



UNIVERSITAS INDONESIA

**SISTEM KENDALI TEMPERATUR ALAT PENDINGIN
TEMPERATUR TERKENDALI**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Ahli Madya
(A.Md)**

**AYU ANASTASIA
0706228792**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
DEPARTEMENT FISIKA
PROGRAM D3 INSTRUMENTASI INDUSTRI
DEPOK
2010**

HALAMAN PERNYATAAN ORISINALITAS

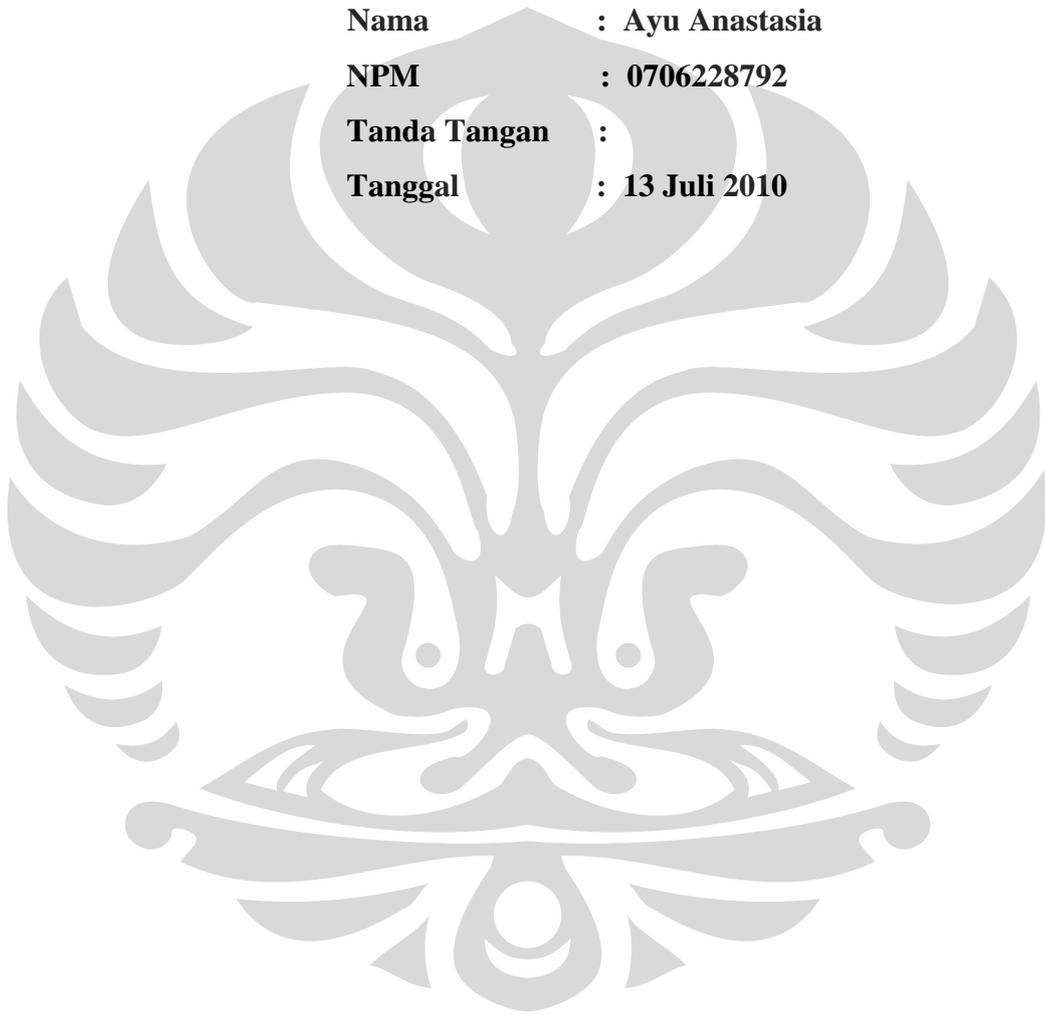
**Tugas akhir ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : Ayu Anastasia

NPM : 0706228792

Tanda Tangan :

Tanggal : 13 Juli 2010



HALAMAN PENGESAHAN

Tugas Akhir ini diajukan oleh :
Nama : Ayu Anastasia
NPM : 0706228792
Program Studi : Fisika Instrumentasi Industri
Judul Tugas Akhir : Sistem Kendali Temperatur Alat Pendingin Temperatur
Terkendali

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Ahli Madya (A.MD) pada Program Studi Diploma 3 Instrumentasi Elektronika Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Drs. Arief Sudarmaji, M.T ()

Penguji : Isom Mudzakir, M.Si ()

Penguji : Djati Handoko, M.Si ()

Ditetapkan di : Ruang Seminar Fakultas MIPA, Kampus UI Depok

Tanggal : 13 Juli 2010

KATA PENGANTAR

Puji dan syukur kami panjatkan kepada Allah SWT beserta Nabi Muhammad SAW, yang telah melimpahkan segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Laporan Tugas Akhir ini. Walaupun dalam penyusunan Laporan Tugas Akhir ini penulis menemukan berbagai macam kesulitan, tetapi Allah SWT senantiasa memberikan tetesan rahmat-Nya sehingga semua rintangan dan tantangan dapat dilalui dengan ridho-Nya.

Penyusunan Laporan Tugas Akhir yang berjudul “ **Sistem Kendali Temperatur Alat Pendingin Temperatur Terkendali** ” yang bertujuan untuk memenuhi syarat dalam menyelesaikan pendidikan program studi Diploma 3 Instrumentasi Elektronika dan Industri, Departemen Fisika, FMIPA, Universitas Indonesia.

Dalam melaksanakan Tugas Akhir sampai penyelesaian Laporan Tugas Akhir ini, penulis banyak mendapat bantuan dan bimbingan dari berbagai pihak. Untuk itu pada kesempatan ini penulis ingin menyampaikan terima kasih dan rasa hormat kepada:

1. Kedua orangtua dan adiku untuk kasih sayang, doa dan dukungannya sehingga laporan ini selesai dibuat.
2. Dr. Prawito, selaku Ketua program Diploma 3 Instrumentasi Elektronika dan Industri Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia
3. Bapak Surya Dharma, M.Si selaku koordinator Tugas Akhir program Diploma 3 Instrumentasi Elektronika dan Industri Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia.
4. Drs. Arief Sudarmaji, M.T, selaku dosen pembimbing yang telah memberikan petunjuk, kemudahan dalam berpikir dan bimbingan dalam penyelesaian tugas akhir ini. Terima kasih atas bimbingan dan kerjasamanya selama proses penelitian sehingga penelitian berjalan baik dan menyenangkan.

5. Seluruh dosen dan staff pengajar Program Diploma 3 Instrumentasi Elektronika dan Industri Departemen Fisika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Indonesia.
6. Pak Katman dan Pak Parno yang selalu membantu dan memudahkan dalam peminjaman alat selama penelitian di Laboratorium Elektronika dan Bengkel Mekanik.
7. Pacarku tersayang, Moch. Gito Adrian yang selalu mengerti di saat-saat susah dan senang orang yang stress TA.
8. Kawan kerja yaitu M. Akbar Barrinaya, yang selalu dikerjain dan direpotkan oleh partnernya sendiri.
9. Semua angkatan 2007 lainnya yang selalu berjuang dan bersemangat bareng-bareng buat ngerjain TA kita.
10. Dan semua pihak yang secara tidak langsung ikut terlibat dalam pembuatan tugas akhir ini yang tidak saya sebutkan satu persatu, semoga amal baik yang telah dilakukan dibalas oleh ALLAH SWT.

Menyadari keterbatasan pengalaman serta kemampuan yang dimiliki penulis, sudah tentu terdapat kekurangan serta kemungkinan jauh dari sempurna, untuk itu penulis tidak menutup diri dan mengharapkan adanya saran serta kritik dari berbagai pihak yang sifatnya membangun guna menyempurnakan penyusunan tugas akhir ini.

Akhir kata semoga penyusunan tugas akhir ini dapat memberikan manfaat bagi semua pihak yang bersangkutan, khususnya bagi penulis dan umumnya bagi para pembaca.

Depok, Juni 2010

Penulis

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

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

Nama : Ayu Anastasia
NPM : 0706228792
Program Studi : Diploma 3 Instrumentasi Industri
Departemen : Fisika
Fakultas : Matematika dan Ilmu Pengetahuan Alam
Jenis karya : Laporan Tugas Akhir

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

Sistem Kendali Temperatur Alat Pendingin Temperatur Terkendali

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

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : 13 Juli 2010
Yang menyatakan

(Ayu Anastasia)

ABSTRAK

Nama : Ayu Anastasia
Program Studi : Diploma 3 Instrumentasi Industri
Judul : Sistem Kendali Temperatur Alat Pendingin Temperatur
Terkendali

Telah dilakukan perancangan dan realisasi sistem. Alat ini dapat dimanfaatkan sebagai sistem kendali serta kontrol otomatis temperatur alat pendingin. Sistem ini terdiri atas perangkat keras dan perangkat lunak. Perangkat keras terdiri atas sebuah mikrokontroler, sensor LM 35, LCD (*Liquid Cristal Display*), ADC (*Analog to Digital Converter*), pemanas, pendingin (AC), rangkaian *keypad*. Program aplikasi ini bekerja pada saat dinyalakan maka inisialisasi *hardware* dilakukan kemudian menampilkan temperatur yang terdeteksi oleh LM 35 pada LCD. Memasukkan *Set point* melalui *keypad* dan proses eksekusi dilakukan oleh mikrokontroler untuk menentukan mati/hidup sistem pendingin dan pemanas. Setelah setpoint dimasukkan maka nilai *set point* dibandingkan temperatur sebenarnya jika setpoint lebih besar maka *heater* akan hidup dan sebaliknya jika *set point* lebih kecil maka *chiller* hidup. Temperatur yang bisa dikendalikan oleh alat pengontrol temperatur.

Kata Kunci : Pemanas, Pendingin, ADC, LM35, *Set point*.

ABSTRACT

Name : Ayu Anastasia
Study program: Instrumentation Industry
Title : Temperature Control Chiller System

Design and system realization have been done. This device used as monitoring system and automatic control of room temperature. The system consists of hardware and software. Hardware consists of a microcontroller, LM 35 sensor, LCD (Liquid Crystal Display), ADC (Analog to Digital Converter), heater, Chiller (AC), keypad circuit. This application program works when it is being turned on, hardware initialization begins, then it shows temperature detected by LM 35 on LCD. Entering setpoint through keypad and execution process is done by microcontroller to determine whether cooling and heating system is on/off. Temperature which can be controlled by temperature controller

Key Word : Heater, Chiller, ADC, LM35, Set point.

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
HALAMAN PERSETUJUAN PUBLIKASI	vi
ABSTRAK	vii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii
DAFTAR LAMPIRAN	xiii
BAB 1 PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Tujuan Penelitian	2
1.3. Deskripsi Singkat.....	2
1.4. Batasan Masalah.....	3
1.5. Metodologi Penelitian	4
1.6. Sistematika Penulisan	5
BAB 2 TEORI DASAR	
2.1. Temperatur	7
2.2. Sensor dan Sinyal.....	8
2.3 IC LM35.....	9
2.4. Mikrokontroler.....	10
2.4.1 Mikrokontroler AVR Atmega16.....	11
2.4.2 Fitur Atmega16.....	11
2.4.3 Konfigurasi Atmega16.....	12
2.4.4 Komunikasi Serial Pada Atmega16.....	13
2.5. BASCOM AVR.....	15
2.5.1 Instruksi dalam BASCOM AVR.....	15
2.6. LCD.....	19

2.7. Downloader.....	19
BAB 3 PERANCANGAN DAN CARA KERJA SISTEM	
3.1. Perancangan Kerja Sistem.....	21
3.2. Perancangan Perangkat Keras (<i>Hardware</i>).....	22
3.2.1 Rangkaian Minimum Sistem Atmega16.....	22
3.2.2 Konsep I/O Pada Mikrokontroler AVR Atmega.....	24
3.2.3 Rangkaian Keypad.....	25
3.3. Perancangan Perangkat Lunak (<i>Software</i>).....	26
3.3.1 Proses Pada Keypad.....	26
3.3.2 Pengambilan Data.....	29
3.3.3 Proses Pada Sistem Minimum Atmega16.....	29
BAB 4 HASIL PENGUJIAN DAN PEMBAHASAN	
4.1 Pengujian Keypad.....	33
4.2 Pengujian LM35.....	34
4.2 Pengujian Sistem Kendali Temperatur.....	35
BAB 5 PENUTUP	
5.1. Kesimpulan.....	37
5.2. Saran.....	37
DAFTAR ACUAN.....	38
LAMPIRAN	

DAFTAR GAMBAR

	Halaman
1.1 Bagan Keseluruhan Sistem.....	3
2.1 Bentuk LM35.....	10
2.2 Konfigurasi Atmega16.....	13
2.3 Konfigurasi Slot Konektor Serial DB-9	14
2.4 Layout Posisi Komponen USB AVR ISP (USB AVR Downloader).....	20
2.5 Keterangan Pin.....	20
3.1 Diagram Blok Sistem.....	21
3.2 Blok Diagram Pengendali Temperatur	23
3.3 Rangkaian Dasar Keypad	26
3.4 Tampilan Awal LCD	26
3.5 Flowchart Program Pengendali.....	31
4.1 Grafik Nilai Rata-Rata Perubahan Nilai ADC Terhadap Temperatur.....	34

DAFTAR TABEL

	Halaman
2.1 Daftar nama pin dan sinyal konektor serial DB-9.....	15
4.1 Data Keypad	32
4.2 Tabel Pengendalian	34



DAFTAR LAMPIRAN

Lampiran 1: Rangkaian *Mynsis*

Lampiran 2: Program Utama

Lampiran 3: *Data Sheet*

Lampiran 4: Data Pengamatan



BAB 1

PENDAHULUAN

Pada Bab ini penulis menjelaskan mengenai latar belakang masalah mengapa alat ini dibuat, tujuan dari penelitian, batasan masalah dari alat yang akan dibuat oleh penulis, deskripsi singkat mengenai alat yang akan dibuat, metodologi penelitian, dan sistematika penulisan laporan.

1.1 Latar Belakang

Perkembangan zaman yang semakin maju menuntut perkembangan teknologi yang semakin modern dan canggih serta handal. Kestabilan merupakan satu tuntutan yang harus dipenuhi pada sebuah sistem teknologi, untuk membuat suatu sistem dengan keadaan yang stabil tidaklah mudah karena banyak sekali faktor yang dapat mempengaruhi suatu kestabilan dari sistem tersebut, salah satu faktor yang mempengaruhi kestabilan dari sistem adalah temperatur, perubahan temperatur dapat mempengaruhi karakteristik dari suatu sistem yang mengakibatkan terjadi ketidak stabilan. Seperti pada suatu reaksi kimia dalam hal ini suatu reaktor, suatu reaktor yang harus dilengkapi dengan suatu sistem pendingin, yang berfungsi untuk menjaga atau mengontrol temperatur agar tetap pada temperatur tertentu.

Oleh karena itu untuk memenuhi tuntutan dunia yang semakin maju yang membutuhkan suatu sistem yang memiliki kestabilan yang baik maka penulis merancang suatu sistem kendali temperatur alat pendingin dimana alat pendingin dapat di-*set* serta ditampilkan. Sistem yang dibuat ini memanfaatkan kemampuan mikrokontroler Atmega16 dalam akuisisi data dan mengambil keputusan. Kawasan temperatur yang bisa dikendalikan adalah keadaan temperatur lingkungan sampai dengan kemampuan minimal serta maksimal dari alat pendingin. Pengambilan aplikasi tentang temperatur ini di dasarkan pada besarnya pengaruh temperatur yang tidak hanya sebagai *noise* pada dunia elektronika tapi juga pengaruh pada dunia industri.

Keuntungan dari sistem ini adalah komponen rangkaian yang banyak di pasaran yang harganya cukup terjangkau sehingga dalam penggunaannya efisiensi

biaya dapat dicapai, mudah dalam perawatan, temperatur dapat dipantau langsung pada layar LCD, penyetingan temperatur sesuai keinginan dengan memasukkan *set point* pada *keypad*, kemudahan dalam pengoperasian. Sistem pengontrolan pada alat yang dirancang adalah menggunakan sistem pengatur *ON/OFF* dan mikrokontroler digunakan sebagai pusat untuk kontrol proses.

1.2 Tujuan Penelitian

Tujuan dari penelitian ini antara lain:

1. Tujuan Umum :

Merupakan tujuan utama penulis sebagai syarat dalam menyelesaikan kurikulum Program D3 Jurusan Fisika Instrumentasi, Departemen Fisika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia.

2. Tujuan Khusus :

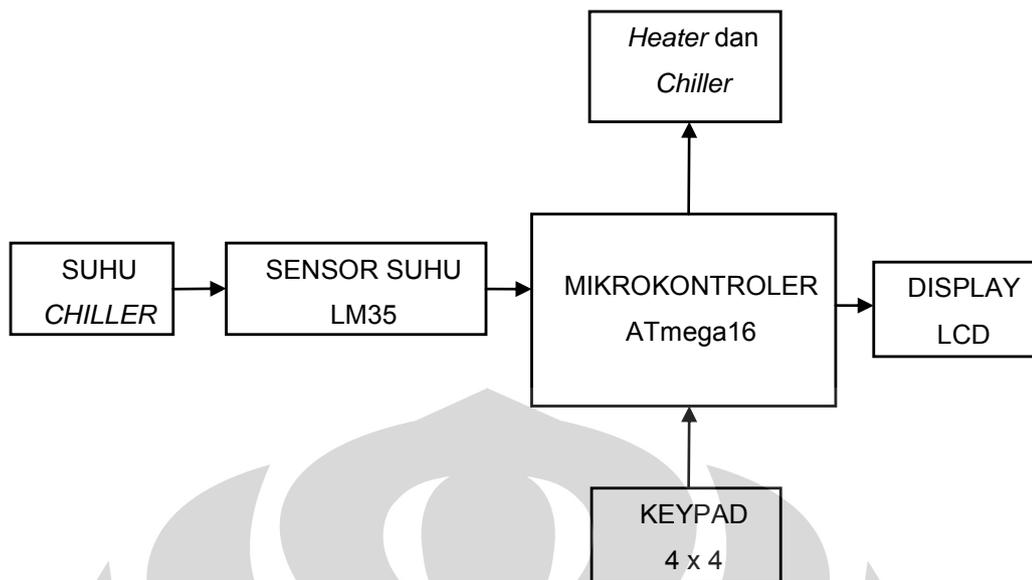
Tujuan khusus dari penelitian ini adalah untuk membuat suatu alat pendingin dengan temperatur yang terkendali yang dapat di-*set* temperaturnya melalui *keypad*.

1.3 Deskripsi Singkat

Sebagai salah satu aplikasi teknologi, pada tugas akhir ini dibuat “**Sistem Kendali Temperatur Alat Pendingin Temperatur Terkendali**” dengan maksud untuk mengendalikan temperatur alat pendingin secara terkendali serta efisien.

Cara kerja alat ini adalah untuk menghasilkan temperatur yang diinginkan, maka *keypad* ditekan untuk memasukan *set point* temperatur. Sehingga terjadi proses pendinginan/pemanasan, yang sebelumnya telah diatur temperaturnya.

Sistem ini terdiri dari sebuah sensor temperatur, sebuah mikrokontroler sebagai pengolah data, *heater* yang berfungsi sebagai pemanas dan *display* hasil dari pendeteksian tersebut. Gambar di bawah adalah bagian dari keseluruhan sistem :



Gambar 1.1 Bagan Keseluruhan Sistem

Dari bagan terlihat pada tangki terdapat kondensor sebagai penyalur dingin, sensor dipasang pada tangki untuk mengetahui besarnya temperatur air pada tangki, besar nilai temperatur air pada tangki kemudian dikirimkan ke mikrokontroler lalu nilai akan diolah apakah nilai sesuai dengan *set point* dari temperatur yang telah ditentukan, jika sesuai atau kurang dari *set point* maka program akan menghentikan kerja kompresor sedangkan jika temperatur pada sistem yang akan dikontrol melebihi dari *set point* yang telah ditentukan maka program akan menjalankan kompresor agar dapat menurunkan temperatur air pada tangki ke temperatur *set point*-nya.

1.4 Batasan Masalah

Pada proyek tugas akhir ini, pengerjaan proyek dibagi 2 bagian, antara lain yaitu:

1. Rancang Bangun Sistem Mekanik Alat Pendingin Temperatur Terkendali
2. Sistem Kendali Temperatur Alat Pendingin Temperatur Terkendali

Pembahasan ini hanya terbatas pada poin 2 saja yang penulis kerjakan yaitu sistem kendali temperatur dengan temperatur dapat dikendalikan yakni pembuatan program (*software*) dan *hardware* alat pendingin berbasiskan

mikrokontroler dengan pemrograman *on/off* AC 1, AC 2, *heater*, serta pompa dan pengendalian temperatur dengan menggunakan *keypad*.

1.5 Metodologi Penelitian

Metode yang digunakan untuk pengerjaan dan penulisan Tugas Akhir antara lain:

1. Studi Literatur

Metode ini digunakan untuk memperoleh informasi dan data yang berkaitan dengan penelitian yang penulis lakukan tentang teori – teori dasar sebagai sumber penulisan skripsi. Informasi dan pustaka yang berkaitan dengan masalah ini diperoleh dari literatur, penjelasan yang diberikan dosen pembimbing, rekan – rekan mahasiswa, internet, *data sheet*, dan buku – buku yang berhubungan dengan proyek tugas akhir penulis.

2. Perancangan dan Pembuatan Sistem

Pada tahapan perancangan ini, meliputi perancangan perangkat Mekanik dan perancangan perangkat keras/elektronik (*hardware*).

Pada perancangan mekanik, dibangun lilitan pipa untuk tempat keluar-masuk air (perpindahan panas-dinginnya air) tersebut, sedangkan pada perancangan elektronik (*hardware*) dirancang suatu rangkaian elektronik untuk menggerakkan motor dc, dan sensor yang terdapat pada sistem mekanik tersebut.

3. Uji Sistem

Untuk melihat keberhasilan dari rancang bangun dan sistem yang dibuat maka pada tahap ini dilakukan pengujian. Pengujian dilakukan dengan melihat hubungan dan komunikasi dari *hardware* dan *software* yang dirancang agar sesuai dengan tujuan dari perancangan.

4. Metode Analisis

Metode ini merupakan pengamatan terhadap data yang diperoleh dari pengujian sistem alat tersebut.

5. Penulisan Penelitian

Dari hasil pengujian dan pengambilan data kemudian dilakukan suatu analisa sehingga dapat diambil suatu kesimpulan. Dengan adanya beberapa saran juga dapat kita ajukan sebagai bahan perbaikan untuk penelitian lebih lanjut.

1.6 Sistematika Penulisan

Pada penulisan laporan Tugas Akhir ini, dapat dibuat urutan bab serta isinya secara garis besar. Diuraikan sebagai berikut :

BAB 1 PENDAHULUAN

Pada bab ini berisi tentang latar belakang, tujuan penelitian, pembatasan masalah yang dibahas dalam proyek tugas akhir ini, metodologi penelitian yang digunakan dalam penulisan laporan, dan sistematika penulisan.

BAB 2 TEORI DASAR

Teori dasar berisi landasan-landasan teori sebagai hasil dari studi literatur yang berhubungan dalam perancangan dan pembuatan alat program (*software*).

BAB 3 PERANCANGAN DAN CARA KERJA SISTEM ALAT

Pada bab ini akan dijelaskan secara keseluruhan sistem kerja dari sistem kendali alat pendingin dengan temperatur yang dapat dikendalikan.

BAB 4 PENGUJIAN ALAT DAN PENGAMBILAN DATA

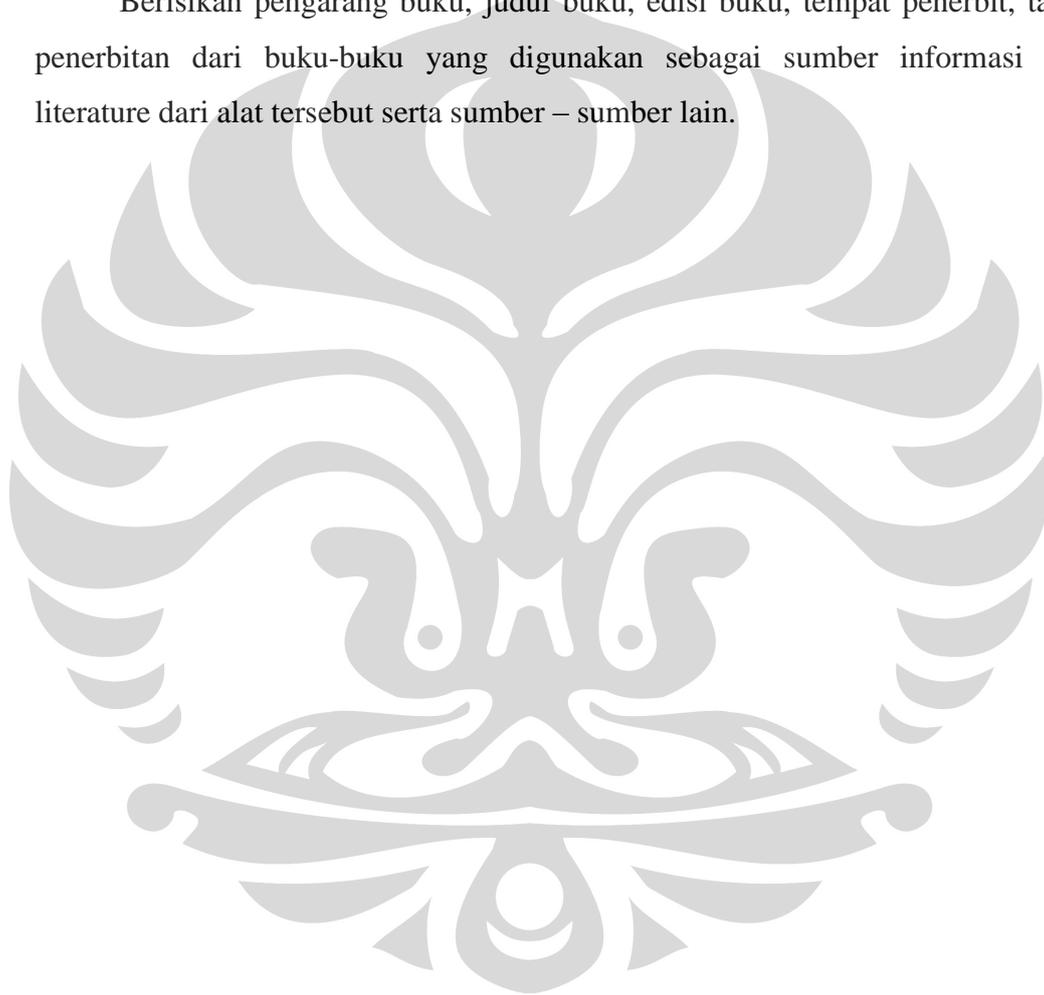
Bab ini berisi tentang unjuk kerja alat sebagai hasil dari perancangan sistem. Pengujian akhir dilakukan dengan menyatukan seluruh bagian-bagian kecil dari sistem untuk memastikan bahwa sistem dapat berfungsi sesuai dengan tujuan awal. Setelah sistem berfungsi dengan baik maka dilanjutkan dengan pengambilan data untuk memastikan kapabilitas dari sistem yang dibangun.

BAB 5 PENUTUP

Penutup berisi kesimpulan yang diperoleh dari pengujian sistem dan pengambilan data selama penelitian berlangsung, selain itu juga penutup memuat saran untuk pengembangan lebih lanjut dari penelitian ini baik dari segi perangkat keras (*hardware*) dan program (*software*).

DAFTAR ACUAN

Berisikan pengarang buku, judul buku, edisi buku, tempat penerbit, tahun penerbitan dari buku-buku yang digunakan sebagai sumber informasi atau literature dari alat tersebut serta sumber – sumber lain.



BAB 2

TEORI DASAR

Pada bab ini, penulis akan membahas mengenai teori dasar berisi landasan–landasan teori sebagai hasil dari studi literatur yang berhubungan dalam perancangan serta pembuatan program (*software*). Dimana mikrokontroler yang nantinya akan menjadi pusat kendali yang mengendalikan temperatur pada pendingin dengan memasukan nilai temperatur melalui *keypad* yang nantinya sensor pada pendingin akan membaca serta melakukan perintah sesuai dari masukan yang diterima, sehingga dengan begitu kita dapat mengontrol yang nantinya tertampil pada *display* LCD.

2.1 Temperatur

Temperatur adalah besaran yang menyatakan tingkat panas atau dinginnya suatu benda. Perpindahan panas dapat terjadi karena adanya beda temperatur antara dua bagian benda. Panas akan mengalir dari benda yang bertemperatur tinggi ke benda yang bertemperatur rendah akibat adanya pergerakan molekul yang saling berinteraksi melalui tumbukan.

Apabila temperatur naik maka molekul bergerak lebih cepat sehingga energi kinetik juga semakin meningkat. Jadi energi kinetik dari gerak molekular meningkat seiring dengan peningkatan temperatur. Semakin besar temperatur suatu material, maka semakin besar kecenderungan bagi molekul-molekul material tersebut untuk saling terlempar menjauh seperti yang terjadi pada perubahan fasa cair menjadi gas. Bila temperatur menurun, maka energi kinetik menurun.

Cara yang paling sederhana untuk membedakan benda-benda panas dari benda-benda dingin adalah melalui sentuhan. Hal ini merupakan sebuah prosedur yang sangat subjektif untuk menentukan temperatur sebuah benda dan sudah tentu tidak dapat digunakan untuk tujuan-tujuan ilmiah sehingga dibuatlah alat untuk pengukur temperatur dengan akurat atau yang lebih dikenal dengan nama Termometer.

Termometer merupakan sebuah alat untuk mengukur temperatur. Temperatur mengukur seberapa panasnya sebuah benda, yang sebanding dengan rata-rata energi kinetik (energi gerak) molekul dalam benda tersebut. Alat pengukur temperatur sebenarnya sudah dikenal orang sejak lama, akan tetapi orang pertama yang dianggap sebagai penemunya atau penciptanya adalah Galileo pada tahun 1593. Termometer Galileo adalah alat ukur temperatur berdasarkan perbedaan massa jenis zat cair yang ada didalam labu.

2.2 Sensor dan Sinyal

Sensor merupakan suatu alat yang dapat menerima sinyal atau rangsangan dan mengubahnya menjadi besaran listrik atau sinyal sehingga dapat dianalisa dalam rangkaian listrik tertentu. Rangsangan ini berasal dari lingkungan atau berada di luar sistem sensor, rangsangan ini dapat berupa besaran-besaran fisika. Dalam pengertian secara umumnya sensor merupakan pengubah besaran non-elektrik ke besaran elektrik.

Sensor merupakan suatu piranti pengindra kualitas fisis, mekanik atau optis yang ditransformasikan menjadi tegangan atau arus listrik. Besaran elektrik ini berupa sinyal yang dapat diolah, disalurkan, dikuatkan, dan dimodifikasi dengan menggunakan rangkaian elektronik. Sinyal keluaran sensor dapat berupa tegangan, arus, atau muatan.

Pengertian sensor ini harus dibedakan dengan pengertian transduser, transduser adalah pengubah suatu bentuk energi ke dalam bentuk energi yang lain, transduser bukan digunakan sebagai sensor, tapi sebagai aktuator, misalnya loudspeaker yang mengubah sinyal listrik menjadi energi suara. Sensor tidak dapat bekerja sendiri karena sensor merupakan bagian dari sistem yang lebih besar yang digabungkan dengan detektor lain, seperti pengolah sinyal, pengolah sinyal, alat memori, penyimpanan data, dan aktuator. Sensor terdiri atas dua jenis, yaitu pasif dan aktif. Sensor pasif adalah sensor yang langsung dapat menghasilkan sinyal listrik yang diakibatkan oleh rangsangan luar, yang mana energi rangsangan *input* diubah oleh sensor menjadi energi *output* tanpa membutuhkan sumber daya tambahan. Sedangkan sensor aktif adalah sensor yang memerlukan daya luar untuk pengoperasiannya, daya luar ini disebut sinyal eksitasi, sinyal ini diubah

oleh sensor untuk menghasilkan sinyal keluaran. Prinsip kerja sensor adalah melakukan pengindraan temperatur lingkungan kemudian meresponnya menjadi sinyal listrik dalam bentuk tegangan *analog*.

2.3 IC LM35

Sensor temperatur adalah alat yang digunakan untuk merubah besaran panas menjadi besaran listrik yang dapat dengan mudah dianalisis besarnya. Pembuatan sensor temperatur tergantung pada material penyusunnya yang memiliki besaran yang berubah terhadap perubahan temperatur.

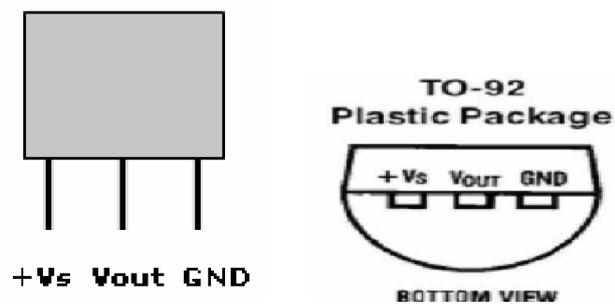
Metode pembuatan sensor temperatur secara garis besar dibedakan menjadi 2 yaitu:

1. Metode bahan logam, yaitu logam akan bertambah besar hambatannya terhadap arus listrik jika panasnya bertambah.
2. Menggunakan bahan semikonduktor, yaitu bahan semikonduktor mempunyai sifat terbalik dari logam, semakin besar temperatur maka nilai hambatan akan semakin turun.

Pada masa sekarang ini bidang elektronika mengalami kemajuan yang sangat pesat termasuk perancangan alat, demikian juga untuk mengetahui perubahan temperatur dari bernilai *analog* menjadi *digital* yang kemudian diolah oleh mikrokontroler dan ditampilkan pada sebuah LCD sehingga pemantauan terhadap proses dapat dilakukan dengan lebih mudah.

IC LM35 adalah sensor temperatur yang terkemas dalam bentuk integrated circuit, merupakan komponen elektronika yang memiliki fungsi untuk mengubah besaran temperatur menjadi besaran listrik dalam bentuk tegangan, dimana *output* tegangan keluarannya sangat linier. Sensor temperatur IC LM35 dapat diaplikasikan pada berbagai rangkaian elektronika, antara lain untuk pengukuran dan pengontrolan temperatur dengan mengkonversi temperatur ketegangan dan selanjutnya mengkonversi tegangan *analog* kedalam bentuk *digital*. IC LM35 biasanya memiliki banyak variasi jenis tergantung *range* temperatur yang biasa dioperasikan seperti :

1. LM35A memiliki *range* temperatur -55°C sampai dengan 150°C .
2. LM35CA/C memiliki *range* temperatur -40°C sampai dengan 110°C .
3. LM35D memiliki *range* temperatur 0°C sampai dengan 100°C .



Gambar 2.1 Bentuk LM35^[1]

Gambar 2.1 di atas menunjukkan bentuk dari LM35 tampak depan dan tampak bawah. tiga pin LM35 menunjukkan fungsi masing-masing pin. Pin satu berfungsi sebagai sumber tegangan kerja dari LM35, pin dua atau tengah digunakan sebagai tegangan keluaran atau Vout. Secara prinsip sensor akan melakukan penginderaan pada saat perubahan temperatur, setiap perubahan temperatur 1 °C akan menunjukkan perubahan tegangan sebesar 10 mV.

2.4 Mikrokontroler

Mikrokontroler AVR merupakan pengontrol utama standar industri dan riset saat ini. Hal ini dikarenakan berbagai kelebihan yang dimilikinya dibandingkan mikroprosesor, yaitu murah, dukungan *software* dan dokumentasi yang memadai, dan memerlukan komponen pendukung yang sangat sedikit. Salah satu tipe mikrokontroler AVR untuk aplikasi standar yang memiliki fitur memuaskan ialah Atmega16. Anda juga tidak boleh tergantung hanya dengan satu tipe mikrokontroler saja, karena berbagai produsen mikrokontroler terkenal seperti ATMEL, Microchip, Maxim, dan Philips menciptakan mikrokontroler dengan keunggulan masing-masing. Untuk menentukan mikrokontroler mana yang ingin Anda gunakan, ada baiknya Anda tentukan spesifikasi yang Anda inginkan, lalu pilih mikrokontroler yang sesuai dengan spesifikasi Anda tersebut.

Beberapa faktor penting yang menjadi pertimbangan antara lain :

- Harga mikrokontroler
- Ukuran memori mikrokontroler
- Fitur ADC, *Timer*, dan fasilitas komunikasi I²C

- Fitur utama lain : seperti pengontrol utama akuisisi data, penampil LCD, dan lainnya.
- Kecepatan eksekusi instruksi
- Fasilitas *single cycle Hardware Multiplier* (untuk aplikasi DSP)
- Dukungan *software* yang dapat digunakan

2.4.1 Mikrokontroler AVR Atmega16

Mikrokontroler AVR (*Alf and Vogard's Risc Processor*) standar memiliki arsitektur 8 bit, dimana semua instruksi dikemas dalam kode 16-bit, dan sebagian besar instruksi dieksekusikan dalam 1 siklus *clock*. AVR berteknologi RISC (*Reduced Instruction Set Computing*). AVR dapat dikelompokkan menjadi empat kelas, yaitu keluarga *Attiny*, keluarga *AT90SXX*, keluarga *Atmega*, dan *AT86RFXX*. Pada dasarnya, yang membedakan masing-masing kelas adalah memori, peripheral, dan fungsinya.

Untuk mikrokontroler AVR yang berukuran lebih kecil, Anda dapat mencoba *Atmega8* atau *Attiny2313* dengan ukuran *Flash Memory* 2KB dengan dua *input analog*. Mikrokontroler AVR yang perlu Anda kuasai. Selain *Atmega16*, sangat direkomendasikan untuk mencoba *Atmega32* dan *Atmega128*. Selain itu, kuasai juga jenis mikrokontroler lain produksi Maxim seperti *DSC80C400* dan *MAXQ2000*.

2.4.2 Fitur Atmega16

Di dalam mikrokontroler *Atmega 16* sudah terdiri dari :

- Saluran I/O ada 32 buah, yaitu *port A*, *Port B*, *Port C*, dan *Port D*
- ADC (*Analog to Digital Converter*) 10 bit sebanyak 8 *channel*
- Tiga buah *Timer/ Counter* dengan kemampuan perbandingan
- CPU yang terdiri 32 buah *register*
- 131 instruksi andal yang umumnya hanya membutuhkan 1 siklus *clock*
- *Watchdog timer* dengan osilator *internal*
- Dua buah *timer/ counter* 8 bit
- Satu buah *timer/ counter* 16 bit
- *Internal SRAM* sebesar 1KB

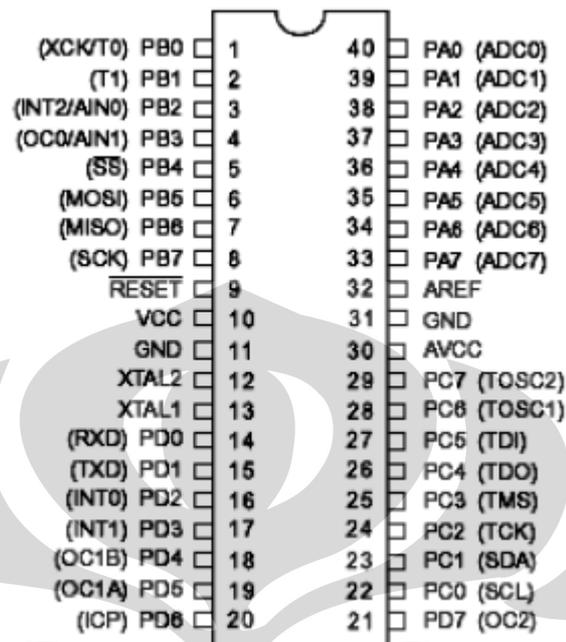
- *Memory Flash* sebesar 16KB dengan kemampuan *Read While Write*
- Unit interupsi *internal* dan *eksternal*
- *Port* antarmuka SPI
- EEPROM sebesar 512 byte yang dapat deprogram saat operasi
- Antarmuka komparator *analog*
- *Channel PWM*
- 32 x 8 *general purpose register*
- Hampir mencapai 16 MIPS pada kristal 16MHz
- *Port USART programmable* untuk komunikasi serial

2.4.3 Konfigurasi Atmega16

Berikut penjelasan umum susunan kaki Atmega16 :

- VCC merupakan pin masukan positif catu daya. Setiap peralatan elektronika *digital* tentunya butuh sumber catu daya yang umumnya sebesar 5V, itulah sebabnya di PCB kit mikrokontroler selalu ada IC regulator 7805
- GND sebagai pin *Ground*
- *Port A* (PA0 . . PA7) merupakan pin I/O dua arah dan dapat diprogram sebagai pin masukan ADC
- *Port B* (PB0 . . PB7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu *timer/ counter*, komparator *analog*, dan SPI
- *Port C* (PC0 . . PC7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu TWI, komparator *analog*, dan *Timer Osilator*
- *Port D* (PD0 . . PD7) merupakan dua arah dan pin fungsi khusus, yaitu komparator *analog*, interupsi eksternal, dan komunikasi serial
- *Reset* merupakan pin yang digunakan untuk me-*reset* mikrokontroler
- XTAL1 dan XTAL2 sebagai pin masukan *clock eksternal*. Suatu mikrokontroler membutuhkan sumber detak (*clock*) agar dapat mengeksekusi instruksi yang ada di memori. Semakin tinggi nilai kristalnya, maka semakin cepat mikrokontroler tersebut
- AVCC sebagai pin masukan tegangan untuk ADC

- AREF sebagai pin masukan tegangan referensi



Gambar 2.2 Konfigurasi Atmega16 ^[2]

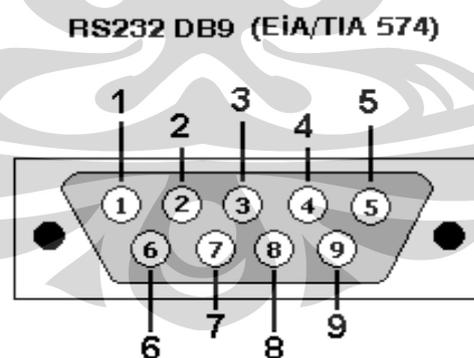
2.4.4 Komunikasi Serial Pada Atmega16

Komunikasi serial adalah pengiriman data secara serial (data dikirim satu persatu secara berurutan) sehingga komunikasi serial jauh lebih lambat daripada komunikasi yang paralel. Karena peralatan berkomunikasi menggunakan transmisi serial sedangkan data dikomputer diolah secara paralel, oleh karena itu harus dikonversikan dahulu ke bentuk paralel. Jika menggunakan perangkat keras hal ini bisa dilakukan oleh *Universal Asynchronous Receiver Transmitter (USART)*, yang membutuhkan perangkat untuk menanganinya. Komunikasi serial merupakan salah satu cara untuk mengkomunikasikan data dari suatu peralatan ke peralatan lain dengan cara menggunakan data serial, misalnya mengkomunikasikan antara HP dengan Mikrokontroler, HP dengan PC, Printer dengan PC, dll. Pada PC komunikasi serial RS232 dapat dilakukan melalui serial *port* (COM *port*). Komunikasi data serial dapat dilakukan dengan mempresentasikan data dalam bentuk level “1” atau “0”. Kelebihan komunikasi serial adalah jangkauan panjang kabel yang lebih jauh dibanding paralel karena serial *port* mengirimkan logika 1 dengan kisaran tegangan -3 Volt hingga -25 Volt dan logika 0 sebagai +3 Volt

hingga +25 Volt sehingga kehilangan daya karena panjang kabel bukan masalah utama. Selain itu juga komunikasi serial *port* bersifat asinkron sehingga sinyal detak tidak dikirim bersama data. Setiap word disinkronkan dengan *start* bit dan sebuah *clock internal* di kedua sisi menjaga bagian data saat pewaktuan (*timing*).

Perangkat keras pada komunikasi serial dibagi menjadi dua kelompok, yaitu *Data Communication Equipment* (DCE) dan *Data Terminal Equipment* (DTE). Modem salah satu contoh DCE, sedangkan terminal yang terdapat di komputer salah satu contoh dari DTE. Null modem digunakan untuk menghubungkan dua buah data DTE, hal ini biasanya dilakukan untuk mentransfer file antar komputer .

Komunikasi serial RS232 merupakan komunikasi asinkron sehingga sinyal *clock* tidak dikirim bersamaan dengan data. Setiap data disinkronisasikan dengan menggunakan *start* bit dan *clock internal* pada setiap bit. *Port* RS232 pada komputer harus memenuhi standar RS232. Agar level tegangan sesuai dengan tegangan TTL/CMOS diperlukan RS232 level konverter. IC yang banyak digunakan untuk ini adalah MAX-232. Konfigurasi slot DB-9 *female* adalah sebagai berikut :



Gambar 2.3 Konfigurasi Slot Konektor Serial DB-9 ^[3]

Tabel 2.1 Daftar Nama Pin dan Sinyal Konektor Serial DB-9 ^[3]

Pin DB-9	Nama Pin	Keterangan
1	<i>DCD</i>	<i>Data Carrier DetectData Port (DP0 - DP9)</i>
2	<i>RD</i>	<i>Receive Data (a.k.a RxD, Rx)</i>
3	<i>TD</i>	<i>Transmit Data (a.k.a TxD, Tx)</i>
4	<i>DTR</i>	<i>Data Terminal Ready</i>
5	<i>SGND</i>	<i>Ground</i>
6	<i>DSR</i>	<i>Data Set Ready</i>
7	<i>RTS</i>	<i>Request To Send</i>
8	<i>CTS</i>	<i>Clear To Send</i>
9	<i>RI</i>	<i>Ring Indicator</i>

2.5 BASCOM AVR

BASCOM-AVR adalah program *Basic Compiler* berbasis Windows untuk mikrokontroler keluarga AVR seperti Atxmega, Atmega, Attiny, AT90USBXXXX, AT90SXXXX, , dan yang lainnya. BASCOM-AVR merupakan pemrograman dengan bahasa tingkat tinggi *basic* yang dikembangkan dan dikeluarkan oleh MCS Elektronik.

Kita akan membahas penggunaan karakter, tipe data, variabel, konstanta, operasi-operasi aritmatika dan logika, array, dan kontrol program.

2.5.1 Instruksi dalam BASCOM AVR

Alv and Vegard's Risc Processor atau biasa di singkat AVR merupakan mikrokontroler *Reduced Instruction Set Computer* (RISC) 8 bit, instruksi yang dimiliki terbatas, tetapi memiliki fasilitas yang lebih banyak. Karena RISC inilah sebagian besar kode instruksinya dikemas dalam satu siklus *clock*. AVR adalah jenis mikrokontroler yang paling sering digunakan dalam bidang elektronika dan instrumentasi.

Secara umum, AVR dapat dikelompokkan dalam 4 kelas. Pada dasarnya yang membedakan masing – masing kelas adalah memori, peripheral dan fungsinya. Keempat kelas tersebut adalah keluarga Attiny, keluarga AT90Sxx, keluarga Atmega dan AT86RFxx.

Secara keseluruhan BASCOM AVR mempunyai beberapa macam instruksi, yang dibentuk dengan menkombinasikan dan operan. Penjelasan berikut adalah beberapa instruksi yang dipakai dalam pemrograman pada pembuatan tugas akhir ini :

- **DO – LOOP**

Instruksi **Do – Loop** berfungsi untuk mengulangi suatu blok statemen sampai kondisi adalah benar atau kita juga dapat keluar dari suatu **Do - Loop** dengan statemen **Exit Do**. **Do - Loop** selalu dilakukan sedikitnya sekali. Sintaksis instruksi **Do - Loop** adalah sebagai berikut :

Do

Statements

Loop [Until expression]

Instruksi Do – Loop akan terus menerus mengerjakan suatu blok statemen yang berada di dalam suatu Do – Loop sampai kondisi terpenuhi. misalnya jika kita ingin menggerakkan sebuah motor dc sampai motor dc itu berhenti ketika terkena sebuah sensor Limit Switch (aktif *high*).

Contoh programnya adalah :

Do

M1 = 1 ; variabel M1 berlogika 1 atau *high*

M2 = 0 ; variabel M2 berlogika 0 atau *low*

Loop Until Ls = 1 ; mengulang blok statemen sampai variabel Ls = 1

Maksud variabel nilai M1 dan M2 diberi logika I/O adalah untuk mengkondisikan gerak motor ke kanan atau ke kiri.

- **IF – THEN**

Instruksi **If – Then** digunakan jika hanya terdapat satu syarat yang digunakan untuk menguji apakah suatu instruksi akan diproses atau tidak, jika syarat terpenuhi maka instruksi tersebut akan diproses terlebih dahulu sebelum mekanjutkan ke eksekusi baris instruksi selanjutnya, jika syarat

tidak terpenuhi maka program akan langsung melanjutkan dengan mengeksekusi baris instruksi selanjutnya. Sintaksis instruksi **If – Then** adalah sebagai berikut :

If expression (syarat) **Then**

statements (instruksi)

End If

- **IF – THEN – ELSE**

Instruksi **If – Then – Else** digunakan jika terdapat satu syarat yang digunakan untuk memutuskan alternatif instruksi mana yang akan diproses, jika syarat terpenuhi maka instruksi akan diproses lebih dahulu sebelum mengeksekusi baris instruksi yang selanjutnya, jika syarat tidak terpenuhi maka program akan mengeksekusi program instruksi-2 sebelum melanjutkannya dengan mengeksekusi baris instruksi selanjutnya. Sintaksis instruksi **If – Then - Else** adalah sebagai berikut :

If expression (syarat) **Then**

statements (instruksi)

Else

Statements-2 (instruksi-2)

End If

contoh penggunaan instruksi **If – Then – Else** :

If $x > y$ **then**

LCD “ X Lebih besar dari Y”

Else

LCD “ X Lebih kecil dari Y”

End If

- **IF – THEN – ELSEIF**

Instruksi **If – Then – Elseif** digunakan jika terdapat beberapa syarat yang digunakan untuk memutuskan alternatif instruksi mana yang akan diproses, misalkan terdapat beberapa pilihan huruf, “a”, “b”, dan “c”. jika syarat pada “a” terpenuhi, maka instruksi akan diproses. Namun jika syarat pada “a” tidak terpenuhi, maka instruksi yang akan diproses

adalah instruksi pada syarat “b” atau “c”. begitu pula dengan “b”, jika syarat pada “b” terpenuhi, maka instruksi akan diproses. Namun jika syarat pada “b” tidak terpenuhi, maka instruksi yang akan diproses adalah instruksi pada syarat “a” atau “c”. begitu pula sebaliknya. Sintaksis instruksi **If – Then - Elseif** adalah sebagai berikut :

If expression (syarat) **Then**

statements (instruksi)

Elseif expression (syarat) **Then**

Statements-2 (instruksi-2)

Elseif expression (syarat) **Then**

Statements-3 (instruksi-3)

•
•
•
•

Elseif expression (syarat) **Then**

Statements-n (instruksi-n)

End If

contoh penggunaan instruksi **If – Then – Elseif** :

If A = 0 **then**

LCD “ Nilai A adalah 0”

Elseif A = 1 **then**

LCD “ Nilai A adalah 1”

Elseif A = 2 **then**

LCD “ Nilai A adalah 2”

End If

- **FOR – NEXT**

Instruksi **for – next** adalah sebuah instruksi untuk perulangan yang hanya menggunakan sebuah variabel counter. Untuk perulangan yang bersifat menambahkan, kita mesti menggunakan **To**, sedangkan untuk perulangan yang bersifat mengurangi, kita harus menggunakan **Step** yang nilainya negatif. Sintaksis instruksi **for - next** adalah sebagai berikut:

For var = start **To** end [**Step** value]
 statements (instruksi)

Next var

contoh penggunaan instruksi **For - Next** :

For A = 1 **To** 10 **Step** 1

LCD "This is A " ; A

Next A

2.6 LCD

LCD berfungsi untuk menampilkan data yang digunakan. LCD sangat mudah untuk pembacaan karakter yang cukup banyak bila dibandingkan dengan 7 segment. Pada *software* Bascom sangat mudah digunakan karena sudah memiliki konfigurasi LCD. LCD yang digunakan adalah 4 x 20, mempunyai tampilan sebanyak 20 karakter 4 baris. LCD yang umum, ada yang panjangnya hingga 40 karakter dimana kita menggunakan DDRAM untuk mengatur tempat penyimpanan karakter tersebut.

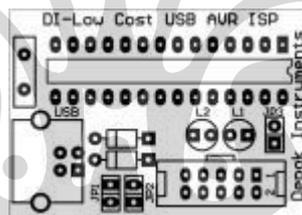
2.7 Downloader

USB AVR *Downloader* adalah modul pengunduh (*downloader*) skrip program dari PC/Notebook ke IC mikrokontroler AVR yang memiliki fitur pemrograman ISP (*In System Programming*). ISP (*In-System Programmable*) adalah salah satu metoda memprogram mikrokontroler dengan lebih mudah dan mengasyikan. Mikrokontroler yang akan diprogram dinamakan mikrokontroler target, sedangkan pemrogramnya ada yang harus menggunakan mikrokontroler lagi atau tidak tergantung metodenya. Banyak *programmer* atau *downloader* yang biasa didapat di internet atau lainnya. Untuk memprogram mikrontroler secara ISP, ada beberapa metoda, yaitu :

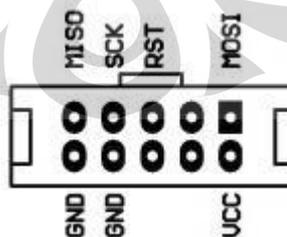
- Metoda *paralel*, dimana antarmukanya dengan komputer melalui *paralel port / printer port*, dimana dengan tambahan sedikit komponen *hardware* ditambah program *downloader*-nya, Anda bisa memprogram mikrokontroler.

Kelebihannya, tidak memerlukan mikrokontroler lain ditambah *firmware* (program dalam mikrokontroler-*programmer*). Sedangkan kelemahannya, sering tidak kompatibel pada komputer tertentu, atau saat ini PC sudah tidak punya *parallel port* lagi.

- Antarmuka *serial*;
 - USB, *serial* bekecepatan tinggi ini saat ini menjadi standar antarmuka PC dengan peripheral luarnya. *Programmer* melalui USB dimungkinkan dengan membangun *programmer* yang sedikit lebih rumit, bisa dilengkapi dengan mikrokontroler yang bertindak sebagai *programmer* (jadi mikrokontroler memprogram mikrokontroler), berisi *firmware* yang rumit, karena USB memerlukan protokol tertentu untuk komunikasinya. Kelebihannya adalah semua PC punya USB. Sedangkan kekurangannya, *firmware* rumit, diperlukan konversi sinyal USB ke data *serial digital*.



Gambar 2.4 Layout Posisi Komponen USB AVR ISP (USB AVR Downloader)^[4]



Gambar 2.5 Keterangan Pin^[4]

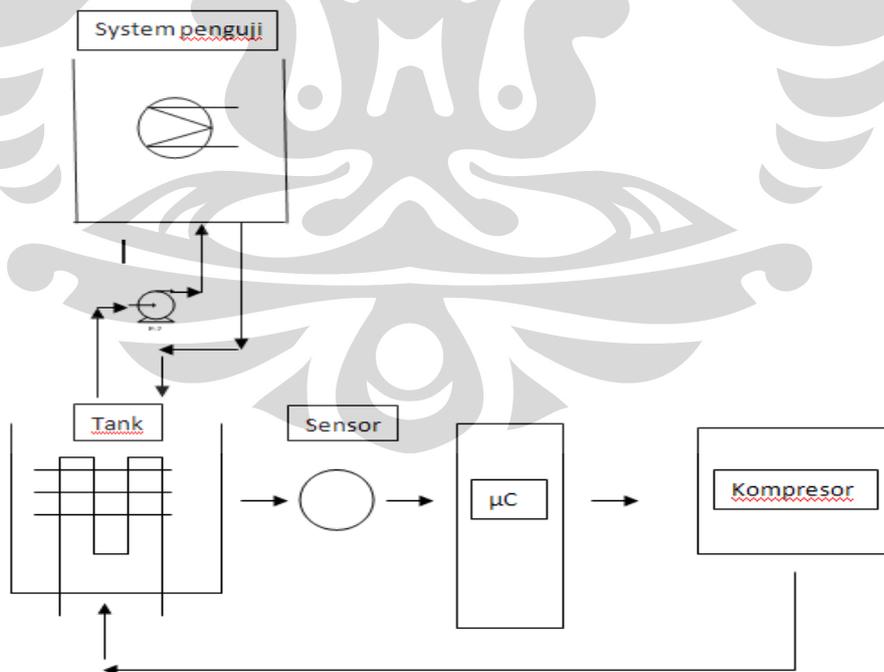
BAB 3 PERANCANGAN DAN CARA KERJA SISTEM

Bab ini akan membahas mengenai perancangan sistem beserta cara kerja dari masing-masing *software* dan *hardware* yang digunakan penulis dalam penyusunan alat “Sistem Kendali Temperatur Alat Pendingin Temperatur Terkendali”.

3.1 Perancangan Kerja Sistem

Sistem kendali alat pendingin dengan temperatur yang dapat dikendalikan berfungsi untuk mengendalikan temperatur pada sebuah wadah atau tangki air agar temperaturnya terjaga sesuai yang diinginkan.

Dan untuk mempermudah dalam perancangan dan pemahaman sistem rangkaian, maka perancangan dibuat berdasarkan blok. Dimana tiap blok mempunyai fungsi dan kerja tertentu, blok diagram yang satu dengan yang lain berhubungan dan saling mendukung hingga terbentuk suatu sistem rangkaian yang mempunyai satu fungsi dan kerja khusus.



3.1 Gambar Diagram Blok Sistem

Dari blok diagram di atas terdapat *keypad* sebagai data *tranceiver* untuk mengirimkan nilai *Set point (SP)* yang berupa bilangan ke dalam mikrokontroler dan menampilkan nilai temperatur dari sensor. Proses pengendalian *chiller, heater* dan temperaturnya terjadi di dalam mikrokontroler. Pada dasarnya, alat ini dikendalikan oleh sebuah sistem pengendali yaitu pengendali temperatur. Sistem kendali yang penulis rancang akan mengukur temperatur pada suatu penampung (air) dengan menggunakan sensor LM35 untuk mengukur temperatur dalam satuan derajat Celcius. Ketika sensor telah mendapatkan temperatur yaitu dengan menerjemahkan sifat fisis temperatur menjadi sinyal listrik yaitu perubahan tegangan *output* sensor, maka kemudian *output* ini dibaca oleh ADC *internal* dari mikrokontroler Atmega16 dan kemudian data dikalkulasikan dengan rumusan tertentu sehingga pada tahap berikutnya sistem dapat menentukan apakah temperatur air sudah sesuai atau belum. Jika temperatur terlalu panas maka kompressor AC (*Chiller*) akan diaktifkan. Jika temperatur sebenarnya hampir mendekati *set point*, maka AC 1 akan mati. Tetapi, apabila temperatur sebenarnya sama dengan nilai *set point*, maka AC akan mati secara bersamaan (mati dua-duanya). Nilai temperatur yang sedang di pantau oleh sensor akan ditampilkan ke *display LCD (Liquid Crystal Display)*.

3.2 Perancangan Perangkat Keras (*Hardware*)

Di dalam pembuatan alat ini perangkat kerasnya (*hardware*) terdapat beberapa bagian, yakni rangkaian *minimum system*, serta rangkaian *keypad*.

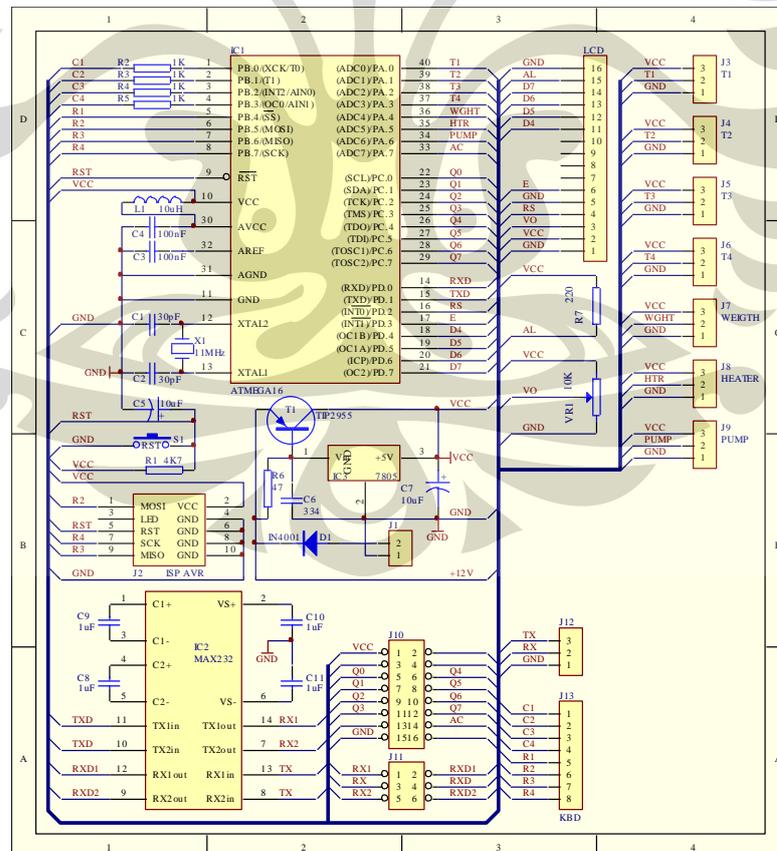
3.2.1 Rangkaian Sistem Minimum Mikrokontroler Atmega16

Rangkaian minimum sistem adalah rangkaian yang digunakan agar mikrokontroler dapat bekerja. Dalam proses pembuatan model pengontrolan objek yang diinginkan IC mikrokontroler yang digunakan adalah Atmega16. Alasan penulis menggunakan IC ini karena IC ini sangat mudah dipahami, memori lebih besar, banyak terdapat dipasaran dan memiliki ADC sehingga tidak perlu membuat rangkaian ADC.

Rangkaian minimum sistem yang digunakan penulis pada tugas akhir ini yaitu menggunakan mikrokontroler AVR Atmega 16. Mikrokontroler adalah suatu

piranti yang digunakan untuk mengolah data-data biner (*digital*) yang didalamnya merupakan gabungan dari rangkaian-rangkaian *elektronik* yang dikemas dalam bentuk suatu *chip* (IC). Pada umumnya mikrokontroler terdiri dari bagian-bagian sebagai berikut: Alamat (*address*), Data, Pengendali, Memori (RAM atau ROM), dan bagian *input-Output*. AVR merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur RISC (*Reduced Instruction Set Computer*). Hampir semua instruksi dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 *register general-purpose*, *timer/counter* fleksibel dengan *mode compare*, *interrupt internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, dan *mode power saving*.

Blok diagram pengendali temperatur dapat dilihat pada gambar di bawah. Sebagai otak dari sistem ini menggunakan mikrokontroler produk dari atmel keluarga AVR seri Atmega 16. Rangkaian ini dibutuhkan agar dapat mengisi (*me-download*) program yang telah dibuat. Program tersebut meliputi program pengendalian temperatur serta pengiriman data komunikasi serial.



Gambar 3.2 Blok Diagram Pengendali Temperatur

Sistem minimum ini dilengkapi dengan 4 buah port (port A – port D) dan masing-masing port memiliki cara fungsi sendiri. Port A akan tersambung dengan AC1, AC2, *Heater* serta pompa, port B tersambung untuk downloader, port C akan tersambung rangkain *keypad*, dan port D akan tersambung dengan rangkaian LCD.

Port A digunakan sebagai pin masukan ADC. Pada port A yang digunakan adalah port A.0 hingga port A.4. Port A.0 dihubungkan dengan AC1, Port A.1 dihubungkan dengan AC2, Port A.2 dihubungkan dengan *heater*, port A.3 dihubungkan dengan pompa, dan port A.4 digunakan sebagai ADC Internal untuk LM35.

Pin 12 dan 13 dihubungkan ke XTAL 1 MHz dan dua buah kapasitor 30 pF. XTAL ini akan mempengaruhi kecepatan mikrokontroler Atmega16 dalam mengeksekusi setiap perintah dalam program. Pin 9 merupakan masukan *reset* (aktif rendah). Pulsa transisi dari tinggi ke rendah akan me-*reset* mikrokontroler ini.

Untuk men-*download* file heksadesimal ke mikrokontroler, Mosi, Miso, Sck, *Reset*, Vcc dan Gnd dari kaki mikrokontroler dihubungkan ke J2. J2 sebagai konektor yang akan dihubungkan ke *ISP Programmer*.

Kaki Mosi, Miso, Sck, *Reset*, Vcc dan Gnd pada mikrokontroler terletak pada kaki 6, 7, 8, 9, 10 dan 11. Apabila terjadi keterbalikan pemasangan jalur ke *ISP Programmer*, maka pemrograman mikrokontroler tidak dapat dilakukan karena mikrokontroler tidak akan bisa merespon.

3.2.2 Konsep I/O pada mikrokontroler AVR Atmega16

Pemrograman I/O pada mikrokontroler merupakan dasar dari prinsip pengontrolan berbasis mikrokontroler, dimana orientasi dari penerapan mikrokontroler ialah untuk mengendalikan suatu sistem berdasarkan informasi *input* yang diterima, lalu diproses oleh mikrokontroler dan dilakukan aksi pada bagian *output* sesuai program yang telah ditentukan sebelumnya pada gambar di atas, terdapat empat buah port, yaitu PA, PB, PC, dan PD yang semuanya dapat diprogram sebagai *input* ataupun *output*. Jika dilihat lebih detail lagi pada pemroses mikrokontroler ini, terdapat unit CPU utama untuk memastikan

eksekusi program. CPU juga dapat mengakses memori, melakukan kalkulasi, pengontrolan dan penanganan interupsi dengan menggunakan arsitektur Harvard (bus untuk memori dan program dan data terpisah) sehingga dihasilkan performa yang tinggi.

Hal ini di karenakan instruksi pada memori program dieksekusi dengan *single level pipelining*, dengan demikian pada saat sebuah instruksi dieksekusi, instruksi berikutnya dapat diakses dari memori program.

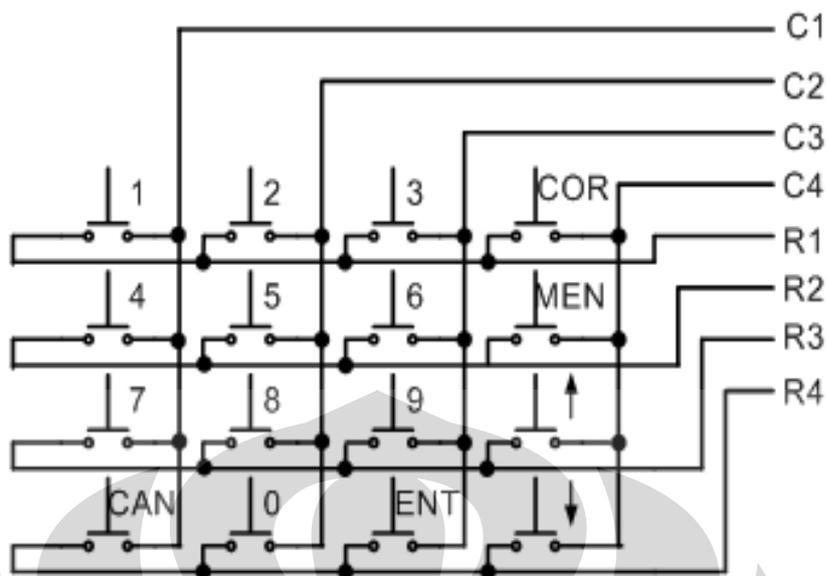
3.2.3 Rangkaian Keypad

Keypad merupakan perangkat *input* berupa saklar *push button* yang terdiri dari susunan baris dan kolom. Bentuk dan susunan *keypad* dapat dilihat pada gambar 3.3. Dasarnya *keypad* 4×4 adalah 16 tombol *push button* dalam matriks. *Interfacing keypad* 4×4 dengan pemindaian pola tidak mudah dilakukan.

Untuk mengetahui saklar mana yang ditekan, periksa baris dan kolom, Misalnya, jika saklar merupakan angka 1 yang ditekan, maka baris 1 dan kolom 1 (R1 dan C1) akan terhubung singkat. Agar mikrokontroler mendeteksi baris dan kolom mana yang terhubung, tekniknya ialah: sebagian port difungsikan sebagai *output* dan sebagian lagi sebagai *input*.

Pada gambar 3.3 terlihat bahwa C1-C4 difungsikan sebagai *output* dengan R1–R4 difungsikan sebagai *input*. *Scanning* dilakukan pada port sebagai *output*, yaitu dengan mengirimkan logika 0 pada kolom 1, dan pada kolom lainnya berlogika 1. Setelah itu mikrokontroler akan membaca *input*, jika semua *input* berlogika 1, maka tidak ada tombol yang ditekan.

Langkah kedua ialah mengirimkan logika 0 untuk kolom berikutnya (kolom 2, C2 = 0) dan kolom lainnya berlogika 1. Setelah itu mikrokontroler akan membaca *input*, jika semua *input* berlogika 1 maka tidak ada tombol yang ditekan. Begitu seterusnya untuk kolom 3 dan 4. Jika tidak ada tombol yang ditekan, maka prose *scanning* kembali ke kolom 1.



Gambar 3.3 Rangkaian Dasar Keypad 4x4 ^[5]

3.3 Perancang Perangkat Lunak (*Software*)

Pada alat ini menggunakan perangkat lunak (*software*) didalam pengendaliannya. *Software* yang digunakan adalah Bascom AVR dan pengambilan data secara manual dengan melihat data langsung pada *display* LCD karena rangkaian komunikasi pada minsis rusak dan tidak dapat mengirim data.

3.3.1 Proses Pada Keypad

Dapat dijelaskan bahwa pada awal proses, program akan menginisialisai variabel-variabel yang digunakan pada program *keypad* tersebut, pada keadaan awal ini LCD akan menampilkan tulisan utama. Setelah tampilan LCD, kita dapat menekan *keypad* A, B, C, D, atau “*”. Masing-masing *keypad* tersebut memiliki fungsi tersendiri dimana telah disesuaikan pada program pengendalian. *Keypad* A digunakan untuk *auto mode controller*, sedangkan *keypad* B digunakan *test mode controller*. Untuk *keypad* C serta D berfungsi sebagai *set point* dari temperatur serta daya *heater*. Selain itu, *keypad* “*” berfungsi untuk menjalankan program pengendalian. Sebagai contoh, ketika kita ingin mengendalikan program *test mode controller*, maka kita memberikan *set point* dengan menekan C (*set point* temperatur) dan D (*set point heater*) dengan memasukkan nilai parameter Temp dan *Heater Power*. Setelah memasukkan nilai-nilai *set point* tersebut, akan

disimpan setelah menekan tombol “#”. Ketika tombol “*” ditekan, maka seluruh proses kerja alat akan berjalan. Apabila alat sedang bekerja, perubahan masukan hanya dapat dilakukan perubahan temperatur dan *Heater Power*. Saat proses berjalan, data dikirimkan ke Bascom Atmega16 untuk ditampilkan pada LCD. Sehingga perubahan data saat proses berjalan dapat diketahui.

Berikut program Bascom *keypad* :

```

$regfile = "M16def.dat"
$crystal = 11059200
$baud = 19200

Config Lcd = 20 * 4           'LCD 4 x 20
Config Lcdpin = Pin , Db4 = Portd.4 , Db5 = Portd.5 , Db6 = Portd.6 , Db7 =
Portd.7 , E = Portd.3 , Rs = Portd.2
Config Kbd = Portb Debounce = 50

Dim Keypad As Byte
Dim Number As Byte
Dim Nilai As Long
Dim Ulang As Bit
Dim Balik As Bit

Main_program:
Cls
Locate 1 , 1
Lcd " PROGRAM UJI KEYPAD "
Locate 2 , 1
Lcd "  MATRIKS 4X4  "
Locate 3 , 1
Lcd "KEYPAD VALUE = " ; Keypad
Do
  Balik = 1
  Nilai = 0
  Do
    Keypad = Getkbd()
    If Keypad < 16 Then
      Select Case Keypad
        Case 0
          Number = 1
        Case 1
          Number = 4
        Case 2
          Number = 7
        Case 3
          Number = 14
        Case 4

```

```

    Number = 2
Case 5
    Number = 5
Case 6
    Number = 8
Case 7
    Number = 0
Case 8
    Number = 3
Case 9
    Number = 6
Case 10
    Number = 9
Case 11
    Number = 15
Case 12
    Number = 10
Case 13
    Number = 11
Case 14
    Number = 12
Case 15
    Number = 13
End Select
Ulang = 1
Do
    Keypad = Getkbd()
    If Keypad = 16 Then Ulang = 0
Loop Until Ulang = 0
If Number < 10 Then
    Nilai = Nilai * 10
    Nilai = Nilai + Number
Else
    If Number = 15 Then
        Balik = 0
    End If
End If
Locate 3 , 16
Lcd "  "
Locate 3 , 16
Lcd Nilai
End If
Loop Until Balik = 0

Waitms 50
Loop
End

```

3.3.2 Pengambilan Data

Sebuah proses untuk mengambil data, yaitu untuk mengambil data temperatur. Bila salah satu dari tombol perintah ditekan maka *lowerline* pada LCD akan berubah sesuai dengan tombol yang ditekan. Kemudian memasukkan nilai temperatur dan *Heater Power*. Dan sistem akan menunggu hingga tombol ”#” ditekan. Setelah tombol ”#” ditekan maka nilai tersebut tersimpan dalam eeprom dan sistem balik ke awal program.

3.3.3 Proses Pada Sistem Minimum Atmega16

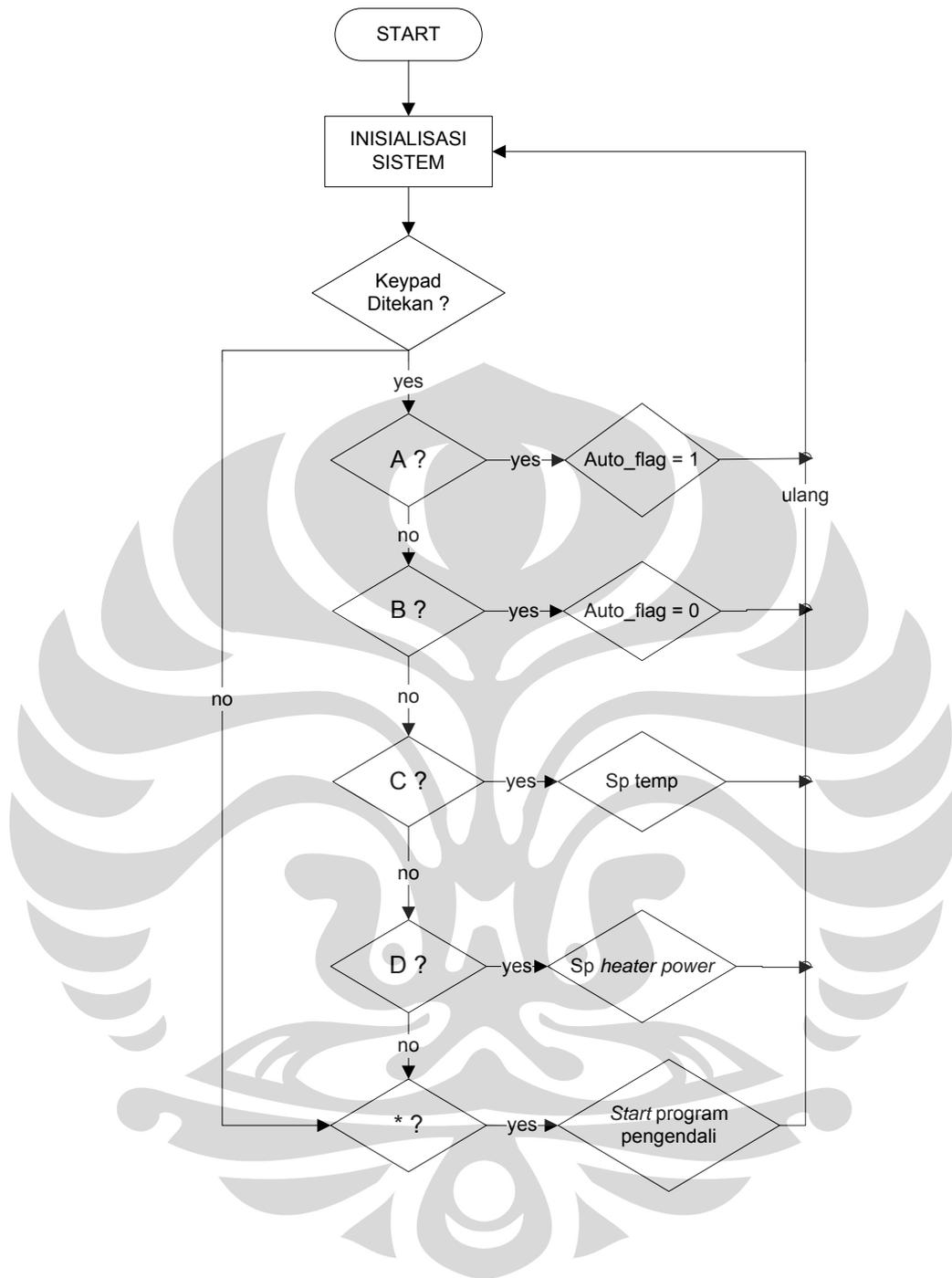
Pada *flowchart* dibawah adalah pengendali untuk *heater* dimana pada Atmega16 ini *inputnya* dari *keypad*. Awalnya Atmega16 ini menginisialisasi perintah dari *keypad* yaitu mengisi data yang akan dimasukkan kedalam rumus untuk dijalankan. Apabila temperatur dan *heater power*, sudah disimpan pada eeprom, maka data-data tersebut akan diolah untuk diproses. Data-data temperatur dan *heater power*, akan langsung dibandingkan untuk dijalankan karena didalam program AVR telah terdapat persamaan yang telah diambil dari pengambilan dari data alat sebelumnya. Data data pada eeprom adalah data *input* dari *keypad*.

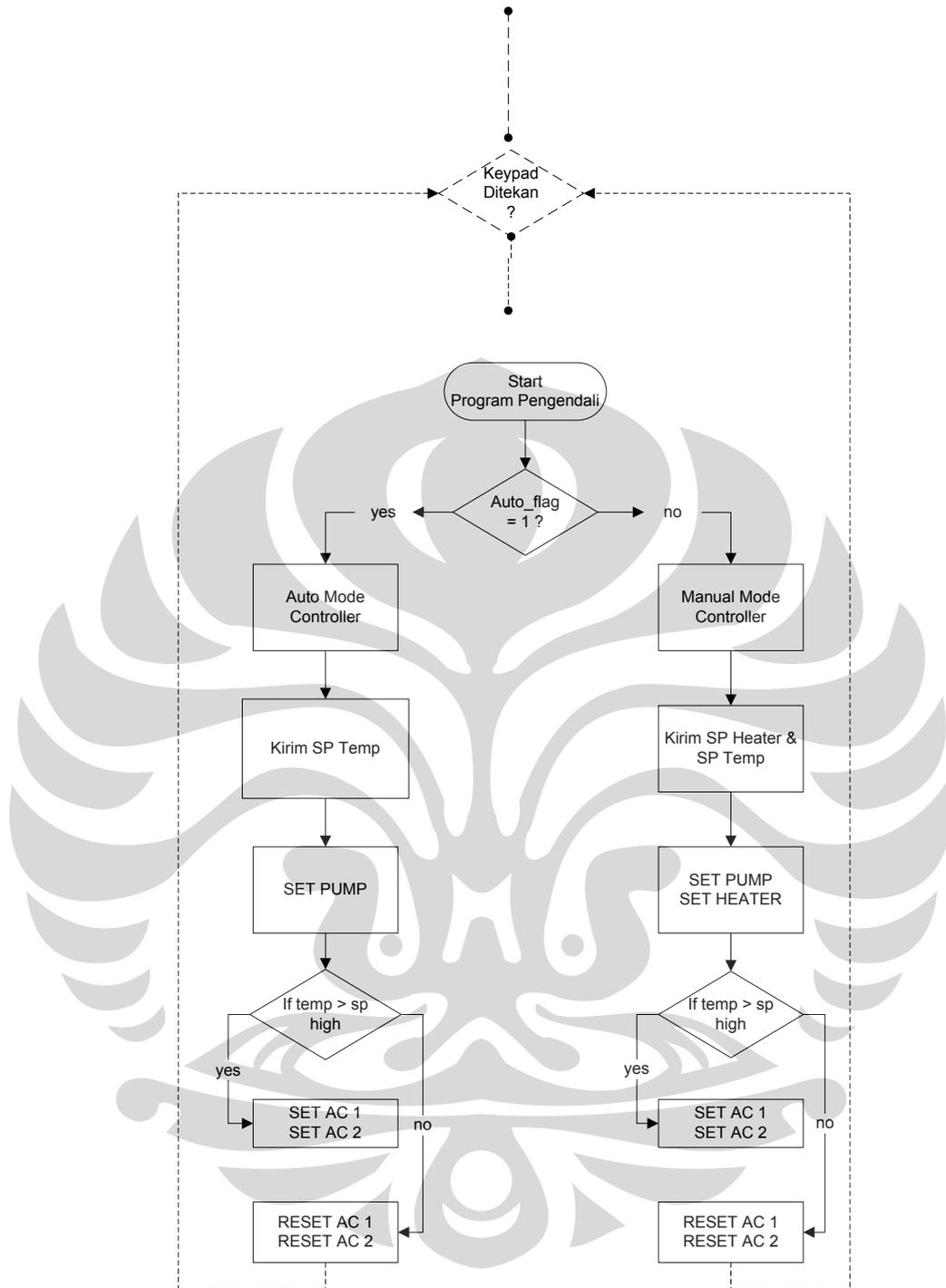
Apabila sudah membaca data pada eeprom AVR akan membaca nilai SP yaitu nilai untuk mengendalikan alat. Nilai SP ini juga *inputan* dari *keypad* setelah membaca nilai SP kemudian di *Start* maka AVR akan memproses data-data tersebut hingga nilai dari semua data terpenuhi. Proses ini akan berjalan terus - menerus sampai program di *STOP*, atau waktu telah habis. Pengendalian temperature dilakukan dengan mengendalikan tegangan yang lewat pada *Solid State Relay (SSR)*, sehingga SSR akan otomatis nyala mati untuk menstabilkan tegangan untuk AC1, AC2, *heater*, serta pompa sesuai dengan perintah pada program Bascom pada mikrokontroler.

Dari *flowchart* diatas dapat penulis jelaskan bahwa ketika *start* terjadi inisialisasi LCD diikuti dengan *ON-*nya mikro. Kemudian pengecekan status, jika *auto_flag* = 1, maka AC1, AC2, *Heater* serta pompa akan *ter-reset*. Lalu cek *keypad*. Ketika *input keypad* = A, maka *auto_flag* = 1, program auto akan aktif. Namun ketika *keypad* = B, maka *auto_flag* = 0, maka program test mode akan aktif serta menjalankan keseluruhan program AC, *heater* serta pompa.

Kemudian cek *keypad* kembali. Fungsi dari *keypad C*, yaitu ketika berfungsi sebagai *Set point* temperatur. Lalu klik *keypad* ” # ” untuk menyimpan nilai yang telah dimasukkan. Setelah memasukkan nilai *set point* temperatur (pada *auto_flag*) proses selanjutnya adalah cek *keypad* kembali. Jika *keypad* = D, yang harus dilakukan adalah menentukan *power heater* untuk menentukan berapa persen kemampuan dari *heater*. Lalu klik *keypad* ” # ” untuk menyimpan nilai yang telah dimasukkan. Selanjutnya, yang terakhir adalah mengklik ” * ” untuk memulai proses (*Start*).

Jika telah memasukan nilai temperatur atau *heater power*. Setelah itu, untuk mengecek apakah program yang dibuat berjalan dengan benar atau tidaknya, maka dapat memberi *input keypad* = A, program auto akan aktif. Pada program auto akan mengendalikan AC1, AC2 serta pompa. Dimana AC1 dan AC2 akan *ON* jika $Temp > Sp_high$. Sebaliknya, AC1 dan AC2 akan *OFF* jika $Temp < Sp_low$. Sedangkan, pompa akan aktif dari awal memilih program dan semua program auto akan mati ketika menekan tombol “#”. Namun ketika *keypad* = B, maka *auto_flag* = 0, maka program test mode akan aktif serta menjalankan keseluruhan program AC , *heater* serta pompa. . Dimana AC1 dan AC2 akan *ON* jika $Temp > Sp_high$. Sebaliknya, AC1 dan AC2 akan *OFF* jika $Temp < Sp_low$. Sedangkan, pompa dan heater akan aktif dari awal memilih program dan semua program test mode akan mati ketika menekan tombol “#”. Untuk mengetahui lebih tentang program pengendalian dapat terlihat pada *flowchart* di bawah .





Gambar 3.4 Flowchart Program Pengendali

BAB 4

HASIL PENGUJIAN DAN PEMBAHASAN

Setelah dilakukan pengerjaan keseluruhan sistem, maka perlu dilakukan pengujian alat serta penganalisaan terhadap alat, apakah sistem sudah bekerja dengan baik atau tidak. Pengujian-pengujian tersebut meliputi :

- Pengujian *Keypad*
- Pengujian LM35
- Pengujian Sistem Kendali Temperatur

4.1 Pengujian Keypad

Pada pengujian LM35 ini bertujuan untuk mengetahui berapa nilai-nilai yang dihasilkan dari *keypad* yang digunakan sehingga ketika menekan angka yang keluar sesuai dengan yang diinginkan . Berikut program Bascom yang dipergunakan untuk mengecek nilai *keypad* :

Tabel 4.1 Tabel Data Keypad

KEYPAD	NILAI
1	0
2	4
3	8
4	1
5	5
6	9
7	2
8	6
9	10
0	13
A	10
B	11
C	12
D	13
*	14
#	15

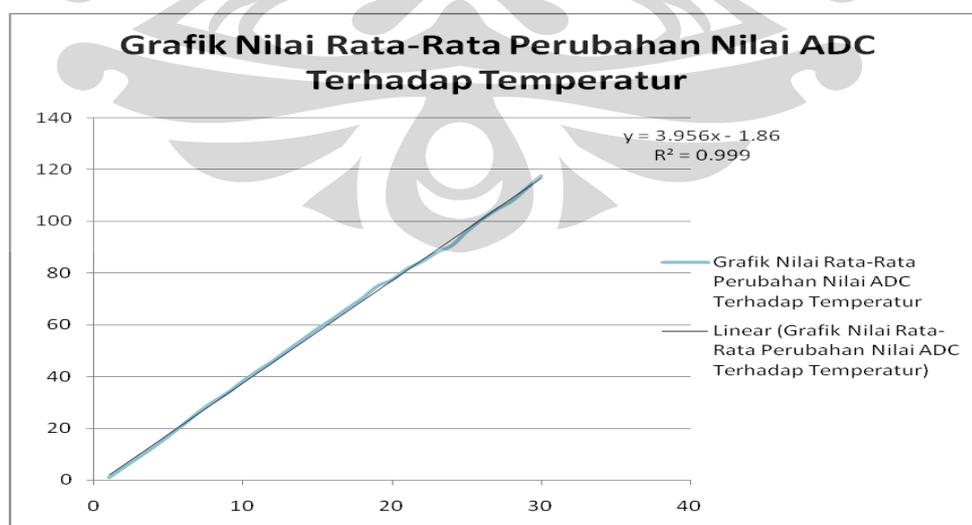
Dapat terlihat pada tabel data *keypad*, dimana nilai-nilai yang dihasilkan *keypad* memiliki nilai yang berbeda-beda. Ketika menekan angka 1 yang keluar pada tampilan LCD yakni berupa angka 0. Ketika menekan angka 2 yang keluar pada tampilan LCD yakni berupa angka 4. Ketika menekan "*" yang keluar pada tampilan LCD yaitu angka 14. Begitu pula selanjutnya, setiap *keypad* memiliki nilainya masing-masing.

Dari data tersebut nantinya akan dibuat program *keypad* pada Bascom, kita akan menggunakan *case*. Sehingga, nilai yang keluar ketika memilih angka pada *keypad* akan sesuai dengan keluarannya. Nilai *keypad* pun nantinya akan disesuaikan, sehingga dapat menekan angka *keypad* lebih dari satu digit.

4.2 Pengujian LM35

Pada pengujian LM35 ini bertujuan untuk mengetahui seberapa besar respon nilai ADC yang diukur oleh LM35. Metode yang digunakan pada pengambilan data ini, prinsipnya dilakukan dengan cara mengubah temperatur air secara kontinu dan berurutan. Dalam hal ini skala pengukuran yang dilakukan adalah pada temperatur 1⁰C sampai 30⁰C. Pengambilan data dilakukan selama 5 kali. Hal ini dimaksudkan untuk mendapatkan serta membandingkan keakuratan data yang dihasilkan.

Berikut grafik hasil pengambilan data perubahan nilai ADC sensor terhadap perubahan waktu.



Gambar 4.1 Grafik Nilai Rata-Rata Perubahan Nilai ADC Terhadap Temperatur

Dari grafik di bawah dapat diketahui bahwa perubahan nilai ADC sensor LM35 terhadap perubahan temperatur memiliki karakteristik kelinearitasan yang baik, yang mana respon tersebut dapat dinyatakan dengan persamaan:

$$Y = 3.956X - 1.86$$

$$R^2 = 0.999$$

Dimana x adalah nilai temperatur dan y adalah nilai bit dari ADC yang terukur pada termometer. Dari persamaan garis didapatkan nilai $R^2 = 0,999$, artinya sensor temperatur yang digunakan dalam pengukuran temperaturnya dapat dikatakan baik.

4.3 Pengujian Sistem Kendali Temperatur

Pada pengujian kendali ini bertujuan untuk mengendalikan temperatur. Dimana kita dapat mengetahui waktu yang diperlukan untuk menjaga nilai temperatur sehingga sesuai setpoint suhu yang digunakan. Metode yang digunakan pada pengambilan kali ini adalah dengan menggunakan setpoint suhu sebesar 15°C dengan set daya heater yang bervariasi 10 – 100 %.

Tabel 4.2 Tabel Pengendalian

Set Heater	waktu pendinginan (detik)	waktu pemanasan (detik)
100	515	250
90	475	349
80	455	478
70	383	503
60	369	572
50	362	693
40	326	696
30	252	931
20	212	1144
10	200	1819

Pada tabel di atas terlihat bahawa ketika *heater* di-*set* 10 %, maka waktu pendinginannya selama 200 detik dan waktu pemanasannya selama 1819 detik. Ketika *heater* di-*set* 100 %, maka waktu pendinginannya selama 515 detik dan waktu pemanasan selama 250 detik.

Sehingga dapat disimpulkan bahwa semakin rendah *set heater power* yang digunakan, maka waktu pendinginannya semakin cepat tetapi waktu pemanasannya semakin lama. Semakin tinggi *set heater power* yang digunakan, maka waktu pendinginannya semakin lama tetapi waktu pemanasannya semakin cepat.



BAB 5

PENUTUP

Pada bab ini kesimpulan yang diperoleh penulis setelah melakukan pembuatan tugas akhir ini serta saran-saran diharapkan untuk kedepannya didapatkan suatu sistem dan hasil yang lebih baik lagi dimasa yang akan mendatang.

5.1 Kesimpulan

- 1) LM35 yang digunakan adalah LM35D .
- 2) Diperoleh nilai $R^2 = 0.999$ yang berarti bahwa sensor suhu bekerja dengan baik.
- 3) *Set point* yang didapat berasal dari *keypad*.
- 4) Alat ini berjalan cukup baik dalam proses pemanasan dan pendinginan dengan mengatur temperatur.
- 5) Pengendalian temperaturnya berjalan dengan baik sesuai perintah pada pemrograman.
- 6) Temperatur lingkungan serta perputaran air pada bak yang kurang baik menyebabkan terganggunya proses pembacaan temperatur pada sensor LM35.

5.2. Saran

1. Sistem isolasi temperatur air pada bak penampung harus baik.
2. Penggunaan alat keselamatan untuk menghindari kontak dengan listrik, sebab daya yang digunakan terhitung besar.
3. Dilakukan percobaan berulang-ulang agar mendapatkan hasil yang akurat.

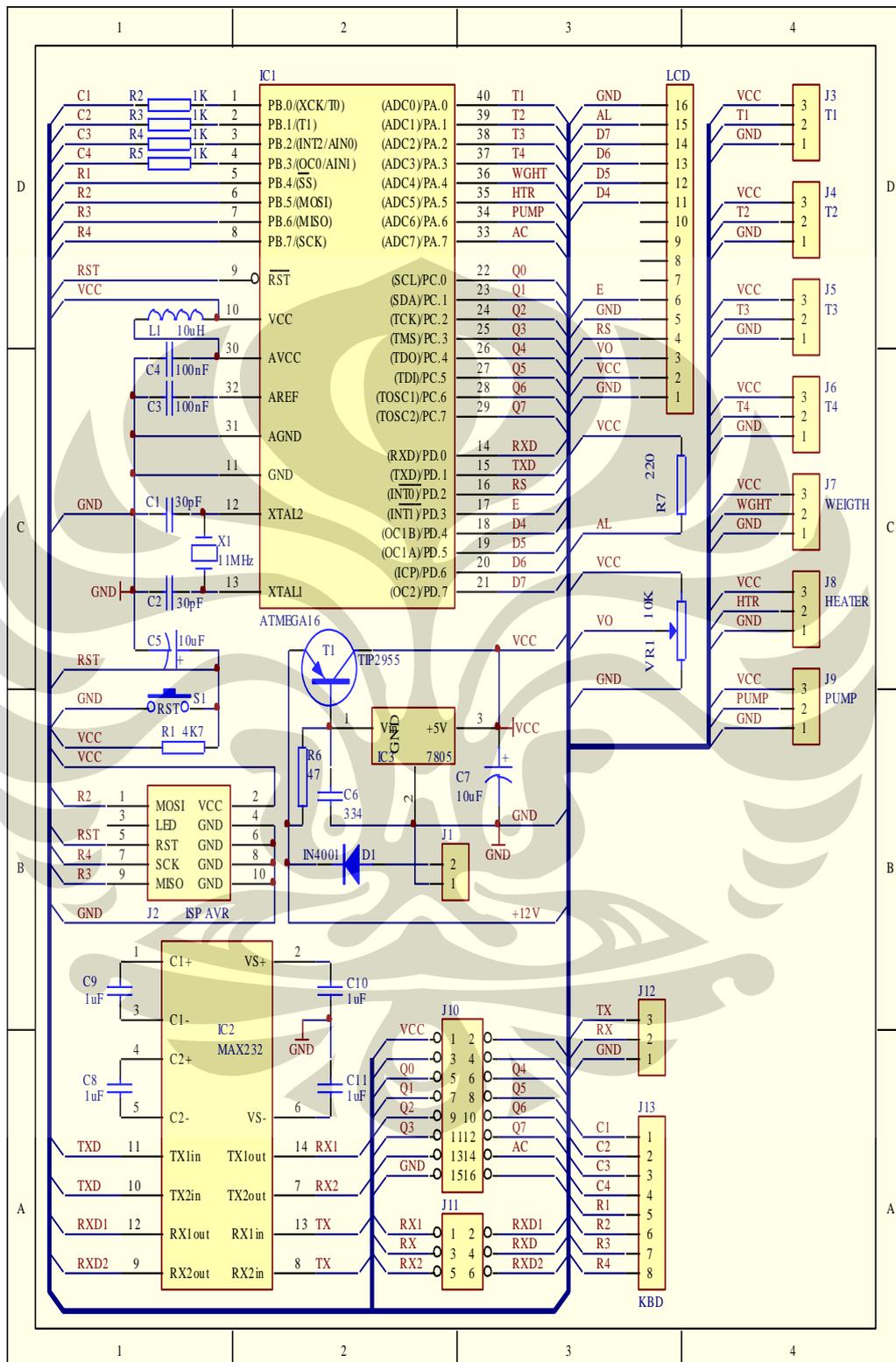
DAFTAR ACUAN

- [1] Datasheet LM35, <http://www.alldatasheet.com>, dibuka tanggal 13 Mei 2010
- [2] Datasheet ATMEGA 16, <http://www.alldatasheet.com>, dibuka tanggal 13 Mei 2010
- [3] Konfigurasi DB9 dan Tabel DB9, <http://duniaengineering.blogdetik.com>, dibuka tanggal 13 Mei 2010
- [4] Layout Posisi Komponen USB AVR ISP (USB AVR Downloader) dan Keterangan Pin, <http://depokinstruments.com>, dibuka tanggal 13 Mei 2010
- [5] Rangkaian Keypad, <http://anotherorions.blogspot.com>, dibuka tanggal 13 Mei 2010
- [6] Budiharto, Widodo . 2007. *12 Proyek Sistem Akusisi Data* . Jakarta : PT Elex Media Komputindo
- [7] Budiharto, Widodo . 2007. *Panduan Praktikum Mikrokontroler AVR Atmega16*. Jakarta : PT Elex Media Komputindo
- [9] Termometer Galileo, <http://www.Wikipedia.com>, dibuka tanggal 13 Mei 2010



LAMPIRAN

LAMPIRAN 1



Dim Keypad As Byte
Dim Start_heater_flag As Bit
Dim Batas As Bit
Dim Setpoint As Byte
Dim Sp_high As Single
Dim Sp_low As Single
Dim Sp_heater As Byte
Dim Nilai As Byte
Dim Number As Byte
Dim Periode_1s As Byte
Dim Periode_10ms As Byte
Dim Daya_listrik As Byte
Dim Data_adc As Word
Dim Temp As Single
Dim Temp_chr As String * 4

Start Adc

Main_program:

Reset Ac1

Reset Ac2

Reset Heater

Reset Pump

Error_old = 0

Start_heater_flag = 0

Cursor Off

Cls

Locate 1 , 1

Lcd " CHILLER SYSTEM "

Locate 2 , 1

Lcd "PHYSICS DEPARTEMENT"

Locate 3 , 1

Lcd "D3 INSTRUMENTASI 07"

Locate 4 , 1

Lcd "DESIGN BY AYU & SULE"

Ulang = 1

Balik = 1

Do

Keypad = Getkbd()

If Keypad < 16 Then Ulang = 0

Loop Until Ulang = 0

Select Case Keypad

Case 12

Do

Keypad = Getkbd()

If Keypad = 16 Then Balik = 0

Loop Until Balik = 0

Goto Auto_prog

Case 13

Do

Keypad = Getkbd()

If Keypad = 16 Then Balik = 0

Loop Until Balik = 0

Goto Test_prog

Case 14

Do

Keypad = Getkbd()

If Keypad = 16 Then Balik = 0

Loop Until Balik = 0

Goto Sp_temp_prog

Case 15

Do

Keypad = Getkbd()

If Keypad = 16 Then Balik = 0

Loop Until Balik = 0

Goto Sp_daya_prog

Case 3

Do

```
Keypad = Getkbd()
If Keypad = 16 Then Balik = 0
Loop Until Balik = 0
Goto Chiller_prog
End Select
Goto Main_program
```

```
Auto_prog:
Auto_flag = 1
Locate 4 , 1
Lcd "      "
Locate 4 , 1
Lcd "AUTO MODE CONTROLLER"
Wait 1
Goto Main_program
```

```
Test_prog:
Auto_flag = 0
Locate 4 , 1
Lcd "      "
Locate 4 , 1
Lcd "TEST MODE CONTROLLER"
Wait 1
Goto Main_program
```

```
Sp_temp_prog:
Nilai = 0
Locate 4 , 1
Lcd "      "
Locate 4 , 1
Lcd "SP TEMP = " ; Setpoint ; Chr(223) ; "C"
```

```
Balik = 1
Do
  Keypad = Getkbd()
  If Keypad < 16 Then
    Select Case Keypad
      Case 0
        Number = 1
      Case 1
        Number = 4
      Case 2
        Number = 7
      Case 4
        Number = 2
      Case 5
        Number = 5
      Case 6
        Number = 8
      Case 7
        Number = 0
      Case 8
        Number = 3
      Case 9
        Number = 6
      Case 10
        Number = 9
      Case 11
        Number = 15
    End Select
  Ulang = 1
Do
  Keypad = Getkbd()
  If Keypad = 16 Then Ulang = 0
Loop Until Ulang = 0
If Number < 10 Then
```

```

    Nilai = Nilai * 10
    Nilai = Nilai + Number
Else
    If Number = 15 Then
        Balik = 0
    End If
End If
Sp_high = Nilai + 1
Sp_low = Nilai - 1
Setpoint = Nilai
Locate 4 , 11
Lcd "      "
Locate 4 , 11
Lcd Setpoint ; Chr(223) ; "C"
End If
Loop Until Balik = 0
Goto Main_program

Sp_daya_prog:
Nilai = 0
Locate 4 , 1
Lcd "      "
Locate 4 , 1
Lcd "SP HEATER PWR = " ; Sp_heater ; "%"
Balik = 1
Do
    Keypad = Getkbd()
    If Keypad < 16 Then
        Select Case Keypad
            Case 0
                Number = 1
            Case 1
                Number = 4

```

```
Case 2
    Number = 7
Case 4
    Number = 2
Case 5
    Number = 5
Case 6
    Number = 8
Case 7
    Number = 0
Case 8
    Number = 3
Case 9
    Number = 6
Case 10
    Number = 9
Case 11
    Number = 15
End Select
Ulang = 1
Do
    Keypad = Getkbd()
    If Keypad = 16 Then Ulang = 0
Loop Until Ulang = 0
If Number < 10 Then
    Nilai = Nilai * 10
    Nilai = Nilai + Number
Else
    If Number = 15 Then
        Balik = 0
    End If
End If
Sp_heater = Nilai
Locate 4 , 17
```

```

    Lcd " "
    Locate 4 , 17
    Lcd Sp_heater ; "%"
End If
Loop Until Balik = 0
Goto Main_program

```

```

Serial_in:
Disable Interrupts
Data_seri = Inkey()
If Data_seri = "*" Then
    Data_seri = Waitkey()
    If Data_seri = "G" Then Kirim_flag = 1
End If
Enable Interrupts
Return

```

```

Chiller_prog:
If Auto_flag = 1 Then
    Cls
    Locate 1 , 1
    Lcd " CHILLER SYSTEM "
    Locate 2 , 1
    Lcd " ON AUTOMATIC MODE "
    Locate 3 , 1
    Lcd "SET POINT = " ; Setpoint ; Chr(223) ; "C"
    Locate 4 , 1
    Lcd "TEMPERATURE = " ; Temp_chr ; Chr(223) ; "C"
    Set Pump
    Ulang = 1
    Start_heater_flag = 0
Do

```

```
If Pengendalian_flag = 1 Then
  Pengendalian_flag = 0
  Data_adc = Getadc(4)
  Temp = Data_adc / 4
  Temp_chr = Fusing(temp , "#.#")
  Locate 4 , 15
  Lcd "    "
  Locate 4 , 15
  Lcd Temp_chr ; Chr(223) ; "C"
  If Temp < Sp_low Then
    If Batas = 1 Then
      Batas = 0
      Reset Ac1
      Reset Ac2
    End If
  End If
  If Temp > Sp_high Then
    If Batas = 0 Then
      Batas = 1
      Set Ac1
      Set Ac2
    End If
  End If
  End If
  Keypad = Getkbd()
  If Keypad = 15 Then
    Reset Ac1
    Reset Ac2
    Reset Pump
    Ulang = 0
  End If
  Loop Until Ulang = 0
Else
  Cls
```

```

Locate 1 , 1
Lcd "TEST MODE CONTROLLER"
Locate 2 , 1
Lcd "HEATER POWER = " ; Sp_heater ; " %"
Locate 3 , 1
Lcd "SET POINT  = " ; Setpoint ; Chr(223) ; "C"
Locate 4 , 1
Lcd "TEMPERATURE = " ; Temp_chr ; Chr(223) ; "C"
Locate 4 , 15
Lcd "  "
Locate 4 , 15
Lcd Temp_chr ; Chr(223) ; "C"
Set Pump
Set Heater
Ulang = 1
Start_heater_flag = 1
Do
  If Pengendalian_flag = 1 Then
    Pengendalian_flag = 0
    Data_adc = Getadc(4)
    Temp = Data_adc / 4
    Temp_chr = Fusing(temp , "#.#")
    Locate 4 , 15
    Lcd "  "
    Locate 4 , 15
    Lcd Temp_chr ; Chr(223) ; "C"
    If Temp < Sp_low Then
      If Batas = 1 Then
        Batas = 0
        Reset Ac1
        Reset Ac2
      End If
    End If
    If Temp > Sp_high Then

```

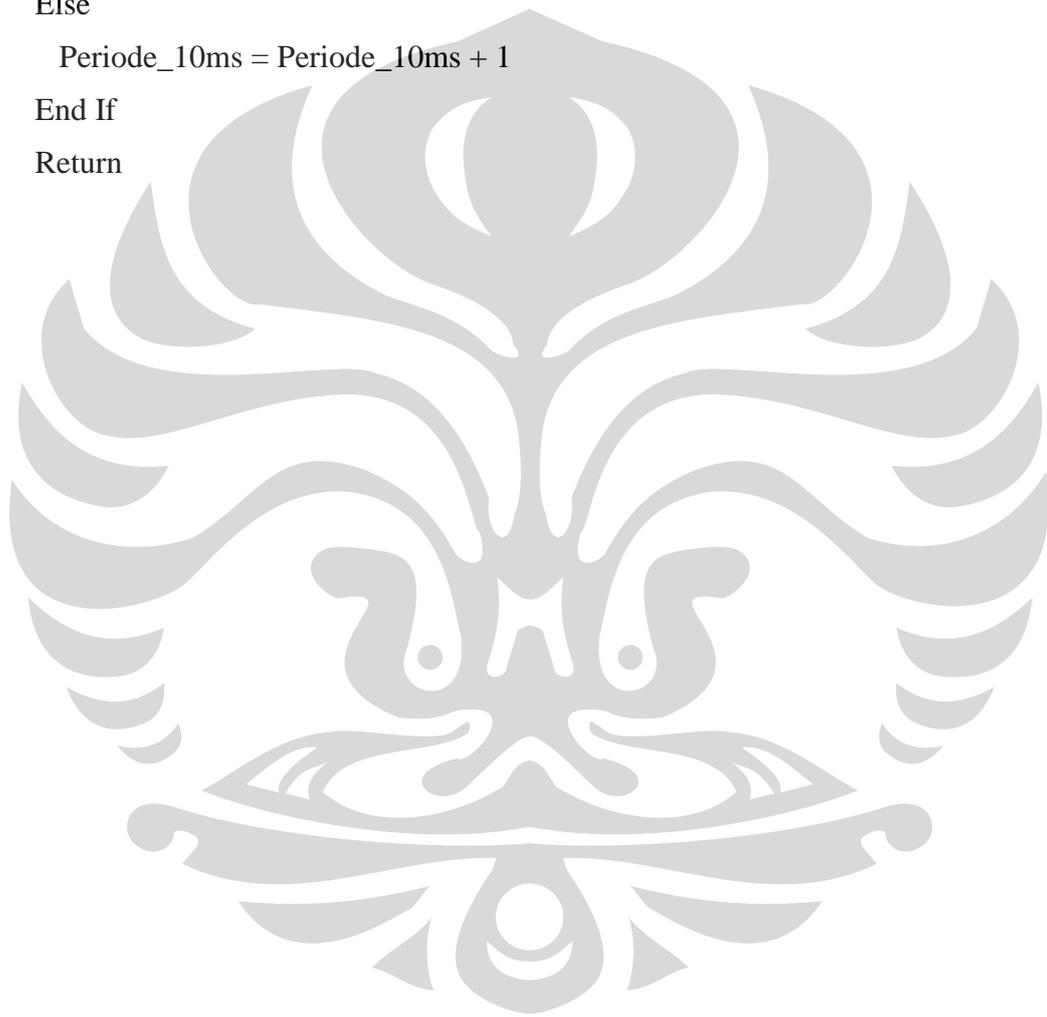
```

    If Batas = 0 Then
        Batas = 1
        Set Ac1
        Set Ac2
    End If
End If
End If
Keypad = Getkbd()
If Keypad = 15 Then
    Reset Ac1
    Reset Ac2
    Reset Pump
    Reset Heater
    Ulang = 0
End If
Loop Until Ulang = 0
End If
Goto Main_program

Time_base:
If Periode_10ms = 216 Then
    Periode_10ms = 0
    If Periode_1s = 100 Then
        Periode_1s = 0
        Daya_listrik = Sp_heater
        Pengendalian_flag = 1
    Else
        Periode_1s = Periode_1s + 1
    End If
    If Start_heater_flag = 1 Then
        If Daya_listrik = 0 Then
            Reset Heater

```

```
Else
    Daya_listrik = Daya_listrik - 1
    Set Heater
End If
Else
    Reset Heater
End If
Else
    Periode_10ms = Periode_10ms + 1
End If
Return
```



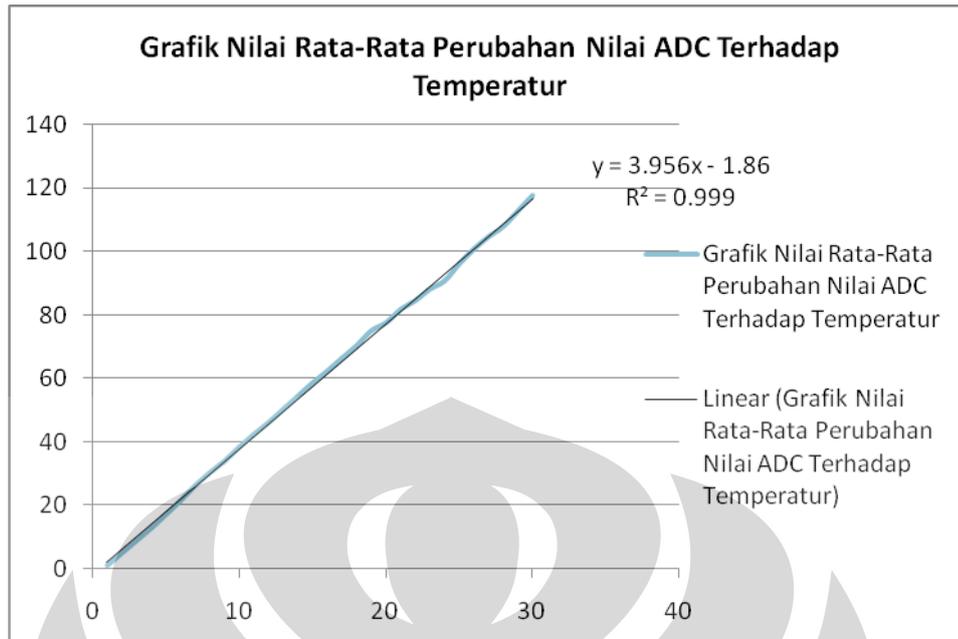
LAMPIRAN 4

Tabel Data LM35 (Temperatur Turun)

Temperatur	Data ADC 1	Data ADC 2	Data ADC 3	Data ADC 4	Data ADC 5
30	117	113	111	120	115
29	110	110	104	110	108
28	103	108	103	104	101
27	101	104	100	102	97
26	97	100	98	99	92
25	95	89	94	95	87
24	89	80	91	92	80
23	90	77	88	88	78
22	81	76	85	84	76
21	83	71	81	80	74
20	74	68	76	76	70
19	76	67	71	73	69
18	68	65	67	69	67
17	62	63	64	64	64
16	58	60	59	61	59
15	57	57	55	57	55
14	51	54	51	54	51
13	49	52	49	47	42
12	48	46	43	41	41
11	40	42	40	37	40
10	39	38	33	32	35
9	34	34	29	30	32
8	29	28	28	27	2
7	28	24	21	24	24
6	22	19	17	19	19
5	18	16	14	16	13
4	15	9	9	14	9
3	12	7	6	8	8
2	4	5	5	6	4
1	0	3	0	1	0

Tabel Data LM35 (Temperatur Naik)

Temperatur	ADC (data 1)	ADC (data 2)	ADC (data 3)	ADC (data 4)	ADC (data 5)
1	2	2	2	1	0
2	5	7	6	3	6
3	10	9	10	9	11
4	15	16	14	13	15
5	17	21	19	19	19
6	25	26	23	23	21
7	30	29	29	27	27
8	33	33	32	32	34
9	36	38	36	35	36
10	42	42	40	40	43
11	43	47	44	43	48
12	46	48	47	49	52
13	54	51	51	53	55
14	56	55	57	56	59
15	58	63	61	61	62
16	62	67	64	65	68
17	67	73	69	67	71
18	71	76	72	73	75
19	79	80	76	78	81
20	81	85	80	82	83
21	85	88	83	85	87
22	90	89	85	89	90
23	93	91	90	93	93
24	94	96	94	97	94
25	97	99	97	101	98
26	103	104	103	105	106
27	109	108	107	109	109
28	113	110	112	112	112
29	120	115	115	118	117
30	122	119	116	123	120



Gambar Grafik Data ADC (Rata-rata keseluruhan data naik maupun turun)

Tabel Data Pengendalian

Set Heater	waktu pendinginan	waktu pemanasan
100	515	250
90	475	349
80	455	478
70	383	503
60	369	572
50	362	693
40	326	696
30	252	931
20	212	1144
10	200	1819