**PROYEK AKHIR**

**RANCANG BANGUN KOPER SMART GESTURE**

**BERBASIS ARDUINO**

**Oleh :**

**Dedi Munandar**

**2015302008**



**PROGRAM STUDI DIPLOMA III**

**TEKNIK INFORMATIKA**

**POLITEKNIK ACEH**

**2018**

**PROYEK AKHIR**

**RANCANG BANGUN KOPER SMART GESTURE**

**BERBASIS ARDUINO**

**Disusun Sebagai Salah Satu Syarat Penyelesaian Program Diploma Tiga**

**Program Studi Teknik Informatika Politeknik Aceh**

**Oleh :**

**Dedi Munandar**

**2015302008**



**PROGRAM STUDI DIPLOMA III**

**TEKNIK INFORMATIKA**

**POLITEKNIK ACEH**

**2018**

**PENGESAHAN PROYEK AKHIR**

**RANCANG BANGUN SISTEM PENGAMBILAN GAS BERBASIS RFID**

Disusun oleh:

Nama : Dedi Munandar

NIM : 2015302008

Program Studi : Teknik Informatika

Telah Diujikan Pada Hari Senin/Tanggal 12 November 2018

*dan dinyatakan telah memenuhi syarat untuk memperoleh gelar ahli madya dibidang teknik informatika, oleh dewan penguji proyek akhir*

Pembimbing 1 Pembimbing 2

Safwan, S.T,.M.T Feri Susilawati, S.ST., M.T.

NRP. 120150107 NRP. 120110122

Penguji 1 Penguji 2

Ichsan, S.Kom., M.T., M.Sc. Devi Mulia Sari. S.HI.,S.Si,M.Ag

NRP. 120110112 NRP. 120160103

Ketua Prodi Teknik Informatika

Ichsan, S.Kom., M.T., M.Sc.

NRP. 120110112

# **SURAT PERNYATAAN KEASLIAN PENELITIAN**

Saya yang bertanda tangan di bawah ini :

Nama : Dedi Munandar

NIM : 2015302008

Menyatakan dengan sebenarnya, bahwa Proyek Akhir yang saya susun dengan judul : **RANCANG BANGUN KOPER SMART GESTURE BERBASIS ARDUINO** merupakan hasil karya tulis ilmiah asli yang saya susun sendiri, bukan merupakan plagiasi atau tiruan karya orang lain.

Jika dibelakang hari ternyata bahwa karya saya ini terbukti dengan jelas merupakan hasil plagiasi/tiruan karya seseorang terdahulu maka saya siap menerima sanksi sesuai dengan ketentuan hukum yang berlaku.

Yang menyatakan,

Dedi Munandar

# **LEMBAR PERSETUJUAN PUBLIKASI**

Sebagai mahasiswa Politeknik Aceh, yang bertanda tangan dibawah ini, saya :

Nama : Dedi Munandar

NIM : 2015302008

Karya ilmiah yang berjudul : **RANCANG BANGUN KOPER SMART GESTURE BERBASIS ARDUINO** beserta perangkat yang diperlukan (bila ada). Politeknik Aceh berhak menyimpan, mengalih-mediakan / formatkan, mengelolanya dalam bentuk pangkalan data (*database*), mendistribusikannya di internet atau media lain untuk kepentingan Politeknik Aceh tanpa perlu izin dari saya sebagai penulis.

Demikian pernyataan ini yang saya buat dengan sebenarnya.

Dibuat di : Banda Aceh

Pada Tanggal : Senin, 12 November 2018

Dedi Munandar

# ***ABSTRACT***

*Name : Dedi Munandar*

*NIM : 2015302008*

*Title : Smart Gesture suitcase based on Arduino.*

*Smart Gesture is one of the interesting technologies to be developed. Smart gesture can be harnessed into a vast range of things to ease the people work people. In general the suitcase uses to pack the stuff which is pulled manual. By using a Smart Gestures suitcase, user does not need to pull the trunk anymore, it can be controlled by using gloves. Smart Gesture is applied on a suitcase using the accelerometer sensor to start the motor. It controls the suitcase on the level of the slope of the hands resulting from the accelerometer sensor. To minimize the discharge of excess power in pulling the suitcase, it makes the user to control the suitcase without pulling it directly.*

*Information : Suitcase, Arduino.*

# **ABSTRAK**

Nama : Dedi Munandar

NIM : 2015302008

Judul : Rancang Bangun Koper Smart Gesture Berbasis Arduino

*Smart Gesture* merupakan salah satu teknologi yang menarik untuk dikembangkan. *Smart gesture* dapat dimanfaatkan menjadi berbagai macam hal sehingga memudahkan pekerjaan manusia. Pada umumnya koper digunakan untuk mengemas barang bawaan yang ditarik secara manual, dengan adanya *Smart Gestur* pengguna koper tidak perlu mengeluarkan tenaga untuk menarik koper lagi, cukup mengontrol dengan glove. *Smart Gesture* yang diterapkan pada koper menggunakan sensor *accelerometer* untuk menjalankan motor, sehingga pengontrolan koper hanya dengan mengandalkan tingkat kemiringan tangan yang dihasilkan dari sensor *accelerometer.* Untuk meminimalisir keluarnya tenaga berlebih dalam menarik koper maka para pengguna koper dapat mengontrol koper tanpa harus menariknya secara langsung.

Kata Kunci : *Koper, Smart Gesture, Pengontrolan*

# **KATA PENGANTAR**

Puji syukur penulis panjatkan kehadirat Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan Proyek Akhir ini yang berjudul **“RANCANG BANGUN KOPER SMART GESTURE BERBASIS ARDUINO”** yang merupakan salah satu syarat untuk menyelesaikan pendidikan Diploma III pada program studi Teknik Informatika Politeknik Aceh.

Pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada dosen pembimbing dan teman-teman yang telah banyak membantu sehingga penulis dapat menyelesaikan proyek akhir ini masih jauh dari kata sempurna. Untuk itu, penilis mengharapkan kritik dan saran yang membangun agar penulis dapat memperbaiki kekurangan yang ada.

Akhir kata, penulis berharap semoga berguna kepada seluruh konsumen untuk memudahkan dalam penggunaan koper.

Banda Aceh, 12 November 2018

Penulis

**DAFTAR ISI**

LEMBAR PENGESAHAN ii  
[SURAT PERNYATAAN KEASLIAN PENELITIAN ii](#_Toc530095608)

[LEMBAR PERSETUJUAN PUBLIKASI iv](#_Toc530095609)

*[ABSTRACT](#_Toc530095610)* [v](#_Toc530095610)

[ABSTRAK vi](#_Toc530095611)

KATA PENGANTAR vii

DAFTAR ISI viii

[DAFTAR GAMBAR x](#_Toc530095612)

[DAFTAR TABEL xi](#_Toc530095613)

[BAB I PENDAHULUAN 1](#_Toc530095614)

[1.1. Latar Belakang 1](#_Toc530095615)

[1.2. Tujuan 2](#_Toc530095616)

[1.3. Manfaat 2](#_Toc530095617)

[1.4. Rumusan Masalah. 2](#_Toc530095618)

[1.5 Batasan Masalah. 3](#_Toc530095619)

[BAB II](#_Toc530095620) [DASAR TEORI 4](#_Toc530095621)

[2.1. Pengenalan Hardware 4](#_Toc530095622)

[2.1.1 Arduino Uno 4](#_Toc530095623)

[2.1.2 Arduino Nano 5](#_Toc530095624)

[2.1.3 Relay 4 Channel 5V 6](#_Toc530095625)

[2.1.4 Accelerometer 7](#_Toc530095626)

[2.1.5 Kabel Jumper 8](#_Toc530095627)

[2.1.6 Modul RF Rx/Tx 433 MHz 9](#_Toc530095628)

[2.1.7 Projct Board / BreadBoard 10](#_Toc530095629)

[2.1.8 Motor DC 11](#_Toc530095630)

[BAB III](#_Toc530095631) [PERANCANGAN SISTEM 12](#_Toc530095632)

[3.1 Analisis Sistem 12](#_Toc530095633)

[3.2 Blok Diagram 12](#_Toc530095634)

[3.3 Desain Alur / Flowchart Koper. 14](#_Toc530095635)

[3.4. Desain Alur / Flowchart Glove 16](#_Toc530095636)

[3.5 Rancangan Arduino RF Rx/Tx 433 MHz 17](#_Toc530095637)

[3.6 Rancangan Master Arduino Nano. 18](#_Toc530095641)

[3.7 Rancangan Master Arduino Uno 19](#_Toc530095647)

[BAB IV](#_Toc530095650) [HASIL 21](#_Toc530095651)

[4.1 Rancangan Smart Gesture 21](#_Toc530095652)

[4.2. Rancangan Koper 22](#_Toc530095655)

[4.3 Hasil Analisa Sistem 23](#_Toc530095660)

[BAB V](#_Toc530095661) [PENUTUP 25](#_Toc530095662)

[5.1. Kesimpulan. 25](#_Toc530095663)

[5.2. Saran. 25](#_Toc530095664)

[DAFTAR PUSTAKA 26](#_Toc530095665)

[Lampiran 1. Program Grove. 27](#_Toc530095666)

[Lampiran 2. Program Koper. 32](#_Toc530095667)

[BIODATA PENULIS 36](#_Toc530095668)

# **DAFTAR GAMBAR**

Gambar 2.1 *Arduino Uno R3 4*

Gambar 2.2 Arduino Nano 5

Gambar 2.3 Accelerometer Relay 4 Channel 5V 6

Gambar 2.4 Accelerometer 8

Gambar 2.5 Kabel Jumper 9

Gambar 2.6 Modul RF Rx/Tx 433 MHz 10

Gambar 2.7 Breadboard 10

Gambar 2.8 Motor DC 11

Gambar 3.1 Blok Diagram 12

Gambar 3.2 Diagram Alur Koper 14

Gambar 3.3 Diagram Alur Glove 16

Gambar 3.4 Rancangan Arduino *Module* RF Rx/Tx 433 MHz 17

Gambar 3.5 Rancangan Master Arduino Nano 18

Gambar 3.6. Rancangan Arduino Uno 20

Gambar 4.1 Rancangan Smart Gesutre 21

Gambar 4.2 Lampiran Program Koper 22

# **DAFTAR TABEL**

Tabel 4.1 Tabel Pengujian 23

Tabet 4.2 Tabel Kondisi24

**BAB I**

**PENDAHULUAN**

## **Latar Belakang**

Setiap tahunnya bahkan setiap harinya banyak orang yang berpergian keluar negeri atau pulang ke kampung halaman. Koper adalah salah satu solusi untuk melakukan perjalanan tersebut dikarenakan fungsi koper sangatlah penting untuk membawa barang yang diperlukan.

Koper atau juga disebut tempat penyimpanan barang atau pakaian merupakan alat yang sangat sering digunakan untuk orang berpergian. Banyak koper yang digunakan dalam bentuk manual yaitu dengan cara mendorong atau menarik koper tersebut, sehingga banyak orang yang merasa kesakitan di bagian tangannya.

Dengan perkembangan teknologi saat ini Koper dapat kita gunakan secara otomatis berdasarkan gerak tangan atau yang dikenal sebagai teknologi Smart Gesture.

Smart Gesture adalah salah satu cara untuk membantu dan memudahkan mengerakkan koper tanpa menggunakan tenaga manusia yang ekstra, karena alat ini dapat menggerakkan suatu objek berdasarkan gerakan tangan.

Berdasarkan latar belakang tersebut maka peneliti mengambil proyek akhir dengan judul **“Rancang Bangun Koper Smart Gesture Berbasis Arduino”** dengan harapan kedepannya dapat memudahkan pengguna dalam penggunaan koper.

## **Tujuan**

Tujuan penulisan proyek akhir ini adalah :

1. Untuk memudahkan dan membantu para pengguna dalam membawa barang.
2. Memudahkan pengendalian dengan menggunakan sarung tangan.
3. Menggunakan *Accelerometer* sebagai pengendalinya *(Smart Gesture)*

## **Manfaat**

Adapun manfaat dari penelitian Tugas Akhir Rancang Bangun Smart Gesture Koper adalah :

1. Mempermudah para pengguna koper.
2. Meminimalisir tenaga yang di gunakan untuk membawa koper.

## **Rumusan Masalah.**

Perumusan masalah proyek akhir yang akan dibahas adalah :

1. Desain alat yang baik sehingga dapat berintegrasi secara realtime.
2. Membuat para pengguna nyaman dikarenakan tidak menarik / mendorong koper

.

## **1.5 Batasan Masalah.**

Batasan masalah ini akan dibatasi pada hal-hal berikut :

1. Jarak pengontrolan tidak lebih dari 10 meter.
2. Alat ini hanya dapat dijalankan dengan sarung tangan yang sudah di konfigurasi terlebih dahulu.
3. Alat ini hanya sanggup menahan beban 4 kg.

# **BAB II**

# **DASAR TEORI**

## **2.1. Pengenalan Hardware**

### 2.1.1 Arduino Uno

Arduino adalah pengendali mikro single-board yang bersifat *open-source* diturunkan dari *Wiring Platform*  yang dirancang untuk memudahkan pengguna elektronik dalam berbagai bidang. Arduino juga merupakan *platform hardware* terbuka berdasarkan hardware dan software yang fleksibel dan mudah digunakan.

Arduino mungkin terdengar cukup berbeda dengan perangkat elektronik yang pernah ada selama ini, seperti LED (led emitting diode), atau resistor atau transistor bahkan mikrokontroler seperti AVR atau PIC atau yang lainnya.

Arduino uno memiliki 14 pin input dari output digital dimana 6 input tersebut dapat digunakan sebagai output PVM dan 6 pin analog, 16 MHz osilator kristal koneksi USB jack power, ICSP header, dan tombol reset.



**Gambar 2.1** *Arduino Uno R3*

(Sumber [*https://www.arduino.cc/en/Main/Products*)](https://www.arduino.cc/en/Main/Products))

### 2.1.2 Arduino Nano

Arduino Nano adalah salah satu papan pengembangan mikrokontroler yang berukuran kecil, lengkap dan mendukung penggunaan breadboard. Arduino Nano diciptakan dengan basis mikrokontroler ATmega328 (untuk Arduino Nano versi 3.x) atau ATmega 168 (untuk Arduino versi 2.x).

Arduino Nano kurang lebih memiliki fungsi yang sama dengan Arduino Duemilanove, tetapi dalam paket yang berbeda. Arduino Nano tidak menyertakan colokan DC berjenis Barrel Jack, dan dihubungkan ke komputer menggunakan port USB Mini-B. Arduino Nano dirancang dan diproduksi oleh perusahaan Gravitech. Arduino Nano dapat diaktifkan melalui koneksi USB Mini-B, atau melalui catu daya eksternal dengan tegangan belum teregulasi antara 6-20 Volt yang dihubungkan melalui pin 30 atau pin VIN, atau melalui catu daya eksternal dengan tegangan teregulasi 5 volt melalui pin 27 atau pin 5V. Sumber daya akan secara otomatis dipilih dari sumber tegangan yang lebih tinggi.

Chip FTDI FT232L pada Arduino Nano akan aktif apabila memperoleh daya melalui USB, ketika Arduino Nano diberikan daya dari luar (Non-USB) maka Chip FTDI tidak aktif dan pin 3.3V pun tidak tersedia (tidak mengeluarkan tegangan), sedangkan

LED TX dan RX pun berkedip apabila pin digital 0 dan 1 berada pada posisi HIGH.

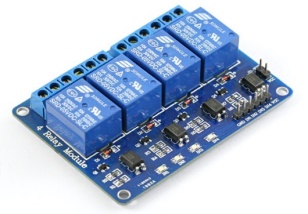


**Gambar 2.2** Arduino Nano  
(Sumber <https://store.arduino.cc/usa/arduino-nano>)

### 2.1.3 Relay 4 Channel 5V

Relay adalah Saklar (Switch) yang dioperasikan secara listrik dan merupakan komponen Electromechanical (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (Coil) dan Mekanikal (seperangkat Kontak 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 menggerakan Armature Relay (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A.

Kegunaan relay dalam dunia elektronika sebenarnya juga sama seperti dalam teknik listrik. Hanya saja kebanyakan relay yang digunakan dalam teknik elektronik adalah relay dengan voltase kecil seperti 6 Volt, 12 Volt, 24 Volt berbeda dengan teknik listrik yang memakai relay 220 Volt dan 110 Volt. Namun ada juga dalam teknik elektronik yang memakai relay dengan voltase tinggi. Walau ada perbedaan pemakaian voltase pada relay, sebenarnya relay memiliki fungsi atau kegunaan yang sama yakni sebagai alat pengganti saklar yang bekerja untuk mengontrol atau membagi arus listrik ataupun sinyal lain ke sirkuit rangkaian lainnya.



**Gambar 2.3** Accelerometer Relay 4 Channel 5V  
(Sumber <https://teknikelektronika.com/pengertian-relay-fungsi-relay/>)

### 2.1.4 Accelerometer

Accelerometer adalah alat sensor untuk mengukur dan menganalisa keadaan suatu perangkat yang kemudian dapat merespon dan bertindak secara otomatis. Sensor tersebut mampu menganalisa data dengan presisi dan akurasi tinggi, berguna untuk memantau pergerakan perangkat secara tiga dimensi atau posisi, atau memantau perubahan lingkungan dekat perangkat.

Sesuai namanya accelerometer atau akselerasi ini mengukur percepatan bahwa perangkat mengalami perubahan yang relatif sesuai dengan tiga sumbu XYZ atau kanan, kiri, atas, bawah, dan datar. Burton McCollum dan Orville Peters pertama kali merancang accelerometer komersial yang dokumentasi bisa ditemukan. Itu terdiri dari kerangka E-Shape mengandung  cakram karbon. Saat terkena percepatan, bagian atas E akan menuju ke kompresi dan bagian bawah akan masuk ke ketegangan. Bagian-bagian ini diatur dalam  konfigurasi jembatan setengah Wheatstone. Dokumentasi kinerja dan aplikasi ditulis pada tahun 1923. Itu  digunakan di Jerman dan dikomersialisasikan di Amerika Serikat di  1927 oleh Southwark, kemudian Baldwin-Southwark, dan sekarang BLH  Elektronik. Frekuensi resonansi berkisar antara 250 Hz.

Prinsip kerja dari tranduser ini berdasarkan hukum fisika bahwa apabila suatu konduktor digerakkan melalui suatu medan magnet, atau jika suatu medan magnet digerakkan melalui suatu konduktor, maka akan timbul suatu tegangan induksi pada konduktor tersebut. Accelerometer yang diletakan di permukaan bumi dapat mendeteksi percepatan 1g (ukuran gravitasi bumi) pada titik vertikalnya, untuk percepatan yang dikarenakan oleh pergerakan horizontal maka accelerometer akan mengukur percepatannya secara langsung ketika bergerak secara horizontal.



**Gambar 2.4** Accelerometer(Sumber [*http://www.instructables.com/id/AT-command-mode-of-HC-05-Bluetooth-module/*)](http://www.instructables.com/id/AT-command-mode-of-HC-05-Bluetooth-module/))

### 2.1.5 Kabel Jumper

Kabel jumper atau kabel penghubung tidak lepas perlengkapan untuk uji coba rangkaian di papan trainer (*BreadBoard*) warna putih yang terdiri dari banyak lubang komponen yang terhubungkan perkolom dan perbaris. Kabel penghubung (*jumper wired*) yang digunakan di dalam membuat rangkaian biasanya memiliki port yang terbuat dari plastik berwarna hitam yang memungkinkan kawat kabel penghubung saja yang terhubung langsung ke papan.



**Gambar 2.5** Kabel Jumper

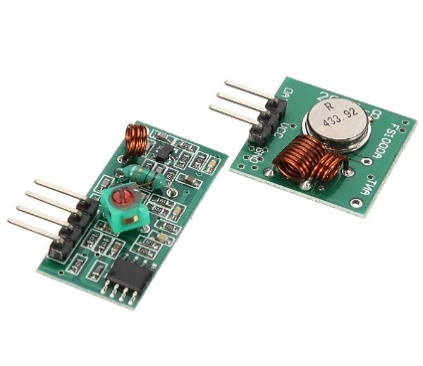
(Sumber *https://www.caratekno.com/2015/09/cara-membuat-kabel-jumper-untuk-arduino-. html*)

### 2.1.6 Modul RF Rx/Tx 433 MHz

Penggunaan modul RF 433MHz sebagai alternatif pengganti bluetooth, maupun module NRF240+ yang sebelumnya pernah dibahas. Penggunaan modul RF (Radio Frequency) menjadi alternatif lain yang lebih efisien dan murah untuk transmisi data. Komunikasi nirkabel yaitu komunikasi yang dimana antara transceiver dan reciever tidak perlu kabel sebagai sarana komunikasinya.

Paket 433Mhz RF link kit terdiri dari pemancar (Tx) dan penerima (Rx), yang secara umum digunakan untuk remote control.

Jenis atau model WLS107B4B, dengan berat sekitar 5 g, dengan frekuensi sebesar 433Mhz, Modulasi ASK, keluaran data penerima : tinggi - 1/2 Vcc, rendah - 0.7v, Tegangan masukkan transmisi : 3-12V (semakin tegangan masukannya tinggi maka kekuatan transmisi juga lebih baik, Tegangan masukkan penerima : 3.3-6V (semakin tegangan masukannya tinggi maka kekuatan penerimaan juga semakin baik).

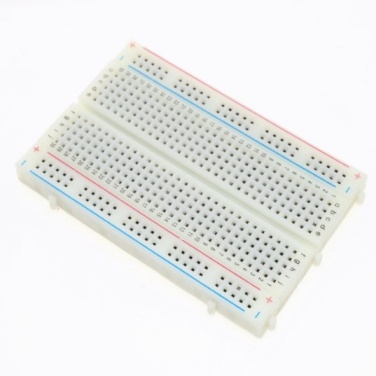


**Gambar 2.6**  Modul RF Rx/Tx 433 MHz

(Sumber <https://www.instructables.com/id/RF-315433-MHz-Transmitter-receiver-Module-and-Ardu/>)

### 2.1.7 Projct Board / BreadBoard

Breadboard adalah dasar kontruksi sebuah sirkuit elektronik dan merupakan *prototype* dari suatu rangkaian elektronik, Breadboard bisa diganti dengan menggunakan papan PCB (*Printed circuit board*). Breadboard dapat digunakan untuk merangkai komponen karena pengunaan breadboard tidak menggunakan proses penyolder karena papan breadboard dapat langsung ditancap dengan adanya kabel jumper dan bisa langsung terhubung dengan mikrokontroller yang ingin kita gunakan.



**Gambar 2.7** Breadboard  
(Sumber <http://www.robotedukasi.com/mengenal-papan-proyek-projectboard/>)

### 2.1.8 Motor DC

Motor DC adalah jenis motor listrik yang berkerja menggunakan sumber tengangan arus yang searah atau arus DC (*Dirrect Current*) pada kumpalan medan untuk diubah menjadi energi mekanik. Kumparan medan pada motor tersebut disebut stator, dan kumparan jangkar tersebut rotor.

Motor DC memiki 3 bagian atau komponen utama untuk dapat berputar yaitu adalah dengan adanya Kutub Medan, *Dinamo*, *Commutator*. Keuntungan dalam penggunaan motor DC ialah pengendali kecepatan yang tidak mempengaruhi kualitas pasokan daya.



**Gambar 2.8** Motor DC

(Sumber *http://scdc.binus.ac.id/himtek/2017/05/08/motor-dc-dan-jenis-jenisnya/*

# **BAB III**

# **PERANCANGAN SISTEM**

Bab ini akan menjelaskan mengenai perancangan sistem aplikasi yang akan dibuat berdasarkan teori yang telah dijelaskan pada bab sebelumnya, Sistem aplikasi ini disebut “Rancang Bangun Koper Smart Gesture Bebasis Arduino”.

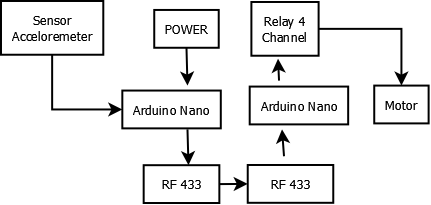
## **3.1 Analisis Sistem**

Berdasarkan latar belakang yang telah dibahas pada bab satu, yaitu salah satu untuk memudahkan dalam penggerakan koper tersebut engan cara menggunakan *smart gesture* atau menggunakan sensor *accelerometer*, dan jaringan yang digunakan untuk menghubungkan sarung tangan dengan *slave* atau motor menggunakan *Module* RF Rx/Tx 433 MHz.

## **3.2 Blok Diagram**

Berikut ini adalah bentuk blok diagram dari Koper Smart Gesture yang terdiri dari arduino.

**Gambar 3.1** Blok Diagram



Fungsi dari block diagram di atas adalah sebagai berikut:

1. Sensor Acceloremeter

Untuk menentukan nilai x dan y .

2. Power

Untuk menghidupkan dan mematikan alat tersebut .

3. Arduino Nano

Berfungsi sebagai mikroprocesor.

4. RF 433

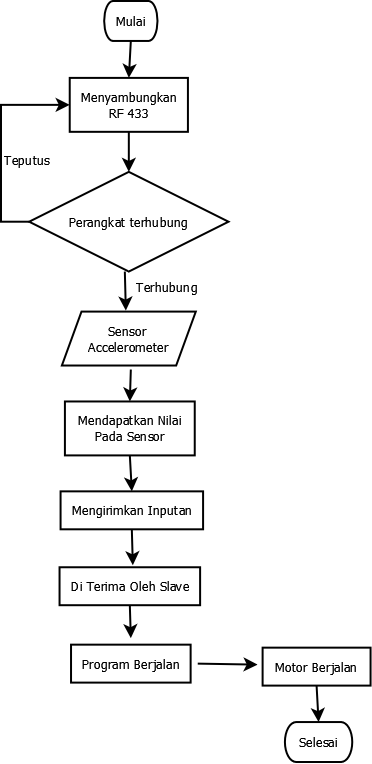
Sebagai penghubung antara koper dan glove.

5. Motor DC

Berfungsi sebagai pengerak roda pada Koper

Dari Blok Diagram diatas dapat dijelaskan prinsip kerja dari sistem ini adalah slave akan terhubung dengan arduino menggunakan *module* RF 433 mHz melalui pinout *TX* yang akan menuliskan *command* di monitoring kemudian arduino uno bekerja sesuai pemograman yang telah di input melalui program.

## **3.3 Desain Alur / Flowchart Koper.**



**Gambar 3.2** Diagram Alur Koper

**Keterangan Alur (FlowChart) Koper :**

1. Mulai

Master dan slave akan langsung terhubung.

1. Perangkat Terhubung

Berfungsi menyambungkan perangkat sarung tangan dengan koper.

1. Sensor Acceloremeter

Sensor ini akan mendapatkan nilai dari percepatan kemiringan

1. Mendapatkan nilai pada sensor.

Proses ini yaitu mendapatkan nilai pada sensor dan mengubah menjadi nilai integer berdasarkan program pada koper

1. Mengirimkan Nilai integer

Proses ini mengirimkan nilai integer dari master kepada slave melalui serial.write.

1. Diterima oleh slave atau koper

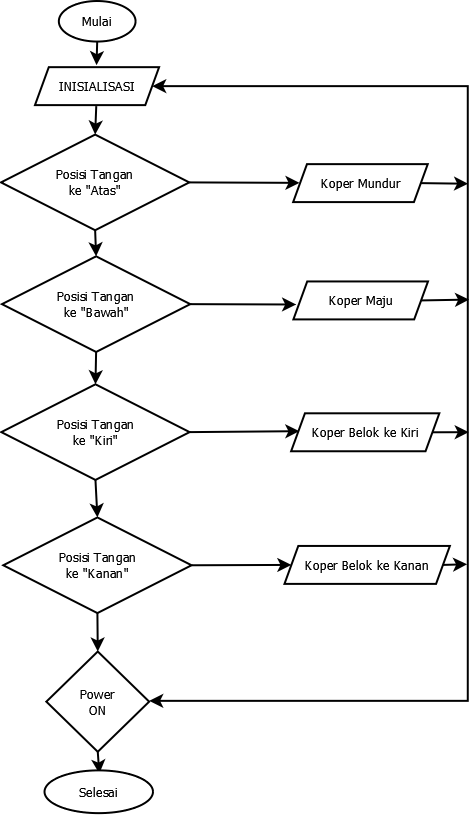
Proses ini untuk mendapatkan nilai integer dari koper melalui program serial read.

1. Program berjalan

Menjalankan program sesuai dengan input untuk menjalankan program dan mengirimkan *ouput* motor.

1. Program motor mundur
2. Apabila sensor terdeteksi makan program akan mengirimkan program mundur pada motor.
3. Motor berjalan

## **3.4. Desain Alir / Flowchart Glove**

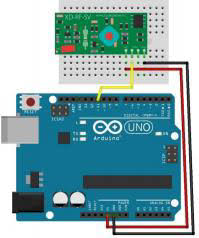
****

**Gambar 3.3** Diagram Alur Glove

**Keterangan Alir (FlowChart) Glove :**

1. Mulai
2. Pertama kita akan mulai menghidukan atau menghubungkan sarung tangan pengendali.
3. Posisi tangan mengarah ke atas
4. Jika tangan mengarah ke atas maka koper akan bergerak mundur kebelakang.
5. Posisi tangan kebawah
6. Jika tangan mengarah ke bawah maka koper akan bergerak maju kedepan
7. Posisi tangan ke kiri
8. Jika tangan mengarah ke kiri atau serong ke kiri maka koper akan berbelok arah ke kiri.
9. Posisi tangan serong ke kanan
10. Jika tangan mengarah ke kanan atau serong ke kanan maka koper akan berbelok arah ke kanan .

## **3.5 Rancangan Arduino RF Rx/Tx 433 MHz**

****

**2**

**3**

**1**

**Gambar 3.4** Rancangan Arduino *Module* RF Rx/Tx 433 MHz

Keterangan gambar :

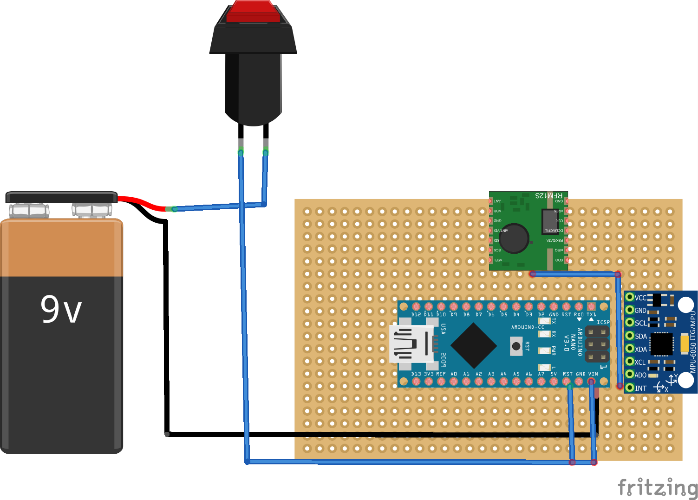
1. RF 433
2. Breadboard
3. Arduino Uno

Penulis menghubungkan arduino dengan menggunakan kabel jumper ke *module* RF Rx/Tx 433 Mhz dengan menghubungkan arus VCC pada 3.3V-6V pada arduino, Chip Atmega328 menyediakan komunikasi serial UART TTL (5V) yang tersedia pada pin 0 (RX) dan pin 1 (TX), RXD pada *module*  terhubung langsung dengan TX yang berfungsi untuk receiver data. TXD terhubung dengan RX pada arduino yang berfungsi untuk mengirimkan data.

## **3.6 Rancangan Master Arduino Nano.**

Penulis menghubungkan arduino nano untuk mengirimkan perintah melalui jaringan *wifi* untuk mengirimkan nilai *value* melalui serial monitor dan akan diterima melalui serial monitor dan kemudian akan memerintah berdasarkan *value*.

Dengan menggunakan pin TX dan RX untuk mengirimkan perintah tersebut, maka *module* RF Rx/Tx 433 MHz dapat langsung berinteraksi dengan *slave* tersebut.



**5**

**1**

**3**

**4**

**2**

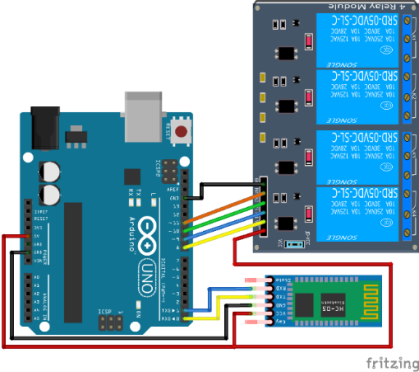
**Gambar 3.5** Rancangan Master Arduino Nano

Keterangan gambar :

1. Baterai 9v
2. Push Button
3. RF 433
4. Accelerometer
5. Arduino Nano

## **3.7 Rancangan Master Arduino Uno**

Penulis menghubungkan *arduino* *nano* yang terkoneksi jaringan *RF 433* untuk menerima perintah *value* melalui *serial begin 38400 (Serial Monitoring)* untuk menjalankan program yang dibuat untuk menjalankan koper.



**1**

**3**

**Gambar 3.6**. Rancangan Arduino Uno.

Keterangan Gambar :

1. Arduino Uno
2. Relay 4 Channel
3. RF 433

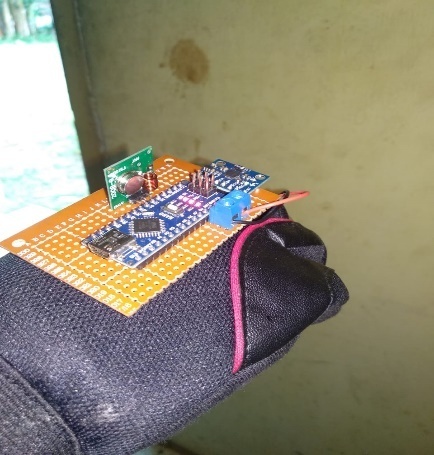
# **BAB IV**

# **HASIL**

Pada bab ini menjelaskan hasil dan *project* yang telah dibuat yaitu berupa sarung tangan *smart gesture,* dan koper untuk *output* yang dihasilkan dari sensor *accelerometer.*

## **4.1 Rancangan Smart Gesture**

Gambar dibawah ini merupakan rangkaian smart gesture yang mengambil data dari sensor accelerometer untuk mengubah menjadi value untuk dapat dikirimkan ke koper untuk dapat mengeluarkan output. Mengirim value tersebut menggunakan Serial.write.



**1a**

**2**

**Gambar 4.1** Rancangan Smart Gesutre

Keterangan gambar :

1. Sensor Accelerometer.
2. Arduino Nano & RF 433.

## **4.2. Rancangan Koper**

Gambar dibawah ini adalah rangkaian koper yang berfungsi menggerakkan motor dengan cara menerima input dari *smart gesture*  dan di olah menjadi perintah untuk menjalankan motor sesuai dengan program, pada rangkaian koper terdapat Arduino uno sebagai *microkontroler,* modul *RF 433* sebagai penghubung antara *smart gesture* dengan koper agar tidak terjadi hal yang tidak diinginkan dan terdapat *Module Relay* 4 *Chennel* yang berfungsi untuk mengaliri arus pada motor.



**4**

**3**

**2**

**1**

**Gambar 4.2** Lampiran Program Koper

Keterangan gambar :

1. Motor DC 775 12 V.
2. Relay 4 Channel.
3. Arduino UNO.
4. RF 433.

## **4.3 Hasil Analisa Sistem**

Pada pengujian terhadap alat yang dilakukan, dengan cara menambahkan beban pada koper untuk menguji seberapa besar berat yang sanggup ditahan oleh koper. Pengujian ini berfungsi untuk memaksimalkan dalam penggunaan koper.

Tabel 4.1 Tabel Pengujian

|  |  |  |
| --- | --- | --- |
| No | Pengujian | Hasil |
| 1 | Beban 1 Kg | Koper berjalan sempurna dengan kecepatan 2 km/h |
| 2 | Beban 2 Kg | Koper berjalan sempurna dengan kecepatan 1.5 km/h |
| 3 | Beban 3 Kg | Koper berjalan sempurna dengan kecepatan 1 km/h |
| 4 | Beban 4 Kg | Koper berjalan sempurna dengan kecepatan 0,5 km/h |
| 5 | Beban 5 Kg | Koper tidak berjalan |

Pada pengujian diatas menjelaskan dengan bobot beban pengujian menunjukan bahwa koper sanggup menahan beban sebesar 3 kg dengan kecepatan 1 km/h dengan posisi baterai penuh, dan sanggup berjalan sekitar 35 menit untuk sekali pengisian baterai.

Tabet 4.2 Tabel Kondisi

|  |  |  |
| --- | --- | --- |
| No | Kondisi | Output |
| 1. | Maju | Koper akan berjalan lurus |
| 2. | Kiri | Koper akan berjalan ke kiri dengan cara mematikan satu buah motor belakang sebalah kanan |
| 3. | Kanan | Koper akan berbelok ke kanan dengan cara mematikan satu motor belakan sebelah kiri |
| 4. | Mundur | Koper akan bergerak mundur |
| 5 | Apabila koper dalam keadaan diam | Koper tidak akan merespon apa-apa |

Pada tabel diatas menjelaskan kondisi, saat koper mengeluarkan *output* yaitu gerakan motor.

# **BAB V**

# **PENUTUP**

## **5.1. Kesimpulan.**

Berdasarkan hasil pembuatan *smart gesture* pada perbelanjaan dapat diambil kesimpulan sebagai berikut :

1. *Smart gesture* ini dapat mempermudah orang dalam pemakain koper.
2. Penggunaan *Smart gesture* dapat menghemat tenaga dalam penggunaan koper.

## **5.2. Saran.**

Sesuai dengan kesimpulan yang telah dipaparkan diatas, saran yang dapat diberikan yaitu :

1. Bagi pembaca disarankan untuk membuat alat ini dapat jalan dengan lurus tanpa berputar.
2. Alat ini disarankan untuk menggunakan PWM (Pulse Width Modulation) yang dapat mengukur kecepatan koper.

# **DAFTAR PUSTAKA**

1. Adi ,Agung Nughroho, 2010. Mekatronika, Yogyakarta : graha Ilmu
2. Suyadhi, T.D.S, 2010. “Buku Pintar Robotika: Bagaimana Merancang dan Membuat Robot Sendiri”.
3. Djuandi, Feri, 2011. “Pengenalan Arduino”. Jakarta : Penerbit Elexmedia.
4. https://create.arduino.cc/prjecthub?f=1

# **Lampiran 1. Program Grove.**

|  |
| --- |
| #include <RH\_ASK.h> //library  #include <SPI.h> // Not actually used but needed to compile  RH\_ASK driver**;**  // I2Cdev and MPU6050 must be installed as libraries, or else the .cpp/.h files  // for both classes must be in the include path of your project  #include "I2Cdev.h"  #include "MPU6050.h"  // Arduino Wire library is required if I2Cdev I2CDEV\_ARDUINO\_WIRE implementation  // is used in I2Cdev.h  #if I2CDEV\_IMPLEMENTATION == I2CDEV\_ARDUINO\_WIRE  #include "Wire.h"  #endif  // class default I2C address is 0x68  // specific I2C addresses may be passed as a parameter here  // AD0 low = 0x68 (default for InvenSense evaluation board)  // AD0 high = 0x69  MPU6050 accelgyro**;**  //MPU6050 accelgyro(0x69); // <-- use for AD0 high  int16\_t ax**,** ay**,** az**;** //variabel  int16\_t gx**,** gy**,** gz**;**  // uncomment "OUTPUT\_READABLE\_ACCELGYRO" if you want to see a tab-separated  // list of the accel X/Y/Z and then gyro X/Y/Z values in decimal. Easy to read,  // not so easy to parse, and slow(er) over UART.  #define OUTPUT\_READABLE\_ACCELGYRO  // uncomment "OUTPUT\_BINARY\_ACCELGYRO" to send all 6 axes of data as 16-bit  // binary, one right after the other. This is very fast (as fast as possible  // without compression or data loss), and easy to parse, but impossible to read  // for a human.  //#define OUTPUT\_BINARY\_ACCELGYRO  #define LED\_PIN 13  bool blinkState **=** false**;**  void setup**()** **{**  // join I2C bus (I2Cdev library doesn't do this automatically)  **if** **(!**driver**.**init**())** //untuk pengecekan koneksi anttara glove dan koper  Serial**.**println**(**"init failed"**);**  #if I2CDEV\_IMPLEMENTATION == I2CDEV\_ARDUINO\_WIRE  Wire**.**begin**();**  #elif I2CDEV\_IMPLEMENTATION == I2CDEV\_BUILTIN\_FASTWIRE  Fastwire**::**setup**(**400**,** true**);**  #endif  // initialize serial communication  // (38400 chosen because it works as well at 8MHz as it does at 16MHz, but  // it's really up to you depending on your project)  Serial**.**begin**(**38400**);**  // initialize device  Serial**.**println**(**"Initializing I2C devices..."**);**  accelgyro**.**initialize**();**  // verify connection  Serial**.**println**(**"Testing device connections..."**);**  Serial**.**println**(**accelgyro**.**testConnection**()** **?** "MPU6050 connection successful" **:** "MPU6050 connection failed"**);**  // use the code below to change accel/gyro offset values    // configure Arduino LED pin for output  pinMode**(**LED\_PIN**,** OUTPUT**);**  **}**  void loop**()** **{**  // read raw accel/gyro measurements from device  accelgyro**.**getMotion6**(&**ax**,** **&**ay**,** **&**az**,** **&**gx**,** **&**gy**,** **&**gz**);** //untuk mengambil nilai dari sensor acceloremeter  // these methods (and a few others) are also available  //accelgyro.getAcceleration(&ax, &ay, &az);  //accelgyro.getRotation(&gx, &gy, &gz);  #ifdef OUTPUT\_READABLE\_ACCELGYRO  // display tab-separated accel/gyro x/y/z values  Serial**.**print**(**"a/g:\t"**);**  Serial**.**print**(**ax**);** Serial**.**print**(**"\t"**);**  Serial**.**print**(**ay**);** Serial**.**print**(**"\t"**);**  Serial**.**print**(**az**);** Serial**.**print**(**"\t"**);**  Serial**.**print**(**gx**);** Serial**.**print**(**"\t"**);**  Serial**.**print**(**gy**);** Serial**.**print**(**"\t"**);**  Serial**.**println**(**gz**);**  #endif  #ifdef OUTPUT\_BINARY\_ACCELGYRO //untuk mengeririm data dari glove ke koper  Serial**.**write**((**uint8\_t**)(**ax **>>** 8**));** Serial**.**write**((**uint8\_t**)(**ax **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**ay **>>** 8**));** Serial**.**write**((**uint8\_t**)(**ay **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**az **>>** 8**));** Serial**.**write**((**uint8\_t**)(**az **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**gx **>>** 8**));** Serial**.**write**((**uint8\_t**)(**gx **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**gy **>>** 8**));** Serial**.**write**((**uint8\_t**)(**gy **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**gz **>>** 8**));** Serial**.**write**((**uint8\_t**)(**gz **&** 0xFF**));**  #endif  // blink LED to indicate activity  blinkState **=** **!**blinkState**;**  digitalWrite**(**LED\_PIN**,** blinkState**);**  **if** **(**ay**>**8000**){**  const char **\***msg **=** "kanan"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**//untuk mengirim perintah kanan  driver**.**waitPacketSent**();**  **}**  **else** **if** **(**ay **<-**5000 **){**  const char **\***msg **=** "kiri"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**  driver**.**waitPacketSent**();**  **}**  **else** **if** **(**ax **<-**4000 **){**  const char **\***msg **=** "mundur"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**  driver**.**waitPacketSent**();**  **}if** **(**ax**>**4000**){**  const char **\***msg **=** "maju"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**  driver**.**waitPacketSent**();**  **}**  **else** **if** **(**ax**>-**1000 **&&** ax**<**1000 **&&** ay**>-**2000 **&&** ay**<**2000**){**  const char **\***msg **=** "diam"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**  driver**.**waitPacketSent**();**  **}**  **}** |

# **Lampiran 2. Program Koper.**

|  |
| --- |
| #include <RH\_ASK.h> //library  #include <SPI.h> // Not actually used but needed to compile  RH\_ASK driver**;**  // I2Cdev and MPU6050 must be installed as libraries, or else the .cpp/.h files  // for both classes must be in the include path of your project  #include "I2Cdev.h"  #include "MPU6050.h"  // Arduino Wire library is required if I2Cdev I2CDEV\_ARDUINO\_WIRE implementation  // is used in I2Cdev.h  #if I2CDEV\_IMPLEMENTATION == I2CDEV\_ARDUINO\_WIRE  #include "Wire.h"  #endif  // class default I2C address is 0x68  // specific I2C addresses may be passed as a parameter here  // AD0 low = 0x68 (default for InvenSense evaluation board)  // AD0 high = 0x69  MPU6050 accelgyro**;**  //MPU6050 accelgyro(0x69); // <-- use for AD0 high  int16\_t ax**,** ay**,** az**;** //variabel  int16\_t gx**,** gy**,** gz**;**  // uncomment "OUTPUT\_READABLE\_ACCELGYRO" if you want to see a tab-separated  // list of the accel X/Y/Z and then gyro X/Y/Z values in decimal. Easy to read,  // not so easy to parse, and slow(er) over UART.  #define OUTPUT\_READABLE\_ACCELGYRO  // uncomment "OUTPUT\_BINARY\_ACCELGYRO" to send all 6 axes of data as 16-bit  // binary, one right after the other. This is very fast (as fast as possible  // without compression or data loss), and easy to parse, but impossible to read  // for a human.  //#define OUTPUT\_BINARY\_ACCELGYRO  #define LED\_PIN 13  bool blinkState **=** false**;**  void setup**()** **{**  // join I2C bus (I2Cdev library doesn't do this automatically)  **if** **(!**driver**.**init**())** //untuk pengecekan koneksi anttara glove dan koper  Serial**.**println**(**"init failed"**);**  #if I2CDEV\_IMPLEMENTATION == I2CDEV\_ARDUINO\_WIRE  Wire**.**begin**();**  #elif I2CDEV\_IMPLEMENTATION == I2CDEV\_BUILTIN\_FASTWIRE  Fastwire**::**setup**(**400**,** true**);**  #endif  // initialize serial communication  // (38400 chosen because it works as well at 8MHz as it does at 16MHz, but  // it's really up to you depending on your project)  Serial**.**begin**(**38400**);**  // initialize device  Serial**.**println**(**"Initializing I2C devices..."**);**  accelgyro**.**initialize**();**  // verify connection  Serial**.**println**(**"Testing device connections..."**);**  Serial**.**println**(**accelgyro**.**testConnection**()** **?** "MPU6050 connection successful" **:** "MPU6050 connection failed"**);**  // use the code below to change accel/gyro offset values    // configure Arduino LED pin for output  pinMode**(**LED\_PIN**,** OUTPUT**);**  **}**  void loop**()** **{**  // read raw accel/gyro measurements from device  accelgyro**.**getMotion6**(&**ax**,** **&**ay**,** **&**az**,** **&**gx**,** **&**gy**,** **&**gz**);** //untuk mengambil nilai dari sensor acceloremeter  // these methods (and a few others) are also available  //accelgyro.getAcceleration(&ax, &ay, &az);  //accelgyro.getRotation(&gx, &gy, &gz);  #ifdef OUTPUT\_READABLE\_ACCELGYRO  // display tab-separated accel/gyro x/y/z values  Serial**.**print**(**"a/g:\t"**);**  Serial**.**print**(**ax**);** Serial**.**print**(**"\t"**);**  Serial**.**print**(**ay**);** Serial**.**print**(**"\t"**);**  Serial**.**print**(**az**);** Serial**.**print**(**"\t"**);**  Serial**.**print**(**gx**);** Serial**.**print**(**"\t"**);**  Serial**.**print**(**gy**);** Serial**.**print**(**"\t"**);**  Serial**.**println**(**gz**);**  #endif  #ifdef OUTPUT\_BINARY\_ACCELGYRO //untuk mengeririm data dari glove ke koper  Serial**.**write**((**uin.t8\_t**)(**ax **>>** 8**));** Serial**.**write**((**uint8\_t**)(**ax **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**ay **>>** 8**));** Serial**.**write**((**uint8\_t**)(**ay **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**az **>>** 8**));** Serial**.**write**((**uint8\_t**)(**az **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**gx **>>** 8**));** Serial**.**write**((**uint8\_t**)(**gx **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**gy **>>** 8**));** Serial**.**write**((**uint8\_t**)(**gy **&** 0xFF**));**  Serial**.**write**((**uint8\_t**)(**gz **>>** 8**));** Serial**.**write**((**uint8\_t**)(**gz **&** 0xFF**));**  #endif  // blink LED to indicate activity  blinkState **=** **!**blinkState**;**  digitalWrite**(**LED\_PIN**,** blinkState**);**  **if** **(**ay**>**8000**){**  const char **\***msg **=** "kanan"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**//untuk mengirim perintah kanan  driver**.**waitPacketSent**();**  **}**  **else** **if** **(**ay **<-**5000 **){**  const char **\***msg **=** "kiri"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**  driver**.**waitPacketSent**();**  **}**  **else** **if** **(**ax **<-**4000 **){**  const char **\***msg **=** "mundur"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**  driver**.**waitPacketSent**();**  **}if** **(**ax**>**4000**){**  const char **\***msg **=** "maju"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**  driver**.**waitPacketSent**();**  **}**  **else** **if** **(**ax**>-**1000 **&&** ax**<**1000 **&&** ay**>-**2000 **&&** ay**<**2000**){**  const char **\***msg **=** "diam"**;**  driver**.**send**((**uint8\_t **\*)**msg**,** strlen**(**msg**));**  driver**.**waitPacketSent**();**  **}**  **}** |

# **BIODATA PENULIS**

Dedi Munandar lahir di Banda Aceh pada tanggal 09 Desember 1996. Penulis bertempat tinggal di Ateuk Pahlawan, Desa Labui, Kec. Baiturrahman, Kab.Banda Aceh. Hobi Penulis Sepak bola dan Gaming. Motto Hidup “Jangan menyerah”. Sekolah Dasar di selesaikan di Min Mesjid Raya Banda Aceh pada tahun 2003-2009. Sekolah menengah pertama di selesaikan pada SMP Negeri 3 Banda Aceh pada tahun 2009-2012. Sekolah Menengah Atas di selesaikan pada SMA Negeri 9 Banda Aceh. Penulis di terima di Politeknik Aceh pada program studi Teknik Informatika tahun 2015. Penulis Menyelesaikan studi D III Teknik Informatika Politeknik Aceh pada tahun 2018. Di bawah Bimbingan Safwan, S.T dan Feri Susilawati, S.ST Penulis mengerjakan proyek akhir yang berjudul “RANCANG BANGUN KOPER SMART GESTURE BERBASIS ARDUINO” Yang di gunakan sebagai syarat kelulusan dari politelnik aceh untuk mendapatkan gelar Ahli Madya.

Contact Person : 082274784311

Email : dedimunandar63@yahoo.co.id