Jurnal Ilmiah Teknik Elektro UML (JITRO-UML)
Vol. 06, No. 01, Hlm. 20 - 25, 2023
e-ISSN 2746-1750

Analisa Mesin Penyiram Tanaman Jamur Menggunakam Arduino ATmega328 Berbasis Internet dan Android

Agus Santoso[1], Syamsul Rois[2], Ancolo[3],

[123] Prodi Teknik Elektro, Fakultas Teknik
Universitas Muhammadiyah Lampung
Jl. H. Zainal Abidin Pagar Alam No.14 Bandar Lampung 35142
Email : Samsulrois32@gmail.com

Abstrak

Watering mushroom plants manually or with human power is currently not effective. This is a problem for mushroom farmers, because every time the temperature and humidity in the mushroom cultivation room increases, that's when the farmers have to water their mushroom plants. An alternative that can be an option to overcome this problem is to use a mushroom plant sprinkler remotely. The design of the tool used in this study is a remote mushroom sprinkler machine that uses Arduino as a programming tool and is combined with DHT11 temperature and humidity sensors and uses NodeMCU ESP8266 as an internet network connection. Based on the results of this research analysis, the machine can be turned on remotely via an Android cellphone. As well as getting the error accuracy results between the Hunidity and Temperature Sensor (DHT 11) temperature and humidity sensors with the Thermo Hygrometer Clock (HTC-1) sensor in the form of a percentage.

Keywords: Machine, Arduino, Internet

1. Pendahuluan

Kebutuhan sistem pengendalian jarak jauh semakin meningkat di zaman ini, terutama di kota besar aktifitas setiap masyarakat sangatlah padat dengan berbagai macam pekerjaannya, tentunya memakan waktu dari pagi hingga malam hari. Akibatnya banyak kegiatan yang tertunda, seperti menghidupkan atau mematikan pompa air untuk menyiram tanaman jamur saat pagi dan sore hari. Berdasarkan hal tersebut aka dilakukanlah penelitian tentang analisa mesin penyiram tanaman jamur. Saat ini pengendali mesin berbasis *internet* atau *android* sudah mulai banyak dikembangkan. Dengan memanfaatkan jaringan *internet* mesin dapat diaktifkan atau dinonaktifkan dari jarak jauh. Selain untuk pengendali jarak jauh, *internet* juga dapat memantau mesin yang sedang beroperasi, ketika kita sedang tidak ada di lokasi. Salah satu yang dapat dikendalikan melalui *internet* adalah mesin penyiram tanaman jamur menggunakan *mikrokontroler*. Mesin pengendali penyiram tanaman jamur berbasis *internet* sebagai solusi alternatif baru untuk pengendalian jarak jauh, sistem pengendalian yang dibangun memanfaatkan jaringan *internet* untuk pengiriman instruksi pengendaliannya ke *Arduino*.

2. Tinjauan Pustaka

2.1. *Arduino ATmega328*

Arduino Uno adalah papan sirkuit berbasis *mikrokontroler ATmega328*. *integrated* circuit (*IC*) ini memiliki 14 *input*/*output* digital (6 *output* untuk *Pulse Width Modulation*), 6 analog *input*, *resonator* kristal keramik 16 *MHz*, Koneksi *Universal Serial Bus* (*USB*), soket adaptor, pin *header In Circuit Serial Programming* (*ICSP*), dan tombol *reset*. Hal inilah yang dibutuhkan untuk *mensupport mikrokontrol* secara mudah terhubung dengan kabel *power USB* atau kabel *power supply* adaptor *AC* ke *DC* atau juga *battery*.



Gambar 2.1. Arduino Uno ATmega328

Uno berbeda dari semua *board mikrokontrol* diawal-awal yang tidak menggunakan *chip* khusus *driver FTDI USB-to-serial*. Sebagai penggantinya penerapan *USB-to-serial* adalah *ATmega16U2* versi R2 (versi sebelumnya *ATmega8U2*). Versi *Arduino Uno Rev*.2 dilengkapi *resistor* ke *8U2* ke garis *ground* yang lebih mudah diberikan ke *mode Device Frimware Update* (*DFU*). Bahasa "*UNO*" berasal dari bahasa Italia yang artinya

Satu, ditandai dengan peluncuran pertama *Arduino* 1.0, *Uno* pada versi 1.0 sebagai referensi untuk *Arduino* yang selanjutnya, seri *Uno* versi terbaru dilengkapi *USB*

2.2. *NodeMCU Esp8266*

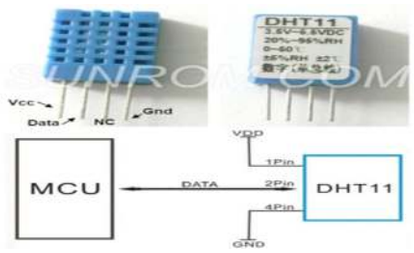
NodeMCU ESP8266 merupakan modul turunan pengembangan dari modul *platform IoT* (*Internet of Things*) keluarga *ESP8266* tipe *ESP-12*. Secara fungsi modul ini hampir menyerupai dengan *platform* modul arduino, tetapi yang membedakan yaitu dikhususkan untuk "*Connected to Internet*".



Gambar 2.2. *Versi* *NodeMCU ESP8266*

2.3. Sensor Suhu Dan Kelembaban

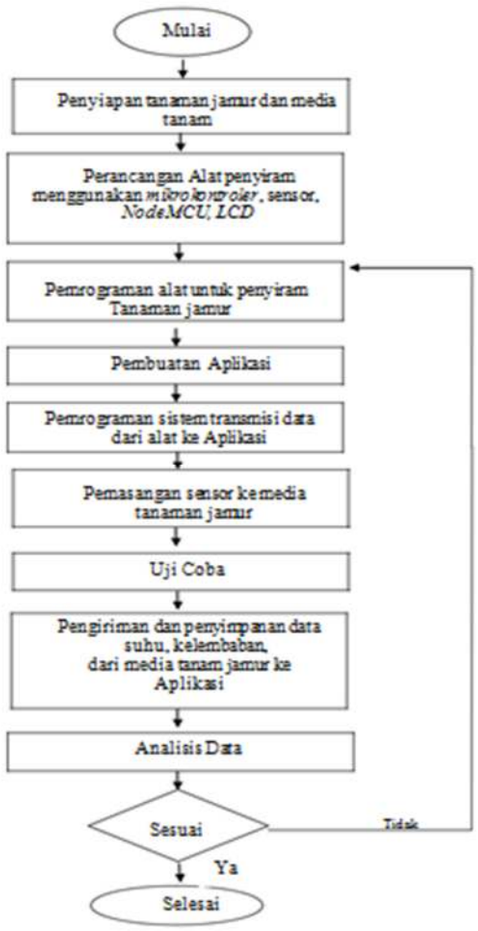
DHT11 adalah sensor digital yang dapat mengukur suhu dan kelembaban udara di sekitarnya. Sensor ini sangat mudah digunakan bersama dengan *Arduino*. Memiliki tingkat *stabilitas* yang sangat baik serta fitur kalibrasi yang sangat akurat. Koefisien kalibrasi disimpan dalam *OTP* program *memory*, sehingga ketika *internal* sensor mendeteksi sesuatu, maka *module* ini menyertakan koefisien tersebut dalam kalkulasinya, *DHT11* ini termasuk sensor yang memiliki kualitas terbaik, dinilai dari respon, pembacaan data yang cepat, dan kemampuan anti-*interference*. Ukurannya yang kecil, dan dengan transmisi sinyal hingga 20 meter.



Gambar 2.3. Sensor Suhu dan Kelembaban *DHT 11*

3. Metodologi Penelitian

Penelitian ini dimulai dengan tahap menyiapkan tanaman jamur, pemrograman alat, pengiriman data suhu, kelembaban. Status aktivasi pompa *on/off* hingga pengambilan data yang ditampilkan di *handphone* dan dilanjutkan dengan analisis data. Prosedur penelitian dapat dilihat pada diagram diatas.



Gambar 3. Diagram Alir Metodologi

4. Pembahasan

4.1. Tujuan Pengujian

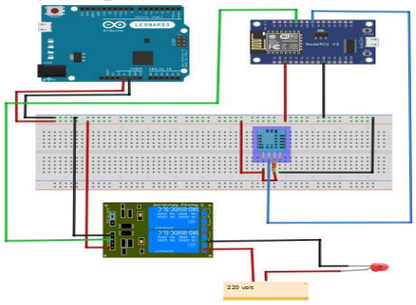
Pengujian ini dilakukan untuk mengetahui apakah *NodeMCU ESP*8266 dapat menerima sinyal *wifi dan* dapat menerima sinyal perintah yang dikirim dari *handphone android*.

4.2. Alat Penguji

- *NodeMCU ESSP*8266
- *Handpone android*
- *Arduino UNO Atmega* 328
- *DHT11*

4.3. Langkah Pengujian

- Rangkai *NodeMCU ESSP*8266 pada *Arduino UNO ATmega* 328 seperti pada gambar di bawah ini.



Gambar 4.1 Rangkaian *NodeMCU ESP*8266

- Memberikan *sketch* program pada papan *NodeMCU ESP8266*.

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include
<BlynkSimpleEsp8266.h>
#include <DHT.h>
#include <SimpleTimer.h>
#define DHTPIN 5
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

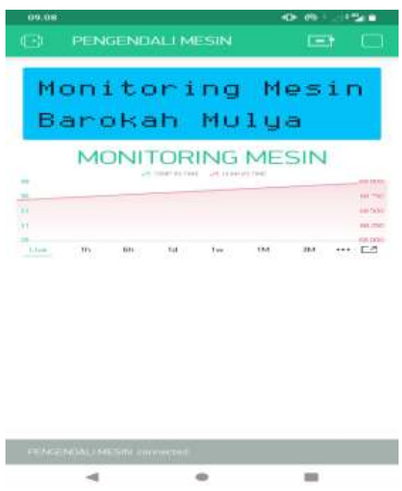
SimpleTimer timer; float humi,
temp;
charauth[]
="AkI7G63_wdRC3RoTRYp7gAF6r
vPBozfc";
char ssid[] = "Rois";
char pass[] = "1234567890";
char outstr[15];
WidgetLCD lcd(V0);

void setup()
{
Serial.begin(115200);
delay(10);
Blynk.begin(auth, ssid, pass);
lcd.clear();
lcd.print(0, 0, "Monitoring Mesin");
lcd.print(0, 1, "Barokah Mulya");
delay(2000);
lcd.clear();
lcd.print(0, 0, "Temp=  C");
lcd.print(0, 1, "Humi=  %RH");
timer.setInterval(1000,
sendData);
}
void sendData()
{

humi = dht.readHumidity();
temp = dht.readTemperature();
if (isnan(humi) || isnan(temp)) {
Serial.println("DHT11 tidak
terbaca... !");
return;
}
else{ Serial.println(temp);
Serial.println(humi);
Blynk.virtualWrite(1, temp);
Blynk.virtualWrite(2, humi);
dtostrf(temp,4, 2, outstr);
lcd.print(5, 0, outstr);
dtostrf(humi,4, 2, outstr);
lcd.print(5, 1, outstr);
}
}
void loop()
{
Blynk.run();
timer.run();
}
```

4.4. Hasil Pengujian

Setelah di lakukan pemograman pada *board NodeMcu Esp8266* di *arduino IDE*, lalu kita buka aplikasi *blynk* pad *handphone android*, maka hasilnya akan seperti gambar di bawah ini :



Gambar 4.2 Tampilan Awal Pada *Blynk* Setelah di Program



Gambar 4.3 Tampilan Pada *Blynk* Saat Pompa Mati



Gambar 4.4 Tampilan Pada *Blynk* Saat Pompa Hidup

4.5. Hasil perbandingan suhu dan kelembaban.

Setelah melakukan langkah – langkah seperti di atas, terlihat bahwa fungsi dari alat ini berfungsi untuk mengetahui suhu dan kelembaban dan menampilkannya pada *LCD*. Begitu juga yang terlihat pada *HTC*-1 Akan tetapi ada perbedaan suhu dan kelembaban yang tertera di LCD dan HTC-1. Perbedaan ini yang akan dihitung apakah sesuai dengan akurasi sensor *DHT* 11 atau tidak. Data di ambil pada siang hari selama lima hari berturut-turut.

Tabel 4.1. Pengambilan Data Suhu dan Kelembaban

No	Hari Pengambilan Data	Terbaca Pada HTC-1	Terbaca Pada DHT11
1	Pertama	T, 27.6°C, H, 76%	T, 31°C H, 67%
2	Kedua	T, 27.3°C H, 74%	T, 29°C H, 77%
3	Ketiga	T, 27.4°C H, 74%	T, 31°C H, 67%
4	Keempat	T, 27.9°C H, 72%	T, 31°C H, 66%
5	Kelima	T, 28,3°C H, 71%	T, 31°C H, 67%

4.6. Analisa Perbandingan

Dari data hasil perbandingan di atas maka dapat dihitung persentase (%) kesalahan pengukuran suhu dan kelembaban dengan perhitungan sebagai berikut :

- Kesalahan (%) hasil dari pembacaan sensor *HTC*-1 dan *DHT* 11 pada hari pertama :

a. Suhu

$$= \frac{(HTC1\ suhu) - DHT11\ suhu}{(HTC - 1\ suhu)} \times 100$$

$$= \frac{27{,}6 - 31}{27{,}6} \times 100\%$$

$$= 10{,}9\ \%$$

b. Kelembaban

$$= \frac{(HTC1\ lmb) - DHT11\ lmb}{(HTC - 1\ lmb)} \times 100\%$$

$$= \frac{76 - 67}{76} \times 100\%$$

$$= 11{,}8\ \%$$

- Kesalahan (%) hasil dari pembacaan sensor *HTC*-1 dan *DHT* 11 pada hari kedua :

a. Suhu

$$= \frac{(HTC1\ suhu) - DHT11\ suhu}{(HTC - 1\ suhu)} \times 100\%$$

$$= \frac{27{,}3 - 29}{27{,}3} \times 100\%$$

$$= 5.8\ \%$$

b. Kelembaban

$$= \frac{(HTC1\ lmb) - DHT11\ lmb}{(HTC - 1\ lmb)} \times 100\%$$

$$= \frac{74 - 77}{74} \times 100\%$$

$$= 4\ \%$$

- Kesalahan (%) hasil dari pembacaan sensor

HTC-1 dan *DHT* 11 pada hari ketiga :

a. Suhu

$$= \frac{(HTC1\ suhu) - DHT11\ suhu}{(HTC - 1\ suhu)} \times 100\%$$

$$= \frac{27{,}4 - 31}{27{,}4} \times 100\%$$

$$= 13.1\ \%$$

b. Kelembaban

$$= \frac{(HTC1\ lmb) - DHT11\ lmb}{(HTC - 1\ lmb)} \times 100\%$$

$$= \frac{74 - 67}{74} \times 100\%$$

$$= 9.4\ \%$$

❖ Kesalahan (%) hasil dari pembacaan sensor *HTC*-1 dan *DHT* 11 pada hari keempat :

a. Suhu

$$= \frac{(HTC1\ suhu) - DHT11\ suhu}{(HTC - 1\ suhu)} \times 100\%$$

$$= \frac{27{,}9 - 31}{27{,}9} \times 100\%$$

$$= 11{,}1\ \%$$

b. Kelembaban

$$= \frac{(HTC1\ lmb) - DHT11\ lmb}{(HTC - 1\ lmb)} \times 100\%$$

$$= \frac{72 - 66}{72} \times 100\%$$

$$= 8{,}3\ \%$$

❖ Kesalahan (%) hasil dari pembacaan sensor *HTC*-1 dan *DHT* 11 pada hari kelima :

a. Suhu

$$= \frac{(HTC1\ suhu) - DHT11\ suhu}{(HTC - 1\ suhu)} \times 100\%$$

$$= \frac{28{,}3 - 31}{28{,}3} \times 100\%$$

$$= 9{,}5\ \%$$

b. Kelembaban

$$= \frac{(HTC - 1\ lmb) - DHT11\ lmb}{(HTC - 1\ lmb)} \times 100\%$$

$$= \frac{71 - 67}{71} \times 100\%$$

$$= 5{,}6\ \%$$

5. Kesimpulan

Setelah melakukan analisa pada mesin peyiram tanaman jamur, berikut adalah kesimpulan yang didapat serta diusulkan saran dan untuk kedepannya diharapkan adanya pengembangan dari mesin penyiram tanaman jamur ini.

5.1. Kesimpulan

- Perbandingan antara sensor suhu dan kelembaban *HTC*-1 dengan *DHT* 11 sangatlah tidak sesuai dengan rentang pengukuran kesalahan pada *DHT* 11, yaitu untuk untuk suhu sebesar ± 2% dan kelembaban sebesar ± 5%.
- *Prototype* dapat mengirim suhu dan kelembaban serta melakukan *on/off* pada mesin penyiram tanaman jamur melalui *handphone android* dari jarak jauh selama *prototype* masih mendapatkan sinyal *internet*.

5.2. Saran

- Untuk mengatasi kegagalan sistem kerja pada pengendali mesin penyiram tanaman jamur yang diakibatkan oleh buruknya jaringan *internet*, maka disarankan dipasang juga sensor suhu dan kelembaban *DHT11* yang langsung terhubung pada mesin pompa air, hal ini bertujuan supaya mesin tetap dapat melakukan penyiraman meskipun tidak di kendalikan diri jarak jauh dan tetap dapat dimonitor dari *android* dalam bentuk angka serta grafik suh dan kelembaban.
- Adanya pengembangan pada alat ini di masa mendatang. Hal ini bertujuan untuk lebih memudahkan petani melakukan pengontrolan maupun penyiraman dari jarak jauh.

6. Daftar Pustaka

Andrianto, H. dan Darmawan, A. 2016. *Arduino Belajar Cepat dan Pemrograman.* Penerbit Informatika : Bandung.

Djuandi, F. 2011. *Pengenalan Arduino*. Penerbit Elexmedia: Jakarta.

Gunawan, A.W. 2000. Usaha Pembibitan Jamur. Jakarta: Penebar Swadaya

Romdloni, M.S., Haryanto., Alfita, R., Nahari, R.V. 2017. Prototype Sistem Monitoring dan Pengendalian Pintu Air Otomatis Sebagai Peringatan Dini Bahaya Banjir Berbasis Internet of Things. *Seminar Nasional Matematika dan Aplikasinya*. 21 Oktober 2017. Universitas Erlangga. Surabaya.

Syahwil, M., 2013. Panduan Mudah Simulasi dan Praktik Mikrokontroler Arduino. Penerbit Andi Yogyakarta.

Yahwe, P.C., Isnawaty., Aksara, F.M. 2016. Rancang Bangun *Prototype System* Monitoring Kelembaban Tanah Melalui Sms Berdasarkan Hasil Penyiraman Tanaman Studi Kasus Tanaman Cabai Dan Tomat. Vol. 2(1) : 97-110.