



# Electronics Project, TP Prog

## TP 1 : Premiers programmes

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

*Résumé: Initiation à la cross-compilation et à l'utilisation des registres AVR*

# Table des matières

<b>I</b>	<b>Préambule</b>	<b>2</b>
<b>II</b>	<b>Consignes générales</b>	<b>4</b>
<b>III</b>	<b>Exercice 00 : Breadboard</b>	<b>5</b>
<b>IV</b>	<b>Exercice 01 : Makefile</b>	<b>6</b>
<b>V</b>	<b>Exercice 02 : Une lueur d'espoir</b>	<b>7</b>
<b>VI</b>	<b>Exercice 03 : Clignotant</b>	<b>8</b>
<b>VII</b>	<b>Exercice 04 : Lumos</b>	<b>9</b>
<b>VIII</b>	<b>Exercice 05 : Jour, nuit, jour, nuit</b>	<b>10</b>
<b>IX</b>	<b>Exercice 06 : Compteur binaire</b>	<b>11</b>

# Chapitre I

## Préambule

Le Chikki est une confiserie traditionnelle d'Inde à base de sucre non raffiné. Il y a plusieurs variétés de Chikki, le plus commun étant celui qui est fait avec des arachides. Chaque variété est nommée selon les ingrédients utilisés ; elles peuvent contenir des pois chiches, du sésame, du riz soufflé et/ou de la noix de coco desséchée.

Pour préparer des Chikkis, il vous faudra :

- 150 g d'arachides grillées
- 5 cardamome verte (graines servant à relever le goût)
- 100 g de poudre de sucre non-raffiné.
- un peu de beurre pour graisser le moule
- de l'eau pour arroser au besoin

**Étape 1** : Si vous utilisez des arachides non grillées, mettez une casserole à fond épais sur une flamme moyenne. Ajoutez des arachides dans cette poêle pour rôtir. Remuer et rôtir les cacahuètes à feu doux pendant environ 5 minutes ou jusqu'à ce qu'elles deviennent croquantes.

**Étape 2** : Mettez les arachides dans une assiette et laissez-les refroidir. Une fois refroidies, frottez les cacahuètes entre vos doigts pour retirer facilement les coques. Transférer les arachides sans coque dans une autre assiette.

**Étape 3** : Maintenant, mettez une casserole à fond épais sur une flamme moyenne et ajoutez-y de l'eau. Ajouter du jagré à cette poêle et remuer avec une cuillère ou une spatule jusqu'à ce que le sucre non-raffiné soit complètement dissous.

**Étape 4** : Laisser mijoter le sirop de sucre non-raffiné à feu moyen en remuant fréquemment. Continuez à mélanger jusqu'à ce que les petites bulles apparaissent et atteignent le stade de la bulles dures (120 - 130°C). Réduisez ensuite la flamme.

**Étape 5** : Maintenant, ajoutez les arachides et les cardamomes dans la casserole et retirez-les de la flamme. Laissez ce mélange refroidir un peu. Jusque-là, graisser un plateau ou une assiette avec de l'huile. Vous pouvez utiliser n'importe quel beurre pour le graissage mais vous pouvez utiliser un beurre parfumé pour plus de goût.

**Étape 6** : Une fois que le mélange d'arachides et de sucre non-raffiné est suffisamment

froid pour pouvoir être manipulé, le mouler en forme de médaillon et les conserver sur le plateau graissé pour les laisser refroidir davantage jusqu'à ce qu'ils durcissent.

**Étape 7 :** Vous pouvez ensuite les consommer ou les stocker dans un contenant hermétique. Ils peuvent être consommés durant les 2 semaines suivantes.

Malheureusement, il est déconseillé de manger des Chikkis pendant qu'on manipule de l'électronique car le sucre peut abîmer le matériel.

# Chapitre II


## Consignes générales

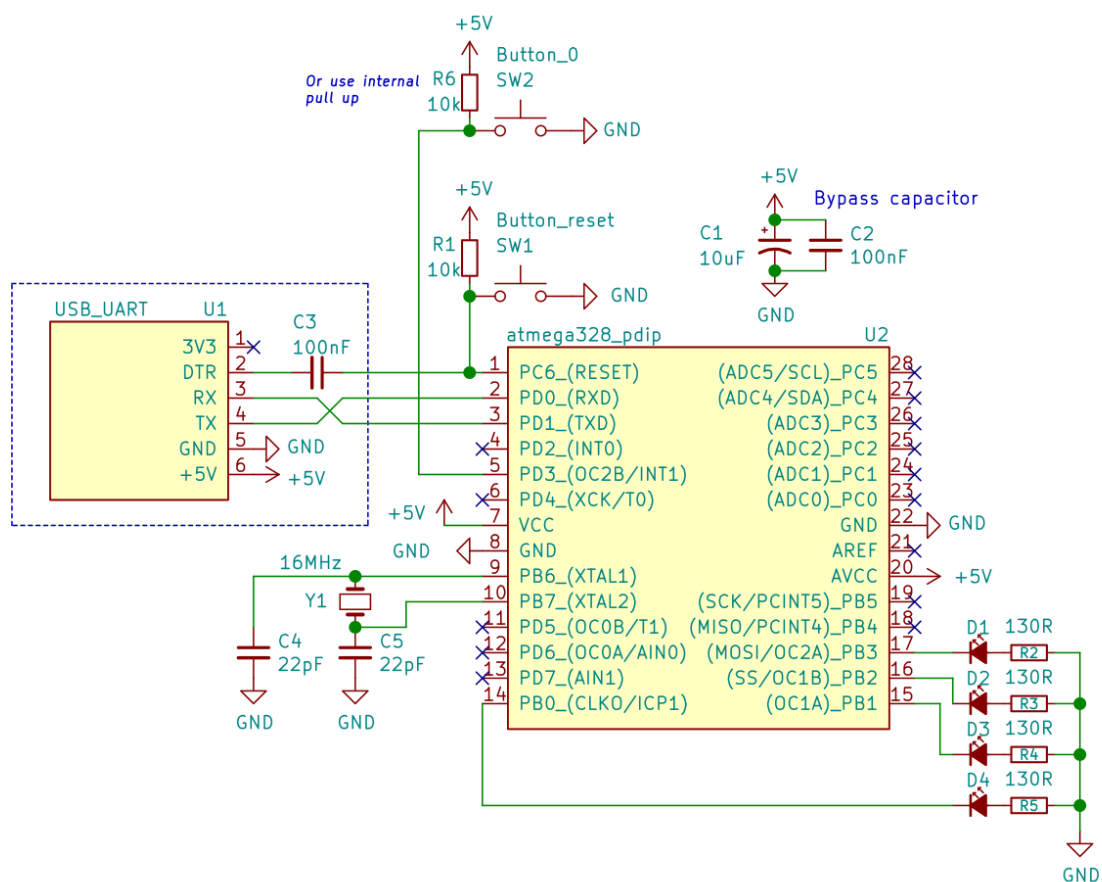
Sauf contradiction explicite, les consignes suivantes seront valables pour tous les Travaux Pratiques de Programmation embarquée.

- 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 : Breadboard


	Exercice : 00
	Breadboard
	Dossier de rendu : <i>ex00/</i>
	Fichiers à rendre :
	Fonctions Autorisées :



- Utilisez les composants, breadboard, et câble mis à votre disposition pour réaliser votre propre devkit

# Chapitre IV


## Exercice 01 : Makefile

	Exercice : 01
Makefile	
Dossier de rendu : <i>ex01/</i>	
Fichiers à rendre : <b>Makefile</b> , <b>main.c</b>	
Fonctions Autorisées : <b>avr-gcc</b> , <b>avr-objcopy</b> , <b>avrdude</b>	

- Le fichier **main.c** doit contenir un programme **main()** ne contenant aucune instruction.
- Écrire un fichier **Makefile** avec la règle **all** qui exécute la règle **hex** puis la règle **flash**.
- La règle **hex** compile le fichier **main.c** en **main.bin** avec une variable **F\_CPU** pour sélectionner la fréquence du microcontrôleur. Ensuite, génère le fichier **main.hex** à partir du fichier **main.bin**.
- La règle **flash** copie ce fichier **main.hex** dans la flash du microcontrôleur.
- Le **Makefile** devra également implémenter la règle **clean** qui supprime les fichiers **main.hex** et **main.bin**.

# Chapitre V

## Exercice 02 : Une lueur d'espoir

	Exercice : 02
Une lueur d'espoir	
Dossier de rendu : <i>ex02/</i>	
Fichiers à rendre : <code>Makefile</code> , <code>main.c</code>	
Fonctions Autorisées : <code>avr/io.h</code>	

- Vous devez écrire un programme qui permet d'allumer la LED D1 (PB3).
- Vous devez utiliser uniquement les registres AVR ( `DDRX` , `PORTX`, `PINX` ).




Vous devez à chaque fois expliquer la fonction et les valeurs assignées aux registres en commentaire !



# Chapitre VI

## Exercice 03 : Clignotant

	Exercice : 03
Clignotant	
Dossier de rendu : <i>ex03/</i>	
Fichiers à rendre : <b>Makefile</b> , <b>main.c</b>	
Fonctions Autorisées : <b>avr/io.h</b>	


- Vous devez écrire un programme qui permet d'allumer et éteindre la LED D1 (PB3) à une fréquence de 1Hz.
- Vous devez écrire un code qui permet d'attendre plusieurs centaines de millisecondes et qui sera inséré dans la boucle infinie du programme ( pas de TIMER hardware ).
- Le changement d'état de la LED doit être fait avec une unique [opération bitwise](#), il ne faut pas utiliser de condition ( if else ).
- Vous devez utiliser uniquement les registres AVR ( DDRX , PORTX, PINX ).



1Hz == ( allumé 0.5sec et éteint 0.5sec )  
for(;;) est fortement recommandé pour faire une boucle infinie

# Chapitre VII


## Exercice 04 : Lumos

	Exercice : 04
Lumos	
Dossier de rendu : <i>ex04/</i>	
Fichiers à rendre : <b>Makefile</b> , <b>main.c</b>	
Fonctions Autorisées : <b>avr/io.h</b>	

- Vous devez écrire un programme qui allume la LED D1 (PB3) lorsqu'on appuie sur le bouton.
- Lorsque le bouton est relâché, la LED s'éteint.
- Vous devez utiliser uniquement les registres AVR ( DDRX, PORTX, PINX ).

# Chapitre VIII

## Exercice 05 : Jour, nuit, jour, nuit

	Exercice : 05
Jour, nuit, jour, nuit	
Dossier de rendu : <i>ex05/</i>	
Fichiers à rendre : <b>Makefile</b> , <b>main.c</b>	
Fonctions Autorisées : <b>avr/io.h</b>	


- Vous devez écrire un programme qui inverse l'état de la LED D1 (PB3) à chaque fois que le bouton passe de l'état relâché à l'état appuyé.
- Vous devez utiliser uniquement les registres AVR ( DDRX, PORTX, PINX )



Attention aux rebonds du bouton !

# Chapitre IX

## Exercice 06 : Compteur binaire

	Exercice : 06
Compteur binaire	
Dossier de rendu : <i>ex06/</i>	
Fichiers à rendre : <b>Makefile</b> , <b>main.c</b>	
Fonctions Autorisées : <b>avr/io.h</b>	

- Vous devez écrire un programme qui chaque que vous pressez un bouton increment une valeur et qui l'affiche sur les LEDs D1 D2 D3 et D4 en binaire.
- Vous devez utiliser uniquement les registres AVR ( DDRX, PORTX, PINX )