

Piscine embarquée

Rush 01 : Final boss

[contact@42chips.fr](mailto:contact@42chips.fr)

*Résumé: Realisation d'un gros projet*

# Chapitre I

## Préambule

La station Saint-Michel, située dans le 5ème arrondissement de Paris, est l'une des stations les plus anciennes et les plus emblématiques du métro parisien. La construction de cette station a été une entreprise colossale, qui a nécessité l'utilisation de techniques de construction innovantes pour s'adapter aux contraintes géologiques et architecturales de la ville.

La station Saint-Michel a été construite en 1910, dans le cadre de l'extension de la ligne 4 du métro parisien. La principale difficulté rencontrée lors de sa construction était liée à la profondeur du sol, qui rendait impossible la construction d'une station en surface. Les ingénieurs ont donc décidé de creuser un tunnel à une profondeur de plus de 20 mètres sous la Seine, afin de pouvoir construire la station sous le niveau du fleuve.


La construction de la station Saint-Michel a également été marquée par des considérations architecturales. La station a été conçue par l'architecte Hector Guimard, qui a créé un design unique inspiré par l'Art nouveau. La station est caractérisée par des piliers en fonte ornés de motifs floraux et des entrées en métal qui ressemblent à des grottes. Aujourd'hui, la station Saint-Michel est l'une des plus célèbres de Paris, et est souvent considérée comme un exemple remarquable de l'architecture Art nouveau.

# Chapitre II

## Consignes générales

Sauf contradiction explicite, les consignes suivantes seront valables pour tous les exercices.

- Le langage utilisé pour ce projet est le C.
- Il n'est pas nécessaire de coder à la norme de 42.
- Les exercices sont très précisément ordonnés du plus simple au plus complexe. En aucun cas nous ne prendrons en compte ni n'évaluerons un exercice complexe si un exercice plus simple n'est pas parfaitement réussi.
- Vos exercices seront évalués par des responsables de l'association 42Chips.
- Vous ne devez laisser aucun autre fichier que ceux explicitement spécifiés par les énoncés des exercices dans votre répertoire lors de la peer-évaluation.
- Toutes les réponses à vos questions techniques se trouvent dans les **datasheets** ou sur Internet. À vous d'utiliser et d'abuser de ces sujets pour comprendre comment réaliser votre exercice.
- Vous devez utiliser la datasheet du microcontrôleur qui vous est fourni et commenter les parties importantes de votre programme en renseignant où vous avez trouvé les indices dans le document, et, si nécessaire, expliquer votre démarche. Ne faites pas des pavés non plus. Il faut que cela reste clair.
- Vous avez une question ? Demandez à votre voisin de droite ou de gauche. Vous pouvez demander sur le salon dédié dans le Discord de la piscine ou en dernier recours à un staff.

	Exercice : 00
Final boss	
Dossier de rendu : <i>ex00/</i>	
Fichiers à rendre : <b>Makefile</b> , <b>*.c</b> , <b>*.h</b>	
Fonctions Autorisées : <b>avr/io.h</b> , <b>util/delay.h</b> , <b>avr/interrupt.h</b>	

Pour ce dernier rush, vous devrez développer un firmware pour la devboard afin de tester que tout fonctionne correctement.

Pour cela, vous devrez utiliser tout ce que vous avez appris jusque là.

Au démarrage du firmware, tous les segments du 7-segment ainsi que **D1**, **D2**, **D3** et **D4** doivent s'allumer pendant 3 secondes, puis s'éteindre pendant 1 seconde.

À tout moment, il doit être possible de tester **SW1**, **SW2**, **SW3**.

Lorsque le bouton **SW1** est appuyé, la LED **D9** s'allume.

Lorsque le bouton **SW2** est appuyé, la LED **D10** s'allume.

Lorsque le bouton **SW3** est appuyé, la LED **D11** s'allume.

Le programme a plusieurs modes pour afficher différentes choses sur le 7-segment.

Lorsque l'on appuie sur **SW1**, on passe au mode suivant.

Lorsque l'on appuie sur **SW2**, on passe au mode précédent.

Après le dernier mode, on retourne au premier.

Les LEDs **D1-D4** affichent en binaire le mode dans lequel on se trouve actuellement.

Voici ce que font chaque mode :

- 0 : affiche la valeur de l'ADC de RV1 en décimal (0 à 1023) ;
- 1 : affiche la valeur de l'ADC branché sur la LDR **R14** en décimal (0 à 1023) ;
- 2 : affiche la valeur de l'ADC branché sur la NTC **R20** en décimal (0 à 1023) ;
- 3 : affiche la valeur de température interne du MCU en décimal (0 à 1023) ;
- 4 : affiche "-42-" et fait clignoter les LEDs **D5**, **D6**, **D7**, **D8** en rouge, puis vert, puis bleu (1 seconde entre chaque couleur) ;
- 6 : affiche la température ambiante en Celsius ;
- 7 : affiche la température ambiante en Fahrenheit ;
- 8 : affiche l'humidité en pourcentage ;
- 9 : affiche l'heure au format 24h (23 :23) ;
- 9 : affiche le jour et le mois ;
- 10 : affiche l'année.

Il doit être possible de régler la date et l'heure grâce au port série.

En envoyant par exemple : "28/02/2024 19 :42 :21"



Attention, l'heure doit être gardée même si on débranche et rebranche la board.

Utilisez la **RTC** de votre board.