

## **Projet Systèmes Microprocesseurs**

### **- Le contrôle de la lumière sans fil -**

#### 1. Les spécifications fonctionnelles

On désire d'avoir trois grandes fonctions pour ce projet :

- La première fonction consiste dans le fait que module Bluetooth marche/fonctionne quand il est alimenté au courant. Il doit être disponible pour que le smartphone puisse se connecter à lui
- La deuxième fonction porte sur le contrôle de la lumière (dans ce cas, le contrôle des LEDs). On décide d'avoir une fonction pour allumer tous les LEDs, une pour les éteindre et la dernière pour le contrôle de chaque LED individuellement
- Avec l'aide d'une application mobile, on a créé une petite interface pour que l'utilisateur manipule l'application plus facile

#### 2. Les matériaux utilisés

Pour ce projet, on a utilisé les suivants matériaux : Arduino Uno, 5 LEDs des couleurs différentes, 5 résistances de 220  $\Omega$ , le module Bluetooth HC-05, des câbles, un smartphone et un breadboard.

#### 3. Les schémas fonctionnels

Tout d'abords, la figure 1 représente le schéma fonctionnel de ce projet. Elle est faite avec le logiciel : Tinkercad. On peut observer que les LEDs sont liés aux pins 9 jusqu'à 13 de la carte et au terre (« GND ») en ajoutant les résistances pour éviter la destruction des LEDs.

Et dans la figure 3, on remarque comment il faut relier le module Bluetooth à Arduino. Le pin « STATE » de module au pin 4 de Arduino, le RX de module au pin 3 de Arduino, le TX de module au pin 2 et aussi à l'alimentation de 3,3V et au GND. Ce module Bluetooth est construit pour l'utilisation avec des tensions réduits, par exemple 3,3V. Alors il faut être attentif comment on le connectait à la carte Arduino pour ne pas le détruire.

Et dans la figure 2, on a créé la petite interface pour le smartphone en utilisant l'application mobile: Bluetooth Electronics.

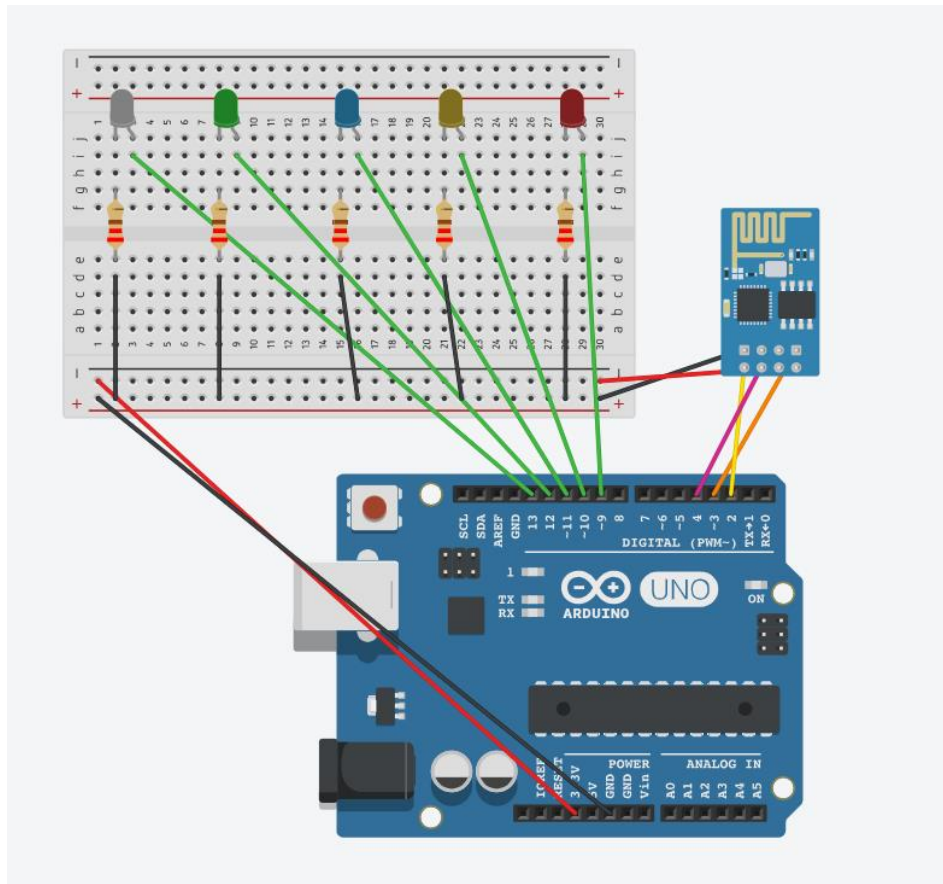


Figure 1: Le schéma fonctionnel du projet

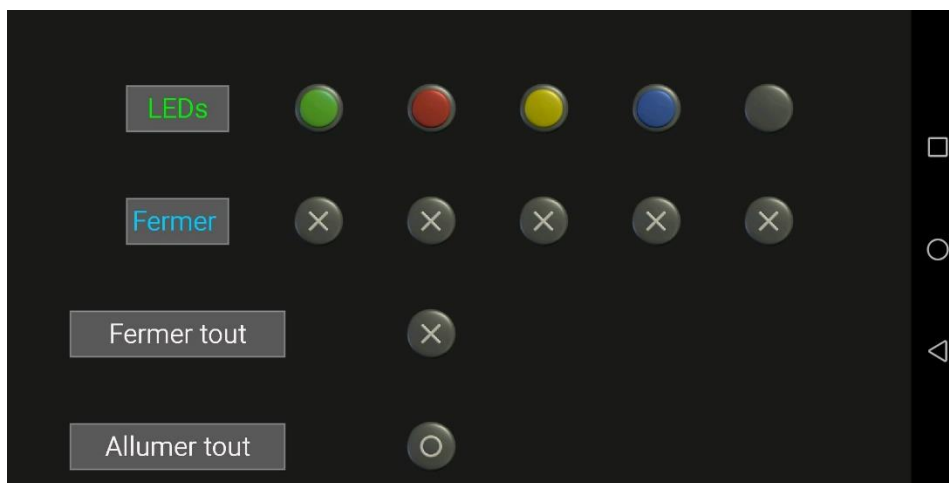


Figure 2: L'interface pour utilisateurs

## HC-05 WITH CONFIRM CONNECTION

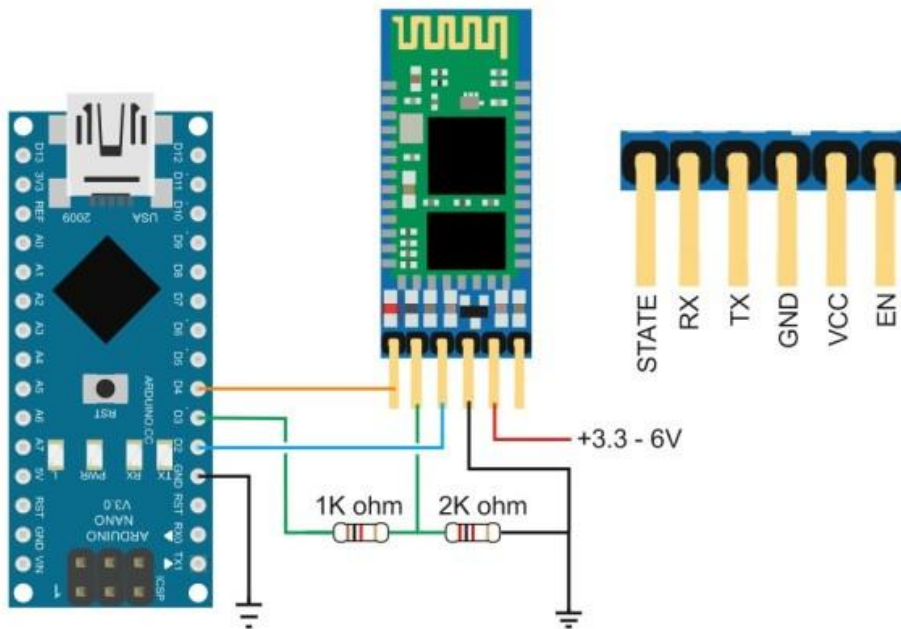


Figure 3: La modalité de connecter le module Bluetooth à Arduino

### 4. La présentation

En partant du code présenté ici <http://www.martyncurrey.com/hc-05-fs-040-state-pin/> on est arrivé à contrôler 5 LEDs par l'intermédiaire de smartphone.

Tout d'abord, on a utilisé la bibliothèque « SoftwareSerial » pour pouvoir utiliser le Serial de Arduino et le module Bluetooth. Le Serial de Arduino a été utilisé pour voir ce que l'utilisateur envoyait et pour afficher un message de succès. Par exemple, s'il veut allumer le LED vert, dans le Serial de Arduino on peut voir le message « Vert on » ou s'il veut l'éteindre, dans le Serial on voit « Vert off ».

« *SoftwareSerial BTserial(2, 3);* » par cette ligne on a créé un objet de type SoftwareSerial. Le paramètre « 2 » représente le pin par lequel on recevait des données (le pin RX) et le deuxième paramètre « 3 » est le pin par lequel on envoyait des données (le pin TX). La communication sera permise par l'appel de la fonction « *begin* ».

Le pin « state » du module Bluetooth est très important car avec lui on peut vérifier s'il y a une connexion ou pas. La valeur default pour lui est « LOW » quand il n'est pas connecté et « HIGH » quand il est connecté à un autre dispositif. Grace a ça, dans la fonction « *setup* » on testait avec un while si le module a établi une connexion. Si la connexion a été établie, alors avec la fonction « *begin* » on permet la communication entre ces deux dispositifs.

Dans la fonction « loop » on a mis le switch qui contrôle les LEDs.

En appuyant sur un bouton de l'interface (soit pour allumer, soit pour éteindre), elle transmet un caractère via la connexion Bluetooth entre le smartphone et le module connecté au breadbord au Arduino qui l'interprète dans cette boucle. C'est-à-dire, la carte va prendre la décision d'allumer ou d'éteindre le / les LED(s) en fonction du caractère envoyé par l'utilisateur. Par exemple : pour allumer le LED rouge, l'utilisateur appuiera sur l'étiquette rouge dans l'interface, elle envoie le caractère « r » et Arduino allume le LED rouge.

De plus, on a testé l'échange entre Arduino (via le module Bluetooth) et le portable en utilisant une application pour le portable qui s'appelle « Serial Bluetooth Terminal ». Ici, on peut voir ce que l'utilisateur envoie vers Arduino et aussi la réponse de la carte.

L'interface utilisée est très facile de la créer en utilisant l'application pour le mobile : Bluetooth Electronics. Elle a déjà quelques exemples, mais tu peux construire tes interfaces en fonction de ce que tu veux faire.

## 5. La bibliographie

- a. <http://www.martyncurrey.com/hc-05-fs-040-state-pin/>
- b. <https://www.heise.de/developer/artikel/Bluetooth-Auf-den-blauen-Zahn-gefuehlt-3361763.html>
- c. <https://www.youtube.com/watch?v=u470ia7uHxU&index=1&list=PLSWF2RZ1vJMFCdxOAJ4bCxeV5lnWMaWT>
- d. <https://www.tinkercad.com/things/gAKm3cs9lEO-copy-of-practica-15-bluetooth/editel?tenant=circuits>
- e. <http://www.electronicaestudio.com/docs/istd016A.pdf>
- f. <https://www.arduino.cc/en/Reference/SoftwareSerialBegin>