



PROYEK AKHIR - VE 180626

**RANCANG BANGUN ANTISIPASI *FREEZING* ALIRAN
GAS PADA PRS 10 MENGGUNAKAN
MIKROKONTROLER DI PT. PGN SOLUTION**

Ahmad Abiyyu
NRP 10311600000041

Dosen Pembimbing
Slamet Budiprayetno, ST., MT.
Lucky Putri Rahayu, S.Si., M.Si.

Pembimbing Lapangan
Ija Misja

DEPARTEMEN TEKNIK ELEKTRO OTOMASI
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya
2019

----Halaman ini sengaja dikosongkan-----



FINAL PROJECT - VE 180626

**PROTOTYPE FREEZING ANTICIPATION OF GAS
FLOWING IN PRS 10 USING MICROCONTROLLER AT
PT. PGN SOLUTION**

Ahmad Abiyyu
NRP 10311600000041

Supervisor
Slamet Budiprayetno, ST., MT.
Lucky Putri Rahayu, S.Si., M.Si.

Field Guide
Ija Misja

DEPARTEMENT OF ELECTRICAL AUTOMATION ENGINEERING
Vocational Faculty
Sepuluh Nopember Institute of Technology
Surabaya 2019

----Halaman ini sengaja dikosongkan-----

PERNYATAAN KEASLIAAN PROYEK AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Proyek Akhir saya dengan judul “Rancang Bangun Antisipasi *Freezing* Pada Aliran Gas Pada PRS 10 Menggunakan Mikrokontroler di PT. PGN Solution” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 20 Juli 2019



Ahmad Abiyyu
NRP 10311600000041

----Halaman ini sengaja dikosongkan-----

**RANCANG BANGUN ANTISIPASI *FREEZING* ALIRAN GAS
PADA PRS 10 MENGGUNAKAN MIKROKONTROLER DI PT.
PGN SOLUTION**

LEMBAR PENGESAHAN

PROYEK AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya
Pada
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember

Menyetujui,

Dosen Pembimbing 1

Dosen Pembimbing 2



Lucky Putri Rahayu, S.Si., M.Si.

NIP. 1991201712058

Pembimbing Lapangan

Ija Misia

**Surabaya
Juni, 2019**

----Halaman ini sengaja dikosongkan----

RANCANG BANGUN ANTISIPASI *FREEZING* PADA ALIRAN GAS PADA PRS 10 MENGGUNAKAN MIKROKONTROLER DI PT. PGN SOLUTION

Nama Mahasiswa : Ahmad Abiyyu
NRP : 10311600000041
Dosen Pembimbing 1 : Slamet Budiprayetno, ST., MT
Dosen Pembimbing 2 : Lucky Putri Rahayu, S.Si., M.Si.
Pembimbing Lapangan : Ija Misja
NIP 1 : 197811132010121002
NIP 2 : 1991201712058

ABSTRAK

Pada pemasangan PRS10 di lokasi konsumen, gas bertekanan tinggi diturunkan dua tahap dengan menggunakan dua regulator yang menjadi satu kesatuan dalam PRS. Tahap pertama dari tekanan 200 BarG diturunkan menjadi maksimal 40 BarG, kemudian tahap kedua dari tekanan 40 BarG diturunkan sesuai dengan tekanan yang dibutuhkan oleh konsumen yang berkisar antara 1 - 4 BarG sesuai dengan kebutuhan. Perbedaan penurunan tekanan yang besar dari keluaran CNG *Cradle* menimbulkan terjadinya bunga es pada regulator yang disebut *freezing*. Adanya bunga es pada regulator menyebabkan kinerja regulator tidak maksimal bahkan dapat mengakibatkan aliran gas terhenti (*blocking*).

Pada alat kontrol suhu ini difokuskan pada implementasi pengontrolan suhu pada regulator dan penggunaan efek peltier dengan menggunakan Sensor LM35 dan peltier sebagai *thermoelectric* yang digunakan sebagai pemanas untuk mencegah pembekuan dari regulator. Sebelum terjadi pembekuan, relay akan menyala setelah suhu terendah untuk mengaktifkan relay di *setting* terlebih dahulu menggunakan *keypad*.

Hasil dari Proyek Akhir ini adalah dapat mengontrol suhu pemanas yang dimana sensor suhu LM35 telah dilakukan kalibrasi dan perbandingan pengukuran dengan kesalahan pengukuran $10^{\circ}\text{C} \pm 0,56$, sedangkan pada $25^{\circ}\text{C} \pm 0,24$ dan $50^{\circ}\text{C} \pm 0,06$. Pemanas akan menyala ketika suhu regulator mencapai 0°C - 10°C dan Pemanas akan mati ketika suhu regulator mencapai 25°C - 40°C .

Kata Kunci : PRS10, Mengontrol, Sensor Suhu LM35

-

----Halaman ini sengaja dikosongkan-----

**PROTOTYPE FREEZING ANTICIPATION OFF GAS
FLOWING IN PRS 10 USING MICROCONTROLLER AT PT.
PGN SOLUTION**

Student Name : Ahmad Abiyyu
Student ID : 10311600000041
Supervisor 1 : Slamet Budiprayetno,ST.,MT
Supervisor 2 : Lucky Putri Rahayu,S.Si., M.Si.
Supervisor 3 : Ija Misja
ID 1 : 197811132010121002
ID 2 : 1991201712058

ABSTRACT

In the installation of PRS10 at the consumer location, high-pressure gas is reduced by two stages by using two regulators which become a single entity in the PRS. The first stage of the pressure of 200 BarG is lowered to a maximum of 40 BarG, then the second stage of pressure 40 BarG is lowered according to the pressure required by consumers ranging from 1 to 4 BarG according to needs. The difference pressure drop from CNG output on the regulator called freezing. The presence of an ice flower on the regulator causes the regulator's performance to not be maximal and can even result in a blocking of gas flow..

In this tool, this temperature control is focused on the implementation of temperature control on the regulator and the use of peltier effects by using LM35 and peltier sensors as thermoelectric used as heaters to prevent freezing from regulators. Before freezing, the relay will turn on after the lowest temperature to activate the relay is set. first use the keypad.

The results of this Final Project are able to control the temperature of the heater where the LM35 temperature sensor has been calibrated and the comparison of measurements with measurement errors is $10^{\circ}\text{C} \pm 0.56$, while at $25^{\circ}\text{C} \pm 0.24$ and $50^{\circ}\text{C} \pm 0.06$. The heater will turn on when the regulator temperature reaches $0^{\circ}\text{C} - 10^{\circ}\text{C}$ and the Heater will turn off when the regulator temperature reaches $25^{\circ}\text{C} - 40^{\circ}\text{C}$.

Key Words : Controlling, Sensor Temperature, PRS10

-

----Halaman ini sengaja dikosongkan-----

KATA PENGANTAR

Puji dan syukur saya panjatkan kepada Allah SWT atas berkat dan rahmat-Nya penulis dapat melaksanakan Proyek Akhir yang berjudul **“Rancang Bangun Antisipasi Freezing Aliran Gas Pada PRS10 Menggunakan Mikrokontroler di PT. PGN Solution”**. Laporan ini disusun sebagai hasil Proyek Akhir pada tempat magang yang dilaksanakan sejak tanggal 2 Juli 2018 s.d. 28 September 2018.

Dalam pelaksanaan Proyek Akhir dan penyusunan laporan ini penulis memperoleh banyak bantuan serta bimbingan dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis mengucapkan banyak terimakasih kepada pihak-pihak yang telah banyak membantu penulis, khususnya pada:

1. Orang tua dan seluruh keluarga yang telah mendukung dan mendoakan penulis selama melaksanakan Proyek Akhir.
2. Bapak Ir.Joko susila MT. selaku ketua Departemen Teknik Elektro Otomasi Fakultas Vokasi Institut Teknologi Sepuluh Nopember.
3. Bapak Slamet Budipriyanto ST, MT., selaku Dosen Pembimbing 1 Proyek Akhir dan Lucky Putri Rahayu,S.Si., M.Si sebagai Dosen Pembimbing 2.
4. Bapak Ija Misja selaku Pembimbing Lapangan yang selalu memberikan masukan.
5. Bapak Tyas, dan staff Departemen Manufaktur dan Repair lainnya yang telah memberikan kesempatan dan membantu penulis sehingga penulis dapat melaksanakan kegiatan Proyek Akhir ini.
6. Rizky Abdul Majid yang dapat menjadi tempat bertukar ilmu dan masukannya
7. Faisal Tataq teman kelompok proyek akhir ini yang dapat menjadi tempat bertukar ilmu dan masukannya.
8. Kakak tingkat 2015 Fajar Alif dan Dwi Prasetyo yang memberikan bantuan,semangat, dan saran untuk bertukar pikiran.
9. Kepada teman-teman Angkatan 2016 Foxvire yang selalu menemani dalam susah maupun senang.
10. Serta semua pihak yang telah membantu saya dalam menyelesaikan Proyek Akhir ini yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Proyek Akhir ini. Akhir kata, semoga Proyek Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, 20 Juli 2019

Ahmad Abiyyu

DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN KEASLIAAN	Error! Bookmark not defined.
LEMBAR PENGESAHAN.....	Error! Bookmark not defined.
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR.....	xvii
DAFTAR TABEL	xix
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Permasalahan.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan	3
1.5 Sistematika Laporan.....	3
1.6 Relevansi	4
BAB II TEORI DASAR.....	5
2.1 Efek Peltier.....	5
2.2 Arduino Atmega 2560	6
2.3 Sensor Suhu LM35.....	6
2.4 Keypad Membrane	8
2.5 Peltier	9
2.6 Liquid Crystal Display	9
2.7 Relay	10
2.8 Regulator Penurun Tekanan	11
2.9 Power Supply	11
BAB III PERANCANGAN HARDWARE DAN SOFTWARE.....	13
3.1 Diagram Sistem Kerja Alat	13
3.2 Perancangan Perangkat Keras Mekanik	14
3.3 Perancangan Perangkat Keras Elektrik.....	18
3.3.1 Perancangan Sensor Suhu LM35 dengan LCD yang dihubungkan dengan Arduino	18
3.3.2 Perancangan Relay 2 Channel Terhadap Suhu dengan Arduino Mega	21
3.3.3 Perancangan LCD yang dihubungkan dengan Arduino Mega 26	26
3.3.4 Perancangan yang dihubungkan dengan Arduino Mega.....	27
3.4 Perancangan Software	31
3.4.1 Pemrograman Pembacaan Sensor.....	31
BAB IV PENGUJIAN DAN ANALISA	35
4.1 Pengujian LCD.....	35

4.2	Pengujian <i>Password</i> dengan Menggunakan <i>Keypad</i>	36
4.3	Sensor Suhu LM35 yang di hubungkan dengan Arduino Mega.....	37
4.4	Pengujian Relay 2 <i>channel</i> Terhadap Suhu.....	47
BAB V PENUTUP		51
5.1	Kesimpulan.....	51
5.2	Saran	51
DAFTAR PUSTAKA		53
LAMPIRAN A		1
LAMPIRAN B		1
LAMPIRAN C		1
DAFTAR RIWAYAT HIDUP		1

DAFTAR GAMBAR

Gambar 2.1. Modifikasi Sistem Peltier.....	5
Gambar 2.2. <i>Board</i> Arduino Atmega 2560.....	6
Gambar 2.3. LM35	7
Gambar 2.4. <i>Keypad</i> 4x4 yang dihubungkan ke Arduino Uno	8
Gambar 2.5. Bentuk <i>Thermo-Electric</i> (TEC)	9
Gambar 2.6. LCD <i>display</i> 20x4.....	10
Gambar 2.7. Modul Relay	10
Gambar 2.8. Regulator	11
Gambar 2.9. <i>Power Supply</i>	12
Gambar 3.1. Diagram Fungsional.....	14
Gambar 3.2. Perancangan Alat Tampak Keseluruhan	15
Gambar 3.3. Perancangan Alat Tampak Depan	16
Gambar 3.4. Perancangan Alat Tampak Atas.....	16
Gambar 3.5. Pemasangan Alat Tampak Belakang.....	17
Gambar 3.6. Pemasangan Peltier pada Regulator PRS10	17
Gambar 3.7. Letak Sensor Suhu LM35	18
Gambar 3.8. Wiring Sensor LM35 Dengan LCD yang dihubungkan Dengan Arduino	19
Gambar 3.9. Alogaritma Program pada Software Arduino Penginisialisasian 2 Sensor Suhu LM35.....	20
Gambar 3.10. Alogaritma Program pada Software Arduino Menampilkan Suhu pada LM35	20
Gambar 3.11. Wiring Modul Relay dengan Peltier yang dihubungkan dengan Arduino.....	21
Gambar 3.12. Alogaritma Program pada Software Arduino Kondisi Inisialisasi relay dan sensor suhu LM35.....	22
Gambar 3.13. Alogaritma Program pada Software Arduino Kondisi untuk Mengaktifkan Relay dan Mematikan Relay	22
Gambar 3.14. Alogaritma Program pada Software Arduino Kondisi untuk mengisi <i>Set Point</i> 1.....	23
Gambar 3.15. Alogaritma Program pada Software Arduino Kondisi untuk mengisi <i>Set Point</i> 2.....	24
Gambar 3.16. Alogaritma Program pada Software Arduino Kondisi Setelah <i>Set Point</i> di Set	25
Gambar 3.17. Wiring LCD 20x4 dengan Modul I2C dengan Arduino	26
Gambar 3.18. Alogaritma Program pada Software Arduino Program Menampilkan LCD.....	27
Gambar 3.19. Wiring <i>Keypad</i> yang dilengkapi LCD dengan Arduino	27
Gambar 3.20. Alogaritma Program pada Software Arduino Program Penginisialisasian <i>Keypad</i> Kolom dan Baris	28

Gambar 3.21. Alogaritma Program pada Software Arduino Untuk menampilkan Angka	29
Gambar 3.22. Alogaritma Program pada Software Arduino Membuat Password	29
Gambar 3.23. Alogaritma Program pada Software Arduino Pengkondisian ketika password benar dan salah.....	30
Gambar 3.24. Alogaritma Program pada Software Arduino Kondisi jika password salah.....	30
Gambar 3.25. Setting Port pada Software Arduino IDE	32
Gambar 3.26. Pseudocode	33
Gambar 4.1. Pengujian LCD	35
Gambar 4.2. Pengujian Tombol <i>Keypad</i> dan <i>Password</i>	36
Gambar 4.3. Tampilan <i>Password</i> Benar	36
Gambar 4.4. Pengujian <i>Password</i> yang Tidak Sesuai	37
Gambar 4.5. Tampilan Jika Password Salah	37
Gambar 4.6. Pengujian Kalibrasi	38
Gambar 4.7. Hasil Pada Kalibrator	39
Gambar 4.8. Hasil Pada Alat	39
Gambar 4.9. Embun pada Regulator1	45
Gambar 4.10. Merupakan Gambar Penampilan 2 Sensor Suhu LM35 pada LCD	46
Gambar 4.11. Pemasangan Pemanas pada Regulator	48
Gambar 4.12. Pengujian Sensor Panas yang Plat Sebagai Penggantinya ...	48
Gambar 4.13. Hasil Pengujian Sensor Panas yang Plat Sebagai Penggantinya	48

DAFTAR TABEL

Tabel 4.1. Hasil Kalibrasi Sensor LM35 Pada Suhu 10°C	40
Tabel 4.2. Kalibrasi Sensor LM35 Pada Suhu 25°C.....	41
Tabel 4.3. Kalibrasi Sensor LM35 Pada Suhu 50°C.....	42
Tabel 4.4. Hasil Pengujian Ketika Pembukaan Gas dari cradle pada regulator1	44
Tabel 4.5. Hasil Pengujian Ketika Pembukaan Gas dari cradle pada regulator2	46
Tabel 4.6. Pengujian Kontrol Relay1 Terhadap Suhu1	47
Tabel 4.7. Pengujian Pemanas Terhadap Regulator1	47

----Halaman ini sengaja dikosongkan-----

BAB I

PENDAHULUAN

1.1 Latar Belakang

PT. Perusahaan Gas Negara Tbk (PT PGN) adalah perusahaan nasional Indonesia terbesar dibidang transportasi dan distribusi gas bumi yang berperan besar dalam pemenuhan gas bumi domestik. PT. PGN terus memperkuat pondasi dan bertransformasi dari perusahaan transmisi dan distribusi gas bumi menjadi penyedia solusi energi terintegrasi, yang mendorong pemanfaatan gas bumi untuk kebutuhan masyarakat dan industri.

PT. PGAS Solution (PGN Solution) didirikan oleh PT. Perusahaan Gas Negara Tbk sebagai salah satu *subsidiary* yang berfokus pada teknis dan operasional di bidang gas pada tanggal 6 Agustus 2009. Pendirian PT. PGAS Solution didasarkan melalui pertimbangan strategis dan diharapkan untuk lebih mandiri serta professional dalam menangani persoalan teknis dan proyek PT. Perusahaan Gas Negara Tbk. Sebagai *subsidiary*, PT.PGAS Solution akan terus mendukung dan meneruskan energi baik kepada area dan masyarakat yang lebih luas lagi melalui pengontrolan, perangkaian dan penjagaan seluruh jaringan pipa milik PT Perusahaan Gas Negara Tbk.

Dalam pendistribusian kepada pelanggan yang tidak dilewati jaringan pipa, PT. PGAS Solution melakukan pendistribusian gas dengan CNG *Cradle* (Tabung Gas Alam bertekanan ± 200 BarG), yang dilengkapi dengan PRS10 (*Pressure Regulating Station*). PRS10 adalah sistem yang menurunkan gas tekanan tinggi yang keluar dari CNG *Cradle* (Tabung Gas Alam bertekanan ± 200 BarG) dimana di dalamnya terdapat beberapa regulator penurun tekanan dengan laju aliran maksimal (*Flow Max*) $10\text{m}^3/\text{jam}$ yang ditempatkan di lokasi konsumen (dipinjamkan) selamamasa kontrak.

Pada pemasangan di lokasi konsumen, gas bertekanan tinggi tersebut diturunkan dua tahap dengan menggunakan 2 regulator yang menjadi satu kesatuan dalam PRS. Tahap pertama dari tekanan 200 BarG diturunkan menjadi maksimal 40 BarG, kemudian tahap kedua dari tekanan 40 BarG diturunkan sesuai dengan tekanan yang dibutuhkan oleh konsumen yang berkisar antara 1 - 4 BarG sesuai dengan kebutuhan mesin/alat produksi.

Perbedaan penurunan tekanan yang besar dari keluaran CNG *Cradle* ketekanan sesuai dengan permintaan pelanggan dan laju alir gas mendekati batas maksimal kemampuan regulator dapat menimbulkan terjadinya bunga es pada regulator tersebut. Adanya bunga es pada regulator menyebabkan kinerja regulator tidak maksimal bahkan dapat mengakibatkan aliran gas terhenti (*blocking*).

Pembekuan merupakan proses penurunan suhu sampai suhu dibawah titik beku sampai berubah menjadi es (Rahayoe, 2004). Biasanya dalam suatu industri dibutuhkan suatu pemanas listrik atau perangkat yang dapat mengontrol suhu agar mendapatkan suhu konstan. Pada PT. PGAS Solution (PGN Solution) menggunakan suatu jenis *heater* yang dapat menghabiskan biaya yang sangat mahal untuk mencegah pembekuan yang terjadi pada satu PRS10. Selain itu, dibutuhkan juga pengontrol dan pendeteksi panas ataupun dingin. Sehingga bisa mendeteksi pembekuan yang terjadi pada suatu benda (Mukhlis, *et al.*, 2010).

Salah satu upaya untuk menggantikan efek dingin yang ditimbulkan pada regulator adalah dengan memanfaatkan salah satu metode *thermoelectric*, yaitu memanfaatkan efek peltier. Beberapa penelitian sudah dilakukan untuk memanfaatkan efek peltier namun hanya pada salah satu sisi saja, yaitu sebagai pemanas saja atau pendingin saja. Pada proyek akhir ini memanfaatkan efek peltier sebagai pemanas (*heater*). Pada penelitian ini juga ditambahkan mikrokontroler “Arduino Atmega 2560” sebagai pengatur suhunya (Purwiyanti, *et al.*, 2017).

1.2 Permasalahan

Pada penelitian ini yang menjadi permasalahan utama perusahaan adalah untuk mencegah atau mengatisipasi bunga es pada sebuah regulator dengan mengatur *setting point* suhu terendah untuk mengaktifkan relay serta peltier dan *setting point* suhu tertinggi mematikan relay serta peltier.

1.3 Batasan Masalah

Dari perumusan masalah diatas, maka batasan masalah dari penelitian ini adalah:

1. Sensor suhu yang digunakan untuk mengukur regulator adalah sensor suhu LM35.

2. Pengukuran suhu dilakukan hanya sebatas pada regulator PRS10.
3. Jenis mikrokontroler yang digunakan pada penelitian ini adalah Arduino Atmega 2560.
4. Jenis pemanas yang digunakan pada penelitian ini adalah peltier dengan menggunakan sisi panasnya.

1.4 Tujuan

Penelitian Rancang Bangun Antisipasi *Freezing* Aliran Gas Pada PRS10 Menggunakan Mikrokontroler di PT. PGN Solution ini bertujuan untuk mencegah pembekuan atau bunga es pada regulator PRS10 dengan memanfaatkan efek panas pada peltier.

1.5 Sistematika Laporan

Pembahasan penelitian ini akan dibagi menjadi lima Bab dengan sistematika sebagai berikut:

Bab I Pendahuluan

Bab ini meliputi latar belakang, permasalahan, batasan masalah, tujuan, metodologi penelitian, sistematika laporan, dan relevansi.

Bab II Teori Dasar

Bab ini menjelaskan tentang tinjauan pustaka, konsep dari efek peltier, Arduino Atmega 2560, LM35, Keypad, Peltier, Module, Regulator Penurun Tekanan, Power Supply.

Bab III Perancangan Sistem

Bab ini membahas perancangan sistem *hardware* maupun *software* pada antisipasi freezing pada aliran gas pada PRS10 menggunakan mikrokontroler air dengan menggunakan efek peltier, sensor suhu, dan pemantauan lokasi berdasarkan teori penunjang pada Bab II.

Bab IV Pengujian dan Analisa Sistem

Bab ini membahas tentang pengukuran, pengujian, serta analisa terhadap prinsip kerja dan proses dari suatu alat yang dibuat.

Bab V Penutup

Bab ini menjelaskan tentang kesimpulan dari Proyek Akhir dan saran – saran untuk pengembangan alat ini lebih lanjut.

1.6 Relevansi

Hasil yang diperoleh dari penelitian ini diharapkan dengan pengembangan lebih lanjut dapat dijadikan sebagai ide dalam kemajuan sistem kontrol suhu dan implementasinya. Dengan implementasi yang menggunakan kontrol suhu sebagai *setting point*-nya semoga dapat membantu pengontrolan otomatis atau manual dari suatu alat yang dapat mudah membeku.

BAB II TEORI DASAR

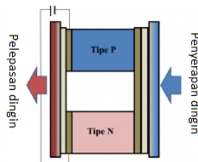
Pada bab ini akan membahas teori dasar dan teori penunjang terkait perangkat dan bahan yang digunakan dalam penelitian ini.

2.1 Efek Peltier

Efek peltier merupakan *thermoelectric* yang prinsip kerjanya merupakan kebalikan dari efek *seeback*. Efek peltier, di temukan oleh Jean Peltier pada tahun 1834, adalah fenomena dimana energi panas dapat diserap pada salah satu sambungan konduktor dan dilepaskan pada sambungan konduktor lainnya ketika arus listrik dialirkan pada suatu rangkaian tertutup atau dengan kata lain efek peltier mengkonversikan energi listrik menjadi perubahan suhu.

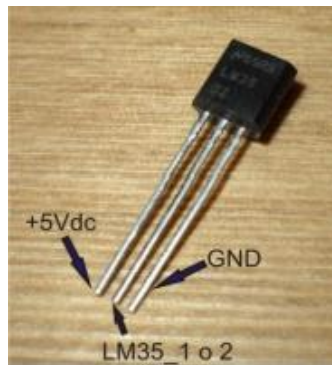
Prinsip kerja peltier diperlihatkan pada Gambar 2.1 sistem peltier, dua jenis semikonduktor (tipe n dan tipe p) disusun berdampingan. Apabila bahan semikonduktor tersebut dihubungkan dengan sumber tegangan, perbedaan energi fermi diantara kedua semikonduktor tersebut menyebabkan elektron akan mengalir dari semikonduktor tipe p ke tipe n dengan melewati *junction*. Elektron yang sampai pada tipe n akan berkombinasi dengan *hole* yang akan melepaskan energi dalam bentuk dingin. Sebaliknya, pada bagian p, elektron akan melepaskan diri dari ikatan valensinya dengan menyerap energi dingin.

Dengan demikian, arus yang melewati *junction*, baik arah maju ataupun mundur, akan menghasilkan perbedaan suhu. Suhu *junction* dingin tersebut dijaga agar tetap rendah dengan mengurangi atau menghilangkan dingin yang dihasilkan dengan menggunakan *heat sink*, sedangkan suhu bagian panas dipertahankan sesuai dengan yang diinginkan. (Purwiyanti, *et al.*, 2017)



Gambar 2.1. Modifikasi Sistem Peltier
(Purwiyanti, *et al.*, 2017)

memiliki *port* ADC (*analog input*) sebanyak 16 buah. Gambar 2.3 menunjukkan bentuk dari LM35. LM35 mempunyai 3 pin yang menunjukkan fungsi masing-masing pin diantaranya, pin 1 berfungsi sebagai sumber tegangan kerja dari LM35, pin 2 atau tengah digunakan sebagai tegangan keluaran atau *Vout* dengan jangkauan kerja dari 0 volt sampai dengan 1,5 volt dengan tegangan operasi sensor LM35 yang dapat digunakan antara 4 volt sampai 30 volt (Zennifa, 2012).



Gambar 2.3. LM35
(Zennifa, 2012)

Sensor suhu LM35 komponen elektronika yang memiliki fungsi untuk mengubah besaran suhu menjadi besaran listrik dalam bentuk tegangan. Tegangan sensor ini dapat mencapai 30 volt tetapi yang diberikan ke sensor sebesar 5 volt, sehingga dapat digunakan dengan satu daya tunggal dengan ketentuan bahwa LM35 hanya membutuhkan arus sebesar 60 μA hal ini berarti LM35 mempunyai kemampuan menghasilkan panas (*self-heating*) dari sensor yang dapat menyebabkan kesalahan pembacaan yang rendah yaitu kurang dari 0,5 $^{\circ}\text{C}$ pada suhu 25 $^{\circ}\text{C}$. Keluaran sensor ini akan naik sebesar 10 mV setiap derajat *celcius* sehingga diperoleh persamaan sebagai berikut:

$$V_{\text{LM35}} = \text{Suhu} * 10 \text{ mV} \quad (2.1)$$

Keterangan:

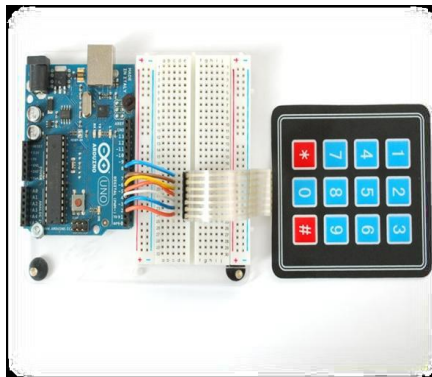
V_{LM35} = Tegangan pada LM35 (mV)

Suhu = Suhu yang terukur ($^{\circ}\text{C}$)

Secara prinsip sensor 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 terserap pada suhu permukaan tersebut.

2.4 *Keypad Membrane*

Keypad membrane pada Gambar 2.4 yang berukuran *keypad* 4×4 , hanya tersusun dari beberapa *push button* yang dikonfigurasi antara kolom dengan baris. *Keypad* ini sering disebut juga *keypad* matriks $n \times m$ (n =kolom m =baris). Kolom dan baris digunakan untuk pendeteksian penekanan tombol.



Gambar 2.4. *Keypad* 4×4 yang dihubungkan ke Arduino Uno (Zennifa, 2012)

Keypad menggunakan sistem matrik di mana kolom dan baris yang sama diserikan satu sama lainnya. Perancangannya menggunakan saklar *Push Button* pada setiap tombolnya, *Push Button* mempunyai tiga masukan yakni untuk kolom, baris, dan *common*. Dengan disetnya *common* dengan *ground*, apabila menekan tombol maka otomatis ketiga masukan terhubung, dengan kata lain kolom dan baris berlogika '0' perubahan logika yang terjadi ini akan diproses oleh mikrokontroler. (Zennifa, 2012)

2.5 Peltier

Prinsip pendinginan *thermoelectric* pada Gambar 2.6 ini ditemukan pertama kali pada tahun 1834 oleh Jean Peltier, sehingga hasil penemuannya ini sering disebut “ Pendingin Peltier ”, apabila ada aliran arus listrik, maka akan disertai dengan panas hasil dari arus tersebut (pemanasan Joule). Jean Peltier mengamati hal ini, bahwa ketika arus listrik melewati pertemuan dua buah konduktor yang berbeda (*thermocouple*), akan ada efek pemanasan yang tidak bisa dijelaskan oleh pemanasan Joule saja. Bahkan tergantung pada arah arus, efeknya bisa berupa pemanasan atau pendinginan.



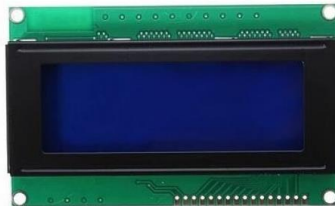
Gambar 2.5. Bentuk *Thermo-Electric* (TEC)

Dalam prakteknya banyak pasangan *thermoelectric* terhubung paralel dan diapit dua buah pelat keramik dalam sebuah *thermoelectric* tunggal. Sedangkan besarnya perbedaan suhu panas dan dingin adalah sebanding dengan arus dan jumlah pasangan semikonduktor di unit. (Santosa, 2015)

2.6 *Liquid Crystal Display*

LCD (*Liquid Crystal Display*) adalah suatu jenis media tampil yang menggunakan kristal cair sebagai penampil utama. LCD sudah digunakan diberbagai bidang misalnya alat-alat elektronik seperti televisi, kalkulator, ataupun layar komputer. Pada proyek akhir yang digunakan kali ini menggunakan aplikasi LCD dot matrik dengan jumlah karakter 20 x 4 yang dipadukan dengan modul I2C pada wiringnya nanti, untuk mempermudah pemasangan serta penghematan pin dan *memory* pada Arduino. Penampilan gambar LCD disajikan pada Gambar 2.6. LCD berfungsi sebagai penampil

yang nantinya akan digunakan untuk menampilkan status kerja alat (Kho, 2017).

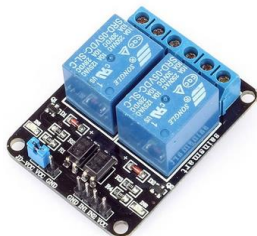


Gambar 2.6. LCD display 20x4

(sumber : <http://buaya-instrument.com/lcd-20x4-blue-screen-display-module-1110000001.html>)

2.7 Relay

Relay adalah Saklar (Switch) yang dioperasikan secara listrik dan merupakan komponen *Electromechanical* (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (*Coil*) dan Mekanikal (seperangkat Kontak Saklar). Relay menggunakan prinsip Elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Sebagai contoh Gambar 2.8 dengan Relay yang menggunakan Elektromagnet 5V dan 50 mA mampu menggerakkan Armature Relay (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A (Kho, *et at.*, 2017).



Gambar 2.7. Modul Relay

(sumber: <https://sunupradana.info/2017/07/15/modul-dua-relay/>)

2.8 Regulator Penurun Tekanan

Jantung dari sistem gas CNG adalah regulator tekanan yang merupakan alat untuk menurunkan tekanan gas CNG dari ± 200 BarG hingga mencapai 1-2 bar seperti yang disajikan pada Gambar 2.8. Sebagian besar material dari regulator tekanan ini adalah aluminium dan kuningan. Regulator tekanan ini terdiri atas tiga element penting yaitu : *spring*, *valve*, dan *membrane*.

Dari hasil dilapangan dan simulasi regulator tekanan, dapat diketahui bahwa regulator tekanan dapat menurunkan tekanan gas CNG dari 200 ± 200 BarG menjadi sekitar maksimal 40 BarG pada tahap pertama, lalu kemudian tahap kedua dari tekanan 40 BarG diturunkan sesuai dengan tekanan yang dibutuhkan oleh konsumen yang berkisar antara 1 - 4 BarG. Perubahan jarak opening pada *valve* level pertama berpengaruh pada perubahan tekanan gas yang melewati regulator tekanan. Semakin besar jarak *opening valve*, maka semakin besar pula tekanan gas yang melewati CNG *pressure* regulator atau semakin kecil *pressure drop*-nya (Ahbab, 2013)



Gambar 2.8. Regulator

2.9 Power Supply

Power Supply adalah alat listrik yang dapat menyediakan energi listrik untuk perangkat listrik ataupun elektronika lainnya. Pada dasarnya *Power Supply* (catu daya) ini memerlukan sumber energi listrik AC dari PLN yang kemudian mengubahnya menjadi energi listrik yang dibutuhkan oleh perangkat elektronika lainnya.

Oleh karena itu, *Power Supply* disebut juga dengan istilah *Electric Power Converter*.



Gambar 2.9. *Power Supply*
(Kho, 2017).

SMPS (*Switch-Mode Power Supply*) seperti pada Gambar 2.9 merupakan jenis *Power Supply* yang langsung menyearahkan (*rectify*) dan menyaring (*filter*) tegangan *Input AC* untuk mendapatkan tegangan *Output DC* yang kemudian tegangan DC tersebut kemudian di *switch* ON dan OFF (Kho, 2017).

BAB III

PERANCANGAN HARDWARE DAN SOFTWARE

Pada bab ini akan membahas mengenai perancangan dan pembuatan “Rancang Bangun Antisipasi *Freezing* Aliran Gas Pada PRS10 Menggunakan Mikrokontroler di PT. PGN Solution”, yang terdiri dari beberapa perancangan yaitu:

1. Perancangan Sistem Kerja Alat.
2. Perancangan Perangkat Keras Mekanik.
3. Perancangan Perangkat Keras Elektrik.
4. Perancangan Perangkat Lunak.

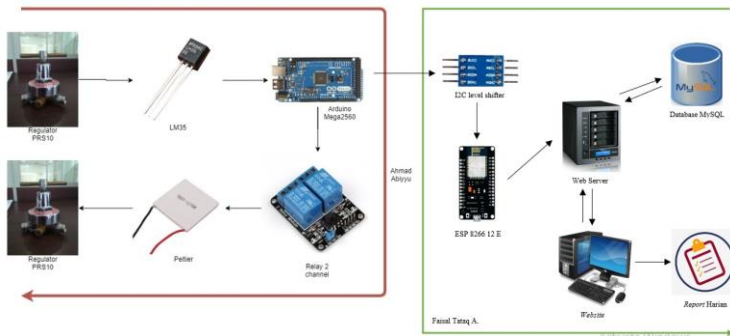
3.1 Diagram Sistem Kerja Alat

Pada perancangan sistem kerja alat, *prototype* alat antisipasi *freezing* pada aliran gas pada PRS10 menggunakan mikrokontroler di PT. PGN Solution akan berisi *power supply*, sensor suhu, Arduino Mega2560, LCD, relay 2 *channel*, peltier, *keypad membrane* 4x4, dan juga regulator. Pada alat ini, kontrol suhu ini difokuskan pada implementasi pengontrolan suhu pada regulator dan penggunaan efek peltier dengan menggunakan Sensor LM35 dan peltier sebagai *thermoelectric* yang digunakan sebagai pemanas untuk mencegah pembekuan dari regulator.

Input sistem ini adalah tekanan yang keluar dari CNG *cradle* (Tabung Gas Alam bertekanan ± 200 BarG). Gas bertekanan tinggi tersebut diturunkan dalam dua tahap dengan menggunakan dua regulator yang menjadi satu kesatuan dalam PRS. Tahap pertama dari tekanan 200 BarG diturunkan menjadi maksimal 40 BarG, kemudian tahap kedua dari tekanan 40 BarG diturunkan sesuai dengan tekanan yang dibutuhkan oleh konsumen yang berkisar antara 1- 4 BarG dengan kebutuhan mesin/alat produksi. Hasil Perbedaan penurunan tekanan yang besar dari keluaran CNG *Cradle* ketekanan sesuai dengan permintaan pelanggan dan laju alir gas mendekati batas maksimal kemampuan regulator dapat menimbulkan terjadinya bunga es pada regulator tersebut. Sebelum terjadi pembekuan, relay akan menyala setelah suhu terendah untuk mengaktifkan relay diatur terlebih dahulu menggunakan *keypad*. Suhu pengaturan untuk mengkatifkan relay akan mengirimkannya ke mikrokontroler Arduino Atmega 2560 untuk diproses dan dibaca bersama suhu yang terdapat pada regulator. Data hasil pembacaan

sensor suhu serta pengaturan suhu akan ditampilkan juga di LCD 20x4. Ketika pertama kali masuk (*login*) menggunakan kata sandi.

Data hasil pembacaan sensor suhu jika sesuai dengan pengaturan yang ditetapkan kemudian akan menuju relay. Setelah dari relay kemudian akan menyalakan *power supply* untuk memberikan tegangan kepada peltier agar menyala. Peltier akan mengaktifkan efeknya, yaitu efek sisi panas yang akan membuang dingin dan menjaga suhu tetap di atas dari titik beku yang ditetapkan atau diatur oleh *keypad*. Untuk cara kerja sistem alat ditunjukkan pada garis merah Gambar 3.1. Alogaritma sistem secara keseluruhan pembuatan alat yang digunakan dapat diperlihatkan pada Gambar 3.26.



Gambar 3.1. Diagram Fungsional

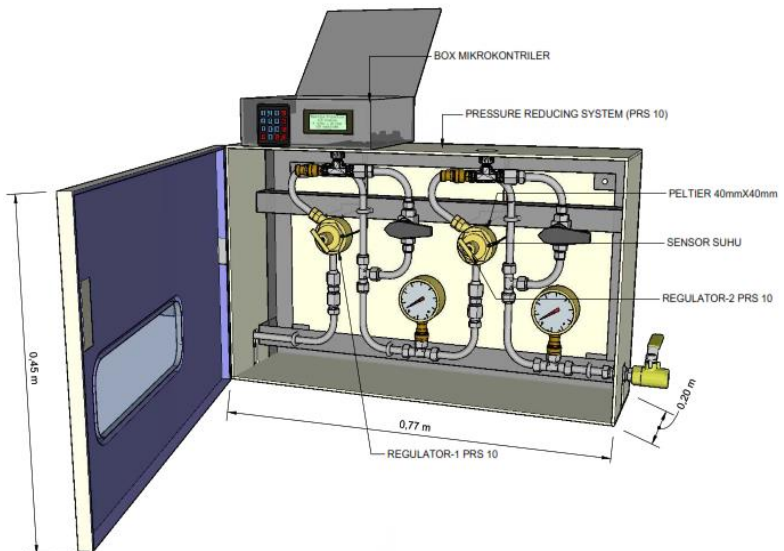
Dalam Gambar 3.1 yang bergaris merah merupakan diagram fungsional pada sistem kerja alat Antisipasi Freezing Aliran Gas Pada PRS10 Menggunakan Mikrokontroler.

3.2 Perancangan Perangkat Keras Mekanik

Pada perancangan perangkat keras mekanik ini terbagi menjadi dua bagian, yaitu perancangan pada bagian kontrol yang bergaris tanda warna merah dan perancangan monitoring yang bergaris tanda warna hijau. Pada proyek akhir ini yang dibahas adalah mengenai kontrol yang ditandai oleh garis merah

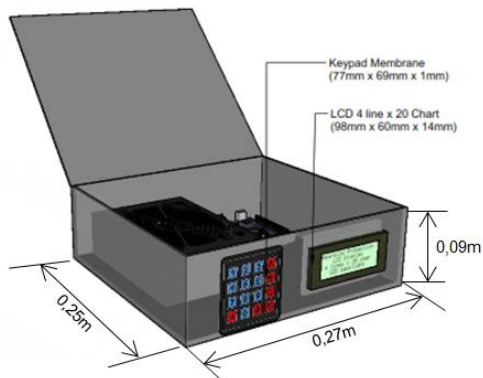
Pada perancangan kontrol merupakan bagian dari komponen penyusun dari proyek akhir ini, antara lain Arduino Mega2560 karena memiliki 54pin digital dan 15pin analog, dan dua sensor suhu

LM35, dan relay terhadap sensor suhu. Seluruh komponen tersebut diletakkan pada kota box akrilik seperti pada Gambar 3.2 untuk tampak keseluruhan dan Gambar 3.3 untuk tampak depan, Gambar 3.4 untuk tampak tampak atas, Gambar 3.5 untuk tampak belakang, Gambar 3.6 pemasangan peltier pada regulator PRS10, Gambar 3.7 letak sensor suhu.



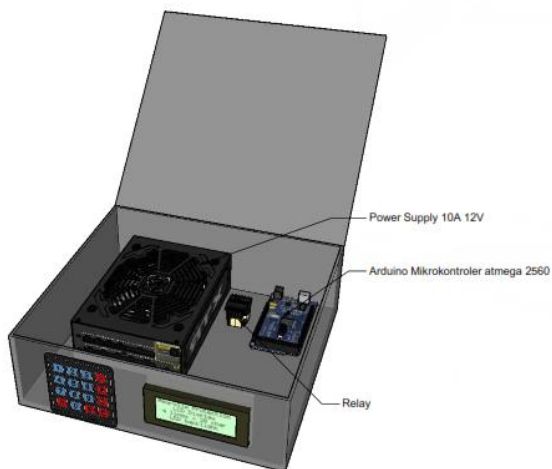
Gambar 3.2. Perancangan Alat Tampak Keseluruhan

Pada Gambar 3.2 merupakan gambar keseluruhan atau gambar penempatan alat yang dibuat pada PRS10. Alat yang dibuat ditaruh di atas PRS10 kemudian sensor suhu dan peltier akan di tempelkan pada regulator seperti yang disajikan oleh Gambar 3.6 dan Gambar 3.7. Kedua sensor suhu yang dipasangan akan mendeteksi perubahan suhu yang terjadi pada kedua regulator PRS10 yang akan di tampilkan pada LCD seperti perancangan tampak depan pada Gambar 3.3, kemudian keluaran sensor suhu itu akan menjadi pembanding untuk menyalakan pemanas regulato atau juga bisa membuat setting pointnya untuk tambahan mematikan atau menyalakan sesuai dengan suhu yang diinginkan.



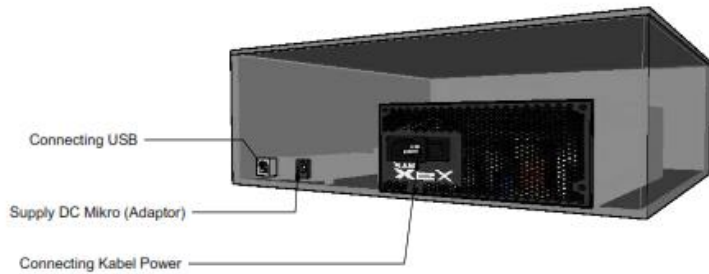
Gambar 3.3. Perancangan Alat Tampak Depan

Pada Gambar 3.3 adalah gambar rancangan alat tampak depan. Pada alat ini bagian depan akan dipasangkan *keypad* dan LCD untuk menampilkan suhunya dan membuat *setting point* untuk mengatur pemanas.



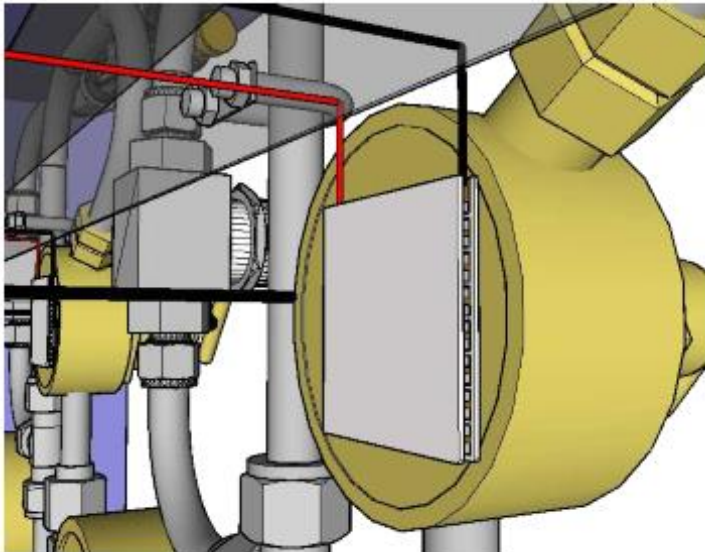
Gambar 3.4. Perancangan Alat Tampak Atas

Pada Gambar 3.4 memperlihatkan bagian dalam dari rancangan yang berisi *power supply*, Arduino, dan relay 2 channel.

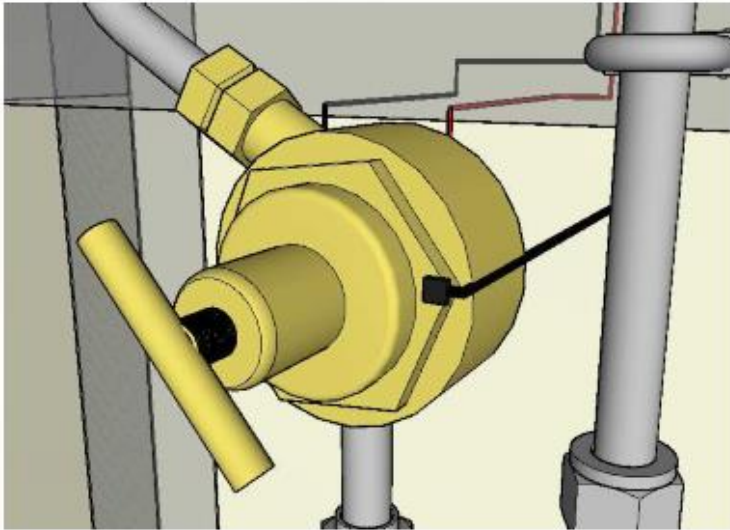


Gambar 3.5. Pemasangan Alat Tampak Belakang

Pada Gambar 3.5. adalah perancangan alat bagian belakang di mana pada bagian belakang merupakan jalur keluarnya kabel-kabel seperti kabel *connector USB*, kabel *power*, dan juga kabel keluaran kedua sensor suhu dan pemanas.



Gambar 3.6. Pemasangan Peltier pada Regulator PRS10



Gambar 3.7. Letak Sensor Suhu LM35

Pada Gambar 3.7 dan Gambar 3.8 merupakan gambar perancangan pemasangan sensor suhu dan peltier yang ditempatkan pada regulator.

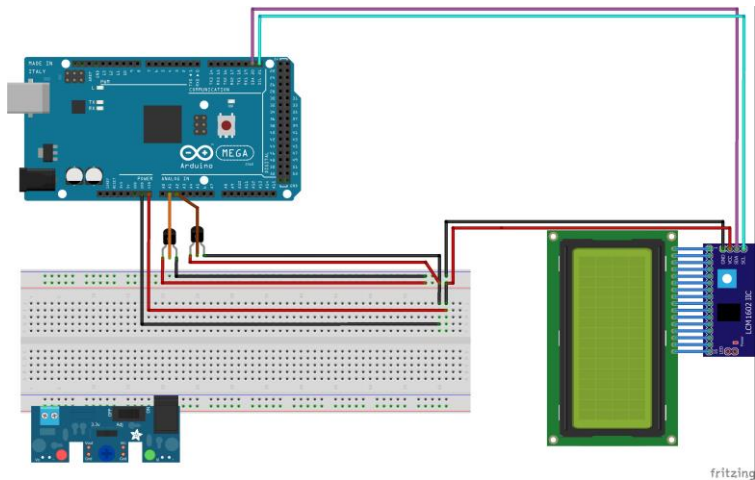
3.3 Perancangan Perangkat Keras Elektrik

Pada perancangan perangkat elektrik (*software*) pada Proyek Akhir ini meliputi perancangan rangkaian sensor suhu LM35, perancangan rangkaian LCD, perancangan rangkaian relay dan perancangan keypad.

3.3.1 Perancangan Sensor Suhu LM35 dengan LCD yang dihubungkan dengan Arduino

Merupakan chip IC produksi dari *National Semiconductor* yang berfungsi untuk mengukur suhu pada suatu objek atau ruangan dengan keluaran dalam bentuk besaran elektrik (tegangan analog). Sensor ini termasuk sensor suhu analog di karenakan *output*-nya yang berupa tegangan analog. Bentuk fisik sensor suhu LM35 menyerupai transistor yang mempunyai 3 kaki yang terdiri dari pin kaki *input* tegangan positif, *output*, dan *ground*. Perancangan kedua

sensor suhu ini nantinya akan menampilkan keluaran suhu yang terukur pada kedua regulator PRS 10. Perancangan sensor suhu LM35 dengan LCD dapat dilihat pada penyajian Gambar 3.8.



Gambar 3.8. Wiring Sensor LM35 Dengan LCD yang dihubungkan Dengan Arduino

Pada Gambar 3.8 merupakan perancangan sensor LM35 pada Arduino ini terlihat pin Vin dengan kabel yang berwarna merah dan Vout kabel berwarna biru, serta *ground* yang berwarna hitam. Kaki pin tengah sensor suhu LM35 sebagai Vin yang berfungsi untuk *input* sensor yang akan disambungkan ke *power supply*. Sensor suhu LM35 memiliki 3 pin, yaitu pin Vin, pin *input*, GND. Vin Arduino disambungkan pada Vin LM35 yang berwarna merah, kemudian GND LM35 disambungkan ke *ground* Arduino yang berwarna hitam sedangkan dua *input* LM35 disambungkan ke pin analog A1 dan A2 Arduino. Tegangan yang dibutuhkan sensor LM35 dari Arduino adalah sebesar 5VDC. Kelauran pembacaan kedua sensor ini akan ditampilkan lewat LCD dengan menggunakan modul I2C untuk mengurangi pin-pin pada LCD, sehingga keluaran dari modul I2C hanya membutuhkan 4 pin untuk menampilkannya. Penggunaan modul I2C ini untuk mempermudah dalam proses *wiring* dan menghemat *memory* Arduino serta mempermudah untuk memprogram

menampilkannya agar mudah terbaca. Untuk program menampilkan suhu dapat dilihat pada tampilan program Gambar 3.9.

```
#include <Keypad.h>
#include <Wire.h> //untuk mengaktifkan fungsi library wire
#include <LiquidCrystal_I2C.h> //untuk mengaktifkan lcd i2c

LiquidCrystal_I2C lcd(0x27, 20, 4);

float tempC;
float tempC2;

int tempPin = A1; //analog A1
int tempPin2 = A2; //analog A2

void setup()
{
    lcd.begin();

    pinMode (A1, INPUT);
    pinMode (A2, INPUT);
}

void loop()
{
    if (count == 20000)
    {
        tempPin = analogRead(A1);
        tempC = tempPin * 0.488;
        tempPin2 = analogRead(A2);
        tempC2 = tempPin2 * 0.488;

        lcd.clear();

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

Gambar 3.9. Alogaritma Program pada Software Arduino Penginisialisasian 2 Sensor Suhu LM35

```
        lcd.print ("suhu1=");
        lcd.print(tempC);
        lcd.setCursor (0, 1);
        lcd.print ("set point 1 :");
        lcd.print(setpoint1);

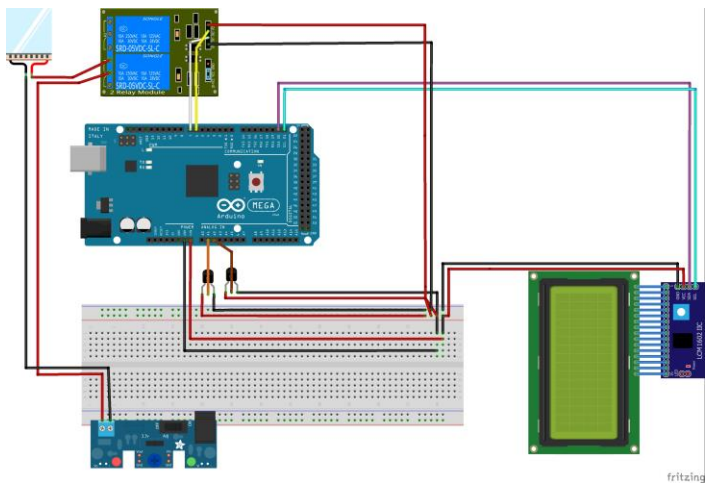
        lcd.setCursor (0, 2);
        lcd.print ("suhu2=");
        lcd.print(tempC2);
        lcd.setCursor (0, 3);
        lcd.print ("set point2 :");
        lcd.print(setpoint2);
    }
}
```

Gambar 3.10. Alogaritma Program pada Software Arduino Manampilkan Suhu pada LM35

Pada Gambar 3.10 merupakan program untuk menampilkan suhu pada LCD di mana suhu yang ditampilkan berupa suhu1 pada kolom 0 baris 0, *setting point* untuk suhu1 dikolom 0 baris ke 1, suhu2 dikolom 0 baris ke 2 dan yang terakhir tata letak *setting point2* dikolom 0 baris ke 3.

3.3.2 Perancangan Relay 2 Channel Terhadap Suhu dengan Arduino Mega

Rangkaian Peltier yang dikoneksikan dengan modul relay 2 *channel* skema konfigurasinya yaitu pin 5V Arduino dihubungkan dengan Vin relay dan GND dengan *ground* Arduino, COM relay 1 dan COM relay 2 di hubungkan ke *power supply*, pin out NO (*Normally Open*) relay 1 dan relay 2 dihubungkan dengan VCC kedua peltier, pin GND Arduino dihubungkan dengan GND Relay dan pin 7 digital Arduino dihubungkan dengan IN1 relay serta pin 8 *digital* Arduino dihubungkan dengan pin IN2 relay. *Wiring* dilihat pada penyajian Gambar 3.11. dan Programnya dapat dilihat pada penyajian Gambar 3.12, Gambar 3.13, Gambar 3.14, Gambar 3.15, Gambar 3.16



Gambar 3.11.Wiring Modul Relay dengan Peltier yang dihubungkan dengan Arduino

```

#include <Keypad.h>
#include <Wire.h> //untuk mengaktifkan fungsi library wire
#include <LiquidCrystal_I2C.h> //untuk mengaktifkan lcd i2c

LiquidCrystal_I2C lcd(0x27, 20, 4);

float tempC;
float tempC2;

int relay1 = 7;
int relay2 = 6;

int tempPin = A1; //analog A1
int tempPin2 = A2; //analog A2

void setup()
{
  lcd.begin();

  pinMode (relay1, OUTPUT);
  pinMode (relay2, OUTPUT);

  pinMode (A1, INPUT);
  pinMode (A2, INPUT);
}

void suhu()
{
  if (state == 1)
  {
    if (25 < tempC < 35 && setpoint1 < 34 && setpoint1 >= 25)
    {
      digitalWrite (relay1, HIGH);
    }
  }

```

Gambar 3.12. Algoritma Program pada Software Arduino Kondisi Inisialisasi relay dan sensor suhu LM35

```

      digitalWrite (relay1, HIGH);
      digitalWrite (ledmerah, HIGH);
      digitalWrite (ledhijau, LOW);
    }
    if (10 >= tempC > 0 && 0 <= setpoint1 && setpoint1 < 10)
    {
      digitalWrite (relay1, LOW);
      digitalWrite (ledhijau, HIGH);
      digitalWrite (ledmerah, LOW);
    }
    if (25 < tempC2 < 35 && setpoint2 < 34 && setpoint2 >= 25)
    {
      digitalWrite (relay2, HIGH);
      digitalWrite (ledhijaul, LOW);
      digitalWrite (ledmerah1, HIGH);
    }
    if (10 >= tempC2 > 0 && 0 <= setpoint2 && setpoint2 < 10)
    {
      digitalWrite (relay2, LOW);
      digitalWrite (ledhijaul, HIGH);
      digitalWrite (ledmerah1, LOW);
    }
  }

  set1();
  set2();

  if (count == 20000)
  {
    tempPin = analogRead(A1);
    tempC = tempPin * 0.488;
    tempPin2 = analogRead(A2);
    tempC2 = tempPin2 * 0.488;
  }

```

Gambar 3.13. Algoritma Program pada Software Arduino Kondisi untuk Mengaktifkan Relay dan Mematikan Relay

```

tempC2 = tempPin2 * 0.488;

lcd.clear();

lcd.setCursor (0, 0);
lcd.print ("suhu1=");
lcd.print(tempC);
lcd.setCursor (0, 1);
lcd.print ("set point 1 :");
lcd.print(setpoint1);

lcd.setCursor (0, 2);
lcd.print ("suhu2=");
lcd.print(tempC2);
lcd.setCursor (0, 3);
lcd.print ("set point2 :");
lcd.print(setpoint2);
}
}

if ( state == 2)
{
    if (count == 20000)
    {
        lcd.setCursor (0, 0);
        lcd.print ("Isi Set Point 1:");
    }
}

if ( state == 3)
{

    if (count == 20000)
    {

```

Gambar 3.14. Alogaritma Program pada Software Arduino Kondisi untuk mengisi *Set Point 1*

```
if (count == 20000)
{
    lcd.setCursor (0, 0);
    lcd.print ("Isi Set Point 2:");
}
}
if (customKey && state == 2)
{
    lcd.setCursor (j, 1);
    lcd.print(customKey);
    j++;
    if (isDigit(customKey)) //pengkondisian konversi char ke keypad saat awal
    {
        inString += customKey; //pengubahan char ke string
    }
    if (isAlpha(customKey))
    {
        inString += customKey;
    }
    if (customKey == '*')
    {
        int nilai = inString.toInt();
        setpoint1 = nilai;
        state = 1;
        j = 0;

        lcd.clear();
        lcd.setCursor (0, 2);
        lcd.print ("OK...");
        delay(500);
        lcd.clear();
        inString = " ";
    }
}
```

Gambar 3.15. Alogaritma Program pada Software Arduino Kondisi untuk mengisi *Set Point 2*

```

        inString = " ";
    }
}

if (customKey == state == 3)
{
    lcd.setCursor(j, 1);
    lcd.print(customKey);
    j++;
    if (isDigit(customKey)) //pengkondisian konversi char ke keypad saat awal
    {
        inString += customKey; //pengubahan char ke string
    }
    if (isAlpha(customKey))
    {
        inString += customKey;
    }
    if (customKey == '*')
    {
        int nilai = inString.toInt();
        setpoint2 = nilai;
        state = 1;
        j = 0;
        lcd.clear();
        lcd.setCursor(0, 2);
        lcd.print("OK...");
        delay(500);
        lcd.clear();
        inString = " ";
    }
}
}
}

```

Gambar 3.16. Alogaritma Program pada Software Arduino Kondisi Setelah Set Point di Set

Gambar 3.11 merupakan rangkaian relay yang dihubungkan dengan peltier. Kedua kaki pin IN1 kabel berwarna putih dan IN2 kabel berwarna kuning relay sendiri dihubungkan dengan kaki pin 7 dan 6 pin *digital* Arduino Mega2560. Kemudian Kaki pin VCC kabel yang berwarna merah relay dihubungkan dengan kaki pin Vin Arduino dan kaki GND relay yang berwarna hitam dihubungkan dengan GND Arduino.

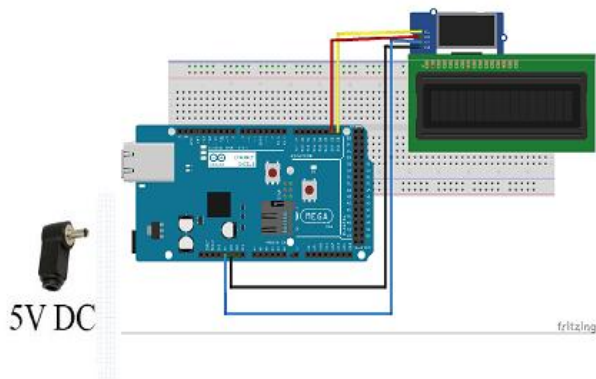
Pada Gambar 3.12 merupakan gambar untuk menginsialisasikan sensor suhu beserta relay, Selanjutnya pada Gambar 3.12 merupakan program untuk mengatur keluaran relay sebagai *output* dan sensor suhu sebagai *input*. Program ini dimaksudkan untuk mengatur suhu sebagai masuknya untuk menyalakan relay yang berfungsi sebagai *output*. Kemudian pada Gambar 3.13 adalah kondisi untuk menyalakan relay yang diatur berdasarkan *set point* yang diatur dan berdasarkan kondisi suhu dari

regulator. Jika sudah semua sudah memenuhi kondisi untuk menyala, maka relay akan menyala, begitupun sebaliknya apabila relay sudah memenuhi kondisi *off*, maka relay akan mati.

Untuk Gambar 3.14 dan Gambar 3.15 merupakan program untuk mengisi dan menampilkan *set point1* dan *set point2*, apa bila kondisi *set poin1* dan *set point2* sudah terisi maka pada Gambar 3.16 program akan menampilkan tampilan “OK...!”.

3.3.3 Perancangan LCD yang dihubungkan dengan Arduino Mega

Pada Proyek Akhir ini, LCD berfungsi sebagai penampilan data suhu yang telah dideteksi oleh sensor. LCD yang digunakan yaitu LCD berukuran 20x4. LCD dapat menampilkan data jika disambungkan dengan kontroler. Kontroler yang digunakan pada Proyek Akhir ini yaitu Arduino Mega 2560. Untuk menyambungkan LCD dengan Arduino, yaitu bisa dengan modul I2C *backpack*. Kaki yang digunakan pada I2C adalah kaki pin SDL dan SDA yang dihubungkan dengan kaki pin SDA dan SDL Arduino Mega2560. Perancangan dan program dan dapat dilihat pada Gambar 3.17 dan Gambar 3.18.



Gambar 3.17. Wiring LCD 20x4 dengan Modul I2C dengan Arudino

```

#include <Wire.h> //untuk mengaktifkan fungsi library wire
#include <LiquidCrystal_I2C.h> //untuk mengaktifkan lcd i2c

LiquidCrystal_I2C lcd(0x27, 20, 4);

void setup()
{
  lcd.begin();
}

void loop()
{
  lcd.setCursor (0, 0);
  lcd.print ("Masukkan Password:");
}

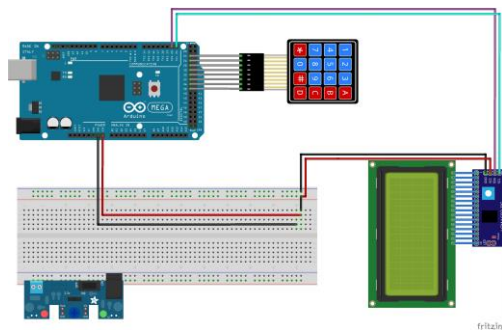
```

Gambar 3.18. Alogaritma Program pada Software Arduino Program Menampilkan LCD

Pada Gambar 3.17 merupakan perancangan LCD dengan isi program untuk menampilkan tulisan “Masukkan *Password:*” dan pada Gambar 3.18 merupakan isi program yang dibuat pada *software* Arduino IDE.

3.3.4 Perancangan yang dihubungkan dengan Arduino Mega

Keypad digunakan sebagai *input* data berupa angka yang nantinya akan digunakan sebagai *input password* dan *setting point*. Penekanan *keypad* dapat disimulasikan dengan penyesuaian tombol *keypad* dan data yang muncul. Gambar rangkaian perancangan *keypad* yang dihubungkan dengan kaki pin digital 24, 26, 28, 30, 32, 34, 36, 38 Arduino Mega2560 dapat dilihat pada penyajian Gambar 3.19.



Gambar 3.19. Wiring *Keypad* yang dilengkapi LCD dengan Arduino

```

#include <Keypad.h>
#include <Wire.h> //untuk mengaktifkan fungsi library wire
#include <LiquidCrystal_I2C.h> //untuk mengaktifkan lcd i2c

LiquidCrystal_I2C lcd(0x27, 20, 4);

int j = 0 ;
int state = 0, kondisi = 0;

const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
//define the cymbols on the buttons of the keypad
char hexaKeys[ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
byte rowPins[ROWS] = {24, 26, 28, 30}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {32, 34, 36, 38}; //connect to the column pinouts of the keypad

//initialize an instance of class NewKeypad
Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

void setup()
{
  lcd.begin();
}

```

Gambar 3.20. Alogaritma Program pada Software Arduino Program Penginisialisasian *Keypad* Kolom dan Baris

Pada Gambar 3.20 merupakan penginisialisasian program kolom dan baris untuk menentukan tombol *input* berupa angka yang nantinya akan digunakan *password* dan untuk *setting point* mengaktifkan relay beserta pemanasnya. Dalam char hexaKeys isi kolom dan baris dapat di isi sesuai dengan kebutuhan Pada Kolom 1 baris 1, angka '1' bisa digantikan huruf atau angka yang digunakan sesuai dengan kebutuhan. Contoh untuk pengaturan angka '1' dalam char bisa diganti dengan huruf 'I' atau huruf lainnya dan dengan angkapun boleh dilakukan. Untuk menampilkan angka dapat dilihat pada Gambar 3.21

```

void loop()
{
    customKey = customKeypad.getKey();
    if (count == 50000)
    {
        count = 0;
    }
    count++;
    suhu();
    if (state == 0)
    {
        if (count == 20000)
        {
            lcd.setCursor (0, 0);
            lcd.print ("Masukkan Password:");
        }

    }
    if (customKey == '#')
    {
        lcd.clear();
        lcd.setCursor (0, 0);
        lcd.print ("Resetting...");
        lcd.clear();
        delay(500);
        j = 0;
        inString = " ";
        state = 0;
        kondisi = 0;
        customKey = false;
    }
}

```

Gambar 3.21. Algoritma Program pada Software Arduino Untuk menampilkan Angka

```

if (customKey && state == 0)
{
    lcd.setCursor (j, 1);
    lcd.print(customKey);
    j++;

    if (isDigit(customKey)) //pengkondisian konversi char ke keypad saat awal
    {
        inString += customKey; //pengubahan char ke string
    }
    if (isAlpha(customKey))
    {
        inString += customKey;
    }

    if (customKey == '+')
    {
        int nilai = inString.toInt();
        if (nilai == 1234 && kondisi == 0)
        {
            customKey = false;
            state = 1;
            j = 0;
            inString = " ";
            lcd.clear();
            lcd.setCursor (0, 2);
            lcd.print ("Completed");
            delay(500);
            lcd.clear();
        }

        if (nilai == 1234 && kondisi == 1)

```

Gambar 3.22. Algoritma Program pada Software Arduino Membuat Password

```

{
    customKey = false;
    state = 2;
    kondisi = 0;
    j = 0;
    inString = " ";
    lcd.clear();
    lcd.setCursor (0, 2);
    lcd.print ("Completed");
    delay(500);
    lcd.clear();
}
if (nilai == 1234 && kondisi == 2)
{
    customKey = false;
    state = 3;
    kondisi = 0;
    j = 0;
    inString = " ";
    lcd.clear();
    lcd.setCursor (0, 2);
    lcd.print ("Completed");
    delay(500);
    lcd.clear();
}
if (nilai != 1234)
{
    customKey = false;
    j = 0;
    inString = " ";
    lcd.clear();
    lcd.setCursor (0, 2);

```

Gambar 3.23. Alogaritma Program pada Software Arduino Pengkondisian ketika *password* benar dan salah

```

        lcd.print ("Access Denied");
        delay(500);
        lcd.clear();
    }
}
}
}

```

Gambar 3.24. Alogaritma Program pada Software Arduino Kondisi jika *password* salah

Pada Gambar 3.21 merupakan program pengulangan untuk menampilkan angka, yaitu “customKey = customKeypad.getKey()”. Sedangkan pada Gambar 3.22 adalah gambar untuk membuat

password yang diinginkan, yaitu “nilai =1234” dan program ketika benar maka akan menampilkan kata “Completed” seperti pada Gambar 3.23 dan ketika salah maka akan menampilkan “Access Denied” pada program Gambar 3.24. Untuk penampilan program secara keseluruhan dapat dilihat pada bagian Lampiran A.

3.4 Perancangan Software

Dalam perancangan perangkat *software* pada Proyek Akhir ini yang dibahas terdiri dari pemrograman pengukuran suhu regulator dengan menggunakan sensor LM35 pada *software* Arduino IDE.

3.4.1 Pemrograman Pembacaan Sensor

Pada perancangan program *software* Arduino dengan fungsi tersebut dibutuhkan beberapa alur kerja dari pembacaan sensor sensor suhu LM35: Untuk memprogram Arduino juga harus dilakukan beberapa tahapan sebagai berikut :

1. **Setting Board Arduino.** Dalam pemrograman *software* Arduino harus di *setting* terlebih dahulu *board* Arduino agar penggunaan Arduino cocok. Dalam purwarupa kali ini Arduino menggunakan Arduino Mega. Untuk *setting board* Arduino bisa masuk ke *tools – board* – setelah itu pilihlah *board* Arduino yang sesuai.
2. **Setting Serial.** *Serial* ini merupakan kabel Arduino yang dihubungkan kepada komputer atau laptop. *Serial* ini mempunyai dua fungsi yang bisa digunakan. Pertama *serial port* digunakan untuk mengunduh program dari Arduino yang kedua *serial* digunakan sebagai komunikasi *serial* pada Arduino dengan komputer. *Setting serial* bisa masuk *tools – serial* - lalu pilih COM yang terbaca pada laptop atau komputer yang sesuai dengan Arduino yang terpasang dan dapat dilihat pada Gambar 3.25.
3. Apabila *program* tidak dapat diunduh karena *serial port*, maka cek terlebih dahulu *serial* yang benar pada *device manager*. Lalu dalam *software* Arduino untuk memilih *serial port*nya samakan dengan *serial port* untuk Arduino dalam *device manager* tersebut. Untuk

masuk ke *device manager* dapat masuk *start windows* – lalu ketika *device manager* klik dua kali dan masuk ke COM.



Gambar 3.25. *Setting Port pada Software Arduino IDE*

Pada penelitian ini *software* yang digunakan adalah program Arduino untuk membuat dan merencanakan program dalam bahasa C. Pemrograman *software* Arduino dirancang dengan menggunakan *software* yang bernama Arduino IDE dengan menggunakan bahasa pemrograman C. Arduino sangatlah berbeda sekali dengan ATmega lainnya. Arduino merupakan sebuah kit mikro-kontroler AVR yang dibuat dalam sebuah *board* (papan PCB). Dikembangkan di Italia sejak tahun 2005. Dalam 1 *board* sudah terdapat mikrokontroler lengkap dengan *pin/port* untuk koneksi serta sudah dilengkapi dengan *downloader*. Dalam segi bahasa pemrograman, Arduino memiliki bahasa pemrograman yang lebih mudah dan sederhana terutama bagi pemula. Alasan bahasa pemrograman Arduino lebih mudah dan sederhana adalah karena didalam Arduino sudah terdapat beberapa *library* yang dapat digunakan untuk merancang pemrograman yang diinginkan. Berikut ini merupakan program *pseudocode* yang digunakan pada penelitian ini. *Pseudocode* tersebut terdapat pada Gambar 3.26 yang menjelaskan tentang pembacaan sensor dengan menggunakan Arduino hingga dapat menyalakan peltier.

```

pseeucode
read keypad, sensor suhu1 and sensor suhu2, relay1 and relay2
if (password right), then print "completed"
else (password wrong), then print "acces denied"
end if
read input sensor suhu LM1 dan suhu LM2 and output relay1 dan relay2
suhu1= pinSuhu1 * 0,488 dan suhu2= pinSuhu2 * 0,488
input set point1 or input set point2
if (suhu1 25 degree celsius into 50 degree celsius and set point1 25 into 35),then relay1 OFF
else if (suhu1 25 degree celsius into 50 degree celsius), then relay1 OFF
end if
if (suhu1 0 degree celsius into 10 degree celsius and set point1 0 into 10),then relay1 ON
else if (suhu1 0degree celsius into 10 degree celsius), then relay1 ON
end if
if (suhu1 25 degree celsius into 50 degree celsius and set point2 25 into 50),then relay2 OFF
else if (suhu1 25 degree celsius into 50 degree celsius), then relay2 OFF
end if
if (suhu1 0 degree celsius into 10 degree celsius and set point2 0 into 10),then relay2 ON
else if (suhu1 0degree celsius into 10 degree celsius), then relay2 ON
end if

```

Gambar 3.26. Pseudocode

Pada Gambar 3.26 merupakan algoritma keseluruhan dan *pseudocode* dari keseluruhan program. *Pseudocode* akan membaca penginisialisasian sensor suhu1 dan sensor suhu2 beserta relay1 dan relay2 dan juga *keypad*. Selanjutnya jika *password* yang dimasukan adalah benar maka akan menampilkan tulisan “Completed” dan lanjut pembacaan sensor suhu1 dan 2 sebagai *input* dan relay1 dan relay2 sebagai output. Jika *password* yang dimasukan adalah salah, maka akan menampilkan tulisan “Access Denied” dan *user* akan diminta memasukan *password* sampai benar. Berikutnya ketika suhu sudah terbaca dan ditampilkan, user diminta untuk mengisikan set point1 dan set point2 untuk mengaktifkan relay1 dan relay2. Apabila suhu1 adalah 25°C sampai dengan 35°C, kondisi relay1 adalah OFF dan ketika suhunya berkisar antara 0°C-10°C, kondisi relay1 ON. Selanjutnya apabila suhu suhu2 adalah 25°C sampai dengan 35°C, kondisi relay2 adalah OFF dan ketika suhunya berkisar antara 0°C-10°C, kondisi relay2. ON.

----Halaman ini sengaja dikosongkan-----

BAB IV

PENGUJIAN DAN ANALISA

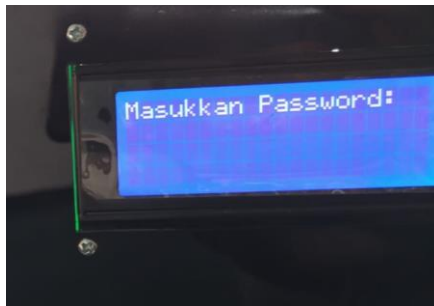
Dalam membuat suatu sistem, pengujian dan analisa sangat diperlukan. Pengujian dan analisa bertujuan untuk mengetahui sistem bekerja sesuai rencana atau tidak. Dari hasil pengujian dan analisa dapat diketahui kelemahan-kelemahan dari suatu sistem, sehingga dapat dilakukan perbaikan, pengembangan, dan penyempurnaan sistem.

Pada bab ini dibahas pengujian serta analisa dari perancangan sistem yang telah dibuat pada proyek akhir. Pengujian dilakukan secara parsial terlebih dahulu, tujuannya adalah untuk mengetahui kinerja *hardware* setiap unit. Adapun beberapa pengujian yang telah dilakukan antara lain:

1. Pengujian LCD.
2. Pengujian *Password* dengan menggunakan *keypad*.
3. Sensor Suhu LM35 yang dihubungkan dengan Arduino Mega2560.
4. Pengujian Relay 2 *channel* terhadap suhu.

4.1 Pegunjian LCD

Pengujian LCD dilakukan agar dapat mengetahui kelayakan LCD tersebut dapat digunakan. Pada Gambar 4.1 dapat dilihat bahwa LCD dapat menampilkan sesuai dengan program.

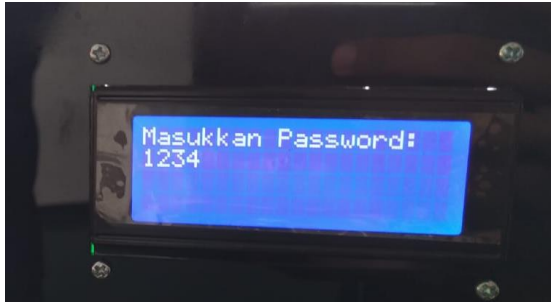


Gambar 4.1. Pengujian LCD

Dari hasil Gambar 4.1 dapat dilihat tampilan awal proses pada LCD terdapat kata “Masukan Password” yang sudah di program adalah sesuai.

4.2 Pengujian *Password* dengan Menggunakan *Keypad*

Pengujian *Keypad* dilakukan untuk membandingkan apakah hasil dari penekanan *Keypad* sama dengan hasil yang keluar pada tampilan LCD.



Gambar 4.2. Pengujian Tombol *Keypad* dan *Password*

Dari Gambar 4.2 merupakan gambar pengujian tombol untuk memasukan *password*.



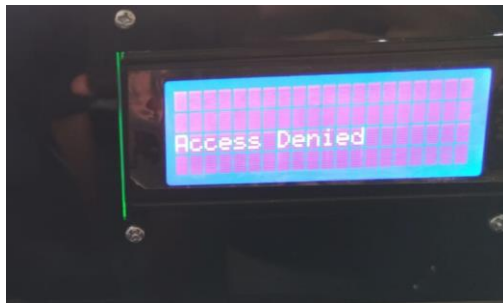
Gambar 4.3. Tampilan *Password* Benar

Pada Gambar 4.3 merupakan tampilan gambar apabila *password* yang dimasukan benar, maka LCD akan menampilkan tulisan "Completed".



Gambar 4.4. Pengujian *Password* yang Tidak Sesuai

Pada Gambar 4.4 merupakan gambar apabila *password* yang di masukan tidak sesuai dengan *password* yang sudah di program.



Gambar 4.5. Tampilan Jika *Password* Salah

Sedangkan dalam Gambar 4.5 merupakan tampilan gambar apabila *password* yang dimasukan salah, maka LCD akan menampilkan tulisan “Access Denied”.

4.3 Sensor Suhu LM35 yang di hubungkan dengan Arduino Mega

Tujuan dari pengukuran dan pengujian Sensor Suhu LM35 adalah untuk pengujian pemanas dan kalibrasi sensor suhu LM35 seperti yang disajikan pada Tabel 4.1, Tabel 4.2, Tabel 4.3 untuk mengetahui apakah sensor berfungsi dengan baik sesuai dengan

standar maka dilakukan kalibrasi dan sensor digunakan sudah sesuai dengan standar perubahan suhu dari regulator dilari gas bertekanan tinggi yang dapat mempengaruhi keadaan dari regulator PRS10.

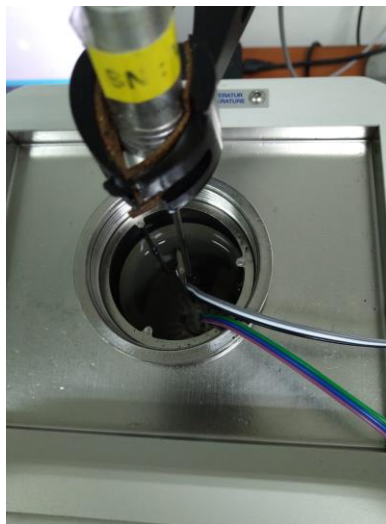
Cara pengambilan data yang digunakan adalah memprogram Arduino menggunakan *stopwatch* sebagai pengukur waktu yang pasti kalibrasinya, sedangkan untuk pengambilan data suhu adalah 2 menit sekali, serta regulator yang digunakan memiliki spesifikasi seperti tekanan gas *cradle* adalah 125bar , regulator 1 20bar, dan regulator 2 adalah 1,3bar. Dalam pengukuran ini dilakukan pada waktu pukul 19.00WIB sampai 20.00WIB, seperti pada penyajian Tabel 4.4 dan Tabel 4.5. Hal ini dilakukan untuk mengetahui parameter yang mempengaruhi antisipasi pembekuan pada regulator PRS10. Pengujian kalibrasi sensor digunakan kalibrator standar yang sudah terkalibrasi, yaitu:

Merek : WIKA

Type/Model: CTH-6500

No.Seri: 750141071543

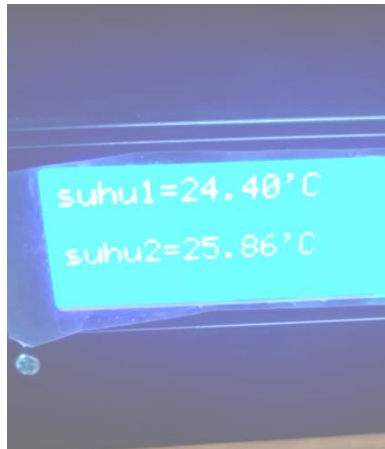
Satuan: °C



Gambar 4.6. Pengujian Kalibrasi



Gambar 4.7. Hasil Pada Kalibrator



Gambar 4.8. Hasil Pada Alat

Pada Gambar 4.6 merupakan pengujian kalibrasi menggunakan *microbath* alat untuk mengkalibrasi dan menjaga suhu tetap konstan pada saat kalibrasi dan untuk mengatur pada span berapa sensor akan dikalibrasi. Dalam *microbath* diberikan cairan air

silicon untuk megaduk air agar tidak membeku saat diberikan suhu rendah dan tidak terlalu panas saat suhunya dinaikan tinggi. Dalam bak *microbath* di dalamnya terdapat sensor suhu PT 100 sebagai kalibrator dan dua sensor suhu LM35 yang mau dikalibrasi. Pada kalibrasi yang dilakukan ini menggunakan span 10°C, 25°C, 50°C, sedangkan Gambar 4.7 merupakan kalibrator penunjukan suhu pada *microbath*. Jenis sensor yang digunakan sebagai kalibratornya adalah sensor suhu PT 100 dan pada Gambar 4.8 adalah penunjukan hasil kalibrasi pada sensor suhu yang digunakan.

Tabel 4.1. Hasil Kalibrasi Sensor LM35 Pada Suhu 10°C

KALIBRASI							
Suhu Naik					Suhu Turun		
No.	Waktu 2 menit	Suhu LM35(°C)	Kalibrator(°C) PT 100	Error (%)	Suhu LM35(°C)	Kalibrator (°C) PT 100	Error (%)
1	I	10,74	10,69	0,005	10,25	10,12	0,013
2	II	11,32	10,69	0,053	10,74	10,04	0,070
3	III	10,74	10,70	0,004	10,25	10,00	0,025
4	IV	10,25	10,70	0,045	10,25	10,00	0,025
5	V	10,74	10,70	0,004	10,25	10,00	0,025
6	VI	10,20	10,70	0,047	9,76	10,00	0,024
Total		10,67	10,70	0,003	10,25	10,03	0,02

Pada pengukuran kalibrasi 10°C hasil seperti pada Tabel 4.1 dilakukan enam kali pengambilan data yang dilakukan dalam waktu 2 menit sekali. Batas *error* yang dizinkan dalam kalibrasi yang dilakukan di Lab kalibrasi *Temperature Gauge*, Departemen Kalibrasi Instrumentasi Wokrshop PGN Jakarta, adalah sebesar 1%. Dalam pengujian yang dilakukan menunjukan hasil sensor yang dikalibrasi memiliki *error* kurang dari 1%. Untuk menghitung *error* dapat digunakan rumus sederhana yaitu:

$$\text{Error} = \frac{\text{[suhu terukur - suhu sebenarnya]}}{\text{suhu sebenarnya}} \times 100\%$$

Kalibrasi dengan menggunakan sensor suhu LM35 tidak dapat menunjukan hasil yang presisi 10°C, melainkan melebihi atau kurang dari 10°C, dikarenakan interger pin Arduino tidak dapat membaca bilang koma, melainkan bilang bulat, sehingga hasil yang

didapat tidak dapat presisi. Untuk kalibrasi atau pengukuran dengan menggunakan sensor ini dalam suhu kurang dari 10°C tidak dianjurkan karena mendapati hasil yang tidak presisi sesuai yang diinginkan. Batas toleransi pengukuran suhu minimal jika menggunakan sensor suhu LM35 dapat dilihat pada Lampiran B. Untuk menghitung rangkuman kesalahan dapat digunakan perhitungan :

$$\text{Suhu}_{\text{rata-rata pengukuran}} = \frac{\text{Suhu1} + \text{Suhu2} + \text{Suhu3} + \text{Suhu4} + \text{Suhu5} + \text{Suhu6}}{6}$$

Rangkuman Kesalahan:

1. $\text{Suhu}_{\text{max}} - \text{Suhu}_{\text{rata-rata pengukuran}} = 11,32 - 10,67 = 0,65^{\circ}\text{C}$
2. $\text{Suhu}_{\text{rata-rata pengukuran}} - \text{Suhu}_{\text{min}} = 10,67 - 10,20 = 0,47^{\circ}\text{C}$
3. Kesalahan Rata-rata = $(0,65 + 0,47) / 2 = 0,56^{\circ}\text{C}$

Jadi rangkuman kesalahan pengukuran ketika suhu $10^{\circ}\text{C} \pm 0,56$.

Faktor yang mempengaruhi pengukuran adalah peletakan posisi sensor terhadap sumber yang mau diukur juga mempengaruhi ketepatan dalam pengukurang. Ketika Sensor diletakan jauh dari sumber yang ingin diukur maka ketepatannya juga akan kurang dari 98%, maka semakin dekat sensor dengan sumber yang diukur semakin besar keberhasilan dari pengukuran atau hasil kalibrasi tersebut.

Tabel 4.2. Kalibrasi Sensor LM35 Pada Suhu 25°C

KALIBRASI							
Suhu Naik				Suhu Turun			
No.	Waktu 2 menit	Suhu LM35(°C)	Kalibrator(°C) PT 100	Error (%)	Suhu LM35(°C)	Kalibrator (°C) PT 100	Error (%)
1	I	25,86	25,16	0,028	25,38	25,10	0,011
2	II	25,38	25,15	0,009	25,38	25,01	0,015
3	III	25,38	25,12	0,010	25,38	25,01	0,015
4	IV	25,38	25,11	0,011	25,86	25,00	0,034
5	V	25,38	25,10	0,011	24,89	24,95	0,002
6	VI	25,38	25,09	0,012	24,89	24,92	0,001
Total		25,46	25,12	0,01	25,30	25,00	0,01

Pada pengukuran kalibrasi 25°C hasil seperti pada Tabel 4.2. Batas *error* yang di izinkan dalam kalibrasi yang dilakukan di Lab kalibrasi *Temperature Gauge*, Departemen Kalibrasi Instrumentasi Wokrshop PGN Jakarta, adalah sebesar 1%. Dalam pengujian yang dilakukan menunjukan hasil sensor yang dikalibrasi memiliki *error* kurang dari 1%. Sama seperti pada pengukuran kalibrasi 10°C, dalam pengikuran 25°C sensor yang digunakan tidak dapat menunjukan ketepatan 25°C.

$$\text{Suhu}_{\text{rata-rata pengukuran}} = \frac{\text{Suhu1} + \text{Suhu2} + \text{Suhu3} + \text{Suhu4} + \text{Suhu5} + \text{Suhu6}}{6}$$

Rangkuman Kesalahan:

1. $\text{Suhu}_{\text{max}} - \text{Suhu}_{\text{rata-rata pengukuran}} = 25,86 - 25,46 = 0,40^{\circ}\text{C}$
2. $\text{Suhu}_{\text{rata-rata pengukuran}} - \text{Suhu}_{\text{min}} = 25,46 - 25,38 = 0,08^{\circ}\text{C}$
3. Kesalahan Rata-rata = $(0,40+0,08)/2 = 0,24^{\circ}\text{C}$

Jadi rangkuman kesalahan pengukuran ketika suhu 25°C±0,24.

Sensor suhu LM35 lebih baik digunakan ketika suhu berada dalam keadaan normal seperti pada suhu ruangan 25°C dibandingkan dengan suhu 10°C.

Tabel 4.3. Kalibrasi Sensor LM35 Pada Suhu 50°C

KALIBRASI							
Suhu Naik					Suhu Turun		
No.	Waktu 2 menit	Suhu LM35(°C)	Kalibrator(°C) PT100	Error (%)	Suhu LM35(°C)	Kalibrator (°C) PT100	Error (%)
1	I	50,26	49,74	0,010	50,56	49,80	0,015
2	II	50,26	49,74	0,010	50,51	49,80	0,014
3	III	50,38	49,74	0,013	50,38	49,81	0,011
4	IV	50,26	49,77	0,010	50,26	49,81	0,009
5	V	50,38	49,79	0,012	50,55	49,81	0,015
6	VI	50,38	49,79	0,012	50,38	49,82	0,011
Total		50.32	49.76	0.01	50.44	49.81	0.01

Sama dengan pengukuran kalibrasi 10°C dan 25°C pada kalibrasi suhu 50°C juga memnunjukan hasil yang baik. *Error* yang dihasilkan masih dalam batas toleransi yang dizinkan.

$$\text{Suhu}_{\text{rata-rata pengukuran}} = \frac{\text{Suhu1} + \text{Suhu2} + \text{Suhu3} + \text{Suhu4} + \text{Suhu5} + \text{Suhu6}}{6}$$

Rangkuman Kesalahan:

1. $\text{Suhu}_{\text{max}} - \text{Suhu}_{\text{rata-rata pengukuran}} = 50,38 - 50,32 = 0,06^{\circ}\text{C}$
2. $\text{Suhu}_{\text{rata-rata pengukuran}} - \text{Suhu}_{\text{min}} = 50,32 - 50,26 = 0,06^{\circ}\text{C}$
3. Kesalahan Rata-rata = $(0,40 + 0,08) / 2 = 0,24^{\circ}\text{C}$

Jadi rangkuman kesalahan pengukuran ketika suhu $50^{\circ}\text{C} \pm 0,06$.

Pada kalibrasi suhu 50°C menggunakan sensor suhu LM35 tidak dapat memiliki ketepatan 50°C . Dikarenakan perhitung sensor LM35 dalam Arduino memiliki rumus:

$$\text{Baca analog} = \text{analogRead}(\text{suhuPin}) \quad (4.1)$$

$$V_{\text{LM35}} = (\text{suhuPin} / 1023) * 5000\text{mV} \quad (4.2)$$

$$V_{\text{LM35}} = \text{suhu} * 10 \text{ mV} \quad (4.3)$$

$$\text{Suhu} = V_{\text{LM35}} / 10\text{mV} \quad (4.4)$$

Ket:

Baca analog : pengkonversian nilai analog menjadi *digital*

V_{LM35} : mengkonversi nilai sensor dari 0-1023 menjadi tegangan 0-5milivolt. (mV)

Suhu : penampilan suhu pada LCD/serialmonitor dalam ($^{\circ}\text{C}$)

Untuk mencari ketepatan pengukuran saat titik yang diuji, sebagai contoh adalah 50°C , maka:

$$50 = V_{\text{LM35}} / 10\text{mV}$$

$$V_{\text{LM35}} = 500\text{mV}$$

$$500 = (\text{suhuPin} / 1023) * 5000\text{mV}$$

$$(500 \times 1023) / 5000 = \text{suhuPin}$$

$$\text{suhuPin} = 102,3$$

Pada Arduino ini memiliki resolusi 10 bit, yaitu $2^{10} = 1023$ yang artinya nilai hasil konversi berkisar dari 0 hingga 1023. *Input* analog pada Arduino merupakan cacahan dari 0 hingga 1023. Artinya, 0 berarti 0 volt, dan 1023 berarti 5 volt. Dalam program yang digunakan menggunakan *interger*, yaitu bilangan bulat maka jika hasilnya adalah 102,3 dibulatkan menjadi 102, sehingga:

$$V_{LM35} = (102/1023) \times 5000$$

$$V_{LM35} = 498,53 \text{mV, maka suhu}$$

$$\text{suhu} = 498,53/10 \text{mV}$$

$$\text{suhu} = 49,85^\circ\text{C}$$

Jadi suhu yang penampilan suhu menggunakan LM35 hanya dapat terbaca $49,85^\circ\text{C}$ tidak dapat persis 50°C . Begitupun dengan kalibrasi suhu 10°C dan 25°C . Untuk spesifikasi toleransi LM35 dapat dilihat pada Lampiran B.

Tabel 4.4. Hasil Pengujian Ketika Pembukaan Gas dari cradle pada regulator1

Regulator 1			
No.	Waktu 2 Menit	Suhu LM35($^\circ\text{C}$)	Fluks ($^\circ\text{C}$)
1	I	27,82	24,3
2	II	20,50	22,6
3	III	19,03	20,7
4	IV	16,10	18,4
5	V	14,08	13,7
6	VI	9,76	10,8
7	VII	5,86	7,6
8	VII	4,48	1,3
9	IX	2,14	0
10	X	1,96	-1,2

Pada Tabel 4.4 pengujian terhadap suhu regulator1 merupakan hasil pengujian regulator PRS10 yang dialiri gas dari

cradle bertekanan 125bar, dengan pembukaan regulator1 sebesar 20bar pada regulator2 yang dilakukan selama 2menit sebanyak 10 kali pengambilan. Semakin besar regulator1 yang dibuka untuk mengalir gas semakin cepat juga pembekuannya, seperti yang di tampilkan pada Gambar 4.9 menguji pengukuran sensor suhu dengan regulator1 yang dialiri gas dengan pembukaan 20bar pada regulator 1, sejak pengambilan data 12 menit regulator 1 sudah mencapai suhu dibawah 10°C. Jika diteruskan sampai 20menit maka suhunya akan mendekati 0°C atau bisa mencapai minus. Kemampuan pembacaan suhu pada sensor yang diuji hanya sebesar 0°C - 100°C, seperti yang terlampir pada Lampiran B. Ketika suhu yang terukur pada fluks lebih dari -1°C, maka pembacaan pada sensor hanya akan menunjukkan 0°C, dikarenakan sensor hanya mampu membaca dalam range 0°C-100°C. Dalam pengujian ini juga suhu pada regulator 1 turun begitu cepat sehingga menyebabkan embun seperti yang ditunjukkan dalam Gambar 4.9.



Gambar 4.9. Embun pada Regulator1

Dalam Gambar 4.9 merupakan embun yang terjadi ketika suhu hamper mendekati 10°C. Embun ini menandakan regulator1 sudah mulai mengalami *freezing* dan *blocking* penyumbatan aliran gas yang akan mengalir ke regulator 2 PRS10. Jika diteruskan lebih dari 20menit regulator1 seluruh permukannya akan membeku menjadi es.

Tabel 4.5. Hasil Pengujian Ketika Pembukaan Gas dari *cradle* pada regulator2

Regulator 2			
No.	Waktu 2 Menit	Suhu LM35(°C)	Fluks (°C)
1	I	27,82	27,4
2	II	22,45	24,5
3	III	21,47	22,4
4	IV	19,57	20,8
5	V	16,10	18,2
6	VI	15,13	16,3
7	VII	14,15	15,7
8	VIII	13,18	14,2
9	IX	11,71	12,6
10	X	10,25	11,9

Pada Tabel 4.5 merupakan tabel pengujian sensor suhu pada regulator 2 ketika pembukaan gas dari *cradle* pada regulator2 yang dilakukan selama 2menit sebanyak 10 kali pengambilan. Pada pengujian sensor suhu ini terhadap regulator 2 yang tekanan keluarnya diatur sebesar 1,3bar menunjukkan hasil bahwa pada regulator 2 PRS10 tidak memnunjukkan pembekuan seperti pada regulator 1 yang datanya terdapat pada Tabel 4.4 serta Gambar 4.9. Dalam pengukuran sensor suhu terhadap regulator 2 ini, penurunan suhu pada regulator 2 terbilang sedikit lebih lambat daripada regulator1. Apabila percobaan dilakukan terus 20menit keatas, maka regulator1 kinerjanya akan terbebani dan akan menyumbat laju aliran gas pada PRS10.



Gambar 4.10. Merupakan Gambar Penampilan 2 Sensor Suhu LM35 pada LCD

4.4 Pengujian Relay 2 *channel* Terhadap Suhu

Tujuan dari pengujian ini adalah untuk mengetahui apakah relay yang menggunakan suhu sebagai kondisi kerjanya atau *input*, dapat bekerja dengan baik untuk menyalakan pemanas. Pada kontrol relay ini hanya bekerja menyala jika suhu regulator yang dialiri gas dari *cradle* ini berada pada *range* 0 °C - 10 °C dan mematikan relay beserta pemanas jika suhunya sudah mencapai 25°C - 40°C. Adapun kondisi dari kontrol relay serta pemanasnya dapat dilihat pada Gambar 4.7. Skematik dan program untuk mengontrol relay yang digunakan pada pengendalian relay ini terdapat pada Gambar 3.11. Gambar 3.12, Gambar 3.13, Gambar 3.14, Gambar 3.15.

Tabel 4.6. Pengujian Kontrol Relay1 Terhadap Suhu1

REGULATOR 1					
No.	Waktu 2 Menit	Suhu1 °C	Suhu1 °F	Status Relay 1	Status Pemanas
1	I	27,82	47.46	OFF	OFF
2	II	20,50	43.39	OFF	OFF
3	III	19,03	42.57	OFF	OFF
4	IV	16,10	40.94	OFF	OFF
5	V	14,08	39.82	OFF	OFF
6	VI	9,76	37.42	ON	ON
7	VII	5,86	35.26	ON	ON
8	VIII	4,48	34.49	ON	ON
9	IX	2,14	33.19	ON	ON
10	X	1,96	33.09	ON	ON

Tabel 4.7. Pengujian Pemanas Terhadap Regulator1

REGULATOR 1							
No.	Waktu 2 Menit	Suhu1 °C	Suhu1 °F	Set Point1	Status Relay 1	Status Pemanas	Suhu pemanas °C
1	I	9,76	37.42	35	ON	ON	27,8
2	II	5,86	42.55	35	ON	ON	28,9
3	III	4,48	40.06	35	ON	ON	31,2
4	IV	2,14	34.85	35	ON	ON	35,6
5	V	1,27	34,29	35	ON	ON	39,4
6	VI	0	32	35	ON	ON	45



Gambar 4.11. Pemasangan Pemanas pada Regulator



Gambar 4.12. Pengujian Sensor Panas yang Plat Sebagai Penggantinya



Gambar 4.13. Hasil Pengujian Sensor Panas yang Plat Sebagai Penggantinya

Pada Tabel 4.6 dan Tabel 4.7 adalah percobaan mengenai relay terhadap sensor suhu pada regulator. Jika suhunya mencapai rentan 0°C - 10°C , maka akan menyalakan relay yang kemudian akan menyalakan pemanas satu juga seperti yang ditampilkan pada Tabel 4.7. Dalam pengujian Tabel 4.7 digunakan lempengan besi yang di tempelkan peltier sebagai panas dikarenakan peltier tidak bisa dipasangkan pada regulator 1 seperti dalam penyajian Gambar 4.8. Dalam penyajian Gambar 4.8 pemanas tidak menempel secara baik dengan regulator, maka digunakan permisalan untuk mengetahui apakah pemanas dapat berjalan dengan baik dalam penyebarannya seperti yang di tampilkan dalam Gambar 4.9.

Dalam Tabel 4.7 merupakan tabel untuk memantau suhu dari pemanas peltier. Pada Tabel 4.7, dalam nomer 1 waktu untuk menyalakan peltier ketika suhu mencapai $9,76^{\circ}\text{C}$ seperti yang ditampilkan pada Tabel 4.6 sebelumnya. Ketika ini peltier yang di misalkan ditempelkan pada besi awalnya dalam kondisi *Off* sehingga, suhu peltier mengikuti suhu lingkungan. Ketika suhu pada regulator sudah mencapai rentan 0°C - 10°C , peltier yang ditempelkan dalam media lempeng besi akan menyala dan hasil pengujiannya pada Gambar 4.10.

Pemanas yang ditempelkan pada lempeng besi menunjukkan suhu awal $27,80$, kemudian semakin turun suhu regulator dari 10°C mendekati 0°C peltier akan semakin panas. Pengujian pemanas dengan pembanding suhu pada regulator1 yang dilakukan ketika suhu $9,76^{\circ}\text{C}$ dalam Tabel 4.7 menyala, kemudian peltier dari suhu normal 27°C mulai menyala selama dua menit, sehingga suhunya mencapai 28°C . Selanjutnya suhu pada regulator1 semakin menurun tetap membuat pemanas menyala sampai batas suhu yang sudah ditentukan dalam program Gambar 3.13. Setelah kondisinya memenuhi maka relay akan otomatis memutus aliran untuk mematikan pemanas. Suhu pemanas diukur menggunakan fluks yang ditembakkan langsung ke pemanas.

Dalam pengujian regulator 2 pemanas tidak menyala dikarenakan tidak mencapai kondisi suhu yang sudah ditentukan. Pada pengujian ini apabila dilanjutkan mencapai suhu dibawah 10°C maka kinerja regulator 1 akan terbebani, sehingga pengujian regulator 2 hanya dapat dilakukan dalam waktu 20 menit. Dalam waktu tersebut pengujian relay 2 *channel* terhadap suhu untuk menyalakan pemanas terhadap regulator tidak dapat dilakukan.

----Halaman ini sengaja dikosongkan-----

BAB V

PENUTUP

5.1 Kesimpulan

Pada penelitian kali ini dapat diambil kesimpulan sebagai berikut:

1. Pemanas akan menyala ketika suhu regulator mencapai $0^{\circ}\text{C} - 10^{\circ}\text{C}$.
2. Pemanas akan mati ketika suhu regulator mencapai $25^{\circ}\text{C} - 40^{\circ}\text{C}$.
3. Sensor suhu LM35 yang digunakan tidak dapat mengukur keadaan dibawah 0°C .
4. Kemampuan sensor saat kalibrasi suhu 10°C memiliki kesalahan pengukuran $10^{\circ}\text{C} \pm 0,56$, sedangkan pada $25^{\circ}\text{C} \pm 0,24$ dan $50^{\circ}\text{C} \pm 0,06$. Sebaiknya sensor digunakan dalam rentang suhu normal, yaitu suhu lingkungan $25^{\circ}\text{C} - 50^{\circ}\text{C}$.
5. Pemanas peltier sebaiknya digunakan pada maksimal suhu 50°C supaya tidak terjadi *over heat*.

5.2 Saran

Untuk penelitian selanjutnya, diharapkan dapat menggunakan sensor suhu yang lebih sensitif lagi terhadap sebuah respon perubahan dingin dan panas seperti thermocouple tipe k dan pemanas yang dapat lebih merata dalam penyebaran panasnya sehingga panas dapat menyebar dengan baik dan mencegah terjadinya pembekuan dalam regulator.

----Halaman ini sengaja dikosongkan-----

DAFTAR PUSTAKA

- Ahbab, M. N. (2013). *Analisa Numerik Pada Rancangan Regulator untuk CNG Tekanan Tinggi*. Yogyakarta: UGM.
- Ariwibowo, B. (2017). *Rancang Bangun Sistem Pengatur Sirkulasi Air Berdasarkan pH dan Temperatur Air Pada Kolam Ikan Gurami Berbasis Arduino*. Surabaya: ITS.
- Habibah, A. (2018). *Monitoring Pemantauan Kualitas Udara*. Surabaya: ITS.
- Kho, D. (2017, July 20). Retrieved from Teknik Elektro: <https://teknikelektronika.com/pengertian-lcd-liquid-crystal-display-prinsip-kerja-lcd/>
- Kho, D. (2017, Juli 28). Retrieved from Teknik Elektro: <https://teknikelektronika.com/pengertian-relay-fungsi-relay/>
- Kho, D. (2017, Maret 03). Retrieved from Teknik Elektronika: <https://teknikelektronika.com/pengertian-power-supply-jenis-catu-daya/>
- Mukhlis, Y., & Yapie, A. K. (2010). Pemanas dengan Sistem Pendeteksi Suhu Otomatis dan Pengamna Kebocoran Panas. *Jurnal Ilmiah Teknologi dan Rekayasa, Volume 15 No. 3*.
- Purwiyanti, S., Setyawan, A., Selviana, W., & Purnamasari, D. (2017). Aplikasi Efek Peltier Sebagai Kotak Penghangat dan Pendinginan Berbais Mikroprosesor Arduino Uno. *Jurnal Rekayasa dan Teknologi Elektro*, 99-100.
- Rahayoe, S. (2004). *Teknik Pendinginan dan Pembekuan*. Yogyakarta: UGM.
- Santosa, N. B. (2015, Januari 29). Retrieved from <http://www.vedcmalang.com/pppptkboemlg/index.php/menutama/listrik-elektro/1292-mengenal-thermo-electric-peltier>
- Zaki, A. (2017). *Kontrol Temperatur dan Kelembaman Pada Alat Penetas Telur*. Surabaya: ITS.
- Zennifa, F. (2012). Perancangan dan Implementasi Pengontrol Suhu Ruangan dengan Menggunakan Sensor LM35 Berbasis Mikrokontroler Arduino Uno.

----Halaman ini sengaja dikosongkan-----

LAMPIRAN A

```
#include <Keypad.h>
#include <Wire.h> //untuk mengaktifkan fungsi library wire
#include <LiquidCrystal_I2C.h> //untuk mengaktifkan lcd i2c
```

```
LiquidCrystal_I2C lcd(0x27, 20, 4);
```

```
int setpoint1 = 0;
int setpoint2 = 0;
```

```
float tempC0;
float tempC;
float tempC2;
float tempC3;
float fahrenheit;
float fahrenheit2;
float fahrenheit3;
```

```
char customKey;
```

```
int tempPin0 = A0;
int tempPin = A1; //analog A1
int tempPin2 = A2; //analog A2
int tempPin3 = A3;
```

```
int relay1 = 7;
int relay2 = 6;
```

```
int ledhijau = 5;
int ledmerah = 4;
int ledhijau1 = 3;
int ledmerah1 = 2;
```

```
String inString = "";
```

```
int count = 0;
int j = 0 ;
int state = 0, kondisi = 0;
```

```

const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
//define the symbols on the buttons of the keypad
char hexaKeys[ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
byte rowPins[ROWS] = {24, 26, 28, 30}; //connect to the row
pinouts of the keypad
byte colPins[COLS] = {32, 34, 36, 38}; //connect to the column
pinouts of the keypad

//initialize an instance of class NewKeypad
Keypad customKeypad = Keypad( makeKeymap(hexaKeys),
rowPins, colPins, ROWS, COLS);

//=====================================================SETUP=====
void setup()
{
  lcd.begin();
  Serial.begin(9600);

  pinMode (A0, INPUT);
  pinMode (A1, INPUT);
  pinMode (A2, INPUT);

  pinMode (relay1, OUTPUT);
  pinMode (relay2, OUTPUT);

  pinMode (ledmerah, OUTPUT);
  pinMode (ledhijau, OUTPUT);

  digitalWrite (relay1, LOW); //relay aktif low
  digitalWrite (relay2, LOW);

  digitalWrite (ledmerah, LOW);
  digitalWrite (ledhijau, LOW);

```

```

digitalWrite (ledmerah1, LOW);
digitalWrite (ledhijau1, LOW);
}

//=====================================================MEMUNCULKAN SUHU=====
void suhu()
{
  if (state == 1)
  {
    //relay1
    if (25 < tempC < 35 && setpoint1 < 34 && setpoint1 >= 25)//
    kondisi relay1
    {
      digitalWrite (relay1, HIGH);
      digitalWrite (ledmerah, HIGH);
      digitalWrite (ledhijau, LOW);
    }
    else if (25 < tempC < 35)
    {
      digitalWrite (relay1, HIGH);
    }

    if (10 >= tempC > 0 && 0 <= setpoint1 && setpoint1 < 10)
    {
      digitalWrite (relay1, LOW);
      digitalWrite (ledhijau, HIGH);
      digitalWrite (ledmerah, LOW);
    }
    else if (10 >= tempC > 0)
    {
      digitalWrite (relay1, LOW);
    }
    if (10 >= tempC > 0)
    {
      digitalWrite (relay1, LOW);
    }
    else if (-10 < tempC <= 0) //kondisi penurunan critical pada
    regulator 1

```

```

{
    digitalWrite (relay1, LOW);
}

if (35 <= tempC < 55)//kondisi naik ekstrim suhu 1 pada regulator
{
    digitalWrite (relay1, HIGH);
}
else if (35 <= tempC0 < 55)//kondisi naik ekstrim suhu1 pada pemanas1
{
    digitalWrite (relay1, HIGH);
}
//relay2
if (25 < tempC2 < 35 && setpoint2 < 34 && setpoint2 >=
25)//kondisi relay2
{
    digitalWrite (relay2, HIGH);
    digitalWrite (ledhijau1, LOW);
    digitalWrite (ledmerah1, HIGH);
}
else if (25 < tempC2 < 35)
{
    digitalWrite (relay2, HIGH);
}
if (10 >= tempC2 > 0 && 0 <= setpoint2 && setpoint2 < 10)
{
    digitalWrite (relay2, LOW);
    digitalWrite (ledhijau1, HIGH);
    digitalWrite (ledmerah1, LOW);
}
else if (10 >= tempC2 > 0)
{
    digitalWrite (relay2, LOW);
}

if (10 >= tempC2 > 0)
{
    digitalWrite (relay2, LOW);
}

```

```

    }
    else if (-10 < tempC2 <= 0) //kondisi penurunan critical pada
regulator 2
    {
        digitalWrite (relay2, LOW);
    }

    set1();
    set2();

    if (count == 20000)
    {
        tempPin0 = analogRead(A0);
        tempC0 = tempPin0 * 0.488; //suhu pemanas1

        tempPin = analogRead(A1);
        tempC = tempPin * 0.488;
        fahrenheit = (tempC * 1.8) + 32;

        tempPin2 = analogRead(A2);
        tempC2 = tempPin2 * 0.488;
        fahrenheit2 = (tempC2 * 1.8) + 32;

        tempPin3 = analogRead(A3);
        tempC3 = tempPin3 * 0.488; //suhu pemanas2

        lcd.clear();

        lcd.setCursor (14, 1);
        lcd.print ("suhu");//suhu pemanas 1
        lcd.print(tempC0);
        lcd.setCursor (14, 3);
        lcd.print ("suhu");//suhu pemanas 2
        lcd.print(tempC3);

        lcd.setCursor (0, 0);
        lcd.print ("suhu1=");

```

```

    lcd.print(tempC);
    lcd.print ("C");
    lcd.setCursor (14, 0);
    lcd.print (fahrenheit);
    lcd.print ("F");
    lcd.setCursor (0, 1);
    lcd.print ("setpoint1:");
    lcd.print(setpoint1);

    lcd.setCursor (0, 2);
    lcd.print ("suhu2=");
    lcd.print(tempC2);
    lcd.print ("C");
    lcd.setCursor (14, 2);
    lcd.print (fahrenheit2);
    lcd.print ("F");
    lcd.setCursor (0, 3);
    lcd.print ("setpoint2:");
    lcd.print(setpoint2);
}
}

if ( state == 2)
{
    if (count == 20000)
    {
        lcd.setCursor (0, 0);
        lcd.print ("Isi Set Point 1:");
    }
}
if ( state == 3)
{

    if (count == 20000)
    {
        lcd.setCursor (0, 0);
        lcd.print ("Isi Set Point 2:");
    }
}
}

```

```

if (customKey && state == 2)
{
    lcd.setCursor(j, 1);
    lcd.print(customKey);
    j++;
    if (isDigit(customKey)) //pengkondisian konversi char ke keypad
saat awal
    {
        inString += customKey; //pengubahan char ke string
    }
    if (isAlpha(customKey))
    {
        inString += customKey;
    }
}

if (customKey && state == 3)
{
    lcd.setCursor(j, 1);
    lcd.print(customKey);
    j++;
    if (isDigit(customKey)) //pengkondisian konversi char ke keypad
saat awal
    {
        inString += customKey; //pengubahan char ke string
    }
    if (isAlpha(customKey))
    {
        inString += customKey;
    }
    if (customKey == '*')
    {
        int nilai = inString.toInt();
        setpoint2 = nilai;
        state = 1;
        j = 0;
        lcd.clear();
        lcd.setCursor(0, 2);
        lcd.print ("OK....");
    }
}

```

```

        delay(500);
        lcd.clear();
        inString = " ";
    }
}

//=====================================================PENGATURAN RELAY1=====
void set1()
{
    if (customKey == 'A' && state == 1)
    {

        state = 0;
        kondisi = 1;
        customKey = false;
        inString = " ";
        lcd.clear();
    }
}

//=====================================================PENGATURAN RELAY2=====
void set2()
{
    if (customKey == 'B' && state == 1)
    {

        state = 0;
        kondisi = 2;
        customKey = false;
        inString = " ";
        lcd.clear();
    }
}

//=====================================================PENGATURAN KEYPAD DAN PASSWORD=====
void loop()
{

```

```

customKey = customKeypad.getKey();
if (count == 50000)
{
    count = 0;
}
count++;
suhu();
if (state == 0)
{
    if (count == 20000)
    {
        lcd.setCursor (0, 0);
        lcd.print ("Masukkan Password:");
    }

}
if (customKey == '#')
{
    lcd.clear();
    lcd.setCursor (0, 0);
    lcd.print ("Reseting...");
    lcd.clear();
    delay(500);
    j = 0;
    inString = " ";
    state = 0;
    kondisi = 0;
    customKey = false;

}
if (customKey && state == 0)
{
    lcd.setCursor (j, 1);
    lcd.print("*");
    j++;

    if (isDigit(customKey)) //pengkondisian konversi char ke keypad
    saat awal
    {

```

```

    inString += customKey; //pengubahan char ke string
}
if (isAlpha(customKey))
{
    inString += customKey;
}

if (customKey == '*')
{

    int nilai = inString.toInt();
    if (nilai == 1234 && kondisi == 0)
    {
        customKey = false;
        state = 1;
        j = 0;
        inString = " ";
        lcd.clear();
        lcd.setCursor (0, 2);
        lcd.print ("Completed");
        delay(500);
        lcd.clear();
    }
    if (nilai == 1234 && kondisi == 1)
    {
        customKey = false;
        state = 2;
        kondisi = 0;
        j = 0;
        inString = " ";
        lcd.clear();
        lcd.setCursor (0, 2);
        lcd.print ("Completed");
        delay(500);
        lcd.clear();
    }
    if (nilai == 1234 && kondisi == 2)
    {
        customKey = false;

```

```

state = 3;
kondisi = 0;
j = 0;
inString = " ";
lcd.clear();
lcd.setCursor (0, 2);
lcd.print ("Completed");
delay(500);
lcd.clear();
}
if (nilai != 1234)
{
    customKey = false;
    j = 0;
    inString = " ";
    lcd.clear();
    lcd.setCursor (0, 2);
    lcd.print ("Access Denied");
    delay(500);
    lcd.clear();
}
}
}
}

```

----Halaman ini sengaja dikosongkan-----

LAMPIRAN B

6.7 Electrical Characteristics: LM35, LM35C, LM35D Limits

Unless otherwise noted, these specifications apply to $-55^{\circ}\text{C} \leq T_A \leq 150^{\circ}\text{C}$ 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 $I_{LOAD} = 50\text{ }\mu\text{A}$, in the circuit of Full-Range Centigrade Temperature Sensor. These specifications also apply from 2°C to T_{MAX} in the circuit of Figure 14.

PARAMETER	TEST CONDITIONS	LM35			LM35C, LM35D			UNIT
		TYP	TESTED LIMIT ⁽¹⁾	DESIGN LIMIT ⁽²⁾	TYP	TESTED LIMIT ⁽¹⁾	DESIGN LIMIT ⁽²⁾	
Accuracy, LM35, LM35C ⁽²⁾	$T_A = 25^{\circ}\text{C}$	± 0.4	± 1		± 0.4	± 1		$^{\circ}\text{C}$
	$T_A = -10^{\circ}\text{C}$	± 0.5			± 0.5		± 1.5	
	$T_A = T_{MAX}$	± 0.8	± 1.5		± 0.8		± 1.5	
	$T_A = T_{MIN}$	± 0.8		± 1.5	± 0.8		± 2	
Accuracy, LM35D ⁽²⁾	$T_A = 25^{\circ}\text{C}$				± 0.6	± 1.5		$^{\circ}\text{C}$
	$T_A = T_{MAX}$				± 0.9		± 2	
	$T_A = T_{MIN}$				± 0.9		± 2	
Nonlinearity ⁽²⁾	$T_{MIN} \leq T_A \leq T_{MAX}$, $-40^{\circ}\text{C} \leq T_J \leq 125^{\circ}\text{C}$	± 0.3		± 0.5	± 0.2		± 0.5	$^{\circ}\text{C}$
Sensor gain (average slope)	$T_{MIN} \leq T_A \leq T_{MAX}$, $-40^{\circ}\text{C} \leq T_J \leq 125^{\circ}\text{C}$	10	9.8		10		9.8	$\text{mV}/^{\circ}\text{C}$
		10	10.2		10		10.2	
Load regulation ⁽²⁾ $0 \leq I_L \leq 1\text{ mA}$	$T_A = 25^{\circ}\text{C}$	± 0.4	± 2		± 0.4	± 2		mV/mA
	$T_{MIN} \leq T_A \leq T_{MAX}$, $-40^{\circ}\text{C} \leq T_J \leq 125^{\circ}\text{C}$	± 0.5		± 5	± 0.5		± 5	
Line regulation ⁽²⁾	$T_A = 25^{\circ}\text{C}$	± 0.01	± 0.1		± 0.01	± 0.1		mV/V
	$4\text{ V} \leq V_S \leq 30\text{ V}$, $-40^{\circ}\text{C} \leq T_J \leq 125^{\circ}\text{C}$	± 0.02		± 0.2	± 0.02		± 0.2	
Quiescent current ⁽²⁾	$V_S = 5\text{ V}$, 25°C	56	80		56	80		μA
	$V_S = 5\text{ V}$, $-40^{\circ}\text{C} \leq T_J \leq 125^{\circ}\text{C}$	106		158	91		138	
	$V_S = 30\text{ V}$, 25°C	56.2	82		56.2	82		
	$V_S = 30\text{ V}$, $-40^{\circ}\text{C} \leq T_J \leq 125^{\circ}\text{C}$	106.5		161	91.5		141	
Change of quiescent current ⁽²⁾	$4\text{ V} \leq V_S \leq 30\text{ V}$, 25°C	0.2	2		0.2	2		μA
	$4\text{ V} \leq V_S \leq 30\text{ V}$, $-40^{\circ}\text{C} \leq T_J \leq 125^{\circ}\text{C}$	0.5		3	0.5		3	
Temperature coefficient of quiescent current	$-40^{\circ}\text{C} \leq T_J \leq 125^{\circ}\text{C}$	0.39		0.7	0.39		0.7	$\mu\text{A}/^{\circ}\text{C}$
Minimum temperature for rate accuracy	In circuit of Figure 14, $I_L = 0$	1.5		2	1.5		2	$^{\circ}\text{C}$
Long term stability	$T_J = T_{MAX}$, for 1000 hours	± 0.08			± 0.08			$^{\circ}\text{C}$

(1) Tested Limits are ensured and 100% tested in production.

----Halaman ini sengaja dikosongkan-----

LAMPIRAN C



Mesin Kalibrator Microbath



Temperature Indicator

K-011/0014		FORMULIR KALIBRASI TEMPERATURE GAUGE												
NO.	1													
No Order	/WO.TG/ KLI/ /													
Nama Pelanggan	Tanggul Harung Diarrima Tanggul Harung Diarrima Suhu Ruang Sebelum Pengujian Humidity Ruang Sebelum Pengujian Suhu Ruang Setelah Pengujian Humidity Ruang Setelah Pengujian													
Alamat Pelanggan														
JENIS	IC	S. DIA	LUT		RANGE	0	150	JENIS	STANDAR					
MERK	LM	S. LENGTH			SATUAN	°C		MERK	Readout Digital - PT 100					
TYPE/MODEL	12Z	RESOLUSI						TYPE/MODEL	WSA					
NO.SERI	38	CLASS						NO.SERI	CTH-6500					
									75014071543					
									°C					
HASIL KALIBRASI														
NO	DATA	UP				DOWN				KETERANGAN				
		TITIK 0°C		TITIK 100°C		TITIK 0°C		TITIK 100°C						
		TRTD	TULT	TRTD	TULT	TRTD	TULT	TRTD	TULT					
1	I	10.69	10.79	25.10	25.38	49.79	50.26	49.80	50.56	25.10	25.38	10.12	10.25	☐ OK
2	II	10.69	11.52	25.15	25.38	49.79	50.26	49.80	50.51	25.01	25.38	10.09	10.79	☐ NG
3	III	10.70	10.79	25.12	25.38	49.75	50.38	49.81	50.38	25.01	25.38	10.00	10.25	☐ SCRAP
4	IV	10.70	10.25	25.11	25.38	49.77	50.26	49.81	50.26	25.00	25.38	10.00	10.25	☐ REPAIR
5	V	10.10	10.79	25.10	25.58	49.75	50.38	49.81	50.55	29.19	29.19	10.00	10.25	
6	VI	10.70	10.20	25.09	25.38	49.79	50.38	49.82	50.38	29.19	29.19	10.00	9.76	
Catatan : Untuk Pengambilan Data Kalibrasi Sensor Suhu LM35F dengan Kalibrator yang sudah terverifikasi														
JAKARTA, 2019														
TEKNSI														

Form Kalibrasi 1

K-011/0014		FORMULIR KALIBRASI TEMPERATURE GAUGE												
NO.	1													
No Order	/WO.TG/ KLI/ /													
Nama Pelanggan	Tanggul Harung Diarrima Tanggul Harung Diarrima Suhu Ruang Sebelum Pengujian Humidity Ruang Sebelum Pengujian Suhu Ruang Setelah Pengujian Humidity Ruang Setelah Pengujian													
Alamat Pelanggan														
JENIS	IC	S. DIA	LUT		RANGE	0	150	JENIS	STANDAR					
MERK	LM	S. LENGTH			SATUAN	°C		MERK	Readout Digital - PT 100					
TYPE/MODEL	12Z	RESOLUSI						TYPE/MODEL	WSA					
NO.SERI	35	CLASS						NO.SERI	CTH-6500					
									75014071543					
									°C					
HASIL KALIBRASI														
NO	DATA	UP				DOWN				KETERANGAN				
		TITIK 0°C		TITIK 100°C		TITIK 0°C		TITIK 100°C						
		TRTD	TULT	TRTD	TULT	TRTD	TULT	TRTD	TULT					
1	I	10.69	10.79	29.81	29.85	49.79	49.78	49.55	49.78	29.42	29.89	10.17	☐ OK	
2	II	10.69	11.11	29.85	29.85	49.78	49.78	49.52	49.78	29.88	29.89	10.06	10.79	☐ NG
3	III	10.70	10.79	29.82	29.85	49.76	49.76	49.52	49.78	29.42	29.40	10.02	10.25	☐ SCRAP
4	IV	10.70	10.25	29.83	29.40	49.79	49.78	49.52	49.52	29.42	29.42	10.01	10.79	☐ REPAIR
5	V	10.70	10.79	29.81	29.89	49.75	49.75	49.50	49.78	29.88	29.88	10.00	10.29	
6	VI	10.70	10.20	29.83	29.40	49.75	49.75	49.51	49.78	29.88	29.88	10.00	10.79	
Catatan : Untuk Pengambilan Data Kalibrasi Sensor Suhu LM35F dengan Kalibrator yang sudah terverifikasi														
JAKARTA, 2019														
TEKNSI														

Form Kalibrasi 2

DAFTAR RIWAYAT HIDUP



Nama : Ahmad Abiyyu
TTL : Bekasi, 4 Mei 1998
Jenis Kelamin : Laki-Laki
Agama : Islam
Alamat Rumah : Jl. Gading Raya2,
No.8
Telpn/HP : 085814401975
E-mail :
sabakusou2@yahoo.com

RIWAYAT PENDIDIKAN

1. 2002 – 2004 : TK Aisyah 86
2. 2004 – 2010 : SDN Rawamangun 01 Pagi
3. 2010 -2013 : SMPN 158 Jakarta
4. 2013 – 2016 : SMAN 31 Jakarta
5. 2016 – 2019 : Departemen Teknik Elektro
Otomasi Program Studi Komputer
Kontrol - Fakultas Vokasi (FV)
Institut Teknologi Sepuluh
Nopember (ITS)

PENGALAMAN KERJA

1. Magang di PT. PGN Solution (Persero) Area Jakarta

PENGALAMAN ORGANISASI

1. Staff Pemetaan Departemen PSDM Periode 2017/2018
Himpunan Mahasiswa D3 Teknik Elektro, Institut Teknologi
Sepuluh Nopember
2. Kabiropemetaan Departemen PSDM Periode 2017/2018
Himpunan Mahasiswa D3 Teknik Elektro, Institut Teknologi
Sepuluh Nopember