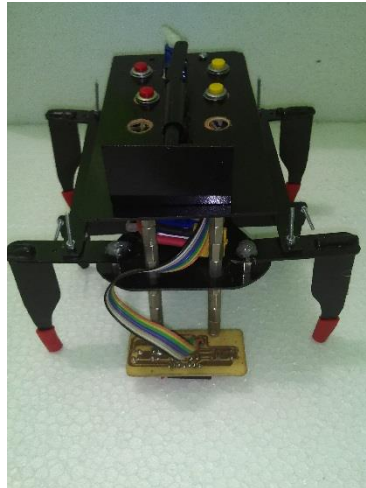




**PROTOTIPE ROBOT *LINE FOLLOWER* BERKAKI  
SEBAGAI PENGANTAR MAKANAN DI RESTORAN  
BERBASIS MIKROKONTROLER ATMEGA328  
PROYEK AKHIR**

Diajukan Kepada Fakultas Teknik Universitas Negeri Yogyakarta Untuk  
Memenuhi Sebagian Persyaratan Guna Memperoleh Gelar Ahli Madya  
Program Studi Teknik Elektronika



**OLEH:  
APRILIA IMAM UDIN  
NIM. 13507134033**

**PROGRAM STUDI TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA  
2016**

**LEMBAR PERSETUJUAN  
PROYEK AKHIR**

**PROTOTIPE ROBOT *LINE FOLLOWER* BERKAKI  
SEBAGAI PENGANTAR MAKANAN DI RESTORAN  
BERBASIS MIKROKONTROLER ATMEGA328**

Oleh

**APRILIA IMAM UDIN**

**NIM. 13507134033**



Yogyakarta, 21 Desember 2016

Mengetahui  
Kaprosdi Teknik Elketronika



**Dra. Sri Waluyanti, M.Pd.**  
**NIP. 19581218 198603 2 001**

Menyetujui  
Pembimbing



**Adi Dewanto, S.T., M.Kom.**  
**NIP. 19721228 200501 1 001**

**LEMBAR PENGESAHAN  
PROYEK AKHIR**

**PROTOTIPE ROBOT *LINE FOLLOWER* BERKAKI  
SEBAGAI PENGANTAR MAKANAN DI RESTORAN  
BERBASIS MIKROKONTROLER ATMEGA328**

Dipersiapkan dan Disusun Oleh:

**APRILIA IMAM UDIN  
13507134033**

Telah dipertahankan didepan Dewan Penguji Proyek Akhir  
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA  
Pada tanggal 28 Oktober 2016

Dan Dinyatakan Telah Memenuhi Syarat Guna Memperoleh Gelar  
**Ahli Madya Teknik**

**SUSUNAN DEWAN PENGUJI**

Jabatan	Nama Lengkap dan Gelar	Tandatangan	Tanggal
1. Ketua Penguji	Adi Dewanto, S.T., M .Kom.		21-12-2016
2. Sekretaris	Handaru Jati, Ph.D.		21-12-2016
3. Penguji Utama	Totok Sukardiyono, M.T.		21-12-2016

Yogyakarta, 21 Desember 2016

Mengetahui,

Dekan Fakultas Teknik UNY



**Dr. Widarto, M.Pd.**

NIP. 19631230 198812 1 001 

## SURAT PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini :

Nama : Aprilia Imam Udin

NIM : 13507134033

Program Studi : Pendidikan Teknik Elektronika

Judul Proyek Akhir : PROTOTIPE ROBOT *LINE FOLLOWER* BERKAKI  
SEBAGAI PENGANTAR MAKANAN DI RESTORAN  
BERBASIS MIKROKONTROLER ATMEGA328.

Menyatakan bahwa proyek akhir ini adalah hasil pekerjaan saya sendiri dan sepanjang pengetahuan saya, tidak berisi materi yang ditulis orang lain sebagai persyaratan penyelesaian studi di Universitas Negeri Yogyakarta atau perguruan tinggi lain, kecuali bagian-bagian tertentu yang saya ambil sebagai acuan dengan mengikuti tata cara dan penulisan karya ilmiah yang lazim. Jika ternyata terbukti pernyataan ini benar, sepenuhnya menjadi tanggung jawab saya.

Yogyakarta, 28 Oktober 2016

Yang Menyatakan,



**Aprilia Imam Udin**  
NIM.13507134033

**PROYEK AKHIR**  
**PROTOTIPE ROBOT *LINE FOLLOWER* BERKAKI**  
**SEBAGAI PENGANTAR MAKANAN DI RESTORAN**  
**BERBASIS MIKROKONTROLER ATMEGA328**

Oleh :  
**Aprilia Imam Udin**  
**1350714033**

**ABSTRAK**

Prototipe Robot *Line Follower* Berkaki merupakan inovasi pada bidang pelayanan makanan khususnya menggunakan robot berkaki. Robot pelayan yang sudah ada kebanyakan menggunakan roda sebagai penggerakannya, sedangkan pada proyek akhir ini prototipe robot pelayan yang dibuat menggunakan kaki sebagai penggerakannya. Proyek akhir ini bertujuan (1) membuat Prototipe Robot *Line Follower* Berkaki sebagai pelayan makanan di restoran, (2) membuat program Prototipe Robot *Line Follower* Berkaki supaya robot pelayan dapat mengikuti jalur dan menaruh nampan berisi makanan ke meja pelanggan, (3) mengetahui unjuk kerja Prototipe Robot *Line Follower* Berkaki sebagai pengantar makanan di restoran.

Perancangan Prototipe Robot *Line Follower* Berkaki ini menggunakan metode rancang bangun yang terdiri atas beberapa tahap yaitu (1) analisa kebutuhan, (2) perancangan rangkaian, (3) langkah pembuatan alat, (4) pengujian alat dan (5) pengambilan data. Prinsip kerja dari Prototipe Robot *Line Follower* Berkaki yaitu mengantar nampan makanan ke meja pelanggan dengan mengikuti garis sesuai nomor meja. Perangkat ini tersusun dari 3 bagian utama : (1) bagian *input*, dengan komponen utama yakni sensor *infrared*, photodiode sebagai sensor garis dan *push button* sebagai inputan berupa perintah langsung, (2) bagian mikrokontroler, dengan komponen utama ATmega328, ATmega328 disini adalah sebagai proses dari inputan yang masuk dan akan diteruskan ke *output*, (3) bagian *output*, dengan komponen utama berupa motor servo yang digunakan sebagai penggerak dari keseluruhan sistem yang ada.

Berdasarkan dari hasil pengujian dapat disimpulkan bahwa Prototipe Robot *Line Follower* Berkaki dapat bekerja sesuai dengan spesifikasi yang telah ditentukan yaitu (1) Robot pelayan dapat mengantar nampan yang berisi makanan ke meja yang telah ditentukan, (2) Perangkat lunak yang digunakan untuk program prototipe robot *line follower* berkaki menggunakan bahasa C dapat bekerja dengan baik sesuai dengan instruksinya (3) Robot pelayan yang dilengkapi dengan sensor *infra red* sebagai penunjuk arah dan motor servo sebagai penggerak kakinya dapat bekerja dengan baik dan robot dapat mengantar makanan ke meja yang ditentukan.

Kata Kunci : ATmega328, Prototipe Robot *Line Follower* Berkaki.

## HALAMAN PERSEMBAHAN

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayahnya, telah memberikanku kekuatan, membekali dengan ilmu dan semua nikmat yang telah dikaruniakan. Atas ridho-NYA akhirnya Laporan Proyek Akhir yang sederhana ini dapat terselesaikan.

Kupersembahkan karya sederhana ini kepada dua insan yang sangat aku kasihi dan sayan, Ibu, Bapak. Sebagai tanda bakti, hormat, dan rasa terima kasih yang tiada terhingga kepada Ibu dan Bapak yang telah memberikan kasih sayang, segala dukungan, dan cinta kasih yang tiada terhingga tercurah kepadaku yang tiada mungkin dapat aku balas, hanya dengan selembar kata cinta dari anakmu dan persembahan. Semoga ini menjadi langkah awal yang untk membuat Ibu dan Bapak bahagia karena kusadar, selama ini belum bisa berbuat yang lebih. Untuk Ibu dan Bapak yang selalu mencurahkan rasa kasih dan sayangnya, dan membuatku selalu termotivasi dan menjadikan diri untuk lebih baik.

Seluruh dosen Pengajar Teknik Elektronika, terima kasih banyak untuk semua ilmu, didikan dan pengalaman yang sangat berarti yang telah kalian berikan. Untuk Dosen Pembimbing saya Adi Dewanto, S.T.,M.Kom. terima kasih bimbingan dan bantuan selama ini, atas nasihat dan pelajaran yang saya dapatkan, saya tidak akan lupa atas bantuan dan kesabaran dari bapak.

Untuk teman-temanku di kelas B Teknik Elektronika 2013 yang tiga tahun terakhir ini bersama kalian, kebersamaan yang kita buat takan pernah terlupakan. Dangin Dauh, Slamet, dan Bondan aku ucapkan terima kasih atas bantuan yang

telah diberikan sepenuhnya. Anak kontrakan dan Anak Kost Bu wardo aku ucapkan banyak terima kasih untuk tempat indah yang kita jalani dan motivasi yang diberikan. Dahlia Nurjanah aku ucapkan banyak terima kasih untuk kebersamaan, motivasi dan dorongan yang telah diberikan. Sungguh aku bahagia bersama kalian memiliki kenangan indah dalam setiap bait paragraf kisah persahabatan dan cinta kita.

## **MOTTO**

1. Hargailah waktumu karena waktu yang lewat takan pernah kembali lagi dan terulang sama lagi.
2. Bersungguh-sungguh ketika melakukan kewajiban dan doa, jangan hanya karena untuk menggugurkan kewajiban saja.

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kehadiran Allah SWT, yang telah melimpahkan rahmat, taufik dan karunia-Nya, sehingga dapat menyelesaikan laporan Proyek Akhir yang berjudul “Robot Pelayan Restoran”. Tujuan dari penyusunan Proyek Akhir ini adalah sebagai syarat kelulusan pada program studi teknik Elektronika D3 Universitas Negeri Yogyakarta.

Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, maka penulisan laporan Tugas Akhir ini tidak akan lancar. Oleh karena itu pada kesempatan ini, izinkanlah penulis menyampaikan ucapan terima kasih kepada :

1. Bapak dan Ibu yang selalu memberikan semangat dan doa-doanya.
2. Bapak Dr. Widarto, M.Pd. selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta
3. Bapak Dr. Fatchul Arifin, M.T. selaku Ketua Jurusan Pendidikan Teknik Elektronika.
4. Ibu Dra. Sri Waluyanti, M.Pd. selaku Ketua Program Proyek Akhir Jurusan Teknik Elektronika Universitas Negeri Yogyakarta
5. Bapak Adi Dewanto, S.T.,M.Kom selaku Dosen Pembimbing Penyusun Laporan Proyek Akhir.
6. Seluruh Dosen dan Karyawan di Jurusan Teknik Elektronika Fakultas Teknik Universitas Negeri Yogyakarta, yang telah

mendidik dan memotivasi selama kuliah di Universitas Negeri Yogyakarta.

7. Keluarga besar yang telah memberikan kasih sayang dan motivasi selama ini.
8. Dahlia Nurjanah yang telah memberikan waktu, dorongan dan motivasi selama ini.
9. Dangin Dauh, slamet dan bondan terima kasih atas ide, bantuan dan masukan – masukannya, semoga selalu sukses.
10. Teman – teman B elektronika, anak kontrakan dan anak kost bu warto atas kenangan dan kebersamaan yang telah kita lewati bersama.
11. Semua pihak yang tidak dapat disebutkan satu per satu yang telah membantu dalam penyelesaian laporan ini.

Akhirnya disadari sepenuhnya bahwa dalam penyusunan Proyek Akhir ini masih jauh dari kesempurnaan sehingga saran, masukan dan kritik sangat diperlukan demi kesempurnaan, semoga penyusunan Proyek Akhir ini dapat memberikan kontribusi bagi semua pihak.

Yogyakarta, Oktober 2016  
Penulis

Aprilia Imam Udin

## DAFTAR ISI

<b>HALAMAN SAMPUL</b> .....	i
<b>LEMBAR PERSETUJUAN</b> .....	ii
<b>LEMBAR PENGESAHAN</b> .....	iii
<b>LEMBAR PERNYATAAN KEASLIAN</b> .....	iv
<b>ABSTRAK</b> .....	v
<b>HALAMAN PERSEMBAHAN</b> .....	vi
<b>MOTTO</b> .....	vii
<b>KATA PENGANTAR</b> .....	viii
<b>DAFTAR ISI</b> .....	xi
<b>DAFTAR GAMBAR</b> .....	xiv
<b>DAFTAR TABEL</b> .....	xvi
<b>DAFTAR LAMPIRAN</b> .....	xvii
<b>BAB I PENDAHULUAN</b> .....	1
A. Latar Belakang Masalah .....	1
B. Identifikasi Masalah .....	2
C. Batasan Masalah .....	2
D. Rumusan Masalah .....	2
E. Tujuan Proyek Akhir .....	3
F. Manfaat Proyek Akhir .....	3
G. Keaslian Gagasan .....	4
<b>BAB II PENDEKATAN PEMECAHAN MASALAH</b> .....	6
A. Robot .....	6
1. Pengertian .....	6
2. Tipe Robot .....	6

3. Karakteristik Robot .....	7
B. Motor Servo .....	8
C. Baterai Lippo .....	13
D. Badan Robot .....	15
E. Sensor Garis .....	16
1. Photodioda .....	17
2. Infra Merah ( <i>infrared</i> ) .....	18
F. Mikrokontroler .....	20
1. Mikrokontroler ATmega328 .....	22
2. Konfigurasi Pin ATmega328 .....	26
3. Sistem Minimum .....	26
G. Catu Daya .....	27
H. Arduino IDE .....	30
<b>BAB III PERANCANGAN DAN PEMBUATAN ALAT .....</b>	<b>33</b>
A. Identifikasi Kebutuhan .....	33
B. Analisis Kebutuhan .....	34
1. Baterai ( <i>power supply</i> ) .....	34
2. Motor Servo .....	34
3. Sensor Photodioda .....	34
4. Sistem Minimum ATmega328 .....	34
5. Tombol Navigasi .....	34
C. Perancangan Alat .....	35
1. Perancangan Mekanik Robot .....	36
D. Perencanaan Rangkaian .....	37
1. Catu Daya .....	37
2. Rangkaian Sistem Minimum mikrokontroler ATmega328 .....	39
3. Rangkaian Servo Kontroler .....	39
E. Langkah Pembuatan Alat .....	41
1. Pembuatan PCB .....	41
2. Pemasangan Komponen .....	42

3. Pembuatan Kerangka Robot .....	43
F. Perancangan Perangkat Lunak .....	43
1. Program .....	43
2. Flowchart .....	44
G. Spesifikasi Alat .....	45
H. Pengujian Alat .....	47
1. Uji Fungsional .....	47
2. Uji Unjuk Kerja .....	47
I. Pengoperasian Alat .....	47
<b>BAB IV PENGUJIAN DAN PEMBAHASAN .....</b>	<b>49</b>
A. Hasil Pengujian Fungsional .....	49
1. Pengujian Tegangan Catu Daya .....	49
2. Pengujian Tegangan Sistem Minimum Saat Tanpa Beban dan Saat ada Beban .....	50
3. Pengujian Sensor Infrared .....	51
4. Pengujian Gerakan Motor Servo Robot .....	54
5. Pengujian Unjuk Kerja Keseluruhan .....	57
B. Pembahasan .....	58
1. Hardware .....	59
2. Software .....	59
3. Unjuk Kerja Keseluruhan .....	64
C. Cara Kerja Prototipe Robot <i>Liner Follower</i> Berkaki .....	65
<b>BAB IV KESIMPULAN DAN SARAN .....</b>	<b>67</b>
A. Kesimpulan .....	67
B. Keterbatasan Alat .....	69
C. Saran .....	69
<b>DAFTAR PUSTAKA .....</b>	<b>71</b>
<b>LAMPIRAN .....</b>	<b>72</b>

## DAFTAR GAMBAR

Gambar 1. Motor Servo .....	9
Gambar 2. Pulsa Kendali Motor Servo .....	10
Gambar 3. Motot Servo 180 <sup>0</sup> .....	11
Gambar 4. Konstruksi Motor Servo 180 <sup>0</sup> .....	11
Gambar 5. Blok Fungsi Diagram Motor Servo .....	13
Gambar 6. Baterai Lithium-Polymer .....	14
Gambar 7. PCB .....	16
Gambar 8. Sensor Garis atau <i>Proximity Sensor</i> .....	16
Gambar 9. Sensor Photodiode .....	18
Gambar 10. Bentuk <i>Infrared Led</i> .....	19
Gambar 11. Blok Diagram ATmega328 .....	23
Gambar 12. Konfigurasi Pin ATmega328 .....	23
Gambar 13. Contoh IC Regulator .....	28
Gambar 14. IC LM 7805 .....	29
Gambar 15. Logo Arduino .....	31
Gambar 16. Arduino IDE .....	31
Gambar 17. Blok Diagram Prototipe Robot <i>Line Follower</i> Berkaki .....	35
Gambar 18. Rancangan Mekanik Robot Tampak Samping .....	36
Gambar 19. Rangkaian Catu Daya .....	38
Gambar 20. Baterai Lippo .....	38

Gambar 21. Rangkaian Skematik Mikrokontroler ATmega328 .....	39
Gambar 22. Rangkaian Servo Kontrol .....	40
Gambar 23. Layout Sistem Minimum ATmega328 .....	41
Gambar 24. Layout Sensor .....	41
Gambar 25. Flowchart Cara Kerja Robot <i>Line Follower</i> Berkaki .....	44
Gambar 26. Diagram Blok Cara Kerja .....	66

## DAFTAR TABEL

Tabel 1. Karakteristik Regulator Tegangan Positif 78xx .....	29
Tabel 2. Pengukuran Tegangan Regulator 7805 .....	50
Tabel 3. Pengukuran Tegangan Sistem Minimum Saat Tanpa Beban dan Saat Ada Beban .....	50
Tabel 4. Pengukuran Sensor <i>Infrared</i> Terhadap Garis Putih .....	51
Tabel 5. Pengukuran Sensor <i>Infrared</i> Terhadap Garis Hitam .....	52
Tabel 6. Pengujian Motor Servo Kaki Samping .....	55
Tabel 7. Pengujian Motor Servo Kaki Bawah .....	57
Tabel 8. Hasil Pengujian Unjuk Kerja Robot .....	59
Tabel 9. Hasil Pengujian Unjuk Kerja Robot .....	65
Tabel 10. Program Sistem Minimum .....	67

## DAFTAR LAMPIRAN

Lampiran 1. Datasheet ATmega328 .....	73
Lampiran 2. Datasheet Photodiode .....	85
Lampiran 3. Lampiran Sensor <i>Infrared</i> .....	92
Lampiran 4. ATmega328 Arduino Pin <i>Mapping</i> .....	100
Lampiran 5. Desain PCB Mikrokontroler ATmega328 .....	101
Lampiran 6. Desain PCB Motor Servo .....	102
Lampiran 7. Prototipe Robot <i>Line Follower</i> Berkaki .....	103
Lampiran 8. Rangkaian Sistem Minimum Robot Pengantar Makanan .....	104
Lampiran 9. Program .....	105

# BAB I

## PENDAHULUAN

### A. Latar Belakang Masalah

Usaha tempat makan, dewasa ini menunjukkan perkembangan yang relatif pesat terbukti semakin banyaknya tempat makan lokal seperti Raja Pedas, Sambel Layah maupun tempat makan asing seperti McDonald's merambah dikota jogja. Hal tersebut mengindikasikan bahwa intensitas persaingan dalam bisnis tempat makan semakin kuat. Jenis tempat makan seperti ini umumnya berada dilokasi-lokasi strategis dengan tampilan yang berbeda-beda. Ini semakin memperketat persaingan untuk menarik pelanggan. Sehingga bagi konsumen yang datang dan makan ditempat makan seperti ini, sedikit tidak terpengaruh dan tidak jarang datang kembali untuk melakukan pembelian (*Repeat Buying* ).

Seiring dengan ketatnya persaingan kebutuhan pokok seperti makan, dalam Bidang Teknologi pun terlihat kemajuan yang sangat pesat khususnya di Indonesia di Bidang Robotika. Teknologi Robotika yang berkembang di Indonesia banyak jenisnya seperti robot *line follower* yang dijadikan sebagai ajang permainan maupun perlombaan, ini bisa dimanfaatkan sebagai Robot *Line Follower* berkaki pelayan makanan maupun minuman ditempat makan, disamping itu masih jarang penggunaan Robot Pelayan makanan di Indonesia. Prototipe Robot *Line Follower* Berkaki yang dibuat adalah menggunakan servo sebagai penggeraknya, tidak seperti line follower pada umumnya yang menggunakan roda sebagai penggeraknya. Pada prinsipnya tujuan dari Prototipe Robot *Line Follower* Berkaki ini adalah mengantar prtototipe

nampan makanan atau minuman ke meja. Hal ini bermanfaat melihat banyaknya pembangunan dikota besar dan persaingan tempat makan dikota besar, seperti tempat makan baru atau lama, tempat makan lokal ataupun tempat makan asing. Ini juga sebagai daya tarik tersendiri untuk menarik minat konsumen untuk merasakan bagaimana rasanya makan dilayani oleh Robot *Line Follower* Berkaki. Oleh karena itu muncul inovasi untuk membuat Prototipe Robot *Line Follower* Berkaki yang dimanfaatkan untuk melayani konsumen khususnya sebagai pengantar makanan.

## **B. Identifikasi Masalah**

Dari uraian latar belakang diatas, maka dapat dibuat suatu identifikasi masalah sebagai berikut :

1. Tingginya tingkat persaingan restoran di Indonesia
2. Belum adanya penggunaan Robot Berkaki sebagai pelayan restoran di Indonesia.

## **C. Batasan Masalah**

Untuk menghadapi persaingan Rumah Makan yang semakin tinggi dan belum adanya penggunaan robot berkaki sebagai pelayan restoran di Indonesia maka dibuatlah inovasi baru berupa Prototipe Robot *Line Follower* Berkaki.

## **D. Rumusan Masalah**

Dari identifikasi yang ada, maka dapat ditarik beberapa rumusan masalah, yaitu:

1. Bagaimana membuat Prototipe Robot *Line Follower* Berkaki sebagai pengantar makanan direstoran.

2. Bagaimana membuat Program Robot Line Follower Berkaki supaya dapat mengikuti jalur dan menaruh makanan pada meja sesuai nomor yang diperintahkan.
3. Bagaimana unjuk kerja Prototipe Robot *Line Follower* Berkaki sebagai pengantar makanan di restoran.

#### **E. Tujuan Proyek Akhir**

Tujuan yang ingin dicapai dalam pembuatan Proyek Akhir ini adalah :

1. Membuat Prototipe Robot *Line Follower* Berkaki sebagai pengantar makanan direstoran.
2. Membuat program Prototipe Robot *Line Follower* Berkaki supaya dapat mengikuti jalur dan menaruh makanan pada meja sesuai nomor yang diperintahkan.
3. Mengetahui unjuk kerja Prototipe Robot *Line Follower* Berkaki sebagai pengantar makanan direstoran.

#### **F. Manfaat Proyek Akhir**

Pembuatan proyek akhir ini diharapkan dapat bermanfaat bagi mahasiswa, lembaga pendidikan, dan industri. Adapun manfaat yang diharapkan dari pembuatan tugas akhir ini antara lain :

1. Bagi mahasiswa
  - a. Sebagai tolak ukur individual setelah mendapatkan ilmu dari bangku kuliah dan kehidupan sehari-hari untuk diimplementasikan dalam bentuk suatu alat.

- b. Untuk mengaplikasikan ilmu yang didapat selama di bangku kuliah dan menerapkan ilmunya secara nyata.
  - c. Dapat digunakan sebagai bahan referensi atau pembelajaran dan penambah wawasan tentang Robot Pelayan serta sebagai kajian untuk pengembangan selanjutnya.
  - d. Sebagai bentuk kontribusi terhadap Universitas dan pengabdian kepada masyarakat dalam bentuk karya alat yang bermanfaat.
2. Bagi Prodi Teknik Elektronika
- a. Sebagai wujud dari perkembangan Ilmu Pengetahuan dan Teknologi (IPTEK).
  - b. Sebagai parameter kualitas dan kuantitas lulusan mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta.
3. Bagi Dunia Industri

Dengan terciptanya Prototipe Robot *Line Follower* Berkaki ini diharapkan dapat tercipta suatu alat yang dapat membantu mengembangkan dan meningkatkan produk suatu industri.

### **G. Keaslian Gagasan**

Otomasi robot telah lama mendominasi sektor faktor. Akan tetapi, akhir-akhir ini banyak robot yang sudah melayani dan menghibur pengunjung di restoran di China, Taiwan, Jepang, Dan Amerika Serikat (AS). Sebagai contoh robot restoran yang sudah ada seperti Noodle Bot (China), Sushi Bot

(Japan), Ice Cream Bot (jepang), Hamburger Bot (AS), Pizza Bot, Bertender Bot (AS), Chef Bot (Jepang), Waiter Bot (Thailand). (Ayunda W Safitri, 2013).

Dari semua robot yang telah ada, penggeraknya adalah menggunakan roda.

Perbedaan dengan tugas Akhir yang penulis buat adalah penulis mengganti penggerak robot pelayan restoran yang yang kebanyakan beroda menjadi berkaki. Gagasan ini asli dari gagasan pribadi yang terinspirasi dari banyaknya masyarakat yang hobi memainkan line follower. Oleh karena itu inovasi ini gabungan dari robot berkaki dan *line follower*.

## **BAB II**

### **PENDEKATAN PEMECAHAN MASALAH**

Dalam pembuatan Prototipe Robot *Line Follower* Berkaki ini diperlukan pemecahan masalah yang meliputi kajian - kajian teori perangkat *hardware* dan *software* yang digunakan.

#### **A. Robot**

##### **1. Pengertian**

Istilah robot berasal dari Czech , *robota*, yang berarti bekerja. Menurut arti bahasa, Robot adalah sebuah alat mekanik yan dapat melakukan tugas fisik, baik menggunakan pengawasan dan kontrol manusia, atau menggunakan program yang telah didefinisikan terlebih dahulu.

Robot diperkenalkan pertama kali oleh orang berkebangsaan Austria yaitu Wright Karel Capek pada tahun 1920 melalui sandiwara yang dibuatnya yaitu R.U.R (*Rossum's Universal Robots*). Dalam sandiwara ini menceritakan seorang tokoh ilmu pengetahuan yang bernama Rossum berhasil menciptakan bahan tiruan daging dan tulang melalui proses biologi dan elektronik. Tujuan pembuatan bahan itu adalah mewujudkan impiannya membuat kehidupan buatan. Namun, ekspeeimen Rosuum gagal. (Winanrno dan Deni Afrianto, 2010).

##### **2. Tipe Robot**

Ada bermacam – macam tipe dan desain robot, hingga saat ini, secara umum dibagi menjadi:

- a) Robot Manipulator , robot manipulator biasanya dicirikan dengan memiliki lengan (*arm robot*).
- b) Robot Mobil , robot yang mengarah ke robot yang bergerak, meskipun nantinya pada bagian robot tersebut juga dipasang manipulator. Pada robot mobil masih dibagi menjadi beberapa tipe lagi , yakni :
  - 1. Robot daratan (*ground robot*)
    - a. Robot beroda
    - b. Robot Berkaki
  - 2. Robot air (*submarine robot*)
  - 3. Robot terbang (*aerial robot*)

### **3. Karakteristik Robot**

Pada umumnya robot memiliki beberapa karakteristik, yaitu:

- a) *Sensing* : Robot harus dapat mendeteksi lingkungan sekitarnya (halangan, panas, suara, dan image)
- b) Mampu Bergerak : robot umumnya bergerak dengan menggunakan kaki atau roda, dan pada beberapa kasus robot diharapkan dapat terbang atau berenang.
- c) Cerdas : robot memiliki kecerdasan buatan agar dapat memutuskan aksi yang tepat dan akurat.

- d) Membutuhkan energi yang memadai : robot membutuhkan catu daya yang memadai agar unit pengontrol dan aktuator dapat menjalankan fungsinya dengan baik.

## **B. Motor Servo**

“Motor DC seringkali disebut juga sebagai motor servo walaupun dalam realitasnya berbeda dengan motor DC. Motor servo merupakan motor DC yang mempunyai kualitas tinggi, yang sudah dilengkapi dengan sistem kontrol yang ada di dalamnya. Dalam aplikasi motor servo sering digunakan sebagai kontrol loop tertutup untuk menangani perubahan posisi secara tepat dan akurat. Begitu juga dengan pengaturan kecepatan dan percepatan.”  
(Widodo Budiharto, 2010:73-74)

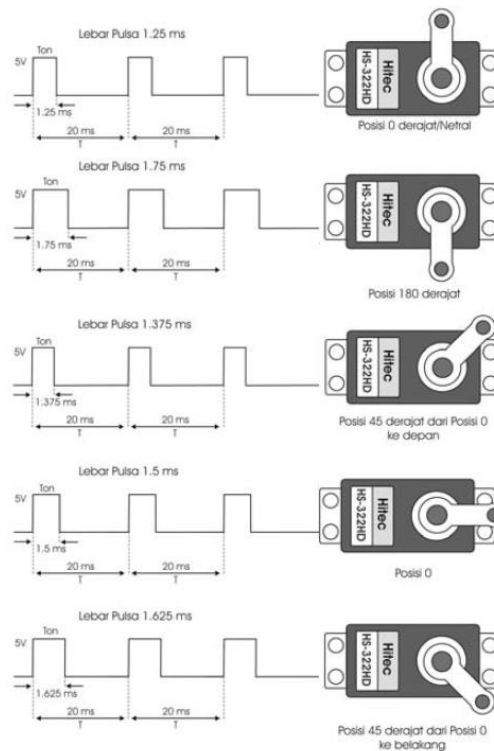
Sistem pengkabelan pada motor servo ada 3 bagian, yaitu Vcc, Ground dan Kontrol/Signal (PWM). Adapaun perbedaan antara penggunaan PWM motor servo dengan motor DC. Pada motor servo, pemberian nilai PWM akan membuat motor servo bergerak pada posisi tertentu dan kemudian akan berhenti (kontrol posisi). Pengaturan pada motor servo ialah menggunakan delay untuk setiap perpindahan dari posisi awal menuju posisi akhir.



Gambar 1. Motor Servo

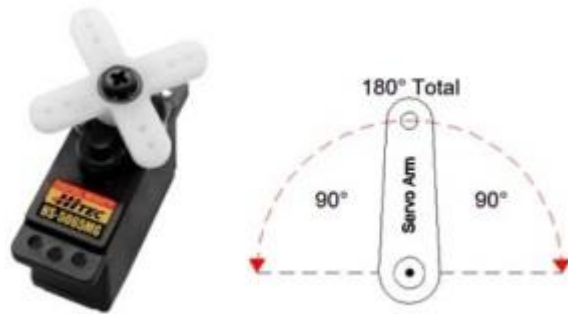
Sumber : <https://www.intorobotics.com/>

Motor ini terdiri dari motor DC, serangkaian gear, Potensiometer, dan serangkaian kontrol. Potensiometer disini berfungsi untuk menentukan batas sudut dari putaran servo, sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor servo. Tampak pada gambar 2 terlihat operasional motor servo dikendalikan pulsa sebesar  $\pm 20$  ms, dimana lebar pulsa antara 0.5 dan 2 ms menyatakan akhir dari range sudut maksimum. Apabila motor servo diberikan pulsa sebesar 1.5 ms mencapai gerakan  $90^{\circ}$ , maka bila kita berikan pulsa kurang dari 1.5 ms maka posisi mendekati  $0^{\circ}$  dan bila kita berikan pulsa lebih dari 1.5 ms maka posisi mendekati  $180^{\circ}$ .



Gambar 2. Pulsa Kendali Motor Servo  
 Sumber : <http://elektronika-dasar.web.id/motor-servo/>

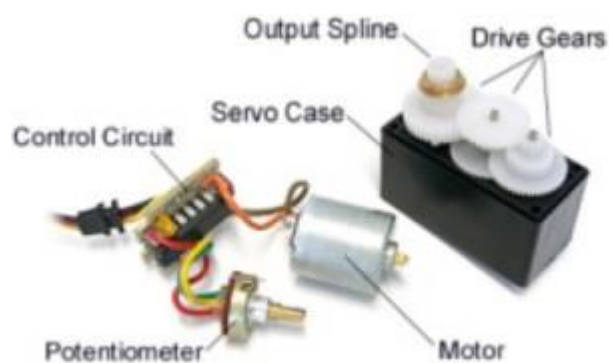
Motor servo dipilih karena motor servo memiliki tenaga atau torsi yang besar, sehingga dapat menggerakkan kaki robot dengan beban yang cukup berat. Pada umumnya motor servo yang dipakai untuk penggerak robot adalah motor servo 180<sup>0</sup>. Motor servo biasanya hanya bergerak mencapai sudut tertentu saja dan tidak kontinyu seperti motor DC maupun motor stepper. Walaupun demikian, untuk beberapa keperluan motor servo dapat dimodifikasi untuk bisa bergerak kontinyu. Pada robot, motor servo sering digunakan untuk bagian kaki, lengan atau bagian-bagian lain yang mempunyai gerakan terbatas dan membutuhkan torsi yang cukup besar.



Gambar 3. Motor servo 180<sup>0</sup>

Sumber : <http://zoniaelektro.net/motor-servo/>

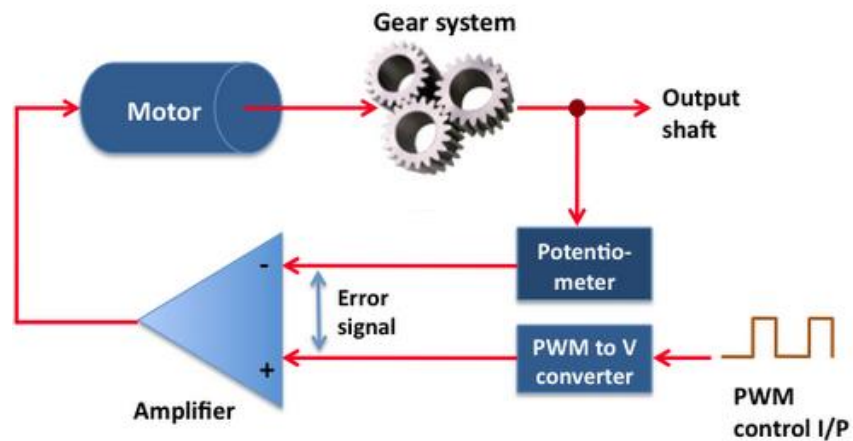
Motor servo adalah motor yang mampu bekerja dua arah (*CW /clock wise* dan *CCW counter clock wise*) dimana arah dan sudut pergerakan rotornya dapat dikendalikan hanya dengan memberikan pengaturan *duty cycle* sinyal PWM (*pulse width modulation*) pada bagian pin kontrolnya. Pada gambar.(diatas ini) motor servo standar 180<sup>0</sup>, motor servo jenis ini hanya mampu bergerak dua arah (*CW* dan *CCW*) dengan defleksi masing-masing sudut mencapai 90<sup>0</sup> sehingga total defleksi sudut dari kanan – tengah – kiri adalah 180<sup>0</sup>.



Gambar 4. Konstruksi Motor Servo 180<sup>0</sup>

Sumber : <http://zoniaelektro.net/motor-servo/>

Komponen diatas merupakan bagian-bagian motor servo dan masing – masing mempunyai fungsi sebagai controller, driver, sensor, girbox, dan aktuator. Motor pada sebuah motor servo adalah motor DC yang dikendalikan oleh bagian controller, kemudian komponen yang berfungsi sebagai sensor adalah potensiometer yang terhubung pada sistem girbox pada motor servo. Untuk mencapai fungsi servo, sebuah informasi sesaat dari poros output diumpamakan kembali ke rangkaian kontrol menggunakan *tranducer*. Sebuah cara sederhana untuk melakukan hal ini adalah dengan melampirkan potensiometer ke poros *output* atau di suatu tempat kereta gigi. Kontrol elektronik membandingkan sinyal umpan balik ( yang berisi posisi saat poros ) dari potensiometer untuk sinyal input kontrol (yang berisi informasi dari posisi yang diinginkan dari poros), dan perbedaan antara nilai aktual dan yang diinginkan (dikenal sebagai sinyal error) diperkuat dan digunakan untuk menggerakkan motor DC dalam arah yang diperlukan untuk mengurangi atau menghilangkan kesalahan. Kesalahan adalah nol ketika poros output sampai ke posisi yang diinginkan. Blok fungsi diagram dari motor servo khas ditunjukkan di bawah ini.



Gambar 5. Blok Fungsi Diagram Motor Servo

Sumber : [www.embedded-lab.com](http://www.embedded-lab.com)

### C. Baterai Lippo

Baterai LI-PO ( LITHIUM-POLIMER ) atau biasa disebut dengan LiPo merupakan salah satu jenis baterai yang sering digunakan dalam dunia RC. Pada tahun 1970-an baterai lithium-polimer mulai berkembang terus menerus. Adapaun desain pertama mereka termasuk elektrolit polimer yang bertekstur padat kering yang mirip film plastik. Lapisan film tersusun berlapis – lapis diantara anoda dan katoda yang mengakibatkan pertukaran ion. Dengan tersusunya seperti film plastik, baterai dibentuk seperti kartu kredit tipis namun tetap memiliki daya tahan baterai yang relatif baik. Disamping itu, baterai Lithium-Polymer sangat ringan dan memiliki tingkat keamanan yang lebih tinggi. Namun, produksi baterai lithium-polimer ini memakan biaya produksi yang cukup tinggi.

(anonim.(2015).*Baterai Li-ion dengan Li-Polimer*.Diakses tanggal 22-9-2016, dari <http://www.webkeren.net/2015/05/spesifikasi-dan-perbedaan-baterai-li-ion-dengan-li-polimer.html>)



Gambar 6. Baterai Lithium-Polymer  
 Sumber : [www.fpv-community.de/](http://www.fpv-community.de/)

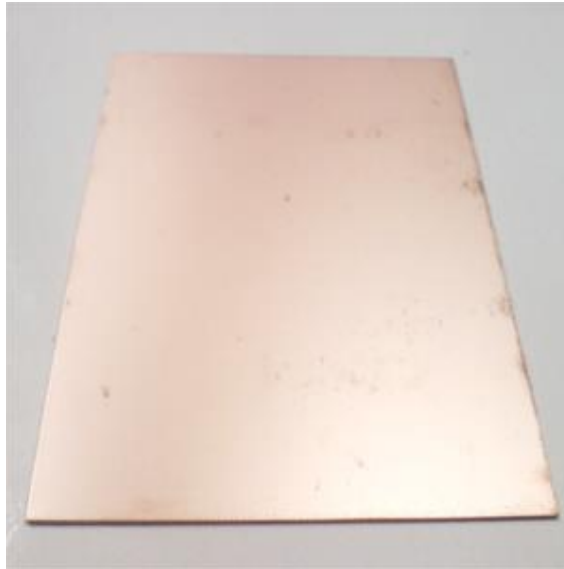
Adapun spesifikasi dari Baterai Lithium-Polymer, yakni:

Jenis	: Sekunder
Reaksi Kimia	: Bervariasi, tergantung pada elektrolit
Suhu Operasional	: Peningkatan kinerja pada suhu rendah dan tinggi
Direkomendasikan untuk	: Telepon seluler, perangkat komputasi mobile
Tegangan awal	: 3.6 & 7.2
Kapasitas	: Bervariasi tergantung pada baterai: unggul standar Lithium-Ion
Tingkat discharger	: Datar
Recharger Hidup	: 30 <sup>0</sup> - 40 <sup>0</sup> siklus
Pengisian Suhu	: 32 <sup>0</sup> F untuk 1400 F ( 0 <sup>0</sup> C sampai 60 <sup>0</sup> C )

Penyimpanan hidup	: Kehilangan kurang dari 0,1 % per bulan
Pembuangan	: Dapat didaur ulang
Catatan Lainnya	: - Biasanya dirancang untuk diisi ulang diperangkat bukan di charger eksternal. - Lebih ringan dari baterai sekunder berbasis nikel dengan ( Ni-Cd dan NiMH ). - Dapat dibuat dalam berbagai bentuk.

#### **D. Badan Robot**

Badan Robot menggunakan PCB (*Printed Circuit Board*) yang berbahan fiber, merupakan PCB dengan bahan yang baik dan bagus. Ini tergolong cukup mahal dibandingkan dengan PCB yang lainnya. Ciri dari PCB ini bagian bawah muka tembaga berwarna putih dan *flexibel*, PCB ini sangat bagus untuk membuat rangkaian dan sangat digemari para perakit rangkaian elektronika. Dalam pembuatan Badan Robot menggunakan PCB susah – susah gampang, karena harus memperhatikan desain Robot jangan sampai rusak dan PCB itu sendiri. Robot yang dibuat merupakan Robot Berkaki jadi untuk pemotongan pcb sangat harus hati – hati dan teliti.

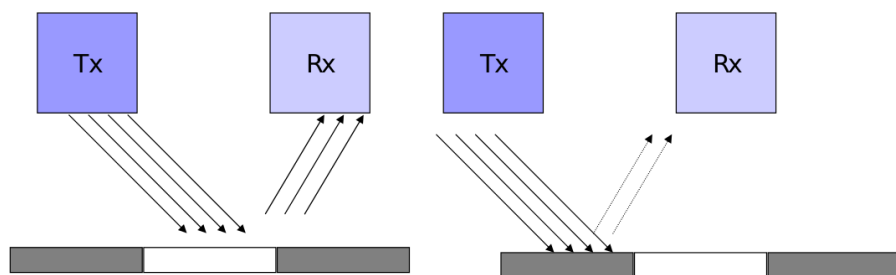


Gambar 7. PCB (*Printed Circuit Board*)

Sumber : <http://picclick.co.uk/>

#### E. Sensor Garis

Sensor garis atau *Proximity sensor* adalah sensor yang berfungsi mendeteksi warna gelap atau warna terang, dimana warna gelap atau terang tersebut terdeteksi akibat pantulan cahaya lampu (biasanya menggunakan lampu LED yang terdapat pada sensor. Lihatlah pada gambar



Gambar 8. Sensor Garis atau *Proximity Sensor*

Sumber : <https://pentriologist.wordpress.com>

Dari gambar diatas dapat kita simpulkan jika pantulan cahaya mengenai bagian yang berwarna gelap maka pantulan cahaya akan sedikit begitu juga sebaliknya jika pantulan cahaya mengenai bagian warna terang pantulan cahaya akan semakin banyak terdeteksi oleh sensor. Sensor garis tidak harus selalu menggunakan warna hitam dan putih untuk mendeteksi adanya suatu garis pada sensor, tapi warna lain pun bisa digunakan asalkan nilainya dapat dibedakan oleh sensor.

### 1. Photodiode

Photodiode adalah sebuah jenis diode yang peka terhadap cahaya, sensor photodiode akan mengalami perubahan resistansi pada saat menerima intensitas cahaya dan akan mengalirkan arus listrik secara forward sebagaimana diode pada umumnya. Sensor photodiode adalah suatu jenis phototransistor, photodiode akan mengalirkan arus yang membentuk fungsi linear terhadap intensitas cahaya yang diterima. Arus ini umumnya teratur terhadap *power density* (DP). Perbandingan antara arus keluaran dengan *power density* disebut sebagai *current responsivity*. Arus yang dimaksud adalah arus bocor ketika photodiode tersebut isinari dan dalam keadaan dipanjar mundur. Photodiode dalam gelap nilai tahanannya sangat besar hingga praktis tidak ada arus yang mengalir. Semakin kuat cahaya yang jatuh pada photodiode maka makin kecil nilai tahanannya, sehingga arus yang mengalir semakin besar. Tanggapan frekuensi sensor photodiode tidak luas. Dari rentang tanggapan itu, sensor photodiode memiliki tanggapan paling baik terhadap cahaya *infrared*, tepatnya pada cahaya dengan panjang

gelombang 0,9  $\mu\text{m}$ . Penggunaan sensor photodiode sebagai sensor robot sebagai pendeteksi cahaya yang dipancarkan *infrared* tentu baik. Hal ini tidak dapat dibuktikan dengan mata telanjang karena cahaya *infrared* merupakan cahaya tidak tampak, namun keberadaan cahaya *infrared* dapat dirasakan yaitu ketika ada rasa hangat, dan bisa melihat cahaya *infrared* menggunakan kamera.

(anonim.(2012, 1 september). *Sensor Photodiode*. Diakses tanggal 22 September 2016, dari <http://elektronika-dasar.web.id/sensor-photodiode/>)



Gambar 9. Sensor Photodiode

Sumber : <http://www.gh-led.com/photo-transistor.html>

## 2. Infra Merah (*infrared*)

Infra Merah (*infrared*) merupakan salah satu jenis LED (Light Emitting Diode) yang dapat memancarkan cahaya infra merah yang tidak kasat mata. Cahaya infra merah merupakan gelombang cahaya yang beberapa pada spectrum cahaya tak kasat mata. LED infra merah dapat memancarkan cahaya infra merah pada saat diode LED ini diberikan tegangan bias maju pada anoda dan katodanya. LED infra merah ini dapat memancarkan

gelombang cahaya infra merah karena dibuat dengan bahan khusus untuk memancarkan cahaya infra merah. “*Infra Red* LED memiliki panjang gelombang sebesar 750-1000nm dan arus maksimal 100mA” (Afrie Setiawan, 2010:12). Dilihat dari jangkauan yang cukup lebar, infra merah sangat fleksibel dalam penggunaannya. LED ini akan menyerap arus yang lebih besar dari pada dioda biasa. Semakin besar arus yang mengalir maka semakin besar daya pancarnya dan semakin jauh jarak sapuannya.



Gambar 10. Bentuk *Infrared led*  
Sumber : <http://www.gh-led.com/>

Cahaya infra merah tidak mudah terkontaminasi atau teresonansi dengan cahaya lain, sehingga dapat digunakan siang maupun malam. Aplikasi dari LED infra merah ini dapat digunakan sebagai *transmitter remote* maupun sebagai sensor pada robot. Cahaya infra merah sendiri dapat digunakan sebagai link pada jaringan telekomunikasi atau dapat juga

dipancarkan pada fiber optic. Sebagai receiver cahaya infra merah dapat digunakan foto dioda. Foto transistor maupun modul *receiver* infra merah.

## **F. Mikrokontroler**

Mikrokontroler adalah pengontrol rangkaian elektronik yang umumnya dapat menyimpan program didalamnya dan berbentuk sebuah chip. “Mikrokontroler juga merupakan dasar dari sebuah prinsip pengontrolan kerja robot, yang dimana orientasi dari penerapan mikrokontroler adalah untuk mengendalikan suatu sistem berdasarkan informasi input yang di terima, yang kemudian akan diproses oleh mikrokontroler, dan dilakukan sebuah aksi pada output sesuai program yang telah ditentukan sebelumnya. Mikrokontroler bisa dikatakan sebagai pengontrol utama perangkat elektronik pada saat ini, termasuk robot tentunya. Mikrokontroler yang mudah didapat dan sering digunakan di Indonesia saat ini ialah 89S51, AVR ATmega8535, ATmega16, ATmega32, dan ATmega 128, inilah beberapa mikrokontroler yang sering digunakan di Indonesia”.(Widodo Budiharto, 2010:77-78). Mikrokontroler terdiri dari CPU (*Central Processing Unit*), I/O tertentu dan unit pendukung seperti Analog to Digital Converter (ADC) yang sudah terintegrasi didalamnya. Secara harfiah bisa disebut pengendali kecil sebuah sistem elektronik yang sebelumnya banyak memerlukan komponen-komponen pendukung seperti IC TTL dan CMOS dapat direduksi atau diperkecil dan akhirnya terpusat serta dikendalikan oleh mikrokontroler.

Secara teknis hanya ada dua jenis mikrokontroler yaitu RISC dan CISC. Masing-masing mempunyai keturunan atau keluarga sendiri-sendiri. RISC kependekan dari *Reduced Instruction Set Computer* yang memiliki instruksi terbatas tapi memiliki fasilitas yang lebih banyak, CISC kependekan dari *Complex Instruction Set Computer* yang memiliki instruksi lebih lengkap tetapi dengan fasilitas secukupnya.

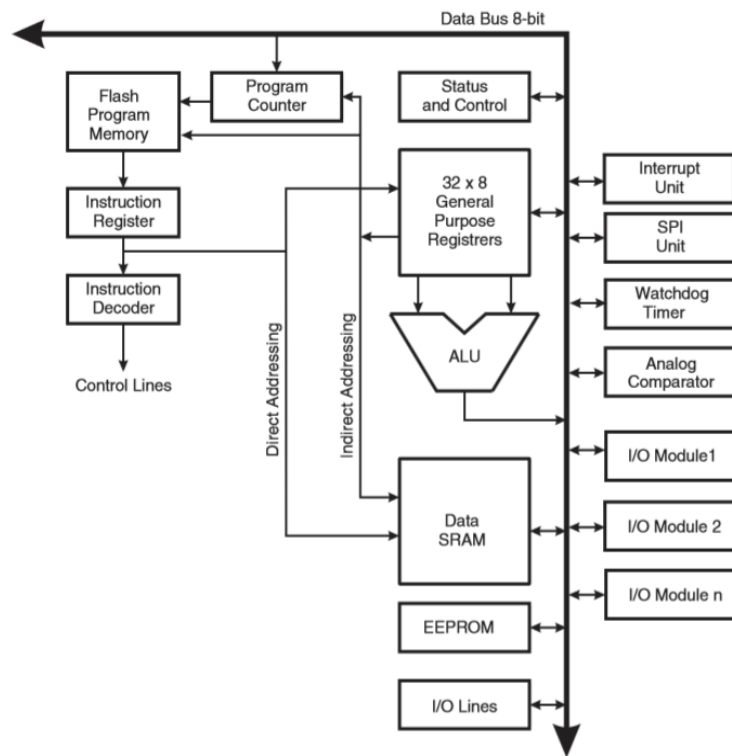
Mikrokontroler sudah mengandung beberapa periperal yang langsung bisa dimanfaatkan, misalnya *port serial*, komparator, konversi, digital ke analog ( DAC ), konversi analog ke digital dan sebagainya hanya menggunakan sistem minimum yang tidak rumit atau kompleks. Mikrokontroler menggunakan bahasa pemrograman *assembly* dengan berpatokan pada kaidah digital dasar sehingga pengoperasian sistem menjadi sangat mudah dikerjakan sesuai logika sistem ( bahasa *assembly* ini mudah dimengerti karena menggunakan bahasa *assembly* aplikasi dimana parameter *input* dan *output* langsung bisa diakses tanpa menggunakan banyak perintah). Penulisan bahasa *assembly* seperti huruf besar dan huruf kecil untuk bahasa *assembly* tetap diwajibkan. Mikrokontroler disusun atas satu chip dimana prosesor, memori, dan I/O terintegrasi menjadi satu kesatuan kontrol sistem sehingga mikrokontroler dapat dikatakan sebagai komputer mini yang dapat bekerja secara inovatif sesuai dengan kebutuhan sistem.

(anonim.(2012, 30 juni).*Pengertian dan kelebihan Mikrokontroler*.Diakses tanggal 24-9-2016, dari <http://elektronika-dasar.web.id/pengertian-dan-kelebihan-mikrokontroler/>)

## 1. Mikrokontroler ATmega 328

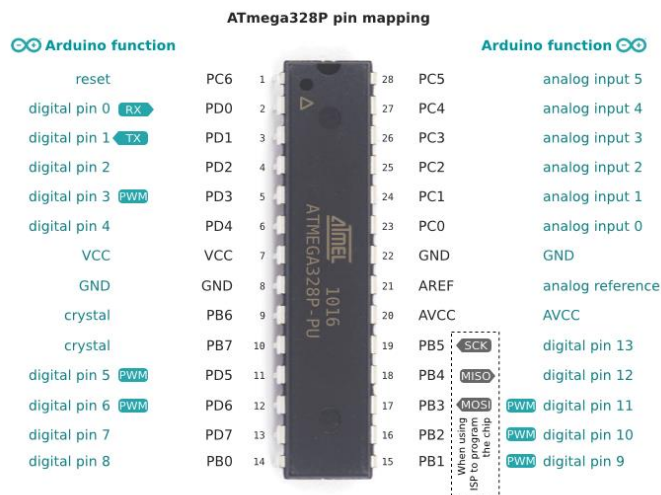
Mikrokontroler ATmega328 adalah mikrokontroler AVR (*Alf and Vegard's Risc Processor*) ATmega328 yang menggunakan *teknologi RISC (Reduce Instruction Set Computing)* dimana program berjalan lebih cepat karena hanya membutuhkan satu siklus clock untuk mengeksekusi satu instruksi program. Secara umum, mikrokontroler AVR dapat dikelompokkan menjadi 4 kelas, yaitu kelas Attiny, keluarga AT90Sxx, keluarga ATmega, dan AT86RFxx. Pada dasarnya yang membedakan masing-masing kelas adalah memori, *peripheral* dan fungsinya. Dari segi arsitektur dan instruksi yang digunakan, mereka bisa dikatakan hampir sama.

Mikrokontroler AVR ATmega328 memiliki fitur yang cukup lengkap. Mikrokontroler AVR ATmega328 telah dilengkapi dengan ADC *internal*, EEPROM *internal*, *Timer/Counter*; *PWM*, *analog comparator*, dan lain-lain (M.Ary Heryanto, 2008). Sehingga dengan fasilitas yang lengkap ini memungkinkan kita belajar mikrokontroler keluarga AVR dengan lebih mudah dan efisien, serta dapat mengembangkan kreatifitas penggunaan mikrokontroler ATmega328. Blok diagram ATmega328 dapat dilihat lebih jelas pada gambar



Gambar 11. Blok Diagram ATmega 328  
(Datasheet ATmega 328 )

## 2. Konfigurasi Pin ATmega 328



Gambar 12. Konfigurasi Pin ATmega328

Sumber : [//hallard.me/blog/wp-content/uploads/2012/10/Arduino-ports.png](http://hallard.me/blog/wp-content/uploads/2012/10/Arduino-ports.png)

ATmega 328 memiliki 28 pin DIP (*Dual Inline Package*) yang masing-masing pin-nya memiliki fungsi yang berbeda-beda baik sebagai port ataupun sebagai fungsi yang lain. Berikut akan dijelaskan tentang kegunaan dari masing-masing kaki pada ATmega 328.

a) VCC

Merupakan *pin* yang berfungsi sebagai masukan (*supply*) catu daya.

b) GND

Merupakan *ground* untuk semua komponen yang membutuhkan *grounding*.

c) AVCC

Merupakan pin tegangan catu untuk *A/D converter*.

d) AREF

Merupakan pin tegangan referensi analog untuk ADC

e) Port B (PortB0...PortB7)

Didalam Port B terdapat XTAL1, XTAL2, TOSC1, TOSC2, jumlah Port B adalah 8 buah pin mulai dari pin B.0 sampai dengan pin B.7. tiap pin dapat digunakan sebagai *input* dan juga *output*. Port B merupakan sebuah I/O port dengan internal pull-up resistor. Sebagai *input*, pin-pin yang terdapat pada port B yang secara eksternal diturunkan, maka akan mengeluarkan arus jika pull-up resistor diakifkan. Jika ingin menggunakan tambahan kristal, maka cukup menghubungkan kaki dari kristal ke kaki pada pin port B. Namun jika tidak digunakan maka cukup dibiarkan saja.

Penggunaan kegunaan dari masing-masing kaki ditentukan dari clock fuse setting-nya.

f) Port C (PortC0...PortC6)

Port C merupakan sebuah 7-bit bit-directional I/O port yang didalam masing-masing pin terdapat pull-up resistor. Jumlah pin-nya hanya 7 buah mulai dari pin C.0 sampai dengan pin C.6. sebagai keluaran / *output*, port C memiliki karakteristik yang sama dalam hal kemampuan menyerap arus (*sink*) ataupun mengeluarkan arus (*source*).

g) PortD (PortD0...PortD4)

Port D merupakan sebuah 5-bit bit-directional I/O yang didalam masing-masing pin terdapat pull-up resistor. Jumlah pinya hanya 5 buah mulai dari pin D.0 sampai dengan D.4. sebagai keluaran / *output* port D memiliki karakteristik yang sama dalam kemampuan menyerap arus (*sink*) ataupun mengeluarkan arus (*source*).

h) Reset / PC 6

Jika RSTDSBL Fuse diprogram maka PC 6 akan berfungsi sebagai pin I/O. Untuk diperhatikan juga bahwa pin ini memiliki karakteristiik yang berbeda dengan pin-pin yang terdapat pada port C. Namun jika RSTDISBL tidak diprogram maka pin ini akan berfungsi sebagai *input reset*. Dan jika level tegangan yang masuk ke pin ini rendah maka pulsa yang ada lebih pendek dari pulsa

minimum, maka akan menghasilkan suatu kondisi reset meskipun clock-nya tidak bekerja.

### 3. Sistem Minimum

Untuk memrogram sebuah mikrokontroler membutuhkan sistem minimum. Rangkaian sistem minimum Mikrokontroler adalah rangkaian elektronika yang terdiri dari komponen dasar yang dibutuhkan oleh suatu uC untuk dapat berfungsi dengan baik. Sistem minimum atau Sismin ini kemudian bisa dihubungkan dengan rangkaian lain untuk menjalankan fungsi tertentu. Selain power supply, pada umumnya suatu mikrokontroler membutuhkan dua elemen untuk berfungsi yaitu kristal osilator dan rangkaian RESET. Analogi fungsi kristal osilator adalah jantung pada tubuh manusia. Perbedaannya jantung memompa darah sedangkan XTAL memompa data. Fungsi rangkaian RESET adalah untuk membuat uC memulai kembali pembacaan program. Ini dibutuhkan saat *device hang* saat eksekusi program. Rangkaian sistem minimum mikrokontroler diatas menggunakan ATmega 328. Komponen yang dibutuhkan untuk membuat sistem minimum mikrokontroler ini ialah :

- a. Mikrokontroler ATmega 328.
- b. Power Supply
- c. Kristal
- d. ISP

e. Push Button

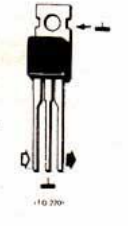
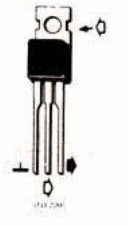
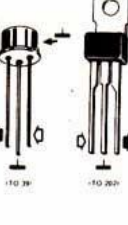
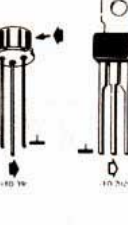
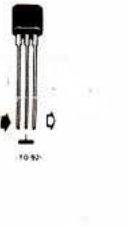
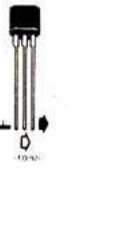
f. Resistor

g. Kapasitor

h. LED

## **G. Catu Daya**

Catu daya adalah suatu sistem filter penyearah (*rectifier-filter*) yang mengubah tegangan AC menjadi tegangan DC murni, atau sebuah peralatan penyedia tegangan sumber daya untuk peralatan elektronika dengan prinsip mengubah prinsip mengubah tegangan listrik yang tersedia dari jaringan distribusi transmisi listrik ke level yang diinginkan sehingga berimplikasi pada perubahan daya listrik. Dalam sistem perubahan daya, terdapat empat jenis proses yang telah dikenal yaitu sistem perubahan daya AC ke DC, DC ke DC, DC ke AC, dan AC ke AC. Masing-masing sistem perubahan memiliki keunikan aplikasi tersendiri, tetapi ada dua yang implementasinya kemudian berkembang pesat dan luas yaitu sistem perubahan AC ke DC (*DC power supply*) dan DC ke DC (*DC-DC converter*).

	<p>7805 7806 7808 7812 7815 7818 7824</p> <p><math>I_{out} = 1\text{ A}</math></p>		<p>7905 7906 7908 7912 7915 7918 7924</p> <p><math>I_{out} = -1\text{ A}</math></p>
	<p>78M05 78M06 78M08 78M12 78M15 78M18 78M24</p> <p><math>I_{out} = 500\text{ mA}</math></p>		<p>79M05 79M06 79M08 79M12 79M15 79M18 79M24</p> <p><math>I_{out} = -500\text{ mA}</math></p>
	<p>78L05 78L06 78L08 78L12 78L15 78L18 78L24</p> <p><math>I_{out} = 100\text{ mA}</math></p>		<p>79L05 79L06 79L08 79L12 79L15 79L18 79L24</p> <p><math>I_{out} = -100\text{ mA}</math></p>

Gambar 13. Contoh IC Regulator  
 Sumber : <http://www.homemade-circuits.com/>

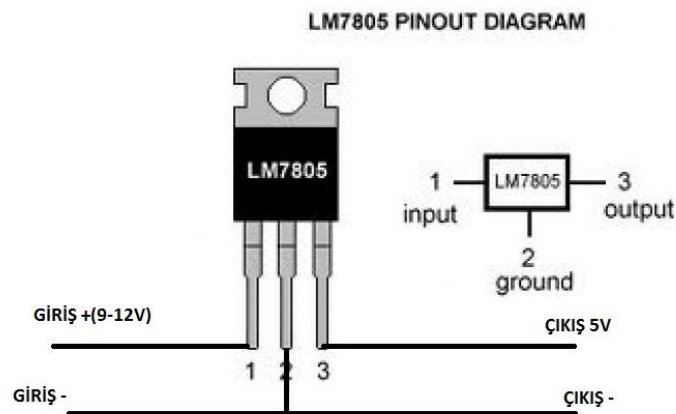
IC jenis pengatur tegangan tetap (*Fixed Voltage Regulator*) ini memiliki nilai yang tetap yang tidak dapat disetel (*di-adjust*) sesuai dengan keinginan rangkaiannya. Sebagai contoh IC Regulator yang tertera pada nomor IC-nya 7805, 7806, 7809, 7812, 7815, akan mengeluarkan tegangan keluaran 5, 6, 9, 12, 15 volt.

Tabel 1. Karakteristik Regulator Tegangan Positif 78xx

Tipe	V Out (V)	I Out (A)			V in (V)	
		78xxC	78Lxx	78Mxx	Min	Max
7805	5	1	0,1	0,5	7,5	20
7806	6	1	0,1	0,5	8,6	21
7808	8	1	0,1	0,5	10,6	23
7809	9	1	0,1	0,5	11,7	24
7810	10	1	0,1	0,5	12,7	25
7812	12	1	0,1	0,5	14,8	27
7815	15	1	0,1	0,5	18	30
7818	18	1	0,1	0,5	21	33
7824	24	1	0,1	0,5	27,3	38

Sumber : [www.elektronika-dasar.web.id](http://www.elektronika-dasar.web.id)

Pada bagan diatas angka xx penulisan tipe regulator 78xx merupakan besar tegangan output regulator. Huruf L, M merupakan besar arus maksimum yang dapat mengalir pada terminal output regulator positif. Dalam proyek akhir ini besarnya catu daya yang dibutuhkan sebesar + 5 volt DC dengan arus maksimal 1 Ampere sehingga menggunakan IC LM 7805.



Gambar 14. IC LM 7805

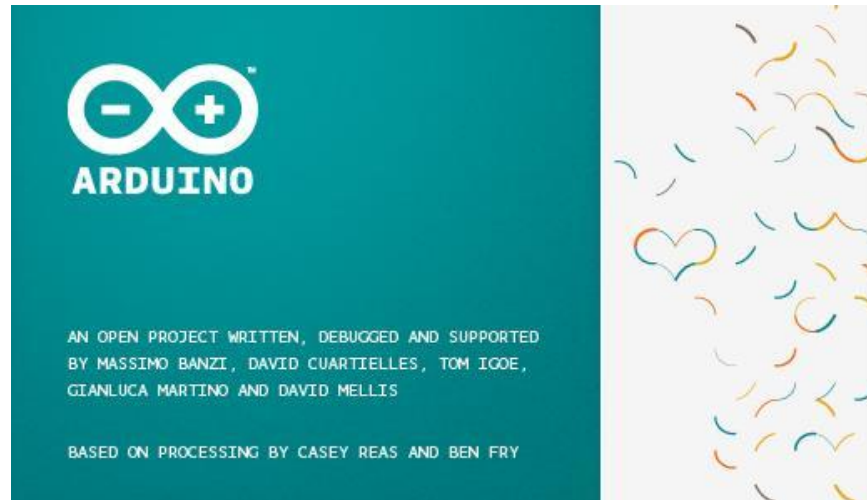
Sumber : <http://elektrikforum.net>

IC LM 7805 (*regulator*) adalah untuk menstabilkan tegangan dari catu daya bila terjadi perubahan tegangan. Adapun keuntungan memakai IC LM 7805 :

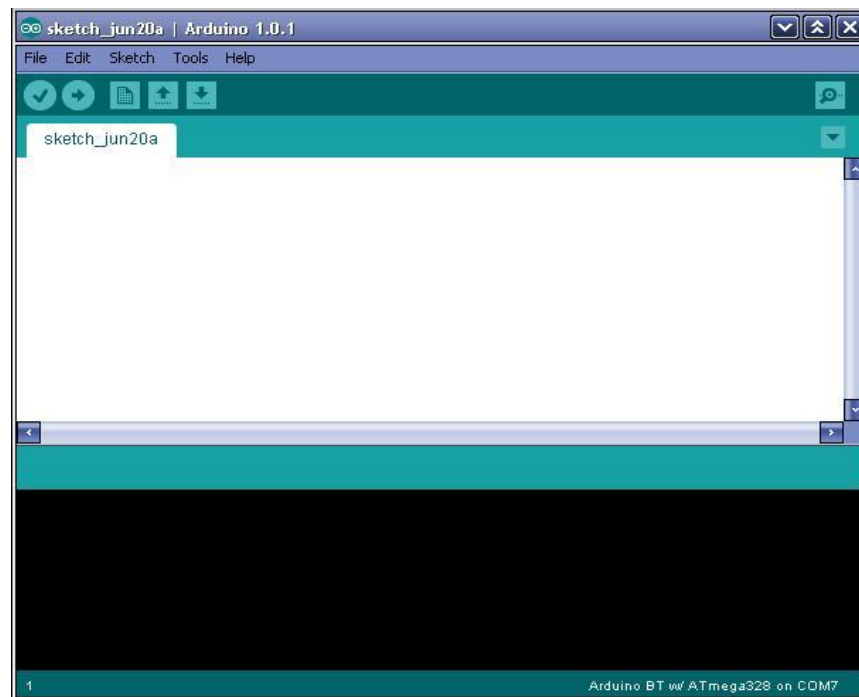
- a. Membutuhkan penambahan komponen luar yang sedikit.
- b. Mempunyai proteksi terhadap arus hubungan siangkat.
- c. Mempunyai tegangan output yang konstan.
- d. Mempunyai arus rendah.
- e. Memiliki *ripple output* yang sangat kecil.
- f. Pembiayaan rendah.

#### **H. Arduino IDE**

Arduino adalah perangkat lunak IDE (*Integrated Development Environment*). Sebuah perangkat lunak yang memudahkan kita mengembangkan aplikasi mikrokontroler mulai dari menuliskan source program, kompilasi, upload hasil kompilasi, dan uji coba secara terminal serial. Arduino ini bisa dijalankan di komputer dengan berbagai macam platform karena didukung atau berbasis Java. *Source* program yang kita buat untuk aplikasi mikrokontroler adalah bahasa C/C++ dan dapat digabungkan dengan *assembly*.



Gambar 15. Logo Arduino



Gambar 16. Arduino IDE

Pada tampilan Arduino IDE terdapat tiga jendela yaitu menu, tombol icon, editor dan pesan. Pada bagian bawah terlihat jenis mikrokontroler atau

board arduino saat ini yaitu Board Arduino BT dengan mikrokontroler 328 dengan menggunakan kanal serial COM7 untuk upload hasil kompilasi dan komunikasi konsol serial. Arduino sangat kaya dengan library karena arduino sifatnya adalah *opensource*. Selain arduino IDE sebagai jantungnya, *bootloader* adalah jantung arduino lainnya yang berupa program kecil yang dieksekusi sesaat setelah mikrokontroler diberi catu daya. *Bootloader* ini berfungsi sebagai pemonitor aktifitas yang diinginkan oleh arduino. Jika dalam IDE terdapat file hasil kompilasi yang akan *diupload*, *bootloader* secara otomatis menyambutnya untuk disimpan dalam memori program. Jika pada saat awal mikrokontroler bekerja, *bootloader* akan mengeksekusi program aplikasi yang telah *diupload* sebelumnya. Jika IDE hendak mengupload program baru, *bootloader* seketika akan menghentikan eksekusi program berganti menerima data program untuk selanjutnya diprogramkan dalam memori program mikrokontroler.

## **BAB III**

### **PERANCANGAN DAN PEMBUATAN ALAT**

Prototipe Robot *Line Follower* Berkaki Sebagai Pengantar Makanan Di Restoran Berbasis Mikrokontroler ATmega328 menggunakan metode rancang bangun. Secara urut metode tersebut adalah identifikasi kebutuhan yang diperlukan kemudian kebutuhan tersebut untuk mendapatkan komponen yang lebih spesifik. Selanjutnya dilakukan perancangan perangkat keras dan perangkat lunak pembuatan serta pengujian.

#### **A. Identifikasi Kebutuhan**

Prototipe Robot *Line Follower* Berkaki Berbasis Mikrokontroler ATmega328 memerlukan adanya identifikasi masalah, adapun identifikasi kebutuhan dari penulis yaitu Prototipe Robot *Line Follower* Berkaki dapat menaruh nampan makanan di meja pelanggan dengan perintah inputan berupa *Push Button* dan menuju ke meja pelanggan mengikuti garis sesuai meja yang dipilih. Prototipe Robot *Line Follower* Berkaki Sebagai Pengantar Makanan Di Restoran Berbasis Mikrokontroler ATmega328 bisa terwujud memerlukan adanya identifikasi kebutuhan terhadap rancang bangun yang akan dibuat, antara lain :

1. Baterai untuk menyuplai daya ke alat.
2. Motor servo sebagai penggerak atau motor yang dikontrol
3. Sensor photodiode sebagai petunjuk arah kemana robot akan berjalan
4. Sistem minimum ATmega328 sebagai prosesor utama
5. Sistem minimum ATmega328 sebagai kontroler servo
6. Sistem minimum ATmega328 sebagai kontroler sensor

7. Tombol Navigasi sebagai perintah robot ke meja yang ditentukan.

## **B. Analisis Kebutuhan**

Berasarkan Identifikasi Kebutuhan yang ada, maka diperlukan beberapa spesifikasi dari komponen atau rangkaian sebagai berikut:

### 1. Baterai (*power supply*)

Baterai dibutuhkan untuk menyuplai daya ke alat. Tegangan baterai yang dibutuhkan 5 Volt sebagai sumber daya pada Robot Line Follower Berkaki.

### 2. Motor Servo

Pada bagian ini digunakan sebagai motor yang dikontrol. Sesuai dengan fungsinya motor servo digunakan sebagai penggerak lengan Pendorong nampun dan kaki robot agar robot bisa berjalan sebagaimana mestinya.

### 3. Sensor Photodiode

Digunakan sebagai penunjuk arah kemana robot akan bergerak / kemeja pelanggan yang sudah ditentukan.

### 4. Sistem Minimum ATmega328

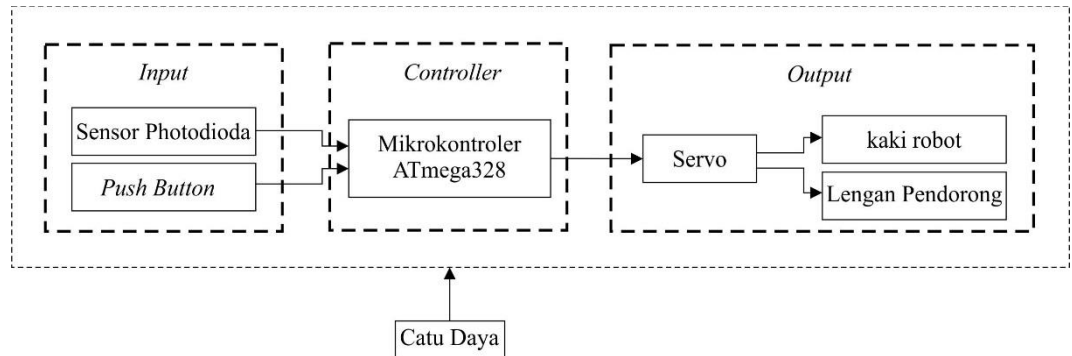
Alat ini menggunakan Mikrokontroler ATmega328. Digunakan sebagai pengendali utama mengenai proses sistem, pengendalian motor servo dan pengendalian sensor.

### 5. Tombol Navigasi

Sebagai Inputan perintah langsung robot untuk mengantar makanan ke meja pelanggan sesuai nomor tombol navigasi yang ditekan.

### C. Perancangan Alat

Rangkaian Prototipe Robot *Line Follower* Berkaki ini terdiri atas beberapa blok, diantaranya sebagai berikut:



Gambar 17. Blok Diagram Prototipe Robot *Line Follower* Berkaki

Pada gambar No.18 dapat dilihat alur proses yang terjadi pada Prototipe Robot *Line Follower* Berkaki. Berikut ini merupakan penjelasan dari masing–masing blok:

Catu Daya (Baterai) merupakan penyuplai tegangan yang digunakan untuk menghidupkan rangkaian sistem mikrokontroler, motor servo dan sensor photodiode.

Blok *Input* terdiri dari Sensor photodiode dan *push button*. Sensor photodiode merupakan sensor pembaca garis, fungsi di alat ini untuk menunjukkan arah gerak Prototipe Robot *Line Follower* Berkaki ke meja yang diinginkan. Push button (1,2,3,4) merupakan inputan untuk memutuskan arah gerak Prototipe Robot *Line Follower* Berkaki ke meja yang ditentukan.

Blok *Controller* menggunakan *Mikrokontroler ATmega328* yang digunakan untuk mengontrol motor servo.

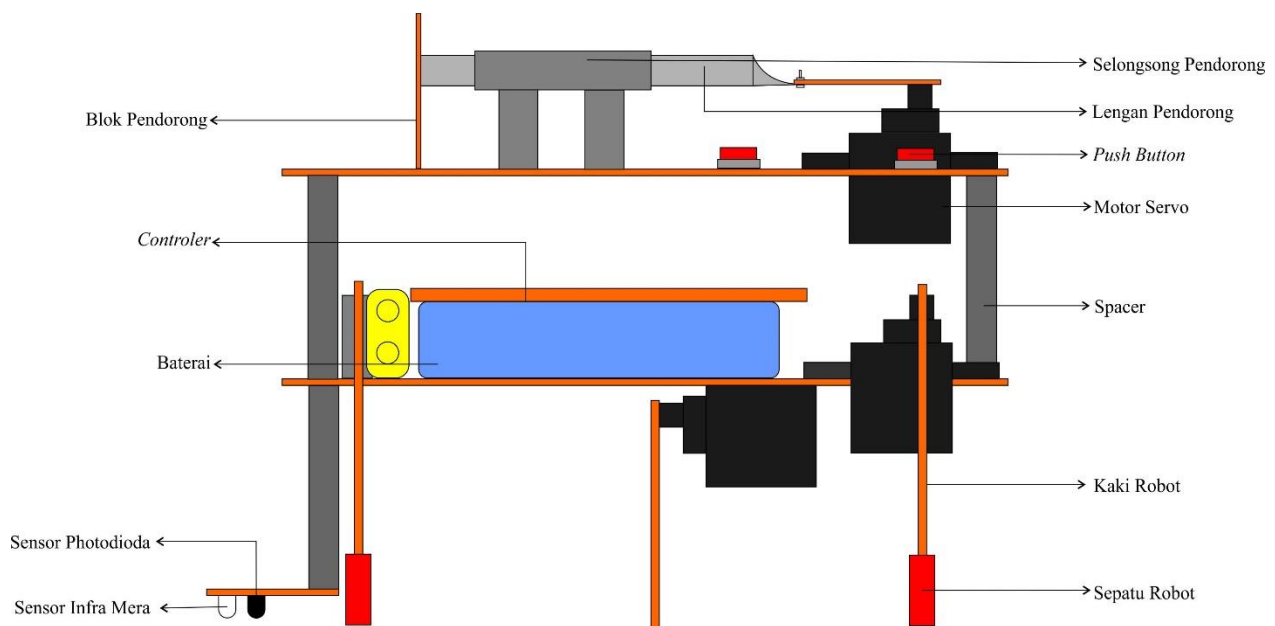
Blok *Output* merupakan motor servo yang digunakan sebagai kaki penggerak robot dan lengan pendorong nampan.

## 1. Perancangan Mekanik Robot

Mekanik robot terdiri dari badan mekanik robot, lengan robot dan kaki robot. Badan mekanik robot digunakan untuk meletakkan baterai, rangkaian elektronik robot, sensor, dan *push button*. Lengan robot digunakan untuk mendorong miniatur nampan makanan. Kaki robot digunakan untuk maju mundurnya robot.

Perancangan mekanik badan robot dapat dilihat pada gambar 11.

### a. Rancangan Mekanik Robot Tampak Samping



Gambar 18. Rancangan Mekanik Robot Tampak Samping

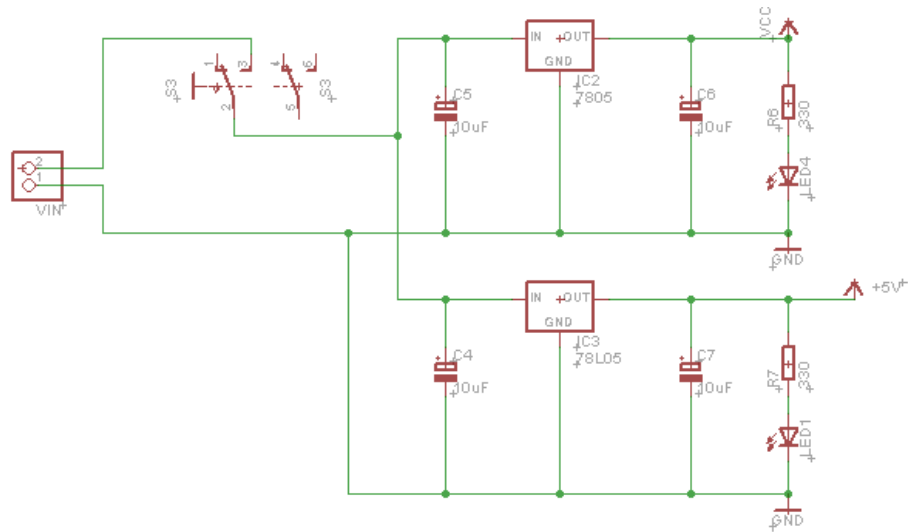
Pada Gambar 19. memperlihatkan mekanik badan robot yang terpasang komponen-komponen robot. Robot diatas merupakan penglihatan robot tampak samping untuk bisa melihat semua

komponen yang terpasang. Motor servo bagian paling atas berfungsi untuk menggerakkan lengan pendorong dan blok pendorong agar mampu mendorong nampan ke meja. Robot menggunakan tiga buah servo sebagai penggerak kaki-kaki robot untuk bisa berjalan menuju meja. Rangkaian sistem minimum, baterai dan tiga buah servo dipasang di lantai 1 pada badan mekanik robot. Sensor infra merah di pasang pada bagian paling bawah badan mekanik badan robot agar mampu menjangkau atau membaca garis sensor yang terpasang dilantai miniatur restoran. Tombol power diletakan pada sistem minimum pada bagian belakang dan mudah dijangkau dalam menyalakan atau mematikan sumber daya.

#### **D. Perencanaan Rangkaian**

##### **1. Catu daya**

Rangkaian catu daya sangat penting karena tanpa catu daya alat ini tidak dapat bekerja. Rangkaian catu daya yang dibutuhkan 5 Volt dengan pemilihan IC regulator 7805 digunakan untuk menyuplai tenaga ke beberapa rangkaian seperti sistem minimum ATmega 328, sensor *infra red* dan motor servo.



Gambar 19. Rangkaian catu daya

Pada Gambar 20. Merupakan rangkaian catu daya yang digunakan untuk menyuplai daya ke seluruh sistem kelistrikan pada robot. Dan daya yang dibutuhkan untuk sistem minimum ATmega328 adalah 5 V.

a. Baterai Lippo

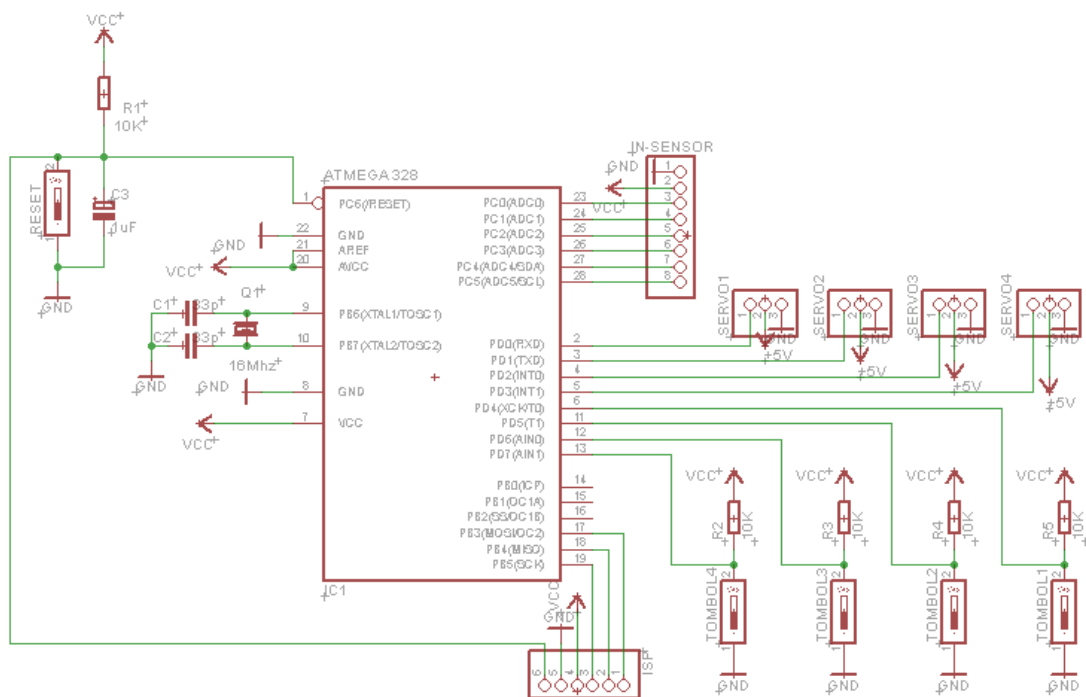
Baterai Lippo merupakan salah satu sumber tegangan yang paling baik, karena dapat menyuplai tegangan DC ( direct current ) secara continue. Karena output baterai sudah berbentuk tegangan DC maka tidak diperlukan rangkaian penyearah dan rangkaian filter.



Gambar 20. Baterai Lippo

## 2. Rangkaian Sistem Minimum Mikrokontroler ATmega328

Kebutuhan Prototipe Robot *Line Follower* Berkaki membutuhkan beberapa sensor dan perangkat pendukung kecerdasan robot. Berikut ini merupakan kebutuhan port yang harus disediakan:



Gambar 21. Rangkaian Skematik Mikrokontroler ATmega328

Rangkaian sistem minimum ini digunakan sebagai pusat kendali dari keseluruhan sistem kaki robot dan sistem lengan pendorong dari robot. Salah satu yang dikendalikan adalah IC servo kontroler dengan menggunakan port D yang terdapat pada kaki IC mikrokontroler ATmega 328.

## 3. Rangkaian Servo Kontroler

Rangkaian servo kontroler merupakan rangkaian yang terbuat dari mikrokontroler ATmega 328 standart yang biasa diprogram sebagai IC

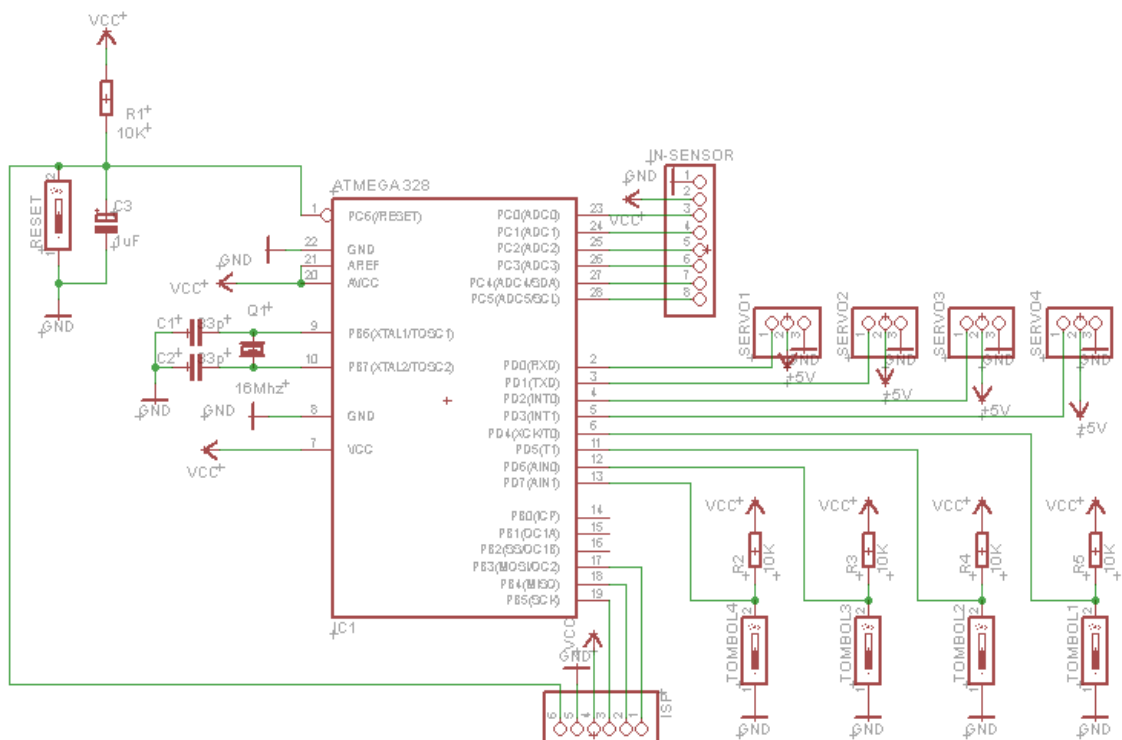
utama. Pada rangkaian ini mikrokontroler ATmega 328 digunakan sebagai pengendali motor servo dengan sensor utama. Sesuai dengan fungsinya rangkaian ini digunakan sebagai pengatur jalan robot dengan mengendalikan motor servo. Berikut ini merupakan masing – masing fungsi port I/O:

a. Port C

Port C digunakan untuk mengendalikan sensor utama

b. Port D

Port D digunakan untuk mengendalikan motor servo dan juga sebagai input dari Tombol perintah.



Gambar 22. Rangkaian Servo Kontrol

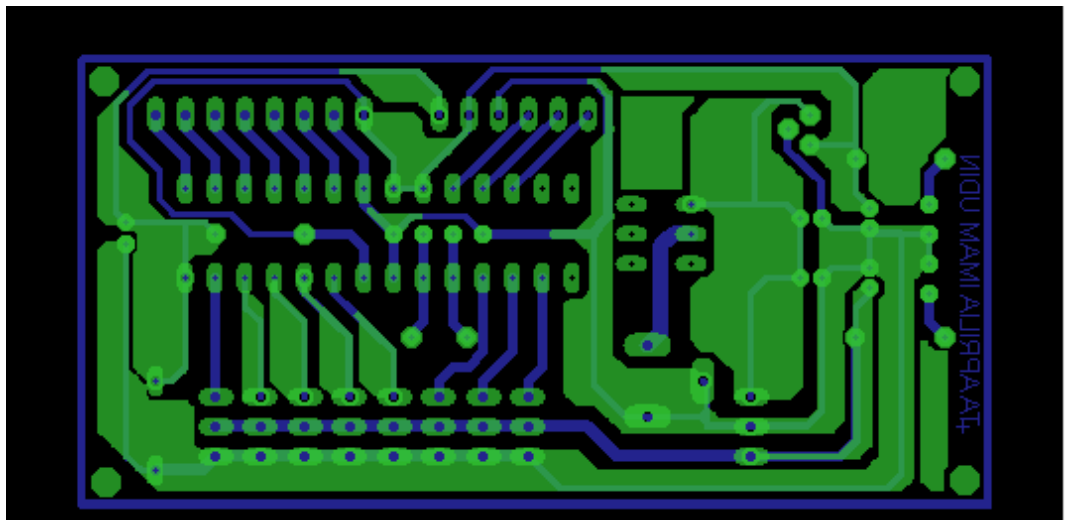
## E. Langkah Pembuatan Alat

Langkah pembuatan alat pada proyek akhir ini terdiri dari pembuatan PCB, pemasangan komponen, pelarutan, dan pengeboran PCB.

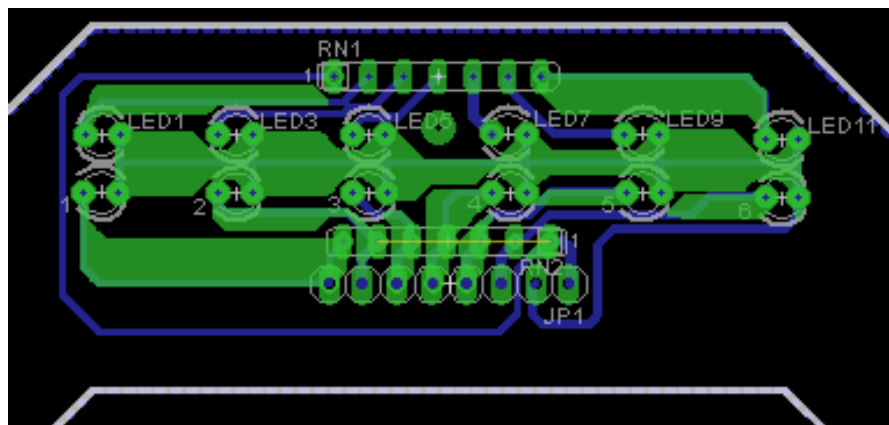
### 1. Pembuatan PCB

#### a. Pembuatan Desain PCB

Langkah awal dari pembuatan PCB adalah menggambar rangkaian dan layout dengan perangkat lunak yang tersedia, penulis disini menggunakan perangkat lunak Eagle.



Gambar 23. Layout Sistem Minimum ATmega328



Gambar 24. Layout Sensor

b. Penyablonan PCB

Desain PCB selesai dibuat, maka langkah selanjutnya adalah penyablonan desain ke PCB polos. Langkah pertama dengan mencetak desain PCB pada kertas glossy secara mirror.

Langkah kedua menempelkan desain yang terdapat pada kertas glossy ke PCB yang polos. Supaya desain dapat tersalin ke PCB maka kertas tersebut disetrika diatas PCB sampai melekat kurang lebih 8 menit.

Setelah kertas melekat pada PCB selanjutnya merendam PCB beserta kertas yang menempel. Setelah beberapa menit hilangkan kertas secara perlahan sampai yang tersisa hanya jalur rangkaian saja.

c. Pelarutan dan Pengeboran PCB

Langkah selanjutnya adalah pelarutan PCB dengan cairan kimia *feri chloride* dan air hangat hingga jalur rangkaian terbentuk. Setelah jalur terbentuk sesuai desain, mengangkat PCB dari cairan *feri chloride* dan membersihkan dengan air sampai bersih dan jalur yang terlihat berbentuk tembaga.

## 2. Pemasangan Komponen

Pada bagian pemasangan komponen dapat dilaksanakan dengan urutan berikut:

a. Menyiapkan komponen yang akan dipasang sesuai dengan desain.

- b. Memasang jumper dan komponen yang kecil dahulu baru mulai yang besar.
- c. Menyolder kaki komponen sampai semua komponen terpasang.

Sebelum menguji rangkaian dengan memberi tegangan terlebih dahulu di cek jalur dan soldirannya agar tidak terjadi konsleting saat pengujian dan penggunaan.

### **3. Pembuatan Kerangka Robot**

Pembuatan kerangka menggunakan bahan dasar PCB yang pemotongan pembentukan bisa menggunakan gergaji besi ataupun gerindra. Pembuatan kerangka melalui beberapa tahapan sebagai berikut:

- a. Desain kerangkai dengan perangkat lunak yang bisa untuk menggambar secara akurat yaitu Corel Draw.
- b. Pemotongan dan pembentukan pada PCB.
- c. Pengeleman dan dibaut bodi robot yang sudah dipotong
- d. Penyatuan kerangka perbagian dengan motor servo dan rangkaian.

Desain kerangka dapat dilihat pada lampiran akhir laporan.

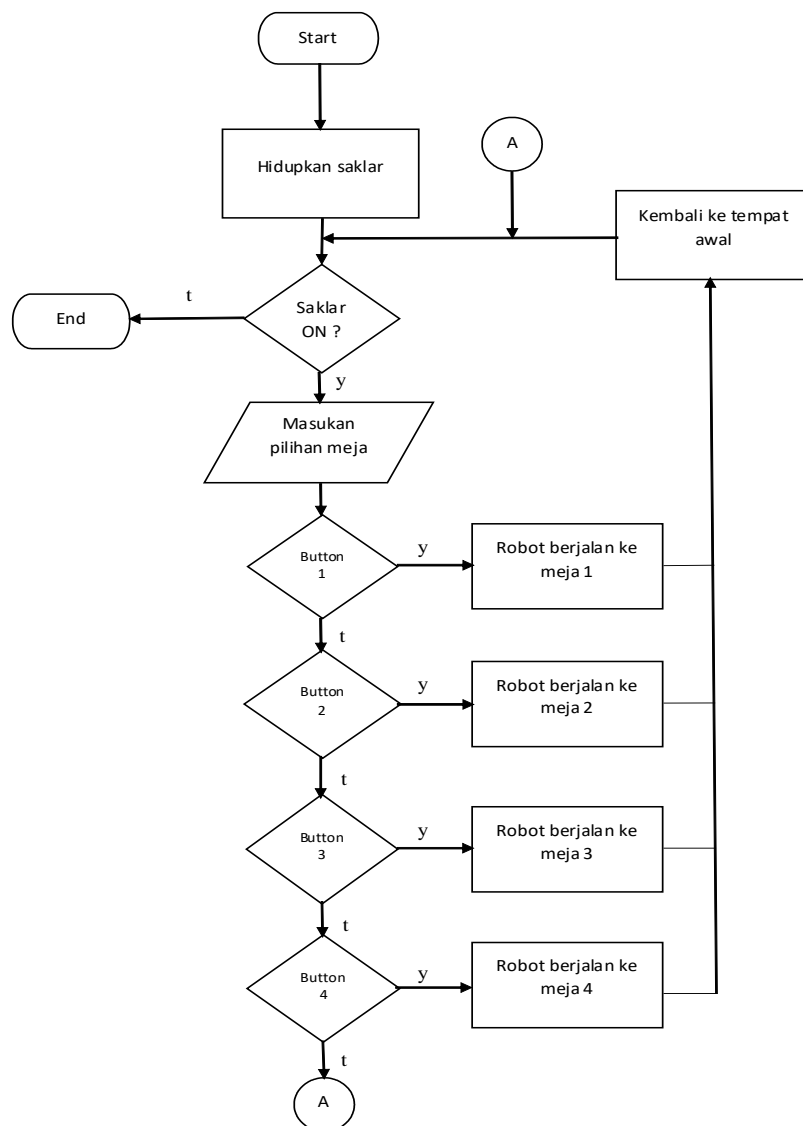
## **F. Perancangan Perangkat Lunak**

### **1. Program**

Sebelum memasukan program yang akan diisikan ke-chip, terlebih dahulu menuliskan programnya. Pemrograman bahasa C dalam laporan ini menggunakan software yang sudah ada yaitu Arduino walaupun dapat menggunakan software yang lain. Software ini maka akan mudah untuk pembuatannya karena menggunakan compiler sebagai penguji

program benar atau masih error. Cara memulai pembuatan program terlebih dahulu jalankan software-nya. jika sudah berjalan akan langsung muncul jendela program Arduino yang sudah langsung bisa digunakan. Untuk beberapa model/tipe software arduino sudah terdapat program bawaan, jadi kita tinggal menambahkan program yang sesuai dengan kebutuhan.

## 2. Flowchart



Gambar 25. Flowchart Cara kerja Robot *Line Follower* Berkaki

Flowchart diatas menjelaskan tentang alur program yang terdapat pada alat yang dibuat yaitu servo kontroler menggunakan ATmega328.

Berikut ini merupakan penjelasan gambar *Flowchart* :

- a. START, menjelaskan mulainya program
- b. Saklar on ?, menjelaskan kondisi robot dalam posisi hidup atau mati
- c. Masukan pilihan meja, menjelaskan bahwa perintah untuk memilih button yang akan ditekan sebagai perintah.
- d. Button 1,2,3,4, menjelaskan alamat yang dituju sesuai dengan inputan push button. Jika alamat sesuai maka akan diteruskan ke proses selanjutnya, tetapi jika tidak akan mencari alamat yang sesuai.
- e. Robot bergerak ke meja 1,2,3,4, menjelaskan sesuai dengan perintah yang diterima dari input 1,2,3,4, dengan kecepatan yang telah ditentukan.
- f. Kembali ke tempat awal, menjelaskan robot kembali ketempat setelah program perintah selesai
- g. End, menjelaskan sebagai akhir/berhentinya program

#### **G. Spesifikasi Alat**

Prototipe Robot *Line Follower* Berkaki ini mempunyai spesifikasi sebagai berikut :

1. Pembuatan rangka menggunakan PCB yang dipotong sesuai desain dan digabung dengan servo menggunakan lem dan mur baut.

2. Sendi – sendi yang bergerak digerakan oleh servo *EMAX ES80MAII*, servo jenis ini mempunyai spesifikasi tegangan kerja 5 volt, merupakan servo standar dan menggunakan metal gear.
3. Unit masukan , sensor *infra red* yang berjumlah 8 sebagai penunjuk arah jalan robot dan tombol navigasi yang berjumlah 4 sebagai inputan perintah langsung robot untuk mengantar makanan ke meja di restoran.
4. Unit keluaran, motor servo sebagai penggerak yang kaki-kaki robot agar bisa berjalan sebagai mana mestinya dengan tegangan 5 VDC dan torsi 1.8 kg.
5. Arah gerak terprogram di chip Mikrokontroler *ATmega328* dan mengirimkan ke servo kontroler.
6. Sistem pengendalian menggunakan satu chip yang merupakan mikrokontroler *ATmega328*.
7. Tegangan kerja yang digunakan adalah 5 Volt DC yang bersumber dari baterai lippo.
8. Keluaran dari robot berkaki berupa gerakan. Jika tombol power dihidupkan makan robot akan menyala dan robot akan mensejajarkan kaki – kakinya, setelah diberi inputan dari *Push Button* robot akan berjalan menuju meja konsumen dengan membawa pesanan konsumen yang telah dipesan. Tombol power dimatikan maka robot baru dapat berhenti.

## **H. Pengujian Alat**

Pengujian alat dilakukan untuk mendapatkan data penelitian. Pengujian alat ini dilakukan dengan dua pengujian, yaitu:

### **1. Uji Fungsional**

Pengujian alat dilakukan dengan cara menguji setiap bagian alat berdasarkan karakteristik dan fungsi masing – masing. Pengujian ini dilakukan untuk mengetahui apakah setiap bagian dari perangkat telah bekerja sesuai dengan fungsi dan keinginan.

### **2. Uji Unjuk Kerja**

Pengujian unjuk kerja alat dilakukan dengan cara melihat unjuk kerja alat. Hal-hal yang perlu diamati antara lain: rangkaian sistem minimum ATmega328 dan pergerakan servo yang telah terpasang. Kondisi baterai ditandai dengan pergerakan robot yang baik atau kurang baik. Dari pengujian ini akan diketahui kinerja dari alat yang dibuat

## **I. Pengoperasian Alat**

Pengoperasian alat ini dapat dilakukan dengan cara sebagai berikut:

1. Pastikan alat terhubung dengan tegangan yang bersumber dari baterai dan sudah distabilkan menjadi 5 v.
2. Pastikan semua kabel terhubung dengan sempurna
3. Tekan saklar pada posisi ON untuk menghidupkan tekan tombol start untuk mulai menjalankan
4. Tekan saklar pada posisi OFF untuk mematikan.

## **BAB IV**

### **PENGUJIAN DAN PEMBAHASAN**

Tujuan dari pengujian dan pembahasan adalah untuk mengetahui kinerja alat baik secara perbagian blok rangkaian maupun sistem keseluruhan apakah sudah seperti yang diharapkan atau belum. Pengujian ini meliputi

#### **A. Hasil pengujian Fungsional**

##### **1. Pengujian Tegangan Catu Daya**

Pengukuran pada catu daya sangat penting karena catu daya merupakan sumber daya untuk seluruh sistem Robot *Line Follower* Berkaki agar dapat berjalan dengan baik. Pengukuran dilakukan pada *input output* catu daya yakni IC Regulator 7805. Hal ini dimaksudkan untuk mengetahui besarnya tegangan kerja yang masuk sebelum rangkaian sistem minimum ATmega328. ATmega328 beroperasi pada tegangan masukan 4.5-6 Volt.

Tabel 2 adalah hasil dari pengukuran catu daya. Pengukuran dilakukan dengan mengukur tegangan *input* dan tegangan *output* untuk mengetahui respon dari robot dengan menganalisa penggunaan baterai, pengukuran juga dilakukan dengan mengukur tegangan *input* dan tegangan *output* saat tanpa beban dan pada saat dengan beban. Pengukuran tegangan dilakukan dengan menggunakan multimeter digital DEKO DM-96L.

Tabel 2. Pengukuran Tegangan Regulator 7805

No	In (Volt)	Out (Volt)	Respon Robot
1	8,4	5,05	Normal
2	8,2	5,05	Normal
3	8,0	5,05	Normal
4	7,8	5,05	Normal
5	7,6	5,05	Normal
6	7,4	5,05	Normal
7	7,2	5,05	Normal
8	7,0	5,05	Normal

## 2. Pengujian Tegangan Sistem Minimum Saat Tanpa Beban Dan Saat Ada Beban.

Tabel 3. Pengukuran Tegangan Sistem Minimum Saat Tanpa Beban Dan Saat Ada Beban

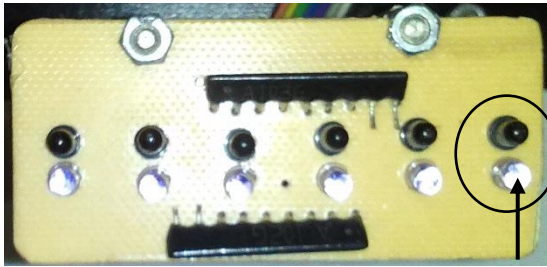
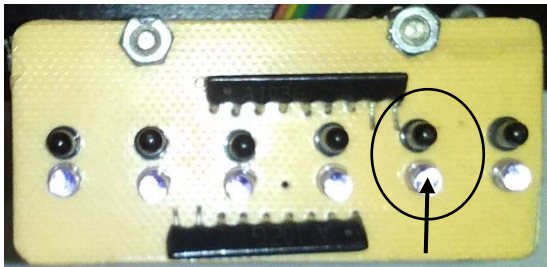
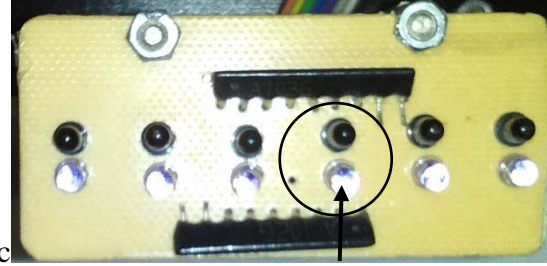
No	Pengukuran	Tegangan Input (V)	Tegangan Output (V)
			LM7805
1	Tanpa Beban	8.04	5,05
2	Dengan Beban	8.04	5,05

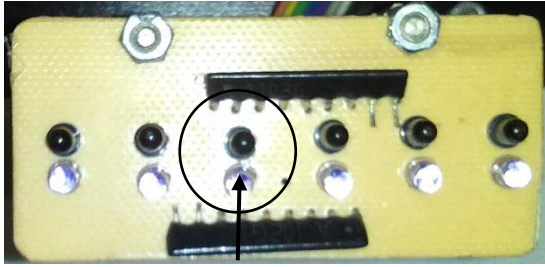
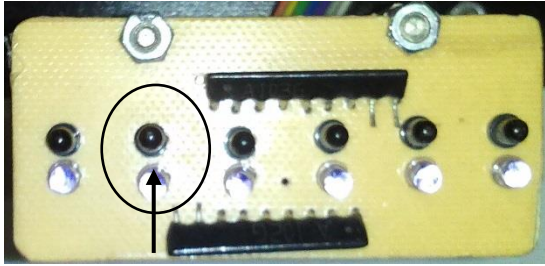

Hasil pengukuran tegangan *output* sistem minimum saat tanpa beban dan saat ada beban adalah 5,05 V, dan keadaan keluaran sistem minimum setabil jadi sistem minimum dapat bekerja dengan baik.

### 3. Pengujian Sensor *Infrared*

Pengujian sensor *Infra Red* bertujuan untuk mengetahui karakteristik sensor, dengan mengukur keluaran tegangan sensor ketika terkena bidang hitam atau putih. Sehingga dilakukan percobaan seperti tabel dibawah ini.

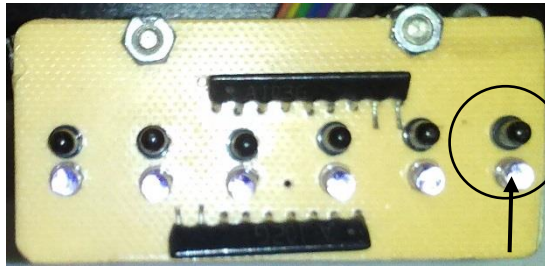
Tabel 4. Pengukuran Sensor *Infrared* Terhadap Garis Putih

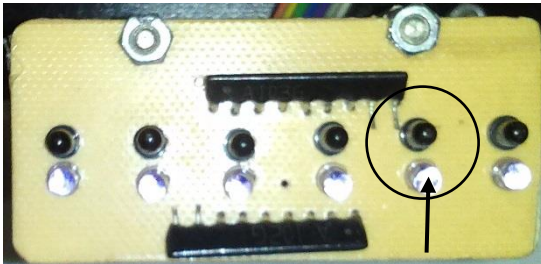
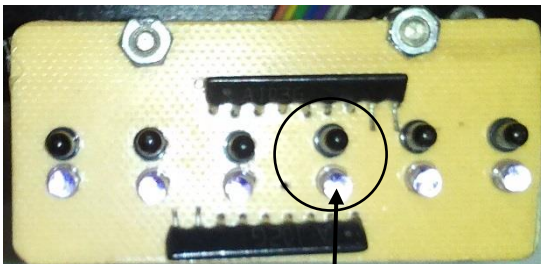
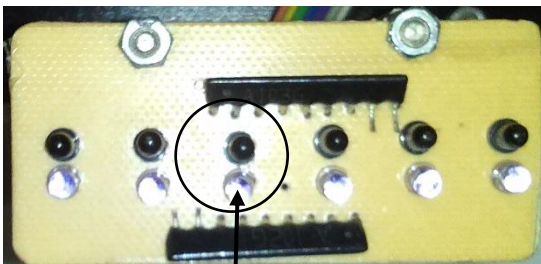
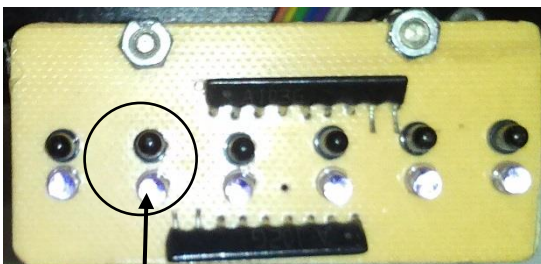
No	Kondisi Sensor	Vout (Volt)	Warna Garis
1		0,24	Putih
2		0,24	Putih
3		0,52	Putih


4		0,19	Putih
5		0,16	Putih
6		0,17	Putih

Pada tabel 4 menunjukkan keadaan tegangan  $V_{out}$  ketika sensor membaca garis putih.

Tabel 5. Pengukuran Sensor *Infra Red* Terhadap Garis Hitam

No	Kondisi Sensor	$V_{out}$ (Volt)	Warna Garis
1		4,72	Hitam

No	Kondisi Sensor	Vout (Volt)	Warna Garis
2		4,74	Hitam
3		4,74	Hitam
4		4,72	Hitam
5		4,71	Hitam

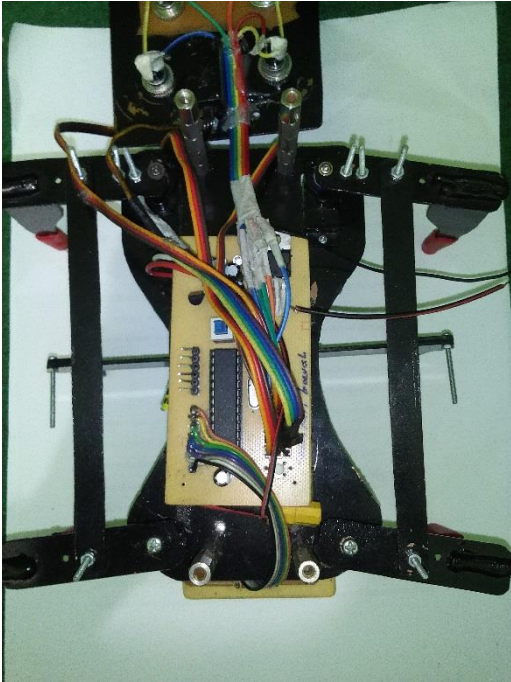
No	Kondisi Sensor	Vout (Volt)	Warna Garis
6		4,64	Hitam

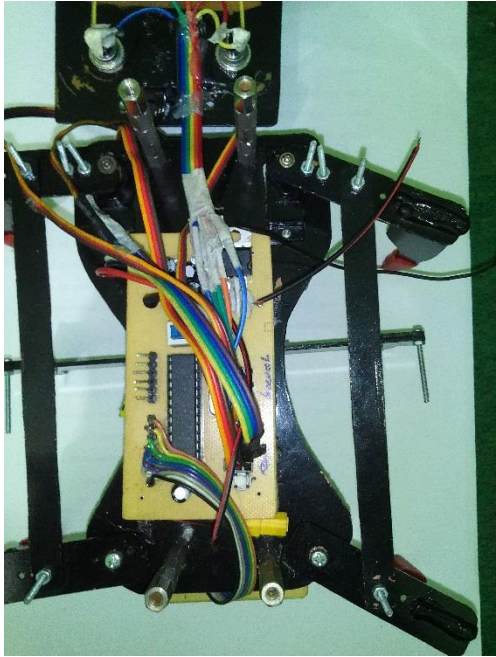
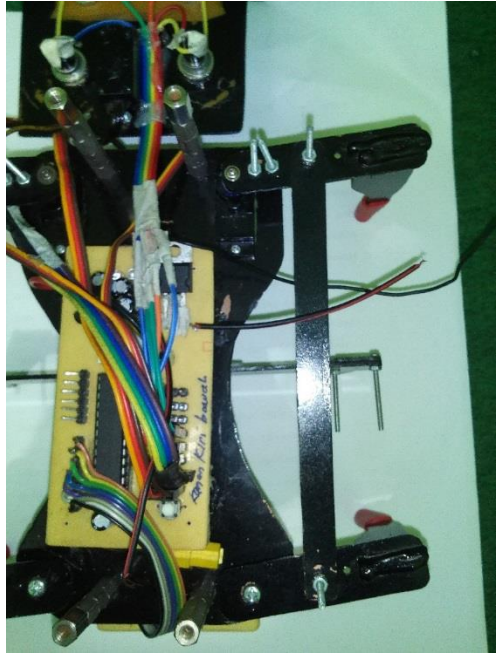
Dari tabel 4 dan 5 digunakan untuk mengetahui kinerja atau baik atau tidak, karena sensor *infra red* akan digunakan sebagai inputan pada robot, yang digunakan oleh robot untuk mendeteksi garis lurus, pertigaan, atau perempatan. Sebagai contoh ketika terkena bidang hitam nilai tegangan sensor infra red 0,24 volt dan ketika terkena bidang putih nilai tegangan sensor infrared 4,72 volt. Dari pengambilan data sensor tersebut dapat disimpulkan bahwa ketika sensor infrared terkena bidang putih nilai tegangannya lebih rendah daripada sensor infrared terkena bidang hitam karena prinsip infrared memancarkan cahaya dan photodioda menangkap cahaya. Jika cahaya tersebut terkena bidang putih maka akan memantul kearah photodioda, jika terkena cahaya garis hitam maka cahaya cenderung terserap oleh warna hitam maka cahaya yang dipantulkan ke photodioda sedikit.

#### 4. Pengujian Gerakan Motor Servo Robot

Pengujian gerakan servo dengan menggunakan servo kontroler sebagai kontrol, dapat dilihat pada tabel 6 Sudut yang diambil merupakan gerakan yang terjadi pada robot.

Tabel 6. Pengujian Motor Servo Kaki Samping

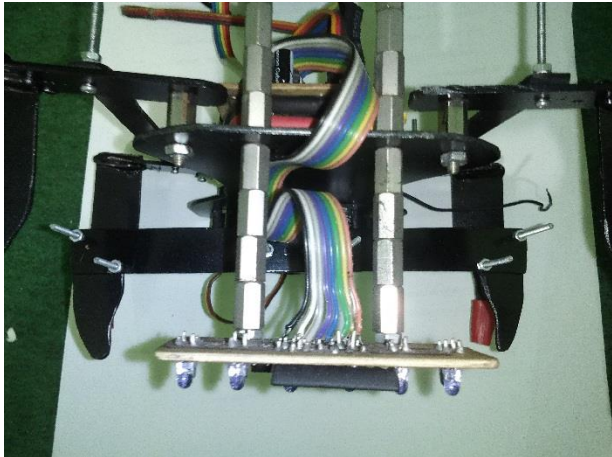

No	Sudut ( $^{\circ}$ )	Kondisi ( gambar )
1	80	


2	45	
3	110	

Pada Tabel 6 menunjukkan sudut pada kaki robot line follower berkaki yang berbeda-beda. Pada sudut 80 derajat, ini merupakan posisi awal kaki

robot saat saklar ON. Pada sudut 45 derajat kaki robot akan bergerak kedepan dengan diikuti oleh kaki depan robot dan kaki belakang robot karena kaki belakang dengan kaki depan robot terhubung oleh penampang. Pada sudut 110 kaki robot akan bergerak ke belakang yakni membentuk sudut 110 derajat, sudut 110 derajat sama prinsipnya dengan sudut 45 derajat, jika kaki belakang bergerak kebelakang maka kaki depan robot akan ikut bergerak membentuk sudut 110 derajat.

Tabel 7. Pengujian Motor Servo Kaki Bawah

No	Sudut ( ° )	Kondisi ( gambar )
1	90	
2	45	

3	135	
---	-----	--

Pada tabel 7 menunjukkan pergerakan kaki bawah pada Robot *Line Follower* Berkaki yang menunjukkan pergerakan mulai dari sudut 90, 45 dan 135. Pada sudut 90 derajat merupakan posisi awal robot saat saklar ON. Pada sudut 45 derajat merupakan gerakan kaki bawah sebelah kanan robot yang bergerak keatas untuk mengangkat badan robot sebelah kiri agar kaki samping kanan robot bisa bergerak kedepan. Pada sudut 135 derajat merupakan gerakan kaki bawah sebelah kiri robot yang bergerak ke atas untuk mengangkat badan robot sebelah kanan agar kaki samping kanan robot dapat bergerak kedepan.

##### **5. Pengujian Unjuk Kerja Keseluruhan**

Pengujian unjuk kerja dilakukan untuk mengetahui kinerja keseluruhan bagian robot sebagai sebuah sistem yang utuh Robot *Line Follower* Berkaki. Pengujian awal akan dilakukan untuk menguji kemampuan robot dalam mengantar nampan ke meja. Tempat yang akan menjadi uji coba

adalah sebuah miniatur restoran yang didalamnya terdapat 4 meja percobaan.

Pengujian akan dilakukan dengan meletakkan Robot *Line Follower* Berkaki untuk bisa meletakkan nampan ke meja yang telah ditentukan. Hasil pengujian dapat dilihat pada tabel

Tabel 8. Hasil Pengujian Unjuk Kerja Robot

No	Push Button	Keberhasilan	Keterangan	Waktu (s)
1	1	Robot dapat mengantar nampan ke meja 1	Robot melewati perempatan, belok kanan ke meja 1	18.32
2	2	Robot dapat mengantar nampan ke meja 2	Robot melewati perempatan, lurus, pertigaan belok kanan ke meja 2	33.31
3	3	Robot dapat mengantar nampan ke meja 3	Robot melewati perempatan, lurus, pertigaan belok kiri ke meja 3	34.22
4	4	Robot dapat mengantar nampan ke meja 4	Robot melewati perempatan belok kiri ke meja 4	18.25

Dari tabel 7. Hasil pengujian yang didapat sudah sesuai dengan yang diinginkan penulis dan dalam pengujian tersebut robot bisa mengantarkan makanan ke meja yang telah ditentukan.

## B. Pembahasan

Berdasarkan tahapan pengujian yang telah dilakukan maka dapat ditarik kesimpulan bahwa sistem yang telah dirancang dapat bekerja sebagaimana mestinya, meskipun terdapat *error* atau kesalahan pada beberapa bagian.

Berikut adalah pembahasan dari pengujian yang telah dilakukan :

## 1. Hardware

### a. Catu Daya

hasil pengukuran catu daya dapat bekerja dengan baik, dengan melihat penurunan tegangan dari 12,6 Volt sampai dengan 11,0 Volt. Tegangan *output* sudah sesuai dengan yang diharapkan dan memenuhi dengan tegangan kerja mikrokontroler ATmega328 yaitu sebesar 5 Volt. Sementara sumber untuk keseluruhan sistem adalah sebesar 12 V.

### b. Sensor *Infra Red*

Sensor *infra red* dapat berfungsi dengan baik dan mempunyai akurasi yang baik. Hasil pengujian menunjukkan bahwa sensor *infra red* memiliki tingkat sensitifitas yang baik dengan rata-rata kesalahan sebesar

### c. Motor Servo

Motor servo dapat bergerak sesuai dengan yang diinginkan untuk menggerakkan kaki robot. Penggerak servo bergerak sesuai

## 2. Software

### a. Pemrograman Sistem Minimum ATmega328

Bahasa pemrograman C# dikenal sebagai bahasa pemrograman yang mudah dipahami bagi pemula untuk belajar bahasa pemrograman karena bahasanya lebih mendekati ke bahasa manusia. Bahasa pemrograman C# sudah banyak digunakan untuk aplikasi mikrokontroler yang didukung dengan *software* Arduino yang dapat dinikmati secara gratis dan memiliki fitur, *user interface* yang dapat diakses secara mudah. Setiap bahasa pemrograman mempunyai standar

penulisan program. Konstruksi dari program bahasa C# untuk Arduino harus memiliki aturan sebagai berikut:

### 1. Definisi *hardware*

Definisi *hardware* adalah pendefinisian dari *hardware* yang diinginkan pada sebuah proyek menggunakan system minimum ATmega328. Penulisan dalam program sebagai berikut:

```
#include <Servo.h>

Servo myservo1; // create servo object to
control a servo
Servo myservo2;
Servo myservo3;
int kecepatan;
int ulang;

int sensor1v;
int sensor2v;
int sensor3v;
int sensor4v;
int sensor5v;
int sensor6v;

int memory;

const int sensor1 = A0;
const int sensor2 = A1;
const int sensor3 = A2;
const int sensor4 = A3;
const int sensor5 = A4;
const int sensor6 = A5;
```

Baris diatas menyatakan bahwa *hardware* yang digunakan berupa motor servo dan juga sensor.

### 2. Definisi Pemrosesan Awal (*Preprocessor*)

*Preprocessor* membaca simbol-simbol khusus di dalam kode yang disebut pengarah *preprocessor* yang

dimasukkan didalam program *compiler* untuk mengompilasi program. Penulisan *preprocessor* dalam program ini adalah sebagai berikut :

```
const int sensor1 = A0;
const int sensor2 = A1;
const int sensor3 = A2;
const int sensor4 = A3;
const int sensor5 = A4;
const int sensor6 = A5;
const int pintombol1 = 4;
const int pintombol2 = 11;
const int pintombol3 = 6;
const int pintombol4 = 7;
```

### 3. Definisi *variable*

*Variabel* adalah suatu pengenal (*identifier*) yang digunakan untuk mewakili suatu nilai tertentu di dalam proses program yang dapat diubah-ubah sesuai dengan kebutuhan. Nama dari *variable* terserah sesuai dengan yang diinginkan namun hal yang terpenting adalah setiap *variabel* diharuskan :

1. Terdiri dari gabungan huruf dan angka dengan karakter pertama harus berupa huruf, *max* 32

karakter.

2. Tidak boleh mengandung spasi atau simbol-simbol khusus seperti : \$, ?, %, #, !, &, \*, (, ), -, +, = dan lain sebagainya kecuali *underscore*.
3. Deklarasi sangat diperlukan bila akan menggunakan pengenalan (*identifier*) dalam suatu program.

#### 4. Deklarasi Fungsi

Fungsi merupakan bagian yang terpisah dari program dan dapat dipanggil di manapun di dalam program. Fungsi dalam Bahasa Basic ada yang sudah disediakan sebagai fungsi pustaka seperti print, *input* data dan untuk menggunakannya tidak perlu dideklarasikan.

#### 5. Deklarasi *Variable*

Bentuk umum pendeklarasian suatu *variable* di Arduino adalah tipe\_data AS nama\_variable

Contoh:

```
int kecepatan;
int ulang;

int sensor1v;
int sensor2v;
int sensor3v;
int sensor4v;
int sensor5v;
int sensor6v;

int memory;
```

## 6. Operator

### ➤ Operator Penugasan

Operator Penugasan (*Assignment operator*) dalam Bahasa

Basic berupa “=”.

### ➤ Operator Aritmatika

\* : untuk perkalian

/ : untuk pembagian

+ : untuk penambahan

- : untuk pengurangan

% : untuk sisa pembagian (modulus)

### ➤ Operator Hubungan (Perbandingan)

Operator hubungan digunakan untuk membandingkan hubungan dua buah operand atau sebuah nilai / variable, misalnya :

= 'Equality  $X = Y$

< 'Less than  $X < Y$

> 'Greater than  $X > Y$

<= 'Less than or equal to  $X <= Y$

>= 'Greater than or equal to  $X >= Y$

## 7. Pernyataan Kondisional (*IF-THEN – END IF*)

Pernyataan ini digunakan untuk melakukan pengambilan keputusan terhadap dua buah bahkan lebih kemungkinan untuk

melakukan suatu blok pernyataan atau tidak. Konstruksi penulisan pernyataan *IF-THEN-ELSE-END IF* pada bahasa BASIC ialah sebagai berikut:

*IF* pernyataan kondisi 1 *THEN*

‘blok pernyataan 1 yang dikerjakan bila kondisi 1 terpenuhi

*IF* pernyataan kondisi 2 *THEN*

‘blok pernyataan 2 yang dikerjakan bila kondisi 2 terpenuhi

*IF* pernyataan kondisi 3 *THEN*

‘blok pernyataan 3 yang dikerjakan bila kondisi 3 terpenuhi

### **3. Unjuk kerja Keseluruhan**

Pengujian unjuk kerja dilakukan untuk mengetahui kinerja keseluruhan bagian robot sebagai sebuah sistem yang utuh Robot Line Follower Berkaki. Pengujian awal akan dilakukan untuk menguji kemampuan robot dalam mengantar nampan ke meja. Tempat yang akan menjadi uji coba adalah sebuah miniatur restoran yang didalamnya terdapat 4 meja percobaan. Berikut ini gambaran miniatur ruangan yang digunakan sebagai tempat pengujian yang terlihat pada gambar 5

Seperti terlihat pada gambar, terdapat 4 buah ruangan yang dilengkapi masing-masing ruangan adalah sebuah meja. Pengujian akan dilakukan dengan meletakkan robot line follower berkaki untuk bisa meletakkan

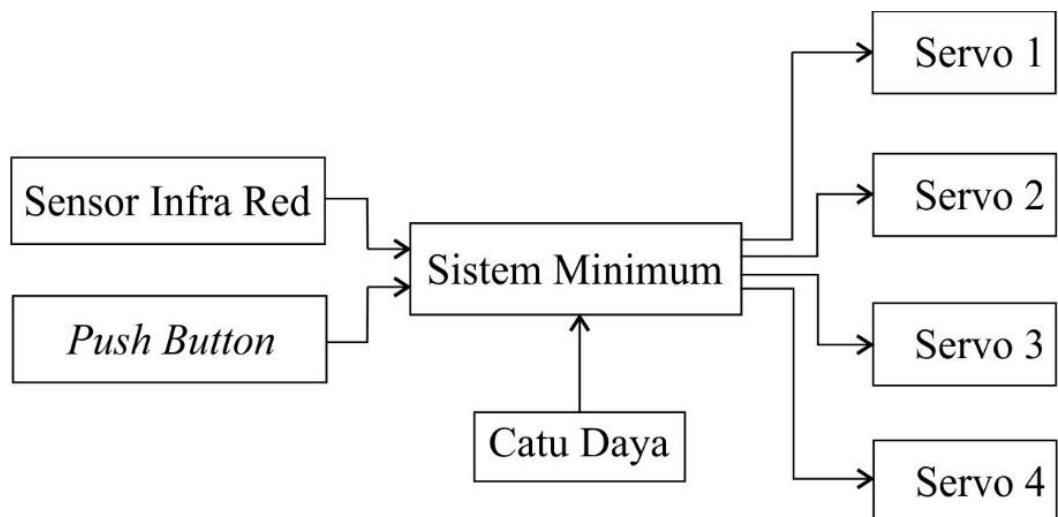
nampan ke meja yang telah ditentukan. Hasil pengujian dapat dilihat pada tabel

Tabel 9. Hasil Pengujian Unjuk Kerja Robot

No	Push Button	Keberhasilan	Keterangan	Waktu (s)
1	1	Robot dapat mengantar nampan ke meja 1	Robot melewati perempatan, belok kanan ke meja 1	18.32
2	2	Robot dapat mengantar nampan ke meja 2	Robot melewati perempatan, lurus, pertigaan belok kanan ke meja 2	33.31
3	3	Robot dapat mengantar nampan ke meja 3	Robot melewati perempatan, lurus, pertigaan belok kiri ke meja 3	34.22
4	4	Robot dapat mengantar nampan ke meja 4	Robot melewati perempatan belok kiri ke meja 4	18.25

Dari tabel 9. Hasil pengujian yang didapat sudah sesuai dengan yang diinginkan penulis dan dalam pengujian tersebut robot bisa mengantarkan makanan ke meja yang telah ditentukan.

### C. Cara Kerja Prototipe Robot *Line Follower* Berkaki



Gambar 26. Diagram Blok Cara Kerja

Pada gambar 26 dapat dilihat bagaimana proses yang dilakukan pada Prototipe Robot *Line Follower* Berkaki sebagai pelayan restoran berbasis Mikrokontroler ATmega328 :

1. Sensor *infrared* merupakan salah satu jenis LED (Light Emitting Diode) yang dapat memancarkan cahaya infra merah yang tidak kasat mata. Pada proyek akhir ini *infrared* digunakan sebagai sensor pendeteksi garis.
2. *Push button* di proyek akhir ini digunakan sebagai *input* perintah untuk menghantarkan nampan ke meja.
3. Sistem minimum sebagai pengolah data input sensor dan *push button*.
4. Catu daya sebagai sumber tegangan robot agar dapat beroperasi sesuai program yang telah dibuat.
5. Servo sebagai *output* untuk menggerakkan robot agar bisa berjalan dan mendorong nampan.

## BAB V

### KESIMPULAN DAN SARAN

#### A. Kesimpulan

Berdasarkan hasil pengujian yang telah dilakukan terhadap Prototipe Robot *Line Follower* Berkaki Pelayan Restoran” maka dapat disimpulkan :

1. Perangkat keras “Prototipe Robot *Line Follower* Berkaki” berhasil dibuat dengan dengan beberapa langkah diantaranya dimulai dari (1) Analisa Kebutuhan, (2) Perancangan rangkaian, (3) Langkah Pembuatan alat, (4) Pengujian alat, dan (5) pengambilan data.
2. Perangkat lunak yang digunakan untuk pembuatan program Prototipe Robot *Line Follower* Berkaki adalah software Arduino untuk memprogram sistem minimum Robot *Line Follower* Berkaki. Secara keseluruhan program yang dibuat sudah dapat bekerja sesuai dengan tujuan.

Tabel. Program Sistem Minimum

KONTRUKSI	PROGRAM	KETERANGAN
Definisi Hardware	Servo myservo1; Servo myservo2; int sensor1v; int sensor2v;	Baris disamping menyatakan bahwa <i>hardware</i> yang digunakan berupa servo , sensor <i>infra red</i>

Definisi Pemrosesan Awal ( <i>Preprocessor</i> )	<pre>const int sensor1 = A0; const int sensor2 = A1;</pre>	Baris disamping merupakan program pengenalan input yang dimana inputnya merupakan sensor <i>infra red</i> .
Definisi Variable	<pre>kecepatan = 130; intensitas = 700; ulang = 0; memory=0;</pre>	<i>Variabel</i> adalah suatu pengenal ( <i>identifier</i> ) yang digunakan untuk mewakili suatu nilai tertentu di dalam proses program yang dapat diubah-ubah sesuai dengan kebutuhan. Nama dari <i>variable</i> terserah sesuai dengan yang diinginkan
Operator	<pre>if (sensor1v&gt;intensitas &amp; sensor2v&lt;intensitas &amp; sensor3v&lt;intensitas &amp; sensor4v&lt;intensitas &amp; sensor5v&lt;intensitas &amp; sensor6v&lt;intensitas) // bagian ini adalah eksekusi pada sensor {</pre>	Operator hubungan digunakan untuk membandingkan hubungan dua buah operand atau sebuah nilai / variable
Pernyataan Kondisional (IF-THEN-END IF)	<pre>} else if (sensor1v&gt;intensitas &amp; sensor2v&gt;intensitas &amp; sensor3v&lt;intensitas &amp; sensor4v&lt;intensitas &amp; sensor5v&lt;intensitas &amp; sensor6v&lt;intensitas) {</pre>	Pernyataan ini digunakan untuk melakukan pengambilan keputusan terhadap dua buah bahkan lebih kemungkinan untuk

		melakukan suatu blok pernyataan atau tidak. <i>IF</i> pernyataan kondisi 1 <i>THEN</i> 'blok pernyataan 1 yang dikerjakan bila kondisi 1 terpenuhi
--	--	---

3. Pengujian unjuk kerja Robot *Line Follower* Berkaki yaitu dengan menguji coba robot untuk mengantar nampan makanan ke meja pelanggan, dan hasilnya berhasil dengan catatan waktu yang berbeda, percobaan dilakukan ke 4 meja yang berbeda. Waktu yang dibutuhkan robot untuk mengantar makanan ke meja 1 yakni (18.32/s), ke meja 2 (33.31/s), ke meja 3 (34.33/s), dan ke meja 4 (18.25/s). Dan robot dapat bekerja sebagaimana mestinya.

#### B. Keterbatasan Alat

Alat yang dibuat masih mempunyai banyak keterbatasan, beberapa keterbatasan antara lain :

1. Suara robot waktu berjalan terdengar keras berdentum ke lantai.
2. Tidak adanya *User Interface* pada robot
3. Sistem berhentinya robot ke meja yakni dengan ditandai kotakan garis hitam, jadi robot kemungkinan kurang presisi masih ada untuk menempel ke meja dan meletakkan nampan ke meja.

C. Saran dalam pembuatan proyek akhir ini tentunya terdapat kekurangan, sehingga diperlukan pengembangan lebih lanjut. Saran membangun yang

dibutuhkan untuk menyempurnakan proyek akhir ini, antara lain sebagai berikut:

1. Membuat dan menggunakan sepatu untuk robot dengan bahan yang halus dan tidak licin seperti spon.
2. Perlu adanya *user interface* agar mudah diakses oleh pengguna dan dimonitoring oleh pengguna.
3. Robot perlu dilengkapi dengan limit switch dibagian depan agar robot bisa berhenti dan menempel ke meja dengan mendekati presisi ke meja untuk meletakkan nampan makanan.

## DAFTAR PUSTAKA

- (anonim.(2012, 30 juni). *Pengertian dan kelebihan Mikrokontroler*. Diakses tanggal 24 September 2016, dari <http://elektronika-dasar.web.id/pengertian-dan-kelebihan-mikrokontroler/>)
- (anonim.(2015). *Baterai Li-ion dengan Li-Polimer*. Diakses tanggal 22 September 2016, dari <http://www.webkeren.net/2015/05/spesifikasi-dan-perbedaan-baterai-li-ion-dengan-li-polimer.html>)
- (anonim.(2012, 30 juni).*Pengertian dan kelebihan Mikrokontroler*.Diakses tanggal 24-9-2016, dari <http://elektronika-dasar.web.id/pengertian-dan-kelebihan-mikrokontroler/>)
- Budiharto.W. (2010). *Robotika –Teori dan Implementasinya*.Yogyakarta: C.V. ANDI OFFSET
- <http://www.alldatasheet.com>, diakses tanggal 14 Oktober 2016
- <http://www.arduino.cc>, diakses tanggal 15 Oktober 2016
- Setiawan. A. (2010) *20 Aplikasi Mikrokontroler ATmega16 menggunakan BASCOM-AVR*.Yogyakarta: C.V. ANDI OFFSET
- Winarno dan Deni Arifianto. (2011). *Bikin Robot Itu Gampang*. Jakarta:PT Kawan Pustaka

# LAMPIRAN



8-bit AVR Microcontrollers

ATmega328/P

DATASHEET SUMMARY

## Introduction

The Atmel® picoPower® ATmega328/P is a low-power CMOS 8-bit microcontroller based on the AVR® enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328/P achieves throughputs close to 1MIPS per MHz. This empowers system designer to optimize the device for power consumption versus processing speed.

## Feature

High Performance, Low Power Atmel®AVR® 8-Bit Microcontroller Family

- Advanced RISC Architecture
  - 131 Powerful Instructions
  - Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20 MIPS Throughput at 20MHz
  - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
  - 32KBytes of In-System Self-Programmable Flash program Memory
  - 1KBytes EEPROM
  - 2KBytes Internal SRAM
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data Retention: 20 years at 85°C/100 years at 25°C<sup>(1)</sup>
  - Optional Boot Code Section with Independent Lock Bits
    - In-System Programming by On-chip Boot Program
    - True Read-While-Write Operation
  - Programming Lock for Software Security
- Atmel® QTouch® Library Support
  - Capacitive Touch Buttons, Sliders and Wheels
  - QTouch and QMatrix® Acquisition
  - Up to 64 sense channels

- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Six PWM Channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package
    - Temperature Measurement
  - 6-channel 10-bit ADC in PDIP Package
    - Temperature Measurement
  - Two Master/Slave SPI Serial Interface
  - One Programmable Serial USART
  - One Byte-oriented 2-wire Serial Interface (Philips I<sup>2</sup>C compatible)
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - One On-chip Analog Comparator
  - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
  - 23 Programmable I/O Lines
  - 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF
- Operating Voltage:
  - 1.8 - 5.5V
- Temperature Range:
  - -40°C to 105°C
- Speed Grade:
  - 0 - 4MHz @ 1.8 - 5.5V
  - 0 - 10MHz @ 2.7 - 5.5V
  - 0 - 20MHz @ 4.5 - 5.5V
- Power Consumption at 1MHz, 1.8V, 25°C
  - Active Mode: 0.2mA
  - Power-down Mode: 0.1µA
  - Power-save Mode: 0.75µA (Including 32kHz RTC)

## 1. Description

The Atmel AVR® core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in a single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega328/P provides the following features: 32Kbytes of In-System Programmable Flash with Read-While-Write capabilities, 1Kbytes EEPROM, 2Kbytes SRAM, 23 general purpose I/O lines, 32 general purpose working registers, Real Time Counter (RTC), three flexible Timer/Counters with compare modes and PWM, 1 serial programmable USARTs, 1 byte-oriented 2-wire Serial Interface (I2C), a 6-channel 10-bit ADC (8 channels in TQFP and QFN/MLF packages), a programmable Watchdog Timer with internal Oscillator, an SPI serial port, and six software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption. In Extended Standby mode, both the main oscillator and the asynchronous timer continue to run.

Atmel offers the QTouch® library for embedding capacitive touch buttons, sliders and wheels functionality into AVR microcontrollers. The patented charge-transfer signal acquisition offers robust sensing and includes fully debounced reporting of touch keys and includes Adjacent Key Suppression® (AKS™) technology for unambiguous detection of key events. The easy-to-use QTouch Suite toolchain allows you to explore, develop and debug your own touch applications.

The device is manufactured using Atmel's high density non-volatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed In-System through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The Boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega328/P is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATmega328/P is supported with a full suite of program and system development tools including: C Compilers, Macro Assemblers, Program Debugger/Simulators, In-Circuit Emulators, and Evaluation kits.

## 2. Configuration Summary

Features	ATmega328/P
Pin Count	28/32
Flash (Bytes)	32K
SRAM (Bytes)	2K
EEPROM (Bytes)	1K
Interrupt Vector Size (instruction word/vector)	1/1/2
General Purpose I/O Lines	23
SPI	2
TWI (I <sup>2</sup> C)	1
USART	1
ADC	10-bit 15kSPS
ADC Channels	8
8-bit Timer/Counters	2
16-bit Timer/Counters	1

and support a real Read-While-Write Self-Programming mechanism. There is a separate Boot Loader Section, and the SPM instruction can only execute from there. In , there is no Read-While-Write support and no separate Boot Loader Section. The SPM instruction can execute from the entire Flash.

### 3. Ordering Information

#### 3.1. ATmega328

Speed [MHz] <sup>(3)</sup>	Power Supply [V]	Ordering Code <sup>(2)</sup>	Package <sup>(1)</sup>	Operational Range
20	1.8 - 5.5	ATmega328-AU ATmega328-AUR <sup>(5)</sup> ATmega328-MMH <sup>(4)</sup> ATmega328-MMHR <sup>(4)(5)</sup> ATmega328-MU ATmega328-MUR <sup>(5)</sup> ATmega328-PU	32A 32A 28M1 28M1 32M1-A 32M1-A 28P3	Industrial (-40°C to 85°C)

**Note:**

1. This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.
2. Pb-free packaging, complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.
3. Please refer to *Speed Grades* for Speed vs.  $V_{CC}$
4. Tape & Reel.
5. NiPdAu Lead Finish.

Package Type	
28M1	28-pad, 4 x 4 x 1.0 body, Lead Pitch 0.45mm Quad Flat No-Lead/Micro Lead Frame Package (QFN/MLF)
28P3	28-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
32M1-A	32-pad, 5 x 5 x 1.0 body, Lead Pitch 0.50mm Quad Flat No-Lead/Micro Lead Frame Package (QFN/MLF)
32A	32-lead, Thin (1.0mm) Plastic Quad Flat Package (TQFP)

### 3.2. ATmega328P

Speed [MHz] <sup>(3)</sup>	Power Supply [V]	Ordering Code <sup>(2)</sup>	Package <sup>(1)</sup>	Operational Range
20	1.8 - 5.5	ATmega328P-AU	32A	Industrial (-40°C to 85°C)
		ATmega328P-AUR <sup>(5)</sup>	32A	
		ATmega328P-MMH <sup>(4)</sup>	28M1	
		ATmega328P-MMHR <sup>(4)(5)</sup>	28M1	
		ATmega328P-MU	32M1-A	
		ATmega328P-MUR <sup>(5)</sup>	32M1-A	
		ATmega328P-PU	28P3	
		ATmega328P-AN	32A	Industrial (-40°C to 105°C)
		ATmega328P-ANR <sup>(5)</sup>	32A	
		ATmega328P-MN	32M1-A	
		ATmega328P-MNR <sup>(5)</sup>	32M1-A	
		ATmega328P-PN	28P3	

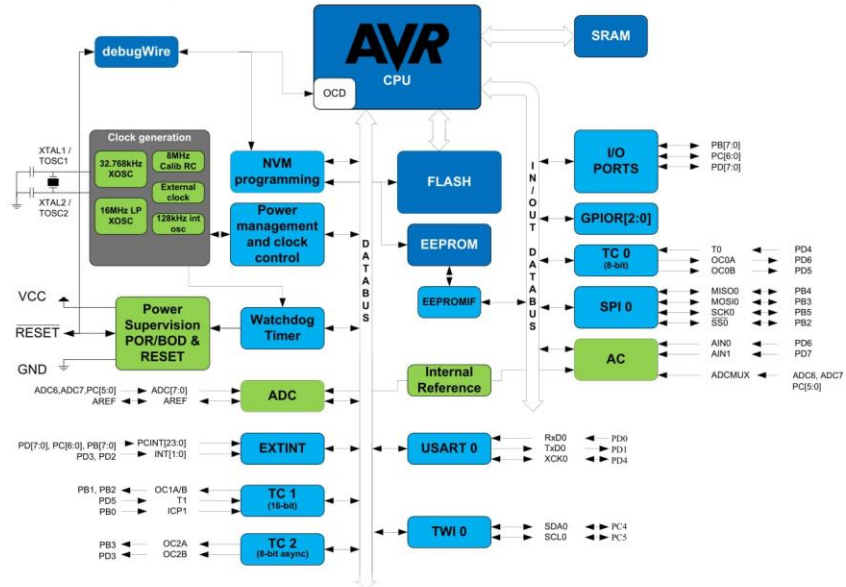
**Note:**

1. This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.
2. Pb-free packaging, complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.
3. Please refer to *Speed Grades* for Speed vs.  $V_{CC}$
4. Tape & Reel.
5. NiPdAu Lead Finish.

Package Type	
28M1	28-pad, 4 x 4 x 1.0 body, Lead Pitch 0.45mm Quad Flat No-Lead/Micro Lead Frame Package (QFN/MLF)
28P3	28-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
32M1-A	32-pad, 5 x 5 x 1.0 body, Lead Pitch 0.50mm Quad Flat No-Lead/Micro Lead Frame Package (QFN/MLF)
32A	32-lead, Thin (1.0mm) Plastic Quad Flat Package (TQFP)

## 4. Block Diagram

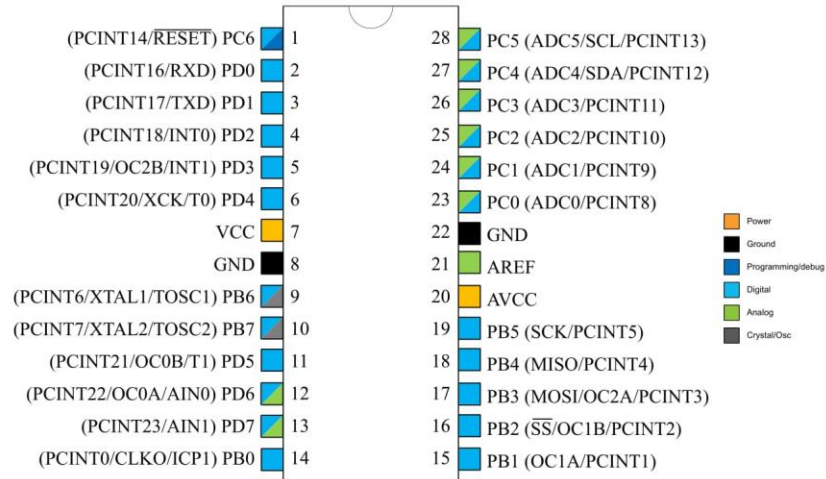
Figure 4-1. Block Diagram



## 5. Pin Configurations

### 5.1. Pin-out

Figure 5-1. 28-pin PDIP



Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB[7:6] is used as TOSC[2:1] input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

#### 5.2.4. Port C (PC[5:0])

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC[5:0] output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

#### 5.2.5. PC6/RESET

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated in the *Alternate Functions of Port C* section.

#### 5.2.6. Port D (PD[7:0])

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

#### 5.2.7. AV<sub>CC</sub>

AV<sub>CC</sub> is the supply voltage pin for the A/D Converter, PC[3:0], and PE[3:2]. It should be externally connected to V<sub>CC</sub>, even if the ADC is not used. If the ADC is used, it should be connected to V<sub>CC</sub> through a low-pass filter. Note that PC[6:4] use digital supply voltage, V<sub>CC</sub>.

#### 5.2.8. AREF

AREF is the analog reference pin for the A/D Converter.

#### 5.2.9. ADC[7:6] (TQFP and VFQFN Package Only)

In the TQFP and VFQFN package, ADC[7:6] serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

## 6. I/O Multiplexing

Each pin is by default controlled by the PORT as a general purpose I/O and alternatively it can be assigned to one of the peripheral functions.

The following table describes the peripheral signals multiplexed to the PORT I/O pins.

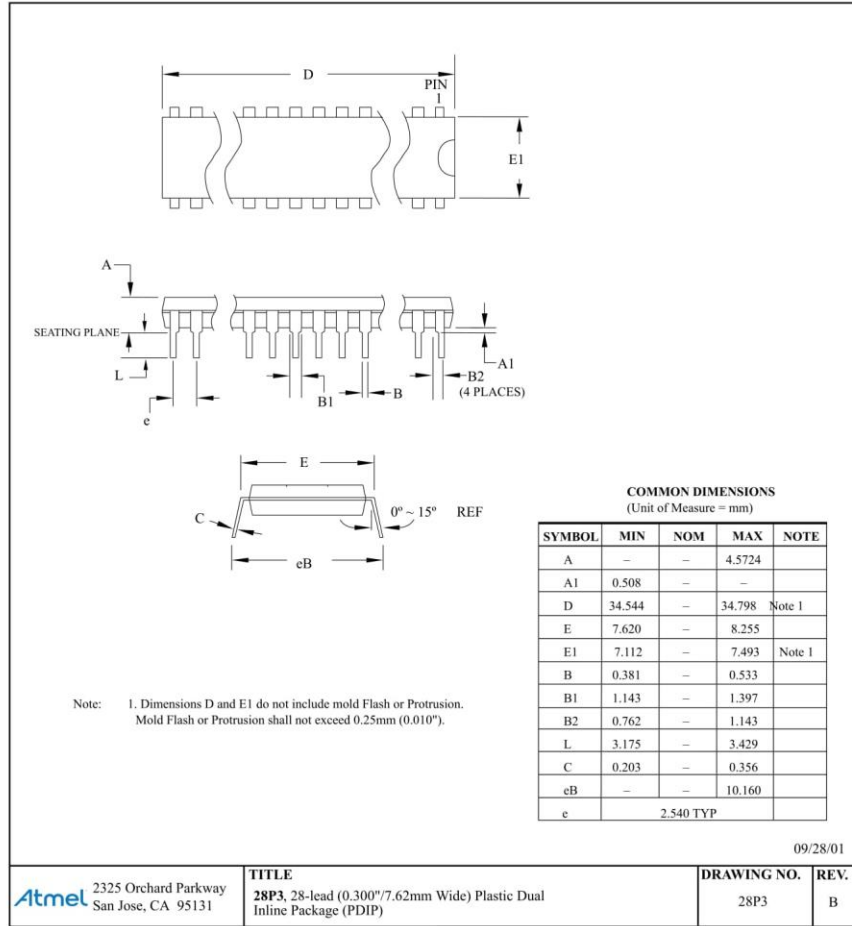
**Table 6-1. PORT Function Multiplexing**

(32-pin MLF/TQFP) Pin#	(28-pin MLF) Pin#	(28-pin PIPD) Pin#	PAD	EXTINT	PCINT	ADC/AC	OSC	T/C #0	T/C #1	USART 0	I2C 0	SPI 0
1	1	5	PD[3]	INT1	PCINT19			OC2B				
2	2	6	PD[4]		PCINT20			T0		XCK0		
3	3	7	VCC									
4	4	8	GND									
5	-	-	VCC									
6	-	-	GND									
7	5	9	PB[6]		PCINT6		XTAL1/ TOSC1					
8	6	10	PB[7]		PCINT7		XTAL2/ TOSC2					
9	7	11	PD[5]		PCINT21			OC0B	T1			
10	8	12	PD[6]		PCINT22	AIN0		OC0A				
11	9	13	PD[7]		PCINT23	AIN1						
12	10	14	PB[0]		PCINT0		CLKO	ICP1				
13	11	15	PB[1]		PCINT1			OC1A				
14	12	16	PB[2]		PCINT2			OC1B				SS0
15	13	17	PB[3]		PCINT3			OC2A				MOSI0
16	14	18	PB[4]		PCINT4							MISO0
17	15	19	PB[5]		PCINT5							SCK0
18	16	20	AVCC									
19	-	-	ADC6			ADC6						
20	17	21	AREF									
21	18	22	GND									
22	-	-	ADC7			ADC7						
23	19	13	PC[0]		PCINT8	ADC0						
24	20	24	PC[1]		PCINT9	ADC1						
25	21	25	PC[2]		PCINT10	ADC2						
26	22	26	PC[3]		PCINT11	ADC3						
27	23	27	PC[4]		PCINT12	ADC4						SDA0
28	24	28	PC[5]		PCINT13	ADC5						SCL0
29	25	1	PC[6]/ RESET		PCINT14							



(32-pin MLF/TQFP) Pin#	(28-pin MLF) Pin#	(28-pin PIPD) Pin#	PAD	EXTINT	PCINT	ADC/AC	OSC	T/C #0	T/C #1	USART 0	I2C 0	SPI 0
30	26	2	PD[0]		PCINT16					RXD0		
31	27	3	PD[1]		PCINT17					TXD0		
32	28	4	PD[2]	INT0	PCINT18							

11.4. 28-pin 28P3





## Technical Data Sheet

### 5mm Silicon PIN Photodiode , T-1 3/4

#### PD333-3B/H0/L2

#### Features

- Fast response time
- High photo sensitivity
- Small junction capacitance
- Pb free
- The product itself will remain within RoHS compliant version.

#### Descriptions

PD333-3B/H0/L2 is a high speed and high sensitive PIN photodiode in a standard 5  $\phi$  plastic package. Due to its black epoxy the device is sensitive to infrared radiation.



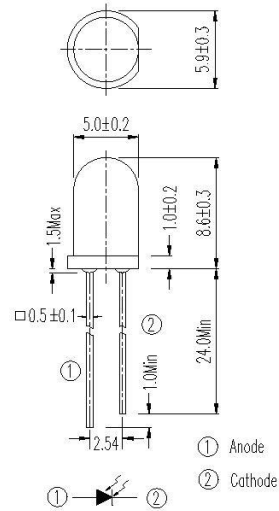
#### Applications

- High speed photo detector
- Security system
- Camera

#### Device Selection Guide

LED Part No.	Chip	Lens Color
	Material	
PD	Silicon	Black

**Package Dimensions**



- Notes:** 1.All dimensions are in millimeters  
2.Tolerances unless dimensions  $\pm 0.25$ mm

**Absolute Maximum Ratings (Ta=25°C)**

Parameter	Symbol	Rating	Units
Reverse Voltage	$V_R$	32	V
Power Dissipation	$P_d$	150	mW
Lead Soldering Temperature	$T_{sol}$	260	°C
Operating Temperature	$T_{opr}$	-40 ~ +85	°C
Storage Temperature	$T_{stg}$	-40 ~ +85	°C

**Notes:** \*1:Soldering time  $\leq 5$  seconds.

**Electro-Optical Characteristics (Ta=25°C)**

Parameter	Symbol	Condition	Min.	Typ.	Max.	Units
Rang of Spectral Bandwidth	$\lambda_{0.5}$	-----	840	---	1100	nm
Wavelength of Peak Sensitivity	$\lambda_p$	-----	---	940	---	nm
Open-Circuit Voltage	$V_{OC}$	Ee=5m W/cm <sup>2</sup> $\lambda_p=940nm$	---	0.39	---	V
Short- Circuit Current	$I_{SC}$	Ee=1m W/cm <sup>2</sup> $\lambda_p=940nm$	---	35	---	$\mu A$
Reverse Light Current	$I_L$	Ee=1m W/cm <sup>2</sup> $\lambda_p=940nm$ $V_R=5V$	25	35	---	
Dark Current	$I_d$	Ee=0m W/cm <sup>2</sup> $V_R=10V$	---	5	30	nA
Reverse Breakdown	$BV_R$	Ee=0m W/cm <sup>2</sup> $I_R=100 \mu A$	32	170	---	V
Total Capacitance	$C_t$	Ee=0m W/cm <sup>2</sup> $V_R=5V$ $f=1MHZ$	---	18	---	pF
Rise/Fall Time	$t_r/t_f$	$V_R=10V$ $R_L=1K \Omega$	---	45/45	---	nS
View Angle	$2 \theta_{1/2}$	$I_F=20mA$	--	80	--	deg



**PD333-3B/H0/L2**

**Reliability Test Item And Condition**

The reliability of products shall be satisfied with items listed below.

Confidence level : 90%

LTPD : 10%

NO.	Item	Test Conditions	Test Hours/ Cycles	Sample Sizes	Failure Judgement Criteria	Ac/R e
1	Solder heat	TEMP. : 260°C±5°C	10secs	22pcs	$I_L \leq L \times 0.8$  L : Lower  Specification Limit	0/1
2	Temperature Cycle	H : +100°C    15mins ↓ 5mins L : -40°C    15mins	300Cycles	22pcs		0/1
3	Thermal Shock	H : +100°C    5mins ↓ 10secs L : -10°C    5mins	300Cycles	22pcs		0/1
4	High Temperature Storage	TEMP. : +100°C	1000hrs	22pcs		0/1
5	Low Temperature Storage	TEMP. : -40°C	1000hrs	22pcs		0/1
6	DC Operating Life	$V_R=5V$	1000hrs	22pcs		0/1
7	High Temperature/ High Humidity	85°C / 85% R.H	1000hrs	22pcs		0/1

**Typical Electro-Optical Characteristics Curves**

Fig.5 Terminal Capacitance vs.

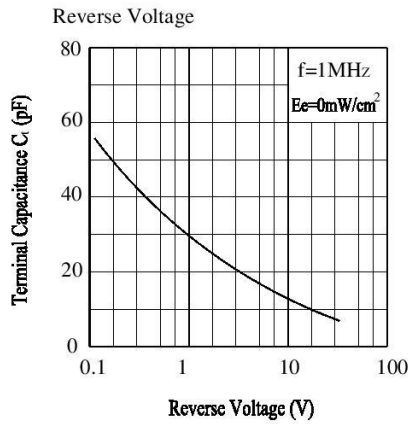


Fig.6 Response Time vs.

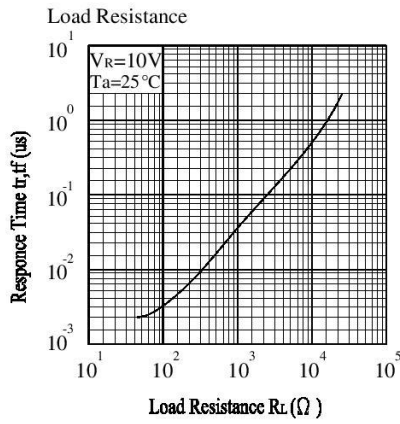


Fig.7 Relative Reverse Light Current vs. Ambient Temperature(°C)

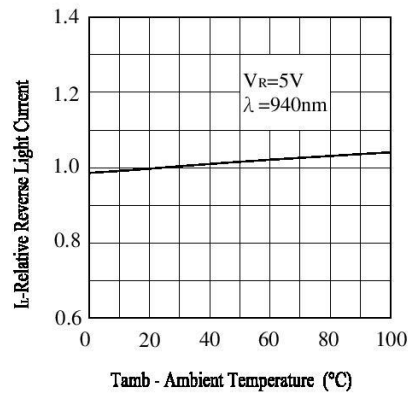
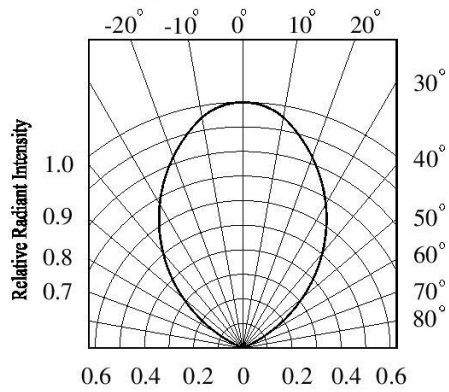


Fig.8 Relative Radiant Intensity vs. Angular Displacement





**PD333-3B/H0/L2**

**Packing Quantity Specification**

1.500PCS/1Bag · 5Bags/1Box  
2.10Boxes/1Carton

**Label Form Specification**



CPN: Customer's Production Number  
P/N : Production Number  
QTY: Packing Quantity  
CAT: Ranks  
HUE: Peak Wavelength  
REF: Reference  
LOT No: Lot Number  
MADE IN TAIWAN: Production Place

**Notes**

1. Above specification may be changed without notice. EVERLIGHT will reserve authority on material change for above specification.
2. When using this product, please observe the absolute maximum ratings and the instructions for using outlined in these specification sheets. EVERLIGHT assumes no responsibility for any damage resulting from use of the product which does not comply with the absolute maximum ratings and the instructions included in these specification sheets.
3. These specification sheets include materials protected under copyright of EVERLIGHT corporation. Please don't reproduce or cause anyone to reproduce them without EVERLIGHT's consent.

**EVERLIGHT ELECTRONICS CO., LTD.**  
Office: No 25, Lane 76, Sec 3, Chung Yang Rd,  
Tucheng, Taipei 236, Taiwan, R.O.C

Tel: 886-2-2267-2000, 2267-9936  
Fax: 886-2267-6244, 2267-6189, 2267-6306  
<http://www.everlight.com>



**PD333-3B/H0/L2**

**Reliability Test Item And Condition**

The reliability of products shall be satisfied with items listed below.

Confidence level : 90%

LTPD : 10%

NO.	Item	Test Conditions	Test Hours/ Cycles	Sample Sizes	Failure Judgement Criteria	Ac/R e
1	Solder heat	TEMP. : 260°C±5°C	10secs	22pcs	$I_L \leq L \times 0.8$  L : Lower  Specification Limit	0/1
2	Temperature Cycle	H : +100°C    15mins ↓ 5mins L : -40°C    15mins	300Cycles	22pcs		0/1
3	Thermal Shock	H : +100°C    5mins ↓ 10secs L : -10°C    5mins	300Cycles	22pcs		0/1
4	High Temperature Storage	TEMP. : +100°C	1000hrs	22pcs		0/1
5	Low Temperature Storage	TEMP. : -40°C	1000hrs	22pcs		0/1
6	DC Operating Life	V <sub>R</sub> =5V	1000hrs	22pcs		0/1
7	High Temperature/ High Humidity	85°C / 85% R.H	1000hrs	22pcs		0/1

## Lampiran 3.Sensor *Infrared*



### 5mm Infrared LED IR533C



#### Features

- High reliability
- High radiant intensity
- Peak wavelength  $\lambda_p=940\text{nm}$
- 2.54mm Lead spacing
- Low forward voltage
- Pb Free
- This product itself will remain within RoHS compliant version.

#### Description

- EVERLIGHT's Infrared Emitting Diode (IR533C) is a high intensity diode , molded in a water clear plastic package.
- The device is spectrally matched with phototransistor , photodiode and infrared receiver module.

#### Applications

- Infrared applied system

I

R	1	Copyright © 2010, Everlight All Rights Reserved. Release Date : MAY 27, 2013. Issue No: DIR-0000843	<a href="http://www.everlight.com">www.everlight.com</a>
LifecyclePhase: Approved		Expired Period: Forever	

**Device Selection Guide**

Chip Materials	Lens Color
GaAlAs	Water clear

**Absolute Maximum Ratings (Ta=25°C)**

Parameter	Symbol	Rating	Unit
Continuous Forward Current	$I_f$	100	mA
Peak Forward Current(*1)	$I_{fp}$	1.0	A
Reverse Voltage	$V_R$	5	V
Operating Temperature	$T_{opr}$	-40 ~ +85	°C
Storage Temperature	$T_{stg}$	-40 ~ +100	°C
Soldering Temperature(*2)	$T_{sol}$	260	°C
Power Dissipation at (or below) 25°C Free Air Temperature	$P_d$	150	mW

Notes: \*1: $I_{fp}$  Conditions--Pulse Width  $\leq 100\mu s$  and Duty  $\leq 1\%$ .  
\*2:Soldering time  $\leq 5$  seconds.

**Electro-Optical Characteristics (Ta=25°C)**

Parameter	Symbol	Min.	Typ.	Max.	Unit	Condition
Radiant Intensity	$I_e$	4.0	7.8	----	mW/sr	$I_f=20mA$
		----	35	----		$I_f=100mA$
Peak Wavelength	$\lambda_p$	----	940	----	nm	$I_f=20mA$
Spectral Bandwidth	$\Delta\lambda$	----	45	----	nm	$I_f=20mA$
Forward Voltage	VF	----	1.2	1.5	V	$I_f=20mA$
		----	1.4	1.8		$I_f=100mA$
Reverse Current	$I_n$	----	----	10	$\mu A$	$V_R=5V$
View Angle	2 $\theta$ 1/2	----	25	----	deg	$I_f=20mA$

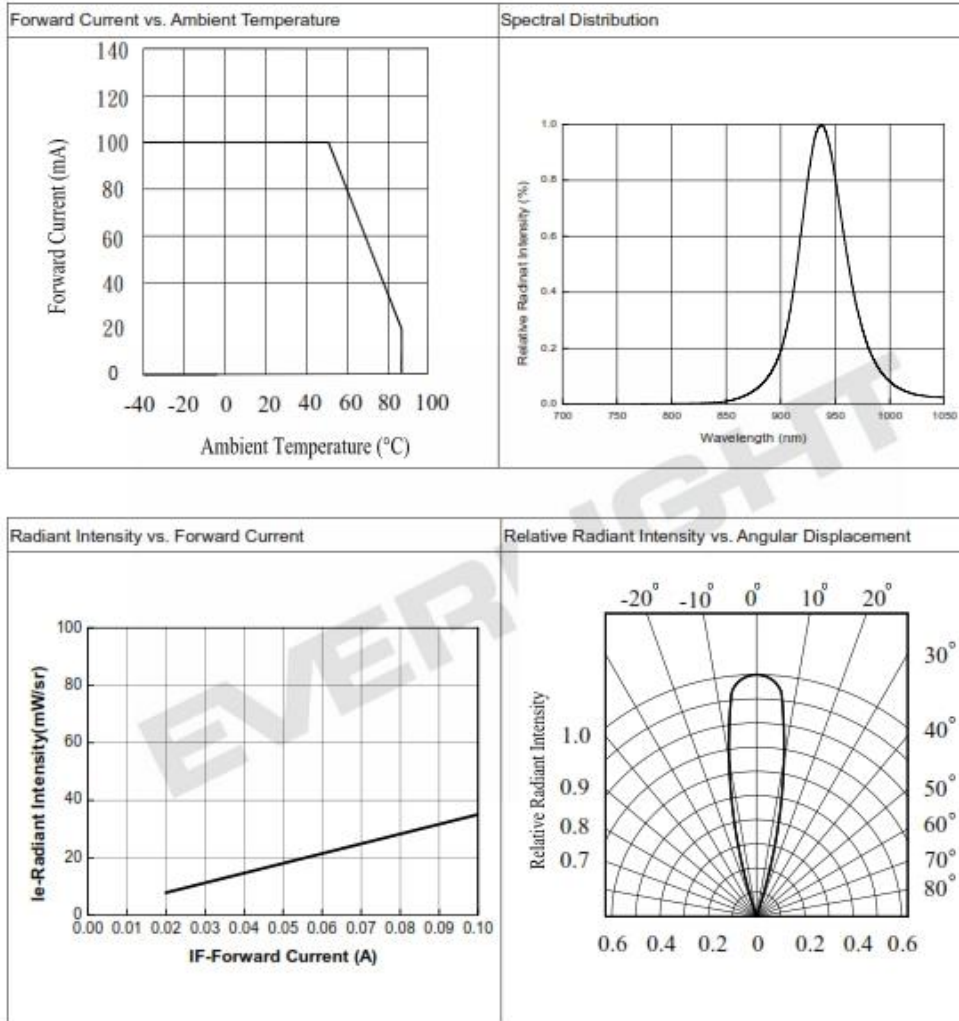
**Rank**

Condition:  $I_f=20mA$

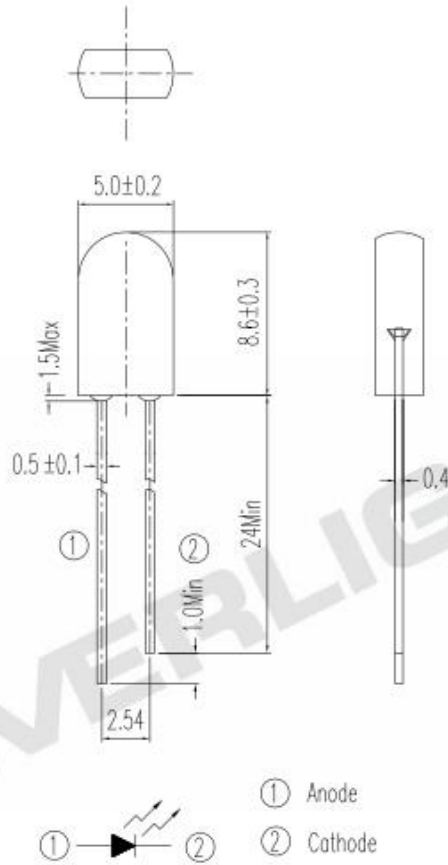
Unit : mW/sr

Bin Number	K	L	M	N	P
Min	4.0	5.6	7.8	11.0	15.0
Max	6.4	8.9	12.5	17.6	24.0

Typical Electro-Optical Characteristics Curves



Package Dimension



Note: Tolerances unless dimensions  $\pm 0.25\text{mm}$

**Label Form Specification**



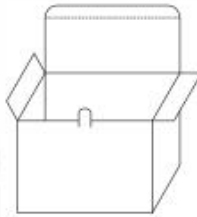
- CPN: Customer's Product Number
- P/N: Product Number
- QTY: Packing Quantity
- CAT: Luminous Intensity Rank
- HUE: Dom. Wavelength Rank
- REF: Forward Voltage Rank
- LOT No: Lot Number
- X: Month
- Reference: Identify Label Number

**Packing Specification**

■ Anti-electrostatic bag



■ Inner Carton



■ Outside Carton



■ Packing Quantity

1. 500 PCS/1 Bag, 6 Bags/1 Inner Carton
2. 10 Inner Cartons/1 Outside Carton

**Notes**

1. Lead Forming
  - During lead formation, the leads should be bent at a point at least 3mm from the base of the epoxy bulb.
  - Lead forming should be done before soldering.
  - Avoid stressing the LED package during leads forming. The stress to the base may damage the LED's characteristics or it may break the LEDs.
  - Cut the LED lead frames at room temperature. Cutting the lead frames at high temperatures may cause failure of the LEDs.
  - When mounting the LEDs onto a PCB, the PCB holes must be aligned exactly with the lead position of the LED. If the LEDs are mounted with stress at the leads, it causes deterioration of the epoxy resin and this will degrade the LEDs.

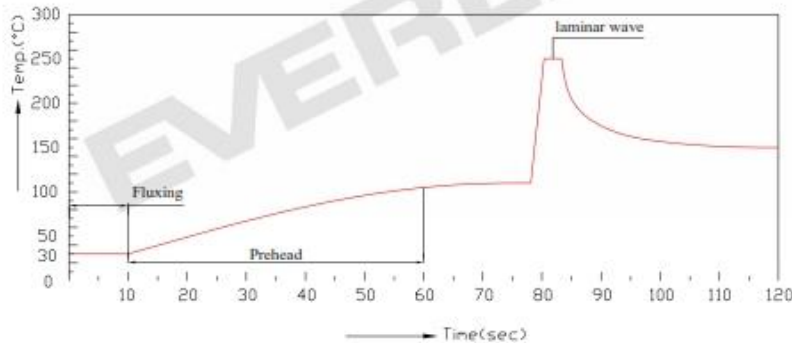
2. Storage
  - The LEDs should be stored at 30°C or less and 70%RH or less after being shipped from Everlight and the storage life limits are 3 months. If the LEDs are stored for 3 months or more, they can be stored for a year in a sealed container with a nitrogen atmosphere and moisture absorbent material.
  - Please avoid rapid transitions in ambient temperature, especially, in high humidity environments where condensation can occur.

3. Soldering
  - Careful attention should be paid during soldering. When soldering, leave more than 3mm from solder joint to epoxy bulb, and soldering beyond the base of the tie bar is recommended.

- Recommended soldering conditions:

Hand Soldering		DIP Soldering	
Temp. at tip of iron	300°C Max. (30W Max.)	Preheat temp.	100°C Max. (60 sec Max.)
Soldering time	3 sec Max.	Bath temp. & time	260 Max., 5 sec Max
Distance	3mm Min.(From solder joint to epoxy bulb)	Distance	3mm Min. (From solder joint to epoxy bulb)

- Recommended soldering profile

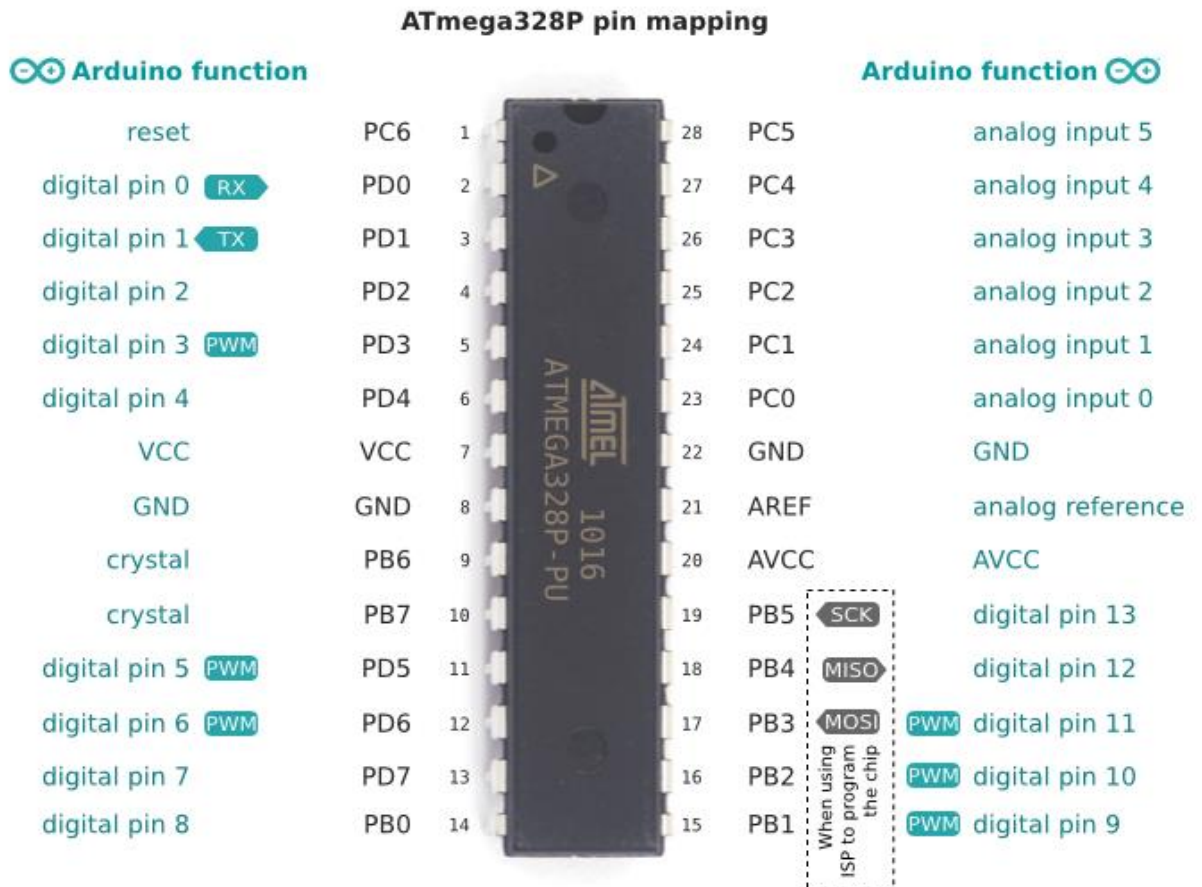


- Avoiding applying any stress to the lead frame while the LEDs are at high temperature particularly when soldering.
- Dip and hand soldering should not be done more than one time
- After soldering the LEDs, the epoxy bulb should be protected from mechanical shock or vibration until the LEDs return to room temperature.
- A rapid-rate process is not recommended for cooling the LEDs down from the peak temperature.

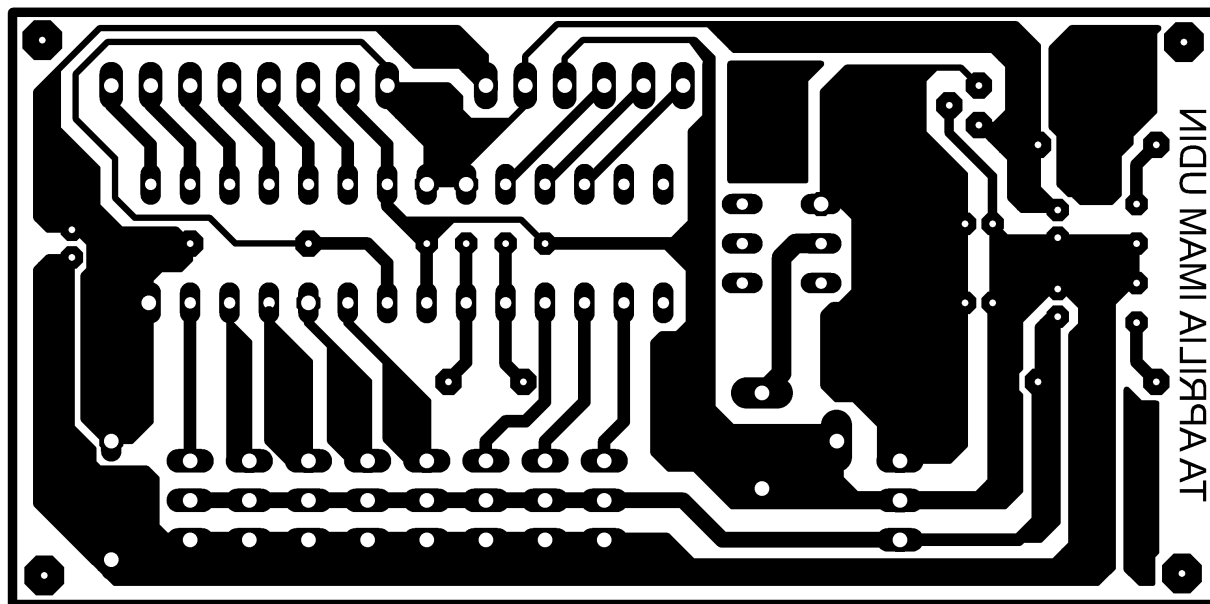
Although the recommended soldering conditions are specified in the above table, dip or hand soldering at the lowest possible temperature is desirable for the LEDs.

- Wave soldering parameter must be set and maintain according to recommended temperature and dwell time in the solder wave.
4. Cleaning
- When necessary, cleaning should occur only with isopropyl alcohol at room temperature for a duration of no more than one minute. Dry at room temperature before use.
  - Do not clean the LEDs by the ultrasonic. When it is absolutely necessary, the influence of ultrasonic cleaning on the LEDs depends on factors such as ultrasonic power and the assembled condition. Ultrasonic cleaning shall be pre-qualified to ensure this will not cause damage to the LED
5. Heat Management
- Heat management of LEDs must be taken into consideration during the design stage of LED application. The current should be de-rated appropriately by referring to the de-rating curve found in each product specification.
  - The temperature surrounding the LED in the application should be controlled. Please refer to the data sheet de-rating curve.
6. ESD (Electrostatic Discharge)
- Electrostatic discharge (ESD) or surge current (EOS) can damage LEDs.
  - An ESD wrist strap, ESD shoe strap or antistatic gloves must be worn whenever handling LEDs.
  - All devices, equipment and machinery must be properly grounded.
  - Use ion blower to neutralize the static charge which might have built up on surface of the LEDs plastic lens as a result of friction between LEDs during storage and handing.
7. Other
- Above specification may be changed without notice. EVERLIGHT will reserve authority on material change for above specification.
  - When using this product, please observe the absolute maximum ratings and the instructions for using outlined in these specification sheets. EVERLIGHT assumes no responsibility for any damage resulting from use of the product which does not comply with the absolute maximum ratings and the instructions included in these specification sheets.
  - These specification sheets include materials protected under copyright of EVERLIGHT corporation. Please don't reproduce or cause anyone to reproduce them without EVERLIGHT's consent.

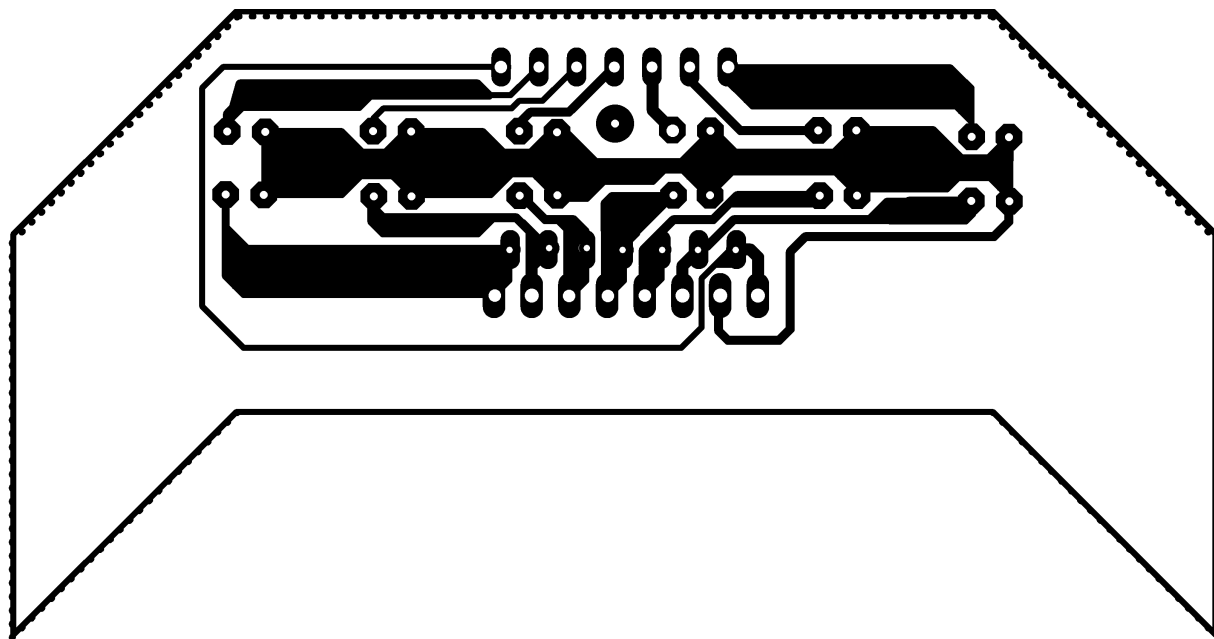
Lampiran 4. ATmega328p Arduino Pin Mapping



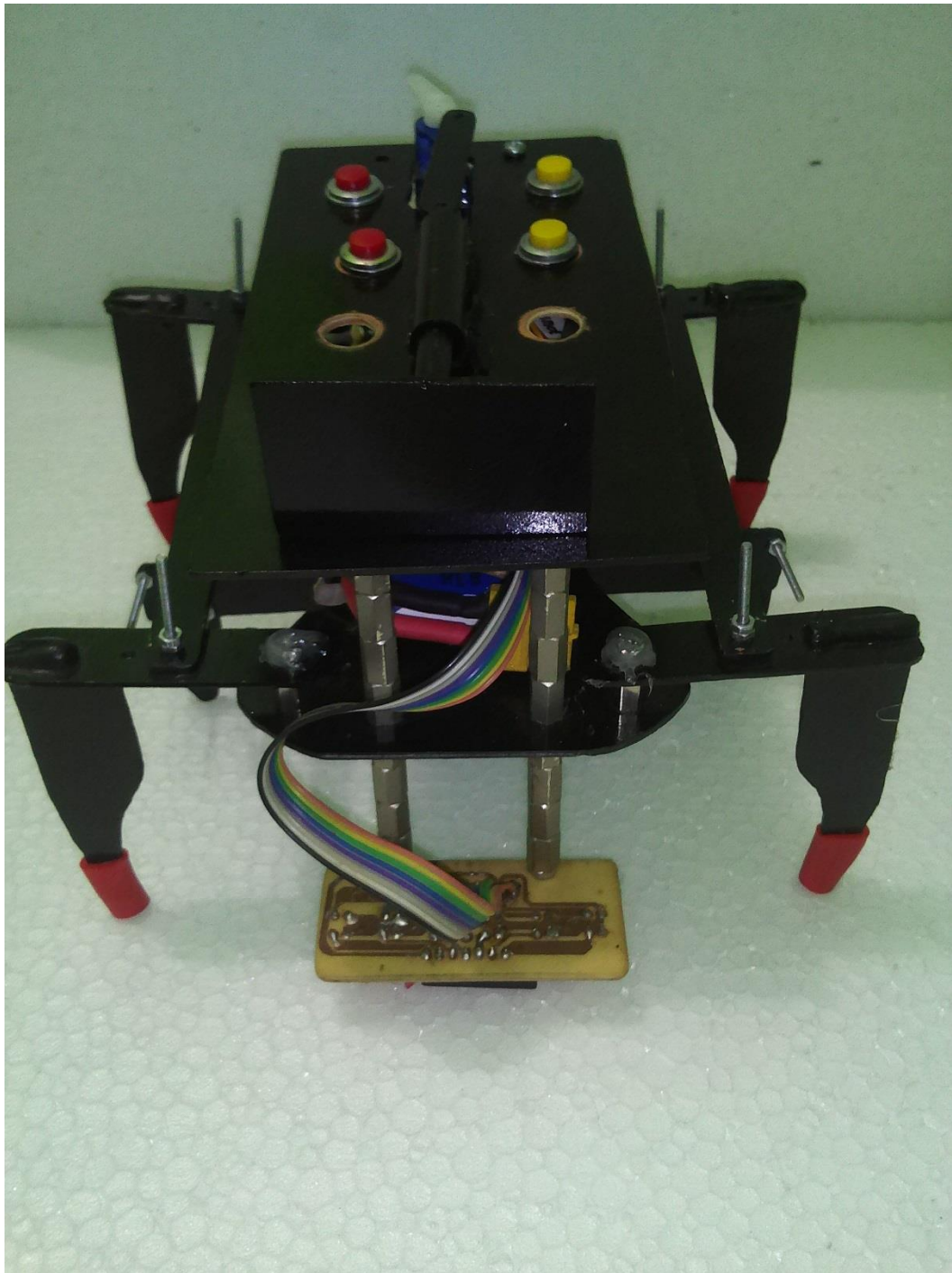
Lampiran 5. Desain PCB Mikrokontroler ATmega328



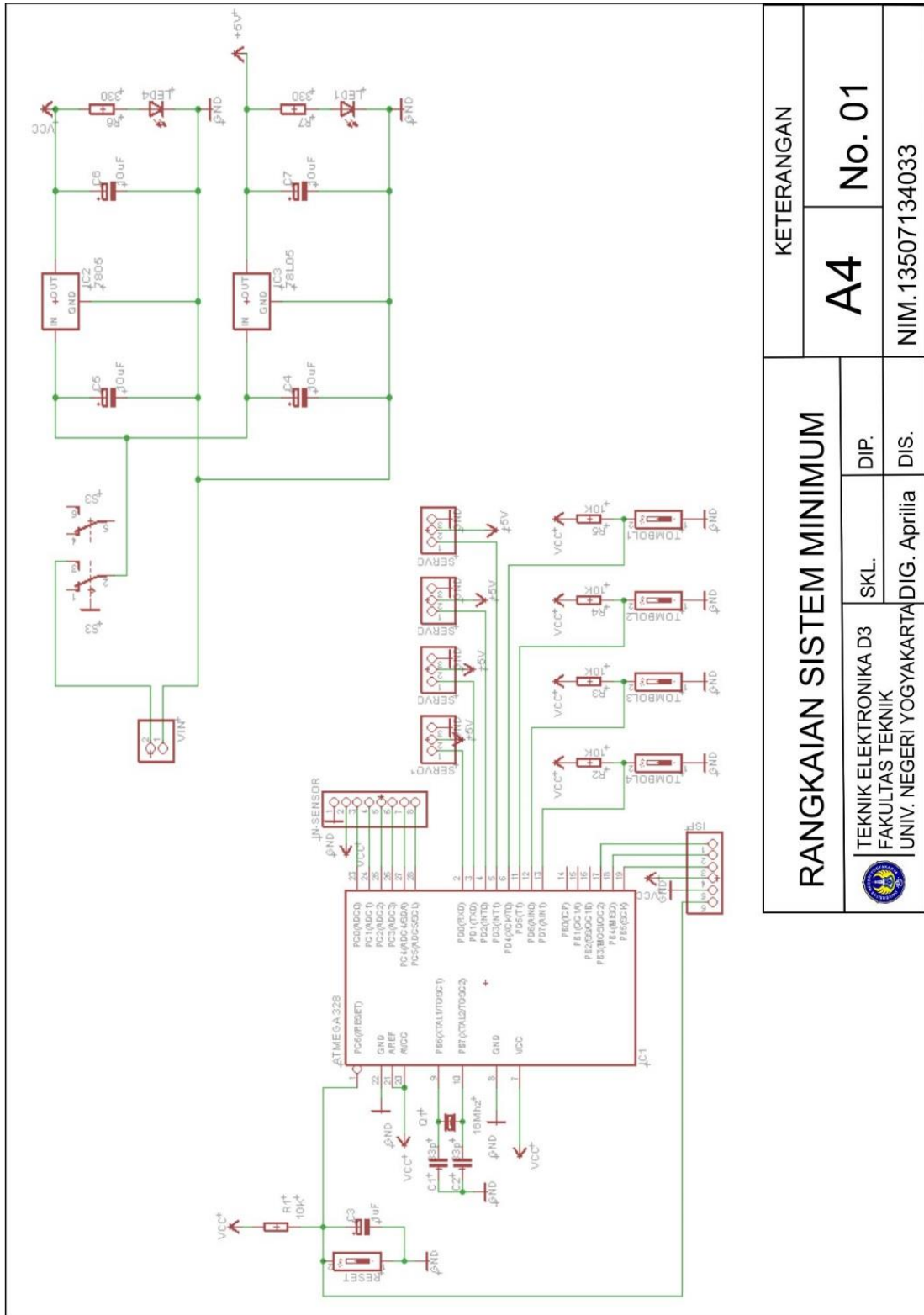
Lampiran 6. Desain PCB Motor Sensor



Lampiran 7. *Prototipe* Robot Line Follower Berkaki



Lampiran 8. Rangkaian System Minimum Robot Pengantar Makaknan



RANGKAIAN SISTEM MINIMUM		KETERANGAN	
		A4	No. 01
TEKNIK ELEKTRONIKA D3		SKL.	DIP.
FAKULTAS TEKNIK		UNIV. NEGERI YOGYAKARTA	
DIG. Aprilia		DIS.	
		NIM. 13507134033	

## Lampiran 9. Program

```
/*
```

```
Prototipe Robot Line Follower Berkaki Sebagai Pengantar Makanan  
Di Restoran Berbasis Mikrokontroler ATmega328
```

```
*/
```

```
#include <Servo.h>
```

```
Servo myservo1;
```

```
Servo myservo2;
```

```
Servo myservo3;
```

```
Servo myservo4;
```

```
int kecepatan;
```

```
int ulang;
```

```
int sensor1v;
```

```
int sensor2v;
```

```
int sensor3v;
```

```
int sensor4v;
```

```
int sensor5v;
```

```
int sensor6v;
```

```
int tombol1;
```

```
int tombol2;
```

```
int tombol3;
```

```
int tombol4;
```

```

int meja = 0;
int persimpanganke = 1;
int allhitam = 1;

int memory;
int intensitas;

const int sensor1 = A0;
const int sensor2 = A1;
const int sensor3 = A2;
const int sensor4 = A3;
const int sensor5 = A4;
const int sensor6 = A5;

const int pintombol1 = 4;
const int pintombol2 = 11;
const int pintombol3 = 6;
const int pintombol4 = 7;

void setup()
{
myservo1.attach(0); // servo kiri
myservo2.attach(1); // servo kanan
myservo3.attach(2); // servo tengah
myservo4.attach(3); // servo pendorong makanan

```

```

myservo1.write(100); //servo kiri tengah +maju
myservo2.write(100); //servo kanan tengah +mundur
myservo3.write(92); //servo tengah tengah +angkat kanan
myservo4.write(10);
delay(1000);

pinMode(pintombol1, INPUT_PULLUP);
pinMode(pintombol2, INPUT_PULLUP);
pinMode(pintombol3, INPUT_PULLUP);
pinMode(pintombol4, INPUT_PULLUP);

kecepatan = 130;
intensitas = 700;
ulang = 0;
memory=0;

selebrasi();
}

//=====
//=====

// PROGRAM UTAMA

//=====
//=====

void loop()
{
    tombol1 = digitalRead(pintombol1); // bagian ini merupakan program untuk
    tombol

```

```
tombol2 = digitalRead(pintombol2);
tombol3 = digitalRead(pintombol3);
tombol4 = digitalRead(pintombol4);
```

```
if (tombol1 == LOW) // bagian ini merupakan bagian eksekusi jika
ada tombol yang ditekan
```

```
{
    meja = 1;
    persimpanganke = 1;
}
```

```
else if (tombol2 == LOW)
```

```
{
    meja = 2;
    persimpanganke = 1;
}
```

```
else if (tombol3 == LOW)
```

```
{
    meja = 3;
    persimpanganke = 2;
}
```

```
else if (tombol4 == LOW)
```

```
{
    meja = 4;
    persimpanganke = 2;
}
```

```
if (meja >= 1)
```

```
{
```

```

        jalan();
    }
}

//=====
//=====

// SUB PROGRAM Jalan

//=====
//=====

void jalan()
{
    sensor1v = analogRead(sensor1); // bagian ini merupakan program untuk
    membaca nilai yang dibaca oleh sensor

    sensor2v = analogRead(sensor2);
    sensor3v = analogRead(sensor3);
    sensor4v = analogRead(sensor4);
    sensor5v = analogRead(sensor5);
    sensor6v = analogRead(sensor6);

    if (sensor1v>intensitas & sensor2v<intensitas & sensor3v<intensitas &
    sensor4v<intensitas & sensor5v<intensitas & sensor6v<intensitas) // bagian ini
    adalah eksekusi pada sensor
    {
        kiri();
        memory=3;
    }

    else if (sensor1v>intensitas & sensor2v>intensitas & sensor3v>intensitas &
    sensor4v>intensitas & sensor5v>intensitas & sensor6v>intensitas)
    {

```

```

    persimpangan();
}
else if (sensor1v>intensitas & sensor2v>intensitas & sensor3v<intensitas &
sensor4v<intensitas & sensor5v<intensitas & sensor6v<intensitas)
{
    kiri();
    memory=3;
}
else if (sensor1v<intensitas & sensor2v>intensitas & sensor3v>intensitas &
sensor4v<intensitas & sensor5v<intensitas & sensor6v<intensitas)
{
    kiri();
    memory=3;
}
else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v>intensitas &
sensor4v>intensitas & sensor5v<intensitas & sensor6v<intensitas)
{
    maju();
}
else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v<intensitas &
sensor4v>intensitas & sensor5v>intensitas & sensor6v>intensitas)
{
    kanan();
    memory=2;
}
else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v<intensitas &
sensor4v<intensitas & sensor5v>intensitas & sensor6v>intensitas)
{
    kanan();
    memory=2;
}

```

```

}

else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v<intensitas &
sensor4v<intensitas & sensor5v>intensitas & sensor6v>intensitas)

{
    kanan();
    memory=2;
}

else if (sensor1v>intensitas & sensor2v<intensitas & sensor3v<intensitas &
sensor4v<intensitas & sensor5v<intensitas & sensor6v<intensitas)

{
    kiri();
    memory=3;
}

else if (sensor1v<intensitas & sensor2v>intensitas & sensor3v<intensitas &
sensor4v<intensitas & sensor5v<intensitas & sensor6v<intensitas)

{
    kiri();
    memory=3;
}

else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v<intensitas &
sensor4v<intensitas & sensor5v<intensitas & sensor6v<intensitas)

{
    if (memory==2)
    {
        kanan();
    }
    else if (memory==3)
    {
        kiri();
    }
}

```

```

    }
    else if (memory==1)
    {
        maju();
    }
}

else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v<intensitas &
sensor4v>intensitas & sensor5v<intensitas & sensor6v<intensitas)
{
    maju();
}

else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v<intensitas &
sensor4v<intensitas & sensor5v>intensitas & sensor6v<intensitas)
{
    kanan();
    memory=2;
}

else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v<intensitas &
sensor4v<intensitas & sensor5v<intensitas & sensor6v>intensitas)
{
    kanan();
    memory=2;
}

else if (sensor1v>intensitas & sensor2v>intensitas & sensor3v>intensitas &
sensor4v<intensitas & sensor5v<intensitas & sensor6v<intensitas)
{
    kiri();
    memory=3;
}

```

```

else if (sensor1v<intensitas & sensor2v>intensitas & sensor3v>intensitas &
sensor4v>intensitas & sensor5v<intensitas & sensor6v<intensitas)
{
    maju();
}

else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v>intensitas &
sensor4v>intensitas & sensor5v>intensitas & sensor6v<intensitas)
{
    maju();
}

else if (sensor1v<intensitas & sensor2v<intensitas & sensor3v<intensitas &
sensor4v>intensitas & sensor5v>intensitas & sensor6v>intensitas)
{
    kanan();
    memory=2;
}

else if (sensor1v>intensitas & sensor2v>intensitas & sensor3v>intensitas &
sensor4v>intensitas & sensor5v>intensitas & sensor6v>intensitas)
{
    persimpangan();
}

else
{
    if (memory==2)
    {
        kanan();
    }
    else if (memory==3)

```

```

    {
        kiri();
    }
    else if (memory==1)
    {
        maju();
    }
}

}

//=====
//=====

//Sub Program Maju
//=====
//=====

void maju()
{
    memory=1;
    for(ulang=0 ; ulang<2 ; ulang++)
    {
        myservo3.write(97);
        delay(kecepatan);
        myservo2.write(110);
        delay(kecepatan);
        myservo3.write(87);
        delay(kecepatan);
        myservo2.write(90);
    }
}

```

```

delay(kecepatan);

myservo3.write(87);
delay(kecepatan);
myservo1.write(90);
delay(kecepatan);
myservo3.write(97);
delay(kecepatan);
myservo1.write(110);
delay(kecepatan);
}
}

//=====
//=====

//Sub Program Mundur
//=====
//=====

void mundur()
{
  for(ulang=0 ; ulang<2 ; ulang++)
  {
    myservo3.write(97);
    delay(kecepatan);
    myservo2.write(90);
    delay(kecepatan);
    myservo3.write(87);
    delay(kecepatan);
  }
}

```

```

myservo2.write(110);
delay(kecepatan);

myservo3.write(87);
delay(kecepatan);
myservo1.write(110);
delay(kecepatan);
myservo3.write(97);
delay(kecepatan);
myservo1.write(90);
delay(kecepatan);
}
}

//=====
=====

//Sub Program Kiri

//=====
=====

void kiri()
{
memory=3;
for(ulang=0 ; ulang<2 ; ulang++)
{
myservo3.write(97);
delay(kecepatan);
myservo2.write(110); //servo kanan
delay(kecepatan);
}
}

```

```

myservo3.write(87);
delay(kecepatan);
myservo2.write(90); //servo kanan
delay(kecepatan);
}
}

//=====
//=====

//Sub Program Kiri Tajam
//=====
//=====

void kiritajam()
{
memory=3;
for(ulang=0 ; ulang<2 ; ulang++)
{
myservo3.write(97);
delay(kecepatan);
myservo2.write(110); //servo kanan
delay(kecepatan);
myservo3.write(87);
delay(kecepatan);
myservo2.write(90); //servo kanan
delay(kecepatan);

myservo3.write(87);
delay(kecepatan);

```

```

myservo1.write(110);
delay(kecepatan);
myservo3.write(97);
delay(kecepatan);
myservo1.write(90);
delay(kecepatan);
}
}

```

```

//=====
=====
//Sub Program Kanan
//=====
=====

```

```

void kanantajam()
{
memory=2;
for(ulang=0 ; ulang<2 ; ulang++)
{
myservo3.write(87);
delay(kecepatan);
myservo1.write(90);
delay(kecepatan);
myservo3.write(97);
delay(kecepatan);
myservo1.write(110);
delay(kecepatan);
}
}

```

```

myservo3.write(97);
delay(kecepatan);
myservo2.write(90);
delay(kecepatan);
myservo3.write(87);
delay(kecepatan);
myservo2.write(110);
delay(kecepatan);
}
}

//=====
=====

//Sub Program Kanan Tajam
//=====
=====

void kanan()
{
memory=2;
for(ulang=0 ; ulang<2 ; ulang++)
{
myservo3.write(87);
delay(kecepatan);
myservo1.write(90);
delay(kecepatan);
myservo3.write(97);
delay(kecepatan);
myservo1.write(110);

```

```
    delay(kecepatan);
  }
}

//=====
//=====

//Sub Program Persimpangan
//=====
//=====

void persimpangan()
{

    allhitam++;

    if (allhitam == 2 & persimpanganke == 1)
    {
        if (meja == 1)
        {
            kiri();
            kiri();
            kiri();
            kiri();
            kiri();
            kiri();
        }
        else if (meja == 2)
        {
            kanan();
        }
    }
}
```

```

        kanan();
        kanan();
        kanan();
        kanan();
        kanan();
    }
}

else if (allhitam == 3 & persimpanganke == 2)
{
    if (meja == 3)
    {
        kiri();
        kiri();
        kiri();
        kiri();
        kiri();
        kiri();
    }
    else if (meja == 4)
    {
        kanan();
        kanan();
        kanan();
        kanan();
        kanan();
        kanan();
    }
}

```

```

}

else if (allhitam == 3 & persimpanganke == 1)
{
  delay(5000);
  myservo4.write(250);
  delay(1000);
  myservo4.write(10);
  delay(1000);

  mundur();
  mundur();
  mundur();
  mundur();
  mundur();

  if (meja == 1)
  {
    kiritajam();
    kiritajam();
    kiritajam();
    kiritajam();
    kiritajam();
  }
  else if (meja == 2)
  {
    kanantajam();
    kanantajam();
  }
}

```

```

        kanantajam();
        kanantajam();
        kanantajam();
    }

}

else if (allhitam == 4 & persimpanganke == 1)
{
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    mundur();
    mundur();
    mundur();
    mundur();
    mundur();
    meja=0;
}

```

```

else if (allhitam == 4 & persimpanganke == 2)
{
    delay(5000);
    myservo4.write(250);
    delay(1000);
    myservo4.write(10);
    delay(1000);

    mundur();
    mundur();
    mundur();
    mundur();
    mundur();

    if (meja == 3)
    {
        kiritajam();
        kiritajam();
        kiritajam();
        kiritajam();
        kiritajam();
    }
    else if (meja == 4)
    {
        kanantajam();
        kanantajam();
        kanantajam();
    }
}

```

```

        kanantajam();
        kanantajam();
    }

}

else if (allhitam == 6 & persimpanganke == 2)
{
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    kanantajam();
    mundur();
    mundur();
    mundur();
    mundur();
    mundur();
    meja=0;
}

```

```

else
{
    maju();
    maju();

}

}

//=====
//=====

//Sub Program Selebrasi

//=====
//=====

void selebrasi()
{

    myservo2.write(120);
    myservo1.write(85);
    myservo3.write(90);
    delay(400);

    myservo1.write(120);
    myservo2.write(85);
    myservo3.write(90);
    delay(400);

    myservo2.write(120);
    myservo1.write(85);

```

```
myservo3.write(90);  
delay(400);  
myservo1.write(120);  
myservo2.write(85);  
myservo3.write(90);  
delay(400);
```

```
myservo2.write(100);  
myservo1.write(100);  
myservo3.write(90);  
delay(1000);
```

```
myservo3.write(75);  
delay(200);  
myservo1.write(95);  
delay(100);  
myservo1.write(120);  
delay(100);  
myservo1.write(95);  
delay(100);  
myservo1.write(120);  
delay(100);  
myservo3.write(90);  
delay(500);
```

```
myservo3.write(115);  
delay(200);  
myservo2.write(95);
```

```
delay(100);  
myservo2.write(120);  
delay(100);  
myservo2.write(95);  
delay(100);  
myservo2.write(120);  
delay(100);  
myservo3.write(90);  
delay(500);  
  
myservo2.write(100);  
myservo1.write(100);  
myservo3.write(90);  
delay(1000);  
  
}
```