



Electronics Project, TP Prog

TP 3 : Interruptions

contact@42chips.fr

Résumé: Initiation aux interruptions système.

Table des matières

I	Préambule	2
II	Consignes générales	3
III	Exercice 00 : Interruption extérieure	4
IV	Exercice 01 : Timer0 interruption et PWM	5

Chapitre I

Préambule

Chapitre II


Consignes générales

Sauf contradiction explicite, les consignes suivantes seront valables pour tous les Travaux Pratiques de Programmation embarquée.

- Le langage utilisé pour ce projet est le C.
- 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.
- Vous avez une question ? Demandez à votre voisin de droite ou de gauche. En dernier recours, vous pouvez demander à un responsable 42Chips.
- Toutes les réponses à vos questions techniques se trouvent dans les **datasheets** ou sur Internet.
- Pensez à discuter sur le Slack !
- Écoutez attentivement les encadrants lors des séances de TP, ils vous donneront des éléments essentiels sur le fonctionnement du microcontrôleur.

Chapitre III


Exercice 00 : Interruption extérieure

	Exercice : 00
Interruption extérieure	
Dossier de rendu : <i>ex00/</i>	
Fichiers à rendre : Makefile , main.c	
Fonctions interdites : PINx , avr/interrupt.h	

- Vous devez écrire un programme qui allume la LED lorsqu'on appuie sur le bouton et reste allumé.
- Vous devez utiliser les interruptions pour lire l'état du bouton. Il est interdit de lire les registres PINx.

Chapitre IV

Exercice 01 : Timer0 interruption et PWM

	Exercice : 01
Timer0 interruption et PWM	
Dossier de rendu : <i>ex01/</i>	
Fichiers à rendre : <code>Makefile</code> , <code>main.c</code>	
Fonctions interdites : <code>avr/interrupt.h</code>	

- Vous devez configurer les registres du `Timer0` et une interruption pour faire varier le rapport cyclique du clignotement de la LED PB1.
- Le rapport cyclique doit varier en boucle de 0% à 100% puis de 100% à 0% en 1 seconde.