

BAB IV

HASIL DAN PEMBAHASAN

4.1 Cara Penggunaan Komponen

Penggunaan alat ini mengendalikan sebuah komponen yang berbagai macam dan untuk komponen seperti Arduino Uno, Ethernet, Display LCD, Power Supply, serta Pompa Air yang sudah di jelaskan dalam Bab sebelumnya. karena itu komponen ini membantu untuk perancangan tugas akhir untuk perancangan alat pengairan lahan otomatis berbasis IoT yang akan dibuat bagi lahan yang kekeringan.

1. Komponen Relay yang dimana Relay digunakan untuk menggerakkan suatu peranti yang bekerja berdasarkan elektromagnetik untuk menggerakkan sejumlah kontaktor yang tersusun atau sebuah saklar elektronis yang dapat dikendalikan dari rangkaian elektronik lainnya dengan memanfaatkan tenaga listrik sebagai sumber energinya yang berjalan pada saat ini sesuai waktu pada Komputer. Relay ini membantu juga karena untuk waktu sudah sesuai serta dihubungkan pada Program yang dimana berfungsi sebagai penyimpan data pengairan lahan yang kering. Disitulah Relay akan bekerja untuk waktu yang sudah ditentukan dalam Program di Komputer.
2. Komponen Ethernet dalam rancangan ini akan dipergunakan sebagai penghubung ke Komputer bila waktu pengairan lahan yang kekeringan akan dilakukan/dimulai akan lebih mudah untuk mengingat waktu pengairan setiap hari.
3. Komponen LCD dalam ini lcd akan memunculkan tgl,bulan,tahun,waktu serta nilai kelembapan tanah sudah melakukan berapa kali.
4. Komponen Hardware mikrokontroler Arduino diprogram menggunakan bahasa pemrograman wiring-based yang berbasiskan syntax dan library. Pemrograman wiring-based ini tidak berbeda dengan C/C++, tetapi dengan beberapa penyederhanaan dan modifikasi untuk memudahkan dalam pengembangan aplikasinya, mikrokontroler Arduino juga menggunakan Integerated Development Environment (IDE) berbasis processing.
5. Sensor ini menggunakan dua probe yang terbuat dari tembaga untuk melewatkan arus yang melalui tanah dan kemudian membaca resistansi untuk mendapatkan tingkat kelembaban, lebih banyak air (basah) membuat tanah menghantarkan listrik lebih mudah (resistansi berkurang), sedangkan tanah kering (tidak basah) maka sangat sangat buruk untuk menghantarkan listrik (resistansi meningkat).

Cara penggunaan pada rangkaian yang sudah dijelaskan akan menghasilkan pengairan lahan yang kekeringan sesuai pembaca sensor yang diprogram serta pompa air akan bergerak secara otomatis dengan bantuan relay sebagai

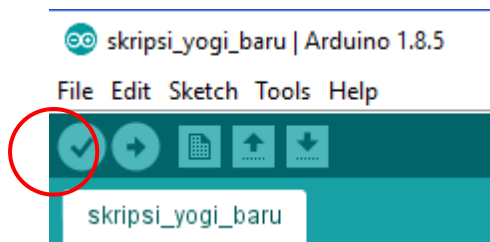
penggerak/peranti dan hasil nilai kelembapan tanah. Pada setiap pengairan akan diperingatkan nilai kelembapan tanah, tanggal serta berapa kali untuk melakukan pengairan.

4.2 Cara Penggunaan Arduino IDE

Aplikasi Arduino IDE sendiri memiliki banyak fitur yang bahkan sudah ada library sesuai komponen yang kita buat pada Tugas Akhir ini. Komponen di Arduino uno bermacam – macam dan juga Example nya, Arduino Uno memiliki macam-macam Board yang akan kita gunakan sehubungan saya menggunakan Arduino Uno pada Tugas Akhir ini saya akan menghubungkan Board ke jenis yang sama yaitu Arduino Uno.

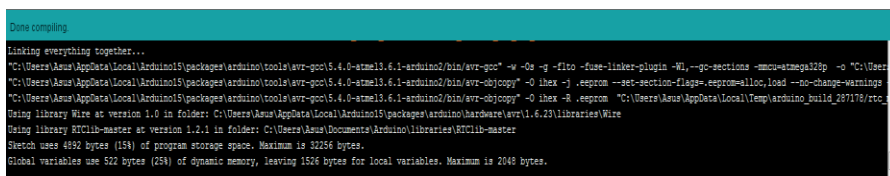
4.2.1 Penggunaan saat Menguji maupun Menggunakan Pemrograman Aplikasi Arduino Uno

Untuk menguji maupun pemrograman arduino uno klik pada tombol centang/Verify tunggu hingga sukses seperti gambar dibawah ini :



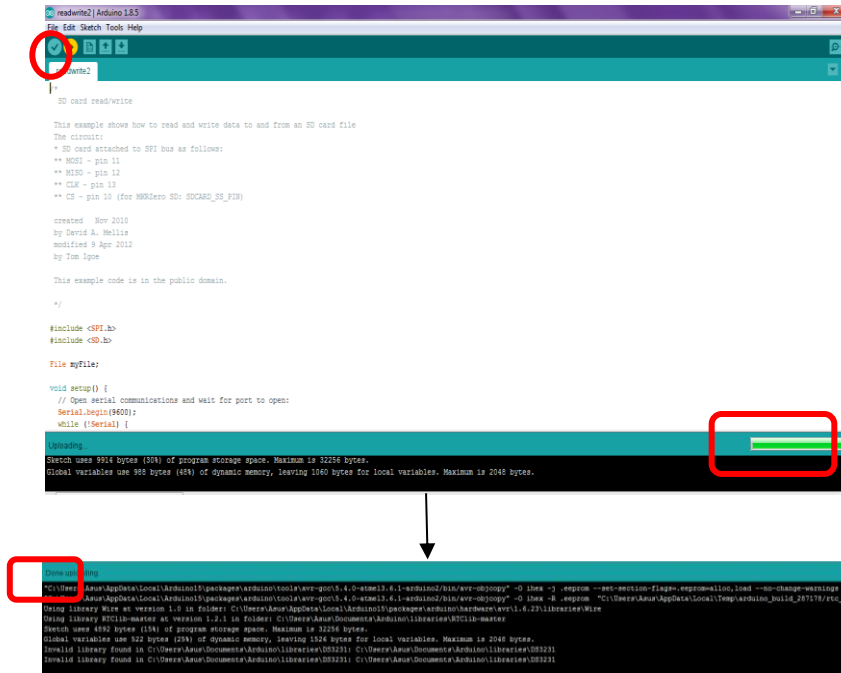
Gambar 4.1 Verify Arduino IDE

Bila sudah klik pada verify dan berhasil tanpa ada peringatan/ eror akan muncul tulisan *Done Compiling* seperti dibawah ini :



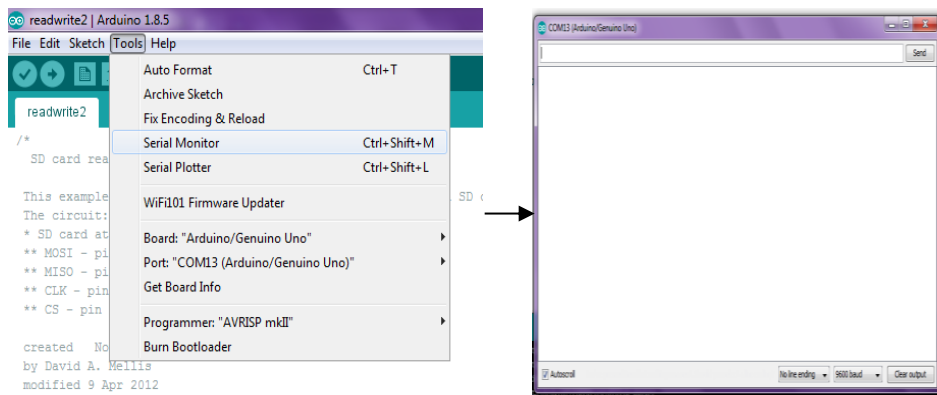
Gambar 4.2 Verify Arduino “Done Compiling”

Lanjut ke Upload klik tombol panah/Upload tunggu hingga upload sukses. Bila Upload sukses atau berhasil codingan yang kita buat berjalan dan akan muncul tulisan *Done Uploading* Seperti Gambar dibawah ini :



Gambar 4.3 Upload Arduino

Setelah Upload sukses bila ingin memunculkan hasil yang kita coding tadi di Arduino IDE klik Tools - Serial Monitor.



Gambar 4.4 Tools - Serial Monitor Arduino

4.3 Uji Permasalahan Pembuatan Alat

Bagaimana kalau kelembapan tanah melebihi dari standart (apa yang dilakukan alat):

- Jika kelembapan tanah melebihi standart atau 96%-300% LCD akan terbaca error atau sama dengan tulisan WARNING!!
- Untuk kelembapan tanah minimal 65.80% tanah dalam keadaan lembab. jika nilai kelembapan (Rh) dibawah 28.61 keadaan tanah mulai kering otomatis pompa air akan aktif. Seperti yang sudah dijelaskan untuk kelembapan tanah

yang artinya jika nilai RH kurang dari 29% maka tanah akan membutuhkan air dan tanah tersebut kekurangan air. mengetahui kelembapan tanah dengan nilai RH yang diartikan kelembapan sudah mulai kering. Serta pada nilai kelembapan 35% yang diartikan tanah mulai kering seperti yang sudah dijelaskan, Batas maksimal kelembapan tanah mulai dari 65.80% 95.63%.

Apa saja untuk menaikkan kelembapan tanah:

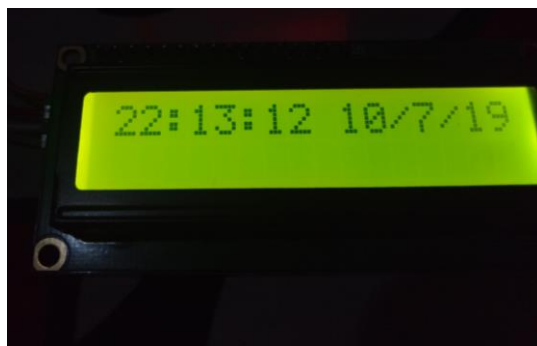
- Mempertahankan kelembapan tanah, karena kelembapan tanah merupakan factor penting bagi peningkatan penyerapan unsur hara
- Mengurangi evaporasi yang berlebihan, sehingga air tanah tidak banyak yang hilang
- Memperbaiki sifat fisik tanah, memperbaiki aerasi tanah sehingga akar dapat berkembang dengan baik dan pertumbuhan tanaman akan lebih subur
- Mengendalikan pertumbuhan gulma.

4.4 Uji Coba Alat/Komponen

Pada Bab 4 ini akan menguji semua komponen yang digunakan pada rancangan tugas akhir. Setiap komponen seperti Module soil moisture sensor, lcd, power supply, relay serta menggerakkan Pompa air seperti halnya akan mengakses Input – Proses – hingga Output.

4.4.1 Pengujian Lcd 16x2

Pengujian seperti gambar dibawah ini adalah uji coba lcd pada komponen ini pin – pin yang di lcd dihubungkan ke pin sensor shield versi 5.



Gambar 4.5 Pengujian lcd

• Source Code lcd

Pada Gambar dibawah ini adalah source code lcd untuk menjalankan.

```
#include Wire.h  
#include <LiquidCrystal_I2C.h>
```

```
skripsi_yogi_baru | Arduino 1.8.5
File Edit Sketch Tools Help

skripsi_yogi_baru

int sensorPin = A0;
int powerPin = 6 ;
#include <Wire.h>
#include <RTClib.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
RTC_DS1307 rtc;
char namaHari[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jumat", "Sabtu"};

void setup() {
  lcd.begin(16, 2);
  pinMode (powerPin, OUTPUT);
  digitalWrite ( powerPin,LOW);
  Serial.begin (9600);
  Serial.begin(9600);
  if (! rtc.begin()) {
    Serial.println("RTC TIDAK TERBACA");
    while (1);
  }

  if (! rtc.isrunning()) {
    Serial.println("RTC is NOT running!");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); //update rtc dari waktu komputer
  }
}

void loop() {
  [ DateTime now = rtc.now();
  Serial.print(namaHari[now.dayOfTheWeek()]);

  Serial.print(', ');
  Serial.print(now.day(), DEC);
  Serial.print('/');
  Serial.print(now.month(), DEC);
  Serial.print('/');
  Serial.print(now.year(), DEC);
  Serial.print(" ");
  Serial.print(now.hour(), DEC);
  Serial.print(':');
  Serial.print(now.minute(), DEC);
  Serial.print(':');
  Serial.print(now.second(), DEC);
  Serial.println();

  lcd.setCursor(0,0);
  lcd.print("HM");
  lcd.setCursor(4,0);
  lcd.print("=");
  lcd.setCursor(5,0);
  lcd.print(" ");
  lcd.print(now.hour(), DEC);
  lcd.print(':');
  lcd.print(now.minute(), DEC);
  lcd.print(':');
  lcd.print(now.second(), DEC);
  lcd.println();
  delay (100);
  (Serial.print("HM");
```

```
Serial.println(bacaSensor());
lcd.setCursor(0,1);
lcd.print("HUM");
lcd.setCursor(4,1);
  lcd.print("=");
lcd.setCursor(8,1);
lcd.print("%");
lcd.setCursor(5,1);
lcd.print (bacaSensor());

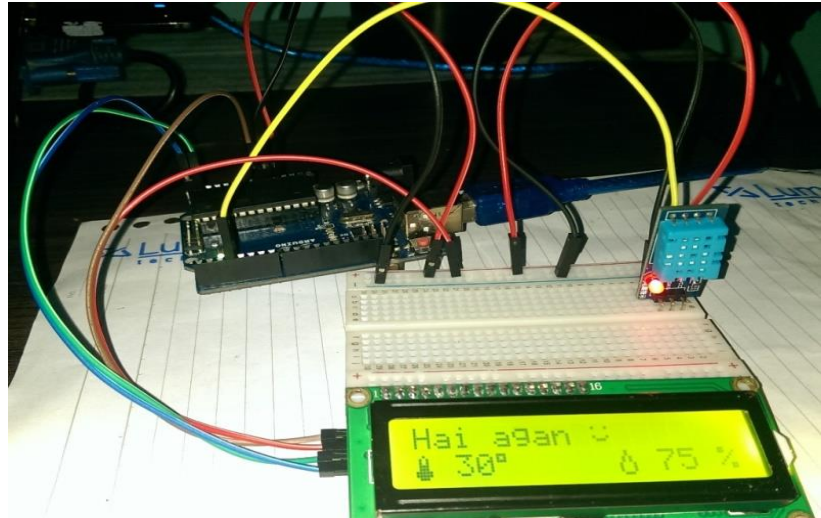
}}

int bacaSensor ()
{
  digitalWrite (powerPin, HIGH);
  delay (500);
  int nilaiSensor = analogRead (sensorPin);
  digitalWrite(powerPin, LOW);
  return 1023 - nilaiSensor;
}
// put your main code here, to run repeatedly;
```

Gambar 4.6 Source menampilkan source code di lcd

- **Serial Monitor lcd/hasil**

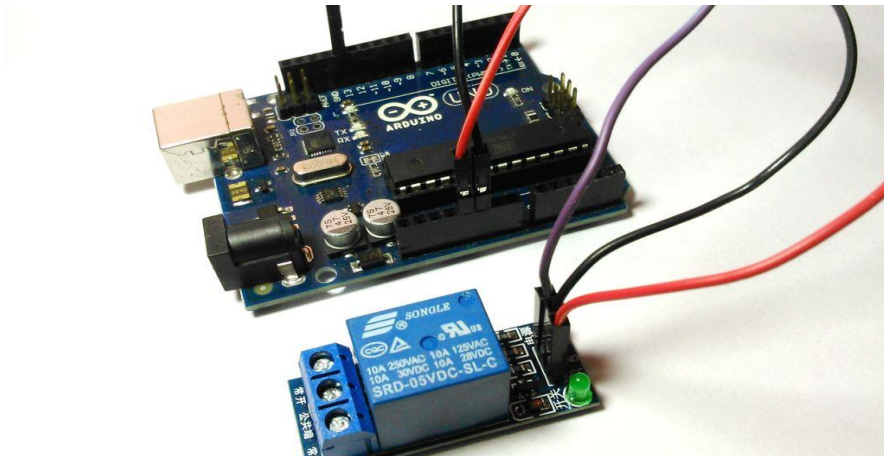
Bila Codingan kita sudah kita verify dan tanpa ada peringatan eror , langsung kita lakukan Upload hingga *done uploading* setelah selesai kita akan melihat hasil lcd apakah sudah muncul perintah yang diinginkan.



Gambar 4.7 Hasil lcd

4.4.2 Pengujian Module RTC(*Real Time Clock*) DS3231

Pengujian seperti gambar dibawah ini adalah uji coba alat atau Komponen module RTC (*real time clock*) pada komponen ini pin – pin yang di module RTC dihubungkan ke pin sensor shield versi 5. Sensor RTC berfungsi menampilkan sesuai waktu hari ini, dan komponen RTC salah satu akan digunakan pada rancangan tugas akhir ini sebagai mengetahui waktu hari ini dan bisa mendeteksi pada scedcule yang sudah ditetapkan pada module microsd card.



Gambar 4.8 Uji Alat Relay

- Pada saat relay kondisi Normally Open (NO) maka saklar atau switch contact akan menghantarkan arus listrik. Tetapi apabila ditemukan kondisi dimana armature kembali ke posisi semula (NC), pada saat itu juga menandakan bahwa module tidak teraliri arus listrik.
NC (Normally Close) : Kondisi awal dimana relai pada posisi tertutup, tetapi saat teraliri arus maka akan ke posisi terbuka
NO (Normally Open) : Merupakan kebalikan dari NC yang dimana kondisi awal relai pada posisi Open, tetapi saat teraliri arus maka akan ke posisi tertutup.
- Source Code Module Ethernet
Source Code Module Ethernet menggunakan ENC28J60. Source Code ini membantu Module Ethernet bekerja saat dijalankan dan dilihat di serial monitor akan muncul waktu saat ini.


```

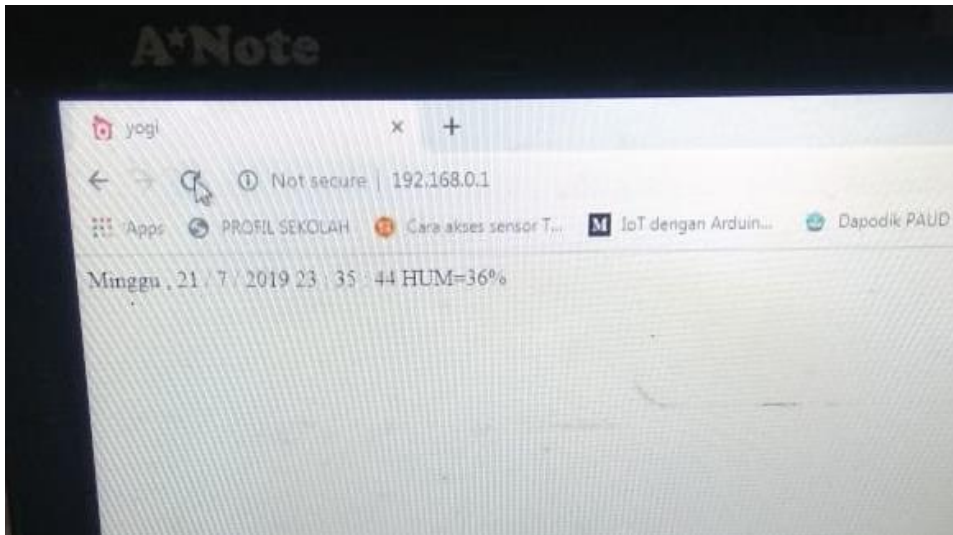
1 #include <SPI.h>
2 #include <UIPEthernet.h>
3
4 byte mac[] = {
5   0xDE, 0xED, 0xBE, 0xEF, 0xFE, 0xED
6 };
7 IPAddress ip(192, 168, 169, 177);
8
9 EthernetServer server(80);
10
11 void setup() {
12   Serial.begin(9600);
13   while (!Serial) {
14     ;
15   }
16
17   Ethernet.begin(mac, ip);
18   server.begin();
19   Serial.print("server is at ");
20   Serial.println(Ethernet.localIP());
21 }
22
23
24 void loop() {
25   EthernetClient client = server.available();
26   if (client) {
27     Serial.println("new client");
28     boolean currentLineIsBlank = true;
29     while (client.connected()) {
30       if (client.available()) {
31         char c = client.read();
32         Serial.write(c);
33         if (c == '\n' && currentLineIsBlank) {
34           client.println("HTTP/1.1 200 OK");
35           client.println("Content-Type: text/html");
36           client.println("Connection: close");
37           client.println("Refresh: 5");
38           client.println();
39           client.println("<!DOCTYPE HTML>");
40           client.println("<html>");
41
42           for (int analogChannel = 0; analogChannel < 6; analogChannel++) {
43             int analogReading = analogRead(analogChannel);
44             client.print("analog input ");
45             client.print(analogChannel);
46             client.print(" is ");
47             client.print(analogReading);
48             client.println("<br />");
49           }
50           client.println("</html>");
51           break;
52         }
53         if (c == '\n') {
54           currentLineIsBlank = true;
55         } else if (c != '\r') {
56           currentLineIsBlank = false;
57         }
58       }
59     }
60     delay(1);
61     client.stop();
62     Serial.println("client disconnected");
63   }
64 }

```

Gambar 4.9 source code connect to Ethernet shield

- **Hasil dari serial monitor**

Pada Hasil serial Monitor ini adalah hasil pada source code connect module Ethernet shield yang dimana hasil ditunjukkan pada waktu saat ini.



Gambar 4.10 Serial Monitor Ethernet ENC28J60

4.4.3 Pengujian Internet of Thing

Pengujian seperti gambar dibawah ini adalah uji coba alat atau Komponen Ethernet Shield pada komponen ini yang dihubungkan ke pin Soil Moisture Sensor. Relay berfungsi Motor penggerak hingga Pompa Air Berjalan dan komponen Relay salah satu akan digunakan pada rancangan tugas akhir ini sebagai penggerak pada siku lengan yang bergerak setiap derajatnya yang dimana akan sangat membantu sekali pada proses alat pengairan lahan otomatis berbasis IoT dan bisa bergerak tanpa bantuan sekeliling lagi.

Tahap ini dilakukan setelah perancangan sistem selesai dilakukan dan selanjutnya diimplementasikan pada bahasa pemrograman. Tujuan implementasi sistem adalah untuk menerapkan perancangan yang telah dilakukan terhadap perangkat lunak sehingga nantinya maksud dan tujuan pembangunan perangkat lunak ini dapat tercapai.

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan dan kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.



Gambar 4.11 Uji alat Code Monitor Kelembapan Tanah

- **Source Code Monitor Kelembapan Tanah**

Source Code Nilai Kelembapan Tanah yang akan dicoding untuk setiap kelembapannya dan diberi delay atau batasan waktu untuk berhenti hingga batas berjalan sesuai dengan program.

pada webservice thingsboard data yang ditampilkan yaitu nilai kelembapan tanah, nilai kelembapan udara, dan suhu lingkungan. Data ditampilkan dalam bentuk diagram, grafik dan juga kondisi secara real time. data dari hasil pembacaan sensor tersebut dapat dilihat dengan media internet pada webservice thingsboard.

Dalam merancang suatu sistem, terlebih dahulu harus menentukan prinsip kerja dari sistem yang dibuat. Untuk itu, penyusunan spesifikasi teknis dari sistem yang dibuat dan blok diagram sangat perlu diperlukan. Tujuannya untuk mempermudah dalam pengimplementasiannya, baik dalam hal perancangan perangkat keras maupun perangkat lunak yaitu berupa laporan data kecepatan di website.

Dalam perancangan ini secara umum dibagi menjadi beberapa rancangan yaitu perangkat keras (Hardware) yang berfungsi untuk mendeteksi kecepatan kendaraan dan menghitung jumlah kendaraan dan rancangan perangkat lunak (Software) website yang berfungsi

untuk mengetahui berupa laporan data kecepatan serta jumlah kendaraan, dari hasil pengukuran perangkat keras yang dikirim menggunakan media ethernet shield. Oleh sebab itu, dibutuhkan kerangka perancangan sebelum melakukan pembuatan sistem baik dari segi perangkat keras (Hardware) maupun perangkat lunak (Software).

menjelaskan bahwa Arduino mendapatkan data yang dikirim oleh sensor ultrasonik untuk mengetahui objek yang dideteksi. Data akan diolah di Arduino untuk mendapatkan hasil kecepatan kendaraan dan dikirim melalui media komunikasi Ethernet shield. Dalam proses ini Ethernet shield memiliki fungsi sebagai media pengiriman data dari arduino yang sudah di proses sebelumnya ke website. Di website data yang diterima akan diolah kembali untuk memvisualisasikan hasil analisa yang sesuai dengan data yang sudah dimiliki.

Dibawah ini adalah beberapa Sourcecode yang dipakai untuk pengairan lahan otomatis berbasis Inthernet Of Thing. Ada beberapa Sourcecode yang akan di tampilkan di LCD jika tata cara dan koding dengan benar, maka LCD akan membaca beberapa sensor yang ada di tanah, untuk kelembapannya, dan akan disambungkan ke Ethernet dengan menggunakan alamat yang disesuaikan dengan RTC Module.

➤ **Sourcecode RTC Module**

```
// Date and time functions using a RTC connected via I2C and Wire lib
#include <Wire.h>
#include "RTCLib.h"

RTC_DS3231 rtc;

char daysOfTheWeek[7][12] = {"Minggu", "Senin", "Selasa", "Rabu",
"Kamis", "Jum'at", "Sabtu"};

void setup () {

#ifndef ESP8266
  while (!Serial);
#endif

  Serial.begin(9600);

  delay(3000); // wait for console opening

  if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
    while (1);
  }
}
```

```

}

if (rtc.lostPower()) {
  Serial.println("RTC lost power, lets set the time!");
  // following line sets the RTC to the date & time this sketch was compiled
  rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  // This line sets the RTC with an explicit date & time, for example to set
  // Mei 1, 2019 at 10am you would call:
  // rtc.adjust(DateTime(2019, 5, 1, 10, 0, 0));
}

void loop () {
  DateTime now = rtc.now();

  Serial.print(now.year(), DEC);
  Serial.print('/');
  Serial.print(now.month(), DEC);
  Serial.print('/');
  Serial.print(now.day(), DEC);
  Serial.print(" (");
  Serial.print(daysOfTheWeek[now.dayOfTheWeek()]);
  Serial.print(") ");
  Serial.print(now.hour(), DEC);
  Serial.print(':');
  Serial.print(now.minute(), DEC);
  Serial.print(':');
  Serial.print(now.second(), DEC);
  Serial.println();
  Serial.println();
  delay(3000);
}

```

➤ Sourcecode Relay/Gabungan

```

#include "Wire.h"
#define DS3231_I2C_ADDRESS 0x68
#include <LiquidCrystal_I2C.h>
#include <Servo.h>

```

```

LiquidCrystal_I2C lcd(0x27, 16, 2);

```

```

Servo myservo;

```

```

int ix;
int mark;
int cacah;
byte second, minute, hour, dayOfWeek, dayOfMonth, month, year;

int buzzer = 2;

// Convert normal decimal numbers to binary coded decimal
byte decToBcd(byte val)
{
  return( (val/10*16) + (val%10) );
}
// Convert binary coded decimal to normal decimal numbers
byte bcdToDec(byte val)
{
  return( (val/16*10) + (val%16) );
}

void setup()
{
  lcd.clear();
  lcd.begin();
  lcd.noCursor();
  myservo.attach(9);

  myservo.write(100);

  pinMode(buzzer,OUTPUT);

  Wire.begin();
  Serial.begin(9600);
  // set the initial time here:
  // DS3231 seconds, minutes, hours, day, date, month, year
  // setDS3231time(0,14,10,2,8,7,19);

}

void setDS3231time(byte second, byte minute, byte hour, byte dayOfWeek, byte
dayOfMonth, byte month, byte year)
{

```

```

// sets time and date data to DS3231
Wire.beginTransmission(DS3231_I2C_ADDRESS);
Wire.write(0); // set next input to start at the seconds register
Wire.write(decToBcd(second)); // set seconds
Wire.write(decToBcd(minute)); // set minutes
Wire.write(decToBcd(hour)); // set hours
Wire.write(decToBcd(dayOfWeek)); // set day of week (1=Sunday, 7=Saturday)
Wire.write(decToBcd(dayOfMonth)); // set date (1 to 31)
Wire.write(decToBcd(month)); // set month
Wire.write(decToBcd(year)); // set year (0 to 99)
Wire.endTransmission();
}
void readDS3231time(byte *second,
byte *minute,
byte *hour,
byte *dayOfWeek,
byte *dayOfMonth,
byte *month,
byte *year)
{
Wire.beginTransmission(DS3231_I2C_ADDRESS);
Wire.write(0); // set DS3231 register pointer to 00h
Wire.endTransmission();
Wire.requestFrom(DS3231_I2C_ADDRESS, 7);
// request seven bytes of data from DS3231 starting from register 00h
*second = bcdToDec(Wire.read() & 0x7f);
*minute = bcdToDec(Wire.read());
*hour = bcdToDec(Wire.read() & 0x3f);
*dayOfWeek = bcdToDec(Wire.read());
*dayOfMonth = bcdToDec(Wire.read());
*month = bcdToDec(Wire.read());
*year = bcdToDec(Wire.read());
}
void displayTime()
{
// retrieve data from DS3231
readDS3231time(&second, &minute, &hour, &dayOfWeek, &dayOfMonth,
&month,
&year);
lcd.setCursor(0,0);
// send it to the serial monitor
lcd.print(hour, DEC);

```

```

// convert the byte variable to a decimal number when displayed
lcd.print(":");
if (minute<10)
{
  lcd.print("0");
}
lcd.print(minute, DEC);
lcd.print(":");
if (second<10)
{
  lcd.print("0");
}
lcd.print(second, DEC);
lcd.print(" ");
lcd.print(dayOfMonth, DEC);
lcd.print("/");
lcd.print(month, DEC);
lcd.print("/");
lcd.print(year, DEC);
lcd.print(" Day of week: ");
switch(dayOfWeek){
case 1:
  Serial.println("Sunday");
  break;
case 2:
  Serial.println("Monday");
  break;
case 3:
  Serial.println("Tuesday");
  break;
case 4:
  Serial.println("Wednesday");
  break;
case 5:
  Serial.println("Thursday");
  break;
case 6:
  Serial.println("Friday");
  break;
case 7:
  Serial.println("Saturday");
  break;
}

```



```
}  
}
```

```
void loop()  
{  
  displayTime(); // display the real-time clock data on the Serial Monitor,
```

```
  if((mark == 0)&&(hour == 13)){  
    digitalWrite(buzzer,HIGH);  
    delay(10000);  
    digitalWrite(buzzer,LOW);  
    gerak();  
  }
```

```
  lcd.setCursor(0,0);  
  lcd.print(" Nayu 1461505106");  
  lcd.setCursor(0,1);  
  lcd.print(" Terapi Selesai");
```

```
  delay(1000);  
}
```

```
void gerak(){  
  
  digitalWrite(buzzer,LOW);  
  
  for(ix=100;ix<=200;ix++){  
    myservo.write(ix);  
    delay(50);  
  
  }  
  delay(1000);  
  
  for(ix=200;ix>=100;ix--){  
    myservo.write(ix);  
    delay(50);  
  
  }  
  delay(1000);
```

```

cacah++;

lcd.setCursor(0,1);
lcd.print("X= ");
lcd.print(cacah);

if(cacah >= 10){
  mark = 1;
  return;

}
gerak();

}

```

➤ Sourcecode

```

int sensorPin = A0;
int powerPin = 6 ;
#include <Wire.h>
#include "RTClib.h"
#include <LiquidCrystal.h>
LiquidCrystal lcd(2, 3, 4, 5, 6 , 7);
RTC_DS1307 rtc;
char namaHari[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis",
"Jumat", "Sabtu"};

void setup() {
  lcd.begin(16, 2);
  pinMode (powerPin, OUTPUT);
  digitalWrite ( powerPin,LOW);
  Serial.begin (9600);
  Serial.begin(9600);
  if (! rtc.begin()) {
    Serial.println("RTC TIDAK TERBACA");
    while (1);
  }

  if (! rtc.isrunning()) {
    Serial.println("RTC is NOT running!");

```

```

    rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); //update rtc dari
waktu komputer
}

}

void loop() {
  { DateTime now = rtc.now();
    Serial.print(namaHari[now.dayOfTheWeek()]);
    Serial.print(',');
    Serial.print(now.day(), DEC);
    Serial.print('/');
    Serial.print(now.month(), DEC);
    Serial.print('/');
    Serial.print(now.year(), DEC);
    Serial.print(" ");
    Serial.print(now.hour(), DEC);
    Serial.print(':');
    Serial.print(now.minute(), DEC);
    Serial.print(':');
    Serial.print(now.second(), DEC);
    Serial.println();

    lcd.setCursor(0,0);
    lcd.print("JAM");
    lcd.setCursor(4,0);
    lcd.print("=");
    lcd.setCursor(5,0);
    lcd.print(" ");
    lcd.print(now.hour(), DEC);
    lcd.print(':');
    lcd.print(now.minute(), DEC);
    lcd.print(':');
    lcd.print(now.second(), DEC);
    lcd.println();
    delay (100);}
  {Serial.print("HUM");
  Serial.println(bacaSensor());
  lcd.setCursor (0,1);
  lcd.print("HUM");
  lcd.setCursor(4,1);
  lcd.print("=");
  lcd.setCursor (8,1);

```

```

lcd.print("%");
lcd.setCursor(5,1);
lcd.print(bacaSensor());

}}

```

```

int bacaSensor ()
{
  digitalWrite (powerPin, HIGH);
  delay (500);
  int nilaiSensor = analogRead (sensorPin);
  digitalWrite(powerPin, LOW);
  return 1023 - nilaiSensor;
}
// put your main code here, to run repeatedly:

```

➤ Sourcecode LCD

```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup()
{
  lcd.begin ();
}

void loop()
{
  lcd.setCursor(0,0);
  // send it to the serial monitor
  lcd.print(hour, DEC);
  // convert the byte variable to a decimal number when displayed
  lcd.print(":");
  if (minute<10)
  {
    lcd.print("0");
  }
  lcd.print(minute, DEC);
}

```

```
lcd.print(":");
if (second<10)
{
  lcd.print("0");
}
lcd.print(second, DEC);
lcd.print(" ");
lcd.print(dayOfMonth, DEC);
lcd.print("/");
lcd.print(month, DEC);
lcd.print("/");
lcd.print(year, DEC);
lcd.print(" Day of week: ");

lcd.setCursor(0,1);
lcd.print("X= ");
lcd.print(cacah);
delay(1000);
}
```

Halaman ini sengaja dikosongkan