

**SISTEM PENGENDALIAN SUHU DAN KELEMBABAN PADA  
BROODER BIBIT AYAM DAY OLD CHICKEN (DOC)  
MENGUNAKAN KONTROLER PID**

**SKRIPSI**

**TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL**

Ditujukan untuk memenuhi persyaratan  
memperoleh gelar Sarjana Teknik



**JOHANES REINHART PANDANG**  
**NIM. 115060307111030**

**UNIVERSITAS BRAWIJAYA**  
**FAKULTAS TEKNIK**

**MALANG**

**2016**

viii

**LEMBAR PENGESAHAN**

**SISTEM PENGENDALIAN SUHU DAN KELEMBABAN PADA  
BROODER BIBIT AYAM DAY OLD CHICKEN (DOC)  
MENGUNAKAN KONTRLER PID**

**SKRIPSI**

**TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL**

Diajukan untuk Memenuhi Persyaratan  
Memperoleh Gelar Sarjana Teknik



**JOHANES REINHART PANDANG  
NIM. 115060307111030**

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Ir. Purwanto, M.T.  
NIP. 19540424 198601 1 001**

**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 ini dapat dibuktikan terdapat unsur-unsur jiplakan, saya bersedia Skripsi dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, pasal 25 ayat 2 dan pasal 70).

Malang, 29 Maret 2016

Mahasiswa,

**JOHANES REINHART PANDANG**

**NIM. 115060307111030**

UNIVERSITAS BRAWIJAYA



*Teriring Ucapan Terima Kasih kepada:  
Ayahanda dan Ibunda tercinta*

## RINGKASAN

**Johanes Reinhart Pandang**, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya, Maret 2016, *Sistem Pengendalian Suhu dan Kelembaban pada Brooder Bibit Ayam Day Old Chicken (DOC) Menggunakan Kontroler PID*, Dosen Pembimbing : Purwanto dan Muhammad Aziz Muslim.

Masalah yang sering dihadapi dalam proses tumbuh kembang bibit ayam atau *Day Old Chicken* adalah kebutuhan suhu dan kelembaban di dalam tempat bibit ayam atau *brooder* harus sesuai dengan kebutuhan bibit, sehingga dibutuhkan sistem pengendalian suhu dan kelembaban yang dapat disesuaikan dengan kebutuhan bibit. Suhu dan kelembaban yang diaplikasikan di dalam *brooder* untuk bibit ayam broiler dengan suhu 32°C dan kelembaban antara 55 - 60%RH.

Pada proses pengendalian suhu di dalam *brooder* menggunakan kontroler PID dan proses pengendalian kelembaban menggunakan kontroler ON – OFF. Metode yang digunakan untuk mendapatkan parameter PID menggunakan metode Ziegler – Nichols 1, sehingga didapatkan nilai  $K_p = 35,56$   $K_i = 1,975$   $K_d = 160,02$ .

Hasil pengujian pada suhu awal 23,21 °C dengan *setpoint* suhu 32°C diperoleh *error steady state* sebesar 0,23% dan *settling time* sebesar 680 detik dan kelembaban berada pada batas 55 – 60%RH. Saat diberi gangguan sistem juga mampu menjaga suhu dan kelembaban yang sesuai dengan *setpoint*.

**Kata kunci** : *Day Old Chicken (DOC)*, Bibit Ayam, *Brooder*, Kontroler PID, Kontroler ON-OFF, Ziegler-Nichols 1

## SUMMARY

**Johanes Reinhart Pandang**, Department of Electrical Engineering, Faculty of Engineering of Brawijaya Univerdity, March 2016, Temperature and Humidity Control System on Chicken Seed Brooder of Day Old Chicken (DOC) Using PID Controller, Supervisors: Purwanto and Muhammad Aziz Muslim.

A problem which is usually met in the process of growing chicken seed or Day Old Chicken is that the temperature and humidity have to be in tune with the seed's needs, so a temperature and humidity control system is needed. Temperature and humidity applied on the brooder for chicken seed is 32°C for temperature and 55 - 60%RH for humidity.

The temperature control process uses PID controller for its control while the humidity control process uses ON – OFF controller. To get PID parameters, Ziegler – Nichols 1 method is used, so the value of 35.56 for Kp, 1.975 for Kd, and 160.02 for Kd is obtained.

Research results with starting temperature of 23.21°C and setpoint of 32°C are steady state error of 0.23% and settling time of 680 seconds while the humidity stays in the range of 55 – 60%RH. When a disturbance is given, the system was able to keep the temperature and humidity stay matched with the setpoint.

**Keywords** : Day Old Chicken (DOC), Chicken Seed, Brooder, PID Controller, ON-OFF Controller, Ziegler-Nichols 1

## PENGANTAR

Puji syukur kepada Tuhan Yesus Kristus atas hikmat, bimbingan, dan kuasa-Nya sehingga penulisan skripsi dengan judul “Sistem Pengendalian Suhu dan Kelembaban pada *Brooder Bibit Ayam Day Old Chicken (DOC)* Menggunakan Kontroler PID” dapat diselesaikan dengan baik dan tepat pada waktunya.

Penulis menyadari bahwa penyusunan skripsi ini tidak lepas dari bantuan, bimbingan serta dorongan dari berbagai pihak. Pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar – besarnya kepada:

- Kedua orang tua Stefanus Charles Pandang dan A.A. Istri Candrawati, dan adik Maria Veronica Pandang telah banyak memberikan doa, kasih sayang, dukungan, serta semangat.
- Bapak M. Aziz Muslim, S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya Malang dan dosen pembimbing skripsi yang telah memberikan banyak waktu dan tenaganya untuk membimbing dari awal, memberikan saran, nasehat-nasehat, dan pelajaran..
- Bapak Ir. Purwanto, M.T. selaku Ketua Kelompok Dosen Keahlian Teknik Kontrol Teknik Elektro, Fakultas Teknik Universitas Brawijaya dan dosen pembimbing skripsi yang telah memberikan banyak waktu dan tenaganya untuk membimbing dari awal, memberikan saran, nasehat-nasehat, dan pelajaran.
- Bapak Hadi Suyono S.T., M.T., Ph.D selaku Sekretaris Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya.
- Bapak M. Ali Mustofa, S.T., M.T. selaku Ketua Program Studi Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya yang banyak memberikan pengarahan dalam hal akademik dan penulisan skripsi.
- **Bapak Ir. Unggul Wibawa, M.Sc.** Selaku dosen penasehat akademik. Yang telah banyak memberikan nasihat-nasihat akademiknya.

- Bapak, Ibu dosen serta segenap staf dan karyawan Jurusan Teknik Elektro baik secara langsung maupun tidak langsung yang telah banyak membantu dalam menyelesaikan skripsi ini.
- Saudara-saudara Inverter 2011, KMK Teknik, dan semua rekan-rekan di Jurusan Teknik Elektro dan Fakultas Teknik Universitas Brawijaya.
- Muhammad Zainudin, ST. selaku pembimbing ketiga dalam penelitian atas segala bimbingan, nasihat, arahan, motivasi, saran dan masukannya.
- Tim *Engine Control Unit (ECU)* Ganda Lesmana, Intanto Oktavian, Yudhanto Iman Noorizky.
- Bu Pri, Renos, Borisxa, Diyan, Asdi, Jonas, Dana, Silas, Bob, Pampam, Tesu, Manda, Wayan, Syahriel, Afrizal, dan Kelita Mea Melaca, serta teman-teman dekat penulis yang berada di Bali ataupun Malang atas dukungannya baik secara materi maupun rohani.
- Seluruh rekan-rekan asisten Laboratorium Sistem Kontrol yang telah memfasilitasi penelitian ini.
- Teman-teman Deworengku. Terima kasih semuanya.
- Teman-teman serta semua pihak yang tidak mungkin dicantumkan namanya satu persatu telah memberikan bantuan serta dukungan baik secara langsung maupun tidak langsung atas penyusunan skripsi ini.

Dalam penyusunan skripsi ini, penulis menyadari bahwa skripsi ini belumlah sempurna, karena keterbatasan ilmu dan kendala – kendala lain yang terjadi selama pengerjaan skripsi ini. Oleh karena itu, penulis berharap kritik dan saran untuk penyempurnaan tulisan di masa yang akan datang, semoga tulisan ini dapat bermanfaat dan dapat digunakan untuk pengembangan lebih lanjut.

Malang, Maret 2016

Penulis



**DAFTAR ISI**

<b>RINGKASAN</b> .....	i
<b>SUMMARY</b> .....	ii
<b>PENGANTAR</b> .....	iii
<b>DAFTAR ISI</b> .....	v
<b>DAFTAR TABEL</b> .....	viii
<b>DAFTAR GAMBAR</b> .....	ix
<b>BAB I PENDAHULUAN</b> .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan .....	2
1.5 Sistematika Pembahasan .....	2
<b>BAB II TINJAUAN PUSTAKA</b> .....	3
2.1 Teori Dasar Kebutuhan Suhu dan Kelembaban untuk Ayam ( <i>DOC</i> ).....	5
2.2 Sensor Suhu dan Kelembaban .....	5
2.3 Pemanas .....	7
2.4 <i>Humidifier</i> .....	7
2.5 <i>Driver Motor L298N</i> .....	7
2.6 Rangkaian <i>Dimmer</i> .....	8
2.6.1 Rangkaian TRIAC .....	9
2.6.2 Rangkaian <i>Zero Cross Detector</i> .....	9
2.7 Arduino Mega .....	10
2.8 Catu Daya .....	11



2.9	PWM .....	11
2.10	Kontroler PID (Proporsional Integral Diferensial) .....	12
2.10.1	Kontroler Proporsional .....	12
2.10.2	Kontroler Integral .....	13
2.10.3	Kontroler Diferensial .....	14
2.10.4	Kontroler Proporsional Integrral Diferensial (PID) .....	14
2.10.5	Metode Tuning PID .....	15

**BAB III METODOLOGI .....** 19

3.1	Studi Literatur .....	19
3.2	Penentuan Spesifikasi Alat .....	19
3.3	Perancangan Sistem .....	20
3.4	Realisasi Pembuatan Sistem .....	20
3.5	Pengujian dan Analisa Data .....	20
3.6	Pengambilan Kesimpulan .....	20

**BAB IV PERANCANGAN DAN PEMBUATAN ALAT .....** 21

4.1	Perancangan Sistem .....	21
4.2	Diagram Blok Sistem .....	21
4.3	Spesifikasi Alat .....	22
4.4	Prinsip Kerja .....	23
4.5	Perancangan Perangkat Keras .....	23
4.5.1	Pemilihan Pemanas .....	23
4.5.2	Rangkaian <i>Dimmer</i> .....	24
4.5.3	Pemodelan Sensor SHT11 .....	25
4.6	Perancangan Algoritma Kontroler PID .....	25
4.7	Modul Arduino Mega 2560 .....	29
4.8	Perancangan Perangkat Keras .....	31

**BAB V PENGUJIAN DAN ANALISIS SISTEM .....** 33

5.1	Pengujian Sensor Suhu dan Kelembaban SHT11 .....	33
5.1.1	Peralatan yang Digunakan .....	33
5.1.2	Prosedur Pengujian .....	33



5.1.3	Hasil Pengujian.....	34
5.2	Pengujian Pengujian Sensor Kelembaban SHT11.....	35
5.2.1	Peralatan yang Digunakan.....	35
5.2.2	Prosedur Pengujian.....	35
5.2.3	Hasil Pengujian.....	36
5.3	Pengujian Penyulutan TRIAC.....	37
5.3.1	Peralatan yang Digunakan.....	37
5.3.2	Prosedur Pengujian.....	37
5.3.3	Hasil Pengujian.....	38
5.4	Pengujian Rangkaian <i>Dimmer</i> .....	43
5.4.1	Peralatan yang Digunakan.....	43
5.4.2	Prosedur Pengujian.....	44
5.4.3	Hasil Pengujian.....	44
5.5	Pengujian Keseluruhan Sistem.....	46
5.5.1	Peralatan yang Digunakan.....	46
5.5.2	Prosedur Pengujian.....	46
5.5.3	Hasil Pengujian.....	47
<b>BAB VI PENUTUP</b> .....		51
6.1	Kesimpulan.....	51
6.2	Saran.....	52
<b>DAFTAR PUSTAKA</b> .....		53
<b>LAMPIRAN 1</b> .....		55
<b>LAMPIRAN 2</b> .....		57
<b>LAMPIRAN 3</b> .....		61

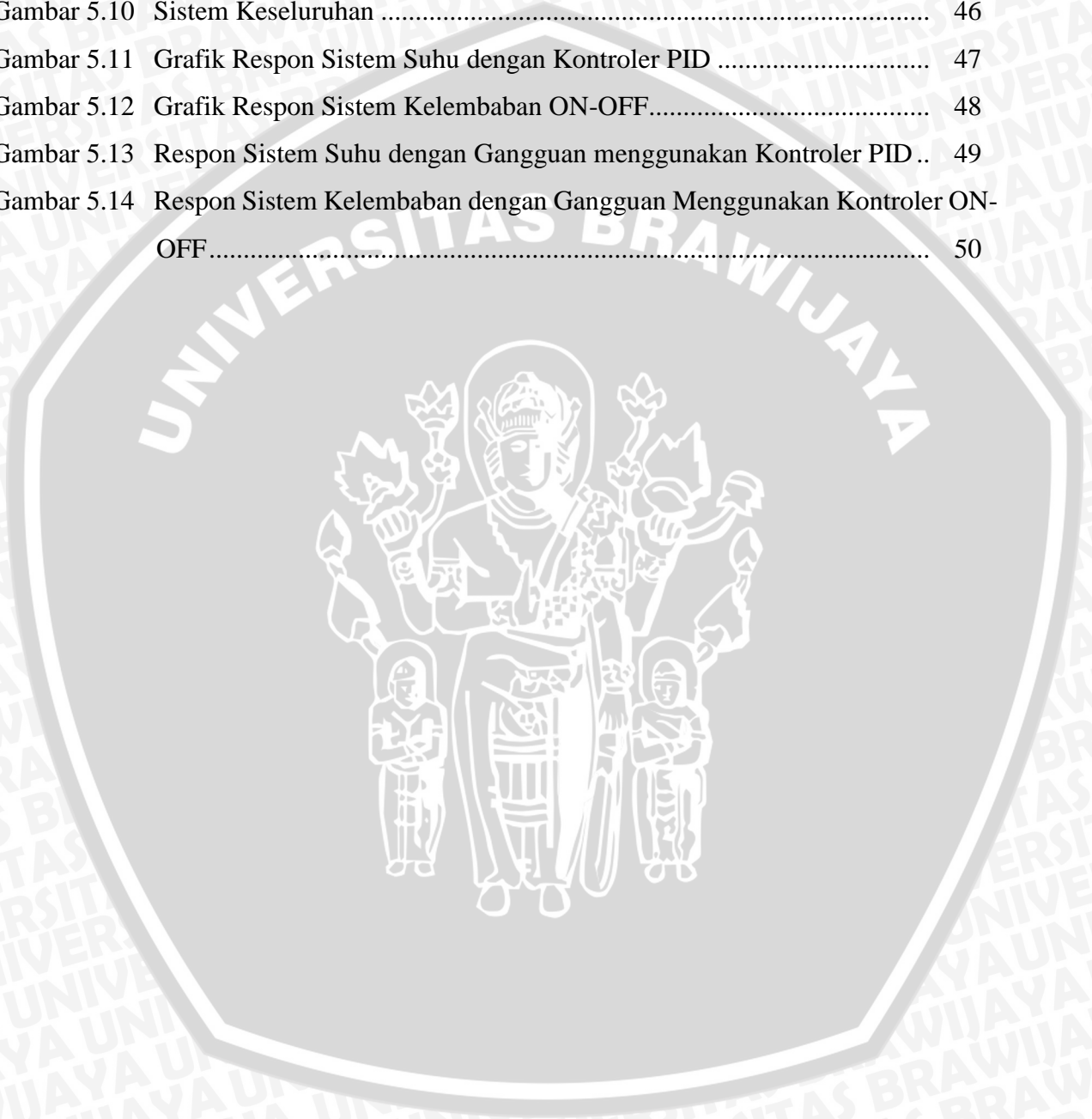
## DAFTAR TABEL

Tabel 2.1	Suhu dan Kelembaban untuk Ayam <i>DOC</i> .....	5
Tabel 2.2	Aturan Penalaan Ziegler-Nichols Berdasarkan Respon Unit Step dari Plant .....	17
Tabel 2.3	Aturan Ziegler-Nichols Berdasarkan <i>Critical Gain Kcr</i> dan <i>Critical Period Pcr</i> .....	18
Tabel 4.1	Penggunaan Pin pada Arduino Mega.....	30
Tabel 5.1	Hasil Pengujian Sensor Suhu SHT11 .....	34
Tabel 5.2	Hasil Pengujian Sensor Kelembaban SHT11 .....	36
Tabel 5.3	Hasil Pengujian Sudut Penyalan TRIAC .....	39
Tabel 5.4	Kesalahan Pengujian PWM Penyalan TRIAC.....	43
Tabel 5.5	Perbandingan Nilai PWM Terhadap Tegangan .....	44
Tabel 5.6	Perbandingan Nilai PWM Terhadap <i>steady state</i> Suhu.....	45
Tabel 5.7	Data Respon Suhu Menggunakan Kontroler PID .....	47
Tabel 5.8	Data Respon Suhu dengan Gangguan Menggunakan Kontroler PID.....	49

## DAFTAR GAMBAR

Gambar 2.1	Bibit Ayam <i>DOC</i> Broiler.....	5
Gambar 2.2	Sensor SHT11 .....	6
Gambar 2.3	Rangkaian Sensor SHT11 .....	6
Gambar 2.4	<i>Humidifier USB</i> .....	7
Gambar 2.5	<i>Driver</i> Motor LN298N.....	8
Gambar 2.6	Kipas DC 12V.....	8
Gambar 2.7	<i>AC Dimmer Module</i> .....	8
Gambar 2.8	Arduino Mega .....	10
Gambar 2.9	Pengaturan Tegangan AC dengan PWM .....	11
Gambar 2.10	Diagram Blok Kontroler Proporsional.....	12
Gambar 2.11	Diagram Blok Kontroler Integral.....	13
Gambar 2.12	Diagram Blok Kontroler Diferensial .....	14
Gambar 2.13	Diagram Blok Kontroler PID.....	15
Gambar 2.14	Kurva Respn <i>Unit Step</i> yang Menunjukkan <i>25% Maximum Overshoot</i> . 15	
Gambar 2.15	Respon Plan Terhadap Masukan Berupa <i>Unit Step</i> .....	16
Gambar 2.16	Respon Plan Berbentuk S.....	16
Gambar 2.17	Sistem Loop Tertutup dengan Kontroler Proporsional.....	17
Gambar 2.18	Osilasi Berkesinambungan dengan Periode <i>Pcr</i> .....	18
Gambar 4.1	Diagram Blok <i>Plant Brooder</i> Bibit Ayam <i>DOC</i> .....	21
Gambar 4.2	Pemanas pada <i>Brooder</i> Bibit Ayam <i>DOC</i> .....	24
Gambar 4.3	Skema Rangkaian <i>Dimmer</i> .....	24
Gambar 4.4	Modul Rangkaian <i>Dimmer</i> .....	25
Gambar 4.5	Kurva Respon yang Berbentuk S .....	26
Gambar 4.6	Aturan Metode 1 <i>Ziegler-Nichols</i> .....	27
Gambar 4.7	Karakteristik <i>Open Loop</i> Respon PWM 35 .....	27
Gambar 4.8	Modul Arduino Mega 2560 .....	30
Gambar 4.9	<i>Flowchart</i> Program Keseluruhan.....	31
Gambar 5.1	Blok Pengujian Sensor Suhu SHT11 .....	34
Gambar 5.2	Perbandingan Suhu Sensor SHT11 terhadap Termometer Digital .....	35
Gambar 5.3	Blok Pengujian Sensor Kelembaban SHT11 .....	36
Gambar 5.4	Grafik Pengujian Sensor Kelembaban SHT11 .....	37

Gambar 5.5	Rangkaian Pengujian Sudut Penyalaan TRIAC.....	38
Gambar 5.6	Bentuk Gelombang Keluaran Arduino .....	38
Gambar 5.7	Bentuk Gelombang Keluaran TRIAC dengan PWM 32.....	39
Gambar 5.8	Grafik Hubungan Nilai PWM dengan Tegangan.....	44
Gambar 5.9	Karakteristik Plant .....	45
Gambar 5.10	Sistem Keseluruhan .....	46
Gambar 5.11	Grafik Respon Sistem Suhu dengan Kontroler PID .....	47
Gambar 5.12	Grafik Respon Sistem Kelembaban ON-OFF.....	48
Gambar 5.13	Respon Sistem Suhu dengan Gangguan menggunakan Kontroler PID ..	49
Gambar 5.14	Respon Sistem Kelembaban dengan Gangguan Menggunakan Kontroler ON-OFF.....	50



## BAB I PENDAHULUAN

### 1.1. Latar Belakang

Perekembangan industri peternakan ayam menjadi salah satu peluang usaha yang menjanjikan bagi para peternak karena jumlah permintaan ayam pedaging maupun petelur yang banyak di Indonesia. Untuk memenuhi permintaan akan ayam yang memiliki kualitas dalam jumlah yang besar, dibutuhkan penyediaan bibit ayam atau yang biasa disebut *Day Old Chicken (DOC)* yang baik.

Pada pemeliharaan boiler secara komersil, pada umumnya anak ayam diperoleh dari penetasan dengan menggunakan mesin tetas. Oleh sebab itu dibutuhkan induk buatan sebagai pengganti untuk melindungi anak ayam dari kondisi lingkungan yang buruk. Dengan adanya induk buatan tersebut maka anak ayam akan dapat tumbuh dengan baik. Sistem induk buatan inilah yang sering dikenal dengan istilah *brooding* (Rima Nastiti, 2014).

Tujuan dari *brooding* adalah untuk menyediakan lingkungan yang sesuai bagi bibit ayam (*DOC*). Pada masa ini merupakan masa yang paling menentukan, karena akan berpengaruh terhadap pertumbuhan masa selanjutnya. Pada saat bibit ayam berumur 0 sampai 14 hari, akan terjadi perbanyakan sel. Perbanyakan sel ini meliputi saluran pencernaan, perkembangan saluran pernapasan dan sistem kekebalan tubuh. Keberhasilan masa *brooding* ini sangat dipengaruhi oleh suhu dan kelembaban di dalam *brooder*. Sebelum umur dua minggu inilah peranan penghangat dan kelembaban buatan di dalam *brooder* sangat penting karena dapat menjaga suhu dan kelembaban dengan kebutuhan bibit ayam.

Pada prinsipnya *brooder* harus memiliki suhu hangat yang stabil sesuai dengan kebutuhan setiap hari. Suhu pada *brooder* dapat ditentukan dari umur bibit ayam. Berdasarkan pada kondisi suhu dan kelembaban yang ideal tersebut dibutuhkan suatu kontrol suhu dan kelembaban yang mampu beradaptasi dengan kondisi suhu lingkungan di luar agar didapatkan suhu dan kelembaban yang sesuai untuk kebutuhan bibit ayam. Salah satu metode yang dapat digunakan adalah pengaturan suhu pada *brooder* bibit ayam (*DOC*) menggunakan kontroler PID untuk pengendalian suhu dan kontroler ON-OFF pada pengendalian kelembaban. Dengan adanya kontroler pada PID pada *brooder* bibit ayam diharapkan suhu dapat diatur sesuai kebutuhan bibit.

### 1.2. Rumusan Masalah

Berdasarkan latar belakang dapat disusun rumusan masalah sebagai berikut: Bagaimana merancang perangkat keras dan perangkat lunak tempat pembibitan ayam (*brooder*) yang dapat mengendalikan suhu dan kelembaban secara otomatis?

### 1.3. Batasan Masalah

Dalam perancangan, masalah-masalah yang dibatasi adalah sebagai berikut:

1. *Brooder* yang digunakan adalah *prototype* dengan desain sendiri berdimensi 80 x 60 x 80 cm.

2. Sistem kontrol PID diimplementasikan menggunakan Arduino Mega 2560.
3. Media pemanas yang digunakan adalah bola lampu pijar AC 60W.
4. Nyala terang redup lampu dikendalikan oleh *dimmer*.
5. Media penambah kelembaban yang digunakan adalah *humidifier*.
6. Kipas DC 12V digunakan untuk sirkulasi udara dan menurunkan suhu yang berlebihan.
7. *Humidifier* dan Kipas diatur menggunakan kontroler ON OFF.
8. Sensor suhu dan kelembaban yang digunakan adalah SHT11.
9. Pengendalian suhu diatur dengan menggunakan 2 bola lampu pijar dengan tipe yang sama dengan masing-masing 60W.
10. Kinerja *driver* dan elektronika tidak dibahas mendalam.
11. Tidak menggunakan bibit ayam (*DOC*) terlebih dahulu sebagai objek percobaan.

#### 1.4. Tujuan

Tujuan dari penelitian ini adalah untuk membuat alat yang mampu mengatur suhu dan kelembaban yang sesuai dengan kebutuhan bibit ayam (*DOC*).

#### 1.5. Sistematika Pembahasan

##### **BAB I Pendahuluan**

Bab ini berisi tentang uraian latar belakang, tujuan, batasan masalah, rumusan masalah, manfaat serta sistematika penulisan.

##### **BAB II Tinjauan Pustaka**

Tinjauan Pustaka berisi dasar teori penunjang penelitian. Pustaka yang diambil adalah pustaka yang relevan dan sesuai serta mendukung penelitian, seperti buku-buku budidaya bibit ayam (*DOC*), teori kontrol PID Ziegler-Nichols 1, sensor suhu dan kelembaban, identifikasi sistem, Arduino dan lain-lain. Selain dari buku pustaka juga akan diambil dari jurnal, internet, dan sumber pengetahuan yang lain.

##### **BAB III Metodologi Penelitian**

Bab ini berisi tentang metode yang digunakan dalam pengerjaan alat seperti perancangan dan pembuatan rangkaian *interface*, pengujian alat, pengambilan data dan analisis data yang digunakan dalam skripsi ini.



#### **BAB IV Perancangan dan Pembuatan**

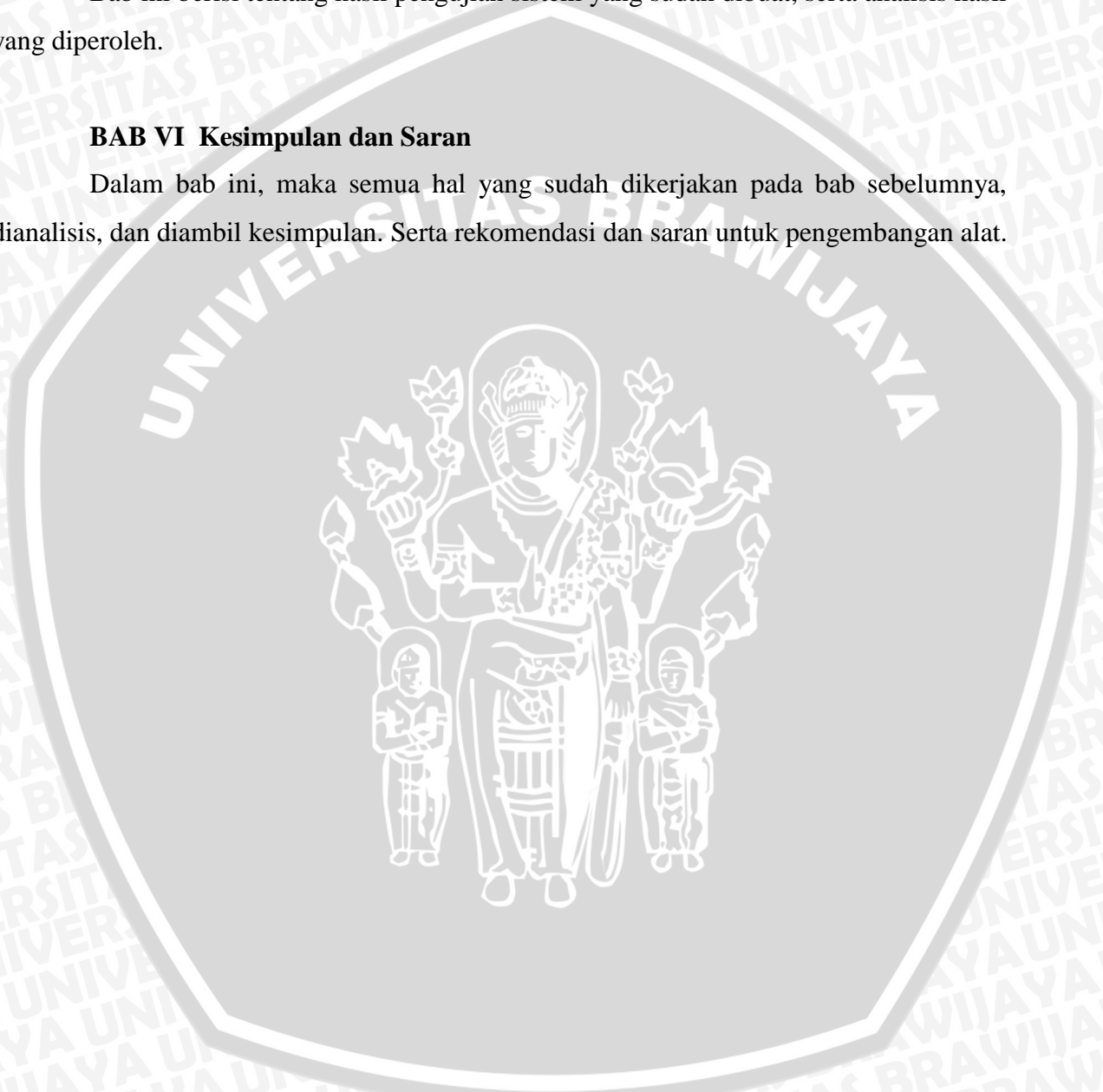
Bab ini menjelaskan tentang perancangan dan pembuatan alat yang meliputi *block diagram* sistem, prinsip kerja alat, perancang perangkat keras dan perangkat lunak.

#### **BAB V Pengujian dan Analisis Sistem**

Bab ini berisi tentang hasil pengujian sistem yang sudah dibuat, serta analisis hasil yang diperoleh.

#### **BAB VI Kesimpulan dan Saran**

Dalam bab ini, maka semua hal yang sudah dikerjakan pada bab sebelumnya, dianalisis, dan diambil kesimpulan. Serta rekomendasi dan saran untuk pengembangan alat.



## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Teori Dasar Kebutuhan Suhu dan Kelembaban untuk Ayam *Day Old Chicken (DOC)*.

Bibit ayam atau anakan ayam yang dapat dikatakan *DOC* merupakan ayam yang baru saja menetas dari telurnya sampai berumur 2 minggu (Insani Galuh Adi, 2010). Kebutuhan suhu pada kandang atau *brooder* ayam *DOC* dapat diberikan dengan bantuan pemanas modern atau pemanas konvensional. Pemanas konvensional biasanya hanya menggunakan lampu petromaks ataupun lampu pijar yang tidak dapat dikontrol intensitas kecerahan pada lampu pijarnya. Untuk penambah kelembaban dapat berupa *humidifier*. Berikut tabel kebutuhan suhu dan kelembaban yang nyaman untuk ayam *DOC*.

**Tabel 2.1** Suhu dan Kelembaban untuk ayam *DOC*

Umur (hari)	Suhu (°C)	Kelembaban (%RH)
0-3	31-33	55-60
4-7	31-32	55-60
8-14	28-30	55-60
15-21	26-28	55-60
22-24	23-26	55-60

(Sumber: Rima Nastiti, 2014)



**Gambar 2.1** Bibit Ayam *DOC* Broiler

(Sumber: [www.ayambroiler.com](http://www.ayambroiler.com))

#### 2.2. Sensor Suhu dan Kelembaban

SHT11 Module merupakan modul sensor suhu dan kelembaban relatif. Modul ini dapat digunakan sebagai alat pengindra suhu dan kelembaban dalam aplikasi pengendali suhu dan kelembaban ruangan maupun aplikasi pemantau suhu dan kelembaban relatif

ruangan. Sensor SHT11 ini adalah sensor suhu dan kelembaban relatif berbasis sensirion SHT11.

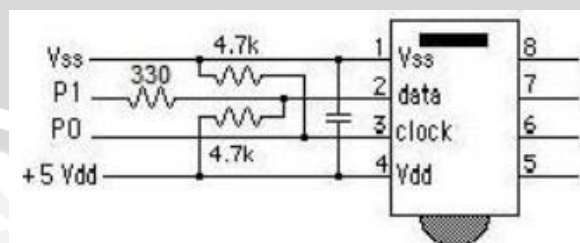
Sensor SHT11 dapat mengukur suhu dari  $-40^{\circ}\text{C}$  hingga  $123,8^{\circ}\text{C}$  atau dari  $-40\text{F}$  hingga  $4,9\text{F}$  dan kelembaban relatif dari  $0\%\text{RH}$  hingga  $1\%\text{RH}$ . Memiliki ketepatan (akurasi) pengukuran suhu hingga  $0,5^{\circ}\text{C}$  pada suhu  $^{\circ}\text{C}$  dan ketepatan (akurasi) pengukuran kelembaban relatif hingga  $3,5\%\text{RH}$ . Memiliki antarmuka serial *synchronous 2-wire*, bukan I2C. Jalur antarmuka telah dilengkapi dengan rangkaian pencegah kondisi sensor *lock-up*. Sensor SHT11 membutuhkan catu daya  $+5\text{V DC}$  dengan konsumsi daya rendah  $30\mu\text{W}$  dan memiliki faktor bentuk 8 pin DIP 0,6 sehingga memudahkan pemasangannya.

Pin yang digunakan pada sensor ini ada dua pin yaitu pin SCK dan pin DATA. Pin SCK berfungsi untuk mensinkronisasikan komunikasi antara sensor dan Arduino, sedangkan pin DATA berfungsi untuk mentransfer data in dan out pada sensor. Dibawah ini gambar menunjukkan sensor SHT11 dan gambar menunjukkan rangkaian sensor suhu dan kelembaban (SHT11).



**Gambar 2.2** Sensor SHT11

(Sumber: [innovativeelectronics.com](http://innovativeelectronics.com))



**Gambar 2.3** Rangkaian Sensor SHT11

(Sumber: [www.sensiron.com](http://www.sensiron.com))

### 2.3. Pemanas (Heater)

Pemanas yang digunakan dalam perancangan alat ini berupa 2 buah lampu pijar AC 60W. Dengan adanya rangkaian dimmer lampu pijar dapat menyesuaikan tingkat cahaya sesuai dengan suhu yang dibutuhkan ayam *DOC* (dari tabel 2.1).

Persyaratan pemanas antara lain :

- Harus tahan lama pada suhu yang dikehendaki.
- Sifat mekanisnya harus kuat pada suhu yang dikehendaki.
- Koefisien muai harus kecil, sehingga perubahan bentuknya pada suhu yang dikehendaki tidak terlalu besar.

### 2.4. Humidifier

*Humidifier* berfungsi sebagai penambah kelembaban pada *brooder* dengan cara menyemburkan kabut dari air melalui proses kavitasi. Aktuator yang digunakan dalam perancangan alat berupa alat penambah kelembaban ruangan yang terdapat dipasaran. Driver yang digunakan untuk menjalankan *humidifier* berupa relay yang aktif ketika diberi tegangan 5V. *Humidifier* diharapkan dapat menjaga kelembaban dengan sesuai kebutuhan ayam *DOC*.

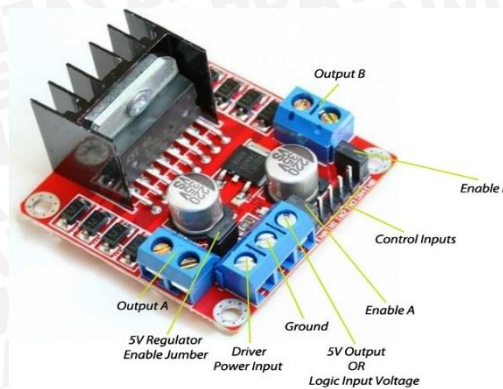


**Gambar 2.4** *Humidifier USB*

(Sumber: [www.tokopedia.com](http://www.tokopedia.com))

### 2.5. Driver Motor L298N

*Driver* motor L298N seperti ditunjukkan Gambar 2.5 digunakan untuk mengendalikan putaran motor DC yang menjadi penggerak kipas DC 12V sebagai media untuk menurunkan suhu pada *brooder*. Modul ini dihubungkan dengan output dari mikrokontroler Arduino Mega 2560.



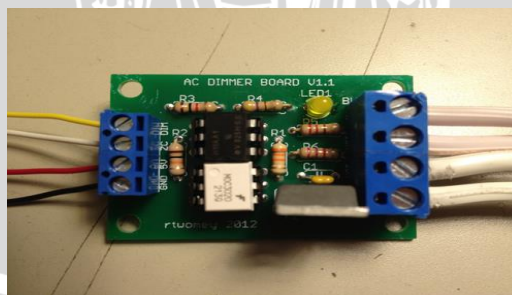
**Gambar 2.5** Driver Motor LN298N  
(Sumber: electrosome.com)



**Gambar 2.6** Kipas DC 12V  
(Sumber: www.globalindustrial.com)

**2.6. Rangkaian Dimmer**

Diperlukan sebuah rangkaian yang bisa digunakan untuk melakukan pengendalian AC, rangkaian tersebut rangkaian *dimmer* yang di dalamnya terdiri dari rangkaian TRIAC dan rangkaian *Zero Cross Detector*. Gambar 2.7 menunjukkan contoh modul rangkaian *dimmer* AC.



**Gambar 2.7** AC Dimmer Module  
(Sumber: www.seeedstudio.com)

### 2.6.1 Rangkaian TRIAC

TRIAC atau *Triode for Alternating Current* (Trioda untuk arus bolak-balik) adalah komponen yang tersusun atas dua buah *thyristor* antiparalel dengan terminal *gate* yang sama. Nama resmi untuk TRIAC adalah *Bidirectional Triode Thyristor*. Ini menunjukkan sakelar dwiarah yang dapat mengalirkan arus listrik ke kedua arah ketika dipicu (dihidupkan). Konfigurasi tersebut membuat TRIAC merupakan komponen yang digunakan untuk mengatur tegangan AC sehingga pemicuan dari *gate* TRIAC juga dapat dilakukan dengan pulsa negatif atau pulsa positif (sesuai tegangan AC) (Ardhito, 2013).

Dalam penggunaannya dibutuhkan DIAC sebagai *driver* dari TRIAC. Selain itu dibutuhkan juga komponen yang mampu berfungsi sebagai isolator antara tegangan 220V dan 5V untuk melindungi mikrokontroler dari tegangan balik. Sehingga komponen yang sesuai dengan spesifikasi di atas adalah *optocoupler*. Optocoupler yang komponen utamanya berupa DIAC adalah MOC3022.

### 2.6.2 Rangkaian Zero Cross Detector

*Zero crossing detector* adalah rangkaian yang digunakan untuk mendeteksi gelombang sinus AC 220 volt saat melewati titik tegangan nol. Seberangan titik nol yang dideteksi adalah peralihan dari positif menuju negatif dan peralihan dari negatif menuju positif. Seberangan-seberangan titik nol ini merupakan acuan yang digunakan sebagai awal pemberian nilai waktu tunda untuk pemicuan TRIAC.

Pada rangkaian *zero cross detector* dibutuhkan rangkaian yang bisa mendeteksi persimpangan antara gelombang tegangan bolak-balik dengan titik nol agar pengaturan fasa dapat dilakukan pada titik nol gelombang tegangan bolak-balik. Metode ini berfungsi untuk menentukan frekuensi suatu gelombang dengan cara mendeteksi banyaknya *zero point* pada suatu rentang waktu. *Zero crossing detector* berfungsi untuk mendeteksi perpotongan gelombang sinus pada tegangan AC dengan *zero point* tegangan AC tersebut, sehingga dapat memberikan sinyal acuan saat dimulainya pemicuan sinyal *Pulse Width Modulation* (PWM). Dengan rangkaian *zero crossing detector*, bisa dilakukan pendeteksian *zero point* sekaligus mengubah suatu sinyal sinusoidal menjadi sinyal kotak. Sinyal keluaran rangkaian *zero cross detector* ini akan dimasukkan ke mikrokontroler, oleh karena itu

dibutuhkan juga komponen yang mampu memisahkan tegangan 5V dan 220V. Untuk rangkaian ini digunakan komponen optocoupler 4N25.

## 2.7. Arduino Mega

Arduino Mega adalah board mikrokontroler berbasis Atmega1280. Memiliki 54 pin *input* dan *output* digital, dimana 14 pin diantara pin tersebut dapat digunakan sebagai *output* PWM dan 16 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 Mega ke komputer dengan menggunakan kabel USB atau dengan adaptor AC-DC untuk menjalankannya.

Secara fisik, ukuran Arduino Mega hampir kurang lebih 2 kali lebih besar dari Arduino Uno, ini untuk mengakomodasi lebih banyaknya pin Digital dan Analog pada board Arduino Mega tersebut. Tampilan Arduino Mega dapat dilihat pada gambar di bawah ini.



**Gambar 2.8** Arduino Mega

(Sumber: <http://arduino.cc/en/Main/ArduinoBoardMega>)

## 2.8. Catu Daya

Arduino mega dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber catu daya akan dipilih secara otomatis. Eksternal (non-USB) catu daya dapat berasal baik dari adaptor AC-DC atau baterai. Adaptor ini dapat dihubungkan dengan menancapkan sebuah 2.1mm konektor pusat-positif ke konektor listrik pada board. Baterai dapat dimasukkan ke dalam header pin Gnd dan Vin dari konektor catu daya.

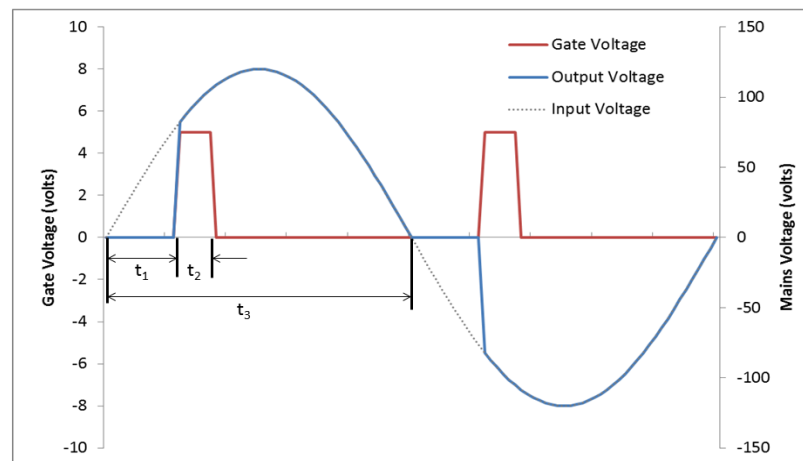
Board dapat beroperasi pada pasokan eksternal 7 sampai 12 volt. apabila dicatu kurang dari 7V, pin 5V dapat menyediakan kurang dari 5V dan board bisa menjadi tidak stabil. Jika menggunakan lebih dari 12V, regulator tegangan bisa panas dan merusak board. Kisaran yang disarankan adalah 7 sampai 12 volt.

Pin untuk catu daya adalah sebagai berikut:

- VIN: Adalah *input* tegangan untuk *board* Arduino ketika menggunakan sumber catu daya eksternal.
- 5V: Sebuah pin yang mengeluarkan tegangan ter-regulator 5 Volt. Dari pin ini tegangan sudah diatur (ter-regulator) dari regulator yang tersedia (built-in) pada *board*. Arduino dapat diaktifkan dengan sumber catu daya baik berasal dari jack power DC (7-12 Volt), konektor USB (5 Volt), atau pin VIN pada *board* (7-12 Volt). Memberikan tegangan melalui pin 5V atau 3.3V secara langsung tanpa melewati regulator dapat merusak *board* Arduino.
- 3.3V: Sebuah pin yang menghasilkan tegangan 3,3 Volt. Tegangan ini dihasilkan oleh regulator yang terdapat pada papan (on-board). Arus maksimum yang dihasilkan adalah 50 mA.
- GND: Pin Ground.

## 2.9. PWM

PWM (*Pulse Width Modulation*) digunakan untuk mengatur tegangan yang masuk pada heater. Gambar 2.9 menunjukkan sinyal PWM (*Pulse Width Modulation*) untuk mengatur tegangan AC yang masuk pada *heater*.



**Gambar 2.9** Pengaturan Tegangan AC dengan PWM

(Sumber: <http://playground.arduino.cc>)

Dalam Gambar 2.9, setelah persimpangan nol gelombang terdeteksi, triac akan off dalam waktu ( $t_1$ ). Saat triac off maka tegangan AC juga masih off. Setelah triac aktif (ditampilkan dalam warna merah), gelombang AC juga akan keluar hingga saat gelombang melewati persimpangan nol. Walaupun triac sudah tidak aktif ( $t_2$ ), gelombang AC akan tetap aktif sampai persimpangan nol. Hal ini akan terus berulang saat gelombang melewati persimpangan nol yang selanjutnya ( $t_3$ ).

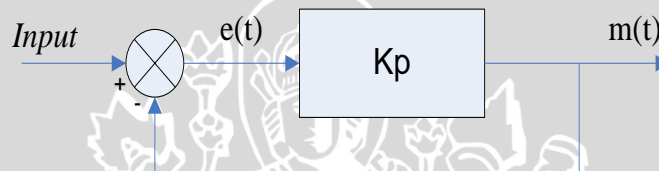


## 2.10. Kontroler PID (Proporsional Integral Diferensial)

### 2.10.1 Kontroler Proporsional

Kontroler proporsional memiliki keluaran yang sebanding/ proporsional dengan besar sinyal kesalahan (selisih antara besaran yang diinginkan dengan harga aktualnya). Secara lebih sederhana dapat dikatakan, bahwa keluaran kontroler proporsional merupakan perkalian antara konstanta proporsional dengan masukannya. Perubahan pada sinyal masukan akan segera menyebabkan sistem secara langsung mengubah keluarannya sebesar konstanta pengalinya.

Pada Gambar 2.10 menunjukkan diagram blok yang menggambarkan hubungan antara *input* (besaran referensi yang diinginkan), besaran aktual dengan besaran keluaran kontroler proporsional, dan besaran kesalahan (*error*). Sinyal kesalahan merupakan selisih antara besaran *setting* dengan besaran aktualnya.



**Gambar 2.10** Diagram Blok Kontroler Proporsional

(Sumber: Katsuhiko Ogata, 1997)

Pada pengendali proporsional hubungan antara keluaran kontroler  $m(t)$  dan sinyal kesalahan  $e(t)$  adalah

$$m(t) = K_p e(t) \dots \dots \dots (2-1)$$

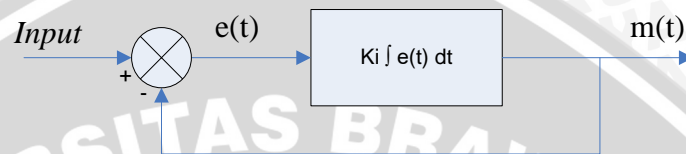
Dengan  $K_p$  adalah penguatan proporsional. Keluaran  $m(t)$  hanya tergantung pada  $K_p$  dan *error*, semakin besar *error* maka semakin besar koreksi yang dilakukan. Penambahan  $K_p$  akan menaikkan penguatan sistem sehingga dapat digunakan untuk memperbesar kecepatan respons dan mengurangi kesalahan keadaan mantap.

### 2.10.2 Kontroler Integral

Kontroler integral berfungsi mengurangi kesalahan keadaan mantap yang dihasilkan pada kontroler proporsional sebelumnya. Kalau sebuah *plant* tidak memiliki unsur integrator ( $1/s$ ),

kontroler proporsional tidak akan mampu menjamin keluaran sistem dengan kesalahan keadaan mantap nol.

Kontroler integral memiliki karakteristik seperti halnya sebuah integral. Keluaran kontroler sangat dipengaruhi oleh perubahan yang sebanding dengan nilai sinyal kesalahan. Keluaran kontroler ini merupakan akumulasi yang terus menerus dari perubahan masukannya. Kalau sinyal kesalahan tidak mengalami perubahan, keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan. Gambar 2.11 menunjukkan diagram blok kontroler integral.



**Gambar 2.11** Diagram Blok Kontroler Integral  
(Sumber: Katsuhiko Ogata, 1997)

Nilai keluaran kontroler  $m(t)$  sebanding dengan integral sinyal kesalahan  $e(t)$ , sehingga :

$$\frac{dm(t)}{dt} = Ki \cdot e(t) \dots\dots\dots(2-2)$$

$$m(t) = Ki \int_0^t e(t)dt \dots\dots\dots(2-3)$$

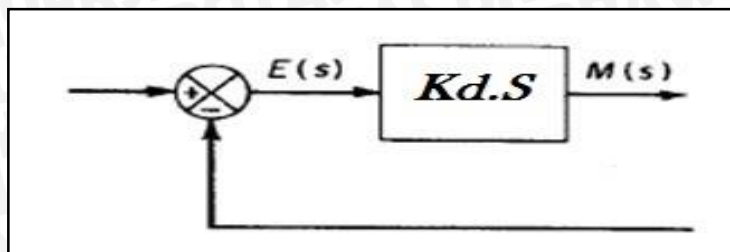
Dengan  $K_i$  adalah konstanta integral. Jika sinyal kesalahan  $e(t)=0$ , maka laju perubahan sinyal kendali integral  $\frac{dm(t)}{dt} = 0$  atau sinyal keluaran kendali akan tetap berada pada nilai yang dicapai sebelumnya. Aksi kontrol integral digunakan untuk menghilangkan kesalahan posisi dalam keadaan mantap (*error steady state*) tanpa memperhitungkan kecepatan respons.

**2.10.3 Kontroler Diferensial**

Kontroler ini digunakan untuk memperbaiki atau mempercepat respon transien sebuah sistem kontrol dengan cara memperbesar *phase lead* terhadap penguatan kontrol dan mengurangi *phase lag* penguatan tersebut. Kontroler diferensial seperti yang terlihat dalam Gambar 2.12 tidak dapat mengeluarkan *output* bila tidak ada perubahan *input*, selain itu kontroler diferensial tidak dapat digunakan untuk proses yang mengandung *noise*. Hubungan antara keluaran kontroler  $m(t)$  dan sinyal kesalahan penggerak  $e(t)$  adalah (Persamaan 2-4):



$$\frac{M(s)}{E(s)} = K_d \cdot s \dots \dots \dots (2-4)$$



**Gambar 2.12** Diagram Blok Kontroler Diferensial

(Sumber: Katsuhiko Ogata, 1997)

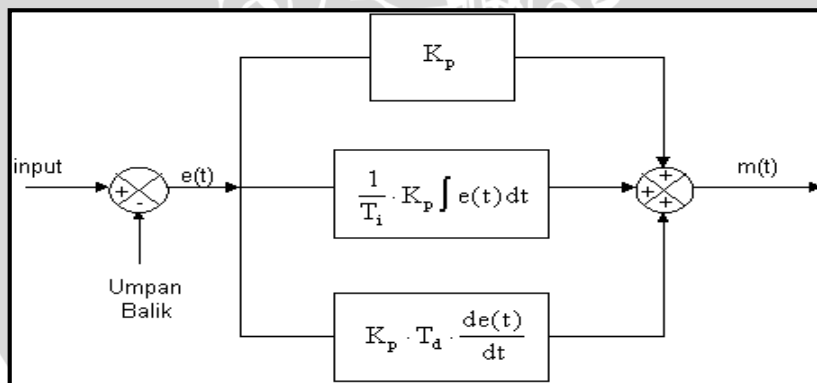
**2.10.4 Kontroler Proporsional Integral Diferensial (PID)**

Aksi kontrol proporsional, integral, dan diferensial yang terlihat dalam Gambar 2.13 mempunyai keunggulan dapat saling menutupi kekurangan dan kelebihan dari masing-masing kontroler. Persamaan kontroler PID ini dapat dinyatakan sebagai berikut (Persamaan 2-5):

$$m(t) = K_p \cdot e(t) + \frac{K_p}{T_i} \cdot e(t)dt + K_p \cdot T_d \frac{de(t)}{dt} \dots \dots \dots (2-5)$$

Dalam transformasi laplace dinyatakan sebagai berikut (Persamaan 2-6):

$$\frac{M(s)}{E(s)} = K_p \left( 1 + \frac{1}{T_i \cdot s} + T_d \cdot s \right) \dots \dots \dots (2-6)$$



**Gambar 2.13** Diagram Blok Kontroler PID

(Sumber: Katsuhiko Ogata, 1997)

**2.10.5 Metode Tuning PID**

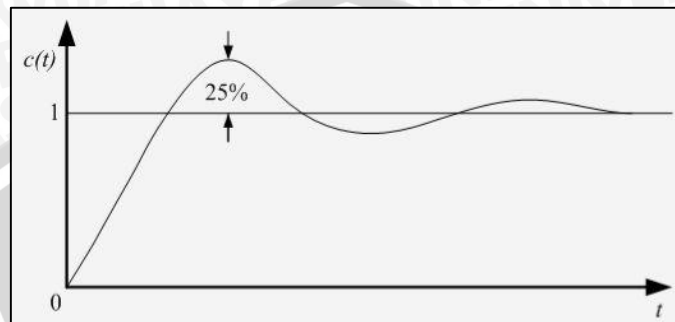
- **Metode Ziegler-Nichols**

Ziegler dan Nichols mengemukakan aturan-aturan untuk menentukan nilai dari gain proporsional (Kp), waktu integral (Ti), dan waktu derivative (Td) berdasarkan karakteristik respon



transien dari *plant* yang diberikan. Penentuan parameter kontroler PID atau penalaan kontroler PID tersebut dapat dilakukan dengan bereksperimen dengan plan (Ogata K., 1997).

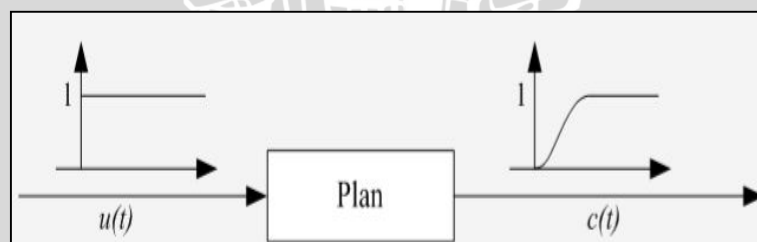
Terdapat dua metode yang disebut aturan penalaan Ziegler-Nichols, pada kedua metode tersebut memiliki tujuan yang sama yaitu untuk mencapai 25% *maximum overshoot* pada respon unit step, seperti ditunjukkan dalam Gambar 2.14.



**Gambar 2.14** Kurva Respon *unit step* yang menunjukkan 25% *Maximum Overshoot*  
(Sumber: Katsuhiko Ogata, 1997)

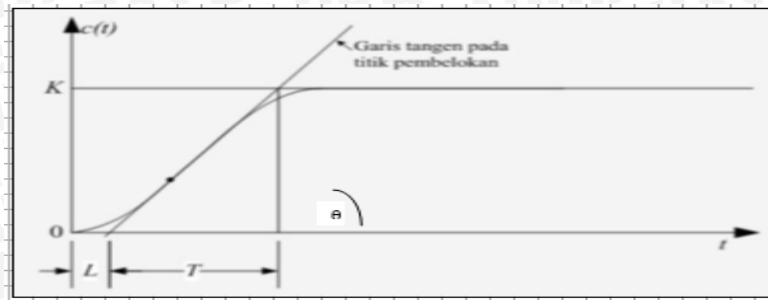
a. Metode Pertama

Metode pertama atau sering disebut metode kurva reaksi, respon dari *plan* dapat diperoleh secara eksperimental dengan masukan berupa *unit step*, seperti yang ditunjukkan dalam Gambar 2.15.



**Gambar 2.15** Respon Plan Terhadap Masukan Berupa *Unit Step*  
(Sumber: Katsuhiko Ogata, 1997)

Jika dalam plan tersebut terdapat integrator atau *dominan complex-conjugate poles*, maka kurva respon unit *step* berbentuk seperti huruf S, seperti dalam Gambar 2.16. Jika respon tidak memberikan bentuk kurva S, maka metode ini tidak berlaku. (Ogata K., 1997).



**Gambar 2.16** Respon Plan berbentuk S  
(Sumber: Katsuhiko Ogata, 1997)

Kurva berbentuk S tersebut dapat dikarakteristikan menjadi dua konstanta yaitu waktu tunda  $L$  dan konstanta waktu  $T$ . Waktu tunda dan konstanta waktu ditentukan dengan menggambar sebuah garis tangen pada titik pembelokan kurva S, dan menentukan perpotongan antara garis tangen dengan sumbu waktu  $t$  dan sumbu  $c(t) = K$ , seperti yang telah ditunjukkan dalam Gambar 2.16, fungsi alih  $C(s)/U(s)$  dapat dilakukan pendekatan dengan sistem orde satu dengan persamaan sebagai berikut:

$$\frac{C(s)}{U(s)} = \frac{Ke^{-Ls}}{Ts + 1} \dots\dots\dots(2-7)$$

Ziegler dan Nichols menyarankan untuk menentukan nilai dari  $K_p$ ,  $T_i$  dan  $T_d$  berdasarkan pada formula yang ditunjukkan dalam Tabel 2.2 (Katsuhiko Ogata, 1997)

**Tabel 2.2** Aturan Penalaan Ziegler-Nichols Berdasarkan Respon Unit Step Dari Plan

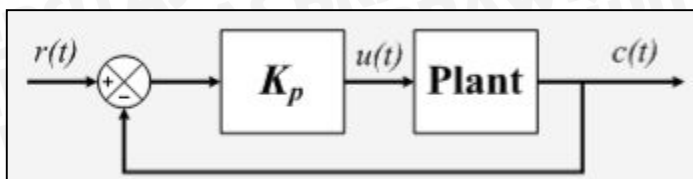
Tipe Kontroler	$K_p$	$T_i$	$T_d$
P	$\frac{T}{L}$	$\infty$	0
PI	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{T}{L}$	$2L$	$0,5 L$

(Sumber: Katsuhiko Ogata, 1997)

b. Metode Kedua

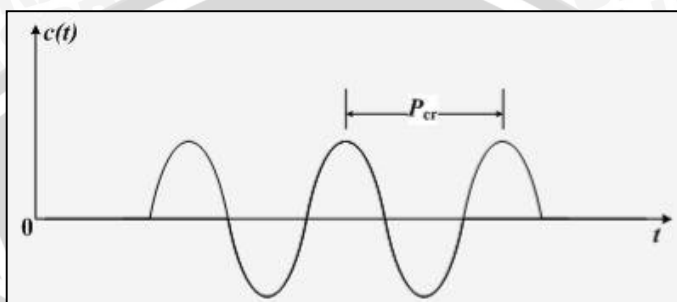
Dalam metode kedua Ziegler-Nichols, mula-mula yang dilakukan adalah membuat  $T_i=0$  dan  $T_d=0$ . Kemudian hanya dengan menggunakan tindakan kontrol proporsional, harga ditingkatkan dari nol ke suatu nilai kritis  $K_{cr}$ , disini mula-mula keluaran memiliki osilasi yang berkesinambungan (jika keluaran tidak memiliki osilasi berkesinambungan untuk nilai  $K_p$  manapun yang telah diambil, maka metode ini tidak berlaku). Dari keluaran yang berosilasi secara berkesinambungan, penguatan kritis  $K_{cr}$  dan periode  $P_{cr}$  dapat ditentukan. Diagram blok sistem loop tertutup dengan kontroler proporsional dapat dilihat dalam Gambar 2.17 dan untuk osilasi berkesinambungan dengan periode  $P_{cr}$  dapat dilihat dalam Gambar 2.18 Ziegler dan Nichols menyarankan penyetapan nilai parameter  $K_p$ ,  $T_i$ ,  $T_d$  dan berdasarkan rumus yang diperlihatkan dalam Tabel 2.3. (Katsuhiko Ogata, 1997)





**Gambar 2.17** Sistem Loop Tertutup dengan Kontroler Proporsional

(Sumber: Katsuhiko Ogata, 1997)



**Gambar 2.18** Osilasi Berkesinambungan dengan periode  $P_{cr}$

(Sumber: Katsuhiko Ogata, 1997)

**Tabel 2.3** Aturan Dasar Ziegler-Nichols Berdasarkan *Critical Gain Kcr* dan *Critical Period Pcr*

Tipe Kontroler	$K_p$	$T_i$	$T_d$
P	$0.5 K_{cr}$	$\infty$	0
PI	$0.45 K_{cr}$	$\frac{1}{1,2} P_{cr}$	0
PID	$0.60 K_{cr}$	$0.5 P_{cr}$	$0.125 P_{cr}$

(Sumber: Katsuhiko Ogata, 1997)

## BAB III METODOLOGI

Untuk merealisasikan alat yang telah dirancang, langkah-langkah yang dilakukan adalah sebagai berikut:

### 3.1. Studi Literatur

Studi literatur fokus dilakukan pada prinsip kerja alat yang dirancang meliputi sistem pengendalian suhu pada *brooder* ayam *DOC*, karakteristik dari komponen yang digunakan serta studi tentang penggunaan perangkat Arduino Mega sebagai perangkat akuisisi data.

Adapun hal-hal yang berhubungan dengan hal tersebut adalah:

- a. Studi tentang budidaya bibit ayam serta suhu yang dibutuhkan oleh bibit ayam.
- b. Studi tentang spesifikasi sistem yang digunakan dan rangkaian elektrik pendukung lainnya.

### 3.2. Penentuan Spesifikasi Alat

Adapun spesifikasi alat sebagai berikut:

- a. *Brooder*, merupakan tempat berlangsungnya proses pengendalian suhu dan kelembaban ayam *DOC* atau bibit ayam dengan dimensi 80 x 60 x 80 cm.
- b. Lampu pijar AC 60 W sebagai elemen pemanas.
- c. Kipas DC 12 V akan membantu proses penurunan suhu di dalam *brooder*.
- d. *Humidifier*, berfungsi sebagai penambah kelembaban di dalam *brooder* dengan cara menyemburkan uap dari proses kavitasi.
- e. Sensor SHT11, untuk membaca suhu dan kelembaban di dalam *brooder*.
- f. Penyalaaan terang redup lampu AC dikendalikan menggunakan *dimmer*.
- g. *Dimmer* menggunakan kontroler PID
- h. Kipas dan *humidifier* menggunakan kontroler ON-OFF.

### 3.3. Perancangan Sistem

Perancangan sistem dilakukan sebagai langkah awal sebelum terbentuknya suatu sistem beserta rangkaian elektronik serta pemrogramannya, hal ini dimaksudkan agar proses pembuatan *brooder* yang sesuai dengan kebutuhan ayam *DOC* dapat berjalan sesuai deskripsi awal yang telah direncanakan. Perancangan sistem yang dilakukan meliputi:

- a. Penentuan dimensi dan bahan yang digunakan dalam pembuatan *brooder*.
- b. Perancangan elektronik.
- c. Perancangan program pada perangkat Arduino Mega sebagai media akuisisi data untuk proses identifikasi *plant*.
- d. Blok diagram sistem.

#### 3.4. Realisasi Pembuatan Sistem

Untuk merealisasikan alat atau sistem yang telah dirancang secara keseluruhan, maka perlu diperhatikan beberapa hal dalam pembuatan alat tersebut, diantaranya yaitu:

- a. Pembuatan purwarupa *brooder* secara keseluruhan yang meliputi pemilihan *heater*, sensor, kipas, *humidifier* dan komponen lainnya.
- b. Pembuatan rangkaian elektronik yang meliputi rangkaian *dimmer*, *driver*, *relay*, LCD dan pengendali sinyal.
- c. Pengisian program yang telah dirancang sebelumnya pada perangkat Arduino Mega 2560.

#### 3.5. Pengujian dan Analisa Data

Setelah semua komponen pada alat sudah terhubung sesuai dengan diagram blok sistem yang telah dirancang dan perangkat lunak untuk mendukung sistem telah dibuat, maka diadakan pengujian dan analisa alat. Pengujian dilakukan secara per blok terlebih dahulu kemudian dilanjutkan secara keseluruhan sistem.

#### 3.6. Pengambilan Kesimpulan

Kesimpulan diambil berdasarkan data yang diperoleh dari pengujian sistem secara keseluruhan. Jika hasil yang didapatkan telah sesuai dengan yang direncanakan sebelumnya, maka proses tersebut telah berhasil memenuhi harapan dan tentunya memerlukan pengembangan lebih lanjut untuk penyempurnaan.



## BAB IV

### PERANCANGAN DAN PEMBUATAN ALAT

Dalam bab ini akan dibahas mengenai perancangan dan pembuatan sistem pengendalian suhu dan kelembaban *brooder* bibit ayam *DOC*. Pada dasarnya perancangan alat meliputi perancangan perangkat keras dan perancangan perangkat lunak.

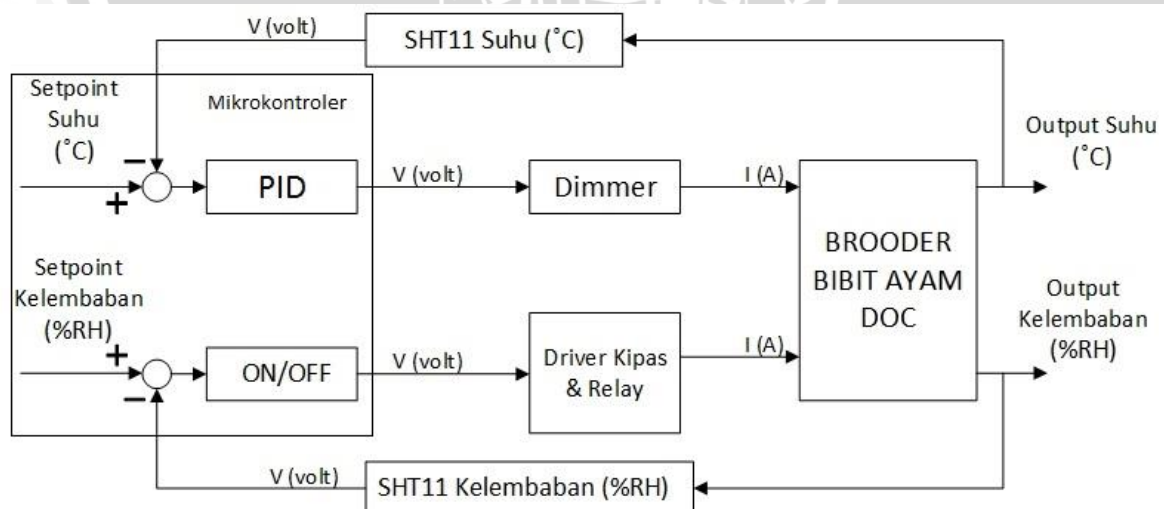
#### 4.1. Perancangan Sistem

Perancangan alat ini dilakukan secara bertahap dalam bentuk blok sehingga akan memudahkan dalam analisis pada setiap bloknnya maupun secara keseluruhan. Perancangan ini terdiri atas:

- a. Perancangan perangkat keras terdiri atas perancangan sensor suhu dan kelembaban SHT11, 2 kipas DC 12V, 2 lampu pijar AC 60W, *humidifier* USB 5V dan rangkaian *dimmer*.
- b. Perancangan perangkat lunak Arduino untuk sensor suhu dan kelembaban dan sinyal PWM untuk driver pemanas, kipas dan relay *humidifier*.

#### 4.2. Diagram Blok Sistem

Secara umum diagram blok dari identifikasi sistem terdiri dari blok pembangkitan sinyal uji, pencatatan sinyal gangguan dan pencatatan sinyal *output*. Pengambilan data sistem dilaksanakan dengan mengondisikan sistem dengan rangkaian *loop* terbuka, seperti ditunjukkan Gambar 4.1



Gambar 4.1. Diagram Blok *Plant Brooder* Bibit Ayam *DOC*

Berdasarkan blok diagram yang ditunjukkan pada Gambar 4.1 dapat dijelaskan sebagai berikut:

- a. Setpoint pada sistem untuk suhu adalah 32°C dan untuk kelembaban adalah 55-60 %RH
- b. Sensor suhu dan kelembaban yang digunakan adalah SHT11 yang berfungsi mendeteksi perubahan suhu dan kelembaban didalam *brooder* bibit ayam *DOC*. Perubahan yang terdeteksi oleh sensor nantinya akan menjadi input untuk mikrokontroler Arduino Mega 2560.
- c. Mikrokontroler yang digunakan adalah Arduino Mega 2560 yang berfungsi untuk memberikan sinyal PWM (Pulse Width Modulation) pada *dimmer*, driver motor DC L289N dan rangkaian *relay* untuk *humidifier*.
- d. Aktuator untuk pengendali suhu menggunakan lampu pijar sebagai elemen pemanas yang dikendalikan menggunakan kontroler PID, kipas sebagai penurun suhu dan *humidifier* sebagai penambah kelembaban yang dikendalikan menggunakan kontroler ON-OFF.

#### 4.3. Spesifikasi Alat

Spesifikasi sistem pengendalian suhu dan kelembaban pada *brooder* bibit ayam *DOC* menggunakan kontroler PID Arduino Mega 2560 adalah sebagai berikut:

1. *Brooder* atau tempat pembibitan ayam *DOC* berupa kayu triplek dengan panjang 80cm, lebar 60cm dan tinggi 80cm.
2. Aktuator yang mempengaruhi suhu dan kelembaban pada *brooder* antara lain: Lampu Pijar AC 60W, Kipas DC 12V dan *humidifier*.
3. Proses penghangat bibit ayam dengan menggunakan 2 buah lampu pijar AC 60W yang dipasang di dalam *brooder*.
4. Proses untuk menjaga aliran udara dengan menggunakan 2 buah Kipas DC 12V yang dipasang pada bagian atas *brooder*.
5. Proses untuk menjaga kelembaban agar stabil menggunakan *humidifier* USB 5V yang diletakkan di dalam *brooder*.
6. Sensor yang digunakan adalah sensor suhu dan kelembaban SHT11.
7. Kriteria *brooder* disesuaikan untuk bibit ayam *DOC* tipe broiler berumur 0-3 hari dengan kebutuhan suhu 32°C dan kelembaban antara 55 – 60 %RH.

#### 4.4. Prinsip Kerja

Prinsip kerja sistem penghangat bibit ayam *DOC* sebagai berikut:

- i. Catu daya yang digunakan ada 6 macam yaitu power supply 12V AC, 5V DC untuk mencatu Arduino Mega 2560, 12V DC untuk mencatu Kipas, 5V DC untuk *humidifier*, 5V untuk mencatu driver motor L298N dan 220V AC untuk mencatu rangkaian *dimmer*.
- j. Sensor SHT11 membaca suhu dan kelembaban pada plant (*brooder* bibit ayam *DOC* kemudian masukan dari sensor suhu akan diproses oleh Arduino Mega 2560 dengan algoritma PID. Kemudian Arduino mengirim output PWM ke rangkaian *dimmer* yang dapat mengatur intensitas cahaya pada lampu pijar yang berfungsi sebagai media pemanas pada *brooder*.
- k. Kipas untuk menurunkan dan mengalirkan udara panas berlebih dan *humidifier* sebagai media penambah kelembaban karena semakin panas suhu pada *brooder* maka kelembaban akan turun, sehingga dibutuhkan alat untuk menambah kelembaban. Kipas dan *humidifier* menggunakan kontroler ON – OFF.

#### 4.5. Perancangan Perangkat Keras

Perancangan perangkat keras terbagi beberapa bagian, diantaranya:

- a. Pemilihan elemen pemanas.
- b. Perancangan rangkaian *dimmer*.
- c. Pemodelan Sensor SHT11.

##### 4.5.1. Pemilihan Pemanas

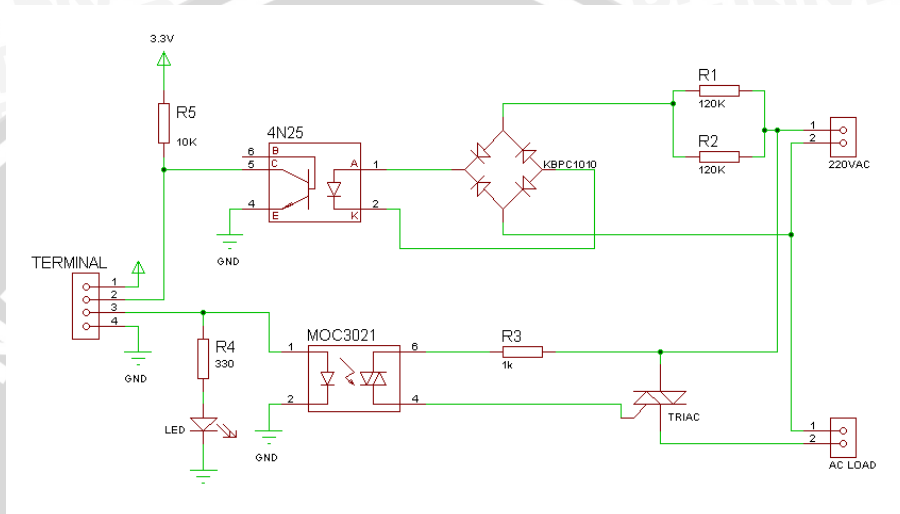
Sistem pemanasan yang digunakan adalah dua (2) lampu pijar AC berdaya 60W untuk setiap lampunya. Pemasangan lampu diletakkan di dalam bagian atas *brooder*, seperti pada Gambar 4.2.



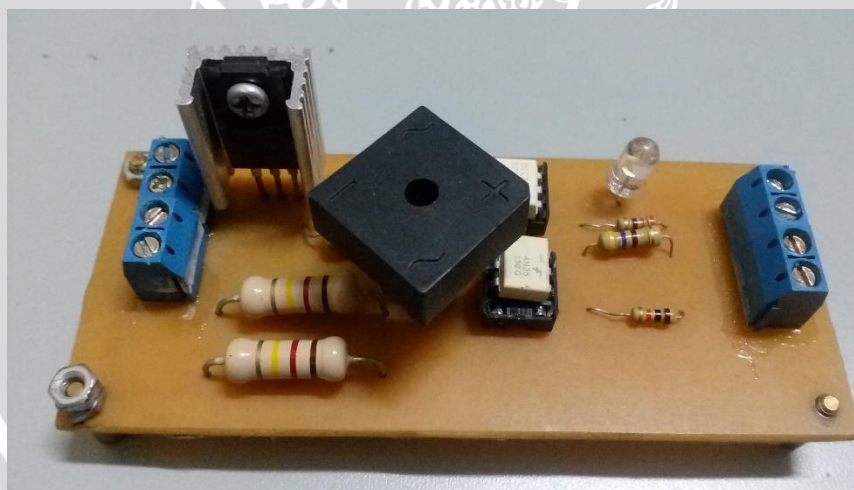
**Gambar 4.2** Pemanas pada *Brooder* Bibit Ayam.

#### 4.5.2. Rangkaian Dimmer

Elemen pemanas yang digunakan adalah elemen pemanas dengan catu daya AC. Untuk mengendalikan besarnya tegangan yang melewati elemen pemanas digunakan rangkaian *dimmer* yang di dalamnya juga terdapat beberapa bagian, diantaranya adalah rangkaian pemicuan *gate* TRIAC dan rangkaian *Zero Cross Detector*. Gambar 4.3 menunjukkan skema rangkaian *dimmer* AC, sedangkan modul rangkaian yang akan dibuat ditunjukkan pada Gambar 4.4.



Gambar 4.3 Skema Rangkaian Dimmer



Gambar 4.4 Modul Rangkaian Dimmer

#### 4.5.3. Pemodelan Sensor SHT11

Jenis sensor yang digunakan untuk mengukur suhu dan kelembaban dalam penelitian ini adalah SHT11. Range pengukuran yang mampu dilakukan sensor SHT11 yaitu  $-40^{\circ}\text{C}$  hingga  $+123,8^{\circ}\text{C}$  atau dari  $-40\text{F}$  hingga  $+4,9\text{F}$  dan

kelembabanrelatif dari 0 %RH hingga 100 %RH. Sensor SHT11 membutuhkan catu daya +5V DC dengan konsumsi daya rendah 30 μW.

Nilai gain dari sensor dapat dihitung dengan Persamaan (4-1): (Seborg dkk,2004:214).

$$K_{SHT} = \frac{\text{range of instrument output}}{\text{range of instrument input}} \dots\dots\dots(4-1)$$

Sedangkan nilai *gain* dari sensor SHT11 suhu adalah:

$$K_{SHT} = \frac{5-0}{123,8-(-40)} = \frac{5}{163.8} = 0,04 \text{ V}/^{\circ}\text{C}$$

Sedangkan nilai *gain* dari sensor SHT11 kelembaban adalah:

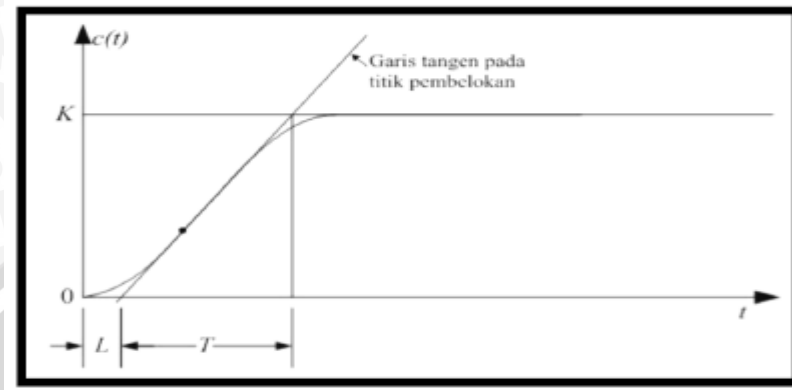
$$K_{SHT} = \frac{5-0}{100-0} = \frac{5}{100} = 0,05 \text{ V}/\%RH$$

**4.6. Perancangan Algoritma Kontroler PID**

Untuk memenuhi tujuan performansi *loop* yang diinginkan, maka perlu ditambahkan kontroler pada sistem tersebut. Kontroler yang dipilih ialah Proporsional Integral Derivatif (PID). Kontroler PID dipilih karena karakteristik plant tersebut membutuhkan kontrol yang cepat dan meminimalkan *overshoot* untuk dapat menghasilkan suhu yang kondusif pada plant *brooder* penghangat bibit ayam DOC.



Ziegler dan Nichols mengemukakan aturan-aturan untuk menentukan nilai dari gain proporsional  $K_p$ , waktu integral  $T_i$ , dan waktu derivatif  $T_d$  berdasarkan karakteristik respon transien dari *plant* yang diberikan. Penentuan parameter kontroler PID atau penalaan kontroler PID tersebut dapat dilakukan dengan bereksperimen dengan plan (Ogata, K.,1997). Pada pengerjaan skripsi ini digunakanlah metode *Ziegler-Nichols* yang pertama sehingga mendapatkan sinyal kurve S seperti pada Gambar 4.5.



**Gambar 4.5** Kurva Respon yang berbentuk S

(Sumber: Katsuhiko Ogata, 1997)

Kurva berbentuk S tersebut dapat dikarakteristikan menjadi dua konstanta yaitu waktu tunda  $L$  dan konstanta waktu  $T$ . Waktu tunda dan konstanta waktu ditentukan dengan menggambar sebuah garis tangen dengan sumbu waktu  $t$  dan sumbu  $c(t) = K$ .

Langkah menentukan *tuning* parameter dengan metode 1 *Ziegler-Nichols* adalah sebagai berikut:

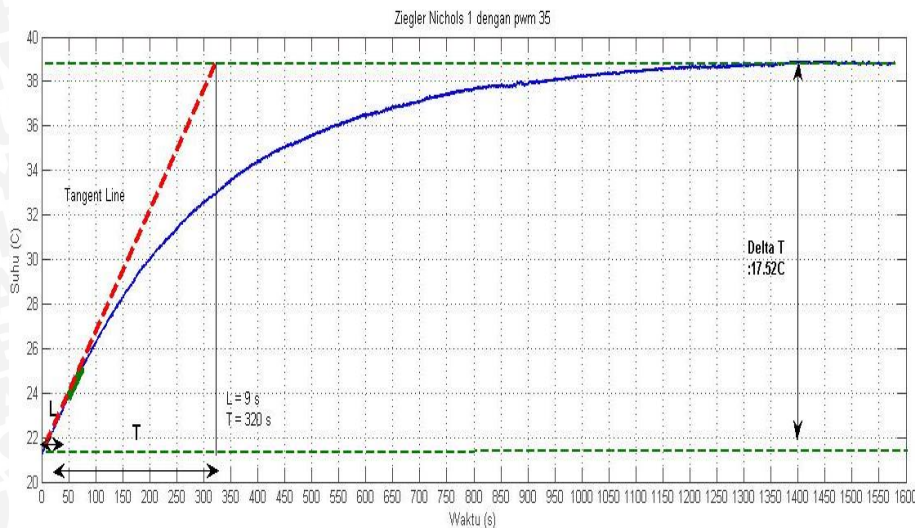
1. Menarik garis *tangent* pada titik infleksi pada grafik karakteristik *plant* seperti pada Gambar 4.5.
2. Menentukan perpotongan garis *tangent* terhadap sumbu waktu  $t$  untuk mendapatkan nilai  $L$ .
3. Menentukan perpotongan garis *tangent* terhadap sumbu *steady* untuk mendapatkan nilai  $T$ .
4. Setelah mendapatkan nilai  $L$  dan  $T$  kemudian menentukan nilai  $K_p$ ,  $T_i$ ,  $T_d$  sesuai ketentuan pada Gambar 4.6

Type Kontrol	$K_p$	$T_i$	$T_d$
P	$\frac{T}{L}$	$\infty$	0
PI	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{T}{L}$	$2L$	$0,5 L$

**Gambar 4.6** Aturan Metode 1 *Ziegler-Nichols*

(Sumber: Katsuhiko Ogata, 1997)

Berikut kurva hasil perancangan kontroler menggunakan metode *Ziegler-Nichols* yang ditunjukkan dalam Gambar 4.7.



**Gambar 4.7** Karakteristik Open Loop Respon PWM 35

Dari grafik *output plant* pada Gambar 4.7 didapatkan nilai:

$$\Delta u = \text{PWM 35} \qquad L = 9 \text{ s}$$

$$\Delta y = 17.52^\circ\text{C} \qquad T = 320 \text{ s}$$

Dari pengujian tegangan pada PWM 35 sebesar 208,7 V, maka:

$$K = \frac{\Delta y}{\Delta u} = \frac{38,8 - 21,28}{208,7} = 0,084^\circ\text{C/V}$$

Setelah mendapatkan parameter yang dibutuhkan maka bias didapatkan fungsi alih *plant* seperti yang ditunjukkan dalam persamaan.

$$G(s) = \frac{Ke^{-Ls}}{Ts + 1} = \frac{0.176e^{-9s}}{320s + 1}$$

Dan diperoleh nilai  $K_p$ ,  $K_i$ ,  $K_d$ :

$$K_p = 1.2 \times \frac{T}{L} = 1.2 \times \frac{320}{9} = 35,56$$

$$K_i = \frac{K_p}{T_i} = \frac{35.56}{2 \times L} = \frac{35,56}{18} = 1,975$$

$$K_d = K_p \times T_d = 35,56 \times (0.5 \times L) = 35,56 \times 4,5 = 160.02$$

Dari perolehan nilai  $K_p$ ,  $K_i$ ,  $K_d$  maka didapatkan persamaan transformasi laplace kontroler PID sebagai berikut:

$$C(s) = \left( K_p + \frac{K_i}{s} + K_d s \right) e(s) \dots\dots\dots(4-1)$$

Persamaan diatas belum bisa dimasukkan kedalam mikrokontroler, maka persamaan kontinyu di atas baru diubah kedalam diskrit melalui Transformasi Z. Dalam Transformasi Z dibutuhkan waktu sampling ( $T_s$ ). Digunakan metode *Backward Difference* sehingga nilai notasi s pada Laplace setara dengan

$$s = \left( \frac{1-z^{-1}}{T_s} \right) \dots\dots\dots(4-2)$$

Bila persamaan (4-2) disubstitusikan ke dalam persamaan (4-1) menjadi

$$C(z) = \left[ Kp + \frac{Ki}{\frac{(1-z^{-1})}{T_s}} + Kd \frac{(1-z^{-1})}{T_s} \right] E(z)$$

$$= \left[ Kp + \frac{Ki \times T_s}{(1-z^{-1})} + \frac{Kd}{T_s} (1-z^{-1}) \right] E(z)$$

Kontroler Proporsional:

$$Cp(z) = KpE(z)$$

Kontroler Integral:

$$Ci(z) = \left( \frac{KiT_s}{1-z^{-1}} \right) xE(z)$$

$$Ci(z) - Ci(z)z^{-1} = KiT_sxE(z)$$

$$Ci(z) = Ci(z)z^{-1} + KiT_sxE(z)$$

Kontroler Diferensial:

$$Cd(z) = \frac{Kd}{T_s} (z)(1-z^{-1})xE(z)$$

$$Cd(z) = \frac{Kd}{T_s} [E(z) - E(z)z^{-1}]$$

Dari persamaan diatas kemudian diubah kedalam persamaan beda sehingga didapatkan persamaan dibawah ini:

Kontroler Proporsional :  $Cp(k) = Kp \times E(k)$

Kontroler Integral :  $Ci(k) = Ci(k - 1) + KiT_s \times E(k)$





Kontroler Diferensial :  $Cd(k) = \frac{Kd}{Ts} [E(k) - E(k - 1)]$

Jika digabungkan menjadi:

$$C(k) = [Kp \times E(k)] + [Ci(k - 1) + KiTs \times E(k)] + \left[ \frac{Kd}{Ts} [E(k) - E(k - 1)] \right]$$

Dimana (k-1) adalah kondisi sebelumnya. Persamaan di atas telah berupa persamaan diskrit sehingga dapat dimasukkan ke dalam mikrokontroler.

#### 4.7. Modul Arduino Mega 2560

Pada sistem pengendalian suhu ini digunakan Arduino Mega 2560 sebagai pengolah data dalam proses pengendalian elemen pemanas. Arduino Mega adalah *board* mikrokontroler berbasis ATmega1280. Memiliki 54 pin *input* dan *output* digital, di mana 14 pin tersebut dapat digunakan sebagai *output* PWM dan 16 pin *input* analog, 16 MHz osilator kristal, koneksi USB, *jack power*, ICSP header, dan tombol reset. Untuk dapat digunakan, mikrokontroler ini dihubungkan ke komputer dengan menggunakan kabel USB atau dengan adaptor AC-DC untuk menjalankannya.

Secara fisik, ukuran Arduino Mega hampir kurang lebih 2 kali lebih besar dari Arduino Uno, ini untuk mengakomodasi lebih banyaknya pin Digital dan Analog pada *board* Arduino Mega tersebut. Tampilan Arduino Mega dapat dilihat pada Gambar 4.8. Sedangkan penggunaan pin pada Arduino Mega disebutkan pada Tabel 4.1



Gambar 4.8 Modul Arduino Mega 2560

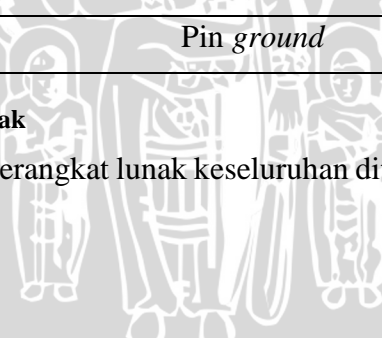
Tabel 4.1 Penggunaan Pin Pada Arduino Mega

No.	Pin	Fungsi
1.	2	Jalur masukan <i>Zero Cross Interrupt</i>

2.	4	Jalur keluaran untuk rangkaian <i>dimmer</i>
3.	8	SHT11
4.	9	SHT11
5.	10	SHT11
6.	11	<i>Driver kipas</i>
7.	12	<i>Driver kipas</i>
8.	13	<i>Enable driver kipas</i>
9.	31	<i>Relay humidifier</i>
10.	22	LCD pin
11.	24	LCD pin
12.	26	LCD pin
13.	28	LCD pin
14.	30	LCD pin
15.	32	LCD pin
16.	34	LCD pin
17.	3,3V	<i>Vcc dimmer</i>
18.	5V	<i>Vcc SHT11, driver kipas, humidifier, LCD</i>
19.	GND	<i>Pin ground</i>

**4.8. Perancangan Perangkat Lunak**

*Flowchart* perancangan perangkat lunak keseluruhan ditunjukkan pada Gambar 4.9





Gambar 4.9 Flowchart Program Keseluruhan

## BAB V

### PENGUJIAN DAN ANALISIS SISTEM

Tujuan pengujian sistem ini adalah untuk menentukan apakah alat yang telah dibuat berfungsi dengan baik dan sesuai dengan perancangan. Pengujian pada sistem ini meliputi pengujian setiap blok maupun pengujian secara keseluruhan. Pengujian setiap blok ini dilakukan untuk menemukan letak kesalahan dan mempermudah analisis pada sistem apabila alat tidak bekerja sesuai dengan perancangan. Pengujian pada sistem ini dibagi menjadi lima bagian, yaitu:

1. Pengujian sensor suhu dan kelembaban SHT11
2. Pengujian penyulutan TRIAC
3. Pengujian rangkaian *dimmer*
4. Pengujian sistem keseluruhan

#### 5.1. Pengujian Sensor Suhu dan Kelembaban SHT11

Pengujian sensor suhu ditujukan untuk mengetahui kemampuan sensor SHT11 dalam membaca perubahan suhu.

##### 5.1.1. Peralatan yang Digunakan

1. Sumber tegangan 5V DC.
2. Arduino Mega 2560.
3. Rangkaian Sensor Suhu dan Kelembaban SHT11.
4. 2 buah lampu pijar AC 60 W sebagai pemanas (*heater*).
5. Komputer.

##### 5.1.2. Prosedur Pengujian

1. Merangkai peralatan seperti blok diagram pada Gambar 5.1.
2. Mengunggah program pembacaan sensor suhu pada arduino.
3. Mengamati dan mencatat nilai pada tampilan serial monitor.



**Gambar 5.1** Blok Pengujian Sensor SHT11 untuk Suhu**5.1.3. Hasil Pengujian**

Hasil pengujian pembacaan suhu yang ditampilkan pada serial monitor dan dibandingkan pada pembacaan suhu dari termometer ditunjukkan pada Tabel 5.1.

**Tabel 5.1** Hasil Pengujian Sensor Suhu SHT11

No.	Suhu Pembacaan Termometer (°C)	Suhu Pembacaan Serial Monitor (°C)
1.	25	26,33
2.	26	26,87
3.	27	27,65
4.	28	28,22
5.	29	29,42
6.	30	29,75
7.	31	30,65
8.	32	31,49
9.	33	32,53
10.	34	34,65
11.	35	36,31
12.	36	36,63
13.	37	37,24
14.	38	37,59

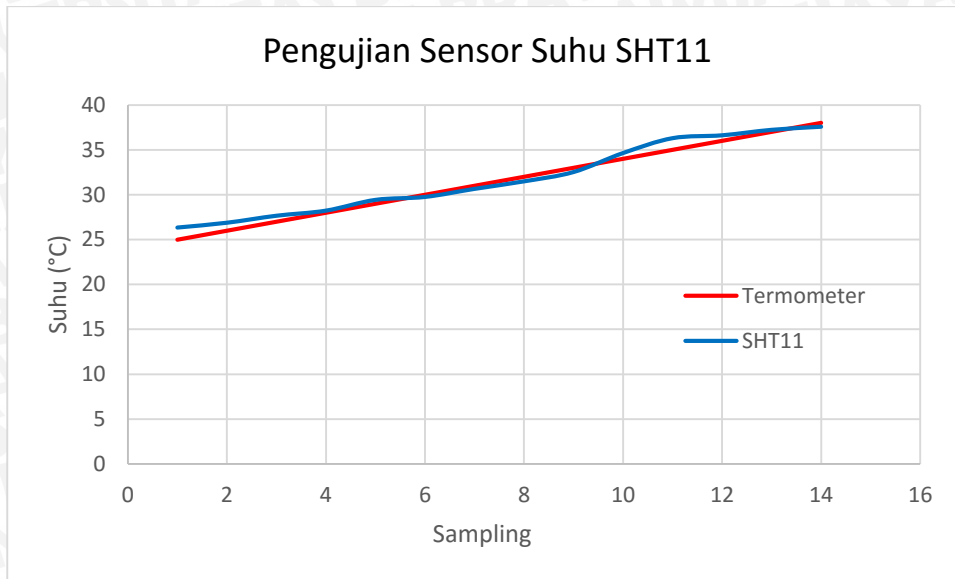
Perhitungan kesalahan pembacaan sensor:

$$\% \text{ kesalahan} = \frac{(\sum \text{Hasil Pengukuran} - \sum \text{Referensi})}{\sum \text{Referensi}} \times 100\%$$

$$\% \text{ kesalahan} = \frac{(445,33 - 441)}{441} \times 100\% = 0,98\%$$

Dapat disimpulkan bahwa sensor SHT11 untuk pembacaan suhu dapat bekerja dengan baik.

Grafik hubungan antara suhu sensor SHT11 dengan thermometer digital dapat dilihat pada Gambar 5.3



**Gambar 5.2** Perbandingan Suhu Sensor SHT11 terhadap Termometer Digital

**5.2. Pengujian Sensor Kelembaban SHT11**

Pengujian sensor SHT11 ditujukan untuk mengetahui apakah sensor dapat membaca kelembaban dengan baik.

**5.2.1. Peralatan yang Digunakan**

1. Sumber tegangan 5V DC.
2. Arduino Mega 2560.
3. *Humidifier* USB 5V DC
4. Rangkaian *relay*.
5. Komputer.

**5.2.2. Prosedur Pengujian**

1. Merangkai peralatan seperti blok diagram pada Gambar 5.3.
2. Mengunggah program pembacaan sensor kelembaban pada arduino.
3. Mengamati dan mencatat nilai pada tampilan serial monitor.



**Gambar 5.3** Blok Pengujian Sensor SHT11 untuk Kelembaban



### 5.2.3. Hasil Pengujian

Hasil pengujian pembacaan kelembaban yang ditampilkan pada serial monitor dan dibandingkan dengan pembacaan dari hygrometer ditunjukkan pada Tabel 5.2

**Tabel 5.2** Hasil Pengujian Sensor Kelembaban SHT11

No.	Kelembaban pada Hygrometer Digital (%RH)	Kelembaban pada Serial Monitor (%RH)
1	52	53,31
2	54	54,56
3	56	55,47
4	58	58,48
5	60	61,33
6	62	62,67
7	64	63,71
8	66	67,36
9	68	69,12
10	69	69,68
11	70	71,24

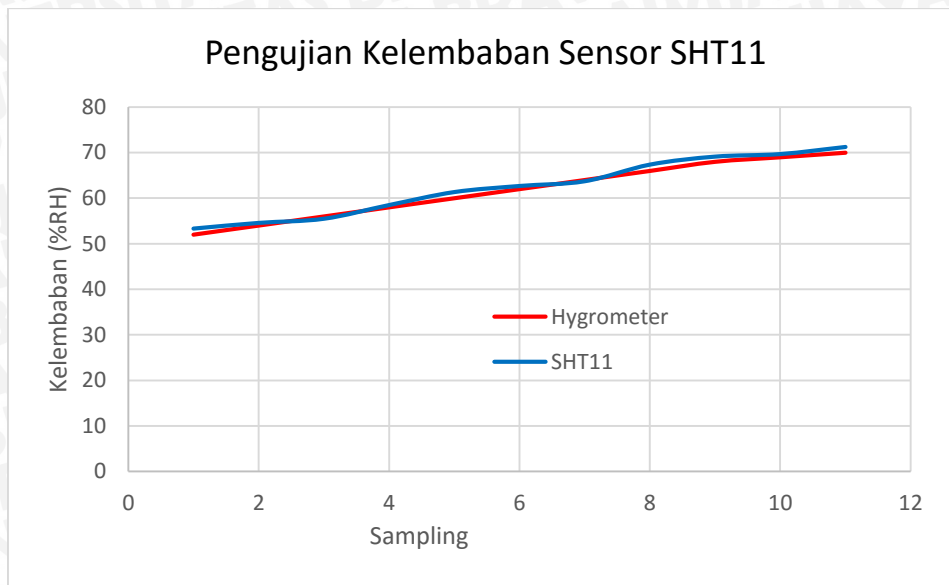
Perhitungan kesalahan pembacaan sensor:

$$\% \text{ kesalahan} = \frac{(\sum \text{Hasil Pengukuran} - \sum \text{Referensi})}{\sum \text{Referensi}} \times 100\%$$

$$\% \text{ kesalahan} = \frac{(686,93 - 679)}{679} \times 100\% = 1,17\%$$

Dapat disimpulkan bahwa sensor SHT11 untuk pembacaan kelembaban dapat bekerja dengan baik.

Grafik hubungan antara suhu sensor SHT11 dengan hygrometer digital dapat dilihat pada Gambar 5.4



**Gambar 5.4** Grafik Pengujian Sensor Kelembaban SHT11

### 5.3. Pengujian Penyulutan TRIAC

Pengujian ini dilakukan untuk mengetahui apakah rangkaian *dimmer* yang telah dibuat bisa mengatur gelombang sinusoida tegangan AC.

#### 5.3.1. Peralatan yang Digunakan

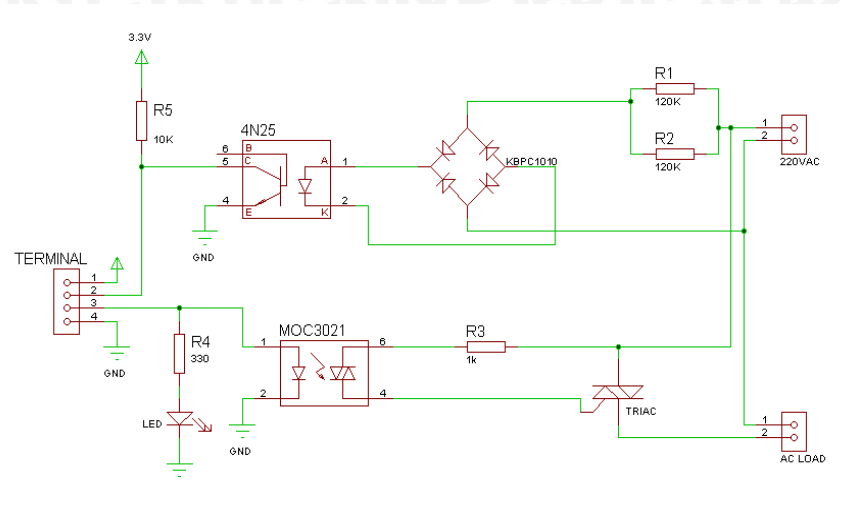
1. Sumber Tegangan 220 VAC.
2. Arduino Mega 2560.
3. Rangkaian *Zero Cross Detector*.
4. Rangkaian TRIAC dan MOC3021.
5. Oscilloscope.
6. Multimeter.
7. Komputer.

#### 5.3.2. Prosedur Pengujian

1. Membuat rangkaian seperti Gambar 5.5.
2. Menghubungkan sumber tegangan 5V ke Arduino dan ke optocoupler 4N25 (terminal 1).
3. Menghubungkan rangkaian ke sumber tegangan 220 VAC.
4. Menghubungkan keluaran optocoupler 4N25 ke pin *interrupt* Arduino (terminal 2).



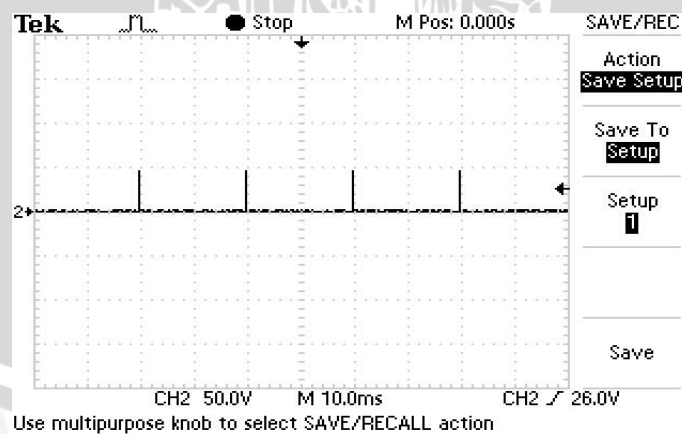
5. Menghubungkan salah satu pin digital I/O dari arduino ke MOC3021 (terminal 3).
6. Menghubungkan *probe 1* oscilloscope ke keluaran Arduino (pin digital I/O) dan *probe 2* ke tegangan keluaran TRIAC (AC Load).



**Gambar 5.5** Rangkaian Pengujian Sudut Penyalan TRIAC

### 5.3.3. Hasil Pengujian

Dari pengujian penyalan TRIAC didapatkan hasil pengujian berupa bentuk gelombang keluaran TRIAC sesuai dengan sudut penyalannya. Contoh hasil gelombang keluaran arduino dan triac sudut penyalan  $45^\circ$  atau PWM 35 ditunjukkan pada Gambar 5.6 dan Gambar 5.7.



**Gambar 5.6** Bentuk Gelombang Keluaran Arduino



**Gambar 5.7** Bentuk gelombang Keluaran TRIAC dengan PWM 32

**Tabel 5.3** Hasil Pengujian Sudut Penyalan TRIAC

PWM	Sudut Penyalan (°)	Perhitungan waktu tunda (ms)	Pengujian waktu tunda (ms)
16	22,5	1,25	1,1
32	45	2,5	2,4
48	67,5	3,75	3,5
64	90	5	4,8
80	112,5	6,25	6
96	135	7,5	7,2
112	157,5	8,75	8,4
128	180	10	9,8

Berdasarkan data hasil pengujian sudut penyalan TRIAC yang didapatkan pada Tabel 5.3 terjadi penyimpangan waktu tunda dengan data hasil perhitungan.

- Perhitungan pengujian data ke-1:

$$\begin{aligned}
 \text{Waktu tunda (ms)} &= \frac{\text{Sudut Penyalan}}{180^\circ} \times 10 \\
 &= \frac{22,5^\circ}{180^\circ} \times 10 \\
 &= 1,25 \text{ ms}
 \end{aligned}$$

Nilai kesalahan yang didapat dari selisih absolut perhitungan dan pengujian sebesar:

$$\begin{aligned}
 \text{Kesalahan (ms)} &= |\text{Perhitungan} - \text{Pengujian}| \\
 &= |1,25 - 1,1| = 0,15
 \end{aligned}$$

$$\begin{aligned}\text{Kesalahan (\%)} &= \frac{\text{Kesalahan}}{\text{Data Perhitungan}} \times 100\% \\ &= \frac{0,15}{1,25} \times 100\% \\ &= 12\%\end{aligned}$$

- Perhitungan pengujian data ke-2:

$$\begin{aligned}\text{Waktu tunda (ms)} &= \frac{\text{Sudut Penyalaan}}{180^\circ} \times 10 \\ &= \frac{45^\circ}{180^\circ} \times 10 \\ &= 2,5 \text{ ms}\end{aligned}$$

Nilai kesalahan yang didapat dari selisih absolut perhitungan dan pengujian sebesar:

$$\begin{aligned}\text{Kesalahan (ms)} &= |\text{Perhitungan} - \text{Pengujian}| \\ &= |2,5 - 2,4| = 0,1\end{aligned}$$

$$\begin{aligned}\text{Kesalahan (\%)} &= \frac{\text{Kesalahan}}{\text{Data Perhitungan}} \times 100\% \\ &= \frac{0,1}{2,5} \times 100\% \\ &= 4\%\end{aligned}$$

- Perhitungan pengujian data ke-3:

$$\begin{aligned}\text{Waktu tunda (ms)} &= \frac{\text{Sudut Penyalaan}}{180^\circ} \times 10 \\ &= \frac{67,5^\circ}{180^\circ} \times 10 \\ &= 3,75 \text{ ms}\end{aligned}$$

Nilai kesalahan yang didapat dari selisih absolut perhitungan dan pengujian sebesar:

$$\begin{aligned}\text{Kesalahan (ms)} &= |\text{Perhitungan} - \text{Pengujian}| \\ &= |3,75 - 3,5| = 0,25\end{aligned}$$

$$\begin{aligned}\text{Kesalahan (\%)} &= \frac{\text{Kesalahan}}{\text{Data Perhitungan}} \times 100\% \\ &= \frac{0,25}{3,75} \times 100\% \\ &= 6,67\%\end{aligned}$$

- Perhitungan pengujian data ke-4:

$$\begin{aligned}\text{Waktu tunda (ms)} &= \frac{\text{Sudut Penyalaan}}{180^\circ} \times 10 \\ &= \frac{90^\circ}{180^\circ} \times 10 \\ &= 5 \text{ ms}\end{aligned}$$

Nilai kesalahan yang didapat dari selisih absolut perhitungan dan pengujian sebesar:

$$\begin{aligned}\text{Kesalahan (ms)} &= |\text{Perhitungan} - \text{Pengujian}| \\ &= |5 - 4,8| = 0,2\end{aligned}$$

$$\begin{aligned}\text{Kesalahan (\%)} &= \frac{\text{Kesalahan}}{\text{Data Perhitungan}} \times 100\% \\ &= \frac{0,2}{5} \times 100\% \\ &= 4\%\end{aligned}$$

- Perhitungan pengujian data ke-5:

$$\begin{aligned}\text{Waktu tunda (ms)} &= \frac{\text{Sudut Penyalaan}}{180^\circ} \times 10 \\ &= \frac{112,5^\circ}{180^\circ} \times 10 \\ &= 6,25 \text{ ms}\end{aligned}$$

Nilai kesalahan yang didapat dari selisih absolut perhitungan dan pengujian sebesar:

$$\begin{aligned}\text{Kesalahan (ms)} &= |\text{Perhitungan} - \text{Pengujian}| \\ &= |6,25 - 6| = 0,25\end{aligned}$$

$$\begin{aligned}\text{Kesalahan (\%)} &= \frac{\text{Kesalahan}}{\text{Data Perhitungan}} \times 100\% \\ &= \frac{0,25}{6,25} \times 100\% \\ &= 4,00\%\end{aligned}$$

- Perhitungan pengujian data ke-6:

$$\begin{aligned}\text{Waktu tunda (ms)} &= \frac{\text{Sudut Penyalaan}}{180^\circ} \times 10 \\ &= \frac{135^\circ}{180^\circ} \times 10 \\ &= 7,5 \text{ ms}\end{aligned}$$

Nilai kesalahan yang didapat dari selisih absolut perhitungan dan pengujian sebesar:

$$\begin{aligned}\text{Kesalahan (ms)} &= |\text{Perhitungan} - \text{Pengujian}| \\ &= |7,5 - 7,2| = 0,3\end{aligned}$$

$$\begin{aligned}\text{Kesalahan (\%)} &= \frac{\text{Kesalahan}}{\text{Data Perhitungan}} \times 100\% \\ &= \frac{0,3}{7,5} \times 100\% \\ &= 4,00\%\end{aligned}$$

- Perhitungan pengujian data ke-7:

$$\begin{aligned}\text{Waktu tunda (ms)} &= \frac{\text{Sudut Penyalaan}}{180^\circ} \times 10 \\ &= \frac{157,5^\circ}{180^\circ} \times 10 \\ &= 8,75 \text{ ms}\end{aligned}$$

Nilai kesalahan yang didapat dari selisih absolut perhitungan dan pengujian sebesar:

$$\begin{aligned}\text{Kesalahan (ms)} &= |\text{Perhitungan} - \text{Pengujian}| \\ &= |8,75 - 8,4| = 0,35\end{aligned}$$

$$\begin{aligned}\text{Kesalahan (\%)} &= \frac{\text{Kesalahan}}{\text{Data Perhitungan}} \times 100\% \\ &= \frac{0,35}{8,75} \times 100\% \\ &= 4,00\%\end{aligned}$$

- Perhitungan pengujian data ke-8:

$$\begin{aligned}\text{Waktu tunda (ms)} &= \frac{\text{Sudut Penyalaan}}{180^\circ} \times 10 \\ &= \frac{180^\circ}{180^\circ} \times 10 \\ &= 10 \text{ ms}\end{aligned}$$

Nilai kesalahan yang didapat dari selisih absolut perhitungan dan pengujian sebesar:

$$\begin{aligned}\text{Kesalahan (ms)} &= |\text{Perhitungan} - \text{Pengujian}| \\ &= |10 - 9,8| = 0,4\end{aligned}$$

$$\begin{aligned}\text{Kesalahan (\%)} &= \frac{\text{Kesalahan}}{\text{Data Perhitungan}} \times 100\% \\ &= \frac{0,4}{10} \times 100\% \\ &= 4\%\end{aligned}$$

Hasil perhitungan kesalahan pengujian secara keseluruhan dapat dilihat pada Tabel 5.4

**Tabel 5.4** Kesalahan Pengujian PWM Penyalan TRIAC

PWM	Perhitungan waktu tunda (ms)	Pengujian waktu tunda (ms)	Kesalahan (ms)	Kesalahan (%)
16	1,25	1,1	0,15	12,00
32	2,5	2,4	0,1	4,00
48	3,75	3,5	0,25	6,67
64	5	4,8	0,2	4,00
80	6,25	6	0,25	4,00
96	7,5	7,2	0,3	4,00
112	8,75	8,4	0,35	4,00
128	10	9,8	0,2	2,00
Rata Rata Kesalahan (%)				5,084

Dari hasil perhitungan kesalahan pengujian secara keseluruhan didapatkan nilai rata-rata kesalahan keseluruhan sebesar 5,084%. Kesalahan ini dikarenakan adanya proses perhitungan dan pembulatan angka oleh mikrokontroler dan sumber AC yang tidak ideal baik dari frekuensi dan bentuk gelombang tegangan, sehingga menyebabkan adanya selisih antara perhitungan dan pengukuran. Berdasarkan analisis di atas bisa disimpulkan bahwa mikrokontroler mampu menghasilkan sudut penyalan yang sesuai dengan yang diberikan.

#### 5.4. Pengujian Rangkaian *Dimmer*

Pengujian ini dilakukan untuk mengetahui apakah rangkaian *dimmer* dapat digunakan sebagai *driver* untuk lampu pijar.

##### 5.4.1. Peralatan yang Digunakan

1. Sumber Tegangan AC.
2. Arduino Mega 2560.
3. Rangkaian *Dimmer*.
4. Multimeter.
5. Komputer.

**5.4.2. Prosedur Pengujian**

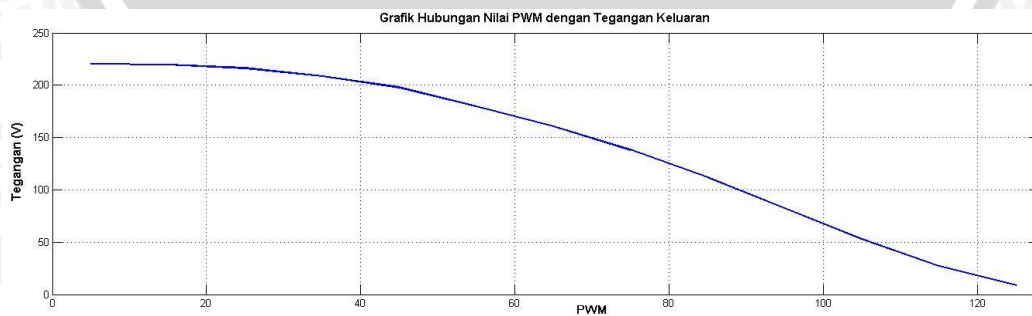
1. Menghubungkan rangkaian *dimmer* dengan arduino.
2. Menghubungkan rangkaian *dimmer* dengan sumber AC.
3. Menghubungkan Arduino dengan komputer.
4. Memberikan beban lampu pada rangkaian *dimmer*.
5. Memberikan nilai PWM pada rangkaian *dimmer* melalui arduino.
6. Mengukur tegangan keluaran rangkaian *dimmer* kepada beban.

**5.4.3. Hasil Pengujian**

Dari hasil pengujian rangkaian *dimmer* didapatkan hasil berupa nilai tegangan yang berbeda-beda sesuai dengan nilai PWM yang diberikan. Hasil pengujian dapat dilihat pada Tabel 5.5

**Tabel 5.5** Perbandingan Nilai PWM Terhadap Tegangan

No.	PWM	Tegangan (V)
1	125	8,92
2	115	27,43
3	105	52,8
4	95	82,5
5	85	111,8
6	75	138,4
7	65	161,1
8	55	180
9	45	198,2
10	35	208,7
11	25	216,5
12	15	220,0
13	5	220,7



**Gambar 5.8.** Grafik Hubungan Nilai PWM dengan Tegangan

Dari Tabel 5.5 dapat disimpulkan bahwa rangkaian *dimmer* dapat berfungsi dengan baik dan dapat digunakan sebagai *driver* aktuator (lampu pijar), ditandai dengan adanya perubahan tegangan akibat perubahan nilai PWM yang diberikan.

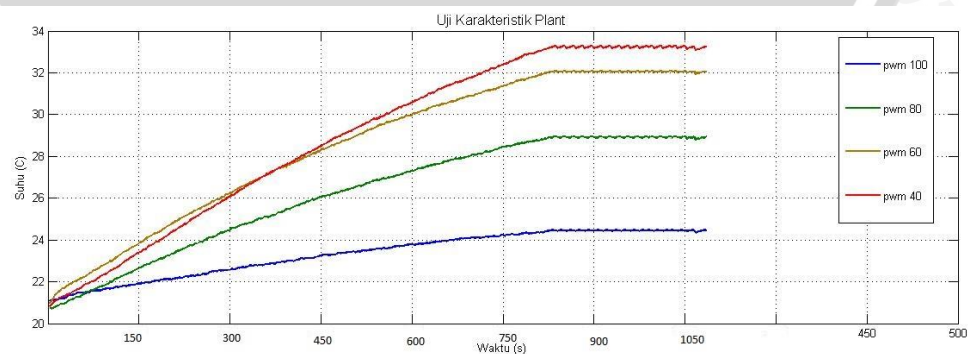
Dari hasil pengujian rangkaian *dimmer* didapatkan juga hasil nilai berupa suhu *steadystate* yang berbeda-beda sesuai dengan nilai PWM yang diberikan. Hasil pengujian dapat dilihat pada Tabel 5.6

**Tabel 5.6** Perbandingan Nilai PWM Terhadap *steadystate* Suhu

No.	PWM	Suhu <i>Steadystate</i> (°C)
1	40	33,27
2	60	32,06
3	80	28,94
4	100	24,45

Dari Tabel 5.8 dapat disimpulkan bahwa rangkaian *dimmer* dapat digunakan sebagai *driver* aktuator (lampu pijar), ditandai dengan adanya perubahan suhu *steadystate* akibat perubahan nilai PWM yang diberikan.

Dari hasil pengujian rangkaian *dimmer* didapatkan hasil nilai karakteristik grafik respon plant *brooder* bibit ayam *DOC*. Hasil pengujian dapat dilihat pada Gambar 5.9.



**Gambar 5.9** Karakteristik Plant



Dari Gambar 5.9 dapat disimpulkan bahwa grafik karakteristik respon pada rangkaian *dimmer* dapat digunakan sebagai *driver* aktuator pemanas, ditandai dengan adanya perubahan suhu *steadystate* akibat perubahan nilai PWM yang diberikan.

### 5.5. Pengujian Keseluruhan Sistem

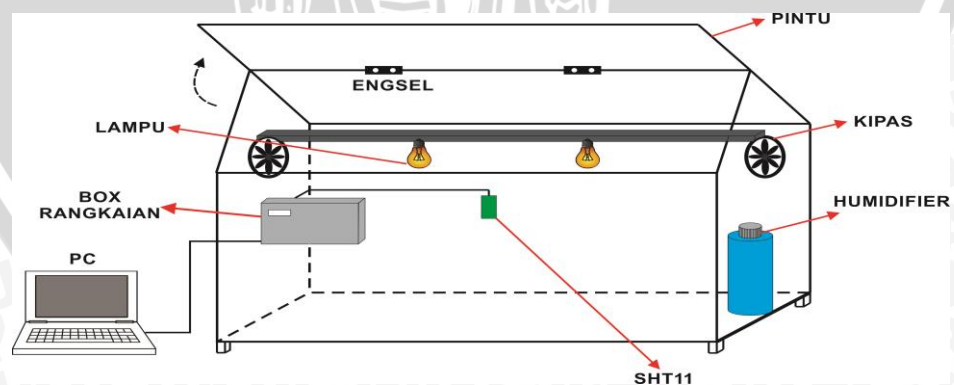
Pengujian ini dilakukan untuk mengetahui seberapa besar nilai kontroler yang dibutuhkan agar sistem bekerja sesuai dengan setpoint 32°C dan kelembaban 55 – 60 %RH.

#### 5.5.1. Peralatan yang Digunakan

1. *Plant brooder* tempat penghangat bibit ayam.
2. Sensor suhu dan kelembaban SHT11.
3. Rangkaian *Dimmer*.
4. *Humidifier* 5V
5. Kipas DC 12V
6. Lampu Pijar AC 60W
7. Arduino Mega 2560.
8. Sumber tegangan AC.
9. Komputer.

#### 5.5.2. Prosedur Pengujian

1. Menghubungkan rangkaian seperti Gambar 5.10



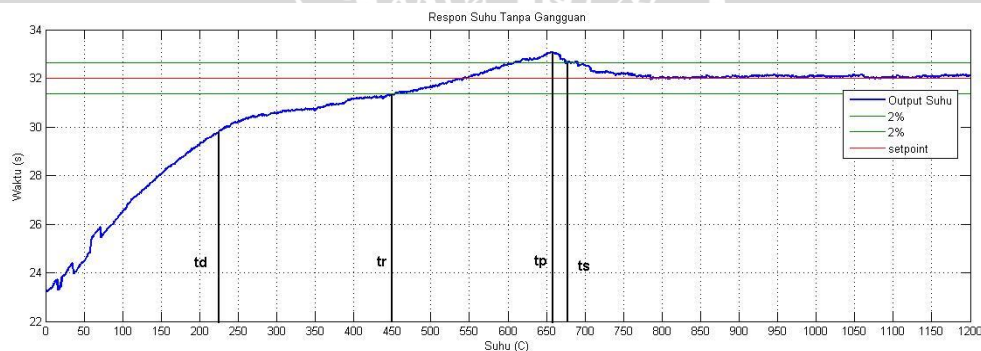
**Gambar 5.10** Sistem Keseluruhan

2. Menghubungkan power supply sebagai sumber catu untuk Arduino Mega 2560 dan komponen lainnya yang membutuhkan catu daya.

3. Mengunduh program dengan kontroler PID sesuai parameter yang telah didapat melalui software Arduino 1.6.5.
4. Setelah memastikan semua rangkaian terpasang dengan benar, program dijalankan.
5. Mengamati dan menganalisa hasil dari kinerja sistem.
6. Pada pengujian pertama kinerja sistem tidak diberi gangguan. Pada pengujian kedua kinerja sistem diberi gangguan dengan cara penutup *brooder* dibuka dan pemanas dimatikan sehingga dapat merubah suhu di dalamnya selama 5 menit.
7. Membuat grafik dari data yang didapat dari hasil pengujian.

### 5.5.3. Hasil Pengujian

Pengujian untuk pengendalian suhu dengan menggunakan kontroler PID dengan  $K_p= 35,56$   $K_i=1,975$  dan  $K_d=160,02$  dan *setpoint* suhu  $32^\circ\text{C}$  didapatkan hasil respon yang ditunjukkan pada Gambar 5.11



**Gambar 5.11** Grafik Respon Sistem Suhu dengan Kontroler PID

Dari grafik hasil pengujian untuk pengendalian suhu dengan suhu awal  $23,21^\circ\text{C}$  dengan menggunakan kontroler PID dengan parameter  $K_p= 35,56$   $K_i=1,975$  dan  $K_d=160,02$  diperoleh data sebagai berikut:

**Tabel 5.7** Data Respon Suhu Menggunakan Kontroler PID

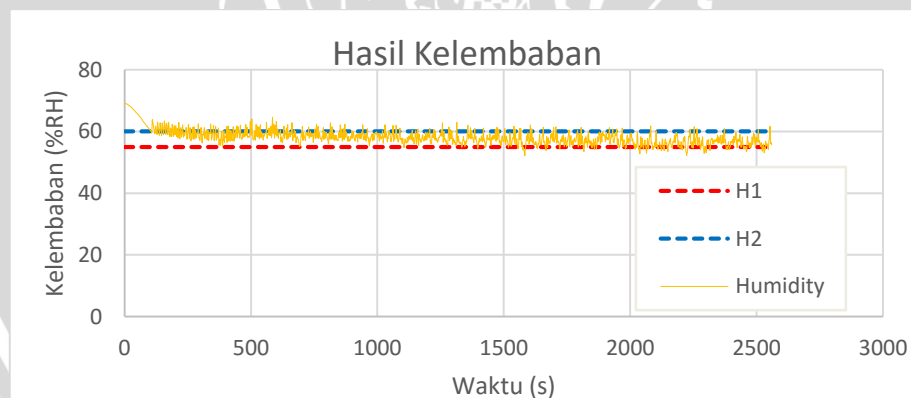
Waktu Tunda (td)	225 detik
Waktu Naik (tr)	450 detik
Waktu Puncak (tp)	658 detik
Waktu Keadaan Mantap (ts)	680 detik

Maksimum <i>Overshoot</i> ( $M_p$ )	3,09 %
<i>Error Steady State</i>	0,23 %

Keterangan :

- Waktu Tunda ( $t_d$ ) adalah waktu yang diperlukan respons untuk mencapai setengah nilai akhir untuk waktu yang pertama
- Waktu Naik ( $t_r$ ) adalah waktu yang diperlukan respons untuk naik dari 0-100% dari nilai akhirnya.
- Waktu Puncak ( $t_p$ ) adalah waktu yang diperlukan respon untuk mencapai nilai puncak respon.
- Waktu Keadaan Mantap ( $t_s$ ) adalah waktu yang dibutuhkan respons untuk mencapai keadaan *steady state*.
- Maksimum Puncak ( $M_p$ ) adalah *overshoot* maksimum yang terjadi saat waktu puncak ( $t_p$ ).

Pada pengujian untuk pengendalian kelembaban dengan menggunakan kontroler ON-OFF dan *range setpoint* kelembaban sebesar 55 – 60 %RH didapatkan hasil respon sistem yang ditunjukkan pada Gambar 5.12.

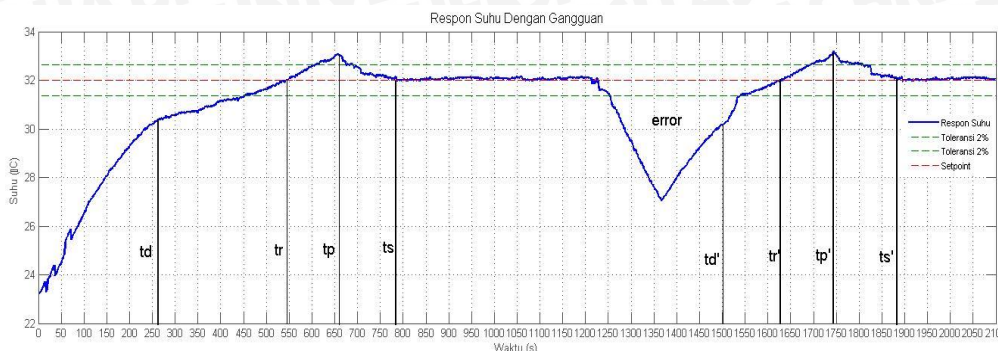


**Gambar 5.12** Grafik Respon Sistem Kelembaban ON – OFF

Tanggapan atau respon kelembaban menggunakan kontroler ON-OFF dengan *range setpoint* 55 – 60%RH telah berada pada *range setpoint*.

Pada pengujian sistem pengendalian suhu dengan gangguan berupa penutup pada *brooder* dibuka dan pemanas dimatikan selama 5 menit. Pemberian gangguan bertujuan untuk mengetahui apakah sistem dapat mempertahankan suhu dan kelembaban udara di dalam *brooder*.

Pengujian sistem suhu dilakukan dengan menggunakan kontroler PID dengan  $K_p= 35,56$   $K_i= 1,975$   $K_d=160,02$  dan *setpoint*  $32^\circ\text{C}$  dengan suhu awal  $23,21^\circ\text{C}$  didapatkan hasil respon yang ditunjukkan pada Gambar 5.13.



**Gambar 5.13** Respon Sistem Suhu dengan Gangguan menggunakan Kontroler PID

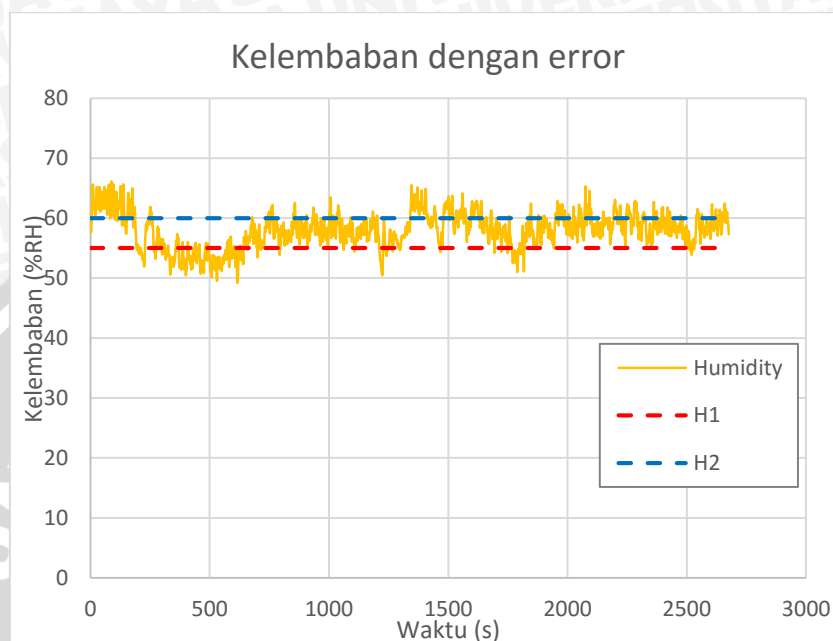
**Tabel 5.8** Data Respon Suhu dengan Gangguan Menggunakan Kontroler PID

Waktu Tunda ( $t_d$ )	130 detik
Waktu Naik ( $t_r$ )	260 detik
Waktu Puncak ( $t_p$ )	370 detik
Waktu Keadaan Mantap ( $t_s$ )	504 detik
Maksimum <i>Overshoot</i> ( $M_p$ )	3,25 %
<i>Error Steady State</i>	0,6 %

**Keterangan:**

- Waktu Tunda ( $t_d$ ) adalah waktu yang diperlukan respons untuk mencapai setengah nilai akhir untuk waktu yang pertama.
- Waktu Naik ( $t_r$ ) adalah waktu yang diperlukan respons untuk naik dari 0-100% dari nilai akhirnya.
- Waktu Puncak ( $t_p$ ) adalah waktu yang diperlukan respon untuk mencapai nilai puncak respon.
- Waktu Keadaan Mantap ( $t_s$ ) adalah waktu yang dibutuhkan respons untuk mencapai keadaan *steady state*.
- Maksimum Puncak ( $M_p$ ) adalah *overshoot* maksimum yang terjadi saat waktu puncak ( $t_p$ ).

Pengujian sistem kelembaban menggunakan kontroler ON-OFF dan *range setpoint* kelembaban sebesar 55-60%RH dengan diberi gangguan berupa penutup *brooder* dibuka dan pemanas dimatikan selama 5 menit. Didapatkan hasil respon sistem sistem yang ditunjukkan pada Gambar 5.14.



**Gambar 5.14** Respon Sistem Kelembaban dengan Gangguan Menggunakan Kontroler ON-OFF

## BAB VI PENUTUP

### 6.1. Kesimpulan

Berdasarkan pengujian yang telah dilakukan, baik pengujian perblok rangkaian maupun pengujian sistem secara keseluruhan diperoleh kesimpulan sebagai berikut:

1. Berdasarkan data respon sistem *open loop* yang diperoleh dari pengujian menggunakan metode Ziegler – Nichols 1, maka parameter kontroler PID didapatkan sebagai berikut:

Kp	Ki	Kd
35,56	1,975	160,02

2. Pada pengujian tanpa gangguan tanggapan atau respon pengendalian suhu menggunakan kontroler PID dengan *setpoint* 32°C, diperoleh  $e_{ss}$  (*error steady state*) sebesar 0,23% dan *settling time* sebesar 680 detik (s)
3. Pada pengujian tanpa gangguan tanggapan atau respon pengendalian kelembaban menggunakan kontroler ON-OFF dengan range *setpoint* 55-60%RH diperoleh *settling time* sebesar 109 detik (s).
4. Hasil pengujian keseluruhan dengan gangguan terhadap kontroler PID pada suhu dan kontrol ON-OFF pada kelembaban, menunjukkan bahwa respon sistem dapat kembali pada keadaan *steady state* dan mengalami proses *recovery* (pemulihan) dalam waktu, suhu sebesar 504 detik (s) dan untuk kelembaban sebesar 138 detik (s).
5. Pemanas dan *humidifier* yang telah dikontrol mampu menyediakan kondisi suhu dan kelembaban di dalam *brooder* sesuai dengan kriteria yang dibutuhkan oleh bibit ayam (*DOC*).

### 6.2. Saran

Dalam perancangan dan pembuatan alat ini masih terdapat kelemahan. Agar tercapainya penyempurnaan kinerja alat dan pengembangan lebih lanjut disarankan

1. Apabila akan diaplikasikan untuk skala *brooder* yang sebenarnya agar pemilihan dan penempatan aktuator yang digunakan disesuaikan dengan ukuran *brooder*.
2. Disarankan penggunaan pemanas disesuaikan dengan luas *brooder* dan pola penyebaran panas diperhatikan agar pemanasan lebih merata.
3. Untuk penelitian lebih lanjut disarankan agar sistem yang telah sesuai dengan kebutuhan suhu dan kelembaban bibit ini diujikan langsung pada ayam *DOC* agar dapat diketahui dampak yang dialami pada ayam *DOC* tersebut.

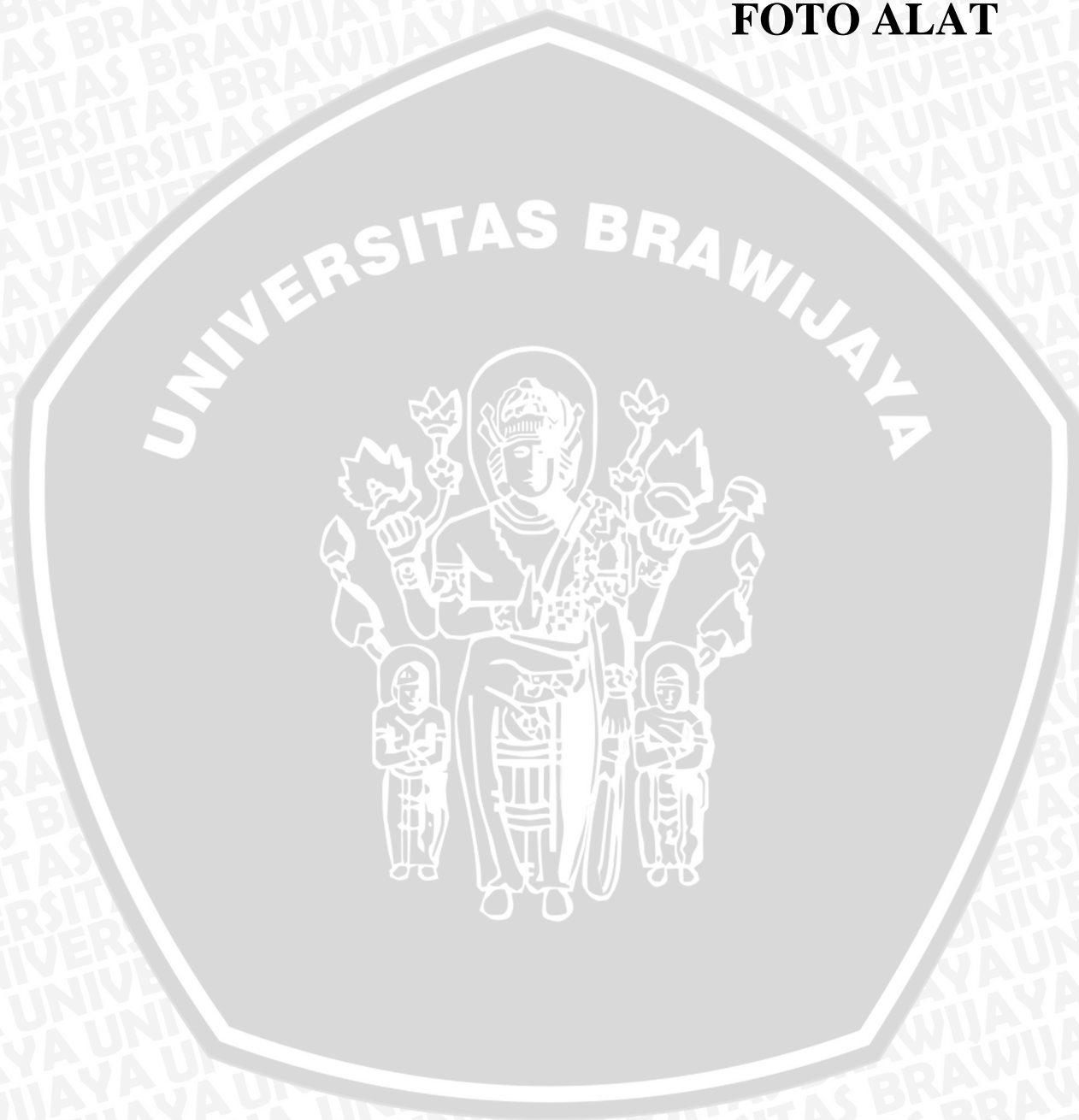
## DAFTAR PUSTAKA

- Dinata, Yuwono Marta. 2014. *Arduino Itu Mudah*. Jakarta: PT. Elex Media Komputindo
- Hario, Samuel. *Pengendalian Temperatur Pada Prototype Kandang Ayam DOC Dengan Menggunakan PLC*. Skripsi Program Sarjana. Universitas Brawijaya.
- Inmojo. 2010. *Digital AC Dimmer Module*. <http://www.inmojo.com/store/inmojo-market/item/digital-ac-dimmer-module/>. (Diakses 7 November 2015).
- Insani, Galuh Adi. 2010. *Beternak Unggas*. Jakarta: PT. Penebar Swadaya.
- Nastiti, Rima. 2014. *Menjadi Milyader Budidaya Ayam Broiler*. Yogyakarta: Pustaka Baru Press.
- Ogata, K. 1997. *Teknik Kontrol Automatik Jilid 2*. Jakarta: Penerbit Erlangga.
- Seborg, D. E., Thomas F. Edgar., & Duncan A. Mellichamp. *Process Dynamics and Control Second Edition*. USA: John Wiley and Sons, Inc.
- Sensirion. 2002. SHT1x Relative Humidity & Temperature Sensor System. <http://www.sensirion.com>. Zurich: Sensirion. (Diakses 1 Desember 2015)



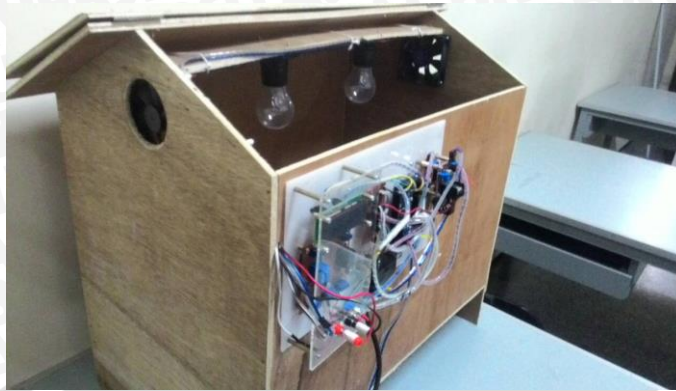
# LAMPIRAN 1

## FOTO ALAT





**Tampak Keseluruhan**



**Tampak Dalam**

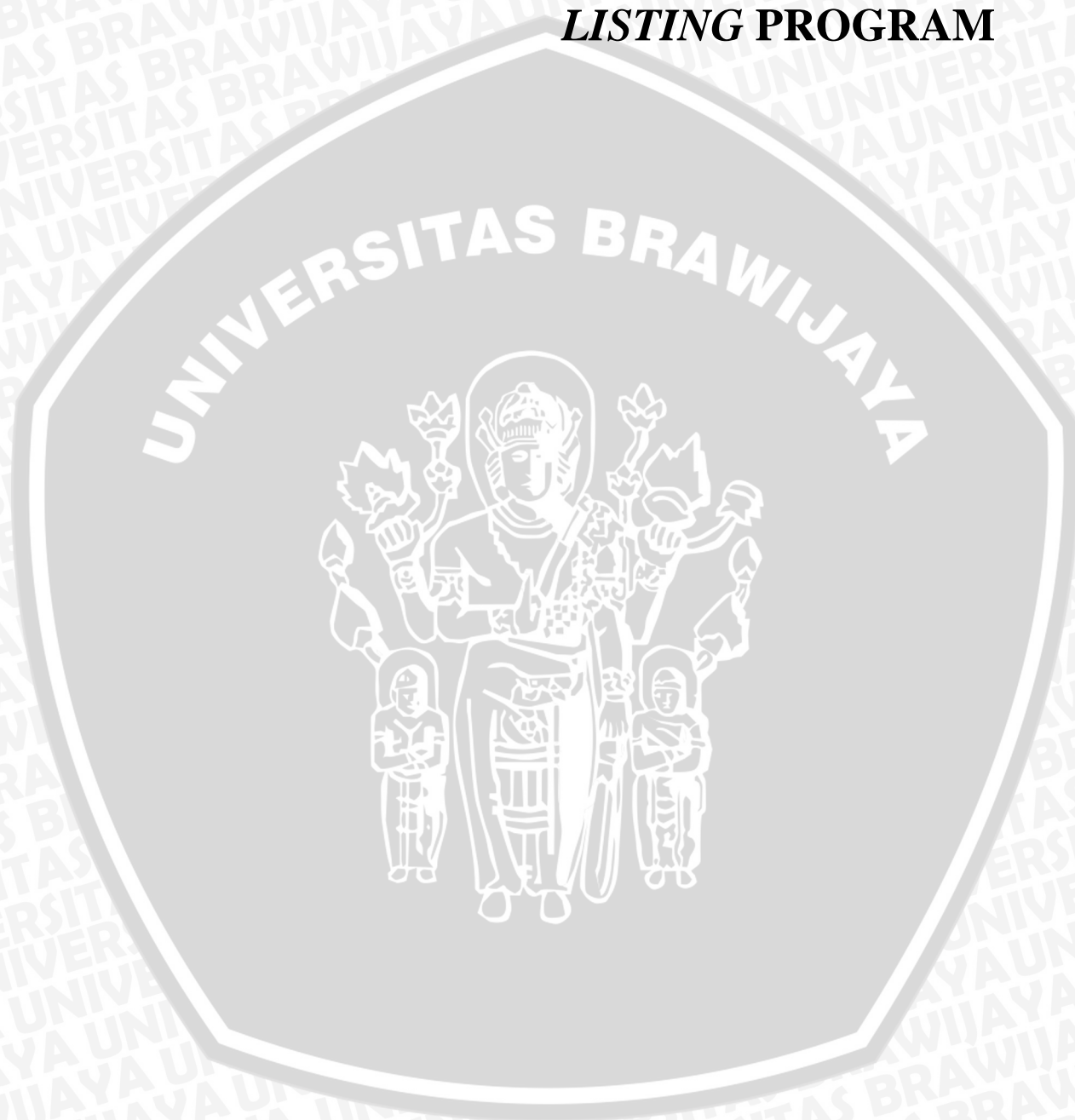


**Rangkaian Elektrik Brooder**



## LAMPIRAN 2

### *LISTING PROGRAM*



## Program Arduino

```
#include <SHT1x.h>

#define dataPin 8 //PIN Data SHT
#define clockPin 9 //PIN Data SHT

SHT1x sht1x(dataPin, clockPin);

int PWM, AC_LOAD=4; //PIN Dimmer
int relay = 31; //PIN Out Humidifier
int enable = 13; //PIN Enable Kipas
int in0 = 11; //PIN Driver Kipas
int in1 = 12; //PIN Driver Kipas

float KP, KI, KD, KI_Last, Kp, Ki, Kd, Ts, PID, delayDIMMER;
float setpoint_temperature, error_temperature, error_temperature1,
temperature;

float setpoint_humidity, error_humidity, humidity;

#include <LiquidCrystal.h>
LiquidCrystal lcd(22, 24, 26, 28, 30, 32);

void setup()
{
    setpoint_temperature=32;
    setpoint_humidity=55;
    Kp=35.56;
    Ki=1.975;
    Kd=160.02;
    Ts=1;

    Serial.begin(9600);
    lcd.begin(16, 2);
    pinMode(34, OUTPUT); //LCD
```

```
digitalWrite(34, HIGH); //LCD
pinMode(10, OUTPUT); //SHT11
digitalWrite(10, HIGH); //SHT11
pinMode(AC_LOAD, OUTPUT); //Dimmer
pinMode(relay, OUTPUT); //Humidifier
attachInterrupt(0, zero_crosss_int, RISING); //ZerroCross Dimmer
digitalWrite(in0, HIGH); //Driver
digitalWrite(in1, LOW); //Driver
```

```
Serial.print("SV_SUHU");
Serial.print("\t\t");
Serial.print("SUHU");
Serial.print("\t\t");
Serial.print("E_SUHU");
Serial.print("\t\t");
Serial.print("PWM");
Serial.print("\t\t");
Serial.print("SV_HMDTY");
Serial.print("\t");
Serial.print("HMDTY");
Serial.print("\t\t");
Serial.print("E_HMDTY");
Serial.print("\n");
}
```

```
//Zerro Cross Dimmer
```

```
void zero_crosss_int()
```

```
{
```

```
int delayDIMMER = (75*PWM);
```

```
delayMicroseconds(delayDIMMER); //t1/ waktu tunda
```

UNIVERSITAS BRAWIJAYA



```
digitalWrite(AC_LOAD, HIGH); //AC_LOAD= trigger triac
delayMicroseconds(10);
digitalWrite(AC_LOAD, LOW);
}

void loop()
{
  //Sensor Read
  temperature = sht1x.readTemperatureC();
  humidity = sht1x.readHumidity();

  //Perhitungan PID
  error_temperature = setpoint_temperature - temperature;
  error_humidity = setpoint_humidity - humidity;

  KP = Kp * error_temperature;
  KI = KI_Last + (Ki * Ts * error_temperature);
  KD = ((Kd/Ts) * error_temperature) - ((Kd/Ts) *
error_temperature1);

  KI_Last = KI;
  error_temperature1 = error_temperature;
  PID = KP+KI+KD;

  //Konversi PID ke PWM
  //if(PID<0){PWM = 125;}
  //else if(PID > 125){PWM = 5;}
  //else {PWM = PID;}

  //PWM 35 - 110
  if(PID<35){PWM = 110;}
  else if(PID > 110){PWM = 35;}
  else {PWM = PID;}
}
```

```
//Aksi Kontrol On-Off Humidifier
if (humidity<60) {
digitalWrite(relay, HIGH);
}
else {
digitalWrite(relay, LOW);
}

//Aksi Kontrol On-Off Penurunan Suhu
if (temperature>33) {
analogWrite(enable, 100); }
else {
analogWrite(enable, 50); }

lcd.setCursor(0,0);
lcd.print("SUHU");
lcd.print(temperature);
lcd.setCursor(0,1);
lcd.print("KELEMBABAN:");
lcd.print(humidity);

Serial.print(setpoint_temperature);
Serial.print("\t\t");
Serial.print(temperature);
Serial.print("\t\t");
Serial.print(error_temperature);
Serial.print("\t\t");
Serial.print(PWM);
Serial.print("\t\t");
Serial.print(setpoint_humidity);
```

```
Serial.print("\t\t");  
Serial.print(humidity);  
Serial.print("\t\t");  
Serial.print(error_humidity);  
Serial.print("\n");  
delay(1050-(delayDIMMER/2.2));  
}
```



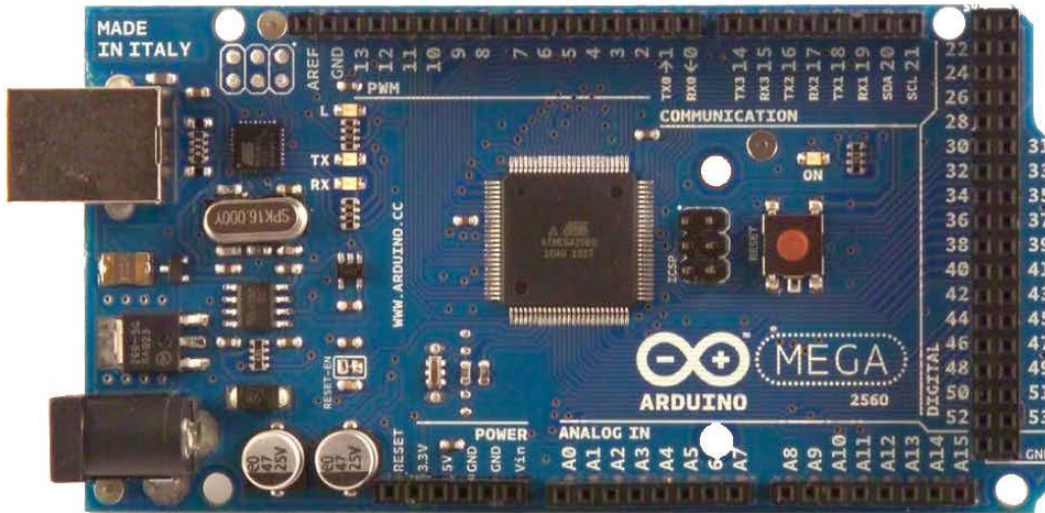
# LAMPIRAN 3

## *DATASHEET*





# Arduino MEGA 2560



## Product Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

## Index

Technical Specifications

Page 2

How to use Arduino  
Programming Environment, Basic Tutorials

Page 6

Terms & Conditions

Page 7

Environmental Policies  
half sqm of green via Impatto Zero®

Page 7



radiospares

RADIONICS





# Technical Specification

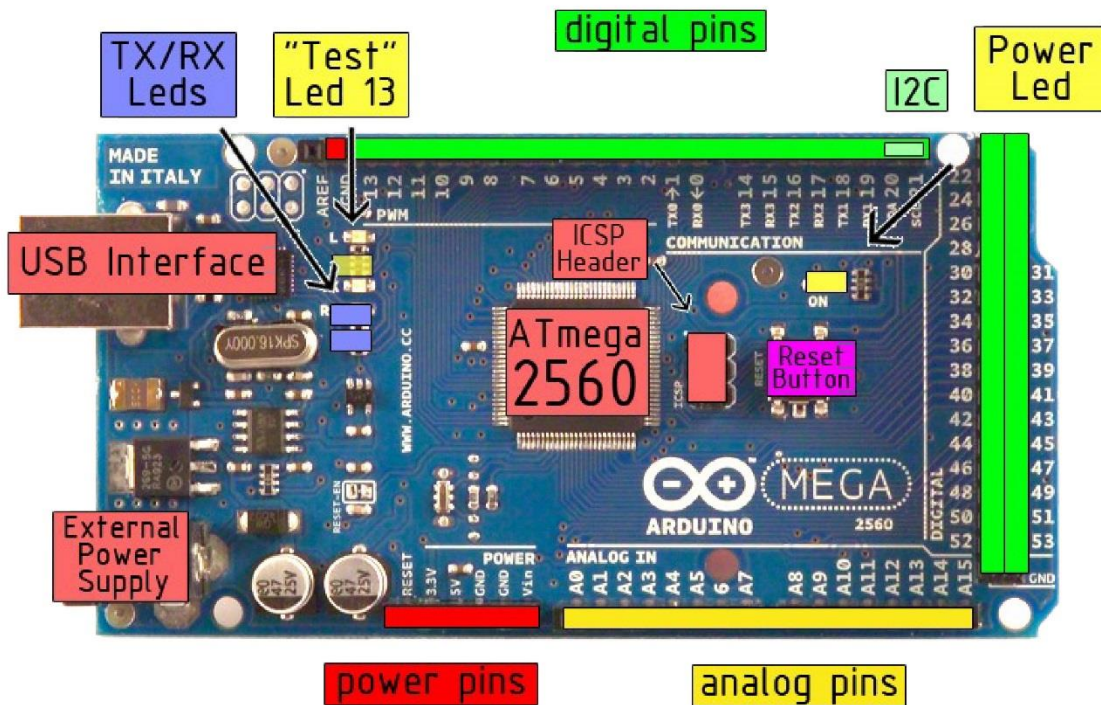


EAGLE files: [arduino-mega2560-reference-design.zip](#) Schematic: [arduino-mega2560-schematic.pdf](#)

## Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

## the board



*radiospares*

*RADIONICS*





## Power

The Arduino Mega2560 can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

## Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

## Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .
- **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2).** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 0 to 13.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Duemilanove and Decimila.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- **I<sup>2</sup>C: 20 (SDA) and 21 (SCL).** Support I<sup>2</sup>C (TWI) communication using the [Wire library](#) (documentation on the Wiring website). Note that these pins are not in the same location as the I<sup>2</sup>C pins on the Duemilanove.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and [analogReference\(\)](#) function.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.



**radiospares**

**RADIONICS**





## Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Mega's digital pins.

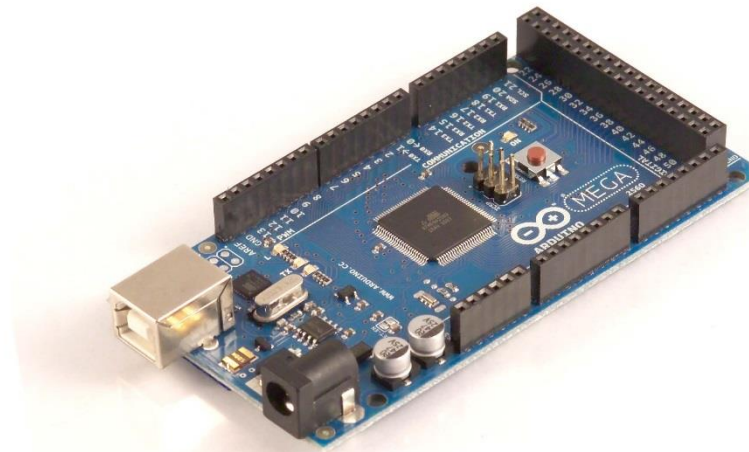
The ATmega2560 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation on the Wiring website](#) for details. To use the SPI communication, please see the ATmega2560 datasheet.

## Programming

The Arduino Mega2560 can be programmed with the Arduino software ([download](#)). For details, see the [reference](#) and [tutorials](#).

The ATmega2560 on the Arduino Mega comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.



*radiospares*

**RADIONICS**







## Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Mega contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

## USB Overcurrent Protection

The Arduino Mega has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

## Physical Characteristics and Shield Compatibility

The maximum length and width of the Mega PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega is designed to be compatible with most shields designed for the Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega and Duemilanove / Diecimila. **Please note that I<sup>2</sup>C is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5).**



radiospares

RADIONICS





# How to use Arduino



Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the [Arduino programming language](#) (based on [Wiring](#)) and the Arduino development environment (based on [Processing](#)). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platoform program. You'll have to follow different instructions for your personal OS. Check on the [Arduino site](#) for the latest instructions. <http://arduino.cc/en/Guide/HomePage>

## Linux Install

## Windows Install

## Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

## Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

**File>Sketchbook>  
Arduino-0017>Examples>  
Digital>Blink**

Once you have your skecth you'll see something very close to the screenshot on the right.

In **Tools>Board** select MEGA

Now you have to go to **Tools>SerialPort** and select the right serial port, the one arduino is attached to.

```

int ledPin = 13; // LED connected to digital pin 13

// The setup() method runs once, when the sketch starts

void setup() {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power

void loop()
{
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(1000); // wait for a second
  digitalWrite(ledPin, LOW); // set the LED off
  delay(1000); // wait for a second
}
    
```



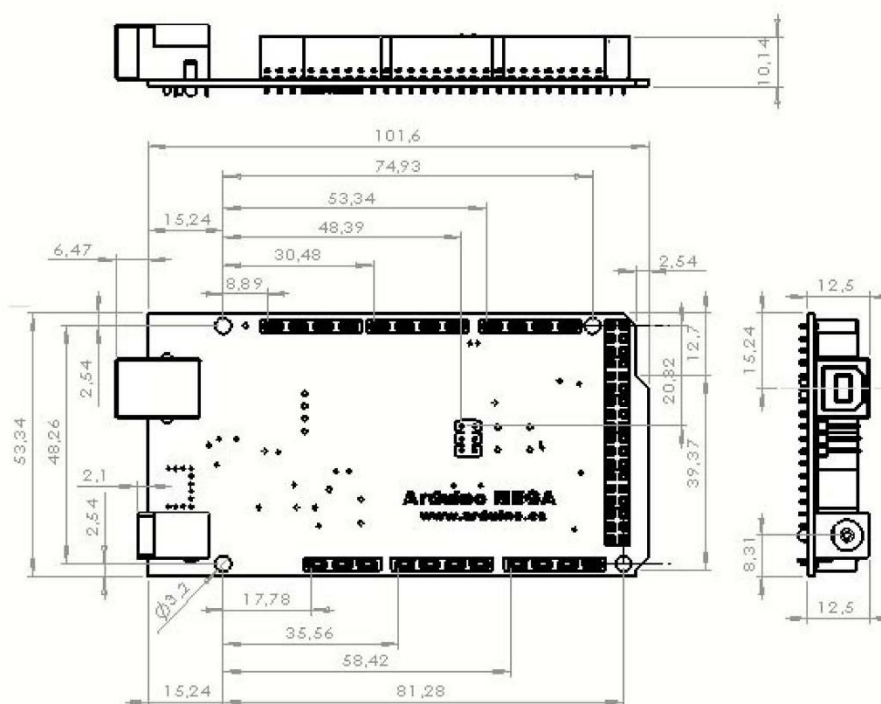
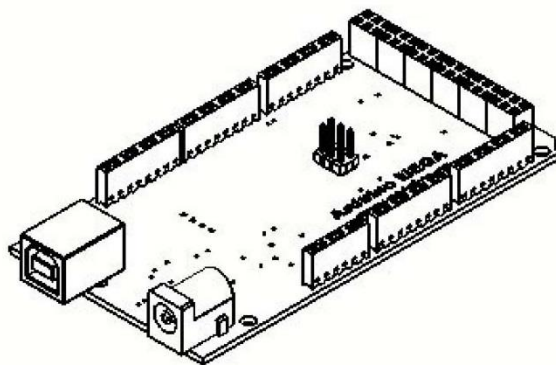
radiospares

RADIONICS





Dimensioned Drawing



*radiospares*

**RADIONICS**





# Terms & Conditions



## 1. Warranties

1.1 The producer warrants that its products will conform to the Specifications. This warranty lasts for one (1) years from the date of the sale. The producer shall not be liable for any defects that are caused by neglect, misuse or mistreatment by the Customer, including improper installation or testing, or for any products that have been altered or modified in any way by a Customer. Moreover, The producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.

1.2 If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are determined by the producer not to conform to such warranty. If the producer elects to replace such products, the producer shall have a reasonable time to replacements. Replaced products shall be warranted for a new full warranty period.

1.3 EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE

1.4 Customer agrees that prior to using any systems that include the producer products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical, applications or design advice, quality characterization, reliability data or other services. Customer acknowledges and agrees that providing these services shall not expand or otherwise alter the producer's warranties, as set forth above, and no additional obligations or liabilities shall arise from the producer providing such services.

1.5 The Arduino™ products are not authorized for use in safety-critical applications where a failure of the product would reasonably be expected to cause severe personal injury or death. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino™ products are neither designed nor intended for use in military or aerospace applications or environments and for automotive applications or environment. Customer acknowledges and agrees that any such use of Arduino™ products which is solely at the Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

1.6 Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino™ products in Customer's applications, notwithstanding any applications-related information or support that may be provided by the producer.

## 2. Indemnification

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third-party losses, damages, liabilities and expenses it incurs to the extent directly caused by: (i) an actual breach by a Customer of the representation and warranties made under this terms and conditions or (ii) the gross negligence or willful misconduct by the Customer.

## 3. Consequential Damages Waiver

In no event the producer shall be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.

## 4. Changes to specifications

The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." The producer reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.



## Environmental Policies



The producer of Arduino™ has joined the Impatto Zero® policy of LifeGate.it. For each Arduino board produced is created / looked after half squared Km of Costa Rica's forest's.



radiospares

RADIONICS



UNIVERSITAS BRAWIJAYA





# DT-SENSE

## SHT11 MODULE

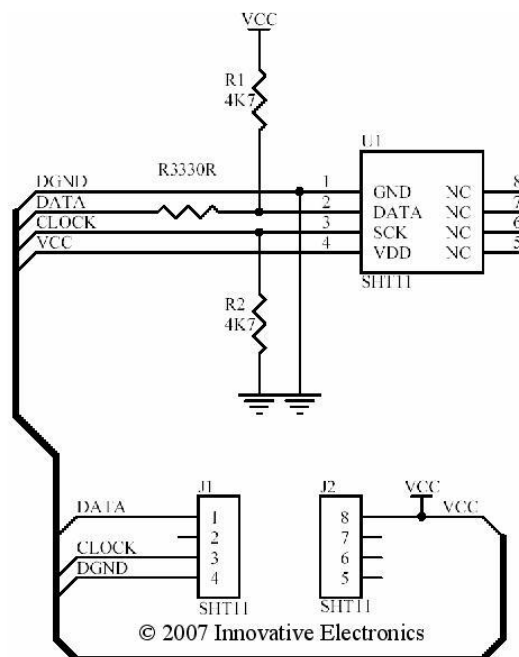
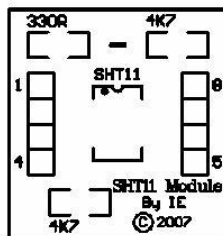
**DT-Sense SHT11 Module** merupakan suatu modul sensor suhu dan kelembaban yang berbasis Sensirion SHT11 ([www.sensirion.com](http://www.sensirion.com)) dengan antarmuka Two-Wire Serial Interface. Modul ini dapat diaplikasikan dalam sistem pengendali suhu dalam ruang atau sistem weather station.

### Spesifikasi Hardware

1. Range suhu : -40°C (-40°F) hingga +123,8°C (+254,9°F)
2. Akurasi suhu : +/- 0,5°C pada 25°C
3. Range kelembaban : 0 hingga 100% RH
4. Akurasi RH absolut : +/- 3,5% RH
5. Faktor bentuk : 8 pin DIP – 0,6"
6. Konsumsi daya rendah (tipikal 30 µW)
7. Tegangan supply +5 VDC

### Tata Letak

Pin	Fungsi
1	Data
3	Clock
4	Ground
8	+5 VDC



Aplikasi terdapat di [www.innovativeelectronics.com](http://www.innovativeelectronics.com).

- ♦ Terima Kasih atas kepercayaan Anda menggunakan produk kami, bila ada kesulitan, pertanyaan atau saran mengenai produk ini silahkan menghubungi technical support kami :

[support@innovativeelectronics.com](mailto:support@innovativeelectronics.com)

# DT-SENSE

## SHT11 MODULE

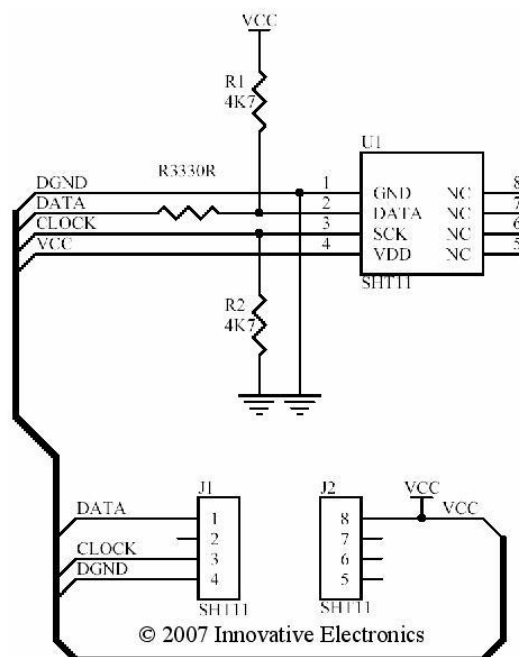
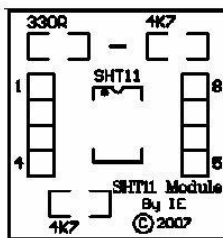
**DT-Sense SHT11 Module** merupakan suatu modul sensor suhu dan kelembaban yang berbasis Sensirion SHT11 ([www.sensirion.com](http://www.sensirion.com)) dengan antarmuka Two-Wire Serial Interface. Modul ini dapat diaplikasikan dalam sistem pengendali suhu dalam ruang atau sistem weather station.

### Spesifikasi Hardware

1. Range suhu : -40°C (-40°F) hingga +123,8°C (+254,9°F)
2. Akurasi suhu : +/- 0,5°C pada 25°C
3. Range kelembaban : 0 hingga 100% RH
4. Akurasi RH absolut : +/- 3,5% RH
5. Faktor bentuk : 8 pin DIP – 0,6"
6. Konsumsi daya rendah (tipikal 30 µW)
7. Tegangan supply +5 VDC

### Tata Letak

Pin	Fungsi
1	Data
3	Clock
4	Ground
8	+5 VDC



Aplikasi terdapat di [www.innovativeelectronics.com](http://www.innovativeelectronics.com).

- ♦ Terima Kasih atas kepercayaan Anda menggunakan produk kami, bila ada kesulitan, pertanyaan atau saran mengenai produk ini silahkan menghubungi technical support kami :

[support@innovativeelectronics.com](mailto:support@innovativeelectronics.com)

