



BÁO CÁO ĐỒ ÁN MÔN HỌC

MÔN NHẬP MÔN KỸ THUẬT

HK1 - NĂM HỌC: 2023-2024

TÊN ĐỒ ÁN:

THIẾT BỊ BÁO CHÁY THÔNG MINH

Lớp 23DTV_CLC1- Nhóm 5

Họ tên thành viên

1	Phạm Minh Thắng	23207027
2	Trần Thành Thịnh	23207029
3	Ngô Khải Du	23207032
4	Thiệu Việt Thành	23207028
5	Lê Minh Trọng	23207030
6	Nguyễn Thanh An	23207034

TP.HCM – 11/2023

MỤC LỤC

MỤC LỤC.....	2
Mục lục hình ảnh.....	4
Mục lục bảng	5
LỜI MỞ ĐẦU.....	6
Chương 1 GIỚI THIỆU VỀ ARDUINO UNO	7
1.1 Đặt vấn đề	7
1.2 Mạch vi điều khiển Arduino.....	7
1.3 Cách kết nối	10
1.4 Giới thiệu về Arduino IDE và ngôn ngữ lập trình cho Arduino	10
1.4.1 Arduino IDE:.....	11
1.4.2 Ngôn ngữ lập trình cho Arduino	11
Chương 2 Giới thiệu cảm biến quang trở sử dụng IC LM393	12
2.1 Cảm biến quang trở là gì?	12
2.2 IC LM393 là gì?	12
2.3 Thông số kĩ thuật.....	13
2.4 Cấu trúc cảm biến	13
2.5 Giao tiếp với cảm biến quang trở.....	14
2.6 Kết nối với Arduino UNO	14
Chương 3 Giới thiệu module cảm biến nhiệt DHT11	15
3.1 DHT11 là gì?	15
3.2 Thông số kĩ thuật.....	15
3.4 Giao tiếp với DHT11	16
3.5 Cài đặt thư viện cho DHT11	16
Chương 4 Giới thiệu module cảm biến khí gas MQ-5.....	18
4.2 Thông số kĩ thuật.....	18
4.3 Cấu trúc cảm biến khí hoá lỏng, khí than MQ-5.....	18
4.4 Giao tiếp với MQ-5	19
4.5 Kết nối với Arduino UNO	19
Chương 5 Buzzer 5VDC	20

5.1 Buzzer là gì ? (TMB12A05)	20
5.2 Thông số buzzer được lắp đặt. (TMB12A05).....	20
5.3 Cấu trúc buzzer (TMB12A05)	20
5.4 Giao tiếp buzzer	21
5.5 Kết nối buzzer với Arduino Uno	21
Chương 6 ESP8266 NodeMCU Lua V3	22
6.1 ESP8266 là gì?	22
6.2 Thông số kĩ thuật.....	22
6.3 Cấu trúc	23
6.4 Giao tiếp.....	23
6.5 Kết nối.....	24
6.6 Cài đặt thư viện cho ESP8266	24
Chương 7 Giới thiệu ứng dụng Blynk.....	25
7.1 Blynk là gì ?	25
7.2 Giao tiếp với Blynk.....	25
7.3 Cài đặt thư viện cho Blynk	28
Chương 8 Sơ đồ giải thuật.....	29
8.1. Chuẩn bị	29
8.2 Code chương trình.....	29
8.3 Triển khai sản phẩm	32
8.4 Sơ đồ giải thuật	33
BÁO CÁO KỸ NĂNG	35
1. Kỹ năng làm việc nhóm và quản lý dự án.....	35
2. Tự đánh giá các kỹ năng/kiến thức	36
TÀI LIỆU THAM KHẢO	37

Mục lục hình ảnh

Hình 1 Board mạch Arduino	7
Hình 2 Bảng mạch điều khiển Arduino.....	8
Hình 3 Sơ đồ chân của Arduino UNO	10
Hình 4 Cảm biến quang trở	13
Hình 5 Sơ đồ chân của cảm biến quang trở	13
Hình 6 Cách kết nối của Arduino UNO	14
Hình 7 Cảm biến nhiệt DHT11	15
Hình 8 Cấu tạo của cảm biến nhiệt DHT11	15
Hình 9 Hướng dẫn tìm kiếm và cài đặt thư viện DHT từ Adafruit.....	16
Hình 10 Hướng dẫn tìm kiếm và cài đặt thư viện DHT từ Adafruit.....	17
Hình 11 Sơ đồ kết nối mạch Arduino UNO	17
Hình 12 Cảm biến khí MQ 5.....	18
Hình 13 Cấu trúc cảm biến MQ-5.....	18
Hình 14 Sơ đồ kết nối chân của MQ-5 với Arduino UNO	19
Hình 15 Cấu trúc buzzer	20
Hình 16 Sơ đồ nguyên lí buzzer.....	21
Hình 17 Kết nối chân	22
Hình 18 Cấu trúc ESP8266	23
Hình 19 Hướng dẫn cài đặt thư viện cho ESP8266	24
Hình 20 Đăng nhập/Đăng kí	25
Hình 21 Tạo mật khẩu.....	25
Hình 22 Copy Template ID và Device Name	26
Hình 23 Nhập datastream.....	26
Hình 24 Lưu cài đặt.....	27
Hình 25 Thêm thiết bị	28
Hình 26 Kiểm tra sản phẩm	32
Hình 27 Liên kết lên app Blynk	32
Hình 28 Sơ đồ giải thuật Arduino	33
Hình 29 Sơ đồ giải thuật ESP8266	34

Mục lục bảng

Table 1 Bảng thông số kỹ thuật của Arduino UNO	9
Table 2Bảng thông số kỹ thuật của Arduino UNO	16
Table 3 Bảng chân kết nối Arduino UNO	17
Table 4 Các kết nối chân tương ứng	19

LỜI MỞ ĐẦU

Ngày nay trước những sự phát triển như vũ bão của khoa học kỹ thuật việc áp dụng khoa học công nghệ vào trong thực tế sản xuất đang được phát triển rộng rãi về mặt quy mô lẫn chất lượng. Trong đó ngành điện tử hóa chiếm một vai trò rất quan trọng không những giảm nhẹ sức lao động cho con người mà còn góp phần rất lớn trong việc nâng cao năng suất lao động, cải thiện chất lượng sản phẩm, chính vì thế ngành điện tử hóa ngày càng khẳng định được vị trí cũng như vai trò của mình trong các ngành công nghiệp và đang được phổ biến rộng rãi trong các hệ thống công nghiệp trên toàn thế giới nói chung và Việt Nam nói riêng. Xuất phát từ thực tế đó, trong quá trình học tập tại trường Đại Học Khoa Học Tự Nhiên được sự chỉ bảo và hướng dẫn tận tình của các thầy, giảng viên và trợ giảng trong khoa Điện Tử-Viễn Thông và đặc biệt là thầy giáo, Lê Đức Hùng, em đã nhận được đề án với đề tài: “ Thiết kế thước đo bằng cảm biến trên bo mạch Adruino ”.Để giúp cho sinh viên có thêm được những hiểu biết về vấn đề này.

Chương 1 GIỚI THIỆU VỀ ARDUINO UNO

1.1 Đặt vấn đề

Ngày nay, ngành vi điều khiển và kỹ thuật số đã chứng kiến sự tự động hóa ngày càng mạnh mẽ. Với sự áp dụng của các công nghệ tiên tiến như vi xử lý và vi mạch số trong lĩnh vực điều khiển, các hệ thống cơ khí thô sơ, với tốc độ xử lý thấp và độ chính xác hạn chế, đã được thay thế bằng các hệ thống điều khiển tự động với các lệnh chương trình được thiết lập trước.

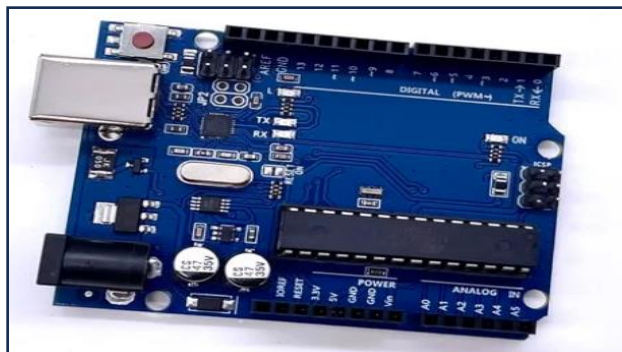
Trong quá trình vận hành của một cỗ máy, vi điều khiển đóng vai trò quan trọng. Tuy nhiên, để đảm bảo máy hoạt động hiệu quả, các thiết bị ngoại vi đầu vào và đầu ra như cảm biến, nút nhấn, LCD, màn hình, bàn phím, van điện, và các thành phần khác cũng là không thể thiếu.

Để hiểu rõ hơn về vi điều khiển từ khía cạnh cơ bản đến cách thức hoạt động và bảo trì, nhóm của chúng tôi đã quyết định thực hiện một đồ án trên mô hình hệ thống báo cháy.

1.2 Mạch vi điều khiển Arduino

- Đã bao giờ bạn muốn tạo ra robot, đèn nhấp nháy theo nhạc, làm hiệu ứng đèn LED nhấp nháy trên các biển quảng cáo,... hay có thể là máy in 3D, máy bay không người lái,... Bạn biết về lập trình nhưng do không có nhiều thời gian và thiếu những kiến thức về điện tử nên gặp rất nhiều khó khăn. Từ những trở ngại đó đã tạo nên sự ra đời một thiết bị tên là Arduino loại bỏ những trở ngại đó giúp bạn.

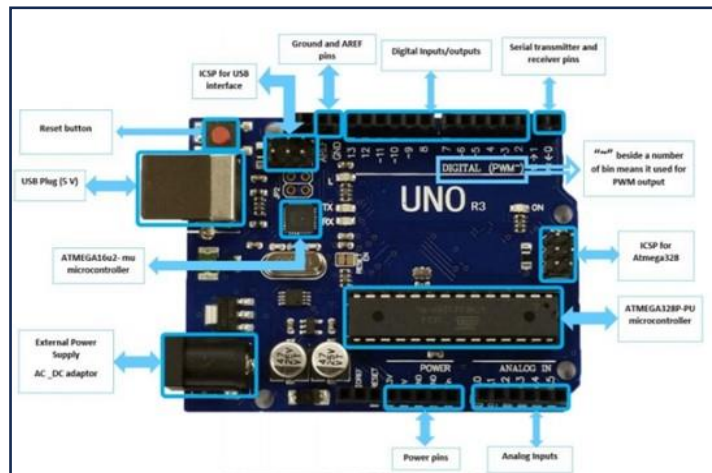
- Arduino Uno là một board mạch vi điều khiển được phát triển bởi Arduino.cc, một nền tảng điện tử mã nguồn mở chủ yếu dựa trên vi điều khiển AVR Atmega328P. Với Arduino chúng ta có thể xây dựng các ứng dụng điện tử tương tác với nhau thông qua phần mềm và phần cứng hỗ trợ.



Hình 1 Board mạch Arduino

- Khi arduino chưa ra đời, để làm được một dự án điện tử nhỏ liên quan đến lập trình, biên dịch, chúng ta cần đến sự hỗ trợ của các thiết bị biên dịch khác để hỗ trợ. Ví dụ như, dùng Vi điều khiển PIC hoặc IC vi điều khiển họ 8051..., chúng ta phải thiết kế chân nạp onboard, hoặc mua các thiết bị hỗ trợ nạp và biên dịch như mạch nạp 8051, mạch nạp PIC...

- Hiện nay Arduino được biết đến ở Việt Nam rất rộng rãi. Từ học sinh trung học, đến sinh viên và người đi làm. Những dự án nhỏ và lớn được thực hiện một cách rất nhanh, các mã nguồn mở được chia sẻ nhiều trên diễn đàn trong nước và nước ngoài. Giúp ích rất nhiều cho những bạn theo đam mê nghiên cứu chế tạo những sản phẩm có ích cho xã hội.



Hình 2 Bảng mạch điều khiển Arduino

- Trong những năm qua, Arduino là bộ não cho hàng ngàn dự án điện tử lớn nhỏ, từ những sản phẩm ra đời ứng dụng đơn giản trong cuộc sống đến những dự án khoa học phức tạp.

Table 1 Bảng thông số kỹ thuật của Arduino UNO

Vi điều khiển	ATmega328
Điện áp hoạt động	5V(cấp qua cổng usb)
Điện áp khuyến nghị	6-9V
Số chân digital I/O	14 chân(6 chân PWM)
Số chân analog	6 chân
Dòng ra tối đa trên mỗi chân I/O	30 mA
Dòng ra tối đa (5V)	500 mA
Dòng ra tối đa(3.3V)	50 mA
Bộ nhớ Flash	32 KB (ATmega328) với 0.5 KB dùng bởi bootloader
SRAM	2 KB(ATmega328)
EEPROM	1 KB(ATmega328)
Giao động của thạch anh	16 MHz

1.3 Cách kết nối

Arduino Uno có 14 chân digital dùng để đọc ghi dữ liệu. Chúng chỉ hoạt động ở 2 mức điện áp 0V và 5V với các dòng vào/ra tối đa trên mỗi chân là 40 mA.

Một số chân digital có chức năng đặc biệt như:

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 các thiết bị khác thông qua 2 chân này, như gắn thêm màn hình LCD để hiển thị.

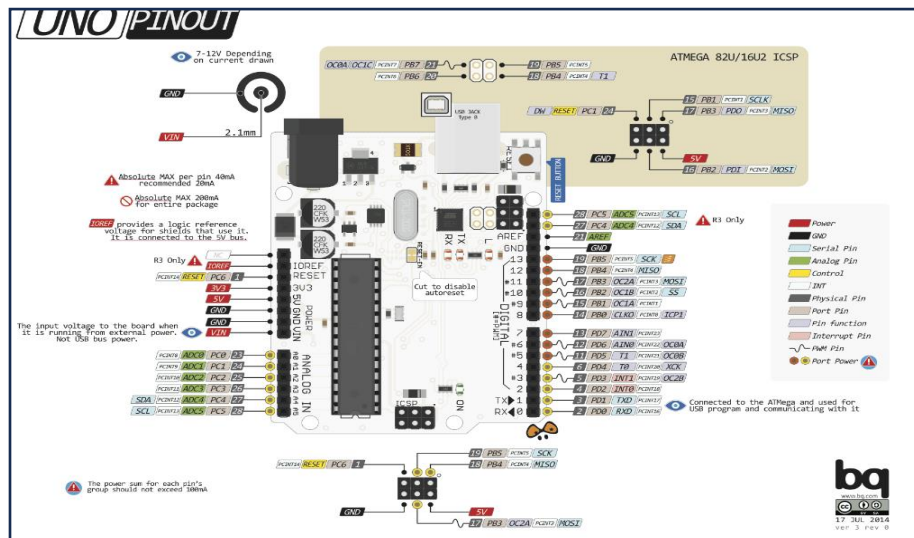
Chân PWM: 3, 5, 6, 9, 10 và 11: Cho phép bạn xuất xung PWM với độ phân giải 8 bit (giá trị từ 0 -> 2^8-1 tương ứng với 0 - 5V).

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

LED 13: Trên arduino có 1 đèn led, khi bấm nút reset thì đèn led này sẽ nhấp nháy để báo hiệu. Nó được nối với chân số 13. Khi chân này được sử dụng, đèn led sẽ sáng.

Arduino Uno R3 có 6 chân analog(A0 -> A5) cung cấp độ phân giải 10 bit($0 \rightarrow 2^{10}-1$) để đọc giá trị điện áp trong khoảng 0 -> 5V.

Arduino Uno còn có 2 chân A4(SDA) và A5(SCL) để hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.



Hình 3 Sơ đồ chân của Arduino UNO

1.4 Giới thiệu về Arduino IDE và ngôn ngữ lập trình cho Arduino

Arduino IDE (Integrated Development Environment) là môi trường phần mềm được thiết kế đặc biệt để lập trình và nạp chương trình cho các board Arduino. Nó cung

cấp một giao diện đơn giản và dễ sử dụng cho việc phát triển ứng dụng điện tử trên nền tảng Arduino. Dưới đây là một giới thiệu về Arduino IDE và ngôn ngữ lập trình cho Arduino:

1.4.1 Arduino IDE:

Giao diện:

- Giao diện của Arduino IDE rất đơn giản và thân thiện với người mới bắt đầu.
- Gồm các thanh công cụ và cửa sổ cho việc viết mã, xem lỗi, và nạp chương trình vào board Arduino.

Editor mã nguồn:

- Cung cấp một trình soạn thảo mã nguồn với các tính năng như làm nổi bật cú pháp, tự động hoàn thành mã, và gợi ý từ khóa.

Chương trình mẫu:

- Arduino IDE đi kèm với nhiều chương trình mẫu (sketches) để giúp người dùng hiểu cách sử dụng các chức năng cơ bản của board.

Upload và Serial Monitor:

- Chức năng "Upload" cho phép bạn nạp chương trình đã viết vào board Arduino.
- Serial Monitor giúp theo dõi và giao tiếp qua cổng Serial (UART) giữa Arduino và máy tính.

Thư viện:

- Arduino IDE có thư viện lớn với nhiều hàm được viết sẵn giúp bạn thực hiện nhanh chóng nhiều chức năng khác nhau.

1.4.2 Ngôn ngữ lập trình cho Arduino

Ngôn ngữ chính được sử dụng trong Arduino IDE là một biến thể của ngôn ngữ C/C++ được tối ưu hóa để làm cho việc lập trình trở nên đơn giản và dễ hiểu, đặc biệt là đối với người mới bắt đầu. Dưới đây là một số đặc điểm.

1. Chương trình cơ bản (Sketch):

- Mọi chương trình trong Arduino được gọi là "sketch" và bao gồm hai hàm chính: ``setup()`` và ``loop()``.
- Hàm ``setup()`` chạy một lần khi board được khởi động, trong khi hàm ``loop()`` chạy liên tục.

2. Biến và Kiểu dữ liệu:

- Sử dụng biến để lưu trữ dữ liệu và kiểu dữ liệu như int, float, char, để định nghĩa loại dữ liệu của biến.

3. Thư viện:

- Sử dụng thư viện để tái sử dụng mã nguồn và thêm các chức năng mở rộng.

4. Hàm và Cú pháp:

- Cú pháp giống với ngôn ngữ C/C++, nhưng có một số chức năng và hàm được đặc biệt hóa cho việc lập trình nhúng.

5. Thư viện và Hàm Built-in:

- Arduino IDE cung cấp nhiều thư viện và hàm built-in để giúp dễ dàng thực hiện các nhiệm vụ phổ biến như điều khiển động cơ, đọc cảm biến, và giao tiếp với các thiết bị ngoại vi.

Arduino IDE và ngôn ngữ lập trình của nó cung cấp một nền tảng mạnh mẽ và thân thiện với người dùng để thực hiện nhanh chóng các dự án điện tử.

Chương 2 Giới thiệu cảm biến quang trở sử dụng IC LM393

2.1 Cảm biến quang trở là gì?

Cảm biến quang trở, hay còn được gọi là cảm biến ánh sáng hoặc cảm biến quang, là một thiết bị dùng để đo lường mức độ ánh sáng. Cảm biến này thường sử dụng các vật liệu có khả năng thay đổi điện trở dưới tác động của ánh sáng. Khi ánh sáng chiếu vào cảm biến, điện trở của vật liệu đó thay đổi, và thông số này được đo để xác định mức độ ánh sáng mà cảm biến đang nhận được.

2.2 IC LM393 là gì?

IC LM393 là một loại IC (Integrated Circuit) thường được sử dụng như một bộ so sánh điện áp. LM393 là tên mã của một dòng bộ so sánh hai đầu vào, nơi mà nó so sánh hai tín hiệu điện áp và đưa ra một tín hiệu kết quả dựa trên kết quả so sánh đó.

IC LM393 thường có 8 chân (pin). Dưới đây là mô tả chức năng của các chân trên IC LM393:

Chân 1 (OUT): Ngõ ra - Kết quả so sánh được đưa ra từ IC.

Chân 2 (V-): Nguồn âm - Kết nối với mức đất (ground).

Chân 3 (V+): Nguồn dương - Nguồn cấp điện dương.

Chân 4 (IN-): Đầu vào âm - Đầu vào inverting (đầu vào mà tín hiệu được so sánh với).

Chân 5 (IN+): Đầu vào dương - Đầu vào non-inverting (đầu vào mà tín hiệu được so sánh với).

Chân 6 (Vcc): Nguồn cấp điện - Điện áp cung cấp cho IC.

Chân 7 (NC): Not Connected - Thường không được kết nối và được để trống.

Chân 8 (GND): Mặt đất - Kết nối với mức đất (ground).

2.3 Thông số kĩ thuật

Sử dụng cảm biến photoresistor loại nhạy

Dùng IC so sánh LM393 cho dòng ra lớn đến 15mA.

Nguồn cấp 3.3~5 VDC

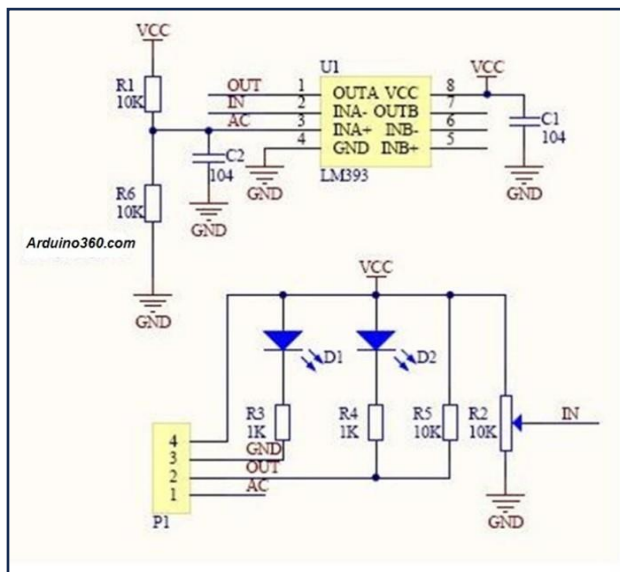
Đầu ra digital DO (0 và 1)

Đầu ra điện áp analog AO

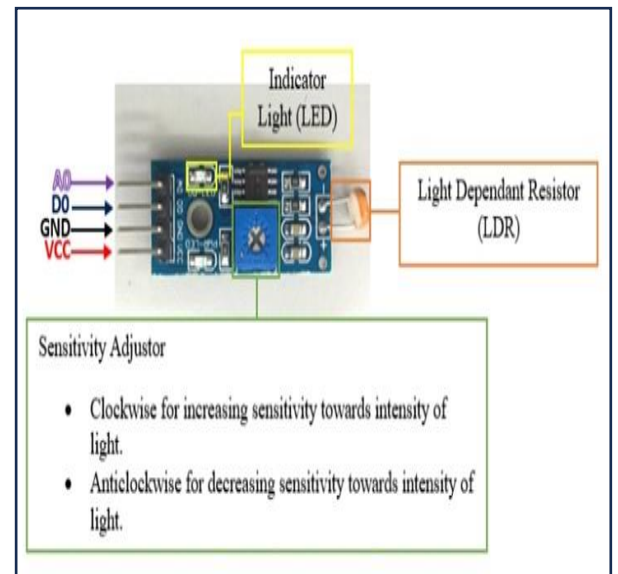
Một lỗ bu lông cố định để lắp đặt dễ dàng

Kích thước: 3.2 x 1.4cm

2.4 Cấu trúc cảm biến



Hình 5 Sơ đồ chân của cảm biến quang trở



Hình 4 Cảm biến quang trở

2.5 Giao tiếp với cảm biến quang trở

Kết nối vật lý:

- Kết nối chân đất (GND) của LM393 với mặt đất (GND) của nguồn cấp điện.
- Kết nối chân nguồn dương (Vcc) của LM393 với nguồn cấp điện dương.
- Kết nối chân đầu vào âm (IN-) của LM393 với một điều kiện cần giám sát hoặc đo lường (ví dụ: cảm biến ánh sáng).

Đọc dữ liệu:

- Kết nối chân đầu vào dương (IN+) của LM393 với nguồn cấp tín hiệu mà bạn muốn so sánh (ví dụ: mức độ ánh sáng).
- Kết nối chân ngõ ra (OUT) của LM393 với linh kiện hoặc mạch điều khiển cần kích hoạt dựa trên kết quả so sánh.

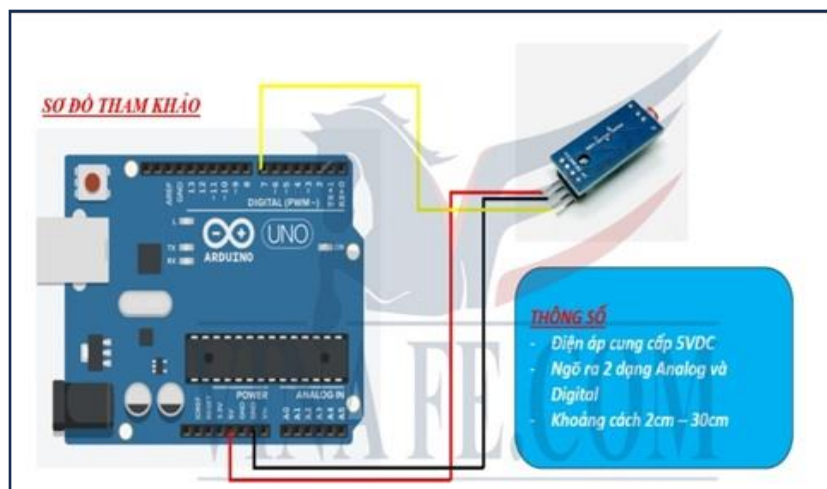
Đọc kết quả:

- Đọc tín hiệu từ chân ngõ ra (OUT) để biết kết quả của so sánh.
- Nếu tín hiệu ở chân ngõ ra ở mức cao, điều kiện so sánh là đúng (ví dụ: mức độ ánh sáng đủ lớn).
- Nếu tín hiệu ở chân ngõ ra ở mức thấp, điều kiện so sánh là sai (ví dụ: mức độ ánh sáng không đủ).

Thiết lập ngưỡng so sánh:

- Đối với mạch sử dụng LM393, bạn có thể cần điều chỉnh ngưỡng so sánh (threshold) bằng cách thay đổi điện áp đầu vào cho chân IN+.

2.6 Kết nối với Arduino UNO



Hình 6 Cách kết nối của Arduino UNO

Chương 3 Giới thiệu module cảm biến nhiệt DHT11

3.1 DHT11 là gì?

- DHT11 là một cảm biến kỹ thuật số giá rẻ để cảm nhận nhiệt độ và độ ẩm. Để đo không khí xung quanh, cảm biến này sử dụng một điện trở nhiệt và một cảm biến độ ẩm điện dung. Cảm biến này có thể dễ dàng giao tiếp với bất kỳ bộ vi điều khiển vi nào như Arduino, Raspberry Pi,... để đo độ ẩm và nhiệt độ ngay lập tức.

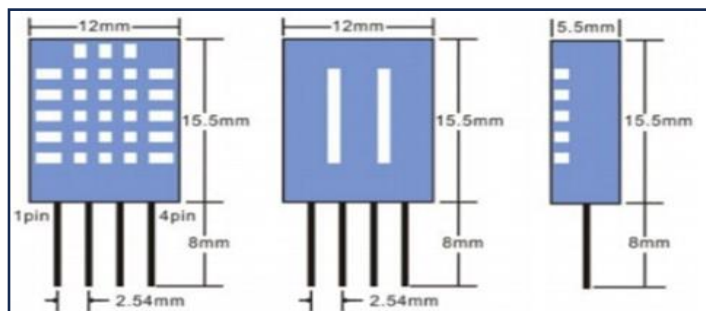


Hình 7 Cảm biến nhiệt DHT11

3.2 Thông số kỹ thuật

- Điện áp hoạt động: 3V - 5V DC
- Dòng điện tiêu thụ: 2.5mA
- Phạm vi cảm biến độ ẩm: 20% - 90% RH, sai số $\pm 5\%RH$
- Phạm vi cảm biến nhiệt độ: $0^{\circ}C \sim 50^{\circ}C$, sai số $\pm 2^{\circ}C$
- Tần số lấy mẫu tối đa: 1Hz
- Kích thước: 23 x 12 x 5 mm

3.3 Cấu trúc cảm biến nhiệt DHT11



Hình 8 Cấu tạo của cảm biến nhiệt DHT11

- Cảm biến DHT11 bao gồm một phần tử cảm biến độ ẩm điện dung và một điện trở nhiệt để cảm nhận nhiệt độ. Tự điện cảm biến độ ẩm có hai điện cực với chất nền giữ ẩm làm chất điện môi giữa chúng. Thay đổi giá trị điện dung xảy ra với sự thay đổi của các mức độ ẩm. IC đo, xử lý các giá trị điện trở đã thay đổi này và chuyển chúng thành dạng kỹ thuật số.

- Để đo nhiệt độ, cảm biến này sử dụng một nhiệt điện trở có hệ số nhiệt độ âm, làm giảm giá trị điện trở của nó khi nhiệt độ tăng. Để có được giá trị điện trở lớn hơn ngay cả đối với sự thay đổi nhỏ nhất của nhiệt độ, cảm biến này thường được làm bằng gốm bán dẫn hoặc polymer.

- Sơ đồ chân DHT11:

Table 2 Bảng thông số kỹ thuật của Arduino UNO

	Tên chân	Mô tả
1	VCC	Nguồn 3.5V đến 5.5V
2	Data	Đầu ra cả nhiệt độ và độ ẩm thông qua dữ liệu nối tiếp
3	NC	Không có kết nối → Không sử dụng
4	Ground	Nối đất

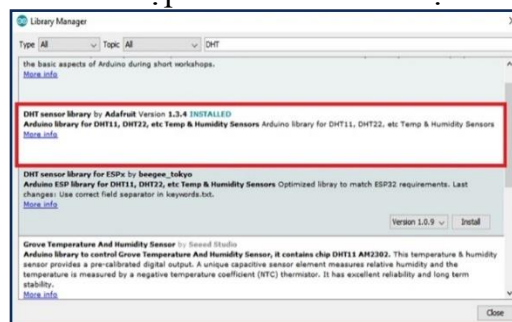
3.4 Giao tiếp với DHT11

- **Giao tiếp DHT11 với Arduino:** Có các thư viện được tạo sẵn sẽ giúp bắt đầu nhanh chóng.

3.5 Cài đặt thư viện cho DHT11

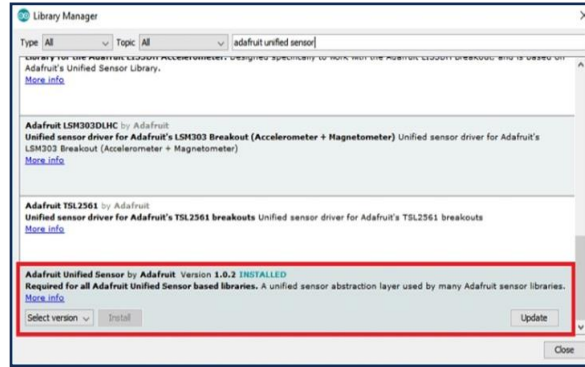
- Để đọc từ cảm biến DHT11, ta sử dụng thư viện DHT từ Adafruit. Để sử dụng thư viện này, ta cần phải cài đặt thư viện Adafruit Unified Sensor.

Bước 1: Tìm kiếm “DHT” trên hộp tìm kiếm và cài đặt thư viện DHT từ Adafruit.



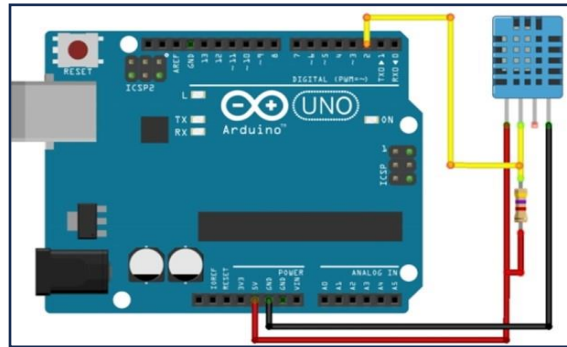
Hình 9 Hướng dẫn tìm kiếm và cài đặt thư viện DHT từ Adafruit

Bước 2: Sau khi cài đặt xong, hãy nhập “Adafruit Unified Sensor” vào hộp tìm kiếm và cài đặt nó.



Hình 10 Hướng dẫn tìm kiếm và cài đặt thư viện DHT từ Adafruit

3.6 Kết nối với ARDUINO UNO R3



Hình 11 Sơ đồ kết nối mạch Arduino UNO

Các kết nối (từ trái sang phải):

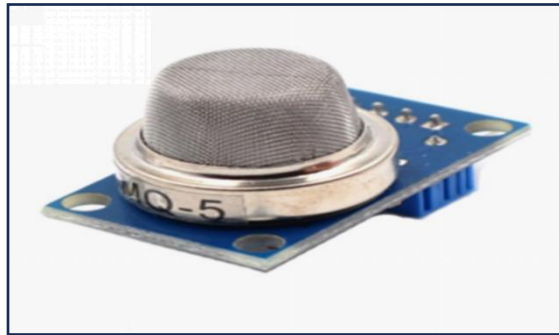
Table 3 Bảng chân kết nối Arduino UNO

DHT11	Arduino
Pin 1	5V
Pin 2	D2 hoặc bất kỳ chân kỹ thuật số nào khác
Pin 3	Không kết nối
Pin 4	GND

Chương 4 Giới thiệu module cảm biến khí gas MQ-5

4.1 MQ-5 là gì-

Cảm biến khí hoá lỏng, khí than MQ-5 là một cảm biến khí có khả năng phát hiện khí LPG, khí thiên nhiên, khí than và được ứng dụng trong các thiết bị phát hiện sự rò rỉ gas trong dân dụng và cả trong công nghiệp.



Hình 12 Cảm biến khí MQ 5

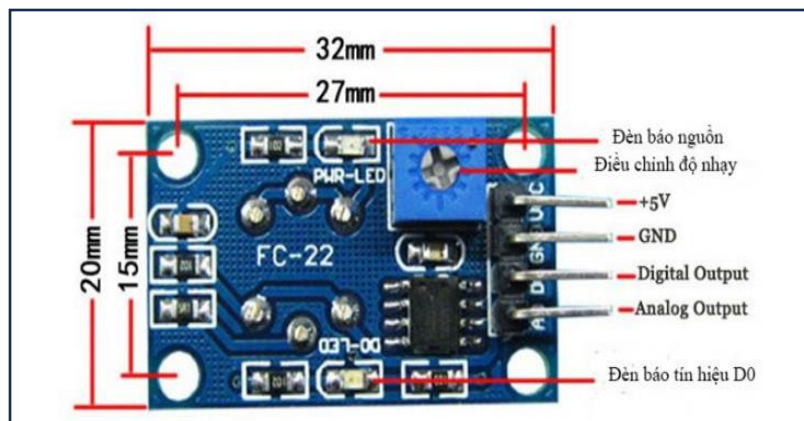
4.2 Thông số kỹ thuật

- Nguồn cung cấp: 2.5V ~ 5V
- Tích hợp MQ-5 gas sensor
- Tín hiệu ra: Analog

Kích thước: 40x21 mm

- LED báo hiệu

4.3 Cấu trúc cảm biến khí hoá lỏng, khí than MQ-5



Hình 13 Cấu trúc cảm biến MQ-5

4.4 Giao tiếp với MQ-5

Khi cảm biến hoạt động nó sẽ truyền tín hiệu từ các chân AOUT và DOUT của mình về vi điều khiển.

- Tín hiệu AOUT:

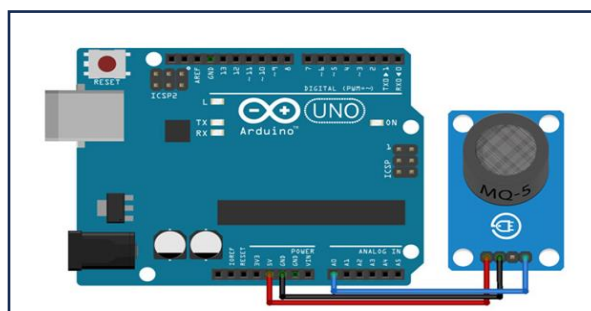
+ Tín hiệu thấp: có khí gas, khí hoá lỏng hoặc khí than.

+ Tín hiệu cao: không có khí gas, khí hoá lỏng hoặc khí than.

- **Tín hiệu DOUT:** cho tín hiệu tương tự

Khi có khí gas 2 đèn LED trên module sẽ phát sáng

4.5 Kết nối với Arduino UNO



Hình 14 Sơ đồ kết nối chân của MQ-5 với Arduino UNO

Các kết nối:

Table 4 Các kết nối chân tương ứng

MQ-5	Arduino
VCC	5V
GND	GND
A0	A0

Chương 5 Buzzer 5VDC

5.1 Buzzer là gì ? (TMB12A05)

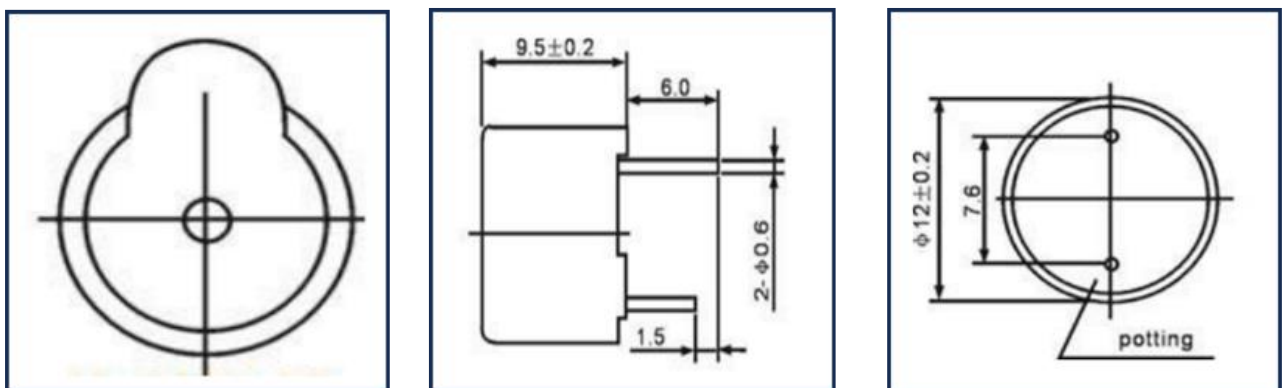
"Còi buzzer 5VDC" là một loại còi buzzer (hoặc buzzer) được thiết kế để hoạt động với nguồn điện có điện áp là 5VDC (Direct Current). Điều này có nghĩa là để còi buzzer này hoạt động, bạn cần cung cấp một nguồn điện với điện áp là 5VDC.

-Thông thường, còi buzzer 5VDC được sử dụng trong các ứng dụng điện tử, như các dự án Arduino, Raspberry Pi, hoặc các mạch điều khiển khác mà điện áp là 5V. Điều này giúp đơn giản hóa quá trình tích hợp và điều khiển, vì nó có thể được nối trực tiếp vào các nguồn cung điện thông dụng trong các dự án điện tử.

5.2 Thông số buzzer được lắp đặt. (TMB12A05)

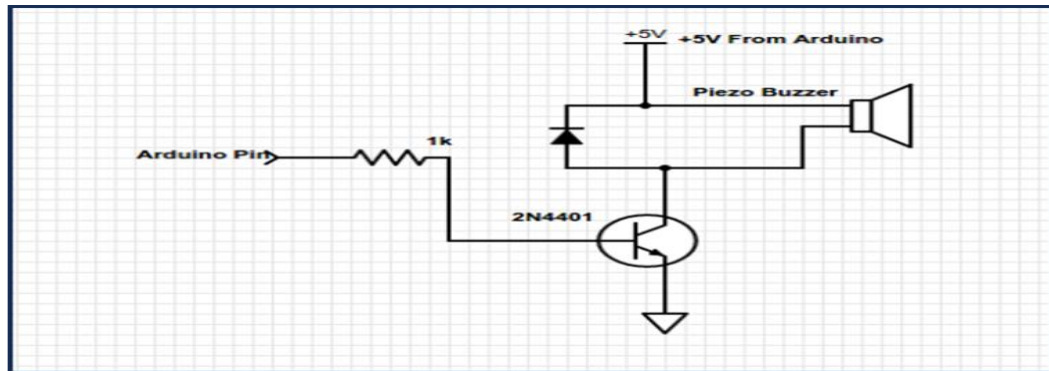
- Nguồn : 4V - 7V
- Điện thế hoạt động tốt : 5V
- Dòng điện tiêu thụ: <25mA
- Tần số cộng hưởng: 2300Hz \pm 500Hz
- Biên độ âm thanh: >85 dB
- Nhiệt độ hoạt động:-20 °C đến +60 °C
- Kích thước : Đường kính 12mm, cao 9,7mm
- Độ dài chân : 6mm

5.3 Cấu trúc buzzer (TMB12A05)



Hình 15 Cấu trúc buzzer

5.4 Giao tiếp buzzer



Hình 16 Sơ đồ nguyên lí buzzer

Xác định Loại Buzzer:

Nếu buzzer là loại thông thường, bạn có thể chỉ cần kết nối một chân với chân GND của Arduino và chân còn lại với một chân số nào đó trên mạch.

Nếu buzzer có tính năng đặc biệt hoặc yêu cầu điều khiển tần số, bạn cần kiểm tra tài liệu kỹ thuật để xác định cách kết nối chính xác.

Lập Trình Arduino:

Sử dụng môi trường lập trình Arduino IDE để viết chương trình.

Xác định chân số mà bạn đã kết nối buzzer và sử dụng hàm pinMode để đặt chân đó làm OUTPUT.

Sử dụng hàm digitalWrite để đặt giá trị HIGH hoặc LOW cho chân, tùy thuộc vào việc bạn muốn kích thích buzzer.

Kiểm Tra Hiệu Suất:

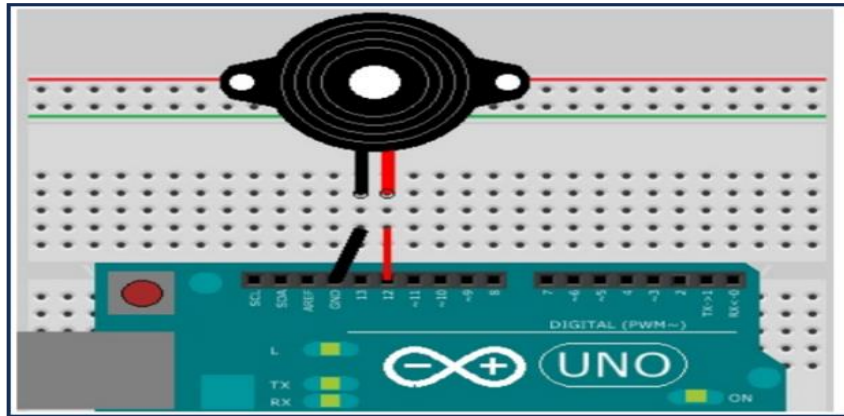
Nạp chương trình lên mạch Arduino.

Khi chương trình chạy, bạn sẽ nghe thấy âm thanh từ buzzer theo cách bạn đã lập trình.

5.5 Kết nối buzzer với Arduino Uno

-Kết nối một chân của buzzer với chân GND của Arduino.

-Kết nối chân còn lại của buzzer với một chân số trên mạch Arduino.



Hình 17 Kết nối chân

Chương 6 ESP8266 NodeMCU Lua V3

6.1 ESP8266 là gì?

-Kit RF thu phát Wifi ESP8266 NodeMCU Lua V3 CH340 là phiên bản NodeMCU sử dụng IC nạp CH340 từ Lolin với bộ xử lý trung tâm là module Wifi SoC ESP8266, kit có 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 V3 CH340 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.

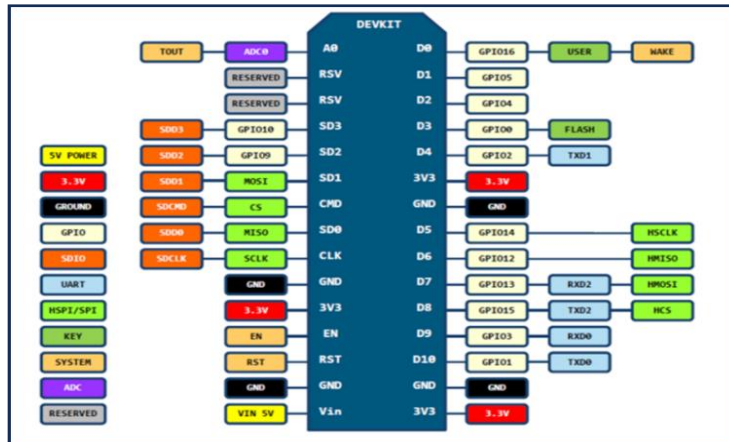
6.2 Thông số kĩ thuật

- Điện áp: 3.3V.
- Wi-Fi Direct (P2P), chế độ soft-AP.
- Điện tiêu thụ hiện tại: 10uA~170mA.
- Bộ nhớ flash có thể kết nối: Tối đa 16MB (bình thường là 512K).
- Bộ giao thức TCP/IP tích hợp.
- Bộ xử lý: Tensilica L106 32-bit.
- Tốc độ xử lý: 80~160MHz.
- RAM: 32K + 80K.
- GPIOs: 17 (đa nhiệm với các chức năng khác).
- Chuyển đổi Analog sang Digital: 1 đầu vào với độ phân giải 1024 bước.
- Công suất đầu ra +19.5dBm ở chế độ 802.11b.

-Hỗ trợ 802.11: b/g/n.

-Số kết nối TCP tối đa đồng thời: 5.

6.3 Cấu trúc



Hình 18 Cấu trúc ESP8266

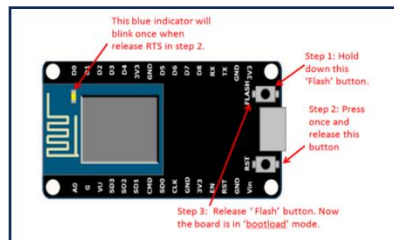
6.4 Giao tiếp

- Cài đặt hỗ trợ ESP8266

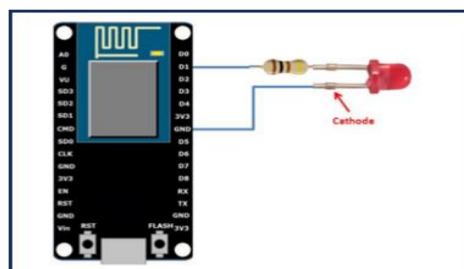
- Thử nghiệm trên Blink

Bước 1: Giữ nút FLASH

Bước 2: Nhấn thả nút RST, sau khi thả thì đèn sẽ nháy 1 lần.



Bước 3: Thả nút FLASH



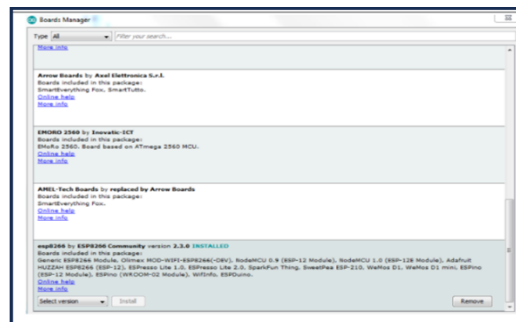
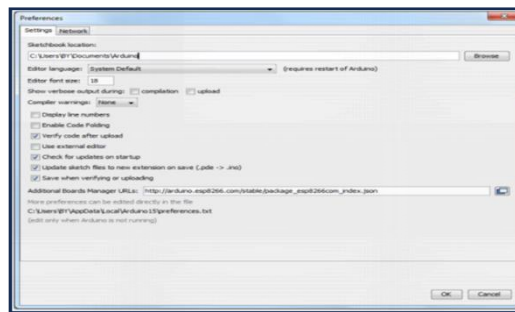
Khi bo mạch ESP đang ở chế độ bootloader, tải lên đoạn mã thông qua môi trường phát triển tích hợp (IDE)

6.5 Kết nối

- Kết nối thông qua wifi

6.6 Cài đặt thư viện cho ESP8266

B1: Nhập http://arduino.esp8266.com/stable/package_esp8266com_index.json vào trường Additional Board Manager URLs trong cài đặt Arduino phiên bản 1.6.4 trở lên



Hình 19 Hướng dẫn cài đặt thư viện cho ESP8266

B2: Sau đó, sử dụng trình quản lý Board để cài đặt gói ESP8266

B3: Cuộn xuống đến 'esp8266 by ESP8266 Community' và nhấp vào nút 'Install' để cài đặt gói thư viện ESP8266.

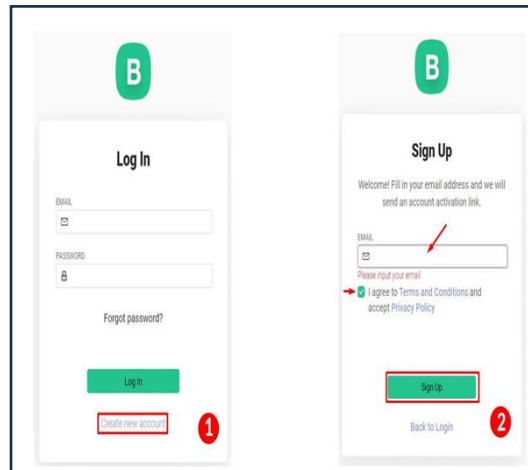
Chương 7 Giới thiệu ứng dụng Blynk

7.1 Blynk là gì ?

- Blynk cung cấp một cách thuận tiện để tạo ứng dụng di động để giám sát và điều khiển các thiết bị IoT. Người dùng có thể sử dụng giao diện đồ họa trực quan để tạo các bảng điều khiển tương tác cho các dự án IoT của họ mà không cần kiến thức lập trình sâu. Nền tảng này thường được sử dụng để kết nối các thiết bị như Arduino, Raspberry Pi và ESP8266 với điện thoại di động, giúp tạo ra các ứng dụng IoT cá nhân hoặc dự án tự làm.

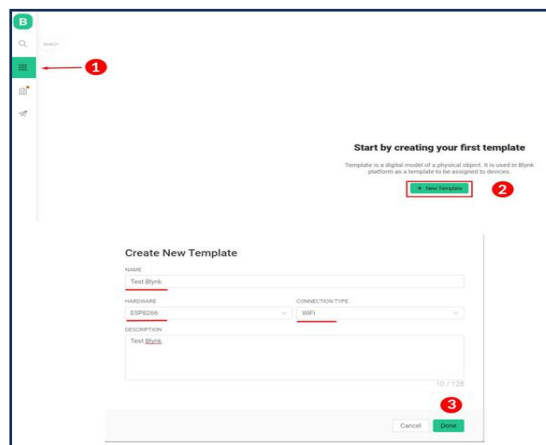
7.2 Giao tiếp với Blynk

- Đầu tiên truy cập vào <https://blynk.cloud/> đăng nhập, nếu chưa có tài khoản thì chọn **Create new account** để đăng ký. Nhập tên email vào, sau đó tích chọn **Sign Up**.



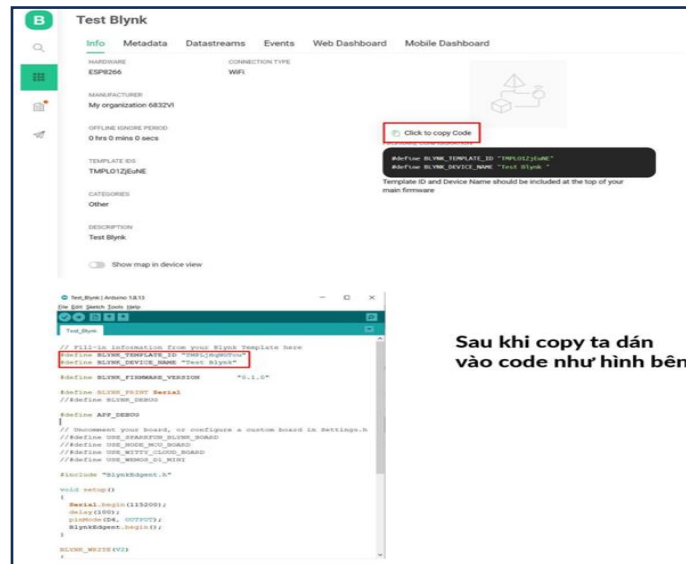
Hình 20 Đăng nhập/Đăng kí

- Họ sẽ gửi mail về cho bạn, sau đó chọn **Create Password** để tạo mật khẩu. Sau khi có tài khoản, bạn đăng nhập vào chọn New Template, nhập tên và chọn đầy đủ như hình dưới:



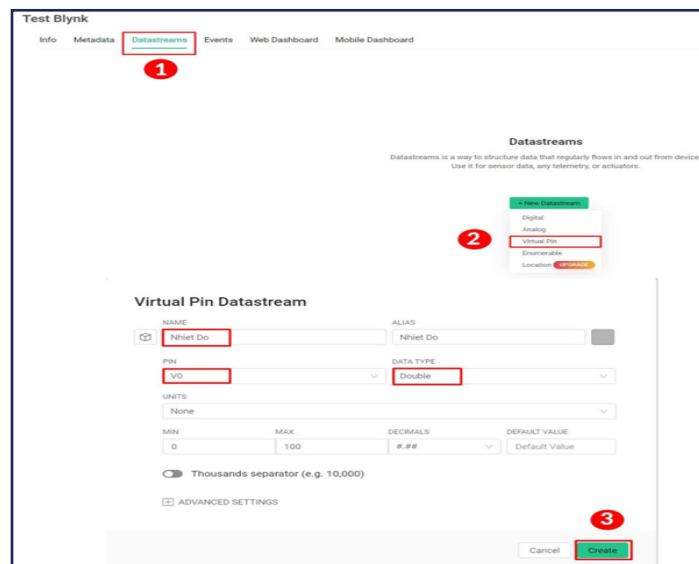
Hình 21 Tạo mật khẩu

- Sau khi tạo xong sẽ hiện giao diện bên dưới, ta copy mã Template để dán vào code, link tải code mình sẽ để ở phần dưới.



Hình 22 Copy Template ID và Device Name

-Tiếp theo, chọn **Datastreams** -> **Virtual Pin** -> nhập đầy đủ datastream của Pin -> **Create**



Hình 23 Nhập datastream

-Ở ví dụ này mình làm gửi dữ liệu cảm biến nhiệt độ, độ ẩm lên Blynk nên các bạn tạo tương tự một Pin V1 là độ ẩm, và tạo V3 là Pin điều khiển Led:

Virtual Pin Datastream

NAME: Led ALIAS: Led

PIN: V2 DATA TYPE: Integer

UNITS: None

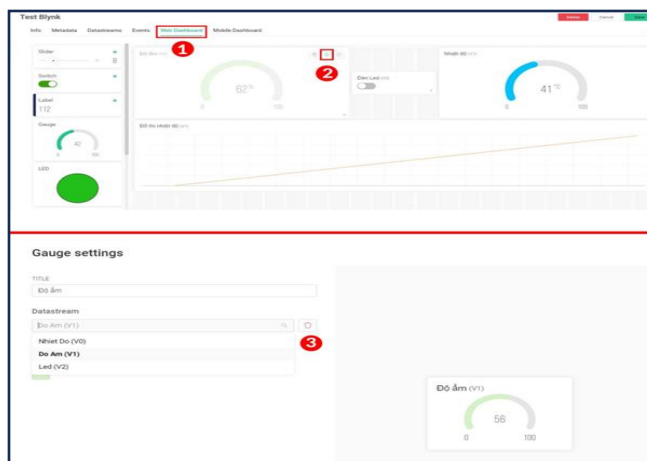
MIN: 0 MAX: 1 DEFAULT VALUE: Default Value

☐ Thousands separator (e.g. 10,000)

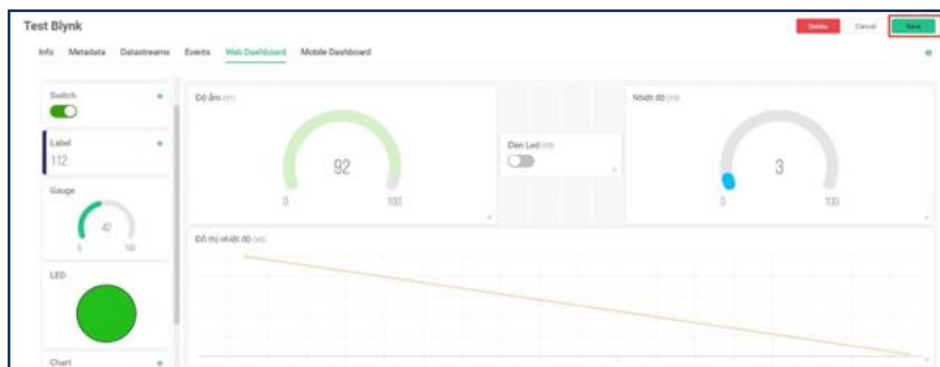
[ADVANCED SETTINGS](#)

Cancel Create

-Sau khi tạo Pin xong, ta chọn **Web Dashboard**, kéo các **Gauge** bên trái qua để hiển thị nhiệt độ và độ ẩm, **Switch** để làm công tắc bật tắt Led, một biểu đồ chart để hiển thị nhiệt độ (bản free chỉ hiển thị được 1 biểu đồ), nhấn biểu tượng cài đặt để chọn từng Pin hiển thị phù hợp.

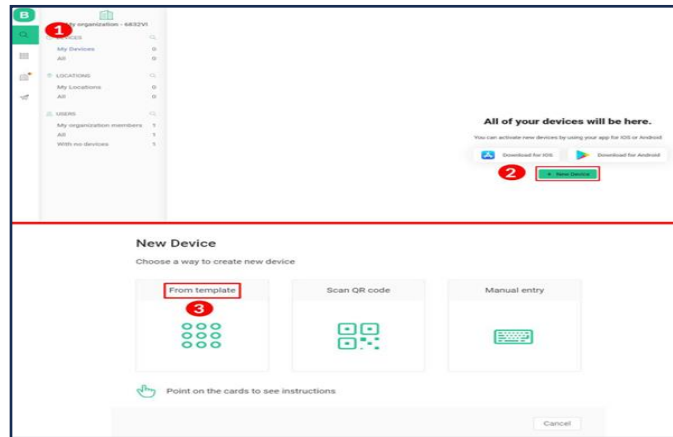


-Sau khi chọn xong ta ấn Save để lưu cài đặt:



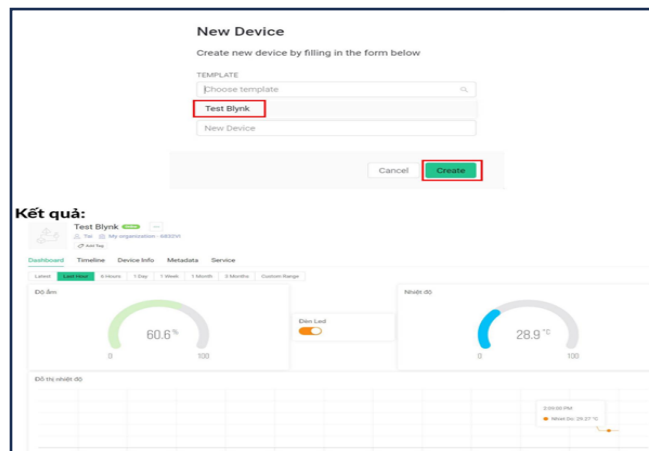
Hình 24 Lưu cài đặt

-Tiếp theo, chọn biểu tượng **Search** -> **New Device** để chọn thiết bị từ From template:



Hình 25 Thêm thiết bị

-Chọn tên template mà bạn đã tạo -> **Create**, sau đó xem kết quả:



7.3 Cài đặt thư viện cho Blynk

Để cài đặt thư viện Blynk cho Arduino, bạn có thể tuân theo các bước sau:

1. Mở Arduino IDE

- Mở Arduino IDE trên máy tính của bạn.

2. Mở Thư Viện Quản Lý

- Điều này có thể được thực hiện bằng cách chọn "Sketch" trong thanh menu, sau đó chọn "Include Library" > "Manage Libraries..."

3. Tìm Kiếm Thư Viện Blynk

- Trong ô tìm kiếm của cửa sổ Quản lý thư viện, nhập "Blynk".

- Bạn sẽ thấy thư viện "Blynk by Blynk Community".

4. Cài Đặt Thư Viện

- Chọn thư viện "Blynk by Blynk Community".
- Bạn sẽ thấy nút "Install". Nhấn vào nút này để cài đặt thư viện.

5. Kiểm Tra Cài Đặt

- Sau khi quá trình cài đặt hoàn tất, bạn có thể kiểm tra xem thư viện đã được cài đặt thành công hay chưa bằng cách kiểm tra trong "Sketch" > "Include Library". Bạn sẽ thấy "Blynk" hiển thị ở đó.

Chương 8 Sơ đồ giải thuật

8.1. Chuẩn bị

- Board arduino uno
- ESP8266-CH340
- Module cảm biến nhiệt DHT11
- Module cảm biến khí gas MQ-5
- Module quang trở LM393
- Led 5mm
- Buzzer 5V

8.2 Code chương trình

-Code cho ESP8266:

```
#define BLYNK_TEMPLATE_ID "TMPL60fofB7Bz"
#define BLYNK_TEMPLATE_NAME "BÁO CHÁY"
#define BLYNK_AUTH_TOKEN "FCOGzticfy1kwQs2hIbkEq5XCTiXRMHq"
#include <DHT.h> //Khai báo thư viện DHT để làm việc với cảm biến nhiệt độ và độ ẩm DHT
#include <BlynkSimpleEsp8266.h> //Khai báo thư viện Blynk để kết nối với nền tảng IoT Blynk sử dụng ESP8266.
#define DHTTYPE DHT11

const int DHTPin = D1;
const int analogMQ5 = A0;
float analogValue;
DHT dht(DHTPin, DHTTYPE);
static char temp[7];
```

```
char auth[] = "FCOGzticfy1kwQs2hIbkEq5XCTiXRMHq"; // Thay thế bằng Auth
Token bạn nhận được
char ssid[] = "THEBEONE"; // Thay thế bằng tên mạng Wi-Fi của bạn
char pass[] = "05072001"; // Thay thế bằng mật khẩu Wi-Fi của bạn
void setup() {
    Serial.begin(9600);
    delay(10);
    dht.begin();
    Blynk.begin(auth, ssid, pass);
}

void loop() {
    analogValue = analogRead(analogMQ5); //Đọc giá trị analog từ cảm biến
gas MQ5 và in ra nếu phát hiện gas vượt quá một ngưỡng
    if (analogValue >= 400) { //Nếu nồng độ khí gas quá 400ppm
        thì in ra màn hình "Phát hiện khí gas" và ngược lại
        Serial.println("Phát hiện gas!");
    }
    else Serial.println("Không phát hiện gas");
    Serial.println(analogValue);
    //cảm biến nhiệt DHT11
    float t = dht.readTemperature(); //Khai báo biến
    delay(2000);
    Serial.print("nhiệt độ: "); //đọc nhiệt độ từ cảm biến DHT, in ra màn
hình
    Serial.println(t);
    Blynk.virtualWrite(V6, t);Gửi giá trị nhiệt độ và giá trị nồng độ khí
đến các cảm biến ảo trên Blynk
    Blynk.virtualWrite(V5, analogValue);
    Blynk.run(); // Duy trì kết nối và xử lý sự kiện Blynk
}
```

-Code cho Arduino UNO

```
#include <DHT.h> //Khai báo thư viện
const int quangtroPin = A0;
const int ledPin = 3;
const int analogMQ5 = A1;
float analogValue;
const int DHTPin = 4;
const int DHTTYPE = DHT11;
DHT dht(DHTPin, DHTTYPE);
const int buzzer = 11;

void setup() {
```

```

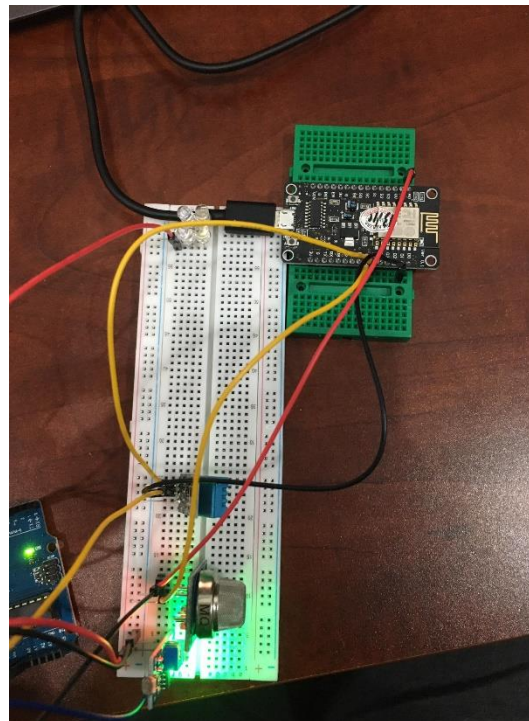
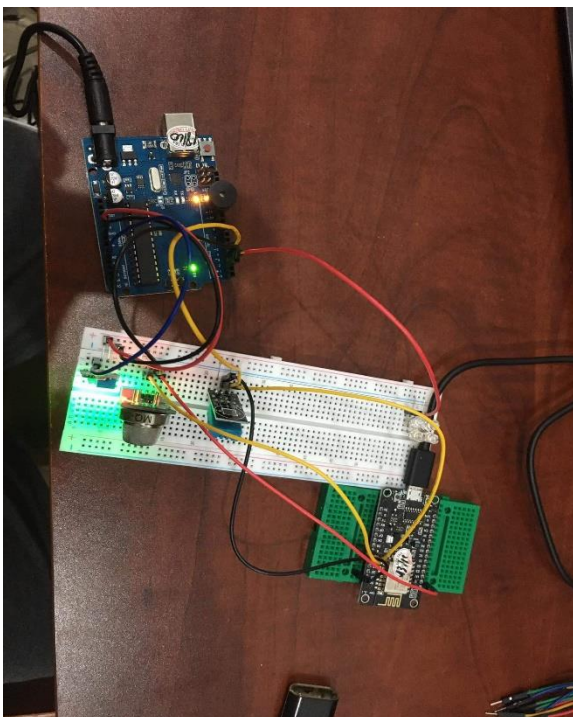
pinMode(ledPin, OUTPUT); //Đèn led
Serial.begin(9600);
pinMode(analogMQ5, INPUT); //cảm biến khí gas/khói
pinMode(buzzer, OUTPUT); //Còi buzzer
dht.begin();
}

void loop() {
// quang trở
int quangtroValue = analogRead(quangtroPin);

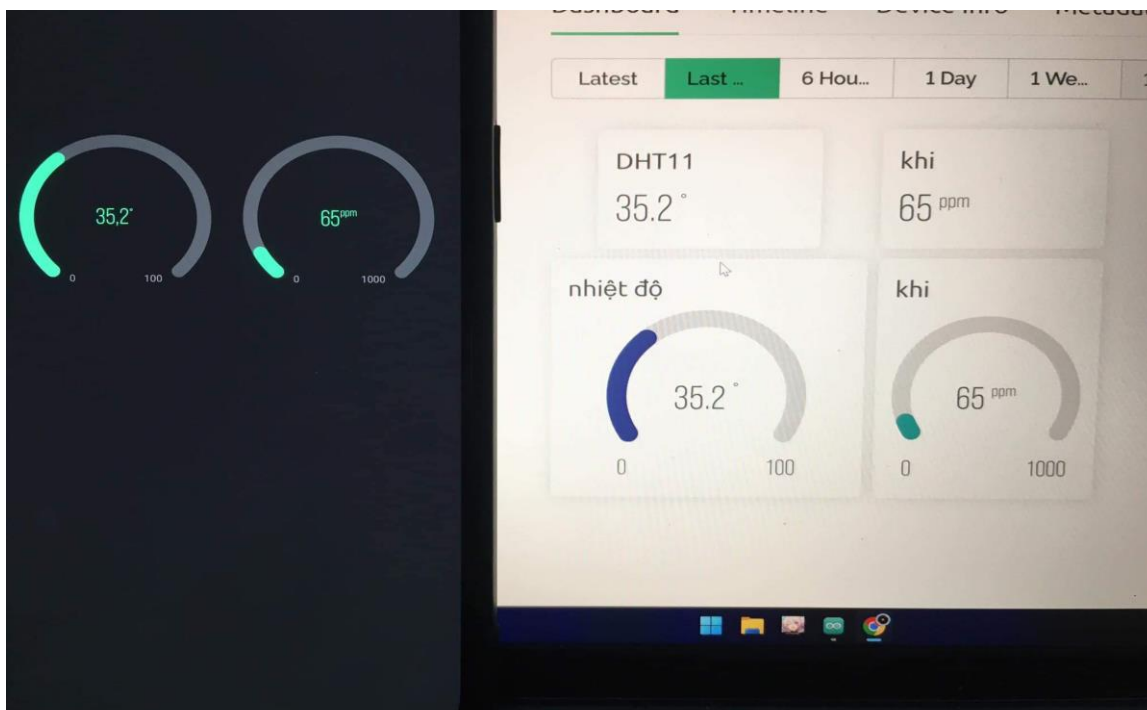
if (quangtroValue < 100) digitalWrite(ledPin, LOW); //Giá trị Analog của
quang trở > 100 thì đèn sáng và ngược lại
else digitalWrite(ledPin, HIGH);
// cảm biến khí gas MQ-5
analogValue = analogRead(analogMQ5);
// Nếu nồng độ gas >= 400ppm thì in ra màn hình "Phát hiện khí gas" + Còi
buzzer sẽ phát ra tiếng và ngược lại.
if (analogValue >= 400) {
    Serial.println("Phát hiện gas!");
    tone(buzzer, 2000);
    delay(1000);
    noTone(buzzer);
    delay(500);
}
else Serial.println("Không phát hiện gas");
//cảm biến nhiệt DHT11
float doC = dht.readTemperature(); // Đọc nhiệt độ từ cảm biến DHT
Serial.print("nhiệt độ: ");
Serial.println(doC);
if (doC >= 40.0){ //Nếu nhiệt độ trên 40 độ C thì còi sẽ phát ra tiếng
    tone(buzzer, 2000);
    delay(1000);
    noTone(buzzer);
    delay(500);
}
delay(500);
}
}

```

8.3 Triển khai sản phẩm

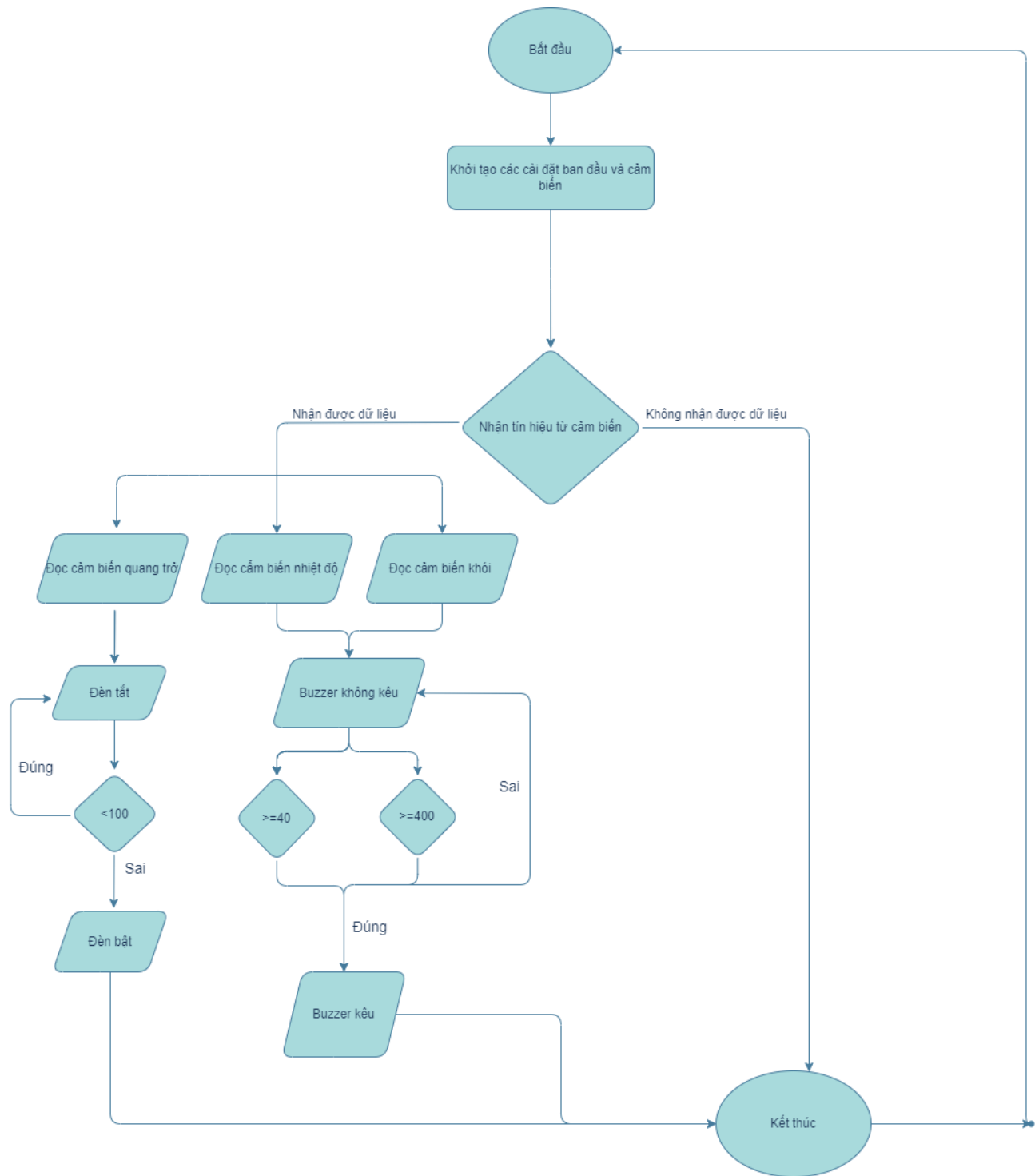


Hình 26 Kiểm tra sản phẩm

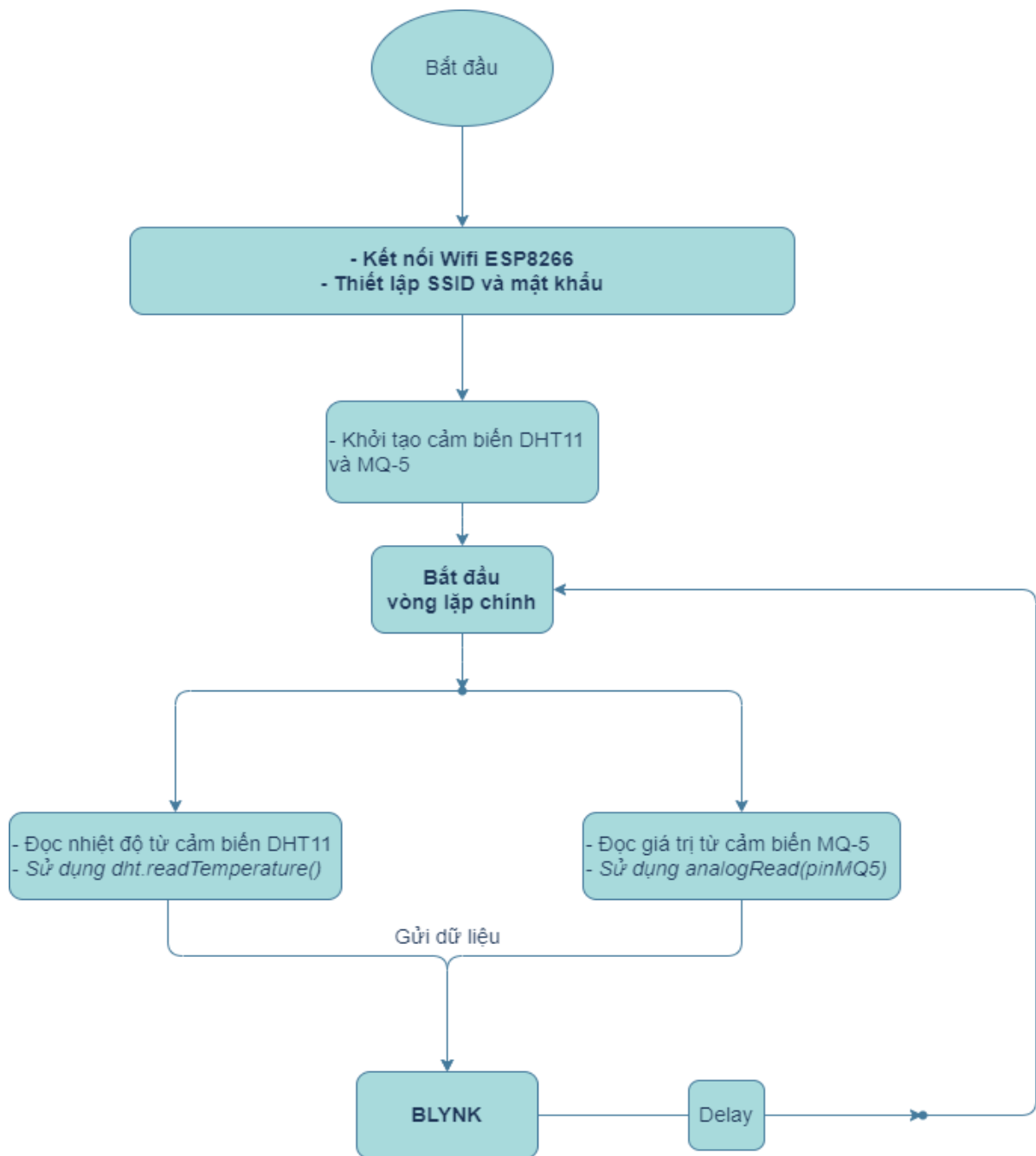


Hình 27 Liên kết lên app Blynk

8.4 Sơ đồ giải thuật



Hình 28 Sơ đồ giải thuật Arduino



Hình 29 Sơ đồ giải thuật ESP8266

BÁO CÁO KỸ NĂNG

1. Kỹ năng làm việc nhóm và quản lý dự án

- Bảng phân công công việc, mức độ đóng góp, hoàn thành
- Dùng biểu đồ Gantt, lập bảng quản lý dự án (Excel) theo mẫu dưới
- Bảng đánh giá điểm từng thành viên (dựa theo mức độ tham gia, đóng góp, hoàn thành): Kỹ năng làm việc nhóm, giao tiếp, hành xử chuyên nghiệp, khả năng lãnh đạo và làm việc độc lập
- Lập bảng kỹ năng ra quyết định (nếu có)

Thứ tự	Nhiệm vụ	Giao cho	Ưu tiên thứ	Tình trạng	Bắt đầu	Kết thúc
1	Kịch bản	Thịnh	2	Hoàn thành	10/11	11/11
2	Sản phẩm	Thắng (chính)	1	Hoàn thành	7/11	14/11
3	PPT	Du	2	Hoàn thành	11/11	14/11
4	Báo cáo	Trọng + An	1	Hoàn thành	13/11	15/11
5	Thuyết trình	Thành	3	Đang thực hiện	11/11	16/11
6	Phần cứng	Thắng	1	Hoàn thành	8/11	14/11
7	Phần mềm	Thắng + Thịnh	2	Hoàn thành	8/11	14/11
8	Câu hỏi	Thịnh	4	Hoàn thành	13/11	14/11
9	Duyệt sản phẩm	An	3	Hoàn thành	14/11	15/11
10	Duyệt PPT	An	3	Hoàn thành	14/11	15/11
11	Duyệt Thuyết trình	An	5	Chưa hoàn thành	14/11	
12	Duyệt Kịch bản	An	5	Hoàn thành	11/11	12/11
13	Duyệt báo cáo	An	5	Hoàn thành	15/11	15/11

Tỉ lệ phần trăm hoàn thành

Hoàn thành	84,6%
Đang thực hiện	7,7%
Quá hạn	7,7%

2. Tự đánh giá các kỹ năng/kiến thức

Nhóm tự đánh giá tổng hợp các kỹ năng, kiến thức đạt được qua đồ án môn học

STT	Các kỹ năng	Đánh giá (*)
1	Kỹ năng làm việc nhóm, giao tiếp, hành xử chuyên nghiệp, khả năng lãnh đạo và làm việc độc lập	A
2	Kỹ năng tư duy phản biện	B
3	Kỹ năng thuyết trình	B
4	Giao tiếp kỹ thuật (viết báo cáo kỹ thuật)	B
5	Kỹ năng tư duy sáng tạo	A
6	Kỹ năng quản lý dự án/thời gian thực hiện dự án	B
7	Hình thành nội dung, xác định vấn đề và kỹ năng giải quyết vấn đề	B
8	Kiến thức, thực nghiệm qua đồ án môn học	A

(*)**Ghi chú:** Đánh giá theo mức A/B/C/D (A: Rất tốt, B: Tốt, C: Trung bình, D: Chưa tốt)

BẢNG THỐNG KÊ CẬP NHẬT TÀI LIỆU KỸ THUẬT

Phiên bản	Mô tả	Thành viên	Ngày	Xác nhận (Trưởng nhóm)	Ngày xác nhận
1.0	Tạo tài liệu	L. M. Trọng	13/11/23	N. T. An	15/11/23
2.0	Tạo/Cập nhật chương 1	T. T. Thịnh	13/11/23	N. T. An	15/11/23
3.0	Tạo/Cập nhật chương 2	T. T. Thịnh	13/11/23	N. T. An	15/11/23
4.0	Tạo/Cập nhật chương 3	N. K. Du	14/11/23	N. T. An	15/11/23
5.0	Tạo/Cập nhật chương 4	N. K. Du	14/11/23	N. T. An	15/11/23
6.0	Tạo/Cập nhật chương 5	N. T. An	13/11/23	N. T. An	15/11/23
7.0	Tạo/Cập nhật chương 6	N. T. An	13/11/23	N. T. An	15/11/23
8.0	Tạo/Cập nhật chương 7	P. M. Thắng	14/11/23	N. T. An	15/11/23
9.0	Tạo/Cập nhật chương 8	P. M. Thắng	14/11/23	N. T. An	15/11/23
10.0	Báo cáo kỹ năng	L. M. Trọng	14/11/23	N. T. An	15/11/23
Nhập môn Kỹ thuật – 23DTV					

TÀI LIỆU THAM KHẢO

Nguồn buzzer:

https://www.quick\teck.co.uk/Management/EEUploadFile/1420788438.pdf?fbclid=IwAR1Bb3wC5xNwPM_59NF_8PtG9nKjQcrrSZsgj8lBQdKPVI0BCFXG40bcl4w

Nguồn esp 8266:

Nguồn: <https://handsontec.com/dataspecs/module/esp8266-V13.pdf>

Nguồn ARDUINO :

<https://hshop.vn/products/arduino-uno-r3>

<https://imaker.vn/lm393-cam-bien-anh-sang-quang-tro-4-chan>

Nguồn của quang trở :

<https://imaker.vn/lm393-cam-bien-anh-sang-quang-tro-4-chan>

Nguồn của cảm biến nhiệt độ , độ ẩm:

<https://dientutuonglai.com/cam-bien-nhiet-do-va-do-am-dht11.html>