**Compte rendu**

**Présentation**

Ce projet est réalisé dans le cadre de notre cours d’Arduino en PEIP2 de l'école Polytech Nice-Sophia. Nous allons développer un système permettant d'apprendre à lire le Braille en autonomie. Nous utiliserons pour cela une carte Arduino.

**Séance du 15 Décembre :**

Lors de cette séance je devais en apprendre plus sur la manière dont je pouvais créer un code Arduino capable d’associer chaque lettre de l’alphabet a un code en braille. Pour ce faire je me suis dans un premier temps renseigné à l’aide de projet déjà préexistant sur le net. C’est alors que je suis tombé sur le projet de Cesare qui est très similaire au notre.

(Projet braille similaire (ASCII)

<https://create.arduino.cc/projecthub/CesareBrizio/ascii-braille-real-time-translation-via-arduino-dd97a9> )

Je me suis donc attelé à la compréhension de son code. Son code est assez simple à comprendre puisque très bien commenter.

On a aussi pu avoir accès au github du groupe d’il y a trois an. Nous avons donc pris la décision de reprendre leur code puisque celui-ci était censé fonctionner.

(Projet braille d’il y a 3 ans <https://github.com/BlindTouch/Projet> )

La compréhension de celui-ci a été plus difficile mais petit à petit on a pu comprendre quelles étaient les rôles de tels ou tels variables, fonctions.

J’ai ensuite essayé d’adapter en partie leur programme en un nouveau capable de lire un mot donné dans le terminal et de convertir chaque lettre en code binaire (sur six bit ?) et de nous renvoyer celui-ci. Je n’ai pas eu le temps de finir correctement ce programme et il y a encore de choses à revoir dans le programme des anciens étudiants, je continuerais cela chez moi.

**Séance du 5 Janvier :**

J’ai expliqué à mon camarade les parties du code que j’ai comprises (et ses nouvelles modifications) et je l’ai questionné sur celles non comprises. Cela ne m’a pas beaucoup avancé, il y a toujours des fonctions que je ne comprends pas, mais je pense avoir trouvé une solution puisqu’en fin de séance j’ai compris un détail (conversion pour stocker en hexadécimal) qui peut me permettre de mieux comprendre ses zones d’ombres.

Jean a identifié quelle broche correspondait au plus et au moins, nous avons ensuite reçu quelques instructions quant aux risques de mal brancher le générateur 12V (le circuit est alimenté en 12V). Il faut mettre le plus sur Vin pour ne pas faire fondre la carte Arduino ou endommager mon ordinateur.

Pour vérifier les branchements nous avons d’abord tester l’initialisation du programme qui fait bouger les solénoïdes, l’Arduino était alimenté par ma batterie externe pour ne pas casser mon ordinateur en cas d’erreur de branchement.

Après vérification, rien n’a pris feu et l’initialisation fonctionnait, on l’a donc branché à mon ordinateur pour tester le programme. Nous nous sommes malheureusement rendu compte que le programme ne fonctionne pas correctement puisqu’au-delà d’un mot de plus de deux lettres, les lettres sont mal traduites, il y a donc un problème de conversion en binaire (au moment du stockage puis de la redistribution des informations), je pense savoir d’où vient le problème.

Nous avons aussi réalisé quelques soudures au cours de cette séance notamment pour réparer le câblage du plus du circuit.

Après plusieurs minutes de test le système (les solénoïdes) chauffe un peu, mais nous sommes rassurés car on pensait qu’il chaufferait beaucoup plus.

Une image contenant équipement électronique, adaptateur

Description générée automatiquementUne image contenant texte, intérieur

Description générée automatiquement(images des solénoïdes en mouvement (en position haute/pas tous) et des branchements)

**Séance du 12 Janvier :**

J’ai pu me rendre compte qu’un certain nombre de problèmes n’avait pas été soulever/noter au cours de la dernière séance par exemple le fait que l’initialisation ne marche pas toujours. Je ne sais pas si cela vient du fait que les solénoïdes chauffent et donc reste en haut ou si cela vient d’un dysfonctionnement ou même d’un bug lier au code.

Cela nous a amené au second problème, étant donné que certains solénoïdes restent bloquer en haut, il chauffe, on n’avait pas remarqué que ça chauffait autant la semaine dernière étant donné qu’on le coupait après chaque test. Aujourd’hui on l’a laissé activer plus longtemps et ça a nettement plus chauffer (certains solénoïdes se sont même décollés).

Le dernier problème vient du code, j’ai passé l’entièreté de la séance à essayer de trouver l’erreur de traduction. J’ai avancé et pu comprendre d’où provenait l’erreur (conversion), mais je me suis confronté à un nouveau problème, désormais le code me renvoie les bonnes informations (c’est-à-dire la bonne combinaison de chiffres) mais le système avec les solénoïdes ne renvoie pas le bon visuel. Je pense que cela provient du fait qu’une certaine partie d’entre eux étant à l’état haut (bien qu’initialiser à l’état bas) l’information n’est pas stockée et on ne peut traduire un mot complet. La traduction ne fonctionnant que sur deux lettres (avec malgré tous des bugs de temps en temps).

(Après différents tests, je me suis rendu compte que la semaine dernière la traduction était faussé une fois sur deux, elle ne fonctionnait qu’après la première initialisation pour un mot de deux lettres).

Le point clé de résolution de ses problèmes réside donc dans la compréhension des modules ULN2803 et 74HC595.