

**BỘ CÔNG THƯƠNG  
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM**



---

# **CÔNG NGHỆ MỚI TRONG PHÁT TRIỂN ỨNG DỤNG CNTT**

---

**Đề tài: Chăm sóc cây trồng thông minh**

SVTH: Nguyễn Bình An  
Nguyễn Tấn Đạt  
Nguyễn Anh Đô  
Lê Duy

15063451 (Nhóm trưởng)  
15074461 (Thư kí)  
15066241  
15076841

GVHD: ThS. Nguyễn Thành Thái

**Nhận xét của giảng viên:**

[illegible]

## MỤC LỤC

CHƯƠNG 1 : GIỚI THIỆU .....	5
1.1 Tổng quan .....	5
1.2 Mục tiêu đề tài .....	5
1.3 Nhiệm vụ.....	5
1.4 Danh sách thiết bị .....	6
1.5 Mô hình mẫu của Project.....	8
CHƯƠNG 2 : CƠ SỞ LÝ THUYẾT .....	10
2.1 Tìm hiểu về Internet of Things .....	10
2.1.1 Khái niệm IoT .....	10
2.1.2 Xu hướng và tính chất .....	10
2.1.3 Ứng dụng.....	12
2.1.4 Những tác nhân ngăn cản sự phát triển của IoT .....	12
2.2 Tìm hiểu về công nghệ wifi trong IoT .....	14
2.2.1 Giới thiệu.....	14
2.2.2 Các thành phần của wifi .....	14
2.2.3 Mô hình wifi network.....	14
2.2.4 Hoạt động .....	15
2.3 Tìm hiểu về mạch Arduino Uno R3 .....	15
2.4 Tìm hiểu về ESP8266 .....	18
2.5 Tìm hiểu về Sensor .....	20
2.5.1 Cảm biến nhiệt độ độ ẩm DHT21 .....	20
2.5.2 Cảm biến độ ẩm đất.....	21
2.5.3 Cảm biến ánh sáng .....	22
2.5.4 Cảm biến mưa.....	23
2.6 Tìm hiểu về lập trình trên Arduino IDE .....	24
2.7 Problem-based Learning .....	27
2.7.1 Đo nhiệt độ độ ẩm và điều khiển quạt khi nhiệt độ cao .....	27
2.7.2 Đo độ ẩm đất và điều khiển máy bơm khi độ ẩm đất thấp.....	28
2.7.3 Sử dụng cảm biến ánh sáng và điều khiển bật đèn khi trời tối.....	29
2.7.4 Sử dụng cảm biến mưa và điều khiển led khi trời mưa.....	29

2.7.5 Tạo một AccessPoint và chạy một web server trên AP .....	30
2.7.6 Lập trình điều khiển 2 led thông qua web server sử dụng esp8266 .....	30
2.7.7 Đo nhiệt độ và độ ẩm gửi lên dịch vụ Thingspeak.....	31
CHƯƠNG 3 : THIẾT KẾ VÀ HIỆN THỰC .....	32
3.1 Thiết kế mô hình .....	32
3.1.1 Chuẩn bị các thiết bị.....	32
3.1.2 Kết nối các thiết bị.....	32
3.2 Chức năng của mô hình .....	32
3.3 Mô hình hoàn chỉnh .....	33
CHƯƠNG 4 : KẾT LUẬN .....	35
4.1 Kết quả đạt được .....	35
4.2 Hạn chế của mô hình .....	35
4.3 Hướng phát triển .....	35
TÀI LIỆU THAM KHẢO.....	36



# CHƯƠNG 1: GIỚI THIỆU

## 1.1 Tổng quan

Đề tài xây dựng hệ thống chăm sóc cây trồng tự động áp dụng IoT.

## 1.2 Mục tiêu đề tài

Xây dựng hệ thống chăm sóc cây trồng tự động, giúp cây trồng phát triển tốt nhất và an toàn. Không sử dụng các chất hóa học, quy trình trồng không cần sự trợ giúp của con người.

Sử dụng các cảm biến như nhiệt độ độ ẩm, ánh sáng, độ ẩm đất, mưa ... và dịch vụ cloud của Thingspeak để giám sát điều khiển tốt nhất cho mỗi loại cây trồng.

Sử dụng servo để quạt điều hòa không khí, bơm tưới nước cho cây có hỗn hợp chất dinh dưỡng trong nước.

## 1.3 Nhiệm vụ

Tuần	Nội dung thực hiện	Người thực hiện
1, 2	Tìm hiểu IoT Chuẩn bị thiết bị arduino, sensor, esp8266, điện trở, led,.. Tìm hiểu mạch Arduino Tìm hiểu arduino IDE Code 1 và 3 đèn led chớp tắt 1s trên Arduino Uno	
3	Kết nối Arduino với các thiết bị Sensor: + Kết nối Arduino với cảm biến mưa + Kết nối Arduino với cảm biến nhiệt độ, độ ẩm + ESP8266 chạy chương trình quét AP + ESP8266 chạy chương trình bật tắt 1 và 2 đèn led qua giao diện web	
4	Tạo một AP trên ESP8266 và chạy 1 server web trên AP Kết nối ESP8266 với một AP Đăng kí dịch vụ Thingspeak	
5	Xây dựng một soft AP / web server trên ESP8266. Thực hiện điều khiển 1 và 2 led.	







	Kết nối ESP8266 với cảm biến nhiệt độ độ ẩm và độ ẩm đất gửi lên thingspeak	
6	Tổng hợp hoàn thành các nhiệm vụ từ tuần 1 -> 5 Triển khai project chăm sóc cây trồng thông minh	
7	Đo độ ẩm đất, xử lý tưới cây và đưa dữ liệu lên thingspeak	
8	Đo độ nhiệt độ độ ẩm, đưa dữ liệu lên thingspeak. Xử lý điều khiển nhiệt độ độ ẩm cho phù hợp với từng loại cây trồng.	
9	Xây dựng mô hình và lắp thiết bị vào mô hình.	
10	Hoàn thành project, viết báo cáo đề tài.	

Tiến trình thực hiện và code đề tài:







<https://github.com/nguyenan97/SmartGarden/tree/master/SMARTGARDEN>

#### 1.4 Danh sách thiết bị

Sản phẩm	Tên sản phẩm	Số lượng	Giá tiền
	Arduino Uno R3 Không kèm cáp USB	1	135,000đ
	Kit RF thu phát Wifi ESP8266 NodeMCU Lua CP2102	1	115,000đ
	Cảm biến độ ẩm đất	1	30,000đ
	Cảm biến độ ẩm, nhiệt độ DHT21 AM2301	1	85,000đ
	Cảm biến ánh sáng Photodiode	1	25,000đ

	Cảm biến mưa	1	30,000đ
	Dây Cắm Test Board Đực Đực 30cm (40 sợi)	1	22,000đ
	Dây Cắm Test Board Đực Cái 30cm (40 sợi)	1	22,000đ
	Đầu chuyển Domino sang Giắc DC đực 2.1mm	1	5,000đ
	Testboard thường loại lớn	1	35,000đ
	Động cơ bơm P385 12VDC 3W	1	65,000đ

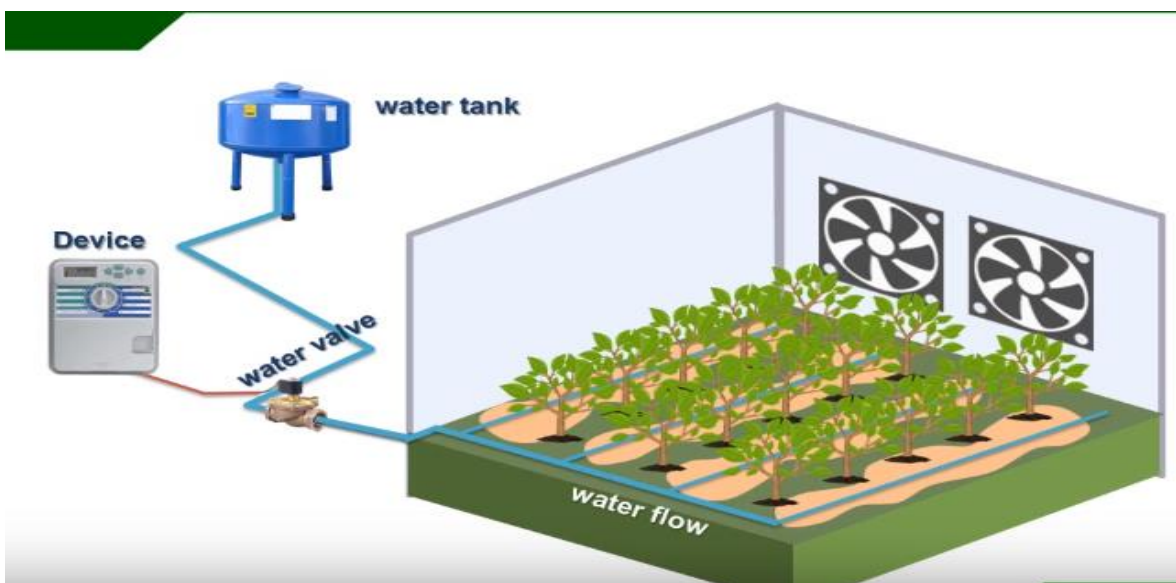
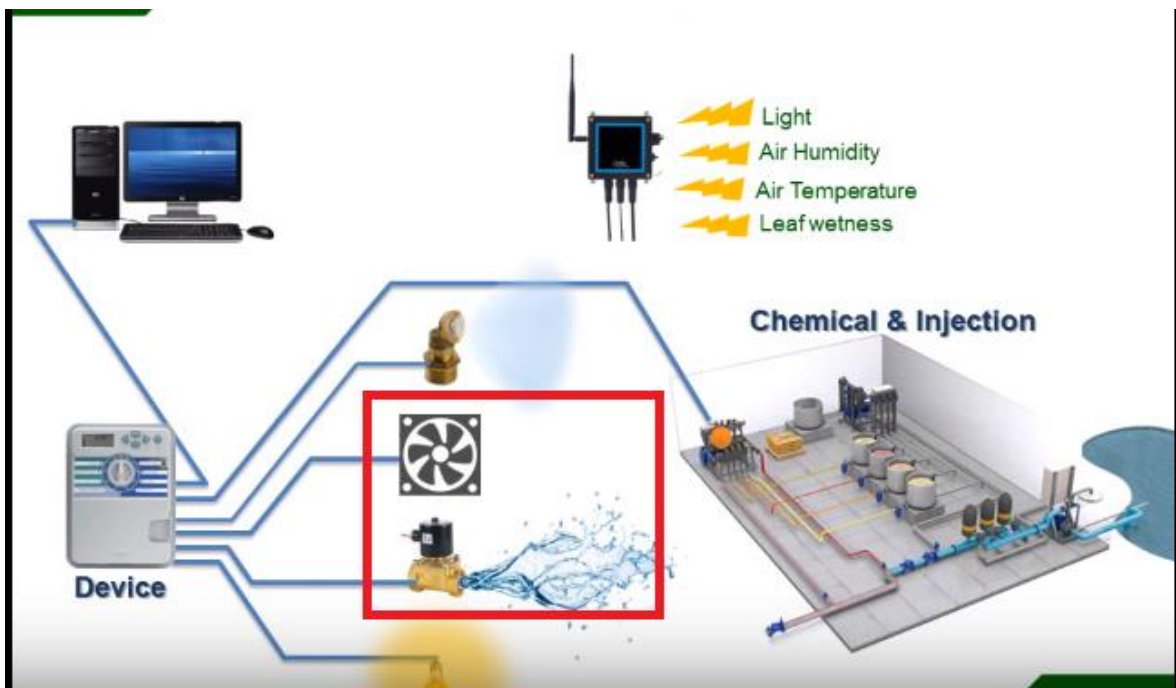
  

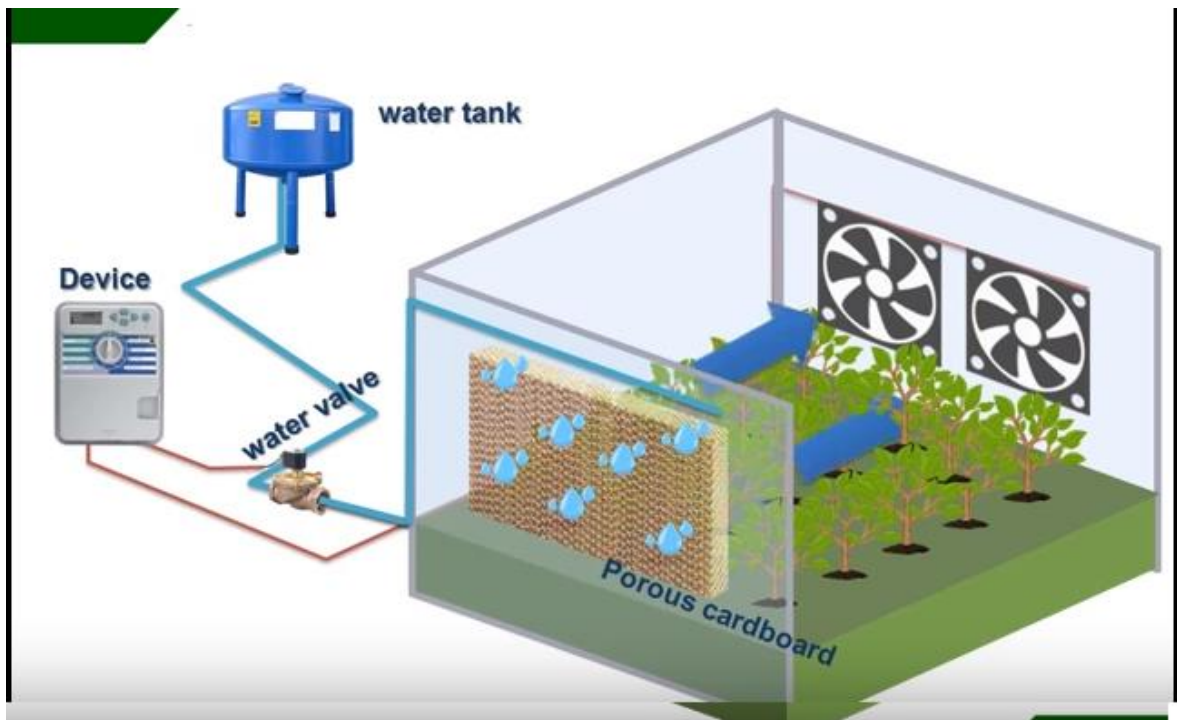
	Cặp động cơ coreless 716 kèm cánh quạt	1	55,000đ
	LCD text LCD1602 xanh lá	1	38,000đ
	Mạch chuyển giao tiếp LCD1602, LCD1604, LCD2004 sang I2C	1	18,000đ
	Nguồn Adaptor AC-DC 5V 1A	1	38,000đ
	Module 1 Relay Với Opto Cách Ly kích H/L (5VDC)	2	60,000đ
	Kit Arduino Wifi ESP8266 NodeMCU Lua WeMos D1 R2	2	330,000đ



	Led màu chân cắm 5mm (10 con)	2	8,000đ	
	Ống mềm đường kính trong 6mm	1	12,000đ	
Tổng tiền:			1,128,000đ	

## 1.5 Mô hình mẫu của Project





## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

### 2.1 Tìm hiểu về Internet of Things

#### 2.1.1 Khái niệm IoT

Theo wikipedia: Internet of Things( IoT), hay cụ thể hơn là Mạng lưới vạn vật kết nối Internet hoặc là Mạng lưới thiết bị kết nối Internet là một liên mạng, trong đó các thiết bị, phương tiện vận tải (được gọi là "thiết bị kết nối" và "thiết bị thông minh"), phòng ốc và các trang thiết bị khác được nhúng với các bộ phận điện tử, phần mềm, cảm biến, cơ cấu chấp hành cùng với khả năng kết nối mạng máy tính giúp cho các thiết bị này có thể thu thập và truyền tải dữ liệu. Năm 2013, tổ chức Global Standards Initiative on Internet of Things (IoT-GSI) định nghĩa IoT là "hạ tầng cơ sở toàn cầu phục vụ cho xã hội thông tin, hỗ trợ các dịch vụ (điện toán) chuyên sâu thông qua các vật thể (cả thực lẫn ảo) được kết nối với nhau nhờ vào công nghệ thông tin và truyền thông hiện hữu được tích hợp,"[3] và với mục đích ấy một "vật" là "một thứ trong thế giới thực (vật thực) hoặc thế giới thông tin (vật ảo), mà vật đó có thể được nhận dạng và được tích hợp vào một mạng lưới truyền thông".[4] Hệ thống IoT cho phép vật được cảm nhận hoặc được điều khiển từ xa thông qua hạ tầng mạng hiện hữu, tạo cơ hội cho thế giới thực được tích hợp trực tiếp hơn vào hệ thống điện toán, hệ quả là hiệu năng, độ tin cậy và lợi ích kinh tế được tăng cường bên cạnh việc giảm thiểu sự can dự của con người. Khi IoT được gia tố cảm biến và cơ cấu chấp hành, công nghệ này trở thành một dạng thức của hệ thống ảo-thực với tính tổng quát cao hơn, bao gồm luôn cả những công nghệ như điện lưới thông minh, nhà máy điện ảo, nhà thông minh, vận tải thông minh và thành phố thông minh. Mỗi vật được nhận dạng riêng biệt trong hệ thống điện toán nhúng và có khả năng phối hợp với nhau trong cùng hạ tầng Internet hiện hữu.

#### 2.1.2 Xu hướng và tính chất

##### *Thông minh*

Sự thông minh và tự động trong điều khiển thực chất không phải là một phần trong ý tưởng về IoT. Các máy móc có thể dễ dàng nhận biết và phản hồi lại môi trường xung quanh, chúng cũng có thể tự điều khiển bản thân mà không cần đến kết nối mạng. Tuy nhiên, trong thời gian gần đây người ta bắt đầu nghiên cứu kết hợp hai khái niệm IoT và autonomous control lại với nhau. Tương lai của IoT có thể là một

mạng lưới các thực thể thông minh có khả năng tự tổ chức và hoạt động riêng lẻ tùy theo tình huống, môi trường, đồng thời chúng cũng có thể liên lạc với nhau để trao đổi thông tin, dữ liệu.

Việc tích hợp trí thông minh vào IoT còn có thể giúp các thiết bị, máy móc, phần mềm thu thập và phân tích các dấu vết điện tử của con người khi chúng ta tương tác với những thứ thông minh, từ đó phát hiện ra các tri thức mới liên quan tới cuộc sống, môi trường, các mối tương tác xã hội cũng như hành vi con người.

#### *Kiến trúc dựa trên sự kiện*

Các thực thể, máy móc trong IoT sẽ phản hồi dựa theo các sự kiện diễn ra trong lúc chúng hoạt động theo thời gian thực. Một số nhà nghiên cứu từng nói rằng một mạng lưới các sensor chính là một thành phần đơn giản của IoT.

#### *Là một hệ thống phức tạp*

Trong một thế giới mở, IoT sẽ mang tính chất phức tạp bởi nó bao gồm một lượng lớn các đường liên kết giữa những thiết bị, máy móc, dịch vụ với nhau, ngoài ra còn bởi khả năng thêm vào các nhân tố mới.

#### *Kích thước*

Một mạng lưới IoT có thể chứa đến 50 đến 100 nghìn tỉ đối tượng được kết nối và mạng lưới này có thể theo dõi sự di chuyển của từng đối tượng. Một con người sống trong thành thị có thể bị bao bọc xung quanh bởi 1000 đến 5000 đối tượng có khả năng theo dõi.

#### *Vấn đề không gian, thời gian*

Trong IoT, vị trí địa lý chính xác của một vật nào đó là rất quan trọng. Hiện nay, Internet chủ yếu được sử dụng để quản lý thông tin được xử lý bởi con người. Do đó những thông tin như địa điểm, thời gian, không gian của đối tượng không mấy quan trọng bởi người xử lý thông tin có thể quyết định các thông tin này có cần thiết hay không, và nếu cần thì họ có thể bổ sung thêm. Trong khi đó, IoT về lý thuyết sẽ thu thập rất nhiều dữ liệu, trong đó có thể có dữ liệu thừa về địa điểm, và việc xử lý dữ liệu đó được xem như không hiệu quả. Ngoài ra, việc xử lý một khối lượng lớn dữ liệu trong thời gian ngắn đủ để đáp ứng cho hoạt động của các đối tượng cũng là một thách thức hiện nay

### 2.1.3 Ứng dụng

IoT có ứng dụng rộng vô cùng, có thể kể ra một số như sau:

- Quản lý chất thải
- Quản lý và lập kế hoạch quản lý đô thị
- Quản lý môi trường
- Phản hồi trong các tình huống khẩn cấp
- Mua sắm thông minh
- Quản lý các thiết bị cá nhân
- Đồng hồ đo thông minh
- Tự động hóa ngôi nhà

Một trong những vấn đề với IoT đó là khả năng tạo ra một ứng dụng IoT nhanh chóng. Để khắc phục, hiện nay nhiều hãng, công ty, tổ chức trên thế giới đang nghiên cứu các nền tảng giúp xây dựng nhanh ứng dụng dành cho IoT. Đại học British Columbia ở Canada hiện đang tập trung vào một bộ toolkit cho phép phát triển phần mềm IoT chỉ bằng các công nghệ/tiêu chuẩn Web cũng như giao thức phổ biến. Công ty như ioBridge thì cung cấp giải pháp kết nối và điều khiển hầu như bất kì thiết bị nào có khả năng kết nối Internet, kể cả đèn bàn, quạt máy...

### 2.1.4 Những tác nhân ngăn cản sự phát triển của IoT

*Chưa có một ngôn ngữ chung*

Ở mức cơ bản nhất, Internet là một mạng dùng để nối thiết bị này với thiết bị khác. Nếu chỉ riêng có kết nối không thôi thì không có gì đảm bảo rằng các thiết bị biết cách nói chuyện với nhau. Ví dụ, bạn có thể đi từ Việt Nam đến Mỹ, nhưng không đảm bảo rằng bạn có thể nói chuyện với người Mỹ.

Để các thiết bị có thể giao tiếp với nhau, chúng sẽ cần một hoặc nhiều giao thức (protocols), có thể xem là một thứ ngôn ngữ chuyên biệt để giải quyết một tác vụ nào đó. Chắc chắn bạn đã ít nhiều sử dụng một trong những giao thức phổ biến nhất thế giới, đó là HyperText Transfer Protocol (HTTP) để tải web. Ngoài ra chúng ta còn có SMTP, POP, IMAP dành cho email, FTP dùng để trao đổi file, ...

Những giao thức như thế này hoạt động ổn bởi các máy chủ web, mail và FTP thường không phải nói với nhau nhiều, khi cần, một phần mềm phiên dịch đơn giản sẽ đứng ra làm trung gian để hai bên hiểu nhau. Còn với các thiết bị IoT, chúng phải

đảm đương rất nhiều thứ, phải nói chuyện với nhiều loại máy móc thiết bị khác nhau. Đáng tiếc rằng hiện người ta chưa có nhiều sự đồng thuận về các giao thức để IoT trao đổi dữ liệu. Nói cách khác, tình huống này gọi là "giao tiếp thất bại", một bên nói nhưng bên kia không thềm (và không thể) nghe.

#### *Hàng rào subnetwork*

Như đã nói ở trên, thay vì giao tiếp trực tiếp với nhau, các thiết bị IoT hiện nay chủ yếu kết nối đến một máy chủ trung tâm do hãng sản xuất một nhà phát triển nào đó quản lí. Cách này cũng vẫn ổn thôi, những thiết bị vẫn hoàn toàn nói được với nhau thông qua chức năng phiên dịch của máy chủ rồi. Thế nhưng mọi chuyện không đơn giản như thế, cứ mỗi một mạng lưới như thế tạo thành một subnetwork riêng, và buồn thay các máy móc nằm trong subnetwork này không thể giao tiếp tốt với subnetwork khác.

Một số trong những vấn đề nói trên chỉ đơn giản là vấn đề về kiến trúc mạng, về kết nối mà các thiết bị sẽ liên lạc với nhau (Wifi, Bluetooth, NFC,...). Những thứ này thì tương đối dễ khắc phục với công nghệ không dây ngày nay. Còn với các vấn đề về giao thức thì phức tạp hơn rất nhiều, nó chính là vật vản lớn và trực tiếp trên còn đường phát triển của Internet of Things.

#### *Có quá nhiều "ngôn ngữ địa phương"*

Bây giờ giả sử như các nhà sản xuất xe ô tô nhận thấy rằng họ cần một giao thức chung để xe của nhiều hãng có thể trao đổi dữ liệu cho nhau và họ đã phát triển thành công giao thức đó. Thế nhưng vấn đề vẫn chưa được giải quyết. Nếu các trạm thu phí đường bộ, các trạm bơm xăng muốn giao tiếp với xe thì sao? Mỗi một loại thiết bị lại sử dụng một "ngôn ngữ địa phương" riêng thì mục đích của IoT vẫn chưa đạt được đến mức tối đa. Đồng ý rằng chúng ta vẫn có thể có một trạm kiểm soát trung tâm, thế nhưng các thiết bị vẫn chưa thật sự nói được với nhau.

#### *Tiền và chi phí*

Cách duy nhất để các thiết bị IoT có thể thật sự giao tiếp đó là khi có một động lực kinh tế đủ mạnh khiến các nhà sản xuất đồng ý chia sẻ quyền điều khiển cũng như dữ liệu mà các thiết bị của họ thu thập được. Hiện tại, các động lực này không nhiều. Có thể xét đến ví dụ sau: một công ty thu gom rác muốn kiểm tra xem các thùng rác có đầy hay chưa. Khi đó, họ phải gặp nhà sản xuất thùng rác, đảm bảo rằng họ có thể truy cập vào hệ thống quản lí của từng thùng một. Điều đó khiến chi phí bị đội lên,

và công ty thu gom rác có thể đơn giản chọn giải pháp cho một người chạy xe kiểm tra từng thùng một.

## 2.2 Tìm hiểu về công nghệ wifi trong IoT

### 2.2.1 Giới thiệu

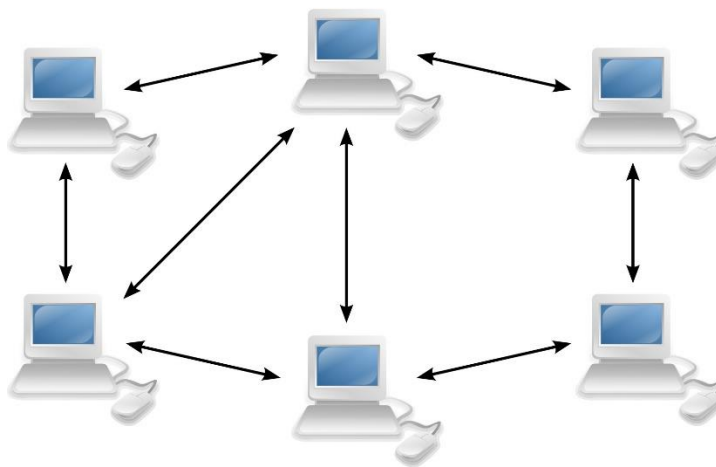
- WiFi: Wireless Fidelity( Chuẩn IEEE 802.11).
- WLAN: Wireless LAN.
- Radio Waves( sóng Radio).
- Hostpots: Một trạm/thiết bị để tất cả mọi người truy cập vào Internet.

### 2.2.2 Các thành phần của wifi

- Access Point: BS( Base Station)
- WiFi Cards
- Safeguards: FireWall, Antivirus Software

### 2.2.3 Mô hình wifi network

- Peer-to-Peer

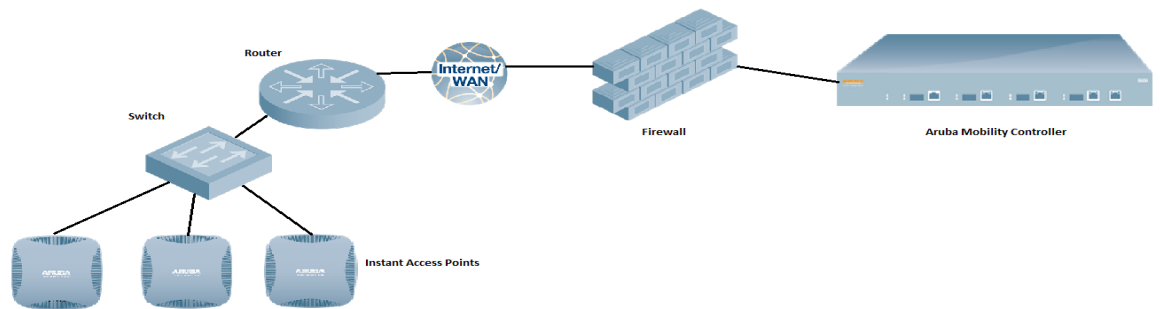


Không Access Point

Các thiết bị kết nối trực tiếp

Nhanh chóng dễ dàng

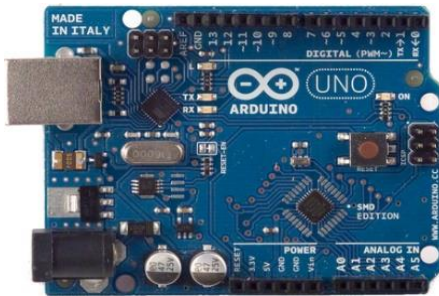
- AP-based Topology



#### 2.2.4 Hoạt động

- WiFi Hostpots để truy cập Internet
- AP-base station.
- Hệ thống phát hiện AP -> Yêu cầu kết nối.
- Trong nhà: 100-150 feets.
- Ngoài trời: 300 feets.
- Nhiều AP kết nối qua Switches

#### 2.3 Tìm hiểu về mạch Arduino Uno R3



Thông số kỹ thuật của Arduino Uno R3:

- Chip điều khiển chính: ATmega328
- Chip nạp và giao tiếp UART: ATmega16U2
- Nguồn nuôi mạch: 5VDC từ cổng USB hoặc nguồn ngoài cắm từ giắc tròn DC (khuyến dùng 7-9VDC để đảm bảo mạch hoạt động tốt. Nếu bạn cắm 12V thì IC ổn áp rất dễ chết và gây hư hỏng mạch).
- Số chân Digital: 14 (hỗ trợ 6 chân PWM)
- Số chân Analog: 6
- Dòng ra tối đa trên GPIO: 40mA
- Dòng ra tối đa trên chân cấp nguồn 3.3VDC: 150 mA
- Dung lượng bộ nhớ Flash: 32 KB, 0.5 KB used by bootloader.
- SRAM: 2 KB



- EEPROM: 1 KB
- Tốc độ thạch anh: 16 MHz
- Arduino UNO có thể sử dụng 3 vi điều khiển họ 8bit AVR là ATmega8, ATmega168, ATmega328. Bộ não này có thể xử lý những tác vụ đơn giản như điều khiển đèn LED nhấp nháy, xử lý tín hiệu cho xe điều khiển từ xa, làm một trạm đo nhiệt độ - độ ẩm và hiển thị lên màn hình LCD,...

#### *Nguồn cho Arduino :*

Arduino UNO R3 có thể được cấp nguồn 5VDC thông qua cổng USB hoặc cấp nguồn ngoài thông qua Adaptor với điện áp khuyến dùng là 7-9VDC, thường thì nên cấp nguồn bằng pin 9V là hợp lý nhất nếu bạn không có sẵn nguồn từ cổng USB.

Nếu cấp nguồn vượt quá ngưỡng giới hạn như trên sẽ làm hỏng Arduino UNO R3.

GND (Ground): cực âm của nguồn điện cấp cho Arduino UNO. Khi bạn dùng các thiết bị sử dụng những nguồn điện riêng biệt thì những chân này phải được nối với nhau.

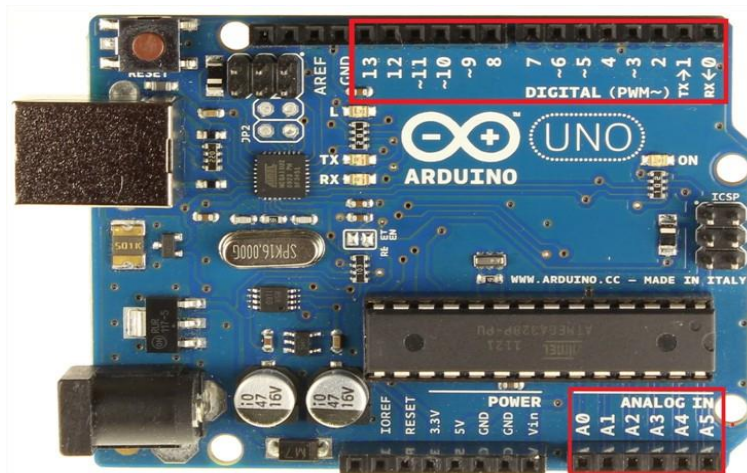
5V: cấp điện áp 5V đầu ra. Dòng tối đa cho phép ở chân này là 500mA.

3.3V: cấp điện áp 3.3V đầu ra. Dòng tối đa cho phép ở chân này là 150mA.

Vin (Voltage Input): Để cấp nguồn ngoài cho Arduino UNO, bạn nối cực dương của nguồn với chân này và cực âm của nguồn với chân GND.

RESET: Khi nhấn nút Reset trên board để Reset vi điều khiển tương đương với việc chân Reset được nối với GND qua 1 điện trở 10KΩ.

#### *Các chân vào ra của Arduino Uno R3:*



Arduino UNO có 14 chân digital dùng để đọc hoặc xuất tín hiệu. Chúng chỉ có 2 mức điện áp là 0V và 5V với dòng vào/ra tối đa trên mỗi chân là 40mA. Ở mỗi chân đều có các điện trở pull-up từ được cài đặt ngay trong vi điều khiển ATmega328 (mặc định thì các điện trở này không được kết nối).

*Một số chân digital có các chức năng đặc biệt như sau:*

2 chân Serial: 0 (RX) và 1 (TX): dùng để gửi (transmit – TX) và nhận (receive – RX) dữ liệu TTL Serial. Arduino Uno có thể giao tiếp với thiết bị khác thông qua 2 chân này. Kết nối bluetooth thường thấy nói nôm na chính là kết nối Serial không dây. Nếu không cần giao tiếp Serial, bạn không nên sử dụng 2 chân này nếu không cần thiết

Chân PWM (~): 3, 5, 6, 9, 10, và 11: cho phép bạn xuất ra xung PWM với độ phân giải 8bit (giá trị từ 0 → 255 tương ứng với 0V → 5V) bằng hàm `analogWrite()`. Nói một cách đơn giản, bạn có thể điều chỉnh được điện áp ra ở chân này từ mức 0V đến 5V thay vì chỉ cố định ở mức 0V và 5V như những chân khác.

Chân giao tiếp SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ngoài các chức năng thông thường, 4 chân này còn dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác.

LED 13: trên Arduino UNO có 1 đèn led màu cam (kí hiệu chữ L). Khi bấm nút Reset, bạn sẽ thấy đèn này nhấp nháy để báo hiệu. Nó được nối với chân số 13. Khi chân này được người dùng sử dụng, LED sẽ sáng.

Arduino UNO có 6 chân analog (A0 → A5) cung cấp độ phân giải tín hiệu 10bit (0 → 1023) để đọc giá trị điện áp trong khoảng 0V → 5V. Với chân AREF trên board, bạn có thể để đưa vào điện áp tham chiếu khi sử dụng các chân analog. Tức là

nếu bạn cấp điện áp 2.5V vào chân này thì bạn có thể dùng các chân analog để đo điện áp trong khoảng từ 0V → 2.5V với độ phân giải vẫn là 10bit.

Đặc biệt, Arduino UNO có 2 chân A4 (SDA) và A5 (SCL) hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.

## 2.4 Tìm hiểu về ESP8266



### MÔ TẢ

Kit RF thu phát Wifi ESP8266 NodeMCU Lua là kit phát triển dựa trên nền chip Wifi SoC ESP8266 với thiết kế dễ sử dụng và đặc biệt là có thể sử dụng trực tiếp trình biên dịch của Arduino để lập trình và nạp code, điều này khiến việc sử dụng và lập trình các ứng dụng trên ESP8266 trở nên rất đơn giản.

Kit RF thu phát Wifi ESP8266 NodeMCU Lua được dùng cho các ứng dụng cần kết nối, thu thập dữ liệu và điều khiển qua sóng Wifi, đặc biệt là các ứng dụng liên quan đến IoT

Kit RF thu phát Wifi ESP8266 NodeMCU Lua tại Hshop.vn sử dụng chip nạp và giao tiếp UART mới và ổn định nhất là CP2102 có khả năng tự nhận Driver trên tất cả các hệ điều hành Window và Linux, đây là phiên bản nâng cấp từ các phiên bản sử dụng IC nạp giá rẻ CH340.

IC chính: ESP8266 Wifi SoC.

Phiên bản firmware: NodeMCU Lua

Chip nạp và giao tiếp UART: CP2102.

GPIO tương thích hoàn toàn với firmware Node MCU.

Cấp nguồn: 5VDC MicroUSB hoặc Vin.

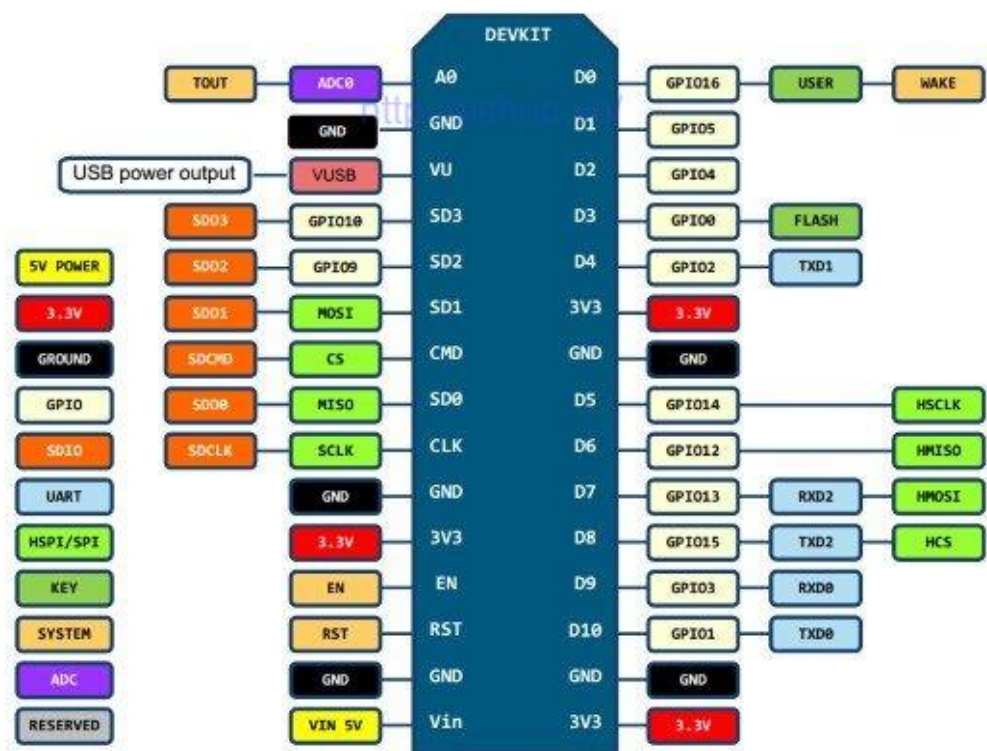
GPIO giao tiếp mức 3.3VDC

Tích hợp Led báo trạng thái, nút Reset, Flash.

Tương thích hoàn toàn với trình biên dịch Arduino.

Kích thước: 25 x 50 mm

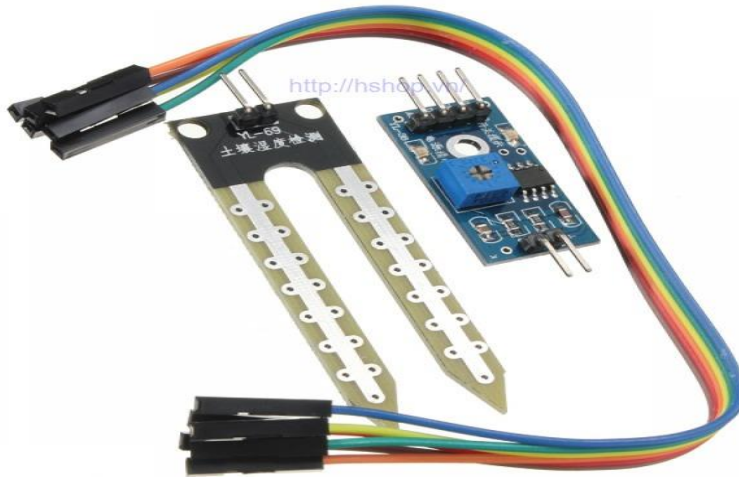
### PIN DEFINITION



D0(GPIO16) can only be used as gpio read/write, no interrupt supported, no pwm/i2c/ow supported.

## 2.5 Tìm hiểu về Sensor

### 2.5.1 Cảm biến nhiệt độ độ ẩm DHT21



#### **MÔ TẢ:**

Cảm biến độ ẩm đất thường được sử dụng trong các mô hình tưới nước tự động, vườn thông minh,..., cảm biến giúp xác định độ ẩm của đất qua đầu dò và trả về giá trị Analog, Digital qua 2 chân tương ứng để giao tiếp với Vi điều khiển để thực hiện vô số các ứng dụng khác nhau.

#### **Thông số kỹ thuật:**

- Điện áp hoạt động: 3.3~5VDC
- Tín hiệu đầu ra:
- Analog: theo điện áp cấp nguồn tương ứng.
- Digital: High hoặc Low, có thể điều chỉnh độ ẩm mong muốn bằng biến trở thông qua mạch so sánh LM393 tích hợp.
- Kích thước: 3 x 1.6cm.

Sơ đồ chân:

VCC	3.3V ~ 5V
GND	GND của nguồn ngoài
DO	Đầu ra tín hiệu số (mức cao hoặc mức thấp)
AO	Đầu ra tín hiệu tương tự (Analog)

### 2.5.2 Cảm biến độ ẩm đất



#### **Mô tả:**

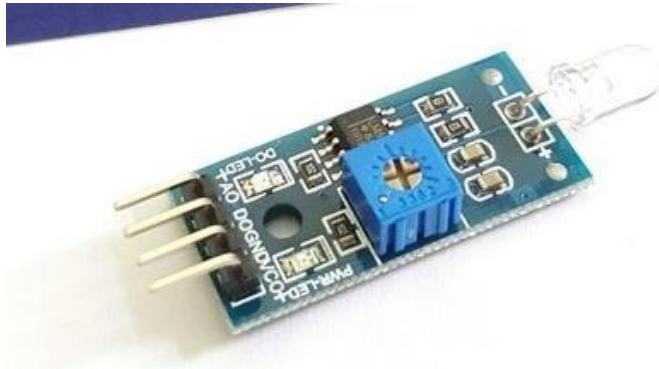
Cảm biến độ ẩm, nhiệt độ DHT21 AM2301 tích hợp cảm biến độ ẩm điện dung và cảm biến nhiệt độ có độ chính xác cao, đầu ra tín hiệu số có thể kết nối với một Vi điều khiển 8-bit, Sản phẩm chất lượng cao, đáp ứng nhanh, khả năng chống nhiễu mạnh, giao tiếp duy nhất 1 dây. Kích thước nhỏ, tiêu thụ điện năng thấp, khoảng cách truyền dẫn tín hiệu lên đến 20m. Điện năng tiêu thụ cực thấp, khoảng cách truyền dẫn, hiệu chuẩn hoàn toàn tự động, sử dụng các cảm biến độ ẩm điện dung, hoàn toàn hoán đổi cho nhau, tiêu chuẩn kỹ thuật số đầu ra duy nhất- một bus, ổn định lâu dài tuyệt vời, thiết bị đo nhiệt độ chính xác cao.

#### **Thông số kỹ thuật:**

- Áp nguồn: 3.3 - 5V
- Dòng tiêu thụ: 300 uA
- Kích thước: 58.8 x 26.7 x 13.8 (mm)
- Model: AM2301
- Độ phân giải chính xác: 0.1
- Khoảng đo: 0100% RH
- Khoảng đo nhiệt độ: -40 °C ~ 80 °C
- Đo lường chính xác độ ẩm:  $\pm 3\%$  RH
- Đo lường chính xác nhiệt độ:  $\pm 0.5$  °C



### 2.5.3 Cảm biến ánh sáng



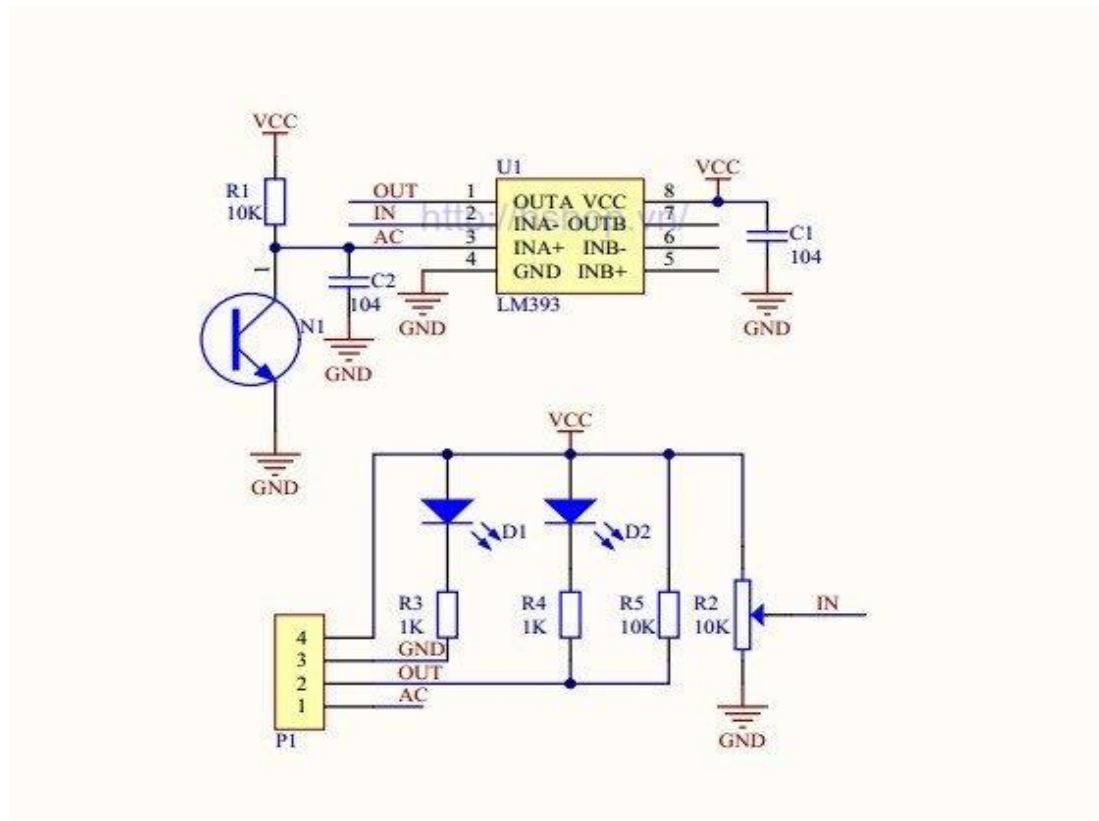
#### **Mô tả:**

Cảm biến ánh sáng Photodiode sử dụng Photodiode thay vì quang trở để cảm biến ánh sáng nên cho độ chính xác và độ ổn định cao, cảm biến có Opamp tích hợp cho cả hai ngõ ra tín hiệu là Digital và Analog rất dễ sử dụng, thích hợp cho các ứng dụng nhận biết sáng tối, cảm biến cường độ sáng,...

#### **Thông số kỹ thuật:**

- Nguồn: 3.3 -> 5VDC
- Sử dụng Photodiode cho độ chính xác cao.
- Xuất tín hiệu Digital tinh chỉnh bằng biến trở trên mạch hoặc Analog rất dễ sử dụng.
- Kích thước: 30 x 16mm

A0	Ngõ ra tín hiệu Analog
DO	Ngõ ra tín hiệu Digital
VCC	Nguồn 3.3 ~ 5VDC
GND	Mass



### 2.5.4 Cảm biến mưa



**Mô tả:**

- Raindrops và bộ điều khiển riêng biệt, dễ dàng nối dây.
- Tấm cảm biến mưa lớn, thuận lợi để phát hiện mưa.
- Board có lỗ định vị dễ dàng lắp đặt
- LM393 khoảng so sánh điện áp rộng



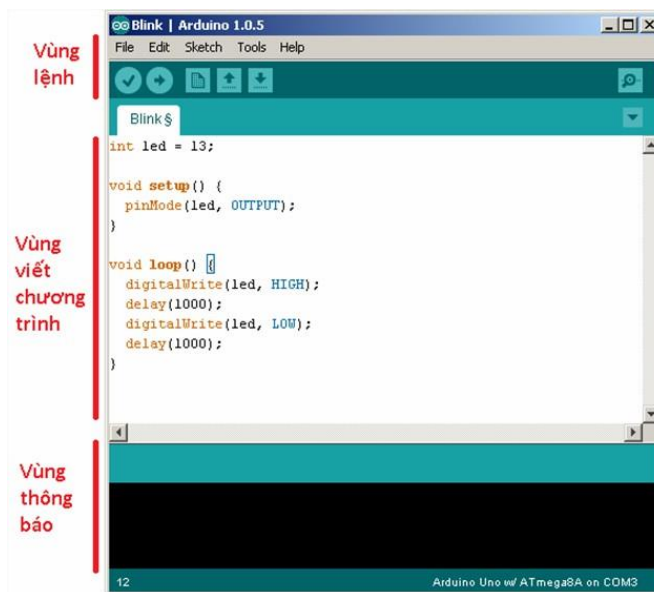
**Thông số kỹ thuật:**

- Kích thước tấm cảm biến mưa: 54 x 40mm
- Kích thước board PCB: 30 x 16mm
- Điện áp: 5V
- Đầu ra: đầu ra kỹ thuật số (0 và 1) và đầu ra tương tự điện áp A0 ;
- Lỗ cố định bu lông dễ dàng để cài đặt
- Có đèn báo hiệu nguồn và đầu ra
- Đầu ra TTL, tín hiệu đầu ra TTL có giá trị thấp. Có thể điều khiển trực tiếp relay, buzzer, a small fan...
- Độ nhạy có thể được điều chỉnh thông qua chiết áp
- LED sáng lên khi không có mưa đầu ra cao, có mưa, đầu ra thấp LED tắt.
- chế độ kết nối:
- VCC: Nguồn
- GND: Đất
- D0: Đầu ra tín hiệu TTL chuyển đổi
- A0: Đầu ra tín hiệu Analog

## 2.6 Tìm hiểu về lập trình trên Arduino IDE







Arduino IDE: Arduino Intergrated Development Environment

**Giao diện:**



### Vùng lệnh:

Bao gồm các nút lệnh menu (File, Edit, Sketch, Tools, Help). Phía dưới là các icon cho phép sử dụng nhanh các chức năng thường dùng của IDE được miêu tả như sau:

Icon	Chức năng
	Biên dịch chương trình đang soạn thảo để kiểm tra các lỗi lập trình.
	Biên dịch và upload chương trình đang soạn thảo.
	Mở một trang soạn thảo mới.
	Mở các chương trình đã lưu.
	Lưu chương trình đang soạn.
	Mở cửa sổ Serial Monitor để gửi và nhận dữ liệu giữa máy tính và board Arduino.

### Vùng viết chương trình:

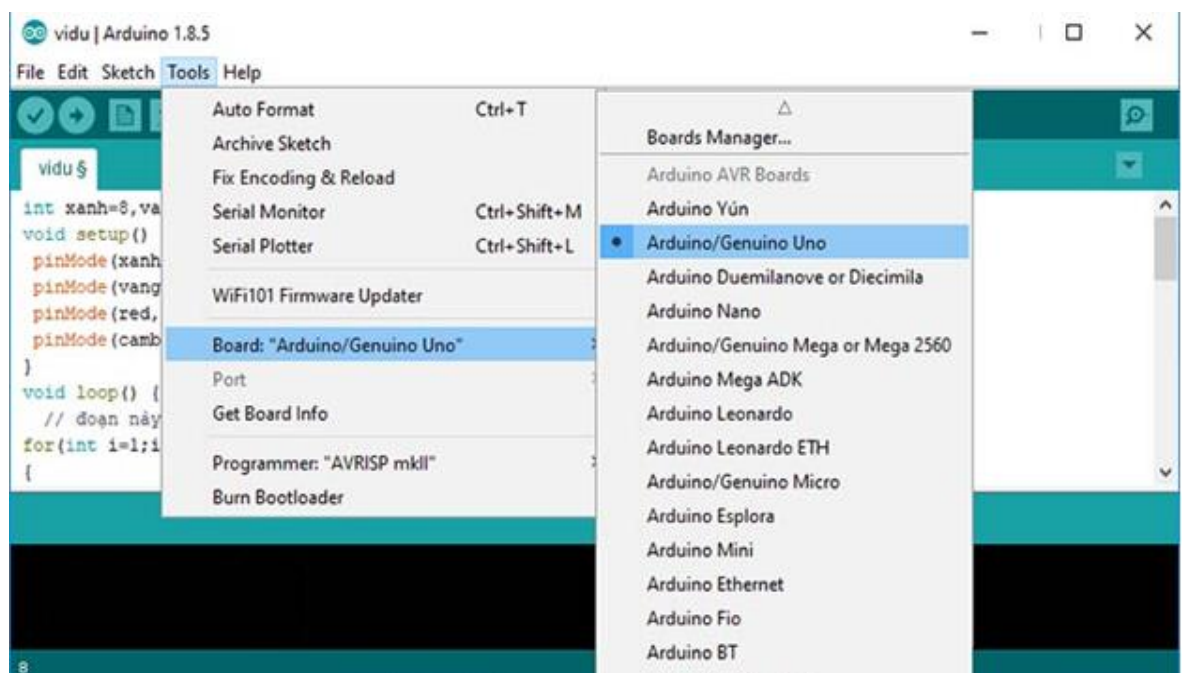
Bạn sẽ viết các đoạn mã của mình tại đây. Tên chương trình của bạn được hiển thị ngay dưới dãy các Icon, ở đây nó tên là “Blink”. Để ý rằng phía sau tên chương trình có một dấu “\$”. Điều đó có nghĩa là đoạn chương trình của bạn chưa được lưu lại.

### **Vùng thông báo (debug):**

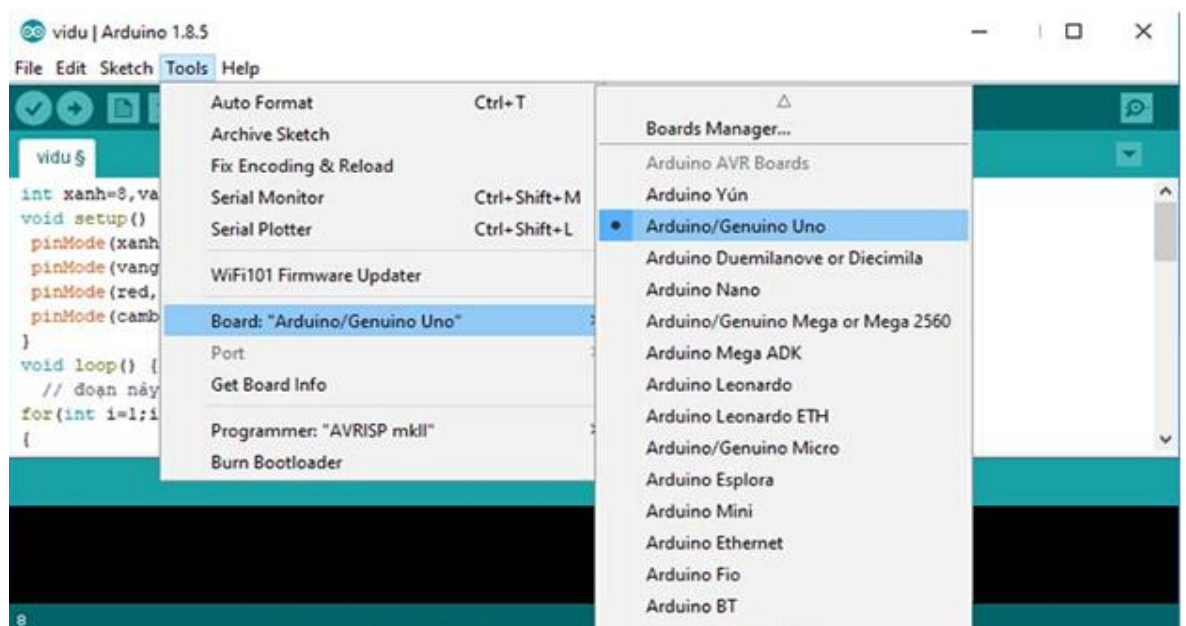
Những thông báo từ IDE sẽ được hiển thị tại đây. Để ý rằng góc dưới cùng bên phải hiển thị loại board Arduino và cổng COM được sử dụng. Luôn chú ý tới mục này bởi nếu chọn sai loại board hoặc cổng COM, bạn sẽ không thể upload được code của mình.

### **Một số Menu thông dụng:**

Phần Example (ví dụ) đưa ra các ví dụ sẵn để người lập trình có thể tham khảo, giảm bớt thời gian lập trình.



Phần mềm chọn sẵn kiểu bo là bo Arduino Uno, nếu người dùng dùng kiểu bo khác thì chọn kiểu bo đang dùng.



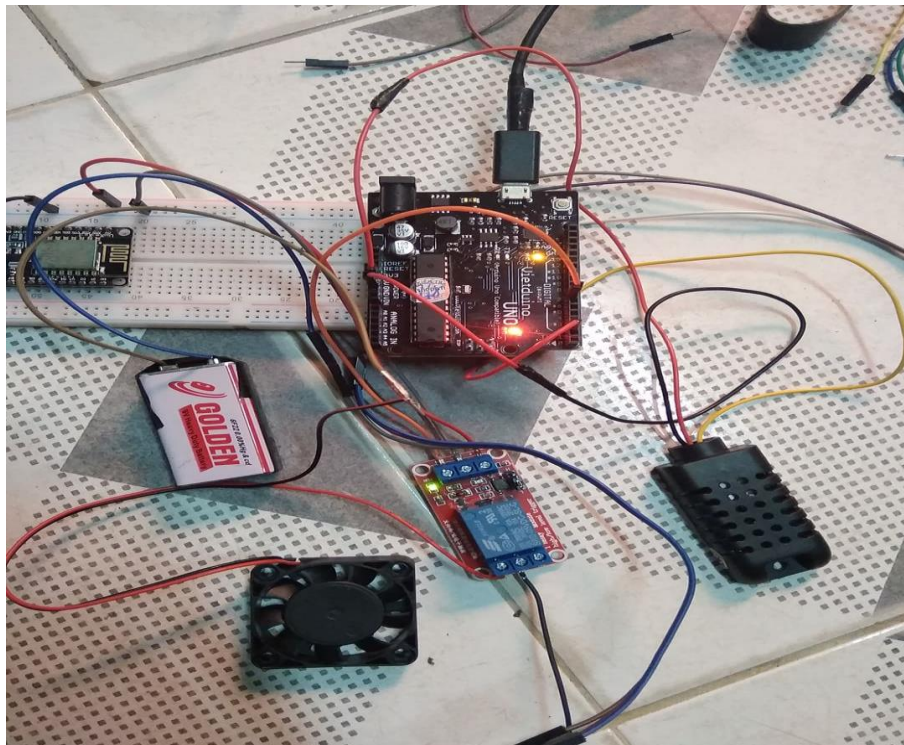
Code mẫu đèn chớp tắt 1s:

```
Blink $  
  
void setup() {  
  pinMode(13, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(13, HIGH);  
  delay(1000);  
  digitalWrite(13, LOW);  
  delay(1000);  
}
```

## 2.7 Problem-based Learning

### 2.7.1 Đo nhiệt độ độ ẩm và điều khiển quạt khi nhiệt độ cao

Mô hình:

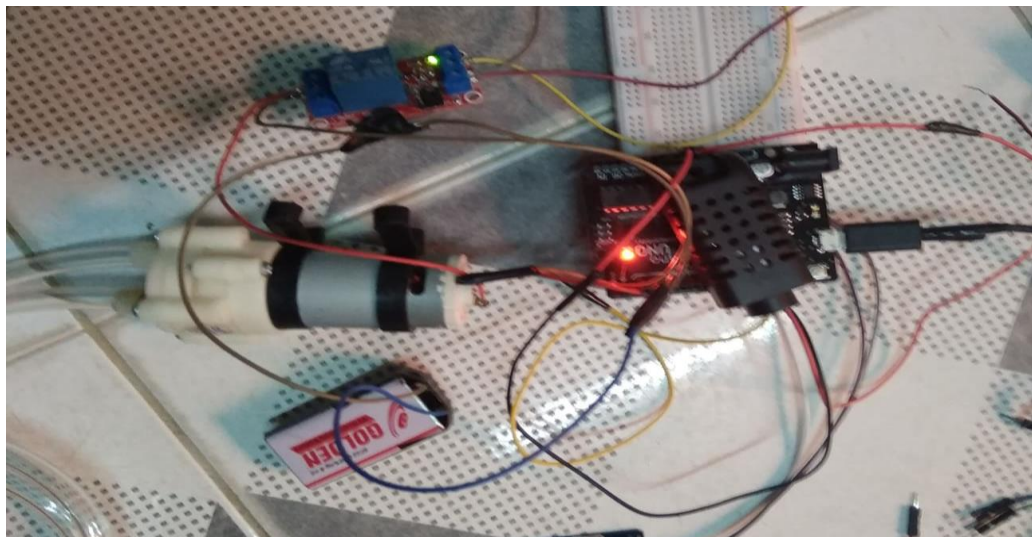


*Ứng dụng: Đo nhiệt độ và khi nhiệt độ cao thì bật quạt. Ứng dụng trong nhà thông minh hoặc điều hòa không khí trong smart garden*

*Code: [https://github.com/nguyenan97/SmartGarden/tree/master/DHT\\_Quat](https://github.com/nguyenan97/SmartGarden/tree/master/DHT_Quat)*

### **2.7.2 Đo độ ẩm đất và điều khiển máy bơm khi độ ẩm đất thấp**

*Mô hình:*



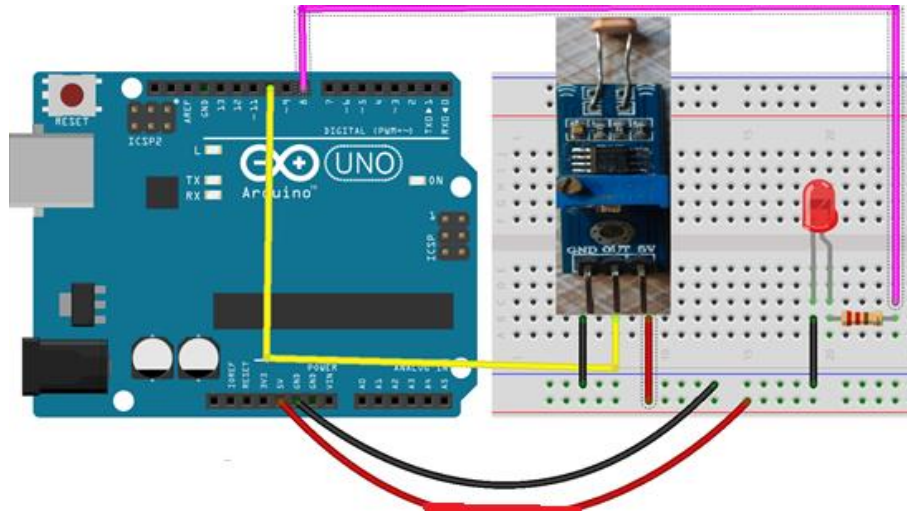
*Ứng dụng: Đo độ ẩm đất và điều khiển máy bơm ứng dụng nhiều trong chăm sóc cây tự động, ...*



Code: [https://github.com/nguyenan97/SmartGarden/tree/master/Dat\\_Bom](https://github.com/nguyenan97/SmartGarden/tree/master/Dat_Bom)

### 2.7.3 Sử dụng cảm biến ánh sáng và điều khiển bật đèn khi trời tối

Mô hình:

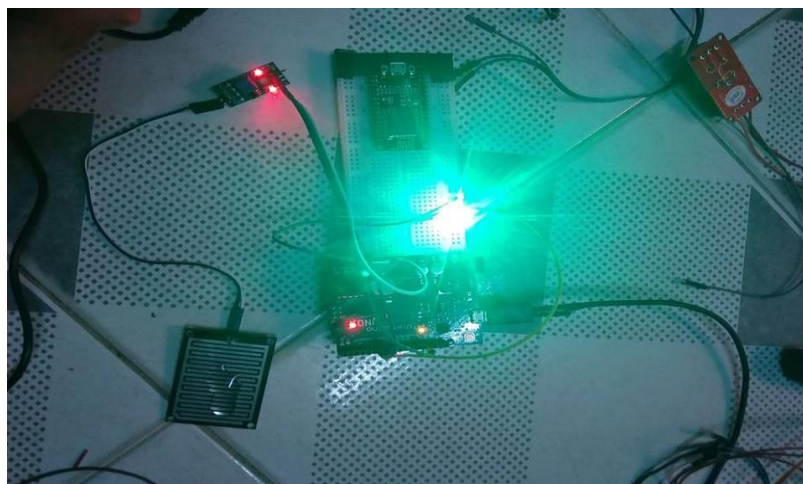


Ứng dụng: Ứng dụng nhiều trong nhà thông minh khi mà trời tối thì có thể tự bật tắt đèn hay ứng dụng trong chăm sóc cây trồng,...

Code: [https://github.com/nguyenan97/SmartGarden/tree/master/AS\\_Led](https://github.com/nguyenan97/SmartGarden/tree/master/AS_Led)

### 2.7.4 Sử dụng cảm biến mưa và điều khiển led khi trời mưa

Mô hình:

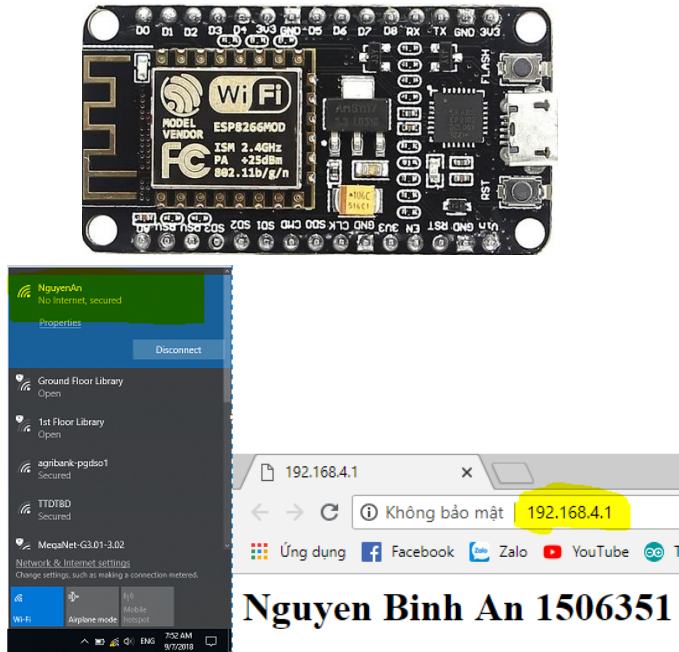


Ứng dụng: Ứng dụng nhiều trong nhà thông minh khi mà trời mưa thì có thể kéo máy che,... hoặc ứng dụng trong chăm sóc cây trồng khi trời mưa thì có thể kéo màn che lại,...

Code: [https://github.com/nguyenan97/SmartGarden/tree/master/Mua\\_Led](https://github.com/nguyenan97/SmartGarden/tree/master/Mua_Led)

### 2.7.5 Tạo một AccessPoint và chạy một web server trên AP

Mô hình:

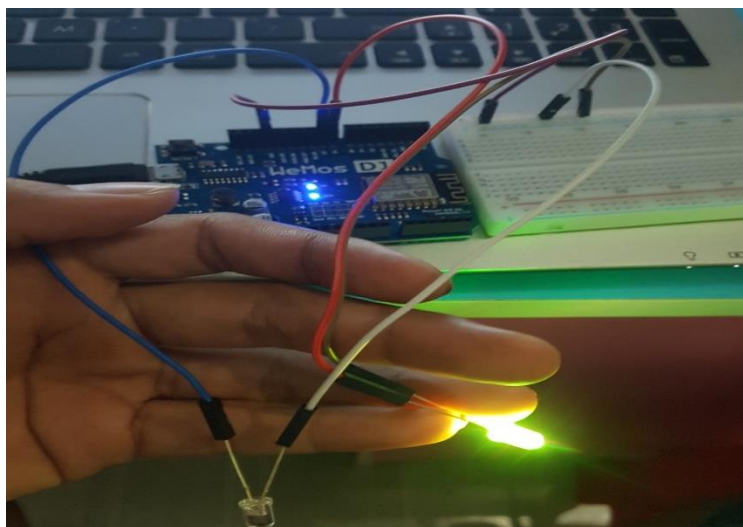


Ứng dụng: Ứng dụng lập một AP cung cấp thông tin hoặc dịch vụ hay điều khiển thiết bị bật tắt, ...

Code: <https://github.com/nguyenan97/SmartGarden/tree/master/AccessPoint>

### 2.7.6 Lập trình điều khiển 2 led thông qua web server sử dụng esp8266

Mô hình:



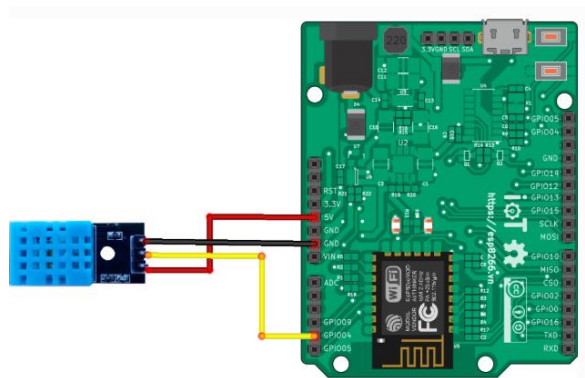
Ứng dụng: Điều khiển các thiết bị thông qua Access Point

Code:

<https://github.com/nguyenan97/SmartGarden/tree/master/DieuKhien2Led>

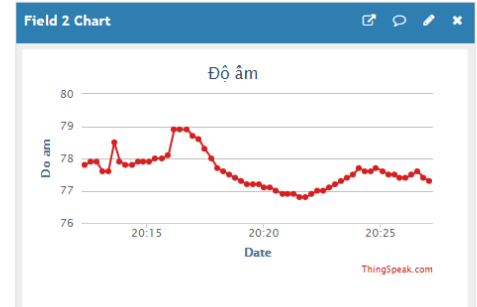
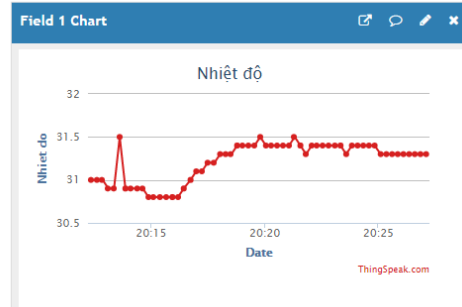
### 2.7.7 Đo nhiệt độ và độ ẩm gửi lên dịch vụ Thingspeak

Mô hình:



Channel Stats

Created: [about 2 hours ago](#)  
Updated: [about 2 hours ago](#)  
Last entry: [22 minutes ago](#)  
Entries: 1



Ứng dụng: Đo được nhiệt độ độ ẩm và gửi lên cloud để có thể giám sát nhiệt độ độ ẩm bất kì đâu chỉ cần có wifi.

Code:

[https://github.com/nguyenan97/SmartGarden/tree/master/DHT\\_Thingspeak](https://github.com/nguyenan97/SmartGarden/tree/master/DHT_Thingspeak)



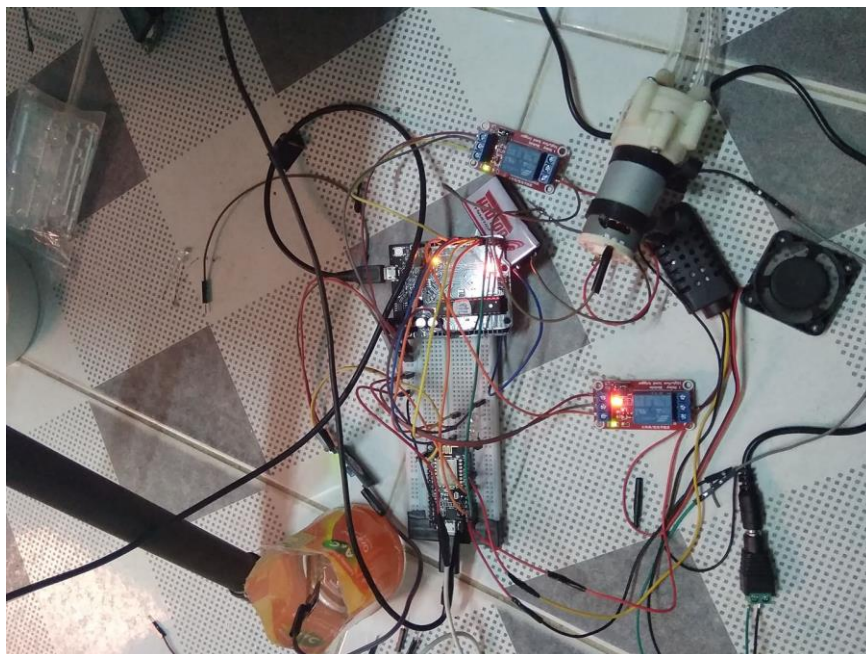
## CHƯƠNG 3: THIẾT KẾ VÀ HIỆN THỰC

### 3.1 Thiết kế mô hình

#### 3.1.1 Chuẩn bị các thiết bị

- 1 ESP8266 nodeMCU
- 1 Arduino Uno R3
- 1 Testboard
- 1 Màn hình LCD
- 4 Led
- 20 Dây đực đực
- 20 Dây đực cái
- 1 Cảm biến DHT21
- 1 Cảm biến Đất

#### 3.1.2 Kết nối các thiết bị

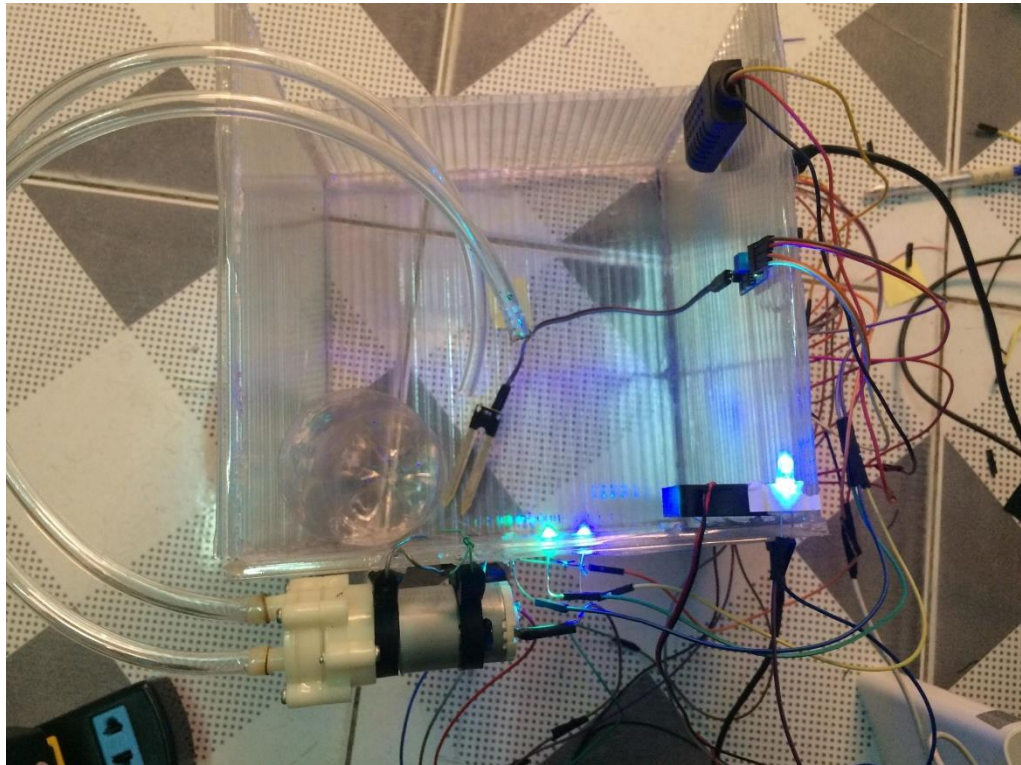


### 3.2 Chức năng của mô hình

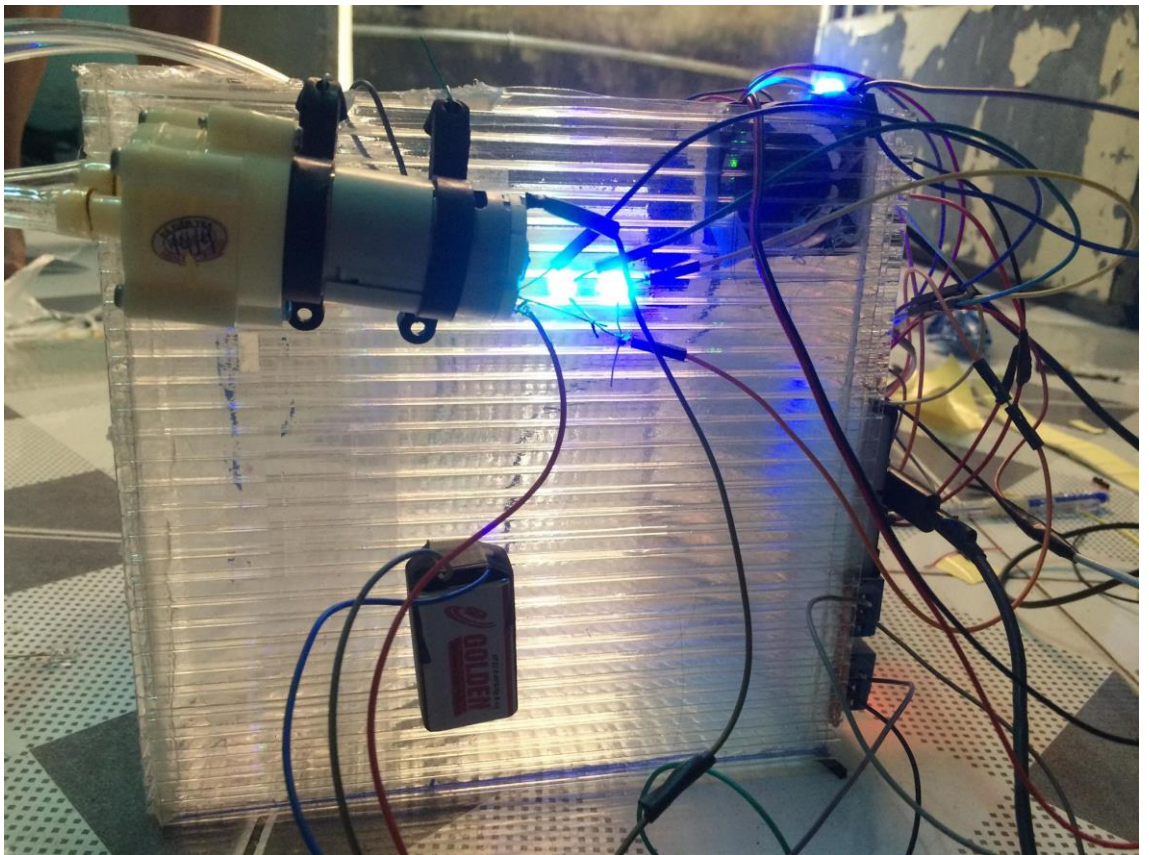
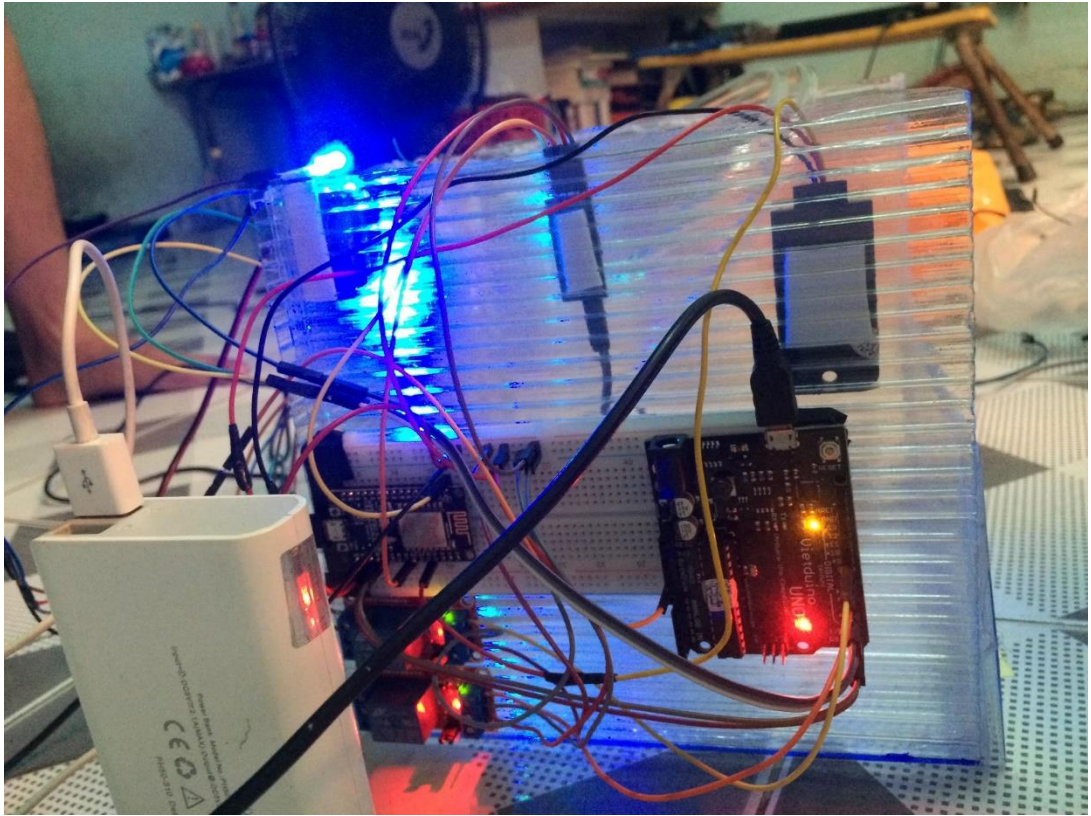
Sử dụng 3 đèn led để thông báo 3 mức độ là đất đang thiếu nước, đủ nước hay thừa nước. Nếu đang thiếu nước thì hệ thống sẽ tự động bật máy bơm để tưới cây.

Sử dụng quạt để điều hòa nhiệt độ không khí khi nhiệt độ quá cao

### 3.3 Mô hình hoàn chỉnh







## CHƯƠNG 4: KẾT LUẬN

### 4.1 Kết quả đạt được

Hoàn thành được mô hình chăm sóc cây tự động có 2 chức năng chính là:

- Đo nhiệt độ và điều khiển quạt để điều hòa không khí tự động
- Đo được độ ẩm đất và gửi dữ liệu lên thingspeak
- Tất cả dữ liệu của cảm biến được lưu trên thingspeak

### 4.2 Hạn chế của mô hình

Chi phí thiết bị cao nên còn thiếu một số chức năng

Chưa hiện thị dữ liệu trên màn hình LCD cho dễ quan sát( thay vào đó đã có đèn led thông báo)

Ứng dụng thêm phun sương để điều hòa độ ẩm trong không khí

Ứng dụng cảm biến mưa khi trời mưa có thể kéo màn che lại

Ứng dụng cảm biến ánh sáng khi trời tối có thể tự bật tắt đèn cho cây phát triển.

### 4.3 Hướng phát triển

Với mô hình chăm sóc cây thông minh chúng ta có thể nghĩ đến việc tạo ra các mô hình trồng rau sạch trong các hộ gia đình để giúp tiết kiệm thời gian, chăm sóc một cách hợp lý...có thể đem lại hiệu quả cao.

Chúng ta cũng có thể đưa hệ thống chăm sóc cây thông minh vào các dự án nông nghiệp lớn để giúp tăng năng suất, giảm nhân công, chi phí,....

Thêm một số chức năng mà đề tài chưa có như sử dụng cảm biến ánh sáng để bật tắt đèn tự động để thúc đẩy cây phát triển, sử dụng cảm biến độ ẩm để bật tắt phun sương tự động để thay đổi độ ẩm giúp tạo ra môi trường phù hợp để cây phát triển tốt nhất.

Nghiên cứu môi trường phát triển của nhiều loại cây trồng.

## TÀI LIỆU THAM KHẢO

- [1]. <https://iotmakervn.github.io/iot-starter-book/>
- [2]. <https://hshop.vn/>
- [3]. <https://arduino.esp8266.vn/projects/dht11-thingspeak.html>
- [4]. <http://arduino.vn/tags/software-serial>
- [5]. <https://github.com/hieucaotuoicay-tudong-arduino>
- [6]. [https://create.arduino.cc/projecthub/patelDeep/smart-garden-cda82f?ref=search&ref\\_id=smart%20garden&offset=0](https://create.arduino.cc/projecthub/patelDeep/smart-garden-cda82f?ref=search&ref_id=smart%20garden&offset=0)

