**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA ĐIỆN – ĐIỆN TỬ

**BỘ MÔN ĐIỆN TỬ**

---------------o0o---------------



**ĐỒ ÁN MÔN HỌC**

**TIÊU ĐỀ ĐỒ ÁN MÔN HỌC**

**ĐIỀU KHIỂN LED BẰNG GIỌNG NÓI ( CONTROL LED USING YOUR VOICE)**

**GVHD: Võ Tấn Thông**

**SVTH: Hồ Việt Anh**

**MSSV: 1912565**

MỤC LỤC

[CHƯƠNG 1: GIỚI THIỆU 2](#_Toc114254789)

[Giới thiệu về Inter of Things (IoT): 2](#_Toc114254790)

[CHƯƠNG 2: LÝ THUYẾT 3](#_Toc114254791)

[1. Sơ đồ khối 4](#_Toc114254792)

[2. Tổng quan về vi điều khiển ATMega329P 4](#_Toc114254793)

[CHƯƠNG 3: THIẾT KẾ VÀ THỰC HIỆN PHẦN CỨNG 6](#_Toc114254794)

[1. Phần cứng: 6](#_Toc114254795)

[2. Sơ đồ mạch: 13](#_Toc114254796)

[3.Mạch khi lắp ráp: 14](#_Toc114254797)

[CHƯƠNG 4: THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM 15](#_Toc114254798)

[1. Tổng quan về Arduino IE 15](#_Toc114254799)

[2. Thiết kế và thực hiện: 23](#_Toc114254800)

[2.2. Lập trình cho Arduino 25](#_Toc114254801)

[CHƯƠNG 5: KẾT QUẢ THỰC HIỆN 29](#_Toc114254802)

[CHƯƠNG 6: TÀI LIỆU THAM KHẢO 36](#_Toc114254803)

# CHƯƠNG 1: GIỚI THIỆU

## Giới thiệu về Inter of Things (IoT):

IoT là sự phát triển của các ứng dụng di động, gia đình và nhúng đang được kết nối với internet tích hợp khả năng tính toán cao hơn và sử dụng phân tích dữ liệu để trích xuất thông tin có ý nghĩa. Hàng tỷ thiết bị sẽ được kết nối với internet và sắp tới là hàng trăm tỷ thiết bị. Khi các thiết bị liên quan kết nối với nhau, chúng có thể trở thành một hệ thống thông minh của hệ thống. Khi các thiết bị và hệ thống thông minh này chia sẻ dữ liệu qua đám mây và phân tích nó, chúng có thể biến đổi hoạt động kinh doanh, cuộc sống và thế giới của chúng ta theo vô số cách.

Cho dù đó là cải thiện kết quả y tế, tạo ra sản phẩm tốt hơn nhanh hơn với chi phí phát triển thấp hơn, giúp mua sắm thú vị hơn hay tối ưu hóa việc tạo và tiêu thụ năng lượng.

Đây là một ví dụ về bức tranh lớn. Hãy tưởng tượng một thiết bị thông minh chẳng hạn như camera giao thông thông minh, camera có thể giám sát tình trạng tắc nghẽn, tai nạn và thời tiết trên đường và thông báo trạng thái đó đến một cổng kết hợp với dữ liệu từ các camera khác tạo ra một hệ thống giao thông thông minh trên toàn thành phố.

Hãy tưởng tượng hệ thống giao thông thông minh đó được kết nối với hệ thống giao thông khác trên toàn thành phố. Bây giờ hãy hình dung hệ thống giao thông thông minh được kết nối với các hệ thống giao thông khác trên toàn thành phố lấy dữ liệu từ các thiết bị thông minh của riêng chúng, tạo ra một hệ thống hệ thống thậm chí còn lớn hơn.

Giả sử hệ thống giao thông thông minh của thành phố phát hiện ra tắc nghẽn lớn do tai nạn. Thông tin chi tiết đó có thể được gửi đến hệ thống giao thông toàn thành phố, hệ thống này có thể phân tích tác động của vụ tai nạn đối với các hệ thống khác của thành phố. Nhận thấy vụ tai nạn ở gần sân bay và hai trường học của thành phố, nó có thể thông báo cho các hệ thống đó để họ có thể điều chỉnh lịch trình chuyến bay và trường học. Nó cũng có thể phân tích và tìm ra các tuyến đường tối ưu xung quanh vụ tai nạn và gửi những chỉ dẫn đó đến hệ thống biển báo kỹ thuật số của thành phố để hướng dẫn người lái xe xung quanh vụ tai nạn. Đó chỉ là một ví dụ về những lợi ích tiềm năng có thể xảy ra khi các thiết bị thông minh chia sẻ thông tin chi tiết với các hệ thống khác, tạo thành các hệ thống ngày càng mở rộng.

# CHƯƠNG 2: LÝ THUYẾT

Phần này chỉ giới thiệu những nét chung nhất của Vi điều khiển 8 bit ATMEGA328P.

## Sơ đồ khối

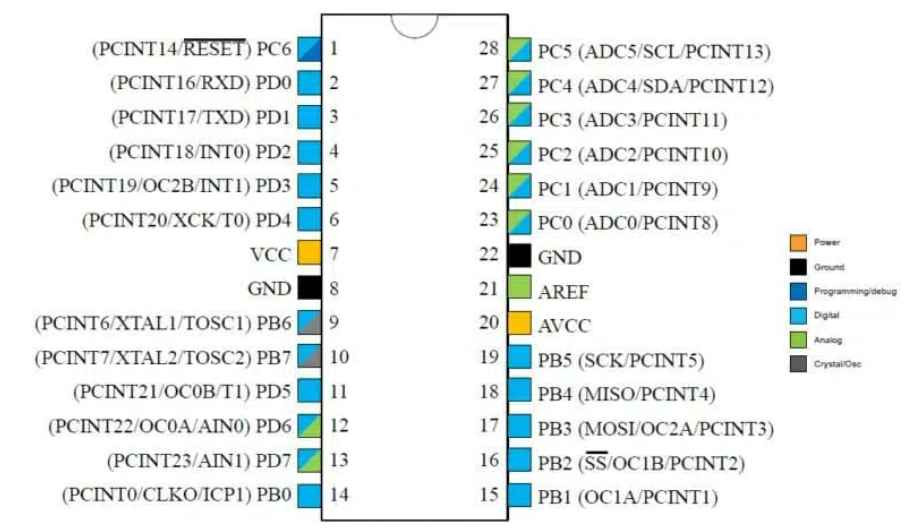


Đóng vai trò bộ điều khiển trung tâm, chịu trách nhiệm về tất cả các họat động của mạch, board mạch chủ với bộ xử lí trung tâm là ATMega328P cung cấp mộ công cụ có thể thực hiện điều khiển theo bất kì phương pháp nào, chỉ cần có giải thuật và chuyển đổi thành chương trình nạp vào bộ nhớ của ATMega328P.

## Tổng quan về vi điều khiển ATMega329P

ATmega328P là một trong những vi điều khiển công nghệ AVR hiệu suất cao với số lượng chân cắm và tính năng lớn. Được thiết kế bằng công nghệ CMOS 8-bit và CPU RSIC giúp nâng cao hiệu suất và tối ưu mức sử dụng năng lượng nhờ có chế độ ngủ tự động và cảm biến nhiệt độ bên trong. ATmega328P có mạch bảo vệ bên trong và có nhiều cách lập trình giúp các kỹ sư sử dụng linh hoạt ở các tính huống khác nhau. IC hỗ trợ nhiều giao thức giao tiếp hiện đại cho các module khác và chính bộ vi điều khiển, đó là lý do tại sao ATmega328P được sử dụng phổ biến.

* 1. Sơ đồ chân ATmega328P

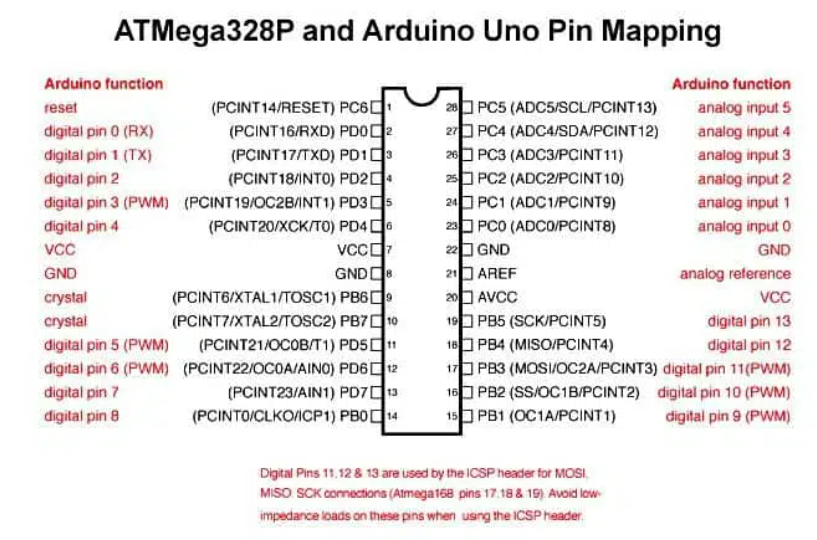
Trong phần này, chúng ta sẽ xem chi tiết cách sử dụng của từng chân và sơ đồ chân của vi điều khiển 28 chân này. Hình ảnh này cho thấy sơ đồ chân:

Các lựa chọn thay thế vi điều khiển atmel: ATtiny45, ATtiny88, ATtiny85.

* 1. **Sơ đồ chân của** **ATmega328P với Arduino:**

ATmega328 tương thích với các chân Arduino, giúp người dùng viết code chương trình bằng Arduino thay vì hợp ngữ (assembly) hoặc các ngôn ngữ điều khiển khác.

Arduino phổ biến vì có thông tin trên mạng khổng lồ và lập trình với ngôn ngữ cấp cao, và có thể giúp viết code chương trình điều khiển trong Arduino và chuyển đổi thành code của vi điều khiển ATmega328P. Trong trường hợp với Arduino, cấu hình chân cho bộ điều khiển sẽ như sau:

Trong trường hợp với Arduino, các chân đặc trưng với chức năng của chúng. Khi sử dụng trình biên dịch của ATmega328 hầu như tất cả các chân đều có thể được sử dụng làm GPIO.

Tuy nhiên, trong khi sử dụng trong Arduino, mỗi chân sẽ thực hiện một chức năng cụ thể, nhưng bộ điều khiển vẫn có thể thực hiện tất cả các chức năng giống như ATmega328.

# CHƯƠNG 3: THIẾT KẾ VÀ THỰC HIỆN PHẦN CỨNG

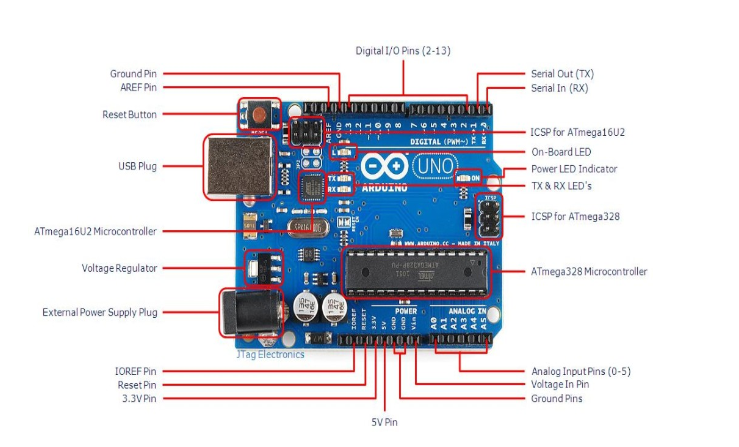
## Phần cứng:

* Arduino Uno R3
* Module Bluetooth HC-05 hoặc tương tự
* 3 Led 5mm
* 3 điện trở 1kΩ
* 1 điện thoại Android có kết nối mạng
  1. Arduino Uno R3:

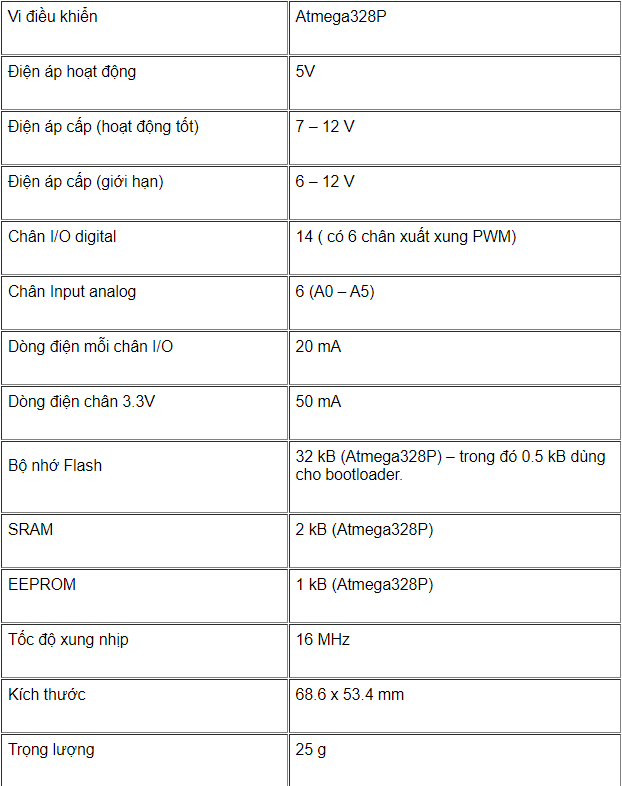
**Arduino Uno** là một bo mạch vi điều khiển dựa trên chip Atmega328P. Uno có 14 chân I/O digital ( trong đó có 6 chân xuất xung PWM), 6 chân Input analog, 1 thạch anh 16MHz, 1 cổng USB, 1 jack nguồn DC, 1 nút reset.

Uno hỗ trợ đầy đủ những thứ cần thiết để chúng ta có thể bắt đầu làm việc.

* + 1. Sơ đồ chân:



* + 1. Thông số kỹ thuật



* 1. Module Bluetooth HC-05:
     1. Tổng quan:

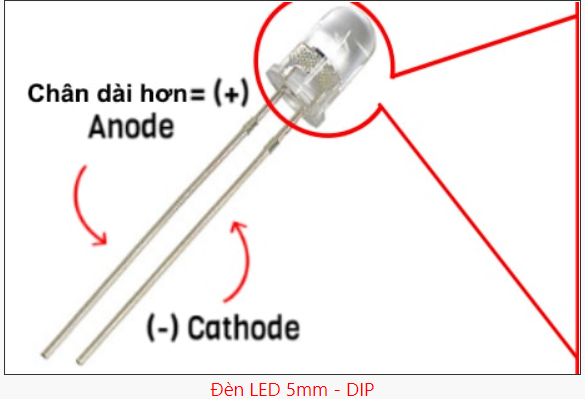
[**Module thu phát Bluetooth HC-05**](https://nshopvn.com/product/module-thu-phat-bluetooth-hc-05/) dùng để thiết lập kết nối Serial giữa 2 thiết bị bằng sóng bluetooth. Điểm đặc biệt của module bluetooth HC-05 là module có thể hoạt động được ở 2 chế độ: MASTER hoặc SLAVE. Trong khi đó, **[bluetooth module HC-06](https://nshopvn.com/product/module-thu-phat-bluetooth-hc-06/)** chỉ hoạt động ở chế độ SLAVE.

+ Ở chế độ SLAVE: bạn cần thiết lập kết nối từ smartphone, laptop, usb bluetooth để dò tìm module sau đó pair với mã PIN là 1234. Sau khi pair thành công, bạn đã có 1 cổng serial từ xa hoạt động ở baud rate 9600.

+ Ở chế độ MASTER: module sẽ tự động dò tìm thiết bị bluetooth khác (1 module bluetooth HC-06, usb bluetooth, bluetooth của laptop…) và tiến hành pair chủ động mà không cần thiết lập gì từ máy tính hoặc smartphone.

[**Module Bluetooth thu phát HC-05**](https://youtu.be/CWeKN8WNm3o) được thiết kế nhỏ gọn ra chân tín hiệu giao tiếp cơ bản và nút bấm để vào chế độ AT COMMAND, mạch được thiết kế để có thể cấp nguồn và giao tiếp qua 3.3VDC hoặc 5VDC, thích hợp cho nhiều ứng dụng khác nhau: Robot Bluetooth, điều khiển thiết bị qua Bluetooth,…

* + 1. Thông số kỹ thuật:
* Điện áp hoạt động: 3.3 ~ 5VDC
* Mức điện áp chân giao tiếp: TTL tương thích 3.3VDC và 5VDC.
* Dòng điện khi hoạt động: khi Pairing 30 mA, sau khi pairing hoạt động truyền nhận bình thường 8 mA.
* Baudrate UART có thể chọn được: 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
* Support profiles: Bluetooth serial port (master and slave)
* Bluetooth protocol: Bluetooth specification v2.0 + EDR
* Frequency: 2.4 GHz ISM band
* Modulation: GFSK (Gaussian frequency shift keying)
* Transmit power: =4 dBm, class 2
* Sensitivity: =-84 dBm at 0.1% BER
* Rate: Asynchronous: 2.1 Mbps (max.)/160 kbps
* Synchronous: 1 Mbps/1 Mbps
* Security features: authentication and encryption
* Kích thước: 15.2 x 35.7 x 5.6mm.
  + 1. Thiết lập mặc định:
* Thiết lập UART mặc định: Baudrate 9600, N, 8, 1.
* Pairing code mặc định: 1234 hoặc 0000.
* Để vào chế độ AT COMMAND, bấm và giữ nút trước khi cấp nguồn, LED sẽ nháy 2s. baud rate cho chế đọ AT COMMAND là 38400. Chân Tx nối với chân Rx. Lưu ý các lệnh AT đều là chữa in hoa.
* Cấp nguồn và không nhấn nút sẽ chạy bình thường. LED sẽ nháy nhanh
* Chân EN chỉ nhận mức logic TTL 3V3. Không có chức năng chọn vào chế độ AT COMMAND.
  1. Led:



* + 1. Tổng quan:

LED là một biến thể trên diode (điốt) cơ bản. Diode là một thành phần điện tử chỉ dẫn điện theo một hướng. Nó xác định độ chênh điện áp nhỏ nhất giữa Anode (+) và Cathode (-). LED là cơ bản giống như một Diode, sự khác biệt ở đây là nó tạo ra ánh sáng khi dòng điện đi qua.

* + 1. Điện áp:

Mỗi đèn LED có một mức điện áp đầu vào nhất định để tạo ra ánh sáng. Nếu bạn cung cấp ít hơn mức này, nó sẽ không phát sáng. Nếu bạn có nhiều đèn được nối tiếp trong một mạch, bạn phải tính mức điện áp cần cấp cho loạt này.  
  
Ví dụ, ta có một đèn LED 5mm có mức điện áp đầu vào là 3,4V. Chúng ta lấy cấp nguồn cho nó là một pin AA 1,5V nó sẽ không phát sáng. Tuy nhiên, nếu chúng ta thêm một pin AA nữa để cấp nguồn, tức là khoảng 3V thì đèn LED sẽ sáng. Trị số thập phân 0,4 có thể bỏ qua trong trường hợp này.

* + 1. Dòng điện:

Dòng điện cũng là trị số quan trọng hàng đầu cần quan tâm. Nhiệt độ là kẻ thù của công nghệ LED, nếu bạn cung cấp một dòng điện vượt qua ngưỡng cho phép, tương đương với việc làm gia tăng nhiệt độ và làm đèn nhanh chóng bị hỏng.  
  
Dòng điện phù hợp với đèn LED 5mm thường ở mức 20mA, tối đa có thể tới 30mA. Chúng ta có thể kiểm soát dòng điện bằng cách đặt một điện trở nối tiếp với đèn LED. Nó giúp dòng điện cấp cho đèn luôn ở mức cho phép.

* + 1. Ánh sáng:

Bước sóng là cách chính xác để giải thích màu sắc của ánh sáng. Đối với đèn LED, sẽ có sự khác biệt về màu sắc do quá trình sản xuất. Trên bảng đèn LED 5mm, bạn sẽ thấy bước sóng tối thiểu và tối đa. Các biến thể luôn trong cùng một dải, chỉ khi bạn mua các LED có cùng màu trong nhiều đợt khác nhau, có thể có một số biến thể nhỏ   
  
Bước sóng này thực sự được quyết định bởi loại vật liệu bán dẫn được sử dụng để tạo ra diode bên trong gói 5mm này. Cấu trúc dải năng lượng của chất bán dẫn thay đổi giữa các vật liệu, do đó photon phát ra các tần số khác nhau có ảnh hưởng đến ánh sáng chúng ta thấy.

* 1. Điện trở:



**Điện trở** hay **Resistor** là một [linh kiện điện tử thụ động](https://vi.wikipedia.org/wiki/Linh_ki%E1%BB%87n_%C4%91i%E1%BB%87n_t%E1%BB%AD_th%E1%BB%A5_%C4%91%E1%BB%99ng" \o "Linh kiện điện tử thụ động) gồm 2 tiếp điểm kết nối, thường được dùng để hạn chế cường độ dòng điện chảy trong mạch

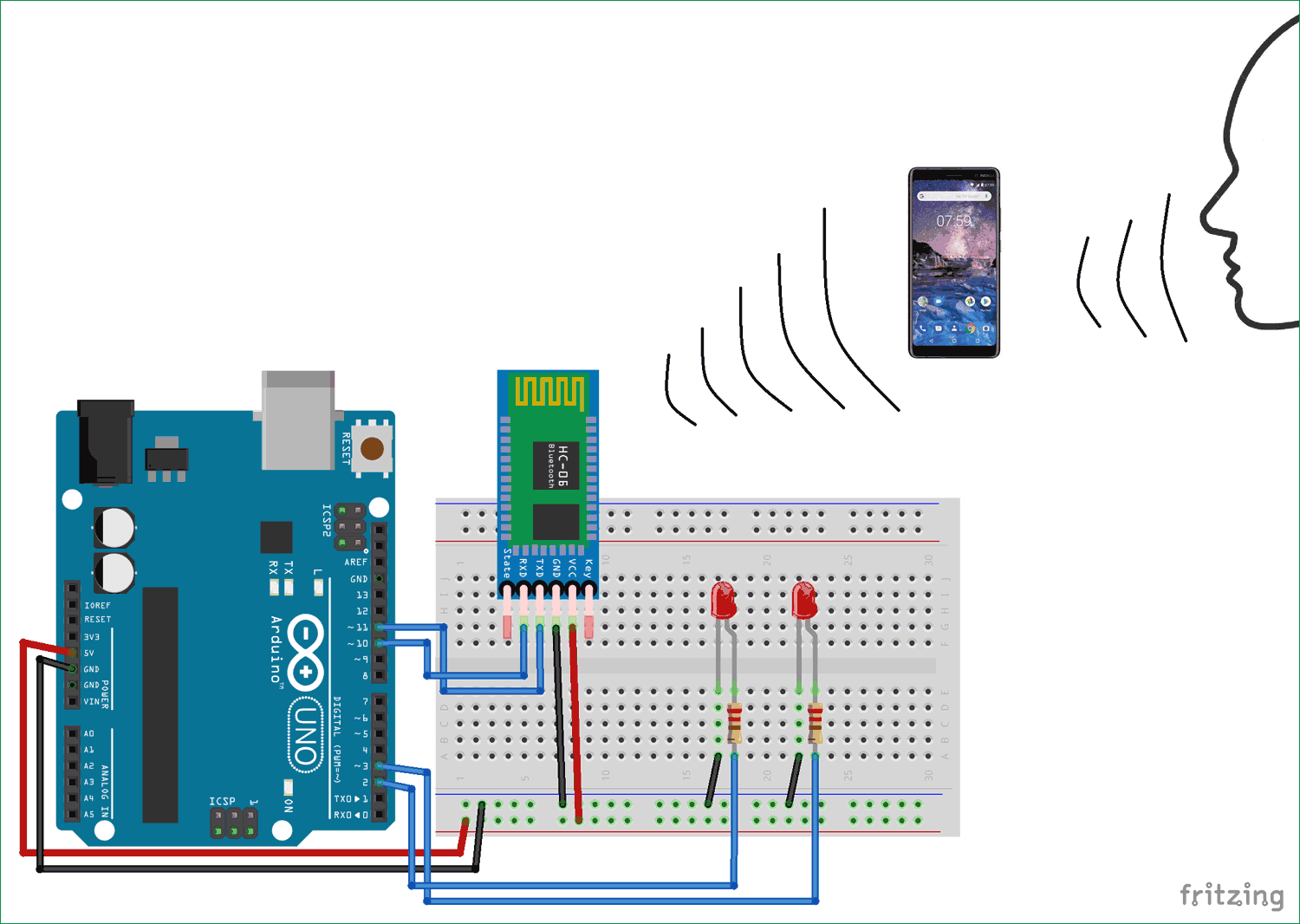
* + 1. Định luật Ohm:
* Theo định luật Ohm: điện áp (V) đi qua điện trở tỉ lệ thuận với cường độ dòng điện (I) và tỉ lệ này là 1 hằng số điện trở (R).
* Công thức của định luật Ohm như sau: V=I\*R

**Ví dụ**: Nếu một điện trở 400Ω được nối vào điện áp một chiều 14V, thì cường độ dòng điện đi qua điện trở lúc này sẽ là 14/ 400 = 0.035 Amperes.

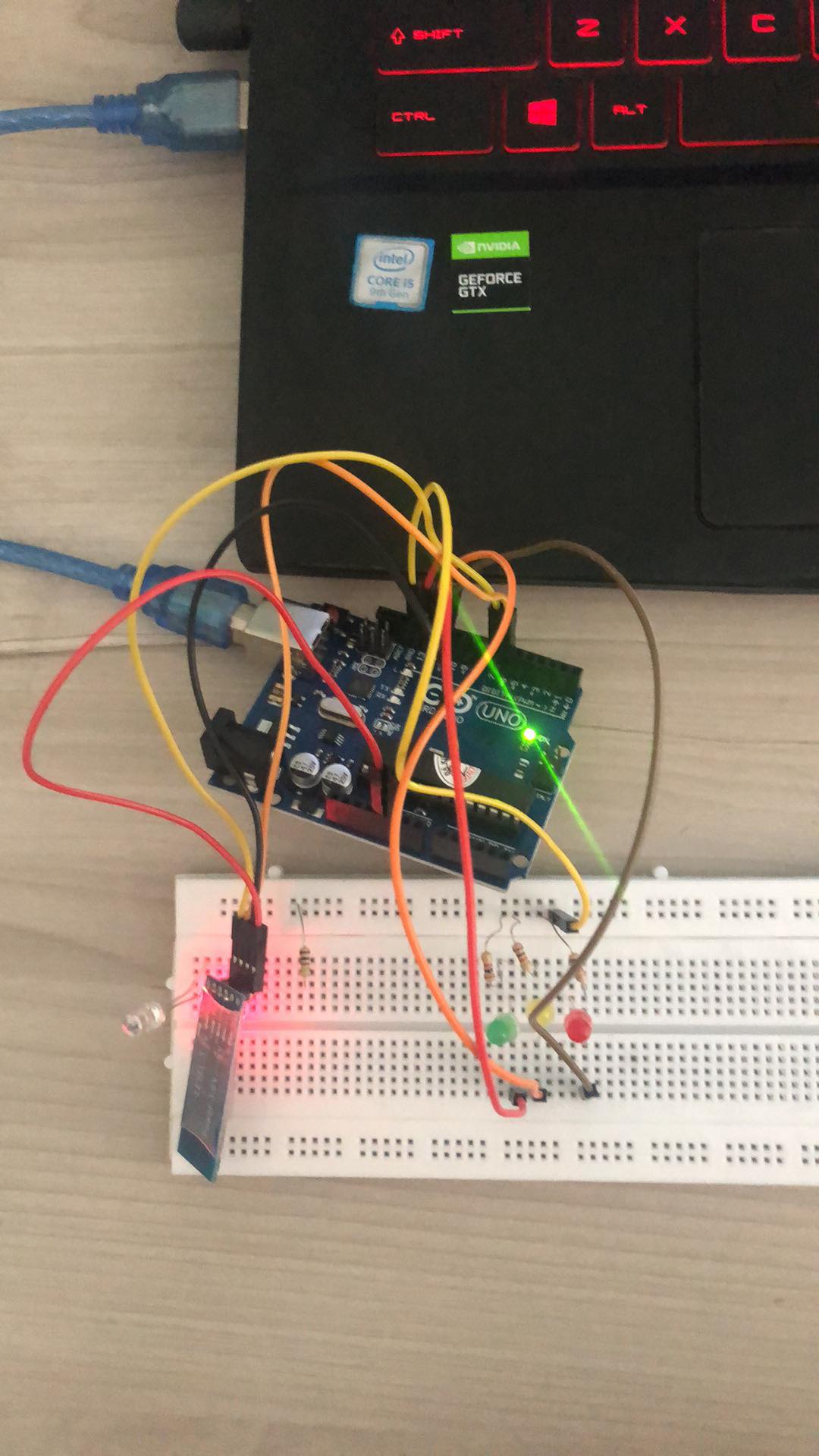
Điện trở thực tế hiện nay cũng có một số điện cảm và điện dung có ảnh hưởng trực tiếp đến mối quan hệ giữa điện áp và dòng điện trong mạch xoay chiều.

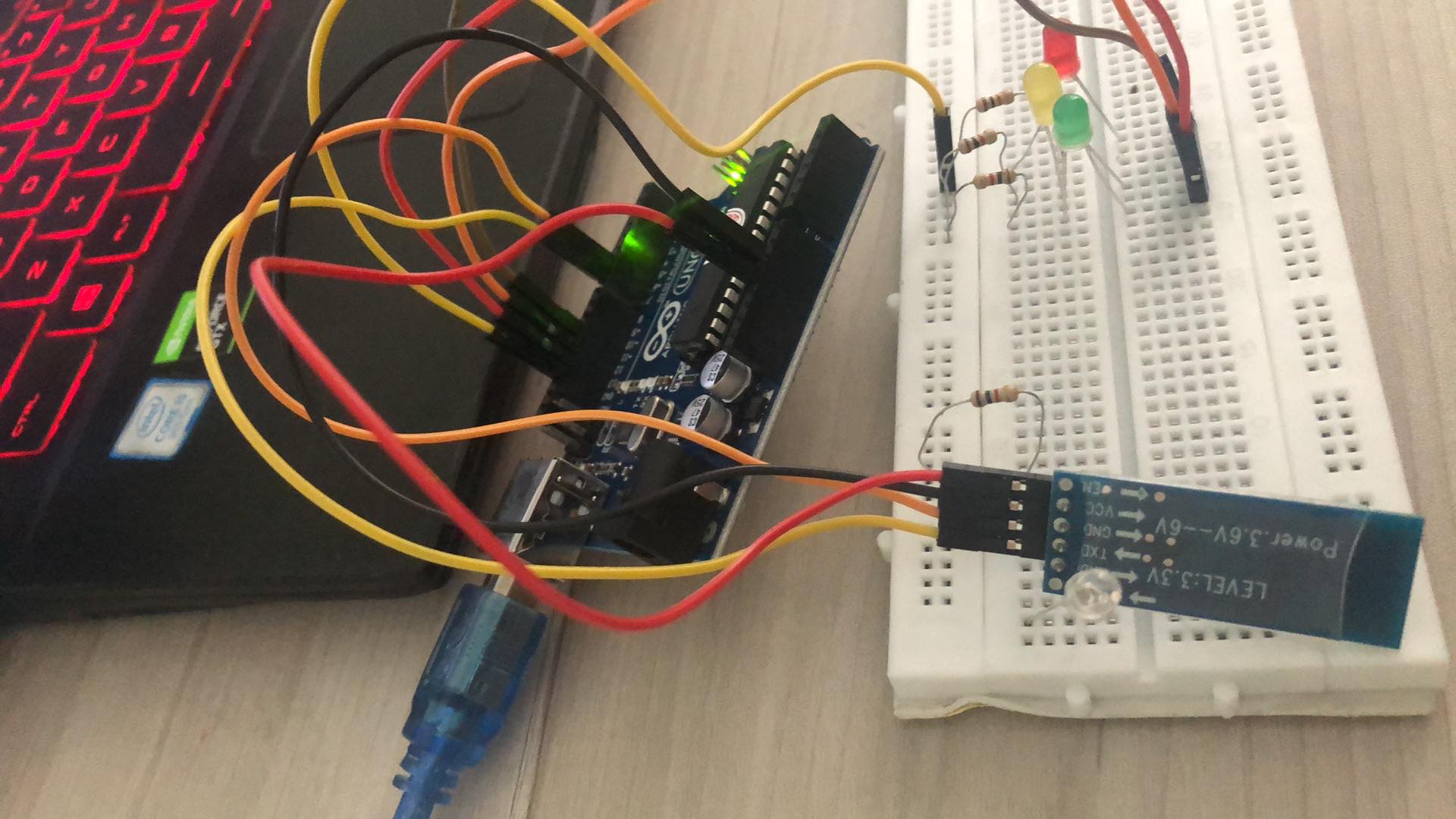
* ***Chức năng:***
* Là một linh kiện điện tử thụ động với 2 tiếp điểm kết nối, chức năng dùng để điều chỉnh mức độ tín hiệu, hạn chế cường độ dòng điện chảy trong mạch, dùng để chia điện áp, kích hoạt các linh kiện điện tử chủ động như transistor, tiếp điểm cuối trong đường truyền điện và có trong rất nhiều ứng dụng khác.
* Giúp tiêu tán một lượng lớn điện năng chuyển sang nhiệt năng trong các hệ thống phân phối điện, trong các bộ điều khiển động cơ. Các điện trở thường có trở kháng cố định, ít bị thay đổi bởi nhiệt độ và điện áp hoạt động.

## Sơ đồ mạch:



## 3.Mạch khi lắp ráp:

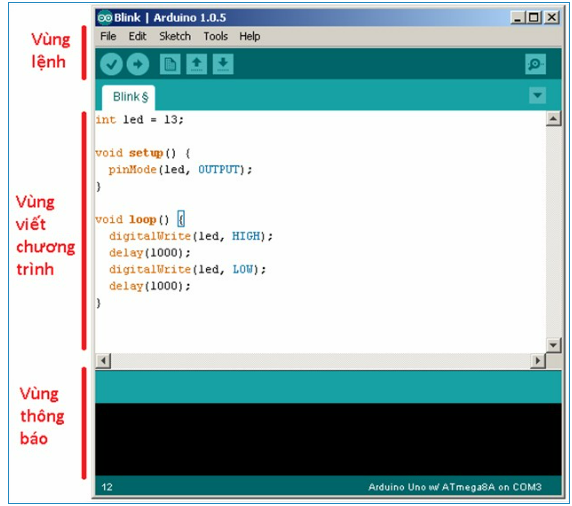




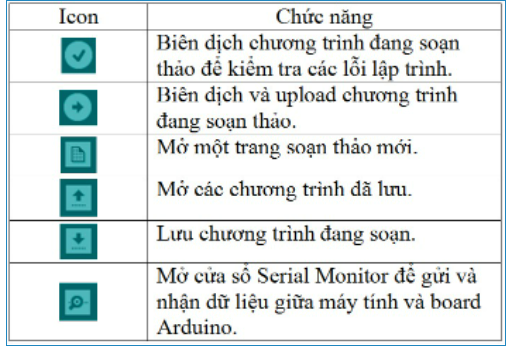
# CHƯƠNG 4: THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM

## 1**. Tổng quan về Arduino IE**

* 1. . Giao diện:



1.2 Vùng lệnh:

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

* 1. **Vùng viết chương trình**

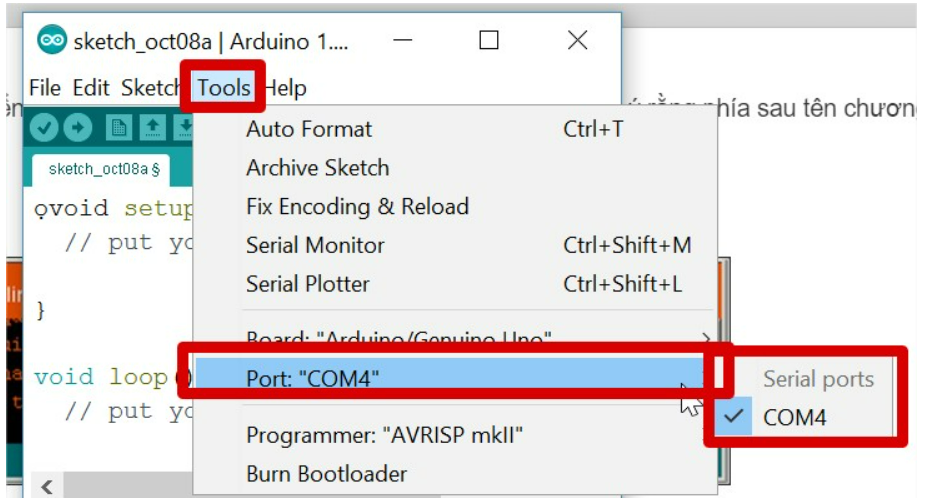
Ta sẽ viết các đoạn mã của mình tại đây. Tên chương trình của ta được hiển thị ngay dưới dãy các Icon, ở đây nó tên là “Blink”. Để ý rằng phía sau tên chương trình có một dấu “§”. Điều đó có nghĩa là đoạn chương trình của ta chưa được lưu lại.

* 1. **Vùng thông báo (debug)**

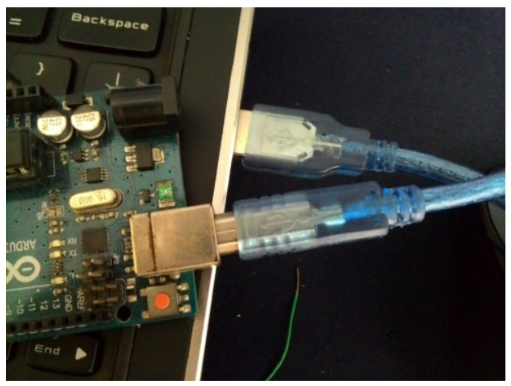
Những thông báo từ IDE sẽ được hiển thị tại đây. Để ý rằng góc dưới cùng bên phải hiển thị loại board Arduino và cổng COM được sử dụng. Luôn chú ý tới mục này bởi nếu chọn sai loại board hoặc cổng COM, ta sẽ không thể upload được code của mình.

* 1. Một số lưu ý:

Khi lập trình, các ta cần chọn port (cổng kết nối khi gắn board vào) và board (tên board mà ta sử dụng). Giả sử, ta đang dùng mạch Arduino Uno, và khi gắn board này vào máy tính bằng cáp USB nó được nhận là COM4 thì ta chỉnh như thế này là có thể lập trình đươc nhé.

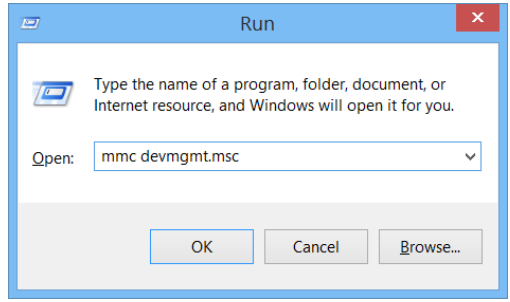


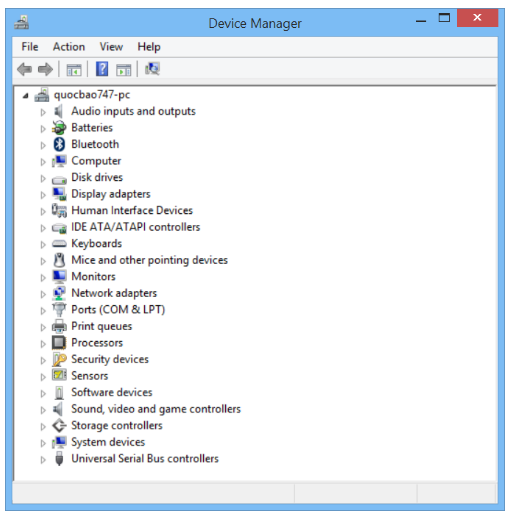
1. Hướng dẫn nạp code cho Arduino uno R3:

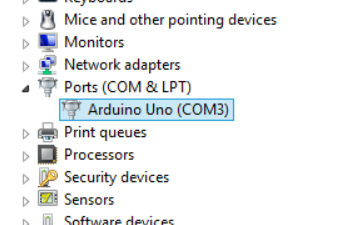
**Bước 1:** Kết nối Arduino UNO R3 vào máy tính

**Bước 2:** Tìm cổng kết nối của Arduino Uno R3 với máy tính

Khi Arduino Uno R3 kết nối với máy tính, nó sẽ sử dụng một cổng COM (Communication port - cổng dữ liệu ảo) để máy tính và bo mạch có thể truyền tải dữ liệu qua lại thông qua cổng này. Windows có thể quản lí đến 256 cổng COM. Để tìm được cổng COM đang được sử dụng để máy tính và mạch Arduino UNO R3 giao tiếp với nhau, ta phải mở chức năng Device Manager của Windows.

Ta mở cửa sổ Run và gõ lệnh mmc devmgmt.msc.

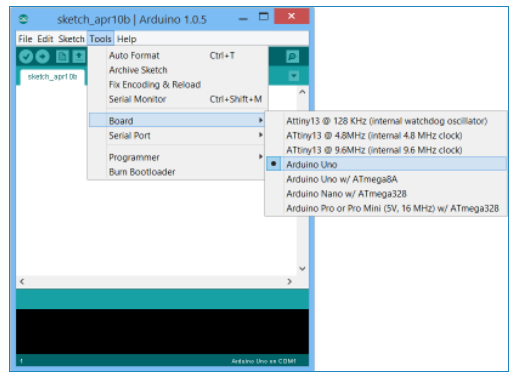
Sau đó bấm Enter, cửa sổ Device Manager sẽ hiện lên.

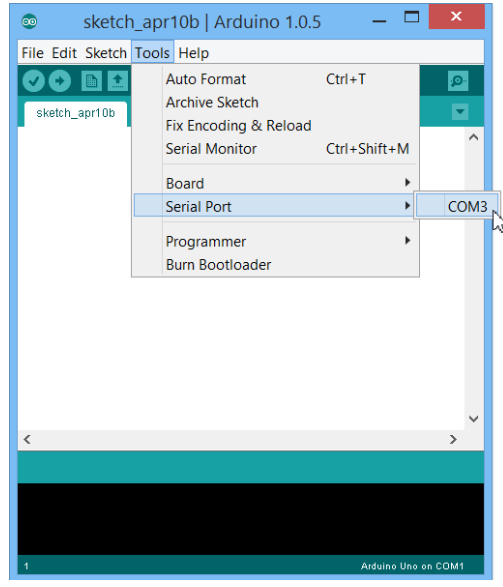
Mở mục Ports (COM & LPT), ta sẽ thấy cổng COM Arduino Uno R3 đang kết nối.

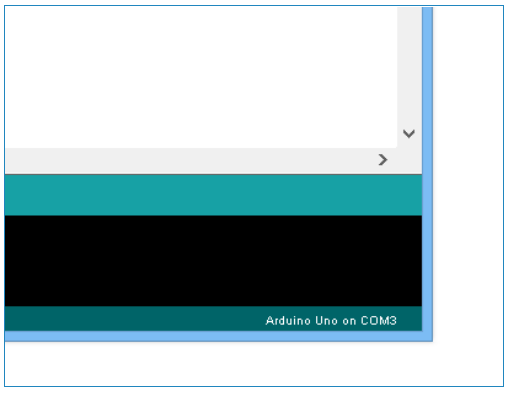
Cổng kết nối ở đây là COM3. Thông thường, trong những lần kết nối tiếp theo, Windows sẽ sử dụng lại cổng COM3 để kết nối nên ta không cần thực hiện thêm thao tác tìm cổng COM này nữa.

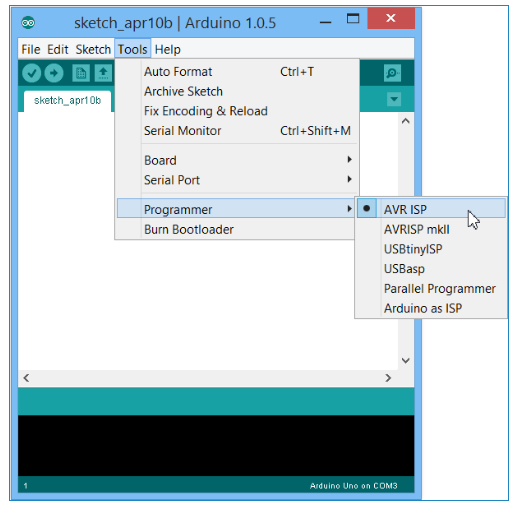
**Bước 3:** Khởi động Arduino IDE

**Bước 4:** Cấu hình phiên làm việc cho Arduino IDE

Vào menu Tools 🡢 Board 🡢 chọn Arduino Uno

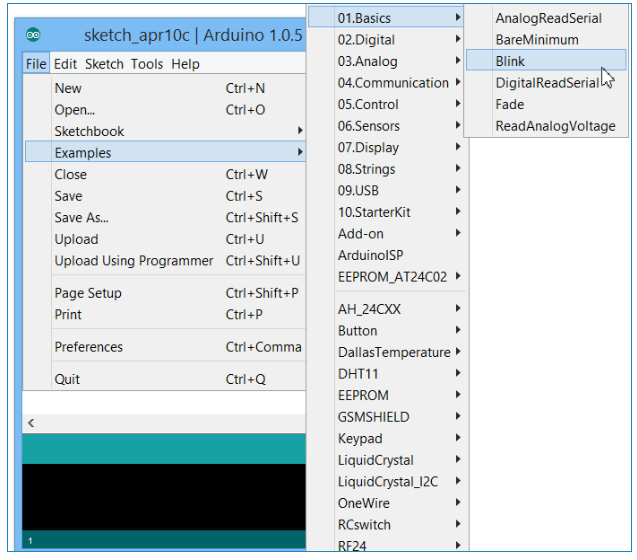
Vào menu Tools 🡢 Serial Port 🡢 chọn cổng Arduino đang kết nối với máy tính. Ở máy của mình là COM3.

Xác nhận cổng COM của Arduino IDE ở góc dưới cùng bên phải cửa sổ làm việc.

Vào menu Tools 🡢 Programmer 🡢 chọn AVR ISP

**Bước 5:** Mở và nạp mã nguồn chương trình mẫu:

Nạp một chương trình mẫu bằng cách vào menu File 🡢 Examples 🡢 01.Basics 🡢 chọn Blink.



Ta sẽ thấy Arduino IDE mở một cửa sổ mới chứa mã nguồn Blink. Mã này có chức năng là điều khiển đèn LED màu cam trên mạch Arduino Uno R3 nhấp nháy với chu kì 1 giây. Bấm tổ hợp phím Ctrl + U để tải chương trình lên mạch Arduino Uno R3.Kết quả thu được là đèn LED màu cam trên mạch sẽ nhấp nháy với chu kì 1 giây.

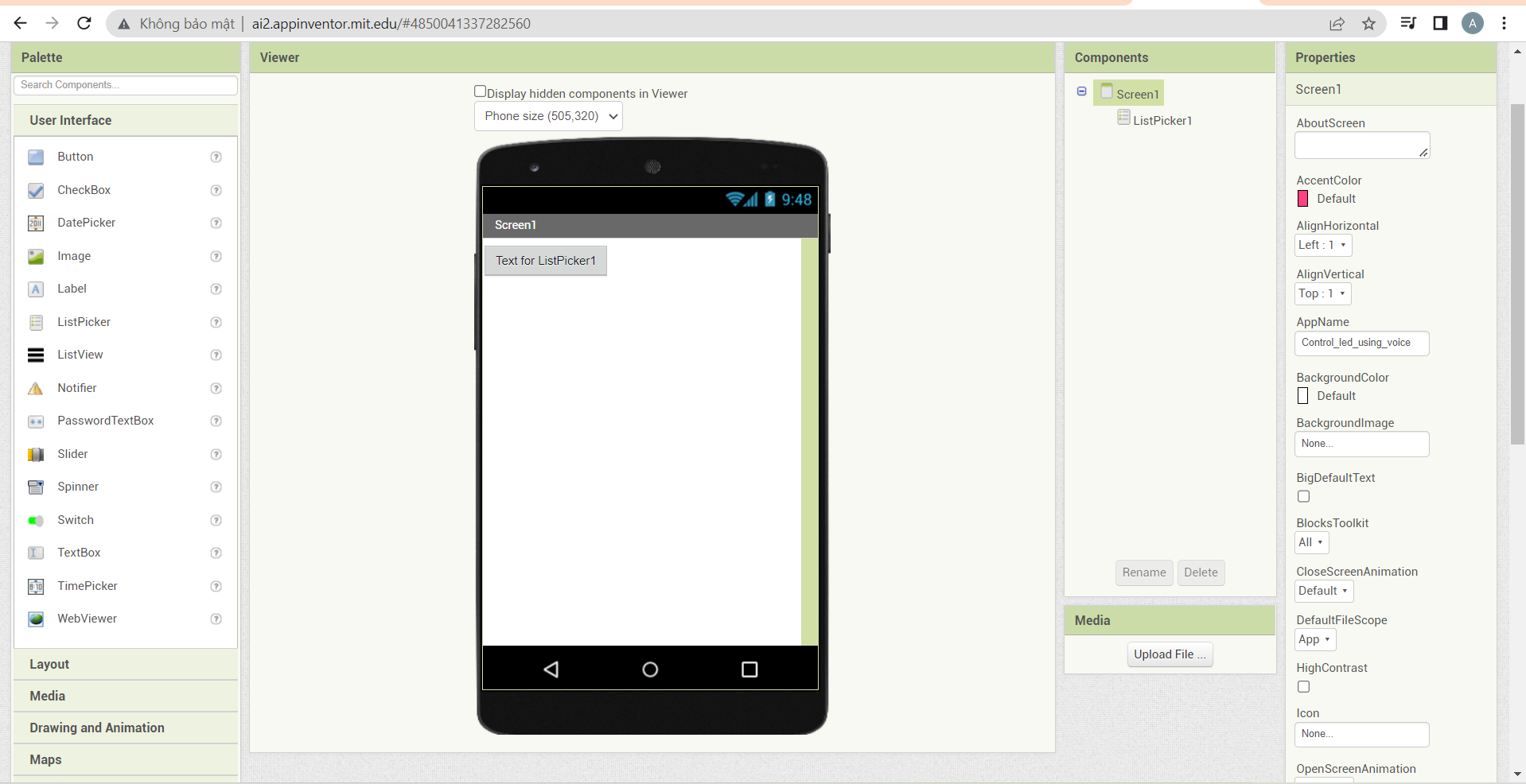
Ngoài sử dụng Arduino IDE để lập trình cho vi điều khiển, em còn sử dụng Matlab/Simulink để chạy mô phỏng.

## 2. **Thiết kế và thực hiện:**

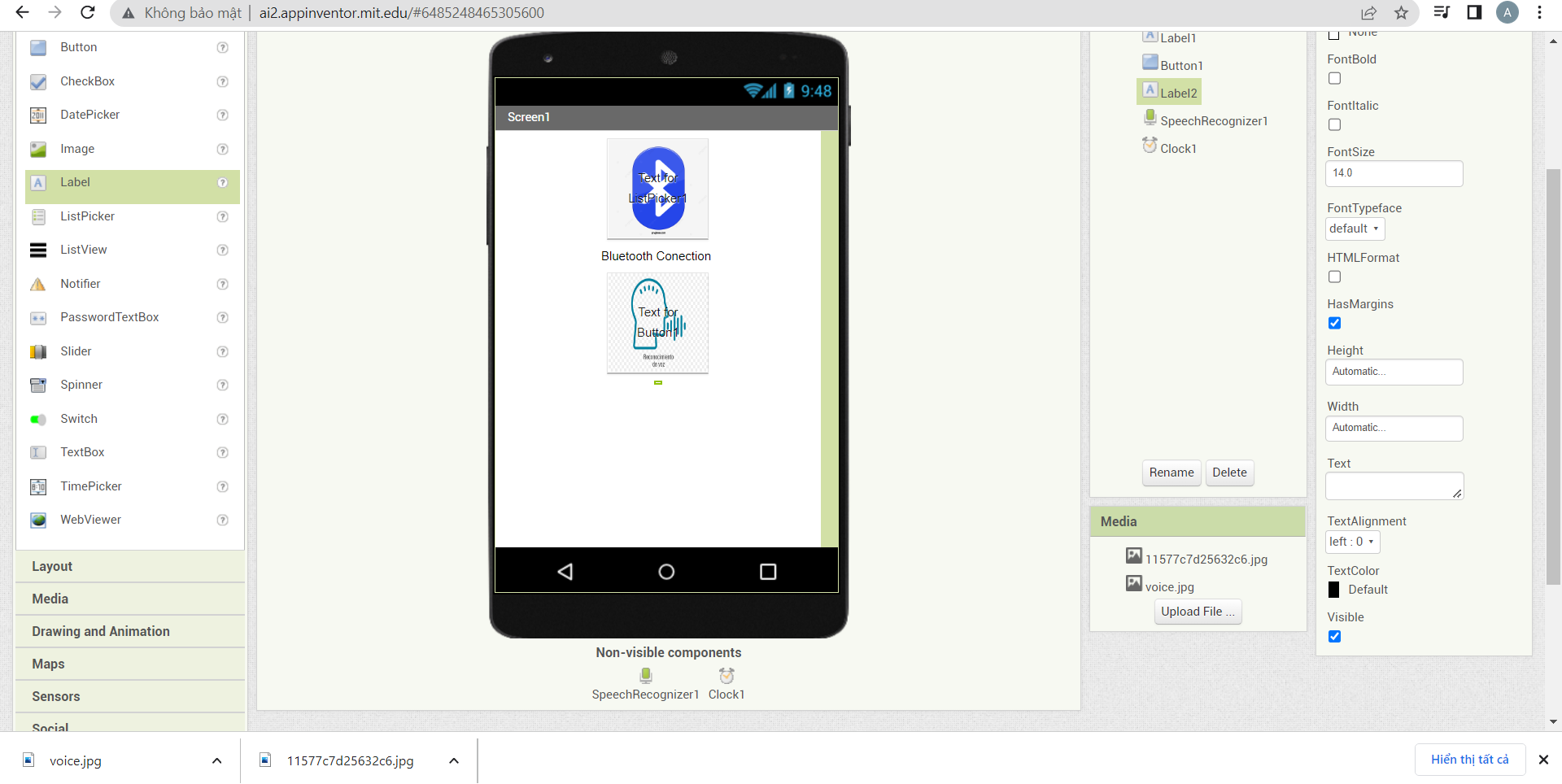
* 1. Lập trình cho ứng dụng điều khiển :

Để có thể điều khiển bằng giọng nói ta cần một ứng dụng để có thể nhận diện được giọng nói củng chúng ta.

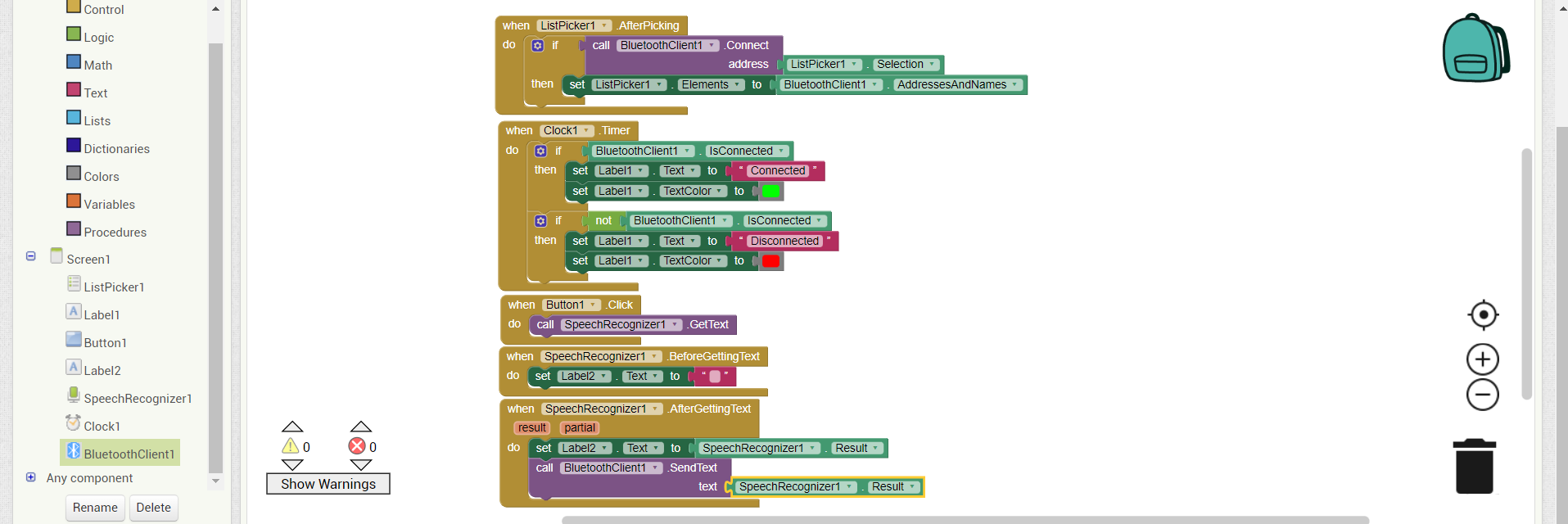
Truy cập trang web https://appinventor.mit.edu/



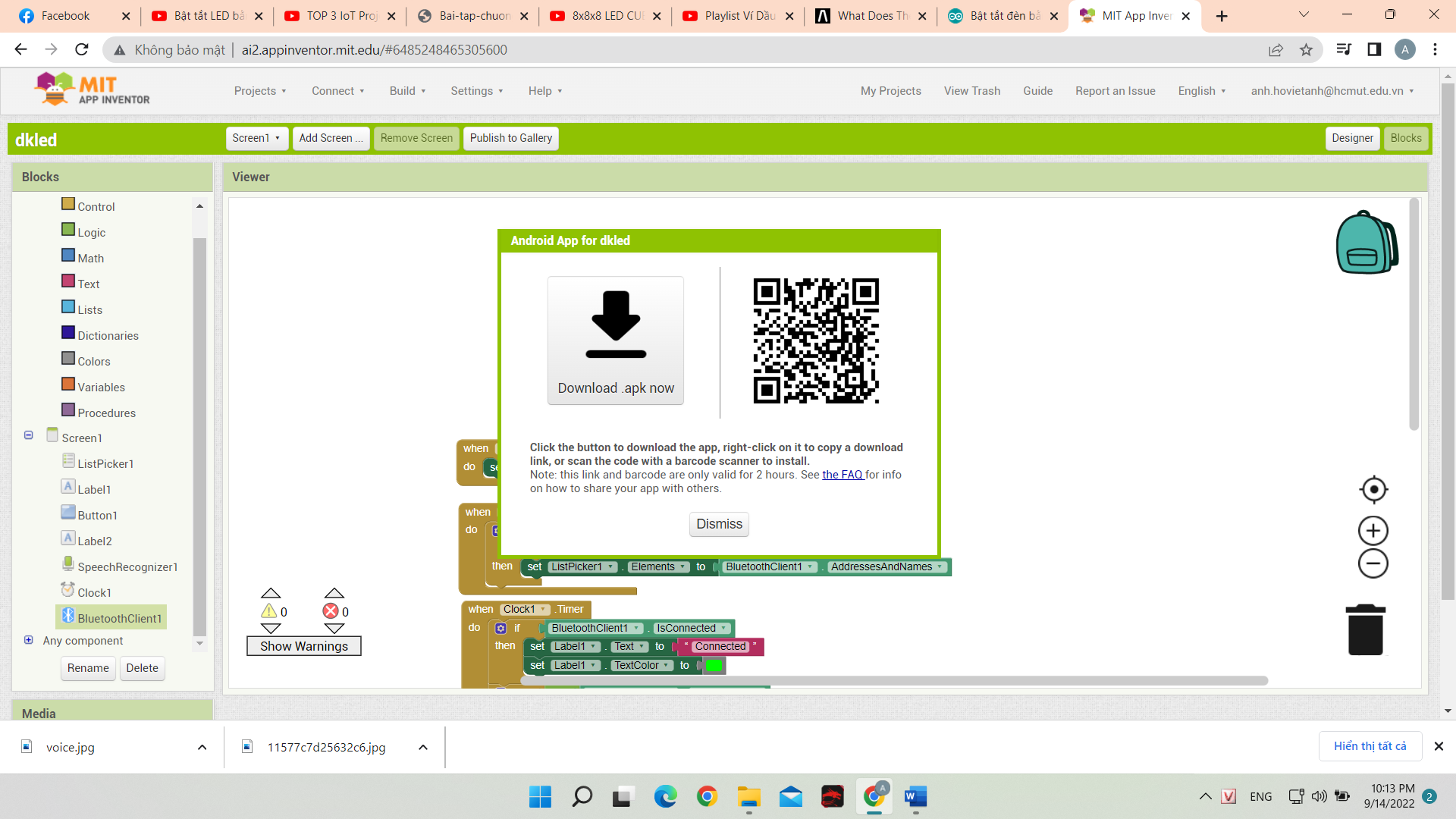
Thiết kế :



Code:



Sau khi biên dịch sẽ cho mã QR 🡪 lấy điện thoại quét và tải về



* 1. Lập trình cho Arduino:

#include <SoftwareSerial.h>

#define TX\_PIN 7

#define RX\_PIN 6

char value;

SoftwareSerial bluetooth(RX\_PIN, TX\_PIN);

int baudRate[] = {300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200};

char kytu;

String chuoi;

int n=0;

int ledxanh=13;

int leddo=11;

int ledvang=12;

void setup() {

Serial.begin(9600);

pinMode(ledxanh, OUTPUT);

pinMode(leddo, OUTPUT);

pinMode(ledvang, OUTPUT);

while (!Serial) {}

Serial.println("Configuring, please wait...");

for (int i = 0 ; i < 9 ; i++) {

bluetooth.begin(baudRate[i]);

String cmd = "AT+BAUD4";

bluetooth.print(cmd);

bluetooth.flush();

delay(100);

}

bluetooth.begin(9600);

Serial.println("Config done");

while (!bluetooth) {}

Serial.println("Enter AT commands:");

}

void loop() {

if (bluetooth.available()) {

kytu=bluetooth.read();

chuoi=chuoi+kytu;

Serial.println(kytu);

if(chuoi.indexOf("bật đèn xanh")>=0){

digitalWrite(ledxanh,HIGH);

Serial.println(chuoi);

chuoi="";

n=0;

}

if(chuoi.indexOf("tắt đèn xanh")>=0){

digitalWrite(ledxanh,LOW);

Serial.println(chuoi);

chuoi="";

n=0;

}

if(chuoi.indexOf("bật đèn đỏ")>=0){

digitalWrite(leddo,HIGH);

Serial.println(chuoi);

chuoi="";

n=0;

}

if(chuoi.indexOf("tắt đèn đỏ")>=0){

digitalWrite(leddo,LOW);

Serial.println(chuoi);

chuoi="";

n=0;

}

if(chuoi.indexOf("bật đèn vàng")>=0){

digitalWrite(ledvang,HIGH);

Serial.println(chuoi);

chuoi="";

n=0;

}

if(chuoi.indexOf("tắt đèn vàng")>=0){

digitalWrite(ledvang,LOW);

Serial.println(chuoi);

chuoi="";

n=0;

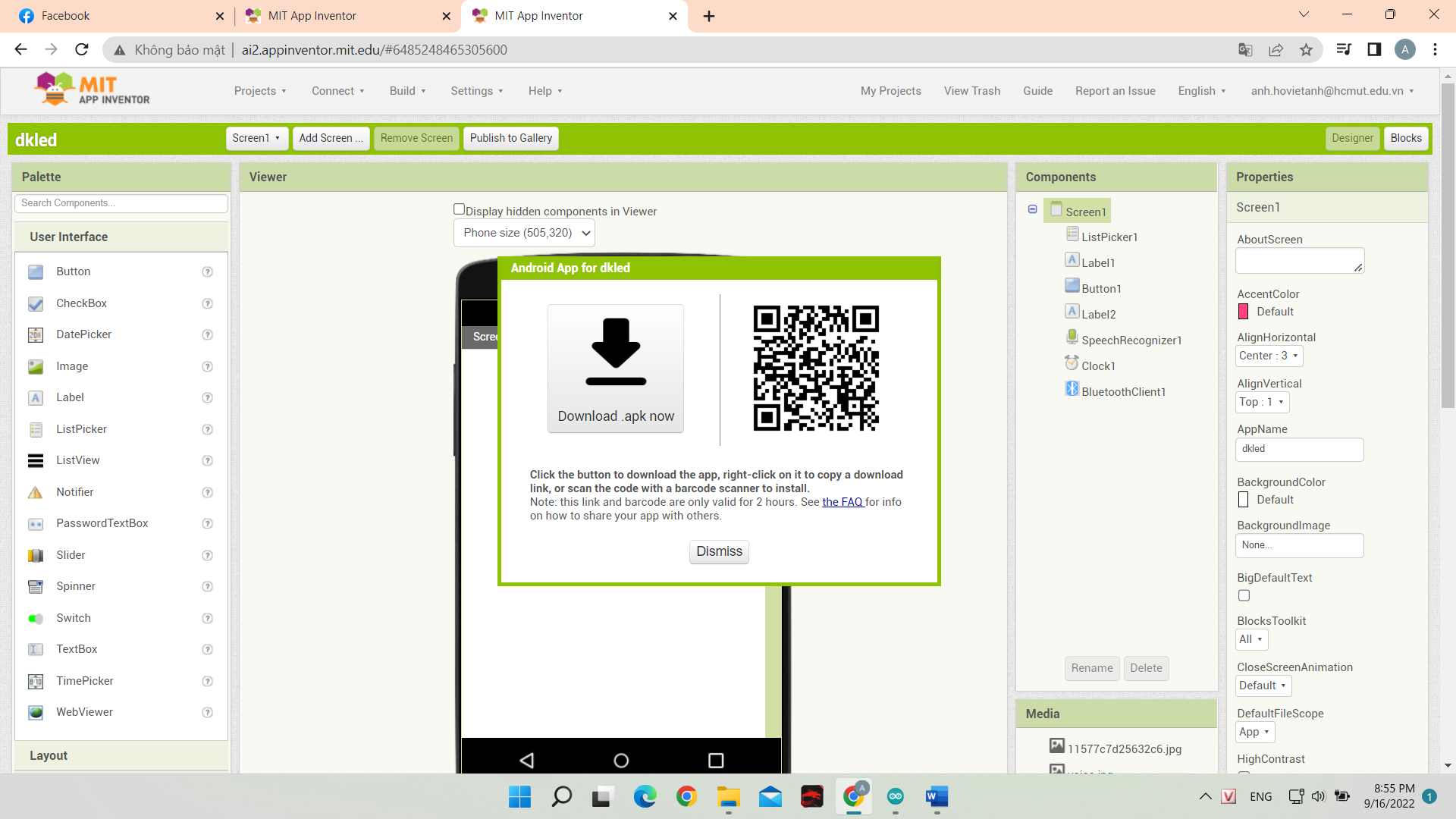
}

}

}

# CHƯƠNG 5: KẾT QUẢ THỰC HIỆN

Sau khi thiết kế app bấm build để nhận mã QR:



Quét mã QR và tải về điện thoại:

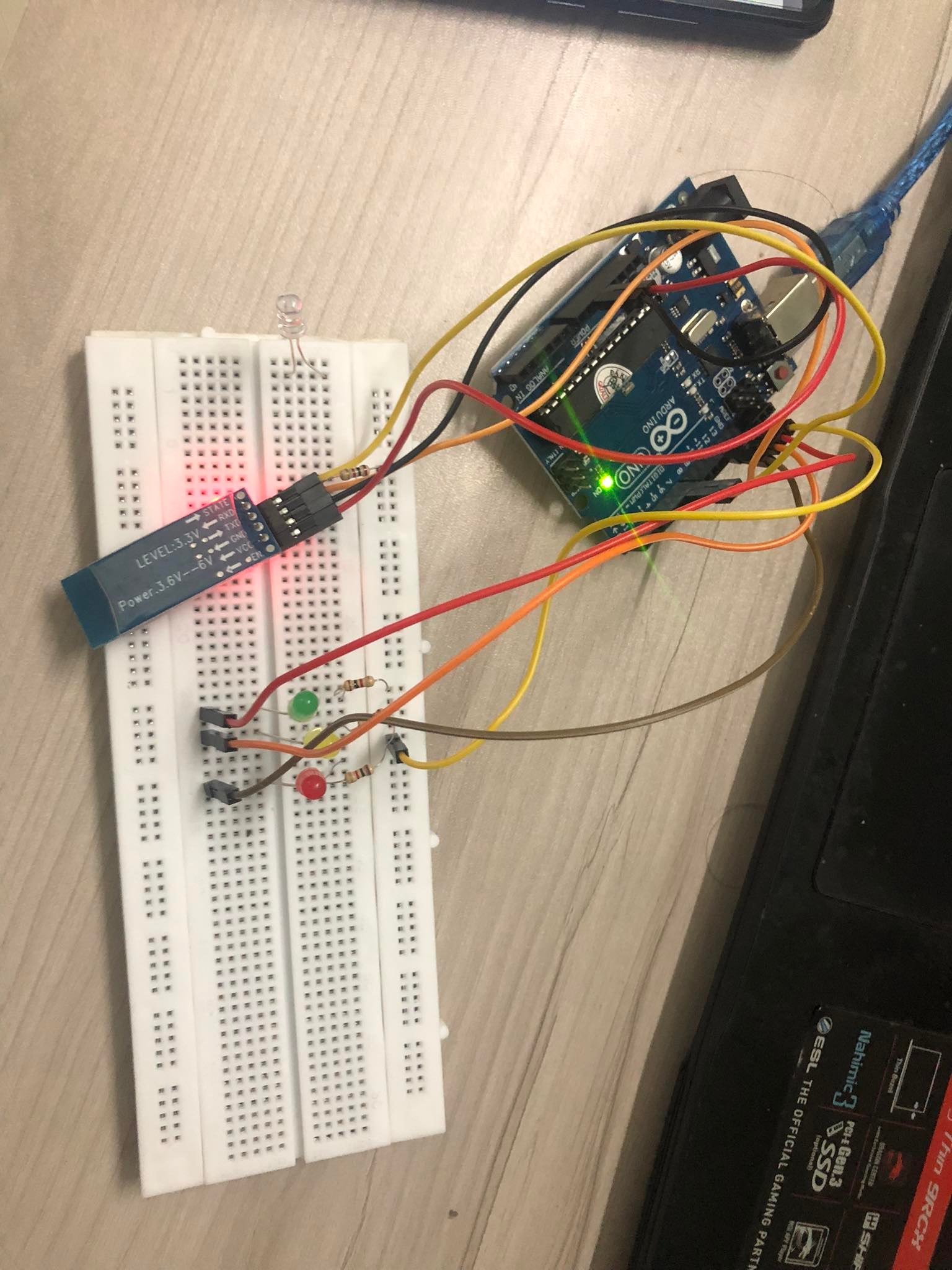
+ Khi chưa kết nối bluetooth



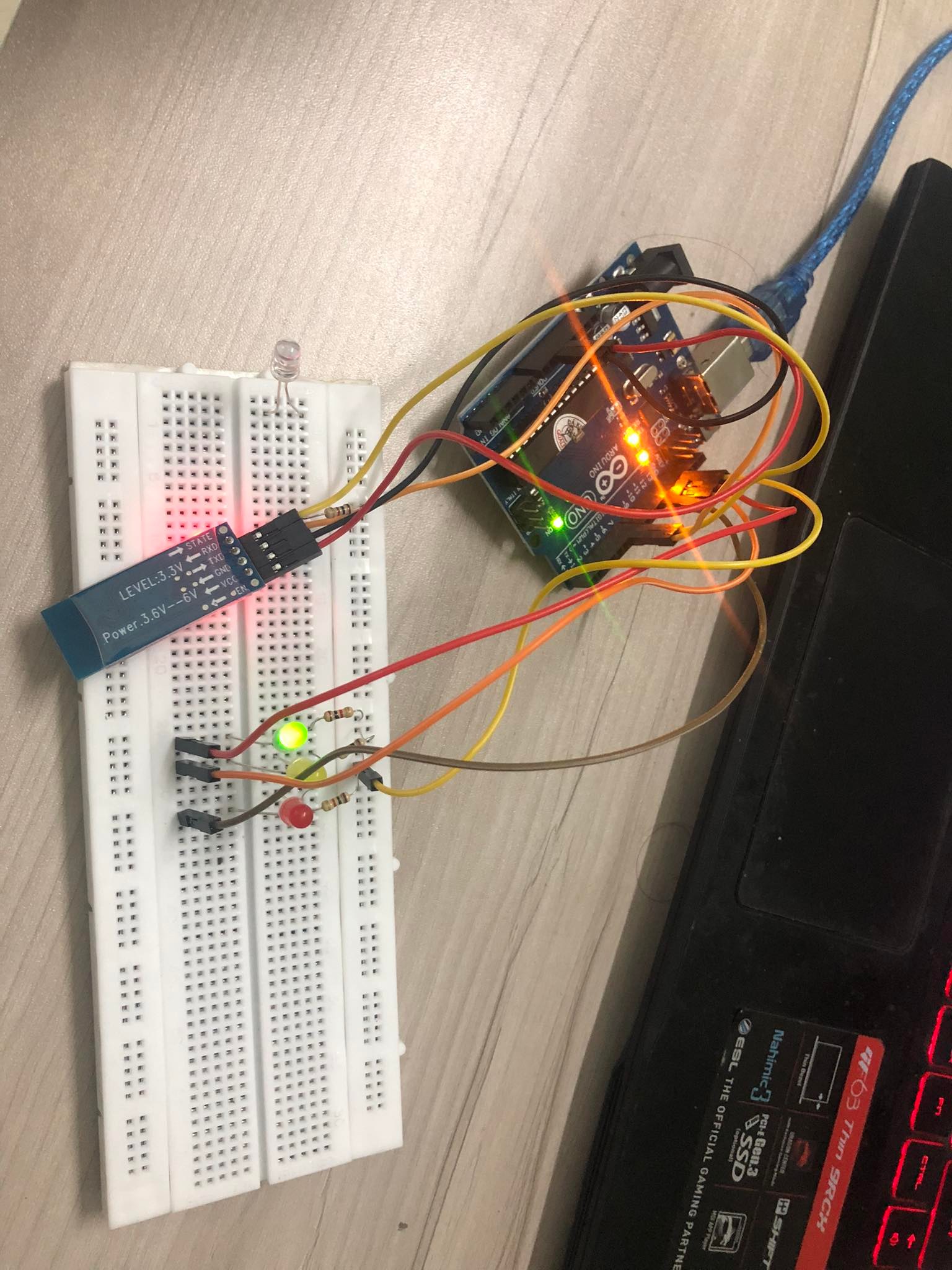
+ Kết nối Bluetooth thành công:



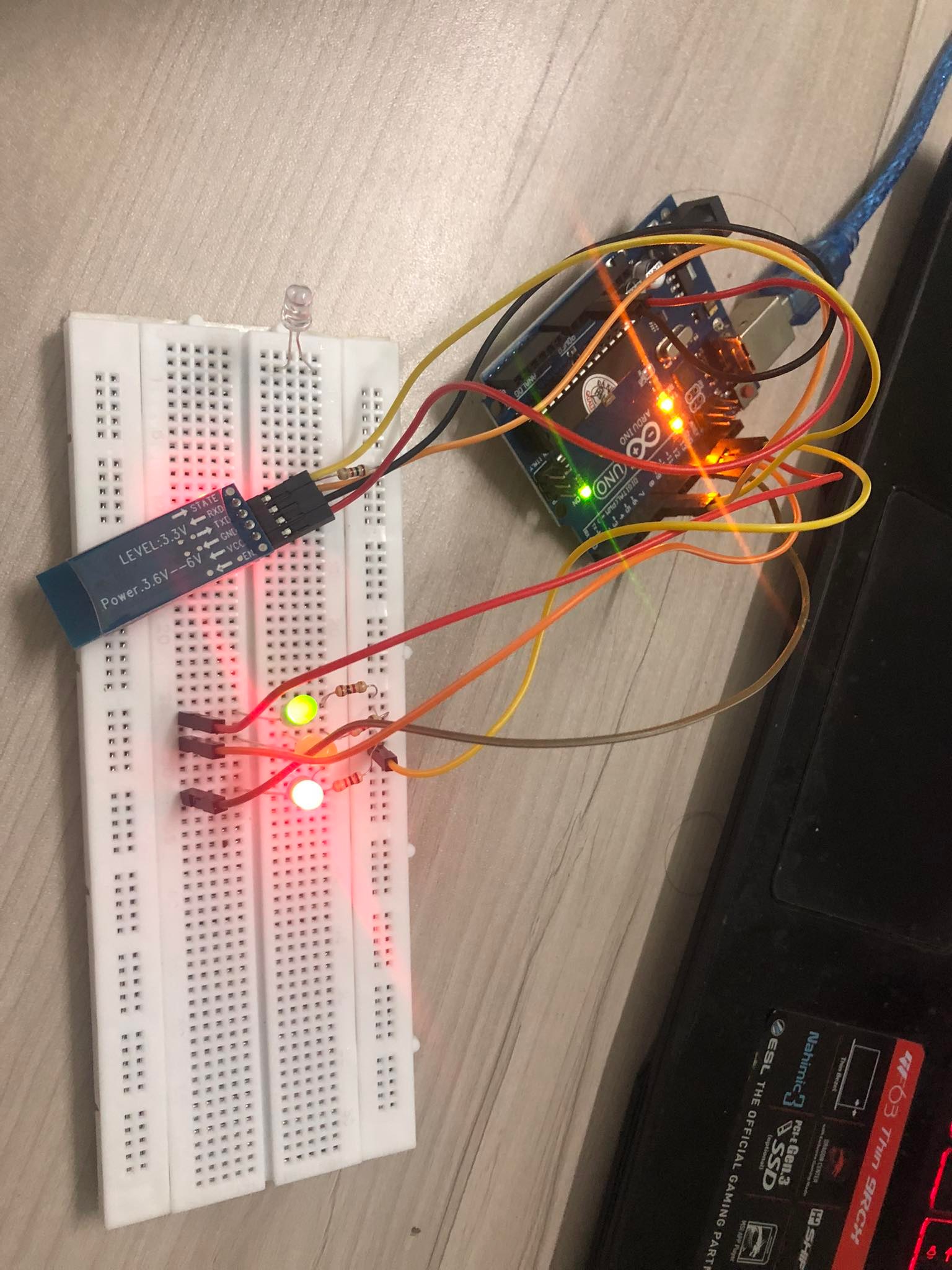
Mạch khi nạp code:



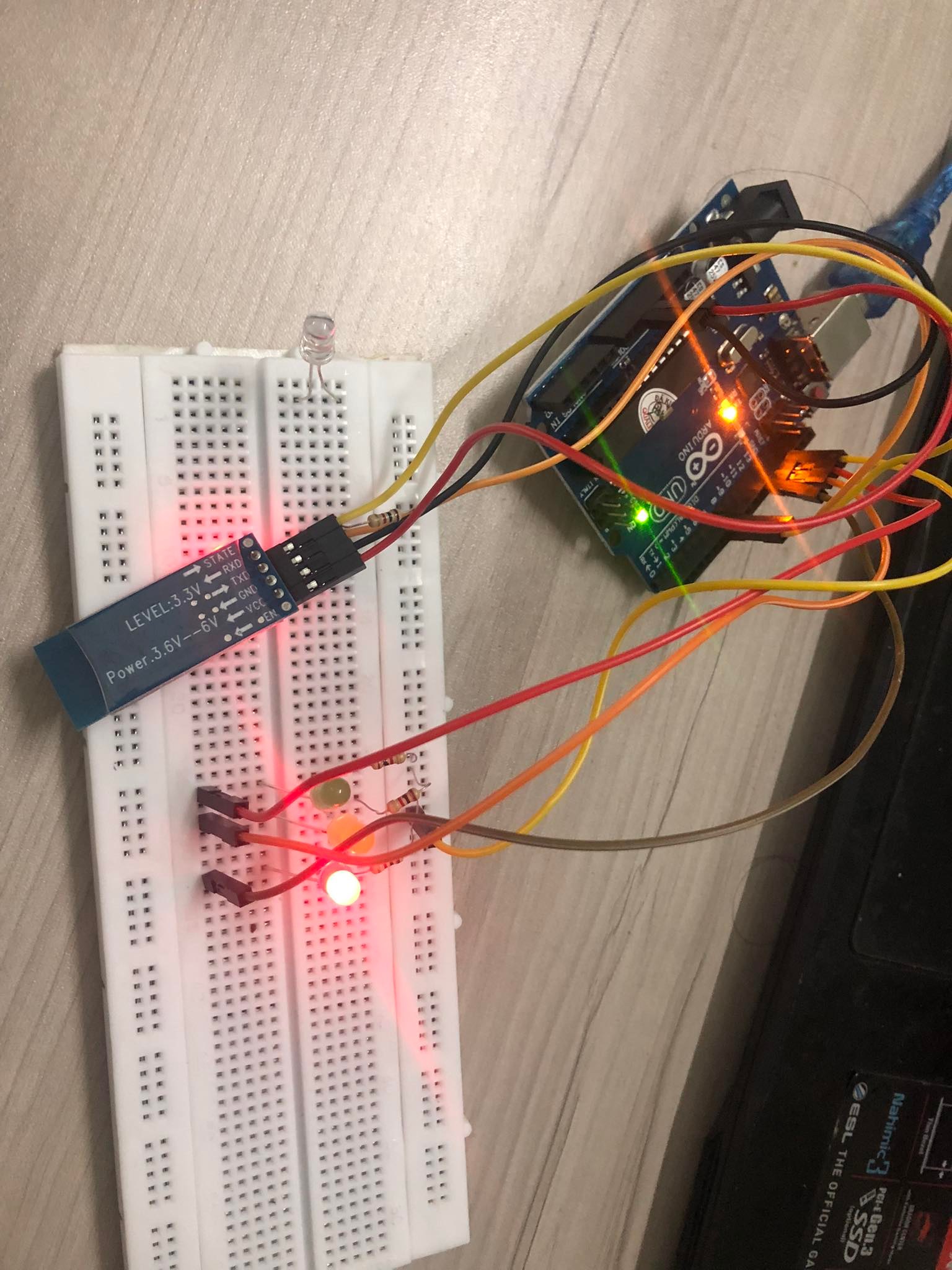
Mạch khi nói “ bật đèn xanh”:

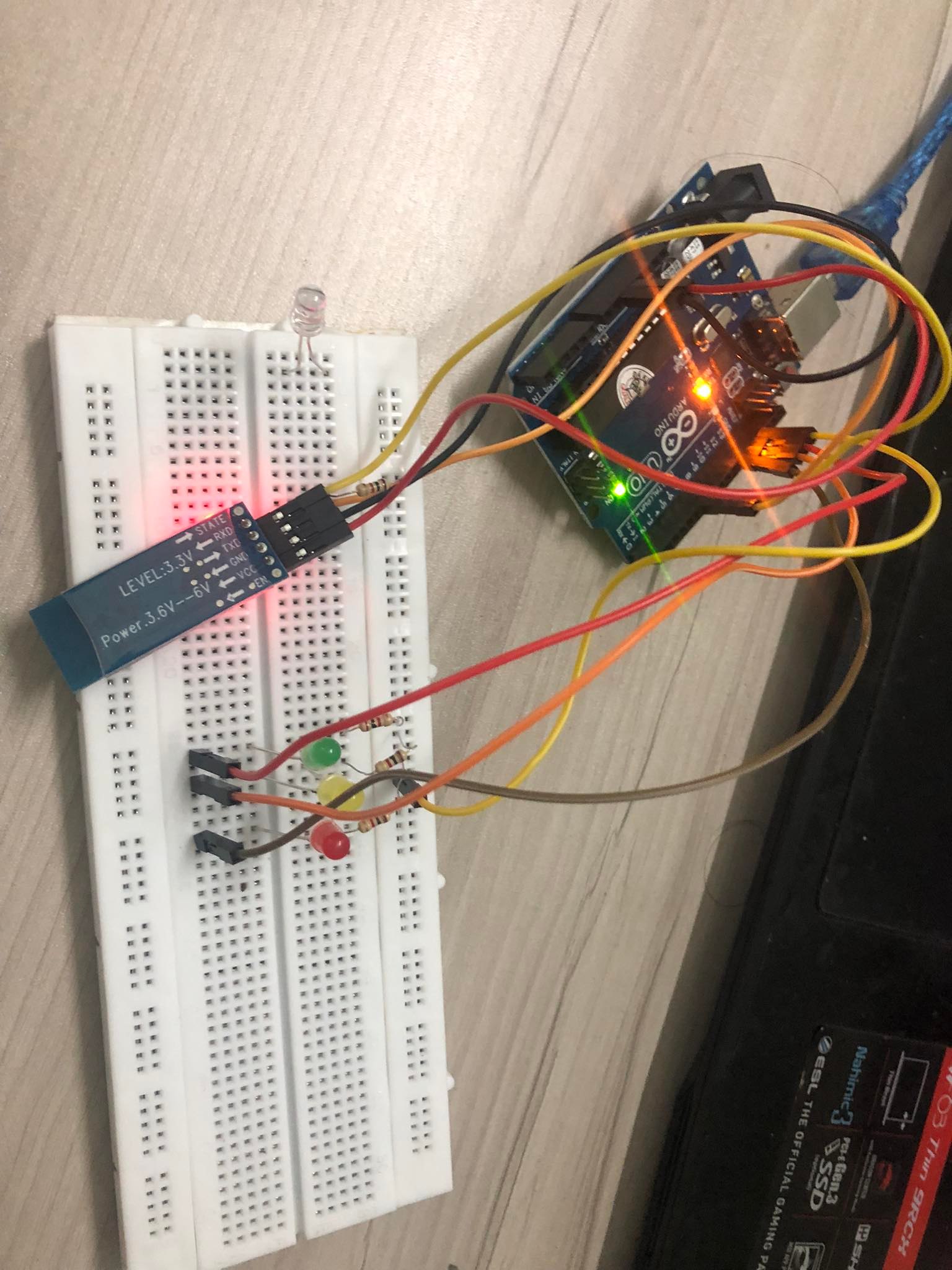


Mạch khi nói “bật đèn đỏ”:



Mạch khi nói “tắt đèn xanh”:



Mạch khi nói “tắt đèn đỏ”:

# CHƯƠNG 6: TÀI LIỆU THAM KHẢO

1. <http://arduino.vn/bai-viet/754-bat-tat-den-bang-giong-noi-voi-arduino>
2. <https://create.arduino.cc/projecthub/iotboys/control-led-using-your-voice-command-b396df>
3. <https://nshopvn.com/product/module-thu-phat-bluetooth-hc-05/>
4. <https://maysanxuattudong.com/mach-kit-arduino-uno-r3/>
5. Các tài liệu từ Internet, từ diễn đàn http://arduino.vn/ và tài liệu của các anh chị khóa trên*.*