

2017

Pengontrol Suhu Air Menggunakan Sensor DS18B20 Berbasis Arduino Uno

B, Alfian Akbar

<http://repositori.usu.ac.id/handle/123456789/3441>

Downloaded from Repositori Institusi USU, Universitas Sumatera Utara

**PENGONTROL SUHU AIR MENGGUNAKAN SENSOR
DS18B20 BERBASIS ARDUINO UNO**

TUGAS AKHIR

**ALFIAN AKBAR B
142408048**



**PROGRAM STUDI D-III FISIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SUMATERA UTARA
MEDAN
2017**

**PENGONTROL SUHU AIR MENGGUNAKAN SENSOR DS18B20
BERBASIS ARDUINO UNO**

TUGAS AKHIR

**Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh
Ahli Madya**

**ALFIAN AKBAR B
142408048**



**PROGRAM STUDI D-III FISIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SUMATERA UTARA
MEDAN
2017**

LEMBAR PERSETUJUAN

Judul :Pengontrol Suhu Air Menggunakan Sensor
DS18B20 Berbasis Arduino Uno
Kategori : Tugas Akhir
Nama : Alfian Akbar B
Nomor Induk Mahasiswa :142408048
Program Studi : Diploma (D-III) Fisika
Departemen : Fisika
Fakultas : Matematika Dan Ilmu Pengetahuan Alam
Universitas Sumatra Utara

Disetujui di
Medan, Juni 2017

Disetujui Oleh

Departemen Matematika FMIPA USU

Ketua Program Studi

Pembimbing,

Drs. Takdir Tamba, M.Eng.Sc

NIP. 196006031980032001

Dr.Kerista Tarigan.M.Eng.Sc.

NIP: 196002031980032001

PERNYATAAN

PENGONTROL SUHU AIR MENGGUNAKAN SENSOR DS18B20 BERBASIS ARDUINO UNO

TUGAS AKHIR

Saya mengakui bahwa tugas akhir ini adalah hasil karya sendiri. Kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, Juni 2017

Alfian Akbar B
142408048

ABSTRAK

Telah dilakukan perancangan dan realisasi sistem. Alat ini dapat dimanfaatkan sebagai sistem pengontrol suhu air otomatis. Sistem ini terdiri atas perangkat keras dan perangkat lunak. Perangkat keras terdiri atas sebuah mikrokontroler, sensor ds18b20, LCD (*Liquid Cristal Display*), ADC(*Analog to Digital Converter*), *buzzer*, rangkaian *keypad*. Perangkat lunak pada sistem ini dibuat dengan menggunakan program CodeVisionAVR. Program aplikasi ini bekerja pada saat dinyalakan maka inisialisasi hardware dilakukan kemudian menampilkan temperatur yang terdeteksi oleh ds18b20 pada LCD. Memasukkan *Setpoint* melalui *keypad* dan proses eksekusi dilakukan oleh mikrokontroler untuk menentukan suhu berapa dalam air . Setelah setpoint dimasukkan maka nilai setpoint disesuaikan dengan temperature, jika setpoint sudah mencapai suhu yang kita inginkan maka buzzer akan hidup. Suhu yang ditentukan akan bertahan sesuai range yang kita akan naik dan turun sesuai dengan set yang kita atur dengan range. Temperatur yang bisa dikendalikan oleh alat pengontrol suhu air bisa mencapai $\pm 100^{\circ}$ Celcius.

Kata Kunci : Mikrokontroler ATmega8, Sensor ds18b20, Buzzer, Water Heater.

KATA PENGANTAR

Dengan memanjatkan puji syukur kehadiran Allah SWT atas segala limpah rahmat, taufiq, dan hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan baik dan lancar. Penulis menyampaikan ucapan terima kasih kepada semua pihak yang telah membantu dalam penyusunan tugas akhir ini.

Ucapan terima kasih terutama penulis sampaikan kepada yang terhormat Ayah dan Ibu tercinta yang telah memberikan bantuan berupa dukungan moril dan material yang sangat membantu dalam menyelesaikan Tugas Akhir, Bapak Dr.Kerista Sebayang, M.S selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sumatra Utara, Bapak Drs.Takdir Tamba, M.Eng.Sc. selaku Ketua Program Studi D-III Fisika Fakultas MIPA Universitas Sumatra Utara, Bapak Dr.Kerista Tarigan.M.Eng.Sc. selaku Pembimbing yang telah membimbing dan mengarahkan kepada Penulis dalam menyelesaikan Tugas Akhir, Seluruh Staf Pengajar/Pegawai Program Studi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sumatra Utara, Ulfah Hidayah Al-madany yang selalu memberikan motivasi dan dukungan dalam menyelesaikan Tugas Akhir ini dan Rekan Fisika Instrumentasi D-III yang memberikan bantuan penulisan untuk menyelesaikan Tugas Akhir.

Penulis menyadari bahwa penyusunan Tugas Akhir ini masih terdapat banyak kekurangan dan kelemahan.Untuk itu penulis mengharapkan kritik dan saran dari semua pihak guna penyempurnaan Tugas Akhir dimasa yang akan datang.

Akhir kata, semoga Tugas Akhir ini dapat bermanfaat bagi rekan-rekan Mahasiswa dan pembaca sekalian demi menambah pengetahuan tentang Tugas Akhir.

Medan, Juni 2017

Alfian Akbar B

DAFTAR ISI

ABSTRAK	i
KATA PENGANTAR	ii
DAFTAR ISI	iv
DAFTAR GAMBAR	vii
DAFTAR TABEL	viii
 BAB I PENDAHULUAN	 1
1.1. Latar Belakang	1
1.2. Tujuan Penelitian	1
1.3. Rumusan Masalah	1
1.4. Batasan Masalah	2
1.5. Manfaat	2
1.6. Sistematika Penulisan	2
 BAB II LANDASAN TEORI	 4
2.1. Mikrokontroler Arduino	4
2.1.1. Power Supply	5
2.1.2. Memori	6
2.1.3. Input dan Output	6
2.1.4. Komunikasi	7
2.1.5. Software Arduino	7
2.1.6. Bahasa Pemrograman Arduino	7
2.1.7. Reset Otomatis (Software)	9
2.1.8. Proteksi Arus Lebih USB	10
2.1.9. Karakteristik Fisik	11
2.1.10. Konfigurasi Pin Arduino Uno R3	11
2.2. Pemanas Air Listrik (Water Heater)	12
2.3. Sensor Suhu DS18B20	12
2.3.1. Sensor Fisika	12
2.3.2. Sensor Kimia	13
2.4. Buzzer	14
2.5. Liquid Crystal Display (LCD)	15
2.6. Relay	16
 BAB III PERANCANGAN DAN PEMBUATAN	 18
3.1. Diagram Blok System	18
3.1.1. Fungsi-Fungsi Diagram Blok	18
3.1.2. Teknik Pengumpulan Data	19

3.1.3. Instrumen	19
3.1.4. Teknik Analisis data	19
3.2. Perancangan Pembuatan Sistem Minimum Mikrokontroler	19
3.3. Flowchart Software	22
 BAB IV PENGUJIAN DAN PEMBAHASAN SISTEM	 24
4.1. Pengujian LCD	24
4.2. Pengujian Kerja Alat Pengontrol Temperatur Air	29
4.3. Grafik Set Suhu Pada Malam Hari	30
4.4. Grafik Set Suhu Pada Siang Hari	31
4.5. Grafik Set Suhu Pada Pagi Hari	32
4.6. Pembahasan Hasil Pengujian Alat Pengontrol Temperatur Air	33
 BAB V PENUTUP	 34
5.1. Kesimpulan	34
5.2. Saran	34
 DAFTAR PUSTAKA	 35
DAFTAR LAMPIRAN	36

DAFTAR GAMBAR

2.1. Arduino	11
2.2. Konfigurasi pin Arduino	11
2.3. Elemen pemanas pada heater	12
2.4. Sensor Suhu DS18B20	13
2.5. Buzzer	14
2.6. Relay	16
3.1. Diagram blok	18
3.2. Skema rangkain sistem mikrokontroler	20
3.3. <i>Flowchart software</i>	22
4.1. Grafik Set Suhu Pada Malam Hari	30
4.2. Grafik Set Suhu Pada Siang Hari	31
4.3. Grafik Set Suhu Pada Pagi Hari	32

DAFTAR TABEL

2.1. Deskripsi Arduino	5
2.2. Koneksi Antara Modul LCD dengan Mikrokontroler	15
4.1. Hasil pengujian kerja alat	29
4.2. Perubahan suhu pada malam hari	30
4.3. Perubahan suhu pada siang hari	31
4.4. Perubahan suhu pada pagi hari	32

DAFTAR LAMPIRAN

1. Program Lengkap	36
2. Rangkaian Lengkap	41
3. Data shett	42

BAB I PENDAHULUAN

1.1 Latar Belakang

Teknologi sekarang sangat memegang peranan penting, teknologi yang modern harus mencakup secara sinergi antara efisiensi biaya, sumber daya alam serta sumber daya manusianya. Jika salah satu diabaikan akan timbul masalah dikemudian hari.

Pengaturan pengendalian secara otomatis diberbagai bidang pada saat ini sering dikembangkan diantaranya adalah aplikasi pengendalian temperatur yang banyak ditemui. Tujuan dari tugas Akhir ini adalah sistem pengontrolan suhu air yang bisa diset dan ditampilkan. Sistem yang dibuat ini memanfaatkan kemampuan Arduino uno dalam akuisisi data dan mengambil keputusan. Kawasan temperatur yang bisa dikendalikan adalah keadaan temperatur lingkungan sampai dengan 90 - 100° Celcius.

Keuntungan dari sistem ini adalah komponen rangkaian yang banyak dipasaran yang harganya cukup terjangkau sehingga dalam penggunaannya efisiensi biaya dapat dicapai, mudah dalam perawatan, temperatur dapat dipantau langsung pada layar LCD, penyetingan temperatur sesuai keinginan dengan memasukkan *setpoint* pada *keypad*, kemudahan dalam pengoperasian. Sistem pengontrolan temperatur pada alat yang dirancang dapat mencapai 90 - 100° Celcius.

1.2 Rumusan Masalah

Berdasarkan permasalahan yang terdapat dalam latar belakang maka pada penelitian yang ditujukan untuk bagaimana merancang, mengaplikasikan dan mengontrol sistem mikrokontroler dalam pengendalian suhu air.

1.3. Tujuan Penelitian

Adapun tujuan dari dibuatnya alat ini adalah :

1. Mengaplikasikan system pemograman arduino untuk pengontrolan temperature air.
2. Mengontrol range (batasan suhu) pada relay agar suhu dipertahankan suhu.

3. Mengembangkan ilmu pengetahuan dan teknologi, khususnya dibidang system kendali alat dan mengimplementasikan ilmu yang didapat selama kuliah agar lebih bermanfaat.

1.4 Batasan Masalah

Batasan masalah yang digunakan dalam tugas akhir ini adalah :

1. Sensor temperatur yang digunakan yaitu menggunakan sensor ds18b20.
2. Rangkaian Mikrokontroller yang di gunakan adalah mikrokontroller ATmega8.
3. Ketika suhu air sudah mencapai suhu yang kita inginkan maka sensor akan mempertahankan suhu dengan range yang ditentukan dan buzzer akan aktif.
4. Program yang digunakan adalah Codevision AVR.

1.5 Metode Penulisan

Adapun metode penulisan yang digunakan dalam menyusun dan menganalisa proyek akhir ini adalah:

1. Studi literatur yang berhubungan dengan perancangan dan pembuatan alat ini.
2. Perencanaan dan pembuatan alat Merencanakan peralatan yang telah dirancang baik software maupun hardware.
3. Pengujian alat, Peralatan yang telah dibuat kemudian diuji apakah telah sesuai dengan yang telah direncanakan.

1.6 Sistematika Penulisan

Untuk mempermudah penyusunan laporan, maka dalam hal ini penulis membagi dalam beberapa bab, serta memberikan gambaran secara garis besar isi dari tiap-tiap bab.

BAB I : PENDAHULUAN

Bab ini berisikan latar belakang masalah, tujuan dan manfaat penelitian, identifikasi masalah, pembatasan masalah, rumusan masalah, metode penelitian, serta sistematika penulisan.

BAB II : LANDASAN TEORI

Bab ini merupakan landasan teori yang membahas tentang teori-teori yang mendukung dalam penyelesaian masalah.

BAB III : METODOLOGI PENELITIAN

Meliputi metode, bahan alat, perancangan dan pengambilan data penelitian.

BAB IV : HASIL dan ANALISA

Meliputi hasil penelitian dan pembahasan.

BAB V : KESIMPULAN dan SARAN

Berisikan kesimpulan tentang hasil rancangan yang telah dibuat serta saran dalam pengembangan rancangan tersebut.

BAB II TINJAUAN PUSTAKA

2.1. Mikrokontroler Arduino

Arduino adalah sebuah board mikrokontroller yang berbasis ATmega328. Arduino memiliki 14 pin input/output yang mana 6 pin dapat digunakan sebagai output PWM, 6 analog input, crystal osilator 16 MHz, koneksi USB, jack power, kepala ICSP, dan tombol reset. Arduino mampu *men-support* mikrokontroller; dapat dikoneksikan dengan komputer menggunakan kabel USB.

Arduino memiliki kelebihan tersendiri disbanding board mikrokontroler yang lain selain bersifat open source, arduino juga mempunyai bahasa pemrogramannya sendiri yang berupa bahasa C. Selain itu dalam *board* arduino sendiri sudah terdapat *loader* yang berupa USB sehingga memudahkan kita ketika kita memprogram mikrokontroler didalam arduino. Sedangkan pada kebanyakan *board* mikrokontroler yang lain yang masih membutuhkan rangkaian *loader* terpisah untuk memasukkan program ketika kita memprogram mikrokontroler. Port USB tersebut selain untuk *loader* ketika memprogram, bisa juga difungsikan sebagai port komunikasi serial.

Arduino menyediakan 20 pin I/O, yang terdiri dari 6 pin input analog dan 14 pin digital input/output. Untuk 6 pin analog sendiri bisa juga difungsikan sebagai output digital jika diperlukan output digital tambahan selain 14 pin yang sudah tersedia. Untuk mengubah pin analog menjadi digital cukup mengubah konfigurasi pin pada program.

Dalam *board* kita bisa lihat pin digital diberi keterangan 0-13, jadi untuk menggunakan pin analog menjadi output digital, pin analog yang pada keterangan board 0-5 kita ubah menjadi pin 14-19. dengan kata lain pin analog 0-5 berfungsi juga sebagai pin output digital 14-16. Sifat *open source* arduino juga banyak memberikan keuntungan tersendiri untuk kita dalam menggunakan *board* ini, karena dengan sifat open source komponen yang kita pakai tidak hanya tergantung pada satu merek, namun memungkinkan kita bisa memakai semua komponen yang ada dipasaran.

Bahasa pemrograman arduino merupakan bahasa C yang sudah disederhanakan dengan bahasa pemrogramannya sehingga mempermudah kita dalam mempelajari dan mendalami mikrokontroller.

Deskripsi Arduino UNO:

Tabel 2.1. Deskripsi Arduino

Mikrokontroller	ATmega 328
Tegangan Pengoperasian	5 V
Tegangan Input yang disarankan	7 – 12 V
Batas Tegangan Input	6 – 20 V
Jumlah pin I/O digital	14 pin digital (6 diantaranya menyediakan keluaran PWM)
Jumlah pin input Analog	6 pin
Arus DC tiap pin I/O	40mA
Arus DC untuk pin 3,3 V	50mA
Memori Flash	32 KB (ATmega 328) sekitar 0,5 KB digunakan oleh bootloader
SRAM	2 KB (ATmega 328)
EPROM	1 KB (ATmega 328)
Clock Speed	16 MHz

2.1.1. Power Supply

Arduino dapat diberikan *power* melalui koneksi USB atau *power supply*. *Power*nya diselek secara otomatis. *Power supply* dapat menggunakan adaptor DC atau baterai. Adaptor dapat dikoneksikan dengan mencolok *jack* adaptor pada koneksi port input *supply*. *Board* arduino dapat dioperasikan menggunakan *supply* dari luar sebesar 6 - 20 volt. Jika *supply* kurang dari 7V, kadangkala pin 5V akan menyuplai kurang dari 5 volt dan *board* bisa menjadi tidak stabil. Jika menggunakan lebih dari 12 V, tegangan di regulator bisa menjadi sangat panas dan menyebabkan kerusakan pada *board*. Rekomendasi tegangan ada pada 7 sampai 12 volt.

Penjelasan pada pin power adalah sebagai berikut :

a. Vin

Tegangan input ke board arduino ketika menggunakan tegangan dari luar (seperti yang disebutkan 5 volt dari koneksi USB atau tegangan yang diregulasikan). Pengguna dapat memberikan tegangan melalui pin ini, atau jika tegangan suplai menggunakan *power jack*, aksesnya menggunakan pin ini.

b. 5V

Regulasi *power supply* digunakan untuk power mikrokontroller dan komponen lainnya pada board. 5V dapat melalui Vin menggunakan regulator pada board, atau *supply* oleh USB atau *supply* regulasi 5V lainnya.

c.3V3

Suplai 3.3 volt didapat oleh FTDI chip yang ada di board. Arus maximumnya adalah 50mA

d.Pin Ground

berfungsi sebagai jalur ground pada arduino.

2.1.2. Memori

ATmega328 memiliki 32 KB flash memori untuk menyimpan kode, juga 2 KB yang digunakan untuk bootloader. ATmega328 memiliki 2 KB untuk SRAM dan 1 KB untuk EEPROM.

2.1.3. Input & Output

Setiap 14 pin digital pada arduino dapat digunakan sebagai input atau output, menggunakan fungsi `pinMode()`, `digitalWrite()`, dan `digitalRead()`. Input/output dioperasikan pada 5 volt. Setiap pin dapat menghasilkan atau menerima maximum 40 mA dan memiliki internal pull-up resistor (disconnected oleh default) 20-50K Ohm.

Beberapa pin memiliki fungsi sebagai berikut :

- a) Serial : 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan mengirim (TX) TTL data serial. Pin ini terhubung pada pin yang koresponding dari USB ke TTL chip serial.
- b) Interrupt eksternal : 2 dan 3. Pin ini dapat dikonfigurasi untuk trigger sebuah interap pada low value, rising atau falling edge, atau perubahan nilai.
- c) PWM : 3, 5, 6, 9, 10, dan 11. Mendukung 8-bit output PWM dengan fungsi `analogWrite()`.
- d) SPI : 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Pin ini mensupport komunikasi SPI, yang mana masih mendukung hardware, yang tidak termasuk pada bahasa arduino.
- e) LED : 13. Ini adalah dibuat untuk koneksi LED ke digital pin 13. Ketika pin bernilai HIGH, LED hidup, ketika pin LOW, LED mati.

2.1.4. Komunikasi

Uno Arduino memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lain. ATmega328 ini menyediakan UART TTL (5V) komunikasi serial, yang tersedia pada pin digital 0 (RX) dan 1 (TX). *Firmware* Arduino menggunakan USB *driver* standar COM, dan tidak ada *driver* eksternal yang dibutuhkan. Namun, pada Windows, file. Ini diperlukan. Perangkat lunak Arduino termasuk monitor serial yang memungkinkan data sederhana yang akan dikirim ke *board* Arduino. RX dan TX LED di *board* akan berkedip ketika data sedang dikirim melalui chip USB-to-serial dan koneksi USB ke komputer.

2.1.5. Software Arduino

Arduino Uno dapat diprogram dengan perangkat lunak Arduino . Pada ATmega328 di Arduino terdapat *bootloader* yang memungkinkan Anda untuk meng-*upload* kode baru untuk itu tanpa menggunakan *programmer hardware eksternal*.

IDE Arduino adalah *software* yang sangat canggih ditulis dengan menggunakan Java. IDE Arduino terdiri dari:

1. Editor program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *Processing*.
2. *Compiler*, sebuah modul yang mengubah kode program (bahasa *Processing*) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *Processing*. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya *compiler* diperlukan dalam hal ini.
3. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam memory didalam papan Arduino.

Sebuah kode program Arduino umumnya disebut dengan istilah *sketch*. Kata "*sketch*" digunakan secara bergantian dengan "kode program" dimana keduanya memiliki arti yang sama.

2.1.6. Bahasa Pemrograman Arduino

Seperti yang telah dijelaskan diatas program Arduino sendiri menggunakan bahasa C. walaupun banyak sekali terdapat bahasa pemrograman tingkat tinggi (*high*

level language) seperti pascal, basic, cobol, dan lainnya. Walaupun demikian, sebagian besar dari paraprogramer profesional masih tetap memilih bahasa C sebagai bahasa yang lebih unggul, berikut alasan-alasannya:

Bahasa C merupakan bahasa yang *powerful* dan *fleksibel* yang telah terbukti dapat menyelesaikan program-program besar seperti pembuatan sistem operasi, pengolah gambar (seperti pembuatan game) dan juga pembuatan kompilator bahasa pemrograman baru.

Bahasa C merupakan bahasa yang *portabel* sehingga dapat dijalankan di beberapa sistem operasi yang berbeda. Sebagai contoh program yang kita tulis dalam sistem operasi windows dapat kita kompilasi didalam sistem operasi linux dengan sedikit ataupun tanpa perubahan sama sekali.

Bahasa C merupakan bahasa yang sangat populer dan banyak digunakan oleh programer berpengalaman sehingga kemungkinan besar *library* pemrograman telah banyak disediakan oleh pihak luar/lain dan dapat diperoleh dengan mudah. Bahasa C merupakan bahasa yang bersifat modular, yaitu tersusun atas rutin-rutin tertentu yang dinamakan dengan fungsi (*function*) dan fungsi-fungsi tersebut dapat digunakan kembali untuk pembuatan program-program lainnya tanpa harus menulis ulang implementasinya.

Bahasa C merupakan bahasa tingkat menengah (*middle level language*) sehingga mudah untuk melakukan interface (pembuatan program antar muka) ke perangkat keras. Struktur penulisan program dalam bahasa C harus memiliki fungsi utama, yang bernama `main()`. Fungsi inilah yang akan dipanggil pertama kali pada saat proses eksekusi program. Artinya apabila kita mempunyai fungsi lain selain fungsi utama, maka fungsi lain tersebut baru akan dipanggil pada saat digunakan.

Oleh karena itu bahasa C merupakan bahasa prosedural yang menerapkan konsep runtutan (program dieksekusi per baris dari atas ke bawah secara berurutan), maka apabila kita menuliskan fungsi-fungsi lain tersebut dibawah fungsi utama, maka kita harus menuliskan bagian prototipe (*prototype*), hal ini dimaksudkan untuk mengenalkan terlebih dahulu kepada kompilator daftar fungsi yang akan digunakan di dalam program. Namun apabila kita menuliskan fungsi-fungsi lain tersebut diatas atau sebelum fungsi utama, maka kita tidak perlu lagi untuk menuliskan bagian prototipe diatas.

Selain itu juga dalam bahasa C kita akan mengenal *file header*, biasa ditulis dengan ekstensi *h(*.h)*, adalah file bantuan yang digunakan untuk menyimpan daftar-daftar fungsi yang akan digunakan dalam program. Bagi anda yang sebelumnya pernah mempelajari bahasa pascal, *file header* ini serupa dengan unit. Dalam bahasa C, file header standar yang untuk proses *input/output* adalah *<stdio.h>*.

Perlu sekali untuk diperhatikan bahwa apabila kita menggunakan *file header* yang telah disediakan oleh kompilator, maka kita harus menuliskannya didalam tanda ‘<’ dan ‘>’ (misalnya *<stdio.h>*). Namun apabila menggunakan *file header* yang kita buat sendiri, maka file tersebut ditulis diantara tanda “ dan ” (misalnya “*cobaheader.h*”). perbedaan antara keduanya terletak pada saat pencerian file tersebut. Apabila kita menggunakan tanda <>, maka file tersebut dianggap berada pada direktori default yang telah ditentukan oleh kompilator. Sedangkan apabila kita menggunakan tanda “”, maka *file header* dapat kita dapat tentukan sendiri lokasinya.

File header yang akan kita gunakan harus kita daftarkan dengan menggunakan directive *#include*. Directive *#include* ini berfungsi untuk memberi tahu kepada kompilator bahwa program yang kita buat akan menggunakan file-file yang didaftarkan. Berikut ini contoh penggunaan directive *#include*.

```
#include<stdio.h>
#include<stdlib.h>
#include"myheader.h"
```

Setiap kita akan menggunakan fungsi tertentu yang disimpan dalam sebuah *file header*, maka kita juga harus mendaftarkan *file header*nya dengan menggunakan directive *#include*. Sebagai contoh, kita akan menggunakan fungsi *getch()* dalam program, maka kita harus mendaftarkan *file header* *<conio.h>*.

2.1.7. Reset Otomatis (Software)

Dari pada mengharuskan sebuah penekanan fisik dari tombol reset sebelum sebuah penguploadan, Arduino Uno didesain pada sebuah cara yang memungkinkannya untuk direset dengan software yang sedang berjalan pada komputer yang sedang terhubung. Salah satu garis kontrol aliran hardware (DTR)

dari ATmega8U2/16U2 dihubungkan ke garis reset dari ATmega328 melalui sebuah kapasitor 100 nanofarad. Ketika saluran ini dipaksakan (diambil rendah), garis reset jatuh cukup panjang untuk mereset chip. Software Arduino menggunakan kemampuan ini untuk memungkinkan kita untuk mengupload kode dengan mudah menekan tombol upload di software Arduino. Ini berarti bahwa bootloader dapat mempunyai sebuah batas waktu yang lebih singkat, sebagai penurunan dari DTR yang dapat menjadi koordinasi yang baik.

Pengaturan ini mempunyai implikasi. Ketika Arduino Uno dihubungkan ke sebuah komputer lain yang sedang running menggunakan OS Mac X atau Linux, Arduino Uno mereset setiap kali sebuah koneksi dibuat dari software (melalui USB). Untuk berikutnya, setengah-detik atau lebih, bootloader sedang berjalan pada Arduino UNO. Ketika Arduino UNO diprogram untuk mengabaikan data yang cacat/salah (contohnya apa saja selain sebuah penguploadan kode baru) untuk menahan beberapa bit pertama dari data yang dikirim ke board setelah sebuah koneksi dibuka.

Jika sebuah sketch sedang berjalan pada board menerima satu kali konfigurasi atau data lain ketika sketch pertama mulai, memastikan bahwa software yang berkomunikasi menunggu satu detik setelah membuka koneksi dan sebelum mengirim data ini. Arduino Uno berisikan sebuah jejak yang dapat dihapus untuk mencegah reset otomatis. Pad pada salah satu sisi dari jejak dapat disolder bersama untuk mengaktifkan kembali. Pad itu diberi label “RESET-RN” Kita juga dapat menonaktifkan reset otomatis dengan menghubungkan sebuah resistor 110 ohm dari tegangan 5V ke garis reset.

2.1.8. Proteksi Arus Lebih USB

Arduino UNO mempunyai sebuah sekering reset yang memproteksi port USB komputer dari hubungan pendek dan arus lebih. Walaupun sebagian besar komputer menyediakan proteksi internal sendiri, sekering menyediakan sebuah proteksi tambahan. Jika lebih dari 500 mA diterima port USB, sekering secara otomatis akan memutuskan koneksi sampai hubungan pendek atau kelebihan beban hilang.

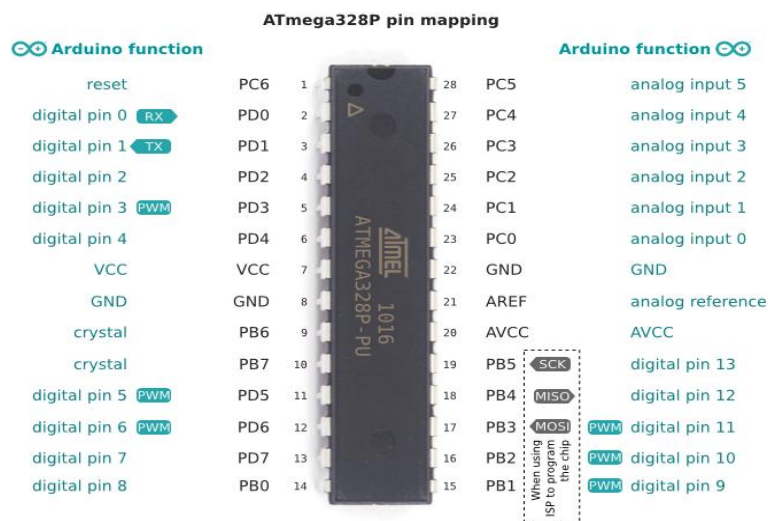
2.1.9. Karakteristik Fisik

Panjang dan lebar maksimum dari PCB Arduino UNO masing-masingnya adalah 2.7 dan 2.1 inci, dengan konektor USB dan power jack yang memperluas dimensinya. Empat lubang sekrup memungkinkan board untuk dipasangkan ke sebuah permukaan atau kotak. Sebagai catatan, bahwa jarak antara pin digital 7 dan 8 adalah 160 mil. (0.16"), bukan sebuah kelipatan genap dari jarak 100 mil dari pin lainnya.



Gambar 2.1. Arduino Uno R3

2.1.10. Konfigurasi Pin arduino Uno R3



Gambar 2.2. Konfigurasi Pin Arduino (ATmega 8)

2.2. Pemanas Air Listrik (*Water Heater*)

Pemanas air listrik memiliki elemen pemanas untuk dapat mengubah energi listrik menjadi kalor. Elemen pemanas terbuat dari bahan *stainless* dan tidak karatan dan didalamnya terdapat belitan kawat yang apabila diberi tegangan akan menghasilkan panas secara terus menerus. Elemen pemanas yang digunakan dalam pembuatan alat ini menggunakan elemen pemanas jenis mini *heater* yang dikendalikan dengan rangkaian penstabil suhu. Adapun *heater* yang digunakan dalam pembuatan alat ini memakai sebuah sumber AC 220 Volt. Sistem kerja *heater* memiliki kesamaan dengan mesin boiler sebagai penghasil panas. Namun hasil panas dari heater dimanfaatkan untuk memasak atau memanaskan air sedangkan mesin boiler untuk menghasilkan uap air yang biasanya dimanfaatkan diindustri besar.



Gambar 2.3. Elemen pemanas pada heater

2.3. Sensor suhu DS18B20

Pengertian sensor secara umum adalah alat yang digunakan untuk mendeteksi dan mengukur magnitudo sesuatu. Dapat didefinisikan sensor merupakan jenis transduser yang digunakan untuk mengubah variasi mekanis, magnetis, panas, cahaya dan kimia menjadi tegangan dan arus listrik.

Sensor dapat dibedakan menjadi 2 jenis yaitu:

2.3.1. Sensor Fisika

Sensor fisika mendeteksi suatu besaran berdasarkan hukum-hukum fisika. Contoh sensor fisika adalah sensor cahaya, sensor suara, sensor kecepatan, dan sensor suhu.

2.3.2.Sensor Kimia

Sensor kimia mendeteksi jumlah suatu zat kimia dengan cara mengubah besaran kimia menjadi besaran listrik. Biasanya melibatkan beberapa reaksi kimia. Contoh sensor kimia adalah sensor pH dan sensor gas.

Sensor suhu DS18B20 berfungsi untuk merubah besaran panas yang di tangkap menjadi besaran tegangan. Jenis sensor suhu yang digunakan dalam sistem ini adalah IC DS18B20, sensor ini memiliki presisi tinggi. Sensor ini sangat sederhana dengan hanya memiliki buah 3 kaki. Kaki pertama IC DS18B20 dihubung kesumber daya, kaki kedua sebagai output dan kaki ketiga di hubungkan ke ground.



Gambar 2.4. Sensor Suhu DS18B20

Karakteristik dari IC DS18B20 adalah sebagai berikut:

1. Dapat dikalibrasikan langsung ke dalam besaran Celsius.
2. Faktor skala linear $+10\text{mV}/^{\circ}\text{C}$.
3. Tingkat akurasi $0,5^{\circ}\text{C}$. Saat suhu kamar (25°C).
4. Jangkauan suhu antara -55°C . Sampai 150°C .
5. Bekerja pada tegangan 4 volt sehingga 30 volt.
6. Arus kerja kurang dari $60\ \mu\text{A}$.
7. Impedensi keluaran rendah $0,1\Omega$ untuk beban 1 mA.

Sensor DS18B20 bekerja dengan mengubah besaran suhu menjadi besaran tegangan. Tegangan ideal yang keluar dari DS18B20 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*) rangkain control yang sangat mudah.(Arduino,2010)

Mikrokontroler Ic DS18B20 dapat langsung dihubungkan dengan PINA. Dimana PINA merupakan PIN mikrokontroler yang dapat mengkonversi tegangan

menjadi bilangan digital (*analog digital conversation*) atau lebih dikenal dengan ADC.

2.4. Buzzer

Buzzer adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja buzzer hampir sama dengan loud speaker, jadi buzzer juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara. Buzzer biasa digunakan sebagai indikator bahwa proses telah selesai atau terjadi suatu kesalahan pada sebuah alat (alarm).



Gambar 2.5. Buzzer

Buzzer elektronik ini dapat diciptakan dengan merangkaikan beberapa komponen yang pada prinsipnya alat ini dapat menimbulkan pulsa dimana arus listrik adalah sebagai indikator terciptanya pulsa tersebut. Pada pembuatan alat ini, kami menggunakan IC NE555 sebagai sistem pembangkit pulsa yang tentunya arus listrik adalah indikator utama daripada pembangkit sinyalnya. Namun IC NE555 bukanlah satu – satunya komponen yang digunakan. Disini kami menambahkan beberapa komponen pelengkap yang sama bergunanya dalam pembuatan alat ini. Komponen – komponen itu antara lain adalah 1 buah potensiometer 10k, 2 buah kapasitor 0,01uF, 1 buah kapasitor 1uF, 1 buah IC NE555, 1 buah Speaker 8Ω, 0,5watt, 1 buah baterai 9volt, dan 1 buah transistor NPN 9013.

2.5. Liquid Crystal Display (LCD)

Liquid Crystal Display (LCD) merupakan perangkat display yang paling umum dipasangkan ke mikrokontroller, mengingat ukurannya yang kecil dan kemampuan menampilkan karakter atau grafik yang lebih baik dibandingkan display 7segment ataupun alphanumeric.

Pada pengembangan sistem *embedded*, LCD mutlak diperlukan sebagai sumber pemberi informasi utama, misalnya alat pengukur kadar gula darah, penampil jam, penampil *counter* putaran motor industri dan lainnya.

LCD (*Liquid Crystal Display*) adalah modul penampil yang banyak digunakan karena tampilannya menarik. LCD yang paling banyak digunakan saat ini adalah LCD M1632 karena harganya cukup murah. LCD M1632 merupakan modul LCD dengan tampilan (2baris x 16 kolom) dengan konsumsi daya rendah.

Tabel 2.2. Koneksi Antara Modul LCD dengan Mikrokontroller

Pin LCD	Keterangan	Pin Mikrokontroller	Keterangan
1	GND	11	GND
2	+5 V	10	VCC
4	RS	22	Port C.0
5	RD	23	Port C.1
6	EN	24	Port C.2
11	D4	26	Port C.4
12	D5	27	Port C.5

13	D6	28	Port C.6
14	D7	29	Port C.7

2.6 Relay

Relay merupakan salah satu jenis komponen elektronika yang banyak diaplikasi di berbagai macam rangkaian elektronika mulai dari yang sederhana sampai dengan yang rumit. Perlu diketahui bahwa relay sebenarnya adalah sebuah saklar elektrik yang digerakkan oleh listrik. Cara kerja relay sebenarnya tak berbeda dibanding dengan saklar biasa. Perlu diketahui bahwa relay menggunakan prinsip elektromagnetik sebagai penggerak kontak saklar. Jadi tanpa listrik komponen relay tidak dapat digunakan. Relay adalah komponen elektronika yang berupa saklar atau switch elektrik yang dioperasikan menggunakan listrik. Relay juga biasa disebut sebagai komponen electromechanical atau elektromekanikal yang terdiri dari dua bagian utama yaitu coil atau elektromagnet dan kontak saklar atau mekanikal. Komponen relay menggunakan prinsip elektromagnetik sebagai penggerak kontak saklar, sehingga dengan menggunakan arus listrik yang kecil atau low power, dapat menghantarkan arus listrik yang memiliki tegangan lebih tinggi. Berikut adalah gambar relay



Gambar 2.6 relay

Seperti yang telah dikatakan tadi bahwa relay memiliki fungsi sebagai saklar elektrik. Namun jika diaplikasikan ke dalam rangkaian elektronika, relay memiliki

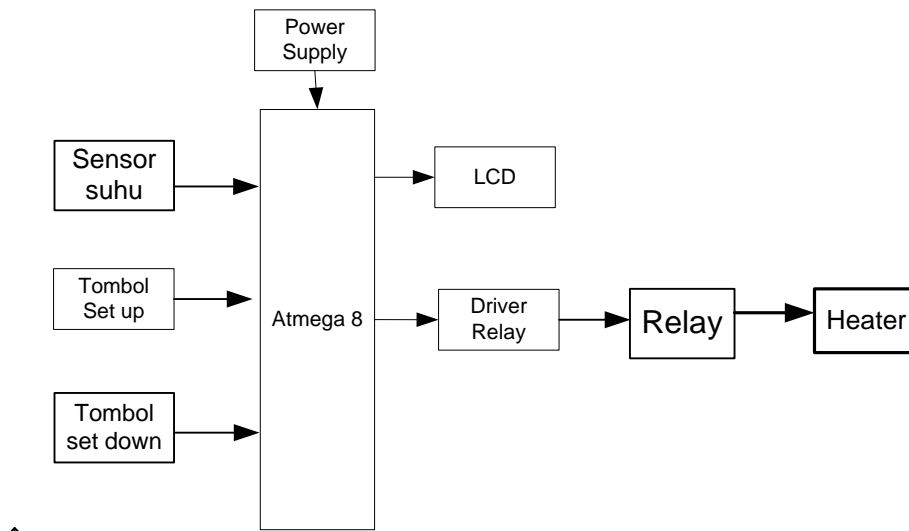
beberapa fungsi yang cukup unik. Berikut adalah beberapa fungsi komponen relay saat diaplikasikan ke dalam sebuah rangkaian elektronika.

1. Mengendalikan sirkuit tegangan tinggi dengan menggunakan bantuan signal tegangan rendah
2. Menjalankan fungsi logika alias logic function
3. Memberikan fungsi penundaan waktu alias time delay function
4. Melindungi motor atau komponen lainnya dari kelebihan tegangan atau korsleting

Setelah mengetahui pengertian dan fungsi relay, berikut adalah cara kerja atau prinsip kerja relay yang juga harus anda ketahui. Namun sebelumnya anda perlu tahu bahwa dalam sebuah relay terdapat 4 buah bagian penting yakni Electromagnet (Coil), Armature, Switch Contact Point (Saklar), dan Spring. Sebuah Besi (Iron Core) yang dililit oleh kumparan Coil, berfungsi untuk mengendalikan Besi tersebut. Apabila Kumparan Coil dialiri arus listrik, maka akan muncul gaya elektromagnetik yang dapat menarik Armature sehingga dapat berpindah dari posisi sebelumnya tertutup (NC) menjadi posisi baru yakni terbuka (NO).

BAB 3 PERANCANGAN DAN PEMBUATAN

3.1. Diagram Blok Sistem



Gambar 3.1 Diagram blok system

3.1.1. Fungsi-fungsi diagram blok

1. Blok sensor 1 sebagai pendeteksi suhu.
2. Blok Tombol set up sebagai input / pengatur suhu (naik).
3. Blok Tombol set down sebagai input / pengatur suhu (turun).
4. Blok Supply sebagai sumber tegangan.
5. Blok Relay sebagai pengatur arus yang masuk (pengontrol).
6. Blok LCD sebagai tampilan.
7. Blok heater sebagai output.

Alat pengontrol temperature air menggunakan mikrokontroler sebagai inti atau otak untuk menjelaskan kerja alat. Penggunaan sensor pada alat pengontrol temperatur air adalah sebagai pendeteksi suhu panas air. Ketika alat ON *Heater* akan berlangsung aktif memanaskan air dan miktrokontroler akan bekerja menstabilkan suhu, kemudian LCD menampilkan suhu pembacaan dari sensor juga untuk menampilkan set suhu dan selisih dari suhu yang terbaca dengan suhu yang diinginkan.

3.1.2. Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan adalah observasi dan pengukuran. Observasi disini adalah melakukan pengamatan terhadap objek yang diuji, selanjutnya dari pengujian tersebut dilakukan pengukuran. Pengujian tersebut dimaksudkan untuk mengetahui kualitas alat yang dirancang apakah sesuai dengan yang dirancang (target) atau tidak. Apabila sudah dapat seperti target atau dapat mendekati target maka alat tersebut dapat dikatakan bagus. Target disini berdasarkan perencanaan alat yang dibuat.

3.1.3. Instrumen

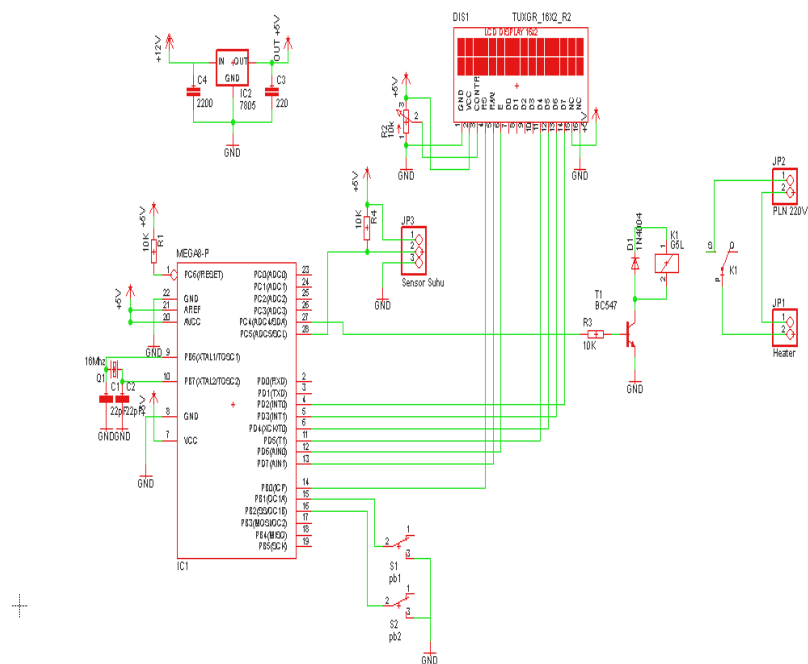
Instrumen adalah alat ukur yang digunakan untuk pengukuran dalam eksperimen. Alat-alat ukur yang digunakan harus mempunyai tingkat validitas yang tinggi artinya sebuah instrumen dikatakan valid apabila mampu mengukur secara tepat atau mendeteksi harga sesungguhnya. Selain valid, sebuah instrument juga harus mempunyai tingkat realibilitas yang baik. Instrumen hanya dapat dipercaya bila data yang diperoleh sesuai dengan kenyataan.

3.1.4. Teknik Analisa Data

Pengukuran untuk kerja alat ini bertujuan untuk mengetahui kualitas kerja apakah alat ini dapat bekerja sesuai dengan harapan dalam perancangan atau tidak. Teknik analisa data disini menggunakan metode analisi diskriptif yaitu membandingkan antara perhitungan perencanaan dengan pengukuran atau pengamatan hasil eksperimen. Apabila terjadi penyimpangan dilakukan identifikasi dari penyimpangan tersebut.

3.2 Perancangan Pembuatan Sistem Mikrokontroler

Langkah pertama dalam perancangan *hardware* adalah membuat PCB *mainboard*. Sebelum membuat PCB terlebih dahulu merancang skema rangkain *layout* PCBnya. Gambar 3.3 terlihat skema rangkain sistem minimum mikrokontroler sebagai pengendali.



Gambar 3.2. Skema rangkain sistem minimum mikrokontroler

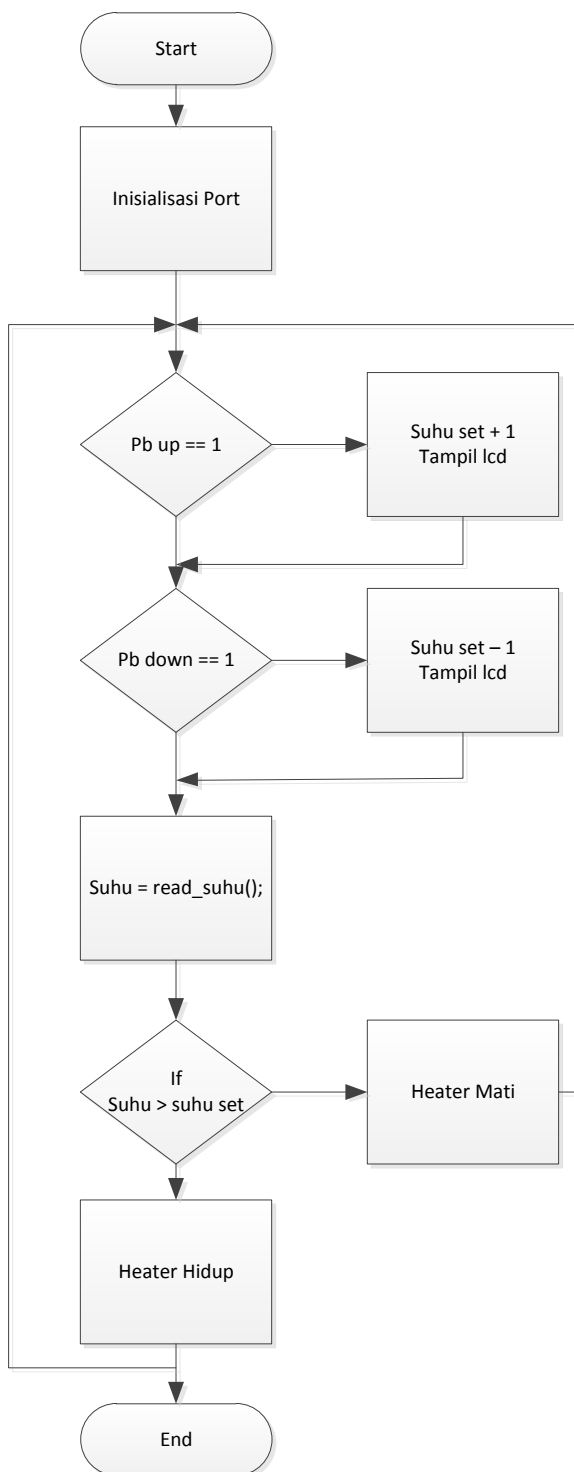
Selanjutnya setelah skema rangkain selesai dibuat adalah membuat PCB. Dalam pembuatan PCB terdapat beberapa tahapan yang harus dilakukan. *Software* yang digunakan dalam pembuatan PCB ini adalah *Eagle PCB*. Di dalam *software* tersebut sudah terdapat *database*. Komponen-komponen yang dibutuhkan sehingga dapat langsung menggunakannya sesuai kebutuhan.

Langkah-langkah dalam pembuatan PCB adalah sebagai berikut:

1. Terlebih dahulu buat desaint layout dengan *software Eagle PCB*
2. Setelah layout jadi kemudian diprint dengan printer laser dan menggunakan kertas foto.
3. Langkah selanjutnya adalah menyablon layout tersebut di PCB polos yaitu dengan cara menyetrika. Sebelum disetrika sebaiknya PCB yang akan dicuci menggunakan bensin untuk menghilangkan debu-debu menempel pada lapisan tembaga sehingga tinta menempel dengan baik saat disetrika.
4. Setelah tinta menempel dengan baik pada PCB polos maka langkah selanjutnya adalah melarutkannya. Untuk melarutkan jalur tembaga yang tidak digunakan dapat menggunakan larutan *feri chloride (FeCl13)*.

5. Selanjutnya setelah jalur rangkain yang diinginkan sudah sesuai dengan yang diinginkan maka langkah selanjutnya adalah melakukan pengeboran pada trik-trik yang digunakan untuk dudukan komponen.
6. Setelah komponen terpasang sesuai dengan tata letaknya maka langkah selanjutnya adalah melakukan penyolderan. Pastikan bahwa penyolderan tonol menempel dengan baik sehingga komponen akan terhubung dengan baik.
7. Langkah terbaik adalah melakukan pengecekan pada jalur rangkain.

3.3 Flowchart Software



Gambar 3.3. *Flowchart software*

Penjelasan *flowchat software* yang dibuat adalah sebagai berikut:

Ketika mulai (alat dinyalakan), sistem akan bekerja mendeteksi dan menginisialisasi variable, kemudian menampilkan tampilan awal alat pada LCD berupa tampilan “pengontrol suhu air”. Mikrokontroler memproses input dari keluaran sensor digital akan ditampilkan pada LCD berupa pembacaan suhu dalam °C dan menampilkan selisih antara pembacaan dengan batas suhu yang ditentukan.

Set suhu kita atur menggunakan pussbuttom naik atau turunnya suhu, batas suhu yang telah ditentukan adalah 100 °C. Ketika alat ON mikrokontroler akan langsung bekerja untuk memulai pemanasan untuk mengaktifkan heater yang berfungsi sebagai pemanas air. Kemudian mikrokontroler membaca batas suhu air, apabila suhu pembacaan belum mencapai batas suhu yang ditentukan maka mikrokontroler akan terus bekerja mengaktifkan pemanas (*heater*) secara bertahap untuk memanaskan air agar sampai pada batas suhu yang ditentukan dengan dikontrol oleh relay. Apabila batas suhu telah tercapai sebaliknya pemanas akan berhenti memanaskan air. Proses penstabilan tersebut akan terus berulang sebelum menekan tombol OFF dan mencabut stop kontak rangkaian catu daya dari sumber tegangan AC.

BAB IV PENGUJIAN DAN PEMBAHASAN SISTEM

4.1. Pengujian LCD

Pengujian LCD ini bertujuan mengetahui apakah LCD yang digunakan dalam keadaan baik atau tidak. Pengujian dilakukan dengan memberikan tegangan dengan kaki catu (kaki 2 dan 15), kemudian mengisi program ke sistem mikrokontroler yang akan ditampilkan oleh LCD melalui PORT C sebagai contoh penampilakan Tulisan **“T air dan T set”**

```
#include <LiquidCrystal.h>

#include <OneWire.h>

LiquidCrystal lcd(8, 6, 5, 4, 3, 2);

#define buzzer 1

OneWire  ds(A5);

#define pb1 9
#define pb2 10
#define pb3 11
#define heater A4

byte i;

byte present = 0;

byte type_s;

byte data[12];

byte addr[8];

float celsius, fahrenheit;

int suhuset=30;

int temp;
```

```
void suhu(){  
    if ( !ds.search(addr)) {  
        ds.reset_search();  
        delay(250);  
    }  
    for( i = 0; i < 8; i++) {  
    }  
    if (OneWire::crc8(addr, 7) != addr[7]) {  
        return;  
    }  
    switch (addr[0]) {  
        case 0x10:  
            type_s = 1;  
            break;  
        case 0x28:  
            type_s = 0;  
            break;  
        case 0x22:  
            type_s = 0;  
            break;  
        default:  
            return;  
    }  
  
    ds.reset();  
    ds.select(addr);
```

```

ds.write(0x44, 1);
present = ds.reset();
ds.select(addr);
ds.write(0xBE);
for ( i = 0; i < 9; i++) {
    data[i] = ds.read();
}
int16_t raw = (data[1] << 8) | data[0];
if (type_s) {
    raw = raw << 3;
    if (data[7] == 0x10) {
        raw = (raw & 0xFFF0) + 12 - data[6];
    }
} else {
    byte cfg = (data[4] & 0x60);
    if (cfg == 0x00) raw = raw & ~7;
    else if (cfg == 0x20) raw = raw & ~3;
    else if (cfg == 0x40) raw = raw & ~1;
}
celsius = (float)raw / 16.0;
}

void setup() {
    lcd.begin(16,2);
    pinMode(pb1 ,INPUT);
    pinMode(pb2 ,INPUT);
}

```

```

pinMode(pb3 ,INPUT);

pinMode(heater ,OUTPUT);

pinMode(buzzer ,OUTPUT);

}

void loop() {

    suhu();

    temp=celsius;

    digitalWrite(pb1,HIGH);

    digitalWrite(pb2,HIGH);

    digitalWrite(pb3,HIGH);

    if (digitalRead(pb1)==0){suhuset--;}

    else if (digitalRead(pb2)==0){suhuset++;}

    else if (temp < suhuset) {digitalWrite(heater,HIGH);
lcd.setCursor(14,1);lcd.print("H");}

    else if (temp > suhuset) {digitalWrite(heater,LOW);}

    else if
(temp==suhuset){;digitalWrite(buzzer,HIGH);delay(50);digitalWrite(buzzer,LOW);delay(500);}

    lcd.clear();

    lcd.setCursor(0,0);

    lcd.print("T Air: ");

    lcd.print(celsius);

```

```
    lcd.print(" ");  
    lcd.write(0xDF);  
    lcd.print("C");  
    lcd.setCursor(0,1);  
    lcd.print("T Set: ");  
    lcd.print(suhuset);  
    delay(200);  
}
```

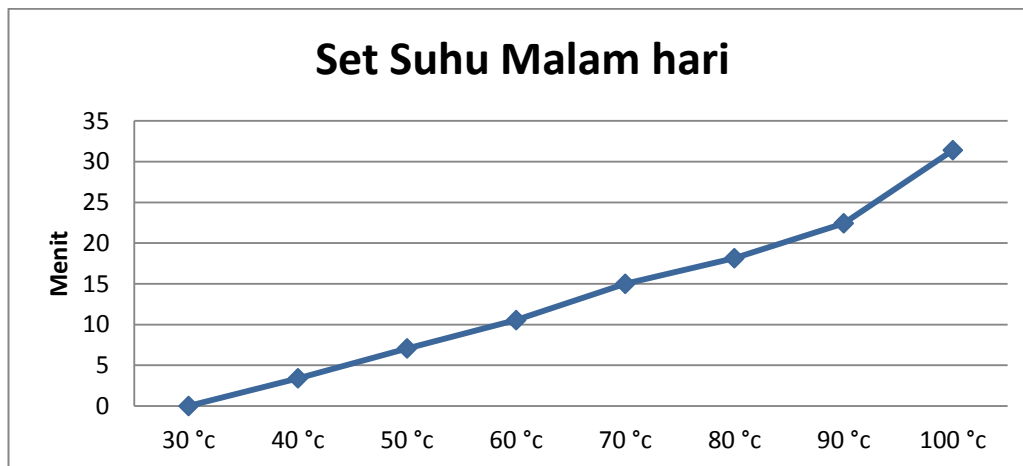

4.2. Pengujian Kerja Alat Pengontrol Temperatur Air

Dalam pengujian kerja alat rangkaian sensor suhu DS18S20 yang diaplikasikan pada *water heater* dengan tampilan LCD ini. Bahwa suhu yang terbaca pada sensor dengan yang ditampilkan pada LCD sangat baik. Hasil perbandingan pengukuran tersebut dengan volume air 3000 cc dapat ditunjukkan pada beberapa tabel percobaan dibawah ini :

Tabel 4.1. Hasil pengujian kerja alat.

Pengujian	Perubahan suhu	
	Menit	Kenaikan
Pagi	37,28	100 °C
Siang	35,46	100 °C
Malam	31,40	100 °C

4.5. Grafik Set Suhu Pada Malam Hari

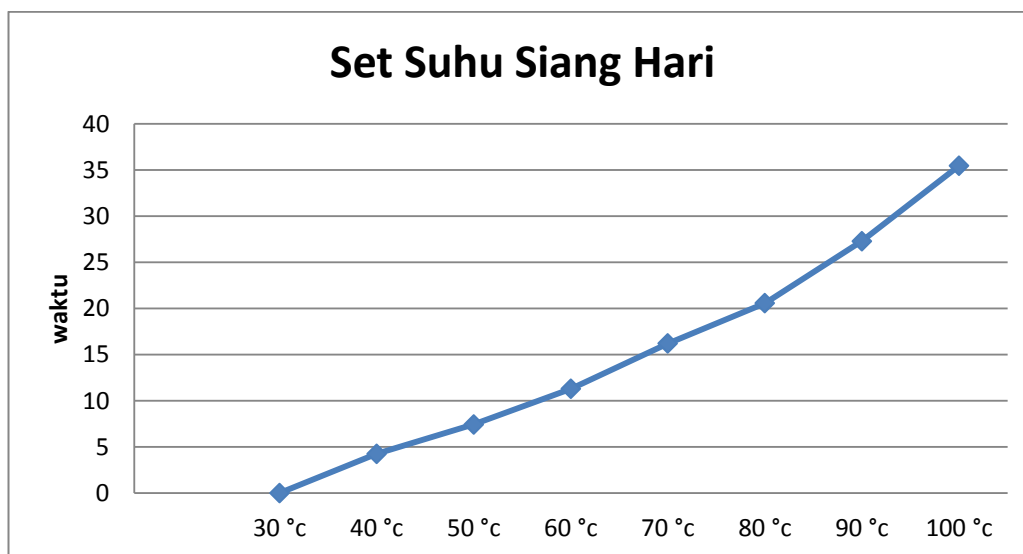


Gambar 4.1. Grafik set suhu pada Malam hari

Waktu (menit)	Suhu
0	30 °C
03:46	40 °C
07:06	50 °C
10:56	60 °C
15:01	70 °C
18:15	80 °C
22:43	90 °C
31:40	100 °C

Tabel 4.2 Perubahan suhu pada malam hari

4.6. Grafik Set Suhu Pada Siang Hari

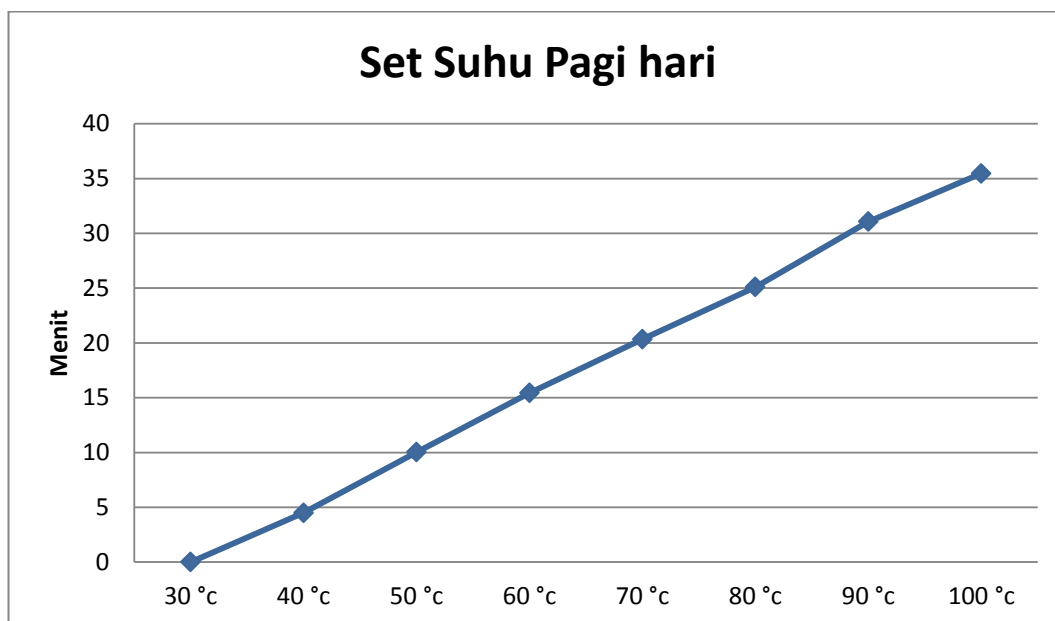


Gambar 4.2. Grafik set suhu pada siang hari

Waktu (menit)	Suhu
0	30 °C
04:25	40 °C
07:44	50 °C
11:30	60 °C
16:21	70 °C
20:57	80 °C
27:30	90 °C
35:46	100 °C

Tabel 4.3 Perubahan suhu pada siang hari

4.6. Grafik Set Suhu Pada Pagi Hari



Gambar 4.3. Grafik set suhu pada pagi hari

Waktu (menit)	Suhu
0	30 °C
04:50	40 °C
10:04	50 °C
15:44	60 °C
20:35	70 °C
25:10	80 °C
31:07	90 °C
37:28	100 °C

Tabel 4.4 Perubahan suhu pada pagi hari

4.6. Pembahasan Hasil Pengujian Alat Pengontrol Temperatur Air

Dalam pembuatan rangkaian sensor DS18S20 yang di aplikasikan ke water heater dengan tampilan LCD dikatakan sangat baik. Alat bekerja dengan baik karena mampu menampilkan suhu terukur atau terbaca oleh sensor, set suhu yang telah ditentukan menggunakan, dengan selisih suhu antara suhu terukur dengan set suhu yang ditentukan pada LCD. Sensor mendeteksi suhu awal air kemudian diolah pada mikrokontroller sehingga hasilnya dapat ditampilkan pada LCD. Saat suhu kurang atau lebih dari 100 °C, maka relay secara otomatis akan mengatur tegangan masukan ke elemen pemanas(heater) sehingga suhu air akan stabil. Di dalam mikrokontroller diatur range dari 100 °C adalah ± 1 °C.

Hasil dari pengujian alat penstabil temperatur air diatas dilakukan dengan membandingkan 3 kali hasil pengujian yang telah dilakukan. Pada pengujian dengan set suhu 100 °C, yang terlihat pada tabel 4.1 menunjukkan bahwa waktu tempuh yang dicapai untuk mencapai suhu 100 °C tersebut adalah 37,28 menit, 35,46 menit, dan 31,40 menit. Dilihat dari hasil setiap pengujian diatas, pada pengujian pagi hari waktu yang dibutuhkan lebih lama dari pengujian siang dan malam hari dikarenakan suhu ruang relatif lebih dingin sehingga mempengaruhi suhu air dan lama waktu pemanasan air. Ketika dilihat dari rata-rata waktu tempuh seluruh pengujian yang telah dilakukan, waktu tempuh tersebut dirasa sangat lama karena daya yang masuk ke elemen pemanas kecil. Daya yang masuk tersebut kecil karena pengaturan kestabilan yang terdapat didalam alat. Pengaturan tersebut dilakukan agar kenaikan suhu saat memanaskan air lebih teratur dengan baik.

Meski terdapat kenaikan suhu dalam setiap pengujian alat hingga mencapai angka 98 °C. Hasil tersebut masuk kedalam kategori yang sangat baik. Terjadinya kenaikan suhu dikarenakan ketika proses pemanasan air water heater tidak mati tapi masih terdapat daya yang masuk kedalam elemen pemanas yang diatur oleh relay sehingga masih terjadi proses pemanasan air.

BAB V PENUTUP

5.1.Kesimpulan

Setelah melakukan tahap perancangan dan pembuatan sistem yang kemudian dilanjutkan dengan tahap pengujian dan analisa maka dapat diambil kesimpulan sebagai berikut :

1. Prinsip Kerja pada Relay adalah menggerakkan kontak saklar sehingga dengan arus kecil dapat menghantar listrik dengan tegangan yang lebih tinggi yang dapat mengontrol masuknya tegangan ke dalam heater.
2. Dalam aplikasi alat kontrol suhu air ini dapat disimpulkan bahwa ketika sensor mendeteksi perubahan suhu pada air ,Maka Mikrokontroler akan menampilkan data ke LCD, Jika suhu mencapai 40 ⁰C maka suhu akan dipertahankan oleh relay dengan range yang sudah diatur satu dan buzzer akan hidup ketika set suhu sudah mencapai angka yang diinginkan yang dikontrol menggunakan keypad.

5.2.Saran

Dari hasil Tugas Akhir ini masih terdapat beberapa kekurangan dan dimungkinkan untuk pengembangan lebih lanjut. Oleh karenanya penulis merasa perlu untuk memberi saran sebagai berikut:

1. Pada Tugas akhir ini dengan menggunakan Alat pengontrol suhu otomatis pada air, Agar lebih teliti lagi dalam mengambil data agar data yang di dapat lebih akurat.

DAFTAR PUSTAKA

Akmal, Imelda. 2005. *Seni Menata Kamar Mandi*. Yudistira:Jakarta

Anonim. 1995. *Ds18B20 Precision Centigrade Temperatur Sensor*. Nasional Semiconductors: Jakarta

Bejo, Agus. 2008. *C&AVR Rahasia Kemudahan Bahasa C dalam Mikrokontr*. Graha Ilmu : Yogyakarta

Nugroho. Agfianto. 2008. *LCD MI632*. Seiko Instrumen: Japan

<http://shatomedia.com/2008/12/sensor-suhu-DS18B20/>

<http://www.digi-were.com/LCD/>

<http://www.texasinstrumentsincorporated.com/>

Kurniawan, Kuku. 2011. *Penstabil Temperatur Air Berbasis Mikrokontroler*. Researchgate:Universitas Negeri Semarang.

<http://belajarelelektronika.net/pengertian-fungsi-dan-cara-kerja-relay/>

Lampiran 1

PROGRAM LENGKAP

```
#include <LiquidCrystal.h>

#include <OneWire.h>

LiquidCrystal lcd(8, 6, 5, 4, 3, 2);

#define buzzer 1


OneWire ds(A5);

#define pb1 9

#define pb2 10

#define pb3 11

#define heater A4


byte i;

byte present = 0;

byte type_s;

byte data[12];

byte addr[8];

float celsius, fahrenheit;

int suhuset=30;

int temp;


void suhu(){

    if ( !ds.search(addr)) {

        ds.reset_search();
```



```
    delay(250);  
}  
for( i = 0; i < 8; i++) {  
}  
if (OneWire::crc8(addr, 7) != addr[7]) {  
    return;  
}  
switch (addr[0]) {  
    case 0x10:  
        type_s = 1;  
        break;  
    case 0x28:  
        type_s = 0;  
        break;  
    case 0x22:  
        type_s = 0;  
        break;  
    default:  
        return;  
}  
  
ds.reset();  
ds.select(addr);  
ds.write(0x44, 1);  
present = ds.reset();  
ds.select(addr);
```

```

ds.write(0xBE);

for ( i = 0; i < 9; i++) {

    data[i] = ds.read();

}

int16_t raw = (data[1] << 8) | data[0];

if (type_s) {

    raw = raw << 3;

    if (data[7] == 0x10) {

        raw = (raw & 0xFFF0) + 12 - data[6];

    }

} else {

    byte cfg = (data[4] & 0x60);

    if (cfg == 0x00) raw = raw & ~7;

    else if (cfg == 0x20) raw = raw & ~3;

    else if (cfg == 0x40) raw = raw & ~1;

}

celsius = (float)raw / 16.0;

}

```

```

void setup() {

lcd.begin(16,2);

pinMode(pb1 ,INPUT);

pinMode(pb2 ,INPUT);

pinMode(pb3 ,INPUT);

pinMode(heater ,OUTPUT);

pinMode(buzzer ,OUTPUT);

```

```
}
```

```
void loop() {
```

```
    suhu();
```

```
    temp=celsius;
```

```
    digitalWrite(pb1,HIGH);
```

```
    digitalWrite(pb2,HIGH);
```

```
    digitalWrite(pb3,HIGH);
```

```
    if (digitalRead(pb1)==0){ suhuset--;}

```

```
    else if (digitalRead(pb2)==0){ suhuset++;}

```

```
    else if (temp < suhuset) { digitalWrite(heater,HIGH);
    lcd.setCursor(14,1);lcd.print("H");}
```

```
    else if (temp > suhuset) { digitalWrite(heater,LOW);}
```

```
    else if
    (temp==suhuset){ ;digitalWrite(buzzer,HIGH);delay(50);digitalWrite(buzzer,LOW);
    delay(500);}
```

```
    lcd.clear();
```

```
    lcd.setCursor(0,0);
```

```
    lcd.print("T Air: ");
```

```
    lcd.print(celsius);
```

```
    lcd.print(" ");
```

```
    lcd.write(0xDF);  
    lcd.print("C");  
    lcd.setCursor(0,1);  
    lcd.print("T Set: ");  
    lcd.print(suhuset);  
    delay(200);  
}
```


Lampiran 3

DATA SHEET