**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH**

------------------------------

**BÁO CÁO ĐỀ TÀI**

**XÂY DỰNG CÁC HỆ THỐNG NHÚNG**

***Đề tài : “Hệ thống chăm sóc cây trồng tự động”***

**Sinh viên thực hiện :**

**Nguyễn Vũ Huy – N112104037**

**Nguyễn Hồng Phấn – N112104047**

**Diệp Thời Quyền– N112104056**

**GVHD : Th.s Huỳnh Trọng Thưa**

**Thành phố Hồ Chí Minh**

**Năm 2015**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Công việc | Huy | Phấn | Quyền |
| 1 | Tìm hiểu tính chất môi trường trồng cây và các loại cảm biến tương ứng |  | X |  |
| 2 | Thiết kế hệ thống + lựa chọn linh kiện (Cảm biến) |  |  | X |
| 3 | Thiết kế mạch phần cứng | X |  |  |
| 4 | Phân tích và vẽ sơ đồ nguyên lý hoạt động |  |  | X |
| 5 | Phân tích và vẽ sơ đồ tương tác phần mềm |  | X |  |
| 6 | Giải thuật phần mềm | X | X | X |
| 7 | Cài đặt hệ thống | X |  |  |
| 8 | Kiểm thử | X | X | X |

BẢNG PHÂN CHIA CÔNG VIỆC

**SOURCE CODE**

1. **CODE XỬ LÝ CHÍNH**

int h= A2; // khai báo chân Analog 2 là chân tín hiệu độ ẩm

#include <LiquidCrystal.h>

#include <Wire.h>

#include "DHT.h"

#define DHTPIN 2

#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

#define den 13

#define den1 12

#define IN 9

#define IN1 10

LiquidCrystal lcd(8, 7, 6, 5, 4, 3);

void setup()

{

Serial.begin(9600);

lcd.begin(16, 2);

dht.begin();

//Khai báo các chân

pinMode(den,OUTPUT);

pinMode(den1,OUTPUT);

pinMode(IN, OUTPUT);

pinMode(IN1, OUTPUT);

pinMode(IN2, OUTPUT);

digitalWrite(IN, LOW);

digitalWrite(IN1, LOW);

digitalWrite(IN2, LOW);

}

void loop()

{

// Đọc và tính toán độ ẩm đất

float a=analogRead(h); // Đọc giá trị độ ẩm đất từ chân A2

float doamdat = map(a, 0, 1023, 100, 0);

// map(a, A1, A2, B1, B2) là hàm chuyển đơn vị từ thang đo A1-A2 sang thang đo B1-B2

long map(long x, long in\_min, long in\_max, long out\_min, long out\_max)

{

return (x - in\_min) \* (out\_max - out\_min) / (in\_max - in\_min) + out\_min;

}

ở đây là từ thang đo Analog 0-1023 sang 0-100 percent

// giá trị độ ẩm và nhiệt độ môi trường được đọc qua chân Digital 2

// xử lý bằng các hàm read() của thư viện DHT11 được cung cấp sẵn, sẽ giải thích hàm ở dưới

float h = dht.readHumidity();

float t = dht.readTemperature();

//In ra LCD các giá trị độ ẩm, nhiệt độ

lcd.setCursor(0, 0);

lcd.print(doamdat);

lcd.print("%");

lcd.print(" ");

lcd.print(h);

lcd.print("%");

lcd.setCursor(5, 2);

lcd.print(t);

lcd.print("\*C");

// in ra màn hình Serial (Để kiểm chứng với LCD)

Serial.print("Do am la:");

Serial.println(doamdat);//In ra độ ẩm đất

Serial.print("DoAm: ");

Serial.print(h);

Serial.print(" %\t");

Serial.print("NhietDo: ");

Serial.print(t);

Serial.println(" \*C");

//Khối lệnh xử lý độ ẩm đất (Tự đặt theo kinh nghiệm thực tế)

//Bơm nước khi đất khô (độ ẩm dưới 35% )

if(doamdat<35)

{

Serial.println("Thieu nuoc, can tuoi cay");

digitalWrite(den,HIGH);

digitalWrite(IN, HIGH);

}

else if(doamdat>60)

{

Serial.println("Thua nuoc");

digitalWrite(IN, LOW);

}

else

{

Serial.println("Du nuoc");

digitalWrite(IN, LOW);

}

//Khối lệnh xử lý độ nhiệt độ môi trường (Tự đặt theo kinh nghiệm thực tế)

if (t>30){

Serial.println("Nhietdocao");

digitalWrite(IN1, HIGH);

}

else if (t<30)

{

Serial.println("Mat");

digitalWrite(IN1, LOW);

}

//Khối lệnh xử lý độ ẩm không khí (Tự đặt theo kinh nghiệm thực tế)

if (h<40){

Serial.println("Do am thap");

digitalWrite(IN2, HIGH);

}

else if (h>40)

{

Serial.println("Du am");

digitalWrite(IN2, LOW);

}

Serial.println();

Serial.println();

delay (50000); // Delay 1 phút

}

1. **THƯ VIỆN CỦA CÁC LINH KIỆN TRONG HỆ THỐNG**
2. LCD 1602– Có sẵn khi cài đặt Arduno IDE

* LiquidCrystal.cpp
* LiquidCrystal.h

1. DHT11 Semperature Sensor - Có sẵn khi cài đặt Arduno IDE

* DHT11.cpp
* Hàm đọc và tính nhiệt độ

float DHT::readTemperature(bool S) {

float f;

if (read()) {

switch (\_type) {

case DHT11:

f = data[2];

if(S)

f = convertCtoF(f);

return f;

case DHT22:

case DHT21:

f = data[2] & 0x7F;

f \*= 256;

f += data[3];

f /= 10;

if (data[2] & 0x80)

f \*= -1;

if(S)

f = convertCtoF(f);

return f;

}

}

return NAN;

}

float DHT::convertCtoF(float c) {

return c \* 9 / 5 + 32;

}

float DHT::convertFtoC(float f) {

return (f - 32) \* 5 / 9;

}

* Hàm đọc và tính toán độ ẩm

float DHT::readHumidity(void) {

float f;

if (read()) {

switch (\_type) {

case DHT11:

f = data[0];

return f;

case DHT22:

case DHT21:

f = data[0];

f \*= 256;

f += data[1];

f /= 10;

return f;

}

}

return NAN;

}

* DHT11.h