**Bibliographie**

*Charaix Fiona & Drouvin Marine*

*Polytech PEIP2, G3*

**Introduction :**

Notre projet consiste à créer un jeu vidéo : Un Pac-Man !

C’est un jeu plutôt simple, où un personnage se déplace à travers un labyrinthe et doit collecter tous les points sans se faire attraper par les fantômes, pour passer au niveau suivant.

Nous avons donc effectué de nombreuses recherches afin de trouver la meilleure solution pour atteindre notre but. Nous avons d’abord réfléchi à l’affichage de notre jeu, et étudié les différentes possibilités. Puis nous nous sommes intéressées au déplacement du personnage. Pour cela nous avons pensé rajouter une petite subtilité : nous voulons faire en sorte que le joueur puisse commander Pac-Man sans manette, sans télécommande, uniquement avec le mouvement de sa main ou de son corps.

Nous allons ici voir à travers différents chapitres les solutions possibles pour créer notre projet :

**Chapitre 1 :** Gérer l’affichage du jeu

1. Ecran LCD 16\*2
2. Ecran ILI9341
3. Panneau LED RGB 32\*32
4. Projet grandeur nature

**Chapitre 2 :** Communication entre 2 cartes Arduino

1. Le module NRF24L01
2. Le Xbee série 1
3. Le Bluetooth

**Chapitre 3 :** Contrôle télécommandé

1. Le contrôle à l’aide d’un potentiomètre
2. Utiliser un Gyroscope accéléromètre

- A. Accéléromètre analogique Adafruit

- B. L’accéléromètre et gyroscope à trois axes sparkFun

**Chapitre 4 :** Capteur de mouvement

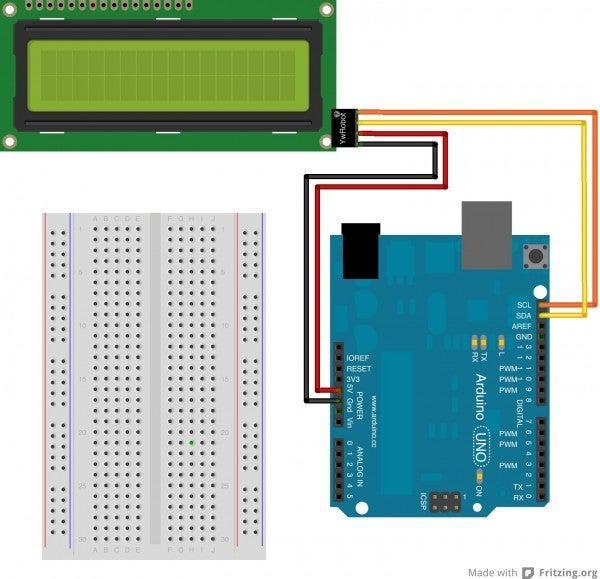
1. Echo comme capteur de mouvement
2. Le capteur de mouvement APDS-9960
3. Le détecteur de mouvement HC-SR501

**CHAPITRE 1 : Gérer l’affichage du jeu**

1. **Ecran LCD 16\*2**

Nous avons commencé par regarder des projets qui existaient déjà sur ce thème.  
Le premier que nous avons trouvé est le suivant : <https://www.instructables.com/PacMan-and-custom-characters-on-Arduino-with-a-16x/>  
Dans ce site, on utilise un écran LCD 16\*2 comme ceux que nous utilisons en cours ainsi que la bibliothèque LiquidCrystal\_l2Cv1-1.rar qui permet d’avoir des caractères personnalisés.

Ci-dessous on trouve le montage Arduino correspondant :



Ecran LCD 16\*2

Carte Arduino Uno

Carte de test 830 points

Figure 1.1 : Schéma du montage de l’écran lcd 16\*2

Ce projet est plutôt simple mais nous ne pouvons pas diriger Pac-Man.

De plus, la taille de l’écran reste beaucoup trop petite pour l’utilisation que nous voulons en faire. C’est pourquoi nous avons envisagé d’autres possibilités d’affichage comme par exemple l’écran ILI9341.

1. **Ecran ILI9341**



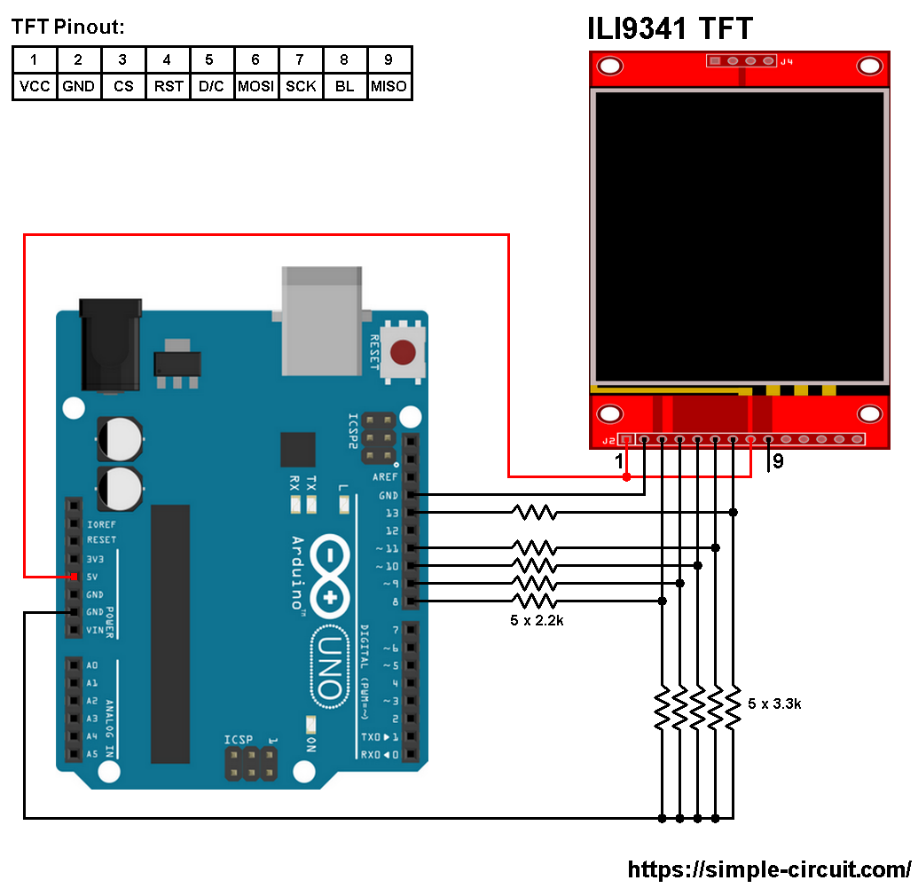
Figure 1.2 : Ecran ILI9341

Ce dernier nous permettrait d’obtenir quelque chose de ce type-là :



Ecran ILI9341

Figure 1.3 : Image du Pac-Man sur un écran ILI9341

Voici un exemple de montage possible :

Ecran ILI9341

Carte Arduino Uno

Résistances de 2,2k et 3,3k ohm

Figure 1.4 : Schéma du montage de l’écranILI9341 avec une carte Arduino UNO

(Lien : <https://simple-circuit.com/interfacing-arduino-ili9341-tft-display/> )

L’écran ILI9341 possède 14 entrées. Les 9 premières sont consacrées à l’affichage et les 5 autres au module tactile.

Cet écran ne fonctionne qu’avec du 3,3V or ici ils ont décidé d’utiliser la sortie 5V. C’est pourquoi ils font intervenir des diviseurs de tension pour chaque sortie. Ici par exemple on utilise des résistances de 2,2k et 3,3k ohm pour chaque diviseur, ce qui permet de diminuer la tension à 3V.

Afin de pouvoir afficher une image sur l’écran on utilise 2 librairies : « Adafruit\_ILI9341.h » et « Adafruit\_GFX.h »

L’avantage de cet écran, c’est qu’il est plus grand que le LCD 16\*2 décrit précédemment mais toujours pas suffisamment par rapport à ce que nous imaginions.

1. **Panneau LED RGB 32\*32**

Nous avons donc continué nos recherches et avons peut-être trouvé quelque chose qui correspondrait à nos attentes :



Figure 1.5 : Ecran LED RGB 32\*32

Un panneau LED RGB 32\*32 (une SmartMatrix). C’est ce qu’utilise la personne qui a réalisé ce projet (lien : <https://github.com/szczys/matrixman>)

L’écran ci-dessus (Figure 1.5) fonctionne avec des rubans NeoPixel (Figure 1.6), que l’on appelle aussi RGB adressable (nom qu’on trouve souvent sur internet). Les leds adressables ont l’avantage de pouvoir être contrôlées individuellement, contrairement aux leds non adressables, à qui chaque commande affectée à une led en affecte toutes les autres. Les rubans NeoPixel permettent également de jouer avec la luminosité des led ce qui peut être intéressant pour notre jeu.

Figure 1.6 : Rubans NeoPixel

L’utilisation des rubans NeoPixel nécessite un microcontrôleur, ici Arduino, et un programme. La plupart des projets que nous avons regardé utilisent la librairie NeoPixel qui permet de contrôler, de piloter les leds de type NeoPixel.

Voici 2 projets montrant l’utilisation de cette librairie :

<https://byfeel.info/bibliotheque-neopixel-avec-un-esp8266/>

<http://www.fablabredon.org/wordpress/2017/12/17/lumiere-sur-larduino-avec-de-la-couleur-et-des-led/>

Dans ces deux projets, on cherche à allumer des leds NeoPixel.

Ces rubans ont cependant quelques inconvénients. Les leds sont sensibles aux pics de courant, il est donc plus prudent de mettre une résistance et un condensateur. De plus, qui dit plus de pixel dit plus de RAM, plus de puissance ainsi que plus de temps et donc une cadence d’image plus faible. Cela va donc nous embêter lorsque nous allons créer notre labyrinthe ainsi que nos personnages.

Pour résoudre le problème nous avons regardé s’il n’était pas possible d’utiliser une carte SD pour avoir un espace de stockage plus important.

Voici le lien vers une vidéo montrant comment réaliser ceci : <https://www.youtube.com/watch?v=iswES_5HkeM>

Dans ce projet, on utilise une carte micro SD placée sur un « Ready Micro-sd Adafruit breakout », ainsi qu’un capteur d’humidité « DHD11 » pour simplement donner des data à enregistrer sur la carte SD.

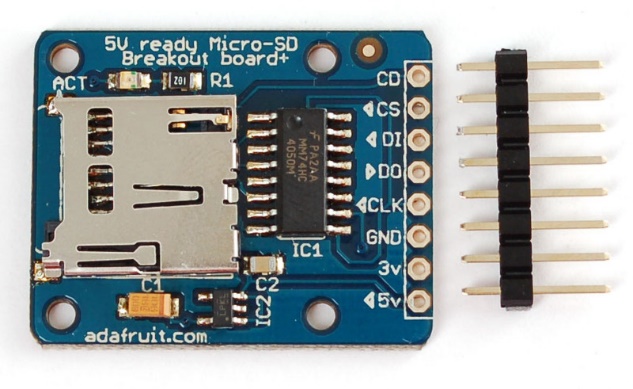
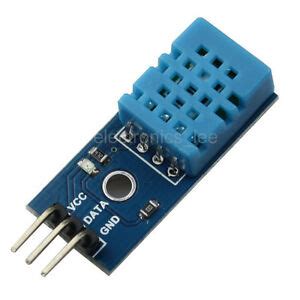
 

Figure 1.7 : Ready Micro-sd Adafruit breakout Figure 1.8 : DHD11

Voici un montage simplement avec le ready Micro-SD Adafruit breakout :

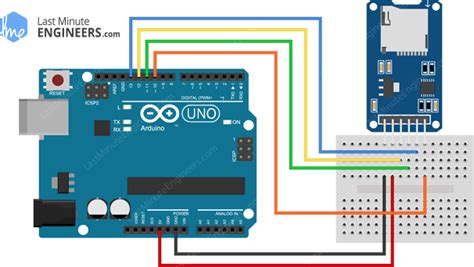


Figure 1.9 : montage avec Micro-SD

Ce module utilise 3V ou 5V et consomme 150 mA.

Grâce à cette astuce nous allons récupérer de l’espace pour pouvoir coder notre affichage et nos personnages sans problème !

1. **Projet grandeur nature**

En continuant nos recherches nous avons trouvé un autre projet original et plutôt intéressant (lien : <https://magpi.raspberrypi.org/articles/robot-pac-man>).

Les créateurs ont voulu créer un Pac-Man grandeur nature :

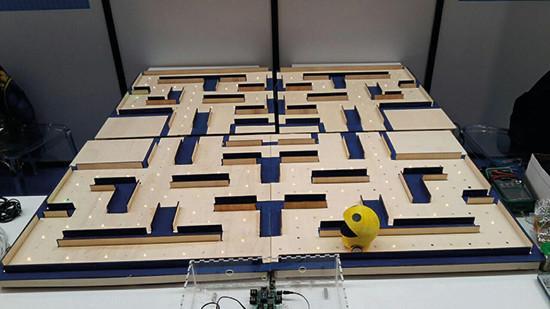


Figure 1.10

Pour cela, ils ont créé un labyrinthe représentant le terrain de jeu et des robots pour les personnages.

Ces derniers sont reliés chacun à une carte Arduino Uno communiquant avec des joysticks par fréquence radio de 2,4 GHz.

De nombreuses cartes (fabriquées par les créateurs eux-mêmes) contenant des leds et des commutateurs Reed connectés à une Arduino Mini, sont disposées en dessous du labyrinthe. Sous le personnage principal se trouve un aimant. Lorsqu’il se déplace l’Arduino Mini comprend quel commutateur Reed est activé et détermine ainsi la led à éteindre.

Malheureusement, pour réaliser un projet de cette ampleur il nous faudrait plus de temps et peut être plus d’expérience. C’est pourquoi nous avons décidé de privilégier l’écran pour afficher notre jeu.

**CHAPITRE 2 : Communication entre 2 cartes Arduino**

Dans cette deuxième partie nous verrons comment on peut communiquer entre deux cartes Arduino et quel module correspond au mieux à notre projet.

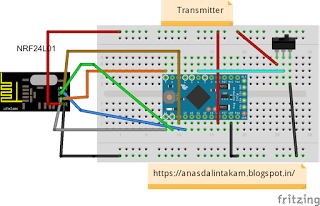
**1) Le module NRF24L01**

Dans un premier temps, on peut utiliser pour communiquer à distance une NRF24L01

Figure 2.1

Ce module permet de connecter deux cartes Arduino entre elles.   
Sur ce site on a un branchement entre un émetteur et un récepteur permettant d’allumer une led à distance.

Schéma de l’émetteur :

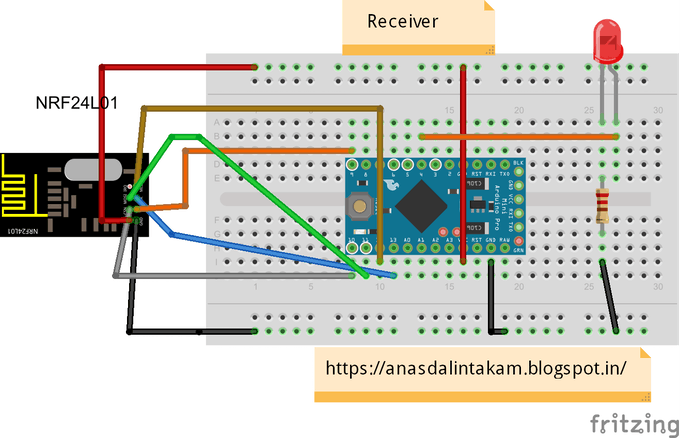


Carte Arduino Uno

Module NRF24L01

Figure 2.2

Schéma du récepteur**:**



Carte Arduino Uno

Module NRF24L01

Figure 2.3

Ce module permet de communiquer à une portée de 100 m pour la version de base et 1km pour la version avec antenne. Cependant la version de base aura une portée suffisante. Il fonctionne par onde radio de 2.4Ghz et utilise la bibliothèque Arduino Mirf ou la bibliothèque RF24.h, nRF24L01.h et [SPI.h](https://www.arduino.cc/en/reference/SPI) que l’on a rencontré dans nos recherches. Ce module utilise une alimentation 3V3 et 5 sorties numériques. Le module NRF24L01 consomme au maximum 12.5 mA. Il reprend le principe de maitre et esclave. L’un donne des ordres : l’émetteur, et l’autre les exécute : le récepteur (dans notre projet il s’agira de l’écran). Ce module peut être intéressant au premier à bord notamment pour sa portée de communication, cependant il utilise quand même 5 entrées logiques sur la carte Arduino.

**2) Le Xbee Série 1**

Un des autres modules permettant de communiquer entre deux cartes Arduino est le module Xbee. Voici comment marche ce dernier : Il est branché à une tension de 3.3V et consomme 20mA, du moins pour la version standard. Le module communique par radiofréquence (2.4GHz) et a une portée de 30 mètres en intérieur.

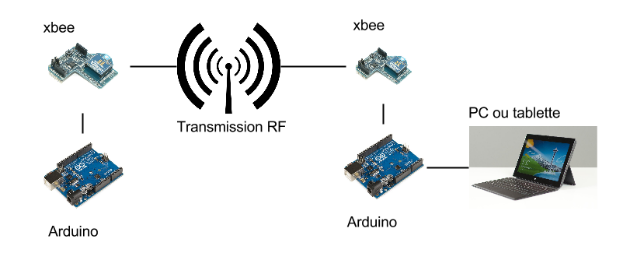


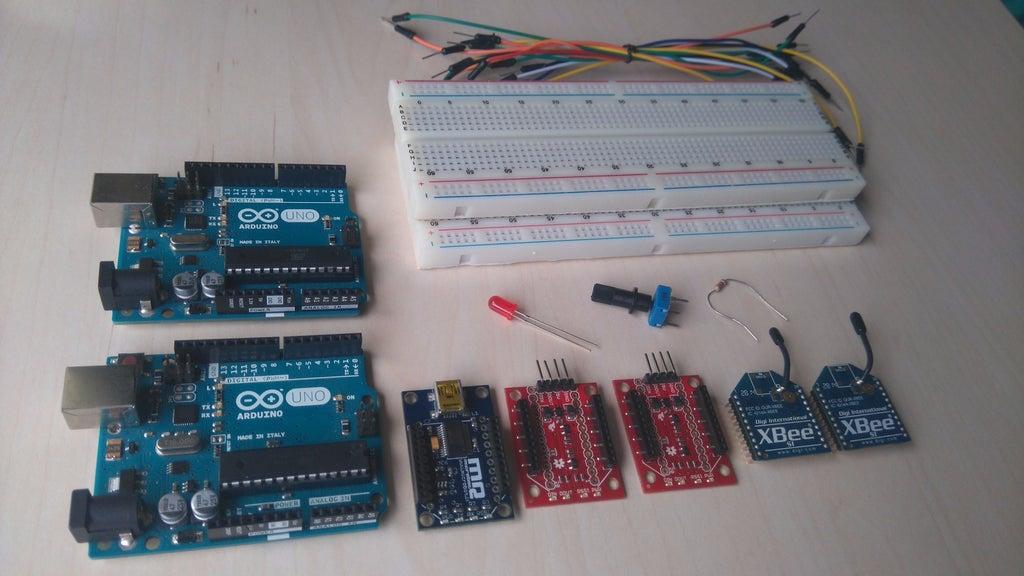
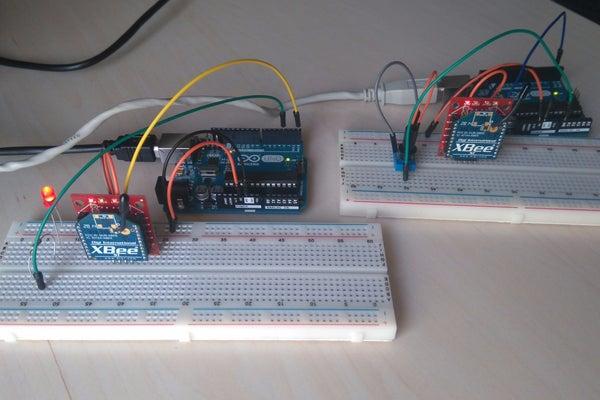
Figure 2.4

Dans ce site (<https://www.instructables.com/How-to-Use-XBee-Modules-As-Transmitter-Receiver-Ar/>) on utilise ce principe mais avec un potentiomètre.

Potentiomètre

LED



Module XBee série 1

XBee Explorer USB

Carte Arduino Uno

SparkFun XBee Explorer Regulated

Figure 2.5

C’est le module XBee série 1 qui permet de communiquer entre les deux cartes Arduino comme expliqué précédemment.

Le module XBee Explorer USB permet, une fois connecté à notre ordinateur (depuis l’application XCTU), d’indiquer les commandes sans avoir à coder les deux modules individuellement. Cela permet de débugger séparément l’émetteur et le récepteur. Ce n’est pas obligatoire dans le montage on peut directement coder avec la carte Arduino dans le loop() et le setup() même si c’est peu pratique.

Le module SparkFun XBee Explorer Regulated permet quant à lui de réguler le signal en signal 3.3V. On peut aussi le remplacer par une résistance ou utiliser la sortie 3V3 de notre carte comme présenté sur la figure 2.6. Ce module fonctionne avec tous les modules Xbee.

Nous pourrons aussi utiliser la librairie Xbee.h pour nous aider dans notre projet.

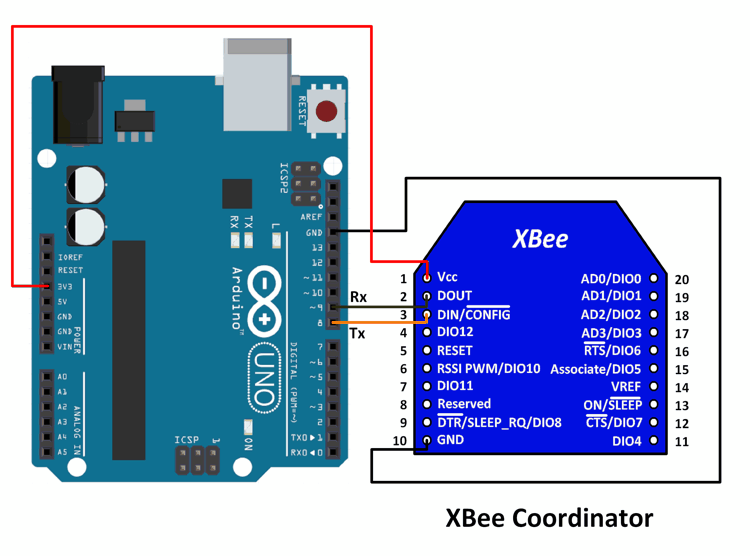
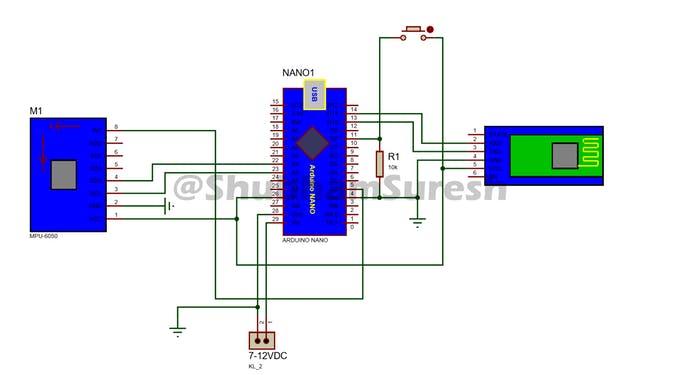


Figure 2.6 : schéma du montage avec le module Xbee

**3) Le Bluetooth HC-05.**

Voici un autre projet qui utilise le Bluetooth pour communiquer entre deux cartes Arduino : <https://create.arduino.cc/projecthub/shubhamsuresh/how-to-make-a-gesture-control-robot-at-home-a3f4a4>

La configuration est simple à utiliser et à programmer ce qui est un avantage. De plus le module Bluetooth permet de communiquer sur une distance de 9mètres, distance largement suffisante dans notre cas. Ce projet présente la communication entre deux modules Bluetooth HC-05.   
Voici le schéma du gant servant d’émetteur :



Accéléromètre et gyroscope à trois axes sparkFun MPU-6050

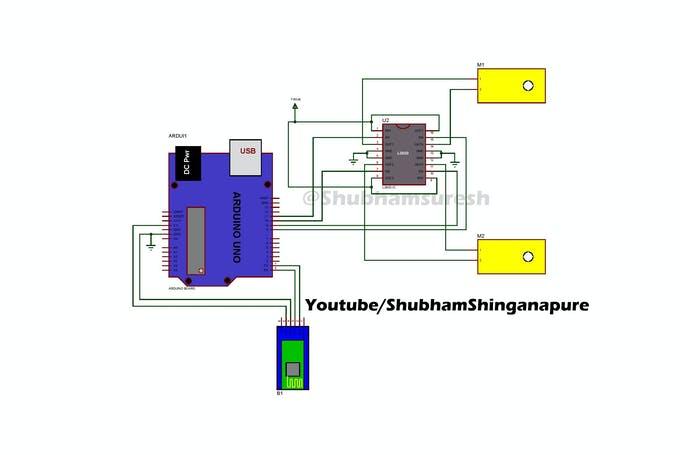
Résistance

Module Bluetooth HC-05

Carte Arduino nano

Figure 2.7

Partie réceptrice, le robot :

Carte Arduino Uno

Module Bluetooth HC-05

Cerveau moteur Texas instruments dual HbridL293D

Figure 2.8

Le module Bluetooth utilise une sortie branchée en 5V et une sortie branchée à la masse. On a ensuite la broche de réception : RX, connectée à la broche de transmission : TX de l’Arduino. Ou alors, la broche de transmission : TX connectée à la broche de réception : RX de l’Arduino. (Attention on ne peut pas brancher RX et TX en même temps sur le module).

La branche State retourne 1 dans le moniteur lorsque le module est connecté. La branche Key ou EN doit être alimentée pour entrer dans le mode de configuration. On ne la branchera alors que lorsque l’on programmera le module.

La tension de fonctionnement peut aller de 3.6V à 6V et consomme 50mA.

L’un des modules se comporte comme un esclave (récepteur) tandis que l’autre agira comme un maitre (l’émetteur) en envoyant les ordres à l’esclave. Ils seront liés automatiquement. Pour programmer les deux modules, on utilisera la librairie SoftwareSerial.h. On peut directement utiliser l’interface Arduino pour programmer inutile d’utiliser un logiciel particulier.   
L’avantage de ce module c’est qu’il est simple à utiliser et prend très peu de sorties logiques.

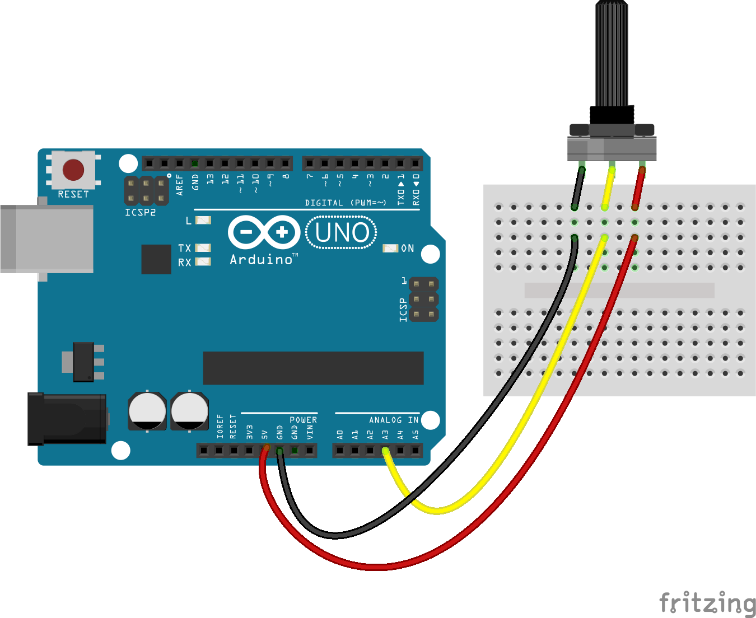
**CHAPITRE 3 :** **Contrôle télécommandé**

Pour contrôler notre Pac-Man, nous pouvons nous servir de notre main sans utiliser un capteur de mouvement mais d’autres moyens. Ainsi dans cette partie nous regarderons les différentes possibilités que nous possédons.

**1) Contrôle à l’aide d’un potentiomètre**

Comme nous avons pu voir précédemment nous pouvons utiliser un potentiomètre. Sur ce site (<https://create.arduino.cc/projecthub/matejadjukic03/potentiometer-game-05ee93>), la personne utilise un potentiomètre pour contrôler son Pong.

Le montage est basique et identique à celui vu en cours.

 Figure 3.1

Potentiomètre

Carte Arduino Uno

Le contrôle du Pac-Man ne demande pas de librairie particulière.

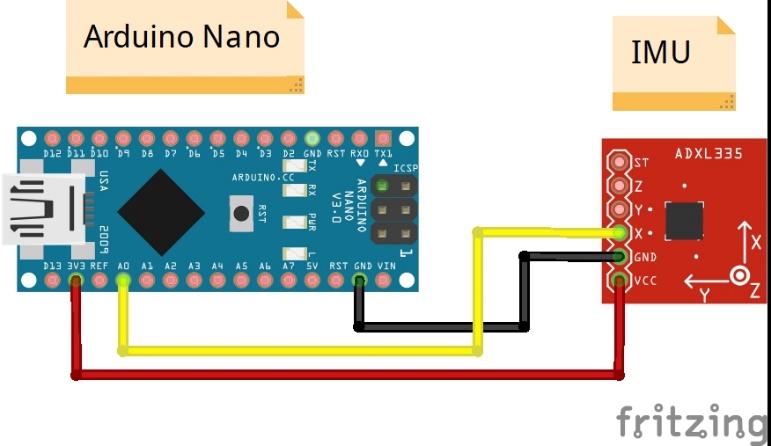
Le voltage peut varier de 0V à 5V et le courant dépendra du potentiomètre que l’on choisit.  
Cependant nous souhaitons utiliser la main. C’est pourquoi nous ne choisirons pas cette option.

**2) Utiliser un Gyroscope accéléromètre.**

Le Gyroscope accéléromètre peut aussi être un moyen efficace de contrôler Pac-Man grâce à nos mouvements.

A- Accéléromètre analogique Adafruit

Dans ce projet : <https://create.arduino.cc/projecthub/mircemk/interactive-pong-pc-game-with-adxl335-accelerometer-24c7bb> nous pouvons contrôler le jeu pong avec un accéléromètre analogique Adafruit. L'ADXL335 est un petit accéléromètre complet à 3 axes, mince, de faible puissance, avec des sorties de tension conditionnées par le signal.



Carte arduino nano

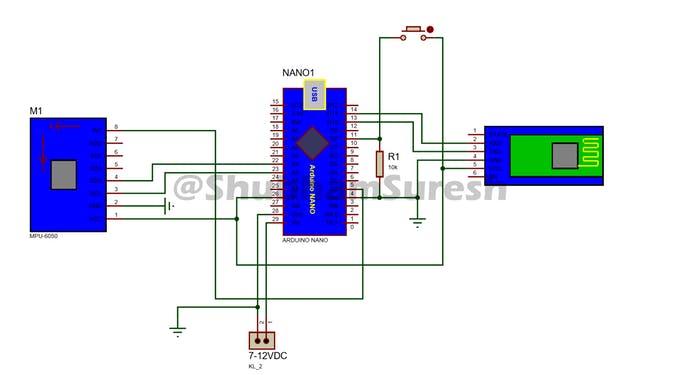
Accéléromètre analogique Adafruit

Figure 3.2

Ce module consomme très peu de puissance, seulement 350µA et possède une tension variant de 1.8V à 3.6V. Il faudra alors le brancher sur la sortie 3V3.

B) L’accéléromètre et gyroscope à trois axe sparkFun

On peut aussi prendre l’accéléromètre et gyroscope à trois axe sparkFun vu dans le grand deux qui marche avec un précédant à savoir le contrôle du Bluetooth. Nous ne représenterons pas de nouveau tous les schémas de ce projet mais seulement la partie concernant le module que l’on vous présente.



**Module Bluetooth HC-05**

**Résistance**

Accéléromètre et gyroscope à trois axe sparkFun MPU-6050

**Carte Arduino** nano

Figure3.3

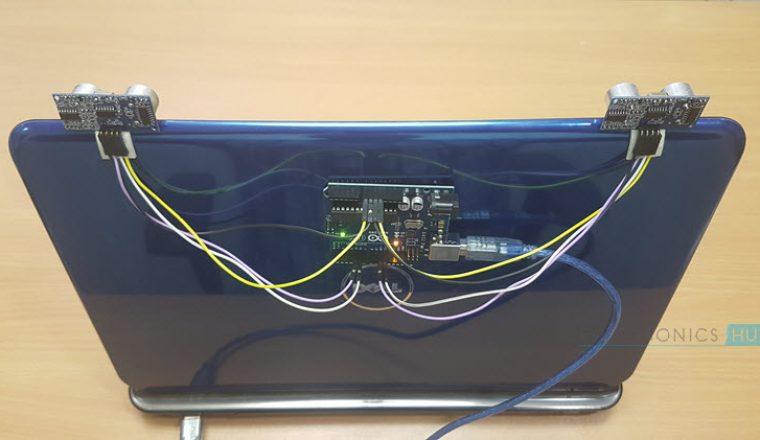
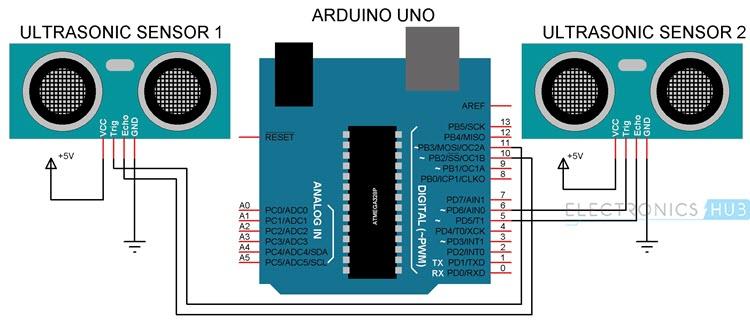
Pour le capteur MPU-6050 (le gyroscope accéléromètre) on devra utiliser la librairie MPU60 50\_6Axis\_MotionApps20.h et MPU6050.h.   
Celle-ci nous permettra alors de configurer notre Module MPU-6050. Ce module utilise 2.3 à 3.4V. L’avantage du gyroscope trois axe. C’est que l’on peut lui donner les contrôles simples et cela sera plus précis qu’un capteur comme écho. La programmation est quant à elle assez complexe mais faisable.

**Chapitre 4 : Capteur de mouvement**

Dans cette dernière partie, nous regarderons comment utiliser un capteur de mouvement pour déplacer notre Pac-Man. Et quels modules existent-ils pour cette fonction.

1. **Echo comme capteur de mouvement**

La première solution que nous avons envisagé serait d’utiliser la mesure de distance avec le module HC-SR04 comme présenter sur ce site : <https://www.electronicshub.org/arduino-based-hand-gesture-control-computer/>

 Figure 4.1

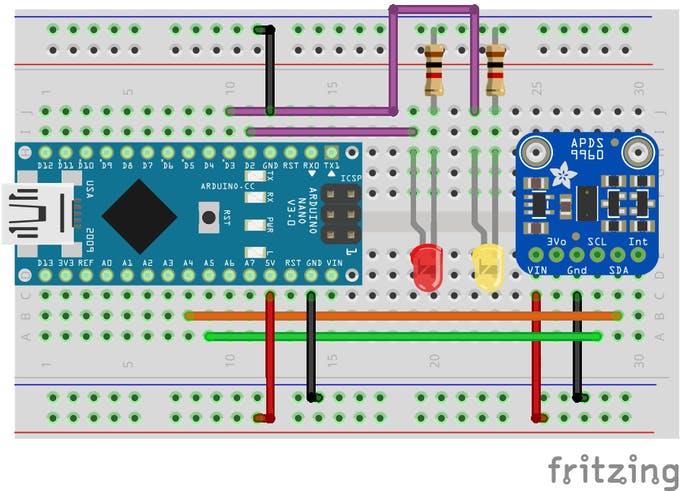
Carte arduino Uno

Module HC-SR04

Figure 4.2

Dans ce projet, il utilise deux capteurs ultrasons (comme ceux vu en cours) placé à chaque extrémité supérieure de l’ordinateur. Grâce à ces deux capteurs, que l’on relit à un carte Arduino, on pourra alors définir les 4 trajectoires que notre Pac-Man doit effectuer. Cependant cette méthode manque de précision pour le contrôle d’un personnage de jeu vidéo.

**2) Le capteur de mouvement APDS-9960**Ce deuxième projet (<https://create.arduino.cc/projecthub/moreirarobotics/arduino-menu-with-apds-9960-gesture-sensor-control-b9c71e> ) utilise le capteur de geste APDS-9960 pour détecter ces mouvements et ainsi changer la lumière. Ce capteur capte les mouvements de bas en haut et de droite à gauche. Comme nous souhaitons seulement bouger notre Pac-Man dans 4 directions cela sera largement suffisant. Cependant c’est un capteur de proximité, il capte donc à distance de 10 à 20 cm. C’est son plus gros désavantage.



Carte nano

Capteur de geste APDS-9960

Résistance

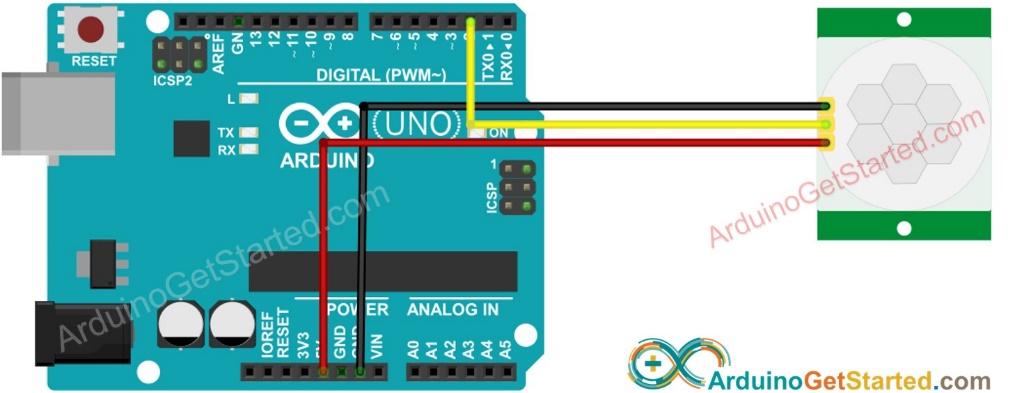
LED

Figure 4.3

Dans ce projet, la personne utilise Adafruit APDS9960 comme librairie comme pour le capteur. Il consomme 250µA et varie de 2.4 à 3.6V.  
L’inconvenant de ce capteur de mouvement est que la portée de la détection des mouvements est trop faible. Il vaut mieux trouver un autre moyen.

**3) Le détecteur de mouvement HC-SR501**

Ce module est un détecteur de mouvement, il repère les changements de rayonnement infrarouge d’un object en mouvement. Le capteur HC-SR501 se branche de la manière suivante : <https://create.arduino.cc/projecthub/electropeak/pir-motion-sensor-how-to-use-pirs-w-arduino-raspberry-pi-18d7fa> :

Carte Arduino Uno

Le capteur HC-SR501

Figure 4.4

Le branchement du capteur est basique. Une partie est branchée au 5V de la carte Arduino. Une autre (le branchement du milieu) est branchée sur une sortie numérique. Le dernier emplacement est pour la masse. Il possède aussi deux potentiomètres qui permettent d’ajuster la portée du signal en fonction de notre projet et de nos besoins. Il a une synchronisation de sortie de 0.5S à 200S et il fonctionne de 4.5V à 20V.  
Il est donc assez simple d’utilisation. Si nous devions l’utiliser dans notre projet il nous faudrait hélas plusieurs modules car il ne capte que la présence. Cela le rend moins pratique pour notre projet et moins intéressant au vu de ce qu’on a déjà précédemment.

CONCLUSION :

Pour réaliser notre projet nous avons donc effectué de nombreuses recherches et comparé les différentes solutions envisageables. Concernant l’affichage notre choix s’est porté sur l’écran 32\*32. En effet, il nous permettrait d’obtenir un affichage plus pratique pour le joueur.

Pour la communication entre plusieurs cartes Arduino nous avons décidé que le Bluetooth HC-05 reste la meilleure solution. Le principe « maître-esclave » est ce qui convient le mieux pour notre projet. De plus la distance de communication (9 m) est largement suffisante. Enfin, il utilise très peu d’entrées logiques comparées aux autres modules.

Parmi toutes les méthodes vues pour contrôler notre personnage, nous avons décidé de privilégier le module gyroscope à 3 axes SparkFun MPU6050. Il est plus précis que les détecteurs de mouvement. Le contrôle du Pac-Man sera plus simple.

Finalement pour ce projet nous avons besoin de :

* Ecran Led RGB 32 par 32
* Module « Ready Micro-sd Adafruit breakout »
* Carte SD
* 2 modules Bluetooth HC-05
* Gant classique (sur lequel on puisse souder)
* Module Gyroscope accéléromètre à 3 axes SparkFun MPU6050
* 1 ou 2 Carte(s) UNO Méga sinon 1 carte UNO méga et une carte Xplained mini