



Piscine embarquée

Rush 00 : Multiplayer

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

*Résumé: Initiation à la communication entre circuits intégrés*

# Chapitre I

## Préambule

Tennis for Two est un jeu vidéo de tennis conçu en 1958 par [William Higinbotham](#) et développé par [Robert Dvorak](#), souvent considéré comme le premier jeu vidéo de l'histoire.

Le jeu est développé sur un ordinateur analogique Donner Model 30 dédié et relié à un oscilloscope qui sert d'écran. Le concept est présenté en 1958 et 1959 lors des portes ouvertes du laboratoire national de Brookhaven.

Le jeu, développé en quelques semaines, a pour but de montrer les avancées technologiques au grand public, à l'occasion des portes ouvertes du laboratoire.

En période de guerre froide, Higinbotham souhaite montrer une image plus sympathique du monde de la recherche, alors que les esprits sont encore marqués par les premiers essais nucléaires des années 1950.

À partir de documents sur la simulation de trajectoires de balles, il crée un concept de divertissement, sans utilité académique particulière, à l'aide d'un oscilloscope et d'un ordinateur analogique.

Le jeu reçoit un bon accueil auprès du public local, où des files d'attente se forment pour accéder à la machine, et il est présenté au public deux ans de suite. Malgré l'innovativité du concept pour l'époque, le jeu n'est plus utilisé après 1959, le laboratoire décidant de remplacer cette attraction par une autre aux portes ouvertes suivantes et de réutiliser les pièces pour d'autres recherches.



Ces exercices sont à faire en binôme. Vous pouvez choisir votre coéquipier.



Mais les évaluations sont en solo. Vous devez connaître et comprendre tout le code rendu !

# 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.

# Chapitre III

## Partie obligatoire

	Exercice : 00
Howdy cowboy	
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>util/twi.h</b>	

L'objectif de cet exercice est de faire un jeu de rapidité où 2 microcontrôleurs communiquent ensemble. Vous devez utiliser vos breadboards de TP.

- Connecter les 2 microcontrôleurs à l'aide de 2 fils.
- Le microcontrôleur AVR ATmega328P possède 1 interface I2C que vous devez utiliser dans cet exercice pour communiquer avec un autre microcontrôleur.
- l'I2C du MCU doit être configuré avec une fréquence de 100 kHz.
- Vous devez écrire une fonction qui décide quel microcontrôleur à la main sur l'autre (qui est le master).
- Les MCUs doivent être capables de communiquer si 1 des 2 vient de redémarrer.



Le programme doit être le même pour chaque microcontrôleur !

Le jeu est plutôt simple :

- Quand les 2 joueurs ont appuyé sur le bouton le jeu commence.
- Un compte à rebours s'affiche sur les leds.
- Le joueur le plus rapide à avoir appuyé sur le bouton à la fin du compte à rebours gagne la partie.
- Une autre partie peut maintenant commencer.




Si le joueur appuie sur le bouton avant la fin du compte à rebours alors il perd la partie. Nous vous laissons créer un signallement lumineux indiquant le vainqueur et le perdant. N'oubliez pas de commenter les règles pour la correction !

Le compte à rebours peut être aléatoire dans sa durée mais doit être au minimum de 2 secondes et au maximum de 10 secondes.

# Chapitre IV

## Bonus

	Exercice : 01
Mexican Standoff	
Dossier de rendu : <i>ex01/</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>util/twi.h</b>	

Maintenant que vous avez 2 joueurs il serait temps, si vous en avez encore, d'en ajouter un autre pour les meilleures sensations du Farwest !

L'I2C est un bus ! Il serait temps d'utiliser ses capacités à fond !

Vous pouvez aussi ajouter ce que vous voulez du moment que la partie obligatoire reste la même. Amusez-vous !



Si vous avez bien pensé à comment faire votre exercice 00 alors les modifications devraient être minimales ;)