LAMPIRAN

Source Code Arduino :

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <EEPROM.h>
#include "RTClib.h"
#include <SoftwareSerial.h>

SoftwareSerial mySerial(3, 2);        // RX, TX
RTC_DS1307 rtc;

// Set the LCD address to 0x27 for a 16 chars and 2 line
display
LiquidCrystal_I2C lcd(0x27, 20, 4);        //Alamat I2C SDA
SCL

void sendMessage( String msg ){ //Kirim pesan ke nodemcu
     for( int i = 0 ; i < msg.length() ; i++ ){
           mySerial.write( msg.charAt (i) );
           delay(2);
     }
mySerial.write('@');
}

//================================================>
void setup() {
Serial.begin(9600);
mySerial.begin(9600);
while(false){
     sendMessage("Hello");
     delay(100);
}
```

```
//================================================>
lcd.begin();
//================================================>
lcd.backlight();
//================================================>

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

if (! rtc.isrunning()) {
     rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
     }
}

//================================================>
String getJam(){                //Ambil data jam rtc
     DateTime now = rtc.now();
     returnString(now.hour()) + ":" + String(now.minute()) +
":" +
     String(now.second());
}

String getTanggal(){            //Ambil data tanggal rtc
     DateTime now = rtc.now();
     return String(now.year()) + "-" + String(now.month()) +
"-" +
     String(now.day());
}

//================================================>
float mmPerTip = 0.2 ;
float curahHujan = 0 ;
```

```
float counterCurahHujan = 0 ;
int pinTippingSensor = A0 ;

void readTippingBucket(){       //Deteksi sensor
bool sensorActive = false ;
pinMode( pinTippingSensor , INPUT );
unsigned long tmTipping = millis();
counterCurahHujan = 0 ;
//=========================================>
Serial.println(); Serial.println();
Serial.println("Mode Membaca Tipping Bucket-start");
int counttip = 0 ;

//=========================================>
while( (millis()-tmTipping) < 90000 ){
     if( digitalRead(A0) == sensorActive ){
          counterCurahHujan = counterCurahHujan + 1 ;
//==================================>

Serial.println("#");
if( counttip > 15 ){
     Serial.println();
     counttip = 0 ;
     }else{
          counttip = counttip + 1 ;
     }
     delay(150);
     while( digitalRead(A0) == sensorActive ){
          delay(50);
     }
}
lcdShow();
}
counterCurahHujan;
//================================>
Serial.println("Mode Membaca Tipping Bucket-selesai");
```

```
//==================================>
curahHujan = ( counterCurahHujan * mmPerTip ) ;
//==============================>
tmTipping = millis();
}

//==========================================>
unsigned long tmlcd = millis();
void lcdShow(){
//==========================================>

if( (millis()-tmlcd) > 1000 ){
//==========================================>
lcd.clear();
//==========================================>
lcd.setCursor(0,0);lcd.print("Tictoc  :  "  +    String  (
counterCurahHujan,0 ));
lcd.setCursor(0,1);lcd.print("Curah Hujan : " +  String (
curahHujan ));
lcd.setCursor(0,2);lcd.print("Tanggal : " +  getTanggal());
lcd.setCursor(0,3);lcd.print("Jam : " +  getJam());
tmlcd = millis();
     }
}


//==========================================>
unsigned long tmSend = millis();
void loop() {
//==========================================>
readTippingBucket();
//==========================================>
lcdShow();
//==========================================>
if( (millis()-tmSend) > 1000 && curahHujan > 0 ){
//Mengirim ke esp
```

```
String data = String(curahHujan) + "," + getTanggal() + ","
+ getJam() + ",";
sendMessage(data);
Serial.println();Serial.println();
Serial.println("================");
Serial.println(data);
tmSend = millis();
     }
}
```

Source Code ESP8266 NodeMCU

```
#include <Wire.h>
//============================>
#include <ESP8266HTTPClient.h>
#include <ESP8266WiFi.h>
#include <SoftwareSerial.h>
SoftwareSerial ard;
//============================>
const char* ssid      = "rouuter"   ;
const char* password  = ""  ;

//===============> inisial server <====================//
WiFiServer server(80);
//============================>
void connectWifi(){ //koneksi
//============================>
WiFi.begin(ssid, password);
//============================>
Serial.println("Connecting..");
int countFailed = 0 ;
unsigned long tmOut = millis();
while (WiFi.status() != WL_CONNECTED) {
      delay(500);
      Serial.print(".");
      if( countFailed > 10 ){
            countFailed = 0 ;
            Serial.println();
            WiFi.begin(ssid, password);
      }
      countFailed++ ;
}

server.begin();
Serial.println("Server started");
```

```
//============================>
Serial.println("inisial hotspot finish");
Serial.println(WiFi.localIP());
}

//======================================>
String data = "12" ;
String tgl = "2020-12-02" ;
String jam = "12:45:11" ;
//======================================>
String servername = "192.168.43.40/monitoringcurahhujan" ;
byte mode_uploaddata = 1 ;
byte mode_uploadlogdata = 2 ;

String createUrl( int mode ){
      String filephpname = "" ;
      if( mode == mode_uploaddata ){
            filephpname = "uploaddata.php" ;
      }
      if( mode == mode_uploadlogdata ){
            filephpname = "uploadlogdata.php" ;
      }
return "http://"+servername+"/"+filephpname+"?data=" + data
+ "&tgl=" + tgl + "&jam=" + jam ;
}
//====================>




String cmd = "";
void sendReqCommand(String str){
      Serial.println(str);
```

```
      HTTPClient http;
      http.begin(str);
      int httpCode= http.GET();
      if(httpCode>0){
                  Serial.println("Output dari server");
            cmd = http.getString();
                  for( int i = 0 ; i < cmd.length() ; i++ ){

      Serial.print(cmd.charAt(i));delay(0.1);
                  }
            Serial.println("success");
      }else{
                  Serial.println("No Connection");
      }
      http.end();
}

//=========================================>
String getCompMessageStr(){ //
      String temps = "" ;
      while( Serial.available() ){
            char chr = char(Serial.read());
            temps += chr ;
            delay(4);
       }
      return temps ;
}




String getArdMessageStr(){       //mengambil pesan arduino
      String temps = "" ;
      while( true ){
```

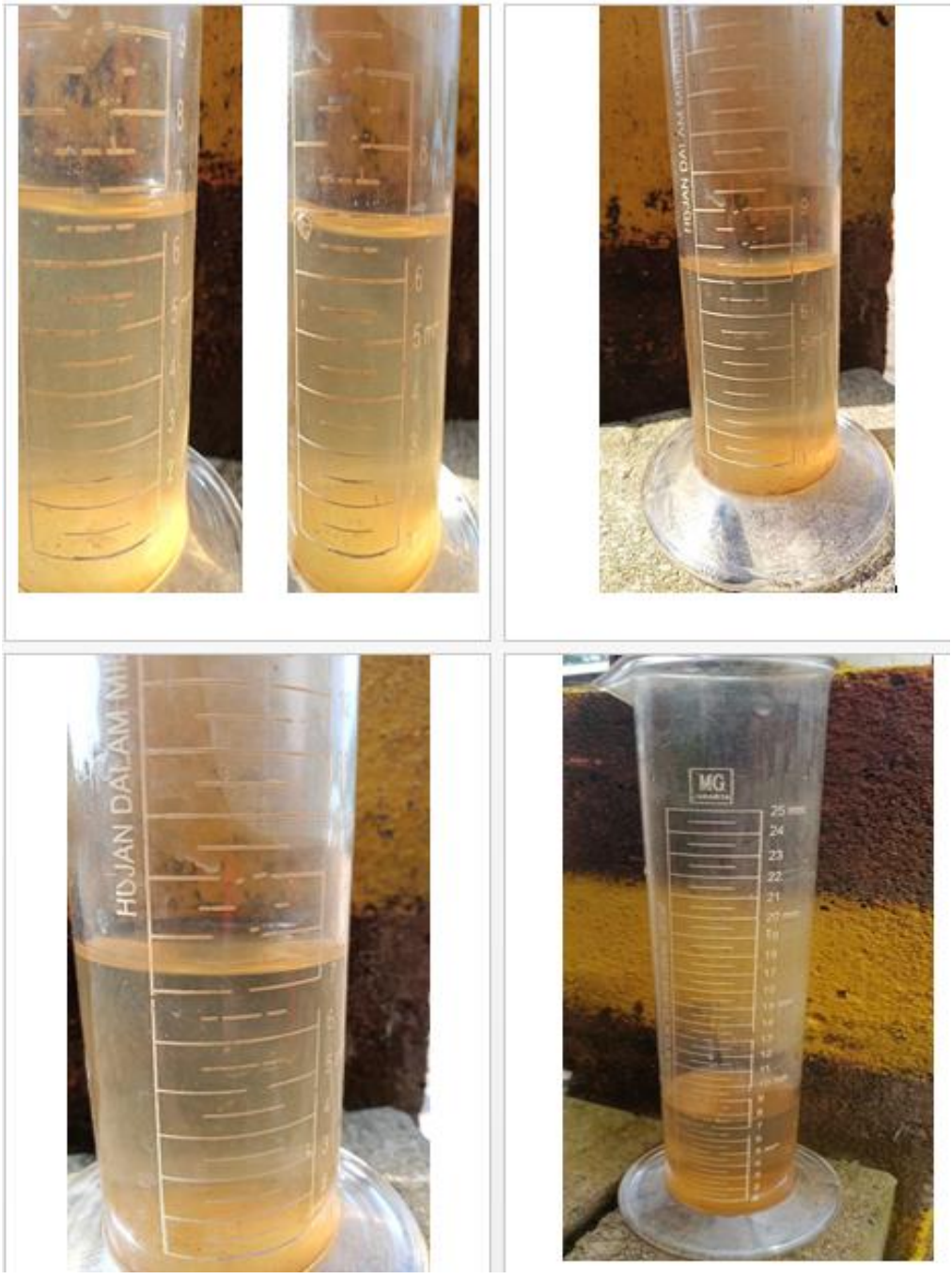
```
//=========================================>
char chr = ' ' ;
unsigned long tms = millis();
while((millis()-tms) < 1000 ){
if( ard.available() ){
      chr = char(ard.read());
            break ;
      }
}
if( chr == '@' )
break ;
if( chr != ' ' ){
      temps += chr ;
      }
}
return temps ;
}

//==> data masuk "12,2020-12-02,13:15:14"
String parseMessage( String msg ){
      String temp = "" ;
      int index = 0 ;
      for( int i = 0 ; i < msg.length() ; i++ ){
             char temps = msg.charAt(i);
      if( temps == ',' ){
                  Serial.println("Num : " + String( index ) +
", Value : " + temp );
            if( index == 0 ){
                  data = temp ;
      }
            else if( index == 1 ){
                  tgl = temp ;
            }
            else if( index == 2 ){
                  jam = temp ;
            }
```

```
            temp = "" ;
            index++ ;
      }else{
                  temp += temps ;
      }
      delay(1);
       }
return temp ;
}

//============================>
String msg = "" ; //Hardware serial koneksi esp ke arduino
//============================>
void setup() {
      Serial.begin(9600);
      ard.begin(9600, SWSERIAL_8N1, 14, 12, false, 100, 11);
//============================>
while(false){
      String msg = getArdMessageStr();
      if( msg.length() > 0 ){
                  Serial.println(msg);
      }
      delay(10);
      }
//=========================> inisial wifi .
connectWifi();
}
//================================>
unsigned long tmSendLogData = millis();

if( (millis()-tmSendLogData) > 30000){
      Serial.println("Kirim data log curah hujan ...");
      sendReqCommand( createUrl( mode_uploadlogdata ) );
      tmSendLogData = millis();
      }
String dt = "" ;
```

```
String dt_ard = getArdMessageStr();
String dt_comp = getCompMessageStr();
if( dt_ard.length() > 0 ){
      dt = dt_ard ;
      }else{
            dt = dt_comp ;
      }

if( dt.length() > 1 ){                  //Membuka pesan ardino
      Serial.println("Data : " + dt );
      parseMessage( dt );
      }
}
```

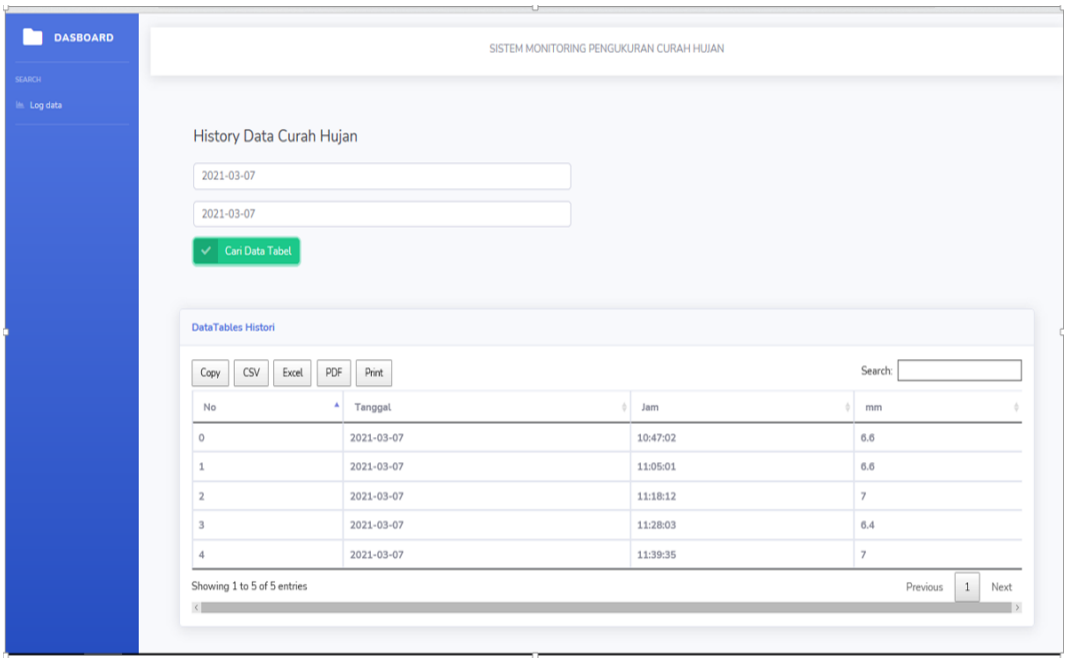
Dokumentasi

Berikut lampiran dokumentasi dari hasil pengujian perangkat manual :




OMBROMETER

Berikut dokumentasi hasil dari pengujian perangkat otomatis :




OMBROMETER