

**SISTEM MONITORING DAN PERINGATAN KETINGGIAN AIR KOLAM IKAN DENGAN SENSOR HC-SR04**

Muhammad Rafli<sup>1</sup>, Nonot Wisnu Karyanto<sup>2</sup>

<sup>1,2</sup> Program Studi Teknik Informatika, Fakultas Teknik, Universitas Wijaya Kusuma Surabaya  
 muhrafli329@gmail.com<sup>1</sup>, nonotwk@uwks.ac.id<sup>2</sup>

**Abstrak**

Banyaknya pembudidaya ikan yang merugi akibat ikan keluar dari kolam atau mati karena kekurangan air dikarenakan kurangnya pengawasan terhadap kolam ikan. Alat monitoring dan peringatan ketinggian air kolam ikan adalah alat yang dapat digolongkan kedalam sistem otomatis yang dapat bekerja secara otomatis sehingga tidak perlu melakukan pengawasan secara manual. Terciptanya alat ini akan membantu peternak ikan Nila dalam menjalankan usahanya. Dengan adanya alat sensor *HC-SR04* ini akan dapat mengurangi kerugian peternak ikan karena ketinggian kolam ikan selalu terkendali. Pembuatan sistem monitoring dan peringatan ketinggian air kolam ikan bekerja secara otomatis melalui beberapa tahapan. 1) Perancangan sistem; 2) Analisa kebutuhan; 3) Desain sistem; 4) Pengodean program; 5) Pengujian; 6) Hasil uji coba. Berdasarkan hasil pengujian bisa disimpulkan bahwa alat ini bekerja dengan baik. Alat ini juga bisa memberikan alarm berupa *led* dan *buzzer* ketika ketinggian air kurang atau lebih dari batas aman dan akan memberikan notifikasi *Twitter*.

**Kata Kunci** : *Monitoring, HC-SR04, Kolam ikan Nila, Twitter*

**Abstract**

*Many fish cultivators lose money due to fish leaving the pond or die from lack of water due to lack of supervision of fish ponds. The fish pond water level monitoring and warning tool is a tool that can be classified into an automatic system that can work automatically so there is no need for manual supervision. The creation of this tool will help Tilapia farmers in running their business. With the HC-SR04 sensor, it will be able to reduce the losses of fish farmers because the height of the fish pond is always under control. The creation of a fish pond water level monitoring and warning system works automatically through several stages. 1) System design; 2) Needs analysis; 3) System design; 4) Program coding; 5) Testing; 6) Test results. Based on the test results, it can be concluded that this tool works well. This tool can also provide alarms in the form of LEDs and buzzers when the water level is less or more than the safe limit and will provide Twitter notifications.*

**Keywords** : *Monitoring, HC-SR04, Tilapia fish pond, Twitter*

**I PENDAHULUAN**

Dengan banyaknya sumber air di Indonesia akan membuka banyak peluang bagi masyarakat yang akan membuka usaha pembudidayaan ikan. Budidaya ikan banyak dikembangkan bukan hanya sebagai peliharaan melainkan menjadi usaha. Ikan Nila yang dipelihara mulai dari bibit, ikan kecil, dan ikan yang siap untuk dijual. Pembudidaya ikan Nila sering kali banyak menggunakan media kolam tanah atau tembok semen. [1].

Kolam ikan adalah salah satu faktor pendukung berhasil tidaknya usaha budidaya ikan Nila. Kolam ikan berfungsi sebagai penampung buatan sebagai pengganti habitat alami ikan yang sengaja diciptakan agar ikan bisa hidup dan beranak pinak dengan baik. Apakah yang dimaksud dengan kolam ikan? Kolam ikan adalah perairan yang luasnya terbatas dan sengaja dibuat oleh pembudidaya sebagai pengganti habitat alami ikan. Yang artinya kolam mudah diisi air dan mudah

dikeringkan sebagai mudah dikelola untuk mendapatkan hasil yang maksimal. [2].



Gambar 1 kolam ikan

Teknologi dari hasil peradapan manusia semakin maju, akan sangat membantu memenuhi kebutuhan pada zaman modern. Monitoring kolam ikan ini dibuat agar pembudidaya dapat memantau ketinggian air kolam

ikan. [3]. Dengan menggunakan *NodeMCU* sebagai mikrokontroler yang akan menjadi otak dari alat monitoring kolam ikan [4].

*NodeMCU* dapat diartikan pengembang dari ESP 8266 dengan *firmware* yang berbasis e-Lua. Di dalam *NodeMCU* terdapat *micro USB port* yang berfungsi sebagai penyambung antara *microcontroller* dengan *pc* untuk memasukkan pemrograman dan juga dapat digunakan sebagai *power supply*. Selain itu ada juga tombol *push button* yaitu sebagai tombol *reset* ataupun *flash*. Bahasa yang digunakan *NodeMCU* adalah bahasa pemrograman *Lua* yang berupa modul-modul dari *ESP8266*. Bahasa *Lua* adalah bahasa yang menggunakan logika pemrograman yang sama dengan Bahasa C. akan tetapi berbeda *syntak*. Bahasa *Lua* menggunakan *tool Lua loader* maupun *Lua uploder*. *NodeMCU* juga bisa dapat menggunakan *Arduino IDE* sebagai *software* pengkodean dengan sedikit mengubah *board manager* pada *Arduino IDE* dan memilih *NodeMCU* sebagai *board* yang digunakan sebelum *board* dapat digunakan *board* terlebih dahulu harus di *Flash* agar *board* siap digunakan atau *support* pada *tool* yang digunakan. Jika akan menggunakan *Arduino IDE* menggunakan *firmware* yang cocok yaitu *firmware* keluaran *AT Command*. Untuk penggunaan *tool loader Firmware* yang di gunakan adalah *firmware NodeMCU*. [5].

Namun, jika curah hujan sangat tinggi dapat menyebabkan air kolam ikan meluap dan ketika cuaca sangat panas air pada kolam ikan akan berkurang dikarenakan air pada kolam ikan menguap. Meluapnya air kolam ikan ataupun menguapnya air kolam ikan dapat menyebabkan air kolam ikan mati. Sehingga menyebabkan pembudidaya kolam ikan Nila mengalami kerugian karena ikan mati. [3]

Untuk meminimalisir kerugian pembudidaya ikan Nila dapat digunakan sistem monitoring ketinggian air kolam ikan Nila yang dapat memberikan peringatan yang akan memberikan informasi jika ketinggian air kolam melewati batas aman dan akan memberikan notifikasi melalui *Twitter*.

Penelitian ini akan membuat sistem monitoring ketinggian air kolam ikan dengan sensor HC-SR04 sebagai pendeteksi ketinggian air kolam ikan Nila dan akan ada alrm berupa *buzzer* dan *led* sebagai peringatan ketinggian air kolam ikan Nila melewati batas aman. Pemberian notifikasi berupa *Twitter*.

## II. METODE

Metodologi yang digunakan pada penelitian ini adalah alur siklus mulai dari perancangan sistem, analisis kebutuhan, perancangan alat dan bangun alat. Agar sistem berjalan dengan baik sesuai apa yang diinginkan.

### 1 : Perancangan Sistem

**Perancangan Input :** Pada tahap perancangan input ada beberapa inputan yang digunakan yaitu Sensor HC-SR04 sebagai pendeteksi ketinggian air kolam ikan, dan Sensor DHT11 sebagai pendeteksi suhu udara yang ada di sekitar sensor.

**Perancangan Output :** Pada tahap perancangan output terdapat beberapa *output* yaitu pemberian Notifikasi *Twitter*, *LED* sebagai indikator , *Buzzer* yang juga sebagai indikator ,dan *LCD 16x2* untuk memunculkan hasil angka yang berhasil di tangkap sensor.

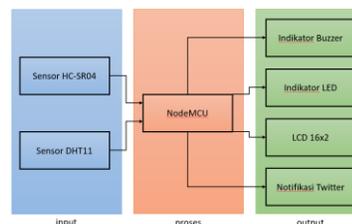
**Perancangan Proses :** Pada tahap perancangan alat terdapat input, proses, dan *output*. Data yang berasal dari penangkapan oleh sensor yang kemudian akan diolah *NodeMCU*.

### 2 : Analisa Kebutuhan

**Bahan penelitian :** Bahan-bahan yang akan digunakan pada alat ini adalah *NodeMCU*, *Breadboard*, *DHT11*, *HC-SR04*, *LED*, *Buzzer*, *LCD 16x2*, *Modul I2C*, Kabel *Jumper*.

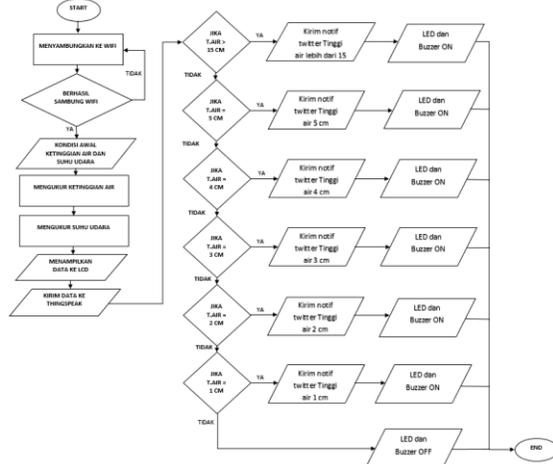
**Perangkat penelitian :** Pada penelitian perangkat ini menggunakan perangkat keras yaitu PC dengan spesifikasi sebagai berikut : *Processor Intel Core i3*, RAM 4 GB, *Hardisk 1 TB*, *Handphone android* minimal *Jellybean (cocktail)*, Kabel USB. Perangkat lunak yang digunakan pada alat ini adalah pemrograman yang digunakan pada penilaian ini sebagai berikut: *Microsoft Windows 7*, *Arduino IDE 1.8.15*, Bahasa pemrograman C.

**Desain Sistem :** Pada desain sistem kerja dapat disimpulkan secara garis besar adalah sebagai berikut: Perancangan alat merupakan proses terpenting pada proses pembuatan alat. Tahapan yang pertama adalah pembuatan blok diagram, *flowchat* dan rangkaian alat keseluruhan.



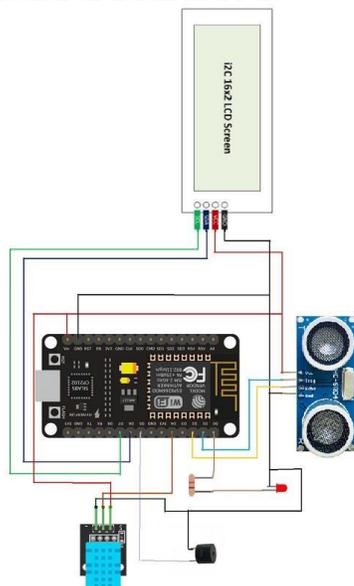
Gambar 2 Blok Diagram

Pada gambar blok diagram tersebut menggambarkan alur penelitian secara garis besar. Dengan mekanisme cara kerja, yaitu user dapat memantau ketinggian dan kerendahan kolam ikan Nila dan *NodeMCU* akan memberikan informasi ke user dengan indikator lampu, *Buzzer* dan *Twitter*.



Gambar 3 Flowchart diagram

*flowchart* adalah yang akan menjelaskan alur dari sistem monitoring ketinggian air dan kerendahan air kolam ikan Nila.



Gambar 4 rangkaian keseluruhan

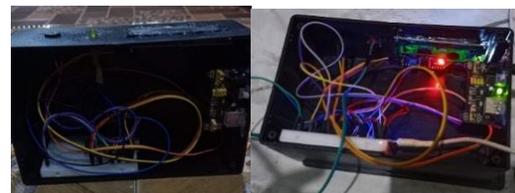
Pada gambar 4 dapat kita lihat hubungan hubungan antar pin mulai dari *nodemcu* dengan *HC-RS04*, *NodeMCU* dengan *DHT11*, *NodeMCU* dengan *LCD16x2* dan *I2c*, *LED* dengan *NodeMCU*, dan *Buzzer NodeMCU*, dan resistor untuk *LED*.

**Pengkodean Program :** Pada tahap ini saya menggunakan *arduino IDE* sebagai *software developmen* dan bahasa C di masukkan ke *NodeMCU* sebagai Mikrokontroler.

**Pengujian :** Pada tahap ini perancangan skenario pengujian yaitu pada sensor alat, indikator peringatan yang berupa *LED* dan *Buzzer*.

### III. HASIL DAN PEMBAHASAN

Pada tahap ini akan menjelaskan beberapa tahap uji coba dan menganalisa kerja alat. Sebelum pengujian alat dilakukan sebelumnya harus mengetahui cara menggunakan alat monitoring ketinggian air kolam ikan. Pastikan semua rangkaian komponen tersusun dengan benar dan pastikan program telah di *upload* ke *NodeMCU*. Setelah semua telah dilakukan hubungkan alat dengan adaptor sebagai sumber power untuk memulai mengaktifkan alat akan bekerja dengan lancar. Antara lain sebagai berikut : *NodeMCU* sebagai *mikrokontroler* dan mengenali bahasa pemrograman, *NodeMCU* sebagai modul penghubung ke internet, *HC-SR04* sebagai sensor pendeteksi ketinggian air, *DHT11* sebagai sensor pendeteksi suhu dan kelembaban, Adaptor sebagai pengubah tegangan listrik nilai yang tinggi menjadi 9 volt, *Buzzer* sebagai alarm jika ketinggian air sudah melewati batas aman, *LED* sebagai indikator jika tinggi air sudah melewati batas aman.



Gambar 5 Rangkaian Alat

Pada gambar 5 bisa dilihat rangkaian keseluruhan rangkaian alat dalam keadaan *off* dan *on*. Dimana alat yang sedang *on* bisa dilihat dengan indikasi lampu yang menyala.

Selanjutnya adalah proses pengkodean program.

Tabel 1 Modul yang digunakan

Modul yang digunakan
1 : #include <ESP8266WiFi.h>
2 : #include <WiFiClientSecure.h>
3 : #include <LiquidCrystal_I2C.h>
4 : #include <Wire.h>
#include <DHT.h>

Pada tabel 1 adalah *library* apa saja yang digunakan.

Tabel 2 pengaturan LCD dengan I2C  
Pengaturan lcd dengan menggunakan i2c

```
1: LiquidCrystal I2C lcd(0x27, 16, 2);
```

Pada tabel 2 adalah penyetingan *lcd* dengan *i2c*.

Tabel 3 mendefinisikan pin-pin *Nodemcu*  
Mendefinisikan pin-pin *NodeMCU*

```
1 : #define triggerPinD6
2 : #define echoPinD7
3 : #define pinBuzzerD3
4 : #define pinLedD5
5 : #define DHTPIN D4
```

Pada tabel 3 adalah mendefinisikan pin-pin pada *NodeMCU*.

Tabel 4 Deklarasi sensor DHT11

```
Deklarasi sensor DHT11
1 : #define DHTTYPE DHT11DHT
    dht(DHTPIN, DHTTYPE);
```

Pada tabel 4 adalah deklarasi untuk sensor *DHT11*.

Tabel 5 informasi penyambungan wifi

```
Informasi penyambungan wifi
1 : #ifndef STASSID
2 : #define STASSID "Rusiaa"
    // deklarasi SSID atau nama wifi
3 : #define STAPSK "Merdeka123"
    // deklarasi Pass atau password
    wifi
4 : #endif
```

Pada tabel 5 adalah informasi wifi yaitu deklarasi SSID dan password wifi yang digunakan.

Tabel 6 deklarasi variabel yang digunakan

```
Deklarasi variabel yang digunakan
1 : long duration;
2 : long jarak;
3 : long tinggiAir;
4 : long batas=44; //tinggi sensor
    dengan dasar kolam
5 : long peringatan1 =15; //batas
    peringatan jika air lebih dari 15
    cm alm akan bunyi
6 : long peringatan2 =5; //batas
    peringatan jika air kurang dari 5
    cm alm akan bunyi
7 : float h ; //deklarasi kelembaban
8 : float t ; //deklarasi suhu
```

Pada tabel 6 adalah deklarasi variabel apa saja yang digunakan.

Tabel 7 Informasi koneksi wifi dan host yang digunakan  
Informasi tentang koneksi *wifi* dan host yang digunakan

```
1 : const char* username = STASSID;
2 : const char* password = STAPSK;
3 : const char* host =
    "api.thingspeak.com";
4 : const int httpsPort = 443;
5 : const char fingerprint[] PROGMEM =
    "271892dda426c30709b97ae6c521b95b48f
    716e1";
```

Pada tabel 7 adalah informasi *wifi* dan *host* yang digunakan.

Tabel 8 Informasi Wifi ke serial monitor

```
Informasi Wifi ke serial monitor
//memunculkan informasi wifi
1 : Serial.print("sambungkan ke ");
2 : Serial.println(ssid);
//proses menyambungkan ke wifi
3 : WiFi.mode(WIFI_STA);
4 : WiFi.begin(ssid, password);
5 : while (WiFi.status() !=
    WL_CONNECTED) {
6 : delay(1000); //delay 500
    milisecond
7 : Serial.print(".");}
//memunculkan ke serial monitor jika
    wifi connect
8 : Serial.println(" ");
9 : Serial.println("berhasil
    menyambungkan");
10 : Serial.println("Welcome
    Rafli");
11 : Serial.println("IP ");
12 : Serial.println(WiFi.localIP());
```

Pada tabel 8 adalah informasi *wifi* yang akan dimunculkan ke serial monitor.

Tabel 9 Deklarasi pin sebagai input dan output

```
Deklarasi pin sebagai input dan output
1 : pinMode(triggerPin, OUTPUT);
2 : pinMode(echoPin, INPUT);
3 : pinMode(pinBuzzer, OUTPUT);
4 : pinMode(pinLed, OUTPUT);
5 : dht.begin();
6 : lcd.init();
7 : lcd.backlight();
```

Pada tabel 9 adalah pendeklarasian pin sebagai *input* dan *output*

Tabel 10 Fungsi pembacaan ketinggian air

Fungsi pembacaan ketinggian air	
	// pembacaan ketinggian air
1 :	void baca_sensor_hcsr(){
2 :	digitalWrite(triggerPin, LOW);
3 :	delay(2);
4 :	digitalWrite(triggerPin, HIGH);
5 :	delay(10);
6 :	digitalWrite(triggerPin, LOW);
7 :	dur = pulseIn(echoPin, HIGH);
8 :	jarak = (dur/2) / 29.1;
9 :	tinggiair=batas-jarak;
	//menampilkan tinggi air ke serial monitor
10 :	Serial.print("tinggi air : ");
11 :	Serial.print(tinggiair);
12 :	Serial.print("cm");
13 :	Serial.println("");
	//menampilkan tinggi air ke lcd
14 :	lcd.setCursor(0,0);
15 :	lcd.print("Tinggi air:");
16 :	lcd.print(tinggiair);
17 :	lcd.print("cm");
	// kondisi untuk alarm hidup atau mati
18 :	if(tinggiair>=peringatan1){
19 :	digitalWrite(pinBuzzer, HIGH);
20 :	digitalWrite(pinLed, HIGH);
21 :	delay(2000);
22 :	}else if(tinggiair<=peringatan2){
23 :	digitalWrite(pinBuzzer, HIGH);

Fungsi pembacaan ketinggian air

```

24 : digitalWrite(pinLed, HIGH);
25 : delay(2000);
26 : }else{
27 : digitalWrite(pinBuzzer, LOW);
28 : digitalWrite(pinLed, LOW); }}

```

Pada tabel 10 adalah perintah pembacaan ketinggian air oleh sensor *HC-SR04*.

Tabel 11 Fungsi pembacaan Suhu

Fungsi pembacaan Suhu	
1 :	baca_sensor_dht11(){
2 :	h = dht.readHumidity(); // untuk mendeteksi kelembapan
3 :	t = dht.readTemperature();
4 :	Serial.print("kelembapan udara: ");
5 :	Serial.print(h);
6 :	Serial.print("\n");
7 :	Serial.print("Suhu udara: ");
8 :	Serial.print(t);
9 :	Serial.print("°C ");
10 :	Serial.print("\n");
	//menampilkan suhu ke lcd i2c
11 :	lcd.setCursor(0,1);
12 :	lcd.print("Suhu :");
13 :	lcd.print(t);
	lcd.print(" C ");}

Pada tabel 11 adalah perintah pembacaan Suhu udara oleh sensor *DHT11*

Tabel 12 Fungsi kirim data

Fungsi kirim data	
1 :	void kirimdata(){
2 :	WiFiClientSecure client;

Fungsi kirim data	
3 :	Serial.print("menyambungkan ");
4 :	Serial.println(host);
5 :	if (!client.connect(host, httpsPort)) { Serial.println("gagal menyambungkan"); return;} //perbarui tinggi air dan suhu pada thingspeak
6 :	String url = "/update?api_key=U5Z4D963JM7ZG FSN&field1=";
7 :	url += tinggiair;
8 :	url += "&field2=";
9 :	url += t; //menampilkan informasi jika sukses kirim data pada serial monitor
10 :	client.print(String("GET ") + url + " HTTP/1.1\r\n" + "User-Agent: BuildFailureDetectorESP8266\r\n" + "Connection: close\r\n\r\n");
11 :	Serial.println("kirim permintaan");
12 :	while (client.connected()) { String line = client.readStringUntil('\n'); if (line == "\r") {Serial.println("request"); break;}}
13 :	String line = client.readString();
14 :	If (line.toInt() > 0){ Serial.print("kirim data berhasil!"); delay(1000);} else { Serial.println("kirim data gagal")}
15 :	Serial.println("-----");
16 :	Serial.println(line);
17 :	Serial.println("-----");
18 :	Serial.println("selesai tutup koneksi");}

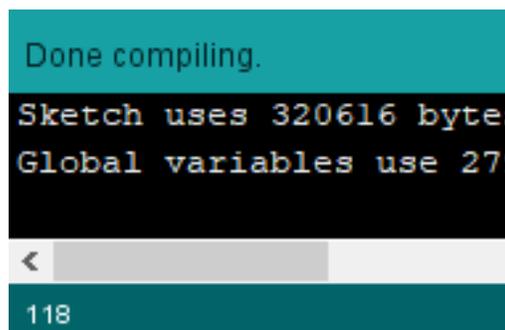
Pada tabel 12 adalah perintah pengiriman data ke thingspeak

Tabel 13 Pemanggilan fungsi fungsi

Pemanggilan fungsi fungsi	
1 :	void loop() {
2 :	baca_sensor_hcsr(); // perintah menjalankan fungsi baca sensor hcsr
3 :	baca_sensor_dht11(); // perintah menjalankan

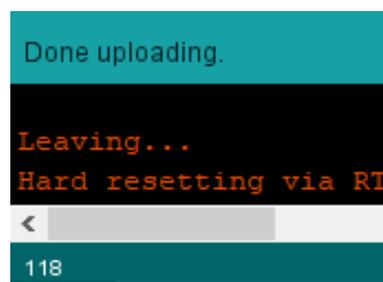
Pemanggilan fungsi fungsi	
	fungsi baca sensor dht11
4 :	delay(100);
5 :	kirimdata(); // perintah menjalankan kirim data ke thingspeak
6 :	delay(2000);}

Pada tabel 12 adalah pemanggilan fungsi atau perintah berjalannya program.



Gambar 6 compile done

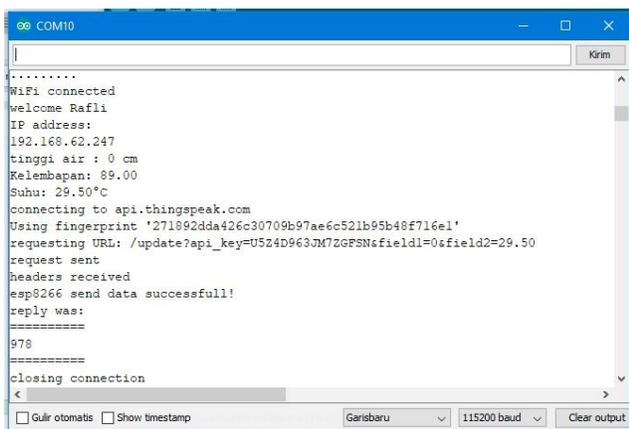
Setelah proses *coding* selesai, cek *coding* apakah ada yang *error*. Dengan cara *compile*. Jika proses *compile* sukses akan ada hasil "Done compiling" seperti pada gambar 6.



Gambar 7 done upload

Setelah *compile* sukses selanjutnya adalah *upload* program arduino ide ke *board nodemcu*. Jika upload sukses akan muncul "Done uploading" seperti pada gambar 7.

Selanjutnya adalah pengecekan program di serial monitor arduino ide



Gambar 8 Serial monitor

Pada gambar 8 bisa dijelaskan sebagai berikut:

1. Pada baris pertama “...” yaitu
2. sebagai indikator alat sedang mencoba konek ke *wifi*
3. Baris ke 2 “*WiFi connected*” yaitu *wifi* berhasil di sambungkan
4. Baris ke 4 dan ke 5 adalah indikator alat sudah sukses konek ke *wifi* dengan memunculkan ip adres di serial monitor
5. Baris ke 6 adalah Tinggi air kolam ikan
6. Baris ke 7 dan 8 adalah suhu dan kelembapan
7. Baris ke 9-11 adalah penyambungan api thingspeak
8. Baris ke 15 indikasi sukses kirim data ke *thingspeak*
9. “978” adalah data yang masuk ke *thingspeak*
10. “*closing connection*” adalah memutuskan koneksi *wifi*

Setelah pembuatan software selanjutnya akan dilakukan pembuatan alat penunjang untuk proses pengujian alat



Gambar 9 pembuatan box alat

Pada gambar 9 adalah pembuatan lubang pada *box* dengan menggunakan bor dan *cater*



Gambar 10 pemotongan gavalum

Pada gambar 10 adalah proses pemotongan gavalum sebagai penyangga sensor HC-SR04 dan DHT11.



Gambar 11 penyangga sensor

Pada gambar 11 adalah hasil pemotongan gavalum untuk penyangga sensor HC-SR04 dan DHT11.



Gambar 12 ember dan penggaris

Pada gambar 12 adalah bahan untuk pengujian yaitu ember sebagai penampungan air dan penggaris sebagai alat untuk mengukur ketinggian air.

Selanjutnya adalah proses pengujian alat



Gambar 13 pengisian air

Langkah pertama yaitu pengisian air dapat dilihat pada gambar 13 yaitu sebagai ketinggian air yang akan diuji.



Gambar 14 pengecekan ketinggian air

Setelah melakukan pengisian air akan dilakukan pengukuran ketinggian air kolam ikan dengan penggaris pada gambar 14.



Gambar 15 cek LCD

Selanjutnya pengecekan ketinggian air yang berhasil di deteksi alat, apakah ketinggian air yang muncul pada alat sama dengan ketinggian air sesungguhnya. bisa dilihat di gambar 15.



Gambar 16 pengecekan pada web

Selanjutnya setelah pengecekan ketinggian air pada LCD alat dan ketinggian sesungguhnya yaitu pengecekan pada Web *Thingspeak* apakah alat berhasil kirim data ke Web. Dan bisa dilihat pada gambar 16.

Pengujian alat dimulai dari pengisian ember dengan air. Lalu pengukuran tinggi air menggunakan penggaris, lalu pengecekan data apakah sama tinggi air yang diukur dengan penggaris dan juga pengukuran yang dilakukan sensor. Terakhir pengecekan data ke *Thingspeak* apakah data masuk atau tidak.

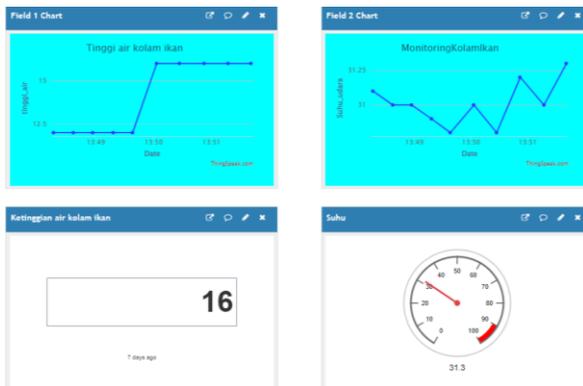
Selanjutnya adalah pengujian alarm dan indikator led



Gambar 17 led dan buzzer on

Pengujian alarm dan *LED*, jadi di pengujian ini sama seperti pengujian alat tetapi air yang diberikan sengaja diberi melebihi batas aman yang telah disetting. Demikian pada gambar 17 dimana indikator lampu *LED* dan alarm yang berupa *Buzzer* menyala sebagai indikator ketinggian air sudah melewati batas aman. Sebagai bukti jika alarm dan *LED* berfungsi dengan baik.

Selanjutnya adalah pengecekan data pada web apakah tersimpan dengan baik atau tidak



Gambar 18 data pada web

Pada gambar 18 adalah tampilan *web* dimana terdapat 2 *field* dan 2 grafik sebagai hasil data yang didapat dan menunjukkan data berhasil disimpan dengan baik pada *web*.

Penjelasan gambar 18:

*Field* 1 adalah grafik yang didapat dari data ketinggian air. Di mana data yang dapat dari sensor *HC-SR04* dikirim oleh *NodeMCU* ke *Thingspeak*.

Pada *field* 2 adalah grafik dari data suhu. Dimana data tersebut didapat dari pembacaan sensor *DHT11* yang dikirim oleh *NodeMCU* ke *Thingspeak*.

Setelah melakukan pengujian tersebut menghasilkan data sebagai berikut:

### Hasil uji coba sensor *HC-SR04*

Tabel 14 data sensor *HC-SR04*

No	Deteksi tinggi air	Tinggi air Sesungguhnya	Data Diterima
1	0	0	Valid
2	0	0	Valid
3	10	10	Valid
4	9	10	Invalid
5	10	10	Valid
6	10	10	Valid
7	14	14	Valid
8	13	14	Invalid
9	14	14	Valid
10	16	16	Valid
11	15	17	Invalid
12	15	15	Valid
13	12	12	Valid
14	14	12	Invalid
15	18	18	Valid
	Rata-Rata		Valid

Dari hasil pada tabel 14 bisa disimpulkan bahwa sensor *HC-SR04* bekerja sesuai ketinggian air sesungguhnya.

Tabel 15 kesimpulan sensor *HC-SR04*

Butir Uji	Aspek yang dilakukan	Hasil yang diharapkan	Hasil Percobaan	Kesimpulan

Pengujian Hardware	Sensor HC- SR04	Sensor pendeteksi ketinggian Air kolam ikan	Sensor mendeteksi ketinggian air	Berhasil
--------------------	-----------------	---	----------------------------------	----------

Pada tabel 15 adalah tabel pengujian sensor *HC-SR04* untuk mendeteksi ketinggian air kolam ikan bekerja sesuai perintah atau tidak.

### Hasil uji coba sensor DHT11

Tabel 16 data sensor *DHT11*

No	Deteksi Suhu	Suhu Sesungguhnya	Hasil
1	30	30	Valid
2	30	30	Valid
3	30	30	Valid
4	31	30	Invalid
5	31	30	Invalid
6	30	30	Valid
7	30	30	Valid
8	31	30	Invalid
9	29	30	Invalid
10	29	30	Invalid
11	30	30	Valid
12	30	30	Valid
13	30	30	Valid
14	30	30	Valid
15	30	30	Valid
Rata-rata			Valid

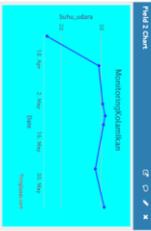
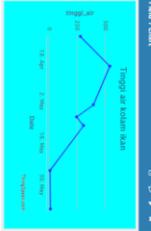
Dari hasil pengujian pada tabel 16 bisa ditarik kesimpulan sensor bekerja dengan baik.

Tabel 17 Kesimpulan Pengujian Sensor DHT

Butir Uji	Aspek yang dilakukan	Hasil yang diharapkan	Hasil Percobaan	Kesimpulan
Pengujian Hardware	Meletakkan sensor DHT11	Sensor mendeteksi suhu dan kelembaban	Sensor mendeteksi suhu dan kelembaban	Berhasil

Pada tabel 17 adalah pengujian alat yaitu DHT11 sebagai sensor suhu bekerja dengan baik.

Tabel 18 pengujian blackbox

Field suhu	Field ketinggian air	Kelas uji
Menampilkan grafik suhu	Menampilkan grafik ketinggian air	Butir uji
Menampilkan grafik data rata-rata per hari	Menampilkan grafik data rata-rata per hari	Skenario
Menampilkan grafik data rata-rata per hari	Menampilkan grafik data rata-rata per hari	Keterangan
		Bukti
Berhasil	Berhasil	Kesimpulan

Pada tabel 18 adalah pengujian *blackbox* dimana alat berhasil kirim data ke *thingspeak* dengan bukti memunculkan *grafik*.

Selanjutnya adalah Metode pengujian yang dilakukan untuk mendapatkan hasil pengujian kali ini yaitu *white box* testing dengan cara mengidentifikasi dan menganalisa script program yang telah dibuat, apakah ada salah atau tidak.

Metode *whitebox* dengan cara melakukan penulisan kode program.

## Pembacaan data

Tabel 18 pembacaan data

Pembacaan ketinggian air dan suhu
// pembacaan ketinggian air
1 : void baca_sensor_hcsr(){
2 :   digitalWrite(triggerPin, LOW);
3 :   delay(2);
4 :   digitalWrite(triggerPin, HIGH);
5 :   delay(10);
6 :   digitalWrite(triggerPin, LOW);
7 :   dur = pulseIn(echoPin, HIGH);
8 :   jarak = (dur/2) / 29.1;
9 :   tinggiair=batas-jarak;
//menampilkan tinggi air ke serial monitor
10 :   Serial.print("tinggi air : ");
11 :   Serial.print(tinggiair);
12 :   Serial.print("cm");
13 :   Serial.println("");
//menampilkan tinggi air ke lcd
14 :   lcd.setCursor(0,0);
15 :   lcd.print("Tinggi air:");
16 :   lcd.print(tinggiair);
17 :   lcd.print("cm");
// kondisi untuk alarm hidup atau mati
18 :   if(tinggiair>=peringatan1){
19 :     digitalWrite(pinBuzzer, HIGH);
20 :     digitalWrite(pinLed, HIGH);
21 :     delay(2000);
22 :   } else if(tinggiair<=peringatan2){
23 :     digitalWrite(pinBuzzer, HIGH);
24 :     digitalWrite(pinLed, HIGH);
25 :     delay(2000);
26 :   } else{
27 :     digitalWrite(pinBuzzer, LOW);
28 :     digitalWrite(pinLed, LOW); } }
//pembaca suhu dan kelembapan
29 :   baca_sensor_dht11(){
30 :     h = dht.readHumidity(); // untuk mendeteksi kelembapan
31 :     t = dht.readTemperature();
32 :     Serial.print("kelembapan udara: ");
33 :     Serial.print(h);
34 :     Serial.print("\n");
35 :     Serial.print("Suhu udara: ");
36 :     Serial.print(t);

## Pembacaan ketinggian air dan suhu

```

37 :   Serial.print("°C ");
38 :   Serial.print("\n");
      //menampilkan suhu ke lcd i2c
39 :   lcd.setCursor(0,1);
40 :   lcd.print("Suhu : ");
41 :   lcd.print(t);
42 :   lcd.print(" C ");}

```

Pada tabel 19 adalah proses pembacaan ketinggian air dengan *HC-SR04* dan suhu udara pada *DHT11* yang akan di cetak pada *serial monitor* dan *LCD*.

Tabel 19 setting pengiriman data ke thingspeak

Kirim data ke *thingspeak*

```

1 : void kirimdata(){
2 :   WiFiClientSecure client;
3 :   Serial.print("menyambungkan ");
4 :   Serial.println(host);
5 :   if (!client.connect(host, httpsPort)) {
      Serial.println("gagal menyambungkan");
      return;}
      //perbarui tinggi air dan suhu pada thingspeak
6 :   String url =
      "/update?api_key=U5Z4D963JM7ZGFSN
      &field1=";
7 :   url += tinggiair;
8 :   url += "&field2=";
9 :   url += t;
      //menampilkan informasi jika sukses kirim
      data pada serial monitor
10 :  client.print(String("GET ") + url +
      "HTTP/1.1\r\n" +
      "Host: " + host + "\r\n" +
      "User-Agent:
      BuildFailureDetectorESP8266\r\n" +
      "Connection: close\r\n\r\n");
11 :  Serial.println("kirim permintaan");
12 :  while (client.connected()) {
      String line = client.readStringUntil('\n');
      if (line == "\r") {Serial.println("request");
      break;}}
13 :  String line = client.readString();
14 :  If (line.toInt() > 0){
      Serial.print("kirim data berhasil!");
      delay(1000);} else {
      Serial.println("kirim data gagal")}
15 :  Serial.println("-----");
16 :  Serial.println(line);
17 :  Serial.println("-----");
      Serial.println("selesai tutup koneksi");}

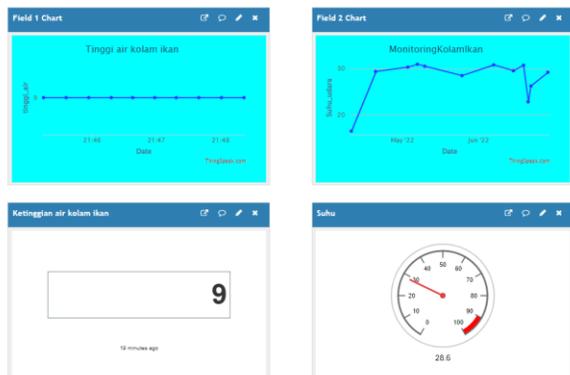
```

Pada tabel 20 diatas adalah pembacaan sensor dan pengiriman data ke *thingspeak*

```
tinggi air : 0 cm
Kelembapan: 89.00
Suhu: 29.50°C
```

Gambar 19 Output pada serial monitor

Pada gambar 19 dimana ketinggian air, kelembapan, dan suhu berhasil muncul ke *serial monitor* dapat disimpulkan bahwa data telah berhasil dikirim ke *thingspeak*.



Gambar 20 Tampilan data pada thingspeak

Pada gambar 20 adalah tampilan data yang telah di kirim ke *thingspeak* yaitu berupa grafik

#### IV. PENUTUP

Berdasarkan hasil penelitian dan pembuatan sistem monitoring dan peringatan ketinggian air kolam ikan dengan sensor HC-SR04 dan disimpulkan dengan adanya sistem monitoring dan peringatan ketinggian air kolam ikan dapat membantu peternak ikan memantau ketinggian air kolam ikan, yang dahulu harus cek langsung ke kolam sekarang bisa memantau lewat *web browser*, dengan adanya sistem monitoring dan peringatan ketinggian air kolam ikan peternak ikan bisa mendapatkan notifikasi *Twitter* jika ketinggian air kolam ikan melewati batas aman. Dan saran bagi pengembang sistem monitoring agar menggunakan koneksi internet yang stabil agar alat bekerja secara maksimal dan juga disarankan menggunakan *handpone* yang *on data* 24 jam agar siap menerima notifikasi *Twitter* jika ketinggian air kolam ikan melewati batas aman.

#### DAFTAR PUSTAKA

- [1] Ahmad , T., Agustina, N., & Amaliyah, A. N. (2020). *PENGENALAN DAN PENGETESAN KOMPONEN ELEKTRONIKA*.
- [2] Arifin, R. D. (2020, 9 19). <https://dianisa.com>. Retrieved from Pengertian Twitter | Sejarah, Fitur, Manfaat: <https://dianisa.com/pengertian-twitter/>
- [3] Dharma, I. L., & Tansa, S. (2019). *Jurnal Teknik. Perancangan Alat Pengendali Pintu Air Sawah Otomatis dengan SIM800L*.
- [4] Kho, D. (2021, 11 4). *Teknik Elektronika*. Retrieved from Pengertian LED (Light Emitting Diode) dan Cara Kerjanya: <https://teknikelektronika.com/pengertian-led-light-emitting-diode-cara-kerja/>
- [5] Laura, A. S. (2017). *SISTEM PENGONTROLAN SUHU DAN KELEMBABAN PADA .*
- [6] Michael, D. (2019). *RANCANG BANGUN PROTOTYPE MONITORING KAPASITAS AIR. IKRA-ITH Informatika, 2580-4316.*
- [7] perius, y. (2021, 11 5). *duniapengertian.com*. Retrieved from Pengertian Kolam: <https://www.duniapengertian.com/2015/11/pengertian-kolam.html>
- [8] Putra, S. F., & Ridwan, A. S. (2017). *JURNAL EINSTEIN. IMPLEMENTASI SENSOR ULTRASONIK HC-SR04 SEBAGAI SENSOR.*
- [9] Razor, A. (2020, 4). *Kabel Jumper Arduino: Pengertian, Fungsi, Jenis, dan Harga*. Retrieved from [AldyRazor.com: https://www.aldyrazor.com/2020/04/kabel-jumper-arduino.html](https://www.aldyrazor.com/2020/04/kabel-jumper-arduino.html)
- [10] Riauwaty, M., & Syawal , H. (2019). *Gambaran Darah Ikan Nila (Oreochromis niloticus) di Kolam . JURNAL PERIKANAN DAN KELAUTAN.*
- [11] Saputra, D. A., Amarudin,, Utami, N., & Setiawan, R. (2020). *RANCANG BANGUN ALAT PEMBERI PAKAN IKAN MENGGUNAKAN MIKROKONTROLER.*
- [12] Setiawan, A. T. (2019). *RANCANG BANGUN BEL SEKOLAH OTOMATIS MENGGUNAKAN MIKROKONTROLER ARDUINO MEGA DENGAN ANTARMUKA*

- BERBASIS WEB. *Seminar Nasional Inovasi Teknologi*.
- [13] Sumadikarta, I., & Isro'I, M. M. (2020). Jurnal Ilmiah Fakultas Teknik LIMIT'S. *Perancangan Smarhome Berbasis Arduino Nodemcu Esp8266*.
- [14] Syaifudin, M., & Akbar, M. (2021). Rancang Bangun Monitoring Sirkulasi Air pada Kolam Ikan Nila Berbasis Arduino. *Jurnal Nasional Informatika dan*.
- [15] D. Michael and D. Gustina, "Rancang Bangun Prototype Monitoring Kapasitas Air Pada Kolam Ikan Secara Otomatis Dengan Menggunakan Mikrokontroler Arduino," *J. IKRA-ITH Inform*, vol. 3, no. 2580-4316, p. 2, 2019.
- [16] Sriani, "Pemanfaatan Sistem Pengendali Water Level Control Untuk Budidaya Ikan Gurame Pada Kolam Terpal Menggunakan Logika Fuzzy Berbasis Mikrokontroler," *Elkawnie*, vol. 3, no. 10.22373/ekw.v5i1.3766, p. 1, 2019.
- [17] Purwangingtyas, "Sistem Water Level Control Untuk Budidaya Ikan Gurame Menggunakan Logika Fuzzy Berbasis Mikrokontroler," *pp*, vol. 6341, p. 48–57, 2019.
- [18] Bandong and K. , "Rancang Bangun Sistem Kontrol Suhu Dan Ketinggian Air Untuk Pemijahan Ikan Dengan Menggunakan Logika Fuzzy Design and Construct of the Temperature and Water Levels Controller for Spawning Fish by Using the Fuzzy Logic Control," *JdC*, vol. 4, no. 3–11, p. 2, 2015.
- [18] Satyani and P. , "Penggunaan Berbagai Wadah Untuk Pembudidayaan Ikan Hias Air Tawar," *Media Akuakultur*, vol. 7, no. 10.15578/ma.7.1.2012.14-19, p. 1, 2012.
- [19] Kadir and I. , "Mobile Iot ( Internet of Things ) Untuk Pemantauan Kualitas Air Habitat Ikan Hias Pada Akuarium Menggunakan Metode Logika," *pp*, vol. 3, no. 298–305, p. 1, 2019.
- [20] Yanuar, "Perbedaan Suhu Air Dalam Akuarium Pemeliharaan Terhadap Laju Pertumbuhan Benih Ikan Nila (*Oreochromis niloticus*)," *Juristek*, vol. 5, no. 152-158, p. 1, 2016.

*[Halaman ini dibiarkan kosong]*