



**SISTEM PENGENDALI TEMPERATUR DAN KELEMBABAN  
BERBASIS *PID EXPERT* UNTUK INKUBATOR BAYI  
PREMATUR**

**SKRIPSI**

Oleh :

**Artika Sari Ramadhani**

**NIM 161910201010**

**PROGRAM STUDI STRATA 1 TEKNIK ELEKTRO  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS JEMBER  
2020**



**SISTEM PENGENDALI TEMPERATUR DAN KELEMBABAN  
BERBASIS *PID EXPERT* UNTUK INKUBATOR BAYI  
PREMATUR**

**SKRIPSI**

Diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat untuk menyelesaikan Program Studi Teknik Elektro (S1) dan mencapai gelar Sarjana Teknik

Oleh:

**Artika Sari Ramadhani  
NIM 161910201010**

**PROGRAM STUDI TEKNIK ELEKTRO STRATA 1  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS JEMBER  
2020**

## PERSEMBAHAN

Dengan mengucapkan puji dan syukur kehadirat Allah SWT atas nikmat dan karuniaNya yang telah melimpahkan rahmat yang tak ternilai sehingga saya bisa menyelesaikan skripsi ini. Banyak doa, semangat dan bantuan yang saya dapatkan dari berbagai pihak dalam melangsungkan penelitian. Dengan tulus ikhlas dan penuh kerendahan hati, skripsi saya persembahkan untuk:

1. Allah SWT, yang Maha Pengasih lagi Maha Penyayang;
2. Nabi besar Muhammad SAW, yang menjadi suri tauladan bagi seluruh umat;
3. Kedua orang tua yang amat saya sayangi, yakni Bapak Joko Wiyono dan Ibunda Masri'in yang selalu membimbing dan mendidik saya. Serta adikku Maulana Firmansyah dan Shakilla Ainara Atmarini yang selalu menghibur dan memberikan semangat. Serta seluruh keluarga besar saya yang selalu memberi nasihat serta dukungan;
4. Dr. Ir. Satryo Budi Utomo, S.T., M.T selaku Dosen Pembimbing Utama dan Arizal Mujibtama Nanda Imron, S.T., M.T selaku Dosen Pembimbing Anggota yang telah membimbing dengan sangat sabar dan memberikan saran-saran demi kesempurnaan skripsi ini;
5. Bapak Alfredo Bayu Satriya, S.T., M.T., selaku dosen pembimbing akademik yang telah membimbing dan menanamkan rasa tanggung jawab selama penulis menjadi mahasiswa.
6. Rosida Amalia Nurul Qoyima teman satu perjuangan sejak mahasiswa baru sampai dengan pengerjaan tugas akhir , *girls squad* yang selalu membantu dan menghibur semasa kuliah, serta Syaiful, Tigo, Fikry, Zaqi, Cahya, Mas Turasno yang telah membantu saya dalam penelitian ini terimakasih untuk waktu dan ilmu yang telah diberikan;

7. Kakak sepupu saya Fiolina dan Amalia terimakasih karena selalu memberikan saya saran, kritik, serta dukungan.
8. Elok, Fahir, Nadya, Alma, dan seluruh BPH BEM FT 2019. Terima kasih telah memberikan semangat dan dukungan.
9. Alda, Anggi, Iza, Tiara, Indah, Rizka, dan teman – teman KKN Jebung Lor 2019 terima kasih sudah memberikan semangat;
10. Sahabat saya Linda, Inggil, Lutfia, dan teman – teman dari saya SMA yang selalu menemani dan membantu penulis selama menjadi mahasiswa.
11. Untuk seluruh teman-teman INDUKTRO 2016 yang telah menjadikan kehidupan semasa kuliah sangat berwarna;
12. Para guru sejak Taman Kanak-kanak hingga Perguruan Tinggi yang terhormat, terima kasih telah memberikan ilmu;

**MOTTO**

“Dua tentara yang tak terkalahkan yaitu hati yang ikhlas dan doa yang tulus”

**(Ibnu Taimiyah)**

“Barangsiapa yang menempuh jalan untuk menuntut ilmu, Allah Ta’ala akan memudahkan baginya jalan menuju surga”

**(HR. Muslim)**

“Terkadang langkah terkecil kearah yang benar berakhir menjadi langkah terbesar dalam hidupmu”

**(Artika Sari Ramadhani)**



## PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Artika Sari Ramadhani

NIM : 161910201010

Menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Sistem Pengendali Temperatur dan Kelembaban Berbasis *PID Expert* untuk Inkubator Bayi Prematur” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada industri manapun dan bukan karya jiplakan. Saya bertanggung jawab penuh atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapatkan sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, 07 Juni 2020

Yang menyatakan,

Artika Sari Ramadhani

NIM 161910201010

**SKRIPSI**

**SISTEM PENGENDALI TEMPERATUR DAN KELEMBABAN  
BERBASIS *PID EXPERT* UNTUK INKUBATOR BAYI PREMATUR**

Oleh

**Artika Sari Ramadhani**

**NIM 161910201010**

Pembimbing:

Dosen Pembimbing Utama : Dr. Ir. Satryo Budi Utomo, S.T., M.T.

Dosen Pembimbing Anggota : Arizal Mujibtamala Nanda Imron, S.T., M.T.

## PENGESAHAN

Skripsi berjudul “**Sistem Pengendali Temperatur dan Kelembaban Berbasis *PID Expert* untuk Inkubator Bayi Prematur**” Ramadhani Sari

Ramadhani telah diuji dan disahkan pada:

Hari : Rabu

Tanggal : 08 Juli 2020

Tempat : Fakultas Teknik Universitas Jember

Tim penguji,

Ketua,

Anggota I,

Dr. Ir. Satryo Budi Utomo, S.T., M.T.

NIP. 198501262008011002

Anggota II,

Arizal Mujibtamala Nanda Imron, S.T., M.T.

NIP. 760017099

Anggota III,

Dodi Setiabudi, S.T., M.T.

NIP. 198405312008121004

Guido Dias Kalandro, S.ST., M.Eng

NIP. 760015734

Mengesahkan,

Dekan,

Dr. Triwahju Hardianto, S.T., M.T.

NIP. 1970082619970210011

## RINGKASAN

### **Sistem Pengendali Temperatur dan Kelembaban Berbasis *PID Expert* untuk Bayi Prematur**

Ariska Sari Ramadhani; 161910201010; 2020; 85 Halaman; Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Angka kejadian bayi lahir prematur dengan berat badan lahir rendah (BBLR) di Indonesia masih tinggi. Prevalensi bayi prematur dan BBLR di Indonesia sekitar 10,2% dari jumlah kelahiran, bahkan beberapa daerah mencapai 17%. Angka ini lebih besar bila dibandingkan pada negara berkembang sekitar 5-9%. Bayi prematur adalah bayi yang lahir pada usia gestasi 20-37 minggu. Pada umumnya bayi prematur berat badan lahirnya rendah kurang dari 2500 gram. Faktor – faktor yang perlu diperhatikan pada inkubator adalah suhu inkubator, kelembaban inkubator, serta suhu bayi. Suhu inkubator bayi adalah 32°C-35°C tapi tetap dilihat dari dari berat badan bayi , semakin semakin berat semakin rendah suhu dari inkubator bayi.

Inkubator bayi merupakan salah satu teknologi yang dibutuhkan pada dunia kedokteran, khususnya pada masalah bayi lahir prematur. Alat ini membantu perawat maupun bidan dalam melakukan penyelamatan. Tetapi harga untuk mendapatkan inkubator bayi cukup mahal bahkan sampai puluhan juta rupiah, serta pemeliharaan inkubator mahal dan masalah lainnya adalah inkubator membutuhkan listrik yang memadai. Bagi sejumlah rumah sakit besar untuk mendapatkan beberapa inkubator bayi ini tidak sulit, tetapi bagi bidan atau puskesmas di daerah pelosok hal ini sangat memberatkan. Untuk itu perlu dirancang dan dibuat sistem inkubator bayi dengan biaya yang lebih murah.

Berdasarkan latar belakang di atas maka perlu dilakukan penelitian dan perancangan alat dengan judul “Sistem Pengendali Temperatur dan Kelembaban Berbasis *PID Expert* untuk Inkubator Bayi Prematur” yang bertujuan untuk membuat suatu teknologi inkubator yang menghasilkan kontrol dan sistem monitoring yang baik dengan harga murah. Dengan metode kontrol PID expert maka untuk menentukan masing-masing parameter PID dengan melakukan proses identifikasi karakteristik plant dan berdasarkan pengalaman para ahli. Sistem kontrol PID Expert yang diaplikasikan dengan sensor DHT22 sebagai modul kontrol temperatur dan monitoring kelembaban di dalam sistem yang menggunakan Arduino sebagai piranti data akuisisi untuk membaca sensor temperatur dan kelembaban serta mengendalikan *AC Light Dimmer Module* untuk pengendali lampu agar bisa meratakan temperatur secara keseluruhan dalam sistem dan modul *driver* L298N sebagai pengendali untuk kipas. Data sample pengontrolan temperatur dan memonitoring temperatur kelembapan dapat dilihat pada LCD.

Dilakukan beberapa macam pengujian dengan *set point* yang berbeda – beda yaitu 32°C, 33°C, 34°C, dan 35°C untuk mengetahui respon sistem dengan beban bayi yang disimulasikan dengan air yang memiliki suhu 36,5°C – 37,5°C yang merupakan suhu normal menyerupai bayi baru lahir. Untuk mencapai set point 32°C diperlukan waktu 110 detik tanpa beban dan 102 detik dengan beban. Untuk mencapai set point 33°C diperlukan waktu 156 detik tanpa beban dan 142 detik dengan beban. Untuk mencapai setpoint 34°C diperlukan waktu 194 detik tanpa beban dan 198 detik dengan beban. Untuk mencapai set point 35°C diperlukan waktu 246 detik tanpa beban dan 242 detik dengan beban.

Dengan metode PID dan *Expert PID Control* memiliki waktu untuk mencapai *rise time* yang sama yaitu pada 246 detik. Untuk *settling time* saat pengujian menggunakan metode *Expert PID Control* lebih cepat yaitu pada 272 detik, sedangkan *settling time* pada pengujian dengan menggunakan metode PID mencapai *settling time* pada 768 detik. Untuk besar *overshoot* pada pengujian menggunakan metode *Expert PID Control* lebih kecil yaitu sebesar 0,21%, daripada saat pengujian dengan menggunakan metode PID yaitu sebesar 1,79%.

## SUMMARY

### **Expert PID Based Temperature and Humidity Control System for Premature Baby Incubators**

Artika Sari Ramadhani; 161910201010; 2020; 85 pages; Department of Electrical Engineering, Faculty of Engineering, Jember University.

The incidence of premature babies with low birth weight (LBW) in Indonesia is still high. The prevalence of premature babies and LBW in Indonesia around 10.2% of the number of births, even some areas reached 17%. This figure is greater when compared to developing countries around 5-9%. Premature babies are babies born at 20-37 weeks' gestation. In general, premature babies with low birth weight of less than 2500 grams. Factors that need to be considered in an incubator are incubator temperature, incubator humidity, and baby's temperature. The temperature of the baby incubator is 32°C - 35°C but still seen from the weight of the baby, the heavier the lower the temperature of the baby incubator.

Baby incubator is one of the technologies needed in medicine, especially in the case of babies born prematurely. This tool helps nurses and midwives to rescue. But the price for getting a baby incubator is quite expensive even up to tens of millions of rupiah, and the maintenance of an expensive incubator and other problems is that the incubator requires adequate electricity. For a number of large hospitals to get some baby incubators is not difficult, but for midwives or puskesmas in remote areas this is very burdensome. For this reason, it is necessary to design and create a baby incubator system at a lower cost.

Based on the above background, it is necessary to research and design a tool with the title "Expert PID Based Temperature and Humidity Control System for Premature Baby Incubators" which aims to create an incubator technology that produces good control and monitoring systems at low prices. With the expert PID control method, then to determine each PID parameter by identifying the plant characteristics and based on the experience of the experts. Expert PID control system is applied with the DHT22 sensor as a temperature control module and humidity monitoring in a system that uses Arduino as a data acquisition tool to read temperature and humidity sensors and to control the AC Light Dimmer Module for lighting controllers to be able to level the overall temperature in the system and L298N driver module as a controller for the fan. Data for controlling temperature and monitoring humidity temperatures can be seen on the LCD.

Several kinds of tests were carried out with different set points namely 32 °C, 33 °C, 34 °C and 35 °C to determine the response of the system to the baby's load simulated with water that has a temperature of 36.5 °C - 37.5 °C which is a normal temperature resembling newborn baby. To reach 32 °C set point, it takes 110 seconds without load and 102 seconds with load. To reach the set point 33 °C takes 156 seconds without load and 142 seconds with load. To reach setpoint 34 °C takes 194 seconds without load and 198 seconds with load. To reach the set point of 35 °C takes 246 seconds without load and 242 seconds with load.

With the PID method and Expert PID Control has the time to reach the same rise time at 246 seconds. The settling time when testing using the Expert PID Control method is faster at 272 seconds, while the settling time on testing using the PID method reaches a settling time at 768 seconds. For overshoot, the testing using the Expert PID Control method is smaller that is equal to 0.21%, than when testing using the PID method which is equal to 1.79%.

## PRAKATA

Puji syukur kehadiran Allah SWT atas nikmat dan karuniaNya yang telah melimpahkan rahmat yang tak ternilai sehingga saya bisa menyelesaikan penelitian sekaligus penyusunan skripsi yang berjudul ” **Sistem Pengendali Temperatur dan Kelembaban Berbasis *PID Expert* untuk Inkubator Bayi Prematur** ” disusun untuk memenuhi salah satu syarat dalam penyelesaian pendidikan strata satu (S1) pada Jurusan Teknik Elektro Fakultas Teknik Universitas Jember. Selama penyusunan skripsi ini tentunya terdapat banyak pihak yang telah memberikan bantuan berupa saran, motivasi, fasilitas, doa dan dukungan lainnya. Oleh karena itu penulis menyampaikan terima kasih kepada:

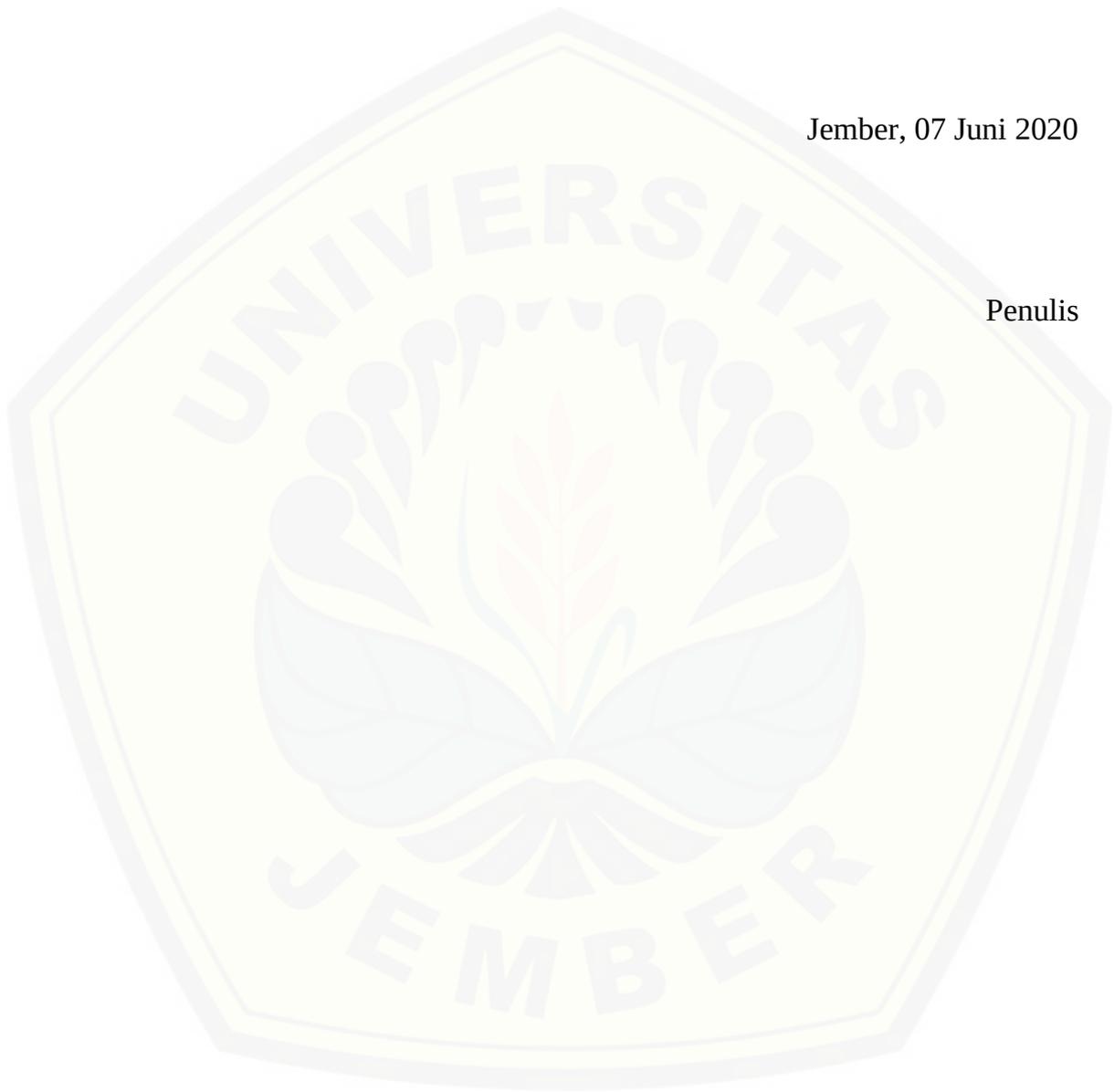
1. Bapak Dr. Triwahju Hardianto, S.T., M.T. selaku Dekan Fakultas Teknik Universitas Jember;
2. Bapak Dr. Bambang Sri Kaloko S.T., M.T . selaku Ketua Jurusan Teknik Elektro Universitas Jember;
3. Dr. Ir. Satriyo Budi Utomo, S.T., M.T selaku Dosen Pembimbing Utama dan Arizal Mujibtama Nanda Imron, S.T., M.T selaku Dosen Pembimbing Anggota yang telah membimbing dengan sangat sabar dan memberikan saran-saran demi kesempurnaan skripsi ini;
4. Bapak Dodi Setiabudi, S.T., M.T. dan Bapak Guido Dias Kalandro,. S.ST., M.Eng. selaku dosen penguji yang telah memberikan saran untuk memperbaiki dan menyempurnakan skripsi ini;
5. Keluarga besar Teknik Elektro 2016 (INDUKTRO), terima kasih telah menjadi keluarga yang sangat baik dan luar biasa;
6. Serta seluruh teman-teman seperjuangan yang tidak dapat saya sebutkan satu-satu. Saya sampaikan terima kasih banyak atas semangat, doa dan saran yang telah diberikan.

Penulis menyadari bahwa selama penyusunan skripsi ini masih terdapat banyak kekurangan. Harapan penulis adalah semoga skripsi ini dapat bermanfaat dalam mengembangkan ilmu pengetahuan khususnya dalam bidang teknik

elektro. Kritik dan saran diharapkan dapat terus berlanjut sehingga dapat memperbaiki dan menyempurnakan skripsi ini untuk penelitian selanjutnya.

Jember, 07 Juni 2020

Penulis



DAFTAR ISI

	Halaman
<b>HALAMAN SAMPUL</b> .....	i
<b>HALAMAN JUDUL</b> .....	ii
<b>PERSEMBAHAN</b> .....	iii
<b>MOTTO</b> .....	v
<b>SKRIPSI</b> .....	vii
<b>RINGKASAN</b> .....	viix
<b>SUMMARY</b> .....	viii
<b>PRAKATA</b> .....	xiii
<b>DAFTAR ISI</b> .....	xv
<b>DAFTAR GAMBAR</b> .....	xvii
<b>DAFTAR TABEL</b> .....	xix
<b>BAB 1. PENDAHULUAN</b> .....	1
<b>1.1 Latar Belakang</b> .....	1
<b>1.2 Rumusan Masalah</b> .....	3
<b>1.4 Tujuan</b> .....	4
<b>1.5 Manfaat</b> .....	4
<b>BAB 2. TINJAUAN PUSTAKA</b> .....	5
<b>2.1 Sensor Suhu dan Kelembaban.DHT22</b> .....	5
<b>2.2 AC Light Dimmer Module</b> .....	6
<b>2.3 <i>Expert PID Control</i></b> .....	7
<b>2.4 Arduino Uno</b> .....	9
<b>2.5 LCD (<i>Liquid Crystal Display</i>)</b> .....	12
<b>2.7 Inkubator Bayi</b> .....	15
<b>2.8 Module Driver Motor L298N</b> .....	17
<b>2.9 Heater</b> .....	19
<b>BAB 3. METODOLOGI PENELITIAN</b> .....	21
<b>3.1 Tempat Penelitian</b> .....	21
<b>3.2 Waktu Penelitian</b> .....	21
<b>3.3 Tahapan Penelitian</b> .....	22
<b>3.4 Alat dan Bahan Penelitian</b> .....	24
<b>3.5 Rancangan Penelitian</b> .....	24
3.5.1 Desain Mekanik Alat .....	24
3.5.2 Desain Diagram Elektrikal Alat .....	25
3.5.3 Diagram Blok Sistem .....	30
<b>3.6 Diagram Blok <i>Expert PID Control</i></b> .....	31
3.6.1 Diagram Blok Kontrol Suhu dengan metode Expert PID Control	31

3.6.2 Diagram Alir Expert PID Control .....	32
<b>3.7 Variabel Penelitian dan Pengamatan.....</b>	<b>34</b>
3.7.1 Pengujian Sensor Suhu .....	34
3.7.2 Pengujian Sensor Kelembaban .....	35
3.7.3 Pengujian Aktuator Suhu .....	35
3.7.4 Pengujian Metode <i>Expert PID Control</i> .....	35
<b>BAB 4. HASIL DAN PEMBAHASAN.....</b>	<b>37</b>
<b>4.1 Pengujian Sensor Suhu.....</b>	<b>37</b>
<b>4.2 Pengujian Sensor Kelembaban.....</b>	<b>40</b>
<b>4.3 Pengujian Aktuator Suhu.....</b>	<b>40</b>
<b>4.3 Pengujian Sistem <i>Expert PID Control</i>.....</b>	<b>43</b>
4.3.1 Mencari Nilai Penguat PID dengan PID Tuning pada <i>Dimmer</i> lampu 44	
4.3.2 Mencari Nilai Penguat PID dengan PID Tuning pada <i>Drive</i> Kipas ..	49
4.3.3 Pengujian Penggabungan kontrol PID <i>Dimmer</i> lampu dan <i>Drive</i> Kipas .....	53
4.3.4 Pengujian Expert PID Control untuk Kendali Suhu tanpa Beban dan dengan Beban .....	54
<b>4.4 Perbandingan Metode <i>PID Expert Control</i> dengan Kontrol PID untuk Suhu.....</b>	<b>62</b>
<b>4.5 Monitoring Nilai Kelembaban.....</b>	<b>63</b>
<b>BAB 5. PENUTUP.....</b>	<b>65</b>
<b>5.1 Kesimpulan.....</b>	<b>65</b>
<b>5.2 Saran.....</b>	<b>65</b>
<b>DAFTAR PUSTAKA.....</b>	<b>67</b>
<b>LAMPIRAN.....</b>	<b>70</b>

**DAFTAR GAMBAR**

	Halaman
2.1 Sensor DHT22 .....	5
2.2 <i>AC Light Dimmer Module</i> .....	6
2.3 Arduino Uno .....	10
2.4 Konfigurasi pin Atmega 328P .....	11
2.5 <i>Liquid Crystal Display</i> .....	13
2.6 <i>Fan</i> .....	15
2.7 Inkubator Bayi .....	16
2.8 Modul Driver L298N .....	18
3.1 Tahapan Rangkaian Penelitian .....	23
3.2 Desain Mekanik .....	26
3.3 Rangkaian DHT22 dengan Arduino .....	27
3.4 Rangkaian <i>Heater</i> dengan Modul <i>AC Light Dimmer</i> dan Arduino Uno.....	27
3.5 Skema Rangkaian <i>Dimmer</i> dengan Arduino .....	28
3.6 Rangkaian Kipas Aksial dengan Modul Driver L298N dan Arduino Uno.....	29
3.7 Diagram Blok Sistem .....	30
3.8 Gambar Diagram Blok <i>Expert PID Control</i> Suhu .....	32
3.9 Diagram Alir <i>Expert PID Control</i> untuk Suhu.....	33
4.1 Grafik Perbandingan Suhu Dari Termometer dan Sensor Suhu .....	40
4.2 Grafik Perbandingan Kelembaban Dari <i>Hygrometer</i> dan Sensor Kelembaban.....	43
4.3 Grafik Respon Keluaran Kontrol P pada <i>Dimmer</i> rampu .....	45
4.4 Grafik Respon Keluaran Kontrol PI pada <i>Dimmer</i> rampu .....	46
4.5 Grafik Respon Keluaran Kontrol PD pada <i>Dimmer</i> rampu .....	47
4.6 Grafik Respon Keluaran Kontrol PID pada <i>Dimmer</i> rampu .....	48
4.7 Grafik Respon Keluaran Kontrol P pada <i>Driver</i> Kipas .....	49
4.8 Grafik Respon Keluaran Kontrol PI pada <i>Driver</i> Kipas .....	50
4.9 Grafik Respon Keluaran Kontrol PD pada <i>Driver</i> Kipas .....	51

4.10	Grafik Respon Keluaran Kontrol PID pada <i>Drive</i> Kipas.....	52
4.11	Grafik Respon Keluaran Kontrol PID .....	53
4.12	Grafik Respon Waktu Suhu Saat <i>set point</i> 32°C .....	55
4.13	Grafik Respon Waktu Suhu Saat <i>set point</i> 32°C dengan Beban .....	56
4.14	Grafik Respon Waktu Suhu Saat <i>set point</i> 33°C .....	57
4.15	Grafik Respon Waktu Suhu Saat <i>set point</i> 33°C dengan Beban .....	57
4.16	Grafik Respon Waktu Suhu Saat <i>set point</i> 34°C .....	59
4.17	Grafik Respon Waktu Suhu Saat <i>set point</i> 34°C dengan Beban .....	59
4.18	Grafik Respon Waktu Suhu Saat <i>set point</i> 35°C .....	61
4.19	Grafik Respon Waktu Suhu Saat <i>set point</i> 35°C dengan Beban .....	61
4.20	Grafik perbandingan respon waktu Expert PID Control dengan Kontrol PID suhu saat <i>set point</i> 35°C .....	63

**DAFTAR TABEL**

	Halaman
2.1 Aturan Suhu Inkubator Sesuai Umur dan Berat Bayi .....	17
3.1 Rencana dan Jadwal Pelaksanaan Penelitian .....	22
3.2 Pengujian Aktuator Suhu .....	35
4.1 Pengujian Sensor Suhu DHT22 Sebelum Kalibrasi .....	39
4.2 Pengujian Sensor Suhu DHT22 Setelah Kalibrasi .....	41
4.3 Hasil Pengukuran Sensor Kelembaban DHT22 Sebelum Dikalibrasi...42	42
4.4 Hasil Pengukuran Sensor Kelembaban DHT22 Setelah Dikalibrasi .....	44
4.5 Hasil Pengujian Aktuator Suhu .....	45
4.6 Tanggapan Suhu Dengan Kontrol P pada <i>Dimmer</i> Lampu .....	45
4.7 Tanggapan Suhu Dengan Kontrol PI pada <i>Dimmer</i> Lampu .....	46
4.8 Tanggapan Suhu Dengan Kontrol PD pada <i>Dimmer</i> Lampu .....	47
4.9 Tanggapan Suhu Dengan Kontrol PID pada <i>Dimmer</i> Lampu .....	49
4.10 Tanggapan Suhu dengan Kontrol P pada <i>Driver</i> Kipas .....	50
4.11 Tanggapan Suhu dengan Kontrol PI pada <i>Driver</i> Kipas .....	51
4.12 Tanggapan Suhu dengan Kontrol PD pada <i>Driver</i> Kipas .....	52
4.13 Tanggapan Suhu dengan Kontrol PID pada <i>Driver</i> Kipas .....	53
4.14 Tanggapan Suhu dengan Kontrol PID .....	54
4.15 Respon Keadaan <i>Duty Cycle</i> dan PWM .....	54
4.16 Respon Waktu Suhu Saat <i>set point</i> 32°C.....	57
4.17 Respon Waktu Suhu Saat <i>set point</i> 33°C.....	58
4.18 Respon Waktu Suhu Saat <i>set point</i> 34°C.....	60
4.19 Respon Waktu Suhu Saat <i>set point</i> 35°C.....	62
4.20 Perbandingan Respon Waktu Suhu pada saat <i>set point</i> 35°C .....	63
4.21 Hasil Monitoring Kelembaban saat <i>set point</i> 35°C .....	64

## BAB 1. PENDAHULUAN

### 1.1 Latar Belakang

Bayi yang baru terlahir prematur memerlukan perhatian khusus setelah proses persalinan, hal ini berpengaruh pada kesehatan bayi itu sendiri. Kemajuan teknologi di bidang kesehatan sangat penting, terutama pada keselamatan bayi setelah proses persalinan. Hal ini menjadi penting dikarenakan terdapat banyak kasus terjadi kematian bayi yang terlahir dengan keadaan prematur yang disebabkan oleh tidak tertangani dengan baik fasilitas dan sarana kesehatan. Bayi yang terlahir dalam keadaan prematur mempunyai tingkat sensitivitas yang tinggi terhadap lingkungan disekitarnya dan juga rentan terhadap penyakit yang sebagian besar disebabkan oleh bakteri yang berkembang disekitar bayi dan hal tersebut terjadi karena suhu sekitar bayi tidak normal.

Angka kejadian bayi lahir prematur dengan berat badan lahir rendah (BBLR) di Indonesia masih tinggi. Prevalensi bayi prematur dan BBLR di Indonesia sekitar 10,2% dari jumlah kelahiran, bahkan beberapa daerah mencapai 17%. Angka ini lebih besar bila dibandingkan pada negara berkembang sekitar 5-9%. Bayi prematur adalah bayi yang lahir pada usia gestasi 20-37 minggu. Pada umumnya bayi prematur berat badan lahirnya rendah kurang dari 2500 gram (Etika Emaliyawati, 2017). Terkait hal tersebut, kebutuhan alat inkubator merupakan hal penting terutama di ruang perawatan bayi. Alat inkubator merupakan alat yang digunakan untuk membantu menormalkan suhu dan kelembaban di sekitar tubuh bayi. Bayi yang mengalami lahir prematur membutuhkan perawatan intensif dan tingkat kehangatan yang harus selalu stabil dikarenakan bayi prematur belum terbiasa beradaptasi dengan suhu diluar kandungan sang ibu. Faktor – faktor yang perlu diperhatikan pada inkubator adalah suhu inkubator, kelembaban inkubator, serta suhu bayi. Pengaturan suhu yang cukup hangat pada inkubator bayi sangat diperlukan untuk mencegah hipotermi yang akan terjadi pada bayi prematur, sementara perlu dilakukan pengontrolan kelembaban agar sistem pernafasan bayi dalam kondisi optimal. Pengaturan suhu dan kelembaban pada inkubator bayi dipengaruhi oleh berat

badan bayi dan umur bayi sehingga dapat ditetapkan pemilihan suhu pada inkubator bayi. Suhu inkubator bayi adalah 32°C-35°C tapi tetap dilihat dari berat badan bayi, semakin semakin berat semakin rendah suhu dari inkubator bayi.

Inkubator bayi merupakan salah satu teknologi yang sangat dibutuhkan pada dunia kedokteran, khususnya pada masalah bayi yang lahir prematur. Alat ini sangat membantu dan meringankan perawat maupun bidan dalam melakukan penyelamatan. Tetapi harga untuk mendapatkan inkubator bayi cukup mahal bahkan sampai puluhan juta rupiah, serta pemeliharaan inkubator mahal dan masalah lainnya adalah inkubator membutuhkan listrik yang memadai (Eny Meiliya. 2008:203). Bagi sejumlah rumah sakit besar untuk mendapatkan beberapa inkubator bayi ini tidak sulit, tetapi bagi bidan atau puskesmas di daerah pelosok hal ini sangat memberatkan. Untuk itu perlu dirancang dan dibuat sistem inkubator bayi dengan biaya yang lebih murah (Noor Yulita Dwi Setyaningsih, 2010).

Dengan metode kontrol PID maka untuk menentukan masing-masing parameter dilakukan proses identifikasi *plant* untuk mendapatkan model matematis dengan identifikasi Metode Ziegler-Nichols. Sistem kontrol PID yang diaplikasikan sebagai modul kontrol temperatur di dalam sistem yang menggunakan Matlab dan Arduino sebagai piranti. Hasil penelitian menunjukkan nilai konstanta PID  $K_p = 90$ ,  $K_i = 50$ ,  $K_d = 0$ , Dengan nilai konstanta PID tersebut mampu menghasilkan respon sistem yang stabil pada inkubator (Andris Kurniawan, 2018).

Berdasarkan latar belakang di atas maka perlu dilakukan penelitian dan perancangan alat dengan judul “Sistem Pengendali Temperatur dan Kelembaban Berbasis *PID Expert* untuk Inkubator Bayi Prematur” yang bertujuan untuk membuat suatu teknologi inkubator yang menghasilkan kontrol dan sistem monitoring yang baik dengan harga murah. Dengan metode kontrol PID expert maka untuk menentukan masing-masing parameter PID dengan melakukan proses identifikasi karakteristik plant dan berdasarkan pengalaman para ahli. Sistem kontrol PID Expert yang diaplikasikan dengan sensor DHT22 sebagai modul

kontrol temperatur dan monitoring kelembaban di dalam sistem yang menggunakan Arduino sebagai piranti data akuisisi untuk membaca sensor temperatur dan kelembaban serta mengendalikan *AC Light Dimmer Module* sebagai pengendali lampu agar bisa meratakan temperatur secara keseluruhan dalam sistem dan modul *driver*L298N sebagai pengendali untuk kipas. Data sample pengontrolan temperatur dan memonitoring temperatur kelembaban dapat dilihat pada LCD.

### 1.2 Rumusan Masalah

Berdasarkan latar belakang yang sudah dipaparkan di atas, didapatkan permasalahan dari pembuatan alat ini yaitu :

- a. Bagaimana membuat desain sistem pengendalian temperatur dan kelembaban pada inkubator dengan menggunakan metode *PID Expert*
- b. Bagaimana respon *transient* *PID* Kontrol saat sistem dijalankan ?
- c. Bagaimana respon *transient* *Expert PID Control* saat sistem dijalankan ?

### 1.3 Batasan Masalah

Pembuatan alat ini memiliki beberapa batasan masalah yaitu :

- a. Menggunakan sensor DHT22 sebanyak 4 buah yang dipasang 2 buah sensor diatas dan 2 buah sensor dibawah.
- b. Menggunakan 4 buah *fan* kipas sebagai bantuan untuk mempercepat penurunan suhu yang serta membantu menurunkan serta menaikkan kelembaban.
- c. Menggunakan metode Kontroler *PID Expert* sebagai algoritma.
- d. Penelitian ini menggunakan Arduino Uno sebagai mikrokontrolernya.
- e. Penelitian ini berfokus hanya pada pengendalian suhu dan memonitoring kelembaban pada inkubator bayi saat metode *PID Expert Control* diterapkan.
- f. Menggunakan modul *AC Ligth Dimmer* sebagai pengatur nyala lampu. Serta menggunakan *Modul Driver*L298N sebagai pengatur nyala kipas.

#### 1.4 Tujuan

- a. Membuat alat inkubator bayi yang dapat berjalan otomatis dan optimal dengan biaya terjangkau menggunakan metode *PID Expert* dan berbasis Arduino Uno.
- b. Dapat mengaplikasikan sensor suhu dan kelembaban DHT22 dengan metode kontroler *PID Expert* pada alat inkubator bayi.

#### 1.5 Manfaat

Manfaat yang diperoleh dari penelitian ini adalah:

- a. Diharapkan dapat bermanfaat untuk membantu pengontrolan suhu dan memonitoring kelembaban pada inkubator bayi yang dapat dilakukan secara otomatis dan optimal.
- b. Mengetahui penerapan teori elektronika yang diterapkan pada proses pengendalian suhu dan monitoring kelembaban pada inkubator bayi.

## BAB 2. TINJAUAN PUSTAKA

### 2.1 Sensor Suhu dan Kelembaban DHT22

DHT - 22 (juga disebut sebagai AM2302 ) adalah kelembaban dan suhu relatif sensor digital *output* Menggunakan sensor kelembaban kapasitif dan thermistor untuk mengukur udara di sekitarnya , dan keluar sinyal digital pada pin data. Dalam proyek ini menggunakan sensor ini dengan Arduino uno . Suhu ruangan dan kelembaban akan dicetak ke monitor serial. DHT22 adalah sensor digital yang dapat mengukur suhu dan kelembaban udara di sekitarnya. Sensor ini sangat mudah digunakan bersama dengan Arduino. Memiliki tingkat stabilitas yang sangat baik serta fitur kalibrasi yang sangat akurat. Koefisien kalibrasi disimpan dalam OTP program memory, sehingga ketika internal sensor mendeteksi sesuatu, maka module ini menyertakan koefisien tersebut dalam kalkulasinya. DHT22 termasuk sensor yang memiliki kualitas terbaik, dinilai dari respon, pembacaan data yang cepat, dan kemampuan anti-interference. Ukurannya yang kecil, dan dengan transmisi sinyal hingga 20 meter, membuat produk ini cocok digunakan untuk banyak aplikasi-aplikasi pengukuran suhu dan kelembaban.



Gambar 2.1 Sensor DHT22

(Sumber :

<http://repository.usu.ac.id/bitstream/handle/123456789/65305/Chapter%20II.pdf?sequence=3&isAllowed=y>

Karakteristik sensor DHT22 :

- Biaya rendah
- Untuk daya 5V dan I / O
- 2.5mA penggunaan saat max selama konversi (sementara meminta data)
- Baik untuk 0-100 % kelembaban pembacaan dengan akurasi 2-5 %
- Baik untuk -40 sampai 80 ° C pembacaan suhu  $\pm 0,5$  ° C akurasi Tidak lebih dari 0,5 Hz sampling rate ( sekali setiap 2 detik )
- Tubuh ukuran 27mm x 59mm x 13.5mm ( 1,05 " x 2.32 " x 0.53" )  
4 pin , 0,1 " jarak
- Berat ( hanya DHT22 ) : 2.4g

## 2.2 AC Light Dimmer Module

*AC dimmer* adalah circuit yang dapat mengontrol jumlah tegangan AC yang akan diberikan ke perangkat apapun. *AC light dimmer module* adalah modul *AC dimmer* yang sinyal PWM-nya dapat dikontrol langsung dengan mikrokontroler.



Gambar 2.2 AC Light Dimmer Module

(Sumber : <http://eprints.umm.ac.id/40871/3/BAB%20II.pdf>)

Dengan adanya fitur pin zero crossing detector di modul seperti pada gambar 2.2, membuat mikrokontroler dapat mengetahui timing yang tepat untuk mengirim sinyal PWM. Tanpa timing yang tepat, arus AC dengan TRIAC jika gatenya dicontrol PWM malah akan kacau sinyal outputnya, menyebabkan dimmer tidak berfungsi dalam menghasilkan sinyal PSM dengan kurva yang benar.

### Spesifikasi *AC Light Dimmer Module*

- Tipe TRIAC : Tri-Ad BTA16
- Arus maksimal 2 – 5A
- Tegangan AC 110/220 V
- Frekuensi AC 50/60 Hz
- Zero-Cross *detection (with zero/cross output)*
- *PWM controllable via pin PWM to give PSM output result*
- Pin Input : TTL level 3,3V - 5V

### 2.3 *Expert PID Control*

*Expert PID Control* adalah algoritma kontrol pakar langsung, yang menggabungkan pengalaman ahli dengan algoritma PID konvensional. *Expert PID Control* terdiri dari basis pengetahuan tentang fakta domain dan pemecahan masalah yang terkait dengan masalah tersebut, prosedur inferensi dengan memanfaatkan basis pengetahuan dalam penyelesaian masalah, dan memori yang berfungsi sebagai basis data global, untuk melacak status masalah, input data untuk masalah tertentu dan riwayat yang relevan dari apa yang sejauh ini telah dilakukan (R.Devanathan, Chan Chee Keong, Tong Lee Kin and Yip Chee Soon, 1988).

Selama menggunakan kontrol PID, masalah krusial adalah penyetelan parameter PID. Sistem yang sebenarnya selalu tidak linier dan waktu bervariasi ini dapat menyebabkan kontrol kinerja yang buruk. Untuk mengatasi masalah ini, untuk waktu yang lama, ilmuwan telah mencari teknologi kontrol PID *self-tuning* untuk beradaptasi dengan kondisi operasi yang kompleks dan persyaratan pelacakan kinerja tinggi. Kontrol PID ahli (*ExpertPID*) adalah kombinasi pengalaman ahli dengan kontrol PID konvensional, dan parameternya dapat dicari secara online. F. Kang dll menerapkan algoritma *Expert PID* ke sistem servo, dan hasilnya menunjukkan bahwa algoritma *Expert PID* dapat memperoleh kualitas kontrol yang sangat baik. Hasil yang diperoleh menunjukkan algoritma yang diusulkan stabil dan dapat diandalkan. Dibandingkan dengan algoritma PID ahli konvensional, respons sistem cepat, tidak memiliki osilasi dan overshoot, dan kinerja dinamis dan mapan sangat baik (Jinqiang Xu, 2017).

Hasil penelitian menunjukkan bahwa *Expert PID Control* dapat menyesuaikan parameter PID secara *real time* sesuai dengan pengalaman ahli, sehingga dapat mewujudkan kontrol efektif dari objek yang dikendalikan. Aturan kontrol sepotong – sepotong untuk mengontrol objek yang dikontrol, ambang batas dapat disesuaikan sesuai dengan pengalaman praktis, sistem memiliki presisi kontrol yang lebih tinggi dan secara efektif menyadari kelancaran sistem, dan dapat membuat sistem stabil dan operasi cepat (Qi Xing Liu, 2016).

a. Diskrit PID

Algoritma kontrol PID konvensional adalah sebagai berikut :

$$u(k) = k_p e(k) + k_i \sum_{i=1}^k e(i) T_s + k_d \frac{e(k) - e(k-1)}{T_s} \quad (2.1)$$

Dimana  $u(k)$  adalah *output* dari pengontrol,  $K_p$ ,  $K_i$ ,  $K_d$  adalah parameter proporsional, integral dan derivatif masing-masing,  $e(k)$  adalah penyimpangan kontrol waktu pengambilan sampel saat ini.  $e(k-1)$  adalah penyimpangan kontrol waktu pengambilan sampel terakhir,  $T_s$  adalah periode pengambilan sampel.

b. *Expert PID Control*

$$\Delta e(k) = e(k) - e(k-1) \quad (2.2)$$

$$\Delta e(k-1) = e(k-1) - e(k-2) \quad (2.3)$$

Berdasarkan pengalaman ahli, Persamaan 2.1 dapat dianalisa sebagai berikut :

1. Ketika  $|u(k)| \geq A_1$ , yang berarti nilai *error* kontrol yang terbesar, sehingga *output* dari pengontrol harus konstan, yaitu :

$$u(k) = a \quad (2.4)$$

2. Ketika  $A_2 < |e(k)| < A_1$ ,  $e(k) * \Delta e(k) > 0$  atau  $\Delta e(k) = 0$ , yang berarti nilai *error* kontrol lebih besar, dan nilai *error* berubah ke arah peningkatan nilai absolut, atau nilai *error* tetap konstan, pengontrol harus mampu menggunakan PID yang kuat, *output* dari pengontrol yaitu :

$$u(k) = \alpha \left[ K_p e(k) + K_d \frac{e(k) - e(k-1)}{T_s} \right], \alpha > 1 \quad (2.5)$$

3. Ketika  $A_3 < |e(k)| < A_2$  dan  $e(k) * \Delta e(k) > 0$  atau  $\Delta e(k) = 0$ , yang berarti nilai *error* kontrol lebih kecil, dan nilai *error* berubah ke arah peningkatan nilai absolut, atau nilai *error* tetap konstan, pengontrol dapat menggunakan PID yang lemah, sehingga *output* dari *controller* adalah :

$$u(k) = K_p e(k) + K_d \frac{e(k) - e(k-1)}{T_s} \quad (2.6)$$

4. Ketika  $e(k) * \Delta e(k) < 0$  dan  $\Delta e(k) * \Delta e(k-1) > 0$  atau  $e(k) = 0$ , berarti penyimpangan kontrol berubah ke arah penurunan nilai absolut, atau sistem sudah mencapai keseimbangan, sehingga *output* dari pengontrol harus tetap tidak berubah, yaitu :

$$u(k) = u(k-1) \quad (2.7)$$

5. Ketika  $e(k) * \Delta e(k) < 0$  dan  $\Delta e(k) * \Delta e(k-1) < 0$ , berarti penyimpangan kontrol adalah ekstrim, jika penyimpangan itu lebih besar, seperti  $|e(k)| \geq A_2$ , *controller* harus memiliki kontrol yang kuat, sehingga *output controller* harus lebih besar,

$$u(k) = \alpha k_p e_m(k), \alpha > 1 \quad (2.8)$$

$e_m(k)$  adalah nilai penyimpangan ekstrim k.

6. Ketika  $e(k) * \Delta e(k) < 0$  dan  $\Delta e(k) * \Delta e(k-1) < 0$ , berarti penyimpangan kontrol ekstrim, jika penyimpangan kecil, seperti  $|e(k)| < A_2$ , *controller* harus memiliki kontrol yang lemah, sehingga *output controller* harus lebih kecil,

$$u(k) = \beta k_p e_m(k), \beta < 1 \quad (2.9)$$

## 2.4 Arduino Uno

Arduino Uno R3 adalah board sistem minimum berbasis mikrokontroler ATmega328P jenis AVR. Arduino Uno R3 memiliki 14 digital input/output (6 diantaranya dapat digunakan untuk PWM output), 6 analog input, 16 MHz osilator kristal, USB connection, power jack, ICSP header dan tombol reset. Arduino menyediakan 20 pin I/O, yang terdiri dari 6 pin input analog dan 14 pin digital input/output. Untuk 6 pin analog sendiri bisa juga difungsikan sebagai output digital jika diperlukan output digital tambahan selain 14 pin yang sudah tersedia. Untuk

mengubah pin analog menjadi digital cukup mengubah konfigurasi pin pada program. Dalam *board* kita bisa lihat pin digital diberi keterangan 0-13, jadi untuk menggunakan pin analog menjadi output digital, pin analog yang pada keterangan board 0-5 kita ubah menjadi pin 14-19. dengan kata lain pin analog 0-5 berfungsi juga sebagai pin output digital 14-16.



Gambar 2.3 Arduino Uno

(Sumber : <https://ilearning.me/sample-page-162/arduino/pengertian-arduino-uno>)

Spesifikasi dari arduino uno seperti gambar 2.3 adalah sebagai berikut :

- Microcontroller ATmega328
- Operasi dengan daya 5V Voltage
- Input Tegangan (disarankan) 7-12V
- Input Tegangan (batas) 6-20V
- Digital I / O Pins 14 (dimana 6 memberikan output PWM)
- Analog Input Pin 6
- DC Lancar per I / O Pin 40 mA
- Saat 3.3V Pin 50 mA DC
- Flash Memory 32 KB (ATmega328) yang 0,5 KB digunakan oleh bootloader
- SRAM 2 KB (ATmega328)

- EEPROM 1 KB (ATmega328)
- Clock Speed 16 MHz



Gambar 2.4 Konfigurasi pin Atmega 328P

(Sumber : Riky Ovile, 2014)

Arduino Uno R3 menggunakan mikrokontroler yang dikontrol secara penuh oleh mikroprosesor ATmega328P. Mikroprosesor yang digunakan ini sudah dilengkapi dengan konverter sinyal analog ke digital (ADC) sehingga tidak diperlukan penambahan ADC eksternal. Pada Gambar 2.4 diatas merupakan penjelasan melalui gambar mengenai konfigurasi pin-pin yang merupakan bagian dari mikrokontoller ATmega328 yang digunakan didalam modul board arduino.

Pin Masukan dan Keluaran Arduino Uno Masing-masing dari 14 pin digital arduino uno dapat digunakan sebagai masukan atau keluaran menggunakan fungsi `pinMode()`, `digitalWrite()` dan `digitalRead()`. Setiap pin beroperasi pada tegangan 5 volt. Setiap pin mampu menerima atau menghasilkan arus maksimum sebesar 40 mA dan memiliki 10 resistor *pull-up internal* (putus secara default) sebesar 20-30 KOhm. Sebagai tambahan, beberapa pin masukan digital memiliki kegunaan khusus yaitu:

- Komunikasi serial: pin 0 (RX) dan pin 1 (TX), digunakan untuk menerima (RX) dan mengirim (TX) data secara serial.

- *External Interrupt*: pin 2 dan pin 3, pin ini dapat dikonfigurasi untuk memicu sebuah interrupt pada nilai rendah, sisi naik atau turun, atau pada saat terjadi perubahan nilai.
- *Pulse-width modulation (PWM)*: pin 3, 5, 6, 9, 10 dan 11, menyediakan keluaran PWM 8-bit dengan menggunakan fungsi `analogWrite()`.
- *Serial Peripheral Interface (SPI)*: pin 10 (SS), 11 (MOSI), 12 (MISO) dan 13 (SCK), pin ini mendukung komunikasi SPI dengan menggunakan SPI library.
- *LED*: pin 13, terdapat built-in LED yang terhubung ke pin digital 13. Ketika pin bernilai High maka LED menyala, sebaliknya ketika pin bernilai Low maka LED akan padam.

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

- *TWI*: A4 atau SDA pin dan A5 atau SCL pin. Mendukung komunikasi TWI.
- *Aref*. Referensi tegangan untuk input analog. Digunakan dengan `analogReference()`
- *Reset*

## 2.5 LCD (Liquid Crystal Display)

LCD adalah media tampilan yang menggunakan kristal cair sebagai penampil utama. LCD terdiri dari beberapa lapisan cairan kristal diantara 2 plat kaca. Ada beberapa jenis LCD diantaranya : Segment LCD, Dot Matrix character LCD, dan Graphic LCD



Gambar 2.5 *Liquid Crystal Display*

(Sumber : <http://eprints.umm.ac.id/40871/3/BAB%20II.pdf>)

Display elektronik seperti pada gambar 2.5 adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. LCD (*Liquid Cristal Display*) adalah salah satu jenis display elektronik yang dibuat dengan teknologi CMOS logic yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap front-lit atau mentransmisikan cahaya dari back-lit. LCD (*Liquid Cristal Display*) berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik.

Dalam modul LCD (*Liquid Cristal Display*) terdapat microcontroller yang berfungsi sebagai pengendali tampilan karakter LCD (*Liquid Cristal Display*). Microcontroller pada suatu LCD (*Liquid Cristal Display*) dilengkapi dengan memori dan register. Memori yang digunakan microcontroller internal LCD adalah :

- DDRAM (*Display Data Random Access Memory*) merupakan memori tempat karakter yang akan ditampilkan berada.
- CGRAM (*Character Generator Random Access Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai dengan keinginan.
- CGROM (*Character Generator Read Only Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut merupakan karakter dasar yang sudah ditentukan secara permanen oleh

pabrik pembuat LCD (Liquid Cristal Display) tersebut sehingga pengguna tinggal mengambilnya sesuai alamat memorinya dan tidak dapat merubah karakter dasar yang ada dalam CGROM.

Register control yang terdapat dalam suatu LCD diantaranya adalah :

- Register perintah yaitu register yang berisi perintah-perintah dari mikrokontroler ke panel LCD (*Liquid Cristal Display*) dan saat proses penulisan data atau tempat status dari panel LCD (*Liquid Cristal Display*) dapat dibaca pada saat pembacaan data.
- Register data yaitu register untuk menuliskan atau membaca data dari atau ke DDRAM. Penulisan data pada register akan menempatkan data tersebut ke DDRAM sesuai dengan alamat yang telah diatur sebelumnya.

Pin, kaki atau jalur input dan kontrol dalam suatu LCD (*Liquid Cristal Display*) diantaranya adalah :

- Pin data adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan LCD (*Liquid Cristal Display*) dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit.
- Pin RS (*Register Select*) berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah.
- *Logika low* menunjukkan yang masuk adalah perintah, sedangkan logika high menunjukkan data.
- Pin R/W (*Read Write*) berfungsi sebagai instruksi pada modul jika *low* tulis data, sedangkan *high* baca data.
- Pin E (*Enabled*) digunakan untuk memegang data baik masuk atau keluar.
- Pin VLCD berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 Kohm, jika tidak digunakan dihubungkan ke *ground*, sedangkan tegangan catu daya ke LCD sebesar 5 Volt.

## 2.6 Fan

Pengertian *Fan* adalah semua kipas yang terdapat di dalam inkubator. Saat lampu menyala di dalam inkubator dan digunakan untuk beberapa waktu lamanya, maka sensor akan mendeteksi suhu tersebut, dan apabila panas pada inkubator melebihi yang ditentukan. Untuk mendinginkan suhu inkubator tersebut yang berlebih maka digunakan kipas dalamnya.



Gambar 2.6 *Fan*

(Sumber : Jaenal Arifin, 2017)

Fungsi utama dari *fan* pada gambar 2.6 adalah menjaga agar inkubator tetap dalam suhu yang masih dapat ditolerir oleh bayi. Sehingga kelembaban juga akan terjaga serta bayi tetap terjaga keselamatan serta kesehatannya.

## 2.7 Inkubator Bayi

Inkubator Bayi merupakan suatu alat biomedis yang berfungsi memberikan kehangatan dan kelembaban dimana seluruh lingkungannya terkontrol dan diperlukan bagi bayi yang terlahir prematur. Umumnya inkubator bayi dirancang secara otomatis sehingga suhu dan kelembaban dalam ruangan tetap stabil. Inkubator

biasanya berbentuk troli dengan matras kecil untuk meletakkan bayi serta penutup di atasnya. Penutup ini berfungsi melindungi bayi dari debu, asap, serta suara bising dari luar. Sensor suhu pada inkubator biasanya ditempatkan dibagian atas atau langsung pada tubuh bayi. Sementara sistem kelembaban serta pemanas terdapat pada bagian bawah.



Gambar 2.7 Inkubator bayi

(Sumber : Penulis)

Inkubator Bayi seperti gambar 2.7 juga membutuhkan kelembaban yang stabil sehingga kondisi di dalamnya tetap terjaga sesuai dengan yang diinginkan. Menurut data statistik tentang pengukuran dan kalibrasi yang dilakukan oleh BPFK Surabaya pada tahun 2006 hingga 2007, terjadi kecenderungan masalah pada kelembaban dan *overheat* pada matras. Supaya masalah tersebut teratasi maka

perlu dirancang sebuah alat inkubator yang terdapat pengendalian temperatur dan kelembaban secara otomatis dan optimal.

Tabel 2.1 Aturan Suhu Inkubator Sesuai Umur dan Berat Bayi

Berat Badan	Umur	Suhu (°C)
<b>&lt; 1500 gram</b>	1 – 10 Hari	35
	11 Hari – 3 Minggu	34
	3 – 5 Minggu	33
	> 5 Minggu	32
<b>1500 - 2000 gram</b>	1 – 10 Hari	34
	11 Hari – 4 Minggu	33
	> 4 Minggu	32
<b>2100 - 2500 gram</b>	1 – 2 Hari	34
	3 Hari – 3 Minggu	33
	> 3 Minggu	32
<b>&gt; 2500 gram</b>	1 - 2 Hari	33
	> 2 Hari	32

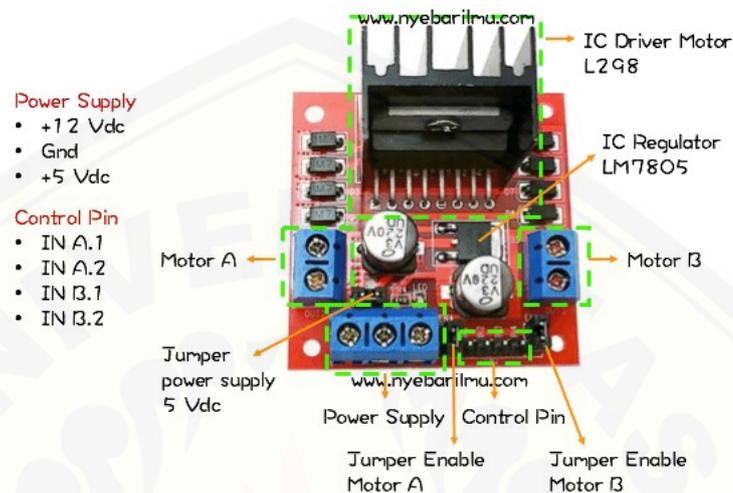
(Sumber : Pratiwi, dkk. 2014)

Tabel 2.1 merupakan tabel aturan suhu pada inkubator yang sesuai dengan standar kesehatan dan keselamatan bayi yang telah disesuaikan dengan umur serta berat badan bayi yang terlahir prematur dan perlu penanganan khusus untuk menjaga kehangatan tubuh bayi.

## 2.8 Module Driver Motor L298

Driver motor L298N merupakan modul *driver* motor DC yang difungsikan untuk mengontrol kecepatan serta arah perputaran motor DC. Mampu mengeluarkan *output* tegangan untuk Motor dc dan motor stepper sebesar 5 – 35 volt. IC l298 terdiri dari transistor-transistor logik (TTL) dengan gerbang nand yang

memudahkan dalam menentukan arah putaran suatu motor dc dan motor stepper. Dapat mengendalikan 2 untuk motor dc namun hanya dapat mengendalikan 1 motor stepper.



Gambar 2.8 Modul Driver L298N

(Sumber :

<http://repository.umy.ac.id/bitstream/handle/123456789/6507/BAB%20II.pdf?sequence=9&isAllowed=y>

Keterangan dari gambar 2.8 tersebut yaitu sebagai berikut :

- Enable A : berfungsi untuk mengaktifkan bagian *output* motor A
- Enable B : berfungsi untuk mengaktifkan bagian *output* motor B
- Jumper 5vdc : sebagai mode pemilihan sumber tegangan 5Vdc, jika tidak dijumper maka akan ke mode sumber tegangan 12 Vdc
- Control Pin : Sebagai kendali perputaran dan kecepatan motor yang dihubungkan ke Mikrokontroler

Spesifikasi dari Modul Driver Motor L298N :

- Menggunakan IC L298N (*Double H bridge Drive Chip*)
- Tegangan minimal untuk masukan power antara 5V-35V

- Tegangan operasional : 5V
- Arus untuk masukan antara 0-36mA
- Arus maksimal untuk keluaran per *Output* A maupun B yaitu 2A
- Daya maksimal yaitu 25W
- Dimensi modul yaitu 43 x 43 x 26mm
- Berat : 26g

## 2.9 Heater

Lampu Pijar atau disebut juga *Incandescent Lamp* adalah jenis lampu listrik yang menghasilkan cahaya dengan cara memanaskan kawat filamen di dalam bola kaca yang diisi dengan gas tertentu seperti nitrogen, argon, kripton atau hidrogen. Lampu pijar tergolong dalam lampu listrik generasi awal yang masih digunakan hingga saat ini. Cahaya lampu pijar dibangkitkan dengan mengalirkan arus listrik dalam suatu filament yang pada akhirnya energi listrik diubah menjadi energi panas dan cahaya yang dapat digunakan sebagai pemanas atau *heater*. Arus listrik dalam filament merupakan gerakan elektron- elektron bebas yang karena pergerakannya mengakibatkan terjadinya benturan dengan elektron - elektron yang terikat pada inti atom.



Gambar 2.10 Lampu

(Sumber : Bima Brilliando Agam, 2015)

Energi listrik yang diperlukan lampu pijar pada gambar 2.9 untuk menghasilkan cahaya yang terang lebih besar dibandingkan dengan sumber cahaya buatan lainnya seperti lampu pendar dan dioda cahaya, maka secara bertahap pada beberapa negara peredaran lampu pijar mulai dibatasi. Disamping memanfaatkan cahaya yang dihasilkan, beberapa penggunaan lampu pijar lebih memanfaatkan panas yang dihasilkan, contohnya adalah pemanas pada inkubator, kandang ayam, dan pemanas inframerah dalam proses pemanasan di bidang industri. Komponen utama dari lampu pijar adalah bola lampu yang terbuat dari kaca, filamen yang terbuat dari wolfram, dasar lampu yang terdiri dari filamen, bola lampu, gas pengisi, dan kaki lampu (Didik Supriyono,2014).

Pada dasarnya filamen pada sebuah lampu pijar adalah sebuah resistor. Saat dialiri arus listrik, filamen tersebut menjadi sangat panas, berkisar antara 2800° Kelvin hingga maksimum 3700° Kelvin. Ini menyebabkan warna cahaya yang dipancarkan oleh lampu pijar biasanya berwarna kuning kemerahan. Pada temperatur yang sangat tinggi itulah filamen mulai menghasilkan cahaya pada panjang gelombang yang kasat mata. Hal ini sejalan dengan teori radiasi benda hitam. Dengan mengalirnya arus elektron melalui filamen tipis menghasilkan tumbukan- tumbukan elektron didalam filamen sehingga membentuk cahaya serta panas sebagai residu.

Seiring bertambahnya voltase listrik yang digunakan maka semakin tinggi intensitas cahaya, sebab naiknya tegangan memacu naiknya aliran listrik karena resisten bersifat tetap jika kenaikan suhu dihiraukan, dan membesarnya aliran listrik maka membesar pula probabilitas terjadinya tumbukan didalam filamen.(Didik Supriyono,2002).

### BAB 3. METODOLOGI PENELITIAN

#### 3.1 Tempat Penelitian

Penelitian ini dilakukan di Laboratorium Elektronika dan Terapan Fakultas Teknik Universitas Jember, Jl. Slamet Riyadi No.62, Patrang, Kabupaten Jember.

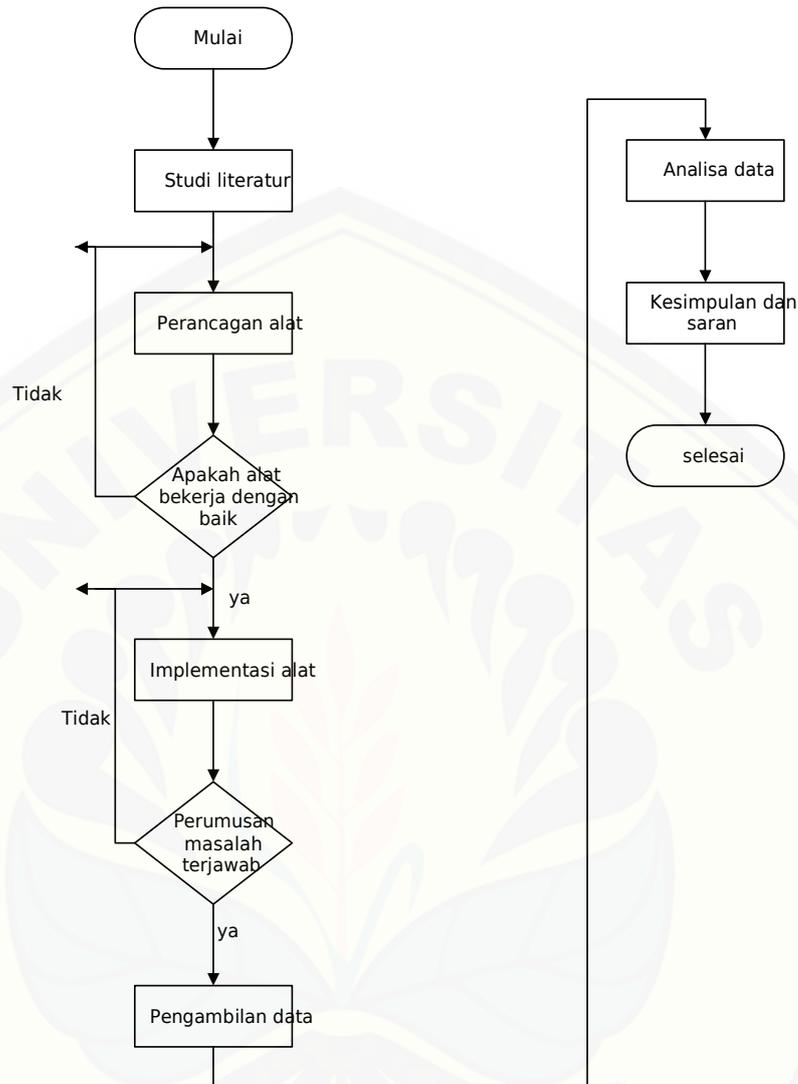
#### 3.2 Waktu Penelitian

Waktu penelitian ini dimulai pada bulan Desember 2019 – Mei 2020, dengan rincian sebagai berikut :

Tabel 3.1 Rencana dan Jadwal Pelaksanaan Penelitian

No	Kegiatan	Januari				Februari				Maret				April				Mei			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Studi Literatur	■	■	■	■																
2	Perancangan Elektrikal dan Box Inkubator					■	■	■	■												
3	Pembelian Alat dan Bahan									■	■	■	■								
4	Pembuatan Inkubator Bayi													■	■	■	■				
5	Pengujian																	■	■	■	■
6	Analisis Data																			■	■
7	Kesimpulan																				■

### 3.3 Tahapan Penelitian



Gambar 3.1 Tahapan Pelaksanaan Penelitian

Langkah-langkah yang akan dilakukan dalam tahapan penelitian seperti pada gambar 3.1 adalah sebagai berikut :

#### 1. Studi Literatur

Tahap awal pelaksanaan penelitian ini adalah dengan mencari literatur dari hasil penelitian sebelumnya melalui buku atau internet untuk mengetahui karakteristik komponen sistem, prinsip kerja serta teori yang menunjang

lainnya. Diharapkan dengan literatur yang telah didapat dapat memberikan arahan untuk mengurangi kesalahan dalam penelitian.

## 2. Perancangan Alat

Tahapan ini merupakan tahap merancang konstruksi secara sistematis dari alat yang akan dilakukan penelitian. Diharapkan dari proses perancangan konstruksi yang sistematis ini, alat yang nantinya akan diteliti dapat terbentuk. Hal – hal lain yang dilakukan yakni seperti penyoderan, perancangan sistem monitoring.

## 3. Implementasi Alat

Setelah *hardware* dan *software* terbentuk maka dilakukan pengujian pada masing – masing blok dan kemudian pengujian dilakukan pada keseluruhan sistem. Dalam implementasi alat ini juga dilakukan proses kalibrasi dimana pada proses kalibrasi ini bertujuan agar pembacaan sensor akurat sesuai dengan alat ukur.

## 4. Analisa dan Pengambilan Data

Setelah melakukan pengujian pada keseluruhan sistem dan memastikan bekerja dengan baik dan hasilnya memenuhi target, maka yang dilakukan selanjutnya adalah pengambilan data yang diperlukan untuk kemudian dianalisa dari data yang telah didapatkan. Analisa yang dilakukan adalah kecepatan menuju steady state dan ketahanan terhadap perubahan beban.

## 5. Penyusunan Laporan

Pada tahap akhir ini, hasil pengambilan data dan analisa dimasukkan ke pembahasan. Kemudian, dari apa yang telah dianalisa dapat ditarik beberapa kesimpulan yang menyangkut kinerja dari alat yang dibuat dan memberikan saran untuk memperbaiki kekurangan yang ada, kemungkinan pengembangan, serta penyempurnaan alat di masa mendatang.

### 3.4 Alat dan Bahan Penelitian

Peralatan – peralatan yang diperlukan dalam penelitian adalah :

1. *Personal Computer*
2. Alat – alat ukur : Multimeter, AC *Power Supply*
3. Perkakas mekanis : Palu, Solder, Tank, Obeng, Gergaji, Gerindra, Bor.
4. Perangkat lunak: Windows 10, Microsoft Office, Matlab, Arduino IDE

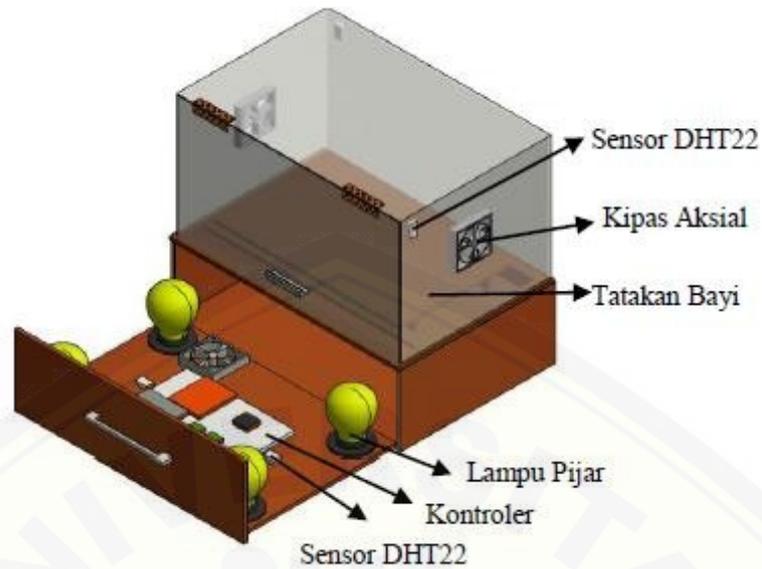
Beberapa komponen yang diperlukan dalam penelitian ini adalah :

1. Arduino Uno R3
2. Sensor suhu dan kelembaban DHT22
3. LCD 16x2
4. *Fan*
5. PCB polos
6. *Module AC Light Dimmer*
7. Modul driver L298N
8. Lampu

### 3.5 Rancangan Penelitian

#### 3.5.1 Desain Mekanik Alat

Desain mekanik dari inkubator bayi seperti pada gambar 3.2 memiliki panjang 65cm, lebar 45cm, dan 50cm serta berbentuk seperti balok, didalamnya terdapat ventilasi untuk pergantian keluar masuknya udara. Inkubator dibagi menjadi dua bagian, yaitu box atas dan box bawah. Untuk bagian inkubator box bawah digunakan sebagai tempat rangkaian elektronika seperti pada gambar 3.x desain mekanik inkubator bayi prematur. Kemudian pada bagian inkubator box atas digunakan sebagai tempat menaruh bayi dan terdapat sensor DHT22 serta LCD 16x2.



Gambar 3.2 Desain mekanik

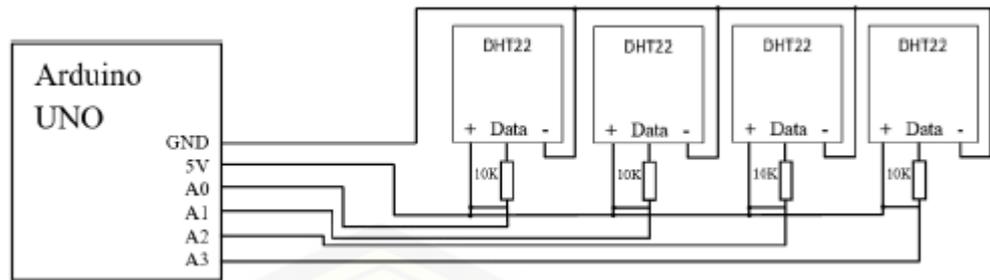
(Sumber : Penulis)

Pada inkubator menggunakan dua buah sensor DHT22 yang ditempatkan dibagian box bawah pada inkubator. Serta terdapat dua buah sensor DHT22 yang dipasang dibagian box atas dari inkubator, sensor DHT22 ini berfungsi sebagai pendeteksi besarnya nilai suhu dan kelembaban didalam inkubator. Kemudian digunakan 2 *fan* kipas dibagian atas serta dua buah dibagian box bawah sebagai pendingin apabila suhu melebihi batas yang ditentukan dengan modul driver L298N sebagai pengatur kecepatan putarannya. Serta digunakan modul AC light dimmer untuk menaikkan suhu dengan cara mengatur *duty cycle* dari lampu.

### 3.5.2 Desain Diagram Elektrikal Alat

- Rangkaian Sensor DHT22

Sensor DHT22 dihubungkan dengan mikrokontroler Arduino Uno. Komunikasi antara sensor DHT22 dengan Arduino Uno menggunakan pin analog A0, A1, A2 dan A3 sebagai input, digunakan empat buah pin karena digunakan empat buah sensor. Kemudian untuk catu daya dihubungkan dengan pin power 5V dan ground dapat dilihat pada Gambar 3.3.



Gambar 3.3 Rangkaian DHT22 dengan Arduino

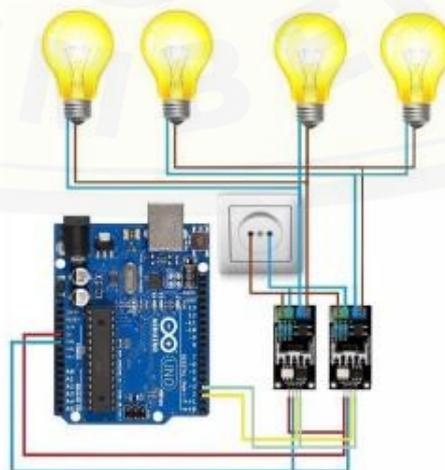
(Sumber : Penulis)

```
1 | #include <DHT.h>
2 | DHT dht(A0, DHT22); //Pin, Jenis DHT
3 | DHT dht2(A1, DHT22);
4 | DHT dht3(A2, DHT22);
5 | DHT dht4(A3, DHT22);
6 | float kelembaban1 = dht.readHumidity();
7 | float kelembaban2 = dht2.readHumidity();
8 | float kelembaban3 = dht3.readHumidity();
9 | float kelembaban4 = dht4.readHumidity();
10| float suhu1 = dht.readTemperature();
11| float suhu2 = dht2.readTemperature();
12| float suhu3 = dht3.readTemperature();
13| float suhu4 = dht4.readTemperature();
```

Pada baris pertama dalam listing program arduino merupakan *library* dari sensor DHT. Pada baris ke-2 sampai baris ke-5 merupakan inisialisasi dari variabel pin dan jenis DHT yang digunakan, dalam penelitian ini digunakan empat buah sensor DHT22 sehingga terdapat empat pin yang digunakan sebagai input. Pada baris ke-6 hingga baris ke-9 digunakan untuk menyimpan nilai kelembaban yang terbaca oleh sensor kedalam variabel kelembaban1 sampai kelembaban4 dengan tipe data desimal (*float*). Kemudian pada baris ke-10 hingga baris ke-13 digunakan untuk menyimpan nilai suhu yang terbaca oleh sensor kedalam variabel *suhu1* sampai *suhu4* dengan tipe data desimal (*float*).

- Rangkaian Heater dengan Modul AC Light Dimmer

Heater yang digunakan pada rangkaian ini adalah sebuah lampu pijar dan langsung terhubung dengan pin *load* pada modul *AC light dimmer*. Kemudian modul *AC light dimmer* dihubungkan dengan Arduino Uno dan sumber tegangan sebesar 220V. Komunikasi antara modul *AC light dimmer* dengan Arduino uno menggunakan pin 2 dan pin 9 sebagai output *Duty Cycle*. Selain dihubungkan dengan sumber tegangan sebesar 220V, modul ini juga dihubungkan dengan pin power 5V dan ground pada Arduino uno. Rangkaian heater dengan modul AC light dimmer dan Arduino uno, dimana dari setiap modul ac light dimmer yang digunakan terdapat 2 lampu yang diseri dapat dilihat pada Gambar 3.4.



Gambar 3.4 Rangkaian *Heater* dengan Modul *AC Light Dimmer* dan Arduino Uno

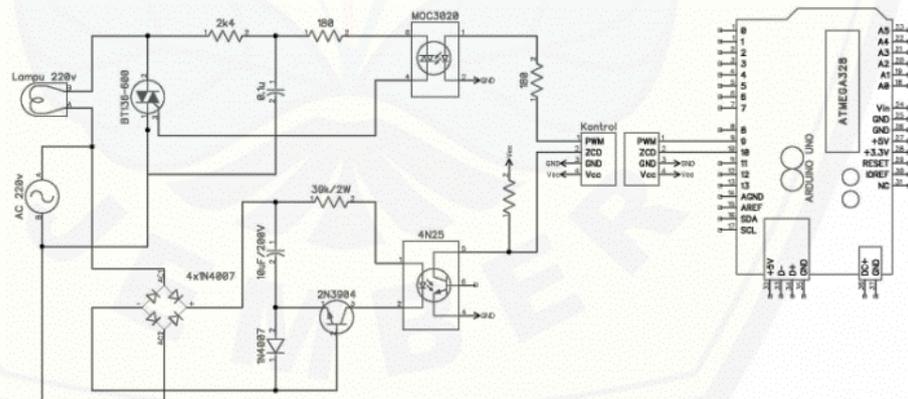
(Sumber : Penulis)

```

1| #include "RBDdimmer.h"
2| #define USE_SERIAL Serial
3| #define outputPin 3
4| #define zerocross 2

```

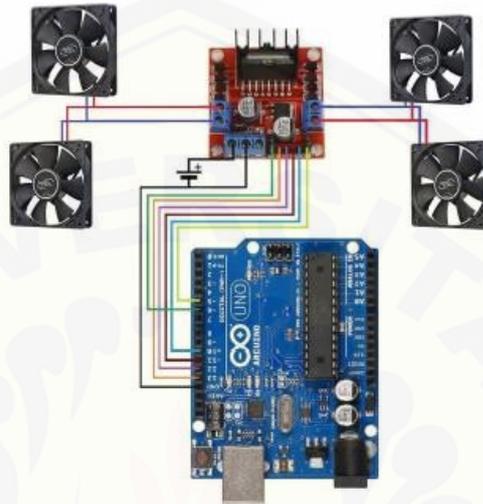
Pada baris ke-1 listing program Arduino merupakan *library* dari modul *AC light dimmer* pada baris ke-2 merupakan inisialisasi dari variabel *USE\_SERIAL* yang digunakan dari modul *AC light dimmer* pada baris ke-3 merupakan inisialisasi dari variabel *outputPin* dari modul *AC light dimmer* pada baris ke-4 merupakan inisialisasi dari variabel *zerocross* dari modul *AC light dimmer*



Gambar 3.5 Skema Rangkaian *Dimmer* dan Arduino Uno

(Sumber : <https://www.semesin.com/project/2018/05/01/dimmer-pwm-arduino/>)

- Rangkaian *Fan*Kipas Aksial dengan Modul Driver L289N

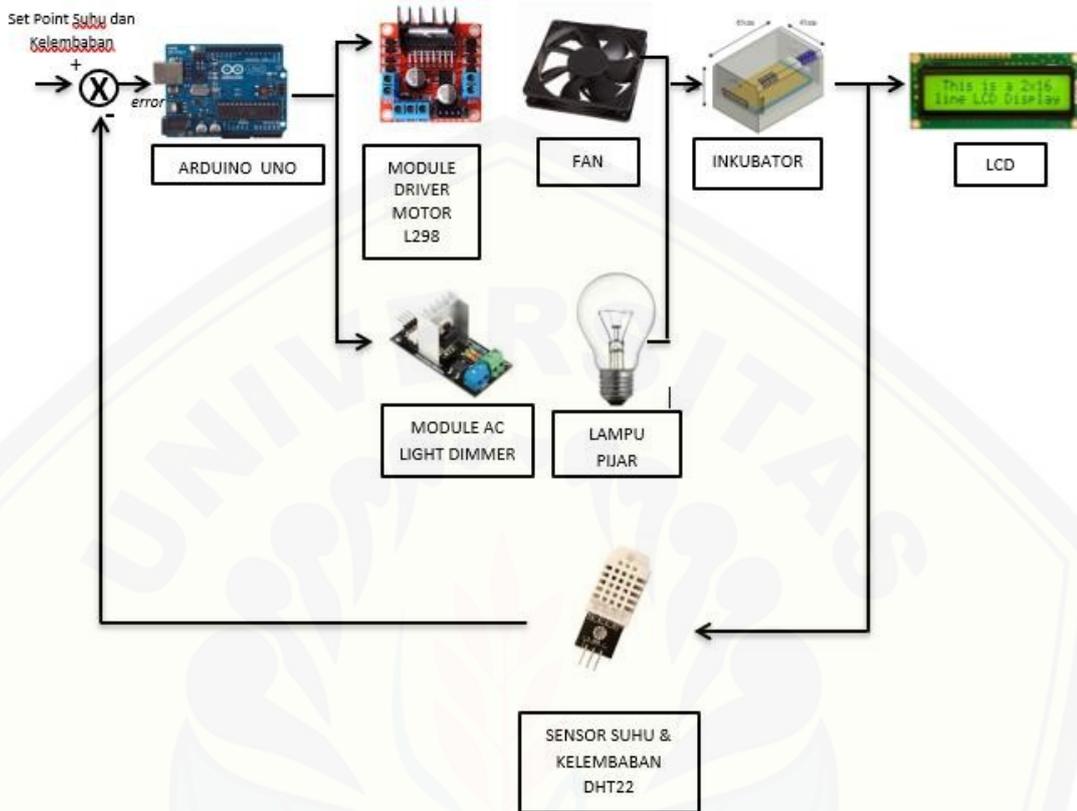


Gambar 3.6 Rangkaian Kipas Aksial dengan Modul Driver L298N dan Arduino Uno.

(Sumber : Penulis)

Pada gambar diatas menunjukkan rangkaian kipas aksial dengan modul driver l298n dan Arduino uno, digunakan 2 buah kipas aksial disetiap pin *output* yang ada pada modul. Fan/Kipas Aksial langsung terhubung dengan pin *output* A dan B pada modul driver L298N yang berfungsi sebagai pengontrol menggunakan nilai PWM dengan cara melepas jumper pada pin enable A dan B. Kemudian modul driver L298N dihubungkan dengan Arduino uno pada pin 10, 11, 12 dan 13 sebagai input serta pin 5 dan 6 sebagai *output* untuk catu daya dihubungkan dengan pin power 5V dan ground.

### 3.5.3 Diagram Blok Sistem



Gambar 3.7 Diagram Blok Sistem

(Sumber : Penulis)

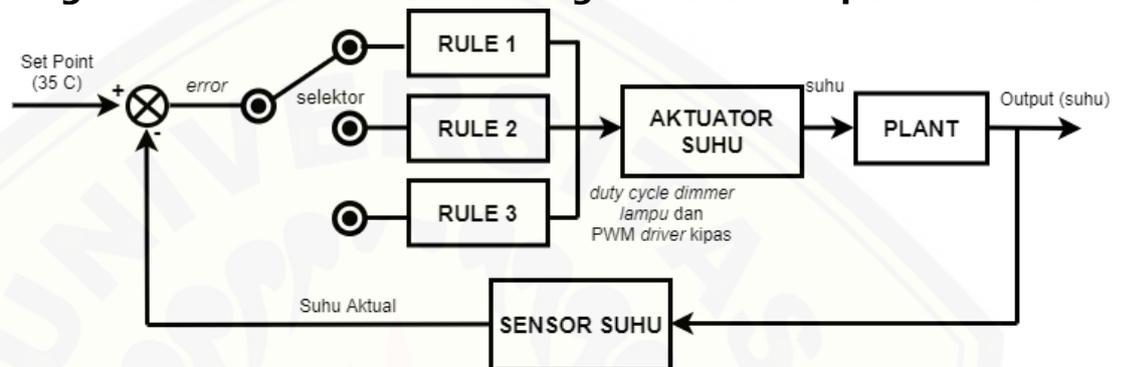
Penjelasan desain sistem pada gambar 3.6 tersebut menggunakan kontrol *close loop*. Pada sistem digunakan Arduino Uno untuk mengatur inkubator bayi. Untuk suhu terdapat sebuah dimmer lampu (pemanas) yang diletakkan di dalam box inkubator bayi. Kemudian terdapat sensor DHT22 yang digunakan untuk mendeteksi suhu pada inkubator bayi. Kemudian sensor akan mengirim data pada mikrokontroler untuk diolah dan ditampilkan pada LCD dalam bentuk derajat celsius. Setelah sensor dan mikrokontroler membaca suhu, terdapat tahap dimana kontrol membaca suhu yang didapat kemudian menentukan gerak *dimmer* atau *driver* yang dibutuhkan. Selain itu terdapat sensor kelembaban yang tergabung pada

sensor suhu. Setelah itu mikrokontroler akan membaca data kelembaban dan akan ditampilkan pada LCD.

### 3.6 Diagram Blok Expert PID Control

Berikut ini adalah diagram blok sistem kontrol *Expert PID* yang akan digunakan pada penelitian ini.

#### 3.6.1 Diagram Blok Kontrol Suhu dengan Metode Expert PID Control

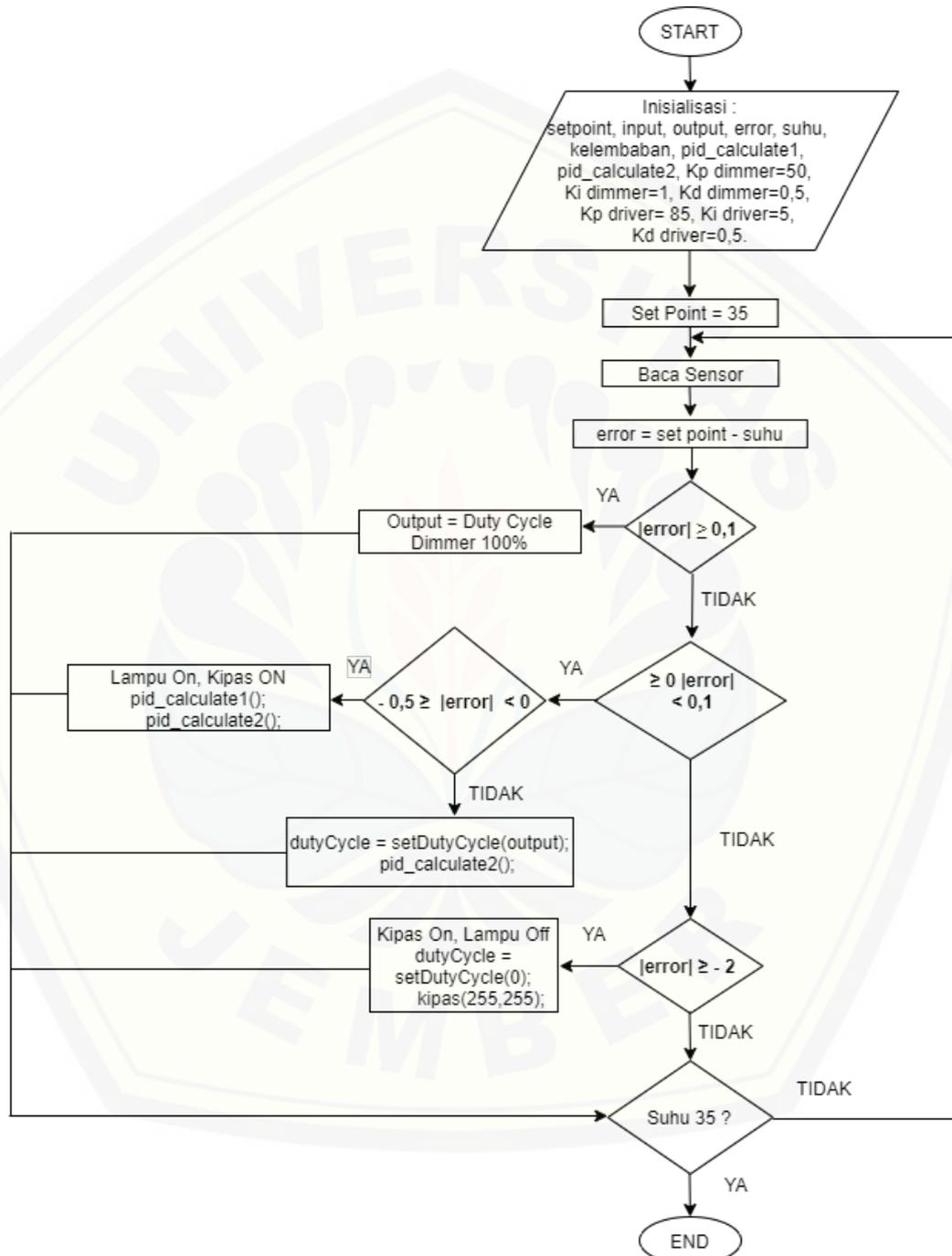


3.8 Gambar Diagram Blok *Expert PID Control* Suhu

(Sumber : Penulis)

Pada gambar 3.7 di atas dapat dilihat bahwa kontrol ini menggunakan *Expert PID Control*. Langkah awal dalam sistem ini yaitu dengan memasukkan *set point* sebesar  $35^{\circ}\text{C}$  untuk mengatur nilai suhu akan mencapai titik sesuai dengan *set point* yang telah diberikan. Sesuai dengan nilai *error* tersebut maka mikrokontroler akan bekerja sesuai dengan *rules* yang telah ditetapkan sebelumnya. Setelah rules bekerja maka aktuator suhu yang berupa *dimmer* dan *driver* akan mengatur nilai PWM (*Pulse Width Modulation*) untuk menggerakkan lampu atau kipas agar suhu dapat mencapai set point yang diinginkan. Setelah itu akan dilakukan pembacaan sensor untuk memantau hasil keluaran suhu pada inkubator. Dan sistem ini akan terus mengulang langkah yang sama atau *looping*.

## 3.6.2 Diagram Alir Expert PID Control

Gambar 3.9 Diagram Alir *Expert PID Control* untuk Suhu

(Sumber : Penulis)

Flowchart sistem diatas menunjukkan langkah – langkah sebagai berikut :

### 1. Inisialisasi

Inisialisasi merupakan pemberian tugas nilai awal untuk mendeklarasikan sebuah variabel atau objek. Variabel yang dideklarasikan pada sistem ini diantaranya:

- a. *Set point* adalah nilai referensi dari variabel yang diinginkan atau dikehendaki untuk mengendalikan sistem.
- b. *Input* adalah semua data dan perintah yang dimasukkan kedalam memori sistem. Pada sistem ini *input* yang dimasukkan berupa *set point* pembacaan sensor suhu.
- c. *Output* adalah data yang telah diproses. *Output* pada sistem ini berupa suhu yang dikendalikan oleh nilai PWM (*Pulse Width Modulation*) *dimmer* dan *driver*
- d. *Error* adalah banyaknya perbedaan dari nilai *set point* dengan nilai pembacaan sensor.
- e. *PID Calculate* adalah perhitungan PID yang tepat untuk *dimmer* sehingga alat dapat berjalan sesuai dengan respon sistem yang diinginkan.
- f. *PID Calculate* adalah perhitungan PID yang tepat untuk *driver*, sehingga alat dapat berjalan sesuai dengan respon sistem yang diinginkan.
- g. Suhu adalah nilai suhu yang terbaca pada sensor.
- h. Kelembaban adalah nilai kelembaban yang terbaca pada sensor
- i.  $K_i$  merupakan konstanta integratif yang berfungsi menghasilkan keluaran yang tetap pada keadaan *error steady state* dengan nol.
- j.  $K_p$  merupakan konstanta proporsional yang berfungsi sebagai penguat
- k.  $K_d$  merupakan konstanta derivatif yang berfungsi untuk memberikan sebuah *impuls* pada sistem

2. Menentukan *set point* adalah langkah selanjutnya setelah melakukan inisialisasi pada sistem. Langkah ini bertujuan untuk menentukan target yang ingin ditetapkan pada sistem.

3. Setelah *set point* ditentukan langkah selanjutnya adalah pembacaan sensor suhu untuk mengetahui nilai suhu pada sistem.
4. Kemudian nilai *error* akan diketahui dengan cara mencari selisih antara nilai *set point* itu dengan nilai pembacaan sensor.
5. *Decision* adalah proses pengambilan keputusan. Pada sistem ini dibuat empat buah pengambilan keputusan yang terdapat pada sistem ini diantaranya:
  - a. *Decision* (rule 1) berfungsi jika nilai *error* lebih dari 0,1.
  - b. *Decision* (rule 2) berfungsi jika nilai *error* kurang dari 0,1 dan lebih dari -0,5.
  - c. *Decision* (rule 3) berfungsi jika nilai *error* lebih dari -2.
  - d. *Decision* (rule 1) berfungsi sebagai pengambilan keputusan untuk menyelesaikan sistem.

### 3.7 Variabel Penelitian dan Pengamatan

Dalam Penelitian ini, terdapat tiga buah variabel yang digunakan sebagai acuan penelitian yaitu variabel bebas, variabel kontrol, dan variabel terikat. Variabel bebas pada penelitian ini adalah suhu ruangan. Variabel terikat pada penelitian ini adalah air yang diubah – ubah suhunya yang digunakan untuk pengujian. Dan variabel kontrol pada penelitian adalah suhu dan kelembaban pada inkubator.

#### 3.7.1 Pengujian Sensor Suhu

Pengujian sensor suhu dilakukan guna mengetahui kualitas kerja dari sensor dimana kemampuan sensor untuk mendapatkan nilai yang benar – benar sesuai dengan nilai aslinya, sehingga perlu dilakukannya kalibrasi. Proses kalibrasi dilakukan dengan menggunakan *termometer*. Proses kalibrasi diharapkan mendapatkan maksimal nilai *error* persen pada hasil pembacaan sebesar 10%. Perumusan dalam mencari *error* persen dapat dilihat seperti pada Persamaan 3.1.

$$E\% = \left| \frac{HT-HP}{HT} \right| \times 100\% \quad (3.1)$$

dimana :

HT = Hasil Teori (pengukuran *termometer*)

HP = Hasil Pembacaan (pembacaan pada sensor)

### 3.7.2 Pengujian Sensor Kelembaban

Pengujian sensor kelembaban dilakukan untuk mengetahui kualitas kerja dari sensor, dimana kemampuan sensor untuk mendapatkan nilai yang benar – benar sesuai dengan nilai aslinya, sehingga perlu dilakukan proses kalibrasi. Proses kalibrasi dilakukan dengan menggunakan *hygrometer* digital. Proses kalibrasi diharapkan mendapatkan hasil nilai *error* yang pembacaan hasilnya kurang dari 10%. Perumusan dalam mencari nilai dari *error* persen dapat dilihat pada Persamaan 3.2.

$$E\% = \left| \frac{HT-HP}{HT} \right| \times 100\% \quad (3.2)$$

dimana :

HT = Hasil Teori (pengukuran *hygrometer*)

HP = Hasil Pembacaan (pembacaan pada sensor)

### 3.7.3 Pengujian Aktuator Suhu

Pada tahap pengujian ini dilakukan untuk mengetahui bagaimana kerja dari aktuator suhu, sudah sesuai atau belum. Akan diambil beberapa data seperti pada tabel 3.7.

Tabel 3.2 Pengujian Aktuator Suhu

Duty Cycle (%)	PWM	Tegangan Keluaran (V )
10%	X	X
20%	X	X
Dan Seterusnya	X	X

### 3.7.4 Pengujian Metode *Expert PID Control*

Pengujian ini dilakukan setelah semua sensor dan aktuator telah berjalan dengan baik. Tahap ini dilakukan dengan cara menjalankan sistem

dengan menerapkan metode *Expert PID Control* selanjutnya dilakukan pengujian dengan *set point* yang berbeda – beda sehingga dapat dianalisa jalannya dari sistem tersebut apakah sudah baik dan sesuai dengan metode yang telah diterapkan. Sebelum melakukan pengujian sistem ini, terlebih dahulu dilakukan proses *tuning PID* untuk mencari nilai konstanta PID yang sesuai dan akan digunakan dalam sistem tersebut. Berikut langkah – langkah untuk mendapatkan konstanta PID dengan *tuning PID* manual.

1. Menuliskan program pada *Software* arduino untuk *tuning* PID manual pada *dimmer* lampu yaitu :
 

```
pid_output1 = pid_p1 * pid_error1 + pid_i_mem1 + pid_d1 * (pid_error1 - pid_last_error1);
      if (pid_output1 > pid_max1) pid_output1 = pid_max1;
      else if (pid_output1 < pid_max1 * -1) pid_output1 = pid_max1 * -1;
```
2. Menuliskan program pada *Software* arduino untuk *tuning* PID manual pada *drive* kipas yaitu :
 

```
pid_output2 = pid_p2 * pid_error2 + pid_i_mem2 + pid_d2 * (pid_error2 - pid_last_error2);
      if (pid_output2 > pid_max2) pid_output2 = pid_max2;
      else if (pid_output2 < pid_max2 * -1) pid_output2 = pid_max2 * -1;
```
3. Memberikan nilai *input* parameter Kp untuk mengontrol *output* plan sehingga mencapai nilai sesuai dengan *set point*
4. Memberikan nilai *input* parameter Ki untuk mengontrol *output* plan sehingga mencapai nilai sesuai dengan *set point* pada pemberian nilai *input* ini akan meningkat *overshoot* pada respon sistem.
5. Memberikan nilai *input* parameter Kd untuk mengontrol *output* plan sehingga mencapai nilai sesuai dengan *set point*
6. Pada pemberian input untuk Kp, Ki, Kd dilakukan secara berulang – ulang untuk mendapatkan respon sistem yang telah sesuai.

## BAB 5. PENUTUP

Berdasarkan pengujian pada inkubator bayi yang telah dilakukan dengan menggunakan metode Expert PID Control sebagai kontrol suhu dan monitoring kelembaban maka dapat diambil beberapa kesimpulan dan saran yang dapat digunakan sebagai pertimbangan untuk penelitian selanjutnya.

### 5.1 Kesimpulan

Setelah dilakukan beberapa pengujian dan analisa data, maka dapat diambil kesimpulan sebagai berikut :

1. Desain sistem pengendali suhu dan kelembaban pada inkubator bayi prematur ini menggunakan *Module AC Light Dimmer* sebagai aktuator yang mengatur *heater* (lampu) yang menghasilkan panas sebagai keluarannya. Serta menggunakan *Modul Driver L298N* sebagai pengatur nyala kipas untuk membantu menurunkan nilai suhu.
2. Sistem kontrol dengan metode Kontrol PID memiliki respon *transient* yang kurang stabil dibandingkan dengan metode *Expert PID Control*. Pada Kontrol PID untuk mencapai *rise time* dapat dicapai pada waktu 246 detik. Sedangkan *settling time* dapat dicapai pada waktu 768 detik. Dengan overshoot sebesar 1,79%.
3. Sistem kontrol *Expert PID Control* memiliki *output* yang lebih stabil dan lebih optimal dibandingkan dengan kontrol PID biasa dalam mengatur suhu, hal ini dapat dilihat pada gambar 4.20 dan tabel 4.19. Pada *Expert PID Control* besar *overshoot* lebih kecil yaitu 0,21% serta lebih cepat dalam mencapai *settling time* yaitu pada waktu 272 detik, sementara untuk mencapai *rise time* waktu yang dibutuhkan sama seperti pada kontrol PID yaitu 246 detik.

### 5.2 Saran

Setelah dilakukan beberapa pengujian dan analisa data dapat diberikan beberapa saran untuk penelitian ini agar lebih baik sebagai berikut :

1. Disarankan menggunakan sensor suhu yang memiliki kualitas yang lebih baik dan mekanik yang lebih sempurna.
2. Ditambahkan jumlah sensor untuk mengurangi kesalahan pengukuran suhu, sehingga dapat lebih akurat.



## DAFTAR PUSTAKA

- Agam, B. B., & Prihandono, T. (2015). Pengaruh Jenis dan Bentuk Lampu Terhadap Intensitas Pencahayaan dan Energi Buangan Melalui Perhitungan Nilai Efikasi Luminus. *JURNAL PEMBELAJARAN FISIKA* 3(4).
- Alex, A. A. (2019). *STUDI PENGGUNAAN CATU DAYA METODE PWM (PULSE WIDTH MODULATION) 2 PULSA BERBEDA 180° PADA LAMPU LED (LIGHT EMITTING DIODE)* (Doctoral dissertation, Universitas Andalas).
- Ali, M. (2004). Pembelajaran Perancangan Sistem Kontrol PID Dengan Software Matlab. *Jurnal Edukasi@ Elektro* 1(1), 1-8.
- Hernandes, Novita Murti. 2019. Desain Sistem Kontrol Suhu dan PH Untuk Fermentor Stater Mocaf Dengan Menggunakan *Expert PID Controller*
- Hao, Y., Yang, X., Chen, H., & Guan, H. (2010, August). An improved expert intelligent PID algorithm in AQM network. In *2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering* (pp. 234-237). IEEE.
- Islam, H. I., Nabilah, N., Atsaurry, S. S. I., Saputra, D. H., Pradipta, G. M., Kurniawan, A., ... & Irzaman, I. (2016, October). Sistem kendali suhu dan pemantauan kelembaban udara ruangan berbasis arduino uno dengan menggunakan sensor dht22 dan passive infrared (pir). In *Prosiding Seminar Nasional Fisika (E-Journal)* 5, pp. SNF2016-CIP).
- Jacob, M. K., Sawardekar, K. P., Ayoub, H. G., & Al Busaidi, I. (2016). Validation of the existing modified screening criteria for detection of all cases of Retinopathy of Prematurity in preterm babies–11 year study from a governorate referral hospital in Oman. *Saudi Journal of Ophthalmology* 30(1), 3-8.
- Kurniawan, A., & Suprianto, B. (2018). Rancang Bangun Sistem Pengendalian Suhu Pada Inkubator Bayi Berbasis PID Dan LabVIEW 2014. *JURNAL TEKNIK ELEKTRO* 7(3).
- Marwani, Lenty dan Nico Demus River Firman Hutabarat. 2017. Penggunaan

Sensor DHT11 Sebagai Indikator Suhu dan Kelembaban Pada *Baby Incubator*

Meiliya, E., & Pamilih, E. K. (2008). Buku Saku Manajemen Masalah Bayi Baru Lahir Panduan Untuk Dokter, Perawat dan Bidan.

Oliveira, G. H., Amorim, M. F., & Latawiec, K. J. (2005). Multiple model identification and control of neonate incubators using laguerre basis *IAC Proceedings Volume 36* (2005), 289-294.

Osowski, S., Hoai, L. T., & Markiewicz, T. (2004). Support vector machine-based expert system for reliable heartbeat recognition. *IEEE transaction on biomedical engineering* (51), 582-589.

Ovile, R. (2014). *Pengendali Beban Listrik Dengan Sistem Operasi Android Menggunakan Bluetooth HC-06 Berbasis ATMEGA328* Doctoral dissertation, Politeknik Negeri Sriwijaya).

Pratiwi, D. P. (2014). Pemantau dan Pengatur Suhu Inkubator Bayi Berbasis WIFI. *Fakultas Teknik Elektro Universitas Telkom*

SAIFOEDDIN, S. R., Emilia, O., & SpOG, M. M. (2010) *Perawatan metode kanguru pada bayi berat lahir rendah di Rumah Sakit Umum Kabupaten Manokwari tahun 2009* Doctoral dissertation, [Yogyakarta]: Universitas Gadjah Mada).

Setyaningsih, N. Y. D. (2010). *Pembuat Perangkat Lunak Monitoring Temperatur dan Kelembaban Pada Inkubator Bayi Menggunakan Borland Delphi 7.0* Doctoral dissertation, Diponegoro University).

Siwi, A. S., Fatimah, S., & Emaliyawati, E. (2017). Faktor-faktor yang Berhubungan dengan Kecemasan dan Analisis Kebutuhan Orangtua yang Mengalami Kecemasan dengan Bayi Sakit Kritis di NICU RSUD Prof. Dr. Margono Soekardjo Purwokerto. *Journal of Holistic Nursing Science* (4), 32-44.

Sugihatmoko, J. (2015). APLIKASI KONTROLER PID DALAM PENGENDALIAN SUHU INKUBATOR BAYI PREMATUR BERBASIS MIKROKONTROLER ATMEGA2560. *Jurnal Mahasiswa TEUB* (2).

Sugito, H., & Suryono, S. Rancang Bangun Sistem Pengaturan Suhu Ruang

Inkubator Bayi Berbasis Microcontroller AT89S51. *Berkala Fisika* (22), 55-62.

Wijaya, R. A., Lestari, S. W. L. W., & Mardiono, M. (2018). Rancang Bangun Alat Monitoring Suhu dan Kelembaban Pada Alat Baby Incubator Berbasis Internet Of Things. *Jurnal Teknologi* (1), 52.

Wijayanto, R., Subari, A., Manan, S., & Mangkusasmito, F. (2019). RANCANG BANGUN SISTEM HUMIDIFIER DAN DETEKTOR URINASI BAYI UNTUK MENINGKATKAN KAPABILITAS INKUBATOR BAYI BERBASIS MIKROKONTROLLER. *Prosiding SNST Fakultas Teknik*

Xu, J. (2017, March). An expert PID control algorithm based on anti-integration saturation. In *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (AEAE3)* (pp. 1536-1539). IEEE.

## LAMPIRAN

### 1. Listing Program Arduino

```
#include <DHT.h>
#include <PID_v1.h>
#define PWM          3
#define ZCD          2 //uno hanya 2
int frekuensi = 100;
#define inputSerial  0
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Set the LCD address to 0x27 for a 16 chars and
// 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);
DHT dht(A0, DHT22);
DHT dht2(A1, DHT22);
DHT dht3(A2, DHT22);
DHT dht4(A3, DHT22);
volatile uint16_t dutyCycle;
#define maxDutyCycle ((16000000L / (2 * 1024L *
frekuensi)) - 1)
double error;
//Tentukan variabel yang akan disambungkan
//Define Variables we'll be connecting to
double Input, Output;
    unsigned long    previousMillis;

//Specify the links and initial tuning parameters
```

```
int en1 = 6; //inisial en1 masuk pin pwm yaitu 6
int en2 = 5; //inisial en2 masuk pin pwm yaitu 5
int in1 = 13; //inisial pin input1/in1 masuk pin
13
int in2 = 12; //inisial pin input2/in2 masuk pin
12
int in3 = 11; //inisial pin input3/in3 masuk pin
11
int in4 = 9; //inisial pin input4/in4 masuk pin 10
volatile intptr_t *portPWM;
byte bitPWM;

int output;
float out, pid_output1, pid_i_mem1, pid_error1,
pid_last_error1, pid_p1 = 50, pid_i1 = 1, pid_d1 =
0.5, pid_max1 = 100;
double setpoint = 35;
float pid_output2, pid_i_mem2, pid_error2,
pid_last_error2, pid_p2 = 85, pid_i2 = 5, pid_d2 =
0.5, pid_max2 = 255;
int data, data2;
float kelembaban;
float suhu;
float kelembaban11, kelembaban21, kelembaban31,
kelembaban41;
float suhu11, suhu21, suhu31, suhu41;
float a, b, c, d;
float x, y, z, w, dataa, dataa2;
byte termometru[8] = //icon for termometer
{
    B00100,
```

```
B01010,  
B01010,  
B01110,  
B01110,  
B11111,  
B11111,  
B01110  
};  
  
byte picatura[8] = //icon for water droplet  
{  
  B00100,  
  B00100,  
  B01010,  
  B01010,  
  B10001,  
  B10001,  
  B10001,  
  B01110,  
};  
  
void setup() {  
  lcd.begin();  
  
  // Turn on the backlight and print a message.  
  lcd.backlight();  
  lcd.print("S1 Teknik Elektro");  
  lcd.setCursor(1, 1);  
  lcd.print("Artika");  
  lcd.setCursor(8, 1);  
  lcd.print("Rosida");
```

```
pinMode(en1, OUTPUT); //pin en1 sebagai OUTPUT
pinMode(en2, OUTPUT); //pin en2 sebagai OUTPUT
pinMode(in1, OUTPUT); //pin in1 sebagai OUTPUT
pinMode(in2, OUTPUT); //pin in2 sebagai OUTPUT
pinMode(in3, OUTPUT); //pin in3 sebagai OUTPUT
pinMode(in4, OUTPUT); //pin in4 sebagai OUTPUT
Serial.begin(9600);
lcd.createChar(1, termometru);
lcd.createChar(2, picatura);
dht.begin();
dht2.begin();
dht3.begin();
dht4.begin();
TCCR2A = _BV(WGM21);
TCCR2B = _BV(CS22) | _BV(CS21) | _BV(CS20);
OCR2A = maxDutyCycle;
OCR2B = setDutyCycle(0);
TIMSK2 = _BV(OCIE2B);
*digitalPinToPCMSK(ZCD) |= bit
(digitalPinToPCMSKbit(ZCD));
PCIFR |= bit (digitalPinToPCICRbit(ZCD));
PCICR |= bit (digitalPinToPCICRbit(ZCD));
portPWM = (volatile intptr_t *)
portOutputRegister(digitalPinToPort(PWM));
bitPWM = digitalPinToBitMask(PWM);
dutyCycle = setDutyCycle(0);
pinMode(PWM, OUTPUT);
//turn the PID on
// delay(2000);
}
```

```
void loop() {
  lcd.clear();
  lcd.setCursor (0, 1);
  //munculkan icon
  lcd.write(1);
  // Kolom ke 2 baris ke 2
  lcd.setCursor (2, 1);
  lcd.print(suhu);
  lcd.print((char)223);
  lcd.setCursor (10, 1);
  lcd.write(2);
  lcd.setCursor (12, 1);
  lcd.print(kelembaban);

  float kelembaban1 = dht.readHumidity();
  float kelembaban2 = dht2.readHumidity();
  float kelembaban3 = dht3.readHumidity();
  float kelembaban4 = dht4.readHumidity();
  float suhu1 = dht.readTemperature();
  float suhu2 = dht2.readTemperature();
  float suhu3 = dht3.readTemperature();
  float suhu4 = dht4.readTemperature();
  //rumus kalibrasi DHT22
  a = (kelembaban1 + 4.9318) / 1.0128;
  kelembaban11 = a; //inisialisasi kelembaban1
  setelah kalibrasi
  b = (kelembaban2 + 4.9318) / 1.0128;
  kelembaban21 = b; //inisialisasi kelembaban2
  setelah kalibrasi
  c = (kelembaban3 + 4.9318) / 1.0128;
```

```
    kelembaban31 = c; //inisialisasi kelembaban1
setelah kalibrasi
    d = (kelembaban4 + 4.9318) / 1.0128;
    kelembaban41 = d; //inisialisasi kelembaban2
setelah kalibrasi

    kelembaban = (kelembaban11 + kelembaban21 +
kelembaban31 + kelembaban41 ) / 4;

    x = ( suhu1 + 6.5135) / 1.2743;
    suhu11 = x;
    y = ( suhu2 + 6.5135) / 1.2743;
    suhu21 = y;
    z = ( suhu3 + 6.5135) / 1.2743;
    suhu31 = z;
    w = ( suhu4 + 6.5135) / 1.2743;
    suhu41 = w;

    suhu = (suhu11 + suhu21 + suhu31 + suhu41 ) / 4;
    ExpertPID();
    unsigned long    currentMillis = millis();
    if (currentMillis - previousMillis >=1000){
        Serial.print("Suhu: ");
        Serial.print(suhu);
        Serial.print(" Kelembaban: ");
        Serial.print(" ");
        Serial.println(kelembaban);
    }
    previousMillis = currentMillis;
}

    delay(1000);
```

```
uint16_t setDutyCycle(uint8_t dutyCycle)
{
    return maxDutyCycle - (1.0 * dutyCycle / 100 *
(maxDutyCycle - 1)) + 1;
}
#if defined(__AVR_ATmega328P__)
# if ((ZCD >= 0) & (ZCD <= 7))
ISR (PCINT2_vect)
# elif ((ZCD >= 8) & (ZCD <= 13))
ISR (PCINT0_vect)
# elif ((ZCD >= 14) & (ZCD <= 19))
ISR (PCINT1_vect)
# endif
#elif defined(__AVR_ATmega2560__)
# if ((ZCD >= 50) & (ZCD <= 53)) || ((ZCD >= 10)
& (ZCD <= 13))
ISR (PCINT0_vect)
# elif ((ZCD >= 14) & (ZCD <= 15))
ISR (PCINT1_vect)
# elif ((ZCD >= A8) & (ZCD <= A15))
ISR (PCINT2_vect)
# endif
#endif
{
    OCR2B = dutyCycle;
    TCNT2 = 0;
}
ISR (TIMER2_COMPB_vect)
{
    *portPWM |= bitPWM;
    delayMicroseconds(30);
}
```

```
    *portPWM &= ~bitPWM;
}
void kipas(int s1, int s2) {
    analogWrite(en1, s1);
    digitalWrite(in1, LOW);
    digitalWrite(in2, HIGH);
    analogWrite(en2, s2);
    digitalWrite(in3, HIGH);
    digitalWrite(in4, LOW);
}
}
```

## 2. Listing Program Expert PID Control

```
void ExpertPID() {
    error = setpoint - suhu ;
    //Rule 1
    if (error >= 0.1 ) {
        output = 100;
        dutyCycle = setDutyCycle(output);
        kipas(0,0);
    }
    //Rule PID
    else if (error >= 0 && error < 0.1){
        pid_calculate1();
        pid_calculate2();
    }
    else if (error >= (-0.5) && error < 0 ) {
        output = 0;
        dutyCycle = setDutyCycle(output);
        pid_calculate2();
    }
}
```

```
    }  
    //Rule 3  
    else if (error >= (-2) && error < (-0.5)) {  
        dutyCycle = setDutyCycle(0);  
        kipas(255,255);  
    }  
    else if (error <(-2)) {  
        dutyCycle = setDutyCycle(0);  
        kipas(255,255);  
    }  
}
```

### 3. Listing Program PID Controller

```
void pid_calculate1() { //Dimmer Lampu  
    pid_error1 = suhu - setpoint;  
    pid_i_mem1 += pid_i1 * pid_error1;  
    if (pid_i_mem1 > pid_max1)pid_i_mem1 = pid_max1;  
    else if (pid_i_mem1 < pid_max1 * -1)pid_i_mem1 =  
pid_max1 * -1;  
  
    pid_output1 = pid_p1 * pid_error1 + pid_i_mem1 +  
pid_d1 * (pid_error1 - pid_last_error1);  
    if (pid_output1 > pid_max1)pid_output1 =  
pid_max1;  
    else if (pid_output1 < pid_max1 * -1)pid_output1  
= pid_max1 * -1;  
  
    pid_last_error1 = pid_error1;  
  
    data = abs(pid_output1);  
    // dataa = dataa + data;  
    // dutyCycle = setDutyCycle(dataa);
```

```
    if (pid_output1 < 0) {
        dutyCycle = setDutyCycle(data);
    }
    else {
        dutyCycle = setDutyCycle(0);
    }
}

void pid_calculate2() { //Driver kipas
    pid_error2 = suhu - setpoint;
    pid_i_mem2 += pid_i2 * pid_error2;
    if (pid_i_mem2 > pid_max2)pid_i_mem2 = pid_max2;
    else if (pid_i_mem2 < pid_max2 * -1)pid_i_mem2 =
pid_max2 * -1;

    pid_output2 = pid_p2 * pid_error2 + pid_i_mem2 +
pid_d2 * (pid_error2 - pid_last_error2);
    if (pid_output2 > pid_max2)pid_output2 =
pid_max2;
    else if (pid_output2 < pid_max2 * -1)pid_output2
= pid_max2 * -1;

    pid_last_error2 = pid_error2;
    data2 = abs(pid_output2 * 100);
    dataa2 = dataa2 + data;

    if (pid_output2 < 0) {
        kipas(0, 0);
    }
    else {
        kipas(data2, data2);
    }
}
```

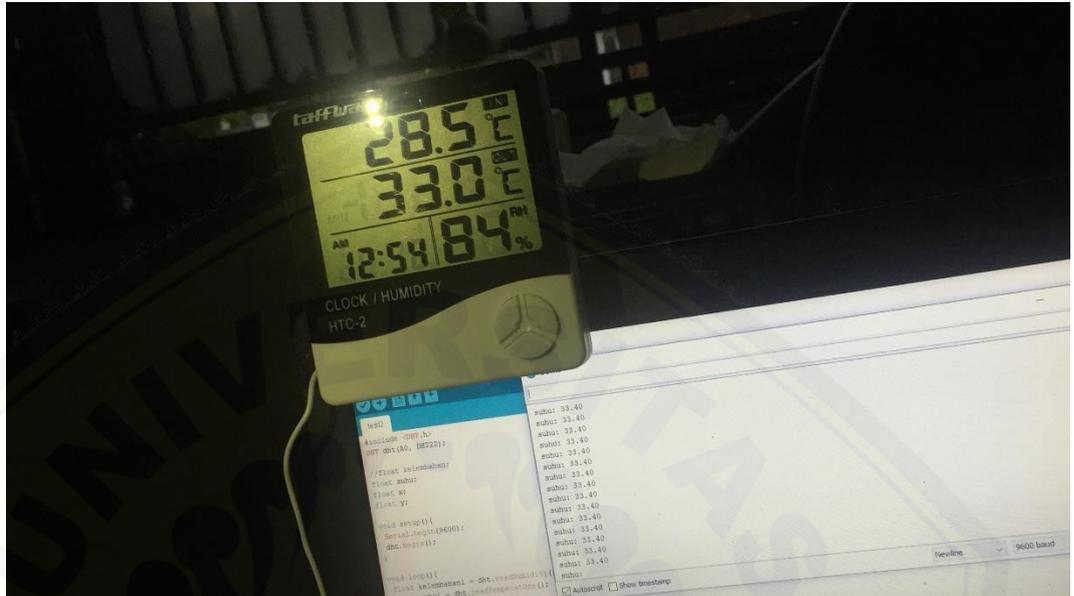
```
}  
}
```

#### 4. Listing Program Pengujian Aktuator Suhu

```
#include "RBDdimmer.h"  
//#define USE_SERIAL SerialUSB //Serial for  
boards whith USB serial port  
#define USE_SERIAL Serial  
#define outputPin 3  
#define zerocross 2 // for boards with CHANGEABLE  
input pins  
  
//dimmerLamp dimmer(outputPin, zerocross);  
//initialase port for dimmer for ESP8266, ESP32,  
Arduino due boards  
dimmerLamp dimmer(outputPin); //initialase port  
for dimmer for MEGA, Leonardo, UNO, Arduino M0,  
Arduino Zero  
  
int outVal = 0;  
  
void setup() {  
    USE_SERIAL.begin(9600);  
    dimmer.begin(NORMAL_MODE, ON); //dimmer  
initialisation: name.begin(MODE, STATE)  
    USE_SERIAL.println("Dimmer Program is  
starting...");  
    USE_SERIAL.println("Set value");  
}  
void printSpace(int val)  
{
```

```
    if ((val / 100) == 0) USE_SERIAL.print(" ");
    if ((val / 10) == 0) USE_SERIAL.print(" ");
}
void loop() {
    int preVal = outVal;
    if (USE_SERIAL.available())
    {
        int buf = USE_SERIAL.parseInt();
        if (buf != 0) outVal = buf;
        delay(200);
    }
    dimmer.setPower(outVal); // setPower(0-100%);
    if (preVal != outVal)
    {
        USE_SERIAL.print("lampValue -> ");
        printSpace(dimmer.getPower());
        USE_SERIAL.print(outVal);
        USE_SERIAL.println("%");
    }
    delay(50);
}
```

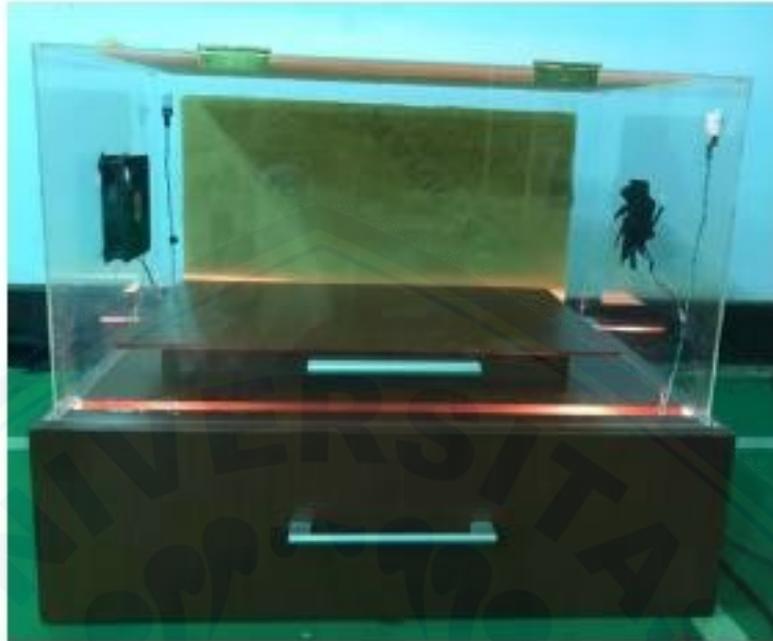
## LAMPIRAN 2



Gambar 1. Proses Kalibrasi Sensor Suhu



Gambar 2. Proses Kalibrasi Sensor Kelembaban



Gambar 3. Proses Pengujian Suhu Tanpa Beban



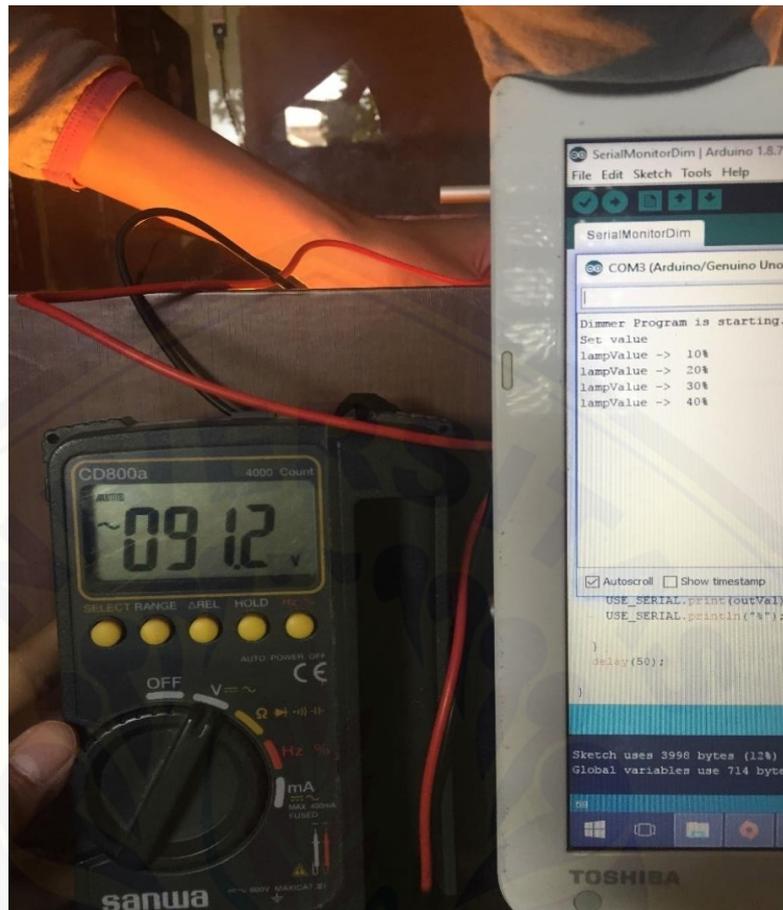
Gambar 4. Proses Pengujian Suhu dengan Beban



Gambar 5. Proses Pengambilan Data Kelembaban



Gambar 6. Tampilan Suhu dan Kelembaban pada LCD



Gambar 6. Proses Pengujian Aktuator Suhu