

PENGEMBANGAN KONTROL LEVEL DAN TEMPERATUR AIR  
BERBASIS MIKROKONTROLER



SKRIPSI

Diajukan untuk memenuhi salah satu syarat untuk menyelesaikan  
Pendidikan Diploma empat (D-4) Program Studi D4 Teknik Manufaktur

Jurusan Teknik Mesin

Politeknik Negeri Ujung Pandang

RONALDI REXY MAWATANG 443 13 040

RAHMAT S 443 13 050

PROGRAM STUDI D4 TEKNIK MESIN MEKATRONIKA

JURUSAN TEKNIK MESIN

POLITEKNIK NEGERI UJUNG PANDANG

MAKASSAR

2017

## HALAMAN PENGESAHAN

Skripsi dengan judul **“Pengembangan Kontrol Level dan Temperatur Air Berbasis Mikrokontroler”** oleh Ronaldi Remy Mawatang NIM 443 13 040 dan Rahmat S NIM 443 13 050 telah diterima dan disahkan sebagai salah satu syarat untuk memperoleh gelar S1 Terapan pada program studi D4 Teknik Manufaktur jurusan Teknik Mesin Politeknik Negeri Ujung Pandang.

Makassar, Agustus 2017



Mengesahkan

Pembimbing I,

Pembimbing II,

Handwritten signature of Ir. Simon Ka'ka, M.T.Handwritten signature of Ir. Lewi, M.T.

Ir. Simon Ka'ka, M.T.  
NIP. 19590913 198803 1 001

Ir. Lewi, M.T.  
NIP. 19650913 199103 1 006



Mengetahui,

Ketua Program Studi

D4 Teknik Manufaktur

Handwritten signature of Ir. Abdul Salam, M.T.

Ir. Abdul Salam, M.T.  
NIP. 19601224 1991103 1 001

## HALAMAN PENERIMAAN

Pada hari ini Selasa tanggal 29 Agustus 2017. Panitia Ujian Sidang Skripsi telah menerima dengan baik hasil Skripsi oleh mahasiswa: Ronaldi Remy Mawatang NIM 443 13 040 dan Rahmat S NIM 443 13 050 dengan judul “**Pengembangan Kontrol Level dan Temperatur Air Berbasis Mikrokontroler**”.

Makassar, Agustus 2017



Tim Ujian Sidang Skripsi :

- |  |               |  |
|--|---------------|--|
| 1. Dr.Eng. Abdul Kadir Muhammad, ST., M.Eng. | Ketua         | ( <i>Abdul Kadir</i> )                               |
| 2. Dr.Eng. Akhmad Taufik, ST., MT.           | Sekretaris    | ( <i>Akhmad Taufik</i> ) <sup>02/08/21</sup>         |
| 3. A.M. Anzari, ST., MT.                     | Anggota I     | ( <i>A.M. Anzari</i> ) <sup>06/08/21</sup>           |
| 4. Dr.Ir. Muhammad .Arsyad Habe,MT.          | Anggota II    | ( <i>Muhammad .Arsyad Habe</i> ) <sup>06/10/21</sup> |
| 5. Ir. Simon Ka'ka, MT.                      | Pembimbing I  | ( <i>Simon Ka'ka</i> )                               |
| 6. Ir. Lewi, MT.                             | Pembimbing II | ( <i>Lewi</i> )                                      |

## KATA PENGANTAR

Puji dan syukur penulis ucapkan kehadirat Tuhan Yang Maha Esa atas rahmat dan berkahnya penulis dapat menyelesaikan Skripsi ini dengan judul **“Pengembangan Kontrol Level dan Temperatur Air Berbasis Mikrokontroler”**. Skripsi ini disusun sebagai salah satu syarat guna memperoleh gelar Sarjana Sains Terapan di Politeknik Negeri Ujung Pandang Jurusan Teknik Mesin Program Studi D-4 Teknik Manufaktur Konsentrasi Mekatronika.

Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, maka penulisan Skripsi ini tidak akan lancar. Oleh karena itu pada kesempatan ini, izinkanlah penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. Ir. Hamzah Yusuf, M.S. selaku Direktur Politeknik Negeri Ujung Pandang.
2. Bapak Dr. Jamal, ST., M.T selaku ketua jurusan Teknik Mesin.
3. Bapak Ir. Abdul Salam, M.T. selaku selaku Ketua Program Studi D4 Teknik Mesin Mekatronika Jurusan Teknik Mesin Politeknik Negeri Ujung Pandang.
4. Bapak Ir. Simon Ka'ka, M.T., selaku dosen pembimbing I yang telah memberikan petunjuk, bimbingan, dan arahan selama kami menyelesaikan skripsi.
5. Bapak Ir. Lewi, M.T selaku dosen pembimbing II yang telah memberikan petunjuk, bimbingan, dan arahan selama kami menyelesaikan skripsi.

6. Segenap dosen maupun teknisi di Jurusan Teknik Mesin Politeknik Negeri Ujung Pandang yang telah banyak memberikan dukungan moral maupun materi.
7. Rekan-rekan mahasiswa Jurusan Teknik Mesin terutama teman-teman seperjuangan angkatan 2013 yang telah banyak membantu dan memberikan dukungan serta mendoakan kami dalam pembuatan Skripsi ini.

Penulis menyadari bahwa penulisan Skripsi ini masih jauh dari sempurna, untuk itu penulis mohon kritik, saran dan masukan yang bersifat membangun demi kesempurnaan. Semoga Tuhan Yang Maha Kuasa membalas kebaikan dan ketulusan semua pihak yang telah membantu menyelesaikan skripsi ini dengan melimpahkan rahmat dan karunia-Nya. Semoga skripsi ini dapat memberi manfaat dan kebaikan bagi banyak pihak demi kemaslahatan serta bernilai ibadah di hadapan Tuhan Yang Maha Kuasa.

Makassar, Agustus 2017

Penulis

## DAFTAR ISI

	hlm.
HALAMAN SAMBUNG.....	i
HALAMAN PERSETUJUAN .....	ii
HALAMAN PENERIMAAN .....	iii
KATA PENGANTAR .....	iv
DAFTAR ISI .....	vi
DAFTAR TABEL .....	viii
DAFTAR GAMBAR.....	ix
DAFTAR LAMPIRAN.....	xi
SURAT PERNYATAAN.....	xii
RINGKASAN.....	xiv
SUMMARY .....	xv
BAB 1. PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	3
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	3
BAB II. TINJAUAN PUSTAKA .....	5
2.1. Kontrol Level Air.....	5
2.2. Kontrol Temperatur Air.....	5

2.3. Sistem Kontrol .....	5
2.4. Pompa Air.....	12
2.5. Heater.....	14
2.6. Sensor .....	14
2.7. Driver Motor.....	24
2.8. Driver Heater.....	24
2.9. Bahasa Pemograman MATLAB.....	25
<b>BAB III. METODE PENELITIAN .....</b>	<b>28</b>
3.1. Tempat dan Waktu Penelitian.....	28
3.2. Alat dan Bahan.....	28
3.3. Metode Penelitian .....	29
3.4. Diagram Alir Kontrol Level dan Temperatur Air.....	31
3.5. Diagram Blok Kontrol level dan Temperatur Air.....	32
3.6. Desain Alat Kontrol Level dan Temperatur Air.....	32
<b>BAB IV. HASIL DAN PEMBAHASAN .....</b>	<b>40</b>
4.1 Hasil .....	40
4.2 Pembahasan .....	46
<b>BAB V PENUTUP .....</b>	<b>52</b>
5.1 Kesimpulan.....	52
5.2 Saran .....	52
<b>DAFTAR PUSTAKA .....</b>	<b>54</b>
<b>LAMPIRAN .....</b>	<b>56</b>

## DAFTAR TABEL

	hlm.
Tabel 2.1 <i>Output</i> Data Digital dari Sensor DS18b20 .....	23
Tabel 3.1 Daftar Alat dan Bahan .....	28
Tabel 4.1 Data Hasil Pengujian <i>Heater</i> .....	49





## DAFTAR GAMBAR

	hlm.
Gambar 2.1 Arduino AT Mega .....	7
Gambar 2.2 Kontrol Loop Tertutup.....	8
Gambar 2.3 Diagram Kontrol P.....	8
Gambar 2.4 Kontrol P Pada Motor DC.....	9
Gambar 2.5 Diagram Kontrol I.....	10
Gambar 2.6 Kontrol PI Pada Motor DC .....	11
Gambar 2.7 Diagram Kontrol D.....	11
Gambar 2.8 Kontrol PD Pada Motor DC.....	12
Gambar 2.9 Pompa Air .....	12
Gambar 2.10 <i>Heater</i> .....	14
Gambar 2.11 Prinsip Kerja Sensor Ultrasonik .....	17
Gambar 2.12 Prinsip Pemantulan Sensor Ultrasonik .....	18
Gambar 2.13 Pemancar Ultrasonik <i>Transmitter</i> .....	19
Gambar 2.14 Sensor Ultrasonik HC-SR04 .....	21
Gambar 2.15 Format Register Suhu pada DS18B20.....	22
Gambar 2.16 Sensor Temperatur DS18B20.....	24
Gambar 2.17 Driver Motor BTS7960.....	24
Gambar 2.18 <i>Driver Heater</i> .....	25
Gambar 3.1 Diagram Alir Penelitian .....	31

Gambar 3.2 Diagram Alir Kontrol Level dan Temperatur Air .....	31
Gambar 3.3 Diagram Blok Kontrol Level dan Temperatur Air .....	32
Gambar 3.4 Desain Alat kontrol level dan temperatur air .....	33
Gambar 3.6 Gambar Tangki B .....	34
Gambar 3.7 Dudukan Tangki .....	34
Gambar 4.1 Hasil Perancangan Mekanik.....	41
Gambar 4.2 <i>Schematic</i> Rangkaian Kontrol Level dan Temperatur Air.....	43
Gambar 4.3 <i>layout</i> Rangkaian Kontrol Level dan Temperatur Air.....	43
Gambar 4.4 <i>layout Driver Heater</i> .....	43
Gambar 4.5 Diagram Blok Simulink Kontrol Level Air.....	44
Gambar 4.6 Grafik Simulink Kontrol level Air.....	45
Gambar 4.7 Diagram Blok Simulink Kontrol Temperatur Air.....	45
Gambar 4.8 Grafik Simulink Kontrol Temperatur Air.....	46
Gambar 4.9 Kondisi Awal Sensor Temperatur.....	49
Gambar 4.10 Kondisi Saat Mencapai Suhu yang Diinginkan.....	49
Gambar 4.11 Grafik Hasil Pengujian <i>Heater</i> .....	45
Gambar 4.12 Pengujian Grafik <i>Real Time</i> .....	46

## DAFTAR LAMPIRAN

	hlm.
Lampiran 1 Spesiikasi Arduino Mega .....	53
Lampiran 2 Desain Alat .....	57
Lampiran 3 Program Kontrol Level dan Temperatur Air .....	63
Lampiran 4 Dokumentasi Penelitian.....	74
Lampiran 5 Lembar Asistensi.....	80



## SURAT PERNYATAAN

Kami yang bertanda tangan di bawah ini:

Nama : Ronaldi Rexy Mawatang

NIM : 443 13 040

menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam skripsi ini yang berjudul "Pengembangan Kontrol Level dan Temperatur Air Berbasis Mikrokontroler" merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing dan belum pernah diajukan dalam bentuk apa pun pada perguruan tinggi dan instansi mana pun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya penulis lain telah disebutkan dalam naskah dan dicantumkan dalam skripsi ini.

Jika pernyataan kami tersebut di atas tidak benar, kami siap menanggung risiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, Agustus 2017

Ronaldi Rexy Mawatang  
443 13 040

## SURAT PERNYATAAN

Kami yang bertanda tangan di bawah ini:

Nama : Rahmat S

NIM : 443 13 050

menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam skripsi ini yang berjudul "Pengembangan Kontrol Level dan Temperatur Air Berbasis Mikrokontroler" merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing dan belum pernah diajukan dalam bentuk apa pun pada perguruan tinggi dan instansi mana pun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya penulis lain telah disebutkan dalam naskah dan dicantumkan dalam skripsi ini.

Jika pernyataan kami tersebut di atas tidak benar, kami siap menanggung risiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, Agustus 2017

Rahmat S  
443 13 050

# PENGEMBANGAN KONTROL LEVEL DAN TEMPERATUR AIR BERBASIS MIKROKONTROLER

## RINGKASAN

Pengembangan Kontrol Level dan Temperatur Air Berbasis Mikrokontroler merupakan pengembangan dari Rancang Bangun Kontrol Level Air Berbasis Mikrokontroler. Alat ini merupakan alat yang dapat digunakan untuk memonitor, serta mengontrol level dan temperatur air.

Tujuan dari penelitian ini adalah untuk membuat alat kontrol level dan temperatur air berbasis mikrokontroler, dan menampilkan informasi level dan temperatur air pada komputer. Metode penelitian yang digunakan selama proses pelaksanaan penelitian ini yakni : studi literatur, perancangan hardware dan software, serta melakukan pengambilan data. Kontrol Level dan Temperatur Air Berbasis Mikrokontroler menggunakan dua buah sensor yakni: sensor ultrasonic untuk mendeteksi level ketinggian air, kemudian sensor yang kedua adalah sensor termokopel yang digunakan untuk mendeteksi temperatur air. Alat ini menggunakan pompa air sebagai penyalur air dari tangki satu ke tangki yang lain, dan *heater* yang digunakan untuk memanaskan air. Alat ini memiliki 2 tombol, dimana tombol yang pertama digunakan untuk mengontrol level air, tombol yang kedua digunakan untuk mengontrol level dan temperatur air.

Berdasarkan hasil penelitian dapat disimpulkan bahwa sistem kontrol level dan temperatur air ini bekerja dengan baik sesuai dengan yang diharapkan, yaitu dimana alat ini menggunakan mode sistem kontrol multi posisi yang telah diprogram dalam *software* Arduino dan dapat menampilkan hasil pembacaan sensor berupa grafik menggunakan *software* matlab.

**Kata Kunci :** Mikrokontroler Arduino , Sensor Ultrasonik, Sensor Termokopel, Pompa Air, *Heater*, PC

# DEVELOPMENT OF LEVEL CONTROL AND WATER TEMPERATURE BASED ON MICROCONTROLLER

## SUMMARY

Development of Level Control and Water Temperature Based on Microcontroller is the Development of the Design of Water Level Control Based on Microcontroller. This tool is a tool that can be used to monitor, and control the level and temperature of water.

The purpose of this research is to create microcontroller based level and temperature controller, and display the water level and temperature information of the computer. Research methods used during the implementation of this research process are: literature studies, designing hardware and software, and do data retrieval. Level Control and Water Temperature Based Microcontroller uses two sensors: ultrasonic sensor to detect water level, then the second sensor is a thermocouple sensor used to detect water temperature. It uses a water pump as a water supply from one tank to another, and a heater used to heat water. This tool has 2 buttons, where the first button is used to control the water level, the second button is used to control the water level and temperature.

Based on the result of this research, it can be concluded that the system of water level and temperature control works well as expected, that is, this tool uses multi-position control system mode programmed in arduino software and can display the sensor readings in the form of graphic using matlab software.

**Keywords:** Arduino Microcontroller, Ultrasonic Sensor, Thermocouple Sensor, Water Pump, Heater, PC



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan teknologi elektronika saat ini mengalami peningkatan yang sangat pesat. Perkembangan ini cenderung mengarah pada peningkatan optimalisasi kecepatan kerja dan minimalisasi. Artinya peralatan dan komponen elektronika diupayakan menggunakan materi dan ukuran yang semakin kecil tetapi mempunyai kemampuan kecepatan maupun kerja yang lebih tinggi.

Sistem kontrol level air merupakan sistem yang digunakan untuk menjamin kontinuitas persediaan air dalam sebuah tangki penampungan yang akan digunakan untuk proses industri atau dalam kehidupan sehari-hari. Misalnya, pada industri minuman, industri pengolahan air bersih, pembangkit listrik tenaga uap (PLTU), industri kimia, dan lain-lain. Selain itu sistem kontrol ini juga dapat digunakan untuk memonitoring proses industri yang dikendalikan dari jauh, sehingga bisa menghemat biaya, waktu dan tenaga.

Sistem kontrol temperatur air merupakan sistem yang digunakan untuk menjamin suhu air tetap berada pada suhu yang diinginkan pada sebuah tangki penampungan, yang nantinya akan digunakan untuk proses industri atau dalam kehidupan sehari-hari. Misalnya, pada pembudidayaan ikan *black ghost* suhu air yang diijinkan berkisar antar rentang tertentu yaitu “*temperatur air selama*



*masa pembudidayaan harus selalu dipertahankan yaitu berkisar antara 26 sampai dengan 27 derajat celcius. (Yuono, 2006).*

Untuk menjaga level ketinggian dan temperatur air pada tangki maka digunakan sebuah alat kontrol, yaitu mikrokontroler untuk mengatur kecepatan pompa, dan heater sesuai dengan ketinggian dan suhu air yang diinginkan. Selain itu juga dibutuhkan sebuah sensor yang berfungsi untuk mendeteksi setiap penurunan level dan temperatur air dan ditindaklanjuti melalui sistem kontrol tersebut.

Penelitian sebelumnya mengenai sistem kontrol level, dilakukan pada tahun 2016 oleh Irman Puluhulawa dan Mangku Alam, alumni Politeknik Negeri Ujung Pandang dengan judul “Rancang Bangun Kontrol Level Air Berbasis Mikrokontroler Menggunakan Sensor Ultrasonik”. Pada kesempatan ini kami akan mengembangkan sistem tersebut, dengan judul “Pengembangan Kontrol Level dan Temperatur Air Berbasis Mikrokontroler”, kami menambahkan sebuah kontrol PID, dan menambah satu sistem yaitu sistem kontrol temperatur.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang sudah dipaparkan sebelumnya dapat dirumuskan sebagai berikut:

1. Bagaimana membuat sistem kontrol level dan temperatur air berbasis mikrokontroler?
2. Bagaimana cara mempertahankan level dan temperatur air?

3. Bagaimana cara menampilkan informasi level ketinggian dan temperatur air pada *interface*?

### **1.3 Batasan Masalah**

Dalam pembuatan Skripsi ini dibatasi beberapa hal sebagai berikut:

1. Sensor yang digunakan untuk mendeteksi level air adalah sensor ultrasonik.
2. Sensor yang digunakan untuk mendeteksi suhu air adalah sensor temperatur DS18B20.
3. Arduino AT Mega sebagai alat untuk memproses data dan pengatur dari seluruh kegiatan sistem yang dibuat.
4. Suhu air maksimal pada tangki penampungan adalah 40° C.
5. Ketinggian maksimal air yang diinginkan adalah 16 cm.

### **1.4 Tujuan Penelitian**

Adapun tujuan dari Skripsi ini adalah sebagai berikut:

1. Untuk membuat sistem kontrol level dan temperatur air berbasis mikrokontroler.
2. Untuk mempertahankan level dan temperatur air.
3. Untuk menampilkan informasi level ketinggian dan temperatur air pada *interface*.

### **1.5 Manfaat Penelitian**

Adapun manfaat yang akan diperoleh dari Skripsi ini adalah sebagai berikut:

1. Membantu manusia dalam mengukur, memantau, dan mengontrol level dan temperatur air di dalam tangki penampungan.
2. Menjaga kontinuitas persediaan air dalam tangki air yang akan digunakan untuk proses industri.
3. Menjaga level dan temperatur air agar tetap pada ketinggian dan suhu yang diinginkan.
4. Dapat digunakan sebagai media pembelajaran untuk mahasiswa.



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Kontrol Level Air

“*Water Level Control (WLC)*” merupakan sebuah alat yang digunakan untuk mengidentifikasi ketinggian air yang berada di dalam penampungan dengan memanfaatkan sensor sebagai pendeteksi ketinggian dari air dan memiliki aktuator seperti motor sebagai pemompa air. Sistem ini banyak digunakan di dalam dunia industri maupun di dalam kehidupan sehari-hari, karena sederhana dan perawatannya mudah.

#### 2.2 Kontrol Temperatur Air

Kontrol temperatur air merupakan sebuah alat yang dapat mengukur atau mengontrol besarnya suhu air yang terdapat pada suatu benda, bidang atau ruang, untuk diproses lebih lanjut. Input atau masukan pada umumnya menggunakan sensor temperatur, kemudian besarnya nilai suhu yang diukur akan ditampilkan pada layar.

#### 2.3 Sistem Kontrol

Sistem kontrol adalah proses pengaturan pengendalian terhadap satu atau beberapa besaran (*variable, parameter*) sehingga berada pada suatu kisaran

tertentu. Gabungan kerja dari berbagai alat-alat kontrol dalam proses produksi dinamakan sistem pengontrolan proses (*process control system*), sedangkan semua peralatan yang membentuk sistem pengontrolan disebut pengontrolan instrumentasi proses (*process control instrumentation*).

Menurut Curtis D. Johnson (1989), mode alat kontrol terbagi atas dua macam, yaitu mode alat kontrol diskontinyu dan mode alat kontrol kontinyu. Mode alat kontrol diskontinyu terbagi atas 2 macam, yaitu mode dua posisi dan mode multi posisi. Mode dua posisi ini hanya bekerja pada dua posisi, yaitu *on* dan *off*. Kerja kontroler *on-off* banyak digunakan pada aksi pengontrolan yang sederhana karena harganya murah dan mode alat kontrol ini merupakan yang paling dasar. Mode multi posisi adalah mode alat kontrol dua posisi yang diberikan beberapa pengaturan lanjutan (*intermediate setting*).

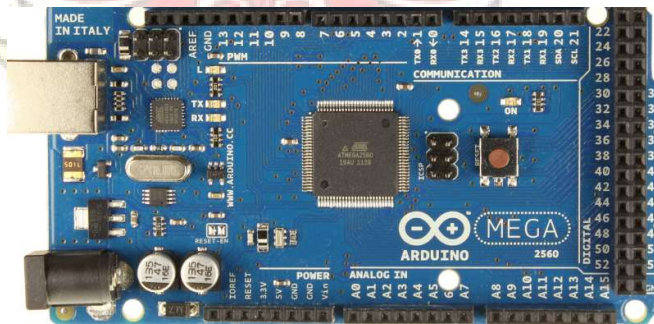
### **2.3.1 Mikrokontroler**

Mikrokontroler adalah sebuah sistem komputer fungsional dalam sebuah *chip* berupa IC (*Integrated Circuit*) yang dapat menerima sinyal input, mengolahnya dan memberikan sinyal output sesuai dengan program yang diisikan ke dalamnya. Dengan kata lain, mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus, cara kerja mikrokontroler sebenarnya membaca dan menulis data.

Mikrokontroler pada dasarnya adalah komputer dalam satu *chip*, yang di dalamnya terdapat mikroprosesor, memori, jalur *Input/Output*

(I/O) dan perangkat pelengkap lainnya. Kecepatan pengolahan data pada mikrokontroler lebih rendah jika dibandingkan dengan PC. Pada PC kecepatan mikroprosesor yang digunakan saat ini telah mencapai orde GHz, sedangkan kecepatan operasi mikrokontroler pada umumnya berkisar antara 1 – 16 MHz. Begitu juga kapasitas RAM dan ROM pada PC yang bisa mencapai orde *Gbyte*, dibandingkan dengan mikrokontroler yang hanya berkisar pada orde *byte/Kbyte*.

Salah satu mikrokontroler yang sering digunakan adalah Arduino merupakan rangkaian elektronik yang bersifat *open source*, serta memiliki perangkat keras dan lunak yang mudah untuk digunakan. Arduino dapat mengenali lingkungan sekitarnya melalui berbagai jenis sensor dan dapat mengendalikan lampu, motor, dan berbagai jenis aktuator lainnya. Arduino mempunyai banyak jenis, di antaranya Arduino Uno, Arduino Mega 2560, Arduino Fio, dan lainnya.



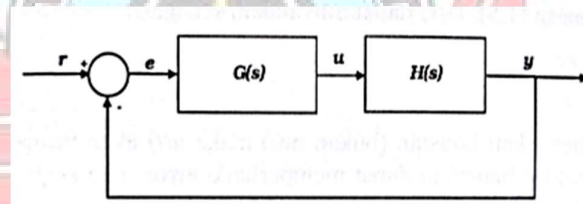
Gambar 2.1 Arduino AT Mega.

Arduino adalah sebuah *board mikrokontroller* yang berbasis ATmega328. Arduino memiliki 14 pin input/output yang mana 6 pin dapat digunakan sebagai output PWM, 6 *analog input*, *crystal osilator* 16 MHz,

koneksi USB, *jack power*, kepala ICSP, dan tombol *reset*. Arduino mampu men-*support* mikrokontroler dan dapat dikoneksikan dengan komputer menggunakan kabel USB. (Feri Djuandi, 2011).

### 2.3.2 Kontrol (*Proporsional, Integral, Derifatif*) PID

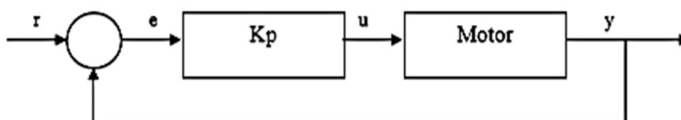
Kita akan menggambarkan bentuk pernyataan standar dalam sistem kontrol seperti pada gambar dibawah berikut. Dalam gambar,  $r$  adalah input,  $e$  adalah *error*,  $u$  adalah sinyal *output* kontroler,  $G(s)$  adalah kontroler,  $H(s)$  adalah dinamik robot, dan  $y$  adalah output. Sekarang permasalahannya adalah bagaimana  $G(s)$  didisain.



Gambar 2.2. Kontrol Loop tertutup

#### 2.3.2.1 Kontrol Proporsional (P) untuk motor DC

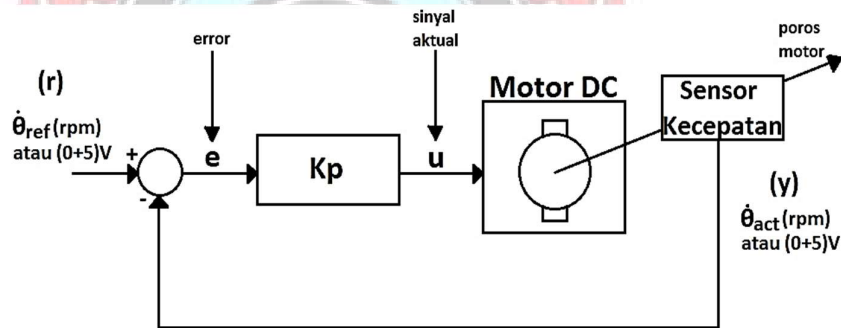
Kontroler adalah kontrol P jika  $G(s) = k$ , dengan  $k$  adalah konstanta jika,  $u = G(s).e$  maka Persamaan output kontroler  $u = k_p.e$  Kontrol P untuk sebuah motor dapat diilustrasikan sebagai berikut.



Gambar 2.3 Diagram Kontrol P

Dengan  $K_p$  adalah konstanta proportional.  $K_p$  berlaku sebagai *Gain* (Penguat) saja tanpa memberikan efek dinamik pada kinerja kontroler.

Penggunaan kontrol P memiliki berbagai keterbatasan karena sifat kontrol yang tidak dinamik ini. Walaupun demikian dalam aplikasi aplikasi dasar sederhana, kontrol P ini cukup mampu untuk konvergensi meskipun *error* keadaan tenangnya (*steady state*) relatif besar. Sebagai contoh, kita akan mengontrol kecepatan putar sebuah motor DC dengan menggunakan kontrol P, ilustrasinya adalah sebagai berikut:



Gambar 2.4 Kontrol P pada motor DC

Dalam fungsi  $t$  maka output kontroler  $u(t)$  dapat ditulis:

$$u(t) = K_p \cdot (\theta_{ref}(t) - \theta_{act}(t)) \dots\dots\dots(1)$$

Nilai output kontroler tergantung kepada perwakilan antara *error*, yakni kecepatan referensi dikurangi kecepatan aktual, dengan lebih kecil daripada kecepatan referensi. Jika *error* negatif berarti kecepatan aktual lebih besar daripada kecepatan referensi. Jika

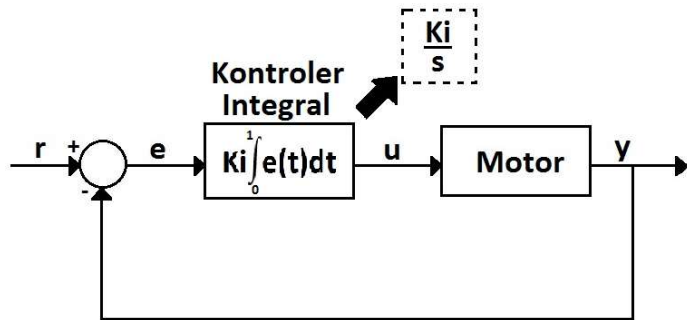


kecepatan aktual sama dengan kecepatan referensi maka sinyal output akan menjadi nol karena error nol, dari segi rangkaian, sinyal output ini akan menyebabkan motor tidak mendapat sinyal aktuator lagi. Akibatnya poros motor akan berhenti berputar. Begitu putaran poros mulai berkurang maka sensor akan mendeteksi bahwa kecepatan output tidak lagi sama dengan kecepatan referensi. Kecepatan aktual akan lebih kecil dari kecepatan referensi. Artinya error tidak lagi nol. Akibatnya, kontroler akan mulai lagi mengirim sinyal aktuator  $u(t)$  positif sehingga motor kembali menambah kecepatannya.

Demikian hal ini berulang seterusnya sehingga error pada kontrol P ini tidak dapat mempertahankan error selalu nol atau dengan kata lain dalam kondisi tetap (*steady state*), error pada kontrol P tidak bisa nol. Hal ini dikenal sebagai *steady state error* (ess).

### 2.3.2.2 Kontrol Integral (I) untuk motor DC

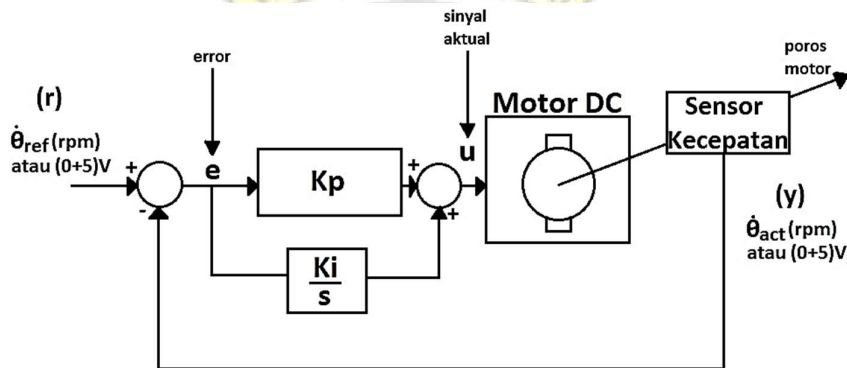
Fungsi dasar dari kontrol I adalah menurunkan *steady state error*. Kontrol I jarang digunakan sendirian dalam aplikasi. Biasanya selalu dikombinasikan dengan kontrol P untuk memperbaiki respon guna mencapai error minimum. Parameter kontrol I dapat diilustrasikan dalam diagram kontrol motor sebagai berikut.



Gambar 2.5 Diagram Kontrol I

Gabungan kontrol I dengan kontrol P untuk sebuah motor

DC dapat digambarkan sebagai berikut:



Gambar 2.6 Kontrol PI pada motor DC

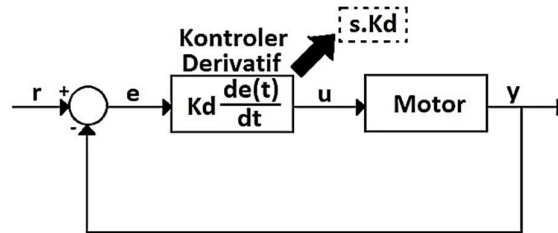
Persamaan output kontroler diatas adalah:

$$u(t) = Kp \cdot (\theta_{ref}(t) - \theta_{act}(t)) + Ki \int_0^t (\theta_{ref}(t) - \theta_{act}(t)) dt \dots \dots \dots (2)$$

### 2.3.2.3 Kontrol Derifatif (D) untuk motor DC

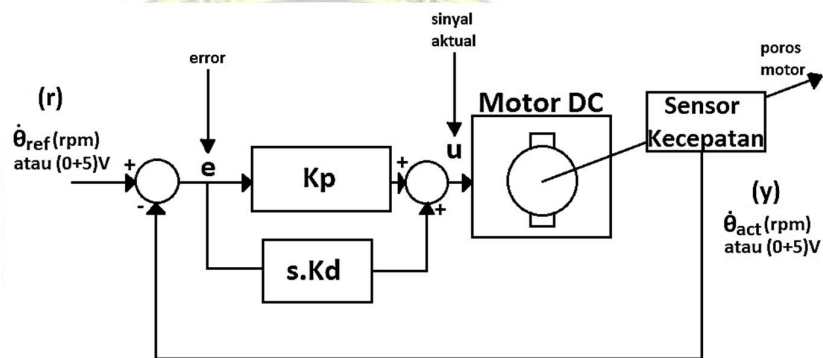
Parameter D bekerja dalam konteks rate/kecepatan dari error sehingga dapat mengurangi efek overshoot (respon berlebihan) dalam menuju steady state. Dengan kata lain, seolah olah kontrol D mampu memprediksi error yang akan terjadi sebagai efek dari kecepatan error yang dihitung sebelumnya.

Parameter kontrol D dapat diilustrasikan dalam diagram kontrol motor sebagai berikut:



Gambar 2.7 Diagram Kontrol D

Gabungan kontrol D dengan kontrol P untuk sebuah motor DC dapat digambarkan sebagai berikut:



Gambar 2.8 Kontrol PD pada motor DC

Sehingga persamaan output kontroler  $u(t)$ ,

$$u(t) = Kp.(\theta_{ref}(t) - \theta_{act}(t)) + Kd. \frac{d(\theta_{ref}(t) - \theta_{act}(t))}{dt} \dots\dots\dots(3)$$

## 2.4 Pompa Air

Pompa adalah salah satu mesin fluida yang termasuk dalam golongan mesin kerja. Pompa berfungsi untuk mengalirkan fluida dari tempat yang rendah ke tempat yang lebih tinggi atau dari tempat yang bertekanan rendah ke tempat yang bertekanan lebih tinggi. Disamping itu juga pompa digunakan untuk memindahkan fluida dari tempat yang lebih tinggi ke tempat yang lebih rendah, misalnya pada sistem pemipaan yang panjang dan berkelok-kelok.



Gambar 2.9. Pompa Air

### 2.4.1 Klasifikasi Pompa

Pompa dapat diklasifikasikan secara umum berdasarkan pemberian energi pada fluida kerjanya, yaitu:

1. Pompa Tekanan Statis (*Positive Displacement Pump*)
2. Pompa Tekanan Dinamis (*Dynamic Pressure Pump*)

### 2.4.2 Daya Pompa

Daya yang diberikan kepada pompa harus lebih besar dari daya akibat fluida dan akibat dari kerugian-kerugian yang terjadi. Daya pompa dapat dihitung dengan persamaan:

$$P_p = \frac{\rho \cdot Q \cdot H \cdot g}{\eta_p} \dots\dots\dots(4)$$

Dimana:

$P_p$  = daya pompa [watt]

$H$  = ketinggian [m]

$\rho$  = massa jenis fluida [ $\text{kg/m}^3$ ]

$Q$  = kapasitas pompa [ $\text{m}^3/\text{s}$ ]

$\eta_p$  = efisiensi pompa [%]

$g$  = gravitasi bumi [ $\text{m/s}^2$ ]

## 2.5 Heater

*Heater* adalah sistem atau mesin yang dirancang untuk menaikkan suhu (yang digunakan untuk pemanasan tergantung pada sifat udara pada waktu tertentu). *Water heater* bekerja dengan cara mengeluarkan panas. Energi panas yang dipakai berasal dari hasil pemanasan *heater*. Secara garis besar, peralatan ini terbuat dari metal (*metal housing*) yang dilapisi *refractory* pada bagian dalamnya sebagai isolasi panas sehingga panas tidak terbuang keluar. Material yang dipanaskan/*charge* bisa berbentuk padat, cair atau gas. Dalam rangkaian Skripsi ini *heater* digunakan untuk menaikkan suhu air dalam bak sesuai dengan suhu yang telah di set. Oleh karena itu, diperlukan rangkaian *driver heater*. Saat suhu dalam bak lebih kecil dari target yang telah diatur, maka akan mengaktifkan rangkaian *driver heater* kemudian *heater* akan menyala.



Gambar 2.10 *Heater*

## 2.6 Sensor

Adapun sensor yang digunakan pada Skripsi ini yaitu:

### 2.6.1 Sensor Ultrasonik

Penelitian tentang sensor ultrasonik telah banyak dilakukan oleh peneliti-peneliti sebelumnya di antaranya oleh Heru Rianto (2010) dalam

penelitiannya yang berjudul Pengaman Parkir Mobil Menggunakan Sensor Ultrasonik Berbasis Mikrokontroler. Hasil yang diperoleh dari penelitian ini adalah berupa alat yang dapat mendeteksi adanya benda penghalang yang berada di sekitar mobil.

K.G. Suastika, M. Nawir, P. Yunus dalam Jurnal Pendidikan Fisika Indonesia (2013), Universitas Palangka Raya tentang Sensor Ultrasonik sebagai Alat Pengukur Kecepatan Aliran Udara Dalam Pipa. Penelitian yang dilakukan adalah penelitian tentang pengukuran kecepatan aliran udara dalam pipa menggunakan sensor ultrasonik dan gelombang ultrasonik yang digunakan pada penelitian ini adalah gelombang ultrasonik yang memiliki frekuensi kerja sebesar 300 kHz. Prinsip pengukuran yang digunakan dalam penelitian ini menggunakan metode waktu tempuh gelombang ultrasonik (*time of flight*) dengan memanfaatkan perubahan karakteristik gelombang ultrasonik ketika melewati kondisi aliran udara yang berbeda yaitu *upstream* dan *downstream*. Selain itu, sebagai pembanding (tingkat akurasi) dalam penelitian ini digunakan alat pengukur kecepatan aliran udara standar yaitu *anemometer*. Dari hasil penelitian didapat bahwa tingkat akurasi sebesar 99% dan dengan korelasi sebesar 0,99 (korelasi sangat tinggi). Berdasarkan hasil tersebut, dapat disimpulkan bahwa sensor ultrasonik *valid* dan dapat digunakan sebagai perangkat pengukur kecepatan aliran udara dalam pipa. Berdasarkan penelitian tersebut, sensor ultrasonik baik

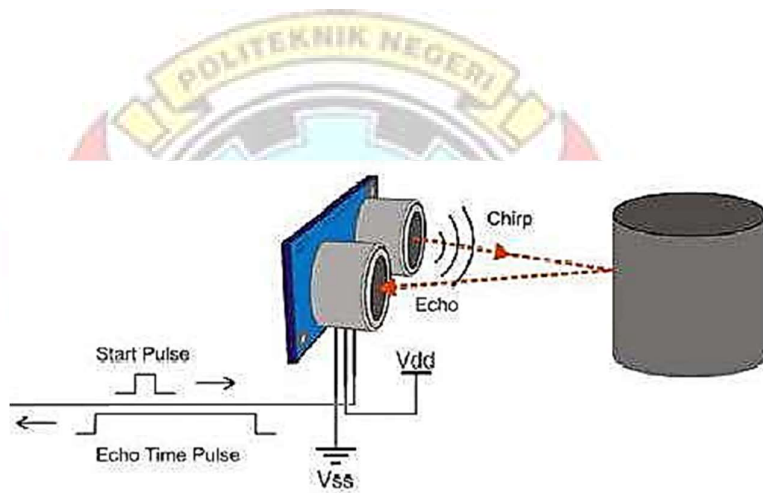
digunakan untuk mengukur jarak suatu benda yang bersifat padat, cair, dan gas.

#### 2.6.1.1 Pengertian Sensor Ultrasonik

Sensor adalah peralatan yang digunakan untuk merubah suatu besaran fisik menjadi besaran listrik sehingga dapat dianalisa dengan rangkain listrik tertentu (Budiarso: 2011). Sensor ultrasonik adalah sensor yang bekerja berdasarkan prinsip pantulan gelombang suara dan digunakan untuk mendeteksi keberadaan suatu objek tertentu di depannya, frekuensi kerjanya pada daerah di atas gelombang suara dari 40 KHz hingga 400 KHz. Sensor ultrasonik terdiri dari dari dua unit, yaitu unit pemancar dan unit penerima. Struktur unit pemancar dan penerima sangatlah sederhana, sebuah kristal *piezoelectric* dihubungkan dengan mekanik jangkar dan hanya dihubungkan dengan diafragma penggetar. Tegangan bolak-balik yang memiliki frekuensi kerja 40 KHz – 400 KHz diberikan pada plat logam. Struktur atom dari kristal *piezoelectric* akan berkontraksi (mengikat), mengembang atau menyusut terhadap polaritas tegangan yang diberikan dan ini disebut dengan efek piezoelectric.

Kontraksi yang terjadi diteruskan ke diafragma penggetar sehingga terjadi gelombang ultrasonik yang dipancarkan ke udara (tempat sekitarnya). Pantulan gelombang ultrasonik akan terjadi bila ada objek tertentu dan pantulan gelombang ultrasonik akan

diterima kembali oleh unit sensor penerima. Selanjutnya unit sensor penerima akan menyebabkan diafragma penggetar akan bergetar dan efek *piezoelectric* menghasilkan sebuah tegangan bolak-balik dengan frekuensi yang sama. Untuk lebih jelas tentang prinsip kerja dari sensor ultrasonik dapat dilihat prinsip dari sensor ultrasonic pada gambar berikut :



Gambar 2.11. Prinsip kerja sensor ultrasonik

Besar amplitudo sinyal elektrik yang dihasilkan sensor penerima tergantung dari jauh dekatnya objek yang dideteksi serta kualitas dari sensor pemancar dan sensor penerima. Proses yang dilakukan pada sensor ini menggunakan metode pantulan untuk menghitung jarak antara sensor dengan obyek sasaran. Jarak antara sensor tersebut dihitung dengan cara mengalikan setengah waktu yang digunakan oleh sinyal ultrasonik dalam perjalanannya dari rangkaian pengirim sampai diterima oleh rangkaian penerima,



dengan kecepatan rambat dari sinyal ultrasonik tersebut pada media rambat yang digunakannya, yaitu udara. Pantulan gelombang ultrasonik tersebut dapat dimanfaatkan untuk mengukur jarak antara sensor dan benda yang secara ideal dapat dihitung dengan rumus sebagai berikut :

$$s = 0,5 \cdot v \cdot t \dots\dots\dots(5)$$

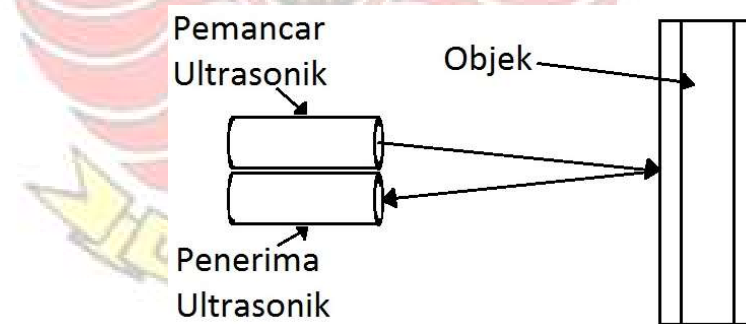
keterangan :

s = jarak objek dengan sensor (m)

v = cepat rambat suara pada medium yaitu 344 m/detik

t = waktu tempuh (detik)

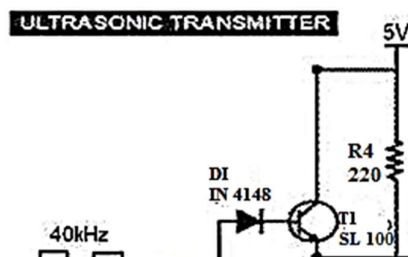
Prinsip pantulan dari sensor ultrasonik ini dapat dilihat pada gambar sebagai berikut:



Gambar 2.12 Prinsip pemantulan sensor ultrasonik

### A. Prinsip Kerja Pemancar Ultrasonik (*Transmitter*)

Pemancar Ultrasonik ini berupa rangkaian yang memancarkan sinyal sinusoidal berfrekuensi di atas 20 KHz menggunakan sebuah *transducer transmitter* ultrasonik.



Gambar 2.13 Pemancar ultrasonik transmitter

1. Sinyal 40 kHz dibangkitkan melalui mikrokontroler.
2. Sinyal tersebut dilewatkan pada sebuah resistor sebesar 3 K ohm untuk pengaman ketika sinyal tersebut membias maju rangkaian dioda dan transistor.
3. Kemudian sinyal tersebut dimasukkan ke rangkaian penguat arus yang merupakan kombinasi dari 2 buah dioda dan 2 buah transistor.
4. Ketika sinyal dari masukan berlogika tinggi (+5V) maka arus akan melewati dioda D1 (D1 on), kemudian arus tersebut akan membias transistor T1, sehingga arus yang akan mengalir pada kolektor T1 akan besar sesuai dari penguatan dari transistor.
5. Ketika sinyal dari masukan berlogika tinggi (0V) maka arus

akan melewati dioda D2 (D2 ON), kemudian arus tersebut akan membias transistor T2, sehingga arus yang akan mengalir pada kolektor T2 akan besar sesuai dari penguatan dari transistor.

6. Resistor R4 dan R6 berfungsi untuk membagi tegangan menjadi 2,5 V. Sehingga pemancar ultrasonik akan menerima tegangan bolak – balik dengan V adalah 5V (+2,5 V s.d -2,5 V).

### **B. Prinsip Kerja Penerima Ultrasonik (*Receiver*)**

Penerima Ultrasonik ini akan menerima sinyal ultrasonik yang dipancarkan oleh pemancar ultrasonik dengan karakteristik frekuensi yang sesuai. Sinyal yang diterima tersebut akan melalui proses filterisasi frekuensi dengan menggunakan rangkaian *band pass filter* (penyaring pelewat pita), dengan nilai frekuensi yang dilewatkan telah ditentukan.

Kemudian sinyal keluarannya akan dikuatkan dan dilewatkan ke rangkaian komparator (pembanding) dengan tegangan referensi ditentukan berdasarkan tegangan keluaran penguat pada saat jarak antara sensor kendaraan mini dengan sekat/dinding pembatas mencapai jarak minimum untuk berbelok arah. Dapat dianggap keluaran komparator pada kondisi ini adalah *high* (logika 1) sedangkan jarak yang lebih jauh adalah

*low* (logika 0). Logika-logika biner ini kemudian diteruskan ke rangkaian pengendali (mikrokontroler).



Gambar 2.14 Sensor Ultrasonik HC-SR04

### 2.6.2 Sensor Temperatur DS18B20

DS18b20 adalah sensor suhu digital yang di dalamnya sudah terdapat ADC (*Analog to Digital Converter*) dengan resolusi 12 bit. Sensor ini memiliki tingkat akurasi cukup tinggi yaitu  $\pm 0.5^{\circ}\text{C}$  di rentang suhu  $-10^{\circ}\text{C}$  sampai  $+85^{\circ}\text{C}$  dan secara keseluruhan dapat mengukur suhu dari  $-55^{\circ}\text{C}$  sampai  $+125^{\circ}\text{C}$ .

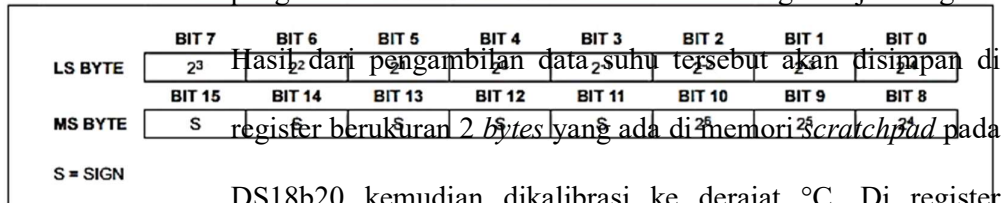
#### A. Prinsip Kerja Sensor Temperatur DS18B20

Sensor DS18b20 ini bekerja dengan konsep *direct-to-digital-temperature-sensors* dan resolusi ADC-nya bisa dikonfigurasi (9, 10, 11 atau 12 bit). Resolusi ADC ini akan berkaitan dengan kenaikan suhu tiap level analog seperti dibawah ini:

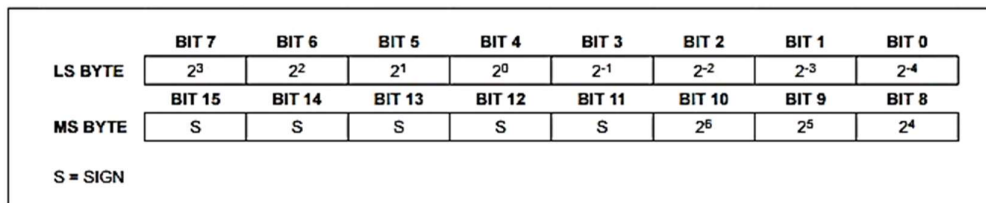
- Resolusi 9 *bit* = Kenaikan suhu tiap level analognya  $0.5^{\circ}\text{C}$

- Resolusi 10 *bit* = Kenaikan suhu tiap level analognya 0.25°C
- Resolusi 11 *bit* = Kenaikan suhu tiap level analognya 0.125°C
- Resolusi 12 *bit* = Kenaikan suhu tiap level analognya 0.0625°C

Semakin besar Resolusi ADC yang dipakai, semakin teliti sensor akan mendeteksi suhu disekitarnya. Secara *default*, DS18B20 saat kita hubungkan ke mikrokontroler resolusi ADC-nya langsung 12 *bit*, dalam hal ini mikrokontroler harus menginisiasi, *function command* ke DS18B20 untuk melakukan pengukuran suhu dan dikonversi dari analog menjadi digital.



Hasil dari pengambilan data suhu tersebut akan disimpan di register berukuran 2 bytes yang ada di memori *scratchpad* pada DS18B20 kemudian dikalibrasi ke derajat °C. Di register berukuran 2 bytes atau 16 bit terdapat *Sign-bits* (S) yang berfungsi untuk mengindikasikan apakah suhu dalam keadaan positif (makin panas) atau dalam keadaan negatif (makin dingin).



Gambar 2.15 Format Register Suhu pada DS18B20

Apabila suhu dalam keadaan positif maka nilai S=0 dan pada saat suhu dalam keadaan negatif, nilai S=1. Berikut ini adalah contoh *output* data digital dari sensor DS18B20 yang sebanding dengan nilai suhu dalam resolusi ADC 12 bit.

Tabel 2.1 *Output* Data Digital dari Sensor DS18B20

TEMPERATURE (°C)	DIGITAL OUTPUT (BINARY)	DIGITAL OUTPUT (HEX)
+125	0000 0111 1101 0000	07D0h
+85*	0000 0101 0101 0000	0550h
+25.0625	0000 0001 1001 0001	0191h
+10.125	0000 0000 1010 0010	00A2h
+0.5	0000 0000 0000 1000	0008h
0	0000 0000 0000 0000	0000h
-0.5	1111 1111 1111 1000	FFF8h
-10.125	1111 1111 0101 1110	FF5Eh
-25.0625	1111 1110 0110 1111	FE6Fh
-55	1111 1100 1001 0000	FC90h

Untuk berkomunikasi dengan mikrokontroler, sensor DS18B20 menggunakan antarmuka *1-wire*, sensor ini berarti hanya menggunakan 1 jalur data (dan *Ground* tentunya) untuk mengirim dan menerima data dari mikrokontroler.

### B. Spesifikasi Sensor Temperatur DS18B20

- Dimensi pipa *stainless steel*: diameter 0,6 cm dan panjang 3,6 cm.
- Panjang kabel *jumper*: 91 cm diameter 0.4 cm.
- Jarak ukur suhu: -55°C sampai +125°C.
- Akurasi:  $\pm 0.5^{\circ}\text{C}$  di rentang suhu -10°C sampai +85°C.
- Tegangan *input*: 3.0 V – 5 V.
- Antarmuka: *Max 1-Wire Bus Protocol*.

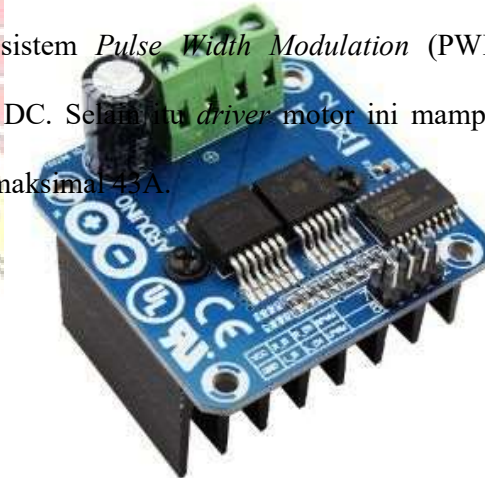
- Waktu konversi ADC 12-bit data suhu analog ke digital: 750 ms (maksimal).
- Dapat menggunakan 127 DS18B20 sekaligus secara paralel.



Gambar 2.16 Sensor Temperatur DS18B20

## 2.7 Driver Motor

*Driver* motor merupakan suatu rangkaian khusus yang memiliki fungsi untuk mengatur arah dan putaran kecepatan pada motor DC. *Driver* motor DC ini menerapkan sistem *Pulse Width Modulation* (PWM) untuk mengatur kecepatan motor DC. Selain itu *driver* motor ini mampu mengontrol motor DC dengan arus maksimal 3A.



Gambar 2.17 *Driver* Motor BTS7960

## 2.8 Driver Heater

*Driver Heater* yang digunakan terdiri dari *relay* 12 V sebagai *on/off heater* dari data kontrol 5 V Arduino, TRIAC BT 136, DIAC, potensiometer 500K, dan *relay* 5 VDC 220 VAC untuk *switching* tegangan. Dimana 3 *relay* yang dipasang pada *switching* tegangan untuk kondisi tegangan rendah, sedang dan tinggi dengan data kontrol 5 V dari Arduino sebagai pemicu untuk aktifnya tiap kondisi.

Pembuatan rangkaian *driver* ini dibuat menggunakan aplikasi *Easily Applicable Graphical Layout Editor* (EAGLE) dan dicetak dipapan *Printed Circuit Board* (PCB), serta pemasangan komponen-komponen elektronika yang dipakai seperti resistor, *Light Emitting Diode* (LED), pin *header*, potensiometer, *relay*, soket 2 pin, dan kapasitor.



Gambar 2.18 *Driver Heater*

## 2.9 Bahasa Pemograman MATLAB

Ada beberapa hal dasar yang perlu diketahui dalam Bahasa Pemograman MATLAB yaitu sebagai berikut



### 2.9.1 Definisi pemrograman MATLAB

Matlab merupakan bahasa pemrograman yang hadir dengan fungsi dan karakteristik yang berbeda dengan bahasa pemrograman lain yang sudah ada. Matlab merupakan bahasa pemrograman level tinggi yang dikhususkan untuk kebutuhan komputasi teknis, visualisasi dan pemrograman seperti komputasi matematik, analisis data, pengembangan algoritma, simulasi dan pemodelan dan grafik-grafik perhitungan. (Andrew knight,1999)

### 2.9.2 Matriks

Dapat diasumsikan bahwa didalam matlab setiap data akan disimpan dalam bentuk matriks. Dalam membuat suatu data matriks pada matlab, setiap isi data harus dimulai dari kurung siku '[' dan diakhiri dengan kurung siku tutup ']'. Untuk membuat variabel dengan data yang terdiri beberapa baris, gunakan tanda 'titik koma' (;) untuk memisahkan data tiap barisnya. Matlab menyediakan beberapa fungsi yang dapat kita gunakan untuk menghasilkan bentuk-bentuk matriks yang diinginkan. Fungsi-fungsi tersebut antara lain untuk membuat matriks yang semua datanya bernilai 0 (Zeros), matriks yang semua datanya bernilai 1 (Ones), matriks dengan data random dengan menggunakan distribusi uniform (Rand), matriks dengan data random dengan menggunakan distribusi normal (Randn), untuk menghasilkan matriks identitas (Eye). (Gunaidi Abdia Away, 2006)

### 2.9.3 M File

Di dalam matlab, kita dapat menyimpan semua script yang akan digunakan dalam file pada matlab dengan ekstensi M-File dapat dipanggil dengan memilih menu file->new->M-File. Fungsi M-file harus mengikuti beberapa aturan dan sejumlah sifat penting. Aturan-aturan dan sifat-sifat tersebut meliputi nama fungsi dan nama file harus identik misalnya anda membuat fungsi dengan nama pangkat maka anda memberi nama M-file anda pangkat juga. Baris komentar sampai dengan baris bukan komentar yang pertama adalah teks help yang ditampilkan jika anda meminta help dari fungsi yang anda buat. Setiap fungsi mempunyai ruang kerjanya sendiri yang berbeda dengan ruang kerja MATLAB. Satu-satunya hubungan antara ruang kerja matlab dengan variable-variabel dalam fungsi adalah variabel-variabel input dan output fungsi. Jika suatu fungsi mengubah nilai dalam suatu variabel input, perubahan itu hanya tampak dalam fungsi dan tidak mempengaruhi variable yang ada dalam ruang kerja MATLAB. Variabel yang dibuat oleh suatu fungsi tinggal hanya dalam ruang kerja fungsi. Jumlah dari argumen input dan output yang digunakan jika suatu fungsi dipanggil hanya ada dalam fungsi tersebut. Fungsi ruang kerja memuat jumlah argument input. Fungsi kerja nargout memuat jumlah argument output. Dalam praktek, variabel-variabel nargout dan nargin biasanya digunakan untuk mengeset variabel input standar dan menentukan variable output yang diperlukan user. Di dalam M-File, kita dapat menyimpan semua perintah dan

menjalankan dengan menekan tombol atau mengetikkan nama M-File yang kita buat pada *command window*.



### **BAB III**

## **METODE PENELITIAN**

### **3.1 Tempat dan Waktu Penelitian**

Skripsi ini dilaksanakan di Laboratorium Mekatronika dan Bengkel Mekanik Jurusan Teknik Mesin, Politeknik Negeri Ujung Pandang, Makassar pada bulan Februari sampai dengan bulan Agustus 2017.

### 3.2 Alat dan Bahan

Dalam metode perancangan ini, diperlukan sejumlah alat dan bahan untuk merakit alat ini sehingga tercipta sesuai dengan apa yang kita inginkan.

Adapun alat dan bahan yang

Tabel 3.1 Daftar Alat dan Bahan  
perancangan ini adalah sebagai berikut:

digunakan dalam

<b>Alat</b>	<b>Bahan</b>
Laptop/PC	Sensor Ultrasonik
Solder	Sensor Temperatur DS18B20
Penghisap Timah	Arduino AT Mega
Bor Tangan	Pompa Air
Mistar	<i>Heater</i>
Gerinda	Papan PCB
Kikir	Kawat timah
Lem Lilin	Pipa
<i>Toolset</i>	Larutan HCL & H2SO4
	Acrilic
	LCD 20x4
	Adapter 12 VDC
	<i>Push Button</i>
	<i>Eletronic Fans Kit Breadboard</i>

### 3.3 Metode Penelitian

Untuk memperoleh suatu alat yang baik dari segi mutu serta mempertimbangkan segi ekonomis, maka dibutuhkan langkah – langkah sebagai berikut:

#### 3.3.1 Studi literatur

Dalam perancangan alat ini, langkah awal yang dilakukan adalah

mencari sebanyak – banyaknya data serta informasi melalui berbagai media cetak maupun elektronik, dimana informasi tersebut harus relevan dengan alat yang akan dibuat.

### 3.3.2 Eksperimen dan Simulasi

Eksperimen terbagi atas tiga, yakni:

1. Mekanik

Pembuatan perangkat keras dari sistem kontrol level dan temperatur air yang telah dirancang.

2. Elektronik

Pembuatan rangkaian elektronika untuk *driver* pompa air dan *driver heater*.

3. Kontrol

Pembuatan perangkat lunak (program) pada sistem kontrol level air dan temperatur air.

Simulasi dilakukan untuk melihat respon sistem menggunakan *software* MATLAB/SIMULINK.

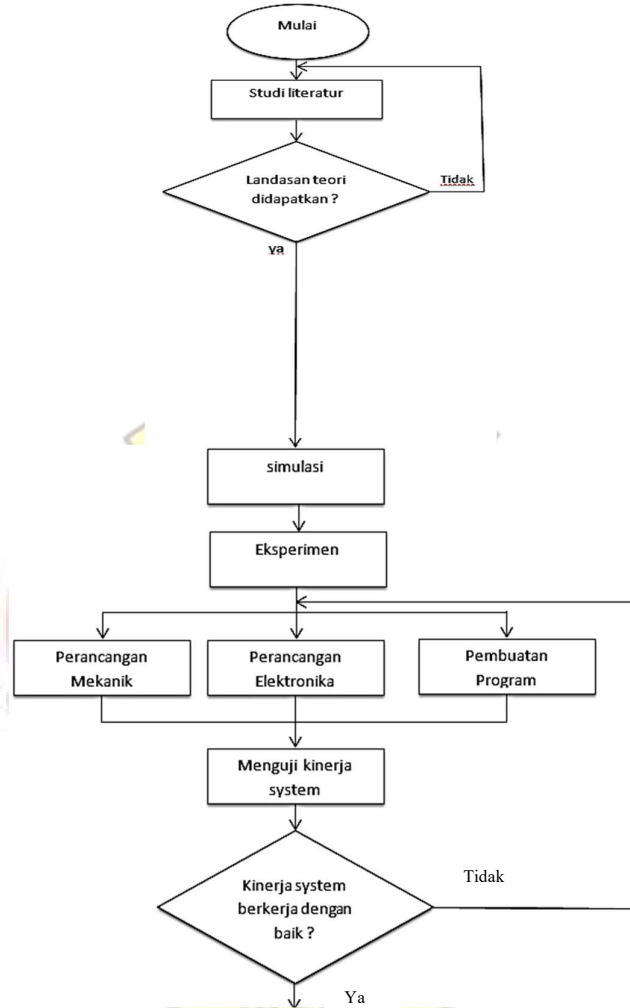
### 3.3.3 Pengujian dan Pengambilan Data

Pada proses ini sistem kontrol yang telah dipasang dan dirakit selanjutnya diuji dan melakukan pengambilan data.

### 3.3.4 Penarikan Kesimpulan

Penarikan kesimpulan akhir dari proses tugas akhir ini yang dilakukan setelah pengujian dan pengambilan data pada Skripsi ini.

Adapun diagram alir penelitian yang kami gunakan seperti pada gambar di bawah ini:



Gambar 3.1 Diagram Alir Penelitian

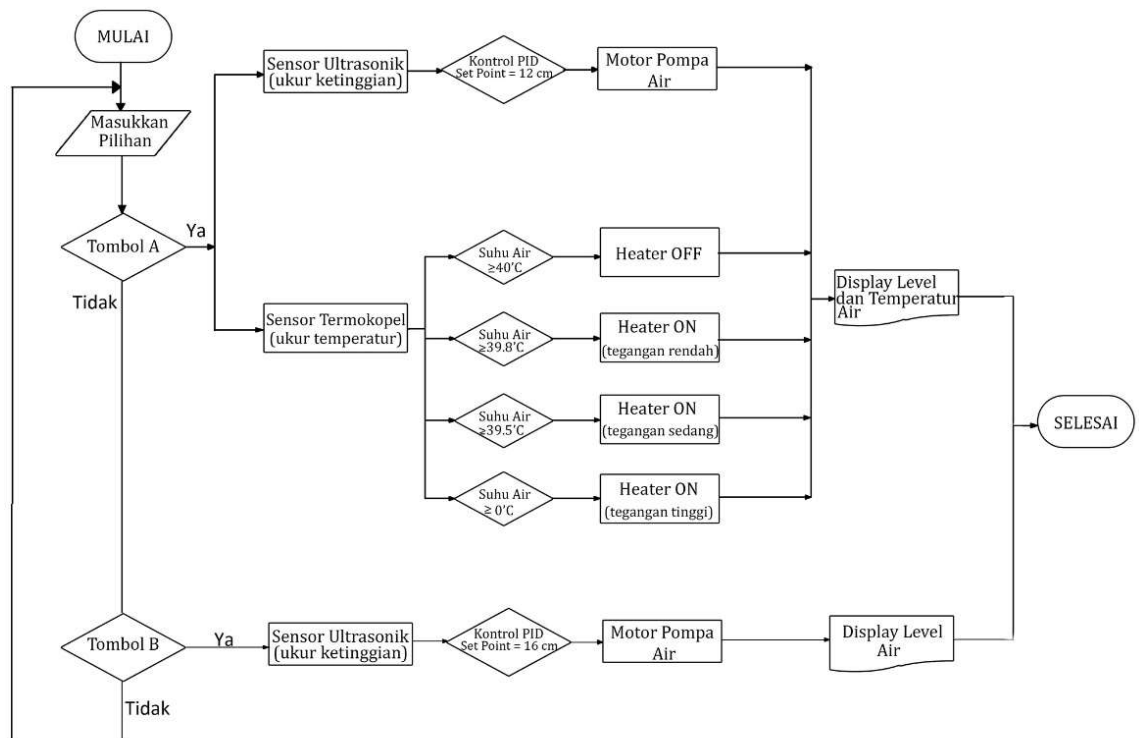
Lanjutan Gambar 3.1 Diagram Alir Penelitian



Gambar 3.1 Diagram Alir Penelitian

### 3.4 Diagram Alir Kontrol Level dan Temperatur Air

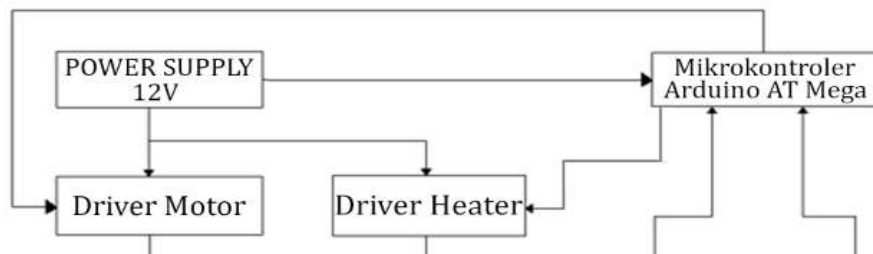
Pembuatan diagram alir ini dibuat untuk mempermudah perancangan program dan untuk membantu melacak kebenaran logika sebuah program. Adapun diagram alir pada rancang bangun kontrol level dan temperatur air berbasis mikrokontroler dengan menggunakan sensor ultrasonik, adalah sebagai berikut:



Gambar 3.2 Diagram Alir Kontrol Level dan Temperatur Air

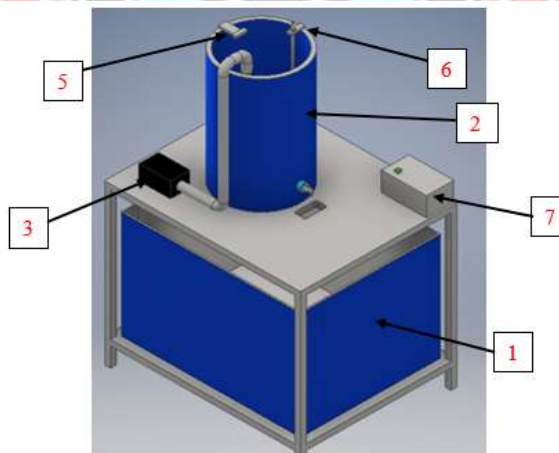
### 3.5 Diagram Blok Kontrol Level dan Temperatur Air

Tahap awal kontrol level dan temperatur air berbasis mikrokontroler menggunakan sensor ultrasonik dan temperatur ini diperlukan gambaran awal tentang bagaimana sistem kerja dari alat tersebut. Secara garis besar, menggunakan sensor ultrasonik, sensor temperatur, komputer, Arduino AT Mega, driver motor, pompa air, heater dan tangki penampungan.



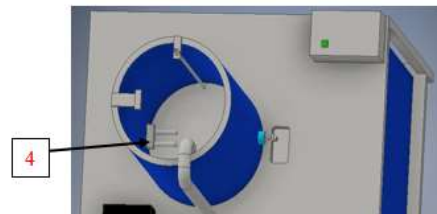
Gambar 3.3 Diagram Blok Kontrol Level dan Temperatur Air

### 3.6 Desain Alat Kontrol Level dan Temperatur Air



Lanjutan Gambar 3.4 Desain Alat kontrol level dan temperatur air

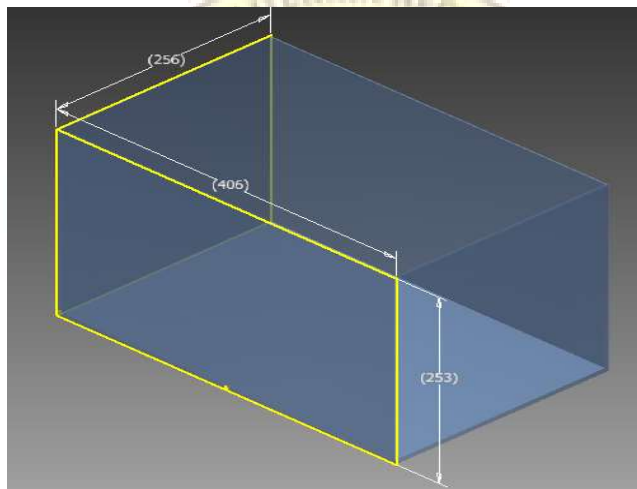
Gambar 3.4 Desain Alat kontrol level dan temperatur air



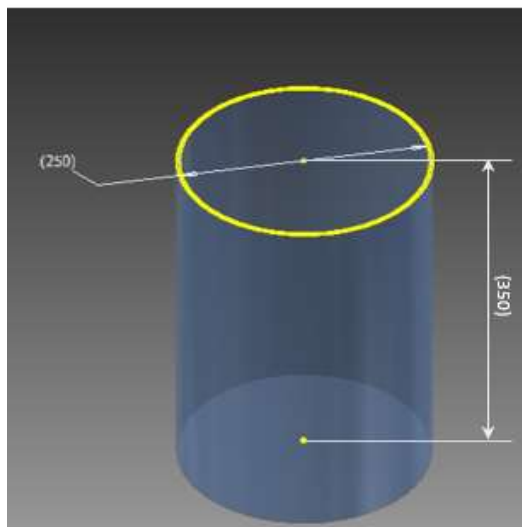
Keterangan :

1. Tangki A
2. Tangki B
3. Pompa Air
4. Heater
5. Sensor Ultrasonik
6. Sensor Temperatur
7. Driver Motor dan Heater





Gambar 3.5 Gambar Tangki A



G Gambar 3.7 Dudukan Tangki



## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1 HASIL**

##### **4.1.1 Hasil Perancangan**

## A. Mekanik

Berdasarkan hasil perancangan kontrol level dan temperatur air, maka alat yang dirancang mampu mengontrol level ketinggian dan temperatur air pada tangki penampungan. Hasil dari perancangan perangkat keras alat ini berupa 2 buah wadah, yang dimana wadah pertama berfungsi sebagai tempat penampungan air yang akan di kontrol yang berbentuk tabung dengan alas berdiameter 250 mm, dan tinggi 350 mm, sedangkan wadah yang kedua berfungsi sebagai tempat penampungan air utama dengan panjang 400 mm, lebar 250 mm, dan tinggi 250 mm, yang posisinya berada dibawah tangki pertama. Kedua wadah tersebut diletakkan pada sebuah dudukan yang terbuat dari besi *hollow* dan plat, dengan panjang 500 mm, lebar 350 mm, dan tinggi 470 mm. Selain itu terdapat box persegi panjang sebagai tempat mikrokontroler.

Hasil perancangan Mekanik dapat dilihat pada gambar di bawah ini:





Gambar 4.1 Hasil Perancangan Mekanik.

## **B. Elektronik**

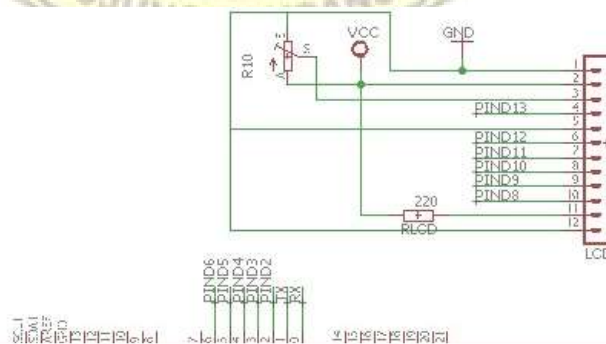
Pada rangkaian elektronika untuk kontrol level dan temperatur air ini, menggunakan beberapa komponen, seperti:

- Arduino AT Mega, sebagai pemroses data dan pengatur dari seluruh kegiatan sistem yang dibuat.
- Sensor Ultrasonik, sebagai pendeteksi ketinggian air dan hasilnya akan ditampilkan pada LCD dalam bentuk angka dan Komputer dalam bentuk Grafik.
- Pompa air DC, berfungsi untuk mengis air pada tangki penampungan.
- *Driver* motor BTS7960, berfungsi untuk mengatur arah putaran motor dan kecepatan motor.

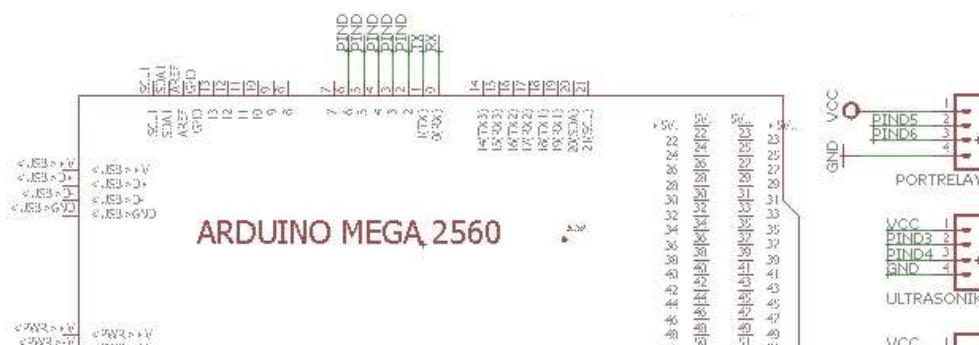
- Sensor Temperatur, sebagai pendeteksi suhu air dan akan ditampilkan pada LCD dalam bentuk angka dan Komputer dalam bentuk Grafik.
- *Heater*, berfungsi untuk memanaskan air.
- *Driver heater / switching* tegangan, Dimana 3 *relay* yang dipasang pada *switching* tegangan untuk kondisi tegangan rendah, sedang dan tinggi dengan data kontrol 5 V dari Arduino sebagai pemicu untuk aktifnya tiap kondisi.

Pembuatan rangkaian elektronika untuk kontrol level dan temperatur air dibuat di aplikasi EAGLE dan dicetak dipapan PCB, serta pemasangan komponen-komponen elektronika yang dipakai seperti resistor, *LED*, *pin header*, potensiometer, relay, soket 2 pin, dan kapasitor.

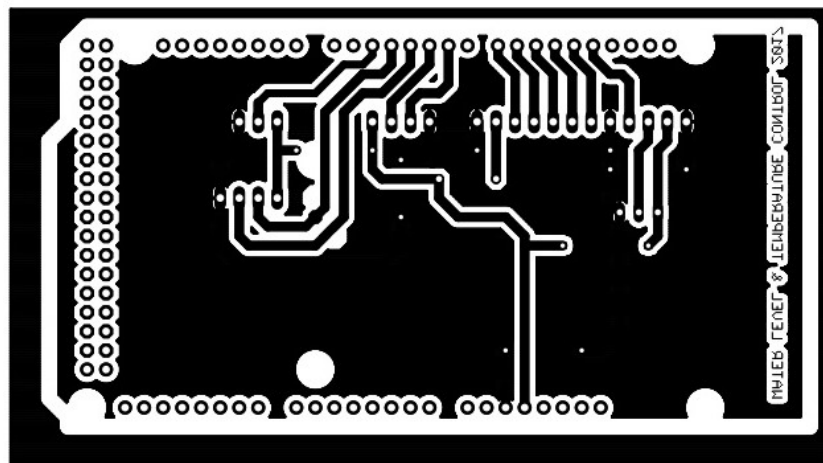
Hasil perancangan elektronik dapat dilihat pada gambar di bawah ini:



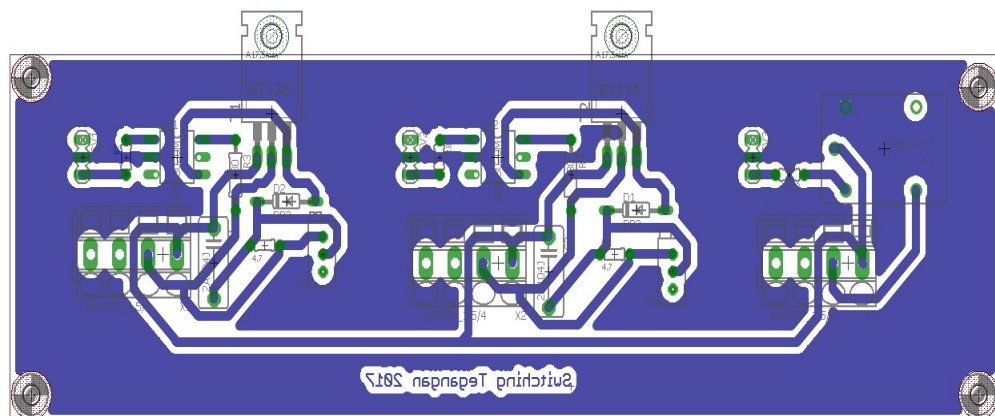
Lanjutan Gambar 4.2 *Schematic* Rangkaian Kontrol Level dan Temperatur Air



Gambar 4.2 *Schematic* Rangkaian Kontrol Level dan Temperatur Air.



Gambar 4.3 *layout* Rangkaian Kontrol Level dan Temperatur Air.



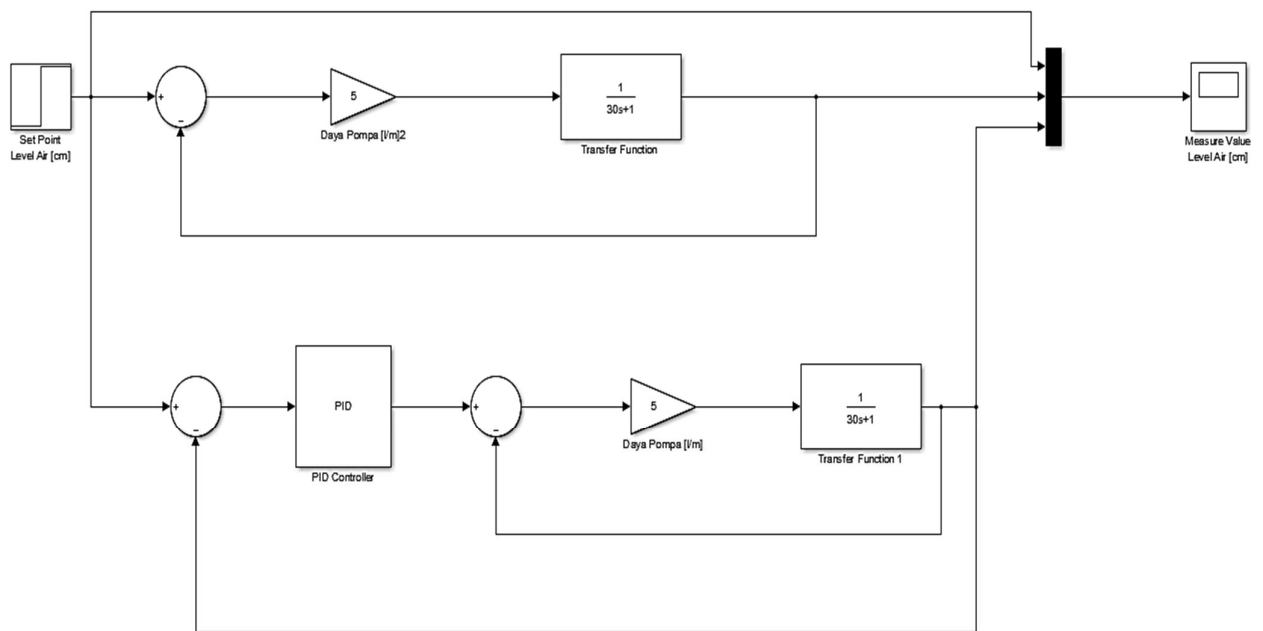
4.1.2 Hasil Simulasi **Gambar 4.4** *layout Driver Heater*.

Simulasi dikerjakan menggunakan *software* MATLAB /

Simulink, adapun hasil dari simulasi tersebut, sebagai berikut:

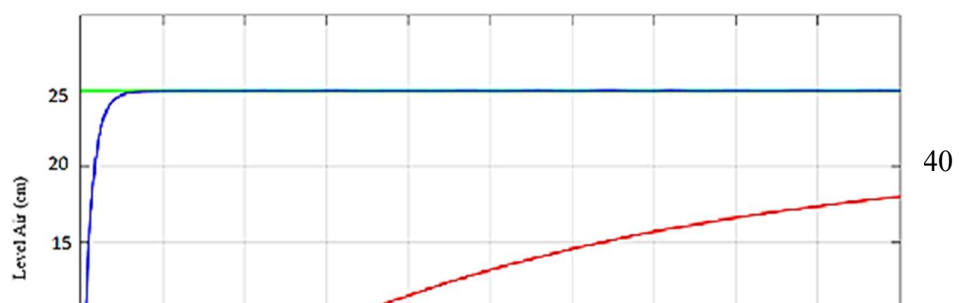
### A. Kontrol Level Air

Pada *setpoint* atau tinggi air yang diinginkan kami *set* 25 cm, kemudian pada kontrol PID kami *set* kontrol P = 95, I = 11, D = 3.5.



Gambar 4.5 Diagram Blok Simulink Kontrol Level Air.

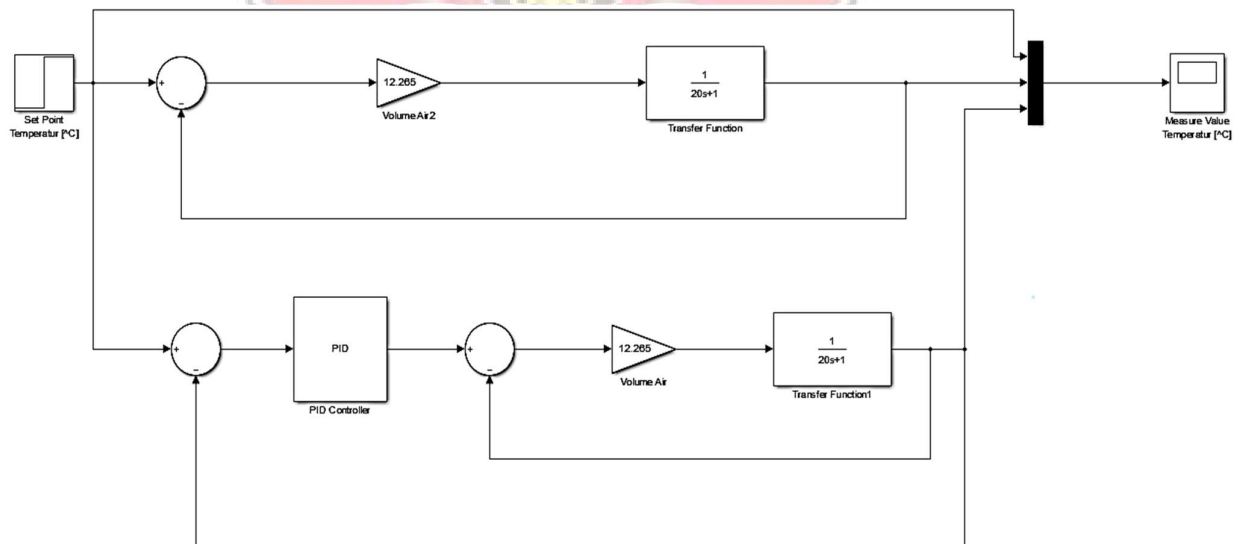
Pada grafik terlihat garis yang berwarna hijau adalah *setpoint*, biru adalah sistem dengan kontrol PID, dan merah adalah sistem tanpa kontrol PID.



Gambar 4.6 Grafik Simulink Kontrol level Air.

### B. Kontrol Temperatur Air

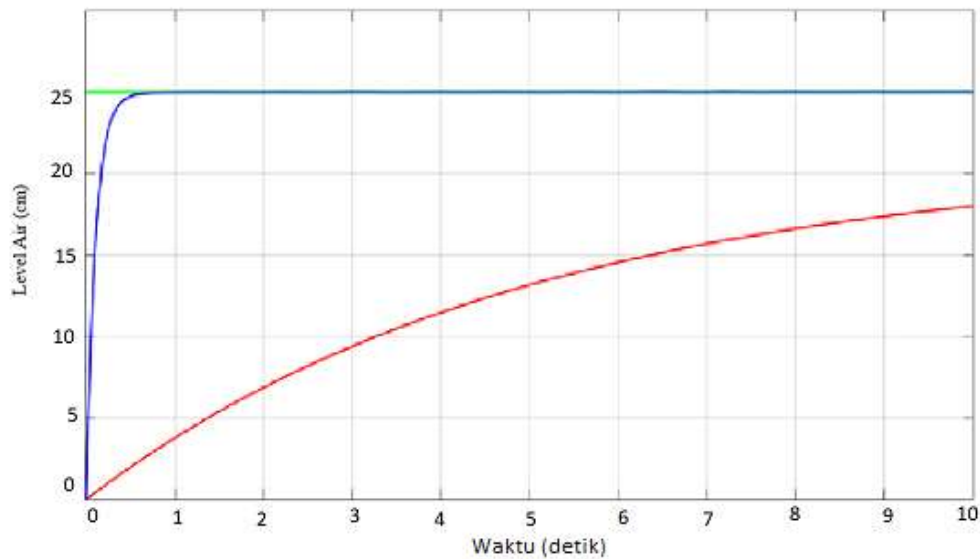
Pada *setpoint* atau temperatur yang diinginkan kami *set*  $60^{\circ}\text{C}$ , kemudian pada kontrol PID kami *set* kontrol  $P = 60$ ,  $I = 16,50$ ,  $D = 2$ .



Gambar 4.7 Diagram Blok Simulink Kontrol Temperatur Air.



Pada grafik terlihat garis yang berwarna hijau adalah *setpoint*, biru adalah sistem dengan kontrol PID, dan merah adalah sistem tanpa kontrol PID.



Gambar 4.8 Grafik Simulink Kontrol Temperatur Air.

### 4.1.3 Hasil Pemograman

Untuk pemograman sistem kontrol level dan temperatur air kami menggunakan dua *software* yaitu Arduino dan MATLAB. Adapun *listing* programnya dapat dilihat pada lampiran halaman.

## 4.2 PEMBAHASAN

### 4.2.1 Proses Kerja Alat

Pada saat alat di jalankan, akan muncul pilihan pada LCD. Dimana tombol A untuk mengontrol suhu dan level air, dan tombol B untuk mengontrol level air saja.

Pada saat tombol A di tekan, maka sensor ultrasonic akan membaca ketinggian air kemudian dikirimkan ke arduino lalu diproses untuk mengontrol pompa air, dimana di dalam program Arduino juga terdapat kontrol PID. Pada saat level air mencapai 12 cm, maka pompa akan mati, secara bersamaan pada saat tombol A di tekan, maka sensor temperatur akan aktif untuk membaca suhu air kemudian akan di kirim ke Arduino untuk mengontrol *heater*. *Heater* akan berfungsi untuk memanaskan air pada saat level ketinggian air lebih tinggi dari 8 cm dan temperatur yang ingin di capai adalah 40 derajat celsius.

Pada saat tombol B di tekan, maka yang akan di kontrol hanya level air, di mana prinsip kerjanya sama dengan pada saat tombol A di tekan, tetapi hanya sensor ultrasonik saja yang bekerja untuk mengirimkan data ke Arduino dan akan mengontrol pompa air untuk mencapai level air, dimana ketinggian air yang di inginkan adalah 16 cm.

#### 4.2.2 Cara Pengoperasian Alat

1. Hubungkan *Power Supply / Adapter* 12 VDC dengan sumber 220 VAC.
2. Hubungkan *driver heater* dengan sumber 220 VAC.
3. Koneksikan mikrokontroler ke komputer dengan menggunakan kabel *interface*.

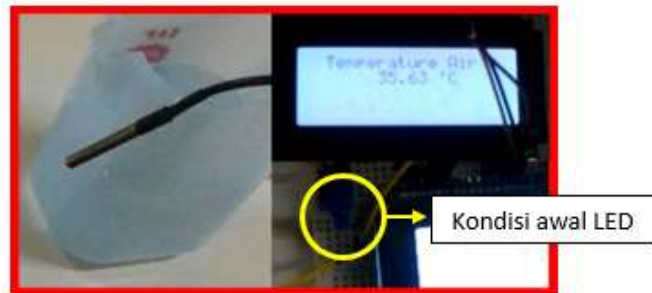
4. Buka aplikasi Arduino dan Matlab, kemudian buka program yang telah dibuat.
5. Upload program Arduino tersebut ke mikrokontroler.
6. Pilih salah satu tombol pada box kontroler sesuai dengan apa yang anda ingin kontrol (tombol A untuk mengontrol suhu dan level air, dan tombol B untuk mengontrol level air saja).

#### **4.2.3 Pengujian Kinerja Sistem**

Pengujian merupakan langkah yang digunakan untuk mengetahui sejauh mana kesesuaian antara rancangan dengan kenyataan pada alat yang telah dibuat, apakah sudah sesuai dengan yang diharapkan atau belum. Pengujian alat ini juga berguna untuk mengetahui tingkat kinerja dari alat tersebut. Setelah dilakukan pengujian, maka hendaknya melakukan pengujian ukuran atau analisa terhadap apa yang diuji untuk mengetahui keberhasilan dari alat yang dibuat dalam tugas akhir ini.

##### **4.2.3.2 Pengujian Sensor Temperatur DS18B20**

Pengujian dilakukan dengan memberi program pengaktifan sensor pada mikrokontroler kemudian sensor diletakkan pada dua kondisi suhu yang berbeda, kondisi awal adalah suhu normal kemudian kondisi kedua adalah suhu yang diinginkan. Pengujian sensor temperatur dapat dilihat pada gambar berikut.



Gambar 4.9 Kondisi Awal Sensor Temperatur.



Gambar 4.10 Kondisi Saat Mencapai Suhu yang Diinginkan.

#### 4.2.3.4 Pengujian *Heater*

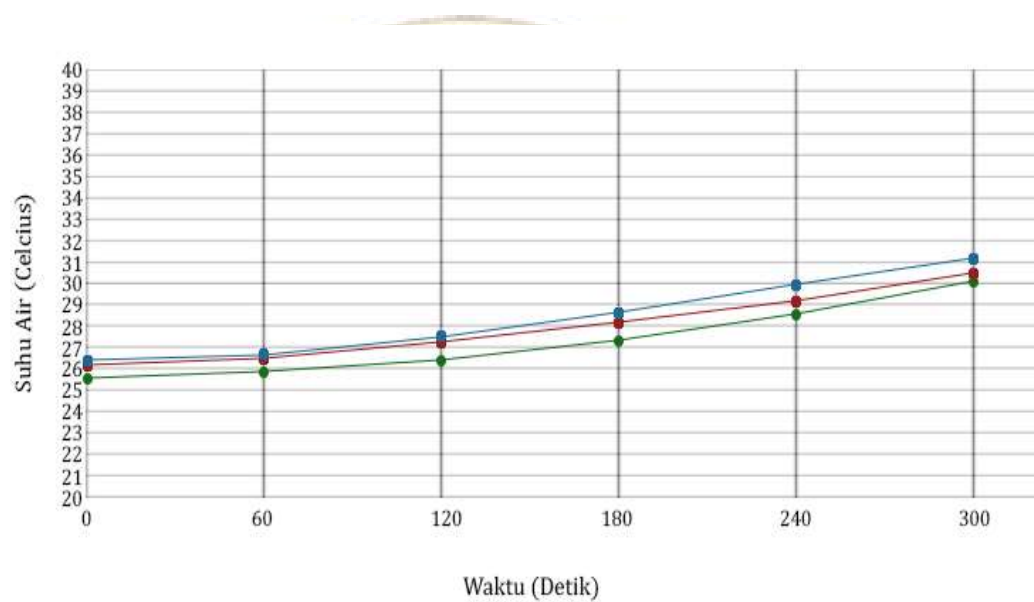
Pada pengujian ini, *heater* diuji agar dapat mengetahui berapa suhu rata-rata air yang di panaskan setiap menit dengan tegangan 220v.

Rata-rata =

$$\frac{(t5 - t4) + (t4 - t3) + (t3 - t2) + (t2 - t1) + (t1 - t0)}{5}$$

Tabel 4.1 Data Hasil Pengujian *Heater*

Level (cm)	Waktu (s)					Suhu (celcius)						
	I	II	III	IV	V	t0	t1	t2	t3	t4	t5	Rata-rata
12	60	120	180	240	300	25,69	25,82	26,27	27,29	28,53	30,02	0,86
10						26,02	26,49	27,25	28,18	29,26	30,45	0,88
8						26,17	26,86	27,69	28,78	29,96	31,25	1,01



Gambar 4.11 Grafik Hasil Pengujian *Heater*

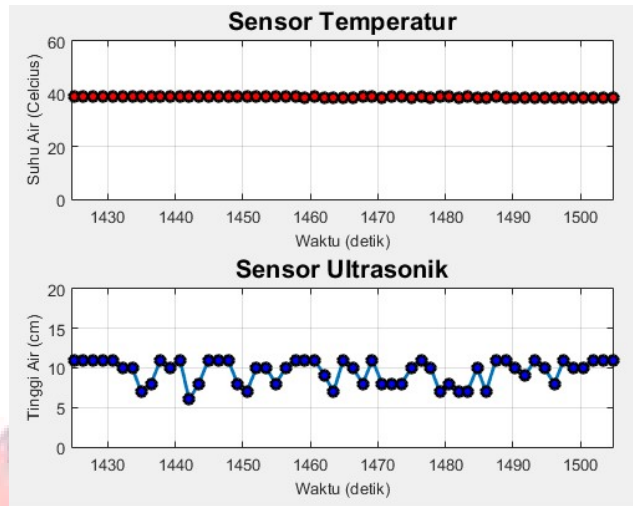
- = Level Ketingian Air 8 cm
- = Level Ketingian Air 10 cm
- = Level Ketingian Air 12 cm

#### 4.2.3.5 Pengujian Grafik Real Time

Adapun pengujian grafik real time dilakukan untuk mengetes grafik yang di tampilkan di komputer

berdasarkan dengan tujuan penelitian ini di lakukan.

Pengujian grafik dapat di lihat pada gambar berikut.



Gambar 4.12 Pengujian Grafik Real Time

#### 4.2.3.6 Analisa Hasil Pengujian

Tahap pengujian dilakukan sebanyak 4 tahap, yang pertama pengujian terhadap sensor temperatur, pengujian terhadap heater, dan Pengujian sistem menggunakan kontrol PID. Pada percobaan pengujian sensor temperatur apabila suhu yang di baca oleh sensor lebih besar dari 60 derajat celcius, maka LED akan menyala. Data dari hasil pengujian, pada gambar 4.5 suhu yang dibaca sensor temperatur adalah 35,63 derajat celcius dan LED dalam keadaan mati, sedangkan pada gambar 4.6 suhu yang dibaca adalah 61,19 derajat celcius dan LED dalam

keadaan menyala. dapat disimpulkan bahwa sensor termokopel bekerja dengan baik dan dapat mengontrol LED sesuai dengan kondisi suhu yang dibaca.

Pada percobaan pengujian heater berdasarkan data hasil pengujian heater, diperoleh hasil bahwa kekuatan memanaskan heater berbeda-beda sesuai dengan tegangan dan level air yang berada dalam tangki. Bisa kita lihat pada tabel 4.1, waktu yang diberikan di setiap percobaan sama yaitu 5 menit, namun panas yang di hasilkan berbeda-beda. Pada saat level air 8 cm dan tegangan 220 volt *heater* paling cepat memanaskan, sedangkan pada saat level air 12 cm dan tegangan yang diberikan 150 volt *heater* sangat lama memanaskan.

Berdasarkan simulasi kontrol PID menggunakan matlab, dapat dilihat perbandingan antara kontrol tanpa PID dan kontrol menggunakan PID. Pada saat sistem tidak menggunakan kontrol PID, sistem sangat lambat mencapai titik yang diinginkan, sebaliknya ketika sistem menggunakan kontrol PID, sistem lebih cepat mencapai titik yang diinginkan.

## BAB V

### PENUTUP

#### 5.1. Kesimpulan

Berdasarkan perancangan dan pengujian sistem dapat diambil kesimpulan sebagai berikut:

1. Sistem kontrol level ketinggian dan temperatur air ini terbukti dapat digunakan sebagai salah satu alternatif pengontrolan level ketinggian dan temperatur air pada tangki penampungan. Dengan dikendalikan menggunakan mikrokontroler dan sensor ultrasonik sebagai pendeteksi level ketinggian air, serta sensor temperatur DS18B20 sebagai pendeteksi suhu air.
2. Dari pengujian yang dilakukan, kerja dari rangkaian kontrol pengendali motor / pompa air DC, dan *heater* berfungsi dengan baik dalam mengontrol level ketinggian dan temperatur air pada tangki penampungan dan dapat dimonitor pada layar komputer dengan menggunakan *software* MATLAB.

#### 5.2. Saran

Berdasarkan hasil penelitian yang telah dilakukan, diperoleh beberapa hal yang dapat dijadikan sebagai saran untuk dilakukan pengembangan lebih lanjut yaitu:



1. Alat pengontrolan level dan temperatur air ini sebaiknya dapat dikembangkan dengan menggunakan katub solenoid agar keluaran air dapat otomatis.
2. Pengontrolan temperatur air ini dapat dikembangkan dengan menggunakan kontrol PID.
3. Menambahkan *keypad* pada box kontrol agar pengguna dari alat dapat memasukkan nilai level dan temperatur air sesuai yang diinginkan.



## DAFTAR PUSTAKA

- Budiarso Z, Nurrahharzo E. (2011). *Sistem Monitoring Ketinggian Air Bendungan Berbasis Mikrokontroler*. Jurnal Dinamika Informatika.
- Johnson, D. Curtis.(1989). "*Teknik Instrumentasi Kontrol Proses*". Edisi I, diterjemahkan oleh PEDC. Bandung.
- Gultom, Iwan Primus (2013). "Makalah MATLAB", (Online), (<http://iwangoeltom.blogspot.co.id/2013/08/makalah-mat-lab.html>). Diakses pada tanggal 12 April 2017.
- K.G. Suastika, M. Nawir, P. Yunus. (2013). "Sensor Ultrasonik Sebagai Alat Pengukur Kecepatan Aliran Udara Dalam Pipa". Dalam jurnal Pendidikan Fisika Indonesia. Universitas palangkaraya.
- Puluhulawa, Irman dan Mangku Alam (2016). "Rancang Bangun Kontrol Level Air Berbasis Mikrokontroler Menggunakan Sensor Ultrasonik". Skripsi. Makassar: Politeknik Negeri Ujung Pandang.
- Q-ELECTRONICS*, (2016). "Sensor #4 : Cara Menggunakan Sensor DS18B20 dengan NodeMCU" (Online), (<http://www.qelectronics.id/tutorial-sensor-ds18b20-dengan-nodemcu/>). Diakses pada Tanggal 17 Juli 2017.
- Rianto, Heru. (2010). "Pengaman Parkir Mobil menggunakan sensor Ultrasonik Berbasis Mikrokontroler". Skripsi. Fakultas Teknik Universitas Negeri Semarang. Semarang.
- Surindra, M. Deni. (2012) "Sistem Kontrol Level Air Pada Feed Water Tank Menggunakan Mikrokontroler ATmega8535 Dengan Delphi Sebagai Media Tampilannya". Jurusan Teknik Mesin Politeknik Negeri Semarang.
- Susanti, Lely (2013). "Laporan Tugas Rancang Dasar Elektronika *Water Level Control*" (Online), (<https://www.academia.edu/6520482/Makalah>

Tugas\_Rancang\_Water\_Level\_Control\_). Diakses pada tanggal 12 April 2017.

Setyo Deny, (2013). “Modul Kontrol Temperatur Air Berbasis PID Menggunakan MATLAB” (Online), ([http://www.academia.edu/9629321/MODUL\\_KONTROL\\_TEMPERATUR\\_AIR\\_BERBASIS\\_PID\\_MENGGUNAKAN\\_MATLAB\\_SIMULINK\\_Disusun\\_Oleh](http://www.academia.edu/9629321/MODUL_KONTROL_TEMPERATUR_AIR_BERBASIS_PID_MENGGUNAKAN_MATLAB_SIMULINK_Disusun_Oleh)). Diakses pada tanggal 17 Juli 2017.

Yuono, (2006). “Sistem Kontrol Temperatur Air Pada Proses Pembudidayaan Ikan Berbasis Mikrokontroler, (Online), ([aresearch.upi.edu/operator/upload/s\\_fis\\_0708829\\_chapter1.pdf](http://aresearch.upi.edu/operator/upload/s_fis_0708829_chapter1.pdf)), diakses pada tanggal 13 Maret 2017.





# LAMPIRAN



**LAMPIRAN 1**  
**Spesifikasi Arduino Mega**

## 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.



## Technical Specification

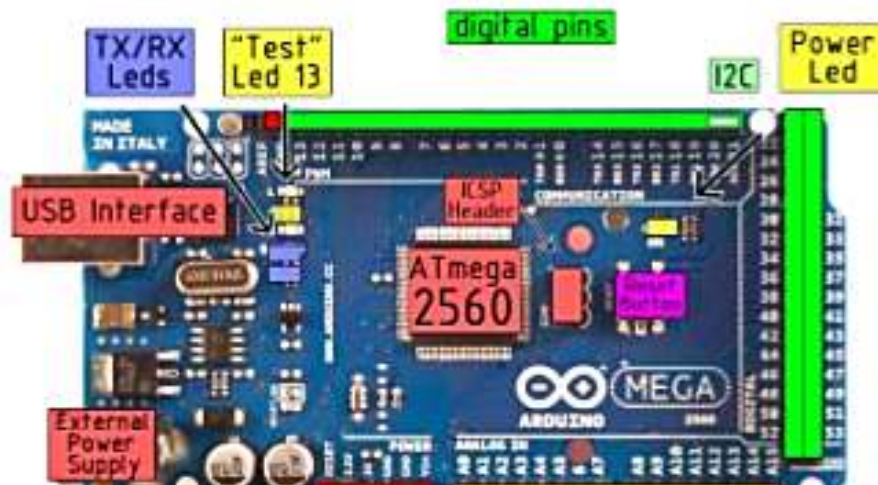


EAGLE files: [arduino-mega2560-reference-design.zip](#) Schematic: [arduino-mega2560-schematic.pdf](#)

### 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



## 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

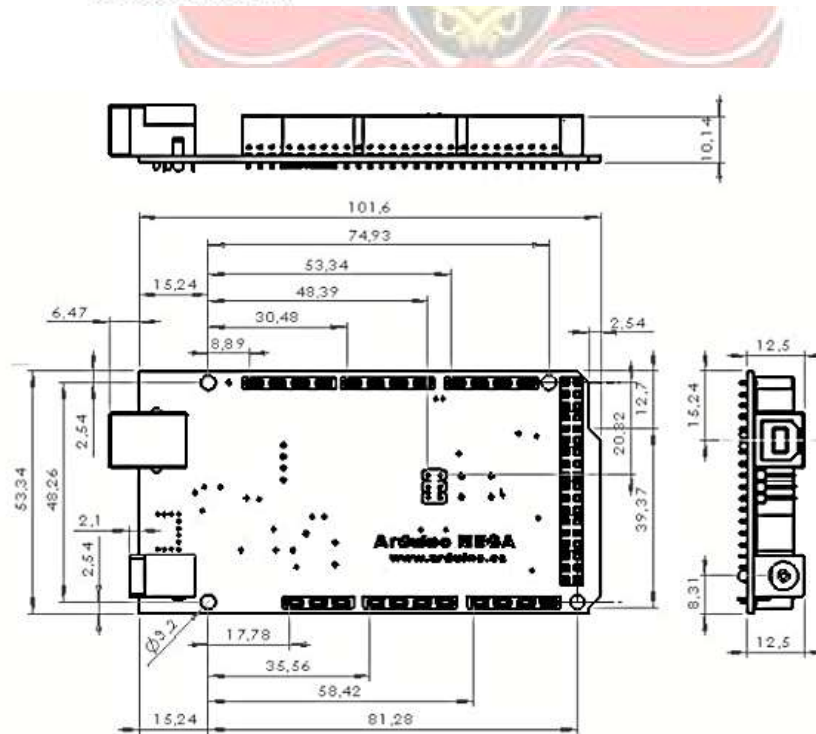
Each of the 54 digital pins on the Mega can be used as an Input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In 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: 16 (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 (SS), 53 (SCK), 54 (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 Diecimila.
- **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 it is 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.

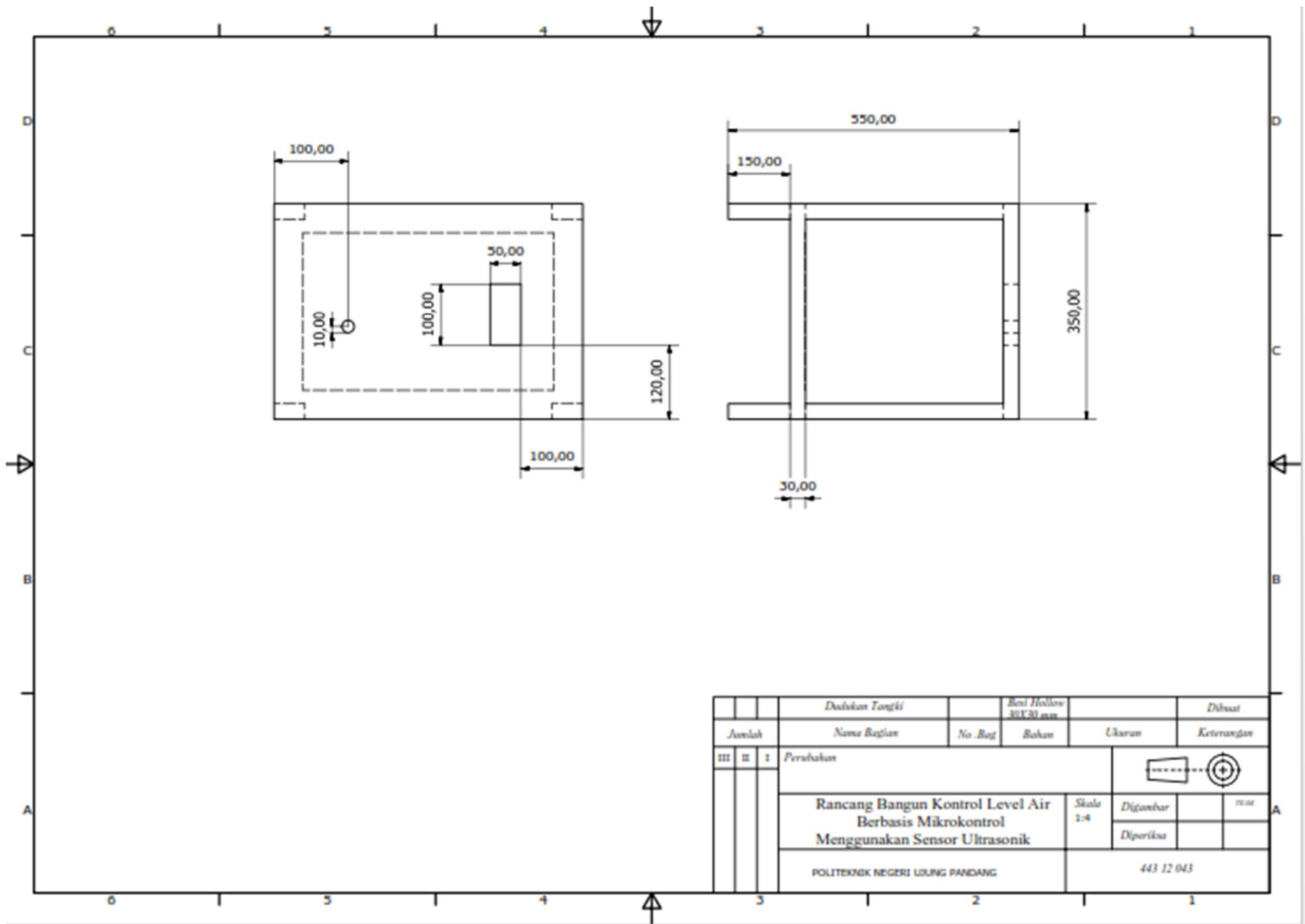




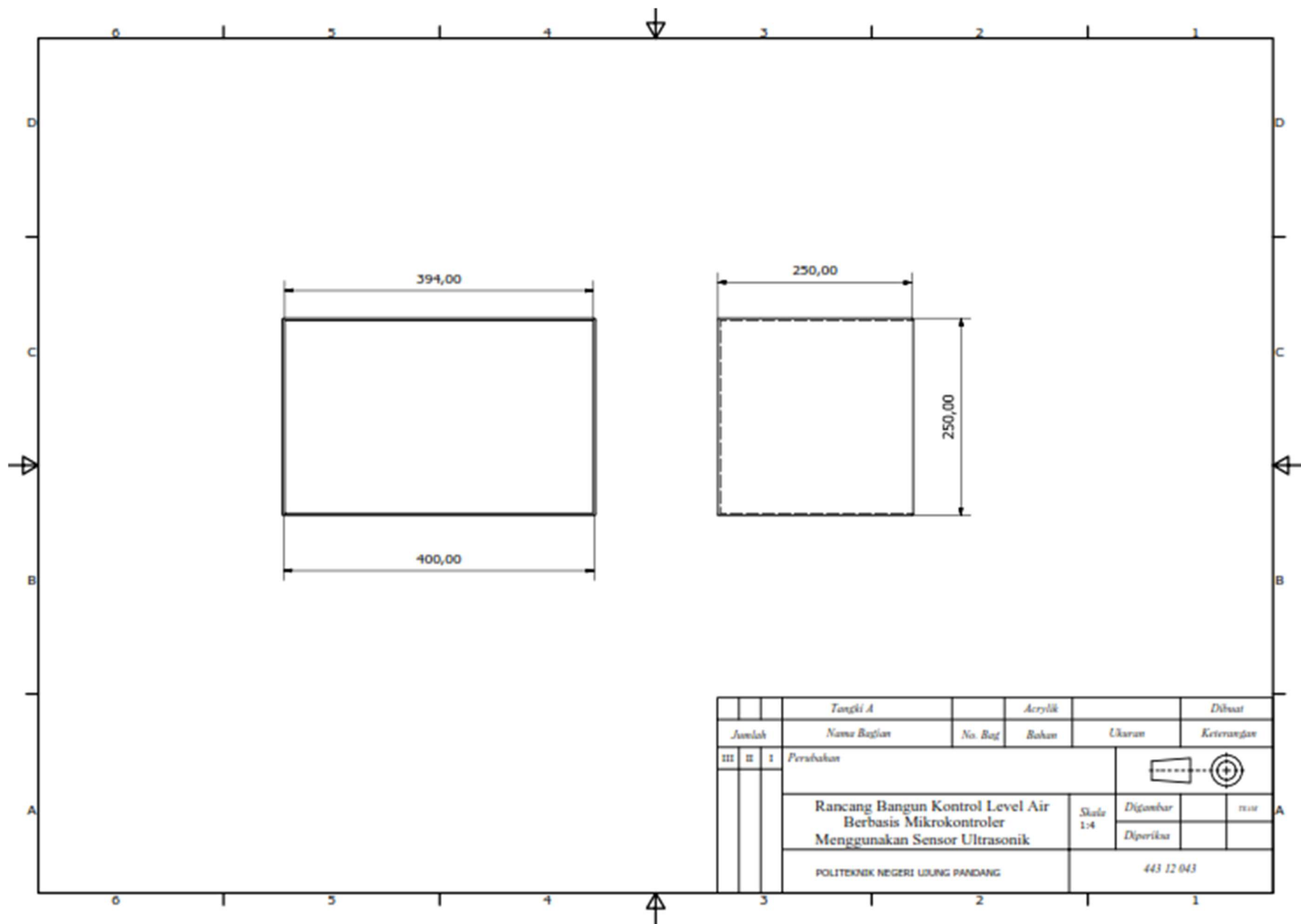


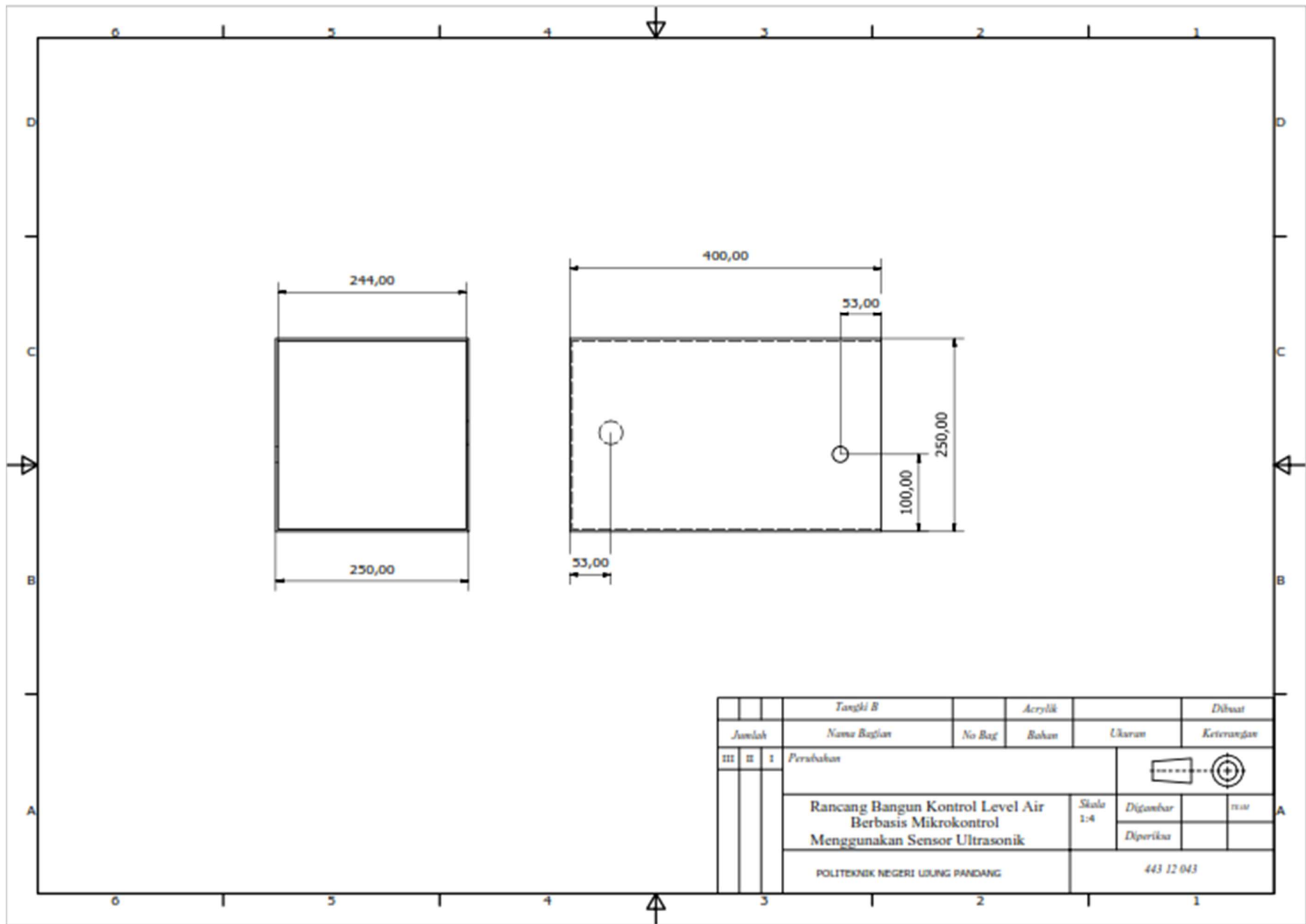
## **LAMPIRAN 2**

### **Desain Alat**



Jumlah			Dudukan Tangki	Besi Hollow 30X30 mm	Dibuat		
III	II	I	Nama Bagian	No. Bag	Bahan	Ukuran	Keterangan
		1	Perubahan				
			Rancang Bangun Kontrol Level Air Berbasis Mikrokontrol Menggunakan Sensor Ultrasonik		Skala 1:4	Digambar	10.02
			POLITEKNIK NEGERI LUUNG PANDANG		443 12 043		









**LAMPIRAN 3**  
**Program Kontrol Level dan Temperatur Air**



## Listing Program MATLAB

```
clear
clc

%Inisialisasi serial dan grafik
serialPort = 'COM8'; % isi dgn COM yg
sesuai
judulGrafik = 'Grafik Kontrol Level Air'; % Judul grafik
xLabel1 = 'Waktu (detik)'; % x-axis label 1
yLabel1 = 'Suhu Air (Celcius)'; % y-axis label 1
xLabel2 = 'Waktu (detik)'; % x-axis label 2
yLabel2 = 'Tinggi Air (cm)'; % y-axis label 2
plotGrid = 'on'; % aktifkan grid
min = [0,0]; % minimum axis-y
max = [60,20]; % maksimum axis-y
lebarScroll = 80; % display data
delay = .05; % waktu cuplik

%Inisialisasi variabel
waktu = 0;
data = 0;
cacah = 0;

%Persiapkan grafik
sub(1) = subplot(2,1,1); % subplot atas
sub(2) = subplot(2,1,2); % subplot tengah

plotGraph(1) = plot(sub(1),waktu,data,'-o',...
'LineWidth',2,...
'MarkerSize',6,...
'MarkerEdgeColor','k',...
'MarkerFaceColor','r');
plotGraph(2) = plot(sub(2),waktu,data,'-o',...
'LineWidth',2,...
'MarkerSize',6,...
'MarkerEdgeColor','k',...
'MarkerFaceColor','b');

title(sub(1),'Sensor Temperatur','FontSize',15);
title(sub(2),'Sensor Ultrasonik','FontSize',15);

xlabel(sub(1), xLabel1,'FontSize',10);
xlabel(sub(2), xLabel2,'FontSize',10);

ylabel(sub(1), yLabel1,'FontSize',10);
ylabel(sub(2), yLabel2,'FontSize',10);

axis(sub(1),[0 10 min(1) max(1)]);
axis(sub(2),[0 10 min(2) max(2)]);

grid(sub(1),plotGrid);
grid(sub(2),plotGrid);
```



```

%Buka komunikasi melalui port COM
delete(instrfindall);
s = serial(serialPort);
disp('Tutup jendela grafik untuk mengakhiri logger');
fopen(s);

tic                                     %aktifkan deteksi waktu
while ishandle(plotGraph)             %Terus looping semasih plot
aktif

nilaiInput = fscanf(s);
nilaiInput = strsplit(nilaiInput, ',');
dataSensor(1) = str2double(cell2mat(nilaiInput(1)));
dataSensor(2) = str2double(cell2mat(nilaiInput(2)));

%Pastikan data yg diterima benar
if(~isempty(dataSensor(1)) && isfloat(dataSensor(1)))
    cacah = cacah + 1;
    waktu(cacah) = toc;                %ambil waktu saat ini

    data1(cacah) = dataSensor(1);      %ambil data saat ini utk x
    data2(cacah) = dataSensor(2);      %ambil data saat ini utk y

    %===== grafik 1 untuk kelembaban =====
    %Set Axis sesuai dengan nilai lebarScroll
    if(lebarScroll > 0)
        set(plotGraph(1), 'XData', waktu(waktu > waktu(cacah) -
lebarScroll), ...
            'YData', data1(waktu > waktu(cacah) - lebarScroll));
        axis(sub(1), [waktu(cacah) - lebarScroll waktu(cacah) min(1)
max(1)]);
    else
        set(plotGraph(1), 'XData', waktu, 'YData', data1);
        axis(sub(1), [0 waktu(cacah) min(1) max(1)]);
    end

    %===== grafik 2 untuk suhu =====
    %Set Axis sesuai dengan nilai lebarScroll
    if(lebarScroll > 0)
        set(plotGraph(2), 'XData', waktu(waktu > waktu(cacah) -
lebarScroll), ...
            'YData', data2(waktu > waktu(cacah) - lebarScroll));
        axis(sub(2), [waktu(cacah) - lebarScroll waktu(cacah) min(2)
max(2)]);
    else
        set(plotGraph(2), 'XData', waktu, 'YData', data1);
        axis(sub(2), [0 waktu(cacah) min(2) max(2)]);
    end

    %Beri waktu sesaat utk Update Plot
    pause(delay);
end
end

```

```
% Tutup serial port dan delete variabel yg sudah terpakai  
fclose(s);  
clear;  
disp('Logger berakhir...');
```



## Listing Program Arduino

```
int matlabData;
//////////////////////////////////PIDLEVEL//////////////////////////////////
#include <PID_v1.h>

#define PIN_OUTPUT 7 /// pin pwm pompa
double Setpoint, Input, Output;
double Kp=150, Ki=0, Kd=0;
PID myPID(&Input, &Output, &Setpoint, Kp, Ki, Kd, DIRECT);

//////////////////////////////////LCD//////////////////////////////////
#include <LiquidCrystal.h>
LiquidCrystal lcd (13, 12, 11, 10, 9, 8);

//////////////////////////////////ULTRASONIC//////////////////////////////////
const int pingPin = 3;
const int pingPina = 4;
int cmlcd;
int cm;
int A=32;
int B=0;
int C=1;
int D=1;
int tegangan;
int waktu;
int poin2;
//////////////////////////////////HEATER//////////////////////////////////
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 2
OneWire oneWire (ONE_WIRE_BUS);
DallasTemperature sensors (&oneWire);
//long HEAT;
//////////////////////////////////POMPA//////////////////////////////////
const int PWM_1 = 5;
const int PWM_2 = 6;

//////////////////////////////////SWITCHING TEGANGAN//////////////////////////////////
const int Relay1Ex = 53;
const int Relay2Ex = 51;
const int Relay3Ex = 49;

const int level=40;
const int heater=42;
int level1;
```

```

int heater1;
////////////////////////////////TOMBOL////////////////////////////////
const int TombolA =43;
const int TombolB =41;
int TombolA1 = 0;
int TombolB1 = 0;
int Led1 =39;
int Led2 =37;
int Led3 =35;
void setup()

{
////////////////////////////////PIDLEVEL////////////////////////////////
//initialize the variables we're linked to

//turn the PID on
myPID.SetMode(AUTOMATIC);
////////////////////////////////PIDHEATER////////////////////////////////
/* Setpoint2 = 2900;
//turn the PID on
myPID.SetMode(AUTOMATIC);
*/
////////////////////////////////LCD////////////////////////////////
lcd.begin (20,4);
Serial.begin(9600);

////////////////////////////////ULTRASONIC////////////////////////////////
pinMode(pingPin , OUTPUT);
pinMode(pingPina, INPUT);

////////////////////////////////HEATER////////////////////////////////
sensors.begin();

////////////////////////////////POMPA////////////////////////////////
pinMode (47,OUTPUT);digitalWrite (47,HIGH);
pinMode (45,OUTPUT);digitalWrite (45,HIGH);

pinMode (level,INPUT);
pinMode (heater,INPUT);

////////////////////////////////HEATER////////////////////////////////
pinMode (Relay1Ex,OUTPUT);
pinMode (Relay2Ex,OUTPUT);
pinMode (Relay3Ex,OUTPUT);
pinMode (Led1, OUTPUT);
pinMode (Led2, OUTPUT);

```

```

pinMode (Led3, OUTPUT);
pinMode (TombolA,INPUT);
pinMode (TombolB,INPUT);
}
void level_ air ()
{
  myPID.Compute();
  analogWrite(PIN_ OUTPUT, Output);
  analogWrite (PWM_1,Output);
  analogWrite(PWM_2, 0);
}

void heater_ air()
{
  if(sensors.getTempCByIndex(0)>= 40 && cmlcd >= 8) //heater mati
  {
    digitalWrite(Relay1Ex, LOW);
    digitalWrite(Relay2Ex, LOW);
    digitalWrite(Relay3Ex, LOW);
    tegangan=0;
  }
  else if(sensors.getTempCByIndex(0)>= 39.8 && cmlcd >= 8) //heater lambat
  {
    digitalWrite(Relay1Ex, HIGH);
    digitalWrite(Relay2Ex, LOW);
    digitalWrite(Relay3Ex, LOW);
    tegangan=150;
  }
  else if(sensors.getTempCByIndex(0)>= 39.5 && cmlcd >= 8) //heater sedang
  {
    digitalWrite(Relay1Ex, LOW);
    digitalWrite(Relay2Ex, HIGH);
    digitalWrite(Relay3Ex, LOW);
    tegangan=200;
  }
  else if(sensors.getTempCByIndex(0)>= 0 && cmlcd >= 8) //heater penuh
  {
    digitalWrite(Relay1Ex, LOW);
    digitalWrite(Relay2Ex, LOW );
    digitalWrite(Relay3Ex, HIGH);
    tegangan=220;
  }
}

long microsecondsToInches(long microseconds)
{

```

```

return microseconds / 74 / 2;
}
long microsecondsToCentimeters(long microseconds)
{
return microseconds / 29 / 2;
}

```

```

void loop()
{
lcd.setCursor(0,0);
lcd.print(" TUGAS AKHIR ");
lcd.setCursor(0,1);
lcd.print(" KONTROL LEVEL DAN ");
lcd.setCursor(0,2);
lcd.print(" TEMPERATUR AIR ");
lcd.setCursor(0,3);
lcd.print(" BY RAHMAT & ALDY ");
delay (2000);
lcd.clear();
Poin0:
lcd.setCursor(0,0);
lcd.print("Silahkan Tekan ");
lcd.setCursor(0,1);
lcd.print("tombol sesuai jenis");
lcd.setCursor(0,2);
lcd.print("pengontrolan");
lcd.setCursor(0,3);
waktu=(6-(millis() / 1000));
lcd.print(waktu);
TombolA1 = digitalRead(TombolA);
TombolB1 = digitalRead(TombolB);

if (waktu == 0)
{
lcd.clear();
goto Poin1;
}
goto Poin0;
Poin1:

```

```

////////////////////Pembacaan Level Air////////////////////////////////////
long duration, inches, cm;
digitalWrite(pingPin, LOW);
delayMicroseconds(2);
digitalWrite(pingPin, HIGH);
delayMicroseconds(5);

```

```

digitalWrite(pingPin, LOW);
duration = pulseIn(pingPina, HIGH);
inches = microsecondsToInches(duration);
cm = microsecondsToCentimeters(duration);
cmlcd = A-cm;
if (cmlcd>7 && B==0)
{
  A=32;
  B=1;
  C=0;
}
if (cmlcd>10 && C==0)
{
  A=33;
  C=1;
  D=0;
}
if (cmlcd>17 && D==0)
{
  A=33;
  D=1;
}
Input = cmlcd;
//////////////////////Pembacaan Heater Air//////////////////////
sensors.requestTemperatures(); // Send the command to get temperature readings
level1=1;
heater1=1;
//////////////////////Logika Program//////////////////////
if (TombolA1 == LOW)
{
  Setpoint = 12;
  level_air();
  if (Output > 150){
    digitalWrite(Led1, HIGH);
    digitalWrite(Led2, LOW);
    digitalWrite(Led3, LOW);
  }
  else if (Output > 100){
    digitalWrite(Led1, LOW);
    digitalWrite(Led2, HIGH);
    digitalWrite(Led3, LOW);
  }
  else if (Output >= 0){
    digitalWrite(Led1, LOW);
    digitalWrite(Led2, LOW);
    digitalWrite(Led3, HIGH);
  }
}

```



```

}
if (cmlcd>7)
{
  heater_air();
}
else
{
  digitalWrite(Relay1Ex, LOW);
  digitalWrite(Relay2Ex, LOW);
  digitalWrite(Relay3Ex, LOW);
  tegangan=0;
}
Serial.print(sensors.getTempCByIndex(0));
Serial.print(",");
Serial.print(cmlcd);
Serial.print(",");
Serial.println( );
}
else if (TombolB1 == LOW)
{
  Setpoint = 16;
  level_air();
  if (Output > 150){
    digitalWrite(Led1, HIGH);
    digitalWrite(Led2, LOW);
    digitalWrite(Led3, LOW);
  }
  else if (Output > 100){
    digitalWrite(Led1, LOW);
    digitalWrite(Led2, HIGH);
    digitalWrite(Led3, LOW);
  }
  else if (Output >= 0){
    digitalWrite(Led1, LOW);
    digitalWrite(Led2, LOW);
    digitalWrite(Led3, HIGH);
  }
  Serial.print(sensors.getTempCByIndex(0));
  Serial.print(",");
  Serial.print(cmlcd);
  Serial.print(",");
  Serial.println( );
}
else

```





```

{
  lcd.setCursor(0,0);
  lcd.print(" Tidak ada tombol ");
  lcd.setCursor(0,1);
  lcd.print("   Tertekan   ");
  digitalWrite(Relay1Ex, LOW);
  digitalWrite(Relay2Ex, LOW);
  digitalWrite(Relay3Ex, LOW);
  tegangan=0;
  analogWrite (PWM_1,0);
  analogWrite(PWM_2, 0);
  //lcd.setCursor(0,0);
  //lcd.print("2. Heater");
  lcd.setCursor(0,3);
  lcd.print(" Silahkan Tekan :) ");
  delay (2000);
  // millis.clear();
  lcd.clear();
  goto Poin0;
}
lcd.setCursor(0,0);
lcd.print("Tinggi Air: ");
lcd.setCursor(0,1);
lcd.print(cmlcd);
lcd.setCursor(5,1);
lcd.print("cm ");
lcd.setCursor(0, 2);
lcd.print("Temperatur Air");
lcd.setCursor(6, 3);
lcd.print (" 'C");
lcd.setCursor(0, 3);
lcd.print(sensors.getTempCByIndex(0));

delay (500);
lcd.setCursor(0,1);
lcd.print(" ");
goto Poin1;
}

```



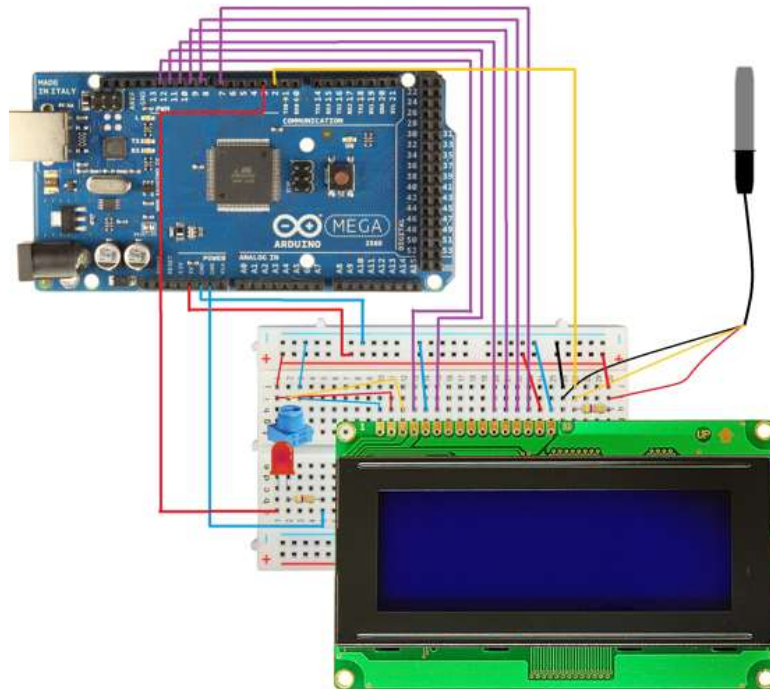


## **LAMPIRAN 4**

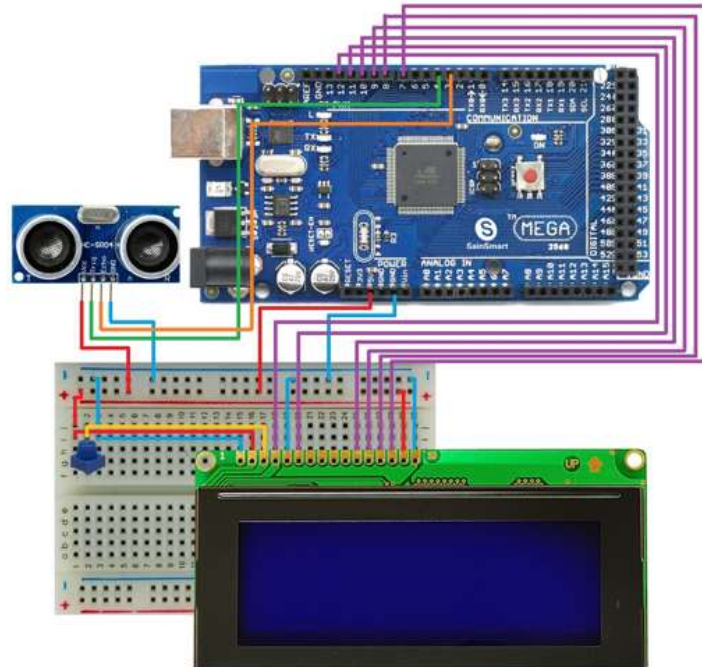
### **Dokumentasi Penelitian**



## UJI COBA SISTEM TEMPERATUR AIR



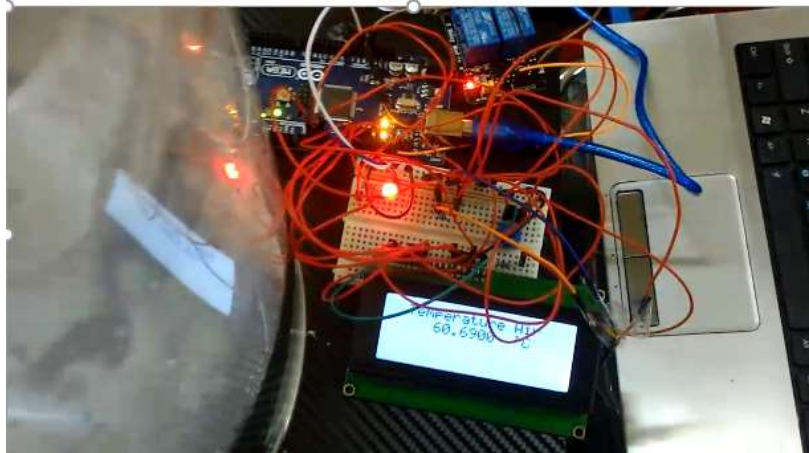
## UJI COBA SISTEM LEVEL AIR



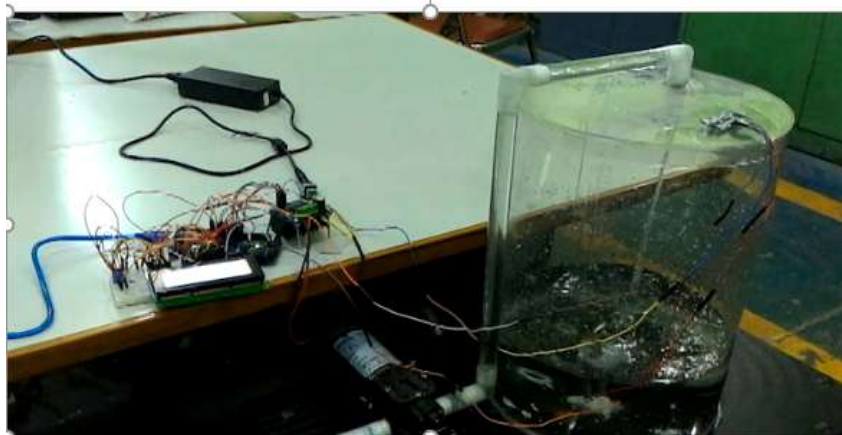
## UJI COBA SISTEM MEKANIK

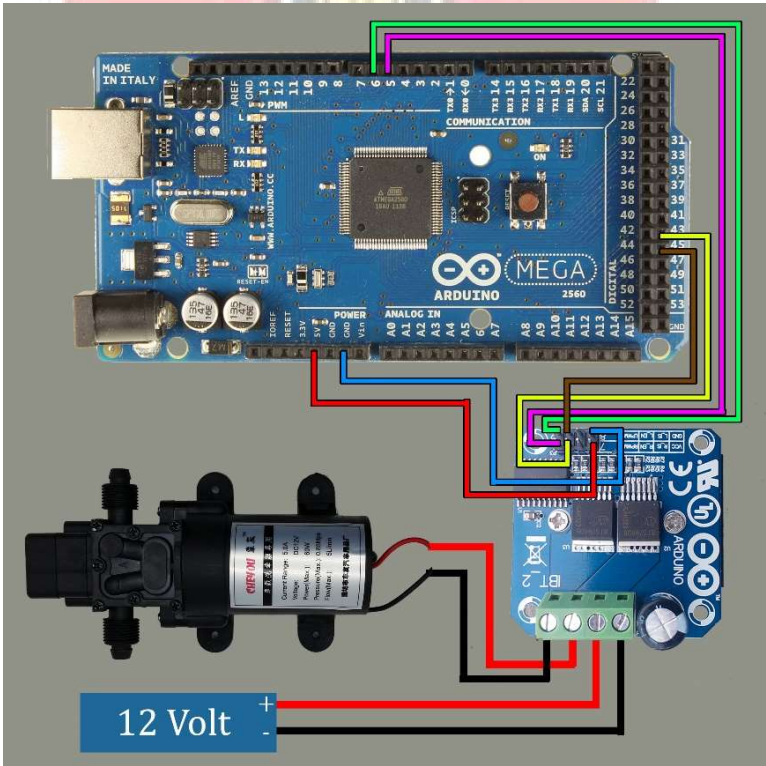
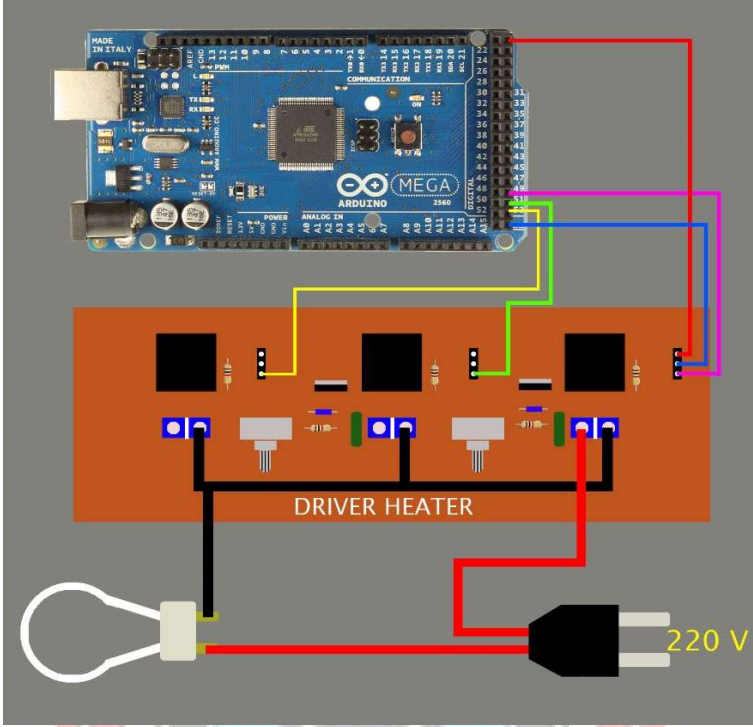


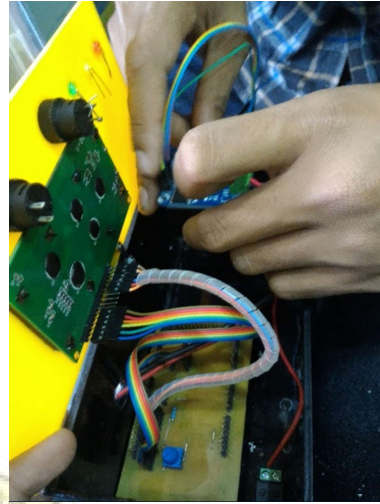
## UJI COBA KONTROL TEMPERATUR AIR



## UJI COBA KONTROL LEVEL AIR









## **LAMPIRAN 5**

### **Lembar Asistensi**





KEMENTERIAN RISET, TEKNOLOGI, DAN PENDIDIKAN TINGGI  
POLITEKNIK NEGERI UJUNG PANDANG

Jalan Perintis kemerdekaan Km. 10 Tamalanrea, Makassar 90245

☎ 0411-585368, 585367, 585365 Fax. 0411-586043

E-mail : [pnup@poliupg.ac.id](mailto:pnup@poliupg.ac.id)

Home Page : <http://www.poliupg.ac.id>

LEMBAR ASISTENSI

SKRIPSI  
"PENGEMBANGAN KONTROL LEVEL DAN TEMPERATUR AIR  
BERBASIS MIKROKONTROLER"

NAMA : Ronaldi Rexy Mawatang (443 13 040)  
Rahmat S (443 13 050)

NO	HARI / TANGGAL	REVISI	PARAF
①	01-08-2017	- Kutipan penulisan - Daftar pustaka	<i>[Signature]</i>
②	04-08-2017	- penulisan level → probant - Instalasi tegangan pada array Set point	<i>[Signature]</i>
③	08-08-2017	- Uji Matlab/Simulink → grafik - Uji Program Arduino/ matlab	<i>[Signature]</i>
4.	15-08-2017	- Grafik Level (matlab) - Grafik Temperatur (matlab)	<i>[Signature]</i>
5.	16-08-2017	- Grafik Tampilan Simulink → Berjelas	<i>[Signature]</i>
6.	18-08-2017	- Data Heater → buat Grafik	<i>[Signature]</i>
7.	24/08/2017	- probant Grafik 4.6-4.8 - pengujian suhu rata-rata heater All of Summary hasil	<i>[Signature]</i>
8.	25/08/2017	- Berjelas nilai parameter - grafik	<i>[Signature]</i>

Makassar, 24 Agustus 2017

Pembimbing I,

Ir. Simon Ka'ka, M.T.  
NIP. 19590913 198803 1 001



KEMENTERIAN RISET, TEKNOLOGI, DAN PENDIDIKAN TINGGI  
POLITEKNIK NEGERI UJUNG PANDANG

Jalan Perintis kemerdekaan Km. 10 Tamalanrea, Makassar 90245

☎ 0411-585368, 585367, 585365 Fax: 0411-586043

E-mail : [pnup@poliunp.ac.id](mailto:pnup@poliunp.ac.id)

Home Page : <http://www.poliunp.ac.id>

LEMBAR ASISTENSI  
SKRIPSI

"PENGEMBANGAN KONTROL LEVEL DAN TEMPERATUR AIR BERBASIS MIKROKONTROLER"

NAMA : Ronaldi Raxy Mawatang (443 13 040)  
Rahmat S. (443 13 050)

No.	Hari/ Tanggal	Uraian Revisi	Paraf
1.	Selasa, 8 Agt.2017	Secara umum: perhatikan penulisan kata sambung dan tanda baca sesuaikan EYD. Penulisan kutipan disinkronkan dengan daftar pustaka.	Ok
2.	Rabu, 9 Agt.2017	Format penulisan disesuaikan dengan buku panduan yang berlaku di PNUP. Perbaiki batasan masalah dan tujuan pen.	Ok
3.	Jumat, 11 Agust.2017	Tinjauan pustaka: semua pustaka yang dirujuk, cantumkan di daftar pustaka.	Ok
4.	Senin, 14 Agust.2017	Perbaiki flowchart gambar 3-1 dan 3-2.	Ok
5.	Rabu, 16 Agust.2017	Jelaskan cara memperoleh model plant (pada kontrol level dan temperatur)	Ok
6.	Jumat, 18 Agust.2017	Lengkapi pembahasan dan daftar pustaka	Ok
7.	Senin, 21 Agust.2017	Sesuaikan kesimpulan dengan rumusan masalah dan tujuan penelitian.	Ok
8.	Selasa, 22 Agust.2017	Rapikan secara keseluruhan.	Ok
9.	Rabu, 23 Agust.2017	Acc untuk ujian	Acc

Makassar , 23 Agustus 2017

Pembimbing I,

Ir. Lewi M.T.




NIP. 19650913 199103 1 006

**LEMBAR REVISI JUDUL PROYEK / TUGAS AKHIR**

162

NAMA : Rahmat S / Haradi Pery Mawatang  
 STAMBUK : 40720871 - 0313046

**Catatan Penguji :**

No	Nama	Uraian	Tanda Tangan
1	Dr. Ir. Muhammad Anwar, MT	<ul style="list-style-type: none"> <li>- 10 (hal 2), tambahkan kelengkapan pendahuluan lainnya</li> <li>- Perbaiki nama pengarang (hal 6)</li> <li>- (hal 7) Fery Djunaidi, tambahkan di Daftar Pustaka</li> <li>- Gambar 1 diteliti dg nomor 14 lengkap (11-2010-12-10) baik diuraikan</li> <li>- perbaiki diagram abstrak (hal 30, 31)</li> <li>- perbaiki ukuran foto (hal 33)</li> <li>- (hal 33 dt) ganti foto profil dg gambar - (kandung no. 30)</li> <li>- (hal 34) tambahkan gambar</li> <li>- hasil jangan hambatan kuantitas</li> <li>- (hal 47) semua foto diteliti no. dan keceringan (optimal)</li> </ul>	
2	Dr. Eng. Achmad Tazki	<ul style="list-style-type: none"> <li>- k. simpulan diteliti</li> <li>- revisi pengantar</li> <li>- revisi gambar 3.1.</li> </ul>	
3	Dr. Eng. Abdul Kadir Muhammad, ST, MT	<ul style="list-style-type: none"> <li>- setiap media masalah</li> <li>- buatlah masalah, probkuan</li> <li>- 30.1 (hal 3) 2 → ke awal</li> <li>- Revisi simulasi (nilai K, L)</li> <li>- perbaiki abstrak 4.5 dan 4.6</li> <li>- Evisi Hal 40: Kp, K2, K1</li> <li>- Revisi Tabel 4.1</li> </ul>	

Makassar,  
 Ketua / Sekretaris Penguji Ujian Sidang



Dr. Eng. Abdul Kadir Muhammad, ST, MT, Eng  
 NIP. 19750902 200312 1 002

Catatan: Jika ada perubahan Judul Tugas Akhir konfirmasi secepatnya ke bagian