

**PENGATURAN ARAH DAN KECEPATAN PUTAR DISC
INERTIA UNTUK MENJAGA KESEIMBANGAN MINIATUR
SEPEDA MOTOR MENGGUNAKAN KONTROLER PID**

SKRIPSI

JURUSAN TEKNIK ELEKTRO

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh :

ANDIKA WIJAYA KUSUMAH

NIM 0810630031 - 63

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN NASIONAL
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2012**

LEMBAR PERSETUJUAN

**PENGATURAN ARAH DAN KECEPATAN PUTAR *DISC*
INERTIA UNTUK MENJAGA KESEIMBANGAN MINIATUR
SEPEDA MOTOR MENGGUNAKAN KONTROLER PID**

SKRIPSI

JURUSAN TEKNIK ELEKTRO

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh :

ANDIKA WIJAYA KUSUMAH

NIM. 0810630031 - 63

Telah diperiksa dan disetujui oleh

Dosen Pembimbing

Pembimbing I

Pembimbing II

Dr. Ir. Erny Yudaningsas, MT
NIP. 19650913 199002 2 001

Ir. Retnowati, MT
NIP. 19511224 198203 2 001

LEMBAR PENGESAHAN

**PENGATURAN ARAH DAN KECEPATAN PUTAR *DISC*
INERTIA UNTUK MENJAGA KESEIMBANGAN MINIATUR
SEPEDA MOTOR MENGGUNAKAN KONTROLER PID**

Disusun Oleh :

ANDIKA WIJAYA KUSUMAH

NIM : 0810630031 – 63

Skripsi ini telah diuji dan dinyatakan lulus
pada tanggal 6 Agustus 2012

Majelis Penguji :

Goegoes Dwi Nusantoro, ST., MT
NIP. 19711013 200604 1 001

Muhammad Aziz Muslim, ST, MT, Ph.D
NIP. 19741203 200012 1 001

Ir. Purwanto, MT
NIP. 19540424 198601 1 001

Mengetahui :
Ketua Jurusan Teknik Elektro

Dr. Ir. Sholeh Hadi Pramono, MS
NIP. 19580728 198201 1 001

KATA PENGANTAR

Alhamdulillah, segala puji hanya bagi Allah Subhanahu Wa Taala, Rabb alam semesta. Sungguh hanya melalui Pertolongan dan Perlindungan Allah SWT semata sehingga kami dapat menyelesaikan skripsi ini. Dengan seizin Allah SWT, di kesempatan yang baik ini saya ingin menyampaikan rasa terima kasih dan penghargaan yang sebesar besarnya atas bantuan sehingga terselesainya skripsi ini kepada:

1. Ayah dan Ibu tercinta atas doa, semangat, bimbingan serta kasih sayang kepada saya.
2. Bapak Dr.Ir.Sholeh Hadi Pramono, MS., selaku Ketua Jurusan Teknik Elektro Universitas Brawijaya dan Bapak Muhammad Aziz Muslim, ST., MT., Ph.D. selaku Sekretaris Jurusan Teknik Elektro.
3. Ibu Dr.Ir.Erni Yudaningsy, MT. dan Ibu Ir.Retnowati, MT selaku dosen pembimbing skripsi yang telah memberikan masukan, saran, solusi atas segala permasalahan serta kesabaran dalam proses membimbing hingga terselesaikannya skripsi ini.
4. Bapak Ali Mustofa, ST., MT. selaku dosen pembimbing akademik.
5. Keluarga besar Angkatan 2008 "C2K8". Untuk teman-teman seperjuangan Rio, Mahendra, Wahyu, dan Pepi, terima kasih sudah memberi banyak inspirasi dalam pengerjaan skripsi ini.
6. Tim sepak bola teknik elektro khususnya angkatan 2008, prestasi yang kita ukir selama PONSEL tidak akan terlupakan.
7. Semua pihak yang telah memberikan bantuan serta dukungan baik secara langsung maupun tidak langsung atas penyusunan skripsi ini.

Sekiranya Allâh SWT mencatat amalan ikhlas kami dan semua pihak yang turut membantu sehingga skripsi ini terselesaikan. Akhirnya, kami menyadari bahwa skripsi ini masih jauh dari sempurna namun semoga skripsi ini dapat memberikan manfaat bagi kita semua. Amin.

Malang, Juli 2012

Penulis

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
DAFTAR GAMBAR	v
DAFTAR TABEL	vii
ABSTRAK	viii

BAB I PENDAHULUAN

1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	3
1.5 Sistematika Penulisan.....	3

BAB II TINJAUAN PUSTAKA

2.1 Momen Inersia.....	4
2.2 Accelerometer.....	5
2.3 Mikrokontroler ATmega8.....	6
2.3.1 Arsitektur <i>Alf Vegard and RISC</i> (AVR).....	6
2.3.2 Struktur dan Operasi <i>Port</i>	8
2.3.3 <i>Timer/Counter</i>	9
2.4 Motor Direct Current (DC).....	10
2.5 <i>Driver</i> Motor DC.....	12
2.6 <i>Encoder</i>	13
2.7 <i>Pulse Width Modulation</i> (PWM).....	14
2.8 Kontroler <i>Proportional Integral Derivative</i> (PID).....	16
2.8.1 Kontroler Proporsional.....	17
2.8.2 Kontroler Integral.....	18
2.8.3 Kontroler Differensial.....	19
2.8.4 Kontroler PID.....	19

2.8.5	Ziegler-Nichols Kontroler PID.....	20
-------	------------------------------------	----

BAB III METODOLOGI

3.1	Perancangan Sistem	22
3.2	Realisasi Pembuatan Sistem.....	22
3.3	Pengujian dan Analisis Data	23
3.4	Pengambilan Kesimpulan.....	23

BAB IV PERANCANGAN DAN PEMBUATAN ALAT

4.1	Perancangan Sistem	24
4.1.1	Diagram Blok Sistem	24
4.1.2	Prinsip Kerja Alat.....	25
4.2	Perancangan Perangkat Keras (<i>Hardware</i>).....	25
4.2.1	<i>Accelerometer</i>	25
4.2.2	Perancangan Driver Motor	27
4.2.3	Rangkaian <i>Encoder</i>	29
4.2.4	Rangkaian Mikrokontroler ATmega8.....	30
4.2.5	Perancangan Mekanik <i>Miniatur Sepeda Motor</i>	32
4.2.5.1	Spesifikasi Alat.....	33
4.2.5.2	Pemilihan Motor DC.....	33
4.3	Perancangan Perangkat Lunak (<i>Software</i>).....	35

BAB V PENGUJIAN DAN ANALISIS SISTEM

5.1	Pengujian Motor DC.....	36
5.1.1	Tujuan.....	36
5.1.2	Peralatan yang Digunakan.....	36
5.1.3	Langkah Pengujian.....	36
5.2	Pengujian Driver Motor	
5.2.1	Tujuan.....	38
5.2.2	Peralatan yang Digunakan.....	38
5.2.3	Langkah Pengujian.....	39

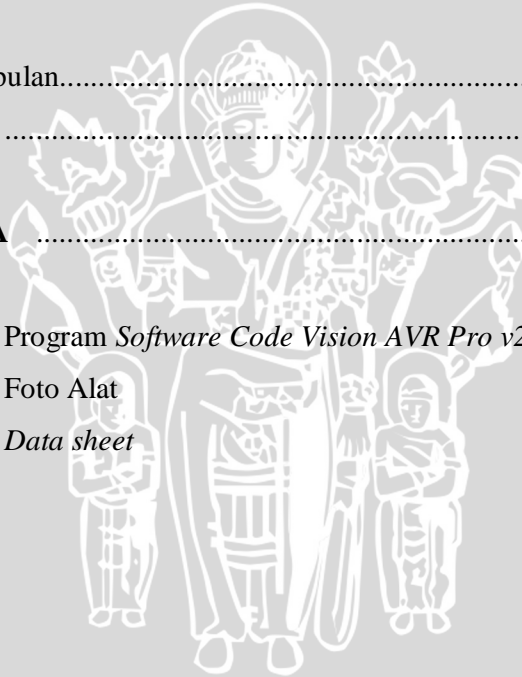
5.3	Pengujian <i>Accelerometer</i>	42
5.3.1	Tujuan Pengujian	42
5.3.2	Peralatan yang Digunakan	42
5.3.3	Langkah Pengujian	42
5.4	Pengujian Keseluruhan	43
5.4.1	Tujuan Pengujian	44
5.3.2	Peralatan yang Digunakan	44
5.3.3	Langkah Pengujian	44
5.4.4	Hasil Pengujian Keseluruhan	46

BAB VI PENUTUP

6.1	Kesimpulan	51
6.2	Saran	51

DAFTAR PUSTAKA	53
-----------------------------	----

LAMPIRAN 1	<i>Program Software Code Vision AVR Pro v2.03.9.</i>
LAMPIRAN 2	Foto Alat
LAMPIRAN 3	<i>Data sheet</i>



DAFTAR GAMBAR

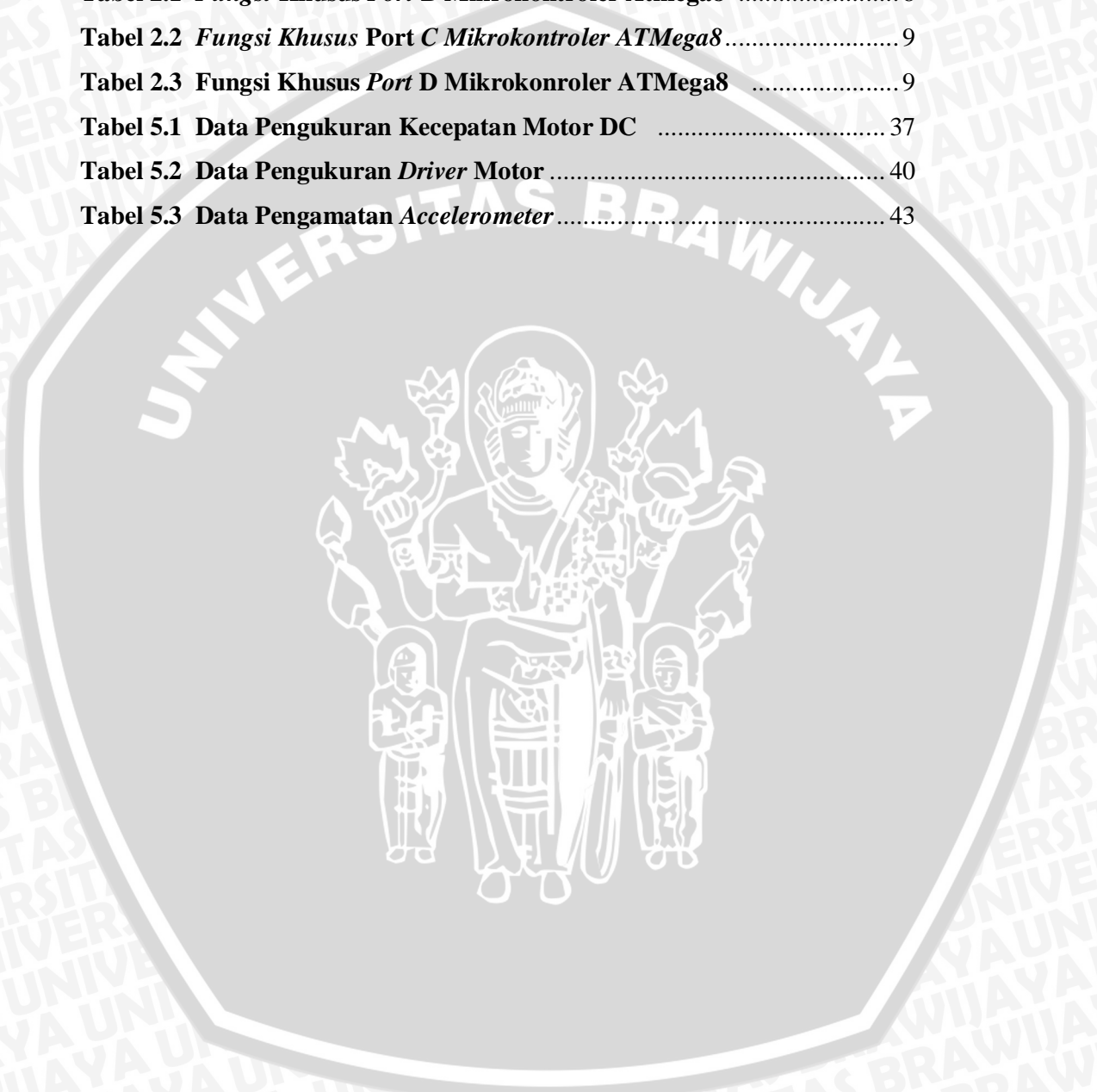
Gambar 2.1 Prinsip Kerja Momen Inersia Pada Beberapa Benda.....	4
Gambar 2.2 Momen Inersia Pada Silinder Pejal	5
Gambar 2.3 Analogi 3 Axis Accelerometer	5
Gambar 2.4 Arsitektur Alf Vegard and RISC (AVR)	7
Gambar 2.5 Konfigurasi Pin ATmega8	7
Gambar 2.6 Penampang Melintang Motor DC	11
Gambar 2.7 Proses Kerja Motor DC	11
Gambar 2.8 Pin Pada Relay DPDT	13
Gambar 2.9 Kombinasi Emmitor dan Detector	13
Gambar 2.10 Prinsip Kerja Dasar Encoder.....	14
Gambar 2.11 Sinyal PWM Secara Umum	14
Gambar 2.12 Contoh Lebar <i>Duty Cycle Pulse Width Modulation</i> (PWM) ..	15
Gambar 2.13 <i>Duty Cycle</i> untuk Perhitungan Tegangan <i>Output Motor</i>	15
Gambar 2.14 Proses Pengontrolan dengan Kontrol PID	16
Gambar 2.15 Diagram Blok Kontroler	17
Gambar 2.16 Diagram Blok Kontroler Integral.....	18
Gambar 2.17 Diagram Blok Kontroler Differensial.....	19
Gambar 2.18 Diagram Blok Kontroler PID.....	20
Gambar 4.1 Diagram Sistem Keseimbangan Miniatur Sepeda Motor.....	24
Gambar 4.2 Prinsip Kerja Accelerometer	26
Gambar 4.3 Accelerometer MMA7455 3-Axis Accel Module	26
Gambar 4.4 Skema Prinsip Dasar Kerja Relay DPDT	27
Gambar 4.5 Rangkaian Driver Motor DC.....	28
Gambar 4.6 Driver Motor DC.....	28
Gambar 4.7 Prinsip Umum Kerja Encoder	29
Gambar 4.8 Rangkaian Elektrik Enkoder.....	29
Gambar 4.9 Mikrokontroler ATmega8	30
Gambar 4.10 Modul Mikrokontroler ATmega8	31

Gambar 4.11 Implementasi Desain Bagian Pertama “Posisi Disc Inertia dan Posisi Sensor Accelerometer”	32
Gambar 4.12 Miniatur Sepeda Motor dan Mekanis Penyeimbang	33
Gambar 4.13 Motor DC gearbox 12 VDC 200rpm	34
Gambar 4.14 Flowchart Program	35
Gambar 5.1 Diagram Blok Pengujian Motor	37
Gambar 5.2 Grafik Pengukuran Kecepatan motor DC	38
Gambar 5.3 Diagram Blok Pengujian Driver Motor	39
Gambar 5.4 Hubungan Duty Cycle PWM dan Tegangan Motor DC	41
Gambar 5.5 Hubungan Duty Cycle PWM dan Kecepatan Motor DC	41
Gambar 5.6 Diagram Blok Pengujian Accelerometer	42
Gambar 5.7 Grafik Pengujian Accelerometer	44
Gambar 5.8 Blok Diagram Pengujian Sistem Keseluruhan	45
Gambar 5.9 Hasil Uji Coba 1	46
Gambar 5.10 Hasil Uji Coba 2	47
Gambar 5.11 Hasil Uji Coba 3	48
Gambar 5.12 Hasil Uji Coba 4	49
Gambar 5.13 Grafik Perubahan Sudut Selama Sistem Bekerja	50



DAFTAR TABEL

Tabel 2.1 <i>Fungsi Khusus Port B Mikrokontroler Atmega8</i>	8
Tabel 2.2 <i>Fungsi Khusus Port C Mikrokontroler ATmega8</i>	9
Tabel 2.3 <i>Fungsi Khusus Port D Mikrokonroler ATmega8</i>	9
Tabel 5.1 <i>Data Pengukuran Kecepatan Motor DC</i>	37
Tabel 5.2 <i>Data Pengukuran Driver Motor</i>	40
Tabel 5.3 <i>Data Pengamatan Accelerometer</i>	43



Abstrak

Andika Wijaya Kusumah. Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya Malang, judul penelitian “Pengaturan Arah dan Kecepatan Putar *Disc Inertia* Untuk menjaga Keseimbangan Miniatur Sepeda Motor Menggunakan Kontroler PID”. Dosen pembimbing: Dr.,Ir. Erni Yudaningtyas, MT. dan Retnowati, Ir., MT.

Perkembangan dunia teknologi saat ini sangatlah pesat. Hampir semua kegiatan manusia dibantu dengan kemajuan teknologi khususnya dalam bidang elektronika. Hal ini menyebabkan perkembangan dunia robotika juga semakin pesat. Salah satu kemampuan robot yang saat ini sedang dikembangkan adalah kemampuan dalam menjaga keseimbangan. Implementasi keseimbangan tersebut dapat diterapkan dalam berbagai bentuk, diantaranya adalah kemampuan menjaga keseimbangan pada miniatur sepeda motor. Dengan memanfaatkan momen inersia pada sebuah disc, miniatur sepeda motor diharapkan dapat tetap berdiri dengan seimbang. *Accelerometer* adalah alat digunakan sebagai pembaca sudut dan sistem kontrol yang digunakan untuk menstabilkan sistem adalah kontroler Proporsional, Integral dan Diferensial (PID) yang ditanamkan pada mikrokontroler ATmega8. Salah satu keuntungan sistem kontrol PID adalah merupakan sebuah sistem yang sederhana sehingga lebih cepat dalam mengambil sebuah keputusan dan mudah dalam analisis. Dengan metode Ziegler-Nichols, maka pencarian parameter K_p , K_i dan K_d dapat lebih tepat dan sesuai dengan kinerja sistem.

Kata kunci : *Momen inersia, accelerometer, kontrol PID*

BAB I

PENDAHULUAN

1.1 Latar Belakang

Teknologi adalah gaya hidup, begitulah kenyataannya dari masa kemasa hingga pada abad ke-21 sekarang ini. Kehidupan manusia modern saat ini tidak lepas dari peran teknologi, hampir setiap aktivitas manusia saat ini menggunakan ilmu teknologi. Menciptakan suatu teknologi baru yang mendekati kemampuan layaknya manusia sudah menjadi ajang lomba dan gengsi ilmu pengetahuan antar universitas maupun kalangan umum.

Seiring dengan perkembangan teknologi dan ilmu pengetahuan terutama kemajuan dalam bidang robotika, mesin maupun robot dirancang agar mempunyai kemampuan yang sama bahkan melebihi kapasitas manusia. Namun ada beberapa kemampuan manusia yang sampai saat ini belum bisa dimiliki oleh robot. Salah satu kemampuan manusia yang tidak dimiliki robot adalah kemampuan menjaga keseimbangan secara mandiri, dimana manusia memiliki sensor dan aktuator penyeimbang yang sangat kompleks.

Sistem keseimbangan otomatis "*autobalance*" pada miniatur sepeda motor adalah sebuah sistem sensorik dan mekanis yang akan menjaga miniatur sepeda motor dalam keadaan seimbang. Model miniatur sepeda motor yang dirancang tidaklah seimbang, agar tetap seimbang seluruh sistem harus bekerja secara sempurna. Sistem pengontrolan dan mekanis penyeimbang harus sesuai dengan data yang diperoleh dari sensor, sehingga tiga elemen (sensor, mekanis, dan kontroler) dituntut bekerja dengan sangat baik.

Keseimbangan miniatur sepeda motor dijaga dengan menggunakan sensor keseimbangan "*accelerometer*", mekanis penyeimbang "*disc inertia*" dan kecerdasan buatan yang ditanam ke dalam mikrokontroler ATMega8. Karena pada sistem keseimbangan ini dibutuhkan respon dengan akurasi tinggi, maka mekanis penyeimbang harus dihitung persamaan matematisnya agar dapat dikalkulasi perkiraan sinyal kontrol yang dikeluarkan oleh kontroler proporsional, integral dan derivatif (PID) yang ditanam di mikrokontroler ATMega8.

1.2 Rumusan Masalah

Rumusan masalah yang dibuat berdasarkan masalah yang terdapat pada latar belakang di atas adalah :

1. Bagaimana membuat model miniatur sepeda motor dengan *disc inertia* sebagai penyeimbang beserta komponen elektriknya?
2. Bagaimana membuat *software* untuk mengatur keseimbangan dengan kontrol PID menggunakan *accelerometer* dengan kecepatan putar serta arah putar *disc inertia* sebagai penyeimbang?

1.3 Batasan Masalah

Dalam perancangan, permasalahan dibatasi oleh hal-hal sebagai berikut :

1. Plant yang digunakan adalah sebuah miniatur sepeda motor dengan spesifikasi sebagai berikut:
 - a. Motor yang digunakan adalah jenis motor DC gearbox 12VDC – 200rpm.
 - b. Sensor yang digunakan adalah *accelerometer* dengan merek MMA7455 3-Axis Accel Module
 - c. *Disc inertia* digunakan sebagai penjaga keseimbangan miniatur sepeda motor berdiameter 8,5cm dan tebal 3mm.
 - d. Rangka sepeda motor mempunyai panjang 160mm dan kedua rodanya berdiameter 55mm.
2. Metode kontrol yang digunakan adalah metode Proporsional, Integral dan Derivatif (PID)
3. Pada skripsi ini digunakan *software Code Vision AVR Pro v2.03.9*.
4. Pusat pengontrolan adalah pada mikrokontroler ATmega8.
5. *Driver* motor yang digunakan adalah PWM mosfet IRFZ44 dan relay *Dual Pole Dual Throw (DPDT)*.
6. Miniatur sepeda motor hanya berdiri diam dan akan diberi gangguan berupa kemiringan sudut.

1.4 Tujuan

Tujuan yang ingin didapat pada penelitian skripsi ini adalah membuat model miniatur sepeda motor yang seimbang dengan adanya pengontrolan *disc inertia* tanpa adanya gangguan maupun diberi gangguan.

1.5 Sistematika Penulisan

BAB I Pendahuluan

Memuat latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi pembahasan, dan sistematika pembahasan.

BAB II Teori Penunjang

Membahas teori-teori yang mendukung dalam perencanaan dan pembuatan alat.

BAB III Metodologi

Berisi tentang metode penelitian dan perencanaan alat serta pengujian.

BAB IV Perencanaan dan Pembuatan Alat

Perancangan dan perealisasiian alat yang belum ada judulnya.

BAB V Pengujian Alat

Memuat hasil pengujian terhadap alat yang telah direalisasikan.

BAB VI Kesimpulan dan Saran

Memuat kesimpulan dan saran-saran.

BAB II

TINJAUAN PUSTAKA

2.1 Momen Inersia

Momen inersia adalah suatu besaran yang memperlihatkan tentang usaha suatu sistem benda untuk menentang gerak rotasinya. Besaran ini dimiliki oleh sistem benda (khususnya padat) apapun bentuknya (bulat, persegi, segitiga, dll). Oleh karena itu momen inersia didefinisikan sebagai kecenderungan suatu sistem benda untuk berputar terus atau diam sebagai reaksi terhadap gaya torsi dari luar.

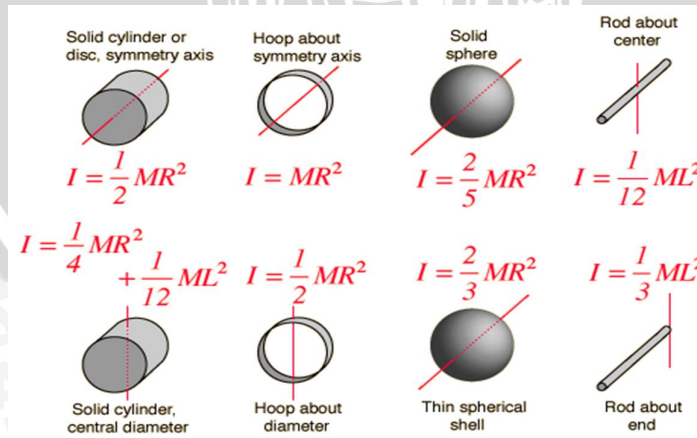
Pada dasarnya menentukan momen inersia benda berwujud tertentu seperti silinder pejal, bola dan sebagainya cenderung lebih mudah dibandingkan jika menentukan besar momen inersia untuk benda yang tidak beraturan dengan distribusi masa yang tidak sama. (Fisika, jilid 2, Halliday & Resnick)

Momen inersia dari beberapa partikel diperoleh dengan menjumlahkan secara aljabar biasa tiap-tiap momen inersia.

$$I = \sum_i m_i r_i^2 \dots\dots\dots (2.1)$$

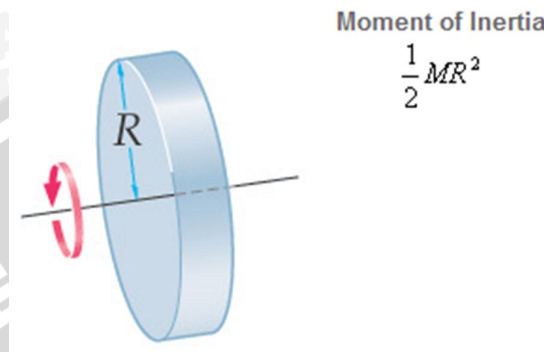
Dimana I adalah momen inersia, m adalah massa benda dan r adalah jari-jari objek dari pusat massa (dalam beberapa kasus yang digunakan adalah panjang objek).

Momen inersia pada beberapa benda dapat dilihat pada Gambar 2.1.



Gambar 2.1 Prinsip Kerja Momen Inersia Pada Beberapa Benda (<http://hyperphysics.phy-astr.gsu.edu/hbase/mi.html>)

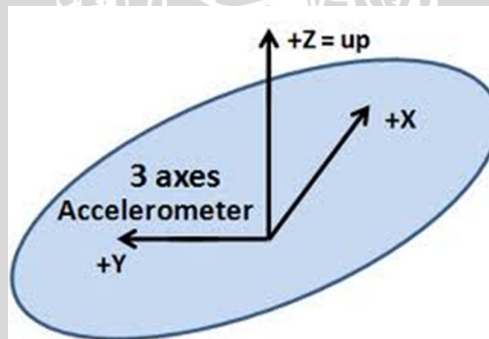
Pada penelitian ini digunakan prinsip kerja momen inersia pada silinder pejal. *Disc inertia* yang digunakan merupakan silinder pejal, seperti terlihat pada Gambar 2.2.



Gambar 2.2 Momen Inersia Pada Silinder Pejal (<http://www.livephysics.com/physical-constants/mechanics-pc/moment-inertia-uniform-objects/>)

2.2 Accelerometer

Accelerometer adalah suatu transduser untuk mengukur percepatan sehingga dapat mendeteksi adanya perubahan posisi *device* dan berapa banyak perubahan itu terjadi. *Accelerometer* merupakan sensor yang peka terhadap gerak linier terhadap 3 sumbu gerakan yakni kanan-kiri, atas-bawah dan depan-belakang. Gambar 2.3 menunjukkan daerah kerja *accelerometer*.



Gambar 2.3 Analogi 3 Axis Accelerometer

(<http://memsblog.wordpress.com/2012/04/04/degrees-of-freedom-vs-axes/>)

Hampir semua *accelerometer* sudah dalam bentuk digital dan bukan dengan sistem mekanik sehingga cara kerjanya hanya berdasarkan temperatur yang diolah secara digital dalam satu chip.

2.3 Mikrokontroler ATmega8

Mikrokontroler ATmega8 merupakan mikrokontroler keluaran *Alf Vegard and RISC* (AVR) yang merupakan mikrokontroler AVR CMOS 8 bit berdaya rendah. Mikrokontroler ATmega8 dapat mengeksekusi instruksi hingga 1 MIPS per MHz dalam satu siklus waktu. Karakteristik utama yang dimiliki oleh mikrokontroler ATmega8 adalah:

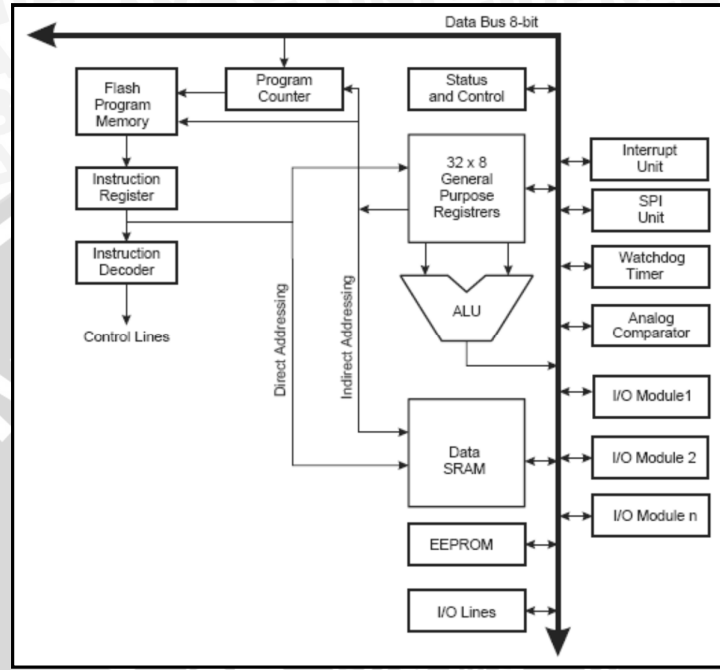
- 1). Memori program dan data yang *nonvolatile*.
- 2). Sistem *self-programable flash* 8 kByte.
- 3). EEPROM sebesar 512 Byte, dan 1kByte SRAM internal.
- 4). 23 saluran I/O dan 32 *general purpose register*.
- 5). Dua *timer/counter* 8 bit dengan *prescaler* terpisah, dan mode pembandingan (*compare mode*).
- 6). Satu buah *timer/counter* 16 bit dengan *prescaler* terpisah, mode pembandingan dan perekam (*capture*).
- 7). Internal dan eksternal *interrupt*.
- 8). Tiga buah pin *Pulse Width Modulation* (PWM)
- 9). Serial USART.

2.3.1 Arsitektur Alf Vegard and RISC (AVR)

Mikrokontroler *Alf Vegard and RISC* (AVR) menggunakan arsitektur Harvard yang memisahkan memori dan bus untuk program dan data sehingga memaksimalkan performa. Instruksi pada memori program dieksekusi secara *pipeline*. Ketika satu instruksi masih dieksekusi, instruksi selanjutnya sudah disiapkan untuk eksekusi tanpa menunggu eksekusi pertama selesai. Konsep ini memungkinkan instruksi untuk dieksekusi setiap satu siklus waktu.

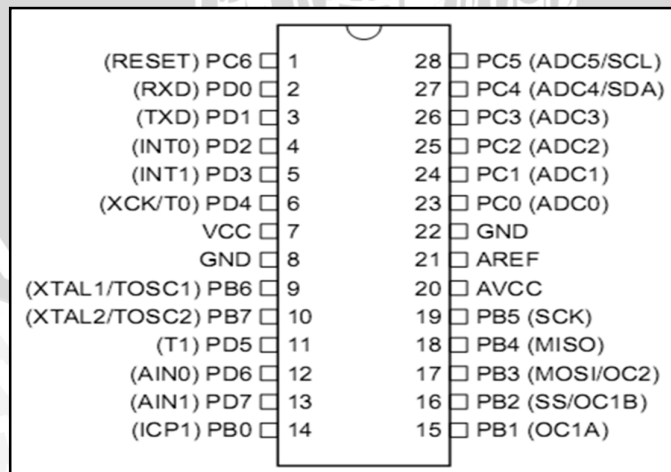
Register file terdiri atas 32 x 8 bit *general purpose working register* dengan waktu akses satu siklus waktu, yang memungkinkan operasi *Arithmetic Logic Unit* (ALU) dijalankan dalam satu siklus waktu. Dua *operand* diambil dari register, operasi ALU dijalankan, dan hasilnya disimpan kembali dalam *register file*. Operasi aritmetika dan logika dapat dijalankan oleh ALU, baik antar register atau antara register dengan konstanta. Hasil dari operasi aritmetika disimpan

dalam register status (*Status Register*), menggantikan isi yang sebelumnya. Arsitektur AVR ditunjukkan pada Gambar 2.4.



Gambar 2.4 Arsitektur Alf Vegard and RISC (AVR) (Atmel, 2004 : 7)

Program *Flash memory* dibagi dalam dua bagian, *Boot Program* dan *Application Program*. Kedua bagian memiliki *Lock bits* untuk mengunci operasi tulis (*write*) dan baca/tulis (*read/write*). Konfigurasi pin mikrokontroler ATmega8 dapat dilihat pada Gambar 2.5.



Gambar 2.5 Konfigurasi Pin ATmega8 (Atmel, 2004)

2.3.2 Struktur dan Operasi Port

Mikrokontroler ATmega8 ini mempunyai 4 buah *port*, yang memiliki 8 buah jalur I/O. Beberapa karakteristik *port* mikrokontroler ATmega8 dijelaskan secara singkat:

- 1). Unit I/O dapat dialamati perjalur atau per *port*.
- 2). Setiap jalur I/O memiliki *buffer*, penahan (*latch*), kemudi *input* dan kemudi *output*.
- 3). Setiap jalur I/O terdapat register pengatur apakah dijadikan *input* atau dijadikan *output*.
- 4). *Port B* adalah I/O *bi-directional* 8 bit dengan resistor *pull-up* internal. Sebagai masukan, pin *port B* yang diberi *pull-low* secara eksternal akan mengalirkan arus bila resistor *pull-up* diaktifkan. *Port B* juga memiliki fungsi khusus, seperti yang terlihat dalam Tabel 2.1.

Tabel 2.1. Fungsi Khusus Port B Mikrokontroler ATmega8 (Atmel, 2004 : 56)

Port Pin	Alternate Functions
PB7	XTAL2 (Chip Clock Oscillator pin 2) TOSC2 (Timer Oscillator pin 2)
PB6	XTAL1 (Chip Clock Oscillator pin 1 or External clock input) TOSC1 (Timer Oscillator pin 1)
PB5	SCK (SPI Bus Master clock Input)
PB4	MISO (SPI Bus Master Input/Slave Output)
PB3	MOSI (SPI Bus Master Output/Slave Input) OC2 (Timer/Counter2 Output Compare Match Output)
PB2	\overline{SS} (SPI Bus Master Slave select) OC1B (Timer/Counter1 Output Compare Match B Output)
PB1	OC1A (Timer/Counter1 Output Compare Match A Output)
PB0	ICP1 (Timer/Counter1 Input Capture Pin)

- 5). *Port C* adalah I/O *bi-directional* 8 bit dengan resistor *pull-up* internal. Sebagai masukan, pin *port C* yang diberi *pull-low* secara eksternal akan mengalirkan arus bila resistor *pull-up* diaktifkan. *Port C* juga memiliki fungsi khusus, seperti yang terlihat pada Tabel 2.2.

Tabel 2.2. Fungsi Khusus Port C Mikrokontroler ATmega8 (Atmel, 2004 : 59)

Port Pin	Alternate Function
PC6	RESET (Reset pin)
PC5	ADC5 (ADC Input Channel 5) SCL (Two-wire Serial Bus Clock Line)
PC4	ADC4 (ADC Input Channel 4) SDA (Two-wire Serial Bus Data Input/Output Line)
PC3	ADC3 (ADC Input Channel 3)
PC2	ADC2 (ADC Input Channel 2)
PC1	ADC1 (ADC Input Channel 1)
PC0	ADC0 (ADC Input Channel 0)

- 6). Port D adalah I/O *bi-directional* 8 bit dengan resistor *pull-up* internal. Sebagai masukan, pin port D yang diberi *pull-low* secara eksternal akan mengalirkan arus bila resistor *pull-up* diaktifkan. Port D juga memiliki fungsi khusus, seperti yang terlihat pada Tabel 2.3.

Tabel 2.3. Fungsi Khusus Port D Mikrokontroler ATmega8 (Sumber: Atmel, 2004 : 61)

Port Pin	Alternate Function
PD7	AIN1 (Analog Comparator Negative Input)
PD6	AIN0 (Analog Comparator Positive Input)
PD5	T1 (Timer/Counter 1 External Counter Input)
PD4	XCK (USART External Clock Input/Output) T0 (Timer/Counter 0 External Counter Input)
PD3	INT1 (External Interrupt 1 Input)
PD2	INT0 (External Interrupt 0 Input)
PD1	TXD (USART Output Pin)
PD0	RXD (USART Input Pin)

2.3.3 Timer/Counter

Mikrokontroler ATmega8 memiliki 3 buah *timer/counter* yang terdiri atas 2 buah *timer/counter* 8 bit dan 1 buah *timer/counter* 16 bit. Ketiga *timer/counter* ini dapat diatur dalam mode yang berbeda. Selain itu semua *timer/counter* dapat difungsikan sebagai sumber interupsi. *Timer/counter* dapat digunakan dalam 4 mode operasi, yaitu :

- 1). Mode pertama (*mode 0*) adalah *mode normal*, *timer* digunakan sebagai pencacah tunggal yang dapat mencacah dari 0x00 sampai dengan 0xFF.

Setelah mencapai nilai 0xFF maka register *counter* akan *reset* atau kembali ke 0x00.

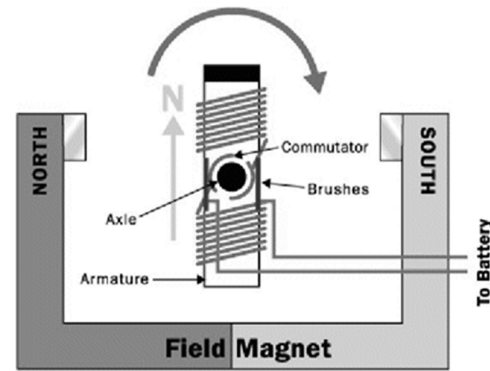
- 2). *Mode* kedua (*mode* 1) adalah *Phase Correct PWM* (PCP). *Mode* ini digunakan untuk menghasilkan sinyal PWM dimana nilai register *counter* yang mencacah naik dan turun secara terus menerus akan selalu dibandingkan dengan register pembanding OCRn. Hasil perbandingan register *counter* dan OCRn digunakan untuk membangkitkan sinyal PWM yang dikeluarkan pada pin OCn.
- 3). *Mode* ketiga (*mode* 2) adalah *clear timer on compare match* (CTC). Register *counter* akan mencacah naik kemudian akan direset atau kembali menjadi 0x00 pada saat nilai TCNT sama dengan OCRn.
- 4). *Mode* keempat (*mode* 3) adalah *fast PWM*. *Mode* ini hampir sama dengan *mode phase correct PWM*, hanya perbedaannya adalah register *counter* mencacah naik saja dan tidak mencacah turun.

2.4 Motor Direct Current (DC)

Segulung kawat yang dialiri arus listrik dan ditempatkan di dalam suatu medan magnet akan mengalami gaya yang sebanding dengan arus dan kekuatan medan magnetnya. Gaya yang ditimbulkan disebut dengan Gaya Lorentz (Soemarwanto, 1997) yang dapat dirumuskan sebagai berikut:

$$F = B \cdot I \cdot L \dots (N) \dots \dots \dots (2.2)$$

Dalam hal ini B adalah kerapatan *fluks* magnet (Wb/m^2), I adalah arus yang mengalir (A) dan L adalah panjang kawat (m). Gambar penampang melintang dari motor DC ditunjukkan pada Gambar 2.6.

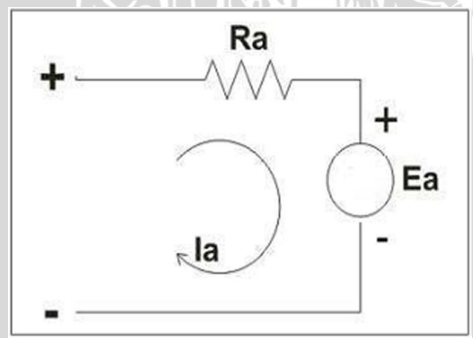


Gambar 2.6 Penampang Melintang Motor DC (www.electronics-scheme.com)

Persamaan 2.2 merupakan prinsip dasar sebuah motor, di mana terjadinya proses perubahan energi listrik (I) menjadi energi mekanik (F). Jika motor mempunyai jari-jari sebesar r , maka akan menimbulkan torsi sebesar :

$$\tau = F \cdot r = B \cdot I \cdot L \cdot r \dots\dots\dots(2.3)$$

Pada saat dibangkitkan, konduktor akan bergerak di dalam medan magnet dan akan menimbulkan gaya gerak listrik (ggl) yang merupakan reaksi lawan terhadap tegangan penyebabnya. Proses konversi energi listrik menjadi energi mekanik dapat berlangsung jika tegangan sumber lebih besar dari gaya gerak listrik lawan (Soemarwanto,1997).



Gambar 2.7 Proses Kerja Motor DC (www.mikron123.com)

Motor dapat berputar jika tegangan masukan motor lebih besar dari ggl yang timbul. Hubungan antara tegangan sumber dan ggl lawan seperti ditunjukkan dalam Gambar 2.7 dapat dirumuskan sebagai berikut:

$$E_a = V_{in} - I_a \cdot R_a \dots(V) \dots\dots\dots(2.4)$$

Dalam hal ini E_a adalah tegangan pada jangkar, V_{in} adalah tegangan masukan, I_a adalah arus jangkar dan R_a adalah tahanan jangkar, sedangkan induksi yang timbul adalah:

$$E_a = C n \Phi \dots(V) \dots\dots\dots(2.5)$$

Dengan C adalah konstanta, n adalah kecepatan motor, dan Φ adalah fluks magnetik yang besarnya sebanding dengan arus penguatan torsi. Torsi pada motor juga sebanding dengan fluks magnetik dan arus. Hal ini ditunjukkan dalam Persamaan 2.6 berikut :

$$\tau = C \Phi I_a \dots(Nm) \dots\dots\dots(2.6)$$

Jika diketahui kecepatan sudut ω adalah :

$$\omega = \frac{2\pi.n}{60} \dots(rad/s) \dots\dots\dots(2.7)$$

Maka hubungan torsi dan kecepatan motor adalah :

$$\tau = P/\omega \dots (Nm) \dots\dots\dots(2.8)$$

$$\tau = \frac{P}{2\pi n/60} \dots(Nm) \dots\dots\dots(2.9)$$

Apabila persamaan 2.4 disubstitusikan ke dalam persamaan 2.5 akan didapatkan rumus kecepatan motor (n) sebagai berikut:

$$n = \frac{V_{in}-I_a R_a}{C \Phi} \dots(rpm) \dots\dots\dots(2.10)$$

Dari Persamaan 2.10 terlihat bahwa kecepatan motor sebanding dengan tegangan masukan (V_{in}) (Zuhal, 1993). Jadi apabila tegangan masukan besar maka kecepatan motor akan cepat, demikian pula sebaliknya, jika tegangan masukan kecil maka kecepatan motor akan lambat.

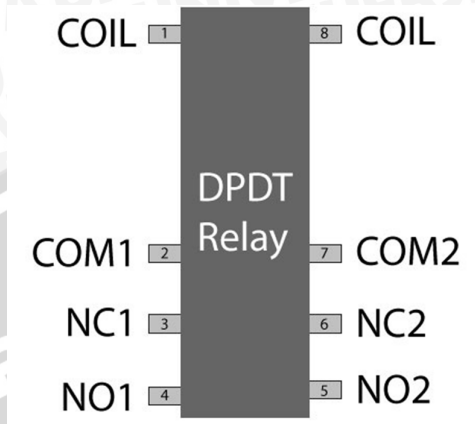
2.5 Driver Motor DC

Driver Motor DC yang digunakan adalah modul relay *dual pole dual throw* (DPDT) sebagai pengatur arah putaran motor DC dan rangkaian mosfet yang berfungsi sebagai penguat tegangan masukan dari mikrokontroler ATMega8.

Relay adalah sebuah saklar elektronis yang dapat dikendalikan dari rangkaian elektronik lainnya. Relay terdiri dari 3 bagian utama, yaitu:

1. koil : lilitan dari relay
2. *common* : bagian yang tersambung dengan NC(dlm keadaan normal)
3. kontak : terdiri dari NC dan NO

Gambar skematik pin dari relay DPDT ditunjukkan pada Gambar 2.8.

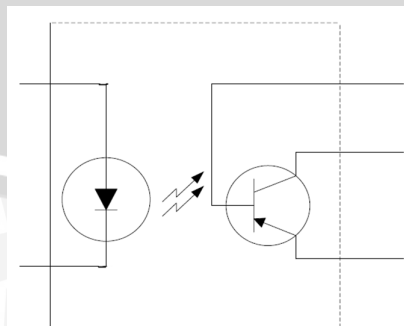


Gambar 2.8 Pin Pada Relay DPDT

Relay DPDT dapat digunakan untuk mengganti atau membalik polaritas dari terminal *device* yang terhubung dengan output. Sebagai contohnya yaitu untuk mengontrol motor DC searah jarum jam dan berlawanan arah jarum jam, dapat menggunakan kombinasi berikut. Pin 2 dan 7 masing-masing dihubungkan ke *Vcc* dan *ground*. Terminal motor awalnya dihubungkan ke pin 3 dan 4 sementara terminal lainnya terhubung ke pin 5 dan 6. Pada saat tidak ada sinyal input yang diberikan maka motor akan berputar dalam satu arah putaran (misalnya, searah jarum jam), dan pada saat diberikan sinyal input, posisi kontak akan berubah dan menyebabkan perubahan arah putaran motor (berlawanan arah jarum jam).

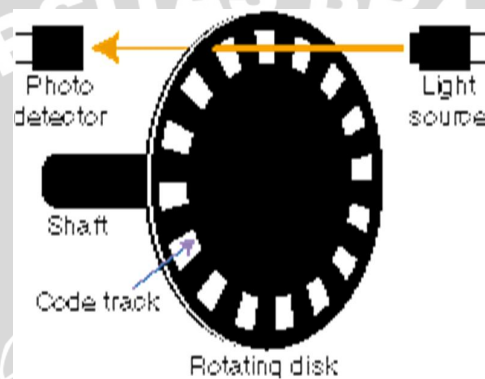
2.6 Encoder

Isolator optik (*Opto – Isolator*) atau sering disebut dengan optocoupler adalah rangkaian terpadu yang terdiri dari fototransistor dan *Light Emitting Diode* (LED) kombinasi antara *emitter* dan *detector*. Susunan dari optocoupler diperlihatkan pada Gambar 2.9.



Gambar 2.9 Kombinasi Emmitor Dan Detector

Penerimaan cahaya infra merah akan membuat transistor peka cahaya menjadi konduksi. Ketika transistor konduksi, keluarannya akan menjadi masukan logika rendah. Jadi keluaran rangkaian transduser akan berlogika rendah ketika transistor konduksi. Pada saat LED tidak konduksi (*off*), LED tidak akan memancarkan cahaya infra merah sehingga transistor juga akan mati. Hal ini akan memberi keluaran dari transistor berlogika tinggi. Dasar prinsip kerja dari *encoder* bisa dilihat pada Gambar 2.10.

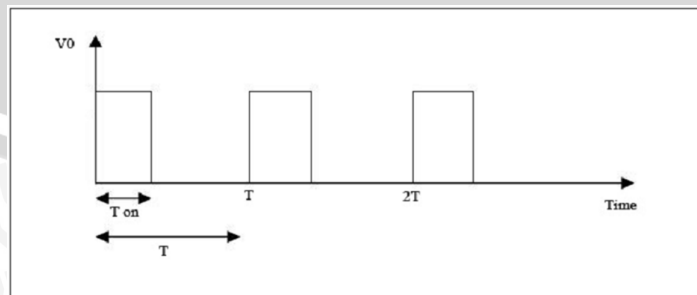


Gambar 2.10 Prinsip Kerja Dasar Encoder (<http://www.ni.com/white-paper/4672/en>)

2.7 Pulse Width Modulation (PWM)

Pulse Width Modulation (PWM) digunakan untuk mengatur kecepatan motor DC. Dimana kecepatan motor DC tergantung pada besarnya *Duty cycle* yang diberikan pada motor DC tersebut.

Pada sinyal PWM, frekuensi sinyal konstan sedangkan *Duty cycle* bervariasi dari 0%-100%. Dengan mengatur *Duty cycle* akan diperoleh keluaran yang diinginkan. Sinyal PWM secara umum dapat dilihat pada Gambar 2.11.



Gambar 2.11 Sinyal PWM Secara Umum (www.electronics-scheme.com)

Dengan cara mengatur lebar pulsa “on” dan “off” dalam satu perioda gelombang melalui pemberian besar sinyal referensi output dari suatu PWM akan didapat duty cycle yang diinginkan. *Duty cycle* dari PWM dapat dinyatakan sebagai:

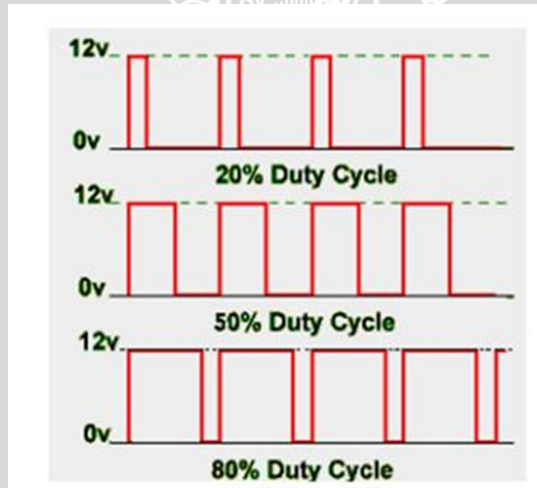
$$Duty cycle = \frac{T_{on}}{T} \times 100\% \dots (\%) \dots \dots \dots (2.11)$$

Dengan :

T on = Periode logika tinggi

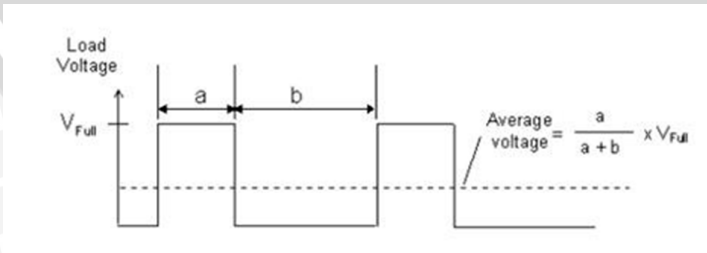
T = Periode keseluruhan

Duty cycle 100% berarti sinyal tegangan pengatur motor dilewatkan seluruhnya. Jika tegangan catu 100V, maka motor akan mendapat tegangan 100V. pada *duty cycle* 50%, tegangan pada motor hanya akan diberikan 50% dari total tegangan yang ada, begitu seterusnya. Contoh lebar *duty cycle* dapat dilihat pada Gambar 2.12.



Gambar 2.12 Contoh Lebar *Duty Cycle Pulse Width Modulation (PWM)* (<http://digilib.itelkom.ac.id>)

Perhitungan Pengontrolan tegangan *output* motor dengan metode PWM cukup sederhana.



Gambar 2.13 *Duty Cycle untuk Perhitungan Tegangan Output Motor* (<http://digilib.itelkom.ac.id>)

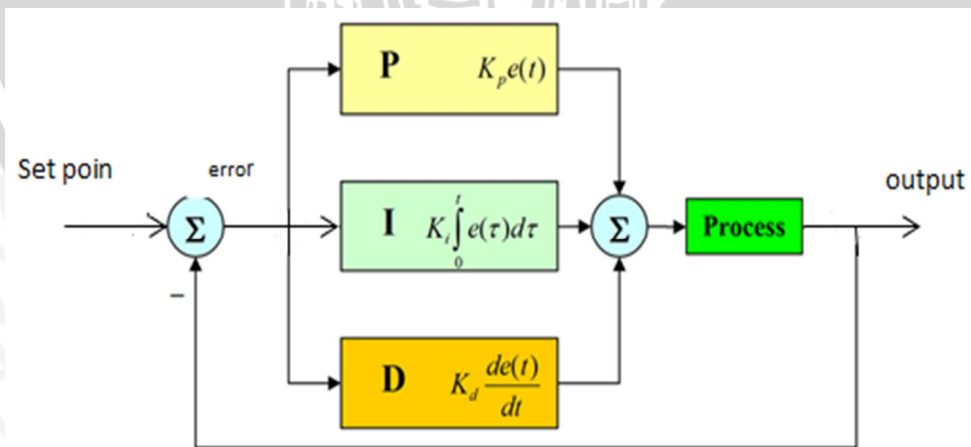
Dengan menghitung *duty cycle* yang diberikan, akan didapat tegangan *output* yang dihasilkan. Sesuai dengan rumus yang telah dijelaskan pada gambar 2.13

$$\text{Average Voltage} = \frac{a}{a + b} \times V_{full} \dots\dots\dots (2.12)$$

Average voltage merupakan tegangan output pada motor yang dikontrol oleh sinyal PWM. *a* adalah nilai *duty cycle* saat kondisi sinyal “on”. *b* adalah nilai *duty cycle* saat kondisi sinyal “off”. *V full* adalah tegangan maximum pada motor. Dengan menggunakan rumus di atas, maka akan didapatkan tegangan output sesuai dengan sinyal kontrol PWM yang dibangkitkan.

2.8 Kontroler Proportional Integral Derivative (PID)

PID Kontroler adalah gabungan antara algoritma kontrol *Proportional*, *Integral*, dan *Derivative*. Sistem pengendalian dirancang untuk melakukan dan menyelesaikan tugas tertentu. Syarat utama sistem pengendalian adalah harus stabil. Di samping kestabilan mutlak, maka sistem harus memiliki kestabilan secara relatif, yakni tolak ukur kualitas kestabilan sistem dengan menganalisis sampai sejauh mana batas-batas kestabilan sistem tersebut jika dikenai gangguan (OgataK,1997). Selain itu analisis juga dilakukan untuk mengetahui bagaimana kecepatan sistem dalam merespons *input*, dan bagaimana peredaman terhadap adanya lonjakan (*over shoot*).Blok diagram kontrol PID terlihat pada Gambar 2.14.



Gambar 2.14 Proses Pengontrolan dengan Kontrol PID

Suatu sistem dikatakan stabil jika diberi gangguan maka sistem tersebut akan kembali ke keadaan *steady state* di mana *output* berada dalam keadaan tetap seperti tidak ada gangguan. Sistem dikatakan tidak stabil jika *output* nya berosilasi terus menerus ketika dikenai suatu gangguan. Karena suatu sistem pengendalian biasanya melibatkan penyimpanan energi maka *output* sistem ketika diberi suatu *input*, tidak dapat mengikuti *input* secara serentak, tapi menunjukkan respons transien berupa suatu osilasi teredam sebelum mencapai *steady state*.

2.8.1 Kontroler Proporsional

Pengontrol proporsional memiliki keluaran yang sebanding atau proporsional dengan besarnya sinyal kesalahan (selisih antara besaran yang diinginkan dengan harga aktualnya). Secara lebih sederhana dapat dikatakan bahwa keluaran pengontrol proporsional merupakan perkalian antara konstanta proporsional dengan masukannya. Perubahan pada sinyal masukan akan segera menyebabkan sistem secara langsung mengeluarkan output sinyal sebesar konstanta pengalinya.

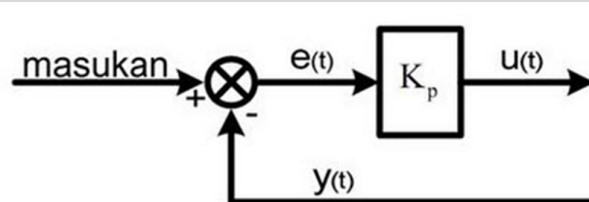
Untuk kontroler dengan aksi kontrol proporsional, hubungan antara keluaran kontroler $m(t)$, K_p adalah kepekaan proporsional atau penguatan dan sinyal kesalahan penggerak $e(t)$ adalah(2.13)

$$m(t) = K_p e(t)$$

atau, dalam besaran transformasi Laplace,

$$\frac{M(s)}{E(s)} = K_p \text{(2.14)}$$

Apapun wujud mekanisme yang sebenarnya dan apapun bentuk daya penggeraknya, kontroler proporsional pada dasarnya merupakan penguat dengan penguatan yang dapat diatur (Ogata.K.,1997). Diagram blok kontroler proporsional ditunjukkan pada Gambar 2.15.



Gambar 2.15 Diagram Blok Kontroler (Ogata, K., 1997)

2.8.2 Kontroler Integral

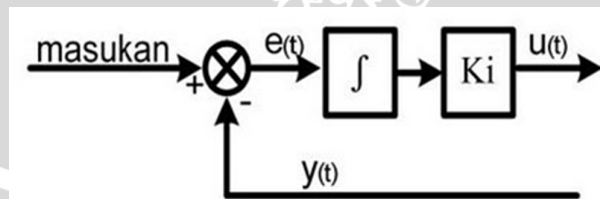
Pengontrol integral berfungsi menghasilkan respon sistem yang memiliki kesalahan keadaan stabil nol. Jika sebuah plant tidak memiliki unsur integrator (1/s), pengontrol proposional tidak akan mampu menjamin keluaran sistem dengan kesalahan keadaan stabilnya nol. Dengan pengontrol integral, respon sistem dapat diperbaiki, yaitu mempunyai kesalahan keadaan stabilnya nol. Pengontrol integral memiliki karakteristik seperti halnya sebuah integral. Keluaran sangat dipengaruhi oleh perubahan yang sebanding dengan nilai sinyal kesalahan.

Keluaran pengontrol ini merupakan penjumlahan yang terus menerus dari perubahan masukannya. Kalau sinyal kesalahan tidak mengalami perubahan, keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan. Sinyal keluaran pengontrol integral merupakan luas bidang yang dibentuk oleh kurva kesalahan penggerak. Sinyal keluaran akan berharga sama dengan harga sebelumnya ketika sinyal kesalahan berharga nol.

Pada kontroler dengan aksi integral, harga keluaran kontroler $m(t)$ diubah dengan laju yang sebanding dengan sinyal kesalahan penggerak $e(t)$. K_i adalah konstanta proporsionalitas.

Jadi,
$$\frac{dm(t)}{dt} = K_i e(t) \dots\dots\dots(2.15)$$

Jika harga $e(t)$ diduakalikan, maka harga $m(t)$ berubah dengan laju perubahan menjadi dua kali semula. Jika kesalahan penggerak nol, maka harga $m(t)$ tetap stasioner. Aksi kontrol integral seringkali disebut kontrol *reset* (Ogata, K., 1997). Gambar 2.16 menunjukkan diagram blok kontroler integral.



Gambar 2.16 Diagram Blok Kontroler Integral (Ogata, K., 1997)

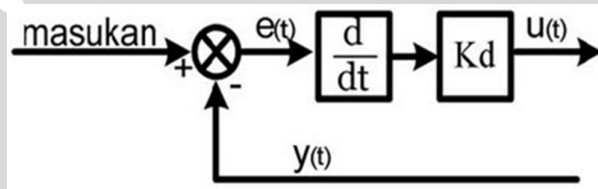
2.8.3 Kontroler Differensial

Kontroler ini digunakan untuk memperbaiki atau mempercepat respons transien sebuah sistem kontrol dengan cara memperbesar *phase lead* terhadap

penguatan kontrol dan mengurangi *phase lag* penguatan tersebut (Ogata.K.,1997). Kontroler differensial tidak dapat mengeluarkan *output* bila tidak ada perubahan *input*, selain itu kontroler differensial tidak dapat digunakan untuk proses yang mengandung *noise*. Hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ adalah :

$$\frac{M(s)}{E(s)} = Kd \cdot s \dots\dots\dots(2.16)$$

Kd adalah konstanta proporsionalitas dan lebih dikenal sebagai turunan terhadap waktu dengan satuan waktu. Gambar 2.17 menunjukkan diagram blok kontroler differensial.



Gambar 2.17 Diagram Blok Kontroler Differensial (Ogata, K., 1997)

2.8.4 Kontroler Proporsional Integral Differensial (PID)

Gabungan aksi kontrol proporsional, integral, dan differensial mempunyai keunggulan dibandingkan dengan masing-masing dari tiga aksi kontrol tersebut.

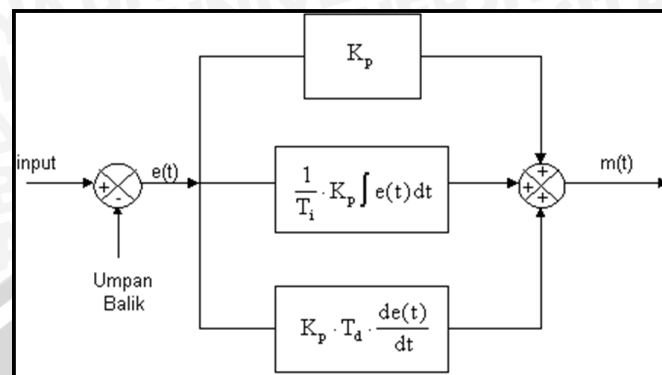
Persamaan kontroler PID ini dapat dinyatakan sebagai berikut :

$$m(t) = Kp \cdot e(t) + \frac{Kp}{Ti} \cdot e(t)dt + Kp \cdot Td \frac{de(t)}{dt} \dots\dots\dots(2.17)$$

Dalam transformasi Laplace dinyatakan sebagai berikut :

$$\frac{M(s)}{E(s)} = Kp \left(1 + \frac{1}{Ti \cdot s} + Td \cdot s \right) \dots\dots\dots(2.18)$$

Gambar 2.18 menunjukkan diagram blok kontroler PID



Gambar 2.18 Diagram Blok Kontroler PID (Ogata, K., 1997.)

Karakteristik pengontrol PID sangat dipengaruhi oleh kontribusi besar dari ketiga parameter P, I dan D. Pengaturan konstanta K_p , T_i , dan T_d akan mengakibatkan penonjolan sifat dari masing-masing elemen. Satu atau dua dari ketiga konstanta tersebut dapat diatur lebih menonjol dibanding yang lain. Konstanta yang menonjol itulah akan memberikan kontribusi pengaruh pada respon sistem secara keseluruhan.

2.8.5 Hand tuning Kontroler PID

Kontroler PID dapat di *tuning* dalam beberapa cara, antara lain Ziegler-Nichols *tuning*, *loop shaping*, metode analitis, optimisasi, *pole placement*, *auto tuning* dan *hand tuning* (Smith, 1979; Astrom & Hagglund, 1995). Pada skripsi ini digunakan cara *hand tuning* untuk menentukan besar K_p , K_i , dan K_d . Hal ini dilakukan karena ada kendala untuk melakukan cara lain yang disebutkan di atas. Kendala tersebut adalah tidak dapat melihat respons motor secara langsung karena tidak digunakannya sensor untuk mengukur kecepatan motor saat sistem berjalan. Selain itu tidak adanya model matematis dari motor membuat analisis sulit untuk dilakukan.

Menurut Smith (1979), untuk melakukan *hand tuning* prosedur yang dilakukan adalah sebagai berikut :

1. Melepaskan kontroler integral dan differensial dengan memberikan nilai $K_i = 0$ dan $K_d = 0$.

2. Mengatur nilai K_p hingga didapatkan respons yang diinginkan, dengan mengabaikan *offset* dari *setpoint*.
3. Dengan terus menaikkan nilai K_p , nilai dari K_d dinaikkan untuk mengurangi *overshoot* yang terjadi.
4. Naikkan nilai K_i untuk mengurangi *offset*.

Keuntungan dari *hand tuning* adalah prosedur diatas dapat dilakukan dengan segera, *online* dan dapat melihat dengan cepat respon sistem setelah perubahan K_p , K_i dan K_d . Kerugian dari cara ini adalah kesulitan untuk melihat apakah *setting* akhir dari kontroler merupakan nilai optimal atau tidak (Jantzen,2001).



BAB III

METODOLOGI

Kajian dalam skripsi ini merupakan penelitian yang bersifat aplikatif, yaitu perencanaan dan pembuatan miniatur sepeda motor dengan *disc inertia* sebagai penyeimbang menggunakan kontrol *Proporsional Integral Differensial* (PID). Bertujuan agar dapat menampilkan performansi sistem yang seimbang sesuai dengan yang direncanakan

Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang akan dibuat adalah sebagai berikut:

3.1 Perancangan Sistem

Perancangan sistem dilakukan sebagai langkah awal sebelum terbentuknya suatu sistem beserta rangkaian elektronik pendukungnya. Hal ini dimaksudkan agar sistem tersebut dapat berjalan sesuai dengan deskripsi awal yang telah direncanakan. Perancangan sistem yang dilakukan meliputi:

- a) Penentuan bentuk dimensi dan bahan yang digunakan dalam miniatur sepeda motor dengan *disc inersia*.
- b) Penentuan rangkaian elektronik yang digunakan.
- c) Penentuan spesifikasi sistem yang akan dibuat, meliputi:
 - Penentuan deskripsi kerja sistem secara keseluruhan.
 - Kontroler PID dan rangkaian elektronik pendukung.
- d) Perancangan program pada mikrokontroler ATmega8

3.2 Realisasi Pembuatan Sistem

Untuk merealisasikan alat atau sistem yang telah dirancang sebelumnya maka perlu diperhatikan beberapa tahapan dalam pembuatan alat tersebut, diantaranya yaitu :

- Pembuatan mekanik miniatur sepeda motor yang meliputi penentuan motor yang digunakan, posisi *disc inertia* serta rangkaian elektroniknya.
- Pembuatan rangkaian elektronik meliputi sensor *accelerometer* dan *driver* motor.

- Pengisian program yang telah dirancang dalam mikrokontroler ATmega8 dengan kendali kontroler PID.

3.3 Pengujian dan Analisis Data

Setelah semua komponen pada alat sudah terhubung sesuai dengan sistem yang telah dirancang dan perangkat lunak untuk mendukung sistem telah dibuat, maka diadakan pengujian dan analisis alat.

3.4 Pengambilan Kesimpulan

Kesimpulan diambil berdasarkan data yang diperoleh dari pengujian sistem secara keseluruhan. Jika hasil yang didapatkan telah sesuai dengan yang direncanakan sebelumnya, maka sistem kendali tersebut telah berhasil memenuhi harapan dan tentunya memerlukan pengembangan lebih lanjut untuk penyempurnaan.



BAB IV

PERANCANGAN DAN PEMBUATAN ALAT

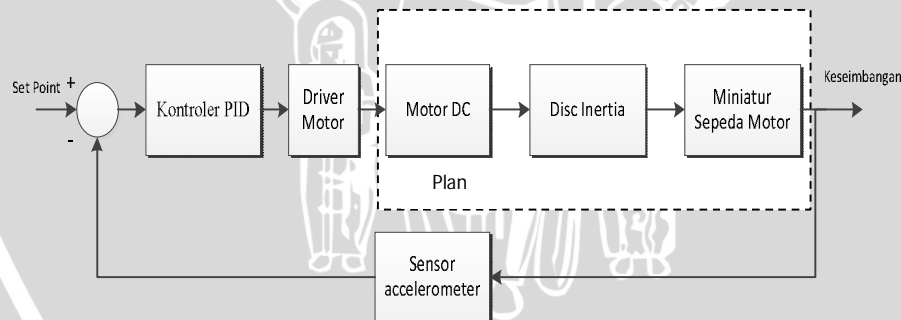
4.1 Perancangan Sistem

Perancangan alat ini dilakukan secara bertahap yaitu blok demi blok, sehingga akan memudahkan dalam menganalisa setiap bloknnya maupun secara keseluruhan. Perancangan dan pembuatan sistem ini terdiri atas dua bagian yaitu perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*). Perancangan ini meliputi :

- Cara kerja alat.
- Spesifikasi alat.
- Perancangan sistem mekanik miniatur sepeda motor.
- Perancangan sistem elektrik miniatur sepeda motor.
- Perancangan algoritma PID pada mikrokontroler ATmega8

4.1.1 Diagram Blok Sistem

Untuk mempermudah pengerjaan alat maka perlu adanya diagram blok umum sistem yang ditunjukkan pada Gambar 4.1.



Gambar 4.1 Diagram Blok Sistem Keseimbangan Miniatur Sepeda Motor

Keterangan dari blok diagram diatas adalah sebagai berikut :

- a. *Accelerometer* berfungsi untuk mengukur kemiringan miniatur sepeda motor terhadap titik tegaknya. *Accelerometer* yang digunakan adalah MMA7455 3-Axis Accel Module.

- b. Pusat pengendalian sistem menggunakan Mikrokontroler ATmega8 yang memberikan keluaran berupa *duty cycle* PWM kepada *driver* motor.
- c. Motor yang digunakan adalah motor DC, yang berfungsi sebagai pemutar *disc inertia*.
- d. *Driver* motor menggunakan *driver PWM Mosfet IRF Z44* dan *Transistor H-bridge 59013 NPN*.

4.1.2 Prinsip Kerja Alat

Miniatur sepeda motor dikatakan seimbang apabila *accelerometer* memberikan tegangan yang menunjukkan dimana miniatur sepeda motor tegak lurus terhadap permukaan bumi. Apabila miniatur sepeda motor tidak tegak lurus terhadap garis normal atau dalam posisi miring maka kontroler akan memberi perintah pada *driver* motor untuk mengendalikan motor untuk memutar *disc inertia* sesuai arah kemiringan miniatur sepeda motor sehingga keadaan miniatur sepeda motor tetap seimbang.

Masukan kontroler berupa kemiringan, sementara keluaran kontroler adalah berupa *duty cycle* PWM yang kemudian diteruskan ke- *driver* motor dan menghasilkan tegangan untuk memutar motor DC dengan arah dan kecepatan tertentu.

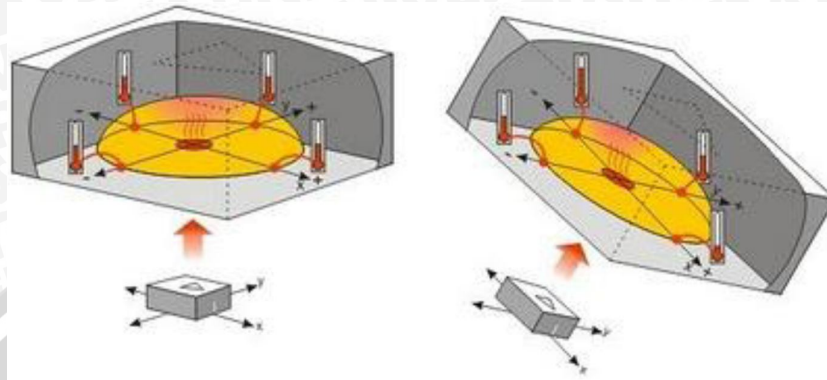
4.2 Perancangan Perangkat Keras (Hardware)

Berdasarkan diagram blok perancangan alat yang telah disusun, perancangan perangkat keras meliputi perancangan mekanik, sensor *accelerometer* dan *driver* motor serta penggunaan mikrokontroler ATmega8.

4.2.1 Accelerometer

Accelerometer digunakan sebagai pendeteksi kemiringan miniatur sepeda motor terhadap permukaan bumi. Setiap perubahan sudut pada miniatur sepeda motor akan terdeteksi oleh *accelerometer* yang kemudian akan mengeluarkan tegangan sesuai dengan besaran kemiringan sudut miniatur sepeda motor terhadap permukaan bumi.

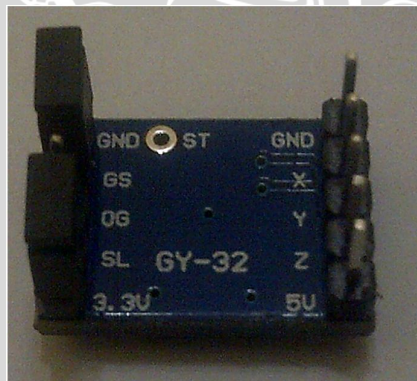
Prinsip kerja sensor *Accelerometer* seperti yang ditunjukkan pada Gambar 4.2.



Gambar 4.2 Prinsip Kerja *Accelerometer*

Sensor menghasilkan panas yang mana jika dimiringkan maka panas tersebut akan memanasi bagian pinggir sensor yang merupakan bahan *thermocoupler*. Sehingga jika dipanaskan akan menghasilkan perubahan dan perbedaan panas yang mengakibatkan terjadinya perubahan tegangan yang nantinya akan dimasukkan kedalam ADC pada mikrokontroler.

Gambar 4.3 adalah *accelerometer* MMA7455 3-Axis Accel Module yang digunakan pada miniatur sepeda motor.



Gambar 4.3 *Accelerometer* MMA7455 3-Axis Accel Module

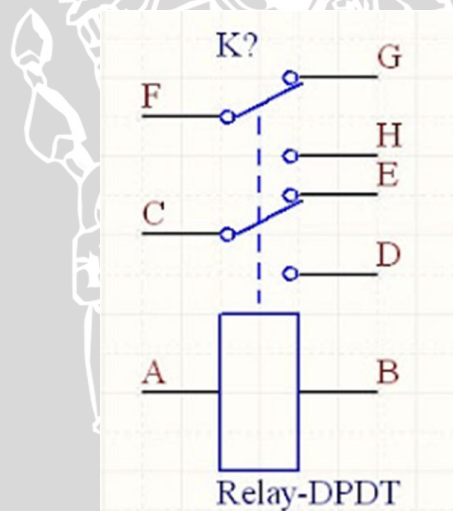
Range kerja *accelerometer* disini berkisar antara 0-5 volt. Keluaran dari *accelerometer* dihubungkan dengan *Port* PC0 (ADC0) pada mikrokontroler ATmega8, keluaran *accelerometer* yang berupa sinyal analog kemudian *convert* menjadi sinyal digital.

4.2.2 Perancangan Driver Motor

Untuk *driver* motor DC digunakan PWM Mosfet IRFZ44 dan *relay dual pole dual throw (DPDT)*. Mosfet IRFZ44 digunakan sebagai penguat PWM yang masuk ke *driver* untuk diteruskan ke motor DC dan *relay dual pole dual throw* digunakan untuk mengatur arah putaran motor.

Relay terdiri dari *Coil & Contact*. *Coil* adalah gulungan kawat yang mendapat arus listrik, sedang *contact* adalah sejenis saklar yang pergerakannya tergantung dari ada tidaknya arus listrik di *coil*.

Contact ada 2 jenis : *Normally Open* (kondisi awal sebelum diaktifkan *open*), dan *Normally Closed* (kondisi awal sebelum diaktifkan *close*). Secara sederhana berikut ini prinsip kerja dari relay, ketika *Coil* mendapat energi listrik (*energized*), akan timbul gaya elektromagnet yang akan menarik *armature* yang berpegas, dan *contact* akan menutup. Skema *relay DPDT* dapat dilihat pada Gambar 4.4.

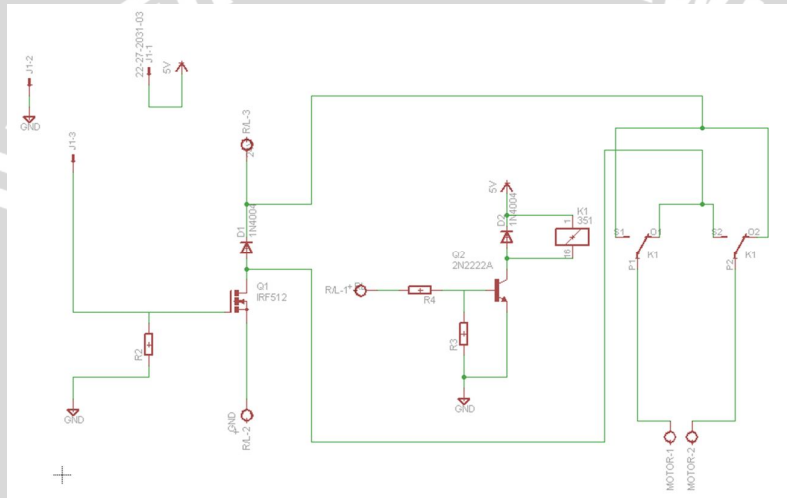


Gambar 4.4 Skema Prinsip Dasar Kerja Relay DPDT

Blok AB terdapat kumparan sebagai driver. Prinsip kerja dari relay ini yaitu, ketika AB belum dilewati arus, maka terminal CE dan FG akan tersambung, dan pada saat AB dilewati arus maka plat C dan F akan berpindah sehingga terminal CD dan FH akan tersambung. Jadi ketika driver AB dilewati arus plat yang berpindah ada 2 yaitu C dan F.

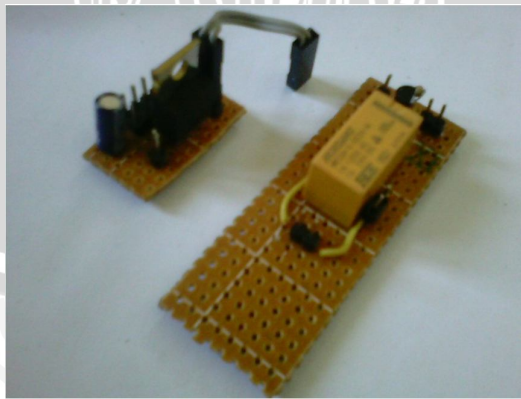
Alasan pemilihan IRFZ44N sebagai komponen driver yang akan digunakan adalah karena IRFZ44N memiliki $I_{D\max}$ cukup besar dan $R_{DS\ on\ max}$ lebih kecil dibanding beberapa jenis E-MOSFET kanal N lain, contohnya IRF520, IRF540, IRF630, IRF720, IRF820 dan IRFZ40. Dengan $I_{D\max}$ yang cukup besar dan $R_{DS\ on\ max}$ yang kecil membuat IRFZ44N tidak cepat panas ketika dilewati arus yang cukup besar.

Rangkaian elektrik driver motor relay dan mosfet yang digunakan untuk mengatur motor DC dapat dilihat pada Gambar 4.5



Gambar 4.5 Rangkaian Driver Motor DC.

Berikut adalah gambar implementasi *driver* motor DC pada miniatur sepeda motor.

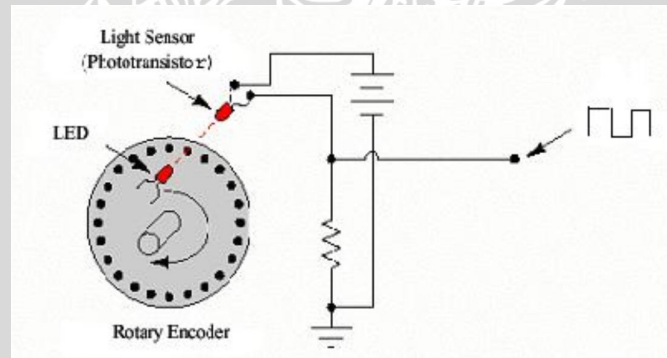


Gambar 4.6 Driver Motor DC

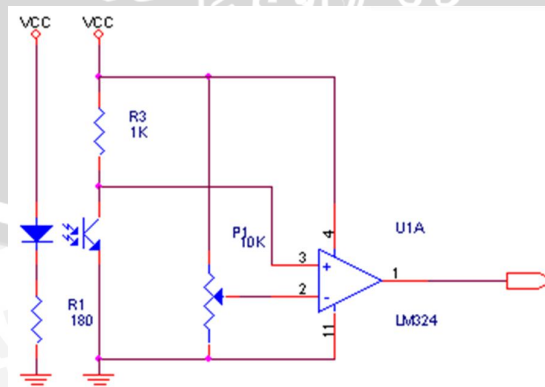
4.2.3 Perancangan Encoder

Fungsi encoder disini adalah sebagai sensor pembaca kecepatan putar motor DC. Bagian sensor dibentuk oleh sebuah optocoupler dan sebuah Op AMP yang bekerja sebagai Voltage Comparator. Optocoupler memancarkan cahaya infrared dari bagian dioda LED yang berfungsi sebagai pemancar dan diterima oleh bagian phototransistor yang berfungsi sebagai penerima. Saat plat yang menempel pada bagian roda memotong pancaran sinar infrared ke bagian penerima, maka phototransistor dari optocoupler akan berkondisi cut off, sehingga tegangan keluaran optocoupler akan berada di atas tegangan pembanding Op Amp. Dengan adanya tegangan di bagian input positif lebih tinggi dari bagian input negatif, maka keluaran Op Amp akan berlogika 1.

Prinsip kerja umum *encoder* dapat dilihat pada Gambar 4.7 dan pada Gambar 4.8 adalah rangkaian *encoder* yang ditanamkan pada miniatur sepeda motor.



Gambar 4.7 Prinsip Umum Kerja Encoder.

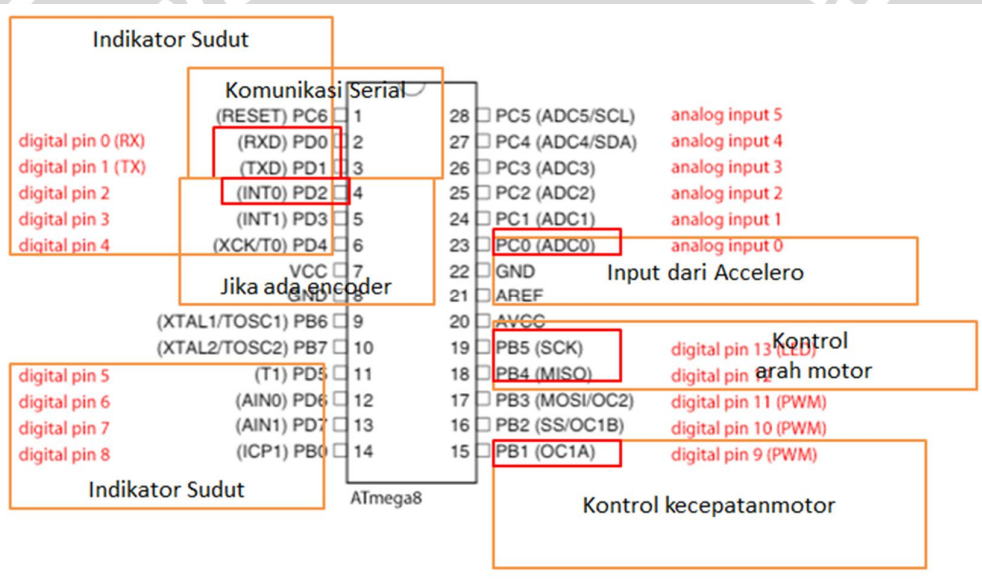


Gambar 4.8 Rangkaian Elektrik Enkoder.

Pada penelitian ini encoder yang digunakan mempunyai 16 kisi atau ada sebanyak 16 celah yang terpasang pada disc inertia sehingga dapat dikatakan ada sebanyak 16 pulsa encoder dalam satu kali rotasi *disc inertia*. Pembacaan dalam skala pulse/second. Jadi apabila terbaca 160 pulse/second = $160/16 = 10$ rotasi disk/second atau sama saja dengan 600 rotasi disk / menit.

4.2.4 Rangkaian Mikrokontroler ATmega8

Pada miniatur sepeda motor ini digunakan Mikrokontroler ATmega8 sebagai pusat dari pengolahan data. Skematik Mikrokontroler ATmega8 dan perancangan secara keseluruhan ditunjukkan pada Gambar 4.9.



Gambar 4.9 Mikrokontroler ATmega8

Agar sebuah Mikrokontroler ATmega8 dapat bekerja sebagai pengontrol, kaki-kaki/*pin* Mikrokontroler ATmega8 dihubungkan dalam rangkaian-rangkaian eksternal. Dalam perancangan ini, *port* yang digunakan adalah sebagai berikut :

1. *Port* PB5 (SCK) dan PB (MISO)

Port ini terhubung dengan motor DC pemutar *disc inertia* sebagai pengatur arah putar motor.

2. *Port* PB1 (OC1A)

Port ini difungsikan sebagai keluaran *pulse width modulation* (PWM) berfungsi sebagai kontrol kecepatan motor yang memutar *disc inertia* agar miniatur sepeda motor tetap seimbang.

3. Port PC0 (ADC0)

Port ini sebagai masukan dari *accelerometer*. Port ini dapat digunakan untuk mengubah *input* yang berupa tegangan analog menjadi data digital.

4. Port PD0 (RXD) dan PD1 (TXD)

Merupakan jalur data komunikasi serial dengan level sinyal TTL. Pin TXD berfungsi untuk mengirimkan data serial, sedangkan RXD kebalikannya yaitu sebagai pin yang berfungsi untuk menerima data serial.

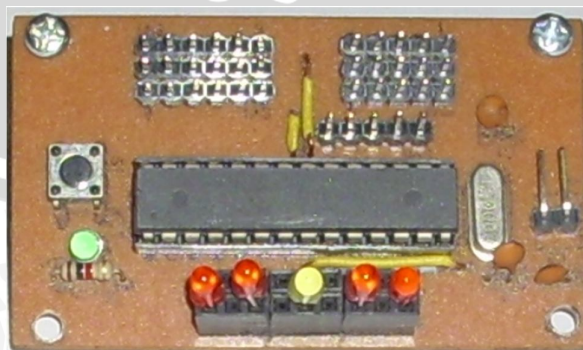
5. Port PD5 (T1), PD6 (AIN0), PD7 (AIN1), PB0 (ICP1)

Berfungsi sebagai indikator sudut ditunjukkan oleh lampu *Light Emmiting Diode* (LED) yang di pasang pada perangkat elektronik miniatur sepeda motor

6. Port VCC dan GND

Port VCC merupakan port suplai tegangan digital. Besarnya tegangan berkisar antara 4,5 – 5,5V. Dan GND adalah Referensi nol suplai tegangan digital.

Gambar 4.10 merupakan gambar modul mikrokontroler ATmega8 yang digunakan pada miniatur sepeda motor.



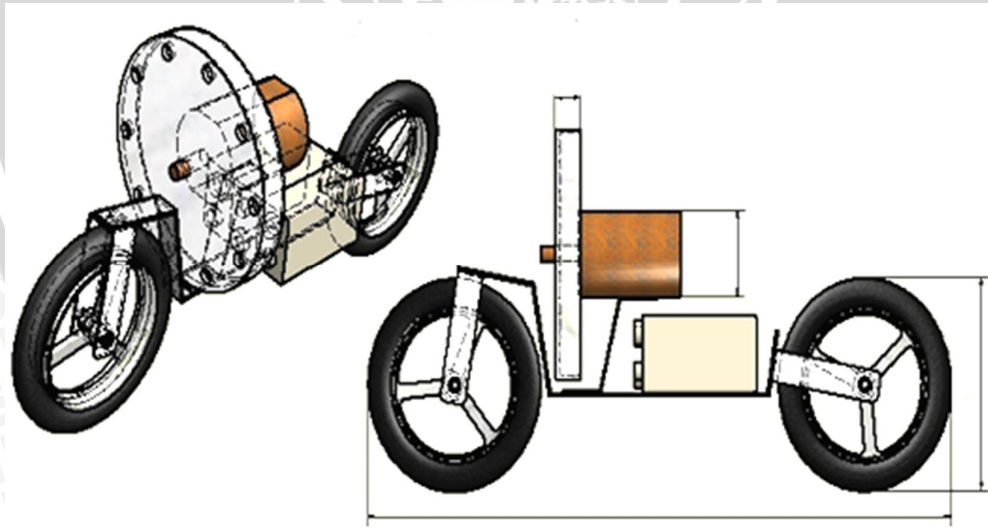
Gambar 4.10 Modul Mikrokontroler ATmega8

Selanjutnya, modul Mikrokontroler ATmega8 dihubungkan dengan komputer dengan menggunakan komunikasi *paralel*. Komunikasi ini bertujuan agar dapat dilakukan pemrograman di komputer dan selanjutnya program tersebut dapat ditanamkan pada Mikrokontroler ATmega8.

4.2.5 Perancangan Mekanik Miniatur Sepeda Motor

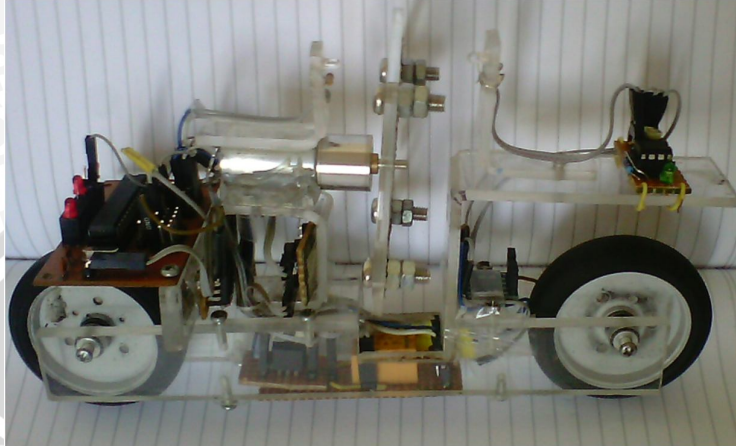
Miniatur sepeda motor mempunyai beberapa bagian yang penting dalam perancangannya, yaitu: bentuk dan posisi *disc inertia*, posisi motor DC dan *accelerometer* serta kerangka miniatur sepeda motor. Motor DC dipasang ditengah dan sejajar dengan miniatur sepeda motor agar posisi *disc inertia* berpotongan dengan posisi motor. Sedangkan posisi dari *accelerometer* yaitu dipasangkan sejajar dengan posisi pada saat motor berada dalam keadaan seimbang, sehingga setiap perubahan posisi miniatur sepeda motor akan terbaca oleh *accelerometer*.

Rencana desain awal mekanik yang akan dibuat pada penelitian ini ditunjukkan pada Gambar 4.11.



Gambar 4.11 Implementasi Desain Bagian Pertama “Posisi *Disc Inertia* dan Posisi Sensor *Accelerometer*”

Gambar 4.12 berikut adalah miniatur sepeda motor beserta komponen elektrik, motor DC dan mekanis penyeimbang *disc inertia*.



Gambar 4.12 Miniatur Sepeda Motor dan Mekanis Penyeimbang

4.2.5.1 Spesifikasi Alat

Spesifikasi alat yang dibuat adalah sebagai berikut:

- Sistem penyeimbang menggunakan *disc inertia* dengan diameter 8.5mm & tebal 3mm.
- Menggunakan catu daya 5V untuk catu *accelerometer* dan mikrokontroler ATmega8
- Menggunakan catu daya 12V untuk motor DC.
- Sensor kemiringan menggunakan *accelerometer*.
- Menggunakan mikrokontroler ATmega8 sebagai pengendali utama
- Driver motor berupa PWM Mosfet IRFZ44 dan *relay DPDT*
- Diameter kedua roda miniatur sepeda motor 55mm.
- Panjang kerangka sepeda motor 160mm

4.2.5.2 Pemilihan Motor DC

Motor yang dipakai pada alat harus memenuhi spesifikasi dari perancangan meliputi torsi dan kecepatannya. Torsi motor berhubungan dengan beban yang ditanggung oleh motor meliputi: berat *disc inertia* dan kekuatan motor untuk berputar ke arah yang berlawanan. Sedangkan kecepatan motor berhubungan dengan kecepatan jatuh dari posisi tegak lurus ke posisi miring atau tidak tegak.

Motor yang digunakan pada penelitian ini adalah motor DC 12 VDC 200 rpm. Seperti yang terlihat pada gambar 4.13.



Gambar 4.13 Motor DC gearbox 12 VDC 200rpm

Keuntungan utama motor DC adalah sebagai pengendali kecepatan, yang tidak mempengaruhi kualitas pasokan daya. Motor ini dapat dikendalikan dengan mengatur:

- Tegangan motor: Meningkatkan tegangan motor akan meningkatkan kecepatan
- Arus medan: Menurunkan arus medan akan meningkatkan kecepatan.

Hubungan antara kecepatan, flux medan dan tegangan dinamo ditunjukkan dalam Persamaan 4.1 dan 4.2.

$$\text{Gaya elektromagnetik: } E = K \cdot \Phi \cdot N \dots\dots\dots (4.1)$$

$$\text{Torque: } T = K \cdot \Phi \cdot I_a \dots\dots\dots (4.2)$$

Dimana:

E = gaya elektromagnetik yang dikembangkan pada terminal dinamo (volt)

Φ = flux medan yang berbanding lurus dengan arus medan

N = kecepatan dalam RPM (putaran per menit)

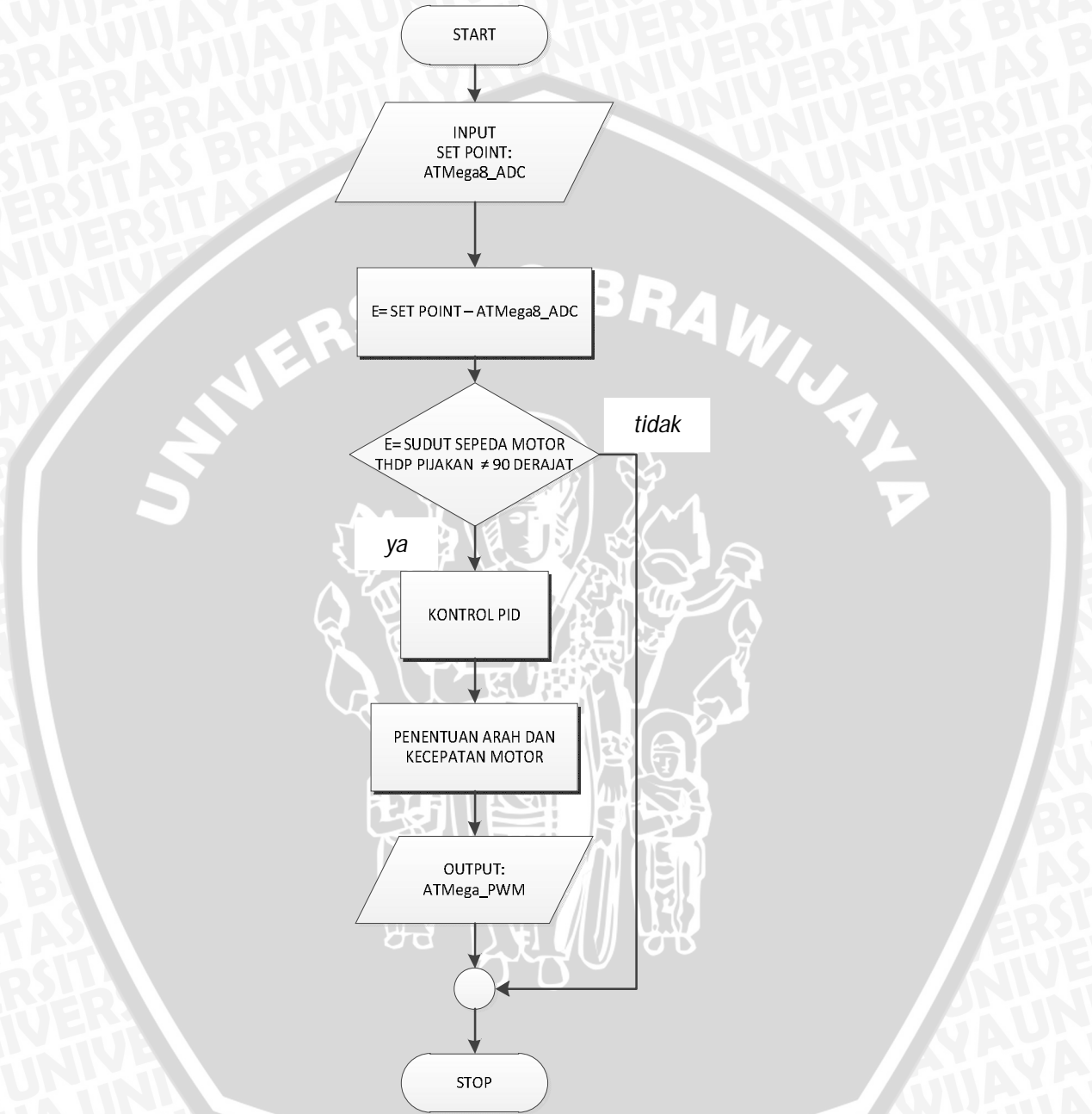
T = torque elektromagnetik

I_a = arus motor

K = konstanta persamaan

Secara umum, kecepatan rotasi dari motor DC adalah sebanding dengan tegangan yang diterapkan untuk itu, dan torsi adalah sebanding dengan arus. Kontrol kecepatan dapat dicapai dengan baterai variabel, tegangan suplai variabel, resistor atau kontrol elektronik. Arah dari putaran motor DC dapat berubah dengan membalik kutub tegangan.

4.3 Perancangan Perangkat Lunak (*software*)



Gambar 4.14 Flowchart Program

Perangkat lunak pada skripsi ini dikerjakan pada program Code Vision AVR Pro v2.03.9. *Tuning* kontroler PID yang digunakan adalah metode *hand tuning* atau dikenal dengan *trial and error*. *Flowchart* perancangan perangkat lunak ditunjukkan pada Gambar 4.14.

BAB V

PENGUJIAN DAN ANALISIS SISTEM

Tujuan pengujian sistem ini adalah untuk menentukan apakah alat yang telah dibuat berfungsi dengan baik dan sesuai dengan perancangan. Pengujian pada sistem ini meliputi pengujian setiap blok maupun pengujian secara keseluruhan. Pengujian setiap blok ini dilakukan untuk menemukan letak kesalahan dan mempermudah analisis pada sistem apabila alat tidak bekerja sesuai dengan perancangan. Pengujian pada sistem ini dibagi menjadi lima bagian, yaitu:

- Pengujian motor DC
- Pengujian *driver* Motor DC
- Pengujian *Accelerometer*
- Pengujian keseluruhan sistem

5.1 Pengujian Motor DC

5.1.1 Tujuan

Mengetahui *output* dari motor apabila diberi *input* yang berbeda-beda.

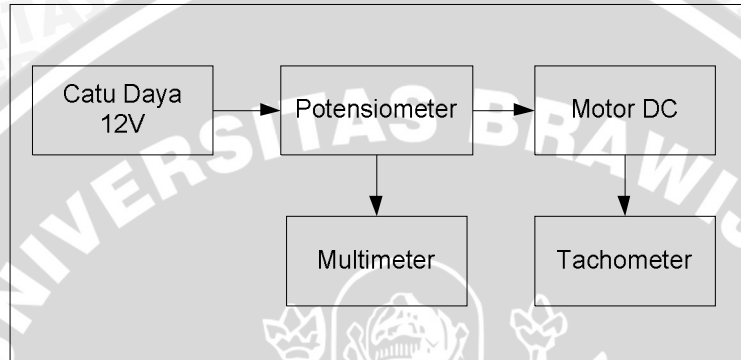
5.1.2 Peralatan yang digunakan

- Mikrokontroler ATmega 8
- Motor DC *gearbox*
- Komputer
- *Software CodeVisionAVR Pro v2.03.9.*
- Catu daya 12 Volt
- Potensiometer
- Serial port max 232

5.1.3 Langkah pengujian

1. Merangkai alat seperti pada Gambar 5.1.
2. Membuat program untuk menentukan arah dan kecepatan motor pada *software CodeVisionAVR Pro v2.03.9.*
3. *Load Flash* program yang telah dibuat, hapus program yang ada dalam mikrokontroler ATmega8, tulis program yang telah dibuat

- ke dalam mikrokontroler ATmega8 menggunakan *software Code Vision Pro v.2.03.9* melalui kabel serial port ke PC.
4. Mengukur kecepatan pada motor yang terbaca oleh encoder melalui serial port yang terhubung ke komputer.
 5. Menggambar data yang didapatkan pada grafik.



Gambar 5.1 Diagram Blok Pengujian Motor

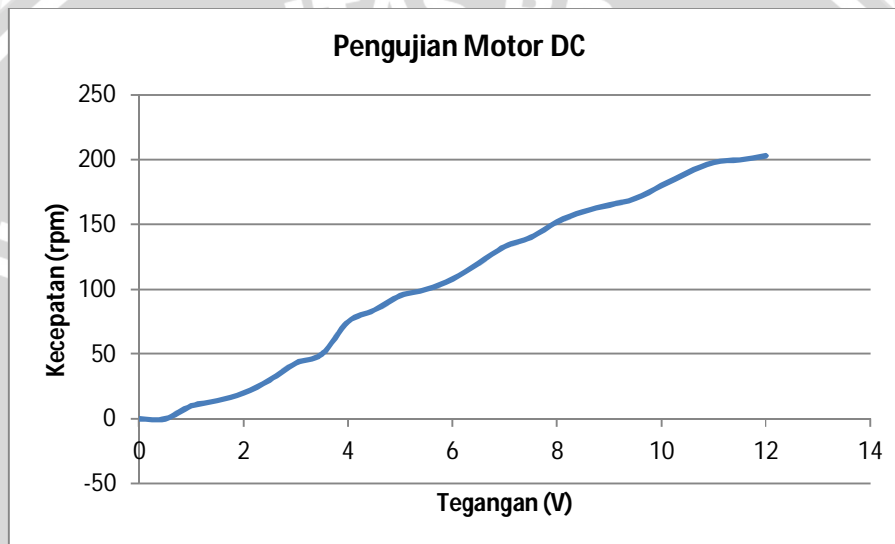
Gambar 5.1 menunjukkan diagram blok pengujian motor DC. Catu daya yang digunakan adalah catu daya sebesar 12 volt. Catu daya ini dipilih berdasarkan spesifikasi motor yang dapat bekerja dengan tegangan maksimal sebesar 12 volt. Tegangan dari catu daya diatur menggunakan sebuah potensiometer mulai dari 0 volt hingga 12 volt dengan interval sebesar 1 volt. Motor akan berputar dengan kecepatan tertentu dan diukur dengan menggunakan sebuah tachometer. Grafik karakteristik motor ditunjukkan pada gambar 5.8 dan tabel 5.2 menunjukkan data dari pengujian motor. Tabel 5.1 menunjukkan data pengukuran motor DC.

Tabel 5.1 Data Pengukuran Kecepatan Motor DC

Tegangan Motor DC (V)	Kecepatan Motor DC (rpm)	Tegangan Motor DC(V)	Kecepatan Motor DC (rpm)
0	0	6.5	120
0.5	0	7	133
1	10	7.5	140
1.5	14	8	149
2	20	8.5	155
2.5	30	9	165
3	43	9.5	169
3.5	50	10	175

4	75	10.5	185
4.5	84	11	198
5	95	11.5	200
5.5	100	12	203
6	108		

Gambar 5.2 menunjukkan grafik kecepatan putar motor DC sesuai dengan pengukuran.



Gambar 5.2 Grafik Pengukuran Kecepatan Motor DC

Dari hasil pengujian dapat dilihat bahwa motor dapat berputar pada saat tegangan minimum 1V volt, motor DC gearbox ini dapat berputar pada range tegangan yang relatif kecil. Dan dari grafik terlihat hubungan antara tegangan dan kecepatan motor DC hampir linier. Sehingga dapat diambil kesimpulan bahwa motor DC gearbox ini sudah baik untuk digunakan dalam penelitian ini.

5.2 Pengujian Driver Motor

5.2.1 Tujuan

Untuk mengetahui output dari *driver* motor apabila diberikan input PWM yang berbeda.

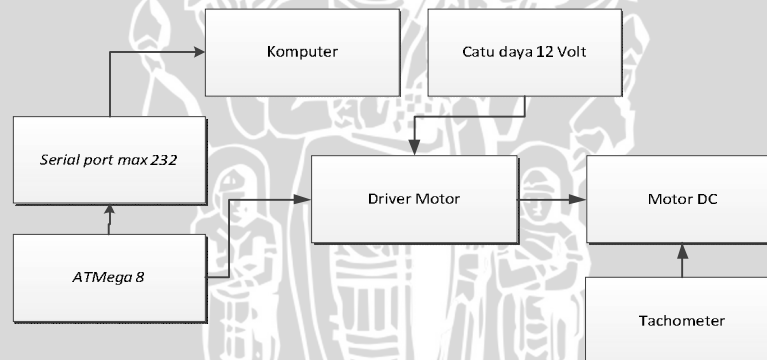
5.2.2 Peralatan yang Digunakan

- Mikrokontroler ATmega 8
- *Driver* motor relay dan Mosfet.

- Komputer
- Serial port Max 232
- Motor DC
- Program *CodeVisionAVR Pro v2.03.9*
- Multimeter
- Catu daya 12 Volt
- Tachometer

5.2.3 Langkah Pengujian

1. Merangkai alat seperti pada Gambar 5.3.
2. Membuat program untuk menentukan arah dan kecepatan motor pada *software CodeVisionAVR Pro v2.03.9*
3. Mengukur tegangan pada kutub motor DC, untuk mengetahui tegangan keluaran dari *driver* motor yang masuk ke motor DC.
4. Menggambar data yang didapatkan pada grafik.



Gambar 5.3 Diagram Blok Pengujian Driver Motor

Gambar 5.3 menunjukkan diagram blok pengujian *driver* motor untuk melihat *output* serta *input* dari *driver* motor apakah sudah sesuai dengan yang diharapkan. *Driver* motor merupakan relay dan mosfet, yang dicatu oleh tegangan 12 V untuk memutar motor DC. Kemudian motor DC yang diputar diukur kecepatannya menggunakan *tachometer*. Sedangkan untuk mengatur besar kecilnya tegangan PWM digunakan serial port max 232 yang terhubung ke computer dan kemudian diatur melalui program *CodeVisionAVR Pro v2.03.9*.

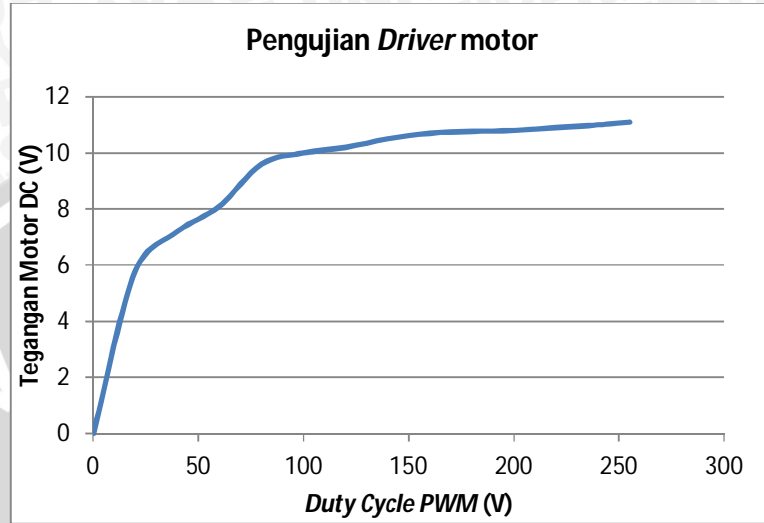
Pada Tabel 5.2 dapat dilihat data hasil pengukuran pada *driver* motor yang dilakukan dengan langkah-langkah pengujian. Dan selanjutnya pada Gambar 5.4 menunjukkan grafik hubungan antara *duty cycle* PWM yang masuk ke *driver* motor dengan tegangan yang dikeluarkan oleh *driver* driver motor. Serta Gambar 5.5 menunjukkan grafik hubungan antara *duty cycle* PWM dengan kecepatan motor DC.

Tabel 5.2 Data Pengukuran Driver Motor

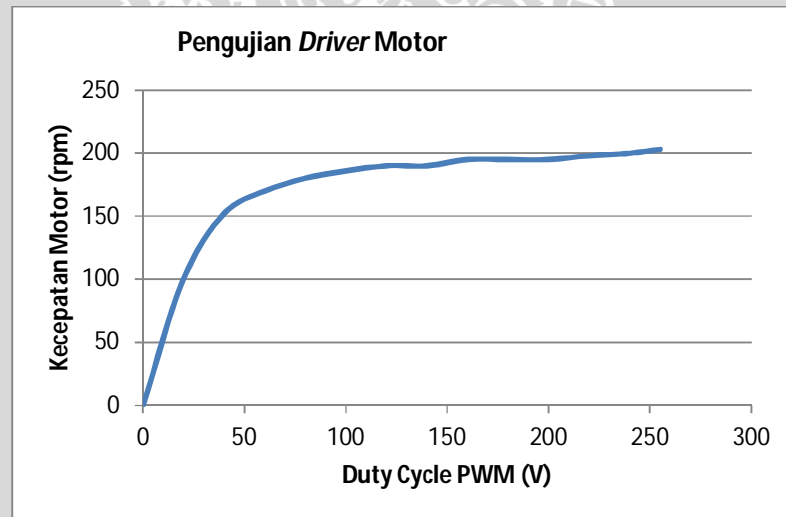
PIN PB5	PWM (PIN PB1)	Tegangan Keluaran (V)	Kecepatan Motor (rpm)	Arah Putaran motor
0	0	0	0	Diam
0	20	5.8	100	Kiri
0	40	7.2	152	Kiri
0	60	8.1	170	Kiri
0	80	9.6	180	Kiri
0	100	10	186	Kiri
0	120	10.2	190	Kiri
0	140	10.5	190	Kiri
0	160	10.7	195	Kiri
0	180	10.77	195	Kiri
0	200	10.8	195	Kiri
0	220	10.9	198	Kiri
0	240	11	200	Kiri
0	255	11.1	203	Kiri
1	0	0	0	Diam
1	20	5.8	100	Kanan
1	40	7.2	152	Kanan
1	60	8.1	170	Kanan
1	80	9.6	180	Kanan
1	100	10	186	Kanan
1	120	10.2	190	Kanan
1	140	10.5	190	Kanan
1	160	10.7	195	Kanan
1	180	10.77	195	Kanan
1	200	10.8	195	Kanan
1	220	10.9	198	Kanan
1	240	11	200	Kanan
1	255	11.1	203	Kanan

Dari Tabel 5.2 terlihat bahwa motor akan diam apabila diberi tegan 0V sedangkan arah putan motor bergantung pada Pin PB5. Pin PB 5 yang mengatur *on-off* dari *relay* pada *driver* motor, saat PB 5 berlogika 0 dan tegangan diberikan maka motor akan berputar ke arah kiri dan pada

saat PB 5 berlogika 1 atau *relay* pada keadaan *on* maka motor akan berputar kearah kanan.



Gambar 5.4 Grafik Hubungan *Duty Cycle PWM* dan Tegangan Motor DC



Gambar 5.5 Grafik Hubungan *Duty Cycle PWM* dan Kecepatan Motor DC

Dari Gambar 5.4 dan Gambar 5.5 terlihat bahwa hubungan antara *input* PWM dan *output* tegangan serta kecepatan motor DC relatif linier. Sehingga dapat dikatakan bahwa driver motor bekerja dengan cukup baik dan dapat digunakan pada pengontrolan motor DC pada miniatur sepeda motor.

5.3 Pengujian *Accelerometer*

5.3.1 Tujuan

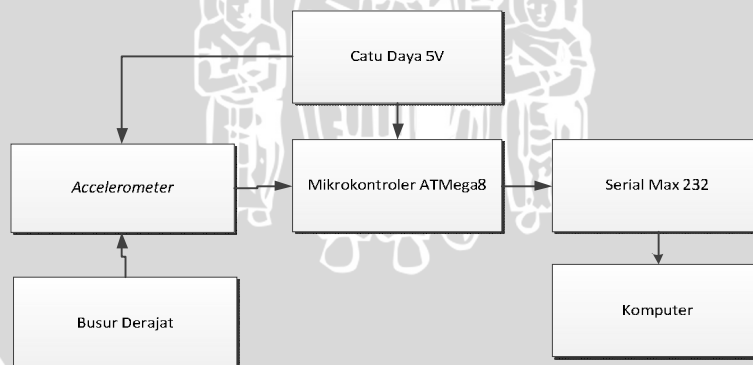
Mengetahui tingkat keakuratan dari *Accelerometer* dalam membaca perubahan sudut dari miniatur sepeda motor.

5.3.2 Peralatan yang Digunakan

- Sistem mekanik miniatur sepeda motor yang terhubung dengan *accelerometer* dan mikrokontroler ATmega8
- Penggaris busur.
- Catu daya 5 volt.
- Komputer.

5.3.3 Langkah pengujian

1. Memasang alat seperti pada Gambar 5.6.
2. Mengubah posisi kemiringan dari miniatur sepeda motor dan mengukur sudut dengan menggunakan penggaris busur
3. Mengukur tegangan keluaran *accelerometer* melalui program Visual Basic 6.
4. Menggambar data tegangan keluaran *accelerometer* pada grafik untuk melihat kelinieran dari *accelerometer*.



Gambar 5.6 Diagram Blok Pengujian *Accelerometer*

Gambar 5.6 menunjukkan diagram blok pengujian *accelerometer*. *Accelerometer* terhubung dengan mekanik miniatur sepeda motor dan mikrokontroler ATmega8. Kemiringan dari miniatur sepeda motor diubah-ubah

lalu komputer akan menunjukkan perubahan tegangan dari *accelerometer*. Tabel 5.3 menunjukkan pengamatan pada *accelerometer*.

Tabel 5.3 Menunjukkan Data Pengamatan Accelerometer

No.	Data				
	Tegangan accelerometer (Volt)	ADC	Sudut Accelerometer(°)	Sudut Aktual (°)	error
1	1.275492	261	41	41	0
2	1.298876	266	44	43	1
3	1.315445	269	45	45	0
4	1.332014	273	47	47	0
5	1.348583	276	49	49	0
6	1.413242	289	56	55	1
7	1.655906	339	83	82	1
8	1.676876	343	85	85	0
9	1.697846	348	88	88	0
10	1.718816	352	90	90	0
11	1.769997	362	96	96	0
12	1.810996	371	100	100	0
13	1.851995	379	105	105	0
14	1.892994	388	109	108	1
15	1.933993	396	114	115	1
16	1.974992	404	118	118	0
17	2.015991	413	123	124	1
18	2.05699	421	127	128	1
19	2.097989	430	132	132	0
20	2.138988	438	136	137	1

Rata – rata kesalahan *Accelerometer* dapat dihitung dengan menggunakan rumus 5.1.

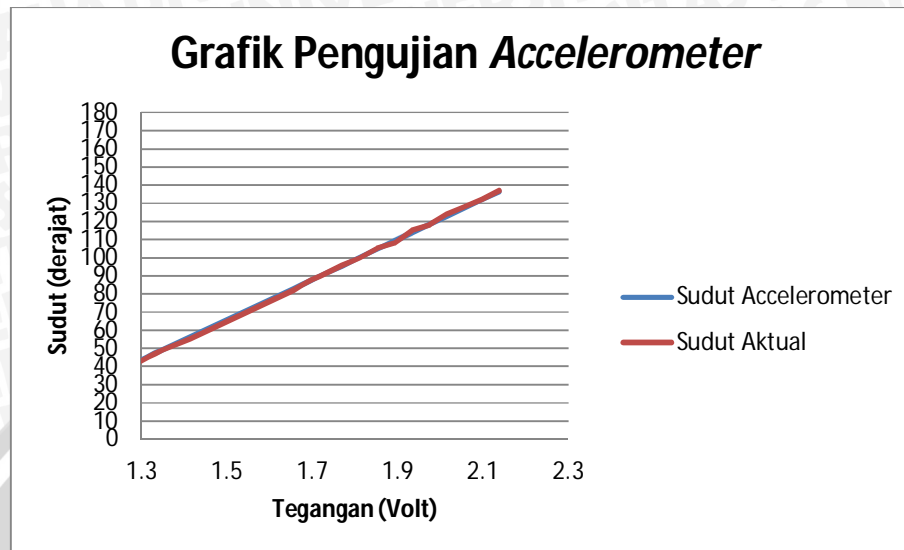
$$\text{rata – rata kesalahan (\%)} = \frac{\sum \text{error}}{\text{jumlah data}} \dots\dots\dots(5.1)$$

Sehingga didapat:

$$\text{rata- rata kesalahan (\%)} = \frac{8}{20} \dots\dots\dots(5.2)$$

$$\text{rata- rata kesalahan (\%)} = 0,4 \dots\dots\dots (5.3)$$

Gambar 5.7 menunjukkan grafik sudut pembacaan *Accelerometer* dan sudut aktual.



Gambar 5.7 Grafik Pengujian *Accelerometer*

Dari tabel 5.3 dan Gambar 5.7 terlihat bahwa pembacaan sudut *Accelerometer* sudah akurat, terlihat dari selisih error dengan sudut aktual yang sangat kecil. Dapat disimpulkan bahwa *Accelerometer* sudah bekerja dengan baik.

5.4 Pengujian Keseluruhan

5.4.1 Tujuan

Untuk mengetahui kerja dari perangkat keras dan perangkat lunak setelah diintegrasikan dalam sebuah sistem terpadu.

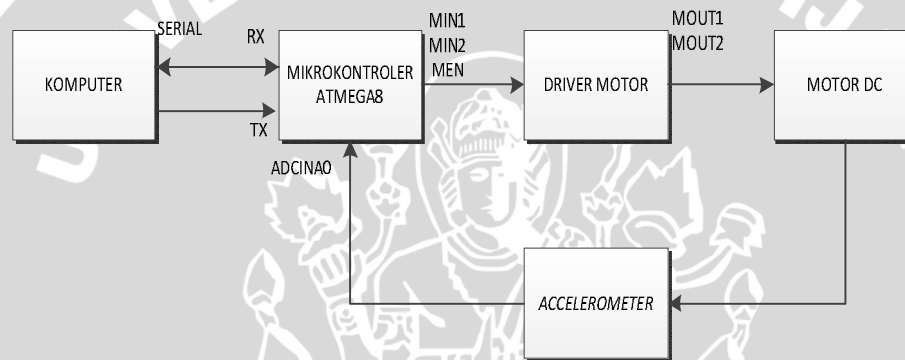
5.4.2 Peralatan yang Digunakan

- Plant miniatur sepeda motor lengkap dengan *accelerometer* dan motor DC
- Mikrokontroler ATmega8
- *Driver* motor
- Serial Port Max 232
- Program *Microsoft Excel*
- *Software Code Vision AVR Pro v2.03.9.*
- Program *Visual Basic 6* untuk menampilkan performansi sistem.

5.4.3 Langkah Pengujian

1. Memasang alat yang dibutuhkan seperti pada Gambar 5.8.
2. Mengaktifkan semua catu daya.

3. Membuat program untuk pengendalian miniatur sepeda motor dengan kontroler PID pada *Software Code Vision AVR Pro v2.03.9*.
4. Melakukan *make a project* pada *software Software Code Vision AVR Pro v2.03.9*
5. Menegakkan posisi miniatur sepeda motor terhadap posisi pijak, kemudian sistem diaktifkan.
6. Mengamati kinerja dari miniatur sepeda motor dalam menjaga keseimbangan.
7. Merekam data secara *real time* melalui program *Visual Basic 6*.



Gambar 5.8 Blok Diagram Pengujian Sistem Secara Keseluruhan

Gambar 5.8 menunjukkan diagram blok pengujian keseluruhan sistem keseimbangan miniatur sepeda motor. Mula-mula miniatur sepeda motor diberdirikan 90^0 kemudian dilepaskan untuk melihat kemampuan sistem penyeimbang dalam menjaga keseimbangan miniatur sepeda motor.

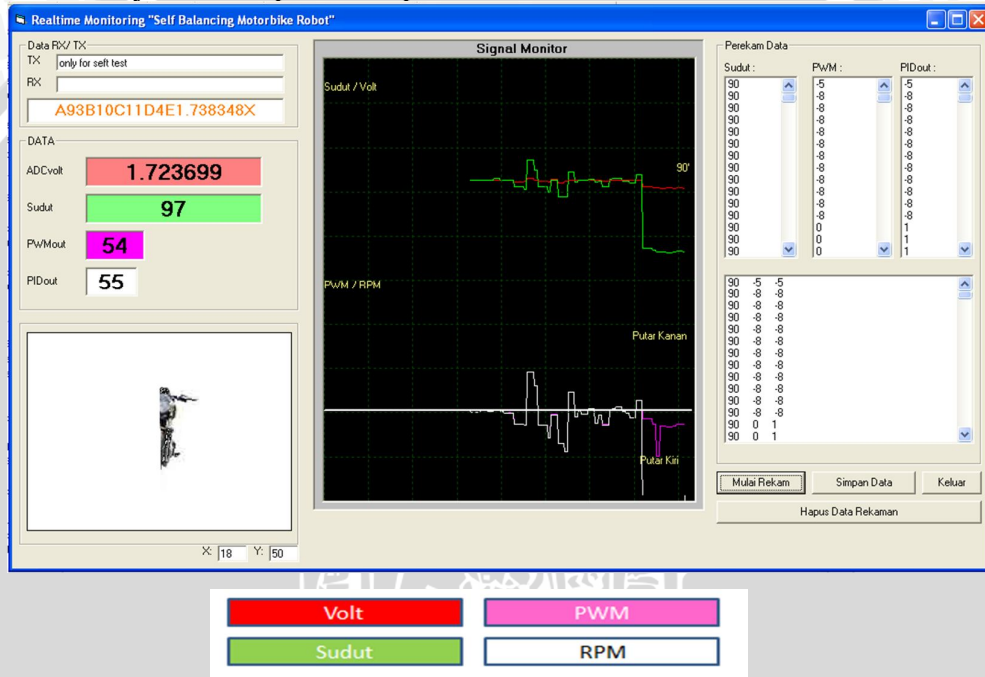
Pada pengujian ini *tuning PID* dilakukan dengan metode *hand tuning* yaitu sebuah metode dengan mengubah nilai K_p , K_i dan K_d secara manual tanpa perhitungan matematis yang didasari pengamatan terhadap respons sistem (Smith,1979). Cara ini biasa disebut dengan metode *trial and error*. Prosedur untuk melakukan *hand tuning* dapat dilihat pada bab dasar teori.

Parameter yang diamati pada skripsi ini adalah lama waktu sistem dapat bertahan pada keadaan setimbang pada posisi 90^0 sebelum *collapse* atau terjatuh. Untuk setiap parameter kontroler pengujian dilakukan hingga miniatur sepeda motor terjatuh.

Pengujian pada saat motor posisi tegak, posisi miring dan posisi jatuh untuk melihat pembacaan *accelerometer*, kinerja motor DC, PWM, dan ADC pada keseluruhan sistem serta melihat kerja *software Visual Basic 6*, sehingga bisa mengoptimalkan kinerja kontroler PID.

5.4.4 Hasil Pengujian Keseluruhan

- a) Uji Coba 1 : Parameter $K_p = 1$; $K_d = 1$; $K_i = 0.588$;
 Hasil uji coba 1 dapat dilihat pada Gambar 5.9

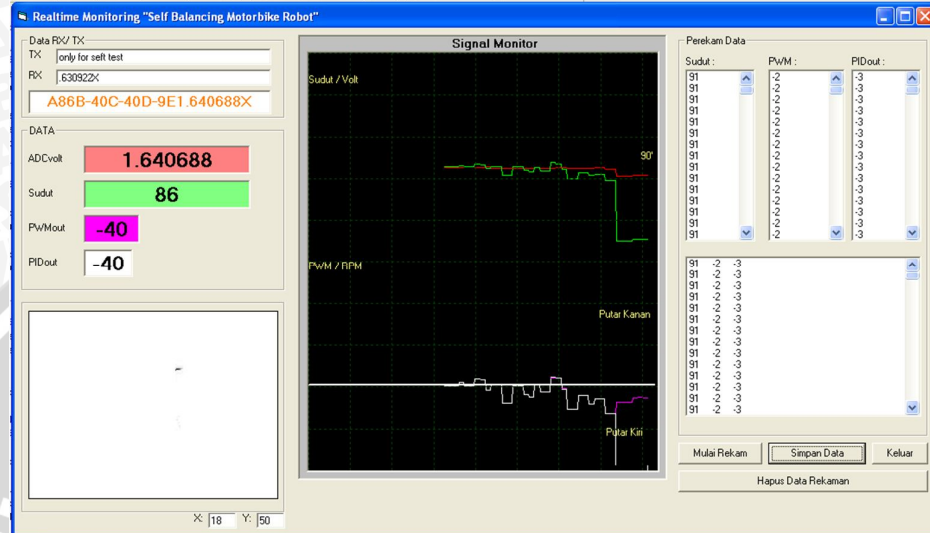


Gambar 5.9 Hasil Uji Coba 1

Dari uji coba 1 terlihat keseimbangan miniatur sepeda motor belumlah stabil, dan belum bisa mempertahankan keseimbangan miniatur sepeda motor.

b) Uji coba 2 : Parameter $K_p = 11$; $K_d = 1$; $K_i = 0.588$;

Hasil uji coba 2 dapat dilihat pada Gambar 5.10.

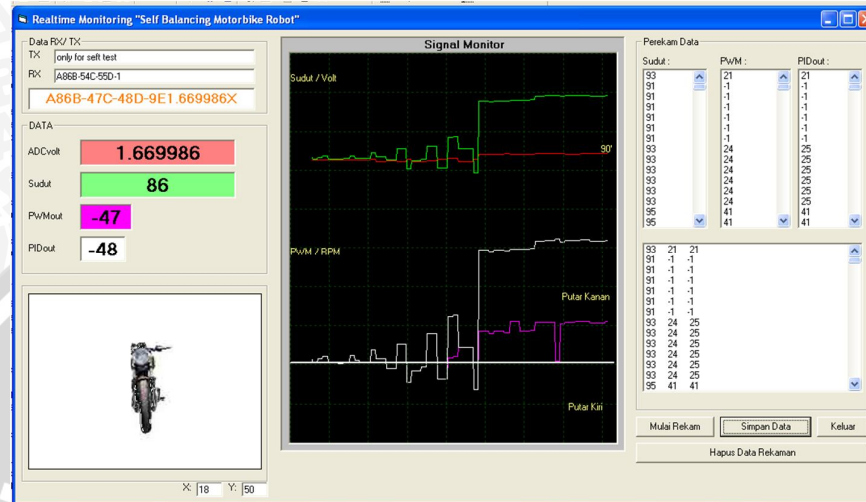


Gambar 5.10 Hasil Uji Coba 2

Dari pengujian 2 dapat dilihat bahwa dengan penambahan nilai parameter K_p , dapat memperbaiki kinerja sistem sehingga sistem dapat mempertahankan keseimbangannya kurang lebih selama 4 detik.

c) Uji Coba 3 : Parameter: $K_p = 11$; $K_d = 30$; $K_i = 0.588$;

Hasil uji coba 3 dapat dilihat pada Gambar 5.11.

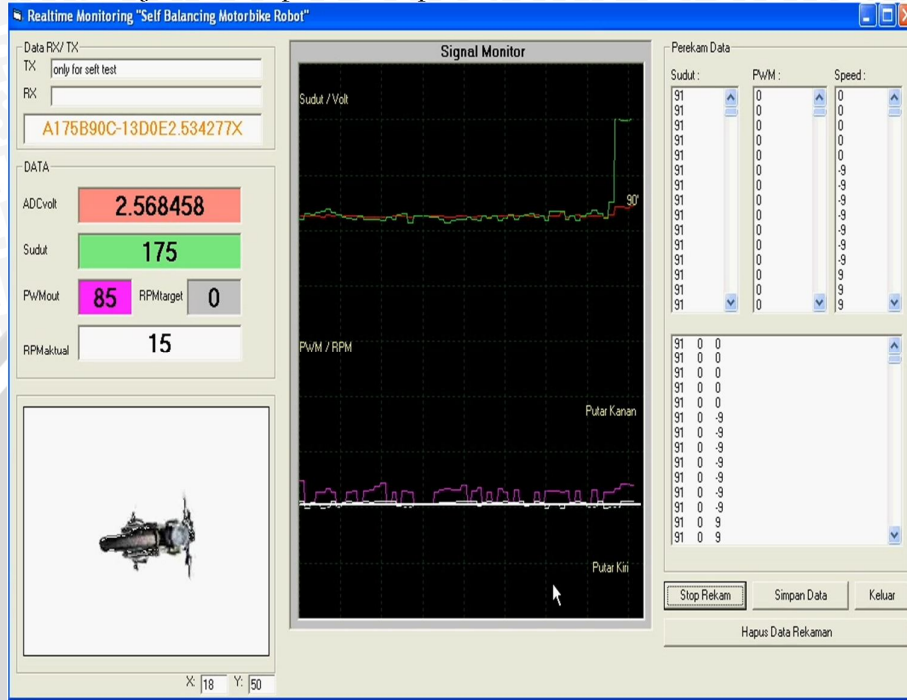


Gambar 5.11 Hasil Uji Coba 3.

Pada percobaan 3 terlihat bahwa respon sistem semakin baik dengan penambahan nilai K_d memperbaiki respon sistem dan keseimbangan miniatur sepeda motor dapat bertahan selama kurang lebih 7 detik.

d) Uji Coba 4 : Parameter: $K_p = 5$; $K_d = 58$; $K_i = 0.1$;

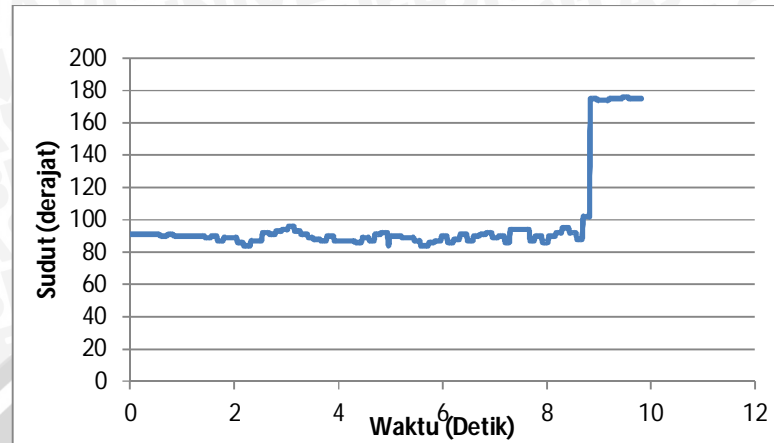
Hasil uji coba 4 dapat dilihat pada Gambar 5.12.



Gambar 5.12 Pengambilan Data Sistem Secara Keseluruhan

Berdasarkan data yang didapat pada percobaan, dengan pengambilan data selama kurang lebih 10 detik sebelum akhirnya miniatur sepeda motor terjatuh, dapat dilihat grafik respon *Accelerometer*, PWM, dan kecepatan serta arah putar motor DC.

Seperti ditunjukkan pada Gambar 5.13 adalah grafik dari perubahan sudut selama sistem bekerja yang terbaca oleh *Accelerometer*.



Gambar 5.13 Grafik Perubahan Sudut Selama Sistem Bekerja.

Pada Gambar 5.13 dapat dilihat grafik respon kecepatan putar motor DC, PWM, dan *Accelerometer* terhadap perubahan sudut pada sistem secara keseluruhan. Pada pengujian ini dapat dilihat bahwa miniatur sepeda motor yang dirancang telah berjalan dengan baik. Proses pengontrolan PID dengan nilai $K_p=5$, $K_i=58$, $K_d=0.1$. Proses penyeimbangan dengan bergerak sesuai arah dan kecepatan tertentu dapat berjalan sebagaimana mestinya.

BAB VI PENUTUP

6.1 Kesimpulan

Dari perancangan, pengujian dan pengamatan yang telah dilakukan pada miniatur sepeda motor maka dapat diambil kesimpulan sebagai berikut :

1. Penggunaan *accelerometer* sebagai sensor pembacaan kemiringan sudut sudah bekerja dengan keakuratan yang baik begitu pula dengan kinerja motor DC gearbox yang mengatur *disc inertia* sebagai upaya untuk menjaga keseimbangan sudah sesuai harapan. Namun, perancangan mekanik miniatur sepeda motor belumlah sempurna, dimana belum adanya penentuan titik keseimbangan, sehingga belum tercapainya performansi sistem sesuai dengan yang diinginkan.
2. Dengan menggunakan teorema kontroler *Proporsional, Integral dan Derivatif (PID)* yang ditanamkan pada mikrokontroler ATmega8, menggunakan *software Code Vision AVR Pro v2.03.9*, sistem dapat mengambil keputusan untuk pengaturan arah dan kecepatan motor DC sebagai upaya untuk menyeimbangkan miniatur sepeda motor. Dalam penelitian ini sistem dapat mempertahankan keseimbangan selama kurang lebih 10 detik.

6.2 Saran

Dalam perancangan dan pembuatan alat ini masih terdapat kelemahan. Untuk memperbaiki kinerja alat dan pengembangan lebih lanjut disarankan:

1. Penyempurnaan konstruksi mekanik pada sistem miniatur sepeda motor agar dicari titik keseimbangannya sehingga motor dapat berdiri tegak sebelum adanya aksi pengontrolan. Penyempurnaan pada *disc inertia* untuk dibuat lebih simetris, hal ini untuk mengurangi *noise* dan *disturbance* saat sistem sedang berjalan.
2. *Accelerometer* yang digunakan sebagai pembacaan sudut dapat diganti dengan menggunakan *gyroscope*. Dapat juga menggunakan kombinasi antara *accelerometer* dan *gyroscope* agar pembacaan sudut lebih baik dan lebih tepat.

3. Dilakukan metode pengontrolan selain menggunakan kontroler PID, untuk melihat performansi sistem apa bila menggunakan kontroler lain.
4. Menambahkan data pengujian yang digunakan untuk mendapatkan rata-rata waktu pengontrolan pada pengujian keseluruhan.



DAFTAR PUSTAKA

- ATMEL. 2007. *ATMEGA8535/ATMEGA8535L, 8-bit AVR Microcontroller with 8 Kbytes in System Programmable Flash*.
- Jantzen, J. 1998. *Tuning of Fuzzy PID Controller*, Technical University of Denmark, Department of Automation. Denmark.
- Jayanti, R. 2010. *Perancangan dan Pembuatan Alat Pengering pada Sale Pisang Menggunakan Kendali PI (Proporsional-Integral)*. Malang. Fakultas Teknik Universitas Brawijaya. Skripsi.
- Kamigh,2008 : Motor-Bike balancing robot. <http://www.youtube.com>. Diakses pada tanggal 20 februari 2012
- Ogata, K. 1997. Terjemahan *Teknik Kontrol Automatik Jilid 1*. Jakarta. Penerbit Erlangga.
- Ogata,K. 1997. Terjemahan *Teknik Kontrol Automatik Jilid 2*. Jakarta. Penerbit Erlangga
- Paralax MMA7455 3-Axis Accelerometer data sheet
- Sulistiono, A. 2010. *Pengenalan : Pulse Width Modulation*. <http://www.aristysty.cz/cc/2010/02/pulse-width-modulation-pwm-pengenalan.html>

UNIVERSITAS BRAWIJAYA

LAMPIRAN I

Program Software Code Vision AVR Pro
v2.03.9.



/******

This program was produced by the
CodeWizardAVR V2.03.9 Standard
Automatic Program Generator
© Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.
<http://www.hpinfootech.com>

Project :
Version :
Date : 5/14/2012
Author : ---
Company :
Comments:

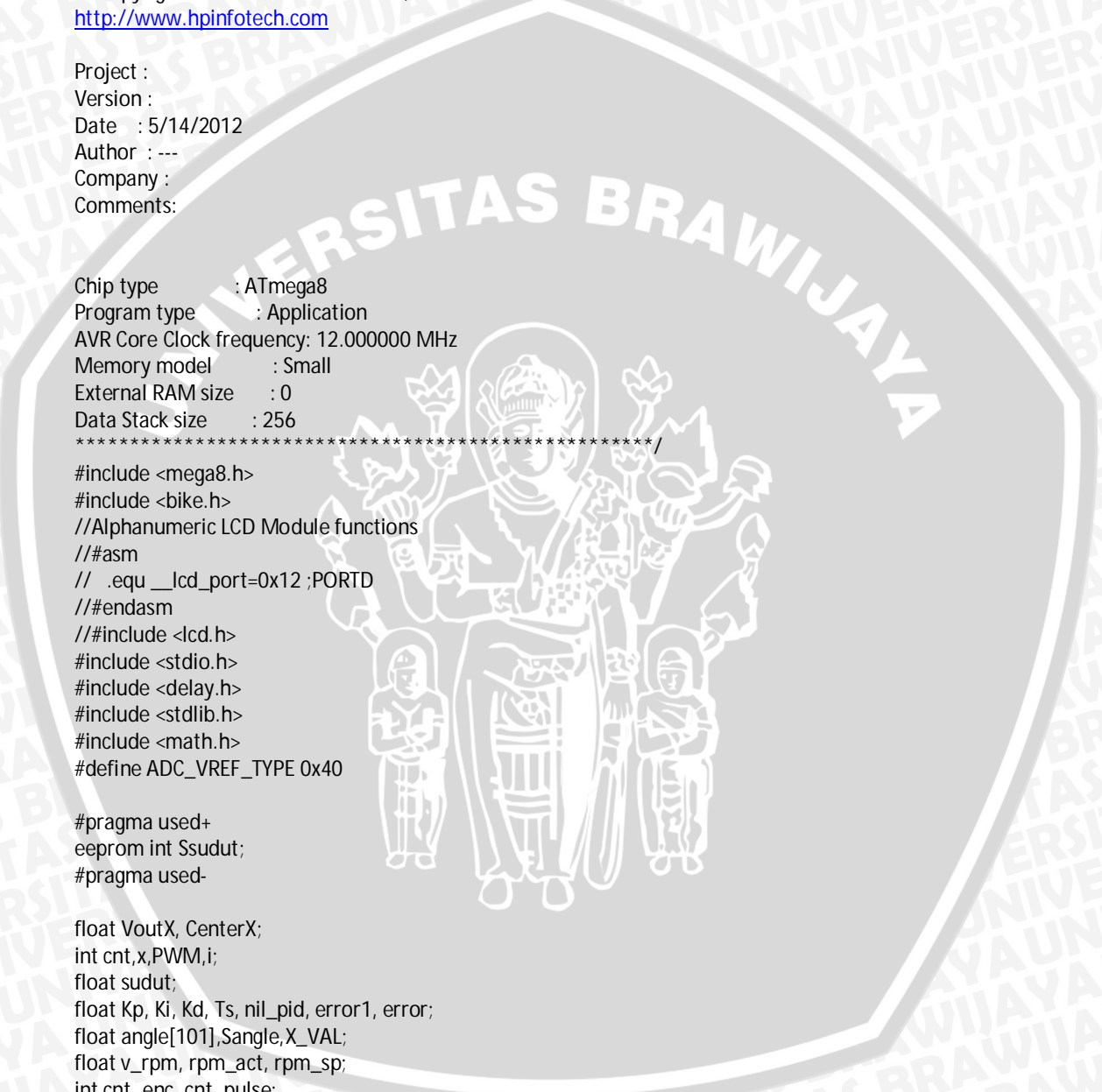
Chip type : ATmega8
Program type : Application
AVR Core Clock frequency: 12.000000 MHz
Memory model : Small
External RAM size : 0
Data Stack size : 256

*****/

```
#include <mega8.h>
#include <bike.h>
//Alphanumeric LCD Module functions
//#asm
// .equ __lcd_port=0x12 ;PORTD
//#endasm
#include <lcd.h>
#include <stdio.h>
#include <delay.h>
#include <stdlib.h>
#include <math.h>
#define ADC_VREF_TYPE 0x40
```

```
#pragma used+
eeprom int Ssudut;
#pragma used-
```

```
float VoutX, CenterX;
int cnt,x,PWM,i;
float sudut;
float Kp, Ki, Kd, Ts, nil_pid, error1, error;
float angle[101],Sangle,X_VAL;
float v_rpm, rpm_act, rpm_sp;
int cnt_enc, cnt_pulse;
int val_tunda1, val_tunda2, val_tunda3;
int flag_tunda1, flag_tunda2,flag_tunda3, flag_center, flag_motor;
int set_speed, max_pulse, cnt_rx;
float tengah;
// Deklarasi variable pada EEPROM
```



```

// Read the AD conversion result
unsigned int read_adc(unsigned char adc_input)
{
    ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
    // Delay needed for the stabilization of the ADC input voltage
    delay_us(10);
    // Start the AD conversion
    ADCSRA|=0x40;
    // Wait for the AD conversion to complete
    while ((ADCSRA & 0x10)==0);
    ADCSRA|=0x10;
    return ADCW;
}

void baca_adc()
{
    VoutX = (float)read_adc(0);
    v_rpm = (float)read_adc(1);
}

int realtime_sudut()
{
    baca_adc();           //iterasi pembacaan input ADC / sensor accelero

    for(i=0;i<100;i++)    //Filter Rata-rata 100 data
    {
        baca_adc();
        CenterX=CenterX+VoutX;
    }

    CenterX=(CenterX/100);

    for(i=0;i<100;i++)    //Filter Rata-rata 100 data
    {
        if(CenterX<=357)    //nilai 357 = nilai ADC accelero kondisi tegak lurus 90°
        {
            sudut=(((CenterX-180))*(0.564972))*0.9;
        }
        else
        {
            sudut=90+(((CenterX-357))*(0.584795))*0.9;
        }
    }

    angle[i]=sudut;
}

for(i=0;i<100;i++)
{
    Sangle=Sangle+angle[i];
}

Sangle=Sangle/100

```



```

    return Sangle;
}
void indikator(float data)
{
    if(data>95 && data<180)
    {
        PORTD.0=1;
    }
    else{PORTD.0=0;}

    //if(data>91 && data<95) {PORTD.2=1;} else{PORTD.2=0;}

    if(data>89 && data<91) {PORTD.4=1;} else{PORTD.4=0;}

    if(data<89&& data>85) {PORTD.6=1;} else{PORTD.6=0;}

    if(data<85 && data>10) {PORTB.0=1;} else{PORTB.0=0;}
}
// External Interrupt 0 service routine
interrupt [EXT_INT0] void ext_int0_isr(void)
{
    // counter encoder
    cnt_enc++;
    // cnt_pulse++;
}
// Timer 2 overflow interrupt service routine
interrupt [TIM2_OVF] void timer2_ovf_isr(void)
{
    cnt++;
    if(flag_tunda1==1){val_tunda1++;}
    if(flag_tunda2==1){val_tunda2++;}
    if(x==1)
    {
        OCR1A=abs(PWM);
    }

    cnt_pulse++;
    if(cnt>24000) //reset counter 1 detik
    {

        //rpm_act=(cnt_enc/16)*60; //RPM tidak dipakai
        rpm_act=cnt_enc; //ini yang dipakai Pulse per second yg lebih presisi
        cnt_enc=0;
        cnt=0;
    }

    cnt_rx++;
    if(cnt_rx>12000)
    { if(Sangle<180 && Sangle>0)
    {

```

```

if(PWM>=0)
{
printf("A%.0fB%.0fC%.0fD%.0fE%fX",Sangle,abs(PWM),rpm_act,rpm_sp,VoutX*0.004883);
}

if(PWM<0)
{
printf("A%.0fB-%.0fC%.0fD%.0fE%fX",Sangle,abs(PWM),rpm_act,rpm_sp,VoutX*0.004883);
}
}
cnt_rx=0;
}
}
void putar_motor(int set_speed, int max_pulse)
{
if(flag_motor==0){cnt_pulse=0;}

if(cnt_pulse<max_pulse)
{
if(PWM<set_speed)
{
PWM=PWM+5;
}
else
{
PWM=PWM-5;
}
flag_motor=1;
}
else
{
PWM=0;
}
}

// Declare your global variables here

void main(void)
{
setup();
flag_tunda1=0;
// ADC initialization
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x84;
#asm("sei")
//PWM=255;
while (1)
{
//=====//
//proses ini digunakan untuk 'PERBAIKAN ALAT'
if(x==0) //Proses Kalibrasi mencari nilai ADC/Sensor Accelero/tegak lurus/90'
{ //dilakukakn hanya saat awal program.

```



```

for(i=0;i<100;i++)
{
    baca_adc();
    CenterX=CenterX+VoutX;
}

CenterX=CenterX/100;
X_VAL=CenterX;
x=1;

printf("Loading data..");
delay_ms(5000);
}

//=====PROSES PEMBACAAN
SUDUT=====//

    realtime_sudut();        //mendapatkan nilai sudut secara realtime "Sangle"

    indikator(Sangle);        //tampilan indikator sudut 0 - 180' menggunakan LED

//=====END=====
==//

//=====iterasi kontroler PID / Set Kecepatan Motor=====//

Kp=5;        //PARAMETER KONTROLER
Ki=58;
Kd=01;
Ts=0.001;

error = tengah; //rpm_sp - rpm_act;
nil_pid=((Kp*error)+((Ki/10)*(error+error1)*Ts)+((Kd/Ts)*(error-error1)));
PWM=PWM+nil_pid;

PWM=PWM-1;

if(PWM>254)        //Limit
{
    PWM=254;
}
if(PWM<-255)
{
    PWM=-255;
}

error1=error;

//=====END=====
//

```



```
//=====BALANCING
SYSTEM=====//
tengah=CenterX-357;

if(tengah>1)
{
    tengah=tengah*1.15;
}
if(tengah>2)
{
    tengah=tengah*1.1;
}
if(tengah>3)
{
    tengah=tengah*1.1;
}
if(tengah>4)
{
    tengah=tengah*1;
}
if(tengah>5)
{
    tengah=tengah*0.9;
}

if(tengah<-1)
{
    tengah=tengah*1.2;
}
if(tengah<-2)
{
    tengah=tengah*1;
}
if(tengah<-3)
{
    tengah=tengah*0.9;
}
if(tengah<-4)
{
    tengah=tengah*0.8;
}
if(tengah<-5)
{
    tengah=tengah*0.7;
}

//PWM=pow(tengah,2);
PWM=nil_pid;
```



```

if(tengah<0){PWM=PWM*-1;}

if(PWM<-1){ PORTB.4=1;PORTB.5=0;}
if(PWM>1){ PORTB.4=0;PORTB.5=1;}

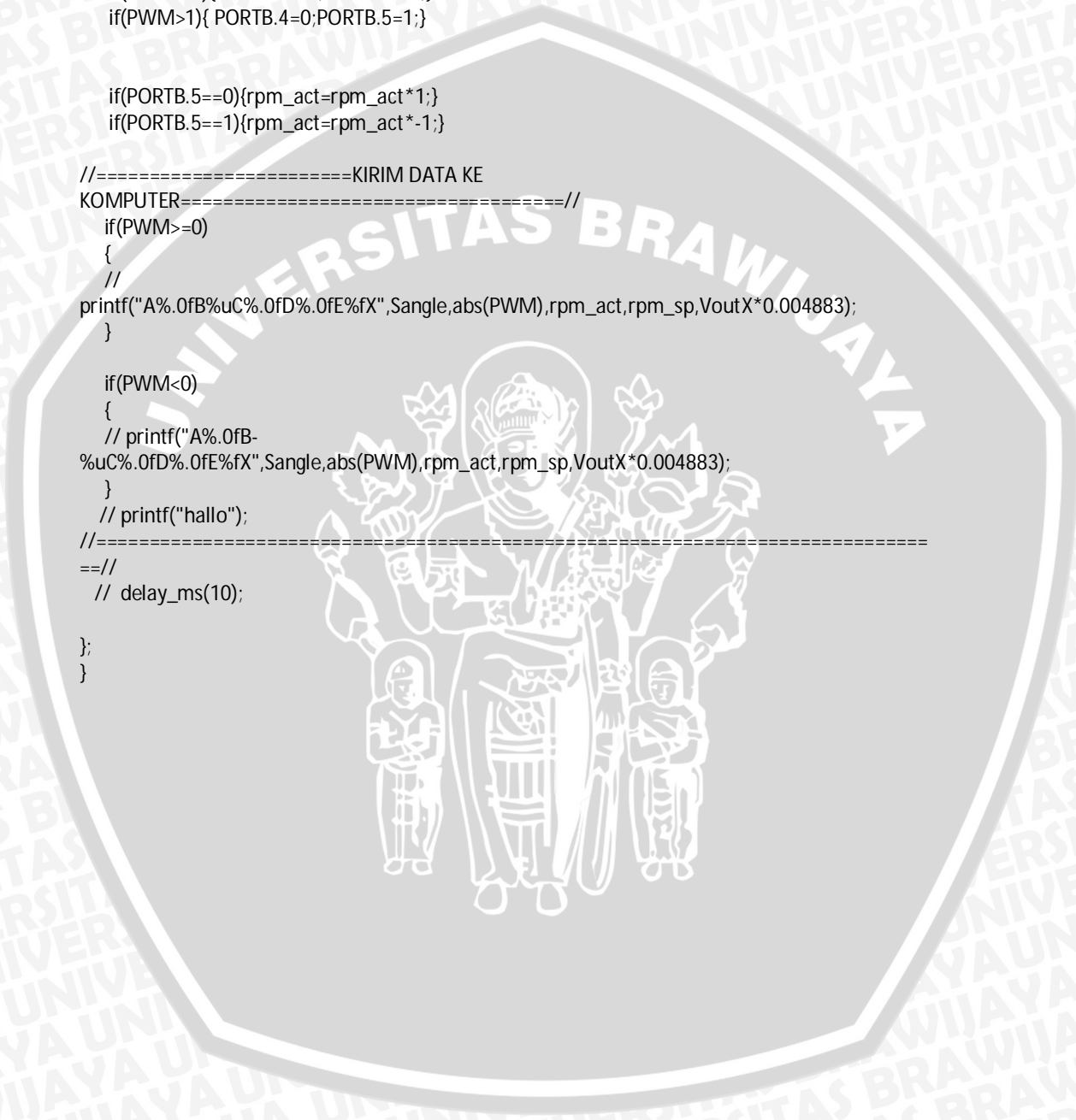
if(PORTB.5==0){rpm_act=rpm_act*1;}
if(PORTB.5==1){rpm_act=rpm_act*-1;}

//=====KIRIM DATA KE
KOMPUTER=====//
if(PWM>=0)
{
//
printf("A%.0fB%.0fD%.0fE%fX",Sangle,abs(PWM),rpm_act,rpm_sp,VoutX*0.004883);
}

if(PWM<0)
{
// printf("A%.0fB-
%.0fD%.0fE%fX",Sangle,abs(PWM),rpm_act,rpm_sp,VoutX*0.004883);
}
// printf("hallo");
//=====
==//
// delay_ms(10);

};
}

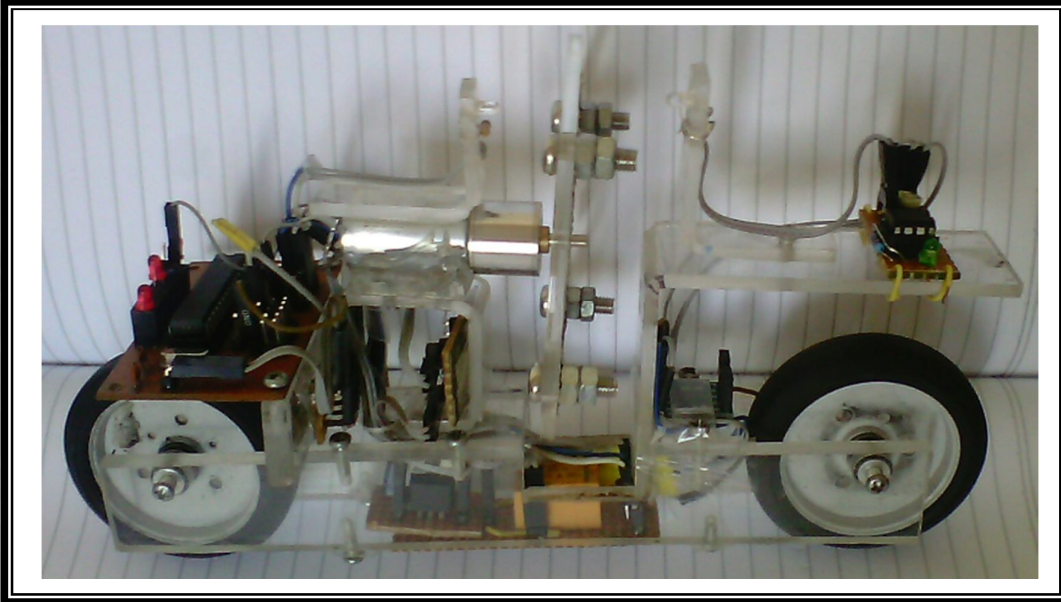
```



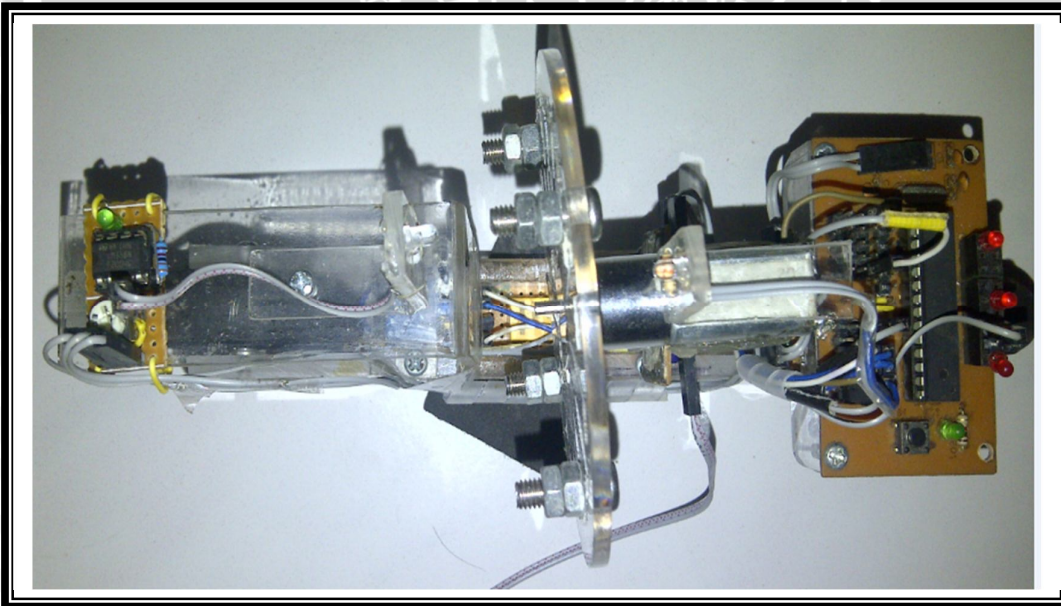


LAMPIRAN II

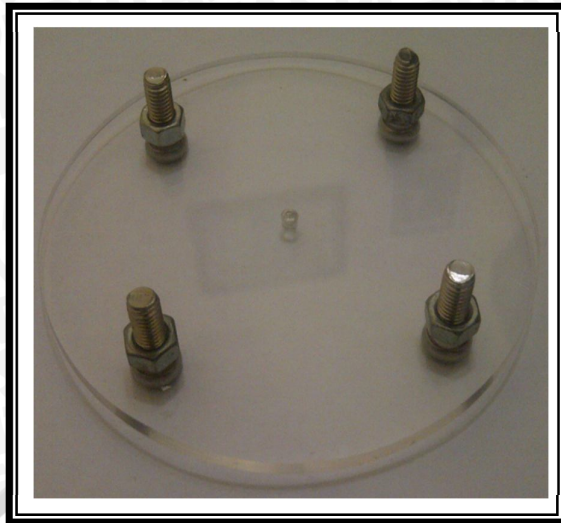
Foto Miniatur Sepeda Motor



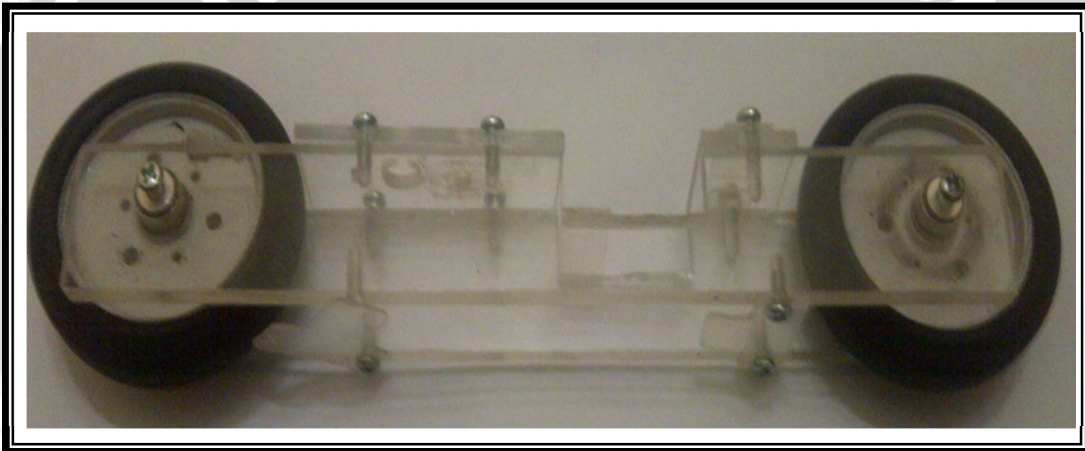
Miniatur Sepeda Motor Tampak Samping



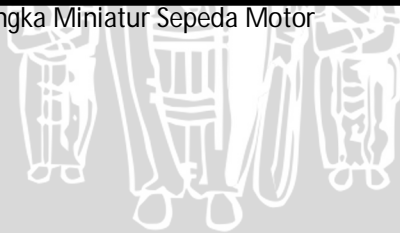
Miniatur Sepeda Motor Tampak Atas



Disc Inertia



Kerangka Miniatur Sepeda Motor





LAMPIRAN III

Datasheet

Features

- High-performance, Low-power Atmel® AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 130 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16MIPS Throughput at 16MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
 - 8Kbytes of In-System Self-programmable Flash program memory
 - 512Bytes EEPROM
 - 1Kbyte Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Three PWM Channels
 - 8-channel ADC in TQFP and QFNMLF package
 - Eight Channels 10-bit Accuracy
 - 6-channel ADC in PDIP package
 - Six Channels 10-bit Accuracy
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-lead PDIP, 32-lead TQFP, and 32-pad QFNMLF
- Operating Voltages
 - 2.7V - 5.5V (ATmega8L)
 - 4.5V - 5.5V (ATmega8)
- Speed Grades
 - 0 - 8MHz (ATmega8L)
 - 0 - 16MHz (ATmega8)
- Power Consumption at 4MHz, 3V, 25°C
 - Active: 3.6mA
 - Idle Mode: 1.0mA
 - Power-down Mode: 0.5µA



8-bit **AVR**[®]
with 8KBytes
In-System
Programmable
Flash

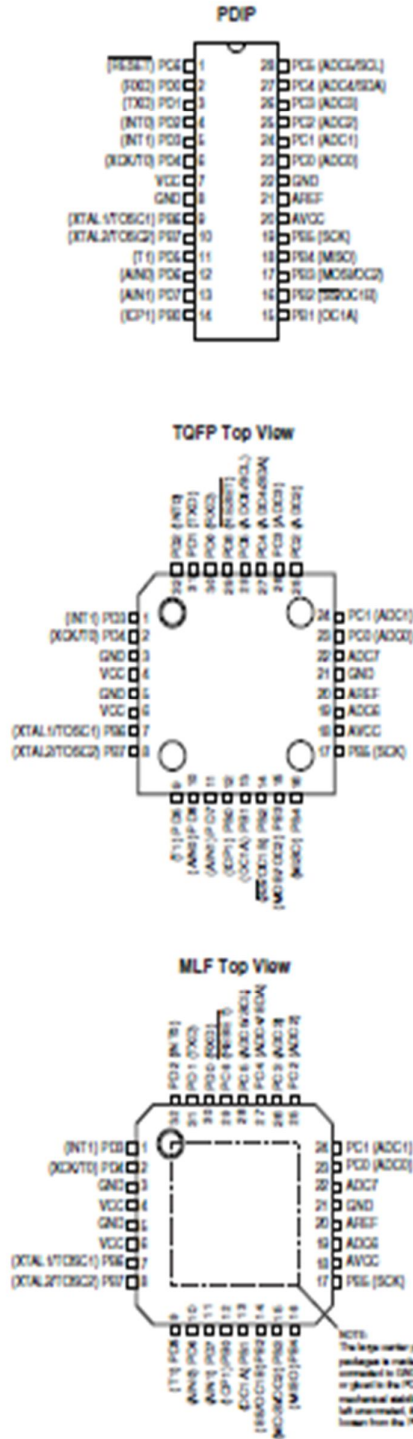
ATmega8
ATmega8L

Rev 2486Z-AVR1-02/11





Pin Configurations



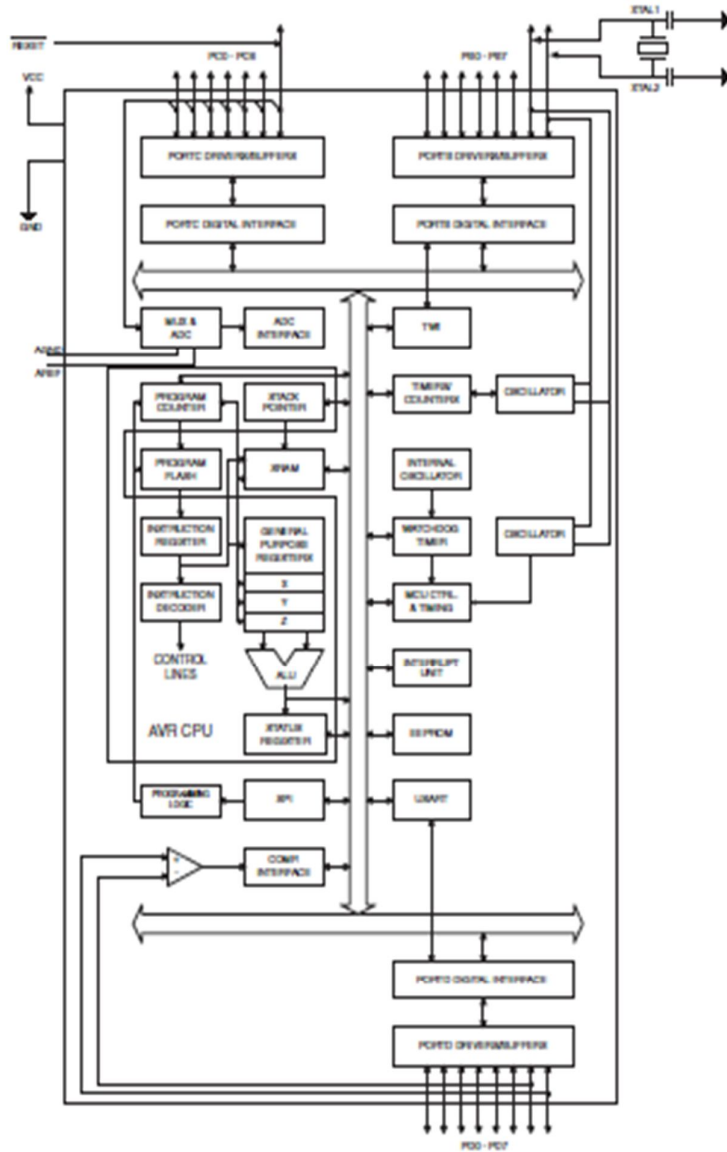
ATmega8(L)

Overview

The Atmel®AVR® ATmega8 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8 achieves throughputs approaching 1MIPS per MHz, allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 1. Block Diagram





Web Site: www.parallax.com
 Forums: forums.parallax.com
 Sales: sales@parallax.com
 Technical: support@parallax.com

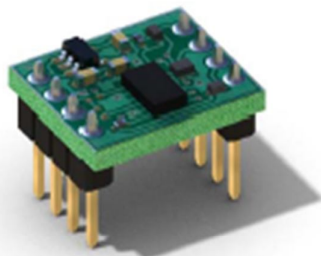
Office: (916) 824-8333
 Fax: (916) 824-8003
 Sales: (888) 512-1024
 Tech Support: (888) 967-8267

MMA7455 3-Axis Accelerometer Module (#28526)

General Description

The Freescale Semiconductor MMA7455L 3-Axis Digital Output Accelerometer is a low power, micro-machined sensor capable of measuring acceleration along its X, Y, and Z axes. It offers several convenient integrated features including an analog to digital converter (ADC), digital low-pass filter, and selectable sensitivity ranges of $\pm 2g$, $\pm 4g$, or $\pm 8g$. This device can be easily configured to detect quick motion pulses as single-taps, double-taps, and 0g (free-fall) conditions on any or all axes and provides configurable interrupt pins (INT1 and INT2) for each type of event.

Flexibility and compatibility are designed into this module. An on-board voltage regulator and I/O voltage level-shifters make this module especially easy to connect to virtually any microcontroller. It operates over a wide range of supply voltages from 2.5 VDC to 5.5 VDC and communicates via Serial Peripheral Interface (SPI) or Inter-Integrated Circuit (I²C) bus.



Features

- Wide input voltage range
- Digital output (SPI/I²C)
- Selectable sensitivity: 8-bit mode ($\pm 2g/\pm 4g/\pm 8g$) or 10-bit mode ($\pm 8g$)
- User-configurable interrupts (INT1/INT2)
- User-settable registers for calibrating each axis
- Low current operation
- Compact, 8-pin DIP breadboard-friendly package: 0.5" x 0.6" (12.7 x 15.2 mm)

Application Ideas

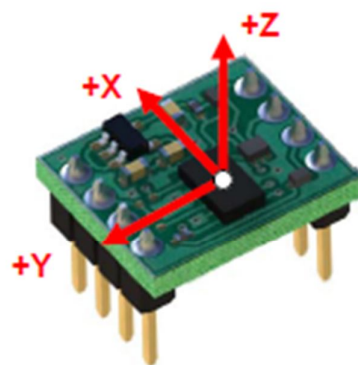
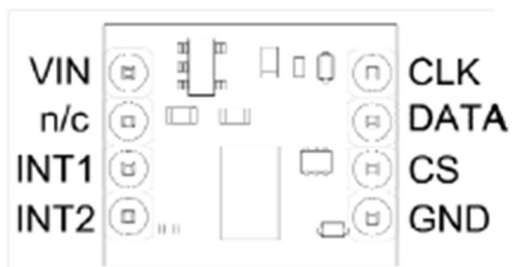
- Inertial measurement units
- Tilt sensing for remote control systems
- Single-tap or double-tap user input
- Multi-axis vibration measurement
- Free-fall detection
- Motion sensing and data logging
- Model airplane flight stabilization

Key Specifications

- Power requirements: 2.5 – 5.5 VDC; 26 μ A standby, 0.5 – 3 mA active
- Communication: SPI or I²C
- Dimensions: 0.5 x 0.6 x 0.46 in (12.7 x 15.2 x 11.7 mm)
- Operating temperature: -40 to +185 °F (-40 to +85 °C)

Note: Demonstration software files may be downloaded from the 28526 product page at www.parallax.com.

Package and Pin Descriptions



Pin #	Pin Name	Description	Pin Type
1	VIN	Input supply voltage	Power
2	n/c	Not connected – reserved for future use	n/a
3	INT1	Interrupt 1/Data Ready	Output
4	INT2	Interrupt 2	Output
5	GND	Ground	Power
6	CS	Chip select - SPI Enable (0), I ² C Enable (1)	Input
7	DATA	I ² C Serial Data (SDA), SPI Serial Data (SDI/SDO)	Input/Output
8	CLK	I ² C Serial Clock (SCL), SPI Serial Clock (SPC)	Input

Absolute Maximum Ratings

NOTE: These are conditions beyond which permanent device damage may occur; it is not implied that the device will function properly throughout this range. For operational ranges, see the Operating Conditions section below.

Parameter	Min	Typ	Max	Unit
V _{IN} (Measured from GND)	0	--	6.5	V
Storage temperature	-40	--	+125	°C
DATA Pin				
Input Voltage	-0.5	--	V _{IN} + 0.5	V
Pin Current	-50	--	+50	mA
Maximum Acceleration (all axes)	--	--	5000	g

Operating Conditions

Parameter	Min	Typ	Max	Unit
V _{IN} Supply Voltage	2.5	3.3 – 5	5.5	V
Supply Current ⁽¹⁾				
Operational (Sensing) Mode ⁽¹⁾⁽²⁾	--	437	555	µA
Operational Mode & Pins Loaded ⁽¹⁾⁽³⁾	--	<3	4.3	mA
Standby Mode ⁽¹⁾⁽⁴⁾	--	28	42	µA
Operating Temperature Range	-40	25	85	°C
Output Data (Measurement) Rate				
Data Rate is 125 Hz when 82.5 Hz BW selected	--	125	--	Hz
Data Rate is 250 Hz when 125 Hz BW selected	--	250	--	Hz
Sensitivity ⁽⁵⁾ ±2g range, 8-bit	58	64	70	count/g
0g Output Signal ⁽⁵⁾ ±2g range, 8-bit	-21	0	21	count
SPI Clock Frequency	--	4	--	MHz

Notes:

- Supply current measured flowing into VIN pin. V_{IN} = 5.5 V
- Operational Mode current measured with MMA7455L device placed in Measurement Mode and CLK, DATA, and CS pins disconnected.
- Operational Mode & Pins Loaded is measured as worst-case current draw with CLK, DATA, and CS pins all connected to ground; INT1 and INT2 pins left disconnected.
- Standby Mode current measured with MMA7455L device placed in standby mode and CLK and DATA lines left high (connected to V_{IN}) or disconnected. Value may be higher if these lines are loaded.
- Values taken from Freescale Semiconductor MMA7455L device specifications.

Principle of Operation

The Freescale MMA7455L is a surface-machined Micro Electro-mechanical System (MEMS) formed by masking and etching semiconductor materials. It can be visualized as a very small mass suspended in the center of the device by tiny springs. When an acceleration is applied to the device, the mass deflects along one or more axes inside the device. Built-in circuitry constantly measures the amount of deflection along each axis and translates it into acceleration data. Then the data becomes available to be read by an external microcontroller. For more detailed information refer to the Freescale MMA7455L product documentation.

I/O Level Shifting

Part of what makes this 3-Axis Accelerometer module so useful is its wide power supply and I/O voltage compatibility range of 2.5 V to 5.5 V. The MMA7455L device operates over a supply voltage range of 2.4 V to 3.6 V; so to make it compatible with devices operating up to 5.5 V, a 3.3 V Low Dropout (LDO) voltage regulator was installed, as well as two types of level shifting circuits. This allows the device to operate and communicate over the full voltage supply range.

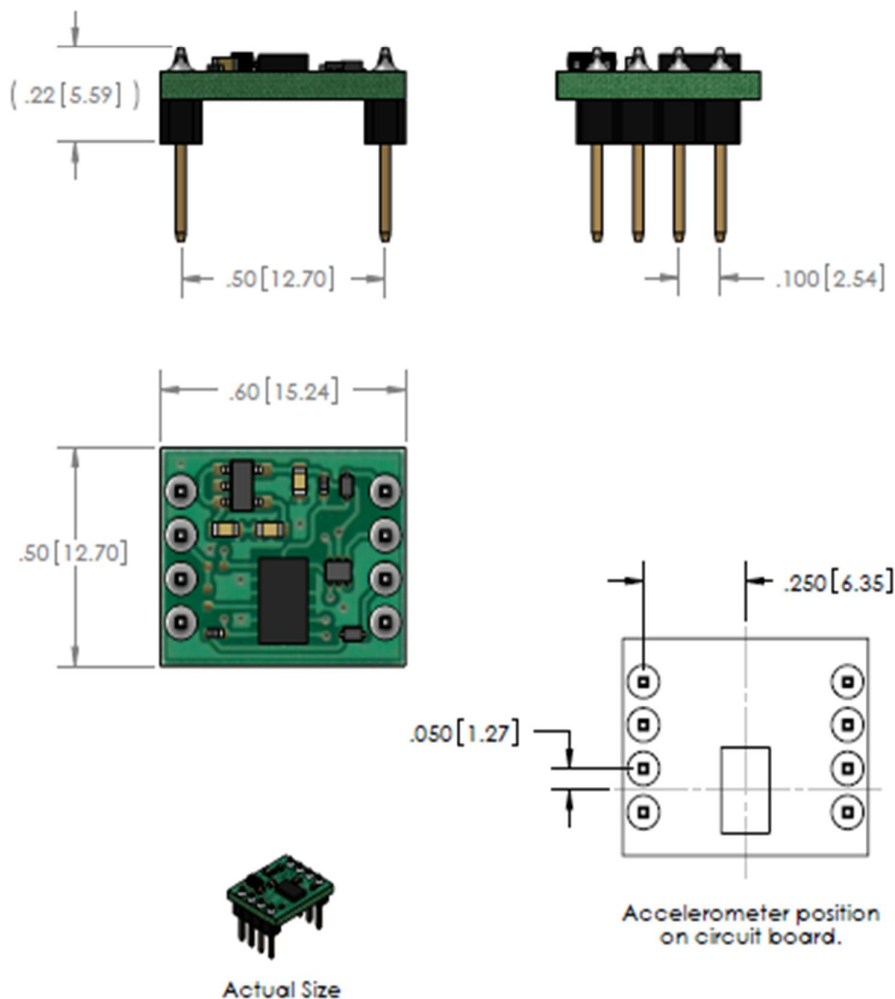
For the DATA pin, a bi-directional level shifting IC is used to translate the lower I/O voltage of the MMA7455L to the possibly higher voltage used by an externally connected device. This particular level shifter has built-in pull-up resistors to each supply rail so it can be driven by open drain outputs for I²C; but can also be driven low/high for SPI protocol. Additionally, it automatically detects the direction of data flow and switches the output drivers accordingly.

The CLK and CS pins operate exclusively as inputs, so simpler and cheaper level shifting circuits can be used. Only a resistor and Schottky diode are required for each pin. See the Module Schematic section for details. When an external device drives the CLK or CS pin low, the I/O pin on the MMA7455L is pulled

Material: ---

NOTES

1. This drawing is intended for customer reference only.
2. Dimensions and board layout are subject to change without notice.



PROPRIETARY INFORMATION THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF PARALLAX, INC. ANY REPRODUCTION IN PART OR IN WHOLE WITHOUT THE WRITTEN PERMISSION OF PARALLAX, INC. IS PROHIBITED.	PARALLAX <small>INC.</small>		DATE 20090813
	599 MENLO DRIVE ROCKLIN, CA 95765 www.PARALLAX.com (916) 624-8333	Parent Part # 28526	REV A
UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES (mm) LENGTH TOLERANCES: X.XX ± 0.020 IN X.XXX ± 0.005 IN ANGLE TOLERANCES: X ± 10° X.X ± 5°	TITLE MMA7455 3-Axis Accelerometer		
SCALE: 3:1	DRAWN BY: K. McCullough	SHEET 1 OF 1	