

**SISTEM PENGENDALIAN SUHU MENGGUNAKAN
METODE *FUZZY* PADA ALAT PENDINGIN JAMUR TIRAM
SEBAGAI ALTERNATIF PENJEMURAN MATAHARI**

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL

**Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik**



**ANDREW KRISTANTYO
NIM. 0910633001**

**UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2016**

LEMBAR PENGESAHAN

**SISTEM PENGENDALIAN SUHU MENGGUNAKAN METODE *FUZZY*
PADA ALAT PENGERING JAMUR TIRAM SEBAGAI ALTERNATIF
PENJEMURAN MATAHARI**

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL

**Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik**



ANDREW KRISTANTYO

NIM. 0910633001

Telah direvisi dan disetujui oleh dosen pembimbing
Pada tanggal 25 Agustus 2016

Dosen Pembimbing I

Dosen Pembimbing II

Dr.Ir. Bambang Siswojo, MT.

NIP. 196212111988021001

Ir. Retnowati, MT.

NIP.195112241982032001

Mengetahui

Ketua jurusan

M. Aziz Muslim, S.T, M.T., Ph.D.

NIP : 19741203 200012 1 001

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya dan berdasarkan hasil penelusuran berbagai karya ilmiah, gagasan dan masalah ilmiah yang diteliti dan diulas di dalam Naskah Skripsi ini adalah asli dari pemikiran saya. tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu Perguruan Tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka. Apabila ternyata di dalam naskah Skripsi/Tesis/Disertasi ini dapat dibuktikan terdapat unsur-unsur jiplakan, saya bersedia Skripsi/Tesis/Disertasi dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, pasal 25 ayat 2 dan pasal 70).

Malang, 24 Agustus 2016

Mahasiswa,

Andrew Kristantyo

NIM. 0910633001





Teriring Ucapan Terima Kasih Kepada,
Ayahanda dan Ibunda Tercinta



KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa, Tuhan Yesus, atas limpahan Rahmat dan Karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul: Sistem Pengendalian Suhu Menggunakan Metode Fuzzy pada Alat Pengering Jamur Tiram Sebagai alternatif Penjemuran Matahari. Ini untuk memenuhi salah satu syarat menyelesaikan studi serta dalam rangka memperoleh gelar Sarjana Pendidikan Strata Satu pada Program Studi Teknik Elektro Universitas Brawijaya Malang

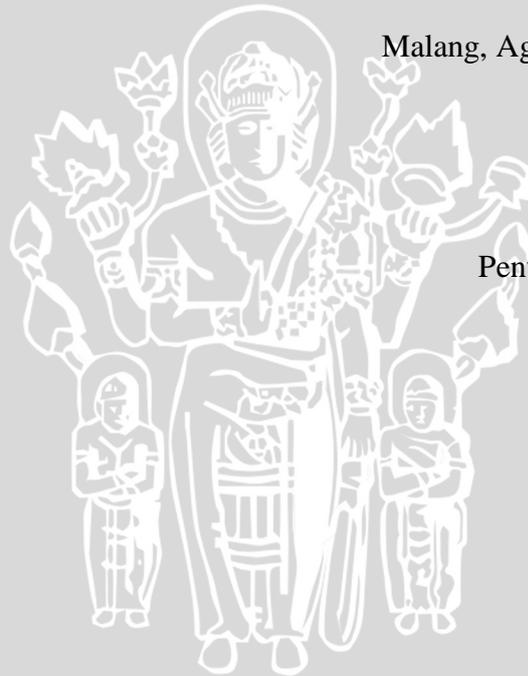
Penghargaan sebesar-besarnya dan terima kasih yang setulus-tulusnya kepada Ayah tercinta Bapak Subandi dan Ibunda yang kusayangi Ibu Ngatinah yang telah mencurahkan dukungan dan kasih sayang serta perhatian moral maupun materil. Semoga Tuhan Yang Maha Esa selalu melimpahkan Rahmat, Kesehatan, Karunia dan keberkahan di dunia dan di akhirat atas budi baik yang telah diberikan kepada penulis. Penghargaan dan terima kasih juga penulis berikan kepada Bapak Dr.Ir. Bambang Siswojo, M.T. selaku Pembimbing I dan Ir. Retnowati, MT selaku Pembimbing II yang telah membantu penulisan skripsi ini. Serta ucapan terima kasih kepada :

1. Bapak M. Aziz Muslim, ST., MT., Ph.D Selaku ketua jurusan Teknik Elektro Universitas Brawijaya.
2. Bapak Hadi Suyono, ST., MT., Ph.D Selaku Sekretaris Jurusan Teknik Elektro Universitas Brawijaya.
3. Bapak Ali Mustofa, ST.,M.T. Selaku Ketua Program Studi Jurusan Teknik Elektro atas arahnya.
4. Bapak Ir. Purwanto , MT selaku KKDK Teknik Kontrol.
5. Bapak Dr.Ir. Bambang Siswojo, M.T. sebagai Dosen Pembimbing I atas segala bimbingan, pengarahan, gagasan, ide, saran, motivasi, yang telah diberikan.
6. Ibu Ir. Retnowati, M.T. sebagai Dosen Pembimbing II atas segala bimbingan, pengarahan, gagasan, ide, saran, motivasi, yang telah diberikan.
7. Bapak ibu Dosen, karyawan, staf rekording dan RBTE atas segala bantuan dan kemudahan.

8. Orang tua saya, Ayah dan Ibu serta adik adik saya atas dukungannya.
9. Teman – teman seperjuangan dalam mengerjakan skripsi, terima kasih atas dukungan dan bantuannya.
10. Keluarga besar angkatan 2009 AMPERE atas do'a semangat, serta dukungan baik secara langsung maupun tidak langsung kepada penulis.
11. Semua pihak yang telah memberikan bantuan serta dukungan baik secara langsung maupun tidak langsung.

Sekiranya Tuhan Yang mencatat amalan ikhlas kami dan semua pihak yang turut membantu sehingga skripsi ini terselesaikan. Akhirnya, kami menyadari bahwa skripsi ini masih jauh dari sempurna namun semoga skripsi ini dapat memberikan manfaat bagi kita semua.

Malang, Agustus 2016



Penulis

RINGKASAN

Andrew Kristantyo, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya, Agustus 2016, *Sistem Pengendalian Suhu Menggunakan Metode Fuzzy pada Alat Pengering Jamur Tiram Sebagai alternatif Penjemuran Matahari*, Dosen Pembimbing : Dr.Ir. Bambang Siswojo, MT dan Ir. Retnowati, MT.

Dengan perkembangan zaman yang semakin maju yang segalanya dapat dilakukan dengan alat elektronik, sehingga hal ini mendorong manusia untuk selalu mencari cara yang termudah untuk melakukan pekerjaan yang sulit. Salah satunya adalah inovasi yang akan diteliti kali ini yaitu Alat Pengering Jamur Tiram. Seperti yang kita pahami jamur tiram ini memiliki kandungan gizi yang sangat baik untuk tubuh kita, Dari alasan tersebut di lakukan penelitian ini. Alat ini di harapkan mampu memangkas waktu yang di butuhkan untuk pengeringan jamur sehingga dapat meningkatkan hasil produksi jamur tiram kering.

Pada penelitian ini di gunakan sampel jamur 500gr dan suhu sebesar 62-69 °C, Dengan target mengurangi kadar air jamur yang semula 90% menjadi 4%, Nilai dari suhu dan kondisi jamur akan ditampilkan dalam *Liquid Crystal Display* (LCD) sehingga pemantauan akan lebih mudah yaitu hanya dengan melihat dari LCD. Hasil pengujian pengeringan jamur dengan menggunakan metode *fuzzy* berlangsung selama 9 jam, dimana sistem mempunyai parameter kontrol ime delay = 38Menit, Time rise = 77Menit, Time peak =90Menit. Proses pengeringan ini jauh lebih hemat waktu jika di bandingkan dengan proses pengeringan manual di bawah terik matahari yang membutuhkan waktu kurang lebih 2 hari.

Kata Kunci— Jamur Tiram, Pengeringan, , Kontroller, Fuzzy, temperatur

SUMMARY

Andrew Kristantyo, Department of Electrical Engineering, Faculty of Engineering, University of Brawijaya, August 2016, *Temperature Control Systems Using Fuzzy Method on Oyster Mushroom Dryer Device as alternative sun drying method*. Academic Supervisor : Dr.Ir. Bambang Siswojo, MT and Ir. Retnowati, MT.

With the time changing and technology advances in which everything can be done with an electronic device , it make people look for the easiest way to do that difficult work . One of them is automatic oyster mushroom dryer device. As we know the oyster mushroom has a very good nutrients for our body. For that reasons this research was invented. The device is expected to be able to cut the time needed for drying mushrooms so that it can increase production of dried oyster mushrooms .

This study used samples 500gr of mushrooms and 62-69 'C temperature, with a target of reducing the water content of the mushroom which was originally 90% to 4%, the value of the temperature and condition of the mushrooms will be displayed in Liquid Crystal Display (LCD) so will be easily to monitoring, is just by looking at the LCD. The test results of drying mushrooms by using fuzzy lasted 9 hours, where the system has a control parameter Time delay = 38 minutes, Time rise= 77 minutes, Time peak = 90 minutes. As result the drying process is much more time-efficient when compared with the manual process of drying in the sun that takes approximately two days.

Keywords : *Oyster Mushroom, Drying, controller, Fuzzy, temperature*

DAFTAR ISI

PENGANTAR

.....

i

RINGKASAN

.....

iii

SUMMARY

.....

iv

DAFTAR

ISI

.....

v

DAFTAR

GAMBAR

.....

vii

DAFTAR

TABEL

.....

viii

BAB I

PENDAHULUAN

.....

1

1.1 Latar Belakang

.....

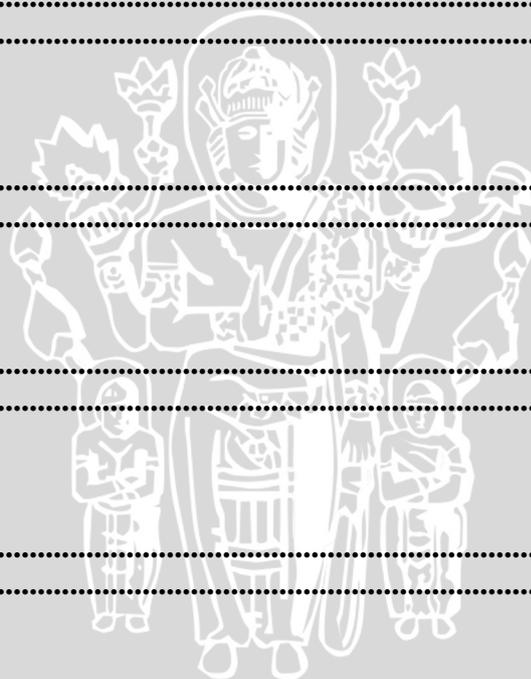
1.2 Rumusan Masalah

.....

1.3 Batasan Masalah

.....

2



1.4 Tujuan

2

1.5 Sistematika
Penulisan

3

BAB II TINJAUAN PUSTAKA

5

2.1 Karakteristik Jamur

Tiram

5

2.2 Pengeringan

6

2.3 Sensor Suhu

LM35

7

2.4 Load

Cell

9

2.5 Timer/ Counter

0

10

2.6 Sistem

Pemanas

10

2.7 LCD

11

2.8 Kontroler

13

2.8.1 Logika

Fuzzy

14

2.8.2 Fungsi Keanggotaan

Fuzzy

16

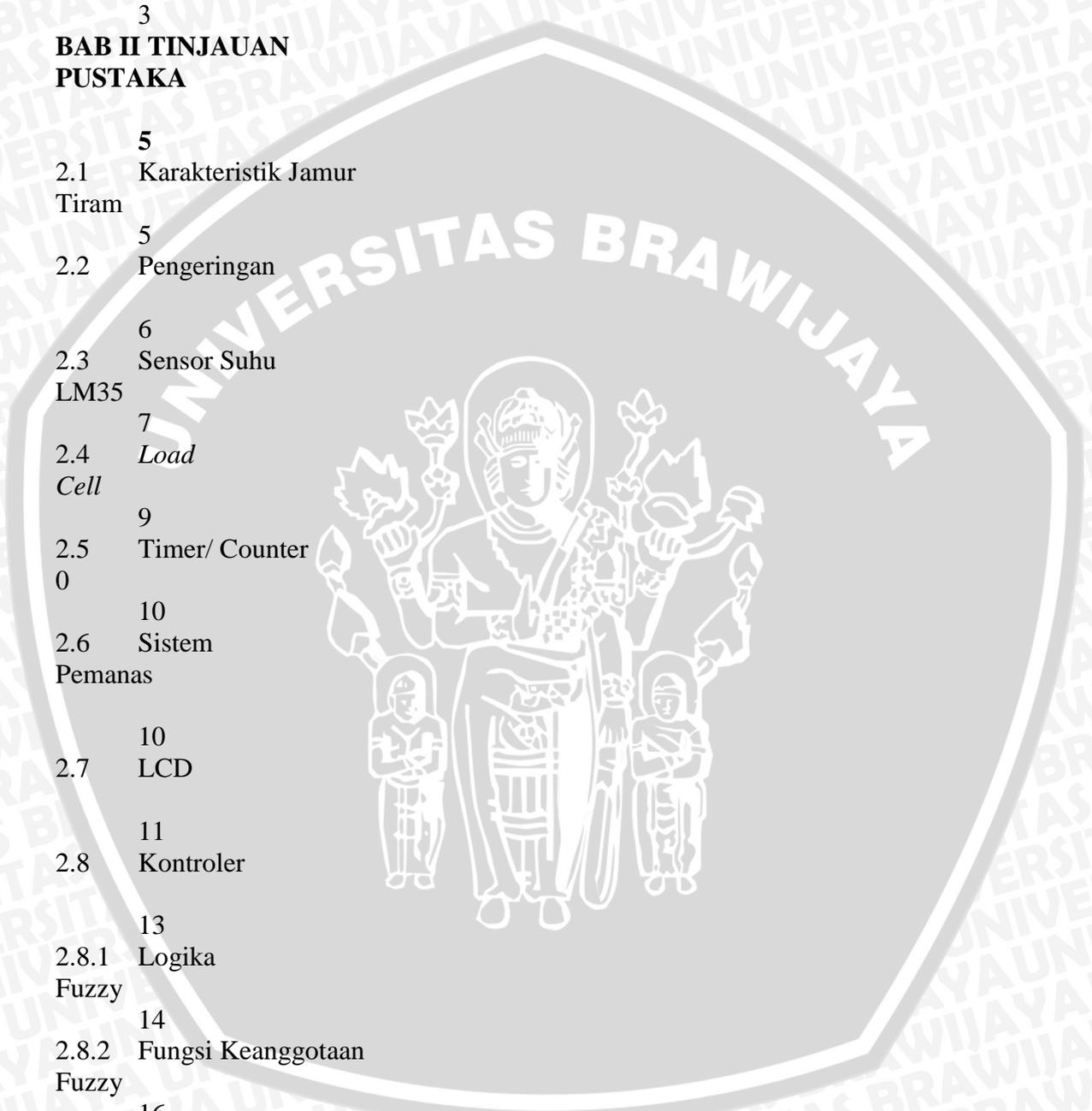
2.8.3 Sistem Inferensi

Fuzzy

18

2.8.4 Defuzzifikasi

19



2.8.5 Centroid
Method

19

2.9 Arduino
Uno

19

2.9.1 Daya

20

2.9.2 Memori

20

2.9.3 Input
Output

20

2.9.4 Komunikasi

21

BAB III METODOLOGI PENELITIAN

23

3.1 Penentuan Spesifikasi
Sistem

23

3.2 Studi
Literatur

24

3.3 Perancangan
Alat

24

3.3.1 Perancangan Perangkat
Keras

24

3.3.2 Perancangan Perangkat
Lunak

24

3.4 Pengujian
Sistem

25

BAB IV PERANCANGAN DAN PEMBUATAN ALAT

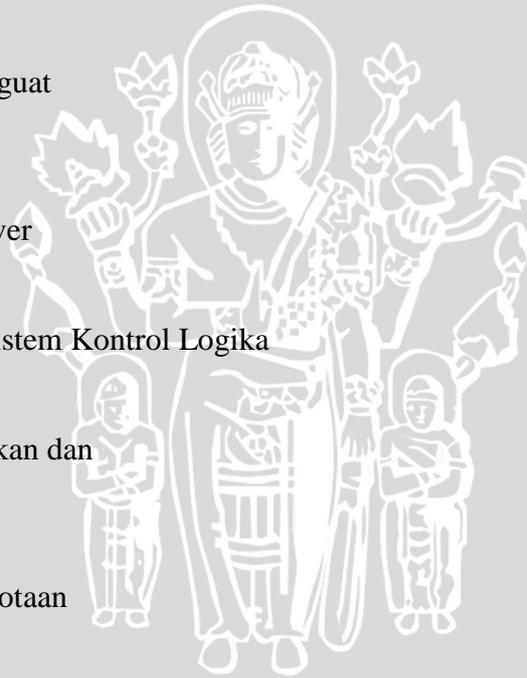
27

4.1 Penentuan Spesifikasi
Alat

27



4.2	Perancangan Sistem	28
4.3	Pronsip Kerja Alat	29
4.4	Perancangan Perangkat Keras	29
4.4.1	Rangkaian Sensor Suhu LM35	29
4.4.2	Rangkaian Penguat Tak Membalik	30
4.4.3	Rangkaian Load Cell	33
4.4.4	Rangkaian Penguat Instrumentasi	34
4.4.5	Rangkaian Driver Relay	37
4.5	Perancangan Sistem Kontrol Logika Fuzzy	39
4.5.1	Variabel Masukan dan Keluaran	39
4.5.2	Fungsi Keanggotaan Masukan	40
4.5.3	Fungsi Keanggotaan Keluaran	40
4.5.4	Perancangan Aturan Fuzzy	41
4.5.5	Metode Inferensi Min – Max	42



4.5.6	Defuzzifikasi	47
4.6	Perancangan Perangkat Lunak	49
4.6.1	Perancangan Program Utama	49
4.7	Sub Program Tampilan LCD	50
4.8	Sub Program Baca Suhu LM35	51
4.9	Sub Program Load Cell	52

BAB V PENGUJIAN DAN ANALISIS

		55
5.1	Pengujian Load Cell	55
5.1.1	Peralatan Pengujian	55
5.1.2	Prosedur Pengujian	55
5.1.3	Hasil Pengujian	56
5.2	Pengujian Penguat Instrumentasi INA125	57
5.2.1	Peralatan Pengujian	57
5.2.2	Prosedur Pengujian	



57

5.2.3 Hasil
Pengujian

57

5.3 Pengujian Sensor Suhu
LM35

59

5.3.1 Peralatan
Pengujian

59

5.3.2 Prosedur
Pengujian

59

5.3.3 Hasil
Pengujian

59

5.4 Pengujian Sensor Suhu LM35 Dengan Penguat Tak
Membalik

60

5.4.1 Peralatan
Pengujian

60

5.4.2 Prosedur
Pengujian

61

5.4.3 Hasil
Pengujian

61

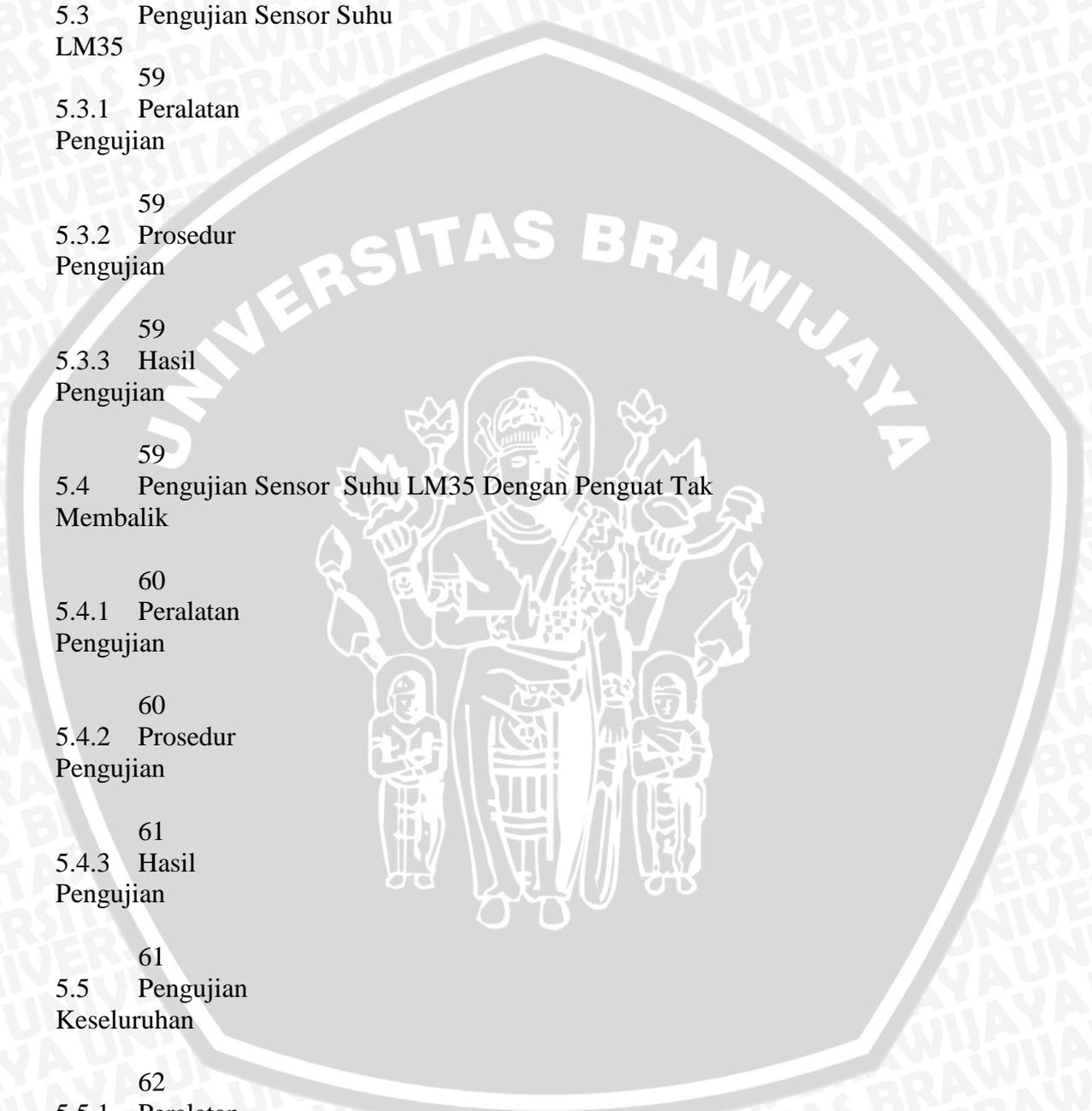
5.5 Pengujian
Keseluruhan

62

5.5.1 Peralatan
Pengujian

62

5.5.2 Prosedur
pengujian



62	5.5.3 Hasil Pengujian
----	-----------------------

63
BAB VI
PENUTUP

65	6.1 Kesimpulan
----	----------------

65	6.2 Saran
----	-----------

65
DAFTAR
PUSTAKA

66



DAFTAR GAMBAR

Gambar 2.1. Jamur Tiram (*Pleurotus ostreatus*).

6
Gambar 2.2. Sampel Jamur Tiram Segar dan Jamur Tiram Kering

7
Gambar 2.3. Sensor Suhu LM35

8
Gambar 2.4 Bentuk Fisis Load Cell

9
Gambar 2.5 Definisi Strain

10

Gambar 2.6 Ilustrasi Sistem Pemanas

11

Gambar 2.7 Tampilan Fisik LCD M6132 15x2 Karakter

13

Gambar 2.8. Fungsi keanggotaan Linear naik

16

Gambar 2.9. Reperentasi Linear Turun

17

Gambar 2.10 Kurva Segitiga

17

Gambar 2.11 Kurva trapesium

18

Gambar 2.12 Pencarian Centroid Method

19

Gambar 2.13 Tampak depan Arduino Uno

20

Gambar 4. 1 Blok diagram system

28

Gambar 4. 2 Rangkaian sensor suhu LM35 dan printed board

29

Gambar 4. 3 Konfigurasi Pin LM358

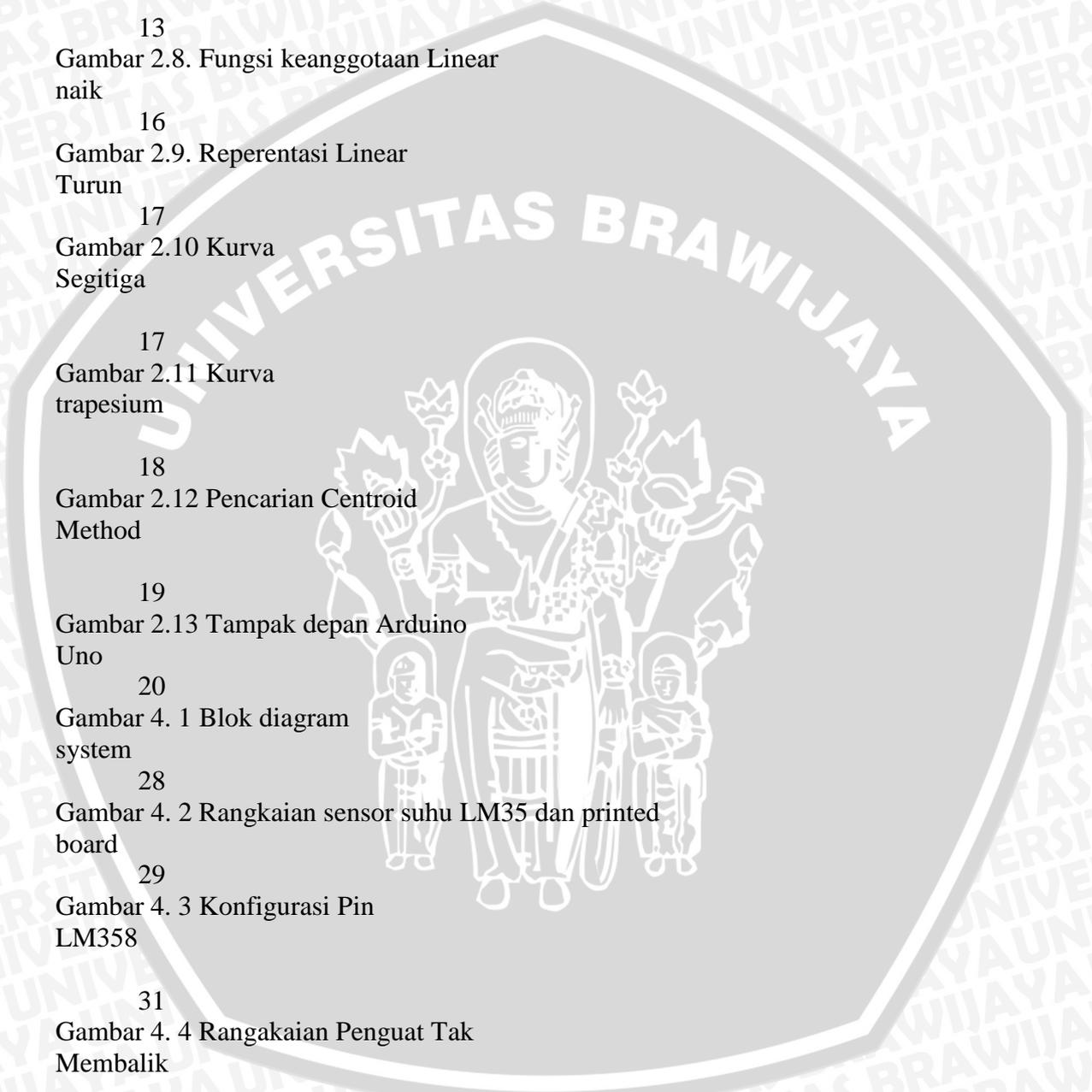
31

Gambar 4. 4 Rangkaian Penguat Tak Membalik

32

Gambar 4.5 Spesifikasi Input dan Output Pengkondisi Sinyal

33



Gambar 4.6 Rangkaian Sensor Load Cell

33

Gambar 4.7 Konfigurasi Pin INA125

34

Gambar 4. 8 Spesifikasi Input dan Output penguat instrumentasi

35

Gambar 4.9 Rangkaian Penguat Instrumentasi

36

Gambar 4. 10 Printed Board Blok Rangkaian Pengkondisi Sinyal

37

Gambar 4. 11 Rangkaian Driver Relay

37

Gambar 4. 12 Printed Board Rangkaian Driver Relay

38

Gambar 4.13 Fungsi Keanggotaan Masukan Error

40

Gambar 4.14 Fungsi Keanggotaan Masukan Delta Error

40

Gambar 4.15 Fungsi Keanggotaan Keluaran Suhu

41

Gambar 4.16 Inferensi Min-Max Saat Sample ke-1001

43

Gambar 4.17 Inferensi Min-Max Saat Sample ke-1002

44

Gambar 4.18 Inferensi Min-Max Saat Sample ke-1003

45

Gambar 4.19 Inferensi Min-Max Saat Sample ke-1004

46

Gambar 4.20 Inferensi Min-Max Saat Sample ke-1005

47

Gambar 4.21 Flowchart Program Sistem Keseluruhan

50

Gambar 4.22 Flowchart Program LCD

51

Gambar 4.23 Flowchart Baca suhu LM35

52

Gambar 4.24 Flowchart Baca Load Cell

53

Gambar 5.1 Grafik Tegangan Keluaran Terhadap Berat Pada Sensor Load Cell

56

Gambar 5.2 Grafik Tegangan Keluaran Terhadap Berat Pada Penguat Instrumentasi INA125

58

Gambar 5.3 Grafik Tegangan Terhadap Suhu Pada Sensor Suhu LM35

60

Gambar 5.4 Grafik Tegangan Terhadap Suhu Pada Rangkaian Penguat Tak Membalik

62

Gambar 5.5 Grafik Suhu Terhadap Waktu Pada Pengujian Keseluruhan

63

DAFTAR TABEL

Tabel 1 Konfigurasi PIN LCD M1632

12

Tabel 4.1 Konfigurasi Pin LM358

31

Tabel 4.2 Keterangan Fungsi Pin INA125

34

Tabel 4.3 Aturan Fuzzy 41

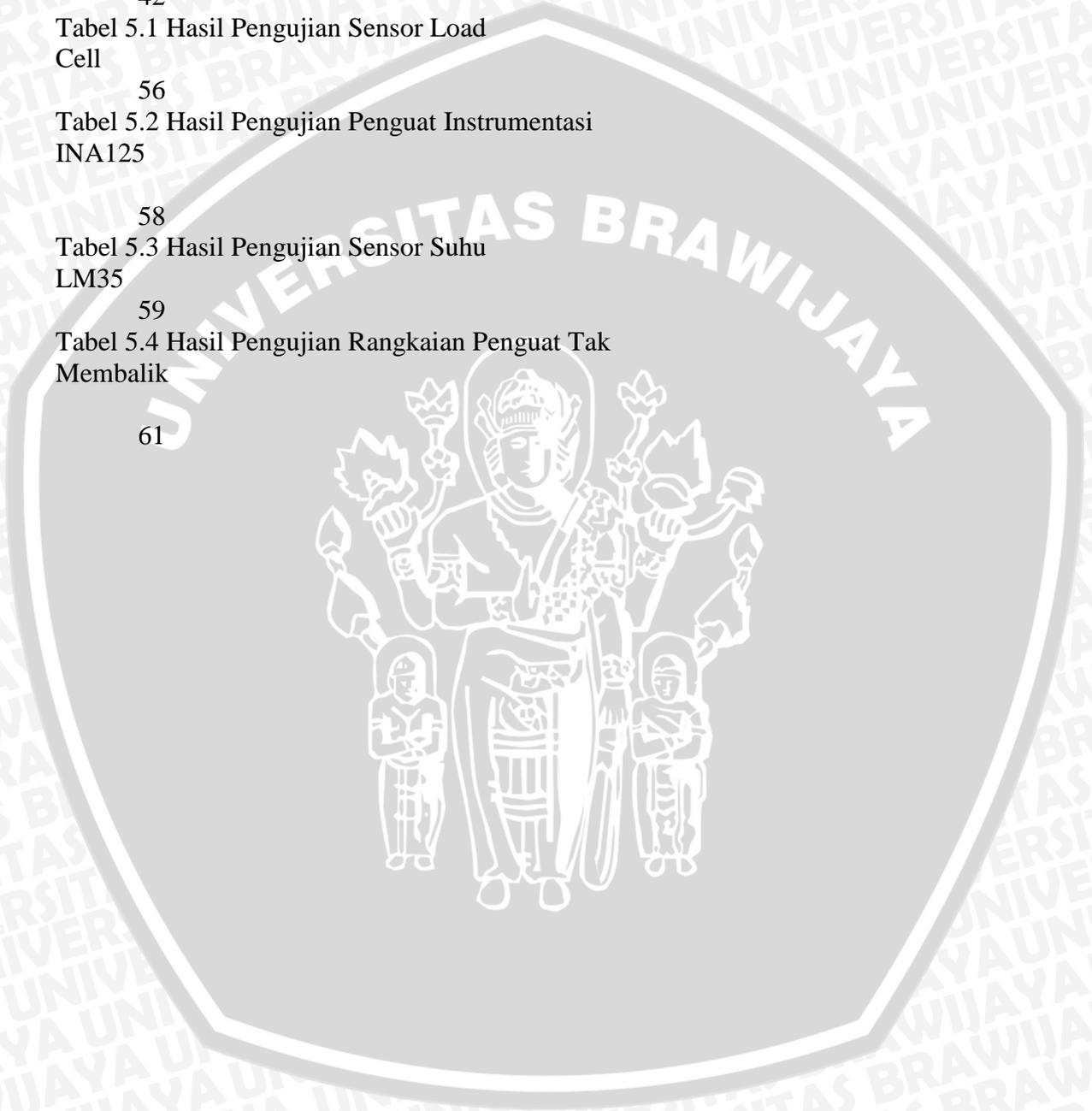
Tabel 4.4 set point dan keluaran 42

Tabel 5.1 Hasil Pengujian Sensor Load Cell 56

Tabel 5.2 Hasil Pengujian Penguat Instrumentasi INA125 58

Tabel 5.3 Hasil Pengujian Sensor Suhu LM35 59

Tabel 5.4 Hasil Pengujian Rangkaian Penguat Tak Membalik 61





BAB I PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan zaman yang semakin maju yang segalanya dapat dilakukan dengan alat elektronik, sehingga hal ini mendorong manusia untuk selalu mencari cara yang termudah untuk melakukan pekerjaan yang sulit. Salah satunya adalah inovasi yang akan diteliti kali ini yaitu Alat Pengering Jamur Tiram. Seperti yang kita pahami jamur tiram ini memiliki kandungan gizi yang sangat baik untuk tubuh kita diantaranya adalah protein, air, kalori, karbohidrat, dan sisanya berupa serat zat besi, kalsium, vitamin B1, vitamin B2, dan vitamin C. (Eger G, Eden G, Wissig E. 1976).

Melihat banyaknya kandungan zat pada jamur tiram yang baik untuk kesehatan tubuh manusia maka banyak olahan jamur yang kini telah tersedia dimana semua olahan tersebut ditujukan untuk mengenalkan jamur tiram kepada masyarakat. Untuk mengkonsumsi jamur tiram ini pun beragam ada yang mengolah jamur tiram ini seusai panen (masih basah) ataupun mengolah setelah jamur tiram dalam kondisi kering. Untuk mengolah jamur pada kondisi kering, jelas jamur harus dikeringkan terlebih dahulu biasanya hal ini dilakukan selain untuk olahan kripik jamur juga bertujuan untuk penyimpanan jamur yang lebih lama. Untuk mengeringkan jamur dapat menggunakan dua cara, yaitu dengan cara konvensional yaitu dengan menjemur jamur tiram di bawah sinar matahari dan dengan cara mengeringkan dengan menggunakan alat pengering. Permasalahan yang sering dikeluhkan oleh para pengusaha kripik jamur adalah jika menggunakan sinar matahari, jelas proses pengeringan harus menunggu musim panas tiba, karena pada saat musim hujan jamur tidak akan dapat kering secara sempurna mengingat jika matahari bersinar terik itupun membutuhkan waktu seharian agar jamur dapat kering sempurna. Jika proses pengeringan jamur tiram menggunakan alat pengering, maka diperlukan suatu alat yang mampu menjaga kestabilan suhu untuk proses pengeringannya pada suatu nilai tertentu sehingga diperoleh tingkat kekeringan yang sempurna.

Dari alasan diatas itulah dilakukan penelitian yang berjudul “Sistem Pengendalian Suhu Menggunakan Metode Fuzzy Pada Alat Pengering Jamur Tiram Sebagai Alternatif

Penjemuran Matahari”. Nilai dari suhu dan kondisi jamur akan ditampilkan dalam *Liquid Crystal Display* (LCD) sehingga pemantauan akan lebih mudah yaitu hanya dengan melihat dari LCD. Selain itu diharapkan alat dapat membantu para pengusaha jamur dalam menghemat waktu untuk pengeringan sehingga dapat meningkatkan produktifitas dan meningkatkan keuntungan dalam waktu yang terhitung singkat.

Pada penelitian kali ini mempunyai tujuan untuk mengetahui dan membandingkan karakteristik dan respon antara metode fuzzy dan metode yang telah digunakan sebelumnya. Dimana pada penelitian sebelumnya sudah menggunakan metode on-off dan juga PID.

1.2 RUMUSAN MASALAH

Mengacu pada permasalahan yang diuraikan pada latar belakang, maka rumusan masalah dapat ditekankan pada:

1. Bagaimana merancang dan membuat sistem pengendalian suhu pada alat pengering jamur tiram menggunakan metode fuzzy?
2. Bagaimana merancang *hardware* dan *software* sistem pengendalian suhu pada alat pengering jamur tiram menggunakan metode fuzzy?

1.3 BATASAN MASALAH

Mengacu pada permasalahan yang ada maka skripsi ini dibatasi pada:

1. Parameter keberhasilan alat adalah sesuai dengan spesifikasi alat yang digunakan.
2. Sample jamur sebesar 500 gram.
3. Parameter kadar air kering dan kadar air basah menggunakan referensi yang sudah ada.
4. Pembahasan mekanik, model sistematis, serta rangkaian elektronika tidak dibahas secara mendalam dan ditekankan lebih pada penggunaan metode fuzzy pada sistem.

1.4 TUJUAN

Penelitian ini bertujuan untuk merancang dan membuat suatu sistem pengendalian suhu pada alat pengering jamur tiram menggunakan metode fuzzy. sehingga nantinya dapat digunakan untuk mengeringkan jamur tiram dengan waktu yang lebih singkat dan tanpa bantuan sinar matahari sebagai sarana pengawetan dan untuk olahan makanan dengan bahan baku jamur tiram kering.

1.5 SISTEMATIKA PENULISAN

Sistematika penulisan dalam skripsi ini adalah sebagai berikut:

BAB I Pendahuluan

Memuat latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, dan sistematika pembahasan.

BAB II Dasar Teori

Membahas teori- teori yang mendukung dalam perencanaan dan pembuatan alat.

BAB III Metodologi Penelitian

Berisi tentang metode penelitian dan perencanaan alat serta pengujian.

BAB IV Perencanaan dan Pembuatan Alat

Perancangan dan perealisasiian alat yang meliputi spesifikasi, perencanaan blok diagram, prinsip kerja dan realisasi alat.

BAB V Pengujian Alat

Memuat hasil pengujian terhadap alat yang telah direalisasikan.

BAB VI Kesimpulan dan Saran

Berisi tentang kesimpulan berdasarkan hasil yang sudah diperoleh dari tujuan, perancangan, dan pengujian. Selain itu terdapat juga saran untuk melakukan pengembangan dan perbaikan terhadap alat



BAB II

TINJAUAN PUSTAKA

Dalam merencanakan dan merealisasikan sistem pada Alat Pengering Jamur Tiram dibutuhkan pemahaman yang mendalam mengenai teori- teori penunjang yang mendukung. Pemahaman ini akan bermanfaat dalam perencanaan dan perealisasiian perangkat keras dan perangkat lunak dari sistem. Teori-teori penunjang tersebut meliputi karakteristik jamur tiram, pengeringan, sensor suhu LM 35, *Loadcell*, LCD 16X2, Arduino UNO, *driver heater*, dan *driver blower*.

2.1 Karakteristik Jamur Tiram

Jamur tiram memiliki banyak jenis, salah satunya adalah jamur tiram putih. Jenis ini merupakan jenis yang paling banyak dikonsumsi pada umumnya selain karena mudah dibudidayakan, jamur ini juga termasuk sayuran yang mudah diolah dan dikonsumsi. Jamur ini memiliki ciri spesifik:

- Mempunyai spora berwarna putih.
- Batang atau stipes terletak di tepian tudung jamur.
- Lebar tudung 3 cm hingga 14 cm (Jaelani, 2008).

Jamur bersifat saprofit, yaitu penyerapan zat makanannya di serap dari limbah pertanian menurut Chang and Hayes, (1978) dalam Rachmat, E.A., (1997). Oleh karena itu, jamur tidak mempunyai akar, batang dan daun sejati dan termasuk dalam tumbuhan *Thallophyta*. Secara umum pertumbuhan jamur di bagi menjadi dua fase yaitu vegetatif dan generatif.

Jamur tiram ini, disamping mempunyai cita rasa dan tekstur yang spesifik juga mengandung asam amino yang cukup lengkap (Rismunandar, 1982 dalam Rachmat, E.A., 1997). Tubuh buah jamur *Pleurotus sp.* mengandung protein sebesar 30.4 %, lemak 2.2 %, karbohidrat 57.6 %, serat 8.7 %, dan abu 9.8 % per berat kering dengan kadar air sebesar 90.8%. Kandungan vitaminnya meliputi thiamin, riboflavin, dan niasin. Mineral yang dikandungnya antara lain: kalsium, fosfor, besi, natrium, dan kalium (Chang dan Hayes, 1978 dalam Rachmat, E.A., 1997). Gambar 2.1 menunjukkan karakteristik fisik jamur tiram berbentuk lebar.



Gambar 2.1. Jamur Tiram (*Pleurotus ostreatus*).

Sumber: Jaelani, 2008:17

Daya tarik jamur tiram adalah pada warna tubuh buahnya yang bervariasi, mulai putih (sehingga disebut tiram putih), kecoklatan, keabuan, kekuning-kuningan, kemerah-merahan. Nama tiram didasarkan warna tubuhnya (Suriawiria, 2010). Pengeringan pada jamur tiram umumnya masih sangat jarang, karena kandungan kadar air dalam jamur masih terbilang tinggi, yaitu mencapai 90%. Dari beberapa penelitian pengeringan jamur tentang kadar air menyebutkan bahwa kadar air jamur sebesar 89,41% (Kala, 2009) dan 90,12% (Tulek, 2011) oleh karena itu dalam pengeringan jamur harus seragam.

2.2 Pengeringan

Untuk pengeringan jamur tiram secara konvensional, di gunakan cara menjemur langsung di bawah terik matahari. Butuh waktu 2 hari untuk mendapatkan jamur kering. (Witi, 1990) Sehingga akan memakan waktu lama.

Untuk parameter pengeringan jamur, ada beberapa parameter yang sesuai agar hasil pengeringan didapatkan hasil yang baik. Diantaranya suhu pada saat pengeringan dan berat jamur setelah dan sebelum proses pengeringan. Secara normal jika dikeringkan dengan sinar matahari langsung didapatkan suhu sekitar 20 – 40°C, namun dalam penelitian ini digunakan suhu sebesar 65°C- 69°C, selama masih di bawah batas suhu maksimal yaitu 80 °C, dan di rekomendasikan 70 °C (Witi, 1990) dan literatur lain dikatakan Jamur yang akan dikeringkan dengan menggunakan oven akan baik jika menggunakan suhu diatas 50°C (Syah, 1993), sehingga dapat disimpulkan suhu optimum untuk pengeringan jamur adalah diantara 50 hingga 70 derajat celcius.

Sedangkan dari parameter kadar air, Steinkraus et al. (1985) menyatakan bahwa tingkat kadar air 2% sampai 8% atau 4% dari berat total sampel merupakan hasil akhir pengeringan yang aman dari risiko adanya pertumbuhan mikroorganisme kontaminan sehingga baik untuk penyimpanan maupun untuk diolah menjadi keripik jamur. Untuk parameter waktu dalam pengeringan jamur menggunakan pemanas optimalnya dilakukan tidak lebih dari 24 jam (Netty, 2004).

Dalam parameter kadar air nantinya akan didapatkan berat jamur yang diinginkan agar kadar air yang didapatkan sesuai dengan tingkat kadar air yang disarankan yaitu berkisar antara 2% hingga 8% atau sebesar 4% dari berat total sampel.

Gambar 2.2 menunjukkan struktur fisik jamur yang masih basah dan jamur yang telah kering. Terlihat bahwa jamur yang kering jelas memiliki massa yang lebih ringan karena adanya penyusutan kadar air.



Gambar 2.2. Sampel Jamur Tiram Segar dan Jamur Tiram Kering

Sumber: Jaelani, 2008:13

Untuk mendapatkan berat kering yang diinginkan sesuai parameter dari kadar air yang di anjurkan, perhitungan prosentase sampel bahan terlebih dahulu harus dilakukan. Dengan mengambil referensi data yang ada yaitu sebesar 4% atau $1/25$ nantinya akan dikalkulasikan dengan berat sampel bahan yang akan dikeringkan. Dengan melakukan perhitungan matematika sederhana maka parameter untuk berat bahwa jamur telah kering akan didapatkan.

Sedangkan untuk daya yang dibutuhkan dalam proses pengeringan, daya sebesar 300 watt sudah cukup dalam proses pengeringan jamur tiram. Namun semakin tinggi daya yang digunakan akan mempercepat respon pemanasan dalam ruang pengeringan.

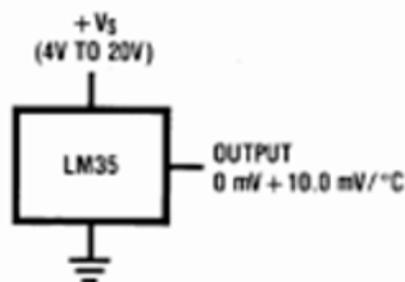
2.3 Sensor Suhu LM35

LM 35 sebagai sensor suhu yang teliti dan terkemas dalam bentuk Integrated Circuit (IC), di mana *output* tegangan keluaran sangat linear terhadap perubahan suhu. Sensor ini berfungsi sebagai pengubah dari besaran fisis suhu ke besaran tegangan yang memiliki koefisien sebesar $10 \text{ mV}/^{\circ}\text{C}$ yang berarti bahwa kenaikan suhu 1°C maka akan terjadi kenaikan tegangan sebesar 10 mV . IC LM 35 ini tidak memerlukan pengkalibrasian atau penyetelan dari luar karena ketelitiannya sampai lebih kurang seperempat derajat celsius pada temperatur ruang.

Sensor ini digunakan karena memiliki Jangkauan mulai dari -55°C sampai dengan 150°C , IC LM35 penggunaannya sangat mudah, difungsikan sebagai kontrol dari indikator

tampilan catu daya terbelah. IC LM 35 dapat dialiri arus 60 μ A dari supply sehingga panas yang ditimbulkan sendiri sangat rendah kurang dari 0 ° C di dalam suhu ruangan.

Untuk mendeteksi suhu digunakan sebuah sensor suhu LM35 yang dapat dikalibrasikan langsung dalam C (celcius). LM35 ini difungsikan sebagai sensor suhu, ditunjukkan dalam Gambar 2.3.



Gambar 2.3. Sensor Suhu LM35

Sumber: Texas Instrumen, 2002:2

V_{out} LM 35 ini dihubungkan dengan ADC (*Analog To Digital Converter*). Dalam suhu kamar (25°C) transduser ini mampu mengeluarkan tegangan 250mV dan 1,5V pada suhu 150°C dengan kenaikan sebesar 10mV/°C.

Selain itu IC LM35 memiliki keistimewaan sebagai berikut:

- Kalibrasi sudah dalam derajat celcius
- Linearitas +10 mV/°C
- Akurasi 0,5 °C pada suhu ruang
- Range +2 °C - 150 °C
- Dioperasikan dengan catu daya 4V - 30 V
- Arus yang mengalir kurang dari 30 μ A

Secara umum Rumus perbandingan besarnya tegangan yang dihasilkan dengan suhu yang terdeteksi oleh sensor adalah sebagai berikut:

$$\text{Suhu} = \frac{V}{10} \quad (1)$$

Keterangan:

Suhu = suhu yang terdeteksi oleh sensor LM35 (°C)

V= besar tegangan yang dihasilkan (mV)

Agar mikrokontroler mampu membaca suhu yang diterima oleh LM35 dengan tepat, maka besarnya suhu yang diterima oleh LM35 harus dikonversikan terlebih dahulu menggunakan persamaan berikut:

$$ADC = \left(\frac{V_{\text{terukur}}}{V_{\text{reff}}} \right) \times 255 \quad (2)$$

Keterangan:

ADC= nilai ADC yang terbaca mikrokontroler

V_{terukur} = tegangan yang dikeluarkan oleh sensor LM35

V_{reff} = tegangan *full scale* di mana sensor dalam keadaan suhu maksimum

Setelah ditemukan nilai ADC dengan rumus di atas, maka hasil suhu dapat ditentukan dengan persamaan berikut:

$$\text{Suhu} = \left(\frac{ADC}{255} \right) \times T_{\text{max}} \quad (3)$$

Keterangan:

T_{max} = suhu pembacaan maksimum dari sensor

ADC= nilai ADC yang terbaca mikrokontroler

2.4 Load Cell

Load cell merupakan divais yang menggunakan prinsip kerja jembatan *wheat stone*. Selain itu secara elektrik *load cell* menggunakan prinsip efek piezo resistif dengan bentuk fisik ditunjukkan dalam Gambar 2.4.



Gambar 2.4 Bentuk Fisis *Load Cell*

Sumber: Black, 1968:13

Prinsip kerja *load cell* adalah strain gauge. *Strain* adalah sejumlah deformasi pada material sebagai pengaruh dari aplikasi gaya. Lebih spesifik, *strain*(ϵ) didefinisikan sebagai perbandingan perubahan panjangnya, sebagaimana ditunjukkan dalam Gambar 2.5



Gambar 2.5 Definisi Strain

Sumber: Black, 1968:13

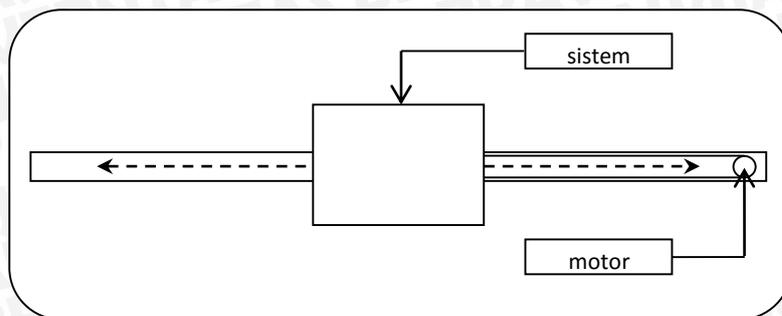
2.5 Timer/ Counter 0

Timer/counter 0 adalah sebuah *timer/counter* yang dapat mencacah sumber pulsa/clock baik dari dalam *chip* (*timer*) ataupun dari luar *chip* (*counter*) dengan kapasitas 8bi tatau 256 cacahan. *Timer/counter* dapat digunakan untuk :

- *Timer/counter* biasa
- *Clear Timer on Compare Match*
- Generator frekuensi
- *Counter* pulsa eksternal

2.6 Sistem Pemanas

Dalam perancangan ini untuk membuat suhu didalam pengering mengalami peningkatan dari suhu ruangan normal menjadi yang diinginkan dibutuhkan sebuah *heater*. *Heater* yang dipakai merupakan *heater* yang umumnya digunakan dalam seterika elektrik. *Heater* nantinya akan dihubungkan dengan rangkaian *zero crossing* untuk dapat dikontrol panas yang dihasilkan oleh *heater*. Kemudian *heater* akan dipasang dengan kipas yang berfungsi untuk menyebarkan panas ke seluruh ruangan. Prinsip yang digunakan hampir mirip dengan pemanas *hair dryer*. Dan juga sistem pemanas akan digerakkan motor untuk bergerak ke kanan dan kiri ruangan pemanas. Hal ini bertujuan agar sistem pemanasan jamur lebih merata. Ilustrasi sistem pemanas ditunjukkan dalam Gambar 2.6



Gambar 2.6 Ilustrasi Sistem Pemanas

Sumber: Dokumentasi, 2016

Sedangkan cara kerja *heater* adalah menggunakan prinsip termodinamika, yaitu menggunakan panas lalu dialirkan ke wilayah bertemperatur rendah agar menjadi lebih hangat. Perpindahan panas tersebut terjadi secara spontan, yang kita butuhkan hanyalah panas untuk dipindahkan ke temperatur yang lebih tinggi.

2.7 LCD (Liquid Crsytal Display)

LCD adalah suatu jenis media tampilan yang menggunakan kristal cair sebagai penampil utama. LCD bisa memunculkan Gambar atau tulisan dikarenakan terdapat banyak sekali titik cahaya (piksel) yang terdiri dari satu buah kristal cair sebagai sebuah titik cahaya. Meskipun disebut sebagai titik cahaya, namun kristal cair ini tidak memancarkan cahaya sendiri. Sumber cahaya di dalam sebuah perangkat LCD adalah lampu neon berwarna putih di bagian belakang susunan kristal cair tadi.

Titik cahaya yang jumlahnya puluhan ribu bahkan jutaan inilah yang membentuk tampilan citra. Kutub kristal cair yang dilewati arus listrik akan berubah karena pengaruh polarisasi medan magnetik yang timbul dan oleh karenanya akan membiarkan beberapa warna diteruskan sedangkan warna lainnya tersaring.

LCD banyak diaplikasikan untuk menampilkan karakter untuk membantu menginformasikan proses dan control. LCD yang digunakan adalah LCD tipe karakter karena LCD ini dapat menampilkan data. Keuntungan yang dapat diperoleh dengan menggunakan LCD adalah:

- 1) Dapat menampilkan karakter ASCII, sehingga memudahkan untuk program menampilkannya
- 2) Mudah dihubungkan dengan port I/O karena hanya menggunakan 4 bit data dan 3 bit control
- 3) Ukuran dari modul yang proporsional

Fungsi pin LCD M1632 ditunjukkan dalam Tabel 1.

Tabel 1 Konfigurasi PIN LCD M1632

NO	NAMA PIN	LEVEL	DESKRIPSI
1	VSS	0V	Ground
2	VDD	+5V	Tegangan suplai logika
3	VEE	VARIABEL	Tegangan kontras
4	RS	H/L	Register select, 0= perintah, 1= data
5	R/W	H/L	1= baca data, 0= tulis data
6	E	H/L	Enable clock
7	D0	H/L	Data bus 0
8	D1	H/L	Data bus 1
9	D2	H/L	Data bus 2
10	D3	H/L	Data bus 3
11	D4	H/L	Data bus 4
12	D5	H/L	Data bus 5
13	D6	H/L	Data bus 6
14	D7	H/L	Data bus 7
15	ANODE	4.2V- 4.6V	Tegangan positif backlight
16	KATODE	0V	Tegangan negative backlight

Sumber: delta-electronic.com/article/wp-content/.../09/an0034.pdf, 2013

Lcd tipe M1632 ini memiliki spesifikasi sebagai berikut:

1. LCD ini terdiri dari 32 karakter dengan 2 baris masing- masing 16 karakter dengan display dot matrik
2. Karakter generator ROM dengan 192 tipe karakter
3. Karakter generator RAM dengan 8 bit karakter
4. Data RAM80x8 bit display
5. Dapat diinterfacekan ke MCU 8 atau 4
6. Dilengkapi fungsi tambahan antara lain display clear, cursor home, display on/off, display character blink, cursor shift, display shift
7. Internal data

8. Internal otomatis, reset pada saat power ON
9. Tegangan +5v PSU tunggal

Modul peraga yang digunakan dalam perancangan ini adalah LCD modul M1632. Gambar 2.7 menunjukkan tampilan fisik LCD M6132.



Gambar 2.7 Tampilan Fisik LCD M6132 15x2 Karakter
Sumber: Dokumentasi, 2013

Modul LCD ini membutuhkan daya yang kecil dan dilengkapi dengan panel LCD dengan tingkat kontras yang cukup tinggi serta pengendali LCD CMOS yang terpasang dalam modul tersebut. Pengendali mempunyai pembangkit karakter ROM/RAM dan display dan RAM. Semua fungsi display diatur oleh instruksi- instruksi, sehingga modul LCD ini dapat dengan mudah dihubungkan dengan unit mikroprosesor. LCD tipe ini tersusun sebanyak dua baris dengan 16 karakter masing barisnya.

2.8 Kontroler

Sistem pengendalian dirancang untuk melakukan dan menyelesaikan tugas tertentu. Syarat utama sistem pengendalian adalah harus stabil. Di samping kestabilan mutlak, maka sistem harus memiliki kestabilan secara relatif, yakni tolak ukur kualitas kestabilan sistem dengan menganalisis sampai sejauh mana batas-batas kestabilan sistem tersebut jika dikenai gangguan (Ogata K.,1997). Selain itu analisis juga dilakukan untuk mengetahui bagaimana kecepatan sistem dalam merespons *input*, dan bagaimana peredaman terhadap adanya lonjakan (*over shoot*).

Suatu sistem dikatakan stabil jika diberi gangguan maka sistem tersebut akan kembali ke keadaan *steady state* di mana *output* berada dalam keadaan tetap seperti tidak ada gangguan. Sistem dikatakan tidak stabil jika *outputnya* berosilasi terus menerus ketika dikenai suatu gangguan. Karena suatu sistem pengendalian biasanya melibatkan penyimpanan energi maka *output* sistem ketika diberi suatu *input*, tidak dapat mengikuti *input* secara serentak, tapi menunjukkan respons transien berupa suatu osilasi teredam sebelum mencapai *steady state*.

Dalam sistem pengendalian terdapat 2 macam *loop*:

1. Pengendalian dengan *loop* terbuka

Sistem kontrol *loop* terbuka adalah sistem kontrol yang keluarannya tidak berpengaruh pada aksi pengontrolan. Jadi pada sistem kontrol *loop* terbuka, keluaran tidak diukur atau diumpan balikan untuk dibandingkan dengan masukan.

2. Pengendalian dengan *loop* tertutup

Sistem kontrol *loop* tertutup adalah sistem kontrol yang keluarannya mempunyai pengaruh langsung pada aksi pengontrolan. Disebut juga sistem kontrol yang menggunakan umpan balik untuk memperkecil kesalahan sistem.

2.8.1 Logika Fuzzy

Sistem *fuzzy* ditemukan pertama kali oleh Prof. Lotfi Zadeh pada pertengahan tahun 1965. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*. Pada teori himpunan *fuzzy*, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting.

Nilai keanggotaan atau derajat keanggotaan atau *membership function* menjadi ciri utama dalam penalaran dengan logika *fuzzy* tersebut. Sistem ini diciptakan karena Boolean *logic* tidak memberikan ketelitian yang tinggi, hanya mempunyai logika 0 dan 1 saja. Sehingga untuk membuat sistem yang mempunyai ketelitian yang tinggi, maka dapat digunakan Boolean *logic* (Fauziah, 2009).

Teori himpunan *fuzzy* merupakan perluasan dari himpunan klasik (*crisp*). Pada himpunan klasik (*crisp*), nilai keanggotaan suatu (x) dalam suatu himpunan (A), sering dikenal dengan derajat keanggotaan, dinotasikan dengan $\mu_A[x]$, memiliki 2 kemungkinan (Fauziah, 2009) :

1. Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan
2. Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

Ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy* :

- a. Variabel *fuzzy*

Variable *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*. Contoh: umur, temperature, permintaan, dan sebagainya.

b. Himpunan *fuzzy*

Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*. Himpunan *fuzzy* memiliki 2 atribut, yaitu :

1. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: DINGIN, NORMAL, PANAS.
2. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti: 40, 25, 50.

c. Semesta Pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan real yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Ada kalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya. Contoh : semesta pembicaraan untuk variabel temperatur : $[0 \ 40]$

d. Domain

Domain himpunan *fuzzy* adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan real yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif dan bilangan negatif. Contoh domain himpunan *fuzzy* :

Sedikit = $[0 \ 15]$

Sedang = $[5 \ 25]$

Banyak = $[25 \ 45]$

Ada beberapa alasan mengapa orang menggunakan logika *fuzzy* (Fauziah, 2009), yaitu :

1. Konsep logika *fuzzy* mudah dimengerti. Konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
2. Logika *fuzzy* sangat fleksibel.
3. Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.
4. Logika *fuzzy* mampu memodelkan fungsi-fungsi nonlinear yang sangat kompleks.

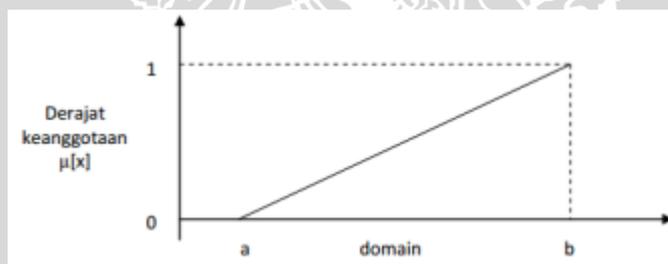
5. Logika *fuzzy* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
6. Logika *fuzzy* dapat bekerjasama dengan teknik-teknik kendali konvensional.
7. Logika *fuzzy* didasarkan pada bahasa alami.

2.8.2. Fungsi Keanggotaan Fuzzy

Fungsi keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1 (Fauziah, 2009). Ada beberapa fungsi yang bisa digunakan untuk membuat fungsi keanggotaan *fuzzy*, yaitu :

a. Fungsi Keanggotaan Linear

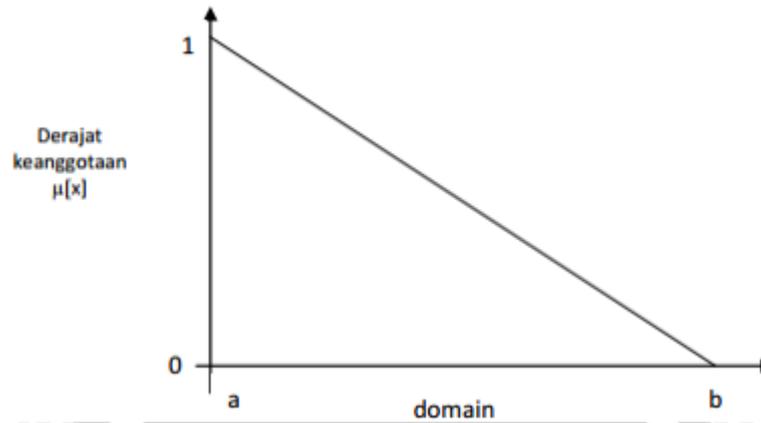
Pada fungsi keanggotaan linear, pemetaan *input* ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Bentuk ini paling sederhana dan menjadi pilihan yang baik untuk mendekati suatu konsep yang kurang jelas. Gambar 2.8 menunjukkan Fungsi keanggotaan linear naik.



Gambar 2.8. Fungsi keanggotaan Linear naik

$$\mu[x] = \begin{cases} 0; & x \leq a \\ (x - a) / (b - a); & a < x < b \\ 1; & x \geq b \end{cases}$$

Kedua, merupakan kebalikan dari yang pertama. Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah. Gambar 2.9 menunjukkan representasi linear turun.



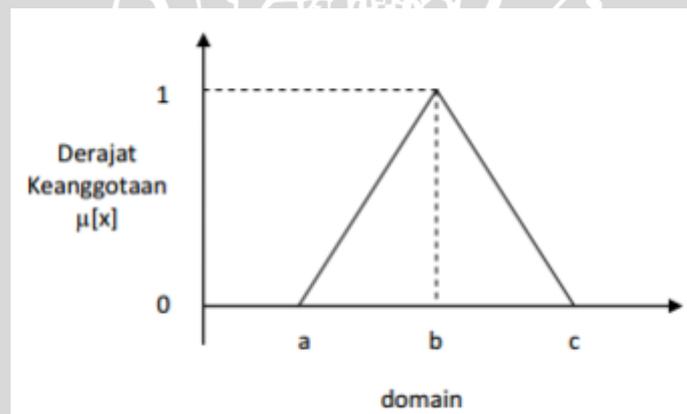
Gambar 2.9. Reperentasi Linear Turun

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} 0; & x \geq b \\ (b - x)/(b - a); & a < x < b \\ 1; & x \leq a \end{cases}$$

b. Fungsi Keanggotaan Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linear). Gambar 2.10 menunjukkan kurva segitiga



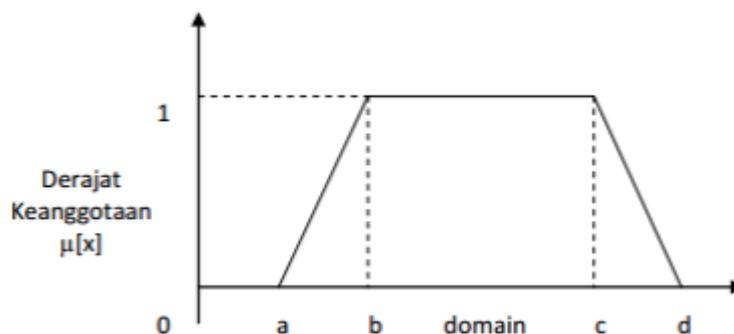
Gambar 2.10 Kurva Segitiga

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} 0; & x \geq c \text{ atau } x \leq a \\ (x - a)/(b - a); & a < x < b \\ (c - x)/(c - b); & b < x < c \end{cases}$$

c. Fungsi Keanggotaan Trapezium

Kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada titik yang memiliki nilai keanggotaan 1. Gambar 2.11 menunjukkan kurva trapezium.



Gambar 2.11 Kurva trapesium

Fungsi keanggotaan :

$$\mu[x] = \begin{cases} 0; & x \geq d \text{ atau } x \leq a \\ (x - a) / (b - a); & a < x < b \\ (d - x) / (d - c); & c < x < d \\ 1; & b \leq x \leq c \end{cases}$$

2.8.3. Sistem Inferensi Fuzzy

Sistem inferensi *fuzzy* diartikan sebagai penarikan kesimpulan dari sekumpulan aturan atau kaidah *fuzzy* yang didasarkan pada teori himpunan *fuzzy*, aturan *fuzzy* berbentuk IF-THEN, dan penalaran yang memiliki masukan dan keluaran berupa *crisp value*.

Masukan yang berupa *crisp value* diproses ke dalam himpunan *fuzzy* dan ditentukan derajat keanggotaannya di dalam himpunan *fuzzy* tersebut. Selanjutnya nilai dicari derajat kebenarannya menggunakan operasi logika AND, OR, ataupun NOT. Kemudian nilai akan dibandingkan dengan IF-THEN *rule* yang ada.

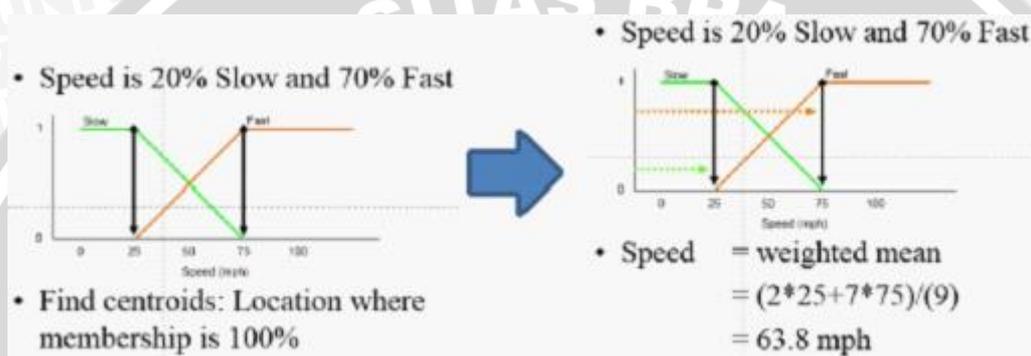
Sehingga didapatkan keluaran yang bisa saja lebih dari satu. Jika keluaran lebih dari satu maka akan dilakukan proses agregasi atau kombinasi dua nilai atau lebih menggunakan metode konjungsi maupun disjungsi menjadi keluaran tunggal. Hasil keluaran tunggal tersebut dikembalikan ke *crisp value* sehingga didapatkan keluaran yang sesungguhnya

2.8.4. Defuzzifikasi

Defuzzifikasi diartikan sebagai proses perubahan keluaran *fuzzy* menjadi *crisp value* sesuai dengan fungsi keanggotaan yang ditentukan. Banyak cara untuk melakukan proses ini, salah satunya adalah menggunakan centroid method.

2.8.5. Centroid Method

Metode ini digunakan dengan cara mencari *centroid* atau pusat dari sebuah daerah *fuzzy*. Pada metode ini nilai keluarannya diperoleh dari pengambilan titik yang memiliki nilai derajat keanggotaan maksimal atau bernilai 100% yang kemudian dicari nilai *weighted mean* dari titik-titik tersebut. Pencarian tersebut ditunjukkan oleh Gambar 2.12.



Gambar 2.12 Pencarian *Centroid Method*

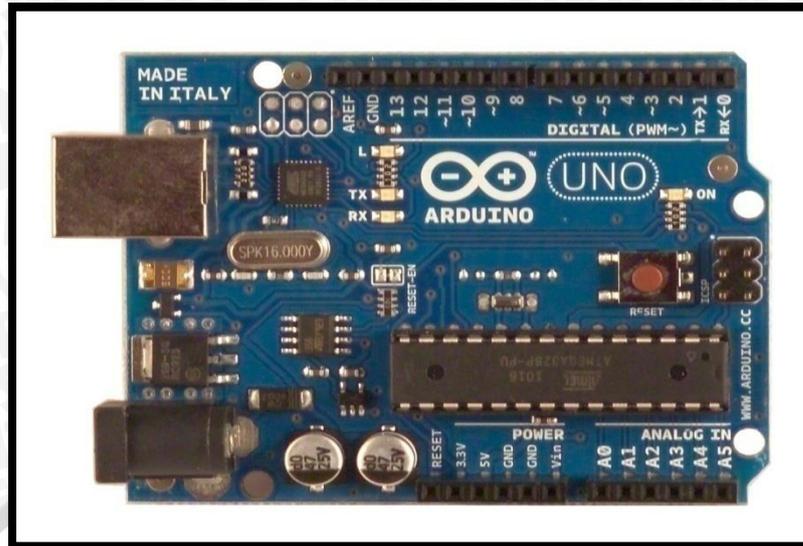
Sumber : (Nelson, 2014)

2.9 Arduino Uno

Arduino Uno adalah *board* mikrokontroler berbasis ATmega328. Memiliki 14 pin *input* dari *output* digital di mana 6 pin *input* tersebut dapat digunakan sebagai *output* PWM dan 6 pin *input* analog, 16 MHz osilator kristal, koneksi USB, *jack power*, ICSP header, dan tombol reset. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan Board Arduino Uno ke komputer dengan menggunakan kabel USB atau listrik dengan AC yang-ke adaptor-DC atau baterai untuk menjalankannya.

Uno berbeda dengan semua board sebelumnya dalam hal koneksi USB-to-serial yaitu menggunakan fitur Atmega8U2 yang diprogram sebagai konverter USB-to-serial berbeda dengan board sebelumnya yang menggunakan chip FTDI driver USB-to-serial.

Gambar 2.13 merupakan bentuk fisik dari Arduino Uno.



Gambar 2.13 Tampak depan Arduino Uno

Nama “Uno” berarti *satu* dalam bahasa Italia, untuk menandai peluncuran Arduino 1.0. Uno dan versi 1.0 akan menjadi versi referensi dari Arduino. Uno adalah yang terbaru dalam serangkaian board USB Arduino, dan sebagai model referensi untuk platform Arduino, untuk perbandingan dengan versi sebelumnya.

2.9.1 Daya

Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal (otomatis). Eksternal (non-USB) daya dapat berasal baik dari AC-ke adaptor-DC atau baterai. Adaptor ini dapat dihubungkan dengan menancapkan *plug jack* pusat-positif ukuran 2.1mm konektor POWER. Ujung kepala dari baterai dapat dimasukkan kedalam Gnd dan Vin pin *header* dari konektor POWER. Kisaran kebutuhan daya yang disarankan untuk board Uno adalah 7 sampai dengan 12 volt, jika diberi daya kurang dari 7 volt kemungkinan pin 5v Uno dapat beroperasi tetapi tidak stabil kemudian jika diberi daya lebih dari 12V, regulator tegangan bisa panas dan dapat merusak *board* Uno.

2.9.2 Memori

ATmega328 memiliki 32 KB (dengan 0,5 KB digunakan untuk bootloader), 2 KB dari SRAM dan 1 KB EEPROM

2.9.3 Input Output

Masing-masing dari 14 pin digital di Uno dapat digunakan sebagai *input* atau *output*, dengan menggunakan fungsi *pinMode ()*, *digitalWrite ()*, dan *digitalRead ()*, beroperasi

dengan daya 5 volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki internal *pull-up* resistor (secara *default* terputus) dari 20-50 kOhms. Selain itu, beberapa pin memiliki fungsi khusus:

- *Serial*: 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data serial. Pin ini dihubungkan ke pin yang berkaitan dengan chip Serial ATmega8U2 USB-to-TTL.
- *Eksternal menyela*: 2 dan 3. Pin ini dapat dikonfigurasi untuk memicu *interrupt* pada nilai yang rendah, dengan batasan tepi naik atau turun, atau perubahan nilai. *PWM*: 3, 5, 6, 9, 10, dan 11. Menyediakan *output* PWM 8-bit dengan fungsi *analogWrite* ().
- *SPI*: 10 (SS), 11 (Mosi), 12 (MISO), 13 (SCK). Pin ini mendukung komunikasi SPI menggunakan *SPI library*.
- *LED*: 13. Ada *built-in* LED terhubung ke pin digital 13. Ketika pin bernilai nilai HIGH, LED on, ketika pin bernilai LOW, LED *off*. Uno memiliki 6 masukan analog, berlabel A0 sampai dengan A5, yang masing-masing menyediakan 10 bit dengan resolusi (yaitu 1024 nilai yang berbeda). Selain itu, beberapa pin memiliki fungsi khusus:
- *I2C*: A4 (*SDA*) dan A5 (*SCL*). Dukungan I2C (TWI) komunikasi menggunakan perpustakaan Wire.
- *Aref*. Tegangan referensi (0 sampai 5V saja) untuk *input* analog. Digunakan dengan fungsi *analogReference* ().
- *Reset*. Bawa baris ini LOW untuk me-reset mikrokontroler.

2.9.4 Komunikasi

Arduino Uno memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya. ATmega328 menyediakan UART TTL (5V) untuk komunikasi serial, yang tersedia di pin digital 0 (RX) dan 1 (TX). Sebuah ATmega8U2 sebagai saluran komunikasi serial melalui USB dan sebagai *port virtual com* untuk perangkat lunak pada komputer. Firmware '8 U2 menggunakan *driver* USB standar COM, dan tidak ada *driver* eksternal yang diperlukan. Namun, pada *Windows* diperlukan, sebuah *file inf*. Perangkat lunak Arduino terdapat monitor serial yang memungkinkan digunakan memonitor data tekstual sederhana yang akan dikirim ke atau dari board Arduino. LED RX

dan TX di papan tulis akan berkedip ketika data sedang dikirim melalui chip USB-to-serial dengan koneksi USB ke komputer (tetapi tidak untuk komunikasi serial pada pin 0 dan 1.

Sebuah *Software Serial library* memungkinkan untuk berkomunikasi secara serial pada salah satu pin digital pada *board* Uno's. ATmega328 juga mendukung I2C (TWI) dan komunikasi SPI. Perangkat lunak Arduino termasuk perpustakaan Kawat untuk menyederhanakan penggunaan bus I2

BAB III

METODE PENELITIAN

Penyusunan penelitian ini berdasarkan pada masalah yang bersifat aplikatif, yaitu perencanaan dan perealisasiannya agar dapat bekerja sesuai dengan yang direncanakan dengan mengacu pada rumusan masalah.

Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang akan dibuat adalah sebagai berikut:

3.1 Penentuan Spesifikasi Sistem

Penentuan spesifikasi alat ini, bertujuan agar dapat dibuat sesuai yang diinginkan dan dapat bekerja dengan efektif serta efisien. Alat yang dirancang memiliki spesifikasi sebagai berikut:

- 1) Alat mampu mengeringkan jamur tiram dengan waktu lebih singkat dibandingkan menggunakan sinar matahari, dengan berat kering menjadi 4% dari berat sampel.
- 2) Menggunakan sensor suhu untuk mengatur suhu sesuai dengan anjuran pengeringan jamur tiram.
- 3) Menggunakan sensor berat (*load cell*) untuk mengetahui parameter jamur telah kering.
- 4) Mikrokontroler yang digunakan adalah Arduino Uno
- 5) Menggunakan *Heater* sebagai penghasil panas pada alat yang menghasilkan suhu 65°C - 69°C
- 6) Menggunakan kipas DC 5V sebagai *blower* yang digunakan untuk meratakan panas dalam ruang.
- 7) Bentuk fisik alat berupa oven konvensional dengan komponen-komponen pendukung agar alat dapat bekerja dengan baik.
- 8) Sampel berat jamur tiram sebesar 500 gr, dengan hasil akhir setelah kering sebesar 4% dari berat sampel.

3.2 Studi Literatur

Dalam penyusunan karya tulis ini, pengumpulan data dilakukan dengan melakukan studi literatur (*library research*), penelusuran informasi digital, dan wawancara narasumber dengan sasaran tinjauan antara lain:

- 1) Informasi internet.
- 2) Pustaka-pustaka referensi.
- 3) Pustaka penunjang.

Studi literatur yang dilakukan bertujuan untuk mengkaji hal-hal yang berhubungan dengan teori-teori yang mendukung dalam perencanaan dan perealisasi alat. Teori-teori yang dikaji adalah sebagai berikut:

- 1) Teori mengenai rangkaian sensor yang berfungsi merubah besaran fisik menjadi besaran elektrik.
- 2) Teori umum mengenai Jamur Tiram
- 3) Teori umum mengenai sensor suhu LM 35.
- 4) Teori umum mengenai sensor berat *Load cell*.
- 5) Teori umum mengenai LCD 16x2 karakter
- 6) Teori mengenai rangkaian sensor Suhu LM35
- 7) Teori mengenai rangkaian LCD 16x2 karakter
- 8) Sistem Pemrograman Arduino UNO

3.3 Perancangan Alat

Dalam perancangan membahas tentang diagram blok, perencanaan perangkat keras, cara kerja sistem dan perencanaan perangkat lunak.

3.3.1 Perencanaan Perangkat Keras

Perencanaan perangkat keras meliputi meliputi perencanaan rangkaian sensor LM35, rangkaian *Load Cell*, blok rangkaian pengkondisi sinyal, rangkaian sistem *heater*, rangkaian *system blower*, dan rangkaian Arduino UNO.

3.3.2 Perencanaan Perangkat Lunak

Perancangan perangkat lunak dibutuhkan mikrokontroler dan laptop/ *Unit Personal Computer* untuk mengendalikan perangkat keras. Perancangan perangkat lunak dilakukan dengan cara membuat *listing program* untuk mikrokontroler yang dilakukan dengan

pembuatan *flowchart* terlebih dahulu kemudian pembuatan programnya. Bahasa pemrograman yang dipakai mikrokontroler adalah pemrograman bahasa C.

3.4 Pengujian Sistem

Untuk menganalisis kinerja alat apakah sesuai dengan yang direncanakan maka dilakukan pengujian rangkaian. Pengujian dilakukan pada masing-masing blok dan kemudian secara keseluruhan sistem. Pengujian alat dilakukan sebagai berikut:

1) Pengujian Perangkat Keras

a. Pengujian Sensor Suhu LM 35

Pengujian ini dilakukan untuk mengetahui respon sensor terhadap suhu sebelum dikuatkan. Pengujian dilakukan dengan membaca suhu di dalam suatu wadah berisi air.

b. Pengujian Sensor Berat Load Cell

Pengujian ini dilakukan untuk melihat respon sensor sebelum dan setelah dikuatkan dengan mencoba pembacaan suatu sample dan dibandingkan dengan pembacaan sample contoh dengan timbangan lain.

c. Pengujian ADC

Pengujian ADC dilakukan untuk mengetahui apakah ADC dapat bekerja dengan baik untuk mengkonversi sinyal analog dari sensor LM 35 dan sensor berat Load Cell menjadi sinyal digital. Dalam pengujian ini digunakan ADC internal yang terdapat dalam mikrokontroler AVR ATMEGA 8.

d. Pengujian Penguat Tak Membalik

Pengujian dilakukan untuk melihat hasil tegangan dari *output* LM 35 yang dikuatkan oleh penguat pembalik agar sesuai dengan spesifikasi yang diinginkan.

e. Pengujian Penguat Instrumentasi INA125

Pengujian dilakukan untuk melihat hasil tegangan dari *output load cell* setelah dikuatkan oleh penguat instrumentasi INA125 agar sesuai dengan spesifikasi tegangan yang diinginkan.

f. Pengujian keseluruhan system

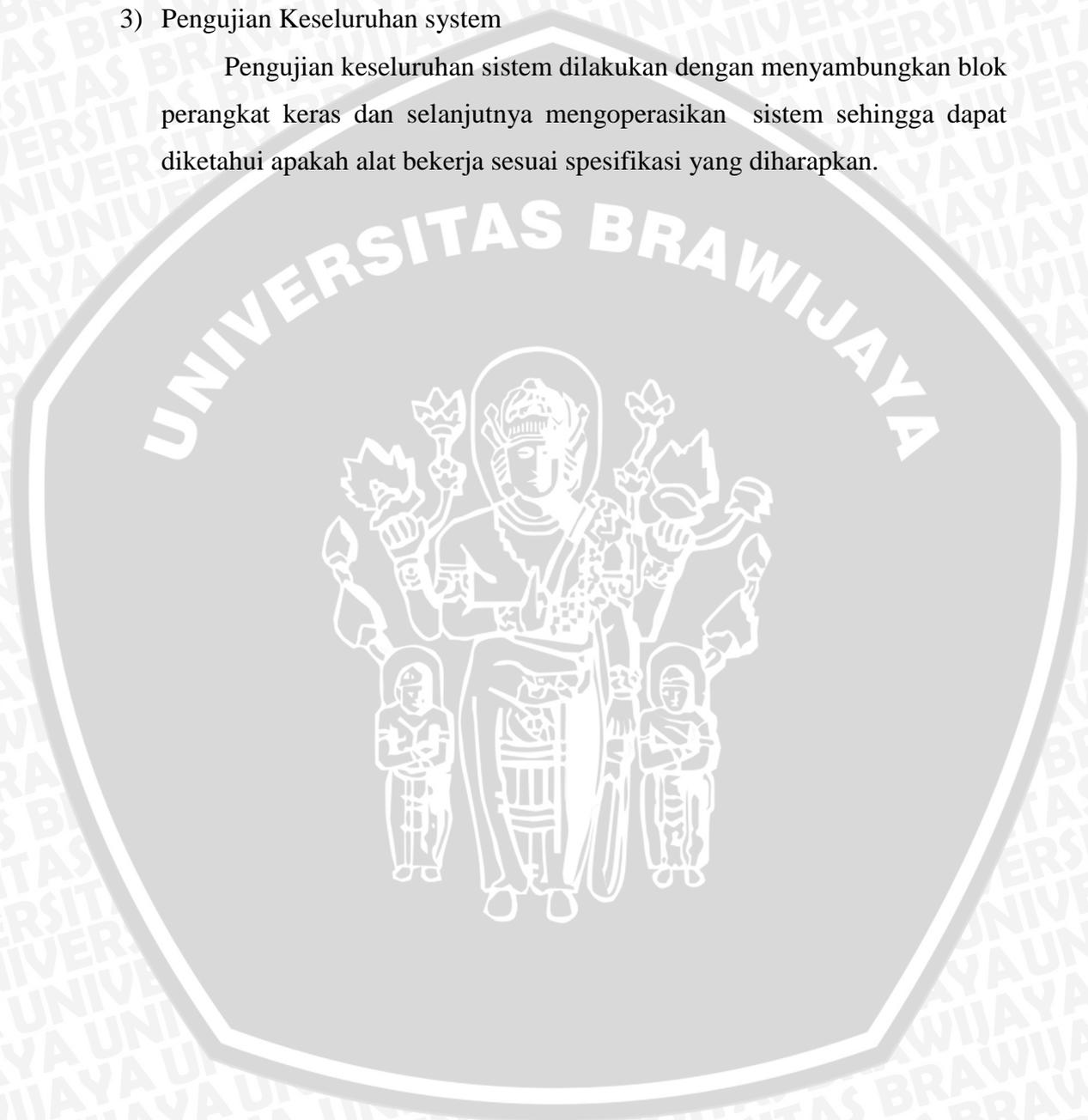
Pengujian ini bertujuan untuk mengetahui apakah *system* yang telah dibuat telah berjalan dengan baik.

2) Pengujian Perangkat Lunak

Untuk pengujian perangkat lunak diterapkan langsung terhadap rangkaian perangkat keras yang telah dirancang dan direalisasikan sebelumnya. Pengujian ini dilakukan untuk mengetahui apakah perangkat lunak mampu menangani perangkat keras yang ada.

3) Pengujian Keseluruhan system

Pengujian keseluruhan sistem dilakukan dengan menyambungkan blok perangkat keras dan selanjutnya mengoperasikan sistem sehingga dapat diketahui apakah alat bekerja sesuai spesifikasi yang diharapkan.



BAB IV

PERANCANGAN DAN PEMBUATAN ALAT

Dalam bab ini akan dibahas cara perancangan dan pembuatan Alat Pengering Jamur Tiram Sebagai Alternatif Penjemuran Matahari dengan menggunakan metode *fuzzy*. Perancangan alat meliputi perancangan elektrik dan perancangan software.

Perancangan perangkat keras terdiri dari perancangan rangkaian catu daya, perancangan rangkaian sensor suhu, perancangan rangkaian sensor load cell, perancangan rangkaian driver relay, perancangan blok rangkaian pengkondisi sinyal dan blok mikrokontroler (Arduino Uno). Sedangkan untuk perancangan software meliputi program untuk arduino uno.

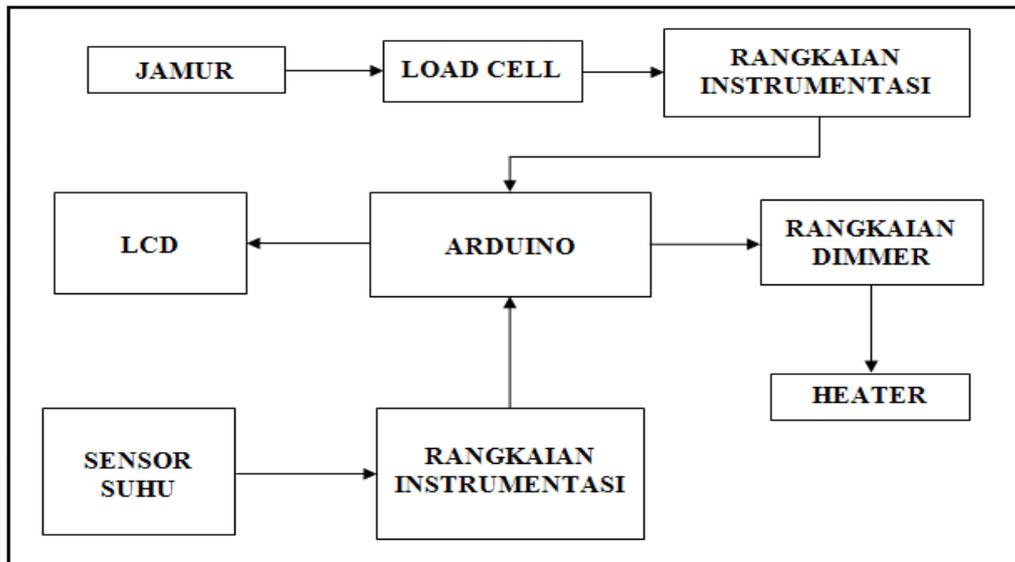
4.1 Penentuan Spesifikasi Alat

Dalam menyusun rangkaian alat perlu ditentukan spesifikasi alat terlebih dahulu. Alat yang dirancang memiliki spesifikasi sebagai berikut:

- 1) Menggunakan sensor suhu LM35 untuk memonitoring suhu sesuai dengan anjuran pengeringan jamur tiram
- 2) Menggunakan sensor berat (*load cell*) dengan kapasitas *maximum* 2kg. untuk mengetahui parameter jamur telah kering.
- 3) Catu daya 5V untuk penguat pembalik, penguat instrumentasi dan untuk *driver relay* dan *blower*.
- 4) Menggunakan arduino uno sebagai pengontrol utama
- 5) Menggunakan *Heater* sebagai penghasil panas pada alat yang menghasilkan suhu 65⁰C - 69⁰C
- 6) Menggunakan kipas DC 12V sebagai *blower* yang digunakan untuk meratakan panas dalam ruang.
- 7) Menggunakan LCD 16x2 sebagai tampilan display.
- 8) Menggunakan IC INA125 dan LM358 sebagai penguat *Load Cell* dan penguat tak membalik.
- 9) Menggunakan Sampel berat jamur tiram sebesar 500 gr.

10) Menggunakan elemen pemanas setrika sebagai *Heater*

Pembuatan blok diagram Perancangan alat pengering jamur tiram sebagai alternatif penjemuran matahari ini merupakan dasar dari perancangan system. Diagram blok ini nantinya akan digunakan untuk pengecekan masing- masing blok dari rangkaian yang ditunjukkan dalam Gambar 4.1



Gambar 4. 1 Blok diagram *system*

Sumber: Perancangan, 2016

Dalam Gambar 4.1 dapat dijelaskan secara umum mengenai bagian- bagian yang menyusun keseluruhan sistem dari alat ini, diantaranya:

1) LCD

Digunakan untuk menampilkan hasil pembacaan suhu dan monitoring berat pada proses pengeringan jamur tiram.

2) *Heater*

Digunakan untuk memanaskan ruang dalam proses pengeringan.

3) *Blower*

Digunakan untuk meratakan panas dalam ruang agar suhu disetiap sudut dan sisi diharapkan sama.

4) Arduino Uno

Arduino Uno digunakan sebagai pusat pengolahan data yang diterima dan sebagai pusat kontrol dari keseluruhan system dari alat.

5) Driver Relay

Driver relay digunakan untuk melakukan *switching* pada *blower* dan *heater* guna proses control dalam proses pengeringan.

6) Rangkaian Pengkondisi Sinyal

Rangkaian pengkondisi sinyal terbagi dari dua rangkaian yang terhubung dengan 2 sensor yang digunakan untuk menguatkan tegangan yang dikeluarkan oleh sensor.

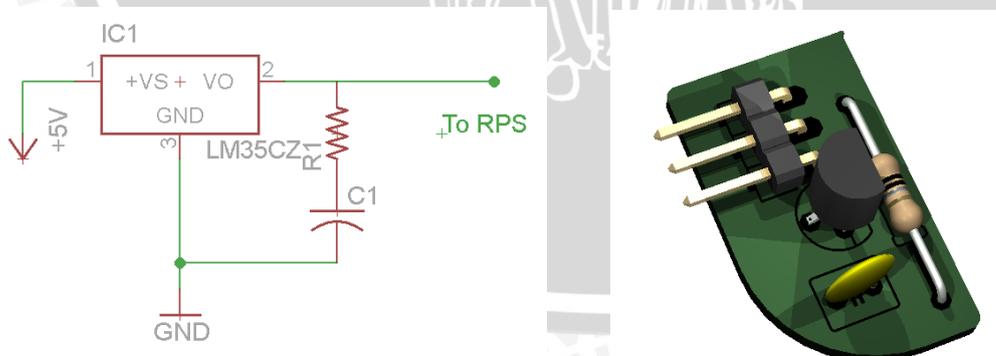
4.2 Prinsip Kerja Alat

Alat bekerja dengan cara mengatur panas yang dihasilkan oleh *heater*. Panas yang dihasilkan oleh heater akan di ratakan pada seluruh ruangan dengan cara mengaktifkan *blower*. Untuk mengatur panas di dalam ruangan pemanas maka digunakan sensor suhu LM35 sebagai *feedback* sistem. Di mana sensor suhu LM35 berfungsi untuk membaca suhu ruangan secara *real time*. Kemudian untuk mengetahui tingkat kering nya jamur digunakan sensor berat *load cell*. Dimana sensor berat berfungsi untuk mengetahui penyusutan berat jamur yang dikeringkan. Ketika berat jamur yang di keringkan sudah sesuai dengan parameter yang di tentukan, maka sistem akan berhenti secara otomatis. Nilai temperatur dan berat pada proses pengeringan akan di tampilkan pada LCD 16x2.

4.3 Perancangan Perangkat Keras

4.4.1 Rangkaian Sensor Suhu LM35

Keluaran sensor suhu LM35 akan dihubungkan langsung ke Rangkaian pengkondisi sinyal. Sensor LM35 harus dihubungkan terlebih dahulu dengan resistor dan kapasitor dikarenakan sensor dihubungkan beban kapasitif yang berupa kabel yang dapat mempengaruhi hasil *output* jika kabel terlalu panjang, hal ini sesuai dengan spesifikasi yang tertera dalam *datasheet*. Gambar perencanaan ditunjukkan dalam Gambar 4.2



Gambar 4. 2 Rangkaian sensor suhu LM35 dan *printed board*

Sumber: Perancangan, 2016

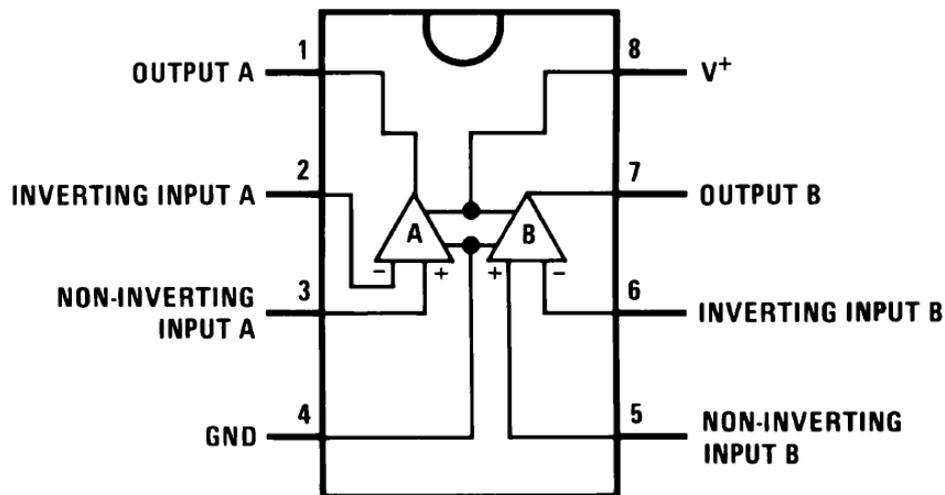
LM35 memiliki kemampuan terbatas untuk menahan beban kapasitif berat. LM35 dengan sendirinya mampu menahan 50 pF tanpa tindakan pencegahan khusus. Jika beban lebih berat diantisipasi, mudah untuk mengisolasi atau memisahkan beban dengan resistor. Atau Anda dapat meningkatkan toleransi kapasitansi dengan serangkaian RC peredam dari *output* ke *Ground*. Namun, seperti dengan rangkaian linier yang terhubung ke kabel di lingkungan yang tidak bersahabat atau terpengaruh oleh *noise*, kinerjanya dapat terpengaruh menjadi tidak maksimal, seperti *relay*, pemancar radio, motor dengan sikat lengkung, SCR transien, kabel, dan lain lain.

Output dari LM35 ini akan dihubungkan langsung ke rangkaian pengkondisi sinyal. LM35 memiliki sensitivitas 10 mV/°C dengan *range* suhu +2°C sampai +150°C (sumber: *National Instrument*, 2000:1).

4.4.2 Rangkaian Penguat Tak Membalik

Rangkaian penguat tak membalik digunakan untuk menguatkan *Output* dari sensor suhu LM35. Rangkaian Penguat Tak Membalik merupakan penguat sinyal dengan karakteristik dasar sinyal *output* yang dikuatkan memiliki fasa yang sama dengan sinyal *input*. Penguat tak-membalik (*non-inverting amplifier*) dapat dibangun menggunakan penguat operasional, karena penguat operasional memang didesain untuk penguat sinyal baik membalik ataupun tak membalik.

Rangkaian penguat tak-membalik ini dapat digunakan untuk memperkuat isyarat AC maupun DC dengan keluaran yang tetap satu fase dengan sinyal *inputnya*. Impedansi masukan dari rangkaian penguat tak-membalik (*non-inverting amplifier*) berharga sangat tinggi dengan nilai impedansi sekitar 100 MOhm. IC LM358 digunakan sebagai IC penguat dikarenakan dapat bekerja dengan daya rendah (*low power amplifier*) dan relative murah dengan spesifikasi yang baik. Gambar 4.3 menunjukkan konfigurasi Pin dari LM 358.



Gambar 4. 3 Konfigurasi Pin LM358

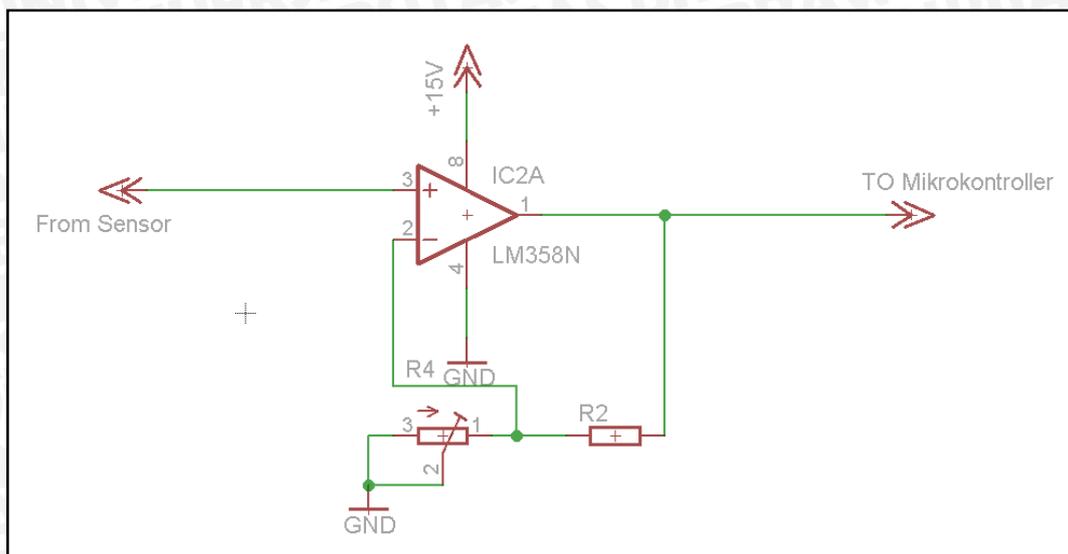
Sumber: Texas Instrument, 2016

Penjelasan kaki pin dari LM 358 yang terdapat dalam Gambar 4.3 ditunjukkan dalam tabel 4.1 sebagai berikut:

Tabel 4.1 Konfigurasi Pin LM358

NO	Nama	Keterangan
1	Output A	Merupakan keluaran dari op amp A
2	Inverting Input A	Merupakan <i>input</i> kutub inverting dari op- amp A
3	Non-Invert Input A	Merupakan <i>input</i> kutub inverting dari op- amp A
4	GND/V-	GND atau catu negatif untuk saturasi
5	Non-Invert Input A	Merupakan <i>input</i> kutub non- inverting dari op- amp B
6	Inverting Input A	Merupakan <i>input</i> kutub inverting dari op- amp B
7	Output B	Merupakan keluaran dari op- amp B
8	V+	Catu positif untuk saturasi

Gambar rangkaian penguat tak membalik ditunjukkan dalam Gambar 4.4



Gambar 4. 4 Rangkaian Penguat Tak Membalik

Sumber: Perancangan, 2016

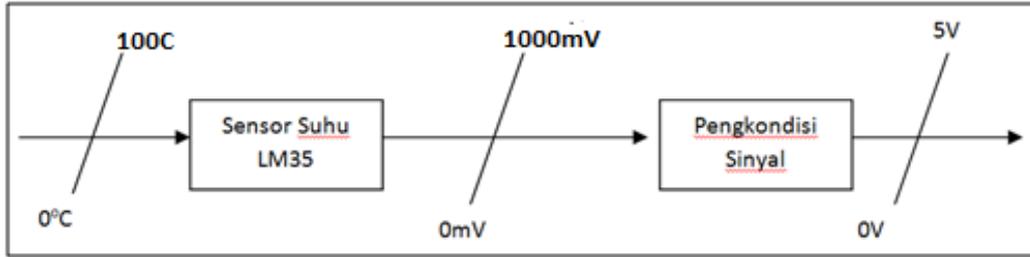
Karena sifat ideal *op- amp*, yaitu impedansi yang tak berhingga besarnya, maka tidak ada arus yang masuk melalui kedua terminal masukan. Dan semua arus akan melewati R1 dan R2, seperti yang ditunjukkan pada Gambar 4.6. Sehingga penguatan dari penguat tak membalik dapat ditulis dengan persamaan 4.1:

$$Av = \frac{R2}{R1} + 1 \quad (4.1)$$

Persamaan diatas menunjukkan bahwa perolehan penguatan bergantung pada R2 dan R1. Selanjutnya V_{out} dapat ditemukan dengan persamaan 4.2 (sumber: handbook, 2013:8):

$$V_{out} = \left(\frac{R2}{R1} + 1 \right) V_i \quad (4.2)$$

Dalam perancangan rangkaian penguat tak membalik untuk rangkaian sensor suhu LM35 terlebih dahulu harus disesuaikan dengan spesifikasi suhu yang diinginkan. Karena orde dari *output* LM35 adalah milivolt maka dibutuhkan *gain*, maka dapat dirancang spesifikasi yang ditunjukkan Gambar 4.5.



Gambar 4.5 Spesifikasi *Input* dan *Output* Pengkondisi Sinyal

Sumber: Perancangan, 2016

Sesuai persamaan (4.2) maka R1 dapat dihitung dengan asumsi bahwa besar R2= 1kΩ.

$$5 = \left(\frac{1000\Omega}{R1} + 1 \right) 1$$

$$5 = \left(\frac{1000\Omega}{R1} + 1 \right)$$

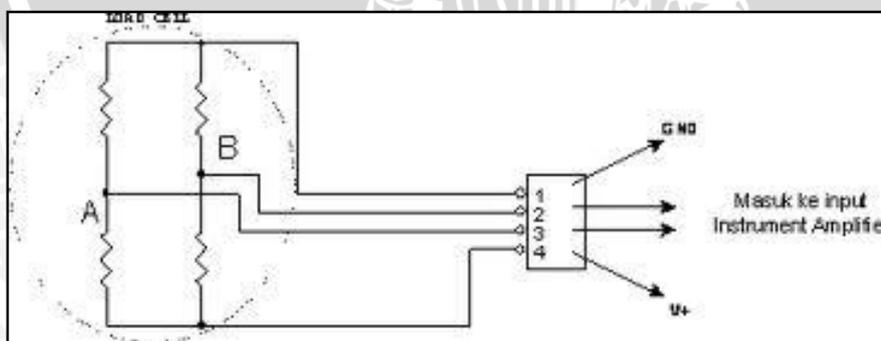
$$5R1 = 1000 + R1$$

$$R1 = 250\Omega$$

Jadi R1 ditetapkan sebesar 250 Ω dan *gain* sebesar 5.

4.4.3 Rangkaian Load Cell

Load cell yang digunakan dalam perancangan alat ini adalah load cell dengan kapasitas maksimum 2kg, dengan sensitivitas sensor ±0.6 mV/V. Rangkaian sensor load cell ditunjukkan dalam Gambar 4.6.



Gambar 4.6 Rangkaian Sensor Load Cell

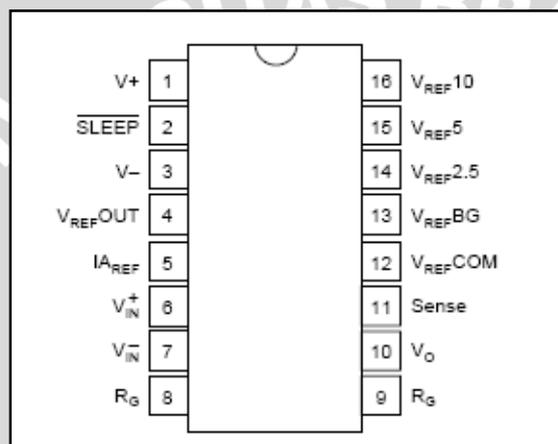
Sumber: delta-electronic.com

Load cell bekerja dengan diberikan tegangan V+ sebesar 5V dengan prinsip kerja *strain gauges* dan jembatan wheatstone. Perbedaan resistansi menyebabkan adanya tegangan yang keluar yang langsung dihubungkan ke rangkaian pengkondisi sinyal.

Load cell akan diberi $V+$ sebesar 5 V. Maka besarnya tegangan full scale dari sensor adalah $0.6\text{mV/V} \times 5\text{V} = 3.0\text{mV}$ sehingga dapat disimpulkan tegangan *output* sebesar 3.0mV adalah berat maksimal yang dicapai yaitu 2Kg.

4.4.4 Rangkaian Penguat Instrumentasi

Rangkaian penguat instrumentasi dalam perancangan alat ini menggunakan INA125. Yang merupakan IC khusus untuk penguatan instrumentasi yang memiliki tegangan referensi yang variatif, memiliki tegangan off se yang kecil maksimum $250\mu\text{V}$ dan memiliki power supply range yang lebar yaitu $\pm 1.35\text{V}$ hingga $\pm 18\text{V}$. Gambar 4.7 menunjukkan konfigurasi pin dari INA125



Gambar 4.7 Konfigurasi Pin INA125

Sumber: Burr- Brown, 2013:4

Penjelasan kaki pin AD 620 ditunjukkan pada tabel 4.2

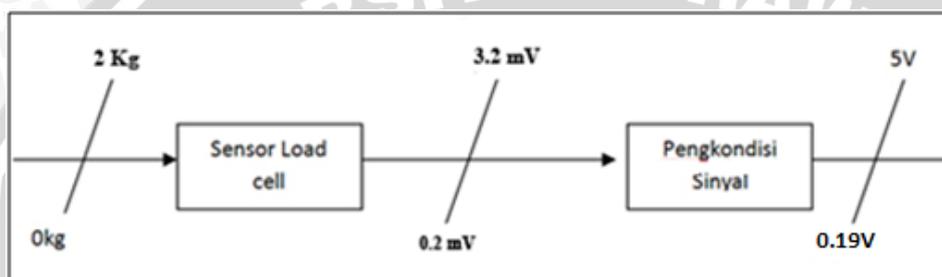
Tabel 4.2 Keterangan Fungsi Pin INA125

NO	Notasi	Keterangan
1	$V+$	Tegangan sumber
2	Sleep	Fasilitas Sleep Automatis
3	$V-$	Ground atau Tegangan negative (untuk dual supply)
4	$V_{\text{ref}} \text{ Out}$	Tegangan referensi yang dihasilkan.
5	$I_{\text{A}}\text{Reff}$	Arus referensi jika ada
6	$V^+\text{In}$	Tegangan positif <i>input</i>
7	$V\text{In}$	Tegangan negative <i>input</i>
8	R_g	Set untuk gain external resistor
9	R_g	Set untuk gain external resistor

10	V_o	Tegangan <i>Output</i> penguat instrumentasi
11	Sense	Dihubungkan dengan V_o
12	$V_{ref} Com$	Gnd untuk tegangan referensi
13	$V_{ref} BG$	Tegangan referensi setengah dr tegangan sumber
14	$V_{ref} 2.5$	Tegangan referensi bernilai 2.5V
15	$V_{ref} 5$	Tegangan referensi bernilai 5V
16	$V_{ref} 10$	Tegangan referensi bernilai 10V

Sumber: Burr- Brown, 2013:4

Penentuan spesifikasi untuk sensor load cell hingga menjadi *input* yang sesuai yang diinginkan ditunjukkan dalam Gambar 4.8.



Gambar 4. 8 Spesifikasi *Input* dan *Output* penguat instrumentasi

Sumber: perancangan, 2016

Sebelum menentukan R_G , terlebih dahulu harus di tentukan *gain* dari penguat instrumentasi. *Gain* ditentukan dengan persamaan berikut (sumber: datasheet INA125, 2016:10)

$$Gain = 4 + \frac{60K\Omega}{R_G} \quad (4.3)$$

Namun karena nilai R_G tidak dikehui maka terlebih dahulu kita menghitung gain dengan cara perbandingan dua persamaan sebagai berikut,

$$Gain = \frac{V_{outamp} - V_{inamp}}{V_{inbef} - V_{outbef}} \quad (4.4)$$

$$\text{Gain} = \frac{(5 - 0.19) \text{ Volt}}{(0.0032 - 0.0002) \text{ Volt}} = 1603.33$$

Dengan

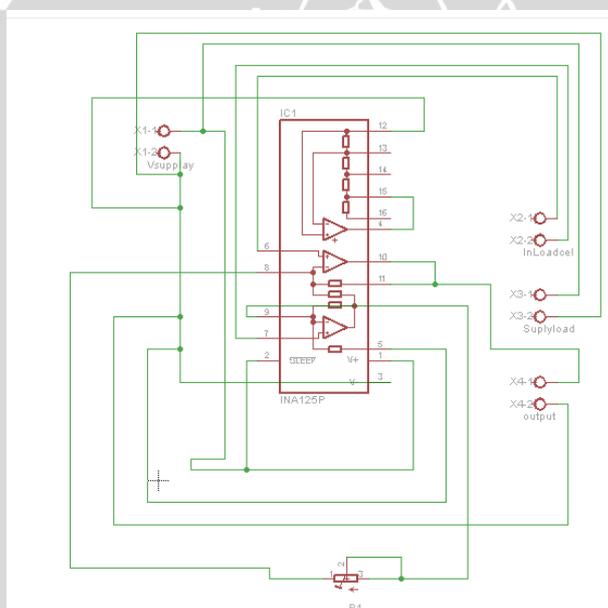
V_{inamp} = tegangan yg di inginkan awal

V_{outamp} = tegangan akhir yang diinginkan

V_{inbef} = tegangan awal sensor

V_{inbef} = tegangan akhir sensor

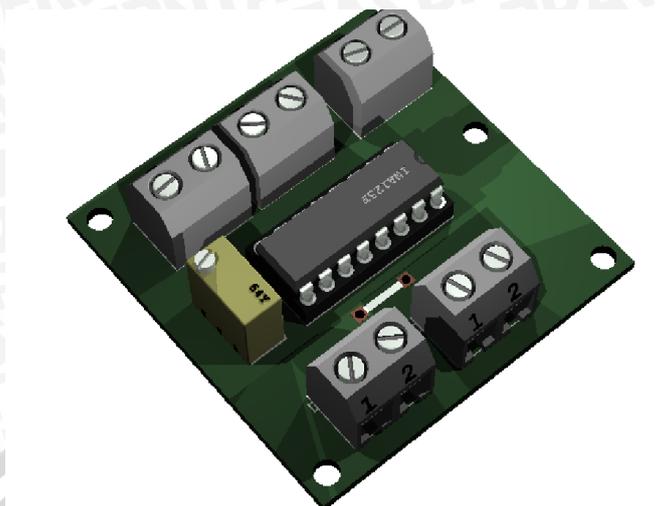
Maka didapatkan nilai gain sebesar 1603.33 dan setelah dimasukkan ke persamaan (7) didapatkan R_g sebesar 37.51Ω , hal ini diperlukan agar *output* dari penguat instrumentasi sesuai dengan spesifikasi *input* dan *output* yang diinginkan yaitu dari 0.19- 5V. Rangkaian penguat instrumentasi ditunjukkan pada Gambar 4.9 berikut:



Gambar 4.9 Rangkaian Penguat Instrumentasi

Sumber: Perancangan, 2016

Rangkaian penguat instrumentasi ditunjukkan pada Gambar 4.10 berikut.

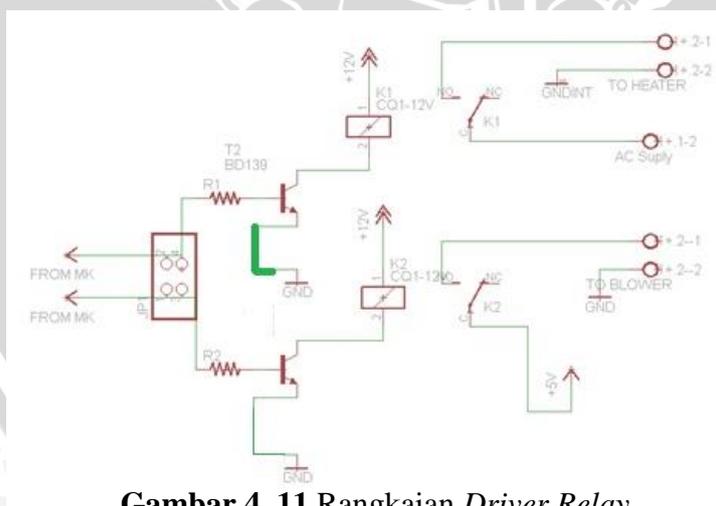


Gambar 4. 10 Printed Board Blok Rangkaian Pengkondisi Sinyal

4.4.5 Rangkaian Driver Relay

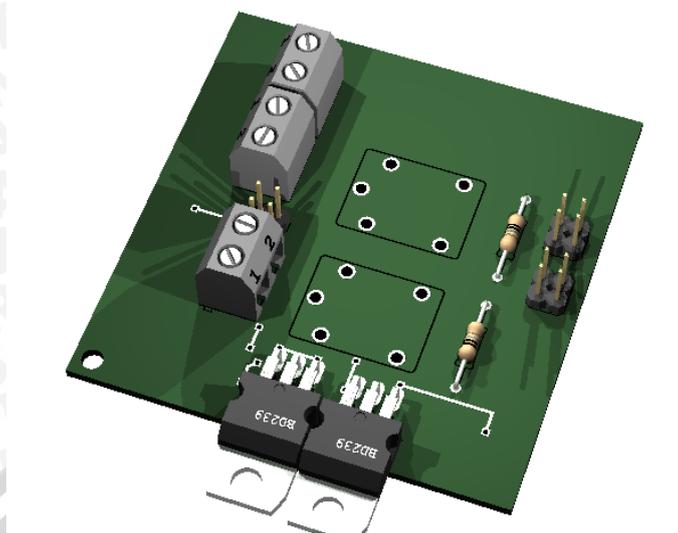
Relay dalam perancangan ini digunakan dalam menghidupkan dan mematikan dua komponen actuator yaitu *heater* dan *blower*. Relay yang digunakan dalam perancangan ini adalah tipe SPDT.

Rangkaian driver relay dan *printed board* ditunjukkan dalam Gambar 4.11 dan Gambar 4.12.



Gambar 4. 11 Rangkaian Driver Relay

Sumber: Perancangan, 2016



Gambar 4. 12 Printed Board Rangkaian *Driver Relay*

Sumber: Perancangan, 2016

Untuk bekerja sesuai yang diinginkan relay dengan tahanan coil sebesar 55Ω perlu di *trigger* terlebih dahulu oleh transistor, diakrenakan tegangan mikrokontroler tidak bisa langsung mengaktifkan relay. Untuk men- *drive* Coil dari sebuah relay, diperlukan rangkaian switching transistor, karena biasanya *output* digital dari sebuah IC atau TTL arusnya tidak cukup kuat melakukannya. Disini tipe transistor yang digunakan adalah BD139 NPN Silicon Transistor dan Relay SPST 5V. R1 dan R2 (Rb) pun harus dihitung terlebih dahulu agar dapat memicu transistor sesuai dengan aturan yang ada.

Untuk menghitung Rb pertama kita harus menghitung I_L (arus beban). Dari data yang sudah ada berikut.

- $I_C \text{ MAX}$ = 1.5 A
- $V_{CE} \text{ Saturasi}$ = 0.5 V
- V_{BE} = 1 V
- h_{fe} = 40

Nilai arus kolektor (I_C) yang mengalir pada kolektor pada saat kondisi saturasi dapat dihitung berdasarkan persamaan:

$$\begin{aligned}
 V_{CC} &= I_C \cdot R_{COIL} + V_{CE} \\
 5V &= I_C \cdot 55\Omega + 0,5V \\
 I_C &= \frac{5V - 0,5V}{55\Omega} \\
 &= 81.81 \text{ mA}
 \end{aligned}$$

Nilai ini masih dibawah nilai arus maskimal pada kolektor sebesar 1500 mA. Nilai arus basis (I_B) yang mengalir pada kaki basis transistor BD139 dapat dihitung berdasarkan persamaan:

$$I_B = \frac{I_C}{h_{fe}}$$

$$I_B = \frac{81.81 \text{ mA}}{60}$$

$$= 1,36 \text{ mA}$$

Nilai R_B dapat dihitung berdasarkan persamaan berikut:

$$R_B = \frac{V_{IN} - V_{BE}}{I_B}$$

$$= \frac{(5 - 1)V}{(1.36)mA} = 2.94k\Omega$$

Nilai resistor 2,94k Ω merupakan nilai maksimal yang harus terpasang pada kaki basis pada transistor, maka digunakan resistor yang dijual dipasaran dengan nilai 2.7k Ω .

4.5 Perancangan Sistem Kontrol Logika Fuzzy

4.5.1 Variabel Masukan dan Keluaran

Sistem kontrol logika *fuzzy* yang dikembangkan dalam penelitian ini mempunyai dua *crisp input* yaitu *error* posisi dan *delta error* posisi serta satu *crisp output* yaitu perubahan tegangan. *Error* dan $\Delta Error$, didefinisikan dengan perumusan sebagai berikut:

$$Error(t) = SP - PV(t) \quad (4.5)$$

Dimana,

$$SP = \text{Set point (Nilai yang diinginkan)}$$

$$PV(t) = \text{Present Value pada waktu } t \text{ (Nilai aktual)}$$

$$\Delta Error(t) = Error(t) - Error(t-1) \quad (4.6)$$

Dimana,

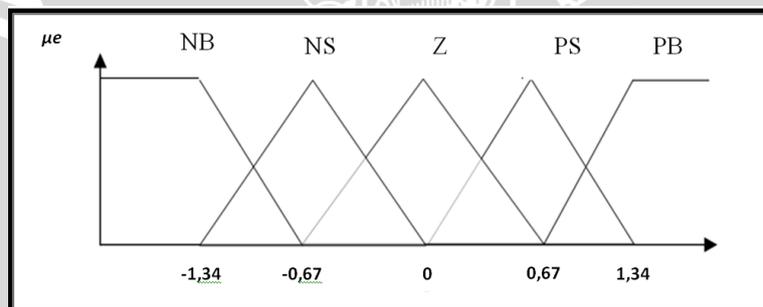
$$Error(t) = \text{Error pada waktu } t$$

$$Error(t-1) = \text{Error pada waktu } (t-1)$$

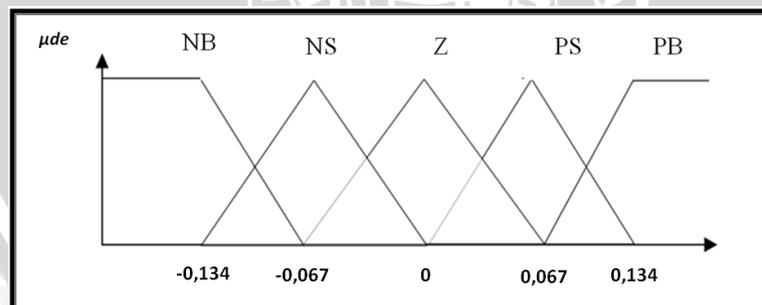
4.5.2 Fungsi Keanggotaan Masukan

Fungsi keanggotaan dari *error* dan *delta error* terdiri dari lima label, yaitu *Negative Big* (NB), *Negative Small* (NS), *Zero* (Z), *Positive Small* (PS), *Positive Big* (PB). *Error* adalah nilai *set point* dikurangi nilai sebenarnya, sedangkan *delta error* didapat dari *error* nilai kecepatan sekarang dikurangi dengan *error* nilai kecepatan yang sebelumnya. Data masukan dari fungsi keanggotaan didapat dari nilai suhu yang dikeluarkan oleh sensor suhu LM35.

Label masing-masing fungsi keanggotaan masukan berjumlah 5 dengan tujuan agar rule yang dihasilkan semakin banyak. Semakin banyak rule maka semakin banyak juga kemungkinan kondisi yang tercipta untuk mencapai keluaran yang diinginkan. Penentuan besarnya fungsi keanggotaan *error* dilakukan dengan perhitungan toleransi *error* 2% dari *set point* yang ditentukan, sedangkan untuk fungsi keanggotaan *delta error* ditentukan dari 10% fungsi keanggotaan *error*. Berikut Gambar 4.13 dan 4.14 yang merupakan *membership function* yang di gunakan:



Gambar 4.13 Fungsi Keanggotaan Masukan *Error*



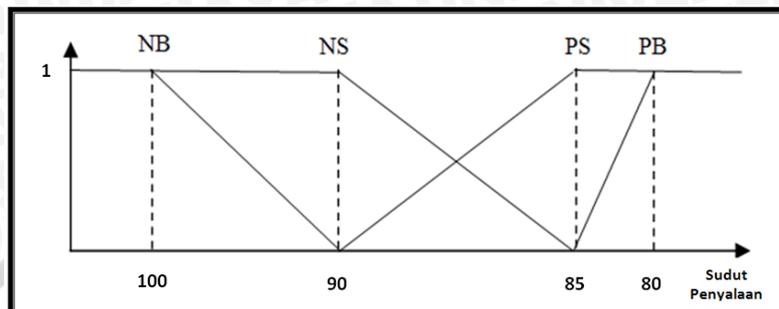
Gambar 4.14 Fungsi Keanggotaan Masukan *Delta Error*

4.5.3 Fungsi Keanggotaan Keluaran

Fungsi keanggotaan keluaran heater merupakan representasi suhu penegeringan jamur dan besarnya nilai penyalaan rangkaian dimmer yang dikeluarkan oleh Arduino berupa hasil dari perhitungan Metode defuzzifikasi *Weighted Average* yang telah dibulatkan oleh integer di dalam pemrograman. Pemilihan defuzzifikasi sebagai penentu keluaran

dipilih dengan alasan agar mempercepat eksekusi program dan kapasitas program yang ditulis tidak melebihi kapasitas memori dari Arduino sebesar 32 kilobyte.

Sudut penyalan yang dikeluarkan oleh arduino antara 0° - 180° dengan kecepatan. Dalam pemrograman sudut penyalan yang terpakai antara 80-100, dengan 80 untuk kecepatan terendah dan 100 untuk suhu tertinggi.



Gambar 4.15 Fungsi Keanggotaan Keluaran Suhu

Keterangan:

NB : *Negatif Big*

NS : *Negatif Small*

PS : *Positif Small*

PB : *Positif Big*

4.5.4 Perancangan Aturan Fuzzy

Rule (aturan) fuzzy digunakan sebagai penentu keluaran dari fuzzifikasi yang akan diolah dalam proses defuzzifikasi, dengan jumlah rule 25 macam. *Rule* tersebut didapat

Tabel 4.3 Aturan Fuzzy

e de	NB	NS	Z	PS	PB
NB	PB	PB	PB	PS	PS
NS	PB	PB	PS	NS	NS
Z	PB	PS	NS	NS	NB
PS	PS	PS	NS	NB	NB
PB	PS	NS	NS	NB	NB

4.5.5 Metode Inferensi Min-Max

Untuk mengetahui metode ini pada sistem, perlu melakukan beberapa perhitungan pembuktian metode min-max pada sistem. Pengujian hanya dilakukan pada *set point* tertentu sebagai perwakilan atas keseluruhan sistem. Berikut hasil perhitungan dan Tabel 4.4 *set point* dan keluaran yang dihasilkan.

Sample ke-	Setpoint	Keluaran
1000	67	67,3
1001	67	67,5
1002	67	67,4
1003	67	67,3
1004	67	67,6
1005	67	67,5

Tabel 4.4 *set point* dan keluaran

Perhitungan inferensi min-max pada sistem

$$\text{Error}_{(t)} = \text{SP}_{(t)} - \text{PV}_{(t)}$$

$$\text{Delta Error}_{(t)} = \text{Error}_{(t)} - \text{Error}_{(t-1)}$$

Keterangan :

$\text{Error}_{(t)}$: Simpangan / kesalahan pada detik (t)

$\text{Error}_{(t-1)}$: Simpangan / kesalahan sebelum detik (t)

$\text{SP}_{(t)}$: *Set point*

$\text{PV}_{(t)}$: Nilai Sekarang

$\text{Delta Error}_{(t)}$: Simpangan antara error pada detik (t) dengan (t-1)

1. Inferensi Min-Max Sample ke 1001

$$\text{Error}_{(t)} = \text{SP}_{(t)} - \text{PV}_{(t)}$$

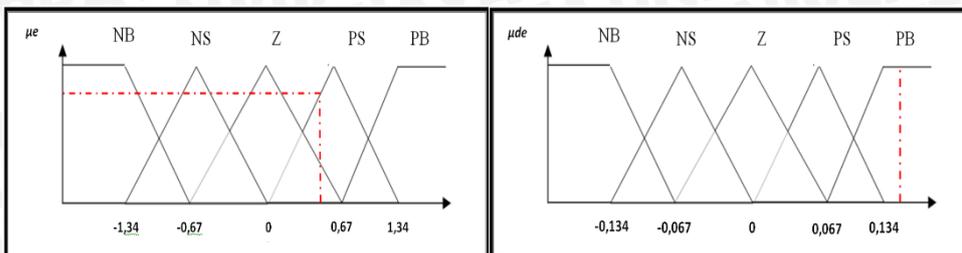
$$\begin{aligned} \text{Error}_{(1000)} &= \text{SP} - \text{PV}_{(1000)} \\ &= 67 - 67,3 \\ &= 0,3 \end{aligned}$$

$$\text{Error}_{(t)} = \text{SP}_{(t)} - \text{PV}_{(t)}$$

$$\begin{aligned} \text{Error}_{(1001)} &= \text{SP} - \text{PV}_{(1001)} \\ &= 67 - 67,5 \\ &= 0,5 \end{aligned}$$

$$\begin{aligned}\text{Delta Error}(t) &= \text{Error}(t) - \text{Error}(t-1) \\ \text{Delta Error}(1001) &= \text{Error}(1001) - \text{Error}(1000) \\ &= 0,5 - 0,3 \\ &= 0,2\end{aligned}$$

Gambar 4.16 adalah Inferensi Min-Max Saat *Sample* ke-1001



Gambar 4.16 Inferensi Min-Max Saat *Sample* ke-1001

Nilai derajat keanggotaan Error:

$$\text{Error} = 0.25[Z]$$

$$\text{Error} = 0.75[PS]$$

Nilai derajat keanggotaan DelError:

Hasil :

$$\text{Error}_1 = 0.25[Z] \rightarrow \text{Del Error}_1 = 1[PB] = 0.25[NS]$$

$$\text{Error}_2 = 0.25[Z] \rightarrow \text{Del Error}_2 = 0[PS] = 0[NS]$$

2. Inferensi Min-Max Sample ke 1002

$$\text{Error}(t) = \text{SP}(t) - \text{PV}(t)$$

$$\begin{aligned}\text{Error}(1001) &= \text{SP} - \text{PV}(1001) \\ &= 67 - 67,5 \\ &= 0,5\end{aligned}$$

$$\text{Error}(t) = \text{SP}(t) - \text{PV}(t)$$

$$\begin{aligned}\text{Error}(1002) &= \text{SP} - \text{PV}(1002) \\ &= 67 - 67,4 \\ &= 0,4\end{aligned}$$

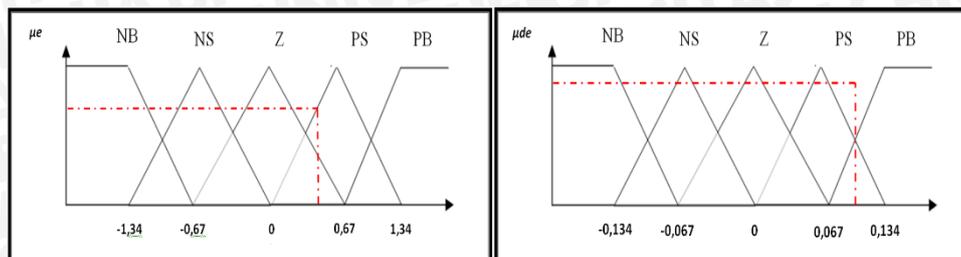
$$\text{Delta Error}(t) = \text{Error}(t) - \text{Error}(t-1)$$

$$\text{Delta Error}(1002) = \text{Error}(1002) - \text{Error}(1001)$$

$$= 0,4 - 0,5$$

$$= 0,1$$

Gambar 4.17 menunjukkan Inferensi Min-Max Saat *Sample* ke-1002



Gambar 4.17 Inferensi Min-Max Saat *Sample* ke-1002

Nilai derajat keanggotaan Error:

$$\text{Error} = 0.4[Z]$$

$$\text{Error} = 0.6[PS]$$

Nilai derajat keanggotaan DelError:

Hasil :

$$\text{Error}_1 = 0.4[Z] \rightarrow \text{Del Error}_1 = 0,5[PB] = 0.4[NS]$$

$$\text{Error}_2 = 0.4[Z] \rightarrow \text{Del Error}_2 = 0,5[PS] = 0[NS]$$

3. Inferensi Min-Max Sample ke 100

$$\text{Error}(t) = SP(t) - PV(t)$$

$$\text{Error}(1002) = SP - PV_{(1002)}$$

$$= 67 - 67,4$$

$$= 0,4$$

$$\text{Error}(t) = SP(t) - PV(t)$$

$$\text{Error}(1003) = SP - PV_{(1003)}$$

$$= 67 - 67,$$

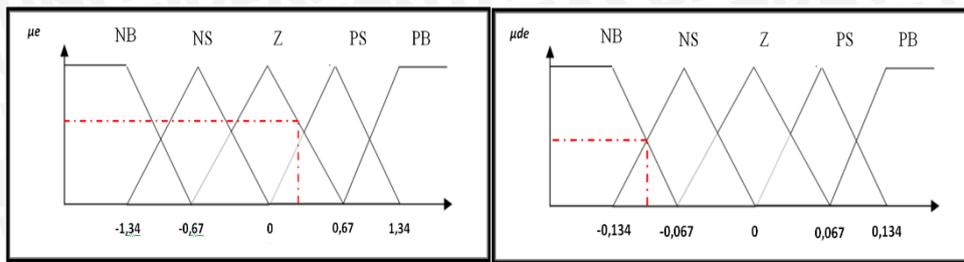
$$= 0,3$$

$$\text{Delta Error}(t) = \text{Error}(t) - \text{Error}(t-1)$$

$$\text{Delta Error}(1003) = \text{Error}(1003) - \text{Error}(1002)$$

$$= 0,3 - 0,4$$

$$= -0,1$$

Gambar 4.18 menunjukkan Inferensi Min-Max Saat *Sample* ke-1003Gambar 4.18 Inferensi Min-Max Saat *Sample* ke-1003

Nilai derajat keanggotaan Error:

$$\text{Error} = 0.44[Z]$$

$$\text{Error} = 0.56[PS]$$

Nilai derajat keanggotaan DelError:

Hasil :

$$\text{Error}_1 = 0.44[Z] \rightarrow \text{Del Error}_1 = 0,5[NB] = 0.44[PB]$$

$$\text{Error}_1 = 0.44[Z] \rightarrow \text{Del Error}_2 = 0,5[NS] = 0[PS]$$

4. Inferensi Min-Max *Sample* ke 1004 $\text{Error}(t) = \text{SP}(t) - \text{PV}(t)$

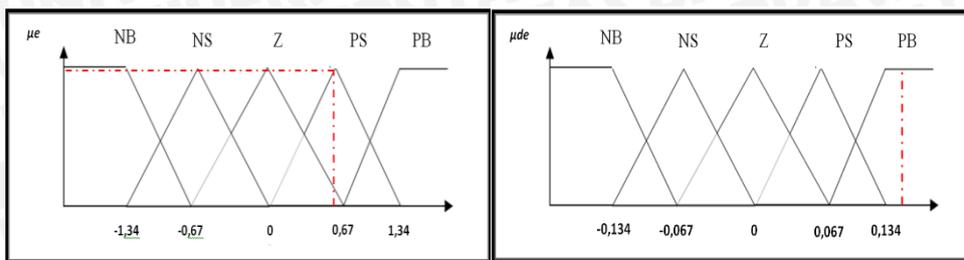
$$\begin{aligned} \text{Error}(1003) &= \text{SP} - \text{PV}_{(1003)} \\ &= 67 - 67,3 \\ &= 0,3 \end{aligned}$$

$$\text{Error}(t) = \text{SP}(t) - \text{PV}(t)$$

$$\begin{aligned} \text{Error}(1004) &= \text{SP} - \text{PV}_{(1004)} \\ &= 67 - 67,6 \\ &= 0,6 \end{aligned}$$

$$\text{Delta Error}(t) = \text{Error}(t) - \text{Error}(t-1)$$

$$\begin{aligned} \text{Delta Error}(1004) &= \text{Error}(1004) - \text{Error}(1003) \\ &= 0,6 - 0,3 \\ &= 0,3 \end{aligned}$$

Gambar 4.19 menunjukkan Inferensi Min-Max Saat *Sample* ke-1004Gambar 4.19 Inferensi Min-Max Saat *Sample* ke-1004

Nilai derajat keanggotaan Error:

$$\text{Error} = 0.10[Z]$$

$$\text{Error} = 0.90[PS]$$

Nilai derajat keanggotaan Δ Error:

Hasil :

$$\text{Error}_1 = 0.10[Z] \rightarrow \Delta \text{Error}_1 = 1[PB] = 0.10[NS]$$

$$\text{Error}_2 = 0.10[Z] \rightarrow \Delta \text{Error}_2 = 1[PS] = 1[NS]$$

$$\text{Error}(t) = SP(t) - PV(t)$$

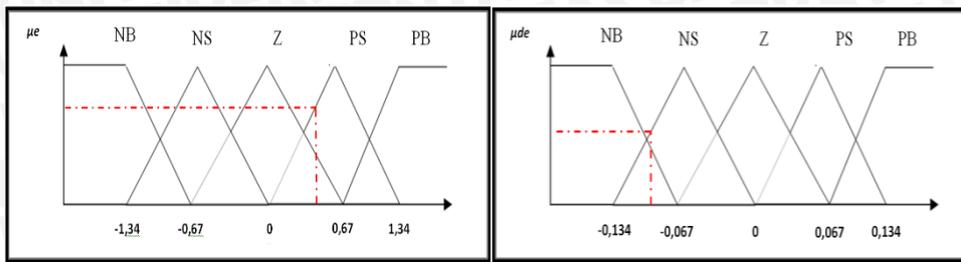
$$\begin{aligned} \text{Error}(1004) &= SP - PV_{(1004)} \\ &= 67 - 67,6 \\ &= 0,6 \end{aligned}$$

$$\text{Error}(t) = SP(t) - PV(t)$$

$$\begin{aligned} \text{Error}(1005) &= SP - PV_{(1005)} \\ &= 67 - 67,5 \\ &= 0,5 \end{aligned}$$

$$\Delta \text{Error}(t) = \text{Error}(t) - \text{Error}(t-1)$$

$$\begin{aligned} \Delta \text{Error}(1005) &= \text{Error}(1005) - \text{Error}(1004) \\ &= 0,5 - 0,6 \\ &= -0,1 \end{aligned}$$

Gambar 4.20 menunjukkan Inferensi Min-Max Saat *Sample* ke-1005**Gambar 4.20** Inferensi Min-Max Saat *Sample* ke-1005

Nilai derajat keanggotaan Error:

$$\text{Error} = 0.4[Z]$$

$$\text{Error} = 0.6[PS]$$

Nilai derajat keanggotaan DelError:

Hasil :

$$\text{Error1} = 0.4[Z] \rightarrow \text{Del Error1} = 0,5[PB] = 0.4[NS]$$

$$\text{Error1} = 0.4[Z] \rightarrow \text{Del Error2} = 0,5[PS] = 0[NS]$$

4.5.6 Defuzzifikasi

Defuzzifikasi adalah proses untuk mengubah keluaran *fuzzy* menjadi keluaran *crisp*. Hasil defuzzifikasi inilah yang digunakan untuk mengatur besarnya sudut penyalan rangkaian dimmer. Metode defuzzifikasi yang digunakan adalah *Weighted Average*.

Proses defuzzifikasi menggunakan keluaran Tsukamoto yang memiliki nilai monoton. Pemilihan penggunaan keluaran Tsukamoto dikarenakan metode ini mampu menutupi hasil lingkup keluaran yang tidak mampu dijangkau oleh keluaran singleton.

Perhitungan nilai keluaran sistem dilakukan dengan perkalian fungsi keanggotaan dengan hasil perkalian dari persamaan garis keluaran Tsukamoto dengan hasil inferensi dari fuzzifikasi.

Berdasarkan contoh hasil proses inferensi, maka keluaran sudut penyalan rangkaian dimmer:

1. Defuzzifikasi untuk Inferensi Min-Max saat sampel ke 1001

$$\begin{aligned}
 U &= \frac{\sum_{i=1}^n w_i u_i}{\sum_{i=1}^n w_i} \\
 &= \frac{0.25 \times NS + 0.75 \times NB}{0.25 + 0.75} \\
 &= \frac{0.25 \times (100 - (0.25 \times 10)) + 0.75 \times (90 + (0.75 \times 10))}{0.25 + 0.75} \\
 &= \frac{24.375 + 73.125}{0.25 + 0.75} \\
 &= 97.5
 \end{aligned}$$

2. Defuzzifikasi untuk Inferensi Min-Max saat sampel ke 1002

$$\begin{aligned}
 U &= \frac{\sum_{i=1}^n w_i u_i}{\sum_{i=1}^n w_i} \\
 &= \frac{0.4 \times NS + 0.6 \times NB}{0.4 + 0.6} \\
 &= \frac{0.4 \times (100 - (0.4 \times 10)) + 0.6 \times (90 + (0.6 \times 10))}{0.4 + 0.6} \\
 &= \frac{38.4 + 57.6}{0.4 + 0.6} \\
 &= 96
 \end{aligned}$$

3. Defuzzifikasi untuk Inferensi Min-Max saat sampel ke 1003

$$\begin{aligned}
 U &= \frac{\sum_{i=1}^n w_i u_i}{\sum_{i=1}^n w_i} \\
 &= \frac{0.44 \times PB + 0.56 \times PS}{0.44 + 0.56} \\
 &= \frac{0.44 \times (85 - (0.44 \times 5)) + 0.56 \times (90 - (0.56 \times 5))}{0.44 + 0.56} \\
 &= \frac{36.432 + 48.832}{0.44 + 0.56} \\
 &= 85.264
 \end{aligned}$$

4. Defuzzifikasi untuk Inferensi Min-Max saat sampel ke 1004

$$\begin{aligned}
 U &= \frac{\sum_{i=1}^n w_i u_i}{\sum_{i=1}^n w_i} \\
 &= \frac{0.10 \times NS + 0.90 \times NB}{0.10 + 0.90} \\
 &= \frac{0.10 \times (100 - (0.10 \times 10)) + 0.90 \times (90 + (0.90 \times 10))}{0.10 + 0.90} \\
 &= \frac{9.9 + 89.1}{0.10 + 0.90} \\
 &= 99
 \end{aligned}$$

5. Defuzzifikasi untuk Inferensi Min-Max saat sampel ke 1005

$$\begin{aligned}
 U &= \frac{\sum_{i=1}^n w_i u_i}{\sum_{i=1}^n w_i} \\
 &= \frac{0.40 \times NS + 0.60 \times NB}{0.4 + 0.6} \\
 &= \frac{0.40 \times (100 - (0.10 \times 10)) + 0.60 \times (90 + (0.90 \times 10))}{0.40 + 0.60} \\
 &= \frac{39.6 + 59.4}{0.40 + 0.60} \\
 &= 99
 \end{aligned}$$

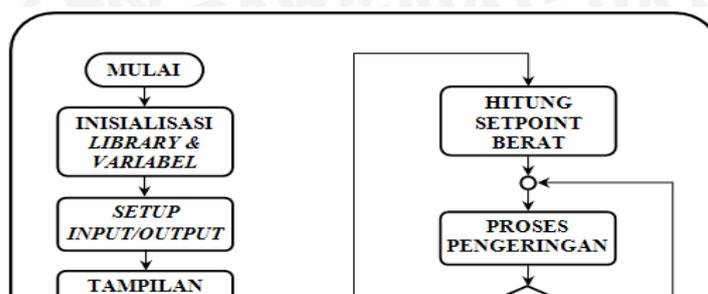
4.6 Perancangan Perangkat Lunak

Perangkat lunak pada perancangan alat ini dirancang dengan menggunakan *software* arduino.ide. Untuk memberikan Gambaran umum jalannya program dan memudahkan pembuatan perangkat lunak, maka dibuat flowchart yang menunjukkan jalannya program.

4.6.1 Perancangan Program Utama

Flow chart program utama dalam perancangan alat ini ditunjukkan pada Gambar 4.21.

Pada diagram alir proses inisialisasi awal pada mikrokontroler ini meliputi pengaturan port *input/output* arduino uno, pengaturan adc, pengaturan LCD.



Gambar 4.21 Flowchart Program Sistem Keseluruhan

Sumber: Perancangan, 2016

Dalam alat ini proses dimulai dengan memberikan nilai berat benda yang akan dikeringkan, nilai yang telah diberikan berikutnya menjadi *input* dalam proses pengeringan. Setelah tombol *start* ditekan maka proses pengeringan akan dimulai dengan suhu yang telah ditentukan.

Dalam alat ini Proses pengeringan akan berhenti jika berat benda telah mencapai berat yang sesuai dengan parameter berat benda telah kering. Sehingga pengguna tidak direpotkan dengan memasukkan data secara manual. Dalam proses pengeringan nilai suhu dan berat selama proses pengeringan akan ditampilkan dalam LCD, sehingga memudahkan pengguna dalam pemantauan proses pengeringan.

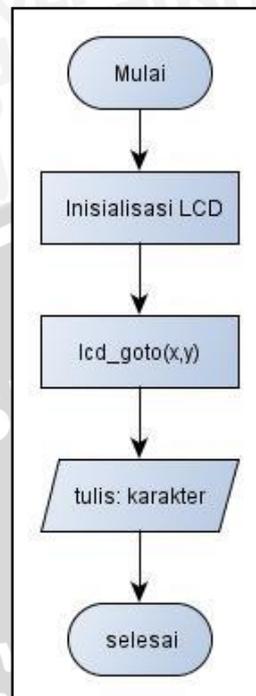
4.7 Sub Program Tampil LCD

Program untuk meng- akses LCD dirancang menggunakan *file header* "lcd.h". lcd digunakan untuk memantau dan mengamati proses selama pengeringan berlangsung. Diagram alir program ditunjukkan dalam Gambar 4.22.

Hal yang terpenting untuk menggunakan LCD adalah dengan melakukan inisialisasi LCD terlebih dahulu. Mode operasi yang digunakan dalam perancangan alat ini adalah mode operasi 4 bit, yang artinya hanya ada 4 port dari LCD yang dihubungkan keport Mikrokontroler.

Dalam perancangan ini port yang digunakan sebagai *output* ke LCD adalah PC.3-PC.5 dan PD.3 sebagai jalur data bus yang diperlukan dalam proses transfer data. Pada mode

4 bit ini data yang akan dikirim terlebih dahulu adalah MSB dan yang akan dilanjutkan data 4bit LSB

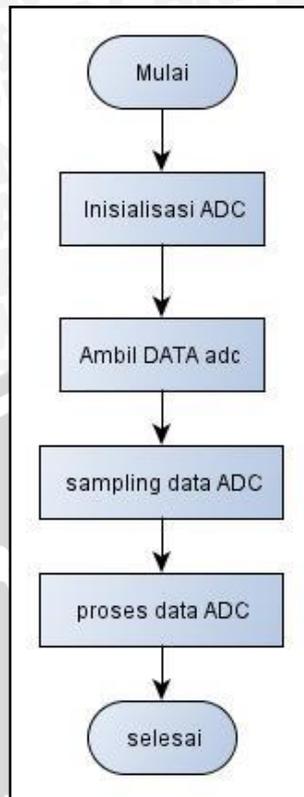


Gambar 4.22 Flowchart Program LCD

Sumber: Perancangan, 2016

4.8 Sub Program Baca Suhu LM35

Sub program Baca Suhu LM35 merupakan bagian dari program utama agar proses pengeringan berjalan sesuai dengan parameter- parameter yang telah ditentukan sebelumnya. Flow chart dibuat agar LM35 dapat membaca suhu yang sesuai dan tepat dengan suhu yang ada disekitar sensor. Flowchart ditunjukkan pada Gambar 4.23.



Gambar 4. 23 Flowchart Baca suhu LM35

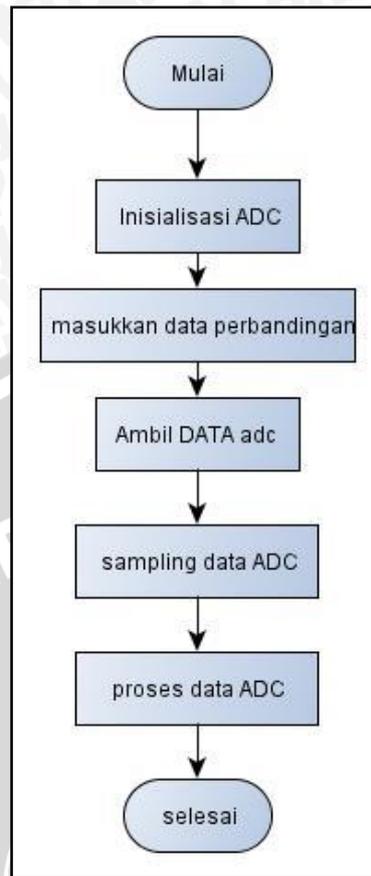
Sumber: Perancangan, 2016

Dalam sub program baca suhu LM35 ini menggunakan ADC sehingga hal yang pertama kali dilakukan adalah inisialisasi ADC terlebih dahulu. Pada inisialisasi ADC ini dilakukan pengaktifan ADC, pengaturan resolusi ADC sebesar 10 bit, tegangan referensi yang digunakan sebesar 5 Volt, dan pemilihan mode konversi *free- running*.

Kemudian data ADC diambil dari pin analog 1 yang terhubung langsung dengan *output* rangkaian instrumentasi sensor suhu LM35. Untuk hasil yang terbaik maka pengukuran suhu dilakukan dengan mengambil rata-rata data dari hasil sampling ADC sebanyak sampling yang telah ditentukan. Data hasil sampling ini diproses sehingga didapat data suhu yang sesuai dengan sebenarnya.

4.9 Sub Program Baca Load Cell

Sub program Baca *Load cell* ini merupakan bagian utama dalam proses pengeringan. Proses pengeringan sangat bergantung pada hasil berat yang dihasilkan. Untuk membaca perubahan berat benda selama proses pengeringan maka dibuat flow chart agar berat benda sesuai dengan berat sesungguhnya. Flowchart baca *Load cell* ditunjukkan pada Gambar 4.24.



Gambar 4. 24 Flowchart Baca *Load Cell*

Sumber: Perancangan, 2016

Sama seperti pada proses baca Suhu LM35, untuk membaca *load cell* juga memerlukan inialisasi adc, sampling data, dan proses data adc. Namun agar berat benda sesuai dengan berat benda sesungguhnya perlu terlebih dahulu di hitung dua berat benda yang telah diketahui untuk kemudian dimasukkan sebagai konstanta dalam program nantinya agar didapat berat yang sesuai dengan berat sesungguhnya. Hal ini dapat disebut juga sebagai kalibrasi sensor *load cell*



BAB V

PENGUJIAN DAN ANALISIS

Bab ini membahas pengujian dan analisis alat yang telah dirancang dan direalisasikan. Pengujian dilakukan di tiap-tiap blok dengan tujuan untuk mengamati apakah tiap blok rangkaian sudah sesuai dengan perancangan, kemudian dilanjutkan dengan pengujian keseluruhan sistem. Pengujian ini terdiri atas pengujian perangkat keras dan pengujian perangkat lunak. Jika semua pengujian telah dilakukan maka dilakukan pengujian secara keseluruhan.

Pengujian yang dilakukan adalah sebagai berikut:

- 1) Pengujian *Load Cell*
- 2) Pengujian Penguat Instrumentasi INA 125
- 3) Pengujian Sensor Suhu LM35
- 4) Pengujian Sensor Suhu Dengan Penguat Pembalik
- 5) Pengujian Sistem Secara Keseluruhan

5.1 Pengujian *Load Cell*

Tujuan pengujian ini adalah untuk mengetahui hasil pengukuran berat dari sebuah benda dalam sebuah respon perubahan benda dengan cara mengukur tegangan yang dihasilkan oleh *Load Cell* sebelum dan setelah dikuatkan dalam proses pengukuran.

5.1.1 Peralatan Pengujian

1. *Load Cell* kapasitas 2kg
2. Beberapa bandul pemberat
3. Timbangan Acuan
4. Catu daya +5 Volt
5. Multimeter

5.1.2 Prosedur Pengujian

Pengujian dilakukan dengan cara mengukur tegangan keluaran sensor *load cell*, dimana sebelumnya sensor *load cell* diberikan catu daya 5V.

Pengujian dilakukan untuk membandingkan antara beban yang diukur sensor *load cell* dengan tegangan keluaran yang dihasilkan.

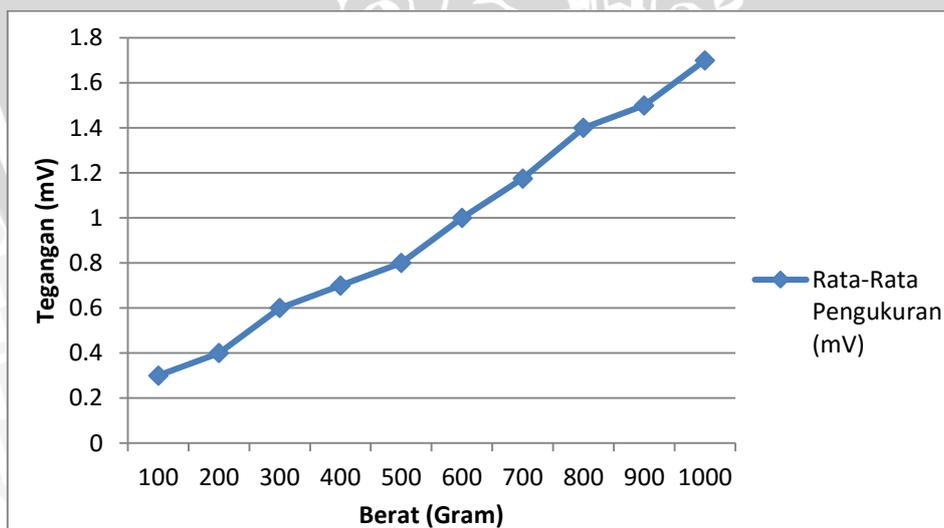
5.1.3 Hasil Pengujian

Hasil pengujian sensor *load cell* ditunjukkan dalam Tabel 5.1

Tabel 5.1 Hasil Pengujian Sensor Load Cell

No.	Berat (Gram)	Tegangan Pengukuran (mV)				Rata-Rata Pengukuran (mV)
		1	2	3	4	
1	100	0,3	0,3	0,3	0,3	0,3
2	200	0,4	0,4	0,4	0,4	0,4
3	300	0,6	0,6	0,6	0,6	0,6
4	400	0,7	0,7	0,7	0,7	0,7
5	500	0,8	0,8	0,8	0,8	0,8
6	600	1	1	1	1	1
7	700	1,2	1,2	1,1	1,2	1,175
8	800	1,4	1,4	1,4	1,4	1,4
9	900	1,5	1,5	1,5	1,5	1,5
10	1000	1,7	1,7	1,7	1,7	1,7

Grafik hasil pengujian sensor *load cell* ditunjukkan dalam Gambar 5.1.



Gambar 5.1 Grafik Tegangan Keluaran Terhadap Berat Pada Sensor *Load Cell*

Berdasarkan hasil pengujian sensor berat *load cell* didapatkan hasil pengujian karakteristik sensor yang mendekati linier antara tegangan keluaran sensor berat dengan nilai berat yang diukur dalam satuan gram.

5.2 Pengujian Penguat Instrumentasi INA125

Tujuan pengujian penguat instrumentasi INA125 adalah untuk mengetahui hasil tegangan *output* dari Penguat instrumentasi INA125 yang digunakan untuk mengukur berat dari sebuah benda.

5.2.1 Peralatan Pengujian

1. *Load Cell* kapasitas 2kg
2. Beberapa bandul pemberat
3. Timbangan Acuan
4. Catu daya +5 Volt
5. Multimeter
6. Penguat Instrumentasi INA 125

5.2.2 Prosedur Pengujian

Pengujian dilakukan dengan cara menghubungkan sensor *load cell* dengan rangkaian instrumentasi INA125. Kemudian sensor *load cell* diberikan beban secara bertahap dari yang terkecil. Tegangan keluaran rangkaian instrumentasi diukur menggunakan multimeter dan kemudian dicatat.

5.2.3 Hasil Pengujian

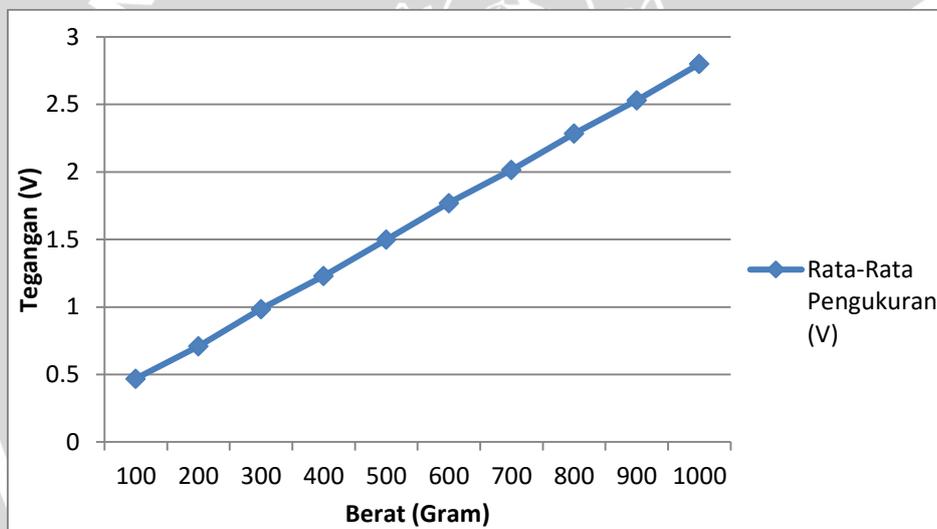
Hasil pengujian penguat instrumentasi INA125 ditunjukkan dalam Tabel 5.2

Tabel 5.2 Hasil Pengujian Penguat Instrumentasi INA125

No.	Berat (Gram)	Tegangan Pengukuran (V)	Rata-Rata
-----	--------------	-------------------------	-----------

		1	2	3	4	Pengukuran (V)
1	100	0,46	0,5	0,46	0,46	0,47
2	200	0,71	0,71	0,71	0,71	0,71
3	300	0,98	1	0,98	0,98	0,985
4	400	1,24	1,2	1,24	1,24	1,23
5	500	1,5	1,5	1,5	1,5	1,5
6	600	1,76	1,8	1,76	1,76	1,77
7	700	2,02	2	2,02	2,02	2,015
8	800	2,28	2,3	2,28	2,28	2,285
9	900	2,54	2,5	2,54	2,54	2,53
10	1000	2,8	2,8	2,8	2,8	2,8

Grafik hasil pengujian penguat instrumentasi INA125 ditunjukkan dalam Gambar 5.2.



Gambar 5.2 Grafik Tegangan Keluaran Terhadap Berat Pada Penguat Instrumentasi INA125

Berdasarkan hasil pengujian penguat instrumentasi INA125 didapatkan grafik pengujian yang linier antara tegangan keluaran sensor berat dengan nilai berat yang diukur dalam satuan gram.

5.3 Pengujian Sensor Suhu LM35

Pengujian sensor suhu bertujuan untuk mengetahui karakteristik dari sensor suhu LM35 ketika melakukan proses pembacaan dan pengukuran suhu.

5.3.1 Peralatan Pengujian

1. Sensor Suhu LM35
2. Termometer Raksa
3. Catu Daya 5V
4. Multimeter
5. Bejana dengan air panas

5.3.2 Prosedur Pengujian

Pengujian dilakukan dengan cara memasukkan sensor suhu LM35 dan juga termometer air raksa ke dalam bejana dengan isi air panas. Kemudian suhu yang terbaca pada termometer air raksa dibandingkan dengan tegangan keluaran pada sensor LM35 yang diukur menggunakan multimeter. Hasil pengujian kemudian dicatat dan dianalisis.

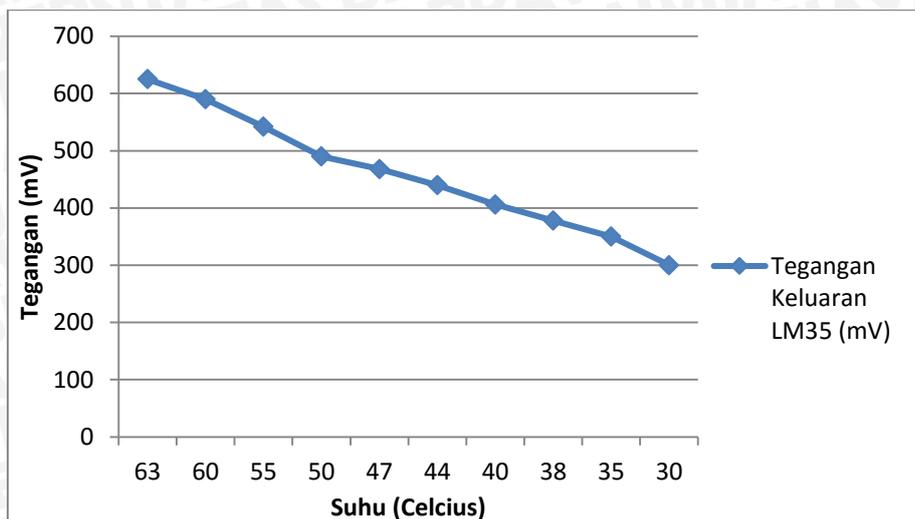
5.3.3 Hasil Pengujian

Hasil pengujian sensor suhu LM35 ditunjukkan dalam Tabel 5.3

Tabel 5.3 Hasil Pengujian Sensor Suhu LM35

No.	Suhu Termometer ($^{\circ}\text{C}$)	Tegangan Keluaran LM35 (mV)
1	63	625
2	60	590
3	55	542
4	50	490
5	47	468
6	44	440
7	40	406
8	38	378
9	35	350
10	30	300

Grafik hasil pengujian sensor suhu LM35 ditunjukkan dalam Gambar 5.3.



Gambar 5.3 Grafik Tegangan Terhadap Suhu Pada Sensor Suhu LM35

Berdasarkan hasil pengujian sensor suhu yang ditunjukkan dalam Tabel 5.3 dan Gambar 5.3 dapat diketahui bahwa sensor suhu dapat berfungsi sebagaimana mestinya. Hal ini dapat dilihat dari tegangan keluaran sensor suhu yang berubah sesuai dengan perubahan suhu. Grafik karakteristik sensor suhu LM35 mempunyai tren grafik linier.

5.4 Pengujian Sensor Suhu LM35 Dengan Penguat Tak Membalik

Tujuan dari pengujian sensor suhu dengan penguat tak membalik adalah untuk mengetahui tegangan keluaran rangkaian penguat tak membalik ketika sensor digunakan untuk membaca suhu. Tegangan keluaran rangkaian penguat tak membalik akan dibaca oleh arduino uno.

5.4.1 Peralatan Pengujian

1. Sensor Suhu LM35
2. Rangkaian Penguat Tak Membalik
3. Termometer Raksa
4. Catu Daya 5V
5. Multimeter
6. Bejana dengan air panas

5.4.2 Prosedur Pengujian

Pengujian dilakukan dengan cara menghubungkan keluaran sensor suhu LM35 ke rangkaian penguat tak membalik. Kemudian sensor suhu LM35 dan juga termometer air raksa dimasukkan ke dalam bejana dengan isi air panas. Kemudian suhu yang terbaca pada termometer air raksa dibandingkan dengan tegangan keluaran pada rangkaian penguat tak membalik yang diukur menggunakan multimeter. Hasil pengujian kemudian dicatat dan dianalisis.

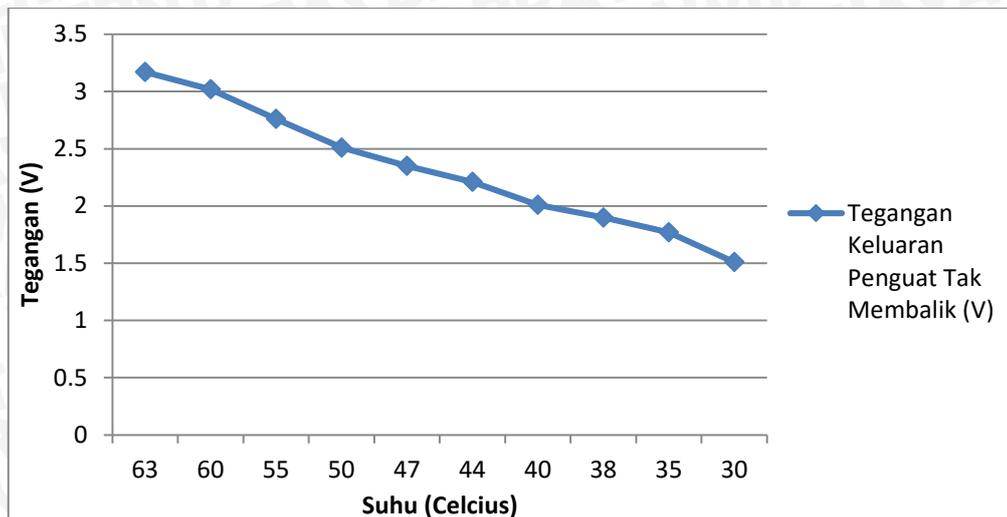
5.4.3 Hasil Pengujian

Hasil pengujian penguat tak membalik ditunjukkan dalam Tabel 5.4

Tabel 5.4 Hasil Pengujian Rangkaian Penguat Tak Membalik

No.	Suhu Termometer (°C)	Tegangan Keluaran Penguat Tak Membalik (V)
1	63	3,17
2	60	3,02
3	55	2,76
4	50	2,51
5	47	2,35
6	44	2,21
7	40	2,01
8	38	1,9
9	35	1,77
10	30	1,51

Grafik hasil pengujian rangkaian penguat tak membalik ditunjukkan dalam Gambar 5.4.



Gambar 5.4 Grafik Tegangan Terhadap Suhu Pada Rangkaian Penguat Tak Membalik

Berdasarkan hasil pengujian rangkaian penguat tak membalik yang ditunjukkan dalam Tabel 5.4 dan Gambar 5.4 dapat diketahui bahwa rangkaian penguat tak membalik dapat berfungsi sebagaimana mestinya. Hal ini dapat dilihat dari tegangan keluaran rangkaian penguat tak membalik yang berubah sesuai dengan perubahan suhu. Grafik karakteristik sensor suhu LM35 mempunyai tren grafik linier.

5.5 Pengujian Keseluruhan

Pengujian keseluruhan bertujuan untuk mengetahui karakteristik respon dari plan yang dikontrol. Pada penelitian ini respon plan yang dikontrol adalah suhu pemanasan pada proses pengeringan jamur tiram. *Setpoint* suhu pada proses pengeringan adalah 67⁰C.

5.5.1 Peralatan Pengujian

1. Arduino Uno
2. Sensor Suhu LM35
3. Heater
4. Sensor Load Cell
5. Jamur Tiram 500 mg
5. Komputer

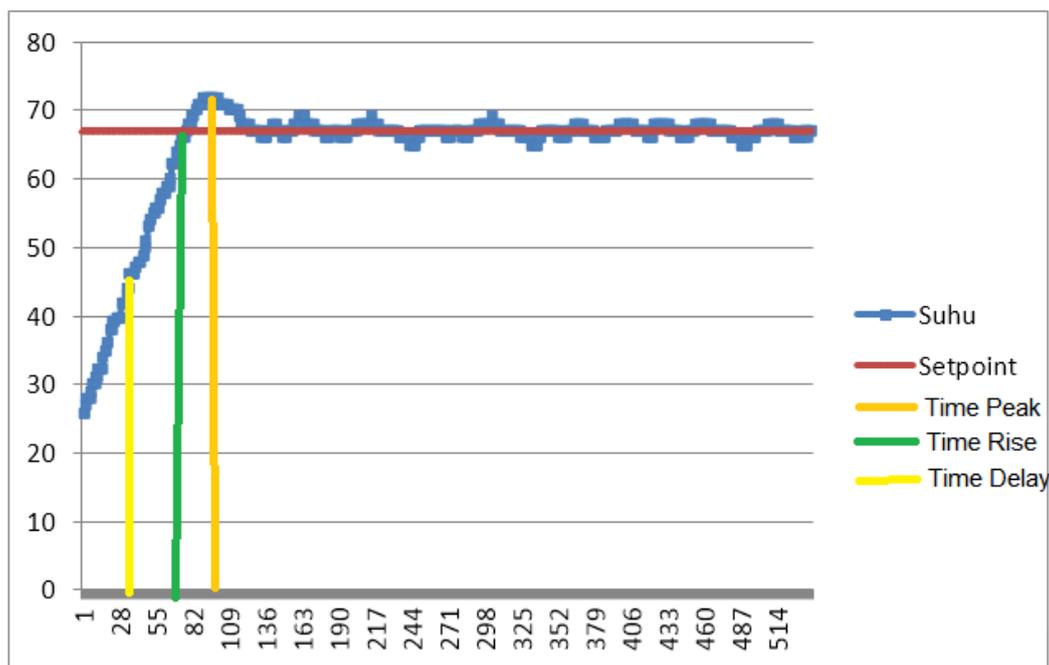
5.5.2 Prosedur Pengujian

Pengujian dilakukan dengan cara menyiapkan jamur seberat tiram 0,5 kg yang dimasukkan kedalam ruang pemanasan. Kemudian sistem diaktifkan untuk memulai

proses pengeringan jamur. Nilai suhu dikirimkan dari arduino uno ke computer tiap menit. Kemudian nilai yang dikirimkan dirubah menjadi grafik dan kemudian dianalisis.

5.5.3 Hasil Pengujian

Hasil pengujian keseluruhan sistem ditunjukkan dalam gambar 5.5.



Gambar 5.5 Grafik Suhu Terhadap Waktu Pada Pengujian Keseluruhan

Berdasarkan hasil pengujian keseluruhan system yang ditunjukkan dalam Gambar 12 dapat diketahui bahwa respon suhu pada plan pengeringan jamur tiram sudah dapat steady di setpoint. didapatkan parameter pengontrolan Time Delay = 38 Menit, Time Rise = 77 Menit, Time Peak = 90 Menit.



BAB VI PENUTUP

A. Kesimpulan

1. Sistem pengendalian suhu menggunakan metode fuzzy pada proses pengeringan jamur tiram menggunakan metode *fuzzy logic*, dengan setpoint sebesar 67°C , menggunakan *sample* jamur seberat 500gram.
2. Pada pengujian keseluruhan proses selesai dalam jangka waktu 9 jam dengan parameter control Time Delay = 38 Menit, Time Rise = 77 Menit, Time Peak = 90 Menit. Proses pengeringan jamur tiram dapat di lakukan lebih cepat daripada proses pengeringan manual, yaitu menjemur di bawah sinar matahari yang membutuhkan waktu 2 hari dan tergantung pada kondisi cuaca.

B. Saran

1. Menggunakan heater lebih dari satu untuk mempercepat proses kenaikan suhu ruangan.
2. Menggunakan metode lain sebagai perbandingan dari hasil pengujian menggunakan metode fuzzy.
3. Selain mengatur suhu, juga dianjurkan mengatur aliran udara yang diharapkan dapat memperbaiki kualitas jamur hasil pengeringan.



DAFTAR PUSTAKA

- [1] Eger, G.; Eden, G.; Wissig, E. (1976): *Pleurotus ostreatus* — *breeding potential of a new cultivated mushroom*. Theor. Appl. Genetics. 47, 155–163.
- [2] Fatimah, Yumi. 2006. *Pengeringan Jamur Tiram (Pleurotus Ostreatus) Menggunakan Oven Gelombang Mikro(Microwave Oven)*. Repositort Institut Pertanian Bogor. Bogor
- [3] Henderson, D.R. and R.L. Perry. 1976. *Agricultural Process Engineering*. AVI.Pub.Co. Inc.Connecticut.
- [4] Thomas Andrew Black. 1968. *Hydraulic Load Cell Lysimeter, Construction, Calibration, Tests and Measurement of Evaporation and Drainage*. University of Wisconsin Madison.
- [5] Ihsan, Fathul. 2009. *Alat Pengukur Suhu dengan Output Suara Berbasis Mikrokontroler ATMEGA8535*. Skripsi. Tidak dipublikasikan. Malang
- [6] Tohari, Affandi Nur Azis. 2011. *Penerapan Fuzzy Logic Pada Pengaturan Jumlah Air Berdasarkan Suhu dan Kelembaban*. Semarang : Teknik Komputer Universitas Negeri Semarang.



LAMPIRAN

Program Keseluruhan

```
#include <LiquidCrystal.h>
#if defined(ARDUINO) && ARDUINO >= 100
#include "Arduino.h"
#else
#include "WProgram.h"
#endif
#define LOW 0
#define MED 1
#define HIGH 2
#define NB 0
#define NS 1
#define Z 2
#define PS 3
#define PB 4
#define ledPin 13

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

int value;
int x0;
int x1;
int x2;
int x3;
float data_Error;
float last_Error;
float data_delError;
float set_point=30;
float sudut;
float alpha=0,alpha2[5][5],alpha_nilaiSudut=0,alpha_nilaiSudut2;
float Error[5];
float delError[5];
float sudut_out[25];
float hasil;
int banyakData,lalala;
float totalSuhu,nilaiSuhu;
long currentMillis,previousMillis = 0;
long interval = 10000000;
long X,Y;
boolean set_FlagA = false,set_FlagB = false,set_FlagD = false,set_FlagE = false;

unsigned char rule[5][5]=
{
    {PB,PB,PB,PS,Z},
    {PB,PB,PS,Z,NS},
```



```
{PB,PS,Z,NS,NB},  
    {PS,Z,NS,NB,NB},  
    {Z,NS,NB,NB,NB}  
};
```

```
float triangle(int value,float x0,float x1, float x2);  
float grade(int value, float x0, float x1);  
float reverse_grade(int value, float x0, float x1);  
void fuzzy();  
void check_rule();  
float defuzzy();
```

```
void setup() {  
    Serial.begin(9600);  
}
```

```
void loop() {  
    int sensorValue = analogRead(A1);  
    float voltage = sensorValue * (5.0 / 1024.0);  
    float constanta = ((0.454545+(voltage / 11.5106383)) /5);  
    float resistance = ((1000 * constanta) / (1-constanta));  
    float temperature = ((resistance - 100)/0.385);  
    if (temperature<=10){  
        temperature = temperature - (3.168831169+1.308038961);  
    }  
    else if (temperature<=20){  
        temperature = temperature - (3.272727273+1.308038961);  
    }  
    else if (temperature<=30){  
        temperature = temperature - (3.324675325+1.308038961);  
    }  
    else if (temperature<=40){  
        temperature = temperature - (3.376623377+1.308038961);  
    }  
    else if (temperature<=50){  
        temperature = temperature - (3.376623377+1.308038961);  
    }  
    else if (temperature<=60){  
        temperature = temperature - (3.350649351+1.308038961);  
    }  
    else if (temperature<=70){  
        temperature = temperature - (3.298701299+1.308038961);  
    }  
    else if (temperature<=80){  
        temperature = temperature - (3.220779221+1.308038961);  
    }  
    else if (temperature<=90){  
        temperature = temperature - (3.090909091+1.308038961);  
    }  
    else if (temperature<=100){  
        temperature = temperature - (2.961038961+1.308038961);  
    }  
    else if (temperature<=110){  
        temperature = temperature - (2.779220779+1.308038961);  
    }
```

```
}  
else if (temperature<=120){  
temperature = temperature - (2.571428571+1.308038961);  
}  
else if (temperature<=130){  
temperature = temperature - (2.337662338+1.308038961);  
}  
else if (temperature<=140){  
temperature = temperature - (2.077922078+1.308038961);  
}  
else if (temperature<=150){  
temperature = temperature - (1.766233766+1.308038961);  
}  
else if (temperature<=160){  
temperature = temperature - (1.454545455+1.308038961);  
}  
else if (temperature<=170){  
temperature = temperature - (1.090909091+1.308038961);  
}  
else if (temperature<=180){  
temperature = temperature - (0.701298701+1.308038961);  
}  
else if (temperature<=190){  
temperature = temperature - (0.285714286+1.308038961);  
}  
else if (temperature<=200){  
temperature = temperature - (-0.155844156+1.308038961);  
}  
else if (temperature<=210){  
temperature = temperature - (-0.623376623+1.308038961);  
}  
else if (temperature<=220){  
temperature = temperature - (-1.142857143+1.308038961);  
}  
else if (temperature<=230){  
temperature = temperature - (-1.662337662+1.308038961);  
}  
else if (temperature<=240){  
temperature = temperature - (-2.233766234+1.308038961);  
}  
else if (temperature<=250){  
temperature = temperature - (-2.831168831+1.308038961);  
}  
else if (temperature<=260){  
temperature = temperature - (-3.454545455+1.308038961);  
}  
else if (temperature<=270){  
temperature = temperature - (-4.12987013+1.308038961);  
}  
else if (temperature<=280){  
temperature = temperature - (-4.805194805+1.308038961);  
}  
else if (temperature<=290){  
temperature = temperature - (-5.532467532+1.308038961);  
}  
else if (temperature<=300){  
temperature = temperature - (-6.25974026+1.308038961);  
}
```

```
}
    if (banyakData <400){
        totalSuhu = totalSuhu + temperature;
        banyakData++;
    }
    else{
        nilaiSuhu = totalSuhu/400;
        Serial.println(nilaiSuhu);
        banyakData=0;
        totalSuhu=0.0;
        delay(100);
    }
}

unsigned long currentMillis = millis();
X = currentMillis;

if (currentMillis<=1125000){
    set_point=(((20.0/1125.0)*(X/1000.0))+30.0);
}

else if (currentMillis<=4500000){
    set_point=(((25.0/1125.0)*(X/1000.0))+25.0);
}

else if (currentMillis<=5400000){
    set_point=125;
}

else if (currentMillis>5400000 && currentMillis<=8550000){
    Y = currentMillis - 5400000;
    set_point=(((50.0/900.0)*(Y/1000.0))+125.0);
}

else if (currentMillis<=9000000){
    set_point=300;
}

else if (currentMillis>9000000){
    set_point=300.5;
}

lcd.setCursor(10,0);
lcd.print(set_point);
delay (10);

data_Error= set_point - nilaiSuhu;
data_delError= data_Error - last_Error;

lcd.begin(16, 2);
lcd.setCursor(0,0);
lcd.print(nilaiSuhu);
lcd.setCursor(6,0);
lcd.print("Cel");
lcd.setCursor(0,1);
lcd.print(millis()/1000);

fuzzy();
check_rule();
sudut=defuzzy();
last_Error=data_Error;
}
}
```

```

float triangle(int value, float x0, float x1, float x2)
{
    float result = 0;
    float x;
    x = value;
    if((x <= x0) || (x >= x2))
        {result = 0;}
    else if(x == x1)
        {result = 1;}
    else if((x > x0) && (x < x1))
        {result = ((x-x0)/(x1-x0));}
    else
        {result = (((-x)+x2)/(x2-x1));}
    return result;
}

```

```

float grade(int value, float x0, float x1)
{
    float result = 0;
    float x;
    x = value;
    if (x <= x0)
        {result = 0;}
    else if(x >= x1)
        {result = 1;}
    else
        {result = ((x-x0)/(x1-x0));}
    return result;
}

```

```

float reverse_grade(int value, float x0, float x1)
{
    float result = 0;
    float x;
    x = value;
    if(x <= x0)
        {result = 1;}
    else if(x >= x1)
        {result = 0;}
    else
        {result = (((-x)+x1)/(x1-x0));}
    return result;
}

```

```

float trapesium(int value, float x0, float x1, float x2, float x3)
{
    float result = 0;
    float x;
    x = value;

    if((x <= x0) || (x >= x3)) {result = 0;}
    else if((x >= x1) && (x <= x2)){result = 1;}
    else if((x > x0) && (x < x1)){result = ((x-x0)/(x1-x0));}
    else {result = ((-x+x3)/(x3-x2));}
    return result;
}

```

UNIVERSITAS BRAWIJAYA



```
void fuzzy()
{
  if (set_point<=50){

    Error[NB] = reverse_grade(data_Error,-3,-0.5);
    Error[NS] = triangle(data_Error,-3,-1.5,0);
    Error[Z] = triangle(data_Error,-0.5,0,1.5);
    Error[PS] = triangle(data_Error,0,1.5,4);
    Error[PB] = grade(data_Error,1.5,4);

    delError[NB] = reverse_grade(data_delError,-0.5,-0.2);
    delError[NS] = triangle(data_delError,-0.5,-0.2,0);
    delError[Z] = triangle(data_delError,-0.2,0,0.2);
    delError[PS] = triangle(data_delError,0,0.2,0.5);
    delError[PB] = grade(data_delError,0.2,0.5);
  }

  else if (set_point>50 && set_point<=124){

    Error[NB] = reverse_grade(data_Error,-3,-0.5);
    Error[NS] = triangle(data_Error,-3,-1.5,0);
    Error[Z] = triangle(data_Error,-0.5,0,1.5);
    Error[PS] = triangle(data_Error,0,1.5,4);
    Error[PB] = grade(data_Error,1.5,4);

    delError[NB] = reverse_grade(data_delError,-0.5,-0.3);
    delError[NS] = triangle(data_delError,-0.5,-0.3,0);
    delError[Z] = triangle(data_delError,-0.3,0,0.3);
    delError[PS] = triangle(data_delError,0,0.3,0.5);
    delError[PB] = grade(data_delError,0.3,0.5);
  }

  else if (set_point>124 && set_point<=125){

    Error[NB] = reverse_grade(data_Error,-0.75,-0.4);
    Error[NS] = triangle(data_Error,-0.75,-0.4,0);
    Error[Z] = triangle(data_Error,-0.4,0,0.4);
    Error[PS] = triangle(data_Error,0,0.4,0.75);
    Error[PB] = grade(data_Error,0.4,0.75);

    delError[NB] = reverse_grade(data_delError,-0.4,-0.2);
    delError[NS] = triangle(data_delError,-0.4,-0.2,0);
    delError[Z] = triangle(data_delError,-0.2,0,0.2);
    delError[PS] = triangle(data_delError,0,0.2,0.4);
    delError[PB] = grade(data_delError,0.2,0.4);
  }

  else if (set_point>125 && set_point<=299.5){

    Error[NB] = reverse_grade(data_Error,-3,-0.5);
    Error[NS] = triangle(data_Error,-3,-1.5,0);
    Error[Z] = triangle(data_Error,-0.5,0,1.5);
    Error[PS] = triangle(data_Error,0,1.5,4);
    Error[PB] = grade(data_Error,1.5,4);

    delError[NB] = reverse_grade(data_delError,-0.7,-0.4);
```

```

delError[NS] = triangle(data_delError,-0.7,-0.4,0);
delError[Z] = triangle(data_delError,-0.4,0,0.4);
delError[PS] = triangle(data_delError,0,0.4,0.7);
delError[PB] = grade(data_delError,0.4,0.7);
}
else if (set_point>299.5 && set_point<=300){

Error[NB] = reverse_grade(data_Error,-0.75,-0.4);
Error[NS] = triangle(data_Error,-0.75,-0.4,0);
Error[Z] = triangle(data_Error,-0.4,0,0.4);
Error[PS] = triangle(data_Error,0,0.4,0.75);
Error[PB] = grade(data_Error,0.4,0.75);

delError[NB] = reverse_grade(data_delError,-0.4,-0.2);
delError[NS] = triangle(data_delError,-0.4,-0.2,0);
delError[Z] = triangle(data_delError,-0.2,0,0.2);
delError[PS] = triangle(data_delError,0,0.2,0.4);
delError[PB] = grade(data_delError,0.2,0.4);
}
}

void check_rule()
{
float u_min = 0;
float u_max = 0;
char x,y;
for (x = 0; x < 10; x++)
{
sudut_out[x] = 0;
}

for (x = 0; x < 5; x++)
{
for (y = 0; y < 5; y++)
{

if (Error[y] > 0 && delError[x] > 0)
{
u_min = (float)fmin(Error[y],delError[x]);
sudut_out[rule[x][y]] = (float)fmax(u_min,sudut_out[rule[x][y]]);
}
}k
}
}

float defuzzy()
{
float Sudut;
int nilaiSudut;

if (set_point<=50 ){
alpha_nilaiSudut
sudut_out[0]*150+sudut_out[1]*151+sudut_out[2]*158+sudut_out[3]*161+sudut_out[4]*163;
}

else if (set_point>50 && set_point<=75){

```

```
alpha_nilaiSudut =
sudut_out[0]*149+sudut_out[1]*150+sudut_out[2]*156+sudut_out[3]*161+sudut_out[4]*163;
}

else if (set_point>75 && set_point<=100){
alpha_nilaiSudut =
sudut_out[0]*148+sudut_out[1]*149+sudut_out[2]*155+sudut_out[3]*159+sudut_out[4]*160;
}

else if (set_point>100 && set_point<=124){
alpha_nilaiSudut =
sudut_out[0]*147+sudut_out[1]*146+sudut_out[2]*153.5+sudut_out[3]*158+sudut_out[4]*160;
}

else if (set_point>124 && set_point<=125){
alpha_nilaiSudut =
sudut_out[0]*153+sudut_out[1]*154+sudut_out[2]*155+sudut_out[3]*158+sudut_out[4]*160;
}

else if (set_point>125 && set_point<=150){
alpha_nilaiSudut =
sudut_out[0]*133+sudut_out[1]*134+sudut_out[2]*139.5+sudut_out[3]*152+sudut_out[4]*158;
}

else if (set_point>150 && set_point<=175){
alpha_nilaiSudut =
sudut_out[0]*128+sudut_out[1]*132.5+sudut_out[2]*137+sudut_out[3]*152+sudut_out[4]*158;
}

else if (set_point>175 && set_point<=200){
alpha_nilaiSudut =
sudut_out[0]*122+sudut_out[1]*130+sudut_out[2]*135+sudut_out[3]*153+sudut_out[4]*152;
}

else if (set_point>200 && set_point<=225){
alpha_nilaiSudut =
sudut_out[0]*127+sudut_out[1]*129+sudut_out[2]*132+sudut_out[3]*137+sudut_out[4]*145;
}
else if (set_point>225 && set_point<=250){
alpha_nilaiSudut =
sudut_out[0]*125+sudut_out[1]*127+sudut_out[2]*130+sudut_out[3]*132+sudut_out[4]*145;
}

else if (set_point>250 && set_point<=275){
alpha_nilaiSudut =
sudut_out[0]*121+sudut_out[1]*123+sudut_out[2]*128+sudut_out[3]*130+sudut_out[4]*140;
}

else if (set_point>275 && set_point<=299.5){
alpha_nilaiSudut =
sudut_out[0]*120+sudut_out[1]*121+sudut_out[2]*126+sudut_out[3]*128+sudut_out[4]*140;
}

else if (set_point>299.5 && set_point<=300){
alpha_nilaiSudut =
sudut_out[0]*127+sudut_out[1]*125+sudut_out[2]*133+sudut_out[3]*138+sudut_out[4]*140.5;
}
}
```

```
else if (set_point>300){
    alpha_nilaiSudut =
    sudut_out[0]*170+sudut_out[1]*170+sudut_out[2]*170+sudut_out[3]*170+sudut_out[4]*170;
}
alpha = sudut_out[0]+sudut_out[1]+sudut_out[2]+sudut_out[3]+sudut_out[4];
hasil = (float) (alpha_nilaiSudut / alpha);
Sudut=hasil;
    lcd.setCursor(5,1);
    lcd.print(hasil);
```



Hasil Pengujian Sistem Keseluruhan

Menit	Suhu												
1	26	41	47	81	68	121	68	161	68	201	67	241	66
2	26	42	48	82	69	122	68	162	69	202	67	242	66
3	27	43	48	83	69	123	68	163	69	203	67	243	65
4	28	44	48	84	70	124	68	164	69	204	68	244	65
5	28	45	49	85	70	125	67	165	69	205	68	245	65
6	28	46	49	86	71	126	67	166	68	206	68	246	65
7	29	47	50	87	71	127	67	167	68	207	68	247	65
8	30	48	51	88	71	128	67	168	68	208	68	248	66
9	30	49	53	89	71	129	67	169	68	209	68	249	66
10	30	50	53	90	72	130	67	170	68	210	68	250	67
11	31	51	54	91	72	131	67	171	68	211	68	251	67
12	31	52	54	92	72	132	67	172	67	212	68	252	67
13	32	53	55	93	72	133	66	173	67	213	68	253	67
14	32	54	55	94	72	134	66	174	67	214	69	254	67
15	32	55	56	95	72	135	66	175	67	215	68	255	67
16	34	56	56	96	72	136	66	176	67	216	68	256	67
17	35	57	56	97	72	137	66	177	67	217	68	257	67
18	35	58	57	98	72	138	67	178	67	218	68	258	67
19	36	59	57	99	72	139	67	179	67	219	68	259	67
20	36	60	58	100	72	140	67	180	66	220	67	260	67
21	38	61	58	101	71	141	67	181	66	221	67	261	67
22	38	62	58	102	71	142	68	182	66	222	67	262	67
23	39	63	59	103	71	143	68	183	66	223	67	263	67
24	39	64	59	104	71	144	68	184	67	224	67	264	67
25	39	65	59	105	71	145	67	185	67	225	67	265	67
26	40	66	60	106	71	146	67	186	67	226	67	266	67
27	40	67	62	107	71	147	67	187	67	227	67	267	67
28	40	68	62	108	71	148	67	188	67	228	67	268	67
29	40	69	62	109	70	149	66	189	67	229	67	269	67
30	42	70	64	110	70	150	66	190	67	230	67	270	66
31	42	71	64	111	70	151	66	191	67	231	67	271	66
32	42	72	65	112	70	152	67	192	66	232	67	272	66
33	43	73	65	113	70	153	67	193	66	233	67	273	66
34	44	74	66	114	70	154	67	194	66	234	67	274	67
35	44	75	66	115	70	155	67	195	66	235	66	275	67
36	46	76	66	116	69	156	67	196	66	236	66	276	67
37	46	77	67	117	69	157	68	197	67	237	66	277	67
38	46	78	67	118	68	158	68	198	67	238	66	278	67
39	46	79	68	119	68	159	68	199	67	239	66	279	67
40	47	80	68	120	68	160	68	200	67	240	66	280	67

Men it	Suh u
281	67
282	66
283	66
284	66
285	66
286	66
287	67
288	67
289	67
290	67
291	67
292	67
293	67
294	68
295	68
296	68
297	68
298	68
299	68
300	68
301	68
302	68
303	68
304	69
305	68
306	68
307	68
308	68
309	68
310	67
311	67
312	67
313	67
314	67
315	67
316	67
317	67
318	67
319	67
320	67

Men it	Suh u
321	67
322	67
323	67
324	67
325	66
326	66
327	66
328	66
329	66
330	66
331	66
332	66
333	65
334	65
335	65
336	65
337	65
338	66
339	66
340	67
341	67
342	67
343	67
344	67
345	67
346	67
347	67
348	67
349	67
350	67
351	67
352	67
353	66
354	66
355	66
356	66
357	66
358	66
359	67
360	67

Men it	Suh u
361	67
362	67
363	67
364	67
365	68
366	68
367	68
368	68
369	68
370	68
371	67
372	67
373	67
374	67
375	67
376	67
377	67
378	67
379	66
380	66
381	66
382	66
383	66
384	66
385	66
386	66
387	66
388	67
389	67
390	67
391	67
392	67
393	67
394	67
395	67
396	67
397	68
398	68
399	68
400	68

Men it	Suh u
401	68
402	68
403	68
404	68
405	68
406	68
407	68
408	68
409	67
410	67
411	67
412	67
413	67
414	67
415	67
416	67
417	67
418	67
419	66
420	66
421	66
422	66
423	68
424	68
425	68
426	68
427	68
428	68
429	68
430	68
431	68
432	68
433	68
434	68
435	67
436	67
437	67
438	67
439	67
440	67

Men it	Suh u
441	67
442	67
443	66
444	66
445	66
446	66
447	66
448	66
449	67
450	67
451	67
452	67
453	67
454	68
455	68
456	68
457	68
458	68
459	68
460	68
461	68
462	68
463	68
464	68
465	68
466	67
467	67
468	67
469	67
470	67
471	67
472	67
473	67
474	67
475	67
476	67
477	67
478	67
479	67
480	66

Men it	Suh u
481	66
482	66
483	66
484	66
485	66
486	66
487	65
488	65
489	65
490	65
491	65
492	65
493	66
494	66
495	66
496	66
497	66
498	66
499	66
500	67
501	67
502	67
503	67
504	67
505	67
506	67
507	67
508	68
509	68
510	68
511	68
512	68
513	68
514	68
515	68
516	67
517	67
518	67
519	67
520	67

Men it	Suh u
521	67
522	67
523	67
524	67
525	67
526	66
527	66
528	66
529	66
530	66
531	67
532	66
533	66
534	67
535	67
536	67
537	67
538	66
539	67
540	67



