LAMPIRAN

Program Arduino Keseluruhan

```
#include <IRremote.h>
IRsend irsend;

//LIBRARY//
#define BLYNK_PRINT Serial
#include "Wire.h"
#define TINY_GSM_MODEM_SIM900
#include <TinyGsmClient.h>
#include <BlynkSimpleTinyGSM.h>

#include <SimpleTimer.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,20,4);  // LCD2004
#include <Sodaq_DS3231.h>
String strNamaHari[] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jumat",
"Sabtu"};
SimpleTimer timer;
String currentTime;
String currentDate;

//Setting APN Internet
char auth[] = "a-to_KP6C4-3qFdRk3hnlAywaBEj5QBg";
char apn[]  = "internet";
char user[] = "";
char pass[] = "";

#define SerialAT Serial1
TinyGsm modem(SerialAT);
```

```
//deklarasi pin sensor YL
int pot1, pot2, pot3, pot4, pot5, pot6, pot7, pot8, pot9, pot10, pot11, pot12;

//Set Point
const int spYL = 600;
const int spTFG = 20;
const int AIRC = 20;

//deklarasi pin sensor Float
int FloatSensor = 47;
int Float = 1;

//DHT21
#include "DHT.h"
#define DHTPIN 39
#define DHTTYPE DHT21
DHT dht(DHTPIN, DHTTYPE);
float h,t;

//Battery
#define Battery_PIN A13
float adc_voltage = 0.0;
float Battery = 0.0;
int adc_value = 0;
float R1 = 30000.0;
float R2 = 7500.0;
float ref_voltage = 5.0;

//Arus beban
const int pinADC = A14;
int sensitivitas = 66; //tegantung sensor arus yang digunakan, yang ini 5A
```

```
int nilaiadc= 00;
int teganganoffset = 2500; //nilai pembacaan offset saat tidak ada arus yang lewat
double tegangan = 00;
double Amps = 00;

//ultrasonic
const int trigPin = 42;
const int echoPin = 41;
const int trigPin1 = 37;
const int echoPin1 = 36;
long duration, duration1, Water_Fogging, Waterpot;
int distance,distance1;

//deklarasi pin Relay
#define Fogging_Pump 29
#define Fogging_valve 27
#define Pump_Waterpot 31
#define Valve_Waterpot 26
#define Pesticide_Pump 30
#define Exhaust_Fan 33
#define Air_Cooler 32
#define Air_Cooler_Pump 28

int Cooler;
int
SET = 1;
BLYNK_WRITE(V30)
{Cooler = param.asInt();}

void setup(){
  Serial.begin(115200);
```

```
SerialAT.begin(9600);
rtc.begin();
lcd.init();                    // initialize the lcd

//inisialisasi PIN
pinMode(A1, INPUT);
pinMode(A2, INPUT);
pinMode(A3, INPUT);
pinMode(A4, INPUT);
pinMode(A5, INPUT);
pinMode(A6, INPUT);
pinMode(A7, INPUT);
pinMode(A8, INPUT);
pinMode(A9, INPUT);
pinMode(A10, INPUT);
pinMode(A11, INPUT);
pinMode(A12, INPUT);
pinMode(Float, INPUT);
pinMode(Battery_PIN, INPUT);
pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin, INPUT); // Sets the echoPin as an Input
pinMode(trigPin1, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin1, INPUT); // Sets the echoPin as an Input

pinMode(Fogging_Pump, OUTPUT);
pinMode(Fogging_valve, OUTPUT);
pinMode(Pump_Waterpot, OUTPUT);
pinMode(Valve_Waterpot, OUTPUT);
pinMode(Pesticide_Pump, OUTPUT);
pinMode(Exhaust_Fan, OUTPUT);
pinMode(Air_Cooler, OUTPUT);
```

```
pinMode(Air_Cooler_Pump, OUTPUT);

//kondisi awal relay
digitalWrite(Fogging_Pump, HIGH);
digitalWrite(Fogging_valve, HIGH);
digitalWrite(Pump_Waterpot, HIGH);
digitalWrite(Valve_Waterpot, HIGH);
digitalWrite(Pesticide_Pump, HIGH);
digitalWrite(Exhaust_Fan, HIGH);
digitalWrite(Air_Cooler, HIGH);
digitalWrite(Air_Cooler_Pump, HIGH);

// Print a message to the LCD.
lcd.backlight();
lcd.setCursor(0,0);
lcd.print("   SELAMAT DATANG");
lcd.setCursor(0,1);
lcd.print("GREEN HOUSE PROJECT");
lcd.setCursor(0,2);
lcd.print("    POWERED BY");
lcd.setCursor(0,3);
lcd.print("   UNITY STATION");
delay(2000);
//timer.setInterval(1000L, getSendData);
SerialAT.println("AT");
Serial.println("Initializing modem...");
modem.restart();
Blynk.begin(auth, modem, apn, user, pass);
dht.begin();
lcd.clear();
delay(10);
```

```
}

void loop(){
  //timer.run();
  Blynk.run();
  systemReset();
  running_code();
  LCD();
  data_sensor();
//  if (Cooler == 1){
//    Cooler_Aktif();
//    Blynk.virtualWrite(V30, LOW);
//  }
}

//void getSendData(){
//  data_sensor();
//}

void data_sensor(){
  ruleLEVELf();
  ruleLEVELp();
  h = dht.readHumidity();
  t = dht.readTemperature();
  ruleFG();
  ruleEX();
  ruleTFG();
  ruleAIRC();

  adc_value = analogRead(Battery_PIN);
  adc_voltage  = (adc_value * ref_voltage) / 1024.0;
```

```
  Battery = adc_voltage / (R2/(R1+R2));

  nilaiadc = analogRead(pinADC);
  tegangan = (nilaiadc / 1024.0) * 5000;
  Amps = ((tegangan - teganganoffset) / sensitivitas);

  ruleYL();

  Blynk.virtualWrite(V0, pot1);
  Blynk.virtualWrite(V1, pot2);
  Blynk.virtualWrite(V2, pot3);
  Blynk.virtualWrite(V3, pot4);
  Blynk.virtualWrite(V4, pot5);
  Blynk.virtualWrite(V5, pot6);
  Blynk.virtualWrite(V6, pot7);
  Blynk.virtualWrite(V7, pot8);
  Blynk.virtualWrite(V8, pot9);
  Blynk.virtualWrite(V9, pot10);
  Blynk.virtualWrite(V10, pot11);
  Blynk.virtualWrite(V11, pot12);
  Blynk.virtualWrite(V12, Water_Fogging);
  Blynk.virtualWrite(V13, Waterpot);
  Blynk.virtualWrite(V14, h);
  Blynk.virtualWrite(V15, t);
  Blynk.virtualWrite(V17, Battery);
  Blynk.virtualWrite(V18, Amps);
  Blynk.virtualWrite(V19, Float);



}
```

```
void running_code(){
  DateTime now = rtc.now();
  currentTime = String(now.hour()) + ":" + now.minute() + ":" + now.second();
  currentDate = String(now.date()) + " " + now.month() + " " + now.year();

  if ((strNamaHari[now.dayOfWeek()-1] == "Senin" || strNamaHari[now.dayOfWeek()-1] == "Kamis") && (now.hour() == 10 && now.minute() == 0 &&  now.second()>= 0 && now.second()<=10)){
    digitalWrite(Pesticide_Pump, LOW); //penyiraman pestisida
    delay(5000);
  }
  else {
    digitalWrite(Pesticide_Pump, HIGH);
    Serial.println("Koding jalan");
    FloatR();
    data_sensor();
  }
}

void YLsensor(){
  pot1 = analogRead(A1);
  pot2 = analogRead(A2);
  pot3 = analogRead(A3);
  pot4 = analogRead(A4);
  pot5 = analogRead(A5);
  pot6 = analogRead(A6);
  pot7 = analogRead(A7);
  pot8 = analogRead(A8);
  pot9 = analogRead(A9);
  pot10 = analogRead(A10);
```

```
  pot11 = analogRead(A11);
  pot12 = analogRead(A12);
}

void ultrasonic(){
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  // Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(echoPin, HIGH);
  // Calculating the distance
  distance = duration * 0.034 / 2;
  Water_Fogging = distance;

  digitalWrite(trigPin1, LOW);
  delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigPin1, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin1, LOW);
  // Reads the echoPin, returns the sound wave travel time in microseconds
  duration1 = pulseIn(echoPin1, HIGH);
  // Calculating the distance
  distance1 = duration1 * 0.034 / 2;
  Waterpot = distance1;
}

void ruleYL(){        //rule Sensor YL terhadap penyiraman
```

```
  YLsensor();
  if (pot1 > 600 || pot2 > 600 || pot3 > 600 || pot4 > 600 || pot5 > 600 || pot6 > 600 ||
pot7 > 600 || pot8 > 600 || pot9 > 600 || pot10 > 600 || pot11 > 600 || pot12 > 600 ){
    digitalWrite(Pump_Waterpot, LOW); //Aktif
    Serial.println("Penyiraman Aktif");
  }
  else if (pot1 <= 600 || pot2 <= 600 || pot3 <= 600 || pot4 <= 600 || pot5 <= 600 ||
pot6 <= 600 || pot7 <= 600 || pot8 <= 600 || pot9 <= 600 || pot10 <= 600 || pot11 <=
600 || pot12 <= 600 ){
    digitalWrite(Pump_Waterpot, HIGH);
    Serial.println("Penyiraman Mati");
  }
}

void ruleFG(){      //rule Pompa Fogging & Exhaust Fan
  if (h <60){
    Serial.println("Pompa Fogging Aktif");
    digitalWrite(Fogging_Pump, LOW);  //Aktif
  }
  else if (h >= 60){
    Serial.println("Pompa Fogging Mati");
    digitalWrite(Fogging_Pump, HIGH);
  }
}

void ruleEX(){
  if (h <80){
    Serial.println("EX Aktif");
    digitalWrite(Exhaust_Fan, LOW);   //Aktif
  }
  else if (h >= 80){
```

```
    Serial.println("EX Mati");
    digitalWrite(Exhaust_Fan, HIGH);
  }
}

void ruleTFG(){       //rule Pompa Fogging inputan Temperature Udara
  if (t > 36){
    Serial.println("Fogging Aktif");
    digitalWrite(Fogging_Pump, LOW);  //Aktif
  }
  else if (t <= 36){
    Serial.println("Fogging Mati");
    digitalWrite(Fogging_Pump, HIGH);
  }
}

void ruleAIRC(){
  if (t > AIRC && SET == 1){
    Serial.println("Air Cooler Aktif");
    digitalWrite(Air_Cooler, LOW);    //Aktif
    delay(2000);
    Cooler_Aktif();
    SET = 0;
  }
  else if (t <= AIRC){
    Serial.println("Air Cooler Mati");
    digitalWrite(Air_Cooler, HIGH);
    SET = 1;
  }
}
```

```
void ruleLEVELf(){
  ultrasonic();
  if (Water_Fogging > 26){
    Serial.println("valve Fogging Aktif");
    digitalWrite(Fogging_valve, LOW);  //Aktif
  }
  if (Water_Fogging < 17){
    Serial.println("valve Fogging Mati");
    digitalWrite(Fogging_valve, HIGH);
  }
}

void ruleLEVELp(){
  ultrasonic();
  if (Waterpot > 21){
    Serial.println("valve Penyiraman Aktif");
    digitalWrite(Valve_Waterpot, LOW);  //Aktif
  }
  if (Waterpot < 17){
    Serial.println("valve Penyiraman Mati");
    digitalWrite(Valve_Waterpot, HIGH);
  }
}

void FloatR(){
  Float = digitalRead(FloatSensor);
  if (Float == 0){
    Blynk.virtualWrite(V16, LOW);
    digitalWrite(Air_Cooler_Pump, LOW);
    }
  if (Float == 1) {
```

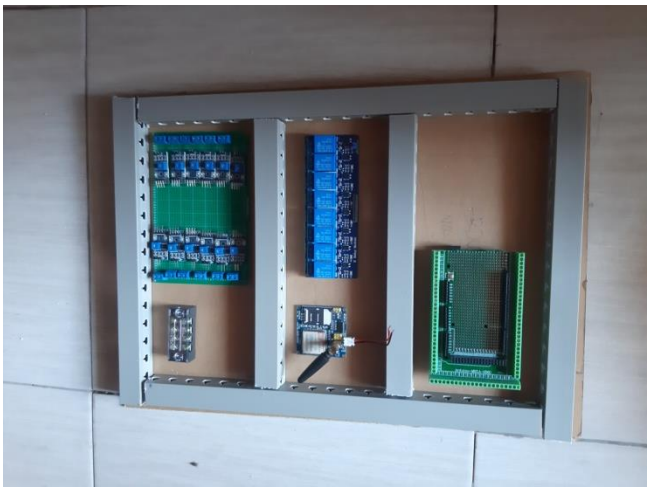
```
    Blynk.virtualWrite(V16, HIGH);
    Serial.println("Pompa Air Cooler Mati");
    digitalWrite(Air_Cooler_Pump, HIGH); //Mati
    }
}

void Cooler_Aktif(){
  unsigned int ON_OFF[24] =
{1250,450,1250,400,450,1250,1250,450,1250,400,400,1300,450,1200,450,1250,4
50,1200,450,1250,400,1250,1250,};
  irsend.sendRaw(ON_OFF,24,38);
  delay(200);
  unsigned int Kecepatan[24] =
{1300,400,1250,450,450,1200,1300,400,1250,400,500,1200,500,1200,450,1200,4
50,1250,450,1200,1250,450,500,};
  irsend.sendRaw(Kecepatan,24,38);
  delay(200);
  irsend.sendRaw(Kecepatan,24,38);
  delay(200);
  irsend.sendRaw(Kecepatan,24,38);
  delay(200);
  unsigned int Pendingin[24] =
{1250,450,1200,500,400,1250,1200,500,1200,500,1150,500,400,1300,400,1250,4
00,1300,400,1250,1200,500,1200,};
  irsend.sendRaw(Pendingin,24,38);
  delay(200);
  unsigned int Arah[24] =
{1250,450,1200,500,400,1250,1200,500,1200,450,450,1250,400,1250,1250,450,4
00,1300,400,1250,400,1300,400,};
  irsend.sendRaw(Arah,24,38);
  delay(10);
```

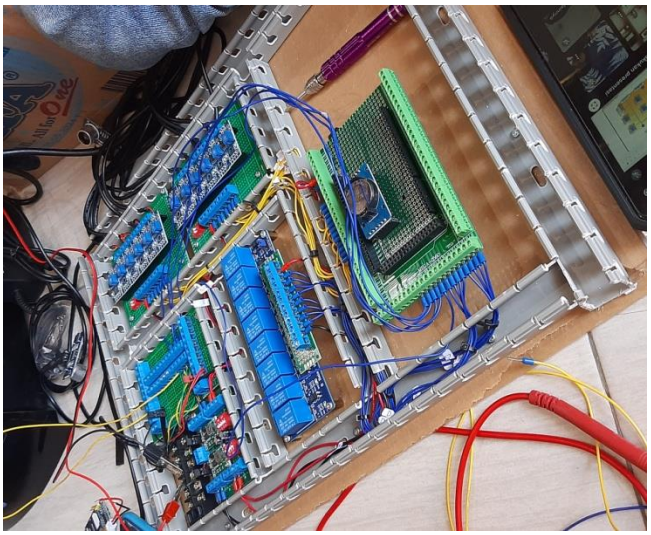
```
}

void LCD(){
  lcd.setCursor(0,0);
  lcd.print("DF:");
  lcd.setCursor(4,0);
  lcd.print(int (Water_Fogging)); //Data
  lcd.setCursor(7,0);
  lcd.print("CM | DP:");
  lcd.setCursor(15,0);
  lcd.print(int(Waterpot)); // DAta
  lcd.setCursor(18,0);
  lcd.print("CM");

  lcd.setCursor(0,1);
  lcd.print("Ta:");
  lcd.setCursor(4,1);
  lcd.print(int(t)); //Data
  lcd.setCursor(7,1);
  lcd.print("'C | Ha:");
  lcd.setCursor(15,1);
  lcd.print(int(h)); // DAta
  lcd.setCursor(18,1);
  lcd.print("%");

  lcd.setCursor(0,2);
  lcd.print("Vo:");
  lcd.setCursor(4,2);
  lcd.print(Battery,1); //Data
  lcd.setCursor(7,2);
  lcd.print("Vdc| Io:");
```

```
  lcd.setCursor(15,2);
  lcd.print(Amps,2); // DAta
  //lcd.setCursor(18,2);
  //lcd.print("A");

  lcd.setCursor(0,3);
  lcd.print("He:");
  lcd.setCursor(4,3);
  lcd.print(int(pot3)); //Data
  lcd.setCursor(9,3);
  lcd.print(" | C*:");
  lcd.setCursor(15,3);
  if (Float == LOW){lcd.print("Full");}
  else if (Float == HIGH){lcd.print("Empt");}
}

void systemReset(){
  DateTime now = rtc.now();
  currentTime = String(now.hour()) + ":" + now.minute() + ":" + now.second();
  currentDate = String(now.date()) + " " + now.month() + " " + now.year();
  if (now.hour() == 1 && now.minute() == 0 && now.second()>= 0 &&
now.second()<=10){
    modem.restart();
    Blynk.begin(auth, modem, apn, user, pass);
  }
  if (now.hour() == 4 && now.minute() == 0 && now.second()>= 0 &&
now.second()<=10){
    modem.restart();
    Blynk.begin(auth, modem, apn, user, pass);
  }
```

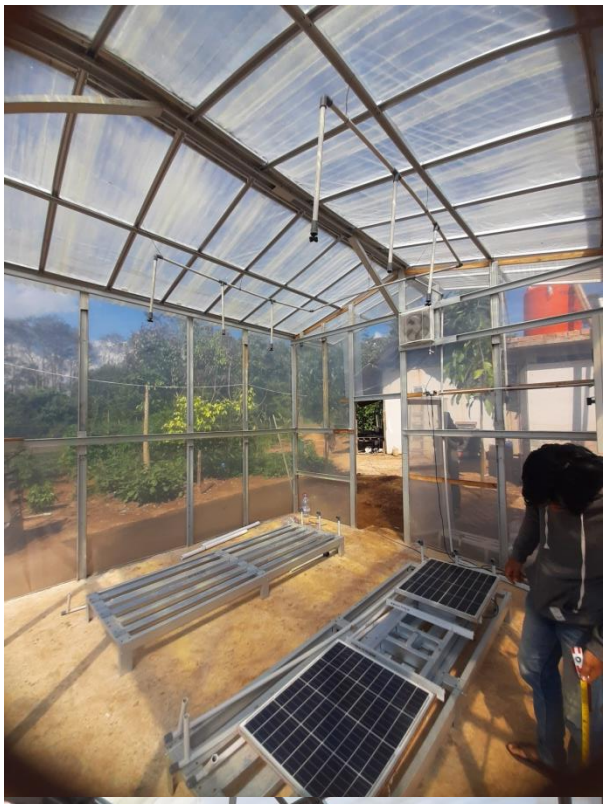
```
  if (now.hour() == 7 && now.minute() == 0 &&  now.second()>= 0 && now.second()<=10){
    modem.restart();
    Blynk.begin(auth, modem, apn, user, pass);
  }
  if (now.hour() == 10 && now.minute() == 0 &&  now.second()>= 0 && now.second()<=10){
    modem.restart();
    Blynk.begin(auth, modem, apn, user, pass);
  }
  if (now.hour() == 13 && now.minute() == 0 &&  now.second()>= 0 && now.second()<=10){
    modem.restart();
    Blynk.begin(auth, modem, apn, user, pass);
  }
  if (now.hour() == 16 && now.minute() == 0 &&  now.second()>= 0 && now.second()<=10){
    modem.restart();
    Blynk.begin(auth, modem, apn, user, pass);
  }
  if (now.hour() == 19 && now.minute() == 0 &&  now.second()>= 0 && now.second()<=10){
    modem.restart();
    Blynk.begin(auth, modem, apn, user, pass);
  }
  if (now.hour() == 22 && now.minute() == 0 &&  now.second()>= 0 && now.second()<=10){
    modem.restart();
    Blynk.begin(auth, modem, apn, user, pass);
  }
}
```

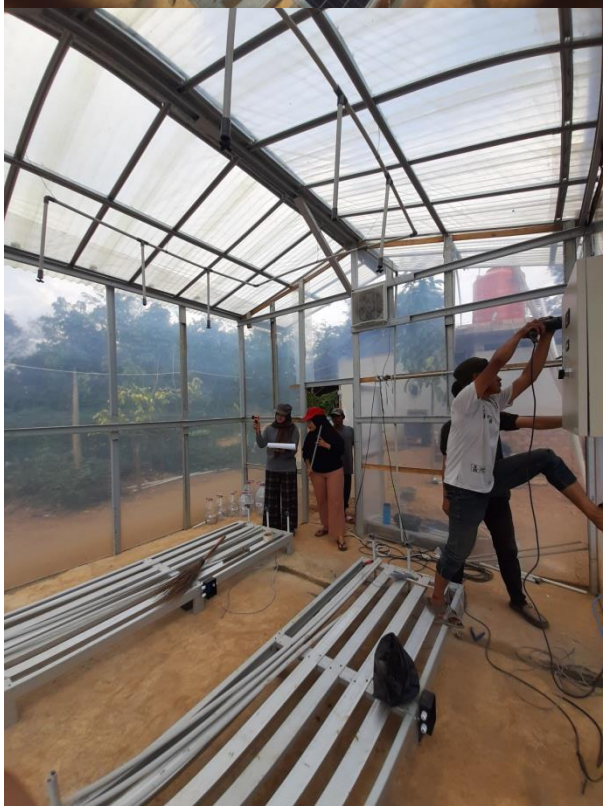
DOKUMENTASI






ALAT UKUR DIGITAL





KEMENTERIAN PENDIDIKAN,KEBUDAYAAN
RISET,DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA
Jalan Srijaya Negara, Palembang 30139
Telp. 0711-353414 fax. 0711-355918
Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id

REKOMENDASI UJIAN LAPORAN AKHIR (LA)

Pembimbing Laporan Akhir memberikan rekomendasi kepada,

Nama	:	Indri Rosanti
NIM	:	061930320047
Jurusan/Program Studi	:	Teknik Elektro/DIII Teknik Elektronika
Judul Laporan Akhir	:	Sistem Pengaturan Kelembaban Udara dan Pendeteksi Kelembaban Tanah pada *Smart Greenhouse*

Mahasiswa tersebut telah memenuhi persyaratan dan dapat mengikuti Ujian Laporan Akhir (LA) pada Tahun Akademik 2021/2022

Palembang, 28 Juli 2022

Pembimbing I,	Pembimbing II,
(Ir. Pola Risma,M.T)	(Dr. Eng. Tresna Dewi,ST., M.Eng)
NIP. 196303281990032001	NIP. 197711252000032001

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
RISET DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA
Jalan Srijaya Negara, Palembang 30139
Telp. 0711-353414 Fax. 0711-355918
Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id

KESEPAKATAN BIMBINGAN LAPORAN AKHIR (LA)

Kami yang berlanda tangan di bawah ini,

Pihak Pertama

Nama	: Indri Rsanti
NPM	061930320047
Jurusan	: Teknik Elektro
Program Studi	: DIII Teknik Elektronika

Pihak Kedua

Nama	: Ir. Pola Risma,M.T.
NIP	196303281990032001
Jurusan	: Teknik Elektro
Program Studi	: DIII Teknik Elektronika

Pada hari ini 26 Juli 2021 telah sepakat untuk melakukan konsultasi bimbingan Laporan Akhir.

Konsultasi bimbingan sekurang-kurangnya 1 (satu) kali dalam satu minggu. Pelaksanaan bimbingan pada setiap hari Senin pukul 80.00 – 12.00 WIB tempat di Politeknik Negeri Sriwijaya.

Demikianlah kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Laporan Akhir.

Palembang, 26 Juli 2021

Pihak Pertama,

(Indr Rosanti)
NPM 061930320047

Pihak Kedua,

(Ir. Pola Risma,M.T.)
NIP 196303281990032001

Mengetahui,
Ketua Jurusan

(Ir. Iskandar Lutfi, M.T)
NIP 196501291991031002

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
RISET DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA
Jalan Srijaya Negara, Palembang 30139
Telp. 0711-353414 Fax. 0711-355918
Website : www.polisriwijaya.ac.id E-mail : info@polsri.ac.id

KESEPAKATAN BIMBINGAN LAPORAN AKHIR (LA)

Kami yang bertanda tangan di bawah ini,

Pihak Pertama

Nama	: Indri Rosanti
NPM	061930320047
Jurusan	: Teknik Elektro
Program Studi	: DIII Teknik Elektronika

Pihak Kedua

Nama	: Dr. Eng. Tresna Dewi, S.T., M.Eng.
NIP	197711252000032001
Jurusan	: Teknik Elektro
Program Studi	: DIII Teknik Elektronika

Pada hari ini ..Jum'at 2022..telah sepakat untuk melakukan konsultasi bimbingan Laporan Akhir.

Konsultasi bimbingan sekurang-kurangnya 1 (satu) kali dalam satu minggu. Pelaksanaan bimbingan pada setiap hari Senin pukul 80.00 – 12.00 WIB tempat di Politeknik Negeri Sriwijaya.

Demikianlah kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Laporan Akhir.

Palembang, 28 Juli 2022

Pihak Pertama,

(Indri Rosanti)
NPM 06193032047

Pihak Kedua,

(Dr. Eng. Tresna Dewi, S.T., M.Eng.)
NIP 197711252000032001

Mengetahui,
Ketua Jurusan

(Ir. Iskandar Lutfi, M.T)
NIP 196501291991031002

No. Dok. : F-PBM-23 No. Rev. : 00

KEMENTERIAN PENDIDIKAN,KEBUDAYAAN
RISET,DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA
Jalan Srijaya Negara, Palembang 30139
Telp. 0711-353414 fax. 0711-355918
Website : www.polisriwijaya.ac.id **E-mail :** info@polsri.ac.id

PELAKSANAAN REVISI LAPORAN AKHIR

Mahasiswa berikut,

Nama	:	Indri Rosanti
NIM	:	061930320047
Jurusan/Program Studi	:	Teknik Elektro/DIII Teknik Elektronika
Judul Laporan Akhir	:	Sistem Pengaturan Kelembaban Udara dan Pendeteksi Kelembaban Tanah pada *Smart Green House*

Telah melaksanakan revisi terhadap Laporan Akhir yang diujikan pada hari ..... tanggal ..... bulan ..... tahun ..... Pelaksanaan revisi terhadap Laporan Akhir tersebut telah disetujui oleh Dosen Penguji yang memberikan revisi:

No.	Komentar	Nama Dosen Penguji *)	Tanggal	Tanda Tangan
		Ir. Pola Risma, MT.		
2	Acc	Ir. Faisal Damsi, MT.,	30/08 22	
		Selamat Muslimin, ST., Mkom.,		
1	Acc	Tegar Prasetyo,S.Pd., M.Eng	30/8 22	

Palembang, ...
Ketua Penguji **),

(.....)
NIP .....

Catatan:
*) Dosen penguji yang memberikan revisi saat ujian laporan akhir
**) Dosen penguji yang ditugaskan sebagai Ketua Penguji saat ujian LA
Lembaran pelaksanaan revisi ini harus dilampirkan dalam Laporan Akhir.

COMPONENTS101

Soil Moisture Sensor Module

16 April 2020 · 0 Comments

(/sites/default/files/components/Moisture-Sensor-Module.png)

Soil Moisture Sensor Module

(/sites/default/files/component_pin/Moisture-Sensor-Module-Pinout.jpg)

Soil Moisture Sensor Module

This **soil moisture sensor module** is used to detect the moisture of the soil. It measures the volumetric content of water inside the soil and gives us the moisture level as output. The module has both digital and analog outputs and a potentiometer to adjust the threshold level.

COMPONENTS101

Pin Name	Description
VCC	The Vcc pin powers the module, typically with +5V
GND	Power Supply Ground
DO	Digital Out Pin for Digital Output.
AO	Analog Out Pin for Analog Output

Soil Moisture Sensor Module Features & Specifications

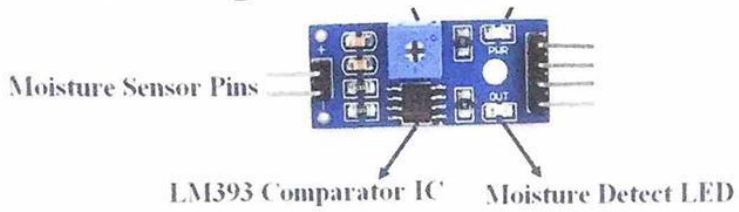
- Operating Voltage: 3.3V to 5V DC
- Operating Current: 15mA
- Output Digital - 0V to 5V, Adjustable trigger level from preset
- Output Analog - 0V to 5V based on infrared radiation from fire flame falling on the sensor
- LEDs indicating output and power
- PCB Size: 3.2cm x 1.4cm
- LM393 based design
- Easy to use with Microcontrollers or even with normal Digital/Analog IC
- Small, cheap and easily available

Alternate Sensor Modules: IR Sensor Module (https://components101.com/sensors/ir-sensor-module), LDR Sensor Module, Flame Sensor Module, TP4056A Li-ion Battery Charging/Discharging Module (https://components101.com/tp4056a-li-ion-battery-chargingdischarging-module), DS3231 RTC Module (https://components101.com/modules/ds3231-rtc-module-pinout-circuit-datasheet), TMC2209 Stepper Motor Driver Module (https://components101.com/modules/tmc2209-stepper-motor-driver-module), DRV8825 Stepper Motor Driver Module (https://components101.com/modules/drv8825-stepper-motor-driver-module), A4988 Stepper Motor Driver Module (https://components101.com/modules/a4988-stepper-motor-driver-module), NEO-6MV2 GPS Module (https://components101.com/modules/neo-6mv2-gps-module), Joystick Module (https://components101.com/modules/joystick-module), EM18 - RFID Reader Module (https://components101.com/modules/em18-rfid-reader-module), ADXL335 Accelerometer Module, HMC5883L Magnetometer Module

Related Components: LM393 Comparator IC (https://components101.com/ics/lm393-low-offset-voltage-dual-comparators), 10K Potentiometer (https://components101.com/potentiometer), Capacitor (https://components101.com/capacitors)

Brief about Soil Moisture Sensor Module

This Moisture sensor module consists of a Moisture sensor, Resistors, Capacitor, Potentiometer, Comparator LM393 IC, Power and Status LED in an integrated circuit.



LM393 IC

LM393 Comparator IC (https://components101.com/ics/lm393-low-offset-voltage-dual-comparators) is used as a voltage comparator in this Moisture sensor module. Pin 2 of LM393 is connected to Preset (10KΩ Pot) while pin 3 is connected to Moisture sensor pin. The comparator IC will compare the threshold voltage set using the preset (pin2) and the sensor pin (pin3).

Moisture Sensor

The moisture sensor consists of two probes that are used to detect the moisture of the soil. The moisture sensor probes are coated with immersion gold that protects Nickel from oxidation. These two probes are used to pass the current through the soil and then the sensor reads the resistance to get the moisture values.

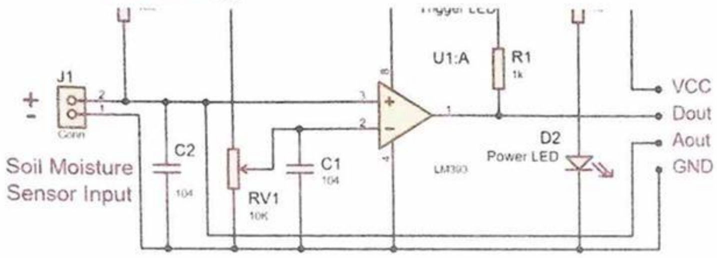
Preset (Trimmer pot)

Using the onboard preset you can adjust the threshold (sensitivity) of the digital output.

How to Use Soil Moisture Sensor Module

Moisture sensor module consists of four pins i.e. VCC, GND, DO, AO. Digital out pin is connected to the output pin of LM393 comparator IC while the analog pin is connected to Moisture sensor. The internal Circuit diagram of the Moisture sensor module is given below.

COMPONENTS101



Using a Moisture sensor module with a microcontroller is very easy. Connect the Analog/Digital Output pin of the module to the Analog/Digital pin of Microcontroller. Connect VCC and GND pins to 5V and GND pins of Microcontroller. After that insert the probe inside the soil. When there is more water presented in the soil, it will conduct more electricity that means resistance will be low and the moisture level will be high.

Applications of Soil Moisture Sensor

- Gardening
- Irrigation Systems
- Used in Controlled Environments

Tags

SOIL MOISTURE SENSOR (/TAGS/SOIL-MOISTURE-SENSOR) SENSORS (/TAGS/SENSORS)

Related Post

Comments

VELVET

14 September 2020

Permalink (/comment/475#comment-475)

how much area can the soil... (/comment/475#comment-475)

Comment

how much area can the soil moisture sensor measure? pls answer

COMPONENTS101

ADMIN

13 August 2021

Permalink (/comment/1300#comment-1300)

The sensor can measure... (/comment/1300#comment-1300)

Comment

The sensor can measure moisture only of the soil that is in immediate contact on the sensor.

LOG IN (/USER/LOGIN?DESTINATION=/MODULES/SOIL-MOISTURE-SENSOR-MODULE%23COMMENT-FORM) or **REGISTER (/USER/REGISTER?DESTINATION=/MODULES/SOIL-MOISTURE-SENSOR-MODULE%23COMMENT-FORM)** to post comments

DENDRITE

9 January 2022

Permalink (/comment/2768#comment-2768)

Comment on Schematic (/comment/2768#comment-2768)

Comment

I tried to build this circuit on a breadboard using the schematic provided, but it didn't seem to work. After examining it more closely and comparing it to additional online resources of voltage comparator circuits for moisture, I found most similar circuits connect pin 4 of the IC directly to ground, not through a capacitor. This capacitor is usually found connecting from the Vcc line to ground. Once I made these changes, the circuit worked fine.

LOG IN (/USER/LOGIN?DESTINATION=/MODULES/SOIL-MOISTURE-SENSOR-MODULE%23COMMENT-FORM) or **REGISTER (/USER/REGISTER?DESTINATION=/MODULES/SOIL-MOISTURE-SENSOR-MODULE%23COMMENT-FORM)** to post comments

LATEST PRODUCTS

(/news/7-channel-dmos-transistor-array-for-industrial-and-agricultural-automation)

7-Channel DMOS Transistor Array for Industrial and Agricultural Automation (/news/7-channel-dmos-transistor-array-for-industrial-and-agricultural-automation)

July 28, 2022

COMPONENTS101

July 27, 2022

(/news/programmable-power-management-ic-with-various-protection-functions-optimized-for-automotive-applications)

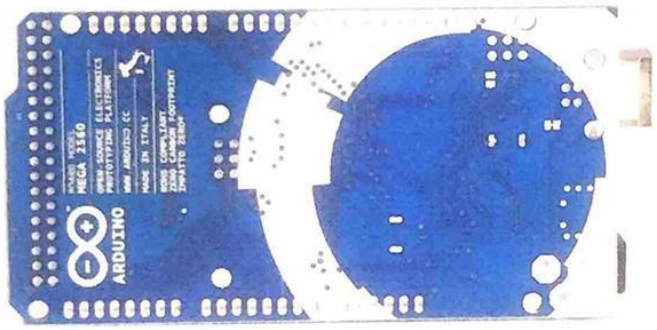
Programmable Power Management IC with Various Protection Functions Optimized for Automotive Applications
(/news/programmable-power-management-ic-with-various-protection-functions-optimized-for-automotive-applications)

July 26, 2022

Components101 is a resource dedicated for electronics design engineers, covering product news, analysis and articles on latest electronics components.

IMPORTANT LINKS

POPULAR TAGS



Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 (datasheet). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Schematic & Reference Design

EAGLE files: arduino-mega2560-reference-design.zip

Schematic: arduino-mega2560-schematic.pdf

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Power

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using pinMode() , digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2).** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- **PWM: 0 to 13.** Provide 8-bit PWM output with the analogWrite() function.
- **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).** These pins support SPI communication using the SPI library. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Uno, Duemilanove and Diecimila.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH

value, the LED is on, when the pin is LOW, it's off.
- **I2C: 20 (SDA) and 21 (SCL).** Support I2C (TWI) communication using the Wire library (documentation on the Wiring website). Note that these pins are not in the same location as the I2C pins on the Duemilanove or Diecimila.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and analogReference() function.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with analogReference().
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).
A SoftwareSerial library allows for serial communication on any of the Mega2560's digital pins.
The ATmega2560 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation on the Wiring website for details. For SPI communication, use the SPI library.

Programming

The Arduino Mega can be programmed with the Arduino software (download). For details, see the reference and tutorials.
The ATmega2560 on the Arduino Mega comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It

communicates using the original STK500 protocol (reference, C header files).
You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see these instructions for details.

Automatic (Software) Reset

Rather then requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.
This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.
The Mega2560 contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

USB Overcurrent Protection

The Arduino Mega2560 has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics and Shield Compatibility

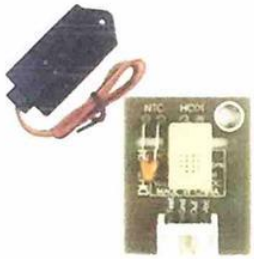
The maximum length and width of the Mega2560 PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega2560 is designed to be compatible with most shields designed for the Uno, Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega2560 and Duemilanove / Diecimila. *Please note that I2C is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5).*

HM2301 Digital-output humidity and temperature sensor

Feature & Application:

- Full range temperature compensated
- Relative humidity and temperature measurement
- Calibrated digital signal
- Outstanding long-term stability
- Extra components not needed
- Long transmission distance
- Low power consumption
- 4 pins packaged and fully interchangeable



Description:

output calibrated digital signal. It utilizes exclusive digital-signal-collecting-technique and humidity sensing technology, assuring its reliability and stability.Its sensing elements is connected with 8-bit single-chip computer.

Every sensor of this model is temperature compensated and calibrated in accurate calibration chamber and the calibration-coefficient is saved in type of programme in OTP memory, when the sensor is detecting, it will cite coefficient from memory.

Small size & low consumption & long transmission distance(20m) enable DHT21 to be suited in all kinds of harsh application occasions.

Single-row packaged with four pins, making the connection very convenient.

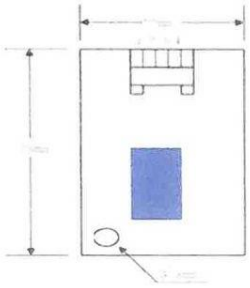
Technical Specification:

Model	HM2301
Power supply	3.3-5V DC
Output signal	digital signal via single-bus
Sensing element	Polymer humidity capacitor

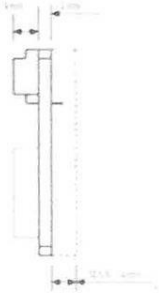
Hanwei
Street Address: No.169 Xuesong Road,National&High Tech Zone,Zhengzhou **Zip:** 450001
Telephone: 86-371-67169080/70 **Fax:** 86-371-67169090
Website: http://www.hwsensor.com **E-mail:** sales@hwsensor.com ; hwsensor@163.com

Measuring range	humidity 0-100%RH; temperature -40 - 80Celsius
Accuracy	humidity +/-3%RH(Max +-5%RH); temperature < +/-1Celsius
Resolution or sensitivity	humidity 0.1%RH; temperature 0.1Celsius
Repeatability	humidity +/-1%RH; temperature +/-0.2Celsius
Humidity hysteresis	+/-0.3%RH
Long-term Stability	+/-0.5%RH/year
Sensing period	Average: 2s
Interchangeability	fully interchangeable
Dimensions	size 22*28*5mm

Dimensions: (unit----mm)



Front view



Side elevation

Electrical connection diagram:

Operating specifications:

Hanwei
Street Address: No.169 Xuesong Road,National&High Tech Zone,Zhengzhou **Zip:** 450001
Telephone: 86-371-67169080/70 **Fax:** 86-371-67169090
Website: http://www.hwsensor.com **E-mail:** sales@hwsensor.com ; hwsensor@163.com

(1) Power and Pins

Power's voltage should be 3.3-5V DC. When power is supplied to sensor, don't send any instruction to the sensor within one second to pass unstable status. One capacitor valued 100nF can be added between VDD and GND for wave filtering.

(2)Communication and signal

Single-bus data is used for communication between MCU and HM2301, it costs 5ms for single one time communication.

Data is comprised of integral and decimal part, the following is the formula for data.

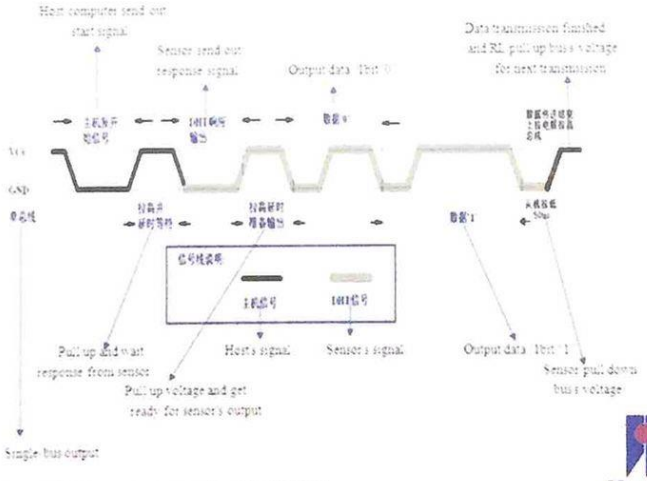
HM2301send out higher data bit firstly!

DATA=8 bit integral RH data+8 bit decimal RH data+8 bit integral T data+8 bit decimal T data+8 bit check-sum

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

When MCU send start signal, HM2301 change from low-power-consumption-mode to running-mode. When MCU finishs sending the start signal, HM2301 will send response signal of 40-bit data that reflect the relative humidity and temperature information to MCU. Without start signal from MCU, DHT21 will not give response signal to MCU. One start signal for one time's response data that reflect the relative humidity and temperature information from HM2301. HM2301 will change to low-power-consumption-mode when data collecting finish if it don't receive start signal from MCU again.

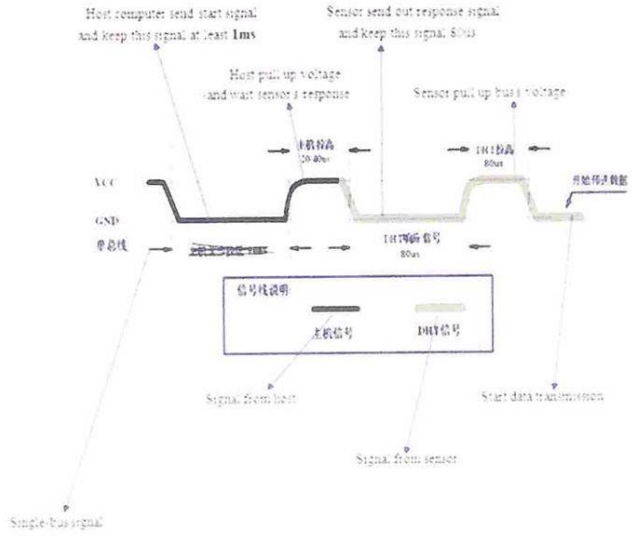
1)Check bellow picture for overall communication process:



Step 1: MCU send out start signal to HM2301

Data-bus's free status is high voltage level. When communication between MCU and DHT21 begin,

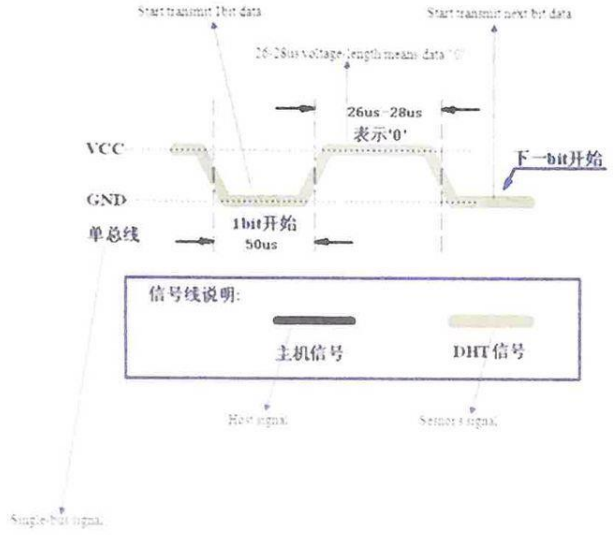
program of MCU will transform data-bus's voltage level from high to low level and this process must beyond at least 1ms to ensure HM2301 could detect MCU's signal, then MCU will wait 20-40us for DHT21's response.



Step 2: HM2301 send response signal to MCU

When HM2301 detect the start signal, DHT21 will send out low-voltage-level signal and this signal last 80us as response signal, then program of DHT21 transform data-bus's voltage level from low to high level and last 80us for HM2301's preparation to send data.

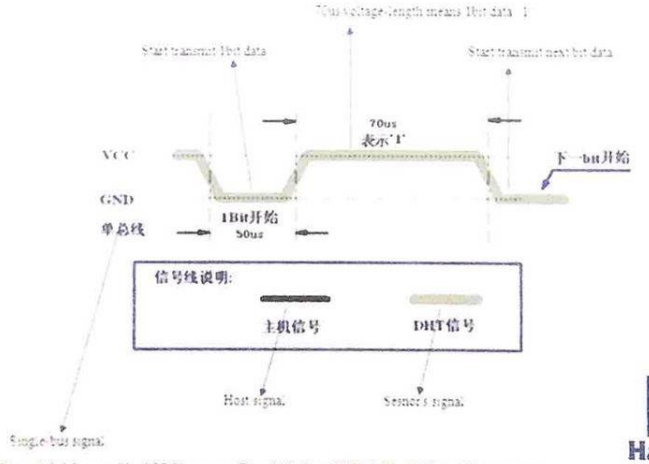
Check bellow picture for step 2:



Step 3: HM2301send data to MCU

When HM2301 is sending data to MCU, every bit's transmission begin with low-voltage-level that last 50us, the following high-voltage-level signal's length decide the bit is "1" or "0".

Check bellow picture for step 3.



If signal from HM2301 is always high-voltage-level, it means HM2301is not working properly, please check the electrical connection status.

Electrical Characteristics:

Item	Condition	Min	Typical	Max	Unit
Power supply	DC	3.3	5	5.5	V
Current supply	Measuring	1.3	1.5	2.1	mA
	Average	0.5	0.8	1.1	mA
Collecting period	Second	1.7		2	Second

- Collecting period should be : >1.7 second.

Attentions of application:

(1) Operating and storage conditions

We don't recommend the applying RH-range beyond the range stated in this specification. The DHT21 sensor can recover after working in non-normal operating condition to calibrated status, but will accelerate sensors' aging.

(2) Attentions to chemical materials

Vapor from chemical materials may interfere HM2301's sensitive-elements and debase V's sensitivity.

(3)Disposal when (1) & (2) happens

Step one: Keep the HM2301sensor at condition of Temperature 50~60Celsius, humidity <10%RH for 2 hours.

Step two: After step one, keep the DHT21 sensor at condition of Temperature 20~30Celsius, humidity >70%RH for 5 hours.

(4)Attention to temperature's affection

Relative humidity strongly depend on temperature, that is why we use temperature compensation technology to ensure accurate measurement of RH. But it's still be much better to keep the sensor at same temperature when sensing.

HM2301 should be mounted at the place as far as possible from parts that may cause change to temperature.

(5)Attentions to light

Long time exposure to strong light and ultraviolet may debase HM2301's performance.

(6)Attentions to connection wires

The connection wires' quality will effect communication's quality and distance, high quality shielding-wire is recommended.

(7)Other attentions

- Welding temperature should be bellow 260Celsius.
- Avoid using the sensor under dew condition.
- Don't use this product in safety or emergency stop devices or any other occasion that failure of HM2301may cause personal injury.

Hanwei

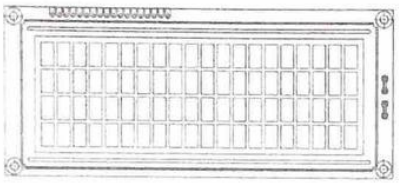
Street Address: No.169 Xuesong Road,National&High Tech Zone,Zhengzhou **Zip:** 450001
Telephone: 86-371-67169080/70 **Fax:** 86-371-67169090
Website: http://www.hwsensor.com **E-mail:** sales@hwsensor.com ; hwsensor@163.com

Fans & Air Purifiers Ventilating Fan

FV-30RUN5

GENERAL	Voltage		220 V
	Frequency		50 Hz
	Weight	Max	39
		Min	31
Power Consumption (W)	Intake	Max	24
		Min	19
Rotation per minute (RPM)	Exhaust	Max	833
		Min	833
	Intake	Max	879
		Min	741
Air Volume (m3/min)		Exhaust	19.2
		Intake	12.5
Noise (dB)		Exhaust	43
		Intake	44

20 x 4 Character LCD



FEATURES

- Type: Character
- Display format: 20 x 4 characters
- Built-in controller: ST 7066 (or equivalent)
- Duty cycle: 1/16
- 5 x 8 dots includes cursor
- + 5 V power supply (also available for + 3 V)
- LED can be driven by pin 1, pin 2, pin 15, pin 16 or A and K
- N.V. optional for + 3 V power supply
- Material categorization: For definitions of compliance please see www.vishay.com/doc?99912

RoHS COMPLIANT

MECHANICAL DATA

ITEM	STANDARD VALUE	UNIT
Module Dimension	146.0 x 62.5	mm
Viewing Area	123.5 x 43.0	
Dot Size	0.92 x 1.10	
Dot Pitch	0.98 x 1.16	
Mounting Hole	139.0 x 55.5	
Character Size	4.84 x 9.22	

ABSOLUTE MAXIMUM RATINGS

ITEM	SYMBOL	STANDARD VALUE			UNIT
		MIN.	TYP.	MAX.	
Power Supply	V_{DD} to V_{SS}	- 0.3	-	7.0	V
Input Voltage	V_I	- 0.3	-	V_{DD}	

Note

- V_{SS} = 0 V, V_{DD} = 5.0 V

ELECTRICAL CHARACTERISTICS

ITEM	SYMBOL	CONDITION	STANDARD VALUE			UNIT
			MIN.	TYP.	MAX.	
Input Voltage	V_{DD}	V_{DD} = + 5 V	4.7	5.0	5.3	V
		V_{DD} = + 3 V	2.7	3.0	5.3	
Supply Current	I_{DD}	V_{DD} = + 5 V	-	8.0	10.0	mA
Recommended LC Driving Voltage for Normal Temperature Version Module	V_{DD} to V_0	- 20 °C	5.0	5.1	5.7	V
		0 °C	4.6	4.8	5.2	
		25 °C	4.1	4.5	4.7	
		50 °C	3.9	4.2	4.5	
		70 °C	3.7	3.9	4.3	
LED Forward Voltage	V_F	25 °C	-	4.2	4.6	V
LED Forward Current	I_F	25 °C	-	540	1080	mA
EL Power Supply Current	I_{EL}	V_{EL} = 110 V_{AC}, 400 Hz	-	-	5.0	mA

OPTIONS

PROCESS COLOR						BACKLIGHT			
TN	STN Gray	STN Yellow	STN Blue	FSTN B&W	STN Color	None	LED	EL	CCFL
x	x	x	x	x		x	x	x	

For detailed information, please see the "Product Numbering System" document.

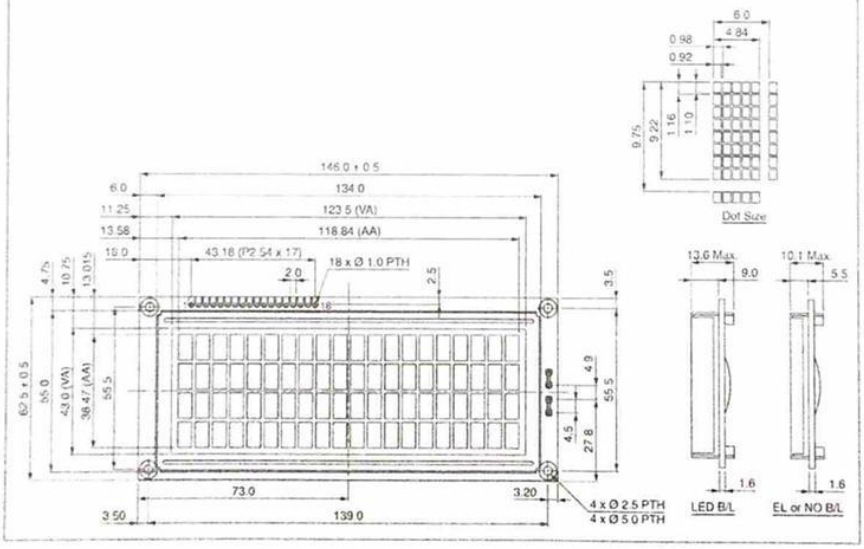
DISPLAY CHARACTER ADDRESS CODE

Display Position

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
DD RAM Address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
DD RAM Address	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53
DD RAM Address	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27
DD RAM Address	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67

INTERFACE PIN FUNCTION		
PIN NO.	SYMBOL	FUNCTION
1	V_{SS}	Ground
2	V_{DD}	+ 3 V or + 5 V
3	V_O	Contrast adjustment
4	RS	H/L register select signal
5	R/W	H/L read/write signal
6	E	H → L enable signal
7	DB0	H/L data bus line
8	DB1	H/L data bus line
9	DB2	H/L data bus line
10	DB3	H/L data bus line
11	DB4	H/L data bus line
12	DB5	H/L data bus line
13	DB6	H/L data bus line
14	DB7	H/L data bus line
15	A	Power supply for LED (4.2 V)
16	K	Power supply for B/L (0 V)
17	NC/V_{EE}	NC or negative voltage output
18	NC	NC connection

DIMENSIONS in millimeters



Disclaimer

ALL PRODUCT, PRODUCT SPECIFICATIONS AND DATA ARE SUBJECT TO CHANGE WITHOUT NOTICE TO IMPROVE RELIABILITY, FUNCTION OR DESIGN OR OTHERWISE.

Vishay Intertechnology, Inc., its affiliates, agents, and employees, and all persons acting on its or their behalf (collectively, "Vishay"), disclaim any and all liability for any errors, inaccuracies or incompleteness contained in any datasheet or in any other disclosure relating to any product.

Vishay makes no warranty, representation or guarantee regarding the suitability of the products for any particular purpose or the continuing production of any product. To the maximum extent permitted by applicable law, Vishay disclaims (i) any and all liability arising out of the application or use of any product, (ii) any and all liability, including without limitation special, consequential or incidental damages, and (iii) any and all implied warranties, including warranties of fitness for particular purpose, non-infringement and merchantability.

Statements regarding the suitability of products for certain types of applications are based on Vishay's knowledge of typical requirements that are often placed on Vishay products in generic applications. Such statements are not binding statements about the suitability of products for a particular application. It is the customer's responsibility to validate that a particular product with the properties described in the product specification is suitable for use in a particular application. Parameters provided in datasheets and / or specifications may vary in different applications and performance may vary over time. All operating parameters, including typical parameters, must be validated for each customer application by the customer's technical experts. Product specifications do not expand or otherwise modify Vishay's terms and conditions of purchase, including but not limited to the warranty expressed therein.

Hyperlinks included in this datasheet may direct users to third-party websites. These links are provided as a convenience and for informational purposes only. Inclusion of these hyperlinks does not constitute an endorsement or an approval by Vishay of any of the products, services or opinions of the corporation, organization or individual associated with the third-party website. Vishay disclaims any and all liability and bears no responsibility for the accuracy, legality or content of the third-party website or for that of subsequent links.

Except as expressly indicated in writing, Vishay products are not designed for use in medical, life-saving, or life-sustaining applications or for any other application in which the failure of the Vishay product could result in personal injury or death. Customers using or selling Vishay products not expressly indicated for use in such applications do so at their own risk. Please contact authorized Vishay personnel to obtain written terms and conditions regarding products designed for such applications.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document or by any conduct of Vishay. Product names and markings noted herein may be trademarks of their respective owners.

4 channel 12V Relay Module

Introduction:

The 4 channel Relay Board can be used to turn lights, fans and other devices on/off while keeping them isolated from your microcontroller.

Relay's coil voltage is 12V, so you can use directly with any 12V regulator. (We suggest separate regulators between your Arduino and Relay board).
This Relay Board allows you to control high-power devices (up to 10 A) via the on-board relay.
Control of the relay is provided via a 1 x 3 header – friendly to servo cables and convenient to connect to many development boards.

Key Features:

Control high-power devices up to 10 A with a simple high/low signal Provides isolation between the microcontroller and device being controlled Screw terminals for relay connections 3-pin servo-style header for power/signal interface LED indicator provides visual feedback

Specification:

Control signal: TTL voltage
Active at HIGH level
Rated load: 10A 250VAC / 10A 30DC

SIM900
GSM/GPRS Module

The SIM900 is a complete Quad-band GSM/GPRS solution in a SMT module which can be embedded in the customer applications.

Featuring an industry-standard interface, the SIM900 delivers GSM/GPRS 850/900/1800/1900MHz performance for voice, SMS, Data, and Fax in a small form factor and with low power consumption. With a tiny configuration of 24mm x 24mm x 3 mm, SIM900 can fit almost all the space requirements in your M2M application, especially for slim and compact demand of design.

- SIM900 is designed with a very powerful single-chip processor integrating AMR926EJ-S core
- Quad - band GSM/GPRS module with a size of 24mmx24mmx3mm
- SMT type suit for customer application
- An embedded Powerful TCP/IP protocol stack
- Based upon mature and field-proven platform, backed up by our support service, from definition to design and production

General featrues

- Quad-Band 850/ 900/ 1800/ 1900 MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 2/2+
  - Class 4 (2 W @850/ 900 MHz)
  - Class 1 (1 W @ 1800/1900MHz)
- Dimensions: 24* 24 * 3 mm
- Weight: 3.4g
- Control via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)
- SIM application toolkit
- Supply voltage range 3.4 ... 4.5 V
- Low power consumption
- Operation temperature: -30 °C to +80 °C

Specifications for fax

- Group 3, class 1

Specifications for data

- GPRS class 10: max. 85.6 kbps (downlink)
- PBCCH support
- Coding schemes CS 1, 2, 3, 4
- CSD up to 14.4 kbps
- USSD
- Non transparent mode
- PPP-stack

Specifications for SMS via GSM / GPRS

- Point-to-point MO and MT
- SMS cell broadcast
- Text and PDU mode

Drivers

- MUX Driver

Specifications for voice

- Tncodec
  - Half rate (HR)
  - Full rate (FR)
  - Enhanced Full rate (EFR)
- Hands-free operation (Echo suppression)
- AMR
  Half Rate(HR)
  Full Rate(FR)

Interfaces

- Interface to external SIM 3V/ 1.8V
- analog audio interface
- RTC backup
- SPI interface
- Serial interface
- Antenna pad
- I2C
- GPIO
- PWM
- ADC

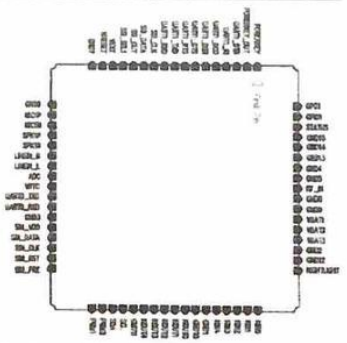
Compatibility

- AT cellular command interface

Approvals (in planning)

- CE
- FCC
- ROHS
- PTCRB
- GCF
- AT&T
- IC
- TA

Pin Assignment



More about SIM900 module, Please contact: Tel:+86 21 32523300
Fax:+86 21 32523301
Email:simcom@sim.com