

**RANCANG BANGUN PROTOTYPE PALANG PARKIR
MENGUNAKAN MIKROKONTROLER ATMEGA 8535**

Sakti Raharja,
Program Studi Teknik Mesin
Fakultas Teknik Universitas Pasir Pengaraian
E-mail : Setiawansakti0@gmail.com

ABSTRAK

Penelitian ini bertujuan untuk merancang bangun *prototype* palang parkir menggunakan mikrokontroler ATmega 8535 serta membuat program untuk menjalankan sistem palang parkir secara otomatis. Metodologi yang digunakan dalam penelitian ini diantaranya (1) Merancang program menggunakan aplikasi *CodeVision AVR*, (2) Mensimulasikan program menggunakan aplikasi *Proteus 8 Personal* dan (3) Pembuatan *prototype* palang parkir otomatis. Hasil yang dicapai dari penelitian yang dilakukan yakni berupa *prototype* palang parkir otomatis menggunakan mikrokontroler ATmega 8535 dengan kapasitas 4 unit. Kesimpulan dari penelitian yang dilakukan yaitu sistem palang parkir otomatis menggunakan ATmega 8535 dapat dioperasikan sesuai dengan program yang telah dibuat.

Kata Kunci : *Prototype, Palang Parkir, ATmega 8535*

**BAB I
PENDAHULUAN**

1.1. Latar Belakang

Kebutuhan perparkiran menjadi kebutuhan yang mendesak dewasa ini, terutama pada daerah yang berkembang pesat. Perparkiran merupakan bidang jasa yang harus dikelola dengan baik seperti di rumah sakit, perkantoran, perbelanjaan, dan tempat lainnya. Jika tidak dikelola penataan area parkir dengan benar maka dikhawatirkan kemacetan dan kedisiplinan kendaraan yang ada di area parkir tidak dapat dihindari.

Secara umum sistem parkir masih dikendalikan secara manual dimana sistem buka tutup palang dikerjakan oleh petugas parkir. Selain itu, masih ada juga sistem parkir yang hanya dijaga oleh manusia, ketika kendaraan masuk petugas parkir memberi karcis kepada pemilik kendaraan. Sistem pengelolaan area parkir tersebut memiliki kelemahan seperti tidak adanya informasi yang akurat mengenai jumlah data parkir yang kosong didalamnya. sehingga ketika kapasitas area parkir ternyata sudah penuh bisa berakibat ketidakteraturan area parkir yang akan menyulitkan kendaraan yang hendak keluar dari area parkir.

Bertitik tolak dari uraian diatas timbul suatu gagasan penulis untuk merancang bangun

prototype suatu alat dalam bentuk SKRIPSI dengan judul Rancang Bangun *Prototype* Palang Parkir Menggunakan Mikrokontroler ATmega 8535. Dengan alat ini diharapkan bisa memberikan data parkir yang akurat sehingga dapat meningkatkan ketertiban pada area parkir yang ada. Selain dari itu dengan mempelajari sistem kontrol menggunakan mikrokontroler dapat memacu minat serta bakat mahasiswa untuk lebih banyak menggali ilmu dan berkarya dengan menggunakan sistem mikrokontroler.

**BAB II
TINJAUAN PUSTAKA**

2.1. Mikrokontroler ATmega 8535

Mikrokontroler adalah sistem *mikroprosesor* yang terkandung di dalam sebuah *chip*. Mikrokontroler pada umumnya telah berisi komponen pendukung sistem minimal *mikroprosesor* yakni memori dan pemrograman *input – output*. Cara kerja mikrokontroler sebenarnya membaca dan menulis data. Mikrokontroler memiliki fungsi khusus untuk mengontrol suatu alat.



Gambar 2.1 Mikrokontroler ATmega 8535

Fitur - fitur yang dimiliki oleh mikrokontroler ATmega 8535 adalah sebagai berikut:

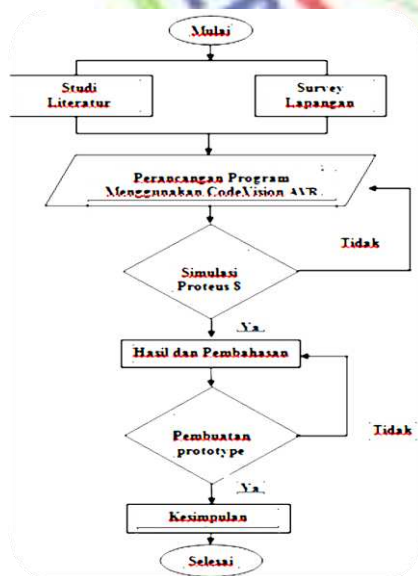
1. Saluran I/O sebanyak 32 buah, yaitu *port A, port B, port C, dan port D.*
2. *ADC* internal sebanyak 8 saluran.
3. Tiga buah *Timer/Counter* dengan kemampuan perbandingan.
4. *CPU (Central Prosesor Unit)* yang terdiri atas 32 buah *register.*
5. *SRAM* sebesar 512 *byte.*
6. Memori *Flash* sebesar 8 kb dengan kemampuan *Read While Write.*
7. *Port* antarmuka *SPI*
8. *EEPROM* sebesar 512 *byte* yang dapat diprogram saat operasi.
9. Antarmuka komparator analog.
10. *Port USART* untuk komunikasi serial fitur ATmega 8535
11. Sistem mikroprosesor 8 bit berbasis *RISC* dengan kecepatan maksimal 16 MHz. (<http://joaldera.com/2008/10/mikrokontroler-atmega-8535/>)

Keterangan :

Rancang bangun *prototype* palang parkir akan dilakukan sesuai dengan langkah – langkah sebagai berikut :

- a. Studi Literatur
Studi Literatur yang dilakukan meliputi pencarian bahan – bahan yang akan digunakan dengan cara mempelajari buku – buku yang berkaitan dengan pembuatan alat yang akan dibuat serta mengunjungi situs – situs yang berkaitan dengan pembuatan SKRIPSI ini.
- b. Survei Lapangan
Survei lapangan dilakukan untuk pengamatan dari sistem yang sudah ada dan kemudian dipelajari sehingga memperkaya referensi penelitian yang dilakukan.
- c. Perancangan Program *CodeVision AVR*
Perancangan berupa perancangan program yang akan digunakan pada sistem dan akan dimasukkan kedalam *chip* mikrokontroler .
- d. Simulasi *Proteus 8 Personal*
Simulasi dilakukan dengan *Proteus 8 Personal*, hal ini bertujuan untuk meminimalisir kesalahan – kesalahan yang terjadi pada tahap pembuatan *prototype.*
- e. Hasil dan Pembahasan
Berupa hasil program yang sudah diuji secara simulasi pada aplikasi *proteus 8 personal*
- f. Pembuatan *Prototype*
Melakukan pembuatan *prototype* palang parkir otomatis menggunakan mikrokontroler ATmega 8535
- g. Kesimpulan
Kesimpulan berupa hasil dari rancang bangun yang telah dibuat dan diuji secara keseluruhan

BAB III METODOLOGI PENELITIAN



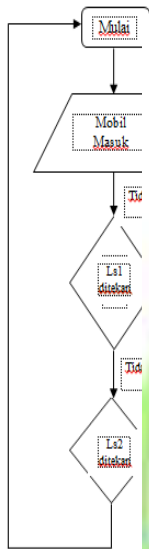
Gambar 3.1 Diagram Alir Penelitian

**BAB IV
HASIL DAN PEMBAHASAN**

4.1. Perancangan Sistem

4.1.1. Perancangan Program

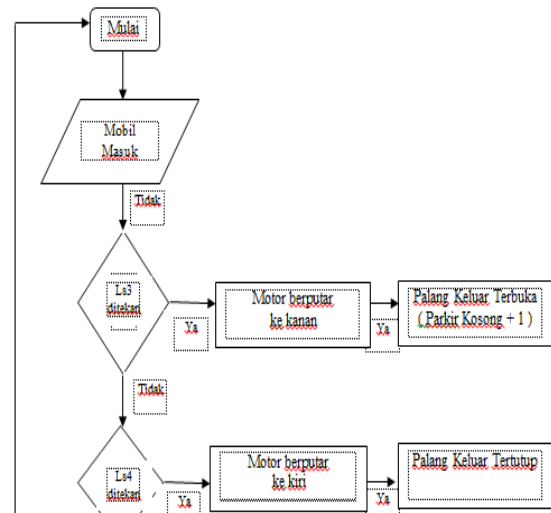
Sebelum membuat sebuah program diawali dengan pembuatan diagram alir program. Diagram alir program berupa penjelasan secara sederhana prinsip kerja daripada sistem yang akan dibuat. Berikut merupakan diagram alir rancang bangun *prototype* palang parkir otomatis :



Gambar 4.1 Flowchat Pintu Masuk

Dilihat dari gambar 4.1 dapat dijelaskan bahwa :

- Ketika mobil menuju pintu masuk kemudian menekan sensor Ls1.
- Pertama program akan langsung memberikan sinyal untuk mengaktifkan Motor DC 1 (pintu masuk) untuk berputar ke atas sehingga palang masuk terbuka
- Kedua, secara program juga memberikan sinyal untuk mengaktifkan LCD sehingga akan mengurangi data parkir yang ada (data -1).
- Ketika mobil menekan Ls2 program akan memberikan sinyal untuk mengaktifkan motor DC 1 (pintu masuk) untuk berputar ke bawah sehingga palang masuk tertutup.



Gambar 4.2 Flowchat Pintu Keluar

Dilihat dari gambar 4.2 dapat dijelaskan bahwa :

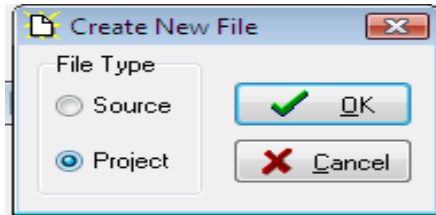
- Ketika mobil menuju pintu masuk kemudian menekan sensor Ls3.
- Pertama program akan langsung memberikan sinyal untuk mengaktifkan Motor DC 2 (pintu keluar) untuk berputar ke atas sehingga palang keluar terbuka
- Kedua, secara program juga memberikan sinyal untuk mengaktifkan LCD sehingga menambah data parkir yang ada (data +1).
- Ketika mobil menekan Ls4 ditekan program akan memberikan sinyal untuk mengaktifkan motor DC 2 (pintu keluar) untuk berputar ke bawah sehingga palang keluar tertutup.

4.1.2. Pembuatan Program

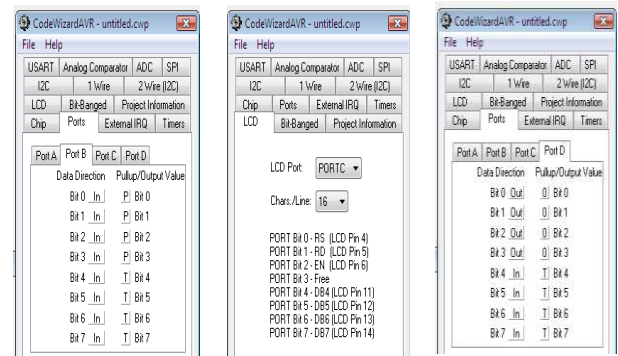
Pembuatan program penulis menggunakan CodeVision AVR kemudian dapat didownload ke dalam chip dengan downloader K-125 R Uno dan jika ada kesalahan dapat diatur kembali. Pada tool Codevision AVR bisa ditentukan port - port dari mikrokontroler AVR yang berfungsi sebagai input atau output serta bisa juga ditentukan tentang penggunaan fungsi – fungsi internal dari AVR. Dalam sebuah program terdapat sebuah intruksi – intruksi lalu kemudian diproses sehingga *prototype* palang parkir dapat melakukan gerakan – gerakan yang telah disusun dalam program.

Langkah – langkah dalam pembuatan program tersebut adalah sebagai berikut :

1. Jalankan *CodeVision AVR* , kemudian pilih **File** lalu **New**, pilih **Project**. Pastikan *software CodeVision AVR* telah terinstal, kemudian buka *software* tersebut.



Gambar 4.3 Membuat File baru



Gambar 4.6 Menentukan input, output dengan CodeWizard

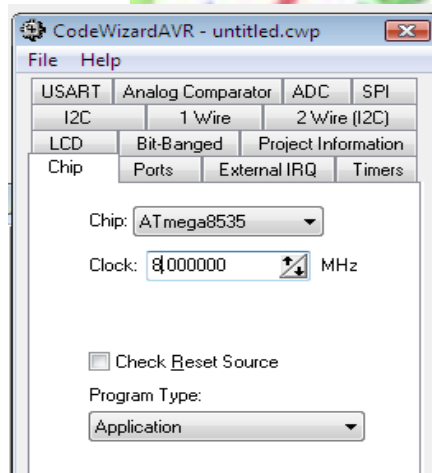
2. Pilih **Yes** ketika muncul tampilan “Do you want to use the CodeWizardAVR ?”



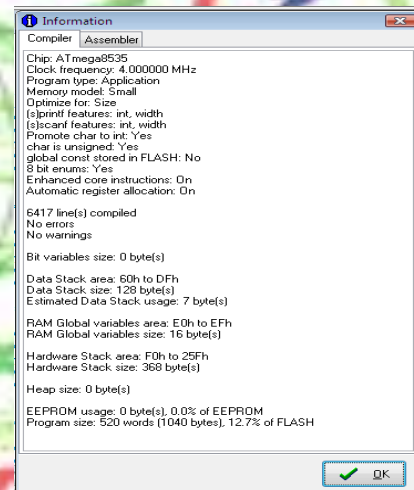
Gambar 4.4 Mengaktifkan Codewizard

5. Pilih **File** lalu pilih **Generate, Save and Exit**.
6. Buatlah *listing* program seperti pada lampiran.
7. Setelah selesai, maka program harus dikompilasi. Untuk melakukan perintah ini bisa dilakukan dengan cara tekan tombol shift + F9.

3. Pilih *chip* yang digunakan untuk program palang parkir otomatis.



Gambar 4.5 Membuka codewizard



Gambar 4.7 Tampilan Proses Kompilasi

4. Lakukan pengaturan sebagai berikut :

Port B : Sebagai *input limit switch*
Port C : *LCD*
Port D : Sebagai *output* dan *IC* L293D motor

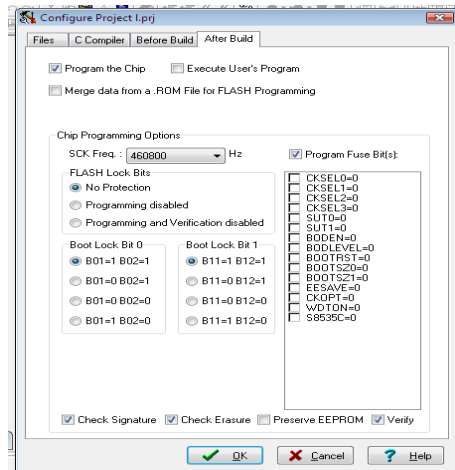
8. Pilih **Ok**.

4.1.3. Pengisian Program Pada

Mikrokontroler

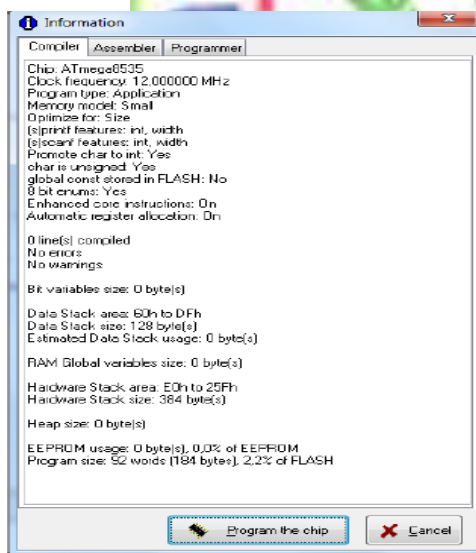
Pengisian program dilakukan dengan menggunakan *downloader K-125R Uno*. Adapun proses *mendownload* program ke *chip* mikrokontroler ATmega 8535 adalah sebagai berikut :

- a. Hubungkan terlebih dahulu dari perangkat komputer ke *downloader K-125R Uno* yang telah terhubung dengan Mikrokontroler. Kemudian buka program yang telah dibuat pada *CodeVision AVR*.
- b. Pilih **Project** lalu *configure*, kemudian pilih menu **After build** dan aktifkan **Program the chip**. Pilih **OK**



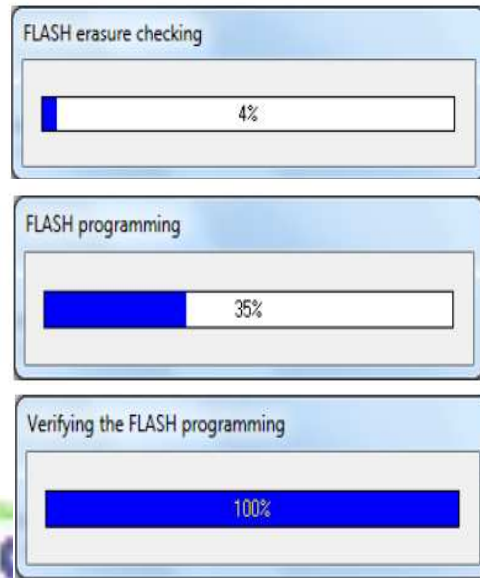
Gambar 4.8 Mengaktifkan *toolbar after build*

- c. Setelah itu pilih **Project**, klik **Build**
- d. Setelah itu akan muncul tampilan seperti gambar



Gambar 4.9 Tampilan setelah di *Compile*

- e. Pilih **Program the chip**, maka program akan *terdownload* ke mikrokontroler.

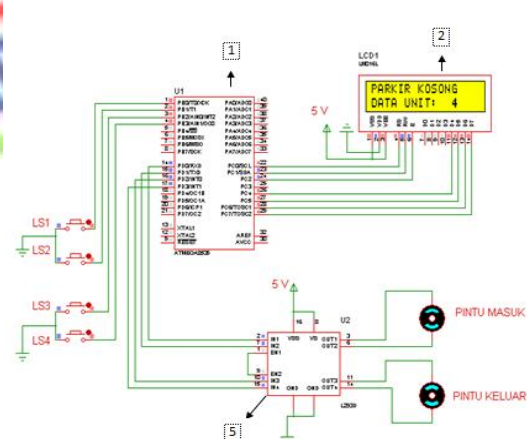


Gambar 4.10 Proses Pengisian Program

- f. Setelah semua proses di atas selesai, maka mikrokontroler siap untuk digunakan.

4.1.4. Pembuatan Simulasi Rangkaian

Penggunaan aplikasi *Proteus 8 Personal* bertujuan untuk meminimalisir kesalahan yang terjadi pada rangkaian *prototype* yang akan dibuat sebelum dibuat kedalam bentuk aslinya. Dengan demikian setelah rangkaian dibuat kemudian program *didownload* ke *proteus* dan dapat disimulasikan cara kerja sistem yang dibuat. Apabila berhasil disimulasikan berarti program yang telah dibuat tidak akan mengalami masalah bila *didownload* kedalam *chip* mikrokontroler ATmega 8535 yang sebenarnya dan penelitian dapat dilanjutkan kedalam tahap pembuatan dan perakitan komponen



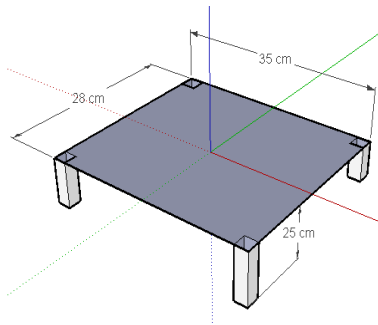
Gambar 4.11 Tampilan Simulasi Pada Proteus

Keterangan:

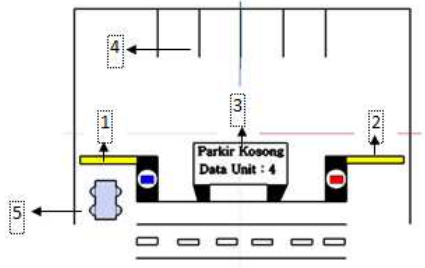
1. Mikrokontroler ATmega 8535
2. Liquid Crystal Display
3. Limit Switch
4. Motor DC
5. IC Driver Motor DC

4.1.5. Perancangan Sistem Mekanik

Tahap ini merupakan tahap awal yang harus dikerjakan. Yang dilakukan dalam perancangan konstruksi mekanik ini adalah merancang tempat untuk meletakkan alat yang berupa *prototype*.



Gambar 4.12 Letak Prototype



Gambar 4.13 Gambar Design Palang Parkir

Keterangan Gambar :

1. Pintu masuk
2. Pintu keluar
3. Tampilan LCD
4. Area Parkir
5. Mobil masuk

4.2. Tahap Pengujian Program

Tahap pengujian penulis lakukan untuk menguji program yang telah dibuat dan dimasukkan ke mikrokontroler ATmega 8535. Pada pengujian ini penulis membatasi data parkiran mobil yang kosong sebanyak 4 unit. Adapun langkah – langkah dalam pengujian yang dilakukan yaitu :

- a. Hubungkan *downloader K-125R Uno* pada mikrokontroler ke perangkat komputer.
- b. Setelah itu akan muncul tulisan pada layar LCD pada *prototype*. Tulisan pada LCD muncul berdasarkan program yang telah kita buat.



Gambar 4.14 Tampilan LCD

- c. Pada gambar 4.14 merupakan tampilan awal LCD pada lokasi parkir. Jumlah kapasitas data parkir yang kosong sebanyak 4 unit.
- d. Ketika ada mobil masuk ke areal parkir melalui pintu masuk, kemudian menekan sensor Ls1 maka akan mengakibatkan palang terbuka. Tampilan LCD untuk data parkir akan berkurang satu. Proses ini akan terjadi secara terus menerus ketika ada *input* yang diterima oleh Ls1 atau tertekan ketika ada mobil yang masuk.



Gambar 4.15 Palang Terbuka dan Pengurangan Data Parkir

- e. Kemudian ketika mobil menekan sensor Ls2 yang berada pada palang masuk maka akan mengakibatkan palang kembali menutup. Selanjutnya Mobil menuju lokasi parkir yang kosong.



Gambar 4.16 Palang Tertutup

- f. Sebaliknya ketika ada mobil keluar dari area parkir melalui pintu keluar menekan Ls3 maka akan mengakibatkan palang terbuka. Tampilan LCD untuk data parkir akan bertambah satu.



Gambar 4.17 Palang Keluar Terbuka

- g. Ketika mobil menekan sensor Ls4 maka akan mengakibatkan palang kembali menutup. Mobil keluar meninggalkan area parkir.



Gambar 4.18 Palang Keluar Tertutup

- h. Dan ketika kapasitas area parkir penuh maka tampilan LCD akan seperti pada gambar berikut.



Gambar 4.19 Data Parkir Pada Kondisi Penuh



Gambar 4.20 Prototype

4.3. Listing Program

Berikut adalah listing program pada mikrokontroler ATmega 8535 untuk mengatur palang parkir otomatis.

1. Listing program input.

```

while (1)
{
// Place your code here
if (s1==0) {data--; // jika s1 ditekan maka
data parkir akan berkurang satu
if (data<1) { data=0;}
PORTD.0=1; // motor beputar kekanan
(Palang membuka)
PORTD.1=0;
delay_ms(200);
PORTD.0=0; // motor berhenti
PORTD.1=0;
delay_ms(4000);
}
if(s2==0) //jika s2 ditekan maka
{
PORTD.0=0; // motor berputar kekiri(palang
menutup)
PORTD.1=1;
delay_ms(200);
PORTD.0=0; // motor berhenti
PORTD.1=0;
delay_ms(4000);
}
if (s3==0){data++; // jika s3 ditekan maka
data parkir akan bertambah satu
if (data>=4){data=4;}
PORTD.2=1; // motor beputar kekanan
(Palang membuka)
PORTD.3=0;
delay_ms(200);
PORTD.2=0; // motor berhenti
PORTD.3=0;
delay_ms(4000);
}
}
if(s4==0) //jika s4 ditekan maka
{

```

```
PORTD.2=0; // motor akan berputar
kekiri(palang menutup)
PORTD.3=1;
delay_ms(200);
PORTD.2=0; // motor berhenti
PORTD.3=0;
delay_ms(4000);
}
if (data==0)
{
lcd_gotoxy(0,0);
lcd_putsf("PARKIR PENUH ");
lcd_gotoxy(0,1);
lcd_putsf("DATA UNIT:");
itoa(data,tampil);
lcd_gotoxy(12,1);
lcd_puts(tampil);
}
else
{
lcd_gotoxy(0,0);
lcd_putsf("PARKIR KOSONG ");
lcd_gotoxy(0,1);
lcd_putsf("DATA UNIT:");
itoa(data,tampil);
lcd_gotoxy(12,1);
lcd_puts(tampil);
}
};
}
```

Keterangan program :

Pada saat program dijalankan, program akan terus mengecek input sampai terjadi penekanan pada sensor. Apabila penekanan sensor belum terjadi maka secara otomatis program akan *looping* terus sampai terjadi penekanan sensor. Dalam hal ini *input* yang digunakan pada mikrokontroler menggunakan *port B* (PinB.0 – PinB.4). Sedangkan outputnya menggunakan *port D* (PinD.0 – PinD.4) untuk menggerakkan motor *DC* dan *port C* (PinC.0 – PinC.7) tampilan pada *Liquid Crystal Display*.

Pada kondisi pertama, ketika sensor 1 ditekan oleh mobil yang masuk, program akan memberikan sinyal untuk mengaktifkan *PortD.0* dan *PortD.1* sehingga motor akan berputar ke atas selama 200 ms. Kondisi tersebut mengakibatkan palang parkir membuka selama 4000 ms. Kemudian data parkir kosong akan berkurang satu yang akan ditampilkan oleh *LCD*.

Pada kondisi kedua, ketika sensor 2 ditekan oleh mobil yang masuk, program akan

memberikan sinyal untuk mengaktifkan *PortD.0* dan *PortD.1* maka yang terjadi motor akan berputar ke bawah selama 200 ms. Kondisi tersebut akan mengakibatkan palang parkir menutup selama 4000 ms.

Pada kondisi ketiga, ketika sensor 3 ditekan oleh mobil yang keluar program akan memberikan sinyal untuk mengaktifkan *PortD.2* dan *PortD.3* maka yang terjadi motor akan bergerak ke atas selama 200 ms. Kondisi tersebut mengakibatkan palang parkir membuka selama 4000 ms. Data parkir kosong akan bertambah satu yang akan ditampilkan oleh *LCD*.

Pada kondisi keempat, Ketika sensor 4 ditekan oleh mobil yang keluar Program akan memberikan sinyal untuk mengaktifkan *PortD.2* dan *PortD.3* maka yang terjadi motor akan bergerak ke bawah selama 200 ms. Kondisi tersebut mengakibatkan palang parkir menutup selama 4000 ms. Kejadian seperti itu akan terus berulang selama terjadi penekanan pada sensor.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian dan pengujian yang telah dilakukan penulis dapat menyimpulkan yaitu secara keseluruhan sistem palang parkir otomatis menggunakan ATmega 8535 dapat dioperasikan sesuai dengan program yang dibuat.

5.2 Saran

Adanya segala kekurangan serta kelemahan dalam pembuatan rancang bangun *prototype* palang parkir otomatis ini dirasakan perlu adanya perbaikan Saran yang bisa penulis berikan antara lain :

- Apabila *prototype* ini ingin diaplikasikan dalam bentuk nyatanya maka terdapat beberapa komponen yang harus diganti seperti halnya sensor *limit switch*, motor *DC*, serta sistem mekanik palang parkir.
- Untuk pengembangan lebih lanjut jumlah kapasitas parkir dapat diubah sesuai dengan kebutuhan yang diinginkan.
- Agar kinerja motor *DC* lebih maksimal maka pengaturan waktu serta pergerakan motor harus benar – benar diperhatikan sehingga motor *DC* mampu melaksanakan tugasnya seperti yang diharapkan.

DAFTAR PUSTAKA

- Ari Beni Santoso,2013, Jurnal FEMA, Volume 1, Nomor 1, Hal 18
- Ardi Winoto.2008.Mikrokontroler AVR ATMEGA 8/16/32/8535 dan Pemrogramandengan Bahasa C. Bandung : PT. Informatika Bandung.
- Buku Manual Modul *Liquid Crystal Display* LM016L
- Buku Manual *K-125 R UNO Board AVR Programmer*
- Dian Wirdasari,2010, Jurnal SAINTIKOM, Volume 8, Nomor 1, Hal 397
- Donna Marissa. 2007. “Rancang Bangun Miniatur Mesin Pembengkok Pipa Otomatis Berbasis Mikrokontroler AT89C51 (*Software*)”.Skripsi tidak dipublikasikan. Mahasiswa Teknik Mekatronika, Politeknik Caltex Riau
- Goenawan,Kian.2008. Izin Beres Bisnis Sukses.: Pustaka Grhatama,Yogyakarta
- Teknik Elektronika.com/IC-L293D, 2014. Diakses pada 06 Maret 2016
- Heryanto, Ary dan Wisnu, Adi. 2008. Pemrograman Bahasa C untuk Mikrokontroler ATMEGA8535. Yogyakarta
- <http://joaldera.com/2008/10/mikrokontoler-atmega-8535/>. Diakses pada 04 Maret 2016
- <http://www.mikorn123.com/index.php/Tutorial-AVR/Arsitektur-Mikrokontoler-AVR.html>. Diakses pada 04 Maret 2016
- <https://www.google.com/search?q=adaptor&ie=utf-8&oe=utf-8&client=firefox-b-ab#q=jurnal+tentang+adaptor> Diakses pada 22 Juni 2016
- Robyn Frannado. 2010.“Aplikasi Robot Mobil Dalam Pengembangan Prototype Robot Mobil Wisata”. Mahasiswa Teknik Elektro, Universitas Indonesia
- Tianur, Praktikum Mikrokontroler AVR, Politeknik Caltex Riau,2014
- www.inverterplus.com/dayatahaninverter. Diakses pada 22 Juni 2016.