

Sau một thời gian đi quảng cáo spam khắp nơi thì mình thấy hệ thống nông nghiệp thông minh được các bạn ủng hộ khá nhiều, nhưng với một số bạn không có TIVA mà lại sẵn Arduino thì quả là vất vả, giờ lại phải đi mua kit TIVA mà không có nhu cầu dùng tới thì hơi tốn kém. Do đó thì mình phải thay đổi code một chút để nó phù hợp hơn với Arduino, hi vọng nó là cái để mọi người có thể tham khảo và xây dựng riêng một hệ thống nông nghiệp thông minh đơn giản cho riêng mình.

Trong bài viết này mình chỉ tập trung vào chương trình và một số thay đổi, về phần giải thích thêm bớt các bạn có thể coi ở bài viết [Hệ thống nông nghiệp thông minh](#) đơn giản nhé.

Bài viết này nằm trong serie về Nông nghiệp thông minh

- [Hệ thống nông nghiệp thông minh phiên bản Arduino](#)
- [Hệ thống nông nghiệp thông minh phiên bản ESP8266](#)
- [Hệ thống nông nghiệp thông minh phiên bản ARM TIVA](#)

## Chuẩn bị

### Phần cứng

- Arduino
- ESP8266 v1 hoặc v7/v12 (mình dùng v1 có sẵn)
- Cảm biến nhiệt độ, độ ẩm DHT22
- Cảm biến ánh sáng (mình tự chế)
- Modul Relay (2 relay)
- LCD 16×2
- Nút nhấn và LED (mình tận dụng có sẵn trên kit TIVA C)
- Dây nối
- Nguồn 3.3V
- Testboard (Breadboard)

### Phần mềm

- Arduino IDE
- Thư viện [DHT22](#)

## DHT22

Mình thử kiểm tra DHT22 hoạt động xem sao, kết nối dây như hình dưới



Chương trình đọc nhiệt độ, độ ẩm hiển thị lên máy tính với tốc độ baud là 115200

```
// Chương trình đọc nhiệt độ, độ ẩm từ cảm biến DHT
// Written by ladyada, public domain
// Chỉnh sửa cho Arduino bởi hocARM.org
```

```
#include "DHT.h"

#define DHTPIN 2      // Chân DATA nối với chân 2

// Uncomment loại cảm biến bạn sử dụng, nếu DHT11 thì uncomment DHT11
// và comment DHT22
// #define DHTTYPE DHT11    // DHT 11
#define DHTTYPE DHT22    // DHT 22 (AM2302), AM2321
// #define DHTTYPE DHT21    // DHT 21 (AM2301)

// Kết nối
// DHT      | Arduino
// -----
// VCC(1)    | 5V
// DATA(2)   | 2
// NC(3)      | x
// GND(4)     | GND

// Kết nối chân 1 của DHT với 3.3V
// Chân 2 kết nối với các chân nào của TIVA C
// Chân 4 nối với GND
// Nối trở 10k giữa chân 1 và chân 2

// Khởi tạo cảm biến
DHT dht(DHTPIN, DHTTYPE);

void setup() {
    // Khởi tạo cổng serial baud 115200
    Serial.begin(115200);
    Serial.println("DHTxx test!");
    // Bắt đầu đọc dữ liệu
    dht.begin();
}

void loop() {
    // Đợi chuyển đổi dữ liệu khoảng 2s
    delay(2000);

    float h = dht.readHumidity();
    // Đọc giá trị nhiệt độ C (mặc định)
    float t = dht.readTemperature();
    // Đọc giá trị nhiệt độ F (isFahrenheit = true)
    float f = dht.readTemperature(true);

    // Kiểm tra quá trình đọc thành công hay không
    if (isnan(h) || isnan(t) || isnan(f)) {
```

```
Serial.println("Failed to read from DHT sensor!");
return;
}

// Tính chỉ số nhiệt độ F (mặc định)
// float hif = dht.computeHeatIndex(f, h);
// Tính chỉ số nhiệt độ C (isFahrenheit = false)
// float hic = dht.computeHeatIndex(t, h, false);

// IN thông tin ra màn hình
Serial.print("Do am: ");
Serial.print(h);
Serial.print(" %t");
Serial.print("Nhiệt độ: ");
Serial.print(t);
Serial.println(" *C ");

}
```

## Cảm biến ánh sáng

Mình vẫn giữ kết nối như cũ, thêm vào cảm biến ánh sáng với kết nối như hình



Chương trình đọc đồng thời cảm biến ánh sáng và DHT22, ở đây do Arduino ADC có 10 bit nên độ phân giải chỉ là 1024(0-1023)

```
// Chương trình đọc nhiệt độ, độ ẩm từ cảm biến DHT
// Thêm chức năng đọc cảm biến ánh sáng
// Written by ladyada, public domain
// Chọn số cho Arduino Uno bởi hocARM.org

#include "DHT.h"

#define DHTPIN 2      // Chân DATA nối với 2
#define LDR_PIN A0    // Chân A0 nối với chân OUT cảm biến as
// Uncomment loại cảm biến bạn sử dụng, nếu DHT11 thì uncomment DHT11
// và comment DHT22
// #define DHTTYPE DHT11 // DHT 11
#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
// #define DHTTYPE DHT21 // DHT 21 (AM2301)

// Kết nối
// DHT      | Arduino
// -----
// VCC(1)    | 5V
```

```
// DATA(2)    |    2
// NC(3)       |    x
// GND(4)      |    GND
// Nối trỏ 10k giữa chân 1 và chân 2

// Khởi tạo cảm biến
DHT dht(DHTPIN, DHTTYPE);

void setup() {
    // Khởi tạo cổng serial baud 115200
    Serial.begin(115200);
    Serial.println("DHTxx test!");
    // Bật đầu đọc dữ liệu
    dht.begin();
}

void loop() {
    // Đợi chuyển đổi dữ liệu khoảng 2s
    delay(2000);

    float h = dht.readHumidity();
    // Đọc giá trị nhiệt độ C (mặc định)
    float t = dht.readTemperature();
    // Đọc giá trị nhiệt độ F (isFahrenheit = true)
    float f = dht.readTemperature(true);

    // Kiểm tra quá trình đọc thành công hay không
    if (isnan(h) || isnan(t) || isnan(f)) {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }

    // Tính chỉ số nhiệt độ F (mặc định)
    // float hif = dht.computeHeatIndex(f, h);
    // Tính chỉ số nhiệt độ C (isFahrenheit = false)
    // float hic = dht.computeHeatIndex(t, h, false);
    // Đọc dữ liệu cảm biến ánh sáng
    int lumen = getLumen(LDR_PIN);
    // IN thông tin ra màn hình
    Serial.print("Đo am: ");
    Serial.print(h);
    Serial.print(" %t");
    Serial.print("Nhiệt độ: ");
    Serial.print(t);
    Serial.print(" *Ct");
    Serial.print("Ánh sáng: ");
```

```
Serial.print(lumen);
Serial.println(" %");
}
int getLumen(int anaPin)
{
    int anaValue = 0;
    for (int i = 0; i < 10; i++) // Đọc giá trị cảm biến 10 lần và lấy
    giá trị trung bình
    {
        anaValue += analogRead(anaPin);
        delay(50);
    }

    anaValue = anaValue / 10;
    anaValue = map(anaValue, 1023, 0, 0, 100); //Tỉ lệ:0 ==> Sáng 100%

    return anaValue;
}
```

## Cảm biến độ ẩm đất

Cảm biến độ ẩm này mình chọn hình mình bạn hơi khác một chút về thứ tự so với bản thực, nhưng cách kết nối thì khá dễ, chân A0 của cảm biến nối với chân A1 của Arduino, 2 chân còn lại ta cấp nguồn. Bạn có thể xem thêm trong code.

Lưu ý các kết nối chân ở trên vẫn giữ như cũ nhé.



Chương trình đọc DHT22, cảm biến ánh sáng, độ ẩm đất

```
// Chương trình đọc nhiệt độ, độ ẩm từ cảm biến DHT
// Thêm chức năng đọc cảm biến ánh sáng
// Thêm chức năng đọc cảm biến độ ẩm đất
// Written by ladyada, public domain
// Chuyển sửa cho Arduino Uno bởi hocARM.org
// Kết nối
// DHT      | Arduino Uno
//-----
// VCC(1)    | 5V
// DATA(2)   | 2
// NC(3)      | x
// GND(4)     | GND
// Nối trở 10k giữa chân 1 và chân 2
//-----
//Cảm biến độ ẩm | Arduino
//-----
```

```
// VCC(1)      | 5V
// GND(2)      | GND
// D0(3)        | x
// A0(4)        | A1
// Cảm biến ánh sáng nối chân A0

#include "DHT.h"

#define DHTPIN 2      // Chân DATA nối với 2
#define LDR_PIN A0    // Chân A0 nối với chân OUT cảm biến as
#define SOIL_MOIST_1_PIN A1 // Chân A1 nối với cảm biến độ ẩm

// Uncomment loại cảm biến bạn sử dụng, nếu DHT11 thì uncomment DHT11
// và comment DHT22
// #define DHTTYPE DHT11 // DHT 11
#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
// #define DHTTYPE DHT21 // DHT 21 (AM2301)
int humDHT;
int tempDHT;
int lumen;
int soilMoist;
// Khởi tạo cảm biến
DHT dht(DHTPIN, DHTTYPE);

void setup() {
    // Khởi tạo cổng serial baud 115200
    Serial.begin(115200);
    Serial.println("DHTxx test!");
    // Bật đọc dữ liệu
    dht.begin();
}

void loop() {
    readSensors();
    // IN thông tin ra màn hình
    Serial.print("Do am: ");
    Serial.print(humDHT);
    Serial.print(" %t");
    Serial.print("Nhiệt độ: ");
    Serial.print(tempDHT);
    Serial.print(" *Ct");
    Serial.print("Ánh sáng: ");
    Serial.print(lumen);
    Serial.print(" %t");
    Serial.print("Do ẩm đất: ");
    Serial.print(soilMoist);
```

```
Serial.println(" %");
}
int getLumen(int anaPin)
{
    int anaValue = 0;
    for (int i = 0; i < 10; i++) // Đọc giá trị cảm biến 10 lần và lấy
    giá trị trung bình
    {
        anaValue += analogRead(anaPin);
        delay(50);
    }

    anaValue = anaValue / 10;
    anaValue = map(anaValue, 1023, 0, 0, 100); //Tí:0 ==> Sáng 100%

    return anaValue;
}
int getSoilMoist()
{
    int i = 0;
    int anaValue = 0;
    for (i = 0; i < 10; i++) //
    {
        anaValue += analogRead(SOIL_MOIST_1_PIN); //Đọc giá trị cảm biến
    độ ẩm đất
        delay(50); // Đợi đọc giá trị ADC
    }
    anaValue = anaValue / (i);
    anaValue = map(anaValue, 1023, 0, 0, 100); //Ít nước:0% ==> Nhiều
    nước 100%
    return anaValue;
}
void readSensors(void)
{
    tempDHT = dht.readTemperature(); //Đọc nhiệt độ DHT22
    humDHT = dht.readHumidity(); //Đọc độ ẩm DHT22
    lumen = getLumen(LDR_PIN); //Đọc ánh sáng
    soilMoist = getSoilMoist(); //Đọc cảm biến độ ẩm đất
}
```

## LCD 16×2

Đây là điểm khác biệt so với phiên bản TIVA, mình thay LCD này vì có 2 lý do

- Phổ biến: dễ mua và dễ tìm.
- Thư viện LCD Gphone sau khi build đã ăn mất 50% RAM, thêm linh tinh các chương

Tài liệu này được tải từ website: <http://linhkienthaomay.com>. Zalo hỗ trợ: 0389937723

trình khác nữa thì hết sạch RAM và chương trình không chạy được.

Tuy nhiên dùng LCD16x2 có nhược điểm là chỉ có 2 dòng hiển thị, trong khi đó thông tin mình cần hiển thị cần ít nhất 6 dòng (nhiệt độ, độ ẩm, ánh sáng, trạng thái 2 bơm), không sao cả mình khắc phục bằng cách hiển thị 2 dòng nhiệt độ, độ ẩm trước, sau đó là ánh sáng và độ ẩm đất

Đầu tiên là kết nối



Mình thử nạp chương trình và in dòng chữ hocarm.org xuống LCD

```
// Thêm thư viện có sẵn LCD
#include <LiquidCrystal.h>

//Khởi tạo với các chân
LiquidCrystal lcd(9, 8, 7, 6, 5, 4);

void setup() {
  //Khai báo LCD 16x2
  lcd.begin(16, 2);
  //In ra màn hình HocARM.org
  lcd.print("HocARM.org!");
}

void loop() {
  // đặt con trỏ vào cột 0, dòng 1
  lcd.setCursor(0, 1);
  // In ra dòng chữ Hello
  lcd.print(" Hello");
}
```

Lưu ý nhỏ là nếu LCD không hiển thị thì bạn xoay từ từ biến trở sẽ hiện màn hình

Chương trình hiển thị các thông tin của cảm biến lên LCD

```
// Chương trình đọc nhiệt độ, độ ẩm từ cảm biến DHT
// Thêm chức năng đọc cảm biến ánh sáng
// Thêm chức năng đọc cảm biến độ ẩm đất
// Chọn số chân cho Arduino Uno bởi hocARM.org
// Kết nối
// DHT          | Arduino Uno
//-----
```



```
// VCC(1)      | 5V
// DATA(2)    | 2
// NC(3)       | x
// GND(4)      | GND
// Nối trỏ 10k giữa chân 1 và chân 2
//-----
//Cảm biến độ ẩm | Arduino
//-----
// VCC(1)      | 5V
// GND(2)      | GND
// D0(3)       | x
// A0(4)       | A1
// Cảm biến ánh sáng nối chân A0

#include "DHT.h"
//#include "homephone.h"
#include <LiquidCrystal.h>

#define DHTPIN 2    // Chân DATA nối với PD0
#define LDR_PIN A0  // Chân PE3 nối với chân OUT cảm biến as
#define SOIL_MOIST_1_PIN A1 // Chân PE4 nối với cảm biến độ ẩm

// Uncomment loại cảm biến bạn sử dụng, nếu DHT11 thì uncomment DHT11
// và comment DHT22
//#define DHTTYPE DHT11    // DHT 11
#define DHTTYPE DHT22    // DHT 22 (AM2302), AM2321
//#define DHTTYPE DHT21    // DHT 21 (AM2301)

int humDHT;
int tempDHT;
int lumen;
int soilMoist;
// Khởi tạo cảm biến
DHT dht(DHTPIN, DHTTYPE);
// Khởi tạo LCD
LiquidCrystal lcd(9, 8, 7, 6, 5, 4);

void setup() {
    // Khởi tạo cổng serial baud 115200
    Serial.begin(115200);
    Serial.println("HocARM NDTR Bot!");
    // Bật đọc dữ liệu
    dht.begin();
    //Thông báo đây là LCD 1602
    lcd.begin(16, 2);
}
```

```
//In ra màn hình lcd dòng chữ
lcd.print("HocARM NDTR Bot!*");
lcd.setCursor(0, 1);
lcd.print("Xin chao ong chu");
}

void loop() {
    delay(2000);
    readSensors();
    printData();
    showDataLCD();
}

int getLumen(int anaPin)
{
    int anaValue = 0;
    for (int i = 0; i < 10; i++) // Đọc giá trị cảm biến 10 lần và lấy
    giá trị trung bình
    {
        anaValue += analogRead(anaPin);
        delay(50);
    }

    anaValue = anaValue / 10;
    anaValue = map(anaValue, 1023, 0, 0, 100); //Tối:0 ==> Sáng 100%

    return anaValue;
}

int getSoilMoist()
{
    int i = 0;
    int anaValue = 0;
    for (i = 0; i < 10; i++) //
    {
        anaValue += analogRead(SOIL_MOIST_1_PIN); //Đọc giá trị cảm biến
    độ ẩm đất
        delay(50); // Đợi đọc giá trị ADC
    }
    anaValue = anaValue / (i);
    anaValue = map(anaValue, 1023, 0, 0, 100); //Ít nước:0% ==> Nhiều
    nước 100%
    return anaValue;
}

void readSensors(void)
{
    tempDHT = dht.readTemperature(); //Đọc nhiệt độ DHT22
    humDHT = dht.readHumidity(); //Đọc độ ẩm DHT22
```

```
    lumen = getLumen(LDR_PIN);          //Đọc ánh sáng
    soilMoist = getSoilMoist();          //Đọc cảm biến độ ẩm đất
}
void showDataLCD(void)
{
    lcd.clear();                        //Xóa màn hình
    lcd.setCursor(0, 1);
    lcd.print(" DO.AM% = ");
    lcd.print(humDHT);
    lcd.println(" % " );

    lcd.setCursor(1, 0);
    lcd.print(" NH.DO = ");
    lcd.print(tempDHT);
    lcd.println(" *C ");
    delay(5000);
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print(" AM.DAT% = ");
    lcd.print(soilMoist);
    lcd.println(" % " );

    lcd.setCursor(1, 0);
    lcd.print("A.SANG% = ");
    lcd.print(lumen);
    lcd.println(" % ");
}
void printData(void)
{
    // IN thông tin ra màn hình
    Serial.print("Do am: ");
    Serial.print(humDHT);
    Serial.print(" %t");
    Serial.print("Nhiệt độ: ");
    Serial.print(tempDHT);
    Serial.print(" *Ct");
    Serial.print("Ánh sáng: ");
    Serial.print(lumen);
    Serial.print(" %t");
    Serial.print("Độ ẩm đất: ");
    Serial.print(soilMoist);
    Serial.println(" %");
}
```

## Relay và nút nhấn

Do Arduino không có 2 nút nhấn có sẵn như TIVA nên mình phải thêm 2 nút nhấn gắn ngoài vào, thêm một lưu ý nữa là nếu relay dùng nguồn ngoài thì cần nối chung GND với Arduino, kết nối vẫn giữ nguyên kết nối bên trên và thêm vào relay và nút nhấn nhé.



Giờ thì thêm chương trình đọc nút nhấn và điều khiển relay cho chế độ thủ công

```
/* Chương trình đọc nhiệt độ, độ ẩm từ cảm biến DHT
  Thêm chức năng đọc cảm biến ánh sáng
  Thêm chức năng đọc cảm biến độ ẩm đất
  Thêm hiển thị LCD
  Thêm chức năng điểu khiển tưới tiêu bằng tay 2 bơm
  HocARM NDTR BOT for Arduino by hocARM.org
  -----
  // Chương trình đọc nhiệt độ, độ ẩm từ cảm biến DHT
  // Thêm chức năng đọc cảm biến ánh sáng
  // Thêm chức năng đọc cảm biến độ ẩm đất
  // Chọn sạ cho Arduino Uno bởi hocARM.org
  // Kết nối
  // DHT          | Arduino Uno
  //-----
  // VCC(1)       | 5V
  // DATA(2)     | 2
  // NC(3)        | x
  // GND(4)       | GND
  // Nối trỏ 10k giữa chân 1 và chân 2
  //-----
  //Cảm biến độ ẩm | Arduino
  //-----
  // VCC(1)       | 5V
  // GND(2)       | GND
  // D0(3)        | x
  // A0(4)        | A1
  // Cảm biến ánh sáng nối chân A0
  //-----
  // 2 nút nhấn sẽ được kết nối với chân A2 và A3
  // 2 relay nối với chân 3 và chân 12
  */
#include "DHT.h"
//#include "homephone.h"
#include <LiquidCrystal.h>

#define DHTPIN 2      // Chân DATA nối với PD0
```

```
#define LDR_PIN A0      // Chân PE3 nối với chân OUT cảm biến as
#define SOIL_MOIST_1_PIN A1 // Chân PE4 nối với cảm biến độ ẩm
// Relay, nút nhấn
#define PUMPW_ON A2     //Nút có sọc trên kit
#define PUMPW_PIN 3
#define PUMPS_ON A3     //Nút có sọc trên kit
#define PUMPS_PIN 12
// Uncomment loại cảm biến bạn sử dụng, nếu DHT11 thì uncomment DHT11
// và comment DHT22
// #define DHTTYPE DHT11    // DHT 11
#define DHTTYPE DHT22      // DHT 22  (AM2302), AM2321
// #define DHTTYPE DHT21    // DHT 21 (AM2301)

// Biến lưu các giá trị cảm biến
int humDHT;
int tempDHT;
int lumen;
int soilMoist;
// Biến lưu trạng thái bơm
boolean pumpWaterStatus = 0;
boolean pumpPesStatus = 0;
int timePumpOn = 10; // Thời gian bật bơm nước
// Biến cho timer
long sampleTimingSeconds = 50; // ==> Thời gian đọc cảm biến (s)
long startTiming = 0;
long elapsedTime = 0;
// Khởi tạo cảm biến
DHT dht(DHTPIN, DHTTYPE);
// Khởi tạo LCD
// homephone lcd (PD_1, PD_2, PD_3, PE_1, PE_2);
LiquidCrystal lcd(9, 8, 7, 6, 5, 4);

void setup() {
    pinMode(PUMPW_PIN, OUTPUT);
    pinMode(PUMPS_PIN, OUTPUT);
    pinMode(PUMPW_ON, INPUT_PULLUP); // Button
    pinMode(PUMPS_ON, INPUT_PULLUP); // Button
    aplyCmd();
    // Khởi tạo cổng serial baud 115200
    Serial.begin(115200);
    Serial.println("HocARM NDTR Bot!");
    // Bật đầu đọc dữ liệu
    dht.begin();
    lcd.begin(16, 2);
    lcd.print("HocARM NDTR Bot!*");
    lcd.setCursor(0, 1);
```

```
    lcd.print("Xin chao ong chu");
    readSensors(); // Khi tạo đọc cảm biến
    startTiming = millis(); // Bật đầu đếm thời gian
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print("Vui long cho ...");

}

void loop() {
    // Khi tạo timer
    elapsedTime = millis() - startTiming;
    readLocalCmd();
    if (elapsedTime > (sampleTimingSeconds * 1000))
    {
        readSensors();
        printData();
        showDataLCD();
        startTiming = millis();
    }
}

int getLumen(int anaPin)
{
    int anaValue = 0;
    for (int i = 0; i < 10; i++) // Đọc giá trị cảm biến 10 lần và lấy
    giá trị trung bình
    {
        anaValue += analogRead(anaPin);
        delay(50);
    }

    anaValue = anaValue / 10;
    anaValue = map(anaValue, 1023, 0, 0, 100); //Tỉ lệ:0 ==> Sáng 100%

    return anaValue;
}

int getSoilMoist()
{
    int i = 0;
    int anaValue = 0;
    for (i = 0; i < 10; i++) //
    {
        anaValue += analogRead(SOIL_MOIST_1_PIN); //Đọc giá trị cảm biến
    độ ẩm đất
        delay(50); // Đợi đọc giá trị ADC
    }
}
```

```
    anaValue = anaValue / (i);
    anaValue = map(anaValue, 1023, 0, 0, 100); //Ít nước:0% ==> Nhiều
nước 100%
    return anaValue;
}
void readSensors(void)
{
    tempDHT = dht.readTemperature(); //Đọc nhiệt độ DHT22
    humDHT = dht.readHumidity();      //Đọc độ ẩm DHT22
    lumen = getLumen(LDR_PIN);        //Đọc ánh sáng
    soilMoist = getSoilMoist();        //Đọc cảm biến độ ẩm đất
}
void showDataLCD(void)
{
    lcd.setCursor(0, 1);
    lcd.print(" DO.AM% = ");
    lcd.print(humDHT);
    lcd.println(" % " );

    lcd.setCursor(1, 0);
    lcd.print(" NH.DO = ");
    lcd.print(tempDHT);
    lcd.println(" *C ");
    delay(1000);
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print(" AM.DAT% = ");
    lcd.print(soilMoist);
    lcd.println(" % " );

    lcd.setCursor(1, 0);
    lcd.print("A.SANG% = ");
    lcd.print(lumen);
    lcd.println(" % " );
    delay(1000);
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print(" BOM.NC = ");
    lcd.print(pumpWaterStatus);
    lcd.println(" " );

    lcd.setCursor(1, 0);
    lcd.print("BOM.SAU = ");
    lcd.print(pumpPesStatus);
    lcd.println(" ");
    delay(1000);
```

```
}  
void printData(void)  
{  
    // IN thông tin ra màn hình  
    Serial.print("Do am: ");  
    Serial.print(humDHT);  
    Serial.print(" %t");  
    Serial.print("Nhiệt độ: ");  
    Serial.print(tempDHT);  
    Serial.print(" *Ct");  
    Serial.print("Ánh sáng: ");  
    Serial.print(lumen);  
    Serial.print(" %t");  
    Serial.print("Độ ẩm đất: ");  
    Serial.print(soilMoist);  
    Serial.println(" %");  
}  
  
/*****  
    Hàm đọc trạng thái bơm và kiểm tra nút nhấn  
    (Nút nhấn mặc định là mức "CA0"):  
*****/  
void readLocalCmd()  
{  
    int digiValue = debounce(PUMPW_ON);  
    // Serial.println(digiValue);  
    if (!digiValue)  
    {  
        // Serial.println(" %");  
        pumpWaterStatus = !pumpWaterStatus;  
        showDataLCD();  
        applyCmd();  
    }  
    digiValue = debounce(PUMPS_ON);  
    if (!digiValue)  
    {  
        pumpPesStatus = !pumpPesStatus;  
        showDataLCD();  
        applyCmd();  
    }  
}  
  
/*****  
    Thực hiện điều khiển các bơm  
*****/  
void applyCmd()  
{
```



```
if (pumpWaterStatus == 1) digitalWrite(PUMPW_PIN, LOW);
if (pumpWaterStatus == 0) digitalWrite(PUMPW_PIN, HIGH);

if (pumpPesStatus == 1) digitalWrite(PUMPS_PIN, LOW);
if (pumpPesStatus == 0) digitalWrite(PUMPS_PIN, HIGH);
}
/*****
Hàm kiểm tra trạng thái phím bấm
*****/
boolean debounce(int pin)
{
    boolean state;
    boolean previousState;
    const int debounceDelay = 60;

    previousState = digitalRead(pin);
    for (int counter = 0; counter < debounceDelay; counter++)
    {
        delay(1);
        state = digitalRead(pin);
        if (state != previousState)
        {
            counter = 0;
            previousState = state;
        }
    }
    return state;
}
```

## Chế độ tự động

Giờ thì coi như phần cứng và các kết nối đã tạm ổn, lập trình thêm một chút để có thể tự điều khiển bơm khi độ ẩm quá thấp

```
/* Chương trình đọc nhiệt độ, độ ẩm từ cảm biến DHT
Thêm chức năng đọc cảm biến ánh sáng
Thêm chức năng đọc cảm biến độ ẩm đất
Thêm hiển thị LCD
Thêm chức năng điều khiển tưới tiêu bằng tay 2 bơm
Thêm chức năng điều khiển tự động bơm khi thiếu nước
HocARM NDTR BOT for Arduino by hocARM.org
-----
// Chương trình đọc nhiệt độ, độ ẩm từ cảm biến DHT
// Thêm chức năng đọc cảm biến ánh sáng
// Thêm chức năng đọc cảm biến độ ẩm đất
// Chỉnh sửa cho Arduino Uno bởi hocARM.org
```

```
// Kẹt nôi
// DHT | Arduino Uno
//-----
// VCC(1) | 5V
// DATA(2) | 2
// NC(3) | x
// GND(4) | GND
// Nối tr 10k giữa chân 1 và chân 2
//-----
//Cảm biến độ ẩm | Arduino
//-----
// VCC(1) | 5V
// GND(2) | GND
// D0(3) | x
// A0(4) | A1
// Cảm biến ánh sáng nối chân A0
//-----
// 2 nút nh sẽ được kết nối với chân A2 và A3
// 2 relay nối với chân 3 và chân 12
*/
#include "DHT.h"
//#include "homephone.h"
#include <LiquidCrystal.h>
#define DHTPIN 2 // Chân DATA nối với PD0
#define LDR_PIN A0 // Chân PE3 nối với chân OUT cảm biến as
#define SOIL_MOIST_1_PIN A1 // Chân PE4 nối với cảm biến độ ẩm
// Relay, nút nh
#define PUMPW_ON A2 //Nút có s trên kit
#define PUMPW_PIN 3
#define PUMPS_ON A3 //Nút có s trên kit
#define PUMPS_PIN 12
// Uncomment loại cảm biến bạn s dụng, nếu DHT11 thì uncomment DHT11
// và comment DHT22
//#define DHTTYPE DHT11 // DHT 11
#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
//#define DHTTYPE DHT21 // DHT 21 (AM2301)

// Biến lưu các giá trị cảm biến
int humDHT;
int tempDHT;

int lumen;
int DARK_LIGHT = 40;

int soilMoist;
int DRY_SOIL = 40;
```

```
int WET_SOIL = 60;

// Biến lưu trạng thái bơm
boolean pumpWaterStatus = 0;
boolean pumpPesStatus = 0;
int timePumpOn = 10; // Thời gian bật bơm nước
// Biến cho timer
long sampleTimingSeconds = 50; // ==> Thời gian đọc cảm biến (s)
long startTiming = 0;
long elapsedTime = 0;
// Khởi tạo cảm biến
DHT dht(DHTPIN, DHTTYPE);
// Khởi tạo LCD
LiquidCrystal lcd(9, 8, 7, 6, 5, 4);
void setup() {
    pinMode(PUMPW_PIN, OUTPUT);
    pinMode(PUMPS_PIN, OUTPUT);
    pinMode(PUMPW_ON, INPUT_PULLUP); // Button
    pinMode(PUMPS_ON, INPUT_PULLUP); // Button
    applyCmd();
    // Khởi tạo cổng serial baud 115200
    Serial.begin(115200);
    Serial.println("HocARM NDTR Bot!");
    lcd.begin(16, 2);
    lcd.print("HocARM NDTR Bot!*");
    lcd.setCursor(0, 1);
    lcd.print("Xin chào ông chủ");
    // Bật đầu đọc dữ liệu
    readSensors(); // Khởi tạo đọc cảm biến
    startTiming = millis(); // Bật đầu đếm thời gian
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print("Vui lòng chờ ...");
}

void loop() {
    // Khởi tạo timer
    elapsedTime = millis() - startTiming;
    readLocalCmd();
    if (elapsedTime > (sampleTimingSeconds * 1000))
    {
        readSensors();
        printData();
        showDataLCD();
        autoControlPlantation();
        startTiming = millis();
    }
}
```

```
}  
}  
int getLumen(int anaPin)  
{  
    int anaValue = 0;  
    for (int i = 0; i < 10; i++) // Đọc giá trị cảm biến 10 lần và lấy  
    giá trị trung bình  
    {  
        anaValue += analogRead(anaPin);  
        delay(50);  
    }  
  
    anaValue = anaValue / 10;  
    anaValue = map(anaValue, 1023, 0, 0, 100); //Tí:0 ==> Sáng 100%  
  
    return anaValue;  
}  
int getSoilMoist()  
{  
    int i = 0;  
    int anaValue = 0;  
    for (i = 0; i < 10; i++) //  
    {  
        anaValue += analogRead(SOIL_MOIST_1_PIN); //Đọc giá trị cảm biến  
    độ ẩm đất  
        delay(50); // Đợi đọc giá trị ADC  
    }  
    anaValue = anaValue / (i);  
    anaValue = map(anaValue, 1023, 0, 0, 100); //Ít nước:0% ==> Nhiều  
    nước 100%  
    return anaValue;  
}  
void readSensors(void)  
{  
    tempDHT = dht.readTemperature(); //Đọc nhiệt độ DHT22  
    humDHT = dht.readHumidity(); //Đọc độ ẩm DHT22  
    lumen = getLumen(LDR_PIN); //Đọc ánh sáng  
    soilMoist = getSoilMoist(); //Đọc cảm biến độ ẩm đất  
}  
void showDataLCD(void)  
{  
    lcd.setCursor(0, 1);  
    lcd.print(" DO.AM% = ");  
    lcd.print(humDHT);  
    lcd.println(" % " );
```

```

    lcd.setCursor(1, 0);
    lcd.print(" NH.DO = ");
    lcd.print(tempDHT);
    lcd.println(" *C ");
    delay(1000);
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print(" AM.DAT% = ");
    lcd.print(soilMoist);
    lcd.println(" %  " );

    lcd.setCursor(1, 0);
    lcd.print("A.SANG% = ");
    lcd.print(lumen);
    lcd.println(" %  ");
    delay(1000);
    lcd.clear();
    lcd.setCursor(0, 1);
    lcd.print(" BOM.NC = ");
    lcd.print(pumpWaterStatus);
    lcd.println("      " );

    lcd.setCursor(1, 0);
    lcd.print("BOM.SAU = ");
    lcd.print(pumpPesStatus);
    lcd.println("      ");
    delay(1000);
}
void printData(void)
{
    // IN thông tin ra màn hình
    Serial.print("Do am: ");
    Serial.print(humDHT);
    Serial.print(" %t");
    Serial.print("Nhiệt độ: ");
    Serial.print(tempDHT);
    Serial.print(" *Ct");
    Serial.print("Ánh sáng: ");
    Serial.print(lumen);
    Serial.print(" %t");
    Serial.print("Độ ẩm đất: ");
    Serial.print(soilMoist);
    Serial.println(" %");
}
/*****
Hàm bật bơm nước

```

```
*****/
void turnPumpOn()
{
    digitalWrite(PUMPW_PIN, LOW);
    pumpWaterStatus = 1;
    showDataLCD();
    delay (timePumpOn * 1000);
    digitalWrite(PUMPW_PIN, HIGH);
    pumpWaterStatus = 0;
    showDataLCD();
}
/*****
    Hàm đọc trạng thái bơm và kiểm tra nút nhàn
    (Nút nhàn mặc định là mức "CAO"):
*****/
void readLocalCmd()
{
    int digiValue = debounce(PUMPW_ON);
    if (!digiValue)
    {
        pumpWaterStatus = !pumpWaterStatus;
        showDataLCD();
        aplyCmd();
    }
    digiValue = debounce(PUMPS_ON);
    if (!digiValue)
    {
        pumpPesStatus = !pumpPesStatus;
        showDataLCD();
        aplyCmd();
    }
}
/*****
    Thực hiện điều khiển các bơm
*****/
void aplyCmd()
{
    if (pumpWaterStatus == 1) digitalWrite(PUMPW_PIN, LOW);
    if (pumpWaterStatus == 0) digitalWrite(PUMPW_PIN, HIGH);

    if (pumpPesStatus == 1) digitalWrite(PUMPS_PIN, LOW);
    if (pumpPesStatus == 0) digitalWrite(PUMPS_PIN, HIGH);
}
/*****
    Hàm kiểm tra trạng thái phím bấm
*****/
```

```
boolean debounce(int pin)
{
    boolean state;
    boolean previousState;
    const int debounceDelay = 60;

    previousState = digitalRead(pin);
    for (int counter = 0; counter < debounceDelay; counter++)
    {
        delay(1);
        state = digitalRead(pin);
        if (state != previousState)
        {
            counter = 0;
            previousState = state;
        }
    }
    return state;
}

/*****
    Chức năng tự động tưới tiêu
*****/
void autoControlPlantation()
{
    //----- BƠM NƯỚC -----//
    if (soilMoist < DRY_SOIL && lumen > DARK_LIGHT)
    {
        turnPumpOn();
    }
}
```

## Gửi dữ liệu lên thingspeak

Để gửi dữ liệu lên thingspeak thì cần có kết nối wifi và có modul ESP8266, mình sẽ mình bạn kết nối với ESP8266 v1



Lưu ý quan trọng:

Nguồn cho ESP8266 phải là 3.3V, và nên lấy từ nguồn ngoài, lấy nguồn 3v3 từ Arduino nhiều lúc sẽ không hoạt động được do thiếu dòng.

Nếu bạn có mua mới ESP8266 thì nên mua bản V7 hoặc V12 để có nhiều IO và làm được nhiều việc khác hơn. Như học ESP8266 không kết hợp với Uno hay TIVA gì cả

Do Arduino không có Serial1 nên mình phải sử dụng thư viện có sẵn để cấu hình chân giao tiếp với ESP8266 là SoftwareSerial.h trên 2 chân 10,11

## TRUNG TÂM SỬA CHỮA ĐIỆN TỬ QUẢNG BÌNH

MR. XÔ - 0901.679.359 - 80 Võ Thị Sáu, Phường Quảng Thuận, tx Ba Đồn, tỉnh Quảng Bình

GIÁ RẺ

NHANH CHÓNG

LINH KIỆN CHÍNH HÃNG



## TRUNG TÂM SỬA CHỮA ĐIỆN TỬ XÔ NGUYỄN

- Dịch vụ sửa chữa điện tử tại nhà
- Cung cấp linh kiện điện tử
- Tư vấn lắp đặt nhà thông minh

Đc: Quảng Thuận, tx Ba Đồn,  
tỉnh Quảng Bình - 0901.679.359

```
SoftwareSerial Serial1(10, 11); // RX, TX
```

Chương trình tổng hợp các chức năng đọc dữ liệu, hiển thị LCD, điều khiển thủ công, tự động và gửi thông tin lên cloud

```
/* Chương trình đọc nhiệt độ, độ ẩm từ cảm biến DHT
  Thêm chức năng đọc cảm biến ánh sáng
  Thêm chức năng đọc cảm biến độ ẩm đất
  Thêm hiển thị LCD
  Thêm chức năng điều khiển tưới tiêu bằng tay 2 bơm
  Thêm chức năng điều khiển tự động bơm khi thiếu nước
  Thêm chức năng cập nhật dữ liệu qua cloud thingspeak
  HocARM NDTR BOT by hocARM.org
```

```
-----
// Kết nối
// DHT      | Arduino Uno
//-----
// VCC(1)   | 5V
// DATA(2) | 2
// NC(3)    | x
// GND(4)   | GND
// Nối trở 10k giữa chân 1 và chân 2
//-----
// Cảm biến độ ẩm | Arduino
//-----
// VCC(1)   | 5V
// GND(2)   | GND
```



```
// D0(3)          | x
// A0(4)          | A1
// Cảm biến ánh sáng nối chân A0
//-----
// 2 nút nhấn sẽ được kết nối với chân A2 và A3
// 2 relay nối với chân 3 và chân 12
*/

#include "DHT.h"

#include <LiquidCrystal.h>
#include <SoftwareSerial.h>

SoftwareSerial Serial1(10, 11); // RX, TX

#define DHTPIN 2      // Chân DATA nối với PD0
#define LDR_PIN A0    // Chân PE3 nối với chân OUT cảm biến as
#define SOIL_MOIST_1_PIN A1 // Chân PE4 nối với cảm biến độ ẩm
// Relay, nút nhấn
#define PUMPW_ON A2    //Nút có sọc trên kit
#define PUMPW_PIN 3
#define PUMPS_ON A3    //Nút có sọc trên kit
#define PUMPS_PIN 12

#define IP "184.106.153.149"// thingspeak.com ip
//#define GREEN_LED 13

// Uncomment loại cảm biến bạn sử dụng, nếu DHT11 thì uncomment DHT11
// và comment DHT22
//#define DHTTYPE DHT11    // DHT 11
#define DHTTYPE DHT22    // DHT 22  (AM2302), AM2321
//#define DHTTYPE DHT21    // DHT 21 (AM2301)
// Biến lưu các giá trị cảm biến
int humDHT;
int tempDHT;

int lumen;
int DARK_LIGHT = 40;

int soilMoist;
int DRY_SOIL = 40;
int WET_SOIL = 60;

// Biến lưu trạng thái bơm
boolean pumpWaterStatus = 0;
boolean pumpPesStatus = 0;
```

```
int timePumpOn = 30; // Thời gian bật bơm nước
// Bật cho timer
long sampleTimingSeconds = 60; // ==> Thời gian đọc cảm biến (s)
long startTiming = 0;
long elapsedTime = 0;

char msg[] = "GET /update?key=xxxxxxxxxxxx"; // Thay xxxxxxxxxxxx bằng
API của bạn
char cmd[100];
char aux_str[100];
int legth;

// Khởi tạo cảm biến
DHT dht(DHTPIN, DHTTYPE);
// Khởi tạo LCD
LiquidCrystal lcd(9, 8, 7, 6, 5, 4);
void setup() {
    pinMode(PUMPW_PIN, OUTPUT);
    pinMode(PUMPS_PIN, OUTPUT);
    pinMode(PUMPW_ON, INPUT_PULLUP); // Nut pullup
    pinMode(PUMPS_ON, INPUT_PULLUP); // Nut pullup
    aplyCmd();
    // Khởi tạo cổng serial baud 115200
    Serial.begin(115200);
    Serial1.begin(9600);
    Serial.println("HocARM NDTR Bot!");
    // Bật đầu đọc dữ liệu
    dht.begin();
    lcd.begin(16, 2);
    lcd.print("HocARM NDTR Bot!*");
    lcd.setCursor(0, 1);
    lcd.print("Xin chào ông chủ");
    connectWiFi();
    readSensors(); // Khởi tạo đọc cảm biến
    startTiming = millis(); // Bật đầu đếm thời gian
    lcd.clear();
    lcd.setCursor(1, 0);
    lcd.print("Đã kết nối wifi");
    lcd.setCursor(0, 1);
    lcd.print("Vui lòng chờ ...");
}

void loop() {
    // Khởi tạo timer
    elapsedTime = millis() - startTiming;
    readLocalCmd();
```

```

if (elapsedTime > (sampleTimingSeconds * 1000))
{
    readSensors();
    printData();
    showDataLCD();
    autoControlPlantation();
    updateDataThingSpeak();
    startTiming = millis();
}
}

int getLumen(int anaPin)
{
    int anaValue = 0;
    for (int i = 0; i < 10; i++) // Đọc giá trị cảm biến 10 lần và lấy
    giá trị trung bình
    {
        anaValue += analogRead(anaPin);
        delay(50);
    }

    anaValue = anaValue / 10;
    anaValue = map(anaValue, 1023, 0, 0, 100); //Tí:0 ==> Sáng 100%

    return anaValue;
}

int getSoilMoist()
{
    int i = 0;
    int anaValue = 0;
    for (i = 0; i < 10; i++) //
    {
        anaValue += analogRead(SOIL_MOIST_1_PIN); //Đọc giá trị cảm biến
    độ ẩm đất
        delay(50); // Đợi đọc giá trị ADC
    }
    anaValue = anaValue / (i);
    anaValue = map(anaValue, 1023, 0, 0, 100); //Ít nước:0% ==> Nhiều
    nước 100%
    return anaValue;
}

void readSensors(void)
{
    tempDHT = dht.readTemperature(); //Đọc nhiệt độ DHT22
    humDHT = dht.readHumidity(); //Đọc độ ẩm DHT22
    lumen = getLumen(LDR_PIN); //Đọc ánh sáng
    soilMoist = getSoilMoist(); //Đọc cảm biến độ ẩm đất

```

```
}  
void showDataLCD(void)  
{  
    // lcd.clear();          //Xóa màn hình  
    lcd.setCursor(0, 1);  
    lcd.print(" D0.AM% = ");  
    lcd.print(humDHT);  
    lcd.println("  %  ");  
  
    lcd.setCursor(1, 0);  
    lcd.print(" NH.D0 = ");  
    lcd.print(tempDHT);  
    lcd.println(" *C ");  
    delay(1000);  
    lcd.clear();  
    lcd.setCursor(0, 1);  
    lcd.print(" AM.DAT% = ");  
    lcd.print(soilMoist);  
    lcd.println("  %   " );  
  
    lcd.setCursor(1, 0);  
    lcd.print("A.SANG% = ");  
    lcd.print(lumen);  
    lcd.println(" %  ");  
    delay(1000);  
    lcd.clear();  
    lcd.setCursor(0, 1);  
    lcd.print("      BOM.NUOC = ");  
    lcd.print(pumpWaterStatus);  
    lcd.println("      " );  
  
    lcd.setCursor(1, 0);  
    lcd.print("BOM.TRU.SAU = ");  
    lcd.print(pumpPesStatus);  
    lcd.println("      ");  
    delay(1000);  
}  
void printData(void)  
{  
    // IN thông tin ra màn hình  
    Serial.print("Do am: ");  
    Serial.print(humDHT);  
    Serial.print(" %t");  
    Serial.print("Nhiệt độ: ");  
    Serial.print(tempDHT);  
    Serial.print(" *Ct");
```

```

Serial.print("Anh sang: ");
Serial.print(lumen);
Serial.print(" %t");
Serial.print("Do am dat: ");
Serial.print(soilMoist);
Serial.println(" %");
}
void showPumpLCD(void)
{

    lcd.clear();
    lcd.setCursor(0, 1);
    //lcd.setCursor(0, 1);
    lcd.print("      BOM.NUOC = ");
    lcd.print(pumpWaterStatus);
    lcd.println("      ");

    lcd.setCursor(1, 0);
    //lcd.setCursor (1, 0);
    lcd.print("BOM.TRU.SAU = ");
    lcd.print(pumpPesStatus);
    lcd.println("      ");
    // delay(1000);
}
/*****
    Hàm bật bơm nước
    *****/
void turnPumpOn()
{
    digitalWrite(PUMPW_PIN, LOW);
    pumpWaterStatus = 1;
    showPumpLCD();
    updateCmdThingSpeak();
    delay (timePumpOn * 1000);
    digitalWrite(PUMPW_PIN, HIGH);
    pumpWaterStatus = 0;
    showPumpLCD();
    updateCmdThingSpeak();
}
/*****
    Hàm đọc trạng thái bơm và kiểm tra nút nhàn
    (Nút nhàn mặc định là mức "CAO"):
    *****/
void readLocalCmd()
{
    int digiValue = debounce(PUMPW_ON);

```

```
if (!digiValue)
{
    pumpWaterStatus = !pumpWaterStatus;
    showPumpLCD();
    aplyCmd();
    updateCmdThingSpeak();
}
digiValue = debounce(PUMPS_ON);
if (!digiValue)
{
    pumpPesStatus = !pumpPesStatus;
    showPumpLCD();
    aplyCmd();
    updateCmdThingSpeak();
}
}
/*****
Thực hiện đi[u]n khi[u]n các bơm
*****/
void aplyCmd()
{
    if (pumpWaterStatus == 1) digitalWrite(PUMPW_PIN, LOW);
    if (pumpWaterStatus == 0) digitalWrite(PUMPW_PIN, HIGH);

    if (pumpPesStatus == 1) digitalWrite(PUMPS_PIN, LOW);
    if (pumpPesStatus == 0) digitalWrite(PUMPS_PIN, HIGH);
}
/*****
Hàm ki[m tra trạng thái ph[m b[m
*****/
boolean debounce(int pin)
{
    boolean state;
    boolean previousState;
    const int debounceDelay = 60;

    previousState = digitalRead(pin);
    for (int counter = 0; counter < debounceDelay; counter++)
    {
        delay(1);
        state = digitalRead(pin);
        if (state != previousState)
        {
            counter = 0;
            previousState = state;
        }
    }
}
```

```
}
return state;
}
/*****
Chức năng tự động tưới tiêu
*****/
void autoControlPlantation()
{
    //----- BƠM NƯỚC -----//
    if (soilMoist < DRY_SOIL && lumen > DARK_LIGHT)
    {
        turnPumpOn();
    }
}
// Hàm gửi lệnh AT
int8_t sendATcommand(char* ATcommand, char* expected_answer, unsigned
int timeout) {

    uint8_t x = 0, answer = 0;
    char response[100];
    unsigned long previous;

    memset(response, '', 100);    // xóa buffer

    delay(100);

    while ( Serial1.available() > 0) Serial1.read();    // đọc input

    Serial1.println(ATcommand);    // Gửi lệnh AT

    x = 0;
    previous = millis();

    // Chờ phản hồi
    do {
        if (Serial1.available() != 0) {
            // Nếu có dữ liệu trong buffer UART, đọc và kiểm tra nó với
            expected_answer
            response[x] = Serial1.read();
            x++;
            // Nếu đúng thì trả kết quả answer = 1, thoát hàm
            if (strstr(response, expected_answer) != NULL)
            {
                answer = 1;
            }
        }
    }
```

```
    }
    } while ((answer == 0) && ((millis() - previous) < timeout)); // Nếu
    sai thì tiếp tục thõ lại cho tới hết thời gian timeout
    Serial.println(response); // In giá trị thõy để debug
    return answer;
}
// Hàm gửi lệnh AT 2 để gửi dữ liệu
int8_t sendATcommand2(char* ATcommand, char* expected_answer1,
                      char* expected_answer2, unsigned int timeout) {

    uint8_t x = 0, answer = 0;
    char response[100];
    unsigned long previous;

    memset(response, '', 100); // Khởi tạo lại chuỗi về 0

    delay(100);

    while ( Serial1.available() > 0) Serial1.read(); // Xóa buffer

    Serial1.println(ATcommand); // Gửi lệnh AT

    x = 0;
    previous = millis();

    // Chờ phản hồi
    do {
        // Nếu có dữ liệu từ UART thì đọc và kiểm tra
        if (Serial1.available() != 0) {
            response[x] = Serial1.read();
            x++;
            // Trõ về giá trị 1 nếu thõy expected_answer1
            if (strstr(response, expected_answer1) != NULL)
            {
                answer = 1;
            }
            // Trõ về giá trị 2 nếu thõy expected_answer2
            else if (strstr(response, expected_answer2) != NULL)
            {
                answer = 2;
            }
        }
    }
    }
    // Đợi time out
    while ((answer == 0) && ((millis() - previous) < timeout));
    Serial.println(response); // In giá trị thõy để debug
```



```
    return answer;
}
/*****
    Kết nối wifi
*****/
void connectWiFi(void)
{
    sendATcommand("AT", "OK", 5000);           //Kiểm tra kết nối
    sendATcommand("AT+CWMODE=1", "OK", 5000);   //Cấu hình chế độ
station
    sendATcommand("AT+CWJAP=\"ten_wifi\",\"mat_khau\"", "OK", 5000); //Thay
ten_wifi và mat_khau của bạn
    sendATcommand("AT+CIPMUX=1", "OK", 5000);   // Bật chế độ đa kết
nối
    sendATcommand("AT+CIFSR", "OK", 5000); // Hiện thị ip
    Serial.println("ESP8266 Connected");
}

/*****
    Kết nối với ThingsSpeak.com
*****/
void startThingSpeakCmd(void)
{
    memset(aux_str, '', 100);
    sprintf(aux_str, sizeof(aux_str), "AT+CIPSTART=1,\"TCP\",\"%s\",80",
IP);
    if (sendATcommand2(aux_str, "OK", "ERROR", 20000) == 1)
    {
        Serial.println("OK Connected Thingspeak");
    }
}

/*****
    Gửi data lên channel ThingsSpeak.com
*****/
void sendThingSpeakCmd(void)
{
    memset(aux_str, '', 100);
    sprintf(aux_str, "AT+CIPSEND=1,%d", length);
    if (sendATcommand2(aux_str, ">", "ERROR", 20000) == 1)
    {
        Serial.println(cmd);
        sendATcommand2(cmd, "SEND OK", "ERROR", 30000);
    }
}

/*****
    Truy cập dữ liệu lên thingspeak.com và đóng kết nối
*****/
```

```
*****/
void updateDataThingSpeak(void)
{
    startThingSpeakCmd();
    // Gửi toàn bộ thông tin cảm biến cũng như trạng thái lên thingspeak
    sprintf(cmd, "%s&field1=%d&field2=%d&field3=%d&field4=%d&field5=%d&field6=%d", msg,
tempDHT, humDHT, lumen, soilMoist, pumpWaterStatus, pumpPesStatus);
    legth = strlen(cmd) + 2;
    sendThingSpeakCmd();
    sendATcommand("AT+CIPCLOSE=1", "OK", 5000);
}

/*****
    Cập nhật trạng thái bơm lên thingspeak.com
*****/
void updateCmdThingSpeak(void)
{
    for (int i = 0; i < 1; i++) // Thực hiện 2 lần cho chắc ăn
    {
        startThingSpeakCmd ();
        // Cập nhật trạng thái bơm
        sprintf(cmd, "%s&field5=%d&field6=%d", msg, pumpWaterStatus,
pumpPesStatus);
        legth = strlen(cmd) + 2;
        sendThingSpeakCmd();
        sendATcommand("AT+CIPCLOSE=1", "OK", 5000);
    }
}
```

Source code của 2 phiên bản bạn có thể tải tại [đây](#)

## Tạm kết

Vậy là mình đã hướng dẫn từng bước để các bạn có thể kiểm tra từng cảm biến và kết hợp nó với nhau, tạo thành một hệ thống nông nghiệp đơn giản. Nhược điểm của Arduino Uno đã bộc lộ khi mình làm kết hợp nhiều thứ, thiếu RAM, có thể khắc phục bằng nhiều cách như tối ưu chương trình, nâng cấp lên bản Mega, hoặc dùng TIVA,... Nếu bạn cần tìm hiểu thêm kỹ hơn 1 chút có thể xem lại bài viết về nông nghiệp với TIVA mình có đề cập ở trên.

## Các bài viết tương tự:

1. [Cách khắc phục lỗi. 1:không stand by được 2:stand by sau 2-5s thì quay trở lại màn hình log 3:Mờ biểu tượng stand by. 4: mục system trong windows task manager chêm dụng trên 20% cpu khiến hiệu suất hoạt động máy giảm sút \(lỗi 2 là nguyên nhân cơ bản\) khi mắc lỗi này máy sẽ gặp vấn đề về stand by-computer sleep. - Ai đang gặp](#)

Tài liệu này được tải từ website: <http://linhkienthaomay.com>. Zalo hỗ trợ: 0389937723

[những lỗi trên vui lòng liên hệ với tiny mino trên facebook http://facebook.com/tiny.mino.3](http://facebook.com/tiny.mino.3) hoặc LH: 016577082380 để được hướng dẫn.

2. [chao cac ban. - dan am thanh KENWOOD rat mong cac ban gop y giúp mình chuyên la the nay mình moi nhan dc cua khách nho sua chua cho dan am thanh kenwood chạy radio va doc dia CD nhưngmay khách mang toi trong tình trạng chấp phan cong suat .DIA va RADIO van hoạt động bình thường nhưng bị chấp CONG SUAT nen kep loa vào dinh u neu de lau loa se bị chạy .hien gio mình van chua dám làm gì cả moi kiểm tra so bo thi thay chạy con STK4150 mình nhìn ma đã thay chuối rồi vì hàng xách tay mà lại thay con STK4150 mình chưa thay gặp con này bao giờ vì vay nhờ các bạn gop y và giúp mình xem trên thị trường có con này không vay?tro giới liệu có ko các bạn nhi?ban nào đã từng làm qua ban này xin giúp đỡ mình một tay.thank các bạn nhiều.](#)
3. [Hệ thống nông nghiệp thông minh phiên bản ARM TIVA](#)
4. [Hệ thống nông nghiệp thông minh phiên bản ESP8266](#)
5. [Main PC-g31 b - Mong tất cả các đồng nghiệp giúp đỡ mình.hiện em nó khởi động không lên màn hình.led báo cây đang hoạt động kg sáng.quạt cpu vẫn quay.cpu và chip bắc,nam vẫn nóng.ram bình thường.các bạn cho mình hướng để sửa chữa em nó nhé.cây này của mình.nên mình muốn tự sửa và đi sâu vào main.mình chuyên tivi.](#)
6. [máy lạnh daikin.... - các bạn cho mình hỏi. ví dụ như 1 máy lạnh đang chạy bình thường nhưng ví 1 lí do nào đó mà lock chạy cao dòng nên thermit ngắt lock. như vậy là điện cấp ra dàn nóng vẫn còn đúng ko các bạn. chỉ có thermit ngắt thôi fai ko hay là boar ngắt điện ra dàn nóng luôn](#)
7. [máy tính của m dùng 2 cây ram 128mb, cpu chắc của intel lúc còn dùng FDD, máy bị lỗi như thế này: + khi cắm dây nguồn vào bộ nguồn, máy tự chạy + nguồn, quạt cpu chạy mà màn hình không lên + đèn bàn phím chớp cái mất luôn \( màn hình ok không bị hư gì hết, mình đem qua màn hình và bàn phím khác thử rồi\) mà nó vẫn không lên màn hình + bàn phím không cháy + trên main không có hiện tượng bị phù tụ gì hết, mình đã tháo 2 cây ram, cpu ra vệ sinh sạch sẽ và cắm dây cáp vào thật chắc rồi v=> vậy máy tính m bị gì vậy các bạn, mong các bạn giúp đỡ, mặc dù đầu năm , nhưng m mong sẽ có bạn online cảm ơn cả nhà - bật máy nguồn và quạt chip chạy](#)
8. [Nạp ga tủ lạnh Sharp - Tủ dùng được 1 năm bảo hành đến thay block.Ban đầu đến dàn nóng. phin. block nóng dữ dội tuyết bám đầu dàn.Em đã kiểm tra block khỏe.Thông sức xịt ngâm bằng nito phin e thay mới, cấp e cân xuống 170 khi nạp ga sờ đầu cũng nóng đường hồi động sương mặc dù cho ít ga ạ.](#)
9. [tủ lạnh đông tuyết. \( tủ bảo ôn\) - bực giận. hết ga, mình đã lên giàn nóng riêng. và đã biết giàn nóng bị thủng. giờ mình muốn kiểm tra nguyên giàn lạnh xem có bị thủng ko mà mình ko nghĩ ra cách nào. vì mình mới vào nghề chưa am hiểu và chưa có kinh nghiệm j cả. vì cos một thợ trước đến nhà khách kiểm tra cái tủ này. ong thợ kia phán với chủ nhà là thủng giàn lạnh. giờ mình mới kiểm tra đc mỗi giàn nóng.](#)
10. [tủ lạnh hitachi 154l nội địa - xì dàn nóng,, cấp dàn nóng rồi mà bây giờ sao lâu đông đá quá, dàn nóng nóng dữ dội, block thì nóng không chạm tay vào được luôn, phin lọc cũng nóng luôn.](#)
11. [tulanh vtb quá gio vn - tu bị thung dan nong,minh đã thay dan moi,khi bom ga thay dan nong, nong nua dan con nua dan ve phin kg nong.đau dan lanh co it tuyet bam.mình đã kiểm tra 2dan nonglanh thay kg bị tuc.](#)
12. [tuyển 5 thợ sửa chữa laptop chuyên nghiệp . 0904677396. Mr : hưng - sửa chữa phần cứng laptop chuyên nghiệp](#)