

**TRƯỜNG ĐẠI HỌC TIỀN GIANG**

-----🙞🙜🕮🙞🙜-----

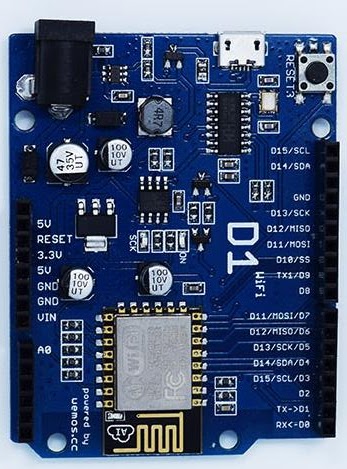
**Khoa Công Nghệ Thông Tin**



**ĐỒ ÁN CHUYÊN NGÀNH**

**CÔNG NGHỆ THÔNG TIN**

**Đề tài: Tìm hiểu IoT xây dựng ứng dụng**

**điều khiển đèn LED bằng Anduino trên Web**

Giảng viên hướng dẫn: ***ThS NGUYỄN MINH KHOA***

Sinh viên thực hiện:

* Phạm Ngọc Đăng Khoa 014101010
* Nguyễn Hoàng Minh 014101011

TP.Mỹ Tho, tháng 06 năm 2017

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC TIỀN GIANG  **KHOA CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập – Tự do – Hạnh phúc** |
|  | *Tiền Giang, ngày ….. tháng …. năm ……* |

**PHIẾU CHẤM ĐIỂM ĐỒ ÁN CHUYÊN NGÀNH**

(Dùng cho người hướng dẫn và người phản biện)

**Tên đề tài**:

**Họ và tên sinh viên thực hiện:**

1. ……………………………………………. MSSV: ………………….

2. ……………………………………………. MSSV: ………………….

**Họ và tên người chấm điểm:**

**Người chấm điểm là:**  🞏 Người hướng dẫn 🞏 Người phản biện

Người chấm điểm dựa vào việc đọc quyển báo cáo, xem demo sản phẩm, dự buổi báo cáo để ghi nhận xét trên Phiếu đánh giá đồ án. Dựa trên Phiếu đánh giá, người chấm cho điểm đồ án theo thang điểm sau:

|  |  |  |
| --- | --- | --- |
| **Nội dung đánh giá** | **Điểm tối đa** | **Điểm thực chấm** |
| **1. QUYỂN BÁO CÁO** | **4** |  |
| **Về hình thức** | 0.5 |  |
| **Về tóm tắt** | 0.5 |  |
| **Về phần tổng quan** | 1 |  |
| **Về phần cơ sở lý thuyết và thực tiễn** | 0.5 |  |
| **Về phân tích, thiết kế, cài đặt và kết quả** | 1.5 |  |
| **2. SẢN PHẨM/ CHƯƠNG TRÌNH DEMO** | **3** |  |
| Chức năng của sản phẩm đáp ứng yêu cầu đặt ra | 2 |  |
| Sản phẩm có khả năng ứng dụng trong thực tiễn | 1 |  |
| **3. BÁO CÁO VÀ TRẢ LỜI CHẤT VẤN** | **2** |  |
| Báo cáo tốt | 1 |  |
| Trả lời chất vấn tốt | 1 |  |
| **4. TINH THẦN THÁI ĐỘ LÀM VIỆC** | **1** |  |
| **TỔNG CỘNG** | **10** |  |

|  |  |
| --- | --- |
|  | **Người chấm điểm**  (Ký và ghi họ tên) |

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC TIỀN GIANG  **KHOA CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập – Tự do – Hạnh phúc** |
|  | *Tiền Giang, ngày ….. tháng …. năm .…* |

**PHIẾU ĐÁNH GIÁ ĐỒ ÁN CHUYÊN NGÀNH**

(Dùng cho người hướng dẫn và người phản biện đồ án)

**Tên đề tài**:

**Họ và tên sinh viên thực hiện:**

1. ……………………………………………. MSSV: …………………

2. ……………………………………………. MSSV: …………………

**Họ và tên người đánh giá:**

**Người đánh giá là:**  🞏 Người hướng dẫn 🞏 Người phản biện

Người đánh giá dựa vào việc đọc quyển báo cáo, xem demo sản phẩm, dự phiên chấm đồ án để nhận xét đồ án theo các gợi ý sau:

**1. Về quyển báo cáo**

**Về hình thức**: quyển báo cáo có hình thức trình bày và cấu trúc theo đúng quy định? Câu văn có mạch lạc, sáng sủa? Không có lỗi chính tả?

**Về phần tổng quan**: tác giả có trình bày một cách rõ ràng các nội dung cần thiết hay không? Nội dung cần thiết có thể bao gồm: mục tiêu, nhiệm vụ, phạm vi của đồ án; các chức năng, công dụng của phần mềm cần đạt được trong đồ án; bối cảnh thực tiễn, ứng dụng của đồ án; giải thuật, phương pháp, cách thức giải quyết bài toán, nhiệm vụ đặt ra trong đồ án; công nghệ được sử dụng trong đồ án...

**Về phân tích, thiết kế, cài đặt giải pháp và kết quả đề tài:** Các giải pháp phân tích, thiết kế, cài đặt để giải quyết nhiệm vụ của đồ án có hợp lý? Đề tài mới hoặc phương pháp thực hiện có tính sáng tạo? Chất lượng và ý nghĩa của kết quả đạt được như thế nào?

**2. Về sản phẩm/ chương trình demo:** các chức năng của chương trình có đáp ứng yêu cầu đặt ra? Sản phẩm có khả năng ứng dụng trong thực tiễn?

**3. Về báo cáo và trả lời chất vấn:** báo cáo có tốt không? (chuẩn bị slide tốt, trình bày rõ ràng, đúng thời gian). Trả lời chất vấn có tốt không (hiểu đúng câu hỏi, trả lời đúng và tập trung vào vấn đề đặt ra, không lạc đề)?

|  |  |
| --- | --- |
|  | **Người đánh giá**  (Ký và ghi họ tên) |

**MỤC LỤC**

[Lời Nói Đầu. 7](#_Toc517712468)

[Chương 1: Tổng quan về Arduino Wemos D1. 1](#_Toc517712469)

[1. Tổng quan. 1](#_Toc517712470)

[2. Sơ đồ chân của WEMOS D1. 3](#_Toc517712471)

[Chương 2: Cài đặt chương trình Arduino IDE và Driver cho Arduino 5](#_Toc517712472)

[1. Cài đặt Java Runtime Environment (JRE) 5](#_Toc517712473)

[2. Cài đặt Arduino IDE 5](#_Toc517712474)

[3. Cài đặt Driver 9](#_Toc517712475)

[4. Dạo một vòng Arduino IDE 12](#_Toc517712476)

[**4.1 Giao diện** 12](#_Toc517712477)

[**4.3 Vùng viết chương trình** 13](#_Toc517712478)

[**4.4 Vùng thông báo (debug)** 13](#_Toc517712479)

[**4.5 Một số lưu ý** 14](#_Toc517712480)

[Chương 3:Điều khiển bóng đèn led thông qua wifi. 15](#_Toc517712481)

[1) Phần 1: Lập trình mạch . 15](#_Toc517712482)

[Tài liệu tham khảo. 24](#_Toc517712483)

**DANH SÁCH HÌNH**

[Hình 1:Arduino WeMos D1 R2 3](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712484)

[Hình 2: Bản JRE mới nhất là 7u80 5](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712485)

[Hình 3: Tải xuống Arduino IDE 6](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712486)

[Hình 4: Đóng góp cho phần mềm Arduino 6](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712487)

[Hình 5: Tải phần mềm Arduino 7](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712488)

[Hình 6: Giải nén file vừa tải 7](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712489)

[Hình 7: Giao diện phần mềm Arduino 8](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712490)

[Hình 8: Device Driver Installation Wizard 9](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712491)

[Hình 9: Bắt đầu cài đặt driver 10](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712492)

[Hình 10: Quá trình cài đặt driver 10](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712493)

[Hình 11: Hoàn tất quá trình cài đặt driver 11](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712494)

[Hình 12: Phân tích giao diện Arduino IDE 12](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712495)

[Hình 13: Các công cụ và chức năng của các ion 13](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712496)

[Hình 14: Vùng thông báo 14](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712497)

[Hình 15: Một số lưu ý 14](file:///C:\Users\DANKHOA\Desktop\DO_AN_CHUYEN_NGANH\DOANCHUYENNGANH.docx#_Toc517712498)

# Lời Nói Đầu

Arduino đã và đang được sử dụng rất rộng rãi trên thế giới, và ngày càng chứng tỏ được sức mạnh của chúng thông qua vô số ứng dụng độc đáo của người dùng trong cộng đồng nguồn mở. Arduino thực sự đã gây sóng gió trên thị trường người dùng trên toàn thế giới trong vài năm gần đây, số lượng người dùng cực lớn và đa dạng với trình độ trải rộng từ bậc phổ thông lên đến đại học đã làm cho ngay cả những người tạo ra chúng phải ngạc nhiên về mức độ phổ biến. Tuy nhiên tại Việt Nam Arduino vẫn còn chưa được biết đến nhiều, các tài liệu liên quan đến nó vẫn còn rất hạn chế. Được sự giới thiệu và chỉ dẫn của thầy **NGUYỄN MINH KHOA** sau một thời gian tìm hiểu chúng em đã biên soạn tài liệu “**Lập trình Arduino xây dựng ứng dụng điều khiển đèn LED**”. Trong tài liệu này cung cấp cho bạn đọc một lượng kiến thức cơ bản nhất về Arduino cũng như các ứng dụng thực tế của nó.

Tài liệu gồm có các nội dung sau:

*Chương 1: Tổng quan về Arduino Wemos D1.*

*Chương 2: Cài đặt chương trình Arduino IDE và Driver cho Arduino. Chương 3: Điều khiển bóng đèn led thông qua wifi.*

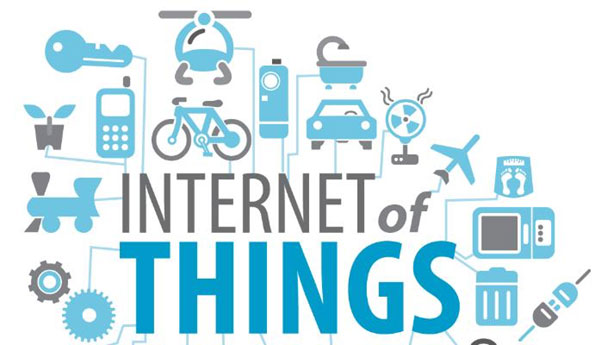
Khi viết chúng em đã có gắng để tài liệu được hoàn chỉnh nhất song chắc chắn không tránh khỏi sai sót, vì vậy rất mong nhận được sự góp ý kiến từ phía thầy cô.

# Chương 1: TỔNG QUAN VỀ INTERNET OF THINGS VÀ ARDUINO

## Tổng quan:

* 1. **Internet of things (IoT) là gì?**

Thuật ngữ ” *Internet of things*”( viết tắt là IoT) dạo gần đây xuất hiện khá nhiều và thu hút không ít sự quan tâm chú ý của thế giới công nghệ. Vì sự bùng nổ của IoT trong tương lai sẽ có tác động mãnh mẽ tới cuộc sống, công việc và xã hội loài người.

Thực tế, Internet of things đã manh nha từ nhiều thập kỹ trước. Tuy nhiên mãi đến năm 1999 cụm từ IoT mới được đưa ra bởi Kevin Ashton, Ông là một nhà khoa học đã sáng lập ra Trung tâm Auto-ID ở đại học MIT, nơi thiết lập các quy chuẩn toàn cầu cho RFID (một phương thức giao tiếp không dây dùng sóng radio) cũng như một số loại cảm biến khác.

Hình 1: Internet of things

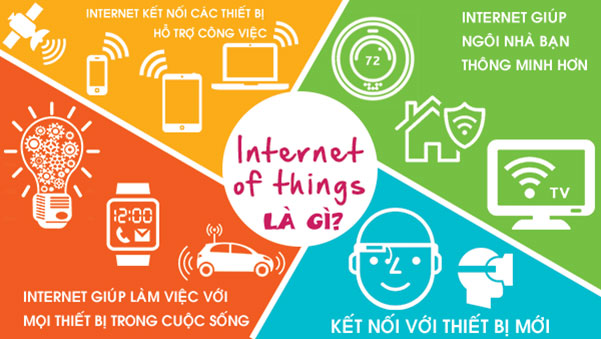
**Internet of things là gì?**  
Mạng lưới vạn vật kết nối Internet hoặc là Mạng lưới thiết bị kết nối Internet viết tắt là IoT (tiếng Anh: Internet of Things) là một kịch bản của thế giới, khi mà mỗi đồ vật, con người được cung cấp một định danh của riêng mình, và tất cả có khả năng truyền tải, trao đổi thông tin, dữ liệu qua một mạng duy nhất mà không cần đến sự tương tác trực tiếp giữa người với người, hay người với máy tính. IoT đã phát triển từ sự hội tụ của công nghệ không dây, công nghệ vi cơ điện tử và Internet. Nói đơn giản là một tập hợp các thiết bị có khả năng kết nối với nhau, với Internet và với thế giới bên ngoài để thực hiện một công việc nào đó.

Hay hiểu một cách đơn giản IoT là tất cả các thiết bị có thể kết nối với nhau . Việc kết nối thì có thể thực hiện qua Wi-Fi, mạng viễn thông băng rộng (3G, 4G), Bluetooth, ZigBee, hồng ngoại… Các thiết bị có thể là điện thoại thông minh, máy pha cafe, máy giặt, tai nghe, bóng đèn, và nhiều thiết bị khác. Cisco, nhà cung cấp giải pháp và thiết bị mạng hàng đầu hiện nay dự báo: Đến năm 2020, sẽ có khoảng 50 tỷ đồ vật kết nối vào Internet, thậm chí con số này còn gia tăng nhiều hơn nữa. IoT sẽ là mạng khổng lồ kết nối tất cả mọi thứ, bao gồm cả con người và sẽ tồn tại các mối quan hệ giữa người và người, người và thiết bị, thiết bị và thiết bị. 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.

**Đặc tính cơ bản**  
– Tính kết nối liên thông(interconnectivity): với IoT, bất cứ điều gì cũng có thể kết nối với nhau thông qua mạng lưới thông tin và cơ sở hạ tầng liên lạc tổng thể.

– Những dịch vụ liên quan đến “Things”: hệ thống IoT có khả năng cung cấp các dịch vụ liên quan đến “Things”, chẳng hạn như bảo vệ sự riêng tư và nhất quán giữa Physical Thing và Virtual Thing. Để cung cấp được dịch vụ này, cả công nghệ phần cứng và công nghệ thông tin(phần mềm) sẽ phải thay đổi.

– Tính không đồng nhất: Các thiết bị trong IoT là không đồng nhất vì nó có phần cứng khác nhau, và network khác nhau. Các thiết bị giữa các network có thể tương tác với nhau nhờ vào sự liên kết của các network.

– Thay đổi linh hoạt: Status của các thiết bị tự động thay đổi, ví dụ, ngủ và thức dậy, kết nối hoặc bị ngắt, vị trí thiết bị đã thay đổi,và tốc độ đã thay đổi… Hơn nữa, số lượng thiết bị có thể tự động thay đổi.

Hình 2: Thiết bị kết nối

– Quy mô lớn: Sẽ có một số lượng rất lớn các thiết bị được quản lý và giao tiếp với nhau. Số lượng này lớn hơn nhiều so với số lượng máy tính kết nối Internet hiện nay. Số lượng các thông tin được truyền bởi thiết bị sẽ lớn hơn nhiều so với được truyền bởi con người.

* 1. **Arduino là gì?**

**Arduino** là một board mạch vi xử lý, nhằm xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. Phần cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý AVR Atmel 8bit, hoặc ARM Atmel 32-bit. Những Model hiện tại được trang bị gồm 1 cổng giao tiếp USB, 6 chân đầu vào analog, 14 chân I/O kỹ thuật số tương thích với nhiều board mở rộng khác nhau.

Được giới thiệu vào năm 2005, Những nhà thiết kế của Arduino cố gắng mang đến một phương thức dễ dàng, không tốn kém cho những người yêu thích, sinh viên và giới chuyên nghiệp để tạo ra những thiết bị có khả năng tương tác với môi trường thông qua các cảm biến và các cơ cấu chấp hành. Những ví dụ phổ biến cho những người yêu thích mới bắt đầu bao gồm các robot đơn giản, điều khiển nhiệt độ và phát hiện chuyển động. Đi cùng với nó là một môi trường phát triển tích hợp (IDE) chạy trên các máy tính cá nhân thông thường và cho phép người dùng viết các chương trình cho Aduino bằng ngôn ngữ C hoặc C++.

WEMOS D1 R2 là kit phát triển phiên bản mới nhất từ WeMos, kit được thiết kế với hình dáng tương tự Arduino Uno nhưng trung tâm lại là module wifi Soc ESP8266EX được build lại firmware để có thể chạy với chương trình Arduino.  Kit thích hợp và dễ dàng thực hiện các ứng dụng thu thập dữ liệu và điều khiển qua Wifi.

Một số thông số kỹ thuật như sau :

|  |  |
| --- | --- |
| **Vi điều khiển** | **ESP8266EX** |
| Điện áp hoạt động | 3V3 |
| I/O Digital Pin | 11 |
| Analog Pin | 1 (Max input=3V2) |
| Xung clock | 80MHz/160MHz |
| Flash | 4Mb |
| Khối lượng | 25g |
| Kích thước | 68.6mmX53.4mm |

**Sơ đồ pin digital**

|  |  |
| --- | --- |
| **Wemos** | **ESP8266** |
| D0 | GPIO16 |
| D1 | GPIO5 |
| D2 | GPIO4 |
| D3,D15 | GPIO0 |
| D4,D14 | GPIO2 |
| D5,D13 | GPIO14 |
| D6,D12 | GPIO12 |
| D7,D11 | GPIO13 |
| D8 | GPIO15 |
| D9 | GPIO3 |
| D10 | GPIO1 |

## Sơ đồ chân của WEMOS D1:

Hình 4: Arduino WeMos D1 R2

1. USB (1):

Arduino sử dụng cáp USB để giao tiếp với máy tính. Thông qua cáp USB chúng ta có thể Upload chương trình cho Arduino hoạt động, ngoài ra USB còn là nguồn cho Arduino.

1. Nguồn ( 2 và 3 ):

Khi không sử dụng USB làm nguồn thì chúng ta có thể sử dụng nguồn ngoài thông qua jack cắm 2.1mm ( cực dương ở giửa ) hoặc có thể sử dụng 2 chân Vin và GND để cấp nguồn cho Arduino.

Bo mạch hoạt động với nguồn ngoài ở điện áp từ 5 – 20 volt. Chúng ta có thể cấp một áp lớn hơn tuy nhiên chân 5V sẽ có mực điện áp lớn hơn 5 volt. Và nếu sử dụng nguồn lớn hơn 12 volt thì sẽ có hiện tượng nóng và làm hỏng bo mạch. Khuyết cáo các bạn nên dùng nguồn ổn định là 5 đến dưới 12 volt.

Chân 5V và chân 3.3V (Output voltage) : các chân này dùng để lấy nguồn ra từ nguồn mà chúng ta đã cung cấp cho Arduino. Lưu ý : không được cấp nguồn vào các chân này vì sẽ làm hỏng Arduino.

1. Chip Wifi SoC ESP8266.

Kit RF thu phát Wifi ESP8266 NodeMCU Lua D1 Mini có thiết kế nhỏ gọn, tích hợp sẵn mạch nạp chương trình và giao tiếp UART CH340, thường 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..

1. Input và Output ( 4, 5 và 6).

Arduino Wemos D1 Wifi SoC ESP8266 có 14 chân digital với chức năng input và output sử dụng các hàm *pinMode(), digitalWrite()* để điều khiển các chân này tôi sẽ đề cập chúng ở các phần sau.

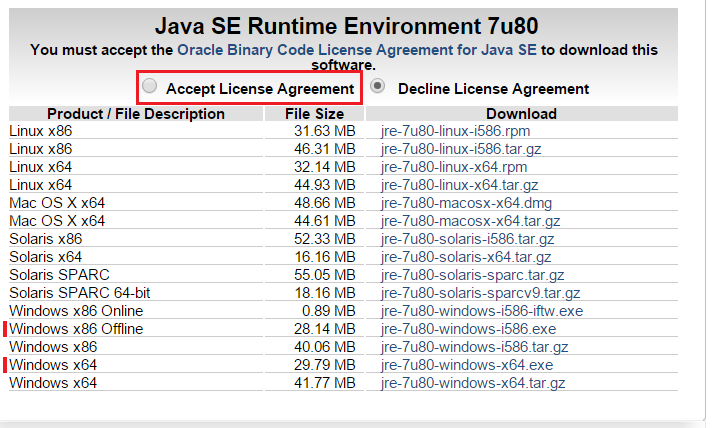
Cũng trên 14 chân digital này chúng ta còn một số chân chức năng đó là:

1. Reset (7): dùng để reset Arduino.

# Chương 2: CÀI ĐẶT CHƯƠNG TRÌNH ARDUINO IDE VÀ DRIVER CHO ARDUINO

## ****1. Cài đặt Java Runtime Environment (JRE):****

Vì **Arduino IDE** được viết trên **Java**nên bạn cần phải cài đặt **JRE**trước**Arduino IDE.**

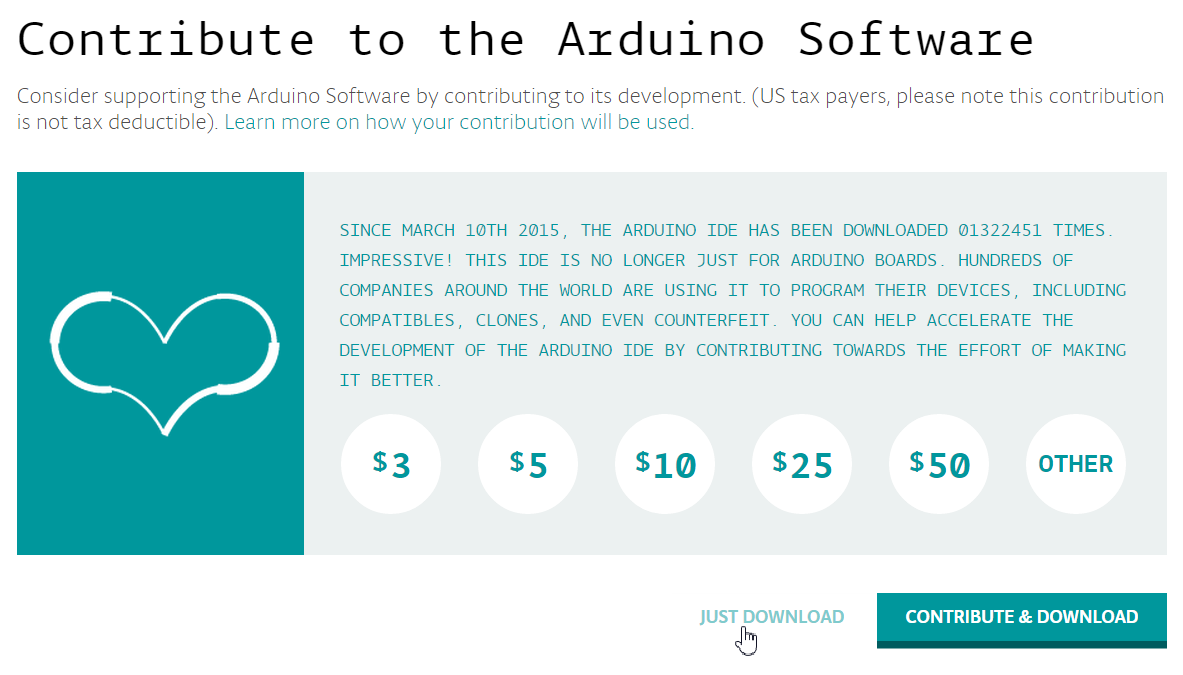
[](http://www.oracle.com/technetwork/java/javase/downloads/jre7-downloads-1880261.html)Link tải: http://www.oracle.com/technetwork/ja 2 bản **JRE**phổ biến nhất là bản dành cho Windows 32bit (x86) và Windows 64bit (x64) mình đã đánh dấu trong hình. Nhớ chọn "***Accept License Agreement***".

Hình 5: Bản JRE mới nhất là 7u80

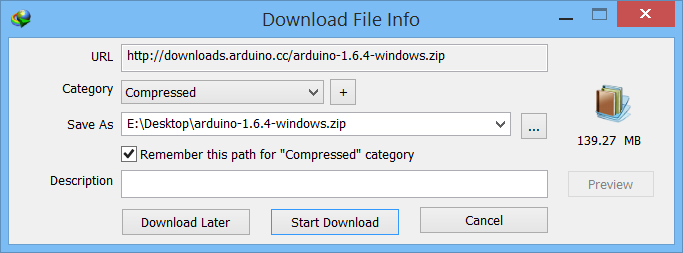
## ****2. Cài đặt Arduino IDE****:

[](http://k3.arduino.vn/img/2015/05/12/0/1338_81220-1431420080-0-2015-05-12-21h45-54.png)**Bước 1**: Truy cập địa chỉ [http://arduino.cc/en/Main/Software/...](http://arduino.cc/en/Main/Software/) . Đây là nơi lưu trữ cũng như cập nhật các bản IDE của Arduino. Bấm vào mục [**Windows ZIP**](http://arduino.cc/download_handler.php?f=/arduino-1.6.4-windows.zip)[file for non admin install](http://arduino.cc/download_handler.php?f=/arduino-1.6.4-windows.zip) như hình minh họa.

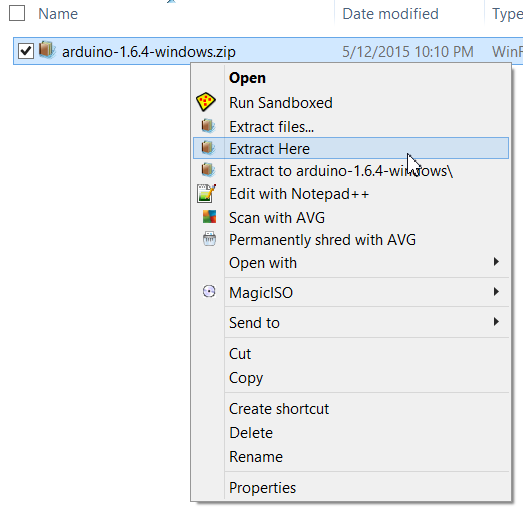
Hình 6: Tải xuống Arduino IDE

[](http://k1.arduino.vn/img/2015/05/12/0/1394_12320-1431420084-0-2015-05-12-21h46-45.png)Bạn sẽ được chuyển đến một trang mời quyền góp tiền để phát triển phần mềm cho Arduino, tiếp tục bấm [**JUST DOWNLOAD**](http://downloads.arduino.cc/arduino-1.6.4-windows.zip)để bắt đầu tải.

Hình 7: Đóng góp cho phần mềm Arduino

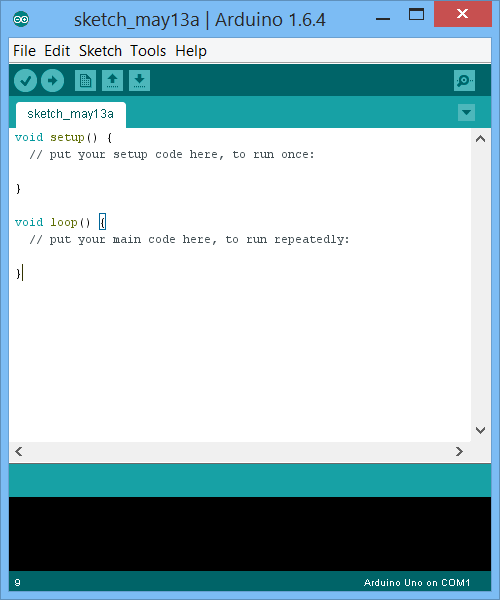
[](http://k2.arduino.vn/img/2015/05/12/0/1359_88220-1431420086-0-2015-05-12-21h49-09.png)

Hình 8: Tải phần mềm Arduino

[](http://k2.arduino.vn/img/2015/05/13/0/1364_88220-1431517904-0-2015-05-13-18h50-56.png)**Bước 2:** Sau khi download xong, các bạn bấm chuột phải vào file vừa download **arduino-1.6.4-windows.zip**và chọn “**Extract here**” để giải nén.

Hình 9: Giải nén file vừa tải

**Bước 3:** Copy thư mục **arduino-1.6.4** vừa giải nén đến nơi lưu trữ.

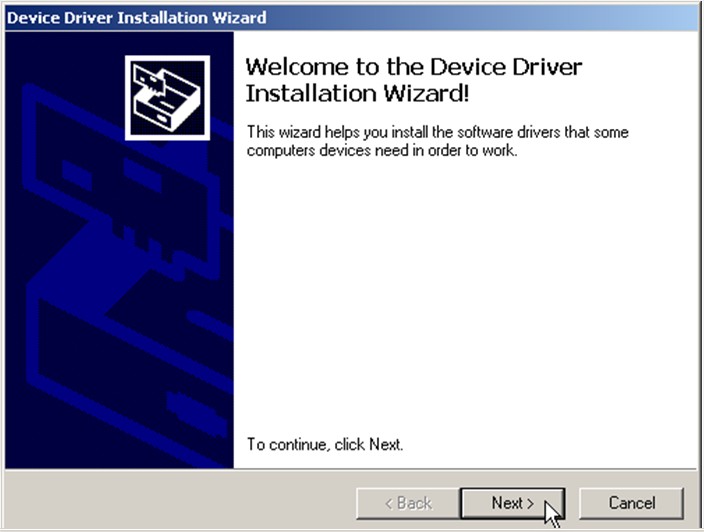
[](http://k1.arduino.vn/img/2015/05/13/0/1398_12320-1431518163-0-2015-05-13-18h55-51.png)**Bước 4:** Chạy file [http://k3.arduino.vn/img/2015/05/13/0/1342_81220-1431518023-0-2015-05-13-18h53-25.png](http://k3.arduino.vn/img/2015/05/13/0/1342_81220-1431518023-0-2015-05-13-18h53-25.png) trong thư mục **arduino-1.6.4\**để khởi động **Arduino IDE**

Hình 10: Giao diện phần mềm Arduino

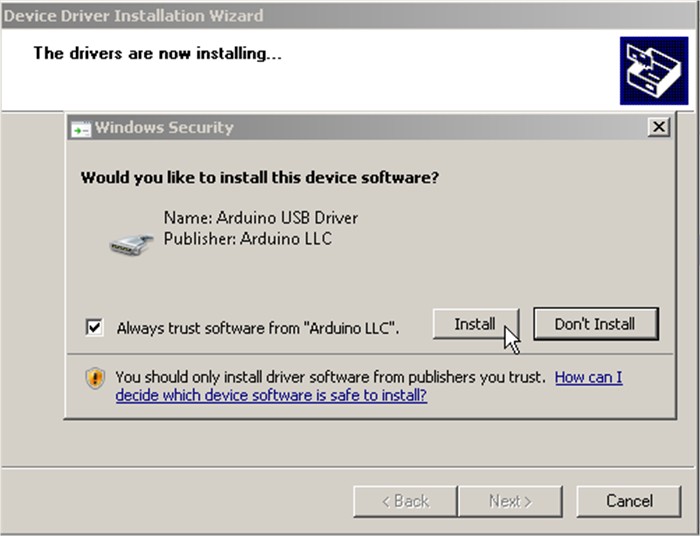
## ****3. Cài đặt Driver:****

Để máy tính của bạn và board Arduino giao tiếp được với nhau, chúng ta cần phải cài đặt driver trước tiên.

Nếu bạn dùng Windows 8, trong một số trường hợp Windows **không cho phép** bạn cài Arduino driver (do driver không được kí bằng chữ kí số hợp lệ). Do vậy bạn cần vào Windows ở chế độ **Disable driver signature enforcement**thì mới cài được driver

[](http://k1.arduino.vn/img/2014/05/24/0/495_1231-1400927587-0--ne.jpg)Đầu tiên, các bạn chạy file **arduino-1.6.4\drivers\dpinst-x86.exe**(Windows x86) hoặc **arduino-1.6.4\drivers\dpinst-amd64.exe** (Windows x64). Cửa sổ “**Device Driver Installation Wizard**” hiện ra, các bạn chọn **Next** để tiếp tục.

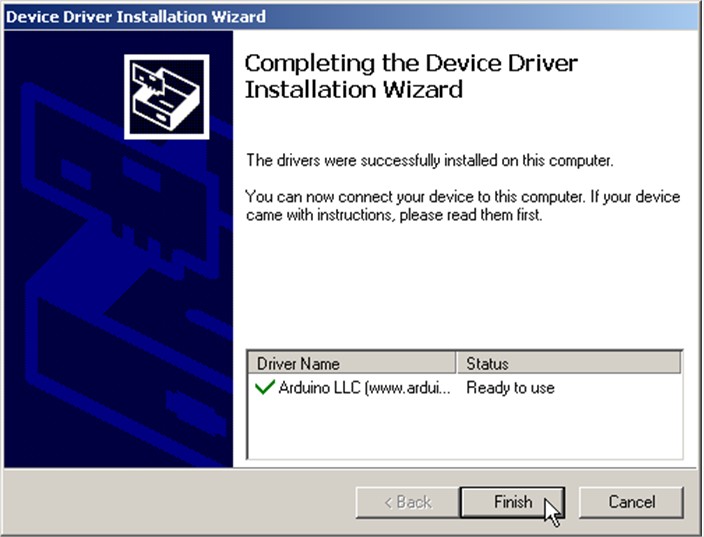
Hình 11: Device Driver Installation Wizard

[](http://k2.arduino.vn/img/2014/05/24/0/463_8821-1400927653-0--ins.jpg)[](http://k3.arduino.vn/img/2014/05/24/0/464_8121-1400927732-0--ins2.jpg)Khi có yêu cầu xác nhận cài đặt driver, chọn “**Install**”

Hình 12: Bắt đầu cài đặt driver

Đợi khoảng 10 giây trong lúc quá trình cài đặt diễn ra …

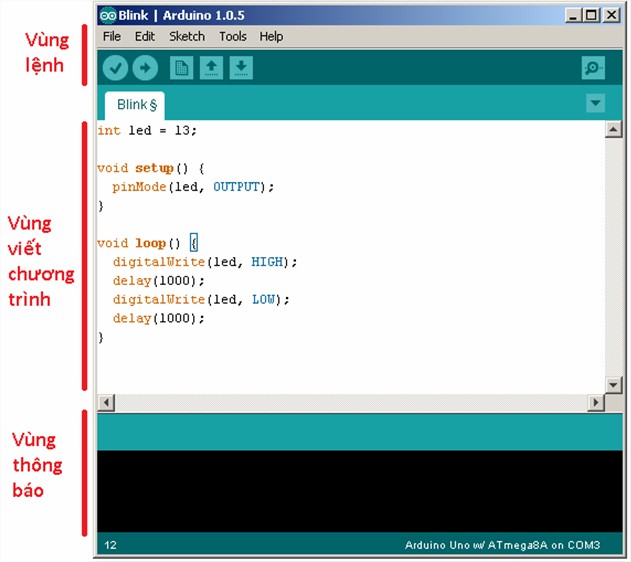
Hình 13: Quá trình cài đặt driver

[](http://k1.arduino.vn/img/2014/05/24/0/496_1231-1400927775-0--ins3.jpg)Quá trình cài đặt đã hoàn tất. Bấm “Finish” để thoát.

Hình 14: Hoàn tất quá trình cài đặt driver

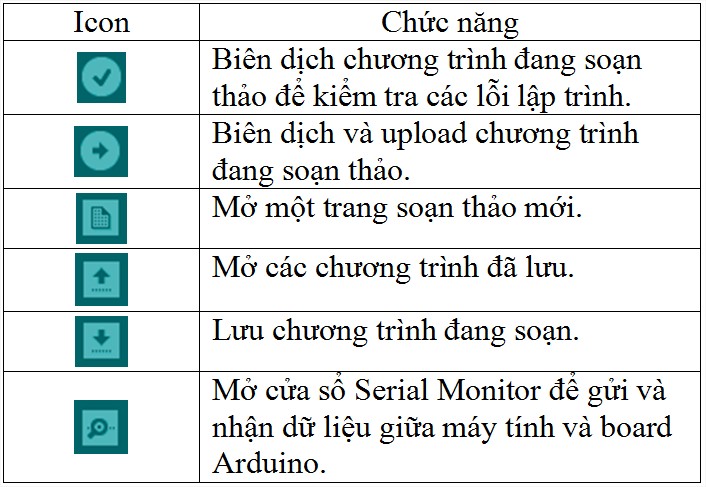
## ****4. Dạo một vòng Arduino IDE:****

### **4.1 Giao diện:**

[](http://k2.arduino.vn/img/2014/05/24/0/464_8821-1400927858-0--a4.jpg)**4.2 Vùng lệnh:**

Hình 15: Phân tích giao diện Arduino IDE

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:

[](http://k3.arduino.vn/img/2014/05/24/0/465_8121-1400927938-0--table.jpg)**4.3 Vùng viết chương trình:**

Hình 16: Các công cụ và chức năng của các ion

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.

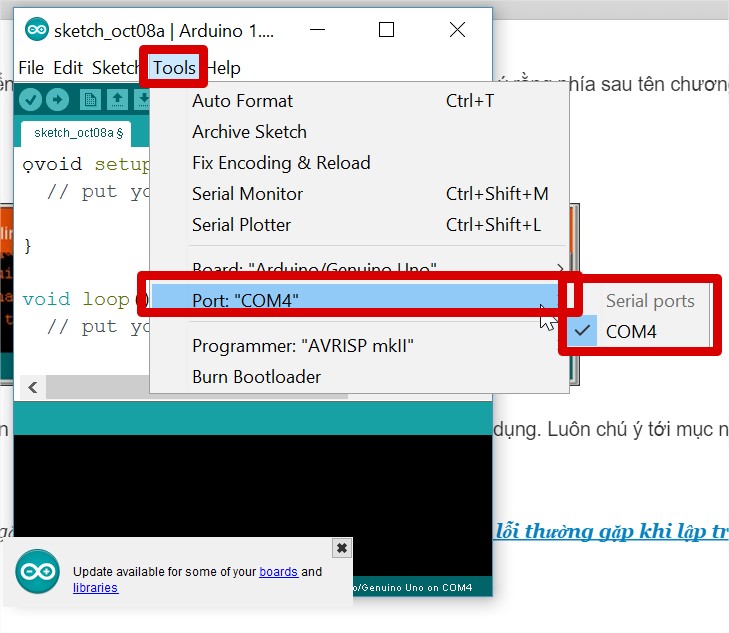
### **4.4 Vùng thông báo (debug):**

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.

### <http://k1.arduino.vn/img/2014/05/24/0/497_1231-1400928031-0--deb.jpg>**4.5 Một số lưu ý:**

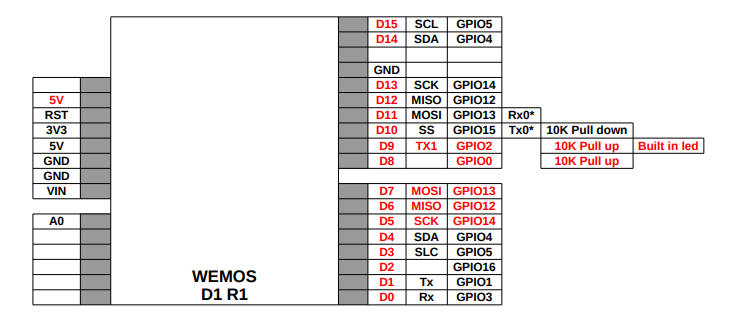
Hình 17: Một số lưu ý

Hình 18: Vùng thông báo

[](http://k2.arduino.vn/img/2016/10/08/0/3204_882450-1475933079-0-sketch-oct08a-arduino-1.6.7.jpg)Khi lập trình, các bạn cần chọn port (cổng kết nối khi gắn board vào) và board (tên board mà bạn sử dụng). Giả sử, bạn đang dùng mạch Arduino Uno, và khi gắn board này vào máy tính bằng cáp USB nó được nhận là COM4 thì bạn chỉnh như thế này là có thể lập trình đươc nhé.

# Chương 3: ĐIỀU KHIỂN BÓNG ĐÈN LED THÔNG QUA WIFI

## Lập trình mạch :

* 1. Sơ đồ mạch:

Hình 19: Sơ đồ mạch Wemos D1

* 1. Code chương trình:

*#include <ESP8266WiFi.h>*

*#include <WiFiClient.h>*

*#include <ESP8266WebServer.h>*

*#include <ESP8266mDNS.h>*

*MDNSResponder mdns;*

*ESP8266WebServer server(80);*

*String webPage;*

*const char\* ssid = "****minh****"; //Thay tên wifi ở dây*

*const char\* password = "****minh1234567****";//Thay tên password ở đây*

*void setup() {*

*pinMode(2, OUTPUT);*

*pinMode(14, OUTPUT);*

*pinMode(13, OUTPUT);*

*pinMode(12, OUTPUT);*

*webPage += "<meta charshet=utf-8 /><h1>ÐIỀU KHIỂN ÐÈN LED</h1><p>Ðèn 1 <a href=\"socket1On\"><button>ON</button></a>&nbsp;<a href=\"socket1Off\"><button>OFF</button></a></p>";*

*webPage += "<meta charshet=utf-8 /><p>Ðèn 2 <a href=\"socket2On\"><button>ON</button></a>&nbsp;<a href=\"socket2Off\"><button>OFF</button></a></p>";*

*webPage += "<meta charshet=utf-8 /><p>Ðèn 3 <a href=\"socket3On\"><button>ON</button></a>&nbsp;<a href=\"socket3Off\"><button>OFF</button></a></p>";*

*webPage += "<meta charshet=utf-8 /><p><h1>LED 4</h1> <a href=\"socket4On\"><button>ON</button></a>&nbsp;<a href=\"socket4Off\"><button>OFF</button></a></p>";*

*Serial.begin(115200);*

*delay(100);*

*Serial.println();*

*Serial.println();*

*Serial.print("Connecting to ");*

*Serial.println(ssid);*

*WiFi.begin(ssid, password);*

*while (WiFi.status() != WL\_CONNECTED) {*

*delay(500);*

*Serial.print(".");*

*}*

*Serial.println("");*

*Serial.println("WiFi connected");*

*Serial.println("IP address: ");*

*Serial.println(WiFi.localIP());*

*if (mdns.begin("esp8266", WiFi.localIP()))*

*Serial.println("MDNS responder started");*

*server.on("/", [](){*

*server.send(200, "text/html", webPage);*

*});*

*server.on("/socket1On", [](){*

*server.send(200, "text/html", webPage);*

*//Bậtt led 2*

*digitalWrite(2, HIGH);*

*delay(1000);*

*digitalWrite(2, LOW);*

*delay(1000);*

*});*

*server.on("/socket1Off", [](){*

*server.send(200, "text/html", webPage);*

*//Tắt led 2*

*digitalWrite(2, LOW);*

*delay(1000);*

*});*

*server.on("/socket2On", [](){*

*server.send(200, "text/html", webPage);*

*//Bật led 14*

*digitalWrite(14, HIGH);*

*delay(1000);*

*});*

*server.on("/socket2Off", [](){*

*server.send(200, "text/html", webPage);*

*//Tắt led 14*

*digitalWrite(14, LOW);*

*delay(1000);*

*});*

*server.on("/socket3On", [](){*

*server.send(200, "text/html", webPage);*

*//Bật led 13*

*digitalWrite(13, HIGH);*

*delay(1000);*

*});*

*server.on("/socket3Off", [](){*

*server.send(200, "text/html", webPage);*

*//Tắt led 13*

*digitalWrite(13, LOW);*

*delay(1000);*

*});*

*server.on("/socket4On", [](){*

*server.send(200, "text/html", webPage);*

*//Bật led 12*

*digitalWrite(12, HIGH);*

*delay(1000);*

*});*

*server.on("/socket4Off", [](){*

*server.send(200, "text/html", webPage);*

*//Tắt led 12*

*digitalWrite(12, LOW);*

*delay(1000);*

*});*

*server.begin();*

*Serial.println("HTTP server started");*

*}*

*void loop() {*

*server.handleClient();*

*digitalWrite(2, HIGH);*

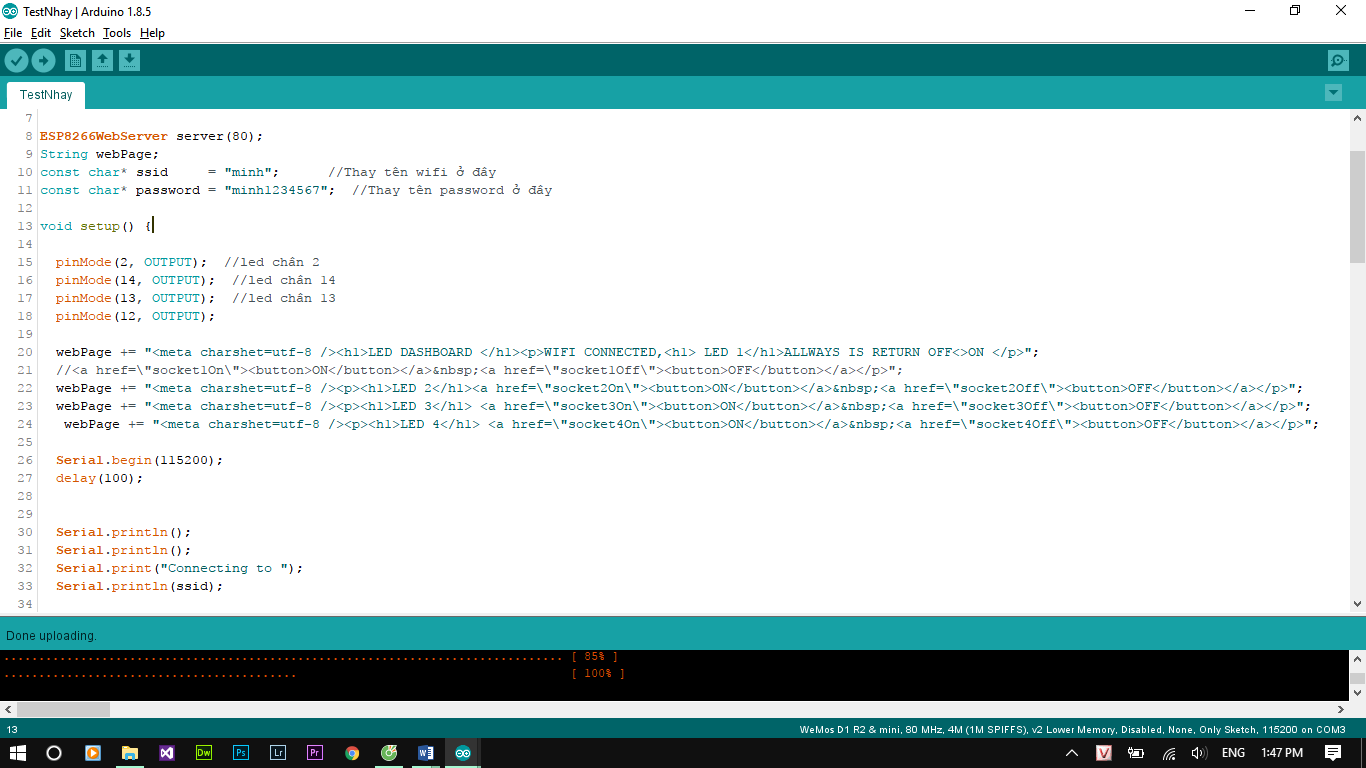
*delay(500);*

*digitalWrite(2, LOW);*

*delay(500);*

*}*

Sau khi gõ code vào chương trình soạn thảo bạn cần click và để kiểm tra lỗi



1. Giải thích chương trình.
2. #include <ESP8266WiFi.h>
3. #include <WiFiClient.h>
4. #include <ESP8266WebServer.h>
5. #include <ESP8266mDNS.h>

Khai báo các thư viện có sẵn trong chương trình.

*void setup() {*

*pinMode(2, OUTPUT); //led chân 2*

*pinMode(14, OUTPUT); //led chân 14*

*pinMode(13, OUTPUT); //led chân 13*

*pinMode(13, OUTPUT); //led chân 12*

Khai báo các chân led dùng, ví dụ chúng ta dùng chân 2, 14, 13,12 thì khai theo cú pháp pinMode(số chân, OUTPUT);

Trong Arduino Wemos D1 sketch cần phải có hàm setup() và loop() nếu không có thì chương trình báo lỗi. Hàm Setup() chỉ chạy một lần kể từ khi bắt đầu chương trình. Hàm này có chức năng thiết lập chế độ vào, ra cho các chân digital hay tốc độ baud cho giao tiếp Serial, còn hàm loop() thì khai báo cho vòng lặp vô hạn ( sẽ được nói đến ở phần sau)..

Cấu trúc của hàm *pinMode()* là như sau:

*pinMode(vị trí chân,Mode);*

*Mode*: là chế độ vào ( INPUT), ra (OUTPUT). Lệnh tiếp theo.

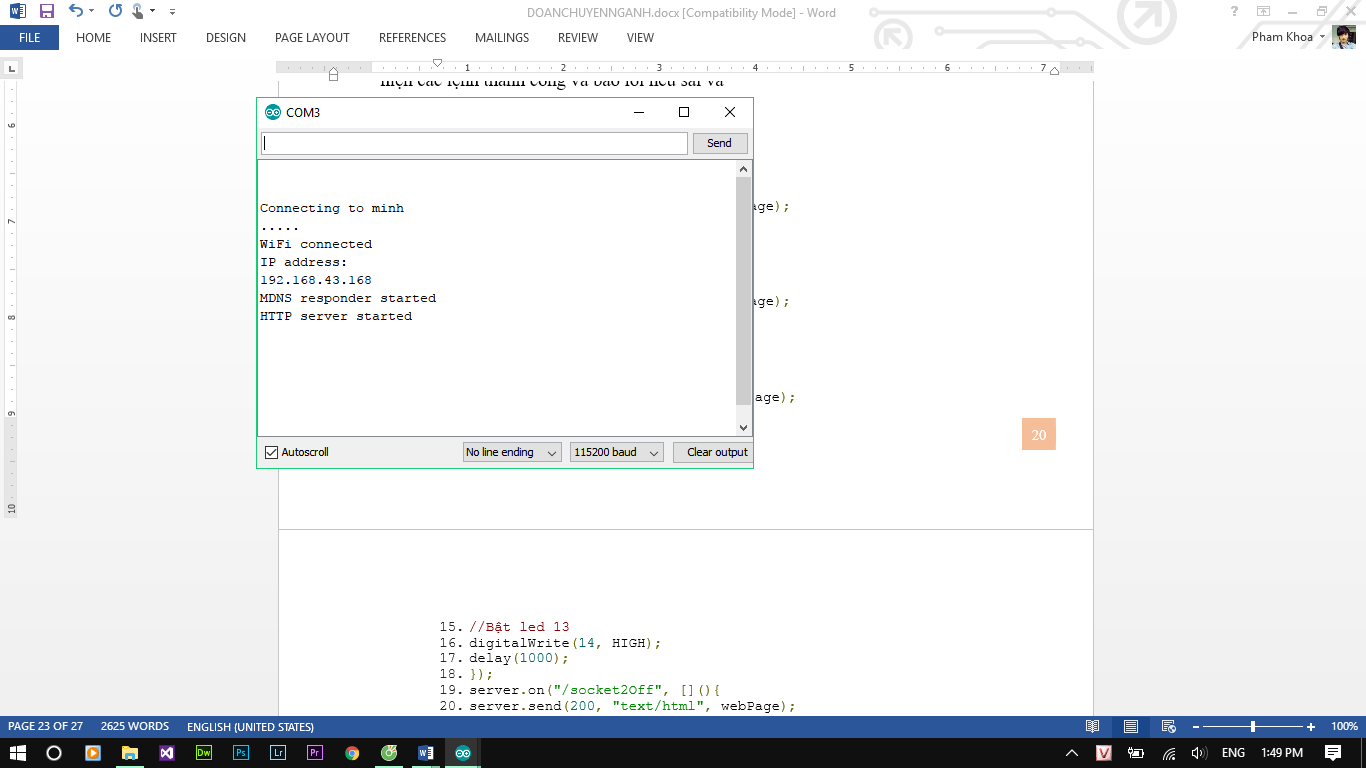
webPage += "<h1>ESP8266 Web Server</h1><p>Socket #1 <a href=\"socket1On\"><button>ON</button></a>&nbsp;<a href=\"socket1Off\"><button>OFF</button></a></p>";

webPage += "<p>Socket #2 <a href=\"socket2On\"><button>ON</button></a>&nbsp;<a href=\"socket2Off\"><button>OFF</button></a></p>";

Lệnh này để lập trình tạo ra giao diện trang web khi chúng ta điều khiển trên môi trường HTML.

1. Serial.begin(115200);
2. delay(100);
3. Serial.println();
4. Serial.println();
5. Serial.print("Connecting to ");
6. Serial.println(ssid);
7. WiFi.begin(ssid, password);
8. while (WiFi.status() != WL\_CONNECTED) {
9. delay(500);
10. Serial.print(".");
11. }
12. if (mdns.begin("esp8266", WiFi.localIP()))
13. Serial.println("MDNS responder started");
15. server.on("/", [](){
16. server.send(200, "text/html", webPage);
17. });

Lệnh code trên có nghĩa xuất ra thông báo như bên dưới của Aduino IDE khi thực hiện các lệnh thành công và báo lỗi nếu sai thì chương trình không xuất hiện các thông báo kết nói wifi.



Lệnh tiếp theo

1. server.on("/socket1On", [](){
2. server.send(200, "text/html", webPage);
3. //Bật led 2
4. digitalWrite(2, HIGH);
5. delay(1000);
6. });
7. server.on("/socket1Off", [](){
8. server.send(200, "text/html", webPage);
9. //Tắt led 14
10. digitalWrite(14, LOW);
11. delay(1000);
12. });
13. server.on("/socket2On", [](){
14. server.send(200, "text/html", webPage);
15. //Bật led 13
16. digitalWrite(14, HIGH);
17. delay(1000);
18. });
19. server.on("/socket2Off", [](){
20. server.send(200, "text/html", webPage);
21. //Tắt led 13
22. digitalWrite(13, LOW);
23. delay(1000);
24. });
25. server.begin();
26. Serial.println("HTTP server started");
27. }

Các lệnh ở trên để khai báo bật tắt các chân thiết bị Off là tắt thì khai báo LOW và ngược lại với On thì dung HIGH. Trong lập trình trên Arduino, ***HIGH***là một hằng số có giá trị nguyên là 1. Trong điện tử, ***HIGH***là một mức điện áp lớn hơn 0V. G*iá trị của****HIGH****được định nghĩa khác nhau trong các mạch điện khác nhau, nhưng thường được quy ước ở các mức như 1.8V, 2.7V, 3.3V 5V, 12V, …*

***LOW***là một hằng số có giá trị nguyên là 0. Trong điện tử, ***LOW***là mức điện áp 0V hoặc gần bằng 0V, giá trị này được định nghĩa khác nhau trong các mạch điện khác nhau, nhưng thường là 0V hoặc hơn một chút xíu

server.send(200, "text/html", webPage);

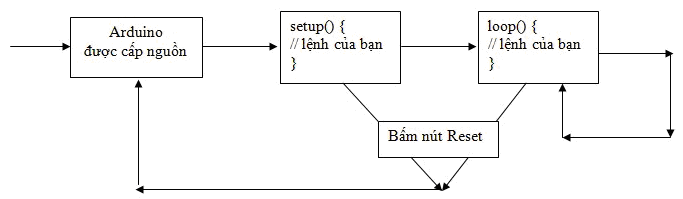
200 là khai báo số ký tự có trong trang webpage mà mình giới hạn khi lập trình web.

Lệnh tiếp theo:

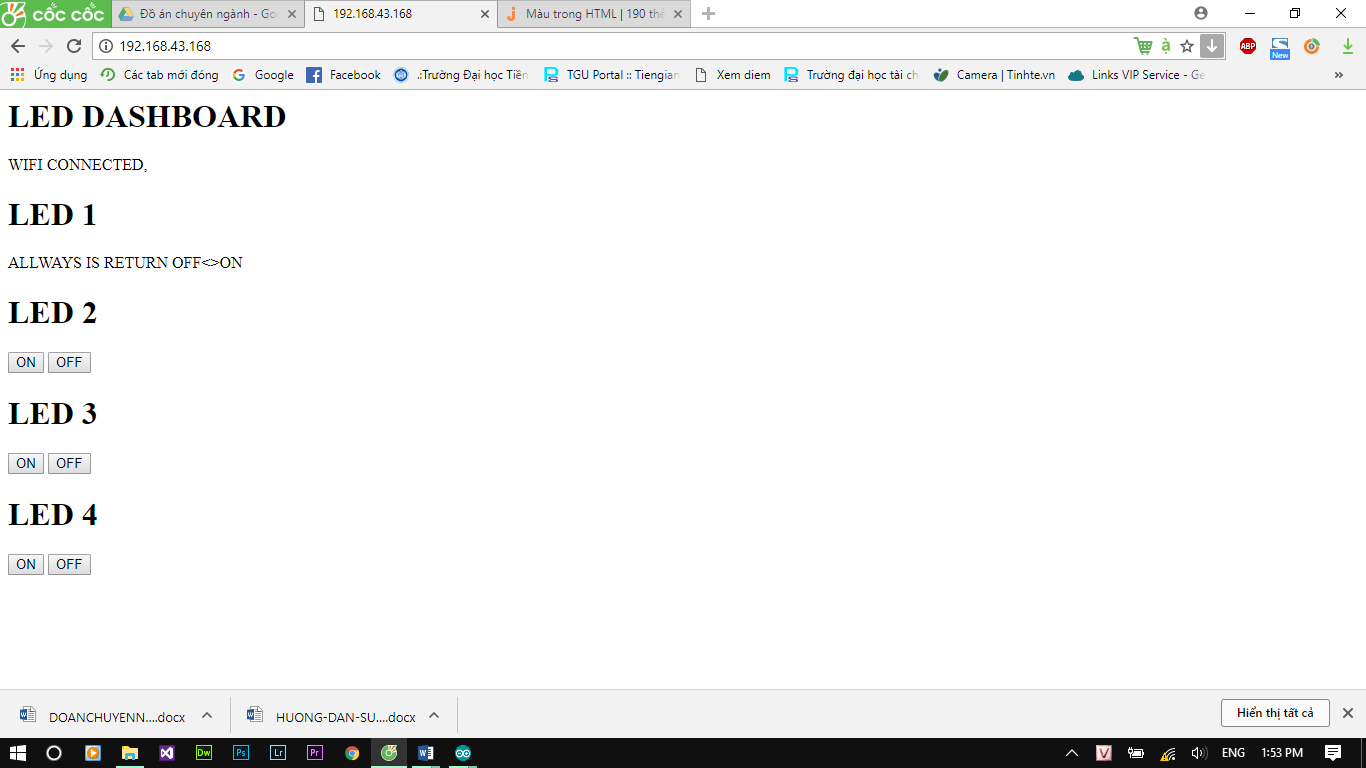
1. *void loop() {*
2. *server.handleClient();*
3. *digitalWrite(2, HIGH);*
4. *delay(500);*
5. *digitalWrite(2, LOW);*
6. *delay(500);*
7. *}*

Bắt buộc khai báo một hàm loop() trong Arduino IDE. Hàm này là vòng lặp vô hạn để bóng đèn chớp tắt vô hạn , và chân đèn số 2 sẽ chớp tắt liên tục.

Chúng ta có thể hiểu hàm loop và setup hoạt động theo trình tự như hình bên dưới



Khi đã hoàn tất việc lập trình thì chúng ta nhấn nút  để nạp vào Aduino Wemos

và kết quả khi chạy hoàn tất

Sau khi hoàn thành phần lập trình cho aduino thì chúng ta bắt đầu thiết kế mạch, dây dẫn, bóng đèn..v.v đảm bảo về mặt thẩm mỹ và chất liệu an toàn.

Bên dưới là mô hình dự kiến của nhóm chúng em



Trên đây là dự kiến sản phẩm của nhóm chúng em, trong thời gian sắp tới nhóm có dự định phát triển Aduino để làm đèn trong ngôi nhà, và điều khiển những vật dụng trong gia đình.

## Tài liệu tham khảo.

* + - Beginning Arduino - Mike McRoberts
    - Arduino cookbook – Michael Margolis
    - [http://arduino.cc](http://arduino.cc/)
    - [http://blogembarcado.blogspot.com](http://blogembarcado.blogspot.com/)

## 