

Lesson 2 Bluetooth Car

Points of this section

Arduino を学ぶとき、それは非常に重要です、特定の場所でワイヤレスであなたの車を制御するためにとてもクールである。だからレッスンで私たちはあなたに Bluetooth によって車をコントロールする方法を教えます

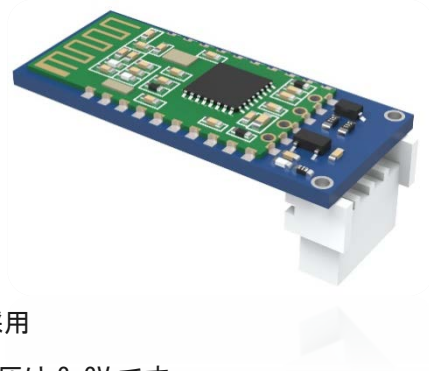
Learning Parts:

- ◆ Bluetooth モジュールと Bluetooth APP の使用方法を学ぶ
- ◆ Bluetooth 経由で車両を制御する方法を学ぶ
- ◆ この関数を実装するプログラムを書く

Preparations:

- ◆ 車両（バッテリー搭載）
- ◆ USB ケーブル
- ◆ Bluetooth モジュール
- ◆ iPhone またはタブレット

I . Bluetooth module

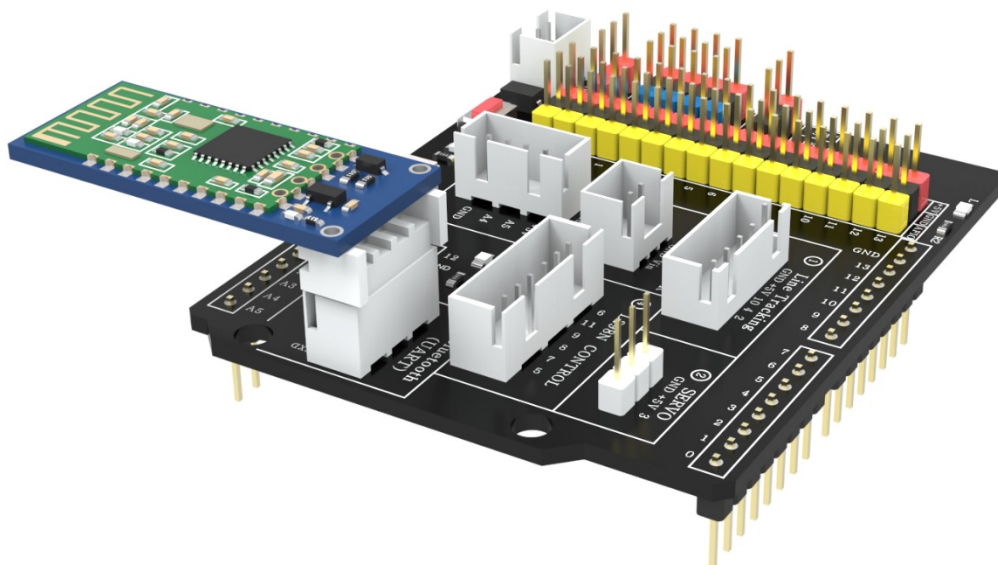


Bluetooth モジュールの説明：

1. 主流の Bluetooth V4.0. チップを採用
2. シリアルポートのアナログ動作電圧は 3.3V です
3. ユーザーはボーレートを 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 に設定できます。
4. キーモジュールの寸法は 28mm x 15mm x 2.35mm です。
5. 作動電流：40mA。
6. 休眠電流：1mA 未満。
7. GPS ナビゲーションシステム、水力ガスリーディングシステム、産業フィールドマイニング制御システムに使用されています。
8. Bluetooth ノートパソコン、Bluetooth アダプタ、PDA などのコンピュータに接続できます。

これは、UNO コントローラボードに接続された Bluetooth モジュールの回路図です。

実験では、拡張ボード V5 を介して UNO ボードに接続します。



II. Bluetooth APP 入門

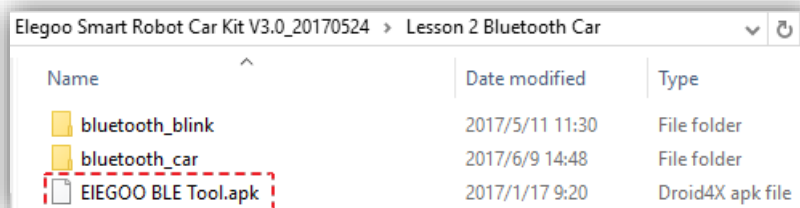
始める前に、HC-08 Bluetooth モジュールを拡張ボードに接続し、電源をオンにしてください。

STEP1:アプリケーションをインストールします。

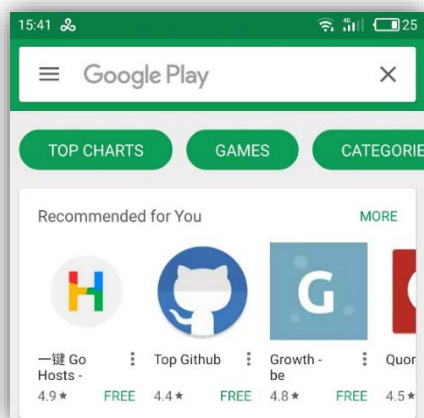
For Android

アプリケーションをインストールする方法は 2 通りあります。

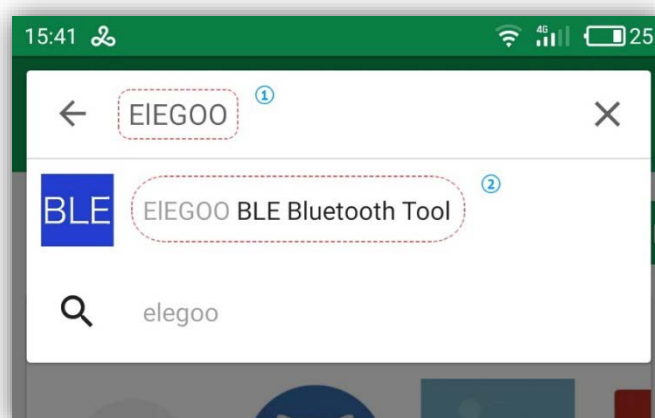
1. ELEGOO BLE Tool.apk ファイルを Android 製品にコピーしてインストールします。



2. Google Play ストアで「ELEGOO BLE Bluetooth Tool」を検索してインストールします。



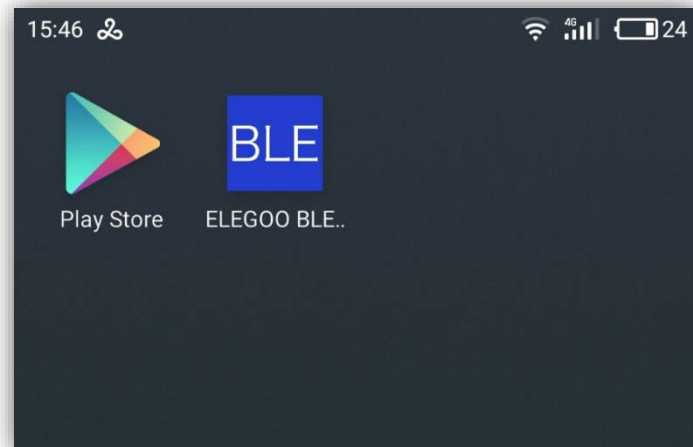
1. Launch Google Play



2. Search "ELEGOO BLE Bluetooth Tool"



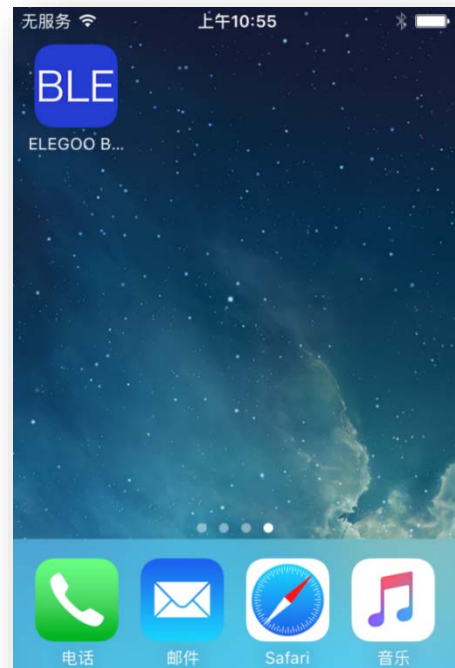
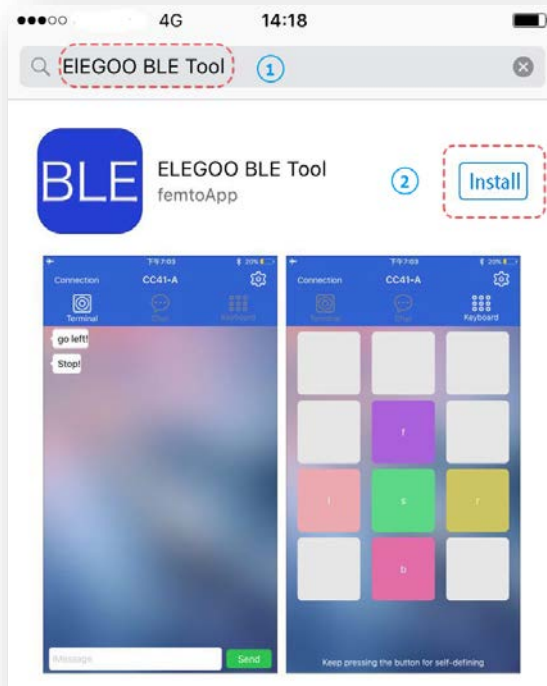
3. INSTSL



4. Finish the installation

For iOS

Apple APP Storeで「ELEGOO BLE ツール」を検索してインストールしてください。



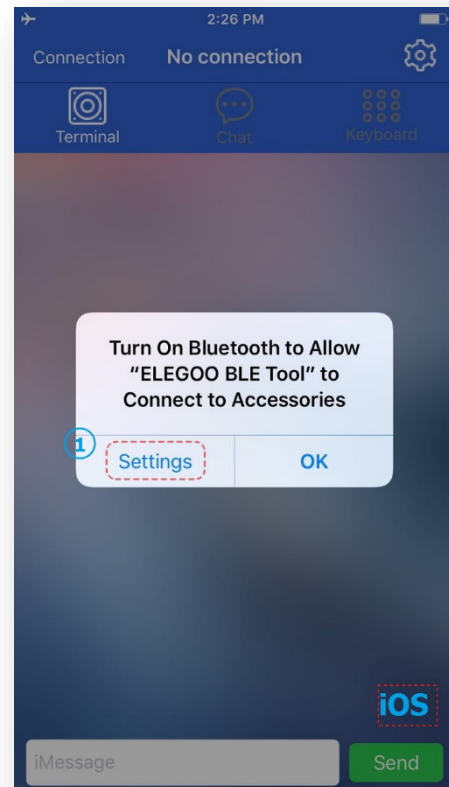
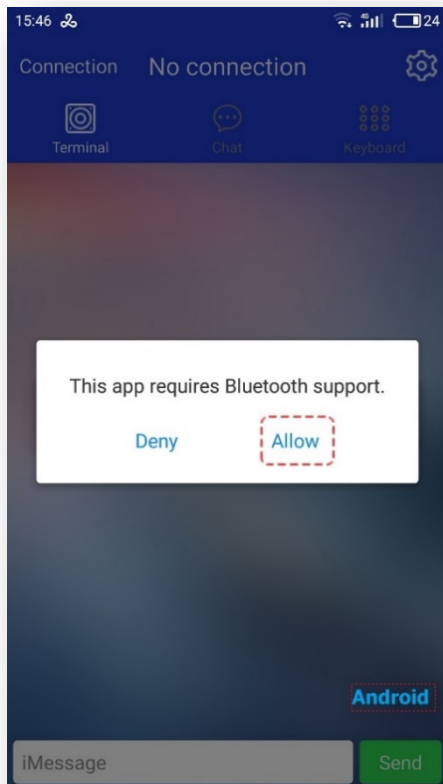
STEP2: Application Settings.

アプリケーションを起動します。



Bluetooth を開かないと、Bluetooth 機能がオンになるように通知されます。Android の場合は、[許可する]をクリックすると、Bluetooth が自動的に開きます。 iOS の場合は、[設定]をクリックします。

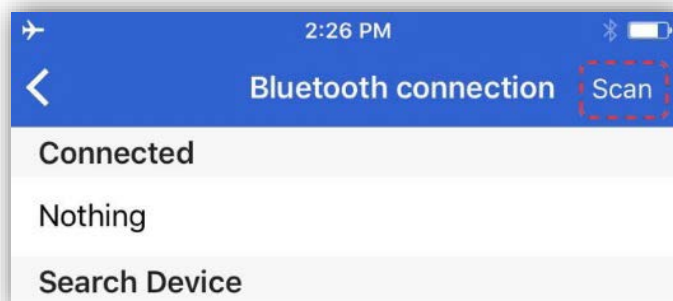
Bluetooth 設定インターフェースにジャンプするには、Bluetooth をオンにしてから、「ELEGOO BLE ツール」APP インターフェースに戻る必要があります。



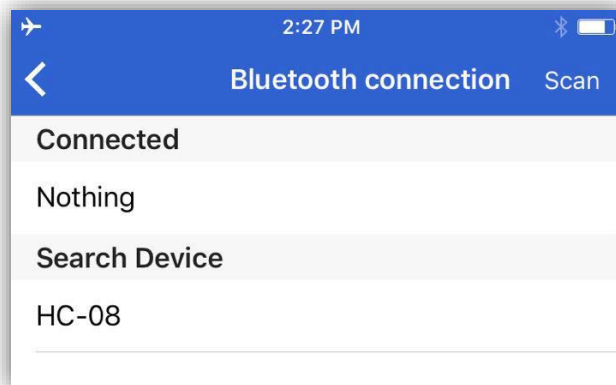
Bluetooth をオンにすると、APP インターフェースに Bluetooth アイコンが表示されます。「接続」をクリックします。



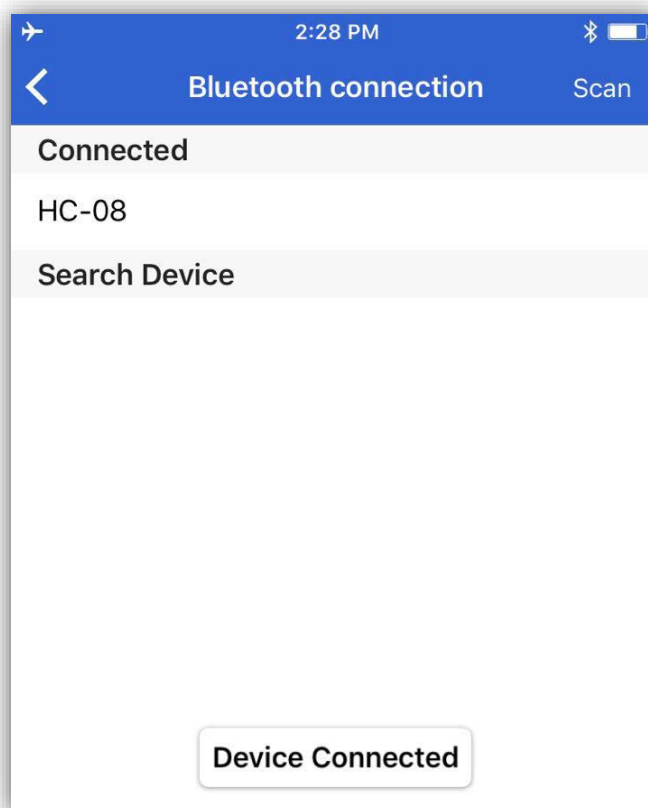
Bluetooth 信号をスキャンします。



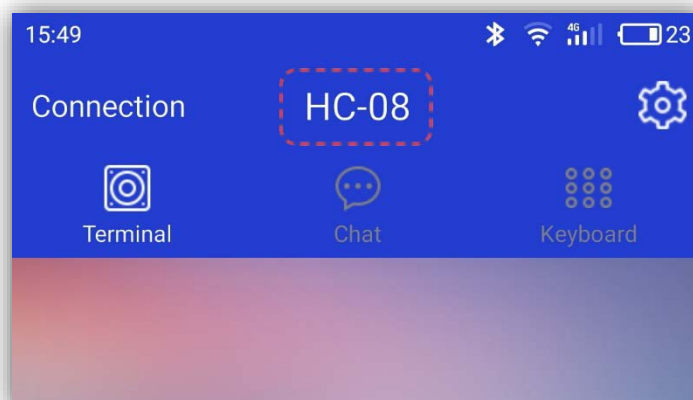
それから、お近くの Bluetooth 機器を電話で検索します。 HC-08 デバイスが表示されます。



Bluetooth 名 "HC-08"をクリックすると、接続に成功すると "Device Connected"と表示されます。



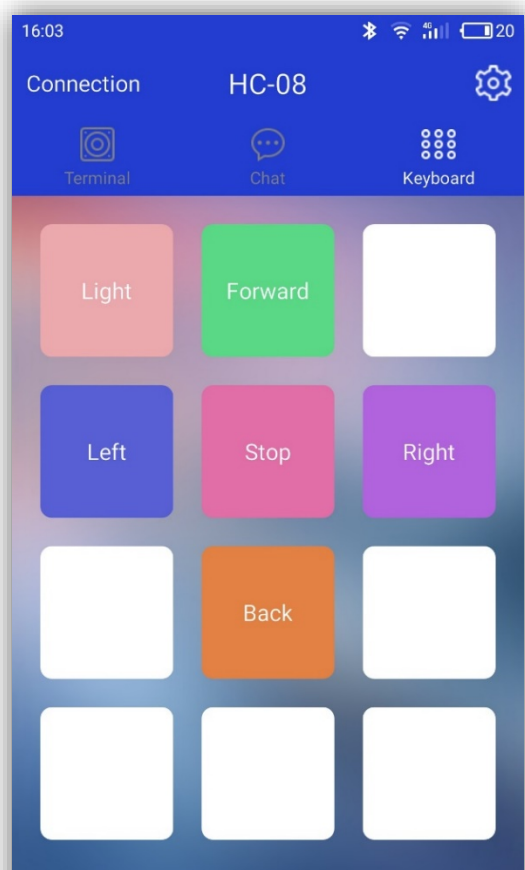
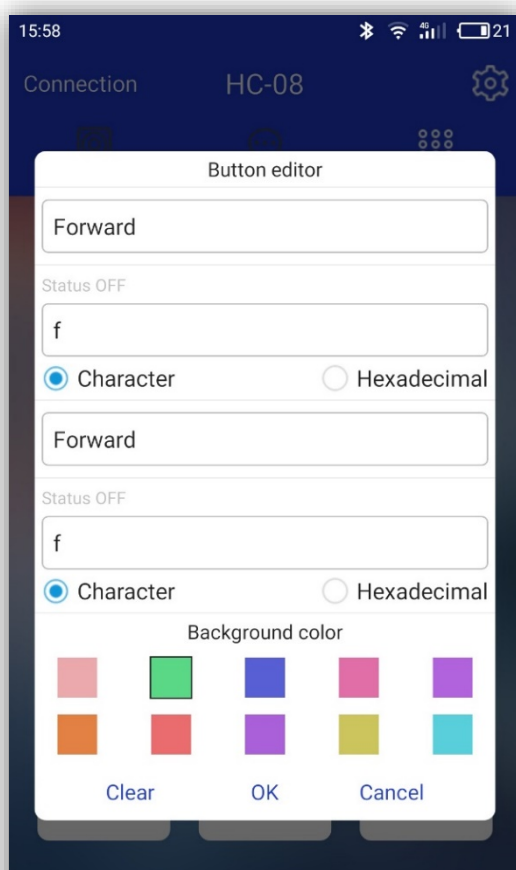
アプリケーションのメインインターフェイスを返します。 Bluetooth 接続後、画面に Bluetooth 名が表示されます。



次に、指で画面を右にスライドさせると、次のようなキーパターンが得られます。

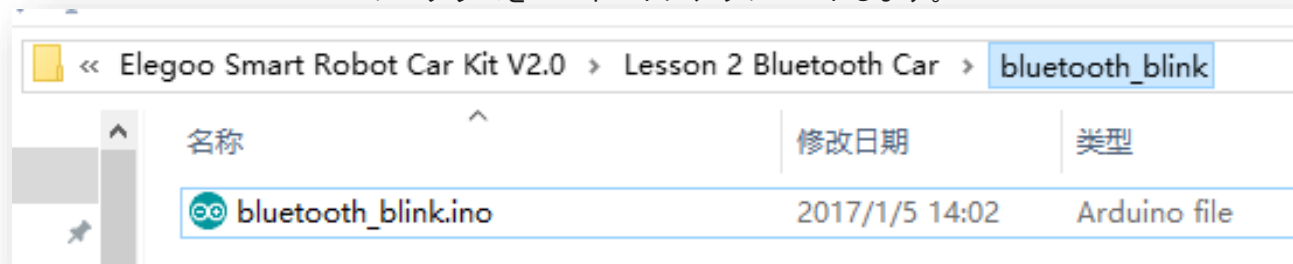


最後に、各ボタンの定義を設定します。たとえば、「進む」とします。以下を参照してください。残りのキー値も同様に設定されます。



III. Testing

"\ Elegoo Smart Robot Car Kit V3.0 \ bluetooth blinc \ bluetooth blink.ino"のパスにあるコードファイルを開き、プログラムを UNO ボードにアップロードします。



Code preview:

```
//www.elegoo.com

#define LED 13    //Define 13 pin for LED
bool state = LOW; //The initial state of the function is defined as a low level
char getstr;      //Defines a function that receives the Bluetooth character

void setup() {
    pinMode(LED, OUTPUT);
    Serial.begin(9600);
}

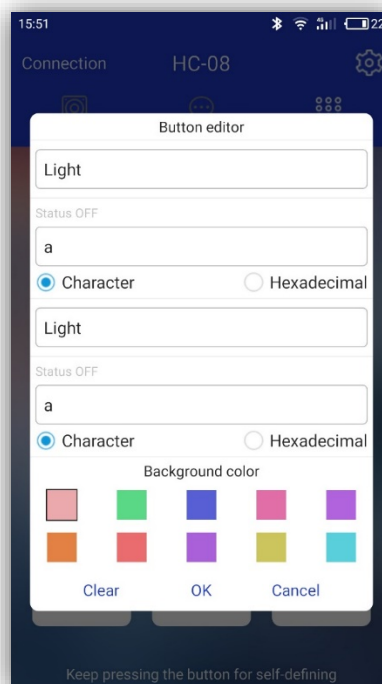
//Control LED sub function
void stateChange() {
    state = !state;
    digitalWrite(LED, state);
}

void loop() {
    //The Bluetooth serial port to receive the data in the function
    getstr = Serial.read();
    if(getstr == 'a'){
        stateChange();
    }
}
```

それをコンピュータから切り離し、車の電源を入れます。(ヒント：あなたがプログラムをアップロードするときに Bluetooth モジュールを引き出す必要があります。そうしないと、プログラムのアップロードに失敗します)。

Open APP

Bluetooth を使用して電話を車に接続した後、次のようにデータを設定します。



セットアップ後、このボタンを押してください。 UNO ボードのライトがスイッチで変わることがわかります。

The code

```
Serial.begin(9600);
```

このコードブロックの目的は、UNO コントロールボードのボーレートを 9600 に設定し、シリアルポートを開くことです。このようにして、Bluetooth モジュールの元のボーレートが 9600 であるため、それらは互いに通信することができます。

```
getstr = Serial.read(); //The Bluetooth serial port to receive the data in the function
if(getstr == 'a'){
    stateChange();
}
```

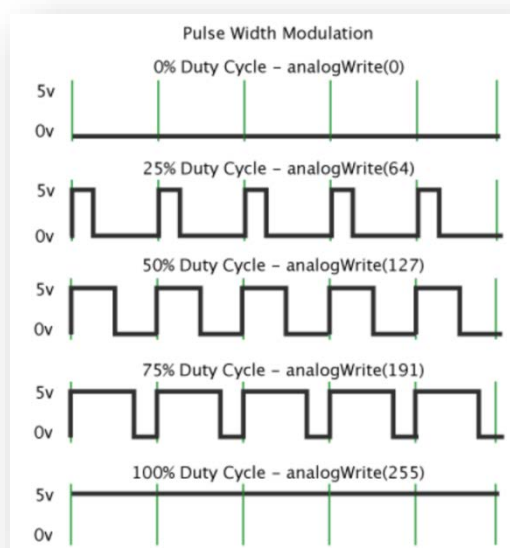
この機能は、循環機能内で繰り返し実行されます。まず、シリアルポートからデータを読み込み、データをチェックします。条件を満たす場合は、対応するサブ関数を実行します。たとえば、シリアルポートから文字「a」を読み取ると、LED ライトをオン/オフするサブファンクションが実行されます。

IV. Make a Bluetooth Car

車が左右に回転すると、速度をあまりにも速く設定する必要はありません。逆に、我々は車の速度を制御する必要があります。しかし、どのように制御する？

答えは PWM です。

PWMは「Pulse Width Modulation」の略でパルス変調と呼ばれ、アナログ回路をマイクロプロセッサのデジタル出力で制御するのに有効な技術であり、矩形波のデューティサイクルを変えることでモータの速度を変化させます。言い換えれば、モータの両側間を絶えず接続し、回路を遮断し、モータの仕事を保持するスイッチであり、速いスピードのために電源がオフになるとモータがオフにならない。したがって、パワーオンタイムとパワーオフタイムの特定の値を制御すると、車の速度を制御できます。回路が静止しているときは車の速度は最大になります。回路が停止しているとき車の速度は最小になります。車の速度は、半分の時間で中央値になります。PWMはデジタル方式でアナログ量を得る技術です。矩形波はデジタル制御で形成され、矩形波信号はオンとオフの2つの状態（デジタルピンの高低）のみを有する。持続時間のオン/オフ時間の特定の値を制御することにより、電圧を0~5Vに変化させる。占有時間（アカデミーでは高レベル）はパルス幅と呼ばれるので、PWMはパルス幅変調とも呼ばれます。下の5つの方形波によるPWMについて学びましょう。



上の緑の縦線は方形波の周期を表します。各アナログライト（ピン、値）に書き込まれる値はパーセンテージに対応し、パーセンテージはデューティサイクルとも呼ばれ、期間内のハイレベルとローレベルの間の特定の値から得られたパーセンテージを参照します。図の上から順に、最初の矩形波、デューティ・サイクルは0%、対応する値は0です。出力回路電流は最小であり、モーターは静止しています。継続時間が長いほど、回路電流が大きくなり速くなります。したがって、最終的なデューティサイクルは100%、対応する値は255、モータは最高速度で回転します。50%は中低速回転速度であり、25%は比較的遅く、始動できない（回路電流は静止摩擦のためにモータを起動するために比較的大きい）。PWMは主にLEDの光を調整してモータの回転速度を調整するために使用され、モータによって制御される車輪速度は容易に制御されます。PWMの利点は、あなたがArduinoのいくつかの車で遊ぶときに、より反映されます。

```
analogWrite(pin, value);
```

analogWrite () は、PWM ポートに対して 0~255 のアナログ値を書き込むために使用されます。あなたが注意する必要があるのは、analogWrite () は PWM の機能を持つデジタルピンにのみ使用されるということです。 UNO で PWM 機能を持つピンは 3, 5, 6, 9, 10, 11 のデジタルピンのみです。

私たちの車の速度は、ENA と ENB のピン 5 とピン 6 を接続することによって制御されます。 以下のプログラムは、デジタル関数 int carSpeed = 150 を設定しています。

速度はプログラムの下で制御されるので、あなた自身でスピードをコントロールすることができます。

```
analogWrite(ENA, carSpeed);  
analogWrite(ENB, carSpeed);
```

基本知識を習得した後、下記のようにプログラムを車にアップロードし、 "Eleugo Smart Robot Car Kit V3.0 ¥ bluetooth_car ¥ bluetooth_car.ino" のパスにコードファイルを開き、プログラムを UNO コントロールにアップロードします ボード。

Code preview:

```
//www.elegoo.com  
  
#define ENA 5  
#define ENB 6  
#define IN1 7  
#define IN2 8  
#define IN3 9  
#define IN4 11  
#define LED 13  
  
unsigned char carSpeed = 150;  
bool state = LOW;  
char getstr;  
  
void forward(){  
    digitalWrite(ENA,HIGH);  
    digitalWrite(ENB,HIGH);  
    digitalWrite(IN1,HIGH);  
    digitalWrite(IN2,LOW);  
    digitalWrite(IN3,LOW);  
    digitalWrite(IN4,HIGH);  
    Serial.println("Forward");  
}
```

```
void back(){
    digitalWrite(ENA,HIGH);
    digitalWrite(ENB,HIGH);
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);
    digitalWrite(IN3,HIGH);
    digitalWrite(IN4,LOW);
    Serial.println("Back");
}

void left(){
    analogWrite(ENA,carSpeed);
    analogWrite(ENB,carSpeed);
    digitalWrite(IN1,LOW);
    digitalWrite(IN2,HIGH);
    digitalWrite(IN3,LOW);
    digitalWrite(IN4,HIGH);
    Serial.println("Left");
}

void right(){
    analogWrite(ENA,carSpeed);
    analogWrite(ENB,carSpeed);
    digitalWrite(IN1,HIGH);
    digitalWrite(IN2,LOW);
    digitalWrite(IN3,HIGH);
    digitalWrite(IN4,LOW);
    Serial.println("Right");
}

void stop(){
    digitalWrite(ENA,LOW);
    digitalWrite(ENB,LOW);
    Serial.println("Stop!");
}

void stateChange(){
    state = !state;
    digitalWrite(LED, state);
}

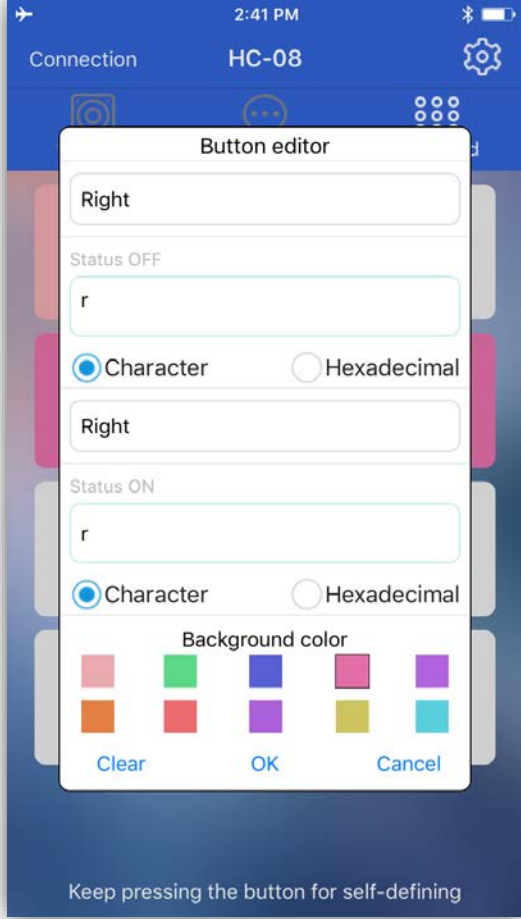
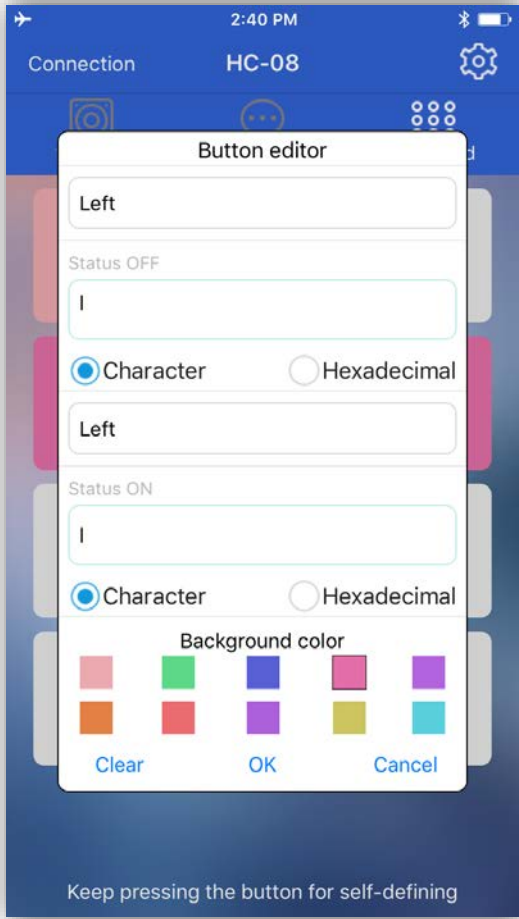
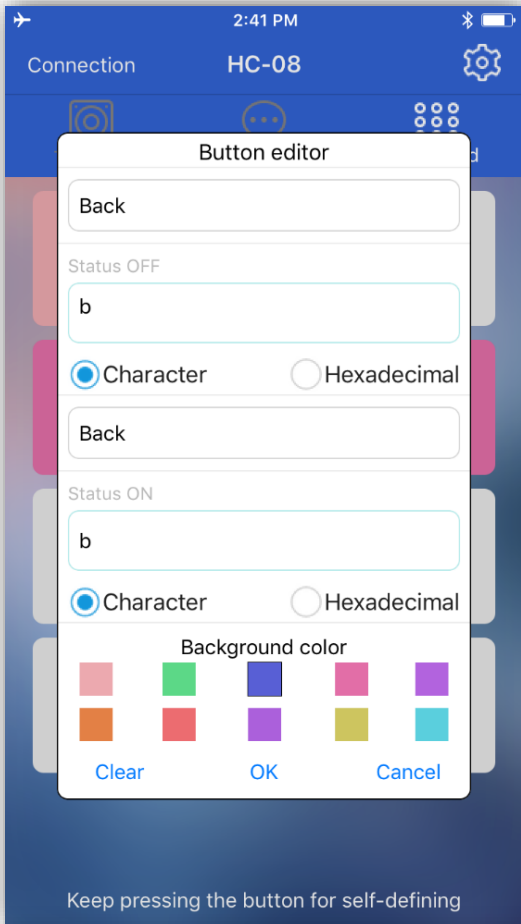
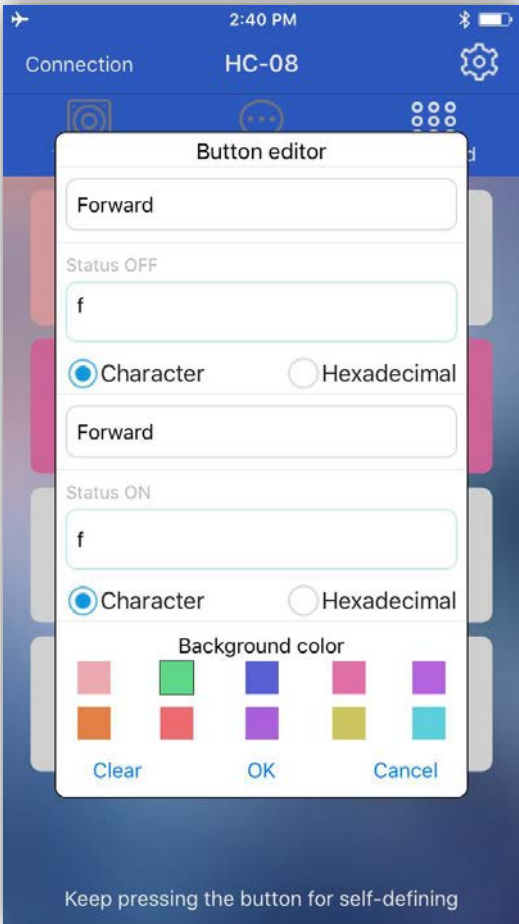
void setup() {
    Serial.begin(9600);
```

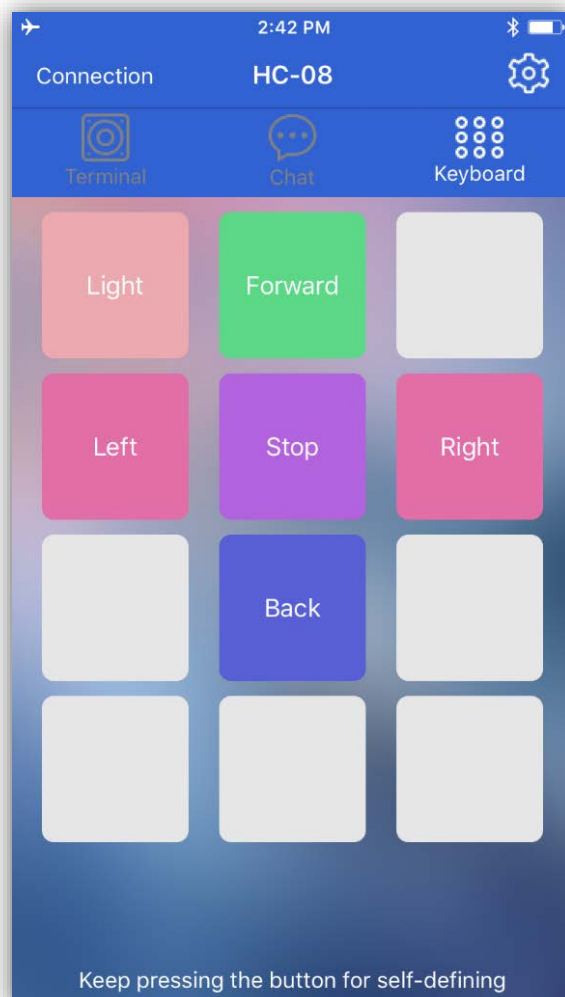
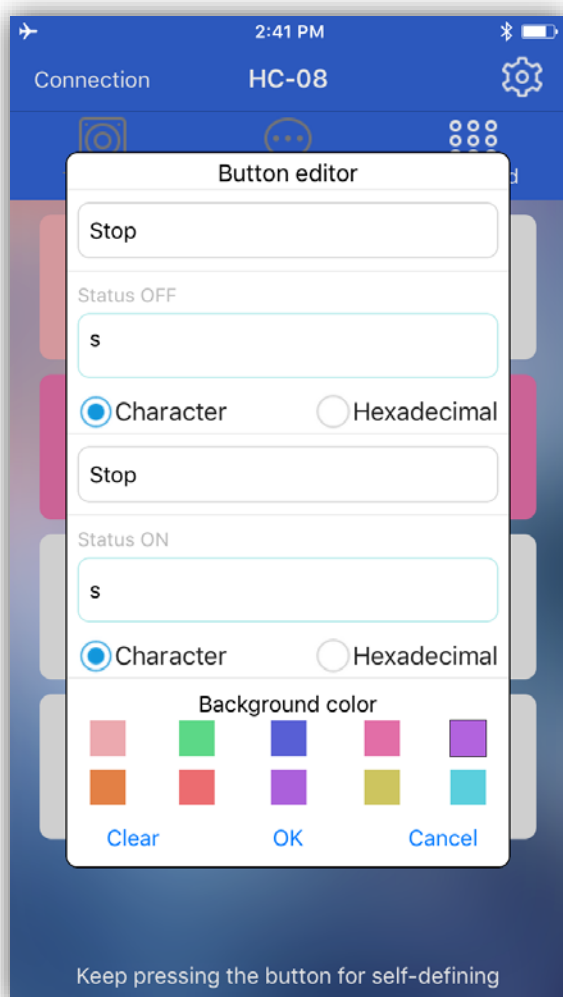
```
pinMode(LED, OUTPUT);
pinMode(IN1,OUTPUT);
pinMode(IN2,OUTPUT);
pinMode(IN3,OUTPUT);
pinMode(IN4,OUTPUT);
pinMode(ENA,OUTPUT);
pinMode(ENB,OUTPUT);
stop();
}

void loop() {
  getstr = Serial.read();
  switch(getstr){
    case 'f': forward(); break;
    case 'b': back(); break;
    case 'l': left(); break;
    case 'r': right(); break;
    case 's': stop(); break;
    case 'a': stateChange(); break;
    default: break;
  }
}
```

車両の電源を入れ、地上に置きます。

モバイル APP を開き、次のようにパラメータを設定します。





今我々は、Bluetooth で車を制御し、それで遊ぶことができます。