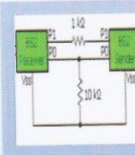
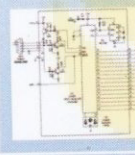
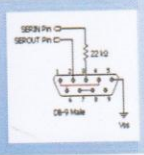
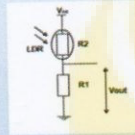
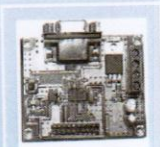


2016



# Belajar Mikrokontroler PIC16C57 dengan Bahasa Pemrograman Pbasic

**UNNES**  
UNIVERSITAS NEGERI SEMARANG

Dr. Sujarwata, M.T.

## UU No. 19 Tahun 2002 Tentang Hak Cipta

### Fungsi dan Sifat Hak Cipta Pasal 2

1. Hak Cipta merupakan hak eksklusif bagi pencipta atau pemegang Hak Cipta untuk mengumumkan atau memperbanyak ciptaannya, yang timbul secara otomatis setelah suatu ciptaan dilahirkan tanpa mengurangi pembatasan menurut peraturan perundang-undangan yang berlaku.

### Hak Terkait Pasal 49

1. Pelaku memiliki hak eksklusif untuk memberikan izin atau melarang pihak lain yang tanpa persetujuannya membuat, memperbanyak, atau menyiarkan rekaman suara dan/atau gambar pertunjukannya.

### Sanksi Pelanggaran Pasal 72

1. Barangsiapa dengan sengaja dan tanpa hak melakukan perbuatan sebagaimana dimaksud dalam pasal 2 ayat (1) atau pasal 49 ayat (2) dipidana dengan pidana penjara masing-masing paling singkat 1 (satu) bulan dan/atau denda paling sedikit Rp 1.000.000,00 (satu juta rupiah), atau pidana penjara paling lama 7 (tujuh) tahun dan/atau denda paling banyak Rp 5.000.000.000,00 (lima miliar rupiah).
2. Barangsiapa dengan sengaja menyiarkan, memamerkan, mengedarkan, atau menjual kepada umum suatu ciptaan atau barang hasil pelanggaran Hak Cipta sebagaimana dimaksud dalam ayat (1), dipidana dengan pidana penjara paling lama 5 (lima) tahun dan/atau denda paling banyak Rp 500.000.000,00 (lima ratus juta rupiah)

**BELAJAR MIKROKONTROLER PIC16C57  
DENGAN BAHASA PEMROGAMAN PBASIC**

**Sujarwata**

Desain Cover : Herlambang Rahmadhani  
Tata Letak Isi : Cinthia Morris Sartono

Cetakan Pertama: November 2016

Hak Cipta 2016, Pada Penulis

---

Isi diluar tanggung jawab percetakan

---

Copyright © 2016 by Deepublish Publisher  
All Right Reserved

Hak cipta dilindungi undang-undang  
Dilarang keras menerjemahkan, memfotokopi, atau  
memperbanyak sebagian atau seluruh isi buku ini  
tanpa izin tertulis dari Penerbit.

**PENERBIT DEEPUBLISH**  
**(Grup Penerbitan CV BUDI UTAMA)**  
Anggota IKAPI (076/DIY/2012)

Jl.Rajawali, G. Elang 6, No 3, Drono, Sardonoharjo, Ngaglik, Sleman  
Jl.Kaliurang Km.9,3 – Yogyakarta 55581  
Telp/Faks: (0274) 4533427  
Website: www.deepublish.co.id  
www.penerbitdeepublish.com  
E-mail: deepublish@ymail.com

---

**Katalog Dalam Terbitan (KDT)**

---

**SUJARWATA**

Belajar Mikrokontroler PIC16C57 dengan Bahasa Pemrograman Pbasic/oleh  
Sujarwata.--Ed.1, Cet. 1--Yogyakarta: Deepublish, November 2016.

viii, 113 hlm.; Uk:17.5x25cm

ISBN 978-602-401-586-2

1. Elektronika

I. Judul  
621.381

**Belajar**

**Mikrokontroler PIC16C57 Dengan Bahasa  
Pemrograman Pbasic**



Dr. Sujarwata, M.T.

# Kata Pengantar

Assalamualaikum Wr. Wb.

Alhamdulillah robbil 'alamin. Segala puji bagi Allah S.W.T. atas segala nikmat, rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan buku berjudul : Mikrokontroler PIC16C57 Dengan Bahasa Pemrograman Pbasic. Dalam penyusunannya buku, penulis memperoleh banyak bantuan berbagai pihak, karena itu ucapan terima kasih yang sebesar-besarnya kepada: segenap dosen fisika FMPIA UNNES dan teman-teman Program Pascasarjana Teknik Elektro UGM.

Isi buku ini sengaja disajikan secara praktis dan lengkap sehingga dapat membantu para siswa, mahasiswa, dosen, guru serta para praktisi industri. Penekanan dan cakupan bidang yang dibahas dalam buku ini sangat membantu dan berperan sebagai sumbangsih pemikiran dalam mendukung pemecahan permasalahan yang selalu muncul didalam disain, pengendalian sistem. Oleh karena itu, buku ini disusun secara integratif antar disiplin ilmu yaitu elektronika analog, elektronika daya, teknik digital dan pemrograman, sehingga skill yang diperlukan terkait satu dengan lainnya.

Tiada gading yang tak retak, begitupun dengan buku ini masih banyak kekurangan dan jauh dari kesempurnaan. Untuk itu melalui kata pengantar ini penulis sangat terbuka menerima kritik dan saran membangun sehingga secara bertahap penulis dapat memperbaikinya. Saya ucapkan selamat membaca dan mencoba program-program PBasic yang dijelaskan dalam buku ini. Semoga bermanfaat dan bisa memberikan pencerahan bagi dosen, guru, mahasiswa ,hobist dan pembaca yang mengawali belajar Robotik. Akhir kata penulis berharap agar buku ini bermanfaat bagi semua pembaca.

Amiin Ya Robbal Alamin .Wassalamualaikum Wr.Wb

Semarang, 2016

Sujarwata

# Daftar Isi

<b>1</b>	<b>PENDAHULUAN</b>	<b>1</b>
1.1	Latar Belakang . . . . .	4
1.2	Permasalahan . . . . .	6
1.3	Belajar Mikrokontroler . . . . .	7
1.4	Tujuan . . . . .	7
1.5	Metodologi Pemecahan Masalah . . . . .	9
<b>2</b>	<b>Mikrokontroler <i>Basic Stamp</i></b>	<b>11</b>
2.1	Spesifikasi Mikrokontroler <i>BS2SX (IC PIC16C57)</i> . . . . .	13
2.2	Pemrograman BS2SX (IC PIC16C57) . . . . .	16
2.3	Koneksi Pemrograman <i>Basic Stamp</i> . . . . .	17
2.4	Aplikasi Mikrokontroler <i>BS2SX</i> . . . . .	18
2.5	Organisasi memori <i>Basic Stamp</i> . . . . .	20
2.5.1	<i>RAM</i> . . . . .	20
2.5.2	<i>EEPROM</i> . . . . .	21
<b>3</b>	<b>Motor Servo</b>	<b>24</b>
3.1	Bagian-bagian dari sebuah motor servo standar . . . . .	25
3.2	Tipe Motor Servo . . . . .	26
3.3	PWM ( <i>Pulse Width Modulation</i> ) . . . . .	27
3.4	Modifikasi Motor Servo . . . . .	28
3.4.1	Cara Modifikasi Komponen Motor Servo . . . . .	28
3.4.2	Cara Modifikasi Konektor Kabel Motor Servo . . . . .	32
3.5	Pemrograman cara kerja motor servo . . . . .	33
3.6	Pengujian Motor Servo . . . . .	34

3.7	Hasil Pengujian Motor Servo . . . . .	34
<b>4</b>	<b>LDR (<i>Light Dependent Resistor</i>)</b>	<b>36</b>
4.1	Karakteristik Sensor Cahaya LDR . . . . .	38
4.1.1	Laju Recovery Sensor Cahaya LDR . . . . .	38
4.1.2	Respon Spektral Sensor Cahaya LDR . . . . .	38
4.1.3	Prinsip Kerja Sensor Cahaya LDR . . . . .	39
4.1.4	Kendali Sensor LDR dengan Mikrokontroler . . . . .	39
4.1.5	Bagaimana Cara Menampilkan Nilai <i>R<sub>C</sub>time</i> Photoresistor . . . . .	42
4.2	Aplikasi LDR ( <i>Light Dependent Resistor</i> ) . . . . .	43
<b>5</b>	<b>Sensor Gelombang Ultrasonik</b>	<b>46</b>
5.1	Gelombang Ultrasonik . . . . .	48
5.2	Sensor gelombang ultrasonik . . . . .	50
5.3	Panjang gelombang ultrasonik dan frekuensi . . . . .	52
5.4	Pengujian sensor gelombang ultrasonik . . . . .	53
5.4.1	Pengujian variabel jarak R ke T dipasang secara horisontal . . . . .	53
5.4.2	Pengujian jarak antara R dan T dipasang secara vertikal . . . . .	53
5.4.3	Pengujian variabel panjang <i>loop</i> dipasang secara horisontal . . . . .	54
5.4.4	Pengujian variabel panjang <i>loop</i> dipasang secara vertikal . . . . .	54
<b>6</b>	<b>Perancangan dan Realisasi Sistem Mobil Robot</b>	<b>58</b>
6.1	Peralatan yang Digunakan . . . . .	61
6.1.1	Perangkat Keras . . . . .	62
6.1.2	Perangkat Lunak . . . . .	62
6.1.3	Ketentuan Pemrograman <i>Parallax Basic (PBasic)</i> . . . . .	64
6.2	Pengujian Robot mobil . . . . .	64
6.2.1	Pengujian hubungan antara jumlah pulsa diinputkan dengan jarak yang ditempuh oleh mobil robot bergerak maju . . . . .	64
6.2.2	Pengujian hubungan jumlah pulsa diinputkan dengan sudut belok mobil robot . . . . .	65
6.3	BASIC Stamp Editor . . . . .	68
6.4	Komunikasi <i>Basic Stamp</i> . . . . .	72
6.5	<i>Port Serial/ RS-232</i> . . . . .	73

6.6	Contoh Pemrograman <i>PBasic</i> . . . . .	75
<b>7</b>	<b>Mobil Robot Produksi Parallax,Inc</b>	<b>82</b>
7.1	Konektor Motor Servo DC . . . . .	83
7.2	Contoh Beberapa Gambar Sensor . . . . .	84
7.2.1	Sensor Gelombang Ultrasonik . . . . .	84
7.2.2	Sensor Inframerah . . . . .	85
7.2.3	Sensor <i>UV-Tron</i> . . . . .	86
7.2.4	Sensor Kompas . . . . .	86
7.2.5	Sensor gerak atau PIR . . . . .	87
7.2.6	Sensor Penyandi ( <i>Encoder</i> ) . . . . .	88
7.2.7	Sensor Suhu (Temperatur/ Thermal) . . . . .	89
7.2.8	Sensor Cahaya . . . . .	93
7.2.9	Sensor Tekanan MPX4100 . . . . .	98
7.2.10	Sensor gas . . . . .	100
<b>8</b>	<b>Penutup</b>	<b>105</b>
8.1	KESIMPULAN . . . . .	106
8.2	SARAN . . . . .	107
	<b>Daftar Pustaka</b>	<b>108</b>
	<b>Ideks</b>	<b>108</b>
	<b>Glosarium</b>	<b>110</b>





# Daftar Tabel

2.1	Organisasi <i>RAM</i> . . . . .	20
2.2	Deklarasi variabel . . . . .	21
2.3	Cara deklarasi variabel . . . . .	22
2.4	Lokasi <i>EEPROM</i> . . . . .	23
3.1	Hubungan jumlah pulsa dengan jarak tempuh . . . . .	35
3.2	Hubungan jumlah pulsa dengan sudut putaran motor servo . . . . .	35
5.1	Hasil pengujian variabel jarak R dan T dipasang secara horisontal . . . . .	53
5.2	Hasil pengujian variabel jarak R dan T dipasang secara vertikal . . . . .	53
5.3	Hasil pengujian variabel panjang <i>loop</i> dipasang secara horisontal . . . . .	54
5.4	Hasil pengujian variabel panjang <i>loop</i> dipasang secara vertikal . . . . .	54
6.1	Hubungan antara pulsa dengan jarak pada saat mobil robot bergerak maju . . . . .	65
6.2	Hubungan jumlah pulsa diinputkan dengan sudut belok mobil robot . . . . .	66
6.3	Konfigurasi <i>pin</i> dan nama sinyal konektor <i>serial DB-9</i> . . . . .	76

# Daftar Gambar

2.1	Mikrokontroler <i>Basic Stam Bs2x</i> . . . . .	12
2.2	Pemrograman dan Koneksi Komunikasi <i>Run-time</i> Untuk Seluruh <i>Model BS2</i>	17
2.3	Diagram skema mikrokontroler BS2SX . . . . .	18
3.1	Motor Servo Standard . . . . .	25
3.2	Pensinyalan motor servo . . . . .	27
3.3	Perlengkapan untuk modifikasi motor servo . . . . .	28
3.4	Membuka <i>Case Bottom</i> . . . . .	29
3.5	Melepas papan rangkaian pada motor servo . . . . .	29
3.6	4 <i>gear</i> penggerak . . . . .	29
3.7	<i>Tab Stop</i> pada <i>gear</i> penggerak . . . . .	30
3.8	Memotong ketiga kaki potensiometer . . . . .	30
3.9	Menyolder dan membentuk kedua resistormenjadi satu . . . . .	31
3.10	Menghubungkan resistor pengganti dengan potongan potensiometer . . . . .	31
3.11	Pemasangan pada titik-titik solder bekas potensiometer . . . . .	31
3.12	Susunan kabel konektor servo pabrikasi standar . . . . .	32
3.13	Mengganti posisi kabel Merah dengan Putih . . . . .	32
3.14	Susunan kabel Hitam, Putih/ Kuning dan Merah . . . . .	32
3.15	Deretan pulsa kotak . . . . .	33
4.1	Sensor Cahaya LDR ( <i>Light Dependent Resistor</i> ) . . . . .	37
4.2	Rangkaian dua LDR dengan RC . . . . .	40
5.1	Detektor ultrasonik (Joseph J, Corr, 1993) . . . . .	50
5.2	Arah rambat gelombang . . . . .	51
6.1	Skema Proses pembuatan mobil robot . . . . .	60

6.2	Skema Pemasangan Penerima (R) dan Pemancar (T) . . . . .	63
6.3	Menjalankan <i>BASIC Stamp Editor V 2.4</i> . . . . .	69
6.4	Tampilan Utama <i>BASIC Stamp Editor V 2.4</i> . . . . .	69
6.5	Tipe Mikrokontroler Melalui Menu Utama . . . . .	69
6.6	Tabel penyesuaian <i>baudrate</i> . . . . .	70
6.7	Hasil Pemeriksaan Sintak yang Benar . . . . .	71
6.8	Tampilan Pendownload Program . . . . .	71
6.9	Komunikasi <i>Basic Stamp</i> . . . . .	72
6.10	Komunikasi <i>Basic Stamp</i> ke <i>Basic Stamp</i> . . . . .	73
6.11	Contoh program <i>SEROUT</i> . . . . .	73
6.12	Contoh program <i>SERIN</i> . . . . .	74
6.13	Konektor DB-9 pada bagian belakang CPU . . . . .	74
6.14	Komunikasi RS232 ke <i>Basic Stamp</i> . . . . .	75
6.15	Instruksi perangkat lunak Windows . . . . .	77
7.1	Gambar Mobil Robot Produksi Parallax . . . . .	82
7.2	Diagram Skema konektor Motor Servo DC . . . . .	83
7.3	Motor Servo DC dengan 3 kabel konektor . . . . .	83
7.4	Skema Diagram Konektor Motor Servo DC . . . . .	84
7.5	Sensor Gelombang Ultrasonik . . . . .	85
7.6	Sensor Inframerah . . . . .	85
7.7	Sensor UVTron . . . . .	86
7.8	Sensor Kompas . . . . .	87
7.9	Sensor gerak . . . . .	87
7.10	Sensor penyandi . . . . .	89
7.11	<i>Thermocouple</i> . . . . .	90
7.12	<i>Thermocouple</i> . . . . .	91
7.13	<i>Thermistor</i> . . . . .	92
7.14	LDR ( <i>Light Depending Resistor</i> ) . . . . .	94
7.15	sel surya . . . . .	95
7.16	konstruksi dari sensor cahaya tipe fotovoltaik . . . . .	95
7.17	Photo transistor . . . . .	96
7.18	Photo dioda . . . . .	97

7.19	Sensor infra merah . . . . .	97
7.20	Bentuk fisik dari sensor tekanan MPX4100 . . . . .	98
7.21	Diagram Blok Internal Sensor Tekanan MPX4100 . . . . .	99
7.22	Grafik Tegangan Output Sensor Tekanan MPX4100 . . . . .	100
7.23	Skema hasil akhir proses pembuatan OFET . . . . .	101
7.24	Hasil proses pembuatan OFET . . . . .	101
7.25	Hasil pemotongan substrat pada proses pembuatan OFET . . . . .	102
7.26	Hasil akhir proses pembuatan OFET . . . . .	102
7.27	Hasil pembuatan OFET terpasang pada PCB . . . . .	103



# Bab 1

## PENDAHULUAN

Kehidupan manusia mengenal komputer baru pada 50 tahun terakhir, namun mempunyai pengaruh sangat besar dalam merubah kehidupan manusia, bahkan melebihi penemuan-penemuan manusia lainnya seperti: radio, telepon, automobil, dan televisi. Begitu banyak aplikasi memanfaatkan komputer, terutama dalam pemanfaatan kemampuan *chip mikroprosesor* di dalamnya yang dapat melakukan komputasi sangat cepat, dapat bekerja sendiri dengan diprogram dan dilengkapi memori untuk menyimpan begitu banyak data. Seiring dengan perkembangan zaman, semakin luaslah kebutuhan akan kemampuan seperti yang dimiliki oleh komputer, sehingga menyebabkan munculnya terobosan baru yang salah satunya adalah dibuatnya *chip* mikrokontroler.

Seiring pesatnya perkembangan teknologi informasi serta otomasi industri dimana komputer menjadi bagian yang tidak terpisahkan dari teknologi itu sendiri membuat integrasi komputer dengan disiplin ilmu lain seperti elektronika, pemrograman dan mekanika tidak bisa dihindari. Munculnya mekatronika dengan wujud robot sebagai bentuk implementasinya mendorong kami mengembangkan robotika sebagai bagian dari dunia pendidikan kita. Kami menyediakan dan menyalurkan kegemaran dan pengetahuan sistem robotika serta merancang untuk aplikasi kendali dari sensor.

Perkembangan teknologi telah maju dengan pesat dalam perkembangan dunia elektronika, khususnya dunia mikroelektronika. Penemuan silikon menyebabkan bidang ini mampu memberikan sumbangan yang amat berharga bagi perkembangan teknologi modern, dalam hal dinamika robot sangat menjanjikan dalam perolehan kontribusi keilmuan.

Tujuan utama dalam dinamika robot adalah untuk mendapatkan desain kontrol yang kokoh yang mampu meredam gangguan dengan baik. Masih banyak struktur-struktur robot yang kompleks belum dikaji secara mendalam model dinamikanya, oleh karena rumitnya persoalan pemodelan matematik sistem robotik, sifat fisik alami dan lingkungan. Dilihat dari struktur dan fungsi fisiknya (pendekatan visual) robot terdiri dari dua bagian, yaitu *non-mobile robot* dan *mobile robot*.

Kombinasi keduanya dapat menghasilkan kelompok kombinasi konvensional ((*mobile* dan *non-mobile*) dan kelompok non konvensional. Kelompok pertama sengaja diberi nama konvensional karena nama yang sudah umum, seperti *mobile manipulator*, *climbing robot* (robot pemanjat), *walking robot* (misal : bi-ped robot) dan nama-nama lain yang sudah populer. Sedangkan kelompok non-konvensional dapat berupa robot humanoid, animaloid, extra-ordinary, atau segala bentuk inovasi penyerupaan yang bisa dilakukan.

Robot bawah air dan robot terbang lebih banyak dikembangkan sebagai peralatan untuk membantu penelitian yang berkaitan dan untuk proyek pertahanan atau mesin perang. Mobile robot adalah tipe robot yang paling populer dalam penelitian dunia robotik. Mobile robot sering menjadi daya tarik, tidak hanya bagi kalangan peneliti, tapi juga bagi kalangan awam. Dari segi manfaat, penelitian tentang berbagai tipe mobile robot diharapkan dapat membantu manusia dalam melakukan otomasi dalam transportasi, platform bergerak untuk robot industri, eksplorasi tanpa awak dan banyak lagi.

Sebagai objek pelatihan dan pendidikan, siswa/mahasiswa dituntut aktif, kreatif dan terlibat langsung dalam proses perakitan, pengoperasian dan mempraktekan latihan-latihan yang tersedia dalam contoh pemrograman buku ini. Setiap pemrograman mikrokontroler dibuat sesuai tingkat kemampuan siswa. Pengajaran ditingkat SD, lebih pada bermain robot tetapi tetap menjaga unsur pendidikan. Pada tingkat SMU, siswa mulai diberi pengantar pemrograman dan pengenalan teknologi sistem robotika dengan mempelajari struktur pemrograman, perakitan serta bekerja dalam kelompok. Pada tingkat perguruan tinggi, mengajak mahasiswa tidak hanya mengerti prinsip dasar sistem robotika dan pemrograman mikrokontroler namun juga ditantang untuk memodifikasi program dan konstruksi robot.

Mikrokontroler adalah sebuah sistem mikroprosesor lengkap yang terkandung di dalam sebuah *chip*. Mikrokontroler berbeda dari mikroprosesor serba guna yang digunakan dalam sebuah *personal computer* (PC), karena sebuah mikrokontroler pada umumnya telah berisi komponen pendukung sistem minimal mikroprosesor, yakni memori dan antarmuka I/O. Mikrokontroler cenderung beroperasi pada manipulasi *bit*, sedangkan mikroprosesor cenderung beroperasi pada operasi *byte* (8 bit). Mikrokontroler adalah *single chip computer* yang memiliki kemampuan untuk diprogram dan digunakan untuk tugas-tugas yang berorientasi pada pengendali.

Mikrokontroler datang dengan dua alasan utama, yang pertama adalah kebutuhan pasar (*market need*) dan yang kedua adalah perkembangan teknologi baru. Yang dimaksud dengan kebutuhan pasar adalah kebutuhan yang luas dari produk-produk elektronik akan perangkat pintar sebagai pengontrol dan pemroses data. Sedangkan yang dimaksud dengan perkembangan teknologi baru adalah perkembangan teknologi semikonduktor yang memungkinkan pembuatan *chip* dengan kemampuan komputasi yang sangat cepat, bentuk yang semakin mungil, dan harga yang semakin murah.

Sistem robotika yang memanfaatkan mikrokontroler saat ini berkembang dengan pesat. Banyak orang berlomba untuk membuat robot cerdas yang lebih baik dan dapat membantu kebutuhan tertentu manusia. Untuk dapat mengendalikan sebuah robot harus ditanamkan sebuah *microcontroller* atau mikroprosesor sebagai otak dari robot tersebut. Dengan semakin maraknya pengembangan sistem robotik di dalam dan di luar negeri, kami justru melihat bahwa penelitian mengenai sistem robotik ini sangat penting dan perlu di kaji secara mendalam dan dikembangkan. Pada sisi yang lain kurangnya penelitian robotika yang memadai dan pengembangan penelitian lebih lanjut. Sebagai peneliti yang tertarik pada bidang sistem robotika, maka perlu ada pihak yang memulai untuk mengembangkannya atau mengangkat topik penelitian tentang sistem robotika. Robot mobil merupakan salah satu bentuk robot yang dapat digunakan untuk berbagai macam keperluan. Robot mobil dapat dimanfaatkan untuk memindahkan barang dalam suatu pabrik atau rumah sakit, melakukan pemetaan atau observasi pada suatu daerah asing yang belum terjamah, melakukan pengukuran jarak jauh (*telemetry*) pada suatu kondisi yang membahayakan manusia, menjinakkan bom, kendaraan berjalan secara otomatis dan mampu menghindari halangan di depannya, serta masih banyak manfaat yang lainnya.

## 1.1 Latar Belakang

Mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran (I/O) serta pengendali (kontrol) dengan suatu program yang dapat ditulis dan dihapus dengan cara khusus. Prinsip dari cara kerja mikrokontroler sebenarnya hanya membaca dan menulis data melalui suatu pemrograman tertentu. Sebagai contoh, bayangkan diri Pembaca saat mulai belajar membaca dan menulis, ketika Pembaca sudah bisa melakukan seperti hal itu Pembaca mulai bisa membaca tulisan apapun baik itu tulisan buku, majalah, cerpen, artikel dan sebagainya dan Pembacapun mulai bisa menulis hal-hal sebaliknya. Begitu pula jika Pembaca sudah terbiasa bahkan mahir membaca dan menulis data pada mikrokontroler maka Pembaca dapat membuat program untuk membuat suatu sistem pengaturan menggunakan mikrokontroler sesuai dengan keinginan dan kebutuhan Pembaca.

Mikrokontroler merupakan komputer didalam chip yang digunakan untuk mengontrol (mengendalikan) suatu peralatan elektronik, yang menekankan pada efisiensi dan efektifitas biaya. Secara harfiahnya bisa disebut " pengendali kecil " dimana sebuah sistem elektronik yang sebelumnya banyak memerlukan komponen-komponen pendukung seperti *IC TTL* dan *CMOS* dapat direduksi serta akhirnya terpusat pada pengendali oleh mikrokontroler melalui pemrograman.

Manfaat yang dapat diperoleh dalam menggunakan mikrokontroler adalah:

- a) Keandalan tinggi (*high reliability*) dan kemudahan integrasi dengan komponen lain (*high degree of integration*)
- b) Peralatan yang menggunakan sistem elektronik akan menjadi lebih ringkas dan praktis saat diaplikasikan.
- c) Merancang sistem elektronik akan lebih cepat karena sebagian besar dari sistem adalah perangkat lunak yang mudah dimodifikasi.



- d) Pencarian dari sebuah gangguan lebih mudah ditelusuri karena mengaplikasikan sistem yang kompak.
- e) Waktu pembuatan lebih singkat (*shorter development time*) sehingga lebih cepat pula dijual ke pasar sesuai kebutuhan (*shorter time to market*)
- f) Mikrokontroler tersusun dalam satu *chip* dimana prosesor, memori, dan I/O terintegrasi menjadi satu kesatuan kontrol (kendali) sistem sehingga mikrokontroler dapat dikatakan sebagai komputer mini yang dapat bekerja secara inovatif sesuai dengan kebutuhan sistem.

Namun demikian, tidak sepenuhnya mikrokontroler dapat mereduksi dari komponen IC TTL dan CMOS yang seringkali masih diperlukan untuk aplikasi kecepatan tinggi atau sekedar untuk menambah jumlah saluran masukan dan keluaran (I/O). Dengan kata lain, mikrokontroler adalah versi mini atau mikro dari sebuah komputer karena mikrokontroler sudah mengandung beberapa periferal yang langsung bisa dimanfaatkan, misalnya *port paralel*, *port serial*, komparator, konversi digital ke analog (DAC), konversi analog ke digital (ADC) dan sebagainya hanya menggunakan sistem minimum yang tidak rumit atau kompleks.

Masalah yang sering muncul pada sistem robotik ini adalah bagaimana cara merakit dan memasang sensor yang tepat, sehingga robot dapat melakukan gerakan sesuai dengan perintah. Pemasangan sensor sebagai pengindera robot memerlukan teknik tersendiri dengan memperhatikan karakteristik sensor yang digunakan, posisi sensor dan objek yang dideteksi. Sebagai contoh pemasangan sensor gelombang ultrasonik pada robot beroda harus mengetahui spesifikasi sensor, meletakkan sensor dan posisi objek yang dideteksi. Dengan adanya kit mobil robot mahasiswa akan lebih mudah untuk membuat dan mempelajari robot, karena dapat disimulasikan terlebih dahulu sebelum diaplikasikan pada robot.

Salah satu alasan penggunaan gelombang ultrasonik, adalah panjang gelombang yang pendek, difraksi lebih kecil dan berkas gelombang lebih tidak menyebar, sehingga benda yang lebih kecil dapat terdeteksi. Jika panjang gelombang ultrasonik lebih kecil dari ukuran benda yang menghalangi rambat gelombang, maka benda akan memantulkan bagian yang cukup besar dari gelombang. Benda yang berukuran paling kecil yang dapat

dideteksi adalah dalam orde panjang gelombang yang digunakan. Dengan memanfaatkan frekuensi gelombang ultrasonik yang lebih tinggi, maka benda yang berukuran lebih kecil dapat terdeteksi.

## 1.2 Permasalahan

Permasalahan yang sering muncul dalam mempelajari sistem robotika adalah sebagai berikut:

- a) Bagaimana cara merakit kit sensor sehingga siap untuk diaplikasikan pada keperluan tertentu ?
- b) Bagaimana teknik memasang sensor yang tepat sehingga dapat bekerja secara efektif dan menghasilkan peralatan elektronik sesuai dengan kebutuhan ?
- c) Bagaimana cara menyusun program sehingga robot mobil dapat bergerak sesuai dengan perintah dan dapat menghindari halangan ?
- d) Bagaimana cara belajar bahkan menguasai mikrokontroler untuk diaplikasikan sebagai kendali peralatan elektronik?
- e) Bagaimana dengan ketersediaan komponen mikrokontroler ?
- f) Buku-buku referensi dalam bahasa Indonesia apa saja yang sudah beredar di pasaran tentang mikrokontroler ?

Disamping permasalahan di atas adalah masalah yang sangat penting, yaitu memilih jenis mikrokontroler yang akan digunakan sering kurang tepat sehingga mengalami kesulitan dalam menyusun program pengendalinya. Untuk mengatasi hal demikian dapat digunakan mikrokontroler *Basic Stamp* dengan bahasa pemrograman *Pbasic* yang relatif sederhana, bahkan dapat menghindari algoritma yang sangat rumit. Mikrokontroler *Basic Stamp* ini keluarannya sudah dalam bentuk data digital, sehingga tidak lagi memerlukan *interface*. Dalam mikrokontroler *Basic Stamp* juga dilengkapi dengan *intepreter* yang secara otomatis menunjuk register tertentu, dengan demikian akan memudahkan pengguna. Perintah dengan bahasa sederhana dan ditulis langsung ke bentuk desimal, tidak perlu mengubahnya ke bentuk biner.

## 1.3 Belajar Mikrokontroler

Jika Anda ingin belajar atau menguasai mikrokontroler untuk keperluan pengendali peralatan elektronik ada 2 cara yang dapat ditempuh, yaitu:

1. Belajar sendiri tanpa adanya instruktur/pelatih, Anda bisa mempelajari sendiri mikrokontroler dengan panduan buku dan peralatan yang diperlukan. Langkah pertama berawal dari mempelajari contoh-contoh sederhana dalam aplikasi mikrokontroler.. Apabila menemukan suatu kesulitan tanyakan kepada teman Anda yang lebih tahu tentang mikrokontroler atau bisa Anda utarakan ke *mailing* list yang membahas mengenai hal ini.
2. Belajar pada lembaga pendidikan, cara kedua ini bisa Anda dapatkan baik melalui pendidikan formal seperti sekolah, perguruan tinggi, maupun pendidikan non formal (kursus, pelatihan, les dan sejenisnya). Dengan cara ini Anda akan lebih terprogram dan cepat dalam penguasaan mikrokontroler.

Kata kuncinya dari keberhasilan belajar mikrokontroler adalah: kemauan untuk belajar disertai latihan-latihan, uji coba dan eksperimen. Jika Anda berminat mempelajari mikrokontroler mulailah dengan mempelajari teori serta mempraktekkannya, idealnya Anda praktek menggunakan perangkat keras walaupun Anda bisa menggunakan simulator. Anda bisa memulai dengan mengaktifkan sebuah LED, setelah itu cobalah bermain-main menggeser LED dan mencoba instruksi-instruksi lain. Lambat laun Anda akan memahami bagaimana membuat struktur program yang efektif dan efisien. Belajar mikrokontroler sama halnya seperti belajar pemrograman ilmu komputer, maka harus diertai dengan teori dan praktek.

## 1.4 Tujuan

Penulisan buku ini dimaksudkan untuk belajar dan dapat menjadi gambaran bagaimana cara mengendalikan robot beroda menggunakan beberapa komponen dan sensor sebagai pendukungnya, serta memperkenalkan mikrokontroler yang digunakan yaitu *Basic Stamp* yang dibuat oleh perusahaan Parallax dan menghasilkan program yang mampu menggerakkan *robot Calbot* sesuai dengan lintasan tanpa menabrak obyek penghalang yang ada.

Mikrokontroler merupakan komputer didalam *chip* yang digunakan untuk mengontrol peralatan elektronik, yang menekankan efisiensi dan efektifitas biaya. Secara harfiahnya bisa disebut "pengendali kecil" dimana sebuah sistem elektronik yang sebelumnya banyak memerlukan komponen-komponen pendukung seperti IC TTL dan CMOS dapat direduksi dan akhirnya terpusat serta dikendalikan oleh mikrokontroler. Dengan memanfaatkan mikrokontroler ini, maka akan didapatkan beberapa keuntungan, antara lain:

- a) Sistem elektronik akan menjadi lebih ringkas
- b) Rancang bangun sistem elektronik akan lebih cepat karena sebagian besar dari sistem adalah perangkat lunak yang mudah dimodifikasi
- c) Merancang sistem pengendali pada sensor agar dapat memenuhi spesifikasi yang telah ditentukan.
- d) Memasang sensor agar dapat dikendalikan sesuai kebutuhan.
- e) Pengujian sensor bertujuan agar didapatkan teknik pemasangan yang tepat pada saat digunakan.

Bahkan Anda bisa membuat PABX mini, *SMS Gateway*, atau kearah military Anda bisa membuat radio militer frekuensi *hopping* (radio komunikasi anti sadap dengan lompatan frekuensi 100 kali dalam 1 detik), sistem monitoring cuaca dengan balon udara, *automatic vehicel locator* (menggunakan GPS), aplikasi robotik dan sebagainya. Semua itu sekedar contoh, masih banyak lagi yang bisa Anda lakukan dengan mikrokontroler. Sebagai prospek, arah perkembangan dunia elektronika saat ini adalah ke *embedded system* (sistem tertanam) atau *embedded electronic* (elektronik tertanam). salah satunya dengan menggunakan mikrokontroler, jadi jika Anda belajar dan menguasai mikrokontroler sudah tepat pada jalurnya.

Selain tujuan, perlu diketahui juga dukungan apa saja yang ada pada mikrokontroler tersebut. Kriteria ini mungkin tidak terlalu dibutuhkan untuk pengguna-pengguna pemula yang memiliki motivasi tinggi dalam mempelajari hal baru mulai dari awal. Namun untuk pengguna pemula yang ingin lebih cepat menguasai, maka dukungan mikrokontroler perlu menjadi pertimbangan.

Dukungan-dukungan tersebut berupa:

- 1) *Development System* untuk mikrokontroler tersebut di mana pengguna yang masih pemula tidak perlu merancang PCB ataupun menyolder komponen lagi sehingga dapat fokus untuk mempelajari mikrokontrolernya terlebih dahulu.
- 2) Antarmuka-antarmuka yang kompatibel dengan *Development System* yang disediakan.
- 3) Artikel-artikel berupa aplikasi-aplikasi dari mikrokontroler tersebut.

## 1.5 Metodologi Pemecahan Masalah

Dengan menguasai mikrokontroler, kita dapat menerapkan kedalam kehidupan sehari-hari seperti mengendalikan perangkat elektronik dengan berbagai sensor dan kondisi, seperti: cahaya, getaran, panas, dingin dan lain-lain. Contoh sederhana penggunaan mikrokontroler, lihatlah disekitar lingkungan Anda ada mesin cuci dan *microwave*. Kemudian tengoklah didunia pertanian Anda bisa membuat kontrol kelembaban untuk budidaya jamur dsb, didunia perikanan Anda bisa mengendalikan suhu air kolam dan sebagainya.

Penelitian robotika berkaitan dengan mobil robot yang telah dilakukan oleh penulis menggunakan perangkat keras dan perangkat lunak, adapun perangkat tersebut adalah:

- a) Perangkat keras yang terdiri dari: Mikrokontroler *Basic Stamp BS2SX Parallax*, motor servo DC, *Photoresistor*, pemancar ultrasonik dan penerima ultrasonik.
- b) Perangkat lunak (*Software*) yang digunakan riset penulis adalah pemrograman dengan bahasa *Parallax Basic (PBasic)*.

Bahasa pemrograman yang sering digunakan pada mikrokontroler adalah bahasa *Assembly* bahasa C atau Basic, masing masing mempunyai kelebihan dan kekurangan, Untuk bahasa *Assembly* adalah bahasa tingkat rendah atau bahasa mesin terkadang perintah-perintah yang digunakan kadang cukup sulit dimengerti, kelebihanannya dari segi memori mikrokontroler relatif lebih kecil (hal ini penting dalam membuat aplikasi karena adanya pembatasan kapasitas memori dari mikrokontroler). Bahasa C dan Basic sudah digolongkan bahasa pemrograman tingkat tinggi jadi perintah-perintah cukup mudah dimengerti.

Perangkat lunak merupakan faktor yang sangat penting dalam tahapan perancangan sistem robotik. Perangkat lunak ini berupa algoritma gerak dan tugas robot dalam bentuk listing program yang ditanamkan kedalam mikrokontroler. Penyusunan program dapat bermacam-macam bentuk dan bahasa pemrogramannya, sesuai kebutuhan dengan spesifikasi dari mikrokontroler yang digunakan. Mikrokontroler *Basic Stamp* menggunakan bahasa pemrograman Pbasic (Parallax Basic), yang bahasanya hampir sama dengan bahasa Basic. Perangkat lunak (*Software*) yang digunakan pada pemrograman adalah *BASIC Stamp Editor v2.4*.

Pemrograman mikrokontroler menjadi pilihan dimana mahasiswa diajak untuk belajar pemrograman mikrokontroler mulai dari yang sangat sederhana, misalnya mengaktifkan *flip-flop* hingga menjalankan manipulator robot. Selain itu, pelatihan mikrokontroler dengan kit robotik memberikan banyak kemungkinan dan sangat bervariasi. Sistem robotik adalah multidisiplin dengan komputer, elektronika dan mekanika sebagai ilmu dasarnya. Namun begitu tidak terbatas bagi pengajar untuk melibatkan ilmu-ilmu lain seperti biologi dan anatomi. Dengan *robotics walker kit*, para guru/dosen dapat melibatkan siswanya untuk mengamati gerakan kaki serangga dan mensimulasikannya sebelum diprogram *kerobot walking kit* (2 kaki, 4 kaki atau 6 kaki).

Belajar pemrograman dengan objek robot membuat proses belajar menjadi lebih menarik dan nyata. Mulai dengan pemrograman Basic hingga ke pemrograman C, siswa secara bertahap belajar pemrograman sehingga pemahaman komputer hanya untuk mengetik dan bermain akan bergeser menjadi pemikiran bahwa komputer telah menjadi bagian dari teknologi sistem kontrol dan akuisisi data. Robotics kit dapat juga digunakan dalam implementasi artificial intelligent (AI), sistem sensor, model sistem otomatisasi kontrol dan bagi yang sekedar hobi, kit dapat ditambahkan aksesoris seperti sensor, kamera, motor, rangkaian elektronika hingga pengembangan *software* dan modifikasi kit.

## Bab 2

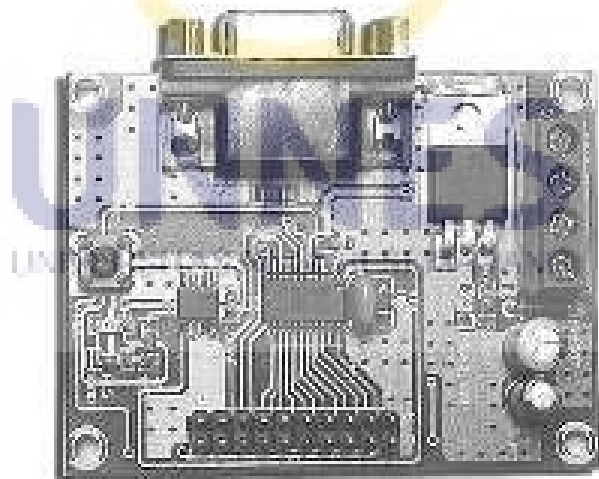
# Mikrokontroler *Basic Stamp*

Mempelajari mikrokontroler umumnya terkendala dengan urusan pemrograman berbasis *assembly*. Bila melihat sejarahnya, mikrokontroler, juga mikroprosesor, hadir dengan kemampuan memahami bahasa tersebut, yang dianggap kurang manusiawi. Namun, dalam perkembangannya, terdapat cukup banyak pilihan pemrograman mikrokontroler dengan bahasa yang lebih beradab, seperti C, BASIC, Pascal, dan yang lainnya. Hanya saja, bagi sebagian orang, rasanya kurang puas, dan kurang sempurna untuk seorang programmer mikrokontroler, kalau belum menaklukkan bahasa *assembly*. Ini adalah suatu tantangan, namun kita tidak boleh terpaksa mengutak-atik bahasa *assembly* saja.

Perkembangan teknologi yang pesat saat sekarang mengarah pada kemudahan pemakaian yang sesuai dengan kebutuhan. Maka dari itu, kalau ada pilihan yang lebih mudah dan aman dalam memprogram mikrokontroler, kenapa kita tidak menggunakannya? Kemungkinan hasilnya yang didapat kurang optimal dibanding *assembly*, namun dengan semakin meningkatnya performa mikrokontroler dan aplikasi yang dibuatpun tidak menuntut proses *realtime*; maka faktor tersebut, menurut hemat penulis, dapat diabaikan. Penulis cenderung menggunakan bahasa pemrograman *BASIC (Basic Stamp)* karena bahasa pemrograman sangat mudah dan banyak aplikasinya serta menggunakan bilangan desimal. Belajar bahasa pemrograman *PBasic* tersebut lebih aman, karena tidak tergantung pada tipe mikrokontroler yang digunakan. Sebagaimana kita ketahui, beda keluarga dari tipe mikrokontroler akan mengakibatkan beda bahasa pemrograman *assemblynya*. Sebagai contoh, keluarga MCS51, keluarga AVR dan keluarga AVR (ketiganya merupakan keluarga mikrokontroler yang cukup populer) dan memiliki pakem *assembly* yang berbeda.

*Basic stamp* adalah mikrokontroler yang dikembangkan oleh *Parallax Inc.* yang mudah diprogram menggunakan format bahasa pemrograman *Basic*. Dipanggil stamp sederhana karena ukurannya yang sebesar perangkopos. Mikrokontroler *Basic Stamp* menggunakan *power supply* saat pemrogramannya dan tidak kehilangan programnya saat baterai atau *power supply* dicabut. Kode *PBasic* disimpan di dalam *EEPROM* serial pada *board Basic Stamp*. *EEPROM* (*Electrically Erasable Programmable Read-Only Memory*) menyediakan penyimpanan yang sulit diubah, yaitu menjaga memory saat kehilangan *power*. *EEPROM* digunakan dalam *Basic Stamp 1* dan *2* yang dijamin berfungsi selama 40 tahun ke depan dan mampu untuk 10.000.000 kali penulisan ulang per lokasi memori. *EEPROM* yang digunakan dalam *Basic Stamp 2e* dan *2sx* dijamin untuk digunakan sampai 1.000.000 kali penulisan ulang per lokasi memori.

Mikrokontroler ini memiliki 16 *Kbyte memory EEPROM*, terdiri dari 8 *section* dan 2 *kybet* tiap *section*, disamping itu juga mempunyai kecepatan eksekusi 10000 instruksi/detik dan dapat dihapus hingga 100.000 kali, I/O pin dapat mengerjakan LED dan menggerakkan 6V servo motor secara langsung dan mempunyai 32 *byte ram* dengan 64 *byte scratchpad RAM*, konsumsi arus 65 mA pada saat beroperasi penuh, dan 200  $\mu$ A saat tidak beroperasi serta pada tegangan 5 s/d 15 volt DC.



Gambar 2.1: Mikrokontroler *Basic Stam Bs2x*



## 2.1 Spesifikasi Mikrokontroler *BS2SX (IC PIC16C57)*

PIC ialah keluarga mikrokontroler tipe RISC buatan Microchip Technology. Bersumber dari PIC1650 yang dibuat oleh Divisi Mikroelektronika General Instruments. Teknologi Microchip tidak menggunakan PIC sebagai akronim, melainkan nama brandnya ialah PICmicro. Hal ini karena PIC singkatan dari Peripheral Interface Controller, tetapi General Instruments mempunyai akronim PIC1650 sebagai Programmable Intelligent Computer.

PIC pada awalnya dibuat menggunakan teknologi General Instruments 16 bit CPU yaitu CP1600. \* bit PIC dibuat pertama kali 1975 untuk meningkatkan performa sistem peningkatan pada I/O). Saat ini PIC telah dilengkapi dengan EPROM dan komunikasi serial, UART, kernel kontrol motor dll serta memori program dari 512 word hingga 32 word. 1 Word disini sama dengan 1 instruksi bahasa assembly yang bervariasi dari 12 hingga 16 bit, tergantung dari tipe PICmicro tersebut. Silahkan kunjungi [www.microchip.com](http://www.microchip.com) untuk melihat berbagai produk chip tersebut.

Pada awalnya, PIC merupakan kependekan dari Programmable Interface Controller. Tetapi pada perkembangannya berubah menjadi Programmable Intelligent Computer. PIC termasuk keluarga mikrokontroler berarsitektur Harvard yang dibuat oleh Microchip Technology. Awalnya dikembangkan oleh Divisi Mikroelektronika General Instruments dengan nama PIC1640. Sekarang Microchip telah mengumumkan pembuatan PIC-nya yang keenam.

PIC cukup populer digunakan oleh para developer dan para penghobi ngoprek karena biayanya yang rendah, ketersediaan dan penggunaan yang luas, database aplikasi yang besar, serta pemrograman (dan pemrograman ulang) melalui hubungan port serial yang terdapat pada komputer.

Saat ini, begitu banyak pengembang mengembangkan turunan dari mikrokontroler yang ada, untuk tujuan kemudahan dalam penggunaan. Pola yang digunakan dalam "menciptakan" mikrokontroler turunan ini umumnya sama, yaitu menanam sebuah firmware khusus sehingga bisa bekerjasama dengan development tool berupa IDE Integrated Development Environment dan compiler, yang dikembangkan bersamaan, agar saling

mendukung. Jadi, pengembangannya dilakukan di dua sisi, sisi mikrokontroler dalam bentuk firmware dan sisi komputer berupa IDE dan compiler untuk pemrogramannya. Kelemahan umum dari mikrokontroler turunan ini adalah, tidak bisa keluar dari pakem atau aturan yang telah ditentukan sejak semula oleh pengembangnya. Misalnya, bila mikrokontroler turunan tersebut fokus pada pemrograman dengan Bahasa BASIC, maka dia tidak bisa diprogram dengan cara lain. Pemrogramannya pun harus menggunakan resources yang disediakan oleh pengembang.

Dari statistik, turunan dari mikrokontroler PIC dari Microchip adalah yang paling banyak. Dua yang cukup populer adalah BASIC Stamp dan PICAXE. Keduanya menggunakan Bahasa BASIC sebagai rujukan. Di satu sisi, keduanya menawarkan kemudahan, dan memang kenyataannya cukup mudah dan nyaman untuk pengguna pemula. Namun di sisi lain, saat kita memikirkan cara yang berbeda untuk memrogramnya, itu tidak bisa dilakukan. Firmware yang ada di dalamnya sudah dikunci dan hanya mengerti token yang dikirimkan oleh IDE terkait.

Pada BASIC Stamp, “compiler” di sisi komputer meng-kompilasi source code ke dalam bentuk token. Token-token tersebut dikirim ke mikrokontroler BASIC Stamp untuk disimpan dalam EEPROM eksternal, kemudian diterjemahkan dan di-eksekusi saat program dipanggil. Mekanisme yang digunakan masih seperti BASIC pada awalnya, yakni interpreter. PICAXE sudah menggunakan konsep compiler. Yang dikirim ke dalam chip sudah berbentuk machine code yang siap di-eksekusi. PICAXE hanya berupa single chip berbeda dengan BASIC Stamp yang berupa rangkaian yang di dalamnya sudah ditanam sebuah firmware, namun masih menyisakan sejumlah ruang untuk program yang kita kembangkan, yang ditempatkan dalam sebuah eeprom eksternal.

Contoh lain dari turunan mikrokontroler PIC adalah OOPIC. Mikrokontroler ini menerapkan konsep OOP Object Oriented Programming berbasis Bahasa BASIC di sisi pemrogramannya. Dengan demikian, mereka yang terbiasa dengan Visual Basic, diharapkan, bisa lebih cepat menguasainya. Skema/ rangkaiannya mirip dengan BASIC Stamp. Ada EEPROM eksternal untuk menyimpan object code hasil kompilasi yang dikirim dari komputer, dan dalam flash internal-nya sudah ditanam firmware yang sesuai dengan tujuan pengembangnya. Sama seperti BASIC Stamp dan PICAXE, OOPIC tidak

bisa diprogram dengan cara lain, selain yang didukungnya. Saat ini OOPIC mendukung Bahasa BASIC, Java dan C. IDE yang disediakan vendor-vendor di atas umumnya free, namun tanpa menggunakan perangkat yang mendukungnya, software tersebut tidak berguna.

Untuk mikrokontroler AVR, saya hanya mencatat satu turunan yang cukup populer, namun masih kalah populer dibanding turunan PIC; yang mengusung isu open source, yakni Arduino. Pemrogramannya menggunakan Bahasa C yang sedikit di-modifikasi untuk mencapai target kemudahan, serta menggunakan C compiler GCC yang juga open source, yang bisa run di Windows maupun Linux. Arduino menggunakan konsep yang berbeda dibanding turunan PIC di atas. Di dalam flash-nya tidak ditanam *firmware* yang spesifik. Jadi, program yang kita buat bisa disimpan dalam flash internal. Yang ditanam di dalamnya hanyalah sebuah bootloader, yang berisi sejumlah inisialisasi. Selebihnya tidak ada yang istimewa. Versi terakhir Arduino menggunakan ATmega168.

Dari contoh-contoh di atas, kita bisa melihat, bahwa banyak pengembangan dilakukan untuk memanjakan pengguna, sehingga bisa lebih produktif. Untuk aplikasi yang memerlukan waktu penyelesaian yang singkat, menggunakan mikrokontroler turunan jelas merupakan pilihan yang disarankan. Memang ada *extra cost* yang harus dibayarkan, namun itu sebanding atau bahkan lebih rendah dibanding *time saving* yang kita dapatkan

Adapun spesifikasi, fungsi kaki mikrokontroler BS2SX (IC PIC16C57 ) adalah sebagai berikut:

- 1) Basic stamp BS2SX ini menggunakan IC PIC16C57.
- 2) Terdiri dari 12 pin.
- 3) 16 Kbyte memory EEPROM, terdiri dari 8 section, 2 KByte/section dan 32 KByte RAM dengan 64 KByte (scratchpad) RAM.
- 4) Kecepatan eksekusi 10.000 instruksi per detik dan dapat dihapus 100.000 kali.
- 5) Konsumsi kuat arus yang digunakan sangat kecil, yaitu 65 mA pada saat operasi penuh dan 200  $\mu$ A saat tidak beroperasi.
- 6) Menggunakan PC serial port untuk *progaming* dan *debugging*.

- 7) Terdiri dari 16 pin I/O independen dan pin I/O dapat menggerakkan LED ( *Light Emitting Diode*).
- 8) Dapat menggerakkan servo motor secara langsung.
- 9) Menggunakan bahasa pemrograman *Pbasic* (*parallax basic*) dan terdapat LED (*power status*).
- 10) Basic Stamp terdiri dari *Interpreter*, *EEPROM* dan *Resonator*.

*Basic Stamp* mempunyai 2 macam memori, yaitu *RAM* dan *EEPROM*. Perbedaan antara *RAM* dan *EEPROM* adalah : *RAM* digunakan untuk variabel dengan program dan dapat kehilangan isi/data ketika *Basic Stamp* dalam kondisi *off* atau kehilangan *power*. *EEPROM* dapat menyimpan memori dengan atau tanpa *power* dan digunakan untuk menyimpan program saja. Penulisan data untuk menyusun suatu program agar dapat disimpan dalam *EEPROM* harus memenuhi kriteria sebagai berikut: Simbol yang digunakan dapat dipilih secara bebas, nama symbol khusus secara otomatis menunjukkan konstanta ke nomor lokasi dari data yang pertama. Item data merupakan konstanta yang mempunyai interval 0 s/d 65555, hal ini berarti bahwa data merupakan indikasi dari sebuah nilai atau kumpulan nilai.

## 2.2 Pemrograman BS2SX (IC PIC16C57)

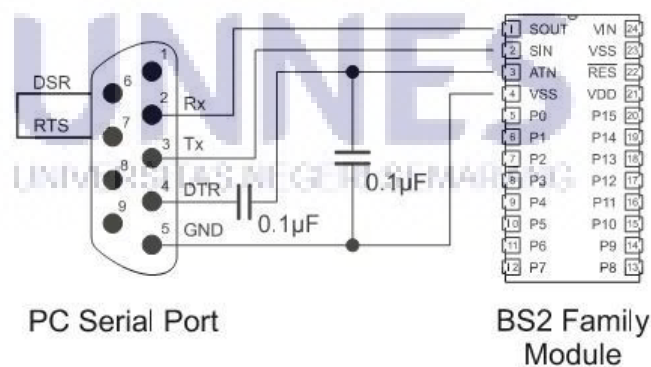
Ketentuan-ketentuan dan penjelasan cara pemrograman mikrokontroler BS2SX (IC PIC16C57), adalah sebagai berikut:

- 1) Untuk menulis program menggunakan Basic Stamp Editor yang sudah di-install sebelumnya pada PC. Setelah menulis program, PC dihubungkan dengan *Basic Stamp Microcontroller*. *Basic Stamp* harus dihubungkan dengan *battery supply*. Tekan *run* pada *Basic Stamp Editor* untuk proses *down-load* program ke *EEPROM*.
- 2) Proses *down-load* program otomatis menghapus program yang ada sebelumnya dan mikrokontroler akan mengeksekusi program yang terakhir.
- 3) Program ditulis dalam *Basic Stamp Editor* kemudian disimpan dalam format *\*BS2SX*.

- 4) Selanjutnya program yang telah disimpan dalam format *\*BS2SX* akan diubah selama proses *down-load* ke dalam format *biner* yang dapat dibaca oleh mikrokontroler menggunakan *interpreter* yang ada pada *Basic Stamp*. Komentar *Tokenzied is successfully* akan muncul apabila program *Pbasic* yang dijalankan telah berhasil.
- 5) Untuk *reprogram Basic Stamp* sangat mudah, yaitu hanya dengan menghubungkan PC, kemudian dijalankan *Basic Stamp Editor*, selanjutnya tekan pada *Alt-R* atau *RUN*.
- 6) Dibandingkan program mikrokontroler lain, pemrograman *Pbasic* akan lebih mudah.
- 7) Setiap perintah dalam program akan menempati ruang dalam *EEPROM*, besarnya bervariasi sesuai dengan kompleksitas dan jenis perintah yang ditulis.

### 2.3 Koneksi Pemrograman *Basic Stamp*

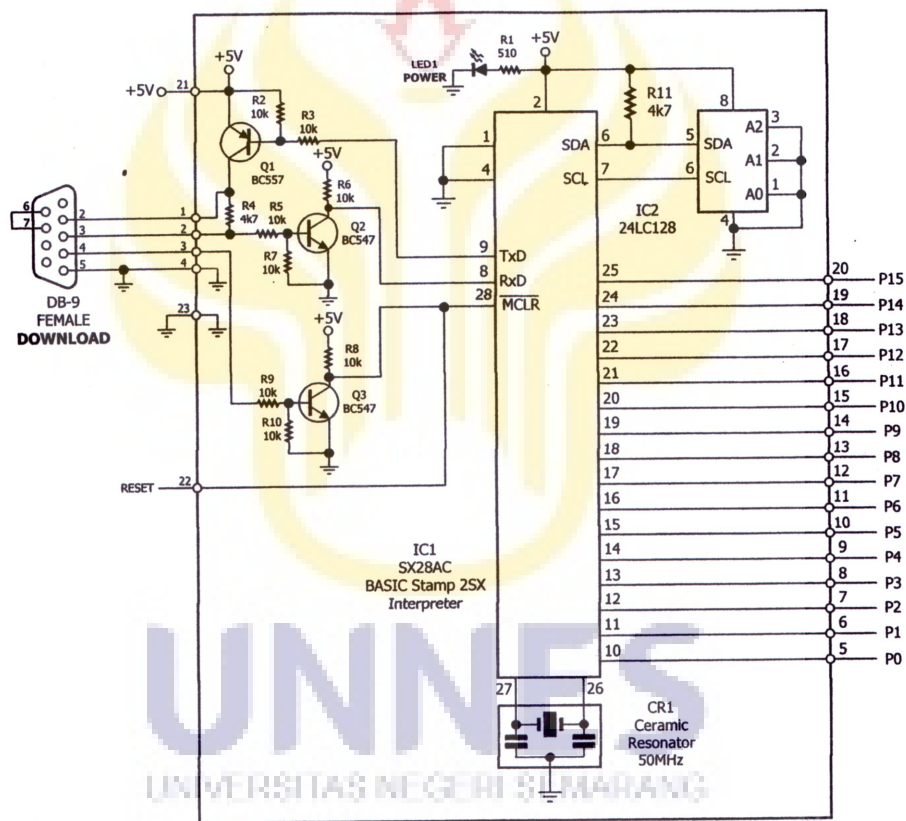
Disarankan untuk menggunakan papan buatan *Parallax* dan kabelnya dalam pemrograman modul *Basic Stamp*. Jika item ini tidak tersedia, anda dapat membuat papan sendiri dengan mencontoh diagram berikut pada circuit. Hati-hati mengikuti diagram ini, yang biasa muncul dalam pemrograman *Basic Stamp* adalah kabel modif buatan yang jelek. Dengan koneksi pemrograman untuk seluruh modul *BS2*, memungkinkan membalikkan pasangan kabel dan masih mendapat hasil yang positif dalam penggunaan pengujian koneksi teknologi *parallax*.



Gambar 2.2: Pemrograman dan Koneksi Komunikasi *Run-time* Untuk Seluruh *Model BS2*

## 2.4 Aplikasi Mikrokontroler BS2SX

Masalah yang sering muncul pada sistem robotik ini adalah bagaimana cara merakit dan memasang sensor yang tepat, sehingga robot dapat melakukan gerakan sesuai dengan perintah. Pemasangan sensor sebagai pengindera robot memerlukan teknik tersendiri dengan memperhatikan karakteristik, posisi sensor dan objek yang dideteksi. Sebagai contoh pemasangan sensor gelombang ultrasonik pada robot beroda harus mengetahui spesifikasi sensor, meletakkan sensor dan posisi objek yang dideteksi.



Gambar 2.3: Diagram skema mikrokontroler BS2SX

Disamping itu dalam memilih tipe mikrokontroler yang akan digunakan sering kurang tepat, sehingga mengalami kesulitan dalam menyusun program pengendali dan aplikasinya. Untuk mengatasi hal demikian dapat digunakan mikrokontroler *Basic Stamp* dengan bahasa pemrograman *Pbasic* yang relatif sederhana dan mudah aplikasinya, bahkan meng-

hindari algoritma yang sangat rumit. Mikrokontroler ini keluarannya sudah dalam bentuk data digital, sehingga tidak lagi memerlukan antar muka (*interface*). Dalam mikrokontroler *Basic Stamp* juga dilengkapi dengan *intepreter* yang secara otomatis menunjuk register tertentu, dengan demikian akan memudahkan pengguna. Perintah dengan bahasa sederhana dan ditulis langsung ke bentuk desimal, tidak perlu mengubahnya ke bentuk *biner*. Dalam buku ini dibahas sistem robotika yang berkaitan dengan *wheeled robot* dilengkapi dengan sensor ultrasonik untuk mendeteksi objek. Robot beroda bergerak aktif terkendali pada lintasan tertentu, dikarenakan adanya penggerak dengan sistem pengaturnya. Oleh karena itu perlu memahami bidang ilmu yang berkaitan dengan robot secara keseluruhan. Adapun bidang ilmu tersebut adalah:

- a) Kinematika, ilmu yang mempelajari geometri gerakan robot.
- b) Dinamika, merupakan bidang ilmu yang mempelajari tentang gaya sebagai penyebab robot beroda dapat bergerak,
- c) Sistem Pengendali (*Control System*), merupakan bidang ilmu yang mempelajari cara agar robot bergerak secara terkendali,
- d) Sensor, bidang ilmu yang diperlukan agar robot dapat mengindra apa yang terjadi pada dirinya dan lingkungannya,
- e) Navigasi, merupakan bidang ilmu yang diperlukan robot agar dapat menentukan arah jalannya sehingga dapat tiba ditempat tujuan dengan posisi orientasi yang benar.
- f) Komputer, bidang ilmu ini perlu dikuasai karena merupakan komponen yang menjadikan otak dalam melakukan aktivitas dari 5 elemen diatas.

Bidang ilmu tersebut, kata kunci yang mendasari dunia robotika adalah sistem pengaturan. Sebab robot pada akhirnya harus bergerak secara terkendali untuk menuju posisi yang telah ditentukan, baik itu dikontrol oleh manusia, oleh dirinya sendiri maupun dikontrol oleh robot lainnya. Perkembangan teknologi mikrokontroler dewasa ini, sangat memacu perkembangan robot, seperti Microcontroler Atmel AT89C51, Motorola 68HC11, dan PIC 16. Mengingat banyaknya kontroler yang dapat digunakan untuk tugas robot yang relatif sederhana, walaupun masih berbasis 8 bit namun kontroler ini memiliki fasilitas yang semakin canggih. Untuk robot dengan tingkat kesulitan yang lebih tinggi dapat digunakan kontroler 16/32 bit atau DSP (*Digital Signal Processing*).

## 2.5 Organisasi memori *Basic Stamp*

### 2.5.1 *RAM*

*Basic Stamp* mempunyai 32 byte *RAM* sebagai lokasi yang tersusun, sebagaimana ditunjukkan pada (Tabel 1.1 ).

Tabel 2.1: Organisasi *RAM*

Nama Word	Nama Byte	Nama Nibble	Nama Bit	Catatan Khusus
INT	INL INH	INA, INB INC, IND	IN0-IN7 IN8- IN15	Kaki-kaki input
OUTS	OUTL OUTH	OUTA, OUTB OUTC, OUTD	OUT0-OUT7 OUT8-OUT15	Kaki-kaki output
DIRS	DIRL DIRH	DIRA, DIRB DIRC, DIRD	DIR0-DIR7 DIR8-DIR15	Kaki I/O sebaga kendali
W0	B0 B1			
W1	B2 B3			
W2	B4 B5			
W3	B6 B7			
W4	B8 B9			
W5	B10 B11			
W6	B12 B13			
W7	B14 B15			
W8	B16 B17			
W9	B18 B19			
W10	B20 B21			
W11	B22 B23			
W12	B24 B25			

Variabel *Word INS* difungsikan khusus untuk membaca saja (*read only*). Variabel *INS* terdiri dari 16 bit menggambarkan keadaan I/O dari *P0* hingga *P15* yang hanya difungsikan untuk membaca saja. *IOUTS* berisi keadaan dari 16 *output*. Sedangkan kontrol *DIRS* merupakan petunjuk (*input* atau *output*) dari setiap 16 *kaki I/O*. Bilangan 0 pada *bit DIRS*



menyatakan hubungan kaki *input* dan bilangan 1 menyatakan kaki *output*. Demikian juga untuk bilangan 5 pada *bit DIRS*, berarti 1 menyatakan (*input* atau 0, sedangkan bilangan 6 menyatakan *output* karena 6 bit dari *bit DIRS* adalah 1.

*Basic Stamp* dalam keadaan *powered up* atau *reset*, semua lokasi dalam *RAM* adalah bersih tanpa memori menuju ke bilangan 0. Dengan demikian pada kondisi pin input (*bit DIRS* = (% 0000000000000000)). Jika kumpulan program *Parallax Basic* pada pin I/O menuju *output*, maka *bit DIRS* = (% 1111111111111111) dan selanjutnya diberi nama *output low*.

Aturan-aturan yang berisi tentang nama simbol harus dimulai dengan tulisan, dapat juga berisi campuran, nomer dan *underscore* ( \_ ). Nama simbol tidak boleh sama dengan kunci bahasa *Pbasic* atau lebar yang digunakan dalam program. Jumlah dari simbol terdiri dari 32 karakter.

Ukuran argumen menunjukkan jumlah karakter dalam suatu variabel yang dimungkinkan untuk menyusun suatu program. Ukuran argumen ini mempunyai 4 pilihan, pilihan tersebut ditunjukkan pada (Tabel 1.2)

Tabel 2.2: Deklarasi variabel

Ukuran Deklarasi	Jumlah bit	Interval bilangan yang mungkin
<i>BIT</i>	1	0 sampai dengan 1
<i>NIB</i>	4	0 sampai dengan 15
<i>BYTE</i>	8	0 sampai dengan 255
<i>WORD</i>	16	0 sampai dengan 65535

Beberapa contoh cara mendeklarasikan suatu variabel yang digunakan dalam pemrograman. Ada 4 cara mendeklarasikan variabel, sebagai berikut:

### 2.5.2 *EEPROM*

Penulisan data untuk menyusun program agar dapat disimpan dalam *EEPROM* harus memenuhi kriteria:

Tabel 2.3: Cara deklarasi variabel

<i>Mouse</i>	<i>VAR</i>	<i>BIT</i>	mempunyai nilai 0 atau 1
<i>Cat</i>	<i>VAR</i>	<i>NIP</i>	mempunya nilai dari 0 hingga 15
<i>Dog</i>	<i>VAR</i>	<i>Ryte</i>	mempunya nilai dari 0 hingga 255
<i>Rhino</i>	<i>VAR</i>	<i>Word</i>	mempunya nilai dari 0 hingga 65535

- a). Simbol yang digunakan pada pemrograman dapat dipilih secara bebas, kemudian untuk nama simbol khusus secara otomatis menunjukkan konstanta ke nomer lokasi dari data pertama.
- b). Item data merupakan konstanta yang mempunyai interval dari 0 sampai dengan 65535, hal ini menunjukkan bahwa data yang ditulis merupakan indikasi dari sebuah nilai atau kumpulan nilai.

Pengisian program ke dalam *EEPROM*, berarti menyimpan sejumlah nilai/data dari pengalamatan tinggi (*highest address*) menuju pengalamatan rendah (*lowest address*). Pengisian nilai/data ke *EEPROM* tidak dapat menggunakan seluruh memori yang ada pada mikrokontroler, sebab ada sebagian memori yang digunakan untuk keperluan lain. Pengertian istilah *DATA* menunjukkan sejumlah kumpulan data yang tersimpan dalam lokasi *EEPROM*.

Contoh:

*DATA*        100, 200, 52, 45

Contoh di atas menunjukkan penyimpanan kumpulan data yang mempunyai nilai, yaitu: 100, 200, 52, 45. Penulisan ke *EEPROM* menunjukkan lokasi 0, 1, 2 dan 3. Sedangkan untuk mengakses kode lokasi dapat digunakan perintah *WRITE* dan *READ*. Penggunaan *DATA* sebagai penghitung dan dinamakan *pointer*, yang dimanfaatkan untuk melanjutkan tersedianya pengalamatan *EEPROM*. Nilai inisial dari *pointer* adalah 0. Apabila terjadi penyimpanan data dari nilai byte pertama, kemudian mengalir pada pengalamatan *pointer* dan terjadi penambahan nilai (*adds 1*) pada *pointer*. Jika program berisi dari sejumlah *DATA*, nilai urutan *DATA* dimulai dengan *pointer* kiri dan selanjutnya sebelah kanannya.

Contoh:

*DATA*            72, 69, 76, 76, 79

*DATA*            104, 101, 108, 108, 111

Penyimpanan *DATA* pertama dalam *EEPROM* dimulai dari nilai lokasi 0, selanjutnya terjadi penambahan nilai data yang lain, sehingga mempunyai nilai 0, 1, 2, 3, 4. Kemudian *DATA* kedua dimulai pada nilai *pointer* 5 dan menuju pada lokasi yang selanjutnya dengan penambahan nilai 1. *DATA* kedua mempunyai nilai 5, 6, 7, 8, 9. Hasil akhir pada penyimpanan data ke dalam *EEPROM* adalah 10 *byte* pertama, hal ini dapat dilihat pada (Tabel 2.4)

Tabel 2.4: Lokasi *EEPROM*

Lokasi <i>EEPROM</i> ( <i>address</i> )										
	0	1	2	3	4	5	6	7	8	9
Isi	72	69	76	76	79	104	101	108	108	111

*Basic Stamp* mempunyai 2 macam memori, yaitu: *RAM* dan *EEPROM*.

- a) *RAM* digunakan untuk variabel dengan program dan dapat kehilangan isi/data ketika *Basic Stamp* dalam kondisi *of* atau kehilangan *power*.
- b) *EEPROM* dapat menyimpan memori dengan atau tanpa *power* dan digunakan untuk menyimpan program saja.

## Bab 3

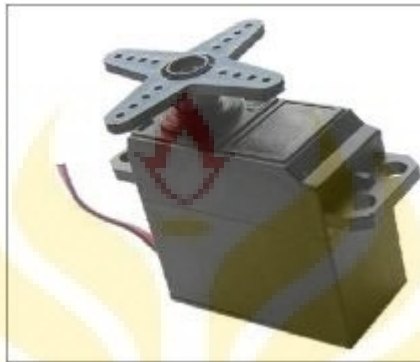
# Motor Servo

Motor Servo merupakan salah satu jenis aktuator yang banyak digunakan dalam bidang industri dan sistem robotika. Motor servo yang dibahas dalam buku ini adalah motor servo yang bertipe *FP-S3001* dengan mempunyai putaran cepat dan dapat dikendalikan dengan program. Sebelum menggunakan motor servo harus dimodifikasi terlebih dahulu, hal ini dikarenakan standar pabrik putaran motor servo hanya mencapai  $180^\circ$ . Oleh sebab itu motor servo yang akan digunakan harus dimodifikasi agar dapat mencapai putaran  $360^\circ$  (satu putaran penuh).

Sistem umpan balik tertutup digunakan dalam motor servo, di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor ini terdiri dari sebuah motor DC, serangkaian *gear*, potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran motor servo. Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Karena motor DC servo merupakan alat untuk mengubah energi listrik menjadi energi mekanik, maka magnet permanen motor DC servolah yang mengubah energi listrik ke dalam energi mekanik melalui interaksi dari dua medan magnet. Salah satu medan dihasilkan oleh magnet permanent dan yang satunya dihasilkan oleh arus yang mengalir dalam kumparan motor. Resultan dari dua medan magnet tersebut menghasilkan torsi yang membangkitkan putaran motor tersebut. Saat motor berputar, arus pada kumparan motor menghasilkan torsi yang nilainya konstan.

Kendali putaran motor servo dengan metode PWM (*Pulse Width Modulation*) dan

menggunakan Mikrokontroler *BASIC Stamp2SX* dengan bahasa pemrograman PBasic. Hasil penelitian yang penulis lakukan diketahui, bahwa: putaran motor servo searah dengan arah jarum jam, jika jumlah pulsa yang harus di-input-kan lebih kecil dari 1600 pulsa (1,6 ms pulsa). Sedangkan motor servo akan berputar berlawanan dengan arah jarum jam, jika jumlah pulsa yang di-input-kan lebih besar dari 1600 pulsa (1,6 ms pulsa).



Gambar 3.1: Motor Servo Standard

### 3.1 Bagian-bagian dari sebuah motor servo standar

Sebuah motor servo standar adalah suatu peralatan yang dapat mengendalikan posisi, dapat membelokkan dan menjaga suatu posisi berdasar penerimaan pada signal elektronik. Karena servo motor DC merupakan alat untuk mengubah energi listrik menjadi energi mekanik, maka magnet permanent servo motor DC yang mengubah energi listrik ke dalam energi mekanik melalui interaksi dari dua medan magnet.

Bagian-bagian dari sebuah motor servo standard adalah sebagai berikut:

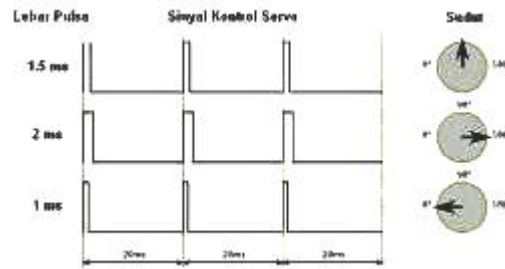
- a) Konektor adalah alat elektronik yang digunakan untuk menghubungkan motor servo DC dengan *Vcc*, *Ground* dan *signal* input yang dihubungkan ke mikrokontroler Basic Stamp.
- b) Kabel untuk menghubungkan *Vcc*, *Ground* dan *signal* sebagai input dari konektor ke motor servo ke mikrokontroler.

- c) Tuas menjadi bagian dari motor servo yang kelihatan seperti suatu bintang four-pointed. Ketika motor servo berputar, tuas motor servo akan bergerak ke bagian yang dikendalikan sesuai dengan program.
- d) *Cassing* berisi bagian untuk mengendalikan kerja motor servo yang pada dasarnya berupa motor DC dan *gear*. Bagian ini bekerja untuk menerima instruksi dari basic stamp dan mengkonversi ke dalam sebuah pulsa untuk menentukan arah/ posisi servo.

## 3.2 Tipe Motor Servo

Secara umum terdapat 2 jenis motor servo yang banyak digunakan dalam sistem robotik. yaitu motor servo standar dan *motor servo continous*. Motor servo tipe standar hanya mampu untuk berputar 180 derajat (setengah putaran). Motor servo standar yang sering digunakan pada sistim robotika, misalnya untuk membuat *Robot Arm (Robot Lengan)*. Sedangkan servo motor kontinu dapat berputar sebesar 360 derajat (satu putaran penuh). motor servo kontinu sering digunakan untuk *Mobile Robot*. Pada badan servo tertulis tipe servo yang bersangkutan. Pengendalian gerakan batang motor servo dapat dilakukan dengan menggunakan metode PWM (*Pulse Width Modulation*). Teknik ini menggunakan sistem lebar pulsa untuk mengemudikan putaran motor servo, agar dapat berputar sesuai yang kita kehendaki.

Sudut sumbu servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Tampak (Gambar 3.2.) dengan pulsa 1.5 ms pada periode selebar 2 ms, maka sudut dari sumbu motor servo akan berada pada posisi tengah. Semakin lebar pulsa *OFF* maka semakin besar gerakan sumbu searah jarum jam dan semakin kecil pulsa, maka semakin besar gerakan sumbu ke arah yang berlawanan jarum jam. Rangkaian kontrol motor servo digunakan untuk kendali motor DC pada motor servo tersebut, oleh sebab itu kendali motor servo cukup hanya dengan memberikan pulsa-pulsa pada rangkain kontrolernya. Menggerakkan servo searah jarum jam atau berlawanan arah jarum jam, bergantung nilai *delay* yang kita berikan ke mikrokontroler. Kendali putaran servo posisi *center*, maka diberikan lebar pulsa 1.5 ms. Untuk memutar motor servo ke kanan, berikan pulsa lebih kecil atau sama dengan 1.3 ms. Untuk pulsa lebih besar atau sama dengan 1.7 ms, maka servo berputar ke kiri dengan *delay* 20 ms, seperti ilustrasi gambar berikut:



Gambar 3.2: Pensinyalan motor servo

### 3.3 PWM (*Pulse Width Modulation*)

PWM (*Pulse Width Modulation*) adalah sebuah metode untuk pengaturan kecepatan perputaran, dalam hal ini adalah motor DC untuk menggerakkan robot. PWM dapat dihasilkan oleh empat metode, sebagai berikut :

- a. Metode analog
- b. Metode digital
- c. IC diskrit
- d. Mikrokontroler

Metode PWM dikerjakan oleh mikrokontroler. Metode PWM ini akan mengatur lebar atau sempitnya periode pulsa aktif yang dikirimkan oleh mikrokontroler ke *driver* motor. Pada pengaturan kecepatan robot, nilai PWM mulai dari 0-255. Secara analog besaran PWM dihitung dalam prosentase, nilai didapat dari perbandingan:  $T_{high} / (T_{high} + T_{low}) * 100\%$ . Dimana T adalah periode atau waktu tempuh untuk sebuah pulsa, yang terbagi menjadi bagian puncak positif ( $T_{high}$ ) dan puncak negatif ( $T_{low}$ ).

Semakin rapat periode antar pulsa, frekuensi yang dihasilkan akan semakin tinggi, ini berarti kecepatan akan bertambah. Semakin lebar jarak antar pulsa, maka frekuensi semakin rendah ini berarti kecepatan berkurang atau menurun. Kondisi pemberian kecepatan harus disesuaikan dengan kondisi *track* yang akan dilewati oleh robot, misal pada saat jalan lurus, naik atau turun harus mendapatkan nilai PWM yang tepat.

## 3.4 Modifikasi Motor Servo

Motor servo harus dimodifikasi terlebih dahulu, hal ini dikarenakan standar pabrik putaran servo hanya mencapai  $180^\circ$ . Oleh sebab itu motor servo dimodifikasi agar dapat mencapai putaran  $360^\circ$  (satu putaran penuh). Modifikasi motor servo dengan cara mengganti R variabel (potensiometer) dengan R tetap, yaitu 2,7 k sebanyak 2 buah dipasang seri dan menghilangkan bagian stop tab pada final gear servo. Sedangkan kabel konektor motor servo harus diubah urutan kabelnya, karena untuk menyesuaikan dengan urutan konektor pada pin I/O *main board*. Ada 3 kabel konektor dengan urutan ( GND), putih untuk *Sources Data* dan merah untuk *Vcc*.

### 3.4.1 Cara Modifikasi Komponen Motor Servo

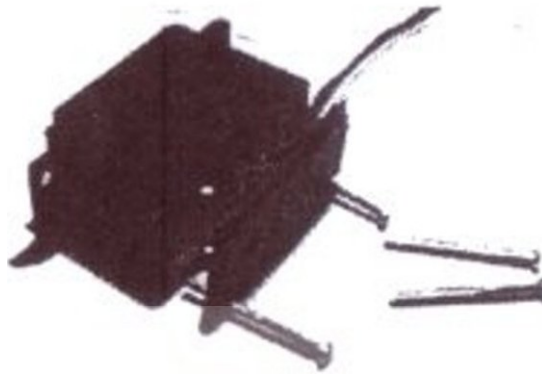
- 1) Siapkan perlengkapan/ *tools* yang diperlukan, seperti obeng, tang, pemotong, solder dan peralatan pendukung lainnya.(Gambar 3.3)



Gambar 3.3: Perlengkapan untuk modifikasi motor servo

- 2) Lepaskan keempat sekrup yang berada pada bagian bawah servo. Gunakan obeng (+), serta sedikit gunakan pengungkit untuk membuka *Case Bottom*. (Gambar 3.4)
- 3) Panaskan dan lepaskan 2 titik solder yang mengikat motor DC pada papan rangkaian, karena hal ini akan mempermudah dalam melepas papan rangkaian.(Gambar 3.5)
- 4) Setelah semua terlepas maka akan didapati 4 *gear* penggerak serta papan rangkaian yang terdiri dari potensiometer dan beberapa komponen lain.(Gambar 3.6)





Gambar 3.4: Membuka *Case Bottom*

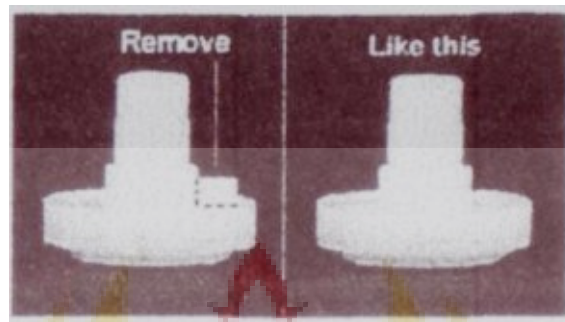


Gambar 3.5: Melepas papan rangkaian pada motor servo



Gambar 3.6: 4 *gear* penggerak

- 5) Perhatikan *Final Gear* (yang mempunyai gear tinggi), potong dan ratakan *Tab Stop* nya seperti terlihat di (Gambar 3.7). Karena *Tab Stop* ini yang akan membatasi gerak dari *gear* tersebut.



Gambar 3.7: *Tab Stop* pada *gear* penggerak

- 6) Setelah modifikasi pada *gear* telah dilakukan, lanjutkan modifikasi pada potensiometer. Potong ketiga kaki potensiometer pada ketinggian yang kira-kira mudah dijangkau. (Gambar 3.8)



Gambar 3.8: Memotong ketiga kaki potensiometer

- 7) Solder dan bentuk kedua resistor menjadi seperti pada (Gambar 3.9) gunakan selongsong kabel dan isolasi untuk mencegah hubungan singkat.
- 8) Sambungkan resistor pengganti dengan potongan potensiometer, posisi penyolder tidak terlalu berpengaruh (boleh terbalik) asalkan posisi kaki yang tengah tetap di tengah (Gambar 3.10).



Gambar 3.9: Menyolder dan membentuk kedua resistormenjadi satu



Gambar 3.10: Menghubungkan resistor pengganti dengan potongan potensiometer

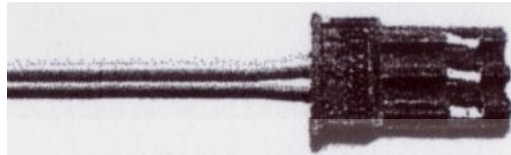
- 9) Atau bila Anda ingin lebih rapi pasang pasang kedua resistor seperti pada (Gambar 3.11). Pastikan pemasangan pada titik-titik solder bekas potensiometer yang benar.



Gambar 3.11: Pemasangan pada titik-titik solder bekas potensiometer

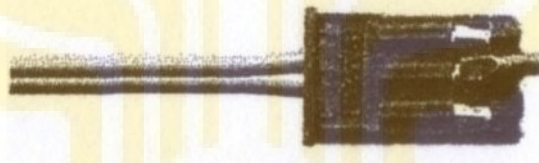
### 3.4.2 Cara Modifikasi Konektor Kabel Motor Servo

- 1) Perhatikan susunan tiga kabel pada konektor servo, secara standar pabrikasi susunannya seperti pada (Gambar 3.12). Hitam, Merah dan putih/ kuning.



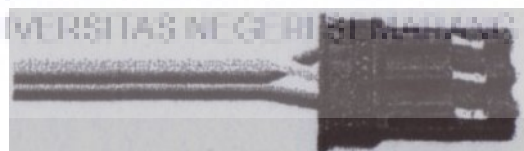
Gambar 3.12: Susunan kabel konektor servo pabrikasi standar

- 2) Ubah posisi kabel Merah dengan Putih, Gunakan pengungkit kecil (obeng minus kecil) untuk mengangkat pengait konektor, sambil tarik kabelnya dari sisi lain (Gambar 3.13)



Gambar 3.13: Mengganti posisi kabel Merah dengan Putih

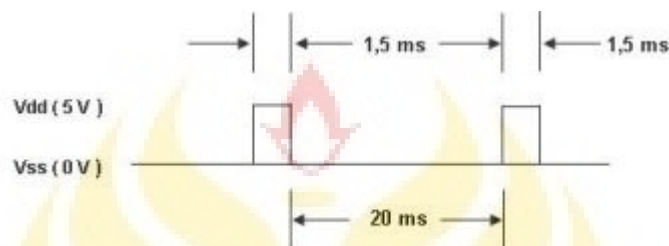
- 3) Ubah untuk kedua kabel, hingga susunannya menjadi Hitam, Putih/ Kuning kemudian Merah (Gambar 3.14).



Gambar 3.14: Susunan kabel Hitam, Putih/ Kuning dan Merah

### 3.5 Pemrograman cara kerja motor servo

Sistem kendali BASIC Stamp 2SX (BS2SX) dengan cara mengirimkan sejumlah deretan pulsa kotak ke motor servo, kemudian motor servo merespon. Deretan pulsa kotak mempunyai *high time* 1,5 ms dan *low time* 20 ms. *High time* seringkali dikaitkan dengan lebar pulsa positif dan mempunyai tegangan puncak 5 volt. Sedangkan *low time* berkaitan dengan dengan keadaan, dimana pulsa mempunyai mempunyai tegangan 0 volt.



Gambar 3.15: Deretan pulsa kotak

Perintah pemrograman PBASIC harus dimulai dengan kata yang ada di daftar tabel secara otomatis. Sebuah kata *loop* merupakan daftar kata bersama dengan perintah program *goto loop* pada akhir suatu program. Sebuah kata *loop* merupakan daftar petunjuk tempat dan sewaktu-waktu dapat memanggil program dengan perintah, misalnya: *loop: label*, hasilnya adalah *pulsout* dan *pause*. Perintah eksekusi dapat diambil berulang-ulang dengan mengirimkan *signal center* ke servo. Penggunaan perintah PBASIC dimulai dengan kata-kata yang ada di daftar tabel secara otomatis. Sebuah kata *loop* merupakan daftar kata bersama dengan perintah *goto loop* pada akhir suatu program. Sebuah kata *loop* juga merupakan daftar petunjuk tempat dan sewaktu-waktu dapat memanggil program dengan perintah, misalnya: *loop: label*, hasilnya adalah *pulsout* dan *pause*.

Kata *loop : .... goto* adalah struktur program yang dikatakan sebagai putaran tak terbatas. Putaran tak terbatas merupakan bagian dari code dalam program akhir dan dapat dimulai lagi tanpa instruksi yang sama. Putaran tak terbatas biasanya digunakan dalam mikrokontroler dalam satu teks, yang ditulis sampai dengan kerangka bagian dari putaran tak terbatas. Dengan BASIC Stamp2SX dapat diselesaikan masalah putaran tak terbatas tanpa dihubungkan ke *power* atau program yang berbeda dijalankan.

### 3.6 Pengujian Motor Servo

Untuk melakukan pengujian motor servo ini, dilakukan dengan cara mengisikan program terlebih dahulu kepada mikrokontroler. Program tersebut adalah program untuk mengaktifkan motor servo. Seperti diketahui bahwa motor servo diaktifkan dengan memberikan sinyal PWM (*Pulse Width Modulation*) dengan lebar pulsa tertentu. Sinyal inilah yang akan memberitahukan motor servo harus berputar CW (*Clock Wise* = searah jarum jam) atau CCW (*Counter Clock Wise* = berlawanan arah jarum jam). Putaran motor servo mempunyai 2 arah, yaitu searah dengan jarum jam dan berlawanan arah dengan jarum jam. Arah putaran motor servo ditentukan dengan cara uji coba-coba (*trial and error*) dari beberapa nilai *pulsout* (pulsa keluaran) dengan program, sebagai berikut:

```
low 12           set P12 to out low ( P12= pin 12)
loop :          label for goto loop to return to
pulsout 12, 1700 send 1,7 ms pulses
pause 500       every 500 ms
goto loop       send program to loop label
```

Pertama kali menentukan *pulsout*, agar motor servo dalam keadaan diam, hal dikatakan bahwa motor servo dalam kondisi titik tengah arah putaran. Kemudian berikan nilai *pulsout* yang lebih besar atau kecil, sehingga terlihat arah putaran motor servo. Amati arah putaran motor servo searah atau berlawanan dengan jarum jam.

### 3.7 Hasil Pengujian Motor Servo

Servo diam, jika pulsa yang diberikan adalah 1600 pulsa (memberikan 1,6 ms pulsa). Putaran servo searah arah jarum jam, maka pulsa yang harus di-input-kan lebih kecil 1600 pulsa (1,6 ms pulsa). Sedangkan servo berputar berlawanan dengan arah jarum jam, maka jumlah pulsa yang di-input-kan harus lebih besar dari 1600 pulsa. Dari hasil eksperimen dapat diaplikasikan untuk menggerakkan motor servo adalah sebagai berikut:

- 1). Gerakan maju, motor servo harus diberikan jumlah pulsa  $>1600$  pulsa (1,6 ms pulsa)
- 2). Gerakan mundur, motor servo diberikan jumlah pulsa  $< 1600$  pulsa (1,6 ms pulsa)

Hasil eksperimen dengan meng-input-kan pulsa ke mikrokontroler, sebagai berikut:

- a) Hubungan antara jumlah pulsa yang di-input-kan dengan jarak yang ditempuh oleh motor servo bergerak maju.

Tabel 3.1: Hubungan jumlah pulsa dengan jarak tempuh

Jumlah pulsa motor servo	Periode	Jumlah Putaran	Waktu (detik)	Kecepatan (cm/s)	Jarak yang ditempuh ditempuh
1750	1000	15,25	23,5	12,25	287,41
1750	500	7,75	12	12,17	146,01
1750	100	1,50	2,4	11,78	18,84
1750	50	0,75	1,3	10,87	14,13

- b) Hubungan antara jumlah pulsa yang di-input-kan dengan sudut putara motor servo.

Tabel 3.2: Hubungan jumlah pulsa dengan sudut putaran motor servo

Jumlah pulsa motor servo	Periode	Sudut putaran motor servo (derajat)	Keterangan
1600	100	0	Diam
1800	50	30	Berputar berlawanan jarum jam
2000	50	60	Berputar berlawanan jarum jam
1700	100	90	Berputar berlawanan jarum jam
1700	100	180	Berputar berlawanan jarum jam
2000	100	270	Berputar berlawanan jarum jam
1600	100	0	Diam

Pada (Tabel 3.1 dan Tabel 3.2) merupakan tabel pengujian menggunakan mikrokontroler BS2SX dengan meng-input-kan jumlah pulsa untuk menggerakkan servo. Dengan memahami eksperimen ini, maka dapat membantu mahasiswa dalam mengembangkan robotika. Kegiatan diatas mendorong berkembangnya robotika ditanah air terutama dikalangan mahasiswa maupun umum. Hal lain yang ikut membantu mengembangkan robot adalah terbitnya buku tentang robot itu sendiri.

## Bab 4

### LDR (*Light Dependent Resistor*)

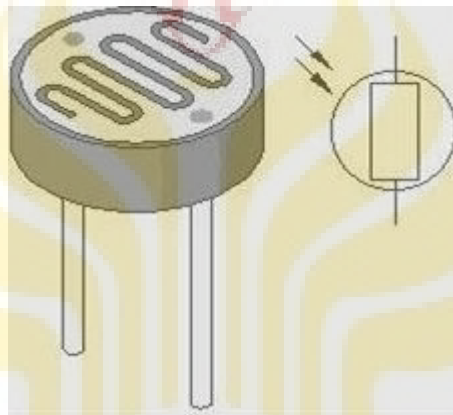
LDR atau *Light Dependent Resistor* adalah salah satu jenis resistor yang nilai hambatannya dipengaruhi oleh cahaya yang diterima olehnya. Besarnya nilai hambatan pada LDR tergantung pada besar kecilnya cahaya yang diterima oleh LDR itu sendiri. Contoh penggunaannya adalah pada lampu taman dan lampu di jalan yang bisa menyala di malam hari dan padam di siang hari secara otomatis. Buku ini akan membahas tentang sensor cahaya LDR (*Light Dependent Resistor*) merupakan jenis resistor yang dapat mengalami perubahan resistansinya apabila mengalami perubahan penerimaan cahaya.

Resistor peka cahaya atau fotoresistor adalah komponen elektronik yang resistansinya akan menurun jika ada penambahan intensitas cahaya yang mengenainya. Fotoresistor dapat merujuk pula pada *light-dependent resistor* (LDR), atau fotokonduktor. Fotoresistor dibuat dari semikonduktor beresistansi tinggi yang tidak dilindungi dari cahaya. Jika cahaya yang mengenainya memiliki frekuensi yang cukup tinggi, foton yang diserap oleh semikonduktor akan menyebabkan elektron memiliki energi yang cukup untuk meloncat ke pita konduksi. Elektron bebas yang dihasilkan (dan pasangan lubangnya) akan mengalirkan listrik, sehingga menurunkan resistansinya ini adalah aplikasi pada lampu taman dan lampu di jalan yang bisa menyala di malam hari dan padam di siang hari secara otomatis. Atau bisa juga kita gunakan di kamar kita sendiri.

Sensor Cahaya LDR (*Light Dependent Resistor*) adalah salah satu jenis resistor yang dapat mengalami perubahan resistansinya apabila mengalami perubahan penerimaan cahaya. Besarnya nilai hambatan pada Sensor Cahaya LDR (*Light Dependent Resistor*)



tergantung pada besar kecilnya cahaya yang diterima oleh LDR itu sendiri. LDR sering disebut dengan alat atau sensor yang berupa resistor yang peka terhadap cahaya. Biasanya LDR terbuat dari *cadmium sulfida* yaitu merupakan bahan semikonduktor yang resistansinya berubah menurut banyaknya cahaya (sinar) yang mengenainya. Resistansi LDR pada tempat yang gelap biasanya mencapai sekitar 10 M, dan ditempat terang LDR mempunyai resistansi yang turun menjadi sekitar 150 . Seperti halnya resistor konvensional, pemasangan LDR dalam suatu rangkaian sama persis seperti pemasangan resistor biasa. Simbol LDR dapat dilihat seperti pada gambar berikut.



Gambar 4.1: Sensor Cahaya LDR (*Light Dependent Resistor*)

Sensor Cahaya LDR (*Light Dependent Resistor*) dapat digunakan sebagai :

- 1) Sensor pada rangkaian saklar cahaya.
- 2) Sensor pada lampu otomatis.
- 3) Sensor pada alarm brankas.
- 4) Sensor pada tracker cahaya matahari.
- 5) Sensor yang diaplikasikan pada kontrol arah *solar cell sensor* pada *robot line follower* (robot pengikut lintasan tertentu).

Dan masih banyak lagi aplikasi rangkaian elektronika yang menggunakan LDR (*Light Dependent Resistor*) sebagai sensor cahaya.

## 4.1 Karakteristik Sensor Cahaya LDR

Sensor cahaya LDR (*Light Dependent Resistor*) adalah suatu bentuk komponen sensor yang mempunyai perubahan resistansi yang besarnya tergantung pada intensitas cahaya yang mengenainya. Biasanya LDR terbuat dari cadmium sulfida yaitu merupakan bahan semikonduktor yang resistansinya berubah-ubah menurut banyaknya cahaya (sinar) yang mengenainya. Resistansi LDR pada tempat yang gelap biasanya mencapai sekitar 10 M, dan ditempat terang LDR mempunyai resistansi yang turun menjadi sekitar 150 . Seperti halnya resistor konvensional, pemasangan LDR dalam suatu rangkaian sama persis seperti pemasangan resistor biasa. Karakteristik LDR terdiri dari dua macam yaitu laju *recovery* dan respon spektral sebagai berikut:

### 4.1.1 Laju Recovery Sensor Cahaya LDR

Bila sebuah Sensor Cahaya LDR (*Light Dependent Resistor*) dibawa dari suatu ruangan terang dengan level kekuatan intensitas cahaya tertentu ke dalam suatu ruangan yang gelap, maka bisa kita amati bahwa nilai resistansi dari LDR tidak akan segera berubah resistansinya pada keadaan ruangan gelap tersebut. Namun LDR tersebut hanya akan bisa mencapai harga di kegelapan setelah mengalami selang waktu tertentu. Laju *recovery* merupakan suatu ukuran praktis dan suatu kenaikan nilai resistansi dalam waktu tertentu. Harga ini ditulis dalam K/detik, untuk LDR tipe arus harganya lebih besar dari 200K/detik (selama 20 menit pertama mulai dari level cahaya 100 lux), kecepatan tersebut akan lebih tinggi pada arah sebaliknya, yaitu perpindahan dari tempat gelap ke tempat terang memerlukan waktu kurang dari 10 ms untuk mencapai resistansi sesuai dengan level cahaya 400 lux.

### 4.1.2 Respon Spektral Sensor Cahaya LDR

Sensor Cahaya LDR (*Light Dependent Resistor*) tidak mempunyai sensitivitas yang sama untuk setiap panjang gelombang cahaya yang jatuh padanya (yaitu warna). Bahan yang biasa digunakan sebagai penghantar arus listrik yaitu tembaga, aluminium, baja, emas dan perak. Dari kelima bahan tersebut tembaga merupakan penghantar yang paling banyak digunakan karena mempunyai daya hantar yang baik. Sensor ini sebagai pengindera yang merupakan elemen yang pertama menerima energi dari media untuk mem-

berikan keluaran berupa perubahan energi. Sensor terdiri dari berbagai macam jenis serta media yang digunakan untuk melakukan perubahan. Media yang digunakan misalnya : panas, cahaya, air, angin, tekanan, dan lain sebagainya. Sedangkan pada rangkaian ini menggunakan sensor LDR yang menggunakan intensitas cahaya, selain LDR dioda foto juga menggunakan intensitas cahaya atau yang peka terhadap cahaya (*photo conductive cell*). Pada rangkaian elektronika, sensor harus dapat mengubah bentuk energi cahaya ke energi listrik, sinyal listrik ini harus sebanding dengan besar energi sumbernya.

### 4.1.3 Prinsip Kerja Sensor Cahaya LDR

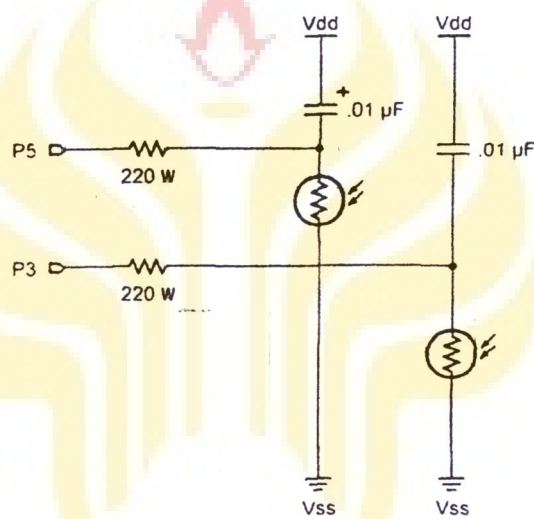
Resistansi Sensor Cahaya LDR (*Light Dependent Resistor*) akan berubah seiring dengan perubahan intensitas cahaya yang mengenainya atau yang ada disekitarnya. Dalam keadaan gelap resistansi LDR sekitar 10 M dan dalam keadaan terang sebesar 1K atau kurang. LDR terbuat dari bahan semikonduktor seperti kadmium sulfida. Dengan bahan ini energi dari cahaya yang jatuh menyebabkan lebih banyak muatan yang dilepas atau arus listrik meningkat. Artinya resistansi bahan telah mengalami penurunan.

### 4.1.4 Kendali Sensor LDR dengan Mikrokontroler

Untuk mengendalikan LDR agar sesuai dengan yang kita butuhkan, maka diperlukan mikrokontroler dengan beberapa *port* yang dapat digunakan. *Port* tersebut antara lain: satu *port* dihubungkan peralatan yang akan dikendalikan dan satu *port* lagi dihubungkan ke sensor LDR. *Port* ini hanya bisa mendeteksi data digital, peralatan digunakan *ON* pada saat diberikan nilai digital 0, maka peralatan yang akan dikendalikan dalam kondisi *ON* (*port* tersebut bernilai 0). Peralatan pada kondisi *OFF* maka *port* tersebut bernilai 1.

Aplikasi kendali/pengontrolan LDR dengan menggunakan mikrokontroler *Basic Stamp BS2SX* dalam buku ini, penulis memaparkan dari hasil penelitian ketika menyelesaikan program pascasarjana S2 Teknik Elektro di UGM. Penelitian yang penulis lakukan menggunakan *photoresistor* yang dihubungkan dengan mikrokontroler *Basic Stamp BS2X*. *Photoresistor* dapat digunakan untuk mendeteksi bermacam-macam tingkatan intensitas cahaya. Dengan mikrokontroler *Basic Stamp BS2X* dan program *Pbasic* dapat merubah *photoresistor* ke bentuk *photophile* (mengikuti cahaya) *photophobe* (menjauhi cahaya). Untuk keperluan riset digunakan dua buah LDR yang dipasang pada bagian bawah robot

beroda. Penggunaan LDR dalam riset penulis bukan untuk pengukuran intensitas cahaya, melainkan membandingkan intensitas cahaya yang diterima LDR. Apabila intensitas yang diterima LDR kanan dan kiri berbeda, maka robot beroda (mobil robot) akan memberikan respon gerakan tertentu sesuai dengan perintah melalui program. LDR mempunyai karakteristik dimana nilai resistansinya yang tinggi ketika tidak terkena cahaya dan nilai resistansinya akan turun ketika LDR terkena cahaya. Besarnya penurunan nilai resistansi LDR juga dipengaruhi oleh intensitas cahaya yang masuk ke LDR, semakin tinggi intensitas cahayanya maka semakin rendah pula nilai resistansinya.



Gambar 4.2: Rangkaian dua LDR dengan RC

Rangkaian dua LDR dengan RC yang ditunjukkan pada (Gambar 1.2) dirancang untuk digunakan dalam perintah pemrograman PBasic (parallax Basic). Perintah ini dapat digunakan dalam rangkaian RC, dimana satu nilai manapun R (Resistor) atau C (Capasitor) mempunyai nilai yang konstan (tetap). Perintah unutu waktu RC digunakan untuk mengukur variabel nilai, sebab mempuunyai keuntungan dari suatu waktu yang bermacam-macam rangkaian RC.

Tegangan pada rangkaian RC berubah bergantung pada hasil perkalian R dan C. RC adalah tetapan yang mempunyai nilai konstan dan ditandai dengan simbol tau ( $\tau_{RC}$ ). Untuk salah satu rangkaian RC yang ditunjukkan pada (Gambar 1.2), langkah pertama

menentukan pengukuran waktu RC pada saat plat kapasitor terbebani tegangan yang lebih rendah dari 5 V. Selanjutnya hubungkan konektor I/O untuk menurunkan tegangan plat kapasitor dengan resistor 220 ohm, memerlukan waktu beberapa ms.

Perintah *RCtime* dapat digunakan untuk mengukur waktu terjadinya perubahan tegangan pada kapasitor dari 5 V menjadi 1,4 V. Tegangan 1,4 V adalah tegangan ambang pada Basic Stamp. Ketika tegangan I/O diatas 1,4 V nilai pada input bit register adalah 1. Ketika tegangan dibawah 1,4 V, nilai bit register masukan adalah 0. Perintah pada rangkaian waktu RC ditunjukkan pada (Gambar 1.2), untuk mengukur berapa lama tegangan di plat kapasitor akan turun 5 V sampai 1,4 V. Waktu RC bervariasi menurut persamaan:

$$\begin{aligned} \frac{t}{RC} &= \ln\left(\frac{V_{initial}}{V_{final}}\right) \\ \frac{t}{R(0,001).10^{-6}} &= \ln\left(\frac{5V}{1,4V}\right)s \\ t &= \ln(3,57).R.(0,01.10^{-6})s \\ t &= (1,27.10^{-8})(R)s \end{aligned} \tag{4.1}$$

Pada persamaan (1.1.) mengindikasikan bahwa waktu yang diperlukan oleh salah satu pelat kapasitor (Gambar 1.2) untuk mencapai tegangan terendah. Pada rangkaian RC tersebut tegangan turun dari 5 volt menjadi 1,4 volt secara proporsional sesuai dengan suatu nilai resistansi dari photoresistor. Perintah pada pemrograman *RCtime* akan mengubah pin I/O dari *output* ke *input*. Begitu pula dengan pin I/O menjadi masukan, tegangan pelat kapasitor menjadi tegangan terendah menurut persamaan (1.1) yang dibahas di atas. Basic Stamp mulai menghitung dalam 2  $\mu$ s sampai dengan tegangan kapasitor menjadi lebih rendah 1.4 V.

Program dari perintah variabel *RCtime* yang memanfaatkan suatu terminal *debug* dengan kabel serial dihubungkan dengan mikrokontroler *Basic Stamp* pada saat listing program sedang berjalan.

Program Listing ” Menampilkan *RCtime* Photoresistor ”

```
..... Declarations .....
```

```
left_photo var word  
right_photo var word
```

```
..... Initialization .....
```

```
debug cls
```

```
..... Main Routine .....
```

```
main:
```

```
’ Mengukur RCtime untuk photoresistor kanan. high 3  
pause 3  
rctime 5, 1 left_photo
```

```
’ Penampilan pengukuran RC time menggunakan Terminal Debug ’  
debug home ”L” , dec5 left_photo, ” R ”, dec5 right_photo  
goto main
```

#### 4.1.5 Bagaimana Cara Menampilkan Nilai *RCtime* Photoresistor

Variabel *word*, adalah : *left\_photo* dan *right\_photo* yang merupakan deklarasi penyimpanan nilai dari *RCtime* untuk *left* (kiri) dan *right* (kanan) photoresistor. Perintah dalam pemrograman *PBasic* , tulisan *Main Routine* untuk mengukur dan menampilkan nilai *RC* time dari setiap rangkaian. Kode untuk membaca rangkaian *RCtime* kanan, sebagai berikut: Pertama, pin I/O P3 dihubungkan pada keluaran yang tinggi. Kedua, pulsa 3 ms yang diperlukan pengosongan kapasitor. Ketiga, setelah 3 ms kapasitor mempunyai tegangan keluaran rendah karena adanya pengosongan. Kemudian kapasitor terjadi pengisian lagi menjadi keluaran tegangan 5 V, ini merupakan langkah pengukuran *RCtime*.

Program perintah untuk pengukuran RCtime pada pin I/O P3, dengan mulai dari kondisi tegangan 1 V atau 5 V dan hasil tersimpan dalam variabel `right_photo` (photoresistor kanan). Ingat, bahwa nilai yang tersimpan pada `right_photo` (photoresistor kanan) adalah berupa bilangan (angka). Angka menerangkan penambahan 2  $\mu$ s menuju pada tegangan rendah kapasitor. Pin I/O kapasitor pada kondisi tegangan rendah (1,4 V) dan merupakan tegangan ambang (*threshold*).

```
high 3
pause
rctime 3, 1 right_photo
```

## 4.2 Aplikasi LDR (*Light Dependent Resistor*)

Aplikasi LDR dimanfaatkan penulis dalam penelitian digunakan dua buah LDR yang dipasang pada mobil robot. Penggunaan 2 buah LDR dalam riset penulis bukan untuk mengukur intensitas cahaya yang ada disekitar mobil robot, melainkan membandingkan intensitas yang diterima oleh LDR. Apabila intensitas yang diterima LDR kanan dan LDR kiri berbeda, maka mobil robot akan memberikan respon gerakan tertentu sesuai dengan perintah melalui program. Contoh aplikasi LDR (sensor cahaya) yang dipasang pada mobil robot untuk memberikan respon gerakan tertentu dengan perintah melalui pemrograman Pbasic.

Program. Gerakan mobil robot (robot beroda) dengan menggunakan 2 buah LDR.

---

*' Define Variables and Constanta*

---

*right\_servo*

*Con 12*

*' P12 sebagai servo kanan*

<i>left_servo</i>	<i>Con 13</i>	<i>' P12 sebagai servo kiri</i>
<i>turnvalue</i>	<i>Con 6</i>	
<i>scale</i>	<i>Con 100</i>	
<i>speed</i>	<i>Con 100</i>	
<i>speaker</i>	<i>Con 2</i>	
<i>rightLDR</i>	<i>Var word</i>	
<i>leftLDR</i>	<i>Var word</i>	
<i>x</i>	<i>Var word</i>	
<i>white</i>	<i>Con 1</i>	
<i>black</i>	<i>Con 0</i>	

---

*' Main Program*

*forward :*

*for x = 1 to 6*

*Pulsout left\_servo, 1600 + speed*

*Pulsout right\_servo, 1650 + speed*

*pause 20 next*

*right\_light*

*high 3*

*pause 10*

*rctime 3,1,rightLDR*

*debug " R ",dec rightLDR*

*freqout speaker,100,rightLDR\*scale +100*

*left\_light*

*high 5*

*pause 10*

*rctime 5,1,rightLDR*

*debug " L ",dec rightLDR*

*freqout speaker,100,leftLDR\*scale +10*

*if rightLDR = white and leftLDR = black then left*



```

if leftLDR = white and rightLDR = black then right
if leftLDR = white and rightLDR = black then backward
right :
high 0
for x = 1 to turnvalue
Pulsout left_servo, 1600 + speed
Pulsout right_servo, 1650 + speed
pause 20

next

low 0
left :
high 0
for x = 1 to turnvalue
Pulsout left_servo, 1600 + speed
Pulsout right_servo, 1650 + speed
pause 20

next

low 0
goto forward
backward :
high 0
for x = 1 to turnvalue
Pulsout left_servo, 1600 + speed
Pulsout right_servo, 1650 + speed
pause 20

next

low 0
goto forward

```

## Bab 5

# Sensor Gelombang Ultrasonik

Sensor gelombang ultrasonik adalah sensor yang mengubah besaran fisis (bunyi) menjadi listrik. Sensor ini gelombang ultrasonik dibangkitkan melalui benda yang disebut piezoelektrik. Piezoelektrik ini akan menghasilkan gelombang ultrasonik dengan frekuensi 40 kHz ketika sebuah osilator diterapkan pada benda tersebut. Sensor ultrasonik secara umum digunakan untuk suatu pengungkapan tak sentuh yang beragam seperti aplikasi pengukuran jarak. Alat ini secara umum memancarkan gelombang suara ultrasonik menuju suatu target yang memantulkan balik gelombang kearah sensor. Kemudian sistem mengukur waktu yang diperlukan untuk pemancaran gelombang sampai kembali kesensor dan menghitung jarak target dengan menggunakan kecepatan suara dalam medium. Rangkaian penyusun sensor gelombang ultrasonik ini terdiri dari *transmitter*, *reiceiver*, dan komparator. Selain itu, gelombang ultrasonik dibangkitkan oleh sebuah kristal tipis bersifat piezoelektrik. Bagian-bagian dari sensor ultrasonik adalah sebagai berikut:

1. Piezoelektrik adalah peralatan berfungsi untuk mengubah energi listrik menjadi mekanik secara langsung. Tegangan input digunakan menyebabkan bagian keramik meregang dan memancarkan gelombang ultrasonik. Tipe operasi transmisi elemen piezoelektrik berkisar frekuensi 32 kHz. Efisiensi lebih baik, jika frekuensi osilator diatur pada frekuensi resonansi piezoelektrik dengan sensitifitas dan efisiensi paling baik. Jika rangkaian pengukur beroperasi pada *mode* pulsa elemen piezoelektrik yang sama dapat digunakan sebagai transmitter dan reiceiver. Frekuensi yang ditimbulkan tergantung osilatornya yang disesuaikan frekuensi kerja masing-masing transduser. Kelebihannya inilah maka tranduser piezoelektrik lebih sesuai digunakan untuk sensor ultrasonik.

2. *Transmitter* (pemancar gelombang ultrasonik) Transmitter adalah sebuah alat yang berfungsi sebagai pemancar gelombang ultrasonik dengan frekuensi sebesar 40 kHz yang dibangkitkan dari sebuah osilator. Untuk menghasilkan frekuensi 40 KHz, harus di buat sebuah rangkaian osilator dan keluaran dari osilator dilanjutkan menuju penguat sinyal. Besarnya frekuensi ditentukan oleh komponen kalang RLC / kristal tergantung dari disain osilator yang digunakan. Penguat sinyal memberikan sebuah sinyal listrik yang diumpankan ke piezoelektrik dan terjadi reaksi mekanik, sehingga bergetar dan memancarkan gelombang yang sesuai dengan frekuensi pada osilator.
3. *Receiver* (penerima gelombang ultrasonik) Receiver terdiri dari transduser ultrasonik menggunakan bahan piezoelektrik, yang berfungsi sebagai penerima gelombang pantulan yang berasal dari *transmitter* yang dikenakan pada permukaan suatu benda atau gelombang langsung LOS (*Line of Sight*) dari *transmitter*. Bahan piezoelektrik memiliki reaksi *reversible*, elemen keramik membangkitkan tegangan listrik saat gelombang datang dengan frekuensi yang resonan dan akan menggetarkan bahan piezoelektrik.

Dunia robotika pada saat ini berkembang dengan sangat pesat, misalnya saja robot Asimo produksi Honda yang dapat bergerak menyerupai manusia. Sekarang semua pun berlomba-lomba untuk menciptakan robot yang lebih baik yang dapat bekerja membantu manusia. Untuk mengendalikan sebuah robot ditanamkan sebuah mikrokontroler atau mikroprosesor sebagai otak dari robot tersebut. Seiring tugas dan fungsi robot tersebut, fungsi dari mikrokontroler pun akan semakin rumit. Semakin maraknya pengembangan sistem robotik di dalam dan di luar negeri, kami justru melihat bahwa penelitian mengenai robotik sangat penting dan perlu di kaji secara mendalam. Pada sisi yang lain kurangnya penelitian mengenai robot yang memadai dan pengembangan lebih lanjut. Sebagai peneliti yang tertarik bidang robot beroda, maka perlu ada pihak yang memulai untuk mengembangkannya atau mengangkat topik penelitian tentang sistem robotik.

Masalah yang muncul pada sistem robotik ini adalah bagaimana cara merakit dan memasang sensor yang tepat, sehingga robot dapat melakukan gerakan sesuai dengan perintah. Pemasangan sensor sebagai pengindera robot memerlukan teknik tersendiri dengan memperhatikan karakteristik, posisi sensor dan objek yang dideteksi. Sebagai contoh pemasangan sensor gelombang ultrasonik pada robot beroda harus mengetahui spesifikasi sensor, meletakkan sensor dan posisi objek yang dideteksi.

Salah satu alasan penggunaan gelombang ultrasonik adalah mempunyai panjang gelombang yang pendek, difraksi lebih kecil dan berkas gelombang lebih tidak menyebar, sehingga benda yang lebih kecil dapat terdeteksi. Jika panjang gelombang ultrasonik lebih kecil dari ukuran benda yang menghalangi rambat gelombang, maka benda akan memantulkan bagian yang cukup besar dari gelombang. Benda yang berukuran paling kecil yang dapat dideteksi adalah dalam orde panjang gelombang yang digunakan. Dengan memanfaatkan frekuensi gelombang ultrasonik yang lebih tinggi, maka benda yang berukuran lebih kecil dapat terdeteksi.

Disamping itu dalam memilih jenis mikrokontroler yang akan digunakan sering kurang tepat, sehingga mengalami kesulitan dalam menyusun program pengendalinya. Untuk mengatasi hal demikian dapat digunakan mikrokontroler Basic Stamp dengan bahasa pemrograman Pbasic yang relatif sederhana, bahkan dapat menghindari algoritma rumit. Mikrokontroler ini keluarannya sudah dalam bentuk data digital, sehingga tidak lagi memerlukan interface. Dalam mikrokontroler juga dilengkapi dengan interpreter yang secara otomatis menunjuk register tertentu, dengan demikian akan memudahkan pengguna. Perintah dengan bahasa sederhana dan ditulis langsung ke bentuk desimal, tidak perlu mengubahnya ke bentuk biner.

Buku ini diharapkan sebagai pemula untuk meneliti dan mengembangkan lebih lanjut tentang sistem robotik, dimana belum dikembangkan dan diteliti secara intensif. Disamping itu pembaca dapat mengembangkan sistem robotik dengan merancang berbagai macam bentuk robot sesuai dengan kebutuhan dan implementasinya. Secara umum sensor gelombang ultrasonik digunakan untuk menghitung jarak dari suatu objek yang berada didepan sensor tersebut. Sehingga dengan fungsinya tersebut, sensor ultrasonik biasa digunakan pada perangkat yang membutuhkan perhitungan jarak. Contoh: smart robot, Kapal laut, kapal selam, mobil robot dan lain-lainnya.

## 5.1 Gelombang Ultrasonik

Dalam buku ini akan dibahas sumber gelombang ultrasonik (sebagai pemancar) dan penerima gelombang ultrasonik sebagai hasil riset penulis dalam menyelesaikan tugas akhir di Program studi Teknik Elektro S2 UGM. Energi dari sumber gelombang ultrasonik ada-

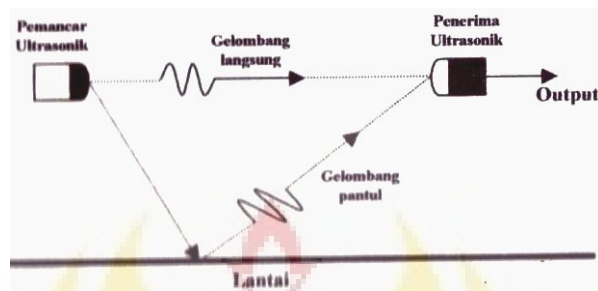
lah sama di semua area, tetapi intensitas yang diterima penerima gelombang ultrasonik bergantung pada jarak atau luasan antara pemancar dan penerima. Semakin jauh jarak antara pemancar dan penerima gelombang ultrasonik, semakin berkurang atau melemah intensitas gelombang yang diterimanya. Oleh karena itu jarak antara pemancar dan penerima gelombang ultrasonik harus diperhitungkan dalam memasang sensor gelombang ultrasonik. Disamping itu sudut gelombang ultrasonik pantul harus dipertimbangkan agar dapat diterima secara tepat oleh transduser, dengan demikian penerima dapat merespon gelombang pantul tersebut. Dengan mempertimbangkan jarak dan sudut pantul gelombang ultrasonik akan didapatkan hasil yang optimal.

Gelombang ultrasonik merupakan gelombang mekanik yang menjalar secara longitudinal dan mempunyai frekuensi di atas 20 kHz. Gelombang ultrasonik dapat merambat dalam medium padat, cair dan gas, hal tersebut disebabkan karena gelombang ultrasonik merupakan rambatan energi dan momentum mekanik, sehingga merambat sebagai interaksi dengan molekul dan sifat elastisitas medium yang dilaluinya. Karakteristik gelombang ultrasonik yang melalui medium mengakibatkan getaran partikel dengan medium amplitudo sejajar dengan arah rambat secara longitudinal, sehingga menyebabkan partikel medium membentuk rapatan (*strain*) dan tegangan (*stress*). Proses kontinu yang menyebabkan terjadinya rapatan dan regangan di dalam medium disebabkan oleh getaran partikel secara periodik selama gelombang ultrasonik melaluinya. Gelombang ultrasonik ini merambat di udara dengan kecepatan 344 meter per detik, mengenai objek dan memantul kembali ke sensor

Tingkat intensitas suara yang dapat menyakitkan telinga berkisar 30 dB ( $I = 1000 I_0$  dengan  $I_0 = 1 \cdot 10^{-6} \text{ W cm}^{-2} = 10 \text{ erg cm}^{-2} \text{ detik}$ ). Selain pembatasan respon terhadap intensitas gelombang bunyi, telinga manusia yang normal juga hanya mempunyai respon frekuensi yang terbatas antara 20 Hz sampai dengan 20.000 Hz. Namun demikian elastis di luar kurun waktu frekuensi itupun lazim dianggap sebagai gelombang bunyi.

Khusus gelombang elastis yang mempunyai frekuensi dengan frekuensi diatas 20.000 Hz disebut gelombang ultrasonik. Ilmu yang mempelajari tentang cara pembuatan sumber bunyi, perambatan dan pengolahannya dalam medium tertentu disebut akustika (*acoustics*) ( M.O., Tjia, 1993). Untuk memudahkan analisis gelombang akustik, maka

perlu diperhatikan adalah arah gelombang yang menuju *receiver* (penerima) dalam satu dimensi. Permukaan gelombang yang menjalar/merambat digambarkan dalam satu dimensi sebagaimana ditunjukkan pada (Gambar 2.1.)



Gambar 5.1: Detektor ultrasonik (Joseph J, Corr, 1993)

Deteksi objek dapat digunakan gelombang akustik dengan frekuensi antara 20 kHz - 60 kHz. Jarak detektor gelombang ultrasonik menggunakan transduser untuk membangkitkan gelombang akustik dan memancarkan ke segala arah. Transduser penerima dimanfaatkan untuk menerima gelombang akustik yang langsung maupun dipantulkan.

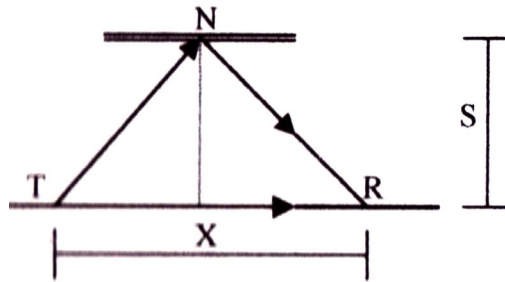
## 5.2 Sensor gelombang ultrasonik

Prinsip kerja sensor ultrasonik adalah adanya pemancar (*transmitter*) dan penerima (*receiver*) gelombang ultrasonik. Pemancar memancarkan gelombang ultrasonik ke arah objek, kemudian objek memantulkan gelombang dan diterima oleh penerima. *Transmitter* ultrasonik yang membentuk muka (*front*) gelombang menjalar ke segala arah secara radial. Setelah gelombang menjalar dan menabrak objek/dinding terjadilah (*front*) gelombang baru yang diakibatkan adanya pantulan. *Front* gelombang baru yang terbentuk akibat pantulan objek menjalar hingga mengenai *R*. Agar lebih jelas letak antara *T* dan *R*, maka dibuat skema sebagaimana Gambar 2.2.

Keterangan:

T = *transmitter* (pemancar gelombang ultrasonik)

R = *receiver* (penerima gelombang ultrasonik)



Gambar 5.2: Arah rambat gelombang

$X$  = jarak antara *transmitter* (T) dan *receiver* (R)

$N$  = objek yang dideteksi dan  $S$  = jarak antara objek dengan *receiver* (R)

Pada (Gambar 2.2) terlihat bahwa terjadi dua (2) fenomena gelombang yang diakibatkan oleh sumber gelombang yang menjalar ke segala arah. Pertama gelombang pantul menjalar dengan lintasan TNR. Kedua gelombang langsung yang menjalar pada dinding dengan lintasan TR. Dari peristiwa 2 fenomena gelombang yang diharapkan terjadi adalah peristiwa pertama, yaitu gelombang pantul yang dapat diterima oleh penerima sehingga dapat merespon dengan baik objek yang terdeteksi.

Peristiwa fenomena gelombang kedua tidak diharapkan karena gelombang ultrasonik dari pemancar belum mengenai sasaran ke objek sudah terdeteksi. Untuk mengeliminir agar gelombang yang dipancarkan sumber gelombang ultrasonik tidak diteruskan/ langsung ke penerima, maka perlu dipasang *loop* pada pemancar dan penerima. Jika hal ini tidak dilakukan akan menyebabkan sensor mendeteksi walaupun tidak adanya objek. Sebagai contoh: sensor ultrasonik dipasang pada mobil robot, jika tidak dipasang *loop* mobil akan membelok walaupun tidak penghalang di depannya. Hal ini disebabkan karena sensor merespon gelombang ultrasonik yang diteruskan, bukan gelombang pantul dari objek.

Hasil penghitungan waktu yang diperlukan sensor gelombang ultrasonik untuk menerima pantulan pada jarak tertentu. Perhitungan ini didapat dari rumus berikut:

$$S = \frac{t.v}{2}$$

Dimana :

S = Jarak antara sensor gelombang ultrasonik dengan objek yang dideteksi

v = Cepat rambat gelombang ultrasonik di udara

t = Selisih waktu pemancaran dan penerimaan pantulan gelombang ultrasonik

Sebelum sensor gelombang ultrasonik digunakan dan dipasang *loop* maka perlu adanya pengujian panjang *loop* dan jarak objek yang terdeteksi. Dengan mengetahui pengaruh panjang *loop* terhadap jarak objek yang dapat dideteksi oleh sensor gelombang ultrasonik akan mempermudah dalam menyusun program dalam mengaplikasikan program.

### 5.3 Panjang gelombang ultrasonik dan frekuensi

Hasil pengukuran frekuensi gelombang ultrasonik yang digunakan pada penelitian penulis adalah 38 kHz.  $f = 50 \text{ kHz}$  dan  $T = 20 \cdot 10^{-6}$

Kecepatan rambat gelombang bunyi (v) adalah 332 m/s (M.O. Tjia, 1992) sehingga panjang gelombang ( $\lambda$ ) dapat dihitung.

$$\lambda = \frac{v}{f} = \frac{332}{50 \cdot 10^3} \text{ meter}$$

Jadi  $\lambda = 6,64 \cdot 10^{-3} \text{ meter} = 6,64 \text{ mm}$ . ( $\lambda$ ) ini akan menentukan ukuran objek terkecil yang dapat dideteksi oleh sensor ultrasonik. Semakin besar frekuensi gelombang ultrasonik yang digunakan semakin kecil objek yang dapat dideteksi. Sedangkan jarak R dan T gelombang ultrasonik menentukan jarak terjauh objek yang dapat dideteksi oleh sensor, sebagaimana diungkapkan pada persamaan, sebagai berikut:

$$I = \frac{\text{Daya}}{\text{Luas}} = \frac{P}{4\pi r^2}$$

Semakin jauh jarak antara penerima (R) dan pemancar (T), intensitas yang diterima penerima gelombang ultrasonik semakin kecil sehingga mempengaruhi kepekaan penerima.



## 5.4 Pengujian sensor gelombang ultrasonik

Pengujian ini bertujuan untuk meneliti pengaruh jarak antara R ke T terhadap jarak objek yang dapat terdeteksi. Disamping untuk meneliti jarak antara R ke T terhadap kepekaan sensor, meneliti posisi sensor yang tepat.

### 5.4.1 Pengujian variabel jarak R ke T dipasang secara horisontal

Tabel 5.1: Hasil pengujian variabel jarak R dan T dipasang secara horisontal

No.	Jarak antara R dan T	Jarak maksimum objek terdeteksi (cm)	Posisi Objek (derajat)
1	0,5	36	90
		27	60
		9	45
		7,5	30
		Tak terdeteksi	Kurang dari 30
2	1,5	21	90
		8	60
		5	45
		Tak terdeteksi	Kurang dari 30
		Tak terdeteksi	Kurang dari 30
3	2,5	19	90
		7	60
		Tak terdeteksi	Kurang dari 45
		Tak terdeteksi	Kurang dari 45
		Tak terdeteksi	Kurang dari 45

### 5.4.2 Pengujian jarak antara R dan T dipasang secara vertikal

Tabel 5.2: Hasil pengujian variabel jarak R dan T dipasang secara vertikal

No.	Jarak antara R dan T	Jarak maksimum objek terdeteksi (cm)	Posisi Objek (derajat)
1	0,5	30	90
		13,5	60
		9	45
		7,5	30
		Tak terdeteksi	Kurang dari 30
2	1,5	30	90
		5,5	60
		Tak terdeteksi	kuran dari 60
		Tak terdeteksi	kuran dari 60
		Tak terdeteksi	kuran dari 60
3	2,5	25	90
		Tak terdeteksi	Kurang dari 90

### 5.4.3 Pengujian variabel panjang *loop* dipasang secara horisontal

Tabel 5.3: Hasil pengujian variabel panjang *loop* dipasang secara horisontal

No.	Panjang loop (cm)	Jarak antara T dan R (cm)	Jarak maksimum objek terdeteksi	Jarak maksimum objek terdeteksi (cm)
1	2	3,5	47	2,5
2	2,5	3,5	51	3
3	3	3,5	57	3,5
4	3,5	3,5	70	4

Pengujian ini bertujuan meneliti pengaruh jarak antara penerima ke pemancar terhadap objek yang dapat terdeteksi oleh sensor. Posisi objek diukur dari arah penerima ke objek. Hasil pengujian pada (Tabel 2.1.) dan (Tabel 2.2.) Pada (Tabel 2.3.) dan (Tabel 2.4.) hasil pengujian untuk mengamati perbedaan pemasangan sensor secara vertikal dan horisontal terhadap objek.

### 5.4.4 Pengujian variabel panjang *loop* dipasang secara vertikal

Tabel 5.4: Hasil pengujian variabel panjang *loop* dipasang secara vertikal

No.	Panjang loop (cm)	Jarak antara T dan R (cm)	Jarak maksimum objek terdeteksi (cm)	Jarak minimum objek terdeteksi (cm)
1	2	3,5	25	2,5
2	2,5	3,5	27	3
3	3	3,5	28	3,5
4	3,5	3,5	28,5	4

Mendeteksi objek dapat menggunakan gelombang ultrasonik dengan frekuensi antara 20 kHz sampai dengan 60 kHz. Jarak detektor gelombang ultrasonik digunakan transduser untuk membangkitkan gelombang dan memancarkan ke segala arah. Transduser penerima dimanfaatkan untuk menerima gelombang ultrasonik yang langsung maupun dipantulkan.

Penerima diletakkan beberapa *inchi* dari pemancar ultrasonik, agar terdeteksi dengan baik. Pengujian panjang *loop* dibuat bervariasi yang mempunyai panjang dari 2 cm sampai dengan 3,5 cm. Semakin panjang loop yang digunakan objek yang dapat terdeteksi oleh sensor semakin jauh. Pada pengujian ini digunakan ukuran objek dan jarak antara pemancar dan penerima gelombang ultrasonik sama. Dari hasil pengujian didapatkan hasil sebagai berikut:

- 1) Ukuran objek terkecil yang dapat dideteksi oleh sensor gelombang ultrasonik adalah 7 mm pada jarak 8 cm.
- 2) Jarak terdekat objek dari mobil robot yang dapat dideteksi adalah 0,5 cm, sedangkan jarak terjauh 85 cm.
- 3) Objek yang membentuk sudut  $0^\circ$  sampai dengan  $90^\circ$  dapat terdeteksi, jika ukuran objek lebih besar dari 6,4 mm.
- 4) Dengan pemasangan *loop* (selongsong) pada transduser penerima akan menyebabkan penerima hanya akan menerima gelombang pantul saja.
- 5) Jarak maksimum objek yang dapat terdeteksi oleh sensor ultrasonik adalah 70 cm, dengan jarak pemasangan penerima dan pemancar gelombang ultrasonik 3,5 cm serta panjang *loop* 3,5 cm.

Objek tidak terdeteksi dengan baik oleh sensor ultrasonik, hal ini disebabkan oleh beberapa kemungkinan, antara lain: ukuran dan bentuk objek, posisi benda dan teknik pemasangan sensor yang tidak tepat. Oleh karena jika kita akan menggunakan sensor gelombang ultrasonik perlu untuk memperhatikan teknik pemasangannya. Perlu ditentukan dengan cara pengujian sensor gelombang ultrasonik, berapa jarak objek ke penerima yang diinginkan agar objek terdeteksi dengan baik. Sebaliknya juga bisa terjadi, sensor penerima ultrasonik menerima bukan gelombang pantul dari objek/benda melainkan gelombang yang diteruskan, sehingga sensor merespon walaupun tidak ada benda/objek. Hal ini terjadi karena sensor R ultrasonik menerima gelombang yang diteruskan dari T ultrasonik. Keadaan ini dapat diatasi dengan memasang *loop* (selongsong) pada sensor ultrasonik. Hasil pengujian untuk mengetahui bahwa teknik pemasangan sensor gelombang ultrasonik berpengaruh terhadap kepekaan mendeteksi objek.

Hasil Riset penulis ada 4 kemungkinan teknik pemasangan sensor gelombang ultrasonik, adalah sebagai berikut:

- 1) Penerima dan pemancar dipasang horisontal dengan panjang *loop* 3 cm.

Hasil pengujian sensor gelombang ultrasonik di atas didapatkan bahwa sensor gelombang ultrasonik dapat mendeteksi objek pada jarak 36 cm, apabila jarak antara penerima dan pemancar adalah 0,5 cm.

Sedangkan sensor penerima dan pemancar dipasang pada jarak 3,5 cm, sensor gelombang ultrasonik hanya mampu mendeteksi objek sejauh 70 cm. Untuk pemasangan antara pemancar dan penerima berjarak 4,5 cm dengan ukuran objek 11 cm tidak dapat terdeteksi dengan baik.

- 2) Penerima dan pemancar dipasang vertikal dengan panjang *loop* 3 cm.

Hasil pengujian didapatkan bahwa sensor dapat mendeteksi objek sejauh 39,5 cm, apabila jarak antara penerima (R) dan pemancar (T) adalah 0,5 cm. Sedangkan penerima (R) dan pemancar (T) dipasang pada jarak 3,5 cm, sensor hanya mampu mendeteksi objek sejauh 5,5 cm. Untuk pemasangan antara penerima (R) dan pemancar (T) 4,5 cm dari posisi objek dengan ukuran objek 11 cm tidak dapat terdeteksi dengan baik.

- 3) Variabel panjang *loop* dengan R dan T dipasang secara horisontal

Hasil pengujian didapatkan bahwa panjang selongsong (*loop*) berpengaruh terhadap jarak objek yang dapat dideteksi oleh sensor gelombang ultrasonik. Untuk panjang *loop* 2 cm jarak terdekat yang dapat dideteksi oleh sensor adalah 2,5 cm, sedangkan jarak objek terjauh yang dapat dideteksi 47 cm.

Panjang *loop* 3,5 cm jarak terdekat yang dapat dideteksi oleh sensor adalah 4 cm, sedangkan jarak terjauh objek yang dapat terdeteksi 70 cm.

- 4) Variabel panjang *loop* dengan R dan T dipasang secara vertikal

Hasil pengujian didapatkan bahwa panjang *loop* berpengaruh terhadap jarak objek yang dapat terdeteksi oleh sensor gelombang ultrasonik. Panjang *loop* 2 cm jarak terdekat yang dapat terdeteksi adalah 2,5 cm, sedangkan jarak objek terjauh adalah 25 cm.

Untuk panjang *loop* 3,5 cm jarak terdekat yang dapat dideteksi oleh sensor gelombang ultrasonik adalah 4 cm, sedangkan jarak terjauh objek yang dapat terdeteksi adalah 28,5 cm. Jarak ini diukur dari objek terhadap permukaan loop yang dipasang pada sensor gelombang ultrasonik.

Pengujian sensor yang dilakukan mempunyai manfaat dalam mengaplikasikan sensor tersebut, sehingga didapatkan hasil yang optimal. Tanpa adanya pengujian sensor menyebabkan hasil riset yang dilakukan mengalami kegagalan. Dari pengujian dapat diketahui sifat sensor gelombang ultrasonik yang dapat memancarkan dan menerima gelombang ultrasonik. Sifat inilah yang mendasari sensor gelombang ultrasonik dimanfaatkan untuk mendeteksi objek/benda. Agar pantulan gelombang ultrasonik dari objek dapat mengenai penerima (R), maka diperlukan pengaturan posisi penerima (R) maupun pemancar (T). Penempatan sensor yang tepat benda yang terdeteksi dapat diketahui posisinya dan dibedakan. Setelah mengetahui penempatan sensor yang tepat, kemudian sensor dapat diaplikasikan pada peralatan yang kita kehendaki.

Berdasarkan eksperimen yang penulis lakukan, dapat diambil kesimpulan, sebagai berikut:

1. Semakin dekat jarak antara sensor penerima (R) dan pemancar (T) gelombang ultrasonik, semakin jauh objek yang dapat dideteksi.
2. Pemasang sensor gelombang ultrasonik secara vertikal atau horisontal akan mempengaruhi jarak objek yang dapat dideteksi.
3. Panjang selongsong (*loop*) yang dipasang pada sensor gelombang ultrasonik akan mempengaruhi jarak objek yang dapat dideteksi dan menghilangkan gelombang yang diteruskan (bukan gelombang pantul).
4. Sensor gelombang ultrasonik yang digunakan dalam riset ini dapat mendeteksi jarak objek terjauh 85 cm, sedangkan jarak terdekat 0,5 cm.
5. Panjang gelombang ultrasonik mempengaruhi ukuran objek/ benda terkecil yang dapat dideteksi.
6. Panjang *loop* yang dipasang adalah 2 cm dan jarak antara penerima (R) dan pemancar (T) 1,5 cm, hal ini akan menghasilkan aplikasi sensor optimal.

## Bab 6

# Perancangan dan Realisasi Sistem Mobil Robot

Robot adalah peralatan manipulator yang mampu diprogram, mempunyai berbagai fungsi, yang dirancang untuk memindahkan barang, komponen-komponen, peralatan, atau alat-alat khusus, melalui berbagai gerakan terprogram untuk pelaksanaan berbagai pekerjaan. Secara mendasar, robot memiliki banyak hal yang sama dengan otomasi internal, mereka memanfaatkan piranti tenaga yang serupa (seperti listrik, hidraulik, atau pneumatik) dan mereka dikendalikan melalui urutan-urutan yang telah dikendalikan melalui program, yang memungkinkan mesin tersebut pada posisi yang diinginkan. Lingkungan seperti ini didefinisikan sebagai lingkungan dalam perkembangan mesin yang terotomatisasi ini akan menjadi bermacam-macam spesifikasi tergantung kebutuhan aktifitas manusia terhadap otomatisasi industri dan robotik.

Sistem robotik merupakan bidang yang dinamis dan maju pesat perkembangannya. Perkembangan itu berkaitan: komputasi, aplikasi mesin dan elektronika dengan teknologi terapan. Penelitian dibidang ini biasanya berakar dari industri dan memecahkan masalah industri dengan teknologi yang ada. Misalnya adalah pengembangan perangkat lunak untuk mendapatkan algoritma baru bagi kendali robot, sistem penglihatan dengan resolusi yang lebih tinggi, perbaikan kemampuan sensor dan pengembangan protokol komunikasi. Sehingga robot diasumsikan sebagai gabungan antara perangkat mekanik dan perangkat elektronik yang berfungsi untuk menggantikan pekerjaan manusia, seperti pekerjaan pada temperatur tinggi, zat kimia, ruang hampa udara, dan pada kondisi yang tidak

mungkin dikerjakan oleh manusia. Banyak robot yang dimanfaatkan sebagai peralatan hiburan (hobbies) dan ada pula robot yang bertugas untuk menggantikan pekerjaan yang menuntut keahlian (*accuracy*), kecepatan dan lain-lain. Ada pula robot yang berfungsi untuk mengerjakan pekerjaan yang rutin seperti pada pemintalan benang. Pada bidang pertahanan keamanan (Hankam), digunakan sebagai penjinak bom. Saat ini robot dikembangkan agar berpikir sendiri dengan logika yang telah ditanamkan pada *software* robot.

Pada pokok bahasan yang sudah dibicarakan diatas berkaitan tentang motor servo dan pengujianya, LDR (*Light Dependent Resistor*), mikrokontroler, pemrograman dengan bahasa PBasic dan teknik pemasangan sensor gelombang ultrasonik dapat diaplikasikan untuk merancang mobil robot. Dalam buku ini yang akan dibahas adalah *wheeled robot* (robot beroda) yang dilengkapi dengan sensor gelombang ultrasonik untuk mendeteksi adanya objek/penghalang dan menghindari benturan. Motor servo digunakan sebagai penggerak dan sensor gelombang ultrasonik dimanfaatkan agar mobil robot dapat menghindari benturan dan mengikuti lintasan tertentu . Dengan pengetahuan dan pengujian yang telah dibahas pada pokok bahasan sebelumnya, maka kita dapat merakit mobil robot lengkap dan siap bergerak sesuai dengan perintah.

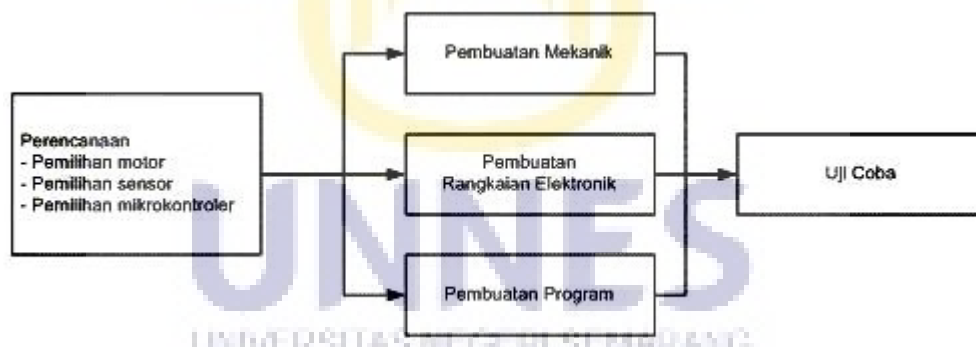
Buku ini diharapkan dapat sebagai langkah awal (pemula) untuk meneliti dan mengembangkan lebih lanjut tentang sistem robotik, dimana hingga saat ini belum dikembangkan dan diteliti secara intensif. Disamping itu bermanfaat bagi: mahasiswa, dosen, guru dan penggemar robot untuk mengembangkan sistem robotik dengan merancang berbagai macam bentuk robot sesuai dengan kebutuhan dan implementasinya. Dalam rangka ikut mengembangkan dan memasyarakatkan dunia robotik di tanah air, maka penulis memberanikan diri untuk merancang dan pengembangan mobil robot. Penulis berharap bahwa nantinya tulisan ini akan ikut memberikan kontribusi bagi perkembangan dunia robot di tanah air secara umum dan dapat dijadikan referensi untuk pembuatan robot.

Robot Mobil *Mobile Robot* adalah konstruksi robot ciri khasnya dengan mempunyai aktuator berupa roda untuk menggerakkan keseluruhan badan robot tersebut, sehingga dapat melakukan gerakan/ perpindahan posisi dari satu titik ke titik yang lain. Robot mobil ini sangat disukai orang yang mulai mempelajari, karena sangat sederhana dan tidak memerlukan biaya yang besar. Hal ini karena membuat robot mobil tidak memerlukan

kerja fisik yang berat. Untuk dapat membuat sebuah robot mobile minimal diperlukan pengetahuan tentang mikrokontroler dan sensor-sensor elektronik. *Base* robot mobil dapat dengan mudah dibuat dengan menggunakan *plywood*, akrilik sampai menggunakan logam.

Robot mobil dapat dibuat dengan tujuan sebagai robot pengikut garis (*Line Follower*), robot pengikut dinding (*Wall Follower*) dan robot pengikut cahaya. Pengembangan mobil robot saat ini banyak dilakukan oleh beberapa anak bangsa bahkan anak SMA pun telah mencoba untuk mengembangkan mobil robot ini untuk beberapa fungsi, diantaranya adalah *robot line follower*, *maze solving*, robot pemadam kebakaran dan beberapa bentuk robot lain yang lebih unik, seperti yang sekarang sedang dikembangkan oleh Ektrakurikuler sistem Robotik dan beberapa kontes atau perlombaan pun diadakan untuk menguji sejauh mana anak-anak mampu untuk mendalami ilmu tentang robot.

Perancangan mobil robot menggunakan perangkat keras dan perangkat lunak untuk merakit sebuah mobil robot yang dapat bergerak secara otomatis dan menghindari benturan penghalang di depannya. Perancangan proses pembuatan mobil robot secara umum mempunyai beberapa tahapan sebagaimana yang ditunjukkan pada (Gambar 6.1)



Gambar 6.1: Skema Proses pembuatan mobil robot

Pembuatan mobil robot ada 3 tahap yang harus diperhatikan, yaitu:

1) Perencanaan dalam pembuatan mobil robot

Pada tahap ini, kita merencanakan pembuatan mobil robot dengan tujuan untuk apa ? Beberapa masalah yang perlu diperhatikan dalam tahap ini, sebagai berikut:



- Dimensi, yaitu panjang, lebar, tinggi, dan perkiraan berat dari robot. Robot KRI berukuran tinggi sekitar 1m, sedangkan tinggi robot KRCI sekitar 25 cm.
- Struktur material, apakah dari alumunium, besi, kayu, plastik, dan sebagainya.
- Cara kerja robot, berisi bagian-bagian robot dan fungsi dari bagian-bagian itu. Misalnya lengan, konveyor, lift, power supply.
- Sensor apa yang akan digunakan pada robot ini.
- Bagaimana mekanisme, agar mekanik/gerakan robot dapat menyelesaikan tugas.
- Metode pengendali, yaitu bagaimana robot dapat dikendalikan dan digerakkan, mikroprosesor yang digunakan, dan blok diagram sistem.
- Strategi untuk memenangkan pertandingan, jika memang robot itu akan diikuti dalam lomba/kontes robot Indonesia/Internasional.

## 2) Sistem elektronika

Sistem elektronika dirancang sesuai dengan kebutuhan dan fungsi yang diinginkan. Misalnya untuk menggerakkan motor pada mobil robot diperlukan servo motor DC dan pengendalinya. Sensor dan mikrokontroler yang digunakan sebagai kendali gerakan mobil robot serta bahasa pemrograman.

## 3) Uji coba

Setelah kita mendownload program ke mikrokontroler (otak robot) berarti kita siap melakukan tahapan terakhir dalam membuat robot, yaitu uji coba. Untuk KRCI, ujicoba dilakukan pada arena seluas sekitar 4 4 meter dan berbentuk seperti *puzzle*. Dalam arena KRCI ini diletakkan lilin-lilin yang harus dipadamkan oleh robot cerdas pemadam api. Contoh gambar robot pemadam api *Ted Larsorn* dan arena Kontes Robot Cerdas Indonesia (KRCI).

## 6.1 Peralatan yang Digunakan

Pada implementasi program, alat-alat yang digunakan yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*) yang dijabarkan sebagai berikut:

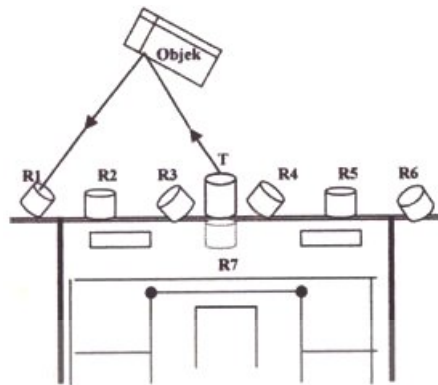
### 6.1.1 Perangkat Keras

- 1) Mikrokontroler Basic Stamp BS2SX
- 2) Motor servo DC 2 buah sebagai penggerak
- 3) Pemancar gelombang ultrasonik
- 4) Penerima gelombang ultrasonik
- 5) LDR (*Light Dependent Resistor*)
- 6) Adaptor 12 volt
- 7) 3 buah roda mobil robot

Setelah semua perangkat keras sudah tersedia, maka langkah selanjutnya adalah: merakit robot beroda, sehingga terbentuknya robot yang lengkap dan siap bergerak, disain PCB untuk sensor ultrasonik dan memasang sensor pada robot beroda agar dapat dikendalikan sesuai kebutuhan. Perancangan mobil robot dalam buku ini menggunakan 3 buah penerima dan satu buah pemancar. Penerima maupun pemancar dipasang di depan mobil robot menghadap ke depan dan dipasang *loop*. Dua penerima dipasang secara horisontal dengan pemancar, satu penerima dipasang secara vertikal di atas pemancar.. Lebar depan mobil robot dibagi menjadi 3 bagian, yaitu kanan, tengah dan kiri. Bagian kanan dan kiri masing-masing dipasang 3 transduser, sedangkan bagian tengah dipasang satu transduser. Penempatan 3 transduser pada bagian kanan dan kiri mobil robot dengan cara: 2 transduser dipasang di tepi dengan membentuk sudut ke arah dalam dan satu transduser dipasang lurus ke depan. Teknik pemasangan ke 7 transduser penerima dan satu transduser pemancar gelombang ultrasonik dapat dilihat pada gambar di bawah.

### 6.1.2 Perangkat Lunak

Perangkat lunak merupakan faktor penting dalam tahap perancangan robot. Perangkat lunak ini merupakan algoritma gerak dan tugas robot dalam bentuk listing program yang ditanamkan ke dalam mikrokontroler. Program dapat bermacam-macam bentuk versi dan bahasa pemrogramannya, sesuai dengan spesifikasi dari mikrokontroler yang digunakan.



Gambar 6.2: Skema Pemasangan Penerima (R) dan Pemancar (T)

Mikrokontroler *basic stamp (BS2X)* menggunakan bahasa pemrograman *Pbasic*. *Software* yang digunakan adalah *basic stamp editor*. *Basic stamp editor* adalah sebuah editor yang dibuat oleh Parallax Inc untuk menulis program, mengkompilasi dan mendownloadnya ke mikrokontroler keluarga basic stamp. Program ini memungkinkan penggunanya memprogram *basic stamp* dengan bahasa basic yang relatif ringan dibandingkan bahasa pemrograman lainnya.

*Software* yang dibahas dalam buku ini adalah mikrokontroler *Basic Stamp BS2SX* dengan bahasa pemrograman *PBasic*. Bahasa pemrograman *PBasic* meliputi:

- a) Merancang sistem pengendali pada robot beroda agar dapat memenuhi spesifikasi yang telah ditentukan.
- b) Menyusun program agar robot beroda dapat bergerak sesuai dengan perintah dan dapat menghindari penghalang.
- c) Pemrograman mikrokontroler dan pengujian robot.
- d) Pemrograman Robotik.

Untuk menulis program menggunakan *Basic Stamp Editor* yang sudah di-install sebelumnya pada PC. Setelah menulis program, PC dihubungkan dengan *Basic Stamp Microcontroller*. Tekan *run* pada *Basic Stamp Editor* untuk proses *down-load* program ke *EEPROM*. Proses *down-load* program otomatis menghapus program yang ada sebelumnya dan mikrokontroler akan mengeksekusi program yang terakhir.

### 6.1.3 Ketentuan Pemrograman *Parallax Basic (PBasic)*

- 1) Program yang telah ditulis dalam *BASIC Stamp Editor* kemudian disimpan dalam format \*BS2SX.
- 2) Selanjutnya program yang telah disimpan dalam format \*BS2SX akan diubah selama proses *down-load* ke dalam format biner yang dapat dibaca oleh mikrokontroler menggunakan *interpreter* yang ada pada *Basic Stamp*. Komentar *Tokenized is successfully* akan muncul bila program *Pbasic* telah berhasil.
- 3) Untuk reprogram *Basic Stamp* hanya dengan menghubungkan PC, kemudian dijalankan *Basic Stamp Editor*, selanjutnya tekan *Alt-R* atau *RUN*.
- 4) Bila dibandingkan dengan program mikrokontroler yang lain, pemrograman *Pbasic* akan lebih mudah.
- 5) Setiap perintah dalam program akan menempati ruang dalam *EEPROM*, besarnya bervariasi sesuai dengan kompleksitas dan jenis perintah yang ditulis.

## 6.2 Pengujian Robot mobil

Sifat sensor ultrasonik yang dapat memancarkan dan menerima pantulan gelombang dapat digunakan untuk mendeteksi objek/benda. Agar pantulan gelombang ultrasonik dari objek dapat mengenai penerima dengan tepat, maka perlu diatur posisi antara sensor penerima maupun pemancar. Penempatan sensor ultrasonik secara tepat, maka benda yang dideteksi dapat dibedakan dan diketahui posisinya, maka perlu adanya pengujian.

### 6.2.1 Pengujian hubungan antara jumlah pulsa diinputkan dengan jarak yang ditempuh oleh mobil robot bergerak maju

Sensor untuk mendeteksi adanya objek dapat diimplementasikan dengan berbagai banyak cara. Cara yang sering digunakan adalah dengan memanfaatkan sifat halangan yang dapat memantulkan gelombang ultrasonik. Untuk sensor yang dapat mendeteksi objek menggunakan sensor gelombang ultrasonik sebagai medianya, gelombang yang digunakan adalah gelombang ultrasonik. Sensor gelombang ultrasonik untuk deteksi objek terdiri

dari 2 bagaian, yaitu bagian *transmitter* (pemancar) dan *receiver* (penerima). Pengujian ini dilakukan untuk mengetahui jumlah pulsa yang diinputkan pada servo motor terhadap jarak tempuh mobil robot yang bergerak. Dengan melakukan pengujian akan didapatkan hubungan antara jumlah pulsa yang diinputkan terhadap jarak tempuh robot. Hal ini perlu dilakukan agar robot bergerak sesuai dengan yang kita kehendaki, apabila salah menginputkan pulsa, robot bergerak tidak sesuai yang kita kehendaki. Hasil pengujian hubungan antara pulsa yang diinputkan ke mikrokontroler terhadap jarak yang ditempuh pada saat mobil robot bergerak maju, adalah sebagai berikut:

Tabel 6.1: Hubungan antara pulsa dengan jarak pada saat mobil robot bergerak maju

No.	Pulsa servo kanan	Pulsa servo kiri	Periode	Jumlah putaran	Waktu (s)	Kecepatan (cm/detik)	Jarak (cm)
1	1500	1750	1000	15,25	23,5	12,23	287,41
2	1500	1750	500	7,75	12	12,17	146,01
3	1500	1750	100	1,50	2,4	11,78	18,84
4	1500	1750	50	0,75	1,3	10,87	14,13

### 6.2.2 Pengujian hubungan jumlah pulsa diinputkan dengan sudut belok mobil robot

Pengujian untuk mengetahui jumlah pulsa yang diinputkan terhadap sudut belok robot. Pengujian ini sangat penting dilakukan sebab kita dapat mengetahui sudut belok yang optimal, sehingga pergerakan mobil robot dapat kontinu. Jika kita salah dalam menginputkan jumlah pulsa tanpa pengujian terlebih dahulu, maka robot akan berputar terus.

Pengujian meneliti pengaruh jarak penerima ke pemancar terhadap jarak objek terdeteksi oleh sensor ultrasonik. Hasil pengujian pada (Tabel 3.1) dan (Tabel 3.2).

Pada (Tabel 2.3) dan (Tabel 2.4.) adalah pengujian untuk mengamati perbedaan pemasangan sensor ultrasonik secara vertikal dan horisontal terhadap jarak objek yang dapat terdeteksi. Panjang loop bervariasi dari 2 cm sampai 3,5 cm. Semakin panjang loop

Tabel 6.2: Hubungan jumlah pulsa diinputkan dengan sudut belok mobil robot

No.	Jumlah pulsa servo kanan	Jumlah pulsa servo kiri	Periode	Sudut belok (derajat)	Keterangan (detik)
1	1650	1600	100	0	Diam
2	1675	1800	50	10	Belok kanan
3	1700	2000	50	60	Belok kanan
4	1650	1700	100	90	Belok kanan
5	1750	1700	100	180	Belok kanan
6	1750	2000	100	270	Belok kanan
7	1750	1600	100	90	Belok kiri
8	1550	1500	100	180	Belok kiri
9	1600	1400	100	270	Belok kiri

yang digunakan objek yang dapat terdeteksi oleh sensor semakin jauh. Pada pengujian ini digunakan ukuran objek dan jarak antara pemancar dan penerima gelombang ultrasonik sama. Dari hasil pengujian didapatkan hasil sebagai berikut:

- 1) Ukuran objek terkecil yang dapat dideteksi oleh sensor gelombang ultrasonik adalah 6,4 mm pada jarak 8 cm.
- 2) Jarak terdekat objek dari mobil robot yang dapat dideteksi adalah 0,5 cm, sedangkan jarak terjauh 85 cm.
- 3) Objek yang membentuk sudut 0<sup>o</sup> sampai dengan 90<sup>o</sup> dapat terdeteksi, jika ukuran objek lebih besar dari 0,64 mm.
- 4) Dengan pemasangan loop pada transduser penerima akan menyebabkan penerima hanya akan menerima gelombang pantul saja.

Dari (Tabel 3.1.) dan (Tabel 3.2.) terlihat bahwa roda kiri pada kondisi diam, jika jumlah pulsa yang diberikan adalah 1600 pulsa. Sedangkan untuk memutar roda searah jarum jam, maka jumlah pulsa yang diinputkan harus lebih kecil dari 1600 pulsa. Untuk putaran roda berlawanan jarum jam, maka jumlah pulsa yang diinputkan harus lebih

besar dari 1600 pulsa.

Roda kanan pada kondisi diam jumlah pulsa yang harus diberikan adalah 1650. Untuk putaran searah dengan arah jarum, maka jumlah pulsa yang harus diinputkan lebih kecil 1650 pulsa. Sedangkan untuk berlawanan dengan arah jarum jam, maka jumlah pulsa yang harus diinputkan lebih besar 1650 pulsa. Dari hasil uji coba di atas dapat diaplikasikan untuk menggerakkan robot beroda, adalah sebagai berikut:

- a) Gerakan maju, roda kiri diberikan jumlah pulsa  $>1600$  pulsa dan roda kanan diberikan jumlah pulsa  $<1650$  pulsa.
- b) Gerakan mundur, roda kiri diberikan jumlah pulsa  $<1600$  pulsa dan roda kanan diberikan jumlah pulsa  $>1650$  pulsa.
- c) Belok kanan, roda kiri diberikan jumlah pulsa  $>1600$  pulsa dan roda kanan diberikan jumlah pulsa  $=1650$  pulsa.
- d) Belok kiri, roda kiri diberikan jumlah pulsa  $=1600$  pulsa dan roda kanan diberikan jumlah pulsa  $<1650$  pulsa.

Hasil pengujian di atas dapat diterapkan pada gerakan mobil robot untuk menghindari benturan. Untuk menghindari benturan dengan objek ada beberapa cara, antara lain:

1. Menghindari benturan dengan melakukan gerakan mundur terlebih dahulu, kemudian membelok.
2. Menghindari benturan dengan mengurangi kecepatan terlebih dahulu, kemudian membelok.

Objek tidak dapat terdeteksi dengan baik oleh sensor ultrasonik disebabkan oleh beberapa kemungkinan, antara lain : ukuran dan bentuk objek, posisi objek dan cara pemasangan sensor ultrasonik. Oleh karena itu pada saat memasang sensor gelombang ultrasonik pada mobil robot perlu diperhatikan letaknya.

Sebaliknya juga bisa terjadi, sensor penerima gelombang ultrasonik menerima gelombang bukan gelombang pantul dari objek, sehingga merespon walaupun tidak ada objek. Hal ini terjadi karena sensor gelombang ultrasonik menerima gelombang yang diteruskan dari

pemancar gelombang ultrasonik. Keadaan ini dapat diatasi dengan memasang *loop* (selongsong) pada sensor gelombang ultrasonik, sehingga penerima hanya akan menerima gelombang pantul saja.

### 6.3 BASIC Stamp Editor

Perangkat lunak merupakan faktor penentu paling akhir dalam tahap perancangan robot. Perangkat lunak ini berupa algoritma gerak dan tugas robot dalam bentuk listing program yang ditanamkan ke dalam mikrokontroler. Program dapat bermacam-macam bentuk versi dan bahasa pemrogramannya, sesuai dengan spesifikasi dari mikrokontroler yang digunakan.

*BASIC Stamp editor* adalah sebuah perangkat lunak (*software*) buatan *parallax* yang berfungsi sebagai mediator pada pemrograman *BASIC Stamp* dan untuk menulis program, meng-compile dan men-download program pada mikrokontroler *BASIC Stamp*. Pada penulisan buku ini, Editor Basic Stamp yang digunakan adalah berbasis pemrograman dengan *Pbasic*. Agar dapat menjalankan *software* dari *Editor Basic Stamp* ini dengan baik dan sesuai dengan kebutuhan, maka sistem minimal komputer yang dibutuhkan adalah sebagai berikut:

- (a) Komputer dengan prosesor 80486 (80286 untuk DOS) atau lebih tinggi.
- (b) Sistem operasi dari Windows 95/ 98/ NT4 / 2000 (DOS 5.0 atau versi yang lebih tinggi).
- (c) RAM (*Random Access Memory*) yang mempunyai kapasitas minimum 16 MB (1 MB untuk DOS).
- (d) Tersedia ruang simpan *hard disk* minimum 1 MB.
- (e) Adanya *Port serial PC* untuk koneksi pemrograman mikrokontroler.

Bahasa yang digunakan dalam pemrograman ini menggunakan bahasa *Pbasic*. Berikut adalah langkah-langkah untuk memulai menggunakan *Basic Stamp Editor*:

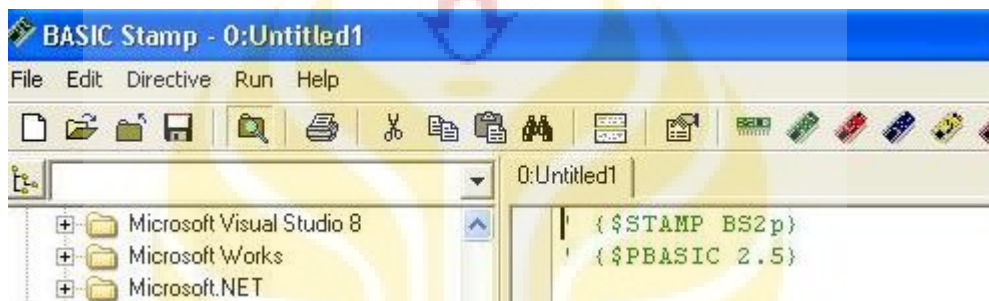
- 1) Menjalankan, BASIC Stamp editor V 2.4





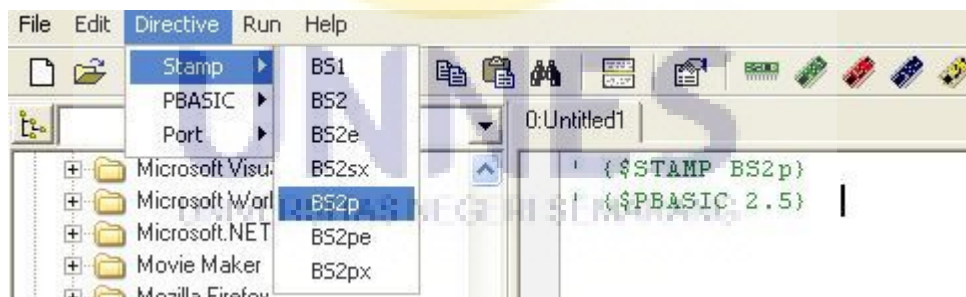
Gambar 6.3: Menjalankan *BASIC Stamp Editor V 2.4*

Setelah memulai untuk menjalankan *BASIC Stamp editor* tersebut, maka selanjutnya *BASIC Stamp editor* siap digunakan. Berikut adalah tampilan utama dari *BASIC Stamp editor V 2.4*.



Gambar 6.4: Tampilan Utama *BASIC Stamp Editor V 2.4*

## 2) Tipe mikrokontroler



Gambar 6.5: Tipe Mikrokontroler Melalui Menu Utama

Dari gambar diatas terlihat bahwa beberapa tipe mikrokontroler BASIC Stamp, pemilihan tipe digunakan sesuai dengan mikrokontroler yang akan dipakai.

3) Menjalankan Program Untuk menjalankan atau mendownload program pada *Basic Stamp* sudah semestinya dilakukan penulisan program minimal satu intruksi. Pemrograman BASIC Stamp bisa dikatakan mudah dibandingkan dengan pemrograman mikrokontroler lainnya. Pada menu help,lalu indek telah terdapat beberapa sintak yang disediakan untuk membuat program pada *BASIC Stamp*. Beberapa sintak yang sering digunakan dalam pembuatan tugas akhir ini adalah sebagai berikut.

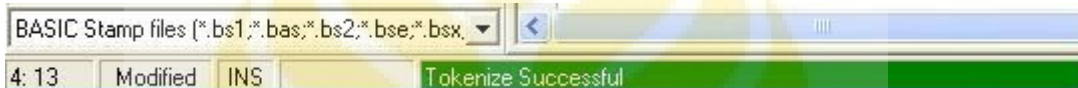
- *DO*, merupakan perintah untuk memulai suatu program yang diakhir program harus ditutup dengan perintah *LOOP*.
- *LOOP*, merupakan perintah pengulangan dimana pada awal program dibuka dengan perintah *DO*.
- *DEBUG*, merupakan perintah untuk menampilkan informasi pada layar PC atau laptop. Seluruh informasi dapat ditampilkan termasuk informasi dari sensor ataupun informasi yang diketik dengan diberi awal dan akhiran tanda kutip dua ( informasi ). Informasi akan muncul apabila baudrate pada *BASIC Stamp* telah disesuaikan. Berikut adalah tabel *baudrate* untuk *DEBUG*

	BS1	BS2, BS2e, BS2sx, BS2p, and BS2pe	BS2px
Serial Protocol	Asynchronous 4800 baud, N, 8, 1 Inverted polarity, Raw Data Custom packetized format	Asynchronous 9600 baud, N, 8, 1 Inverted polarity, Raw Data	Asynchronous 19.2 kBaud, N, 8, 1 Inverted polarity, Raw Data

Gambar 6.6: Tabel penyesuaian *baudrate*

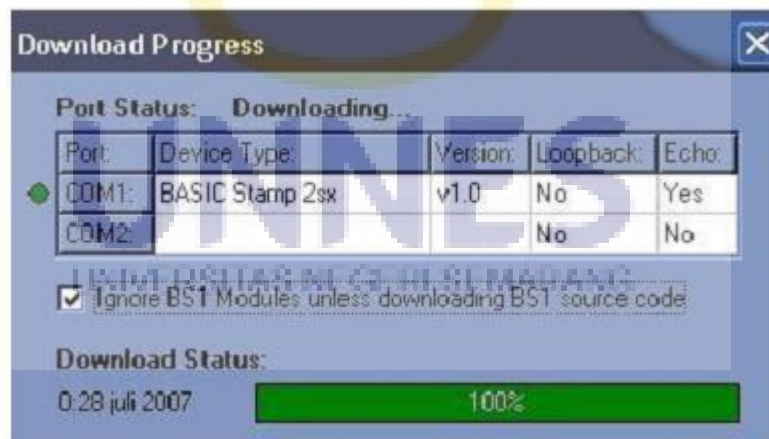
- *PAUSE*, merupakan perintah untuk memberikan *delay* pada program dalam satuan milidetik.
- *SEROUT*, merupakan perintah komunikasi untuk mengirimkan data secara serial.
- *SERIN*, merupakan perintah komunikasi untuk menerima data secara serial.
- *FOR dan NEXT*, merupakan perintah pengulangan yang lebih spesifik. Pengulangan dapat dibatasi sesuai dengan kebutuhan program.
- *GOTO*, merupakan perintah untuk lompat menuju *subroutine*.

- *GOSUB*, merupakan perintah untuk lompat menuju *subroutine*. Fungsi *GOSUB* hampir sama dengan *GOTO*. *GOSUB* merupakan kependekan dari *GO to a SUBroutine*. *GOSUB* biasanya diakhiri oleh fungsi *RETURN*
- *RETURN*, merupakan fungsi yang berdampingan dengan *GOSUB* dimana setelah eksekusi dari *subroutine* selesai maka program akan kembali ke tempat setelah *GOSUB*. Program yang telah di tulis melalui *BASIC Stamp Editor* selanjutnya akan didownload, namun sebelum program didownload sebaiknya diperiksa terlebih dahulu. Untuk memeriksa sintaknya maka tinggal menekan *CTRL+T*. berikut adalah tampilan untuk memeriksa sintak yang sudah benar.



Gambar 6.7: Hasil Pemeriksaan Sintak yang Benar

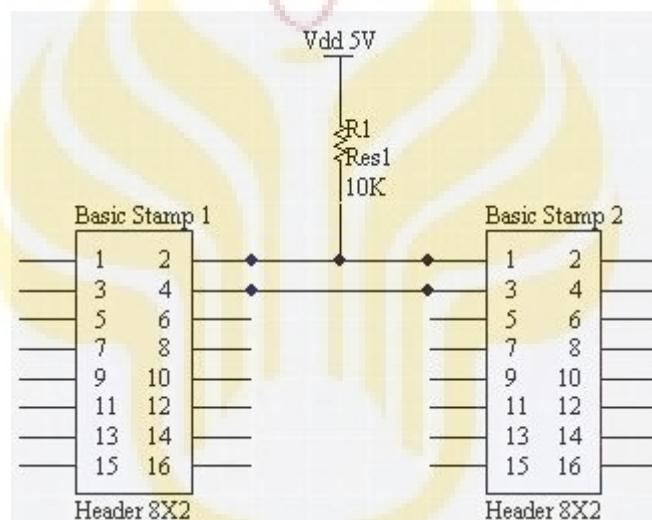
Untuk mengcompile dan mendownload program bisa dilakukan dengan menekan *CTRL+R*. Berikut adalah tampilan jika pendownloadan program sukses.



Gambar 6.8: Tampilan Pendownload Program

## 6.4 Komunikasi *Basic Stamp*

Komunikasi Basic Stamp ke Basic Stamp biasanya dilakukan apabila suatu alat diharuskan menggunakan lebih dari satu buah mikrokontroler. Komunikasi ini pada dasarnya sama dengan komunikasi *Basic Stamp* ke mikrokontroler lain. Perlu diperhatikan *baudrate* pada masing-masing mikrokontroler. Pada umumnya *baudrate* yang dipakai adalah 9600. Apabila terdapat perbedaan *baudrate* pada mikrokontroler yang lain, maka dapat disesuaikan dengan *baudrate* antara mikrokontroler yang satu dengan yang lainnya. Berikut adalah gambar rangkaian komunikasi *Basic Stamp* ke *Basic Stamp*

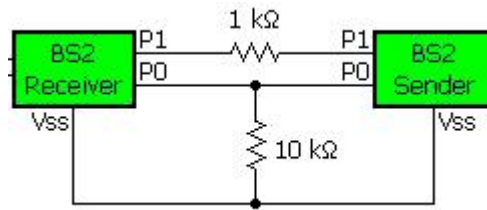


Gambar 6.9: Komunikasi *Basic Stamp*

Adapun alternatif lain dari komunikasi Basic Stamp ke Basic Stamp diperlihatkan pada gambar di bawah ini.

Contoh program dalam komunikasi Basic Stamp adalah sebagai berikut.

Perintah *serout* merupakan perintah pengiriman data, dan dilanjutkan dengan pin yang akan dipakai, kemudian dilanjutkan dengan *baudrate*, selanjutnya data yang akan dikirim. Perintah *serin* merupakan perintah menerima data, dan dilanjutkan dengan pin yang akan dipakai, kemudian dilanjutkan dengan *baudrate*, selanjutnya data yang akan diterima. Data yang akan diterima sebelumnya dilakukan penginisialisasian. Inisialisasi



Gambar 6.10: Komunikasi *Basic Stamp* ke *Basic Stamp*

```

Untitled1 | 0:Untitled2 | Untitled3 | 0:Untitled4 |
' {$STAMP BS2p}
' {$PBASIC 2.5}
Main:
DO
SEROUT 1, 240, ["Hello!", CR]
PAUSE 2500
LOOP

```

Gambar 6.11: Contoh program *SEROUT*

biasanya menggunakan variable bit sampai *word*.

Seperti yang telah dijelaskan sebelumnya, bahwa dalam komunikasi antara mikrokontroler *Basic Stamp* dengan Mikrokontroler *Basic Stamp* atau dengan mikrokontroler lainnya perlu diperhatikan *baudrate* yang akan digunakan dalam aplikasinya untuk pengendali.

## 6.5 Port Serial/ RS-232

Protokol standar yang mengatur komunikasi melalui serial port disebut RS-232 yang dikembangkan oleh EIA (*Elektronik industries Association*). Enkoding yang digunakan dalam komunikasi serial adalah NRZ (*Non-Return-to-Zero*), dimana bit 1 dikirim sebagai high value dan bit 0 dikirimkan sebagai low value. Dalam *interfacing RS-232*, tegangan negatif mempresentasikan bit 1 dan tegangan positif mempresentasikan bit 0. RS-232 serial port juga merupakan rangkaian *converter* komunikasi antara mikrokontroler ke PC atau sebaliknya. PC memiliki *high logic* 5V - 12V dan untuk *low logic* (-5V) (-12V), sedangkan mikrokontroler mempunyai *level TTL low logic* 0-1. 8V dan *high logic* 2.2 - 5V, sehingga diperlukan *converter*. DB-9 adalah konektor yang digunakan untuk menghubungkan per-

```

0:Untitled1
' {$STAMP BS2p}
' {$PBASIC 2.5}

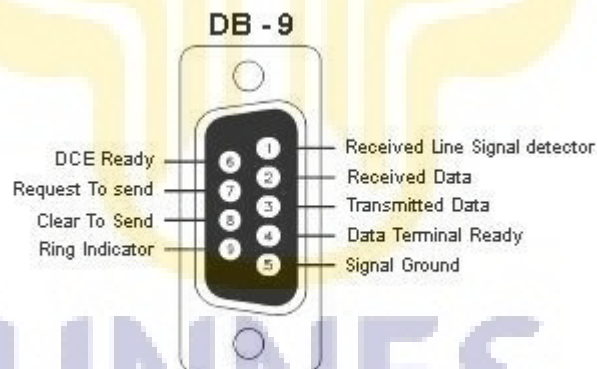
tes VAR Word

main:
SERIN 1,240,[tes]
PAUSE 1000
DEBUG tes,CR
GOTO main

```

Gambar 6.12: Contoh program *SERIN*

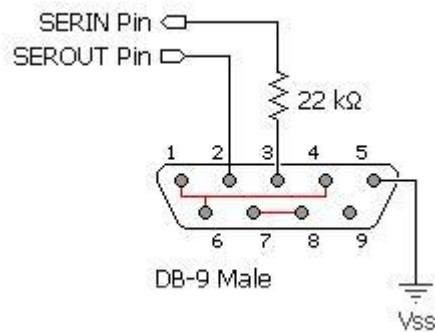
angkat keras luar komputer (*eksternal*) dengan komputer pada komunikasi serial. Pada komputer IBM PC kompatibel biasanya terdapat satu atau dua buah konektor DB-9 yang biasa dinamai *COM 1* dan *COM 2*.



Gambar 6.13: Konektor DB-9 pada bagian belakang CPU

RS 232 dapat berkomunikasi dengan berbagaimacam *device*, salah satunya adalah dengan menggunakan mikrokontroler *Basic Stamp*. Komunikasi ini tidak memerlukan pengubah level tegangan, yang diperlukan hanya sebuah resistor 22 k. Dalam beberapa permasalahan pengubah *level* tegangan akan diperlukan. Berikut adalah gambar komunikasi RS232 ke *Basic Stamp*.

Keterangan mengenai saluran RS232 pada konektor DB-9 sebagai berikut :



Gambar 6.14: Komunikasi RS232 ke *Basic Stamp*

1. *Received Line Signal Detect*, dengan saluran ini *DCE* memberitahukan ke *DTE* bahwa pada terminal masukan ada data masuk.
2. *Receiver Data*, digunakan *DTE* menerima data dari *DCE*.
3. *Transmitter Data*, digunakan *DTE* mengirimkan data ke *DCE*.
4. *Data Terminal Ready*, pada saluran ini *DTE* memberitahukan kesiapan sinyalnya.
5. *Signal Ground*, saluran *Ground*.
6. *Ring Indicator*, pada saluran ini *DCE* memberitahukan ke *DTE* bahwa sebuah stasiun menghendaki hubungannya.
7. *Clear to Send*, dengan saluran ini *DCE* memberitahukan bahwa *DTE* boleh mulai mengirimkan data.
8. *Request to Send*, dengan saluran ini *DCE* diminta mengirim data oleh *DTE*.
9. *DCE Ready*, sinyal aktif pada saluran ini menunjukkan bahwa *DCE* sudah siap.

## 6.6 Contoh Pemrograman *PBasic*

*Program 1. Membuat sebuah LED berkedip, dengan PBasic main:*

Tabel 6.3: Konfigurasi *pin* dan nama sinyal konektor *serial DB-9*

Nomor Pin	Nama Sinyal	Fungsi	Keterangan
1	DCD	<i>Input</i>	<i>Data Carrier Detect/ Received Line Signal Detect</i>
2	RxD	<i>Input</i>	<i>Receive Data</i>
3	TxD	<i>Output</i>	<i>Transmite Data</i>
4	DTR	Output	<i>Data Terminal Ready</i>
5	GND	-	<i>Ground</i>
6	DSR	<i>Input</i>	<i>Data Set Ready</i>
7	RTS	<i>Output</i>	<i>Request to Send</i>
8	CTS	<i>Input</i>	<i>Clear to Send</i>
9	RI	<i>Input</i>	<i>Ring Indicator</i>

```

high 7
pause 1000
low 7
pause 1000
goto main

```

Program ini menggunakan perintah 'tinggi' dan 'rendah' untuk mengontrol pin 7, dan menggunakan perintah 'pause' (berhenti) untuk membuat penundaan (*delay*). Perintah yang terakhir 'goto main' membuat program kembali 'melompat' ke label main pada awal program. Ini berarti program loop selamanya. Perhatikan bahwa waktu pertama label digunakan harus diikuti dengan simbol titik dua (:). Ini merupakan program untuk memberitahu komputer bahwa ada kata adalah label baru.

Kegiatan:

Mulailah (*start up*) dengan perangkat lunak Basic Stamp dan kode dalam program di atas. Simpan program dan kemudian men-download ke mikrokontroler *Basic Stamp* dengan mengklik 'Run'. Lihat lembar instruksi pada dua halaman berikutnya, yang menjelaskan bagaimana untuk melakukan hal ini untuk kedua *Windows* dan *software Basic Stamp*.





Gambar 6.15: Instruksi perangkat lunak Windows

Jika Anda memiliki pemrograman *PBasic* terhubung dengan benar, setelah beberapa detik Anda akan melihat pesan 'Download sukses' pada layar komputer, dan *LED* merah 7 pada kontroler Stamp harus mulai berkedip. Jika Anda mendapatkan pesan kesalahan periksa apakah Anda tidak membuat kesalahan ejaan dalam program ini, dan bahwa mikrokontroler *Basic Stamp* terhubung ke kedua *power supply* dan komputer.

Catatan penting: jika komputer memberikan pesan kesalahan pada baris berisi '*switch*' Anda mungkin memiliki program perangkat lunak dalam modus yang salah. Periksa perangkat lunak dalam model *PBASIC*.

*Program 2. Gerakan maju mobil Robot*

```

Low 8           ' set P12 and to out put_low '
Low 7           ' set P13 and to out put_low '
Loop           ' label for goto loop to return to '
pulsout 12, 1500 ' send 1,3 ms pulses '
pulsout 13, 1750 ' send 1,7 ms pulses '
pause 20       ' every 2 ms '
goto loop      ' send program to loop label'

```

*Program 3. Gerakan maju dan mundur secara rutin mobil Robot*

```

' ..... Main Routine .....
main :           ' main routine

```

```

forward :                ' forward routine
for pulse_count = 1 to 60  ' send 60 forward pulses
pulsout 12, 1500          ' 1,3 ms pulse to righ servo
piulsout 13, 1750        ' 1,7 ms pulse to left servo
pause 20                  ' pause for 20 ms
next
pause 500                  ' pause for 0,5 s
backward :
for pulse_count = 1 to 60  ' send 60 forward pulses
pulsout 12, 1750          ' 1,3 ms pulse to righ servo
piulsout 13, 1500        ' 1,7 ms pulse to left servo
pause 20                  'pause for 20 ms
next
pause 500                  'pause for 0,5 s
stop

```

Program 4. Gerakan belok kanan dan kiri mobil Robot

```

..... Declaration .....
Pulse_count var word      ' declare a word size loop counter
..... Initialization .....
output 2                   ' set P2 to out put
freqout 2,2000,3000       ' send 3 kHz signal for 2 s
low 12                     ' set P12 and P13 to out_ low
low 13
..... Main Routine .....
main :                     ' main routine
left_turn                  ' left turn routine
for pulse_count = 1 to 30  ' send 30 left rotare pulses
pulsout 12, 1500          ' 1,3 ms pulse to righ servo
pulsout 13, 1550          ' 1,3 ms pulse to left servo
pause 20                  ' pause for 2 ms
next

```

```

pause 500          ' pause for 0,5 s
righ_turn         ' right turn routine
for pulse_count = 1 to 30    ' send 30 left rotare pulses
pulsout 12, 1750    ' 1,3 ms pulse to righ servo
pulsout 13, 1700    ' 1,3 ms pulse to left servo
pause 20          ' pause for 2 ms next
pause 500         ' pause for 0,5 s
stop ' stop until reset

```

Program 5. Gerakan mobil Robot dengan sensor ultrasonik

---

Variabel dan konstanta

---

Ultra_kanan	VAR IN6	' P6 untuk variabel input Ultrasonik kanan
Ultra_kiri	VAR IN4	' P4 untuk variabel input Ultrasonik kiri
Ultra_tengah	VAR IN1	' P1 untuk variabel input Ultrasonik tengah
SERVO_R	CON 12	' P12 untuk servo kanan
SERVO_L	CON 13	' P13 untuk servo kiri
tdkada_obj	CON 1	' Logika 1
ada_obj	CON 0	' Logika 0
n	VAR BYTE	

---

Routine Utama

---



---

Routine cek\_sensor\_Ultra

---

```

cek_sensor
if Ultra_kanan =ada_obj and Ultra_kiri =tdkada_obj and Ultra_tengah = tdkada_obj
then mundur belok_kiri

if Ultra_kanan =tdkada_obj and Ultra_kiri =ada_obj and Ultra_tengah = tdkada_obj
then mundur belok_kanan

```

*if Ultra\_kanan =tdkada\_obj and Ultra\_kiri = tdkada\_obj and Ultra\_tengah = ada\_obj  
then mundur belok\_kanan*

*if Ultra\_kanan = ada\_obj and Ultra\_kiri =ada\_obj and Ultra\_tengah = ada\_obj then  
mundur belok\_kanan*

*if Ultra\_kanan = ada\_obj and Ultra\_kiri =tdkada\_obj and Ultra\_tengah = ada\_obj then  
mundur belok\_kiri*

*if Ultra\_kanan = tdkada\_obj and Ultra\_kiri = ada\_obj and Ultra\_tengah = ada\_obj then  
mundur belok\_kanan*

*'----- Navigasi Routine -----'*

*- Maju :*

*Pulsout SERVO\_R, 1000*

*Pulsout SERVO\_L, 2000*

*pause 20*

*goto cek\_sensor*

*belok\_kanan :*

*Pulsout SERVO\_R, 2000*

*Pulsout SERVO\_L, 2000*

*pause 20*

*goto cek\_sensor*

*belok\_kiri :*

*Pulsout SERVO\_R, 1000*

*Pulsout SERVO\_L, 1000*

*pause 20*

*goto cek\_sensor*

*mundur :*

*Pulsout SERVO\_R, 2000*

*Pulsout SERVO\_L, 1000*

*pause 20*

*goto cek\_sensor*

```

mundurbelok_kanan :
  gosub mundur_sebentar
for n = 1 to 30
  Pulsout SERVO_R, 2000
  Pulsout SERVO_L, 2000
  pause 20
  goto cek_sensor

```

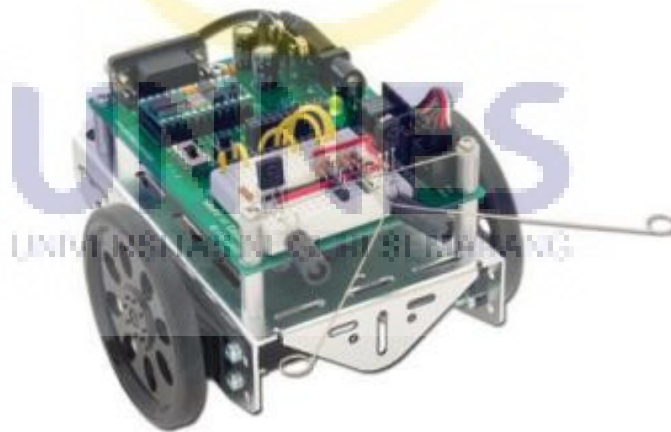
Dari keempat (program 2., program 3., program 4. dan program 5 ) di atas hanya sekedar contoh untuk menggerakkan mobil robot, namun bisa Anda kembangkan sendiri untuk aplikasi yang lain. Pemrograman ini merupakan program dengan bahasa PBasic yang dikembangkan oleh Parallax, Inc. Program yang ditulis dengan PBASIC akan disimpan dalam bentuk *token*, dalam sebuah EEPROM eksternal, kemudian akan dibaca, diterjemahkan dan di-eksekusi saat program dijalankan.

*Interpreter BASIC* ditanam dalam memori program di dalam *chip* mikrokontroler yang mereka jual (berbasis PIC atau *Ubcicom*). Dalam penggunaannya, Parallax menjual modul yang terdiri dari mikrokontroler, *EEPROM serial*, dan komponen pendukung lainnya. Modul ini dapat langsung dihubungkan dengan *port serial* atau *USB* komputer, dan melalui *BASIC Stamp Editor*, kita bisa langsung menggunakannya. *BASIC Stamp* merupakan mikrokontroler yang dikembangkan oleh Parallax Inc. Bahasa pemrograman yang dipakai pada basic stamp adalah bahasa pemrograman basic. Program dibuat pada *software basic stamp editor 2,4*. Program yang telah dibuat selanjutnya bisa diunduh melalui port serial. Mikrokontroler *BASIC Stamp* membutuhkan *power supply* saat mengunduh program dan program yang sudah diunduh tidak akan hilang meskipun baterai atau *power supply* dilepas.

## Bab 7

# Mobil Robot Produksi Parallax, Inc

Robot mobil Boe-Bot produksi perusahaan Parallax tersedia KIT Robotik untuk belajar dan latihan pemrograman. Robot mobil ini menggunakan 2 buah motor servo sebagai penggerak, *chassis* dari aluminium dan pengendali elektronika. Di atas roda motor dipasang roda plastik sebagai pelindung roda agar robot mobil mudah bergerak. Robot mobil ini dapat digerakan dan dipasang sensor untuk kebutuhan tertentu. Prosesor Basic Stamp dengan bahasa pemrograman PBasic dengan peralatan untuk pengembangan sistem robotik. Gambar A1. hanya sebagai salah satu contoh mobil robot produksi Parallax, karena masih banyak produksi mobil robot yang lain.

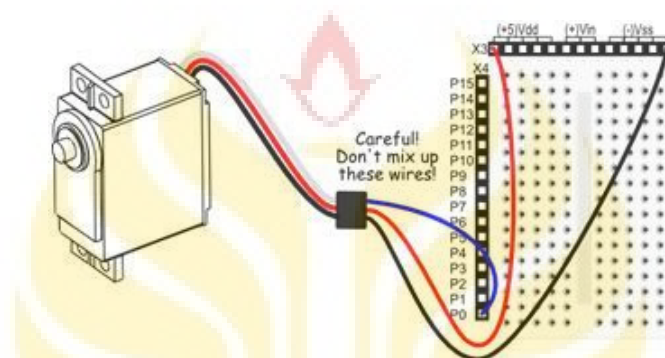


Gambar 7.1: Gambar Mobil Robot Produksi Parallax

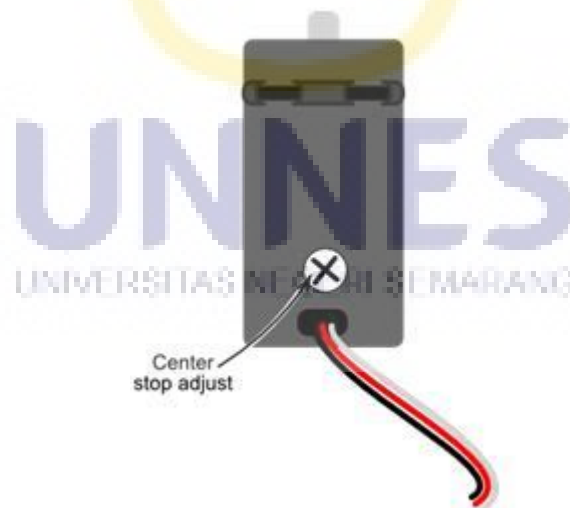
## 7.1 Konektor Motor Servo DC

Konektor Motor Servo untuk dikomunikasi dengan mikrokontroler Basic Stamp.

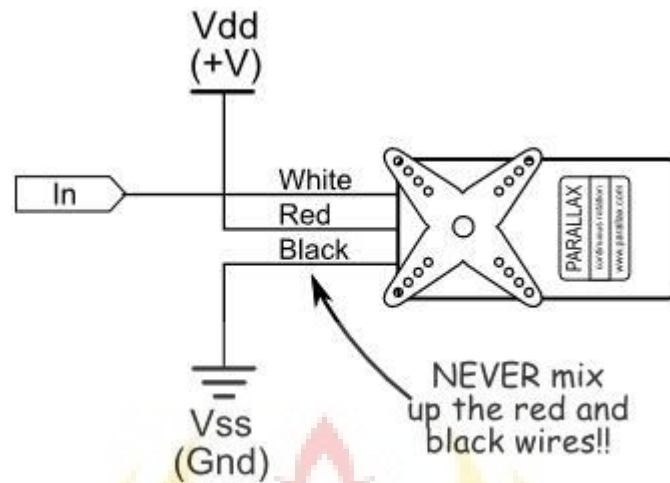
- 1) Memerlukan: tegangan 4 sampai dengan 6 VDC dan arus 15 sampai dengan 200 mA.
- 2) Komunikasi : PWM (*Pulse Width Modulation*)
- 3) Dimensi: 2.2 x 0.8 x 1.6 in (55.8x19x406 mm) tidak termasuk tangkai motor servo.



Gambar 7.2: Diagram Skema konektor Motor Servo DC



Gambar 7.3: Motor Servo DC dengan 3 kabel konektor



Gambar 7.4: Skema Diagram Konektor Motor Servo DC

## 7.2 Contoh Beberapa Gambar Sensor

### 7.2.1 Sensor Gelombang Ultrasonik

Sensor yang bekerja dengan cara memancarkan gelombang dan menghitung waktu pantulan gelombang tersebut. Secara umum sensor ultrasonik digunakan untuk menghitung jarak dari objek yang berada didepan sensor. Sensor ultrasonik biasa digunakan pada perangkat yang membutuhkan perhitungan jarak. Contoh : smart robot, Kapal laut, kapal selam dan lain-lainnya.

Sensor ultrasonik bekerja berdasarkan prinsip pantulan gelombang suara, dimana sensor ini menghasilkan gelombang suara yang kemudian menangkapnya kembali dengan perbedaan waktu sebagai dasar pengindraannya. Perbedaan waktu antara gelombang suara dipancarkan dengan ditangkapnya kembali gelombang suara tersebut adalah berbanding lurus dengan jarak atau tinggi objek yang memantulkannya. Jenis objek yang dapat diindra diantaranya adalah: objek padat, cair, butiran maupun tekstil.

Contoh Aplikasi :

Sensor ultrasonic banyak digunakan di berbagai perangkat pengukur jarak. sebagai contoh di dunia robotika sensor ini digunakan sebagai indra utama untuk navigasi robot.



sebagai contoh tipe ultra sonic yang banyak digunakan adalah tipe SRF, dan PING pada perinsipnya sensor jarak ultra sonic menggunakan prinsip kerja yang sama, yaitu pngirim sinyal dan penerima sinyal (transmitter and receiver). sensor ini bekerja pada frequency 40 Khz.



Gambar 7.5: Sensor Gelombang Ultrasonik

### 7.2.2 Sensor Inframerah

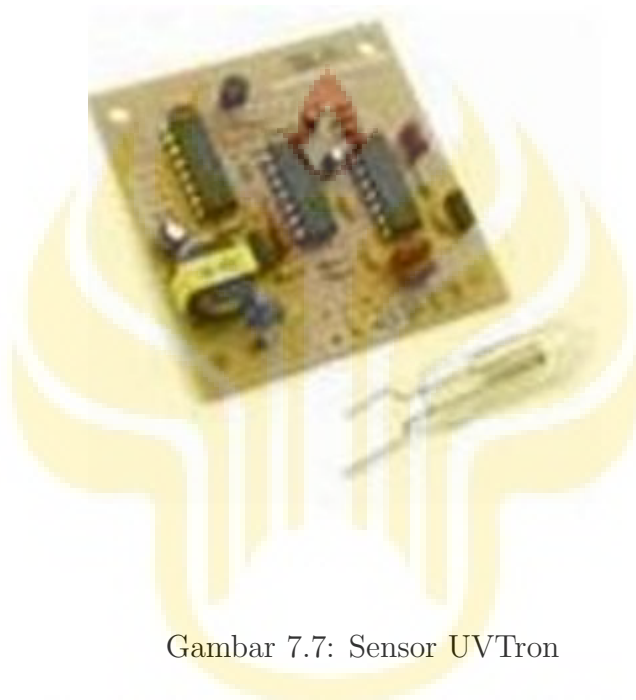
Sistem sensor ini pada dasarnya menggunakan inframerah sebagai media komunikasi yang menghubungkan antara 2 perangkat. Penerapan sistem sensor ini sangat bermanfaat sebagai operator jarak jauh, alarm keamanan dan otomatisasi sistem. Adapun transmisi sistem terdiri atas sebuah LED (*Lightemitting diode*) infra merah yang telah dilengkapi dengan jaringan yang mampu membangkitkan data untuk dikirimkan melalui sinar inframerah, sedangkan bagian penerima biasanya terdapat foto transistor, foto dioda atau modulasi yang berfungsi menerima sinar inframerah yang dikirimkan oleh pemancar.



Gambar 7.6: Sensor Inframerah

### 7.2.3 Sensor UV-Tron

Sensor api UV-Tron adalah sebuah sensor yang mendeteksi adanya nyala api yang memancarkan sinar *ultraviolet*. Pancaran cahaya *ultraviolet* dari sebuah nyala lilin berjarak 5 meter dapat dideteksi oleh sensor ini. Sensor api *UV-Tron* biasanya digunakan pada lomba robot, seperti KRCI (kontes robot cerdas indonesia) yang berfungsi mendeteksi keberadaan lilin yang akan dipadamkan oleh robot. Sensor UV-Tron mengeluarkan logika



Gambar 7.7: Sensor UVTron

*high* (1) jika mendeteksi api dan sebaliknya akan mengeluarkan logika *low* (0) jika tidak mendeteksi api, anda bisa mengecek keluarannya dengan multimeter analog. *Output* keluaran adalah sinyal kotak dengan frekuensi bergantung kapasitor *driver*. Pemilihan kapasitor *driver* harus disesuaikan kebutuhan, jika kita ingin mendapatkan output dengan sampling yang lebih cepat maka gunakan kapasitor dengan kapasitansi yang lebih kecil ( $0.01 \mu F$ ), sebaliknya jika ingin sampling lebih lambat gunakan kapasitansi lebih besar.

### 7.2.4 Sensor Kompas

Sensor kompas adalah sensor yang mampu mendeteksi arah secara horisontal terhadap medan magnet bumi. Sensor ini memanfaatkan efek magnetoresistive untuk mengetahui arah medan magnet yang melewati sensor. Efek magnetoresistive adalah perubahan ni-

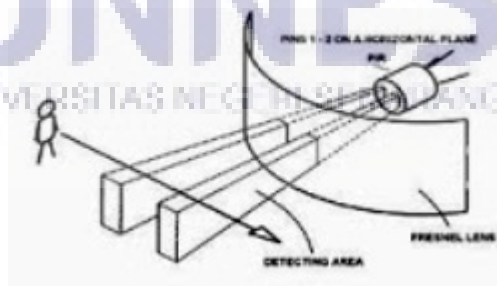
lai hambatan pada suatu penghantar akibat dari perubahan arah medan magnet yang melewatinya.



Gambar 7.8: Sensor Kompas

### 7.2.5 Sensor gerak atau PIR

Sensor gerak atau PIR (*Passive Infra Red*) mempunyai dua bagian utama. Bagian yang pertama pemancar infrared, sedangkan bagian yang kedua yaitu penerima. Bila alat sensor ini ada yang melewatinya bagian pemancar akan mengirim tanda atau sinyal ke bagian penerima. Selanjutnya, penerima akan memberi perintah pada alat lainnya. Misalnya membuka pintu atau mengeluarkan suara, tergantung system aplikasi yang diterapkan.



Gambar 7.9: Sensor gerak

Sensor gerak PIR (*Passive Infra Red*) adalah sensor yang berfungsi untuk pendeteksi

gerakan yang bekerja dengan cara mendeteksi adanya perbedaan/perubahan suhu sekarang dan sebelumnya. Sensor gerak menggunakan modul pir sangat simpel dan mudah diaplikasikan karena Modul PIR hanya membutuhkan tegangan input DC 5V cukup efektif untuk mendeteksi gerakan hingga jarak 5 meter. Ketika tidak mendeteksi gerakan, keluaran modul adalah LOW. Dan ketika mendeteksi adanya gerakan, maka keluaran akan berubah menjadi HIGH. Adapun lebar pulsa HIGH adalah 0,5 detik. Sensitifitas Modul PIR yang mampu mendeteksi adanya gerakan pada jarak 5 meter memungkinkan kita membuat suatu alat pendeteksi gerak dengan keberhasilan lebih besar.

Dengan output yang hanya memberikan 2 logika High dan Low ini kita dapat membuat aplikasi sensor gerak yang berfariatif. Misal kita ingin langsung aplikasikan pada alarm, kita tinggal membuat rangkaian driver untuk mengaktifkan alarm tersebut. Atau misal ingin digunakan untuk mengaktifkan lampu, maka tinggal di buat driver untuk memberikan sumber tegangan ke lampu. Modul sensor gerak PIR memiliki output yang langsung bbisa di hubungkan dengan komponen digital TTL atau CMOS dan juga dapat lansung di hubungkan ke mikrokontroler. Efektifitas pendeteksian gerakan menggunakan sensor gerak ini dipengaruhi oleh faktor penempatan sensor gerak PIR tersebut. Posisi sensor gerak harus diletakan pada lokasi yang dapat membaca semua gerakan yang ada dalam ruangan atau daerah yang dimonitor oleh sensor gerak PIR.

Contoh aplikasi :

Pintu yang bisa membuka sendiri secara otomatis Jika ada yang lewat atau masuk maupun keluar. Alat sensor tersebut melakukan deteksi terhadap suatu gerakan yang disebut dengan namanya adalah PIR.

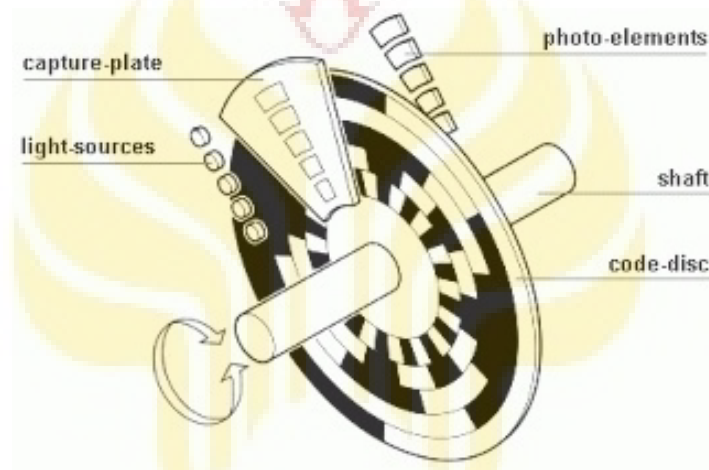
### **7.2.6 Sensor Penyandi (*Encoder*)**

Sensor penyandi (*Encoder*) merupakan salah satu sensor fisika yang digunakan untuk mengubah gerakan linear atau putaran menjadi sinyal digital, dimana sensor putaran memonitor gerakan putar dari suatu alat.

Sensor ini terdiri dari dua lapisan penyandi, yaitu sebagai berikut.  
Penyandi rotary (yang mentransmisikan jumlah tertentu dari pulsa untuk masing-masing

putaran ). Penyandi ini merupakan penyandi tambahan yang akan membangkitkan gelombang kotak pada objek yang diputar. Penyandi absolut (yang memperlengkapi kode binary tertentu untuk masing-masing posisi sudut ) mempunyai cara kerja sama dengan perkecualian, lebih banyak atau lebih rapat pulsa gelombang kotak yang dihasilkan sehingga membentuk suatu pengkodean dalam susunan tertentu.

Contoh Aplikasi : Salah satu aplikasi rotary encoder sebagai sensor posisi digunakan pada Mouse Analog (Mouse yang menggunakan Bola). Kurang lebih Tiga buah Rangkaian Sensor Posisi menggunakan Rotary Encoder.



Gambar 7.10: Sensor penyandi

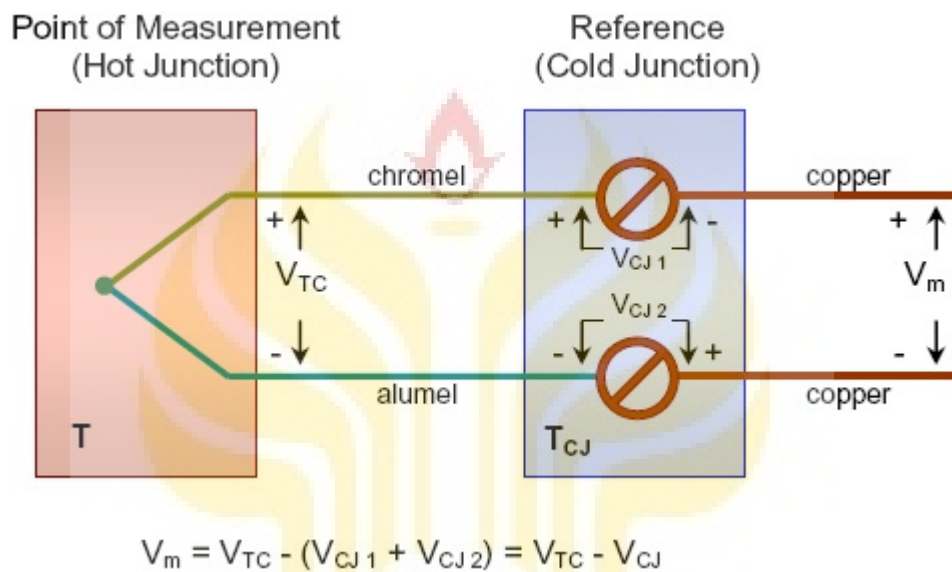
### 7.2.7 Sensor Suhu (Temperatur/ Thermal)

Ada 4 jenis utama dari sensor suhu yaitu *Thermocouple*, RTD, Thermistor, Sensor Suhu Rangkaian Terpadu (IC). Dari ke-4 jenis sensor suhu ini memiliki spesifikasi dan fungsi yang berbeda-beda dan juga mempunyai dan kelebihan masing-masing, mari kita bahas satu persatu.

#### A. *Thermocouple*

Pada intinya Thermocouple terdiri dari sepasang transduser panas dan dingin yang disambungkan dan dilebur bersama, dimana terdapat perbedaan yang timbul antara sambungan tersebut dengan sambungan referensi yang berfungsi sebagai pembanding.

Tipenya terdiri dari berbagai macam, antara lain : Tipe B, R, S, K, E, J, T yang disesuaikan dengan kebutuhan dunia industri. Disamping itu material protection tubenya pun tersedia dalam berbagai ukuran dan jenis material dari SUS 304, SUS 316, SUS 310, Sandvik P4, Inconel 600, Inconel 800, Titanium, UMC0 50, Alsint 99.7%, Pythagoras, Silicon Nitride, dan Silicon Carbide. Sedangkan untuk kabel dari thermocouple ke transmitter umumnya dibuat 1 pair cable (2 kabel). Adapun beberapa kelebihan



Gambar 7.11: *Thermocouple*

yang dimiliki oleh Thermocouple, antara lain :

- Spesifikasi lebih beragam
- Biaya rendah (low cost), dan Kisaran temperatur luas sehingga dapat disesuaikan sampai temperature tinggi.
- Waktu respon cepat

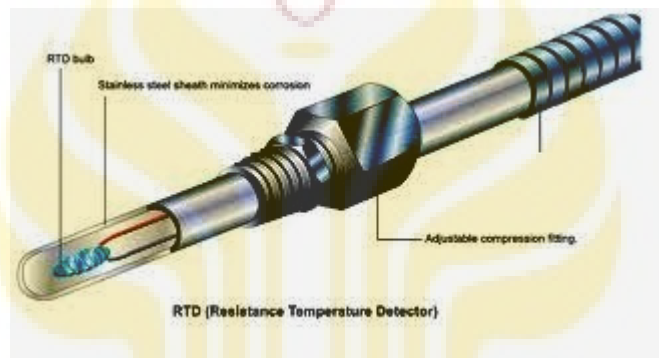
Sedangkan kekurangannya terdiri dari :

- Sensitivitasnya rendah
- Membutuhkan suhu referensi
- *Nonlinearity*

- Terbatasnya akurasi sistem kesalahan kurang dari 1°C yang sulit dicapai.

### B. *Resistance Temperature Detector*

Resistance Temperature Detector (RTD) atau dikenal dengan Detektor Temperatur Tahanan adalah sebuah alat yang digunakan untuk menentukan nilai atau besaran suatu temperatur/suhu dengan menggunakan elemen sensitif dari kawat platina, tembaga, atau nikel murni, yang memberikan nilai tahanan yang terbatas untuk masing-masing temperatur di dalam kisaran suhunya. Semakin panas benda tersebut, semakin besar atau semakin tinggi nilai tahanan listriknya, begitu juga sebaliknya. PT100 merupakan tipe RTD yang paling populer yang digunakan di industri.



Gambar 7.12: *Thermocouple*

Resistance Temperature Detector merupakan sensor pasif, karena sensor ini membutuhkan energi dari luar. Elemen yang umum digunakan pada tahanan resistansi adalah kawat nikel, tembaga, dan platina murni yang dipasang dalam sebuah tabung guna untuk memproteksi terhadap kerusakan mekanis. Resistance Temperature Detector (PT100) digunakan pada kisaran suhu -200 0C sampai dengan 650° C. Prinsip dasar RTD adalah jika pada tahanan listrik dari logam yang bervariasi sebanding dengan suhu. Kesebandingan variasi ini adalah presisi dengan tingkat konsisten/kestabilan yang tinggi pada pendeteksian tahanan. Platina adalah bahan yang sering digunakan karena memiliki tahanan suhu, kelinearan, stabilitas dan reproduksibilitas.

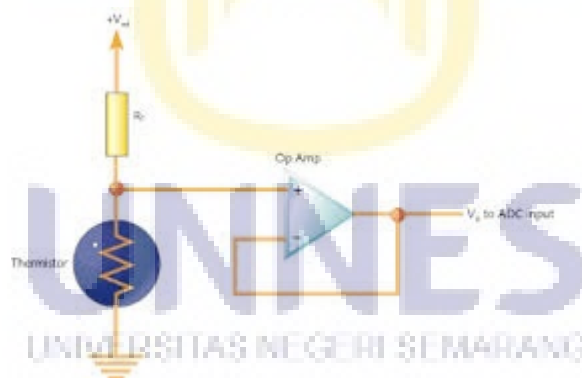
### C. *Thermistor*

Thermistor adalah salah satu jenis sensor suhu yang mempunyai koefisien temperatur yang sangat tinggi. Fungsi utama dari komponen ini dalam suatu rangkaian elek-

tronik adalah untuk mengubah nilai resistansi karena adanya perubahan temperatur dalam rangkaian tersebut. Karakteristik yang demikian ini memungkinkan kita untuk dapat mengatasi beberapa masalah yang sederhana, seperti misalnya yang berkaitan dengan sensor temperatur, kompensasi temperatur, atau masalah sistem pengaturan yang lain.

Thermistor ini dibedakan dalam tiga jenis, yaitu thermistor yang mempunyai koefisien temperatur negatif yang biasa disingkat NTC (*Negative Temperature Coefficient*), thermistor yang mempunyai koefisien temperatur positif yang biasa disingkat PTC (*Positive Temperature Coefficient*), dan thermistor yang mempunyai tahanan temperature kritis yang biasa disingkat CTR (*Critical Temperature Resistance*).

Ketiga jenis thermistor ini masing-masing mempunyai kegunaan yang berbeda, karena karakteristik dari ketiga jenis termistor tersebut berbeda antara yang satu dengan yang lain. Akan tetapi, pada umumnya, bila kita menyebut kata termistor, maka termistor yang dimaksud adalah termistor NTC.



Gambar 7.13: *Thermistor*

- NTC (*Negative Temperature Coefficient*)

NTC adalah resistor yang mempunyai koefisien temperatur negatif yang sangat tinggi. Thermistor jenis ini dibuat dari oksida logam yang terdapat dalam golongan transisi. Oksida-oksida ini sebenarnya mempunyai resistansi yang tinggi,



tetapi dapat diubah menjadi bahan semikonduktor yaitu dengan menambahkan beberapa ion lain (sebagai doping) yang mempunyai valensi yang berbeda. Sedangkan perubahan resistansinya karena pengaruh perubahan temperatur diberikan dalam bentuk kurva resistansi sebagai fungsi temperatur.

- PTC (*Positive Temperature Coefficient*)

PTC merupakan resistor dengan koefisien temperatur positif yang sangat tinggi. Dalam beberapa hal, thermistor PTC berbeda dengan termistor NTC antara lain seperti yang dijelaskan berikut ini:

- Koefisien temperatur dari thermistor PTC bernilai positif hanya dalam interval temperatur tertentu, sehingga di luar interval tersebut, koefisien temperturnya bisa bernilai nol atau negatif.
- Pada umumnya, harga mutlak dari koefisien temperatur dari thermistor PTC jauh lebih besar dari pada thermistor NTC.

- CTR (*Critical Temperature Resistance*)

Thermistor CTR dibuat dari  $V_2O_3$  yang dipanaskan dengan serbuk oksida Ba atau oksida Si dan sebagainya, yang hasilnya dalam bentuk kaca. Thermistor jenis ini merupakan resistor yang mempunyai koefisien temperatur negatif yang sangat tinggi. Penurunan resistansi yang drastis karena adanya pengaruh suhu tersebut terjadi pada transisi logam-semikonduktor dan berubah-ubah tergantung (sebagai fungsi) dari konsentrasi dopant, yaitu oksida logam, seperti Ge, Ni, W, atau M.

IC Sensor adalah sensor suhu dengan rangkaian terpadu yang menggunakan chipsilikon untuk kelemahan penginderanya. Mempunyai konfigurasi output tegangan dan arus yang sangat linear. Ada beberapa jenis IC yang sering digunakan sebagai sensor suhu seperti LM135, LM235, LM335. Ketiga jenis IC ini memiliki karakter yang berbeda-beda dan penggunaannya disesuaikan dengan kebutuhan kita.

### 7.2.8 Sensor Cahaya

Sensor cahaya adalah komponen elektronika yang dapat memberikan perubahan besaran elektrik pada saat terjadi perubahan intensitas cahaya yang diterima oleh sensor cahaya

tersebut. Sensor cahaya dalam kehidupan sehari-hari dapat kita temui pada penerima remote televisi dan pada lampu penerangan jalan otomatis.



Gambar 7.14: LDR (*Light Depending Resistor*)

#### Jenis-Jenis Sensor Cahaya

A. Dilihat dari perubahan output sensor cahaya maka sensor cahaya dapat dibedakan kedalam 2 tipe yaitu :

- Sensor cahaya tipe fotovoltaiik
- Sensor cahaya tipe fotokonduktif

B. Kemudian apabila dilihat dari cahaya yang diterima sensor cahaya tersebut, maka sensor cahaya dapat dibagi dalam beberapa tipe sebagai berikut :

- Sensor cahaya infra merah
- Sensor cahaya ultraviolet

#### 1. Sensor cahaya tipe fotovoltaiik

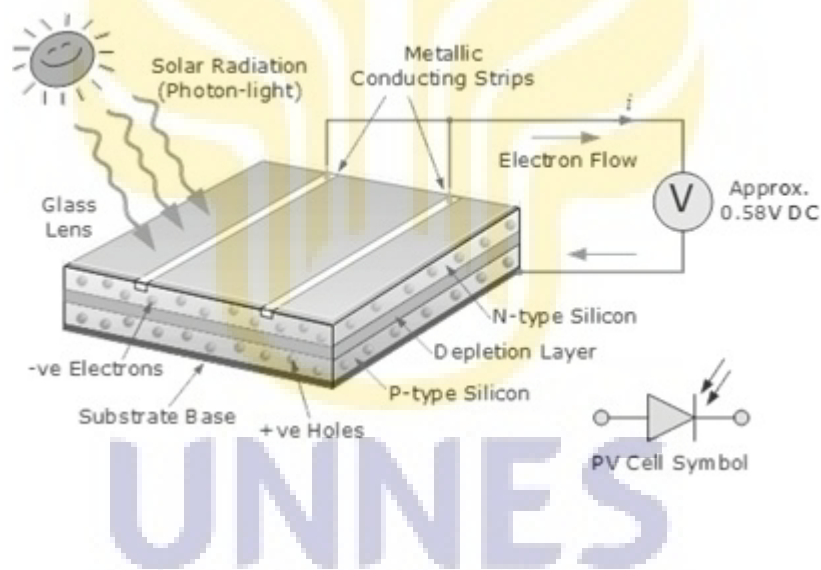
Sensor cahaya tipe fotovolataik adalah sensor cahaya yang dapat memberikan perubahan tegangan pada output sensor cahaya tersebut apabila sensor tersebut menerima intensitas cahaya. Salah satu contoh sensor cahaya tipe fotovoltaiik adalah solar cell atau sel surya.

Sensor cahaya tipe photovoltaic adalah alat sensor sinar yang mengubah energi sinar langsung menjadi energi listrik. Sel solar silikon yang modern pada dasarnya adalah sambungan PN dengan lapisan P yang transparan. Jika ada cahaya pada lapisan

transparan P akan menyebabkan gerakan elektron antara bagian P dan N, jadi menghasilkan tegangan DC yang kecil sekitar 0,5 volt per sel pada sinar matahari penuh. Berikut konstruksi dari sensor cahaya tipe fotovoltaik.



Gambar 7.15: sel surya



Gambar 7.16: konstruksi dari sensor cahaya tipe fotovoltaik

## 2. Sensor Cahaya Fotokonduktif

Sensor cahaya tipe fotokonduktif akan memberikan perubahan resistansi pada terminal outputnya sesuai dengan perubahan intensitas cahaya yang diterimanya. Sensor cahaya tipe fotovoltaik ini ada beberapa jenis diantaranya adalah :

- LDR (*Light Depending Resistor*)
- Photo Transistor

- Photo Dioda

LDR adalah sensor cahaya yang memiliki 2 terminal output, dimana kedua terminal output tersebut memiliki resistansi yang dapat berubah sesuai dengan intensitas cahaya yang diterimanya. Dimana nilai resistansi kedua terminal output LDR akan semakin rendah apabila intensitas cahaya yang diterima oleh LDR semakin tinggi.

### 3. Photo Transistor

Photo transistor adalah suatu transistor yang memiliki resistansi antara kaki ko-



Gambar 7.17: Photo transistor

lektor dan emitor dapat berubah sesuai intensitas cahaya yang diterimanya. Photo transistor memiliki 2 terminal output dengan nama emitor dan kolektor, dimana nilai resistansi emitor dan kolektor tersebut akan semakin rendah apabila intensitas cahaya yang diterima photo transistor semakin tinggi.

### 4. Photo Dioda

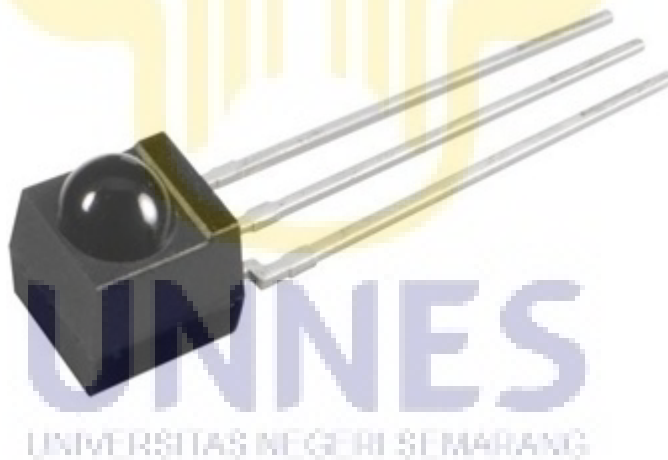
Photo dioda adalah suatu dioda yang akan mengalami perubahan resistansi pada terminal anoda dan katoda apabila terkena cahaya. Nilai resistansi anoda dan katoda pada photo dioda akan semakin rendah apabila intensitas cahaya yang diterima photodioda semakin tinggi.



Gambar 7.18: Photo dioda

#### 5. Sensor Cahaya Infra Merah

Sensor cahaya infra merah adalah sensor cahaya yang hanya akan merespon peru-



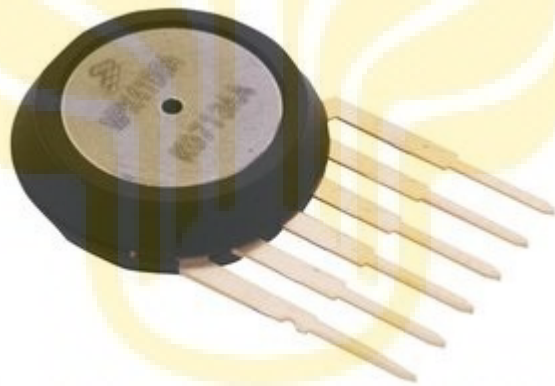
Gambar 7.19: Sensor infra merah

bahan cahaya inframerah. Sensor cahaya infra merah pada umumnya berupa photo transistor atau photo dioda. Dimana apabila sensor cahaya infra merah ini menerima pancaran cahaya infra merah maka pada terminal outputnya akan memberikan perubahan resistansi. Akan tetapi ada juga sensor cahaya yang telah dibuat dalam bentuk chip IC penerima sensor infra merah seperti yang digunakan pada penerima

remote televisi. Dimana chip IC sensor infra merah ini akan memberikan perubahan tegangan output apabila IC sensor infra merah ini menerima pancaran cahaya infra merah. Berikut adalah bentuk dari IC sensor infra merah tersebut.

### 7.2.9 Sensor Tekanan MPX4100

Sensor tekanan MPX4100 merupakan seri Manifold Absolute Pressure (MAP) yaitu sensor tekanan yang dapat membaca tekanan udara dalam suatu manifold. Pada dasarnya sensor tekanan MPX4100 adalah sebuah sensor tekanan yang sudah dilengkapi dengan rangkaian pengkondisi sinyal dan temperatur kalibrator yang membuat sensor ini stabil terhadap perubahan suhu. Untuk akurasi pengukuran sensor ini menggunakan teknik micro machine, thin film metalization dan proses bipolar semiconductor. Bentuk fisik sensor tekanan MPX4100 cukup kecil seperti terlihat pada gambar berikut. Bentuk fisik dari

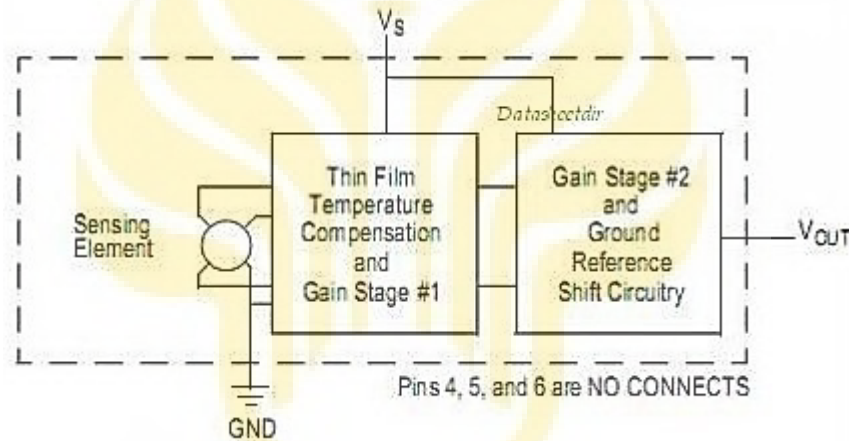


Gambar 7.20: Bentuk fisik dari sensor tekanan MPX4100

sensor tekanan MPX4100 cukup kecil sehingga dapat digunakan dengan lebih praktis dan efisien tempat peletakan sensor tekanan MPX4100 tersebut. Dengan adanya rangkaian pengkondisi sinyal, sensor ini dapat terhubung langsung pada Analog to Digital Converter. Rangkaian pengkondisi sinyal menghasilkan tegangan analog dengan Skala Penuh (Full Scale) hingga 5 volt.

Feature sensor tekanan MPX4100 Sesuai datasheet dari sensor tekanan, feature yang dimiliki oleh sensor tekanan tipe MPX4100 ini adalah sebagai berikut :

- a. 1.8% Maximum Error Over 0° to 85°C
- b. Specifically Designed for Intake Manifold Absolute Pressure Sensing in Engine Control Systems
- c. Ideally Suited for Microprocessor Interfacing
- d. Temperature Compensated Over -40°C to +125°C
- e. Durable Epoxy Unibody Element
- f. Ideal for Non-Automotive Applications

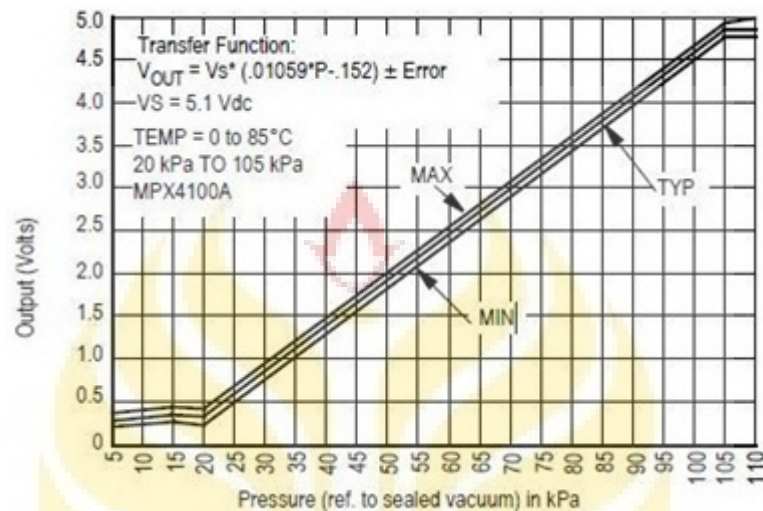


Gambar 7.21: Diagram Blok Internal Sensor Tekanan MPX4100

Sensor ini mempunyai kemampuan untuk mendeteksi tekanan 15 hingga 115 kilo Pascal dan bekerja berdasarkan perbedaan tekanan antara P1 dan P2. P1 atau Pressure Side terdiri dari fluorosilicone gel yang melindunginya dari benda-benda keras. Gambar 7.22 menunjukkan perubahan tekanan terhadap tegangan output dari sensor di mana perubahan bergerak linear setelah 20 kPa. Tampak 3 buah garis pada grafik tersebut yang menunjukkan batas maksimum dan minimum error dari hasil pengukuran sensor. Sensor tekanan pada aplikasi robotik seringkali digunakan sebagai feedback mechanic di mana sistem mikrokontroler dapat mendeteksi kondisi mekanik pada saat itu. Contohnya untuk mendeteksi kuat lemah cengkeraman robot atau menghitung beban yang diletakkan pada robot. Selain itu pengukuran tekanan kompresi pada manifold mesin (otomotif) sering

menggunakan sensor tekanan MPX4100 ini karena tetap stabil dalam perubahan suhu yang tinggi.

Read more at: <http://elektronika-dasar.web.id/sensor-tekanan/> Copyright Elektronika Dasar



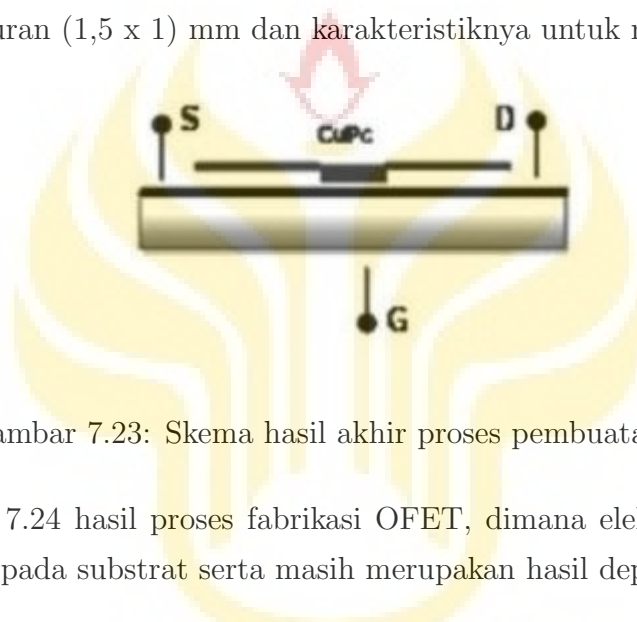
Gambar 7.22: Grafik Tegangan Output Sensor Tekanan MPX4100

### 7.2.10 Sensor gas

Penulis juga telah melakukan penelitian (PENELITIAN HIBAH BERSAING dengan dana DIKTI) yang berkaitan dengan deposisi film tipis CuPc untuk pembuatan OFET (transistor efek medan organik) sebagai sensor gas beracun. Judul penelitian adalah: " Pengembangan Transistor Efek Medan Organik (OFET) Berbasis Film Tipis Dengan Material CuPc Untuk Mendeteksi Gas Beracun " dilakukan pada tahun 2013. Dalam buku penulis tidak mengupas secara lengkap dari proses fabrikasi OFET, namun hanya sekedar hasil yang telah dicapai. Adapun tujuan penelitian: fabrikasi OFET berbasis film tipis semikonduktor organik CuPc (*copper phthalocyanine*) dengan variasi panjang *channel* (saluran), karakterisasi OFET dengan X-RD dan SEM dan membuat alat uji sensor gas dan pengujiannya, meliputi: waktu respon (*respon time*) dan waktu pulih (*recovery time*). Adapun hasil penelitian fabrikasi OFET yang diperoleh sebagai berikut:



- 1, Material yang digunakan adalah semikonduktor organik CuPc untuk membuat transistor efek medan organik (OFET) berbasis film tipis.
- 2, Eksperimen pendahuluan dengan melakukan penumbuhan film tipis CuPc di atas substrat Si/ SiO<sub>2</sub>, kemudian dilanjutkan dengan karakterisasi menggunakan XRD dan SEM. Hasil dari karakterisasi X-RD dan SEM digunakan untuk menentukan film tipis optimum dalam proses pembuatan OFET.
- 3, Tujuan jangka pendek penelitian ini adalah fabrikasi OFET dengan struktur *bottom-contact* berukuran (1,5 x 1) mm dan karakteristiknya untuk mendeteksi gas beracun.



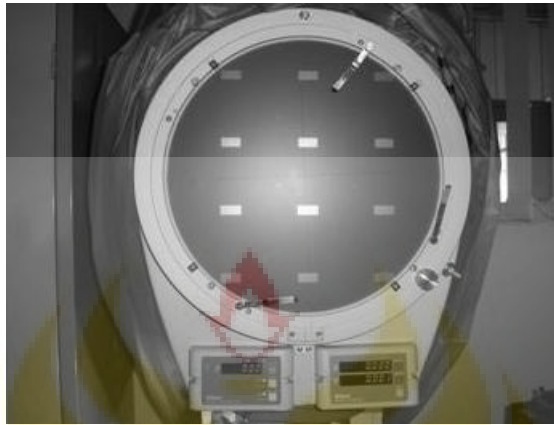
Gambar 7.23: Skema hasil akhir proses pembuatan OFET

Pada Gambar 7.24 hasil proses fabrikasi OFET, dimana elektrode *source* dan *drain* belum terpola pada substrat serta masih merupakan hasil deposisi Au (emas).



Gambar 7.24: Hasil proses pembuatan OFET

Gambar 7.25 hasil proses pembuatan OFET, dimana elektrode *source/ drain* belum terpola dan belum dilakukan pemotongan tiap sampelnya.



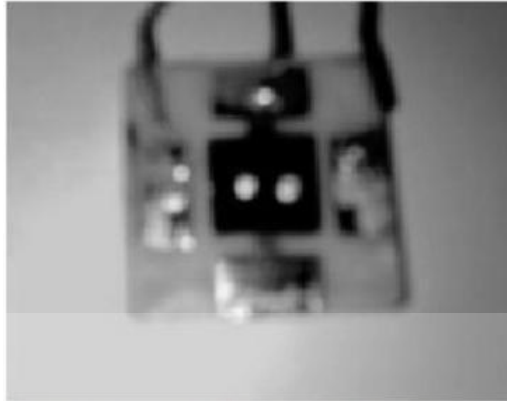
Gambar 7.25: Hasil pemotongan substrat pada proses pembuatan OFET

Gambar 7.26 proses pembuatan OFET, dimana elektrode *source* dan *drain* sudah terpola dan sudah dilakukan pemotongan tiap sampelnya.



Gambar 7.26: Hasil akhir proses pembuatan OFET

Setelah akhir pembuatan OFET, selanjutnya dilakukan pemotongan setiap sampel. Untuk memudahkan, maka sampel OFET dipasang di atas PCB dengan menggunakan *gold wire* dan *silver pasta* dan dipasang kontak resistansi. Pada (Gambar 7.27). adalah OFET yang sudah dilakukan pemotongan dan dipasang pada PCB.



Gambar 7.27: Hasil pembuatan OFET terpasang pada PCB

- 4, Fabrikasi OFET dengan panjang *channel* yang bervariasi, yaitu:  $100\ \mu\text{m}$  ,  $200\ \mu\text{m}$  dan  $300\ \mu\text{m}$  menggunakan metode *vacuum evaporator* (VE), sedangkan proses perencanaan dengan teknik *lithography*.
- 5, Tahapan pembuatan OFET, sebagai berikut: permulaan dilakukan pencucian substrat Si/SiO<sub>2</sub> dengan etanol dalam *ultrasonic cleaner*, kemudian dilakukan pendeposisian elektroda *source* dan *drain* di atas substrat dengan metode VE. Selanjutnya mendeposisikan film tipis diantara *source/drain* dan diakhiri dengan deposisi *gate*.
- 6, OFET yang sudah dibuat kemudian karakterisasi menggunakan El-Kahfi 100 untuk menentukan karakteristik keluaran dan mobilitas pembawa muatan.
- 7, Karakterisasi OFET diperoleh bahwa daerah aktif untuk  $V_D$  adalah (2,79 V sampai dengan 3,43 V) dan kuat arus  $I_D$  ( $0,577 \cdot 10^{-5}$  A sampai dengan 0,0013 A). Sedangkan untuk daerah saturasi OFET pada tegangan  $V_D$  dari 3,43 V sampai dengan 9 V dan ini merupakan daerah *cut off*.
- 8, Panjang *channel* pada OFET berpengaruh terhadap yang dihasilkan. Semakin pendek channel pada OFET, semakin besar arus ( $I_D$ ) yang dihasilkan.
- 9, OFET dengan panjang *channel* yang berbeda : 100, 200 dan  $300\ \mu\text{m}$  mempunyai mobilitas pembawa muatan berturut-turut: (0,10121664 , 0.030546453, 0.000307692)  $\text{cm}^2\ \text{V}^{-1}\ \text{s}^{-1}$ .

10, *Respon time* OFET terhadap gas uji CO dengan volume 1 cc untuk panjang *channel* :100  $\mu\text{m}$  , 200  $\mu\text{m}$  dan 300  $\mu\text{m}$  berturut-turut: 90 s, 120 s dan 135 s. Sedangkan *recovery time* pada OFET berturut-turut: 120 s, 135 s dan 150 s.

Perlu penulis sampaikan bahwa pembahasan buku ini hanya terbatas pada metode deposi dengan metode penguapan hampa udara pada suhu ruang dan hasil karakterisasi film tipis menggunakan SEM/ X-RD, sedangkan yang lainnya belum diungkapkan. Harapan penulis dengan terbitnya buku ini dapat menjadikan inspirasi bagi para pembaca untuk mendalami masalah deposisi film tipis, pembuatan divais dan aplikasinya. Semoga pembaca tertarik untuk mengembangkan film tipis dan aplikasinya dengan hasil penelitian/ riset yang jauh lebih baik.

Teknologi di bidang mikromaterial sangat menentukan kemajuan bangsa di masa akan datang, baik bidang sains, teknologi maupun ekonomi oleh karena itu kita tidak bisa menghindarinya. Sejumlah negara banyak meningkatkan riset karena menyadari hal itu sangat penting untuk meraih keunggulan riset tersebut. Namun peralatan yang diperlukan untuk mengembangkan riset ini, khususnya peralatan karakterisasi, material semikonduktor, deposisi film tipis dan *lithography* sangat mahal. Peralatan seperti *X-Ray Diffraction* (X-RD) dan *Scanning Electron Microscopy* (SEM) merupakan contoh peralatan standar untuk karakterisasi mikromaterial/ nanomaterial dan harganya sangat mahal. Hingga saat ini pemerintah kita belum mampu menyediakan semua peralatan yang dibutuhkan. Di lain pihak, riset tersebut tetap dikembangkan di perguruan tinggi dan lembaga penelitian.

Penulis bertujuan memberikan alternatif untuk memahami karakterisasi film tipis yang dapat kita lakukan dalam kondisi keterbatasan alat. Memang terkadang hasil riset yang kurang eksak, tetapi cukup memberikan informasi tentang sifat-sifat mikromateria/ nanomaterial yang sedang kita pelajari. Jelas, hal ini lebih baik dari pada kita diam saja tidak melakukan apa-apa dan mengeluh karena keterbatasan peralatan yang ada.

## Bab 8

# Penutup

Buku ini diharapkan untuk menjadi panduan bagi pembaca yang belajar atau pemula yang ingin mempelajari dan mengembangkan mobil robot yang dapat mendeteksi adanya objek dan dapat menghindarinya adanya halangan. Mobil robot yang dibahas dalam buku ini dilengkapi sensor gelombang ultrasonik sebagai deteksi objek/halangan atau rintangan pergerakan mobil robot. Sensor gelombang ultrasonik mempunyai 2 fungsi, yaitu memancarkan gelombang ultrasonik yang akan dipantulkan kembali oleh suatu objek/halangan dan diterima oleh suatu *receiver* (R). Kemudian gelombang pantul itu akan diubah menjadi bentuk tegangan oleh transistor oleh mikrokontroler. Mikrokontroler yang digunakan adalah *Basic Stamp* dengan bahasa pemrograman *Parallax Basic (PBasic)*.

Untuk mengontrol semua gerakan robot mobil, berjalan lurus, berbelok ke kiri/kanan, adalah melalui program yang dimasukkan ke prosesor/mikrokontroler. Melalui mikrokontroler ini, semua gerakan robot mobil dapat diprogram sesuai dengan yang diinginkan, mulai dengan mengontrol motor DC sampai menerima hasil respon dari sensor yang kemudian akan diproses menjadi berbagai macam *output*. Tegangan keluaran ini akan diteruskan kepada sistem pembanding. Perbandingan inilah yang kemudian akan diberikan kepada motor servo untuk menggerakkan roda sesuai program pada mikrokontroler. Mobil Robot akan bergerak menelusuri lintasan tertentu dan dapat menghindari benturan dengan objek/rintangan. Ketika robot mobil mendeteksi adanya objek/halangan pada lintasan, maka akan bergerak memutar atau mundur untuk menghindar halangan dan kembali kelintasan yang sudah ditentukan.

Diharapkan pembahasan dalam buku ini dapat menjadi gambaran bagaimana mengendalikan suatu robot beroda (robot mobil) menggunakan beberapa komponen dan sensor sebagai pendukungnya. Disamping untuk memperkenalkan mikrokontroler yang dapat digunakan yaitu *Basic Stamp* diproduksi oleh perusahaan Parallax , sehingga ada pandangan yang luas untuk memilih berbagai tipe mikrokontroler. Pembaca mampu menghasilkan program menggerakkan robot dengan lintasan bebas tanpa menabrak objek yang ada.

## 8.1 KESIMPULAN

Sifat sensor ultrasonik yang dapat memancarkan dan menerima pantulan gelombang ultrasonik dapat digunakan untuk mendeteksi benda. Agar pantulan gelombang ultrasonik dari objek dapat mengenai penerima, maka perlu diatur posisi antara sensor penerima maupun pemancar. Dengan penempatan sensor ultrasonik secara tepat, maka benda yang dideteksi dapat dibedakan dan diketahui posisinya. Dari uraian di atas dapat disimpulkan bahwa :

- 1) Semakin dekat jarak antara sensor penerima dan pemancar gelombang ultrasonik semakin jauh objek yang dapat dideteksi.
- 2) Pemansangan sensor ultrasonik pada robot beroda secara vertikal atau horisontal akan mempengaruhi jarak objek yang dapat dideteksi.
- 3) Panjang loop yang dipasang pada sensor ultrasonik akan mempengaruhi jarak objek yang dapat dideteksi.
- 4) Sensor ultrasonik yang digunakan dapat mendeteksi jarak obyek terjauh 85cm, sedangkan jarak terdekat 0.5 cm.
- 5) Panjang gelombang ultrasonik mempengaruhi ukuran objek terkecil yang dapat dideteksi sensor
- 6) Panjang loop yang di pasang pada robot beroda 2cm jarak antara transduser satu dengan yang lainnya 1.5 cm.

## 8.2 SARAN

- a) Sebaiknya pemasangan transuder dari penerima serapat mungkin dan permukaan dari transuder membentuk garis parabola.
- b) Dengan meningkatkan frekuensi pemancar gelombang ultrasonik dapat mendeteksi objek yang berukuran lebih kecil.
- c) Dapat digunakan dua mikrokontroler untuk tingkah laku robot yang lebih kompleks, sehingga proses pengolahan sinyal kontrol dan fungsi-fungsi tambahan seperti pembacaan jarak maupun pengaturan nilai PWM dapat dilakukan secara terpisah.
- d) Pengembangan lebih lanjut teknik pemasangan sensor gelombang ultrasonik ini dapat diaplikasikan pada penceritaan medis.
- e) Penambahan sensor jarak pada robot mobil ini akan memperluas range pendeteksian benda yang akan dideteksi.



# Daftar Pustaka

Agfianto. 2002. *Belajar mikrokontroler AT89C51/52/53 (Teori dan Aplikasinya)*. Penerbit Gava Media.Yogyakarta.

Anonim. 2000. *Basic Stamp*. Parallax, Inc .California.

Joseph J.Corr.1993, *N Sors, Transducers, and Supporting Circuit For Electronics Instrumentation Measurent and Control*, Prentice Hall.inc.Englewood Cliffs.New Jersey 07632.USA.

M.O.Tjia, 1994, *Gelombang*, Dabara Publishers.Solo.

Masturi dan Sujarwata, 2008, *Penelitian Sistem Robotika Untuk Pengembangan Laboratorium Elektronika Dan Instrumentasi Jurusan Fisika FMIPA UNNES Semarang*

Sujarwata dan Kuwat Triyana, 2010, Studi Penumbuhan Film Tipis CuPc Dengan Metode Penguapan Hampa Udara Pada Suhu Ruang Untuk Aplikasi Sensor Gas, *Saintekmol (Jurnal Sain dan Teknologi)*, Vol 8 No.2.

Sujarwata dan Putut Marwoto, 2013, *Pengembangan Transistor Efek Medan Organik (OFET) Berbasis Film Tipis Dengan Material CuPc Untuk Mendeteksi Gas Beracun*. Laporan penelitian Hibah Bersaing, DIPA Universitas Negeri Semarang Nomor: DIPA-023.04.2.189822/2013.

Sujarwata dan Putut Marwoto, 2013, Sensor Gas Berbasis Film Tipis Dengan Konfigurasi Transistor Efek Medan (FET) Untuk Deteksi Gas CO, *Saintekmol (Jurnal Sain dan Teknologi)*, Vol 11 No.2.

Sutrisno. 1987. *Elektronika Lanjutan. Diklat kuliah Fisika ITB*. Bandung.kedua. Alih bahasa Sutisna. Penerbt Erlangga. Jakarta.



# Indeks

- Respon time*, 104  
*recovery time*, 104  
*X-Ray Diffraction*, 104
- BASIC Stamp, 68, 81  
Basic Stamp, 41, 64, 70, 72, 73, 105, 106  
Basic Stamp BS2X, 40  
BASIC Stamp Editor, 64, 68, 71  
baudrate, 72, 73  
delay, 76  
EEPROM, 64  
Fotoresistor, 37  
gelombang ultrasonik, 52, 53  
gelombang ultrasonik, 48–51, 55, 57, 62, 64, 68  
Interpreter BASIC, 81  
LDR, 37–40, 43  
Light Dependent Resistor, 39, 43  
Line Follower, 61  
loop, 54, 55  
loop, 51, 54, 55, 62, 66, 68, 76  
mikrokontoler, 72  
mikrokontroler, 47, 48, 73, 81, 106  
mobil robot, 51, 61, 62, 64, 65, 67, 105  
OFET, 104  
Parallax, 106  
Parallax Basic, 64  
Parallax Basic, 105  
PBasic, 75, 105  
Pbasic, 43, 48, 64  
Piezoelektrik, 47  
Port Serial/ RS-232, 74  
power supply, 81  
RCTime, 41–43  
Receiver, 47, 50, 75  
robot, 48, 58, 59, 61, 106  
robotik, 47, 48, 58, 59, 61  
sensor, 37, 38, 49, 52–55, 57, 64, 67, 68, 70  
Sensor Gelombang Ultrasonik, 46  
servo motor, 65  
software, 59, 81  
Tokenzied is successfully, 64  
Transmitter, 47, 50, 75

# Glosarium

- Basic Stamp Editor** : *Software beroperasi pada sistem operasi windows dan dapat berjalan pada komputer dengan sistem minimum, tanpa harus membutuhkan spesifikasi komputer yang canggih.*
- Bit** : *Merupakan singkatan dari Binary Digit. Sebuah bit atau digit biner adalah sebuah sinyal yang masing-masing berada dalam dua kondisi yaitu 0 dan 1.*
- Byte** : *Dalam system computer byte bilangan biner sebanyak 8 bit dan 1 karakter dapat disebut dengan satu byte. Pemakaian yang luas dalam komunikasi data dalam system computer mikro disebut American Standar For Information Interchange (ASCII).*
- Clock** : *Pulsa lonceng yang berbentuk gelombang kotak yang tidak mempunyai kondisi setimbang untuk mengontrol serialisasi dan deserialisasi.*
- Data** : *Kata lain dari informasi yang dipakai secara luas yang dibatasi hanya untuk segala hal yang dapat diungkapkan dalam bentuk nomor-nomor dan huruf-huruf. Untuk menyajikan ungkapan dalam bentuk nomor-nomor atau digit-digit harus dalam bentuk sinyal yang disebut sinyal digital.*
- DEBUG** : *Merupakan perintah untuk menampilkan informasi pada layar PC. Seluruh informasi dapat ditampilkan termasuk informasi dari sensor ataupun informasi yang diketik dengan diberi awal dan akhiran tanda kutip dua ( *informasi* ). Informasi akan muncul apabila baudrate pada BASIC Stamp telah disesuaikan.*
- Downloader** : *Peralatan elektronika yang berfungsi untuk memasukkan bahasa pemrograman yang telah dibuat kedalam mikrokontroler.*
- DO** : *Merupakan perintah untuk memulai suatu program yang diakhir program harus ditutup dengan perintah LOOP.*
- EEPROM** : *Memori untuk penyimpanan data secara permanen oleh program yang sedang running.*
- Flash** : *Jenis Read Only Memory yang biasanya diisi dengan program hasil buatan manusia yang harus dijalankan oleh mikrokontroler.*

**Flip-Flop** : Rangkaian bistabil multivibrator yang mempunyai dua keadaan setimbang mantap.

**Gelombang ultrasonik** : Gelombang akustik yang memiliki frekuensi mulai 20 kHz sampai dengan 20 MHz.

**GOSUB** : Merupakan kependekan dari GO to SUBroutine.

**GOTO** : Merupakan perintah untuk lompat menuju subroutine GOSUB. Fungsi GOSUB hampir sama dengan GOTO dan biasanya diakhiri RETURN.

**Hardware** : Perangkat keras pada komputer.

**LOOP** : Merupakan perintah pengulangan dimana pada awal program dibuka dengan perintah DO.

**Mikrokontroler** : Sebuah sistem mikroprosesor lengkap yang terkandung di dalam sebuah chip.

**Mikrokontroler Basic Stamp BS2x** : Mikrokontroler dikembangkan oleh Parallax, Inc yang mudah diprogram menggunakan format bahasa pemrograman Basic.

**Motor Servo** : Jenis aktuator yang cukup banyak digunakan dalam bidang industri atau sistem robotik.

**PAUSE** : Merupakan perintah untuk memberikan delay pada program dalam satuan milidetik (ms).

**Piezoelektrik** : Peralatan piezoelektrik secara langsung mengubah energi listrik menjadi energi mekanik.

**Port I/O** : Kaki untuk jalur keluar atau masuk sinyal sebagai hasil keluaran ataupun masukan bagi program.

**PWM (Pulse Width Modulation)** : Fasilitas untuk membuat modulasi pulsa.

**RAM (Random Acces Memory)** : Memori yang membantu CPU untuk penyimpanan data sementara dan pengolahan data ketika program sedang running.

**RETURN** : Merupakan fungsi yang berdampingan dengan GOSUB dimana setelah eksekusi dari subroutine selesai maka program akan kembali ke tempat setelah GOSUB.

**Receiver** : Transduser gelombang ultrasonic yang menggunakan bahan piezoelektrik berfungsi sebagai penerima gelombang pantulan.

**Register** : Memori (tempat untuk menyimpan instruksi dan data yang diperlukan selama operasi) kecepatan tinggi yang digunakan untuk menyimpan informasi selama operasi CPU (Central Processing Unit).

**Robot Mobil** : Konstruksi robot yang ciri khasnya adalah mempunyai aktuator berupa roda untuk menggerakkan keseluruhan badan robot tersebut, sehingga robot tersebut dapat melakukan perpindahan posisi dari satu titik ke titik yang lain.

**Sensor** : Divais (device) yang menerima signal/stimulus non elektrik (yang me-representasikan sifat-sifat fisis) sebagai masukan dan memberikan tanggapan (response) berupa sinyal elektrik sebagai keluarannya.

**SERIN** : Merupakan perintah komunikasi untuk menerima data secara serial.

**SEROUT** : Merupakan perintah komunikasi untuk mengirimkan data secara serial.

**Sensor cahaya (LDR)** : Sensor berfungsi untuk memberikan nilai masukan pada tingkat intensitas cahaya.

**Software** : Program komputer yang berfungsi sebagai sarana interaksi antara pengguna dan perangkat keras.

**Timer** : Modul dalam hardware yang bekerja untuk menghitung waktu/pulsa.

**Transmitter** : Sebuah alat yang berfungsi sebagai pemancar gelombang ultrasonik dengan frekuensi sebesar 40 kHz yang dibangkitkan dari sebuah osilator.

## Biodata Penulis



Sujarwata lahir di Yogyakarta pada tahun 1961. Pendidikan SD, SMP, SMA ditamatkannya di Yogyakarta. Pada tahun 1982 ia melanjutkan pendidikan di IKIP Yogyakarta Jurusan Pendidikan Fisika dan memperoleh ijazah Sarjana Pendidikan Fisika pada tahun 1988. Ia menjadi dosen di Universitas Negeri Semarang (Unnes) tahun 1989 hingga sekarang. Ia melanjutkan pendidikan S2 di Jurusan Teknik Elektronika UGM, dan memperoleh gelar Magister Teknik (M.T.) pada tahun 2004. Pada tahun 2015 ia menyelesaikan studi pada program S3 Ilmu Fisika di Universitas Gadjah Mada (UGM) dalam bidang Fisika material dan memperoleh gelar Doktor (Dr.). Selama ini, ia bertugas di bidang Elektronika, Instrumentasi dan Mikrokontroler sehingga kegiatan yang telah dilaksanakannya mencakup bidang tersebut.

**UNNES**  
UNIVERSITAS NEGERI SEMARANG