

**MODEL ROBOT TERKENDALI *GESTURE* TANGAN  
BERBASIS ADXL335 dan RF 433 MHz**

**PENELITIAN**



Disusun Oleh :

**GOGOR C. SETYAWAN**

**NIDN 05261270**

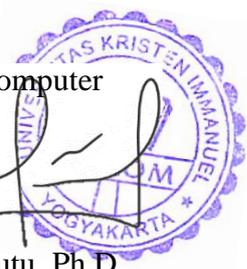
**PRODI TEKNIK INFORMATIKA FAKULTAS SAINS DAN KOMPUTER  
UNIVERSITAS KRISTEN IMMANUEL  
YOGYAKARTA  
2022**

## HALAMAN PENGESAHAN LAPORAN PENELITIAN

1. Judul : **Model Robot Terkendali *Gesture* Tangan Berbasis ADXL 335 dan RF 433 MHz**
2. Program Studi : Informatika
3. Fakultas : Fakultas Sain dan Komputer (FISKOM UKRIM)
4. Bidang Keahlian : Robotika
  
5. Ketua Peneliti : Gogor C. Setyawan, S.Si., M.Cs.
6. NIDN Ketua : 0526127001
7. Pangkat /Gol : -
8. Jabatan Akademik : Asisten Ahli
  
9. Anggota Peneliti : Rutnarnia Sara Stevani Sakoikoi
10. NIDN Anggota : -
  
11. Lokasi Penelitian : FISKOM - UKRIM - Yogyakarta
12. Afiliasi : Universitas Kristen Immanuel
13. Sumber Dana : LPPM UKRIM - TA 2021-2022
14. Besar Dana : Rp. 2.000.000.00

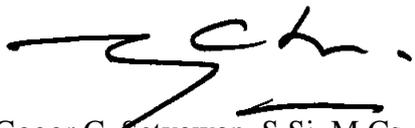
Yogyakarta, 29 maret 2022

Dekan  
Fakultas Sain dan Komputer



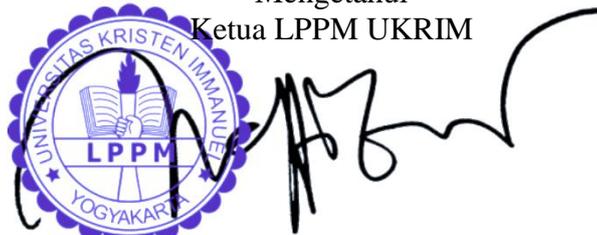
Sunneng Sandino Berutu. Ph.D  
NIDN. 0527027701

Ketua Peneliti



Gogor C. Setyawan, S.Si, M.Cs.  
NIDN: 0526127001

Mengetahui  
Ketua LPPM UKRIM



Ag. Rudatyo Himamunanto, S.Si, M.Kom.  
NIDN: 0517086901

## DAFTAR ISI

HALAMAN SAMBUNG .....	i
HALAMAN PENGESAHAN .....	ii
DAFTAR ISI.....	iii
DAFTAR GAMBAR .....	iv
DAFTAR TABEL.....	vi
ABSTRAK.....	vii
BAB 1 PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan Penelitian .....	2
1.4 Manfaat Penelitian .....	2
1.5 Batasan Masalah.....	3
1.6 Metode Penelitian.....	3
1.6.1 Metode Pengumpulan Data .....	3
1.6.2 Metode Pengembangan Sistem .....	4
1.7 Sistematika Penulisan .....	5
BAB II LANDASAN TEORI .....	6
2.1 Pengertian Mikrokontroler .....	6
2.2 Pengertian Arduino Uno .....	7
2.3 IDE Arduino.....	8
2.4 Pengertian Sensor.....	10
2.5 pengertian Actuators .....	10
2.6 Pengertian <i>Gesture</i> .....	11
2.7 Bahasa Pemrograman C .....	11
2.8 Driver Motor (Motor Shield L298) .....	12
2.9 Motor DC .....	13
2.10 Accelerometer ADXL345 .....	14
2.11 PWM (Pulse Width Modulation) .....	19
2.11.1 Konsep Dasar PWM.....	20
2.12 Sensor Ultrasonic (HCSR-04).....	22
2.12.1 Prinsip Kerja Sensor HCSR-04.....	25
2.13 Modul RF 433Mhz.....	26
BAB III ANALISA DAN PERANCANGAN SISTEM.....	26
3.1 Analisa Kebutuhan Sistem Robot .....	26
3.1.1 Misi Sistem Robot.....	26
3.1.2 Lingkungan Sistem Robot.....	26
3.1.3 Kebutuhan Sistem Sensor .....	29
3.1.4 Kebutuhan Sistem Aktuator dan Lokomosi .....	29
3.1.5 Komunikasi RF433Mhz.....	30
3.1.6 Kebutuhan Perangkat Lunak.....	32

3.2 Perancangan Umum Sistem Robot .....	32
3.3 Rancangan Data .....	33
3.4 Perancangan Perangkat Keras Robot .....	34
3.5 Perancangan Perangkat Lunak Robot .....	36
3.5.1 Algoritma Proses Pengolahan Data Accelerometer .....	36
3.5.2 Algoritma Cek Status Perintah Maju .....	39
3.5.3 Algoritma Cek Status Perintah Mundur .....	39
3.5.4 Algoritma Cek Status Perintah Rol Kanan .....	40
3.5.5 Algoritma Cek Status Perintah Rol Kiri .....	41
3.6 Fungsi Dasar Operational Sistem .....	41
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>42</b>
4.1 Hasil Implementasi .....	42
4.1.1 Hasil Pengujian Analisis PEAS Robot Mobil .....	42
4.1.2 Pembahasan <i>Performance Measure</i> .....	45
4.1.2.1 Pengukuran Performa dan Hasil Analisis .....	45
4.1.2.2 Pembahasan .....	48
4.1.3 Hasil Desain Alat .....	49
4.1.3.1 Rangkaian Gesture Robot Motor .....	49
4.1.3.2 Rangkaian Adxl345 dengan Arduino .....	50
4.1.3.3 Rangkaian Adxl345, Arduino, Power Bank dan Tx433 Mhz .....	51
4.2 Pembahasan .....	51
4.2.1 <i>Source Code</i> Program Khusus .....	52
4.2.1.1 Fungsi untuk Konversi Data Acceleration (x,y,z) .....	52
4.2.1.2 Baca Data Control/Perintah .....	55
4.2.1.3 Code Program untuk Berhenti .....	55
4.2.1.4 Code Program untuk Maju .....	56
4.2.1.5 Code Program untuk Mundur .....	57
4.2.1.6 Code Program untuk Kanan .....	58
4.2.1.7 Code Program untuk Kiri .....	59
4.2.2 Kelemahan Robot Mobil .....	6-
<b>BAB V KESIMPULAN .....</b>	<b>60</b>
5.1 Kesimpulan .....	63
<b>DAFTAR PUSTAKA .....</b>	<b>62</b>

## DAFTAR GAMBAR

Gambar 2.1	Arduino Uno.....	8
Gambar 2.2	Kabel USB <i>Board</i> Arduino Uno .....	8
Gambar 2.3	Tampilan Software IDE Arduino (Arifin, 2016).....	10
Gambar 2.4	Motor driver L298N .....	13
Gambar 2.5	<i>Accelerometer ADXL345 Tripple Axis</i> .....	17
Gambar 2.6	Diagram Blok ADXL345 .....	17
Gambar 2.7	Sumbu pada ADXL345 .....	18
Gambar 2.8	Respon Output Dan Orientasi Gravitasi Sensor Accelerometer.....	18
Gambar 2.9	Sinyal PWM .....	19
Gambar 2.10	Bentuk Gelombang Kotak Pulsa .....	20
Gambar 2.11	Perubahan Duty Cycle .....	21
Gambar 2.12	Variasi Perubahan PWM .....	22
Gambar 2.13	Sensor Ultrasonic (HC-SR04) .....	23
Gambar 2.14	Sistem pewaktu pada sensor HC-SR04 .....	24
Gambar 2.15	Sensor HCSR-04 .....	25
Gambar 2.16	Tampilan MLX90614.....	26
Gambar 3.1	Gambar lingkungan robot <i>gesture</i> .....	28
Gambar 3.2	Rancangan formasi sensor ADXL.....	28
Gambar 3.3	Rancangan Pengolahan Data <i>Acceleration</i> .....	29
Gambar 3.4	Rancangan formasi sensor Ultrasonic .....	30
Gambar 3.5	Rancangan Komunikasi RF433Mhz.....	32
Gambar 3.6	Skema umum perangkat lunka robot <i>gesture</i> .....	34
Gambar 3.7	Peletakan Sensor Ultrasonic pada robot .....	35
Gambar 3.8	Rangkaian Perangkat Keras <i>Receiver</i> dan Pengolah Data .....	36
Gambar 3.9	Diagram Alir Program Secara Keseluruhan .....	37
Gambar 3.10	Diagram Alir Proses Baca <i>acceleration</i> 1 .....	38
Gambar 3.11	Diagram Alir Proses Baca <i>acceleration</i> 2.....	38
Gambar 3.12	Diagram Alir Proses Baca <i>acceleration</i> 3.....	39
Gambar 3.13	Diagram Alir Proses Baca <i>acceleration</i> 4.....	39
Gambar 3.14	Diagram Alir Proses Baca <i>acceleration</i> 5.....	39
Gambar 3.15	Algoritma Cek Status Perintah Maju.....	40
Gambar 3.16	Algoritma Cek Status Perintah Mundur .....	41
Gambar 3.17	Algoritma Cek Status Perintah Rol Kanan.....	42
Gambar 3.17	Algoritma Cek Status Perintah Rol Kiri.....	43
Gambar 4.1	Rangkaian Gesture Robot Motor.....	52
Gambar 4.2	Rangkaian Adx1345 dengan Arduino .....	53
Gambar 4.3	Rangkaian Gesture Robot Controller .....	53

## **DAFTAR TABEL**

Tabel 4.1	Analisis PEAS Robot Mobil .....	46
Tabel 4.2	Hasil Pengujian I Analisis Peas.....	49
Tabel 4.3	Hasil Pengujian Sensitivitas Robot Mobil Terhadap Penabrakan.....	50

# MODEL ROBOT TERKENDALI GESTURE TANGAN BERBASIS ADXL335 dan RF 433 MHz

Gogor C. Setyawan, Rutnarnia Sara Stevani

Program Studi Teknik Informatika Fakultas Sains dan Komputer  
Universitas Kristen Immanuel Yogyakarta

## ABSTRAK

Perkembangan teknologi kontrol robot yang semakin maju pesat dari pergerakan robot yang dikontrol manusia dari jarak dekat, kemudian dapat dikontrol dari jarak jauh menggunakan *joystick* atau *remote control* dengan media kabel. Selanjutnya digantikan dengan media gelombang radio, hingga saat ini kendali robot dikembangkan menggunakan perintah manusia secara langsung, salah satunya menggunakan gesture tangan.

Perangkat keras yang dibutuhkan pada penelitian ini adalah Arduino Uno Motor Driver, Motor DC, ADXL345, Sensor Ultrasonic, Modul RF 433 Mhz, Arduino Mega IO Shield. Setelah dilakukan perancangan maka pengkodean dilakukan pada Transmitter dan Receiver dengan menggunakan aplikasi Arduino Uno IDE.

Kesimpulan yang diperoleh dari penelitian ini adalah Robot Mobil Kendali Jarak Jauh Berbasis Gesture telah dapat dibangun dengan menggunakan sensor ADXL345 dan sesuai apa yang dirancang. *Gesture Remote Robot* berhasil dikendalikan sesuai intruksi yang kirim oleh *Gesture Remote Controller* dengan hasil akurasi rata-rata pengujian yang didapatkan 97,2 point, dimana Robot dapat bergerak maju, mundur, berhenti, rol kiri, rol kanan, drift maju kiri, drift maju kanan, drift mundur kiri dan drift mundur kanan. Sensor ultrasonic telah dapat berfungsi, robot mobil memilih berhenti ketika ada penghalang atau objek didekatnya.

**Kata Kunci** : Robot berbasis Gesture, ADXL345, Sensor Ultrasonic, PWM, Arduino, Modul RX433 Mhz.

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Dunia robotika merupakan salah satu bentuk pengembangan teknologi yang semakin hari semakin banyak dikembangkan. Segala jenis robot diciptakan untuk memenuhi berbagai kebutuhan manusia. Mulai dari jenis robot yang dapat mempermudah pekerjaan rumah, robot yang digunakan dalam industri, maupun jenis robot yang digunakan sebagai hiburan.

Perkembangan teknologi kontrol robot yang semakin maju pesat dari pergerakan robot yang dikontrol manusia dari jarak dekat, kemudian dapat dikontrol dari jarak jauh menggunakan *joystick* atau *remote control* dengan media kabel. Selanjutnya digantikan dengan media gelombang radio, hingga saat ini kendali robot dikembangkan menggunakan perintah manusia secara langsung, salah satunya menggunakan *gesture* tangan.

Manusia lebih terbiasa berinteraksi menggunakan suara, posisi tubuh dan *gesture* (gerakan) tangan, penggunaan *gesture* tangan ini lebih mudah dilakukan karena lebih alami dan interaktif serta proses pembelajaran yang relatif lebih singkat. *Gesture* adalah bentuk komunikasi non-verbal dengan aksi tubuh yang mengkomunikasikan pesan-pesan tertentu, sebagai pengganti ucapan. Sistem pengenalan *gesture* tangan terus berkembang, terutama untuk menggantikan peran perangkat masukan seperti *mouse*, *keyboard* dan *joystick* yang digunakan pada interaksi manusia dengan komputer, namun perangkat ini memiliki

kelemahan diantaranya tidak alami bagi manusia dan jarak pengontrolan masih cukup dekat.

Aplikasi teknologi pengenalan *gesture* tangan telah banyak digunakan dalam berbagai bidang, salah satunya diterapkan dalam bidang robotika. Dalam bidang robotika, tujuan penerapan teknologi ini adalah untuk menciptakan sebuah sistem robotik yang mampu meniru pergerakan jari dan tangan manusia, sehingga dapat membantu manusia untuk melakukan berbagai hal yang tidak bisa dilakukan manusia secara langsung.

## **1.2 Rumusan Masalah**

Berdasarkan dari latar belakang diatas, rumusan masalah dalam penelitian ini adalah bagaimana Membangun Robot Mobil Kendali Jarak Jauh Berbasis *Gesture* Dengan Menggunakan Sensor Adxl345?

## **1.3 Tujuan**

Tujuan dari penelitian ini adalah untuk membangun Robot Mobil Kendali Jarak Jauh Berbasis *Gesture* Dengan Menggunakan Sensor Adxl345.

## **1.4 Manfaat Penelitian**

Manfaat dari penelitian ini diantaranya yaitu:

- a. Dapat mengendalikan robot cukup dengan menggunakan *gesture* (gerakan tangan).

- b. Untuk memperoleh gambaran nyata tentang penerapan dari ilmu dan teori selama ini telah diperoleh pada saat perkuliahan

## **1.5 Batasan Masalah**

Beberapa batasan masalah dalam penelitian ini yaitu:

- 1) Membangun Robot Mobil Kendali Jarak Jauh Berbasis *Gesture* Dengan Menggunakan Sensor Adxl345.
- 2) Aplikasi atau *Software* yang digunakan Arduino IDE.
- 3) Menggunakan sketch (Arduiuno Uno IDE).
- 4) Robot mobil dikendalikan dengan gesture tangan dengan menggunakan sensor ADXL345.
- 5) Perakitan robot mobil dibantu oleh Dosen Pembimbing 1
- 6) Penulis hanya meneliti robot mobil yang telah selesai dirakit dan mencoba mempelajari code program pengendalian robot mobil tersebut.

## **1.6 Metode Penelitian**

### **1.6.1 Metode Pengumpulan Data**

Pada penelitian ini, metode yang digunakan untuk mengumpulkan data yang menunjang penelitian penulis menggunakan beberapa metode antara lain :

1. Studi Literatur

Studi literatur yang digunakan untuk mendapatkan data-data mengenai perakitan robot mobil berbasis gesture tangan, sensor adxl345, arduino, motor

dan segala material penelitian yang digunakan dari berbagai sumber seperti buku, jurnal, penelitian, dan internet.

## 2. Observasi

Pada langkah ini dilakukan pengamatan dan pencatatan secara sistematis untuk melihat dan mengidentifikasi penelitian Robot mobil Kontrol Jarak jauh menggunakan gesture tangan dengan sensor adxl345.

### **1.6.2 Metode Pengembangan Sistem**

#### 1. Tahap analisis

Pada tahap ini bertujuan untuk menganalisa segala kebutuhan luar dan dalam system Robot mobil kendali jarak jauh berbasis gesture dengan menggunakan sensor ADXL345 baik dalam perakitan maupun code program.

#### 2. Tahap Perencanaan

Pada tahap perencanaan ini peneliti membuat prototype untuk sebagai gambaran atau panduan dalam pembuatan robot kendali jarak jauh menggunakan gesture agar lebih terarah dalam proses pengerjaannya.

#### 3. Tahap Pemodelan

Pada tahap ini peneliti membuat sketsa sehingga lebih mudah memahami gambaran besarnya (arsitekturanya) dari apa yang akan diteliti atau bagaimana robot mobil berbasis gesture tersebut akan dirakit atau dirancang.

#### 4. Tahap Pembuatan Program dan Pengujian

Pada tahap ini peneliti melakukan pembuatan program di arduinonya untuk dapat mengelola data dari sensor sehingga dapat melakukan control jarak jauh

dengan gesture tangan, dan setelahnya melakukan pengujian untuk program yang sudah dibuat apakah sudah berjalan dan sesuai dengan yang seharusnya.

### **1.7 Sistematika Penulisan**

Sistematika penulisan pada penelitian ini adalah sebagai berikut:

#### **BAB I : PENDAHULUAN**

Bab ini terdiri dari latar belakang, rumusan masalah, tujuan, manfaat penelitian, batasan masalah, metode penelitian dan sistematika penulisan.

#### **BAB II : LANDASAN TEORI**

Bab ini berisi mengenai dasar-dasar teori yang berhubungan dengan Robot Kendali Jarak Jauh Berbasis Gesture tangan, segala macam sensor yang digunakan dan material pendukung pembuatan projek penelitian.

#### **BAB III : ANALISA DAN PERANCANGAN SISTEM**

Bab ini berisi mengenai analisa dan perancangan Robot Mobil Kendali Jarak Jauh Berbasis *Gesture* tangan dengan menggunakan sensor ADXL345.

#### **BAB IV : HASIL DAN PEMBAHASAN**

Bab ini berisi tentang hasil dan pembahasan perancangan system Robot Mobil Kendali Jarak Jauh Berbasis *Gesture* tangan dengan menggunakan sensor ADXL345 yang telah di rancang dan di analisa.

#### **BAB V : KESIMPULAN DAN SARAN**

Bab ini berisikan kesimpulan dan saran dari penelitian yang telah dibuat.

## BAB II

### KAJIAN PUSTAKA

#### 2.1 Pengertian Mikrokontroler

Mikrokontroler adalah sistem mikroprosesor lengkap yang terkandung di dalam sebuah *chip*. Mikrokontroler berbeda dari mikroprosesor serba guna yang digunakan dalam sebuah PC, karena sebuah mikrokontroler umumnya telah berisi komponen pendukung sistem minimal mikroprosesor, yakni memori dan pemrograman *Input-Output* (Martinus, 2013).

Mikrokontroler adalah Suatu chip atau kepingan yang bisa mengontrol suatu proses atau sebuah rangkaian elektronik. Mikrokontroler terdiri dari CPU, ROM, counter-timer, I/O dan unit pendukung seperti ADC. Pada umumnya kontroler ini bisa menyimpan program yang sudah dibuat untuk mengoperasikan suatu alat. Mikrokontroler berada di dalam chip yang bisa digunakan untuk mengontrol sebuah alat elektronik, selain menghemat biaya juga bisa mengurangi jumlah pemakaian komponen elektronik. Mikrokontroler berbeda dengan mikroprosesor, yang merupakan sebuah chip untuk tujuan umum yang digunakan untuk membuat sebuah komputer multi fungsi atau perangkat yang membutuhkan beberapa chip untuk menangani berbagai tugas. Mikrokontroler dimaksudkan untuk menjadi mandiri dan independen, dan berfungsi sebagai komputer khusus yang kecil, Mikrokontroler biasanya digunakan dalam produk dan alat yang dikendalikan secara otomatis, seperti sistem control mesin, remote control, mesin kantor, peralatan rumah tangga dan mainan.

## 2.2 Board Arduino Uno

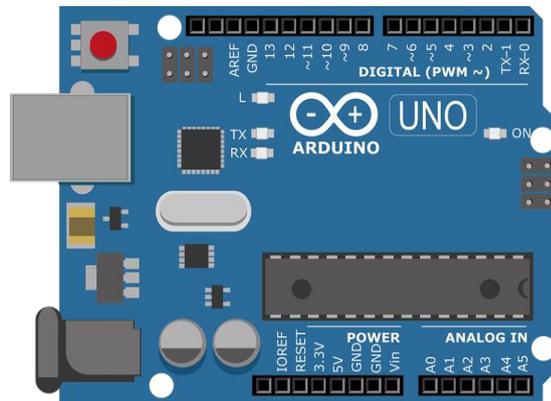
Arduino adalah platform pembuatan prototype elektronik yang bersifat open-source hardware yang berdasarkan pada perangkat keras dan perangkat lunak yang fleksibel dan mudah digunakan. Arduino ditujukan bagi para seniman, desainer, dan siapapun yang tertarik dalam menciptakan objek atau lingkungan yang interaktif (Arifin, 2016).

Arduino Uno sebenarnya adalah salah satu kit mikrokontroler yang berbasis pada ATmega28. Modul ini sudah dilengkapi dengan berbagai hal yang dibutuhkan untuk mendukung mikrokontroler untuk bekerja, tinggal colokkan ke power suply atau sambungkan melalui kabel USB ke PCmu Arduino Uno ini sudah siap sedia.

Arduino Uno memiliki 14 pin digital input/output, 6 analog input, sebuah resonator keramik 16MHz, koneksi USB, colokan power input, ICSP header, dan sebuah tombol reset. Spesifikasi Arduino Uno adalah sebagai berikut :

- Tegangan Input (rekomendasi) 7-12V
- Pin I/O Digital 14 (dengan 6 PWM output)
- Pin Input Analog 6
- Arus DC per Pin I/O 40 mA
- Arus DC per Pin I/O untuk PIN 3.3V 50 mA
- Flash Memory 32 KB (ATmega328) dimana 0.5 KB digunakan oleh bootloader
- SRAM 2 KB (ATmega328)
- EEPROM 1 KB (ATmega328)
- Clock Speed 16 MHz

Deskripsi Arduino Uno tataletak komponen dan konfigurasi pin konektor disajikan pada gambar 2.1 :



Gambar 2.1 Arduino Uno

(Sumber : <https://pixabay.com/en/arduino-arduino-uno-technology-2168193>).



Gambar 2.2 Kabel USB *Board* Arduino Uno

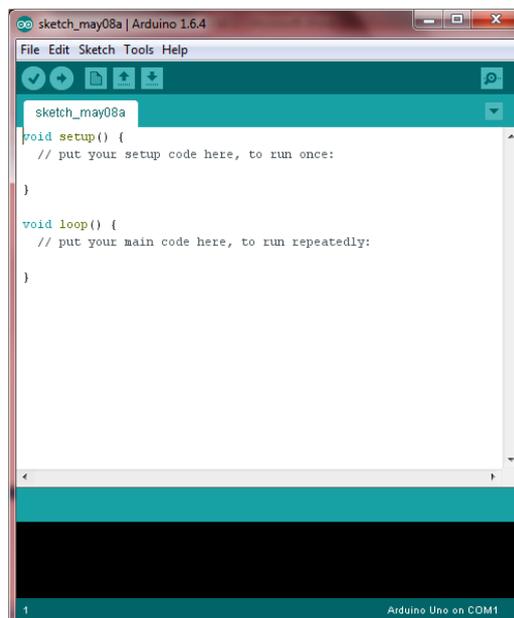
### 2.3 IDE Arduino

Arduino diciptakan untuk para pemula bahkan yang tidak memiliki basic bahasa pemrograman sama sekali karena menggunakan bahasa C++ yang telah dipermudah melalui *library*. Arduino menggunakan Software *Processing* yang digunakan untuk menulis program kedalam Arduino. *Processing* sendiri merupakan penggabungan antara bahasa C++ dan Java.

*Software* Arduino ini dapat di-*install* di berbagai *operating system* (OS) seperti: LINUX, Mac OS, Windows. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi kombinasi dari hardware, bahasa pemrograman dan *Integrated Development Environment* (IDE) yang canggih. IDE adalah sebuah software yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam *memory* microcontroller.

Software IDE Arduino terdiri dari tiga bagian (Arifin, 2016):

- a) Editor program, untuk menulis dan mengedit program dalam bahasa *processing*. *Listing* program pada Arduino disebut *sketch*.
- b) *Compiler*, modul yang berfungsi mengubah bahasa *processing* (kode program) kedalam kode biner karena kode biner adalah satu-satunya bahasa program yang dipahami oleh mikrocontroller.
- c) *Uploader*, modul yang berfungsi memasukkan kode biner kedalam memori mikrocontroller.



Gambar 2.3 Tampilan Software IDE Arduino (Arifin, 2016)

## 2.4 Pengertian Sensor

Sensor adalah alat untuk mendeteksi atau mengukur sesuatu yang digunakan untuk mengubah variasi mekanis, magnetis, panas, sinar dan kimia menjadi tegangan dan arus listrik. Dalam lingkungan sistem pengendali dan robotika, sensor memberikan kesamaan yang menyerupai mata, pendengaran, hidung, lidah yang kemudian akan diolah oleh kontroler sebagai otaknya.

Sensor dalam teknik pengukuran dan pengaturan secara elektronik berfungsi mengubah besaran fisik (misalnya : temperatur, gaya, kecepatan putaran) menjadi besaran listrik yang Proposional. Salah satu sensor yang digunakan dalam pembuatan tugas ini adalah sensor *ultrasonic* (Pratama, 2012).

## 2.5 pengertian Actuators

PEAS merupakan singkatan dari kata *Performance measure, Environment, Actuators, Sensor*. Ketika akan membuat suatu rancangan *agent*, maka sebelumnya harus mengidentifikasi lingkungan masalah atau yang biasa kita sebut dengan “*Task Environment*”.

*Actuators* berisi kemampuan yang dapat *agent* itu lakukan. *Agent* merupakan sesuatu yang dapat mengesan (*perceiving*) lingkungan (*environment*) nya melalui sensors dan bertindak (*acting*) terhadap lingkungan tersebut melalui *actuators*.

## 2.6 Pengertian *Gesture*

*Gesture* adalah gerak atau pose tubuh merupakan suatu bentuk komunikasi non-verbal dengan aksi tubuh yg terlihat mengkomunikasikan pesan tertentu, baik sebagai pengganti wicara atau bersamaan dan paralel dengan kata-kata. *Gesture* mengikutkan pergerakan dari tangan, wajah, atau bagian lain dari tubuh.

Kinesika adalah bidang yang menelaah *gesture*, suatu istilah yang diciptakan seorang perintis studi bahasa nonverbal, *Ray L. Birdwhistell* dalam *Body Talk* (1987). Kinesika adalah pesan non verbal yang dituangkan dalam bentuk bahasa isyarat tubuh atau anggota tubuh. Penggagasan studi mengenai kinesik Ray Birdwhistel, yang menggunakan linguistik sebagai model bagi studi kinesik. *Gesture* tangan merupakan bagian tubuh yang paling sering digunakan untuk mengekspresikan emosi, persahabatan dengan berjabat tangan dan ketidaksenangan dengan menyentuh bagian tertentu dari tubuh sendiri.

## 2.7 Bahasa Pemrograman C

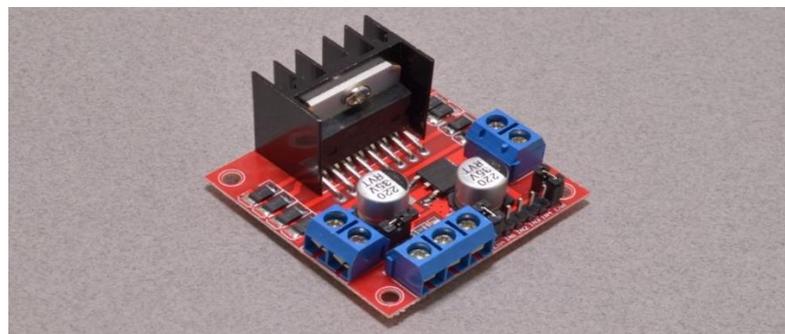
Bahasa pemrograman C adalah sebuah bahasa pemrograman komputer yang bisa digunakan untuk membuat berbagai aplikasi (*general-purpose programming language*), mulai dari sistem operasi (seperti Windows atau Linux), antivirus, software pengolah gambar (*image processing*), hingga *compiler* untuk bahasa pemrograman, dimana C banyak digunakan untuk membuat bahasa pemrograman lain yang salah satunya adalah PHP (Andre, 2017).

Meskipun termasuk *general-purpose programming*, yakni bahasa pemrograman yang bisa membuat berbagai aplikasi, bahasa pemrograman C paling

cocok merancang aplikasi yang berhubungan langsung dengan Sistem Operasi dan Hardware.

## 2.8 Driver Motor (Motor Shield L298)

Driver Motor L298 adalah komponen elektronik yang dipergunakan untuk mengontrol arah putaran motor DC. Satu buah L298 bisa dipergunakan untuk mengontrol dua buah motor DC. Selain bisa dipergunakan untuk mengontrol arah putaran motor DC, L298 ini pun bisa dipergunakan sebagai driver motor Stepper bipolar. IC driver L298 memiliki kemampuan menggerakkan motor DC sampai arus 2A dan tegangan maksimum 40 volt DC untuk satu kanalnya. Pin enable A dan B untuk mengendalikan jalan atau kecepatan motor, pin input 1 sampai 4 digunakan untuk mengendalikan arah putaran. Pin output pada IC L298 13 dihubungkan ke motor DC yang sebelumnya melalui dioda yang disusun secara H-bridge. Pengaturan kecepatan motor digunakan teknik PWM (*pulse width modulation*) yang diinputkan dari mikrokontroler melalui pin *Enable*. PWM untuk kecepatan rotasi yang bervariasi level highnya (Adriansyah, 2013).



Gambar 2.4 Motor driver L298N

(Sumber : <https://www.bc-robotics.com/shop/l298n-motor-driver-board/>)

## 2.9 Motor DC

Motor DC merupakan sebuah perangkat elektromagnetis yang mengubah energi listrik menjadi energi mekanik. Energi mekanik ini digunakan untuk, misalnya, memutar impeller pompa, fan atau blower, menggerakkan kompresor, mengangkat bahan, dll. Motor listrik digunakan juga di rumah (mixer, bor listrik, fan angin) dan di industri. Motor listrik kadangkala disebut “kuda kerja” nya industri sebab diperkirakan bahwa motormotor menggunakan sekitar 70% beban listrik total di industri. Keuntungan utama motor DC adalah sebagai pengendali kecepatan, yang tidak mempengaruhi kualitas pasokan daya. Motor ini dapat dikendalikan dengan mengatur :

1. Tegangan dinamo – meningkatkan tegangan dinamo akan meningkatkan kecepatan.
2. Arus medan – menurunkan arus medan akan meningkatkan kecepatan.

Motor DC tersedia dalam banyak ukuran, namun penggunaannya pada umumnya dibatasi untuk beberapa penggunaan berkecepatan rendah, penggunaan daya rendah hingga sedang seperti peralatan mesin dan rolling mills, sebab sering terjadi masalah dengan perubahan arah arus listrik mekanis pada ukuran yang lebih besar. Juga, motor tersebut dibatasi hanya untuk penggunaan di area yang bersih dan tidak berbahaya sebab resiko percikan api pada sikatnya. Motor DC juga relatif mahal dibanding motor AC (Budi, 2016).

### **Accelerometer ADXL345**

Accelerometer ADXL345 merupakan salah satu sensor accelerometer yang mampu menangkap respon berupa kemiringan dan juga getaran. Accelerometer ADXL345 adalah modul sensor gerak/akselerasi dengan 3 sumbu (triple axis 8194 acceleration sensor module) yang memiliki resolusi 13-bit (2 tingkatan presisi) yang dapat mendeteksi hingga jangkauan 16g ( $16 \times 9,81 \text{ m/s}^2$ ) (Adisty, 2004). Aplikasinya mencakup deteksi kemiringan dengan melihat perubahan gaya statik (static gravity acceleration on tilt sensing application) dan percepatan dinamik (dynamic acceleration) yang timbul akibat gerakan atau tumbukan. Dengan resolusi tinggi yang dihasilkan (3,9mg/LSB high resolution) yang memungkinkan modul sensor akselerometer ini mendeteksi pergerakan dan inklinasi secara halus. Sensor accelerometer ini cocok digunakan pada aplikasi portable dan sangat cocok untuk digunakan pada rangkaian mikrokontroler semacam papan pengembang seperti arduino dan mikrokontroler AVR karena akses data yang mudah lewat antarmuka SPI atau I2C. Data direpresentasikan secara digital dalam format integer 16-bit.

Modul sensor ini dapat mendeteksi status aktivitas gerakan (active/inactive) dengan membandingkan percepatan/akselerasi pada sumbu manapun dengan ambang batas sensitivitas yang dapat disesuaikan lewat kode program. Pada akselerometer ADXL345 juga terdapat pendeteksi ketukan (tap sensing) yang dapat mendeteksi ketukan tunggal maupun ganda pada berbagai arah.

Modul sensor akselerometer ADXL345 ini juga dapat mendeteksi gerak jatuh bebas (free fall sensing), fungsi-fungsi tersebut dapat dipetakan secara terpisah pada dua pin interupsi keluaran (interrupt output pins). Modul sensor

akselerometer ADXL345 ini memiliki sistem pengelolaan memori internal 32-bit bertipe antrian FIFO (First In First Out) yang dapat digunakan untuk menyimpan variabel/data temporer hasil pengukuran sehingga mengurangi beban mikrokontroler sehingga menurunkan konsumsi energi pada sistem. Modul sensor akselerometer ADXL345

ini memiliki sirkuit pengelolaan daya yang baik dimana modul ditempatkan pada moda konsumsi daya yang sangat kecil hingga terdeteksi gerakan yang melewati ambang batas (threshold) tertentu yang mengaktifkan kembali moda normal. Sehingga pembacaan sensor selesai secara otomatis modul dikembalikan ke moda siaga untuk menghemat energi. Beberapa fitur yang dimiliki oleh ADXL345 triple axis ini ialah sebagai berikut:

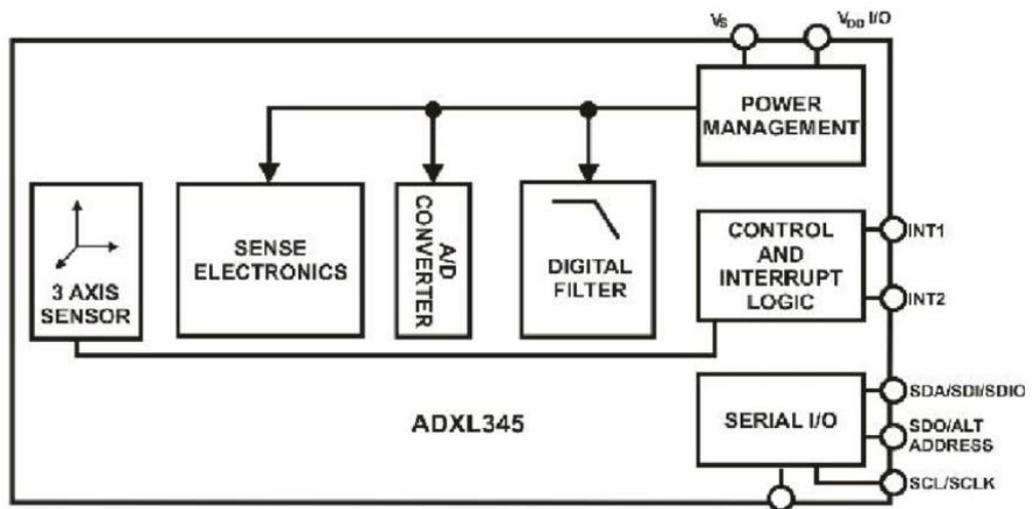
1. Menggunakan chip ADXL345 yang diproduksi oleh Analog Devices Inc.
2. Tipe data keluaran sudah berupa digital
3. Komunikasi data dapat menggunakan I2C atau SPI
4. Jangkauan deteksi dari +2g hingga +16g
5. Catu daya 2 volt - 3,6 volt (kompatibel dengan raspberry-Pi, untuk arduino pasokan daya dapat diambil dari pin 3v3)
6. Pin antarmuka toleran terhadap tegangan 5V (dapat dikoneksikan langsung dengan I/O TTL 5V)
7. Suhu operasional -40° sampai 85° C
8. Konsumsi arus rendah yaitu kurang dari 25 mA pada saat siaga
9. Ukuran modul sensor 27,8 x 16,9 x 11 mm

Berikut adalah gambar penampakan dari Accelerometer ADXL345

triple axis:

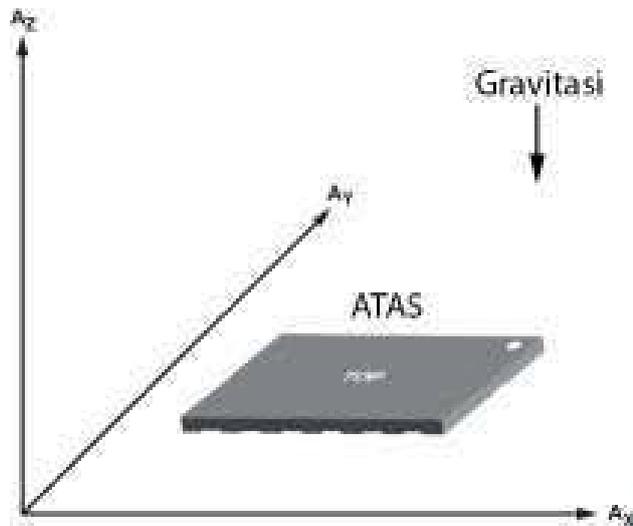


Gambar 2.5 Accelerometer ADXL345 Triple Axis

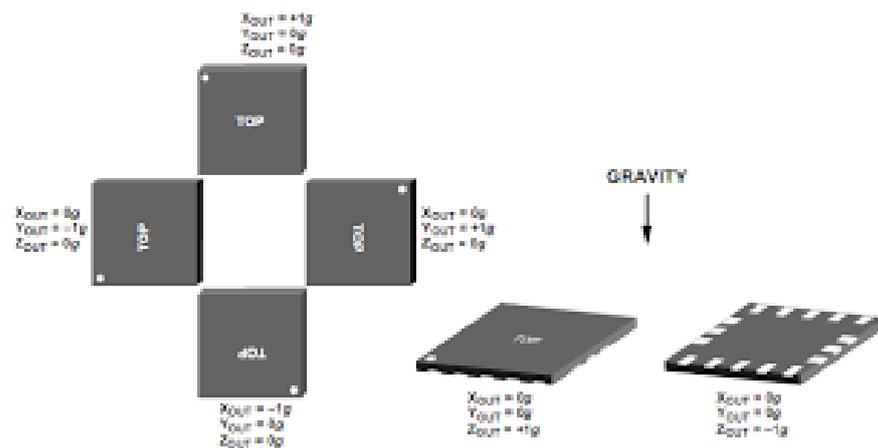


Gambar 2.6 Diagram Blok ADXL345

Dari diagram blok ADXL345 diketahui bahwa dalam accelerometer ADXL345 sudah terdapat ADC dan digital filter sehingga ADXL345 merupakan sensor percepatan yang menggunakan antarmuka digital yaitu dengan komunikasi I2C atau SPI. Dalam penelitian ini menggunakan antarmuka I2C sebagai sarana komunikasinya. Berikut ini akan dijelaskan mengenai sistematika kerja ADXL345, termasuk diantaranya orientasiterhadap sumbu kartesian, konfigurasi pin, serta karakteristik dan speksifikasi ADXL345.



Gambar 2.7 Sumbu pada ADXL345

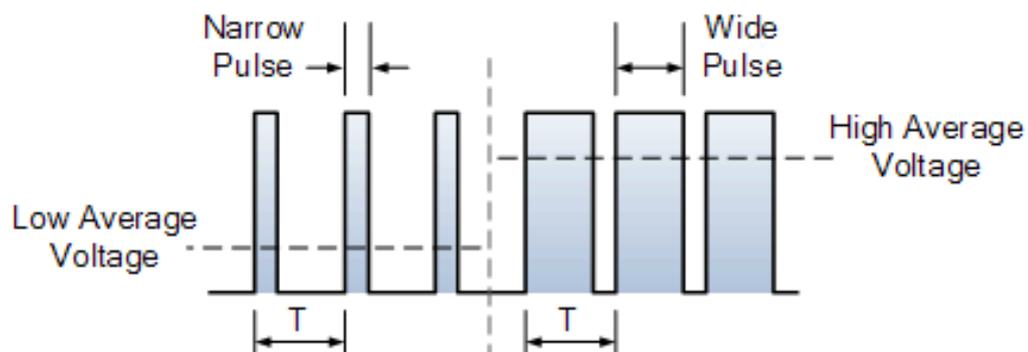


Gambar 2.8 Respon Output Dan Orientasi Terhadap Gravitasi Sensor Accelero.

Dari kedua gambar di atas, kita dapat mengetahui orientasi dan juga karakteristik output ADXL345 terhadap ketiga sumbu (x, y dan z) sehingga dapat dijadikan acuan pada penelitian ini. Saat posisi salah satu sumbu bertolak belakang dengan arah gaya gravitasi, maka output sumbu itu akan sekitar  $\pm 1g$  atau setara  $\pm 9,8$  m/s<sup>2</sup>.

## 2.10 PWM (Pulse Width Modulation)

Pulse Width Modulation (PWM) secara umum adalah sebuah cara memanipulasi lebar sinyal yang dinyatakan dengan pulsa dalam suatu periode, untuk mendapatkan tegangan rata-rata yang berbeda. Aplikasi PWM berbasis mikrokontroler biasanya berupa, pengendalian kecepatan motor DC, Pengendalian Motor Servo, Pengaturan nyala terang LED. Beberapa Contoh aplikasi PWM adalah pemodulasian data untuk telekomunikasi, pengontrolan daya atau tegangan yang masuk ke beban, regulator tegangan, audio effect dan penguatan, serta aplikasi-aplikasi lainnya (Prayogo, 2012).



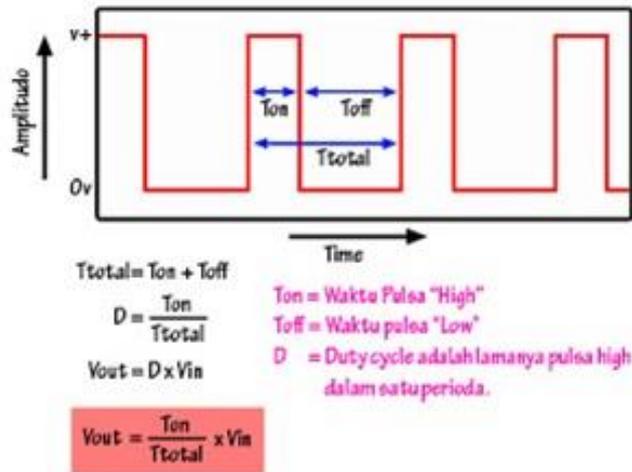
Gambar 2.9 Sinyal PWM

(Sumber : <https://www.electronics-tutorials.ws/articles/pwm2.gif>)

### 2.11.1 Konsep Dasar PWM

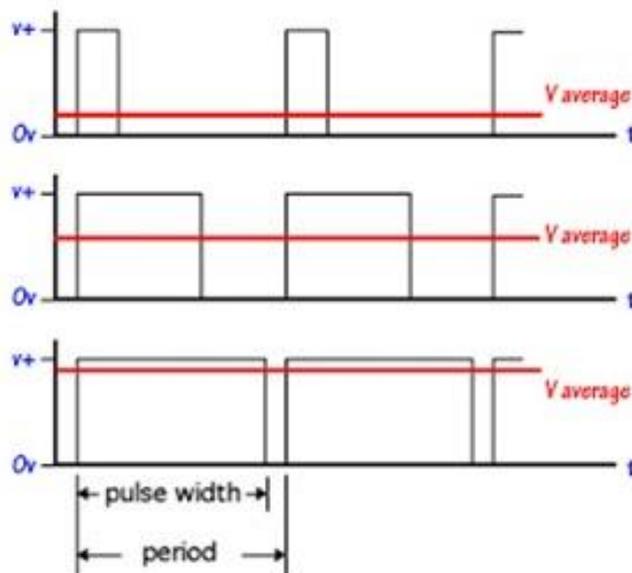
Sinyal PWM pada umumnya memiliki amplitude dan frekuensi dasar yang tetap, namun memiliki lebar pulsa yang bervariasi. Lebar pulsa PWM berbanding lurus dengan amplitude sinyal asli yang belum termodulasi. Artinya, sinyal PWM

memiliki frekuensi gelombang yang tetap namun duty cycle bervariasi antara 0% hingga 100%.



Gambar 2.10 Bentuk Gelombang Kotak Pulsa

Dari persamaan diatas, diketahui bahwa perubahan duty cycle akan merubah tegangan output atau tegangan rata-rata seperti gambar dibawah ini.

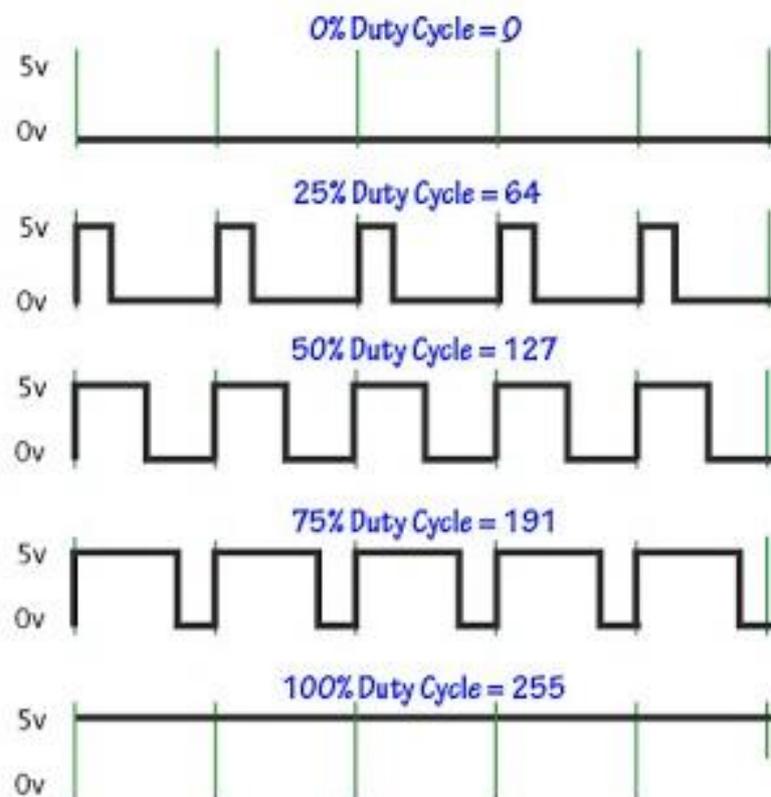


Gambar 2.11 Perubahan Duty Cycle

PWM merupakan salah satu teknik untuk mendapatkan sinyal analog dari sebuah piranti digital. Sebenarnya sinyal PWM dapat dibangkitkan dengan banyak cara, secara analog menggunakan IC op-amp atau secara digital.

Secara analog setiap perubahan PWM-nya sangat halus, sedangkan secara digital setiap perubahan PWM dipengaruhi oleh resolusi PWM itu sendiri. Resolusi adalah jumlah variasi perubahan nilai dalam PWM tersebut. Misalkan suatu PWM memiliki resolusi 8 bit, berarti PWM ini memiliki variasi perubahan nilai sebanyak

256 variasi mulai dari 0 – 255 perubahan nilai yang mewakili duty cycle 0% – 100% dari keluaran PWM tersebut.



Gambar 2.12 Variasi Perubahan PWM

### **2.11 Sensor Ultrasonik (HCSR-04)**

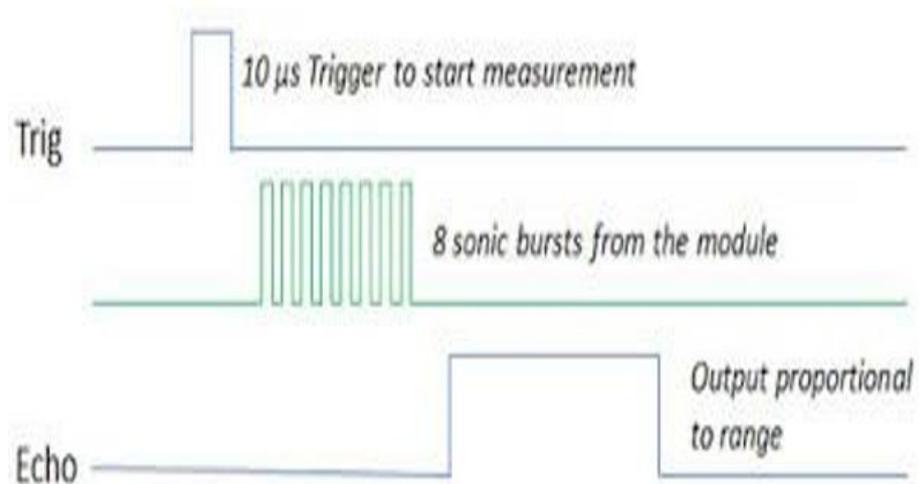
Sensor ultrasonik adalah sebuah sensor yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya. Cara kerja sensor ini didasarkan pada prinsip dari pantulan suatu gelombang suara sehingga dapat dipakai untuk menafsirkan eksistensi (jarak) suatu benda dengan frekuensi tertentu. Disebut sebagai sensor ultrasonik karena sensor ini menggunakan gelombang ultrasonik (bunyi ultrasonik). Sensor ini merupakan sensor ultrasonik siap pakai, satu alat yang berfungsi sebagai pengirim, penerima, dan pengontrol gelombang ultrasonik. Alat ini bisa digunakan untuk mengukur jarak benda dari 2cm - 4m dengan akurasi 3mm. Alat ini memiliki 4 pin, pin Vcc, Gnd, Trigger, dan Echo. Pin Vcc untuk listrik positif dan Gnd untuk ground-nya. Pin Trigger untuk trigger keluarnya sinyal dari sensor dan pin Echo untuk menangkap sinyal pantul dari benda.

Gelombang ultrasonik adalah gelombang bunyi yang mempunyai frekuensi sangat tinggi yaitu 20.000 Hz. Bunyi ultrasonik tidak dapat di dengar oleh telinga manusia. Bunyi ultrasonik dapat didengar oleh anjing, kucing, kelelawar, dan lumba-lumba. Bunyi ultrasonik bisa merambat melalui zat padat, cair dan gas. Reflektivitas bunyi ultrasonik di permukaan zat padat hampir sama dengan reflektivitas bunyi ultrasonik di permukaan zat cair. Akan tetapi, gelombang bunyi ultrasonik akan diserap oleh tekstil dan busa.



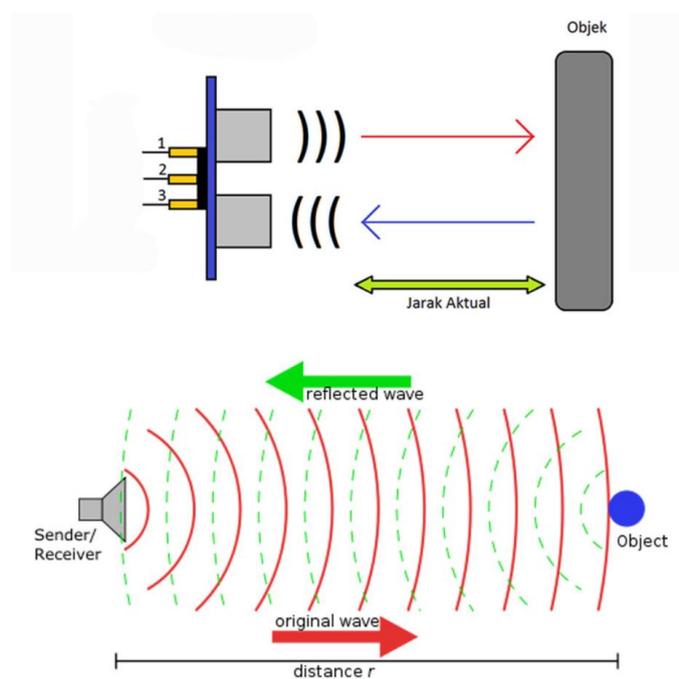
Gambar 2.13 Sensor Ultrasonic (HC-SR04)

Cara menggunakan alat ini yaitu: ketika kita memberikan tegangan positif pada pin Trigger selama 10 $\mu$ S, maka sensor akan mengirimkan 8 step sinyal ultrasonik dengan frekuensi 40kHz. Selanjutnya, sinyal akan diterima pada pin Echo. Untuk mengukur jarak benda yang memantulkan sinyal tersebut, maka selisih waktu ketika mengirim dan menerima sinyal digunakan untuk menentukan jarak benda tersebut. Berikut adalah visualisasi dari sinyal yang dikirimkan oleh sensor HC-SR04.



Gambar 2.14 sistem pewaktu pada sensor HC-SR04

Pada sensor ultrasonik, gelombang ultrasonik dibangkitkan melalui sebuah alat yang disebut dengan piezoelektrik dengan frekuensi tertentu. Piezoelektrik ini akan menghasilkan gelombang ultrasonik (umumnya berfrekuensi 40kHz) ketika sebuah osilator diterapkan pada benda tersebut. Secara umum, alat ini akan menembakkan gelombang ultrasonik menuju suatu area atau suatu target. Setelah gelombang menyentuh permukaan target, maka target akan memantulkan kembali gelombang tersebut. Gelombang pantulan dari target akan ditangkap oleh sensor, kemudian sensor menghitung selisih antara waktu pengiriman gelombang dan waktu gelombang pantul diterima. Gambar cara kerja sensor ultrasonik dengan transmitter dan receiver (atas), sensor ultrasonik dengan single sensor yang berfungsi sebagai transmitter dan receiver sekaligus oleh transmitter dan waktu ketika gelombang pantul diterima receiver.



Gambar 2.15 Sensor HCSR-04

### 2.11.1 Prinsip Kerja Sensor HCSR-04

Secara detail, cara kerja sensor ultrasonik adalah sebagai berikut:

- Sinyal dipancarkan oleh pemancar ultrasonik dengan frekuensi tertentu dan dengan durasi waktu tertentu. Sinyal tersebut berfrekuensi diatas 20kHz. Untuk mengukur jarak benda (sensor jarak), frekuensi yang umum digunakan adalah 40kHz.
- Sinyal yang dipancarkan akan merambat sebagai gelombang bunyi dengan kecepatan sekitar 340 m/s. Ketika menumbuk suatu benda, maka sinyal tersebut akan dipantulkan oleh benda tersebut.
- Setelah gelombang pantulan sampai di alat penerima, maka sinyal tersebut akan diproses untuk menghitung jarak benda tersebut. Jarak benda dihitung berdasarkan rumus :

$$S = 340.t/2$$

Dimana S merupakan jarak antara sensor ultrasonik dengan benda (bidang pantul), dan t adalah selisih antara waktu pemancaran gelombang.

### 2.13 Modul RF 433Mhz

Modul RF 433Mhz terdiri dari 2 rangkaian yaitu rangkaian transmitter (TX) dan rangkaian receiver (RX). Modul RF 433 Mhz ini menggunakan protokol one wire untuk berkomunikasi dengan microcontroller. RF433Mhz merupakan rangkaian pengirim dan penerima data yang berbasis ASK (Aplitude – Shift Keying). Modul tersebut digunakan pada alat penghitung waktu balapan drag sebagai pengganti kabel yang menghubungkan rangkaian pada garis start dengan

rangkaian pada garis finish dimana sistem tersebut dinamakan system wireless (Syahrin, 2016).



Gambar 2.16 Tampilan RF 433Mhz

## **BAB III**

### **ANALISA DAN PERANCANGAN SISTEM**

#### **3.1. ANALISA KEBUTUHAN SISTEM ROBOT**

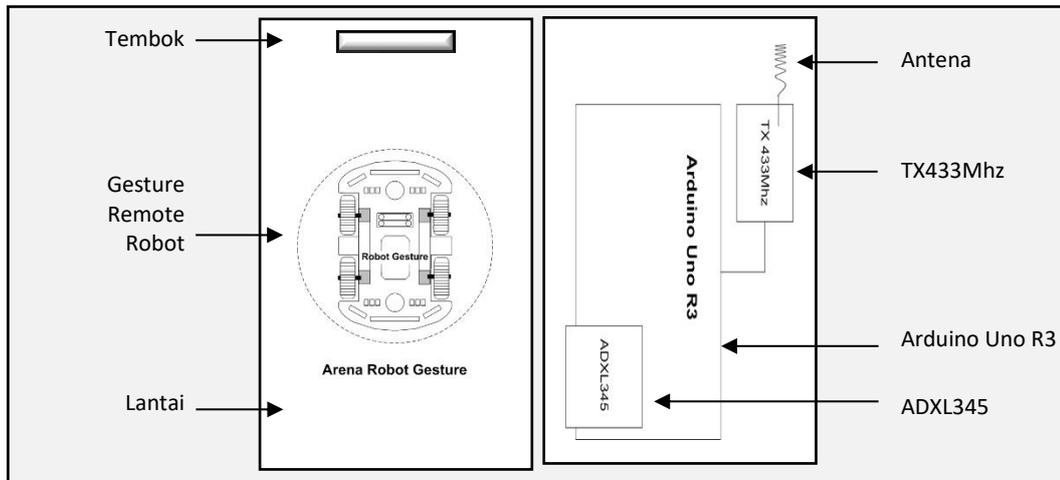
Pada bab ini membahas tentang analisis kebutuhan dan perancangan sistem yang akan dibangun. Analisis sistem ini bertujuan untuk menguraikan segala kebutuhan atau fitur-fitur yang diperlukan, menjelaskan bagaimana sistem bekerja, serta desain rangkaian robot yang akan dibangun agar terdefinisi secara jelas.

##### **3.1.1. Misi Sistem Robot**

Misi sistem Robot Mobil Kendali Jarak Jauh Berbasis *Gesture* Dengan Menggunakan Sensor Adxl345 adalah robot mobil mampu mengikuti arah instruksi dari gesture tangan yang mengendalikan robot mobil tersebut, seperti arah maju, mundur, depan, dan belakang.

##### **3.1.2. Lingkungan Sistem Robot**

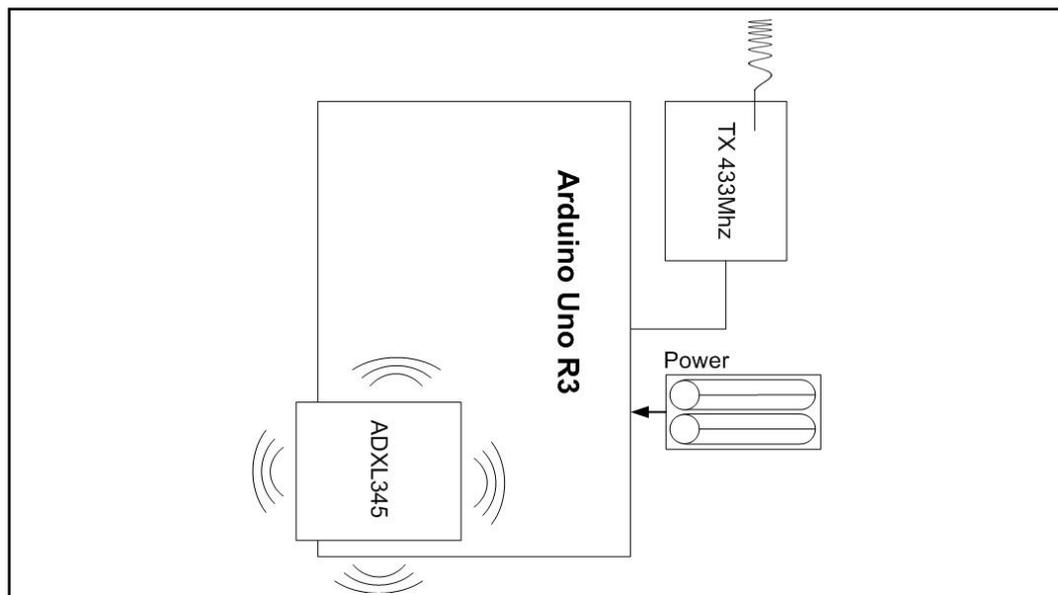
Lingkungan (*workspace*) sistem robot adalah tempat dimana robot akan melakukan tugas atau misinya. Dalam penelitian ini batasan lingkungan sistem robot mobil yang dibangun adalah suatu arena bidang datar, lantai atau jalanan yang rata.



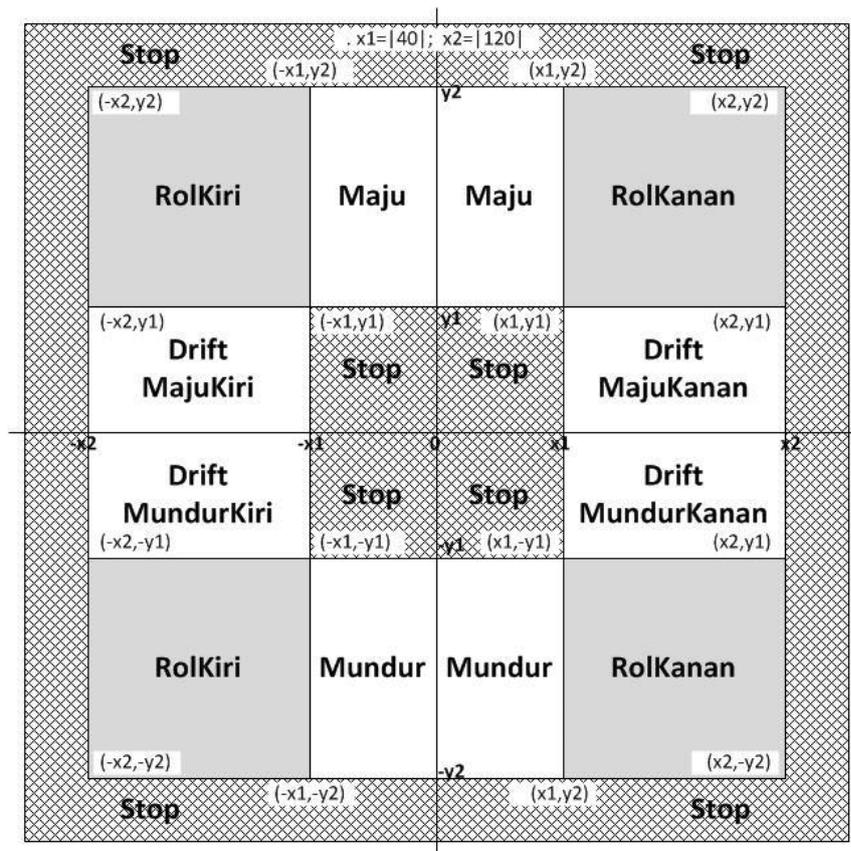
Gambar 3.1. Gambar lingkungan robot *gesture*

### 3.1.3. Kebutuhan Sistem Sensor

**Sensor ADXL:** adalah system sensor yang mampu mendeteksi arah *acceleration*. Sensor ini digunakan untuk menentukan arah gerakan robot mobil, sensor ini diletakan di tangan. Kemudian melalui modul transmitter dikirimkan sinyal arah ke receiver melalui gelombang radio.

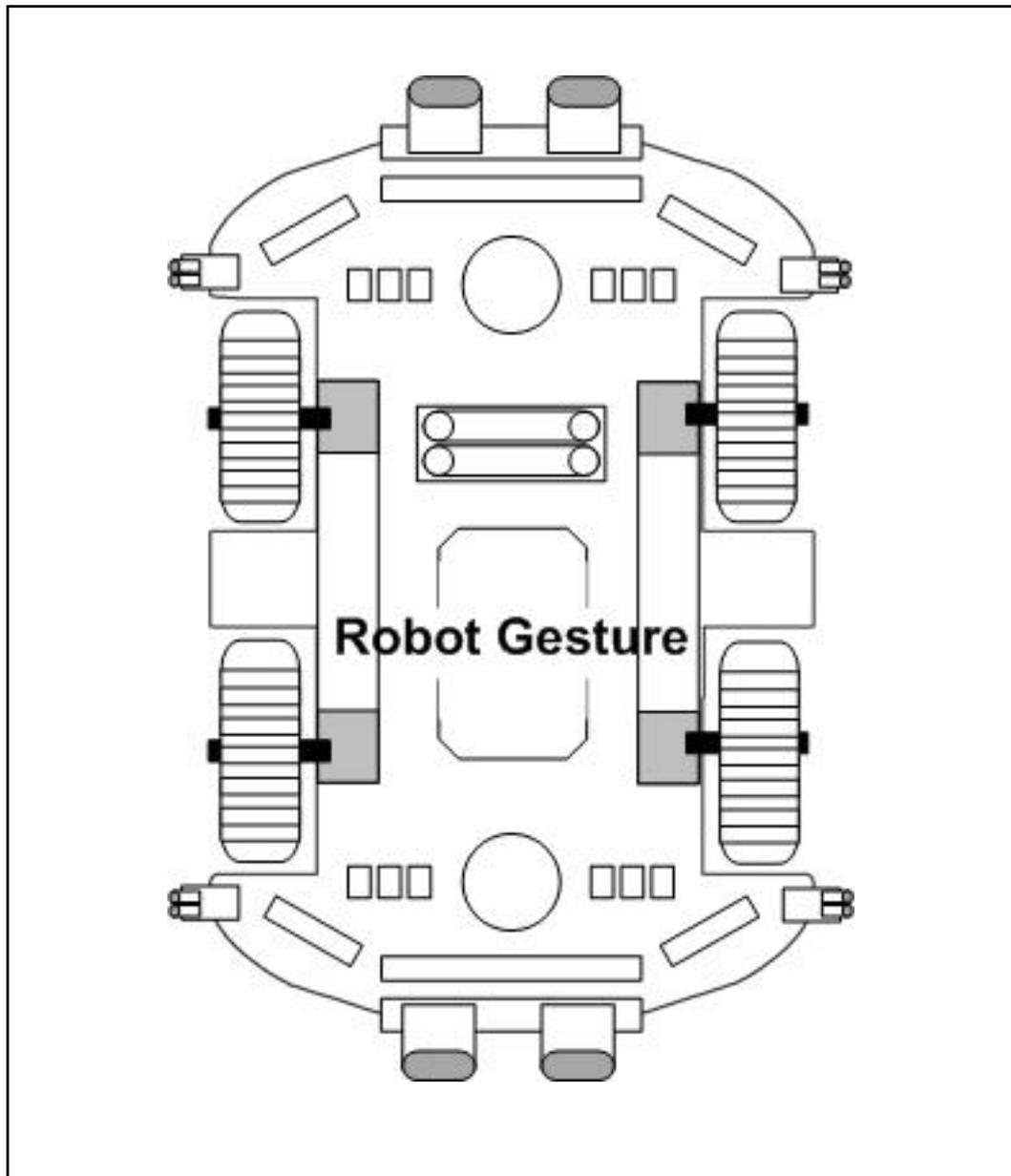


Gambar 3.2. Rancangan formasi sensor ADXL.



Gambar 3.3. Rancangan Pengolahan Data *Acceleration*.

**Sensor Ultrasonic:** sensor yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya. Cara kerja sensor ini didasarkan pada prinsip dari pantulan suatu gelombang suara sehingga dapat dipakai untuk menafsirkan eksistensi (jarak) suatu benda dengan frekuensi tertentu. Rancangan sensor ultrasonic dapat dilihat pada gambar berikut.



Gambar 3.4. Rancangan formasi sensor Ultrasonic.

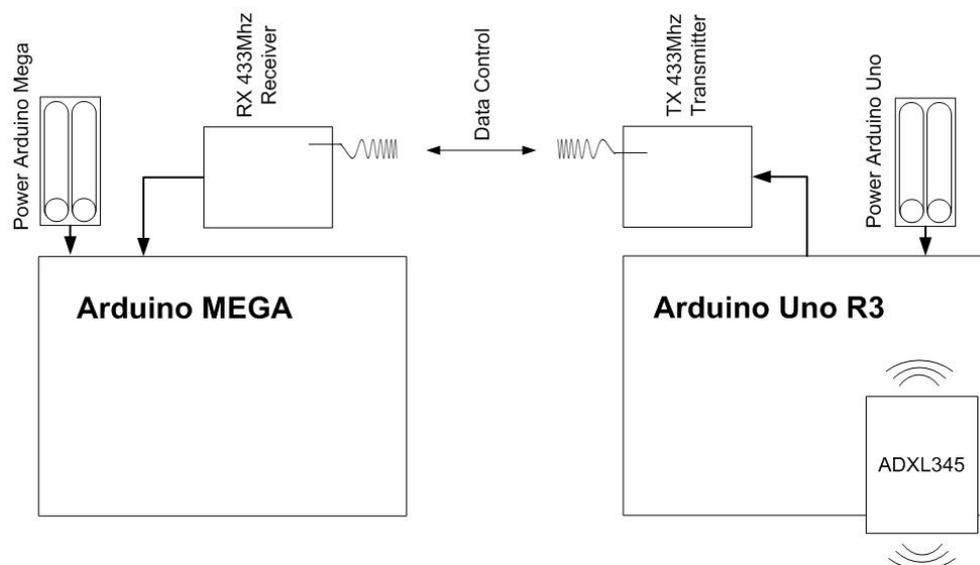
#### 3.1.4. Kebutuhan Sistem Aktuator dan Lokomosi

Sistem aktuator merupakan inti sistem gerak robot. Sistem actuator pada penelitian robot ini ialah menggunakan motor DC. Sistem aktuator menggerakkan sistem lokomosi berbasis roda.

Sistem lokomosi memakai dua aktuator yaitu motor kanan dan kiri yang masing-masing memiliki kontrol gerakan otonom. Gerakan-gerakan robot yang terdiri dari: maju, mundur, berbelok (kiri dan kanan) serta berputar, instruksi arah tersebut dapat disusun dari kombinasi gerakan aktuator kiri dan aktuator kanan.

### 3.1.5 Komunikasi RF433Mhz

Data *acceleration* yang diterima oleh Arduino Uno R3 dari sensor ADXL diproses oleh arduino sehingga bisa menentukan arah *Gesture Remote Robot* untuk bergerak, setelah data diproses maka data tersebut di transmisikan melalui Modul Transmitter RF433Mhz. Data yang dikirim oleh transmitter berupa bilangan sehingga tidak terlalu banyak data yang dikirim. Data yang dikirimkan oleh transmitter akan diterima oleh Receiver RF433Mhz, data tersebut diproses sehingga robot bisa bergerak. Komunikasi RF433Mhz dapat dilihat pada gambar berikut.



Gambar 3.5. Rancangan Komunikasi RF433Mhz.

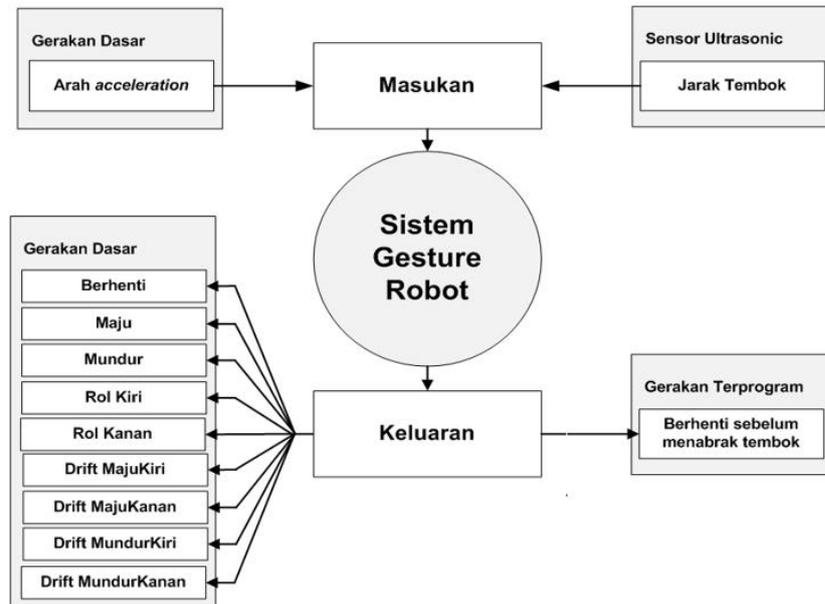
### **3.1.6. Kebutuhan Perangkat Lunak**

Kebutuhan perangkat lunak adalah kebutuhan system pengendalian robot, baik dalam algoritma mentransmisikan data, menerima data serta mengolah data maupun mengeluarkan data kendali actuator. Kebutuhan fungsionalitas sistem robot gesture ini adalah sebagai berikut:

(1). Kebutuhan untuk menerima data sensor ADXL serta mengirimkan data tersebut ke receiver melalui gelombang radio. (2). Kebutuhan untuk menerima data dari transmitter untuk diolah sehingga dapat memberikan output berupa aksi yang dilakuakn oleh robot. (3). Kebutuhan untuk menerjemahkan hasil yang diproses oleh receiver untuk di proses oleh mesin sehingga robot gesture bisa bergerak.

### **3.2. PERANCANGAN UMUM SISTEM ROBOT**

Perangkat lunak dirancang secara modular, pembagian ini bertujuan untuk mempermudah cara pandang terhadap sistem dan menyederhanakan implementasinya. Skema umum sistem robot gesture dapat dilihat pada gambar berikut :

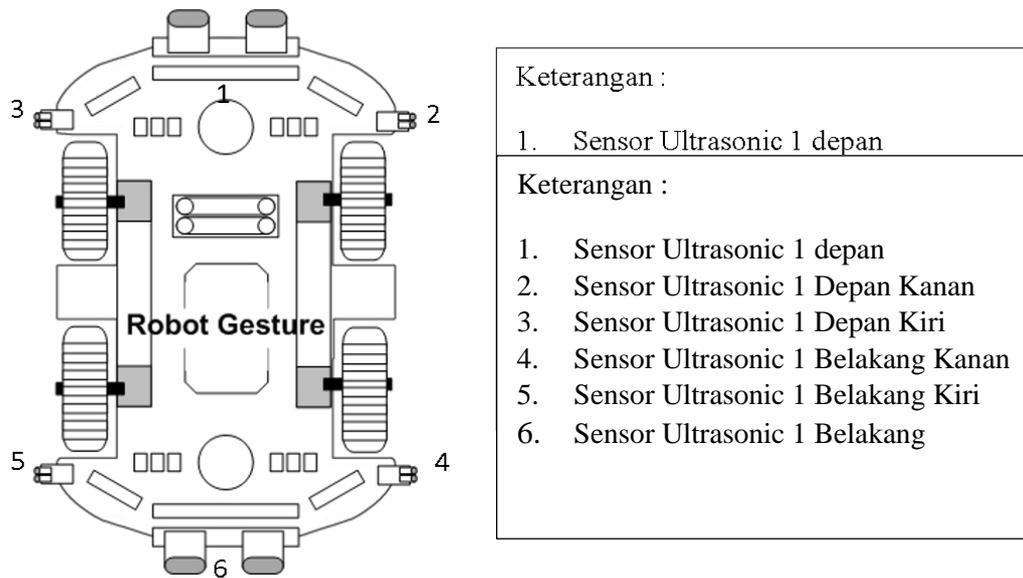


Gambar 3.6. Skema umum perangkat lunak robot gesture

### 3.3 RANCANGAN DATA

Setiap sensor yang digunakan pada pembuatan robot gesture ini memiliki daerah pantauan dan variabel/buffer memory yang mencatat statusnya sendiri-sendiri. Sistem akan mencatat data yang diterima dari sensor ADXL yang membuat sistem dapat mengerti sehingga dapat menerjemahkan perintah tersebut menjadi sebuah aksi bergerak.

Sensor Ultrasonic akan mencatat jarak tembok/penghalang dengan robot jika penghalang semakin dekat maka robot akan diperintahkan untuk berhenti. Konfigurasi jarak penghalang dengan robot dapat di ubah sesuai dengan kebutuhan sistem robot.

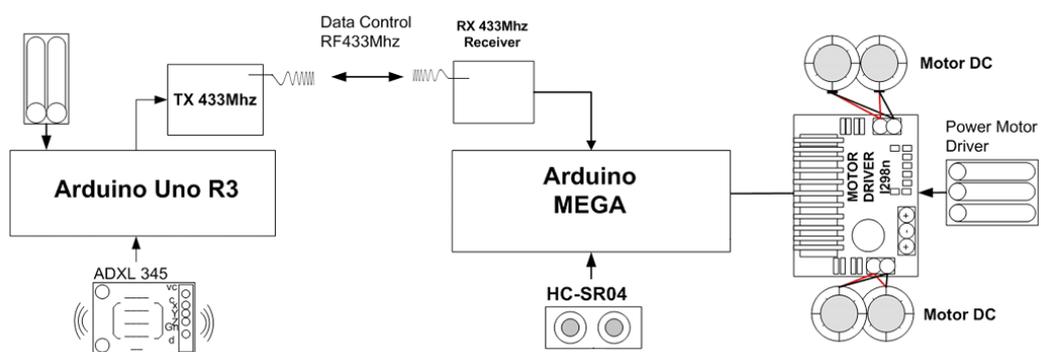


Gambar 3.7 Peletakan Sensor Ultrasonic pada robot

### 3.4 PERACANGAN PERANGKAT KERAS ROBOT

Perangkat keras pada robot gesture terdiri dari 2 bagian yaitu perangkat keras untuk *remote gesture robot*, dan perangkat keras *remote gesture controller*.

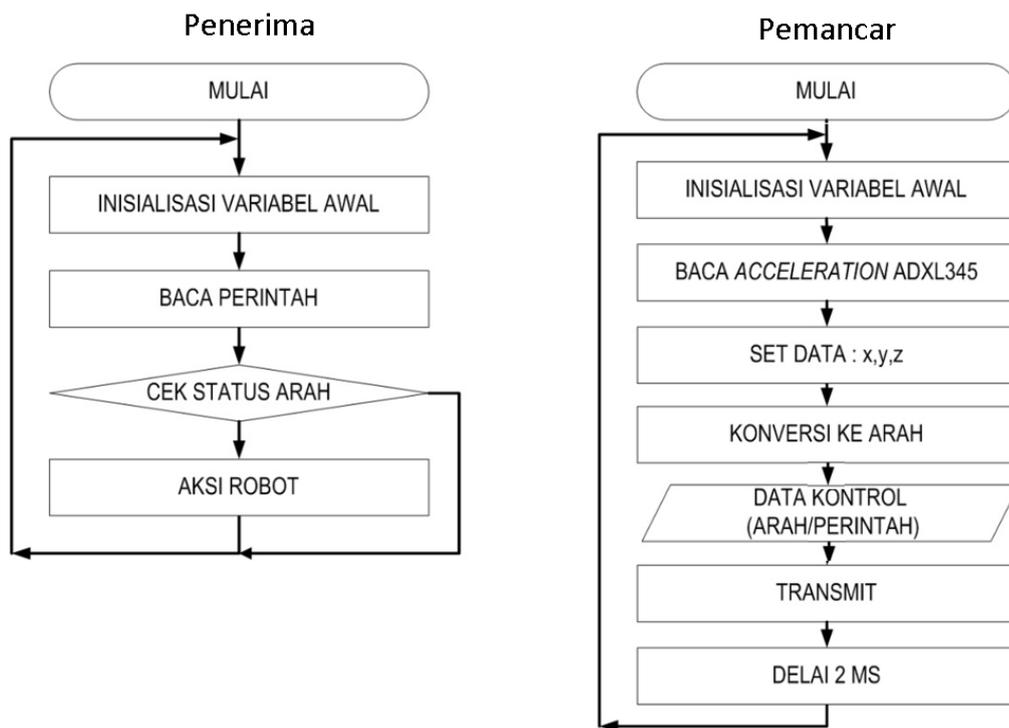
Rangkaian perangkat keras robot dapat dilihat pada gambar berikut.



Gambar 3.8 Rangkaian Perangkat Keras *Receiver* dan Pengolah Data

### 3.5 PERACANGAN PERANGKAT LUNAK ROBOT

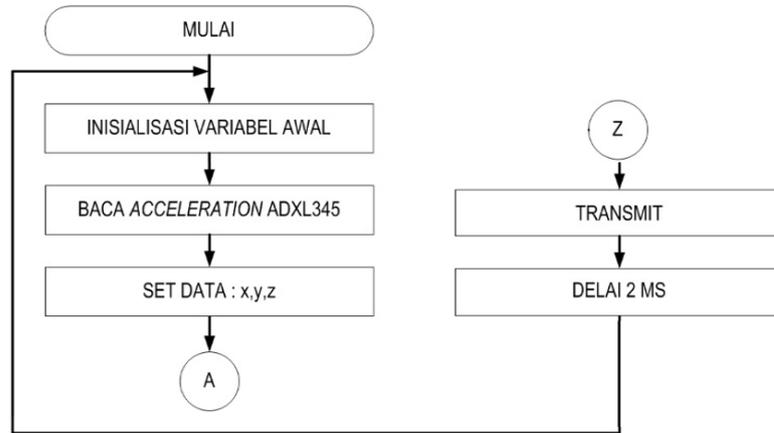
Perancangan perangkat lunak robot yang dibangun dapat dilihat pada penjelasan diagram alir berikut ini.



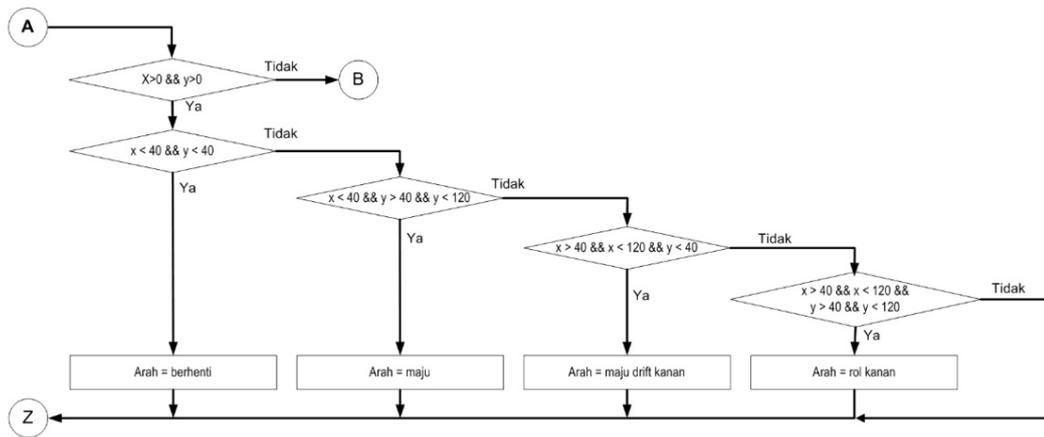
Gambar 3.9 Diagram Alir Program secara Keseluruhan

#### 3.5.1. Algoritma Proses Pengolahan Data Accelerometer

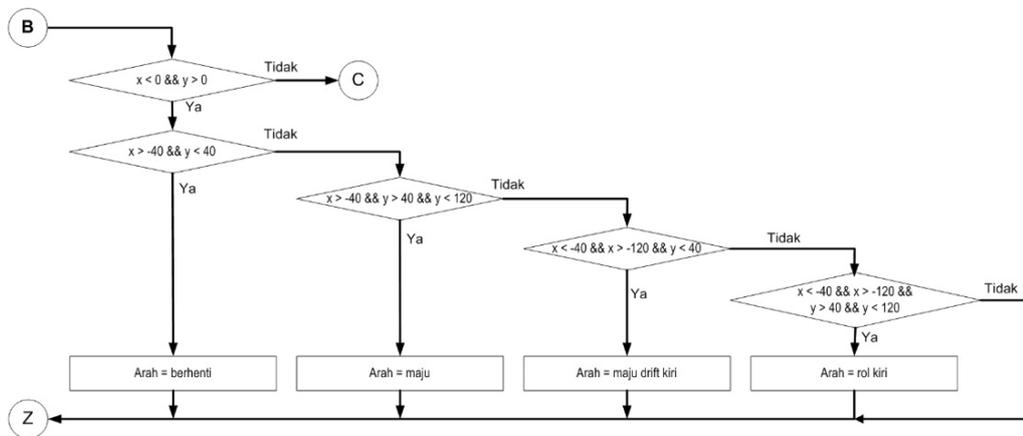
Pada proses pengolahan data accelerometer yang diterima dari ADXL345 berfungsi untuk menentukan arah gerakan robot dari data yang diterima dari sensor. Data yang kirim akan diberikan kode (NODE ID) supaya data yang diterima benar dari ADXL yang digunakan. Untuk penjelasan lebih lanjut proses pengolahan data *acceleration* dapat dilihat pada diagram alir berikut.



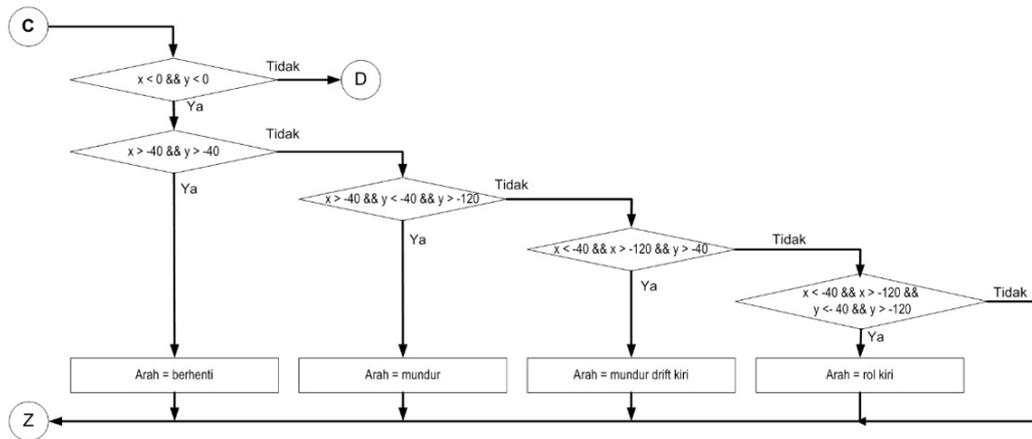
Gambar 3.10 Diagram Alir Proses Baca acceleration 1



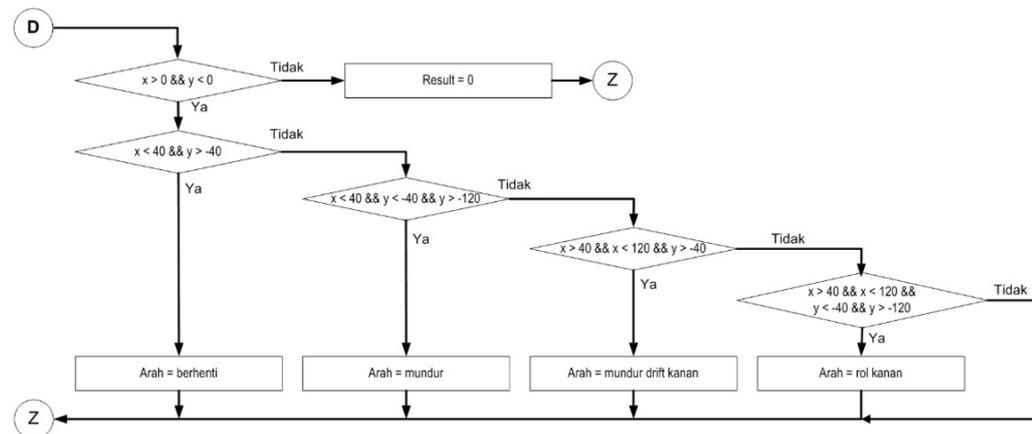
Gambar 3.11 Diagram Alir Proses Baca acceleration 2



Gambar 3.12 Diagram Alir Proses Baca acceleration 3



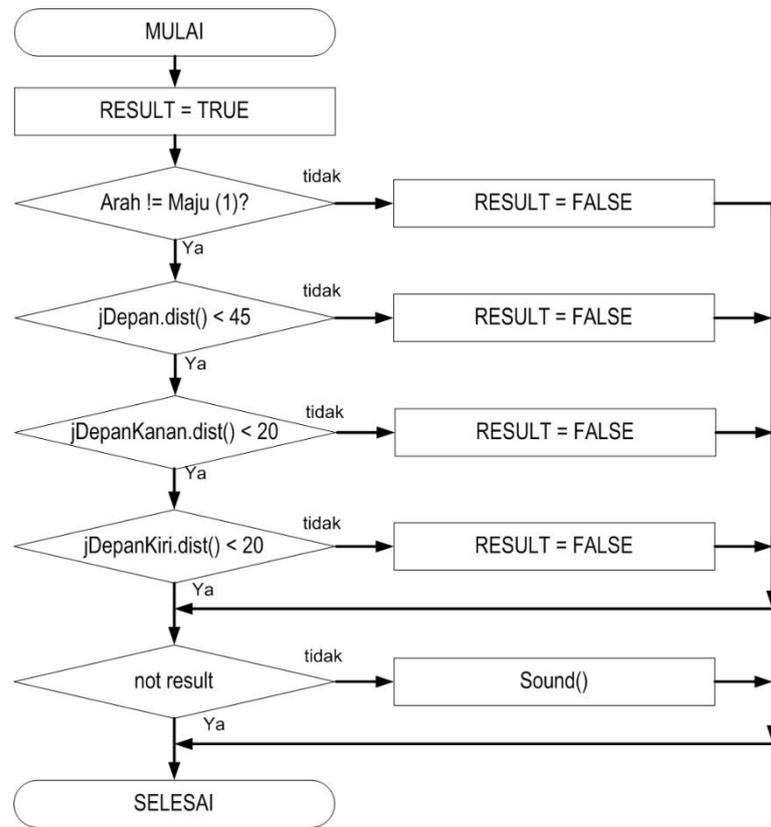
Gambar 3.13 Diagram Alir Proses Baca acceleration 4



Gambar 3.14 Diagram Alir Proses Baca acceleration 5

### 3.5.2. Algoritma Cek Status Perintah Maju

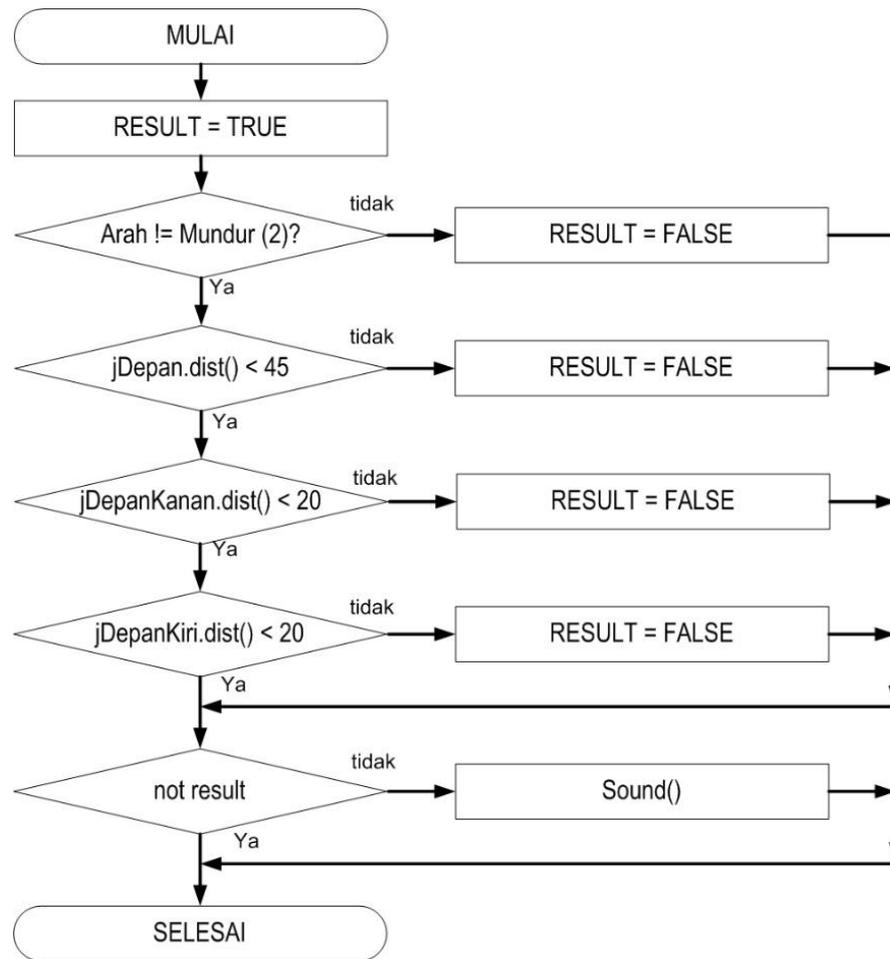
Sebelum *Gesture Robot Motor* melakukan aksi maju maka terlebih dahulu sistem akan melakukan pengecekan status perintah maju. Algoritma cek status perintah maju adalah sebagai berikut;



Gambar 3.15 Algoritma Cek Status Perintah Maju

### 3.5.3. Algoritma Cek Status Perintah Mundur

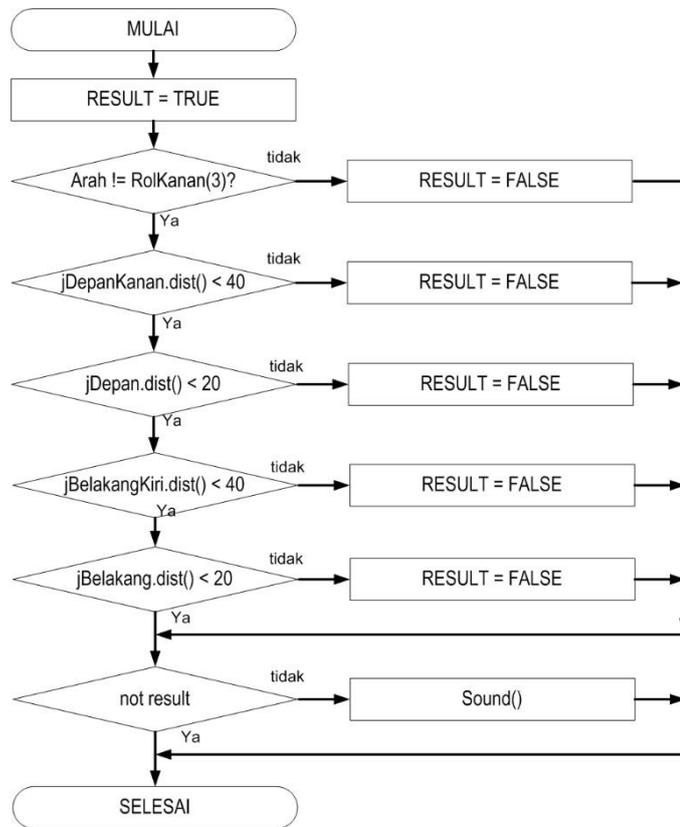
Sebelum *Gesture Robot Motor* melakukan aksi mundur maka terlebih dahulu sistem akan melakukan pengecekan status perintah mundur. Algoritma cek status perintah mundur adalah sebagai berikut;



Gambar 3.16 Algoritma Cek Status Perintah Mundur

#### 3.5.4. Algoritma Cek Status Perintah Rol Kanan

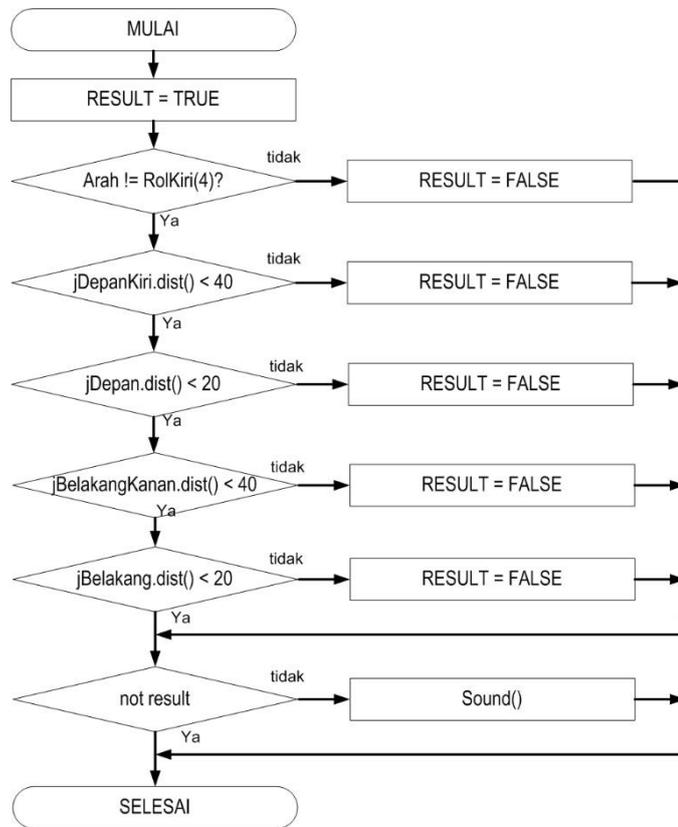
Sebelum *Gesture Robot Motor* melakukan aksi rol kanan maka terlebih dahulu sistem akan melakukan pengecekan status perintah rol kanan. Algoritma cek status perintah rol kanan adalah sebagai berikut;



Gambar 3.17 Algoritma Cek Status Perintah Rol Kanan

### 3.5.5. Algoritma Cek Status Perintah Rol Kiri

Sebelum *Gesture Robot Motor* melakukan aksi rol kiri maka terlebih dahulu sistem akan melakukan pengecekan status perintah rol kiri. Algoritma cek status perintah rol kiri adalah sebagai berikut;



Gambar 3.18 Algoritma Cek Status Perintah Rol Kiri

### 3.6. FUNGSI DASAR OPERATIONAL SISTEM

Fungsi dasar pengendalian motor-motor merupakan kumpulan aksi-aksi dasar robot sebagai hasil keputusan dari pemrosesan data input sensor. Gerakan motor dikendalikan oleh algoritma pwm (*pulse width modulation*). Pwm berperan menentukan kecepatan motor secara programatik. Sistem tidak dirancang untuk dapat mengubah ubah kecepatannya tetapi dipilih konstanta kecepatan normal dan kecepatan kalibrasi, yaitu kecepatan yang paling sesuai untuk implementasi robot sumo dalam arenanya.

Seting gerakan yang dirancang untuk robot gesture ditunjukkan oleh potongan program bahasa Sketch (IDE Arduino Uno) berikut ini :

```
void setMaju(int runDel, int
rlSpeed) {

    if (statusMaju) {

        digitalWrite(INL1, LOW);

        digitalWrite(INL2, HIGH);

        digitalWrite(INR1, HIGH);

        digitalWrite(INR2, LOW);

        analogWrite (PWML, rlSpeed);

        analogWrite (PWMR, rlSpeed);

    } else {

        setStop(stopDel);

    }

    delay(runDel);

}
```

```
void setRolKanan(int runDel, int
rlSpeed) {

    if (statusRolKanan) {

        digitalWrite(INL1, LOW);

        digitalWrite(INL2, HIGH);

        digitalWrite(INR1, LOW);

        digitalWrite(INR2, HIGH);

        analogWrite (PWML, rlSpeed);

        analogWrite (PWMR, rlSpeed);

    } else {

        setStop(stopDel);

    }

    delay(runDel);

}
```

## BAB IV

### HASIL PENELITIAN DAN PEMBAHASAN

#### 4.1. Hasil Implementasi

Pada bagian hasil ini menjelaskan bagaimana hasil dari implementasi dari perancangan-perancangan yang telah di buat dari bab III tentang pembuatan Robot Mobil Kendali Jarak Jauh Berbasis *Gesture* Dengan Menggunakan Sensor Adxl345.

##### 4.1.1 Hasil Pengujian Analisis PEAS Robot Mobil

PEAS merupakan singkatan dari kata *Performance Measure, Environment, Actuators, Sensor*. Ketika kita akan membuat suatu rancangan *agent*, kita harus mengidentifikasi lingkungan masalah atau yang biasa kita sebut dengan “*Task Environment*”.

- Pertama ada *Performance measure*, berisi komponen-komponen yang akan menjadi tolak ukur keberhasilan *agent*.
- Kedua ada *Environment*, berisi kondisi yang dapat mempengaruhi disekitar *agent*.
- Ketiga ada *Actuators*, berisi kemampuan yang dapat *agent* itu lakukan.
- Dan yang terakhir ada *Sensors*, berisi hal-hal apa saja yang dapat diinput *agent*.

Pada penelitian ini maka defenisi PEAS robot mobil yang diteliti adalah sebagai berikut :

Tabel 4.1 Analisis PEAS Robot Mobil

Performance measure	Lingkungan	Aktuator	Sensors
Kebenaran mengikuti perintah atau instruksi	Lantai	Motor DC	ADXL345
sensitivitas terhadap penabrakan	Objek		Ultrasonic

***Task Environment :***

- *Partially Observable*

*Task environment* Robot mobil adalah *Partially Observable* karena ada sebagian kondisi robot mobil yang diteliti hilang dari sensor data, atau dalam kata lain masih terdapat ketidakakuratan dari sensor yang digunakan pada robot mobil (noisy).

- *Single Agent*

*Task environment* Robot mobil adalah *Single agent* karena robot mobil yang dirancang hanya satu robot mobil dan hanya melakukan satu pekerjaan sesuai dengan yang di programkan oleh peneliti.

- *Stochastic*

*Task environment* Robot mobil adalah *Stochastic* karena robot mobil yang dirancang tidak dapat mempelajari atau memprediksi keadaan sebelumnya. Robot mobil yang dirancang tidak mempelajari lingkungan sebelumnya saat melakukan pengujian, tidak mengingat atau mempelajari instruksi arah

gesture tangan pengujian-pengujian sebelumnya ketika akan dilakukan pengetesan di waktu berikutnya.

- Episodik

Robot mobil yang dirancang bersifat episodic, karena pengalaman robot mobil yang terdahulu tidak mempengaruhi apa yang robot mobil itu akan lakukan dimasa depan.

- Discrete

*Task environment* Robot mobil adalah Discrete, nilai yang diberikan pada robot mobil yang dirancang tepat/tentu dan sejumlah terbatas persepsi dan tindakan yang khas dan terdefinisi baik atau dengan kata lain robot mobil yang dirancang dapat melakukan instruksi yang diberikan.

#### **4.1.2. Pembahasan *Performance Measure***

Pengukuran terhadap performa robot gesture adalah sebagai berikut :

##### **4.1.2.1 Pengukuran Performa dan Hasil Analisis**

Penentuan pengukuran performa dari agen:

1. Kebenaran Mengikuti Perintah atau Instruksi
  - Nilai 100 untuk nilai awal
  - -10 ketika agen lambat dan salah mengikuti instruksi atau perintah yang diberikan (misal arah yang diperintahkan adalah maju, yang dilakukan agen adalah mundur)
  - -40 jika agen tidak melakukan apa apa ketika di berikan instruksi (diam).

## 2. Sensitivitas Terhadap Penabrakan

- Nilai awal adalah 100
- -50 jika agen menabrak suatu objek tapi setelahnya memilih diam
- Nilai 0 jika agen menabrak suatu objek yang ada didekatnya dan masih memilih untuk tetap jalan.

Jadi dilakukanlah pengujian dengan tempat atau lingkungan yaitu Lab Riset Jaringan. Hasil dari pengujian tersebut dapat dilihat pada tabel dibawah ini:

Keterangan :

Pengujian dilakukan pada lingkungan (*environment*) lab riset jaringan dengan keadaan lantai datar keramik. Lantai pada lab Riset Jaringan menggunakan keramik yang berukuran 30 x 30 cm per petaknya. Sedangkan luas daerah khusus untuk pengujian robot mobil pada lab riset jaringan sekitar 4 x 3,5 meter. Dan diuji oleh peneliti sendiri. Ada sebanyak 20 kali percobaan yang dilakukan untuk menguji kebenaran robot dalam mengikuti perintah, misal kebenaran dalam setiap instruksi arah yang diberikan (maju, mundur, rol, drift dan sebagainya). Juga pengujian untuk sensor Ultrasonic yang digunakan untuk mengatasi penabrakan pada objek yang ada didekat robot. Adapun hasil dari pengujian tersebut dengan keadaan lantai keramik dapat dilihat pada tabel dibawah ini.

Tabel 4.2 Hasil Pengujian Analisis Peas

Uji Coba	Kebenaran mengikuti perintah atau Instruksi										Hasil Akhir (point)	
	Maju	Mundur	Rol Maju Kiri	Rol Maju Kanan	Rol Mundur Kiri	Rol Mundur Kanan	Drift Maju Kiri	Drift Maju Kanan	Drift Mundur Kiri	Drift Mundur Kanan		
1	100	100	100	60	60	100	100	100	100	100	100	92
2	100	100	100	100	60	60	100	100	100	100	100	92
3	100	100	100	100	60	60	100	100	100	100	100	92
4	100	100	100	100	100	100	100	100	100	100	100	100
5	100	100	60	100	100	100	100	100	100	100	100	96
6	100	100	60	100	100	100	100	100	100	100	60	92
7	100	100	100	100	100	100	100	100	100	100	100	100
8	100	100	100	100	100	60	100	100	100	100	100	96
9	100	100	100	100	100	100	100	100	100	100	100	100
10	100	100	100	100	100	100	100	100	100	100	100	100
11	100	100	100	100	100	100	100	100	100	100	100	100
12	100	100	100	100	100	100	100	100	100	100	100	100
13	100	100	100	100	100	100	100	100	100	100	100	100
14	100	100	100	100	100	100	100	100	100	100	100	100
15	100	100	100	60	100	100	100	100	100	100	100	96
16	100	100	100	60	100	60	100	100	60	100	100	88
17	100	100	100	100	100	100	100	100	100	100	100	100
18	100	100	100	100	100	100	100	100	100	100	100	100
19	100	100	100	100	100	100	100	100	100	100	100	100
20	100	100	100	100	100	100	100	100	100	100	100	100
	<b>Rata-rata</b>										<b>97,2</b>	

Tabel 4.3 Hasil Pengujian Sensitivitas Robot Mobil Terhadap Penabrakan

Uji Coba	Sensitivitas Terhadap Penabrakan (point)
1	100
2	100
3	0
4	100
5	50
6	100
7	0
8	100
9	100
10	100
11	100
12	100
13	0
14	0
15	100
16	100
17	50
18	100
19	100
20	100
Rata-rata	75

#### **4.1.2.2 Pembahasan**

Dengan data-data tersebut dapat diambil kesimpulan bahwa : robot mobil telah dapat mengikuti instruksi dan berjalan sesuai dengan arah yang diperintah kan dengan hasil nilai rata-rata yang didapatkan 97,2 point. Untuk sensor ultrasonic telah dapat berfungsi, robot mobil memilih berhenti ketika ada penghalang atau objek didekatnya. Kecuali dalam beberapa kasus, ada beberapa kali dalam pengujian sensor belum sempurna masih ada penabrakan ketika robot mobil dalam keadaan laju tiba tiba ada objek didepannya, dalam kondisi seperti itu robot mobil masih melakukan penabrakan.

Dari pengujian tersebut, maka untuk akurasi hasil uji sensitivitas robot mobil terhadap penabrakan mendapatkan 7.5 point.

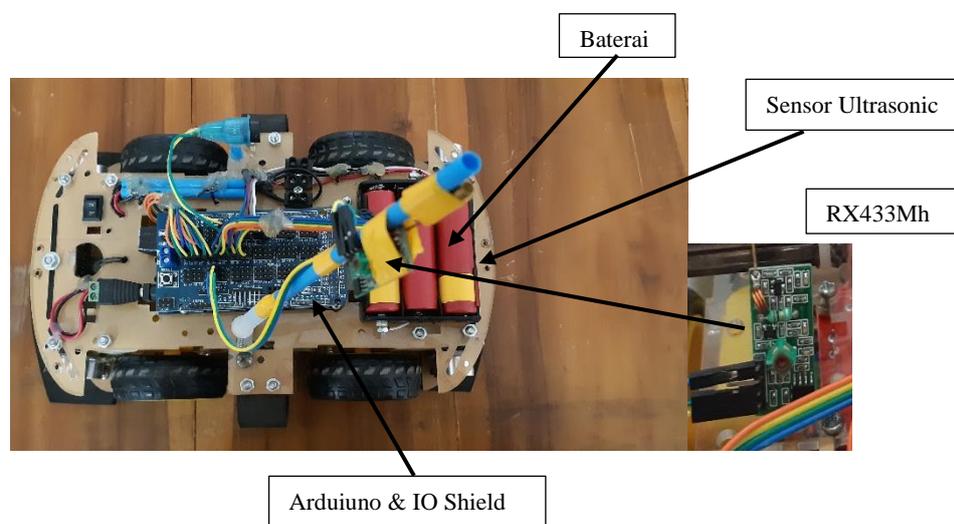
#### **4.1.3 Hasil Desain Alat**

Hasil desain atau perancangan alat yang telah dibuat berdasarkan pada sistem hardware, desain alat meliputi rangkaian arduino, sensor ADXL345, sensor Ultrasonik, Motor DC, Driver Motor, Baterai, Transmitter, Receiver. Untuk lebih jelas dapat dilihat pada penjelasan berikut.

##### **4.1.3.1 Rangkaian Gesture Robot Motor**

Arduino uno pada motor dihubungkan dengan arduino IO Shield yang tepat diletakkan di atasnya. Kemudian receiver yang terdapat pada motor dihubungkan pada arduino IO Shield dengan keterangan gnd (*ground*) pada receiver terhubung

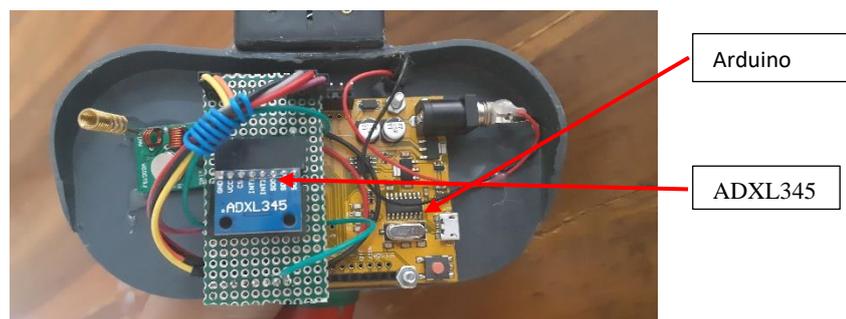
dengan g (ground) IO shield, data pada receiver terhubung pada + IO shield, dan vcc receiver terhubung pada pin S IO shield. Receiver yang terdapat pada motor berfungsi untuk menerima data yang dikirimkan nanti oleh transmitter. Selanjutnya terdapat sensor *ultrasonic*, sensor ini berfungsi sebagai pendeteksi jarak atau objek yang ada disekitarnya sehingga robot mobil tidak menabrak apapun ketika sedang berjalan. Terdapat 10 ultrasonic pada robot mobil yang dirancang dan semua Sensor ini terhubung oleh arduino IO shield dengan keterangan vcc pada ultrasonic terhubung pada pin v IO shield, trig pada ultrasonic terhubung pada trig yang ditambahkan dalam perancangan, echo pada ultrasonic terhubung pada S yang terdapat pada IO Shield, dan gnd (ground) pada ultrasonic terhubung pada g(ground) pada IO shield. Dan pada perancangan penelitian ini juga adanya baterai yang mempunyai peran penting sebagai sumber daya yang membantu menggerakkan robot mobil, dengan rangkaian baterai 1, kabel biru dan hijau terhubung pada arduino motor, dan kabel hitam berkesinambungan dengan motor DC, sedangkan baterai ke 2 menjadi tambahan daya yang dihubungkan langsung pada rangkaian baterai 1 dan motor DC yang dirangkai sebelumnya.



Gambar 4.1 Rangkaian Gesture Robot Motor

#### 4.1.3.2 Rangkaian Adxl345 dengan Arduino

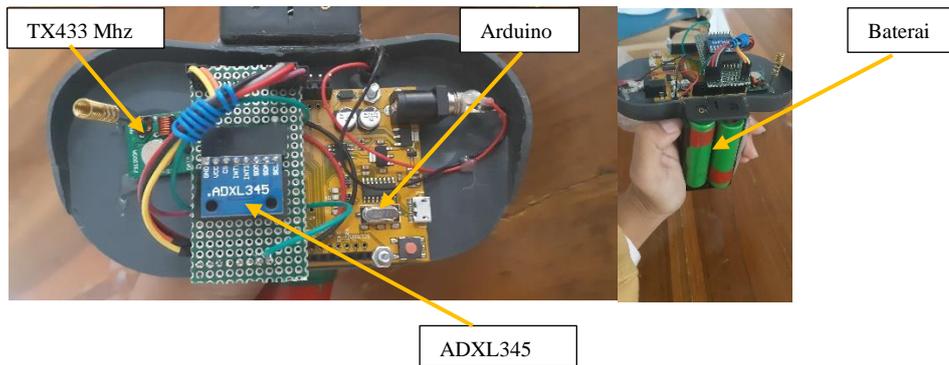
Arduino terhubung dengan ADXL345, kemudian transmitter yang sudah diprogram inilah yang menjadi gesture penggerak yang diletakkan ditangan, yang akan mengirimkan data pada receiver yang terletak pada robot mobil. Adapun keterangan rangkaiannya ialah, gnd pada transmitter terhubung pada gnd yang terdapat pada arduino, data pada transmitter terhubung pada pin 12 arduino, dan vcc terhubung pada 5out arduino.



Gambar 4.2 Rangkaian Adxl345 dengan Arduino

#### 4.1.3.3 Rangkaian Adxl345 dengan Arduino, Baterai dan Tx433 Mhz

Setelah adxl dan arduino terangkai, maka rangkaian tersebut dihubungkan dengan baterai yang memberikan daya untuk adxl bisa menyala dan dapat digunakan, juga dengan Tx433 Mhz yang ditambahkan agar connectivitas atau pengiriman data transmitter ke receiver lebih lancar dan tidak terhambat. Dapat dilihat pada gambar dibawah ini.



Gambar 4.3 Rangkaian Gesture Robot Controller

## 4.2 Pembahasan

Pada bagian ini kita akan membahas mengenai beberapa program khusus robot gesture.

### 4.2.1 Source Code Program Khusus

*Source code* program ini berisi *code* program yang berperan penting pada sistem *Robot Mobil Kendali Jarak Jauh berbasis Gesture tangan dengan menggunakan Sensor ADXL345*, untuk lebih lanjut dapat dilihat pada penjelasan berikut.

#### 4.2.1.1 Fungsi untuk Konversi Data Acceleration (x,y,z)

Berfungsi untuk mengkonversi data ADXL345 yang diterima oleh arduino, *result* dari fungsi ini adalah Data Control Robot berupa arah maju, mundur, kiri, kanan, berhenti. Pada bagian transmitter, langkah pertama yang dilakukan dalam programnya ialah mendefinisikan variabel, dari data data yang telah diinisialisasikan dibuat sebuah fungsi pengambilan data dari adxl345 yang sebelumnya menjadi salah satu inisialisasi variabel. Kemudian di gunakan fungsi

percabangan untuk beberapa kondisi, seperti adxl345 yang telah diteliti jika angka sesuai batasan yang telah ditentukan maka dengan deprogram percabangan dapat ditentukan arah baik itu maju, mundur, kiri maupun kanan. Kode program konversi, dapat lihat pada *source code* berikut.

```
int BacaADXL345() {
  int result = 0;
  readAccel();
  delay(10);
  if (x > 0 && y > 0) {
    if (x < 40 && y < 40) {
      Serial.println("Stop");
      result = henti;
      return result;
    } else if (x < 40 && y > 40 && y < 120) {
      Serial.println("maju");
      result = maju;
      return result;
    } else if (x > 40 && x < 120 && y < 40) {
      Serial.println("majuDriftKanan");
      result = majuDriftKanan;
      return result;
    } else if (x > 40 && x < 120 && y > 40 && y < 120) {
      Serial.println("rolKanan");
      result = rolKanan;
      return result;
    }
  } else if (x < 0 && y > 0) {
    if (x > -40 && y < 40) {
      Serial.println("Stop");
      result = henti;
      return result;
    } else if (x > -40 && y > 40 && y < 120) {
      Serial.println("maju");
      result = maju;
      return result;
    } else if (x < -40 && x > -120 && y < 40) {
      Serial.println("majuDriftKiri");
      result = majuDriftKiri;
      return result;
    } else if (x < -40 && x > -120 && y > 40 && y < 120) {
      Serial.println("rolKiri");
      result = rolKiri;
      return result;
    }
  }
}
```

```

} else if (x < 0 && y < 0) { //-----
-----selektor kuadrant 3
    if (x > -40 && y > -40) {
        Serial.println("Stop");
        result = henti;
        return result;
    } else if (x > -40 && y < -40 && y > -120) {
        Serial.println("mundur");
        result = mundur;
        return result;
    } else if (x < -40 && x > -120 && y > -40) {
        Serial.println("mundurDriftKiri");
        result = mundurDriftKiri;
        return result;
    } else if (x < -40 && x > -120 && y < -40 && y > -120) {
        Serial.println("rolKiri");
        result = rolKiri;
        return result;
    }
} else if (x > 0 && y < 0) { //-----
-----selektor kuadrant 4
    if (x < 40 && y > -40) {
        Serial.println("Stop");
        result = henti;
        return result;
    } else if (x < 40 && y < -40 && y > -120) {
        Serial.println("mundur");
        result = mundur;
        return result;
    } else if (x > 40 && x < 120 && y > -40) {
        Serial.println("mundurDriftKanan");
        result = mundurDriftKanan;
        return result;
    } else if (x > 40 && x < 120 && y < -40 && y > -120) {
        Serial.println("rolKanan");
        result = rolKanan;
        return result;
    }
} else {
    result = 0;
    return result;
}
}

```

#### 4.2.1.2 Baca Data Control/Perintah

Fungsi ini digunakan untuk membaca Data Control yang ditransmisikan oleh Transmitter RF433Mhz, kemudian disimpan ke variabel global supaya bisa diakses pada fungsi-fungsi lainnya. Dibuat sebuah method dengan nama `monitorPerintah` (ini merupakan sebuah metod untuk membaca perintah). Kemudian dibuat kode untuk mengidentifikasi pemancar. Setelah itu diekstrak perintah `msg` ke variabel yang telah di tentukan sebelumnya. Dapat dilihat pada code program dibawah ini.

```
void bacaPerintah() {
  byte len = receiver.recvPackage((byte *)msg, &senderId, &packageId);
  if (senderId == 7) {
    sscanf(msg, "%s%", cArah);
    Serial.print("msg= ");
    Serial.println(cArah);
    iArah = atoi(cArah);
    Serial.println(iArah);
    sound2();
  }
}
```

#### 4.2.1.3 Code Program untuk Berhenti

Fungsi ini berguna untuk memberhentikan robot ketika menerima perintah berhenti, atau ketika robot hendak menabrak maka keputusan robot memilih berhenti. Pada method ini di lakukan pengaturan agar semua roda robot baik kiri mau pun kanan dan pwm nya menjadi low.

```
void setStop(int runDel) {  
    digitalWrite(INL1, LOW);  
    digitalWrite(INL2, LOW);  
    digitalWrite(INR1, LOW);  
    digitalWrite(INR2, LOW);  
    digitalWrite(PWML, iSpeed[0]);  
    digitalWrite(PWMR, iSpeed[0]);  
  
    delay(stopDel);  
}
```

#### 4.2.1.4 Code Program untuk Maju

Fungsi ini berguna untuk membuat motor maju ketika mendapat perintah maju dari data control. Buat method setMaju, kemudian ada percabangan yang membaca perintah jika variabel yang diinisialkan sesuai dengan keadaan yang ditentukan, maka output yang dikeluarkan adalah maju. Jika tidak robot harus berhenti.

Adapun untuk gerak maju dan mundur, ketika diuji pada suatu garis keramik yang lurus masih belum dapat bergerak lurus secara sempurna, robot mobil sedikit melenceng kearah kiri atau kerah kanan.

Cara mengatasi robot yang mengikuti instruksi tapi tidak sempurna ialah dengan cara melakukan kalibrasi. Dalam pengujian untuk instruksi maju, robot mobil dijalankan pada garis lurus, tetapi hasil yang didapat robot mobil sedikit melenceng kekanan. Maka dengan itu dapat dilakukan 2 kalibrasi, bisa jadi roda sebelah kiri ditambahkan kecepatan (pwm nya), atau memilih untuk roda sebelah kanan agar dturunkan kecepatannya.

Adapun beberapa penyebab robot mobil tidak berjalan maju atau mundur secara lurus yang utuh, diantaranya ialah lingkungan uji robot, kualitas motor dc yang berbeda, dan terjadinya sistemik error yaitu dynamo tidak menerima nilai kecepatan yang sama, dan sebagainya. Untuk Code program dapat dilihat dibawah ini.

```

void setMaju(int runDel, int rlSpeed) {
  if (statusMaju) {
    digitalWrite(INL1, LOW);
    digitalWrite(INL2, HIGH);
    digitalWrite(INR1, HIGH);
    digitalWrite(INR2, LOW);
    analogWrite (PWML, rlSpeed);
    analogWrite (PWMR, rlSpeed);
  } else {
    setStop(stopDel);
  }
  delay(runDel);
}

void Maju() {
  setStop(iDel[2]);
  boolean status = statusMaju();
  while (status) {
    setMaju(iDel[2], iSpeed[10]);
    status = statusMaju();
  } setStop(iDel[2]);
}

```

#### 4.2.1.5 Code Program untuk Mundur

Fungsi ini berguna untuk membuat motor mundur ketika mendapat perintah mundur dari data control. Buat method setMaju, kemudian ada percabangan yang membaca perintah jika variabel yang diinisialkan sesuai dengan keadaan yang ditentukan, maka output yang dikeluarkan adalah mundur. Jika tidak robot harus berhenti. Code program dapat dilihat dibawah ini.

```
boolean statusMundur() {
    boolean result = true;
    if (iArah != 2) result = false;
    if (jBelakang.dist() < 45) result = false;
    if (jBelakangKanan.dist() < 20) result = false;
    if (jBelakangKiri.dist() < 20) result = false;
    if (not result) {
        sound();
    }
    return result;
}

void Mundur() {
    setStop(iDel[2]);
    boolean status = statusMundur();
    while (status) {
        setMundur(iDel[2], iSpeed[10]);
        status = statusMundur();
    } setStop(iDel[2]);
}
```

#### 4.2.1.6 Code Program untuk Rol Kanan

Fungsi ini berguna untuk membuat motor belok kanan ketika mendapat perintah belok kanan dari data control. Perintah ini berfungsi untuk mengatur apabila belok kanan maka roda sebelah kanan di set *OFF/LOW* sedangkan roda sebelah kiri diset *ON/HIGH* dengan kecepatan yang didapatkan dari data yang dikirim oleh transmitter. Jika sensor ultrasonic mendeteksi ada penghalang maka keputusan robot adalah berhenti sebelum menabrak objek/penghalang.

```

void setRolKanan(int runDel, int rlSpeed) {
  if (statusRolKanan) {
    digitalWrite(INL1, LOW);
    digitalWrite(INL2, HIGH);
    digitalWrite(INR1, LOW);
    digitalWrite(INR2, HIGH);
    analogWrite (PWML, rlSpeed);
    analogWrite (PWMR, rlSpeed);
  } else {
    setStop(stopDel);
  }
  delay(runDel);
}

void RolKanan() {
  setStop(iDel[0]);
  boolean status = statusRolKanan();
  while (status) {
    setRolKanan(iDel[2], iSpeed[10]);
    status = statusRolKanan();
  } setStop(iDel[0]);
}

```

#### 4.2.1.7 Code Program untuk RolKiri

Fungsi ini berguna untuk membuat motor belok kiri ketika mendapat perintah belok kiri dari data control. Perintah ini berfungsi untuk mengatur apabila belok kiri maka roda sebelah kiri di set *OFF/LOW* sedangkan roda sebelah kanan diset *ON/HIGH* dengan kecepatan yang didapatkan dari data yang dikirim oleh transmitter. Jika sensor ultrasonic mendeteksi ada penghalang maka keputusan robot adalah berhenti sebelum menabrak objek/penghalang.

```

boolean statusRolKiri() {
    boolean result = true;
    if (iArah != 4) result = false;
    if (jDepanKiri.dist() < 40) result = false;
    if (jDepan.dist() < 20) result = false;
    if (jBelakangKanan.dist() < 40) result = false;
    if (jBelakang.dist() < 20) result = false;
    if (not result) {
        sound();
    }
    return result;
}

void RolKiri() {
    setStop(iDel[2]);
    boolean status = statusRolKiri();
    while (status) {
        setRolKiri(iDel[2], iSpeed[10]);
        status = statusRolKiri();
    } setStop(iDel[2]);
}

```

#### 4.2.2. Kelemahan Robot Mobil

Adapun kelemahan dari Robot mobil jarak jauh berbasis gesture tangan dengan menggunakan sensor ADXL345 yang telah dirancang adalah sebagai berikut:

1. sensor *Ultrasonic* yang digunakan sudah dapat berfungsi walaupun dalam beberapa keadaan masih terjadi error (penabrakan). Robot mobil seharusnya sudah berhenti atau memilih bergerak kearah lain sebelum menabrak suatu objek atau orang baik robot mobil dalam keadaan kecepatan laju ataupun lamban.
2. Bidang datar dengan permukaan halus dan kasar mempengaruhi gerakan robot mobil.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Setelah melakukan proses perancangan dan implementasi Robot Mobil Kendali Jarak Jauh Berbasis Gesture dengan Menggunakan Sensor ADXL345 serta pengujian yang telah dilakukan maka di dapat kesimpulan sebagai berikut :

1. Robot Mobil Kendali Jarak Jauh Berbasis Gesture telah dapat dibangun dengan menggunakan sensor ADXL345 dan sesuai apa yang dirancang. Dengan diprogram melalui Arduino UNO IDE, dan rakitan alat alat yang yang digunakan serta penggunaan sensor *Ultrasonic* sebagai pengukur jarak agar tidak menabrak suatu objek didekatnya.
2. *Gesture Remote Robot* berhasil dikendalikan sesuai intruksi yang kirim oleh *Gesture Remote Controller* dengan hasil akurasi rata-rata pengujian yang didapatkan 97,2 point. Robot dapat bergerak maju, mundur, berhenti, rol kiri, rol kanan, drift maju kiri, drift maju kanan, drift mundur kiri dan drift mundur kanan.
3. Sensor ultrasonic telah dapat berfungsi, robot mobil memilih berhenti ketika ada penghalang atau objek didekatnya. Kecuali dalam beberapa kasus, ada beberapa kali dalam pengujian sensor belum sempurna masih terjadi penabrakan ketika robot mobil dalam keadaan cepat ( $>0,4$  m/s) tiba tiba ada objek didepannya, dalam kondisi seperti itu robot mobil masih melakukan

penabrakan. Dan sensor robot mobil bekerja dengan baik ketika kecepatan sedang (0,25 m/s) dan kecepatan lambat (0,15 m/s).

Dari pengujian tersebut, maka untuk akurasi hasil uji sensitivitas robot mobil terhadap penabrakan mendapatkan 7.5 point.

## PUSTAKA

- [1] Adisty, Irman Supriadi. 2014. *Pengembangan Sistem Monitoring Vibrasi Pada Kipas Pendingin Menggunakan Accelerometer ADXL345 dengan Metode FFT Berbasis Labview*. Program Studi Fisika Fakultas Sains dan Teknologi Universitas Islam Syarif Hidayatullah Jakarta.
- [2] Adriansyah, andi. 2013. *Rancang bangun prototipe elevator menggunakan microcontroller arduino atmega 328p*. Jurnal Teknologi Elektro Universitas Mercu Buana, Jakarta, Indonesia.
- [3] Angga Pratama, Rahadhian. 2012. *Sensor Parkir Mobil Berbasis Mikrokontroler AT89S51 Dengan Bantuan Mini Kamera*. Jurnal Komputasi STMIK JAKARTA STI&K.
- [4] Arifin, Jauhari. 2016. *Perancangan muottal otomatis menggunakan Mikrokontroller arduino mega 2560*. Jurnal Media Infotama Program Studi Teknik Komputer Fakultas Ilmu Komputer Universitas Dehasen Bengkulu.
- [5] Budi Setyawan .2016. **ROBOT LINE FOLLOWER MENGGUNAKAN KENDALI JARAK JAUH BERBASIS ANDROID**. Program Studi Sistem Komputer Fakultas Ilmu Komputer Universitas Narotama
- [6] Hasan. 2005. *Sistem Komunikasi Radio untuk Memantau Status Pencacah di Limbah Radioaktif*. Pusat Penelitian dan Pengembangan Teknik Nuklir-Batan.
- [7] Martinus. 2013. *Pembuatan otomasi pengaturan kereta api, pengereman, Dan palang pintu pada rel kereta api mainan berbasis Mikrokontroler*. Jurnal FEMA Jurusan Teknik Mesin, Fakultas Teknik Universitas Lampung.
- [8] Prayogo, Rudito. 2012. *Pengaturan PWM (Pulse Width Modulation) dengan PLC*. Universitas Brawijaya Malang.
- [9] Rohman, Alfas Zainur. 2015. *Rancang Bangun Alat Ukur Getaran Menggunakan Sensor Micro Electro Mechanical System (Mems) Akselerometer*. Edu Elekrika Journal Jurusan Teknik Elektro, Fakultas Teknik, Universitas Negeri Semarang, Indonesia.
- [10] Rois, Muhammad Fadli. 2017. *Perancangan Sistem Robot Pengukur Kondisi Jalan Berbasis Fuzzy Logic*. Skripsi Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.
- [11] Syahrin, Alfi. 2016. *Prototype Sistem Monitoring Temperatur Menggunakan Arduino Uno R3 Dengan Komunikasi Wireless*. Jurnal Teknik Mesin, Program Studi Teknik Elektro, Fakultas Teknik, Universitas Mercu Buana, Jakarta.