



***MONITORING ROBOT WALL-FOLLOWER
MENGUNAKAN BLUETOOTH***

Skripsi

Diajukan sebagai salah satu persyaratan untuk memperoleh gelar Sarjana
Pendidikan Program Studi Pendidikan Teknik Elektro

Oleh

Taufik Hidayat NIM. 5301411049

UNNES

JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS NEGERI SEMARANG

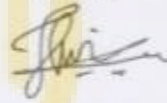
2016

PERNYATAAN

Dengan ini saya menyatakan bahwa semua yang tertulis dalam skripsi yang berjudul "*Monitoring Robot Wall-Follower Menggunakan Bluetooth*" ini adalah benar-benar hasil karya sendiri, bukan plagiat dari hasil karya orang lain. Pendapat atau temuan orang lain yang terdapat dalam skripsi ini dikutip dan dirujuk berdasarkan kode etik karya ilmiah.

Semarang, 28 Juli 2016

Penulis,



Taufik Hidayat

NIM. 5301411049

UNNES
UNIVERSITAS NEGERI SEMARANG

HALAMAN PENGESAHAN

Skripsi dengan judul "*Monitoring Robot Wall-Follower Menggunakan Bluetooth*" telah dipertahankan di depan sidang Panitia Ujian Skripsi Fakultas Teknik UNNES pada tanggal 26 Oktober 2016

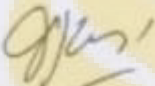
oleh


Nama : Taufik Hidayat
NIM : 5301411049
Program Studi : S-1 Pendidikan Teknik Elektro

Panitia:

Ketua Panitia

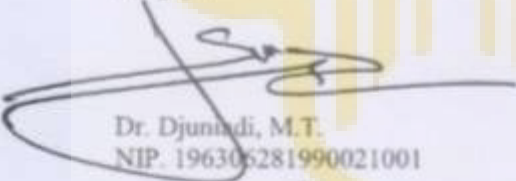
Sekretaris

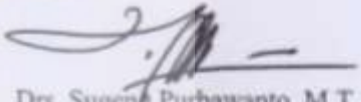

Dr.-Ing. Dhidik Prastiyanto, S.T., M.T.
NIP. 197805312005011002


Drs. Agus Suryanto, M.T.
NIP. 196708181992031004

Penguji I

Penguji II


Dr. Djunardi, M.T.
NIP. 196306281990021001


Drs. Sugeng Purbawanto, M.T.
NIP. 195703281984031001

Penguji III/ Pembimbing


Tatyantoro Andrasto, S.T., M.T.
NIP. 19680316199031001

UNNES
UNIVERSITAS NEGERI SEMARANG

Mengetahui,

Dekan Fakultas Teknik UNNES




Nur Qudus, M.T.

NIP. 196911301994031001

MOTTO DAN PERSEMBAHAN

Motto :

- Awali segala sesuatu dengan “Bismillah” dan lakukan dengan ikhlas
- Yang paling penting bukan seberapa besar cita-cita yang ingin kita raih, tetapi seberapa besar usaha yang kita lakukan untuk meraih cita-cita yang kita inginkan.

Dipersembahkan Untuk :

- Orang tua saya tercinta, Bapak dan Ibu
- TeteH Siti Khodijah dan saudara-saudariku yang telah memberikan dukungan baik do’a maupun materi
- Pengelola Beasiswa Bidikmisi UNNES dan Direktorat Jenderal Perguruan Tinggi
- Sahabat-sahabatku (Jurusan Teknik Elektro, UKM RIPTeK dan Kost Samurai) yang selalu memberikan bantuan, dukungan dan motivasi

UNNES
UNIVERSITAS NEGERI SEMARANG

ABSTRAK

Taufik Hidayat. 2016. *Monitoring Robot Wall-Follower Menggunakan Bluetooth*. Pembimbing Tatyantoro Andrasto. Program Studi S-1 Pendidikan Teknik Elektro Fakultas Teknik Universitas Negeri Semarang.

Robot *wall-follower* merupakan *basic* dari robot pemadam api yang bergerak menelusuri dinding pada suatu trek/lapangan yang berupa ruangan-ruangan dengan misi memadamkan api dan dapat kembali ke titik *start* (*homing*) dengan cepat. Salah satu metode yang dapat menyelesaikan misi tersebut yaitu *simple wall-maze solving*. Akan tetapi, dalam prakteknya tidak jarang robot tetap mengalami *error* atau tersesat dalam suatu ruangan yang bukan merupakan titik tujuan utama robot. Dengan permasalahan tersebut, dapat dilakukan penelitian monitoring robot *wall-follower* melalui *Personal Computer* (PC)/Laptop menggunakan *Bluetooth* yang bertujuan untuk mengetahui pergerakan dan metode robot yang sedang berjalan.

Pada penelitian ini menggunakan metode *Research and Development* (R&D). Prosedur penelitian dilakukan dengan melakukan perancangan, penerapan metode *simple wall-maze solving* pada robot *wall-follower*, pengujian sistem komunikasi antara robot dengan PC/Laptop, pengujian sensor jarak (HC-SR04), *photodiode*, dan motor dc.

Hasil penelitian menunjukkan bahwa tingkat ketelitian sensor jarak HC-SR04 sebesar 98,45% dan sensor *photodiode* dapat membedakan warna hitam, biru dan putih. Metode *simple wall-maze solving* dapat berjalan sesuai dengan misi robot. Sistem komunikasi mikrokontroler AVR ATmega32 menggunakan *Bluetooth* mempunyai *range* data 8bit, sehingga data yang digunakan berupa kode ASCII. Untuk sistem komunikasi yang lebih sempurna, maka harus lebih memahami karakteristik dari fitur USART.

Kata Kunci : Robot Wall-Follower, Simple Wall-Maze Solving, Sistem Komunikasi dan Bluetooth

KATA PENGANTAR

Puji syukur penulis ucapkan kepada Allah SWT yang telah memberikan rahmat taufik dan hidayah-Nya sehingga penulis dapat menyelesaikan penelitian skripsi yang berjudul “*Monitoring Robot Wall-Follower menggunakan Bluetooth*”.

Pada penyusunan skripsi ini tak lepas dari dukungan dan dorongan dari beberapa pihak yang telah membantu. Oleh karena itu, penulis mengucapkan banyak terima kasih kepada :

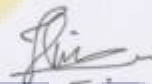
1. Bapak Prof. Dr. Fathur Rokhman, M.Hum selaku Rektor Universitas Negeri Semarang yang telah memberikan kesempatan kepada penulis untuk menempuh pendidikan di Universitas Negeri Semarang
2. Bapak Dr. Nur Qudus, M.T selaku Dekan Fakultas Teknik, Bapak Dr.-Ing. Dhidik Prastiyanto, S.T., M.T selaku Ketua Jurusan Teknik Elektro Universitas Negeri Semarang yang telah memfasilitasi penulis selama menempuh studi.
3. Dra. Dwi Purwanti, Akt.M.S, selaku dosen wali yang telah memberikan pengarahan dan motivasi selama menempuh studi.
4. Bapak Tatyantoro Andrasto, S.T., M.T, selaku Dosen Pembimbing yang selalu membimbing dan memberikan arahan, saran, ilmu dan motivasi selama penyusunan skripsi.
5. Bapak/Ibu dosen Teknik Elektro yang telah memberikan ilmu dan pengalamannya selama menempuh studi.

6. Jurusan Teknik Elektro Universitas Negeri Semarang yang telah memberikan fasilitas untuk tempat penelitian dan pengujian skripsi.
7. Teman-teman Jurusan Teknik Elektro (M.Kamal.W, Mas Nianda A.P, Mas Ugi, Faela Shofa, Syahrin N.A, M.Ridwan, Agung H, dll) yang telah menginspirasi dan memotivasi serta teman-teman yang tidak dapat disebutkan satu-persatu.
8. Pihak-pihak yang telah memberikan fasilitas alat dan bahan guna mendukung penelitian skripsi.
9. Kedua Orang Tua dan segenap keluarga saya yang selalu mendo'akan, memberikan dukungan dan motivasi demi terselesaikannya skripsi ini.

Dalam penyusunan skripsi ini, penulis sangat menyadari adanya kekurangan dan semoga karya tulis ini dapat bermanfaat bagi penulis maupun pembaca.

Semarang, 28 Juli 2016

Penulis,



Taufik Hidayat

NIM. 5301411049



UNNES
UNIVERSITAS NEGERI SEMARANG

DAFTAR ISI

JUDUL	i
PERNYATAAN.....	ii
HALAMAN PENGESAHAN.....	iii
MOTTO DAN PERSEMBAHAN	iv
ABSTRAK	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	3
1.3 Tujuan Penelitian	4
1.4 Manfaat Penelitian	4
1.5 Batasan Masalah.....	4
1.6 Sistematika Penulisan Skripsi	5
BAB II PUSTAKA DAN LANDASAN TEORI.....	7
2.1 Pustaka	7
2.1.1 <i>Maze Solving Algorithm</i>	7
2.1.2 Sistem Komunikasi dan <i>Monitoring</i>	8
2.2 Landasan Teori.....	9
2.2.1 Sistem Robot	9
2.2.2 Sistem Penyimpanan Data	10
2.2.3 Komunikasi Serial USART Mikrokontroler	12
2.2.4 <i>Bluetooth</i> HC06.....	14
2.2.5 ASCII	15
2.2.6 Pemrograman Bahasa C pada CVAVR.....	16

2.2.6.1	Penulisan Program	16
2.2.6.2	Tipe Data	17
2.2.6.3	Perintah Program Kontrol	17
2.2.7	Mikrokontroler AVR ATmega32.....	20
2.2.8	<i>Liquid Crystal Display</i> (LCD)	24
2.2.9	Sensor Jarak HC-SR04.....	25
2.2.10	Sensor Cahaya (<i>Photodiode</i>).....	26
2.2.11	H-Bridge MOSFET	28
2.2.12	PWM (<i>Pulse Width Modulation</i>)	30
2.2.13	Motor DC	34
BAB III	METODE PENELITIAN.....	36
3.1	Tempat Penelitian	36
3.2	Alat dan Bahan	36
3.3	Metode Penelitian	37
3.4	Alur Penelitian	38
3.5	Perancangan Alat Penelitian	39
3.5.1	Perancangan Sistem Mekanik Alat	40
3.5.2	Perancangan <i>Wall-Maze</i> Metode <i>Simple Wall-Maze Solving System</i>	41
3.5.3	Perancangan Sistem Komunikasi dan <i>Monitoring</i> Alat Penelitian.....	44
3.5.4	Perancangan Sistem Kendali <i>Actuator</i> (Pergerakan Robot)	49
3.6	Pengujian dan Pengambilan Data Alat.....	59
BAB IV	PEMBAHASAN.....	61
4.1	Hasil Penelitian	61
4.1.1	Perancangan Robot <i>Wall-Follower</i>	61
4.1.2	Perancangan HDI (<i>Human Development Interface</i>)	62
4.1.3	Pengujian Sistem.....	65
4.1.3.1	Pengujian Perancangan Robot <i>Wall-Follower</i>	65
4.1.3.1.1	Pengujian Sensor Jarak HC-SR04.....	65
4.1.3.1.2	Pengujian Sensor Cahaya (<i>Photodiode</i>).....	66

4.1.3.1.3 Pengujian <i>Actuator</i> /Motor (PWM)	67
4.1.3.2 Pengujian Sistem Komunikasi	68
4.1.3.3 Pengujian Sistem <i>Monitoring Robot Wall-Follower</i>	70
4.2 Pembahasan.....	72
4.2.1 Sistem Perancangan Robot.....	72
4.2.1.1 Sensor Jarak HC-SR04.....	72
4.2.1.2 Sensor Cahaya (<i>Photodiode</i>).....	74
4.2.1.3 <i>Actuator</i> /Motor (PWM)	76
4.2.2 Penerapan Metode <i>Simple Wall-Maze Solving</i>	77
4.2.3 Sistem Komunikasi dan <i>Monitoring Robot</i>	80
4.3 <i>Development</i>	83
BAB V PENEUTUP	84
5.1 Kesimpulan	84
5.2 Saran.....	84
DAFTAR PUSTAKA	85
LAMPIRAN.....	87

DAFTAR TABEL

Tabel 2.1 Pengaturan <i>BaudRate</i>	13
Tabel 2.2 Kode ASCII.....	15
Tabel 2.3 Tipe Data.....	17
Tabel 2.4 Fungsi lain PORTB	22
Tabel 2.5 Fungsi lain PORTC.....	23
Tabel 2.6 Fungsi lain PORTD.....	22
Tabel 3.1 Kode Pergerakan Robot (<i>Motion Code</i>).....	42
Tabel 3.2 Instrumen Pengambilan Data Kalibrasi <i>Rute</i> terpendek Robot	43
Tabel 3.3 Instrumen Pengambilan Data Sistem Komunikasi.....	48
Tabel 3.4 Instrumen Pengambilan Data Kalibrasi robot.....	48
Tabel 3.5 Instrumen Pengambilan Data <i>Monitoring Rute</i> Robot (<i>Start-Finish</i>)... 49	
Tabel 3.6 Instrumen Pengambilan Data <i>Monitoring Rute</i> Robot (<i>Finish-Start</i>)... 49	
Tabel 3.7 Instrumen Pengambilan Data Sensor Jarak HC-SR04.....	52
Tabel 3.8 Instrumen Pengambilan Data Sensor Cahaya (<i>photodiode</i>)	54
Tabel 3.9 Instrumen Pengambilan Data PWM	58
Tabel 4.1 Data Pengujian Sensor Jarak HC-SR04.....	65
Tabel 4.2 Data Pengujian Sensor Cahaya (<i>Photodiode</i>)	67
Tabel 4.3 Data Pengujian <i>Actuator</i> /Motor (PWM).....	68
Tabel 4.4 Data Pengujian Sistem Komunikasi.....	69
Tabel 4.5 Data <i>Scanning</i> atau Kalibrasi <i>Rute</i> Robot	70
Tabel 4.6 Uji <i>Monitoring</i> Eksekusi <i>Rute</i> terpendek Robot dari <i>Start-Finish</i>	71

Tabel 4.7 Uji <i>Monitoring</i> Eksekusi <i>Rute</i> terpendek Robot dari <i>Finish-Start</i>	71
Tabel 4.8 Persamaan Sistem Robot.....	83
Tabel 4.9 Perbedaan Sistem Robot	83



DAFTAR GAMBAR

Gambar 2.1 <i>Rute</i> Aturan Tangan Kanan dan Tangan Kiri.....	8
Gambar 2.2 Kontrol robot <i>loop</i> terbuka (<i>open loop</i>)	9
Gambar 2.3 Kontrol robot <i>loop</i> tertutup (<i>close loop</i>)	9
Gambar 2.4 <i>Bluetooth</i> HC06.....	14
Gambar 2.5 Konfigurasi Pin ATmega8535/16/32	21
Gambar 2.6 <i>Liquid Crystal Display</i> (LCD)	24
Gambar 2.7 Sensor Jarak HC-SR04.....	26
Gambar 2.8 Konsep Pemantulan Cahaya dan Ilustrasi Arus Listrik pada Sensor <i>Photodiode</i>	27
Gambar 2.9 Skematik <i>H-Bridge</i> MOSFET.....	28
Gambar 2.10 Bentuk Sinyal PWM.....	30
Gambar 2.11 Motor DC	35
Gambar 3.1 Diagram Alur Penelitian.....	38
Gambar 3.2 Sistem Komunikasi Robot <i>Wall-Follower</i> dengan PC/Laptop	39
Gambar 3.3 Perancangan Sistem Komunikasi Mikrokontroler pada Robot <i>Wall-Follower</i> dengan PC/Laptop	39
Gambar 3.4 Blok Diagram Sistem Komunikasi Robot <i>Wall-Follower</i> dengan <i>Personal Computer</i> (PC)/Laptop	40
Gambar 3.5 Robot <i>Wall-Follower</i>	41
Gambar 3.6 (a) Bentuk Asli Lingkup Gerak (Trek) Robot (b) Bentuk 2 Dimensi Lingkup Gerak (Trek) Robot	43
Gambar 3.7 Skematik Sistem Komunikasi pada Robot <i>Wall-Follower</i>	44

Gambar 3.8 Sistem <i>Wiring</i> Pengujian Sistem Komunikasi	46
Gambar 3.9 Pengaturan USART pada Sistem Minimum ATmega32	47
Gambar 3.10 Blok Diagram Sistem Kendali <i>Actuator</i>	50
Gambar 3.11 <i>Wiring</i> Sensor Jarak HC-SR04 pada Robot <i>Wall-Follower</i>	50
Gambar 3.12 Pengujian Sensor Jarak HC-SR04.....	52
Gambar 3.13 Pengujian Sensor Cahaya (<i>photodiode</i>)	54
Gambar 3.14 (a) Skematik LCD pada Robot <i>Wall-Follower</i> (b) Pengaturan LCD pada <i>Code Wizard CVAVR</i>	55
Gambar 3.15 Skematik <i>H-Bridge</i> dan Pengujiannya	56
Gambar 3.16 Rangkaian <i>Switch Push Button</i>	59
Gambar 4.1 Rancangan Robot <i>Wall-Follower</i>	61
Gambar 4.2 Tampilan <i>Mode</i> Sistem Komunikasi.....	63
Gambar 4.3 Tampilan <i>Mode</i> Sistem <i>Scanning</i>	64
Gambar 4.4 (a) Tampilan <i>Mode</i> Sistem <i>Monitoring</i> Robot <i>Start-Finish</i> (b) Tampilan <i>Mode</i> Sistem <i>Monitoring</i> Robot <i>Finish-Start</i>	64
Gambar 4.5 Grafik Hubungan Data Jarak Mistar dan Data Jarak Sensor.....	66
Gambar 4.6 Pengaturan ADC pada <i>Code Wizard AVR</i>	75
Gambar 4.7 Tampilan Pengaturan PWM pada <i>Code Wizard AVR</i>	77
Gambar 4.8 Trek <i>Rute</i> Robot <i>Start</i> Ruang 1 dan <i>Finish</i> Ruang 2	78
Gambar 4.9 Pengaturan USART pada <i>Code Wizard AVR</i>	80
Gambar 4.10 Bentuk Sinyal Komunikasi Data “0”	82

DAFTAR LAMPIRAN

Lampiran 1 Rangkaian Robot <i>Wall-Follower</i>	87
Lampiran 2 Tabel Pengaturan Register <i>Timer/Counter0</i> dan <i>Timer/Counter2</i>	88
Lampiran 3 Tabel Pengaturan Register <i>Timer/Counter1</i>	89
Lampiran 4 Kode ASCII	92
Lampiran 5 Data Pengujian Sistem Komunikasi USART	93
Lampiran 6 Data <i>Monitoring Rute</i> Robot dan Analisis <i>Simple Wall-Maze Solving</i>	95
Lampiran 7 Data <i>Monitoring</i> Eksekusi <i>Rute</i> Terpendek Robot <i>Start-Finish</i>	108
Lampiran 8 Data <i>Monitoring</i> Eksekusi <i>Rute</i> Terpendek Robot <i>Finish-Start</i>	109
Lampiran 9 Data Sensor HC-SR04 dan Analisis Regresi.....	110
Lampiran 10 Data PWM Motor dan Analisis Pergerakan Robot	115
Lampiran 11 Data Sensor Cahaya (<i>Photodiode</i>) Analisis Warna Bidang	120
Lampiran 12 Sistet <i>Pairing Bluetooth</i> Pada PC/Laptop	121
Lampiran 13 Dokumentasi.....	122
Lampiran 14 Surat Keterangan Penelitian di Laboratorium Teknik Elektro	123
Lampiran 15 Surat Keterangan Uji Kelayakan di Fornext Robotic Semarang...	124
Lampiran 16 Lembar Evaluasi	125
Lampiran 17 Surat Keputusan Dosen Pembimbing	128
Lampiran 18 Surat Keputusan Penguji	129

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Indonesia telah mengadakan beberapa kali Kontes Robot Indonesia (KRI), dan pemenangnya akan menjadi wakil Indonesia untuk mengikuti kontes robot tingkat internasional yang diselenggarakan di beberapa negara. Dalam Kontes Robot Indonesia (KRI) ini terdiri dari berbagai kategori lomba, yang diantaranya yaitu Kontes Robot Abu Indonesia (KRAI), Kontes Robot Pemadam Api Indonesia (KRPAI), Kontes Robot Sepak Bola Indonesia (KRSBI), dan Kontes Robot Seni Indonesia (KRSI) (Kementerian Riset, Teknologi dan Pendidikan Tinggi, 2015:2).

Pada Kontes Robot Pemadam Api Indonesia (KRPAI) ini robot harus dapat menyelesaikan misi dengan cepat dan sesuai dengan aturan. Pada prinsipnya, robot pemadam api ini harus dapat memadamkan api lilin di beberapa titik api di dalam suatu lapangan pertandingan. Lapangan pertandingan ini merupakan miniatur gedung yang berbentuk ruangan-ruangan. Dalam menjalankan misinya, robot pemadam api ini menelusuri dinding-dinding ruangan hingga menemukan ruangan yang terdapat titik api lilin. Sebagai robot penelusur dinding, robot ini sering disebut juga sebagai robot *wall-follower*. Robot yang bergerak berdasarkan sensor jarak, dalam hal ini yaitu sensor jarak HC SR04.

Untuk dapat menyelesaikan misi, robot *wall-follower* harus dapat mencari jalur terpendek dari lapangan pertandingan tersebut. Salah satu metode penyelesaian misi ini dapat menggunakan metode *Simple Wall-Maze Solving*. Dengan meng-adopsi penelitian sebelumnya yang mempunyai spesifikasi robot dan misi yang sama menyatakan bahwa, “Robot dengan metode *Simple Maze* membutuhkan waktu yang lebih cepat yaitu ± 2 detik untuk menyelesaikan tugas, sedangkan metode konvensional waktu yang dibutuhkan ± 13 detik” (Shofa,2015:35).

Secara teknis, penerapan metode *Simple Wall-Maze Solving* pada robot *wall-follower* dan *Simple Line-Maze Solving* pada robot *line-follower* sangat jauh berbeda. Pada robot *line-follower*, sensor garis (*photodiode*) tidak terlalu banyak gangguan (*noise*) oleh lingkungan sekitar. Sedangkan sensor pada robot *wall-follower* (HC-SR04) sangat mudah terpengaruh oleh *noise* lingkungan dan akan mengganggu pergerakan robot, sehingga dibutuhkan *monitoring* terhadap robot tersebut. Dengan adanya sistem *monitoring*, maka pergerakan (eksekusi metode *Simple Wall-Maze Solving*) dapat terlihat secara *real-time*.

Untuk dapat memonitoring robot, maka dibutuhkan sistem komunikasi *wireless* yang dapat menghubungkan antara robot *wall-follower* dengan *Personal Computer* (PC)/Laptop. Dalam hal ini, sistem komunikasi yang digunakan mempunyai tingkat *error* yang kecil ketika robot/*device* komunikasi (*transmitter/receiver*) bergerak. “Dalam sistem komunikasi yang bergerak, perambatan sinyal antara pemancar dan penerima melalui berbagai

lintasan yang berbeda karena adanya fitur-fitur dan topografi lingkungan sekeliling menjadi permukaan-permukaan pemantul” (Purbawanto,2011:36). Salah satu *device* yang dapat mendukung sistem komunikasi dengan tingkat *error* yang kecil ketika robot bergerak dan fleksibilitas yang tinggi yaitu *Bluetooth modul*. *Bluetooth modul* ini merupakan media yang digunakan untuk menyalurkan data dari robot menuju PC/Laptop maupun sebaliknya. Data harus dapat dimengerti oleh sistem baik pengirim (*transmitter*) dan penerima (*receiver*). “Setiap data (informasi) yang diberikan pada masing-masing robot merupakan informasi maupun perintah bagi robot untuk mengerjakan suatu tugas” (Pratama,2014:1).

Dengan melihat uraian di atas, maka dapat tersusun permasalahan penelitian dalam skripsi ini yaitu dengan judul “***Monitoring Robot Wall-Follower menggunakan Bluetooth***”.

1.2 Perumusan Masalah

Dengan pemaparan latar belakang di atas, maka dalam penelitian skripsi ini dapat disimpulkan bahwa terdapat rumusan masalah sebagai berikut:

1. Bagaimana merancang dan membuat robot *wall-follower*?
2. Bagaimana robot *wall-follower* dapat menerapkan metode *Simple Wall-Maze Solving* untuk dapat menyelesaikan misi?
3. Bagaimana robot *wall-follower* dapat berkomunikasi dengan *Personal Computer* (PC)/Laptop melalui *Bluetooth* sehingga robot dapat termonitoring secara *real-time*?

1.3 Tujuan Penelitian

Tujuan dari penelitian skripsi ini adalah :

1. Merancang dan membuat robot *wall-follower*
2. Robot *wall-follower* dapat menerapkan metode *Simple Wall-Maze Solving* untuk dapat menyelesaikan misi robot
3. Robot dapat berkomunikasi dengan *Personal Computer (PC)/Laptop* melalui *Bluetooth* sehingga dapat termonitoring secara *real-time*.

1.4 Manfaat Penelitian

Manfaat yang didapat dari penelitian skripsi ini yaitu :

1. Metode ini dapat digunakan untuk memonitoring robot sehingga dapat mengetahui pergerakan robot.
2. Dapat digunakan untuk memonitoring sistem otomasi yang ada di industri.
3. Sebagai rujukan dalam penelitian dan pengembangan sistem komunikasi data serial.

1.5 Batasan Masalah

Untuk dapat fokus pada pembahasan skripsi dan tidak menyimpang pada tujuannya, maka perlu adanya pembatasan masalah. Ruang lingkup pada skripsi dengan judul “*MONITORING ROBOT WALL-FOLLOWER MENGGUNAKAN BLUETOOTH*” ini hanya terbatas pada perancangan robot *wall-follower* dengan sensor jarak HC SR04 dan dapat menerapkan metode *Simple Wall-Maze Solving* untuk dapat menyelesaikan misi robot. Untuk dapat memonitoring, robot menggunakan modul *Bluetooth* HC06 sebagai media komunikasinya.

1.6 Sistematika Penulisan Skripsi

Sistematika penulisan skripsi merupakan metode penulisan yang digunakan untuk mempermudah pemahaman tentang struktur penulisan dan isi skripsi. Pada sistematika penulisan skripsi ini terdiri dari tiga bagian utama yaitu : bagian pendahuluan, isi dan penutup.

Bagian pendahuluan, pada bagian pertama skripsi ini berisi : halaman judul, halaman pernyataan, persetujuan pembimbing, halaman pengesahan, motto dan persembahan, abstrak, kata pengantar, daftar isi, daftar tabel, daftar gambar dan daftar lampiran.

Bagian isi, pada bagian ini terdiri dari lima bab yaitu :

Bab 1 **Pendahuluan**, pada bab ini berisi tentang latar belakang, perumusan masalah, tujuan, manfaat, batasan masalah dan sistematika penulisan skripsi.

Bab 2 **Pustaka dan Landasan Teori**, bab ini berisi tentang kajian pustaka mengenai landasan teori dari penelitian skripsi ini.

Bab 3 **Metode penelitian**, berisi tentang tempat pelaksanaan diadakannya penelitian, alat dan bahan yang digunakan, perancangan alat, prinsip kerja dari robot *wall-follower*, metode *Simple Wall-Maze Solving*, sistem komunikasi robot menggunakan *Bluetooth* dan metode pengambilan data penelitian.

Bab 4 **Hasil Penelitian dan Pembahasan**, pada bab ini berisi tentang hasil-hasil dari penelitian pengambilan data dan pembahasan dari penelitian.

Bab 5 **Penutup**, bab penutup terdiri dari kesimpulan dan saran-saran dari hasil penelitian yang telah dilakukan.

Bagian Penutup, pada bagian akhir ini berisi daftar pustaka dan lampiran-lampiran.



BAB II

PUSTAKA DAN LANDASAN TEORI

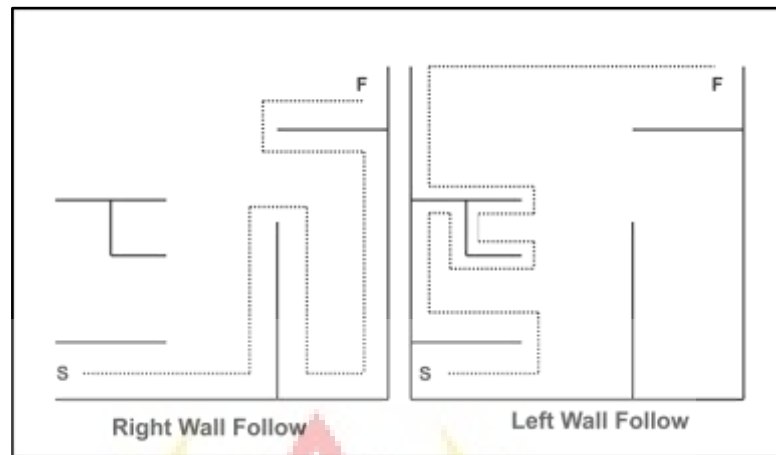
2.1. Pustaka

2.1.1. *Maze Solving Algorithm*

Ada sejumlah algoritma pemecahan *maze* yang berbeda, yaitu *The Random Mouse*, *wall-follower* dan *Pledge*. Algoritma tersebut dirancang untuk digunakan di dalam *maze* oleh *agent* (robot) yang tidak mempunyai pengetahuan sebelumnya dari *maze*.

Algoritma *wall-follower* merupakan algoritma paling mudah dan sering digunakan dalam robot *wall-follower* untuk melintasi suatu *maze*. Algoritma ini sering dikenal dengan aturan tangan kiri (*left-hand rule*) dan aturan tangan kanan (*right-hand rule*). Jika sebuah *maze* seluruhnya saling terhubung maka *agent* (robot) tidak akan pernah tersesat dan pasti menemukan target dengan hanya menggunakan salah satu aturan tangan kiri atau kanan. Tetapi jika ada salah satu *maze* yang tidak terhubung dengan *maze* lainnya maka ada kemungkinan *agent* (robot) dapat tersesat dan tidak menemukan target jika hanya menggunakan salah satu aturan saja (Sartika,2014:11).

Berikut merupakan gambar yang menunjukkan perjalanan robot dengan aturan tangan kiri (*left-hand rule*) dan aturan tangan kanan (*right-hand rule*).



Gambar 2.1 *Rute* Aturan Tangan Kanan dan Tangan Kiri (Sartika,2014:12)

2.1.2. Sistem Komunikasi Data

Komunikasi data adalah pertukaran data antara dua perangkat melalui beberapa bentuk media transmisi seperti kabel kawat. Agar komunikasi data dapat berjalan, perangkat komunikasi harus menjadi bagian dari sistem komunikasi yang terdiri dari kombinasi *hardware* (peralatan fisik) dan perangkat lunak (program). Efektivitas sistem komunikasi data tergantung pada tiga karakteristik mendasar, yaitu: pengiriman (*delivery*), akurasi (*accuracy*), dan Ketepatan waktu (*timeliness*).

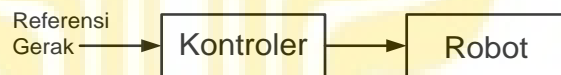
1. Pengiriman (*Delivery*), sistem dapat mengirim data ke tujuan yang benar. Data harus diterima oleh perangkat yang dituju atau pengguna dan hanya perangkat tersebut yang menerimanya.
2. Akurasi (*Accuracy*), sistem dapat mengirim data yang akurat.
3. Ketepatan waktu (*Timeliness*). Sistem harus mengirimkan data pada waktu yang tepat tanpa adanya penundaan waktu yang signifikan. Sistem pengiriman ini biasa disebut transmisi *real-time* (Forouzan,2003:4).

2.2. Landasan Teori

2.2.1. Sistem Robot

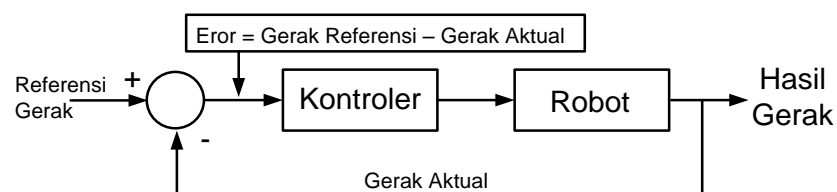
“Teknologi robotika berkembang pesat seiring meningkatnya kebutuhan robot cerdas. Kata robot sudah tidak asing lagi di telinga kita. Kata robot berasal dari bahasa Czeck, *robot* yang berarti ‘bekerja’. Kata robot diperkenalkan oleh Karel Capek saat mementaskan RUR (Rossum’s Universal Robots) pada tahun 1921” (Budiharto,2014:1).

“Sistem robot jika dilihat dari sistem kontrolnya terbagi menjadi dua kelompok, yaitu sistem kontrol *loop* terbuka (*Open Loop*) dan *loop* tertutup (*Close Loop*)” (Pitowarno,2006:22).



Gambar 2.2 Kontrol robot *loop* terbuka (*open loop*)

Kontrol *loop* terbuka atau umpan maju (*feed forward control*) dapat dinyatakan sebagai sistem kontrol yang outputnya tidak diperhitungkan ulang oleh kontroler. Sedangkan pada kontrol *loop* tertutup (*feed beck control*) dapat dinyatakan sebagai system kontrol yang outputnya diperhitungkan ulang oleh kontroler. Seperti pada gambar di bawah.



Gambar 2.3 Kontrol robot *loop* tertutup (*close loop*)

Pada gambar 2.3, jika hasil gerak *actual* telah sama dengan referensi maka *input* kontroler akan nol. Artinya kontroler tidak lagi memberikan sinyal aktual kepada robot karena target akhir perintah gerak telah diperoleh.

2.2.2. Sistem Penyimpanan Data

Data merupakan fakta mengenai suatu objek seperti manusia, benda, peristiwa, konsep, keadaan dan sebagainya yang dapat dicatat. Data dapat dinyatakan dalam bentuk angka, karakter atau simbol.

Data dalam istilah komputer dapat didefinisikan sebagai bahan keterangan tentang kejadian-kejadian nyata atau fakta yang dirumuskan dalam sekelompok lambang tertentu yang tidak acak yang menunjukkan jumlah, tindakan atau hal. Data dapat berupa catatan-catatan dalam kertas, buku, atau tersimpan sebagai *file*. Kumpulan beberapa data atau *file* dalam sistem komputer dapat disebut dengan *database*.

Sistem *database* yang sering digunakan oleh *developer database* salah satunya adalah *database Microsoft office access*. Selain penggunaannya yang mudah sistem *database* ini juga sudah tersedia dalam *Microsoft Office*.

Dalam sistem mikrokontroler ATmega, sistem penyimpanan terletak pada memori internal IC. Memori ATmega terbagi tiga yaitu:

- a. Memori *Flash* adalah memori ROM (*Read Only Memory*) tempat kode-kode program berada. Kata *Flash* menunjukkan jenis ROM yang dapat ditulis dan dihapus secara elektrik.

- b. Memori *Data* adalah memori RAM (*Random Access Memory*) yang digunakan untuk keperluan program.
- c. *EEPROM* (*Electrically Erasable Programmable Read Only Memory*) adalah memori data yang dapat mengendap ketika chip mati (*off*), digunakan untuk keperluan penyimpanan data yang tahan terhadap gangguan catu daya (Winoto,2010:52-53).

Pada sistem robot *wall-follower* ini, EEPROM digunakan untuk menyimpan parameter-parameter yang berupa data yang dibutuhkan dalam menyelesaikan misi tertentu. Penyimpanan data dilakukan ketika robot melakukan kalibrasi baik secara manual maupun otomatis.

Pada pemograman CVAVR, EEPROM digunakan untuk menyimpan parameter-parameter yang diubah dalam bentuk variabel-variabel. Dalam hal ini variabel tersebut bersifat global, sehingga variabel harus dideklarasikan sebagai variabel global.

Sebagai contoh untuk membaca dan menulis data di EEPROM pada pemograman CVAVR dengan ketentuan sebagai berikut:

1. Jika “x” merupakan variabel yang disimpan pada EEPROM dengan tipe data *char*.
2. Jika PINC.2 digunakan sebagai fungsi *input* yang berupa saklar *push button*. Jika ditekan maka akan menambah (*increment*) nilai data pada variabel “x” dan kemudian data disimpan di EEPROM.

3. Jika PINC.3 digunakan sebagai fungsi *input* yang berupa saklar *push button*. Jika ditekan maka akan mengurangi (*decrement*) nilai data yang ada pada variabel “x” dan kemudian disimpan di EEPROM.

Program:

```
#include <mega32.h> //memasukkan library ATmega32 pada program
#include <delay.h> //memasukkan library fungsi tunda pada program
eeprom char x; //pendeclarasian variabel “x” pada EEPROM
void main() {
  PORTC=0xFF; //input pull-up pada PORTC
  DDRC=0x00;} // penggunaan PORTC sebagai input
while( 1){
  If(PINC.2==0) {
    Delay_ms(250); //untuk menghindari efek bouncing
    x++;} //variabel “x” akan ditambahkan dengan 1
  If(PINC.3==0) {
    Delay_ms(250); //untuk menghindari efek bouncing
    x--;} //variabel “x” akan dikurangi dengan 1
}
```

2.2.3. Komunikasi Serial USART Mikrokontroler

“Komunikasi data serial dibedakan menjadi dua macam, yaitu komunikasi data serial sinkron dan komunikasi data asinkron. Perbedaan ini tergantung pada *clock* pendorong data. Dalam komunikasi data serial sinkron, *clock* untuk *shift register* ikut dikirim bersamaan dengan data serial. Sebaliknya dalam komunikasi data serial asinkron, *clock* untuk *shift register* tidak ikut dikirim” (Andrianto,2013:123).

Pada komunikasi USART (*Universal Synchronous and Asynchronous serial Receiver and Transmitter*) mikrokontroler ATmega32 lebih menguntungkan daripada sistem USART, yaitu:

- a. Operasi *full duplex* (mempunyai *register receiver* dan *transmitter* yang terpisah)
- b. Mendukung komunikasi multiprosesor
- c. Mode Kecepatan *transmitter* berorde Mbps

Pada fungsi USART mikrokontroler ATmega32 terdapat beberapa register-register. Register-register tersebut yang perlu ditentukan nilainya dalam menggunakan komunikasi USART diantaranya yaitu UBRR. UBRR (USART *Baud Rate Register*) merupakan *register* 16 bit, yang berfungsi untuk menentukan kecepatan *transmitter* data yang akan digunakan. UBRR ini dibagi menjadi dua yaitu UBRRH dan UBRRL. UBRR [11,...,0] merupakan *bit* penyimpanan konstanta kecepatan komunikasi serial. UBRRH menyimpan 4 *bit* tertinggi, dan UBRRL menyimpan 8 *bit* sisanya. Data yang dimasukkan ke UBRRH dan UBRRL dihitung menggunakan rumus pada tabel di bawah:

Tabel 2.1 Pengaturan *BaudRate*

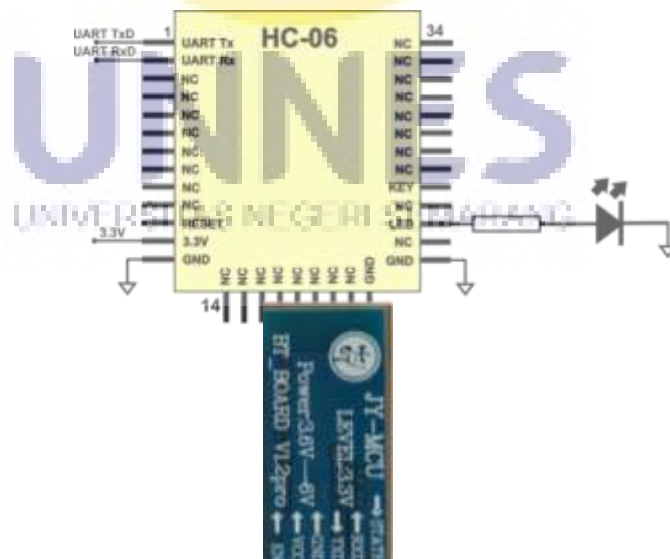
Mode Operasi	Rumus <i>BaudRate</i>	Rumus nilai UBRR
<i>Asynchronous Normal Mode</i> (U2X=0)	$\text{BAUD} = \frac{F_{\text{osc}}}{16 (\text{UBRR} + 1)}$	$\text{UBRR} = \frac{F_{\text{osc}}}{16 \times \text{BAUD}} - 1$
<i>Asynchronous Double Speed Mode</i> (U2X=0)	$\text{BAUD} = \frac{F_{\text{osc}}}{8 (\text{UBRR} + 1)}$	$\text{UBRR} = \frac{F_{\text{osc}}}{8 \times \text{BAUD}} - 1$
<i>Synchronous Master Mode</i>	$\text{BAUD} = \frac{F_{\text{osc}}}{2 (\text{UBRR} + 1)}$	$\text{UBRR} = \frac{F_{\text{osc}}}{2 \times \text{BAUD}} - 1$

(Bejo,2008:59)

2.2.4. Bluetooth HC06

“*Bluetooth* adalah sebuah teknologi komunikasi *wireless* (tanpa kabel) yang beroperasi pada pita frekuensi 2,4 GHz dengan menggunakan sebuah frekuensi *hopping transceiver* yang mampu menyediakan layanan komunikasi data dan suara secara *real-time* antara *host-host Bluetooth* dengan jarak jangkauan yang terbatas (sekitar 10 meter)” (Andrianto,2013:193).

Bluetooth HC06 merupakan *Bluetooth* yang dalam penerimaan dan pengiriman datanya menggunakan komunikasi serial UART. *Bluetooth* HC06 memungkinkan dapat berkomunikasi langsung dengan mikrokontroler melalui jalur TX dan RX yang terdapat pada pin out nya. Pada dasarnya, *Bluetooth* HC06 hanya dapat dikonfigurasi sebagai *slave* tidak bisa digunakan sebagai master. Berikut adalah bentuk fisik dari *Bluetooth* HC06.



Gambar 2.4 *Bluetooth* HC06 (www.electfreaks.com)

Bluetooth HC06 memiliki spesifikasi dalam penggunaannya antara lain:

- a. Nomor Bluetooth : JY-MCU-HC-06, sudah terintegrasi dengan antena
- b. Tegangan Operasional : 5 volt dengan arus 8 ma
- c. Default BaudRate : 9600 bps
- d. Default Pin Pairing : 1234
- e. Kelas : 2, jarak mencapai sekitar 10 meter (Datasheet HC-06:1).

2.2.5. ASCII

“*American Standard Code Information Interchange* (ASCII) adalah kode yang paling umum yang digunakan untuk pengkodean yang berupa karakter. ASCII menggunakan 7 bit pada masing-masing karakter. Sehingga dapat mewakili sampai 128 karakter” (Forouzan, 2003: 861).

Berikut tabel kode ASCII dalam bentuk karakter, biner dan heksadesimal.

Tabel 2.2 Kode ASCII (*Wikipedia* 21 Oktober 2015)



Data					0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b ₄	b ₃	b ₂	b ₁	Column Row	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	\	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	o	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	\$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(8	H	X	h	x
1	0	0	1	9	HT	EM)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	
1	1	0	1	13	CR	GS	-	=	M]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	_	o	DEL

Untuk tabel ASCII yang lebih lengkap dapat melihat pada lembar lampiran.

2.2.6. Pemrograman Bahasa C menggunakan CVAVR

“Bahasa C adalah bahasa yang standart, artinya suatu program yang ditulis dengan versi bahasa C tertentu akan dapat dikompilasi dengan versi bahasa C yang lain dengan sedikit modifikasi” (Jogiyanto,2006:1).

Pada penelitian ini, pemrograman berbasis bahasa C ini digunakan untuk memprogram mikrokontroler ATmega32 sebagai *processing* robot dengan menggunakan *software CodeVisionAVR* (CVAVR).

2.2.6.1. Penulisan Program

Program Bahasa C tidak mengenal aturan penulisan di kolom tertentu. Jadi bias dimulai dari kolom manapun. Namun demikian, untuk mempermudah pembacaan program untuk keperluan dokumentasi, sebaiknya penulisan bahasa C diatur sedemikian rupa sehingga mudah dibaca.

Berikut merupakan contoh struktur penulisan bahasa C pada

CodeVision-AVR.

```

#include <mega32.h>
#define sensor PINA.0
//variable global
unsigned int i, j;
Void main( )
{
//variable local
Char data_rx;
DDRA=0x00;
.....
}
While(1)
{ ..... };
}
    
```

Preprocessor (#) :
 Digunakan untuk memasukan (*include*) *text* dari *file* lain dan mendefinisikan macro yang dapat mengurangi beban kerja pemograman dan meningkatkan *legibility source code* (mudah dibaca)

/*
 ..komentar untuk beberapa baris...
 */
 Atau jika untuk satu baris saja
 // ...komentar...

Inisialisasi

Program Utama
 Program akan berulang terus karena syarat *while* (1) akan selalu menghasilkan nilai benar (*true*)

2.2.6.2. Tipe Data

Tipe data merupakan bagian program yang paling penting karena tipe data mempengaruhi setiap intruksi yang akan dilaksanakan oleh komputer. Berikut ini adalah tipe-tipe data yang ada dalam bahasa C yang dikenali oleh CodeVisionAVR:

Tabel 2.3 Tipe Data

Tipe Data	Ukuran	Jangkauan Nilai
Bit	1 bit	0 atau 1
Char	1 byte	-128 s/d 127
Unsigned char	1 byte	0 s/d 255
Signed char	1 byte	-128 s/d 127
Int	2 byte	-32768 s/d 32767
Short int	2 byte	-32768 s/d 32767
Unsigned int	2 byte	0 s/d 65535
Signed int	2 byte	-32768 s/d 32767
Long int	4 byte	-2147483648 s/d 2147483647
Unsigned long int	4 byte	0 s/d 4294967295

Signed long int	4 byte	-2147483648 s/d 2147483647
Float	4 byte	1.2*10-38 s/d 3.4*10-38
Double	4 byte	1.2*10-38 s/d 3.4*10-38

(Bejo, 2008:85)

2.2.6.3. Perintah Program Kontrol

a. Perintah *if* dan *if ... else ...*

Perintah *if* dan *if... else ...* digunakan untuk melakukan operasi percabangan bersyarat. Fungsi-fungsi untuk menetapkan kondisi dapat dilihat dalam table. Sintaks penulisan *if* dapat ditulis sebagai berikut:

```
If(<expression>) <statement1>;
else<statement2>;
```

b. Perintah percabangan *switch*

Perintah percabangan *if...else ...* dapat digantikan dengan perintah *switch*. Dalam pernyataan *switch*, sebuah variable secara berurutan diuji oleh beberapa konstanta bilangan bulat atau konstanta karakter. Sintaks perintah *switch* dapat ditulis sebagai berikut:

```
Switch(variable)
{
  Case konstanta_1 : statement; break;
  Case konstanta_2 : statement; break;
  Case konstanta_n : statement; break;
  Default; statement;
}
```

c. Perintah *Looping* (pengulangan)

Looping adalah pengulangan satu atau beberapa perintah sampai mencapai keadaan tertentu. Ada tiga perintah *looping*, yaitu:

1. *for*

Untuk pengulangan yang melakukan proses *increment* atau *decrement*. Sintaks perintah *for* dapat dituliskan sebagai berikut:

For

```
(nama_variabel=nilai_awal;syarat_loop;nama_variabel++)
{ statement_yang_diulang; }
```

2. ***while***

Perintah *while* dapat melakukan *looping* apabila persyaratannya benar. Sintaks perintah *while* dapat dituliskan sebagai berikut:

```
nama_variabel = nilai_awal;
while (syarat_loop) {
Statement_yang_akan_diulang;
Nama_variabel++;}
```

3. ***do...while***

Perintah *while* terlebih dahulu melakukan pengujian persyaratan sebelum melakukan *looping*. Kadang-kadang dalam hal ini menimbulkan kerepotan-kerepotan yang tidak perlu, misalnya inisialisasi *variablecontrol*. Salah satu solusi adalah dengan menggunakan *looping do...while...* Berikut adalah sintaks dari perintah *do...while...* :

```
nama_variabel = nilai_awal;
do {
Statement_yang_akan_diulang;
Nama_variabel++; }
while (syarat_loop)
```

d. ***Array***

Array adalah deretan variable yang sejenis dan mempunyai nama yang sama. Setiap anggota deretan (elemen) diberi nomor

yang disebut *indeks*, dimulai dari *indeks nol*. *Array* diatur agar mempunyai lokasi memori yang bersebelahan dengan alamat terkecil menunjuk elemen *array* pertama dan alamat terbesar menunjukkan elemen terakhir. Elemen *array* dapat diakses dengan dengan menunjukkan indeksnya. Berikut deklarasi *array* adalah:

Tipe_data nama_array[ukuran]

Int nilai[100];

Nilai[1]=10;

Nilai[2]=3;

2.2.7. Mikrokontroler AVR ATmega32

“Mikrokontroler adalah sebuah komputer kecil (*“special purpose computers”*) di dalam satu IC yang berisi CPU, memori, *timer*, saluran komunikasi serial dan paralel, port *input/output*, ADC” (Andrianto,2013:1).

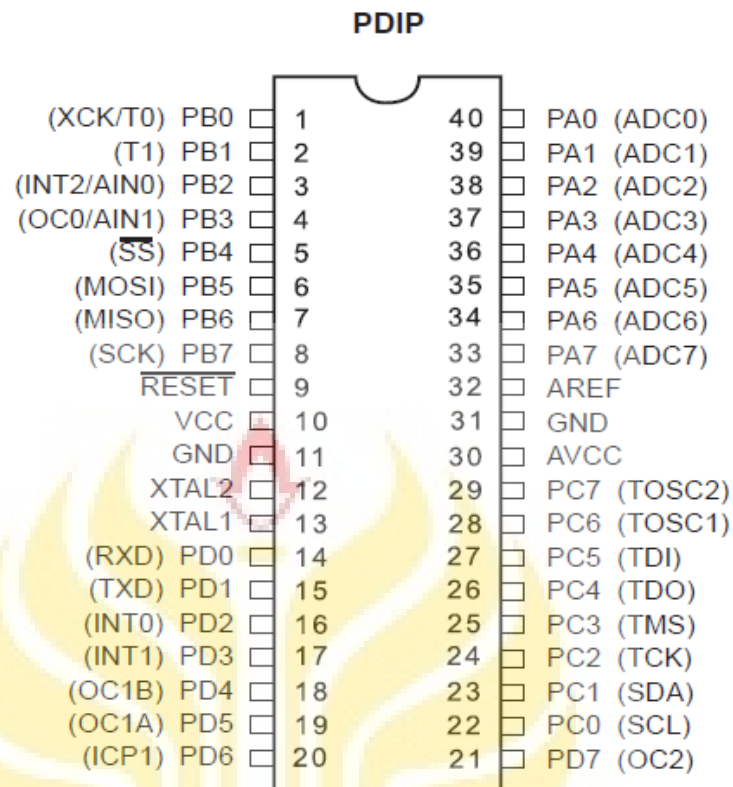
Sebagai chip IC yang dapat diprogram, mikrokontroler dapat digunakan untuk berbagai aplikasi yang dapat mempermudah pekerjaan manusia misalnya untuk sistem kendali, otomasi industri, akuisi data, telekomunikasi dan lain-lain. Saat ini keluarga mikrokontroler yang ada dipasaran yaitu Intel 8048, 8051(MCS51), Motorola 68HC11, *Microchip* PIC, Hitachi H8, Atmel AVR dan lain-lain.

Salah satu mikrokontroler yang banyak digunakan saat ini yaitu mikrokontroler Atmel AVR. AVR adalah mikrokontroler RISC (*Reduce*

Instruction Set Compute) 8 bit berdasarkan arsitektur Harvard, yang dibuat oleh Atmel pada tahun 1996 (Andrianto,2013:6).

Setiap mikrokontroler Atmel AVR mempunyai fitur yang berbeda-beda. Fitur-fitur yang dimiliki mikrokontroler ATmega32 yaitu sebagai berikut:

- a. Mikrokontroler AVR 8 bit yang memiliki kemampuan tinggi dengan daya rendah
- b. Merupakan arsitektur RISC
- c. Memiliki kapasitas *Flash* memori 32 Kbyte, EEPROM 1024 Byte dan SRAM 2 Kbyte
- d. Saluran I/O sebanyak 32 buah, yaitu PORTA, PORTB, PORTC dan PORTD
- e. Terdapat dua buah *timer/counter* 8 bit dan satu buah *timer/counter* 16bit
- f. Piranti komunikasi *Programmable* Serial USART, I2C dan Antarmuka SPI
- g. 4 *channel* PWM
- h. 8 *channel*, 10-bit ADC
- i. *On-chip* Analog Comparator
- j. Tegangan operasional antara 4.5V-5V (Datasheet ATmega32:1)



Gambar 2.5 Konfigurasi Pin ATmega8535/16/32 (Datasheet ATmega32:2)

Pada gambar 2.5, konfigurasi pin ATmega32 dalam kemasan 40 pin DIP (*Dual In-Line Package*). Berikut fungsi-fungsi dari setiap pin pada mikrokontroler ATmega32:

- 1) VCC sebagai masukan catu daya
- 2) GND merupakan pin *Ground*
- 3) PORTA (PA0-PA7) merupakan pin *input/output* dua arah dan mempunyai fungsi khusus sebagai *input* konfersi ADC.
- 4) PORTB (PB0-PB7) merupakan pin *input/output* dua arah. Masing-masing pin menyediakan resistor *pull-up* internal yang dapat diaktifkan. Selain itu, PORTB juga mempunyai fungsi khusus seperti pada tabel 2.4.

Tabel 2.4 Fungsi lain PORTB

Pin	Fungsi Khusus
PORTB.7	SCK (SPI Bus Serial Clock)
PORTB.6	MISO (SPI Bus Master Input Slave Output)
PORTB.5	MOSI (SPI Bus Master Output Slave Input)
PORTB.4	SS (SPI Slave Select Input)
PORTB.3	AIN1 (Analog Comparator Negative Input) OC0 (Timer/Counter0 Output Compare Match Output)
PORTB.2	AIN0 (Analog Comparator Positive Input) INT2 (External Interrupt 2 input)
PORTB.1	T1 (Timer/Counter0 External Counter Input)
PORTB.0	T0 T1 (Timer/Counter0 External Counter Input) XCK (USART External Clock Input/Output)

(Andrianto,2013:10)

- 5) PORTC (PC0-PC7) merupakan pin *input/output* dua arah dan sama seperti pada *port* A dan B yaitu menyediakan resistor *pull-up* internal. Selain itu, PORTC juga mempunyai fungsi khusus seperti pada tabel 2.5.

Tabel 2.5 Fungsi lain PORTC

Pin	Fungsi Khusus
PORTC.7	TOSC2 (Timer Oscilator Pin2)
PORTC.6	TOSC1 (Timer Oscilator Pin1)
PORTC.5	TDI (JTAG Test Data In)
PORTC.4	TDO (JTAG Test Data Out)
PORTC.3	TMS (JTAG Test Mode Select)
PORTC.2	TCK (JTAG Test Clock)
PORTC.1	SDA (Two-wire Serial Bus Data Input/Output Line)
PORTC.0	SDL (Two-wire Serial Bus Data Line)

(Andrianto,2013:11)

- 6) PORTD (PD0-PD7) merupakan pin *input/output* dua arah. Selain itu, PORTD juga mempunyai fungsi khusus seperti pada tabel 2.6.

Tabel 2.6 Fungsi lain PORTD

Pin	Fungsi Khusus
PORTD.7	OC2 (<i>Timer/Counter0 Output Compare Match Output</i>)
PORTD.6	ICP (<i>Timer/Counter1 Input Capture Pin</i>)
PORTD.5	OC1A (<i>Timer/Counter1 Output Compare A Match Output</i>)
PORTD.4	OC1B (<i>Timer/Counter1 Output Compare B Match Output</i>)
PORTD.3	INT1 (<i>External Interrupt 1 Input</i>)
PORTD.2	INT0 (<i>External Interrupt 0 Input</i>)
PORTD.1	TXD (<i>USART Output Pin</i>)
PORTD.0	RXD (<i>USART Input Pin</i>)

(Andrianto,2013:11)

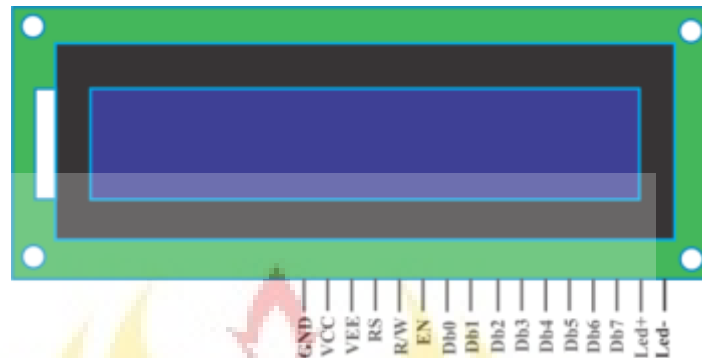
- 7) RESET merupakan pin yang digunakan untuk me-reset mikrokontroler.
- 8) XTAL1 dan XTAL2 merupakan pin masukan *clock* eksternal
- 9) AVCC merupakan pin masukan tegangan untuk ADC
- 10) AREF merupakan pin masukan tegangan untuk referensi ADC

2.2.8. Liquid Crystal Display (LCD)

LCD (*Liquid Crystal Display*) adalah *device output* yang berfungsi sebagai penampil data/informasi baik dalam bentuk karakter, huruf, angka ataupun grafik.

“LCD merupakan salah satu perangkat penampil yang sekarang ini mulai banyak digunakan. Penampil LCD mulai dirasakan menggantikan fungsi dari penampil CRT (*Cathode Ray Tube*), yang sudah berpuluh-puluh

tahun digunakan manusia sebagai penampil gambar/text baik monokrom (hitam dan putih), maupun berwarna” (Setiawan, 2011:24).



Gambar 2.6 *Liquid Crystal Display (LCD)*

Konfigurasi pin atau jalur *input* dan kontrol dalam suatu LCD (*Liquid Crystal Display*) di atas diantaranya adalah :

- a. Pin data adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan LCD (*Liquid Crystal Display*) dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit.
- b. PinRS (*Register Select*) berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika *low* menunjukkan yang masuk adalah perintah, sedangkan logika *high* menunjukkan data.
- c. PinR/W (*Read Write*) berfungsi sebagai instruksi pada modul jika *low* tulis data, sedangkan *high* baca data.
- d. Pin E (*Enable*) digunakan untuk memegang data baik masuk atau keluar.

2.2.9. Sensor Jarak HC-SR04

“Sensor jarak HC-SR04 merupakan modul sensor yang dapat mengukur jarak dengan menggunakan gelombang ultrasonik. Modul sensor ini dapat mengukur jarak antara 2cm – 400cm dengan nilai akurasi 3mm. Secara hardware, modul sensor jarak ini terdiri dari *device* pemancar dan penerima gelombang ultrasonik” (Datasheet HC-SR04:1)

Prinsip kerja sensor jarak HC-SR04 mirip dengan radar ultrasonik. Gelombang ultrasonik yang dipancarkan oleh *transmitter* kemudian diterima kembali oleh *receiver* ultrasonik. Jarak antara waktu pancar dan waktu terima adalah representasi dari jarak objek. Sensor jarak HC-SR04 adalah versi *lowcost* dari sensor jarak PING buatan parallax. Perbedaannya terletak pada pin yang digunakan. HC-SR04 menggunakan 4 pin sedangkan PING buatan parallax menggunakan 3 pin.



Gambar 2.7 Sensor Jarak HC-SR04

Untuk mengetahui nilai jarak pada sensor HC-SR04, yaitu dengan menghitung perbandingan Lebar pulsa *High* (t_{IN}) dengan kecepatan gelombang ultrasonic di udara. Lebar pulsa High (t_{IN}) merupakan nilai yang berupa pulsa *high* yang dihasilkan oleh sensor HC-SR04 ketika mendapat *trigger* (pemicu berupa tegangan 5VDC dalam waktu 10uS) pada pin *Trig*.

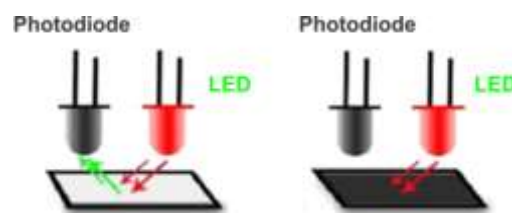
Nilai Lebar pulsa High (t_{IN}) akan sesuai dengan lama waktu tempuh gelombang ultrasonik untuk 2x jarak ukur dengan obyek. Sehingga jarak dapat ditentukan menggunakan rumus dibawah ini:

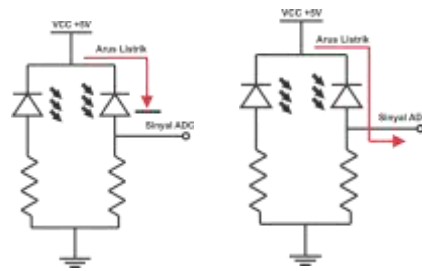
$$jarak = \left(\frac{t_{IN}(s) \times 344 (m/s)}{2} \right) \dots\dots\dots(2.1)$$

2.2.10. Sensor Cahaya (*Photodiode*)

“Sensor cahaya adalah sensor yang peka terhadap intensitas cahaya. Salah satu sensor cahaya adalah *photodiode*. *Photodiode* adalah diode yang bekerja berdasarkan intensitas cahaya, dimana jika *photodiode* terkena cahaya maka *photodiode* bekerja seperti dioda pada umumnya. Akan tetapi, jika *photodiode* tidak mendapat cahaya, maka ia akan berperan seperti resistor dengan nilai tahanan yang besar, sehingga arus listrik tidak dapat mengalir” (Jatmika, 2011:74).

“Prinsip kerja dari sensor cahaya (*photodiode*) yaitu dengan memanfaatkan sifat cahaya yang dipantulkan jika mengenai benda yang berwarna terang dan akan diserap jika mengenai benda yang berwarna gelap” (Jatmiko, 2011:131). Konsep pemantulan sensor *photodiode* terhadap benda atau garis berwarna dijelaskan pada gambar 2.8.



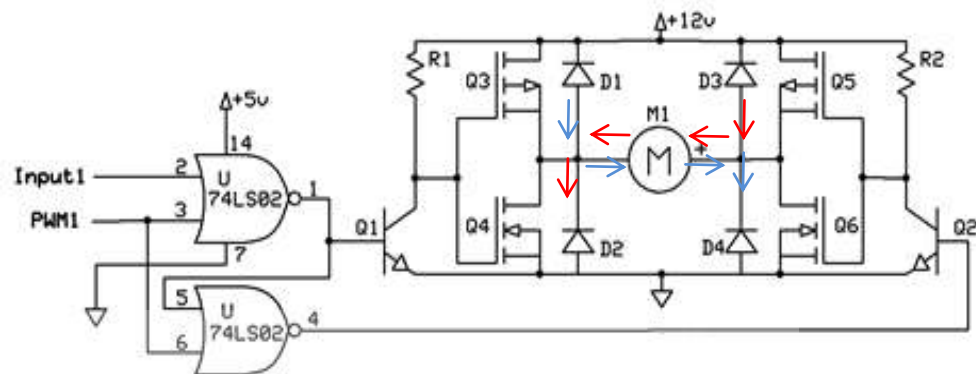


Gambar 2.8 Konsep Pemantulan Cahaya dan Ilustrasi Arus Listrik pada Sensor *Photodiode* (Yultrisna,2013:89)

Pada saat sensor *photodiode* tidak menerima intensitas cahaya, maka nilai resistansinya pada *photodiode* akan besar atau dapat dikatakan tak hingga. Sehingga tidak ada arus bocor yang mengalir menuju *input* sinyal ADC dan sebaliknya pada saat sensor *photodiode* menerima intensitas cahaya, maka sensor *photodiode* akan bersifat sebagai sumber tegangan dan nilai resistansinya akan menjadi kecil, sehingga akan ada arus bocor yang mengalir ke *input* sinyal ADC.

2.2.11. H-Bridge MOSFET

H-Bridge merupakan suatu *device* yang dapat berfungsi sebagai pengatur pergerakan motor baik arah putaran maupun kecepatan putarannya. *Device* ini dinamakan H-Bridge karena bentuk rangkaiannya menyerupai huruf H seperti pada gambar rangkaian di bawah.



Gambar 2.9 Skematik H-Bridge MOSFET

Gambar rangkaian H-Bridge MOSFET di atas terdiri dari dua buah gerbang *logic* NAND yang dalam hal ini menggunakan IC 74LS02. Selain itu, dua buah MOSFET kanal P, dua buah MOSFET kanal N dan dua buah transistor PN. Prinsip kerja dari rangkaian H-Bridge MOSFET ini yaitu dengan mengatur hidup matinya ke empat MOSFET tersebut melalui dua buah transistor PN sebagai *switching*-nya. Huruf M merupakan simbol motor DC 12V. Dengan ter-*pull-up*-nya empat buah MOSFET di atas, maka kondisi motor akan berputar jika basis kedua transistor PN saling berbeda *logic*.

Pada saat Q1 *on* dan Q2 *off*, maka hanya akan mengaktifkan MOSFET Q3 dan MOSFET Q6 sehingga sisi kiri dari gambar motor akan terhubung dengan kutub positif dari catu daya, sedangkan sisi sebelah kanan motor akan terhubung dengan kutub negatif dari catu daya dan motor akan bergerak searah jarum jam. Untuk arah pergerakan arusnya sesuai dengan panah warna biru gambar 2.9.

Sebaliknya, jika Q1 *off* dan Q2 *on* maka MOSFET Q4 dan MOSFET Q5 *on* sedangkan MOSFET Q3 dan MOSFET Q6 *off*, maka sisi kanan motor

akan terhubung dengan kutub positif dari catu daya sedangkan sisi kiri motor terhubung dengan kutub negatif dari catu daya. Maka motor akan bergerak berlawanan arah jarum jam sesuai dengan arah panah warna merah di atas.

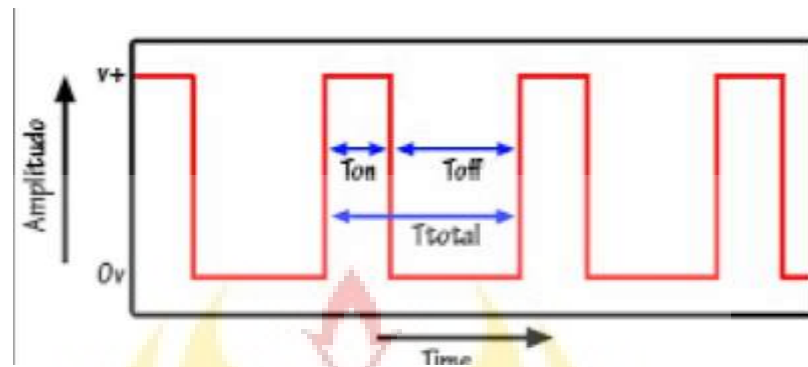
Pada kondisi lainnya adalah apabila kedua transistor PN Q1 dan Q2 dalam kondisi *on*, maka MOSFET Q3 dan MOSFET Q5 *on* sedangkan MOSFET Q4 dan MOSFET Q6 *off*. Kondisi ini akan menyebabkan sisi kiri dan kanan motor terhubung pada kutub yang sama yaitu kutub positif dari catu daya sehingga tidak ada perbedaan tegangan diantara dua buah polaritas motor, sehingga motor akan diam.

Kondisi motor akan diam juga akan terjadi pada saat kondisi kedua transistor PN *off*. Dalam hal ini hanya MOSFET dengan kanal N yang akan *on* karena MOSFET tersebut sudah ter-*pull-up*. Dengan kondisi ini baik sisi kanan maupun kiri motor akan terhubung dengan kutub negatif catu daya.

2.2.12. PWM (*Pulse Width Modulation*)

Pulse Width Modulation (PWM) merupakan salah satu sinyal listrik yang berbentuk gelombang kotak (*square*) yang menjadikan lebar-sempitnya sinyal sebagai acuan parameternya. Sinyal PWM pada umumnya memiliki amplitudo dan frekuensi yang tetap, namun memiliki lebar pulsa yang bervariasi. Lebar Pulsa PWM berbanding lurus dengan amplitudo sinyal asli yang belum termodulasi. Artinya, Sinyal PWM memiliki frekuensi

gelombang yang tetap namun *duty cycle* bervariasi (antara 0% hingga 100%).



Gambar 2.10 Bentuk Sinyal PWM

Besarnya nilai *duty cycle* pada gambar di atas adalah :

Jika D merupakan besarnya nilai *duty cycle*, maka $D = \frac{T_{on}}{T_{total}} \times 100\%$.

Dimana T_{on} merupakan nilai periode ketika sinyal PWM dalam kondisi *High*, T_{off} merupakan nilai periode ketika sinyal PWM dalam kondisi *Low* dan T_{total} adalah penjumlahan dari T_{on} dan T_{off} .

Jika V_{out} adalah besarnya nilai tegangan *output* dari sinyal PWM di atas, maka untuk menghitung besarnya nilai tegangan *output* tersebut yaitu:

$$V_{out} = D \times V_{in} \quad \dots\dots\dots(2.2)$$

$$V_{out} = \frac{T_{on}}{T_{total}} \times 100\% \times V_{in} \quad \dots\dots\dots(2.3)$$

$$V_{out} = \frac{T_{on} \times 100\% \times V_{in}}{T_{total}} \quad \dots\dots\dots(2.4)$$

Pada mikrokontroler ATmega32 terdapat fitur *Pulse Width Modulation* (PWM). PWM ini terbagi menjadi dua tipe, yaitu:

1. *Pulse Width Modulation* (PWM) 8 bit

PWM ini merupakan PWM dengan lebar data 8 bit yang dapat dihasilkan dengan mengaktifkan fungsi *timer0* atau *timer2* pada ATmega32 sebagai PWM generatornya. PWM ini juga bersifat *universal*, artinya PWM dapat dikeluarkan dari semua PORT IC (PORTA, PORTB, PORTC dan PORTD) dengan mengatur *interrupt* programnya.

Berdasarkan resolusi dan bentuk gelombang *Pulse Width Modulation* (PWM), fungsi *timer0* dan *timer2* terbagi menjadi tiga mode operasi yang diantaranya: *Normal Mode*, *Clear Timer on Compare Match* (CTC) *Mode*, *Fast PWM Mode* dan *Phase Correct PWM Mode*.

- a. *Normal Mode*, mode ini merupakan mode operasi *timer* paling mudah tanpa menggunakan fungsi *counter clear*. Pada mode ini fungsi *counter* akan secara otomatis kembali 0 (0x00) jika *counter* telah melewati batas perhitungan (*overflow*) yaitu 8bit (0xFF).
- b. *Clear Timer on Compare Match* (CTC) *Mode*, merupakan mode operasi *timer* yang gelombang pulsa-nya dapat dimanipulasi dengan menyamakan nilai TCNT (*Timer/Counter*) dengan OCR (*Output Compare Register*). Jika nilai TCNT sama dengan OCR, maka nilai counter akan kembali ke 0 (0x00).
- c. *Fast PWM Mode*, merupakan mode operasi *timer* dengan frekuensi tinggi. Mode operasi ini dapat disebut dengan *single-slope operation* yaitu dalam satu periode gelombang, fungsi *counter* akan mulai menghitung dari 0x00 (*Bottom*) sampai dengan 0xFF (*Max*). Mode

ini sangat baik digunakan untuk regulator power, penyearah dan aplikasi DAC.

- d. *Phase Correct PWM Mode*, merupakan mode operasi *timer* yang menghasilkan gelombang pulsa dengan tingkat resolusi tinggi. Mode operasi ini dapat disebut dengan *single-slope operation* yaitu dalam satu periode gelombang, fungsi *counter* akan mulai menghitung dari 0x00 (*Buttom*) sampai dengan 0xFF (*Max*) kemudian dari 0xFF (*Max*) sampai dengan 0x00 (*Buttom*). Mode ini sangat baik digunakan untuk mengatur motor/dinamo dengan tingkat presisi tinggi.

Pada fungsi *timer0* dan *timer2* terbentuk dari beberapa register.

Register-register tersebut diantaranya:

- a. **TCCR0/TCCR2** (*Timer/Counter Control Register*), Register ini berfungsi untuk mengatur mode operasi dan sumber clock *timer/counter*
- b. **TCNT0/TCNT2** (*Timer/Counter*), Register yang berfungsi sebagai reset nilai *counter* pada fungsi *timer/counter*
- c. **OCR0/OCR2** (*Output Compare Register*), Register ini merupakan Keluaran *Timer/Counter*
- d. **TIMSK** (*Timer/Counter Interrupt Mask*), Register yang berfungsi untuk mengaktifkan mode *interrupt timer/counter*

Untuk pengaturan register *timer/counter0* dan *timer/counter2* pada mode-mode operasi di atas yaitu dapat dilihat pada lembar lampiran.

2. *Pulse Width Modulation (PWM) 16 bit*

PWM 16 bit merupakan PWM dengan lebar data 16 bit yang dapat dihasilkan dengan mengaktifkan fungsi *timer1* pada ATmega32 sebagai PWM generatornya. Pada PWM ini bersifat khusus, artinya PWM hanya dapat dikeluarkan dari PORTD.5 dan PORTD.6.

Pada fungsi *timer1* terdapat beberapa mode operasi pembangkit gelombang *Pulse Width Modulation (PWM)* yang diantaranya: *Normal Mode*, *Clear Timer on Compare Match (CTC) Mode*, *Fast PWM Mode*, *Phase Correct PWM Mode* dan *Phase and Frequency Correct PWM Mode*. Untuk pengertian dan penggunaan setiap mode operasi hampir sama dengan mode operasi pada *timer 8 bit*, hanya pada mode tersebut mempunyai lebar data 16 bit.

Seperti pada fungsi *timer0* dan *timer2*, pada *timer1* juga terbentuk dari beberapa register. Berikut merupakan register-register pembentuk fungsi *timer1* diantaranya :

- a. TCCR1A (*Timer/Counter Control Register Unit A*) : Register ini berfungsi untuk mengatur mode operasi *timer/counter*.
- b. TCCR1B (*Timer/Counter Control Register Unit B*) : Register ini berfungsi untuk mengatur mode keluaran yang berupa *inverting/non inverting PWM* pada register OCR1A dan OCR1B.
- c. TCNT1 (*Timer/Counter*) : Register ini berfungsi sebagai reset nilai *counter* pada fungsi *timer/counter1*

- d. OCR1A (*Output Compare Register Unit A*) : Register yang berfungsi sebagai jalur keluaran unit A dari fungsi *Timer/Counter1* yang berupa PWM.
- e. OCR1B (*Output Compare Register Unit B*) : Register yang berfungsi sebagai jalur keluaran unit A dari fungsi *Timer/Counter1* yang berupa PWM.
- f. ICR1 (*Input Compare Register*) : Register ini berfungsi sebagai jalur masukan dari fungsi *timer/counter1* yang dapat digunakan sebagai alternatif nilai *overflow* dari *counter* pada fungsi *timer/counter1*.

Untuk mengaktifkan *timer/counter1* maka dilakukan pengaturan register-register pada fungsi *timer/counter1* tersebut sesuai dengan mode operasinya. Untuk pengaturan register-register berdasarkan mode operasinya dapat dilihat pada lembar lampiran.

2.2.13. Motor DC

Motor DC merupakan salah satu jenis motor listrik yang dapat bergerak jika mendapat *supplay* tegangan searah, semakin besar *supplay* tegangan maka akan semakin cepat putaran yang dihasilkan.

Bagian utama motor DC adalah stator dan rotor dimana kumparan medan pada motor dc disebut stator (bagian yang tidak berputar). Ketika Motor DC mendapat *supplay* tegangan searah maka kumparan rotor akan menghasilkan medan magnet sehingga rotor dapat berputar dan menghasilkan energi mekanik.



Gambar 2.11 Motor DC

Dengan adanya perbedaan potensial atau tegangan yang diberikan pada motor DC maka akan mengakibatkan motor berputar dengan arah tertentu, dan jika polaritas tegangan tersebut dibalik polaritasnya maka putaran motor akan berubah berlawanan dengan arah sebelumnya. Prinsip ini sesuai dengan Hukum gaya *Lorentz* yaitu arus mengalir pada penghantar dalam medan magnet akan menimbulkan gaya yang memiliki arah vektor bergantung pada arah arus yang mengalir.

BAB V

SIMPULAN DAN SARAN

5.1 Simpulan

Dari hasil penelitian ini, dapat disimpulkan bahwa:

1. Sistem perancangan robot *wall-follower* menggunakan sensor jarak HC-SR04 dapat berjalan dengan baik. Dengan nilai ketelitian sensor jarak HC-SR04 sebesar 98,45%, sensor ini sangat layak untuk digunakan sebagai sensor utama pada robot *wall-follower*. Penggunaan sensor *photodiode* yang sangat mudah juga dapat dijadikan alternatif sebagai sensor yang dapat membedakan warna hitam, biru dan putih.
2. Penerapan metode *Simple Wall-Maze Solving* pada robot *wall-follower* dapat membantu menyelesaikan misi penyederhanaan jalur (*maze*) pada suatu trek dengan cepat.
3. Penggunaan *Bluetooth modul* HC06 yang mendukung sistem komunikasi sinkron maupun asinkron sesuai dengan fitur sistem minimum ATmega32 dengan *range data* 8bit dapat digunakan untuk *monitoring* robot *wall-follower*.

5.2 Saran

Untuk penelitian berikutnya yang berhubungan dengan sistem komunikasi robot menggunakan mikrokontroler AVR untuk lebih memahami karakteristik dari fitur USART agar sistem komunikasi lebih sempurna.

DAFTAR PUSTAKA

- Andrianto, H. 2013. *Pemograman Mikrokontroler AVR ATmega16 Menggunakan Bahasa C (CodeVisionAVR)*. INFORMATIKA. Bandung.
- Bejo, A. 2008. *C & AVR Rahasia Kemudahan Bahasa C dalam Mikrokontroler ATmega8535*. Graha Ilmu. Yogyakarta.
- Budiharto, W. 2014. *Robotika Modern-Teori dan Implementasi*. ANDI OFFSET. Yogyakarta.
- Datasheet ATmega32. "8-bit AVR Microcontroller with 32Bytes in –System Programmable Flash" ATMEL.
- Datasheet HC-06. "Bluetooth HC-06 With Serial Port Module Easy Guide" PUNTO FLOTANTE.
- Datasheet HC-SR04. "Ultrasonic Ranging Module HC-SR04" ELECFREAKS.
- Djuniadi, Samsudin A., dan Feddy S.P. 2011. Sistem Akuisisi Data Berbasis Telemetry, *Jurnal Sains dan Teknologi (SainTeknol) UNNES* , Vol.2, No.1, hal. 79.
- Forouzan, Behrouz A. 2003. *Data Communications and Networking*. McGraw-Hill. New York.
- Jogiyanto. 2006. *Konsep Dasar Pemograman Bahasa C*. 4th Wd. ANDI OFFSET. Yogyakarta.
- Kementerian Riset, Teknologi dan Pendidikan Tinggi. 2015. *Panduan Kontes Robot Pemadam Api Indonesia (KRPAI) 2015*. Direktorat Jenderal Perguruan Tinggi. Bagian Penelitian dan Pengabdian Masyarakat. Jakarta.

- Pitowarno, E. 2006. *Robotika: Desain, Kontrol, dan Kecerdasan Buatan*. ANDI OFFSET. Yogyakarta.
- Pratama, N.A, dan Tayantoro A. 2014. Telekomunikasi Pada Robot Swarm Pemadam Api menggunakan Protokol ModBus. *Edu ElektriKa Jurnal*, Vol.6, No.2, hal.1.
- Purbawanto, S. 2011. Pengaruh *Fading* Pada Sistem Komunikasi Gelombang Mikro Tetap dan Bergerak, *Jurnal Teknik Elektro*, Vol.3, No.1, hal. 36.
- Sartika, M.E, Mulyady, Nelson. M.P, dan Junaidi.S. 2014. Sistem Navigasi Berbasis *Maze Mapping* pada Robot Pemadam Api, *Penelitian Terapan, Universitas Kristen Maranatha, Bandung*.
- Setiawan, A. 2011. *MIKROKONTROLER ATMEGA 8535 & ATMEGA 16 menggunakan BASCOM-AVR*. ANDI OFFSET. Yogyakarta.
- Shofa, F., dan Tatyantoro A. 2015. Penerapan Metode Simple Maze pada Robot Wall Follower untuk Menyelesaikan Labirin. *Edu ElektriKa Jurnal*, Vol.2, No.2, hal. 35.
- Sugiyono. 2013. *Metode Penelitian Kuantitatif Kualitatif dan R&D*. ALFABETA. Bandung.
- Winoto, A. 2010. *Mikrokontroler AVR ATmega8/32/16/8535 dan Pemrograman dengan Bahasa C pada WinAVR*. INFORMATIKA. Bandung.
- Yultrisna, dan A. Syofian. 2013. Rancang Bangun Robot *Solving Maze* dengan Algoritma *Depth First Search*, *Jurnal Momentum*, Vol.15, No.2, hal. 89-90.