Rapport de laboratoire

**Ecole supérieure**

Électronique

Laboratoire POBJ

Salle R110

2008 Sablier Electronique

**Réalisé par :**

Rodrigo Martins Silveira

**A l’attention de :**

M. Bovey

**Dates :**

Début du laboratoire : 12 mai 2022

Fin du laboratoire : 6 juin 2022

**Table des matières :**

2008 Sablier Electronique 1

1 Cahier des charges 5

2 Taches à faire 5

3 Planning 5

4 Description état initial du projet 6

4.1 Etat hardware 6

4.2 Etat software 6

4.2.1 Test software 6

4.2.2 Bug 7

4.2.3 Test du fonctionnement USB 7

5 Description de l’état final du projet 8

5.1 Essai de connexion USB 8

5.1.1 Testes effectuer 8

5.2 Etat final projet 8

6 Conclusion 8

7 Annexes 9

# Cahier des charges

Voici les donné sur le projet fournis par M. Bovey.

* Partie uC : implémentation USB – implémentation communication avec Bluetooth
* Partie C# : implémentation communication sérielle plus interface graphique permettant à paramétrer le sablier électronique + affichage en live
* Partie Android : Protocol Bluetooth – interface graphique pour paramétrer le sablier.

# Taches à faire

* Prise en main du projet
* Test de fonctionnement de la dernière version de code
* Dépannage du code
* Développement d’une application USB **(cdc\_com\_port\_single\_2008)**
* Test de l’application de base USB
* Mise en commun des deux applications

# Planning

|  |  |  |
| --- | --- | --- |
| Tâches | Heures planifiées [h] | Heures réelles [h] |
| Prise en main du projet (note 1) | 3 | 2 |
| Test de fonctionnement de la dernière version de code | 1 | 2 |
| Dépannage du code (note 2) | 1 | 3 |
| Développement d’une application USB | 9 | 1 |
| Test de l’application de base USB (note 3) | 0.1 | 2 |
| Mise en commun des deux applications (note 4) | 3 | - |

1. Projet très mal organisé, incompréhension et problème de fichiers.
2. L’application n’a pu être finie, donc le nombre d’heures réelles.
3. En vue des problèmes de communication, temps réels plus élevé.
4. Pas débuté.

# Description état initial du projet

## Etat hardware

J’ai commencé par alimenter le système, celui-ci affichait le mouvement du sablier, mais je ne pouvais pas configurer le temps pour le sablier.

Je présume que le projet était chargé avec une version de test, car dans le rapport, il est mentionné que le sablier fonctionne correctement.

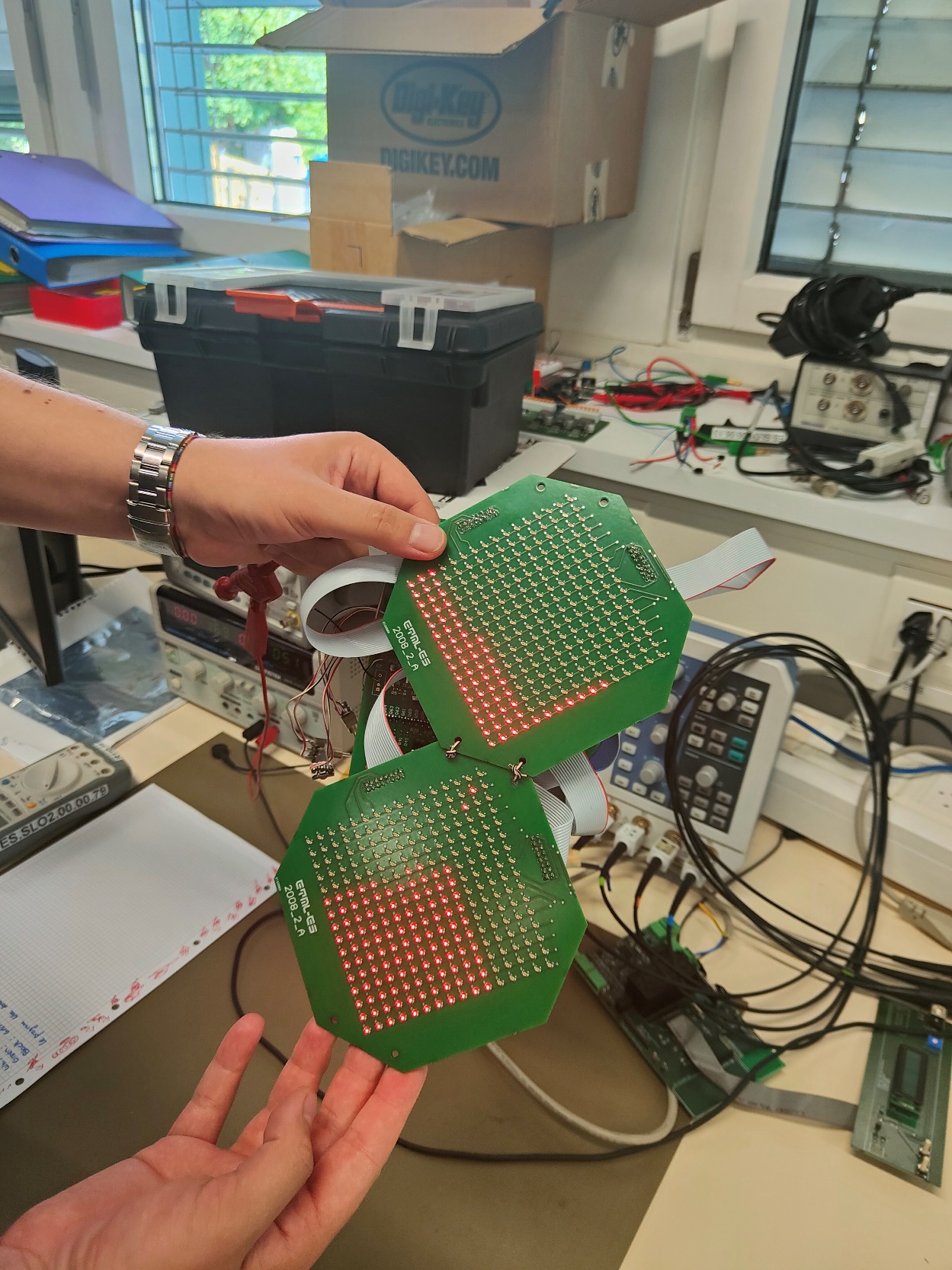
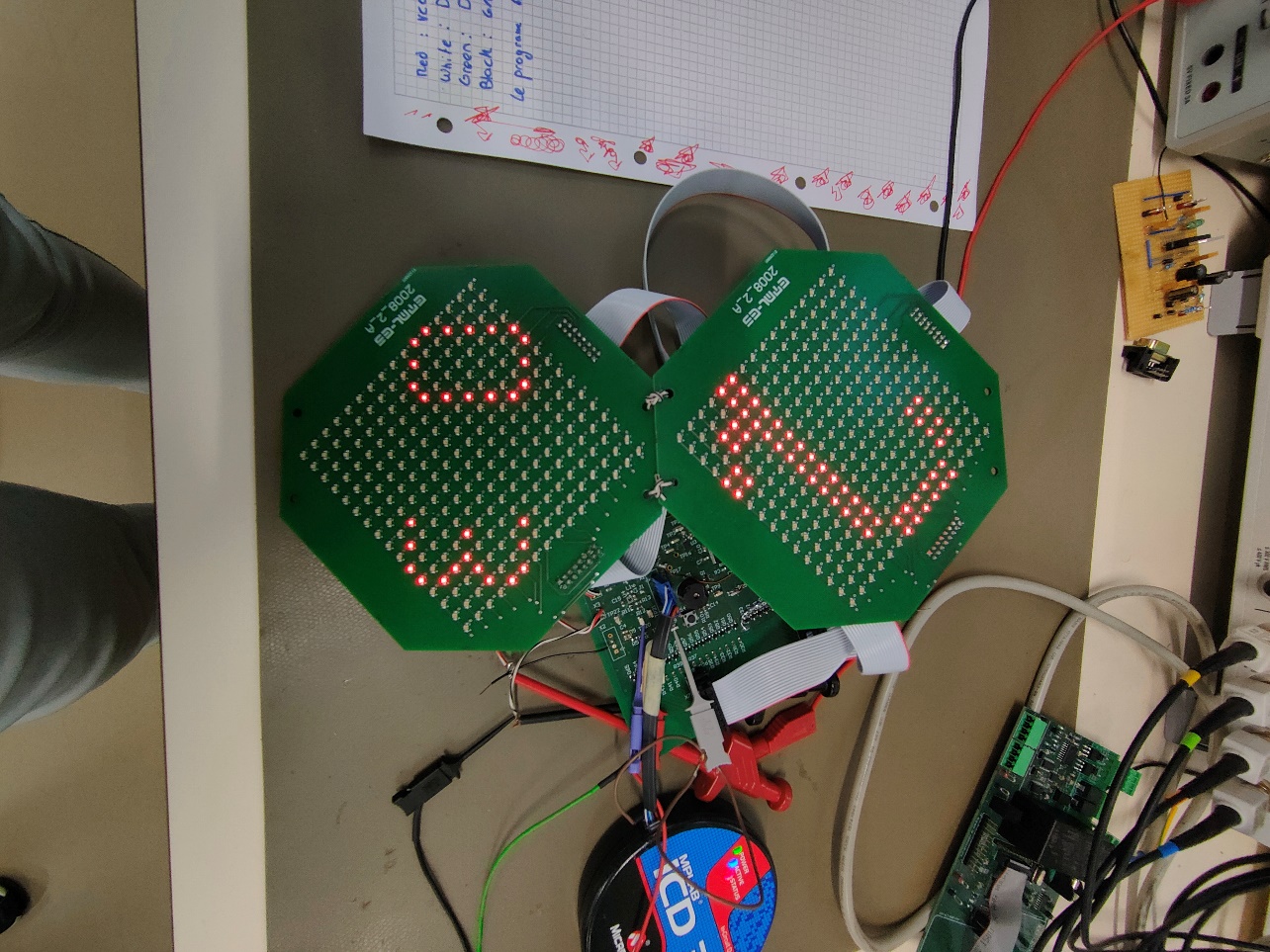
## Etat software

J’ai commencé par lancer la dernière version du code nommée : **« 2008\_SablierElectronique\_B »**.

Lors du lancement de cette version, l’affichage n’est plus fonctionnel, le port USB ne fonctionne pas. Dans cette version du code, il y a 2 applications, l’application du sablier et l’application USB.

### Test software

En vue du résultat obtenu avec la dernière version du code, j’ai opté de faire une copie de l’ancienne version du code nommé **« 2008\_SablierElectronique\_V1new »** et la renommer **« 2008\_SablierElectronique\_V2 »**, dans lequel je pourrais procéder à des modifications et de pouvoir afficher le sablier et choisir un temps.



Aucun changement majeur n’a dû être fait (explication dans le point **Bug**).

Afin de pouvoir tester le fonctionnement du port USB, j’ai repris l’exemple de microchip **« cdc\_com\_port\_single\_2008 »**, je l’ai configuré en utilisant la théorie de MINF (Lien en annexe).

Cette application permet de recevoir un caractère et de renvoyer un autre caractère.

Ex : Putty -> 1 -> uC -> 2 -> Putty

Emplacement : **C:\microchip\harmony\v2\_06\apps\usb\device**

Le projet ne possède pas de quartz externe, donc j’ai dû régler le système avec le quartz interne du uC de 8MHz. J’ai suivi la configuration du dernier projet Version B (Config en annexe).

### Bug

Lors du lancement de la version V2, des erreurs d’emplacement du projet se sont affichées et l’ancien emplacement, c’est affiché. J’ai donc recréé ce chemin.

**Emplacement : C:\microchip\harmony\v2\_06\Diplome**

En deuxième temps, d’autres erreurs sont parvenues, telles que le manque de fichiers source du projet, afin de résoudre ceci, j’ai retiré du projet et j’ai remis les fichiers ayant un problème.

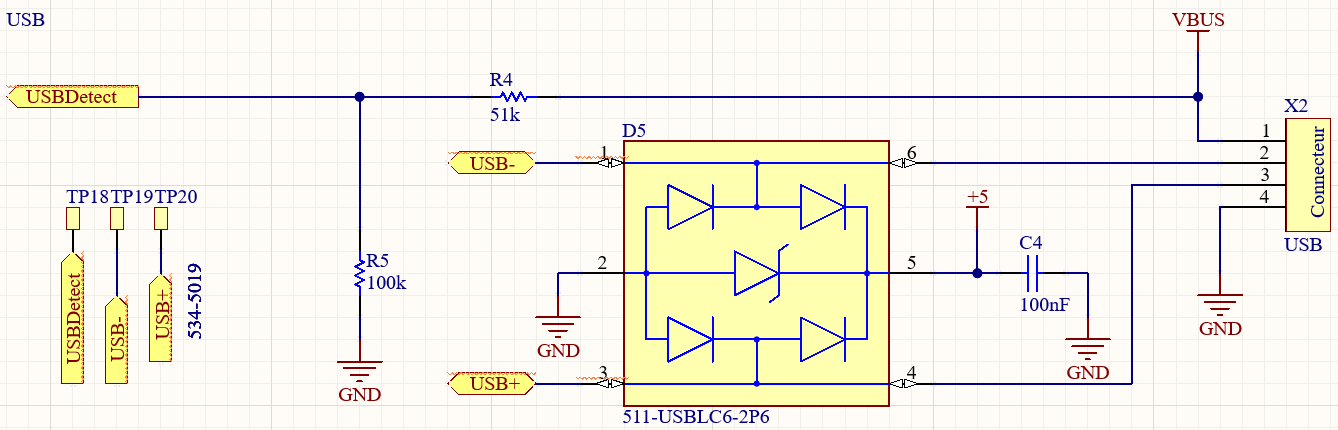
Des erreurs de compilation sont parvenu, certain dû à un fichier test de gestion du driver **testIam20680**.

J’ai retiré du projet ce fichier et également corrigé des erreurs du a des implémentations des timers.

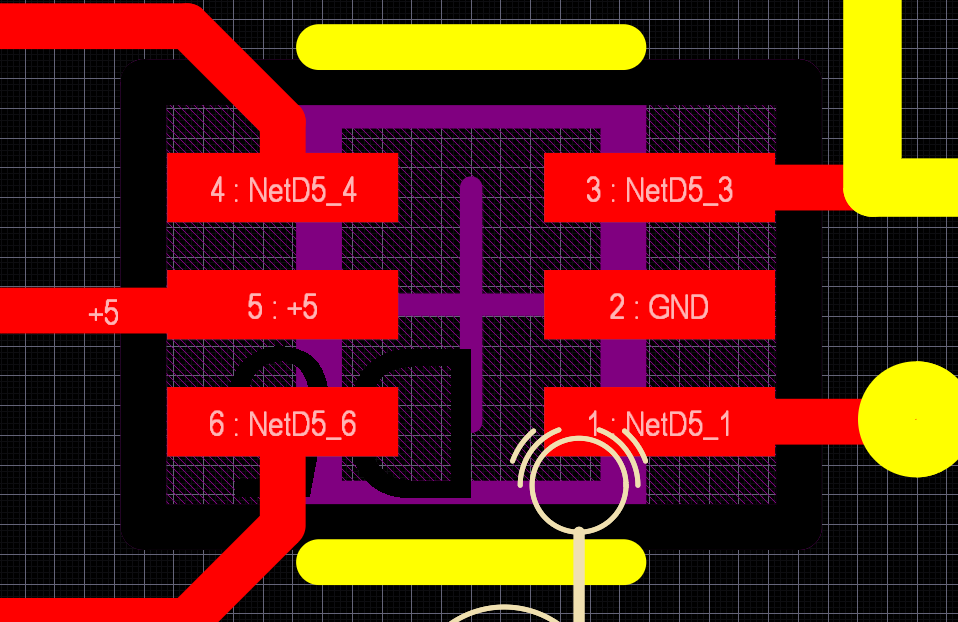
### Test du fonctionnement USB

Dans le rapport, c’est mentionné que la communication USB n’est pas fonctionnelle (point 6.4 Conclusion).

J’ai commencé par vérifier le schéma de USB, et celui-ci correspond à la datasheet et au kit ETML\_ES PIC32 (on peut trouver le même design dans le kit ES)



Mais lors de la vérification du montage PCB, j’ai pu apercevoir que le composant est manquant, mais des connexions entre le uC et le footprint du composant on était faites.



Le GND est connecté à un polygon que j’ai masqué.

Pour parvenir à faire un test, en vue du manque du connecteur USB en vue du mauvais footprint de celui-ci, j’ai utilisé des pointes de teste sur lequel je suis venu connecteur des fils.

Câble USB Pin out :

Fil blanc : D+ TP19

Fil vert : D- TP20

Fil rouge : VCC

Fil Noir : GND

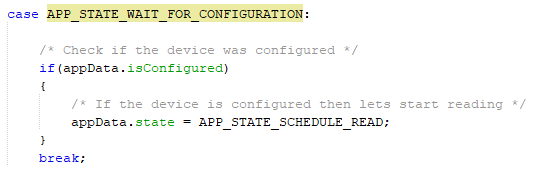
# Description de l’état final du projet

## Essai de connexion USB

Après plusieurs tentatives d’essai de communication, celui-ci n’est pas fonctionnel, lorsque je connecté le câble USB à l’ordinateur, celui-ci n’est pas détecté.

### Teste effectuer

J’ai fait du debug, et j’ai pu observer que celui-ci reste bloqué dans la case **« APP\_STATE\_WAIT\_FOR\_CONFIGURATION »**, c’est pour cette raison que l’USB n’est pas détecté.



Le programme revient plusieurs fois dans ce test, si celui-ci arrive à se connecter, il ira dans APP\_STATE\_SCHEDULE\_READ dans lequel on attend un caractère de la part du master.

## Etat final projet

Le sablier peut être configuré manuellement avec la version du code V2.

La partie USB ne fonctionne toujours pas, manque de composant de protection aux bornes de la sortie du connecteur (USBLC6-2P6).

# Conclusion

Pendant ce projet, j’ai pu m’apercevoir de l’importance de l’organisation et des mises à jour des fichiers avant la clôturassions d’un projet.

Car j’ai dû tester plusieurs versions de code pour trouver une version sur lequelle je pouvais repartir. J’ai dû faire du debugger afin que je puisse avoir une idée du fonctionnement du projet, et de corriger les erreurs de compilation.

Afin d’éviter ceci, il aurait suffi de créer un fichier clair, expliquant quels changements il y t’il entre les différentes versions, quelles fonctionnalités sont fonctionnel.

Le projet ne comporte pas de mode d’emploi ce qui ne facilite pas la tâche à la compréhension du fonctionnement.

Pour chaque fichier de projet programmation, j’ai laissé le chemin dans lequel nous devons le déposer afin que le projet puisse être réutilisé sans difficulté.

Malheureusement, je ne suis pas parvenu à corriger le problème de la connexion USB, mais un fichier avec le code du sablier est fonctionnel.

Le prochain pas serait de braser le composant de protection voir si celui-ci peut changer le comportement du système

Lausanne ETML-ES Rodrigo Martins Silveira 6 juin 2022

# Annexes

Emplacement théorie de labo : Supports de cours\SL229\_MINF\TP\CoursLabo

Emplacement dossier projet 2008 : K:\ES\PROJETS\SLO\2008\_SablierElectronique

Configurations Oscillateur interne Pic32

