

## IMPLEMENTASI ALGORITMA MAZE SOLVING PADA ROBOT *LINE FOLLOWER*

<sup>[1]</sup>Mega Nurmalasari, <sup>[2]</sup>Dedi Triyanto, <sup>[3]</sup>Yulrio Brianorman

<sup>[1]</sup><sup>[2]</sup><sup>[3]</sup>Jurusan Sistem Komputer, Fakultas MIPA Universitas Tanjungpura  
Jalan Prof. Dr. H. Hadari Nawawi, Pontianak  
Telp./Fax.: (0561) 577963

e-mail :

<sup>[1]</sup>[nurmala.mega15@gmail.com](mailto:nurmala.mega15@gmail.com), <sup>[2]</sup>[dedi.triyanto@siskom.untan.ac.id](mailto:dedi.triyanto@siskom.untan.ac.id),  
<sup>[3]</sup>[yulrio.brianorman@siskom.untan.ac.id](mailto:yulrio.brianorman@siskom.untan.ac.id)

### ABSTRAK

Algoritma maze solving merupakan algoritma penyelesaian pada maze untuk mendapatkan jalur terpendek dalam penyelesaian sebuah maze. Maze juga dapat diartikan sebuah labirin, yang memiliki struktur yang kompleks dengan serangkaian jalur yang terhubung. Pada penelitian ini digunakan maze berbentuk garis (*line*), dengan dasar permukaan berwarna putih dan garis yang berwarna hitam. Robot *line follower* yang digunakan untuk menyelesaikan maze ini menggunakan AVR ATmega32 sebagai pengendali utama dalam gerakan robot. Mikrokontroler membaca sinyal masukan dari sensor yang terdapat pada robot. Perbedaan logika antara garis hitam dan putih inilah yang akan diolah oleh mikrokontroler untuk menjalankan robot. Pada penyelesaian maze, robot menggunakan aturan *left hand rules*. Robot akan merekam setiap persimpangan yang dilalui dengan memberikan kode berupa L (*left*) belok kiri, S (*straight*) lurus, dan B (*back*) kembali. Dan untuk mendapatkan jalan terpendek robot akan menyederhanakan kode yang telah didapatnya dengan menggunakan algoritma maze solving, sehingga pada start kedua robot tidak perlu menelusuri setiap persimpangan yang ada di maze, robot dapat langsung menemukan dan berjalan sesuai dengan jalur yang telah disederhanakannya. Hasil akhir dari penelitian ini adalah perbedaan jalur yang lebih singkat setelah penyederhanaan, dibandingkan pada awal penelusuran dalam menemukan kotak finish. Hal tersebut menunjukkan bahwa program dapat berjalan dengan baik.

**Kata kunci :** Algoritma maze solving, mikrokontroler, maze, *line follower*

### 1. PENDAHULUAN

Robot *Line follower* merupakan robot yang hanya dapat bergerak dengan mengikuti panduan garis. Garis yang umumnya digunakan adalah garis yang berwarna hitam dengan latar belakang putih dan juga bisa sebaliknya. Perbedaan warna yang kontras ini memudahkan sensor yang ada pada robot *line follower* untuk dapat membaca garis atau lintasan yang harus dilalui [1].

Bagaimana jika robot *line follower* ini diberi studi kasus dengan memberikan lintasan berbentuk maze yang memiliki garis start dan finish?. Apakah robot *line follower* ini dapat menyelesaikan maze tersebut dengan memecahkan maze dari garis start hingga menemukan garis finish? Apakah

robot dapat berjalan menemukan lintasan yang tepat?. Pada penelitian yang sebelumnya dilakukan oleh Indrawan (2008) [2] dengan judul "*Implementasi Metode Simulated Annealing pada Robot Mobil untuk Mencari Rute Terpendek*" pada penelitian ini kecerdasan buatan yang diaplikasikan pada robot mobil menggunakan metode *Simulated Annealing*. Penelitian ini bertujuan untuk mencari rute terpendek dari suatu jalur. Dalam pengaplikasiannya, posisi tujuan, posisi awal, arah dan *annealing schedule* dari robot harus diketahui terlebih dahulu dan mikrokontroler yang digunakan tipe AT89S51. Sedangkan penelitian yang dilakukan oleh Prasetyo (2010) [3] dengan judul "*Sistem Gerak Robot Line follower Menggunakan Motor DC Berbasis Mikro-*

*kontroler ATmega8535 Dengan Sensor Photodiode*". Perbedaan yang ada di penelitian ini adalah penelitian ini memfokuskan pada perancangan dan realisasi sistem gerak robot *line follower* menggunakan motor DC berbasis mikrokontroler Atmega8535 dengan sensor photodiode. Sistem ini merupakan suatu sistem robot bergerak dengan mengikuti garis sesuai dengan kecepatan yang telah ditentukan sesuai dengan kondisi garis.

Dari pertanyaan-pertanyaan tersebut dan melihat penelitian sebelumnya, peneliti ingin membuat robot *line follower* yang dapat menyelesaikan sebuah *maze* untuk mencari rute terpendek dan tepat, dan pada keadaan awal robot, robot tidak mengetahui titik *finish* yang akan dituju, tapi pada saat memetakan *maze*, robot akan menemukan titik *finish*. Dengan menerapkan algoritma *maze solving* robot akan dapat menyelesaikan sebuah lintasan garis berupa *maze*. Maka diperlukan pengembangan robot *line follower* yang dapat menghafal garis atau lintasan.

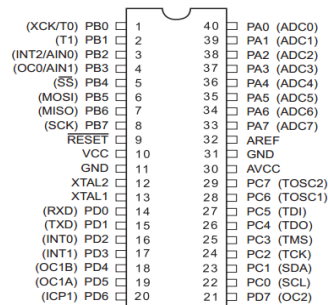
## 2. TEORI PENUNJANG

Pada teori penunjang berisikan beberapa uraian dari fungsi dan kegunaan perangkat keras yang digunakan dan uraian dari teori-teori algoritma dan sistem kendali yang digunakan.

### 2.1 Mikrokontroler AVR ATmega32

Mikrokontroler merupakan sistem pengendali utama pada robot *line follower*. Sebagai masukan dan keluaran dari sistem yang digunakan. Mikrokontroler AVR (Alf and vegard's Rics processor) dari Atmel ini menggunakan arsitektur RISC (Reduced Instruction Set Computer) yang artinya prosesor tersebut memiliki set intruksi program yang lebih sedikit dibandingkan dengan MCS-51 [4].

Mikrokontroler ATmega32 adalah mikrokontroler yang memiliki *clock* dan kinerja yang tinggi hingga 16 Mhz, memiliki 32 KB flash memori untuk menyimpan program, kapasitas SRAM sebesar 2 KB, 32 buah port I/O untuk menghubungkan dengan LCD dan *switch* [5]. Gambar 1 merupakan gambar skematik pin-pin pada mikrokontroler ATmega32.



Gambar 1. Konfigurasi Pin ATmega32

### 2.2 Sensor Garis

Prinsip dasar dari kerja sensor garis ini adalah memantulkan cahaya. Pada sensor garis terdapat dua komponen, sebagai pemancar cahaya dan penerima pantulan cahaya. Sensor garis ini terdiri dari dua komponen yaitu LED dan Photodiode. Cara kerja dari LED (Light Emitting Diode) berfungsi memancarkan cahaya ke garis yang berwarna hitam atau putih. Jika cahaya dipantulkan ke garis hitam maka cahaya akan banyak diserap oleh garis berwarna hitam, dan jika dipantulkan ke bidang berwarna putih maka cahaya akan banyak dipantulkan oleh bidang putih tersebut [1].

Intensitas cahaya yang diserap kembali oleh photodiode berupa tegangan, semakin besar intensitas cahaya yang diserap, semakin besar pula tegangan yang dihasilkan. Tegangan ini berupa sinyal analog. Agar masukkan sinyal analog ini dapat diolah mikrokontroler, sinyal analog ini terlebih dahulu harus diubah ke dalam bentuk sinyal digital dengan bantuan dari ADC (Analog to Digital Converter) pada port A mikrokontroler [1].

### 2.3 Motor Driver dan Motor DC

Driver motor merupakan salah satu perangkat yang digunakan untuk kendali motor DC. Driver motor bertugas mengendalikan arah putaran maupun kecepatan motor DC yang akan dikendalikan. Pada robot *line follower* ini menggunakan driver motor L298. Driver motor L298 memiliki dua buah rangkaian H-Bridge di dalamnya, sehingga dapat digunakan untuk meng-drive dua buah motor DC. Pada aplikasi robotika pergerakan robot beroda umumnya menggunakan motor DC sebagai alat penggerakannya. Motor DC merupakan motor arus searah, perangkat elektromagnetis yang me-

ngubah energi listrik menjadi energi mekanik menjadi tenaga gerak, sehingga tenaga gerak tersebut berupa putaran dari pada motor. Semakin besar tegangan yang diberikan maka putaran poros pada motor DC semakin cepat [6].

#### 2.4 Algoritma Maze Solving

Algoritma *maze solving* adalah algoritma yang digunakan robot untuk memecahkan lintasan yang dibuat untuk mendapatkan jalur yang tepat. *Maze* bisa berupa dinding atau garis. Pada penelitian ini *maze* yang digunakan adalah *maze* garis. Robot yang digunakan berjenis robot *line follower*. Algoritma *maze solving* ini memiliki dua aturan, telusur kiri (*left hand rule*) dan telusur kanan (*right hand rule*) [7].

Hal yang pertama kali dalam menjalankan algoritma ini adalah proses pemetaan (*mapping*) dari *maze* line yang ada. Pada proses pemetaan atau menghafal lintasan di setiap persimpangan yang dilewatinya robot akan memberi kode, kode tersebut disimpan dalam memori robot. Adapun kode yang diberikan adalah sebagai berikut:

1. L (*left*) berbelok ke kiri karena melewati persimpangan 3 ke kiri,
2. S (*straight*) jalan terus jika melewati persimpangan dengan pilihan lurus dan belok kanan,
3. B (*back*) jika robot berjumpa pada jalan buntu maka robot yang dilakukan robot adalah berbalik arah menuju ke persimpangan terakhir yang dilaluinya.

Setelah mencapai garis *finish* dan semua kode tersimpan dan tersusun maka robot dapat menyederhanakan lintasan untuk kembali menuju garis *start* dari garis *finish* tanpa harus melewati lintasan yang memiliki persimpangan dan lintasan terputus.

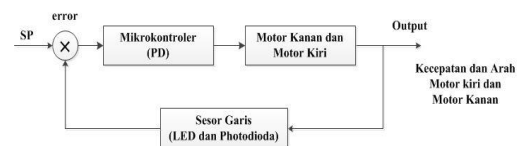
#### 2.4 Sistem Kendali

Pengontrol proses adalah komponen-komponen sistem kontrol yang pada dasarnya memiliki sebuah masukan berupa sinyal *error*, yaitu selisih antara sinyal dengan nilai yang diinginkan dan sinyal umpan balik, serta keluaran berupa sinyal untuk memodifikasi keluaran sistem[8].

Sistem kontrol lup tertutup selalu merujuk kepada sistem yang menggunakan

umpan balik untuk mengurangi *error* sistem. Dalam sistem kontrol lingkaran tertutup, nilai keluaran berpengaruh langsung terhadap aksi pengaturan. Sinyal selisih (*error*) yaitu perbedaan antara masukan acuan dan sinyal umpan balik diberikan kepada kontroler sedemikian sehingga dalam prosesnya memperkecil selisih dan menghasilkan keluaran sistem pada harga atau kondisi yang diinginkan [9].

Untuk proses sistem kendali lup tertutup dapat dilihat pada Gambar 2.



Gambar 2. Diagram blok sistem kendali lup tertutup

### 3. METODOLOGI PENELITIAN

Pada penelitian ini terdapat beberapa tahap yang dilakukan yaitu pada studi lapangan melakukan pengamatan pada robot *line follower* yang pernah digunakan, kemudian studi pustaka mempelajari teori yang terkait dengan penelitian ini, perancangan sistem, *hardware* dan *software*, analisa kebutuhan, integrasi, pengujian terhadap alat yang dibuat, kemudian mengimplementasikan robot pada *maze*. Setelah itu dilakukan analisa terhadap hasil penelitian kemudian ditarik kesimpulan.

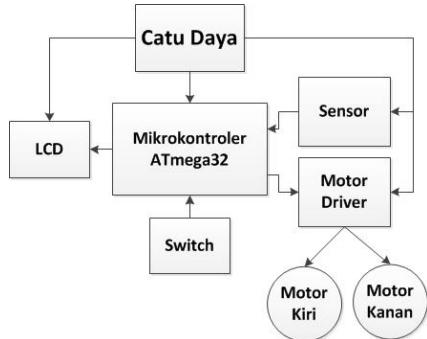
### 4. PERANCANGAN SISTEM

Perancangan sistem terdiri dari dua, yaitu perancangan perangkat keras yang terdiri dari mekanik robot dan skematik rancang bangun robot, dan yang kedua yaitu perancangan perangkat lunak yang terdiri dari pemograman dan algoritma sistem yang digunakan.

#### 4.1 Perancangan Perangkat Keras

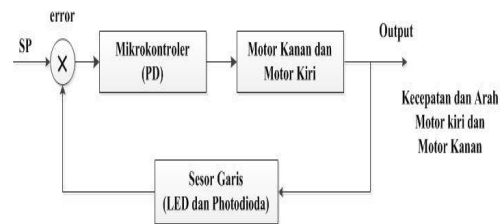
Diagram blok sistem pada Gambar 3 merupakan gambaran keseluruhan dari sistem robot *line follower*. Robot bergerak dengan memperoleh daya dari catu daya. Catu daya memberikan daya pada sensor, mikrokontroler, *driver motor*, switch dan LCD. Sensor kemudian melakukan pembacaan garis, kemudian data yang diteruskan ke mikrokontroler untuk diolah dan diteruskan

ke *driver motor* sebagai penggerak robot, untuk menggerakkan motor kiri dan motor kanan. Mikrokontroler juga memberikan output ke LCD untuk menampilkan menu pengaturan robot.



**Gambar 3.** Diagram Blok Sistem Perangkat Keras

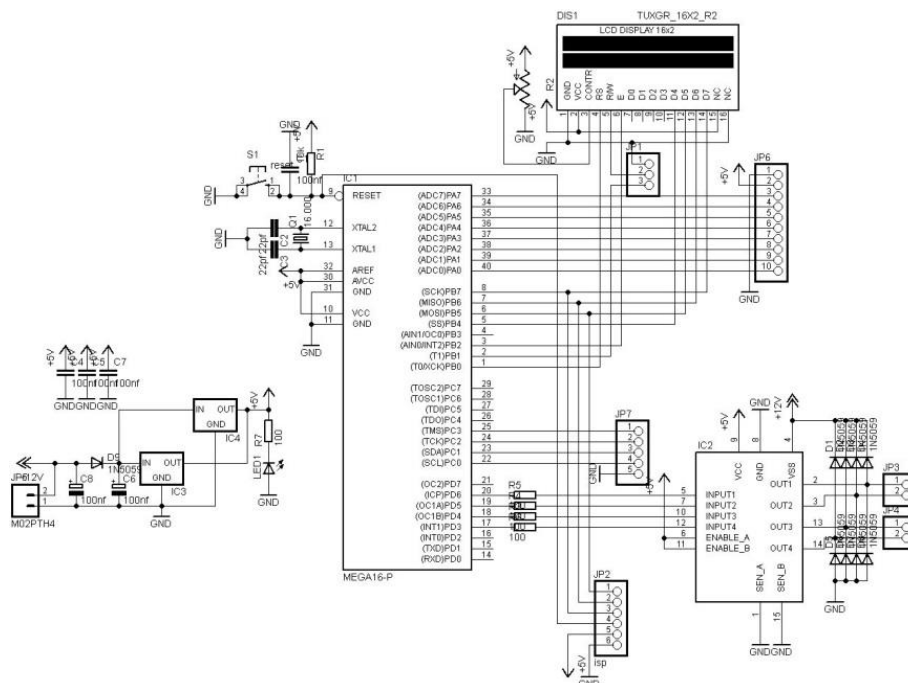
Sistem kendali yang digunakan pada robot *line follower* ini adalah sistem kendali kontinyu yaitu kontroler P-D (Proporsional-Derivatif). Kontrol PD berfungsi sebagai kontrol gerak dari robot, agar robot tetap berada lurus di tengah garis hitam. Pada Gambar 4 dapat dilihat diagram blok dari sistem kendali yang digunakan.



**Gambar 4.** Diagram Blok Perancangan Sistem Kendali Robot *Line follower*

#### 4.1.2 Mikrokontroler

Mikrokontroler yang digunakan merupakan mikrokontroler jenis ATmega32. Pada mikrokontroler ATmega32 terdapat fitur ADC (Analog to Digital Converter) yang memiliki fungsi untuk mengkonversi sinyal analog berupa tegangan listrik menjadi sinyal digital yang kemudian diteruskan oleh mikrokontroler ke motor DC agar dapat menggerakkan robot. Mikrokontroler ATmega32 memiliki 4 buah port I/O sebagai komunikasi, port A, port B, port C dan port D. Konfigurasi dari port mikrokontroler dapat dilihat pada Gambar 5. Berikut adalah konfigurasi pin mikrokontroler yang digunakan.



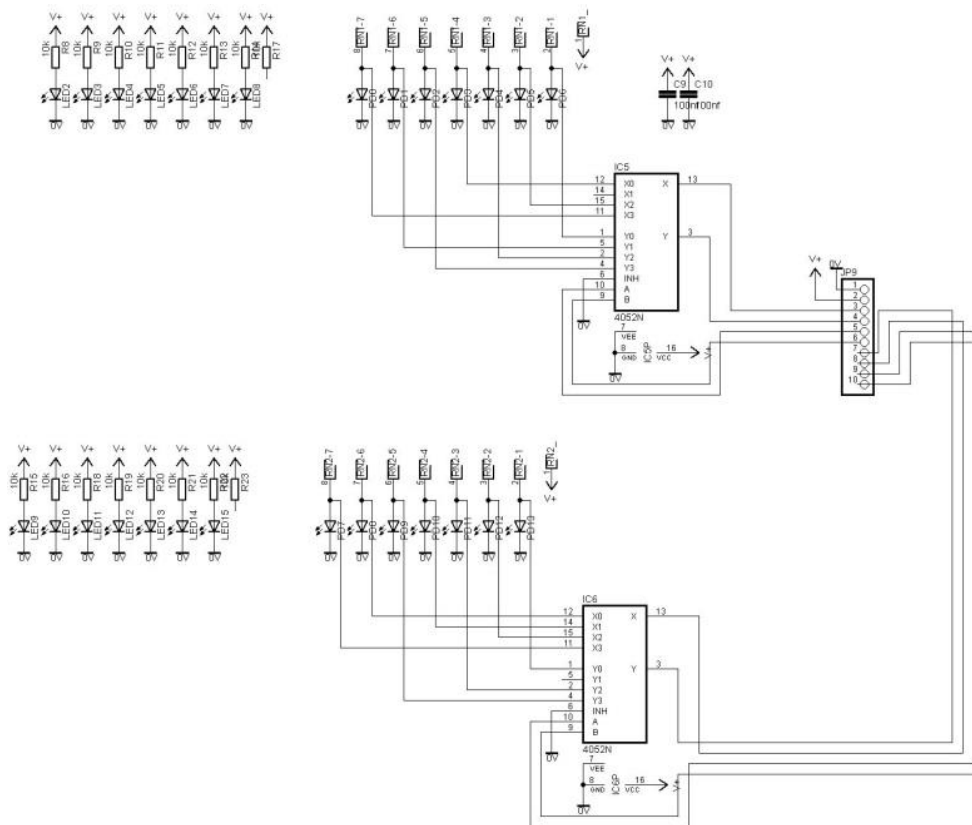
**Gambar 5.** Rangkaian Skematik Minimum Sistem

- a. Port A  
 Port A digunakan sebagai port masukan dari sensor garis dengan menambahkan dua buah multiplekser untuk menambahkan jumlah port pada port A. Di port A terdapat fitur ADC. Pin yang digunakan pada port A sebanyak 4 buah pin. Pin A0, A1, A4, dan A5.
- b. Port B  
 Port B digunakan sebagai port keluaran dari LCD. Pin yang digunakan untuk keluaran port B ini dari pin B4, B5, B6, dan B7.
- c. Port C  
 Port C digunakan sebagai port masukan dari *switch* yang berfungsi sebagai pengaturan pada robot. Pin yang digunakan pada port C ini dari pin C0, C1, C2, dan C3. Port D

- d. Port D  
 Port D digunakan untuk keluran dari *driver motor* yang mengatur motor DC. Pin D6 untuk mengatur arah dari motor kanan. Pin D3 untuk mengatur arah dari motor kiri. Pin D4 dan Pin D5 digunakan untuk mengatur kecepatan motor.

**4.1.3 Sensor Garis**

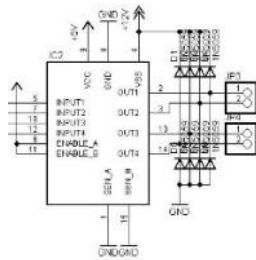
Sensor garis yang digunakan sebanyak 14 pasang sensor garis. Penempatan sensor garis pada robot dapat dilihat pada Gambar 6. Penempatan sensor dibagi menjadi lima bagian, sensor bagian depan, sensor bagian belakang, sensor bagian tengah, sensor bagian kanan dan bagian kiri dari robot. Sensor bagian depan berfungsi sebagai mendeteksi garis yang akan dilewati. Sensor bagian kanan dan kiri robot berfungsi mendeteksi garis persimpangan dan sensor bagian belakang berfungsi sebagai pembaca garis agar pada saat robot bergerak, robot tetap berada ditengah garis.



**Gambar 6.** Rangkaian Skematik Sensor Garis

#### 4.1.4 Driver Motor dan Motor Dc

Driver motor dan motor DC merupakan rangkaian penggerak dari robot *line follower*. Pada rancangan robot *line follower* ini menggunakan dua buah motor DC 5 Volt sebagai penggerak dan driver motor yang bertugas sebagai mengendalikan arah putaran maupun kecepatan roda robot. Driver motor yang digunakan adalah IC L298. Pada driver motor L298 untuk output motor DC digunakan dioda, hal ini ditujukan agar driver motor dapat menahan arus balik yang datang dari motor DC. Masukan tegangan driver motor berasal dari mikrokontroler utama. ENABLE A untuk mengatur kecepatan motor 1 dan ENABLE B untuk mengatur kecepatan motor 2. Pin D3 sebagai pengatur arah kiri dan Pin D6 sebagai pengatur arah kanan. Pada Gambar 7 dapat dilihat rangkaian skematik dari motor driver yang digunakan.



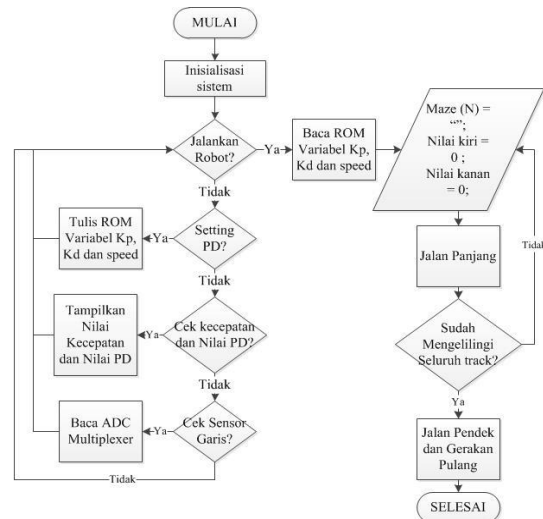
Gambar 7. Rangkaian Skematik Driver Motor

#### 4.1.5 Power Supply

Power supply merupakan sumber tegangan yang diberikan kepada seluruh komponen pada robot *line follower*. Menggunakan baterai Lipo 11,1 Volt dengan tambahan IC LM 7805 untuk menghasilkan tegangan +5 volt dan juga sebagai penyeimbang tegangan dari catu daya bila terjadi perubahan tegangan.

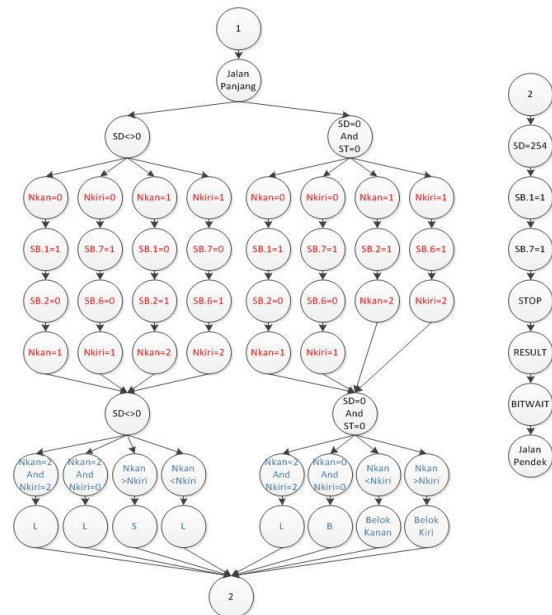
#### 4.2 Perancangan Perangkat Lunak

Diagram alir pada Gambar 8 menunjukkan alir kerja mikrokontroler yang digunakan dalam sistem robot *line follower*. Hal yang pertama kali dilakukan oleh mikrokontroler adalah inialisasi sistem awal program dengan melakukan booting program, booting 1, 2 dan 3.



Gambar 8. Diagram Alir Perangkat Lunak

Kemudian masuk ke program jalan panjang pada Gambar 9 yang merupakan program dari pemetaan maze dan kemudian masuk program jalan pendek dikerjakan setelah menyederhanakan pemetaan pada program jalan panjang pada *start* kedua.



Gambar 9. Diagram Pohon Jalur Panjang

Keterangan Gambar :

1. SD : Sensor Depan
2. SB : Sensor Belakang
3. ST : Sensor Tengah
4. Nkan : Nilai Kanan
5. Nkiri : Nilai Kiri

6. SB.1, SB.2, SB.6 dan SB.7 : Penamaan sensor.

Program jalan panjang pada Gambar 9 merupakan algoritma yang digunakan untuk memetakan *maze* yang dilalui. Pada program jalan panjang menggunakan fungsi perulangan *if then*. Dan untuk memahami algoritma yang digunakan pada gerakan jalan panjang di bawah ini merupakan pohon logika dari program jalan panjang. Program jalan panjang ini terdapat tiga kondisi yaitu,  $SD < 0$ ,  $SD = 0$  and  $ST < 0$  dan kondisi ketiga  $SD = 254$  yang berakhir pada jalan pendek.

Algoritma pada proses jalan panjang adalah sebagai berikut :

1. Bagian tulisan berwarna merah merupakan proses pembacaan keadaan

nilai dari sensor pada saat memetakan *maze*.

2. Beberapa nilai yang dihasilkan pada saat pembacaan garis berupa nilai 0, 1, dan 2.
3. Nilai 0 pada hasil pembacaan sensor bagian belakang merupakan keadaan dimana sensor tersebut tidak bertemu garis hitam.
4. Nilai 1 pada hasil pembacaan sensor bagian belakang merupakan keadaan dimana sensor tersebut bertemu garis hitam.
5. Nilai 0 pada hasil pembacaan nilai kiri dan nilai kanan merupakan nilai pembandingan dalam menentukan keadaan pada sensor kiri dan sensor kanan robot jika robot tidak bertemu garis hitam dan tidak berbelok.

**Tabel 1.** Aturan Pemrosesan Peta

No	Maze (n-2)/x	Maze (n-1)y	Maze (n-2)x	Hasil Simpan Maze	Bentuk Persimpangan	Bentuk Penyederhanaan
1	S	B	L	R		
2	L	B	L	S		
3	L	B	S	R		
4	R	B	L	B		
5	L	B	R	B		
6	S	B	S	B		

6. Nilai 1 pada hasil pembacaan nilai kiri dan nilai kanan merupakan nilai pembandingan dalam menentukan keadaan pada sensor kiri dan sensor kanan robot jika robot bertemu garis hitam dan berbelok
7. Nilai 2 pada hasil pembacaan nilai kiri dan nilai kanan merupakan nilai pembandingan dalam menentukan keadaan pada sensor kiri dan sensor kanan robot jika robot bertemu garis hitam tetapi robot belum pasti berbelok
8. Bagian tulisan berwarna biru merupakan hasil yang didapat dari pemetaan maze dinyatakan dalam bentuk L (left), S (straight) dan B (back).
9. Belok kiri dan Belok kanan pada bagian biru di keadaan  $SD=0$  and  $ST=0$  merupakan sub program untuk mengatur putaran motor pada saat robot melakukan gerakan belok kiri dan belok kanan.

Pada saat memetakan *maze* robot menyimpan kode yang didapat pada setiap persimpangan, kemudian dari kode tersebut robot melakukan penyederhanaan. Pada Tabel 1 dapat dilihat aturan pemrosesan peta.

Keterangan Tabel :

1. L : (Left) Belok kiri
2. R : (Right) Belok kanan
3. S : (Straight) Lurus
4. B : (Back) Kembali jika jalan buntu

### 5. Pengujian dan Analisis

Setelah melakukan perancangan pada perangkat keras dan perangkat lunak pada sistem robot follower, maka selanjutnya adalah pengujian dan analisis yang dikerjakan.

#### 5.1.1 Pengujian Perangkat Keras Sensor Garis

Pengujian sensor garis bertujuan untuk mengetahui apakah sensor dapat bekerja dengan baik dengan melihat perbedaan nilai tegangan pada sensor garis. Masing-masing dari ke 14 photodiode dan LED akan diukur nilai tegangannya, pada saat sensor berada pada garis hitam dan pada permukaan putih. Pada permukaan berwarna hitam, sensor menghasilkan logika 1. Dan pada permukaan berwarna putih sensor menghasilkan logika 0.

Pada Tabel 2 dan Tabel 3 dapat dilihat hasil dari pengukuran besarnya arus dan tegangan pada led dan photodiode.

**Tabel 2.** Pengukuran Tegangan Pada LED

No	Tegangan (V)	
	Gelap	Terang
1	2.27	2.27
2	2.94	2.94
3	2.81	2.81
4	2.74	2.74
5	2.76	2.76
6	2.76	2.76
7	2.75	2.75
8	2.93	2.93
9	2.76	2.76
10	2.63	2.63
11	2.63	2.63
12	2.77	2.77
13	2.96	2.96
14	2.78	2.78

**Tabel 3.** Pengukuran Tegangan Pada Photodiode

No	Tegangan (V)	
	Gelap	Terang
1	4.63	0.13
2	4.06	0.13
3	3.85	0.11
4	4.12	0.11
5	4.63	0.15
6	4.75	0.14
7	4.65	0.26
8	3.98	0.11
9	4.41	0.12
10	4.62	0.14
11	4.62	0.14
12	3.91	0.11
13	4.32	0.15
14	3.45	0.12

#### 5.1.2 Pengujian Pada Sub Program

Tahapan yang dilakukan untuk pengujian program adalah sebagai berikut,

- a. Mengkompilasikan program basic yang telah dibuat kedalam file ekstensi hex dengan menggunakan BASCOM AVR.
- b. Menghubungkan mikrokontroler dengan komputer menggunakan USB AVR ISP *downloader* kemudian mengupload program yang telah dikompilasikan ke



mikrokontroler.

- c. Menghubungkan sensor garis, motor DC, *switch* dan LCD ke mikrokontroler melalui *port* yang telah ditentukan.

Pada percobaan ini port B sebagai output dari LCD dapat bekerja dengan baik. LCD dapat menampilkan menu booting, menu pengaturan kecepatan, kp, dan kd. Port A sebagai masukan pembacaan dari sensor untuk menjalankan fungsi cek sensor dan kalibrasi sensor dapat bekerja dengan baik, LCD juga dapat menampilkan indikator sensor yang bernilai 0 jika bertemu permukaan putih, dan nilai 1 pada LCD jika bertemu garis hitam. Port C sebagai masukan dari *switch*, dapat bekerja dengan baik pada saat memberikan masukan pada pemilihan menu sesuai dengan fungsi yang akan dijalankan.

### 5.1.3 Pengujian Sistem Gerak Robot

Pergerakan pada robot *line follower* terdapat beberapa gerakannya lurus, gerakan belok ke kiri, belok ke kanan, dan berputar 180° pada saat robot menemukan jalan buntu. Untuk melihat apakah robot dapat bergerak sesuai dengan beberapa gerakan yang ada. Di lakukan percobaan sebanyak 3 kali pada setiap gerakan. Tabel 4 merupakan hasil dari percobaan dari sistem gerak robot.

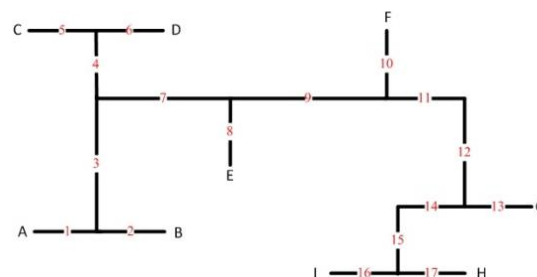
**Tabel 4.** Percobaan Sistem Gerak Robot

Gerakan Robot	Keterangan Percobaan		
	1	2	3
Lurus	Berhasil	Berhasil	Berhasil
Belok Kanan	Berhasil	Berhasil	Berhasil
Belok Kiri	Berhasil	Berhasil	Berhasil
Berputar 180°	Berhasil	Berhasil	Berhasil

### 5.1.4 Pengujian Robot Pada Maze

Pada saat menjalankan robot, posisi baterai pada tegangan penuh 11,1 Volt. Kemudian robot diletakan pada posisi (A, B, C, D, E, F, G, H, I), setelah itu robot menelusuri *maze*, dengan mencatat nomor garis yang dilewati sampai daerah kotak hitam. Kemudian mencatat nomor garis hasil penyederhanaan yang dilakukan oleh robot pada gerakan kedua. Pada Gambar 10

dapat dilihat bentuk *maze* yang dilengkapi dengan huruf sebagai penanda titik mulai dan titik berhenti. Kemudian nomor untuk memberikan penanda pada garis yang telah dilewati oleh robot.



**Gambar 10.** Bentuk Maze Pada Saat Pengujian

Pada Tabel 5 merupakan tabel hasil percobaan pertama robot pada saat bergerak dari titik *start* titik *finish* berupa kotak hitam.

**Tabel 5.** Model Percobaan Pertama

Bentuk Percobaan	Percobaan Ke	Waktu Penelusuran Awal	Garis Penelusuran Awal	Waktu Setelah Penyederhanaan	Garis Setelah Penyederhanaan
Start: A Finish: D	1	00:06.3	1,3,4,5,5,6	00:04.3	1,3,4,6
	2	00:06.2	1,3,4,5,5,6	00:04.0	1,3,4,6
	3	00:06.2	1,3,4,5,5,6	00:04.3	1,3,4,6
	4	00:06.4	1,3,4,5,5,6	00:04.2	1,3,4,6
	5	00:06.3	1,3,4,5,5,6	00:04.4	1,3,4,6

Pada percobaan pertama robot berhasil menelusuri *maze* dan menemukan kotak *finish*. Kemudian pada penelusuran kedua setelah menyederhanakan *maze* yang telah dipetakan robot dapat langsung menuju kotak *finish* tanpa harus melewati persimpangan yang lainnya. Perbedaan jumlah jalur pada saat penelusuran awal dan setelah penyederhanaan yang dilakukan robot menunjukkan jalur yang dilalui oleh robot lebih sedikit.

**Tabel 6.** Model Percobaan Kedua

Bentuk Percobaan	Percobaan Ke	Waktu Penelusuran Awal	Garis Penelusuran Awal	Waktu Setelah Penyederhanaan	Garis Setelah Penyederhanaan
Start: A Finish: F	1	00:12.0	1,3,4,5,6,6,4,7,9,10	00:06.0	1,3,7,9,10
	2	00:12.0	1,3,4,5,6,6,4,7,9,10	00:06.1	1,3,7,9,10
	3	00:11.9	1,3,4,5,6,6,4,7,9,10	00:06.0	1,3,7,9,10
	4	00:11.9	1,3,4,5,6,6,4,7,9,10	00:06.1	1,3,7,9,10
	5	00:12.2	1,3,4,5,6,6,4,7,9,10	00:06.2	1,3,7,9,10

Hasil dari percobaan kedua pada Tabel 6 sama dengan hasil dari percobaan pertama, jalur yang dilewati robot setelah penyeder-

hanaan maze lebih sedikit dibandingkan dengan penelusuran awal.

**Tabel 7.** Model Percobaan Ketiga

Bentuk Percobaan	Per Coba-an Ke	Waktu Penelusuran Awal	Garis Penelusuran Awal	Waktu Setelah Penyederhanaan	Garis Setelah Penyederhanaan
Start: C Finish: G	1	00:11.5	5,6,6,4,7,9,10,10,11,12,13	00:06.9	5,4,7,9,11,12,13
	2	00:11.6	5,6,6,4,7,9,10,10,11,12,13	00:07.1	5,4,7,9,11,12,13
	3	00:11.6	5,6,6,4,7,9,10,10,11,12,13	00:07.3	5,4,7,9,11,12,13
	4	00:11.7	5,6,6,4,7,9,10,10,11,12,13	00:07.3	5,4,7,9,11,12,13
	5	00:11.7	5,6,6,4,7,9,10,10,11,12,13	00:07.1	5,4,7,9,11,12,13

Hasil dari percobaan ketiga pada Tabel 7 sama dengan hasil dari percobaan pertama dan kedua, jalur yang dilewati robot setelah penyederhanaan maze lebih sedikit dibandingkan dengan penelusuran awal.

## 6. Kesimpulan dan Saran

Setelah melakukan pengujian dan analisis dari beberapa pengujian yang dilakukan maka dapat ditarik beberapa kesimpulan serta saran untuk penelitian selanjutnya.

### 6.1 Kesimpulan

Setelah melakukan pengujian dan analisis, dapat ditarik beberapa kesimpulan yaitu:

1. Pengujian robot dalam memetakan *maze* dan menyederhanaan *maze*. Algoritma *maze solving* yang diterapkan dapat bekerja dengan baik. Percobaan pertama pada penelusuran awal robot melewati 6 garis, setelah dilakukan penyederhanaan robot hanya melewati 4 garis. Kemudian pada percobaan kedua robot melewati 10 garis, setelah dilakukan penyederhanaan robot hanya melewati 5 garis. Kemudian pada percobaan ketiga robot melewati 11 garis, setelah dilakukan penyederhanaan robot hanya melewati 7 garis. Perbedaan jumlah garis yang lebih sedikit pada saat pada saat awal penelusuran dan setelah penyederhanaan menunjukkan program dapat berjalan dengan baik.
2. Pengujian pada Tabel 2 dan Tabel 3, menunjukkan bahwa ke 14 sensor garis dapat bekerja dengan baik. Perbedaan tegangan pada garis hitam dan permukaan putih yang dihasilkan photodiode menunjukkan bahwa sensor garis

dapat membedakan antara garis hitam dan permukaan putih.

3. Kendala yang terjadi pada saat pengujian robot pada maze, permukaan maze yang tidak rata dan kotor menyebabkan sensor mengalami kesalahan dalam pembacaan garis. Dan pada saat *power supply* mengalami penurunan daya, robot sedikit kehilangan kendali terhadap gerak, dan sensor tidak bisa membaca garis hitam secara tepat.
4. Kendala yang terjadi pada sensor garis pada saat dilakukan pengujian, penempatan photodiode dan LED harus benar dan pas, sehingga pada saat sensor membaca garis hitam, cahaya yang dipantulkan oleh LED cukup menyinari permukaan garis sehingga cahaya dapat ditangkap kembali oleh photodiode.
5. Pada pengujian beberapa sub program, program dapat bekerja dengan baik, dan LCD dapat menampilkan beberapa sub program yang sedang dijalankan.
6. Sistem gerak robot yaitu *driver* motor dan motor DC dapat bekerja dengan baik, pada saat berjalan luruk, belok, dan memutar. Kendala yang terjadi pada saat pengujian dilakukan jika robot dihidupkan dalam jangka waktu yang lama, IC regulator yang digunakan menjadi panas, dan kecepatan putaran pada motor DC berkurang.

### 6.2 Saran

Pada penelitian ini robot dan algoritma yang diterapkan dapat bekerja dengan baik dan benar. Tetapi penelitian ini masih memerlukan pengembangan lebih lanjut. Dengan merancang mekanik dan program robot yang lebih baik, tanpa terpengaruhi oleh daya dari *power supply*. Merancang mekanik robot dalam peletakan photodiode dan LED yang baik sehingga pada saat pengujian jika photodiode dan LED terbentur sesuatu, bentuk penempatan sensor garis tidak berubah. Penggunaan lintasan *maze* yang lebih baik sehingga tidak mempengaruhi kerja sensor garis dalam membaca garis. Menambahkan jenis sensor yang digunakan atau menambahkan fungsi lain pada robot, seperti menambahkan fungsi lengan untuk mengambil barang, atau pe-

nambahan kamera sebagai pengenalan suatu objek.

#### DAFTAR PUSTAKA

- [1] Suyadhi, D. S. (2008). *Build Your Own Line Followee Robot*. Yogyakarta: Penerbit Andi.
- [2] Indrawan, D. (2008). Implementasi Metode Simulated Annealing pada Robot Mobil untuk Mencari Rute Terpendek.
- [3] Prasetyo, E. (2010). Sistem Gerak Robot *Line follower* Menggunakan Motor DC berbasis Mikrokontroler ATmega8535 Dengan Sensor Photodiode.
- [4] Iswanto. (2008). *Design dan Implementasi Sistem Embedded Mikrokontroler ATmega8535 dengan Bahasa Basic*. Yogyakarta: Penerbit Gava Media.
- [5] Winoto, A. (2008). *Mikrokontroler AVR ATmega8/32/16/8535 dan Programannya dengan Bahasa C pada WinAVR*. Bandung: Penerbit Informatika.
- [6] Bishop, O. (2010). *Dasar-Dasar Elektronika*. Jakarta: Penerbit Erlangga.
- [7] Vannoy II, R. T. (2009). Design a Line Maze Solving Robot.
- [8] W. Bolton. ( 2006). *Sistem Instrumentasi dan Sistem Kontrol*. Jakarta: Penerbit Erlangga.
- [9] Siswoyo. (2008). *Teknik Listrik Industri Jilid 2 untuk SMK*. Jakarta: Direktorat Pembinaan Sekolah Menengah Kejuruan Direktorat Jenderal Manajemen Pendidikan Dasar dan Menengah Departemen Pendidikan Nasional.