

BAB III

METODE PENELITIAN

3.1. Model Pengembangan

Tujuan dari tugas akhir ini yaitu akan membuat sebuah *mobile robot* yang mampu membantu manusia dalam memarkirkan kendaraan pribadinya yaitu mobil. Robot ini mampu membawa mobil ketempat parkirann yang kosong hanya dengan menempelkan kartu RFID pada robot, ketika pemilik mobil menempelkan kartu RFID pada body robot yang sudah diberi RFID reader, robot akan mulai berjalan menghampiri mobil dan ketika mobil sudah ditemukan robot akan mengangkat mobil dan mengantarkan mobil ketempat parkir yang dituju dengan otomatis.

Pada robot ini terdapat sensor garis (Photodiaoda) dan sensor Rotary Encoder, disini sensor garis digunakan sebagai pembacaan jalur yang akan dilewati oleh robot agar robot berjalan pada lajur yang sudah disediakan dan sensor rotary encoder digunakan sebagai penanda titik tujuan dari robot.

3.2. Prosedur Penelitian

Prosedur penelitian yang dipakai dalam pengerjaan tugas akhir ini adalah:

1. Studi literatur

Pada penelitian ini terdapat dua perancangan yang akan dilakukan yaitu, perancangan perangkat keras dan perangkat lunak. Adapun metode penelitian yang dilakukan antara lain: Pencarian data-data literatur untuk perangkat keras dari masing-masing komponen, informasi dari internet dan konsep teoritis dari buku-buku penunjang tugas akhir ini, serta materi-materi perkuliahan yang telah

didapatkan dan perancangan perangkat lunak yaitu menggunakan CoIDE melalui pencarian dari internet, dan konsep-konsep teoritis dari buku-buku penunjang tersebut. Dari kedua bagian tersebut akan dipadukan agar dapat bekerja sama untuk menjalankan sistem dengan baik.

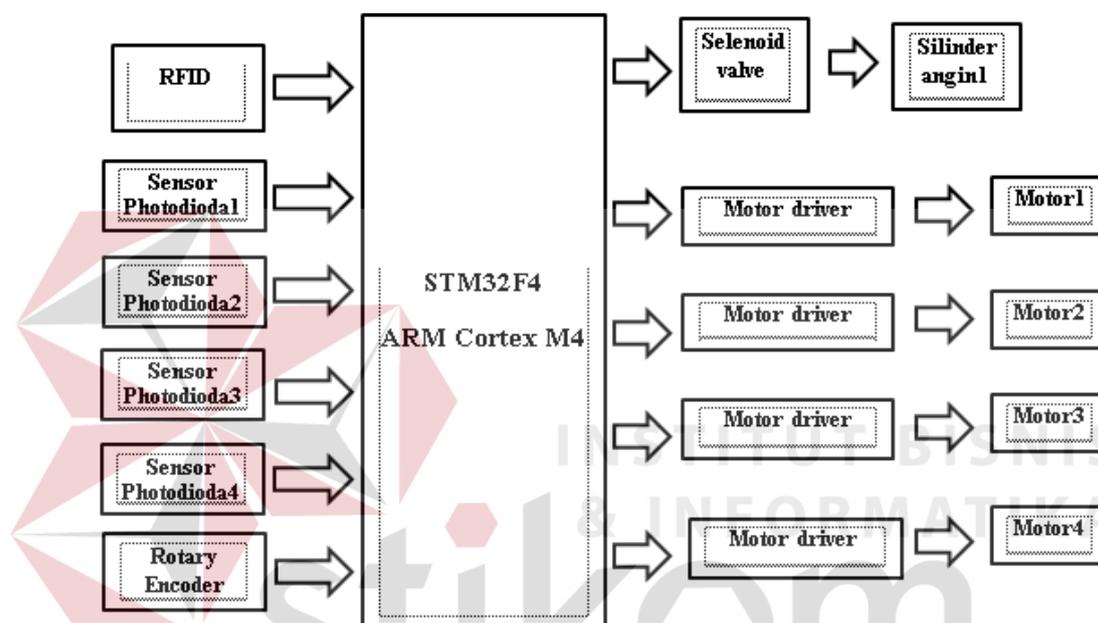
2. Tahap perancangan dan pengembangan sistem

Dalam membuat pengembangan sistem, terdapat beberapa langkah rancangan sistem yang diambil antara lain:

- a. Membuat *flowchart* pada proses sistem secara keseluruhan
- b. Melakukan perancangan perangkat keras yang meliputi:
 1. Merancang rangkaian elektronik yang digunakan pada penelitian ini
 2. Melakukan percobaan tentang cara penggunaan sensor dan *device* yang digunakan pada penelitian ini
 3. Merancang mekanik untuk *mobile robot*
- c. Melakukan perancangan perangkat lunak yang meliputi:
 1. Mengatur protokol pengiriman antara PC dan STM32F4.
 2. Membuat program USART pada STM32F4 agar bisa berkomunikasi dengan RFID Reader untuk pembacaan RFID card.
 3. Membuat program pembacaan garis menggunakan sensor garis.
 4. Membuat program pembacaan Rotary Encoder untuk penanda titik tujuan dari robot.

3.3. Diagram Blok Sistem

Dari penelitian ini terdapat dua proses utama yang akan dijalankan, yaitu proses dimana robot dapat membawa mobil ke tempat parkir yang kosong sesuai inputan yang didapat dari menempelkan RFID card pada RFID reader yang berada di robot. Pada gambar 3.3 adalah diagram blok keseluruhan sistem ini.



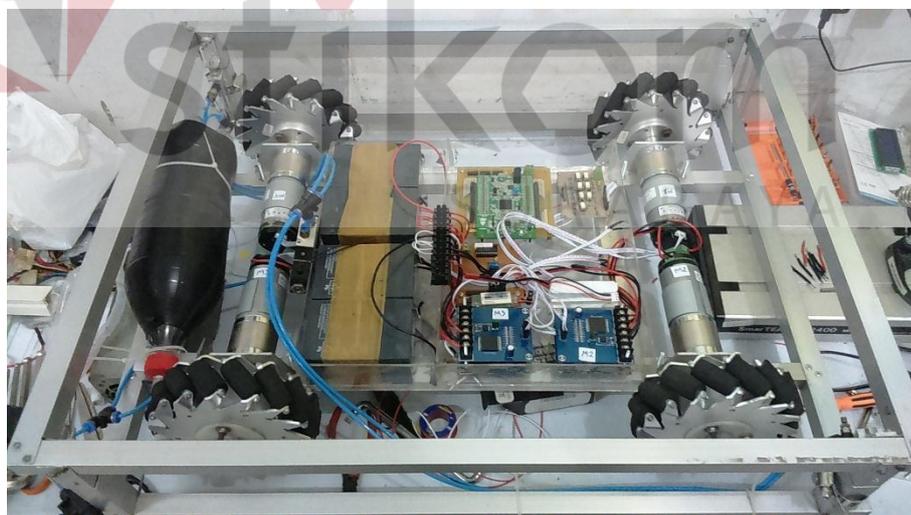
Gambar 3.3 Diagram blok keseluruhan dari system

Pertama – tama user harus menempelkan *chip RFID* ke *RFID Reader* , setelah itu modul akan mengirim instruksi ke ARM Cortex melalui komunikasi USART, setelah intruksi tersebut diterima oleh Arm Cortex M4, Arm Cortex M4 akan memberikan perintah kepada Motor Driver agar motor mulai berjalan, ketika motor berjalan sensor Phodiode akan memandu robot menuju posisi awal mobil dan sensor Rotary Encoder juga mulai membaca tempat titik tujuan robot yang sesuai dengan intruksi yang diterima, setelah robot sampai pada posisi dimana mobil tersebut sudah ditempatkan pemiliknya diarea pemberhentian yang sudah disediakan, motor akan berhenti, ketika motor sudah berhenti Arm Cortex M4

akan memberikan intruksi ke Selenoid Valve agar Silinder Angin mulai mengangkat mobil tersebut, Setelah mobil sudah terangkat motor akan kembali berjalan mengikuti garis dengan menggunakan sensor Phohodiada dan akan menuju posisi titik akhir yaitu tempat mobil itu diparkirkan dengan menggunakan sensor Rotary Encoder, setelah sampai dititik tujuan robot akan berhenti dan meletakkan mobil tersebut, setelah itu robot kembali keposisi awal.

3.4 Perancangan Mekanik Robot

Mekanik robot yang di gunakan adalah dari bahan aluminum persegi ukuran 70cm x 55cm x 20cm yang telah di desain dan dirancang khusus untuk kepentingan penelitian sistem robot ini. Robot ini di design sedemikian rupa agar seluruh elektronika bisa terpasang dengan baik pada robot tersebut, mulai dari rangkaian *STM32F4 Arm CortexM4*, *driver motor Ems 30A H-Bridge*, Sensor Photodiada, sensor Rotary Encoder, RFID Reader.



Gambar 3.4. Tampilan Keseluruhan Robot

3.4.1. Ukuran dimensi *Mobile robot*

Setelah semua komponen tambahan dari penelitian ini dipasangkan ukuran dimensi dari robot:

Ukuran robot : 70 cm (panjang) x 55cm (lebar) x 20cm (tinggi)

3.4.2. Struktur Material *Mobile robot*

Bahan material yang digunakan dalam penelitian ini menggunakan beberapa bahan diantaranya sebagai berikut :

a. Bagian rangka

1. Alumunium
2. Mur dan baut
3. Akrilik

b. Bagian dari penggerak robot

1. Motor DC 24Volt
2. Roda Mecanum
3. Pneumatic valve
4. Pneumatic Cylinder

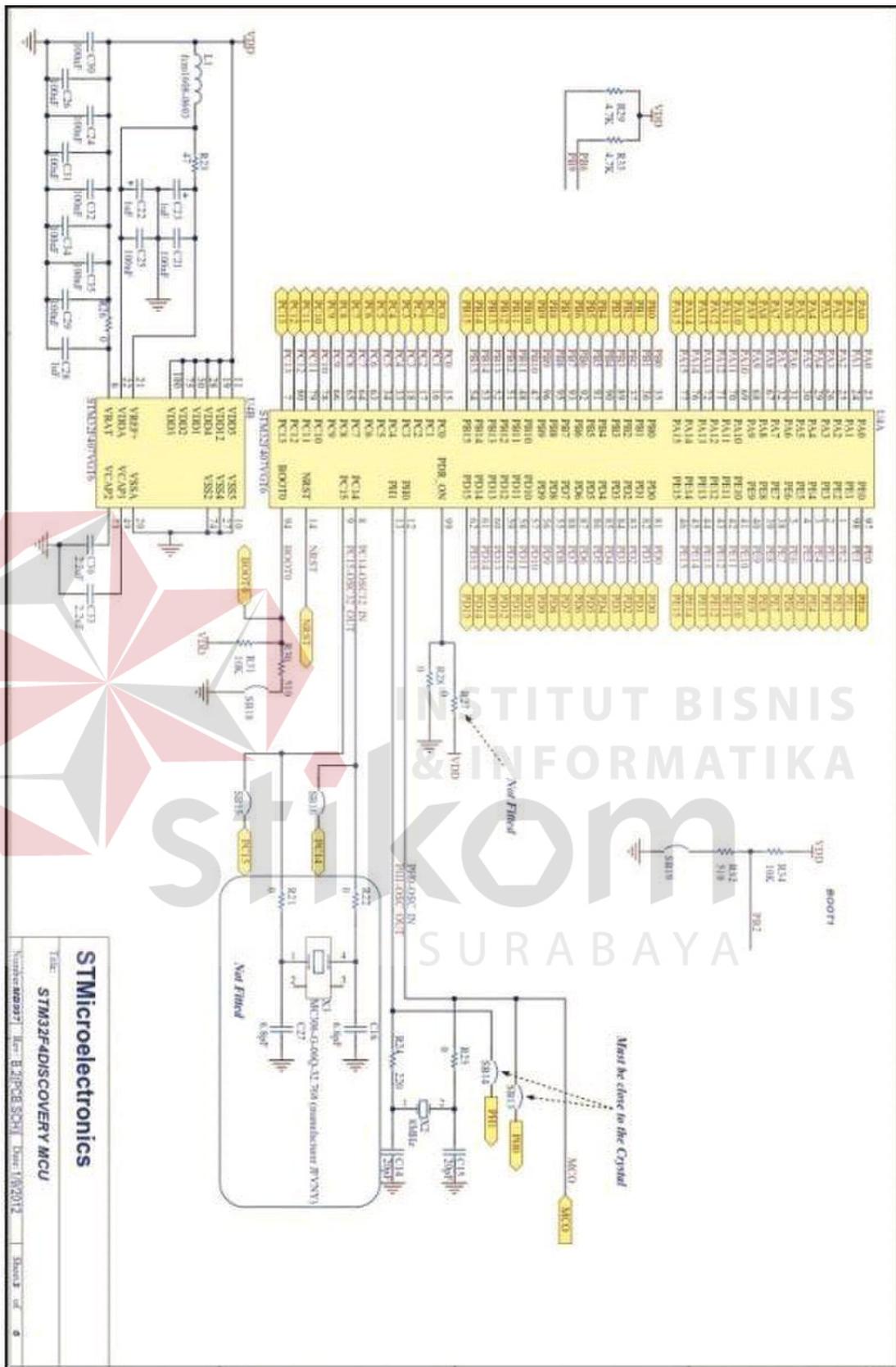
3.4.3 Perancangan *Minimum System Microcontroller ARM Cortex M-4 (STM32F4 DISCOVERY)*

Rangkaian *minimum system* dibuat untuk mendukung kerja dari *microchip* STM32F407VTG dimana *microchip* tidak bisa berdiri sendiri alias harus ada rangkaian dan komponen pendukung seperti halnya rangkaian catu daya, kristal dan lain sebagainya yang biasanya disebut minimum sistem *Microchip* berfungsi sebagai otak dalam mengolah semua instruksi baik *input* maupun *output* seperti halnya pemroses data inputan dari *rotary encoder* dan *sensor photodiode* kemudian memproses data yang diterima dari RFID reader kemudian menjalankan perintah untuk menjalankan motor.

Minimum system ini dirancang untuk *microcontroller ARM Cortex M-4* yang menggunakan chip STM32F407VTG, dalam perancangannya ini memerlukan

n beberapa komponen pendukung seperti kristal, resistor dan variabel resistor, dan kapasitor. Rangkaian ini dalam istilah lainnya disebut *Minimum system ARM Cortex M-4 (STM32F4 DISCOVERY)*. *Microcontroller ARM Cortex M-4 (STM32F4 DISCOVERY)* berfungsi untuk memproses data inputan dari *rotary encoder* dan sensor photodiode sebagai mengeksekusi data dengan mengeluarkan output berupa perintah menjalankan 4 buah motor pada bagian *base robot* yang menggunakan *mecanum wheel*. Berikut ini gambar *Minimum system ARM Cortex M-4 (STM32F4 DISCOVERY)* dapat dilihat pada Gambar 3.5.

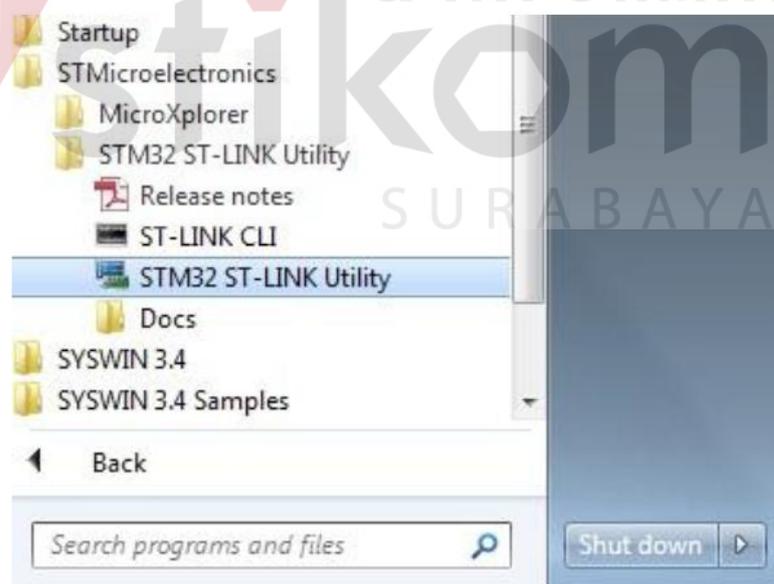




Pada rangkaian minimum system ARM Cortex M-4 (STM32F4 DISCOVERY) memberikan Pin VCC masukan tegangan operasi berkisar antara 2,8 Volt sampai dengan 3,2 Volt. Pin NRST berfungsi untuk masukan reset program secara otomatis atau manual. Sedangkan pin MC0 digunakan untuk keperluan pemrograman microcontroller ARM Cortex M-4 (STM32F4 DISCOVERY). Frekuensi kristal yang dipakai adalah 168 MHz maximum frequency, 210 DMIPS/1.25 DMIPS/MHz.

3.4.4 ST-Link Downloader

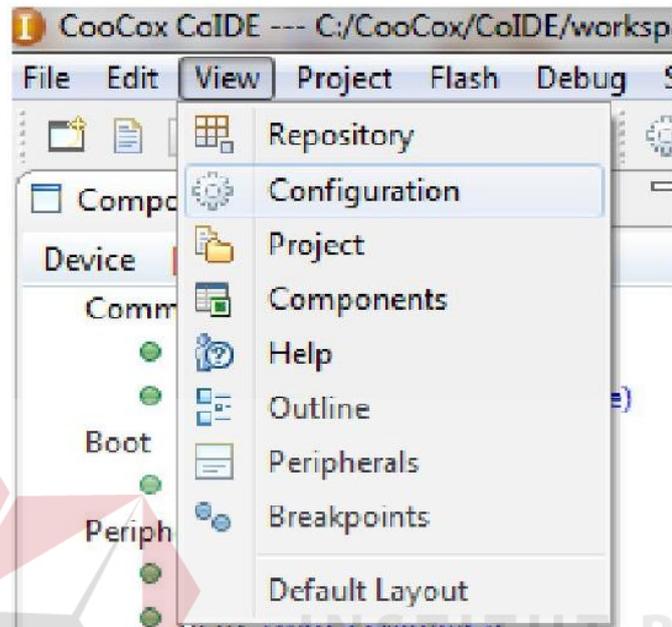
Untuk melakukan proses download program, yaitu file dengan ekstensi “.hex” digunakan perangkat bantu STLink Dongle yang akan dihubungkan dengan port USB (Universal Serial Bus) pada komputer. Sebelum downloader dapat digunakan perlu dilakukan instalasi driver terlebih dahulu, dapat dilihat pada gambar 3.6.



Gambar 3.6 ST-link Utility

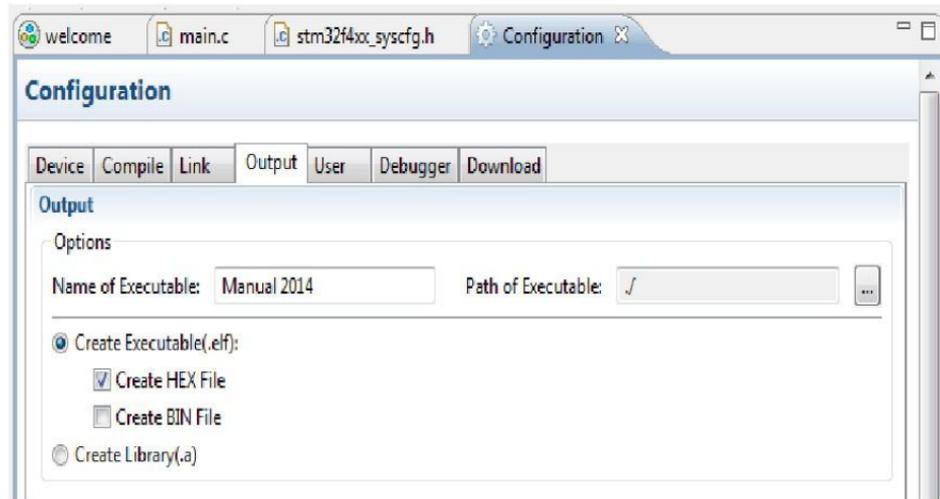
Program editor dan compiler yang digunakan untuk pembuatan program adalah CooCox IDE seri 1.7.3. Proses download file “.hex” dapat dilakukan melalui

program ini. Pengaturan penggunaan downloader pada CooCox Ide dilakukan dengan memilih menu view, kemudian pilihan Configuration seperti yang ditunjukkan pada Gambar 3.7.



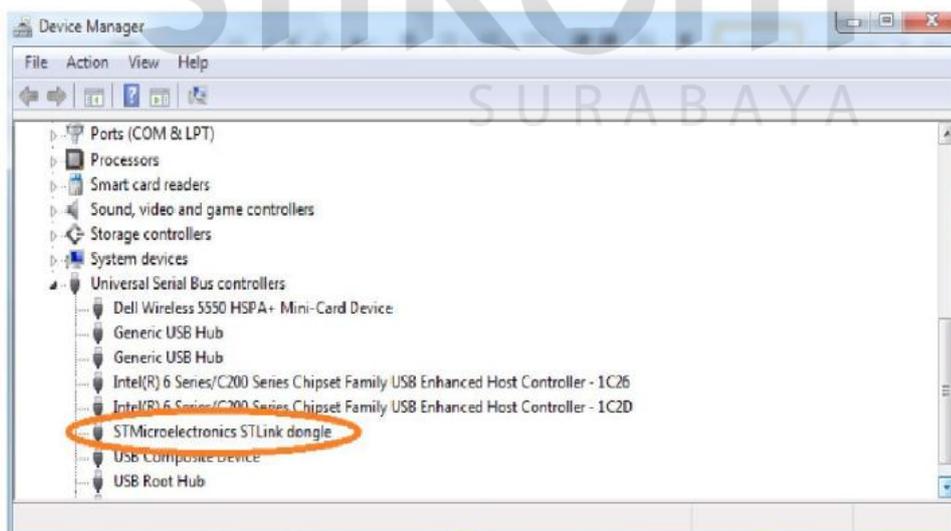
Gambar 3.7 Setting ST-link Utility

Setelah memilih *view* pada menu *configuration*, akan muncul *window configuration* lalu pilih *output*, yang dilanjutkan dengan memilih tipe *Hex File* dan karena *ARM Cortex M-4 (STM32F4 DISCOVERY)* harus menggunakan file extensi “.hex”, Setelah melakukan instalasi *driver STM32 ST-LINK Utility* hubungkan *ARM Cortex M-4 (STM32F4 DISCOVERY)* dengan menggunakan kabel *USB type A* ke *USB type mini-B* dapat di perhatikan pada Gambar 3.8.

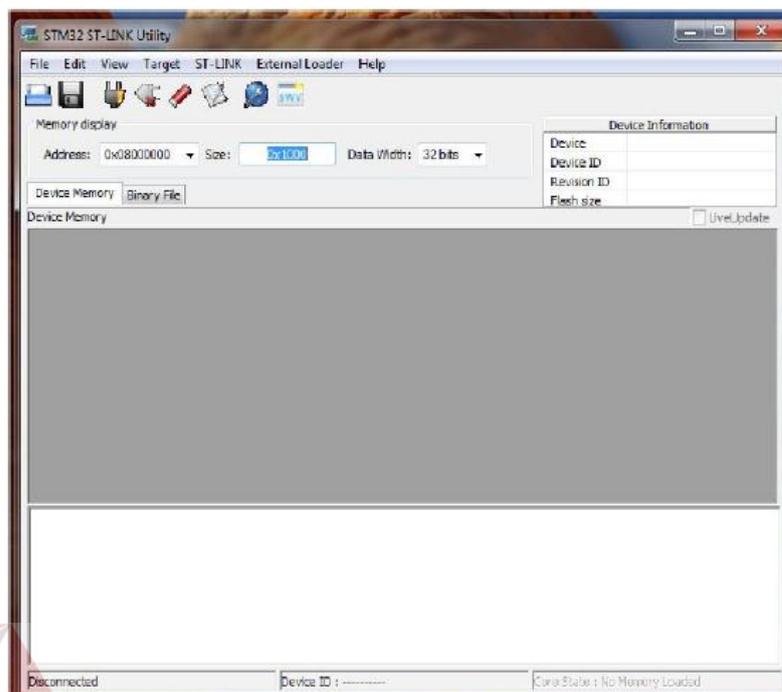


Gambar 3.8. Setting Configuration

Untuk mengetahui proses instalasi perangkat bantu *ST-Link dongle* dengan benar maka buka bagian *device manager* pada PC/Laptop dapat diperlihatkan pada Gambar 3.9, Jika proses tersebut sudah terinstalasi dengan benar maka *ARM Cortex M-4 (STM32F4 DISCOVERY)* sudah dapat digunakan menggunakan *software STM32 ST-LINK Utility* untuk mendownload file ekstensi “.hex” dapat diperlihatkan pada Gambar 3.10.



Gambar 3.9. Setting ComPort



Gambar 3.10. Running Aplikasi ST link Utility

3.4.5 Driver Motor DC 24V (EMS 30A H-Bridge)

Embedded Module Series (EMS) 30 A H-Bridge merupakan *driver* H-Bridge berbasis VNH3SP30 yang didesain untuk menghasilkan *drive* 2 arah dengan arus kontinyu sampai dengan 30 A pada tegangan 5,5 Volt sampai 36 Volt Modul ini dilengkapi dengan rangkaian sensor arus beban yang dapat digunakan sebagai umpan balik ke pengendali. Secara garis besar, fungsi modul pengendali motor ini adalah untuk mengendalikan arah dan kecepatan putaran motor DC sesuai instruksi kendali dari ARM Cortex M-4 (*STM32F4 DISCOVERY*).

3.4.5.1. Konfigurasi Pin Modul EMS 30A H-Bridge

Modul H-Bridge memiliki 1 set *header* (J1) dan 1 set terminal konektor (J2). Pada bagian ini akan dijelaskan deskripsi dan fungsi dari masing-masing *header* dan konektor tersebut. *Interface Header* (J1) berfungsi sebagai input untuk antarmuka dengan input-output digital serta output analog dari modul H-Bridge.

Berikut deskripsi dari masing-masing pin pada *Interface Header* dapat di perlihatkan pada table 3.1 untuk mengetahui layout IC VNH3SP30, konektor J1 dan J2 .

Tabel 3.1. Datasheet port input EMS H-Bridge

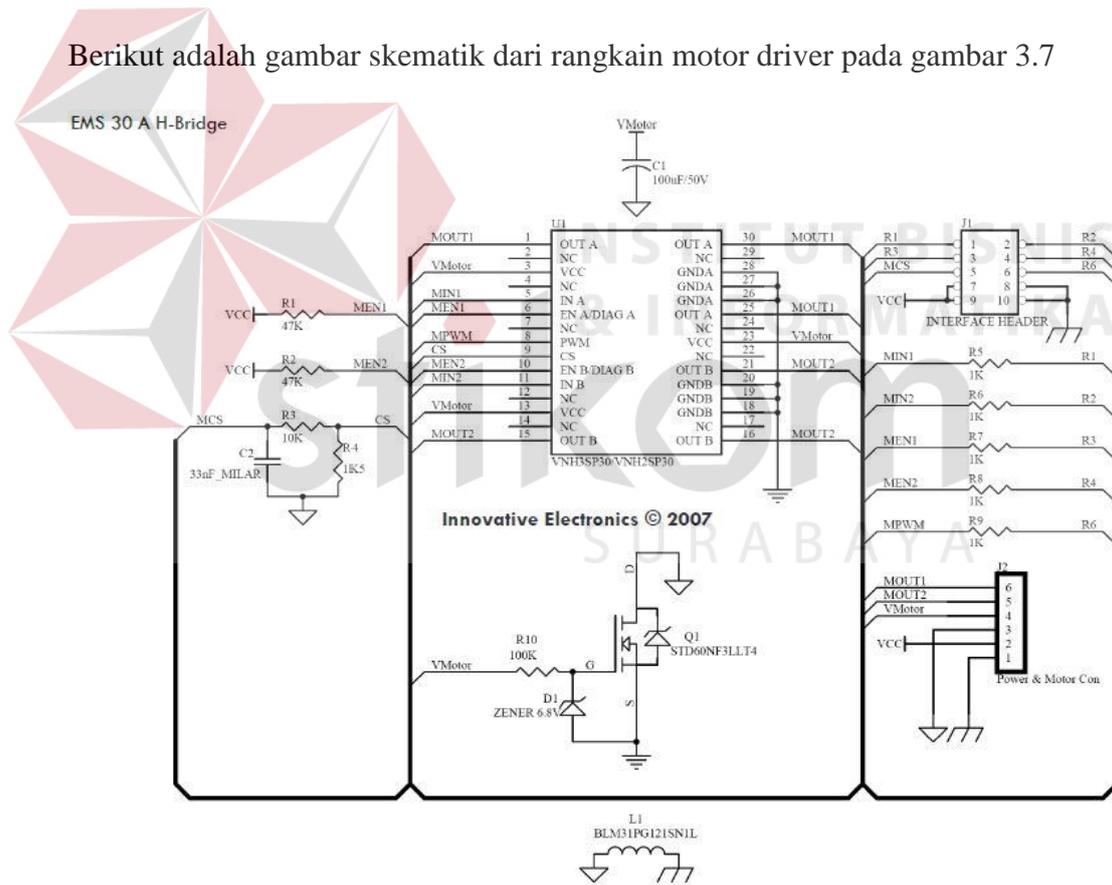
No. Pin	Nama	I/O	Fungsi
1	MIN1	I	Pin input untuk menentukan output MOUT1
2	MIN2	I	Pin input untuk menentukan output MOUT2
3	MEN1	I/O	Pin <i>enable</i> untuk output MOUT1 Diberi logika High untuk mengaktifkan <i>half</i> H-Bridge 1, diberi logika Low secara eksternal untuk menonaktifkan <i>half</i> H-Bridge 1 Jika terjadi kondisi <i>Fault</i> (<i>thermal shutdown, undervoltage, overvoltage, dsb.</i>), maka pin ini akan ditarik Low secara internal oleh modul HBridge untuk melaporkan adanya kondisi <i>Fault</i>
4	MEN2	I/O	Pin <i>enable</i> untuk output MOUT2 Diberi logika High untuk mengaktifkan <i>half</i> H-Bridge 2, diberi logika Low secara eksternal untuk menonaktifkan <i>half</i> H-Bridge 2 Jika terjadi kondisi <i>Fault</i> (<i>thermal shutdown, undervoltage, overvoltage, dsb.</i>), maka pin ini akan ditarik Low secara internal oleh modul HBridge untuk melaporkan adanya kondisi <i>Fault</i>
5	MCS	O	Output tegangan analog yang berbanding lurus dengan arus beban (<i>Range</i> output 0 – 5 Volt) Tersedia untuk IC VNH2SP30
6	MPWM	I	Pin input untuk mengatur kerja modul H-Bridge secara PWM
7,9	VCC	-	Terhubung ke catu daya untuk input (5 Volt)
8,10	Gnd	-	titik referensi untuk catu daya input

Power & Motor Con (J2) berfungsi sebagai konektor untuk catu daya dan beban. Berikut deskripsi dari masing-masing terminal pada Power & Motor Con dapat di lihat pada table 3.2

Tabel 3.2 Datasheet port output EMS H-Bridge.

Nama	Fungsi
PGND	Titik referensi untuk catu daya input
VCC	Terhubung ke catu daya untuk input (5 Volt)
MGND	Titik referensi untuk catu daya output ke beban
VMOTOR(V MOT)	Terhubung ke catu daya untuk output ke beba
MOUT2	Output ke beban dari half H-Bridge kedua
MOUT1	Output ke beban dari half H-Bridge pertama

Berikut adalah gambar skematik dari rangkain motor driver pada gambar 3.7



Gambar 3.11 Skematik Motor Driver EMS- 30A

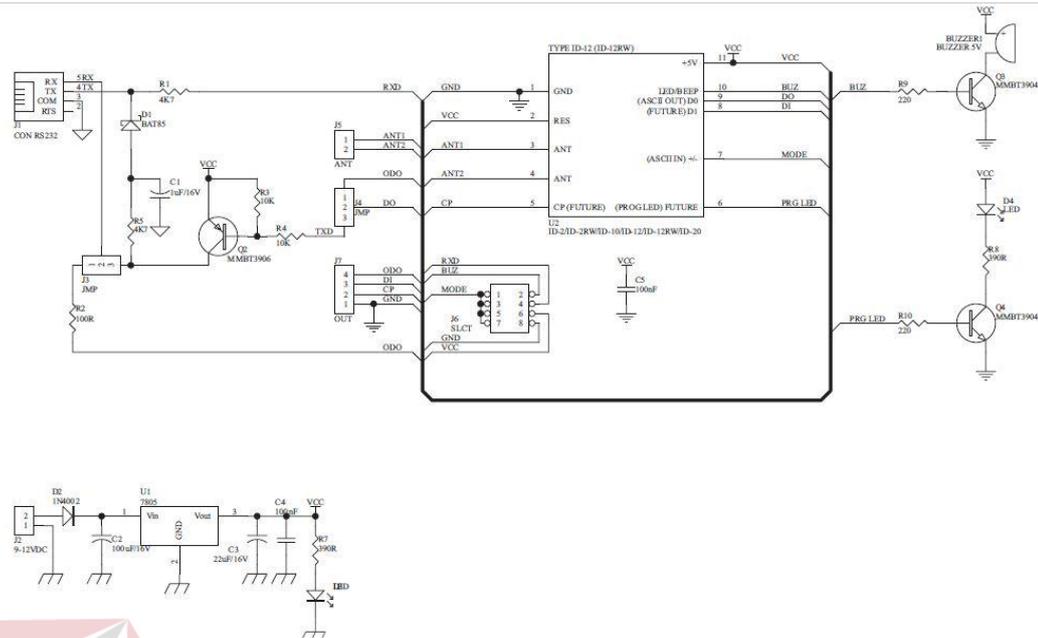
3.4.6 RFID Starter Kit ID-12

RFID Starter Kit merupakan suatu sarana pengembangan RFID berbasis reader tipe ID-12 yang telah dilengkapi dengan jalur komunikasi RS-232 serta indikator buzzer dan LED. Modul ini dapat digunakan dalam aplikasi mesin absensi RFID, RFID access controller, dsb.

Spesifikasi

1. Berbasis RFID reader ID-12 dengan frekuensi kerja 125 kHz untuk kartu berformat EM4001/sejenis dan memiliki jarak baca maksimal 12 cm.
2. Kompatibel dengan varian RFID reader lainnya, antara lain: ID-2, ID-10, dan ID-20.
3. Mendukung varian RFID reader/writer, antara lain: ID-2RW, ID-12RW, dan ID-20RW.
4. Mendukung format data ASCII (UART TTL/RS-232), Wiegand26, maupun Magnetic ABA Track2 (Magnet Emulation).
5. Dilengkapi dengan buzzer sebagai indikator baca, serta LED sebagai indikator tulis.
6. Tersedia jalur komunikasi serial UART RS-232 dengan konektor RJ11.
7. Tegangan input catu daya 9 - 12 VDC (J2).

Berikut adalah gambar skematik dari rangkaian RFID Reader Starter Kit pada gambar 3.12



Gambar 3.12 Skematik RFID Reader

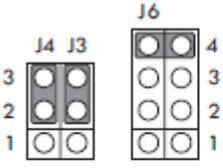
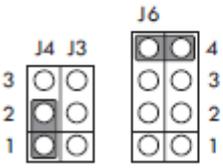
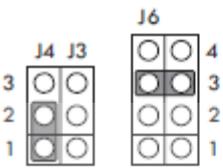
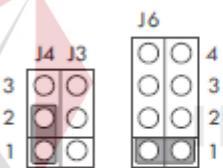
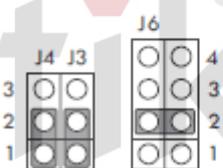
3.4.7. Prosedur Testing

Prosedur testing berikut akan menguji jalur komunikasi RS-232 dan RFID reader ID-12 dalam mode ASCII.

Langkah-langkah testing:

1. Atur RFID Starter Kit agar RFID reader ID-12 bekerja pada mode UART RS-232 (ASCII), yaitu jumper J3 & J4 pada posisi 2-3 dan jumper J6 pada posisi 4.
2. Hubungkan RJ11 (J1) RFID Starter Kit ke COM port computer menggunakan kabel serial.
3. Hubungkan catu daya 9 VDC ke terminal J2 RFID Starter Kit.
4. Jalankan program RFID1.exe, lalu pilih COMport yang sesuai.
5. Nyalakan catu daya, lalu dekatkan RFID transponder ke RFID reader.

Pada program RFID1.exe akan muncul nomor ID dari RFID transponder tersebut.

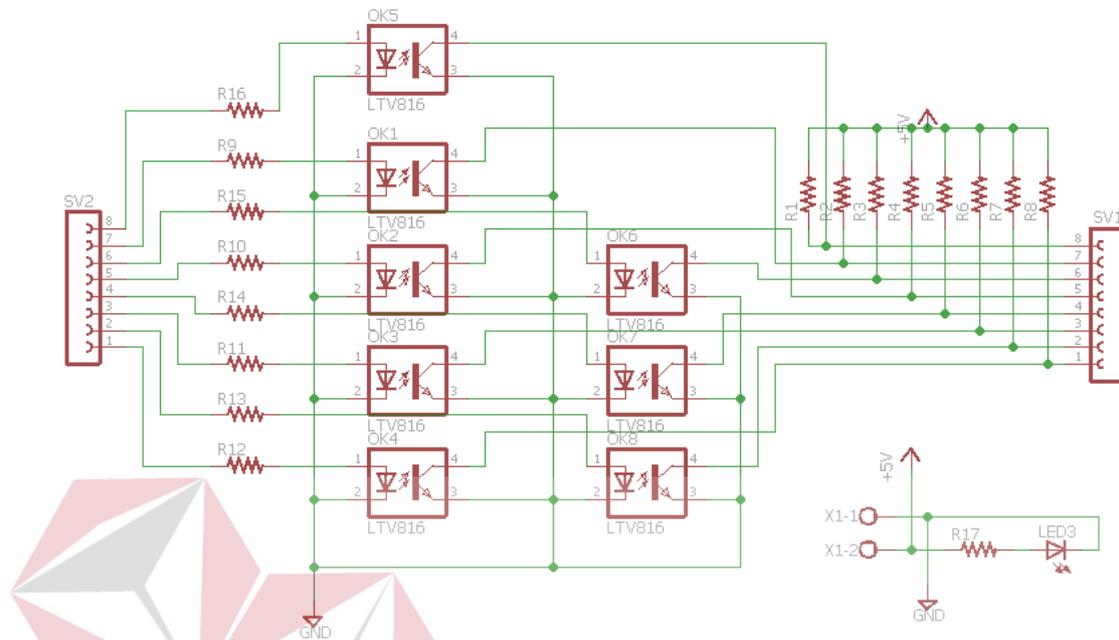
Setting Jumper J3, J4, & J6		
	<p>RFID reader only dengan format data UART RS-232 (ASCII).</p>	
	<p>RFID reader only dengan format data UART TTL (ASCII).</p>	
	<p>RFID reader only dengan format data Wiegand26</p>	
	<p>RFID reader only dengan format data Magnet Emulation</p>	
	<p>RFID reader/writer dengan antarmuka UART RS-232.</p>	

Gamabar 3.13. Datasheet mode jamper modul RFID

3.4.8. Optocoupler

Optocoupler adalah suatu komponen penghubung (coupling) yang bekerja berdasarkan picu cahaya optik, optocoupler adalah sejenis transistor namun dia memiliki kecepatan cahaya jadi sangat cocok digunakan untuk rangkaian yang membutuhkan kecepatan tinggi, dalam tugas akhir ini rangkaian optocoupler digunakan sebagai penguat daya outputan dari Arm Cortex karena daya output

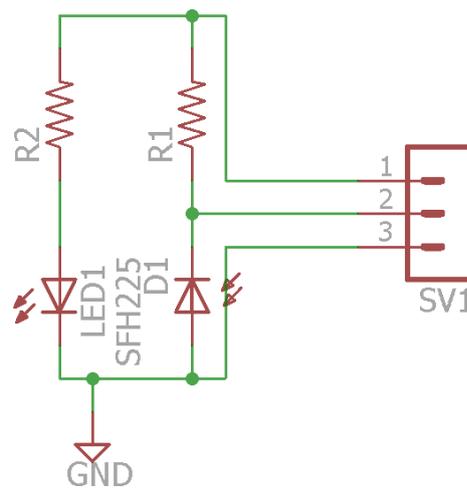
sebenarnya hanyalah 3V. Berikut gambar rangkaian skematik optocoupler pada gambar 3.11.



Gambar 3.14 Skematik Optocoupler

3.4.9 Sensor Photodioda

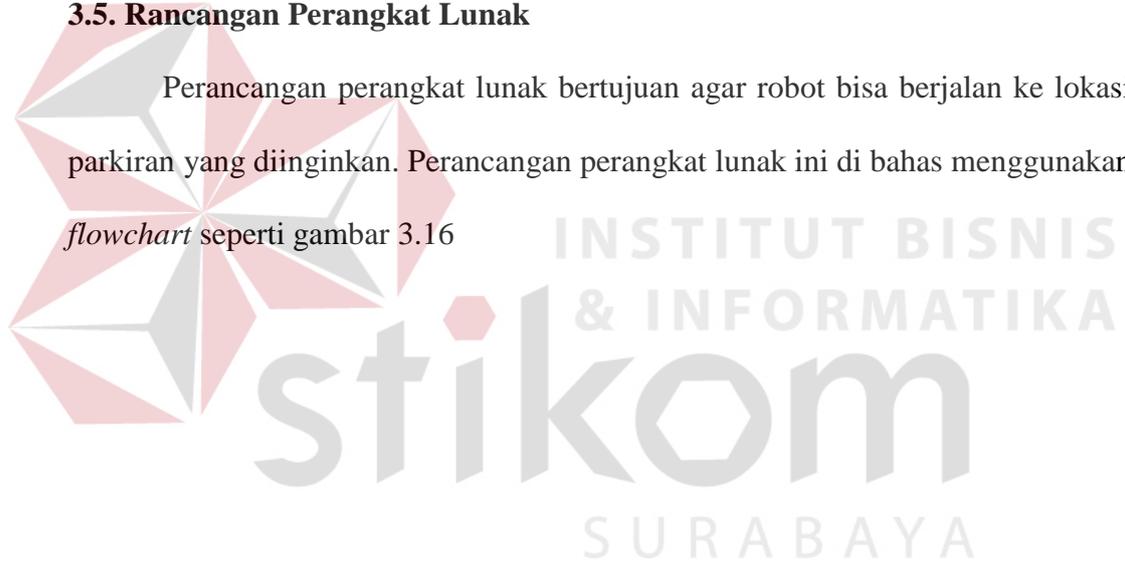
Photodioda adalah sebuah dioda semikonduktor yang berfungsi sebagai sensor cahaya. Photodioda memiliki hambatan yang sangat tinggi pada saat dibias mundur. Hambatan ini akan berkurang ketika photodioda disinari cahaya dengan panjang gelombang yang tepat. Sehingga photodioda dapat digunakan sebagai detektor cahaya dengan memonitoring arus yang mengalir melaluinya. Sensor ini digunakan sebagai receiver dari pantulan sinar dari sebuah led. Dalam tugas akhir ini sensor ini digunakan sebagai pembaca garis atau alur yang akan dilewati oleh robot. Berikut adalah gambar skematik rangkaian photodioda pada gambar 3.12.

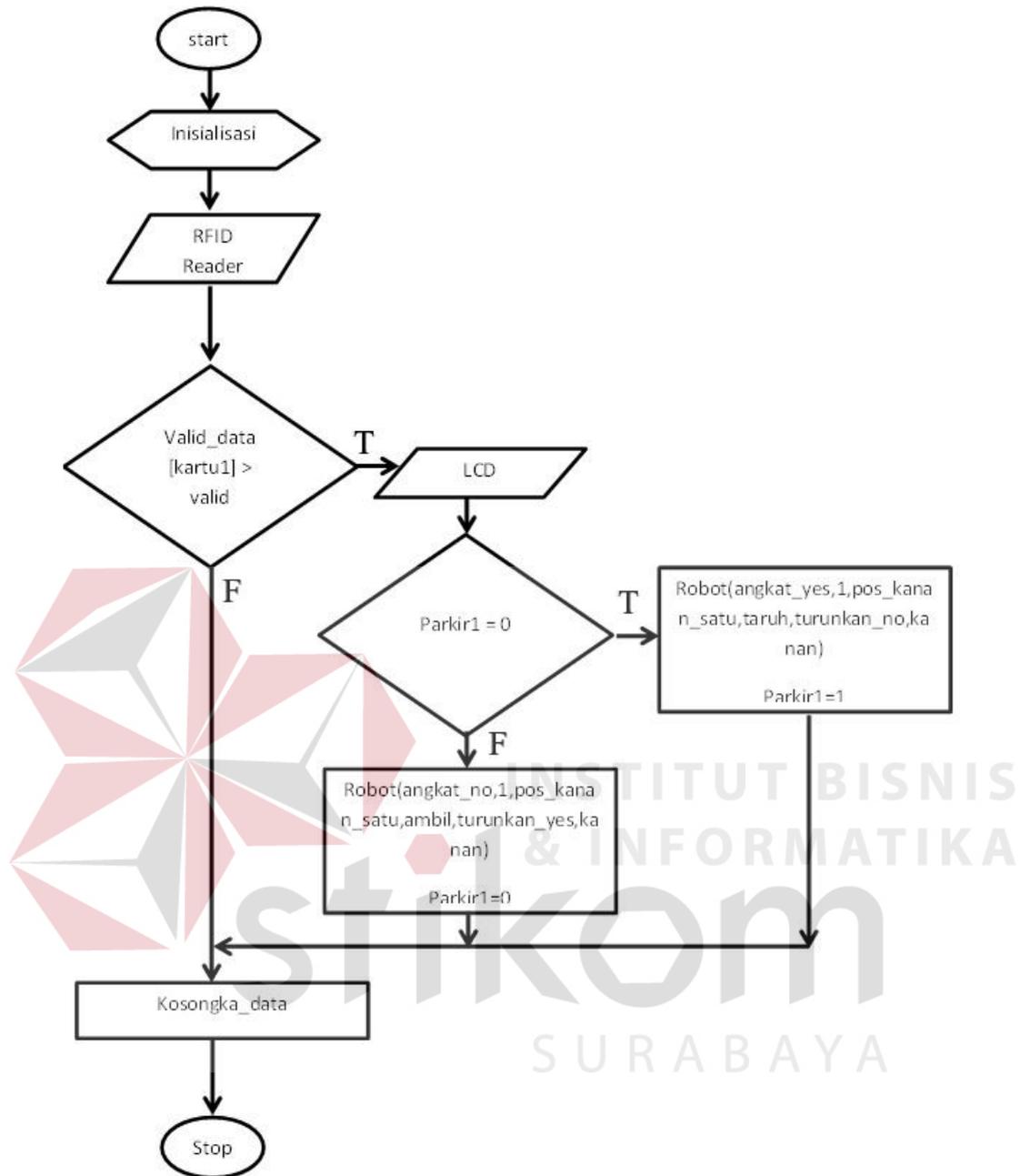


Gambar 3.15 Skematik Sensor Photodiode

3.5. Rancangan Perangkat Lunak

Perancangan perangkat lunak bertujuan agar robot bisa berjalan ke lokasi parkir yang diinginkan. Perancangan perangkat lunak ini di bahas menggunakan *flowchart* seperti gambar 3.16





Gambar 3.16 Flowchart keseluruhan sistem

Pada gambar diatas merupakan alur pergerakan *mobile robot* dari posisi awal hingga akhir robot sampai kembali diposisi awal lagi, disini sensor digunakan untuk membantu robot bisa berjalan dengan baik dijalur nya dan bisa

mengantarkan mobil sampai ditujuan, dan sensor rotary encoder digunakan sebagai membantu penanda pergerakan robot dan sensor photodiode sebagai penanda titik tujuan dari robot.

Disini robot akan mulai berjalan apabila RFID reader sudah membaca serial number dari RFID card, setelah sudah terbaca maka robot akan berjalan menuju lokasi yang akan dituju dan akan kembali lagi keposisi awal ketika tugasnya sudah terpenuhi.

Berikut potongan program dari sistem untuk satu tujuan dan untuk tujuan lainnya bisa mengikuti hanya tinggal menambahkan beberapa coding :

```
int main(void)
{
    init_USART1(9600);
    DDRD();
    PWM();
    INTO();
    INT1();
    INT2();
    INT3();
    adc_configure();

    lcd_init(16);
    lcd_gotoxy(0,0);
    lcd_puts("Bismillah");
    delay_ms(1000);
    lcd_clear();

    while(1)
    {
        sprintf(buff,"%s ",received_string);
        lcd_gotoxy(0,0);
        lcd_puts(buff);

        if(Valid_Data(kartu1) > valid)
        {
            lcd_gotoxy(0,1);
            lcd_puts("Kartu 1");
            if(parkir1 == 0){

robot(angkat_yes,1,Pos_Kanan_Satu,taruh,turunkan_no, kanan);
                parkir1 = 1;
            }
        }
    }
}
```

```

    }
    else {

robot(angkat_no,1,Pos_Kanan_Satu,ambil,turunkan_yes, kanan);
        parkir1 = 0;
    }
    Kosongkan_Data();
}
else if(Valid_Data(kartu2) > valid)
{
    lcd_gotoxy(0,1);
    lcd_puts("Kartu 2");
    if(parkir2 == 0){
        robot(angkat_yes,1,Pos_Kiri_Satu,taruh,turunkan_no,
kiri);

        parkir2 = 1;
    }
    else {
        robot(angkat_no,1,Pos_Kiri_Satu,ambil,turunkan_yes,

        parkir2 = 0;
    }
    Kosongkan_Data();
}
else if(Valid_Data(kartu3) > valid)
{
    lcd_gotoxy(0,1);
    lcd_puts("Kartu 3");
    if(parkir3 == 0){
        robot(angkat_yes,1,Pos_Kanan_Dua,taruh,turunkan_no,
kanan);

        parkir3 = 1;
    }
    else {

robot(angkat_no,1,Pos_Kanan_Dua,ambil,turunkan_yes, kanan);
        parkir3 = 0;
    }
    Kosongkan_Data();
}
else if(Valid_Data(kartu4) > valid)
{
    lcd_gotoxy(0,1);
    lcd_puts("Kartu 4");
    if(parkir4 == 0){
        robot(angkat_yes,1,Pos_Kiri_Dua,taruh,turunkan_no,
kiri);

        parkir4 = 1;
    }
    else {

```

```

        robot(angkat_no,1,Pos_Kiri_Dua,ambil,turunkan_yes,
kiri);
        parkir4 = 0;
    }
    Kosongkan_Data();
}
else if(Valid_Data(kartu5) > valid)
{
    lcd_gotoxy(0,1);
    lcd_puts("Kartu 5");
    if(parkir5 == 0){
        robot(angkat_yes,2,Pos_Kanan_Satu,taruh,turunkan_no, kanan);
        parkir5 = 1;
    }
    else {
        robot(angkat_no,2,Pos_Kanan_Satu,ambil,turunkan_yes, kanan);
        parkir5 = 0;
    }
    Kosongkan_Data();
}
else if(Valid_Data(kartu6) > valid)
{
    lcd_gotoxy(0,1);
    lcd_puts("Kartu 6");
    if(parkir6 == 0){
        robot(angkat_yes,2,Pos_Kiri_Satu,taruh,turunkan_no,
kiri);
        parkir6 = 1;
    }
    else {
        robot(angkat_no,2,Pos_Kiri_Satu,ambil,turunkan_yes,
kiri);
        parkir6 = 0;
    }
    Kosongkan_Data();
}
else if(Valid_Data(kartu7) > valid)
{
    lcd_gotoxy(0,1);
    lcd_puts("Kartu 7");
    if(parkir7 == 0){
        robot(angkat_yes,2,Pos_Kanan_Dua,taruh,turunkan_no,
kanan);
        parkir7 = 1;
    }
    else {
        robot(angkat_no,2,Pos_Kanan_Dua,ambil,turunkan_yes, kanan);

```

```

        parkir7 = 0;
    }
    Kosongkan_Data();
}
else if(Valid_Data(kartu8) > valid)
{
    lcd_gotoxy(0,1);
    lcd_puts("Kartu 8");
    if(parkir8 == 0){
        robot(angkat_yes,2,Pos_Kiri_Dua,taruh,turunkan_no,
kiri);

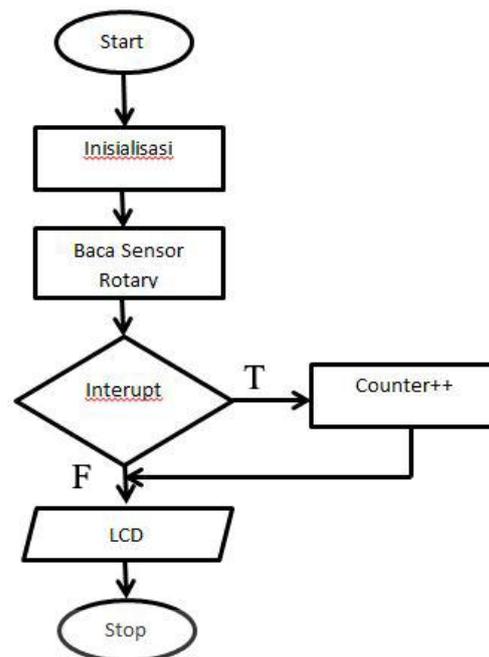
        parkir8 = 1;
    }
    else {
        robot(angkat_no,2,Pos_Kiri_Dua,ambil,turunkan_yes,
kiri);

        parkir8 = 0;
    }
    Kosongkan_Data();
}
else
{
    lcd_gotoxy(0,1);
    lcd_puts("waiting");
}
}
}

```

3.5.1 Program Membaca Sensor *Rotary Encoder*

Program pembacaan *rotary encoder* menggunakan mode interrupt, kenapa menggunakan interrupt karena pembacaan sensor terus-menerus jadi tidak bisa kalau harus menunggu siklus suatu program selesai berjalan.



Gambar 3.17. Flowchat Program Sensor Rotary

Pada flowchart di atas terlihat bagaimana cara untuk pembacaan sensor rotary encoder, dalam flowchart untuk pembacaan rotary encoder menggunakan mode interupt dan jika interrupt terbaca maka counter akan menambah satu per satu dan begitu seterusnya.

Berikut potongan program pembacaan Rotary Encoder:

```

void EXTI0_IRQHandler(void)
{
    a++;
    EXTI_ClearITPendingBit(EXTI_Line0);
    EXTI_ClearFlag (EXTI_Line0);
}
void EXTI1_IRQHandler(void)
{
    x++;
    b++;
    EXTI_ClearITPendingBit(EXTI_Line1);
    EXTI_ClearFlag (EXTI_Line1);
}
void EXTI2_IRQHandler(void)
{
    c++;
  
```

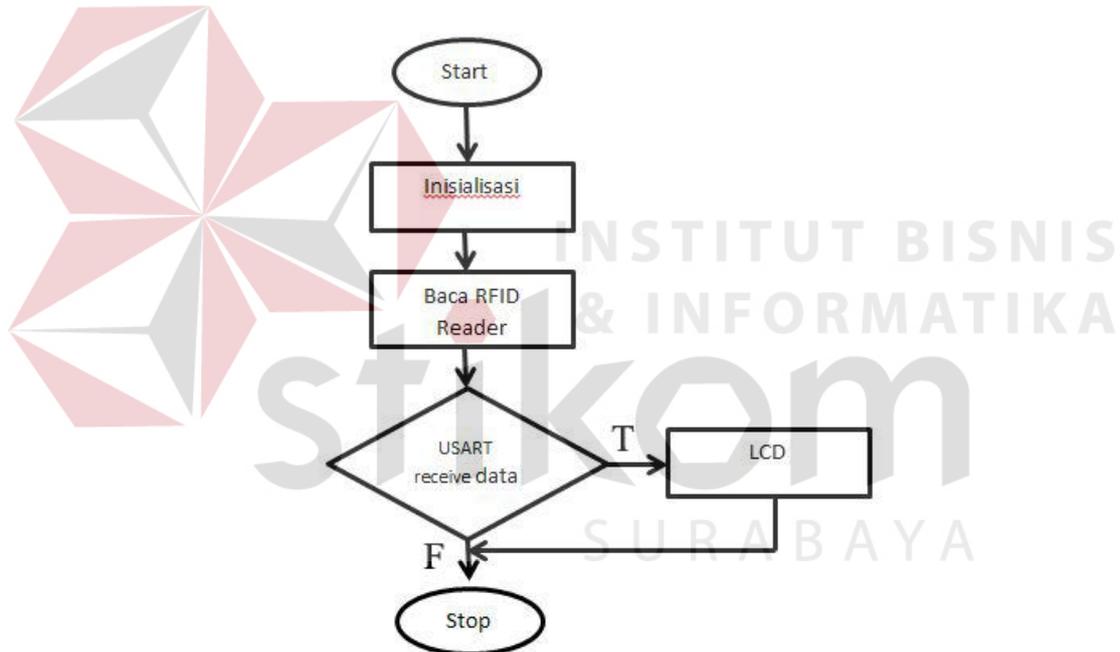
```

EXTI_ClearITPendingBit(EXTI_Line2);
EXTI_ClearFlag (EXTI_Line2);
}
void EXTI3_IRQHandler(void)
{
    d++;
    EXTI_ClearITPendingBit(EXTI_Line3);
    EXTI_ClearFlag (EXTI_Line3);
}

```

3.5.2 Program Pembacaan RFID

Pada Pengujian pembacaan RFID dapat dilihat dalam flowchart berikut.



Gambar 3.18 Flowchart Pembacaan kartu RFID

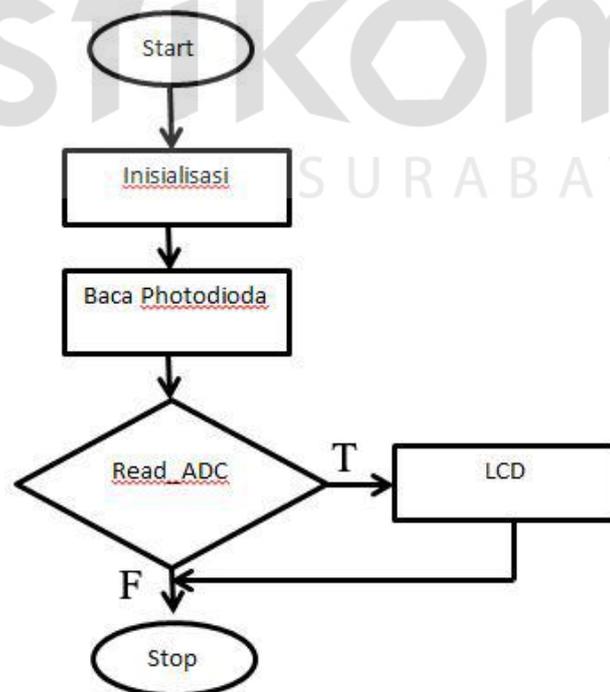
Dalam flowchart diperlihatkan bagaimana cara untuk membaca RFID, disini saya menggunakan mode USART untuk membaca RFID karena tipikal data output an dari modul RFID adalah data serial, Didalam flowchart data yang

diterima langsung di outputkan ke LCD agar dapat kita lihat data yang dikirimkan oleh modul RFID ke STM32F4 Berikut potongan program pembacaan RFID :

```
main()
{
    data_terima = USART_ReceiveData(USART3);
    lcd_gotoxy(9,1);
    sprintf(b, "%c", data_terima);
    lcd_puts (b);
}
```

3.5.3 Program Pembacaan Sensor Photodiode

Program pembacaan sensor photodiode menggunakan mode ADC (Analog Digital Converter), Kenapa menggunakan mode ADC karena data output dari sensor adalah data analog sedangkan minimum system STM32F4 hanya bisa menerima data digital, maka dari itu data perlu di convert menggunakan mode ADC.



Gambar 3.19 Flowchart program sensor photodiode

Dari flowchart di atas bisa dilihat bagaimana cara pembacaan sensor photodiode menggunakan mode ADC agar minimum system bisa membaca data dari sensor photodiode karena data analog nya dirubah ke data digital, dalam flowchart tersebut ketika data inputan dari sensor data tersebut akan diolah oleh mode ADC terlebih dahulu dan lalu akan di inputkan ke lcd sebagai display. Berikut adalah potongan program dari pembacaan sensor.

```
int main()
{
    while(1)
    {
        depan_kanan = read_adc(0);
        depan_kiri = read_adc(1);
        belakang_kanan = read_adc(2);
        belakang_kiri = read_adc(3);
        tengah_kanan = read_adc(4);
        tengah_kiri = read_adc(5);
        counter_kiri = read_adc(6);
        counter_kanan = read_adc(7);

        delay_ms(100);
        lcd_clear();

        sprintf(buff,"%d %d %d
        %d",depan_kanan,depan_kiri,belakang_kanan,belakang_kiri);

        lcd_gotoxy(0,0);
        lcd_puts(buff);

        sprintf(buff,"%d %d %d
        %d",tengah_kanan,tengah_kiri,counter_kiri,counter_kanan);

        lcd_gotoxy(0,1);
        lcd_puts(buff);
    }
}
```

3.6 Metode Pengujian dan Evaluasi Sistem

Dalam pengujian sistem ini pengujian akan dilakukan pada perangkat keras serta perangkat lunak yang telah dibuat. Pengujian yang telah dilakukan dimulai dari pengujian minimum sistem, pengujian motor DC, pengujian sensor rotary encoder, pengujian sensor photodiode, serta pengujian pembacaan RFID reader.

3.6.1 Pengujian dan Evaluasi Minimum Sistem (STM32F4 Arm Cortex M4)

Pengujian minimum sistem ini bertujuan untuk mengetahui apakah minimum sistem dalam *robot* dapat melakukan proses *signature* dan *download* program ke mikrokontroler dengan baik. Pengujian ini dilakukan dengan cara mengaktifkan *power supply* dan hubungannya dengan minimum sistem. Sambungkan minimum sistem dengan komputer menggunakan kabel *downloader* lalu jalankan *compiler ST-Link Utility* pada komputer tersebut. Setelah itu lakukan proses *download* pada mikrokontroler, jika proses *download* berhasil maka akan *ST-Link Utility* akan mengeluarkan tampilan proses *download*.

3.6.2 Pengujian dan Evaluasi Sensor Rotary Encoder

Pengujian sensor rotary encoder ini bertujuan untuk mengetahui informasi apakah sensor mampu mengeluarkan pulse nya sesuai dengan ppr nya. Dalam hal ini pengujian dilakukan dengan memberikan tegangan pada sensor rotary encoder dan output data sensor dimasukkan ke osiloskop untuk mengetahui pulse yang keluar.

3.6.3. Pengujian dan Evaluasi Motor DC

Pengujian motor DC ini bertujuan untuk mengetahui apakah motor DC bergerak atau berputar sesuai dengan yang diharapkan dan berputar dengan *pwm* tertentu sesuai dengan program yang telah di tentukan. Dalam hal ini pengujian dilakukan dengan memberikan tegangan pada driver motor DC dan melakukan koneksi antara driver motor DC dengan minimum sistem. Setelah itu proses dilanjutkan dengan menggunakan program untuk menggerakkan motor DC pada *CoIDE* yang di *download* pada STM32F4. Jika proses pengujian tersebut berhasil maka motor DC akan bergerak sesuai dengan yang diperintahkan dalam program yang terdapat pada mikrokontroler.

3.6.4 Pengujian dan Evaluasi Program Sensor Photodiode

Pengujian ini dilakukan untuk menguji apakah sensor dapat membedakan warna hitam dan putih, cara pengujian adalah output sensor akan dimasukan pada comparator dan kita tentukan threshold nya mengguna variable resistor setelah itu output dari komparator kita cek menggunakan avometer, dari situ kita bisa melihat data yang keluar 0V atau 5V ketika mengenai garis.

3.6. Pengujian RFID Reader

Pengujian ini dilakukan untuk menguji apakah rfid reader bisa membaca data yang ada dalam chip rfid card, langkah-langkah pengujian nya adalah kita beri rfid reader catu daya 12V sesuai datasheet nya, setelah itu kita hubungkan port rs232 yang ada pada rfid reader ke kabel usb to serial agar rfid bisa terkoneksi dengan PC, setelah itu kita nyalakan aplikasi rfid.exe setelah itu kita mulai tapping kan kartu ke rfid reader, ketika buzzer berbunyi dan data muncul pada aplikasi berarti rfid reader sudah bisa dipastikan berfungsi dengan baik.