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

**CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH**

**KHOA KỸ THUẬT ĐIỆN TỬ 2**

**\_\_\_\_\_\_\_\_\_\_\_\_\_**

**BÁO CÁO**

**ĐỒ ÁN TỐT NGHIỆP**

**CHUYÊN NGÀNH: KỸ THUẬT ĐIỆN - ĐIỆN TỬ**

**HỆ ĐẠI HỌC CHÍNH QUY**

**NIÊN KHÓA: 2014-2019**

***Đề tài:***

**MẠCH HẸN GIỜ PHÁT THÔNG TIN**

**Sinh viên thực hiện : HUỲNH ĐỨC LINH**

**MSSV : N14DCDT112**

**Lớp : D14CQKD01-N**

**Giáo viên hướng dẫn : ThS. NGUYỄN LAN ANH**

**11/2018**

**TP.HCM – 2018**

**MỤC LỤC**

[CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc519927493)

[1.1 Tổng Quan Về Lĩnh Vực Nghiên Cứu: 1](#_Toc519927494)

[1.2: Mục Tiêu Đề Tài 2](#_Toc519927495)

[1.3. Nhiệm Vụ Và Giới Hạn Của Đề Tài: 2](#_Toc519927496)

[1.4. Phương pháp nghiên cứu: 2](#_Toc519927497)

[CHƯƠNG II: GIỚI THIỆU CÁC MODULE DS3232, DFplayer mini 3](#_Toc519927499)

[2.1 Giới thiệu tổng quát về vi điều khiển ESP8266 NODEMCU, arduino mega 2560, màn hình 3.2 TFT, DS1307, DFplayer mini 3](#_Toc519927500)

[2.1.1 Giới thiệu tổng quát 3](#_Toc519927501)

[2.1.2 Module DS1307 15](#_Toc519927507)

[2.1.3 Module DFplayer mini 18](#_Toc519927516)

[2.1.4 Màn hình TFT 3.2 và TFT sheild LCD module 19](#_Toc519927518)

[2.1.5 Arduino mega 2560 20](#_Toc519927519)

[2.1.6 Loa 0.5 w 20](#_Toc519927519)

[2.2. Giới thiệu tổng quát về Blynk 21](#_Toc519927520)

[2.2.1 App Blynk là gì? 21](#_Toc519927521)

[2.2.2 Tại sao phải dùng App Blynk 21](#_Toc519927522)

[2.3. Hướng dẫn sử dụng App Blynk 22](#_Toc519927523)

[2.3.1 Tải ứng dụng Blynk trên ANDROID hoặc IOS 22](#_Toc519927524)

[2.3.2 Lấy mã AUTH TOKEN 22](#_Toc519927525)

[2.3.3 Cài đặt thư viện Blynk 23](#_Toc519927526)

[2.3.4 Tạo code ví dụ 24](#_Toc519927527)

[2.3.5 Dán mã xác thực 26](#_Toc519927530)

[CHƯƠNG III: Thiết kế mô hình phần cứng và phần mềm lâp trình ứng dụng 32](#_Toc519927536)

[3.0. Tìm hiểu phần mềm lập trình Arduino IDE: 27](#_Toc519927531)

[3.1 Thiết kế mô hình phần cứng và phần mềm lâp trình cho hệ thống. 29](#_Toc519927532)

[3.1 Sơ đồ kết nối hệ thống mạch hẹn giờ phát nhạc: 29](#_Toc519927533)

[3.2 Sơ đồ giải thuật: 30](#_Toc519927534)

# CHƯƠNG IV: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

[TÀI LIỆU THAM KHẢO 33](#_Toc519927537)

# MỤC LỤC HÌNH

[Hình 2.1: Module ESP8266 version 12 3](#_Toc519927248)

[Hình 2.2: ESP8266 NODEMCU 4](#_Toc519927249)

[Hình 2.3: Sơ đồ ra chân ESP8266 NODEMCU 5](#_Toc519927250)

[Hình 2.4: Tải firmware cho module ESP8266 NODEMCU 6](#_Toc519927251)

[Hình 2.5: Dán đường dẫn file .json vào Arduino IDE 7](#_Toc519927252)

[Hình 2.6: Cài Firmware ESP8266 NODEMCU cho Arduino 8](#_Toc519927253)

[Hình 2.7 Cài đặt gói thư viện cho ESP8266 NODEMCU 9](#_Toc519927254)

[Hình 2.8: Chọn NodeMCU Board trong Adruino IDE 9](file:///C:\Users\Administrator\Downloads\HuynhDucLinh_N14DCDT112.docx#_Toc519927255)

[Hình 2.9: Thiết lập cổng COM port kết nối máy tính với board 10](#_Toc519927256)

[Hình 2.10: Kiểm tra kết nối với chương trình Blink có sẵn 11](#_Toc519927257)

[Hính 2.11: Compile chương trình nạp xuống Kit 12](#_Toc519927258)

[Hình 2..12: Lỗi kết nối với máy tính thường gặp 13](#_Toc519927259)

[Hình 2.13: Update driver để sửa lỗi 14](#_Toc519927260)

[Hình 2.19: App Blynk 21](#_Toc519927266)

[Hình 2.20: Auth token 23](#_Toc519927267)

[Hình 2.21 Giao diện của App Blynk 24](#_Toc519927268)

[Hình 2.22. Thanh công cụ của Blynk 25](#_Toc519927269)

[Hình 2.23 Thiết lập cụ để Blynk 25](#_Toc519927270)

[Hình 2.24: Phần mềm lập trình Arduino IDE 27](#_Toc519927271)

[Hình 2.25.: Sơ đồ kết nối hệ thống với ESP8266 NODEMCU 29](#_Toc519927272)

# 

# 

**Lời nói đầu**

Ngày nay việc ứng dụng vi điều khiển, vi xử lý đang ngày càng phát triển rộng rãi và thâm nhập ngày càng nhiều vào các lĩnh vực kỹ thuật và đời sống xã hội. Tuy nhiên ứng dụng cho các hệ thống nhúng ngày nay không đơn giản chỉ dừng lại ở điều khiển đèn nhấp nháy, đếm số người vào/ra, hiển thị dòng thông báo trên matrix led hay điều khiển ON-OFF của động cơ… mà nó ngày càng trở nên phức tạp. Và với xu hướng tất yếu này cùng với sự phát triển mạnh mẽ của công nghệ chế tạo vi mạch, người ta đã tạo ra những vi điều khiển có cấu trúc mạnh hơn, đáp ứng thời gian thực tốt hơn, chuẩn hóa hơn so với các vi điều khiển 8 bit trước đây.

Với sự phát triển mạnh mẽ của khoa học, đặc biệt là ngành điện tử, sự phát minh ra các linh kiện điện tử đã và đang ngày càng đáp ứng được yêu cầu của các hệ thống. Ưu điểm của việc sử dụng các linh kiện điện tử làm cho các hệ thống linh hoạt và đa dạng hơn, giá thành thấp hơn và độ chính xác cao hơn.

Sau gần 4 năm học tập và nghiên cứu ở trường, em đã được làm quen với các môn học chuyên ngành. Để áp dụng lý thuyết với thực tế em được giao đồ án môn cuối khóa với tên đề tài là “**Mạch hẹn giờ phát thông tin**”

Tuy nhiên do kiến thức chuyên môn còn hạn chế, tài liệu tham khảo có giới hạn nên còn xảy ra nhiều sai sót. Em rất mong mong thầy và các bạn góp ý bổ sung để bản đồ án của em được hoàn thiện hơn và giúp em hiểu biết hơn trong quá trình học tập tiếp theo.

## Em xin chân thành cảm ơn!

Huỳnh Đức Linh

# CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI

## 1.1 Tổng Quan Về Lĩnh Vực Nghiên Cứu:

Trong những năm gần đây, các hệ thống điều khiển theo thời gian thực là một trong những lĩnh vực thu hút nhiều sự chú ý trong giới khoa học nghiên cứu về khoa học máy tính. Trong đó, vấn đề điều hành thời gian thực và vấn đề lập lịch là đặc biệt quan trọng. Một trong ứng dụng quan trọng của hệ thống thời gian thực (RTS) đã và đang được ứng dụng rộng rãi hiện nay là các dây truyền sản xuất tự động, robot, điều khiển các thí nghiệm tự động, trong thiết kế đồng hồ hiển thị thời gian thực…Thế hệ ứng dụng tiếp theo của hệ thống này sẽ điều khiển robot giống con người, hệ thống kiểm soát thông minh trong các nhà máy công nghiệp, điều khiển các trạm không gian.

Và cho ra đời các “**Mạch hẹn giờ phát nhạc** “được phát minh ngày càng nhiều để hỗ trợ cuộc sống của con người. Đã bao giờ bạn cảm thấy khó chịu khi phải tự bật nhạc vào mỗi buổi sáng hay cài chuông báo thức chưa, thật phức tạp và tốn thời gian, và mạch hẹn giờ phát nhạc phục vụ có nhiều ứng dụng mang lại tiện ích và tiết kiệm tiền bạc cho gia đình bạn. Những sản phẩm này giúp bạn có thể chủ động hơn trong những công việc cần lập trình về giờ giấc, ngoài ra còn giúp bạn tiết kiệm điện năng, dẽ dàng cài đặt để thông báo cho các công xưởng vào thời gian chính xác, …

Cụ thể hơn, ứng dụng này sẽ thông báo những sự kiện quan trọng cho người dùng, nhắc nhở làm một công việc nào đó như đến giờ uống thuốc, tập thể dục, sinh hoạt cá nhân, cập nhật tin tức mới từ internet hoặc gặp gỡ khách hàng v.v...

## 1.2: Mục Tiêu Đề Tài

* Tìm hiểu nguyên lý mạch giải mã phát nhac và các mô hình điều khiển bằng wiffi.
* Tìm hiểu các thiết bị linh kiện sử dụng trong hệ thống như: Module wifi ESP8266, vi điều khiển, các module DFplayer, DS1307...
* Tìm hiểu các phần mềm thiết kế và thực hiện hệ thống: Phần mềm tạo APP, phần mềm lập trình vi điều khiển ...
* Thực hiện thiết kế mạch hẹn giờ phát nhạc.

## 1.3. Nhiệm Vụ Và Giới Hạn Của Đề Tài:

Với sự hạn chế về kinh nghiệm, kinh phí và thời gian thực hiện nên em xin giới hạn phạm vi thực hiện đề tài là:

- Tìm hiểu nguyên lý mạch giải mã phát nhạc và các mô hình điều khiển bằng wifi.

- Tìm hiểu các module cảm biến hiện có trên thị trường hiện nay.  
- Tìm hiểu các phần mềm thiết kế và thực hiện hệ thống: Phần mềm tạo APP, phần mềm lập trình vi điều khiển...

- Thực hiện thiết kế mạch hẹn giờ phát nhạc.

## 1.4. Phương pháp nghiên cứu:

Nghiên cứu dựa vào các nguồn tài liệu tìm kiếm được từ các ngiên cứu và các bài báo trong nước và cả nước ngoài. Nghiên cứu của các sinh viên trong và ngoài nước. Xây dựng mạch hẹn giờ phát nhạc từ các thiết bị có sẵn như vi điều khiển, các module cảm biến.

# CHƯƠNG II: GIỚI THIỆU CÁC MODULE DS3232, DFplayer mini

## 2.1 Giới thiệu tổng quát về vi điều khiển ESP8266 NODEMCU:

### 2.1.1 Giới thiệu tổng quát

- [ESP8266](http://hshop.vn/collections/nguyen-lieu-iot) là dạng Vi điều khiển tích hợp Wifi (Wifi SoC) được phát triển bởi [Espressif Systems](http://espressif.com/), một nhà sản xuất Trung Quốc có trụ sở tại Thượng Hải. Với Vi điều khiển và Wifi tích hợp, ESP8266 cho phép lập trình viên có thể thực hiện vô số các tác vụ TCP/IP đơn giản để thực hiện vô số các ứng dụng khác nhau, đặc biệt là các ứng dụng IoT

- Module [ESP8266](http://hshop.vn/collections/nguyen-lieu-iot) có giá thành rẻ, phải nói là rẻ nhất trong tất cả các loại Wifi SoC từ trước tới nay (trước ESP8266 có series CC3xxx từ Ti rất mắc nên không phổ biến), chỉ khoảng 2USD cho phiên bản đầu tiên, điều này đã thu hút các IoT-er khám phá cũng như dịch các tài liệu của ESP8266 sang tiếng Anh và phát triển vô số các ứng dụng kèm theo. Sau nhiều năm phát triển, hiện nay đã có hơn [14 phiên bản ESP](http://www.esp8266.com/wiki/doku.php?id=esp8266-module-family) ra đời, trong đó phổ biến nhất là ESP-12.



Hình 2.1: Module ESP8266 version 12

Module ESP-12 kết hợp với firmware ESP8266 trên Arduino và thiết kế phần cứng giao tiếp tiêu chuẩn đã tạo nên [NodeMCU, loại Kit phát triển ESP8266](http://hshop.vn/products/kit-rf-thu-phat-wifi-esp8266-nodemcu) phổ biến nhất trong thời điểm hiện tại. Với cách sử dụng, kết nối dễ dàng, có thể lập trình, nạp chương trình trực tiếp trên phần mềm Arduino, đồng thời tương tích với các bộ thư viện Arduino sẵn có, [NodeMCU](http://hshop.vn/collections/nguyen-lieu-iot) là sự lựa chọn hàng đầu cho các bạn muốn tìm hiểu về ESP8266 hiện nay.



Hình 2.2: ESP8266 NODEMCU

### 2.1.2 Giới thiệu sơ bộ về ESP8266 NODEMCU

### ****Khả năng hoạt động như một modem wifi:****

- Có thể quét và kết nối đến một mạng wifi bất kỳ (Wifi Client) để thực hiện các tác vụ như lưu trữ, truy cập dữ liệu từ server.

- Tạo điểm truy cập wifi (Wifi **A**ccess **P**oint) cho phép các thiết bị khác kết nối, giao tiếp và điều khiển.

- Là một server để xử lý dữ liệu từ các thiết bị sử dụng internet khác.

**Nguồn vào và nguồn ra**

[ESP8266 NodeMCU](http://hshop.vn/products/kit-rf-thu-phat-wifi-esp8266-nodemcu) nhận nguồn từ cổng micro USB tích hợp sẵn trên mạch, giúp việc nạp code trở nên dễ dàng hơn. Bên cạnh đó, việc cấp nguồn cho module cũng linh động hơn vì bạn có thể sử dụng sạc dự phòng thay cho nguồn từ USB trên máy tính (nguồn cấp tối đa là 5V).

[ESP8266 NodeMCU](http://hshop.vn/products/kit-rf-thu-phat-wifi-esp8266-nodemcu) có thể cung cấp nguồn cho tối đa 4 thiết bị: 3 nguồn ra 3.3V và một nguồn từ chân Vin (điện thế bằng điện thế từ cổng micro USB). Khi sử dụng các chân cấp nguồn này, hãy luôn kiểm tra để chắc chắn không cắm nhầm chân dương (trên mạch in là 3v3 và Vin) và chân âm (GND). Tuy nhiên, 3 chân 3.3V đều được bảo vệ, khi cắm ngược cực, module sẽ chỉ nóng lên và dừng hoạt động. Chân Vin thì KHÔNG, cắm ngược cực ở chân này là module bốc khói theo nghĩa đen nhé (kinh nghiệm xương máu).

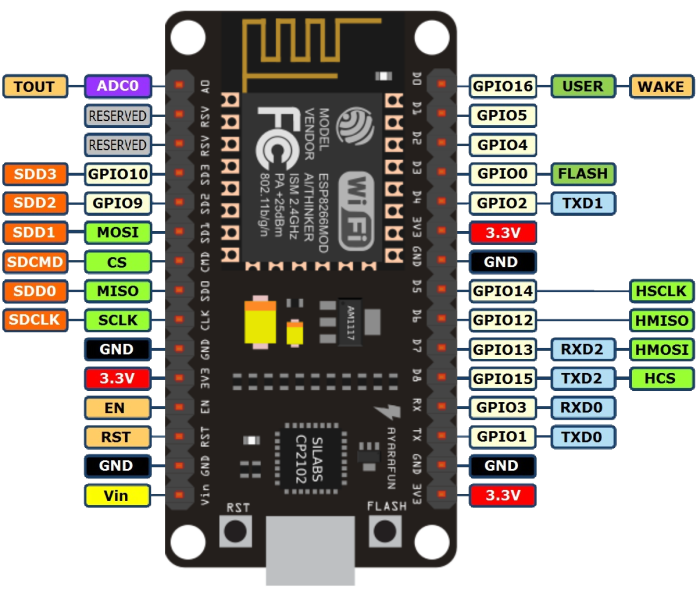
**Truyền và nhận tín hiệu**

[ESP8266 NodeMCU](http://hshop.vn/products/kit-rf-thu-phat-wifi-esp8266-nodemcu) có tổng cộng 13 chân GPIO (General-purpose input/output) – chân có thể truyền/nhận tín hiệu (trên mạch in từ D0 đến D8 và RX, TX, SD2, SD3).

### Bắt đầu sử dụng

Nắm được sơ lược lý thuyết rồi, chúng ta hãy bắt đầu thực hành luôn nhé. Phần này sẽ hướng dẫn bạn cài firmware cho [ESP8266 NodeMCU](http://hshop.vn/products/kit-rf-thu-phat-wifi-esp8266-nodemcu) và cài đặt chương trình đầu tiên.

#### ****Bước 1: Bản đồ chân Pin trên ESP8266 NodeMCU****

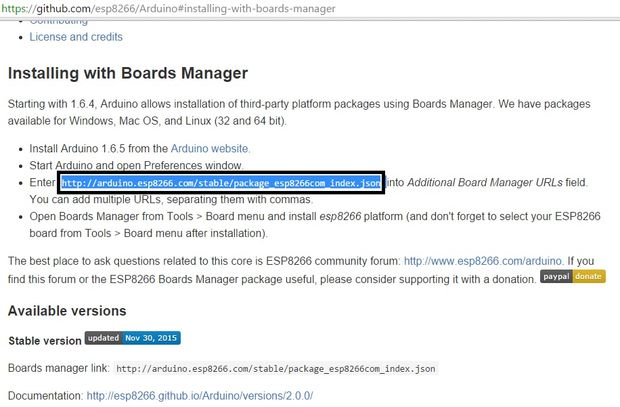


Hình 2.3: Sơ đồ ra chân ESP8266 NODEMCU

**LƯU Ý:**

* Module chỉ có thế kết nối tới nguồn tối đa 5V qua cổng micro USB.
* Các chân I/O chỉ có thể giao tiếp với các linh kiện qua điện thế tối đa là 3.3V

**Bước 2: Tải firmware cho module**



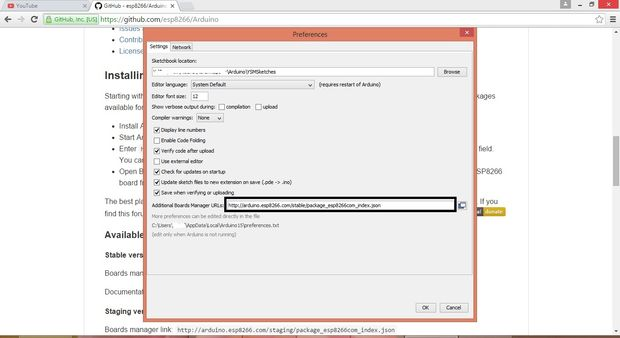
Hình 2.4: Tải firmware cho module ESP8266 NODEMCU

Như nội dung trong hình, bạn hãy **copy đường dẫn** của firmware mới nhất từ [trang Github này](https://github.com/esp8266/Arduino#installing-with-boards-manager).

Đường dẫn nhìn như thế này:

<http://arduino.esp8266.com/stable/package_esp8266com_index.json>

**Bước 3: Dán đường dẫn của file .json đó vào Adruino IDE**



Hình 2.5: Dán đường dẫn file .json vào Arduino IDE

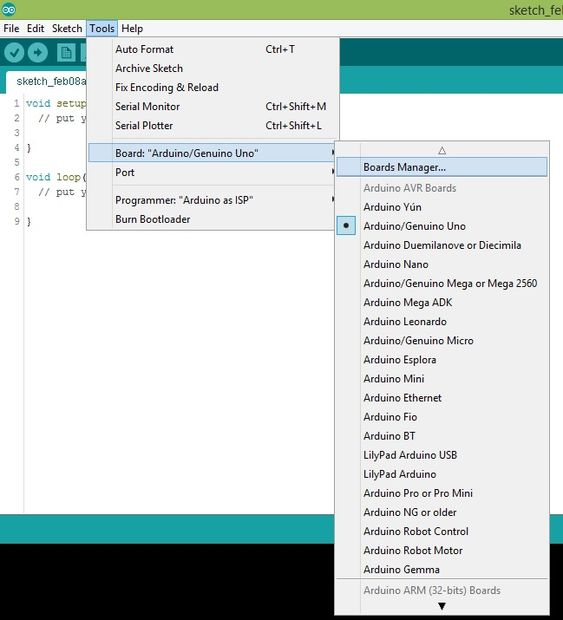
Vào File -> Preference -> dán đường dẫn vào Additional Boards Manager URLs -> OK -> khởi động lại IDE.

Trong trường hợp máy bạn chưa cài Adruino IDE (môi trường lập trình cho các mạch Adruino). Hãy tải ở link này:

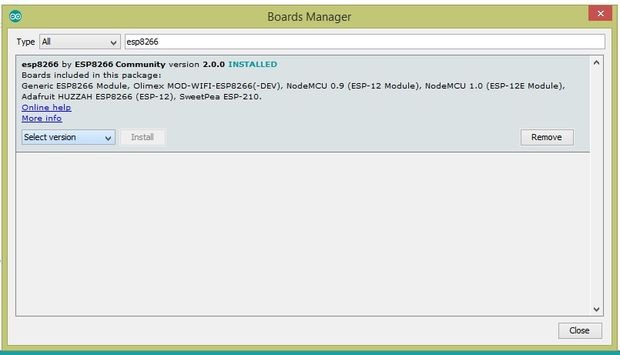
* Tải trực tiếp từ Windows Store: [https://www.microsoft.com/store/apps/9nblggh4rsd8](https://www.arduino.cc/download_handler.php?f=https://www.microsoft.com/store/apps/9nblggh4rsd8?ocid=badge)
* Tải file cài đặt: <https://www.arduino.cc/download_handler.php>

Hoặc vào [trang download của Adruino](https://www.arduino.cc/en/Main/Software) và tải phiên bản bạn mong muốn.

**Bước 4: Cài đặt Firmware ESP8266 cho Arduino**



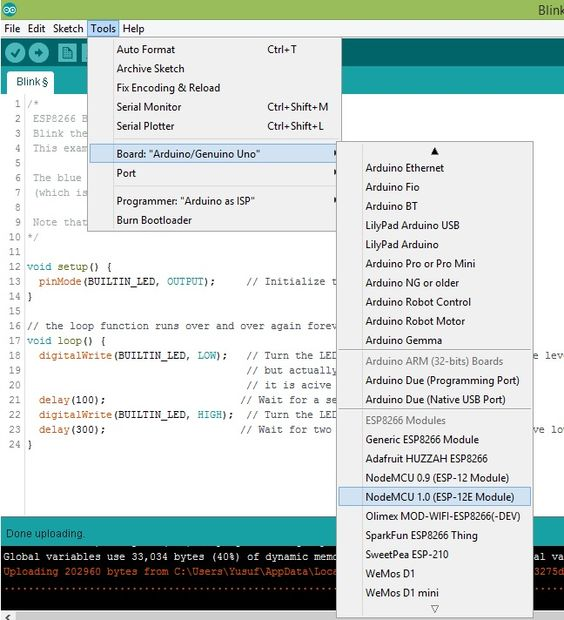
**Hình 2.6: Cài Firmware** ESP8266 NODEMCU **cho Arduino**



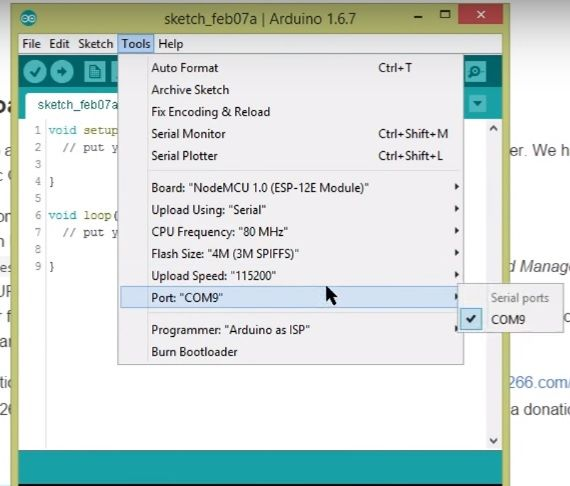
Hình 2.7 Cài đặt gói thư viện cho ESP8266 NODEMCU

Vào Tools -> Boards Manager -> tìm tên “esp8266” -> Install -> Khởi động lại IDE

**Bước 5: Chọn NodeMCU Board trong Adruino IDE**



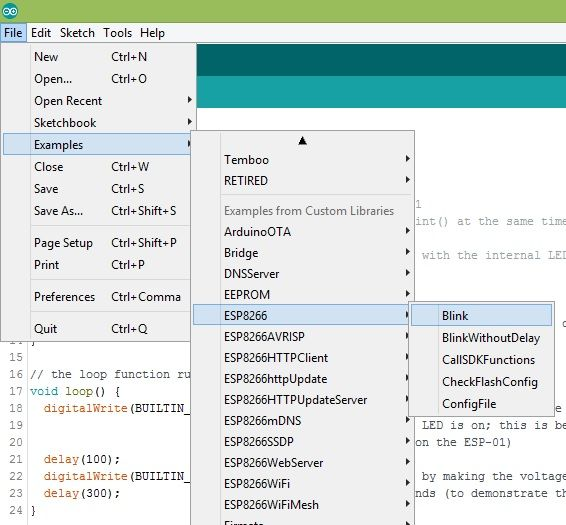
Hình 2.8: Chọn NodeMCU Board trong Adruino IDE



Hình 2.9: Thiết lập cổng COM port kết nối máy tính với board

Vào Tools -> Board -> kéo tìm và chọn ***NodeMCU 1.0 ( ESP-12EModule)***và ***thiết đặt COM Port*** kết nối máy tính của Board, vậy là bạn đã sẵn sàng để chạy chương trình đầu tiên.

**Bước 6: LED Blink – kết nối đến đèn Led có sẵn trên module**



Hình 2.10: Kiểm tra kết nối với chương trình Blink có sẵn

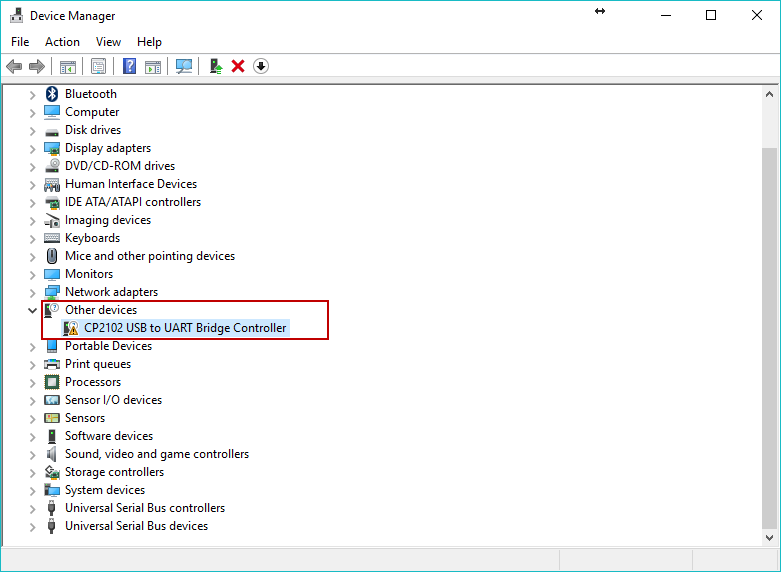


Hính 2.11: Compile chương trình nạp xuống Kit

Phần code sẽ như thế này, sau đó chọn nút Upload, chờ IDE nạp code vào module và xem thành quả.

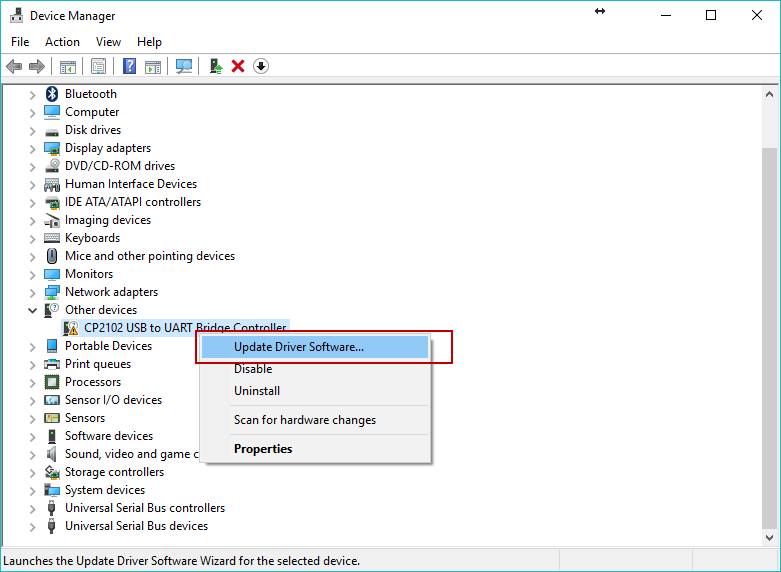
### Máy tính của bạn không kết nối được với ESP8266 NodeMCU?

Trong trường hợp đã cắm module vào máy tính nhưng không thấy tín hiệu gì, hãy kiểm tra lại driver đã nhận hay chưa bằng cách vào Device Manager



Hình 2..12: Lỗi kết nối với máy tính thường gặp

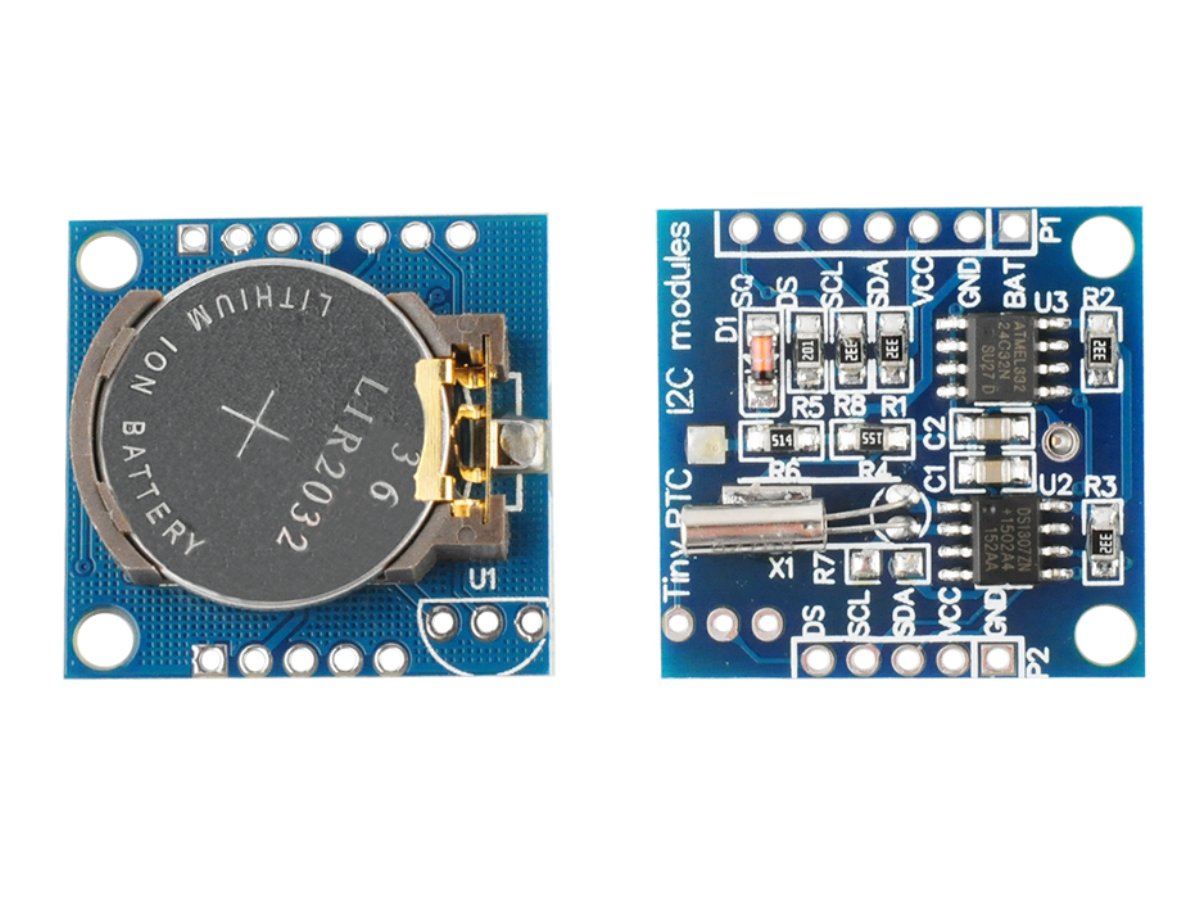
Nếu bạn thấy có dấu chấm than như hình nghĩa là chưa có driver. Để khắc phục, hãy tải driver rồi giải nén. Sau đó click phải chuột chọn **Update Driver Software**



Hình 2.13: Update driver để sửa lỗi

Rồi chọn **Browse my computer for driver software** và tìm đến thư mục chứa driver để cài đặt. Sau khi cài driver xong, kiểm tra lại trong Device Manager thấy không còn dấu chấm than là OK.

## 2.1.2 Module DS1307

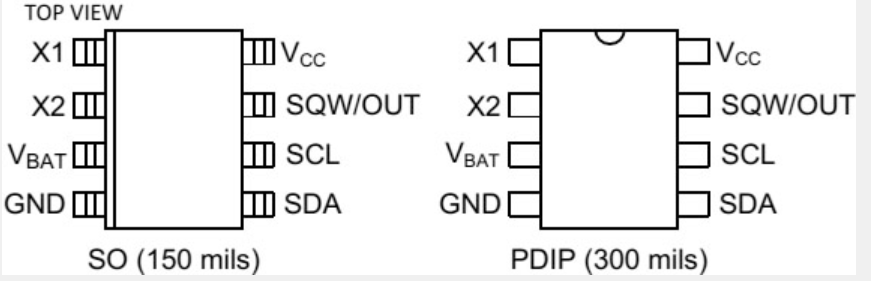


Hình 2.14: Module RTC DS1307

### 2.1.2.1 Tổng quát Module RTC DS3232

- DS1307 là chip thời gian thực hay RTC (Read time clock). Đây là một IC tích hợp cho thời gian bởi vì tính chính xác về thời gian tuyệt đối cho thời gian: Thứ, ngày, tháng, năm, giờ, phút, giây. DS1307 là chế tạo bởi Dallas. Chip này có 7 thanh ghi 8 bit mỗi thanh ghi này chứa: Thứ , ngày, tháng, năm, giờ , phút, giây. Ngoài ra DS1307 còn chứa 1 thanh ghi điều khiển ngõ ra phụ và 56 thanh ghi trống các thanh ghi này có thể dùng như là RAM. DS1307 được đọc thông qua chuẩn truyền thông I2C nên do đó để đọc được và ghi từ DS1307 thông qua chuẩn truyền thông này. Do nó được giao tiếp chuẩn I2C nên cấu tạo bên ngoài nó rất đơn giản.

**Sơ đồ chân:**



Trong đó:

|  |  |  |
| --- | --- | --- |
| **Chân** | **Tên** | **Chức năng** |
| 1 | X1 | Kết nối đến thạch anh  32.768Khz làm nguồn dao động cho chip |
| 2 | X2 |
| 3 | VBat | Kết nối đến cực dương của Pin dự phòng, có điện áp tiêu chuẩn khoảng 3V |
| 4 | GND | Kết nối đến mass |
| 5 | SDA | Chân dữ liệu khi kết nối đến bus I2C |
| 6 | SCL | Chân nhận xung clock đồng bộ khi kết nối bus I2C |
| 7 | SQW/OUT | Ngõ xuất xung vuông, tần số có thể lập trình để thay đổi từ 1Hz, 4Khz, 8 Khz, 32 Khz |
| 8 | VCC | Nguồn cấp chính, khoảng 5VDC |

**2.2.1.2 Cách lấy thông tin về thời gian và lịch**

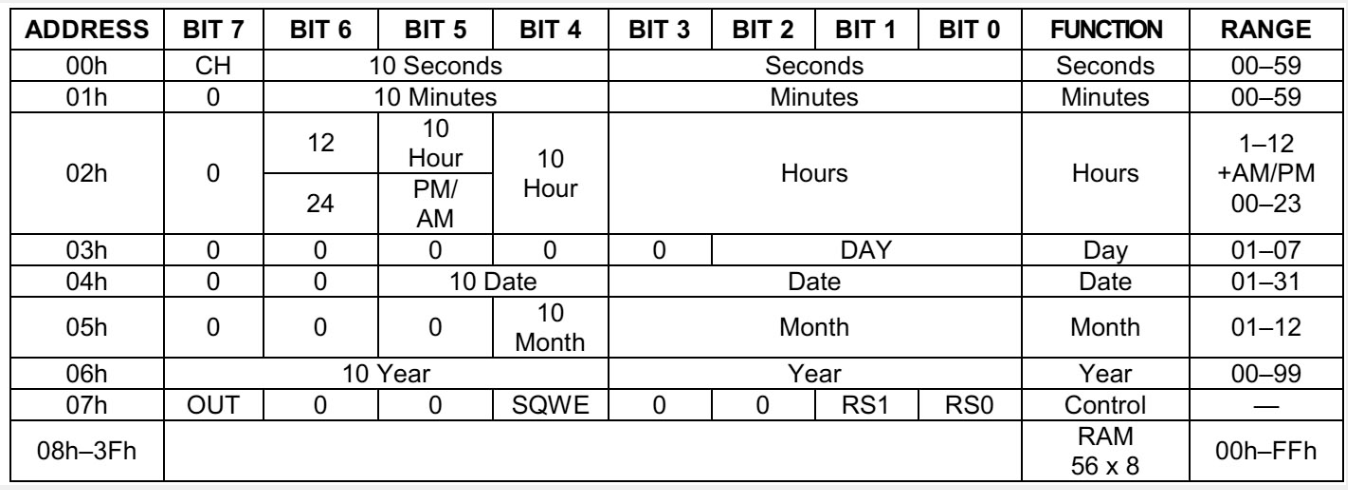
Bảng 1 là địa chỉ của các thanh ghi RAM và thanh ghi RTC của DS 1307. Thanh ghi RTC có địa chỉ từ 00-07h. Các thanh ghi RAM chiếm địa chỉ từ 08h - 3Fh**.**

Các thông tin về thời gian và lịch thu được bằng cách đọc các byte trong thanh ghi tương ứng. Thời gian và lịch được thiết lập hoặc khởi tạo bằng cách viết các byte thanh ghi thích hợp. Lưu ý dữ liệu lưu thời gian chứa trong thanh ghi đều theo định dạng BCD. Thứ trong tuần thay đổi tại lúc nữa đêm, lưu dưới dạng con số (Ví dụ 1 là Chủ Nhật, 2 là Thứ hai, 3 là thứ 3 ...). Khi lần đầu tiên IC được cấp nguồn các thanh ghi thời gian và lịch reset về 01/01/00 01 00:00:00. (MM/DD/YY DOW HH:MM:SS)



### 2.1.2.3 **Chi tiết các thanh ghi.**

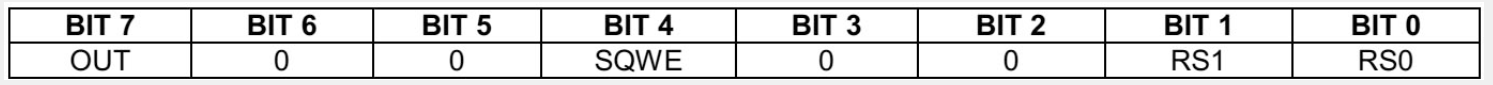
Đây là các Timekeeper registers của DS1307, chúng ta sẽ dựa vào bảng này để read/write IC DS1307 qua I2C.



\*0 luôn luôn đọc là 0

**Bảng 1: Các thanh ghi lưu giữ thời gian**

**Thanh ghi control**



**Bit 7: Output Control (OUT)** thanh ghi điều khiển ngõ ra. Nó ảnh hưởng đến chân số 7 (chân SQW/OUT). Khi sóng vuông xuất ra chân này bị disable, nếu bit OUT=1 khi đó chân số 7 ở mức cao, ngược lại bit OUT=0 khi đó chân số 7 ở mức thấp

**Bit 6:** Luôn luôn đọc bằng 0

**Bit 5:** Luôn luôn đọc bằng 0

**Bit 4: Square-Wave Enable (SQWE):**khi bit này được thiết lập bằng 1, cho phép xuất ra xung vuông tại chân số 7, Tần số của của xung vuông phụ thuộc vào bit RS0 và RS1. Khi tần số sóng vuông được thiết lập là 1 Hz. Các thanh ghi thời gian được cập nhật tại cạnh xuống của xung vuông. Khi lần đầu khởi tạo cấp nguồn đến thiết bị, bit này bằng 0.

**Bits 1 and 0:**Lựa chọn tốc độ(RS[1:0]). Những bit này điều khiển tần số của tần số sóng vuông, khi sóng vuông được cho phép .

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RS1** | **RS0** | **SÓNG VUÔNG NGÕ RA SQW/OUT** | **SQWE** | **OUT** |
| 0 | 0 | 1Hz | 1 | X |
| 0 | 1 | 4.096kHz | 1 | X |
| 1 | 0 | 8.192kHz | 1 | X |
| 1 | 1 | 32.768kHz | 1 | X |
| X | X | 0 | 0 | 0 |
| X | X | 1 | 0 | 1 |

**Thanh ghi giây (địa chỉ 00h)**: 4 bit thấp chứa hàng đơn vị, 3 bit cao chứa hàng chục của giây. Ngoài ra bit thứ 7 có tên là CH, nếu bít này được thiết lập bằng 1 thì đồng hồ không hoạt động. Vì vậy phải thiết lập bit này bằng không ngay từ đầu.

**Thanh ghi phút (địa chỉ 01h)**: 4 bit thấp chứa hàng đơn vị, 3 bit cao chứa hàng chục của phút. Ngoài ra bit thứ 7 luôn bằng 0.

**Thanh ghi giờ (địa chỉ 02h):**4 bit thấp của thanh ghi này chứa hàng đơn vị của giờ, bit thứ 6 quy định chế độ 12 h (bit6 =1) hoặc 24 h (bit 6=0). Nếu ở chế độ 24h thì bit 4 và bit 5 quy định hàng chục của giờ. Nếu chế độ 12h thì bit 4 quy định hàng chục, bit 5 khi đó quy định (PM: buổi chiều hoặc AM: buổi sáng

**Thanh ghi thứ (địa chỉ 03h):**ba bit đầu quy định thứ trong tuần (Ví dụ 1 là Chủ Nhật, 2 là Thứ hai, 3 là thứ 3 ...). Các bit còn lại luôn bằng 0.

**Thanh ghi ngày(địa chỉ 04h):**4 bit đầu lưu hàng đơn vị của ngày, bit 4 và bit 5 quy định hàng chục. Bit 6 và bit 7 luôn luôn bằng 0.

**Thanh ghi tháng(địa chỉ 05h):**4 bit đầu lưu hàng đơn vị của tháng, bit thứ 4 quy định hàng chục. Các bit còn lại luôn bằng 0.

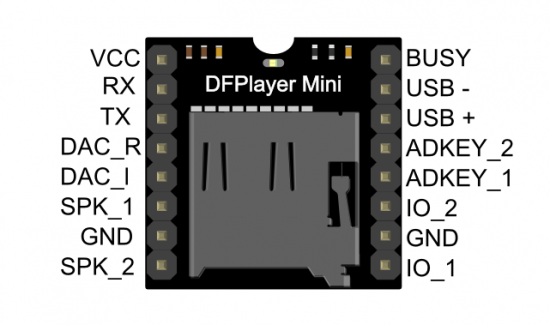
**Thanh ghi năm (địa chỉ 06h):** 4 bit thấp lưu hàng đơn vị và 4 bit cao hàng chục của năm (từ 00 - 99).

### Lập trình Module Ds1307

Chúng ta cần các thư viện giúp ra bóc rách dữ liệu được phân tích phía trên được dẽ dàng. Gồm có rất nhiều thư viện gồm #include <Wire.h>, #include <ds1307.h> .Bạn chỉ add thư viện vào IDE (phần mền Arduino). Cách sử dụng ví dụ bạn dùng thư viên bạn chỉ cần gõ lên.

* + 1. **Module DFplayer mini**

2.1.3.1 **Giới thiệu về DFplayer**



DFPlayer Mini MP3 là mạch phát tập tin âm thanh kiểu máy chơi nhạc MP3. Mạch có thể được sử dụng riêng lẻ chỉ cần pin, loa và nút nhấn hoặc kết hợp với vi điều khiển có giao tiếp chuẩn USART.

**2.1.3.2 Tính năng cơ bản gồm có:**

* Hỗ trợ tập tin mp3, wmv với hầu hết loại tần số lẫy mẫu chuẩn.
* Lối ra DAC 24 bit âm thanh chuẩn
* Thẻ nhớ định dạng FAT16, 32 lên đến 32GB
* Đọc dữ liệu trong tối đa 100 thự mục, mỗi thư mục 255 bài
* Âm lượng 30 mức với 6 loại EQ
* Có sẵn mạch khuếch đại ra loa công suất 2W

**2.3.3 Mạch có thể được sử dụng trong nhiều ứng dụng:**

* Hướng dẫn phân luồng giao thông
* Trạm thông báo, nhắc nhở
* Báo động hoặc cảnh báo ngoài trời, trong nhà
* Quảng báo bán hàng
* Hướng dẫn thông tin, du lịch…

## 

* + - 1. **Chế độ hoạt động**

1. Chế độ Serial

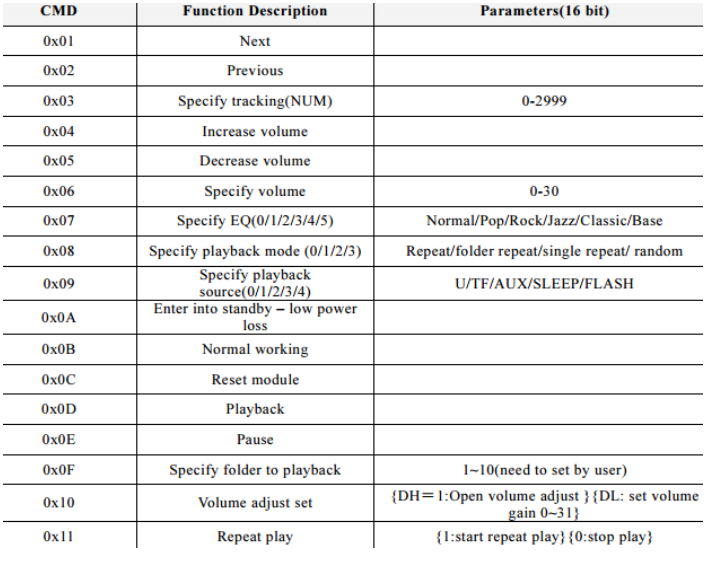
Có hổ trợ trên giao thức serial của cổng com trên máy tính.

Tốc độ board: 9600 bps

Bit Data: 1 bytes

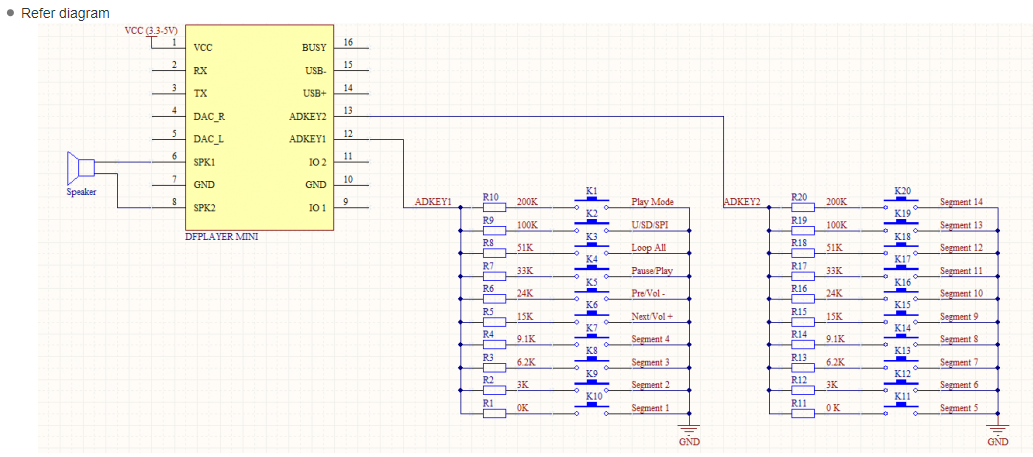
Check Sum: none

Bit Parity: node



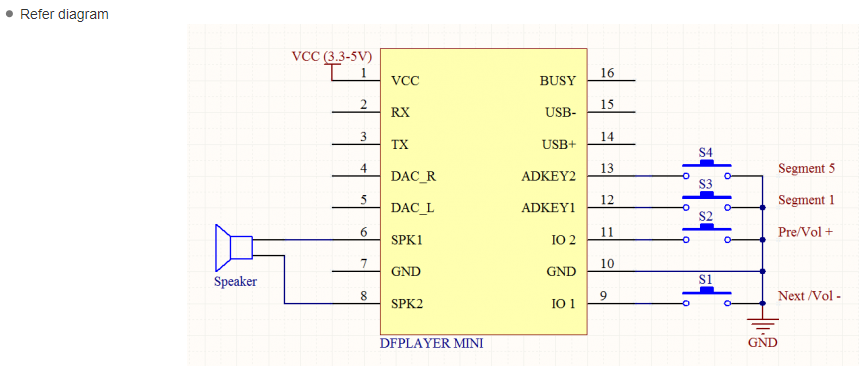
2. Ad key Mode

Chúng tôi sử dụng các phím mô-đun AD, thay vì phương pháp kết nối bàn phím ma trận truyền thống, đó là tận dụng chức năng MCU AD ngày càng mạnh mẽ, Cấu hình mặc định của mô-đun 2 cổng AD, 20 nút nhấn.



### 3) I/O Mode

Đây là cách đơn giản để sử dụng Module.



Trong chế độ này thì các nút S1 và S2 được nhấn trong thời gian ngắn ở port IO2, IO1 là play bài tiếp theo hoặc là play bài ngược lại, còn nếu nhấn lâu hơn thì sẽ chuyển chế độ tăng hoặc giảm âm thanh.

* + - 1. **Lập trình cho DFplayer mini**

Chúng ta đã tạo một thư viện Arduino cho DFPlayer Mini để đơn giản hóa phương pháp giúp làm việc. Kết nối phần cứng như hình trên được hiển thị và phát cùng với mã mẫu. Bạn có thể tải xuống thư viện mới nhất tại đây: **DFRobotDFPlayerMini**.

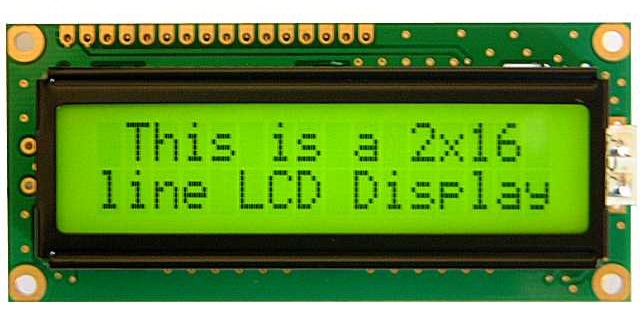
## 2.1.4 Màn hìnhTFT 3.2 và TFT sheild LCD modual



Hình 2.17: Màn hình TFT 3.2

**2.1.4.1 Màn hình TFT 3.2 có chức năng là gì?**

Trước khi nói đến màn hình cảm ứng TFT ta đến với màn hình **LCD** (Liquid Crystal Display - màn hình tinh thể lỏng) là một công nghệ màn hình thường được sử dụng trên nhiều thiết bị. Màn hình LCD không tự tạo ánh sáng mà phải nhờ đến đèn nền để phát sáng.



Mật độ của màn hình LCD rất thấp vì vậy trong ánh sáng mặt trời màu sắc xuất hiện rất kém. Chất lượng của màn hình LCD thay đổi tùy theo quá trình sản xuất và sử dụng, hầu hết các màn hình trên điện thoại giá rẻ hiện nay điều được làm từ màn hình LCD cung cấp màu sắc và góc nhìn rất hẹp.

**Màn hình TFT - LCD**  
  
Màn hình **TFT**(Thin Film Transistor – bóng bán dẫn dạng phim mỏng) bắt đầu được đưa vào smartphone vào năm 2005, có khả năng tái tạo màu tốt hơn và độ phân giải hình ảnh cao hơn so với các màn hình LCD thế hệ trước đó. Do chi phí sản xuất màn hình TFT đã giảm đáng kể từ sau 2005, công nghệ màn hình này đã xuất hiện phổ biến trên điện thoại cơ bản và smartphone giá thấp.



Các bạn đã từng trải nghiệm nhiều phiên bản LCD như LCD 16x02, 20x04, 5110 nokia, Oled, Homephone,… nhưng đối với LCD TFT thì nó sẽ lung linh hơn nhiều, thư viện UTFT thì lại cũng dễ sử dụng như một số LCD cơ bản- Với LCD TFT chúng ta sẽ DIY với những cảm biến như gia tốc, nhiệt độ, ánh sáng,lập một menu button điều khiển xe không cần android … thậm chí nó có thể mang đến cảm giác hoàn hảo tuyệt vời khi chế tạo cùng với chế độ điều khiển servo, robot, led, làm menu smartphone, vận hành máy in3D với loại TFT cao cấp hơn…

## 2.1.4.1****. Thông số kỹ thuật****

Ø Điện áp hoạt động: 5V

Ø Độ phân giải 240x320 với 262K màu

Ø Chuẩn giao tiếp 2 chế độ 8 bit và 16 bit.

Ø Hỗ trợ cảm ứng điện trở, khe cắm thẻ nhớ SD

Sơ đồ chân:

|  |  |  |
| --- | --- | --- |
| **Số thứ tự** | **Tên chân** | **Chức năng** |
| **1** | **GND** | **Nối mass nguồn 5V** |
| **2** | **VCC** | **Nối nguồn 5V** |
| **3,18,19,20,21,22,29** | **NC** | **Not connect** |
| **4** | **RS** | **RS=0 ghi lệnh**  **RS=1 ghi data** |
| **5** | **WR** | **1 xung cạnh lên trên chân WR, sẽ ghi data lên LCD** |
| **6** | **RD** | **1 xung cạnh lên trên chân RD, đọc trạng thái LCD** |
| **7-14** | **DB8-DB15** | **8 bit cao data** |
| **15** | **CS** | **CS=0 : Gửi lệnh lên LCD**  **CS=1: Gửi data lên LCD** |
| **16** | **F-CS** | **F-CS: Sử dụng bộ nhớ flash ngoài** |
| **17** | **REST** | **Reset tích cực âm** |
| **19** | **LED\_A** | **Chân Anot đèn nền** |
| **23** | **SD\_CS** | **Sử dụng giao tiếp SPI với thẻ nhớ SD (chân chọn Slave)** |
| **24** | **SD\_DIN** | **Sử dụng giao tiếp SPI với thẻ nhớ SD (gửi lệnh command qua chân này)** |
| **25** | **SD\_CLK** | **Sử dụng giao tiếp SPI với thẻ nhớ SD (chân tạo xung clock)** |
| **26** | **SD\_DO** | **Sử dụng giao tiếp SPI với thẻ nhớ SD (chân đọc data từ thẻ nhớ SD)** |
| **27** | **T-IRQ** | **Khi có cảm ứng, chân này được kéo xuống mức 0.** |
| **28** | **T-DO** | **Sử dụng chức năng cảm ứng, giao tiếp SPI (đọc data-nối MISO)** |
| **30** | **T-DIN** | **Sử dụng chức năng cảm ứng, giao tiếp SPI (gửi data-nối MOSI)** |
| **31** | **T-CS** | **Sử dụng chức năng cảm ứng, giao tiếp SPI (chọn Slave)** |
| **32** | **T-CLK** | **Sử dụng chức năng cảm ứng, giao tiếp SPI (chân tạo xung clock)** |
| **33-40** | **DB7-DB0** | **8 bit thấp data(Nối GND trong mode 8 bits)** |

**2.1.4.1 Phương thức giao tiếp.**

Giao tiếp TFT với vi điều khiển: Sử dụng vào ra IO, giao tiếp SPI

Chức năng screen (hiển thị):

+Giao tiếp 8 bit: Sử dụng 8 bit cao, 8 bit thấp nối GND.

+Giao tiếp 16 bit: Sử dụng 8 bit cao và 8 bit thấp.

Trong chế độ 8 bit: Gửi 8 bit cao trước, sau đó gửi 8 bit thấp.

Ví dụ : VH, VL theo thứ tự là 8 bits cao và 8 bits thấp của data.

xuat\_data\_8 bit(VH);

LCD\_WR\_OFF;

LCD\_WR\_ON;

xuat\_data\_8 bit(VL);

LCD\_WR\_OFF;

LCD\_WR\_ON;

- Nhận xét:

+ Ưu điểm: Tiết kiệm được chân vi điều khiển (chỉ sử dụng 8 chân để truyền data cho màn hình TFT)

+ Nhược điểm: Tốc độ chậm hơn so với 16 bit.

Trong chế độ 16 bit: Sử dụng cả 16 chân DB0-DB15 để truyền data

Sample: VH, VL theo thứ tự là 8 bits cao và 8 bits thấp của data.

PORT\_H\_Data=VH;

LCD\_WR\_OFF;

LCD\_WR\_ON;

PORT\_L\_Data=VL;

LCD\_WR\_OFF;

LCD\_WR\_ON;

- Nhận xét:

+ Ưu điểm:Tốc độ nhanh vì chế độ là song song, trực tiếp.

+ Nhược điểm: Không tiết kiệm chân vi điều khiển sử dụng

-Quá trình khởi tạo màn hình TFT (theo thứ tự hàm init)

\* Đưa chân CS =1, tạo 1 xung cạnh xuống và 1 xung cạnh lên trên chân RESET để đánh thức IC màn hình TFT

\* Tiếp tục đưa chân RD và chân WR =1 để ghi lệnh cho TFT.

* + 1. **Arduino mega 2560**

Mega 2560 ATmega2560-16A U Ban đối với arduino + 3.2 "TFT LCD + 3.2" TFT LCD Lá Chắn

Mega 2560 ATmega2560-16AU Board Arduino tương thích:

Mega 2560 là bo mạch vi điều khiển dựa trên ATmega2560. Nó có 54 chân đầu vào / đầu ra kỹ thuật số (trong đó có 14 đầu ra PWM), 16 đầu vào analog, 4 UART (cổng nối tiếp phần cứng), bộ dao động tinh thể 16 MHz, kết nối USB, giắc cắm nguồn, tiêu đề ICSP và nút đặt lại. Nó chứa mọi thứ cần thiết để hỗ trợ vi điều khiển; chỉ cần kết nối nó với một máy tính bằng cáp USB hoặc cấp điện cho nó bằng bộ chuyển đổi AC-to-DC hoặc pin để bắt đầu. Mega tương thích với hầu hết các khiên được thiết kế cho Arduino Duemilanove hoặc Diecimila.



Mega 2560 là bản cập nhật cho Arduino Mega. Tóm lược vi điều khiển ATmega2560, điện áp hoạt động 5V, điện áp đầu vào (được khuyến nghị) 7-12V, điện áp đầu vào (giới hạn) 6-20V, chân I / O kỹ thuật số 54 (trong đó 14 cung cấp đầu ra PWM). Pins đầu vào tương tự 16, DC hiện tại cho mỗi I / O Pin 40 mA DC hiện tại cho 3.3V Pin 50 mA, bộ nhớ flash 256 KB trong đó 8 KB được trình tải khởi động sử dụng SRAM 8 KB, EEPROM 4 KB, xung clock : 16 MHz.

Arduino Mega 2560 là một vi điều khiển hoạt động dựa trên chip ATmega2560. Bao gồm:

* 54 chân digital (trong đó có 15 chân có thể được sủ dụng như những chân PWM là từ chân số 2 → 13 và chân 44 45 46).
* 6 ngắt ngoài: chân 2 (interrupt 0), chân 3 (interrupt 1), chân 18 (interrupt 5), chân 19 (interrupt 4), chân 20 (interrupt 3), and chân 21 (interrupt 2).
* 16 chân vào analog (từ A0 đến A15).
* 4 cổng Serial giao tiếp với phần cứng:

| **CỔNG SERIAL** | **CHÂN RX** | **CHÂN TX** |
| --- | --- | --- |
| Cổng 0 | 0 | 1 |
| Cổng 1 | 19 | 18 |
| Cổng 2 | 17 | 16 |
| Cổng 3 | 15 | 14 |

* 1 thạch anh với tần số dao động 16 MHz.
* 1 cổng kết nối USB.
* 1 jack cắm điện.
* 1 đầu ICSP.
* 1 nút reset.
  + 1. Loa 0.5 w



Tính năng chính:

Loa màng rung 8 ohm 0.5W có chất lượng hoàn thiện tốt, được sử dụng rộng rãi cho các dự án âm thanh của bạn. Lắp đặt dễ dàng, thuận tiện.

Ứng dụng: loa 8 ohm 0.5W thích họp cho việc chế tạo loa hộp mini, máy nghe nhạc, chuông cửa, máy tính, loa đa phương tiện, các dự án về cảnh báo âm thanh, âm ly đơn giản, các dự án chế tạo robot…

Thông số kỹ thuật:

Trở kháng: 8ohm.

Công suất: 0.5W.

Hình dạng: dạng tròn.

Đường kính loa: 50mm.

Chất liệu: nhôm, nhựa, nam châm.

Trọng lượng: 42g.

## 

## 2.6. Giới thiệu tổng quát về Blynk

### 2.6.1 App Blynk là gì?

Blynk là một ứng dụng chạy trên nền tảng iOS và Android để điều khiển và giám sát thiết bị thông qua internet. Blynk không bị ràng buộc với những phần cứng cụ thể nào cả, thay vào đó, nó hỗ trợ phần cứng cho bạn lựa chọn như Arduino, Raspberry Pi, ESP8266 và nhiều module phần cứng phổ biến khác.



Hình 2.19: App Blynk

### 2.6.2 Tại sao phải dùng App Blynk

– Dễ sử dụng: việc cài đặt ứng dụng và đăng ký tài khoản trên điện thoại rất đơn giản cho cả IOS và Android

– Chức năng phong phú: Blynk hỗ trợ rất nhiều chức năng với giao diện đẹp và thân thiện, bạn chỉ việc kéo thả đối tượng và sử dụng nó.

– Không phải lập trình ứng dụng: nếu bạn không có kiến thức về lập trình app cho Android cũng như IOS thì Blynk là một ứng dụng tuyệt vời để giúp bạn khám phá thế giới IOTs.

– Điều khiển, giám sát thiết bị ở bất kì đâu thông qua internet với khả năng đồng bộ hóa trạng thái và thiết bị.

**- Dễ sử dụng**: Quá đơn giản, chỉ việc vào store, cài đặt, sau đó đăng ký tài khoản và mất không quá 5 phút để làm quen.

**- Đẹp và đầy đủ**: Giao diện của Blynk quá tuyệt vời, sử dụng bằng cách kéo thả, bạn cần nút bấm, kéo thả nút bấm, bạn cần đồ thị, kéo thả đồ thị, bạn cần LCD, kéo thả LCD, tóm lại là bạn cần gì thì kéo thả cái đó.

**- Thử nghiệm nhanh chóng, có thể điều khiển giám sát ở bất kỳ nơi nào có internet.**

## 2.7. Hướng dẫn sử dụng App Blynk

Blynk thực ra là một cái app trên điện thoại, cho phép người dùng có thể tạo ra giao diện và điều khiển thiết bị theo ý thích của cá nhân:

Tất nhiên ngoài những điểm lợi từ blynk thì còn có những cái hạn chế như phải mua energy để tạo được nhiều giao diện và chia sẻ giao diện cho người khác. Những cái này cũng không phải là vấn đề lớn lắm.

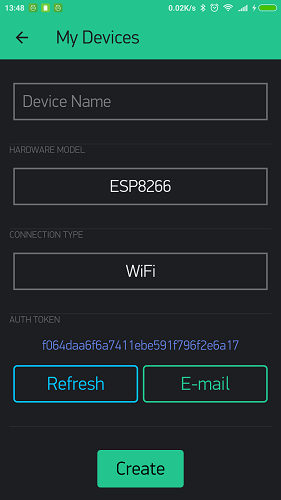
### 2.7.1 Tải ứng dụng Blynk trên ANDROID hoặc IOS



### 2.7.2 Lấy mã AUTH TOKEN

Để kết nối với ứng dụng Blynk và phần cứng của bạn, bạn cần có một mã Token Xác thực.

1. Tạo tài khoản trên ứng dụng Blynk của máy chủ Việt Nam. Phần Custom điền (IP: cloud.blynk.vn và Port: 8443)
2. Tạo một dự án mới. Sau đó, chọn bảng kết nối mà bạn sẽ sử dụng.
3. Sau khi tạo dự án mới thành công, bạn cần sao chép hoặc gửi mã xác thực Auth Token qua email tài khoản.
4. Kiểm tra email trong hộp thư đến và tìm mã xác thực Auth Token.



Hình 2.20: Auth token

### 2.7.3 Cài đặt thư viện Blynk

* **Thư viện Blynk nên được cài đặt bằng tay. Làm theo hướng dẫn:**

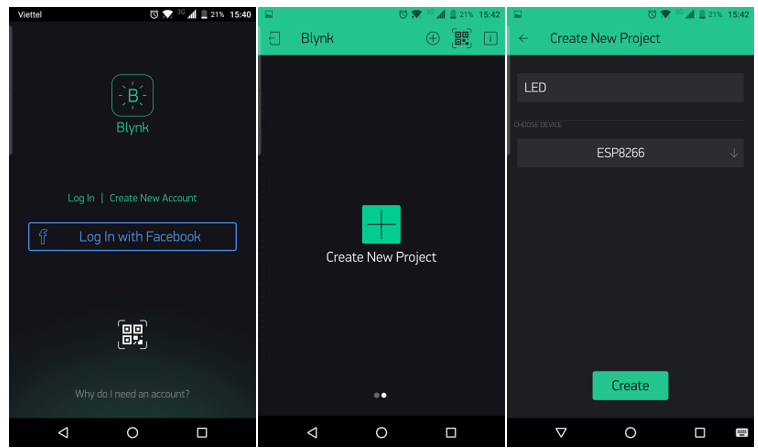


1. Tải thư viện **Blynk.zip** phía trên.
2. Giải nén thư viện **Blynk.zip**. Bạn sẽ thấy thư mục chứa thư viện Blynk.
3. Sao chép thư mục **Blynk**vào **thư viện libraries trên Arduino IDE** (thư mục libraries trên  **Arduino IDE**. Nếu không thấy xin vào trên phần mềm  **Arduino IDE: Chọn *File -> Preferences***(lúc này sẽ thấy đường dẫn chứa thư viện libraries *)*

Ví Dụ: **Documents\Arduino\libraries\Blynk**...

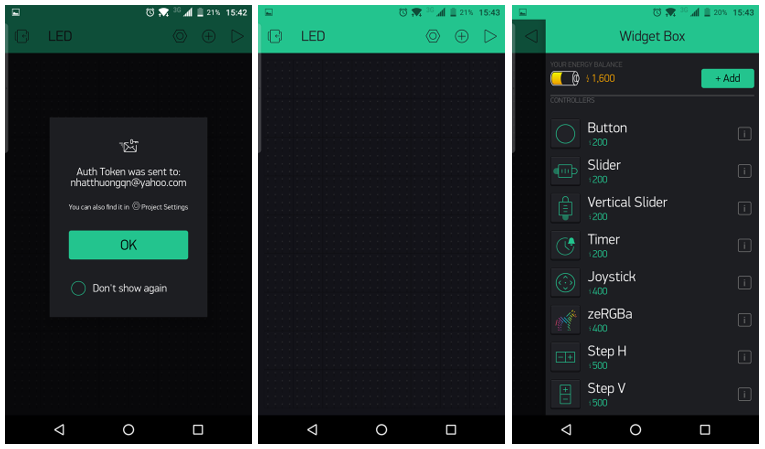
 Lưu ý rằng thư viện Blynk sẽ hoạt động nếu bạn chọn đúng đường dẫn thư viện.

### 2.7.4 Tạo code ví dụ

Để kết nối với ứng dụng Blynk và bộ điều khiển bạn cần có một mã Token xác thực.**Trước hết bạn phải có một tài khoản gmail, yahoo ,Facebook**….hoặc chúng ta tạo một tài khoản mới. Sau khi có tài khoản, khởi động chương trình Blynk được cài đặt trên điện thoại lúc này giao diện blynk và thực hiện các bước như hình sau:  
1.Tạo một project, đây được hiểu giống như là một ứng dụng.  
2. Điền tên Project và chọn Board phần cứng (Các bạn có thể chọn NodeMCU hoặc ESP8266).

Hình 2.21 Giao diện của App Blynk

### Mỗi project, Blynk sẽ gửi cho bạn 1 mã Auth Token để nhập vào trong code của Board mạch điều khiển NodeMCU, Các bạn có thể lựa chọn nhiều chức năng như nút bấm, hẹn giờ, LCD… để đưa vào project của mình. Mỗi đối tượng các bạn chọn sẽ tốn energy (1 đơn vị giới hạn khi bạn dùng server miễn phí).



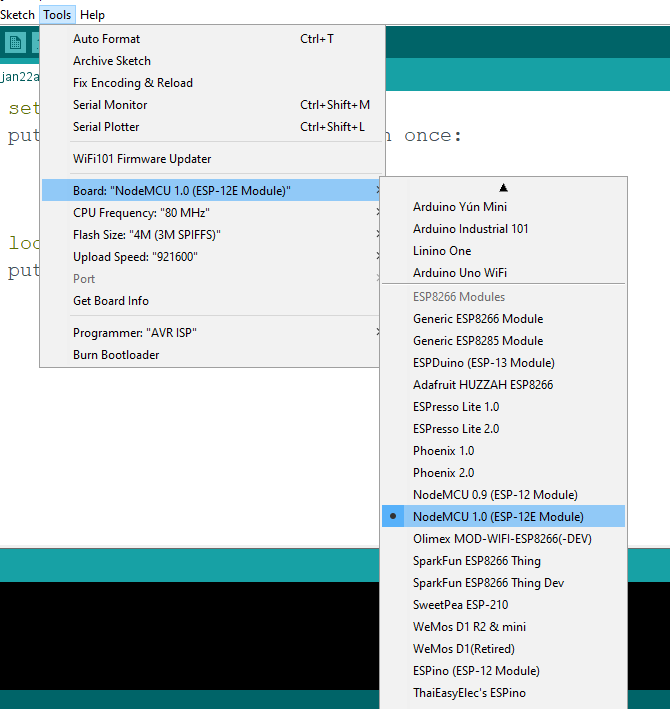
Hình 2.22. Thanh công cụ của Blynk

Sau khi chọn xong đối tượng, các bạn nhấp vào để cấu hình chân, các mức logic… Các cấu hình này sẽ tác động đến board phần cứng. Ví dụ nhất nút thì chân GPIO16 sẽ chuyển từ trạng thái logic 1 sang logic 0:

### Kết quả hình ảnh cho Cấu hình app Blynk theo các bước sau:

Hình 2.23 Thiết lập cụ để Blynk

Sau khi cài đặt xong trên điện thoại thì các bạn lập trình cho board phần cứng. Thư viện Blynk trên Arduino IDE hỗ trợ rất nhiều ví dụ cho Blynk để các bạn có thể dùng thử, hiểu cách thức hoạt động…



### 2.7.5 Dán mã xác thực

• Tên code ví dụ bạn tìm dòng code dưới đây:

char auth[] = "YourAuthToken";

• Sửa mã xác thực **YourAuthToken** (Mã xác thực nằm trong email của bạn sau khi bạn tạo dự án trong ứng dụng Blynk

Ví dụ: char auth[] = "53e4da8793764b6197fc44a673ce4e21";

• Upload code cho phần cứng của bạn

• Mở seri monitor trên Arduino IDE. Bạn sẽ thấy nónhư thế này

     \_\_\_  \_\_          \_\_

    / \_ )/ /\_ \_\_\_\_\_  / /\_\_

   / \_  / / // / \_ \/  '\_/

  /\_\_\_\_/\_/\\_, /\_//\_/\_/\\_\

         /\_\_\_/    v0.4.4

    [1240] Connecting to YourWiFi

    [1240] Connected to YourWiFi

    [1240] My IP: 192.168.10.172

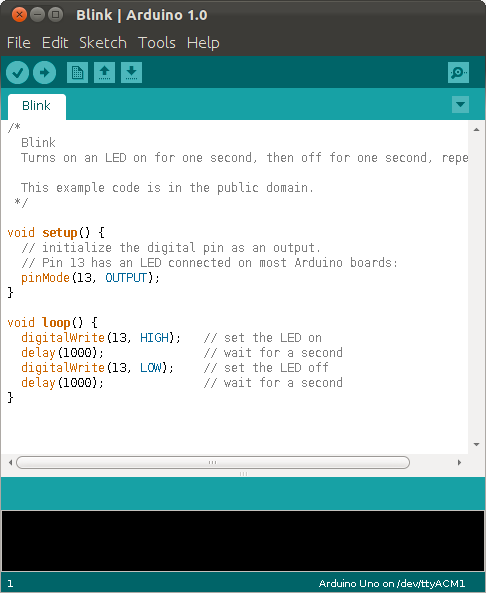
    [1240] Blynk v0.4.4

    [5001] Connecting to cloud.blynk.vn:8442

    [5329] Ready (ping: 1ms)

 **Hoàn tất! nó đã kết nối!**

## 2.8. Tìm hiểu phần mềm lập trình Arduino IDE:



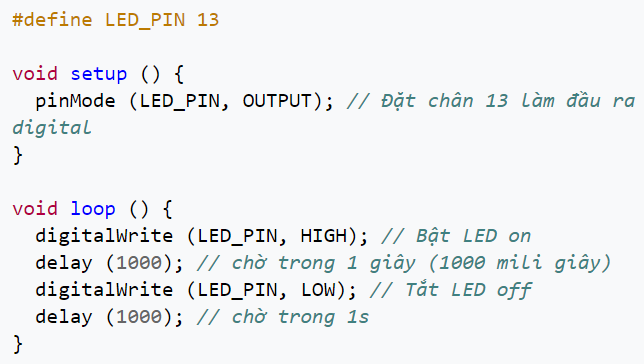
Hình 2.24: Phần mềm lập trình Arduino IDE

Arduino IDE là một ứng dụng [cross-platform](https://vi.wikipedia.org/wiki/Cross-platform) (nền tảng) được viết bằng [Java](https://vi.wikipedia.org/w/index.php?title=Java_(programming_language)&action=edit&redlink=1), và từ IDE này sẽ được sử dụng cho [Ngôn ngữ lập trình xử lý](https://vi.wikipedia.org/w/index.php?title=Ngôn_ngữ_lập_trình_xử_lý&action=edit&redlink=1) (Processing programming language) và project [Wiring](https://vi.wikipedia.org/w/index.php?title=Wiring_(development_platform)&action=edit&redlink=1). Nó được thiết kế để dành cho những người mới tập làm quen với lĩnh vực phát triển phần mềm. Nó bao gồm một chương trình code editor với các chức năng như đánh dấu cú pháp, tự động [brace matching](https://vi.wikipedia.org/w/index.php?title=Brace_matching&action=edit&redlink=1), và tự động canh lề, cũng như compile(biên dịch) và upload chương trình lên board chỉ với 1 cú nhấp chuột. Một chương trình hoặc code viết cho Arduino được gọi là một *sketch*.

Các chương trình Arduino được viết bằng [C](https://vi.wikipedia.org/w/index.php?title=C_(programming_language)&action=edit&redlink=1) hoặc [C++](https://vi.wikipedia.org/wiki/C%2B%2B). Arduino IDE đi kèm với một [thư viện phần mềm](https://vi.wikipedia.org/w/index.php?title=Thư_viện_phần_mềm&action=edit&redlink=1) được gọi là "[Wiring](https://vi.wikipedia.org/w/index.php?title=Wiring_(development_platform)&action=edit&redlink=1)", từ project Wiring gốc, có thể giúp các thao tác input/output được dễ dàng hơn. Người dùng chỉ cần định nghĩa 2 hàm để tạo ra một chương trình [vòng thực thi](https://vi.wikipedia.org/w/index.php?title=Vòng_thực_thi&action=edit&redlink=1) (cyclic executive) có thể chạy được:

* setup(): hàm này chạy mỗi khi khởi động một chương trình, dùng để thiết lập các cài đặt
* loop(): hàm này được gọi lặp lại cho đến khi tắt nguồn board mạch

Một chương trình điển hình cho một bộ vi điều khiển đơn giản chỉ là làm cho một bóng đèn Led sáng/tắt. Trong môi trường Arduino, ta sẽ phải viết một chương trình giống như sau:



Arduino IDE này sử dụng [GNU toolchain](https://vi.wikipedia.org/w/index.php?title=GNU_toolchain&action=edit&redlink=1) và AVR Libc để biên dịch chương trình, và sử dụng avrdude để upload chương trình lên board.

## 2.9 Thiết kế mô hình phần cứng và phần mềm lâp trình cho hệ thống.

### 2.9.1 Sơ đồ kết nối

Hình 6.1: Sơ đồ kết nối hệ thống với ESP8266 NODEMCU

### 2.9.2 Sơ đồ giải thuật:

Đọc từ tọa độ bằng module neo 6m

Đưa vi trị lên Map ở App Blynk

Lưu vị trị đó vào thẻ nhớ

Module SD

Wiffi bị mất

App blynk bật thông báo “ Wifi is offine” và

Báo về Gmail của mình

Tắt thông báo

Đúng

Sai

Nodemcu esp8266

# 

Nhận dữ liệu Module Neo 6-m từ NodeMCU hiển thị và lưu tọa độ

Đọc trạng thái Relay từ App truyền xuống NodeMCU

App Blynk

# CHƯƠNG III: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Sau nhiều tháng thực hiện nỗ lực và cố gắng cùng với sự hướng dẫn tận tình của cô Ths.Nguyễn Lan Anh, tập nghiên cứu đã hoàn thành đúng thời hạn theo quy định, theo yêu cầu đặt ra nghiên cứu ứng dụng quản lý trẻ em bằng định GPS:

Tìm hiểu nguyên lý định vị GPS và các mô hình điều khiển bằng wiffi.

Tìm hiểu các thiết bị linh kiện sử dụng trong hệ thống như: Module wifi ESP8266, vi điều khiển, các module định vị ...

Tìm hiểu các phần mềm thiết kế và thực hiện hệ thống: Phần mềm tạo APP, phần mềm lập trình vi điều khiển ...

**Hướng phát triễn của đề tài**:

- Cần nghiên cứu thực tế khi GPS không nhận được tọa độ.

- Phát triển, nâng cấp các cảm biến nhằm thu thập được chính xác tọa độ khi ở trong nhà ở hay nhà xe.

- Kết hợp sử dụng điều khiển bằng giọng nói, đồng thời không cần sử dụng đến wiffi .

- Nên tích hợp cực nhỏ có thể.

Một lần nữa em chân thành cảm ơn Ths. Nguyễn Lan Anh đã tận tình giúp đỡ, chỉ bảo và đưa hướng đi đúng đắn cho em trong thời gian qua, giúp em có thể hoàn thành tốt những nhiệm vụ đã đặt ra.

Trong quá trình thực hiện nghiên cứu không tránh khỏi những sai sót, em rất mong nhận được sự chia sẻ, chỉ bảo của thầy cô và bạn bè. Em xin chân thành cảm ơn.

# TÀI LIỆU THAM KHẢO

[1] Tài liệu tự học ESP8266 NODEMCU – Câu lạc bộ điện tử đại học Bách Khoa Tp.HCM

[2] Th.s Phạm Thế Duy, Bài giảng Kĩ thuật vi xử lý

[3] Phan Hữu Phước, Tài liệu Mạng Máy Tính

**Các trang web tham khảo:**

[4] https://hocarm.org/esp8266-cho-nguoi-khong-biet-gi/

[5] <https://arduino.esp8266.vn/basic/led.html>

[6] https://khoere.com/21/gps-la-gi-lap-trinh-giao-tiep-voi-no-dung-dieu-khien-nhu-the-nao

[7] https://www.youtube.com/watch?v=6XBIsBQcYtA&t=550s