

**RANCANG BANGUN SISTEM KONTROL AIR MINUM KANDANG
BURUNG PUYUH BERBASIS *INTERNET OF THINGS*
MENGUNAKAN METODE *FUZZY LOGIC***

SKRIPSI

**Disusun Oleh:
TEGUH ARIEF RAMADHAN
NIM. 16650088**



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2020**

**RANCANG BANGUN SISTEM KONTROL AIR MINUM KANDANG
BURUNG PUYUH BERBASIS *INTERNET OF THINGS*
MENGUNAKAN METODE *FUZZY LOGIC***

SKRIPSI

**Diajukan kepada:
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh :
TEGUH ARIEF RAMADHAN
NIM. 16650088**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2020**

LEMBAR PERSETUJUAN

RANCANG BANGUN SISTEM KONTROL AIR MINUM KANDANG BURUNG PUYUH BERBASIS *INTERNET OF THINGS* MENGUNAKAN METODE *FUZZY LOGIC*

SKRIPSI

Oleh :
TEGUH ARIEF RAMADHAN
NIM. 16650088

Telah Diperiksa dan Disetujui untuk Diuji
Tanggal : 30 Desember 2020

Pembimbing I

Pembimbing II

Prof. Dr. Suhartono , S.Si, M.Kom
NIP. 19680519 200312 1 001

Fresy Nugroho, S.T, M.T
NIP. 19710722 201101 1 001

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

LEMBAR PENGESAHAN

RANCANG BANGUN SISTEM KONTROL AIR MINUM KANDANG BURUNG PUYUH BERBASIS *INTERNET OF THINGS* MENGUNAKAN METODE *FUZZY LOGIC*

SKRIPSI

Oleh :
TEGUH ARIEF RAMADHAN
NIM. 16650088

Telah Dipertahankan di Depan Dewan Penguji Skripsi dan
Dinyatakan Diterima sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal : 30 Desember 2020

Susunan Dewan Penguji :

Tanda Tangan

Penguji Utama	: <u>Roro Inda Melani, M.T, M.Sc</u> NIP. 19780925 200501 2 008	()
Ketua Penguji	: <u>Yunifa Miftachul Arif, M.T</u> NIP. 19830616 201101 1 004	()
Sekretaris Penguji	: <u>Prof. Dr. Suhartono, S.Si, M.Kom</u> NIP. 19680519 200312 1 001	()
Anggota Penguji	: <u>Fresy Nugroho, S.T, M.T</u> NIP. 19710722 201101 1 001	()

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Teguh Arief Ramadhan

NIM : 16650088

Fakultas / Jurusan : Sains dan Teknologi / Teknik Informatika

Judul Skripsi : Rancang Bangun Sistem Kontrol Air Minum Kandang
Burung Puyuh Berbasis *Internet of Things* Menggunakan
Metode *Fuzzy Logic*

Menyatakan dengan sebenarnya bahwa skripsi saya tulis ini merupakan benar-benar hasil karya tulis saya sendiri, bukan merupakan pengambilan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil karya tulis atau pikiran saya sendiri, kecuali dengan menyertakan sumber kutipan pada daftar pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa skripsi saya hasil plagiasi, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 30 Desember 2020
Saya membuat pernyataan,




Teguh Arief Ramadhan
NIM. 16650088

HALAMAN MOTTO



HALAMAN PERSEMBAHAN

الحمد لله رب العالمين

Puji syukur kehadiran Allah SWT

Shalawat serta salam kepada Rasulullah SAW

Dengan segenap hati, penulis mempersembahkan karya ini kepada:

Kedua orang tua penulis tercinta, Bapak Teguh Ichsan dan Ibu Martha Trisnati yang selalu dengan senang hati membimbing dan memotivasi diri penulis, yang tak pernah putus doa nya untuk penulis, yang selalu mendukung untuk menjadi yang terbaik, dan yang kasih sayangnya tak terhingga hingga kapan pun.

Dosen pembimbing, Bapak Prof. Dr. Suhartono, S.Si, M.Kom dan Bapak Fresy Nugroho, S.T, M.T yang telah dengan sabarnya memberikan bimbingan di tengah jalannya penelitian skripsi ini dan yang selalu mengalirkan stimulus yang positif untuk menjaga semangat dalam menempuh setiap tahap ujian skripsi.

Dosen penguji, Ibu Roro Inda Melani, M.T, M.Sc dan Bapak Yunifa Miftachul Arif, M.T telah memberikan bimbingan dan kritik yang sangat membangun kepada penulis semata agar tergapai hasil skripsi yang lebih baik.

Seluruh dosen Teknik Informatika UIN Maulana Malik Ibrahim Malang, dan seluruh guru-guru penulis yang telah memberikan bimbingan dan ilmunya yang sangat bermanfaat khususnya bagi diri penulis.

Keluarga Teknik Informatika UIN Maulana Malik Ibrahim Malang, terutama kawan seperjuangan Andromeda (Angkatan 2016) yang selalu menyebarkan semangat dan do'anya.

Penulis menyampaikan “jazakumullah khairan katsiiraan”. Semoga ukhuwah kita tetap terjalin dan tentunya diridhoi Allah SWT. Aamiin Ya Rabbal ‘Alamiin.

KATA PENGANTAR

Bismillahirrahmanirrahim

Assalamu'alaikum Warahmatullahi Wabarakaatuhu

Puji dan syukur kami panjatkan kehadirat Allah SWT yang telah menganugrahkan rahmat serta karunia-Nya, yang dengan itu semua penulis dapat menyelesaikan penulisan skripsi yang berjudul “ Rancang Bangun Sistem Kontrol Air Minum Kandang Burung Puyuh Berbasis *Internet of Things* Menggunakan Metode *Fuzzy Logic*” ini dengan lancar dan baik. Dan tak lupa sholawat serta salam selalu kami haturkan kepada junjungan kita Nabi Muhammad SAW yang telah menuntun umat manusia dari zaman kegelapan ke zaman yang terang benderang ini. Tujuan dari penyusunan skripsi ini adalah untuk memenuhi salah satu syarat untuk bisa menempuh ujian sarjana komputer pada Fakultas Sains dan Teknologi (FSAINTEK) Program Studi Teknik Informatika di Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.

Tentu saja selama proses pengerjaan skripsi ini penulis mendapat banyak sekali dukungan dan bantuan dari berbagai pihak. Oleh karena itu, di sini penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. Abdul Haris, M.Ag selaku Rektor Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M.Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
3. Dr. Cahyo Crysodian, selaku Ketua Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
4. Prof. Dr. Suhartono, S.Si, M.Kom dan Fresy Nugroho, M.T, selaku Dosen Pembimbing yang senantiasa sabar dalam membimbing penulis, memberikan kritik, saran serta motivasi sehingga penulis tidak hanya mampu menyelesaikan skripsi namun juga dapat memberikan nilai manfaat nantinya dari skripsi ini.

5. Roro Inda Melani, M.T, M.Sc dan Yunifa Miftachul Arif, M.T, selaku Dosen Penguji yang senantiasa memberikan bimbingan dan kritik yang membangun kepada penulis sehingga tergapai hasil skripsi yang lebih baik.
6. Khadijah Fahmi H, M.Kom, selaku Dosen Wali yang tak pernah berhenti menyalurkan motivasi juga sarannya untuk kebaikan penulis.
7. Orang tua tercinta yang tak ada hentinya mendo'akan dan mendukung penulis dari segi moril maupun materiil hingga terselesaikannya penulisan skripsi ini.
8. Anggota keluarga dan kerabat yang senantiasa menyalurkan do'a serta dukungan semangat kepada penulis.
9. Kawan seperjuangan Andromeda (Angkatan 2016) yang selalu menebarkan rasa semangat dan do'a kepada penulis.
10. Semua pihak yang telah banyak membantu dalam penyusunan skripsi ini yang tidak bisa penulis sebutkan semuanya tanpa mengurangi rasa hormat dan terima kasih.
11. Dan tak lupa pada diri penulis sendiri yang sudah berusaha mengupayakan yang terbaik, yang tak pernah menyerah meski banyak rintangan yang dihadapi, yang selalu bersemangat dalam mengupayakan yang terbaik, dan yang selalu percaya bahwa dari setiap proses tak ada yang sia-sia.

Penulis sadar bahwa skripsi yang telah disusun ini masih terdapat banyak sekali kekurangan dan tentunya masih jauh untuk dapat dikatakan sempurna. Namun penulis berharap semoga dengan terselesaikannya skripsi ini dapat memberikan setidaknya sedikit manfaat kepada para pembaca dan khususnya bagi penulis secara pribadi.

Malang, 30 Desember 2020

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN.....	iii
PERNYATAAN KEASLIAN TULISAN	iv
HALAMAN MOTTO	v
HALAMAN PERSEMBAHAN	vi
KATA PENGANTAR.....	vii
DAFTAR ISI	ix
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL	xiv
ABSTRAK.....	xv
ABSTRACT	xvi
المخلص.....	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pernyataan Masalah.....	4
1.3 Tujuan Penelitian	6
1.4 Batasan Masalah.....	6
1.5 Manfaat Penelitian.....	6
BAB II STUDI PUSTAKA	7
2.1 Penelitian Terkait	7
2.2 Landasan Teori.....	10
2.2.1 <i>IoT (Internet of Things)</i>	10
2.2.2 <i>Arduino Uno</i>	11
2.2.3 <i>NodeMCU-ESP8266</i>	12
2.2.4 <i>Sensor Ultrasonic HC SR-04</i>	14
2.2.5 <i>Turbidity Sensor SKU SEN0189</i>	16

2.2.6 Pompa Air	18
2.2.7 Relay	19
2.2.8 LCD (<i>Liquid Crystal Display</i>).....	21
2.2.9 Fuzzy Logic	21
2.2.9.1 Himpunan Fuzzy	22
2.2.9.2 Domain.....	24
2.2.9.3 Fuzzifikasi	25
2.2.9.4 Rule Base.....	26
2.2.9.5 Defuzzifikasi	26
BAB III METODE PENELITIAN	29
3.1 Desain Penelitian.....	29
3.1.1 Perangkat Keras.....	29
3.1.2 Perangkat Lunak.....	30
3.2 Desain Sistem.....	30
3.2.1 Blok Diagram Sistem.....	31
3.2.2 Flowchart Sistem	32
3.3 Implementasi Metode Fuzzy Logic	33
3.3.1 Fuzzifikasi	34
3.3.2 Fuzzy Rule.....	38
3.3.3 Defuzzifikasi	40
BAB IV PENGUJIAN DAN PEMBAHASAN	43
4.1 Hasil Pengujian	43
4.1.1 Pengujian Alat.....	47
4.1.1.1 Pengujian Sensor Level (<i>Ultrasonic HCSR-04</i>)	47
4.1.1.2 Pengujian Sensor Kekeruhan (<i>Turbidity Sensor SKU SEN0189</i>)...48	
4.1.2 Pengujian Sistem Mikrokontroler dengan Matlab	50

4.2 Pembahasan	52
4.2.1 Pembahasan Perhitungan <i>Fuzzy Logic</i>	52
4.2.1.1 Perhitungan pada Matlab	52
4.2.1.2 Perhitungan pada <i>Mikrokontroller</i>	57
4.2.2 Sistem <i>Hardware</i>	62
4.2.2.1 <i>Arduino UNO</i>	62
4.2.2.2 <i>Node MCU ESP8266</i>	64
4.2.3 Sistem <i>Interface</i>	64
4.2.3.1 Tampilan <i>Home</i>	64
4.2.3.2 Tampilan Data Level	65
4.2.3.3 Tampilan Data Kekerusuhan	66
4.2.4 <i>Pengontrolan Air Minum Kandang Burung Puyuh</i>	67
4.3 Integrasi Islam.....	70
BAB V PENUTUP	72
5.1 Kesimpulan	72
5.2 Saran.....	73
DAFTAR PUSTAKA.....	74
LAMPIRAN	77

DAFTAR GAMBAR

Gambar 2. 1 Bentuk Fisik <i>Arduino Uno</i>	12
Gambar 2. 2 Bentuk Fisik <i>NodeMCU-ESP8266</i>	14
Gambar 2. 3 Prinsip Sensor <i>Ultrasonic</i>	15
Gambar 2. 4 Prinsip Pemantulan <i>Ultrasonic</i>	16
Gambar 2. 5 Bentuk Fisik <i>Turbidity Sensor SKU SEN0189</i>	17
Gambar 2. 6 Bentuk Fisik Pompa Air.....	19
Gambar 2. 7 Bentuk Fisik <i>Relay</i>	20
Gambar 2. 8 Cara Kerja <i>Relay</i>	20
Gambar 2. 9 Bentuk Fisik <i>LCD (Liquid Crystal Display)</i>	21
Gambar 2. 10 Himpunan : MUDA, PAROBAYA, dan TUA.....	23
Gambar 2. 11 Himpunan <i>Fuzzy</i> Untuk Variable Umur	24
Gambar 2. 12 Proses <i>Fuzzifikasi</i>	25
Gambar 2. 13 Proses <i>Rule Base</i>	26
Gambar 2. 14 Metode <i>Defuzzifikasi</i>	27
Gambar 3. 1 Blok Diagram Alur Penelitian	29
Gambar 3. 2 Blok Diagram Sistem.....	31
Gambar 3. 3 <i>Flowchart</i> Sistem.....	32
Gambar 3. 4 <i>Flowchart</i> Sistem Kontrol <i>Fuzzy Logic</i>	34
Gambar 3. 5 Fungsi Keanggotaan <i>Level</i>	36
Gambar 3. 6 Fungsi Keanggotaan Kekeruhan.....	37
Gambar 3. 7 Fungsi Keanggotaan Waktu	38
Gambar 3. 8 Nilai Tengah Fungsi Keanggotaan Waktu	42
Gambar 4. 1 Grafik Kondisi Level Air	45
Gambar 4. 2 Grafik Kondisi Kekeruhan Air	46
Gambar 4. 3 Fungsi Keanggotaan Level.....	53
Gambar 4. 4 Fungsi Keanggotaan Kekeruhan.....	54
Gambar 4. 5 Pembuktian Perhitungan Matlab	57
Gambar 4. 6 Rangkaian Pada Panel <i>Arduino</i> Menggunakan <i>Breadbord</i>	63
Gambar 4. 7 Rangkaian Pada Panel <i>Arduino</i> Menggunakan PCB Titik.....	63
Gambar 4. 8 <i>Node MCU ESP8266</i>	64

Gambar 4. 9 Tampilan <i>Home</i>	65
Gambar 4. 10 Tampilan Data Level.....	66
Gambar 4. 11 Tampilan Data Kekeruhan	67
Gambar 4. 12 Penataan <i>Prototype</i> Kandang Burung Puyuh.....	68
Gambar 4. 13 Penempatan Sensor <i>Ultrasonic</i>	68
Gambar 4. 14 Penempatan Sensor <i>Turbidity</i>	69



DAFTAR TABEL

Tabel 2. 1 Spesifikasi <i>Arduino</i>	12
Tabel 2. 2 Spesifikasi <i>NodeMCU-ESP8266</i>	14
Tabel 2. 3 Spesifikasi <i>Sensor Ultrasonic HC-SR04</i>	16
Tabel 2. 4 Spesifikasi <i>Turbidity Sensor SKU SEN0189</i>	18
Tabel 3. 1 <i>Fuzzy Rule</i> Penambahan	38
Tabel 3. 2 <i>Fuzzy Rule</i> Pengurangan.....	39
Tabel 3. 3 Data Luas dan Titik Tengah.....	42
Tabel 4. 1 Data Uji Coba Sistem Kontrol Air Minum Kandang Burung Puyuh...	43
Tabel 4. 2 Pengujian Sensor Level (<i>Ultrasonic HCSR-04</i>) dengan penggaris.....	47
Tabel 4. 3 Pengujian Sensor Kekeruhan (<i>Turbidity SKU SEN0189</i>) dengan <i>sample</i> air	49
Tabel 4. 4 Pengujian <i>Mikrokontroler</i> dengan Matlab.....	50
Tabel 4. 5 Perhitungan Aturan <i>Fuzzy</i>	55
Tabel 4. 6 Rule Luas dan CoA.....	56

ABSTRAK

Ramadhan, Teguh Arief. 2020. **Rancang Bangun Sistem Kontrol Air Minum Kandang Burung Puyuh Berbasis *Internet Of Things* Menggunakan *Fuzzy Logic***. Skripsi. Jurusan Teknik Informatika, Fakultas Sains dan Teknologi. Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. Suhartono, S.Si, M.Kom., (II) Fresy Nugroho, M.T.

Kata Kunci : *Fuzzy Logic*, *Internet of Things*, Kontrol Air, Burung Puyuh

Pada era revolusi industri 4.0 ini, *Internet of Things (IoT)* mulai banyak dikembangkan dan digunakan untuk memudahkan berbagai pekerjaan manusia. Di Indonesia, beternak merupakan bagian dari pembangunan nasional yang bertujuan untuk menyediakan pangan hewani berupa daging, susu, dan telur yang bernilai gizi tinggi. Salah satu usaha peternakan yang banyak diminati adalah ternak burung puyuh. Air minum menjadi salah satu faktor yang mempengaruhi produktivitas burung puyuh. Dengan adanya kontrol air minum kandang burung puyuh berbasis *Internet of Things (IoT)* diharapkan dapat memberi kemudahan bagi peternak dalam melakukan control pada air minum kandang puyuh.

Penelitian yang dilakukan oleh peneliti menggunakan *mikrokontroler* berupa *Arduino Uno* serta *Node MCU ESP8266*, juga menggunakan 2 buah sensor yaitu sensor *Ultrasonic HC SR-04* dan sensor *Turbidity SKU SEN0189*. Metode yang digunakan adalah metode *Fuzzy Logic*. Metode *fuzzy* melakukan perhitungan dengan digunakannya dua buah parameter *input* yaitu level dan kekeruhan. Dan akan dihasilkan *output* dari perhitungan *fuzzy* berupa waktu, yaitu waktu pengisian dan pengurasan air.

Dari penelitian yang telah dilakukan, sistem kontrol air minum kandang burung puyuh berhasil dibangun. Pengujian dilakukan pada sensor dan perhitungan metode. Pada sensor *Ultrasonic HC SR-04* didapat rata-rata *error* sebesar 4.71% dan pada sensor *Turbidity SKU SEN0189* didapat rata-rata *error* sebesar 3.96%. Pengujian perhitungan metode dilakukan dengan cara membandingkan nilai *output* yang dihasilkan oleh program dengan nilai yang dihasilkan oleh perhitungan Matlab. Dan didapatkan rata-rata *error* sebesar 6.17 %. Sehingga pada penelitian ini peneliti mendapatkan presentase keberhasilan sistem sebesar 93,83%.

ABSTRACT

Ramadhan, Teguh Arief. 2020. **Design and Development of an Internet of Things-Based Quail Drinking Water Control System Using Fuzzy Logic**. Essay. Informatics Engineering Department, Faculty of Science and Technology. Islamic State University of Maulana Malik Ibrahim Malang. Advisor: (I) Prof. Dr. Suhartono, S.Si, M.Kom., (II) Fresy Nugroho, M.T.

Keywords: Fuzzy Logic, Internet of Things, Water Control, Quail

In the era of the industrial revolution 4.0, the Internet of Things (IoT) developed and used to facilitate a variety of human work. In Indonesia, breeding is part of national development that aims to provide animal food such as meat, milk, and eggs of high nutritional value. One of the farm businesses are much in demand is the livestock quail. Drinking water to be one of the factors that affect the productivity of quail. With the control of drinking water of cage birds quail-based Internet of Things (IoT) is expected to provide convenience for the breeder in the conduct of control to the drinking water in the cage of quail.

Research conducted by the researchers using a microcontroller in the form of an Arduino Uno and the Node MCU ESP8266, also use 2 pieces of the sensor i.e. Ultrasonic sensor HC SR-04 and Turbidity sensors SKU SEN0189. The method used is the method of Fuzzy Logic. Fuzzy method performs the calculation with the use of two input parameters, namely the level and turbidity. And the output from the calculation of the fuzzy form of time, namely the time of charging and draining the water.

From the research that has been done, the control system of drinking water of cage birds quail successfully built. Testing done on the sensors and calculation methods. The Ultrasonic sensor HC SR-04 is obtained an average error of 4.71% and in Turbidity sensors SKU SEN0189 obtained an average error of 3.96%. Testing the calculation method was done by comparing the output value generated by the program with the value that is produced by calculation of Matlab. And obtained an average error of 6.17 %. So in this research, the researchers get a percentage of the success of the system amounted to 93,83%.

الملخص

رمضان ، تيكوه عارف. ٢٠٢٠. تقييم تقاسم البيانات لمقصد السياح في مدينة مالانغ تصميم وتطوير نظام التحكم في مياه الشرب السمان القائم على إنترنت الأشياء باستخدام المنطق الضبابي. مقال. قسم المعلوماتية ، كلية العلوم والتكنولوجيا. جامعة مولانا مالك إبراهيم الإسلامية الحكومية في مالانغ. المشرف : (١) الأستاذ الدكتور سوهارتنو ، الماجستير. (٢) فريسي نوغروهو ، الماجستير.

الكلمات الرئيسية: المنطق الضبابي, إنترنت الأشياء, التحكم في المياه ، السمان

في عصر الثورة الصناعية 4.0 ، تم تطوير إنترنت الأشياء (IoT) واستخدامه لتسهيل مجموعة متنوعة من العمل البشري. في إندونيسيا ، تعد التربية جزءا من التنمية الوطنية التي تهدف إلى توفير الغذاء الحيواني مثل اللحوم والحليب والبيض ذات القيمة الغذائية العالية. واحدة من الشركات الزراعية هي الكثير في الطلب هو السمان الماشية. شرب الماء من العوامل التي تؤثر على إنتاجية السمان. مع السيطرة على مياه الشرب من الطيور قفص السمان القائم على الإنترنت من الأشياء (تقنيات عمليات) ومن المتوقع أن توفر الراحة للمربي في إجراء السيطرة على مياه الشرب في قفص السمان.

البحوث التي أجراها الباحثون باستخدام متحكم في شكل اردوينو أونو والعقدة MCU ESP8266 ، وأيضا استخدام 2 قطعة من أجهزة الاستشعار أي الاستشعار بالموجات فوق الصوتية HC SR-04 وأجهزة الاستشعار التعكر SKU SEN0189. الطريقة المستخدمة هي طريقة المنطق الضبابي. طريقة غامض ينفذ الحساب مع استخدام اثنين من المعلومات المدخلات ، وهي مستوى والتعكر. والإخراج من حساب شكل غامض من الوقت ، وهي وقت الشحن واستنزاف المياه.

من البحث الذي تم القيام به ، ونظام التحكم في مياه الشرب من الطيور قفص السمان بنيت بنجاح. اختبار القيام به على أجهزة الاستشعار و طرق الحساب. يتم الحصول على جهاز استشعار بالموجات فوق الصوتية HC SR-04 متوسط خطأ 4.71٪. وفي أجهزة استشعار التعكر SKU SEN0189 حصلت على متوسط خطأ 3.96٪. تم اختبار طريقة الحساب من خلال مقارنة قيمة المخرجات التي تم إنشاؤها بواسطة البرنامج بالقيمة التي يتم إنتاجها عن طريق حساب Matlab. وحصل على متوسط خطأ 6.17٪. حتى في هذا البحث ، والباحثين الحصول على نسبة من نجاح النظام بلغت 93,83٪.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Menurut Kamus Besar Bahasa Indonesia (KBBI), air adalah cairan jernih tidak berwarna, tidak berasa, dan tidak berbau yang terdapat dan diperlukan dalam kehidupan manusia, hewan, dan tumbuhan yang secara kimiawi mengandung hidrogen dan oksigen.

Air merupakan unsur terpenting bagi tubuh makhluk hidup dengan rasio perbandingan sebesar 60% hingga 70% dibandingkan dengan unsur yang lain. Besarnya rasio tentu menjadi suatu keharusan untuk memperhatikan kelayakan air yang digunakan untuk dikonsumsi. Secara fisis air bersih diindikasikan dengan keadaannya yang bening, tidak berwarna dan tidak berbau. Kondisi seperti ini terjadi jika air tidak dikotori oleh bahan organik dan anorganik. Sedangkan secara optis, air yang tercampur oleh bahan pengotor, keadaannya akan mengalami perubahan, mungkin menjadi berwarna atau menjadi keruh. (Peslinof, 2013)

Tingkat kekeruhan air adalah suatu studi dari sifat-sifat optis yang menyebabkan cahaya yang melewati air menjadi terhambur dan terserap dari cahaya yang dipancarkan dalam garis lurus (Fairuz dan Zubir, 2009). Kekeruhan menyebabkan air menjadi seperti berkabut atau berkurangnya transparansi dari air. Arah dari berkas cahaya yang dipancarkan akan berubah ketika cahaya berbenturan dengan partikel di dalam air. Jika level kekeruhan rendah maka sedikit cahaya yang akan dihamburkan dan dibiaskan dari arah asalnya.

Air dikatakan keruh apabila air tersebut mengandung begitu banyak partikel bahan yang tersuspensi sehingga memberikan warna/rupa yang berlumpur dan

kotor. Bahan-bahan yang menyebabkan kekeruhan meliputi lumpur, bahan-bahan organik yang tersebut secara baik dan partikel-partikel yang tersuspensi lainnya.

Di Indonesia, beternak merupakan bagian dari pembangunan nasional yang bertujuan untuk menyediakan pangan hewani berupa daging, susu, dan telur yang bernilai gizi tinggi. Serta untuk meningkatkan pendapatan peternak, menambah devisa dan memperluas kesempatan kerja (Saragih, 2000).

Salah satu usaha peternakan yang banyak diminati adalah ternak burung puyuh, karena peternakan burung puyuh merupakan salah satu sektor peternakan yang paling efisien dalam menyediakan daging dan telur serta merupakan bahan makanan sumber hewani yang bergizi tinggi. Kemampuan tumbuh dan berkembang biak puyuh sangat cepat, dalam waktu sekitar 42 hari puyuh telah mampu memproduksi dan dalam waktu satu tahun dapat menghasilkan tiga sampai empat keturunan. Dalam setahun puyuh mampu menghasilkan 250 – 300 butir telur. Konsumsi pakan puyuh relatif sedikit (sekitar 20 gram per ekor per hari). Hal ini sangat menguntungkan peternak karena dapat menghemat biaya pakan (Listiyowati, E., dan Kinanti, R., 2009).

Di samping itu, populasi burung puyuh di Indonesia mengalami peningkatan dari tahun ke tahun. Berdasarkan data Statistik Peternakan dan Kesehatan Hewan tahun 2018, populasi burung puyuh di Indonesia tahun 2014-2018 secara berurutan mencapai 12,692 juta ekor, 13,781 juta ekor, 14,087 juta ekor, 14,569 juta ekor, 14,877 juta ekor. Hal ini menunjukkan bahwa populasi puyuh meningkat pesat dan tampaknya terus berkembang sejalan dengan munculnya berbagai olahan burung puyuh. Hal yang paling sering kita temukan dengan mudah adalah sate telur puyuh.

Di warung angkringan, soto, bahkan bubur ayam selalu menyajikan telur puyuh sebagai pelengkap untuk menyantap menu utama yang ada. Tak hanya menjadi favorit dewasa saja, anak-anak juga gemar menyantap telur burung puyuh.

Namun pengelolaan peternakan burung puyuh tidaklah mudah, banyak faktor yang perlu diperhatikan seperti suhu kandang, kelembaban kandang dan juga air minum ternak. Suhu kandang yang optimal untuk pertumbuhan puyuh selama produksi adalah 20-25°C. Kelembaban dalam kandang sangat penting untuk diperhatikan karena akan mempengaruhi kesehatan ternak. Kelembaban dalam kandang idealnya 30-80%. (Tetty, 2002). Ternak unggas yang menderita *heat stress* akan terlihat banyak mengonsumsi air minum, nafsu makan menurun, gelisah dan mengepakkan sayap di lantai kandang (Tamzil, 2014). Air yang digunakan untuk puyuh harus air bersih yang segar, tidak berbau, dan tidak beracun. Air minum bisa berasal dari air sumur, air ledeng atau dari sumber mata air. Air minum harus diberikan secara tidak terbatas (*adlibitum*), sehingga kapan pun puyuh akan minum, air selalu tersedia di dalam tempat minum (Sugiharto, 2005).

Pemberian air minum pada burung puyuh merupakan hal yang penting dalam perawatan burung puyuh, karena air minum sangat mempengaruhi produktivitas burung puyuh dan juga merupakan media untuk vaksinasi dan pemberian vitamin untuk mencegah penyakit dan meningkatkan kekebalan tubuh puyuh. Pemberian air minum harus dilakukan setiap saat dengan cara mengalirkan air dari penampungan air ke tempat minum puyuh yang ada pada setiap kandang puyuh melalui pipa air. Pengecekan kondisi air harus dilakukan setiap saat, karena jika air dalam kondisi kotor dapat mempengaruhi kesehatan dan produktivitas pada puyuh dan tentunya merugikan peternak itu sendiri. Hal tersebut tentunya cepat atau

lambat akan mempersulit pekerjaan para peternak untuk berada di dekat kandang dan selalu mengecek keadaan air yang ada di penampungan air, sedangkan para peternak selayaknya manusia pasti pernah lalai dan mungkin adanya kesibukan yang mengharuskan mereka untuk meninggalkan kandang puyuh tersebut.

Pesatnya kemajuan teknologi yang ada saat ini banyak membawa dampak positif bagi manusia, di antaranya bahkan dapat membantu pekerjaan yang dahulu hanya dapat dilakukan oleh manusia. Pada era revolusi industri 4.0 ini, *Internet of Things (IoT)* mulai banyak dikembangkan dan digunakan untuk memudahkan berbagai pekerjaan manusia, di antaranya dalam bidang industri dan pertanian. Namun menurut peneliti, hal ini masih jarang ditemukan dalam bidang peternakan, oleh karena itu dengan dibuatnya alat ini nanti, dapat mempermudah pekerjaan para peternak dalam melakukan proses ternak khususnya dalam pengontrolan air minum kandang burung puyuh.

Adanya sistem kontrol air minum dengan basis *Internet of Things* diharapkan dapat memberi kemudahan bagi peternak dalam melakukan kontrol pada kondisi air minum secara langsung (*realtime*). Dari sensor-sensor nantinya didapat data yang akan digunakan sebagai variabel perhitungan dalam sistem pengontrolan air minum. Hasilnya akan dikirimkan ke *database* yang nantinya akan dapat diakses oleh *smartphone* milik peternak.

Rancang bangun sistem kontrol air minum kandang burung puyuh ini dilakukan dengan menggunakan metode *fuzzy logic*. *Fuzzy* secara bahasa diartikan sebagai kabur atau samar-samar. Suatu nilai dapat bernilai besar atau salah secara bersamaan. *Fuzzy Logic* pertama kali diperkenalkan oleh Jan Lukasiewicz pada

tahun 1920-an yaitu sebagai teori kemungkinan. Logika kemungkinan ini memperluas jangkauan dari nilai kebenaran untuk semua bilangan riil pada interval antara 0 dan 1. Selanjutnya, Max Black melakukan penelitian lebih lanjut pada tahun 1930-an dalam penelitiannya tentang ke tidak jelasan (*vagueness*) sebuah latihan pada analisis logis.

Dan pada tahun 1965, profesor dan kepala departemen teknik elektrik di Universitas California, Barkeley, Lotfi Zadeh, menemukan kembali, mengidentifikasi, mengeksplorasi, mempromosikan, dan berjuang untuk *Fuzzy logic*. Profesor Zadeh memperluas ruang kerja teori kemungkinan menjadi sistem logika matematika formal dan konsep baru untuk mengaplikasikan istilah bahasa alami pada penelitiannya, yaitu '*Fuzzy Sets*'. Logika baru ini dinamakan *Fuzzy Logic*. *Fuzzy Logic* banyak digunakan karena mirip dengan cara berpikir manusia. Sistem *Fuzzy Logic* dapat merepresentasikan pengetahuan manusia dalam bentuk matematis dengan menyerupai cara berpikir manusia. (Budiharto dan Suhartono, 2014)

Dari permasalahan tersebut, peneliti ingin membuat suatu alat yang dapat mengontrol air minum kandang puyuh secara otomatis serta dapat melakukan *monitoring* secara *realtime* dengan menggunakan teknologi *IoT*. Diharapkan dengan dibuatnya alat dapat menjadi solusi dan memberi kemudahan pada para peternak puyuh dalam melakukan kontrol air minum kandang puyuh.

1.2 Pernyataan Masalah

Dari latar belakang yang disebutkan di atas dapat dipahami bahwasanya menjaga kualitas dan ketersediaan air penting bagi puyuh, maka permasalahan yang

diangkat pada penelitian ini adalah bagaimana mengontrol air minum kandang burung puyuh berbasis *Internet of Things* menggunakan metode *Fuzzy Logic*?

1.3 Tujuan Penelitian

Dari pernyataan masalah di atas, penelitian ini memiliki tujuan untuk membangun sistem otomatis untuk kontrol air minum kandang burung puyuh berbasis *Internet of Things* menggunakan metode *Fuzzy Logic*.

1.4 Batasan Masalah

Agar pembahasan yang dibahas oleh peneliti tidak menyimpang dari apa yang telah dirumuskan, maka penulis memberi batas-batas penelitian sebagai berikut:

1. Data yang diambil berasal dari sensor *Ultrasonic HC SR04* dan sensor *Turbidity SKU SEN0189*.
2. Tempat penampungan air yang digunakan pada *prototype* ini memiliki tinggi 15 cm.

1.5 Manfaat Penelitian

Adapun manfaat yang dapat diambil dari penelitian ini nantinya adalah sebagai berikut :

1. Bagi penulis adalah untuk menerapkan atau mengaplikasikan teori-teori yang telah didapat di bangku perkuliahan selama ini serta untuk memenuhi tugas akhir.
2. Bagi masyarakat khususnya peternak adalah mempermudah proses kontrol air minum kandang menggunakan kendali cerdas berbasis *Internet of Things*.

BAB II

STUDI PUSTAKA

2.1 Penelitian Terkait

(Cholilulloh, Syauqi & Tibyani, 2018) Dalam penelitiannya yang berjudul Implementasi Metode *Fuzzy* Pada Kualitas Air Kolam Bibit Lele Berdasarkan Suhu dan Kekeruhan, mengatakan bahwa kualitas air kolam yang kurang baik dapat menyebabkan kesehatan ikan akan terganggu, pertumbuhan ikan akan menjadi lambat dan bisa mengancam kegagalan panen. Faktor-faktor yang dapat mempengaruhi kualitas air meliputi suhu dan kekeruhan. Suhu air untuk pembudidayaan bibit ikan lele berkisar $23^{\circ}\text{C} - 30^{\circ}\text{C}$ dan kekeruhan air dapat dilihat dari kejernihan air berdasarkan warnanya. Warna keruh pada air menyebabkan pantulan cahaya matahari tidak dapat masuk ke dalam air. Padahal pada proses pertumbuhan bibit ikan lele dibutuhkan cahaya sinar matahari yang cukup. Pada penelitian ini di gunakan metode *Fuzzy takagi-sugeno* untuk mengukur kualitas air pada kolam bibit lele berdasarkan suhu dan kekeruhan. Parameter yang di gunakan ada dua yaitu parameter suhu dan sensor keruh. *Output* dari penelitian ini adalah kontrol pompa yang berfungsi untuk mengganti atau mengisi air kolam. Air kolam akan di ganti apabila sudah melewati batas *range* kekeruhan dan di isi apabila melebihi *range* suhu dingin ataupun panas yang telah ditentukan. Berdasarkan hasil pengujian sistem ini dapat membaca nilai suhu dengan hasil rata-rata 2,39 % dan nilai kekeruhan dapat mengukur tingkat kepekatan air, serta dapat mengontrol kualitas air dengan baik.

(Qirom, Niam & Sungkar, 2019) Pada penelitiannya yang berjudul Sistem *Monitoring* Pengairan Otomatis Dengan Metode Logika *Fuzzy*, menjelaskan bahwa

sistem pengairan merupakan kebutuhan utama dalam dunia pertanian, khususnya tanaman padi. Banyaknya kendala dalam pengairan konvensional diperlukan pengairan secara otomatis. Beberapa penelitian pengairan otomatis hanya menggunakan satu atau 2 parameter saja dan hanya menggunakan *Fuzzy* saja atau *Internet of Things* saja. Metode yang di tawarkan dalam penelitian ini menggunakan logika *Fuzzy* dengan 3 *input* dan menggabungkan sistem *monitoring* secara *real time* berbasis *Internet of Things*. Tujuan dari penelitian ini adalah untuk mengetahui efektivitas logika *Fuzzy* dengan tiga *input* untuk mengontrol sistem pengairan otomatis dengan pemantauan secara *realtime* dengan *Internet of Things*. Data diperoleh dengan melakukan pengujian di waktu pagi, siang, sore, malam dan dengan perlakuan panas dan hujan kemudian dibandingkan dengan perhitungan Matlab. Dari pengujian alat diperoleh rata-rata presisi dari perbandingan alat dengan perhitungan adalah sebesar 77,13%.

(Mindriawan, Arimbawa, & Wijaya, 2018) Dalam penelitian dengan judul Implementasi *Internet of Things* Pada Sistem *Monitoring* Suhu dan Kontrol Air Pada Kandang Burung Puyuh Petelur dengan Menggunakan Protokol MQTT, dalam penelitian ini penulis ingin membuat sebuah alat untuk mengontrol kondisi penampungan air dengan memanfaatkan *Wireless sensor Network* berbasis *Internet of Thing* menggunakan protokol MQTT (*Message Queuing Telemetry Transport*). Sensor yang dihubungkan pada penampungan air adalah sensor ketinggian air, sensor jarak, sensor suhu yang kemudian data dari sensor-sensor tersebut dikirimkan ke Broker melalui *Wi-Fi*, lalu pada kran penampungan air akan digunakan dua *servo* yang dihubungkan dengan internet sehingga pengontrolan air

dapat dilakukan secara *online* dan peternak juga mendapat notifikasi waktu air pada penampungan penuh atau habis

(Achmad Fauzi, 2017) Pada penelitian yang berjudul Sistem Kontrol Suhu Ruangan Pada Inkubator Anak Ayam Menggunakan *ESP WEMOS D1* Berbasis *IoT*, dijelaskan bahwa dengan semakin majunya ilmu teknologi saat ini dan meningkatnya jumlah konsumsi terhadap daging ayam. Ayam dapat dikatakan berkualitas dan sehat ditandai dengan terjaganya suhu kandang ayam yang stabil dan mendapatkan asupan makanan yang cukup dan tepat waktu. Oleh karena itu, dalam pembuatan alat ini peneliti menggunakan *mikrokontroler ESP Wemos D1*, Sensor suhu (*DHT11*). Pengembangan alat ini sangat berguna sekali bagi para peternak ayam, karena dapat membantu melakukan kontrol suhu kandang anak ayam (*incubator*) dengan penghangat dan pendingin melalui gadget via aplikasi yang dapat di akses dari gadget yang sedang digunakan. Pengontrolan suhu hangat dilakukan dengan menggunakan *dimmer* untuk mengatur panas pada 2 buah lampu dan pengontrolan suhu dingin menggunakan kipas blower 1 buah yang terhubung ke *relay* sebagai saklar otomatis semua *output* alat dan kegiatan alat dapat dilihat melalui *smartphone* Android atau iPhone yang sedang digunakan.

(Muta'affif, dkk 2017) Pada penelitian berjudul Sistem Kendali Peternakan Jarak Jauh Berbasis *Internet of Things*, menjelaskan bahwa Peternakan merupakan satu bidang yang sangat berpotensi untuk dikembangkan di Indonesia. Dengan adanya penerapan aspek teknologi berupa *internet of things*, para peternak dapat melakukan *monitoring* dan mengontrol peternakan meskipun terpaut jarak yang sangat jauh sekalipun hanya dengan berbekal *smartphone*. Sehingga peternak yang berada di rumahnya dapat melakukan *monitoring* dan kontrol parameter kadar

pakan dan air minum. Dengan adanya sistem ini maka kuantitas konsumsi ternak terhadap pakan dan air minum dapat dikontrol sehingga dapat lebih efisien dan efektif. Pengontrolan ini juga menjadikan hewan ternak lebih sehat sehingga pertumbuhannya semakin cepat dan berbobot sesuai dengan harapan. Harapannya sistem ini nantinya dapat diaplikasikan di dunia peternakan, sehingga banyak orang yang berminat untuk beternak dan peternakan di Indonesia pun dapat mendominasi dan menyuplai konsumsi hewan ternak sendiri bahkan bisa ekspor ke luar negeri.

2.2 Landasan Teori

Berikut adalah landasan teori yang digunakan oleh peneliti sebagai acuan dasar dalam pengembangan sistem pada penelitian ini.

2.2.1 *IoT (Internet of Things)*

Menurut Arafat, M. K. (2016) *Internet of Things* atau dikenal juga dengan singkatan *IoT*, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus menerus yang memungkinkan kita untuk menghubungkan mesin, peralatan, dan benda fisik lainnya dengan sensor jaringan dan *actuator* untuk memperoleh data dan mengelola kinerjanya sendiri, sehingga memungkinkan mesin untuk berkolaborasi dan bahkan bertindak berdasarkan informasi baru yang diperoleh secara independen. *Internet of Things* atau sering disebut *IoT* adalah sebuah gagasan di mana semua benda di dunia nyata dapat berkomunikasi satu dengan yang lain sebagai bagian dari satu kesatuan sistem terpadu menggunakan jaringan internet sebagai penghubung, misalnya CCTV yang terpasang di sepanjang jalan dihubungkan dengan koneksi internet dan di satukan di ruang kontrol yang jaraknya mungkin puluhan kilometer.

Atau sebuah rumah cerdas yang dapat diatur lewat *smartphone* dengan bantuan koneksi internet. Pada dasarnya perangkat *IoT* terdiri dari sensor sebagai media pengumpul data, sambungan internet sebagai media komunikasi dan server sebagai pengumpul informasi yang diterima sensor dan untuk analisa. Ide awal *Internet of Things* pertama kali dimunculkan oleh Kevin Ashton pada tahun 1999 di salah satu presentasinya. Kini banyak perusahaan besar mulai mendalami *Internet of Things* sebut saja Intel, Microsoft, Oracle, dan banyak lainnya. Banyak yang memprediksi bahwa pengaruh *Internet of Things* adalah “*the next big thing*” di dunia teknologi informasi, hal ini karena *IoT* menawarkan banyak potensi yang bisa digali. Contoh sederhana manfaat dan implementasi dari *Internet of Things* misalnya adalah kulkas yang dapat memberitahukan kepada pemiliknya via SMS atau email tentang makanan dan minuman apa saja yang sudah habis dan harus distok lagi.

2.2.2 *Arduino UNO*

Arduino UNO adalah sebuah *board mikrokontroler* yang didasarkan pada ATmega328. *Arduino UNO* mempunyai 14 *pin digital input/output* (6 di antaranya dapat digunakan sebagai *output PWM*), 6 *input analog*, sebuah osilator Kristal 16 *MHz*, sebuah koneksi USB, sebuah *power jack*, sebuah *ICSP header*, dan sebuah tombol reset. *Arduino UNO* memuat semua yang dibutuhkan untuk menunjang *mikrokontroler*, mudah menghubungkannya ke sebuah komputer dengan sebuah kabel USB atau menyuplainya dengan sebuah adaptor AC ke DC atau menggunakan baterai untuk memulainya (Hidyatama, 2013).

Untuk lebih jelas tentang bagaimana bentuk fisik dari *Arduino Uno* dapat dilihat pada gambar 2.1 berikut ini.



Gambar 2.1 Bentuk Fisik Arduino UNO

Sumber: <https://store.arduino.cc/usa/arduino-uno-rev3/>

Tabel 2.1 Spesifikasi Arduino UNO

Mikrokontroler	Atmega 328
Tegangan Pengoperasian	5V
Tegangan Out yang disarankan	7 - 12 V
Batas Tegangan Input	6 – 20 V
Jumlah Pin I/O Digital	14 <i>pin</i> digital (6 di antaranya menyediakan keluaran PWM)
Jumlah Pin <i>input</i> Analog	6 <i>pin</i>
Arus DC tiap <i>pin</i> I/O	40 mA
Arus DC untuk <i>pin</i> 3,3 V	50 mA
Memory Flash	32 KB (Atmega) sekitar 0,5 KB digunakan oleh <i>bootloader</i>
SRAM	2 KB (ATmega 328)
EPROM	1 KB (ATmega 328)
Clock Speed	16 Hz

2.2.3 NodeMCU-ESP8266

NodeMcu merupakan sebuah *opensource* platform *IoT* dan pengembangan *Kit* yang menggunakan bahasa pemrograman *Lua* untuk membantu *programmer* dalam

membuat *prototype* produk *IoT* atau bisa dengan memakai *sketch* dengan *arduino* IDE. Pengembangan *Kit* ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (*Pulse Width Modulation*), IIC , 1-Wire dan ADC (*Analog to Digital Converter*) semua dalam satu *board*. Keunikan dari *NodeMcu* ini sendiri yaitu *board* yang berukuran sangat kecil yaitu panjang 4.83cm, lebar 2.54cm, dan dengan berat 7 gram. Tapi walaupun ukurannya yang kecil, *board* ini sudah dilengkapi dengan fitur *wifi* dan *firmware*-nya yang bersifat *opensource*.

Penggunaan *NodeMcu* lebih menguntungkan dari segi biaya maupun efisiensi tempat, karena *NodeMcu* yang ukurannya kecil, lebih praktis dan harganya jauh lebih murah dibandingkan dengan *Arduino Uno*. *Arduino Uno* sendiri merupakan salah satu jenis *mikrokontroler* yang banyak diminati dan memiliki bahasa pemrograman C++ sama seperti *NodeMcu*, namun *Arduino Uno* belum memiliki modul *wifi* dan belum berbasis *IoT*. Untuk dapat menggunakan WIFI, *Arduino Uno* memerlukan perangkat tambahan berupa *wifi shield*. *NodeMcu* merupakan salah satu produk yang mendapatkan hak khusus dari *Arduino* untuk dapat menggunakan aplikasi *Arduino* sehingga bahasa pemrograman yang digunakan sama dengan *board Arduino* pada umumnya (Tulle, 2017).

Untuk lebih jelas tentang bagaimana bentuk fisik dari *NodeMCU-ESP8266* dapat dilihat pada gambar 2.2 berikut ini.



Gambar 2.2 Bentuk Fisik *NodeMCU-ESP8266*

Sumber: <https://esp8266-shop.com/product/nodemcu-esp8266-esp-12e/>

Tabel 2.2 Spesifikasi *NodeMCU-ESP8266*

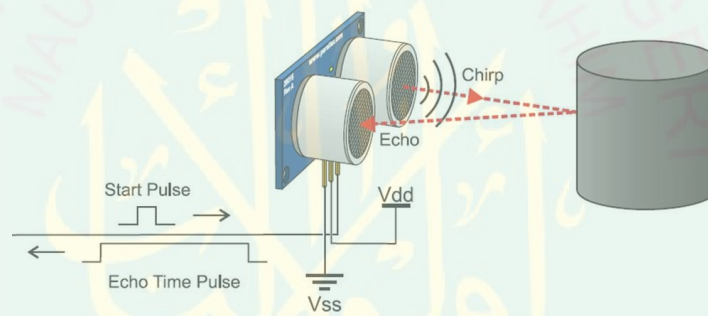
Vendor Pembuat	<i>Amica</i>
Tipe ESP8266	ESP-12E
USB port	<i>Micro USB</i>
GPIO Pin	13
ADC	1 pin (10 bit)
USB to Serial Converter	CP2102
Power Input	5 Vdc
Ukuran Modul	47 x 24 mm

2.2.4 Sensor *Ultrasonic HC SR-04*

Sensor *ultrasonic* adalah sensor yang bekerja berdasarkan prinsip pantulan gelombang suara dan digunakan untuk mendeteksi keberadaan suatu objek atau benda tertentu di depan frekuensi kerja pada daerah di atas gelombang suara dari 20 kHz hingga 2 MHz. Sensor *ultrasonic* terdiri dari dua unit, yaitu unit pemancar dan unit penerima. Struktur unit pemancar dan penerima sangat sederhana, yakni sebuah kristal *piezoelectric* dihubungkan dengan mekanik jangkar dan hanya dihubungkan dengan diafragma penggetar tegangan bolak-balik yang memiliki frekuensi kerja 20 kHz hingga 2 MHz. Struktur atom dari Kristal *piezoelectric*

menyebabkan berkontraksi mengembang atau menyusut, sebuah polaritas tegangan yang diberikan dan ini disebut dengan efek *piezoelectric* pada sensor *ultrasonic* (Suprianto, 2017).

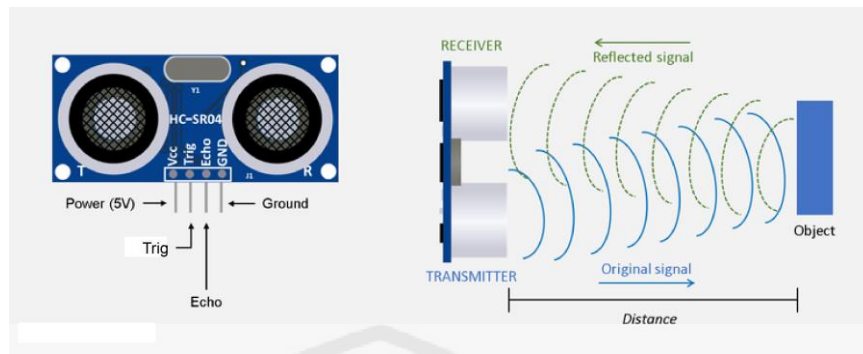
Pantulan gelombang *ultrasonic* terjadi bila ada objek tertentu dan pantulan gelombang *ultrasonic* akan diterima kembali oleh unit sensor penerima. Selanjutnya unit sensor penerima akan menyebabkan diafragma penggetar akan bergetar dan efek *piezoelectric* menghasilkan sebuah tegangan bolak-balik dengan frekuensi yang sama. Untuk lebih jelas tentang prinsip kerja dari sensor *ultrasonic* dapat dilihat prinsip dari sensor *Ultrasonic* pada gambar 2.3 berikut ini :



Gambar 2.3 Prinsip Sensor *Ultrasonic*

Sumber: https://users.encs.concordia.ca/~bwgordon/arduino_lab3.html

Besar amplitudo sebuah sinyal elektrik yang dihasilkan sensor penerima tergantung dari jauh dekatnya sebuah objek yang akan dideteksi serta kualitas dari sensor pemancar dan sensor penerima. Proses *censoring* yang dilakukan pada sensor ini menggunakan metode pantulan untuk menghitung jarak antara sensor dengan objek sasaran. Prinsip pemantulan dari sensor *ultrasonic* dapat dilihat pada gambar 2.4 berikut ini:



Gambar 2.4 Prinsip Pemantulan Ultrasonic

Sumber: <https://diyi0t.com/ultrasonic-sensor-tutorial-for-arduino-and-esp8266/>

Tabel 2.3 Spesifikasi Sensor Ultrasonic HC-SR04

Operating Voltage	5V
Operating Current	15 mA
Operating Frequency	40 KHz
Max. Range	4 m
Nearest Range	2 cm
Measuring Angle	15 Degrees
Input Trigger Signal	10us min. TTL pulse
Output Echo Signal	TTL level signal, proportional to distance
Board Dimensions	1-13/16" X 13/16" X 5/8"
Board Connection	4 X 0.1" Pitch Right Angle Header Pins

2.2.5 Turbidity Sensor SKU SEN0189

Sensor ini berfungsi mengukur kekeruhan dari air. Sensor kekeruhan yang digunakan adalah Analog Turbidity Sensor SKU SEN0189 pabrikan *dfrobot*. Sensor ini berupa *optical* sensor yang mengukur kekeruhan menggunakan pembiasan gelombang antara *phototransistor* dan *IR LED (dioda)*. Dengan menggunakan *phototransistor* dan *dioda*, dapat diukur jumlah cahaya yang datang dari sumber cahaya (*IR LED*) ke penerima cahaya (*phototransistor*), untuk menghitung

kekeruhan air. Pada prinsipnya, *phototransistor* merupakan salah satu komponen yang berfungsi sebagai detektor cahaya yang dapat mengubah efek cahaya menjadi sinyal listrik. Apabila cahaya yang dipancarkan oleh *IR LED* menuju *phototransistor* terhalang seperti oleh air keruh maka akan mengakibatkan perubahan tegangan pada *phototransistor* (Fakhri, dkk. 2017).

Sinyal masukan berupa emisi cahaya yang dikeluarkan oleh *LED* akan ditangkap oleh *phototransistor*. Cahaya yang dihasilkan *LED* berdasarkan tingkat emisi cahaya yang berbeda jika di air bersih dan air yang keruh. Prinsipnya pada air yang keruh terdapat bahan-bahan anorganik atau organik yang bisa mengabsorpsi emisi cahaya *LED* sehingga intensitas cahaya menjadi berkurang, maka tegangan yang dibaca oleh *phototransistor* menjadi berbeda antara air yang bersih dan air yang keruh. Ketika sejumlah cahaya melalui air, jumlah cahaya yang lewat pada air tergantung banyaknya kotoran/keruhnya air. Jika jumlah kotoran meningkat, cahaya yang menembus air berkurang dan begitu sebaliknya (Hedlyni, 2011).

Untuk lebih jelas tentang bagaimana bentuk fisik dari Sensor *Turbidity SKU SEN0189* dapat dilihat pada gambar 2.5 berikut ini.



Gambar 2.5 Bentuk Fisik *Turbidity Sensor SKU SEN0189*

Sumber: <https://www.dfrobot.com/product-1394.html>

Tabel 2.4 Spesifikasi *Turbidity Sensor SKU SEN0189*

<i>Operating Voltage</i>	5V DC
<i>Operating Current</i>	40 mA
<i>Response Time</i>	100 Min
<i>Output Method</i>	Analog Output: 0-4.5 V Digital Output: High/Low level signal
<i>Operating Temperature</i>	5°C~90°C
<i>Storage Temperature</i>	-10°C~90°C
<i>Weight</i>	30g
<i>Adapter Dimensions</i>	38mm*28mm*10mm/1.5inches *1.1inches*0.4inches

2.2.6 Pompa Air

Pompa adalah suatu mesin/alat yang digunakan untuk menaikkan cairan dari permukaan yang lebih rendah ke permukaan yang lebih tinggi atau memindahkan cairan dari tempat yang bertekanan rendah ke tempat yang bertekanan lebih tinggi. Pompa bekerja dengan cara mentransfer energi mekanis dari suatu sumber energi luar ke cairan yang mengalir melaluinya. Pompa menaikkan energi cairan yang mengalir melaluinya, sehingga cairan tersebut dapat mengalir dari permukaan yang lebih rendah ke permukaan yang lebih tinggi maupun dari tempat bertekanan rendah ke tempat yang bertekanan lebih tinggi dan bersamaan dengan itu bisa juga mengatasi tahanan hidrolik sepanjang pipa yang dipakai (Wijono, 2017).

Untuk lebih jelas tentang bagaimana bentuk fisik dari pompa air dapat dilihat pada gambar 2.6 berikut ini.



Gambar 2.6 Bentuk Fisik Pompa Air

Sumber : <https://store.roboticsbd.com/robotics-parts/949-5v-dc-mini-water-pump-micro-submersible-motor-pump-25-6v-120lh-robotics-bangladesh.html>

2.2.7 Relay

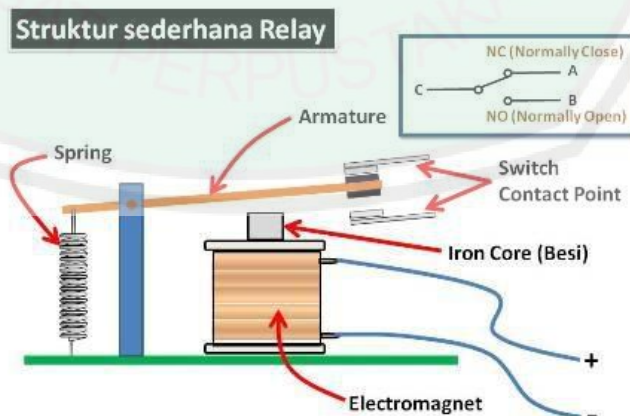
Relay adalah Saklar (*Switch*) yang dioperasikan secara listrik dan merupakan komponen *Electromechanical* (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (*Coil*) dan Mekanikal (seperangkat Kontak Saklar/*Switch*). *Relay* menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Sebagai contoh, dengan *Relay* yang menggunakan Elektromagnet 5V dan 50 mA mampu menggerakkan *Armature Relay* (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A. (Nugrahanto, 2017) .Untuk mengetahui bentuk *relay* dapat dilihat pada gambar 2.7 berikut ini.



Gambar 2.7 Bentuk Fisik *Relay*

Sumber: <https://www.immersa-lab.com/pengertian-relay-fungsi-dan-cara-kerjarelay.htm>

Pada *relay*, terdapat besi atau yang disebut dengan nama iron core dililit oleh sebuah kumparan yang berfungsi sebagai pengendali. Sehingga ketika kumparan *coil* diberikan arus listrik maka akan menghasilkan gaya elektromagnet. Gaya tersebut selanjutnya akan menarik armature untuk pindah posisi dari *normally close* ke *normally open*. Dengan demikian saklar menjadi pada posisi baru *normally open* yang dapat menghantarkan arus listrik. Ketika armature sudah tidak dialiri arus listrik lagi maka ia akan kembali pada posisi awal, yaitu *normally close*. Berikut adalah penampakan cara kerja *relay*.



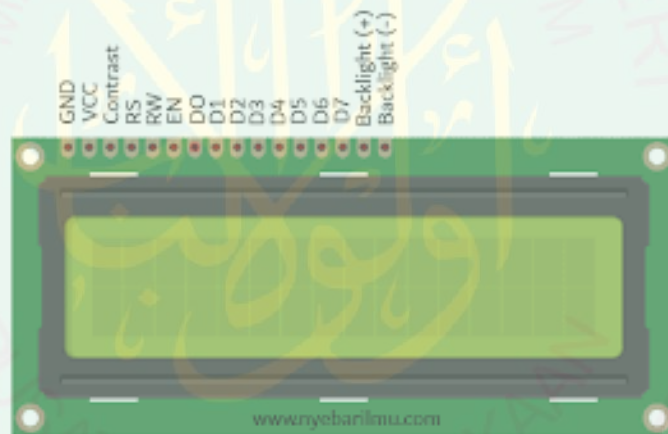
Gambar 2.8 Cara Kerja *Relay*

Sumber: <https://www.immersa-lab.com/pengertian-relay-fungsi-dan-cara-kerjarelay.htm>

2.2.8 LCD (*Liquid Crystal Display*)

LCD merupakan sebuah alat yang berfungsi untuk menampilkan suatu ukuran besaran atau angka, sehingga dapat dilihat dan diketahui melalui tampilan layar kristalnya. Di mana penggunaan *LCD* dalam penelitian ini menggunakan *LCD* dengan 16x2 karakter (2 baris 16 karakter). *LCD* 16x2 memiliki 16 nomor *pin*, di mana setiap *pin* memiliki tanda simbol dan juga fungsi-fungsinya. *LCD* 16x2 ini beroperasi pada *power supply* 5V, tetapi juga dapat beroperasi pada *power supply* 3V (Budiyanto, 2012).

Untuk lebih jelas tentang bagaimana bentuk fisik dari *LCD* (*Liquid Crystal Display*) dapat dilihat pada gambar 2.8 berikut ini.



Gambar 2.9 Bentuk Fisik *LCD* (*Liquid Crystal Display*)

Sumber: <https://www.nyebarilmu.com/cara-mengakses-modul-display-lcd-16x2/>

2.2.9 Fuzzy Logic

Fuzzy secara bahasa diartikan sebagai kabur atau samar-samar. Suatu nilai dapat bernilai besar atau salah secara bersamaan. Dalam *Fuzzy* dikenal derajat keanggotaan yang memiliki rentang nilai 0 (nol) hingga 1(satu). Berbeda dengan himpunan tegas yang memiliki nilai 1 atau 0 (Ismail, Ikhsan. 2015). Logika *Fuzzy*

digunakan untuk menerjemahkan suatu besaran yang diekspresikan menggunakan bahasa (*linguistic*). Misalnya, besaran kecepatan laju kendaraan yang diekspresikan dengan pelan, agak cepat, cepat, dan sangat cepat.

Fuzzy Logic pertama kali diperkenalkan oleh Jan Lukasiewicz pada tahun 1920-an yaitu sebagai teori kemungkinan. Logika kemungkinan ini memperluas jangkauan dari nilai kebenaran untuk semua bilangan riil pada interval antara 0 dan 1. Selanjutnya, Max Black melakukan penelitian lebih lanjut pada tahun 1930-an dalam penelitiannya tentang ke tidak jelasan (*vagueness*) sebuah latihan pada analisis logis.

Dan pada tahun 1965, profesor dan kepala departemen teknik elektrik di Universitas California, Barkeley, Lotfi Zadeh, menemukan kembali, mengidentifikasi, mengeksplorasi, mempromosikan, dan berjuang untuk *Fuzzy logic*. Profesor Zadeh memperluas ruang kerja teori kemungkinan menjadi sistem logika matematika formal dan konsep baru untuk mengaplikasikan istilah bahasa alami pada penelitiannya, yaitu '*Fuzzy Sets*'. Logika baru ini dinamakan *Fuzzy Logic*. *Fuzzy Logic* banyak digunakan karena mirip dengan cara berpikir manusia. Sistem *Fuzzy Logic* dapat merepresentasikan pengetahuan manusia dalam bentuk matematis dengan menyerupai cara berpikir manusia. (Budiharto dan Suhartono, 2014)

2.2.9.1 Himpunan *Fuzzy*

Pada himpunan tegas (*Crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $A[x]$, memiliki 2 kemungkinan, yaitu:

- a. satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan.
- b. nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

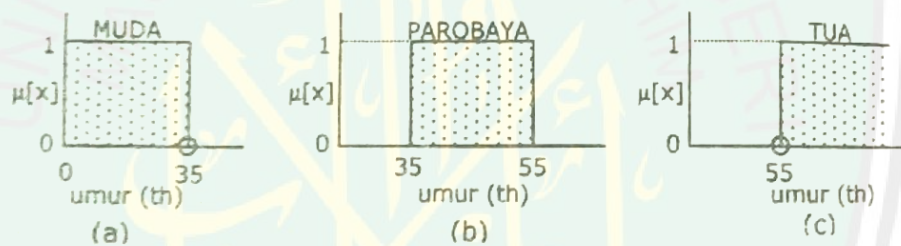
Misalkan variabel umur dibagi menjadi 3 kategori, yaitu:

MUDA umur < 35 tahun

PAROBAYA $35 \leq \text{umur} \leq 55$ tahun

TUA umur > 55 tahun

Nilai keanggotaan secara grafis, himpunan MUDA, PAROBAYA, dan TUA ini dapat dilihat pada gambar 2.9.



Gambar 2.10 Himpunan : MUDA, PAROBAYA, dan TUA

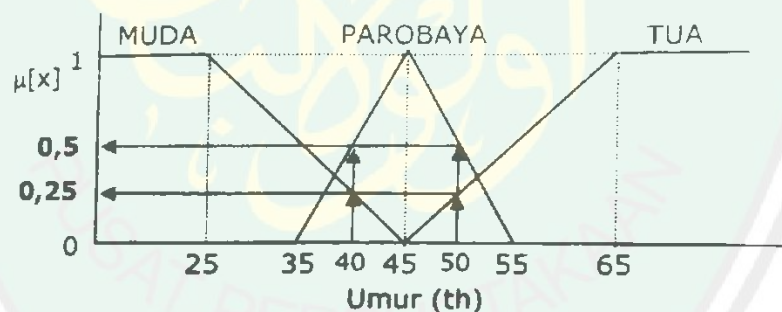
Sumber: (Kusumadewi & Purnomo, 2010)

Menurut gambar di atas dapat dilihat bahwa:

- a. apabila seseorang berusia 34 tahun, maka dapat dikatakan MUDA ($\text{MUDA}[34]=1$);
- b. apabila seseorang berusia 35 tahun, maka dapat dikatakan TIDAK MUDA ($\text{MUDA}[35]=0$);
- c. apabila seseorang berusia 35 tahun kurang 1 hari, maka dapat dikatakan TIDAK MUDA ($\text{MUDA}[35 \text{ tahun} - 1 \text{ hari}]=0$);
- d. apabila seseorang berusia 35 tahun, maka dapat dikatakan PAROBAYA ($\text{PAROBAYA}[35]=1$);

- e. apabila seseorang berusia 34 tahun, maka dapat dikatakan TIDAK PAROBAYA ($\text{PAROBAYA}[34]=0$);
- f. apabila seseorang berusia 36 tahun, maka dapat dikatakan PAROBAYA ($\text{PAROBAYA}[36]=1$);
- g. apabila seseorang berusia 35 tahun kurang 1 hari, maka dapat dikatakan TIDAK PAROBAYA ($\text{PAROBAYA}[35 \text{ tahun} - 1 \text{ hari}]=0$)

Dari sini bisa dikatakan bahwa pemakaian himpunan *crisp* untuk menyatakan umur sangat tidak adil, adanya perubahan kecil saja pada suatu nilai mengakibatkan perbedaan kategori yang cukup signifikan. Himpunan *Fuzzy* digunakan untuk mengantisipasi hal tersebut. Seseorang dapat masuk dalam 2 himpunan berbeda, MUDA dan PAROBAYA, PAROBAYA dan TUA, dsb. Seberapa besar eksistensinya dalam himpunan tersebut dapat dilihat pada gambar 2.10 nilai keanggotaannya.



Gambar 2.11 Himpunan *Fuzzy* Untuk Variabel Umur
Sumber: (Kusumadewi & Purnomo, 2010)

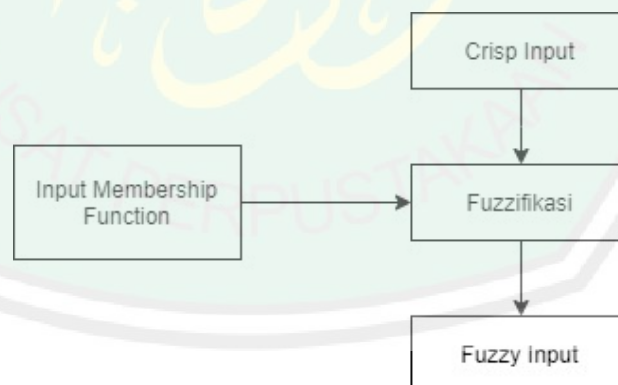
2.2.9.2 Domain

Domain merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai suatu domain bisa saja bernilai positif ataupun negatif. Berikut contoh domain himpunan *Fuzzy*. (Kusumadewi & Purnomo, 2010)

- MUDA = [0 45]
- PAROBAYA = [35 55]
- TUA = [45 ∞]
- DINGIN = [0 20]
- SEJUK = [15 25]
- NORMAL = [20 30]
- HANGAT = [25 35]
- PANAS = [30 40]

2.2.9.3 Fuzzifikasi

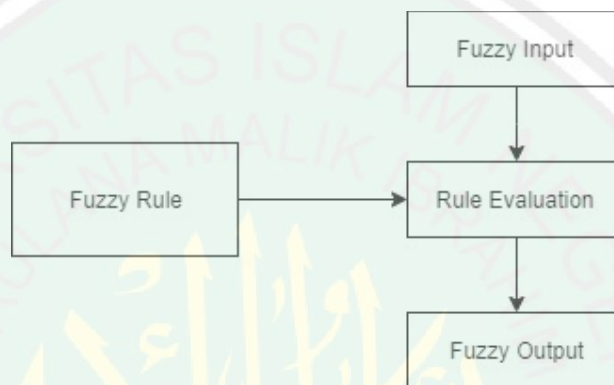
Prosedur pada *fuzzifikasi* digunakan untuk mengubah variabel numerik menjadi variabel linguistik (variabel *Fuzzy*). Nilai yang didapatkan yaitu berupa *data error* dan *delta error* yang diolah oleh *control logika Fuzzy*. Terdapat juga suatu fungsi keanggotaan (*membership function*) yang harus disusun. Dari nilai *error* dan *delta error* yang didapatkan tadi selanjutnya dihitung dan dikelompokkan pada derajat keanggotaan masing-masing nilainya. Proses *fuzzifikasi* dapat dilihat pada gambar 2.11 berikut.



Gambar 2.12 Proses *Fuzzifikasi*

2.2.9.4 Rule Base

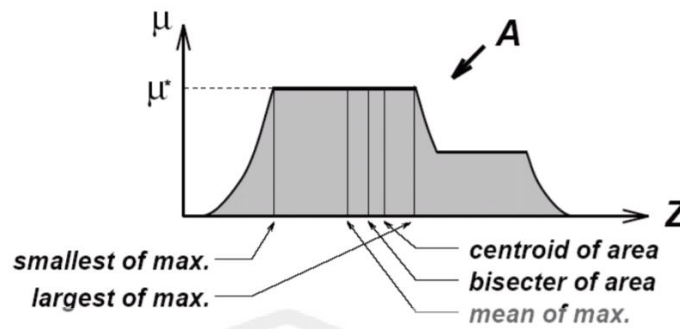
Rule base yang merupakan sekelompok aturan pada *Fuzzy* yang memproses *input Fuzzy* sampai menjadi *output* nilai *Fuzzy*. *Rule* ini merupakan dasar dalam pengambilan keputusan (*inference process*) untuk menghasilkan suatu *output Fuzzy* dari *input error* dan *delta error* dengan menggunakan *rule* yang telah ditetapkan. Proses *Rule Base* dapat dilihat pada gambar 2.12 berikut.



Gambar 2.13 Proses *Rule Base*

2.2.9.5 Defuzzifikasi

Input dari proses *defuzzifikasi* adalah suatu himpunan *Fuzzy* yang diperoleh dari komposisi aturan-aturan *Fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *Fuzzy* tersebut, sehingga jika diberikan suatu himpunan *Fuzzy* dalam *range* tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu sebagai keluarannya. Ada beberapa metode *defuzzifikasi* pada komposisi aturan *Mamdani*, di antaranya yaitu metode COA, *bisector*, MOM, LOM, dan SOM seperti pada Gambar 2.13



Gambar 2.14 Metode Defuzzifikasi

a. Metode COA

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat daerah Fuzzy, secara umum dirumuskan pada persamaan 2 untuk variabel *continue* dan persamaan 3 untuk variabel diskrit.

$$z^* = \frac{\int z \mu(z) dz}{\int \mu(z) dz}$$

b. Metode Bisector

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai pada domain Fuzzy yang memiliki nilai keanggotaan separuh dari jumlah total nilai keanggotaan pada daerah Fuzzy. Secara umum dituliskan pada persamaan 4

$$\int_{\alpha}^{z_{BOA}} \mu(z) dz = \int_{z_{BOA}}^{\beta} \mu(z) dz$$

Di mana:

$$\alpha = \min\{z \mid z \in Z\}$$

$$\beta = \max\{z \mid z \in Z\}$$

c. MOM

Pada solusi ini, solusi *crisp* diperoleh dengan cara mengambil nilai rata-rata domain yang memiliki nilai keanggotaan maksimum.

d. LOM

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terbesar dari domain yang memiliki nilai keanggotaan maksimum.

e. SOM

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.

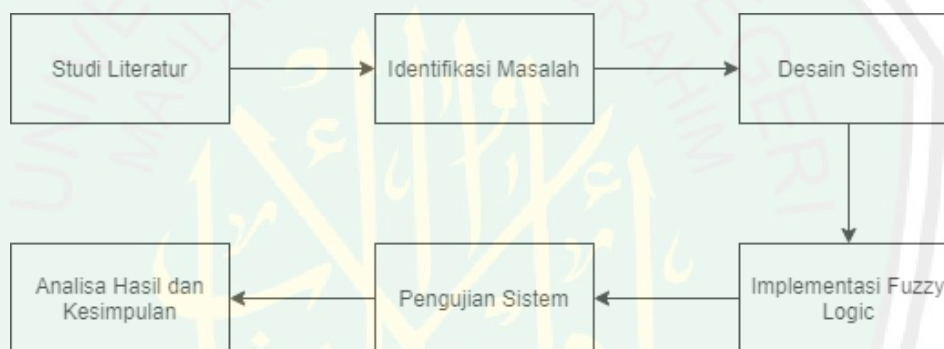


BAB III

METODE PENELITIAN

3.1 Desain Penelitian

Pada bab ini akan dijelaskan tentang tahapan-tahapan penelitian yang dilalui oleh peneliti, yaitu meliputi studi literatur, merumuskan masalah, desain sistem, implementasi *Fuzzy Logic* kepada sistem, pengujian sistem, analisa hasil & kesimpulan. Sistem yang dibangun adalah sistem untuk kontrol air minum kandang puyuh menggunakan metode *Fuzzy Logic* yang berbasis *Internet of Things*. Diagram blok dari alur penelitian ini dapat dilihat pada gambar 3.1 di bawah ini :



Gambar 3.1 Diagram Blok Alur Penelitian

Peneliti melakukan studi literatur dengan tujuan untuk mengkaji hal-hal yang memiliki hubungan dengan konsep-konsep yang menunjang dalam perancangan sistem ini. Literatur-literatur yang digunakan meliputi jurnal-jurnal dan juga buku-buku. Setelah melakukan studi literatur, perumusan masalah dilakukan agar tujuan dari perancangan sistem ini dapat tercapai.

3.1.1 Perangkat Keras

Dalam perancangan sistem ini, peneliti membutuhkan beberapa perangkat keras seperti yang disebutkan sebagai berikut:

1. Laptop
2. *Smartphone Android*
3. *Arduino Uno*
4. *NodeMCU ESP-8266*
5. *Sensor Ultrasonic HC SR04 (Level)*
6. *Sensor Turbidity SKU SEN0189 (Kekeruhan)*
7. *Modul Relay*
8. Pompa Air
9. *LCD I2C*

3.1.2 Perangkat Lunak

Selain perangkat keras, peneliti juga membutuhkan perangkat lunak di antaranya:

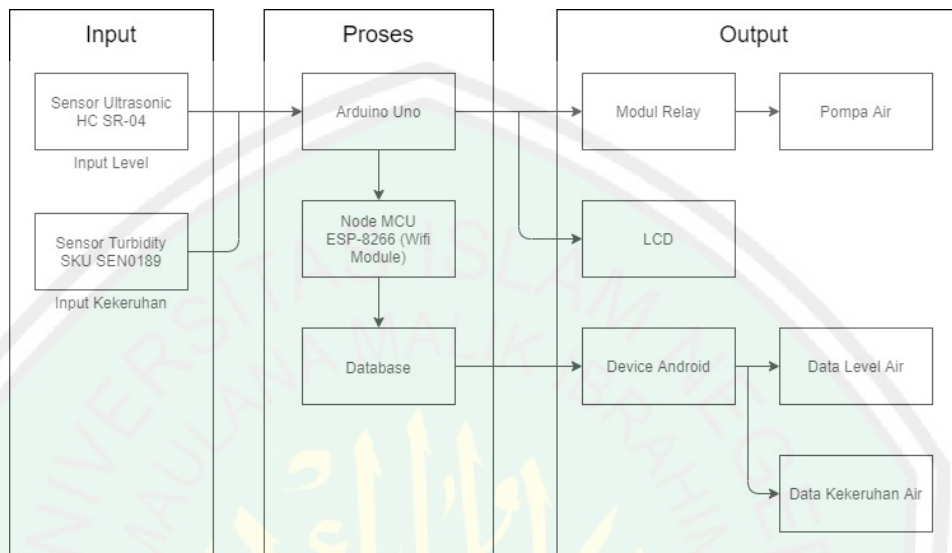
1. *Arduino IDE*
2. *Android Studio*

3.2 Desain Sistem

Pada sistem kontrol air minum kandang puyuh ini akan ditanami sistem cerdas yaitu dengan menggunakan *Fuzzy Logic*. Ada 2 sensor yang digunakan, yaitu sensor level (*Ultrasonic*) yang memberikan *output* berupa jarak antara sensor dan permukaan air sehingga menjadi indikasi tingkat ketinggian/ level air. Dan sensor kekeruhan air (*Turbidity Sensor*) yang akan memberikan *output* berupa tingkat kekeruhan air. *Output* yang diperoleh dari kedua sensor tersebut kemudian akan menjadi *input* bagi mikrokontroler yang pada penelitian ini adalah *Arduino* untuk diolah dan ditentukan *output*-nya yang berupa penambahan air atau pengurangan

air menggunakan metode *Fuzzy Logic*. Kemudian data tersebut akan dikirim ke dalam *database* dan dapat ditampilkan di *smartphone* peternak.

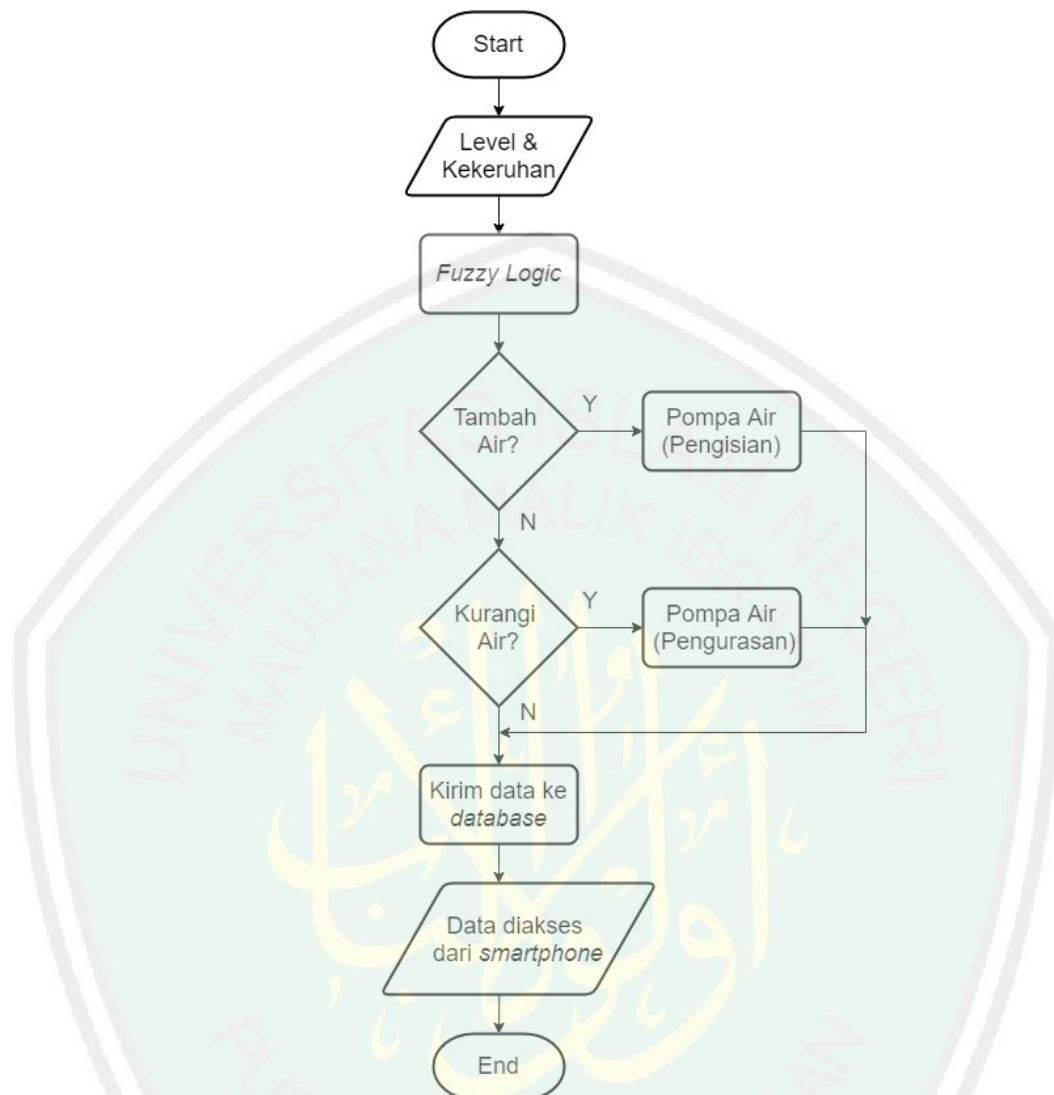
3.2.1 Blok Diagram Sistem



Gambar 3.2 Blok Diagram Sistem

Pada gambar 3.2 di atas dapat dilihat bahwa sensor *Ultrasonic HC SR-04* berperan sebagai sensor penerima *input level* dan sensor *Turbidity SKU SEN0189* berperan sebagai sensor penerima *input kekeruhan*. Data *input* tersebut kemudian diolah di dalam *Arduino Uno* menggunakan metode *Fuzzy Logic* sehingga didapat data *output* berupa waktu pengisian atau waktu pengurasan yang nantinya digunakan untuk melakukan control pada *modul relay* dan pompa air. Data juga dikirimkan kepada *NodeMCU ESP-8266* melalui *wifi* yang kemudian diteruskan ke *database*. Data yang tersimpan di *database* tersebut dapat diakses melalui *smartphone* milik peternak untuk lakukan *monitoring* level air dan kekeruhan air.

3.2.2 Flowchart Sistem



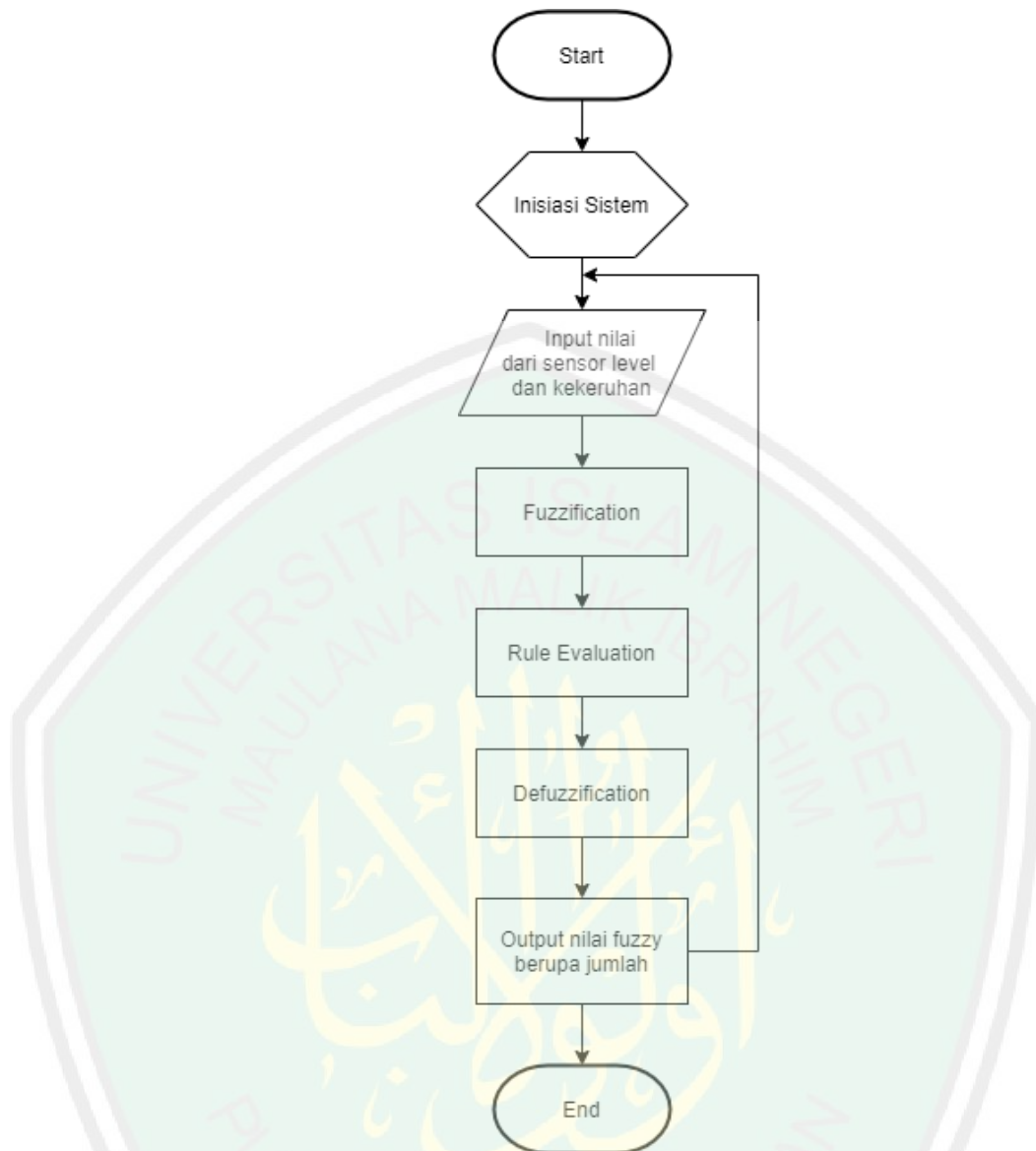
Gambar 3.3 Flowchart Sistem

Saat sistem telah berjalan maka akan diterima nilai *input* dari kedua sensor yang ada yaitu sensor level (Sensor *Ultrasonic HC SR04*) dan sensor kekeruhan air (Sensor *Turbidity SKU SEN0189*). Kemudian nilai *input* akan diproses dengan metode *Fuzzy Logic*. Dan akan dihasilkan *output* berupa waktu air akan diisi atau air akan dikuras, apabila air akan diisi maka pompa air pengisian akan menyala dan air akan diisi sesuai waktu pada *output*, begitu pun apabila air akan dikuras maka pompa air pengurasan akan menyala dan air akan dikuras sesuai waktu pada *output*.

Selanjutnya data akan dikirim ke *database* yang nantinya akan dapat diakses menggunakan *device android* peternak secara langsung.

3.3 Implementasi Metode *Fuzzy Logic*

Pada penelitian ini akan digunakan 2 proses *Fuzzy*, yang pertama adalah *Fuzzy* untuk menentukan *output* yang berupa seberapa banyak air akan ditambahkan ke wadah air minum burung puyuh dan yang kedua adalah *Fuzzy* untuk menentukan *output* berupa seberapa banyak air akan dikurangkan dari wadah air minum burung puyuh. Data *input* didapat dari 2 sensor berbeda yaitu sensor *Ultrasonic HC SR04* dan sensor *Turbidity SKU SEN0189*. Kemudian data *input* yang berbentuk *crisp input* diproses pada *fuzzifikasi*, yaitu proses membuat bilangan renyah (*crisp*) memiliki nilai *Fuzzy* sehingga *input* dapat dikelompokkan dalam himpunan *Fuzzy* yang tepat. Selanjutnya dengan nilai *Fuzzy* dari data *input* tadi disusun *rule* pada sistem inferensi dengan tujuan menjadi rujukan yang akan diterapkan dalam memperjelas hubungan antara variabel *input* dan variabel yang diproses. Dalam implementasinya, *rule* umumnya memakai “*IF-THEN*”. Data yang telah diolah oleh *rule* tadi akan memperoleh hasil berupa nilai *output fuzzy*. Namun nilai yang didapat belum bisa digunakan langsung karena memerlukan proses *defuzzifikasi* dahulu. *Defuzzifikasi* adalah sebuah proses diubahnya variabel *output fuzzy* menjadi data jelas (*crisp*) yang dapat digunakan dalam menentukan seberapa banyak air akan diisi atau seberapa banyak air akan dikuras. Berikut merupakan gambar alur proses *Fuzzy Logic* :



Gambar 3.4 Flowchart Sistem Kontrol Fuzzy Logic

3.3.1 Fuzzifikasi

Pada penelitian ini fungsi keanggotaan didapatkan dari pengukuran dan uji coba dengan dasar sumber yaitu wawancara dengan peternak. Terdapat 2 variabel *input* yaitu variabel level dan variabel kekeruhan air. Tiap *input* mempunyai nilai *linguistic* tersendiri. Pada variabel level terdapat nilai *linguistic* berupa rendah, normal dan tinggi. Dan pada variabel kekeruhan memiliki nilai *linguistic* berupa

jernih, normal dan keruh. Juga pada *output* dari hasil perhitungan metode *fuzzy* yakni variabel waktu yang memiliki nilai *lingistic* berupa sebentar, normal dan lama.

- **Variabel Level**

Variabel level dibuat dengan *range* pengukuran dari 0 sampai dengan 15 cm. Nilai ini diperoleh dari pengukuran jarak antara dasar wadah penyimpanan air dan sensor *Ultrasonic HC SR04*, yang didapatkan hasil yaitu 15 cm. Variabel level memiliki fungsi keanggotaan yaitu rendah, normal dan tinggi. Fungsi keanggotaan rendah memiliki bentuk seperti bahu kiri pada persamaan di bawah ini.

$$\mu[x, a, b, c] = \begin{cases} 0 & , x < 0 \text{ atau } x > 7,5 \\ 1 & , 0 \leq x \leq 3 \\ (7,5 - x)/(7,5 - 3) & , 3 \leq x \leq 7,5 \end{cases}$$

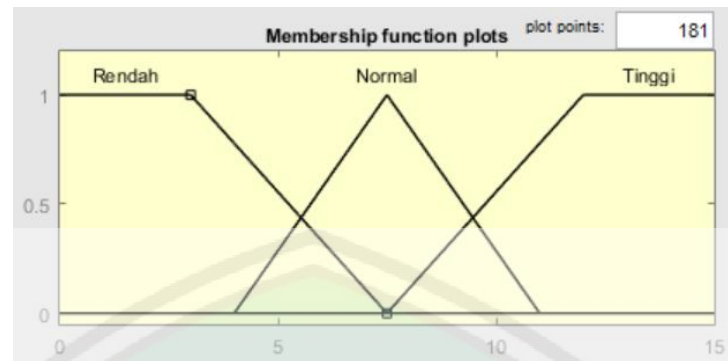
Pada fungsi keanggotaan normal memiliki kurva yang berbentuk segitiga dengan persamaan di bawah ini.

$$\mu[x, a, b, c] = \begin{cases} 0 & , x < 4 \text{ atau } x > 11 \\ (x - 4)/(7,5 - 4) & , 4 \leq x \leq 7,5 \\ (11 - x)/(11 - 7,5) & , 7,5 \leq x \leq 11 \end{cases}$$

Dan pada fungsi keanggotaan tinggi memiliki kurva yang berbentuk bahu kanan dengan persamaan di bawah ini.

$$\mu[x, a, b, c] = \begin{cases} 0 & , x < 7,5 \text{ atau } x > 15 \\ (x - 7,5)/(12 - 7,5) & , 7,5 \leq x \leq 12 \\ 1 & , 12 \leq x \leq 15 \end{cases}$$

Fungsi keanggotaan dari variabel level dapat dilihat pada gambar 3.5 berikut:



Gambar 3.5 Fungsi Keanggotaan Level

- **Variabel Kekeruhan**

Variabel kekeruhan memiliki *range* mulai dari 0 sampai dengan 30 NTU. Nilai tersebut diperoleh dari percobaan peneliti yang berdasar dari wawancara dengan peternak pada beberapa jenis air untuk mengecek nilai NTU dari beberapa jenis kekeruhan air. Variabel kekeruhan memiliki fungsi keanggotaan yaitu jernih, normal dan keruh. Fungsi keanggotaan jernih memiliki bentuk seperti bahu kiri dengan persamaan seperti di bawah ini.

$$\mu[x, a, b, c] = \begin{cases} 0 & , x < 0 \text{ atau } x > 15 \\ 1 & , 0 \leq x \leq 5 \\ (15 - x)/(15 - 5) & , 5 \leq x \leq 15 \end{cases}$$

Pada fungsi keanggotaan normal memiliki kurva yang berbentuk segitiga dengan persamaan di bawah ini.

$$\mu[x, a, b, c] = \begin{cases} 0 & , x < 10 \text{ atau } x > 20 \\ (x - 10)/(15 - 10) & , 10 \leq x \leq 15 \\ (20 - x)/(20 - 15) & , 15 \leq x \leq 20 \end{cases}$$

Dan pada keanggotaan keruh memiliki kurva yang berbentuk bahu kanan dengan persamaan di bawah ini.

$$\mu[x, a, b, c] = \begin{cases} 0 & , x < 15 \text{ atau } x > 30 \\ (x - 15)/(20 - 15) & , 15 \leq x \leq 25 \\ 1 & , 25 \leq x \leq 30 \end{cases}$$

Fungsi keanggotaan *input* level dapat dilihat pada gambar 3.6 berikut :



Gambar 3.6 Fungsi Keanggotaan Kekeruhan

- **Variabel Waktu**

Variabel waktu merupakan hasil akhir atau *output* dari perhitungan variabel level dan variabel kekeruhan, dibuat dengan *range* dari 0 sampai dengan 6 menit. Variabel waktu memiliki fungsi keanggotaan yaitu sebentar, normal dan lama. Fungsi keanggotaan sebentar memiliki bentuk seperti segitiga dengan persamaan di bawah ini.

$$\mu[x, a, b, c] = \begin{cases} 0 & , x < 0 \text{ atau } x > 3 \\ (x - 0)/(1,5 - 0) & , 0 \leq x \leq 1,5 \\ (3 - x)/(3 - 2) & , 1,5 \leq x \leq 3 \end{cases}$$

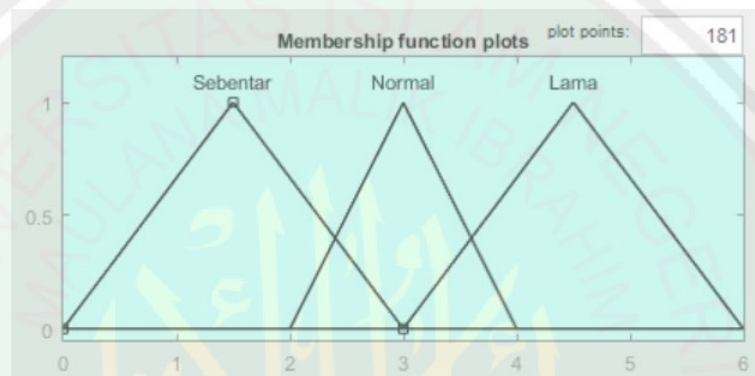
Pada fungsi keanggotaan normal memiliki kurva yang berbentuk segitiga dengan persamaan di bawah ini.

$$\mu[x, a, b, c] = \begin{cases} 0 & , x < 2 \text{ atau } x > 4 \\ (x - 2)/(3 - 2) & , 2 \leq x \leq 3 \\ (4 - x)/(4 - 3) & , 3 \leq x \leq 4 \end{cases}$$

Dan pada fungsi keanggotaan keruh memiliki kurva yang berbentuk bahu kanan dengan persamaan di bawah ini.

$$\mu[x, a, b, c] = \begin{cases} 0 & , x < 3 \text{ atau } x > 6 \\ (x - 3)/(4,5 - 3) & , 3 \leq x \leq 4,5 \\ (6 - x)/(6 - 4,5) & , 4,5 \leq x \leq 6 \end{cases}$$

Adapun Gambar 3.7 di bawah ini merepresentasikan nilai untuk tiap fungsi keanggotaan waktu.



Gambar 3.7 Fungsi Keanggotaan Waktu

3.3.2 Fuzzy Rule

Langkah selanjutnya adalah membuat suatu aturan (*fuzzy rule*). *Fuzzy rule* ini nantinya digunakan untuk mendapatkan hasil *output* berupa waktu pengisian dan pengurasan air. Berikut adalah *Fuzzy rule* yang disusun oleh peneliti pada penelitian ini:

Tabel 3.1 Fuzzy Rule Penambahan

Kekeruhan / Level	Kekeruhan (Jernih)	Kekeruhan (Normal)	Kekeruhan (Keruh)
Level (Rendah)	Lama (R1)	Lama (R4)	Lama (R7)
Level (Normal)	Normal (R2)	Normal (R5)	Sebentar (R8)
Level (Tinggi)	Sebentar (R3)	Sebentar (R6)	Sebentar (R9)

Tabel 3.2 Fuzzy Rule Pengurangan

Kekeruhan / Level	Kekeruhan (Jernih)	Kekeruhan (Normal)	Kekeruhan (Keruh)
Level (Rendah)	Sebentar (R1)	Sebentar (R4)	Sebentar (R7)
Level (Normal)	Sebentar (R2)	Sebentar (R5)	Normal (R8)
Level (Tinggi)	Sebentar (R3)	Lama (R6)	Lama (R9)

Apabila diterapkan berdasarkan *rule base* yang telah kita buat, maka untuk *rule* pengisian air adalah sebagai berikut :

- (R1) *IF* kekeruhan *is* Jernih *AND* level *is* Rendah *THEN* waktu Lama
- (R2) *IF* kekeruhan *is* Jernih *AND* level *is* Normal *THEN* waktu Normal
- (R3) *IF* kekeruhan *is* Jernih *AND* level *is* Tinggi *THEN* waktu Sebentar
- (R4) *IF* kekeruhan *is* Normal *AND* level *is* Rendah *THEN* waktu Lama
- (R5) *IF* kekeruhan *is* Normal *AND* level *is* Normal *THEN* waktu Normal
- (R6) *IF* kekeruhan *is* Normal *AND* level *is* Tinggi *THEN* waktu Sebentar
- (R7) *IF* kekeruhan *is* Keruh *AND* level *is* Rendah *THEN* waktu Lama
- (R8) *IF* kekeruhan *is* Keruh *AND* level *is* Normal *THEN* waktu Sebentar
- (R9) *IF* kekeruhan *is* Keruh *AND* level *is* Tinggi *THEN* waktu Sebentar

Dan untuk *rule* pengurangan air adalah sebagai berikut :

- (R1) *IF* kekeruhan *is* Jernih *AND* level *is* Rendah *THEN* waktu Sebentar
- (R2) *IF* kekeruhan *is* Jernih *AND* level *is* Normal *THEN* waktu Sebentar
- (R3) *IF* kekeruhan *is* Jernih *AND* level *is* Tinggi *THEN* waktu Sebentar
- (R4) *IF* kekeruhan *is* Normal *AND* level *is* Rendah *THEN* waktu Sebentar
- (R5) *IF* kekeruhan *is* Normal *AND* level *is* Normal *THEN* waktu Sebentar
- (R6) *IF* kekeruhan *is* Normal *AND* level *is* Tinggi *THEN* waktu Lama

(R7) *IF* kekeruhan *is* Keruh *AND* level *is* Rendah *THEN* waktu Sebentar

(R8) *IF* kekeruhan *is* Keruh *AND* level *is* Normal *THEN* waktu Normal

(R9) *IF* kekeruhan *is* Keruh *AND* level *is* Tinggi *THEN* waktu Lama

Selanjutnya adalah menghitung hasil *fuzzifikasi* dengan fungsi implikasi untuk diambil nilai terkecilnya dengan menggunakan fungsi *MIN* dengan *AND* sebagai operatornya untuk menentukan nilai keanggotaan terkecil, adapun ilustrasi dari persamaan fungsi *MIN* dengan operator *AND* dapat dilihat pada rumus di bawah ini.

$$\mu_{A \cap B} = \min(\mu_A[x], \mu_B[y])$$

3.3.3 Defuzzifikasi

Dalam proses ini akan terjadi proses pengubahan dari nilai *fuzzy* menjadi nilai *crisp* kembali seperti halnya nilai *input* di awal. Pada penelitian ini metode *defuzzifikasi* yang digunakan oleh peneliti adalah metode *Center of Area (CoA)*. Metode *Center of Area* dilakukan dengan menemukan nilai titik tengah pada fungsi keanggotaan waktu, yang memiliki fungsi keanggotaan sebentar, normal dan lama. Perhitungan *Center of Area* pada fungsi keanggotaan sebentar dilakukan dengan penghitungan pencarian luas segitiga yaitu melakukan perkalian pada panjang alas dan tinggi lalu dibagi dengan dua. Kemudian nilai tengah diperoleh dari hasil perhitungan nilai luas yang dijumlahkan dengan *range* minimum yang ada di fungsi keanggotaan sebentar. Perhitungan pada keanggotaan sebentar adalah sebagai berikut.

$$\text{Panjang} = 3 - 0 = 3$$

$$\text{Luas} = \frac{3 \times 1}{2}$$

$$= 1.5$$

$$\text{CoA} = 0 + 1.5 = 1.5$$

Pada fungsi keanggotaan normal, cara yang digunakan untuk mencari *Center of Area* juga sama seperti perhitungan pada fungsi keanggotaan sebarang, maka perhitungan pada keanggotaan normal adalah sebagai berikut.

$$\text{Panjang} = 2 - 0 = 2$$

$$\text{Luas} = \frac{2 \times 1}{2}$$

$$= 1$$

$$\text{CoA} = 2 + 1 = 3$$

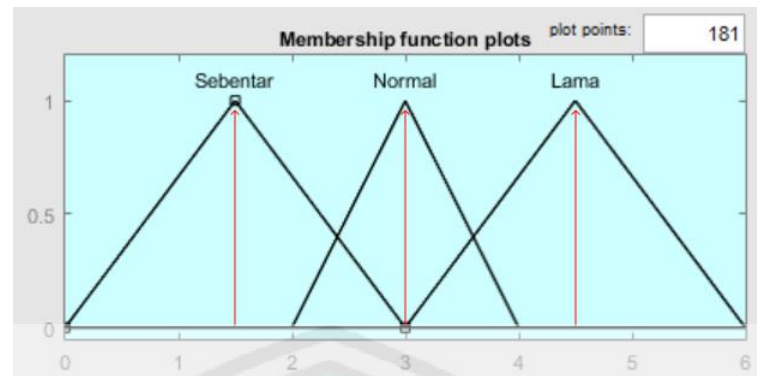
Pada fungsi keanggotaan lama, cara yang digunakan untuk mencari *Center of Area* juga sama seperti perhitungan pada fungsi keanggotaan sebarang dan normal, maka perhitungan pada keanggotaan lama adalah sebagai berikut.

$$\text{Panjang} = 3 - 0 = 3$$

$$\text{Luas} = \frac{3 \times 1}{2}$$

$$= 1.5$$

$$\text{CoA} = 3 + 1.5 = 4.5$$



Gambar 3.8 Nilai Tengah Fungsi Keanggotaan Waktu

Tabel 3.3 Data Luas dan Titik Tengah

Anggota	Luas	Center of Area (CoA)
sebentar	1.5	1.5
normal	1	3
lama	1.5	4.5

Setelah Data Luas dan Titik Tengah diperoleh seperti pada Tabel 3.3, langkah selanjutnya adalah melakukan perhitungan proses *defuzzifikasi* dengan menggunakan metode *Center of Area*. Di bawah ini adalah persamaan untuk menghitung *Center of Area*.

$$Output = \frac{\sum_{i=1}^n rule(i) \times CoA(i) \times luas(i)}{\sum_{i=1}^n rule(i) \times luas(i)}$$

Keterangan :

rule = fungsi implikasi *fuzzifikasi*

CoA = titik tengah fungsi keanggotaan waktu

luas = luas daerah keanggotaan waktu

n = jumlah *rule* yang digunakan

i = nilai perulangan data ke-*n*

BAB IV

PENGUJIAN DAN PEMBAHASAN

4.1 Hasil Pengujian

Pada penelitian ini, peneliti melakukan uji coba selama 7 hari berturut-turut dengan waktu pengontrolan setiap 3 jam sekali. Dengan menggunakan *prototype* kandang dengan dimensi 60x40x40 yang di isi dengan 4 ekor burung puyuh. Dan berikut adalah data uji coba yang penulis dapatkan dengan total sebanyak 57 data.

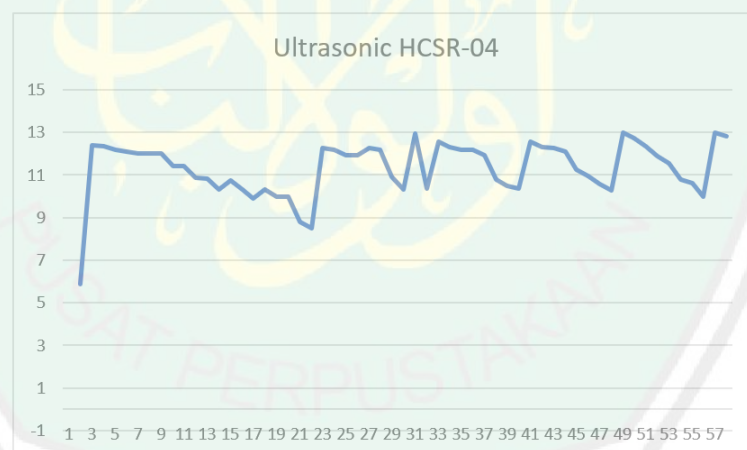
Tabel 4. 1 Data Uji Coba Sistem Kontrol Air Minum Kandang Burung Puyuh

No,	Waktu	Ultrasonic HCSR-04	Turbidity Sensor	Kondisi
				Level / Keruh
1	07:57:27--5:10:2020	5,88	7,5	Rendah / Jernih
2	11:01:24--5:10:2020	12,41	7,38	Tinggi / Jernih
3	14:01:33--5:10:2020	12,34	7,34	Tinggi / Jernih
4	17:05:24--5:10:2020	12,18	7,26	Tinggi / Jernih
5	20:05:33--5:10:2020	12,09	7,23	Tinggi / Jernih
6	23:05:42--5:10:2020	12,01	7,16	Tinggi / Jernih
7	02:05:51--6:10:2020	12,01	7,16	Tinggi / Jernih
8	05:05:59--6:10:2020	12	7,17	Tinggi / Jernih
9	08:06:09--6:10:2020	11,41	7,23	Tinggi / Jernih
10	11:06:18--6:10:2020	11,41	7,23	Tinggi / Jernih
11	14:06:27--6:10:2020	10,85	7,4	Tinggi / Jernih
12	17:06:36--6:10:2020	10,83	7,41	Tinggi / Jernih
13	20:10:11--6:10:2020	10,32	7,43	Normal / Jernih
14	23:12:06--6:10:2020	10,75	7,38	Tinggi / Jernih
15	02:12:16--7:10:2020	10,34	7,44	Normal / Jernih
16	05:17:40--7:10:2020	9,91	6,82	Normal / Jernih
17	08:23:22--7:10:2020	10,32	7,43	Normal / Jernih
18	11:29:03--7:10:2020	10	7,46	Normal / Jernih

19	14:34:53--7:10:2020	9,98	7,46	Normal / Jernih
20	17:41:32--7:10:2020	8,81	8,62	Normal / Jernih
21	20:48:52--7:10:2020	8,5	8,55	Normal / Jernih
22	23:56:07--8:10:2020	12,28	8,03	Tinggi / jernih
23	02:56:11--8:10:2020	12,19	7,98	Tinggi / jernih
24	05:56:19--8:10:2020	11,94	7,15	Tinggi / jernih
25	08:56:29--8:10:2020	11,94	7,15	Tinggi / jernih
26	11:56:37--8:10:2020	12,28	7,06	Tinggi / Jernih
27	15:56:50--8:10:2020	12,19	6,98	Tinggi / Jernih
28	18:56:59--8:10:2020	10,93	7,49	Tinggi / Jernih
29	21:57:02--8:10:2020	10,3	7,42	Normal / Jernih
30	23:57:07--8:10:2020	12,96	7,71	Tinggi / Jernih
31	02:57:23--9:10:2020	10,36	7,43	Normal / Jernih
32	05:59:08--9:10:2020	12,55	7,32	Tinggi / Jernih
33	08:59:45--9:10:2020	12,3	7,28	Tinggi / Jernih
34	11:59:37-9:10:2020	12,2	6,99	Tinggi / Jernih
35	15:59:50--9:10:2020	12,19	7	Tinggi / Jernih
36	18:59:55--9:10:2020	11,93	7,27	Tinggi / Jernih
37	21:59:58--9:10:2020	10,78	9,63	Tinggi / Jernih
38	23:59:07--9:10:2020	10,48	9,42	Tinggi / Jernih
39	02:59:23--10:10:2020	10,36	9,17	Normal / Jernih
40	06:01:08--10:10:2020	12,55	6,82	Tinggi / Jernih
41	09:01:15--10:10:2020	12,3	6,93	Tinggi / Jernih
42	12:01:37--10:10:2020	12,28	7,94	Tinggi / Jernih
43	15:56:50--10:10:2020	12,1	8,12	Tinggi / Jernih
44	18:56:59--10:10:2020	11,23	8,54	Tinggi / Jernih
45	21:57:02--10:10:2020	10,95	8,62	Tinggi / Jernih
46	23:57:07--10:10:2020	10,56	7,8	Tinggi / Jernih
47	02:57:23--11:10:2020	10,28	7,23	Tinggi / Jernih
48	05:59:08--11:10:2020	13	6,82	Tinggi / Jernih

49	08:03:45--11:10:2020	12,75	6,93	Tinggi / Jernih
50	11:56:37--11:10:2020	12,35	7,92	Tinggi / Jernih
51	15:56:50--11:10:2020	11,88	6,98	Tinggi / Jernih
52	18:56:59--11:10:2020	11,53	7,19	Tinggi / Jernih
53	21:57:02--11:10:2020	10,8	7,27	Tinggi / Jernih
54	23:57:07--11:10:2020	10,62	7,81	Tinggi / Jernih
55	02:57:23--12:10:2020	10	7,25	Tinggi / Jernih
56	05:59:08--12:10:2020	13	6,88	Tinggi / Jernih
57	08:03:45--12:10:2020	12,83	6,94	Tinggi / Jernih

Dari table 4.1 di atas, kondisi air minum pada kandang burung puyuh yang terjadi pada 7 hari tersebut dapat terlihat. Dan jika dipresentasikan dalam bentuk grafik, untuk kondisi level air yang didapat dari sensor *Ultrasonic HC SR04* adalah seperti gambar di bawah ini.

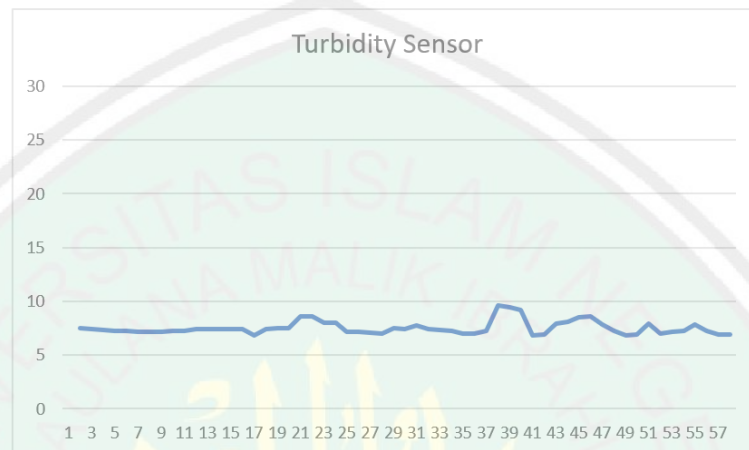


Gambar 4.1 Grafik Kondisi Level Air

Dari gambar 4.1 di atas, terlihat bahwa kondisi level air masih terbilang baik jika mengacu pada fungsi keanggotaan level *fuzzy* yang telah dibuat, air pada wadah penampungan air minum memiliki rata-rata level sebesar 11,32 cm dengan kondisi

level paling rendah yaitu 5,88 cm dan 8,5 cm yang masih tergolong dalam fungsi keanggotaan normal dan tinggi.

Kemudian untuk kondisi kekeruhan air yang didapat dari sensor *Turbidity* *SKU SEN0189* adalah seperti gambar di bawah ini.



Gambar 4.2 Grafik Kondisi Kekeruhan Air

Dan dari gambar 4.2 di atas, terlihat bahwa kondisi juga masih terbilang baik jika mengacu pada fungsi keanggotaan kekeruhan *fuzzy* yang telah dibuat, air pada wadah penampungan air minum memiliki rata-rata kekeruhan 7.52 NTU dengan kondisi kekeruhan paling tinggi yaitu 9,63 masih tergolong dalam fungsi keanggotaan jernih dan normal.

Setelah didapatkan data uji coba, peneliti melakukan pengujian lanjutan yaitu pengujian pada alat dan sistem. Pengujian alat dilakukan untuk mengetahui seberapa baik kinerja peralatan, sedangkan pengujian sistem dilakukan untuk mengukur seberapa baik kinerja metode pada sistem kontrol air minum kandang burung puyuh. Selanjutnya hasil pengujian disusun dalam bentuk tabel yang kemudian akan dilakukan perbandingan agar didapatkan nilai *error*.

4.1.1 Pengujian Alat

Berikut adalah pengujian alat yang dilakukan dalam penelitian ini.

4.1.1.1 Pengujian Sensor Level (*Ultrasonic HCSR-04*)

Hasil pembacaan sensor yang dilakukan oleh sensor *Ultrasonic HCSR-04* dibandingkan dengan alat ukur manual (penggaris). Dengan satuan berupa centimeter (cm) yang digunakan pada *Ultrasonic HCSR-04* dan alat ukur manual/penggaris. Berikut adalah hasil pengujian yang diperoleh.

Tabel 4. 2 Pengujian Sensor Level (*Ultrasonic HCSR-04*) dengan penggaris

No.	Sensor <i>Ultrasonic HCSR-04</i> (cm)	Penggaris (cm)	Error (%)
1.	2.27	2	1.35
2.	3.11	3	3.67
3.	4.25	4	6.25
4.	6.19	6	3.17
5.	7.67	7	9.57
6.	9.71	9	7.89
7.	10.56	10	5.60
8.	12.26	12	2.17
9.	13.14	13	1.07
10	14.09	14	6.42
Rata-rata Error (%)			4.71

Kemudian akan dilakukan perhitungan untuk mendapat presentase nilai *error* dan nilai rata-rata *error*. Pada perhitungan ini, peneliti akan menggunakan sebuah *sample* dari pengujian, yaitu pengujian ke-1 dengan *input* data *Ultrasonic HCSR-04* yang memiliki nilai 2,27 cm dan penggaris memiliki nilai 2 cm.

$$\begin{aligned}
 \text{Error} &= \frac{\text{Ultrasonic HCSR04} - \text{Penggaris}}{\text{Penggaris}} \times 100\% \\
 &= \frac{2.27 - 2.00}{2.00} \times 100\% \\
 &= \frac{0.27}{2.00} \times 100\% \\
 &= 1.35\%
 \end{aligned}$$

Dan untuk perhitungan rata-rata *error* dari data level di atas adalah seperti di bawah ini.

$$\begin{aligned}
 \text{Rata-rata Error} &= \frac{\sum \text{error}}{\sum \text{uji coba}} \\
 &= \frac{47.1}{10} \\
 &= 4.71
 \end{aligned}$$

Setelah dilakukan penghitungan rata-rata *error* dari data level oleh sensor *Ultrasonic HCSR-04* dan penggaris, peneliti mendapat nilai rata-rata nilai *error* sebesar 4,71%.

4.1.1.2 Pengujian Sensor Kekeruhan (*Turbidity Sensor SKU SEN0189*)

Hasil pembacaan sensor yang dilakukan oleh sensor kekeruhan (*Turbidity Sensor SKU SEN0189*) dibandingkan dengan *sample* air dengan beberapa tingkat kekeruhan. Dengan satuan nilai NTU yang digunakan pada *Turbidity Sensor SKU SEN0189* dan *sample* air. Berikut merupakan hasil pengujian yang diperoleh.

Tabel 4. 3 Pengujian Sensor Kekeruhan (*turbidity sensor*) dengan *sample* air

No.	<i>Turbidity Sensor</i> (NTU)	<i>Sample Air</i> (NTU)	<i>Error (%)</i>
1.	7.5	7	7.14
2.	10.4	10	4.00
3.	14.2	14	1.42
4.	3.3	3	10.00
5.	18.7	18.5	1.08
6.	6.4	6	6.66
7.	22.6	22	2.72
8.	12.3	12	2.50
9.	8.1	8	1.25
10.	24.7	24	2.91
Rata-rata <i>Error (%)</i>			3.96

Data yang terdapat pada tabel di atas merupakan data yang didapat dari sensor kekeruhan (*Turbidity Sensor SKU SEN0189*) dan *sample* air dengan beberapa tingkat kekeruhan. Kemudian dilakukan perhitungan pada data untuk memperoleh presentase nilai *error* dan nilai rata-rata *error*. Pada perhitungan ini, peneliti akan menggunakan sebuah *sample* dari pengujian yaitu pengujian ke-1 dengan *input* data sensor kekeruhan (*turbidity sensor*) yang memiliki nilai 7,5 NTU dan *sample* air memiliki nilai 7 NTU.

$$\text{Error} = \frac{\text{Turbidity Sensor} - \text{Sample Air}}{\text{Sample Air}} \times 100\%$$

$$= \frac{7.5 - 7}{7} \times 100\%$$

$$= \frac{0.5}{7} \times 100\%$$

$$= 7.14\%$$

Dan untuk perhitungan rata-rata nilai *error* dari data kekeruhan di atas adalah seperti di bawah ini.

$$\begin{aligned} \text{Rata-rata Error} &= \frac{\sum \text{error}}{\sum \text{uji coba}} \\ &= \frac{39.68}{10} \\ &= 3.96 \end{aligned}$$

Setelah dilakukan penghitungan rata-rata *error* dari data kekeruhan oleh Turbidity Sensor SKU SEN0189 dan sample air, peneliti mendapat nilai rata-rata *error* sebesar 3,96 %.

4.1.2 Pengujian Sistem Mikrokontroler dengan Matlab

Dari *input* level serta kekeruhan pada pengujian yang telah dilakukan maka diperoleh nilai *output* yaitu waktu dalam satuan menit. Selanjutnya dilakukan pengujian dengan membandingkan antara perhitungan pada *mikrokontroler* dan perhitungan pada program matlab. Berikut adalah hasil pengujian antara perhitungan metode pada *mikrokontroler* dengan perhitungan Matlab.

Tabel 4. 4 Pengujian Mikrokontroler dengan Matlab

No.	Input		Output (menit)		Error (%)
	Level	Kekeruhan	Mikrokontroler	Matlab	
1.	7.67	7.50	2.92	2.85	2.09
2.	12.19	8.12	1.43	1.50	4.66
3.	10.32	7.43	1.92	1.76	9.09
4.	10.75	7.38	1.57	1.60	1.87

5.	8.50	8.55	2.78	2.38	16.80
6.	11.41	7.23	1.54	1.50	2.66
7.	5.88	7.50	3.79	3.83	1.04
8.	8.81	8.62	2.54	2.28	11.40
9.	9.91	6.82	2.06	1.90	8.42
10.	9.98	7.46	1.95	1.88	3.72
Rata-rata <i>Error</i> (%)					6.17

Data kemudian akan dihitung guna memperoleh presentase nilai *error* serta nilai rata-rata *error*. Pada perhitungan ini, peneliti akan menggunakan salah satu percobaan yang telah didapat, yaitu pada percobaan nomor 7 di mana memiliki *input* data sensor level (*ultrasonic HCSR-04*) dengan nilai 5,88 cm dan sensor kekeruhan (*Turbidity Sensor*) memiliki nilai 7,5 NTU.

$$\begin{aligned}
 \text{Error} &= \frac{\text{Mikrokontroler} - \text{Matlab}}{\text{Matlab}} \times 100\% \\
 &= \frac{3.79 - 3.83}{3.83} \times 100\% \\
 &= \frac{0.04}{3.83} \times 100\% \\
 &= 1.04 \%
 \end{aligned}$$

Dan untuk perhitungan rata-rata *error* dari perhitungan *mikrokontroler* dan *Matlab* di atas adalah seperti di bawah ini.

$$\begin{aligned}
 \text{Rata-rata Error} &= \frac{\sum \text{error}}{\sum \text{uji coba}} \\
 &= \frac{61.75}{10} \\
 &= 6.17
 \end{aligned}$$

Setelah dilakukan penghitungan rata-rata *error* dari perhitungan pada *mikrokontroler* dan Matlab, peneliti memperoleh nilai rata-rata *error* yaitu 6.17 %.

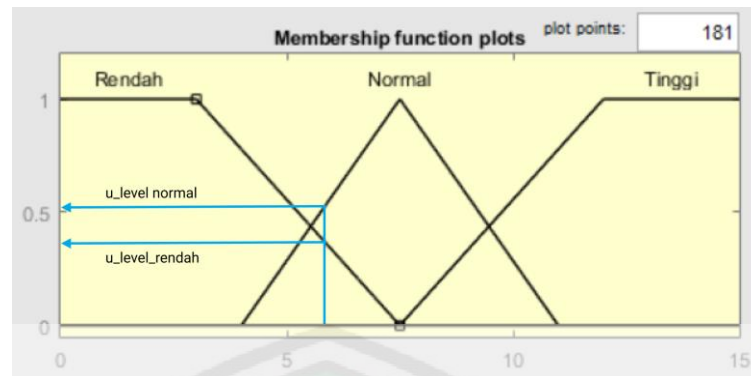
4.2 Pembahasan

4.2.1 Pembahasan Perhitungan *Fuzzy Logic*

4.2.1.1 Perhitungan pada Matlab

Peneliti melakukan perhitungan menggunakan *software* Matlab R2015a. Tahap pertama yang dilakukan peneliti adalah dengan membuka *software* Matlab, lalu mengetikkan “*fuzzy*” pada *command window* di Matlab untuk membuka *fuzzy logic designer*. Selanjutnya adalah mengatur *input*, *rule base* dan *output* sesuai dengan nilai variabel dan data yang telah kita tentukan sebelumnya.

Sebagai contoh, peneliti mengambil salah satu data *input* yang telah didapat. Yaitu nilai *input* level dengan nilai 5,88 cm dan *input* kekeruhan dengan nilai 7,5 NTU. Nilai keanggotaan level dapat dilihat pada gambar di bawah.



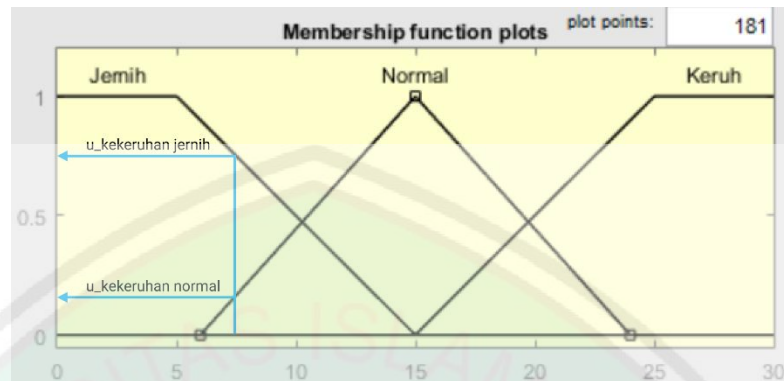
Gambar 4.3 Fungsi Keanggotaan Level

Dan untuk menghitung keanggotaan level normal dan level tinggi akan menggunakan perhitungan sebagai berikut.

$$\begin{aligned}
 u_{\text{Level rendah}} &= \frac{(d-x)}{(d-c)} \\
 &= \frac{(7.5-5.88)}{(7.5-3)} \\
 &= \frac{(1.62)}{(4.5)} \\
 &= 0.36 \\
 u_{\text{Level normal}} &= \frac{(x-a)}{(b-a)} \\
 &= \frac{(5.88-4)}{(7.5-4)} \\
 &= \frac{(1.88)}{(3.5)} \\
 &= 0.53
 \end{aligned}$$

Dari perhitungan di atas, didapat nilai keanggotaan level rendah sebesar 0.36 dan nilai level normal sebesar 0.53.

Selanjutnya untuk nilai keanggotaan kekeruhan dapat dilihat pada gambar di bawah.



Gambar 4.4 Fungsi Keanggotaan Kekeruhan

Dan untuk menghitung keanggotaan kekeruhan rendah dan kekeruhan normal akan menggunakan perhitungan sebagai berikut.

$$u_{\text{Kekeruhan rendah}} = \frac{(d-x)}{(d-c)}$$

$$= \frac{(15-7.5)}{(15-5)}$$

$$= \frac{(7.5)}{(10)}$$

$$= 0.75$$

$$u_{\text{Kekeruhan normal}} = \frac{(x-a)}{(b-a)}$$

$$= \frac{(7.5-6)}{(15-6)}$$

$$= \frac{(1.5)}{(9)}$$

$$= 0.16$$

Dari perhitungan di atas, didapat nilai keanggotaan kekeruhan rendah sebesar 0.75 dan nilai kekeruhan normal sebesar 0.16.

Tahap kedua adalah menyusun *fuzzy rule* yaitu dengan cara melakukan perbandingan pada nilai fuzzifikasi di setiap *input* dan mengimplementasikan fungsi MIN untuk mendapat nilai paling kecil dari tiap perbandingan.

Tabel 4.5 Perhitungan Aturan *Fuzzy*

Kekeruhan	Level	Aturan	Aturan Ke-
0	0	$0, 0 = 0.36$	Aturan 1
	1	$0, 1 = 0.53$	Aturan 2
	2	$0, 2 = 0$	Aturan 3
1	0	$1, 0 = 0.16$	Aturan 4
	1	$1, 1 = 0.16$	Aturan 5
	2	$1, 2 = 0$	Aturan 6
2	0	$2, 0 = 0$	Aturan 7
	1	$2, 1 = 0$	Aturan 8
	2	$2, 2 = 0$	Aturan 9

Tahap ketiga adalah proses perhitungan *defuzzifikasi*. Hasil perhitungan diperoleh dari proses perkalian pada setiap aturan dengan nilai luas dan *CoA* dan fungsi keanggotaan waktu. Di bawah ini adalah tabel *rule* luas serta titik tengah berlandaskan *rule* fungsi keanggotaan variabel waktu.

Tabel 4.6 Rule Luas dan CoA

Aturan	Luas	CoA
1	1.5	4.5
2	1	3
3	1.5	1.5
4	1.5	4.5
5	1	3
6	1.5	1.5
7	1.5	1.5
8	1.5	1.5
9	1.5	1.5

Tahap akhir yaitu mendefinisikan *defuzzifikasi* dengan menggunakan metode *Center of Area (CoA)* di bawah ini.

Defuzzifikasi (CoA)

$$= \frac{\sum_{i=1}^n \text{aturan}(i) \times \text{CoA}(i) \times L(i)}{\sum_{i=1}^n \text{aturan}(i) \times L(i)}$$

Defuzzifikasi (CoA)

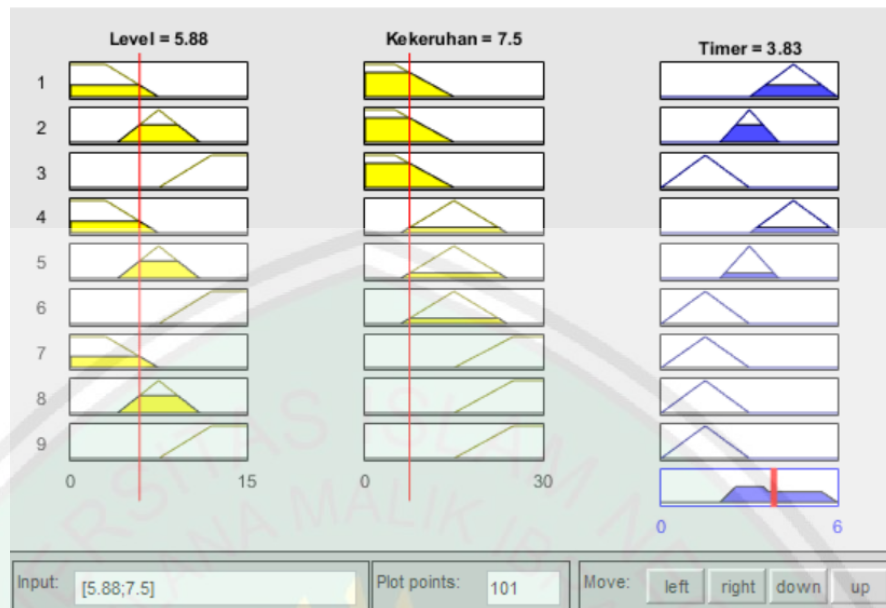
$$= \frac{\text{rule}[0][0] \times \text{CoA}(1) \times L(1) + \text{rule}[0][1] \times \text{CoA}(2) \times L(2) + \text{rule}[1][0] \times \text{CoA}(4) \times L(4) + \text{rule}[1][1] \times \text{CoA}(5) \times L(5)}{\text{rule}[0][0] \times L(1) + \text{rule}[0][1] \times L(2) + \text{rule}[1][0] \times L(4) + \text{rule}[1][1] \times L(5)}$$

$$= \frac{0.36 \times 4.5 \times 1.5 + 0.53 \times 3 \times 1 + 0.16 \times 4.5 \times 1.5 + 0.16 \times 3 \times 1}{0.36 \times 1.5 + 0.53 \times 1 + 0.16 \times 1.5 + 0.16 \times 1}$$

$$= \frac{5.58}{1.47}$$

$$= 3.79$$

Dan apabila dibuktikan dengan Matlab, maka akan terlihat seperti berikut.



Gambar 4. 5 Pembuktian Perhitungan Matlab

Pada contoh perhitungan dari hasil uji coba dengan *input* level yang bernilai 5.88 cm dan kekeruhan yang bernilai 7.5 NTU maka didapatkan *output* hasil perhitungan waktu sebesar 3,83 menit.

4.2.1.2 Perhitungan pada Mikrokontroler

Tahap awal yang dilakukan yakni menemukan nilai *fuzzifikasi* dari *input* level dan kekeruhan. Nilai *fuzzifikasi* pada setiap *input* didapat dengan cara digunakannya rumus *fuzzifikasi* yang terdapat pada program *Arduino*. Nilai *fuzzifikasi* yang diperoleh adalah seperti di bawah ini.

Pada *input* level

Level Rendah [0] = 0

Level Normal [1] = 0

Level Tinggi [2] = 0

Pada *input* kekeruhan

Kekeruhan Rendah [0] = 0

Kekeruhan Normal [1] = 0

Kekeruhan Tinggi [2] = 0

Nilai *fuzzifikasi* level dan kekeruhan di atas diperoleh dari proses hitung *fuzzifikasi* dengan *source code* di bawah ini.

Source code fuzzifikasi untuk level :

```
void FuzzyLevel() {
    if (level < 3) {
        u_level[0] = 1;
        u_level[1] = 0;
        u_level[2] = 0;
    }
    else if (level >= 3 and level <= 7.5) {
        u_level[0] = (7.5 - level) / 4.5;
        u_level[1] = (level - 4) / 3.5;
        u_level[2] = 0;
    }
    else if (level >= 7.5 and level <= 12) {
        u_level[0] = 0;
        u_level[1] = (11 - level) / 3.5;
        u_level[2] = (level - 7.5) / 4.5;
    }
    else {
        u_level[0] = 0;
        u_level[1] = 0;
        u_level[2] = 1;
    }
}
```

Di *source code fuzzifikasi* tersebut, dilakukan pembagian menjadi beberapa kondisi. Kondisi pertama yaitu apabila nilai *input* level kurang dari 3, maka didapat nilai *fuzzifikasi* adalah level rendah [0]. Kondisi kedua yaitu apabila nilai *input* level lebih dari atau sama dengan 3 dan kurang dari atau sama dengan 7.5, maka didapat nilai *fuzzifikasi* adalah pada level rendah [0] dilakukan pembagian dari hasil 7.5 dikurangi nilai *input* level dibagi dengan 4.5 (nilai pembagi 4.5 didapat dari pengurangan 7.5 – 3 pada kurva rendah), dan pada level normal [1] dilakukan pembagian dari hasil nilai *input* level dikurangi 4 dibagi dengan 3.5 (nilai pembagi

3.5 didapat dari $7.5 - 4$ pada kurva normal). Kemudian kondisi ketiga yaitu apabila nilai *input* level lebih dari atau sama dengan 7.5 dan kurang dari atau sama dengan 12, maka didapat nilai *fuzzifikasi* adalah pada level normal [1] dilakukan pembagian dari hasil 11 dikurangi nilai *input* level dibagi dengan 3.5, dan pada level tinggi [2] dilakukan pembagian dari hasil nilai *input* level dikurangi 7.5 dibagi dengan 4.5 (nilai pembagi 4.5 didapat dari 12 dikurangi 7.5). Dan kondisi terakhir yaitu apabila nilai *input* level tidak termasuk dalam kondisi pertama hingga ketiga, maka didapat nilai *fuzzifikasi* adalah level tinggi [2].

Source code *fuzzifikasi* kekeruhan :

```
void FuzzyKekeruhan() {
    if (kekeruhan < 5) {
        u_kekeruhan[0] = 1;
        u_kekeruhan[1] = 0;
        u_kekeruhan[2] = 0;
    }
    else if (kekeruhan >= 5 and kekeruhan <= 15) {
        u_kekeruhan[0] = (15 - kekeruhan) / 10;
        u_kekeruhan[1] = (kekeruhan - 6) / 9;
        u_kekeruhan[2] = 0;
    }
    else if (kekeruhan >= 15 and kekeruhan <= 25) {
        u_kekeruhan[0] = 0;
        u_kekeruhan[1] = (24 - kekeruhan) / 9;
        u_kekeruhan[2] = (kekeruhan - 15) / 10;
    }
    else {
        u_kekeruhan[0] = 0;
        u_kekeruhan[1] = 0;
        u_kekeruhan[2] = 1;
    }
}
```

Pada *source code fuzzifikasi* di atas dilakukan pembagian menjadi beberapa kondisi. Kondisi pertama yaitu apabila nilai *input* kekeruhan kurang dari 8, maka

didapat nilai *fuzzifikasi* adalah kekeruhan jernih [0]. Kondisi kedua yaitu apabila nilai *input* kekeruhan lebih dari atau sama dengan 8 dan kurang dari atau sama dengan 15, maka didapat nilai *fuzzifikasi* adalah pada kekeruhan jernih [0] dilakukan pembagian dari hasil 15 dikurangi nilai *input* kekeruhan dibagi dengan 10 (nilai pembagi 10 didapat dari pengurangan 15 – 5 pada kurva jernih), dan pada kekeruhan normal [1] dilakukan pembagian dari hasil nilai *input* kekeruhan dikurangi 10 dibagi dengan 9 (nilai pembagi 9 didapat dari 15 – 6 pada kurva normal). Kemudian kondisi ketiga yaitu apabila nilai *input* kekeruhan lebih dari atau sama dengan 15 dan kurang dari atau sama dengan 25, maka didapat nilai *fuzzifikasi* adalah pada kekeruhan normal [1] dilakukan pembagian dari hasil 24 dikurangi nilai *input* kekeruhan dibagi dengan 9, dan pada kekeruhan tinggi [2] dilakukan pembagian dari hasil nilai *input* kekeruhan dikurangi 15 dibagi dengan 10 (nilai pembagi 10 didapat dari 25 dikurangi 15). Dan kondisi terakhir yaitu apabila nilai *input* kekeruhan tidak termasuk dalam kondisi pertama hingga ketiga, maka didapat nilai *fuzzifikasi* adalah kekeruhan keruh [2].

Tahap selanjutnya adalah menyusun *fuzzy rule* yaitu dengan cara melakukan perbandingan pada nilai *fuzzifikasi* di setiap *input* dan mengimplementasikan fungsi MIN untuk mendapat nilai paling kecil dari tiap perbandingan. Di bawah ini adalah *source code arduino* guna memperoleh *rule*.

Source code fuzzy rule:

```
void Aturan () {
  aturan[0] = min(u_level[0], u_kekeruhan[0]);
  aturan[1] = min(u_level[1], u_kekeruhan[0]);
  aturan[2] = min(u_level[2], u_kekeruhan[0]);
  aturan[3] = min(u_level[0], u_kekeruhan[1]);
  aturan[4] = min(u_level[1], u_kekeruhan[1]);
```

```

aturan[5] = min(u_level[2], u_kekeruhan[1]);
aturan[6] = min(u_level[0], u_kekeruhan[2]);
aturan[7] = min(u_level[1], u_kekeruhan[2]);
aturan[8] = min(u_level[2], u_kekeruhan[2]);
}

```

Dari *source code* tadi, maka diperoleh *rule* yang memiliki hasil seperti yang ada di tabel 4.4 yang telah dicantumkan sebelumnya.

Dan pada tahap akhir, metode *Center of Area (CoA)* digunakan sebagai metode dalam melakukan perhitungan pada *defuzzifikasi*. Dan didapat dari proses perkalian pada masing-masing *rule* dengan luas dan titik tengah dari fungsi keanggotaan variabel dari waktu. Di bawah ini adalah *source code* untuk proses *defuzzifikasi*.

```

void defuzzifikasi() {
    int i = 0;
    int j = 0;
    float pembilang_a = 0;
    float pembilang_b = 0;
    float penyebut_a = 0;
    float penyebut_b = 0;
    while (i < 9) {
        pembilang_a += aturan[i] * rule_CoA_a[i] * rule_luas_a[i];
        penyebut_a += aturan[i] * rule_luas_a[i];
        i = i + 1;
    }
    while (j < 9) {
        pembilang_b += aturan[j] * rule_CoA_b[j] * rule_luas_b[j];
        penyebut_b += aturan[j] * rule_luas_b[j];
        j = j + 1;
    }
    timer_a = pembilang_a / penyebut_a;
    timer_b = pembilang_b / penyebut_b;
}

```

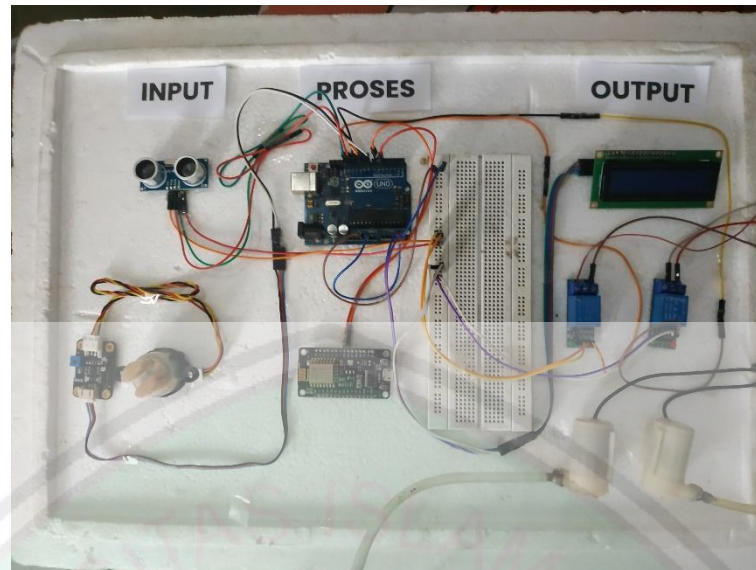
Dari *source code* di atas, didapatkan hasil *output* yaitu waktu yang berfungsi untuk menentukan waktu lamanya pengisian dan pengurasan air pada tempat penampungan air minum di kandang burung puyuh.

4.2.2 Sistem *Hardware*

Pada penelitian ini, sistem *hardware* yang digunakan menggunakan mikrokontroler *Arduino Uno* dan *Node MCU ESP8266*.

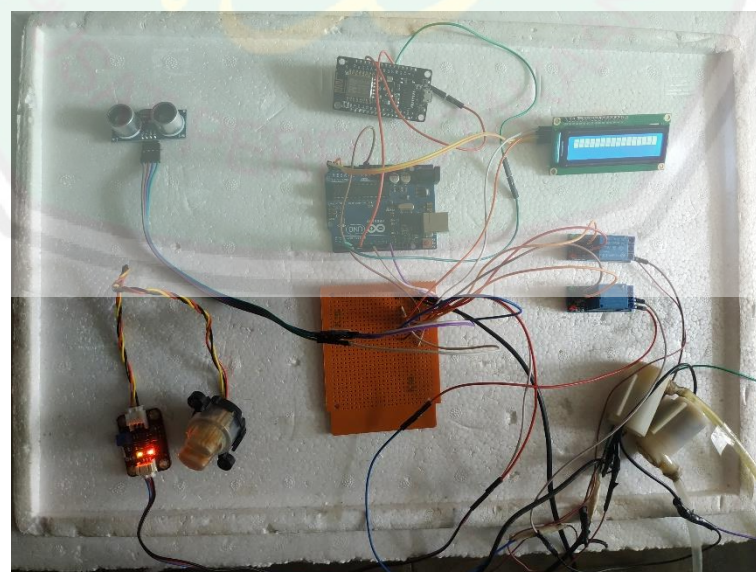
4.2.2.1 *Arduino Uno*

Arduino Uno adalah mikrokontroler yang digunakan sebagai pusat pengelolaan *input* yang akan diproses menjadi *output*. Pada penelitian ini, *input* yang diterima oleh *Arduino* berasal dari sensor *Ultrasonic HC SR04* dan sensor *Turbidity SKU SEN0189* yang selanjutnya akan dilakukan proses *fuzzy* dengan program yang telah ditanamkan pada *Arduino*. Sehingga akan menghasilkan *output* berupa waktu pengisian dan pengurasan air yang akan menjalankan pompa air. Pada rangkaian juga terdapat 2 buah modul *relay* yang berfungsi untuk menyalakan dan mematikan pompa pengisian dan pengurasan. Juga dilengkapi dengan LCD I2C yang berfungsi untuk menampilkan data waktu pengisian maupun pengurasan. Berikut ini adalah rangkaian pada *prototype* yang telah dibuat.



Gambar 4. 6 Rangkaian Pada Panel Arduino Menggunakan Breadbord

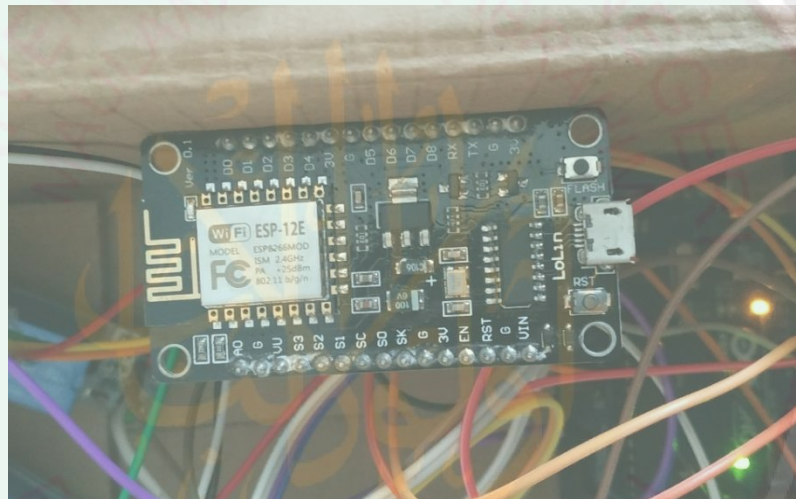
Seperti yang terlihat pada gambar diatas, pada penyusunan rangkaian pertama, peneliti hanya menggunakan *breadboard*. Dan terdapat berbagai masalah diantaranya adalah sering terjadinya ketidakstabilan/ kerusakan pada kabel *jumper*. Oleh karena itu, peneliti mencoba melakukan perbaikan yaitu dengan mengubah *breadboard* menjadi PCB titik. Berikut adalah rangkaian menggunakan PCB titik yang telah dibuat.



Gambar 4. 7 Rangkaian Pada Panel Arduino Menggunakan PCB Titik

4.2.2.2 Node MCU ESP8266

Di penelitian ini juga digunakan sebuah *Node MCU ESP8266*. Fungsi dari *Node MCU ESP8266* adalah sebagai alat untuk mengirim data yang diterima dari *Arduino* kepada *database* yang selanjutnya data pada *database* akan diakses dengan perangkat *smartphone*. Dan juga sebaliknya *Node MCU ESP8266* juga dapat mengirim perintah yang diberikan oleh perangkat *smartphone* kepada *Arduino*. Untuk dapat mengirim data ke *database*, *Node MCU ESP8266* ini akan terkoneksi dengan WIFI terdekat. Berikut adalah gambar dari *Node MCU ESP8266*.



Gambar 4. 8 *Node MCU ESP8266*

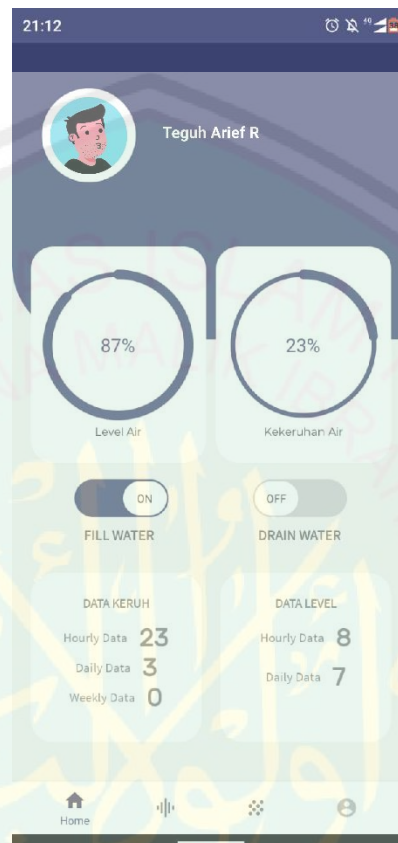
4.2.3 Sistem Interface

Pada penelitian ini, sistem *interface* yang digunakan adalah berbasis *android* yang akan di *install* langsung pada perangkat *smartphone* peternak nantinya.

4.2.3.1 Tampilan Home

Ini adalah tampilan utama ketika kita membuka aplikasi. Pada tampilan *home* ini, ditampilkan informasi-informasi dari sistem, seperti presentase tingkat level

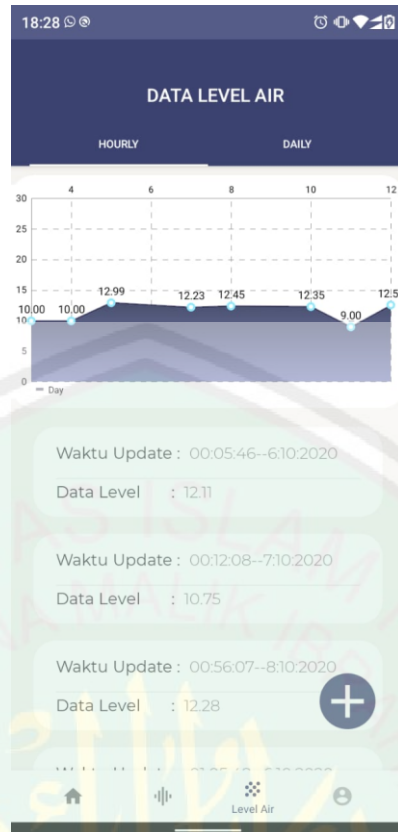
dan kekeruhan secara keseluruhan dan juga terdapat data jumlah data nilai level dan kekeruhan yang masuk tiap jam, hari dan minggunya. Di bawah ini adalah tampilan *home* di aplikasi yang telah dibuat.



Gambar 4. 9 Tampilan *Home*

4.2.3.2 Tampilan Data Level

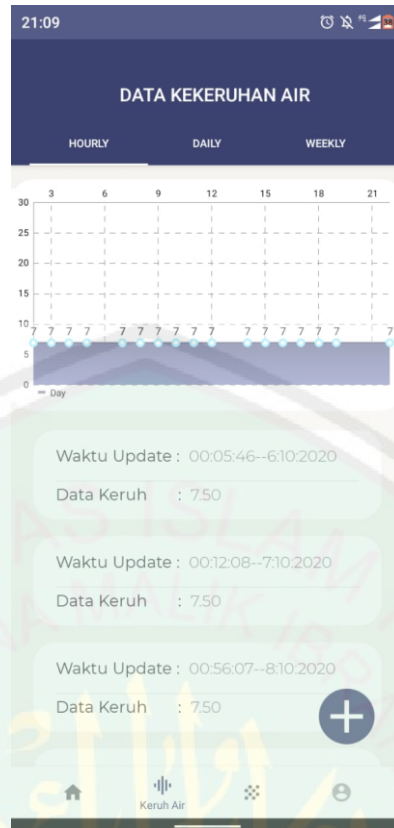
Pada tampilan data level, terdapat data hasil dari pembacaan sensor level yang ditampilkan dalam bentuk grafik dan non-grafik. Berikut adalah tampilan data level pada aplikasi yang dibuat.



Gambar 4. 10 Tampilan Data Level

4.2.3.3 Tampilan Data Kekeruhan

Sama seperti tampilan data level, pada halaman data kekeruhan ini juga terdapat data hasil dari pembacaan sensor kekeruhan yang ditampilkan dalam bentuk grafik dan non-grafik. Berikut adalah tampilan halaman data kekeruhan pada aplikasi yang dibuat.



Gambar 4. 11 Tampilan Data Level

4.2.4 Pengontrolan Air Minum Kandang Burung Puyuh

Pada penelitian ini peneliti menggunakan *prototype* kandang burung puyuh yang berukuran panjang 60 cm, lebar 40 cm dan tinggi 40 cm yang di isi 4 ekor burung puyuh. Kemudian alat pengontrolan air minum diletakkan di samping kandang dan alat akan melakukan pengontrolan setiap 3 jam sekali. Berikut merupakan penataan *prototype* kandang burung puyuh.



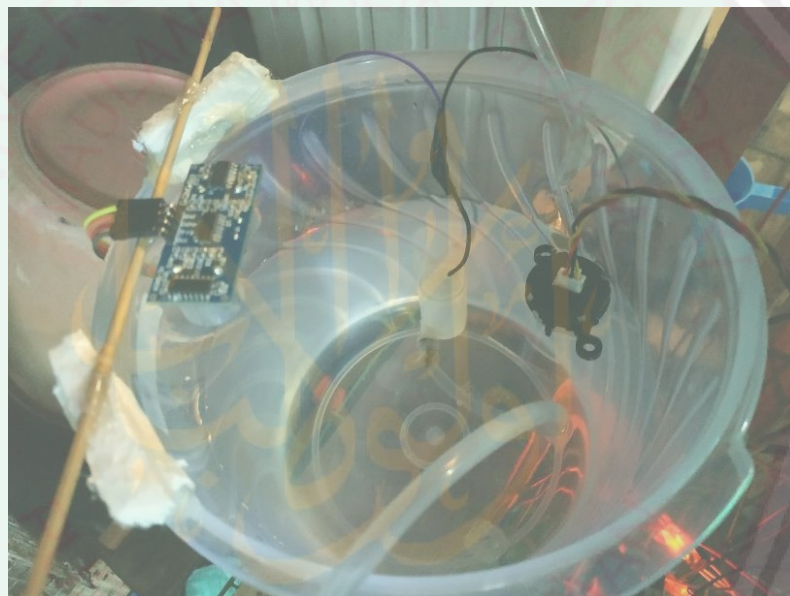
Gambar 4. 12 Penataan *Prototype* Kandang Burung Puyuh

Pemasangan sensor *Ultrasonic HC SR04* yaitu pada bagian atas wadah penampungan air minum menghadap ke bawah agar dapat mengukur ketinggian air yang diperlukan sebagai *input* level pada sistem. Pemasangan sensor *Ultrasonic HC SR04* ini dilakukan dengan sedemikian rupa agar tidak terkena air saat air penuh. Berikut adalah gambar penempatan sensor *Ultrasonic HC SR04* pada wadah penampungan air.



Gambar 4. 13 Penempatan Sensor sensor *Ultrasonic HC SR04*

Dan pemasangan sensor *Turbidity SKU SEN0189* yaitu pada permukaan air di wadah penampungan air. Karena sensor ini fungsinya untuk mengambil data kekeruhan air sebagai *input* kekeruhan pada sistem, maka sensor ini harus bersentuhan langsung dengan air sehingga dipasang dengan cara mengapungkannya di air, tujuannya adalah agar ujung sensor dapat menerima data kekeruhan dan agar kabel serta modul yang ada pada sensor *Turbidity SKU SEN0189* tidak terkena air. Berikut adalah gambar penempatan sensor *Turbidity SKU SEN0189* pada wadah penampungan air.



Gambar 4. 14 Penempatan Sensor *Turbidity SKU SEN0189*

Pengisian dan pengurasan air diproses menggunakan pompa air yang dialirkan dengan menggunakan selang kecil. Untuk pompa pengisian air diletakkan pada sumber air yang pada *prototype* ini adalah dengan menggunakan wadah yang lebih besar dari wadah penampungan air minum kandang puyuh. Kemudian air dialirkan ke wadah penampungan air minum kandang puyuh dengan menggunakan selang kecil. Sedangkan untuk pompa pengurasan air diletakkan pada dasar wadah

penampungan air minum kandang puyuh. Kemudian air dialirkan ke selokan atau wadah khusus pembuangan air dengan menggunakan selang kecil.

4.3 Integrasi Islam

Sebagaimana Allah menciptakan air salah satunya adalah untuk memberi minum untuk hewan ternak yang dalam penelitian ini adalah peternakan burung puyuh. Allah SWT berfirman dalam Qur'an surah Al Furqan (25) ayat 49 sebagai berikut:

لْنُحْيِيَ بِهِ بَلْدَةً مَّيِّتًا وَنُسْقِيَهُ مِمَّا خَلَقْنَا أَنْعَامًا وَأَنَاسِيَّ كَثِيرًا

Artinya : *“Agar Kami menghidupkan dengan air itu negeri (tanah) yang mati, dan agar Kami memberi minum dengan air itu sebagian besar dari makhluk Kami, binatang-binatang ternak dan manusia yang banyak”*.

Pada tafsir Jalalyn, ayat tersebut mempunyai tafsir seperti berikut: (Agar Kami menghidupkan dengan air itu negeri atau tanah yang mati) lafal Maitan dibaca Takhfif bentuk Mudzakkar dan Muannatsnya sama saja. Disebutkan dengan maksud, bahwa yang mati itu adalah tanah negeri itu (dan agar Kami memberi minum dengannya) dengan air itu (sebagian besar dari makhluk Kami, binatang-binatang ternak) unta, sapi dan kambing (dan manusia yang banyak) lafal Anaasiyyu merupakan bentuk jamak dari lafal Insaanun, bentuk asalnya adalah Anaasiinu, kemudian huruf Nun diganti menjadi Ya, lalu huruf Ya yang pertama diidghamkan kepadanya sehingga jadilah Anaasiyyu. Atau lafal Anaasiyyu ini adalah bentuk jamak dari lafal Insiyyun (Al-Mahalli & Jalaludin, 2018)

Dan terciptanya teknologi yang ada hingga saat ini tidak luput dari kekuatan yang Allah beri. Seperti yang difirmankan Allah dalam Quran surah Ar Rahman (55) ayat 33 yang berbunyi:

يَا مَعْشَرَ الْجِنِّ وَالْإِنسِ إِنِ اسْتَطَعْتُمْ أَنْ تَنْفُذُوا مِنْ أَقْطَارِ السَّمَاوَاتِ وَالْأَرْضِ فَانْفُذُوا
لَا تَنْفُذُونَ إِلَّا بِسُلْطَانٍ

Artinya : *"Hai jamaah jin dan manusia, jika kamu sanggup menembus (melintasi) penjuru langit dan bumi, maka lintasilah, kamu tidak dapat menembusnya kecuali dengan kekuatan(dari Allah)".*

Pada tafsir Jalalyn, ayat tersebut mempunyai tafsir seperti berikut: (Hai semua jin dan manusia, jika kalian sanggup menembus) melintasi (penjuru) atau kawasan-kawasan (langit dan bumi, maka lintasilah) perintah di sini mengandung makna yang menunjukkan ketidakmampuan mereka untuk melakukan hal tersebut (kalian tidak dapat menembusnya melainkan dengan kekuatan) dan kalian tidak akan mempunyai kekuatan untuk itu (Al-Mahalli & Jalaludin, 2018)

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan oleh penulis tentang rancang bangun sistem kontrol air minum kandang burung puyuh berbasis *Internet of Thing* menggunakan metode *Fuzzy Logic*, dapat disimpulkan bahwa sistem telah berhasil dibangun. Perancangan *hardware* dilakukan sedemikian rupa untuk membuat sistem ini. *Arduino UNO* digunakan sebagai *mikrokontroler* yang menerima *input* dari sensor *Ultrasonic HC SR-04* dan sensor *Turbidity SKU SEN0189* juga akan memproses *fuzzy* untuk menentukan nilai *output* berupa waktu yang akan menjalankan pompa air. Pada rangkaian ini juga dilengkapi dengan LCD untuk menampilkan informasi jalannya sistem. Pada sensor-sensor dilakukan pengujian untuk mengetahui presentasi *error* dan hasilnya sensor yang digunakan dapat berfungsi seperti yang diharapkan. Pada sensor *Ultrasonic HC SR-04* didapat rata-rata *error* sebesar 4.71% dan pada sensor *Turbidity SKU SEN0189* didapat rata-rata *error* sebesar 3.96%.

Kemudian untuk program yang ditanamkan pada *hardware* adalah program yang ditulis pada *software Arduino IDE* yang menggunakan bahasa C#. Program berisi perhitungan metode *Fuzzy Logic*, yaitu *fuzzifikasi*, *rule base* dan *defuzzifikasi*. Selanjutnya pada program perhitungan ini juga dilakukan pengujian untuk mengetahui presentase *error*. Yaitu dengan cara membandingkan nilai *output* yang dihasilkan oleh program dengan nilai yang dihasilkan oleh perhitungan Matlab. Dan didapatkan rata-rata *error* sebesar 6.17%. Sehingga pada penelitian ini peneliti mendapatkan presentase keberhasilan sistem sebesar 93,83%.

5.2 Saran

Pada hasil penelitian yang telah dilakukan, peneliti sadar jika penelitian ini masih ditemukan banyak kekurangan dan tentunya blm bisa dikatakan sempurna. Maka diharapkan agar penelitian ini bisa dikembangkan lagi sehingga menjadi semakin lebih baik. Agar harapan tersebut dapat tercapai, peneliti ingin menyampaikan beberapa saran sebagai berikut.

1. Pada penelitian yang telah dilakukan ini, masih tergolong sangat dasar karena hanya menggunakan 2 sensor sebagai parameter *input* dalam sistem kontrol air minum kandang puyuh ini, yaitu sensor *input* level dan sensor *input* kekeruhan. Mungkin untuk ke depannya dapat ditambahkan parameter *input* lainnya.
2. Perancangan komponen elektronik yang dilakukan pada penelitian ini masih sangat sederhana yaitu menggunakan *breadboard* yang pada praktiknya menyulitkan kestabilan percobaan dan kemudian kami perbaiki dengan menggunakan PCB titik yang hasilnya masih kurang rapi, diharapkan pada penelitian yang akan datang dapat menggunakan PCB cetak agar lebih stabil dan rapi.
3. Agar sistem lebih lengkap lagi fungsinya, mungkin ke depannya dapat ditambahkan fitur lain seperti pemberian makan otomatis, pembersih kotoran otomatis dan lain sebagainya.

DAFTAR PUSTAKA

- Al-Mahalli, J., & J. A.-S. (2018). *Tafsir Jalalain*. Ummul Quro.
- Achmad Fauzi. (2017). *SISTEM KONTROL SUHU RUANGAN PADA INKUBATOR ANAK AYAM MENGGUNAKAN ESP WEMOS DI BERBASIS IOT*. Universitas Narotama Surabaya. Juli 2017
- Arafat, M. K. (2016). *SISTEM PENGAMANAN PINTU RUMAH BERBASIS Internet Of Things (IoT) Dengan ESP8266*. Jurnal Ilmiah Fakultas Teknik "Technologia," 7(4), 262–268.
- Budiharto, W. dan Suhartono, D. (2014). *Artificial Intelligence*. Yogyakarta: Penerbit ANDI.
- Budiyanto. (2012). *Sistem Logger Suhu dengan Menggunakan Komunikasi Gelombang Radio*. Jurnal Teknologi Elektro, Universitas Mercu Buana. Vol.3 No1.
- Cholilulloh, M, Syauqi, D & Tibyani (2018). *Implementasi Metode Fuzzy Pada Kualitas Air Kolam Bibit Lele Berdasarkan Suhu dan Kekeruhan*. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN: 2548-964X Vol. 2, No. 5, Mei 2018, hlm. 1813-1822
- Fairuz.A.,Zubir.M., (2009). *Turbidimeter Design And Analysis: A Review On Optical Fiber Sensors For The Measurement Of Water Turbidity*, Sensors Vol 9, hal 8311-8335.
- Fakhri, Muh., Umar Hasan. (2017). *Sistem Pemantau Tagihan dan Kualitas Air PDAM Secara Nirkabel*. Fakultas Teknik, Universitas Hasanuddin.
- Hedlyni. (2011). *Pendekteksi Kekeruhan Air Berbasis Mikrokontroler AT89551 dengan Sensor Fotoresistor dan LED Inframerah*. Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Andalas.
- Hidyatama, A. A. (2013). *RANCANG BANGUN PROTOTIPE ELEVATOR MENGGUNAKAN MICROCONTROLLER ARDUINO ATMEGA 328P* . Jurnal Teknologi Elektro, Universitas Mercu.
- Ismail, Ikhsan. (2015). *Implementasi Logika Fuzzy dan Kalman Filter untuk Kendali Lengan Robot Menggunakan Gestur Tangan Manusia*. Tugas Akhir: Jurusan Teknik Elektro Telkom University, Bandung.
- Kusumadewi, S. dan Purnowo, H. (2004). *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu.

- Listiyowati, E. Dan Kinanti R., (2009). *Beternak Puyuh Secara Komersial*. Panebar Swadaya, Jakarta.
- M. Abdur, S. Habib, M. Ali, and S. Ullah. (2017). *Security Issues in the Internet of Things (IoT): A Comprehensive Study*. Int. J. Adv. Comput. Sci. Appl., vol. 8, no. 6, 2017.
- M. Cherian and H. K. P. (2014). *Implementation of a Secure and Smart Lab with Wireless Sensor Network*, vol. 3, no. 6, pp. 2012–2015, 2014.
- Mindriawan, Z, Arimbawa, W.A & Wijaya, G. P. (2018). *Implementasi Internet of Things Pada Sistem Monitoring Suhu dan Kontrol Air Pada Kandang Burung Puyuh Petelur dengan Menggunakan Protokol MQTT*. <http://eprints.unram.ac.id/id/eprint/11498>
- Muta'affif, M.F, dkk. (2017). *Sistem Kendali Peternakan Jarak Jauh Berbasis Internet of Things (IoT)*. PROSIDING SKF 2017.
- Nugrahanto, I. (2017). *Pembuatan water level sebagai pengendali water pump otomatis berbasis transistor*. Jurnal ilmu-ilmu teknik - sistem , vol. 13 no. 1
- Peraturan Menteri Kesehatan No. 416 Tahun 1990. Tentang : Syarat-syarat Dan Pengawasan Kualitas Air
- Peslinof. M. (2013). *Desain Alat Ukur Tingkat Kekeruhan Air Menggunakan Sistem Sensor Serat Optik*, Tesis Universitas Andalas, Padang.
- Qirom, Niam, B & Sungkar, M. S. (2019). *SISTEM MONITORING PENGAIRAN OTOMATIS DENGAN METODE LOGIKA FUZZY*. Jurnal Infotekmesin Vol.10, No.01, Januari 2019
- Saragih, B., (2000). *Agribisnis Berbasis Peternakan : Kumpulan Pemikiran*. IPB Bandung.
- Statistik Peternakan dan Kesehatan Hewan. (2018). *Populasi Puyuh Tahun 2014 - 2018*. Jakarta: Kementrian Pertanian Direktorat Jenderal Peternakan dan Kesehatan Hewan.
- Sudrajat. (2008). *Dasar Dasar Fuzzy Logic*. Bandung.
- Sugiharto, R.E. (2009). *Meningkatkan Keuntungan Beternak Puyuh*, Penerbit : Agromedia Pustaka.
- Suprianto, B. dan Arasada, B. (2017). *Aplikasi Sensor Ultrasonic Untuk Deteksi Posisi Jarak Pada Ruang Menggunakan Arduino Uno*. Jurnal Teknik Elektro. Volume 06 Nomor 02, 137 – 145.
- Sutojo, T., Mulyanto, E. dan Suhartono, V. (2011). *Kecerdasan Buatan*. Yogyakarta: Penerbit ANDI.

Tamzil, M. H. (2014). *Stres Panas Oada Unggas : metabolisme, akibat dan upaya penanggulangannya*, Wartazoa 24 (2) 57-66.

Tetty. (2014). *Puyuh Si Mungil Penuh Potensi*, Agro Media Pustaka, Jakarta.

Tulle, Christian Dendi Novian (2017). *MONITORING VOLUME CAIRAN DALAM TABUNG (DRUM SILINDER) DENGAN SENSOR ULTRASONIC BERBASIS WEB*. Diploma thesis, STMIK AKAKOM Yogyakarta.

Wijono, R. T. (2017). *Perancangan Sistem Monitoring dan Pengontrolan Pompa Air secara Wireless Berbasis Android*. Jurnal Tesla. Volume 19 Nomor 02.



LAMPIRAN

Lampiran 1.

DATA PENELITIAN

Tabel Data Pembacaan Sensor Kandang Puyuh tanggal 05-10-2020 s/d 12-10-2020 menggunakan *Ultrasonic HCSR-04* dan *Turbidity SKU SEN0189*.

No.	Waktu	<i>Ultrasonic HCSR-04</i>	<i>Turbidity SKU SEN0189</i>
1	07:57:27--5:10:2020	5,88	7,50
2	11:01:24--5:10:2020	12,41	7,38
3	14:01:33--5:10:2020	12,34	7,34
4	17:05:24--5:10:2020	12,18	7,26
5	20:05:33--5:10:2020	12,09	7,23
6	23:05:42--5:10:2020	12,01	7,16
7	02:05:51--6:10:2020	12,01	7,16
8	05:05:59--6:10:2020	12,00	7,17
9	08:06:09--6:10:2020	11,41	7,23
10	11:06:18--6:10:2020	11,41	7,23
11	14:06:27--6:10:2020	10,85	7,40
12	17:06:36--6:10:2020	10,83	7,41
13	20:10:11--6:10:2020	10,32	7,43
14	23:12:06--6:10:2020	10,75	7,38
15	02:12:16--7:10:2020	10,34	7,44
16	05:17:40--7:10:2020	9,91	6,82

17	08:23:22--7:10:2020	10,32	7,43
18	11:29:03--7:10:2020	10,00	7,46
19	14:34:53--7:10:2020	9,98	7,46
20	17:41:32--7:10:2020	8,81	8,62
21	20:48:52--7:10:2020	8,50	8,55
22	23:56:07--8:10:2020	12,28	8,03
23	02:56:11--8:10:2020	12,19	7,98
24	05:56:19--8:10:2020	11,94	7,15
25	08:56:29--8:10:2020	11,94	7,15
26	11:56:37--8:10:2020	12,28	7,06
27	15:56:50--8:10:2020	12,19	6,98
28	18:56:59--8:10:2020	10,93	7,49
29	21:57:02--8:10:2020	10,30	7,42
30	23:57:07--8:10:2020	12,96	7,71
31	02:57:23--9:10:2020	10,36	7,43
32	05:59:08--9:10:2020	12,55	7,32
33	08:59:45--9:10:2020	12,30	7,28
34	11:59:37--9:10:2020	12,20	6,99
35	15:59:50--9:10:2020	12,19	7,00
36	18:59:55--9:10:2020	11,93	7,27
37	21:59:58--9:10:2020	10,78	9,63
38	23:59:07--9:10:2020	10,48	9,42
39	02:59:23--10:10:2020	10,36	9,17

40	06:01:08--10:10:2020	12,55	6,82
41	09:01:15--10:10:2020	12,30	6,93
42	12:01:37--10:10:2020	12,28	7,94
43	15:56:50--10:10:2020	12,10	8,12
44	18:56:59--10:10:2020	11,23	8,54
45	21:57:02--10:10:2020	10,95	8,62
46	23:57:07--10:10:2020	10,56	7,80
47	02:57:23--11:10:2020	10,28	7,23
48	05:59:08--11:10:2020	13,00	6,82
49	08:03:45--11:10:2020	12,75	6,93
50	11:56:37--11:10:2020	12,35	7,92
51	15:56:50--11:10:2020	11,88	6,98
52	18:56:59--11:10:2020	11,53	7,19
53	21:57:02--11:10:2020	10,80	7,27
54	23:57:07--11:10:2020	10,62	7,81
55	02:57:23--12:10:2020	10,00	7,25
56	05:59:08--12:10:2020	13,00	6,88
57	08:03:45--12:10:2020	12,83	6,94

Lampiran 2.

VIDEO

Berikut adalah video proses pembuatan PCB yang dilakukan oleh peneliti.

<https://youtu.be/-P04TniLn0o>

