

Leçon 2 – Voiture bluetooth

Point abordés dans cette section

Il est très important et si cool de contrôler votre voiture sans fil dans un certain espace lorsque nous apprenons l'Arduino, alors dans la leçon, nous vous apprendrons à contrôler une voiture par Bluetooth.

Apprentissage

- *Apprenez à utiliser le module Bluetooth et l'appareil Bluetooth*
- *Apprenez comment contrôler le véhicule via Bluetooth*
- *Écrivez des programmes pour implémenter cette fonction*

Préparation

- *Un véhicule (équipé d'une batterie)*
- *Un câble USB*
- *Un module Bluetooth*
- *Un iPhone ou une tablette*

I – Le module Bluetooth

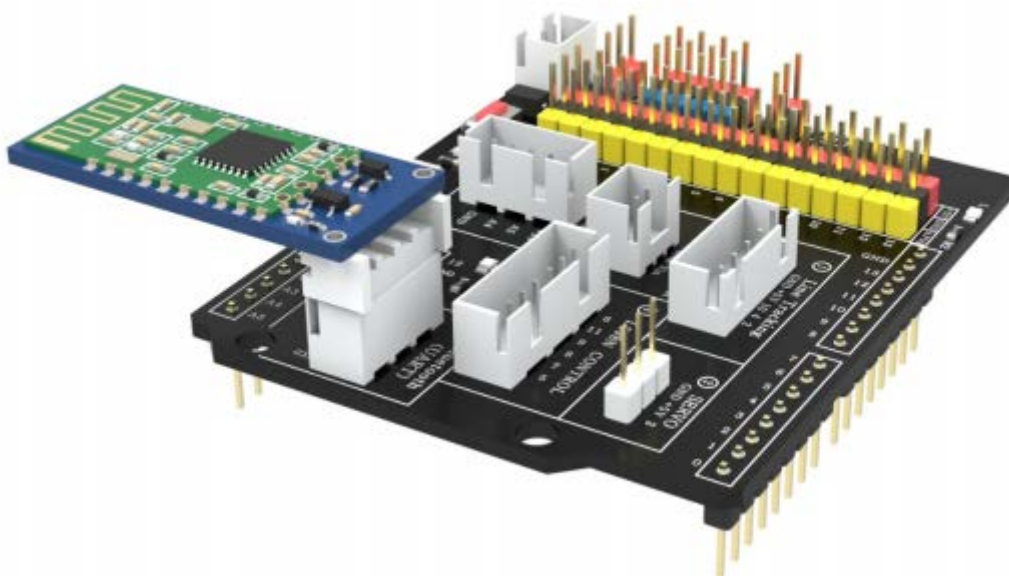


La description du module Bluetooth:

1. Adoptez la puce Bluetooth principale de TI, protocole standard de Bluetooth V4.0.
2. La tension de travail analogique du port série est de 3.3V.
3. Les utilisateurs peuvent régler le débit en bauds 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.
4. La dimension du module de clé est de 28 mm x 15 mm x 2,35 mm.
5. Courant électrique de travail: 40mA.
6. Courant électrique de dormance: moins de 1mA.
7. Utilisé pour le système de navigation GPS, le système de lecture de gaz hydroélectrique, le système industriel de contrôle des mines sur le terrain.
8. Peut être connecté à un ordinateur portable Bluetooth, un ordinateur avec adaptateur Bluetooth, PDA, etc.

Il s'agit du diagramme schématique du module Bluetooth connecté à la carte contrôleur UNO:

In the experiment we will connect it to UNO board via expansion board V5.



II – Mise en route de l'application Bluetooth

Avant de commencer, connectez le module Bluetooth HC-08 à la carte d'extension et allumez l'alimentation.

STEP1: installez l'application.

Pour Android

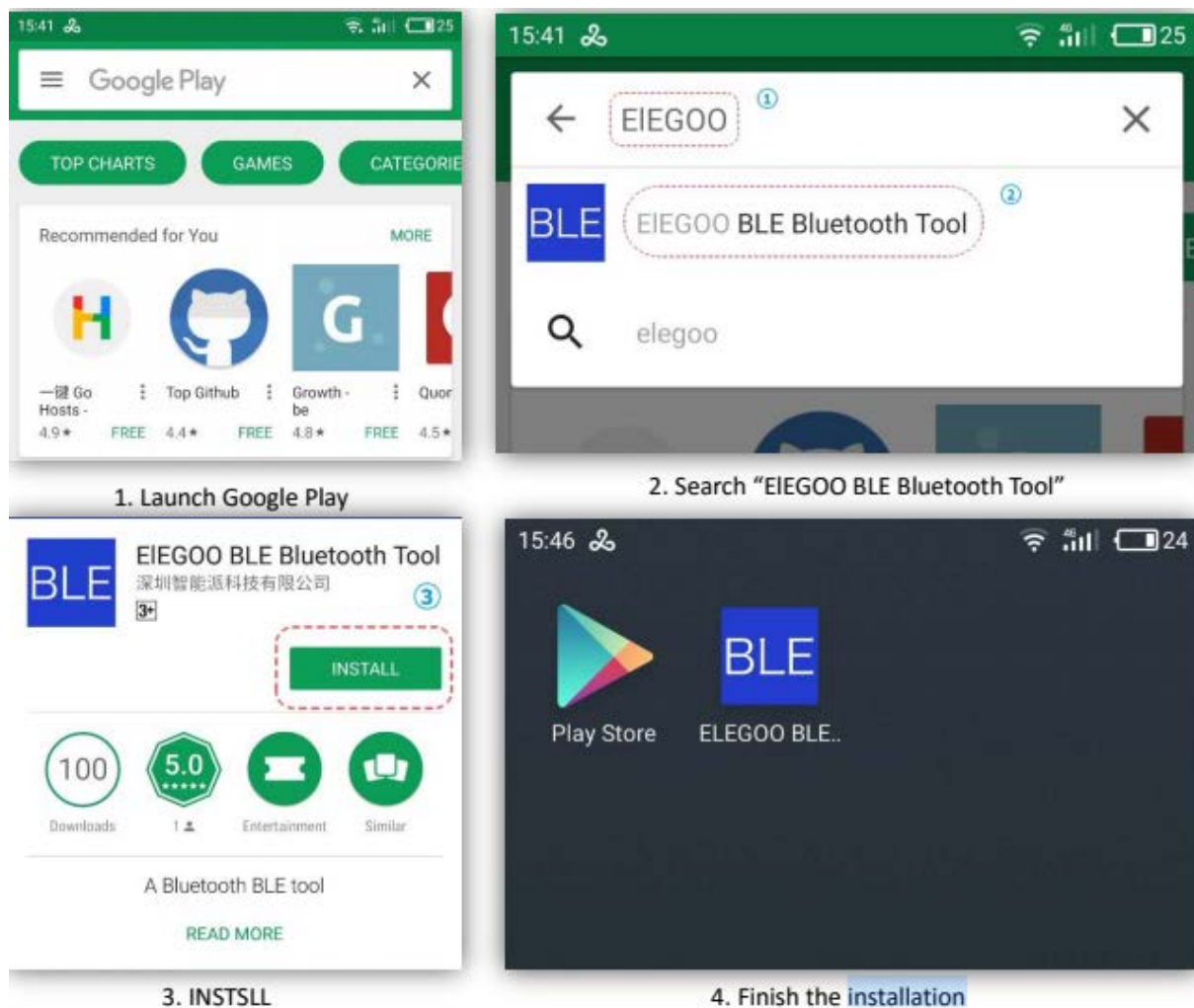
Il existe deux façons d'installer l'application.

1. Copiez le fichier "EIEGOO BLE Tool.apk" sur les produits Android et installez-le.

Elegoo Smart Robot Car Kit V3.0_20170524 > Lesson 2 Bluetooth Car

Name	Date modified	Type
bluetooth_blink	2017/5/11 11:30	File folder
bluetooth_car	2017/6/9 14:48	File folder
EIEGOO BLE Tool.apk	2017/1/17 9:20	Droid4X apk file

2. Recherchez "EIEGOO BLE Bluetooth Tool" dans Google Play Store et installez-le.



Pour iOS

Recherchez "ELEGOO BLE Tool" dans Apple APP Store et installez-le.



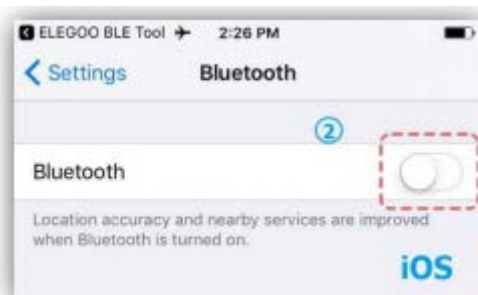
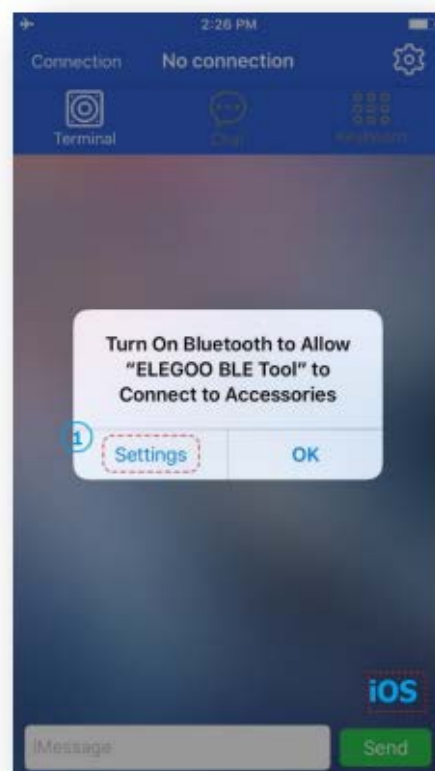
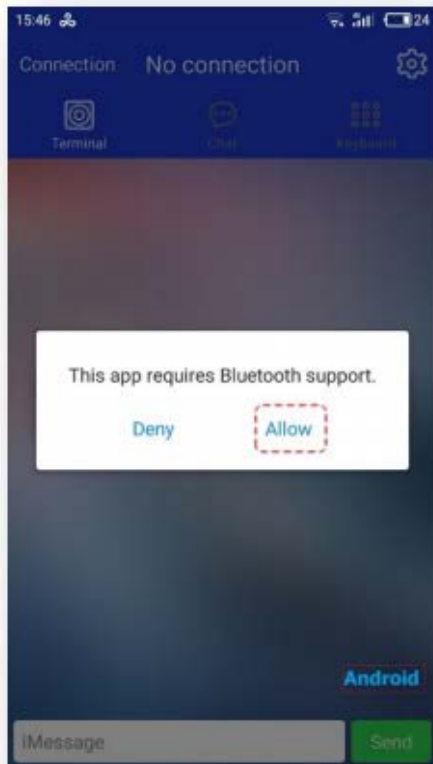
STEP2: Paramètres de l'application.

Lancer l'application

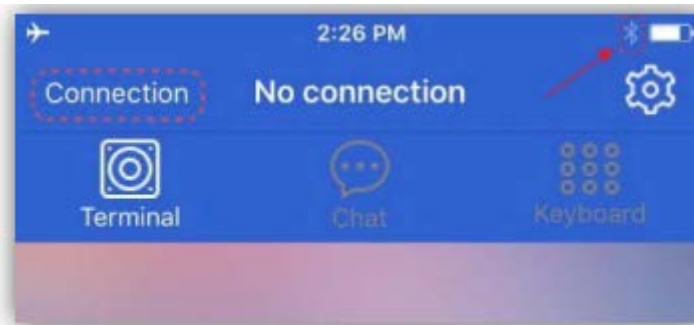


Si vous n'ouvrez pas votre Bluetooth, cela apparaîtra comme un coup et vous conseillera d'activer la fonction Bluetooth.

Pour Android, cliquez sur "Autoriser" qui va ouvrir le bluetooth automatiquement. Pour iOS, cliquez sur "Paramètres" qui va passer à l'interface de configuration bluetooth, vous devez activer le bluetooth, puis revenir à l'interface d'application "ELEGOO BLE Tool".



Après avoir activé Bluetooth, l'icône bluetooth apparaît sur l'interface de l'APP. Cliquez sur "Connexion".



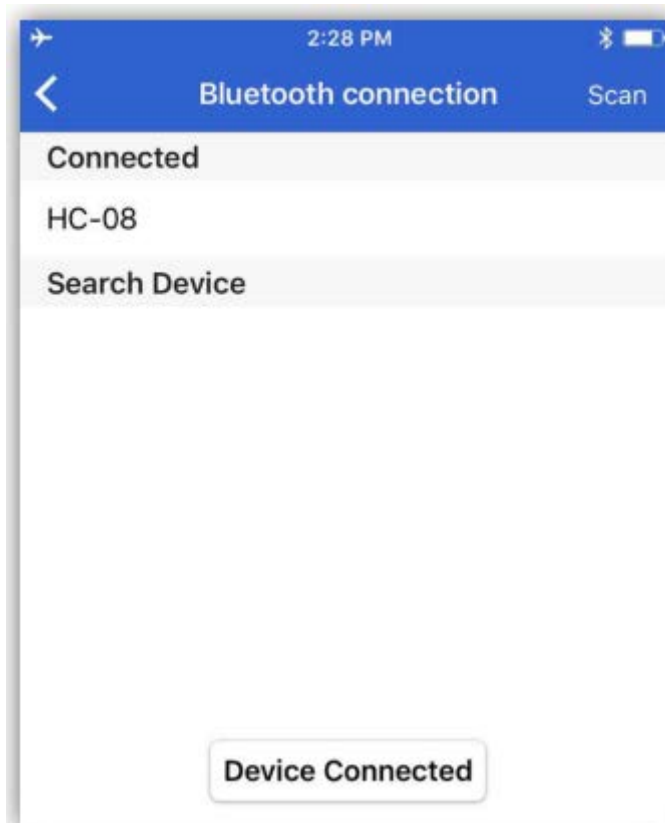
Scanner le signal Bluetooth.



Ensuite, votre téléphone recherche des équipements Bluetooth à proximité. L'appareil HC-08 apparaît.



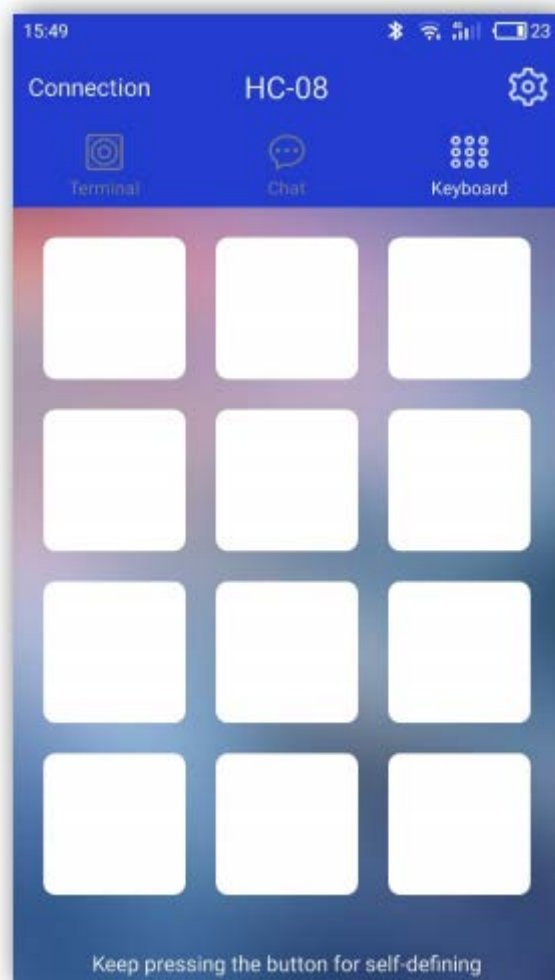
Cliquez sur le nom Bluetooth "HC-08", lorsque la connexion est réussie, l'écran s'affiche "Device Connected".



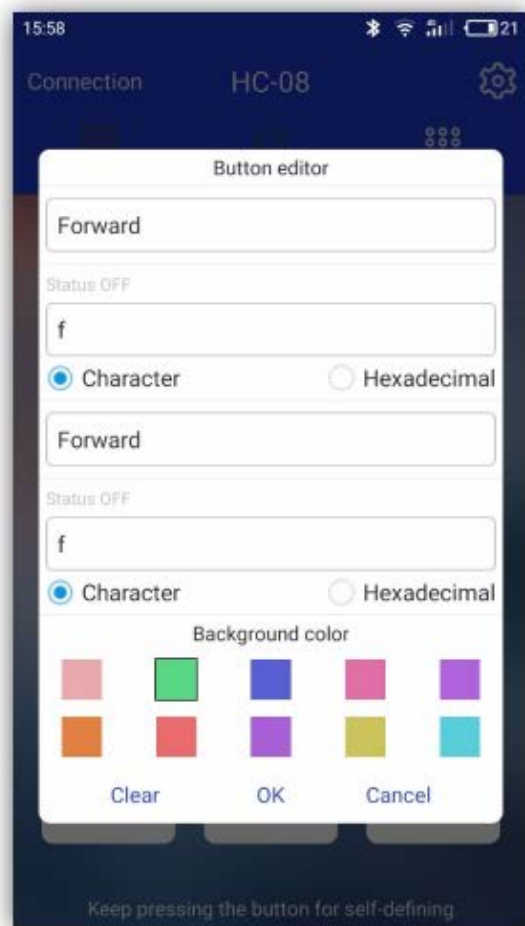
Renvoie l'interface principale de l'application. Après la connexion Bluetooth, le nom du périphérique s'affiche à l'écran.



Ensuite, nous pouvons glisser l'écran vers le côté droit par notre doigt et nous pouvons obtenir le motif clé ci-dessous:

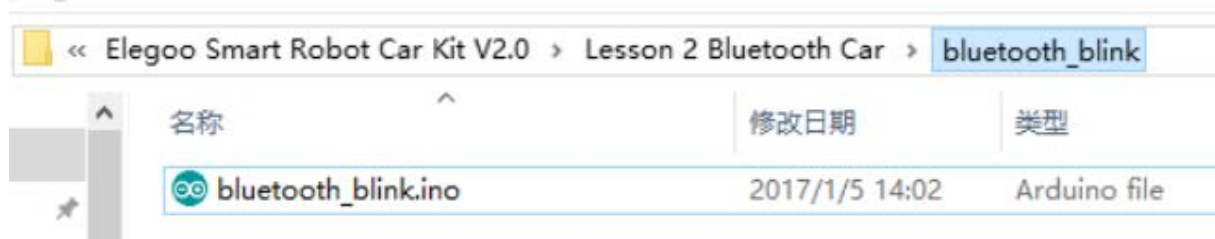


Enfin, nous définissons chaque bouton, nous allons "avancer" par exemple, voir ci-dessous, et les valeurs-clés restantes sont définies de la même manière.



III – Test

Ouvrez le fichier de code dans le chemin "\\ Elegoo Smart Robot Car Kit V3.0 \ bluetooth blinc \ bluetooth blink.ino" et téléchargez le programme sur le panneau UNO.

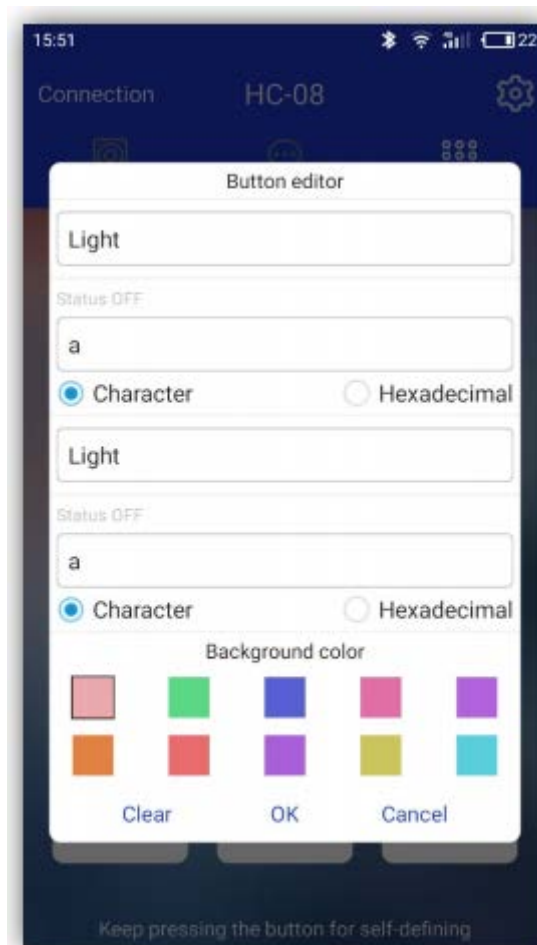


```
//www.elegoo.com
#define LED 13 //Define 13 pin for LED
bool state = LOW; //The initial state of the function is defined as a low level
char getstr; //Defines a function that receives the Bluetooth character
void setup() {
  pinMode(LED, OUTPUT);
  Serial.begin(9600);
}
//Control LED sub function
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();
  }
}
```

Débranchez-le de l'ordinateur, puis allumez l'alimentation de la voiture. **(CONSEILS: Le module Bluetooth doit être retiré lorsque vous téléchargez le programme, ou il sera impossible de télécharger le programme.)**

Ouvrir l'application

Après avoir connecté le téléphone à la voiture via Bluetooth, nous définissons les données ci-dessous:



Après la configuration, appuyez sur ce bouton. Vous constaterez que la lumière sur le panneau UNO change avec le commutateur.

Le code

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

Le but de ce bloc de code est de régler le débit en bauds de la carte de contrôle UNO en 9600 et d'ouvrir le port série. De cette façon, ils peuvent communiquer les uns avec les autres, car la vitesse de transmission originale du module Bluetooth est de 9600.

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

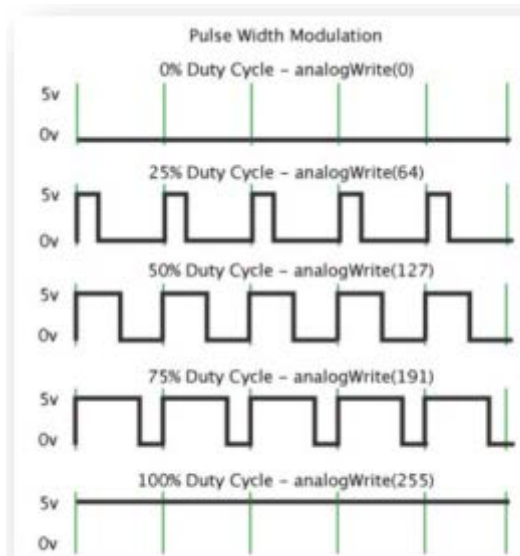
Cette fonction est exécutée à plusieurs reprises dans la fonction de circulation. Il va d'abord lire les données du port série, puis vérifier les données. Si elle répond à la condition, elle exécutera la sous-fonction correspondante. Par exemple, si elle lit la lettre 'a' du port série, elle exécutera la sous-fonction responsable de la mise en marche / arrêt de la lumière LED.

IV – Faire une voiture Bluetooth

Lorsque la voiture tourne à gauche ou à droite, il n'est pas nécessaire de régler la vitesse trop rapidement. Au contraire, nous devons contrôler la vitesse de la voiture. Mais comment contrôler?

La réponse est PWM.

PWM est l'abréviation de "Pulse Width Modulation", s'appelle la modulation d'impulsion, est une technologie efficace pour contrôler le circuit analogique avec la sortie numérique du microprocesseur, la voiture est utilisée pour changer la vitesse du moteur en modifiant le cycle de fonctionnement d'une onde carrée. En d'autres termes, reliez le circuit entre les deux côtés du moteur en permanence, l'interrupteur du fonctionnement du moteur, le moteur ne sera pas éteint lorsque l'alimentation est éteinte en raison de la vitesse rapide. Nous pouvons donc contrôler la vitesse de la voiture si nous contrôlons la valeur spécifique du temps d'alimentation et de la mise hors tension. La vitesse de la voiture sera maximale lorsque le circuit est éteint. La vitesse de la voiture sera minimale si le circuit s'arrête. La vitesse de la voiture sera médiane dans la moitié du temps. PWM est une technologie pour obtenir une quantité analogique grâce à la méthode numérique. Une onde carrée est formée par un contrôle numérique, le signal d'onde carrée ne comporte que deux états de marche et de sortie (c'est-à-dire un nombre élevé de broches numériques). Assurez-vous que la tension change de 0 à 5V en contrôlant la durée de marche et l'arrêt de la durée spécifique. L'heure occupée de l'activation (c'est le niveau élevé dans l'académie) est appelée largeur d'impulsion, de sorte que PWM est également appelé modulation de largeur d'impulsion. Apprenons sur PWM à travers cinq ondes carrées ci-dessous.



La ligne verte verticale ci-dessus représente une période d'onde carrée. La valeur écrite dans chaque `analogWrite` (broche, valeur) correspond au pourcentage, le pourcentage est également appelé Cycle de service, se référer au pourcentage obtenu à partir de la valeur spécifique entre le niveau élevé de la durée et le temps de bas niveau dans une période.

Sur la figure, de haut en bas, la première vague carrée, le cycle de service est de 0%, la valeur correspondante est 0. Le courant du circuit de sortie est minimum, le moteur reste étanche. Le temps de durée plus long, plus le moteur à courant de circuit plus grand obtient, plus la vitesse est rapide. Ainsi, le cycle de service final est de 100%, la valeur correspondante est de 255, le moteur tourne en pleine vitesse. 50% est une vitesse de rotation hyponétique moyenne, 25% est relativement plus lent, même si elle ne peut pas démarrer (le courant du circuit est relativement important pour démarrer le moteur en raison de la friction statique). PWM est principalement utilisé

pour régler la lumière de la LED et faire pivoter la vitesse du moteur, la vitesse de la roue contrôlée par le moteur est facilement contrôlée.

L'avantage de PWM peut être plus réfléchi lorsque vous jouez avec des voitures Arduino.

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

analogWrite () est utilisé pour écrire une valeur analogique de 0 à 255 pour les ports PWM. Ce que vous devez noter, c'est que analogWrite () n'est utilisé que pour les broches numériques avec fonction de PWM. Les broches avec fonction de PWM dans UNO ne sont que des broches numériques de 3,5,6,9,10,11.

La vitesse de notre voiture est contrôlée en connectant pin5 et pin6 de ENA et ENB. Le programme ci-dessous a défini une fonction numérique int carSpeed = 150; La vitesse est contrôlée dans le programme ci-dessous, de sorte que vous pouvez contrôler la vitesse par vous-même.

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

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

Après avoir appris les connaissances de base, nous allons télécharger le programme comme ci-dessous à la voiture, ouvrez le fichier de code dans le chemin "\\ Elegoo Smart Robot Car Kit V3.0 \ bluetooth_car \ bluetooth_car.ino" puis téléchargez le programme dans le contrôle UNO planche.

```
//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(){
```

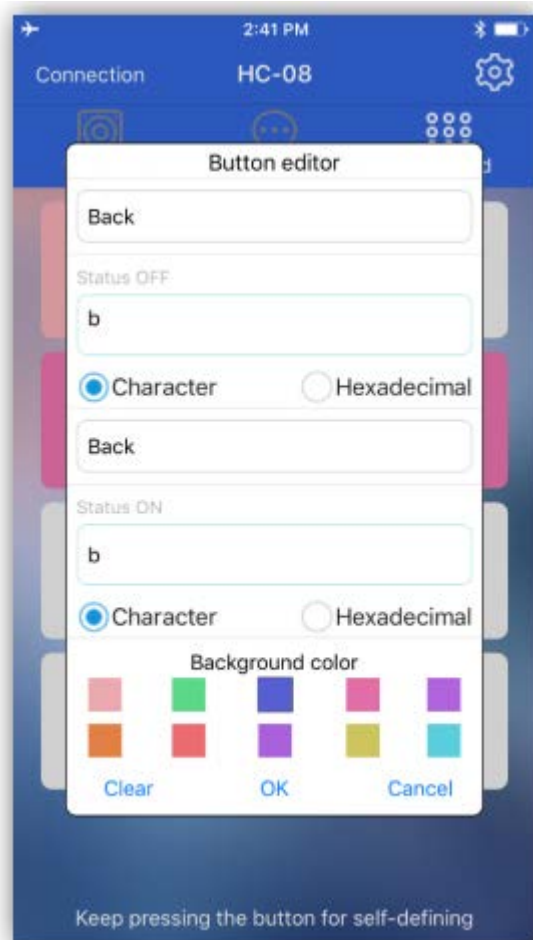
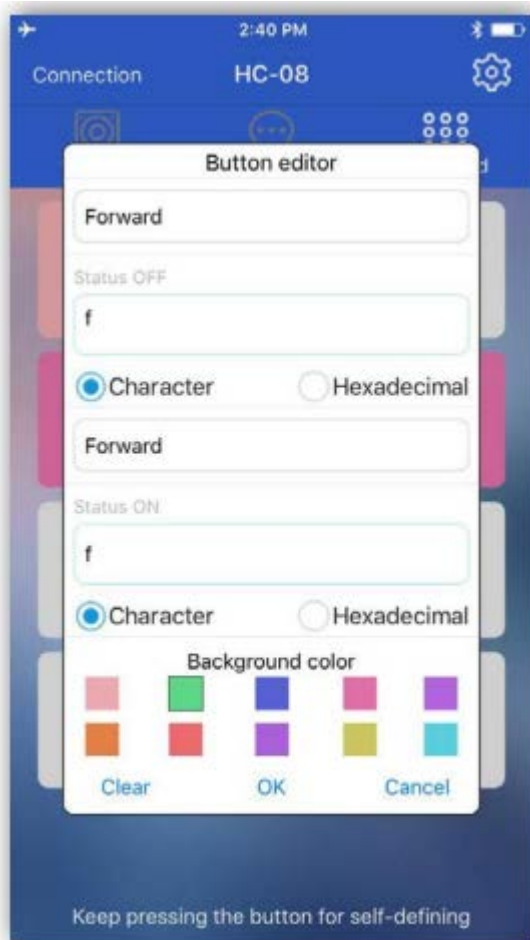
```
state = !state;
digitalWrite(LED, state);
}

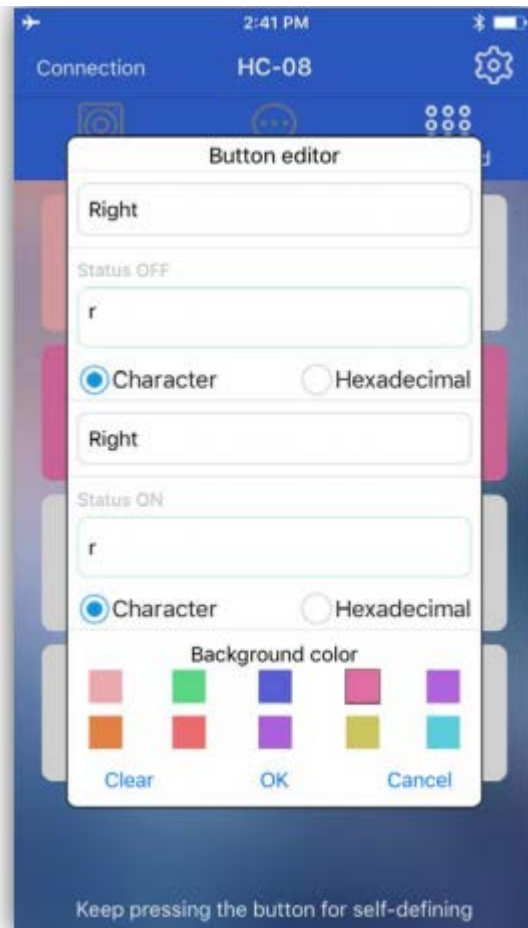
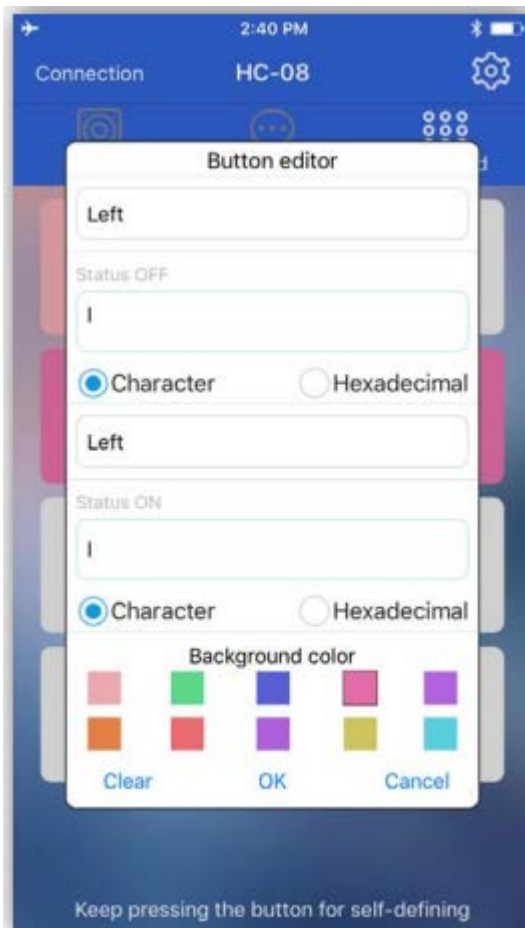
void setup() {
  Serial.begin(9600);
  pinMode(LED, OUTPUT);
  pinMode(IN1,OUTPUT);
  pinMode(IN2,OUTPUT);
  pinMode(IN3,OUTPUT);
  pinMode(IN4,OUTPUT);
  pinMode(ENA,OUTPUT);
  pinMode(ENB,OUTPUT);
  stop();
}

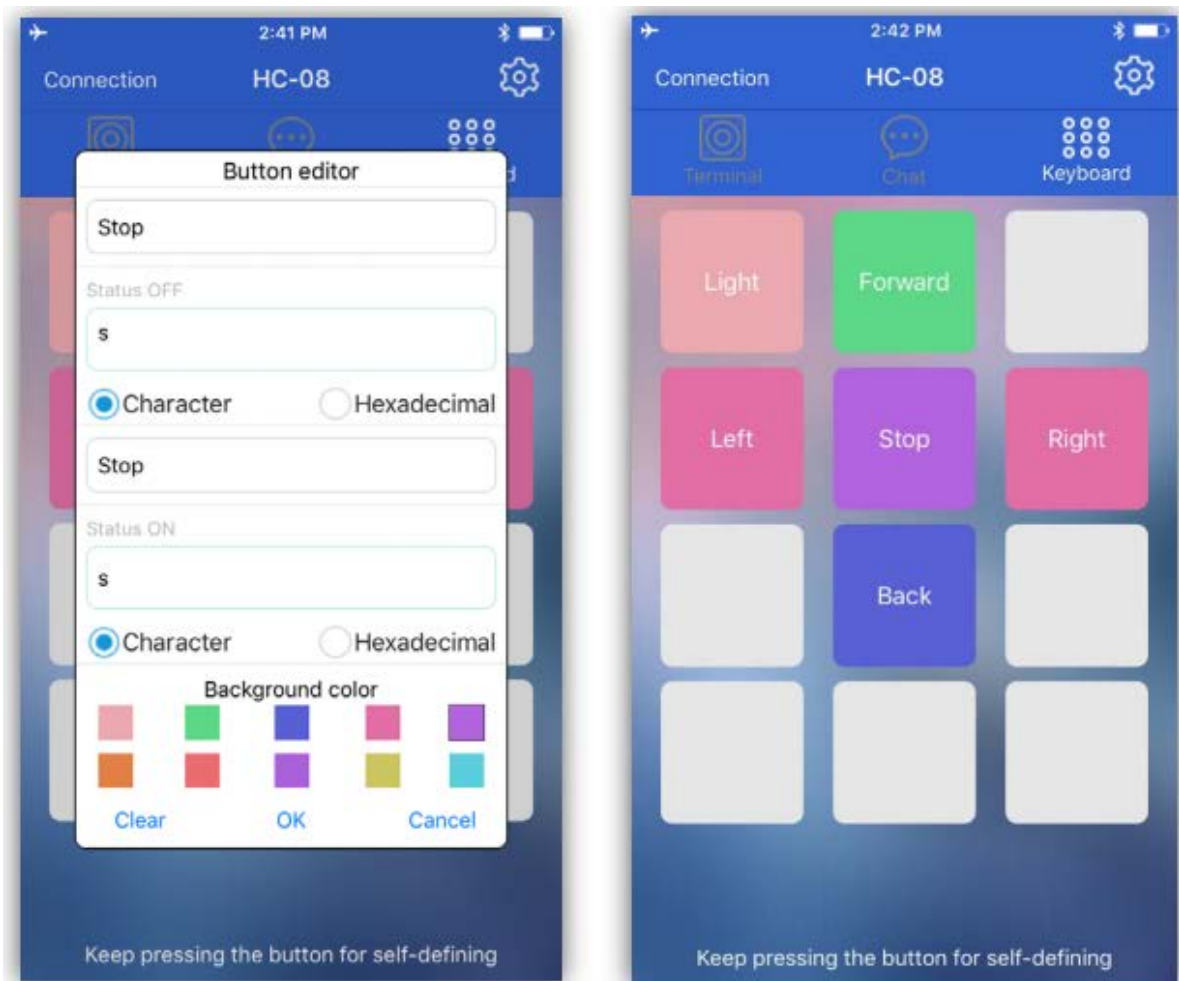
void loop() {
  getstr = Serial.read();
  switch(getstr){
    case 'f': forward(); break;
    case 'b': back(); break;
    case 'l': left(); break;
    case 'r': right(); break;
    case 's': stop(); break;
    case 'a': stateChange(); break;
    default: break;
  }
}
```

Allumez l'alimentation du véhicule et mettez-le au sol.

Ouvrez l'application mobile et configurez les paramètres comme suit.







Maintenant, nous pouvons contrôler la voiture par Bluetooth et jouer avec elle.