sampah Sementara



Gambar Penampung Sampah Sementara

Lampiran 3 Program Algoritma dalam Bahasa C (*coding*)

pemilah_sampah\$	deteksi	function
------------------	---------	----------

```

*/
#include <Servo.h>
#include <NewPing.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,16,2);

#define ind_sensor 17
#define cap_sensor 16
#define IR_sensor 5

#define lim_1 13
#define lim_2 12
#define lim_3 14

#define TRIGGER_PIN 2
#define ECHO_PIN 15
#define MAX_DISTANCE 300 //Maximum sensor distance is rated at 400-500cm.

NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);

static const int servoPin = 4;
Servo servol;

// Stepper
const int ena = 25;
const int dir = 26;
const int pul = 27;
const int interval = 70; //interval
boolean pulse = LOW;
////////////////////

int limit_1;
int limit_2;
int limit_3;

int ind_sensorVal;
int cap_sensorVal;
int IR_sensorVal;

int cm;

int kondisi = 0;
int posisi = 0;

char message[] = "0123456789";

```

```
void setup() {  
  
    servol.attach(servoPin);  
  
    Serial.begin(9600);  
  
    lcd.begin();  
    //lcd.init();  
    lcd.backlight();  
    //lcd.clear();  
  
    pinMode(ena, OUTPUT);  
    pinMode(dir, OUTPUT);  
    pinMode(pul, OUTPUT);  
  
    pinMode(lim_1, INPUT_PULLUP);  
    pinMode(lim_2, INPUT_PULLUP);  
    pinMode(lim_3, INPUT_PULLUP);  
    pinMode(ind_sensor, INPUT_PULLUP);  
    pinMode(cap_sensor, INPUT_PULLUP);  
    pinMode(IR_sensor, INPUT_PULLUP);  
  
}  
  
void loop() {  
    int limit_1 = digitalRead(lim_1);  
    int limit_2 = digitalRead(lim_2);  
    int limit_3 = digitalRead(lim_3);  
  
    int ind_sensorVal = digitalRead(ind_sensor);  
    int cap_sensorVal = digitalRead(cap_sensor);  
    int IR_sensorVal = digitalRead(IR_sensor);  
    deteksi_sampah();  
  
}
```

---

pemilah\_sampah §

deteksi §

function

```

void deteksi_sampah()
{ while (kondisi==0)
  {
  int ind_sensorVal = digitalRead(ind_sensor);
  int cap_sensorVal = digitalRead(cap_sensor);
  int IR_sensorVal = digitalRead(IR_sensor);

  Serial.println("Stanby");
  Serial.print("inductive sensor value : ");
  Serial.println(ind_sensorVal);
  Serial.print("Capacitive sensor value : ");
  Serial.println(cap_sensorVal);
  Serial.print("Infrared sensor value : ");
  Serial.println(IR_sensorVal);
  Serial.println("");

  lcd.setCursor(0,0);
  lcd.print("Kondisi = ");
  lcd.print(kondisi);
  lcd.setCursor(0,1);
  lcd.print("Mode ");
  lcd.print(message[kondisi]);

  if(ind_sensorVal==0 || cap_sensorVal==0 || IR_sensorVal==0)
  { kondisi=1;
    delay(2000);
  }
  }
  |
  ///////////////      KONDISI 1      ///////////////
  while(kondisi==1)
  {
  int ind_sensorVal = digitalRead(ind_sensor);
  int cap_sensorVal = digitalRead(cap_sensor);
  int IR_sensorVal = digitalRead(IR_sensor);

```

```

if(ind_sensorVal==0 && cap_sensorVal==0 && IR_sensorVal==0)
{
  //logam          // posisi 1
  kondisi = 2;
  Serial.println("Kondisi 1");
}

if(ind_sensorVal==1 && cap_sensorVal==1 && IR_sensorVal==0)
{
  //botol plastik, kain, kertas          // posisi 2
  kondisi = 3;
  Serial.println("Kondisi 2");
}

if(ind_sensorVal==1 && cap_sensorVal==0 && IR_sensorVal==0)
{
  //kayu & sampah basah          // posisi 3
  kondisi = 4;

  Serial.println("Kondisi 3");
}
}
delay(50);
}

//////////      KONDISI 2      //////////
while(kondisi==2)
{ int limit_1 = digitalRead(lim_1);
  int limit_2 = digitalRead(lim_2);
  int limit_3 = digitalRead(lim_3);
  if (limit_1==1)
  {
    stepperKR(); // kiri pengguna
  }

  if (limit_1==0)
  {
    kondisi=5;
  }
  //kondisi=0;
}

```



```

////////// KONDISI 3 //////////
while(kondisi==3)
{ int limit_1 = digitalRead(lim_1);
  int limit_2 = digitalRead(lim_2);
  int limit_3 = digitalRead(lim_3);
  if (limit_1==0)
  { posisi = 1;
    while(posisi==1)
    {
      int limit_1 = digitalRead(lim_1);
      int limit_2 = digitalRead(lim_2);
      int limit_3 = digitalRead(lim_3);
      stepperKN(); // kiri pengguna
      if(limit_2==0)
      {
        posisi = 0;
        kondisi=5;
      }
    }
  }
  if (limit_2==0)
  {
    kondisi=5;
  }
  if (limit_3==0)
  { posisi = 3;
    while(posisi==3)
    {
      int limit_1 = digitalRead(lim_1);
      int limit_2 = digitalRead(lim_2);
      int limit_3 = digitalRead(lim_3);
      stepperKR(); // kiri pengguna
      if(limit_2==0)
      {
        posisi = 0;
        kondisi=5;
      }
    }
  }
}

////////// KONDISI 4 //////////
while(kondisi==4)
{ int limit_1 = digitalRead(lim_1);
  int limit_2 = digitalRead(lim_2);
  int limit_3 = digitalRead(lim_3);
  if (limit_3==1)
  {
    stepperKN(); // kiri pengguna
  }
  if (limit_3==0)
  {
    kondisi=5;
  }
}

```

```

//////////      KONDISI 5      //////////
while(kondisi==5)
{
  baca_sensor_sr04();
  if(cm > 10)
  {
    servo_buka();
    delay(1000);
    servo_tutup();
    delay(500);
    kondisi=6;
  }
  if(cm <= 10)
  {
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("SAMPAH PENUH");
    delay(3000);
    lcd.clear();
    kondisi=6;
  }
}
//////////      KONDISI 6      //////////
while(kondisi==6)
{
  int limit_1 = digitalRead(lim_1);
  int limit_2 = digitalRead(lim_2);
  int limit_3 = digitalRead(lim_3);
  if (limit_1==0)
  { posisi = 1;
    while(posisi==1)
    {
      int limit_1 = digitalRead(lim_1);
      int limit_2 = digitalRead(lim_2);
      int limit_3 = digitalRead(lim_3);
      stepperKN(); // kiri pengguna

      if(limit_2==0)
      {
        posisi = 0;
        kondisi=0;
      }
    }
  }
  if (limit_2==0)
  {
    kondisi=0;
  }
}

```

```
if (limit_3==0)
{ posisi = 3;
  while(posisi==3)
  {
    int limit_1 = digitalRead(lim_1);
    int limit_2 = digitalRead(lim_2);
    int limit_3 = digitalRead(lim_3);
    stepperKR(); // kiri pengguna
    if(limit_2==0)
    {
      posisi = 0;
      kondisi=0;
    }
  }
}
}
////////// KEMBALI KE KONDISI 0 //////////
}
```

---

```

pemilah_sampah $ deteksi $ function $
void servo_buka()
{
    servol.write(90);
    Serial.println("buka");
    delay(20);
}
void servo_tutup()
{
    servol.write(0);
    Serial.println("tutup");
    delay(20);
}
void baca_sensor_sr04()
{ //kosong 33 cm    penuh = 10 CM
  cm = sonar.ping_cm(); //          kosong 33 cm
  delay(50);
  Serial.print("Ping: ");
  Serial.print(cm);
  Serial.println("cm");
}
void stepperKN()
{ digitalWrite(ena, LOW);
  digitalWrite(dir, HIGH); // low CW / high CCW
  digitalWrite(pul, HIGH);
  pulse = !pulse;
  digitalWrite(pul, pulse);
  delayMicroseconds(interval);
}
void stepperKR()
{ digitalWrite(ena, LOW);
  digitalWrite(dir, LOW); // low CW / high CCW
  digitalWrite(pul, HIGH);
  pulse = !pulse;
  digitalWrite(pul, pulse);
  delayMicroseconds(interval);
}

```