

RANCANG BANGUN MONITORING DAN  
PENGONTROLAN SUHU PADA KANDANG  
*CLOSED HOUSE* BERBASIS IoT



LAPORAN TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk menyelesaikan pendidikan Diploma Tiga  
(D-3) Program Studi Teknik Telekomunikasi  
Jurusan Teknik Elektro  
Politeknik Negeri Ujung Pandang

ANDI CAROLINE ESTER  
32220022

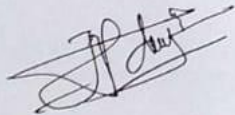
PROGRAM STUDI D-3 TEKNIK TELEKOMUNIKASI  
JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI UJUNG PANDANG  
MAKASSAR  
2023

## HALAMAN PENGESAHAN

Tugas Akhir dengan judul **Rancang Bangun Monitoring dan Pengontrolan Suhu pada Kandang Closed House Berbasis IoT** oleh Andi Caroline Ester Nim 32220022 dinyatakan layak untuk diujikan.

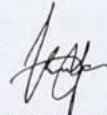
Makassar, 20 September 2023

Pembimbing I,



Lidemar Halide S.T., M.T  
Nip. 19700413 199602 1 001

Pembimbing II,



Rusdi Wartapane, S.T., M.Si  
Nip. 19651022 199103 1 002

Mengetahui,  
Koordinator Program Studi,




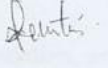
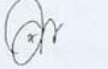
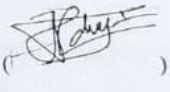

Yuniarli, S.ST., M.T  
Nip. 19770603 200212 2 002

## HALAMAN PENERIMAAN

Pada hari ini Jumat, 29 September 2023, tim penguji ujian sidang Laporan Tugas Akhir telah menerima hasil ujian sidang Laporan Tugas Akhir oleh mahasiswa Andi Caroline Ester dengan judul "Rancang Bangun Monitoring Dan Pengontrolan Suhu Pada Kandang *Closed house* Berbasis IoT"

Makassar, 29 September 2023

Tim Penguji Ujian Sidang Laporan Tugas Akhir:

- |                                |            |  |
|--------------------------------|------------|--|
| 1. Ir. Ichsan Mahjud, M.T      | Ketua      | (  )  |
| 2. Airin Dewi Utama, S.T., M.T | Sekretaris | (  ) |
| 3. Zaini, S.ST., M.T           | Anggota    | (  ) |
| 4. Lidemar Halide, S.T., M.T   | Anggota    | (  ) |
| 5. Rusdi Wartapane, S.T., M.Si | Anggota    | (  ) |

## KATA PENGANTAR

Dengan mengucapkan syukur kepada Tuhan Yang Maha Esa, atas limpahan kasih dan berkatnya , sehingga penulis dapat menyelesaikan Laporan Tugas Akhir ini dengan baik. Adapun judul dari Laporan Penulisan Tugas Akhir ini adalah **“Rancang Bangun Monitoring dan Pengontrolan Suhu pada kandang *Closed House* Berbasis IoT”**

Tujuan dari penulisan laporan Tugas Akhir ini adalah untuk melengkapi syarat kelulusan untuk Jurusan Teknik Elektro Prodi Teknik Telekomunikasi di Politeknik Negeri Ujung Pandang

Penulis menyadari bahwa selama menyusun Laporan Tugas Akhir ini mendapat banyak dukungan, bantuan dan doa dari berbagai pihak, oleh karena itu dalam kesempatan ini penulis mengucapkan terima kasih kepada:

1. Kedua orang tua yang selalu mendukung dan mendoakan, sehingga Tugas Akhir dapat diselesaikan dengan baik.
2. Bapak Prof. Ir. Ilyas Mansyur, M.T. selaku Direktur Politeknik Negeri Ujung Pandang.
3. Bapak Ahmad Rizal Sultan S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.
4. Ibu Yuniarti selaku Ketua Program Studi Teknik Telekomunikasi
5. Pembimbing I, Bapak Lidemar Halide, S.T., M.T
6. Pembimbing II, Bapak Rusdi Wartapane, S.T., M.Si
7. Seluruh Dosen Pengajar dan Staff Pegawai Program Studi Teknik Telekomunikasi Politeknik Negeri Ujung Pandang.
8. Teman-teman seperjuangan kelas 3A D3 Teknik Telekomunikasi.
9. Teman-teman Pengurus harian UKM Bahasa periode 2022-2023 untuk dorongan semangat dan perhatiannya selama penyusunan. #GraspTheWorldByLanguage.
10. Teman-teman dan kerabat keluarga yang selalu mendukung dan

mendoakan.

11. Adik saya, George yang senantiasa membantu saya dan menemani saya untuk melakukan survey ke kandang *closed house*.
12. Semua pihak yang telah membantu dan memberi dukungan kepada penulis dalam menyelesaikan Tugas Akhir.

Penulis mengharapkan kritik dan saran demi penyempurnaan penelitian ini. Penulis memohon maaf jika ada kesalahpahaman yang terdapat di tugas akhir ini dan tidak bermaksud untuk menyinggung pihak manapun. Pada akhir kata penulis mengucapkan terima kasih sebesar-besarnya.

Makassar, 27 September 2023

Penulis



## DAFTAR ISI

HALAMAN SAMPUL .....	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PENERIMAAN .....	iii
KATA PENGANTAR .....	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR .....	viii
DAFTAR TABEL.....	x
DAFTAR LAMPIRAN.....	xi
RINGKASAN .....	xi
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah .....	2
1.3 Ruang Lingkup .....	3
1.4 Tujuan.....	3
1.5 Manfaat.....	3
BAB II TINJAUAN PUSTAKA.....	5
2.1 Kandang <i>Closed House</i> .....	4
2.2 <i>Internet of Things</i> (IoT).....	6
2.3 NodeMCU ESP8266 .....	8
2.4 DHT11 .....	9
2.5 Pulse Width Modulation.....	11
2.6 Arduino IDE .....	13
2.7 MQTT.....	16

BAB III METODE PENELITIAN.....	17
3.1 Tempat dan Waktu Penelitian .....	17
3.2 Alat dan Bahan .....	17
3.3 Prosedur Penelitian.....	18
BAB IV HASIL DAN PEMBAHASAN .....	28
4.1 Hasil Perancangan Sistem .....	28
4.2 Pengujian.....	29
BAB V PENUTUP.....	32
5.1 Kesimpulan.....	32
5.2 Saran.....	32
DAFTAR PUSTAKA .....	33
LAMPIRAN.....	34



## DAFTAR GAMBAR

Gambar 2. 1 Kandang <i>Closed House</i> .....	6
Gambar 2. 2 ESP8266 .....	8
Gambar 2. 3 DHT11 .....	10
Gambar 2. 4 Modul PWM .....	12
Gambar 2. 5 Tampilan Awal Arduino IDE .....	14
Gambar 2. 6 Cara kerja <i>MQTT</i> .....	16
Gambar 3. 1 Diagram Blok .....	20
Gambar 3. 2 Skema Perancangan .....	23
Gambar 3. 3 Flowchart .....	24
Gambar 3. 4 Penulisan Program Arduino IDE .....	26
Gambar 3. 5 Program <i>MQTT</i> .....	27





## DAFTAR TABEL

Tabel 2. 1 Spesifikasi NodeMCU ESP8266 .....	9
Tabel 3. 1 Daftar Alat.....	17
Tabel 3. 2 Daftar Bahan .....	17
Tabel 3. 3 Hubungan antar pin Modul .....	23
Tabel 4.1 Penggunaan aplikasi mudah dipahami.....	30
Tabel 4. 2 Mengatasi permasalahan yang dimiliki oleh peternak.....	30
Tabel 4. 3 Kandang terkesan lebih canggih .....	30
Tabel 4. 4 Alat ini masih memiliki banyak kekurangan .....	31
Tabel 4. 5 Alat ini masih perlu pengembangan lebih lanjut.....	31
Tabel 4. 6 Menyebabkan korsleting pada aliran listrik .....	31



## DAFTAR LAMPIRAN

Lampiran 1 Keadaan di dalam kandang <i>Closed house</i> .....	66
Lampiran 2 Lokasi alat di dalam kadang .....	67
Lampiran 3 Hasil Perancangan Perangkat Keras .....	67
Lampiran 4 Pemograman Arduino IDE.....	68
Lampiran 5 <i>Shortcut MQTT</i> .....	68
Lampiran 6 Tampilan Login <i>MQTT</i> .....	69
Lampiran 7 Tampilan Dashboard <i>MQTT</i> .....	70
Lampiran 8 <i>Datasheet Hardware</i> .....	71
Lampiran 9 Kode Program.....	8



## **RANCANG BANGUN MONITORING DAN PENGONTROLAN SUHU PADA KANDANG CLOSED HOUSE**

### **RINGKASAN**

Suhu dan kelembapan di dalam kandang merupakan komponen kebutuhan yang sangat penting untuk membantu meningkatkan pertumbuhan Ayam broiler. Para peternak ayam broiler masih menggunakan cara manual dalam menjaga suhu optimal kandang. Rutinitas tersebut menimbulkan berbagai masalah yaitu seringkali peternak lupa atau lalai dalam menjaga suhu dan kelembapan pada kandang *closed house*.

Maka, dibuatlah Rancang Bangun Alat monitoring dan mengontrol suhu dan kelembapan berbasis *IoT* yang dapat dikendalikan melalui *smartphone*. Dalam perancangan ini komponen utama yang digunakan adalah ESP8266, DHT11, dan Modul PWM. Hasil dari perancangan ini adalah Alat yang dirancang akan memonitor dan mengontrol suhu dan kelembapan di kandang *closed house* melalui *smartphone*, lalu jika terjadi kenaikan suhu dari suhu optimal maka alat otomatis akan menyalakan blower.

Hasil dari rancang bangun ini adalah mampu mengontrol dan mempertahankan keseimbangan pada suhu 29-32°C dan kelembapan 60-80%, sudah sesuai dengan standar yang dibutuhkan ayam broiler, dapat disimpulkan bahwa Rancang Bangun Monitoring dan Pengontrolan suhu ini sangat membantu para peternak ayam broiler untuk menjaga suhu dan kelembapan kandang *closed house* agar tetap optimal untuk mencapai target keberhasilan bagi para peternak..

## BAB I PENDAHULUAN

### 1.1 Latar Belakang

Seiring dengan meningkatnya konsumsi ayam broiler di kalangan masyarakat akhir-akhir ini, maka peternak ayam broiler dituntut untuk meningkatkan produktivitas ayam serta dapat menghasilkan daging ayam dengan kualitas terbaik. Ayam merupakan unggas yang tergolong ke dalam hewan berdarah panas, artinya tidak memiliki kelenjar keringat dan seluruh tubuhnya dipenuhi oleh bulu. Suhu dan kelembapan di dalam kandang merupakan faktor yang sangat penting untuk mendukung pertumbuhan. Suhu tubuh yang dibutuhkan ayam pada masa umur 2 hari harus dijaga berada di suhu 32°C, umur 14 hari harus dijaga sekitar 30-32°C dan umur 35 hari dijaga pada suhu 28°C, ayam akan mati apabila suhu tubuhnya meningkat 4°C atau lebih. Sedangkan untuk kelembapan berkisar antara 60-80%.

Maka, kandang *closed house* merupakan salah satu komponen penting untuk dapat menciptakan kondisi lingkungan yang nyaman agar mampu menunjang pertumbuhan ayam agar lebih optimal. Di dalam kandang, seluruh kebutuhan ayam wajib tersedia, diantaranya suhu dan kelembapan yang tepat serta air minum dan makanan yang relatif berkualitas. Para peternak ayam broiler masih menggunakan cara manual dalam menjaga suhu dan kelembapan yang optimal. Rutinitas tersebut menyebabkan suatu masalah yaitu peternak terkadang lupa, terkendala jarak dan tidak konsisten dalam menjaga suhu dan kelembapan di dalam kandang.

Dari permasalahan tersebut maka dibutuhkan suatu alat monitoring dan pengontrolan suhu berbasis *IoT*. Beberapa modul akan dirancang menjadi suatu alat yang akan digunakan sebagai kontrol untuk memonitor suhu dan kelembapan pada kandang yang dapat juga ditenagakan dari jarak jauh. Kelebihan alat ini juga adalah dapat menyalakan blower atau pendingin udara di dalam kandang jika terjadi kenaikan suhu dari suhu yang dibutuhkan.

Dalam perancangan alat ini bertujuan untuk membantu peternak ayam broiler memudahkan aktifitas dan memecahkan masalah jika lupa atau lalai dalam melakukan pekerjaannya.. Dengan bantuan alat ini, peternak ayam broiler lebih mudah dalam memonitor dan mengontrol suhu dan kelembapan di kandang dengan menggunakan *smartphone* tanpa harus menyita waktu dan tenaga ataupun mengganggu aktivitas sehari-hari. Cara kerja alat ini nantinya peternak dapat langsung mengontrol dan memonitor suhu dan kelembapan melalui *smartphone* dengan memakai jaringan koneksi *wi-fi*.

Berdasarkan uraian permasalahan tersebut maka penulis merancang alat yang akan direalisasikan dalam tugas akhir berjudul “**Rancang bangun monitoring dan pengontrolan suhu pada kandang *closed house* berbasis *IoT***”. Diharapkan perancangan alat ini dapat membantu peternak ayam broiler dalam menjaga suhu dan kelembapan di dalam kandang agar tetap optimal dan lebih efisien.

## **1.2 Rumusan Masalah**

Berdasarkan uraian masalah dari latar belakang, maka dapat dirumuskan masalah dalam penulisan Tugas Akhir ini adalah apa saja yang perlu disiapkan

serta tahapan pengerjaan untuk merancang alat monitoring dan pengontrolan suhu dan kelembapan pada kandang closed house berbasis IoT sehingga dapat digunakan selanjutnya

### **1.3 Ruang Lingkup**

1. Menggunakan ESP8266 sebagai media pembuatan program untuk menjalankan sensor DHT11 dan modul PCM sebagai alat monitoring dan pengontrolan suhu dan kelembapan di dalam kandang
2. Menggunakan aplikasi MQTT sebagai media untuk mempermudah monitoring dan pengontrolan suhu di dalam kandang yang berbasis *IoT*
3. Penerapan alat monitoring dan pengontrolan suhu ini hanya diperuntukan bagi kandang *closed house* ayam broiler
4. Penerapan alat ini diperuntukan hanya untuk satu kandang *closed house* saja

### **1.4 Tujuan**

1. Merancang sistem pemantauan pengisian air galon berbasis *Internet of Things* (IoT) menggunakan protokol MQTT.
2. Menampilkan data hasil pengisian dan penjualan air galon pada usahadepot air minum isi ulang di website.

### **1.5 Manfaat**

Penulisan Tugas Akhir ini dapat memberikan beberapa manfaat untuk berbagai pihak yakni sebagai berikut:

#### 1.5.1 Manfaat Bagi Penulis:

- a. Penulis dapat mengaplikasikan teori-teori dan praktikum-praktikum yang diperoleh saat perkuliahan
- b. Menambah pengalaman penulis untuk lebih mendalami perakitan dan pembuatan sistem, khususnya pada Arduino.

#### 1.5.2 Manfaat bagi Akademik

- a. Sebagai tolak ukur penilaian bagi mahasiswa dalam menyerap ilmu perkuliahan dan mampu mengimplementasikannya.
- b. Menambah referensi studi pustaka sebagai bahan kajian, bahan acuan dan literatur pada Politeknik Negeri Ujung Pandang

#### 1.5.3 Manfaat bagi pengguna

- a. Dapat memonitoring dan mengontrol suhu lewat smartphone dan dapat dikendalikan dari jarak jauh.
- b. Dapat menjaga suhu dan kelembapan di dalam kandang secara otomatis.

#### 1.5.4 Manfaat bagi pembaca

- a. Dapat digunakan sebagai sumber informasi dan dapat menambah pengetahuan, rujukan, dan referensi
- b. Dapat juga dijadikan bahan acuan untuk melakukan penelitian atau studi lanjut.

## BAB II TINJAUAN PUSTAKA

### 2.1 *Closed House*

*Closed House* adalah kandang dengan sistem tertutup yang dijalankan pada peternakan yang bertujuan untuk menyediakan suhu dan kelembaban ideal bagi ayam, sehingga meminimalkan stres akibat perubahan kondisi lingkungan dan diharapkan mampu meningkatkan produktivitas ayam. Kandang *closed house* dapat meminimalkan kontak langsung ayam dengan organisme lain dan memiliki pengaturan ventilasi yang baik untuk menyediakan kondisi lingkungan yang nyaman bagi ternak (Wurlina, 2012).

Dengan kandang *Closed House* terjadi pergerakan udara yang stabil serta taraf kelembapan udara pada sangkar mampu diatur sinkron sesuai kebutuhan ayam. Semua dinding kandang tertutup. Kelebihan lain dari kandang tipe *Closed House* adalah kapasitas atau populasi jauh lebih banyak dari tipe kandang *Open House*, ayam lebih terjaga dari gangguan luar baik fisik, cuaca, maupun serangan penyakit, terhindar dari polusi, keseragaman ayam lebih bagus, dan pakan lebih efisien. Kandang tipe ini juga memberikan kemudahan karena kondisi angin akan lebih terkontrol.





*Gambar 2. 1 Kandang Closed House*

## **2.2 Internet of Things (IoT)**

Di zaman modern ini perkembangan teknologi sudah semakin canggih dengan perkembangan teknologi yang sudah ada. Terobosan baru mulai bermunculan, salah satunya adalah *Internet of Things* atau disingkat *IoT*.

*Internet of Things* merupakan teknologi yang menginovasi benda-benda sekitar dengan internet agar pekerjaan manusia sehari-hari menjadi lebih mudah dan efisien. Sebagai contoh, kita sedang berada di luar atau jauh dari rumah dan kita lupa untuk mematikan AC, Kompor, TV dan alat elektronik lainnya. Maka, kita bisa mengirim pesan/perintah melalui aplikasi pada smartphone -> diterima perangkat IoT -> Perangkat IoT memberikan perintah untuk mematikan secara otomatis alat elektronik -> alat elektronik mematikan daya.

Sederhananya, IoT bekerja dengan menggunakan instruksi pemograman yang setiap perintahnya dapat menghasilkan

interaksi ke sesama perangkat terhubung secara otomatis, bahkan dalam jarak jauh sekalipun. Adapun secara teknis, *IoT* memiliki beberapa komponen teknologi pendukung agar bekerja secara maksimal, yaitu

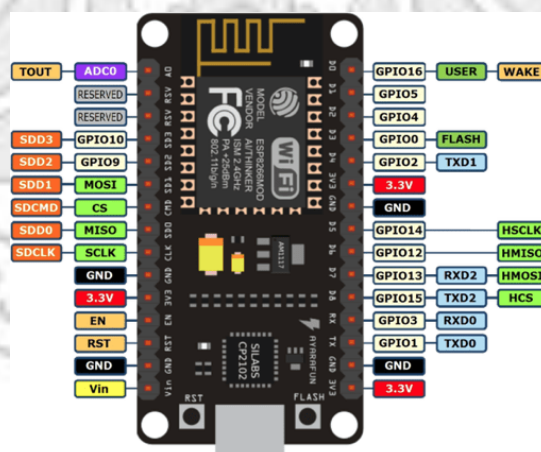
- Sensor = Sensor berfungsi untuk pedeksi dan memberikan data dari lingkungan sekitar. lokasigeografis, persediaan atau sesuatu yang kompleks.
- Koneksi Jaringan = Koneksi jaringan dibutuhkan agar bisa berkomunikasi secara lancar.
- *Artifical intelligence* = Sistem kecerdasan buatan atau AI dapat membuat perangkat atau mesin bekerja dan berpikir layaknya manusia.

### 2.3 NodeMCU ESP8266

ESP8266 merupakan sebuah mikrokontroller yang sering digunakan untuk perangkat *Internet of Things* . Mikrokontroller ini mempunyai fitur yang cukup lengkap dan mudah digunakan. Salah satu fitur yang paling menonjol adalah modul *Wi-Fi*. Artinya, modul ini sering digunakan untuk membuat sensor-sensor kecil, alat kontrol, atau perangkat IoT yang harus terhubung ke jaringan internet. Mikrokontroler ini bisa terhubung ke beberapa protokol komunikasi seperti TCP/IP, UDP, HTTP, dan MQTT.

Bentuk fisik dari ESP8266, terdapat port USB (mini USB) sehingga akan memudahkan dalam pemrogramannya. Secara fungsi modul ini hampir menyerupai dengan platform modul arduino, tetapi yang membedakan yaitu dikhususkan untuk “*Connected to Internet*”.

Terdapat beberapa pin I/O sehingga dapat dikembangkan menjadi sebuah aplikasi monitoring maupun controlling pada proyek IOT. NodeMCU ESP8266



Gambar 2. 2 ESP8266

Berikut ini adalah spesifikasi dari NodeMCU ESP8266:

Tabel 2.1 Spesifikasi NodeMCU ESP8266

Mikrokontoller	ESP8266
Input Tegangan	3.3V ~ 5V
Ukuran Board	57 mm x 30 mm
GPIO	13 pin
Flash Memory	4 MB
Wireless	802.11 b\g\n standart
USB to Serial Converter	CH340G

## 2.4 Sensor DHT11

DHT11 adalah sensor digital yang digunakan untuk mengukur suhu dan kelembapan udara di sekitarnya. Sensor ini mudah diaplikasikan dengan Arduino. DHT11 mempunyai banyak kelebihan, yaitu sensor kualitas tertinggi dalam hal respons, pembacaan data yang cepat, kemampuan anti-interferensi, sangat stabil dan memiliki kalibrasi yang sangat akurat. DHT11 memiliki 3 kaki pin, yaitu vcc, data, dan ground.

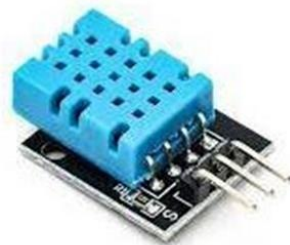
Di dalam tubuh sensor ini terdapat sebuah resistor dengan tipe PTC (Positif Temperature Coefficient) yang memiliki karakteristik dimana nilai resistansi berbanding terbalik dengan kenaikan suhu. Artinya, semakin tinggi suhu ruangan maka nilai resistansi akan semakin besar dan begitupun sebaliknya. Sensor DHT11 bekerja dengan mendeteksi suhu udara dan kelembaban menggunakan termistor dan sensor kelembaban kapasitif.

Berikut adalah Langkah-Langkah kerja sensor DHT11:

- **Penginderaan Suhu:** Sensor DHT11 menggunakan termistor untuk mengukur suhu udara. Termistor adalah sebuah resistor yang resistansinya berubah dengan perubahan suhu. Ketika suhu udara naik, resistansi termistor akan menurun, dan sebaliknya, ketika suhu udara turun,

resistansi termistor akan meningkat.

- **Penginderaan Kelembaban:** Sensor DHT11 menggunakan sensor kelembaban kapasitif untuk mengukur kelembaban udara. Sensor ini terdiri dari dua elektroda dengan substrat penahan kelembaban sebagai dielektrik di antara keduanya. Perubahan tingkat kelembaban akan menyebabkan perubahan nilai kapasitansi pada sensor.
- **Pengolahan Sinyal:** Setelah mendeteksi suhu dan kelembaban, sensor DHT11 akan mengubah nilai resistansi dan kapasitansi menjadi bentuk sinyal digital yang dapat dibaca oleh mikrokontroler atau Arduino.
- **Komunikasi dengan Mikrokontroler:** Sensor DHT11 menggunakan protokol komunikasi satu kawat (single-wire) untuk berkomunikasi dengan mikrokontroler. Mikrokontroler mengirimkan sinyal start ke sensor, kemudian sensor akan mengirimkan data suhu dan kelembaban dalam bentuk sinyal digital.
- **Pemrosesan Data:** Mikrokontroler atau Arduino akan memproses data suhu dan kelembaban yang diterima dari sensor DHT11. Data ini dapat ditampilkan di layar LCD, disimpan dalam memori, atau digunakan untuk mengendalikan sistem lain, seperti sistem pendingin atau pengatur kelembaban



*Gambar 2. 2 DHT11*



Berikut merupakan Spesifikasi dari Sensor DHT11

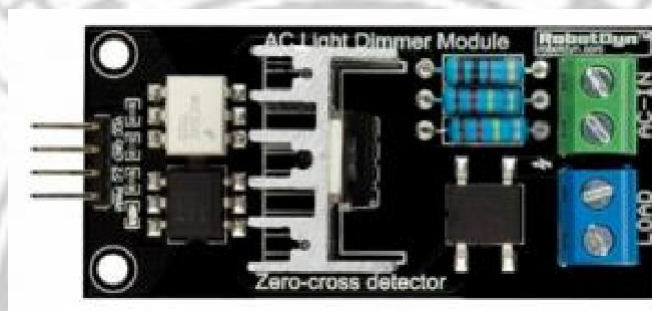
- Tegangan input 3V hingga 5V
- Arus maksimal 2.5mA saat digunakan selama konversi (saat meminta data)
- Kelembapan 20-80% dengan akurasi 5%
- Baik untuk pengukuran suhu 0-50°C dengan akurasi  $\pm 2$  °C
- Pengambilan data minimal 1 Hz (sekali setiap detik)

## 2.5 Pulse Width Modulation (PWM)

Modul PWM (Pulse Width Modulation) adalah perangkat elektronik yang digunakan untuk menghasilkan sinyal PWM, yang digunakan untuk mengendalikan tingkat kecerahan, kecepatan, atau daya dari berbagai perangkat elektronik. Sinyal PWM adalah sinyal digital yang terdiri dari siklus pulsa yang berulang. Modul PWM memungkinkan untuk mengubah duty cycle dari sinyal PWM yang dihasilkan. Duty cycle (rasio antara waktu sinyal ON dan waktu total siklus) dari sinyal PWM yang menentukan tingkat pengendalian perangkat yang terhubung.

Modul PWM dapat digunakan untuk berbagai macam penggunaan yaitu digunakan untuk mengendalikan kecepatan motor DC atau motor servo, mengendalikan tingkat kecerahan lampu LED atau lampu lainnya, digunakan dalam aplikasi pengendalian suhu untuk mengatur elemen pemanas atau elemen pendingin, dan mengendalikan daya yang diberikan ke perangkat elektronik seperti pemanas, blower, atau perangkat AC lainnya. Ini memungkinkan dalam penghematan energi dan pengendalian yang tepat.

Modul PWM memiliki beberapa kaki pin, yaitu pin Z-C (Zero Crossing), Ground, PSM (Phase Shift Modulation) dan VCC. Pin Z-C dipakai untuk mengatur fase sinyal dengan tepat sesuai siklus tegangan AC. Pin PWM digunakan untuk mengontrol fase atau duty cycle dari sinyal PWM yang akan diberikan ke perangkat terhubung.



Gambar 2. 3 Pulse Width Modulation

## 2.6 Monitoring

Menurut *World Health Organization* “Monitoring” adalah suatu proses pengumpulan dan menganalisis informasi dari penerapan suatu program termasuk mengecek secara reguler untuk melihat apakah kegiatan/program itu berjalan sesuai rencana sehingga masalah yang dilihat/ditemui dapat diatasi.

Menurut Peraturan Pemerintah No. 39 Tahun 2006 (dalam IPDN, 2011), disebutkan bahwa *monitoring* adalah suatu kegiatan yang ditujukan untuk mengamati dengan seksama suatu situasi atau kondisi, termasuk perilaku atau kegiatan tertentu. dengan tujuan agar setiap masukan atau informasi yang diperoleh dari pengamatan tersebut dapat

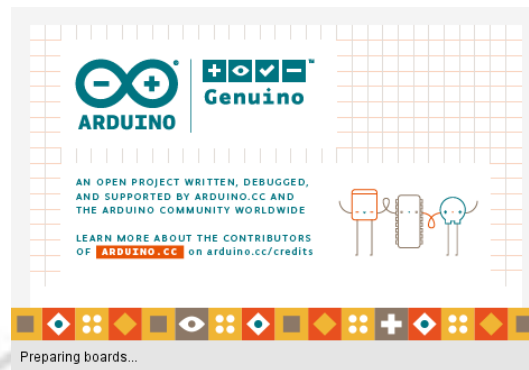
menjadi dasar pengambilan keputusan tindakan selanjutnya yang diperlukan. Tindakan ini diperlukan jika pengamatan menunjukkan bahwa ada hal atau kondisi yang tidak sesuai dengan yang diharapkan semula. Pemantauan dilakukan dengan tujuan agar proyek dapat mencapai tujuannya secara efektif dan efisien dengan memberikan umpan balik kepada manajer proyek di semua tingkatan.

Jadi, Monitoring adalah salah satu bagian dari proses pengumpulan informasi maupun data yang bertujuan untuk menilai hasil yang dilakukan secara berkelanjutan, objektif, meningkatkan efisiensi dan efektivitas program atau kegiatan yang didasarkan pada satuan target dan aktivitas yang direncanakan. Tujuan lainnya yaitu membantu pekerjaan agar tetap di dalam jalur yang tepat, dan memberi tahu manajemen jika terdapat penyimpangan atau kesalahan.

## **2.7 Arduino IDE**

Arduino IDE (Integrated Development Enviroment) adalah software yang digunakan untuk membangun suatu sistem yang secara langsung memberikan perintah atau coding untuk modul atau mikrokontroler. Arduino IDE juga berfungsi untuk membuat, mengedit, memverifikasidan meng-upload sketch prgoram ke dalam suatu modul. Arduino IDE memakai Bahasa pemograman JAVA yang dilengkapi dengan library C/C++, sehingga operasi input dan output menjadi lebih mudah.





Gambar 2. 4 Tampilan Arduino IDE

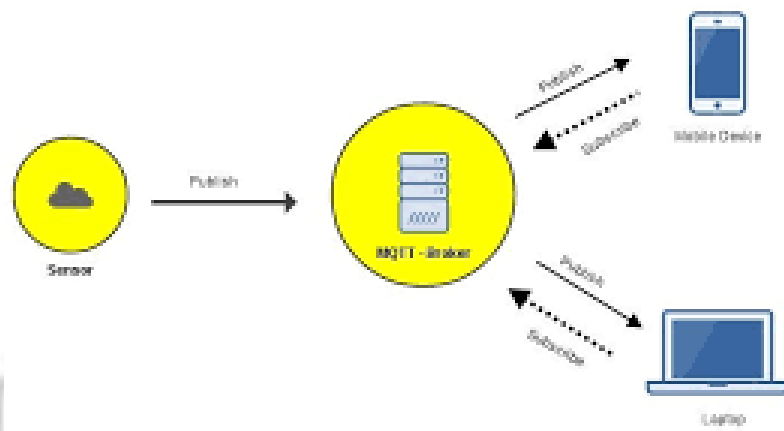
Arduino IDE ini dikembangkan dari software Processing yang dirombak menjadi Arduino IDE khusus untuk pemrograman dengan Arduino. Program yang ditulis dengan menggunakan Arduino Software (IDE) disebut sebagai sketch. Sketch ditulis dalam suatu editor teks dan disimpan dalam file dengan ekstensi. Teks editor pada Arduino Software ini memiliki fitur seperti *cutting/paste* dan *searching/replacing* sehingga memudahkan dalam menulis kode program. Pada tampilan Arduino IDE terdapat beberapa menu yang dibuat untuk mempermudah dalam menulis program. Berikut merupakan fungsi-fungsi pada menu-menu tersebut

1. . Verify berfungsi melakukan kompilasi program yang saat dieditor
2. New berfungsi untuk membuat program baru dengan mengosongkan isi jendela editor saat ini
3. Open berfungsi untuk membuka program yang ada dari sistem file
4. Save berfungsi untuk menyimpan program program saat ini
5. Upload berfungsi untuk mengirim hasil pemrograman dari computer ke memori board mikrokontroler. Saat melakukan upload, harus melakukan pengaturan jenis Arduino.

Pada Software Arduino IDE, terdapat semacam message box berwarna hitam yang berfungsi menampilkan status, seperti pesan *error*, *compile*, dan *upload* program. Di bagian bawah paling kanan Software Arduino IDE, menunjukan *board* yang terkonfigurasi beserta COM Ports yang digunakan. Jika sebuah *sketch* sedang berjalan pada board menerima satu kali konfigurasi atau data lain ketika *sketch* pertama mulai, memastikan bahwa *software* yang berkomunikasi menunggu satu detik setelah membuka koneksi dan sebelum mengirim data ini. Untuk dapat melakukan pemrograman dengan benar maka Arduino IDE harus dikoneksikan dengan *board* Arduino yang telah terinstal pada port tertentu. Untuk mengupload program ke mikrokontroler dapat menggunakan kabel USB sebagai mediana.

## 2.8 MQTT

Message Queuing Telemetry Transport (MQTT) merupakan protokol pengiriman pesan berbasis standar, ringan dan efisien yang menjadi penghubung komunikasi mesin ke mesin. MQTT dikembangkan untuk digunakan dalam aplikasi IoT. MQTT juga merupakan protokol komunikasi yang digunakan pada monitoring jarak jauh. Protokol ini adalah jenis protokol data-agnostic yang artinya protokol ini bisa mengirimkan data apapun seperti data binary, text bahkan XML ataupun JSON dan protokol ini memakai model publish/subscribe daripada model; client server (Atmoko, Riantini, & Hasin, 2017)



Gambar 2. 5 cara kerja MQTT

MQTT memiliki tiga entitas penerbit(*publisher*), pelanggan(*subscriber*) dan broker. *Publisher* adalah perangkat yang mengirim pesan ke broker dengan topik tertentu. *Subscriber* adalah perangkat yang berlangganan untuk menerima pesan dengan topik yang sama. *Broker* adalah perantara yang menghubungkan antara penerima pesan (*subscriber*) dan pengirim pesan(*publisher*) dengan topik yang sama. Kelebihan *MQTT* adalah dirancang khusus dengan bandwidth rendah untuk menyediakan komunikasi yang handal dan efektif di lingkungan dengan sumber daya terbatas, seperti perangkat dengan daya baterai terbatas dan jaringan tidak stabil.

## BAB III METODE PENELITIAN

### 3.1 Tempat dan Waktu Penelitian

Penelitian ini akan dilaksanakan di Kandang Ayam *Closed House* Desa Benteng Gajah, Kecamatan Tompobulu, Kabupaten Maros, Sulawesi Selatan. Waktu pelaksanaan perancangan alat ini selama 6 bulan, yakni bulan April hingga September 2023.

### 3.2 Alat dan Bahan

Adapun daftar alat, bahan yang digunakan yaitu sebagai berikut.

Tabel 3.1 Daftar Alat

No	Nama Alat	Jumlah
1.	Laptop	1 buah
2.	Smartphone	1 buah
3.	Cutter	1 buah
4.	Box	1 buah
5.	Lem Tembak	1 buah

Tabel 3.2 Bahan

No	Nama Bahan	Jumlah
1.	ESP8266	1 buah
2.	Sensor DHT11	1 buah
3.	Modul PWM	1 buah
4.	Kabel Jumper	1 buah
5.	Kabel Listrik 220 V	Secukupnya
6.	Kabel Jumper female-female	Secukupnya
7.	Power Supply	1 buah
8.	Kabel USB	1 buah

### 3.3 Prosedur Penelitian

Prosedur Penelitian merupakan langkah awal dalam merancang suatu alat agar dapat menentukan sistem apa yang akan dibuat sehingga dapat berfungsi secara maksimal. Oleh sebab itu perencanaan harus dilakukan secara matang dan teratur agar sistem yang dihasilkan benar – benar optimal.

Banyaknya permasalahan yang timbul sehingga dibutuhkan teknologi baru untuk menangani beberapa masalah yang sering dialami peternak ayam broiler. Selama ini peternak anak ayam broiler sangat kesusahan dalam menjaga suhu dan kelembapan kandang agar tetap optimal..

Tetapi, jika kita memanfaatkan penggunaan IoT kita dapat memonitoring suhu dan kelembapan melalui smartphone, tentunya akan sangat memudahkan peternak ayam broiler dalam melakukan monitoring, hanya dengan menggunakan jaringan wi-fi.

Berikut akan dijelaskan perencanaan sistem:

- 1) Sistem ini dapat mengontrol dan memonitoring suhu dan kelembapan pada kandang closed house melalui smartphone.
- 2) Sistem ini menggunakan jaringan wi-fi sebagai koneksinya sehingga mudah digunakan oleh peternak ayam broiler.

Dalam prosedur Perancangan alat monitoring ini dimulai dari observasi, studi literatur, analisis permasalahan, perancangan

perangkat keras dan perangkat lunak dan, implementasi sistem, dan pengujian system.

### 3.3.1 Observasi

Observasi merupakan teknik pengumpulan data dengan langsung terjun ke lapangan, dalam hal ini merupakan kandang *closed house* untuk mengamati permasalahan yang terjadi secara langsung di tempat kejadian secara sistematis kejadian-kejadian, perilaku, objek-objek yang dilihat dan hal-hal lain yang diperlukan dalam mendukung penelitian yang sedang berlangsung.

### 3.3.2 Studi Literatur

Dalam penyusunan Tugas Akhir ini, Langkah awal yang dilakukan adalah mencari informasi beserta mengumpulkan data melalui berbagai media cetak atau elektronik yang akan diteliti baik dalam proses perancangan, proses pembuatan sampai pada proses penyelesaian.

### 3.3.3 Analisis Permasalahan

Sebelum merancang Tugas Akhir ini, maka terlebih dahulu harus melakukan identifikasi masalah meliputi:

- 1) Membantu cara kerja peternak ayam boiler dari manual menjadi digital. Agar lebih praktis dan efisien.
- 2) Dapat secara otomatis mengontrol suhu dan kelembapan pada kandang *closed house* agar tetap di suhu standar. Jika terjadi kenaikan

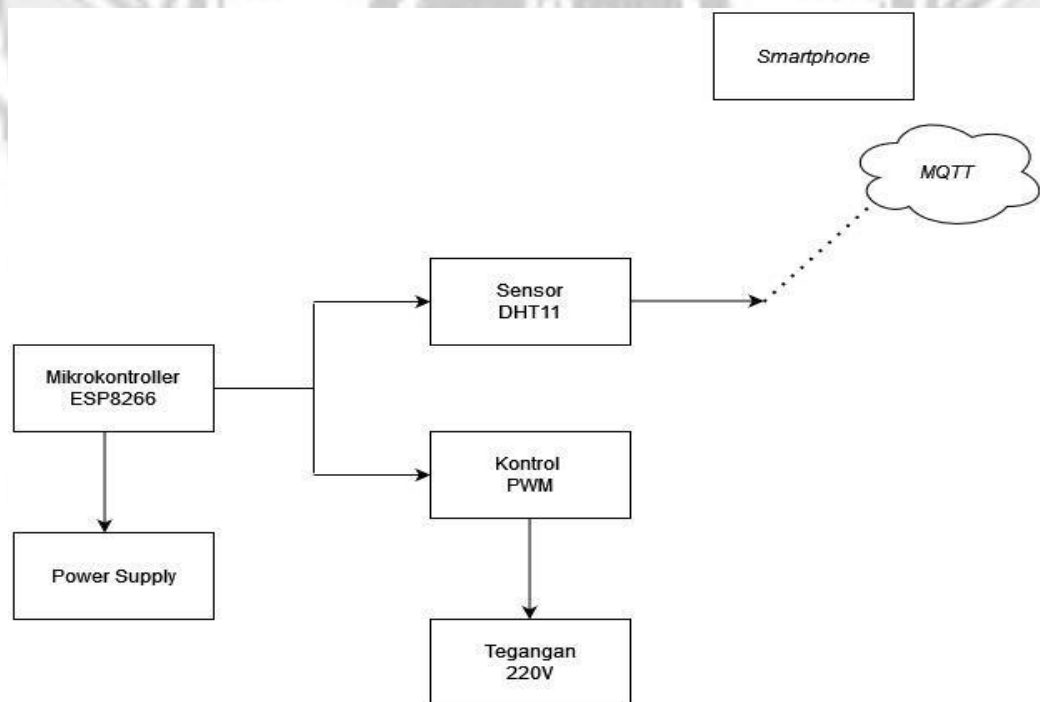
suhu pada kendang maka alat secara otomatis akan menyalakan *blower*

### 3.3.4 Perancangan Perangkat Keras

Pada perancangan perangkat keras (Hardware), terdapat beberapa tahapan yang perlu dilakukan yaitu pembuatan diagram blok, skema perancangan, dan flowchart sistem.

#### a. Diagram Blok

Diagram blok merupakan sebuah diagram berbentuk persegi (blok) yang digunakan untuk menjelaskan suatu proses kerja dari rangkaian tertentu. Berikut merupakan diagram blok dari penelitian ini. Diagram blok sistem dapat dilihat pada gambar 3.1 berikut ini.



Gambar 3 1 Diagram Blok Sistem



Berdasarkan diagram blok sistem seperti yang ditunjukkan pada gambar 3.1, adapun penjelasan dari masing-masing blok diagram sebagai berikut:

- a) ESP8266 berfungsi sebagai mikrokontroler atau pengolah program yang telah dibuat untuk menjalankan beberapa modul yang ada, koneksi ke internet(*wi-fi*), pengirim pesan atau data pembacaan suhu dan kelembapan ke smartphone.
- b) Sensor DHT11 digunakan sebagai pengukur suhu dan kelembapan
- c) Kontrol digunakan untuk mengatur putaran blower
- d) Blower digunakan untuk menurunkan suhu pada kandang jika terjadi kenaikan suhu
- e) *MQTT* digunakan sebagai aplikasi yang menampilkan hasil pengukuran suhu dan kelembapan di dalam kandang
- f) Smartphone digunakan sebagai alat monitoring suhu dan kelembapan
- g) Power supply digunakan sebagai pengisi daya untuk mikrokontroler.
- h) Tegangan 220V digunakan untuk mengaktifkan kontrol

Adapun penjelasan secara keseluruhan sebagai berikut:

ESP8266 yang merupakan mikrokontroler akan mengelola data input atau masukan berupa hasil pengukuran suhu di dalam kandang closed dari sensor suhu DHT11 kemudian data keluaran atau output berupa hasil



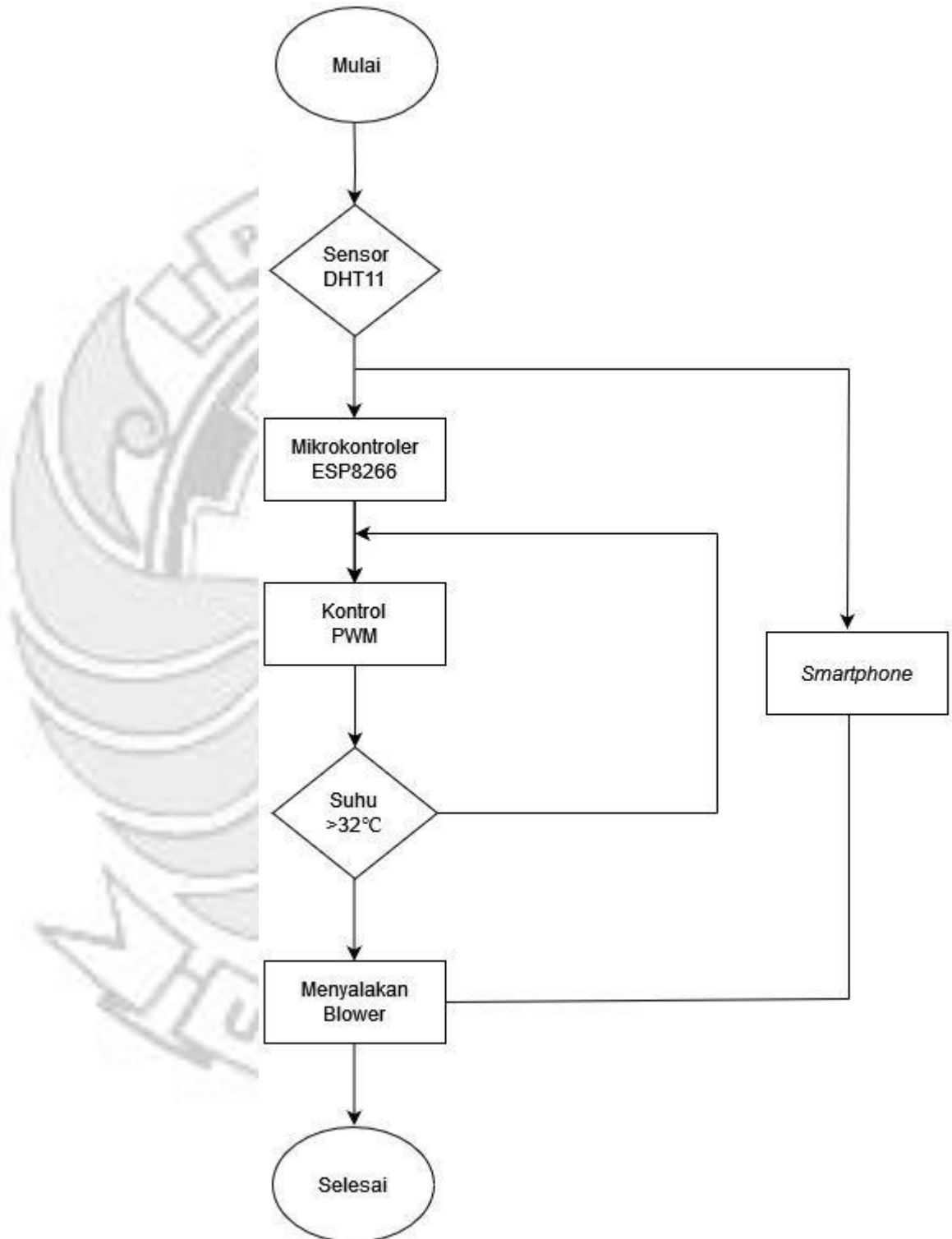
pengukuran suhu yang akan ditampilkan pada aplikasi MQTT yang bisa dipantau melalui smartphone. Jika terjadi kenaikan suhu dari suhu optimal maka mikrokontroler akan memberikan perintah ke modul PWM atau control untuk mengaktifkan blower untuk menurunkan suhu. Mikrokontroler membutuhkan daya dari power supply untuk menyala begitupun dengan control membutuhkan tegangan listrik sebesar 220v

#### b. Skema Perancangan

skema adalah padanan dari „bagan“, „rangka-rangka“, „rancangan“. Definisi lainnya, skema adalah bagan, rangka, kerangka (rancangan dan sebagainya), garis besar, dan denah. Secara etimologi, kata skema adalah kata serapan yang berasal dari bahasa Inggris „schema“. Skema adalah sebuah kata yang sering digunakan oleh seorang perancang atau pembuat suatu alat. Umumnya sebelum membuat sesuatu, perancang akan memikirkan skema alat yang dibuat dengan cara membuat abstrak pada kertas terlebih dahulu atau gambar pada buku.



c. *Flowchart Sistem*



Gambar 3.3 Flowchart Sistem

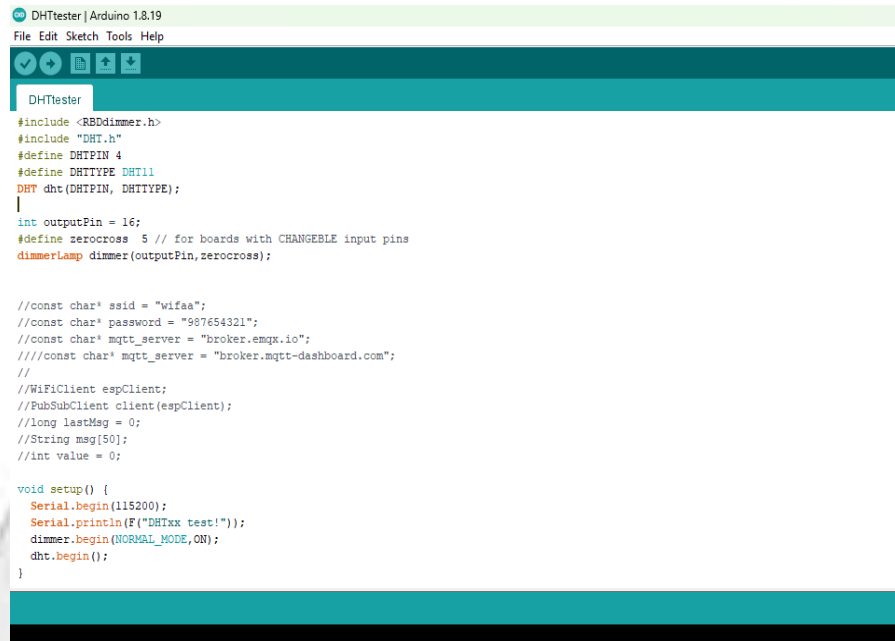
Adapun penjelasan *flowchart* sistem yang ditunjukkan pada gambar 3.3 sebagai berikut:

- Proses dimulai
- Sensor mulai mengukur suhu dan kelembapan di sekitar kandang *closed house*.
- Data pengukuran suhu dari sensor kemudian akan diproses oleh mikrokontroler
- Kemudian data akan di proses juga dari mikrokontroler ke control atau PWM
- Jika suhu  $\geq 32^{\circ}\text{C}$  maka PWM akan secara otomatis menyalakan putaran blower
- Jika suhu tidak  $\geq 32^{\circ}\text{C}$  maka PWM akan menghentikan putaran blower
- Output dari sensor DHT11 adalah smartphone

### 3.3.5 Perancangan Perangkat Lunak (*Software*)

#### 1. Perancangan program pada Arduino IDE

Software Arduino dipakai untuk menulis program yang akan menjalankan system. Program ini digunakan untuk mengolah data hasil pengukuran suhu dan kelembapan pada DHT11 kemudian di proses mikrokontroller dan mengirim data hasil tersebut ke *MQTT* untuk dapat diakses melalui smartphone menggunakan jaringan int

The image shows a screenshot of the Arduino IDE interface. The title bar reads "DHTTester | Arduino 1.8.19". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for opening, saving, and other functions. The main text area contains the following code:

```
DHTTester

#include <RBDdimmer.h>
#include "DHT.h"
#define DHTPIN 4
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
|
int outputPin = 16;
#define zerocross 5 // for boards with CHANGEABLE input pins
dimmerLamp dimmer(outputPin, zerocross);

//const char* ssid = "wifaa";
//const char* password = "987654321";
//const char* mqtt_server = "broker.emqx.io";
//const char* mqtt_server = "broker.mqtt-dashboard.com";
//
//WiFiClient espClient;
//PubSubClient client(espClient);
//long lastMsg = 0;
//String msg[50];
//int value = 0;

void setup() {
  Serial.begin(115200);
  Serial.println(F("DHTxx test!"));
  dimmer.begin(NORMAL_MODE, ON);
  dht.begin();
}
```

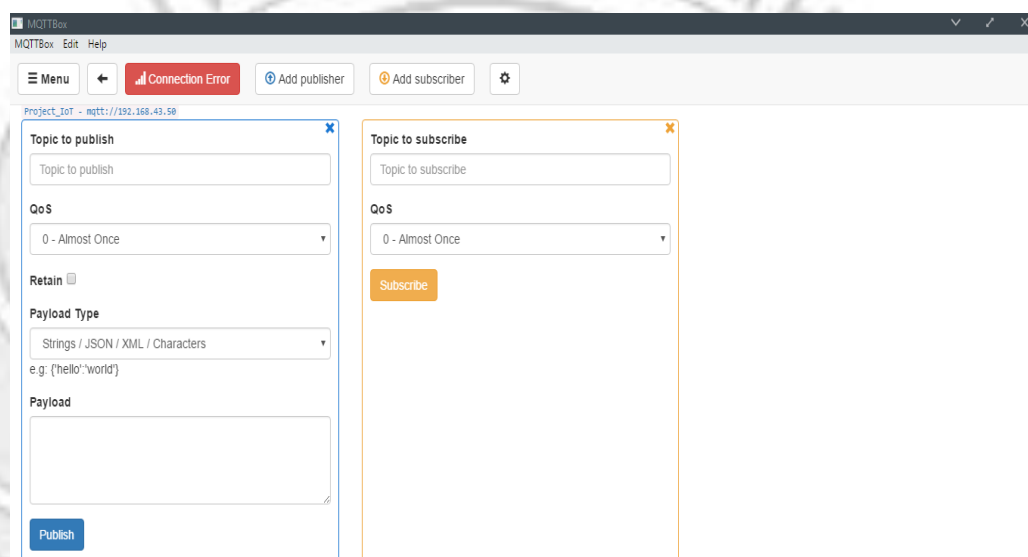
Gambar 3 4 Penulisan Program Arduino IDE

Saat menulis program dipastikan untuk menambah *library* dari beberapa komponen yang digunakan agar alat dan program dapat terintegrasi dan berjalan dengan baik. Sebelum mengunggah *sketch* program pada Arduino IDE, terlebih dahulu melakukan *verify* untuk memastikan tidak terdapat *error* pada penulisan program. Setelah proses *verify* dan tidak terdapat *error* pada *sketch*, maka bisa dilakukan proses *upload* program ke Arduino IDE.

## 2. Perancangan pada protocol MQTT

MQTT *broker* memiliki suatu alamat yang dapat diakses antara *publisher* dan *subscriber* agar dapat melakukan komunikasi dan bertukar informasi. *Broker* berfungsi untuk mengatur lalu lintas informasi di dalam jaringan. MQTT *broker* juga mengenali suatu

data melalui suatu pengelompokkan atau biasa disebut *topic*. Untuk menggunakan protokol MQTT, maka diperlukan *library* PubSubClient pada pemrograman Arduino IDE agar komunikasi melalui protokol MQTT dapat dilakukan.



Gambar 3 5 MQTT



## **BAB V PENUTUP**

### **5.1 Kesimpulan**

Berdasarkan penelitian yang telah dilakukan dari tahap perancangan, pembuatan, dan pengujian, maka dapat disimpulkan sebagai berikut:

1. Alat sistem monitoring dan pengontrolan suhu dan kelembaban pada kandang *closed house* berbasis IoT dapat bekerja dengan baik, dikarenakan semua komponen utama seperti sensor DHT11, ESP8266, dan PWM dapat berjalan dengan baik.
2. Setelah dilakukan pengujian alat monitoring dan pengontrolan suhu, perbandingan hasil sensor DHT11 dan thermometer di kandang. Hasil rata-rata selisih memiliki perbandingan untuk *temperature* sebesar 0,21%
3. Sistem ini mampu mengaktifkan Kontrol untuk mengontrol putaran blower

### **5.2 Saran**

Untuk pengembangan lebih lanjut dalam penulisan laporan Tugas Akhir dengan judul Rancang Bangun Sistem Monitoring dan Pengontrolan suhu pada kandang *closed house* berbasis *IoT*, maka penulis memberikan saran sebagai berikut:

1. Menambah fitur keamanan seperti alarm dan sensor api pada sistem agar
2. Menambahkan untuk memakai fitur bluetooth, agar tidak memerlukan jaringan internet.
3. Membuat aplikasi khusus untuk alat monitoring suhu dan kelembaban pada kandang *closed house* berbasis *IoT*.



## DAFTAR PUSTAKA

- Admin. "Cara Menghubungkan Arduino Dengan Android." <https://al-fikry.com>. Rabu 11 Januari 2023. <https://al-fikry.com/cara->
- Al Khairi, Muhammad Habib. "Menggunakan Sensor Suhu LM35 Dengan Arduino dan Tampilan LCD." [www.mahirelektro.com](http://www.mahirelektro.com). Diakses Senin 9 Januari 2023. <https://www.mahirelektro.com/2020/03/tutorial-arduino-mengakses-sensor-suhu-LM35.html>.
- Athaya, Sabina. "Kandang Ayam Closed House: Pengertian dan Keunggulannya." <https://chickin.id>. Diakses pada Selasa 10 Januari 2023. <https://chickin.id/blog/kandang-ayam-closed-house-pengertian-dan-keunggulannya/>.
- Intial Board. "Merancang Sistem Pengukur Suhu Di Arduino." [www.intialboard.com](http://www.intialboard.com). Senin 9 Januari 2023. <https://www.initialboard.com/merancang-sistem-pengukur-suhu-di-arduino/amp>.
- Marantika, Putri Ayu. "Pengaruh Penggunaan Kandang Modern (Closed House) Terhadap Pendapatan Usah Ternak Ayam Broiler Semantung Farm di Desa Atung Bungsu Kota Pagar Alam." Skripsi, Universitas Sriwijaya, 2020.
- P.G, Fernandus dan Z, Aryawindana. "Perancangan Alat Monitoring Suhu Pada Alat Tambak Udang Dengan Modulasi ASK." Tugas Akhir, Politeknik Negeri Ujung Pandang, 2011.
- R.S. Pressman. "Software Engineering A Practitioner's Approach 7<sup>th</sup> -Roger S Pressman." 2009
- Rahman, Abdul dan Abdullah, M. Baso. "Rancang Bangun Sistem Kontrol, Dan Monitoring Temperatur Dan Kelembapan Pada Ruang Budidaya Jamur Tiram Berbasis Embedded Web Server." Tugas Akhir, Politeknik Negeri Ujung Pandang, 2011.
- Wurlina. 2012

## LAMPIRAN

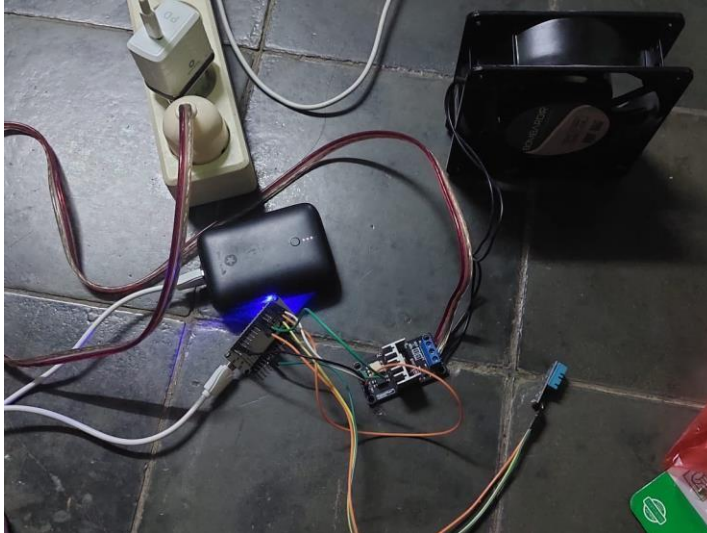
Lampiran 1 Kandang *Closed House*



Lampiran 2 Lokasi alat di dalam kandang



Lampiran 3 Rangkaian Perangkat Keras



Lampiran 4 Pemograman arduino

```

DHT11 | Arduino 1.8.19
File Edit Sketch Tools Help

DHT11

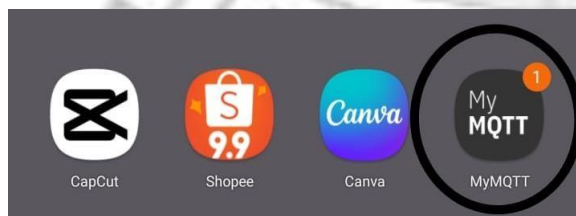
void setup() {
  pinMode(BUILTIN_LED, OUTPUT); // Initialize the BUILTIN_LED pin as an output
  pinMode(v, OUTPUT);
  pinMode(Relay, OUTPUT);
  pinMode(g, OUTPUT);
  digitalWrite(v, HIGH);
  digitalWrite(g, LOW);
  Serial.begin(115200);
  dht.begin();
  //analogReadResolution(10);
  lcd.begin(16, 2);
  lcd.print("Temp.: ");
  setup_wifi();
  client.setServer(mqtt_server, 1883);
  client.setCallback(callback);
}

void loop() {
  int ldr = analogRead(A0);
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  //int t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

  // Check if any reads failed and exit early (to try again).

```

Lampiran 5 Shorcut MQTT pada *smartphone*.



Lampiran 6 Halaman *log in MQTT*



**MQTT Broker**

Host  
broker.emqx.io

Port  
1883 ☐ SSL

**MQTT V3** **MQTT V5**

**Credentials**

Username (optional)

Password (optional)

**Connect**

## Lampiran 7 *Dashboard MQTT*

broker.emqx.io  
Connected

Temperature #31.30, Humidity #68.00, LDR #781  
**outTopicDHT11LdrIdmr** **QoS 0**

Temperature #31.30, Humidity #68.00, LDR #782  
**outTopicDHT11LdrIdmr** **QoS 0**

Temperature #31.30, Humidity #68.00, LDR #781  
**outTopicDHT11LdrIdmr** **QoS 0**

Temperature #31.30, Humidity #68.00, LDR #781  
**outTopicDHT11LdrIdmr** **QoS 0**

Temperature #31.30, Humidity #68.00, LDR #780  
**outTopicDHT11LdrIdmr** **QoS 0**

Temperature #31.30, Humidity #68.00, LDR #781  
**outTopicDHT11LdrIdmr** **QoS 0**

Temperature #31.30, Humidity #68.00, LDR #779  
**outTopicDHT11LdrIdmr** **QoS 0**

Spotify  
Alan Walker **DAPATKAN SPOTIFY**

**Dashboard** **Subscribe** **Publish**

## Lampiran 8 *Datasheet Hardware*

NodeMCU ESP8266

## NodeMCU ESP8266

### *What is NodeMCU?*



The NodeMCU (*Node MicroController Unit*) is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266, designed and manufactured by Espressif Systems, contains the crucial elements of a computer: CPU, RAM, networking (WiFi), and even a modern operating system and SDK. That makes it an excellent choice for Internet of Things (IoT) projects of all kinds.

### *NodeMCU Specifications*

The NodeMCU is available in various package styles. Common to all the designs is the base ESP8266 core. Designs based on the architecture have maintained the standard 30-pin layout. Some designs use the more common narrow (0.9") footprint, while others use a wide (1.1") footprint – an important consideration to be aware of.

The most common models of the NodeMCU are the Amica (based on the standard narrow pin-spacing) and the LoLin which has the wider pin spacing and larger board. The open-source design of the base ESP8266 enables the market to design new variants of the NodeMCU continually.



### ► NodeMCU Technical Specifications

	Official NodeMCU	NodeMCU Carrier Board	LoLin NodeMCU
Microcontroller	ESP-8266 32-bit	ESP-8266 32-bit	ESP-8266 32-bit
NodeMCU Model	Amica	Amica	Clone LoLin
NodeMCU Size	49mm x 26mm	49mm x 26mm	58mm x 32mm
Carrier Board Size	n/a	102mm x 51mm	n/a
Pin Spacing	0.9" (22.86mm)	0.9" (22.86mm)	1.1" (27.94mm)
Clock Speed	80 MHz	80 MHz	80 MHz
USB to Serial	CP2102	CP2102	CH340G
USB Connector	Micro USB	Micro USB	Micro USB
Operating Voltage	3.3V	3.3V	3.3V
Input Voltage	4.5V-10V	4.5V-10V	4.5V-10V
Flash Memory/SRAM	4 MB / 64 KB	4 MB / 64 KB	4 MB / 64 KB

	Official NodeMCU	NodeMCU Carrier Board	LoLin NodeMCU
Digital I/O Pins	11	11	11
Analog In Pins	1	1	1
ADC Range	0-3.3V	0-3.3V	0-3.3V
UART/SPI/I2C	1 / 1 / 1	1 / 1 / 1	1 / 1 / 1
WiFi Built-In	802.11 b/g/n	802.11 b/g/n	802.11 b/g/n
Temperature Range	-40C - 125C	-40C - 125C	-40C - 125C
Product Link		NodeMCU	NodeMCU

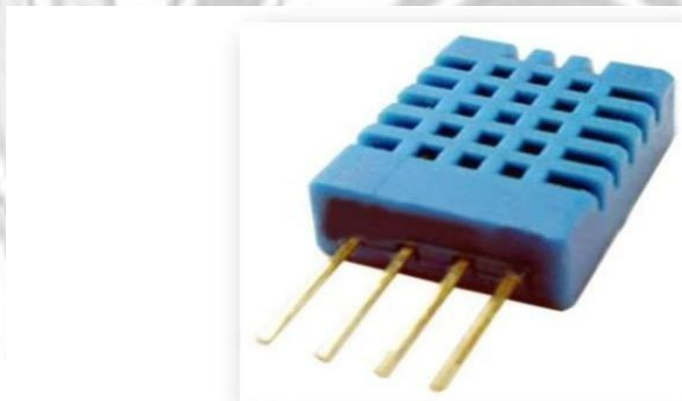


## DHT 11 Humidity & Temperature Sensor

---

### 1. Introduction

DHT11 Temperature & Humidity Sensor features a temperature & humidity sensor complex with a calibrated digital signal output. By using the exclusive digital-signal-acquisition technique and temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability. This sensor includes a resistive-type humidity measurement component and an NTC temperature measurement component, and connects to a high-performance 8-bit microcontroller, offering excellent quality, fast response, anti-interference ability and cost-effectiveness.



Each DHT11 element is strictly calibrated in the laboratory that is extremely accurate on humidity calibration. The calibration coefficients are stored as programmes in the OTP memory, which are used by the sensor's internal signal detecting process. The single-wire serial interface makes system integration quick and easy. Its small size, low power consumption and up-to-20 meter signal transmission making it the best choice for various applications, including those most demanding ones. The component is 4-pin single row pin package. It is convenient to connect and special packages can be provided according to users' request.

### 2. Technical Specifications:

#### Overview:

Item	Measurement Range	Humidity Accuracy	Temperature Accuracy	Resolution	Package
DHT11	20-90%RH 0-50 °C	±5 %RH	±2 °C	1	4 Pin Single Row



### Detailed Specifications:

Parameters	Conditions	Minimum	Typical	Maximum
<b>Humidity</b>				
<b>Resolution</b>		1%RH	1%RH	1%RH
			8 Bit	
<b>Repeatability</b>			± 1%RH	
<b>Accuracy</b>	25 °C		± 4%RH	
	0-50 °C			± 5%RH
<b>Interchangeability</b>	Fully Interchangeable			
<b>Measurement Range</b>	0 °C	30%RH		90%RH
	25 °C	20%RH		90%RH
	50 °C	20%RH		80%RH
<b>Response Time (Seconds)</b>	1/e(63%)25 °C , 1m/s Air	6 S	10 S	15 S
<b>Hysteresis</b>			± 1%RH	
<b>Long-Term Stability</b>	Typical		± 1%RH/year	
<b>Temperature</b>				
<b>Resolution</b>		1 °C	1 °C	1 °C
		8 Bit	8 Bit	8 Bit
<b>Repeatability</b>			± 1 °C	
<b>Accuracy</b>		± 1 °C		± 2 °C
<b>Measurement Range</b>		0 °C		50 °C
<b>Response Time (Seconds)</b>	1/e(63%)	6 S		30 S

### 3. Typical Application (Figure 1)

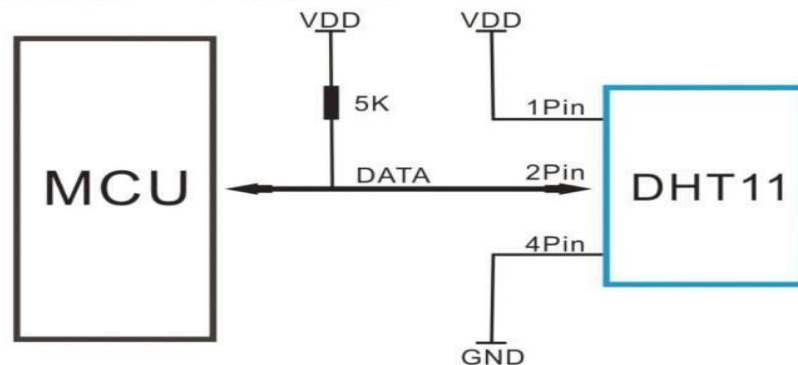


Figure 1 Typical Application

Note: 3Pin – Null; MCU = Micro-computer Unite or single chip Computer

When the connecting cable is shorter than 20 metres, a 5K pull-up resistor is recommended; when the connecting cable is longer than 20 metres, choose a appropriate pull-up resistor as needed.

### 4. Power and Pin

DHT11's power supply is 3-5.5V DC. When power is supplied to the sensor, do not send any instruction to the sensor in within one second in order to pass the unstable status. One capacitor valued 100nF can be added between VDD and GND for power filtering.

### 5. Communication Process: Serial Interface (Single-Wire Two-Way)

Single-bus data format is used for communication and synchronization between MCU and DHT11 sensor. One communication process is about 4ms.

Data consists of decimal and integral parts. A complete data transmission is **40bit**, and the sensor sends **higher data bit** first.

**Data format:** 8bit integral RH data + 8bit decimal RH data + 8bit integral T data + 8bit decimal T data + 8bit check sum. If the data transmission is right, the check-sum should be the last 8bit of "8bit integral RH data + 8bit decimal RH data + 8bit integral T data + 8bit decimal T data".

## Lampiran 9 Kode Pemrograman Arduino IDE

```
#include <ESP8266WiFi.h>
#include <Ultrasonic.h>
#include <Wire.h>
#include <PubSubClient.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16,2);
#define relayPin D0
#define relayPin D4
#define SENSOR D5
#define MSG_BUFFER_SIZE (50)

// Connect to the WiFi
const char* ssid = "IOT";
const char* password = "";
const char* mqtt_server = "192.168.43.50";
WiFiClient espClient;
PubSubClient client(espClient);
unsigned long lastMsg = 0;
char msg[MSG_BUFFER_SIZE];

int Trig_Pin = D6;
int echo_Pin = D7;
long duration;
int jarak;
int value = 0;
long currentMillis = 0;
long previousMillis = 0;
int interval = 1000;
float calibrationFactor = 6.5;
volatile byte pulseCount;
byte pulse1Sec = 0;
float flowRate;
float flowLitres;
float totalLitres;

void IRAM_ATTR pulseCounter()
{
    pulseCount++;
}

void setup_wifi() {

    delay(10);
    // We start by connecting to a WiFi network
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(ssid);

    WiFi.mode(WIFI_STA);
    WiFi.begin(ssid, password);

    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }

    randomSeed(micros());

    Serial.println("");
    Serial.println("WiFi connected");
```

```

Serial.println("IP address: ");
Serial.println(WiFi.localIP());
}

void callback(char* topic, byte* payload, unsigned int length) {
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] ");
  for (int i = 0; i < length; i++) {
    Serial.print((char)payload[i]);
  }
  Serial.println();
}

void reconnect() {
  // Loop until we're reconnected
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
    // Create a random client ID
    String clientId = "ESP8266Client-";
    clientId += String(random(0xffff), HEX);
    // Attempt to connect
    if (client.connect(clientId.c_str())) {
      Serial.println("connected");
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      // Wait 5 seconds before retrying
      delay(5000);
    }
  }
}

void setup()
{
  Serial.begin(9600);
  setup_wifi();
  client.setServer(mqtt_server,1883);
  client.setCallback(callback);

  digitalWrite(relayPin, HIGH);
  pinMode(relayPin, OUTPUT);
  pinMode(relayPin2, OUTPUT);
  digitalWrite(relayPin2, HIGH);
  pinMode(Trig_Pin, OUTPUT);
  pinMode(echo_Pin, INPUT);
  pinMode(SENSOR, INPUT_PULLUP);

  pulseCount = 0;
  flowRate = 0.0;
  flowLitres = 0;
  totalLitres = 0;
  previousMillis = 0;

  attachInterrupt(digitalPinToInterrupt(SENSOR), pulseCounter, FALLING);

  lcd.begin(16,2);
  lcd.init();
  lcd.backlight();
  Serial.begin(115200);
}

```

















