



UNIVERSITAS INDONESIA

**PENGEMBANGAN SISTEM KONTROL MESIN *RAPID PROTOTYPING*
BERBASIS *FUSED DEPOSITION MODELING* (FDM)**

SKRIPSI

**RENDI KURNIAWAN
0606073480**

**FAKULTAS TEKNIK
DEPARTEMEN TEKNIK MESIN
DEPOK
DESEMBER 2010**



UNIVERSITAS INDONESIA

**PENGEMBANGAN SISTEM KONTROL MESIN *RAPID PROTOTYPING*
BERBASIS *FUSED DEPOSITION MODELING* (FDM)**

SKRIPSI

**Diajukan sebagai salah satu syarat
Untuk memperoleh gelar sarjana teknik**

**RENDI KURNIAWAN
0606073480**

**FAKULTAS TEKNIK
DEPARTEMEN TEKNIK MESIN
DEPOK
DESEMBER 2010**

HALAMAN PERNYATAAN ORISIONALITAS

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

Nama : Rendi Kurniawan

NPM : 0606073480

Tanda Tangan :

Tanggal : 22 Desember 2010

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama : Rendi Kurniawan

NPM : 0606073480

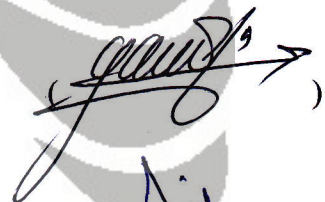
Program Studi : Teknik Mesin

Judul Skripsi : PENGEMBANGAN SISTEM KONTROL MESIN
*RAPID PROTOTYPING BERBASIS FUSED
DEPOSITION MODELING (FDM)*


Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian dari persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi, Teknik Mesin Fakultas Teknik, Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Dr.Ir. Gandjar Kiswanto, M.Eng.



Penguji : Dr.Ir. Ario Sunar Baskoro, ST.,MT.,MEng.



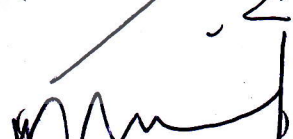
Penguji : Dr.Ir. Danardono AS.



Penguji : Ir. Hendri D.S. Budiono, M.Eng.



Penguji : Ir. Henky S. Nugroho, MT.



Ditetapkan di : Depok

Tanggal : 22 Desember 2010

ABSTRAK

Nama : Rendi Kurniawan
Program Studi : Teknik Mesin
Judul : Pengembangan Sistem Kontrol Mesin *Rapid Prototyping*
Berbasis *Fused Deposition Modeling*

Fused Deposition Modeling adalah metode pembuatan prototipe dengan proses pelelehan material termoplastik dengan menggunakan mekanisme ekstruder. Kemudian proses pembuatan prototipe melalui proses lapis per lapis. Penelitian ini bertujuan untuk mengembangkan mesin RPFDM (*Rapid Prototyping berbasis Fused Deposition Modeling*) dengan harga yang rendah dan sistem kontrol yang *robust*. Sistem kontrol yang dikembangkan berbasis sistem *embeded* untuk mengontrol pergerakan mekanisme mesin maupun sistem secara keseluruhan. Sistem kontrol menerima data dari pengguna melalui komputer (PC), kemudian sistem kontrol akan mengeksekusi dan memerintahkan driver motor agar menggerakkan motor stepper pada setiap sumbu dan motor DC pada ekstruder *nozzle*. Sistem kontrol terdiri dari empat buah mikrokontroller, empat buah driver motor, termokopel *amplifier*, dan LCD.

Kata kunci:

- Sistem kontrol, *rapid prototyping*

ABSTRACT

Name : Rendi Kurniawan
Study Program : Mechanical Engineering
Title : Development of Control System Rapid Prototyping Machine based on Fused Deposition Modeling

Fused Deposition Modeling is method for rapidly making prototype by liquefier process of thermoplastic material with using extruder mechanism. Then, the prototyping process is done layer by layer. This research aims to develop a low cost rapid prototyping machine and robust control system. The control system is developed based on embedded system to control the movement of the mechanism machine or system in general. Control system gets data from user using personal computer (PC), then control system will execute the data and ordering motor driver in order that moving stepper motor of each axis and DC motor of extruder nozzle. Control system consists of four microcontrollers, four driver motor, thermocouple amplifier, and LCD.

Keywords:

- Control system, rapid prototyping

UCAPAN TERIMA KASIH

Puji syukur saya panjatkan kehadiran Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk memperoleh gelar Sarjana Teknik Jurusan Teknik Mesin pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, penyusunan skripsi ini sangatlah sulit bagi saya. Oleh karena itu, saya mengucapkan terima kasih kepada :

1. Dr. Ir. Gandjar Kiswanto, M.Eng selaku dosen pembimbing yang telah meluangkan waktunya selama proses pengerjaan penelitian.
2. Kedua orang tua dan kakak saya yang selalu memberikan dukungan semangat dan materi.
3. Dr. Ir. Harinaldi, M.Eng selaku kepala Departemen Teknik Mesin.
4. Teguh Santoso, Ahmad Zakiyudin, dan Jediel yang telah memberikan bantuan tenaga dan pengetahuan, terutama tentang elektronika.
5. Teman-teman tim penelitian *rapid prototyping*, Andri. S, A. Saroni, Hadi. M, dan Sriyanto.
6. Teman-teman angkatan 2006, yang tidak bisa disebutkan satu persatu.

Semoga Allah S.W.T membalas semua kebaikan dan amal pihak-pihak yang telah membantu selama proses pengerjaan penelitian, dan hasil penelitian yang telah tercapai semoga dapat terus dikembangkan dan bermanfaat untuk kemajuan ilmu pengetahuan Indonesia.

Jakarta, Desember 2010

Penulis

DAFTAR ISI

HALAMAN PERNYATAAN ORISIONALITAS.....	i
HALAMAN PENGESAHAN.....	ii
ABSTRAK.....	iii
ABSTRACK.....	iv
UCAPAN TERIMA KASIH.....	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	viii
DAFTAR TABEL.....	xii
DAFTAR LAMPIRAN.....	xiii
DAFTAR ISTILAH.....	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Tujuan penelitian.....	4
1.3 Perumusan masalah.....	4
1.4 Pembatasan masalah.....	6
1.5 Metodologi penelitian.....	6
1.6 Sistematika penulisan.....	9
BAB 2 SISTEM RAPID PROTOTYPING.....	11
2.1 Teknologi rapid prototyping.....	11
2.2 Proses rapid prototyping.....	12
2.3 Rapid prototyping teknik.....	16
2.4 Material termoplastik.....	22
BAB 3 TINJAUAN SISTEM MEKANIKAL DAN EKSTRUDER RPFDM.....	27
3.1 Desain mesin RPFDM.....	27
3.2 Linier guideway.....	29
3.3 Ballscrew.....	30
3.4 Motor stepper.....	32
3.5 Konsep mekanisme ekstruder.....	38
3.6 Konstruksi mekanisme ekstruder.....	39
BAB 4 PENGEMBANGAN SISTEM KONTROL.....	45
4.1 Sensor.....	46
4.2 Kendali digital.....	51
4.3 Kontrol unit.....	87
BAB 5 PENGEMBANGAN PERANGKAT LUNAK MCU PWM DAN TEMPERATUR KONTROL.....	88
5.1 Algoritma pada MCU PWM kontrol.....	89
5.2 Algoritma pada MCU temperatur kontrol.....	92

BAB 6 PENGUJIAN DAN ANALISA SISTEM KONTROL DRIVER MOTOR STEPPER	96
6.1 Tujuan pengujian.....	96
6.2 Peralatan pengujian.....	96
6.3 Metode pengujian.....	97
6.4 Hasil pengujian.....	106
6.5 Analisa hasil.....	111
BAB 7 PENGUJIAN DAN ANALISA SISTEM KONTROL TERMOKOPEL AMPLIFIER	113
7.1 Tujuan pengujian.....	113
7.2 Peralatan pengujian.....	113
7.3 Metode pengujian.....	114
7.4 Grafik hasil pengukuran.....	115
7.5 Analisa hasil grafik.....	119
BAB 8 PENGUJIAN DAN ANALISA PRODUK AKHIR	121
8.1 Tujuan pengujian.....	121
8.2 Peralatan pengujian.....	121
8.3 Metode pengujian.....	122
8.4 Hasil grafik pengujian.....	127
8.5 Analisa grafik.....	134
8.6 Hasil produk akhir.....	138
8.7 Analisa hasil produk akhir.....	143
BAB 9 KEIMPULAN DAN SARAN PENELITIAN LEBIH LANJUT	148
9.1 Kesimpulan.....	148
9.2 Saran penelitian lebih lanjut.....	148
DAFTAR PUSTAKA	150
LAMPIRAN 1	155
LAMPIRAN 2	158
LAMPIRAN 3	160

DAFTAR GAMBAR

Gambar 1.1 Penggunaan <i>rapid prototyping</i> di dunia.....	1
Gambar 1.2 Ilustrasi proses <i>additive manufacturing layer by layer</i> pada <i>rapid prototyping</i>	3
Gambar 1.3 Arsitektur mesin <i>rapid prototyping</i>	5
Gambar 1.4 Flowchart perancangan sistem kontrol.....	8
Gambar 2.1 <i>User interface</i> SolidWorks 2009.....	13
Gambar 2.2 STL ASCII Format.....	14
Gambar 2.3 Metode <i>slicing</i> pada STL file.....	15
Gambar 2.4 Skema diagram <i>Stereolithography</i>	17
Gambar 2.5 Skema diagram <i>Laminated Object Manufacturing</i>	19
Gambar 2.6 Skema diagram <i>Selective Laser Sintering</i>	19
Gambar 2.7 Skema diagram <i>Fused Deposition Modeling</i>	20
Gambar 2.8 Skema diagram <i>Solid Ground Curing</i>	21
Gambar 2.9 Skema diagram <i>3-D Ink-Jet Printing</i>	22
Gambar 3.1 Mesin RPFDM tiga <i>axis</i>	27
Gambar 3.2 Desain mesin RPFDM.....	28
Gambar 3.3 Dimensi mesin 450x400x449.7.....	28
Gambar 3.4 Tipe MGN7C HIWIN.....	29
Gambar 3.5 Dimensi tipe MGN.....	30
Gambar 3.6 Nut FSI R8-2T3 tampak tiga dimensi.....	31
Gambar 3.7 Dimensi dan spesifikasi FSI R8-2T3.....	31
Gambar 3.8 Motor <i>variable reluctance</i>	32
Gambar 3.9 Unipolar motor dan <i>full step</i>	33
Gambar 3.10 Gerakan <i>full step</i> torsi lebih dan <i>half step</i>	34
Gambar 3.11 Motor bipolar dan pergerakan <i>full step</i>	35
Gambar 3.12 Pergerakan <i>half step</i> motor bipolar.....	35
Gambar 3.13 Lima fasa motor stepper.....	36
Gambar 3.14 Motor stepper tipe AIK-S543(W).....	37
Gambar 3.15 Dimensi motor stepper lima fasa.....	37
Gambar 3.16 Torsi vs kecepatan motor stepper lima fasa.....	38
Gambar 3.17 Skema diagram FDM.....	38
Gambar 3.18 Konstruksi mekanisme ekstruder.....	39
Gambar 3.19 Desain dari mekanisme ekstruder.....	40
Gambar 3.20 Dimensi dari <i>assembly</i> mekanisme ekstruder.....	40
Gambar 3.21 Mekanisme <i>feeder wheels</i>	41
Gambar 3.22 Desain dari <i>driver gear</i>	42
Gambar 3.23 Desain dari komponen barrel.....	43
Gambar 3.24 Dimensi isolator PTFE.....	44
Gambar 3.25 Kawat nikrom dan komponen isolator.....	44
Gambar 4.1 Sistem kontrol mesin RPFDM.....	45

Gambar 4.2 Percobaan Thomas Johann Seebeck.....	46
Gambar 4.3 Termokopel dengan dua buah material yang sama.....	48
Gambar 4.4 Rangkaian dasar dan pengukuran termokopel.....	48
Gambar 4.5 Termokopel tipe-K.....	49
Gambar 4.6 <i>Limit switch</i>	50
Gambar 4.7 Rangkaian skematik <i>limit switch</i>	50
Gambar 4.8 Kristal 16 MHz pada ATmega 2560 dan ATmega 16.....	52
Gambar 4.9 Gelombang sinyal PWM.....	52
Gambar 4.10 <i>Duty cycle 75%</i>	53
Gambar 4.11 Bentuk gelombang <i>FastPWM</i>	53
Gambar 4.12 Gelombang PWM dengan OCR0 = 64.....	54
Gambar 4.13 Aplikasi ADC.....	55
Gambar 4.14 Mikrokontroler ATmega 2560.....	56
Gambar 4.15 Spesifikasi mikrokontroler ATmega 2560.....	57
Gambar 4.16 <i>Pinout</i> ATmega 2560.....	58
Gambar 4.17 Diagram blok dari ATmega 2560.....	59
Gambar 4.18 Desain sistem minimum ATmega 2560.....	59
Gambar 4.19 Mikrokontroler ATmega 16.....	62
Gambar 4.20 Diagram blok dari ATmega 16.....	63
Gambar 4.21 Spesifikasi mikrokontroler ATmega 16.....	64
Gambar 4.22 <i>Pinout</i> dari ATmega 16.....	65
Gambar 4.23 Skema dari desain sistem minimum ATmega 16.....	65
Gambar 4.24 Desain PCB ATmega 16.....	66
Gambar 4.25 Pergantian polaritas pada H-Bridge.....	66
Gambar 4.26 Rangkaian pembatas arus dengan <i>power resistor</i>	68
Gambar 4.27 Rangkaian dasar <i>chopping current</i>	69
Gambar 4.28 Keadaan aktifitas <i>switching</i> pada koil motor stepper.....	69
Gambar 4.29 Gelombang <i>slew rate</i> pada koil motor stepper.....	70
Gambar 4.30 Desain PCB 5 fasa driver motor stepper.....	70
Gambar 4.31 Skema 5 fasa driver motor stepper.....	71
Gambar 4.32 IC L298N.....	72
Gambar 4.33 Diagram blok L298N.....	73
Gambar 4.34 PCB layout motor driver.....	73
Gambar 4.35 Rangkaian AD595 dengan <i>alarm LED</i>	74
Gambar 4.36 Skema PCB termokopel <i>amplifier</i>	76
Gambar 4.37 Desain PCB termokopel <i>amplifier</i>	76
Gambar 4.38 Diagram blok AD595.....	76
Gambar 4.39 Skema diagram prinsip kerja <i>relay</i>	77
Gambar 4.40 <i>Relay</i> HKE.....	78
Gambar 4.41 Skema desain PCB <i>relay</i>	78
Gambar 4.42 Desain PCB <i>relay</i>	78
Gambar 4.43 LCD karakter 20x4.....	79
Gambar 4.44 LCD karakter 16x2.....	79
Gambar 4.45 Skema modul LCD.....	80
Gambar 4.46 Posisi karakter berupa koordinat pada modul LCD.....	81
Gambar 4.47 Menampilkan “hello world” pada LCD.....	81
Gambar 4.48 Diagram blok CP2101.....	83
Gambar 4.49 USB Pololu ke <i>serial board</i>	84

Gambar 4.50 Pinout USB Pololu.....	84
Gambar 4.51 <i>Power supply</i> 5V 30A.....	85
Gambar 4.52 <i>Power supply</i> tipe linier.....	85
Gambar 4.53 Rangkaian lengkap <i>flyback regulator</i>	86
Gambar 4.54 Perangkat keras kontrol unit.....	87
Gambar 6.1 <i>Coordinate Measuring Machine (CMM)</i>	96
Gambar 6.2 Peralatan yang akan diuji.....	97
Gambar 6.3 Berbagai jenis probe dengan diameter dan fungsi yang berbeda-beda	98
Gambar 6.4 Probe dipasang pada dudukannya.....	98
Gambar 6.5 Dudukan probe dibersihkan.....	99
Gambar 6.6 Dudukan probe dipasang pada head mesin CMM.....	99
Gambar 6.7 Awal tampilan pada perangkat lunak.....	99
Gambar 6.8 Proses penentuan titik referensi pada mesin CMM.....	100
Gambar 6.9 Posisi <i>head</i> probe pada CMM.....	100
Gambar 6.10 Buka program MCOSMOS v.3 – <i>manual</i>	101
Gambar 6.11 Pilih menu <i>learn mode</i> untuk memulai pengukuran.....	101
Gambar 6.12 Pilih jenis probe atau kalibrasi ulang.....	102
Gambar 6.13 GUI <i>probe data management</i>	102
Gambar 6.14 Metode pengukuran elemen <i>plane</i>	103
Gambar 6.15 Tampilan pada saat proses pengukuran.....	103
Gambar 6.16 Tampilan setelah diambil 4 titik.....	104
Gambar 6.17 Tampilan elemen <i>angle</i>	105
Gambar 6.18 Grafik data perbedaan sudut delay_us 1000.....	106
Gambar 6.19 Grafik data perbedaan sudut delay_us 800.....	107
Gambar 6.20 Grafik data perbedaan sudut delay_us 600.....	108
Gambar 6.21 Grafik data perbedaan sudut delay_us 400.....	109
Gambar 6.22 Grafik data perbedaan sudut delay_us 400 50 putaran.....	110
Gambar 7.1 Perlengkapan.....	113
Gambar 7.2 Pengukuran temperatur ruang dengan AD595.....	116
Gambar 7.3 Pengukuran temperatur sampai 250 ⁰ C.....	116
Gambar 7.4 V vs T pada termokopel tipe-K.....	117
Gambar 7.5 V vs T diukur saat temperatur naik.....	117
Gambar 7.6 V vs T diukur saat temperatur turun.....	118
Gambar 7.7 Kalibrasi dengan TZN4S.....	118
Gambar 8.1 (a) Mesin RPFDM, (b) Sistem control.....	121
Gambar 8.2 Menunggu <i>heater</i> siap.....	124
Gambar 8.3 Opsi-opsi yang ada pada <i>software interface</i>	125
Gambar 8.4 Mengubah nilai Z pada <i>notepad</i>	125
Gambar 8.5 Merubah parameter kecepatan <i>feeding</i> dengan PWM.....	126
Gambar 8.6 Merubah parameter temperatur.....	126
Gambar 8.7 Grafik <i>prescaler</i> vs putaran motor pada T = 330 ⁰ C.....	127
Gambar 8.8 Grafik <i>prescaler</i> vs <i>feed rate</i> pada T = 330 ⁰ C.....	128
Gambar 8.9 Grafik <i>prescaler</i> vs putaran motor pada T = 340 ⁰ C.....	128
Gambar 8.10 Grafik <i>prescaler</i> vs <i>feed rate</i> pada T = 340 ⁰ C.....	129
Gambar 8.11 Grafik <i>prescaler</i> vs putaran motor pada T = 350 ⁰ C.....	129

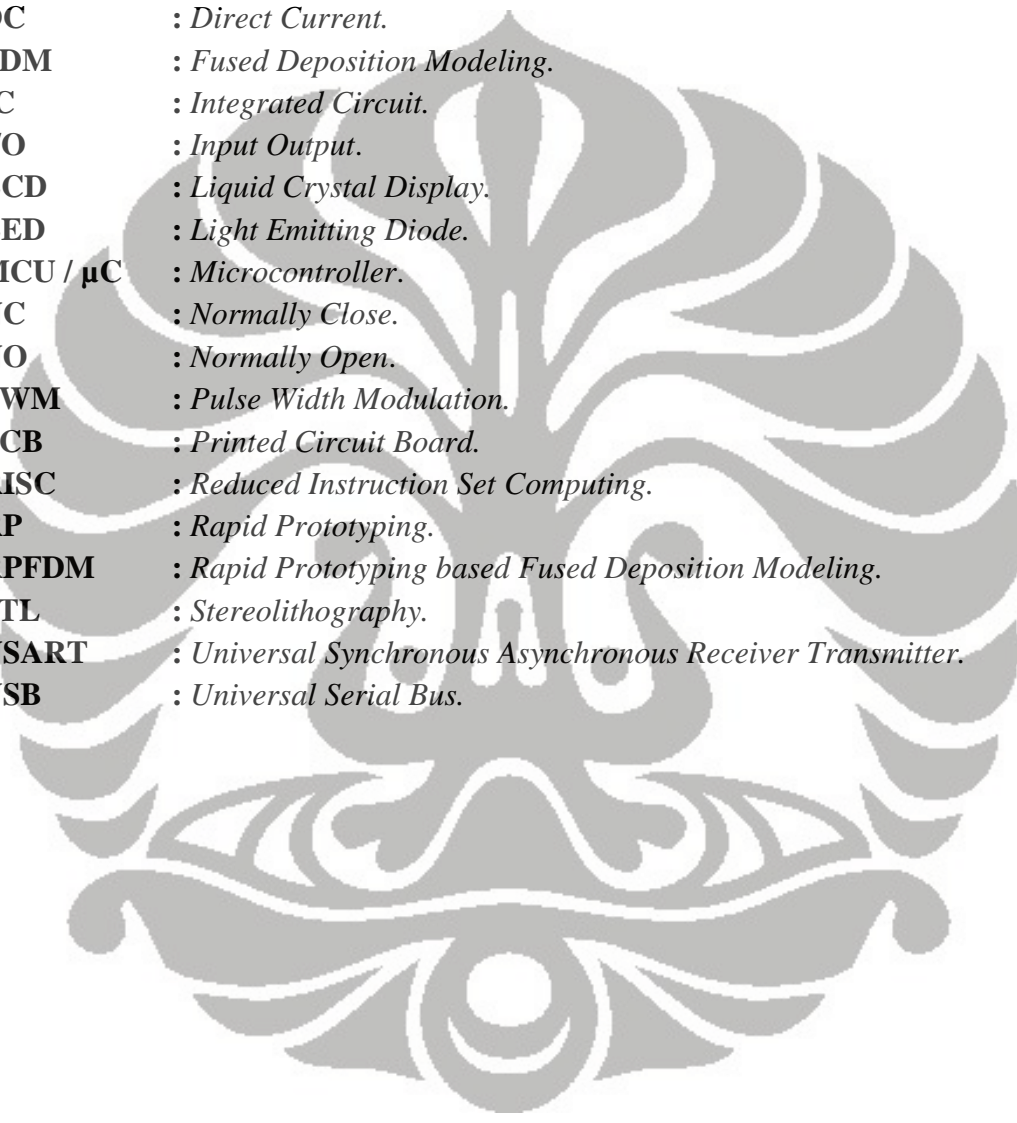
Gambar 8.12 Grafik <i>prescaler</i> vs <i>feed rate</i> pada $T = 350^{\circ}\text{C}$	130
Gambar 8.13 Grafik <i>prescaler</i> vs putaran motor pada $T = 360^{\circ}\text{C}$	130
Gambar 8.14 Grafik <i>prescaler</i> vs <i>feed rate</i> pada $T = 360^{\circ}\text{C}$	131
Gambar 8.15 Grafik <i>delay_us</i> vs kecepatan linier <i>nut</i>	134
Gambar 8.16 Hasil pengujian dengan parameter pengubah ketinggian <i>bed</i>	138
Gambar 8.17 Hasil produk dengan faktor pengubah ketinggian <i>bed</i>	139
Gambar 8.18 Hasil pengujian dengan parameter pengubah kecepatan <i>feeding</i>	140
Gambar 8.19 Hasil produk dengan faktor pengubah kecepatan <i>feeding</i>	141
Gambar 8.20 Hasil pengujian dengan parameter pengubah temperatur.....	142
Gambar 8.21 Hasil produk dengan faktor pengubah temperatur.....	143
Gambar 8.22 Perbandingan hasil produk $z = 320 \text{ step}$ dan $z = 440 \text{ step}$	144
Gambar 8.23 Perbandingan hasil produk PWM = 0,6 dengan PWM 0,65 <i>intermittent</i>	146
Gambar 8.24 Perbandingan hasil produk $T=330^{\circ}\text{C}$ dengan $T=360^{\circ}\text{C}$	147



DAFTAR TABEL

Tabel 2.1 Ringkasan dari proses pabrikan komersial RP.....	16
Tabel 2.2 Karakteristik material termoplastik.....	23
Table 3.1 Spesifikasi tipe MGN.....	30
Tabel 3.2 Spesifikasi motor stepper lima fasa.....	37
Tabel 4.1 <i>Pin Assignment</i> μC Master dan μC Slave.....	60
Tabel 4.2 Spesifikasi L298N.....	72
Tabel 4.3 Temperatur terhadap tegangan keluaran AD595.....	75
Tabel 4.4 Pin antar muka modul LCD.....	80
Tabel 4.5 Tabel ASCII.....	82
Tabel 6.1 Hasil pengukuran dengan delay_us 1000.....	106
Tabel 6.2 Hasil pengukuran dengan delay_us 800.....	107
Tabel 6.3 Hasil pengukuran dengan delay_us 600.....	108
Tabel 6.4 Hasil pengukuran dengan delay_us 400.....	109
Tabel 6.5 Hasil pengukuran dengan delay_us 400.....	110
Tabel 6.6 Tabel perbedaan sudut rata-rata.....	111
Tabel 8.1 Perhitungan <i>feed rate</i> pada $T = 330^{\circ}\text{C}$	132
Tabel 8.2 Perhitungan <i>feed rate</i> pada $T = 340^{\circ}\text{C}$	132
Tabel 8.3 Perhitungan <i>feed rate</i> pada $T = 350^{\circ}\text{C}$	133
Tabel 8.4 Perhitungan <i>feed rate</i> pada $T = 360^{\circ}\text{C}$	133
Tabel 8.5 Linierisasi data, deviasi standar, dan <i>Coefficient of Determination</i>	137
Tabel 8.6 Analisis hasil produk dengan faktor pengubah ketinggian <i>bed</i>	143
Tabel 8.7 Analisis hasil produk dengan faktor pengubah kecepatan <i>feeding</i> ..	145
Tabel 8.8 Analisis hasil produk dengan faktor pengubah temperatur.....	146

DAFTAR ISTILAH



ADC	: <i>Analog Digital Converter.</i>
CAD	: <i>Computer Aided Design.</i>
CAM	: <i>Computer Aided Manufacturing.</i>
CPU	: <i>Central Processing Unit.</i>
DC	: <i>Direct Current.</i>
FDM	: <i>Fused Deposition Modeling.</i>
IC	: <i>Integrated Circuit.</i>
I/O	: <i>Input Output.</i>
LCD	: <i>Liquid Crystal Display.</i>
LED	: <i>Light Emitting Diode.</i>
MCU / μC	: <i>Microcontroller.</i>
NC	: <i>Normally Close.</i>
NO	: <i>Normally Open.</i>
PWM	: <i>Pulse Width Modulation.</i>
PCB	: <i>Printed Circuit Board.</i>
RISC	: <i>Reduced Instruction Set Computing.</i>
RP	: <i>Rapid Prototyping.</i>
RPFDM	: <i>Rapid Prototyping based Fused Deposition Modeling.</i>
STL	: <i>Stereolithography.</i>
USART	: <i>Universal Synchronous Asynchronous Receiver Transmitter.</i>
USB	: <i>Universal Serial Bus.</i>

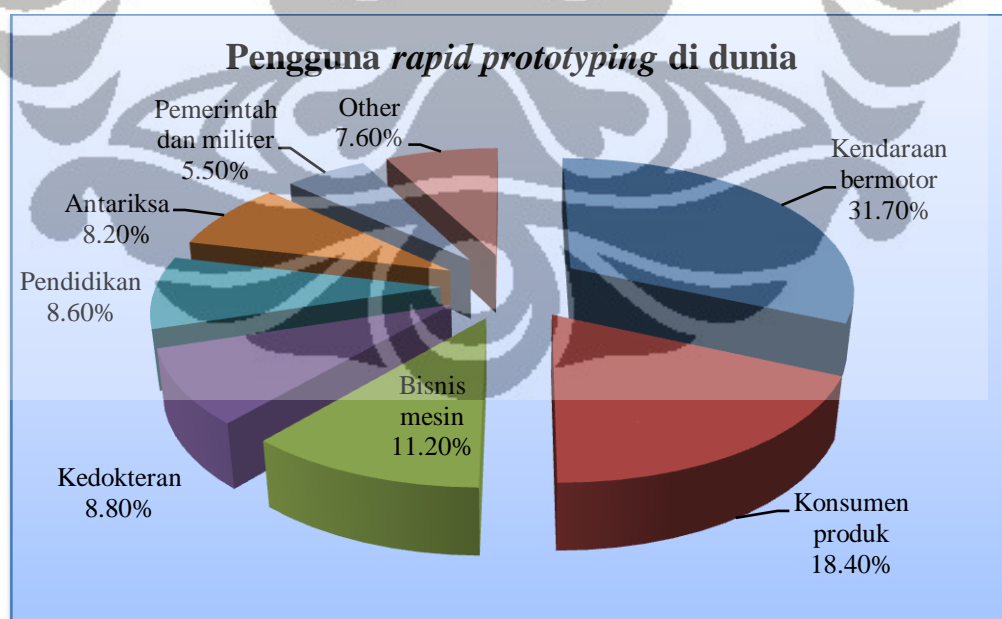
BAB 1

PENDAHULUAN

1.1 Latar belakang

Teknologi *rapid prototyping* masih merupakan teknologi yang relatif baru di dalam dunia industri manufaktur. Kebutuhan akan pasar saat ini mengharuskan industri-industri menghadapi tantangan global, salah satunya adalah kecepatan dalam proses merancang dan membuat konsep desain, sehingga konsep akhir dari sebuah produk menjadi lebih cepat dan terhindar dari kesulitan dan kesalahan dalam proses produksi, perakitan, dan yang lainnya. Selain itu, untuk mendapatkan kepuasan dari konsumen, dipergunakanlah alat peraga berupa prototipe untuk mempresentasikan maksud dan fungsi dari suatu produk.

Prototipe adalah produk fisik model pertama, yang merupakan makna akhir dari verifikasi bentuk, kesesuaian, dan fungsi produk [1]. Oleh karena itu, teknologi *rapid prototyping* ini sangat membantu untuk meningkatkan keuntungan dari perusahaan serta kepuasan dari konsumen.



Gambar 1.1 Penggunaan *rapid prototyping* di dunia [2]

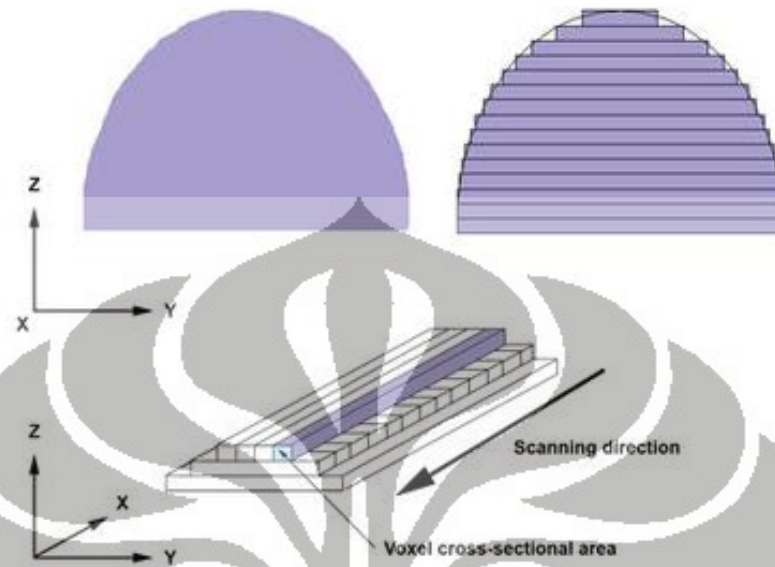
Penggunaan teknologi *rapid prototyping* sekarang sudah hampir merata di setiap bidang ilmu, seperti di bidang kedokteran sebesar 8,8%, pendidikan 8,6%, antariksa 8,2%, pemerintahan/militer 5,5%, bisnis mesin 11,2%, konsumen produk 18,4%, dan bidang lainnya sebesar 7,6%, serta yang terbesar di teknologi manufaktur untuk kendaraan bermotor sebesar 31,7%. Dikarenakan teknologi *rapid prototyping* sudah banyak diaplikasikan diberbagai bidang, terutama di bidang manufaktur. Hal inilah menjadi landasan penelitian *rapid prototyping* perlu dikembangkan dan diaplikasikan didalam dunia perindustrian di Indonesia.

Rapid prototyping adalah proses pembuatan konstruksi obyek fisik secara otomatis dengan proses lapis per lapis secara bertahap. Prosesnya pun berbeda dengan proses pemesinan pada *milling*, dimana proses *rapid prototyping* adalah proses penambahan material model fisik secara lapis per lapis dari lapisan pertama hingga mencapai lapisan teratas, sedangkan pada proses *milling* adalah proses pengurangan material dari material awal dengan menggunakan *cutting tool* sehingga menjadi bentuk yang akan diinginkan [1]. Persamaan dari keduanya adalah kemiripan titik *cutter contact* (cc) pada *cutting tool* dengan titik cc pada lintasan ekstruder (*laser trajectory*) hampir sama hanya terdapat perbedaan sedikit yang telah disebutkan.

Pengembangan dari teknologi *rapid prototyping* yaitu dengan menggunakan teknologi *direct manufacturing*, yaitu suatu produk langsung diproduksi tanpa adanya bantuan atau sentuhan tangan manusia. Prosesnya yaitu mesin mengambil data dari komputer berupa file CAD. Pada umumnya pada sistem CAD (*Computer Aided Design*) berektensi '.dxf, .prt, .sldprt' tergantung dari produsen software yang mengembangkan. Kemudian file CAD tersebut dikonversi menjadi file STL yang merupakan interpresentasi dari permukaan model 3D terdiri dari segitiga-segitiga pembentuk, setelah itu file STL tersebut dikonversi menjadi lintasan ekstruder.

Standar data secara *de-facto* yang digunakan antara sistem CAD/CAM dengan mesin *rapid prototyping* adalah file STL (*stereolithography*). File STL mempresentasikan bentuk dari suatu objek *part* atau *assembly* dengan menggunakan *triangular facet* [2]. *Triangular facet* terdiri dari segitiga-segitiga pembentuk, dimana sebuah segitiga diinterpresentasikan dengan tiga buah titik dan satu vektor normal

sebagai pembentuk segitiga. File STL yang berupa *triangular facet* tersebut kemudian dikonversi menjadi suatu lintasan ekstruder secara lapis per lapis.



Gambar 1.2 Ilustrasi proses *additive manufacturing layer by layer* pada *rapid prototyping* [2]

Teknologi *rapid prototyping* sampai saat ini antara lain adalah *Selective Laser Sintering (SLS)*, *Fused Deposition Modeling (FDM)*, *Stereolithography (SLA)*, *Laminated Object Manufacturing (LOM)*, *Electron Beam Melting (EBM)*, *3D Printing (3DP)*. Dari beberapa teknologi tersebut, terdapat keuntungan dan kerugiannya masing-masing. Contohnya berupa keakuratan suatu produk, semakin tinggi keakuratan suatu produk dalam hal dimensi maupun bentuk yang makin kompleks maupun rumit, hal ini yang menjadikan biaya pembuatan prototipe tersebut yang semakin tinggi.

Penelitian lebih ditekankan dengan menggunakan metode *Fused Deposition Modeling (FDM)*, karena metode ini cukup mudah untuk diaplikasikan dan biaya dapat disesuaikan dengan akurasi dari produk. *Fused Deposition Modeling* adalah metode yang menggunakan *additive manufacturing* yaitu proses pembentukan suatu produk dengan proses lapis per lapis menggunakan material termoplastik yang diumpan kedalam sebuah pengumpan (*feeder*). Kemudian material tersebut dimasukkan kedalam sebuah pemanas berupa tabung (*barrel*) dan dilelehkan pada temperatur leleh hingga berubah fasa menjadi cair (*liquifier*). Setelah material berubah fasa menjadi cair, maka

dilakukanlah proses pembentukan produk atau prototipe dengan proses lapis per lapis, dengan terus-menerus mengumpan material termoplastik masuk ke dalam tabung. Proses tersebut berlangsung secara kontinu, dari proses pembentukan layer pertama sampai dengan layer teratas.

1.2 Tujuan penelitian

Tujuan umum dari penelitian teknologi *rapid prototyping* ini adalah untuk mengembangkan sebuah mesin *rapid prototyping* dengan menggunakan metode *Fused Deposition Modeling*. Adapun tujuan khususnya adalah :

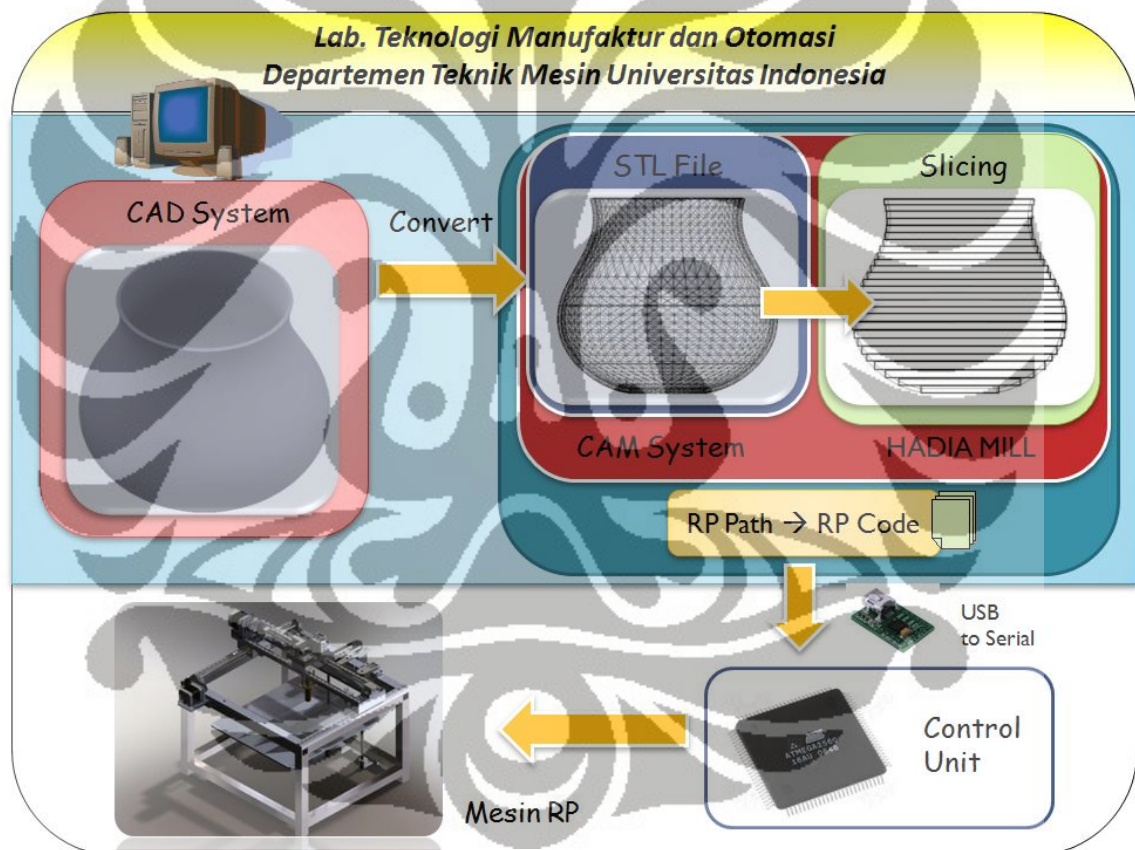
1. Merancang dan mengembangkan mekanisme gerak dan kerja mesin *rapid prototyping* dengan menggunakan metode *fused deposition modeling* dengan biaya yang rendah.
2. Mengembangkan sistem kontrol mesin *rapid* dengan menggunakan mikrokontroler, agar dapat berkomunikasi dengan komputer, mengontrol mesin *rapid*, dan sistem keseluruhan.
3. Mengembangkan mekanisme ekstruder dengan menggunakan kawat nikrom yang dililitkan pada material tembaga sebagai barrel, di desain pada temperatur leleh, agar material yang dihasilkan cukup baik untuk dijadikan prototipe.
4. Mengetahui karakteristik sistem kontrol dan hasil produk dengan perubahan parameter-parameter yang mempengaruhi diantaranya ketinggian *bed*, kecepatan *feeding*, temperatur, dan diameter keluaran.

1.3 Perumusan masalah

Mesin *rapid prototyping* yang dikembangkan menggunakan tiga buah motor stepper, sistem ekstruder, dan sistem komunikasi antara PC dengan mesin. Sistem kontrol harus mengontrol mekanisme pergerakan sistem tersebut dan sistem secara keseluruhan. Perumusan masalah didalam pengembangan sistem kontrol adalah sebagai berikut :

1. Sistem kontrol dapat mengontrol pergerakan tiga buah motor stepper lima fasa dan mengontrol pergerakan motor stepper sumbu x dan sumbu y secara bersamaan.
2. Sistem kontrol dapat mengontrol kecepatan dari motor DC yang terdapat pada mekanisme ekstruder.
3. Sistem kontrol dapat mendeteksi temperatur pada *heater*, dan mampu membuat temperatur pada *heater* tetap konstan.
4. Sistem kontrol dapat berkomunikasi antara mesin dengan komputer, dalam hal proses transfer data dari komputer ke sistem kontrol maupun sebaliknya.

Mesin *rapid prototyping* memiliki arsitektur sebagai berikut :



Gambar 1.3 Arsitektur mesin *rapid prototyping*

1.4 Pembatasan masalah

Pembatasan masalah pada penelitian ini adalah pengembangan dari segi perangkat keras dan sistem kontrol. Dikarenakan mesin *rapid prototyping* merupakan terdiri dari sistem yang kompleks, maka peneliti mengambil bagian sistem elektronik sebagai basis sistem kontrol mesin *rapid prototyping*. Untuk pengembangan yang lain seperti, pengembangan mekanisme konstruksi, analisa material, pengembangan program, dan mekanisme ekstruder akan dijelaskan oleh peneliti lain, yang satu tim dengan peneliti dalam pengembangan mesin *rapid prototyping*.

1.5 Metodologi penelitian

Penelitian dilakukan dengan melakukan penelusuran referensi melalui buku atau skripsi maupun melalui internet. Kebanyakan informasi mengenai perangkat keras elektronik seperti mikrokontroller, *motor driver*, dan yang lainnya terdapat informasi yang melimpah di internet. Berikut metodologi penelitian pengembangan mesin *rapid prototyping* secara umum:

1. Mempelajari referensi literatur melalui buku maupun internet.
2. Perancangan mekanisme mesin sebagai basis konstruksi mesin.
3. Mengembangkan mekanisme ekstruder dengan menggunakan prinsip *Fused Deposition Modeling*.
4. Mengembangkan perangkat keras elektronik dan sistem kontrol motor stepper, ekstruder, dan PWM.
5. Menguji alat dan menganalisa kemampuan mesin rapid dan sistem kontrol yang telah dibuat.

1.5.1 Studi literatur

Studi literatur yang dilakukan dalam penelitian ini dibagi menjadi beberapa bagian. Bagian yang pertama adalah studi literatur sebagai penunjang desain yang meliputi studi mengenai statika struktur dari alat, material teknik dan proses produksi. Bagian kedua adalah bagian perangkat keras sistem kontrol yang memerlukan studi

literatur dari pengetahuan perangkat keras elektronik maupun dari *datasheet* sistem yang digunakan. Bagian ketiga adalah studi literatur mengenai perangkat lunak yang meliputi penulisan program dengan menggunakan bahasa pemrograman yaitu bahasa C.

1.5.2 Perancangan mekanisme mesin

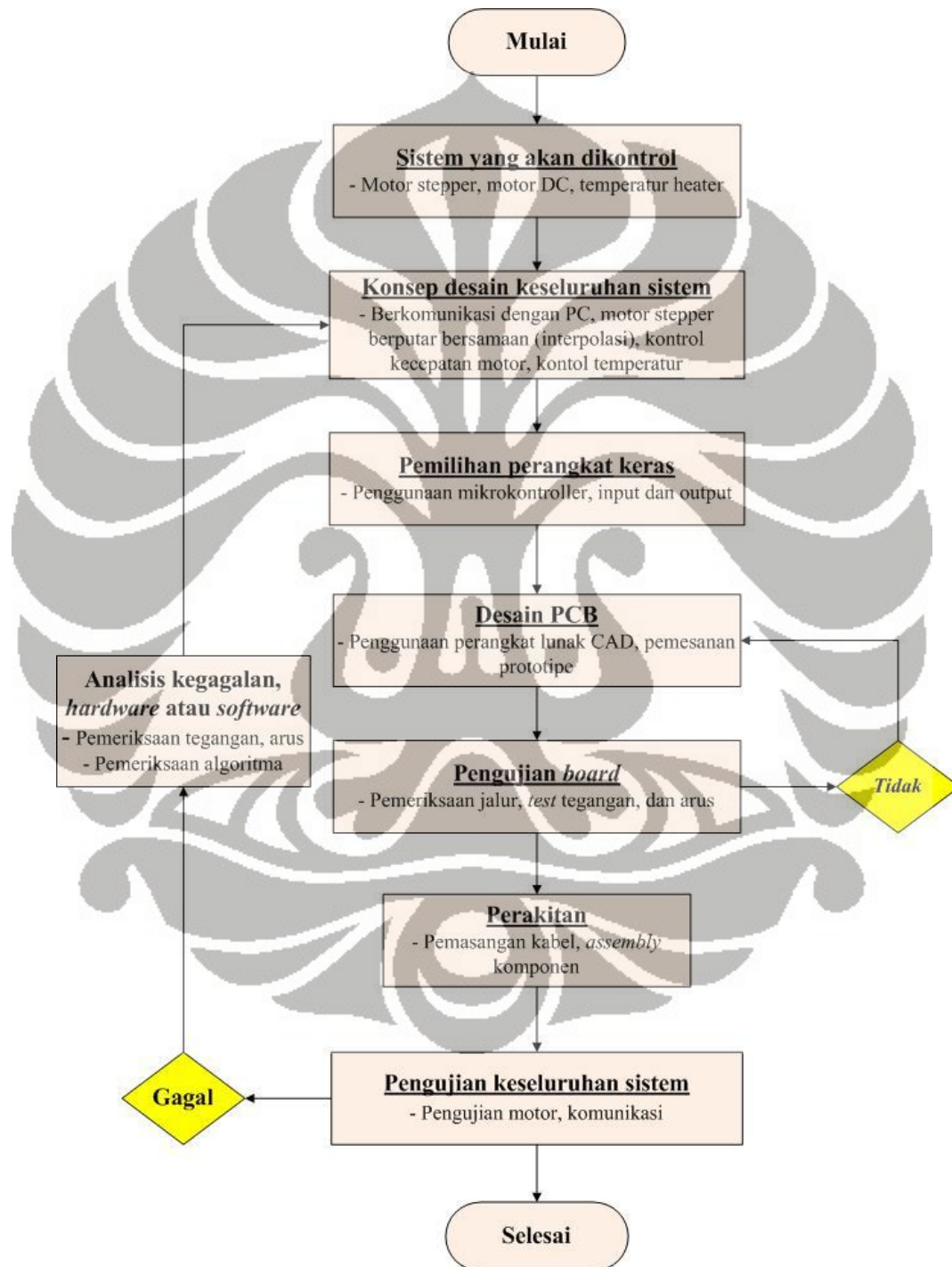
Perancangan desain mekanisme mesin dilakukan dengan menggunakan perangkat lunak CAD (*Computer Aided Design*) serta menerapkan prinsip *Design for Manufacturability* dan *Assembly and Dissassembly* (DFMAD). Proses perancangan mekanisme mesin dilakukan secara bertahap dan melalui tahap evaluasi desain sehingga mendapatkan desain yang paling optimal, baik dari sisi kualitas maupun dari sisi harga. Perancangan mekanisme mesin meliputi perancangan konstruksi mesin *rapid prototyping* dan mekanisme gerak sistem pengumpan material (*feeder*). Perancangan konstruksi mesin *rapid* menggunakan sistem *gantry* berdimensi 400x450x300 mm³ dengan menggunakan material *hollow steel*. Sedangkan pada mekanisme gerak sistem *feeder* dapat bergerak secara linier searah sumbu x, y dan z, dengan menggunakan mekanisme *ballscrew* dan linier *guideway*.

1.5.3 Pengembangan mekanisme ekstruder

Pengembangan desain mekanisme ekstruder juga dilakukan dengan menggunakan perangkat lunak CAD (*Computer Aided Design*) serta menerapkan prinsip *Design for Manufacturability* dan *Assembly and Dissassembly* (DFMAD). Pengembangan mekanisme ekstruder menggunakan prinsip FDM (*Fused Deposition Modeling*). Metode FDM, sangat mudah untuk dibuat dan diterapkan jika ditinjau dari segi komponen dan sistem mekanikal dari keseluruhan mekanisme. Pengembangan mekanisme ekstruder meliputi proses perancangan sistem pengumpan material (*feeder*), sistem pemanas, dan sistem isolator. Pada mekanisme ekstruder sistem feeder terdiri dari *driver gear* dan *driven pulley*, sedangkan sistem pemanas material menggunakan kawat nikrom.

1.5.4 Pengembangan sistem kontrol mesin

Pengembangan sistem kontrol berdasarkan sistem mekanikal dan mekanisme ekstruder yang terdapat pada mesin, berikut skema perancangan sistem kontrol secara keseluruhan :



Gambar 1.4 Flowchart perancangan sistem kontrol

1.5.5 Pengujian alat

Pada tahap ini, mesin yang telah dikembangkan dilakukan pengujian metrologi, sehingga didapatkan tingkat keakurasian untuk pergerakan setiap sumbu. Sedangkan sistem kontrol yang telah dikembangkan juga dilakukan pengujian error pengiriman pulsa terhadap pergerakan motor stepper.

1.6 Sistematika penulisan

BAB 1. PENDAHULUAN

Bab ini menjelaskan mengenai latar belakang masalah, tujuan penelitian, perumusan penelitian, metodologi penelitian, dan sistematika penulisan.

BAB 2. SISTEM *RAPID PROTOTYPING*

Bab ini menjelaskan tentang gambaran secara umum *rapid prototyping*, proses *rapid prototyping*, tipe-tipe teknologi yang digunakan, dan material yang digunakan pada metode FDM.

BAB 3. TINJAUAN SISTEM MEKANIKAL DAN EKSTRUDER RPFDM

Bab ini menjelaskan tentang sistem mekanikal yang digunakan pada *robot cartesian* dan mekanisme sistem dari ekstruder dengan menggunakan metode *Fused Deposition Modelling*.

BAB 4. PENGEMBANGAN SISTEM KONTROL

Bab ini menjelaskan pengembangan arsitektur sistem elektronik yang digunakan sebagai basis sistem kontrol mesin *rapid prototyping*. Sistem kontrol yang telah dikembangkan berdasarkan sistem mekanikal dan mekanisme ekstruder.

BAB 5. PENGEMBANGAN PERANGKAT LUNAK

Pada bab ini dijelaskan mengenai algoritma pada mikrokontroler PWM kontrol dan temperatur kontrol.

BAB 6. PENGUJIAN DAN ANALISA SISTEM KONTROL DRIVER MOTOR STEPPER

Pada bab ini dijelaskan mengenai pengujian dari sistem kontrol khususnya driver motor stepper. Analisa terhadap sistem yang telah dikembangkan dan error yang terjadi pada sistem tersebut.

BAB 7. PENGUJIAN DAN ANALISA SISTEM KONTROL TERMOKOPEL AMPLIFIER

Pada bab ini dijelaskan mengenai pengujian dari sistem kontrol khususnya termokopel *amplifier*. Analisa terhadap sistem yang telah dikembangkan dan error yang terjadi pada sistem tersebut, serta proses kalibrasi sistem terhadap mekanisme alat ukur yang dapat dipercaya.

BAB 8. PENGUJIAN DAN ANALISA PRODUK AKHIR

Pada bab ini dijelaskan pengujian dari kecepatan motor DC dengan menggunakan metode kontrol kecepatan PWM dengan jangkauan temperatur koil 330°C – 360°C , serta analisa produk akhir dengan perubahan parameter tertentu.

BAB 9. KESIMPULAN DAN SARAN PENELITIAN LEBIH LANJUT

Bab ini menjelaskan kesimpulan penelitian dan saran untuk pengembangan penelitian selanjutnya.

BAB 2

SISTEM RAPID PROTOTYPING

2.1 Teknologi *rapid prototyping*

Teknologi *rapid prototyping* (RP) adalah teknologi yang dapat secara otomatis membangun konstruksi physical model (*prototype*) dari data sistem CAD (*Computer Aided Design*). Data tersebut merupakan data file CAD kemudian dikonversi menjadi file STL, yang terdiri dari sekumpulan data segitiga-segitiga pembentuk (*triangular facet*) dari permukaan solid CAD model [1]. Kemudian dilakukan proses *slicing* terhadap segitiga-segitiga pembentuk tersebut, sehingga membentuk data lintasan ekstruder. Dalam membangun konstruksi prototipe yaitu dengan proses penambahan material secara lapis per lapis, proses tersebut disebut sebagai proses *additive manufacturing*. Berbeda dengan proses pemesinan pada *CNC Milling* dimana proses tersebut membuang material dari material awal [4].

Teknologi RP ini sangat cocok untuk visualisasi antara insinyur dengan pelanggan, dan *prototyping* juga dapat digunakan untuk proses pengujian desain. Misalnya, seorang *engineer aerospace* (insinyur aerodinamika) dapat membuat prototipe *airfoil*, dan dapat mengujinya di *wind tunnel* untuk mengukur gaya *lift* dan *drag* [3].

Rapid prototyping dapat juga digunakan untuk membuat peralatan dan juga komponen-komponen produksi yang berkualitas. Untuk produksi dalam jumlah yang kecil dan objek yang kompleks, *rapid prototyping* menjadi pilihan yang terbaik. Istilah *rapid* (cepat) sebenarnya relatif, karena untuk membuat sebagian besar prototipe membutuhkan waktu sekitar 3 jam sampai 72 jam tergantung dari ukuran dan kompleksitas dari objek. Hal seperti ini kelihatannya lambat, akan tetapi sudah jauh lebih cepat jika dibandingkan dengan metode konvensional seperti pemesinan yang membutuhkan waktu berminggu-minggu maupun beberapa bulan lamanya [3].

Menurut Prat dan Whitney, penghematan waktu yang sangat besar tentu saja sangat menguntungkan, mereka dapat menghemat waktu sekitar 70 sampai 90% untuk pembuatan prototipe dari produk *investment casting* mereka.

Teknologi *rapid prototyping* bukanlah teknologi yang sempurna. Volume part pada umumnya terbatas sampai 0,125 m³ atau kurang dari itu, tergantung dari jenis mesin *rapid prototyping*-nya. Pada umumnya teknologi RP sampai saat ini masih menggunakan material termoplastik, dan belum menyentuh *prototyping* dengan bahan metal (*metal prototyping*) dikarenakan prosesnya yang sangat sulit, sehingga prosesnya masih menggunakan metode konvensional (pemesinan). Terlepas dari keterbatasan itu, *rapid prototyping* merupakan teknologi yang luar biasa dan revolusioner di dalam bidang industri manufaktur[3].

Keuntungan menggunakan teknologi *rapid prototyping* [6]:

- a) Menghemat dan mempersingkat proses desain suatu produk.
- b) Dengan membuat prototipe secara otomatis lebih menghemat uang daripada dengan menggunakan teknologi konvensional (pemesinan).
- c) Meningkatkan kualitas produk.
- d) Prototipe lebih komunikatif jika dibandingkan dengan gambar 3D.
- e) Masukkan balik atau *feedback* yang cepat dari pelanggan dapat disampaikan ke *designer/engineer* jika ada kesalahan pada desain atau tidak sesuai dengan permintaan pelanggan.
- f) Fleksibilitas dan kemudahan dalam perubahan desain maupun dimensi suatu produk.
- g) Prototipe juga dapat di-*finishing* dan dicat menyerupai produk akhir.

2.2 Proses *rapid prototyping*

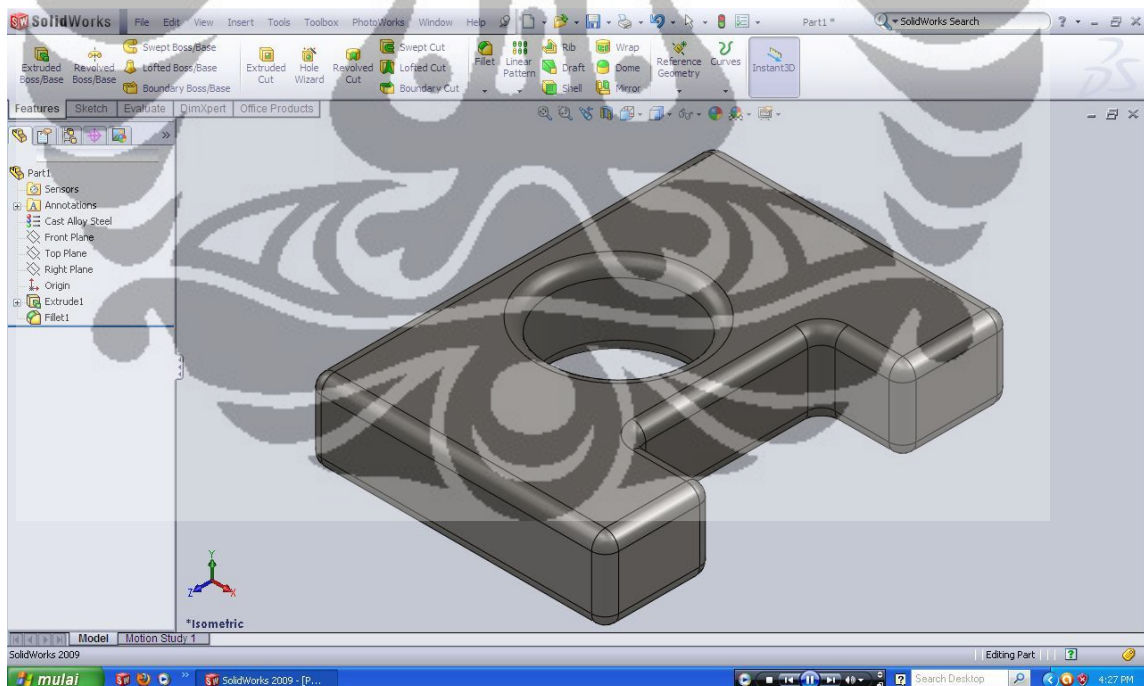
Pada umumnya ada lima langkah-langkah dasar untuk mengaplikasikan teknologi RP ini, yaitu [3]:

- 1) Mendesain model CAD dengan sistem CAD.
- 2) Mengkonversi CAD data menjadi format data STL.

- 3) Proses slicing STL file.
- 4) Membangun model secara lapis per lapis.
- 5) Proses pembersihan dan penyelesaian model.

2.2.1 Mendesain model CAD

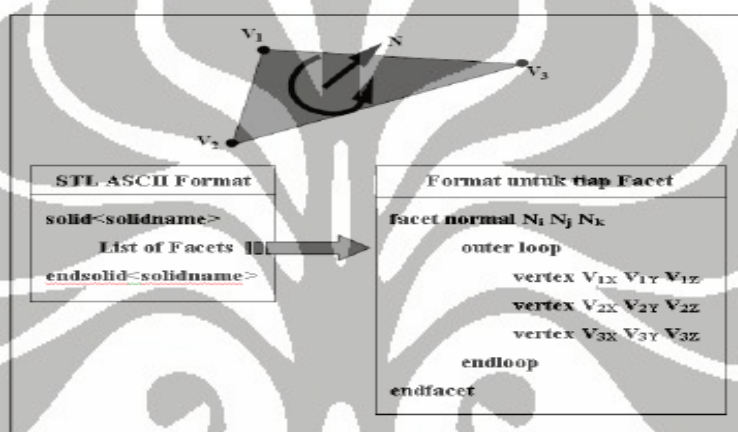
Pertama yang dilakukan didalam proses pembuatan prototipe menggunakan teknologi *rapid prototyping* adalah membuat dan mendesain model CAD. CAD (*Computer Aided Design*) adalah perangkat lunak yang dapat mendefinisikan dan memodelkan komponen mekanik secara geometri, permukaan, dan solid. Pada saat ini ada banyak perangkat lunak sistem CAD yang dapat digunakan untuk membuat model CAD secara tiga dimensi, seperti Pro ENGINEER, SolidEdge, SolidWorks, Autodesk Inventor, Unigraphics UGNX, dan perangkat lunak gratis. File CAD tersebut dapat secara langsung digunakan untuk proses prototyping, yang sebelumnya dikonversi terlebih dahulu menjadi file STL. Proses ini indentik pada semua proses teknik *rapid prototyping* [3].



Gambar 2.1 User interface SolidWorks 2009

2.2.2 Mengkonversi file STL

Beberapa variasi perangkat lunak CAD, seperti SolidWorks, Pro Eng, Inventor dan Unigraphics UNX mempresentasikan suatu obyek *solid* dengan algoritma atau format data yang berbeda-beda. Oleh karena itu, untuk menjaga konsistensi, maka dibentuklah format data STL (*Stereolithography*) untuk menjadi standar didalam dunia industri *rapid prototyping* [3]. Pada langkah kedua, adalah mengkonversi file CAD menjadi file STL. File STL mempresentasikan permukaan tiga dimensi *part model* maupun *assembly* kedalam *planar triangles*.



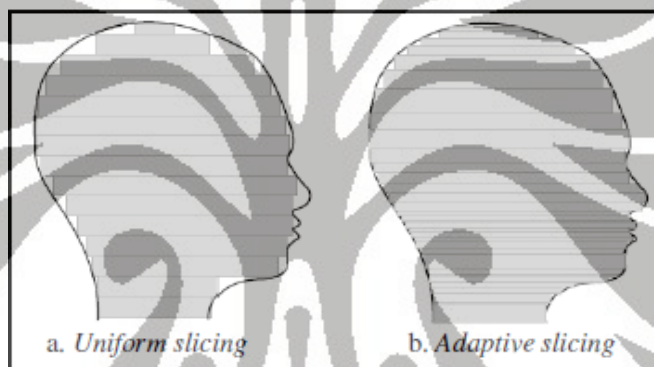
Gambar 2.2 STL ASCII Format [1]

File ini terdiri dari segitiga-segitiga *planar* yang dipresentasikan dengan tiga koordinat *vertex* segitiga dan arah vektor *normal* dari segitiga tersebut. Karena STL merupakan dalam bentuk *planar triangle*, maka tidak dapat mempresentasikan bentuk kurva (*sculpture surface*) secara baik. Salah satu pendekatannya adalah dengan memperbanyak jumlah segitiga, akan tetapi akan memperbesar ukuran dari file tersebut. Karena file STL sangat umum dan *universal*, maka proses ini sangat identik untuk semua teknik *rapid prototyping* [3].

2.2.3 Proses *slicing* file STL

Pada langkah ketiga, adalah proses *slicing* file STL menjadi data untuk lintasan ekstruder pada proses FDM (Fused Deposition Modeling). Sudah banyak perangkat lunak yang tersedia di internet, seperti di RepRap.com. Pengguna diharuskan untuk

menentukan ukuran, lokasi, dan orientasi pada model. Menentukan orientasi pada model menjadi penting karena beberapa alasan. Pertama, karakteristik dari *rapid prototyping* berubah-ubah dari satu arah koordinat ke yang lainnya. Contohnya, prototipe biasanya lebih rendah dan kurang akurat pada arah z (vertikal) dibandingkan bidang x dan y. Orientasi *part* dapat juga menentukan besarnya waktu yang dibutuhkan untuk membangun model. Peletakan dimensi dari *part* yang terpendek kedalam arah z dapat mengurangi jumlah layer, sehingga mengurangi waktu pembuatan [3]. Pendekatan *slicing* pada file STL ada dua, yaitu *uniform* dan *adaptive*. Dimana metode *slicing* dengan ketebalan *uniform*, yaitu semua layer memiliki ketebalan yang sama. Metode *slicing* dengan ketebalan *adaptive*, yaitu ketebalan layer dapat bervariasi menurut kompleksitas geometri [1].



Gambar 2.3 Metode *slicing* pada STL file [1]

2.2.4 Membangun model secara lapis per lapis

Pada langkah keempat, adalah proses aktual dalam membangun *part prototyping*. Prosesnya dengan membuat lapisan terbawah terlebih dahulu kemudian seterusnya dengan menambahkan material pada lapisan berikutnya sampai lapisan teratas sampai terbentuknya *part prototyping*. Ada banyak teknologi yang secara umum digunakan untuk proses ini, seperti *Stereolithography*, *Laminated Object Manufacturing*, *Selective Laser Sintering*, *Fused Deposition Modeling*, *Solid Ground Curing*, dan *3D Ink Jet Printing*. Kebanyakan mesin *rapid prototyping* sampai saat ini masih menggunakan material polimer, kertas, atau *powdered metal* [3].

2.2.5 Proses pembersihan dan penyelesaian model

Proses pembersihan dan penyelesaian model adalah langkah terakhir dalam membuat prototipe. Proses ini mengambil prototipe dari mesin, melepaskan material pendukung jika menggunakan, serta membersihkan prototipe dari sisa-sisa material selama didalam proses pembuatan. Prototipe mungkin juga memerlukan pembersihan dan penanganan permukaan (*surface treatment*) untuk diukur keakuratannya [3].

2.3 Rapid prototyping teknik

Tabel 2.1 Ringkasan dari proses pabrikasi komersial RP [5]

Fabrication Technique	Maximum Work Envelope (mm) (W x L x H)	Build Time (hrs)	Accuracy (mm)	Unit Cost (SAUD)
SFF	240 x 240 x 240 ^[3]	N/A	2.5 ^[3]	\$6000
Fused Deposition Modelling	600 x 500 x 600 ^[3]	13 ^[3]	.125 to .75 ^[2]	\$19,000 to \$300,000 ^[3]
Inkjet Deposition	298 x 185 x 203 ^[3]	N/a	N/a	\$60,000 to \$85,000 ^[3]
Layered Object Manufacturing	560 x 810 x 510 ^[2]	11 ^[3]	.127 to .25 ^[2]	\$15,000 to \$240,000 ^[3]
Selective Inkjet Binding	508 x 609 x 406 ^[3]	N/a	.125 ^[2]	\$20,000 to \$70,000 ^[3]
Selective Laser Sintering	381 x 330 x 457 ^[3]	5 ^[3]	.075 to .4 ^[2]	\$300,000 ^[3]
Solid Ground Curing	500 x 350 x 600 ^[3]	11 ^[3]	.1 to .5 ^[2]	N/a
Stereolithography	500 x 500 x 600 ^[3]	7 ^[3]	.075 to .25 ^[2]	\$75,000 to \$800,000 ^[3]

[1] – Wright, 2001

[2] – Degarmo et al., 2003

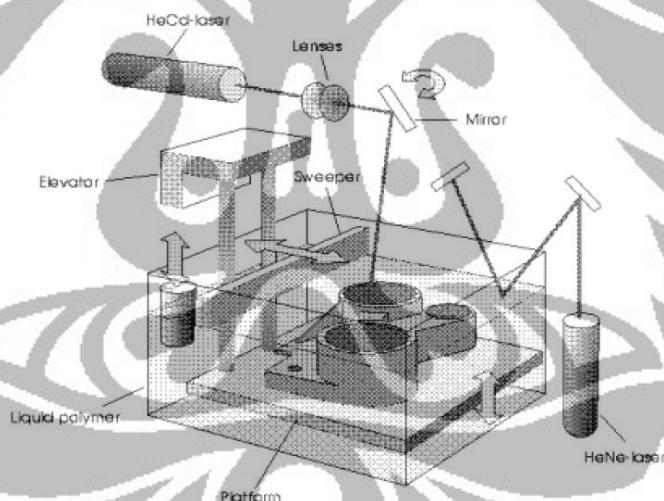
[3] – Grenda, 2006

Produk komersial dari mesin *rapid prototyping* pertama dimulai pada tahun 1980-an. Beberapa teknik revolusioner telah dikembangkan industri dan sekarang telah menghasilkan keuntungan yang signifikan dalam merangkul pelanggan. Ada delapan teknik dasar proses pabrikasi RP diantaranya *Solid Freeform Fabrication (SFF)*, *Fused Deposition Modelling*, *Inkjet Deposition*, *Layered Object Manufacturing*, *Inkjet*

Binding, Laser Sintering, Solid Ground Curing dan *Stereolithography*. Kemampuan setiap teknik pada umumnya dideskripsikan dengan kemampuan dimensi kerja mesin (*work envelope*), waktu pembuatan (*built time*), keakuratan produk (*accuracy*), dan biaya produksi (*unit cost*) seperti yang ditunjukkan oleh Tabel 2.1 [5].

2.3.1 Stereolithography

Stereolithography dikembangkan pertama kali oleh 3D System, dan memulai revolusi-nya di dalam bidang *rapid prototyping*, metode ini kemudian dipatenkan pada tahun 1986, kemudian dikomersilkan oleh 3D System pada tahun 1988. Mesin *Stereolithography Apparatus* (SLA) dibuat pada tahun 1988 oleh 3D System di Valencia, CA. Pada saat ini, 3D System sebagai industri *leader*, sudah menjual banyak mesin *rapid prototyping* ke banyak perusahaan. *Stereolithography* menjadi *benchmark* karena merupakan teknik pertama di dalam teknologi *rapid prototyping*.



Gambar 2.4 Skema diagram *Stereolithography* [3]

Metode ini dapat membangun model tiga dimensi dari polimer *photosensitive* cair (*liquid photosensitive polymers*) yang tersolidifikasi ketika tereksitasi sinar ultraviolet. Model dibangun diatas *platform* (*bed*) yang dikelilingi cairan polimer, dengan intensitas energi yang rendah sinar laser UV membuat lapisan pertama, kemudian cairan dengan sendirinya tersolidifikasi ketika laser UV meninggalkan area cairan. Kemudian, *elevator* sedikit demi sedikit turun, sehingga *platform* terendam

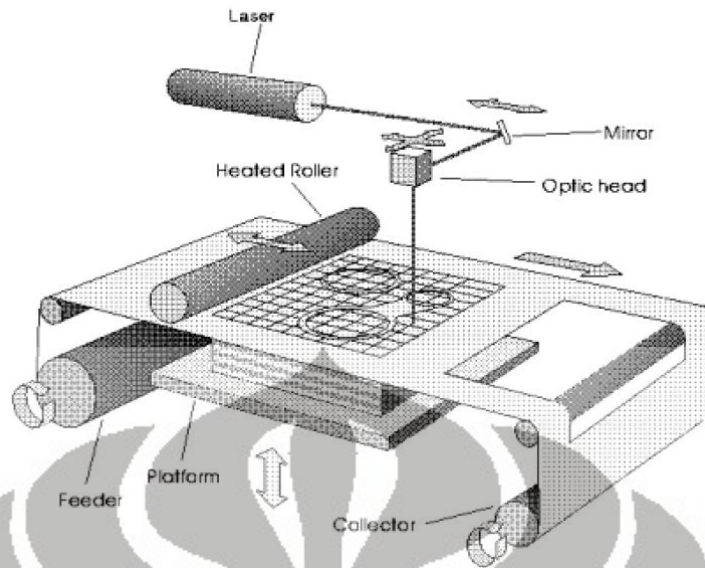
kedalam cairan polimer. *Sweeper* bergerak untuk menutupi lapisan yang sudah tersolidifikasi dengan cairan polimer, kemudian sinar laser UV kembali membuat lapisan kedua diatas lapisan pertama. Proses terus berlanjut sampai prototipe terbentuk [3].

Teknik SLA ini telah sukses mengkonversi desain model CAD menjadi *cross-sectional build layer* dengan ketebalan dari 0.05 sampai 0.15 milimeter. Serta teknik ini dapat membuat model dengan akurasi tertinggi dan permukaan akhir terbaik diantara teknologi *rapid prototyping* yang lain. *Part* model bersifat tahan lama dan dimensi-nya sangat akurat, dengan variasi di antara 0.002 dan 0.005 inci (*Rapid Prototyping Centre, 2006*) [5].

2.3.2 Laminated Object Manufacturing

Teknik *Laminated Object Manufacturing* dikembangkan oleh Helisys dari Torrance, CA. Layer terbuat dari material lembaran yang bersifat *adhesive (adhesive-coated sheet material)* kemudian digabungkan secara bersama sehingga menjadi prototipe. Material aslinya terdiri dari kertas yang dilapisi dengan lem aktif ketika dipanaskan (*heat-activated glue*), dan kertas digulung pada *roll*. Mekanisme pengumpan (*feeder*) dan pengumpul (*collector*) berfungsi untuk mengatur posisi kertas diatas *platform*, dimana sebagai dasar peletakan untuk membuat lapisan pertama. Kemudian, *roller* yang dipanaskan (*heated roller*) memberikan tekanan pada kertas sehingga menempel pada *platform*. Setelah itu, laser memotong keliling terluar dari lapisan pertama pada kertas sehingga terbentuk *cross-hatches (negative space)*. *Cross-hatching* ini memotong area ekstra dan sangat mudah dibelah pada proses *post-processing*. Proses tersebut diulang secara kontinu bersamaan dengan turunnya *platform* sehingga terbentuk prototipe [3].

Helisys mengembangkan beberapa jenis material lembaran baru, termasuk plastik, kertas *water-repellent*, keramik, dan *metal powder tapes*. Memasuki tahun 2001, Helisys sudah tidak lagi bisnis dibidang *rapid prototyping*.



Gambar 2.5 Skema diagram *Laminated Object Manufacturing* [3]

2.3.3 Selective Laser Sintering



Gambar 2.6 Skema diagram *Selective Laser Sintering* [3]

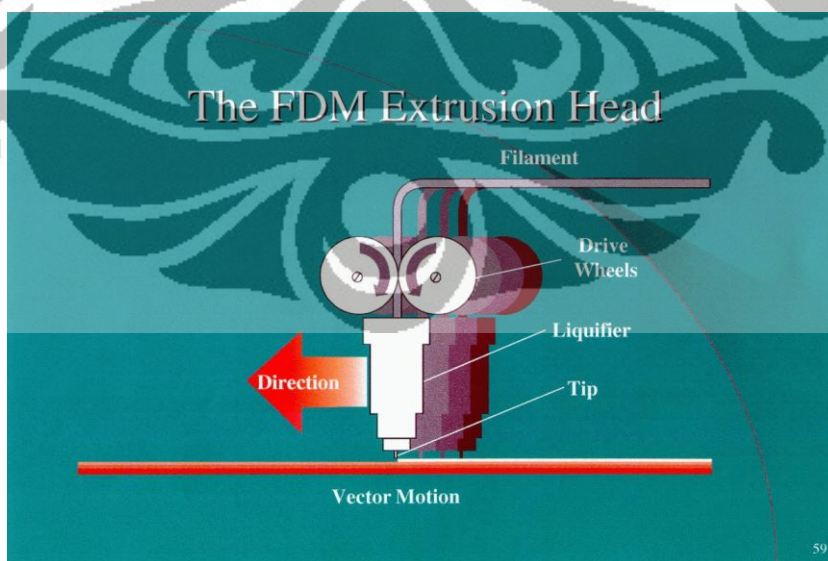
Dikembangkan oleh Carl Deckard saat beliau mengambil gelar Master di Universitas Texas. Teknik ini kemudian dipatenkan pada tahun 1989. Teknik ini menggunakan sinar laser untuk menyatukan serbuk material secara selektif, seperti nilon, elastomer, dan metal menjadi objek solid. Part dibangun diatas *platform* dimana permukaan atasnya diberikan bubuk yang akan menyatu jika diberikan panas (*heat*

fusible powder). Sinar laser akan membuat pola yang akan membentuk lapisan pertama. Kemudian *platform* akan turun sesuai dengan ketebalan lapisan berikutnya dan bubuk disinari laser kembali untuk membentuk lapisan kedua. Proses berjalan secara kontinu sampai terbentuk prototipe [3].

2.3.4 Fused Deposition Modeling

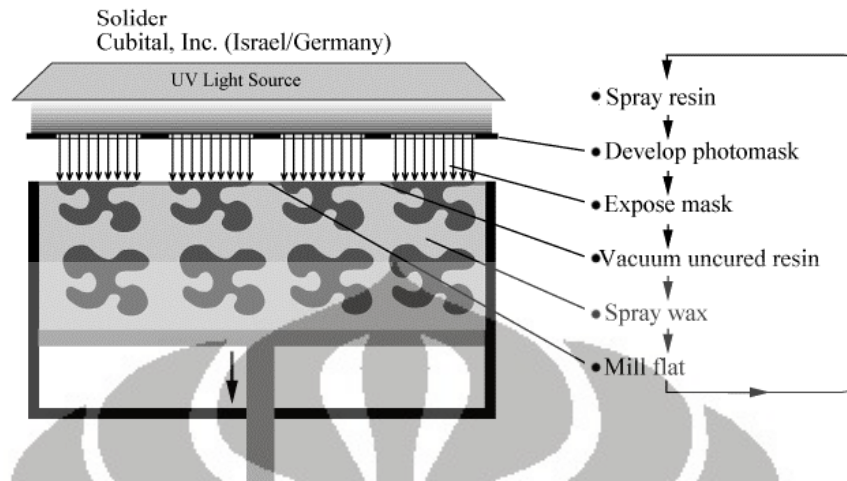
FDM (*Fused Deposition Modeling*) merupakan teknik *rapid prototyping* yang menggunakan prinsip pemanasan material termoplastik. Material termoplastik dapat berupa ABS (*Acrylonitrile Butadiene Styrene*), nilon, *polyurethane*, *polycarbonate*, dan lainnya. Material masuk ke dalam ekstruder yang diberikan panas, kemudian material keluar dari *nozzle* sesuai dengan diameter keluaran. Material akan tersolidifikasi dengan cepat pada *platform*. Ekstruder tersebut bergerak searah bidang x dan y dan membentuk pola untuk lapisan pertama, kemudian *platform* turun sesuai dengan ketebalan lapisan berikutnya, dan ekstruder bergerak sesuai pola untuk membentuk lapisan berikutnya. Proses tersebut berlanjut sampai akhirnya terbentuk prototipe.

Penelitian *rapid prototyping* ini, menggunakan metode FDM. Dikarenakan pembuatannya yang relatif mudah dan dapat diaplikasikan sesuai dengan kebutuhan dan juga material yang dibutuhkan tidak terlalu sulit didapatkan.



Gambar 2.7 Skema diagram *Fused Deposition Modeling* [3]

2.3.5 Solid Ground Curing



Gambar 2.8 Skema diagram *Solid Ground Curing* [3]

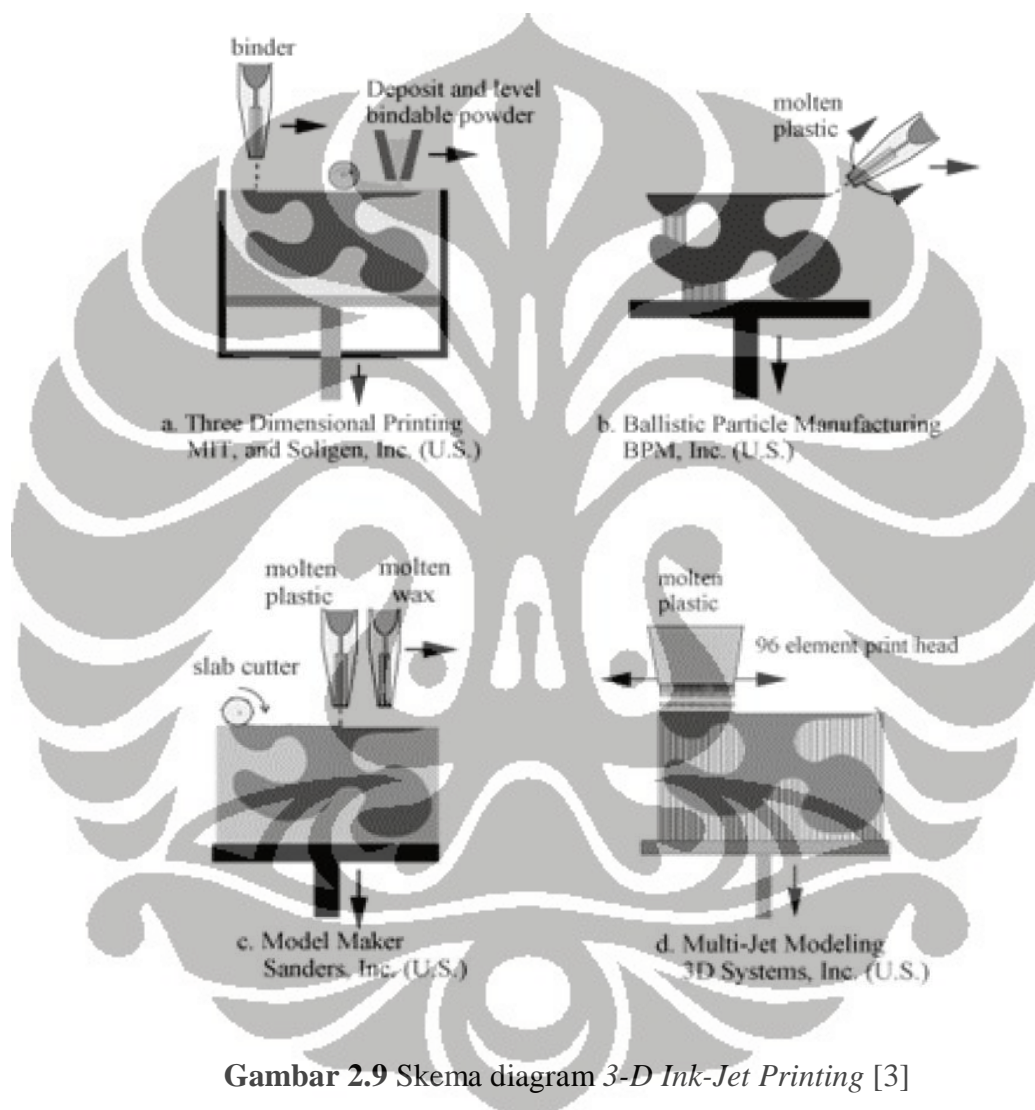
Dikembangkan oleh Cubital, teknik *Solid Ground Curing* (SGC) ini mirip dengan *stereolithography* dimana keduanya menggunakan sinar UV untuk mengeraskan *photosensitive polymer* secara selektif. Pertama-tama, *photosensitive resin* disemprotkan ke *platform (bed)*. Selanjutnya, mesin membuat *photomask* (seperti *stencil*) diatas lapisan yang akan dibuat. *Photomask* ini dicetak pada pelat kaca diatas *platform* menggunakan proses elektrostatis seperti pada mesin fotokopi. *Mask* tersebut disinari sinar UV, sinar yang dapat lolos hanya dapat melewati bagian transparan dari *mask*, kemudian sinar tersebut mengeraskan *photosensitive polymer* dan lapisan pun terbentuk [3].

2.3.6 3-D Ink-Jet Printing

3D Printing (3DP) pertama, dikembangkan oleh MIT dan dilisensikan kepada Soligen Corporation. *3-D Ink-Jet Printing* adalah salah satu teknologi yang telah banyak disebarluaskan dan digunakan dalam berbagai bidang produk percetakan.

Langkah awal dari teknik ini adalah membangun prototipe diatas *platform* dengan menaburkan material bubuk dari wadah secara merata. Selanjutnya, *ink-jet printing head* secara selektif mencetak dengan mengeluarkan cairan (*binder fluid*), sehingga material bubuk tersebut menyatu yang dilewati oleh *head*. Material yang tidak

menyatu akan menjadi material pendukung. *Platform* akan turun, material ditambah dan proses berulang sampai terbentuk model prototipe. Setelah selesai, prototipe dipisahkan dari material pendukung, dan dilapisi oleh *wax*, *CA glue*, atau bahan yang dapat menambahkan ketahanan dan menghaluskan permukaan. Dengan teknik ini tipikal ketebalan lapisan-nya sekitar 0.1 mm. Proses ini sangat cepat dan menghasilkan *part* prototipe dengan permukaan yang bergranular (*grainy surface*) [3].



2.4 Material termoplastik

Material termoplastik merupakan material yang sudah secara luas digunakan untuk *rapid prototyping*. Tabel 2.2 memberikan informasi berbagai karakteristik polimer yang telah digunakan didalam teknologi *rapid prototyping*.

Tabel 2.2 Karakteristik material termoplastik [7]

Material	Structure	Specific Gravity	Rockwell Hardness	UTS Mpa	Elongation of Fracture	Modulus		Impact Notched Izod (J/cm)	Temperature (°C)		
						Tensile Gpa	Flexture Gpa		Deflection at 460 kPa	Glass	Melting
ABS	A	1.05	110 R	42	27%	2.4	2.4	2.5	68-140	107-115	
Acrylic	A	1.18	91 M	68.7	6%	3.1	3.3	0.16	60-103	100-105	130
Nylon	S	1.12	110 R	73.1	83%	2.1	2.4	1.5	85-245		250
Polycarbonate	A	1.21	120 R	69.4	96%	2.6	2.4	6.8	128-174	145-148	
Polyester											
PBT	A,S	1.42	120 R	57.1	36%	2.7	2.9	2.1	95-225		220
PET	A,S	1.32	110 R	55	130%	2.7		1.4	68-72	73-78	250
Polyethylene											
LDPE	S (-55°C)	0.92	60 R	11	190%	0.21	0.27	3.9	40-67		110
HDPE	S (-90°C)	0.96	63 R	20.3	380%	0.91	1.1	1.9	60-104		130
Polypropylene	S	0.94	96 R	36.8	120%	1.9	1.4	0.98	13-238		160
Polyvinylchloride											
Rigid	A	1.4		40	60%		3	2.75	62	75-105	200
Flexible	A			13	320%				62	75-105	

2.4.1 ABS polimer

Polimer ini merupakan struktur *amorphous* dan terbentuk dari tiga buah monomer: *Acrylonitrile* (C_3H_3N), *Butadine* (C_4H_6), dan *Styrene* (C_8H_8). Kombinasi dari monomer-monomer tersebut terdiri dari formasi dua fasa co-polimer yang berbeda untuk membentuk ABS polimer. fasa pertama adalah *hard styrene-butadine co-polymer* dan yang kedua adalah *rubbery styrene-acrylonitrile co-polymer*. ABS polimer sudah digunakan di beberapa aplikasi seperti automotif, pralatan elektronik, dan aplikasi lainnya.

Polimer mempunyai beberapa karakteristik yang sangat diperlukan sekali seperti kekuatan (*strength*) yang baik dan ketahanan (*toughness*) yang relatif tinggi. Karakteristik-nya dapat dimanupulasi dengan mengubah komposisi dari salah satu monomer. ABS digunakan pada teknologi RP seperti *stereolithography* (STL), *fused deposition modeling* (FDM), dan *selective laser sintering* (SLS) [7].

2.4.2 Acrylics

Acrylics adalah polimer dengan struktur *amorphous* yang didapatkan dari *acrylic acid*. Material ini memiliki transparansi yang baik, yang mana dapat meneruskan 90% dari total cahaya. Hal ini membuat mereka menjadi kandidat utama untuk menggantikan kaca; bagaimanapun juga, mereka memiliki ketahanan yang rendah terhadap goresan daripada kaca. *Acrylic* sudah tersedia dengan berbagai macam warna dan salah satu contohnya adalah *Plexiglas*. Mereka digunakan di automotif dan aplikasi peralatan optik. *Acrylic* digunakan untuk membuat prototipe dengan menggunakan teknologi STL [7].

2.4.3 Cellulose

Cellulose adalah polimer alami dengan struktur *amorphous* dan komposisi kimianya adalah $C_6H_{10}O_5$. Kayu, yang mana komposisinya terdiri dari 50% *cellulose*, adalah salah satu sumber utama dari polimer ini. Ketika subjek dipanaskan, *cellulose* terdisintegrasi terlebih dahulu untuk mencapai temperatur lelehnya. Oleh karena itu, *cellulose* membutuhkan kombinasi dengan material lain agar supaya memproduksi termoplastik dengan kelakuan panas yang diinginkan.

Empat grup dari *cellulose* berbasis polimer yang telah tersedia dipasaran :

- 1) *Cellulose acetate*
- 2) *Cellulose acetate butyrate*
- 3) *Cellulose acetate propionate*
- 4) *Cellulose nitrate*

Cellulose digunakan pada *laminated object manufacturing (LOM)* didalam teknologi *rapid prototyping* [7].

2.4.4 Nilon

Nilon adalah anggota dari keluarga *polyamide* (PA) dan kebanyakan memiliki struktur *crystalline*. Kebanyakan nilon yang digunakan adalah *Nylon6 (PA6)*, *Nylon6,6* dan *(PA6,6)*. Nilon memiliki *wear resistance* yang baik dan kekuatan mereka dapat ditingkatkan dengan *reinforcing* mereka dengan *glass fiber*. Nilon adalah material

prototyping utama, digunakan pada laser sintering (LS) teknologi *rapid prototyping* dan juga digunakan pada teknologi FDM [7].

2.4.5 Polycarbonate

Polycarbonate adalah polimer dengan komposisi kimia $[C_3H_6(C_6H_4)_2CO_3]_n$. Mereka memiliki struktur *amorphous* dan memiliki karakteristik *creep resistance* yang baik dan ketahanan yang baik. Mereka memiliki hambatan terhadap panas yang sangat baik jika dibandingkan polimer yang lain. Mereka digunakan di aplikasi *automotive windshield* sama juga seperti produksi *housing*. Mereka digunakan untuk aplikasi *prototyping* STL [7].

2.4.6 Termoplastik Polyester

Polyester adalah polimer yang terdiri dari struktur *semi-crystalline* dan juga terdiri dari dua tipe : *thermoplastic polyester* dan *thermosetting polyester*. Dua tipe *polyester* yang kebanyakan digunakan adalah *polybutylene terephthalate* (PBT) dan *polyethylene terephthalate* (PET). PBT digunakan di bidang manufaktur *automotive luggage rack* dan komponen *headlight*. PET digunakan dibidang pengepakan, automotif, dan industri elektronik disebabkan sifat ketahanan dan hambatan temperatur yang tinggi. *Polyester* digunakan di teknologi *rapid prototyping* SLS[7].

2.4.7 Polyethylene (PE)

Polyethylene memiliki struktur *semi-crystalline* dan komposisi kimianya adalah $[C_2H_4]_n$. *Polyethylene* memiliki ketahanan yang baik dan daya tahan terhadap reaksi kimia yang sangat baik. PE dapat diproses dengan metode produksi termoplastik apa saja, sehingga dapat menggunakan berbagai macam tipe termoplastik.

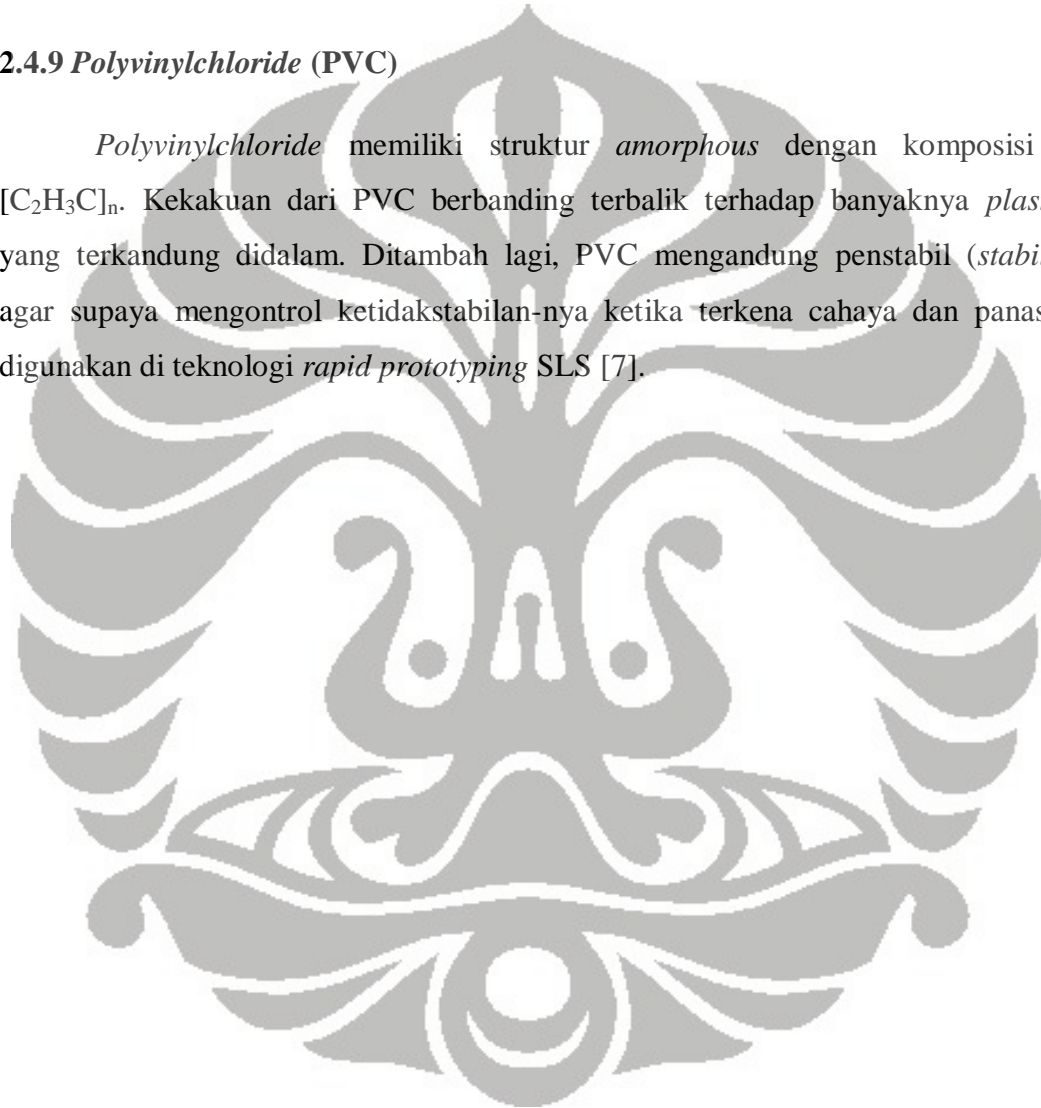
Polyethylene mempunyai dua tipe: *low density polyethylene* (LDPE) dan *high density polyethylene* (HDPE). Perbedaan karakteristik dari kedua tipe tersebut disebabkan dari perbedaan struktur karena massa jenis secara langsung berbanding lurus terhadap derajat kekristallan (% *crystalline*) dari struktur material. *Polyethylene* digunakan di teknologi FDM [7].

2.4.8 Polypropylene (PP)

Polypropylene memiliki struktur *semi-crystalline* dengan derajat kekrystallan yang tinggi. PP, yang mana memiliki komposisi kimia $[C_3H_6]_n$, merupakan plastik yang tersedia saat ini. PP memiliki daya tahan terhadap reaksi kimia yang baik dan karakteristik-nya sebanding dengan HDPE. PP digunakan di dalam manufaktur pembuatan plastik *hinges*. PP digunakan di teknologi *rapid prototyping* FDM [7].

2.4.9 Polyvinylchloride (PVC)

Polyvinylchloride memiliki struktur *amorphous* dengan komposisi kimia $[C_2H_3Cl]_n$. Kekakuan dari PVC berbanding terbalik terhadap banyaknya *plasticiezer* yang terkandung didalam. Ditambah lagi, PVC mengandung penstabil (*stabilisator*) agar supaya mengontrol ketidakstabilan-nya ketika terkena cahaya dan panas. PVC digunakan di teknologi *rapid prototyping* SLS [7].



BAB 3

TINJAUAN SISTEM MEKANIKAL DAN EKSTRUDER RPFDM

Sistem mekanikal mesin *rapid prototyping* berbasis FDM atau disebut juga RPFDM merupakan suatu mekanisme robot cartesian, dimana memiliki 3 sumbu utama, yaitu sumbu x, sumbu y, dan sumbu z, ketiga sumbu merupakan lengan *prismatic joint* (*linier joint*). Desain ketiga sumbu menggunakan sistem *ballscrew*, dan digerakkan oleh motor stepper. Berikut dibawah ini gambar penampakan mesin RPFDM yang telah dibuat.

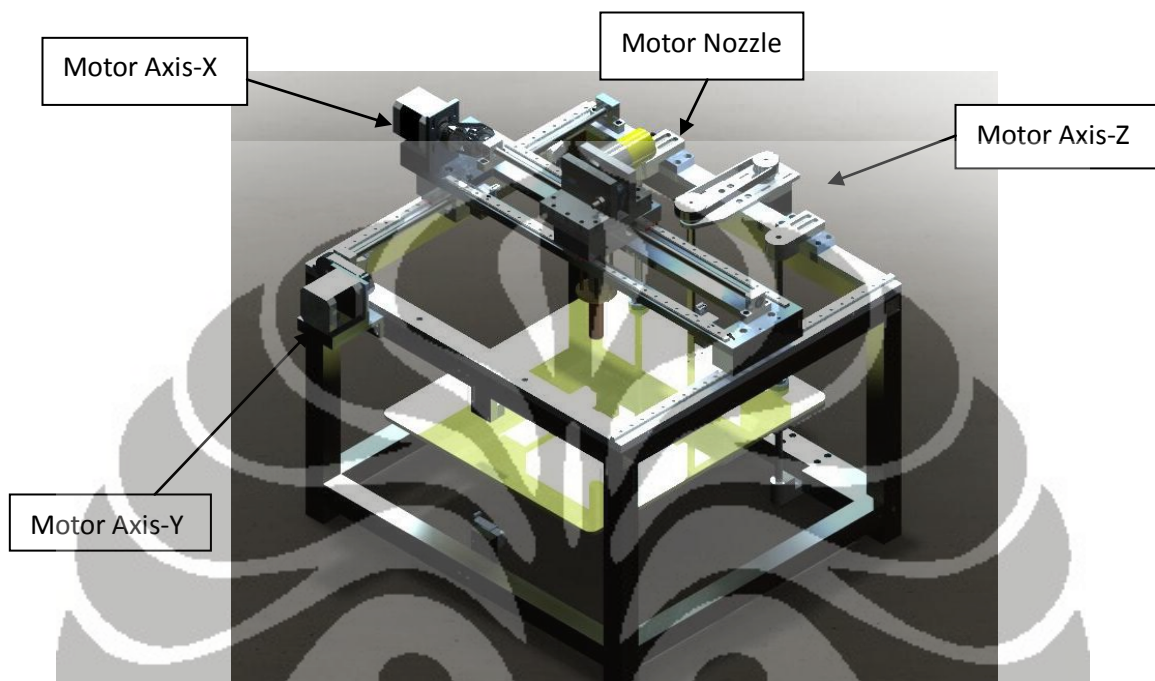


Gambar 3.1 Mesin RPFDM tiga axis

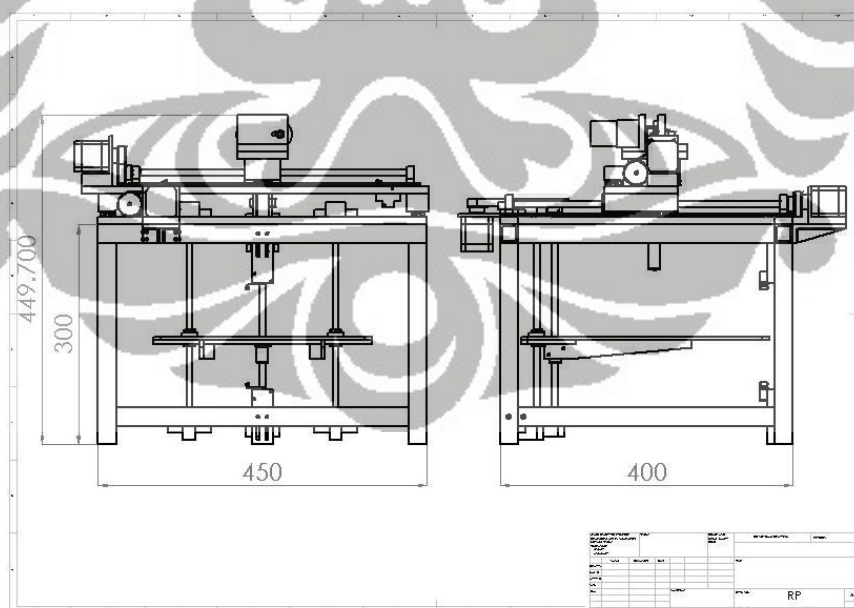
3.1 Desain mesin RPFDM

Desain mesin ini, konsepnya merupakan robot cartesian yang memiliki tiga sumbu yaitu x, y, dan z axis. Mekanisme penggerak sumbu x dan y menggunakan dua buah linier *guide* dan satu buah *shaft ballscrew* sebagai lengan *prismatic joint*.

Sedangkan penggerak sumbu z menggunakan dua buah silinder *guide* dan satu buah *shaft ballscrew*. Ketiga *shaft* digerakkan oleh motor stepper, dengan *pulley ratio* 5:4. Berikut gambar 3D desain dari mesin RPFDM.



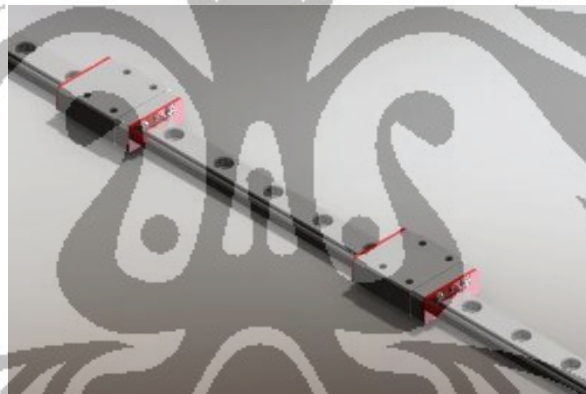
Gambar 3.2 Desain mesin RPFDM



Gambar 3.3 Dimensi mesin 450x400x449.7

3.2 Linier *Guideway*

Desain pada sumbu x dan y menggunakan sistem linier *guideway*, linier *guideway* adalah suatu mekanisme yang berfungsi sebagai tumpuan yang dapat digerakkan dengan koefisien gesek yang sangat kecil. Mekanisme linier *guideway* sudah menggunakan sistem *ball* sehingga membuat koefisien gesek menjadi sangat kecil. Oleh karena itu, pada arah sumbu x dan y digunakan linier *guideway* sebagai tumpuan dan dapat digerakkan ke arah tertentu. Tipe linier *guideway* yang digunakan adalah tipe MGN7C yang diproduksi oleh HIWIN. Tipe ini dipilih karena kemampuan dapat menahan tumpuan beban statik sampai dengan 127 kgf. Tipe MGN, sebenarnya merupakan tipe yang mini dari tipe linier *guideway* lain-nya, karena bentuknya yang mini sehingga harga-nya berbanding terbalik dengan ukuran dimensinya. Linier *guideway* ini sangat presisi untuk pergerakan linier, dan memiliki koefisien gesek yang sangat kecil yaitu 1/50 th.

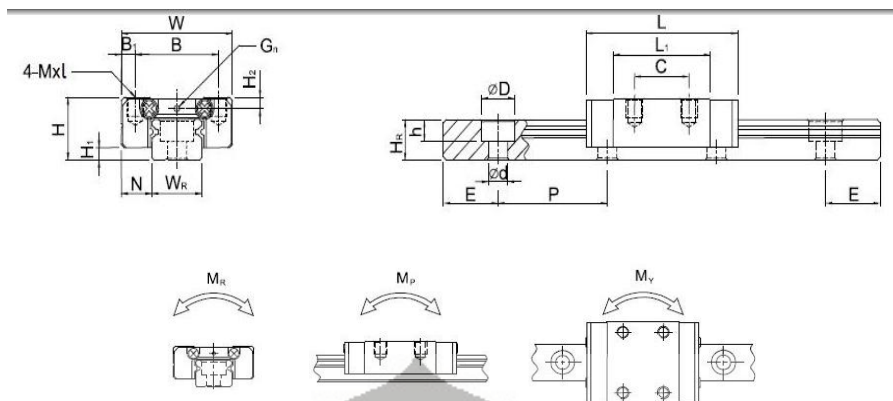


Gambar 3.4 Tipe MGN7C HIWIN

Berikut fitur dari linier *guideway* tipe MGN :

1. Kecil dan memiliki beban yang ringan, cocok untuk peralatan miniatur.
2. Semua material untuk *block* dan *rail*-nya terbuat dari *special grade stainless stell*, dimana sudah terdapat *stell ball*, dan *ball retainer* untuk tujuan anti korosi.
3. Rigiditas dan akurasi yang tinggi.
4. *Stell ball* akan tetap pada tempatnya, jika *block* dilepas dari *rail*.
5. Penggantian dengan tipe yang lain, tersedia.

Berikut dimensi dari tipe MGN :



Gambar 3.5 Dimensi tipe MGN [8]

Berikut table spesifikasi tipe MGN :

Table 3.1 Spesifikasi tipe MGN [8]

Model No.	Dimensions of Assembly (mm)											Dimensions of Block (mm)											Dimensions of Rail (mm)											Mounting Bolt for Rail (mm)	Basic Dynamic Load Rating C ₀ (kgf)	Basic Static Load Rating C ₁ (kgf)	Static Rated Moment			Weight	
	H	H ₁	N	W	B	B ₁	C	L ₁	L	G	G ₁	MxL	H ₂	W ₂	H ₂	D	h	d	P	E	M _R	M _P	M _V	Block	Rail																
	kgf-m	kgf-m	kgf-m	kg	kg																																				
MGN 7C	8	1.5	5	17	12	2.5	8	13.5	22.5	-	0.1.2	M2x2.5	1.5	7	4.8	4.2	2.3	2.4	15	5	M2x6	100	127	0.48	0.29	0.29	0.010	0.22													
MGN 7H	-	-	-	-	-	-	13	21.8	30.8	-	-	-	-	-	-	-	-	-	-	-	-	140	200	0.78	0.49	0.49	0.015	-													
MGN 9C	10	2	5.5	20	15	2.5	10	18.9	28.9	-	0.1.2	M3x3	1.8	9	6.5	6	3.5	3.5	20	7.5	M3x8	190	260	1.2	0.75	0.75	0.016	0.38													
MGN 9H	-	-	-	-	-	-	16	29.9	39.9	-	-	-	-	-	-	-	-	-	-	-	-	260	410	2	1.9	1.9	0.026	-													
MGN 12C	13	3	7.5	27	20	3.5	15	21.7	34.7	-	0.1.4	M3x3.5	2.5	12	8	6	4.5	3.5	25	10	M3x8	290	400	2.6	1.4	1.4	0.034	0.65													
MGN 12H	-	-	-	-	-	-	20	32.4	45.4	-	-	-	-	-	-	-	-	-	-	-	-	380	600	3.9	3.7	3.7	0.054	-													
MGN 15C	16	4	8.5	32	25	3.5	20	26.7	42.1	4.5	M3	M3x4	3	15	10	6	4.5	3.5	40	15	M3x10	470	570	4.6	2.2	2.2	0.059	1.06													
MGN 15H	-	-	-	-	-	-	25	43.4	58.8	-	-	-	-	-	-	-	-	-	-	-	-	650	930	7.5	5.9	5.9	0.092	-													

Note : 1 kgf= 9.81 N

3.3 Ballscrew

Mekanisme mekanikal penggerak sumbu x, y, dan z yaitu menggunakan poros *ballscrew*, dimana perpindahan *angular* dari motor stepper dikonversi menjadi gerak linier dengan menggunakan komponen *ballscrew* ini. *Ballscrew* yang digunakan merupakan tipe FSI berdiameter luar 8 mm dengan *lead/pitch* 2mm diproduksi oleh HIWIN (Taiwan). Pada desain mesin RPFDM, komponen *shaft ballscrew* tidak didesain menerima beban/gaya dari arah vertikal maupun horizontal, kecuali pada sumbu z. Penggunaan *ballscrew* pada sumbu x dan y hanya untuk mengerakkan komponen secara

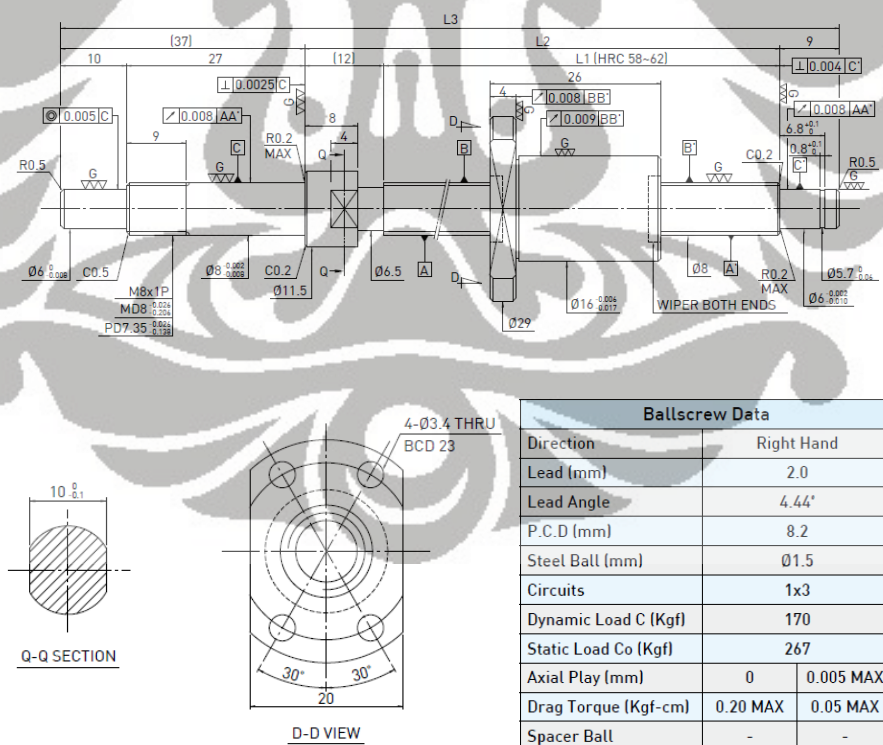
linier. Tujuan utama lainnya adalah agar mendapatkan pergerakan linier yang presisi dan akurasi yang tinggi.

Karakteristik dari *ballscrew* :

1. Efisiensi dan *reversibility* yang tinggi.
2. Pengurangan *backlash* dan *stiffness* (kekakuan) yang tinggi.
3. Akurasi yang tinggi.
4. Umur pakai yang dapat diprediksi.
5. Torsi awal yang rendah dan *smooth running*.



Gambar 3.6 Nut FSI R8-2T3 tampak tiga dimensi



Gambar 3.7 Dimensi dan spesifikasi FSI R8-2T3 [9]

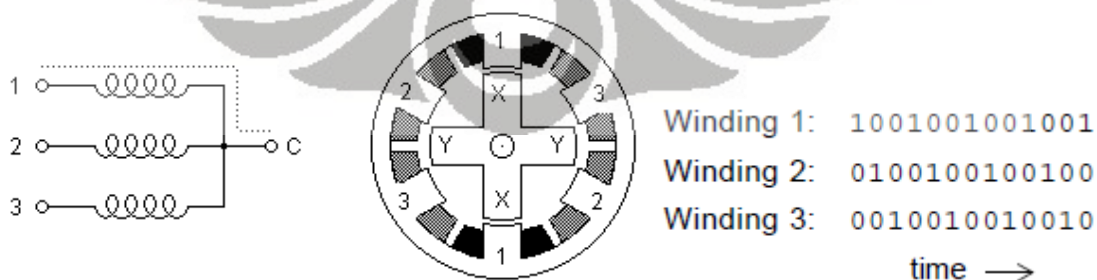
3.4 Motor stepper

Motor stepper prinsip dasarnya sangat berbeda dengan motor DC, motor stepper tidak memiliki komutator pada *rotor*. Motor stepper pada umumnya digunakan untuk aplikasi pengukuran dan kontrol serta membutuhkan akurasi yang tinggi. Contohnya seperti terdapat pada *ink jet printer*, mesin CNC (*Computer Numerically Controlled*), dan penggerak piringan disket atau piringan CD. Pengembangan mesin RPFDM ini, juga membutuhkan akurasi dan presisi yang tinggi, seperti halnya printer.

Tipe motor stepper pada umumnya terbagi menjadi tiga, yaitu, magnet permanen, *variable reluctance*, dan *hybrid*. Tipe magnet permanen memiliki magnet permanen yang terdapat pada *rotor*-nya, sementara *variable reluctance* memiliki gigi *rotor* berbahan besi yang lunak (*soft iron*), sedangkan tipe *hybrid* mengkombinasikan teknologi magnet permanen dan *variable reluctance*. Ditinjau dari segi elektrikal dan sistem kontrol, *variable reluctance* merupakan tipe yang berbeda dari yang lain. Motor magnet permanen dan *hybrid* pada umumnya menggunakan kumparan *unipolar*, *bipolar*, maupun *bifiliar*. [10]

3.4.1 Variable Reluctance

Motor *variable reluctance* disebut juga motor *variable switch reluctance*. Pada umumnya terdapat tiga atau lima kumparan yang terhubung ke terminal. Gambar 3.8, menunjukkan penampang melintang dari tiga kumparan, 30 derajat per step dari *variable reluctance* motor.

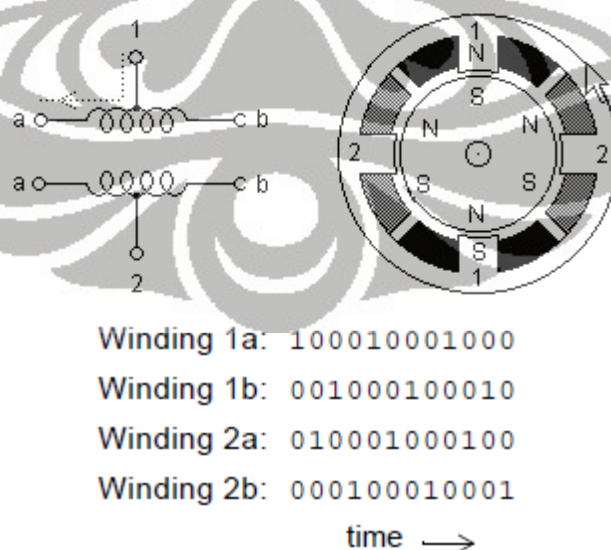


Gambar 3.8 Motor *variable reluctance* [10]

Rotor pada motor ini memiliki empat gigi dan *stator* memiliki enam kutub, yang mana salah satu pasangan kutubnya berlawanan. *Rotor* berlabel X berinteraksi dengan kumparan 1, yang mana kumparan telah diberikan energi listrik. Interaksi ini disebabkan oleh induksi lintasan *flux* magnetik yang dibangkitkan antara koil dan *rotor*. Untuk menggerakkan *rotor* searah jarum jam, maka kumparan 1 dinonaktifkan dan kumparan 2 diaktifkan. Sehingga *rotor* berlabel Y akan berinteraksi dengan kumparan 2, dan menghasilkan gerakan 30 derajat. Untuk melanjutkan gerakan searah jarum jam maka kumparan 3 diaktifkan dan kumparan 2 dinonaktifkan. Sehingga untuk menghasilkan putaran penuh harus mengulang 12 *step/revolution*.

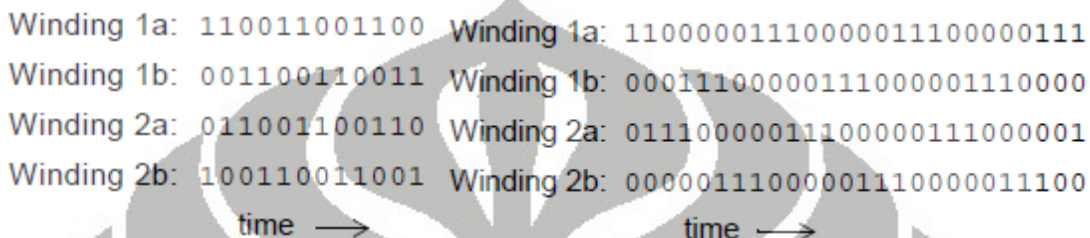
3.4.2 Unipolar

Motor *unipolar* merupakan magnet permanen atau *hybrid* yang pada umumnya terdiri dari dua kumparan, yang mana masing-masing terdapat *center tap*. *Center tap* ini terdapat ditengah-tengah koil dan berupa kabel yang menyatu dengan koil, biasanya bisa terpisah satu sama lain, maupun menyatu menjadi satu. Oleh karena itu, motor *unipolar* terdiri dari 5 atau 6 kabel. Untuk menggerakkan motor *unipolar*, *center tap* dihubungkan dengan sumber tegangan sebagai *ground* (GND), sedangkan 4 kabel yang lain dihubungkan sebagai inputan dari kontroller.



Gambar 3.9 Unipolar motor dan *full step* [10]

Karena kabel 1 dan 2 dihubungkan ke GND sehingga hanya dapat mengalirkan arus dalam satu arah saja. Untuk gerakan *full step*, koil berlabel 1a diaktifkan, sehingga koil aktif dan menjadi magnet dengan kutub yang berlawanan, sehingga terjadi gaya tarik menarik (*magnetic force*) antara *rotor* magnet permanen dengan *stator*. Selanjutnya untuk berputar searah jarum jam, koil berlabel 2a diaktifkan, koil 1a dinonaktifkan, begitu seterusnya sampai mengulang 12 *step/revolution*.



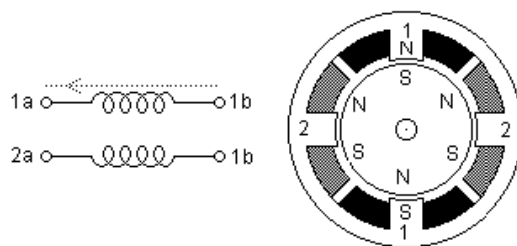
Gambar 3.10 Gerakan *full step* torsi lebih dan *half step* [10]

Untuk mendapatkan torsi lebih besar, konsekuensi metode menggerakannya dengan mengaktifkan kumparan 1a dan 2b secara bersamaan, lihat gambar 3.10. Sedangkan untuk metode *half step*, yaitu dengan menggabungkan metode awal dengan yang kedua. Metode *half step* akan menghasilkan setengah resolusi derajat per step dari gerakan *full step*, dan putaran motor menjadi lebih halus.

3.4.3 Bipolar

Motor *bipolar* adalah motor yang terdiri dari dua kumparan dan memiliki empat kabel input. Tidak seperti motor *unipolar*, motor *bipolar* tidak memiliki *center tap*. Keuntungan tidak adanya *center tap* adalah arus listrik yang melewati kumparan lebih banyak dan tidak terbagi menjadi dua, sehingga torsi motor *bipolar* lebih besar dari pada motor *unipolar*.

Arus listrik yang mengalir pada kumparan dapat dua arah (*bidirectional*), sehingga membutuhkan pergantian polaritas sumber tegangan. Oleh karena itu, sistem kontrol-nya membutuhkan H-bridge agar polaritas tegangan dapat ditukar.



Terminal 1a:	+	0	-	0	+	0	-	0	+	0	-	0
Terminal 1b:	-	0	+	0	-	0	+	0	-	0	+	0
Terminal 2a:	0	+	0	-	0	+	0	-	0	+	0	-
Terminal 2b:	0	-	0	+	0	-	0	+	0	-	0	+
	time →											
Terminal 1a:	+	+	-	-	+	+	-	-	+	+	-	-
Terminal 1b:	-	-	+	+	-	-	+	+	-	-	+	+
Terminal 2a:	-	+	+	-	-	+	+	-	-	+	+	-
Terminal 2b:	+	-	-	+	+	-	-	+	+	-	-	+
	time →											

Gambar 3.11 Motor bipolar dan pergerakan full step [10]

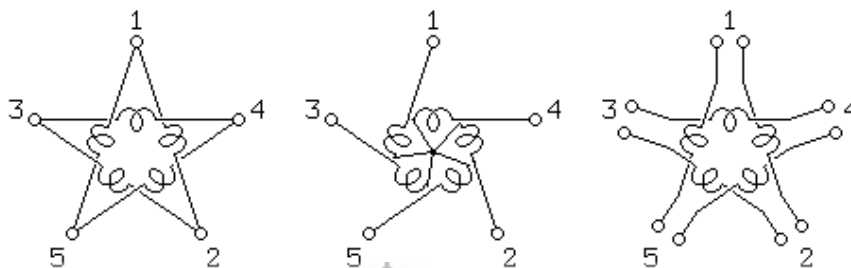
Ilustrasi diatas menggambarkan motor *bipolar* 30 derajat/step. Kumparan 1 terdistribusi di atas dan bawah kutub *stator*, sementara kumparan 2 terdistribusi diantara kiri dan kanan kutub *stator*. Sedangkan *rotor* merupakan magnet permanen dengan 6 kutub, 3 utara dan 3 selatan terdapat disekeliling lingkaran *rotor*. Seperti motor *unipolar*, motor *bipolar* dapat bergerak per-step dengan dua metode. Tanda + dan - digunakan untuk menandakan polaritas dari sumber tegangan yang diaktifkan melewati terminal input, tanda 0 berarti menandakan tidak ada arus listrik yang di-inputkan.

Terminal 1a:	+	+	0	-	-	0	+	+	+	0	-	-	0	+	+	+	0	-	-	0	+	+	+	0	-	-	0	+				
Terminal 1b:	-	-	0	+	+	0	-	-	0	+	+	+	0	-	-	0	+	+	+	0	-	-	0	+	+	+	0	-	-	0	+	
Terminal 2a:	0	+	+	+	0	-	-	0	+	+	+	0	-	-	0	+	+	+	0	-	-	0	+	+	+	0	-	-	0	+		
Terminal 2b:	0	-	-	0	+	+	+	0	-	-	0	+	+	+	0	-	-	0	+	+	+	0	-	-	0	+	+	+	0	-		
	time →																															

Gambar 3.12 Pergerakan half step motor bipolar [10]

Untuk mendapatkan gerakan putaran *half step*, kedua dua metode pada gambar 3.12 dapat digabungkan, sehingga akan menghasilkan 15 derajat per step serta menghasilkan putaran motor yang halus.

3.4.4 Multiphase



Gambar 3.13 Lima fasa motor stepper [11]

Motor *multiphase* merupakan salah satu tipe terupdate dari motor magnet permanen dan *hybrid*, dimana semua kumparan dibuat menjadi sebuah siklus, yang mana terdapat *center tap* di setiap kumparan. Untuk mengontrol motor ini dibutuhkan setidaknya 1 atau 2 H-bridge untuk disetiap terminal dari inputan motor.

Motor lima fasa memiliki kemampuan resolusi mencapai 0,72 derajat / step (*500 step per revolution*). Motor lima fasa ini menghasilkan torsi yang lebih besar dikarenakan empat dari semua kumparan motor stator terinduksi disetiap point dari siklus perputaran.

Pada motor lima fasa, terdapat 10 langkah yang diulang didalam siklus perputaran;

```
Terminal 1 +++-----+++++-----++
Terminal 2 ---++++-----+++++---
Terminal 3 +-----+++++-----++++
Terminal 4 +++-----+++++-----
Terminal 5 ---++++-----+++++
time --->
```

Pada motor tiga fasa, terdapat 6 langkah yang diulang didalam siklus perputaran;

```
Terminal 1 +++-----
Terminal 2 ---++++-----
Terminal 3 +-----++
time --->
```

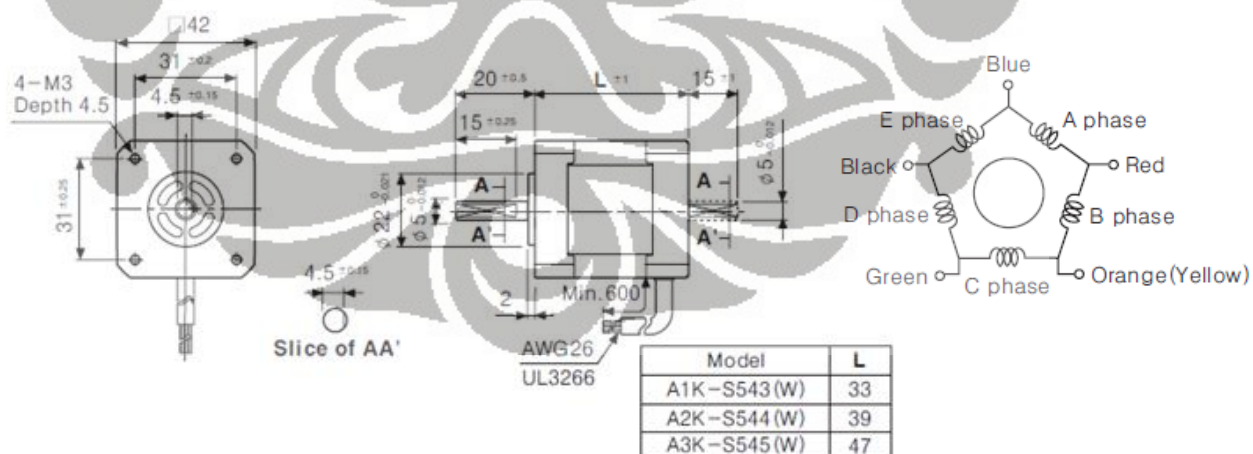


Gambar 3.14 Motor stepper tipe A1K-S543(W)

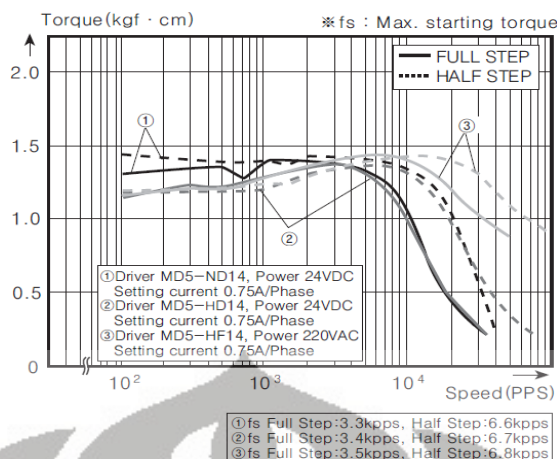
Pada mesin RPFDM, menggunakan tiga buah motor stepper lima fasa yang diproduksi oleh Autonics dengan tipe A1K-S543(W) dengan spesifikasi yang sama, berikut spesifikasi motor tersebut;

Tabel 3.2 Spesifikasi motor stepper lima fasa [12]

Model	A1K-S543(W)-□	A2K-S544(W)-□	A3K-S545(W)-□
Max. Holding torque(*1)	1.3[kgf · cm] 0.13[N · m]	1.8[kgf · cm] 0.18[N · m]	2.4[kgf · cm] 0.24[N · m]
Rotor moment of inertia	35[g · cm ²] 35 × 10 ⁻⁷ [kg · m ²]	54[kg · cm ²] 54 × 10 ⁻⁷ [kg · m ²]	68[g · cm ²] 68 × 10 ⁻⁷ [kg · m ²]
Rated current	0.75 [A/Phase]		
Basic step angle	0.72° /0.36° (Full step/Half step)		
Insulation-class	IEC34-5		
Protection	IP30		
Weight	0.25kg	0.3kg	0.4kg



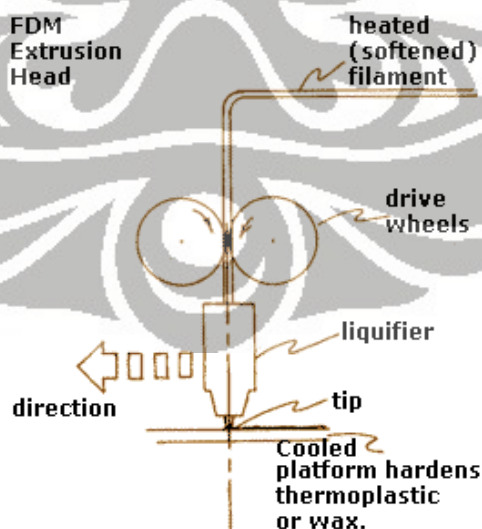
Gambar 3.15 Dimensi motor stepper lima fasa [12]



Gambar 3.16 Torsi vs kecepatan motor stepper lima fasa [12]

3.5 Konsep mekanisme ekstruder

Dalam pengembangan mesin RPFDM, metode yang digunakan adalah metode FDM (*Fused Deposition Modeling*). FDM adalah teknik *rapid prototyping* dimana menggunakan teknik pemanasan material termoplastik pada temperatur lelehnya, kemudian material yang ter-*liquifier* dikeluarkan dengan teknik ekstrusi, kemudian proses pembentukan prototipe dengan proses pembentukan lapis per lapis melalui mekanisme robot cartesian.



Gambar 3.17 Skema diagram FDM [13]

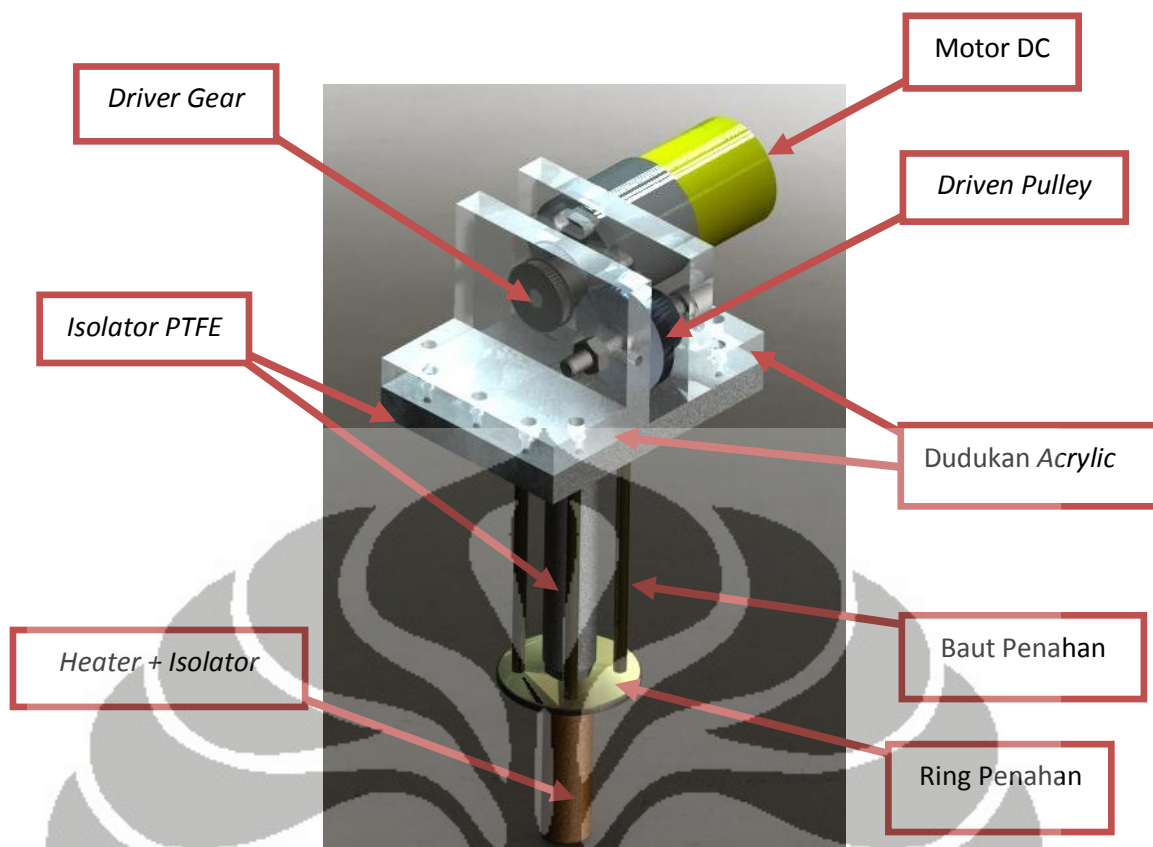
Metode FDM, sangat mudah untuk dibuat dan diterapkan jika ditinjau dari segi komponen dan sistem mekanikal dari keseluruhan mekanisme. Pada mesin RPFDM tidak menggunakan material pendukung, sehingga masih terbatas dalam membuat prototipe dalam bentuk yang sangat rumit. Komponen-komponen penyusun mekanisme sistem ekstruder ini, terdiri dari *feeder wheels*, sistem pemanas, dan sistem *isolator* agar kalor tidak menghambat kinerja sistem penggerak seperti *ballscrew*.

3.6 Konstruksi mekanisme ekstruder

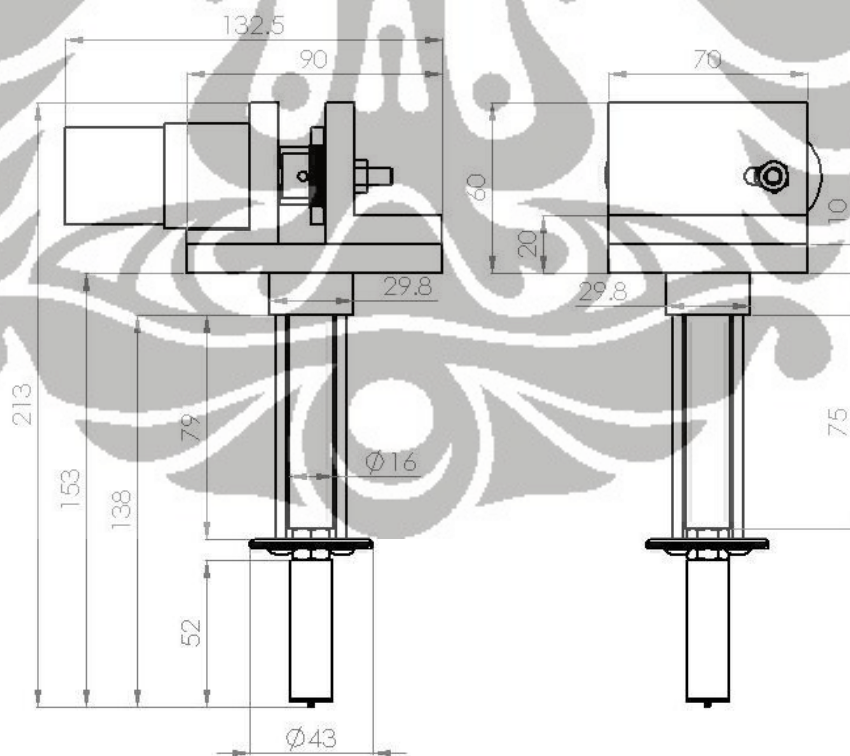
Mekanisme sistem ekstruder didesain berdasarkan metode *fused deposition modeling*. Sehingga secara keseluruhan sistem konstruksi ekstruder material dapat dilihat pada gambar dibawah ini :



Gambar 3.18 Konstruksi mekanisme ekstruder



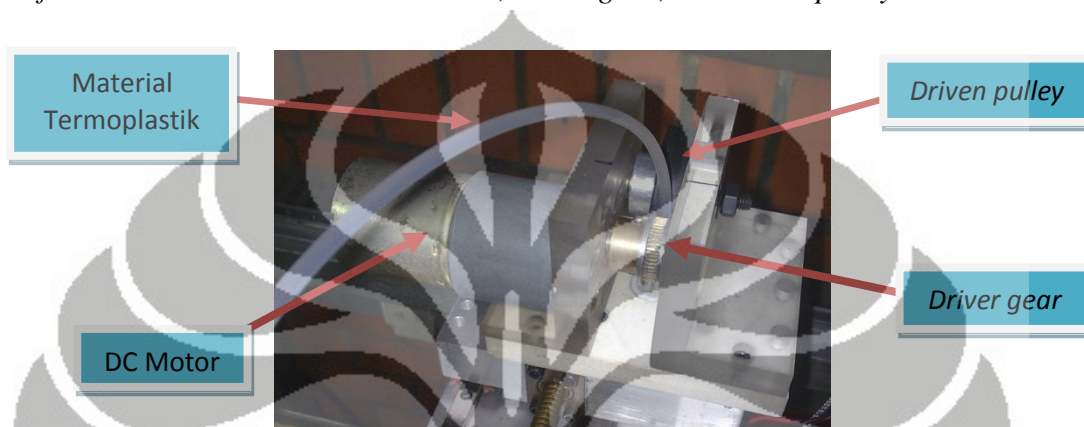
Gambar 3.19 Desain dari mekanisme ekstruder



Gambar 3.20 Dimensi dari *assembly* mekanisme ekstruder

3.6.1 Feeder wheels

Feeder wheels merupakan bagian dari mekanisme sistem pengumpan material pada mekanisme ekstruder. *Feeder wheels* bertujuan agar memberikan gaya dorong supaya material termoplastik terdorong masuk kedalam tabung pemanas (*barrel*), kemudian material dilelehkan didalam tabung pemanas tersebut. Komponen-komponen dari *feeder wheels* terdiri dari motor DC, *driver gear*, dan *driven pulley*.

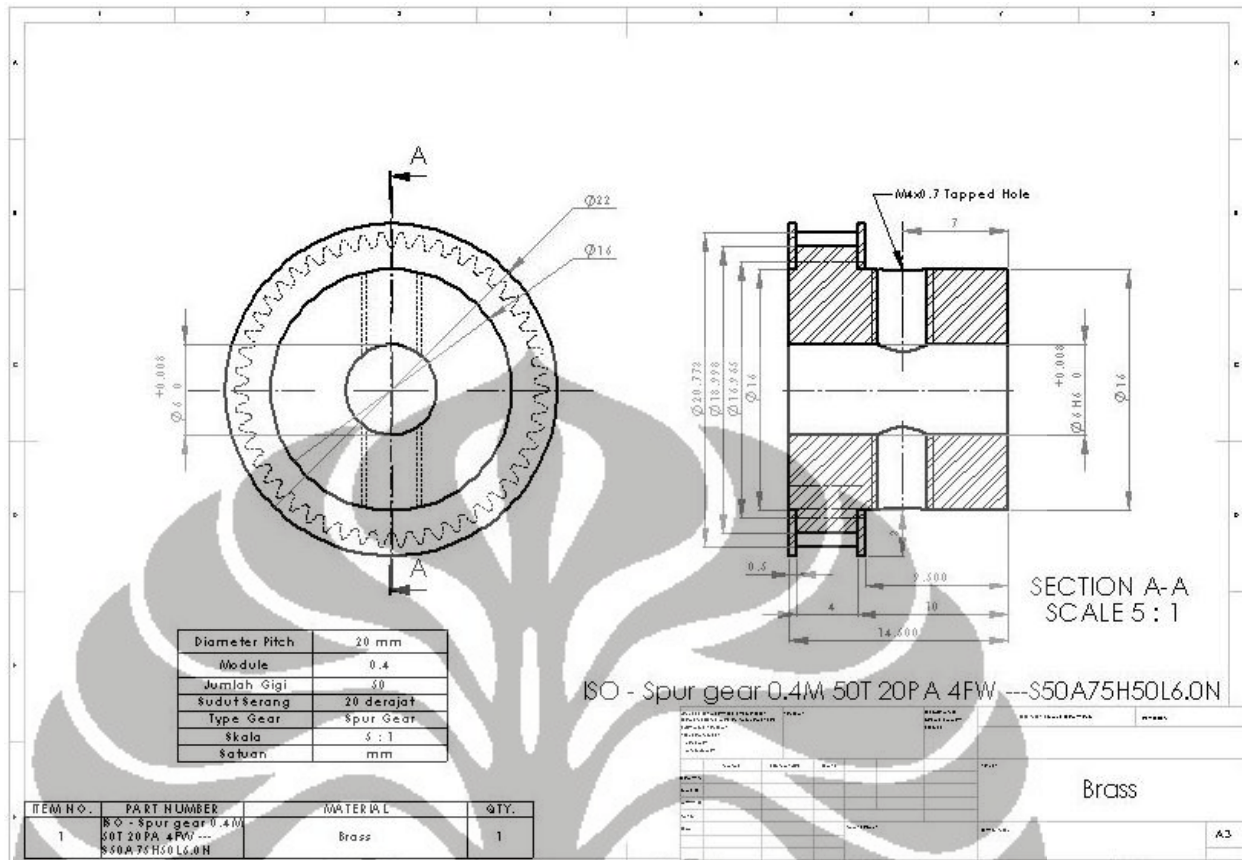


Gambar 3.21 Mekanisme *feeder wheels*

Untuk menggerakkan *driver gear*, digunakan motor DC yang memiliki *gear box*, sehingga dengan memasukkan input tegangan 12V putaran motor menghasilkan keluaran putaran sebesar 10 rpm. Motor DC tersebut dapat dikontrol dengan kecepatan putarannya dengan metode PWM.

Driver gear merupakan *spur gear* yang diberikan *flange* pada kedua sisinya. *Spur gear* didesain memiliki diameter *pitch* 20 mm, modul sebesar 0,4, dan sudut serang 20° . *Driver gear* digerakkan secara langsung oleh motor DC dan terbuat dari material brass yang lebih kuat dari aluminium, sehingga material dapat tergigit dan terdorong kedalam tabung pemanas.

Sedangkan *driven pulley* berfungsi untuk mendukung *driver gear* dan menerima gaya tekan dari *driver gear* agar material terdorong ke dalam tabung pemanas. *Driven pulley* didesain menggunakan bearing pada porosnya agar dapat berputar secara bebas dan mengikuti arah putar *driver gear*. Pada keliling *driven pulley* dilapisi material karet bertujuan agar material yang terdorong tidak mengalami slip dengan meningkatkan koefisien gesek.

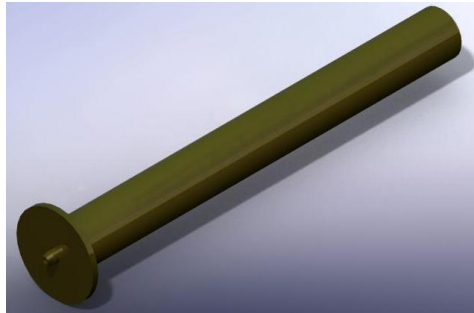


Gambar 3.22 Desain dari *driver gear*

3.6.2 Sistem pemanas (*barrel*)

Barrel merupakan sebuah tabung yang dipanaskan, cara dipanaskannya yaitu dengan cara dililitkannya kawat nikrom pada sisi permukaan luar tabung tersebut. Komponen ini berfungsi untuk memanaskan material termoplastik pada temperatur leleh. Pada ujung komponen ini terdapat *nozzle* dengan diameter keluaran 1 mm dan ada juga dengan diameter keluaran 0,5 mm.

Komponen ini didesain memiliki panjang total 75 mm, berdiameter luar 8 mm, dan diameter dalam sebesar 3,5 mm. Keseluruhan permukaan luar komponen ini terdiri dari ulir luar dengan diameter luar nominal 8 mm dengan jarak puncak ke puncak 1,25 mm. *Barrel* terbuat dari material brass yang memiliki termal konduktivitas 110 W/mK, merupakan termal konduktivitas yang cukup baik untuk dijadikan sebagai konduktor.



Gambar 3.23 Desain dari komponen barrel

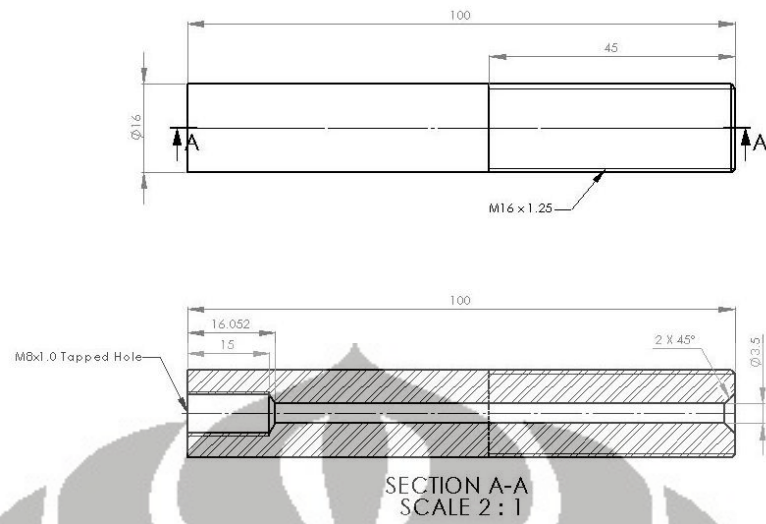
Sistem pemanas pada sistem *barrel* ini, menggunakan kawat nikrom dengan spesifikasi hambatan $10\Omega/m$. Nikrom merupakan material *alloy* campuran 80% nikel dan 20% khromium. Kawat nikrom menggunakan prinsip hambatan untuk menghasilkan panas. Temperatur leleh nikrom ini sebesar 1400°C . Kawat ini dililit langsung pada *barrel*, setelah dilapisi oleh mika sebagai media isolator supaya tidak terjadi arus pendek (*short*).

3.6.3 Sistem isolator

Pada sistem *isolator*, pada intinya adalah mencegah panas pada *barrel* tidak mengalir ke lingkungan dan terbuang percuma. Fungsi yang lain adalah agar mencegah panas mengalir ke komponen dan memberikan efek terhadap kinerja komponen tersebut. Komponen isolator yang digunakan adalah *mica wool*, *mica tape*, dan PTFE.

PTFE (*polytetrafluoroethylene*) merupakan isolator yang berfungsi untuk menyerap kalor supaya kalor dari *barrel* tidak mengalir ke komponen penggerak, seperti *ballscrew*, *linier guideway*, dan komponen-komponen disekitarnya. Material PTFE memiliki termal konduktivitas $0,25\text{ W/mK}$, ditambah temperatur leleh material tersebut adalah 327°C , sehingga cocok sebagai *isolator* pada kondisi kerja sekitar 250°C .

PTFE memiliki dimensi panjang 100 mm dan berdiameter luar 16 mm. Untuk dilewati material termoplastik PTFE memiliki diameter dalam 3,5 mm sepanjang 100 mm. PTFE diletakkan diatas setelah sistem pemanas.



Gambar 3.24 Dimensi isolator PTFE

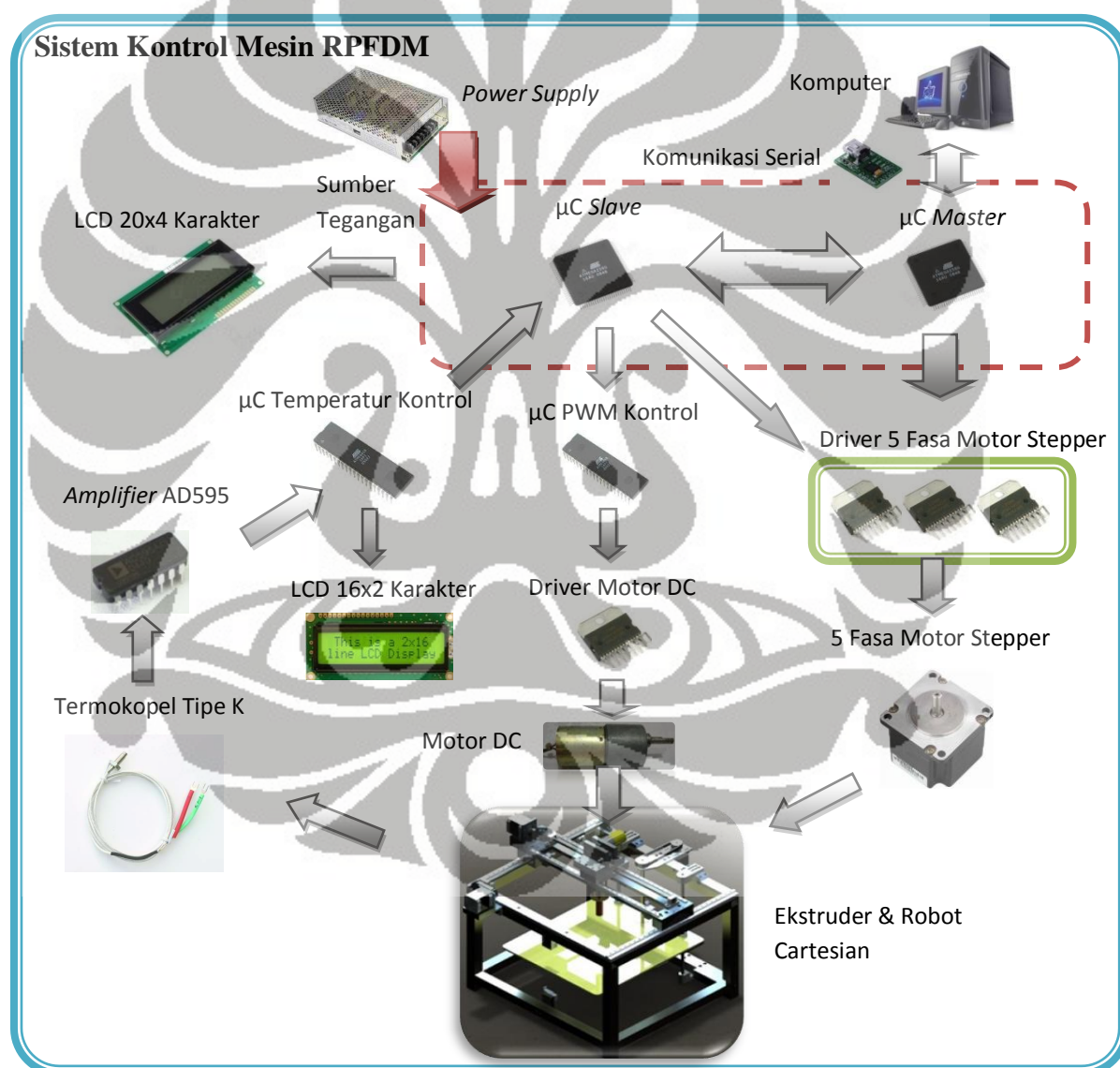


Gambar 3.25 Kawat nikrom dan komponen isolator

BAB 4

PENGEMBANGAN SISTEM KONTROL

Berdasarkan desain sistem mekanikal dan mekanisme ekstruder yang terdapat pada mesin RPFDM, dapat dilakukan pengembangan sistem kontrol berupa perangkat elektronik untuk mengendalikan mesin RPFDM. Berikut desain rancangan sistem kontrol tersebut :



Gambar 4.1 Sistem kontrol mesin RPFDM

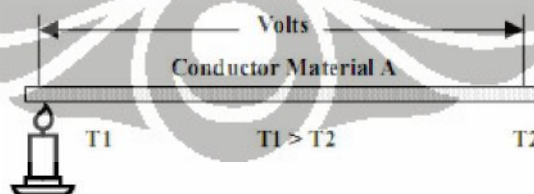
Dengan melihat sistem kontrol seperti pada gambar 4.1, maka akan dibahas komponen-komponen perangkat keras (*hardware*) sistem kontrol yang meliputi sensor dan kendali digital. Untuk sensor komponen-komponennya adalah termokopel dan *limit switch*. Sedangkan untuk kendali digital komponen-komponennya sangat banyak meliputi, mikrokontroler, driver motor stepper, driver motor DC, termokopel *ampilifier*, *relay*, modul LCD, *adaptor* USB-serial, dan *power supply*.

4.1 Sensor

Pada penelitian ini, sensor yang digunakan hanya ada dua, yaitu termokopel dan *limit switch*. Sensor ini berfungsi untuk mendeteksi keadaan mesin RPFDM saat dijalankan. Termokopel berfungsi untuk mendeteksi keadaan temperatur pada sistem pemanas (*barrel*) agar temperatur tetap konstan. Sedangkan *limit switch* digunakan untuk memposisikan posisi awal dari ekstruder pada koordinat (0,0). Data dari kedua sensor, berupa sinyal tegangan kemudian dikirim ke mikrokontroler kemudian diproses sesuai dengan kebutuhan.

4.1.1 Termokopel

Termokopel adalah sensor temperatur dengan menggunakan prinsip *efek Seebeck*, dimana termokopel dapat merubah besaran temperatur menjadi besaran tegangan dalam orde miliVolt (mV).



Gambar 4.2 Percobaan Thomas Johann Seebeck [14]

Pada tahun 1821, Thomas Johann Seebeck telah menemukan secara tidak sengaja bahwa jika suatu logam pada salah satu ujung-nya dipanaskan maka akan

muncul tegangan antara ujung yang lain. Tegangan ini sebanding dengan perbedaan temperatur antara ujung yang dipanaskan dengan ujung yang lain.

Persamaan matematis untuk gambar 4.2 adalah

$$V_{12} = S_a \times T1 - T2$$

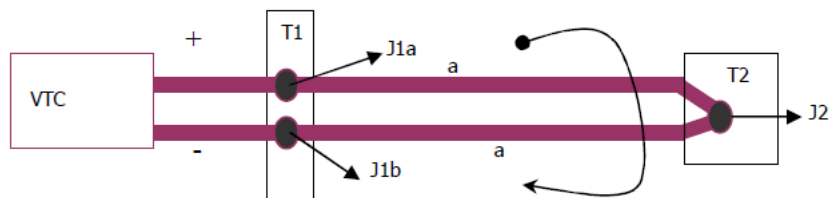
V_{12} adalah tegangan diantara dua ujung logam dalam satuan mV, S_a adalah koefisien Seebeck logam tersebut dalam satuan mV/ $^{\circ}$ C, dan $(T1-T2)$ adalah perbedaan temperatur antara T1 dan T2 dalam $^{\circ}$ C.

Timbul-nya tegangan tersebut dapat dijelaskan dengan teori fisika quantum. Secara teori jika suatu benda terdiri dari atom-atom yang didalamnya terdapat elektron, proton, dan neutron. Proton dan neutron terletak pada inti atom, sedangkan elektron mengitari inti atom. Elektron mudah berpindah dari orbit atom satu ke orbit atom yang lain jika diberikan suatu energi. Perpindahan elektron inilah yang disebut dengan arus listrik.

Jika ujung suatu logam dipanaskan maka elektron pada ujung yang dipanaskan akan memiliki energi yang cukup untuk berpindah orbit. Perpindahan orbit ini adalah dari ujung yang dipanaskan menuju ujung yang tidak dipanaskan. Perpindahan elektron inilah yang mengakibatkan ujung yang dipanaskan memiliki polaritas yang lebih positif dibandingkan ujung yang tidak dipanaskan sehingga peristiwa ini menimbulkan tegangan.

Nilai Koefisien Seebeck (S_a) tergantung pada struktur material bahan tersebut. Berarti jika bahan berbeda maka memiliki S_a yang berbeda pula. Secara teoritis S_a nilainya konstan, akan tetapi pada kenyataannya nilainya berubah tergantung pada temperatur dan struktur molekul bahan.

Termokopel selalu menggunakan dua buah logam yang berbeda yang salah satunya disatukan. Hal ini dapat dijelaskan dengan menggunakan *hukum Kirchhoff* pada gambar 4.3.



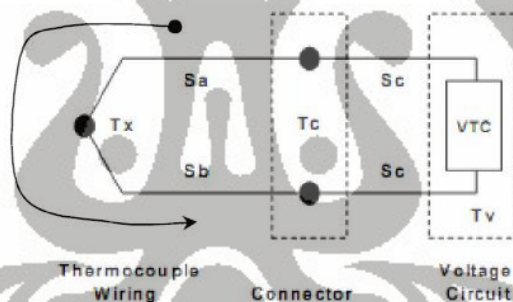
Gambar 4.3 Termokopel dengan dua buah material yang sama [14]

Berdasarkan hukum Kirchoff rangkaian pada gambar 4.3 menghasilkan persamaan dibawah ini :

$$S_a T_1 - T_2 + S_a T_2 - T_1 + VTC = 0$$

$$VTC = 0$$

Persamaan diatas menunjukkan bahwa jika kedua kabel termokopel adalah dua logam yang sama maka tegangan termokopel pada ujung terbuka (J1a dan J1b) adalah 0 Volt. Hal ini lah yang menyebabkan dua kabel termokopel yang ujungnya disatukan harus dari jenis yang berbeda.



Gambar 4.4 Rangkaian dasar dan pengukuran termokopel [14]

Maka dengan bahan material yang berbeda, termokopel dapat mengeluarkan tegangan keluaran dalam jangkauan miliVolt (mV). Pada gambar 4.4, Tx adalah ujung termokopel yang digunakan untuk mengetahui temperatur yang sedang diukur, S_a adalah koefisien Seebeck kabel termokopel a, S_b adalah koefisien Seebeck kabel termokopel b, T_c adalah temperatur pada *Junction* antara kabel termokopel dengan kabel Voltmeter, S_c adalah koefisien Seebeck kabel Voltmeter, VTC adalah tegangan yang terukur Voltmeter, T_v adalah suhu ujung kabel Voltmeter.

Dengan menggunakan hukum Kirchoff, maka pada gambar 4.4 dapat menghasilkan persamaan :

$$S_a \times T_c - T_x + S_b \times T_x - T_c + S_c \times T_c - T_v + VTC + S_c \times T_v - T_c = 0$$

$$VTC = (S_a - S_b) \times T_x - T_c$$

Jika $(S_a - S_b)$ adalah S_{ab} dan jika $(S_b - S_a)$ adalah S_{ba} . S_{ab} inilah yang biasa ditunjukkan sebagai koefisien Seebeck termokopel yang dikeluarkan oleh pabrik pembuat termokopel.

Pada penelitian ini, termokopel yang digunakan adalah termokopel tipe-K yang mampu mengukur temperatur mencapai 1200°C .



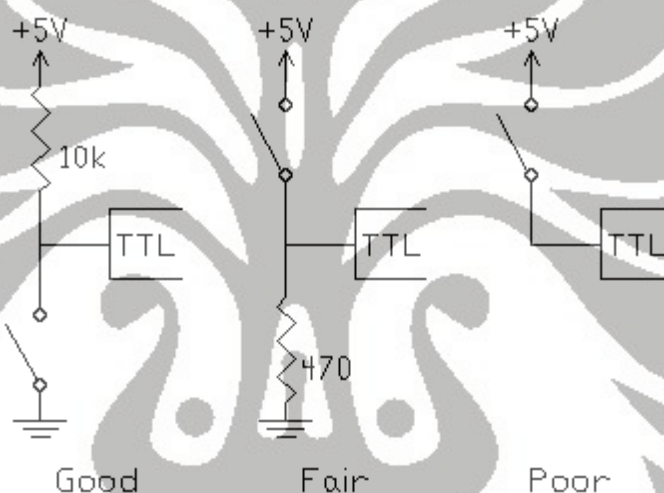
Gambar 4.5 Termokopel tipe-K [15]

4.1.2 *Limit switch*

Mesin RPFDM memiliki enam *limit switch* sebagai balikan untuk menentukan posisi *default* ekstruder. Posisi *default* adalah posisi awal ekstruder sebelum memulai proses pembuatan prototipe. Mesin RPFDM memiliki tiga sumbu, yaitu sumbu x, y, dan z. Pada setiap sumbu memiliki dua *limit switch* yang diletakkan pada setiap akhir pergerakan. *Limit switch* yang digunakan adalah merek ASCO, dengan posisi NO (*Normally Open*). *Limit switch* merupakan perangkat elektromekanikal yang berkerja dengan memanfaatkan sentuhan objek lain. Pada kondisi *normally open*, saklar yang terdapat pada *limit switch* dalam keadaan terbuka, sehingga jika tersentuh dengan objek *limit switch* akan berubah keadaan menjadi *normally closed*. Dalam keadaan *normally closed* arus listrik akan mengalir dan memberikan sinyal ke mikrokontroler sebagai data masukan.



Gambar 4.6 *Limit switch* [16]



Gambar 4.7 Rangkaian skema *limit switch* [17]

Untuk rangkaian listrik antara *limit switch* dengan mikrokontroler harus menggunakan *pull up* resistor sebesar 10 k Ω agar menghindari terjadinya arus listrik yang berlebih yang dapat membahayakan mikrokontroler. Konsekuensinya, apabila saklar terbuka maka rangkaian dalam kondisi logika 1 dengan menerima arus sebesar $5V/10k\Omega=0,5mA$ dan sebaliknya apabila saklar tertutup maka rangkaian TTL terhubung langsung dengan GND, sehingga arus listrik seluruhnya mengalir ke GND, kondisi ini disebut juga logika 0. Hal ini, dapat dikompensasi di dalam program dengan mengganti logika-nya saja. Pada rangkaian tanpa *pull-up* resistor, kondisi logika 1 adalah dimana saklar dalam kondisi tertutup, sedangkan kondisi logika 0, saklar dalam kondisi terbuka [17].

4.2 Kendali Digital

Sistem kendali yang dikembangkan merupakan perangkat elektronik yang terdiri dari mikrokontroler, penggerak motor, dan temperatur kontrol. Sistem kendali digunakan untuk mengontrol mesin RPFDM agar dapat bergerak secara otomatis.

4.2.1 Mikrokontroler

Pada sistem kontrol, kendali logika yang digunakan adalah mikrokontroler. Mikrokontroler (MCU atau μC) merupakan perangkat elektronika yang menyerupai sebuah komputer namun berukuran lebih kecil dan biasanya berbentuk IC. Layaknya sebuah komputer, mikrokontroler merupakan sebuah CPU yang memiliki perangkat lengkap seperti *crystal oscillator*, *timer*, *serial communication*, *digital I/O*, dan *analog I/O*. Mikrokontroler menggunakan memori dengan kapasitas kecil yang dapat diprogram ulang dengan komputer melalui bahasa pemrograman yang bervariasi. Pada penelitian ini, bahasa pemrograman yang digunakan adalah bahasa C [22].

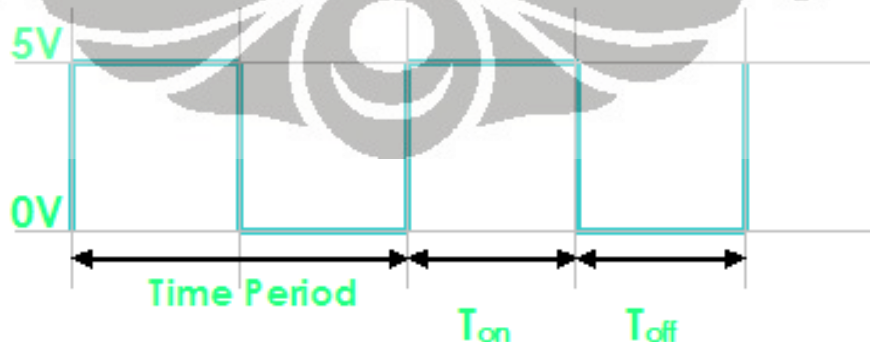
Kecepatan proses data bergantung pada penggunaan *oscillator* pada mikrokontroler. *Oscillator* merupakan sebuah perangkat yang menghasilkan frekuensi tertentu untuk menjadi *clock source* bagi mikrokontroler. *Clock source* pada mikrokontroler menjadi patokan waktu bagi mikrokontroler untuk melakukan sebuah instruksi. Mikrokontroler ATMEL pada umumnya mempunyai karakter satu instruksi dan memiliki *oscillator* internal yang terdapat didalam mikrokontroler. *Oscillator* internal yang terdapat pada mikrokontroler dirasa kurang cukup memadai untuk aplikasi kontrol mesin RPFDM karena frekuensi hanya 1MHz. Dengan frekuensi 1MHz mikrokontroler hanya mampu mencapai kecepatan 1MIPS (*Million Instruction per Second*). Pada sistem kontrol, semua mikrokontroler yang digunakan menggunakan *clock source* 16MHz, dikarenakan untuk mencapai hasil yang maksimal untuk mengendalikan motor stepper. Akan tetapi, hal ini dapat mempengaruhi error pada penggunaan USART (*Universal Synchronous Asynchronous Receiver Transmitter*) sebesar 0,2%, karena semua mikrokontroler saling berkomunikasi satu sama lain. *Oscillator* yang digunakan berupa kristal berukuran kecil bernilai 16 MHz.



Gambar 4.8 Kristal 16 MHz pada ATmega 2560 dan ATmega 16 [18]

Mikrokontroler merupakan sistem *embedded* dimana *processor core*, memori, *input/output*, dan perangkat lainnya seperti ADC sudah terdapat didalam sebuah IC. Fitur-fitur yang terdapat didalam mikrokontroler keluarga AVR seperti *ADC (Analog Digital Converter)*, *USART (Universal Synchronous Asynchronous Receiver Transmitter)*, *PWM (Pulse Width Modulation)*, *SPI (Serial Peripheral Interface)*, *Timers*, dan *Interrupt* sangat memudahkan penggunaannya dalam membuat sebuah sistem kontrol yang ringkas dan sederhana. Fitur-fitur seperti ADC dan PWM sangat dibutuhkan didalam penelitian ini.

PWM (Pulse Width Modulation) adalah sebuah teknik yang digunakan untuk membuat signal analog dari *digital device* seperti mikrokontroler. Kebanyakan mikrokontroler (MCU atau μC) modern saat ini sudah mengaplikasikan fitur PWM. *Digital device* seperti MCU, hanya menghasilkan dua level tegangan keluaran, HIGH = 5V atau LOW = 0V. Lalu, bagaimana jika diinginkan tegangan keluaran sebesar 3,75V ?, oleh karena itu, teknik PWM ini diterapkan.

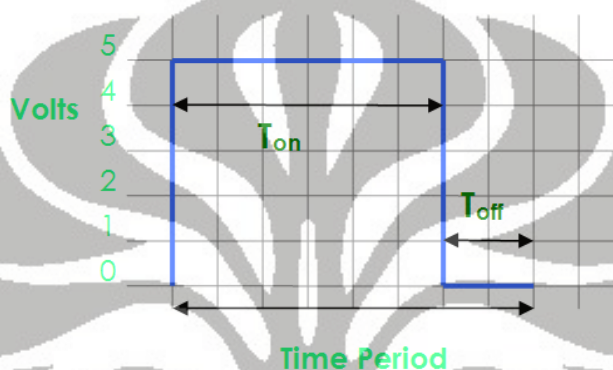


Gambar 4.9 Gelombang sinyal PWM [19]

Didalam gambar 4.9, selama selang periode waktu (T) terdapat dua level logika yang aktif, yaitu HIGH dan LOW, atau bisa HIGH saja maupun LOW saja. Secara sederhana adalah seberapa persen sinyal HIGH muncul dalam satu siklus periode waktu, yang sering disebut juga sebagai *duty cycle*, dengan rumus dasar matematika :

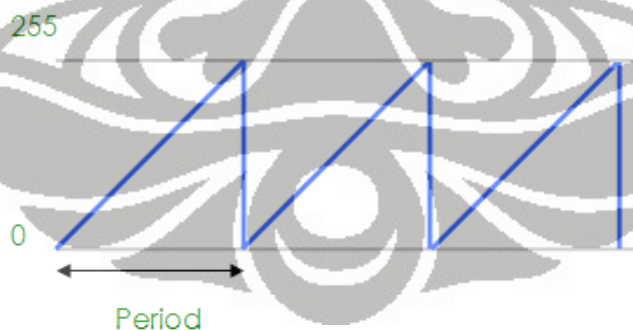
$$\% DC = \frac{T_{on}}{T_{on} + T_{off}} \times 100.$$

Sehingga dengan *duty cycle* 75%, maka akan menghasilkan tegangan keluaran sebesar $75\% \times 5V = 3,75V$ [19].



Gambar 4.10 *Duty cycle* 75% [19]

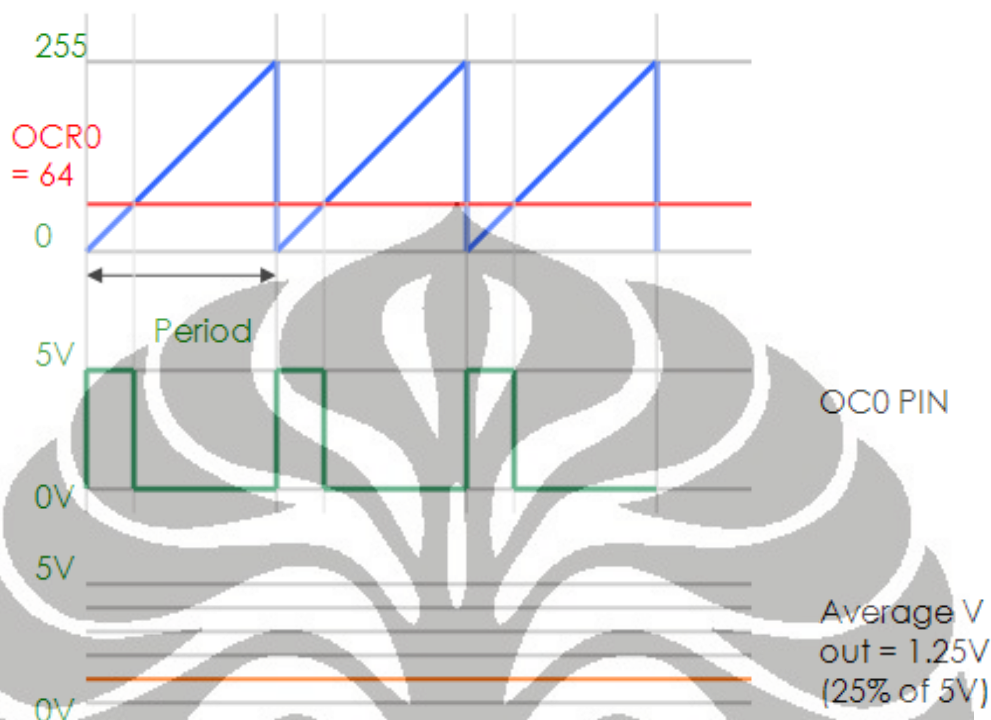
Pada MCU AVR sinyal PWM dibangkitkan melalui unit timer. Metode yang digunakan pada penelitian ini adalah dengan menggunakan *FastPWM*, yang memiliki bentuk gelombang seperti gambar 4.11.



Gambar 4.11 Bentuk gelombang *FastPWM* [20]

Di dalam dunia PWM, akan dikenal yang disebut sebagai *prescaler* (pembagi bilangan bit dari 0-255), karena pada penelitian ini menggunakan timer0 maka akan

dikenal yang namanya OCR0 sebagai *prescaler* sehingga nilai yang dimasukkan adalah diantara 0 sampai 255 ($2^8 = 256$).



Gambar 4.12 Gelombang PWM dengan OCR0 = 64 [20]

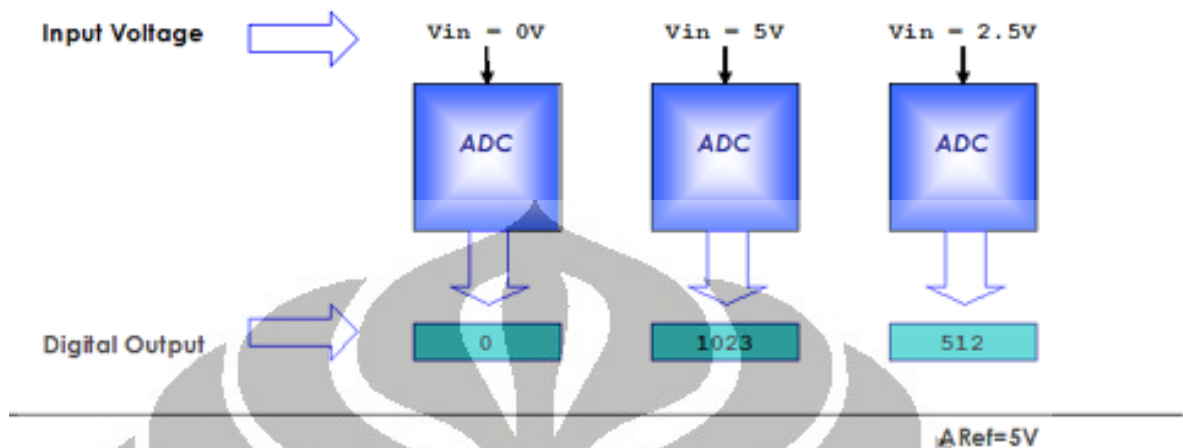
Dari datasheet MCU AVR, untuk perhitungan frekuensi dari OCR0 dengan menggunakan metode *FastPWM* dirumuskan dengan ; $f_{OCnPWM} = \frac{f_{clk_I/O}}{N \cdot 256}$. Sebagai contoh, dengan menggunakan $f_{clk_I/O} = 250$ kHz, dan *prescaler*, $N = 64$ ($64/256 = 25\%$), maka:

$$f_{OCnPWM} = \frac{f_{clk_I/O}}{N \times 256} = \frac{250 \times 10^3 \text{ Hz}}{64 \times 256} = 15,259 \text{ Hz}$$

$$T_{OCnPWM} = \frac{1}{f_{OCnPWM}} = \frac{1}{15,259} = 65,535 \text{ ms}$$

Selain fitur PWM (*Pulse Width Modulation*), fitur ADC (*Analog Digital Converter*) juga digunakan untuk mengkonversi tegangan. ADC (*Analog Digital Converter*) adalah perangkat elektronik yang digunakan untuk mengkonversi sinyal masukan dalam bentuk analog menjadi sinyal keluaran digital serta perangkat ADC ini

sudah terdapat di dalam IC MCU. Tegangan yang masuk sebanding dengan nilai digital yang keluar pada MCU.



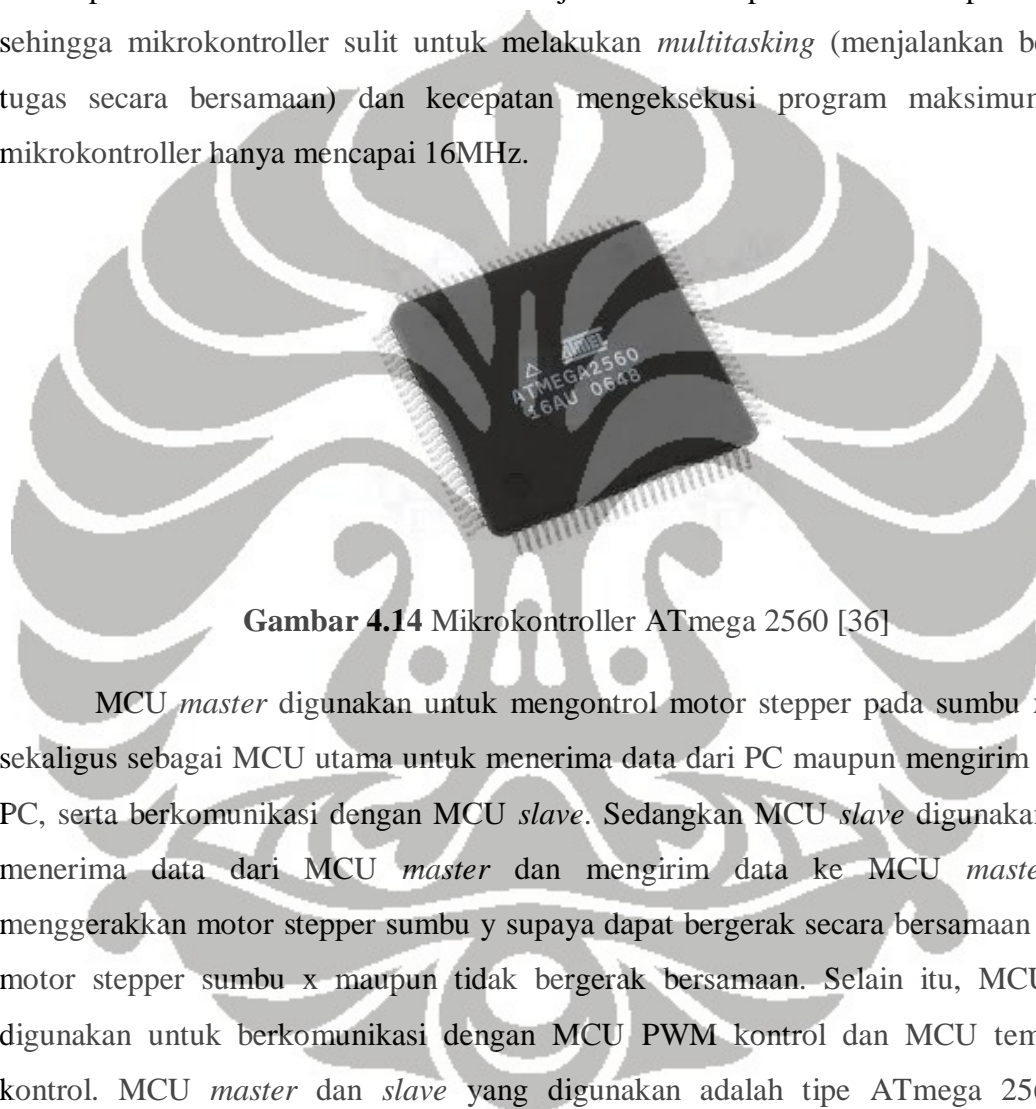
Gambar 4.13 Aplikasi ADC [21]

Pada fitur ADC juga memerlukan frekuensi untuk melakukan proses konversi sinyal. Frekuensi tersebut dihasilkan dari hasil pembagian frekuensi *clock source* mikrokontroler, sehingga menghasilkan frekuensi yang lebih kecil dari *clock source* utama. Pada umumnya ADC membutuhkan frekuensi yang rendah diantara 50kHz dan 200kHz. Pada frekuensi yang tinggi, proses konversi menjadi lebih cepat akan tetapi pada frekuensi yang rendah proses konversi yang dihasilkan menjadi lebih akurat.

Frekuensi sistem (*clock source*) dapat ditentukan seberapa besar nilai-nya dengan menggunakan internal atau eksternal *oscillator*. Sama seperti halnya pada PWM, pada ADC juga membutuhkan *prescaler* (faktor pembagi). *Prescaler* pada ADC adalah faktor pembagi untuk menentukan frekuensi ADC dari frekuensi sistem. Frekuensi sistem dapat dibagi dengan angka 2, 4, 16, 32, 64, dan 128 tergantung pengaturan dari *prescaler*. ADC mengkonversi tegangan masukan menjadi angka. ADC memiliki resolusi, ADC 10 bit berarti memiliki jangkauan angka dari 0 sampai 1023 ($2^{10} = 1024$), dan ADC juga memiliki tegangan referensi (V_{ref}). Ketika tegangan masukan sama dengan 0V (GND) angka keluaran adalah 0, sedangkan jika tegangan masukan sama dengan V_{ref} maka angka keluaran adalah 1023. Maka, jika jangkauan tegangan referensi (0- V_{ref}) maka jangkauan keluaran *digital* adalah (0-1023) dengan ADC 10 bit [21].

4.2.1.1 Mikrokontroller *Master* dan *Slave*

Pengembangan sistem kontrol mesin RPFDM, mikrokontroller digunakan sebagai kontrol logika. Karena kerumitan dan kesulitan untuk mengontrol pergerakan motor stepper secara bersamaan dalam satu instruksi oleh mikrokontroller, maka digunakanlah dua buah mikrokontroller untuk mengontrol motor stepper. Karena fitur dasar pada mikrokontroller adalah menjalankan setiap intruksi setiap baris-nya, sehingga mikrokontroller sulit untuk melakukan *multitasking* (menjalankan beberapa tugas secara bersamaan) dan kecepatan mengeksekusi program maksimum pada mikrokontroller hanya mencapai 16MHz.



Gambar 4.14 Mikrokontroller ATmega 2560 [36]

MCU *master* digunakan untuk mengontrol motor stepper pada sumbu x dan z sekaligus sebagai MCU utama untuk menerima data dari PC maupun mengirim data ke PC, serta berkomunikasi dengan MCU *slave*. Sedangkan MCU *slave* digunakan untuk menerima data dari MCU *master* dan mengirim data ke MCU *master* agar menggerakkan motor stepper sumbu y supaya dapat bergerak secara bersamaan dengan motor stepper sumbu x maupun tidak bergerak bersamaan. Selain itu, MCU *slave* digunakan untuk berkomunikasi dengan MCU PWM kontrol dan MCU temperatur kontrol. MCU *master* dan *slave* yang digunakan adalah tipe ATmega 2560 dari ATMEL. Pemilihan tipe berdasarkan spesifikasi yang dimiliki mikrokontroller tersebut dan kebutuhan dari sistem kontrol mesin RPFDM, yaitu:

- ✚ Memiliki I/O yang banyak, minimal 52.
- ✚ Mampu melakukan komunikasi serial, minimal 4 USART.
- ✚ Mampu mengeksekusi program maupun data dengan cepat.

Features

- High Performance, Low Power AVR[®] 8-Bit Microcontroller
- Advanced RISC Architecture
 - 135 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-Chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 64K/128K/256K Bytes of In-System Self-Programmable Flash
 - 4K Bytes EEPROM
 - 8K Bytes Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/ 100 years at 25°C
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
 - Endurance: Up to 64K Bytes Optional External Memory Space
- JTAG (IEEE std. 1149.1 compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - Four 16-bit Timer/Counter with Separate Prescaler, Compare- and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four 8-bit PWM Channels
 - Six/Twelve PWM Channels with Programmable Resolution from 2 to 16 Bits (ATmega1281/2561, ATmega640/1280/2560)
 - Output Compare Modulator
 - 8/16-channel, 10-bit ADC (ATmega1281/2561, ATmega640/1280/2560)
 - Two/Four Programmable Serial USART (ATmega1281/2561, ATmega640/1280/2560)
 - Master/Slave SPI Serial Interface
 - Byte Oriented 2-wire Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - 54/86 Programmable I/O Lines (ATmega1281/2561, ATmega640/1280/2560)
 - 64-pad QFN/MLF, 64-lead TQFP (ATmega1281/2561)
 - 100-lead TQFP, 100-ball CBGA (ATmega640/1280/2560)
 - RoHS/Fully Green
- Temperature Range:
 - -40°C to 85°C Industrial
- Ultra-Low Power Consumption
 - Active Mode: 1 MHz, 1.8V: 500 μ A
 - Power-down Mode: 0.1 μ A at 1.8V
- Speed Grade:
 - ATmega640V/ATmega1280V/ATmega1281V:
 - 0 - 4 MHz @ 1.8 - 5.5V, 0 - 8 MHz @ 2.7 - 5.5V
 - ATmega2560V/ATmega2561V:
 - 0 - 2 MHz @ 1.8 - 5.5V, 0 - 8 MHz @ 2.7 - 5.5V
 - ATmega640/ATmega1280/ATmega1281:
 - 0 - 8 MHz @ 2.7 - 5.5V, 0 - 16 MHz @ 4.5 - 5.5V
 - ATmega2560/ATmega2561:
 - 0 - 16 MHz @ 4.5 - 5.5V



8-bit **AVR[®]**
Microcontroller
with
64K/128K/256K
Bytes In-System
Programmable
Flash

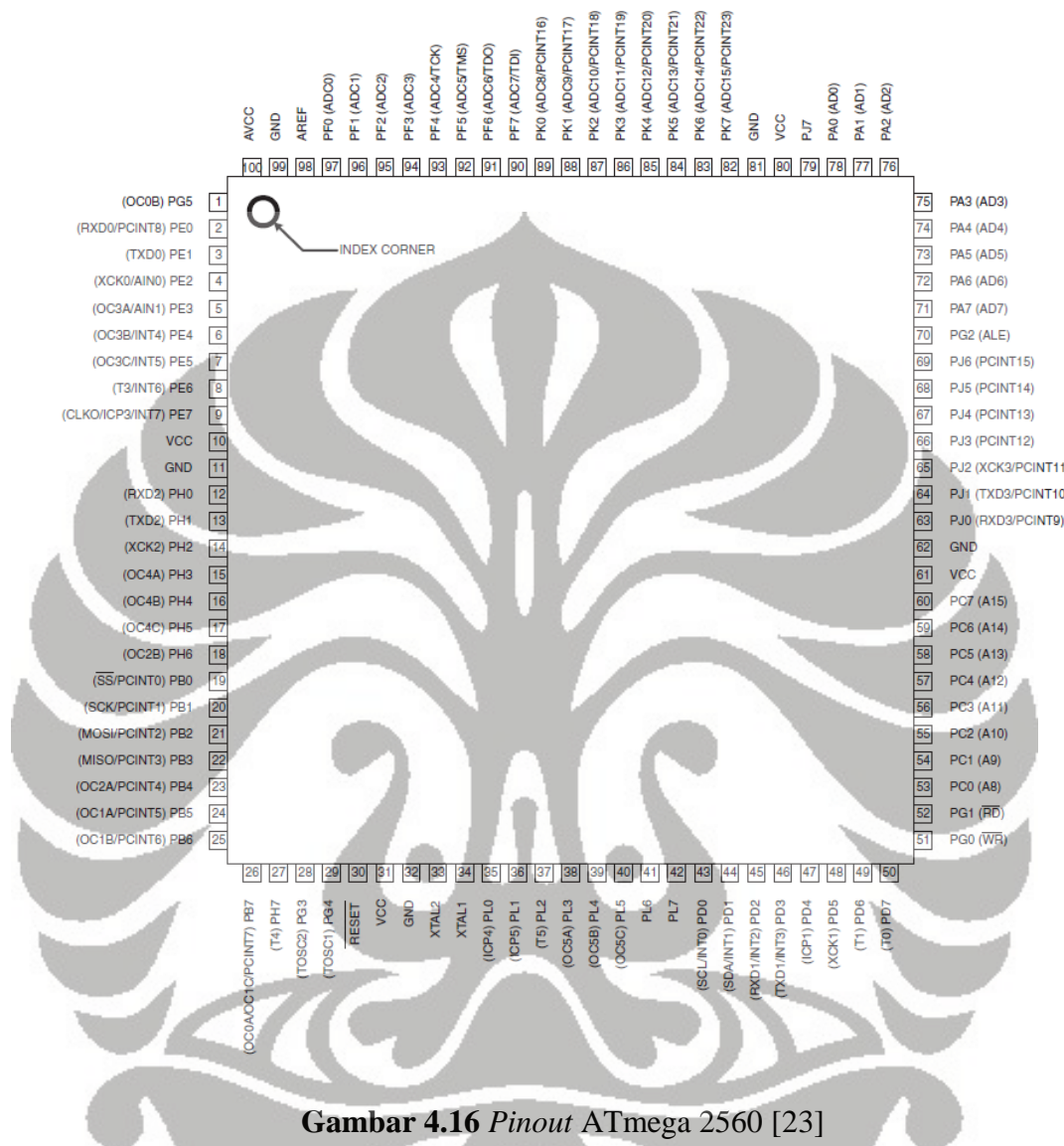
ATmega640/V
ATmega1280/V
ATmega1281/V
ATmega2560/V
ATmega2561/V

Preliminary



Gambar 4.15 Spesifikasi mikrokontroler ATmega 2560 [23]

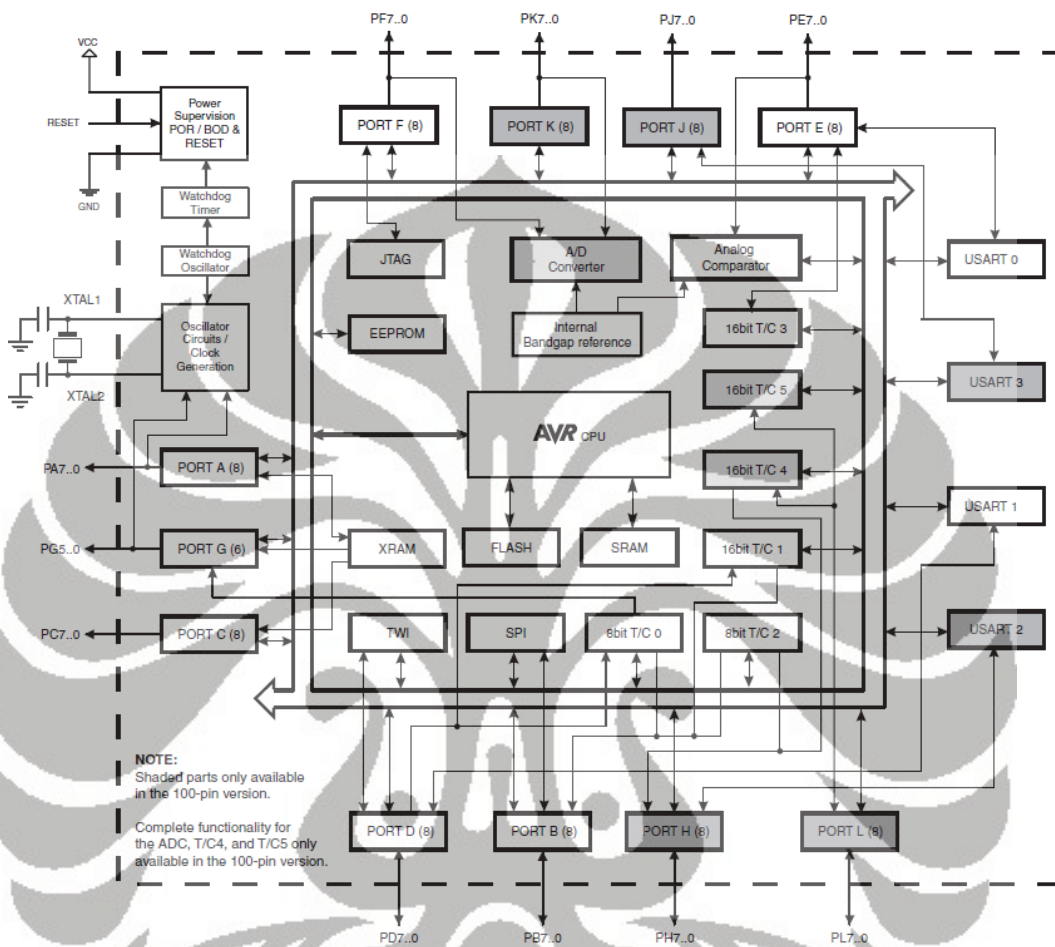
Mikrokontroler ATmega 2560 (*master* dan *slave*) memiliki diagram *pinout* seperti gambar dibawah ini :



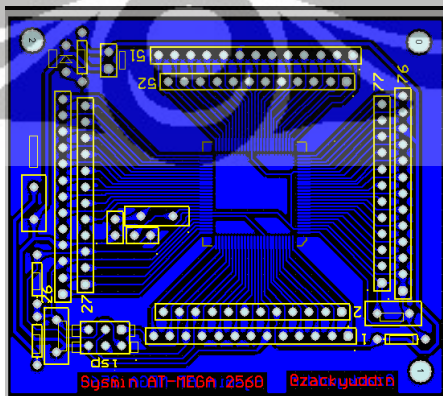
Gambar 4.16 Pinout ATmega 2560 [23]

Seperti terlihat pada gambar 4.15 dan 4.16 fitur-fitur yang dimiliki ATmega 2560 sudah sangat memadai untuk mengontrol mesin RPFDM. Setiap mikrokontroler memiliki sistem arsitektur yang unik, ATmega 2560 merupakan mikrokontroler 8-Bit RISC (*Reduced Instruction Set Computing*), sehingga aplikasinya cukup sederhana karena ada fungsi-fungsi pemrograman yang sudah disederhanakan. RISC merupakan desain dari CPU dimana instruksi-instruksi yang telah disederhanakan untuk meningkatkan performa dan kinerja dikarenakan kecepatan eksekusi yang lebih cepat dari desain CPU CISC (*Complex Instruction Set Computing*).

Berdasarkan datasheet ATmega 2560, mikrokontroler ini memiliki memori 256kb *flash programming*, 4kb EEPROM, 8kb RAM, 86 pin I/O, 12 16bits resolution PWM channels, 4 serial USARTs, dan 16 ADC channels.



Gambar 4.17 Diagram blok dari ATmega 2560 [23]



Gambar 4.18 Desain sistem minimum ATmega 2560

ATmega 2560 diproduksi dalam bentuk IC, sehingga untuk sistem antar muka dengan perangkat elektronika lain diperlukan *expansion board*/sistem minimum (sismin). Untuk itu dirancang *expansion board* yang sederhana menggunakan kristal 16MHz. Perancangan *expansion board* ini berdasarkan datasheet ATmega 2560 dan didesain menggunakan perangkat lunak khusus mendesain PCB yaitu Protel.

Seperti terlihat pada gambar 4.18, setiap pin diekspansi, hanya pin-pin pendukung saja yang ditambahkan perangkat keras, seperti *oscillator* dan *port programming* 6 pin standar Atmel karena untuk memprogram ATmega 2560 digunakan *programmer AVR ISP MKII*. Selain menggunakan *programmer AVR ISP MKII*, mikrokontroler ATmega 2560 juga dapat diprogram dengan menggunakan *programmer AVR ISP 910*. *Programmer AVR ISP 910* merupakan rancangan desain sendiri dengan menggunakan ATmega 8 dengan kecepatan 12MHz. Kelebihan penggunaan *programmer AVR ISP MKII* adalah proses *flash programming* yang lebih cepat dari pada *programmer AVR ISP 910*. Sebenarnya kecepatan proses *flash programming* juga tergantung dari besarnya data program yang di-*compile*.

Sesuai dengan fitur yang terdapat pada ATmega 2560 maka dibuatlah *pin assignment* yang sesuai dengan kebutuhan dan program yang terdapat didalamnya. *Pin assignment* ini dapat dilihat pada tabel 4.1 :

Tabel 4.1 *Pin Assignment* μ C Master dan μ C Slave

μ C Master			μ C Slave		
Fungsi	I/O	μ C I/O	Fungsi	I/O	μ C I/O
Logic 1 Stepper Axis X	output	PORT A.1	Logic 1 Stepper Axis Y	output	PORT C.1
Logic 2 Stepper Axis X	output	PORT A.2	Logic 2 Stepper Axis Y	output	PORT C.2
Logic 3 Stepper Axis X	output	PORT A.3	Logic 3 Stepper Axis Y	output	PORT C.3
Logic 4 Stepper Axis X	output	PORT A.4	Logic 4 Stepper Axis Y	output	PORT C.4
Logic 5 Stepper Axis X	output	PORT A.5	Logic 5 Stepper Axis Y	output	PORT C.5
None	input	PORT J.0	Enable 1 Stepper Axis Y	output	PORT D.1
None	input	PORT J.1	Enable 2 Stepper Axis Y	output	PORT D.2
Enable 1 Stepper Axis X	output	PORT J.2	Enable 3 Stepper Axis Y	output	PORT D.3
Enable 2 Stepper Axis X	output	PORT J.3	Enable 4 Stepper Axis Y	output	PORT D.4

Enable 3 Stepper Axis X	output	PORT J.4	Enable 5 Stepper Axis Y	output	PORT D.5
Enable 4 Stepper Axis X	output	PORT J.5	<i>Receiver USART 2</i>	input	PORT H.0
Enable 5 Stepper Axis X	output	PORT J.6	<i>Transmitter USART 2</i>	input	PORT H.1
LCD Kontrol	output	PORT C.0	LCD Kontrol	output	PORT A.0
LCD Kontrol	output	PORT C.1	LCD Kontrol	output	PORT A.1
LCD Kontrol	output	PORT C.2	LCD Kontrol	output	PORT A.2
LCD Kontrol	output	PORT C.4	LCD Kontrol	output	PORT A.4
LCD Kontrol	output	PORT C.5	LCD Kontrol	output	PORT A.5
LCD Kontrol	output	PORT C.6	LCD Kontrol	output	PORT A.6
LCD Kontrol	output	PORT C.7	LCD Kontrol	output	PORT A.7
Limit Switch Axis X+	input	PORT D.0	<i>Receiver USART 0</i>	input	PORT E.0
Limit Switch Axis X-	input	PORT D.1	<i>Transmitter USART 0</i>	input	PORT E.1
<i>None</i>	input	PORT D.2	<i>Receiver USART 3</i>	input	PORT J.0
<i>None</i>	input	PORT D.3	<i>Transmitter USART 3</i>	input	PORT J.1
Limit Switch Axis Z+	input	PORT D.4			
Limit Switch Axis Z-	input	PORT D.5			
<i>Receiver USART 0</i>	input	PORT E.0			
<i>Transmitter USART 0</i>	input	PORT E.1			
Logic 1 Stepper Axis Z	output	PORT E.2			
Logic 2 Stepper Axis Z	output	PORT E.3			
Logic 3 Stepper Axis Z	output	PORT E.4			
Logic 4 Stepper Axis Z	output	PORT E.5			
Logic 5 Stepper Axis Z	output	PORT E.6			
Enable 1 Stepper Axis Z	output	PORT F.1			
Enable 2 Stepper Axis Z	output	PORT F.2			
Enable 3 Stepper Axis Z	output	PORT F.3			
Enable 4 Stepper Axis Z	output	PORT F.4			
Enable 5 Stepper Axis Z	output	PORT F.5			
<i>Receiver USART 2</i>	input	PORT H.0			
<i>Transmitter USART 2</i>	input	PORT H.1			

- cct : USART 0 μ C Master \rightarrow Serial komunikasi antara μ C Master dengan PC.
 USART 0 μ C Slave \rightarrow Serial komunikasi antara μ C Slave dengan μ C Temperatur Kontrol.
 USART 2 μ C Master & Slave \rightarrow Serial komunikasi antara μ C Master dengan μ C Slave
 USART 3 μ C Slave \rightarrow Serial komunikasi antara μ C Slave dengan μ C PWM Kontrol.

4.2.1.2 Mikrokontroler PWM dan Temperatur

Di dalam arsitektur sistem kontrol, mikrokontroler PWM (*Pulse Width Modulation*) serta temperatur kontrol masing-masing menggunakan mikrokontroler ATmega 16 dengan menggunakan kecepatan eksternal dari kristal 16 MHz. ATmega 16 dipilih dikarenakan fiturnya sudah cukup untuk memenuhi kebutuhan mengontrol kecepatan motor DC dan kebutuhan ADC (*Analog Digital Converter*) 10 bit.

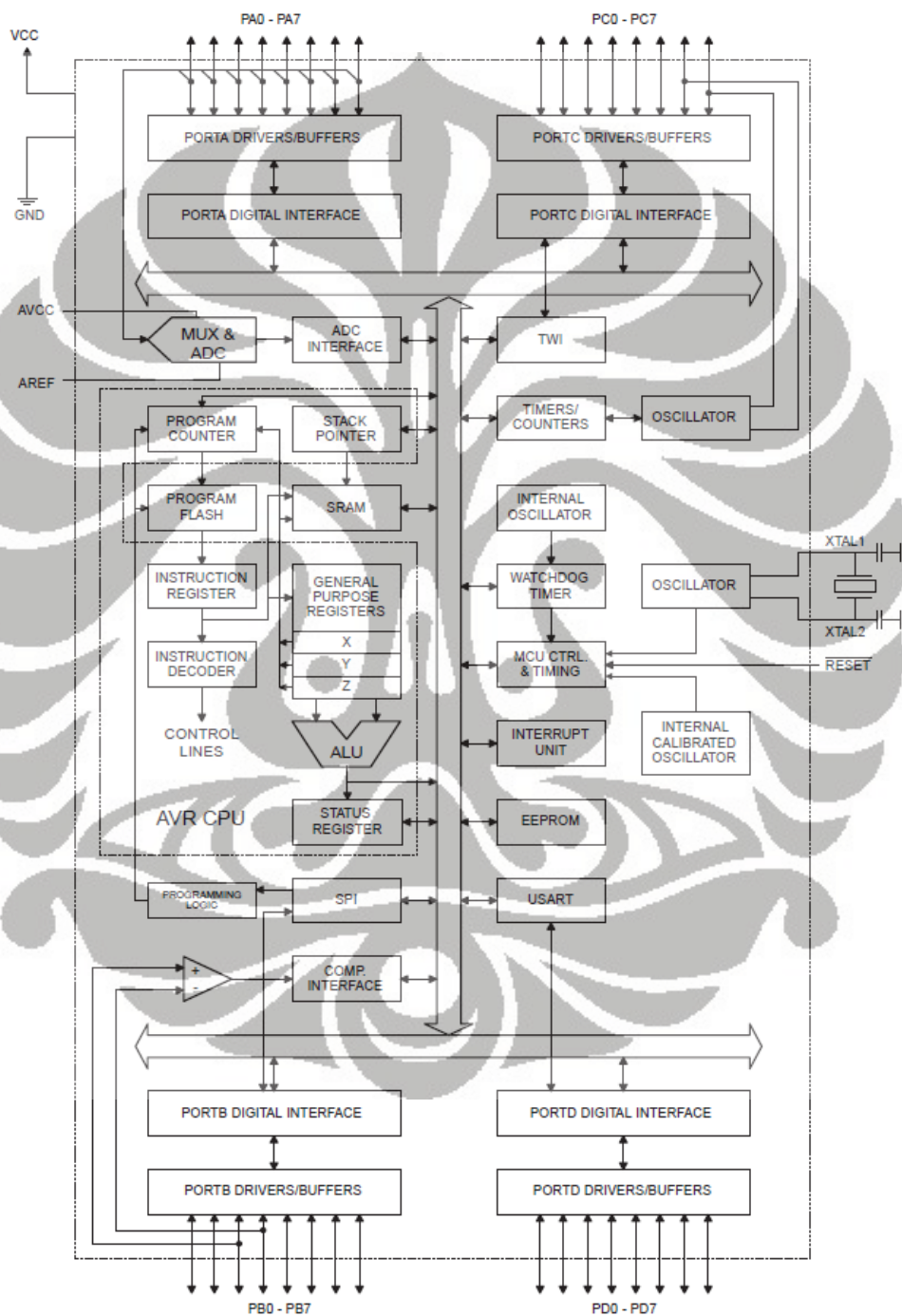
ATmega 16 juga merupakan microcontroller 8 bit RISC (*Reduced Instruction Set Computing*) seperti ATmega 2560. Perbedaan ATmega 16 dengan ATmega 2560 adalah banyaknya pin-pin I/O, dan jumlah fitur yang lain, seperti banyak jumlah PWM, dan ADC.



Gambar 4.19 Mikrokontroler ATmega 16 [37]

Berdasarkan datasheet ATmega 16, mikrokontroler ini memiliki memori 16 kb *flash programming*, 512 bit EEPROM, 1kb *internal SRAM*, 4 PWM *channels*, 8 *channels* 10 bit ADC, dan 1 USART (*Universal Synchronous Asynchronous Receiver*

Transmitter). Dengan fitur-fitur tersebut mikrokontroler tersebut dapat memenuhi kebutuhan untuk mengontrol kecepatan motor DC dengan PWM, dan mengontrol temperatur agar tetap konstan dengan menggunakan bantuan *relay*. Sedangkan adanya fitur USART, digunakan untuk berkomunikasi dengan mikrokontroler *slave* yang tidak lain adalah ATmega 2560. Untuk fitur-fitur yang ada bisa dilihat datasheet gambar 4.21.



Gambar 4.20 Diagram blok dari ATmega 16 [24]

Features

- High-performance, Low-power AVR[®] 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
 - 16K Bytes of In-System Self-programmable Flash program memory
 - 512 Bytes EEPROM
 - 1K Byte Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four PWM Channels
 - 8-channel, 10-bit ADC
 - 8 Single-ended Channels
 - 7 Differential Channels in TQFP Package Only
 - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Operating Voltages
 - 2.7 - 5.5V for ATmega16L
 - 4.5 - 5.5V for ATmega16
- Speed Grades
 - 0 - 8 MHz for ATmega16L
 - 0 - 16 MHz for ATmega16
- Power Consumption @ 1 MHz, 3V, and 25°C for ATmega16L
 - Active: 1.1 mA
 - Idle Mode: 0.35 mA
 - Power-down Mode: < 1 µA



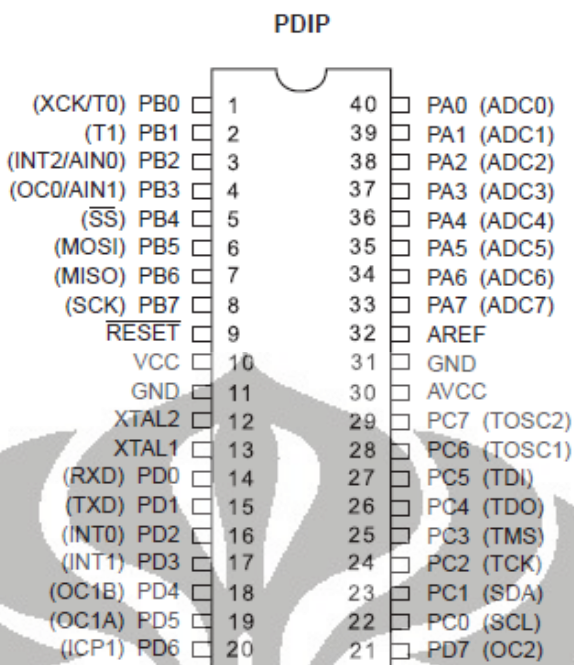
8-bit **AVR[®]**
Microcontroller
 with 16K Bytes
In-System
Programmable
Flash

ATmega16
ATmega16L

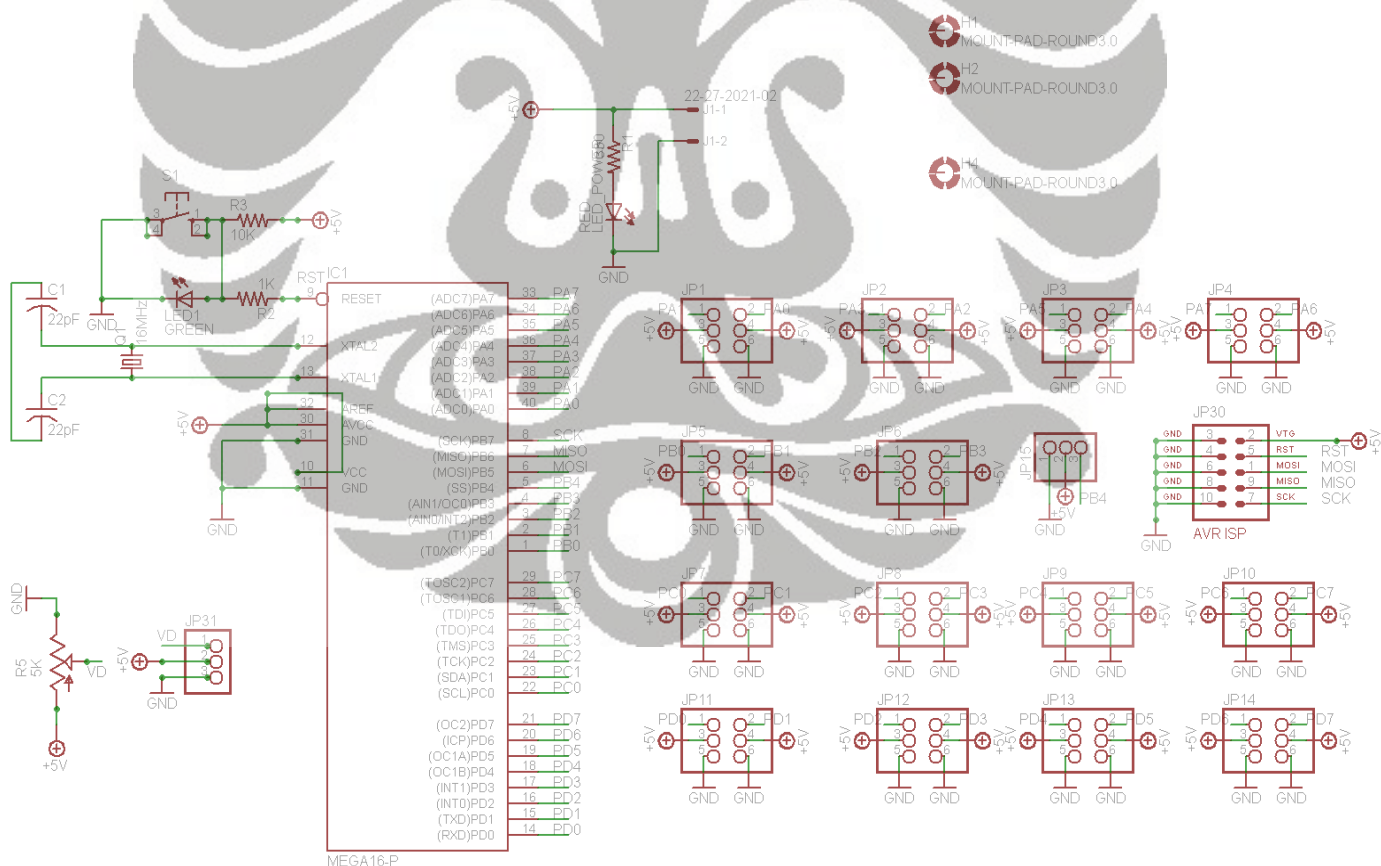
Note: Not recommended for new designs.



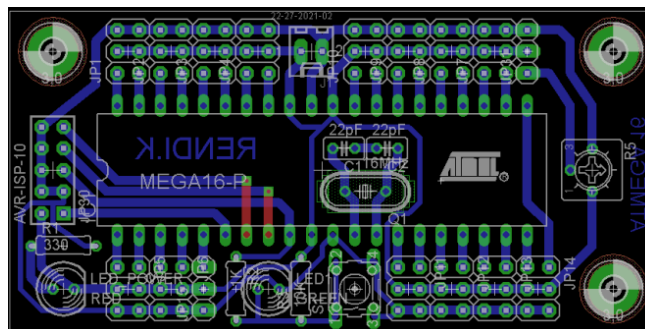
Gambar 4.21 Spesifikasi mikrokontroler ATmega 16 [24]



Gambar 4.22 Pinout dari ATmega 16 [24]



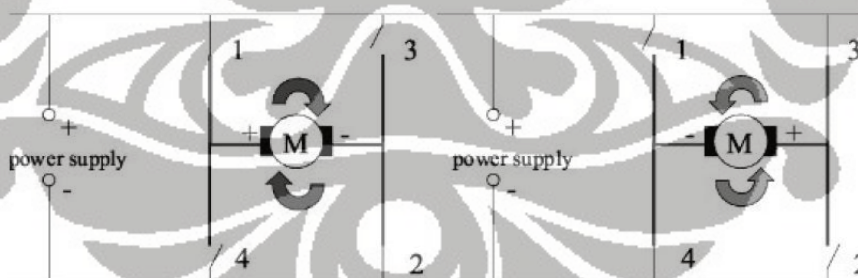
Gambar 4.23 Skema dari desain sistem minimum ATmega 16



Gambar 4.24 Desain PCB ATmega 16

4.2.2 Driver motor stepper

Pada setiap sumbu mesin RPFDM digerakkan oleh tipe motor stepper yang tipikal, yaitu lima fasa pentagon motor stepper. Spesifikasi motor sudah dikemukakan pada bab sebelumnya. Untuk menggerakkan motor *stepper* ini dibutuhkan sinyal logika dan perangkat elektronik tersendiri dengan menggunakan prinsip H-bridge, dikarenakan motor stepper ini adalah tipe *bipolar* dimana koil pada motor harus diubah polaritasnya. Fungsi dari driver motor bukan hanya untuk menggerakkan motor, akan tetapi juga dapat sebagai pengaman untuk mikrokontroler, diakibatkan adanya arus balik yang ditimbulkan oleh gaya induksi magnetik dari motor.



Gambar 4.25 Pergantian polaritas pada H-Bridge [25]

Driver motor stepper ini, menggunakan tiga buah L298N sebagai H-bridge, karena pada satu L298N terdapat 2 buah H-bridge, sedangkan untuk mengontrol motor ini dibutuhkan 5 buah H-bridge. H-bridge atau diterjemahkan secara kasar sebagai “Jembatan H”, adalah sebuah rangkaian dimana motor menjadi titik tengah dengan dua jalur yang bisa dibuka tutup untuk melewati arus pada motor tersebut, persis seperti

huruf 'H' (dengan motor berada pada garis horizontal). Dengan adanya H-bridge ini, memungkinkan koil pada motor stepper dapat berubah polaritasnya sehingga motor stepper dapat berputar.

Pada driver motor telah dibatasi arus listrik yang mengalir ke motor stepper, dikarenakan arus yang dibutuhkan motor stepper untuk berputar adalah arus listrik yang menimbulkan induksi magnetik pada koil, sehingga arus harus dibatasi maksimum 0,75 A/phase dengan hambatan *sense* yang terdapat pada L298N. Pada datasheet L298N diketahui tegangan *sense* pada L298N adalah 2V, maka arus dilimitasi mencapai 0,75 A, sehingga nilai hambatannya adalah :

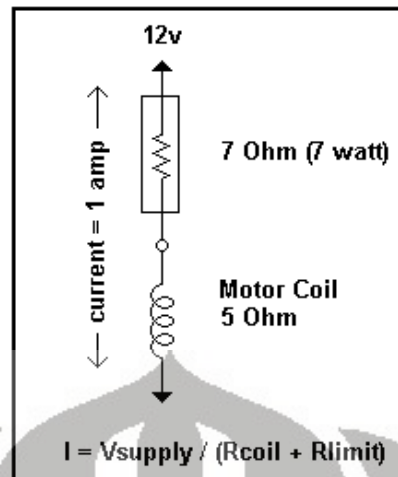
$$R_{sense} = \frac{V_{sense}}{I_{max}} = \frac{2}{0,75} = 2,667\Omega$$

$$Watt = V_{sense} \times I_{max} = 2 \times 0,75 = 1,5Watt$$

Maka, diperoleh nilai hambatan 2,667ohm dengan 1,5 Watt. Akan tetapi sangat sulit diperoleh nilai hambatan senilai 2,667 ohm dengan 1,5 Watt. Oleh karena itu, hambatan dibuat susunan seri dengan nilai masing-masing 1,5 ohm berdaya 0.5 Watt dan 1,2 ohm berdaya 1 Watt. Perhitungan ini, sangat cocok untuk metode pembatasan arus dengan menggunakan komparator dan tegangan referensi 2V.

Pembatasan arus pada tegangan 12V dimaksudkan agar untuk mencapai arus maksimal (0,75 A) lebih cepat dicapai dari pada menggunakan tegangan 5V. Ibaratkan kita menggunakan keran air berdiameter yang besar dibandingkan dengan keran air yang kecil. Sehingga wadah/ember lebih cepat penuh dengan keran air yang berdiameter besar.

Cara termudah untuk mendapatkan jumlah arus yang banyak adalah memperbesar beda tegangan karena besarnya arus berbanding lurus dengan tegangan sesuai dengan hukum Ohm. Karena arus yang besar inilah harus dilakukan pembatasan jumlah arus yang mengalir melewati koil pada motor. Cara termudah untuk membatasi arus yang mengalir ke koil adalah dengan cara merangkaikan hambatan secara seri dengan koil motor stepper, seperti pada gambar 4.26.



Gambar 4.26 Rangkaian pembatas arus dengan *power resistor* [26]

Pada gambar 4.26, rangkaian seri R (resistor) dengan L (induktor) disebut juga rangkaian *resistive*. Untuk perhitungan arus yang dibatasi sebesar, $I = \frac{V_s}{R_{coil} + R_{limit}}$. Jika menggunakan tegangan sumber 24V, dengan hambatan koil motor stepper dilihat dari *datasheet* adalah 1,7 ohm, maka nilai R_{limit} dengan arus yang diinginkan adalah 0,75A.

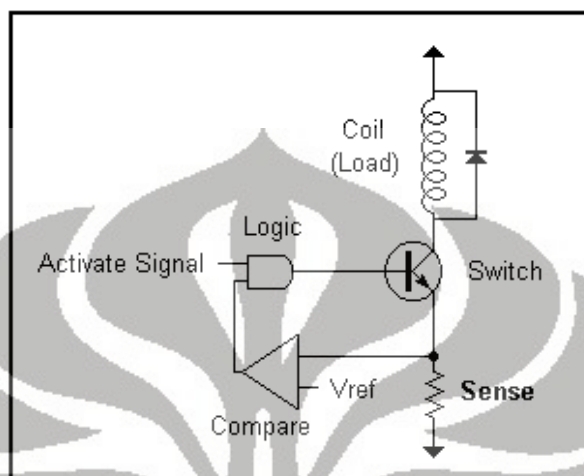
$$R_{limit} = \frac{24V}{0,75A} - 1,7 = 30,3\Omega$$

Hal lain yang harus diperhatikan, adalah panas yang dihasilkan akibat pembatasan arus pada hambatan, dikarenakan adanya penumpukan jumlah arus secara kontinu sehingga menimbulkan *heat dissipations*. Sehingga untuk menerapkan metode ini adalah dibutuhkan resistor yang tahan panas, disebut juga dengan *power resistor*. Besarnya *heat dissipation* tersebut dapat dihitung dengan menggunakan rumus daya, yaitu $P = I^2R$. Sehingga selain diperoleh nilai hambatan, nilai daya pada hambatan sebesar.

$$P = 0,75^2 A \times 30,3\Omega = 17.044\text{Watt}$$

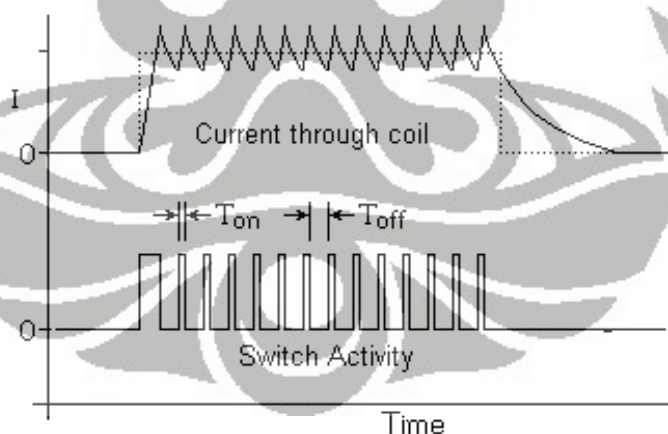
Nilai daya hambatan secara komersil hanya terdapat nilai ¼, 1/2, 1, 2, 5, 10, dan 20 Watt. Sehingga untuk menerapkan metode ini, nilai hambatan yang dibutuhkan adalah 30Ω, 20 Watt.

Selain dengan metode rangkaian *resistive*, ada metode lain yang lebih efisien dan elegan. Yaitu dengan metode rangkaian *chopping current* (sirkuit *Chopper*), pada gambar 4.27.



Gambar 4.27 Rangkaian dasar *chopping current* [26]

Pada rangkaian *chopping current*, yaitu menggunakan hambatan *sense* dan komparator. Prinsip dasar rangkaian *chopping current* adalah membandingkan nilai tegangan antara V_{sense} dengan V_{ref} pada komparator.

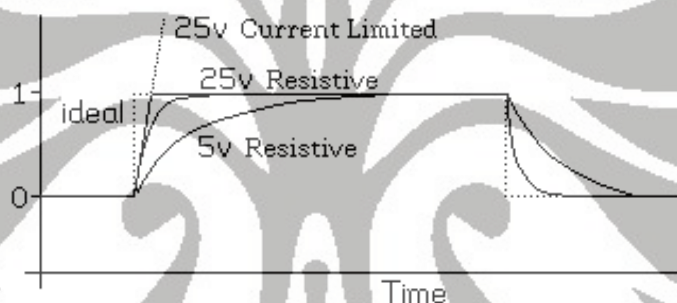


Gambar 4.28 Keadaan aktifitas *switching* pada koil motor stepper [26]

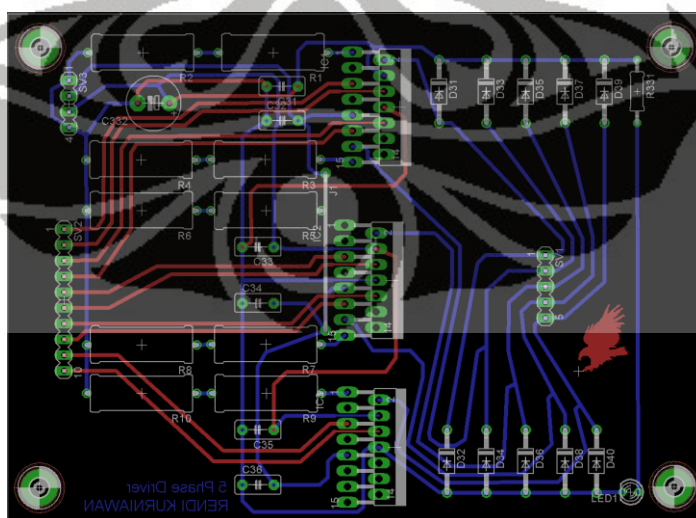
Pada saat *switch* dalam keadaan tertutup, maka arus akan mengalir dari koil ke hambatan *sense*, pada hambatan *sense* terdapat nilai tegangan dan besarnya proporsional terhadap besarnya arus yang mengalir melalui hambatan *sense*. Kemudian besarnya nilai

tegangan pada hambatan *sense* dibandingkan dengan besarnya tegangan referensi (V_{REF}) pada komparator. Jika nilai tegangan pada hambatan *sense* lebih besar dari nilai tegangan referensi pada komparator, maka sinyal logika akan terputus sehingga koil pada motor tidak teraktifasi.

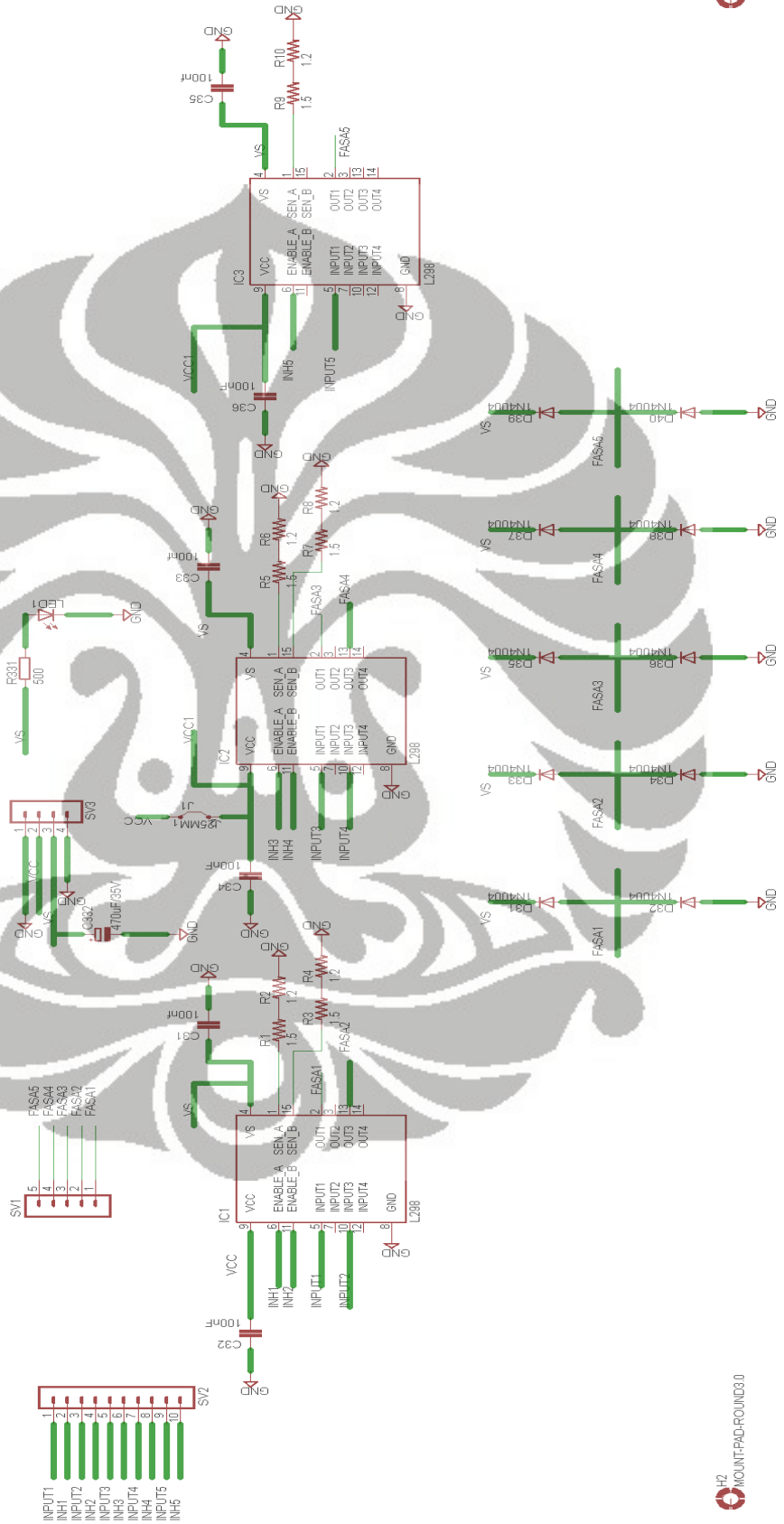
Pada teori pembatasan arus (*chopping current*), dikenal istilah *slew rate*. *Slew rate* adalah kecepatan arus terhadap waktu untuk mencapai nilai maksimum. Pada gambar 4.29, rangkaian *resistive* dengan menggunakan sumber tegangan 5V dibandingkan dengan 25V, *slew rate* yang dihasilkan 25V mendekati bentuk gelombang ideal (*square waveform*). Sedangkan, dengan menggunakan sumber tegangan yang sama (25V), tetapi dengan rangkaian yang berbeda. Rangkaian *chopping current* lebih baik dan mendekati ideal dibandingkan dengan rangkaian *resistive*.



Gambar 4.29 Gelombang *slew rate* pada koil motor stepper [26]



Gambar 4.30 Desain PCB 5 fasa driver motor stepper



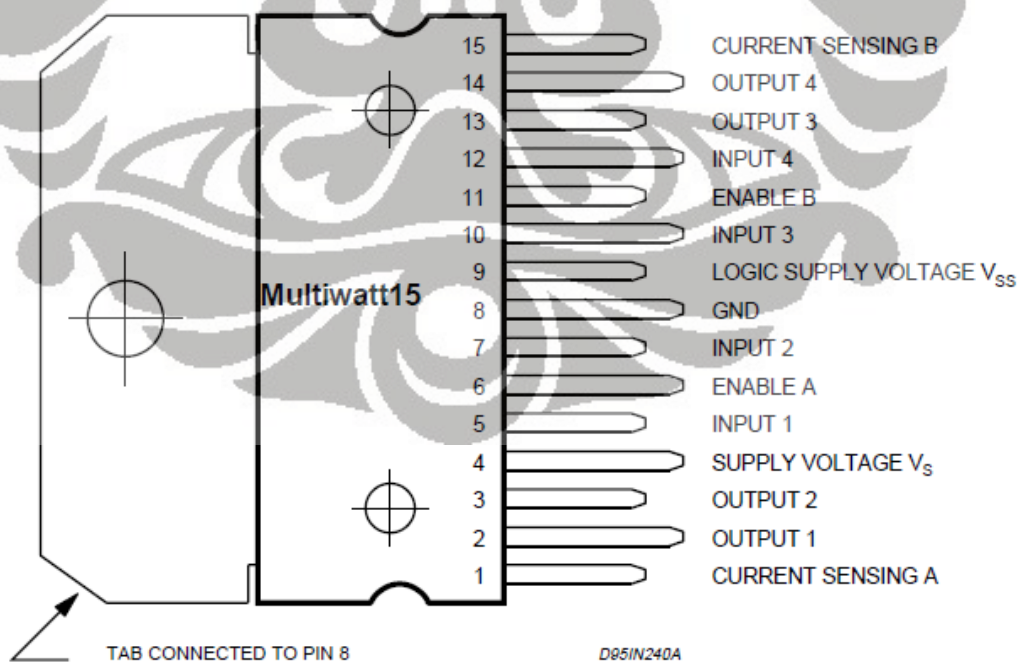
Gambar 4.31 Skema 5 fasa driver motor stepper

4.2.3 Driver motor DC

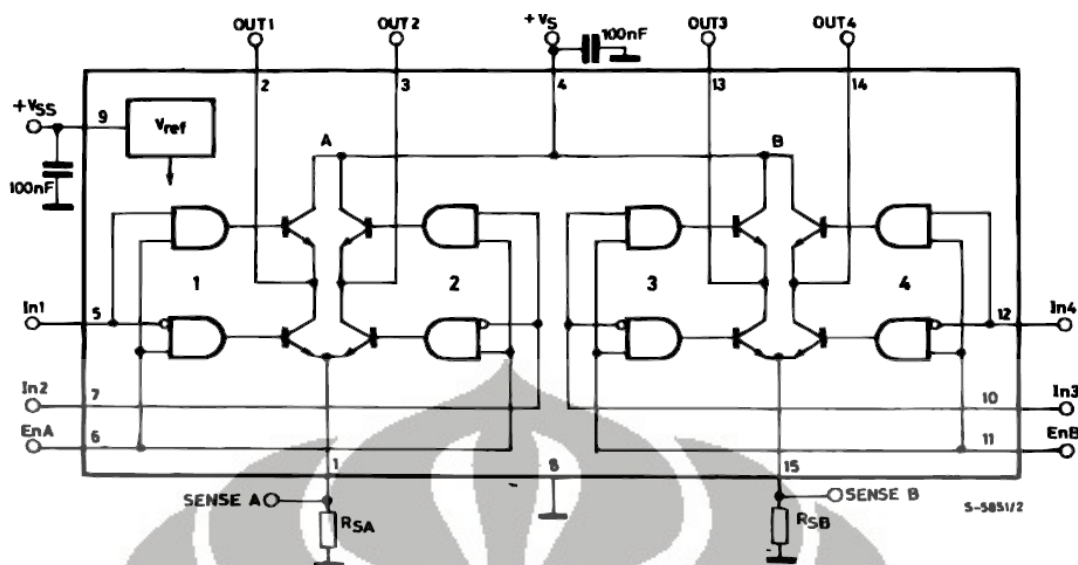
Pada sistem kontrol, driver motor DC yang digunakan adalah *driver* yang sudah dibuat oleh peneliti kontrol robot artikulasi berbasis web [22]. Pada driver motor ini, sama halnya dengan penggunaan L298N pada 5 fasa driver motor stepper. Driver motor DC ini menggunakan L298N sebagai H-bridge, dimana pada L298N terdapat dual H-bridge sebagai pergantian polaritas untuk mengganti arah putar motor DC. Motor DC yang digunakan menggunakan tegangan 12V, terpasang pada mekanisme ekstruder mesin RPFDM. Berikut spesifikasi L298N :

Tabel 4.2 Spesifikasi L298N [27]

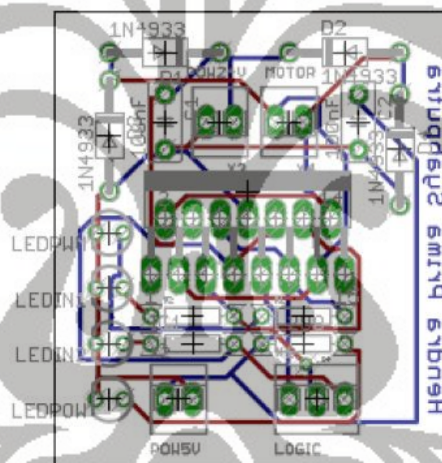
Symbol	Parameter	Value	Unit
V_S	Power Supply	50	V
V_{SS}	Logic Supply Voltage	7	V
V_I, V_{en}	Input and Enable Voltage	-0.3 to 7	V
I_O	Peak Output Current (each Channel)		
	- Non Repetitive ($t = 100\mu s$)	3	A
	- Repetitive (80% on -20% off; $t_{on} = 10ms$)	2.5	A
	-DC Operation	2	A
V_{sens}	Sensing Voltage	-1 to 2.3	V
P_{tot}	Total Power Dissipation ($T_{case} = 75^\circ C$)	25	W
T_{op}	Junction Operating Temperature	-25 to 130	$^\circ C$
T_{stg}, T_j	Storage and Junction Temperature	-40 to 150	$^\circ C$



Gambar 4.32 IC L298N [27]



Gambar 4.33 Diagram blok L298N [27]



Gambar 4.34 PCB layout motor driver

4.2.4 Termokopel *amplifier*

Temperatur kontrol berfungsi untuk mengatur temperatur pencairan material agar tetap konstan. Metode pengontrolan yang digunakan adalah dengan menggunakan mikrokontroler ATmega 16 melalui fitur ADC 10 bit yang sudah terintegrasi didalam chip. Sensor yang digunakan untuk mengukur temperatur tersebut adalah termokopel tipe-K, dan termokopel ini sudah cukup linier antara temperatur dengan tegangan, akan tetapi tegangan yang dihasilkan dalam satuan mV sehingga dibutuhkan termokopel

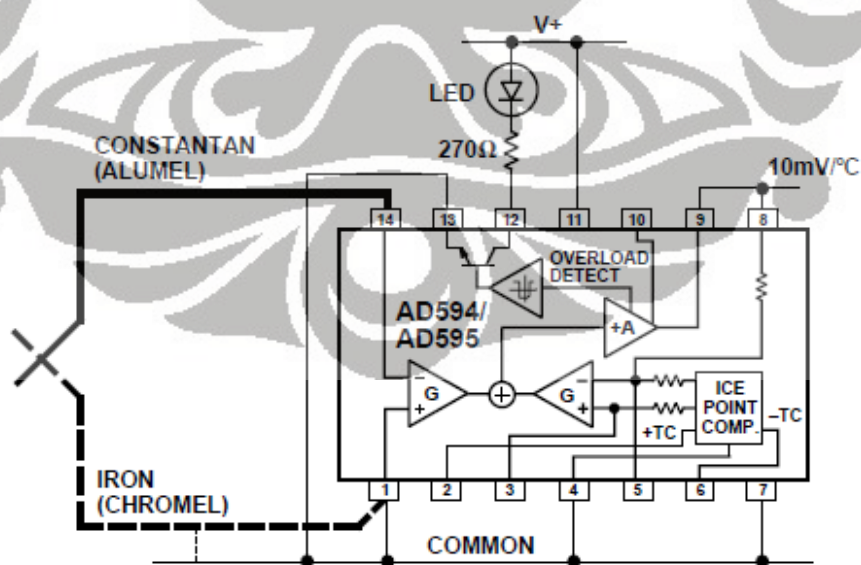
amplifier (penguat tegangan). Setelah melalui termokopel *amplifier* ini, barulah sinyal masuk ke ADC (*Analog Digital Converter*) pada mikrokontroler.

Termokopel *amplifier* yang digunakan adalah AD595 yang mampu menguatkan tegangan setiap kenaikan 10mV per 1 derajat Celcius. Jangkauan temperatur yang akan diukur adalah dari 0 sampai 500 derajat Celcius. Pada mikrokontroler fitur ADC memiliki 10 bit setiap *channel*-nya, sehingga dibutuhkan konversi hitungan sinyal yang masuk ke ADC, karena dengan 10 bit berarti angka karakter mampu mencapai $10 \text{ bit} = 2^{10} = 1024$. Pada ADC menggunakan tegangan referensi AREF 5V, sehingga :

$$\frac{1024 \text{ bit}}{5 \text{ V}} \times 10 \text{ mV} / ^\circ \text{C} = 2,048 \text{ bit} / ^\circ \text{C}$$

Pada perhitungan diatas berarti setiap kenaikan angka 2,048 bit, berarti kenaikan setiap $1 ^\circ \text{C}$, oleh karena itu angka karakter yang didapat melalui ADC dibagi dengan angka 2,048 sehingga jangkauan temperatur yang terukur dari 0 mencapai 500 derajat celcius.

Desain PCB yang dikembangkan menggunakan *alarm*, berupa lampu LED sebagai tanda jika termokopel tidak terpasang dengan baik maka LED akan menyala. Skematik yang dikembangkan mengikuti skematik berdasarkan datasheet AD595.



Gambar 4.35 Rangkaian AD595 dengan *alarm* LED [28]

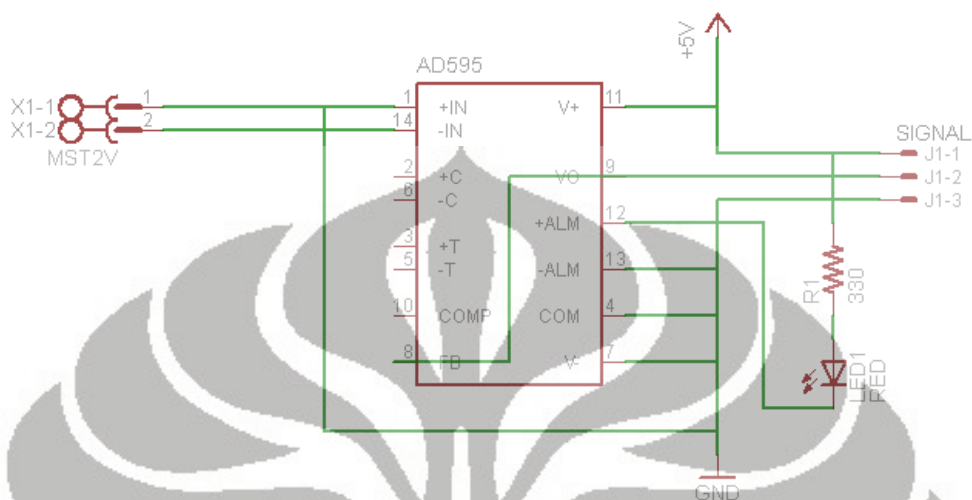
Tabel 4.3 Temperatur terhadap tegangan keluaran AD595 [28]

Thermocouple Temperature °C	Type J Voltage mV	AD594 Output mV	Type K Voltage mV	AD595 Output mV	Thermocouple Temperature °C	Type J Voltage mV	AD594 Output mV	Type K Voltage mV	AD595 Output mV
-200	-7.890	-1523	-5.891	-1454	500	27.388	5300	20.640	5107
-180	-7.402	-1428	-5.550	-1370	520	28.511	5517	21.493	5318
-160	-6.821	-1316	-5.141	-1269	540	29.642	5736	22.346	5529
-140	-6.159	-1188	-4.669	-1152	560	30.782	5956	23.198	5740
-120	-5.426	-1046	-4.138	-1021	580	31.933	6179	24.050	5950
-100	-4.632	-893	-3.553	-876	600	33.096	6404	24.902	6161
-80	-3.785	-729	-2.920	-719	620	34.273	6632	25.751	6371
-60	-2.892	-556	-2.243	-552	640	35.464	6862	26.599	6581
-40	-1.960	-376	-1.527	-375	660	36.671	7095	27.445	6790
-20	-0.995	-189	-0.777	-189	680	37.893	7332	28.288	6998
-10	-0.501	-94	-0.392	-94	700	39.130	7571	29.128	7206
0	0	3.1	0	2.7	720	40.382	7813	29.965	7413
10	0.507	101	0.397	101	740	41.647	8058	30.799	7619
20	1.019	200	0.798	200	750	42.283	8181	31.214	7722
25	1.277	250	1.000	250	760	-	-	31.629	7825
30	1.536	300	1.203	300	780	-	-	32.455	8029
40	2.058	401	1.611	401	800	-	-	33.277	8232
50	2.585	503	2.022	503	820	-	-	34.095	8434
60	3.115	606	2.436	605	840	-	-	34.909	8636
80	4.186	813	3.266	810	860	-	-	35.718	8836
100	5.268	1022	4.095	1015	880	-	-	36.524	9035
120	6.359	1233	4.919	1219	900	-	-	37.325	9233
140	7.457	1445	5.733	1420	920	-	-	38.122	9430
160	8.560	1659	6.539	1620	940	-	-	38.915	9626
180	9.667	1873	7.338	1817	960	-	-	39.703	9821
200	10.777	2087	8.137	2015	980	-	-	40.488	10015
220	11.887	2302	8.938	2213	1000	-	-	41.269	10209
240	12.998	2517	9.745	2413	1020	-	-	42.045	10400
260	14.108	2732	10.560	2614	1040	-	-	42.817	10591
280	15.217	2946	11.381	2817	1060	-	-	43.585	10781
300	16.325	3160	12.207	3022	1080	-	-	44.339	10970
320	17.432	3374	13.039	3227	1100	-	-	45.108	11158
340	18.537	3588	13.874	3434	1120	-	-	45.863	11345
360	19.640	3801	14.712	3641	1140	-	-	46.612	11530
380	20.743	4015	15.552	3849	1160	-	-	47.356	11714
400	21.846	4228	16.395	4057	1180	-	-	48.095	11897
420	22.949	4441	17.241	4266	1200	-	-	48.828	12078
440	24.054	4655	18.088	4476	1220	-	-	49.555	12258
460	25.161	4869	18.938	4686	1240	-	-	50.276	12436
480	26.272	5084	19.788	4896	1250	-	-	50.633	12524

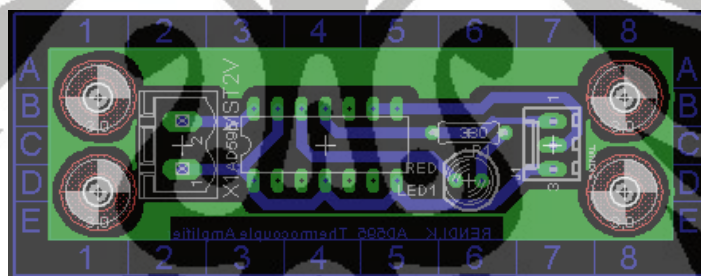
Tabel 4.3 merupakan hubungan antara temperatur dengan tegangan yang terukur dengan menggunakan termokopel *amplifier* (AD595). Hubungan antara temperatur dengan tegangan pada tabel 4.3 sudah mencapai linier. AD595 merupakan termokopel *amplifier* dengan *cold junction compensator* yang sudah terintegrasi di dalam IC. *Compensator* tersebut sudah digabungkan dengan referensi titik beku es dengan *amplifier* yang sudah terkalibrasi sehingga menghasilkan sinyal keluaran $10\text{mV}/^{\circ}\text{C}$ secara langsung dari sinyal masukan dari termokopel. Tegangan keluaran dari termokopel tidak linier terhadap temperatur, oleh karena itu pada AD595 hubungan tersebut dibuatlah *transfer function* agar keluaran tegangan keluaran aktual pada AD595 linier terhadap temperatur.

$$AD595output = (TypeK_Voltage + 11\mu V) \times 247.3$$

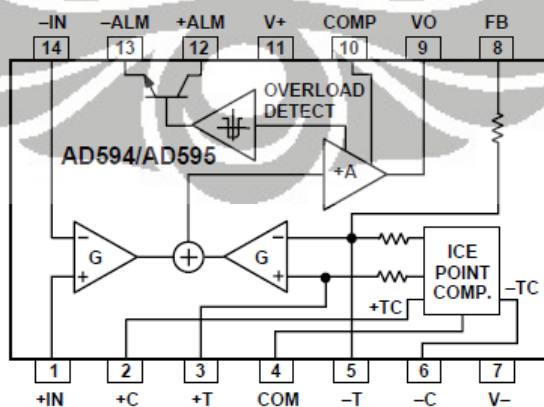
Persamaan diatas adalah hubungan tegangan keluaran termokopel tipe-K dengan tegangan keluaran aktual pada AD595. Dibawah ini, merupakan desain PCB yang telah dikembangkan.



Gambar 4.36 Skema PCB termokopel *amplifier*



Gambar 4.37 Desain PCB termokopel *amplifier*

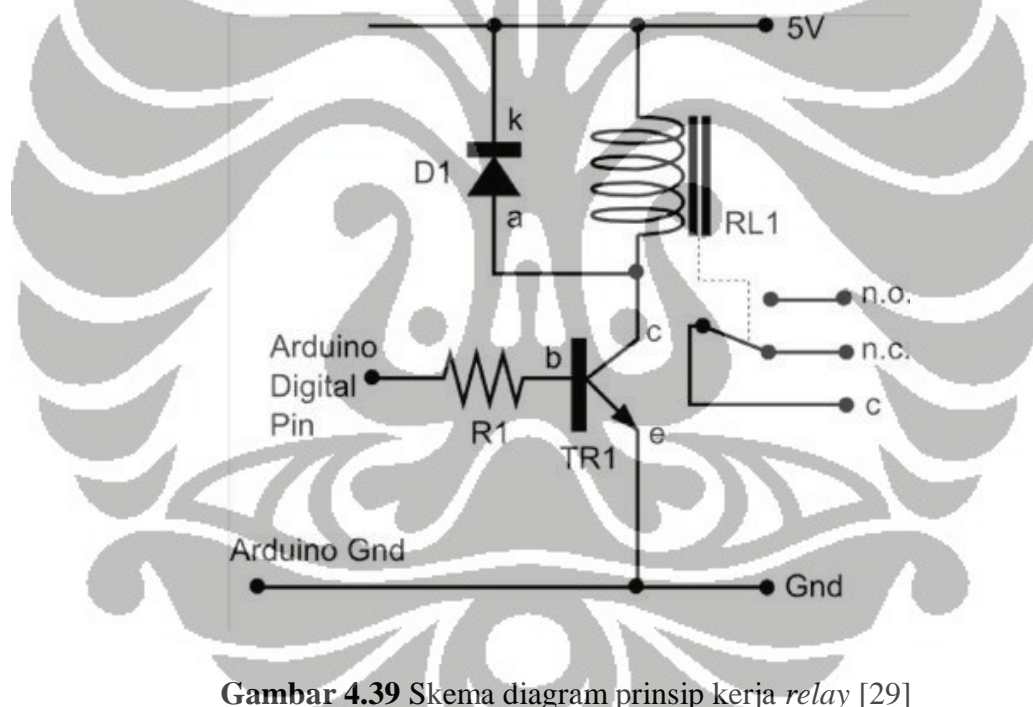


Gambar 4.38 Diagram blok AD595 [28]

4.2.5 Relay

Relay digunakan pada penelitian ini berfungsi sebagai saklar elektromagnetis agar arus yang mengalir ke kawat nicrome dapat diputus jika temperatur yang diharapkan dicapai, sehingga diharapkan temperatur tetap konstan.

Tipe *relay* yang digunakan adalah *solid state relay*, bekerja pada jangkauan 24VDC,10A. Relay dalam kondisi *normally closed*, sehingga setelah mendapat sinyal *trigger* dari mikrokontroller ATmega 16, kondisi relay menjadi *normally open* dan arus berhenti mengalir atau terputus. Oleh karena itu, temperatur yang dicapai tidak pernah konstan pada temperatur tertentu, akan tetapi temperatur akan naik dan turun tidak jauh dari jangkauan temperatur yang dibuat konstan, misalnya temperatur konstan yang diinginkan 250⁰C, maka temperatur yang dicapai bisa pada jangkauan $\pm 10^0$ C.



Gambar 4.39 Skema diagram prinsip kerja *relay* [29]

Ilustrasi pada gambar 4.39 adalah prinsip kerja dari *relay* secara umum. Pada koil atau lilitan dipasangkan diode secara parallel dengan koil, bertujuan agar koil terproteksi terhadap arus balik ketika melakukan proses *switching*. Pada gambar 4.39, kondisi saklar dalam keadaan *normally closed*. Ketika pin yang terhubung ke controller (MCU) memberikan sinyal logika terhadap pin *basis* pada transistor TR1, maka arus dari pin *collector* akan mengalir ke pin *emitter* yang telah terhubung dengan ground

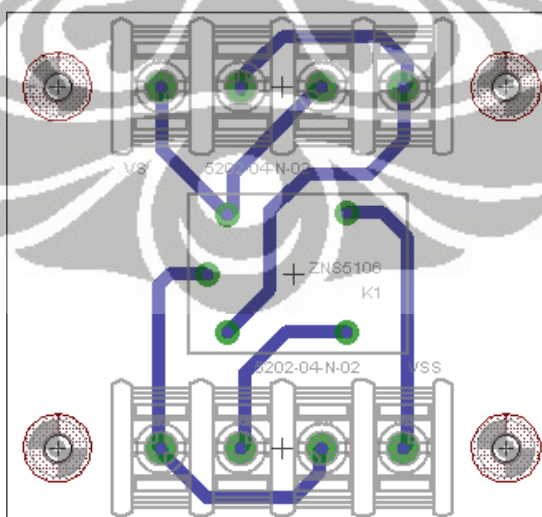
(GND). Maka dengan mengalirnya arus tersebut maka koil RL1 akan teraktifasi dan merubah keadaan saklar dari *normally open* ke *normally closed*. Jangkauan tegangan pada koil *relay* pada umumnya antara 5 sampai 10 Volt, sedangkan untuk jangkauan tegangan saklar bisa mencapai 24V, jangkauan tegangan tersebut tergantung dari produsen yang membuat *relay* tersebut.



Gambar 4.40 Relay HKE [30]



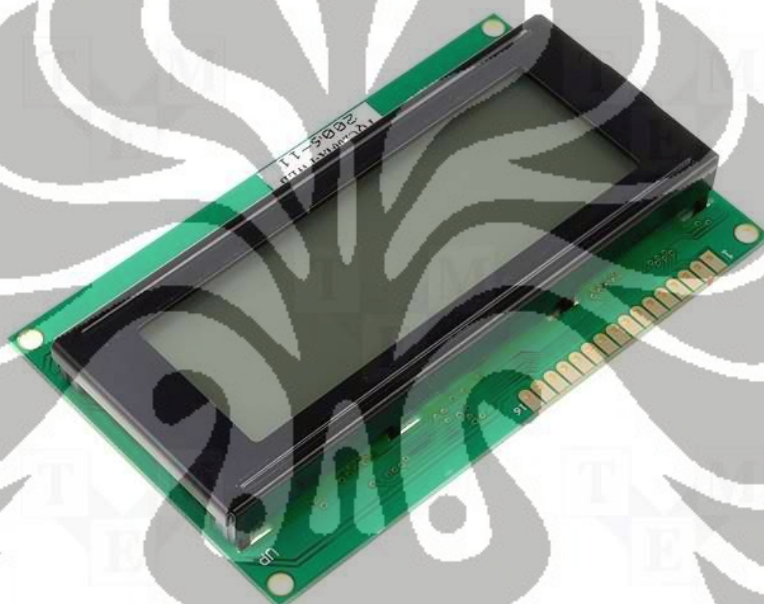
Gambar 4.41 Skema desain PCB relay



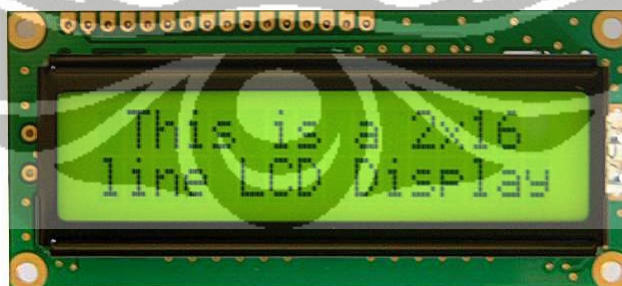
Gambar 4.42 Desain PCB relay

4.2.6 Modul LCD (*Liquid Crystal Display*)

Selain bertujuan untuk *display* modul LCD digunakan pada sistem kontrol bertujuan agar membantu user mengetahui informasi-informasi proses yang sedang berlangsung pada mikrokontroler. LCD yang digunakan ada yang berukuran karakter 20x4 dan 16x2. Mikrokontroler *master* menggunakan modul LCD berkarakter 20x4, sedangkan pada mikrokontroler temperatur kontrol menggunakan modul LCD berkarakter 16x4. Modul LCD ini sangat membantu *programmer* dalam proses *debugging* program pada mikrokontroler.



Gambar 4.43 LCD karakter 20x4 [31]

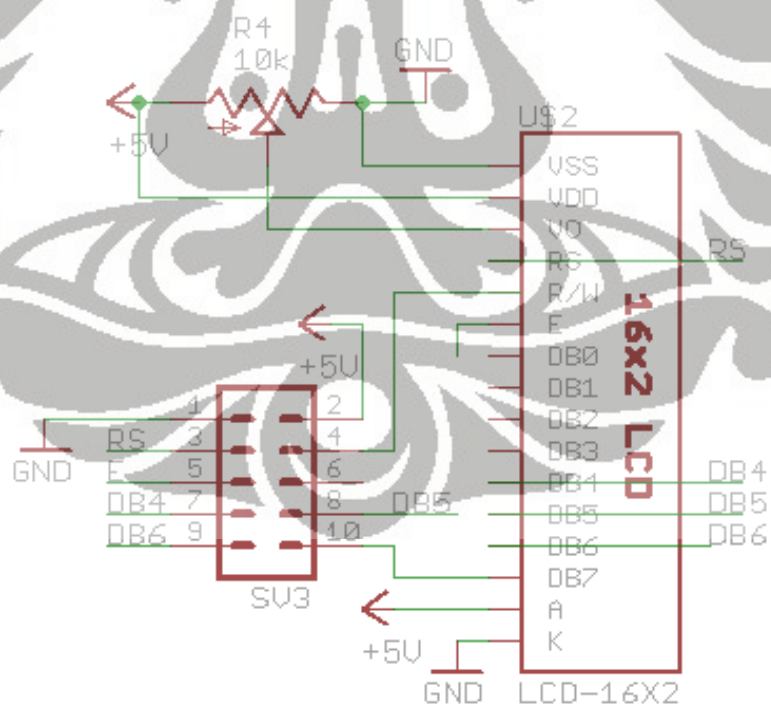


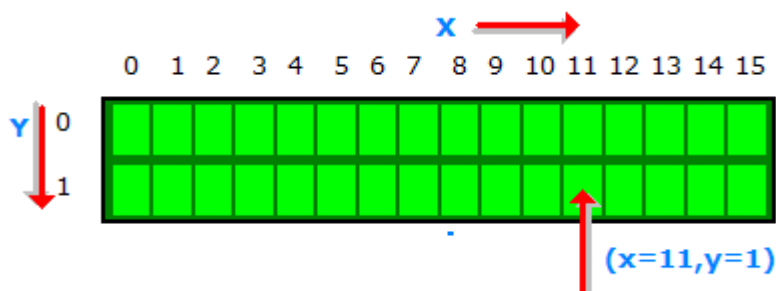
Gambar 4.44 LCD karakter 16x2 [32]

Untuk pin antar muka, modul LCD karakter 16x2 maupun 20x4 sama-sama memiliki 16 pin antar muka untuk dihubungkan langsung ke mikrokontroler. Berikut, tabel pin antar muka untuk modul LCD.

Tabel 4.4 Pin antar muka modul LCD

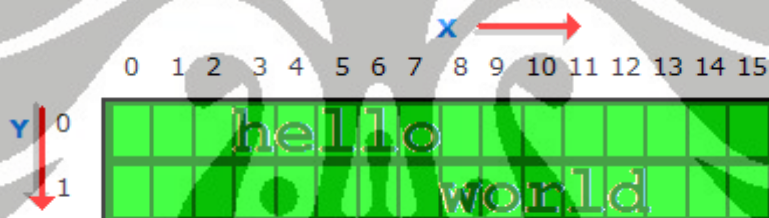
No	Modul LCD	μ C I/O
1	VSS (GND Supply)	GND
2	VCC (+5V)	VCC
3	VEE (Contrast Adjust)	Trimpod 10K
4	RS	PORTC.0
5	R/W	PORTC.1
6	E	PORTC.2
7	DB0	-
8	DB1	-
9	DB2	-
10	DB3	-
11	DB4	PORTC.4
12	DB5	PORTC.5
13	DB6	PORTC.6
14	DB7	PORTC.7
15	LED +	VCC
16	LED -	GND

**Gambar 4.45** Skema modul LCD



Gambar 4.46 Posisi karakter berupa koordinat pada modul LCD [33]

Pada gambar 4.46, merupakan posisi karakter yang akan ditampilkan pada modul LCD. Posisi karakter tersebut digambarkan dengan menggunakan sistem koordinat, untuk sumbu x, dimulai dari posisi 0 sampai 15, sedangkan sumbu y hanya ada dua posisi yaitu 1 dan 0. Setiap blok, hanya dapat diisi oleh satu karakter, jika ingin menampilkan karakter tertentu pada posisi (11,1) diharuskan menuliskan letak posisi karakter tersebut.



Gambar 4.47 Menampilkan "hello world" pada LCD [33]

API (*Application Programming Interface*) sudah dapat digunakan secara langsung tanpa harus mengembangkan pustaka baru. Kumpulan program untuk menampilkan karakter pada modul LCD berupa pustaka dengan bahasa pemrograman yaitu bahasa C. Untuk menampilkan tulisan berupa beberapa karakter harus menyertakan pustaka `<lcd.h>`. Pada pustaka tersebut terdapat program-program diantaranya adalah program untuk meletakkan posisi karakter pada koordinat, menulis karakter pada LCD, menulis *string* pada LCD, dan menghapus karakter yang sudah ditampilkan. Pada modul LCD, data integer tidak dapat ditampilkan secara langsung, akan tetapi harus dikonversi terlebih dahulu ke dalam bentuk karakter kemudian dari karakter ditampilkan ke modul LCD.

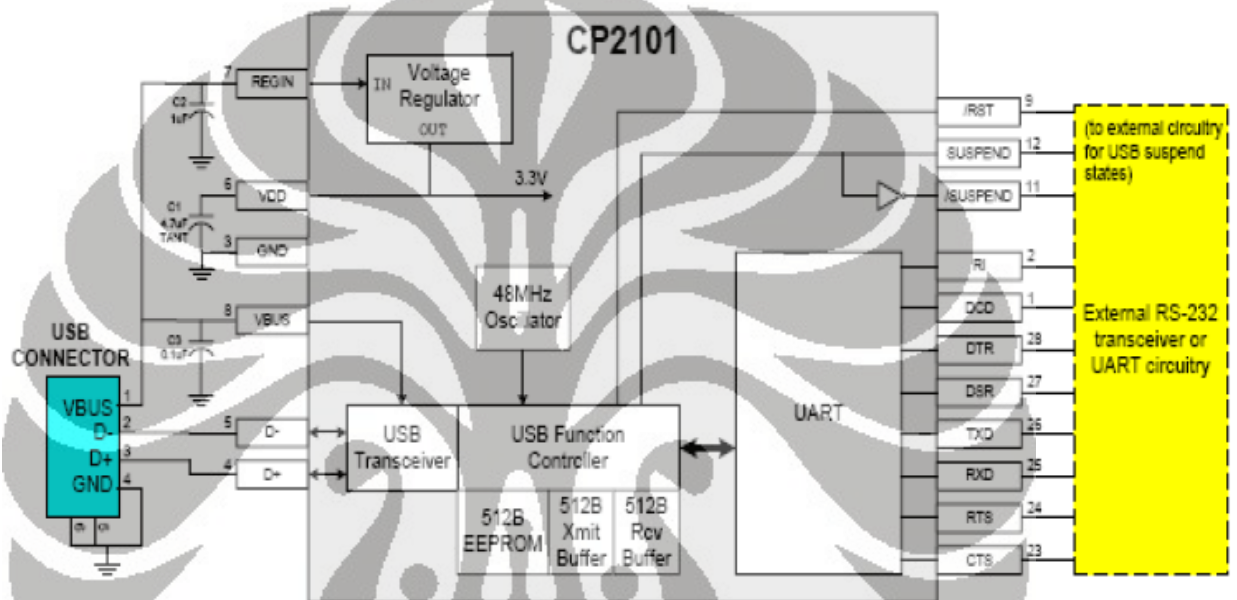
4.2.7 Adaptor USB – Serial

Untuk melakukan transfer data dari *Personal Computer (PC)* ke mikrokontroller diperlukan komunikasi diantara kedua perangkat tersebut. Oleh karena itu, dibutuhkan perangkat elektronik tambahan, yaitu komunikasi serial berupa adaptor USB. Komunikasi serial merupakan komunikasi dimana pengiriman data dilakukan secara per bit. Data yang dikirimkan merupakan data berupa ASCII yang tercantum pada tabel berikut :

Tabel 4.5 Tabel ASCII [22]

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	Ç	160	A0	à	192	C0	L	224	E0	α
129	81	ü	161	A1	á	193	C1	⌞	225	E1	β
130	82	ë	162	A2	â	194	C2	⌟	226	E2	γ
131	83	â	163	A3	ã	195	C3	⌠	227	E3	π
132	84	ä	164	A4	ä	196	C4	⌡	228	E4	Σ
133	85	å	165	A5	å	197	C5	⌢	229	E5	σ
134	86	ä	166	A6	ä	198	C6	⌣	230	E6	μ
135	87	ç	167	A7	ç	199	C7	⌤	231	E7	τ
136	88	ê	168	A8	ê	200	C8	⌥	232	E8	ϕ
137	89	ë	169	A9	ë	201	C9	⌦	233	E9	θ
138	8A	è	170	AA	è	202	CA	⌧	234	EA	ρ
139	8B	ï	171	AB	½	203	CB	⌨	235	EB	δ
140	8C	î	172	AC	¼	204	CC	〈	236	EC	∞
141	8D	ì	173	AD	¼	205	CD	=	237	ED	φ
142	8E	Ë	174	AE	«	206	CE	±	238	EE	ε
143	8F	Ä	175	AF	»	207	CF	±	239	EF	∩
144	90	É	176	B0	⋮	208	D0	⌞	240	F0	≡
145	91	æ	177	B1		209	D1	⌟	241	F1	±
146	92	Œ	178	B2		210	D2	⌠	242	F2	∠
147	93	Ë	179	B3		211	D3	⌡	243	F3	∠
148	94	Ë	180	B4		212	D4	⌢	244	F4	∠
149	95	Ë	181	B5		213	D5	⌣	245	F5	∠
150	96	Ë	182	B6		214	D6	⌤	246	F6	∠
151	97	Ë	183	B7		215	D7	⌥	247	F7	∠
152	98	Ë	184	B8		216	D8	⌦	248	F8	∠
153	99	Ë	185	B9		217	D9	⌧	249	F9	∠
154	9A	Ë	186	BA		218	DA	⌨	250	FA	∠
155	9B	Ë	187	BB		219	DB	■	251	FB	∠
156	9C	Ë	188	BC		220	DC	■	252	FC	∠
157	9D	Ë	189	BD		221	DD	■	253	FD	∠
158	9E	Ë	190	BE		222	DE	■	254	FE	∠
159	9F	Ë	191	BF		223	DF	■	255	FF	∠

Adaptor serial ke USB merupakan perangkat yang menghubungkan mikrokontroller ke komputer dari port RX0 dan TX0 pada mikrokontroller ke port USB pada komputer. Adaptor ini melakukan konversi level tegangan dari komputer ke mikrokontroller dan sebaliknya. Hal ini dilakukan karena komputer menggunakan tegangan logika 12V sedangkan mikrokontroller menggunakan tegangan logika 5V. Adaptor USB – Serial ini menggunakan IC CP2101 single chip USB ke controller UART yang mempunyai diagram blok sebagai berikut :

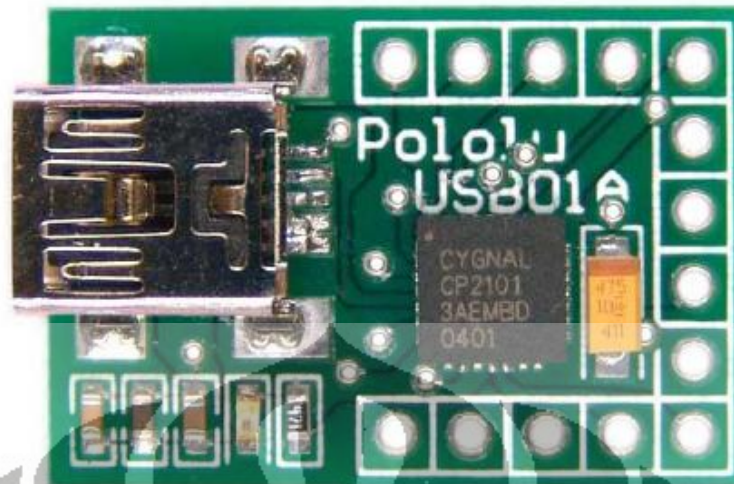


Gambar 4.48 Diagram blok CP2101 [34]

Seperti terlihat pada block diagram, IC tersebut melakukan konversi level tegangan dari PC ke mikrokontroller dan sebaliknya. Untuk konfigurasi komunikasi serial antara komputer dengan mikrokontroller digunakan konfigurasi sebagai berikut :

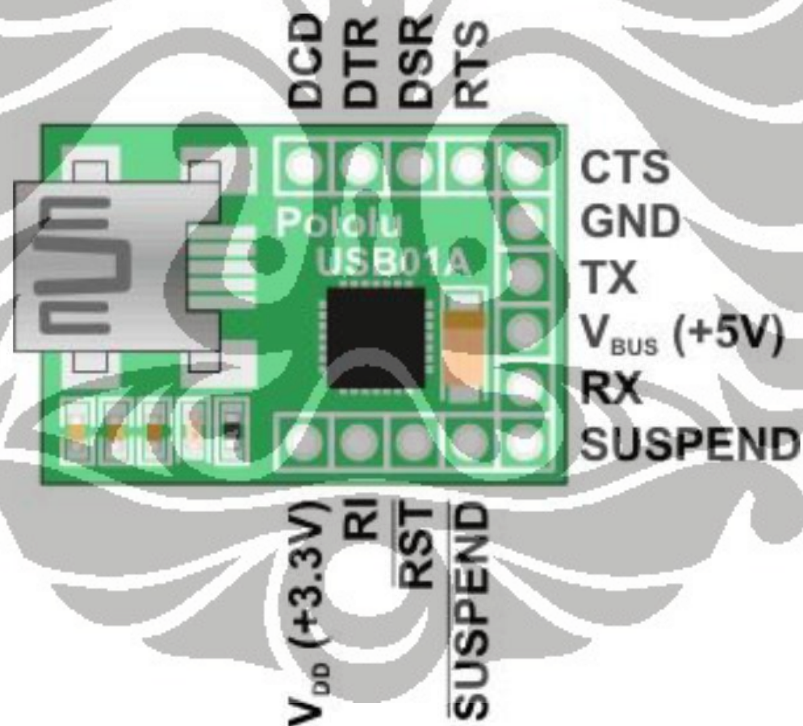
- *Baut Rate* : 9600 bps
- *Data Bits* : 8 bits
- *Parity* : None
- *Stop Bits* : 1
- *Flow Control* : None

Board dari IC CP2101 ini dibuat oleh Pololu (www.pololu.com), berikut adalah skema board IC CP2101 :



Gambar 4.49 USB Pololu ke *serial board* [34]

Berikut adalah *pinout* dari USB Pololu :



Gambar 4.50 Pinout USB Pololu [34]

Board ini mendapatkan daya dari port USB pada komputer yang memiliki tegangan 5V dan dapat langsung terkoneksi ke mikrokontroler dengan menggunakan pin RX dan TX dari board ini.

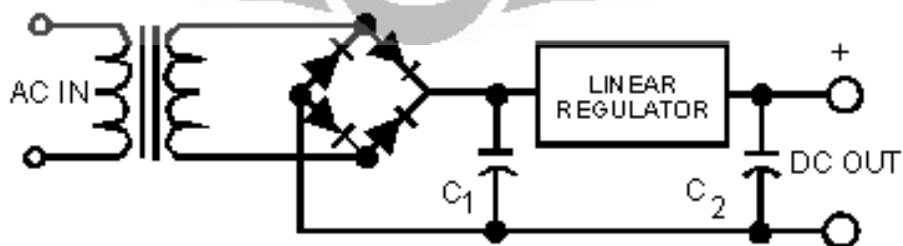
4.2.8 Power supply

Power supply digunakan untuk menyuplai daya ke semua perangkat elektronik, seperti mikrokontroller, driver motor, termokopel *amplifier*, dan modul LCD. *Power supply* tersebut mengkonversi level tegangan dari sumber AC (arus bolak balik) ke tegangan arus searah DC. *Power supply* yang digunakan berdaya 24V 5A, 12V 10A, 5V 30A. *Power supply* yang digunakan pada sistem kontrol, diberikan daya berlebih agar semua perangkat elektronik tidak kekurangan daya. *Power supply* yang digunakan juga sudah ada fitur proteksi *overload* yang sudah terdapat didalam rangkaian *power supply*. *Power supply* yang digunakan merupakan tipe *switching* yang dapat memberikan daya yang stabil dan aman.



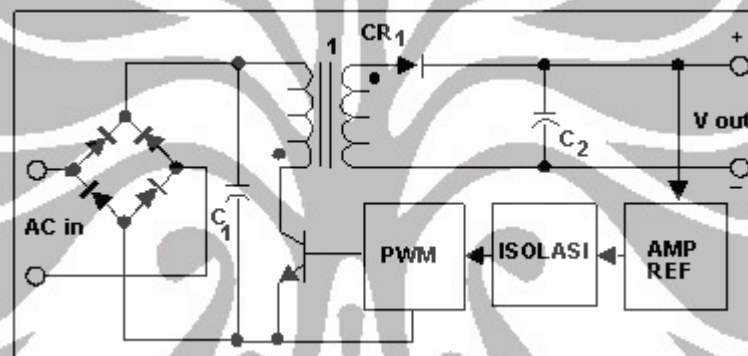
Gambar 4.51 *Power supply* 5V 30A

Power supply tipe *switching* lebih efisien jika dibandingkan dengan tipe linier, pada tipe linier hasil tegangan keluaran penyearah diturunkan tegangannya melalui pengatur linier (*linear regulator*), dan selisih antara tegangan yang masuk dengan tegangan yang dihasilkan dibuang dalam bentuk panas. Akibat pembuangan energi yang besar dalam proses linier tersebut sehingga efisiensinya pun menjadi kecil.



Gambar 4.52 *Power supply* tipe linier [35]

Sedangkan pada *power supply* tipe *switching*, perbaikan efisiensi dicapai dengan cara pengaturan medan magnet akibat selisih dari tegangan masukan dengan keluaran. Pengaturan yang dimaksud berhubungan dengan proses penyimpanan dan pembuangan energi magnet yang mana pada waktu komponen penyimpan energi magnet sampai pada nilai energi tertentu. Maka *switch* yang dipakai untuk mengirim daya ke sisi beban dimatikan (*off state*), dan komponen penyimpan energi magnet tadi kemudian mengambil alih tugas *switch* untuk mengirim daya yang tersimpan menuju ke sisi beban. Apabila ‘tabungan’ energi magnet tadi hampir habis, maka *switch* kembali dihidupkan (*on state*) untuk mengambil alih kembali tugas pengiriman daya ke beban dan secara bersamaan mulai menyimpan kembali energi magnet untuk mengulang proses yang sama.

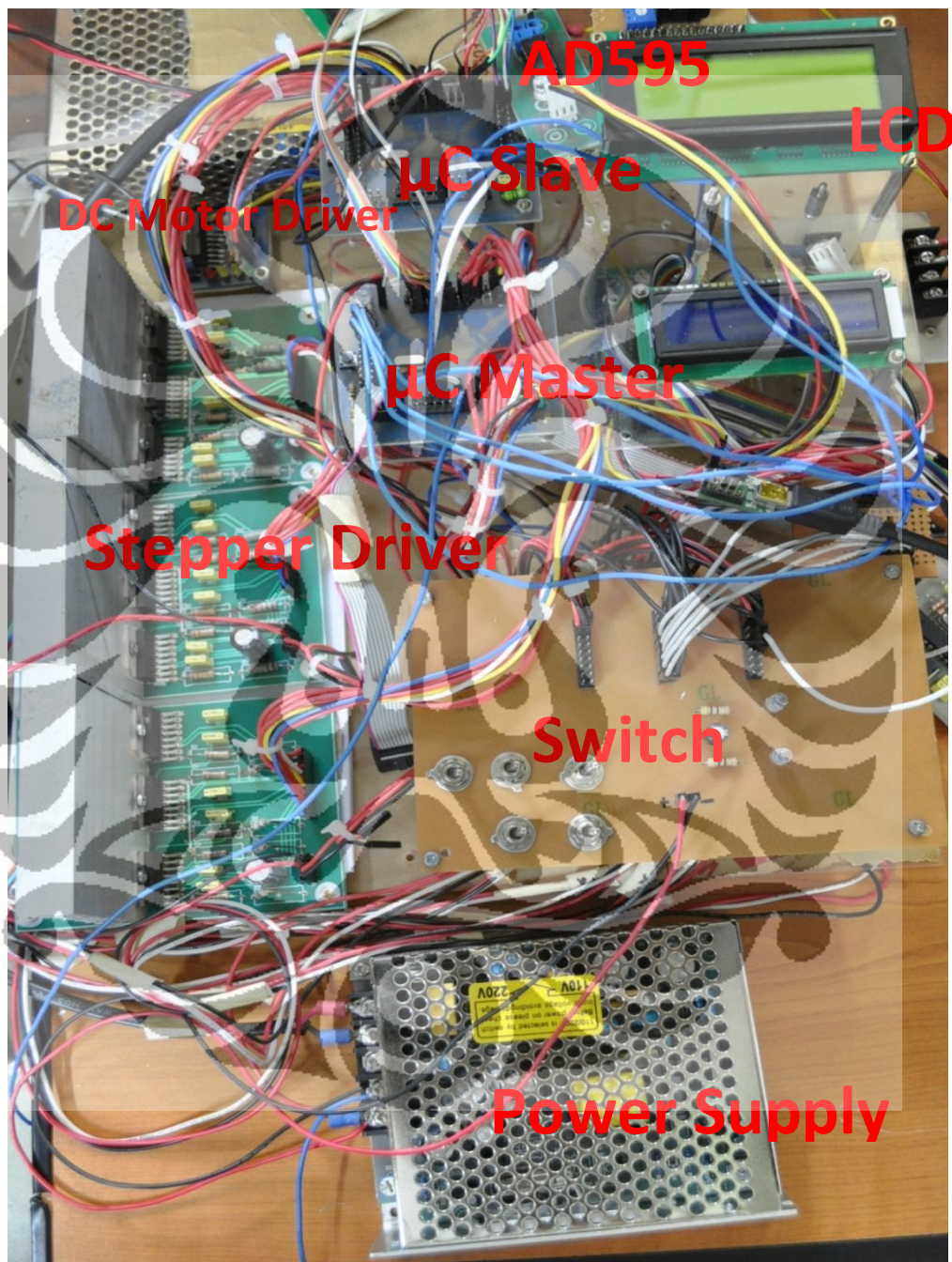


Gambar 4.53 Rangkaian lengkap *flyback regulator* [35]

Salah satu topologi dari *power supply* tipe *switching* adalah metode *flyback* (*flyback regulator*) seperti yang diilustrasikan pada gambar 4.53. Pengaturan besarnya daya keluaran melalui komponen switch dikendalikan dengan metode PWM (*Pulse Width Modulation*) dimana semakin lama *switch* berstatus *ON* semakin banyak energi yang disimpan di dalam *transformer* dan semakin besar pula daya yang dikirim ke beban. Selain itu, untuk menghasilkan tegangan keluaran yang stabil, maka tegangan tersebut dapat diumpun balik dan dibandingkan dengan tegangan referensi (*reference voltage*) dan selisihnya kemudian dapat digunakan untuk mengendalikan lamanya *switch* berstatus *ON* dan *OFF*. Sebutan lain *power supply* tipe *switching* adalah tipe “*off-line*” karena tegangan DC yang menjadi masukan adalah melalui proses penyearah *Bridge* dari tegangan AC atau dari jaringan listrik dengan tanpa menggunakan *transformer* 50 atau 60 Hz.

4.3 Kontrol Unit

Seluruh perangkat elektronik diatas kemudian dirangkai sehingga menjadi sebuah perangkat keras kontrol unit mesin RPFDM. Berikut adalah gambar kontrol unit tersebut :



Gambar 4.54 Perangkat keras kontrol unit

BAB 5

PENGEMBANGAN PERANGKAT LUNAK

MCU PWM DAN TEMPERATUR KONTROL

Pada penelitian pengembangan mesin RPFDM, pengembangan perangkat lunak hanya terbatas untuk MCU PWM dan temperatur kontrol. Disebabkan *software* (perangkat lunak) komunikasi antara *Personal Computer* (PC) dengan MCU *master* maupun perangkat lunak pada MCU *slave* seluruhnya dikembangkan oleh saudara Andry Sulaiman, yang bergabung didalam satu tim dengan peneliti bersama-sama mengembangkan penelitian mesin RPFDM ini.

Dasar pengembangan perangkat lunak, berdasarkan *hardware* maupun fitur-fitur yang digunakan pada mikrokontroler. MCU PWM dan temperatur kontrol merupakan dua *hardware* yang terpisah, masing-masing menggunakan MCU AVR ATmega 16 dari ATMEL. Pada penelitian sebelumnya, MCU PWM dan temperatur kontrol dikonsepsikan agar dapat berkomunikasi dengan MCU *master* tidak lain adalah ATmega 2560, yang memiliki 4 USART (*Universal Synchronous Asynchronous Receiver Transmitter*). Akan tetapi, terjadi hal yang tidak diinginkan yaitu, MCU menjadi cepat panas dan akhirnya rusak. Pada penelitian sekarang, MCU PWM dan temperatur kontrol dikonsepsikan berkomunikasi dengan MCU *slave* sebagai media untuk mengirim dan menerima data dari kedua MCU tersebut. Sampai saat ini, sistem kontrol yang telah dikembangkan berjalan dengan sangat baik.

MCU PWM kontrol dikonsepsikan menerima data dari MCU *slave*, dan setelah itu MCU tersebut memproses algoritma yang telah ditanam didalamnya agar dapat mengatur kecepatan motor DC. Sedangkan, MCU temperatur kontrol dikonsepsikan mengolah data tegangan masukan dari termokopel tipe-K sebagai masukan ke ADC (*Analog Digital Converter*) kemudian mengirim data ke MCU *slave*. Berikut algoritma pada MCU PWM dan temperatur kontrol.

5.1 Algoritma pada MCU PWM kontrol

MCU ATmega 16 digunakan untuk mengontrol kecepatan motor DC pada mekanisme ekstruder. Mikrokontroler ini berkomunikasi dengan MCU *slave*. Dengan mendapatkan sinyal karakter 'y' maka motor DC akan berputar, sedangkan mendapatkan sinyal karakter 'u' motor akan berhenti berputar.

5.1.1 Fungsi utama

Fungsi utama adalah program yang akan dijalankan oleh mikrokontroler PWM kontrol dan siap menerima sinyal dari mikrokontroler *slave*.

Algoritma [158] :

START

- 1> Mendeklarasikan variabel lokal.
- 2> Pengaturan *clock speed* mikrokontroler.
- 3> Mengatur PORT I/O mikrokontroler.
- 4> Inisialisasi USART, komunikasi serial dengan *baud rate* 9600.
- 5> Mengatur *timer* PWM: *Fast PWM*, *clock* 250kHz.
- 6> While, memulai *looping*
 - a. Mengambil data dari USART.
 - b. While, data \neq karakter 'u'
 - i. menjalankan fungsi PWM.
 - ii. mengambil data dari USART.
 - c. While, *looping* kembali
 - i. menghentikan motor.
 - ii. IF, data = 'y'
 - keluar dari looping

END

5.1.2 Pengaturan *clock speed*

Oscillator yang dipakai adalah 16MHz dengan faktor pembagi 1. Kecepatan mikrokontroller dapat diatur dengan menaikkan atau menurunkan faktor pembagi.

Algoritma :

START

- 1> Mengatur *oscillator*.
- 2> Mengatur besarnya faktor pembagi.

END

5.1.3 Pengaturan PORT I/O pada mikrokontroller

Pengaturan PORT I/O bertujuan untuk mengatur fungsi setiap PORT, seperti masukan dan keluaran pada setiap PIN.

Algoritma :

START

- 1> *Setting* PORT A
- 2> *Setting* PORT B
- 3> *Setting* PORT C
- 4> *Setting* PORT D

END

5.1.4 Inisialisasi USART

Fungsi pengaturan parameter komunikasi serial, pada serial PORT diaktifkan dengan *baud rate* 9600.

Algoritma :

START

- 1> *Setting tipe data serial communication, 8 Data, 1 Stop, No Parity.*
- 2> *Aktifasi transmitter receiver.*
- 3> *Setting mode serial asynchronous.*
- 4> *Setting baud rate 9600 bps.*

END

5.1.5 Pengaturan *timer* PWM

Timer yang digunakan adalah *timer0*, maka frekuensi *timer* diambil dari *source clock* mikrokontroller.

Algoritma :

START

- 1> Pengaturan sumber clock, sumber clock dari mikrokontroller yaitu 16MHz.
- 2> Pengaturan frekuensi *timer0* (OCR0), 250kHz.
- 3> Pengaturan mode PWM yaitu *Fast PWM*.

END

5.1.6 Fungsi PWM

Fungsi ini mengaktifasi PWM (*Pulse Width Modulation*). Tipe PWM yang digunakan adalah *fast PWM* dengan rumus $f_{OCnPWM} = \frac{f_{clk_I/O}}{N \times 256}$.

Nilai *prescaler* yang diambil adalah $N=0.65 \times 256=166.4$, dengan $f_{clk_I/O} = 250$ kHz.

Maka frekuensi PWM :

$$f_{OCnPWM} = \frac{f_{clk_I/O}}{N \times 256}$$

$$f_{OCnPWM} = \frac{250.10^3 \text{ Hz}}{166.4 \times 256} = 5,868 \text{ Hz}$$

$$t_{OCnPWM} = \frac{1}{5,868} = 0,17 \text{ sekon}$$

Algoritma :

START

- 1> aktifasi nilai ORCO = N% dari 256. N=nilai *prescaler*.
- 2> Aktifasi PORTD.3 dan PORTD.4 sebagai arah putar motor DC.
- 3> PORTB.3 = 1, mengaktifasi PIN *enable* pada L298.

END

5.2 Algoritma pada MCU temperatur kontrol

Hardware yang digunakan pada temperatur kontrol adalah mikrokontroler ATmega 16 produksi dari ATMEL. MCU ini diprogram untuk membaca data dari ADC (*Analog Digital Converter*) yang tidak lain adalah PORTA. *Channel* yang dibaca adalah *channel 0*, kemudian setelah dibaca data tersebut, dikonversi dan ditampilkan ke LCD. Jika temperatur yang diinginkan sudah dicapai, maka MCU akan mengirim sinyal karakter 'c' ke MCU *slave* melalui USART dan mengaktifasi relay dengan memanfaatkan PORTB.0 sebagai *trigger*.

5.2.1 Fungsi utama

Fungsi utama adalah program yang akan dijalankan oleh mikrokontroler temperatur kontrol. MCU membaca data dari ADC dan menampilkan ke LCD.

Algoritma [156] :

START

- 1> Mendeklarasikan variabel lokal.
- 2> Mengatur *clock speed* mikrokontroller.
- 3> Mengatur PORT I/O mikrokontroller.
- 4> Inisialisasi USART, komunikasi serial dengan *baud rate* 9600.
- 5> Inisialisasi ADC; 125kHz, free running.
- 6> Inisialisasi modul LCD.
- 7> While, memulai *looping*
 - a. For, mengambil data 32 kali
 - b. For, menjumlahkan semua data.
 - c. mencari rata-rata data.
 - d. memasukkan persamaan kalibrasi.
 - e. mengkonversi data *float* ke data *ascii*.
 - f. Menampilkan ke LCD.
 - g. IF, nilai $\geq T$
 - i. PORTB.0 = 1;
 - ii. Mengirim sinyal karakter 'c' melalui USART.
 - h. ELSE, PORTB.0 = 0;

END

5.2.2 Pengaturan *clock speed*

Oscillator yang dipakai adalah 16MHz dengan faktor pembagi 1. Kecepatan mikrokontroller dapat diatur dengan menaikkan atau menurunkan faktor pembagi.

Algoritma :

START

- 1> *Setting oscillator*.
- 2> *Setting faktor pembagi*.

END

5.2.3 Pengaturan PORT I/O mikrokontroller

Pengaturan PORT I/O bertujuan untuk mengatur fungsi setiap PORT, seperti masukan dan keluaran pada setiap PIN.

Algoritma :

START

- 1> *Setting PORT A*
- 2> *Setting PORT B*
- 3> *Setting PORT C*
- 4> *Setting PORT D*

END

5.2.4 Inisialisasi USART

Fungsi pengaturan parameter komunikasi serial, serial PORT diaktifkan dengan *baud rate* 9600.

Algoritma :

START

- 1> *Setting tipe data serial communication, 8 Data, 1 Stop, No Parity.*
- 2> *Aktifasi transmitter receiver.*
- 3> *Setting mode serial asynchronous.*
- 4> *Setting baud rate 9600 bps.*

END

5.2.5 Inisialisasi ADC

Fungsi pengaturan parameter ADC, frekuensi clock, tegangan referensi, metode trigger ADC.

Algoritma :

START

- 1> Pengaturan frekuensi clock ADC 125 kHz.
- 2> Mengatur tegangan referensi, $V_{ref} = \text{PIN AREF}$.
- 3> Mengatur *source trigger* ADC sebagai *free running*.

END

5.2.6 Inisialisasi LCD

Fungsi untuk pengaturan LCD 16 karakter pada fungsi utama dan mengatur LCD pada PORT C.

Algoritma :

START

- 1> Menjalankan fungsi *lcd_init(16)*, yang terdapat pada *library <lcd.h>*.

END

Fungsi pengaturan LCD pada PORTC, fungsi dibawah ini berada diluar fungsi utama.

Algoritma :

START

- 1> Mulai penulisan *in-line assembly language*.
- 2> *Setting* LCD pada PORT C.
- 3> Tutup penulisan *in-line assembly language*.

END

BAB 6

PENGUJIAN DAN ANALISIS SISTEM KONTROL

DRIVER MOTOR STEPPER

6.1 Tujuan pengujian

Pengujian dilakukan untuk mengetahui performa sistem kontrol (driver motor stepper) terhadap hardware (motor stepper). Apakah terjadi error dalam proses pengiriman data *logic* dari mikrokontroler sampai ke pergerakan motor stepper.

6.2 Peralatan pengujian

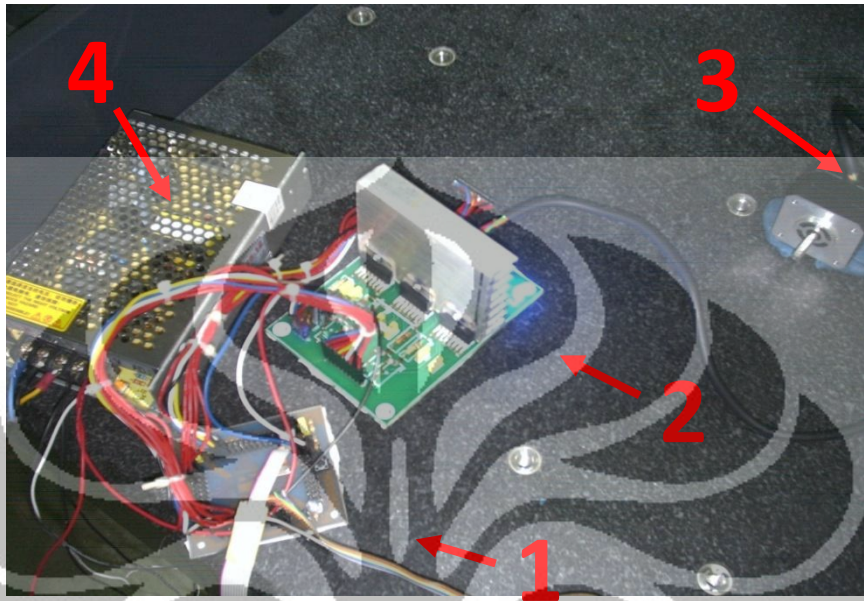


Gambar 6.1 *Coordinate Measuring Machine (CMM)*

Peralatan uji yang digunakan selama proses pengujian adalah sebagai berikut :

1. Satu perangkat alat lengkap CMM (*Coordinate Measuring Machine*) dan Personal Komputer (PC).
2. Satu unit mikrokontroler ATmega 2560.
3. Satu unit driver motor stepper.
4. Satu USB *programmer AVR910*.

5. Satu unit *power supply* 12V, 10 Ampere.
6. Satu unit motor stepper seri A1K-S543(W).
7. Satu perangkat komputer *portable* (Laptop), sebagai *debugging* program.



Gambar 6.2 Peralatan yang akan diuji

- (1) *Microcontroller ATmega 2560*
- (2) *Driver stepper motor*
- (3) *Stepper motor A1K-S543(W)*
- (4) *Power supply 12V, 10 Ampere*

6.3 Metode pengujian

Tujuan dari pengujian ini adalah untuk mengetahui error yang terjadi pada pengiriman data *logic* dari mikrokontroler sampai ke pergerakan motor stepper. Dikarenakan keterbatasan alat untuk menguji pengiriman proses data yang terjadi pada mikrokontroler. Maka pengujian dilakukan pada motor stepper dengan mengukur penyimpangan sudut yang terjadi. Berapa penyimpangan sudut yang terjadi pada poros (*shaft*) jika kita memberikan satu putaran penuh yaitu 500 pulsa pada mikrokontroler. Kemudian dengan merubah kecepatan transfer data dengan kenaikan *delay_us* 200. Pengujian dilakukan dengan dimulai dari *delay_us* 1000, kemudian dirubah menjadi 800, 600, dan terakhir adalah 400. Kemudian pengujian yang lain dilakukan dengan merubah jumlah pulsa dari 500, menjadi 25000 pulsa pada *delay_us* 400.

Pada intinya adalah mendapatkan data sudut pada poros, maka pengukuran dilakukan dengan mengukur bidang datar yang ada pada poros motor stepper. Sehingga, data yang diperoleh adalah data bidang rata-rata dalam dimensi x, y, dan z. Poros motor diputar sebanyak satu kali kemudian bidang poros yang datar diukur, kemudian untuk mendapatkan perbedaan sudut maka vektor bidang rata-rata yang berikutnya dibandingkan dengan vektor bidang rata-rata yang sebelumnya sudah terukur. Berikut penggunaan mesin CMM secara lebih rinci.

Langkah-langkah pengukuran bidang datar dengan CMM :

- 1) Langkah pertama adalah mehidupkan dan mempersiapkan mesin CMM serta jarum probe yang akan digunakan. Pilih probe yang akan digunakan, karena ada berbagai jenis probe dengan diameter yang berbeda, pada pengujian ini, peneliti menggunakan probe dari RENISHAW dengan diameter 1,992 mm.

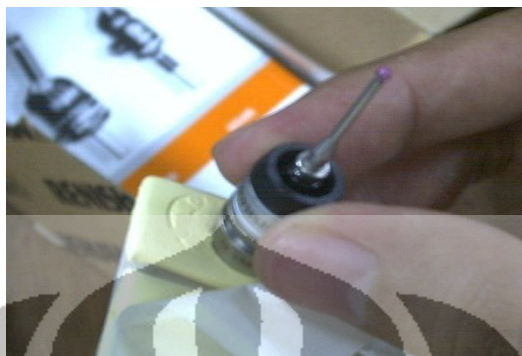


Gambar 6.3 Berbagai jenis probe dengan diameter dan fungsi yang berbeda-beda



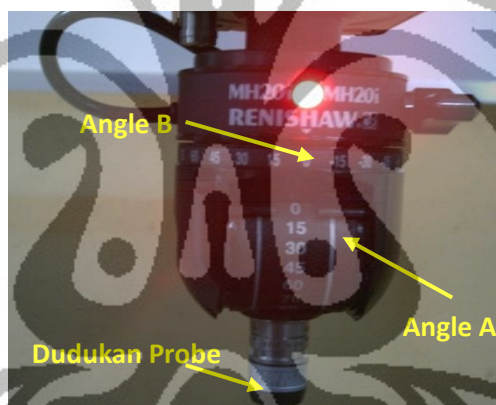
Gambar 6.4 Probe dipasang pada dudukannya

- 2) Kemudian dudukan probe dibersihkan dengan pembersih yang sudah disediakan oleh RENISHAW.



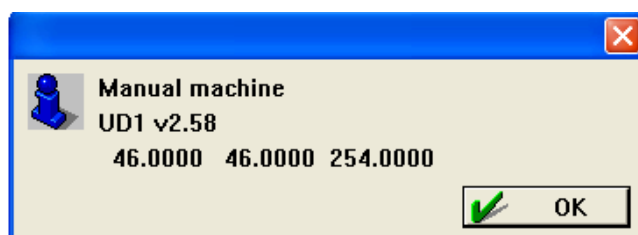
Gambar 6.5 Dudukan probe dibersihkan

- 3) Setelah itu, dudukan tersebut dipasang pada mesin CMM, dengan sudut $A=0^{\circ}$ dan $B=0^{\circ}$.



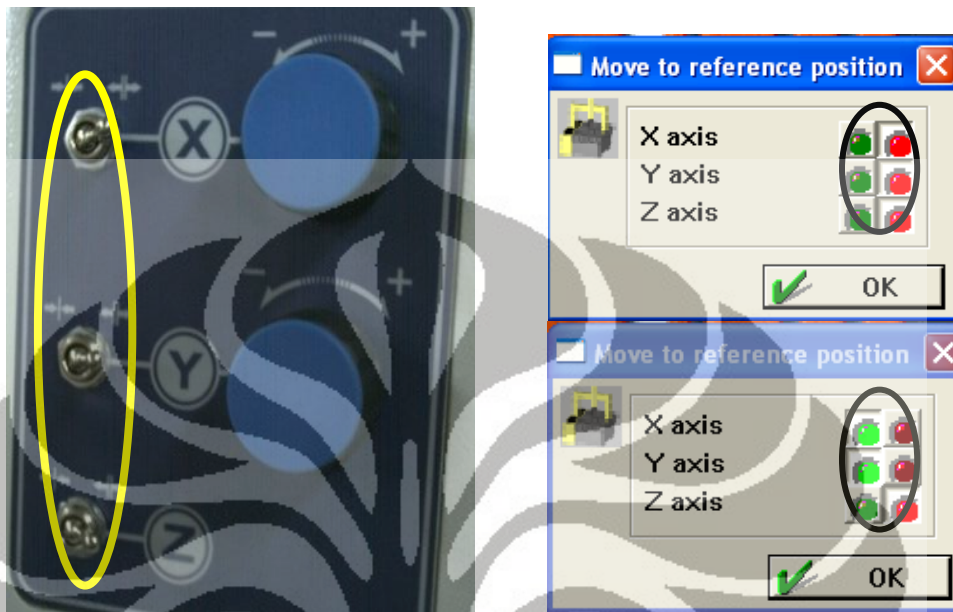
Gambar 6.6 Dudukan probe dipasang pada head mesin CMM

- 4) Setelah persiapan selesai, maka kita akan beralih ke perangkat lunak bawaan dari Mitutoyo pada komputer (PC), klik OK untuk proses penentuan MCS (*Machine Coordinate System*) pada mesin CMM.



Gambar 6.7 Awal tampilan pada perangkat lunak

- 5) Kemudian, buka semua katup pada mesin CMM, dan gerakkan secara manual, pada arah x, y, dan z sampai lambang hijau pada software aktif. Seperti gerakan komponen mesin yang tersentuh oleh *limit switch*.



Gambar 6.8 Proses penentuan titik referensi pada mesin CMM

- 6) Setelah mendapatkan titik referensi mesin atau MCS, maka pada perangkat lunak akan menampilkan koordinat posisi *head* probe pada CMM.



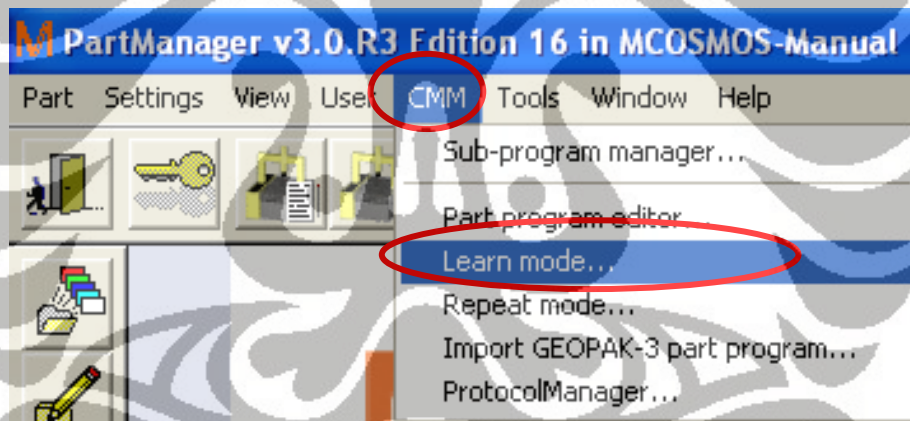
Gambar 6.9 Posisi *head* probe pada CMM

- 7) Setelah mendapatkan titik referensi, kemudian keluar dari tampilan dan buka perangkat lunak bawaan dari Mitutoyo yang ada pada desktop.



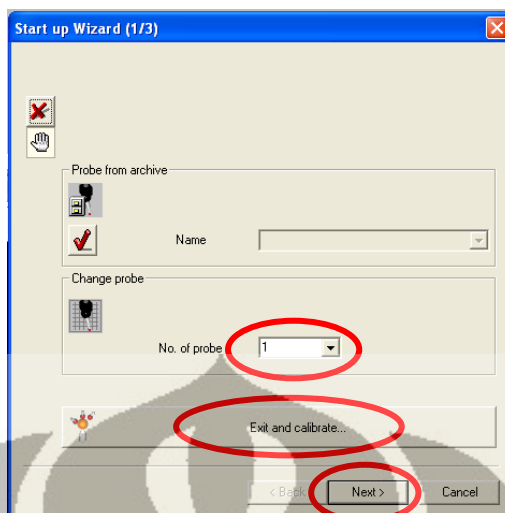
Gambar 6.10 Buka program MCOSMOS v.3 - *manual*

- 8) Setelah masuk program MCOSMOS, buatlah program baru dengan pilih pada *toolbar* > CMM > *learn mode*.



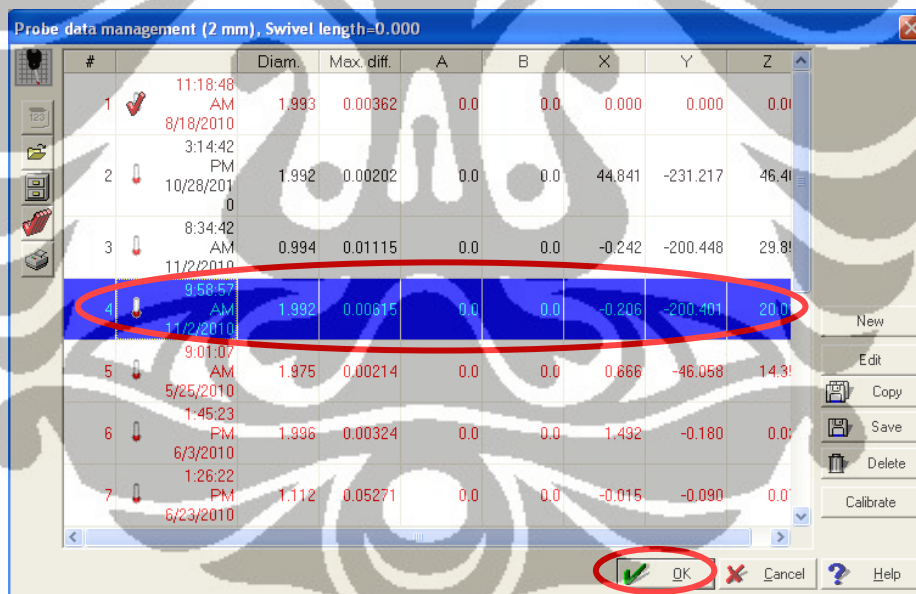
Gambar 6.11 Pilih menu *learn mode* untuk memulai pengukuran

- 9) Setelah memilih mode *learn mode*, untuk memulai pengukuran baru maka langkah selanjutnya adalah proses kalibrasi probe. Proses kalibrasi yaitu dengan menyentuhkan ujung probe sebanyak 10 kali untuk mendapatkan nilai diameter, posisi dan sudut probe. Jika sudah diketahui jenis probe-nya dan pilih probe ke berapa, maka bisa langsung '*Next*', atau jika menginginkan kalibrasi ulang probe dan melihat jenis-jenis probe yang sudah dikalibrasi pilih menu '*Exit and calibrate*'.




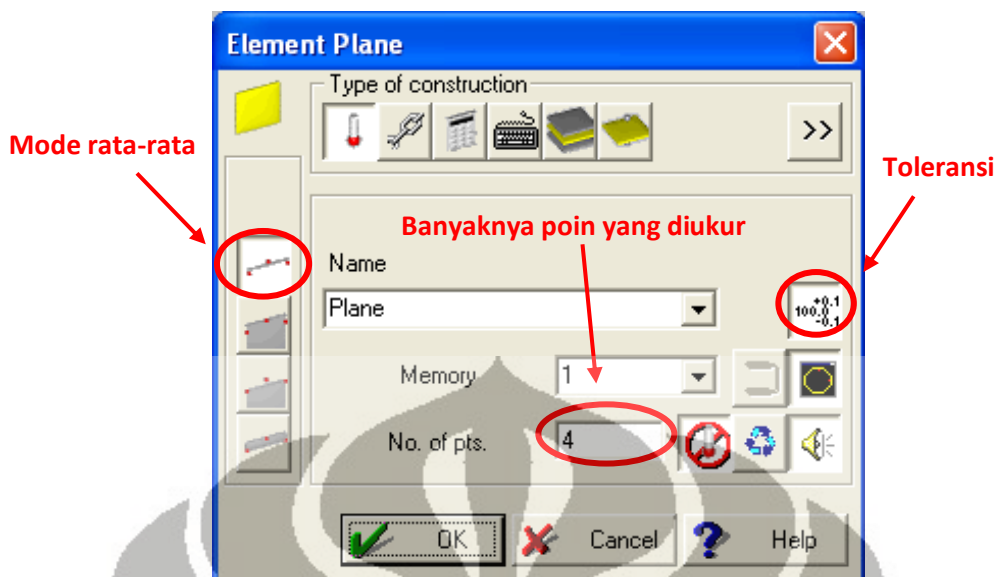
Gambar 6.12 Pilih jenis probe atau kalibrasi ulang

- 10) Pilih button '*Exit and calibrate*', untuk proses kalibrasi ulang dan melihat jenis probe yang sudah di kalibrasi. Pilih jenis probe yang ke-4 dengan diameter 1,992 mm, kemudian '*OK*'.



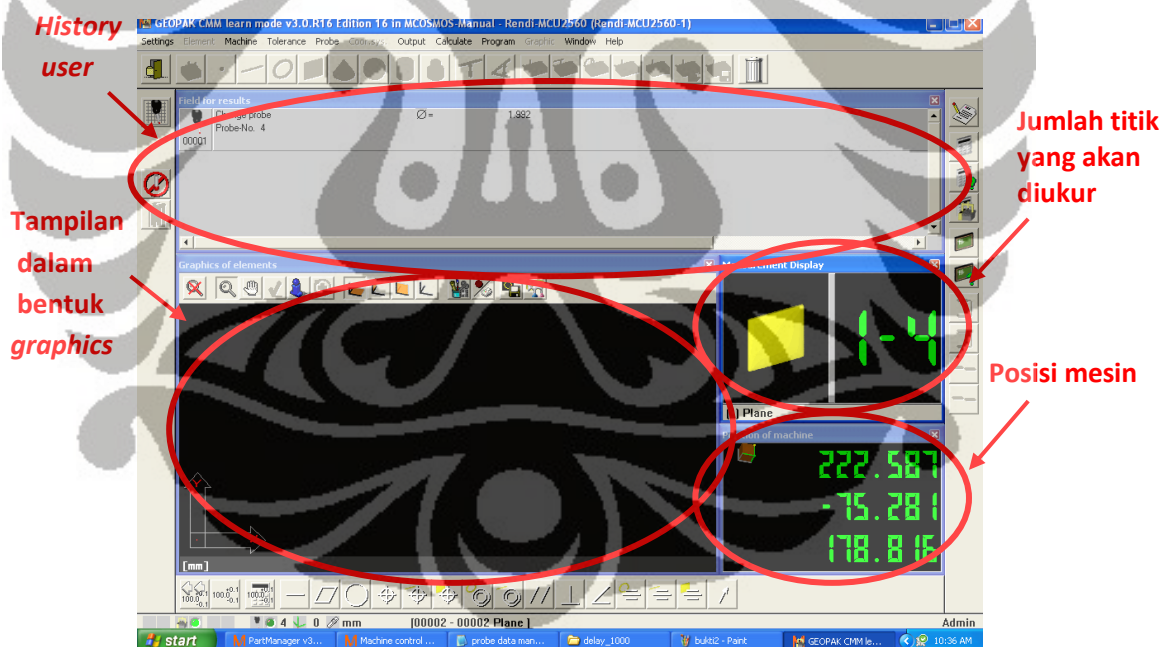
Gambar 6.13 GUI probe data management

- 11) Jika probe sudah dipilih, langkah selanjutnya adalah memilih *button*  untuk mengukur bidang datar atau *plane*.
- 12) Kemudian pilih mode pengukuran rata-rata, dan pilih *button* toleransi, jika diinginkan diketahui apakah bidang datar tersebut melewati toleransi apa tidak.



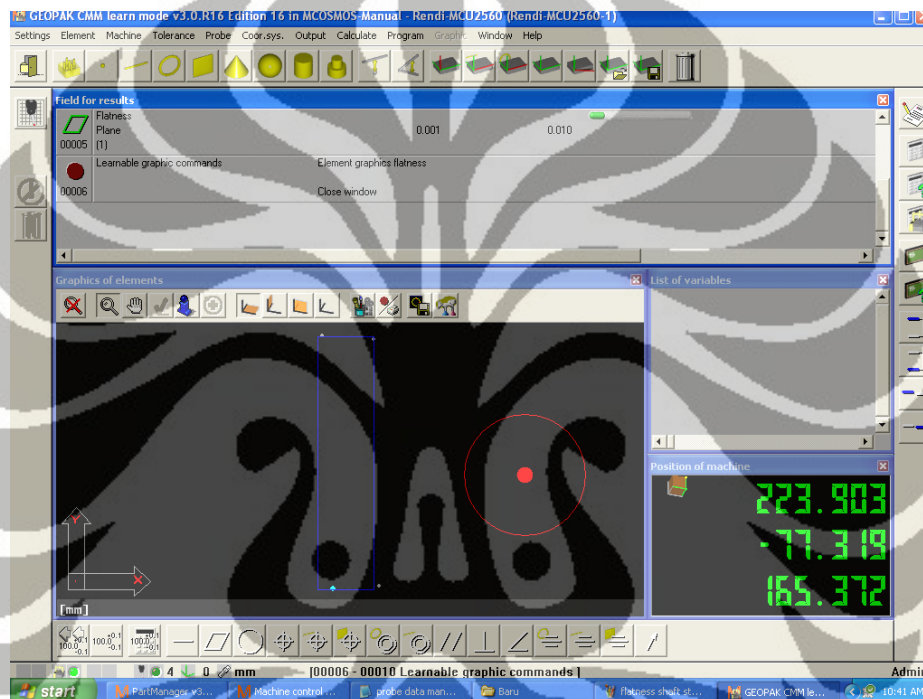
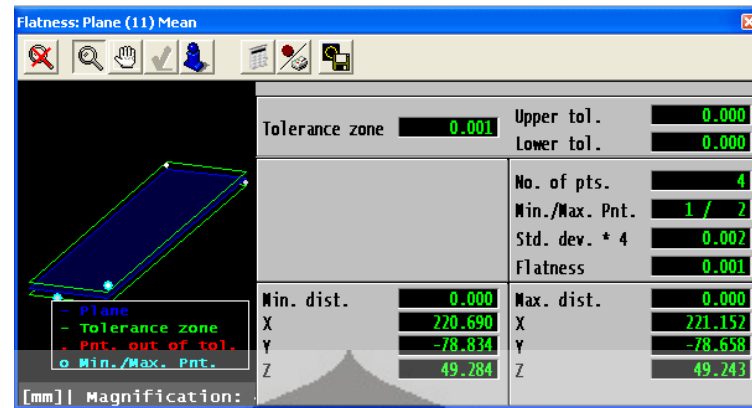
Gambar 6.14 Metode pengukuran elemen plane

- 13) Kemudian akan muncul tampilan sebagai berikut, dan proses pengukuran dilakukan dengan mengukur 4 titik sebagai pembentuk bidang datar.




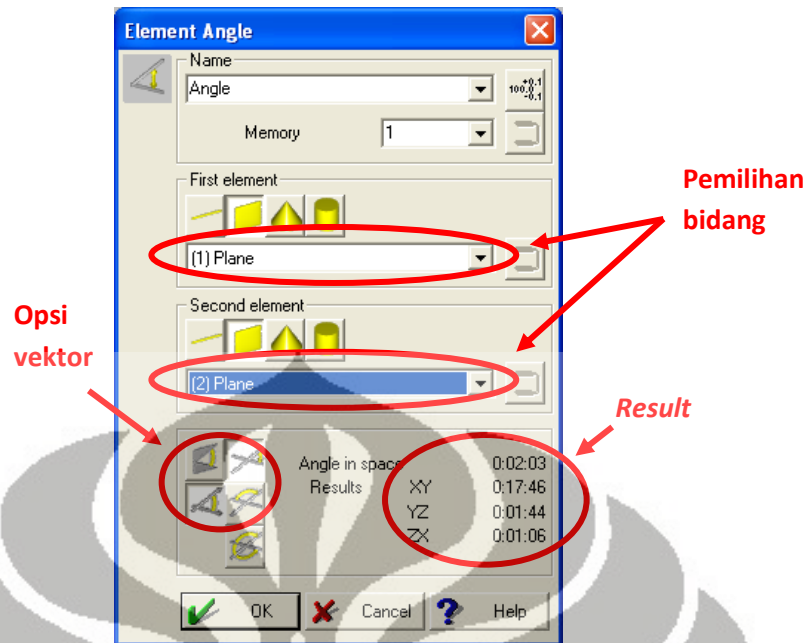
Gambar 6.15 Tampilan pada saat proses pengukuran

- 14) Setelah didapatkan 4 titik, maka hasil yang diperoleh adalah sebagai berikut. Karena menggunakan toleransi 0.001 maka didapatkan hasilnya seperti berikut. Jika tidak memakai toleransi maka tampilan untuk toleransi tidak akan keluar.



Gambar 6.16 Tampilan setelah diambil 4 titik

- 15) Setelah didapatkan 1 bidang maka dilakukan pengukuran dengan cara yang sama dari langkah 11 sampai 14, dengan memutar 1 putaran motor stepper terlebih dahulu hingga didapatkan pengukuran 10 data bidang datar.
- 16) Setelah didapatkan 10 data bidang datar, kemudian kita mengukur perbedaan sudut pada poros bidang datar. Untuk mengukur perbedaan sudut, dipilih *button* . Kemudian pilih bidang yang akan diukur perbedaannya, pilih mode perbedaan sudut vektor. Kemudian hasilnya secara otomatis sudah keluar. Hasil perbedaan sudut yang keluar memiliki satuan jam : menit : detik, bukan dalam satuan derajat pada umumnya.



Gambar 6.17 Tampilan elemen *angle*

17) Perbedaan sudut yang diukur adalah perbedaan antara bidang – bidang sebagai berikut :

- a. bidang 1 dan 2
- b. bidang 2 dan 3
- c. bidang 3 dan 4
- d. bidang 4 dan 5
- e. bidang 5 dan 6
- f. bidang 6 dan 7
- g. bidang 7 dan 8
- h. bidang 8 dan 9
- i. bidang 9 dan 10
- j. bidang 1 dan 10

18) Untuk print out ke dalam format data 'pdf', pilih opsi Output pada *toolbar* > pilih *protocol output* > kemudian pilih tipe protocol data yang diinginkan, protocol data standard dari Mitutoyo juga sudah memenuhi > kemudian pilih *browse* tempat file pada *drive* yang akan disimpan > kemudian 'OK'.

19) Pengukuran diulang dengan langkah yang sama dengan mengganti parameter kecepatan motor dimulay dari delay_us 1000, 800, 600, dan 400.

6.4 Hasil pengujian

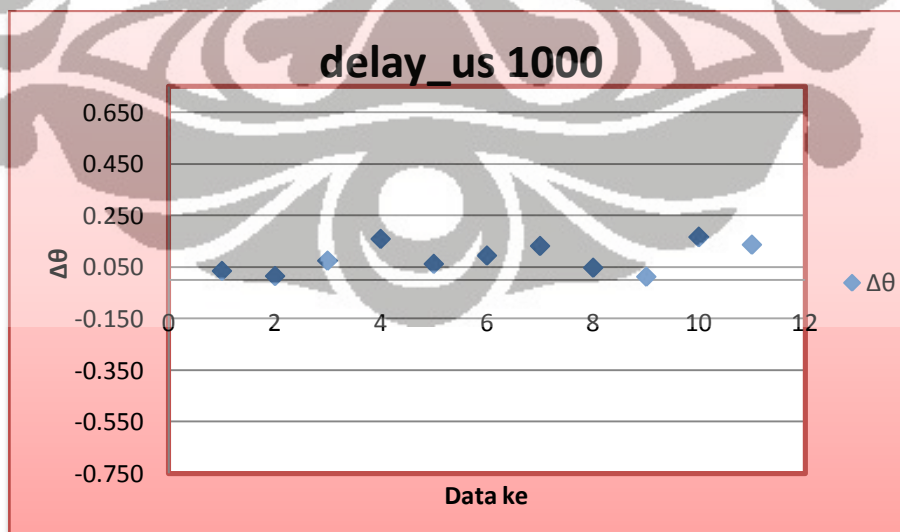
Hasil pengujian yang dihasilkan adalah perbedaan sudut antara bidang yang diukur sesudah dengan sebelumnya. Satuan sudut dalam satuan menit, 1 derajat = 60 menit, 1 menit = 60 detik.

6.4.1 Delay_us 1000 satu putaran

Tabel 6.1 Hasil pengukuran dengan delay_us 1000

No	$\Delta\theta$ (menit : detik)	$\Delta\theta$ (degree)	error(%)
1	0:02:03	0.034	3.417
2	0:00:55	0.015	1.528
3	0:04:32	0.076	7.556
4	0:09:35	0.160	15.972
5	0:03:42	0.062	6.167
6	0:05:40	0.094	9.444
7	0:07:53	0.131	13.139
8	0:02:53	0.048	4.806
9	0:00:45	0.013	1.250
10	0:09:55	0.165	16.528
11	0:08:14	0.137	13.722
Average =			<u>8.503</u>

ctt : 1 putaran, delay_us 1000



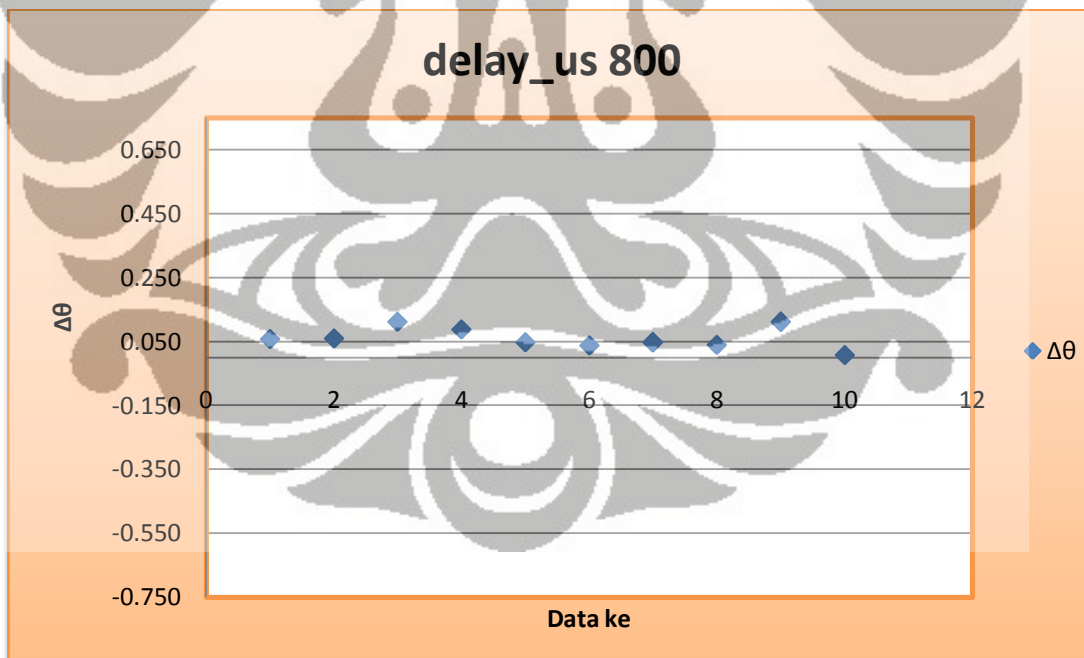
Gambar 6.18 Grafik data perbedaan sudut delay_us 1000

6.4.2 Delay_us 800 satu putaran

Tabel 6.2 Hasil pengukuran dengan delay_us 800

No	$\Delta\theta$ (menit : detik)	$\Delta\theta$ (degree)	error(%)
1	0:03:28	0.058	5.778
2	0:03:35	0.060	5.972
3	0:06:44	0.112	11.222
4	0:05:13	0.087	8.694
5	0:02:49	0.047	4.694
6	0:02:18	0.038	3.833
7	0:02:48	0.047	4.667
8	0:02:22	0.039	3.944
9	0:06:39	0.111	11.083
10	0:00:28	0.008	0.778
Average =			6.067

ctt : 1 putaran, delay_us 800



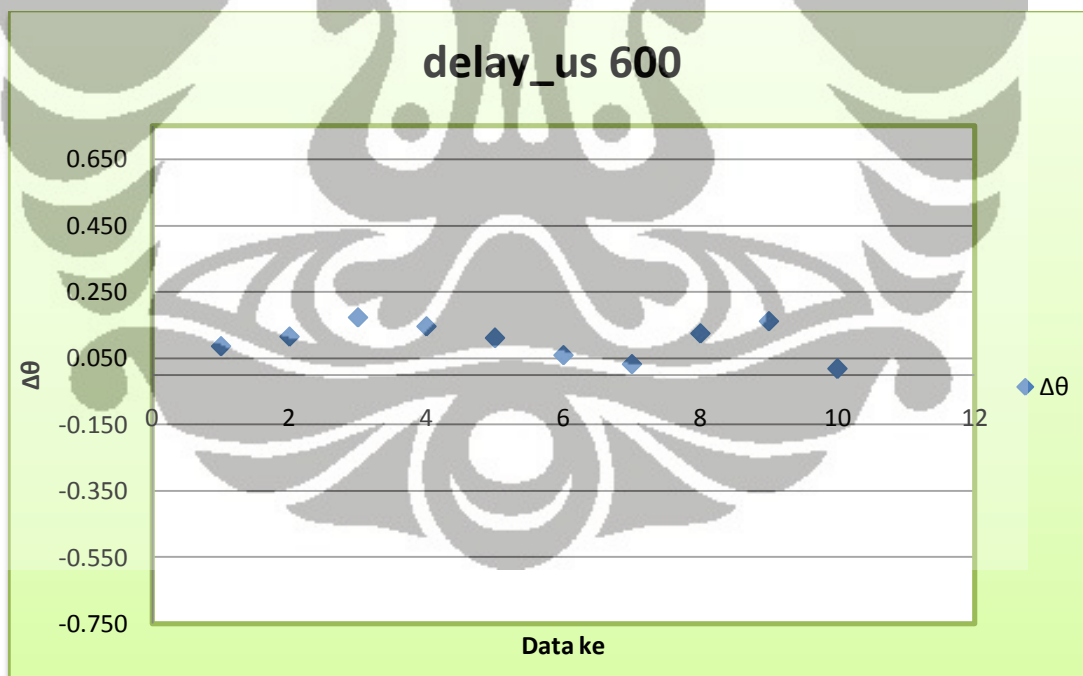
Gambar 6.19 Grafik data perbedaan sudut delay_us 800

6.4.3 Delay_us 600 satu putaran

Tabel 6.3 Hasil pengukuran dengan delay_us 600

No	$\Delta\theta$ (menit : detik)	$\Delta\theta$ (degree)	error(%)
1	0:05:11	0.086	8.639
2	0:06:57	0.116	11.583
3	0:10:25	0.174	17.361
4	0:08:46	0.146	14.611
5	0:06:38	0.111	11.056
6	0:03:32	0.059	5.889
7	0:01:54	0.032	3.167
8	0:07:31	0.125	12.528
9	0:09:39	0.161	16.083
10	0:01:04	0.018	1.778
Average =			<u>10.269</u>

ctt : 1 putaran, delay_us 600



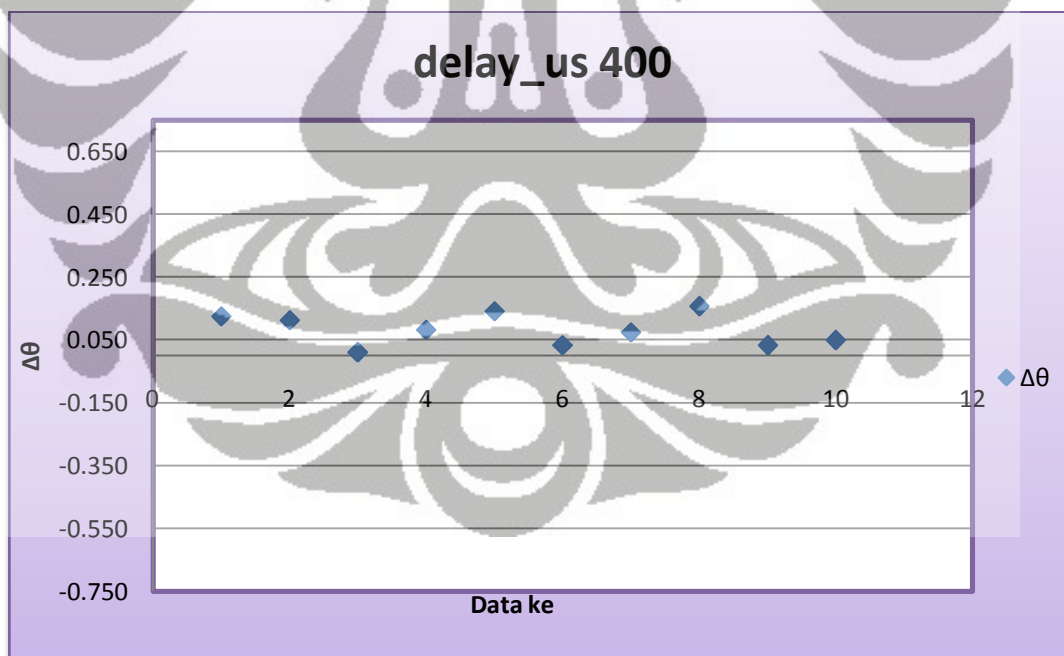
Gambar 6.20 Grafik data perbedaan sudut delay_us 600

6.4.4 Delay_us 400 satu putaran

Tabel 6.4 Hasil pengukuran dengan delay_us 400

No	$\Delta\theta$ (menit : detik)	$\Delta\theta$ (degree)	error(%)
1	0:07:32	0.126	12.556
2	0:06:43	0.112	11.194
3	0:00:36	0.010	1.000
4	0:04:56	0.082	8.222
5	0:08:31	0.142	14.194
6	0:01:58	0.033	3.278
7	0:04:27	0.074	7.417
8	0:09:23	0.156	15.639
9	0:01:56	0.032	3.222
10	0:02:55	0.049	4.861
Average =			8.158

ctt : 1 putaran, delay_us 400



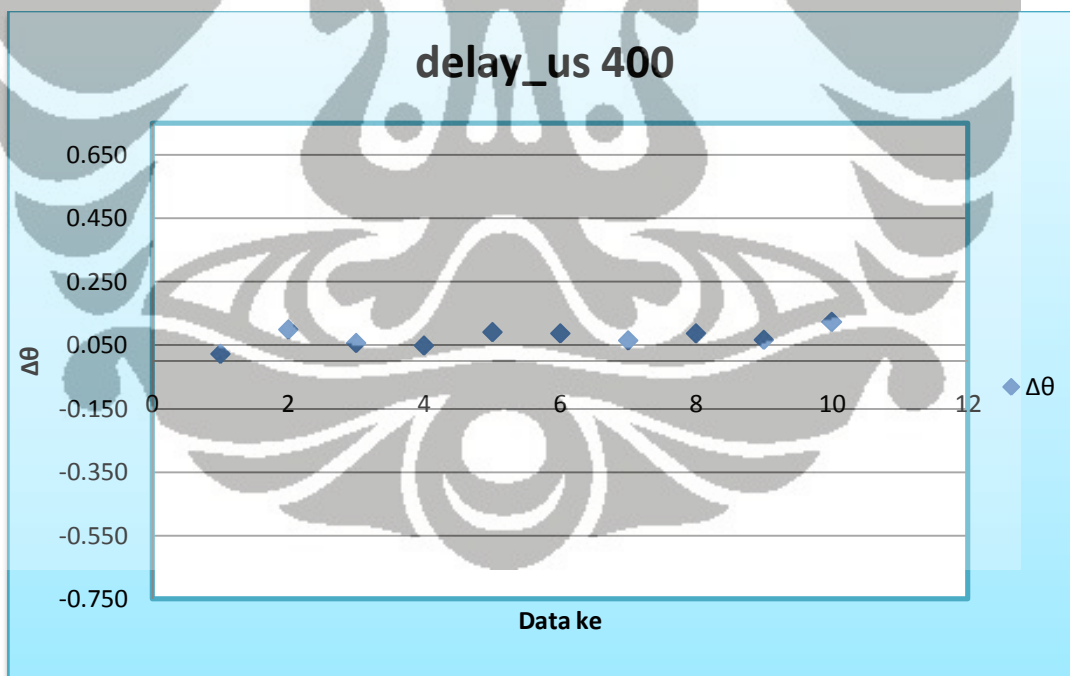
Gambar 6.21 Grafik data perbedaan sudut delay_us 400

6.4.5 Delay_us 400 lima puluh putaran

Tabel 6.5 Hasil pengukuran dengan delay_us 400

No	$\Delta\theta$ (menit : detik)	$\Delta\theta$ (degree)	error(%)
1	0:01:20	0.022	2.222
2	0:05:58	0.099	9.944
3	0:03:21	0.056	5.583
4	0:02:57	0.049	4.917
5	0:05:28	0.091	9.111
6	0:05:13	0.087	8.694
7	0:03:55	0.065	6.528
8	0:05:14	0.087	8.722
9	0:03:59	0.066	6.639
10	0:07:21	0.123	12.250
Average =			<u>7.461</u>

ctt : 50 putaran, delay_us 400



Gambar 6.22 Grafik data perbedaan sudut delay_us 400 50 putaran

6.5 Analisis hasil

Tabel 6.6 Tabel perbedaan sudut rata-rata

No	Pengujian	$\Delta\theta^0$ rata-rata
1	1 putaran dengan delay_us 1000	0,085
2	1 putaran dengan delay_us 800	0,06
3	1 putaran dengan delay_us 600	0,102
4	1 putaran dengan delay_us 400	0,082
5	50 putaran dengan delay_us 400	0,074

Berdasarkan hasil pengukuran pada bidang datar, dan menghasilkan perbedaan sudut yang cukup kecil dengan rata-rata kurang dari 10 menit. Dari hasil percobaan pengaruh kecepatan transfer data terhadap respon dari driver motor stepper dan hasilnya pun cukup memuaskan dikarenakan sudut yang terukur cukup kecil sekitar 10 menit $< \theta < 0$ menit. Pada percobaan ingin dilihat pengaruh terhadap jumlah pulsa yang lebih banyak 50 kali diberikan, dan menunjukkan tidak menunjukkan error yang sangat besar kurang dari 10%. CMM (*Coordinate Measuring Machine*) merupakan alat ukur yang sangat akurat dan presisi, dan hasil pengukuran tergantung dari proses pengambilan data.

Analisis kecepatan data dengan delay_us 1000 sama atau ekuivalen dengan memberikan data *logic* secara terus menerus selama selang waktu $1000\mu s = 1ms = 1/1000$ detik. ATmega2560 memiliki kecepatan mengolah data dengan kecepatan 16MHz atau setara dengan $1/16 \times 10^{-6}$ detik. Dengan kecepatan 1ms, mikrokontroler mengirim data sebanyak 500 pulsa ke driver motor stepper, dan hasil yang diperoleh dengan penyimpangan sudut rata-rata sebesar $0,085^0$. Hasil yang diperoleh menunjukkan bahwa simpangan sudut lebih kecil dari $0,72^0/step$, sehingga dapat disimpulkan data *logic* yang diterima dengan kecepatan 1ms masih dapat diproses dengan baik oleh driver motor stepper dan tidak lebih dari satu step.

Pada kecepatan menggunakan delay_us 800, penyimpangan sudut rata-rata yang terjadi sebesar $0,06^0$ dan ini masih kurang atau lebih kecil dari $0,72^0$, hasilnya kurang lebih sama dengan menggunakan delay_us 1000. Dengan menggunakan delay_us 600,

penyimpangan sudut rata-rata yang terjadi sebesar $0,102^{\circ} < 0,72^{\circ}$, hasilnya kurang lebih sama dengan menggunakan delay_us 800 dan 1000. Kecepatan transfer data *logic* pada mikrokontroller dipercepat menjadi 400 sehingga didapatkan hasil penyimpangan sudutnya sebesar $0,082^{\circ} < 0,72^{\circ}$.

Pengujian berikutnya adalah dengan menambahkan jumlah pulsa yang diberikan 50 kali lebih banyak dari sebelumnya dan menggunakan delay_us 400. Hasil yang diperoleh menunjukkan hasil yang kurang lebih sama dengan hasil pada pengujian dengan menggunakan 500 pulsa. Penyimpangan sudut yang diperoleh didapatkan $0,074^{\circ} < 0,72^{\circ}$, dan tidak melebihi 1 step, jika melebihi 1 step berarti ada error pada sistem kontrol.

Berdasarkan hasil pengujian, dengan pengiriman kecepatan data *logic* sebesar delay_us 1000, 800, 600, 400 dapat disimpulkan bahwa error penyimpangan sudut rata-rata yang terjadi sebesar $0,1^{\circ} < \Delta\theta < 0^{\circ}$. Pada jangkauan error tersebut, dapat dianalisa bahwa error terjadi dapat disebabkan karena adanya gaya momen yang terjadi pada *rotor* motor stepper saat berotasi tiba-tiba mendadak dihentikan.

BAB 7

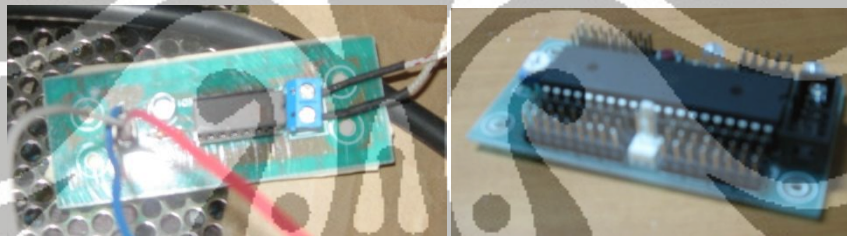
PENGUJIAN DAN ANALISIS SISTEM KONTROL

TERMOKOPEL *AMPLIFIER*

7.1 Tujuan pengujian

Pengujian dilakukan untuk mengetahui karakteristik dari perangkat keras sistem kontrol yaitu temperatur kontrol (termokopel *amplifier*). Seberapa pasti alat ukur dapat mengukur temperatur sehingga dapat dipercaya hasil pengukuran tersebut.

7.2 Peralatan pengujian



(1)

(2)

Gambar 7.1 Perlengkapan

(1) Termokopel *Amplifier*, AD595

(2) Mikrokontroler ATmega 16

Peralatan uji yang digunakan selama proses pengujian adalah sebagai berikut :

1. Mikrokontroler ATmega16.
2. Termokopel *Amplifier*, AD595.
3. Termokopel tipe-K.
4. Modul LCD.
5. *Power supply* 5V dan 24V.
6. Multitester, *voltmeter*.

7. *Temperature Controller* TZN4S (Autonics).

7.3 Metode pengujian

Ada beberapa metode pengujian yang dilakukan, yaitu melakukan pengukuran pada temperatur ruang, temperatur 250⁰C, kelinieran dan kalibrasi sistem.

7.3.1 Temperatur ruang

Metode pengujian yang dilakukan adalah sebagai berikut :

1. Siapkan peralatan yang akan diuji, AD595 dan ATmega16.
2. Letakkan termokopel tipe-K pada ruangan, untuk mengukur temperatur ruang.
3. Menghidupkan sistem, dengan memberikan daya 5V.
4. Mencatat temperatur yang muncul pada LCD, sampai 50 data.

7.3.2 Temperatur 250⁰C

Metode pengujian yang dilakukan adalah sebagai berikut :

1. Siapkan peralatan yang akan diuji, AD595 dan ATmega16.
2. Letakkan termokopel tipe-K pada koil *heater*.
3. Naikkan temperatur sampai 250⁰C, mengubah nilainya pada Codevision, kemudian *compile* program ke mikrokontroler ATmega 16.
4. Menghidupkan sistem dengan memberikan daya 5V.
5. Menghidupkan *heater* dengan memberikan daya 24V.
6. Mencatat temperatur dari awal, sampai temperatur mencapai 250⁰C dan terlihat konstan.

7.3.4 Kelinieran

Metode pengujian yang dilakukan adalah sebagai berikut :

1. Siapkan peralatan yang akan diuji, AD595 dan ATmega16.
2. Letakkan termokopel tipe-K pada koil *heater*.
3. Naikkan temperatur sampai maksimum kemampuan heater 400⁰C, mengubah nilainya pada Codevision, kemudian *compile* program ke mikrokontroller ATmega 16.
4. Menghidupkan sistem dengan memberikan daya 5V.
5. Menghidupkan *heater* dengan memberikan daya 24V.
6. Mencatat temperatur dan tegangan yang terjadi pada termokopel amplifier dari temperatur menunjukkan 100⁰C, sampai temperatur mencapai 400⁰C.
7. Kemudian, matikan *heater* dengan memutuskan daya 24V.
8. Kemudian mencatat temperatur dan tegangan yang terjadi pada termokopel amplifier, sampai temperatur turun.

7.3.3 Kalibrasi

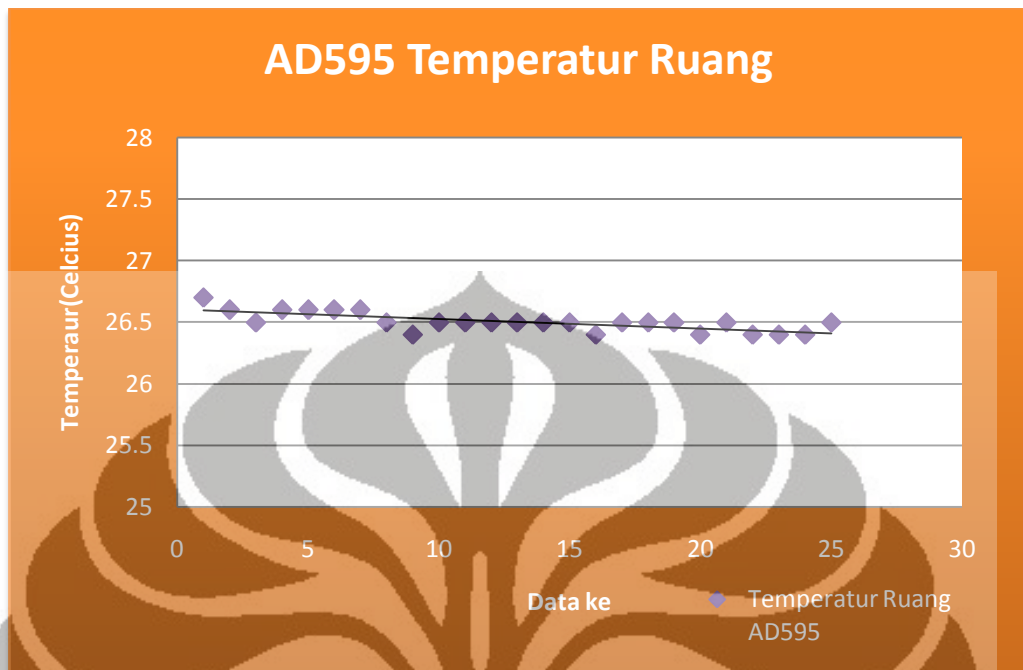
Metode kalibrasi yang dilakukan adalah sebagai berikut :

1. Siapkan peralatan yang akan diuji, AD595, ATmega16 dan TZN4S.
2. Letakkan termokopel tipe-K yang dipasang pada AD595 dan TZN4S pada permukaan luar *heater* ekstuder.
3. Naikkan temperatur sampai 50⁰C, mengubah nilainya pada Codevision, kemudian *compile* program ke mikrokontroller ATmega 16.
4. Menghidupkan sistem dengan memberikan daya 5V.
5. Menghidupkan *heater* dengan memberikan daya 24V.
6. Mencatat temperatur dari awal, sampai temperatur mencapai 50⁰C.
7. Bandingkan pengukuran temperature dengan AD595 dengan alat ukur temperatur TZN4S.

7.4 Grafik hasil pengukuran

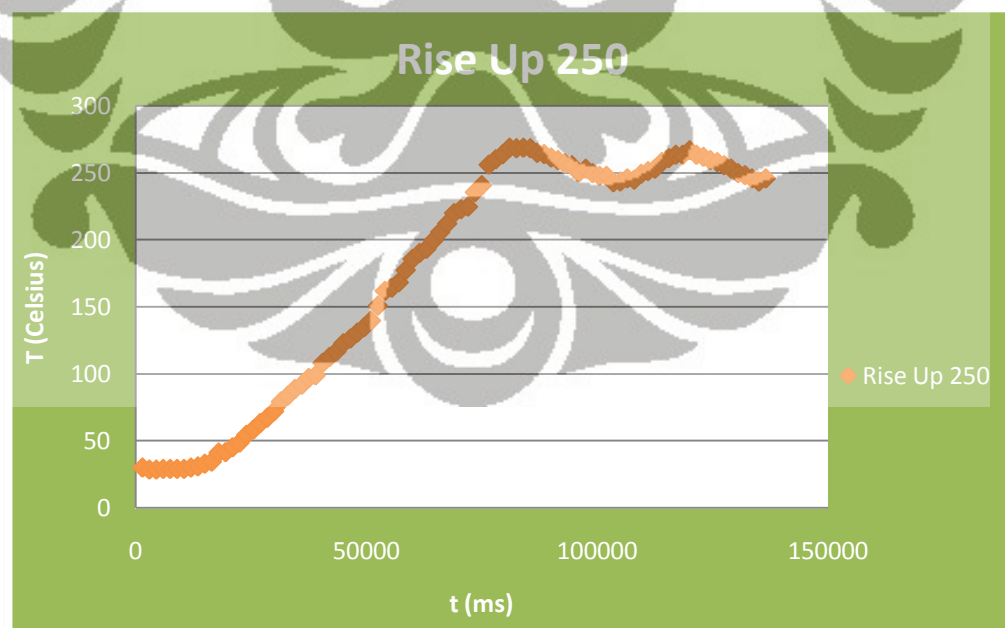
Grafik-grafik dibawah ini adalah hasil dari pengukuran, hasilnya adalah sebagai berikut :

7.4.1 Grafik hasil pengukuran pada temperatur ruang



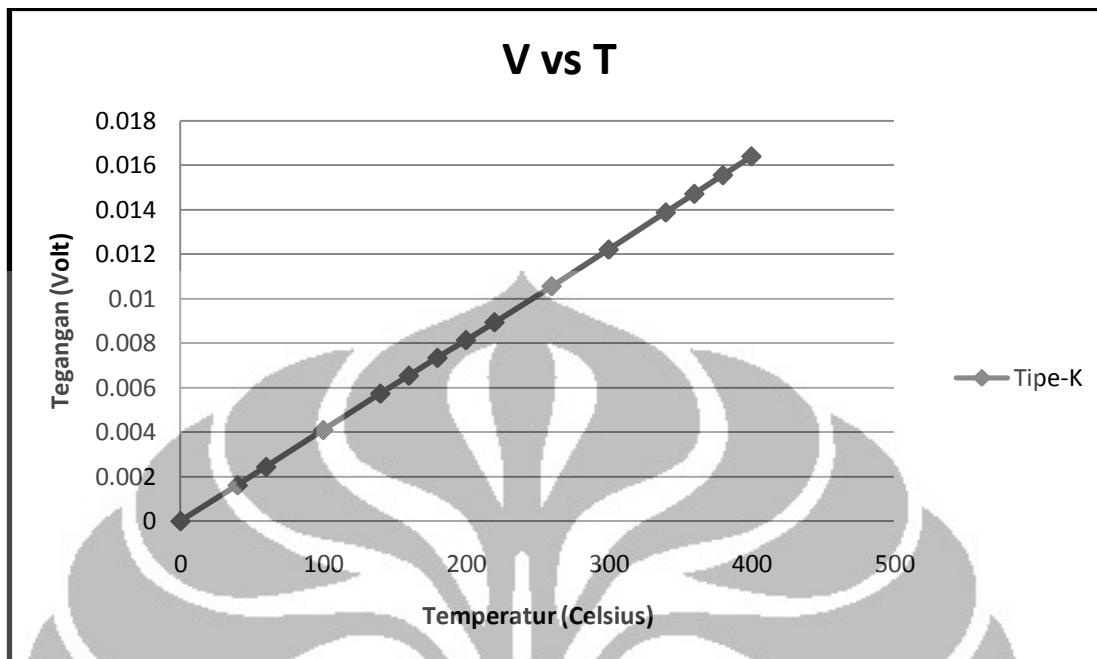
Gambar 7.2 Pengukuran temperatur ruang dengan AD595

7.4.2 Grafik hasil pengukuran temperatur 250°C

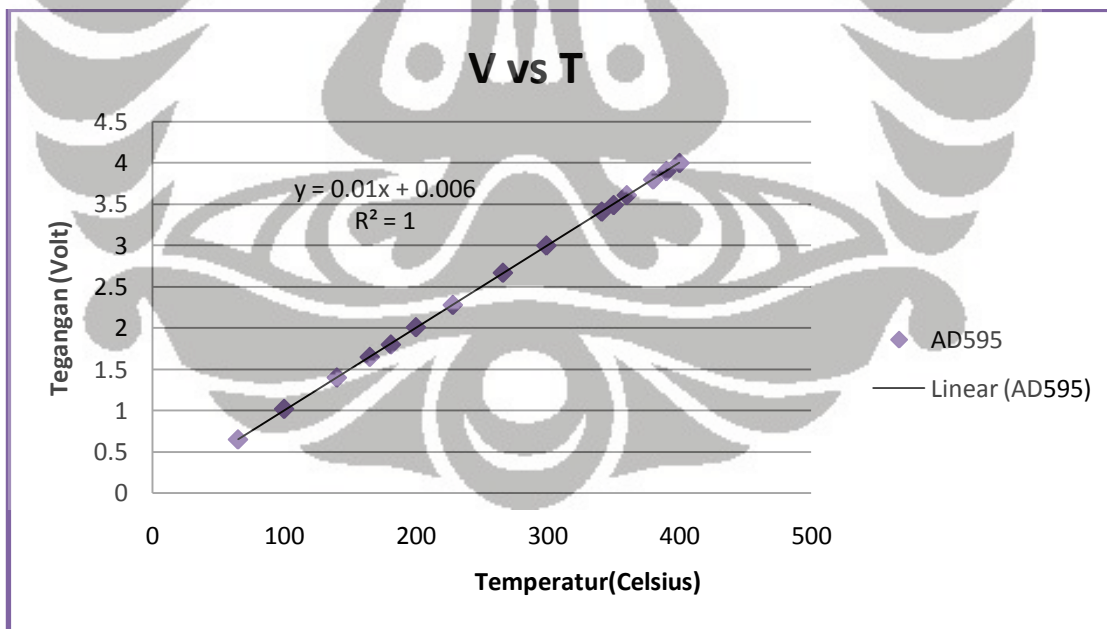


Gambar 7.3 Pengukuran temperatur sampai 250°C

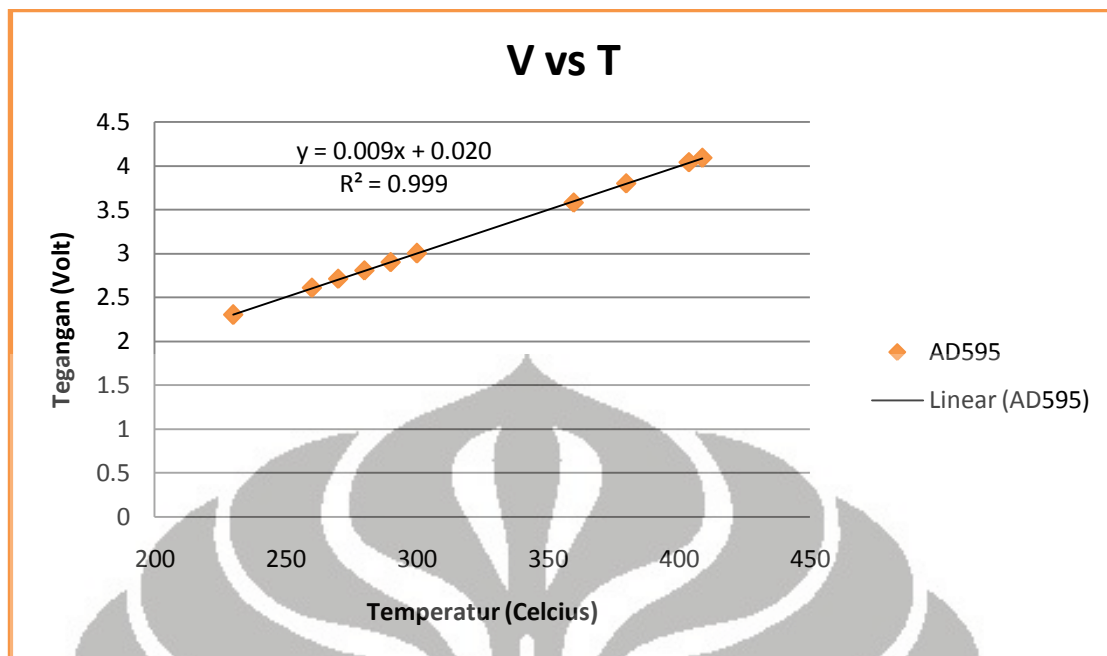
7.4.3 Grafik hasil pengukuran kelinieran



Gambar 7.4 V vs T pada termokopel tipe-K [28]

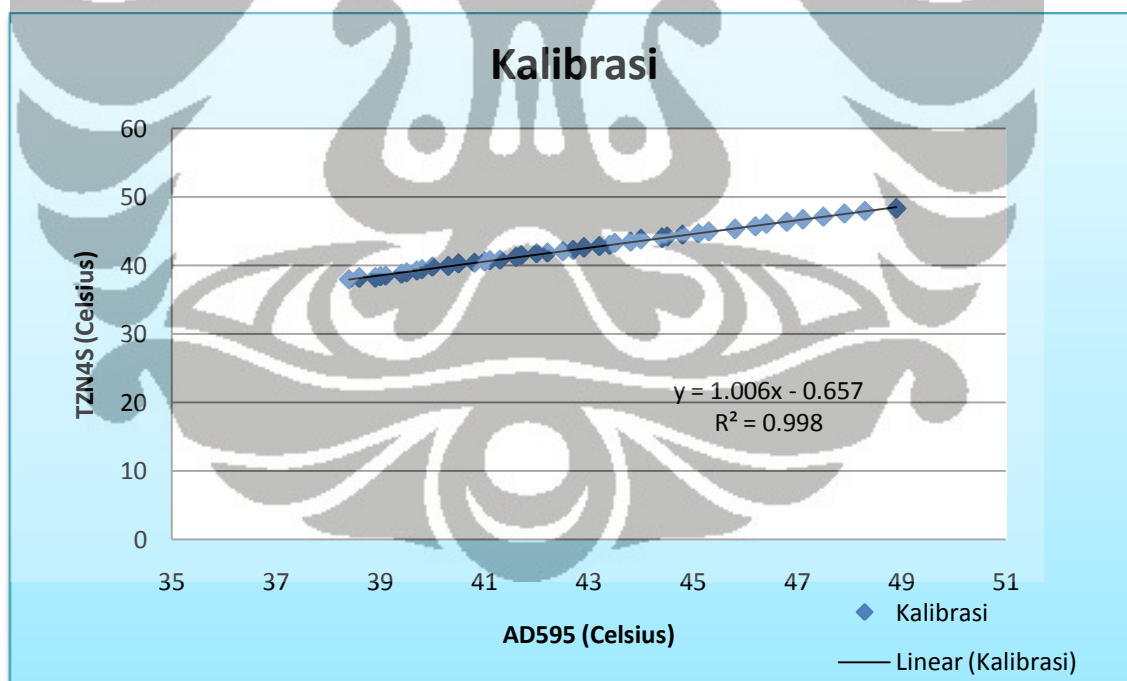


Gambar 7.5 V vs T diukur saat temperatur naik



Gambar 7.6 V vs T diukur saat temperatur turun

7.4.4 Grafik hasil kalibrasi dengan TZN4S



Gambar 7.7 Kalibrasi dengan TZN4S

7.5 Analisis hasil grafik

7.5.1 Analisis grafik pada temperatur ruang

Temperatur diukur pada kondisi ruang dengan menggunakan termokopel *amplifier* AD595. Dari *datasheet* AD595 [28], keakurasian pengukuran adalah $+3/-3^{\circ}\text{C}$. Berdasarkan pengukuran yang dilakukan, pada gambar 7.2, nilai minimum yang dihasilkan dari proses pengukuran adalah 26.4°C , sedangkan nilai maksimum yang diperoleh adalah 26.7°C , dan nilai rata-rata yang diperoleh adalah 26.5°C .

Berdasarkan dari hasil pengukuran menggunakan AD595, temperatur yang diperoleh sudah relatif stabil dan relatif konstan. Penurunan maupun kenaikan nilai temperatur sebesar 0.1°C , kemungkinan terjadinya perubahan temperatur pada kondisi tersebut karena adanya aliran udara bebas pada ruangan. Selama proses pengujian ujung sensor termokopel tidak diisolasi sehingga memungkinkan pengukuran temperatur pada kondisi tersebut dapat berubah-ubah. Nilai perubahan sebesar $\pm 0,1^{\circ}\text{C}$, masih didalam toleransi keakurasian AD595.

7.5.2 Analisis grafik pada temperatur 250°C

Pada pengujian AD595, mengukur kenaikan temperatur pada *heater* mencapai 250°C . Pada pengujian, *heater* yang digunakan adalah kawat nikrom dengan panjang 1m. Kenaikan temperatur terhadap waktu yang terukur dengan AD595 sudah mencapai linier. Pada kondisi temperatur 250°C sudah dicapai, ATmega 16 memberikan sinyal ke *relay*, sehingga arus terputus. Pada kondisi tersebut, temperatur turun hingga lebih rendah dari 250°C , kemudian arus terhubung kembali dan naik kembali mencapai lebih dari 250°C , begitu seterusnya. Maka untuk mencapai temperatur konstan 250°C tidak sepenuhnya konstan, temperatur maksimum adalah 265°C dan temperatur minimum yang dicapai adalah 240°C . Sehingga nilai temperatur pada kondisi menggunakan mekanisme *relay* toleransi-nya adalah $+15^{\circ}\text{C}$ dan -10°C . Sistem kontrol temperatur dengan menggunakan termokopel AD595, dapat mengukur jangkauan temperatur dari 0°C sampai 500°C .

7.5.3 Analisis grafik kelinieran

Pengujian ini, menunjukkan hubungan antara nilai temperatur dengan nilai tegangan yang terukur pada keluaran AD595, apakah sudah mencapai linier. Berdasarkan hasil pengukuran pada grafik gambar 7.5 dan 7.6 nilai R^2 (COD / *Coefficient of Determination*) mendekati nilai 1. Dapat disimpulkan bahwa temperatur yang terukur dengan tegangan keluaran AD595 sudah mencapai linier. Pengukuran dilakukan dari nilai temperatur 65°C sampai temperatur 400°C .

Pada ADC (*Analog Digital Converter*) 10 bit ($2^{10} = 1024$), tegangan referensi yang digunakan adalah 5V, sehingga tegangan masukan pada ADC sebesar 3V akan dikonversi menjadi data *digital* yaitu 600. Nilai temperatur adalah angka digital tersebut yang telah dibagi angka 2,048 kemudian ditampilkan ke dalam LCD. AD595 adalah termokopel *amplifier* linier yang mampu menguatkan tegangan setiap kenaikan 10mV per 1°C Celsius.

7.5.4 Analisis grafik kalibrasi

Kalibrasi merupakan hal yang penting dalam proses pengukuran, karena adanya ketidakpastian didalam setiap proses pengukuran. Oleh karena itu, untuk mendapatkan kebenaran hasil pengukuran yang dihasilkan dari termokopel *amplifier*, maka hasil dari mekanisme sistem tersebut dibandingkan hasilnya dengan alat ukur TZN4S. Pada AD595, panjang termokopel yang digunakan adalah 36cm dengan diameter 0,5 mm, sedangkan pada TZN4S menggunakan termokopel dengan panjang 34cm dengan diameter yang sama.

Berdasarkan hasil grafik pada gambar 7.7, kenaikan temperatur AD595 sangat responsif dibandingkan dengan TZN4S, sehingga grafik membentuk garis linier dengan gradien 1,006. Persamaan garis regresi pada proses kalibrasi $y=1,006x - 0,657$ dengan nilai COD = 0,998 tersebut dimasukkan kedalam program ATmega 16 sebagai hasil temperatur keluaran setelah proses kalibrasi. Perbedaan temperatur sebelum dikalibrasi rata-rata menunjukkan nilai sebesar 0,6 dan menunjukkan perbedaan temperatur yang tidak konstan. Hasil proses kalibrasi ini, juga berpengaruh dari panjangnya termokopel yang digunakan.

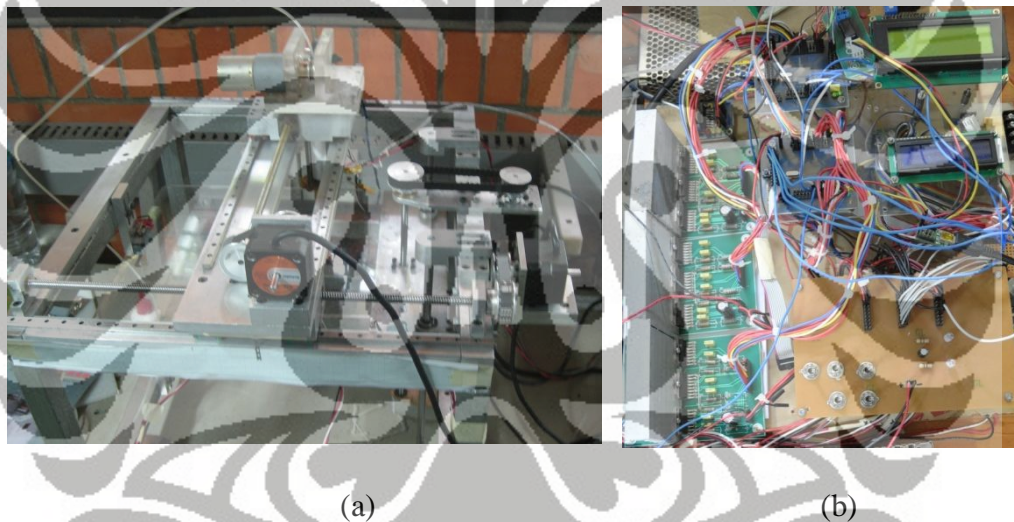
BAB 8

PENGUJIAN DAN ANALISIS PRODUK AKHIR

8.1 Tujuan pengujian

Pengujian dilakukan untuk mengetahui karakteristik suatu produk berbahan dasar material *Nylon PA6* dengan mengubah parameter yang mempengaruhi seperti ketinggian *bed*, kecepatan *feeding*, dan temperatur.

8.2 Peralatan pengujian



Gambar 8.1 (a) Mesin RPFDM, (b) Sistem kontrol

Peralatan uji yang digunakan selama proses pengujian adalah sebagai berikut :

- 1) Satu buah mesin *rapid prototyping*, mesin RPFDM.
- 2) Perangkat lengkap sistem kontrol, terdiri dari mikrokontroler, *amplifier*, termokopel, modul LCD, serial komunikasi dan *power supply*.
- 3) Seperangkat komputer (PC), sebagai menjalankan program dan transfer data melalui *port* USB.
- 4) *Software CodevisionAVR*, sebagai *debugging* program.

- 5) *Software* antar muka, sebagai pengirim data ke sistem kontrol.
- 6) Pencatat waktu, *stopwatch digital*.

8.3 Metode pengujian

Inti dari pengujian ini adalah untuk mendapatkan karakteristik produk yang berkualitas, dengan mengubah parameter-parameter yang mempengaruhi kualitas produk tersebut, seperti ketinggian *bed*, kecepatan *feeding*, temperatur, diameter keluaran *nozzle*, dan jenis material yang digunakan. Untuk parameter diameter keluaran *nozzle*, dan menggunakan jenis material yang berbeda diteliti oleh peneliti lain, yaitu oleh Ahmad Saroni yang satu tim dengan peneliti bersama-sama mengembangkan mesin RPFDM ini.

8.3.1 Uji ketinggian *bed*

Pada pengujian ini, parameter-parameter yang mempengaruhi seperti kecepatan *feeding*, temperatur, diameter keluaran *nozzle*, dan kecepatan linier nut dibuat konstan. Dengan menggunakan parameter-parameter sebagai berikut :

- a) Diameter keluaran *nozzle* = 1 mm.
- b) Temperatur koil pemanas = 350⁰C.
- c) Kecepatan *feeding* dengan menggunakan PWM = 65% duty cycle.
- d) Kecepatan linier *nut* = 1,5 cm/s

Variabel yang didapatkan setelah hasil pengujian :

1. Tebal *layer*

Tebal *layer* secara keseluruhan diukur dihitung dengan alat ukur (sigmat).

Kemudian dibagi banyaknya *layer* yang dibuat sehingga menjadi tebal *layer* rata-rata. *Layer* adalah ketinggian lapisan dalam arah sumbu-Z.

2. Tebal *path*

Tebal *path* atau lintasan adalah ketebalan akhir material yang keluar dari ujung *nozzle*. Material yang keluar dari *nozzle* akan mengalami penyesuaian ketika jatuh pada *bed* dan membentuk prototipe.

8.3.2 Uji kecepatan *feeding*

Pada pengujian ini, parameter-parameter yang mempengaruhi seperti ketinggian *bed*, temperatur, diameter keluaran *nozzle*, dan kecepatan linier *nut* dibuat konstan. Dengan menggunakan parameter-parameter sebagai berikut :

- a) Diameter keluaran *nozzle* = 1 mm.
- b) Temperatur koil pemanas = 350⁰C.
- c) Ketinggian *bed* = 320 step.
- d) Kecepatan linier *nut* = 1,5 cm/s

Variabel yang didapatkan setelah hasil pengujian :

1. Tebal *layer*

Tebal *layer* secara keseluruhan diukur dihitung dengan alat ukur (sigmat). Kemudian dibagi banyaknya *layer* yang dibuat sehingga menjadi tebal *layer* rata-rata. *Layer* adalah ketinggian lapisan dalam arah sumbu-Z.

2. Tebal *path*

Tebal *path* atau lintasan adalah ketebalan akhir material yang keluar dari ujung *nozzle*. Material yang keluar dari *nozzle* akan mengalami penyesuaian ketika jatuh pada *bed* dan membentuk prototipe.

8.3.3 Uji temperatur

Pada pengujian ini, parameter-parameter yang mempengaruhi seperti ketinggian *bed*, kecepatan *feeding*, diameter keluaran *nozzle*, dan kecepatan linier *nut* dibuat konstan. Dengan menggunakan parameter-parameter sebagai berikut :

- a) Diameter keluaran *nozzle* = 1 mm.

- b) Kecepatan *feeding* dengan menggunakan PWM = 65% duty cycle.
- c) Ketinggian *bed* = 320 step.
- d) Kecepatan linier *nut* = 1,5 cm/s

Variabel yang didapatkan setelah hasil pengujian:

1. Tebal *layer*

Tebal *layer* secara keseluruhan diukur dihitung dengan alat ukur (sigmat). Kemudian dibagi banyaknya *layer* yang dibuat sehingga menjadi tebal *layer* rata-rata. *Layer* adalah ketinggian lapisan dalam arah sumbu-Z.

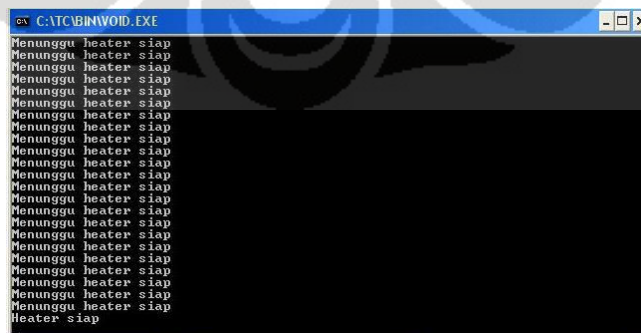
2. Tebal *path*

Tebal *path* atau lintasan adalah ketebalan akhir material yang keluar dari ujung *nozzle*. Material yang keluar dari *nozzle* akan mengalami penyesuaian ketika jatuh pada *bed* dan membentuk prototipe.

8.3.4 Langkah pengujian

Langkah-langkah yang dilakukan dalam pengujian adalah sebagai berikut :

1. Menyiapkan mesin RPFDM, dengan menghidupkan *power supply* 5V, 12V, dan 24V. Kemudian aktifkan *switch* untuk semua sumber daya ke sistem kontrol.
2. Tunggu sampai *heater* memanaskan sampai temperatur pada LCD menunjukkan angka 350⁰C.
3. Buka antar muka perangkat lunak yang telah dikembangkan. Menunggu sampai *heater* siap.



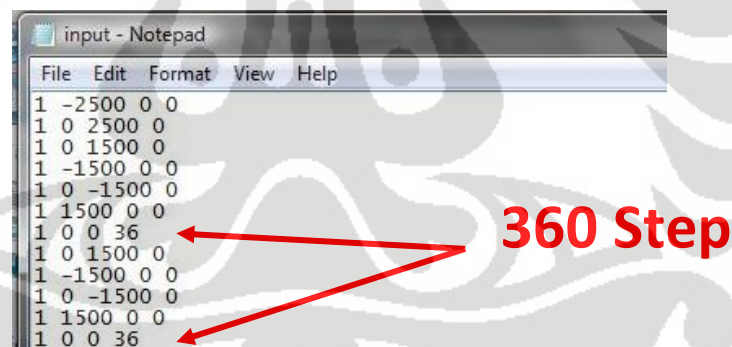
Gambar 8.2 Menunggu *heater* siap

4. Kemudian pilih opsi no. 1 untuk posisi mesin *default*, kemudian pilih opsi no. 2 untuk mengirim data, berikut antar muka perangkat lunak yang telah dibuat.



Gambar 8.3 Opsi-opsi yang ada pada *software interface*

5. Pada pengujian ini, pengujian pertama adalah mengubah parameter ketinggian, dengan mengubah parameter nilai Z pada *notepad*. Nilai Z, dimulai dari nilai 300 step sampai dengan 440 step dengan *increment* 20 step.



Gambar 8.4 Mengubah nilai Z pada *notepad*

6. Pengujian perubahan parameter Z, diulang dari langkah 1 sampai dengan langkah 5.
7. Pengujian kedua adalah merubah parameter kecepatan *feeding* dengan metode PWM, perubahan dilakukan pada perangkat lunak Codevision AVR, kemudian di *flash programming* / di-*compile* ke dalam ATmega 16, 16MHz.

```

80 // Timer/Counter 0 initialization
81 // Clock source: System Clock
82 // Clock value: 250.000 kHz
83 // Mode: Fast PWM top=FFh
84 // OCO output: Non-Inverted PWM
85 TCCR0=0x6B;
86 TCNT0=0x00;
87 OCR0=0x00;
88
89 while (1)
90 {
91 // Place your code here
92 PWM[0]=getchar();
93 while (PWM[0] != 'u')
94 {
95 motor_pwm(0.7,on,off);
96 PWM[0]=getchar();
97 }
98 while(1)
99 {
100 motor_pwm(0,off,off);
101 if (PWM[0] = 'y')
102 {
103 break;
104 }
105 }

```

Parameter kecepatan *feeding* dengan PWM

Gambar 8.5 Merubah parameter kecepatan *feeding* dengan PWM

8. Pengujian ketiga adalah merubah parameter temperatur, dengan merubah nilainya pada perangkat lunak Codevision AVR. Metode ini sama dengan metode ke -7 hanya berbeda program.

```

91 // ADC Voltage Reference: AREF pin
92 // ADC Auto Trigger Sources: Free Running
93 ADMUX=ADC_VREF_TYPE & 0xff;
94 ADCSRA=0xA4;
95 SFIOR&=0x1F;
96
97 // LCD module initialization
98 lcd_init(16);
99
100 while (1)
101 {
102 // Place your code here
103
104 suhu = (read_adc(0)*0.48828125);
105 ftoa (suhu,1,LCD);
106 lcd_clear();
107 lcd_gotoxy(0,0);
108 lcd_puts(LCD);
109 lcd_gotoxy(6,0);
110 lcd_putsf("Celcius");
111 delay_ms(1500);
112 if (suhu >= 250)
113 {
114 PORTB.0 = 1;
115 putchar('c');
116 }
117 else
118 {
119 PORTB.0 = 0;
120 }
121 }
122 };
123 }
124

```

Merubah nilai parameter temperatur

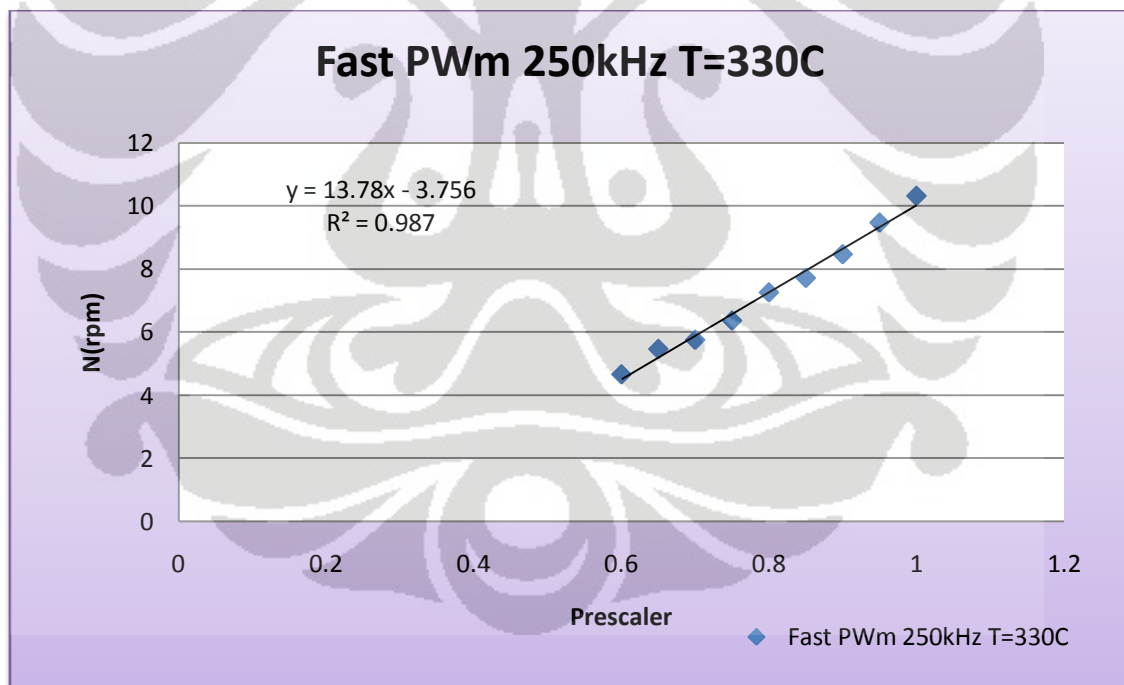
Gambar 8.6 Merubah parameter temperatur

9. Langkah-langkah pengujian perubahan parameter PWM diulang dari langkah 1 sampai langkah 4 dan langkah 7.
10. Langkah-langkah pengujian perubahan parameter temperatur diulang dari langkah 1 sampai langkah 4 dan langkah 8.

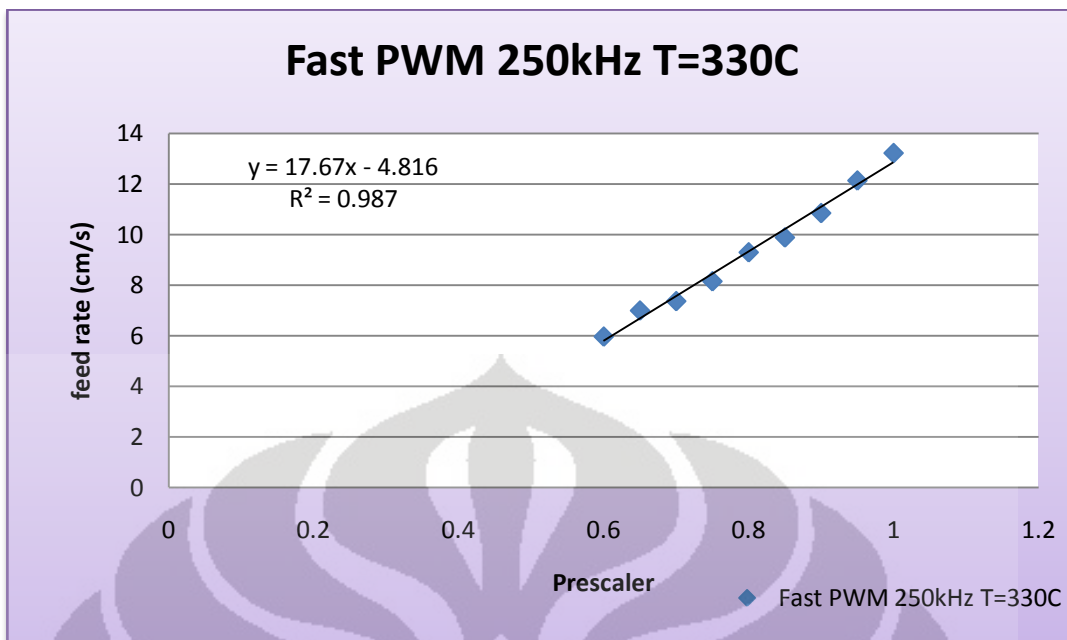
8.4 Hasil grafik pengujian

Hasil dari pengujian berupa hasil grafik *prescaler (duty cycle)* vs putaran motor, dan *prescaler vs feed rate* pada temperatur 330⁰C, 340⁰C, 350⁰C, dan 360⁰C. Pada percobaan ini mencari hubungan kecepatan linier *nut* terhadap lama data *logic* yang dikirim, berupa grafik *delay_us* vs kecepatan linier.

8.4.1 Grafik pengujian pada temperatur 330⁰C

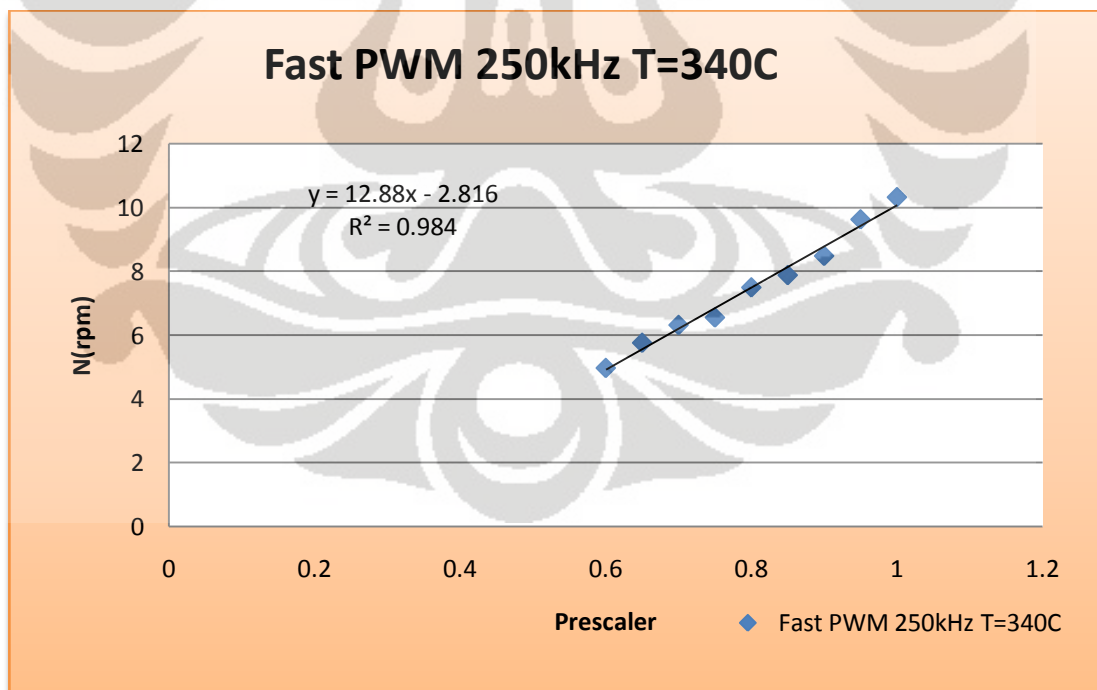


Gambar 8.7 Grafik *prescaler* vs putaran motor pada T = 330⁰C

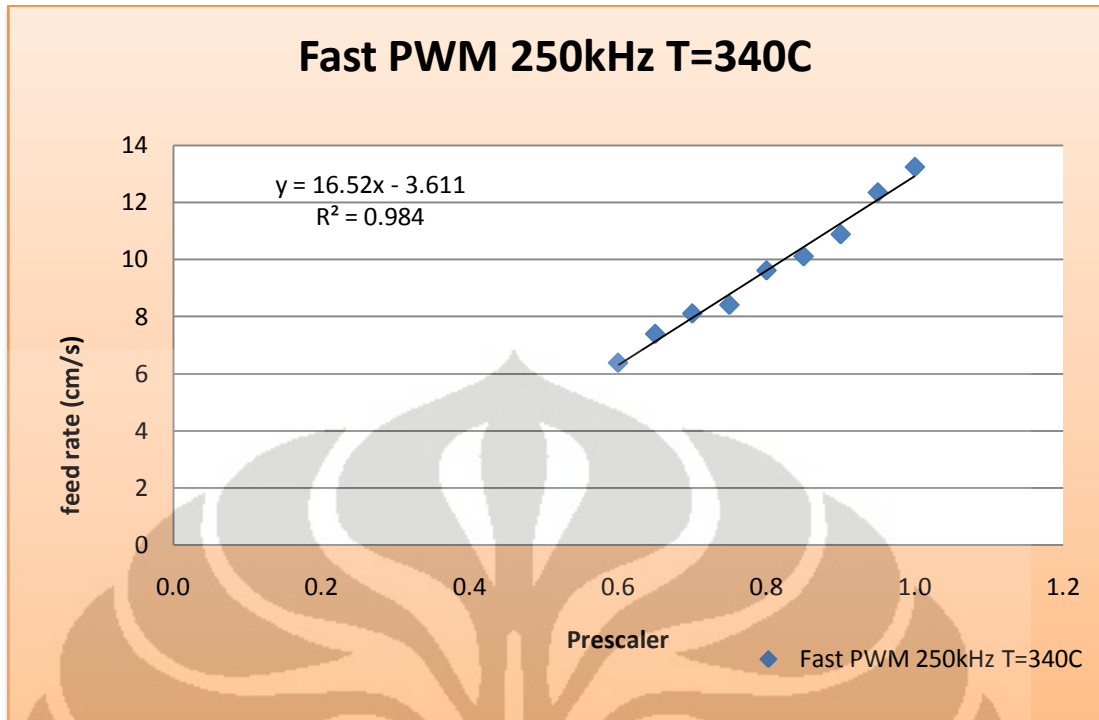


Gambar 8.8 Grafik *prescaler* vs *feed rate* pada $T = 330^{\circ}\text{C}$

8.4.2 Grafik pengujian pada temperatur 340°C

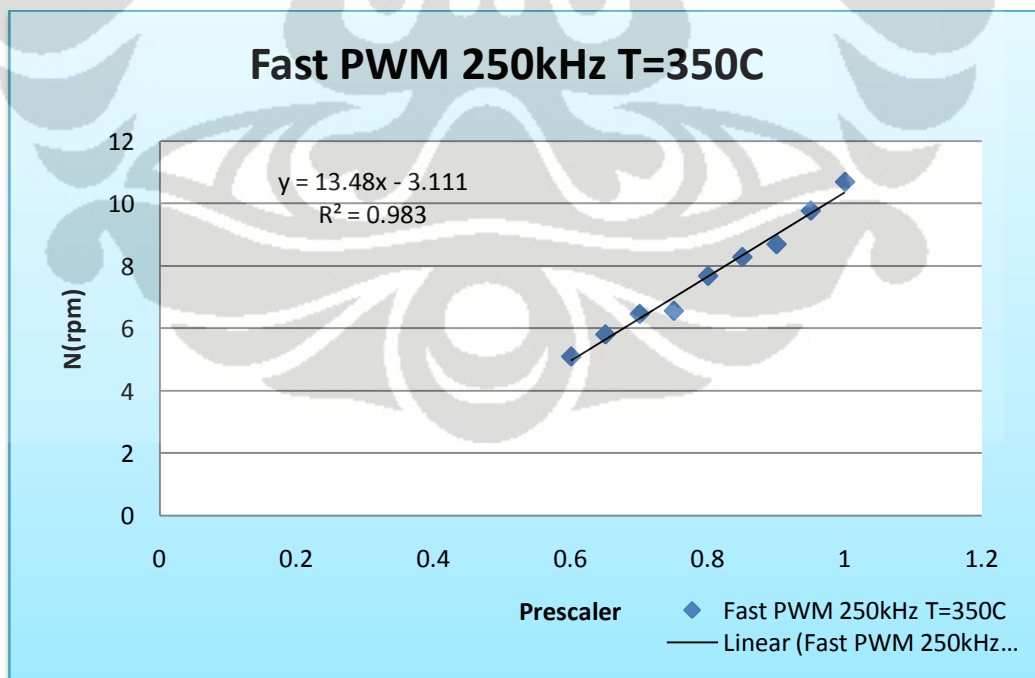


Gambar 8.9 Grafik *prescaler* vs putaran motor pada $T = 340^{\circ}\text{C}$

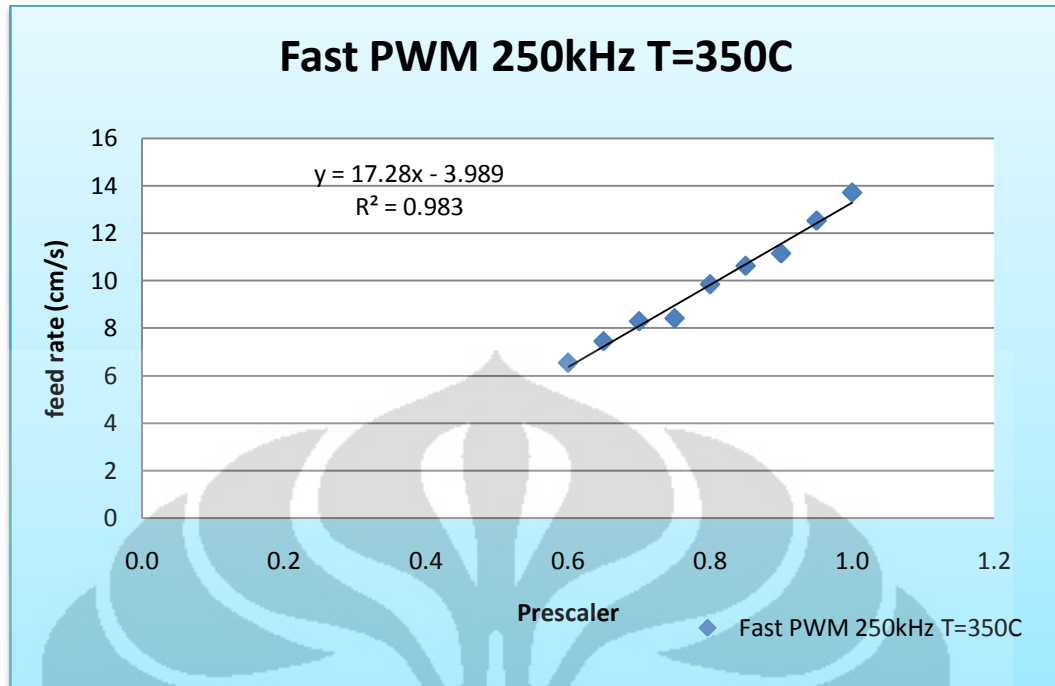


Gambar 8.10 Grafik *prescaler* vs *feed rate* pada $T = 340^{\circ}\text{C}$

8.4.3 Grafik pengujian pada temperatur 350°C

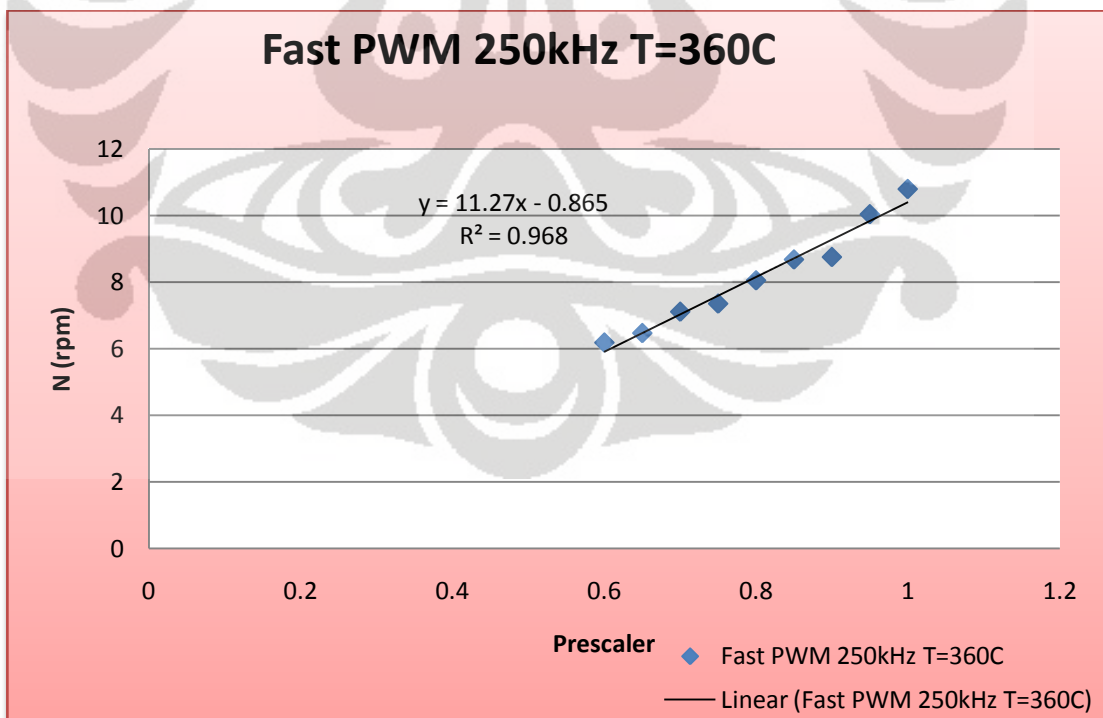


Gambar 8.11 Grafik *prescaler* vs putaran motor pada $T = 350^{\circ}\text{C}$

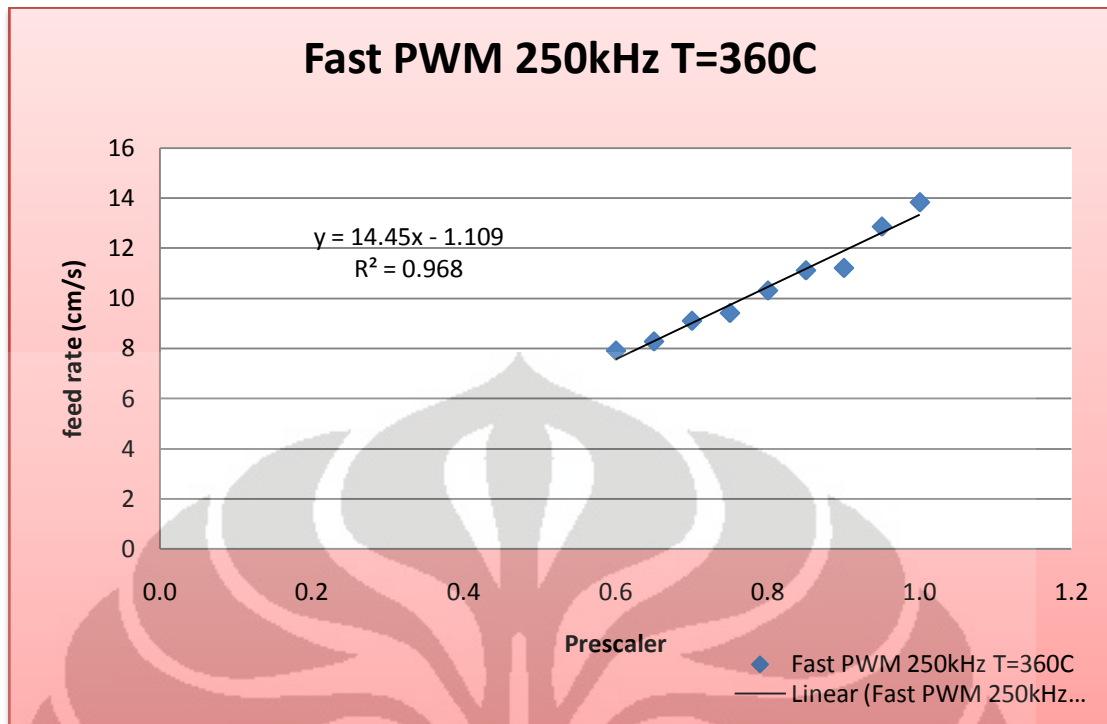


Gambar 8.12 Grafik *prescaler* vs *feed rate* pada $T = 350^{\circ}\text{C}$

8.4.4 Grafik pengujian pada temperatur 360°C



Gambar 8.13 Grafik *prescaler* vs putaran motor pada $T = 360^{\circ}\text{C}$



Gambar 8.14 Grafik *prescaler* vs *feed rate* pada $T = 360^{\circ}\text{C}$

8.4.5 Perhitungan *feed rate*

Dik :

$D_{\text{pulley}} = 20 \text{ mm};$

$D_{\text{in}} = 3,5 \text{ mm};$

$D_{\text{out}} = 1 \text{ mm};$

$N = \text{putaran motor (rpm)};$

$\omega = \text{putaran motor (rad/s)};$

$v_1 = \text{kec. material masuk (cm/s)};$

$v_2 = \text{kec. material keluar (cm/s)};$

$$\omega = \frac{2\pi}{60} \times N ; v_1 = \omega \times R_{\text{pulley}} ; v_2 = v_1 \times \frac{R_{\text{in}}^2}{R_{\text{out}}^2}$$

Tabel 8.1 Perhitungan *feed rate* pada $T = 330^{\circ}\text{C}$

Prescaler	N(rpm)	w(rad/s)	v1(cm/s)	v2(cm/s)
1	10.309	1.079	1.079	13.218
0.95	9.464	0.991	0.991	12.134
0.9	8.463	0.886	0.886	10.851
0.85	7.712	0.807	0.807	9.888
0.8	7.255	0.759	0.759	9.302
0.75	6.363	0.666	0.666	8.158
0.7	5.753	0.602	0.602	7.376
0.65	5.464	0.572	0.572	7.006
0.6	4.662	0.488	0.488	5.977

Tabel 8.2 Perhitungan *feed rate* pada $T = 340^{\circ}\text{C}$

Prescaler	N(rpm)	w(rad/s)	v1(cm/s)	v2(cm/s)
1	10.327	1.081	1.081	13.241
0.95	9.631	1.008	1.008	12.349
0.9	8.487	0.888	0.888	10.882
0.85	7.884	0.825	0.825	10.109
0.8	7.5	0.785	0.785	9.616
0.75	6.557	0.686	0.686	8.407
0.7	6.322	0.662	0.662	8.106
0.65	5.764	0.603	0.603	7.390
0.6	4.975	0.521	0.521	6.379

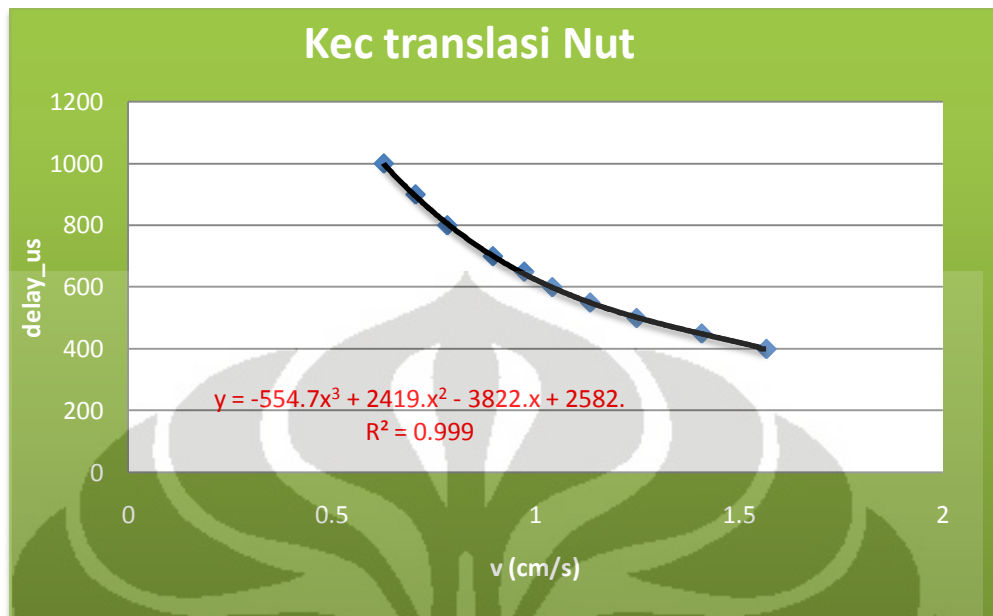
Tabel 8.3 Perhitungan *feed rate* pada $T = 350^{\circ}\text{C}$

Prescaler	N(rpm)	w(rad/s)	v1(cm/s)	v2(cm/s)
1	10.695	1.119	1.119	13.713
0.95	9.772	1.023	1.023	12.529
0.9	8.696	0.910	0.910	11.150
0.85	8.287	0.867	0.867	10.625
0.8	7.682	0.804	0.804	9.850
0.75	6.564	0.687	0.687	8.416
0.7	6.465	0.677	0.677	8.289
0.65	5.808	0.608	0.608	7.447
0.6	5.102	0.534	0.534	6.542

Tabel 8.4 Perhitungan *feed rate* pada $T = 360^{\circ}\text{C}$

Prescaler	N(rpm)	w(rad/s)	v1(cm/s)	v2(cm/s)
1	10.791	1.129	1.129	13.836
0.95	10.033	1.050	1.050	12.864
0.9	8.746	0.915	0.915	11.214
0.85	8.671	0.908	0.908	11.118
0.8	8.043	0.842	0.842	10.312
0.75	7.344	0.769	0.769	9.416
0.7	7.101	0.743	0.743	9.105
0.65	6.459	0.676	0.676	8.282
0.6	6.173	0.646	0.646	7.915

8.4.6 Grafik kecepatan translasi *nut* setiap sumbu



delay_us	v(cm/s)
400	1.567
450	1.408
500	1.248
550	1.134
600	1.041
650	0.972
700	0.895
800	0.783
900	0.705
1000	0.627

Gambar 8.15 Grafik delay_us vs kecepatan linier *nut*

8.5 Analisis grafik

8.5.1 Linierisasi data, deviasi standar dan *coefficient of determination*

Hasil pengujian dapat dipresentasikan dengan menggunakan persamaan garis linier, untuk memperoleh garis linier yang mempresentasikan data-data hasil pengujian dengan keakuratan terbaik digunakan metode *least square*. Persamaan yang digunakan untuk membuat persamaan garis linier tersebut akan dijelaskan sebagai berikut :

Persamaan garis linier dirumuskan sebagai berikut : $y = mx + b$

dimana : y adalah nilai pada ordinat berdasarkan persamaan linier.

Pada pengujian, nilai tersebut berupa putaran motor dan *feed rate*.

x adalah nilai pada absis berdasarkan persamaan linier, pada pengujian, nilai tersebut adalah *prescaler*.

m adalah gradien/kemiringan dari garis linier tersebut.

b adalah nilai \bar{y} yang terpotong oleh garis linier.

maka :

$$m = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sum (x - \bar{x})^2} \text{ dan } b = \bar{y} - m \cdot \bar{x}$$

dimana : x adalah nilai data pada absis yang terukur secara aktual.

y adalah nilai data pada ordinat yang terukur secara aktual.

\bar{x} adalah nilai rata-rata dari data aktual pada absis.

\bar{y} adalah nilai rata-rata dari data aktual pada ordinat.

Nilai deviasi standar menggambarkan besarnya *error* yang dihasilkan oleh persamaan garis linier terhadap nilai aktual yang diperoleh dari hasil pengujian. Berikut persamaan deviasi standar yang digunakan :

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (y_i - \bar{y})^2}{N(N-1)}}$$

$$\bar{y} = \frac{\sum_{i=1}^n y_i}{N}$$

dimana : σ adalah deviasi standar.

i adalah nomer seri data.

n adalah nomer data terakhir.

y_i adalah tipe data ke- i .

\bar{y} adalah rata-rata dari data atau rata-rata aritmatik.

N adalah jumlah data.

Coefficient of Determination (COD) adalah sebuah parameter yang menggambarkan kelinieran suatu sebaran data-data aktual pembentuk persamaan linier. Apabila COD sama dengan 1, maka sebaran data-data aktual tersebut tepat berada di satu garis lurus, sehingga tidak ada perbedaan nilai antara nilai yang diperoleh dari persamaan garis dengan nilai data aktual. Apabila COD adalah 0, hal ini menggambarkan persamaan regresi yang dihasilkan tidak dapat membantu sama sekali untuk memprediksi nilai aktual. Berikut persamaan COD yang digunakan :

$$COD = R^2 = 1 - \frac{\sum_{i=1}^n (y_i - f_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

dimana : R^2 adalah *Coefficient of Determination*.

i adalah nomer seri data.

n adalah nomer data terakhir.

y_i adalah tipe data ke- i .

\bar{y} adalah rata-rata dari data atau rata-rata aritmatik.

f_i adalah nilai data berdasarkan persamaan garis linier yang diperoleh.

8.5.2 Analisis kelinieran

Tabel 8.5 Linierisasi data, deviasi standar, dan *Coefficient of Determination*

No.	Tipe grafik	Persamaan	Deviasi standar	R ²
1	Putaran motor (T=330 ⁰ C)	$y = 13.78x - 3.756$	0.633	0.987
2	<i>Feed rate</i> (T=330 ⁰ C)	$y = 17.67x - 4.816$	0.821	0.987
3	Putaran motor (T=340 ⁰ C)	$y = 12.88x - 2.816$	0.593	0.984
4	<i>Feed rate</i> (T=340 ⁰ C)	$y = 16.52x - 3.611$	0.760	0.984
5	Putaran motor (T=350 ⁰ C)	$y = 13.48x - 3.111$	0.621	0.983
6	<i>Feed rate</i> (T=350 ⁰ C)	$y = 17.28x - 3.989$	0.796	0.983
7	Putaran motor (T=360 ⁰ C)	$y = 11.27x - 0.865$	0.523	0.968
8	<i>Feed rate</i> (T=360 ⁰ C)	$y = 14.45x - 1.109$	0.670	0.968

Berdasarkan nilai *Coefficient of Determination (COD)* yang diperoleh, nilai rata-rata yang diperoleh mendekati 1, dan dapat dikatakan sifat dari perubahan parameter-parameter tersebut bersifat linier. Hubungan antara putaran motor terhadap prescaler maupun kecepatan *feeding* material terhadap prescaler adalah bersifat linier berdasarkan nilai COD berkisar diantara 0,96 dan 0,98 serta nilai deviasi standar rata-rata diantara 0,6 dan 0,8 mendekati nilai 0. Dengan melihat parameter kenaikan temperatur, grafik menunjukkan kenaikan maupun penurunan nilai gradien dari persamaan garis walaupun tidak secara signifikan. Hal ini menunjukkan parameter temperatur sangat berpengaruh terhadap kenaikan maupun kenaikan gradien persamaan linier, walaupun perbedaan temperatur yang hanya 10⁰C saja.

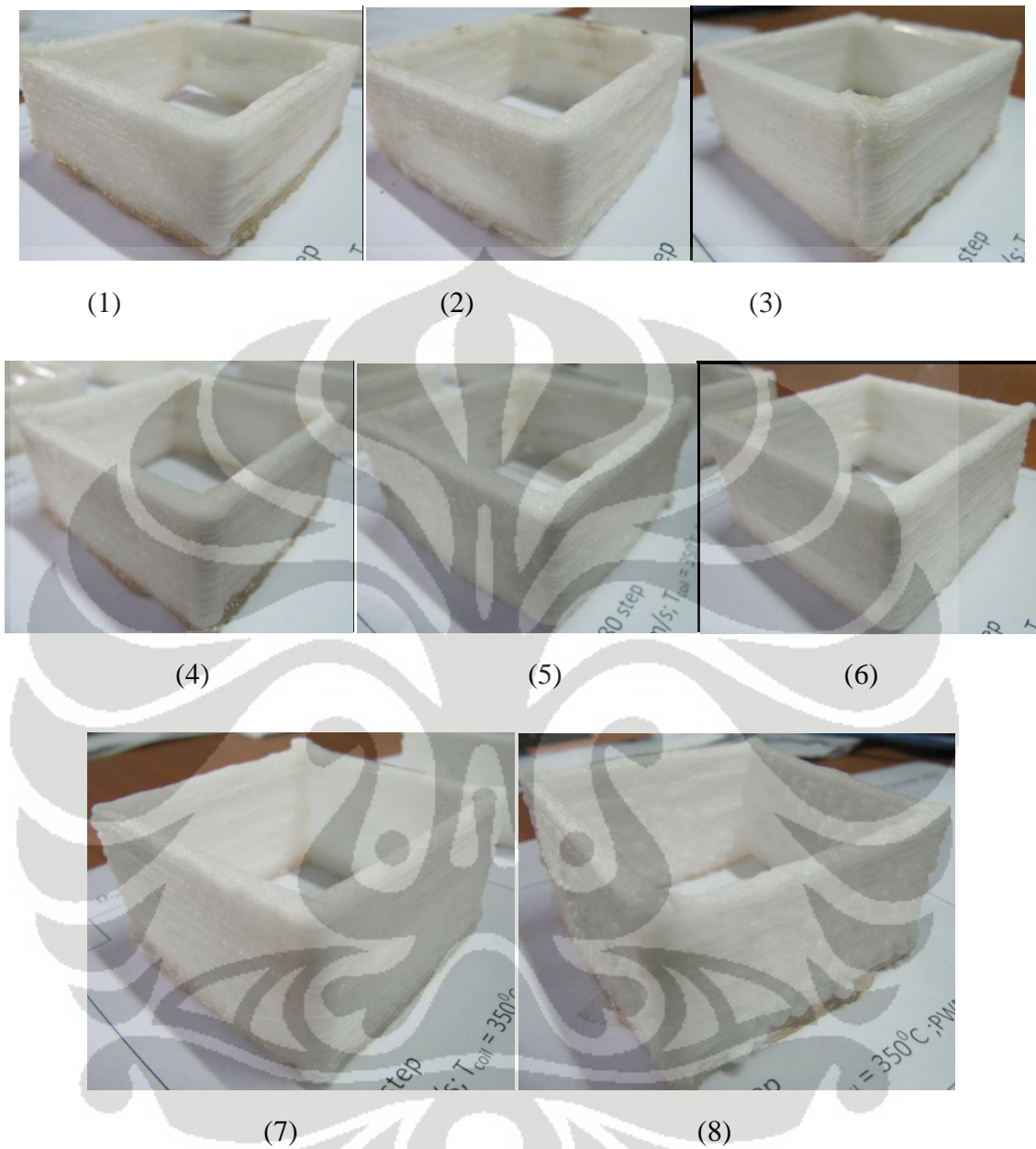
Untuk persamaan kecepatan linier *nut* terhadap *delay_us* tidak menunjukkan hasil data aktual secara linier. Dengan melihat hasil COD yang diperoleh 0,999, grafik tersebut menunjukkan bahwa data yang diperoleh hubungan-nya adalah persamaan orde ke-3. Berdasarkan gambar 8.15, kecepatan motor stepper dengan tegangan 12V hanya mampu mencapai kecepatan linier maksimum 1.567 cm/s dengan *delay_us* 400.

8.6 Hasil produk akhir

8.6.1 Pengujian parameter ketinggian *bed*



Gambar 8.16 Hasil pengujian dengan parameter pengubah ketinggian *bed*



Gambar 8.17 Hasil produk dengan faktor pengubah ketinggian *bed*

(1) $Z = 300 \text{ step}$; (2) $Z = 320 \text{ step}$; (3) $Z = 340 \text{ step}$;

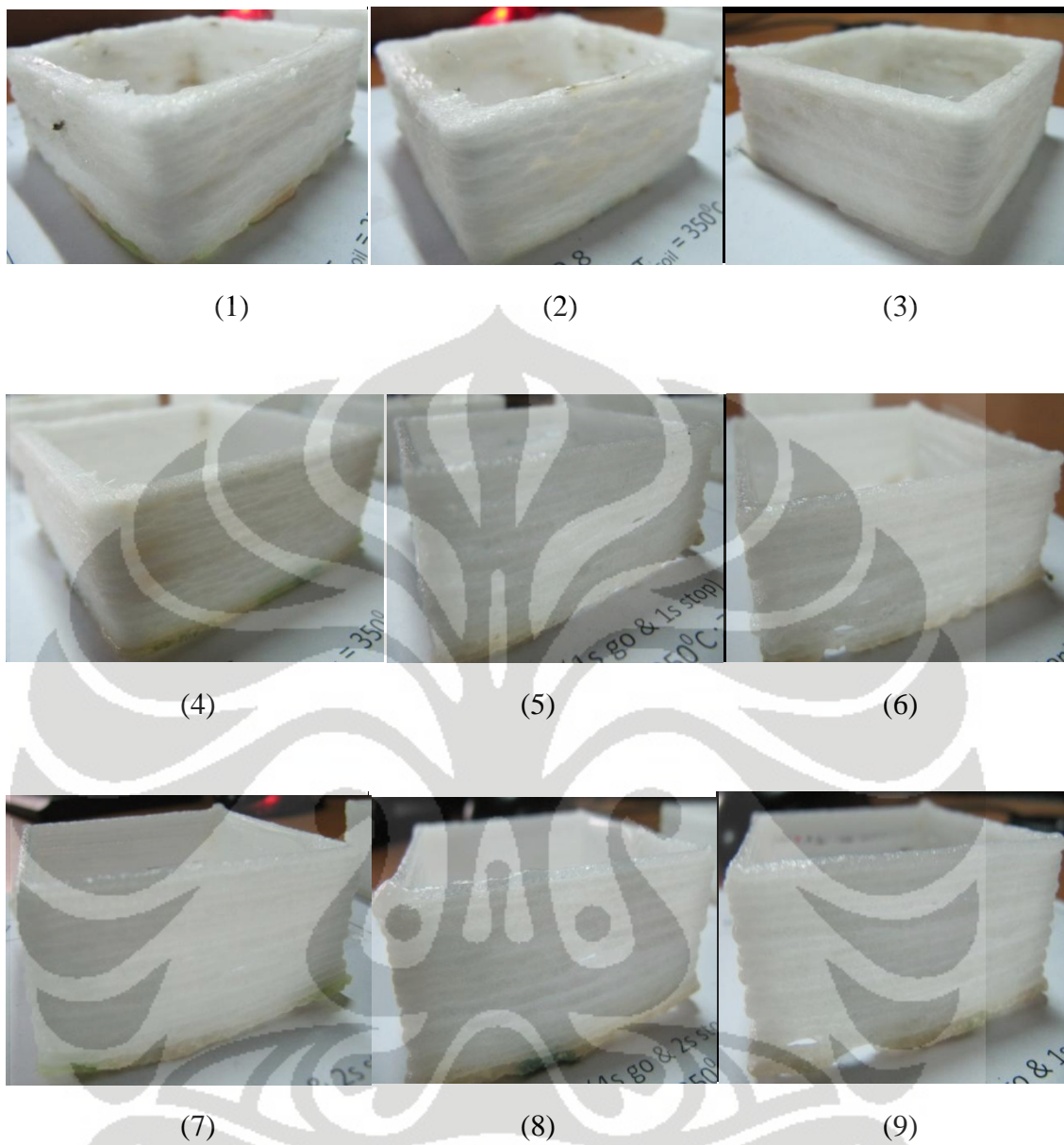
(4) $Z = 360 \text{ step}$; (5) $Z = 380 \text{ step}$; (6) $Z = 400 \text{ step}$;

(7) $Z = 420 \text{ step}$; (8) $Z = 440 \text{ step}$.

8.6.2 Pengujian parameter kecepatan *feeding*



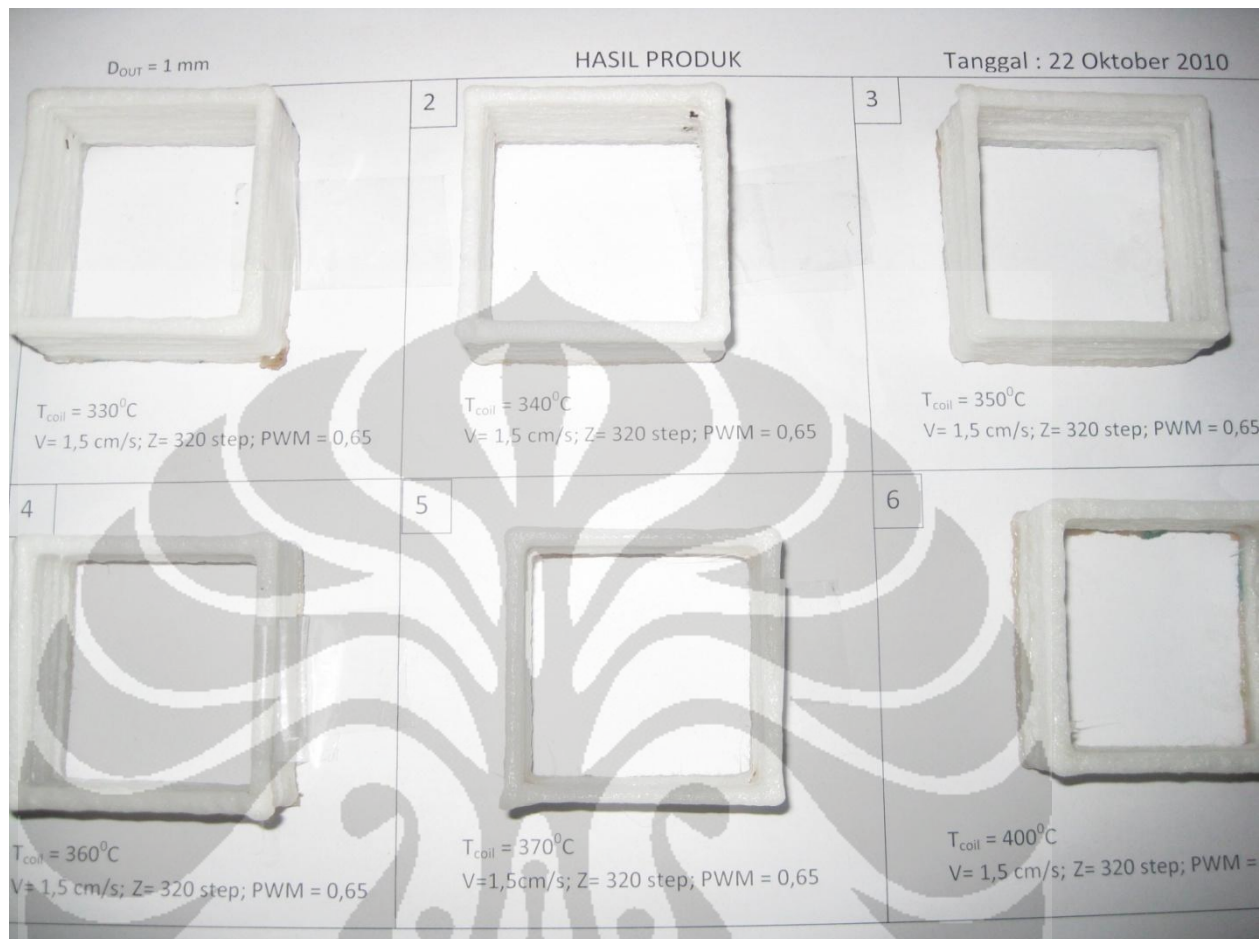
Gambar 8.18 Hasil pengujian dengan parameter pengubah kecepatan *feeding*



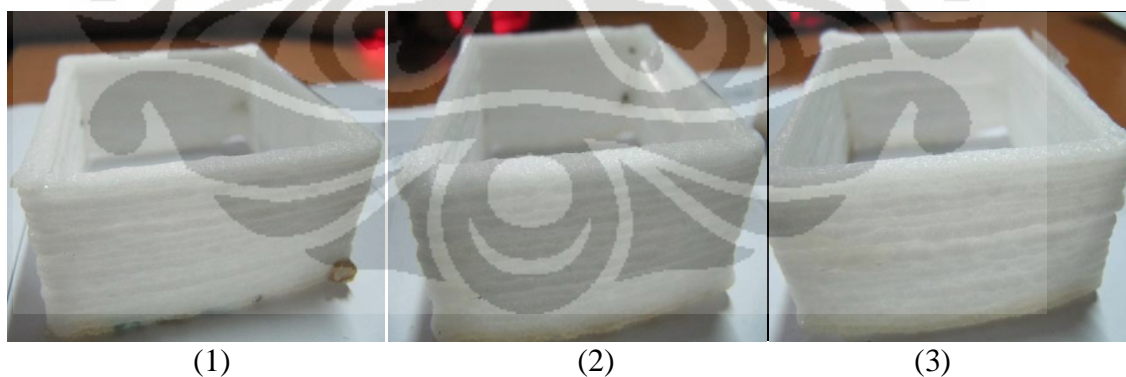
Gambar 8.19 Hasil produk dengan faktor pengubah kecepatan *feeding*

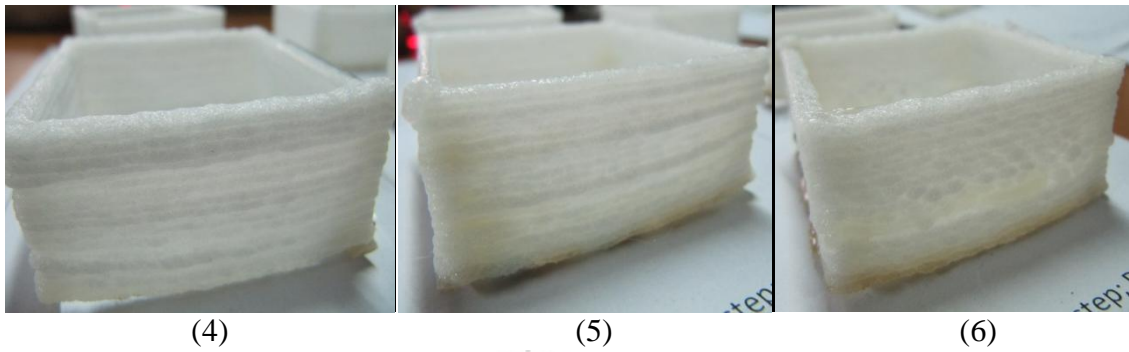
(1) $PWM = 1$; (2) $PWM = 0,8$; (3) $PWM = 0,7$;
 (4) $PWM = 0,6$; (5),(6),(7),(8),(9) $PWM = 0,65$ *intermittent*.

8.6.3 Pengujian parameter temperatur



Gambar 8.20 Hasil pengujian dengan parameter pengubah temperatur





Gambar 8.21 Hasil produk dengan faktor pengubah temperatur

(1) $T = 330^{\circ}\text{C}$; (2) $T = 340^{\circ}\text{C}$; (3) $T = 350^{\circ}\text{C}$;
 (4) $T = 360^{\circ}\text{C}$; (5) $T = 370^{\circ}\text{C}$; (6) $T = 400^{\circ}\text{C}$.

8.7 Analisis hasil produk akhir

Parameter hasil produk akhir yang didapatkan setelah pengujian adalah tebal *layer* dan tebal *path*, dengan mengubah parameter yang mempengaruhi hasil produk seperti ketinggian *bed*, kecepatan *feeding*, dan temperatur.

8.7.1 Parameter ketinggian *bed*

Tabel 8.6 Analisis hasil produk dengan faktor pengubah ketinggian *bed*

No	Z (<i>step</i>)	V (cm/s)	T_{koil} ($^{\circ}\text{C}$)	PWM Motor DC	Tebal <i>Layer</i> (mm)	Tebal <i>Path</i> (mm)
1	300	1,5	350	0,65	1.085	4
2	320	1,5	350	0,65	1.17	4
3	340	1,5	350	0,65	1.22	4
4	360	1,5	350	0,65	1.25	4
5	380	1,5	350	0,65	1.3	4
6	400	1,5	350	0,65	1.35	4
7	420	1,5	350	0,65	1.45	4
8	440	1,5	350	0,65	1.5	5

Pada pengujian ini, parameter ketinggian diubah-ubah untuk dilihat hasil produk yang diperoleh. Penggunaan satuan *step* sebenarnya mengacu pada putaran motor stepper, dengan setiap kenaikan 20 *step*, berarti $20 \times 0,032 \text{ (mm/10step)} = 0,64 \text{ mm}$, setiap kenaikan *nozzle* 0,64 mm.

Berdasarkan grafik PWM pada temperatur 350°C , kecepatan *feeding* pada PWM 0,65 adalah 7,446 cm/s. Idealnya, kecepatan *feeding* sama dengan kecepatan linier ekstruder (*Nut Ballscrew*). Pada percobaan ini kecepatan linier *nut* sudah maksimum yaitu 1,5 cm/s. Jika dilihat dari karakteristik aliran material keluaran sudah terjadi proses perubahan fasa dari *solid* menjadi fasa *liquid* pada temperatur koil 350°C dan aliran material berupa aliran kontinu.

Ditinjau dari segi kualitas produk yang dihasilkan, lihat tabel 8.6, pada ketinggian dengan nilai $z = 300 \text{ step}$ tebal layer didapatkan nilai rata-rata 1,085 mm dengan tebal lintasan sebesar 4 mm dari diameter keluaran *nozzle* sebesar 1 mm. Sehingga didapatkan akurasi lintasan sebesar 1,5 mm. Semakin dinaikkan parameter z mencapai 400 *step*, tebal lintasan menjadi 5 mm dan hasil produk yang didapatkan menjadi buruk. Ditinjau dari warna produk, tidak terlalu tampak perbedaan warna dari setiap hasil produk yang dihasilkan, akan tetapi terdapat perbedaaan ketinggian dari setiap produk. Nilai $Z = 420 \text{ step}$ merupakan nilai yang optimum jika dilihat dari hasil produk yang diperoleh.



Gambar 8.22 Perbandingan hasil produk $z = 320 \text{ step}$ dan $z = 440 \text{ step}$

8.7.2 Parameter kecepatan *feeding*

Tabel 8.7 Analisis hasil produk dengan faktor pengubah kecepatan *feeding*

No	PWM Motor DC	V (cm/s)	T _{koil} (°C)	Z (step)	Tebal Layer (mm)	Tebal Path (mm)
1	1	1,5	350	320	1.15	5.5
2	0,8	1,5	350	320	1.15	5
3	0,7	1,5	350	320	1.15	4
4	0,6	1,5	350	320	1.15	4
5	0,65 (1s go & 1s stop)	1,5	350	320	1.1	2.5
6	0,65 (2s go & 2s stop)	1,5	350	320	1.1	2.5
7	0,65 (3s go & 2s stop)	1,5	350	320	1.135	2.5
8	0,65 (4s go & 2s stop)	1,5	350	320	1.135	2.5
9	0,7 (1s go & 1s stop)	1,5	350	320	1.135	2.7

Pada pengujian ini, parameter kecepatan *feeding* diubah sementara parameter yang lain dijaga agar tetap konstan dengan kecepatan $V = 1,5$ cm/s, temperatur koil 350°C , $Z = 320$ step. Kontrol kecepatan *feeding*, yaitu dengan menggunakan metode PWM (*Pulse Width Modulation*) sehingga kecepatan motor DC 12V dapat diperlambat.

Aliran material yang keluar dari *nozzle* pada temperatur koil 350°C merupakan fasa *liquid*, dan merupakan aliran kontinu. Pada temperatur koil 350°C , peneliti mengacu pada grafik 8.11 dan 8.12. Ditinjau dari hasil produk yang diperoleh, lihat tabel 8.7, pada PWM = 1 ketebalan *path* yang didapat adalah 5,5 mm, sedangkan ketebalan *layer* relatif konstan sampai dengan PWM 0,6. Dengan menggunakan metode *intermittent* pada PWM 0,65, hasil akurasi yang diperoleh relatif lebih baik sekitar 0,75

mm dari pada dengan metode PWM *nonstop*. Akan tetapi, jika dilihat dari hasil produk yang didapatkan, menggunakan metode PWM *intermittent* terdapat kendala, dimana keadaan aliran material menjadi tidak kontinu, sehingga dinding dari produk terlihat terdapat lubang. Hal ini disebabkan oleh, error dari mekanisme *feeder* yang slip.

Dilihat dari segi warna produk, semua warna produk dengan menggunakan metode PWM kontinu berwarna putih dan relatif sama. Dibandingkan dengan menggunakan metode PWM *intermittent*, warna dari produk lebih jernih dan bening. Dari hasil percobaan, hasil produk dan akurasi yang optimum diperlihatkan oleh hasil produk no.8 pada tabel 8.7.



Gambar 8.23 Perbandingan hasil produk PWM = 0,6 dengan PWM 0,65 *intermittent*

8.7.3 Parameter temperatur

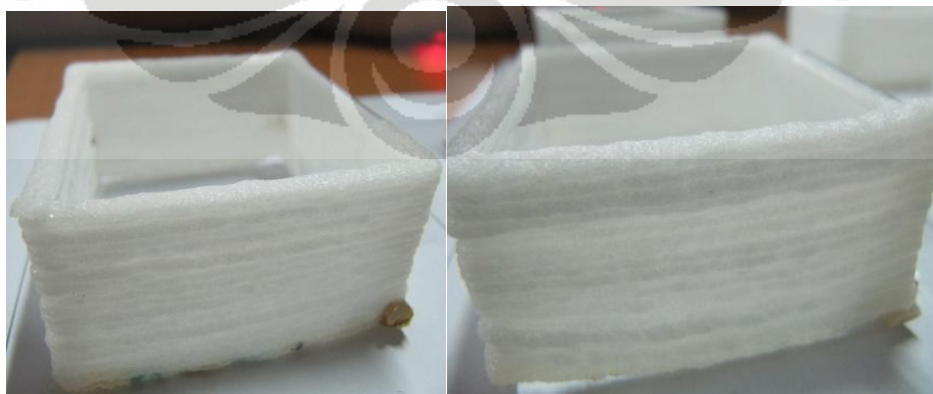
Tabel 8.8 Analisis hasil produk dengan faktor pengubah temperatur

No	T _{koil} (°C)	V (cm/s)	Z (step)	PWM Motor DC	Tebal Layer (mm)	Tebal Path (mm)
1	330	1,5	320	0,65	1,16	4
2	340	1,5	320	0,65	1,16	4
3	350	1,5	320	0,65	1,12	4
4	360	1,5	320	0,65	1,16	4
5	370	1,5	320	0,65	1.125	4
6	400	1,5	320	0,65	1.135	4

Berdasarkan hasil pengujian dengan parameter faktor pengubah temperatur, lihat tabel 8.8, hasil dari produk menunjukkan hasil yang relatif sama berupa tebal *path* rata-rata sama yaitu 4 mm, sedangkan menghasilkan tebal *layer* yang berbeda walaupun perbedaannya hanya 0,04 mm. Berdasarkan dari hasil pengujian, warna dari produk akhir relatif sama untuk setiap perubahan kenaikan temperatur 10°C dengan parameter kecepatan motor DC dengan PWM 0,65 secara kontinu. Penggunaan parameter kecepatan motor DC dengan PWM 0,65 secara kontinu, menghasilkan produk akhir yang relatif sama ditinjau dari kualitas warna dan permukaan produk akhir, sehingga dapat disimpulkan bahwa pengaruh dari temperatur sangat kecil terhadap kualitas produk pada kondisi parameter-parameter tersebut.

Pengujian dilakukan dari temperatur 330°C , dikarenakan pada temperatur 320°C , aliran material tidak keluar. Kemampuan *heater* yang dikembangkan hanya mampu mencapai maksimum 400°C , sehingga pengujian produk akhir hanya mencapai 400°C . Karakteristik permukaan material yang keluar dari nozzle terlihat seperti adanya udara yang terjebak didalam pemanas, mengingat material yang digunakan pada penelitian ini adalah Nilon PA6 yang merupakan stuktur *crystalline*.

Perbandingan produk akhir pada temperatur 330°C dengan 360°C menghasilkan ukuran dimensi tebal *layer* dan *path* yang sama. Perbedaan ketebalan layer produk akhir pada temperatur 350°C , 370°C dan 400°C dengan hasil produk ahir yang lain, kemungkinan disebabkan oleh kontribusi error yang terjadi pada pergerakan linier pada sumbu z. Sehingga dihasilkan tebal layer yang bervariasi dijangkauan 1,12 mm sampai 1,16 mm.



Gambar 8.24 Perbandingan hasil produk $T=330^{\circ}\text{C}$ dengan $T=360^{\circ}\text{C}$

BAB 9

KESIMPULAN DAN SARAN PENELITIAN LEBIH LANJUT

Berdasarkan hasil pengujian dan analisis hasil yang diperoleh, maka kesimpulan dan saran penelitian lebih lanjut adalah sebagai berikut:

9.1 Kesimpulan

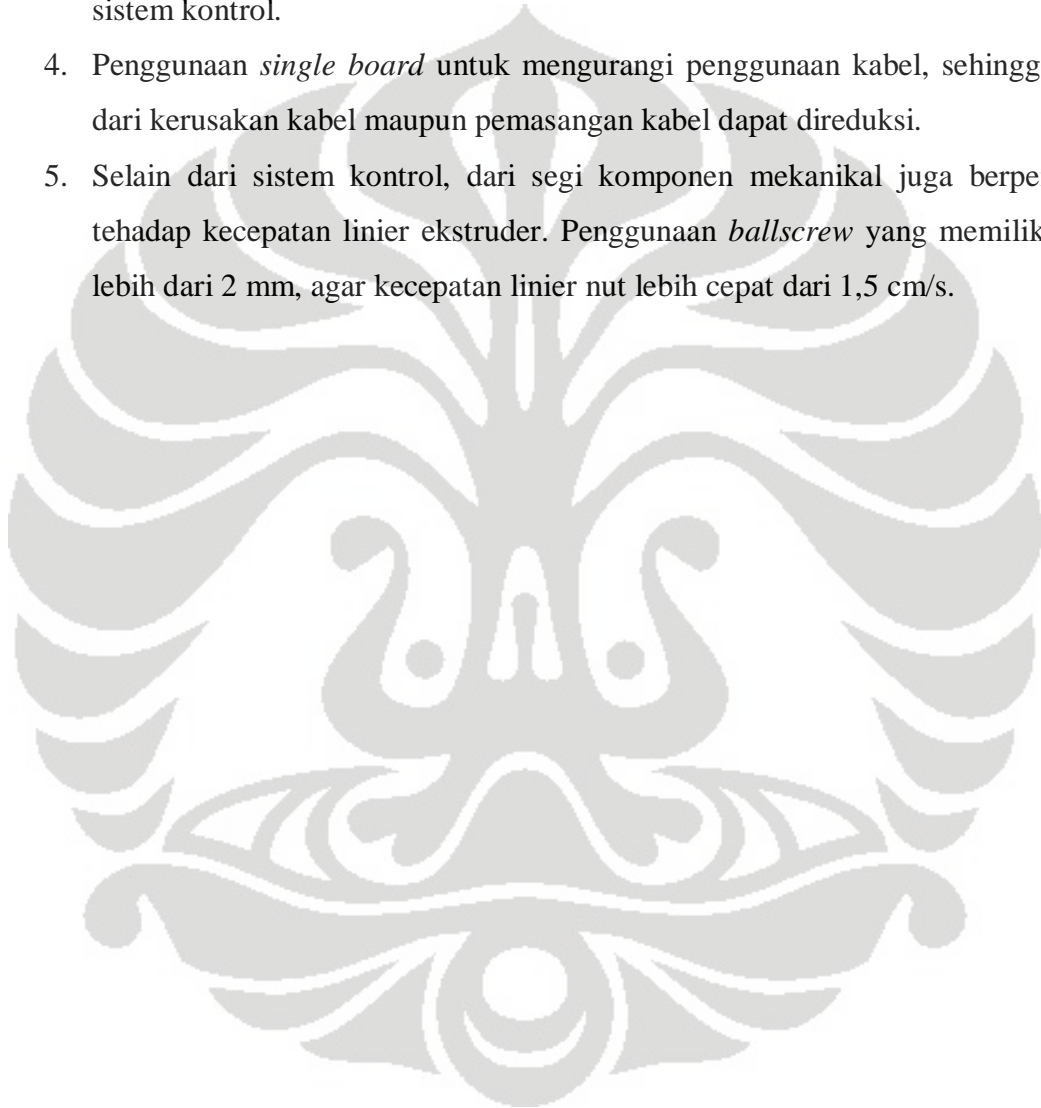
1. Perangkat keras sistem kontrol yang digunakan secara umum menggunakan mikrokontroler, dikarenakan MCU merupakan sistem *embedded* sehingga fitur-fiturnya sangat memudahkan pengguna untuk membangun sistem kontrol.
2. Sistem kontrol yang telah dikembangkan sudah berjalan dengan baik, walaupun masih ada error $0,1^0 < \Delta\theta < 0^0$ pada poros motor stepper. Pada driver motor stepper harus menggunakan *heatsink* karena panas yang dihasilkan cukup tinggi yaitu 80^0C .
3. Sistem komunikasi dengan menggunakan USART (*Universal Synchronous Asynchronous Receiver Transmitter*) antara MCU maupun MCU dengan PC sudah berjalan dengan baik, dengan error 0,2%. Sehingga error tersebut dikompensasi didalam algoritma program MCU.
4. Secara keseluruhan, hasil produk yang dihasilkan sudah cukup memuaskan dengan akurasi rata-rata 1,5mm. Parameter yang berpengaruh pada penelitian ini adalah kontrol kecepatan *feeding*. Dengan nilai optimum PWM 0,65 *intermittent* 4s *go* dan 2s *stop*.

9.2 Saran penelitian lebih lanjut

Untuk pengembangan mesin RPFDM lebih lanjut :

1. Penggunaan kecepatan MCU yang lebih cepat dari 16MHz, sehingga meningkatkan kemampuan kecepatan motor stepper, dan proses *debugging* program.

2. Pengembangan driver motor stepper masih perlu lebih ditingkatkan, dalam proses *switching* time, diperlukan *switching* time yang lebih dari (300 μ s) agar kecepatan motor stepper lebih cepat dari 400 rpm akan tetapi *driver* akan lebih cepat panas.
3. Pengujian untuk setiap komponen perlu dilakukan dengan menggunakan *oscilloscope* untuk mengetahui gelombang pulsa yang terjadi secara aktual pada sistem kontrol.
4. Penggunaan *single board* untuk mengurangi penggunaan kabel, sehingga error dari kerusakan kabel maupun pemasangan kabel dapat direduksi.
5. Selain dari sistem kontrol, dari segi komponen mekanikal juga berpengaruh terhadap kecepatan linier ekstruder. Penggunaan *ballscrew* yang memiliki *pitch* lebih dari 2 mm, agar kecepatan linier nut lebih cepat dari 1,5 cm/s.



DAFTAR PUSTAKA

- [1] Gandjar Kiswanto, Ahmad Kholil. *PENGEMBANGAN LASER TRAJECTORY PROSES RAPID PROTOTYPING UNTUK PRODUK BERKONTUR DAN PRISMATIK*. Universitas Indonesia.
- [2] http://en.wikipedia.org/wiki/Rapid_prototyping, dilihat 2 Juli 2010, jam 10.55.
- [3] http://www.meca.ucl.ac.be/cours/meca2455/cours_02-03/Overview_Rapid_Prototyping.pdf, download 2 Juli 2010, jam 10.41.
- [4] <http://computerassistedurgeryblog.com/2010/05/17/an-overview-of-rapid-prototyping-technology/>, dilihat 2 Juli 2010, jam 11.00.
- [5] Tan, Adrian dan Nixon, Timothy. *Rapid Prototyping Manufacturing System*. University of Adelaide. Australia.
http://fabathome.mae.cornell.edu/wiki/uploads/c/cd/Rapid_Prototype_Manufacturing_System_-_Timothy_Nixon_%26_Adrian_Tan_2007.pdf, download 18 Juni 2010, jam 12.49.
- [6] B. Nasrullah . *Rapid Prototyping* (Teknologi Pembuatan Model Produk).
<http://www.scribd.com/doc/13092349/Tugas-3-Rapid-Pro-to-Typing>, download 2 Juli 2010, jam 13.00.
- [7] Ali K. Kamrani, Emad Abouel Nasr. *Rapid Prototyping Theory and Practice*. Springer. 2006.
- [8] Hiwin, *Linear Guideway Technical Information Catalog*.
http://www.hiwin.com/pdf/lg/0809/Hiwin%20Linear%20Guideway%20Catalog_G99TE13-0809.pdf, download 24 Desember 2009, jam 3.03.
- [9] Hiwin, *Ballscrews Technical Information Catalog*.
<http://www.hiwin.com/pdf/bs/ballscrews.pdf>, download 24 Desember 2009, jam 3.02.

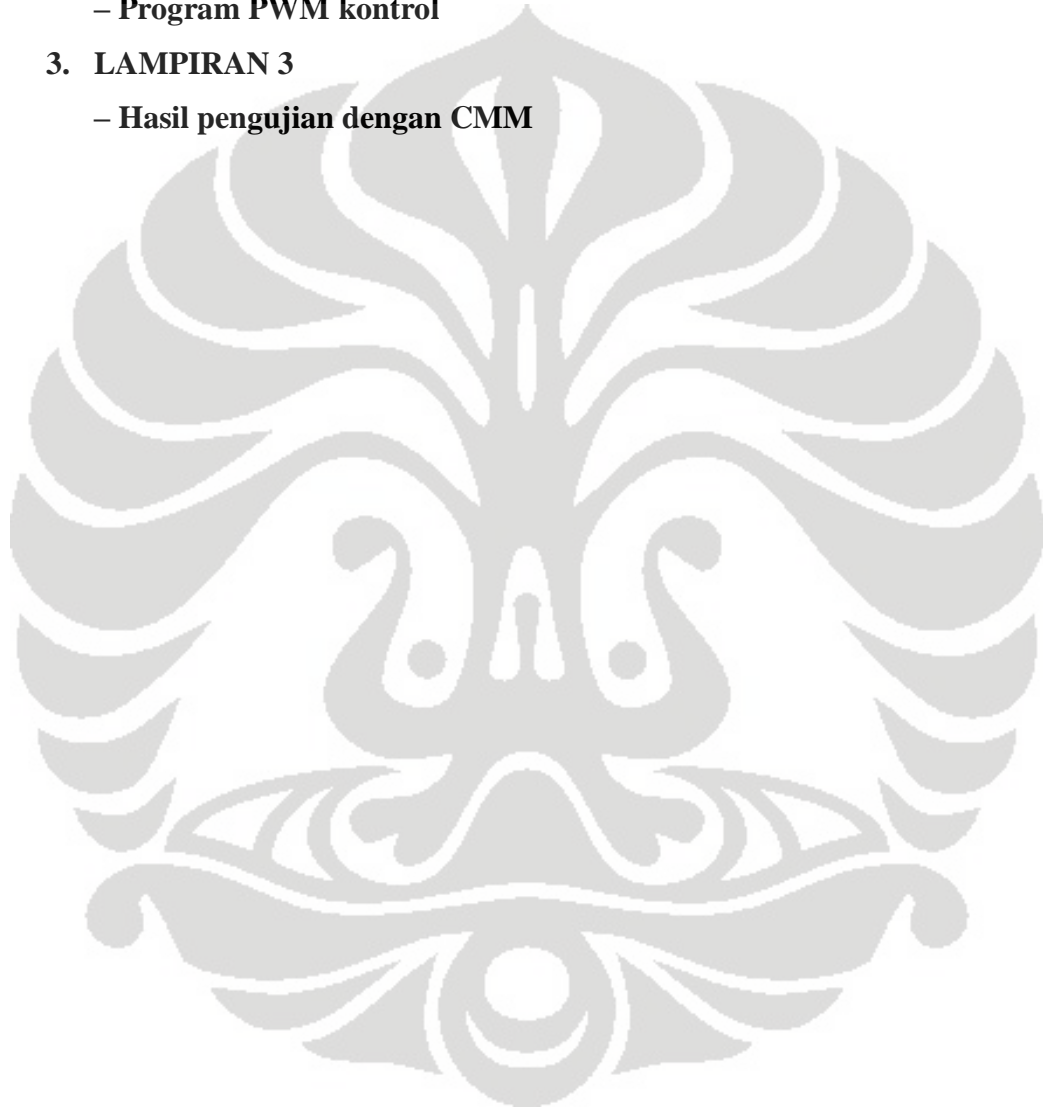
- [10] Dr. Douglas W. Jones. *Stepping Motors Fundamentals*. Microchip Technology Inc.
<http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=1D10B5FA4F619A4C3CD8158B26F81802?doi=10.1.1.127.1172&rep=rep1&type=pdf>, download 2 September 2010, jam 10.39.
- [11] Douglas W. Jones. *Stepping Motor Types*. The University of Iowa Department of Computer Science. <http://www.cs.uiowa.edu/~jones/step/types.html>, dilihat 14 Maret 2010, jam 9.40.
- [12] Autonics. *5 Phase Stepping Motor Manual Catalog*.
http://autonics.com/upload/data/8-N-14-22_5phase_en_100209.pdf, download 11 Maret 2010, jam 12.58.
- [13] http://www.rpworld.cn/en/images/FDM/fdm_1.gif, download 8 September 2010, jam 7.07.
- [14] Abu Ahmad Abdul Alim Ricki, Kurniawan Al Mutafaqqih. Prinsip Dasar Thermokopel. www.dataforth.com, download 14 Mei 2010, jam 8.33.
- [15] <http://www.auberins.com/images/TC-K3.jpg>, download 9 September 2010, jam 2.51.
- [16] https://www.egr.msu.edu/eceshop/Parts_Inventory/images/spdt%20limit%20switch.jpg, download 18 November 2010, jam 9.57.
- [17] <http://www.freewebs.com/maheshwankhede/ports.html>, dilihat 3 Oktober 2010, jam 17.59.
- [18] <http://ucables.com/img/pics1/16MHZ-DIP-CRYSTAL-OSC-R46886.jpg>, download 11 September 2010, jam 18.57.
- [19] <http://extremeelectronics.co.in/avr-tutorials/introduction-to-pwm-pulse-width-modulation/>, dilihat 11 Oktober 2010, jam 15.17.
- [20] <http://extremeelectronics.co.in/avr-tutorials/pwm-signal-generation-by-using-avr-timers/>, dilihat 11 Oktober 2010, jam 15.20.

- [21] <http://extremeelectronics.co.in/avr-tutorials/using-the-analog-to-digital-converter/>, dilihat 12 Oktober 2010, jam 10.13.
- [22] Hendra Prima Syahputra. PENGEMBANGAN SISTEM KONTROL PERGERAKAN ROBOT ARTIKULASI 5 DERAJAT KEBEBASAN BERBASIS WEB.2009.
- [23] ATMEL. ATmega2560 Microcontroller Datasheet. http://www.atmel.com/dyn/resources/prod_documents/doc2549.PDF, download 13 Maret 2010, jam 8.30.
- [24] ATMEL. ATmega16 Microcontroller Datasheet. <http://www.atmel.com/atmel/acrobat/doc2466.pdf>, download 4 Oktober 2009, jam 7.54.
- [25] Sains, Insan (2008). H-Bridge Driver : Kontrol Arah Motor. January 22, 2009. <http://insansainsprojects.wordpress.com/2008/06/05/h-bridge-driver-kontrol-arrah-motor>, dilihat 5 Juni 2010, jam 9.25.
- [26] <http://www.stepperworld.com/Tutorials/pgCurrentControl.htm>, dilihat 18 Oktober 2010, jam 7.01.
- [27] L298 Datasheet. Dual Full-Bridge Driver. <http://www.st.com/stonline/books/pdf/docs/1773.pdf>, download 15 Maret 2010, jam 18.53.
- [28] AD594/AD595 Datasheet. Monolithic Thermocouple Amplifiers with Cold Junction Compensation. http://www.analog.com/static/imported-files/data_sheets/AD594_595.pdf, download 11 Juni 2010, jam 8.01.
- [29] <http://blog.makezine.com/arduinorelay.jpg>, download 23 November 2010, jam 9.24.
- [30] <http://img1.topfreebiz.com/o2010-5/cache/Signal-Relays-HRS1--19114541345.jpg>, download 9 September 2010, jam 18.36.

- [31] http://www.tme.eu/katalog_pics/1/b/7/1b7b0ba2c0b21f9c9781b65b4e1bccc8/lcm2004awsl.jpg, *download* 9 September 2010, jam 18.53.
- [32] <http://www.modtronix.com/images/lcd162b-yhy.jpg>, *download* 9 September 2010, jam 18.53.
- [33] <http://extremeelectronics.co.in/avr-tutorials/using-lcd-module-with-avrs/>, *download* 16 November 2010, jam 14.12.
- [34] <http://www.pololu.com/catalog/product/391>, *download* 8 Agustus 2010, jam 9.32.
- [35] <http://www.elektroindonesia.com/elektro/elek24.html>, *download* 24 November, jam 18.08.
- [36] http://atmega2560-16au.katalog-tme.pl/katalog_pics/1/7/6/1762a41f981fffa14cadf240f8c47717/atmega2560-16au.jpg, *download* 9 September 2010, jam 16.55.
- [37] http://3.bp.blogspot.com/_hgM81e-eGog/S8iJwaOjSYI/AAAAAAAAAw/J_NvFcBmdRs/s1600/Sell_ATmega16-16PU_ATmega16-16PI_ATmega16_AVR_ATMEL_Integrated_Circuits_Manufacturer_exporting_direct_from_China.jpg, *download* 30 November 2010, jam 11.36.

LAMPIRAN

1. LAMPIRAN 1
 - Program Temperatur kontrol
2. LAMPIRAN 2
 - Program PWM kontrol
3. LAMPIRAN 3
 - Hasil pengujian dengan CMM



LAMPIRAN 1

```

/*****
Project           : Program Temperatur Kontrol
Version          : v.1
Date             : 6/10/2010
Author           : Rendi Kurniawan
Company          : Universitas Indonesia
Comments         : -
Chip type        : ATmega16
Program type     : Application
Clock frequency  : 16.000000 MHz
Memory model     : Small
External RAM size : 0
Data Stack size  : 256
*****/

#include <mega16.h>

// Alphanumeric LCD Module functions
#asm
.equ __lcd_port=0x15 ;PORTC
#endasm
#include <lcd.h>
#include <stdlib.h>
#include <delay.h>
#include <math.h>
#include <stdio.h>

#define ADC_VREF_TYPE 0x00

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

```

```

void main(void)
{
// Declare your local variables here
float temp[32];
float suhu,suhu1,total;
int i,j;
char LCD[16];

// Input/Output Ports initialization
// Port A initialization
PORTA=0x00;
DDRA=0x00;

// Port B initialization
PORTB=0x00;
DDRB=0x01;

// Port C initialization
PORTC=0x00;
DDRC=0x00;

// Port D initialization
PORTD=0x00;
DDRD=0x38;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x67;

// ADC initialization
// ADC Clock frequency: 125.000 kHz
// ADC Voltage Reference: AREF pin
// ADC Auto Trigger Source: Free Running
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0xA7;
SFIOR&=0x1F;

// LCD module initialization
lcd_init(16);

```

```
while (1)
{
  // Place your code here
  //Get data 32 times
  for(i=0;i<=31;i++)
  {
    temp[i]= read_adc(0)*0.48828125 ;
  }

  //Sum
  total = 0;
  for(j=0;j<=31;j++)
  {
    total += temp[j];
  }

  //Average
  suhu = (total/32);
  suhu1 = ((suhu*1.006) - 0.657);
  ftoa (suhu1,1,LCD);
  lcd_clear();
  lcd_gotoxy(0,0);
  lcd_puts(LCD);
  lcd_gotoxy(6,0);
  lcd_putsf("Celcius");
  delay_ms(1500);
  if (suhu >=330)
  {
    PORTB.0 = 1;
    putchar('c');
  }
  else
  {
    PORTB.0 = 0;
  }
};
}
```

LAMPIRAN 2

```

/*****
Project           : Program PWM kontrol
Version          : v.1
Date             : 8/16/2010
Author           : Rendi Kurniawan
Company          : Universitas Indonesia
Comments         : Fast PWM
Chip type        : ATmega16
Program type     : Application
Clock frequency  : 16.000000 MHz
Memory model     : Small
External RAM size : 0
Data Stack size  : 256
*****/

#include <mega16.h>
#include <delay.h>
#include <stdio.h>
#include <stdlib.h>

#define on 1
#define off 0
#define MAX 0xFF

// Declare your global variables here
void motor_pwm (float pwm_motor, unsigned int arah1, unsigned int arah2)

{
    OCR0 = MAX*pwm_motor;
    PORTD.3=arah1;
    PORTD.4=arah2;
    PORTB.3=1; //enable
}

void main(void)
{
    // Declare your local variables here
    char PWM[1];
    int i;
    // Input/Output Ports initialization
    // Port A initialization
    PORTA=0x00;
    DDRA=0x00;

```

```

// Port B initialization
PORTB=0x00;
DDRB=0x08;
// Port C initialization
PORTC=0x00;
DDRC=0x00;
// Port D initialization
PORTD=0x00;
DDRD=0x38;
// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x67;
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: 250.000 kHz
// Mode: Fast PWM top=FFh
// OC0 output: Non-Inverted PWM
TCCR0=0x6B;
TCNT0=0x00;
OCR0=0x00;

while (1)
{
// Place your code here
PWM[0]=getchar();
while (PWM[0]!='u')
{
motor_pwm (0.7,on,off);
PWM[0]=getchar();
}
while(1)
{
motor_pwm(0,off,off);
if (PWM[0]='y')
{
break;
}
}
}
}

```

LAMPIRAN 3



User name
Admin
02.11.2010 10:24

Part name
Rendi-MCU2560

1	2	3	4					
El. No.	Line No	Element	Pkt. Ref.	X-Coord. Nominal	Y-Coord. Y-Angle Up/Lo	Z-Coord. Z-Angle Actual	Diam/Size Dist./Ang. Dev./Error	Vari/Rad mm
1	2	Plane Mean	4	5.363	1.096	65.574		0.000
				85:19:31	89:02:46	4:46:17		
2	5	Plane Mean	4	5.339	1.062	65.538		0.001
				85:20:37	89:04:29	4:44:52		
3	8	Plane Mean	4	5.317	1.060	65.485		0.002
				85:21:32	89:04:31	4:43:58		
4	11	Plane Mean	4	5.210	1.046	65.228		0.002
				85:26:03	89:05:02	4:39:26		
5	14	Plane Mean	4	5.433	1.092	65.761		0.002
				85:16:39	89:03:07	4:49:01		
6	17	Plane Mean	4	5.358	1.131	65.545		0.003
				85:19:37	89:00:54	4:46:34		
7	20	Plane Mean	4	5.450	1.051	65.831		0.000
				85:16:05	89:05:20	4:49:09		
8	23	Plane Mean	4	5.260	1.044	65.355		0.001
				85:23:58	89:05:17	4:41:25		
9	26	Plane Mean	4	5.328	1.061	65.544		0.002
				85:21:11	89:04:34	4:44:18		
10	29	Plane Mean	4	5.321	1.046	65.504		0.002
				85:21:23	89:05:17	4:43:57		
11	32	Plane Mean	4	5.560	1.068	66.091		0.001
				85:11:30	89:04:38	4:53:47		
11	35	Plane Flatness			0.010	0.001	0.001	**----
				4P	S/MIN/MAX	0.0005	(1)0.000 (2)0.000	
1	37	Angle					0:02:03	
2	38	Angle					0:00:55	
3	39	Angle					0:04:32	
4	40	Angle					0:09:35	
5	41	Angle					0:03:42	
6	42	Angle					0:07:53	
7	43	Angle					0:05:40	
8	44	Angle					0:02:53	
9	45	Angle					0:00:45	
10	46	Angle					0:09:55	
11	47	Angle					0:08:14	

User name
Admin
 02.11.2010 11:04

Part name
Rendi-MCU2560

1	2	3	4					
El. No.	Line No	Element	Pkt. Ref.	X-Coord. Nominal	Y-Coord. Y-Angle Up/Lo	Z-Coord. Z-Angle Actual	Diam/Size Dist./Ang. Dev./Error	Vari/Rad mm
1	2	Plane Mean	4	5.363	1.046	65.606		0.001
				85:19:39	89:05:24	4:45:38		
1	5	Plane Flatness			0.010	0.001	0.001	**----
				4P	S/MIN/MAX	0.0010	(1)-0.001 (2)0.001	
2	7	Plane Mean	4	5.445	1.032	65.825		0.002
				85:16:18	89:06:17	4:48:46		
3	10	Plane Mean	4	5.358	1.030	65.605		0.001
				85:19:53	89:06:13	4:45:15		
4	13	Plane Mean	4	5.519	1.050	65.997		0.001
				85:13:12	89:05:30	4:51:57		
5	16	Plane Mean	4	5.400	1.086	65.656		0.001
				85:17:57	89:03:19	4:47:43		
6	19	Plane Mean	4	5.368	1.040	65.622		0.001
				85:19:25	89:05:43	4:45:48		
7	22	Plane Mean	4	5.424	1.039	65.764		0.001
				85:17:08	89:05:52	4:48:01		
8	25	Plane Mean	4	5.487	1.017	65.945		0.002
				85:14:40	89:07:11	4:50:12		
9	28	Plane Mean	4	5.514	1.056	65.986		0.000
				85:13:27	89:05:09	4:51:46		
10	31	Plane Mean	4	5.351	1.047	65.571		0.001
				85:20:06	89:05:18	4:45:13		
1	34	Angle					0:03:28	
2	35	Angle					0:03:35	
3	36	Angle					0:06:44	
4	37	Angle					0:05:13	
5	38	Angle					0:02:49	
6	39	Angle					0:02:18	
7	40	Angle					0:02:48	
8	41	Angle					0:02:22	
9	42	Angle					0:06:39	
10	43	Angle					0:00:28	

User name
Admin
02.11.2010 11:36

Part name
Rendi-MCU2560

1	2	3	4					
El. No.	Line No	Element	Pkt. Ref.	X-Coord. Nominal	Y-Coord. Y-Angle Up/Lo	Z-Coord. Z-Angle Actual	Diam/Size Dist./Ang. Dev./Error	Vari/Rad mm
1	2	Plane Mean	4	5.272	1.047	65.365		0.000
				85:23:21	89:05:07	4:42:04		
1	5	Plane Flatness			0.010	0.000	0.000	*-----
				4P	S/MIN/MAX	0.0001	(2)0.000	(1)0.000
2	7	Plane Mean	4	5.185	0.980	65.210		0.001
				85:27:16	89:08:31	4:37:34		
3	10	Plane Mean	4	5.197	0.850	65.367		0.000
				85:27:18	89:15:27	4:36:20		
4	13	Plane Mean	4	5.149	1.042	65.050		0.000
				85:28:29	89:05:06	4:37:02		
5	16	Plane Mean	4	5.361	1.035	65.605		0.002
				85:19:45	89:05:58	4:45:26		
6	19	Plane Mean	4	5.200	1.036	65.189		0.000
				85:26:23	89:05:31	4:39:01		
7	22	Plane Mean	4	5.285	1.038	65.407		0.000
				85:22:51	89:05:38	4:42:27		
8	25	Plane Mean	4	5.241	1.030	65.300		0.000
				85:24:44	89:05:57	4:40:33		
9	28	Plane Mean	4	5.066	1.066	64.812		0.001
				85:31:53	89:03:38	4:34:01		
10	31	Plane Mean	4	5.295	1.037	65.435		0.000
				85:22:27	89:05:43	4:42:50		
1	34	Angle					0:05:11	
2	35	Angle					0:06:57	
3	36	Angle					0:10:25	
4	37	Angle					0:08:46	
5	38	Angle					0:06:38	
6	39	Angle					0:03:32	
7	40	Angle					0:01:54	
8	41	Angle					0:07:31	
9	42	Angle					0:09:39	
10	43	Angle					0:01:04	

User name

Admin

02.11.2010 12:59

Part name

Rendi-MCU2560

1	2	3	4					
El. No.	Line No	Element	Pkt. Ref.	X-Coord. Nominal	Y-Coord. Y-Angle Up/Lo	Z-Coord. Z-Angle Actual	Diam/Size Dist./Ang. Dev./Error	Vari/Rad mm
1	2	Plane Mean	4	5.157	1.025	65.089		0.001
				85:28:12	89:06:01	4:37:08		
1	5	Plane Flatness			0.010	0.001	0.001	**----
				4P	S/MIN/MAX	0.0011	(2)-0.001	(1)0.001
2	7	Plane Mean	4	5.339	1.017	65.568		0.000
				85:20:43	89:06:51	4:44:19		
3	10	Plane Mean	4	5.177	1.024	65.143		0.000
				85:27:25	89:06:08	4:37:53		
4	13	Plane Mean	4	5.166	1.018	65.121		0.000
				85:27:55	89:06:27	4:37:20		
5	16	Plane Mean	4	5.050	1.037	64.799		0.000
				85:32:40	89:05:08	4:32:55		
6	19	Plane Mean	4	5.254	1.040	65.327		0.000
				85:24:10	89:05:27	4:41:12		
7	22	Plane Mean	4	5.299	1.050	65.436		0.000
				85:22:15	89:05:02	4:43:10		
8	25	Plane Mean	4	5.407	1.041	65.722		0.000
				85:17:51	89:05:45	4:47:21		
9	28	Plane Mean	4	5.181	1.026	65.154		0.001
				85:27:13	89:06:03	4:38:05		
10	31	Plane Mean	4	5.228	1.025	65.275		0.000
				85:25:18	89:06:11	4:39:57		
1	34	Angle					0:07:32	
2	35	Angle					0:06:43	
3	36	Angle					0:00:36	
4	37	Angle					0:04:56	
5	38	Angle					0:08:31	
6	39	Angle					0:01:58	
7	40	Angle					0:04:27	
8	41	Angle					0:09:23	
9	42	Angle					0:01:56	
10	43	Angle					0:02:55	

User name
Admin
02.11.2010 13:33

Part name
Rendi-MCU2560

1	2	3	4					
El. No.	Line No	Element	Pkt. Ref.	X-Coord. Nominal	Y-Coord. Y-Angle Up/Lo	Z-Coord. Z-Angle Actual	Diam/Size Dist./Ang. Dev./Error	Vari/Rad mm
1	2	Plane Mean	4	-1.759	0.377	38.199		0.003
				92:38:12	89:26:06	2:41:48		
1	5	Plane Flatness			0.010	0.003	0.003	***_--
				4P	S/MIN/MAX	0.0023	(1)-0.001 (2)0.001	
2	7	Plane Mean	4	-1.770	0.375	38.115		0.001
				92:39:32	89:26:12	2:43:05		
3	10	Plane Mean	4	-1.722	0.373	38.516		0.001
				92:33:35	89:26:46	2:37:09		
4	13	Plane Mean	4	-1.697	0.394	38.662		0.000
				92:30:45	89:24:59	2:34:46		
5	16	Plane Mean	4	-1.721	0.383	38.496		0.000
				92:33:35	89:25:48	2:37:21		
6	19	Plane Mean	4	-1.675	0.388	38.849		0.001
				92:28:07	89:25:40	2:32:03		
7	22	Plane Mean	4	-1.719	0.384	38.510		0.001
				92:33:21	89:25:44	2:37:08		
8	25	Plane Mean	4	-1.688	0.376	38.778		0.000
				92:29:33	89:26:40	2:33:13		
9	28	Plane Mean	4	-1.732	0.373	38.438		0.001
				92:34:47	89:26:41	2:38:20		
10	31	Plane Mean	4	-1.699	0.383	38.681		0.000
				92:30:51	89:26:01	2:34:38		
1	34	Angle					0:01:20	
2	35	Angle					0:05:58	
3	36	Angle					0:03:21	
4	37	Angle					0:02:57	
5	38	Angle					0:05:28	
6	39	Angle					0:05:13	
7	40	Angle					0:03:55	
8	41	Angle					0:05:14	
9	42	Angle					0:03:59	
10	43	Angle					0:07:21	