



UNIVERSITAS INDONESIA

**RANCANGAN SISTEM MEKANIK PENGGERAK DETEKTOR DUA
DIMENSI UNTUK FANTOM AIR BERBASIS MIKROKONTROLER
H8/3069F**

SKRIPSI

RUSYDA TAQIYYA

0706262735

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
PROGRAM STUDI FISIKA
DEPOK
MEI 2012**

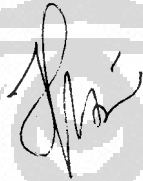
HALAMAN PERNYATAAN ORISINALITAS

**Skripsi ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : RUSYDA TAQIYYA

NPM : 0706262735

Tanda Tangan :



Tanggal : 28 Mei 2012

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama : RUSYDA TAQIYYA

NPM : 0706262735

Program Studi : Instrumentasi Elektronika S1

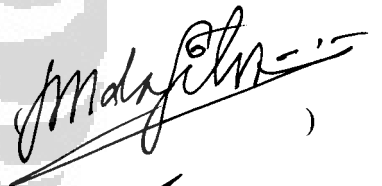
Judul Skripsi :

RANCANGAN SISTEM MEKANIK PENGGERAK DETEKTOR DUA
DIMENSI UNTUK FANTOM AIR BERBASIS MIKROKONTROLER
H8/3069F

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Sains pada Program Studi Fisika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia

DEWAN PENGUJI

Pembimbing I : Prof. Dr. BEF da Silva, M.Sc, DEA




Pembimbing II : Dr. Eng. Supriyanto, M.Sc



Penguji I : Dr. Santoso Soekirno, M.Si



Penguji II : Dr. Tony Mulia, PhD



Ditetapkan di : Depok

Tanggal : 28 Mei 2012

KATA PENGANTAR

Fabi ayyi aala irobbikuma tukadzibaan.. Segala puji bagi Allah SWT yang Maha Pengasih dan Maha Penyayang, atas segala nikmat berupa iman, ihsan dan jasmani sehingga penulis dapat menyelesaikan skripsi ini dengan sebaik-baiknya pada waktu yang tepat.

Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Sains Jurusan Fisika pada Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia.

Penulis menyadari bahwa, selesainya skripsi ini tidak terlepas dari bantuan, bimbingan, dorongan dan doa yang tulus dari banyak pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini. Tanpa itu semua sangatlah sulit bagi penulis untuk menyelesaikan skripsi ini. Untuk itu penulis ingin menyampaikan ucapan terima kasih yang setulus – tulusnya kepada :

1. Keluargaku tercinta : Abi dan ummi yang tidak pernah putus mendo'akan dan memberi dorongan untuk anak-anaknya. Juga saudara-saudaraku: Ka Aida dan keluarga kecilnya : Ka Rahmat dan si kecil Khalid, Milla yang rajin, Umair yang berwibawa dan Usman yang kreatif. *Kalianlah inspirasiku...*
2. Prof. Dr. BEF Da Silva selaku dosen pembimbing pertama yang telah dengan senang hati berdiskusi dengan kami dan meminjamkan buku-bukunya.
3. Dr. Eng. Supriyanto, M.Sc selaku dosen pembimbing kedua yang telah mengenalkan penulis dengan mikrokontroler H8/3069F, membimbing penulis dalam segala hal, baik dalam ilmu pengetahuan, saran, nasehat, dorongan, semangat dan segala fasilitas yang telah diberikan dalam menyelesaikan tugas akhir ini.
4. Bapak Dwi Seno Kuncoro Sihono, M.Si selaku dosen Fisika Medis yang telah banyak membantu penulis dalam hal survey di lapangan.
5. Dr. Santoso Soekirno selaku penguji I dan Dr. Tony Mulia selaku penguji II atas saran dan kritiknya baik sebelum maupun sesudah penulis melakukan sidang.

6. Bapak Lingga Hermanto M.Si yang dengan senang hati meminjamkan ruang Laboratorium Fisika Lanjutan untuk saya dan kawan-kawan dalam rangka menyelesaikan skripsi ini.
7. Dosen-dosen Instrumen Dept. Fisika Universitas Indonesia super baik dan sabarnya : Pak Prawito, Pak Sastra, Pak Arif, Pak Lingga, dan Pak Marta. Terima kasih atas segala bimbingan yang sudah bapak berikan pada kami..
8. Teman-teman seperjuangan dalam menyusun skripsi : Dewi Imas, Yulia Nur Fitri, Melly Amelina, dan Melati Azizka. *Rasanya lepas segala beban dan keluhan kalau udah kumpul dengan kalian...i love you sista..:*)
9. Pa Karno "Bungkar" yang sudah memberikan bimbingan programming kepada penulis. *Lengkap rasanya jadi anak H8 kalo udah dikasih dodol sama bapak, hehe...*
10. Semua anggota tim riset H8 (dan yang ex-) : Ka Muchtar, Ka Ilham, Ka Tyo, Ka Thea, Ka Tiwi, Ka Nurma, Ka Moku, Ka Ami, Ka Rizki, Ka Lia, Ka Faizal, Nella, Harits, Fikri, Salas, Herlina, Yusuf, Faldo, Ferry, Vika, dan Cherry. *Walaupun sekarang kita udah pada mencar-mencar dengan kesibukan masing-masing, berada satu tim dengan kalian adalah hal yang tak terlupakan.. Terima kasih atas semua perhatian dan bantuan kecil atau besar yang sudah kalian berikan..:*)
11. Fisika angkatan 2007, Instrumenista 2007 dan Instrumenista 2008 yang tidak bisa saya sebut satu persatu namanya. *Time runs so fast ya..mau kumpul-kumpul lagi sama kalian..*
12. Mbak Ratna dan kru di sekret, terima kasih atas segala pelayanan yang sudah diberikan untuk para skripsweet yang bolak-balik ngurus ini itu.
13. Saudari-saudariku yang paling solehah: Mbak Lys, Ka Hanifah (Malaysia), Eliza (dan Wi'am kechil), Tanti KD (mentoring), Estry (pulsa), Anifah (don't be panic), Nurul (uyung), Dea (Donghae), Iradah, Sumi dan Ka Hanifah.
14. Rekan-rekan @BeIUC yang dengan sabarnya memahami penulis dengan segala kesibukan skripsi : Bu Yuli, Bu Suji, Pak Syam, Bu Ellya, Pak Andi, @saifulfathan, @robanymushab, @ulil15, @ririrainbow,

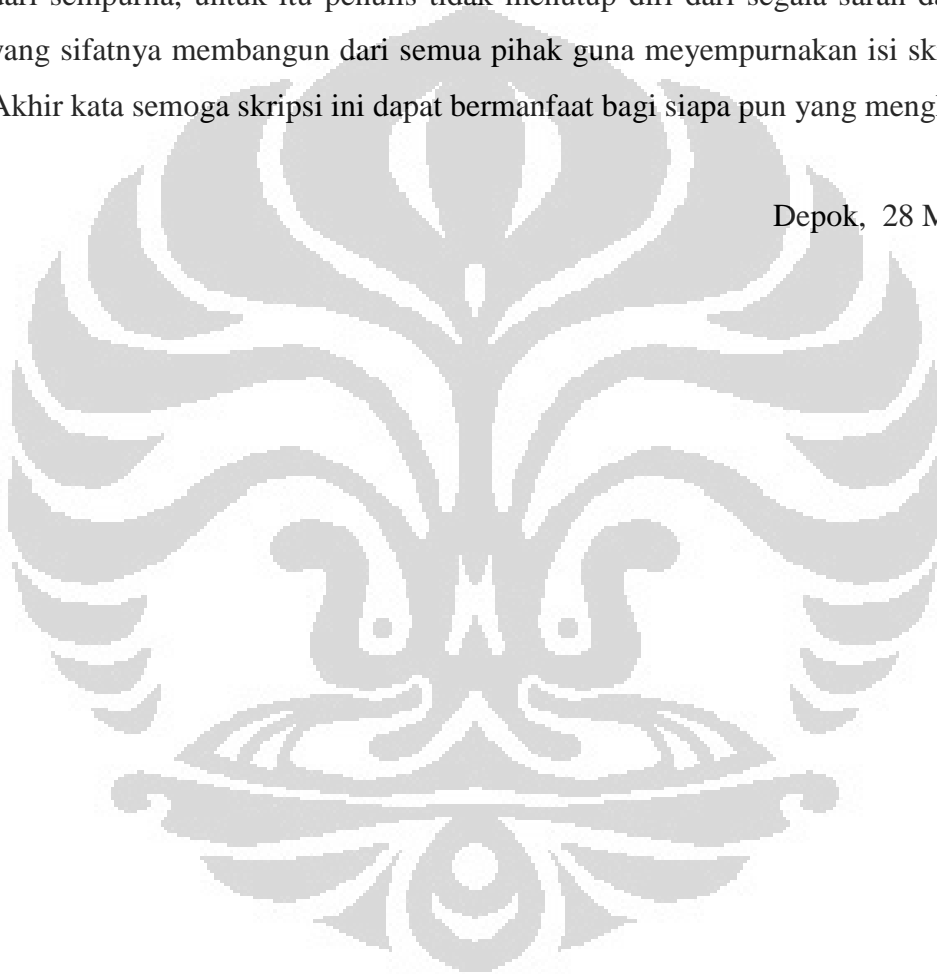
@TheSyaifan, @LailaLN, @ozanbewegung, @amelalwaysjihrah,
@QonitaHakim, @faizrob dan lainnya yang belum follow sayah..;p

15. Pihak lainnya yang secara sadar, tidak sadar, sengaja atau tidak sengaja turut membantu pembuatan skripsi ini. Terima kasih atas do'a dan bantuannya.

Menyadari kekurangan dalam penulisan skripsi ini serta kemungkinan jauh dari sempurna, untuk itu penulis tidak menutup diri dari segala saran dan kritik yang sifatnya membangun dari semua pihak guna menyempurnakan isi skripsi ini. Akhir kata semoga skripsi ini dapat bermanfaat bagi siapa pun yang mengkajinya.

Depok, 28 Mei 2012

Penulis



**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini :

Nama : Rusyda Taqiyya
NPM : 0706262735
Program Studi : Instrumentasi Elektronika S1
Departemen : Fisika
Fakultas : Matematika dan Ilmu pengetahuan Alam
Jenis karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

**RANCANGAN SISTEM MEKANIK PENGGERAK DUA DIMENSI
DETEKTOR UNTUK FANTOM AIR BERBASIS MIKROKONTROLER
H8/3069F**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*Database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : 28 Mei 2012

Yang menyatakan



(Rusyda Taqiyya)

Nama : Rusyda Taqiyya
Program studi : Fisika
Judul skripsi : Rancangan Sistem Penggerak Detektor Fantom Air Dua Dimensi Berbasis Mikrokontroler H8/3069F

ABSTRAK

Rancangan sistem mekanik penggerak detektor dua dimensi untuk fantom air telah dibuat dengan menggunakan mikrokontroler H8/3069F. Sistem ini dilengkapi dengan dua buah servo motor *continuous* yang digunakan untuk memutar poros ulir pada sumbu vertikal dan horizontal sehingga detektor holder dapat bergerak dua dimensi sejauh 30 cm. Arah gerak servo motor *continuous* diatur dengan menggunakan metode *Pulse Width Modulation (PWM)* melalui Bahasa C yang bekerja pada mikrokontroler. Posisi aktual dari detektor dapat diketahui melalui sensor *rotary encoder* yang dipasang satu sumbu dengan motor. Pengaturan posisi detektor secara manual dapat dilakukan melalui PC dengan memanfaatkan tampilan *Graphical User Interface (GUI) Python*. RS-232 digunakan untuk komunikasi antara komputer dan mikrokontroler. Ketelitian dari alat ini adalah 0.67% untuk sumbu vertikal dan 0.33% untuk sumbu horizontal.

Kata Kunci : Fantom air, Ulir, Servo motor *continuous*, Mikrokontroler H8/3069F, PWM, Bahasa C, Rotary Encoder, GUI, Python, RS-232.

Name : Rusyda Taqiyya
Program of study : Physics
Title of Essay : Mechanical Drive System Design of Two-Dimensional
Detector for Water Phantom Based Microcontroller
H8/3069 F

ABSTRACT

Mechanical drive system design of two-dimensional detector for water phantom had been done by using microcontroller H8/3069F. This system is equipped with two continuous servo motors in order to rotate the threaded shaft on vertical and horizontal axis so that the detector holder can move two dimensionally as far as 30 cm. The direction of continuous servo motor is set by using Pulse Width Modulation (PWM) method through C Language that works on the microcontroller. The actual position of detector can be determined by rotary encoder sensor which is assembled in one shaft with motor. Setting the position of the detector can be automatically done through a PC by using the display Graphical User Interface (GUI) in Python. RS-232 is used for communication between computer and microcontroller. The accuracy of this device is 0.67% for vertical axis and 0.33% for horizontal axis.

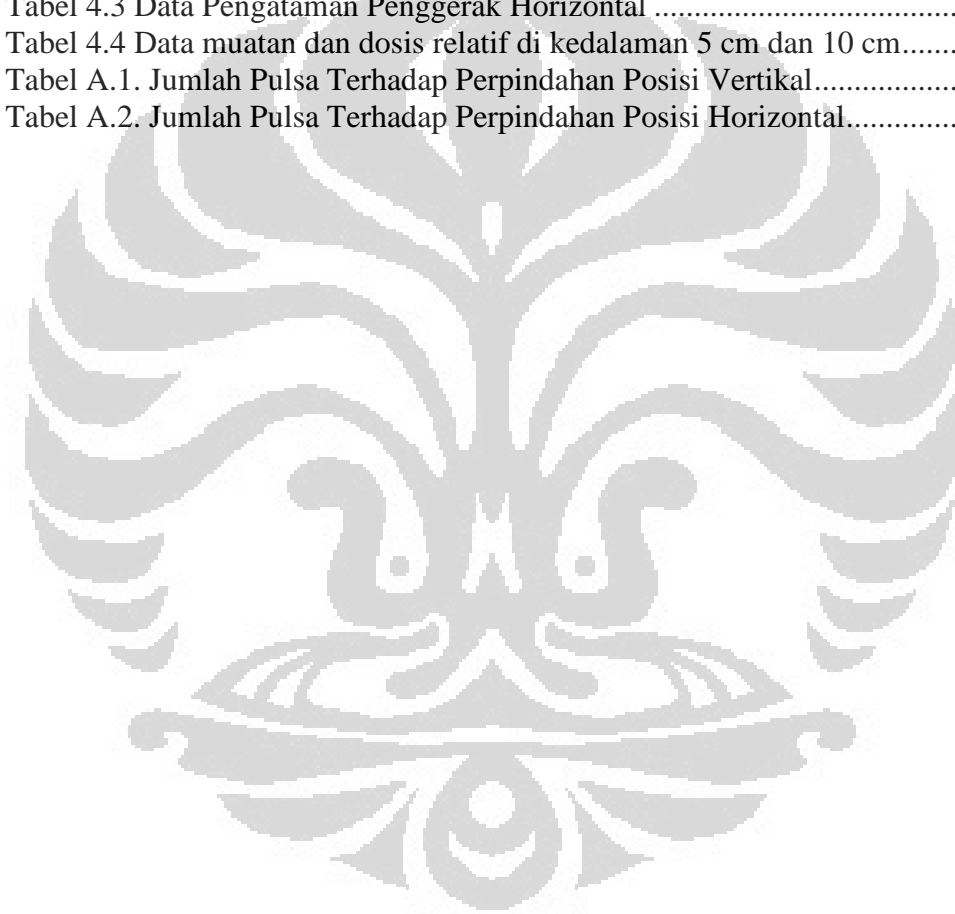
Key Words : Water phantom, Thread, Continuoud servo motor, Microcontroller H8/3069F, PWM, C Language, Rotary Encoder, GUI, Python, RS-232.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH	vi
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiii
BAB 1 PENDAHULUAN	14
1.1 Latar Belakang Masalah	14
1.2 Batasan Masalah	15
1.3 Tujuan dan Manfaat Penelitian	15
1.4 Metodologi Penelitian	15
1.5 Sistematika Penulisan	16
BAB 2 LANDASAN TEORI	17
2.1 Fantom Air pada Radioterapi	18
2.2 Profil Berkas Radiasi	19
2.3 Detektor Radiasi	20
2.4 Servo Motor Continuous	21
2.5 Spur Gear	22
2.6 Rotary Encoder	24
2.7 Batang Ulir	25
2.8 Mikrokontroler H8/3069F	25
2.9 Python	32
BAB 3 PERANCANGAN ALAT	33
3.1 Desain Sistem	33
3.2 Sistem Mekanik	34
3.3 Pemasangan Servo motor Continuous	36
3.4 Sensor dan Rangkaian Pengkonversi Tegangan	37
3.5 Prosedur Pengambilan Data	39
BAB 4 HASIL DAN PEMBAHASAN	41
4.1 Pemrograman Melalui Mikrokontroler H8/3069F	41
4.2 Prosedur Download	46
4.3 GUI	47
4.4 Analisis Servo Motor	48
4.5 Data Perbandingan Pulsa Encoder Terhadap Posisi Vertikal Detektor	50
4.6 Data Perbandingan Pulsa Encoder Terhadap Posisi Horizontal Detektor	52
4.7 Kurva Profil Radiasi Cobalt-60	55
BAB 5 PENUTUP	57
5.1 Kesimpulan	57
5.2 Saran	57
DAFTAR PUSTAKA	58
LAMPIRAN	60

DAFTAR TABEL

Tabel 2.1 Pin-pin Interrupt.....	27
Tabel 2.2 Fungsi bit TPSC pada register 16TCR	31
Tabel 2.3 Fungsi bit CCLR pada register 16TCR.....	31
Tabel 2.4 Fungsi bit STR pada register TSTR.....	31
Tabel 3.1 Fungsi pin pada mikrokontroler H8/3069 F	36
Tabel 3.2 Konfigurasi dan fungsi pin untuk komunikasi serial pada mikrokontroler H8/3069 F.....	37
Tabel 3.3 Output Rotary Encoder	38
Tabel 4.1 Contoh Sistem Kontrol Melalui Hyperterminal.....	41
Tabel 4.2 Data Pengataman Penggerak Vertikal	51
Tabel 4.3 Data Pengataman Penggerak Horizontal	54
Tabel 4.4 Data muatan dan dosis relatif di kedalaman 5 cm dan 10 cm.....	54
Tabel A.1. Jumlah Pulsa Terhadap Perpindahan Posisi Vertikal.....	60
Tabel A.2. Jumlah Pulsa Terhadap Perpindahan Posisi Horizontal.....	68

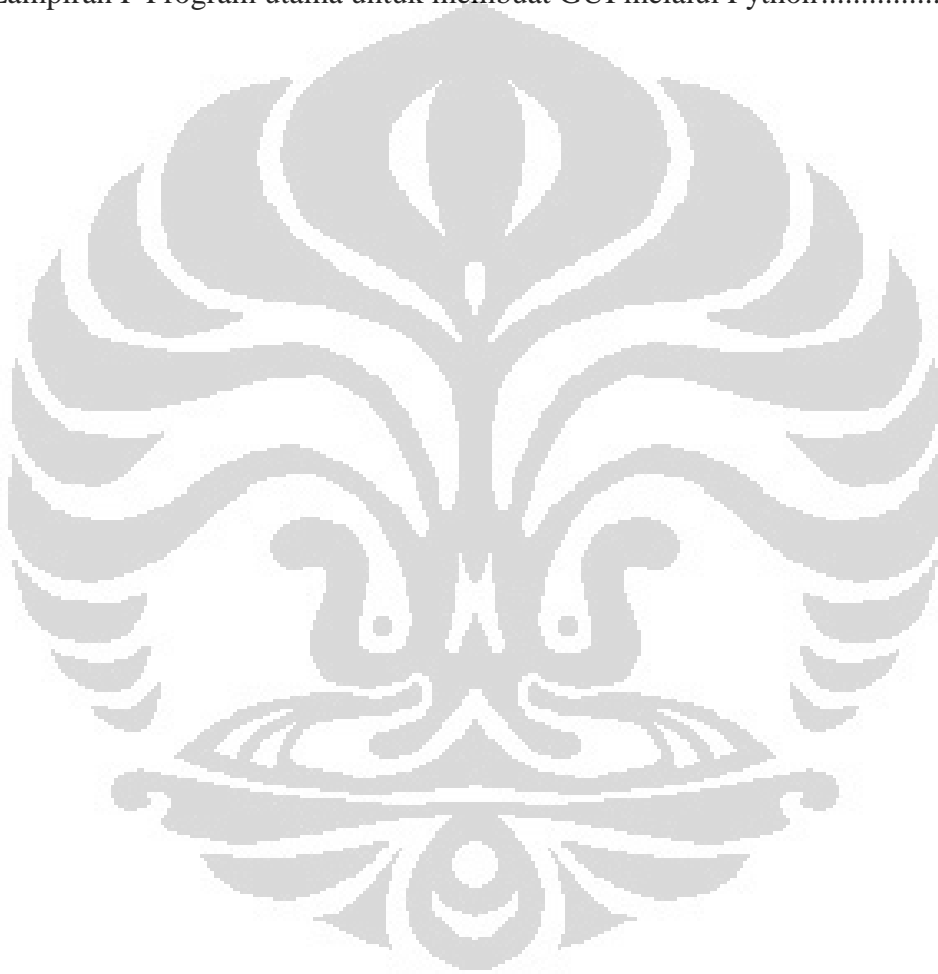


DAFTAR GAMBAR

Gambar 1.1 Diagram alir langkah-langkah penelitian.....	16
Gambar 2.1 Fantom Air.....	18
Gambar 2.2 Perubahan profil berkas sinar-x akibat perjalanan elektron yang berbeda-beda.....	19
Gambar 2.3 Dosis efektif yang diharapkan terhadap 'profil' berkas yang dihasilkan	20
Gambar 2.4 Rangkaian sederhana detektor ionization chamber	20
Gambar 2.5 Detektor ionization chamber	21
Gambar 2.6 Teknik pengaturan lebar pulsa untuk mengatur arah servo motor continuous	22
Gambar 2.7 Susunan kabel servo motor.....	22
Gambar 2.8 Spur gear.....	23
Gambar 2.9 Rotary encoder jenis hollow	24
Gambar 2.10.a Contoh output kedua channel dengan searah jarum jam (CW)	24
Gambar 2.10.b Contoh output kedua channel dengan berlawanan arah jarum jam (CCW).....	24
Gambar 2.11 Batang ulir	25
Gambar 2.12 Minimum sistem mikrokontroler H8/3069F.....	26
Gambar 2.13 Blok diagram interupsi	28
Gambar 2.14.a Mode PWM dengan counter di-clear-kan oleh GRA	29
Gambar 2.14.b Mode PWM dengan counter di-clear-kan oleh GRB	29
Gambar 2.15 Prosedur penggunaan PWM mode pada mikrokontroler H8/3069F	30
Gambar 2.16 Bit-bit pada register 16TCR	30
Gambar 2.17 Bit-bit pada register TSTR	31
Gambar 3.1 Blok diagram perangkat keras sistem pengendali posisi detektor	33
Gambar 3.2 Rancangan sistem mekanik penggerak detektor.....	34
Gambar 3.3 Susunan motor, spur gear dan sensor	35
Gambar 3.4 Sistem pembacaan putaran motor dengan sensor rotary encoder.....	38
Gambar 4.1 Flowchart program interupsi	42
Gambar 4.2 Flowchart program	43
Gambar 4.3 Susunan switch mode Write	46
Gambar 4.4 Cara mendownload program mikrokontroler dengan Bahasa C.....	47
Gambar 4.5 Susunan switch mode Read	47
Gambar 4.6 Tampilan awal GUI.....	48
Gambar 4.7 Pulsa high sebenarnya pada servo motor continuous	50
Gambar 4.8 Grafik perbandingan pulsa rotary encoder terhadap perpindahan posisi vertikal	51
Gambar 4.9 Grafik perbandingan pulsa rotary encoder terhadap perpindahan posisi horizontal	53
Gambar 4.10 Perbandingan profil dosis relatif untuk kedalaman 5 cm dan 10 cm untuk SSD = 80 cm dan lapangan 10 x 10 cm	57
Gambar C.1 <i>Minimum System H8/3069F</i>	76
Gambar C.2 Blok diagram H8/3069F	77
Gambar C.3 <i>Pin arrangement H8/3069F</i>	78
Gambar D.1 Jenis dan spesifikasi sensor rotary encoder	79
Gambar D.2 Koneksi dan output rotary encoder.....	80
Gambar D.3 Dimensi sensor rotary encoder	81

DAFTAR LAMPIRAN

Lampiran A Data Pengamatan Pulsa Terhadap Perpindahan	60
Lampiran B Mikrokontroler H8/3069F	76
Lampiran C Datasheet Rotary Encoder Autonics	79
Lampiran D Datasheet Ionization Chamber	82
Lampiran E Script utama program bahasa C (main.c).....	83
Lampiran F Program utama untuk membuat GUI melalui Python.....	88



BAB I

PENDAHULUAN

1.1 Latar Belakang

Salah satu manfaat radiasi ion aktif pada fisika medis adalah untuk pengobatan penyakit kanker, atau yang lebih dikenal dengan radioterapi. Pengobatan kanker melalui radioterapi ini dilakukan dengan pemaparan radiasi pada bagian yang terkena kanker yang diberikan secara berkala dengan menggunakan pesawat akselerator. Dosis radiasi yang diberikan pada penderita kanker haruslah setepat-tepatnya terhadap jaringan yang sakit (*target volume*) tanpa memberikan efek atau kerusakan yang berarti pada jaringan sehat disekitarnya^[1]. Untuk mengetahui apakah radiasi tepat mengenai target volume atau tidak, kita harus melihat profil dari berkas elektron yang dipaparkan. Oleh karenanya, sebelum digunakan pada penderita, pesawat akselerator haruslah dievaluasi stabilitasnya terlebih dahulu dengan menggunakan fantom air dua dimensi (horizontal dan vertikal) untuk mendapatkan 'profil' berkas radiasi tersebut sehingga didapat jumlah dosis yang efektif.

Fantom air digunakan sebagai model dari tubuh manusia yang akan disinari radiasi. Alat ini dilengkapi dengan penggerak detektor yang mampu bergerak secara vertikal dan horizontal disesuaikan dengan letak sel kanker dalam tubuh. Detektor ini berada di dalam media berupa air yang menyerupai komponen tubuh kita yang mayoritas terdiri oleh cairan.

Akan tetapi, hingga saat ini kebanyakan rumah sakit di Indonesia memiliki fantom air yang diproduksi di luar negeri dengan harga yang cukup mahal. Akibatnya, bukan hanya akan menghambat penerapan radioterapi di banyak wilayah di Indonesia namun juga ketika ada kerusakan alat, akan sangat sulit memperbaikinya karena tidak tersedianya ahli fantom air dan suku cadang yang dibutuhkan.

Salah satu cara untuk mengatasi masalah tersebut adalah dengan mencoba untuk mengembangkan sendiri fantom air buatan anak negeri. Dengan begitu, permasalahan mengenai harga akan terpecahkan karena akan lebih murah baik dari

segi bahan baku dan pembuatan. Selain itu jika terjadi kerusakan, akan mudah memperbaikinya.

Di samping manfaat akan biaya dan perawatan, akan berimbas pula pada sisi pengetahuan dan penerapan ilmu pengetahuan itu sendiri. Desain dan inovasi tentunya akan digunakan dalam penerapan alat kalibrasi fantom air dari waktu ke waktu.

1.2 Batasan Masalah

Permasalahan yang hendak diselesaikan pada penelitian ini adalah bagaimana membuat sistem mekanik penggerak detektor dua dimensi (horizontal dan vertikal) sejauh 30 cm untuk fantom air yang efektif dan harga yang terjangkau.

Gerak dari sistem mekanik ini memanfaatkan gerak servo motor continuous sebagai pemutar drat yang akan menggerakkan detektor holder. Arah gerak motor ini diatur dengan menggunakan metode Pulse Width Modulation (PWM) melalui mikrokontroler H8/3069F. Pengguna dapat mengatur posisi detektor dengan komputer PC melalui *Graphical User Interface (GUI) Python*.

1.3 Tujuan Penelitian

Penelitian ini bertujuan membuat sistem mekanik yang efisien dan terjangkau untuk menggerakkan detektor radiasi secara horizontal dan vertikal pada fantom air yang dapat dikontrol dari jarak jauh melalui mikrokontroler H8/3069F.

1.4 Metodologi Penelitian

Metode penelitian yang akan dilakukan terdiri dari beberapa tahap antara lain :

a. Studi kepustakaan

Studi kepustakaan dilakukan untuk mengumpulkan informasi dan pustaka yang berkaitan dengan penelitian ini baik dari literatur, internet, buku-buku, serta penjelasan informal dari pihak terkait dan lainnya.

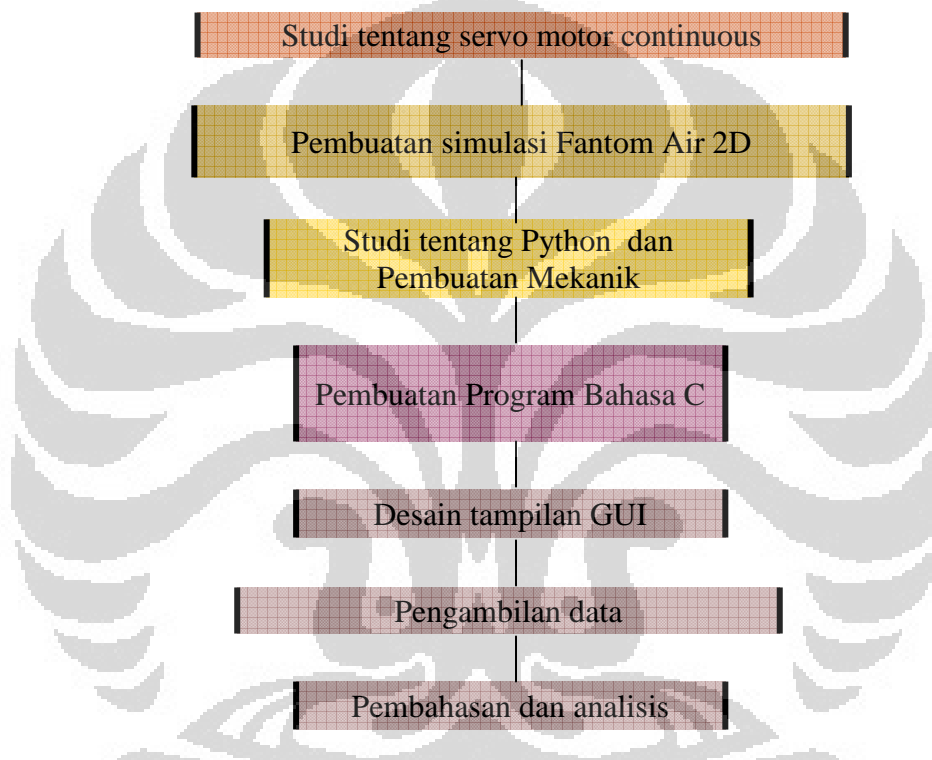
b. Studi peralatan

Studi peralatan bertujuan mempelajari karakteristik dan spesifikasi alat yang akan digunakan sehingga diperoleh pembelajaran yang tepat tentang alat yang akan dipakai.

c. Penelitian laboratorium

Penelitian laboratorium dilakukan untuk merakit alat, pengujian alat dan menganalisis alat yang telah dibuat.

Berikut ini adalah langkah-langkah penelitian :



Gambar 1.1. Diagram alir langkah-langkah penelitian

1.5 Sistematika Penulisan

Sistematika penulisan skripsi terdiri atas lima bab yang secara garis besar dapat diuraikan sebagai berikut :

1. Bab 1 Pendahuluan

Bab ini memuat tentang latar belakang dari penelitian, tujuan, metode yang digunakan, dan juga pembatasan masalah pada penelitian yang dilakukan.

2. Bab 2 Landasan Teori

Bab ini memuat secara garis besar teori dasar yang berhubungan dengan penelitian.

3. Bab 3 Perancangan Alat

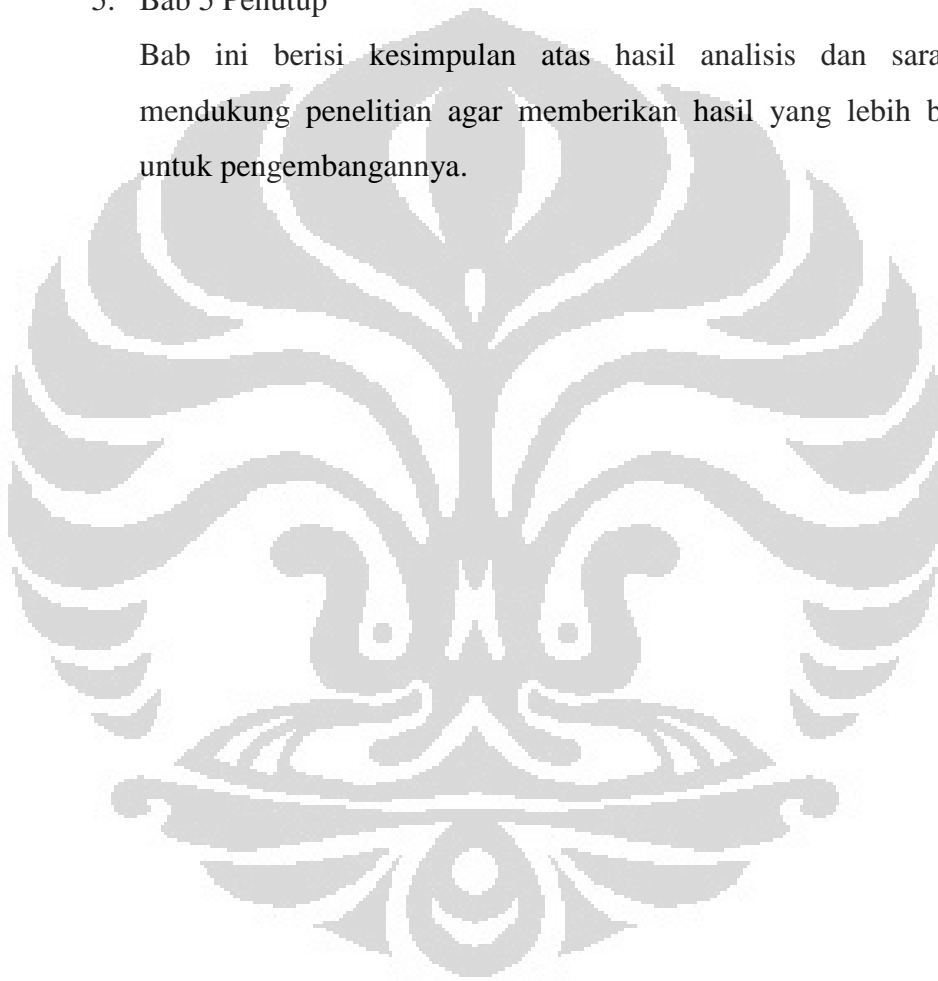
Bab ini memuat perancangan perangkat keras dan perangkat lunak

4. Bab 4 Hasil dan Pembahasan

Bab ini berisi penjelasan mengenai hasil penelitian dan analisis dari sistem yang telah dibuat.

5. Bab 5 Penutup

Bab ini berisi kesimpulan atas hasil analisis dan saran yang mendukung penelitian agar memberikan hasil yang lebih baik lagi untuk pengembangannya.



BAB 2

LANDASAN TEORI

2.1. Fantom Air pada Radioterapi

Radioterapi adalah jenis terapi yang menggunakan radiasi pengion untuk menghancurkan sel-sel kanker. Terapi kanker dengan radiasi dimungkinkan karena sel-sel kanker secara umum mempunyai kepekaan yang lebih tinggi terhadap radiasi pengion dibandingkan sel-sel normal. Dengan pengaturan pemberian dosis radiasi secara tepat, radiasi tersebut dapat membunuh sel-sel kanker dengan sedikit efek kerusakan terhadap sel-sel normal di sekitarnya^[2]. Radiasi pengion jenis foton yang dapat dimanfaatkan untuk radioterapi kanker adalah sinar- γ dan sinar-X. Kedua jenis radiasi tersebut mempunyai daya tembus yang tinggi terhadap organ tubuh dengan kemampuan tembusnya ditentukan oleh besar energi yang dimilikinya. Pesawat teleterapi Co-60 dan pesawat pemercepat linier (linac) dapat menghasilkan kedua jenis radiasi tersebut, yang selanjutnya disebut pesawat radioterapi eksternal.

Setelah pesawat radioterapi eksternal diterima, maka sebelum pesawat tersebut digunakan untuk penyinaran, fisikawan terlebih dahulu harus memperoleh data pengukuran radiasi melalui commissioning yang merupakan karakteristik kemampuan pesawat radioterapi eksternal tersebut. Data-data tersebut nantinya yang akan digunakan dalam perencanaan penyinaran terhadap pasien. Selain itu pesawat radioterapi eksternal juga harus dilakukan kalibrasi untuk memastikan dosis radiasi yang diberikan pasien sudah tepat.

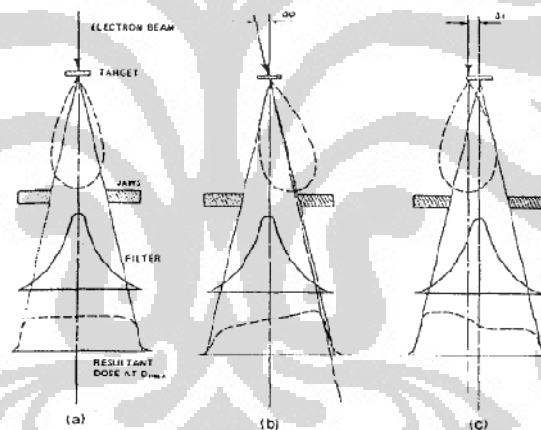


Gambar 2.1 Fantom Air

Commissioning dan kalibrasi pesawat radioterapi eksternal menggunakan alat ukur/detector yang dilengkapi dengan electrometer serta fantom air. Untuk commissioning, fantom air yang digunakan harus memiliki kemampuan pemindai 3D dengan ukuran yang cukup besar ($40 \times 40 \times 40 \text{ cm}^2$). Sedangkan untuk kalibrasi cukup membutuhkan fantom yang lebih kecil serta memiliki holder detektor pada kedalaman tertentu di dalam air.

2.2. Profil Berkas Radiasi

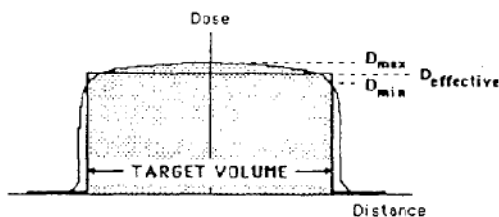
Perjalanan elektron, baik yang dimanfaatkan secara langsung untuk terapi pasien ataupun untuk menghasilkan sinar-x, sangat penting untuk mendapatkan suatu berkas elektron atau radiasi sinar-x yang homogen. Sebagai gambaran perjalanan elektron yang digunakan untuk menghasilkan sinar-x tersebut akan menghasilkan profil berkas seperti pada Gambar 2.2.



Gambar 2. 2. Perubahan profil berkas sinar-x akibat perjalanan elektron yang berbeda-beda^[3]

Kesimetrisan dan kerataan berkas dipengaruhi oleh sistem perjalanan berkas elektron. Ketidakteraturan akan meningkat ketika berkas elektron tidak secara simetris menabrak filter pemerata disamping merupakan sifat alami interaksi elektron berkecepatan tinggi dengan bahan sasaran untuk radiasi sinar-x.

Efek ketidaksimetrian dan ketidakteraturan berkas akan sangat mempengaruhi keberhasilan pengobatan kanker dengan radiasi, karena disamping akan memberikan distribusi dosis yang besar pada jaringan sehat, dan juga distribusi dosis yang sampai ke sel kanker akan tidak efektif. Hal ini dapat digambarkan dalam hubungannya dengan 'target volume' seperti pada Gambar 2.3.

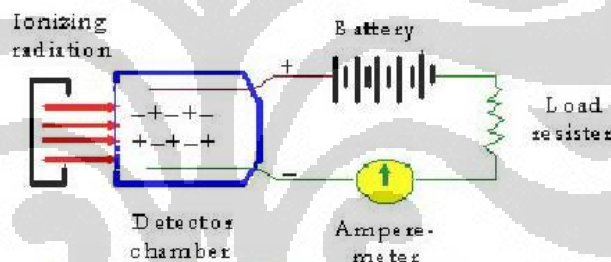


Gambar 2. 3. Dosis efektif yang diharapkan terhadap 'profil' berkas yang dihasilkan^[4]

2.3. Detektor Bilik Ionisasi

Bilik ionisasi (*ionization chamber*) yang banyak digunakan saat ini adalah yang menggunakan udara bebas sebagai gasnya. Ketika partikel radiasi ditembakkan ke dalam tabung (*chamber*) ionisasi, misalkan partikel β , maka partikel tersebut akan mengionisasi gas yang terdapat dalam tabung. Proses tersebut akan menghasilkan ion positif dan ion negatif seperti pada gambar 2.4.

Key Components in a Simple Ionization Chamber



Gambar 2.4. Rangkaian sederhana detector ionization chamber

Dengan beda potensial tertentu maka ion (-) akan tertarik ke anoda (+) dan ion (+) akan tertarik ke katoda (-). Ion (+) bergerak lebih lambat karena lebih massif dari ion (-) atau elektron^[5]. Ionization chamber beroperasi pada beda potensial yang lebih rendah jika dibandingkan dengan proportional counter dan geiger counter. Pada gambar 2.4, terlihat bahwa jika tegangan yang terpasang tidak mencukupi angka tertentu, maka ion yang terbentuk karena radiasi akan menggabungkan diri lagi menjadi partikel gas. Jika tegangan yang diberikan terlalu rendah, maka beberapa elektron dan ion (+) akan bergabung kembali (*recombine*) sebelum mencapai elektroda sehingga ion kembali menjadi molekul tak bermuatan. Dengan potensial tertentu maka akan terdeteksi arus dengan menyimpangnya jarum ampermeter. Arus yang terdeteksi biasanya sangat kecil, sekitar beberapa microampere, namun masih dapat terdeteksi. Sebuah arus listrik

adalah sebuah aliran elektron pada kawat dalam sebuah rangkaian sederhana. Elektron secara terus menerus berputar-putar dalam kawat rangkaian. Ketika elektron meninggalkan satu bagian kawat maka akan segera digantikan oleh elektron selanjutnya. Sebenarnya pada ionization chamber tidak terdapat ion atau elektron. Namun proses radiasi dari sumberlah yang menyebabkan timbulnya ion tersebut dan tertarik ke elektroda sehingga dapat terdeteksi oleh amperemeter. Sumber-sumber yang sangat radiokatif dapat menggantikan ion secara cepat sehingga menghasilkan arus yang besar.

Jenis detektor ionization chamber yang dipakai pada penelitian ini adalah farmer type chamber $0,6 \text{ cm}^3$, type 30013 (gambar 2.5).



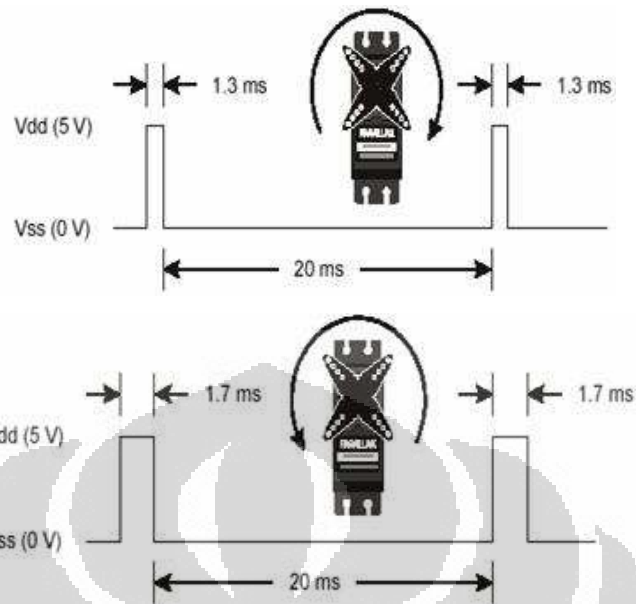
Gambar 2.5. Detektor ionization chamber

2.4. Servo motor continuous

Servo motor adalah sebuah motor dengan sistem *closed feedback* di mana posisi dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalamnya. Motor ini terdiri dari sebuah motor, serangkaian gear, potensiometer dan rangkaian kontrol. Servo motor terdiri dari dua tipe, yaitu tipe standard dan tipe continuous^[6]. Pada penelitian ini, kami menggunakan servo motor tipe continuous.

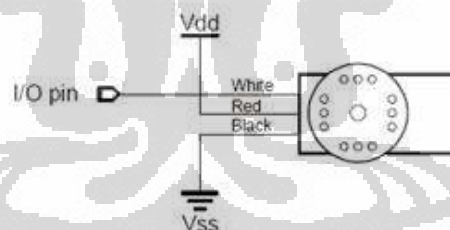
Servo motor continuous dapat diatur arah gerakannya, namun tidak dapat berhenti pada satu sudut/posisi dikarenakan sistem *closed feedback* yang ditiadakan. Untuk menggerakkan servo motor ke kanan atau ke kiri, tergantung dari nilai delay yang kita berikan. Seperti pada gambar 2.6, untuk membuat servo pada posisi center, berikan pulsa 1.5ms. Untuk memutar servo ke kanan, berikan pulsa $\leq 1.3\text{ms}$, dan pulsa $\geq 1.7\text{ms}$ untuk berputar ke kiri dengan periode 20ms,

seperti ilustrasi berikut:



Gambar 2.6. Teknik pengaturan lebar pulsa untuk mengatur arah servo motor continuous

Pada servo motor, terdapat tiga buah kabel yang berfungsi sebagai sumber tegangan, ground dan pin I/O yang berfungsi untuk menerima data pulsa mikrokontroler.

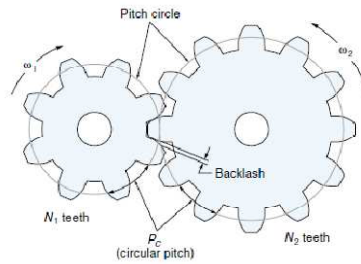


Gambar 2.7. Susunan kabel servo motor

2.5. Spur Gear

Ketika dua buah gear dengan diameter berbeda saling bertautan, mereka akan berotasi dengan kecepatan yang berbeda. Gerakannya mirip dengan dua buah roda dengan diameter yang berbeda berputar terhadap satu sama lain, dimana ukuran masing-masing roda dinyatakan dalam ukuran lingkaran teoritis yang disebut *pitch circle* (gambar 2.8). Diameter dari *pitch circle* disebut *pitch diameter*. *Pitch diameter* lebih kecil dari keseluruhan diameter gear dikarenakan

setiap gigi saling melengkapi pada saat bertautan. Jarak antara pitch circle pada satu gigi dan satu lembah disebut *circular pitch*.



Gambar 2.8. Spur gear

Jumlah gigi pada gear dapat dihitung dengan :

$$N = \frac{\text{circumference}}{\text{distance between teeth}} = \frac{\pi D}{P_c}$$

where
 N = total number of teeth
 D = pitch diameter
 P_c = circular pitch

Pada penelitian ini, spur gear digunakan untuk mentransfer daya dari servo motor continuous ke batang ulir penggerak horizontal dan vertikal. Karena daya (power) adalah torsi \times kecepatan putar, persamaan di atas dapat ditulis kembali sebagai berikut:

$$T_1 \omega_1 = T_2 \omega_2$$

$$\frac{T_2}{T_1} = \frac{\omega_1}{\omega_2} = N_g$$

Dimana:

T = Torsi

ω = Kecepatan putar^[7]

Persamaan ini berarti semakin cepat gear berputar, maka torsi yang dihasilkan semakin berkurang dan gear yang berputar lebih lambat akan menghasilkan torsi yang lebih besar.

2.6. Sensor Rotary Encoder

Rotary encoder atau disebut juga shaft encoder berfungsi sebagai penghitung putaran suatu poros. Umumnya digunakan untuk menghitung putaran

motor. Salah satu bentuk fisik shaft encoder jenis hollow ditunjukkan pada gambar 2.9.



Gambar 2. 9. Rotary Encoder jenis Hollow

Encoder biasanya memiliki 2 keluaran, channel A dan channel B. Bentuk pulsa channel A dan B akan berbeda agar kita bisa membedakan arah putaran, apakah CW(clockwise) atau CCW(counter clockwise). Pada Gambar 2.10 (a) ditunjukkan bentuk pulsa untuk arah putaran CW dan pada gambar 2.10 (b) ditunjukkan bentuk pulsa untuk arah putaran CCW.



Gambar 2.10 . (a) Contoh output kedua channel dengan searah jarum jam (CW)



Gambar 2.10. (b) Contoh output kedua channel dengan berlawanan arah jarum jam (CCW)

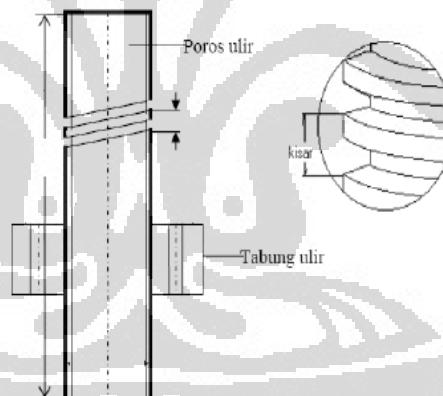
Didalam rotary encoder terdapat sebuah LED dan 2 buah photodioda sebagai detektor^[8]. Detektor tersebut diletakkan sedemikian rupa agar dapat menghasilkan pola pulsa yang berbeda untuk masing – masing arah putaran. Pada bagian dalam shaft encoder terdapat rangkaian elektronik untuk menghasilkan pulsa. Adapun spesifikasinya adalah sebagai berikut :

- Model : E40HB8-500-3-T-24
- Power supply : DC 12-24V \pm 5%

- Black : Out A
- White : Out B
- Orange : Out Z
- Brown : +V
- Blue : -V
- Shield : F, G
- Autonics, Korea^[9]

2.7. Batang ulir

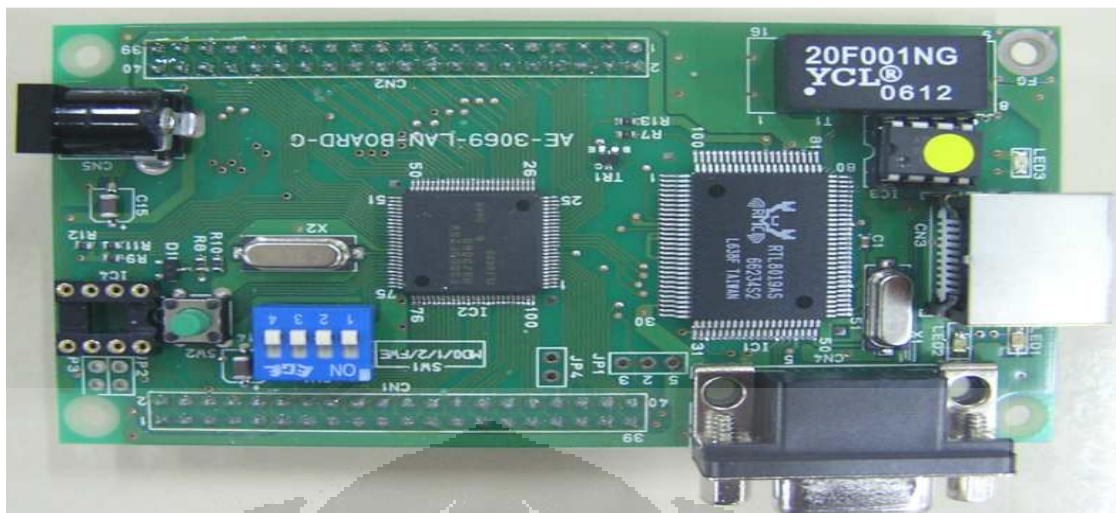
Batang ulir atau drat merupakan baja beton prategang dengan bentuk khusus yang permukaannya memiliki sirip melintang dan rusuk memanjang yang dimaksudkan untuk meningkatkan daya lekat dan guna menahan gerakan membujur dari batang secara relatif terhadap beton^[10].



Gambar 2.11. Batang ulir

2.8. Mikrokontroler H8/3069F

Mikrokontroler yang digunakan pada penelitian ini adalah seri *H8/300H* yaitu *single-chip microcomputer* berperforma tinggi yang mempunyai sebuah CPU 16-bit (*Control Processing Unit*) sebagai prosesornya. Gambar 2.12 menunjukkan *minimum system* mikrokontroler H8/3069F^[11].



Gambar 2.12. Minimum sistem mikrokontroler H8/3069F

H8/3069F adalah salah satu mikrokontroler yang dilengkapi *processor H8/300H*. *H8/3069F* mempunyai sebuah *internal writable flash memory* yang menggunakan *single power supply* (5V). Mikrokontroler ini dapat difungsikan sebagai *single-chip* mikrokomputer karena didalamnya sudah tersedia ROM, RAM dan fungsi I/O. Mikrokontroler ini juga dapat difungsikan sebagai *multi-chip* mikrokomputer saat terjadi penambahan memori ^[11].

2.8.1. I/O Port

Tersedia 11 I/O port, yaitu port 1 sampai port 9, port A dan port B. Masing-masing port memiliki fungsi yang sama yaitu sebagai register, *port data register* (PDR) dan *port data direct register* (PDDR). Beberapa port juga bekerja sesuai *mode* yang ada dan beberapa port memiliki kelebihan sebagai input *Pull-Up MOS Control Register* (PCDR). I/O port dapat digunakan sebagai input status *on/off* atau sinyal dari berbagai sensor ^[11].

2.8.2. Interrupt Controller

Interrupt adalah suatu kejadian atau peristiwa yang menyebabkan mikrokontroler berhenti sejenak untuk melayani *interrupt* tersebut. Program yang dijalankan pada saat melayani *interrupt* disebut ***Interrupt Service Routine*** ^[11].

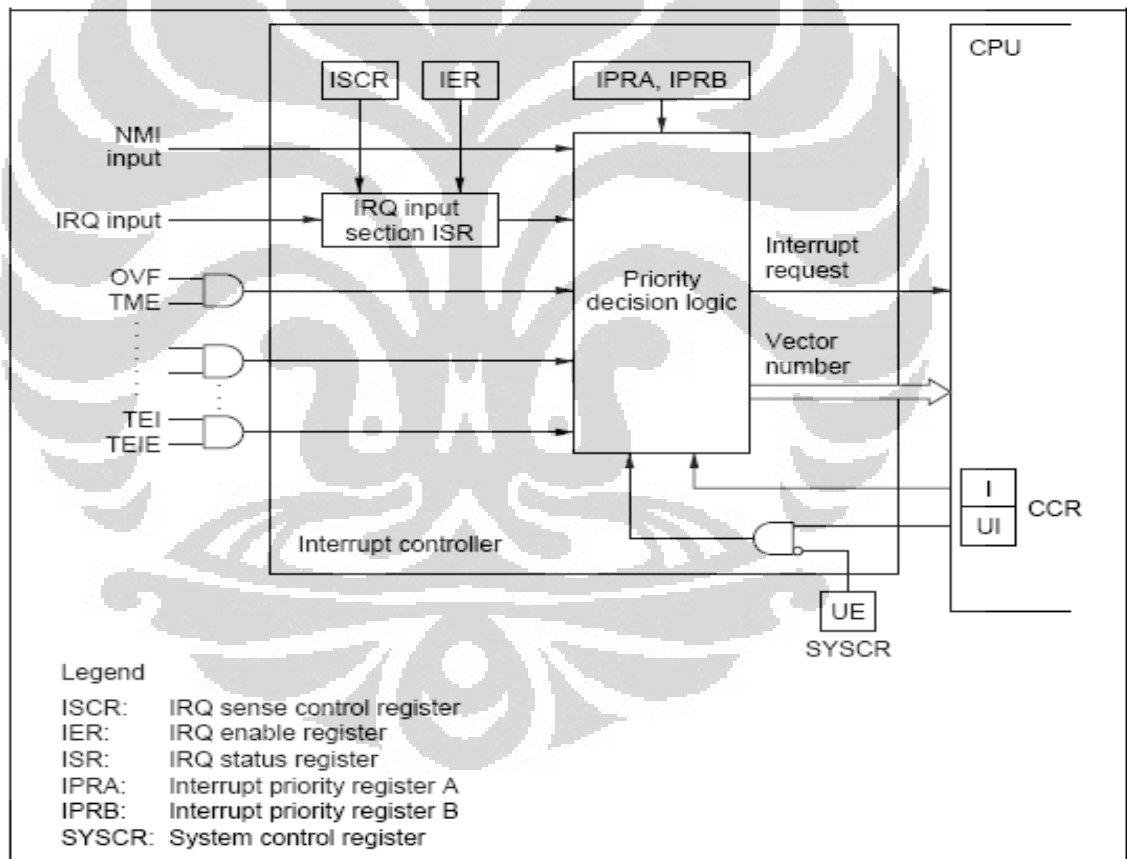
Mikrokontroler H8/3069F memiliki 7 eksternal *interrupt* dengan level prioritas yang dapat diatur dengan menggunakan IPRs (*Interrupt Priority*

Register). NMI (*Nonmaskable interrupt*) adalah prioritas tertinggi dari *interrupt* yang dapat menerima *rising edge* maupun *falling edge*. Sedangkan IRQ0-IRQ5 pemilihan level dapat dipilih secara terpisah. NMI dapat di-*disable* pada kondisi tertentu ^[11].

Tabel. 2.1. Pin-pin *interrupt*

Name	Abbreviation	I/O	Function
Nonmaskable interrupt	NMI	Input	Nonmaskable interrupt*, rising edge or falling edge selectable
External interrupt request 5 to 0	IRQ ₅ to IRQ ₀	Input	Maskable interrupts, falling edge or level sensing selectable

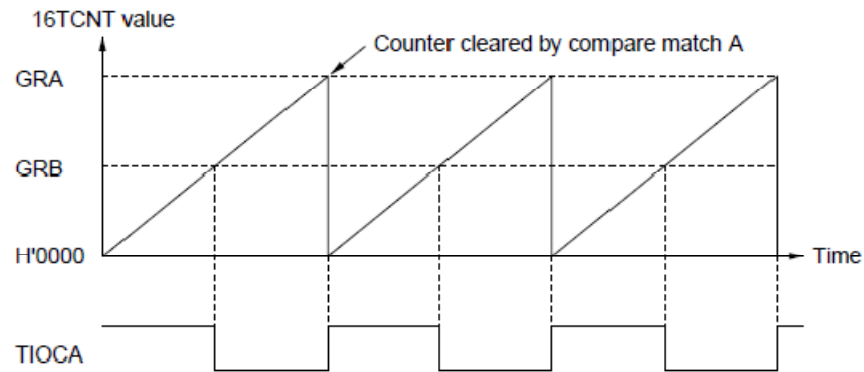
Note: * NMI input is sometimes disabled. For details see 18.9, NMI Input Disabling Conditions.

Gambar 2.13. Blok diagram *interrupt* ^[10]

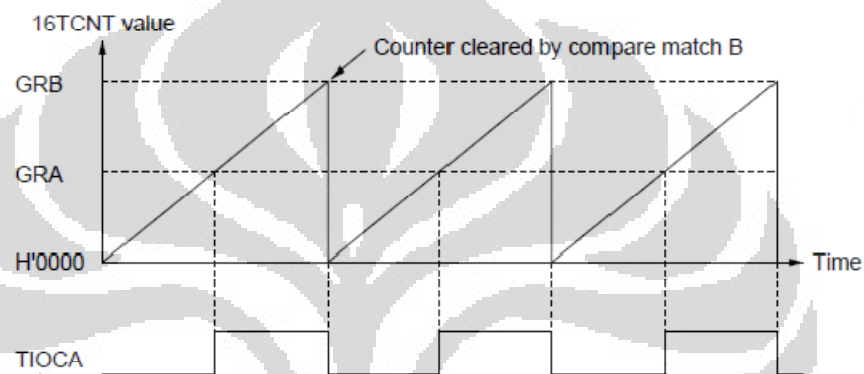
2.8.3. PWM Mode

Untuk menjalankan servo motor tipe continuous, diperlukan teknik PWM (Pulse Width Modulation). Teknik PWM yang diterapkan pada mikrokontroler H8/3069F memanfaatkan 16-bit timer yang terintegrasi di dalamnya^[11]. Beberapa fitur dari timer ini adalah :

1. Dapat memproses hingga 6 pulsa output dan 6 pulsa input
2. Memiliki enam general register (GRA dan GRB di tiap kanal) yang dapat berfungsi sebagai output compare dan input capture
3. Terdapat delapan sumber clock untuk tiap kanal.
Internal clock : Φ , $\Phi/2$, $\Phi/4$, $\Phi/8$
Eksternal clock : TCLKA, TCLKB, TCLKC, TCLKD
4. Lima jenis mode operasi pada tiap kanalnya : *waveform output by compare match*, *input capture function*, *counter clearing function*, *synchronization*, dan *PWM mode*
5. *Phase counting mode* pada kanal 2
6. Akses kecepatan tinggi via bus internal 16 bit
7. Memiliki sembilan sumber *interrupt*



a. Counter cleared by GRA (TOA = 1)



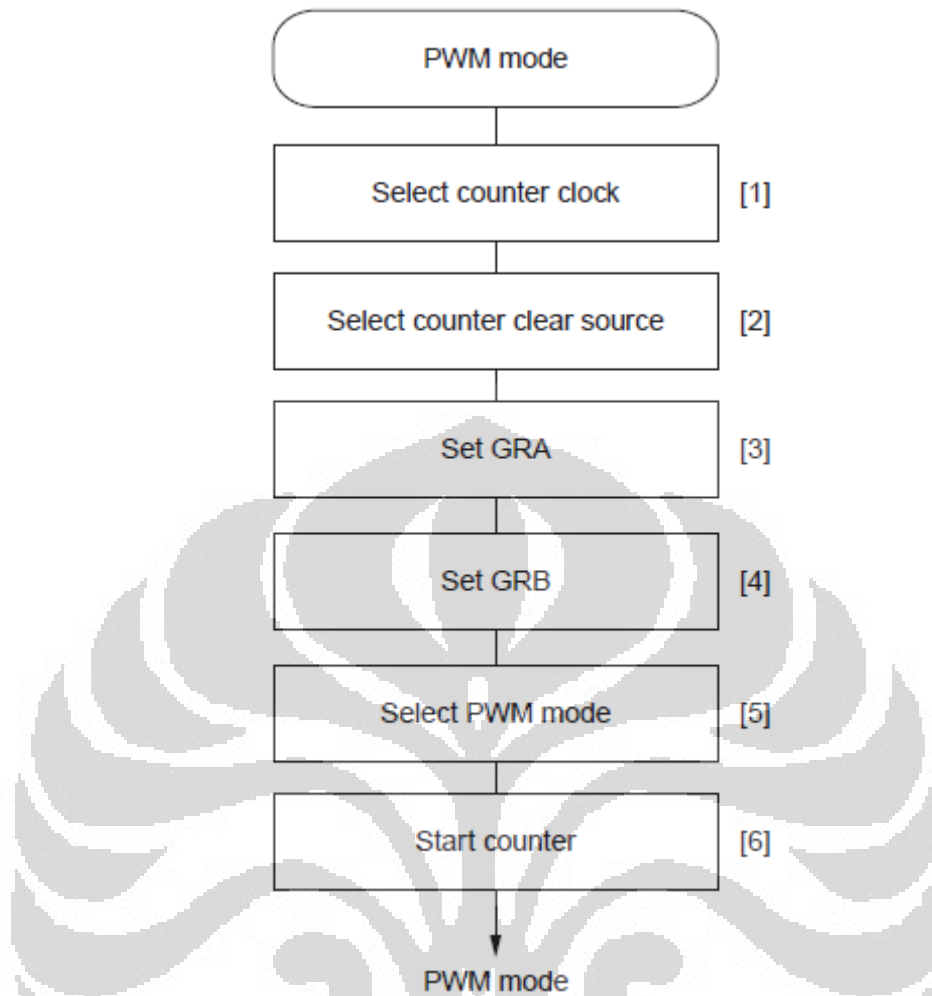
b. Counter cleared by GRB (TOA = 0)

Gambar 2.14. a) Mode PWM dengan counter di-clear-kan oleh GRA

b) Mode PWM dengan counter di-clear-kan oleh GRB

Pada mikrokontroler H8/3069, GRA dan GRB dipasangkan dan gelombang PWM merupakan output dari pin TIOCA. GRA digunakan sebagai spesifikasi waktu agar output PWM berubah menjadi 1, sementara GRB digunakan sebagai spesifikasi waktu agar output PWM berubah menjadi 0 atau sebaliknya, seperti yang dideskripsikan pada gambar 2.14. Perbandingan antara GRA dan GRB disebut *duty cycle*. Apabila GRA atau GRB digunakan sebagai *counter clear source*, maka gelombang PWM dengan *duty cycle* dari 0% sampai 100% dihasilkan oleh pin TIOCA.

Berikut adalah prosedur untuk menggunakan PWM mode pada mikrokontroler H8/3069F:



Gambar 2.15. Prosedur penggunaan PWM mode pada mikrokontroler H8/3069F

[1] Set bit TPSC2 hingga TPSC0 pada register 16TCR untuk memilih sumber clock (counter clock source).

[2] Set bit CCLR1 dan CCLR0 pada register 16TCR untuk memilih counter clear source

Bit	7	6	5	4	3	2	1	0
	—	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
Initial value	1	0	0	0	0	0	0	0
Read/Write	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Gambar 2.16. Bit-bit pada register 16TCR

Dengan fungsi dari tiap bit tersebut adalah sebagai berikut :

Tabel 2.2. Fungsi bit TPSC pada register 16TCR

Timer prescaler 2 to 0

Bit 2	Bit 1	Bit 0	TCNT Clock Source
TPSC2	TPSC1	TPSC0	
0	0	0	Internal clock : ϕ (Initial value)
		1	Internal clock : $\phi / 2$
	1	0	Internal clock : $\phi / 4$
		1	Internal clock : $\phi / 8$
1	0	0	External clock A : TCLKA input
		1	External clock B : TCLKB input
	1	0	External clock C : TCLKC input
		1	External clock D : TCLKD input

Tabel 2.3. Fungsi bit CCLR pada register 16TCR

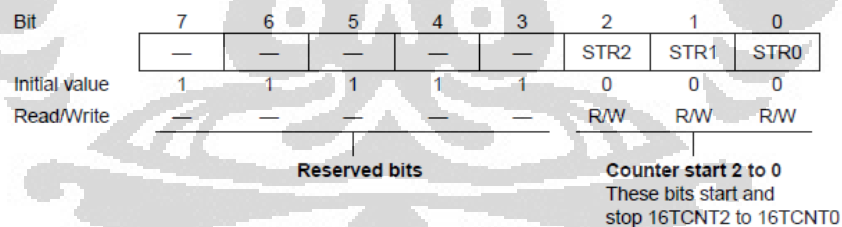
Counter clear 1 and 0

Bit 6	Bit 5	TCNT clear Sources
CCLR1	CCLR0	
0	0	TCNT is not cleared (Initial value)
	1	TCNT is cleared by GRA compare match or input capture
1	0	TCNT is cleared by GRB compare match or input capture
	1	Synchronous clear : TCNT is cleared in synchronization with other synchronized timers

[3] Atur waktu yang diinginkan agar output PWM menjadi 1 melalui GRA

[4] Atur waktu yang diinginkan agar output PWM menjadi 0 melalui GRB

[5] Set STR bit menjadi 1 pada register TSTR untuk memulai counter



Gambar 2.17. Bit-bit pada register TSTR

Dengan fungsi tiap bit sebagai berikut :

Tabel 2.4. Fungsi bit STR pada register TSTR

Bit 0	Description
STR0	
0	16TCNT0 is halted (Initial value)
1	16TCNT0 is counting

2.10. Python

Python merupakan bahasa pemrograman yang bersifat *object-oriented*. Meskipun *Python* tidak terlalu terkenal dibandingkan dengan bahasa pemrograman lainnya, *Python* dapat diandalkan untuk membuat *software* aplikasi di bidang sains dan teknik dengan efisien dan elegan.

Python memiliki kelebihan lain yang sangat penting dibanding bahasa pemrograman yang terdahulu:

- a. *Python* merupakan *open-source software*, yang artinya ia dapat diperoleh secara gratis. Bahkan *Python* sudah otomatis terinstall di Linux.
- b. *Python* tersedia pada semua *operating systems* (OS) terkenal seperti Linux, Unix, Windows, dan MacOS. Suatu *script Python* yang ditulis pada OS tertentu, dapat dijalankan di OS lain tanpa ada modifikasi sedikitpun.
- c. *Python* lebih mudah dipelajari sekaligus lebih mudah dibaca dibandingkan dengan bahasa pemrograman lainnya.
- d. *Python* dan program ekstensinya mudah diinstall.

Python berdiri di atas landasan pondasi *Java* and *C++*. Hal-hal seperti *classes*, *methods*, *inheritance*, yang kerap kali diimplementasikan pada bahasa yang bersifat *object-oriented*, juga dapat diimplementasikan di *Python*. *Python* merupakan salah satu bahasa pemrograman tingkat tinggi. Sintaks inti *Python* dan strukturnya merupakan kode yang mudah dibaca dengan *library* yang luas dan dapat disesuaikan penggunaanya^[12].

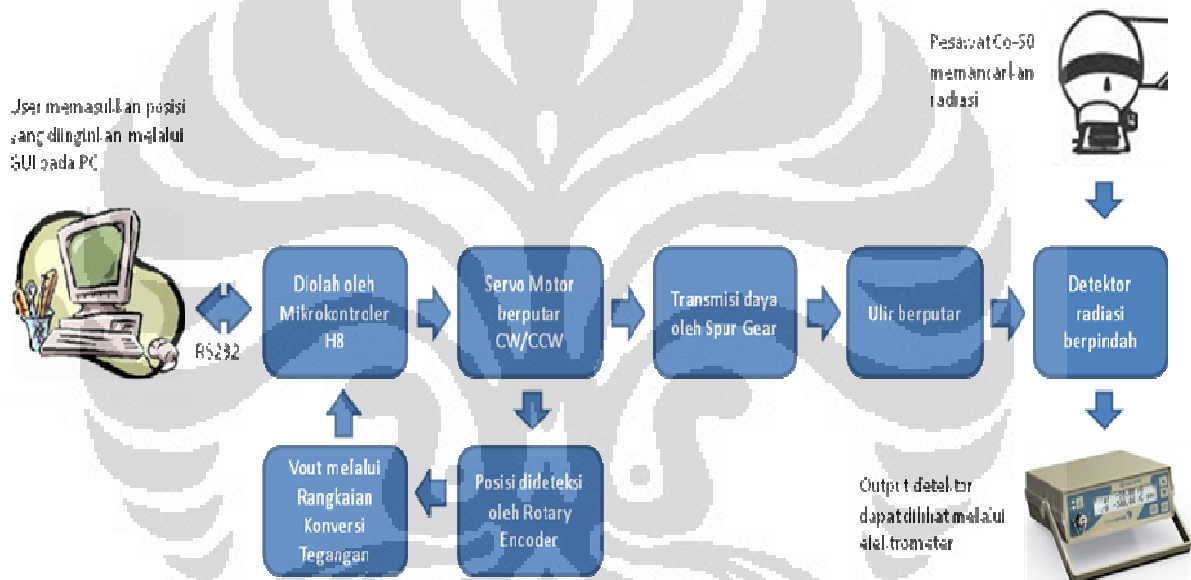
BAB 3

PERANCANGAN ALAT

Pada bab ini akan dijelaskan mengenai skema atau rancangan alat yang digunakan dalam penelitian ini. Apabila dikelompokkan, terdapat 3 sub bab utama yang akan dibahas :

3.1.Desain Sistem

Rancangan sistem penggerak detektor dua dimensi ini terdiri dari PC, mikrokontroler, servo motor, gear, ulir, sensor dan rangkaian konversi tegangan yang terintegrasi. Apabila digambarkan skemanya adalah seperti dibawah :



Gambar 3.1. Blok diagram umum perangkat keras sistem pengendali posisi detektor

User dapat menggunakan *Graphical User Interface* (GUI) yang sudah terinstal pada PC untuk mengatur posisi horizontal dan vertikal detektor radiasi pada fantom air. Posisi ini kemudian diubah menjadi pulsa acuan dengan menggunakan persamaan yang didapat dari proses kalibrasi. Data ini kemudian akan dibaca oleh mikrokontroler dengan memanfaatkan komunikasi RS232. Mikrokontroler sebelumnya sudah diprogram untuk membaca data tersebut dengan menggunakan software Bahasa C. Selain diprogram untuk membaca data dari GUI, mikrokontroler juga diprogram untuk menggerakkan servo motor

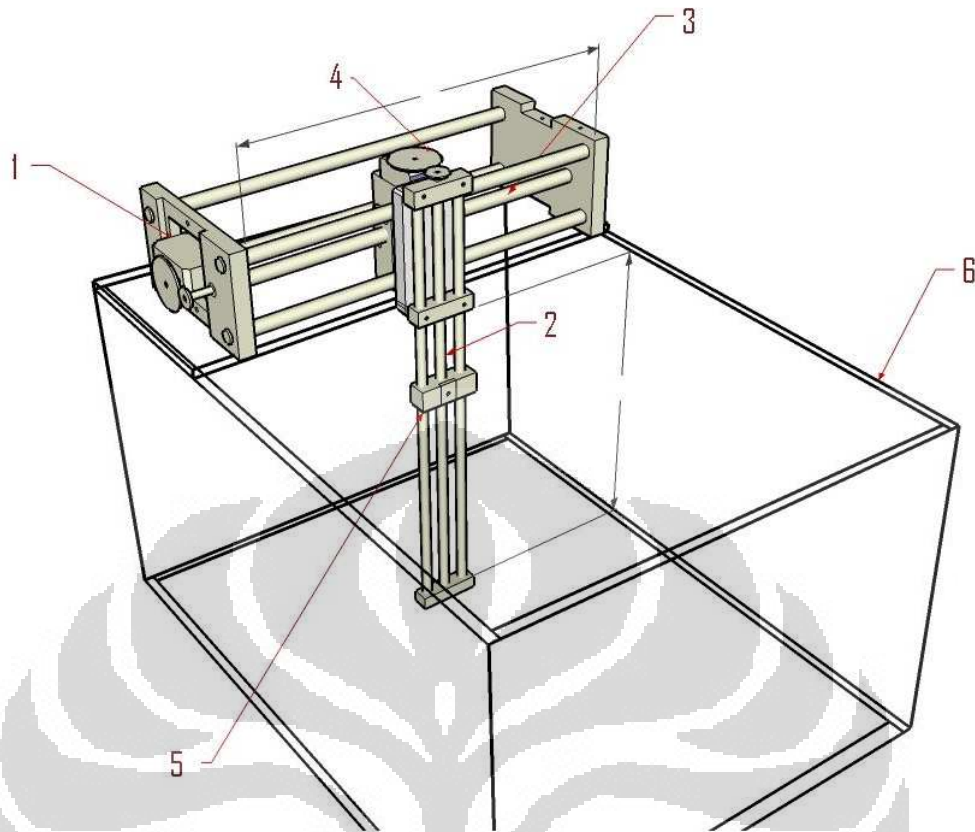
continuous ke arah *clockwise*, *counterclockwise* dan berhenti sesuai dengan letak posisi tujuan.

Setelah data terkirim, maka motor akan berputar dan mentransmisikan daya ke batang ulir melalui spur gear. Motor akan terus berputar sampai jumlah pulsa yang dihasilkan oleh sensor rotary encoder yang dipasang satu sumbu dengan motor sama dengan jumlah pulsa acuan. Setelah itu, pulsa aktual dikonversi kembali menjadi posisi aktual dengan membalikkan persamaan yang didapat pada proses kalibrasi dan ditampilkan pada GUI.

3.2.Sistem Mekanik

Penelitian ini dimulai dengan melakukan rancangan fantom dan mekanik pergerakan detektor dua dimensi. Wadah fantom air terbuat dari akuarium berbahan akrilik dengan ketebalan 1,5 cm dengan volume wadah 40 x 40 x 40. Salah satu jenis fantom air dua dimensi dapat dilihat pada gambar 2.1.

Fantom air yang dibuat pada penelitian ini memakai sistem penggerak berupa batang ulir (drat) yang dilengkapi dengan tiang di kanan dan kiri untuk menjaga kestabilan detektor holder. Ulir penggerak detektor yang terbuat dari bahan yang tahan terhadap air, yaitu *stainless steel*. Pada penelitian ini, ukuran ulir yang dipakai adalah M1 jenis metrik. Artinya, tiap satu putaran penuh dari batang ulir, maka perpindahan posisi detektor adalah sejauh 1 mm.



Gambar 3.2. Rancangan sistem mekanik penggerak detektor

Keterangan gambar :

1. Servo motor Continuous
2. Poros ulir vertikal
3. Poros ulir horizontal
4. Spur gear
5. Detektor holder
6. Akuarium akrilik

Pada bagian atas batang ulir sumbu vertikal dan bagian kanan batang ulir horizontal diberikan pemutar mekanik berupa spur gear yang berfungsi untuk mentransmisikan daya dari servo motor continuous sehingga ulir (drat) dapat berputar dan menggerakkan detektor secara horizontal ataupun vertikal sejauh 30 cm. Gear dengan diameter lebih besar dipasang pada servo motor, sementara gear dengan ukuran yang lebih kecil dipasang pada ulir (gambar 3.2). Jumlah gigi (N) pada gear besar sebanyak 54 dan pada gear kecil sebanyak 24 dengan ukuran gigi (*Diametral Pitch*) sebesar 32.

Pada masing-masing servo motor continuous dikopel sebuah sensor putaran berupa rotary encoder yang dipasang satu sumbu. Output dari sensor yang sudah berupa pulsa ini kemudian dihubungkan dengan rangkaian pengkonversi tegangan berupa *transistor switching* untuk kemudian diolah oleh mikrokontroler. Sensor rotary encoder, servo motor dan spur gear disusun sehingga menjadi seperti gambar di bawah ini.



Gambar 3.3. Susunan motor, spur gear dan sensor

3.3.Pemasangan Servo motor Continuous

Servo motor continuous memiliki tiga buah kabel yang masing-masing berfungsi sebagai sumber tegangan (V_{cc}), ground dan data. Sumber tegangan yang dibutuhkan oleh servo motor continuous adalah 5 v sehingga cukup dengan menggunakan sumber tegangan dari mikrokontroler, begitupun dengan ground.

Teknik menjalankan servo motor continuous adalah dengan memberikan sejumlah pulsa yang sama secara kontinu dan lebar pulsa yang disesuaikan dengan teori yang ada. Pada penelitian digunakan metode *pulse width modulation* (PWM) untuk menjalankannya. Dengan memakai metode ini, kabel data servo motor dihubungkan dengan pin TIOCA yang merupakan bagian dari port Timer pada miktokontroller H8 untuk memberikan informasi arah gerak motor.

Untuk servo motor pada sumbu vertikal dihubungkan dengan pin TIOCA₀ sementara untuk horizontal dihubungkan dengan pin TIOCA₁. Pada tabel 3.1 dijelaskan konfigurasi dan fungsi pin yang dipakai di penelitian ini. Untuk

penjelasan mengenai konfigurasi pin mikrokontroler H8/3069 F secara umum dapat dilihat pada lampiran.

Tabel 3.1. Fungsi pin pada mikrokontroler H8/3069 F

No	Nama Pin	Fungsi
1.	TIOCA ₀	PWM untuk motor pada sumbu vertikal
2.	TIOCA ₁	PWM untuk motor pada sumbu horizontal
3.	IRQ ₀	Pin interrupt untuk output A rotary encoder pada sumbu vertikal
4.	IRQ ₁	Pin interrupt untuk output A rotary encoder pada sumbu horizontal
5.	P1DR.PIN.B ₅	Pin untuk mendeteksi keadaan pulsa output B rotary encoder pada sumbu vertikal
6.	P1DR.PIN.B ₄	Pin untuk mendeteksi keadaan pulsa output B rotary encoder pada sumbu horizontal

Sementara untuk pengiriman data dari PC ke mikrokontroler, digunakan komunikasi serial RS232. Mikrokontroler H8/3069 F memiliki tiga buah channel SCI (*Serial Communication Interface*) yang digunakan untuk komunikasi serial. Konfigurasi pin dan fungsinya dapat dilihat pada tabel 3.2.

Tabel 3.2. Konfigurasi dan fungsi pin untuk komunikasi serial pada mikrokontroler H8/3069 F

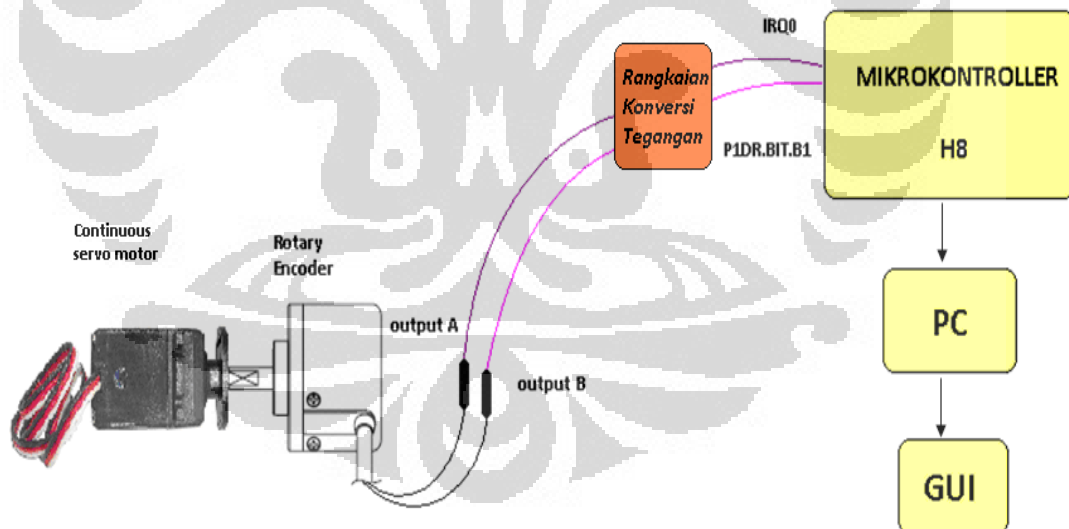
Channel	Name	Abbreviation	I/O	Function
0	Serial clock pin	SCK ₀	Input/output	SCI ₀ clock input/output
	Receive data pin	RxD ₀	Input	SCI ₀ receive data input
	Transmit data pin	TxD ₀	Output	SCI ₀ transmit data output
1	Serial clock pin	SCK ₁	Input/output	SCI ₁ clock input/output
	Receive data pin	RxD ₁	Input	SCI ₁ receive data input
	Transmit data pin	TxD ₁	Output	SCI ₁ transmit data output
2	Serial clock pin	SCK ₂	Input/output	SCI ₂ clock input/output
	Receive data pin	RxD ₂	Input	SCI ₂ receive data input
	Transmit data pin	TxD ₂	Output	SCI ₂ transmit data output

Tiap kanal SCI memiliki tiga buah pin, yaitu SCK, RxD dan TxD. Pin SCK digunakan sebagai clock input/output. Pin RxD digunakan sebagai input penerimaan data SCI. Pin TxD digunakan sebagai pengiriman data SCI.

3.4. Sensor dan Rangkaian Pengkonversi Tegangan

Pemantauan posisi detektor dilakukan dengan menggunakan sensor perpindahan sudut berupa rotary encoder atau shaft encoder. Rotary encoder dipasang satu sumbu dengan servo motor continuous yang mentransmisikan daya ke poros ulir melalui pemasangan spur gear. Oleh karena itu, akan didapat korelasi antara putaran rotary terhadap ketinggian berupa persamaan linear. Persamaan tersebut secara umum dinyatakan dengan $y = ax + b$, Dimana y adalah jumlah pulsa, x adalah perubahan posisi, a dan b adalah koefisien dan konstanta yang didapat dari metode *least square* (kuadrat terkecil).

Data pulsa ini akan dipakai pada program utama dimana detektor akan berhenti pada posisi yang diinginkan berdasarkan jumlah pulsa yang dihasilkan oleh rotary encoder.



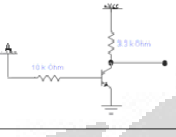
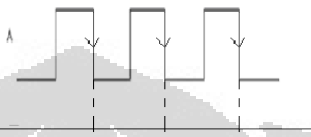
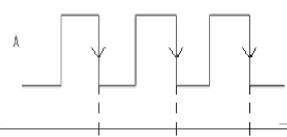
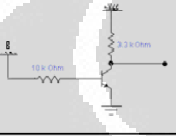
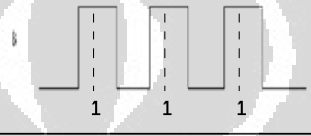
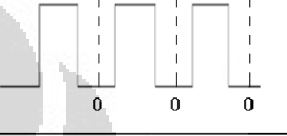
Gambar 3.4. Sistem pembacaan putaran motor dengan sensor rotary encoder

Rotary encoder yang dipakai pada penelitian ini adalah jenis hollow dengan model E40H8-500-3-T-24 buatan Autonics.

Output dari sensor rotary encoder terbagi menjadi dua, yaitu output A dan output B. Output A pada masing-masing rotary encoder digunakan sebagai *interrupt* yang dihubungkan ke pin interrupt (IRQ) pada mikrokontroler dan

output B berfungsi sebagai pendeteksi yang dihubungkan ke pin I/O port (P1DR.BIT.Bn) setelah melewati rangkaian pengkonversi tegangan. Untuk ulir sumbu vertikal, output A dan B masing-masing dihubungkan pada pin IRQ₀ dan P1DR.BIT.B₅. Sementara untuk ulir sumbu horizontal, output A dan B masing-masing dihubungkan pada pin IRQ₁ dan P1DR.BIT.B₄.

Tabel 3.3. Output rotary encoder

	Rangkaian konversi tegangan	counter clock wise	clock wise
OUTPUT A (IRQ)			
OUTPUT B (P1DR.BIT.Bn)			

Interrupt pada pin IRQ akan aktif ketika mikrokontroler melacak falling-edge dan kemudian akan memeriksa keadaan pulsa pada pin P1DR.BIT.Bn. Apabila pin P1DR.BIT.Bn dalam keadaan high (1), maka sensor melacak motor yang berputar CCW dan jumlah pulsa akan berkurang. Dan sebaliknya, apabila ketika interrupt terjadi pin P1DR.BIT.Bn dalam keadaan low (0), maka sensor melacak motor yang berputar CW dan jumlah pulsa akan bertambah.

Rotary encoder yang membutuhkan tegangan input sebesar 12 V akan menghasilkan output sebesar ± 10 V, sedangkan mikrokontroler hanya dapat menerima tegangan input maksimal 5 V. Oleh karena itu, output rotary encoder dihubungkan dengan rangkaian yang akan mengkonversi tegangan output dari rotary encoder sebesar 12 V menjadi 5 V dengan fungsi sensor *active high* untuk input mikrokontroler.

Rangkaian pengkonversi tegangan, dapat dilihat pada tabel 3.3, terdiri dari sebuah transistor NPN dan dua buah resistor. Input dari rotary encoder dihubungkan dengan kaki basis dan Vcc dihubungkan dengan kaki collector pada transistor. Sebuah LED dapat dihubungkan pada kaki collector sebagai indikator tegangan.

3.5. Prosedur Pengambilan Data

Pengambilan data dilakukan di Bagian Instalasi Radioterapi Rumah Sakit Persahabatan dengan sumber radiasi yang dipakai adalah Cobalt-60. Pengambilan data dilakukan di dua ruang, yaitu ruang operator dan ruang pemaparan. Pada saat pengambilan data, wadah fantom air yang berukuran 40 x 40 x 40 diletakkan di atas meja pasien pada ruang pemaparan, sementara pengamat berada di ruang operator. Antara ruang operator dan ruang pemaparan radiasi dipisah oleh dinding yang sangat tebal untuk mencegah paparan radiasi yang dapat membahayakan pengamat.

Sebelum memulai pengambilan data, wadah fantom terlebih dahulu diisi dengan air dan ionization chamber dipasang pada detektor holder. Selain itu, harus dipastikan juga semua konektor kabel tersambung dengan baik untuk menghindari kesalahan pada data. Luas lapangan radiasi yang dipakai pada penelitian ini adalah 10 x 10 cm dengan jarak antara sumber terhadap permukaan fantom (SSD) adalah 80 cm. Pengambilan data dilakukan pada kedalaman 5 cm dan 10 cm dari permukaan dengan perpindahan horizontal tiap 1 cm dengan range 10 cm dari sumbu utama.

Pada ruang operator, pengamat dapat mengatur posisi detektor dan mengamati posisi aktualnya melalui GUI yang sudah terintegrasi di PC. Hasil keluaran detektor ionization chamber yang berupa muatan dapat diamati melalui elektrometer.

BAB 4

HASIL DAN PEMBAHASAN

4.1. Pemrograman Melalui Mikrokontroler H8/3069 F

Agar dapat dikontrol dari jarak jauh, fantom air dilengkapi dengan motor dan konektor serial yang terhubung dengan mikrokontroler *H8/3069 F*. Setelah detektor mencapai posisi yang diinginkan, detektor tersebut diarahkan kembali ke posisi awal sebelum diarahkan ke posisi selanjutnya. Posisi aktual dari detektor dapat diketahui dari jumlah pulsa yang dihasilkan oleh rotary encoder yang pasang satu sumbu dengan servo motor continuous.

Pada pemrograman, untuk kedua sumbu horizontal dan vertikal, terdiri dari perintah yang sama. Hanya saja dilakukan secara bergantian dengan variabel yang berbeda yaitu CW dan CCW. Di sini akan dijelaskan program pada sumbu vertikal. Hal yang sama juga berlaku pada sumbu horizontal. Pada posisi awal pengambilan data posisi, perlu diingat bahwa detektor berada pada posisi paling kiri untuk sumbu horizontal dan posisi paling atas pada sumbu vertikal.

Apabila tombol reset pada mikrokontroler ditekan, maka pengguna dapat memberikan perintah melalui Hyperterminal pada PC arah gerak detektor dalam dua dimensi dan besarnya pulsa yang harus dicapai agar motor berhenti. Arah gerak detektor diberikan dalam bentuk *string* (\$) dan dilanjutkan dengan perintah. Sebagai contoh :

Tabel 4.1. Contoh sistem kontrol melalui hyperterminal

Kode	Perintah	Keterangan
\$U 10	Up	Menggerakkan detektor ke atas sejauh 10 mm
\$D 10	Down	Menggerakkan detektor ke bawah sejauh 10 mm
\$L 10	Left	Menggerakkan detektor ke kiri sejauh 10 mm
\$R 10	Right	Menggerakkan detektor ke kanan sejauh 10 mm

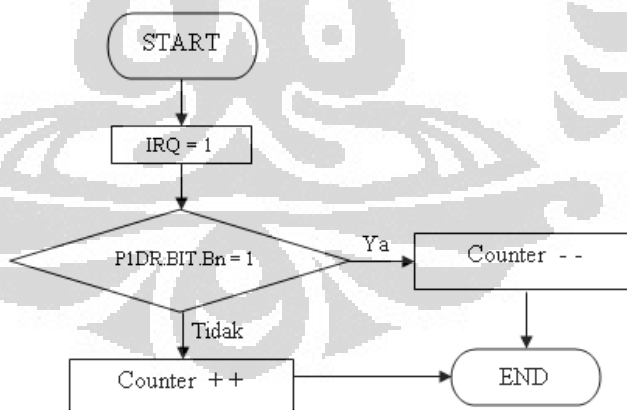
Setelah arah dimasukkan, beri spasi () dan masukkan besar pulsa yang diinginkan. Untuk mempersingkat penulisan, pulsa telah dikonversi menjadi perubahan posisi detektor yang didapat melalui proses kalibrasi. Sehingga, untuk setiap 172 pulsa dikonversi menjadi perpindahan ketinggian sejauh 1 mm pada

sumbu vertikal dan 148 pulsa untuk setiap perpindahan posisi yang sama pada sumbu horizontal.

Setelah memasukkan variabel kontrol, maka motor akan bergerak sesuai dengan arah yang diinginkan. Dengan Bergeraknya motor, maka rotary encoder yang dipasang satu sumbu dengan motor juga ikut berputar dan mulai menghasilkan output berupa pulsa. Output pulsa ini kemudian akan dibandingkan dengan jumlah pulsa yang diinginkan.

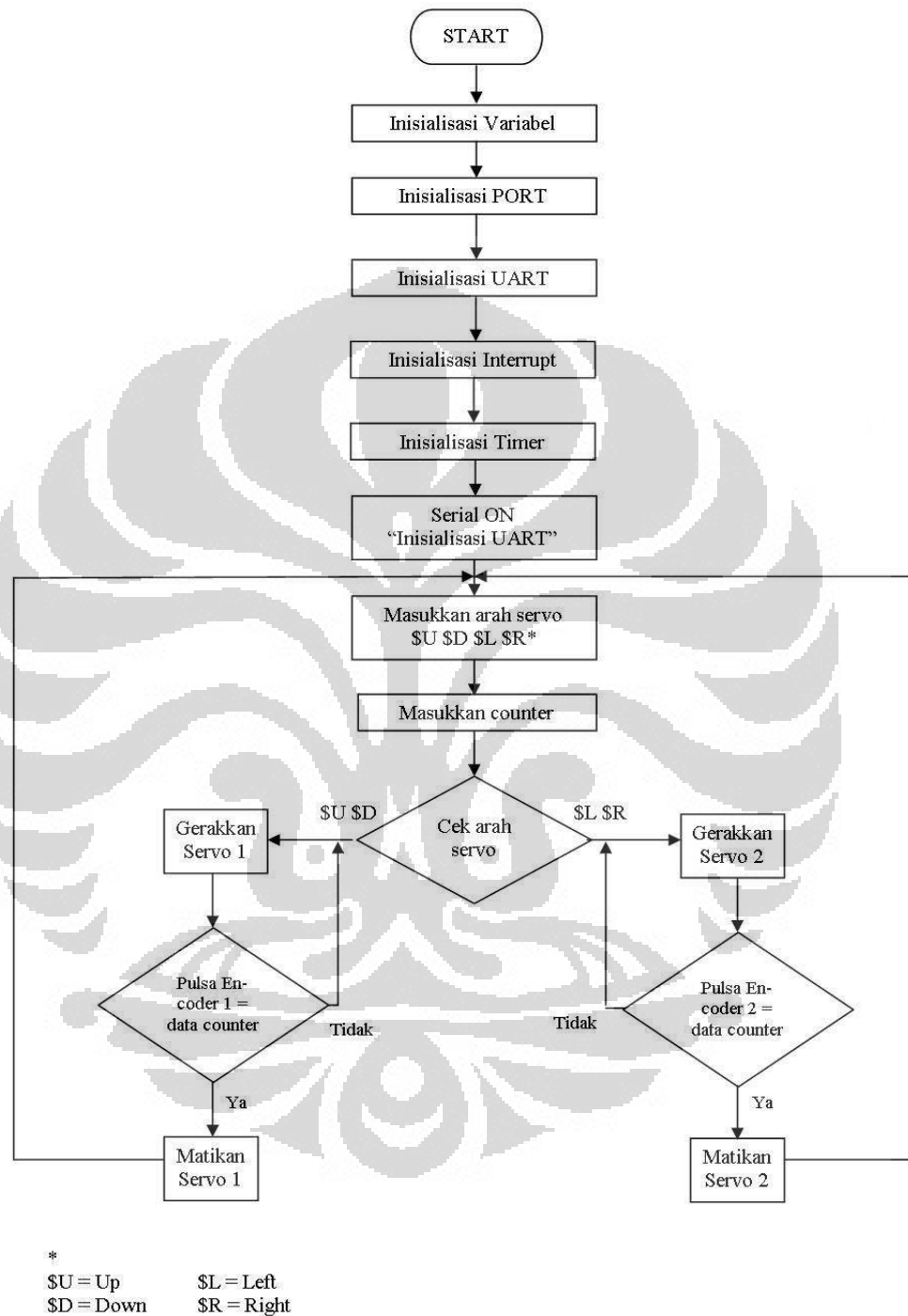
Pulsa dari output A pada masing-masing sensor rotary encoder dihubungkan ke port *interrupt* (IRQ) sebagai penginterupsi sedangkan output B dihubungkan ke port I/O (P1DR.BIT.Bn) sebagai pendeteksi arah putaran servo motor. Ketika IRQ aktif, mikrokontroler akan mendeteksi apakah pin P1DR.BIT.Bn aktif atau tidak. Apabila pin tersebut aktif maka pulsa akan berkurang, dan sebaliknya, jika pin tidak aktif maka pulsa bertambah. Langkah ini dilakukan agar ketika servo motor berbalik arah, maka pulsa yang dihasilkan agar posisi aktual bisa terdeteksi.

Data pulsa ini kemudian dikirimkan melalui komunikasi serial. Apabila pulsa rotary tersebut sudah sama dengan jumlah pulsa yang diinginkan, maka motor akan berhenti. Berikut flowchart program untuk interrupt:



Gambar 4.1. Flowchart program interupsi

Setelah detektor mencapai posisi yang diinginkan dan hendak dikembalikan ke posisi awal, maka yang harus kita lakukan adalah memberikan perintah yang sama dengan sebelumnya, namun dengan arah yang berbeda. Berikut adalah flowchart program secara umum:



Gambar 4.2. Flowchart program

Berikut penjelasan flowchart program :

Pertama, agar variabel yang dipakai pada program ini dikenal oleh mikrokontroler, maka harus diinisialisasi terlebih dahulu dengan mencantumkan ukuran variabel tersebut. Ukuran tiap variabel sebaiknya sesuai dengan kebutuhan. Contoh script untuk program ini adalah sebagai berikut:

```
long int counter = 0;
long int counter2 = 0;
long int posisi, ketinggian;
```

Variabel counter dan counter2 menunjukkan jumlah pulsa yang dihasilkan oleh rotary encoder pada sumbu vertikal dan horizontal.

Kedua, port yang dipakai pada sistem ini juga harus diinisialisasi apakah port tersebut difungsikan sebagai input atau output . Apabila port di-set menjadi 1, maka fungsi port tersebut menjadi output dan apabila port di-set menjadi 0, maka fungsi port tersebut menjadi input. Port 1 dipakai untuk membaca keadaan output B pada rotary encoder. Berikut script-nya:

```
P1DDR = 0x00;
```

Ketiga, dikarenakan pada proses pengiriman dan pengambilan data diperlukan komunikasi RS232, maka diperlukan inisialisasi UART pada awal program untuk memanggil fungsi UART pada library uart.h. Setelah itu, sebagai penanda bahwa fungsi tersebut sudah dapat dipakai, maka pada Hyperterminal akan tampil kalimat "Inisialisasi UART".

```
uart_init();
uart_strP("Inisialisasi Uart\r\n");
```

Keempat, inisialisasi interrupt. Interrupt pada program ini dipakai untuk menghitung pulsa yang dihasilkan oleh sensor rotary encoder. Mode interrupt yang dipakai adalah *falling edge*, yaitu interrupt akan aktif ketika bertemu dengan sisi pulsa dari keadaan high ke low. Register IER untuk mengatur chanel interrupt yang hendak diaktifkan dan register ISCR dipakai untuk memilih mode interrupt.

```
IER |= (1 << IRQ0E);
ISCR |= (1 << IRQ0SC);
```

```
IER |= (1 << IRQ1E);
ISCR |= (1 << IRQ1SC);
```

Kelima adalah menginisialisasi timer yang digunakan untuk membangkitkan pulsa melalui metode PWM. Seperti yang sudah dijelaskan sebelumnya pada prosedur PWM (gambar 2.15), maka konfigurasi bit yang

dibutuhkan untuk melaksanakan mode yang diinginkan dapat diaktifkan pada register-register PWM. Penulisan script-nya adalah sebagai berikut:

```
TCR0_16 |= 0x23; //prescaler 8 --> 25MHz/8 dan Clear
counting bila GRA compare match
TCR1_16 |= 0x23;
TIOR0 |= 0x99; //channel 0 : output 0 saat GRB compare
dan output 1 saat GRA compare
TIOR1 |= 0x99;
TMDR |= 0x03; //pwm mode 0 & 1
TCNT0_16 = 0x0000; //counter mulai dari 0x0000
TCNT1_16 = 0x0000;
TSTR &= 0x00; //belum mulai counting
```

Keenam, setelah inialisasi selesai, maka user dapat memasukkan perintah seperti yang tertera pada tabel 4.1. Apabila perintah \$U atau \$D yang dimasukkan, maka setelah jumlah perpindahan (mm) dimasukkan dan tombol Enter ditekan, servo motor 1 (pada sumbu vertikal) akan berputar dengan menggunakan metode PWM. Motor berputar *counterclockwise* (CCW) untuk \$U dan untuk *clockwise* (CW) \$D sejauh jumlah perpindahan yang dimasukkan. Berikut adalah fungsi untuk putaran CCW pada motor:

```
void up(void)
{
    GRA0 = 62500; //20 ms
    GRB0 = 6250 ; //1.7 ms (CCW)
    TSTR |= (1 << STR0); //mulai counting
    TISRA |= (1 << IMIEA0); //Nyalakan interrupt IMIA
}
```

Ketika motor berputar, maka rotary encoder yang dipasang satu sumbu dengannya juga akan berputar dan menghasilkan pulsa output sehingga interrupt terdeteksi oleh mikrokontroler. Ketika interrupt terjadi, maka mikrokontroler kemudian akan mengecek keadaan pin 5 pada port 1 (output B rotary encoder), apakah dalam keadaan high atau low. Apabila high, maka jumlah pulsa akan berkurang satu dan sebaliknya.

```
void ISR_IRQ0(void)
{
    ISR &= ~(1 << IRQ0F); //check data output B, Jika
    bernilai 0 maka counter naik
    if (P1DR & (1 << PIN5)) {
        counter--;
    } else {
        counter++;
    }
}
```

Jumlah perpindahan akan dikonversi ke dalam jumlah pulsa yang akan dijadikan pulsa acuan, sehingga apabila pulsa aktual yang dihasilkan rotary encoder sama dengan pulsa acuan, maka motor akan berhenti berputar.

4.2. Prosedur Download

Untuk melakukan pengambilan data sensor dan pengiriman data ke serial, mikrokontroler H8/3069F diprogram dengan menggunakan bahasa pemrograman C. Setelah itu, program yang telah dibuat dalam bentuk dot c (.c), di compile kedalam bentuk dot mot (.mot) dan didownload dengan cara masuk pada command sistem windows. Pada saat melakukan download, mode yang digunakan adalah mode *read*, atau mode 3. Pengaturan *switch* pada mode tersebut adalah sebagai berikut:

	Pin-1	Pin-2	Pin-3	Pin-4
ON	O	O		O
OFF			O	

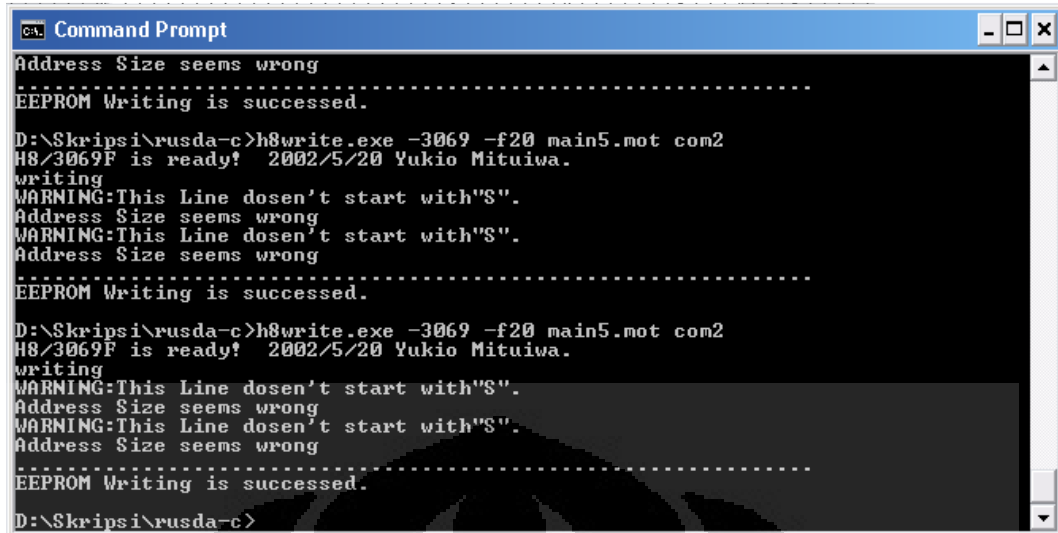
Gambar 4.3. Susunan Switch Mode Write

Untuk mendownload file programnya, buka *DOS Command Line(Command Promp)* pada komputer, arahkan ke folder tempat file programnya (.mot), lalu ketikkan :

```
h8write.exe -3069 -f20 namafile.mot
```

dan tekan tombol *ENTER*.

Pada saat itu akan muncul notifikasi di layar *Command Prompt* yang menandakan bahwa program telah berhasil di download, seperti berikut :



```

ca. Command Prompt
Address Size seems wrong
.....
EEPROM Writing is succeeded.

D:\Skripsi\rusda-c>h8write.exe -3069 -f20 main5.not com2
H8/3069F is ready! 2002/5/20 Yukio Mituiwa.
writing
WARNING:This Line dosen't start with"S".
Address Size seems wrong
WARNING:This Line dosen't start with"S".
Address Size seems wrong
.....
EEPROM Writing is succeeded.

D:\Skripsi\rusda-c>h8write.exe -3069 -f20 main5.not com2
H8/3069F is ready! 2002/5/20 Yukio Mituiwa.
writing
WARNING:This Line dosen't start with"S".
Address Size seems wrong
WARNING:This Line dosen't start with"S".
Address Size seems wrong
.....
EEPROM Writing is succeeded.

D:\Skripsi\rusda-c>

```

Gambar 4.4. Cara Mendownload Program Mikrokontroler dengan Bahasa C

Kemudian tekan tombol reset dan kembalikan lagi pengaturan *switch* H8 ke mode 7 atau mode *read*. Tampilan *switch* untuk mode tersebut adalah sebagai berikut :

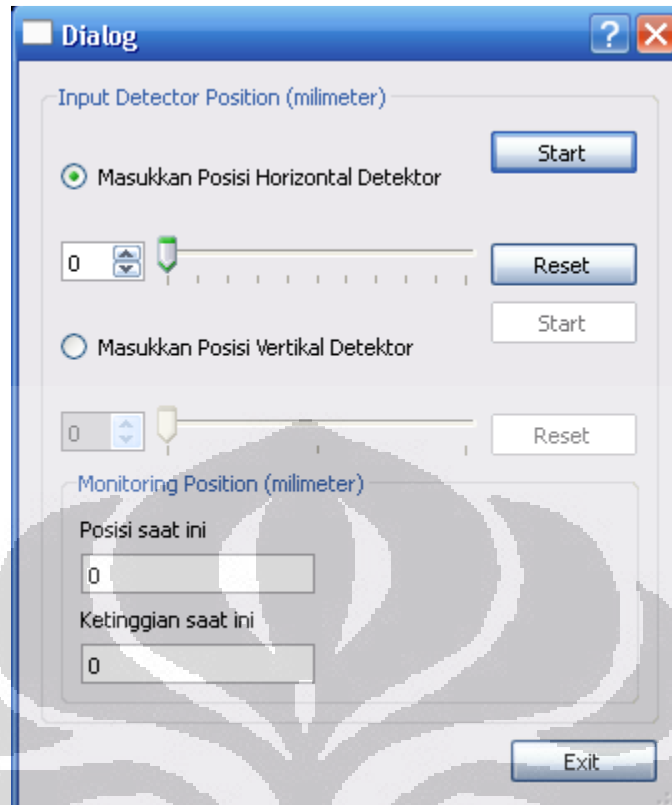
	Pin-1	Pin-2	Pin-3	Pin-4
ON	O	O	O	
OFF				O

Gambar 4.5. Susunan Switch Mode Read

Pada keadaan ini, mikrokontroler akan melakukan perintah yang telah terprogram dan terdownload di dalamnya^[13].

4.3.GUI

GUI atau *Graphic User Interface* dapat dikatakan suatu penghubung yang memudahkan kita dalam membaca data yang telah kita peroleh. Tampilan awal data yang diperoleh dari pembacaan *counter* dapat dilihat pada gambar 4.6.



Gambar 4.6. Tampilan awal GUI

Pertama-tama user akan memilih sumbu mana yang akan dipakai untuk menjalankan detektor. User tidak dapat menjalankan detektor di kedua sumbu secara bersamaan untuk mengurangi error yang disebabkan oleh getaran dari tiap motor. Setelah memilih sumbu, user akan mengisi jumlah perpindahan detektor sesuai dengan yang dikehendaki dalam milimeter pada box 'Input Detector Position'. Perlu diingat bahwa posisi awal detektor sebelum dijalankan harus pada posisi 0 mm. Kemudian tombol 'Start' ditekan, maka detektor mulai berjalan sejauh perpindahan yang diinginkan. Dengan memilih tombol 'Reset', maka posisi detektor akan kembali pada posisi awal (0 mm). Posisi dan ketinggian aktual dapat dilihat pada box 'Posisi saat ini' dan 'Ketinggian saat ini'.

4.4. Analisis Servo Motor

Berdasarkan datasheet servomotor continuous, berikan pulsa 1.5 ms untuk menghentikan servo motor dengan periode 20 ms. Untuk memutar servo ke kanan (CW), berikan pulsa ≤ 1.3 ms, dan pulsa ≥ 1.7 ms untuk berputar ke kiri (CCW) dengan periode masing-masing sebesar 20 ms.

Dengan frekuensi sumber clock (f_{clock}) sebesar 25 MHz, maka frekuensi PWM (f_{PWM}) adalah :

$$\begin{aligned} f_{PWM} &= f_{clock} \times T_{PWM} \\ &= 25 \text{ MHz} \times 20 \text{ ms} \\ &= 500000 \text{ Hz} \end{aligned}$$

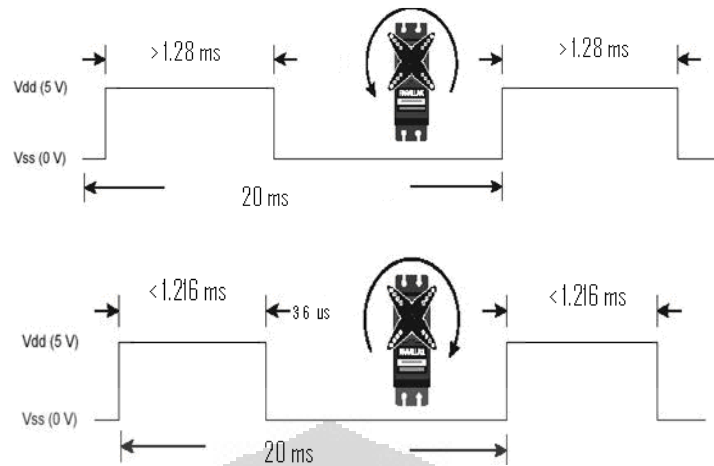
Namun, dikarenakan maksimal counter yang dapat dicapai oleh 16 bit timer adalah $2^{16} = 65536$, maka diperlukan *prescaler* agar PWM ini dapat terpenuhi. *Prescaler* yang dipakai adalah 8 sehingga nilai yang dicapai pulsa sebelum di-clear-kan (GRA) adalah 62500. Sementara itu, nilai GRB didapatkan dari perbandingan pulsa *high* terhadap *low* yang didapat dari datasheet.

Sebagai contoh, untuk putaran ke arah CW, maka dibutuhkan pulsa high ≤ 1.3 ms dengan periode 20 ms. Maka frekuensi untuk pulsa high tersebut adalah :

$$\begin{aligned} \frac{GRA}{GRB} &= \frac{T_{high}}{T_{PWM}} \\ GRB &= \frac{62500 \times 1.3}{20} \\ GRB &= 4062.5 \end{aligned}$$

Kemudian, setelah nilai-nilai tersebut dimasukkan ke dalam program, servo motor hanya berputar pada satu arah, yaitu *counter clockwise*. Oleh karenanya, penulis mencoba untuk mengubah nilai GRB tanpa mengikuti teori sehingga didapat nilai-nilai GRB untuk tiap gerakan sebagai berikut:

$$\begin{aligned} \textit{Clockwise} &= <3800 \text{ (} T \leq 1.216 \text{ ms)} \\ \textit{Counter clockwise} &= >4000 \text{ (} T \geq 1.28 \text{ ms)} \\ \textit{Stop} &= 3900 \text{ (} T = 1.248 \text{ ms)} \end{aligned}$$



Gambar 4.7. Pulsa high sebenarnya pada servo motor continuous

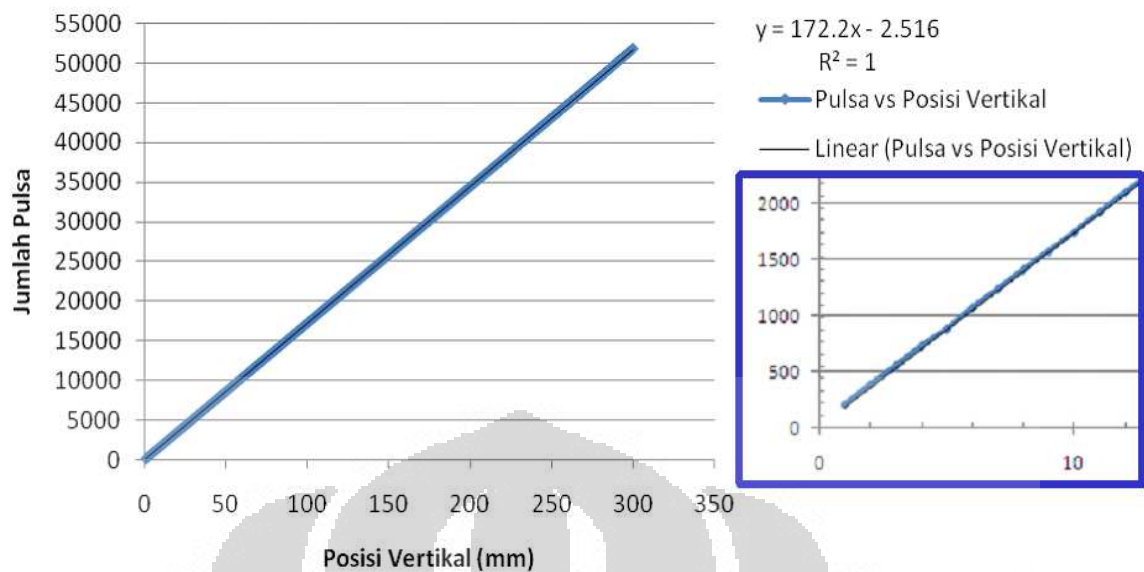
Hal tersebut terjadi dikarenakan belum adanya standarisasi metode PWM yang dipakai pada tiap mikrokontroler berbeda yang didasarkan pada sistem siklus clock masing-masing mikrokontroler.

4.5. Data perbandingan pulsa encoder terhadap posisi vertikal detektor

Untuk mengetahui berapa pulsa yang dihasilkan oleh sensor rotary encoder dalam setiap interval perpindahan 1 mm adalah dengan membuat program untuk menghitung pulsa rotary encoder yang kemudian ditampilkan melalui Hyperterminal pada PC. Agar didapat data yang presisi, maka dilakukan pengambilan data sebanyak tiga kali. Untuk data pada saat detektor bergerak pada perpindahan vertikal adalah seperti yang tertera pada Tabel A.1.

Apabila dilihat pada data tabel A.1, untuk perpindahan posisi vertikal sejauh 0-30 cm, hasil pulsa rata-rata yang didapat adalah 51741. Data tersebut kemudian diubah menjadi grafik seperti yang tampak pada gambar 4.1. Pada grafik tersebut dapat dilihat bahwa perbandingan antara output dari pulsa rotary encoder terhadap input berupa posisi ketinggian adalah mendekati liner dengan persamaan $y = 172,2x - 2,516$. Variabel x adalah posisi vertikal detektor dan variabel y adalah pulsa yang dihasilkan oleh rotary encoder.

Dengan menggunakan metode kuadrat terkecil, maka akan didapat simpangan baku sebesar (σ) 31,46 dan kesalahan relatif (E_r) sebesar 0,012 %. Koefisien korelasi (r) antara x dan y adalah 0,99.



Gambar 4.8. Grafik perbandingan pulsa rotary encoder terhadap perpindahan posisi vertikal detektor

Persamaan ini kemudian dimasukkan kedalam program Bahasa C yang digunakan untuk mengkonversi input berupa ketinggian menjadi jumlah pulsa yang dibutuhkan agar detektor mencapai posisi yang diinginkan. Setelah itu, posisi ketinggian kembali ditampilkan oleh GUI Python dengan cara membalikkan kembali persamaan tersebut. Hasil dari pembacaan posisi horizontal aktual melalui GUI dengan pembacaan posisi horizontal secara langsung dengan menggunakan penggaris dapat dilihat pada Tabel 4.2.

Tabel 4.2. Data Pengamatan Penggerak Vertikal

Posisi (mm)	Jumlah Pulsa		Error Pulsa	Posisi Pengamatan (mm)	
	Pengamatan	Acuan		GUI	Penggaris
0	0	0	0	0	0
10	1681	1679	-2	9.8	10
20	3391	3384	-7	19.7	20
30	5107	5106	-1	29.7	30
40	6831	6822	-9	39.7	40
50	8540	8538	-2	49.7	50
60	10246	10245	-1	59.6	60
70	11961	11958	-3	69.5	70
80	13677	13671	-6	79.5	80
90	15391	15389	-2	89.5	90

100	17105	17095	-10	99.5	100
110	18827	18822	-5	109.5	110
120	20552	20542	-10	119.6	120.5
130	22273	22266	-7	129.5	130.5
140	23984	23976	-8	139.4	140.5
150	25705	25695	-10	149.5	150.5
160	27415	27405	-10	159.4	161
170	29123	29120	-3	169.3	171
180	30838	30835	-3	179.3	180.5
190	32533	32524	-9	189.1	190.5
200	34240	34238	-2	199.1	201
210	35954	35945	-9	209.0	211
220	37674	37663	-11	219.0	221.5
230	39394	39386	-8	229.0	231.5
240	41113	41108	-5	239.0	241.5
250	42830	42824	-6	249.0	251.5
260	44557	44549	-8	259.0	261.5
270	46284	46276	-8	269.0	272
280	47995	47990	-5	279.0	382
290	49711	49706	-5	289.0	392
300	51431	51427	-4	299.0	302

Dari data yang terbaca pada penggaris, nilai error terbesar (E_{max}) adalah 2 mm, sehingga :

$$\% E = \frac{E_{max}}{\text{Nilai Maks}} \times 100\%$$

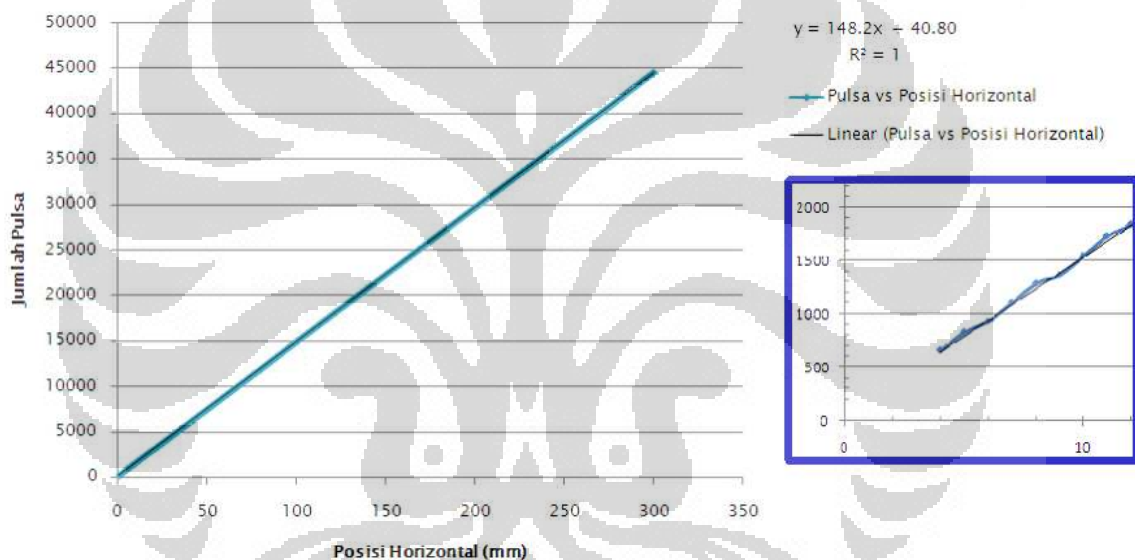
$$= 0,67\%$$

4.6.Data perbandingan pulsa encoder terhadap posisi horizontal detektor

Sama halnya dengan posisi vertikal, untuk mengetahui berapa pulsa yang dihasilkan oleh sensor rotary encoder dalam setiap interval perpindahan jarak 1 mm adalah dengan membuat program untuk menghitung pulsa yang kemudian ditampilkan melalui Hyperterminal pada PC. Agar didapat data yang presisi, maka dilakukan penhambilan data sebanyak tiga kali. Maka untuk data pada saat detektor bergerak pada perpindahan horizontal adalah seperti yang terlihat pada Tabel A.2.

Seperti yang dapat dilihat pada tabel tersebut, dengan range 0-30 cm, hasil pulsa rata-rata yang didapat adalah 44543. Data tersebut kemudian diubah menjadi grafik seperti yang tampak pada gambar 4.3. Pada grafik tersebut dapat dilihat bahwa perbandingan antara output dari pulsa rotary encocer terhadap input berupa posisi ketinggian adalah mendekati liner dengan persamaan $y = 148,2x + 40,80$. Variabel x adalah posisi horizontal detektor dan variabel y adalah pulsa yang dihasilkan oleh rotary encoder.

Dengan menggunakan metode kuadrat terkecil, maka akan didapat simpangan baku (σ) sebesar 24,07 dan kesalahan relatif (E_r) sebesar 0,011 %. Koefisien korelasi (r) antara x dan y adalah 0,99.



Gambar 4.9. Grafik perbandingan pulsa rotary encoder terhadap perpindahan posisi horizontal detektor

Persamaan ini kemudian dimasukkan kedalam program Bahasa C yang digunakan untuk menngkonversi input berupa posisi horizontal menjadi jumlah pulsa yang dibutuhkan agar detektor mencapai posisi yang diinginkan. Setelah itu, posisi ketinggian kembali ditampilkan oleh GUI Python dengan cara membalikkan kembali persamaan tersebut. Hasil dari pembacaan posisi horizontal aktual melalui GUI dengan pembacaan posisi horizontal secara langsung dengan menggunakan penggaris dapat dilihat pada Tabel 4.3.

Tabel 4.3. Data Pengamatan Penggerak Horizontal

Posisi (mm)	Jumlah Pulsa		Error Pulsa	Posisi Pengamatan (mm)	
	Pengamatan	Acuan		GUI	Penggaris
0	0	0	0	0	0
10	1481	1480	-1	10.0	10
20	2975	2984	9	20.1	20
30	4462	4480	18	30.2	30
40	5979	5995	16	40.4	40
50	7487	7505	18	50.8	50
60	8995	9013	18	60.8	60
70	10423	10441	18	70.4	70.5
80	11939	11948	9	80.7	80.5
90	13440	13453	13	90.8	90.5
100	14898	14911	13	100.7	100.5
110	16397	16412	15	110.8	110.5
120	17901	17912	11	121.0	120.5
130	19365	19383	18	130.8	130.5
140	20881	20893	12	141.0	140.5
150	22389	22408	19	151.3	150.5
160	23857	23875	18	161.2	161
170	25301	25312	11	170.1	170.5
180	26807	26817	10	181.1	181
190	28341	28349	8	191.5	190.5
200	29721	29730	-9	200.8	200
210	31216	31233	17	210.9	210
220	32747	32760	13	221.3	220.5
230	34132	34143	11	230.6	230
240	35559	35569	10	240.3	240
250	37059	37051	-8	250.4	250
260	38554	38526	-28	260.5	260
270	40059	39994	-65	270.7	270
280	41580	41499	-81	280.9	280.5
290	43079	42994	-85	291.1	290.5
300	44575	44482	-93	301.2	300.5

Dari data yang terbaca pada penggaris, nilai error terbesar (E_{max}) adalah 1 mm sehingga :

$$\begin{aligned} \% E &= \frac{E_{max}}{\text{Nilai Maks}} \times 100\% \\ &= 0,33\% \end{aligned}$$

Ketidakkuratan yang terjadi dikarenakan servo motor continuous masih bergerak sedikit setelah diperintahkan untuk berhenti. Walaupun pada data-data diawal hal ini tidak begitu terlihat, namun setelah beberapa pengambilan data, ketidakkuratan mulai terlihat dikarenakan perbedaan jumlah pulsa yang semakin bertambah.

Selain itu, kesalahan dapat juga terjadi dikarenakan kesalahan pada desain mekanik alat dan kualitas pekerjaannya. Karena, seperti yang terlihat pada lapangan, pada ulir terlihat masih adanya kecacatan seperti ketidaksimetrisan dan bentuk ulir yang tidak rata.

Dari hasil perbandingan data antara sumbu vertikal dan horizontal dapat dilihat bahwa error pada sumbu vertikal lebih besar daripada error pada sumbu horizontal. Hal ini dikarenakan pada pergerakan pada sumbu vertikal adanya pengaruh gaya gravitasi bumi yang bekerja pada detektor lebih besar daripada pergerakan pada sumbu horizontal.

4.7. Kurva Profil Radiasi Cobalt-60

Profil berkas radiasi Cobalt-60 didapat dengan pengambilan data pada dua titik kedalaman, yaitu kedalaman detektor (d) 5 cm dan 10 cm dari permukaan. Pada tiap kedalaman dilakukan pengambilan data horizontal sejauh 10 cm dengan perpindahan tiap 1 cm. Jarak SSD (*Source-Surface Distance*) adalah 80 cm.

Dengan asumsi bahwa profi dosis yang dihasilkan oleh pesawat Cobalt-60 bersifat simetris, maka pengukuran paparan dosisnya dilakukan pada satu sisi saja dari sumbu berkasnya dengan luas lapangan radiasi 10 x 10 cm.

Hasil keluaran dari detektor ionization chamber yang berupa muatan dapat dilihat pada elektrometer. Dosis relatif radiasi didapat dari normalisasi terhadap muatan pada sumbu berkas $d = 5$ cm.

$$\text{Relative dose (\%)} = \frac{Q_d}{Q_5} \times 100 \%$$

Q_d = Muatan pada kedalaman yang diamati

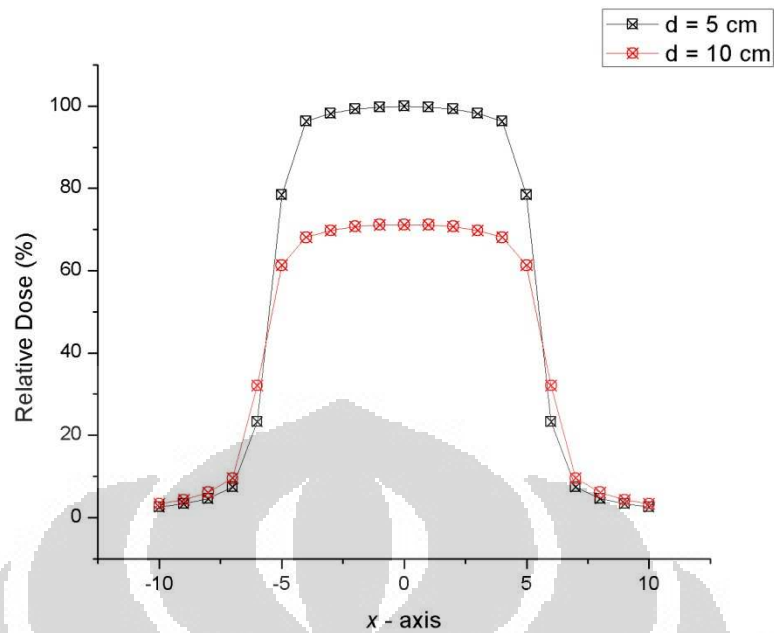
Q_5 = Muatan pada kedalaman 5 cm

Berikut adalah data pengamatan muatan dan dosis relatif radiasi dengan kedalaman detektor 5 cm dan 10 cm :

Tabel 4.4. Data muatan dan dosis relatif di kedalaman 5 cm dan 10 cm

x (cm)	d = 5 cm		d = 10 cm	
	muatan (nC)	% dose	muatan (nC)	% dose
0	17.5	100	12.45	71.14
1	17.46	99.77	12.45	71.14
2	17.37	99.26	12.38	70.74
3	17.18	98.17	12.21	69.77
4	16.86	96.34	11.92	68.11
5	13.72	78.4	10.73	61.31
6	4.08	23.31	5.62	32.11
7	1.31	7.48	1.67	9.54
8	0.8	4.57	1.07	6.12
9	0.59	3.37	0.76	4.34
10	0.45	2.57	0.59	3.37

Bila data di atas dituangkan dalam bentuk grafik yang utuh untuk kedua sisi dan ternormalisasi terhadap paparan dosis sumbu berkas dengan $d = 5$ cm, maka diperoleh grafik profil dosis radiasi terhadap posisi kedalaman detektor yang berbeda seperti pada gambar 4.10.



Gambar 4.10. Perbandingan profil dosis relatif untuk kedalaman 5 cm dan 10 cm untuk SSD = 80 cm dan lapangan 10 x 10 cm

Dari gambar di atas dapat dilihat bahwa untuk SSD yang konstan, dengan semakin besarnya kedalaman detektor, maka sebaran dosis pada daerah sekitar sumbu berkas akan semakin seragam. Sementara itu, jumlah dosis pada daerah di luar lapangan radiasi akan mengalami penurunan secara drastis. Hal ini menunjukkan bahwa pesawat radiasi Cobalt-60 yang digunakan pada penelitian ini masih bekerja dengan baik karena tepat mengenai sasaran yang direpresentasikan oleh luas lapangan sebesar 10 x 10 cm dengan sedikit efek pada daerah di sekitarnya. Hasil ini sesuai dengan prinsip radioterapi yang telah dijelaskan sebelumnya pada Bab II.

BAB 5

PENUTUP

Pada bab ini berisi kesimpulan dan saran dari keseluruhan pengambilan data pada sistem mekanik pengendali posisi detektor dua dimensi dan saran yang dapat digunakan untuk memperbaiki laporan penulis.

5.1. Kesimpulan

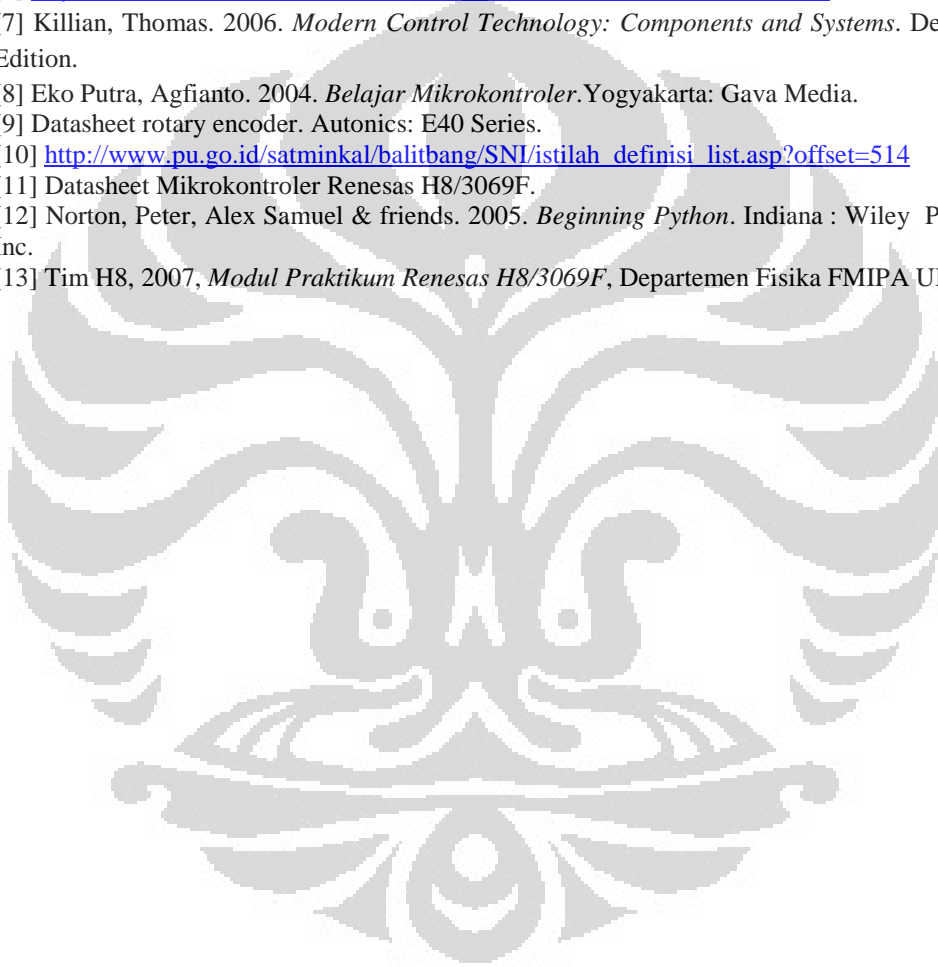
1. Sistem mekanik dengan harga yang terjangkau dan efisien untuk menggerakkan detektor dua dimensi pada fantom air dua dimensi dengan berbasis mikrokontroler H8/3069 F berhasil dibuat dengan volume wadah 40 x 40 x 40 cm.
2. Jangkauan detektor pada sumbu horizontal dan vertikal masing-masing adalah 30 cm.
3. Error posisi pada penggerak vertikal dan horizontal masing-masing adalah 0,67 % dan 0,33 %.

5.2. Saran

1. Apabila ingin membuat sistem mekanik pengendali posisi detektor pada fantom air yang sempurna, dapat ditambahkan fitur pergerakan detektor tiga dimensi dan informasi tambahan dari detektor radiasi yang terintegrasi dengan sistem untuk mempermudah user dalam mengambil data yang lengkap.
2. Mekanik sebaiknya terbuat dari bahan non-logam agar tidak berbahaya apabila terkena paparan radiasi. Selain itu, bahan juga dipilih yang ringan dan kuat.

DAFTAR PUSTAKA

- [1] Podgorsak, E.B. 2003. *Review of Radiation Oncology Physics: A Handbook for Teachers and Students*. Vienna : IAEA
- [2] Nasukha. *Presiding Presentasi Ilmiah Keselaraatan Radiasi dan Lingkungan, 20 - 21 Agustus 1996*. ISSN : 0854-4085
- [3] Karzmark, C.J. 1984. *Advances in Linear Accelerator Design for Radiotherapy*. Medical Physics, 11 (2): 105- 126.
- [4] Goiten, M. Int. 1985. *Journal Radiation Oncology, Biology, Physics*. Vol 12, hal 701-704.
- [5] Cunningham, John Robert, Harold Elford Johns. 1983. *The Physisc of Radiology fourth edition*. USA: Charles C Thomas Publisher.
- [6] <http://www.acroname.com/robotics/info/articles/servo/servo.html#e1>
- [7] Killian, Thomas. 2006. *Modern Control Technology: Components and Systems*. Delmar: 2nd Edition.
- [8] Eko Putra, Agfianto. 2004. *Belajar Mikrokontroler*. Yogyakarta: Gava Media.
- [9] Datasheet rotary encoder. Autonics: E40 Series.
- [10] http://www.pu.go.id/satminkal/balitbang/SNI/istilah_definisi_list.asp?offset=514
- [11] Datasheet Mikrokontroler Renesas H8/3069F.
- [12] Norton, Peter, Alex Samuel & friends. 2005. *Beginning Python*. Indiana : Wiley Publishing, Inc.
- [13] Tim H8, 2007, *Modul Praktikum Renesas H8/3069F*, Departemen Fisika FMIPA UI, Depok.



LAMPIRAN A

Data Pengamatan Jumlah Pulsa Encoder Terhadap Perpindahan Detektor

Tabel A.1. Jumlah Pulsa Terhadap Perpindahan Posisi Vertikal Detektor

Posisi (mm)	Jumlah Pulsa pada Pengambilan Data ke-			Jumlah Pulsa Rata-rata
	1	2	3	
1	237	169	203	203
2	395	354	399	382.6667
3	560	524	568	550.6667
4	735	698	749	727.3333
5	925	856	853	878
6	1088	1042	1070	1066.667
7	1260	1210	1228	1232.667
8	1435	1389	1398	1407.333
9	1605	1549	1564	1572.667
10	1745	1735	1734	1738
11	1952	1895	1912	1919.667
12	2125	2058	2100	2094.333
13	2273	2252	2250	2258.333
14	2446	2436	2420	2434
15	2633	2596	2597	2608.667
16	2787	2755	2771	2771
17	2955	2942	2949	2948.667
18	3152	3116	3123	3130.333
19	3306	3293	3279	3292.667
20	3487	3467	3433	3462.333
21	3664	3630	3625	3639.667
22	3806	3798	3805	3803
23	4006	3963	3963	3977.333
24	4170	4134	4131	4145
25	4324	4307	4325	4318.667
26	4521	4496	4472	4496.333
27	4675	4643	4672	4663.333
28	4820	4809	4800	4809.667
29	5037	4995	4905	4979
30	5194	5178	5194	5188.667
31	5324	5337	5304	5321.667
32	5553	5525	5530	5536
33	5707	5683	5692	5694
34	5865	5850	5849	5854.667
35	6069	6019	6045	6044.333
36	6224	6198	6206	6209.333

37	6411	6363	6336	6370
38	6584	6550	6545	6559.667
39	6744	6709	6719	6724
40	6922	6891	6893	6902
41	7094	7071	7073	7079.333
42	7250	7208	7196	7218
43	7446	7407	7406	7419.667
44	7617	7604	7586	7602.333
45	7745	7747	7750	7747.333
46	7948	7921	7934	7934.333
47	8143	8107	8084	8111.333
48	8279	8246	8235	8253.333
49	8480	8441	8436	8452.333
50	8651	8581	8606	8612.667
51	8785	8745	8780	8770
52	8987	8961	8957	8968.333
53	9169	9137	9138	9148
54	9296	9281	9259	9278.667
55	9512	9478	9470	9486.667
56	9686	9662	9652	9666.667
57	9833	9827	9829	9829.667
58	10016	10006	10002	10008
59	10173	10176	10172	10173.67
60	10356	10338	10301	10331.67
61	10548	10523	10508	10526.33
62	10695	10708	10666	10689.67
63	10890	10849	10823	10854
64	11079	11023	11019	11040.33
65	11191	11205	11188	11194.67
66	11416	11357	11337	11370
67	11598	11545	11531	11558
68	11736	11728	11712	11725.33
69	11947	11873	11855	11891.67
70	12135	12088	12055	12092.67
71	12239	12245	12221	12235
72	12458	12384	12373	12405
73	12619	12596	12556	12590.33
74	12792	12762	12727	12760.33
75	12966	12912	12872	12916.67
76	13152	13089	13061	13100.67
77	13312	13264	13228	13268
78	13482	13420	13351	13417.67
79	13648	13606	13588	13614

80	13817	13778	13751	13782
81	13999	13932	13924	13951.67
82	14135	14121	14127	14127.67
83	14309	14291	14259	14286.33
84	14528	14470	14459	14485.67
85	14655	14655	14617	14642.33
86	14849	14815	14788	14817.33
87	15037	14985	14977	14999.67
88	15179	15167	15135	15160.33
89	15386	15323	15255	15321.33
90	15568	15492	15445	15501.67
91	15692	15667	15642	15667
92	15895	15843	15800	15846
93	16072	16007	16025	16034.67
94	16186	16177	16189	16184
95	16405	16355	16311	16357
96	16576	16524	16516	16538.67
97	16692	16695	16668	16685
98	16917	16890	16851	16886
99	17093	17044	17024	17053.67
100	17242	17236	17188	17222
101	17442	17405	17380	17409
102	17607	17546	17529	17560.67
103	17740	17746	17713	17733
104	17947	17928	17853	17909.33
105	18129	18079	18063	18090.33
106	18291	18240	18239	18256.67
107	18487	18410	18413	18436.67
108	18641	18581	18583	18601.67
109	18818	18755	18754	18775.67
110	18988	18956	18900	18948
111	19159	19112	19083	19118
112	19292	19278	19238	19269.33
113	19488	19429	19412	19443
114	19657	19610	19607	19624.67
115	19799	19777	19776	19784
116	20015	19964	19957	19978.67
117	20169	20132	20110	20137
118	20330	20286	20250	20288.67
119	20526	20471	20461	20486
120	20689	20643	20621	20651
121	20863	20809	20808	20826.67
122	21046	20987	20983	21005.33

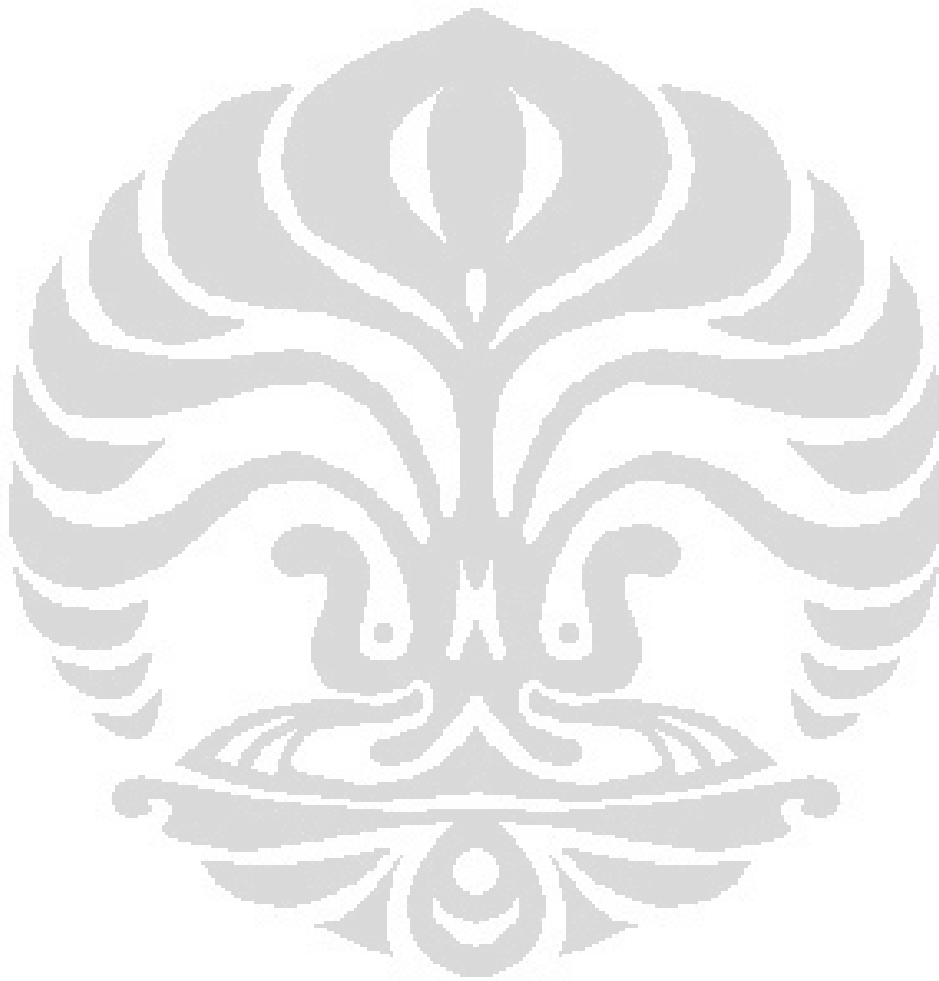
123	21172	21161	21124	21152.33
124	21389	21352	21286	21342.33
125	21538	21488	21509	21511.67
126	21707	21682	21644	21677.67
127	21886	21873	21836	21865
128	22054	22039	22012	22035
129	22244	22193	22182	22206.33
130	22405	22377	22361	22381
131	22577	22542	22532	22550.33
132	22706	22706	22697	22703
133	22922	22892	22890	22901.33
134	23077	23050	23054	23060.33
135	23269	23216	23171	23218.67
136	23499	23398	23389	23428.67
137	23617	23581	23561	23586.33
138	23766	23762	23701	23743
139	23943	23937	23907	23929
140	24123	24083	24075	24093.67
141	24260	24260	24228	24249.33
142	24456	24452	24415	24441
143	24629	24620	24601	24616.67
144	24807	24768	24733	24769.33
145	25006	24962	24929	24965.67
146	25136	25121	25096	25117.67
147	25319	25288	25272	25293
148	25509	25474	25433	25472
149	25668	25645	25594	25635.67
150	25843	25810	25784	25812.33
151	26015	26011	25956	25994
152	26164	26148	26131	26147.67
153	26333	26337	26245	26305
154	26529	26515	26458	26500.67
155	26697	26674	26624	26665
156	26852	26850	26802	26834.67
157	27044	27038	26992	27024.67
158	27228	27183	27153	27188
159	27375	27356	27319	27350
160	27575	27522	27506	27534.33
161	27712	27701	27665	27692.67
162	27905	27869	27817	27863.67
163	28098	28038	28011	28049
164	28241	28208	28178	28209
165	28442	28387	28363	28397.33

166	28588	28554	28544	28562
167	28775	28738	28689	28734
168	28970	28910	28868	28916
169	29100	29072	29061	29077.67
170	29299	29245	29196	29246.67
171	29432	29422	29366	29406.67
172	29610	29585	29561	29585.33
173	29817	29764	29702	29761
174	29999	29930	29912	29947
175	30147	30095	30088	30110
176	30317	30259	30232	30269.33
177	30495	30455	30416	30455.33
178	30664	30620	30585	30623
179	30796	30782	30703	30760.33
180	31039	30988	30922	30983
181	31167	31135	31083	31128.33
182	31322	31280	31253	31285
183	31528	31475	31449	31484
184	31691	31654	31614	31653
185	31879	31825	31747	31817
186	32061	31999	31953	32004.33
187	32223	32174	32105	32167.33
188	32389	32321	32257	32322.33
189	32580	32506	32461	32515.67
190	32731	32702	32630	32687.67
191	32896	32868	32764	32842.67
192	33056	33053	32985	33031.33
193	33225	33223	33148	33198.67
194	33411	33394	33300	33368.33
195	33594	33571	33515	33560
196	33765	33725	33667	33719
197	33946	33891	33830	33889
198	34076	34070	34019	34055
199	34276	34226	34190	34230.67
200	34452	34400	34351	34401
201	34630	34585	34529	34581.33
202	34779	34746	34683	34736
203	34963	34923	34843	34909.67
204	35114	35120	35038	35090.67
205	35291	35272	35201	35254.67
206	35487	35429	35373	35429.67
207	35624	35620	35540	35594.67
208	35837	35772	35750	35786.33

209	36003	35960	35902	35955
210	36134	36121	36057	36104
211	36338	36298	36231	36289
212	36519	36458	36393	36456.67
213	36647	36630	36586	36621
214	36865	36830	36766	36820.33
215	37040	37023	36896	36986.33
216	37164	37151	37066	37127
217	37309	37337	37261	37302.33
218	37507	37531	37432	37490
219	37672	37671	37606	37649.67
220	37916	37861	37800	37859
221	38058	38015	37975	38016
222	38199	38173	38011	38127.67
223	38428	38378	38317	38374.33
224	38583	38552	38483	38539.33
225	38725	38706	38625	38685.33
226	38942	38897	38833	38890.67
227	39131	39053	38988	39057.33
228	39447	39224	39169	39280
229	39564	39417	39381	39454
230	39748	39579	39515	39614
231	39996	39750	39659	39801.67
232	40121	39930	39882	39977.67
233	40263	40094	40013	40123.33
234	40491	40258	40193	40314
235	40605	40455	40401	40487
236	40801	40616	40507	40641.33
237	41038	40769	40710	40839
238	41173	40999	40934	41035.33
239	41320	41119	41054	41164.33
240	41517	41309	41244	41356.67
241	41657	41509	41449	41538.33
242	41855	41649	41578	41694
243	42074	41823	41765	41887.33
244	42184	42006	41960	42050
245	42367	42150	42057	42191.33
246	42575	42326	42262	42387.67
247	42669	42522	42459	42550
248	42863	42677	42624	42721.33
249	43074	42865	42807	42915.33
250	43205	43048	42979	43077.33
251	43399	43210	43117	43242

252	43603	43378	43285	43422
253	43698	43569	43484	43583.67
254	43914	43714	43625	43751
255	44119	43893	43700	43904
256	44233	44078	43992	44101
257	44427	44242	44136	44268.33
258	44647	44427	44342	44472
259	44772	44604	44514	44630
260	44960	44744	44653	44785.67
261	45137	44927	44869	44977.67
262	45318	45113	45023	45151.33
263	45491	45273	45203	45322.33
264	45651	45435	45388	45491.33
265	45833	45614	45546	45664.33
266	46012	45802	45731	45848.33
267	46147	45968	45905	46006.67
268	46306	46134	46041	46160.33
269	46517	46310	46177	46334.67
270	46692	46475	46413	46526.67
271	46872	46655	46550	46692.33
272	47049	46822	46758	46876.33
273	47208	47006	46937	47050.33
274	47372	47190	47063	47208.33
275	47535	47360	47253	47382.67
276	47713	47507	47432	47550.67
277	47872	47692	47592	47718.67
278	48061	47837	47757	47885
279	48233	48024	47944	48067
280	48375	48202	48086	48221
281	48572	48365	48251	48396
282	48757	48545	48473	48591.67
283	48935	48711	48622	48756
284	49100	48896	48744	48913.33
285	49271	49076	48988	49111.67
286	49418	49238	49153	49269.67
287	49588	49598	49302	49496
288	49760	49760	49501	49673.67
289	49925	49921	49675	49840.33
290	50113	50141	49828	50027.33
291	50299	50314	50047	50220
292	50459	50439	50171	50356.33
293	50648	50636	50338	50540.67
294	50818	50802	50500	50706.67

295	50996	50991	50701	50896
296	51142	51148	50849	51046.33
297	51328	51323	51040	51230.33
298	51498	51503	51172	51391
299	51655	51698	51391	51581.33
300	51868	51836	51519	51741



Tabel A.2. Jumlah Pulsa Terhadap Perpindahan Posisi Horizontal Detektor

Posisi (mm)	Jumlah Pulsa pada Pengambilan Data ke-			Jumlah Pulsa Rata- rata
	1	2	3	
1	145	176	156	159
2	266	357	307	310
3	440	522	508	490
4	614	699	651	654.6666667
5	797	840	841	826
6	879	938	963	926.6666667
7	1067	1132	1116	1105
8	1230	1324	1287	1280.3333333
9	1323	1396	1372	1363.6666667
10	1499	1584	1541	1541.3333333
11	1680	1763	1736	1726.3333333
12	1775	1855	1879	1836.3333333
13	1962	2028	1992	1994
14	2115	2205	2160	2160
15	2229	2282	2278	2263
16	2466	2470	2438	2458
17	2569	2601	2597	2589
18	2672	2721	2763	2718.6666667
19	2853	2866	2904	2874.3333333
20	2908	3032	3045	2995
21	3040	3150	3213	3134.3333333
22	3168	3319	3330	3272.3333333
23	3289	3478	3517	3428
24	3417	3600	3689	3568.6666667
25	3548	3790	3784	3707.3333333
26	3735	3942	3966	3881
27	3903	4038	4134	4025
28	4006	4251	4236	4164.3333333
29	4180	4316	4427	4307.6666667
30	4345	4487	4588	4473.3333333
31	4468	4676	4681	4608.3333333
32	4627	4785	4867	4759.6666667
33	4780	4941	5029	4916.6666667
34	4898	5125	5122	5048.3333333
35	5069	5274	5311	5218
36	5224	5380	5476	5360
37	5351	5562	5563	5492
38	5516	5707	5720	5647.6666667

39	5669	5824	5905	5799.333333
40	5792	6014	6018	5941.333333
41	5949	6110	6197	6085.333333
42	6113	6272	6330	6238.333333
43	6232	6470	6457	6386.333333
44	6411	6656	6635	6567.333333
45	6556	6747	6780	6694.333333
46	6667	6917	6899	6827.666667
47	6837	7095	7077	7003
48	6994	7184	7231	7136.333333
49	7111	7350	7351	7270.666667
50	7284	7537	7534	7451.666667
51	7426	7627	7677	7576.666667
52	7562	7790	7812	7721.333333
53	7729	7921	7957	7869
54	7907	8054	8076	8012.333333
55	8046	8232	8215	8164.333333
56	8186	8384	8391	8320.333333
57	8365	8513	8477	8451.666667
58	8544	8689	8660	8631
59	8697	8853	8834	8794.666667
60	8814	8960	8972	8915.333333
61	8983	9122	9116	9073.666667
62	9103	9291	9264	9219.333333
63	9272	9396	9412	9360
64	9423	9580	9553	9518.666667
65	9528	9690	9710	9642.666667
66	9732	9831	9852	9805
67	9859	10016	9991	9955.333333
68	9991	10195	10159	10115
69	10155	10297	10300	10250.666667
70	10303	10455	10439	10399
71	10437	10608	10596	10547
72	10608	10725	10729	10687.333333
73	10743	10908	10888	10846.333333
74	10888	11049	11035	10990.666667
75	11062	11176	11187	11141.666667
76	11209	11353	11343	11301.666667
77	11336	11483	11487	11435.333333
78	11455	11604	11614	11557.666667
79	11632	11797	11750	11726.333333
80	11760	11932	11908	11866.666667
81	11960	12067	12060	12029

82	12069	12238	12215	12174
83	12215	12381	12355	12317
84	12396	12501	12509	12468.66667
85	12514	12687	12675	12625.33333
86	12631	12829	12845	12768.33333
87	12818	12961	12967	12915.33333
88	12966	13120	13111	13065.66667
89	13104	13290	13262	13218.66667
90	13280	13409	13411	13366.66667
91	13437	13580	13559	13525.33333
92	13560	13712	13719	13663.66667
93	13837	13857	13856	13850
94	14003	14020	13999	14007.33333
95	14159	14150	14172	14160.33333
96	14313	14298	14324	14311.66667
97	14427	14472	14460	14453
98	14589	14638	14603	14610
99	14718	14746	14739	14734.33333
100	14888	14914	14894	14898.66667
101	15020	15068	15009	15032.33333
102	15177	15193	15180	15183.33333
103	15332	15365	15334	15343.66667
104	15463	15498	15501	15487.33333
105	15613	15639	15623	15625
106	15772	15808	15773	15784.33333
107	15890	15947	15915	15917.33333
108	16061	16073	16053	16062.33333
109	16229	16251	16206	16228.66667
110	16368	16383	16347	16366
111	16510	16521	16516	16515.66667
112	16655	16685	16672	16670.66667
113	16765	16821	16817	16801
114	16926	16963	17059	16982.66667
115	17119	17122	17123	17121.33333
116	17265	17247	17275	17262.33333
117	17394	17408	17393	17398.33333
118	17550	17573	17533	17552
119	17717	17723	17721	17720.33333
120	17865	17850	17854	17856.33333
121	18006	18011	17992	18003
122	18153	18137	18156	18148.66667
123	18285	18356	18293	18311.33333
124	18444	18445	18439	18442.66667

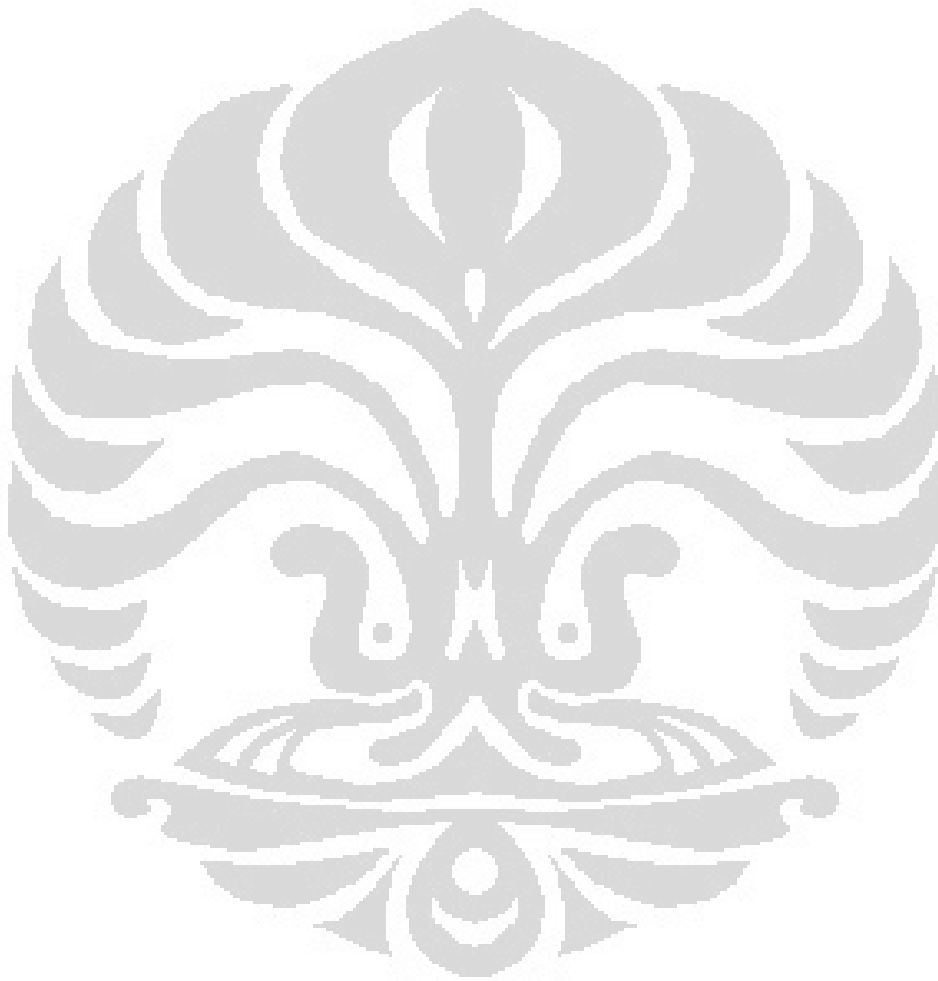
125	18575	18593	18587	18585
126	18727	18745	18711	18727.66667
127	18858	18898	18858	18871.33333
128	19034	19029	19024	19029
129	19166	19184	19160	19170
130	19337	19340	19323	19333.33333
131	19473	19474	19464	19470.33333
132	19609	19627	19610	19615.33333
133	19759	19795	19767	19773.66667
134	19927	19917	19925	19923
135	20041	20058	20064	20054.33333
136	20147	20217	20220	20194.66667
137	20347	20355	20347	20349.66667
138	20501	20517	20492	20503.33333
139	20660	20683	20644	20662.33333
140	20820	20806	20815	20813.66667
141	20957	20956	20953	20955.33333
142	21098	21120	21103	21107
143	21267	21260	21240	21255.66667
144	21384	21399	21394	21392.33333
145	21532	21548	21544	21541.33333
146	21697	21679	21691	21689
147	21825	21842	21844	21837
148	21992	21987	21988	21989
149	22111	22141	22137	22129.66667
150	22277	22301	22282	22286.66667
151	22430	22437	22437	22434.66667
152	22583	22577	22573	22577.66667
153	22727	22738	22713	22726
154	22892	22878	22857	22875.66667
155	23019	23005	23029	23017.66667
156	23160	23167	23156	23161
157	23307	23327	23325	23319.66667
158	23450	23468	23462	23460
159	23618	23632	23598	23616
160	23777	23788	23776	23780.33333
161	23920	23928	23923	23923.66667
162	24050	24074	24061	24061.66667
163	24201	24227	24213	24213.66667
164	24354	24387	24341	24360.66667
165	24496	24524	24507	24509
166	24657	24702	24644	24667.66667
167	24813	24818	24785	24805.33333

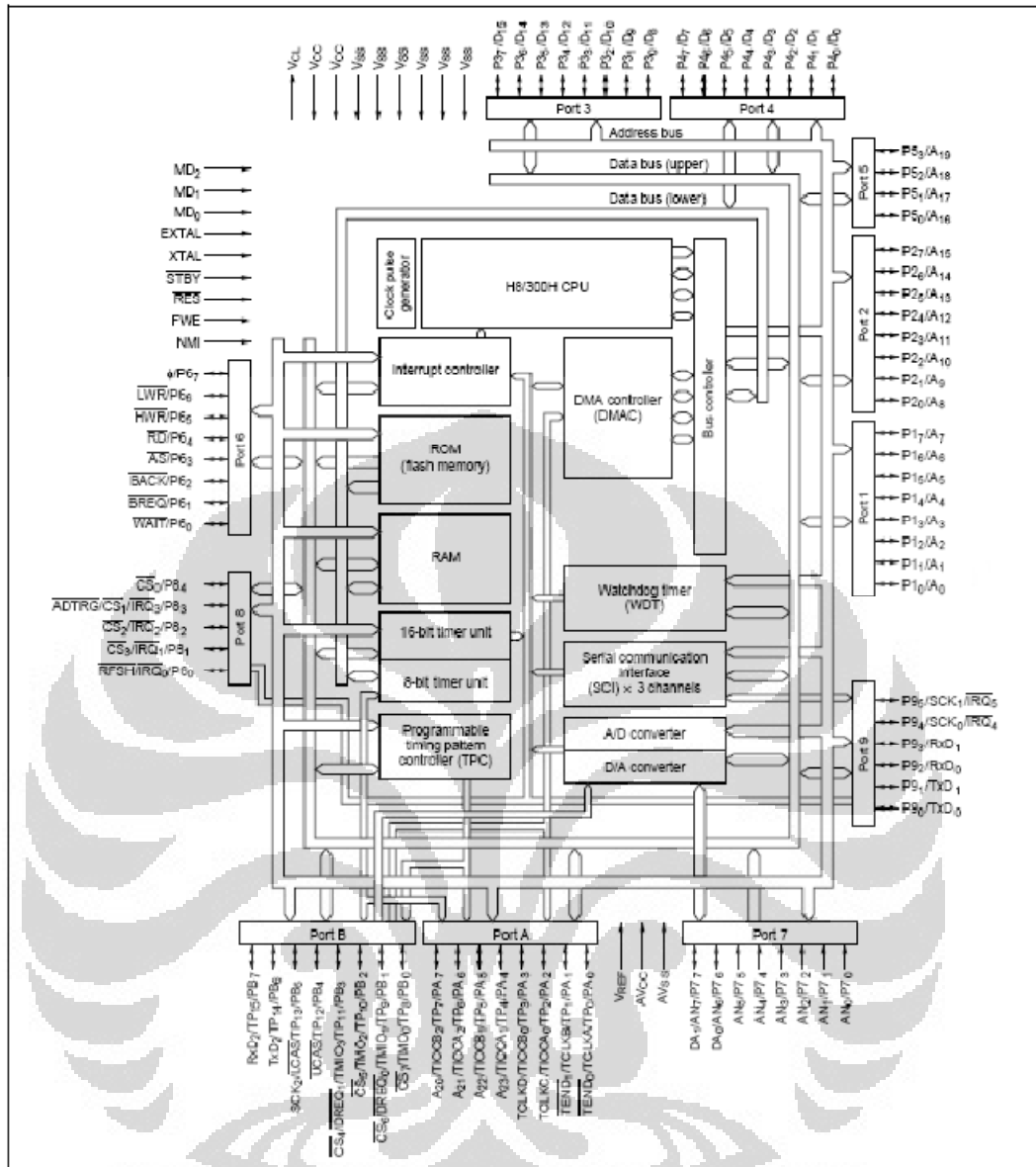
168	24932	24972	24931	24945
169	25089	25126	25097	25104
170	25228	25259	25249	25245.33333
171	25390	25405	25382	25392.33333
172	25549	25572	25539	25553.33333
173	25710	25695	25674	25693
174	25832	25854	25790	25825.33333
175	25988	26019	25982	25996.33333
176	26131	26156	26116	26134.33333
177	26278	26296	26256	26276.66667
178	26440	26458	26401	26433
179	26569	26590	26517	26558.66667
180	26721	26748	26701	26723.33333
181	26879	26900	26852	26877
182	27014	27029	27001	27014.66667
183	27147	27164	27154	27155
184	27302	27339	27312	27317.66667
185	27441	27471	27459	27457
186	27597	27611	27615	27607.66667
187	27754	27776	27753	27761
188	27893	27916	27889	27899.33333
189	28039	28072	28036	28049
190	28205	28238	28193	28212
191	28343	28381	28346	28356.66667
192	28439	28513	28501	28484.33333
193	28651	28659	28665	28658.33333
194	28778	28790	28798	28788.66667
195	28921	28936	28931	28929.33333
196	29092	29091	29082	29088.33333
197	29197	29212	29241	29216.66667
198	29377	29397	29388	29387.33333
199	29516	29516	29533	29521.66667
200	29678	29694	29667	29679.66667
201	29817	29835	29812	29821.33333
202	29984	29985	29965	29978
203	30112	30148	30100	30120
204	30263	30300	30254	30272.33333
205	30422	30472	30400	30431.33333
206	30556	30596	30554	30568.66667
207	30691	30727	30711	30709.66667
208	30875	30890	30851	30872
209	31002	31028	31003	31011
210	31141	31166	31154	31153.66667

211	31302	31327	31310	31313
212	31431	31472	31442	31448.33333
213	31587	31614	31617	31606
214	31759	31775	31743	31759
215	31904	31913	31882	31899.66667
216	32043	32068	32031	32047.33333
217	32209	32223	32180	32204
218	32318	32357	32322	32332.33333
219	32476	32507	32469	32484
220	32642	32673	32607	32640.66667
221	32775	32803	32756	32778
222	32937	32945	32907	32929.66667
223	33085	33108	33054	33082.33333
224	33218	33245	33227	33230
225	33373	33382	33373	33376
226	33541	33551	33509	33533.66667
227	33657	33692	33657	33668.66667
228	33835	33839	33819	33831
229	33960	34003	33967	33976.66667
230	34115	34134	34116	34121.66667
231	34252	34273	34252	34259
232	34462	34441	34420	34441
233	34538	34593	34578	34569.66667
234	34698	34722	34697	34705.66667
235	34841	34868	34864	34857.66667
236	34974	34985	35004	34987.66667
237	35119	35175	35152	35148.66667
238	35256	35320	35298	35291.33333
239	35404	35470	35446	35440
240	35571	35631	35596	35599.33333
241	35717	35760	35745	35740.66667
242	35857	35888	35886	35877
243	36011	36058	36068	36045.66667
244	36192	36234	36218	36214.66667
245	36306	36356	36388	36350
246	36469	36485	36531	36495
247	36588	36662	36653	36634.33333
248	36736	36787	36772	36765
249	36909	36956	36931	36932
250	37048	37136	37083	37089
251	37161	37227	37225	37204.33333
252	37278	37384	37358	37340
253	37456	37543	37489	37496

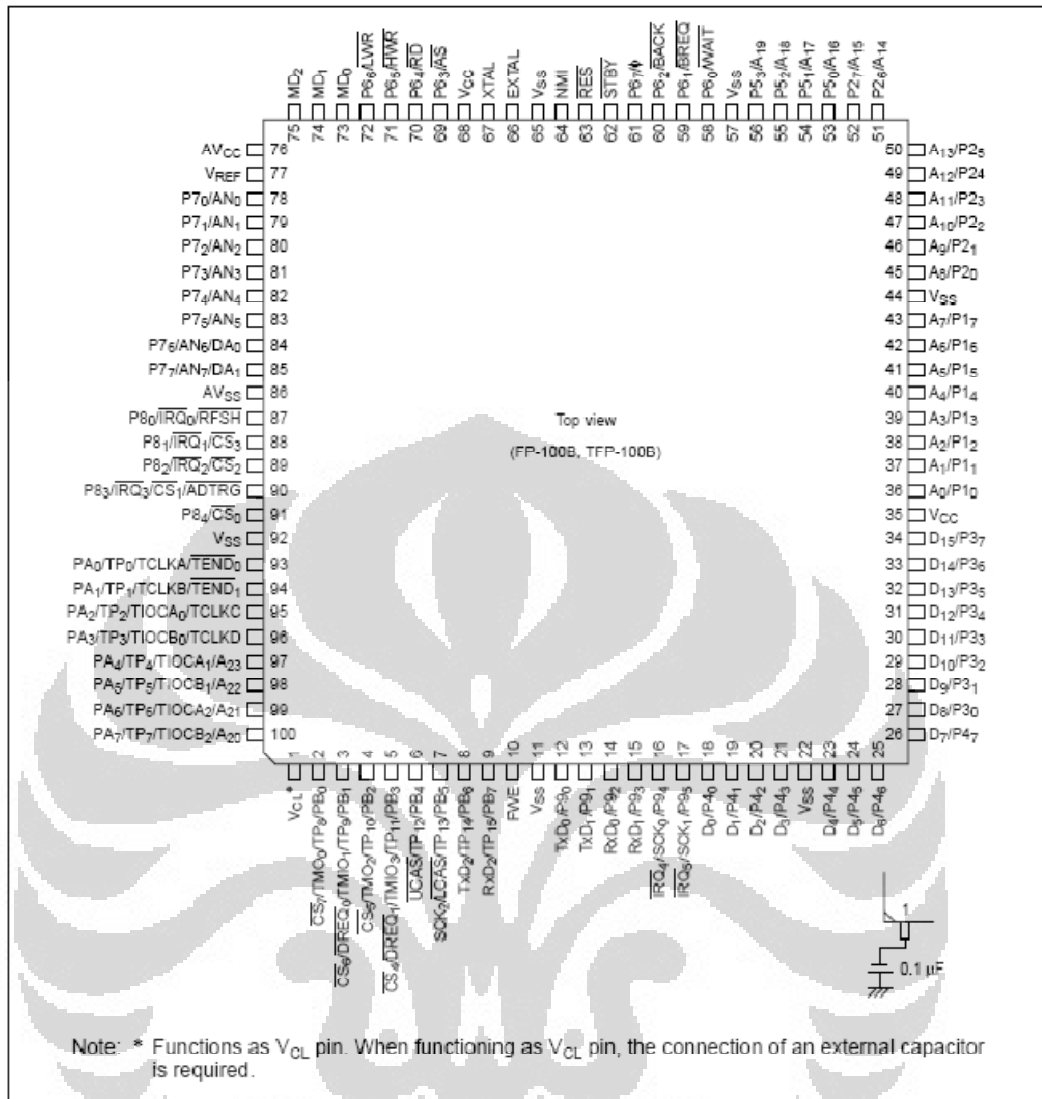
254	37601	37703	37639	37647.66667
255	37770	37875	37807	37817.33333
256	37892	38017	37977	37962
257	38037	38141	38143	38107
258	38197	38311	38262	38256.66667
259	38328	38454	38417	38399.66667
260	38481	38641	38558	38560
261	38628	38757	38712	38699
262	38769	38922	38865	38852
263	38936	39145	39033	39038
264	39081	39207	39172	39153.33333
265	39216	39352	39305	39291
266	39359	39500	39452	39437
267	39482	39631	39598	39570.33333
268	39589	39804	39752	39715
269	39749	39927	39895	39857
270	39911	40090	40042	40014.33333
271	40047	40251	40196	40164.66667
272	40188	40391	40351	40310
273	40328	40530	40490	40449.33333
274	40489	40681	40653	40607.66667
275	40635	40834	40767	40745.33333
276	40780	40978	40902	40886.66667
277	40919	41150	41076	41048.33333
278	41070	41278	41249	41199
279	41208	41415	41386	41336.33333
280	41507	41577	41524	41536
281	41655	41746	41683	41694.66667
282	41790	41865	41827	41827.33333
283	41938	42012	41980	41976.66667
284	42241	42173	42099	42171
285	42392	42318	42248	42319.33333
286	42533	42460	42401	42464.66667
287	42699	42588	42588	42625
288	42860	42752	42703	42771.66667
289	42951	42926	42848	42908.33333
290	43125	43045	42992	43054
291	43289	43213	43125	43209
292	43409	43352	43287	43349.33333
293	43582	43488	43442	43504
294	43734	43648	43586	43656
295	43891	43809	43734	43811.33333
296	44032	43934	43884	43950

297	44178	44081	44019	44092.66667
298	44315	44227	44168	44236.66667
299	44469	44370	44335	44391.33333
300	44632	44525	44472	44543





Gambar B.2 Blok diagram H8/3069F



Gambar B.3. Pin arrangement H8/3069

LAMPIRAN C

Datasheet Sensor Rotary Encoder

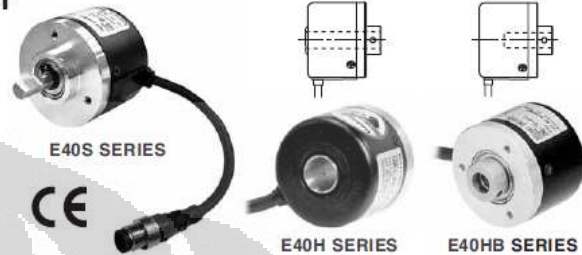
E40 Series

Diameter ϕ 40mm Shaft type/Hollow type/Built-in type
Incremental Rotary encoder

■ Features

- Easy installation at narrow space
- Small moment of inertia
- Power supply :
5VDC, 12-24VDC \pm 5%
- Various output types

⚠ Please read "Caution for your safety" in operation manual before using.



■ Ordering information

E40	[H]	8	5000	3	N	24		
Series	Shaft type	Hollow type	Pulse/1 Revolution	Output phase	Output	Power supply	Cable	
S: Shaft type H: Hollow type HB: Hollow built-in type	External	Inner	Refer to resolution	2: A, B 3: A, B, Z 4: A, \bar{A} , B, \bar{B} 6: A, \bar{A} , B, \bar{B} , Z, \bar{Z}	T: Totem pole output N: NPN open collector output V: Voltage output L: Line driver output(*)	5 : 5VDC \pm 5% 24: 12-24VDC \pm 5%	No mark: Normal type (*) C: Cable outgoing connector type	
	6: ϕ 6mm 8: ϕ 8mm	6: ϕ 6mm 8: ϕ 8mm 10: ϕ 10mm 12: ϕ 12mm						
*Standard : E40S6-[PULSE]-3-N-24 E40H8-[PULSE]-3-N-24 E40HB8-[PULSE]-3-N-24			*Standard: A, B, Z			*The power of Line driver is only for 5VDC		*Cable length : 250mm

■ Specifications

Item	Diameter ϕ 40mm shaft/hollow shaft/built-in type of incremental rotary encoder		
Resolution (P/R)	(Note1)	*1, *2, *5, 10, *12, 15, 20, 23, 25, 30, 35, 40, 45, 50, 60, 75, 100, 120, 150, 192, 200, 240, 250, 256, 300, 360, 400, 500, 512, 600, 800, 1000, 1024, 1200, 1500, 1800, 2000, 2048, 2500, 3000, 3600, 5000 (Not indicated resolution is customizable.)	
Electrical specification	Output phase	A, B, Z phase (Line driver : A, \bar{A} , B, \bar{B} , Z, \bar{Z} phase)	
	Phase difference of output	Phase difference between A and B : $\frac{T}{4} \pm \frac{T}{8}$ (T=1 cycle of A phase)	
	Control output	Totem pole output	• Low \Rightarrow Load current: Max. 30mA, Residual voltage : Max. 0.4VDC • High \Rightarrow Load current: Max. 10mA, Output voltage (Power supply 5VDC): Min. (Power supply-2.0)VDC, Output voltage (Power supply 12-24VDC): Min. (Power supply-3.0)VDC
		NPN open collector output	Load current : Max. 30mA, Residual voltage : Max. 0.4VDC
		Voltage output	Load current : Max. 10mA, Residual voltage : Max. 0.4VDC
		Line driver output	Low \Rightarrow Load current : Max. 20mA, Residual : Max. 0.5VDC High \Rightarrow Load current : Max. 20mA, Output voltage : Min. 2.5VDC
	Response time (Rise/Fall)	Totem pole output	Max. 1 μ s
		NPN open collector output	Max. 1 μ s
		Voltage output	Max. 1 μ s
		Line driver output	Max. 0.5 μ s
Max. Response frequency	300kHz		
Power supply	• 5VDC \pm 5% (Ripple P-P: Max. 5%) • 12-24VDC \pm 5% (Ripple P-P: Max. 5%)		
Current consumption	Max. 80mA (disconnection of the load), Line driver output: Max. 50mA (disconnection of the load)		
Insulation resistance	Min. 100M Ω (at 500VDC mega)		
Dielectric strength	750VAC 50/60Hz for 1 minute (Between all terminals and case)		
Connection	Cable outgoing type, 200mm cable outgoing connector type		
Mechanical specification	Starting torque	Shaft Type : Max. 40gf \cdot cm (0.004N \cdot m), Hole Type : Max. 50gf \cdot cm (0.005N \cdot m)	
	Rotor inertia	Max. 40g \cdot cm ² (4 \times 10 ⁻⁴ kg \cdot m ²)	
	Shaft loading	Radial : Max. 2kgf, Thrust : Max. 1kgf	
	Max. allowable revolution	(Note2)	5000rpm
Vibration	1.5mm amplitude at frequency of 10 ~ 55Hz in each of X, Y, Z directions for 2 hours		
Shock	Max. 50G		
Ambient temperature	-10 ~ 70°C (at non-freezing status), Storage: -25 ~ 85°C		
Ambient humidity	35-85%RH, Storage: 35-90%RH		
Protection	IP50 (IEC standard)		
Cable	ϕ 5mm, 5P, Length : 2m, Shield cable (Line driver output : ϕ 5mm, 8P)		
Accessory	• Shaft type: ϕ 6mm coupling standard, ϕ 8mm coupling (Sold separately) • Hole type : Bracket		
Unit weight	Approx. 160g		
Approval	CE (Except for Line driver output)		

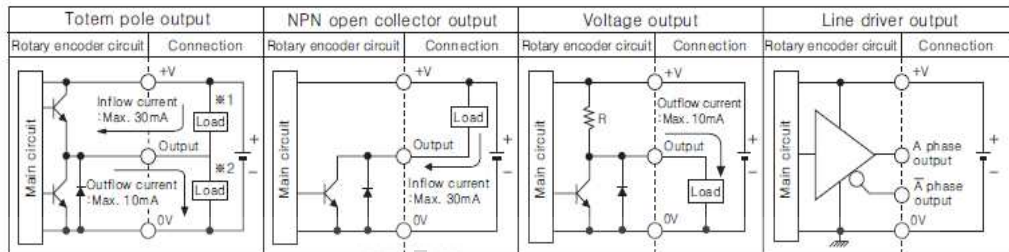
* (Note1) *1 pulse is only for A, B phase (Line Driver output is for A, \bar{A} , B, \bar{B} phase)

* (Note2) Max. allowable revolution \geq Max. response revolution [Max. response revolution (rpm) = $\frac{\text{Max. response frequency}}{\text{Resolution}} \times 60 \text{ sec}$]

Gambar C.1 Jenis dan spesifikasi sensor rotary encoder

Incremental ϕ 40mm Shaft/Hollow Shaft/Built-in Type

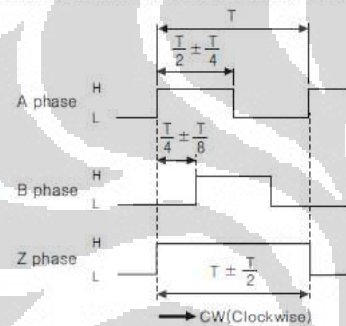
Control output diagram



⇒ Totem pole output type can be used for NPN open collector output type (*1) or Voltage output type (*2).
 ⇒ All output circuits of A, B, Z phase is same. (Line driver output is A, \bar{A} , B, \bar{B} , Z, \bar{Z})

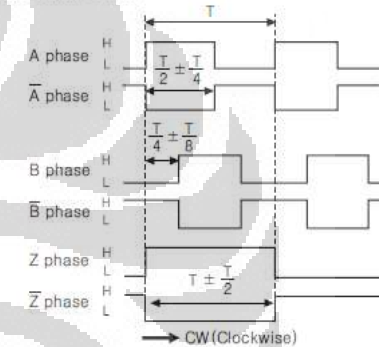
Output waveform

● Totem pole output / NPN open collector output / Voltage output



*CW : As viewed from the shaft

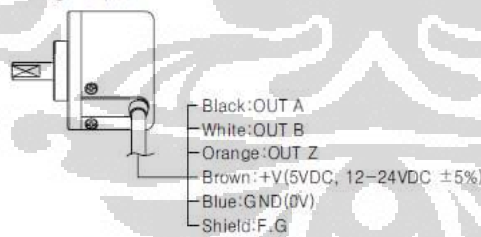
● Line driver output



Connections

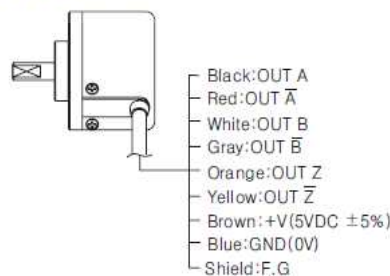
Normal type

● Totem pole output / NPN open collector output / Voltage output



*Unused wires must be insulated.
 *The metal case and shield wire of encoder should be grounded (F.G).

● Line driver output



Cable outgoing connector type

● Totem pole output / NPN open collector output / Voltage output



Totem pole output NPN open collector output Voltage output			Line driver output		
Pin No	Function	Cable color	Pin No	Function	Cable color
①	OUT A	Black	①	OUT A	Black
②	OUT B	White	②	OUT \bar{A}	Red
③	OUT Z	Orange	③	+V	Brown
④	+V	Brown	④	GND	Blue
⑤	GND	Blue	⑤	OUT B	White
⑥	F.G	Shield	⑥	OUT \bar{B}	Gray
			⑦	OUT Z	Orange
			⑧	OUT \bar{Z}	Yellow
			⑨	F.G	Shield

*F.G(Field Ground): It should be grounded separately.

- (A) Counter
- (B) Timer
- (C) Temp. controller
- (D) Power controller
- (E) Panel meter
- (F) Tacho/ Speed/ Pulse meter
- (G) Display unit
- (H) Sensor controller
- (I) Switching power supply
- (J) Proximity sensor
- (K) Photo electric sensor
- (L) Pressure sensor
- (M) Rotary encoder**
- (N) Stepping motor & Driver & Controller
- (O) Graphic panel
- (P) Field network device
- (Q) Production stoppage models & replacement

Gambar C.2. Koneksi dan output rotary encoder

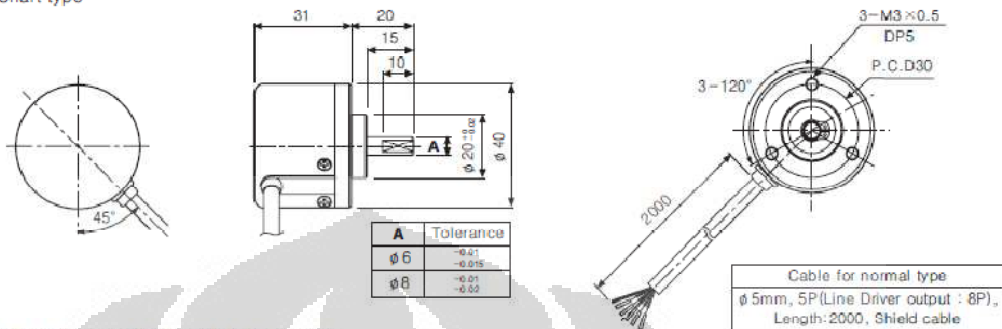
E40 Series

Dimensions

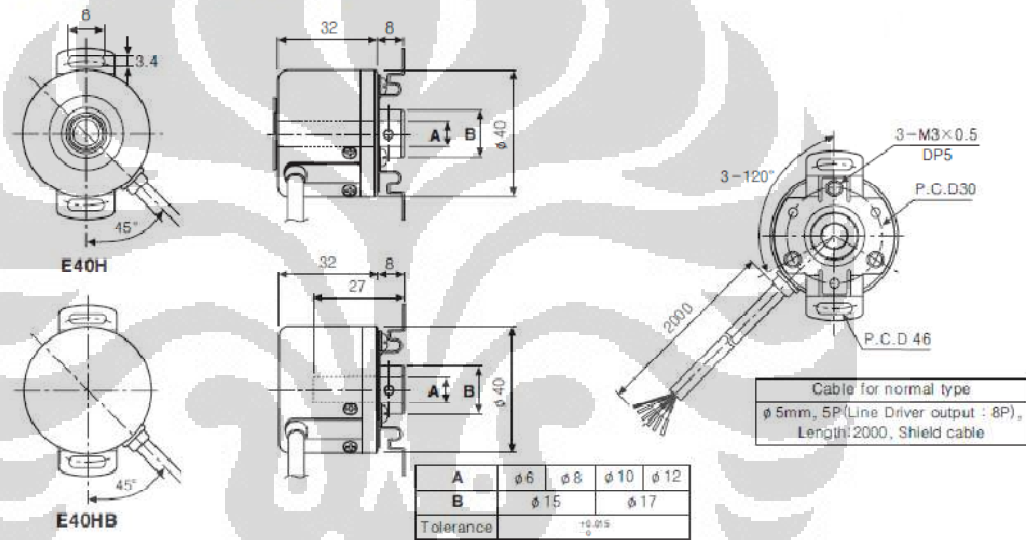
Normal type

Shaft type

(Unit:mm)

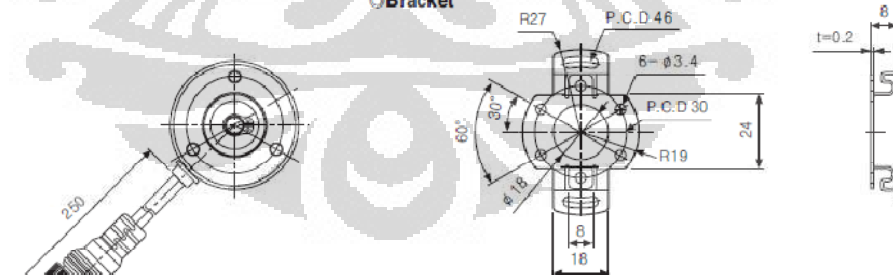


Hollow shaft / Hollow shaft built-in type



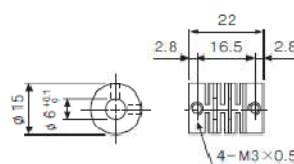
Cable outgoing connector type

Bracket

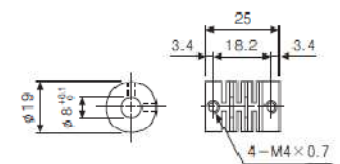


Coupling (E40S)

φ 6 Coupling



φ 8 Coupling



* Connector cable is customizable and see M-57 for specifications.

Gambar C.3 Dimensi sensor rotary encoder

LAMPIRAN E

Script utama program bahasa C (main.c)

```

#include "interrupt.h"
#include "ioh83069.h"
#include "uart.h"
#include <stdio.h>
#include <stdlib.h>

long int counter = 0;
long int counter2 = 0;
long int posisi, ketinggian;
long int ncounter1, ncounter2, ncounter_1, ncounter_2,
nvalue;
unsigned short cmdln;
char data, flag, buf[50], cmdbuf[20];

void ISR_IRQ0(void)
{
    ISR &= ~(1 << IRQ0F);

    //check data proximiti kedua
    //Jika bernilai 0 maka counter naik
    if (P1DR.BIT.BN & (1 << PIN5)) {
        counter--;
    } else {
        counter++;
    }
}

void ISR_IRQ1(void)
{
    ISR &= ~(1 << IRQ1F);

    //check data proximiti kedua
    //Jika bernilai 0 maka counter naik
    if (P1DR.BIT.BN & (1 << PIN4)) {
        counter2--;
    } else {
        counter2++;
    }
}

void TISRA_IMIA0(void)
{
    TISRA &= ~(1 << IMFA0); //clear flag
}

void TISRA_IMIA1(void)

```

```

{
    TISRA &= ~(1 << IMFA1); //clear flag
}

void write(void)
{
    sprintf(&buf[0], "%ld,%ld,%ld,%ld,%ld.%ld,%ld.%ld
\r\n", counter, ncounter1, counter2, ncounter2, ketinggian, (((cou
nter-(ketinggian*170))*100)/170), posisi, (((counter2-
(posisi*148))*100)/148)) ;
    uart_str(&buf[0]);
}

void down(void)
{
    GRA0 = 62500; //20 ms
    GRB0 = 3000; //1.3 ms (CW)
    TSTR |= (1 << STR0); //mulai counting
    TISRA |= (1 << IMIEA0); //Nyalakan interrupt IMIA
}

void up(void)
{
    GRA0 = 62500; //20 ms
    GRB0 = 6250 ; //1.7 ms (CCW)
    TSTR |= (1 << STR0); //mulai counting
    TISRA |= (1 << IMIEA0); //Nyalakan interrupt IMIA
}

void right(void)
{
    GRA1 = 62500; //20 ms
    GRB1 = 3000; //1.3 ms (CW)
    TSTR |= (1 << STR1); //mulai counting
    TISRA |= (1 << IMIEA1); //Nyalakan interrupt IMIA
}

void left(void)
{
    GRA1 = 62500; //20 ms
    GRB1 = 6250 ; //1.7 ms (CCW)
    TSTR |= (1 << STR1); //mulai counting
    TISRA |= (1 << IMIEA1); //Nyalakan interrupt IMIA
}

void ISR_RXI1 (void)
{
    data = RDR1;
    SSR1 &= ~(1 << RDRF);

    if (data != 13) { //kalau enter ga ditekan

```

```

        uart_ch(data);
        cmdbuf[cmdln] = data;
        cmdln++;

    } else {
        uart_strP("\r\n");
        cmdbuf[cmdln] = '\0';

        if (cmdbuf[0] != '$') {
            uart_strP("Invalid Command\r\n");

        } else{
            if (cmdln > 3){
                nvalue = atoi(&cmdbuf[3]); //convert
string to integer
                //i = nvalue;
                if ((cmdbuf[1] == 'D') || (cmdbuf[1]
== 'd')) {
                    ncounter1 = counter +
(nvalue*170);
                    flag = 1;
                } else if ((cmdbuf[1] == 'U') ||
(cmdbuf[1] == 'u')) {
                    ncounter1 = counter -
(nvalue*170);
                    flag = 2;
                } else if ((cmdbuf[1] == 'R') ||
(cmdbuf[1] == 'r')) {
                    ncounter2 = counter2 +
(nvalue*148);
                    flag = 3;
                } else if ((cmdbuf[1] == 'L') ||
(cmdbuf[1] == 'l')) {
                    ncounter2 = counter2 -
(nvalue*148);
                    flag = 4;
                }
            }
        }

        cmdln = 0;
    }
}

int main()
{
    P1DDR = 0x00;

    uart_init();

```

```

uart_strP("Inisialisasi Uart\r\n");

IER |= (1 << IRQ0E);
ISCR |= (1 << IRQ0SC);

IER |= (1 << IRQ1E);
ISCR |= (1 << IRQ1SC);

TCR0_16 |= 0x23; //prescaler 8 --> 25MHz/8 dan Clear
counting bila GRA compare match
TCR1_16 |= 0x23;
TIOR0 |= 0x99; //channel 0 : output 0 saat GRB compare
dan output 1 saat GRA compare
TIOR1 |= 0x99;
TMDR |= 0x03; //pwm mode 0 & 1
TCNT0_16 = 0x0000; //counter mulai dari 0x0000
TCNT1_16 = 0x0000;
TSTR &= 0x00; //belum mulai counting

cmdln = 0;
flag = 0;
nvalue = 0;

sei();

while(1)
{
    while (flag == 0){
        GRA0 = 62500;
        GRB0 = 3900;
        TSTR |= (1 << STR0); //mulai counting
        TISRA |= (1 << IMIEA0); //Nyalakan
interrupt IMIA
    }

    while (flag == 5){
        GRA1 = 62500;
        GRB1 = 3888;
        TSTR |= (1 << STR1); //mulai counting
        TISRA |= (1 << IMIEA1); //Nyalakan
interrupt IMIA
    }

    if (flag == 1) {
        ncounter_1 = ncounter1;
        while (counter <= ncounter_1) {
            down();
            ketinggian = counter/170;
            write();
        }
        flag=0;
        write();
    }
}

```

```
        continue;

    } else if (flag == 2) {
        ncounter_1 = ncounter1 - (nvalue*4);
        while (counter >= ncounter_1) {
            up();
            ketinggian = counter/170;
            write();
        }
        flag=0;
        write();
        continue;

    } else if (flag == 3) {
        ncounter_2 = ncounter2;
        while (counter2 <= ncounter_2) {
            right();
            posisi = counter2/148;
            write();
        }
        flag = 5;
        write();
        continue;

    } else if (flag == 4) {
        ncounter_2 = ncounter2 - (nvalue*4);
        while (counter2 >= ncounter_2) {
            left();
            posisi = counter2/148;
            write();
        }
        flag = 5;
        write();
        continue;
    }
}
return 0;
}
```

LAMPIRAN F

Program utama untuk membuat GUI melalui Python

```

import sys
from PySide.QtCore import *
from PySide.QtGui import *
from ui_FantomAir3 import Ui_Dialog
from serialcom import SerialCom
from math import *

class Dialog (QDialog,Ui_Dialog):
    def __init__ (self,parent=None):
        QDialog.__init__(self,parent)
        self.setupUi (self)

        self.startButton.clicked.connect(self.startHandler)

        self.resetButton.clicked.connect(self.resetHandler)

        self.startButton_2.clicked.connect(self.startHandler_2
)
        self.resetButton_2.clicked.connect(self.resetHandler_2
)

        self.com = SerialCom(self)

        self.com.readyRead.connect(self.serialHandler)

        self.com.start()

        self.lastvalue = 0
        self.lastvalue_2 = 0

    @Slot(int)
    def startHandler(self):
        value = self.horizontalSlider.value()
        if self.lastvalue > value :
            pindah1 = self.lastvalue - value
            self.com.write("$L %s\r" %str(pindah1))
            self.lastvalue = value
        elif self.lastvalue < value :
            pindah2 = value - self.lastvalue
            self.com.write("$R %s\r" %str(pindah2))
            self.lastvalue = value

    @Slot(int)
    def resetHandler(self):
        self.horizontalSlider.setSliderPosition(0)
        value = self.horizontalSlider.value()

```



```

        pindah2 = self.lastvalue - value
        self.com.write("$L %s\r" %str(pindah2))

@Slot(int)
def startHandler_2(self):
    value_2 = self.horizontalSlider_2.value()
    if self.lastvalue_2 < value_2 :
        pindah1 = value_2 - self.lastvalue_2
        self.com.write("$D %s\r" %str(pindah1))
        self.lastvalue_2 = value_2
    elif self.lastvalue_2 > value_2 :
        pindah2 = self.lastvalue_2 - value_2
        self.com.write("$U %s\r" %str(pindah2))
        self.lastvalue_2 = value_2
    #self.textEdit.append(self.lineEdit.text())
    #self.lineEdit.clear()

@Slot(int)
def resetHandler_2(self):
    self.horizontalSlider_2.setSliderPosition(0)
    value_2 = self.horizontalSlider_2.value()
    pindah2 = self.lastvalue_2 - value_2
    self.com.write("$U %s\r" %str(pindah2))

#@Slot(int)
#def startHandler_3(self):
#    pass

#@Slot(int)
#def resetHandler_3(self):
#    pass

@Slot(str)
def serialHandler(self,data):
    tmp = data.split(',')

    counter,ncounter,counter2,ncounter2,ketinggian,posisi
= tmp[0],tmp[1],tmp[2],tmp[3],tmp[4],tmp[5]
    self.ketinggianEdit.setText(ketinggian)
    self.posisiEdit.setText(posisi)

def closeEvent(self,event):
    self.com.stop()

if __name__=='__main__':
    app = QApplication(sys.argv)
    dlg = Dialog()
    dlg.show()
    sys.exit(app.exec_())

```