

## **HƯỚNG DẪN THỰC HÀNH**

### **Lab 8: Các dự án nông nghiệp thông minh**

- Học viên thực hành tất cả các nội dung có trong bài học lý thuyết đồng thời hoàn thành các nội dung bài hướng dẫn thực hành sau đây.
- **Bài 1: Xây dựng mô hình dàn rau thủy canh trong nhà**
- **Bài 2: Xây dựng hệ thống điều khiển vườn ươm tự động bằng Blynk**
- **Bài tập về nhà: Người dạy có thể cung cấp thêm một số bài tập về nhà cho học viên để hoàn thiện các kiến thức liên quan đến nội dung bài học**

<b>Bài 1: Xây dựng mô hình dàn rau thủy canh trong nhà</b>
--

#### **1. Mô tả bài toán**

Mô hình dàn rau thủy canh được xây dựng nhằm mục đích thiết kế một dàn rau nhỏ dành cho những gia đình bận rộn, không có không gian trồng cây như ở vùng thành thị, không có thời gian chăm sóc cây trồng. Hệ thống được xây dựng khá nhỏ nhưng vẫn đủ phạm vi cho cây trồng, sử dụng phân bón bằng dung dịch thay vì bằng chất rắn, dễ dàng cho cây hấp thụ và ít công chăm sóc.



*Mô hình dàn rau thủy canh*

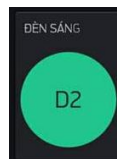
Hệ thống sử dụng một bình chứa dung dịch thủy canh, đo mực nước bình và thông báo cho người dùng khi bình sắp hết nước. Một hệ thống đèn bật tắt tự động, không những thế còn có thể hiển thị thông số nhiệt độ, độ ẩm không khí, hiển thị cường độ ánh sáng và thiết lập bật tắt đèn dựa trên mức độ cao thấp của cường độ ánh sáng. Nếu như chúng ta muốn biết cây trồng của mình đã được bao nhiêu ngày thì hệ thống này cũng cung cấp ngày trồng cũng như ngày bắt đầu trồng cây.

**Một số chức năng cần thực hiện:**

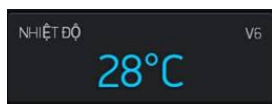
Sau khi thiết lập chương trình và nạp chương trình cho esp8266 chúng ta bắt đầu thiết lập giao diện cho hệ thống dàn rau thủy canh. Hệ thống này được thiết lập trong ứng dụng Blynk trên thiết bị di động dựa trên các hộp tiện ích được hỗ trợ.



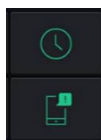
*Giao diện vườn thủy canh trên điện thoại di động*



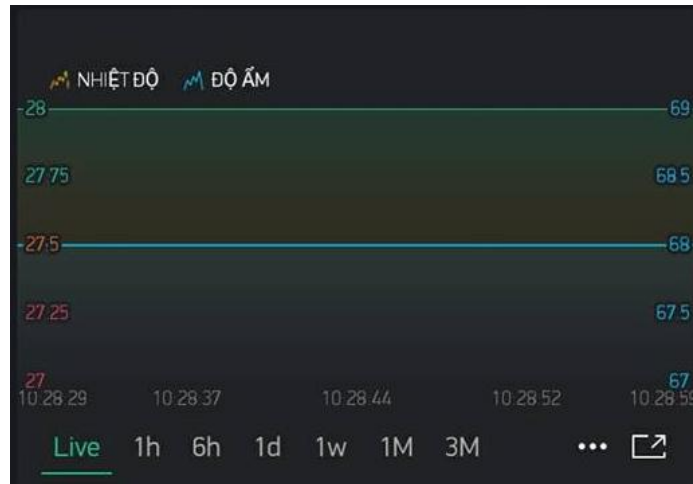
Hệ thống này sử dụng các nút button để bật/ tắt các relay như đèn, hệ thống chống trộm. Có thể cho người dùng tự điều khiển.



Sử dụng thanh để hiển thị thông tin về nhiệt độ độ ẩm không khí, mực nước, ngày ươm. Để biết được các thông số của đàn rau của mình khi cần thiết.

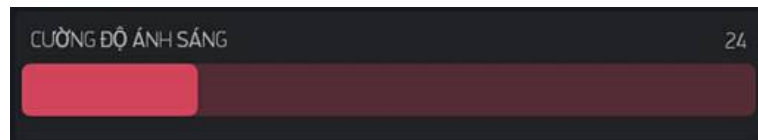


Các thanh dùng để lấy thông số thời gian hiện tại và hiển thị thông báo lên điện thoại.



*Biểu đồ dữ liệu nhiệt độ, độ ẩm*

Biểu đồ này dùng để lưu trữ thông số nhiệt độ, độ ẩm theo từng khoảng thời gian như hiện tại, 1h, 6h, 1 ngày ... chúng ta có thể xem được nhiệt độ, độ ẩm của từng thời gian bằng cách kéo biểu đồ đến thời điểm đó.



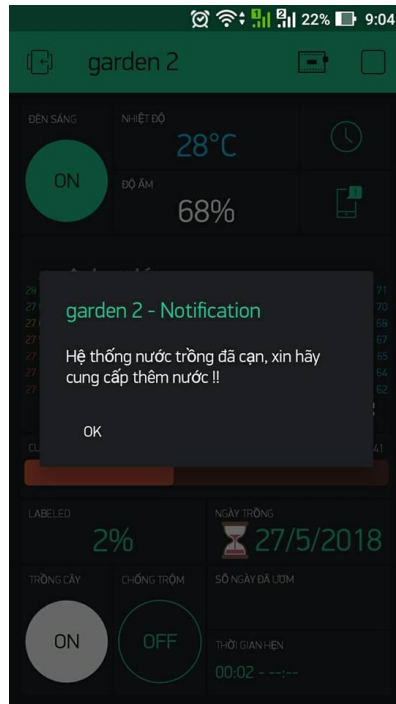
Thanh trạng thái của cường độ ánh sáng

Thanh trạng thái này hiển thị cường độ ánh sáng hiện tại của hệ thống, tùy vào mức độ ánh sáng yếu hay mạnh mà màu sắc của thanh trạng thái cũng thay đổi, nếu cường độ ánh sáng mạnh thì màu sắc sẽ thành màu xanh.



Thanh này dùng để hiển thị ngày bắt đầu trồng khi người dùng nhấn vào nút button trồng cây thì sẽ hiển thị ngày đó vào thanh này.

Hệ thống này sau khi đọc được mực nước và hiển thị lên thanh trạng thái thì sẽ lấy dữ liệu đó để xây dựng thông báo cho người dùng khi hết nước. Thông báo được hiển thị như sau.



Thông báo cung cấp nước cho hệ thống

## 2. Phân tích hệ thống

### a. Mô tả thiết bị

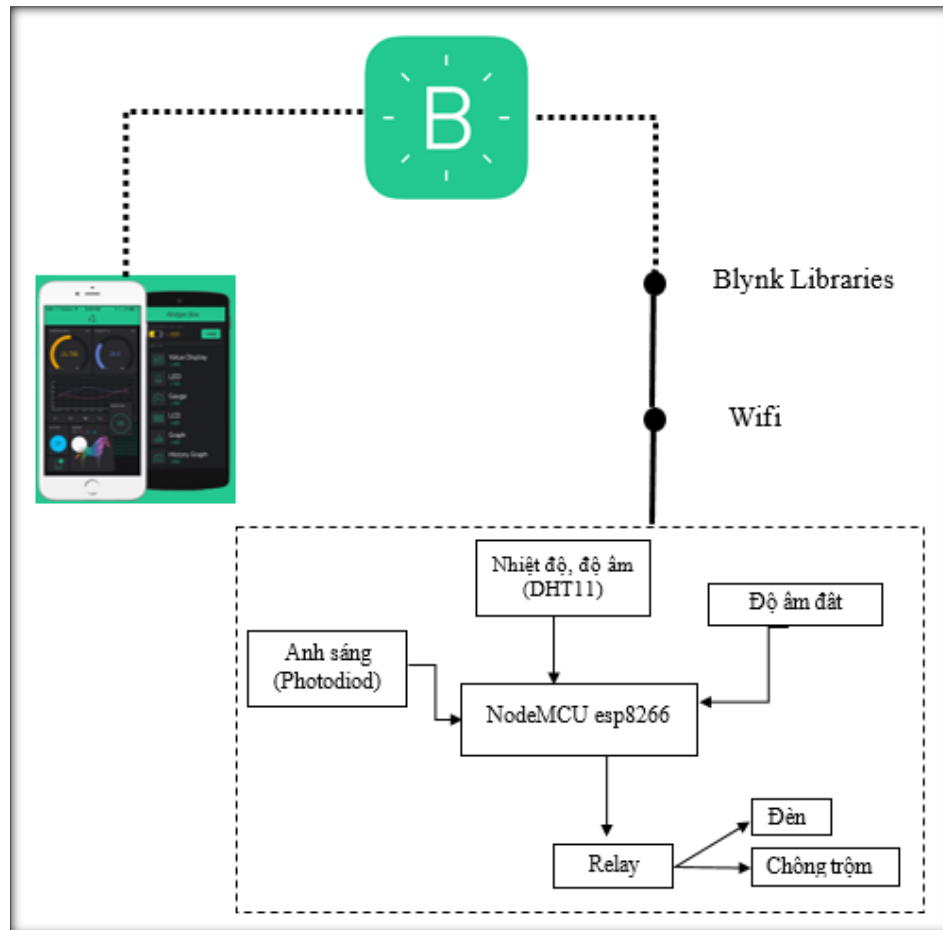
Để xây dựng một hệ thống dàn rau thủy canh trong nhà cần phải hiển thị hay thu thập dữ liệu từ các thiết bị cảm biến để dựa vào các thiết bị đó thiết lập các điều khiển xây dựng hệ thống hoàn thiện hơn, tự động hơn. Hệ thống thủy canh này có thể sử dụng các thiết bị từ mô hình dàn rau thổ canh nhưng xây dựng các chức năng của hệ thống khác nhau.

Các thiết bị sử dụng trong mô hình dàn rau thủy canh

- Dht11: thiết bị này được dùng để đo nhiệt độ, độ ẩm.
- Photodiod: thiết bị dùng để đo cường độ ánh sáng.
- Cảm biến độ ẩm đất: dùng để đo mực nước.
- Relay: dùng để bật tắt các thiết bị như là bật tắt bóng đèn, hệ thống chống trộm.

### b. Sơ đồ hệ thống

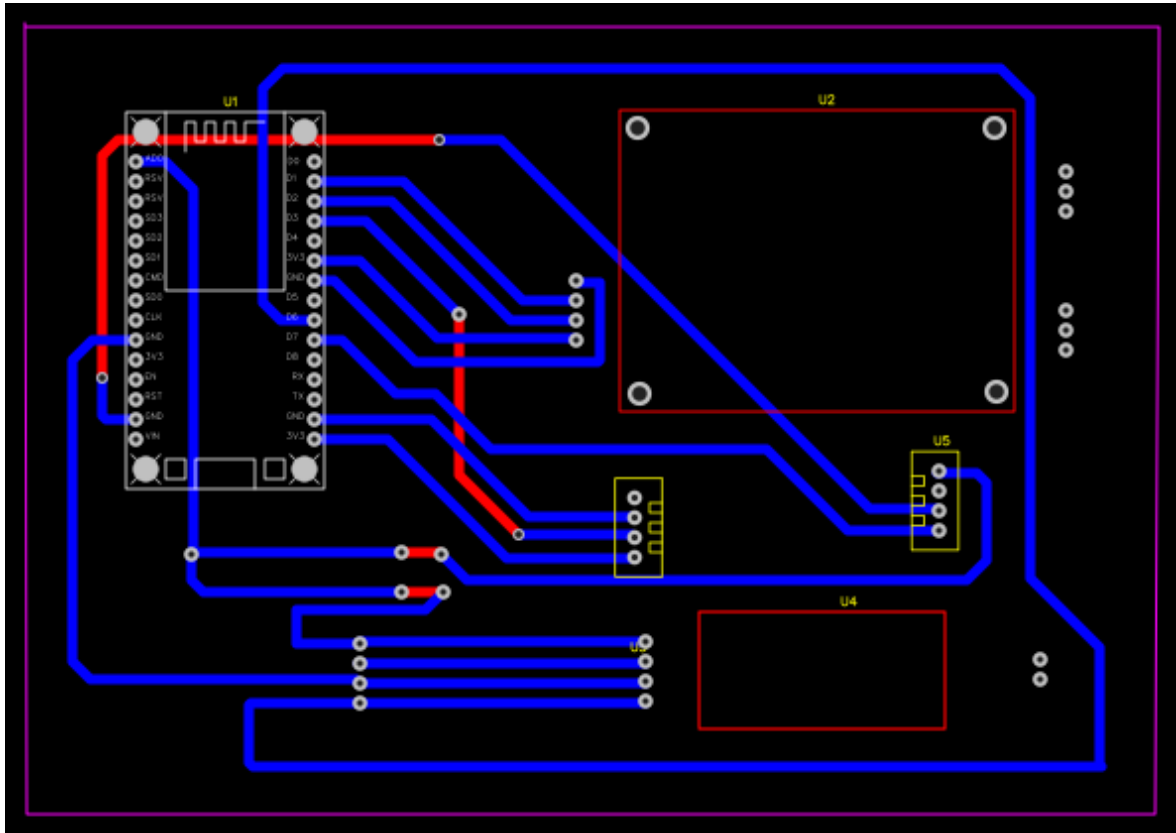
Hệ thống thủy canh sử dụng trong nhà được xây dựng dựa trên sự kết hợp của các thiết bị và mô hình dàn rau thủy canh để xây dựng một hệ thống tự động chăm sóc cây tại nhà.



*Hệ thống dàn rau thủy canh trong nhà*

### c. Mạch PBC

Sau khi xây dựng được hệ thống dàn rau thủy canh thì chúng ta cần xây dựng mạch luân lý PBC tức là mạch sử dụng để lắp các hệ thống vào cùng nhau tạo ra một hệ thống hoàn thiện gắn kết với nhau. Trong đó thể hiện sự kết nối giữa các thiết bị với nhau để tạo thành một bảng mạch. Các chân nối của các thiết bị được nối trực tiếp đến esp8266. Trong đó U1 là esp8266, U2 là relay, U3 là dht11, U4 là độ ẩm đất, U5 là phptodiode.



*Mạch PBC của hệ thống thủy canh*

### **i. Phương pháp thực hiện**

Sau khi xây dựng hệ thống mạch điều khiển và tiến hành đọc từng thiết bị cảm biến (phần 3.1) thì chúng ta tiến hành xây dựng chương trình tự động cho hệ thống thủy canh. Hệ thống được chia làm 3 phần: phần 1 là khai báo thiết bị và lấy thông số từ các chân ảo trên Blynk; phần 2 là tiến hành viết chương trình con đọc các thông số từ các thiết bị cảm biến; phần 3 là tiến hành khai báo chương trình chính, chạy hệ thống kết nối wifi bằng smartconfig và gọi lại chương trình con.

Dựa vào cách xây dựng các chức năng đã được trình bày ở phần 3.1.1 đến 3.1.8 chúng ta tiến hành khai báo thư viện cũng như khai báo các biến lưu trữ.

```
#define BLYNK_PRINT Serial
#include <BlynkSimpleEsp8266.h>
#include <TimeLib.h>
#include <WiFiUdp.h>
#include <WidgetRTC.h>
#include <SPI.h>
#include <DHT.h>
#include <Wire.h>
#include <BH1750.h>
#include <i2cdetect.h>
#include <NTPClient.h> // thư viện hẹn giờ
```

```
#define water D1 // hệ thống chống trộm
#define LED D2 // bật đèn
#define DHTPIN D3
#define DHTTYPE DHT11 // biến khai báo DHT 11
const byte interruptPin = 5; // khai báo lấy giá trị thời gian hiện tại tại D1
volatile byte state = LOW; // khai báo trạng thái bóng đèn
DHT dht(DHTPIN, DHTTYPE);
#define photodiod D7 // cường độ ánh sáng

double sensorValue = 0;
double sensorValue1 = 0;
char auth[] = "d9951d8359a241d88bebddb48fec0573";
WiFiUDP Udp;
WidgetRTC rtc;
SimpleTimer timer;
int pinValue;
int pinValue1;
WiFiUDP u;
int time_start;
int timesecond;
int timehour;
String timeout;
String currentDate;
String now_day;

NTPClient n(u, "2.asia.pool.ntp.org", 7*3600); // lấy khung giờ việt nam
```

Sau khi khai báo thư viện và các biến thực hiện chương trình ta tiến hành lấy thông số các chân ảo của Blynk. Các thông số này có thể là thanh cuộn hay nút button thiết lập để người dùng lựa chọn thông số. Trước khi thực hiện lấy thông số chúng ta tiến hành khai báo chương trình Blynk\_connected(), chương trình này dùng để đồng bộ dữ liệu



trên phần cứng và ứng dụng trên thiết bị di động khi thiết bị đó ngoại tuyến. Trong đó chân V8 dùng để gửi thời gian thực lên server và thiết lập giao diện theo phần 2.7.12, chân V1 dùng để hẹn giờ bật tắt bóng đèn dựa trên phần 2.7.11 (bộ hẹn giờ).

```
BLYNK_CONNECTED() { // đồng bộ dữ liệu được lưu trữ cuối cùng
  Blynk.syncAll();
  rtc.begin();
  Blynk.syncVirtual(V5);
  Blynk.syncVirtual(V6);
  Blynk.syncVirtual(V7);
  Blynk.syncVirtual(V8);
  Blynk.syncVirtual(V9);
  Blynk.syncVirtual(V10);
}

BLYNK_WRITE(V8)
{
  if (param.asInt() == 1)
  {
    currentDate = String(day()) + "/" + String(month()) + "/" + String(year());

    Serial.println(currentDate);
    Blynk.virtualWrite(V9, "\xE2\x8F\xB3", currentDate);
  }
  else
  {
    Blynk.virtualWrite(V9, 0);
  }
}
```

Các chân ảo này được sử dụng như một biến dữ liệu lưu trữ dữ liệu đã được người dùng tác động lên hệ thống, vì vậy các chân này sẽ gửi dữ liệu khi có tác động chứ không liên tục gửi dữ liệu đi.

```
BLYNK_WRITE(V1) //Slider1
{
  TimeInputParam time_input(param);

  // Process timezone
  Serial.println(String("Time zone: ") + time_input.getTZ()+time_input.getStart().hour()
  +time_input.getStart().minute()+time_input.getStart().second());
}
```

Sau khi lấy dữ liệu của các chân ảo thì tiến hành viết chương trình thực hiện hệ thống tự động như đọc các thiết bị cảm biến, hẹn bật tắt đèn nếu trời tối, đo mực nước và xuất thông báo cho người dùng, hiển thị ngày trồng.

```
void sendsensor()
{
    float h = dht.readHumidity(); // Đọc giá trị nhiệt độ C (mặc định)
    float t = dht.readTemperature(); // Đọc giá trị nhiệt độ F (isFahrenheit = true)
    if (isnan(sensorValue) != 0)
    {
        Blynk.virtualWrite(V5, h); // độ ẩm không khí
        Blynk.virtualWrite(V6, t); // nhiệt độ
    }
    //--- đo cường độ ánh sáng
    digitalWrite(D7,HIGH); //pin[6], HIGH);
    delay(2000);
    sensorValue = analogRead(A0);
    sensorValue = constrain(sensorValue, 0, 1023);
    sensorValue = map(sensorValue, 0, 1023, 100, 0);
    Blynk.virtualWrite(V7, sensorValue); // cường độ ánh sáng
    Serial.print("Cường độ ánh sáng: ");
    Serial.println(sensorValue);
    digitalWrite(D7,LOW);
    //---
```

Sử dụng cách đọc các thiết bị cảm biến ở phần 3.2 để đọc nhiệt độ, độ ẩm và gửi dữ liệu lên Blynk qua chân V5, V6; dựa vào phần 3.3 để đọc cường độ ánh sáng và gửi lên Blynk bằng chân V7; dựa vào phần 3.4 để đọc mực nước và gửi dữ liệu lên Blynk bằng chân V2. Tiếp đến sử dụng phần 3.7 để đọc hẹn giờ bật tắt relay sau đó viết chương trình hẹn bật đèn qua relay và viết chương trình thông báo cho người dùng khi hết nước.

Chương trình hẹn bật đèn khi cường độ ánh sáng có độ sáng < 30.

```
void batden()
{
    if (sensorValue < 30)
    {
        digitalWrite(LED,LOW);
    }
    else
    {
        digitalWrite(LED,HIGH);
    }
}
```

Khi hệ thống cung cấp nước thấp hơn so với mức nước quy định thì hệ thống sẽ tự động thiết lập thông báo đến cho người dùng biết rằng hệ thống của họ đã cạn nước và cần cung cấp nước tiếp cho hệ thống.

```
void thongbao_nuoc()
{
    if (sensorValue1 < 80)
    {
        Blynk.notify("Hệ thống nước trồng đã cạn, xin hãy cung cấp thêm nước !!");
    }
}
```

Cuối cùng chúng ta thiết lập chương trình kết nối wifi là smartconfig trong chương trình setup() dựa trên phần 3.1 để thiết lập wifi cho phần cứng và khai báo chạy kết nối đến server Blynk qua câu lệnh Blynk.config(auth,"",8080).

```
Serial.println("");
Serial.println("");

WiFi.printDiag(Serial);
//i2cdetect();
// Khởi tạo server
Udp.begin(49999);
Serial.println("Server started");

// In địa chỉ IP
Serial.println(WiFi.localIP());
Serial.println(WiFi.SSID());
Blynk.config(auth,"cloudblynkserver.ddns.net",8080);
pinMode(LED, INPUT);
pinMode (water, INPUT);
pinMode (D6, OUTPUT); // đo độ ẩm kk
pinMode (D7, OUTPUT);
n.begin();
pinMode(interruptPin, INPUT_PULLUP);
attachInterrupt(digitalPinToInterrupt(interruptPin), blink, RISING);
//timer.setInterval(1000L, hienthigio);
```

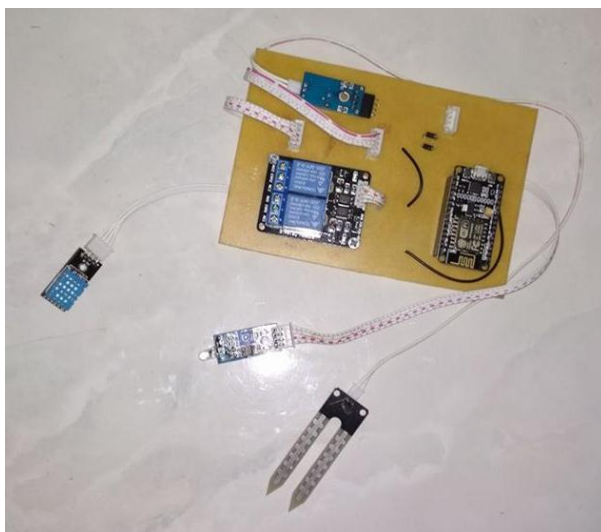
Khai báo hàm setup()

Cuối cùng thiết lập chương trình loop() để chạy các chương trình con và khai báo biến trạng thái.

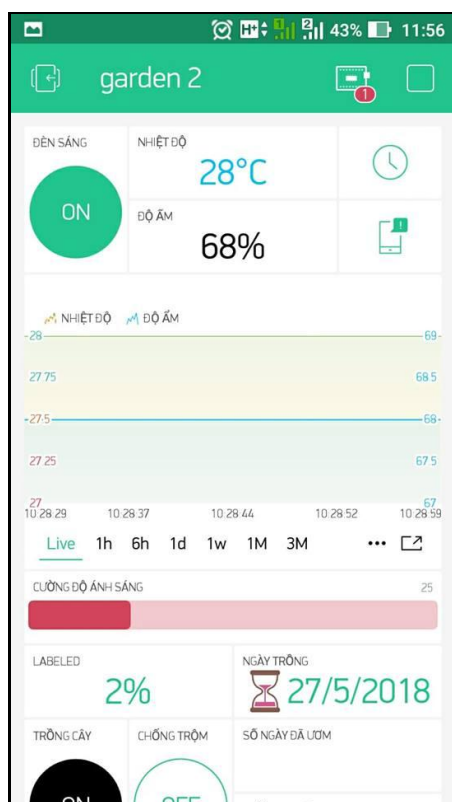
```
void loop()
{
  Blynk.run();
  // Nhận gói tin gửi từ ESPTouch
  //-- hện tắt nước khi bật máy bằng công tắc
  if(state == 0){
    pump_off();
  }
  //--
  n.update();
  //hengio();
  timer.run();
  sendsenser();
  batden();
  thonbao_nuoc();
  //Serial.println(n.getFormattedTime());
  now_day = String(day()) + "/" + String(month()) + "/" + String(year());
  //Serial.println(day_bat);
  Serial.println(now_day);
  timesecond = n.getEpochTime()%60;
  //Serial.println(timesecond);
  digitalWrite(water, state);
  delay(1000);
}
```

### 3. Kết quả

Sau khi thiết lập hệ thống bằng sơ đồ hệ thống thì chúng ta xây dựng được bo mạch điều khiển cũng như xây dựng được giao diện điều khiển hệ thống.



*Mô hình hệ thống thủy canh*



*Giao diện thiết kế điều khiển mô hình thủy canh*

## **Bài 2: Xây dựng hệ thống điều khiển vườn ươm tự động bằng Blynk**

## 1. Mô tả bài toán

Hệ thống này giúp cho chúng ta có thể giám sát được thông tin về vườn ươm cũng như thông tin về các nhân tố bên ngoài tác động đến vườn ươm. Một vườn ươm cây cho đến khi cây đạt đủ độ sinh trưởng để đưa vào hệ thống thủy canh thì được chia làm 2 mốc đó là sơ cấp và thứ cấp. Vườn ươm sơ cấp chỉ cần cung cấp đủ nhiệt độ, độ ẩm để cây có thể nảy mầm và sinh trưởng, khi cây sơ cấp lớn lên đạt đủ độ cao thì chuyển qua hệ thống cây thứ cấp, hệ thống thứ cấp này thì cây sẽ được cung cấp chất dinh dưỡng bằng dung dịch thủy canh nhưng vẫn chưa đưa vào hệ thống trồng cây. Vì vậy chúng tôi xây dựng hệ thống này nhằm giúp cho người sử dụng có thể quản lý được thời gian sinh trưởng của cây cũng như chăm sóc vườn ươm để cây trồng sinh trưởng tốt hơn.



*Mô hình vườn ươm*

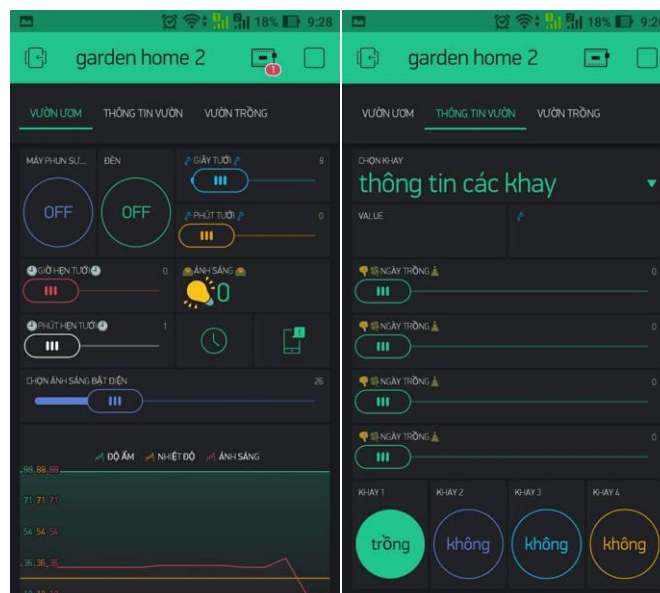
Hệ thống này sử dụng thiết bị đo nhiệt độ, độ ẩm không khí, cho người dùng tự thiết lập thời gian hẹn bật máy phun sương, bật đèn khi trời tối và hiển thị thông số của cường độ ánh sáng, cho người dùng thiết lập mức độ ánh sáng để bật tắt đèn hoặc đèn sẽ được bật từ 7h tối đến 6h sáng. Hiển thị thời gian bắt đầu trồng cây và thông tin của từng giàn ươm. Cũng như hai hệ thống trước hệ thống này cũng sử dụng các thiết bị cảm biến



giống nhau tuy nhiên hệ thống này khác những hệ thống trước ở hệ thống bật máy bơm do người dùng thiết lập, chức năng này được gọi là hẹn giờ lặp lại. Tùy theo khoảng thời gian hẹn của người dùng thì hệ thống sẽ lặp lại khoảng thời gian đó và bật máy phun sương cũng trong khoảng thời gian do người dùng thiết lập.

### Một số chức năng cần thực hiện:

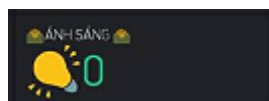
Hệ thống vườn ươm này cũng sử dụng các thiết bị cảm biến như những thiết bị khác. Sau khi thiết lập các chức năng thì chúng ta thiết lập giao diện điều khiển cho người dùng. Giao diện này được chia làm nhiều tab dùng khác nhau tab đầu tiên hiển thị các thông số cơ bản của vườn ươm, tab thứ 2 hiển thị các thông số của từng khay ươm. Giao diện điều khiển ứng dụng trên điện thoại di động như sau:



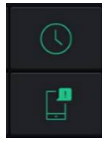
*Hệ thống điều khiển vườn ươm*



Hệ thống này sử dụng các nút button để bật/ tắt các relay như đèn, hệ thống phun sương. Cho phép người dùng tự điều khiển hệ thống của mình hoặc tự động chăm sóc.



Sử dụng thanh để hiển thị thông tin về cường độ ánh sáng đo bằng Lux. Dựa vào ánh sáng này thì sẽ bật đèn nếu như trời tối và tắt đèn nếu trời sáng.

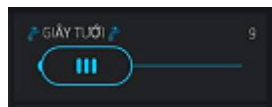


Các thanh dùng để lấy thông số thời gian hiện tại và hiển thị thông báo lên điện thoại. Khi thời gian trồng trong hệ thống thông tin vườn ươm bằng thời gian thực bắt đầu từ ngày trồng thì hệ thống sẽ thông báo đến điện thoại người dùng rằng đã hết thời gian ươm cây và chuyển sang trồng.

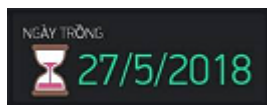


*Biểu đồ thông số nhiệt độ, độ ẩm, cường độ ánh sáng*

Biểu đồ này dùng để lưu trữ thông số nhiệt độ, độ ẩm, cường độ ánh sáng theo từng khoảng thời gian như hiện tại, 1h, 6h, 1 ngày ... chúng ta có thể xem được nhiệt độ, độ ẩm của từng thời gian bằng cách kéo biểu đồ đến thời điểm đó.



Thanh trạng thái này cho người dùng thiết lập thời gian hẹn giờ tưới và hẹn thời gian bật máy phun sương. Hệ thống này cho phép người dùng thiết giờ, phút hẹn tưới và giây hoặc phút thực hiện chức năng tưới.



Thanh dùng để hiển thị ngày bắt đầu trồng khi người dùng nhấn vào nút button trồng cây thì sẽ hiển thị ngày đó vào thanh này.

## 2. Phân tích hệ thống

### a. Mô tả thiết bị

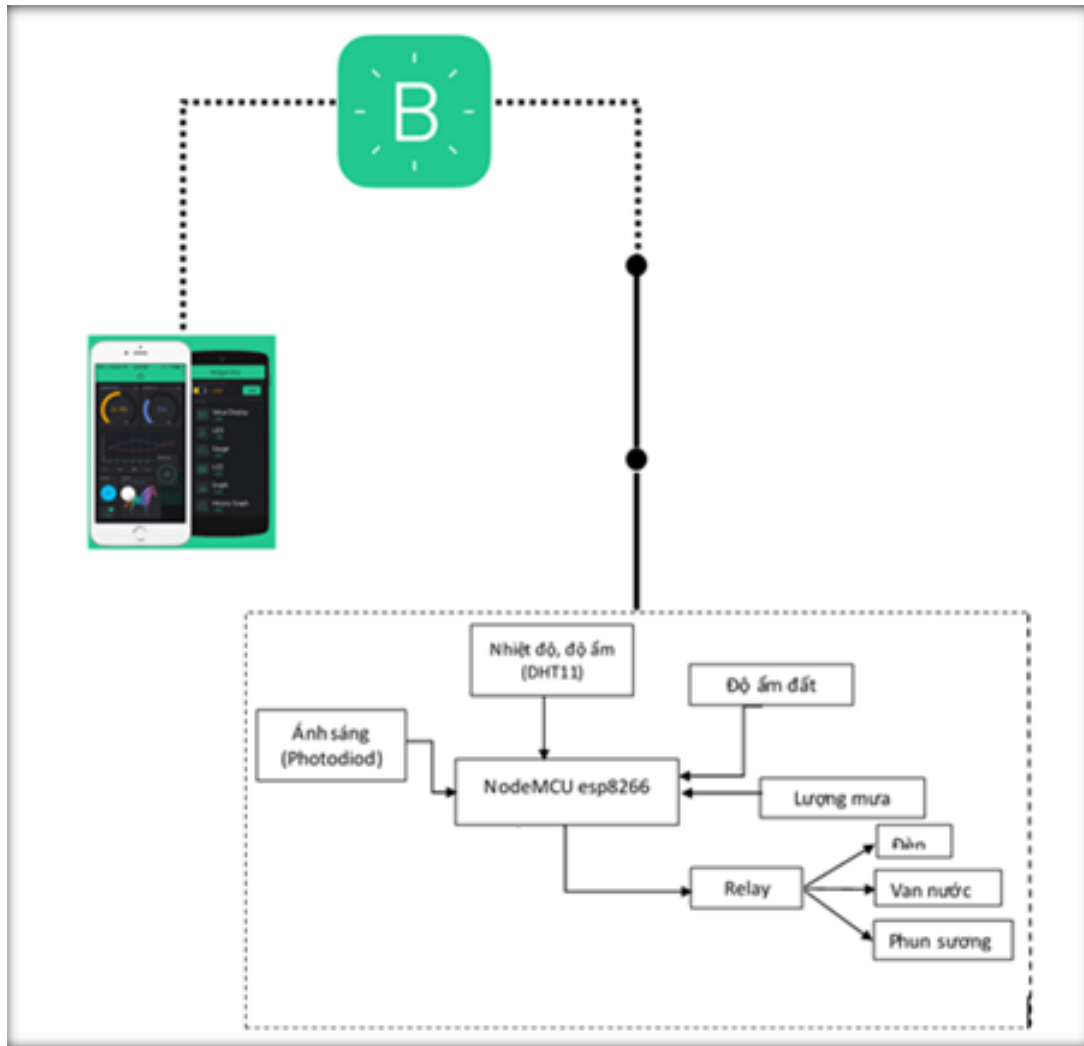


Cũng như các hệ thống khác hệ thống này cũng sử dụng trong hệ thống vườn ươm này gồm thiết bị đo nhiệt độ, độ ẩm, đo cường độ ánh sáng, đo độ ẩm đất, bật tắt đèn.

- DHT11: thiết bị này được dùng để đo nhiệt độ, độ ẩm.
- Photodiod: thiết bị dùng để đo cường độ ánh sáng.
- Cảm biến độ ẩm đất: dùng để đo mực nước.
- Relay: dùng để bật tắt các thiết bị như là bật tắt bóng đèn, bật máy phun sương.

### **b. Sơ đồ hệ thống**

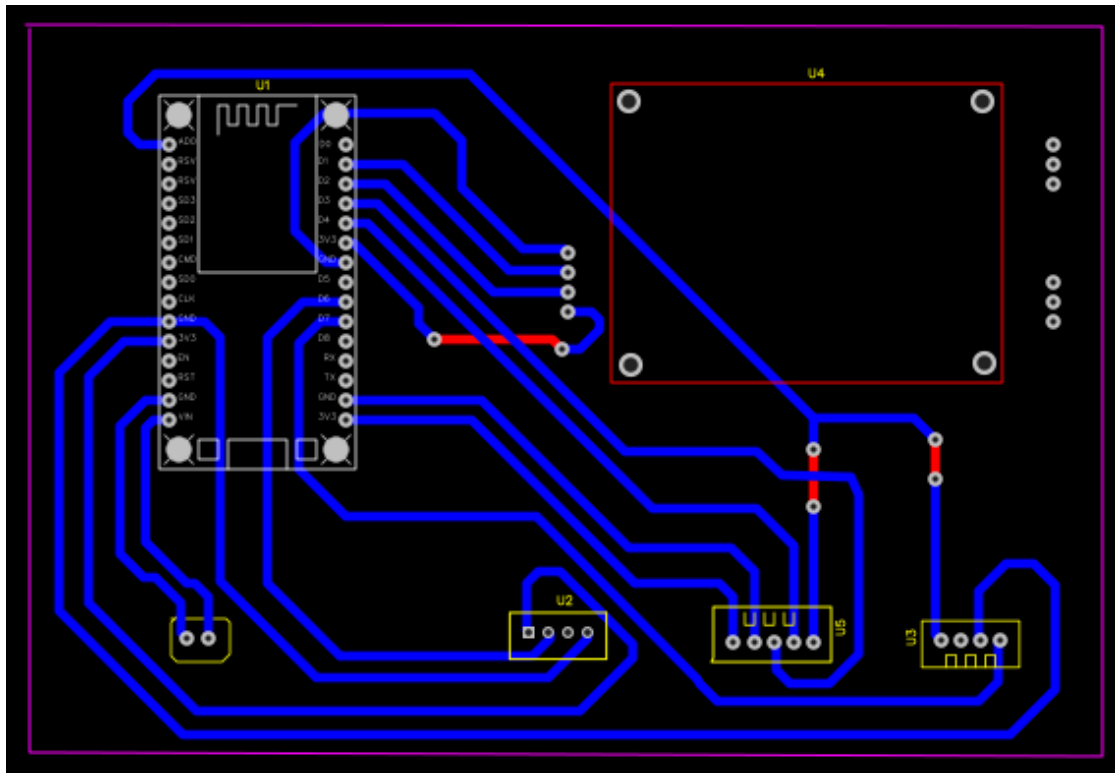
Hệ thống vườn ươm tự động này có mô hình khá giống với hệ thống dàn rau thủy canh, vì các hệ thống điều khiển đều sử dụng các thiết bị cảm biến giống nhau nhưng khác nhau ở các chức năng sử dụng.



*Hệ thống dàn ươm*

### c. Mạch PBC

Dựa vào hệ thống dàn ươm chúng ta xây dựng mạch PBC để gắn các thiết bị lại với nhau nhằm xây dựng một mạch hoàn thiện có thể sử dụng khi nối trực tiếp đến nguồn điện.



*Sơ đồ PBC mạch vườn ươm*

### 3. Phương pháp thực hiện

Sau khi thiết kế sơ đồ hệ thống và mạch luân lý PBC thì chúng ta tiến hành xây dựng chương trình cho hệ thống vườn ươm. Hệ thống này giống hai hệ thống trên là sử dụng thiết bị đo nhiệt độ, độ ẩm, đo cường độ ánh sáng, đo độ ẩm đất. Sau khi tiến hành lấy dữ liệu từ các thiết bị cảm biến dựa trên phần 3.2, 3.3 thì chúng ta tiến hành xây dựng hệ thống bật máy phun sương do người dùng thiết lập.

```
int second_hen;
int minute_hen;
int hour_hen;
int minute_hen_giay;
WiFiUDP u;
int hour_on = 5;
int minute_on = 30;
int time_now;
int time_now_minute;
int time_now_second;
int time_second;
int timehour;
int hour_off = 0;
int minute_off = 0;
int minute_off_hen;
int second_off = 0;
int hour_off_hen;
int second_off_gio_phut;
int timenow;
int so_ngay;
char timeout;
```

### Khai báo biến thực hiện hẹn giờ

Sau khi khai báo các biến cho chương trình chúng ta tiếp tục viết chương trình gửi ngày tháng trông lên Blynk khi chúng ta bắt đầu trông. Chúng ta sử dụng nút button để thiết lập trạng thái trông cây, khi nhấn vào nút button trông thì sẽ hiển thị ngày trông giúp cho người dùng quản lý được thời gian sinh trưởng của cây.

```
BLYNK_WRITE(V10)
{
  if (param.asInt() == 1)
  {
    currentDate = String(day()) + "/" + String(month()) + "/" + String(year());

    Serial.println(currentDate);
    Blynk.virtualWrite(V9, currentDate);
  }
  else
  {
    Blynk.virtualWrite(V9, 0);
  }
}
```

### Thiết lập ngày trông cho hệ thống

Tiếp đến chúng ta thực hiện hẹn giờ lặp lại bật máy bơm. Vì hệ thống này cho người dùng thiết lập nên chúng ta cần lấy các giá trị thiết lập của người dùng như giờ, phút hẹn

máy phun sương bật, phút và giây cho máy phun sương thực hiện trạng thái phun thông qua các chân ảo v1, v2, v3, v4 ở phần 2.7.1.

```

BLYNK_WRITE(V1)
{
    second_hen= param[0].asInt();
    Serial.print(" second value giây chảy is: ");
    Serial.println( second_hen);
}
BLYNK_WRITE(V4)
{
    minute_hen_giay = param[0].asInt();
    Serial.print(" minute value phút is: ");
    Serial.println( minute_hen_giay);
}
BLYNK_WRITE(V2)
{
    minute_hen = param[0].asInt();
    Serial.print(" minute value is: ");
    Serial.println( minute_hen);
}
BLYNK_WRITE(V3)
{
    hour_hen = param[0].asInt();
    Serial.print(" hour value is: ");
    Serial.println( hour_hen);
}
    
```

### Lấy giá trị các biến hẹn thời gian

Sau đó, chúng ta tiếp hành thiết lập thời gian bật máy phun sương và thời gian tắt. Thời gian tắt được thiết lập bằng cách lấy thời gian ngay khi trạng thái máy bơm chuyển từ bật sang tắt. Còn thời gian bật được lấy bằng cách lấy thời gian tắt cộng với thời gian người dùng hẹn. Song vì hệ thống này cần lấy phút và giờ vì vậy cần phải đồng bộ những giá trị này sang giá trị là 24h nếu là giờ, là 60 nếu là phút và giây.

```
void time_on()
{
    int a = minute_off + minute_hen;
    int b = time_second + second_hen;
    int c = hour_off + hour_hen;
    int d = minute_hen_giay + minute_on;
    if((a/60) > 0)
    {
        hour_on = hour_off + hour_hen + (a/60);
        minute_on = a%60;
        if (c/23 > 0)
        {
            hour_on = c%24;
        }
    }
    else{
        hour_on = hour_off + hour_hen;
        minute_on = minute_off + minute_hen;
    }
}
```

### Đồng bộ thời gian thực

Sau khi thiết lập thời gian chúng ta bắt đầu thiết lập lệnh bật/tắt máy phun sao cho đúng với thời gian người dùng thiết lập. Nhưng trước hết cần xây dựng các trạng thái bật/tắt thiết bị.

```
void pump_on()
{
    digitalWrite(water, LOW);
}
void pump_off() // hen giờ tắt nước
{
    digitalWrite(water, HIGH);
}
```

### Khai báo trạng thái của máy phun

Sau đó chúng ta thực hiện việc thiết lập lệnh bật tắt. Lệnh này được thực hiện khi thời gian hiện tại bằng thời gian bật/ tắt thì tiến hành bật/ tắt máy phun sương. Trong chương trình nay chúng ta cũng thực hiện luân lệnh bật tắt ánh sáng khi cường độ ánh sáng thấp.

```

if((hour_on == time_now) && (minute_on == time_now_minute))
{
    pump_on();
}
if ((hour_off_hen == time_now) && (minute_off_hen == time_now_minute) && (second_off <= time_now_second) )
{
    pump_off();
}
if ((time_now >18) || (time_now <6) || (sensorValue1 < anh_sang))
{
    digitalWrite(LED,LOW);
}
else
{
    digitalWrite(LED,HIGH);
}
    
```

Thiết lập lệnh bật tắt máy phun và đèn theo thời gian.

Trong hệ thống này ngoài việc đo cường độ ánh sáng bằng photodiod thì chúng ta còn đo cường độ ánh sáng bằng Lux. Đây cũng là thiết bị đo cường độ ánh sáng nhưng thiết bị này đọc được mức độ của ánh sáng thay vì đọc giá trị từ 0 -> 1023 như photodiod.

```
#include <BH1750.h>
```

Thư viện cho thiết bị đọc cường độ ánh sáng Lux

```

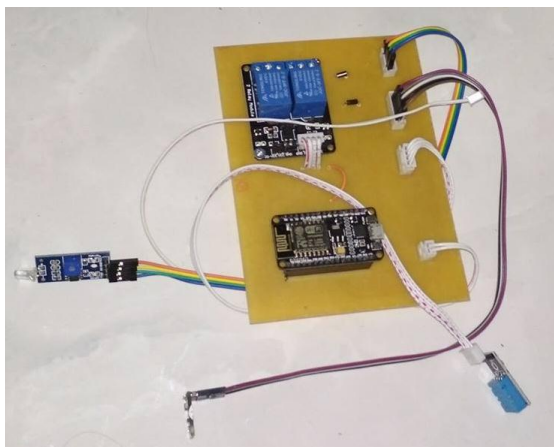
lux = lightMeter.readLightLevel(true);
Serial.print(lux);
lux = map(lux, 0, 1024, 0, 100);
Blynk.virtualWrite(V11, "💡", lux);
Serial.print("Light: ");
Serial.print(lux);
Serial.println(" lx");
//--
    
```

Đọc giá trị ánh sáng bằng Lux

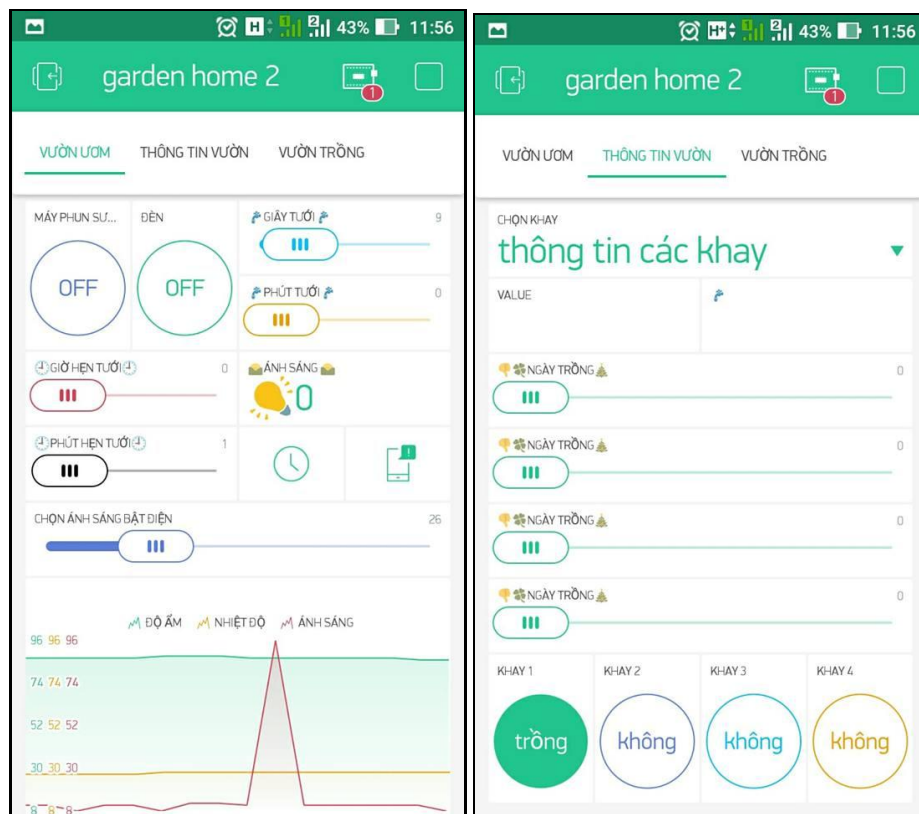
Sau khi đã viết các chương trình thiết lập lệnh và đọc thông số thì chúng ta tiến hành thiết lập hàm setup() khai báo chương trình kết nối đến wifi bằng smartconfig và khai báo các lệnh chạy chương trình. Trong hàm Loop() thì gọi lại các chương trình con.

## 4. Kết quả

Từ các phần mô tả bài toán phân tích hệ thống mô và sơ đồ PBC (phần 3.7.2) chúng ta đã xây dựng được một hệ thống điều khiển dàn rau thủy canh trên ứng dụng Blynk trên thiết bị di động. Hệ thống này sử dụng để điều khiển hệ thống thủy canh này.



*Hệ thống điều khiển vườn ươm*



*Giao diện thiết kế điều khiển vườn ươm*

Giao diện điều khiển thông qua ứng dụng Blynk trên thiết bị di động.