

**SISTEM KONTROL BUKA TUTUP VALVE PADA
PROSES PEMANASAN AIR JAKET**

SKRIPSI

**Diajukan untuk Memenuhi Persyaratan Memperoleh
Gelar Sarjana Teknik**

Oleh :

**SUROSO
11.812.0003**



**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS MEDAN AREA
MEDAN
2016**

LEMBAR PENGESAHAN

**SISTEM KONTROL BUKA TUTUP VALVE PADA PROSES
PEMANASAN AIR JAKET**

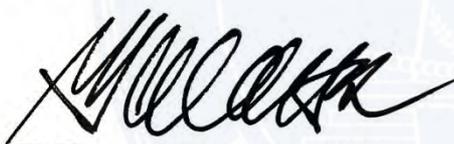
SKRIPSI

Oleh :

SUROSO
11.812.0003

Disetujui oleh :

Pembimbing I

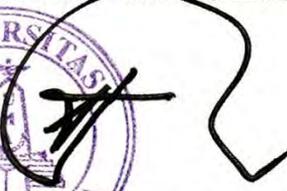

(Ir. H. Usman Harahap, MT)

Pembimbing II


(Faisal Irsan Pasaribu, ST.MT)

Mengetahui :

Dekan Fakultas Teknik


(Prof. Dr. Dadan Ramdan, M.Eng, M.Sc)

Ka. Program Studi


(Faisal Irsan Pasaribu, ST.MT)

LEMBAR PERNYATAAN

Saya menyatakan bahwa skripsi yang disusun, sebagai syarat memperoleh gelar sarjana merupakan hasil karya tulis saya sendiri. Adapun bagian - bagian tertentu dalam penulisan skripsi ini yang saya kutip dari hasil karya orang lain telah dituliskan sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan ilmiah.

Saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi-sanksi lainnya dengan peraturan yang berlaku, apabila dikemudian hari ditemukan adanya plagiat dalam skripsi ini.

Medan, 18 November 2016

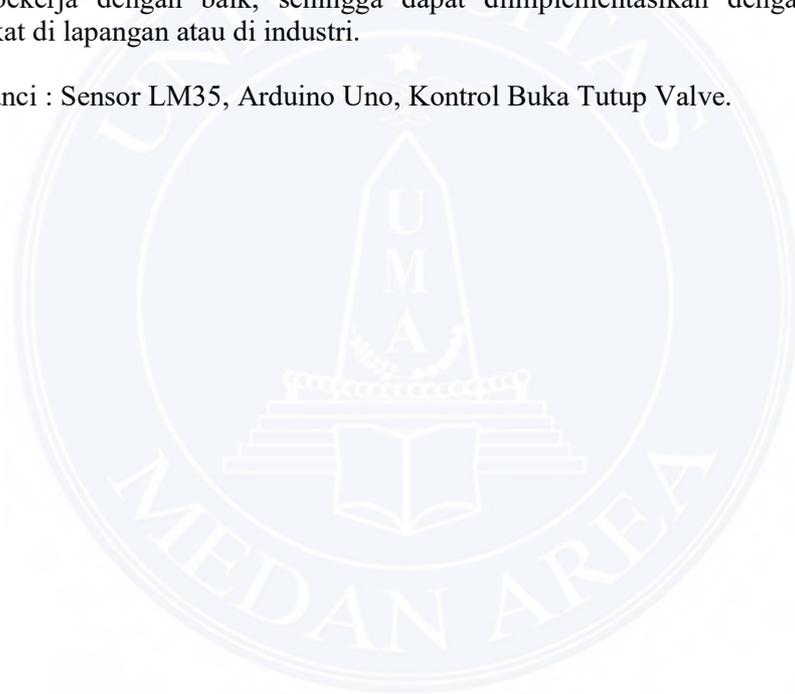


Suroso
11.812.0003

ABSTRAK

Proyek akhir kontrol derajat pembukaan dan penutupan valve sistem distribusi uap pada proses pemanasan air jaket menggunakan pengendali mikrokontroler ATmega 328 ini dirancang untuk dapat memudahkan bagi para pekerja untuk melakukan proses pengontrolan suhu air pada jaket. Manfaat yang lain adalah agar dapat menghindari kegagalan pada proses pencairan minyak kelapa sawit hasil pengolahan sebelumnya sehingga minyak dalam pipa tetap mengalir. Metode yang digunakan dalam proyek akhir ini adalah metode rancang bangun yang terdiri dari beberapa tahap yaitu, (1) Analisis kebutuhan, (2) Perancangan, (3) Implementasi rangkaian, (4) Prosedur Pengujian dan (5) Pengujian alat. Alat ini terbagi menjadi dua bagian, yaitu bagian *hardware* dan *software*. *Hardware* terdiri dari sensor suhu LM35, sistem minimum mikrokontroler ATmega 328 yaitu Arduino Uno sebagai rangkaian pengendali *input* dan *output*, motor servo sebagai output (valve) yang dikendalikan, dan *LCD* sebagai perangkat penampil hasil pendeteksian suhu. Sedangkan *software* yang dibuat menggunakan program arduino dan bahasa “C”. Unjuk kerja alat ini secara keseluruhan dapat bekerja dengan baik, sehingga dapat diimplementasikan dengan pencocokan perangkat di lapangan atau di industri.

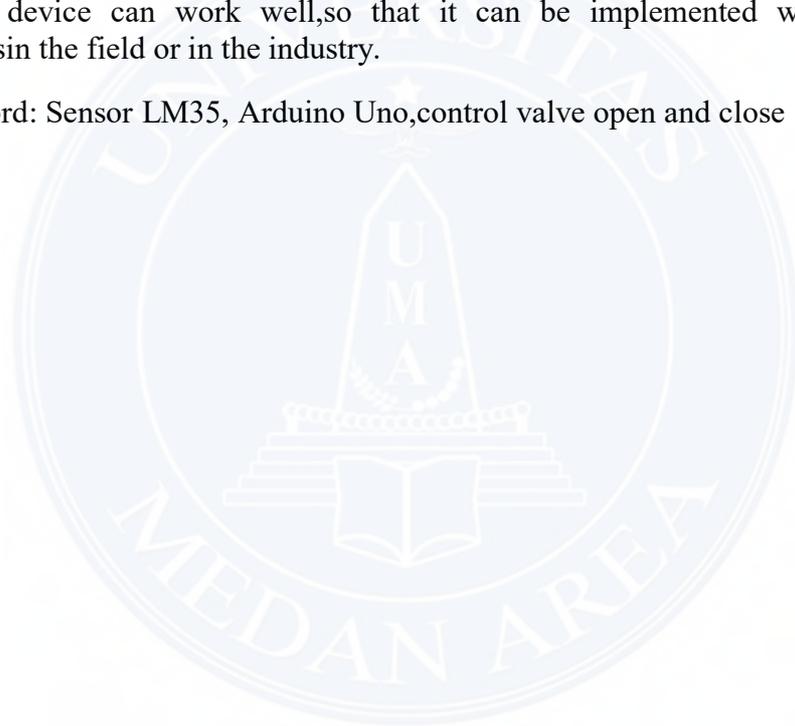
Kata kunci : Sensor LM35, Arduino Uno, Kontrol Buka Tutup Valve.



ABSTRACT

The final project degree of control degree control valve opening and closing of the steam distribution system on the jacket water heating process using the controller 328 ATmega micro controller is designed to make it easier for works to do the process of controlling the temperature of the water in the jacket. Another benefit is to be able to avoid failure in the liquefaction process in palm oil so that the flowing. The method used in this final project is a design method that consist of several phases,namely, (1) analysis of needs, (2) design, (3) implementations of the circuit, (4) testing procedures and (5) testing tools. The tools is divided into two parts, namely the hardware and software. The hardware consist of LM35 temperature sensor,microcontroller minimum system ATmega 328 is Arduino Uno as a series of input and output controllers,servo motor as output (valve) is controlled, and LCD as a viewerdevice temperature detection results.while the software is created using arduino and language "C". The performance of the whole device can work well,so that it can be implemented with matching devicesin the field or in the industry.

Keyword: Sensor LM35, Arduino Uno,control valve open and close



RIWAYAT HIDUP

Penulis dilahirkan di Silakidir Pada tanggal 03 agustus 1989 dari ayah Sagio dan ibu Katemi .Penulis merupakan anak ke 3 dari 4 bersaudara.

Tahun 2007 Penulis lulus dari SMK Negeri 2 Pematang siantar dan pada tahun 2011 terdaftar sebagai mahasiswa Fakultas Teknik Universitas Medan Area.

Penulis melaksanakan Praktek Kerja Lapangan (PKL) di PT. PLN (PERSERO) P3B SUMATERA UNIT PELAYANAN TRANSMISI MEDAN. Jalan listrik No12 medan.

Cabang : TRANSMISI DAN GARDU INDUK PAYA PASIR (GARDU INDUK KIM).Jalan bangkalan-mabar-medan deli.

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Kuasa atas segala karunianya sehingga skripsi ini berhasil diselesaikan. Tema yang dipilih dalam penelitian ini ialah tentang sistem pengontrolan dengan judul “Sistem Kontrol Buka Tutup Valve Pada Proses Pemanasan Air Jacket”.

Terima kasih penulis sampaikan kepada Bapak Faisal Irsan Pasaribu, ST, MT, selaku Ketua Program Studi Teknik Elektro dan sekaligus sebagai Pembimbing II, dan terimakasih juga penulis sampaikan kepada Bapak Ir. H. Usman Harahap, MT, selaku Pembimbing I dimana keduanya telah banyak memberikan arahan dan bimbingan serta saran sehingga skripsi ini dapat terselesaikan dengan baik. Tak lupa juga saya ucapkan terima kasih kepada Bapak Ir. Zulkifi Bahri, MT, selaku ketua dan saya ucapkan terima kasih kepada Ibu Syarifah Muthia Putri, ST, MT selaku sekretaris. Karna sudah memberi masukan yang sangat bermanfaat bagi penulis. Disamping itu penghargaan penulis sampaikan kepada rekan – rekan mahasiswa sepejuangan di jurusan teknik elektro yang telah membantu penulis selama melaksanakan penelitian. Ungkapan terima kasih juga disampaikan kepada ayah dan ibu saya, serta seluruh keluarga atas segala do’a dan perhatiannya.

Semoga skripsi ini bermanfaat.

Penulis

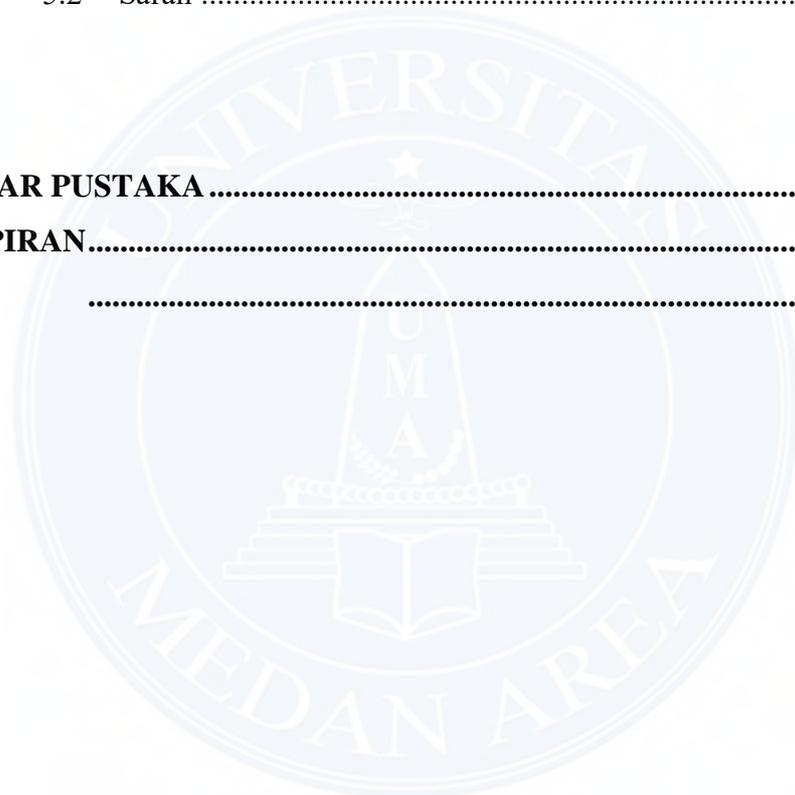


Suroso

ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR.....	iii
DAFTAR ISI	iv
DAFTAR TABEL	vii
DAFTAR GAMBAR	viii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	3
1.6 Sistematika Pembahasan	3
1.7 Kerangka Berfikir.....	4
BAB II TEORI PENUNJANG.....	5
2.1 Umum	5
2.2 Smart Building	5
2.3 Arduino	6
2.3.1 Arduino Uno.....	7
2.3.1.1 Pin Masukan dan Keluaran Arduino Uno	8
2.3.1.2 Sumber Daya dan Pin Tegangan Arduino Uno.....	
2.3.1.3 Peta Memori Arduino Uno.....	
2.3.1.3.1 Memori Program.....	
2.3.1.3.2 Memori Data.....	
2.3.1.3.3 Memori Data EEPROM.....	11

2.3.2	Arduino Ethernet Shield.....	12
2.3.3	Bahasa Pemrograman Arduino	13
2.3.3.1	Struktur.....	13
2.3.3.2	Konstanta.....	14
2.3.3.3	Fungsi Masukan dan Keluaran Digital.....	16
2.3.3.4	Progmem	17
2.3.4	Arduino Development Environment	20
2.4	Sensor	22
2.4.1	LM 35.....	22
2.5	LCD	25
2.6	Motor Servo Standar	29
2.7	Valve (Katup).....	31
2.7.1	Fungsi Valve	31
2.7.2	Jenis-jenis Valve	32
2.8	Jaket Air	34
BAB III	METODELOGI PENELITIAN.....	37
3.1	Lokasi Penelitian	37
3.2	Blok Diagram	37
3.3	Alat, Bahan, dan Desain Penelitian.....	38
3.3.1	Alat dan Bahan	38
3.3.1.1	Penetapan Komponen.....	38
3.3.1.2	Spesifikasi Komponen	39
3.4	Desain Penelitian.....	43
3.5	Skema Rangkaian dan Instalasi Alat.....	43
3.6	Skema Penggabungan Seluruh Instalasi Sistem.....	46
3.7	Metode Pengujian.....	47
3.7.1	Pengujian secara Keseluruhan.....	50
3.7.2	Program Sistem Arduino	51
3.7.3	Flowchart Sistem Kerja Alat	53

BAB IV HASIL DAN PEMBAHASAN.....	54
4.1 Hasil Perancangan Alat	54
4.2 Hasil Pengujian Sensor Suhu LM35	54
4.3 Hasil Pengujian secara Keseluruhan	60
BAB V PENUTUP	63
5.1 Kesimpulan	63
5.2 Saran	63
DAFTAR PUSTAKA	64
LAMPIRAN.....	65
.....	



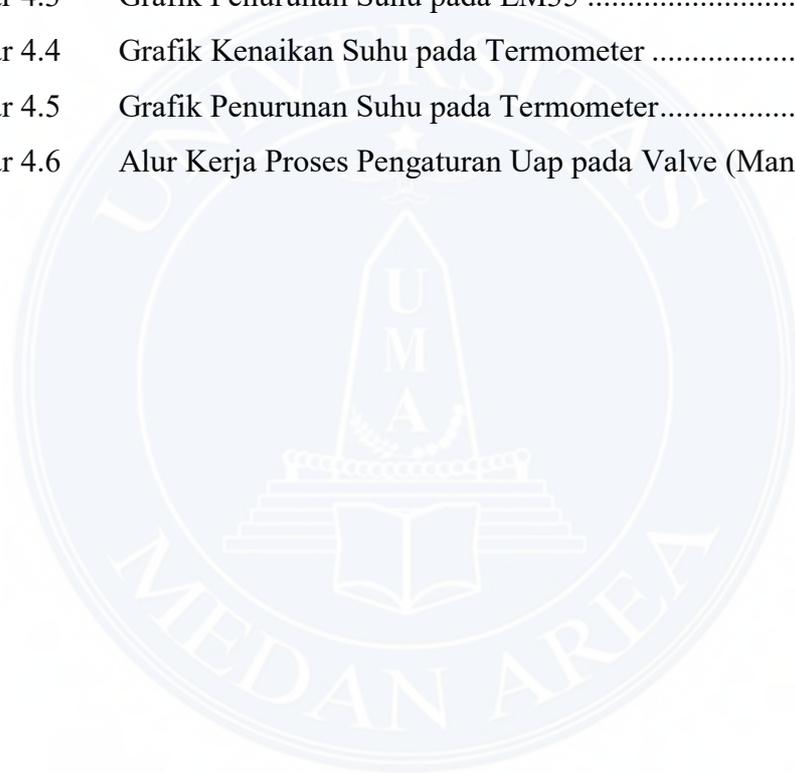
DAFTAR TABEL

Tabel 2.1	Keterangan Pin LCD 16 x 2.....	27
Tabel 2.2	Pengaturan Display LCD	28
Tabel 2.3	Pengaturan Display LCD CGROM	28
Tabel 3.1	Jadwal Pelaksanaan Penelitian.....	37
Tabel 3.2	Daftar Komponen Elektronik.....	38
Tabel 3.3	Daftar Bahan	39
Tabel 3.4	Daftar Alat	39
Tabel 3.5	Spesifikasi LCD 16 x 2	42
Tabel 3.6	Bentuk Pengujian Sensor LM35 (Kenaikan Suhu).....	48
Tabel 3.7	Bentuk Pengujian Sensor LM35 (Penurunan Suhu).....	48
Tabel 3.8	Bentuk Pengujian Termometer (Kenaikan Suhu).....	49
Tabel 3.9	Bentuk Pengujian Termometer (Penurunan Suhu)	49
Tabel 3.10	Bentuk Pengujian Seluruh Sistem.....	50
Tabel 4.1	Hasil Pengujian Sensor LM35 (Kenaikan Suhu)	55
Tabel 4.2	Hasil Pengujian Sensor LM35 (Penurunan Suhu)	56
Tabel 4.3	Hasil Pengujian Termometer (Kenaikan Suhu)	57
Tabel 4.4	Hasil Pengujian Termometer (Penurunan Suhu)	59
Tabel 4.5	Hasil Pengujian Sistem secara Keseluruhan	62

DAFTAR GAMBAR

Gambar 1.1	Flowchart Kerangka Berfikir	4
Gambar 2.1	Blok Diagram Arduino Board	7
Gambar 2.2	Arduino Uno	8
Gambar 2.3	Peta Memori Program ATmega 328	10
Gambar 2.4	Peta Memori Data ATmega 328	11
Gambar 2.5	Ethernet Shield	12
Gambar 2.6	Arduino Development Environment	21
Gambar 2.7	Bentuk Fisik LM35	23
Gambar 2.8	Rangkaian Dalam LM35	24
Gambar 2.9	Susunan Pin LCD 16 x 2	26
Gambar 2.10	Hubungan Power Supply ke LCD	26
Gambar 2.11	Kode Karakter CGROM pada LCD	29
Gambar 2.12	Motor Servo	30
Gambar 2.13	Hubungan Lebar Pulsa PWM Motor Servo	30
Gambar 2.14	Gate Valve	32
Gambar 2.15	Plug Valve	33
Gambar 2.16	Ball Valve	33
Gambar 2.17	Glove Valve	34
Gambar 2.18	Jaket Air	34
Gambar 2.19	Layar Jendela Aplikasi Pemantauan Suhu Air Jacket	35
Gambar 3.1	Blok Diagram Sistem	37
Gambar 3.2	Bentuk Fisik Motor Servo	40
Gambar 3.3	Sensor Suhu LM35	41
Gambar 3.4	Bentuk Fisik Power Supply	41
Gambar 3.5	Bentuk Fisik LCD 16 x 2	42
Gambar 3.6	Tata Letak Komponen	43
Gambar 3.7	Skema Rangkaian Arduino Uno	44

Gambar 3.8	Skema Instalasi Sensor LM35.....	44
Gambar 3.9	Skema Instalasi Motor Servo	45
Gambar 3.10	Skema Instalasi LCD.....	45
Gambar 3.11	Skema Seluruh Sistem.....	46
Gambar 3.12	Skema Pengujian Sensor LM35	48
Gambar 3.13	Jendela Aplikasi Penulisan Program.....	52
Gambar 3.14	Flowchart Penelitian.....	53
Gambar 4.1	Hasil Rancangan Alat.....	54
Gambar 4.2	Grafik Kenaikan Suhu pada LM35	55
Gambar 4.3	Grafik Penurunan Suhu pada LM35	57
Gambar 4.4	Grafik Kenaikan Suhu pada Termometer	58
Gambar 4.5	Grafik Penurunan Suhu pada Termometer.....	59
Gambar 4.6	Alur Kerja Proses Pengaturan Uap pada Valve (Manual)	61



BAB I

PENDAHULUAN

1.1. Latar Belakang

PT. Musim Mas adalah salah satu perusahaan yang selama ini fokus bergerak di industri *oleochemical* atau industri pengolahan minyak kelapa sawit menjadi bahan-bahan kimia seperti asam lemak (*fatty Acid*) dan *gliserin*. Industri ini, dalam melakukan proses pengolahan minyak kelapa sawit mempunyai peralatan-peralatan yang mendukung agar tujuan perusahaan dalam menghasilkan produk yang baik dapat tercapai. Adapun peralatan tersebut adalah sejumlah mesin industri.

Seiring dengan perkembangan IPTEK khususnya sistem otomasi, PT. Musim Mas sebagian besar telah mengadopsi mesin industri berbasis mekatronika dan otomatis. Namun demikian dari sekian banyak mesin otomatis yang diimplementasikan pada pekerjaan pengolahan minyak kelapa sawit di industri ini ada pada sisi lain yang menurut peneliti kurang efektif dalam pengerjaan sebuah proses, yaitu terletak pada proses pengendalian pemanasan pada sistem jaket air.

Jaket air adalah sebuah sistem yang berfungsi sebagai pemanas air dengan menggunakan energi kalor uap yang dihasilkan dari sistem pembakaran dimana air yang dipanaskan tadi akan dipompa untuk dialirkan keseluruh pipa luar saluran minyak kelapa sawit yang telah diolah sebelumnya yang dalam kasus ini dilakukan untuk mencegah pembekuan minyak di dalam pipa salurannya.

Pada proses kerja inilah yang masih melakukan proses kerja yang manual sehingga kurang efektif dilakukan, sebab untuk melakukan pekerjaan ini harus tetap *stand by* di dalam ruangan kendali dan secara monoton tetap memperhatikan pada layar monitor berapa keadaan suhu air jaket yang selanjutnya akan dilakukan proses pembukaan atau penutupan *valve* dengan derajat yang disarankan oleh kebutuhan sistem. Disamping itu pada proses pembukaan dan penutupan *valve* yang dilakukan oleh pekerja juga menghabiskan waktu yang lama, sementara kebutuhan sistem pemanasan harus segera dilakukan. Selain itu ketika sesampainya diruangan distribusi tersebut pekerja harus berhati-hati terhadap

tantangan suara ledakan uap yang *over pressure* sebab dapat mengakibatkan serangan jantung bagi pekerja yang bersangkutan.

1.2. Perumusan Masalah

Adapun yang menjadi rumusan masalah dalam penelitian ini adalah :

1. Bagaimana cara membuat rancangan alat cerdas yang dapat mengontrol buka tutup valve pada distribusi uap.
2. Variabel apa yang menjadi kontrol otomatis sehingga dapat memberikan solusi tepat terhadap masalah yang terjadi pada proses pemanasan jaket air dan di dalam ruangan distribusi uap.
3. Bagaimana hasil perbandingan antara sistem yang manual dengan yang dilengkapi alat cerdas hasil penelitian ini.

1.3. Batasan Masalah

Yang menjadi batasan masalah dalam penelitian ini adalah :

1. Proyek akhir yang disajikan ini adalah berbentuk studi perancangan alat dalam bentuk miniatur yang sumber referensinya adalah jelas dan ilmiah.
2. Ruang lingkup pembahasan meliputi komponen *hardware* (perangkat keras) *software* (perangkat lunak) yang meliputi bahasa pemrograman yaitu menggunakan bahasa "C" dan *software* aplikasi untuk memasukkan program adalah *Arduino*.

1.4. Tujuan Penelitian

1. Tujuan penelitian ini adalah merancang sebuah alat cerdas yang dapat mengontrol suhu air jaket yang dialirkan ke seluruh badan pipa saluran minyak kelapa sawit guna menghindari pembekuan.
2. Membuat rancangan sistem derajat pembukaan dan penutupan valve menjadi otomatis sesuai keinginan kondisi jaket air.
3. Antara penerapan metode yang manual dengan metode yang otomatis dibuat sebuah perbandingan terhadap tingkat *safety* pekerja dan mesin industri.

1.5. Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Dapat memudahkan bagi para pekerja untuk melakukan proses pengontrolan suhu air pada jaket.
2. Dapat menghindari kegagalan pada proses pencairan minyak kelapa sawit hasil pengolahan sebelumnya sehingga minyak dalam pipa tetap mengalir.
3. Peluang kondisi pembekuan minyak kelapa sawit hasil pengolahan akan kecil kemungkinan terjadi selama alat yang dirancang tetap berfungsi dengan baik.
4. Mahasiswa dapat mengasah kemampuan sehingga dapat menciptakan inovasi baru yang konteksnya masih berkaitan dengan masalah ini.

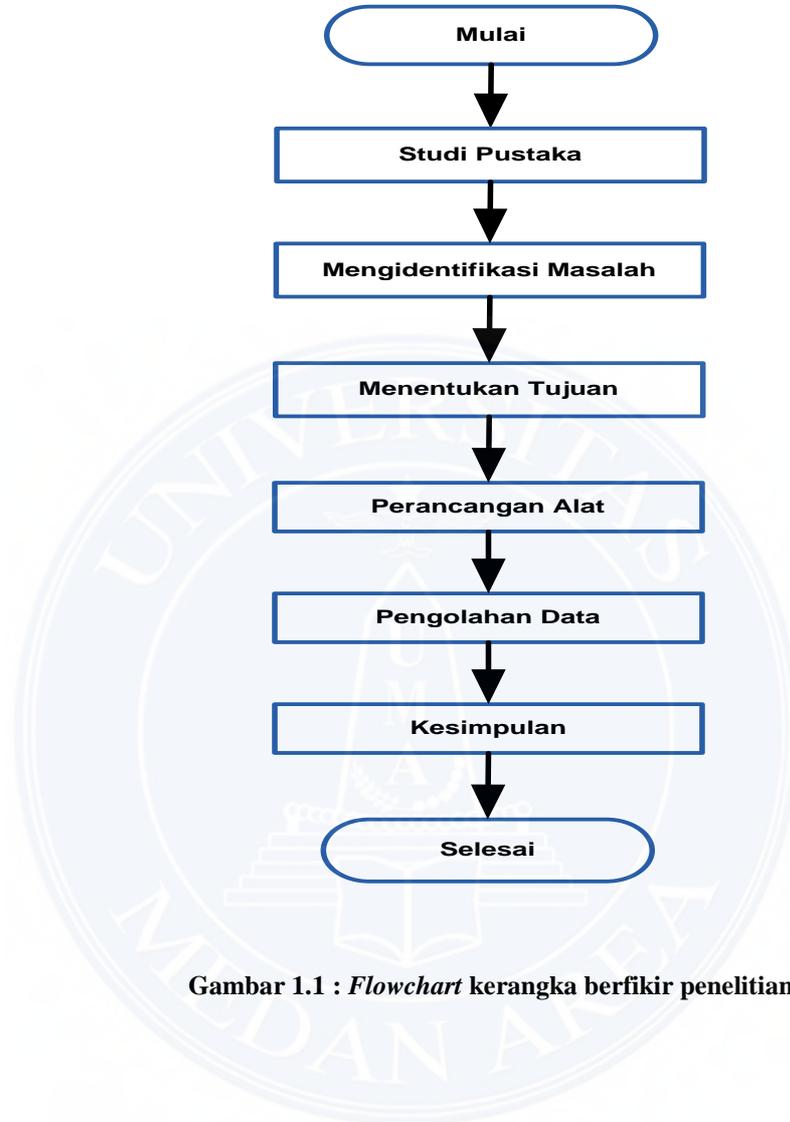
1.6. Sistematika Pembahasan

Sistematika penulisan pada masing-masing bab adalah sebagai berikut :

1. Bab I Pendahuluan
Menjelaskan secara singkat tentang latar belakang penelitian, rumusan masalah penelitian, batasan masalah, tujuan dan manfaat penelitian serta sistematika penulisan.
2. Bab II Teori Penunjang
Bab ini membahas tentang teori-teori pendukung dalam penelitian ini sehingga hasil yang didapatkan lebih optimal.
3. Bab III Metodologi Penelitian
Berisi tentang bagaimana metode penelitian dilakukan, yang meliputi bagaimana cara pengambilan data, dan cara perancangan alat.
4. Bab IV Hasil dan Pembahasan
Berisi penjelasan tentang hasil perancangan alat, alur pengujian alat dan hasil pengujiannya beserta pembahasannya.
5. Bab V Penutup
Berisi tentang kesimpulan dan saran yang didapatkan dari penelitian yang dilakukan.

1.7. Kerangka Berfikir

Berikut Gambar 1.1, yaitu *flowchart* kerangka berfikir dalam penelitian :



Gambar 1.1 : *Flowchart* kerangka berfikir penelitian

BAB II

TEORI PENUNJANG

2.1. Umum

Sistem kontrol derajat pembukaan dan penutupan valve sistem distribusi uap adalah sebuah rancangan alat yang tersusun dari komponen elektronik dan perangkat lunak untuk mengendalikan sebuah sistem pada proses pemanasan air jaket disebuah dapur industri, dimana sistem yang dikendalikan pada proses pemanasan air jaket tersebut adalah kapasitas uap panas dari boiler yang mengalir pada jaket air yang selama ini dikendalikan secara manual dengan menggunakan tangan manusia di ruang distribusi uap berdasarkan suhu jaket air yang tampak dari layar komputer. Dengan rancangan alat ini maka secara otomatis akan melakukan pekerjaan sendiri tanpa bantuan manusia lagi sepenuhnya untuk mengatur derajat pembukaan kapasitas uap panas yang dikopel menggunakan sebuah motor servo berdasarkan informasi dari sebuah sensor pendeteksi suhu jaket air.

2.2. Smart Building

Perkembangan teknologi melaju dengan sangat pesat. Perkembangan teknologi ini merupakan hasil kerja keras dari rasa ingin tahu manusia terhadap suatu hal yang pada akhirnya diharapkan akan mempermudah kehidupan manusia. Salah satu cara untuk mempermudah kehidupan manusia adalah dengan membangun *smart building*.

Smart building sudah menjadi salah satu topik pembahasan dalam karya tulis fiksi ilmiah selama bertahun-tahun, namun baru mulai dicoba untuk diwujudkan pada awal abad ke 20 ketika pemakaian energi listrik telah meluas hingga ke rumah-rumah dan mulai majunya teknologi informasi.

Smart building mengacu pada penggunaan teknologi informasi dan komputer untuk mengendalikan peralatan yang berada di dalam bangunan tersebut (seperti jendela atau lampu). Sistem yang beroperasi dalam *smart building* dapat berupa sistem kontrol jarak jauh yang sederhana dari lampu-lampu hingga sebuah sistem yang berbasis komputer atau mikrokontroler yang memiliki tingkat kecerdasan

yang bervariasi yang secara otomatis mengontrol peralatan yang berada di dalam bangunan.

Elemen dalam sebuah sistem *smart building* terdiri dari sensor (seperti sensor cahaya atau sensor suhu), pengontrol (seperti komputer atau mikrokontroler) dan aktuatur (seperti motor, *relay*, *servo* atau sakelar lampu). Sebuah antarmuka antara manusia dengan mesin diperlukan, agar pemilik bangunan dapat berinteraksi dengan sistem untuk memonitor atau mengontrolnya.

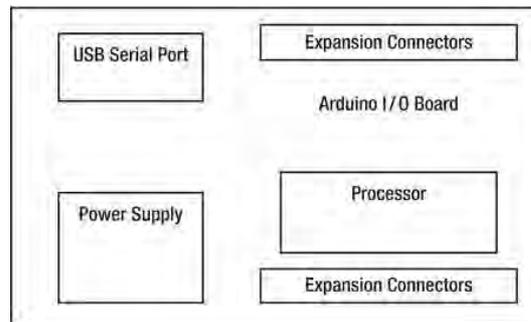
Antarmuka ini dapat berupa terminal khusus atau aplikasi yang berjalan pada telepon genggam atau komputer tablet. Antarmuka tersebut berkomunikasi dengan sistem melalui jaringan kabel atau *wireless* menggunakan satu atau lebih protokol.

2.3. Arduino

Arduino adalah platform pembuatan prototipe elektronik yang bersifat *open-source hardware* yang berdasarkan pada perangkat keras dan perangkat lunak yang fleksibel dan mudah digunakan. Arduino ditujukan bagi para seniman, desainer, dan siapapun yang tertarik dalam menciptakan objek atau lingkungan yang interaktif.

Arduino pada awalnya dikembangkan di Ivrea, Italia. Nama *arduino* adalah sebuah nama maskulin yang berarti teman yang kuat. *Platform arduino* terdiri dari *arduino board*, *shield*, bahasa pemrograman *arduino*, dan *arduino development environment*. *Arduino board* biasanya memiliki sebuah *chip* dasar mikrokontroler Atmel AVR ATmega8 berikut turunannya. Blok diagram *arduino board* yang sudah disederhanakan dapat dilihat pada Gambar 2.1. *Shield* adalah sebuah papan yang dapat dipasang di atas *arduino board* untuk menambah kemampuan dari *arduino board*.

Bahasa pemrograman *arduino* adalah bahasa pemrograman yang umum digunakan untuk membuat perangkat lunak yang ditanamkan pada *arduino board*. Bahasa pemrograman *arduino* mirip dengan bahasa pemrograman C++.



Gambar 2.1 : Blok diagram arduino board

Arduino *Development Environment* adalah perangkat lunak yang digunakan untuk menulis dan meng-*compile* program untuk arduino. *Arduino Development Environment* juga digunakan untuk meng-*upload* program yang sudah di-*compile* ke memori program arduino board.

2.3.1. Arduino Uno

Arduino Uno adalah arduino board yang menggunakan mikrokontroler ATmega328. Arduino Uno memiliki 14 pin digital (6 pin dapat digunakan sebagai output PWM), 6 input analog, sebuah 16 MHz osilator kristal, sebuah koneksi USB, sebuah konektor sumber tegangan, sebuah header ICSP, dan sebuah tombol reset. Arduino Uno memuat segala hal yang dibutuhkan untuk mendukung sebuah mikrokontroler. Hanya dengan menghubungkannya ke sebuah komputer melalui USB atau memberikan tegangan DC dari baterai atau adaptor AC ke DC sudah dapat membuatnya bekerja. Arduino Uno menggunakan ATmega16U2 yang diprogram sebagai *USB-to-serial converter* untuk komunikasi serial ke computer melalui port USB. Tampak atas dari arduino uno dapat dilihat pada Gambar 2.2.

Adapun data teknis board Arduino Uno R3 adalah sebagai berikut:

1. Mikrokontroler : ATmega328
2. Tegangan Operasi : 5V
3. Tegangan Input (recommended) : 7 - 12 V
4. Tegangan Input (limit) : 6-20 V
5. Pin digital I/O : 14 (6 diantaranya pin PWM)
6. Pin Analog input : 6
7. Arus DC per pin I/O : 40 mA

8. Arus DC untuk pin 3.3 V : 150 mA
9. Flash Memory : 32 KB dengan 0.5 KB digunakan untuk bootloader
10. SRAM : 2 KB
11. EEPROM : 1 KB
12. Kecepatan Pewaktuan : 16 Mhz



Gambar 2.2 : Arduino uno

2.3.1.1. Pin Masukan dan Keluaran Arduino Uno

Masing-masing dari 14 pin digital arduino uno dapat digunakan sebagai masukan atau keluaran menggunakan fungsi `pinMode()`, `digitalWrite()` dan `digitalRead()`. Setiap pin beroperasi pada tegangan 5 volt. Setiap pin mampu menerima atau menghasilkan arus maksimum sebesar 40 mA dan memiliki resistor *pull-up* internal (diputus secara default) sebesar 20-30 KOhm. Sebagai tambahan, beberapa pin masukan digital memiliki kegunaan khusus yaitu:

1. Komunikasi serial: pin 0 (RX) dan pin 1 (TX), digunakan untuk menerima(RX) dan mengirim(TX) data secara serial.
2. External Interrupt: pin 2 dan pin 3, pin ini dapat dikonfigurasi untuk memicu sebuah interrupt pada nilai rendah, sisi naik atau turun, atau pada saat terjadi perubahan nilai.

3. Pulse-width modulation (PWM): pin 3,5,6,9,10 dan 11, menyediakan keluaran PWM 8-bit dengan menggunakan fungsi `analogWrite()`.
4. Serial Peripheral Interface (SPI): pin 10 (SS), 11 (MOSI), 12 (MISO) dan 13 (SCK), pin ini mendukung komunikasi SPI dengan menggunakan SPI library.
5. LED: pin 13, terdapat built-in LED yang terhubung ke pin digital 13. Ketika pin bernilai HIGH maka LED menyala, sebaliknya ketika pin bernilai LOW maka LED akan padam.

Arduino Uno memiliki 6 masukan analog yang diberi label A0 sampai A5, setiap pin menyediakan resolusi sebanyak 10 bit (1024 nilai yang berbeda). Secara default pin mengukur nilai tegangan dari ground (0V) hingga 5V, walaupun begitu dimungkinkan untuk mengganti nilai batas atas dengan menggunakan pin AREF dan fungsi `analogReference()`. Sebagai tambahan beberapa pin masukan analog memiliki fungsi khusus yaitu pin A4 (SDA) dan pin A5 (SCL) yang digunakan untuk komunikasi *Two Wire Interface* (TWI) atau *Inter Integrated Circuit* (I2C) dengan menggunakan Wire library.

2.3.1.2. Sumber Daya dan Pin Tegangan Arduino Uno

Arduino uno dapat diberi daya melalui koneksi USB (Universal Serial Bus) atau melalui power supply eksternal. Jika arduino uno dihubungkan ke kedua sumber daya tersebut secara bersamaan maka arduino uno akan memilih salah satu sumber daya secara otomatis untuk digunakan. Power supply external (yang bukan melalui USB) dapat berasal dari adaptor AC ke DC atau baterai. Adaptor dapat dihubungkan ke soket power pada arduino uno. Jika menggunakan baterai, ujung kabel yang dibubungkan ke baterai dimasukkan kedalam pin GND dan Vin yang berada pada konektor Power.

Arduino uno dapat beroperasi pada tegangan 6 sampai 20 Volt. Jika arduino uno diberi tegangan di bawah 7 Volt, maka pin 5V akan menyediakan tegangan di bawah 5 Volt dan arduino uno mungkin bekerja tidak stabil. Jika diberikan tegangan melebihi 12 Volt, penstabil tegangan kemungkinan akan

menjadi terlalu panas dan merusak arduino uno. Tegangan rekomendasi yang diberikan ke arduino uno berkisar antara 7 sampai 12 Volt.

Pin-pin tegangan pada arduino uno adalah sebagai berikut:

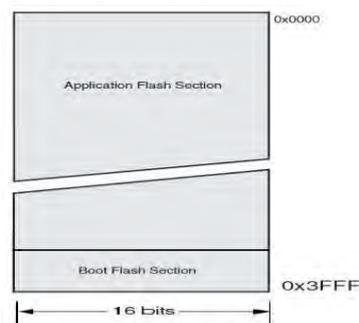
1. Vin adalah pin untuk mengalirkan sumber tegangan ke arduino uno ketika menggunakan sumber daya eksternal (selain dari koneksi USB atau sumber daya yang teregulasi lainnya). Sumber tegangan juga dapat disediakan melalui pin ini jika sumber daya yang digunakan untuk arduino uno dialirkan melalui soket *power*.
2. 5V adalah pin yang menyediakan tegangan teregulasi sebesar 5 Volt berasal dari *regulator* tegangan pada arduino uno.
3. 3V3 adalah pin yang menyediakan tegangan teregulasi sebesar 3,3 Volt berasal dari regulator tegangan pada arduino uno.
4. GND adalah pin *ground*.

2.3.1.3. Peta Memori Arduino Uno

Arduino Uno adalah arduino *board* yang menggunakan mikrokontroler ATmega328. Maka peta memori arduino uno sama dengan peta memori pada mikrokontroler ATmega328.

2.3.1.3.1. Memori Program

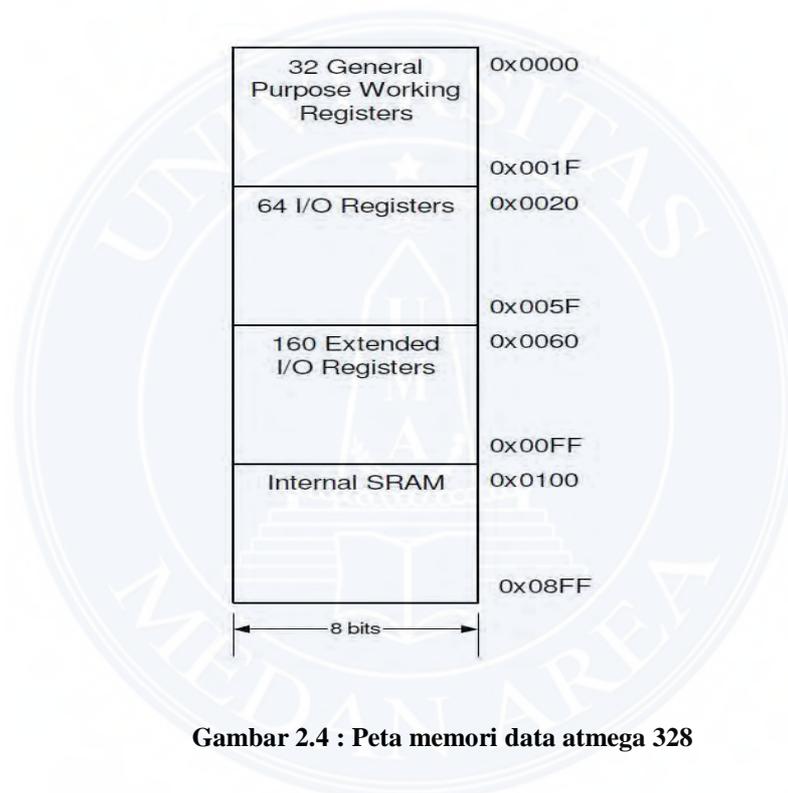
ATMega328 memiliki 32K byte *On-chip In-System Reprogrammable Flash Memory* untuk menyimpan program. Memori *flash* dibagi kedalam dua bagian, yaitu bagian program *bootloader* dan aplikasi seperti terlihat pada Gambar 2.3. *Bootloader* adalah program kecil yang bekerja pada saat sistem dimulai yang dapat memasukkan seluruh program aplikasi ke dalam memori prosesor.



Gambar 2.3 : Peta memori program ATmega 328

2.3.1.3.2. Memori Data

Memori data ATmega328 terbagi menjadi 4 bagian, yaitu 32 lokasi untuk register umum, 64 lokasi untuk register I/O, 160 lokasi untuk register I/O tambahan dan sisanya 2048 lokasi untuk data SRAM internal. Register umum menempati alamat data terbawah, yaitu 0x0000 sampai 0x001F. Register I/O menempati 64 alamat berikutnya mulai dari 0x0020 hingga 0x005F. Register I/O tambahan menempati 160 alamat berikutnya mulai dari 0x0060 hingga 0x00FF. Sisa alamat berikutnya mulai dari 0x0100 hingga 0x08FF digunakan untuk SRAM internal. Peta memori data dari ATmega 328 dapat dilihat pada Gambar 2.4.



Gambar 2.4 : Peta memori data atmega 328

2.3.1.3.3. Memori Data EEPROM

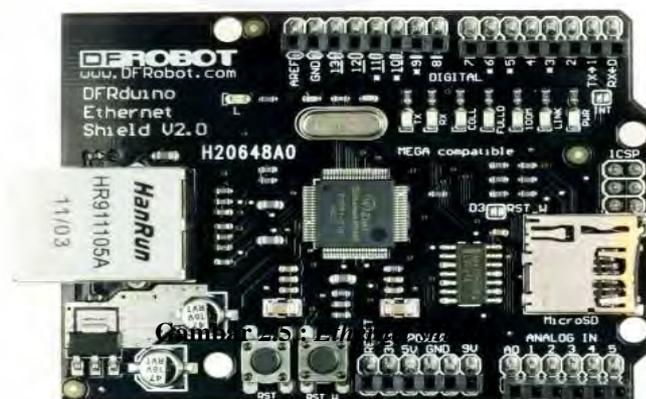
Arduino uno terdiri dari 1 KByte memori data EEPROM. Pada memori EEPROM, data dapat ditulis/dibaca kembali dan ketika catu daya dimatikan, data terakhir yang ditulis pada memori EEPROM masih tersimpan pada memori ini, atau dengan kata lain memori EEPROM bersifat *nonvolatile*. Alamat EEPROM dimulai dari 0x000 hingga 0x3FF.

2.3.2. Arduino Ethernet Shield

Ethernet Shield menambah kemampuan arduino board agar terhubung ke jaringan komputer. Ethernet shield berbasiskan cip ethernet Wiznet W5100. Ethernet *library* digunakan dalam menulis program agar arduino board dapat terhubung ke jaringan dengan menggunakan arduino ethernet shield.

Pada ethernet shield terdapat sebuah slot micro-SD, yang dapat digunakan untuk menyimpan file yang dapat diakses melalui jaringan. Onboard micro-SD card reader diakses dengan menggunakan *SD library*.

Arduino board berkomunikasi dengan W5100 dan SD card menggunakan bus SPI (Serial Peripheral Interface). Komunikasi ini diatur oleh library SPI.h dan Ethernet.h. Bus SPI menggunakan pin digital 11, 12 dan 13 pada Arduino Uno. Pin digital 10 digunakan untuk memilih W5100 dan pin digital 4 digunakan untuk memilih SD card. Pin-pin yang sudah disebutkan sebelumnya tidak dapat digunakan untuk input/output umum ketika kita menggunakan ethernet shield. Karena W5100 dan SD card berbagi bus SPI, hanya salah satu yang dapat aktif pada satu waktu. Jika kita menggunakan kedua perangkat dalam program kita, hal ini akan diatasi oleh *library* yang sesuai. Jika kita tidak menggunakan salah satu perangkat dalam program kita, kiranya kita perlu secara eksplisit men-*deselect*-nya. Untuk melakukan hal ini pada SD card, set pin 4 sebagai output dan menuliskan logika tinggi padanya, sedangkan untuk W5100 yang digunakan adalah pin 10. DFRduino Ethernet shield adalah sebuah *clone* dari arduino Ethernet shield yang dibuat oleh DFRobot. Penampakan DFRduino Ethernet shield dapat dilihat pada Gambar 2.5.



Gambar 2.5 : Arduino Ethernet Shield

2.3.3. Bahasa Pemrograman Arduino

Arduino board merupakan perangkat yang berbasis mikrokontroler. Perangkat lunak (*software*) merupakan komponen yang membuat sebuah mikrokontroler dapat bekerja. Arduino board akan bekerja sesuai dengan perintah yang ada dalam perangkat lunak yang ditanamkan padanya. Bahasa Pemrograman Arduino adalah bahasa pemrograman utama yang digunakan untuk membuat program untuk arduino board. Bahasa pemrograman arduino menggunakan bahasa pemrograman C sebagai dasarnya.

Karena menggunakan bahasa pemrograman C sebagai dasarnya, bahasa pemrograman arduino memiliki banyak sekali kemiripan, walaupun beberapa hal telah berubah.

2.3.3.1. Struktur

Setiap program dalam arduino board terdiri dari dua fungsi utama yaitu `setup()` dan `loop()`. Instruksi yang berada dalam fungsi `setup()` dieksekusi hanya sekali, yaitu ketika arduino board pertama kali dihidupkan. Biasanya instruksi yang berada pada fungsi `setup()` merupakan konfigurasi dan inisialisasi dari arduino board. Instruksi yang berada pada fungsi `loop()` dieksekusi berulang-ulang hingga arduino board dimatikan (catu daya diputus). Fungsi `loop()` merupakan tugas utama dari arduino board. Jadi setiap program yang menggunakan bahasa pemrograman arduino memiliki struktur sebagai berikut:

```
void set up ()
{
  // perintah-perintah untuk konfigurasi dan inisialisasi arduino board
}
Void loop ()
{
  // Perintah-perintah utama arduino board
}
```

Program di atas dapat dianalogikan dalam bahas C sebagai berikut ;

```

Void set up (void) ; //prototype fungsi set up
Void loop (void) ; // prototype fungsi loop
Int main (void) {
Set up () ; //
While (1) {
Loop () ; ulangi terus menerus
}
Return 0 ; // bagian ini tidak akan pernah di eksekusi
}

```

2.3.3.2. Konstanta

Konstanta adalah variable yang sudah ditetapkan sebelumnya dalam bahasa pemrograman arduino. Konstanta digunakan agar program lebih mudah untuk dibaca dan dimengerti. Konstanta dibagi menjadi 3 kelompok yaitu:

1. Konstanta yang digunakan untuk menunjukkan tingkat logika (konstanta Boolean), yaitu *true* dan *false*
2. Konstanta untuk menunjukkan keadaan pin, yaitu HIGH dan LOW
3. Konstanta untuk menunjukkan fungsi pin, yaitu INPUT, INPUT_PULLUP dan OUTPUT

Konstanta yang digunakan untuk menunjukkan benar atau salah dalam bahasa pemrograman arduino adalah *true* dan *false*. *False* lebih mudah didefinisikan daripada *true*. *False* didefinisikan sebagai 0(nol). *True* sering didefinisikan sebagai 1(satu), yang mana hal ini benar, tetapi *true* memiliki definisi yang lebih luas. Setiap integer yang bukan nol adalah *true* dalam pengertian Boolean. Jadi -2, 3 dan -100 semuanya didefinisikan sebagai *true*, juga dalam pengertian Boolean. Tidak seperti konstanta yang lain *true* dan *false* diketik dengan menggunakan huruf kecil.

Ketika membaca atau menulis ke sebuah pin digital, terdapat hanya dua nilai yang dapat diberikan atau diterima, yaitu HIGH dan LOW. HIGH memiliki arti yang berbeda tergantung apakah sebuah pin dikonfigurasi menjadi masukan atau

keluaran. Ketika pin dikonfigurasi sebagai masukan dengan fungsi `pinMode()`, lalu kemudian dibaca dengan fungsi `digitalRead()`, mikrokontroler akan melaporkan nilai HIGH jika tegangan yang ada pada pin tersebut berada pada tegangan 3 volt atau lebih.

Ketika sebuah pin dikonfigurasi sebagai masukan, dan kemudian dibuat bernilai HIGH dengan fungsi `digitalWrite()`, maka resistor *pull-up* internal dari chip ATmega akan aktif, yang akan membawa pin masukan ke nilai HIGH kecuali pin tersebut ditarik (*pull-down*) ke nilai LOW oleh sirkuit dari luar.

Ketika pin dikonfigurasi sebagai keluaran dengan fungsi `pinMode()`, dan diset ke nilai HIGH dengan fungsi `digitalWrite()`, maka pin berada pada tegangan 5 volt. Dalam keadaan ini, pin tersebut dapat memberikan arus, sebagai contoh, untuk menhidupkan LED yang terhubung seri dengan resistor dan ground, atau pin lain yang dikonfigurasi sebagai keluaran dan diberi nilai LOW.

Sama seperti HIGH, LOW juga memiliki arti yang berbeda bergantung pada konfigurasi pin. Ketika pin dikonfigurasi sebagai masukan, maka mikrokontroler akan melaporkan nilai LOW jika tegangan yang terdapat pada pin berada pada tegangan 2 volt atau kurang. Ketika pin dikonfigurasi sebagai keluaran dan diberi nilai LOW maka pin berada pada tegangan 0 volt.

Setiap pin pada arduino dapat dikonfigurasi sebagai masukan, masukan dengan resistor *pull-up* atau keluaran. Untuk mengkonfigurasi fungsi pin pada arduino digunakan konstanta `INPUT`, `INPUT_PULLUP` dan `OUTPUT`. Pin arduino yang dikonfigurasi sebagai masukan dengan fungsi `pinMode()` dikatakan berada dalam kondisi berimpedansi tinggi. Pin yang dikonfigurasi sebagai masukan memiliki permintaan yang sangat kecil kepada sirkuit yang di-sampling-nya, setara dengan sebuah resistor 100 Megaohm dipasang seri dengan pin tersebut. Hal ini membuat pin tersebut berguna untuk membaca sensor, tetapi tidak untuk menhidupkan sebuah LED.

Cip ATmega pada arduino memiliki resistor *pull-up* internal (resistor yang terhubung ke sumber tegangan secara internal) yang dapat digunakan. Untuk menggunakan resistor *pull-up* internal ini kita menggunakan konstanta `INPUT_PULLUP` pada fungsi `pinMode()`.

Pin yang dikonfigurasi menjadi sebuah keluaran dikatakan berada dalam kondisi berimpedansi rendah. Hal ini berarti pin tersebut dapat menyediakan sejumlah besar arus ke sirkuit yang lain. Pin pada ATmega mampu menyediakan arus hingga 40 mA.

2.3.3.3. Fungsi Masukan dan Keluaran Digital

Arduino memiliki 3 fungsi untuk masukan dan keluaran digital pada arduino board, yaitu `pinMode()`, `digitalWrite()` dan `digitalRead()`.

Fungsi `pinMode()` mengkonfigurasi pin tertentu untuk berfungsi sebagai masukan atau keluaran. Sintaksis untuk fungsi `pinMode()` adalah sebagai berikut:

`pinMode(pin, mode)`

Parameter: pin = angka dari pin digital yang akan dikonfigurasi

mode = konfigurasi yang diinginkan (INPUT, INPUT_PULLUP dan OUTPUT).

Fungsi `digitalWrite()` berfungsi untuk memberikan nilai HIGH atau LOW suatu digital pin. Sintaksis untuk fungsi `digitalWrite()` adalah sebagai berikut:

`digitalWrite(pin, value)`

Parameter: pin = angka dari pin digital yang akan dikonfigurasi value = nilai yang diinginkan (HIGH atau LOW).

Fungsi `digitalRead()` bertujuan untuk membaca nilai yang ada pada pin arduino uno. Sintaksis untuk fungsi `digitalRead()` adalah sebagai berikut:

`digitalRead(pin)`

Parameter: pin = angka dari pin digital yang akan dibaca

Berikut ini adalah contoh penggunaan fungsi masukan dan keluaran digital dalam sebuah program:

```
int ledPin = 13; // LED terhubung ke pin digital 13
int inPin = 7; // pushbutton terhubung ke pin digital 7
int val = 0; // variable untuk menyimpan sebuah nilai

void setup()
{
  pinMode(ledPin, OUTPUT); // set pin digital 13 sebagai keluaran
  pinMode(inPin, INPUT); // set pin digital 13 sebagai masukan
}

void loop()
{
  val = digitalRead(inPin); // baca nilai pin input
  digitalWrite(ledPin, val); // sets LED sesuai dengan nilai val}

```

2.3.3.4. Progmem

Terkadang program dibuat memerlukan memori yang lebih besar, misalnya kita perlu menyimpan banyak teks dalam sebuah program penampil LCD. Hal ini bisa kita atasi dengan menyimpan data pada flash memori (memori program). Besar memori program pada arduino uno adalah 32 KByte.

Kata kunci PROGMEM adalah sebuah *variable modifier*, PROGMEM hanya bisa digunakan untuk tipe data yang di definisikan pada *library pgmspace.h*. PROGMEM menjelaskan pada kompiler “letakkan informasi berikut pada memori program”, menggantikan SRAM, dimana biasanya informasi tersebut disimpan.

PROGMEM adalah bagian dari *library pgmspace.h*, jadi kita perlu memasukkan *library pgmspace.h* ke program kita. Sintaksis untuk menggunakan PROGMEM adalah sebagai berikut:

```
dataType variableName[] PROGMEM = {dataInt0, dataInt1, ...};
dataType - tipe variable
variableName - nama array data

```

Karena PROGMEM adalah sebuah *variable modifier*, tidak ada aturan yang baku

untuk tempat peletakkannya. Kompiler arduino menerima semua definisi yang ditunjukkan di bawah.

```
dataType variableName[] PROGMEM = {};
dataType PROGMEM variableName[] = {};
PROGMEM dataType variableName[] = {};
```

PROGMEM juga bisa digunakan untuk menyimpan satu variabel, tetapi karena kerepotan dalam penggunaannya, sebaiknya hanya digunakan untuk menyimpan data yang sangat banyak, yang biasanya lebih mudah menyimpannya di dalam array.

Dalam menggunakan PROGMEM kita perlu melakukan 2 langkah prosedur. Pertama adalah menyimpan data pada memori program, kemudian kita menggunakan fungsi khusus, yang juga didefinisikan di *library pgmspace.h*, untuk membaca data tersebut dari memori program dan menuliskannya ke SRAM, sehingga kita dapat melakukan sesuatu dengan data tersebut.

Berikut adalah tipe data yang didukung oleh *library pgmspace*.

```
Prog_char- assigned char (1 byte) -127 sampai 128
Prog_uchar- an unsigned char (1 byte) 0 sampai 255
Prog_int16_t- a signed int (2 bytes) -32,767 sampai 32,768
Prog_uint16_t- an unsigned int (2 bytes) 0 sampai 65,535
Prog_int32_t- a signed long (4 bytes) -2,147,483,648 sampai *
                2,147,483,647.
Prog_uint32_t- an unsigned long (4 bytes) 0 sampai 4,294,967,295
```

Berikut ini adalah contoh program untuk menggunakan PROGMEM ;

```
#include <avr/pgmspace.h>
// simpan beberapa unsigned int
PROGMEM prog_uint16_t charSet[] = { 65000, 32796, 16843, 10,
11234};
// simpan beberapa karakter
prog_uchar signMessage[] PROGMEM = {"UNIVERSITAS SUMATERA
UTARA"};
unsigned int displayInt;
```

```

int k; // variabel counter
char myChar;
// baca int
displayInt = pgm_read_word_near(charSet + k)
// baca karakter
myChar = pgm_read_byte_near(signMessage + k)

```

Di bawah ini adalah contoh program untuk menyimpan array string pada program memori. Karena string sendiri adalah sebuah array, contoh di bawah ini juga merupakan contoh penggunaan array dua dimensi.

```

#include <avr/pgmspace.h>
prog_char string_0[] PROGMEM = "String 0";
prog_char string_1[] PROGMEM = "String 1";
prog_char string_2[] PROGMEM = "String 2";
prog_char string_3[] PROGMEM = "String 3";
prog_char string_4[] PROGMEM = "String 4";
prog_char string_5[] PROGMEM = "String 5";

//buat table untuk menunjukkan string.
PROGMEM const char *string_table[] =
{
string_0,
string_1,
string_2,
string_3,
string_4,
string_5 };
char buffer[30]; //pastikan buffer cukup untuk menyimpan string
void setup()
{
Serial.begin(9600);
}
void loop()
{

```

*/*menggunakan table string pada memori program membutuhkan fungsi special untuk mengambil data tersebut.*

fungsi strcpy_P ("buffer") menyalin sebuah string dari program memori sebuah string di SRAM ("buffer").

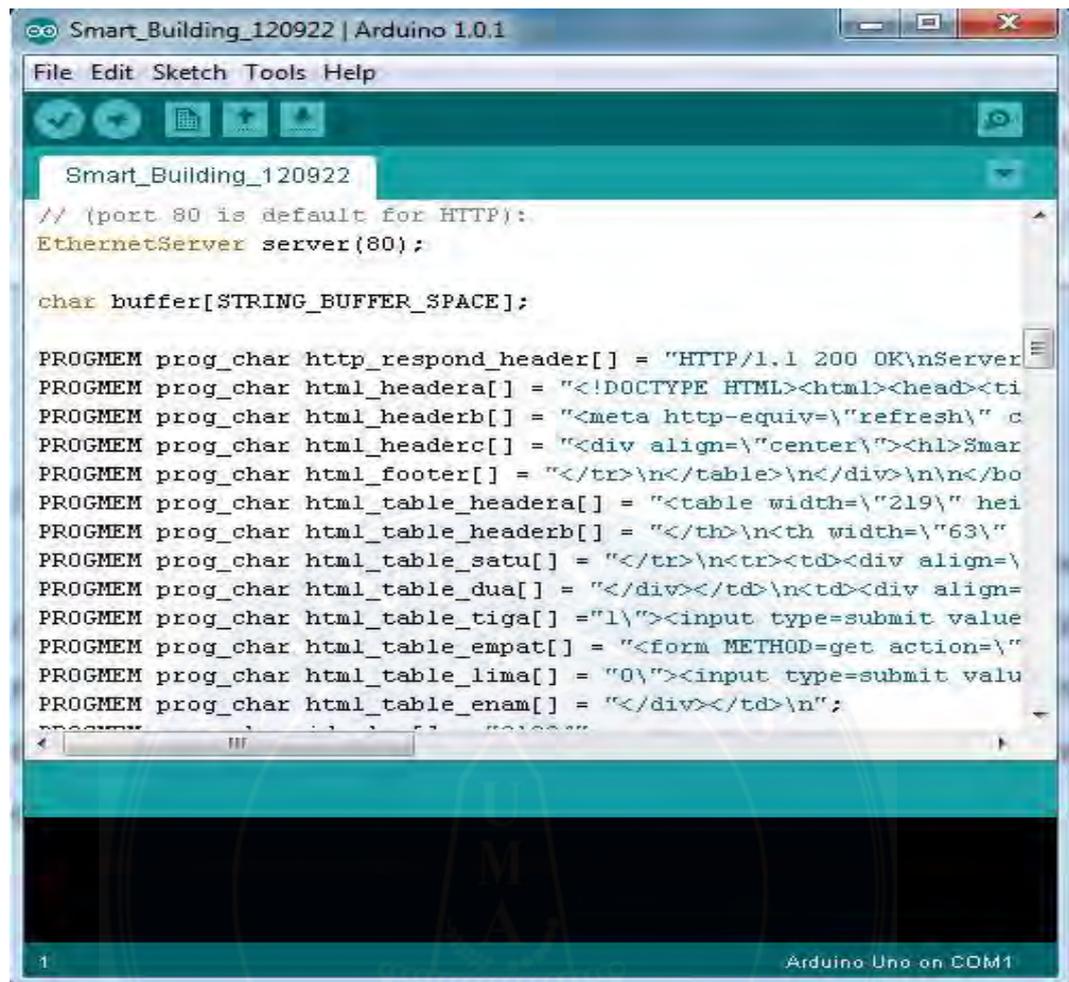
Pastikan bahwa SRAM cukup untuk menampung setiap string yang diambil dari memori program/*

```
for (int i = 0; i < 6; i++)
{
strcpy_P(buffer, (char*)pgm_read_word(&(string_table[i]))); //
salin string ke buffer
Serial.println( buffer );
delay( 500 );
}
}
```

2.3.4. Arduino Development Environment

Arduino *Development Environment* terdiri dari editor teks untuk menulis kode, sebuah area pesan, sebuah konsol, sebuah *toolbar* dengan tombol-tombol untuk fungsi yang umum dan beberapa menu. *Arduino Development Environment* terhubung ke arduino board untuk meng-*upload* program dan juga untuk berkomunikasi dengan arduino board.

Perangkat lunak yang ditulis menggunakan *Arduino Development Environment* disebut *sketch*. *Sketch* ditulis pada editor teks. *Sketch* disimpan dengan file berekstensi .ino. Area pesan memberikan memberikan informasi dan pesan error ketika kita menyimpan atau membuka *sketch*. Konsol menampilkan output teks dari *Arduino Development Environment* dan juga menampilkan pesan error ketika kita mengkompile *sketch*. Pada sudut kanan bawah dari jendela *Arduino Development Environment* menunjukkan jenis board dan port seri yang sedang digunakan. Tombol *toolbar* digunakan untuk mengecek dan mengupload *sketch*, membuat, membuka atau menyimpan *sketch*, dan menampilkan *serial monitor*.



Gambar 2.6 : Arduino development environment

Berikut ini adalah tombol-tombol *toolbar* serta fungsinya:

 *Verify*

mengecek error pada code program.

 *Upload*

meng-*compile* dan meng-upload program ke Arduino board.



New

membuat sketch baru.



Open

menampilkan sebuah menu dari seluruh *sketch* yang berada di dalam *sketchbook*.



Save

menyimpan sketch.



Serial Monitor

membuka serial *monitor*.

Dalam lingkungan arduino digunakan sebuah konsep yang disebut *sketchbook*, yaitu tempat standar untuk menumpukan program (*sketch*). *Sketch* yang ada pada *sketchbook* dapat dibuka dari menu **File > Sketchbook** atau dari tombol *open* pada *toolbar*. Ketika pertama kali menjalankan arduino development environment, sebuah direktori akan dibuat secara otomatis untuk tempat penyimpana *sketchbook*. Kita dapat melihat atau mengganti lokasi dari direktori tersebut dari menu **File > Preferences**.

Serial monitor menampilkan data serial yang sedang dikirim dari arduino board. Untuk mengirim data ke board, masukkan teks dan klik tombol *send* atau tekan *enter* pada *keyboard*.

Sebelum meng-*upload* program, kita perlu mensetting jenis board dan port serial yang sedang kita gunakan melalui menu **Tools > Board** dan **Tools > Serial Port**. Pemilihan board berguna untuk mengeset parameter (contohnya: kecepatan mikrokontroler dan *baud rate*) yang digunakan ketika meng-*compile* dan meng-*upload sketch*.

Setelah memilih board dan port serial yang tepat, tekan tombol *upload* pada

toolbar atau pilih menu **File > Upload**. Arduino board akan me-*reset* secara otomatis dan proses *upload* akan dimulai. Pada kebanyakan board, LED RX dan TX akan berkedip ketika program sedang di-*upload*. *Arduino development environment* akan menampilkan pesan ketika proses *upload* telah selesai, atau menampilkan pesan error.

Ketika sedang meng-*upload* program, arduino bootloader sedang digunakan, Arduino bootloader adalah sebuah program kecil yang telah ditanamkan pada mikrokontroler yang berada pada arduino board. Bootloader ini mengizinkan kita meng-*upload* program tanpa menggunakan perangkat keras tambahan.

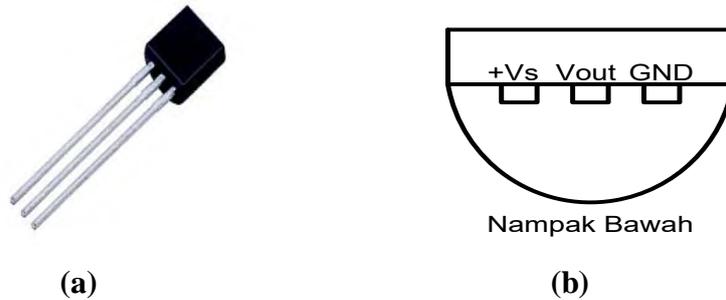
2.4. Sensor

Sensor atau sering disebut juga dengan *Transducer* merupakan piranti yang mentransform (mengubah) suatu nilai (energi) fisik ke nilai fisik yang lain, menghubungkan antara fisik nyata dan piranti elektronika yang berguna untuk *monitoring*, *controlling*, dan proteksi. Sensor adalah peralatan yang digunakan untuk merubah suatu besaran fisik menjadi besaran listrik sehingga dapat dianalisa dengan rangkaian listrik tertentu.

2.4.1. LM35

LM35 adalah IC yang khusus digunakan untuk sensor temperatur/suhu yang hasilnya cukup linier. LM35 tidak memerlukan kalibrasi eksternal ataupun *timing* khusus. LM35 merupakan sensor temperatur yang paling banyak digunakan untuk praktek, karena selain harganya yang cukup murah linearitasnya juga lumayan bagus. Sensor ini dapat beroperasi pada kisaran -55°C hingga $+150^{\circ}\text{C}$ dan mempunyai karakteristik yang linear yaitu pada $10\text{ mV}/^{\circ}\text{C}$.

Sensor ini sangat sederhana namun memiliki presisi yang cukup tinggi. Sensor ini hanya memiliki tiga buah kaki. Kaki pertama IC LM35 dihubungkan ke sumber daya, kaki kedua sebagai *output* dan kaki ketiga dihubungkan ke *ground*. Gambar 2.6. berikut adalah gambar bentuk fisik dan susunan kaki dari sensor LM35.



Gambar 2.7 : (a) Bentuk fisik LM35

(b) Nampak bagian bawah LM35

Disamping bentuk fisiknya yang sangat sederhana, sensor LM35 juga memiliki beberapa kelebihan. Adapun kelebihan yang dimiliki LM35 adalah sebagai berikut:

1. Dapat dikalibrasi langsung ke dalam besaran Celcius.
2. Faktor skala linier $+10\text{mV}/^\circ\text{C}$.
3. Tingkat akurasi $0,5^\circ\text{C}$ saat suhu kamar (25°C).

Prinsip kerja dari rangkaian sensor ini adalah berdasarkan perubahan suhu atau panas yang terjadi di sekitarnya. Dimana apabila suhu di sekitar sensor semakin tinggi maka nilai resistansi di dalam rangkaian sensor yang berfungsi sebagai pengatur besar arus ke *input non-inverting* akan semakin besar dan sebaliknya. Jadi apabila nilai resistansi dari pengatur arus ke *non-inverting* kecil maka arus dari transistor yang sebelumnya telah mengalami proses penguatan terlebih dahulu akan terbagi ke *ground* dan *non-inverting*, sehingga dengan semakin besarnya nilainya resistansi pengatur arus tersebut menyebabkan tegangan yang masuk ke penguat (A2) semakin besar pula. Jadi transistor hanya berfungsi sebagai pengatur besar arus yang dibantu oleh penguat (A1), dan lambang (i) hanya merupakan pengatur tahanan. Untuk cara kerja dari rangkaian dalam sensor LM35 ini dapat dilihat pada Gambar 2.8. berikut ini.

$$V_{LM35} = \text{Suhu} \times 10 \text{ mV} \dots\dots\dots(2.1)$$

Dimana:

Suhu = Perubahan temperatur yang terjadi di sekitar sensor.

Secara prinsip sensor LM35 akan melakukan penginderaan pada saat perubahan suhu setiap suhu 1°C akan menunjukkan tegangan sebesar 10 mV. Pada penempatannya LM35 dapat ditempelkan dengan perekat atau dapat pula disemen pada permukaan akan tetapi suhunya akan sedikit berkurang sekitar $0,01^{\circ}\text{C}$ karena akan terserap pada suhu permukaan tersebut.

IC LM35 juga memiliki karakteristik, berikut ini adalah karakteristik dari sensor LM35 :

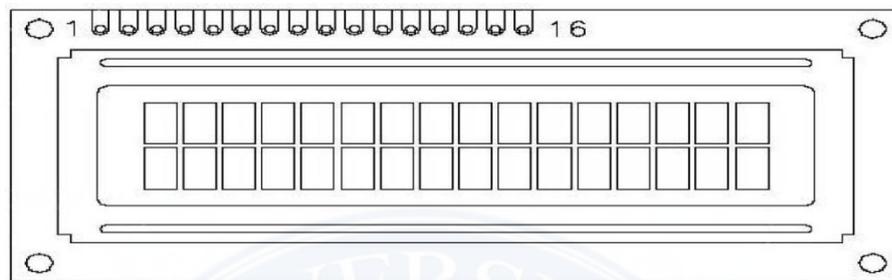
1. Memiliki sensitivitas suhu, dengan faktor skala linier antara tegangan dan suhu $10 \text{ mV}/^{\circ}\text{C}$, sehingga dapat dikalibrasi langsung dalam *Celcius*.
2. Memiliki ketepatan atau akurasi kalibrasi yaitu $0,5^{\circ}\text{C}$ pada suhu 25°C .
3. Memiliki jangkauan maksimal operasi suhu antara -55°C sampai $+150^{\circ}\text{C}$.
4. Bekerja pada tegangan 4 sampai 30 volt.
5. Memiliki arus rendah yaitu kurang dari $60 \mu\text{A}$.
6. Memiliki pemanasan sendiri yang rendah (*low-heating*) yaitu kurang dari $0,1^{\circ}\text{C}$.
7. Memiliki impedansi keluaran yang rendah yaitu $0,1\Omega$ untuk beban 1 mA.
8. Memiliki ketidak linieran hanya $\pm 1/4^{\circ}\text{C}$.

Sensor LM35 bekerja dengan mengubah besaran suhu menjadi besaran tegangan. Tegangan ideal yang keluar dari LM35 mempunyai perbandingan 100°C setara dengan 1 volt. Sensor ini mempunyai pemanasan diri (*self heating*) kurang dari $0,1^{\circ}\text{C}$, dapat dioperasikan dengan menggunakan *power supply* tunggal dan dapat dihubungkan antar muka (*interface*) rangkaian kontrol yang sangat mudah.

2.5. LCD (Liquid Crystal Display) Karakter 16x2

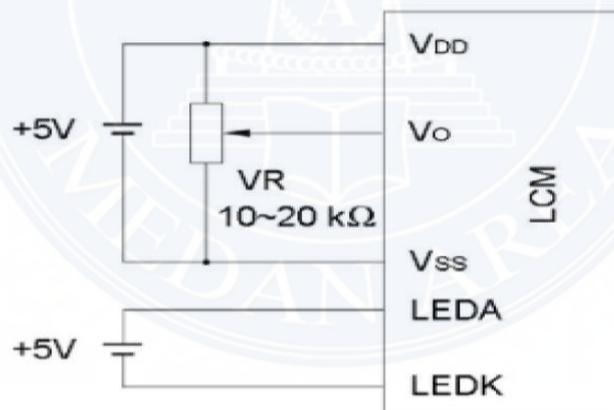
LCD (*liquid crystal display*) adalah piranti output untuk sistem komputer atau mikrokontroler yang dapat menampilkan karakter standar yang telah tersimpan dalam ROM pada LCD tersebut. Karakter yang dapat ditampilkan sebanyak 32

karakter yang tersusun dalam 16 kolom dan dua baris, sehingga sering disebut *LCD 16x2*. *LCD* memiliki 16 pin dan memerlukan tegangan 5V DC. Gambar 2.9 menunjukkan bentuk dan susunan pin *LCD* dan keterangan dari setiap pin diberikan pada Tabel 2.1.



Gambar 2.9 : Susunan pin *LCD* karakter 16x2

Sebelum menggunakan modul *LCD* ini, *power supply* 5V DC harus diberikan sebagai sumber arusnya seperti yang ditunjukkan Gambar 2.10.



Gambar 2.10 : Hubungan *power supply* ke *LCD*

Tabel 2.1: Keterangan pin LCD karakter 16x2

No pin	Nama pin	Level	Fungsi
1	Vss	-	Ground
2	Vdd	-	Power supply untuk chip
3	Vo	0V – 5V	Power supply untuk LCD
4	RS	H/L	Register Selection
5	R/W	H/L	Read atau Write
6	E	H→L	Enable
7	DB0	H/L	Data Bit 0
8	DB1	H/L	Data Bit 1
9	DB2	H/L	Data Bit 2
10	DB3	H/L	Data Bit 3
11	DB4	H/L	Data Bit 4
12	DB5	H/L	Data Bit 5
13	DB6	H/L	Data Bit 6
14	DB7	H/L	Data Bit 7
15	LEDA	-	Power supply untuk LED (+)
16	LEDK	-	Power supply untuk LED (-)

Untuk menampilkan karakter pada *LCD* ada beberapa algoritma yang harus dijalankan. Algoritma ini dilakukan dengan memberikan nilai logika 0 atau 1 pada pin yang bersangkutan. Algoritma untuk penampilan karakter ini antara lain :

1. Mengatur *display LCD*, *cursor* dan *blink*.

Tabel 2.2 : Pengaturan *display LCD*

E	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1→0	0	0	0	0	0	0	1	D	C	B

Keterangan:

D: D=0 *display* mati dan D=1 *display* hidup

C: C=0 *cursor* mati dan C=1 *cursor* hidup

B: B=0 *blink* mati dan B=1 *blink* hidup

2. Mengatur karakter yang akan ditampilkan

Tabel 2.3 : Pengaturan *display LCD*

E	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1→0	0	0	Nilai berdasarkan CGROM							

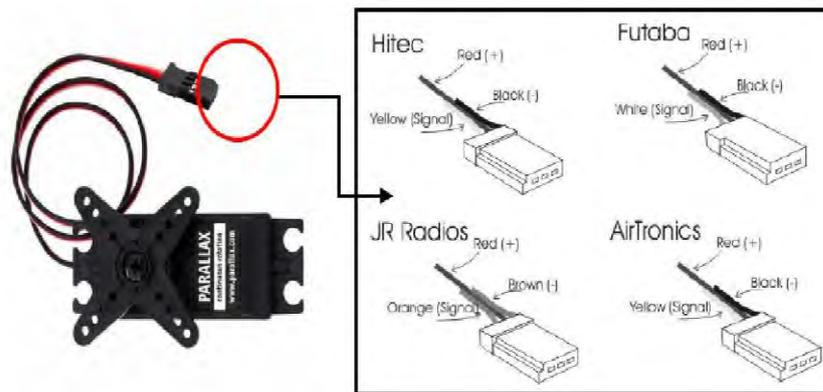
Keterangan: Nilai dari DB0 sampai DB7 yang dimasukkan nantinya akan mengatur tampilan karakter yang berkorespondensi dengan kode karakter CGROM pada chip LCD. Kode karakter CGROM diberikan pada gambar 2.11 yang terdiri dari kolom yang menunjukkan DB6 – DB7 dan baris yang menunjukkan DB0 – DB3.

Upper 4bit Lower 4bit		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	CG RAM (1)																
0001	(2)																
0010	(3)																
0011	(4)																
0100	(5)																
0101	(6)																
0110	(7)																
0111	(8)																
1000	(1)																
1001	(2)																
1010	(3)																
1011	(4)																
1100	(5)																
1101	(6)																
1110	(7)																
1111	(8)																

Gambar2.11 Kode Karakter CGROM pada LCD

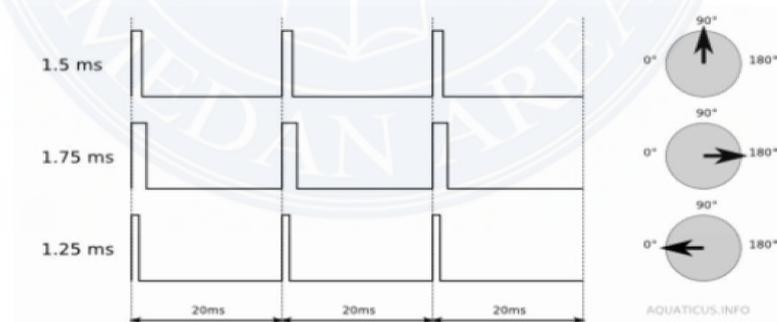
2.6. Motor Servo Standar

Motor servo adalah motor DC yang dilengkapi dengan sistem kontrol. Sistem kontrol ini akan memberikan umpan balik posisi perputaran motor dari 0 sampai 180 derajat. Disamping itu motor ini juga memiliki torsi relatif cukup kuat. Gambar 2.12 menunjukkan penampang dan pengkabelan dari motor servo. Sistem pengkabelan motor servo terdiri atas 3 bagian, yaitu Vcc, Gnd, dan Kontrol (PWM= *Pulse Width Modulation*). Pemberian PWM pada motor servo akan membuat servo bergerak pada posisi tertentu dan kemudian berhenti (kontrol posisi).



Gambar 2.12 : Motor servo dan konfigurasi pin

Prinsip utama dari pengendalian motor servo adalah pemberian nilai PWM pada kontrolnya. Frekuensi PWM yang digunakan pada pengontrol motor servo selalu 50 Hz sehingga pulsa dihasilkan setiap 20 ms. Lebar pulsa akan menentukan posisi servo yang dikehendaki. Pemberian lebar pulsa 1,5 ms akan membuat motor servo berputar ke posisi netral (90 derajat), lebar pulsa 1,75 ms akan membuat motor servo berputar mendekati posisi 180 derajat, dan dengan lebar pulsa 1,25 ms motor servo akan bergerak ke posisi 0 derajat. Gambar 2.13 berikut memperlihatkan hubungan antara lebar pulsa PWM dengan arah putaran motor servo.



Gambar 2.13 : Hubungan lebar pulsa PWM dengan arah putaran motor servo

2.7. Valve (katup)

Pengertian dalam suatu industri terutama yang bergerak dalam pengolahan liquid, tentu memiliki sistem perpipaan yang berfungsi sebagai tempat mengalirnya liquid. Setiap rangkaian pipa pastinya memiliki suatu alat yang digunakan untuk mengatur jumlah aliran agar proses pengolahan dapat berjalan sesuai dengan yang ditentukan. Alat tersebut disebut dengan valve atau sering juga disebut katup, tentu sudah tidak asing lagi bagi kita, contoh sederhananya yaitu kran air yang hampir kita gunakan setiap hari.

Valve atau katup adalah sebuah perangkat yang terpasang pada sistem perpipaan, yang berfungsi untuk mengatur, mengontrol dan mengarahkan laju aliran fluida dengan cara membuka atau menutup sebagian aliran fluida. Katup atau valve memiliki peran penting dalam suatu industri seperti industri migas yang meliputi pengaliran ke dalam kolom destilasi dan mengontrol pengapian pada furnace.

Valve dapat dioperasikan secara manual, baik dengan menggunakan pegangan, tuas pedal dan lain sebagainya, selain dioperasikan secara manual valve dapat juga dioperasikan secara otomatis dengan menggunakan prinsip perubahan aliran, tekanan dan suhu. Perubahan tersebut akan mempengaruhi diafragma, pegas ataupun piston sehingga secara otomatis akan menggerakkan katup dengan sistem buka tutup.

2.7.1. Fungsi Valve

Terdapat berbagai macam jenis valve, beserta dengan kriteria penggunaannya masing-masing. Berikut fungsi-fungsi utama valve:

1. Untuk menutup dan membuka aliran dengan syarat, ketika terbuka memiliki hambatan aliran dan pressure loss yang minimum. Contohnya: gate, ball, plug dan butterfly valve.
2. Untuk mengatur aliran, dengan cara menahan aliran dengan perubahan arah atau menggunakan suatu hambatan bisa juga dengan kombinasi keduanya.
3. Untuk mencegah aliran balik (back flow), biasanya menggunakan check valve (lift check dan swing check). Valve ini akan tetap terbuka dan akan tertutup apabila terdapat aliran yang berlawanan arah.

4. Untuk mengatur tekanan, dalam beberapa aplikasi valve, tekanan yang masuk (line pressure) harus dikurangi untuk mencapai tekanan yang diinginkan. Biasanya menggunakan pressure-reducing valve atau regulator.
5. Untuk pressure relief dengan menggunakan relief valve dan safety valve. Relief valve digunakan untuk mengatasi bila adanya tekanan yang berlebihan yang dapat mengganggu proses aliran bahkan kegagalan proses. Sedangkan safety valve menggunakan per (spring loaded), valve ini akan membuka jika tekanan melebihi batas yang sudah ditentukan.

2.7.2. Jenis - jenis Valve

Berikut berbagai macam jenis valve dengan dengan karakteristik dan cara kerja masing-masing:

1. Gate Valve

Jenis ini didesain untuk membuka dan menutup aliran dengan cara tertutup rapat dan terbuka penuh sehingga valve ini tidak cocok untuk mengatur debit aliran karena kurang akurat dalam hal mengontrol jumlah aliran. Berikut ini adalah Gambar 2.14 yang memperlihatkan bentuk fisik dari gate valve :



Gambar 2.14 : Gate valve

2. Plug Valve

Memiliki fungsi yang sama dengan gate valve yaitu dengan menutup atau membuka aliran secara keseluruhan. Namun beberapa pengaplikasian valve ini digunakan untuk mengontrol aliran seperti pada pengaliran gas. Dan berikut ini adalah Gambar 2.15 yang memperlihatkan bentuk fisik dari plug valve :



Gambar 2.15 : Plug valve

3. Ball Valve

Jenis ini dapat dioperasikan pada fluida bertemperatur -450°F hingga -500°F , ball valve merupakan tipe quick opening valve yang hanya memerlukan $1/4$ putaran dari posisi tertutup penuh ke terbuka penuh. Berikut adalah Gambar 2.16, yang memperlihatkan bentuk fisik dari ball valve :



Gambar 2.16 : Ball valve

4. Globe valve

Aliran dalam valve berubah arah sehingga menghasilkan friksi yang cukup besar meskipun dalam keadaan terbuka lebar. Jenis valve ini cukup penting bila digunakan untuk penutupan yang rapat terutama pada aliran gas. Berikut adalah Gambar 2.17 yang memperlihatkan bentuk fisik globe valve :



Gambar 2.17 : Globe valve

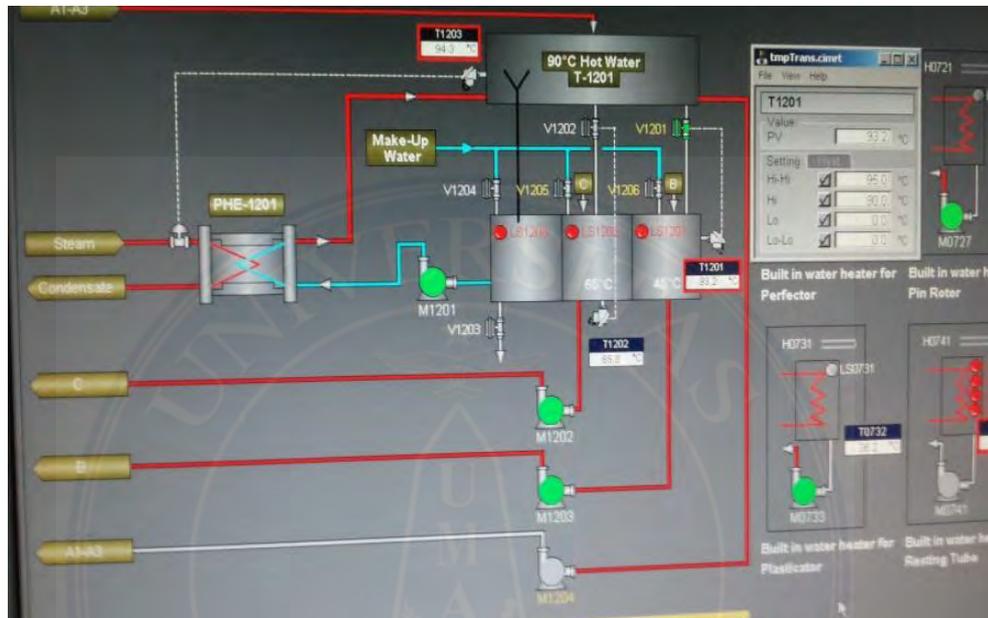
2.8. Jaket Air

Jaket air atau air jaket adalah proses pemanasan air dalam suhu normal yang berasal dari air pam, dimana air tersebut dipanaskan di dalam bak air, kemudian air di dalam bak tersebut dipanaskan oleh steam uap. Fungsi dari air jaket adalah untuk memanaskan minyak melalui jalur pipa yang dilapisi casing, bertujuan agar minyak tidak mengalami pembekuan. Berikut adalah Gambar 2.18 yang memperlihatkan bentuk fisik dari air jaket :



Gambar 2.18 : Jaket air

Cara pemanasan air jaket di perusahaan tempat peneliti mendapatkan ide judul tugas akhir ini adalah dengan cara membuka dan menutup valve steam secara manual, dimana tinggi dan rendah suhu air jaket tersebut bisa dilihat pada layar monitor computer. Berikut adalah Gambar 2.19 yang memperlihatkan layar jendela aplikasinya :



Gambar 2.19 : Layar jendela aplikasi pemantauan suhu air jaket

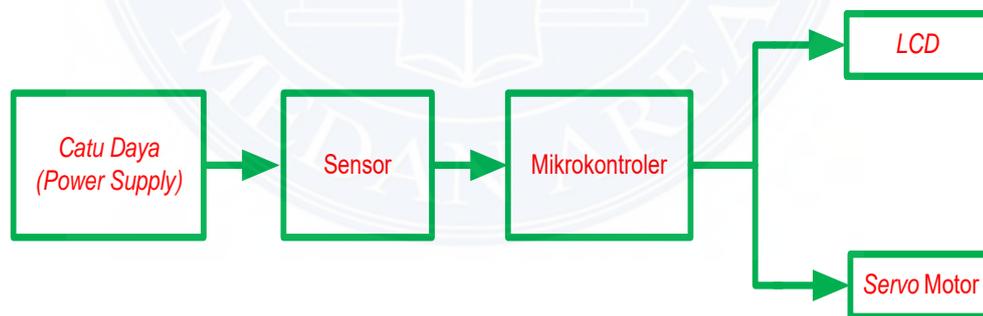
BAB III METODELOGI PENELITIAN

3.1. Lokasi Penelitian

Kegiatan penelitian dilakukan di Laboratorium Dasar Digital Jurusan Teknik Elektro Fakultas Teknik Universitas Medan Area Jl. Kolam No. 1 Medan Estate, Sumatera Utara.

3.2. Blok Diagram

Sistem kontrol derajat pembukaan dan penutupan *valve* distribusi uap yang akan dirancang secara garis besar ditunjukkan pada blok diagram Gambar 3.1 berikut. *Power supply*, merupakan sumber tenaga utama pada alat yang akan dirancang dan dari *power supply* akan diteruskan ke sistem pengendali mikrokontroler dan sensor, kemudian sensor akan membaca dan mengubah besaran fisis yang dideteksi menjadi elektrik dan dikomunikasikan kepada mikrokontroler dan selanjutnya data tersebut akan diproses oleh mikrokontroler dan menghasilkan output untuk mengaktifkan beban yang dikontrol yaitu *servo motor*.



Gambar 3.1 : Blok diagram sistem penelitian

3.3. Alat, Bahan, dan Desain Penelitian

3.3.1. Alat dan Bahan

Adapun deskripsi alat dan bahan yang akan digunakan dalam perancangan sistem otomatis derajat pembukaan dan penutupan valve sistem distribusi uap adalah :

1. Mikrokontroler
2. Sensor
3. Catu Daya
4. Resistor
5. *LCD*
6. Kabel, timah, *PCB*, lem plastik, dan solder
7. Bor listrik
8. Triplek
9. Satu unit komputer
10. Penyedot timah
11. Multimeter
12. Acrelic dan Bak kaleng

3.3.1.1. Penetapan Komponen

Sebelum alat (*prototype*) dirancang maka dilakukan terlebih dahulu penetapan komponen beserta merek ataupun *type* agar alat yang dirancang nantinya dapat terbentuk dengan optimal serta dapat bekerja sesuai tujuan perancangan dalam penelitian ini. Berikut adalah Tabel 3.2, yang menyajikan jenis dan spesifikasi komponen yang digunakan :

Tabel 3.2 : Daftar komponen elektronik

Nama Komponen	Type/Merek	Jumlah
Sistem Minimum Mikrorokontroler ATMega 328	Arduino UNO	1 Buah
Motor Servo	Tower Pro/MG995	1 Buah
Sensor Suhu	LM35	1 Buah
LCD	2 x 16 / 1632	1 Buah
Power Supply	NMJ-1210	1 Buah
Resistor	Fix Resistor/ ¼ Watt	1 Buah
Kabel Data	USB (Universal Serial Bus	1 Buah

Berikut ini adalah Tabel 3.3 yang menyajikan daftar bahan beserta merek/type yang digunakan dalam perancangan alat :

Tabel 3.3 : Daftar bahan

Nama Bahan	Type/Merek	Jumlah
Triplek	Melamin	12 x 15 cm
Acrelic	Bening	10 x 2 cm
Timah	Tinol/Campuran Borax	1 Rol
Lem bakar	Lem Lilin/Glue Gun	2 Batang

Berikut ini adalah Tabel 3.4 yang menyajikan daftar alat beserta merek/type yang digunakan dalam perancangan alat :

Tabel 3.4 : Daftar alat

Nama Alat	Type/Merek	Jumlah
Gerinda		1 Buah
Tools Mechanic	Fukung	1 Set
Mistar	<i>Stationery</i>	1 Buah
Multimeter	DT-830B/ Digital	1 Buah
Komputer	<i>Compac</i>	1 Buah
Solder	MD-80/200 W	1 Buah
Penyedot Timah	ED-300	1 Buah
Bor Listrik	Gun Drill/McCULLOCH	1 Buah
Mata Bor	TS 601200	1 Set

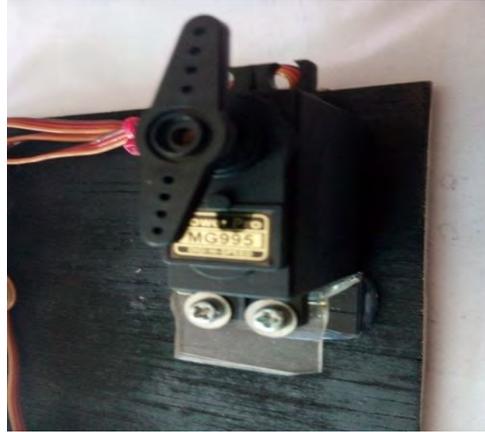
3.3.1.2. Spesifikasi Komponen

Spesifikasi adalah perincian jenis dan level komponen yang akan dipakai. Berikut ini akan dijelaskan spesifikasi masing-masing komponen elektrikal yang digunakan dalam perancangan sistem kontrol derajat pembukaan dan penutupan valve sistem distribusi uap pada proses pemanasan air jaket :

1. Spesifikasi Motor Servo

1. Weight- 55.0g
2. Dimension 40.7*19.7*42.9mm
3. Stall torque 8.5kg/cm(4.8V), 10kg/cm(6V)
4. Operating speed 0.20sec/60degree(4.8v), 0.16sec/60degree(6V)
5. Operating voltage 4.8-7.2V
6. Temperature range 0°C_ 55°C
7. Dead band width 5us

Berikut ini adalah Gambar 3.2, yang memperlihatkan bentuk fisik motor servo yang digunakan dalam penelitian ini :

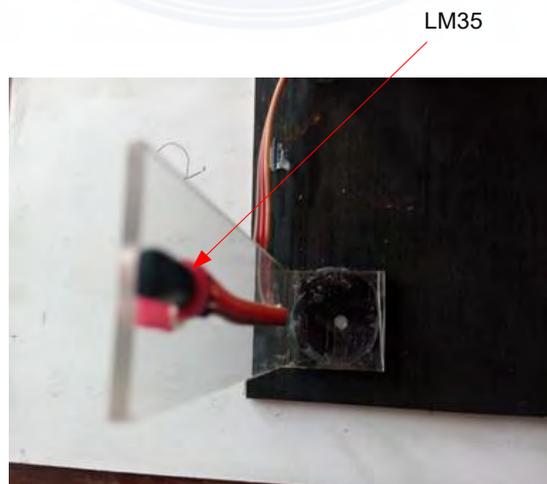


Gambar 3.2 : Bentuk fisik motor servo Tower Pro/MG995

3. Spesifikasi Sensor Suhu

- a. Jenis / tipe : LM35
- b. Satuan kalibrasi : °Celcius
- c. Tegangan kerja : 4 – 30 Vdc
- d. Range suhu : -55°C– 150°C
- e. Linieritas : +10 mV/°C
- f. Akurasi : 0,5°C
- g. Tegangan Output : -1 Vdc s/d + 6 Vdc
- h. Impedansi Output : 0,1 : pada beban 1 mA
- i. Low Self Heating : 0,08°C

Berikut ini adalah Gambar 3.3, yang menampilkan bentuk fisik sensor Suhu LM35 yang digunakan dalam penelitian ini :



Gambar 3.3 : Sensor suhu LM35

4. Spesifikasi Power Supply

Adaptor : AC – DC

Input : AC 110-240V/50Hz

Output : 12 VDC /1A

MADE IN CHINA

Berikut ini adalah Gambar 3.4 yang menampilkan bentuk fisik dari power supply yang digunakan dalam penelitian ini



Gambar 3.4 : Bentuk fisik power supply

5. Spesifikasi LCD

Tabel 3.5 di bawah ini adalah tabel yang menampilkan spesifikasi dari LCD yang digunakan dalam penelitian ini :

Tabel 3.5 : Spesifikasi LCD 2 x 16

Pin	Symbol and Functions
1	GND
2	VCC (+5V)
3	Contrast adjust
4	(RS) → 0 = Instruction input / 1 = Data input
5	(R/W) → 0 = Write to LCD Module / 1 = Read from LCD module
6	(E) → Enable signal

7	(DB0) → Data Pin 0
8	(DB1) → Data Pin 1
9	(DB2) → Data Pin 2
10	(DB3) → Data Pin 3
11	(DB4) → Data Pin 4
12	(DB5) → Data Pin 5
13	(DB6) → Data Pin 6
14	(DB7) → Data Pin 7
15	(VB+) → Back Light (+5V)
16	(VB-) → Back Light (GND)

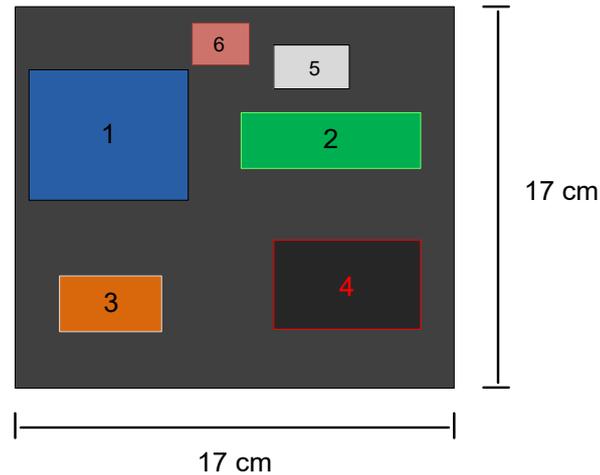
Berikut adalah Gambar 3.5 yang menampilkan bentuk fisik LCD 2x16 yang digunakan dalam penelitian :



Gambar 3.5 : Bentuk fisik LCD x 16 /1632

3.4. Desain Penelitian

Adapun yang menjadi deskripsi tata letak komponen desain penelitian sistem yang akan dirancang adalah dapat dilihat pada Gambar 3.7 berikut :



Gambar 3.6: Tata letak komponen desain penelitian

Keterangan Gambar 3.7 :

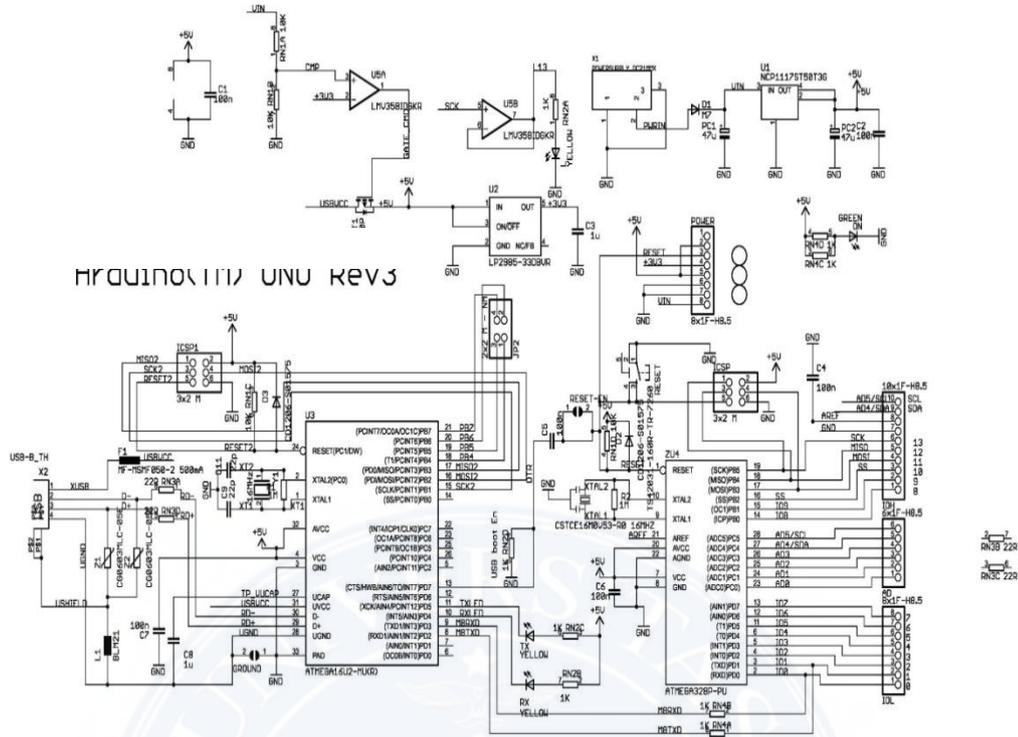
- | | | | |
|----|--------------------|----|-----------------------------|
| 1. | sistem Arduino Uno | 4. | Motor Servo |
| 2. | Display LCD 2 x 16 | 5. | IC Catu Daya |
| 3. | Sensor Suhu LM35 | 6. | Port Input dari Power Suply |

3.5. Skema Rangkaian dan Instalasi Alat yang Dirancang

Berikut ini adalah Gambar 3.8, Gambar 3.9, Gambar 3.10 dan Gambar 3.11 yang menampilkan masing-masing skema rangkaian dan instalasi sistem-sistem pembentuk alat yang dirancang yang meliputi sistem input, sistem pengolahan data, dan output.

1. Skema rangkaian sistem Arduino Uno

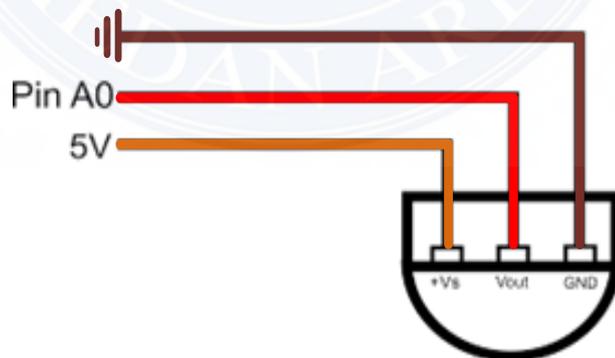
Gambar 3.8 di bawah ini adalah gambar yang menunjukkan skema rangkaian dari sistem minimum Arduino Uno beserta mikrokontroler ATmega 328 :



Gambar 3.7 : Skema rangkaian Arduino Uno

2. Skema instalasi sensor LM35

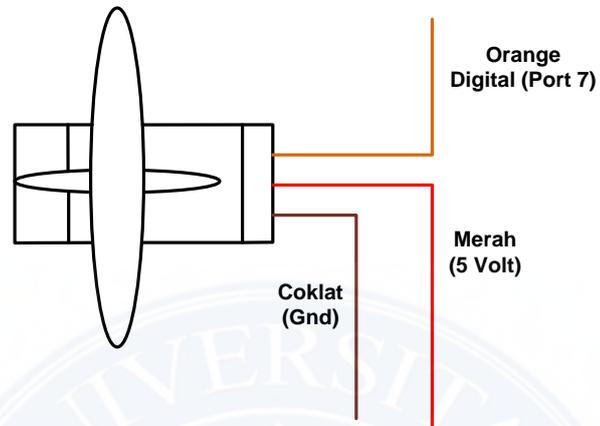
Gambar 3.9 di bawah ini adalah gambar yang menunjukkan skema instalasi dari sensor suhu LM35 :



Gambar 3.8 : Skema instalasi sensor suhu LM35

3. Skema rangkaian motor servo

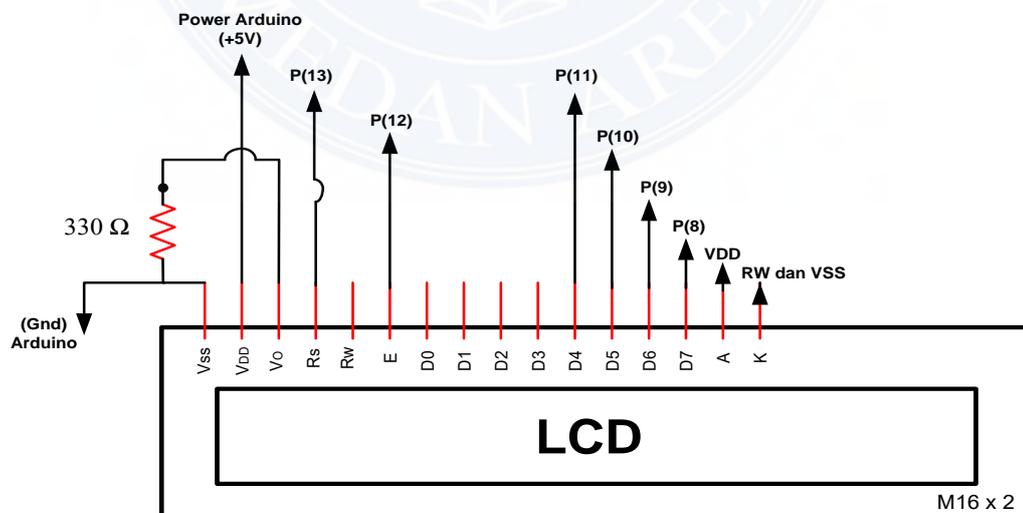
Gambar 3.10 di bawah ini adalah gambar yang menunjukkan skema instalasi dari motor servo :



Gambar 3.9 : Skema instalasi motor servo

4. Skema instalasi *display LCD*

Gambar 3.11 di bawah ini adalah gambar yang menunjukkan skema instalasi dari *display LCD* :



Gambar 3.10 : Skema instalasi LCD

3.7. Metode Pengujian

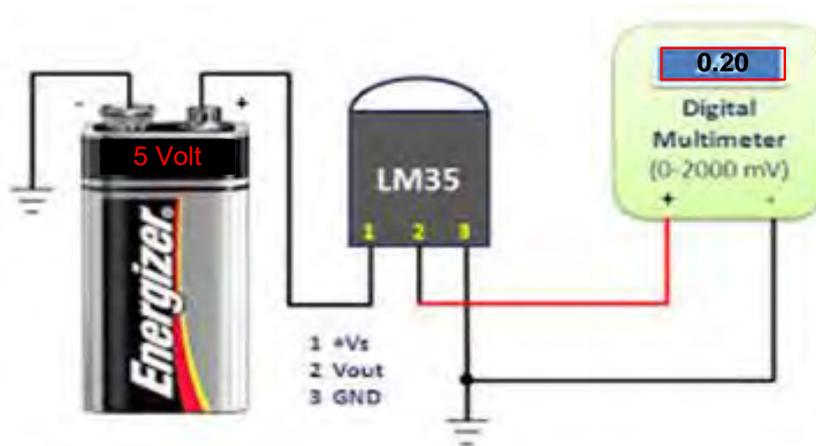
Setelah seluruh komponen pembentuk sistem kontrol derajat pembukaan dan penutupan valve sistem distribusi uap pada proses pemanasan air jaket selesai seperti pada Gambar 3.12 di atas, maka selanjutnya melakukan proses pengujian. Namun sebelum melakukan pengujian secara keseluruhan, terlebih dahulu dalam penelitian ini yang diuji adalah fungsi sensor LM35 dengan tujuan apakah sensor dapat bekerja mendeteksi suhu dengan baik sebelum dihubungkan dengan sistem minimum arduino uno.

Sebagai sensor yang mendeteksi temperatur, IC LM35 merupakan salah satu komponen utama yang harus diuji kondisi dan tingkat kehandalannya. Karena hasil kerja dari LM35 akan sangat mempengaruhi hasil kerja sistem secara keseluruhan. Pengujian ini dilakukan untuk menguji waktu respon dari sensor LM35 terhadap perubahan suhu di sekitarnya dan dibandingkan dengan termometer biasa (standar).

Pengujian ini dilakukan melalui dua tahap, yaitu pengujian respon sensor LM35 terhadap kenaikan suhu di sekitarnya dan pengujian respon sensor LM35 terhadap penurunan suhu di sekitarnya. Dan untuk setiap tahap dilakukan tiga kali percobaan untuk memperoleh data kumulatif sehingga diperoleh hasil rata-rata yang lebih akurat. Hasil percobaan ini akan dibandingkan dengan hasil percobaan yang sama yang dilakukan terhadap termometer biasa.

Sesuai dengan karakteristik sensor LM35 bahwa setiap suhu 1°C akan menunjukkan tegangan 10 mV pada *ouput*-nya, maka proses pengukuran perubahan suhu sensor LM35 ini dilakukan dengan mengukur setiap perubahan tegangan pada *output* sensor tersebut. Pengukuran tegangan *output* ini dilakukan dengan menggunakan alat ukur dengan cara menghubungkan probe (+) Voltmeter dengan kaki V_{OUT} dan probe (-) Voltmeter dengan kaki *Ground* pada sensor.

Gambar 3.13 dan Tabel 3.6 yang menunjukkan skema dan bentuk pengujian sensor LM 35 :



Gambar 3.12 : Skema pengujian sensor LM35

**Tabel 3.6 : Bentuk pengujian sensor LM35 terhadap kenaikan suhu
(yang akan dicari)**

Percobaan		Waktu (s)									
		0	20	40	60	80	100	120	140	160	180
I	Teg. (V)	-	-	-	-	-	-	-	-	-	-
	Suhu ($^{\circ}$ C)	-	-	-	-	-	-	-	-	-	-
II	Teg. (V)	-	-	-	-	-	-	-	-	-	-
	Suhu ($^{\circ}$ C)	-	-	-	-	-	-	-	-	-	-
III	Teg. (V)	-	-	-	-	-	-	-	-	-	-
	Suhu ($^{\circ}$ C)	-	-	-	-	-	-	-	-	-	-

**Tabel 3.7 : Bentuk pengujian sensor LM35 terhadap penurunan suhu
(yang akan dicari)**

Percobaan		Waktu (s)									
		0	20	40	60	80	100	120	140	160	180
I	Teg. (V)	-	-	-	-	-	-	-	-	-	-
	Suhu ($^{\circ}$ C)	-	-	-	-	-	-	-	-	-	-
II	Teg. (V)	-	-	-	-	-	-	-	-	-	-
	Suhu ($^{\circ}$ C)	-	-	-	-	-	-	-	-	-	-
III	Teg. (V)	-	-	-	-	-	-	-	-	-	-
	Suhu ($^{\circ}$ C)	-	-	-	-	-	-	-	-	-	-

Setelah pengujian terhadap sensor LM35 selesai dilakukan baik saat kenaikan suhu maupun penurunan suhu, maka langkah selanjutnya adalah pengujian terhadap termometer biasa (standar) baik pengujian saat kenaikan suhu maupun penurunan suhu.

Pengujian terhadap termometer ini perlu dilakukan sebagai perbandingan terhadap hasil pengukuran sensor apakah hasil pengukuran sensor tersebut sesuai atau tidak. Proses pengujiannya juga sama dengan pengujian terhadap sensor LM35 baik waktu yang digunakan maupun jumlah percobaan pengujiannya. Suhu awal pengukuran juga dimulai dari 10°C.

Bentuk data percobaan pengujian termometer biasa terhadap kenaikan suhu yang akan dicari dapat dilihat seperti pada Tabel 3.8 dan 3.9 berikut ini.

Tabel 3.8 :Bentuk pengujian termometer terhadap kenaikan suhu

Percobaan		Waktu (s)									
		0	20	40	60	80	100	120	140	160	180
I	Suhu (°C)	-	-	-	-	-	-	-	-	-	-
II	Suhu (°C)	-	-	-	-	-	-	-	-	-	-
III	Suhu (°C)	-	-	-	-	-	-	-	-	-	-

Tabel 3.9 :Bentuk pengujian termometer terhadap penurunan suhu

Percobaan		Waktu (s)									
		0	20	40	60	80	100	120	140	160	180
I	Suhu (°C)	-	-	-	-	-	-	-	-	-	-
II	Suhu (°C)	-	-	-	-	-	-	-	-	-	-
III	Suhu (°C)	-	-	-	-	-	-	-	-	-	-

3.7.1. Pengujian secara Keseluruhan

Selanjutnya, melakukan pengujian seluruh sistem yang telah secara keseluruhan terangkai seperti pada Gambar 3.14 di atas, dan pengujian ini adalah pengujian yang merupakan tujuan dari perancangan alat atau penelitian dalam tugas akhir ini. Berikut ini adalah Tabel 3.10 yang menyajikan bentuk pengujian seluruh sistem yang dicari :

Tabel 3.10 : Bentuk pengujian seluruh sistem

No.	Tampilan Suhu (°C) pada LCD	Suhu Pada Termometer	Derajat Putaran Motor Servo	Derajat Putaran Buka/Tutup Valve	Waktu (S)	Tegangan Sensor LM35 (mV)
1°C°C°°SmV
2°C°C°°SmV
3°C°C°°SmV
4°C°C°°SmV

Adapun cara pengujian secara keseluruhan yang merupakan tujuan dalam pembuatan alat ini adalah :

1. Menyalurkan sumber tenaga listrik pada seluruh sistem dengan cara mengecok *power supply (AC to DC adaptor)* ke sumber PLN 220 VAC.
2. Kemudian, memanaskan sensor LM35 dengan cara mendekatkan suhu solder listrik ataupun mendekatkan suhu api dari berbagai sumber api (misalnya : dari mancis, korek api, atau sejenisnya).
3. Memperhatikan suhu yang ditampilkan oleh *display* atau *LCD*, kemudian memperhatikan parameter sebagai berikut :
 - a). Jika suhu yang tampil pada LCD adalah $\leq 35^{\circ}\text{C}$ maka motor berputar $^{\circ}0$
 - b). Jika suhu yang tampil pada LCD adalah $> 36^{\circ}\text{C}$ dan $\leq 49^{\circ}\text{C}$ maka motor berputar $^{\circ}55$
 - c). Jika suhu yang tampil pada LCD adalah $>50^{\circ}\text{C}$ dan $\leq 64^{\circ}\text{C}$ maka motor berputar $^{\circ}95$

3.7.2. Program Sistem Arduino

Adapun rincian program bahasa C yang dimasukkan pada sistem Arduino adalah sebagai berikut :

```
#include <Servo.h> // library servo
#include<LiquidCrystal.h> // library LCD

Servo myServo; // set nama servo
LiquidCrystal myLcd (13,12,11,10,9,8); // set pin LCD
int tempSense = A0; // Set pin LM 35
int temp = 0; // nilai awal temperature
const byte degreeSymbol = B11011111; // simbol derajat

void setup() {
  myServo.attach(7); // set pin servo
  myServo.write(0); // set posisi awal servo
  pinMode(tempSense,INPUT); // set sebagai input/output
  Lcd(); // panggil fungsi LCD
}

void loop() {
  float suhu = analogRead(tempSense); // baca pin analog
  suhu = (5.0 * suhu * 100.0)/1024.0; // proses pembacaan ADC
  temp = (int)suhu; // konversi data

  Lcd(); // panggil fungsi lcd
  delay(500); // delay

  //Cek kondisi suhu untuk perputaran servo
  if(temp <=36 ){
    myServo.write(0);
  }

  else if(temp >= 36 && temp <=39 ){
    myServo.write(40);
  }
  else if(temp >= 40 && temp <=49 ){
    myServo.write(55);
  }
}
```

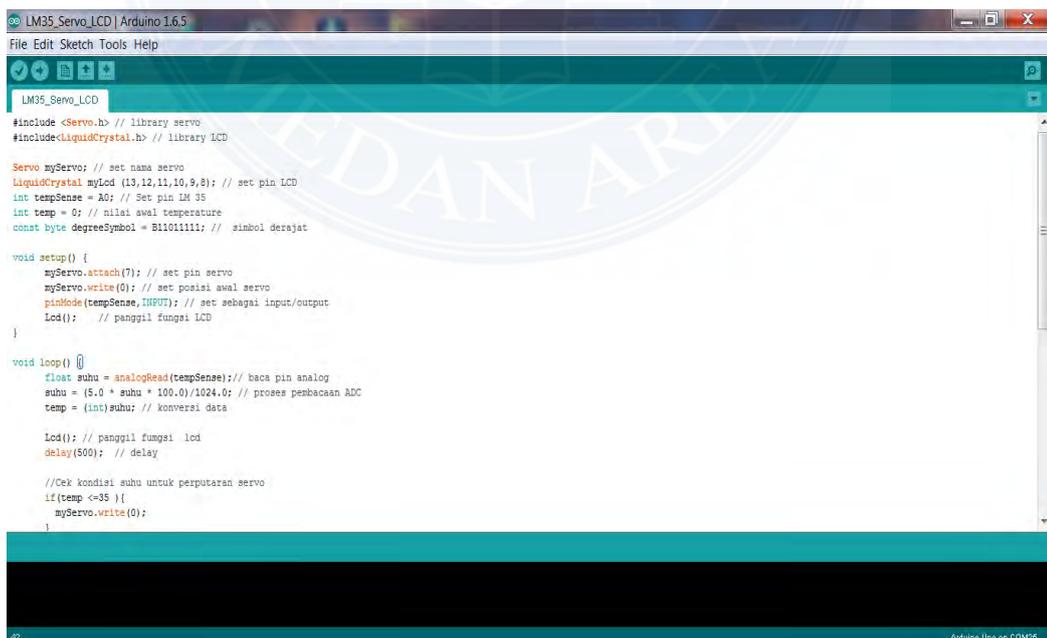
```

else if(temp >= 50 && temp <=64 ){
    myServo.write(95);
}
else if(temp >= 65 ){
    myServo.write(140);
}
}

void Lcd(){ // coding fungsi mylcd "bebas "
    myLcd.begin(16, 2);
    myLcd.clear();
    myLcd.setCursor(0,0);
    myLcd.print("CONTROL SERVO");
    myLcd.setCursor(0,1);
    myLcd.print("Suhu : " + (String)temp + " ");
    myLcd.write(degreeSymbol);
    myLcd.print("C");
}

```

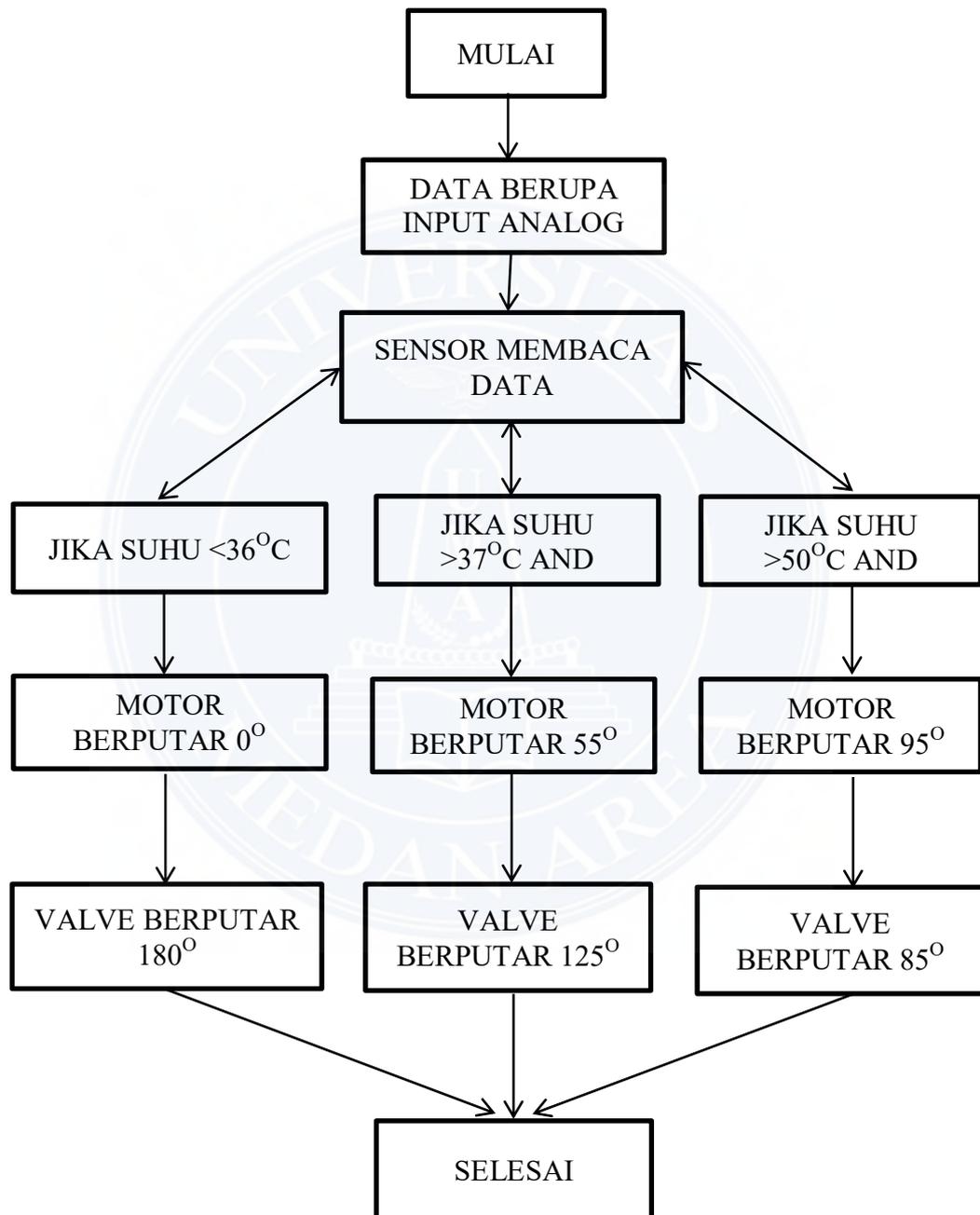
Gambar 3.14 berikut ini adalah gambar yang menampilkan model layar jendela aplikasi untuk menuliskan program bahasa “C” di atas.



Gambar 3.13 : Jendela aplikasi penulisan program

3.7.3. Flowchart Sistem Kerja Alat

Berikut adalah Gambar 3.15 yang memperlihatkan alur kerja sistem kontrol derajat pembukaan dan penutupan valve sistem distribusi uap pada proses pemanasan air jaket.



Gambar 3.15 : Flowchart penelitian

DAFTAR PUSTAKA

1. Budiharto, Widodo. Firmansyah, Sigit. 2004. *Elektronika Digital dan Mikroprosesor*. Yogyakarta : Penerbit ANDI.
2. Sasongko Hari Bagus. (2012). *Pemrograman Mikrokontroler dengan Bahasa C*. Yogyakarta :Penerbit ANDI.
3. Winarno. Arifianto, Deni.2011. *Bikin Robot Itu Gampang*. Jakarta : Kawan Pustaka.
4. <http://www.national.com/ds/LM/LM35.pdf> (diakses tanggal 5 Juni 2016, pukul 15:30)
5. Bayle,J.,2013,*C programming for Arduino*, Birmingham: Packt Publishing.
6. Anonim.*Instrument Engineer's Handbook For DURCO Quarter-turn Control Valves Second Editions*. Flowers Corporation: United States of America.
7. Groover, Mikell P. 1987. *Otomasi, Sistem Produksi, dan Computer-Integrated Manufacturing*. Second Edition, 1st Series. Translated by I Ketut Gunarta. 2006. Surabaya: Guna Widya.
8. Maftor. (2012). *Peralatan dan Mesin dalam Proses Produksi*. Materi Diklat . Gresik: Departemen Rancang Bangun.
9. Bryan L.A., *Programmable Controllers, Theory and Implementation*, Second Edition, 1997, Industrial Text Company.
10. Anonim.2002.*Valve Handbook*. Regin: United Kingdom
11. Anonim.2005.Emerson Proses Management: *Control Valve Handbook Fourth Editions*. Fisher: United States of America
12. Baskerville, N.2012.*Valve: Handbook for New Employees First Editions*. Valve Corporation: Washington U.S.A.

Lampiran 1



November 2000

LM35

Precision Centigrade Temperature Sensors

General Description

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in °Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of $\pm 1/4^\circ\text{C}$ at room temperature and $\pm 3/4^\circ\text{C}$ over a full -55 to $+150^\circ\text{C}$ temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only $60\ \mu\text{A}$ from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a -55° to $+150^\circ\text{C}$ temperature range, while the LM35C is rated for a -40° to $+110^\circ\text{C}$ range (-10° with improved accuracy). The LM35 series is available packaged in hermetic TO-46 transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and a plastic TO-220 package.

Features

- Calibrated directly in ° Celsius (Centigrade)
- Linear + 10.0 mV/°C scale factor
- 0.5°C accuracy guaranteeable (at $+25^\circ\text{C}$)
- Rated for full -55° to $+150^\circ\text{C}$ range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than $60\ \mu\text{A}$ current drain
- Low self-heating, 0.08°C in still air
- Nonlinearity only $\pm 1/4^\circ\text{C}$ typical
- Low impedance output, 0.1 W for 1 mA load

Typical Applications

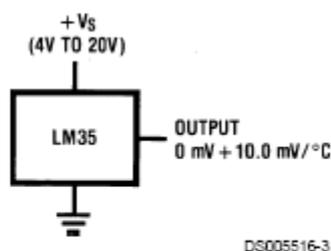
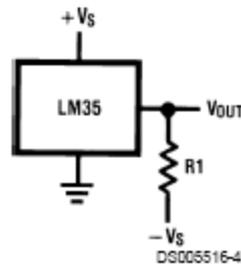


FIGURE 1. Basic Centigrade Temperature Sensor
($+2^\circ\text{C}$ to $+150^\circ\text{C}$)

LM35

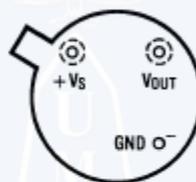


Choose $R_1 = -V_S/50 \mu A$
 $V_{OUT} = +1,500 \text{ mV at } +150^\circ C$
 $= +250 \text{ mV at } +25^\circ C$
 $= -550 \text{ mV at } -55^\circ C$

FIGURE 2. Full-Range Centigrade Temperature Sensor

Connection Diagrams

TO-46
Metal Can Package*



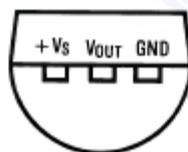
BOTTOM VIEW
DS005516-1

*Case is connected to negative pin (GND)

Order Number LM35H, LM35AH, LM35CH, LM35CAH or LM35DH

See NS Package Number H03H

TO-92
Plastic Package

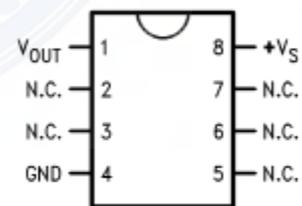


BOTTOM VIEW
DS005516-2

Order Number LM35CZ, LM35CAZ or LM35DZ

See NS Package Number Z03A

SO-8
Small Outline Molded Package



DS005516-21

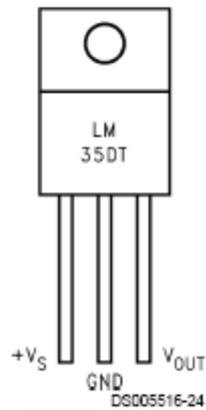
N.C. = No Connection

Top View

Order Number LM35DM

See NS Package Number M08A

**TO-220
Plastic Package***



*Tab is connected to the negative pin (GND).

Note: The LM35DT pinout is different than the discontinued LM35DP.

**Order Number LM35DT
See NS Package Number TA03F**

Absolute Maximum Ratings

(Note 10)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Supply Voltage	+35V to -0.2V
Output Voltage	+6V to -1.0V
Output Current	10 mA
Storage Temp.;	
TO-46 Package,	-60°C to +180°C
TO-92 Package,	-60°C to +150°C
SO-8 Package,	-65°C to +150°C
TO-220 Package,	-65°C to +150°C
Lead Temp.:	
TO-46 Package,	
(Soldering, 10 seconds)	300°C

Electrical Characteristics (Notes 1, 6)

Parameter	Conditions	LM35A			LM35CA			Units (Max.)
		Typical	Tested Limit (Note 4)	Design Limit (Note 5)	Typical	Tested Limit (Note 4)	Design Limit (Note 5)	
Accuracy (Note 7)	$T_A = +25^\circ\text{C}$	± 0.2	± 0.5		± 0.2	± 0.5		$^\circ\text{C}$
	$T_A = -10^\circ\text{C}$	± 0.3			± 0.3		± 1.0	$^\circ\text{C}$
	$T_A = T_{\text{MAX}}$	± 0.4	± 1.0		± 0.4	± 1.0		$^\circ\text{C}$
	$T_A = T_{\text{MIN}}$	± 0.4	± 1.0		± 0.4		± 1.5	$^\circ\text{C}$
Nonlinearity (Note 8)	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	± 0.18		± 0.35	± 0.15		± 0.3	$^\circ\text{C}$
Sensor Gain (Average Slope)	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	$+10.0$	$+9.9,$ $+10.1$		$+10.0$		$+9.9,$ $+10.1$	mV/ $^\circ\text{C}$
Load Regulation (Note 3) $0 \leq I_L \leq 1 \text{ mA}$	$T_A = +25^\circ\text{C}$	± 0.4	± 1.0		± 0.4	± 1.0		mV/mA
	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	± 0.5		± 3.0	± 0.5		± 3.0	mV/mA
Line Regulation (Note 3)	$T_A = +25^\circ\text{C}$	± 0.01	± 0.05		± 0.01	± 0.05		mV/V
	$4\text{V} \leq V_S \leq 30\text{V}$	± 0.02		± 0.1	± 0.02		± 0.1	mV/V
Quiescent Current (Note 9)	$V_S = +5\text{V}, +25^\circ\text{C}$	56	67		56	67		μA
	$V_S = +5\text{V}$	105		131	91		114	μA
	$V_S = +30\text{V}, +25^\circ\text{C}$	56.2	68		56.2	68		μA
	$V_S = +30\text{V}$	105.5		133	91.5		116	μA
Change of Quiescent Current (Note 3)	$4\text{V} \leq V_S \leq 30\text{V}, +25^\circ\text{C}$	0.2	1.0		0.2	1.0		μA
	$4\text{V} \leq V_S \leq 30\text{V}$	0.5		2.0	0.5		2.0	μA
Temperature Coefficient of Quiescent Current		$+0.39$		$+0.5$	$+0.39$		$+0.5$	$\mu\text{A}/^\circ\text{C}$
Minimum Temperature for Rated Accuracy	In circuit of Figure 1, $I_L = 0$	$+1.5$		$+2.0$	$+1.5$		$+2.0$	$^\circ\text{C}$
Long Term Stability	$T_J = T_{\text{MAX}}$, for 1000 hours	± 0.08			± 0.08			$^\circ\text{C}$

Electrical Characteristics

(Notes 1, 6)

Parameter	Conditions	LM35			LM35C, LM35D			Units (Max.)
		Typical	Tested Limit (Note 4)	Design Limit (Note 5)	Typical	Tested Limit (Note 4)	Design Limit (Note 5)	
Accuracy, LM35, LM35C (Note 7)	$T_A=+25^\circ\text{C}$	± 0.4	± 1.0		± 0.4	± 1.0		$^\circ\text{C}$
	$T_A=-10^\circ\text{C}$	± 0.5			± 0.5		± 1.5	$^\circ\text{C}$
	$T_A=T_{\text{MAX}}$	± 0.8	± 1.5		± 0.8		± 1.5	$^\circ\text{C}$
	$T_A=T_{\text{MIN}}$	± 0.8		± 1.5	± 0.8		± 2.0	$^\circ\text{C}$
Accuracy, LM35D (Note 7)	$T_A=+25^\circ\text{C}$				± 0.6	± 1.5		$^\circ\text{C}$
	$T_A=T_{\text{MAX}}$				± 0.9		± 2.0	$^\circ\text{C}$
	$T_A=T_{\text{MIN}}$				± 0.9		± 2.0	$^\circ\text{C}$
Nonlinearity (Note 8)	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	± 0.3		± 0.5	± 0.2		± 0.5	$^\circ\text{C}$
Sensor Gain (Average Slope)	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	+10.0	+9.8, +10.2		+10.0		+9.8, +10.2	mV/ $^\circ\text{C}$
Load Regulation (Note 3) $0 \leq I_L \leq 1 \text{ mA}$	$T_A=+25^\circ\text{C}$	± 0.4	± 2.0		± 0.4	± 2.0		mV/mA
	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	± 0.5		± 5.0	± 0.5		± 5.0	mV/mA
Line Regulation (Note 3)	$T_A=+25^\circ\text{C}$	± 0.01	± 0.1		± 0.01	± 0.1		mV/V
	$4\text{V} \leq V_S \leq 30\text{V}$	± 0.02		± 0.2	± 0.02		± 0.2	mV/V
Quiescent Current (Note 9)	$V_S=+5\text{V}, +25^\circ\text{C}$	56	80		56	80		μA
	$V_S=+5\text{V}$	105		158	91		138	μA
	$V_S=+30\text{V}, +25^\circ\text{C}$	56.2	82		56.2	82		μA
	$V_S=+30\text{V}$	105.5		161	91.5		141	μA
Change of Quiescent Current (Note 3)	$4\text{V} \leq V_S \leq 30\text{V}, +25^\circ\text{C}$	0.2	2.0		0.2	2.0		μA
	$4\text{V} \leq V_S \leq 30\text{V}$	0.5		3.0	0.5		3.0	μA
Temperature Coefficient of Quiescent Current		+0.39		+0.7	+0.39		+0.7	$\mu\text{A}/^\circ\text{C}$
Minimum Temperature for Rated Accuracy	In circuit of Figure 1, $I_L=0$	+1.5		+2.0	+1.5		+2.0	$^\circ\text{C}$
Long Term Stability	$T_J=T_{\text{MAX}}$, for 1000 hours	± 0.08			± 0.08			$^\circ\text{C}$

Note 1:

Unless otherwise noted, these specifications apply: $-55^\circ\text{C} \leq T_J \leq +150^\circ\text{C}$ for the LM35 and LM35A; $-40^\circ\text{C} \leq T_J \leq +110^\circ\text{C}$ for the LM35C and LM35CA; and $0^\circ\text{C} \leq T_J \leq +100^\circ\text{C}$ for the LM35D. $V_S=+5\text{Vdc}$ and $I_{\text{LOAD}}=50 \mu\text{A}$, in the circuit of Figure 2. These specifications also apply from $+2^\circ\text{C}$ to T_{MAX} in the circuit of Figure 1. Specifications in **boldface** apply over the full rated temperature range.

Note 2:

Thermal resistance of the TO-46 package is $400^\circ\text{C}/\text{W}$, junction to ambient, and $24^\circ\text{C}/\text{W}$ junction to case. Thermal resistance of the TO-92 package is $180^\circ\text{C}/\text{W}$ junction to ambient. Thermal resistance of the small outline molded package is $220^\circ\text{C}/\text{W}$ junction to ambient. Thermal resistance of the TO-220 package is $90^\circ\text{C}/\text{W}$ junction to ambient. For additional thermal resistance information see table in the Applications section.

Note 3:

Regulation is measured at constant junction temperature, using pulse testing with a low duty cycle. Changes in output due to heating effects can be computed by multiplying the internal dissipation by the thermal resistance.

Note 4:

Tested Limits are guaranteed and 100% tested in production.

Note 5:

Design Limits are guaranteed (but not 100% production tested) over the indicated temperature and supply voltage ranges. These limits are not used to calculate outgoing quality levels.

Note 6:

Specifications in **boldface** apply over the full rated temperature range.

LM35

Note 7:

Accuracy is defined as the error between the output voltage and $10\text{mV}/^{\circ}\text{C}$ times the device's case temperature, at specified conditions of voltage, current, and temperature (expressed in $^{\circ}\text{C}$).

Note 8:

Nonlinearity is defined as the deviation of the output-voltage-versus-temperature curve from the best-fit straight line, over the device's rated temperature range.

Note 9:

Quiescent current is defined in the circuit of Figure 1.

Note 10:

Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its rated operating conditions. See Note 1.

Note 11:

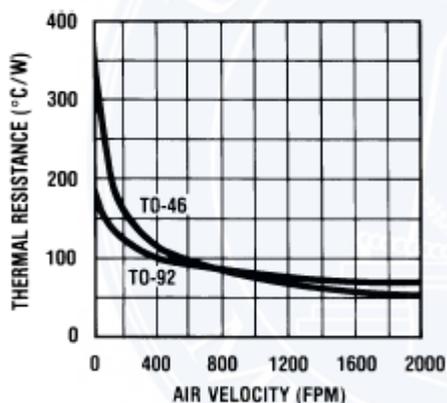
Human body model, 100 pF discharged through a 1.5 kW resistor.

Note 12:

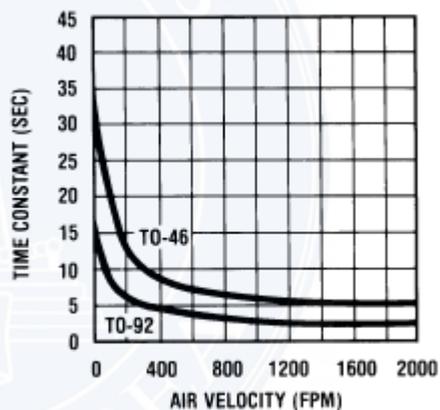
See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" or the section titled "Surface Mount" found in a current National Semiconductor Linear Data Book for other methods of soldering surface mount devices.

Typical Performance Characteristics

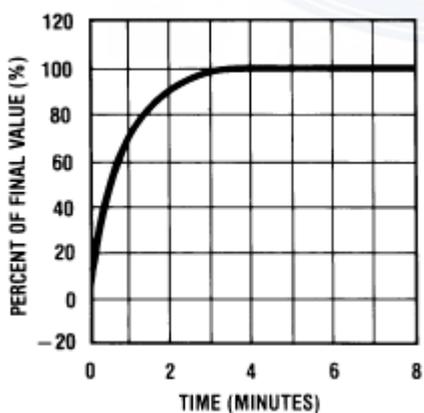
**Thermal Resistance
Junction to Air**



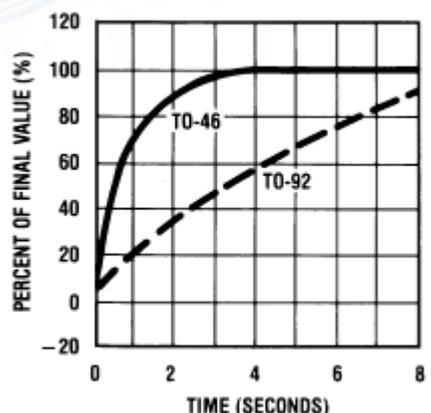
Thermal Time Constant



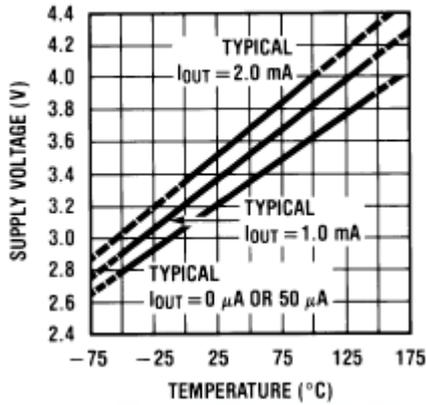
**Thermal Response
in Still Air**



**Thermal Response in
Stirred Oil Bath**

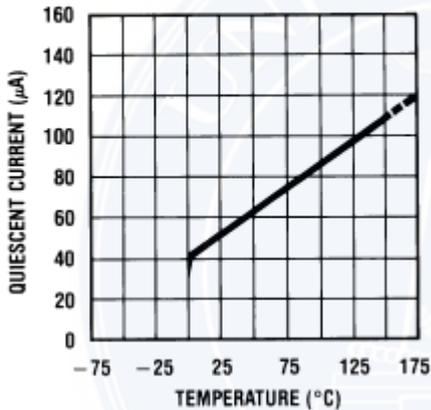


Minimum Supply Voltage vs. Temperature



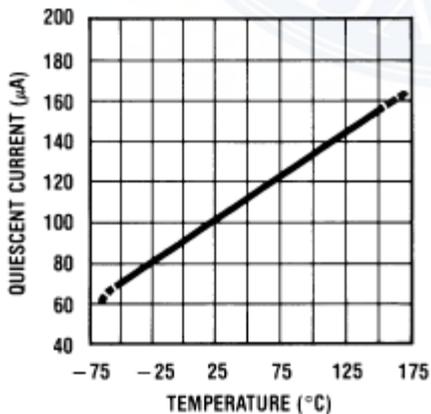
DS005516-29

Quiescent Current vs. Temperature (In Circuit of Figure 1.)



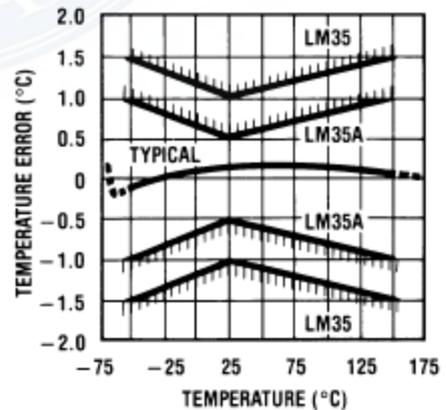
DS005516-30

Quiescent Current vs. Temperature (In Circuit of Figure 2.)



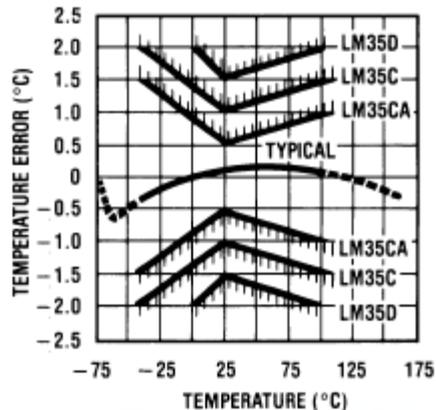
DS005516-31

Accuracy vs. Temperature (Guaranteed)



DS005516-32

Accuracy vs. Temperature (Guaranteed)



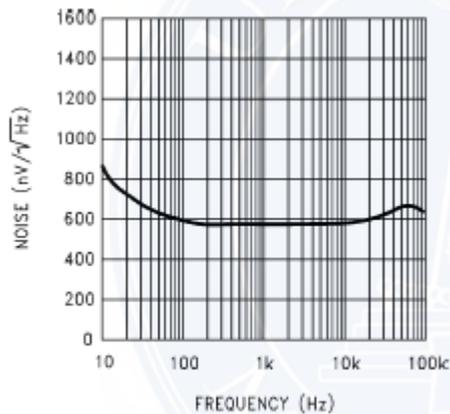
DS005516-33

LM35

Typical Performance Characteristics

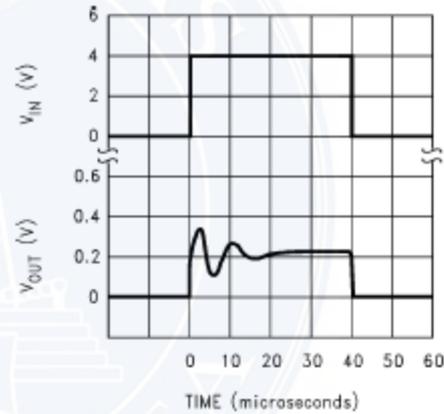
(Continued) www.national.com

Noise Voltage



DS005516-34

Start-Up Response



DS005516-35

Applications

The LM35 can be applied easily in the same way as other integrated-circuit temperature sensors. It can be glued or cemented to a surface and its temperature will be within about 0.01°C of the surface temperature.

This presumes that the ambient air temperature is almost the same as the surface temperature; if the air temperature were much higher or lower than the surface temperature, the actual temperature of the LM35 die would be at an intermediate temperature between the surface temperature and the air temperature. This is especially true for the TO-92 plastic package, where the copper leads are the principal thermal path to carry heat into the device, so its temperature might be closer to the air temperature than to the surface temperature.

To minimize this problem, be sure that the wiring to the LM35, as it leaves the device, is held at the same temperature as the surface of interest. The easiest way to do this is to cover up these wires with a bead of epoxy which will insure that the leads and wires are all at the same temperature as the surface, and that the LM35 die's temperature will not be affected by the air temperature.

The TO-46 metal package can also be soldered to a metal surface or pipe without damage. Of course, in that case the V- terminal of the circuit will be grounded to that metal. Alternatively, the LM35 can be mounted inside a sealed-end metal tube, and can then be dipped into a bath or screwed into a threaded hole in a tank. As with any IC, the LM35 and accompanying wiring and circuits must be kept insulated and dry, to avoid leakage and corrosion. This is especially true if the circuit may operate at cold temperatures where condensation can occur. Printed-circuit coatings and varnishes such as Humiseal and epoxy paints or dips are often used to insure that moisture cannot corrode the LM35 or its connections.

These devices are sometimes soldered to a small light-weight heat fin, to decrease the thermal time constant and speed up the response in slowly-moving air. On the other hand, a small thermal mass may be added to the sensor, to give the steadiest reading despite small deviations in the air temperature.

www.national.com

Temperature Rise of LM35 Due To Self-heating (Thermal Resistance, θ_{JA})

	TO-46, no heat sink	TO-46*, small heat fin	TO-92, no heat sink	TO-92**, small heat fin	SO-8 no heat sink	SO-8** small heat fin	TO-220 no heat sink
Still air	400°C/W	100°C/W	180°C/W	140°C/W	220°C/W	110°C/W	90°C/W
Moving air	100°C/W	40°C/W	90°C/W	70°C/W	105°C/W	90°C/W	28°C/W
Still oil	100°C/W	40°C/W	90°C/W	70°C/W			
Stirred oil	50°C/W	30°C/W	45°C/W	40°C/W			
(Clamped to metal, Infinite heat sink)		(24°C/W)				(55°C/W)	

*Wakefield type 201, or 1" disc of 0.020" sheet brass, soldered to case, or similar.

**TO-92 and SO-8 packages glued and leads soldered to 1" square of 1/16" printed circuit board with 2 oz. foil or similar.

Typical Applications

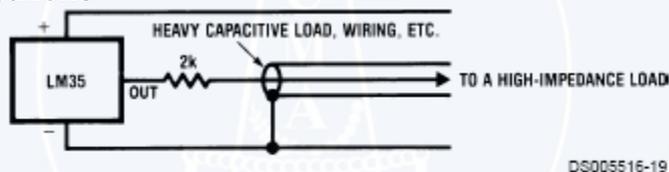


FIGURE 3. LM35 with Decoupling from Capacitive Load

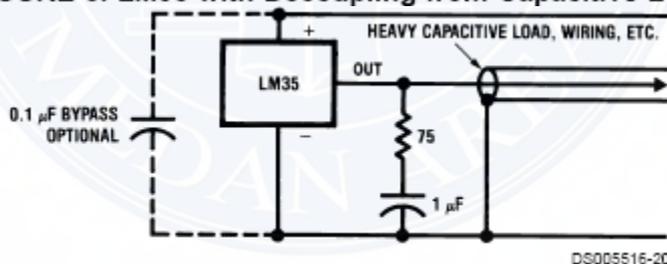


FIGURE 4. LM35 with R-C Damper

CAPACITIVE LOADS

Like most micropower circuits, the LM35 has a limited ability to drive heavy capacitive loads. The LM35 by itself is able to drive 50 pf without special precautions. If heavier loads are anticipated, it is easy to isolate or decouple the load with a resistor; see Figure 3. Or you can improve the tolerance of capacitance with a series R-C damper from output to ground; see Figure 4. When the LM35 is applied with a 200W load resistor as shown in Figure 5, Figure 6 or Figure 8 it is relatively immune to wiring capacitance because the capacitance forms a bypass from ground to input, not on the output. However, as with any linear circuit connected to wires in a hostile environment, its performance can be affected adversely by intense electromagnetic sources such as relays, radio transmitters, motors with arcing brushes, SCR transients, etc, as its wiring can act as a receiving antenna and its internal junctions can act as rectifiers. For best results in such cases, a bypass capacitor from VIN to ground and a series R-C damper such as 75W in series with 0.2 or 1 μ F from output to ground are often useful. These are shown in Figure 13, Figure 14, and Figure 16.

Lampiran 2

1. Tabel kalibrasi indikator suhu saat kenaikan suhu dalam bentuk MSB dan LSB

Suhu (°C)	LSB									MSB	
	1	2	3	4	5	6	7	8	9	10	
33	0	0	0	0	0	0	0	0	1	1	
37	0	0	0	0	0	0	0	0	0	0	
40	0	0	0	0	0	0	0	0	0	0	
43	0	0	0	0	0	0	0	0	0	0	
47	0	0	0	0	0	0	0	0	0	0	
50	0	0	0	0	0	0	0	0	0	0	
53	0	0	0	0	0	0	0	0	0	0	

2. Tabel kalibrasi indikator suhu saat penurunan suhu dalam bentuk msb dan lsb

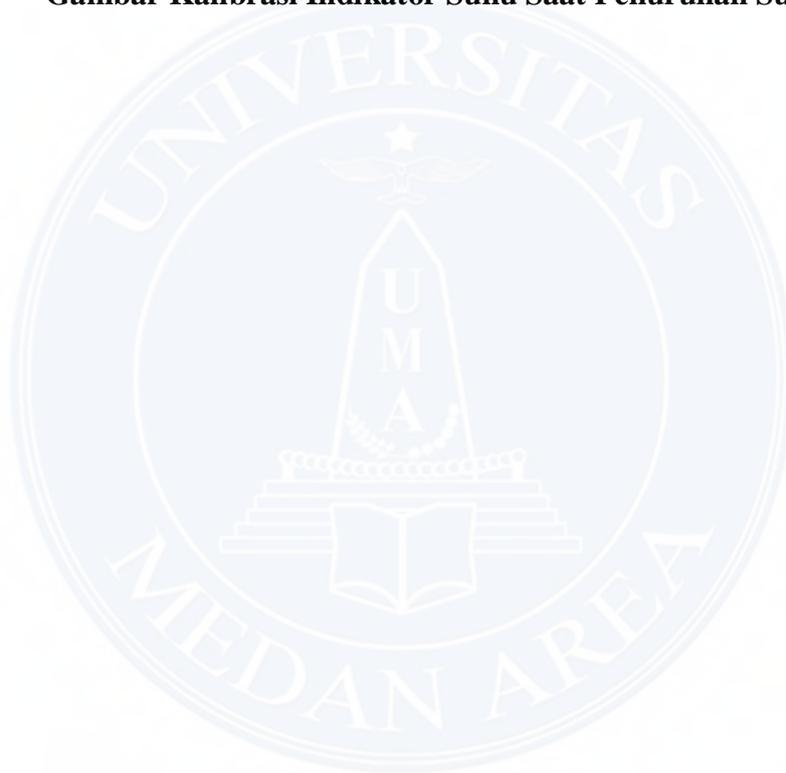
Suhu (°C)	LSB									MSB	
	1	2	3	4	5	6	7	8	9	10	
33	0	0	0	0	0	0	0	0	1	1	
28	0	0	0	0	0	0	1	1	1	1	
25	0	0	0	0	1	1	1	1	1	1	
20	0	0	1	1	1	1	1	1	1	1	
16	1	1	1	1	1	1	1	1	1	1	
13	1	1	1	1	1	1	1	1	1	1	
10	1	1	1	1	1	1	1	1	1	1	



Gambar Kalibrasi Indikator Suhu Saat Kenaikan Suhu



Gambar Kalibrasi Indikator Suhu Saat Penurunan Suhu



Lampiran 3

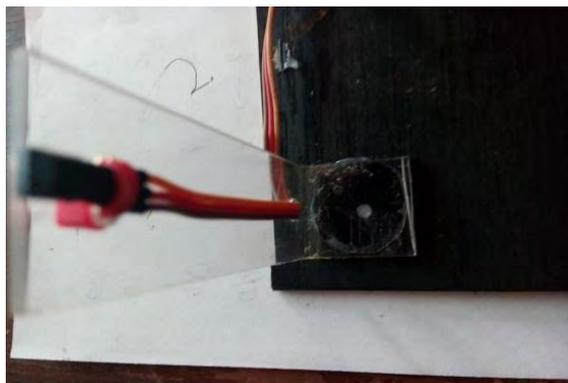
1. Arduino tampak atas



2. Tampilan LCD terhadap suhu terdeteksi

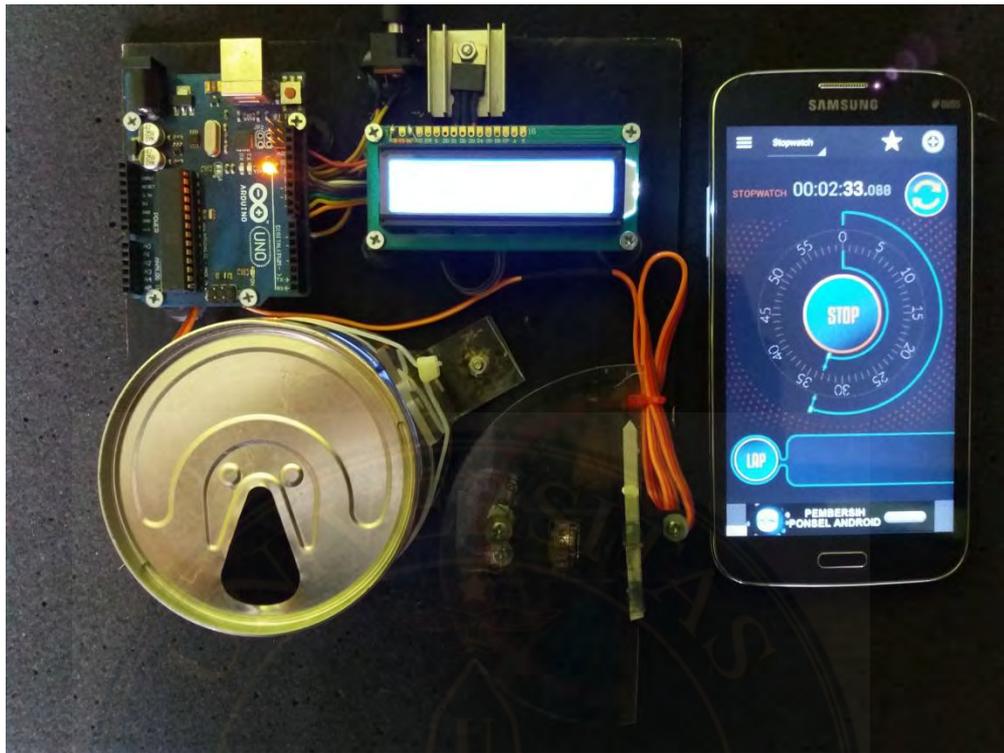


3. Sensor LM35 tampak atas



Lampiran 4

Gambar Hasil Pengujian Sistem Secara keseluruhan



Gambar Hasil Pengujian Suhu Terhadap Waktu

LAP 1: 00:00:01.765	LAP 13: 00:00:19.252
LAP 2: 00:00:02.814	LAP 14: 00:00:21.184
LAP 3: 00:00:04.196	LAP 15: 00:00:23.449
LAP 4: 00:00:05.385	LAP 16: 00:00:25.034
LAP 5: 00:00:06.303	LAP 17: 00:00:27.678
LAP 6: 00:00:07.909	LAP 18: 00:00:30.149
LAP 7: 00:00:09.276	LAP 19: 00:00:32.832
LAP 8: 00:00:10.534	LAP 20: 00:00:34.889
LAP 9: 00:00:11.950	LAP 21: 00:00:37.455
LAP 10: 00:00:14.199	LAP 22: 00:00:41.429
LAP 11: 00:00:15.767	LAP 23: 00:00:44.335
LAP 12: 00:00:17.980	LAP 24: 00:00:48.148

Lampiran 5

Gambar gear box

