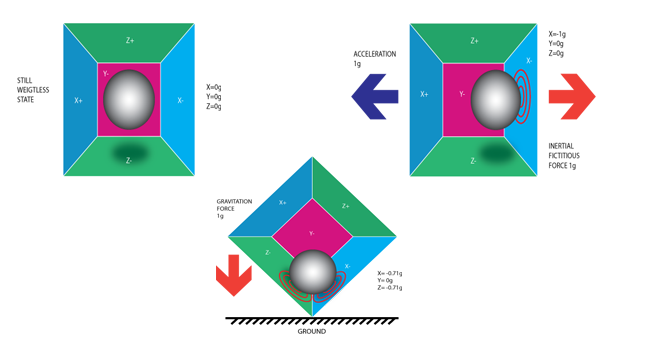
Xavier Corbeil / Tommy Brassard  
1445229 / 1450257  
PROJET D'INTÉGRATION EN SCIENCES INFORMATIQUES ET MATHÉMATIQUES  
420-204-RE gr .101

Documentation

Présenté à   
Robert TURENNE

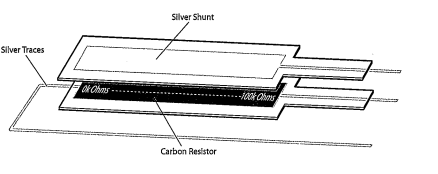
Département d’informatique  
Cégep de Saint-Jérôme  
28 avril 2017

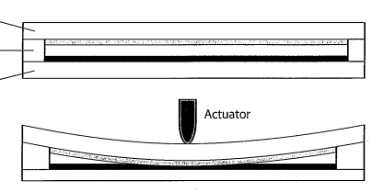
Notre projet consiste à créer un skate qui réplique nos mouvements à l’écran. Ce dernier agit comme un professeur qui nous montre quelles sont nos erreurs pour nous aider à nous améliorer. Sur la planche il y a un MPU6050, c’est-à-dire un gyroscope qui agit aussi comme un accéléromètre. Cette dernière mesure des données jusqu’à un maximum de 8 000 fois par seconde. Une fois ceux-ci acheminés au Arduino, elles seront transférées encore une fois à un autre logiciel nommé Processing. Le gros de notre travail se déroule dans ce programme. Les données seront interprétées et traduites en mouvements et accélération. Une texture de skateboard sera faite dans Processing et les données traduites seront utilisées pour faire bouger cette texture en temps réel avec le vrai skateboard. Lorsque le vrai skate va faire un ollie, c’est-à-dire un saut vers le haut sans variante, le skate à l’écran va faire le même mouvement en temps réel. Ce dernier fera de même peu importe la figure que l’utilisateur fera. Sur la planche il y aura aussi des capteurs de pression, soit des petites pastilles, sur la moitié de la planche en avant. Ceux-ci agiront en tant que switch on/off. Si le pied qui frotte sur la planche passe sur l’un d’eux, il sera en position on. À l’écran il apparaitra en couleur sur la surface de la planche. Le but de ceci est qu’avec plusieurs capteurs nous allons être en mesure de voir la trace du pied qui frotte. Ceci permettra de voir si la trajectoire est bonne ou pas pour être en mesure de réaliser une figure souhaitée. Une fois que le principe du skate sera terminer, nous allons faire un petit menu l’écran avant de démarrer l’expérience en temps réelle. Dans ce menu il y aura une option que servira à calibrer le gyroscope avant de commencer. Cette étape est importante si l’on veut que le skate nous apparaisse à la même position à l’écran que dans la vraie vie. Une fois cela terminé, l’expérience en temps réelle est prête à commencer et le skateur peut essayer plusieurs figures pour voir ce qu’il fait de correct ou pas lors de l’exécution de cette dernière.

**MPU6050** : Ce dernier est muni d’un gyroscope et d’un accéléromètre qui peuvent analyser des mouvements en x, y et z. Le gyroscope est de type piezo, c’est-à-dire qu’à l’intérieur de celui-ci il y a une petite pierre, soit un barreau de piezo, qui vibre continuellement lorsqu’il est alimenté. Si la pierre subit une contrainte mécanique, c’est-à-dire une force de torsion, cela produira un courant électrique. Ce courant sera amplifié et acheminer à un microcontrôleur. Ce dernier les raffinera et les enverra au Arduino, qui lui les enverra à Processing, sous forme de bytes, pour les faire analyser et les utiliser pour faire les mouvements à l’écran. Ensuite, il y a l’accéléromètre. Ce dernier utilise le principe d’une boule emprisonnée dans une boite. Lorsque le système est au repos, la balle est en plein centre du système et ne bouge pas. Cependant, si l’on déplace l’accéléromètre vers la gauche, c’est-à-dire l’axe des x négatifs, la boule sera forcée de se déplacer à l’opposé du mouvement, donc vers la droite. Lors de son déplacement, elle ira se cogner contre le mur qui correspond à l’axe des x négatifs. Ce contact créera un petit courant électrique qui sera traduit pour être en mesure de savoir la direction ainsi que la magnitude du mouvement analysé. Ainsi de suite pour tous les mouvements parce que chaque mur correspond à un axe x, y ou z.

Ce petit appareil se retrouvera sous la planche et sera fixé au Arduino, qui lui aussi sera sous la planche.

**SoftPot Membrane Potentiometer** : Cette bandelette va servir à situer le pied qui servira à effectuer une figure avec la planche. Donc, il y en aura au moins 5 ou 6 sur la planche pour permettre de voir précisément le déplacement du pied. Elle fait 50 cm de long ce qui couvre pratiquement le ¾ de la planche en longueur. Pour les installer sur la planche il faudra découper la forme de la bandelette dans la griptape, c’est-à-dire la surface abrasive de la planche, sinon la pression que cette dernière fera sur chacune des bandelettes faussera les données dû au fait qu’elles sont très sensibles.

Cette bandelette est constituée de deux membranes espacées l’une de l’autre. Lorsque l’on appui sur le dessus de la bandelette, c’est deux membranes se touchent pour créer une variation de résistance. Plus la pression s’éloigne des “pins”, plus la résistance diminue.



La résistance peut varier entre 1000 Ohms et 20 000 Ohms dépendamment d’où la pression est effectuée. Ce qui nous permet de savoir précisément où le pied sur la planche va frotter. Cette résistance sera envoyée sous forme de courant à l’Arduino. Ce dernier va les analyser et les envoyer à Processing. Dans ce logiciel, nous ferons en sorte que la trace de notre pied soit visible sur une autre planche à droite de l’écran. Pour les mettre sur la planche il faudra couper dans la griptape, soit la surface abrasive de la planche, une ouverture de la taille exacte de la surface qui peut capter de la bandelette. Donc, seul la surface qui est active sera à l’air libre. Le reste sera couvert par la griptape. L’embout des bandelettes seront connecté directement dans le Arduino en passant au ****travers la planche.

L’ovale noir représente l’endroit où l’embout des bandelettes entreront dans la planche.

**Boitier de métal** : Celui-ci se trouvera sous la planche pour protéger l’Arduino et tous les branchements de fils, des chocs qu’ils pourraient subir lors de l’utilisation de la planche. Le boitier sera fait en métal pour être suffisamment résistant à de forts impacts. Le rectangle gris le représente.

**Balles de tennis** : Ceux-ci seront placé sur chacune des roues de la planche pour éviter trop de déplacement de celle-ci. De plus, elles serviront à réduire le bruit ainsi que l’impact au sol lors de l’exécution d’une figure.

**Sans fil**: …