



Préparation des modules Bluetooth HM-10

1. Objectifs :

1. Préparer un module Bluetooth HM-10 comme « peripheral » ou « slave » pour son utilisation avec l'App-IA
2. La préparation des modules ne se fait pas avec mBlock ou Blockly. L'utilisation subséquente peut se faire avec Arduino IDE, mBlock ou Blockly.
3. Bonus : configuration d'un 2^e module comme « central » ou « master » pour connecter deux modules ensemble sans ordinateur (ne fonctionne pas avec l'App-IA). Bonus : configuration d'un 2^e module comme « central » ou « master » pour connecter deux modules ensemble sans ordinateur (ne fonctionne pas avec l'App-IA). Attention : les modules HM-10 ne peuvent pas communiquer avec les modules HC-05. Il faut 2 modules HM-10 ou 2 modules HC-05.

2. Matériel :

- Carte Arduino Uno
- Module Bluetooth HM-10
- Platine d'essai
- Plusieurs fils mâles-mâles
- 1 résistance de 2 k Ω et une résistance de 1 k Ω ou 3 résistances de 1 k Ω
- Ordinateur et application Arduino IDE

3. Déroulement :

- a. Faire le circuit
- b. Installer l'application Arduino IDE au besoin
- c. Ouvrir les commandes AT permettant de configurer le module Bluetooth
- d. Configurer le module Bluetooth.

4. Petite introduction au Bluetooth :

La technologie Bluetooth a énormément évolué depuis ses débuts dans les années 1994. La première norme a été proposée par le fabricant suédois Ericsson¹. Cette norme de communication devait permettre un échange entre les ordinateurs et les téléphones cellulaires. Son utilisation a depuis été largement généralisée. Le nom Bluetooth (littéralement « dent bleue ») provient de la traduction anglaise du nom du roi viking Harald Blâtand ou *Harald à la dent bleue*, qui avait unifié les tribus danoises en un seul royaume. La communication Bluetooth, comme le roi qui lui a prêté son nom, permet d'«unifier» la communication entre divers appareils. L'icône représentant le Bluetooth provient des initiales en alphabet runique du fameux roi viking.

Même si la communication Bluetooth peut sembler relativement aisée à utiliser, les choses se compliquent assez rapidement quand on cherche à en comprendre le fonctionnement sous-jacent. La plus grande difficulté réside dans le fait que les nombreuses normes ont apporté des termes différents, des nouveaux rôles pour les appareils et des façons différentes d'établir la connexion. Sans tenter d'offrir des réponses à toutes les questions qui sont soulevées lorsque l'on travaille avec Bluetooth et Arduino, le présent document, et les autres qui traitent de ce sujet, présente des méthodes pour communiquer à l'aide de Bluetooth avec deux appareils qui ont été testées et qui fonctionnent. Le but étant, par exemple, de pouvoir concevoir des robots commandés à distance à l'aide de deux cartes Arduino.

Il est aussi possible de concevoir une application sur votre téléphone cellulaire et de l'utiliser pour contrôler votre robot. Cependant, ce sujet ne sera pas traité ici. Vous pouvez quand même essayer les différentes applications qui existent déjà, mais cela dépasse les objectifs du présent document.

Il existe plusieurs types de Bluetooth, en fonction des possibilités offertes par la technologie et les différentes normes. On retrouve, dans les grandes lignes, le Bluetooth Classique et le Bluetooth *Low Energie* (BLE). Les deux catégories utilisent des façons différentes pour envoyer l'information, des vitesses de transfert différentes, etc. Elles utilisent toutes les deux des ondes radio dont les fréquences sont autour de 2.4 GHz.

¹La plupart des informations de cette section proviennent de : <https://fr.wikipedia.org/wiki/Bluetooth>

Dans tous les cas, un appareil possédant la connectivité Bluetooth possède une antenne qui sert d'émetteur et/ou de récepteur de signal. La portée du signal Bluetooth dépendra de la puissance de cette antenne. De manière générale, on peut s'attendre à une portée d'environ 10 mètres. Le Wi-Fi, les autres appareils Bluetooth et même les fours à micro-ondes (en fonction...) peuvent parfois causer des interférences car ils utilisent des ondes radio à des fréquences similaires.

Choix et rôles des deux cartes (explications très simplifiées des principes de base)

Lorsque l'on utilise habituellement la communication Bluetooth entre deux appareils, c'est généralement pour brancher un périphérique (souris, montre intelligente, clavier, ...) à un ordinateur.

L'ordinateur va rechercher les périphériques disponibles et afficher une liste. Il peut aussi se connecter ou s'associer à plusieurs périphériques différents.

Le périphérique « annonce », quant à lui, son existence aux autres appareils avec connectivité Bluetooth qui sont situés à proximité. Une fois apparié ou associé, il ne se connecte qu'avec un seul appareil.

Dans les réseaux Bluetooth plus simples (*piconets*), l'appareil qui peut s'associer à plusieurs autres se nomment *Master* (maître). Les appareils qui ne se branchent qu'à un seul se nomment *Slave* (esclave). Cependant, des réseaux plus complexes peuvent être formés (*Mesh*) si la technologie le permet, où l'on peut faire l'interconnexion entre plusieurs appareils. Cependant, ce ne sont pas tous les appareils Bluetooth qui permettent ceci. Ici, nous nous contenterons d'utiliser l'ordinateur comme l'appareil central (« *Master* ») deviendra le et le module Bluetooth sera le « peripheral » (« *Slave* »).

Une fois la connexion effectuée, la communication peut être faite dans les deux directions. Cela signifie que l'ordinateur et l'appareil peuvent envoyer une information (écriture) ou demander une information (lecture) à l'autre. Cependant, il peut arriver que l'écriture ou la lecture soit restreinte pour certains appareils ou à certaines valeurs. Ce ne sera pas le cas avec les modules HM-10.

À noter que les modules HM-10 utilisent la norme Bluetooth 4.0 (Bluetooth Low Energy). Ils pourraient donc ne pas être compatibles avec des modules Bluetooth 2.0 (Bluetooth classique) ou de norme plus récente. Il faut toujours vérifier la compatibilité des appareils

Bluetooth, car elle n'est pas garantie, même s'il s'agit de Bluetooth dans tous les cas. Quoiqu'il en soit, les modules HM-10 peuvent communiquer les uns avec les autres.

Ce ne sont pas tous les appareils Bluetooth qui ne permettent pas la connexion avec des appareils dont la norme est différente. Cependant, cela peut arriver et il faut en être conscient lorsque l'on fait des choix technologiques.

Avant d'utiliser les modules, il faut les préparer à l'aide des commandes AT.

5. Le circuit :

Les modules HM-10 ont en général 6 broches : STATE, VCC, GND, TXD, RXD et BRK.

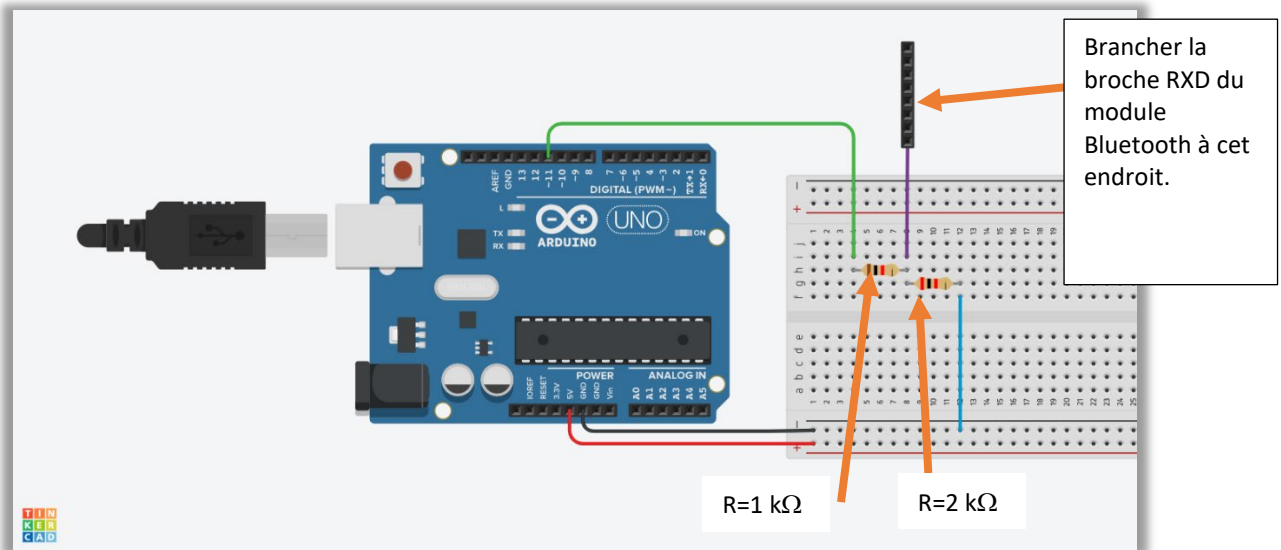
Pour l'instant, nous n'utiliserons pas la broche STATE, donc elle ne sera branchée à rien.

La broche VCC devra être branchée à une source de tension. Dans notre cas, la broche VCC sera branchée à la borne +5V de la carte Arduino.

La broche GND sera branchée sur une broche GND de la carte Arduino.

Certains modules HM-10 pourraient demander une alimentation 3.3 V. Toujours vérifier les informations du fabricant pour être certain d'utiliser les bonnes broches.

En fonction des recommandations du fabricant, les modules HM-10 doivent être utilisés avec un diviseur de tension pour la broche RXD. Dans ce cas, faire le circuit suivant :



On peut utiliser deux résistances de 1 kΩ en série au lieu d'une résistance de 2 kΩ.

Il faudra brancher la broche TXD sur la broche digitale 9 de la carte Arduino.

Il faudra brancher la broche RXD sur la broche digitale 8 de la carte Arduino (ou sur le diviseur de tension qui lui est branché dans la broche 8).

En résumé, les connexions suivantes doivent être faites pour pouvoir configurer un module en commandes AT :

HM-10	Arduino
VCC	+5V ou + 3.3V selon le fabricant
GND	GND
TXD	Broche digitale 8 ou diviseur de tension
RXD	Broche digitale 9
BRK	Rien
STATE	Rien

Une fois que les modules seront configurés, on pourra utiliser une bibliothèque, SoftwareSerial, qui permettra de faire la communication série avec le module Bluetooth à partir de n'importe quelle broche digitale de la carte Arduino, sauf 0 et 1. Il est aussi techniquement possible² de brancher les modules dans les broches digitales 0 et 1, qui

² Je n'ai personnellement pas testé ce branchement avec les modules HM-10. Il est préférable d'utiliser la bibliothèque SoftwareSerial avec les modules Bluetooth pour éviter les conflits potentiels.

jouent déjà le rôle de Rx et Tx et d'utiliser les fonctions Serial de Arduino. Cependant, ces broches sont aussi utilisées lors de la communication via le port USB entre la carte et l'ordinateur. Il peut alors arriver des conflits.

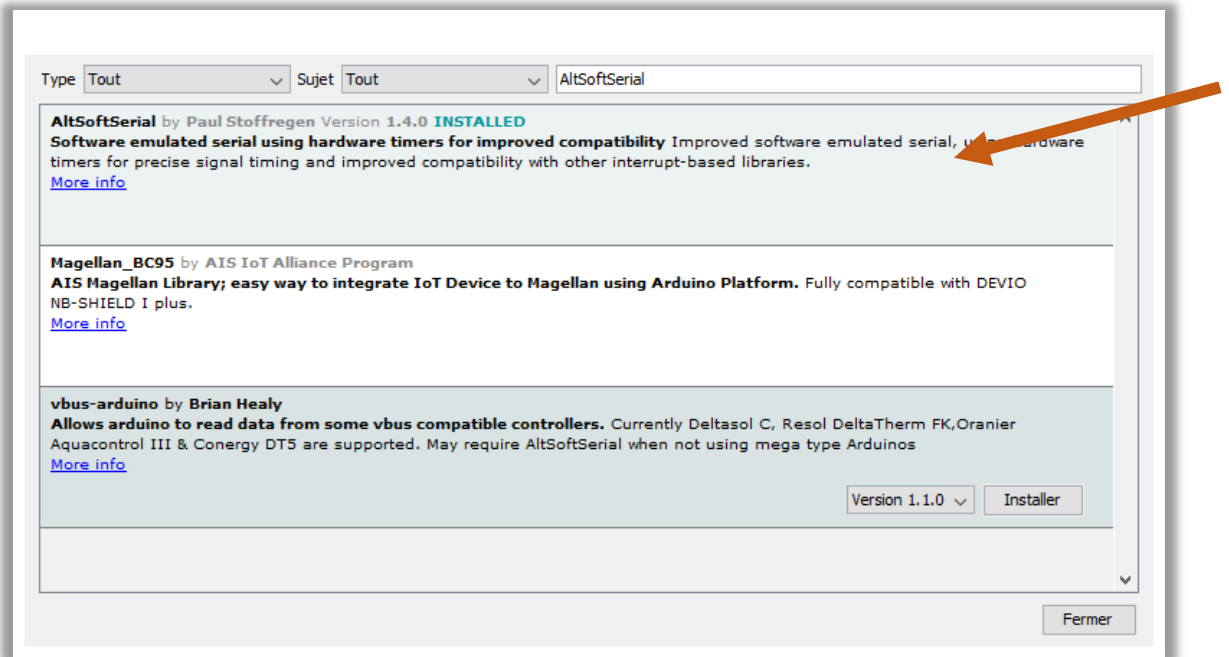
À noter qu'il est toujours préférable de vérifier les branchements recommandés par le fabricant du module.

Pour les modules HM-10, il existe différentes versions du *firmware* (programme interne du module). Cela ne devrait poser de problèmes lors de l'utilisation. Il faut cependant être conscient que les commandes AT peuvent être différentes. On peut se référer aux références du fabricant en cas de problèmes. De plus, la version 7 du *firmware* utilise une vitesse de communication de 115200 bauds par défaut, alors que les versions précédentes utilisent 9600 bauds. On peut aussi changer cette vitesse en la programmant, mais il faut utiliser la bonne vitesse pour accéder aux commandes AT.

6. Configuration du module :

Configuration du module par Commandes AT :

1. Brancher le module Bluetooth à la carte Arduino avec le diviseur de tension, brancher la carte Arduino à l'ordinateur et ouvrir l'application Arduino IDE.
2. On doit ensuite s'assurer que le bon type de carte et le bon port de communication sont utilisés. Dans l'onglet *Outils*, modifier le *Type de carte* dans la liste déroulante si « Arduino/Genuino UNO » n'est pas sélectionné. Sélectionner aussi le bon port de communication.
3. Il faut installer la bibliothèque « AltSoftSerial » dans Arduino IDE :
 - a. Dans Arduino IDE, dans l'onglet *Outils*, choisir *Gérer les bibliothèques*. Dans la fenêtre qui s'ouvre, écrire AltSoftSerial dans l'emplacement pour la recherche. Il faut toujours un certain temps pour que la recherche se fasse, ici, peu importe l'ordinateur. Cliquer sur Installer si la bibliothèque n'est pas déjà installée. Il en faut faire cette étape qu'une seule fois.



4. Copier le programme suivant dans l'application Arduino IDE en remplaçant tout le code existant. Ce programme provient du site : <http://www.martyncurrey.com/hm-10-bluetooth-4ble-modules/>

```
// SerialIn_SerialOut_HM-10_01
//
// Uses hardware serial to talk to the host computer and AltSoftSerial for
// communication with the bluetooth module
//
// What ever is entered in the serial monitor is sent to the connected device
// Anything received from the connected device is copied to the serial monitor
// Does not send line endings to the HM-10
// Pins
// BT VCC to Arduino 5V out.
// BT GND to GND
// Arduino D8 (SS RX) - BT TX no need voltage divider
// Arduino D9 (SS TX) - BT RX through a voltage divider (5v to 3.3v)

#include <AltSoftSerial.h>
AltSoftSerial BTserial;
// https://www.pjrc.com/teensy/td_libs_AltSoftSerial.html

char c=' ';
boolean NL = true;
```

```

void setup()
{
  Serial.begin(9600);
  Serial.print("Sketch: "); Serial.println(__FILE__);
  Serial.print("Uploaded: "); Serial.println(__DATE__);
  Serial.println(" ");

  BTserial.begin(9600);
  Serial.println("BTserial started at 9600");
}

void loop()
{
  // Read from the Bluetooth module and send to the Arduino Serial Monitor
  if (BTserial.available())
  {
    c = BTserial.read();
    Serial.write(c);
  }

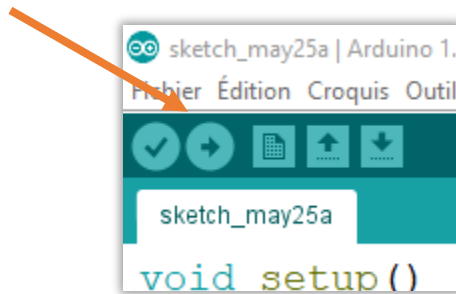
  // Read from the Serial Monitor and send to the Bluetooth module
  if (Serial.available()) {
    c = Serial.read();

    // do not send line end characters to the HM-10
    if (c!=10 & c!=13 ) {
      BTserial.write(c);
    }

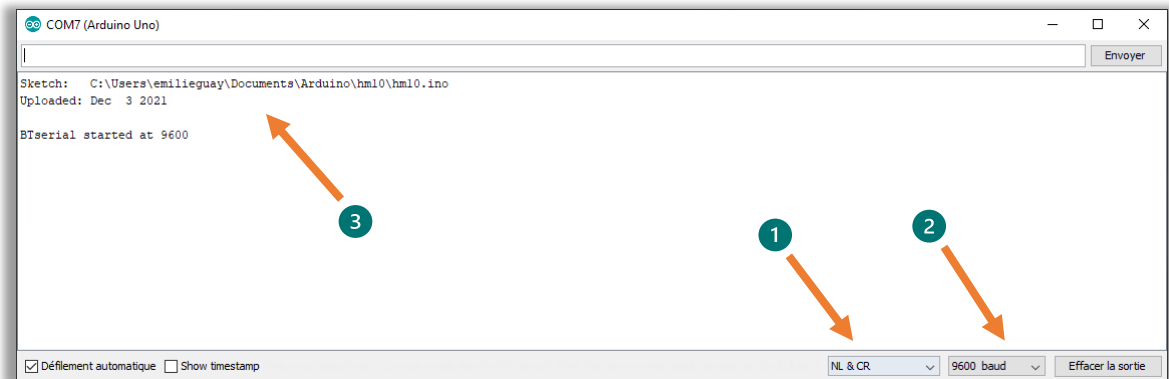
    // Echo the user input to the main window.
    // If there is a new line print the ">" character.
    if (NL) { Serial.print("\r\n>"); NL = false; }
    Serial.write(c);
    if (c==10) { NL = true; }
  }
}

```

5. Cliquer ensuite sur le bouton *Téléverser* pour envoyer le code à la carte Arduino.

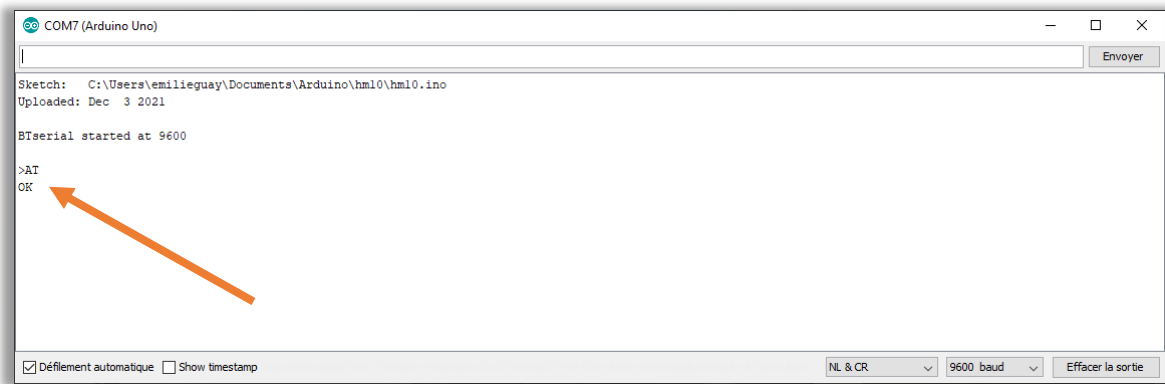


6. Cliquer sur l'onglet « Outils » et ensuite sur « Moniteur série ». C'est par le moniteur série que la configuration de la carte sera faite.
7. Dans la fenêtre du Moniteur série, s'assurer quand même que le moniteur est bien configuré :
 - a. Sélectionner « NL&CR » dans la liste déroulante.
 - b. Sélectionner « 9600 baud » dans la liste déroulante.
 - c. Le message de début que l'on peut voir à la figure suivante devrait apparaître. À noter que le nom de fichier devrait être celui du fichier Arduino que vous utilisez et la date devrait être celle d'aujourd'hui.



8. Entrer « AT » sans les guillemets à l'endroit indiqué. Appuyer sur la touche « Entrée » ou cliquer sur *Envoyer*.

Le message « OK » devrait apparaître. Sinon, vérifier le programme et les connexions du module. Il peut être possible aussi que la vitesse doive être de 115200 Baud au lieu de 9600 Baud.



Si cela ne fonctionne pas, vérifier les branchements, débrancher et rebrancher la carte Arduino.

Le module est ensuite prêt à être configuré.

9. Quelques petits conseils pour les commandes AT, car il peut être légèrement frustrant d'utiliser ces commandes la première fois.
 - a. Les majuscules et minuscules sont importantes. « AT+MODE? » par exemple, n'est pas la même chose que « at+mode? ».
 - b. Lorsqu'une commande est erronée, il n'y a aucun message affiché, sauf une ligne blanche vide. Il faut entrer la bonne commande encore pour continuer.
 - c. Ne jamais entrer les guillemets dans les commandes. Ils sont présents uniquement pour permettre une lecture plus facile.

Configuration de la carte en mode *Peripheral ou slavel* ou pour utilisation avec l'application App-IA

Nous devons d'abord vérifier la configuration de la carte et faire les modifications qui sont nécessaires seulement.

1. Entrer la commande : « AT »
 - a. Si la réponse est : OK passer à l'étape suivante.
 - b. S'il n'y a pas de réponse, vérifier les branchements.
2. Entrer la commande : « AT+ROLE? »
 - a. Si la réponse est : « OK+Get:0 », passer à l'étape suivante. Cela correspond à une connexion à n'importe quelle adresse.

- b. Si la réponse est : « 1 », entrer la commande « AT+ROLE0 ». Si la réponse est « OK+Set:0 », tout va bien. Sinon, réessayer en vérifiant la frappe.

Cela permet de configurer la carte pour qu'elle se branche sur n'importe quelle carte ou ordinateur qui le lui demande.

- 3. Entrer la commande : « AT+BAUD? »
 - a. Si la réponse est : « OK+Get:0 », passer à l'étape suivante.
 - b. Si la réponse est autre chose, entrer la commande « AT+BAUD0 ». Si la réponse « OK+Set:0 » apparaît, tout va bien. Sinon, réessayer en vérifiant la frappe.

Cela permet de configurer la carte pour qu'elle communique au bon rythme. Si l'on choisit d'autres configurations peut donner lieu à des comportements qui ne sont pas ceux attendus. Il est possible de configurer d'autres vitesses de communication. Il est important, cependant, que les vitesses des deux modules ou des modules et de l'ordinateur soient les mêmes. Dans le cas de l'utilisation avec l'application App-IA, on doit utiliser une vitesse de 9600 Bauds, ce qui est correct avec cette configuration.

- 4. Pour utiliser le module Bluetooth avec un autre module Bluetooth, il faut obtenir son adresse. Cela n'est pas nécessaire pour l'utiliser avec l'App-IA, donc on peut sauter cette étape.

Entrer la commande : « AT+ADDR? »

Cela permet d'obtenir l'adresse (*MAC address*) de la carte. Noter la valeur obtenue, c'est-à-dire les chiffres suivants le « : » dans « OK+ADDR: 0035FF209316 ».

- 5. Pour l'utilisation avec l'application, on doit s'assurer que le nom du module commence avec « HMSoft ». On peut toutefois ajouter des chiffres à la fin ou des lettres pour différencier les différents modules. Pour une classe complète, cela peut s'avérer utile.
 - a. Entrer la commande : AT+NAME?
Si la réponse est « OK+NAME:HMSoft », on peut utiliser le module tel quel.
 - b. Pour modifier légèrement le nom, utiliser la commande « AT+NAMEHMSoft01 ». Le nom du module sera changé pour « HMSoft01 ». Il sera plus facile de le repérer dans la liste de modules si plusieurs sont utilisés en même temps.

6. On peut aussi vérifier la version du *firmware* qui est installé dans le module.
 - a. Entrer la commande « AT+VERR? ». La réponse devrait être quelque chose comme « HMSoft V610 ». Ici, la version 6.10 du *firmware* est utilisée.

Pour ce qui est de l'utilisation avec l'App-IA, le module est maintenant prêt.

Pour configurer un 2^e module si on souhaite en utiliser une paire de modules ensemble, ne pas fermer le Moniteur Série. Débrancher la carte Arduino de l'ordinateur. Enlever le module HM-10 et le remplacer par l'autre si l'on souhaite en configurer une paire. Le 2^e module sera configuré comme « central » ou « master ».

7. S'assurer que le bon port est sélectionné pour la carte dans l'onglet Outils.
8. Télécharger le programme dans la carte Arduino.
9. Refaire les étapes 5 à 8 de la Configuration du module par Commandes AT et continuer avec la *Configuration de la carte en mode Master ou Central* :

Configuration de la carte en mode *Master ou Central* (pour utiliser deux modules BLE ensemble sans ordinateur)

1. Entrer la commande : « AT »
 - a. Si la réponse est : OK passer à l'étape suivante.
 - b. S'il n'y a pas de réponse, vérifier les branchements.
2. Entrer la commande : « AT+ROLE? »
 - c. Si la réponse est : « OK+Set:1 », passer à l'étape suivante. Cela correspond à une connexion à n'importe quelle adresse.
 - d. Si la réponse est : « 0 », entrer la commande « AT+ROLE1 ». Si la réponse est « OK+Set:1 », tout va bien. Sinon, réessayer en vérifiant la frappe.
 - e. Cela permet de configurer la carte comme *Master ou Central*
3. Entrer la commande : « AT+BAUD? »
 - c. Si la réponse est : « OK+Get:0 », passer à l'étape suivante.
 - d. Si la réponse est autre chose, entrer la commande « AT+BAUD0 ». Si la réponse « OK+Set:0 » apparaît, tout va bien. Sinon, réessayer en vérifiant la frappe.
4. On peut aussi vérifier la version du *firmware* qui est installé dans le module.

- a. Entrer la commande « AT+VERR? ». La réponse devrait être quelque chose comme « HMSOFT V610 ». Ici, la version 6.10 du *firmware* est utilisée.

Cette partie reste à vérifier :

5. Si on n'utilise qu'un seul autre module HM-10 en mode *peripheral* et qu'aucun autre module HM-10 n'est en fonction, il devrait être possible à partir de ce point que les deux modules se connectent ensemble automatiquement.
6. Si plus d'un module est en fonction dans la même pièce, il faut voir comment connecter les deux. Il ne semble pas exister de commande similaire à AT+BIND comme pour le module HC-05. Il faudrait vérifier...

Le site <http://www.martyncurrey.com/hm-10-bluetooth-4ble-modules/> semble proposer des façons de connecter deux modules. Il reste à vérifier le tout.

La commande AT+CON[adresse] semble prometteuse, mais elle reste à être testée pour voir si cela fonctionne même lorsque les modules ne sont pas branchés à l'ordinateur en commandes AT.

7. Fermer le Moniteur Série et débrancher la carte Arduino.

Les deux modules sont maintenant prêts à être utilisés.

7. Quelques notes

Une fois un module Bluetooth configuré, il conservera cette configuration jusqu'à ce qu'on le reconfigure. Donc, ces étapes n'ont pas à être refaites.

Lorsqu'un module Bluetooth est paillé avec un autre appareil, c'est-à-dire qu'il peut communiquer avec, la DEL qui se trouve dessus sera allumée en tout temps. Lorsque le module recherche une connexion, la DEL clignote.

Pour utiliser deux modules Bluetooth HM-10 ensemble, voir le tutoriel correspondant.