



TP n°1:



Objectifs :

- Etre capable de créer un projet en C avec MPLAB**
- Savoir charger et exécuter un programme avec l'ICD2**
- Savoir rendre la carte cible autonome**
- Savoir utiliser une "Watch Window" pour voir les variables**
- Savoir utiliser les fonctions de la librairie xlcd100**
- Savoir utiliser les entrées/sorties logiques des ports A et B du 18F452.**
- Savoir coder des algorithmes de bases :**
 - comptage , mise à jour sur chagement de valeur, machine d'états**
 - Savoir utiliser le module ADC du pic.**
- Faire le lien entre le programme et le schéma de câblage de la carte.**

1/Création d'un projet

- En utilisant le cours, créer un projet C contenant un programme qui allume la led RB1 lors de l'appui sur RB0.
- Charger le programme sur la carte cible et exécutez le.
- Placer un point d'arrêt dans la clause d'allumage de la Led. Vérifiez l'arrêt lors de l'appui sur le bouton.

2/ Rendre la carte cible autonome

- Avec l'ICD2 sélectionné en tant que debugger (Debugger->Select Tool), le pic attend un ordre de lancement venant de l'ICD2 pour exécuter son programme. La carte n'est donc pas autonome. Pour le vérifier, débranchez l'alimentation de la carte et la liaison vers l'ICD2 et constatez que le programme ne démarre pas lors de la remise sous tension.
 - Sélectionnez maintenant l'ICD2 en tant que programmeur (notez le changement de la checksum). Chargez le programme, débranchez la liaison ICD et vérifiez qu'il démarre dès la mise sous tension.
 - Revenir à la configuration initiale pour la suite (sinon le debug est impossible).
- NB : il faut faire "Reset and Connect ICD) après chaque déconnexion de la carte cible.

3/ Faire clignoter la Led 3

Faire clignoter la led RB3 en inversant l'état de la sortie RB3 une fois sur 1000 tours de la boucle générale.

Par la suite cette led3 devra toujours clignoter montrant ainsi que la boucle générale ne sera jamais bloquée. La diminution de vitesse de son clignotement montrera l'augmentation du temps de la boucle générale au fur et à mesure que vous rajouterez des fonctionnalités à votre programme.

Placer un point d'arrêt dans la boucle et observez l'évolution de votre compteur de tours de boucle.



TP n°1:



La fenêtre Watch apparaît avec View->Watch...

- Ajoutez les registres PORTB, TRISB ainsi que votre compteur dans la fenêtre Watch. Les ports d'entrée/sorties seront observés en binaire tandis que le compteur sera visualisée en décimal.
- Exécutez le programme en pas à pas (Reset puis Step Over) et observer l'évolution des variable dans la fenêtre. Vérifier la cohérence avec le programme C exécuté.
- Modifiez la valeur du compteur pour mettre une valeur égale à 998 et reprendre l'exécution du programme. Constatez l'inversion de la led après deux appuis sur le bouton poussoir.

4/ Compteur / Décompteur

On souhaite incrémenter un compteur à chaque appui sur S3 et le décrémenter à chaque appui sur S2. Sa valeur maximum est 10, sa valeur minimum est 0.

Faire un grafctet et coder le en C.

Allumer la led RB2 si le compteur dépasse 5.

Suivre la valeur de ce compteur dans la Watch Window.

5/ Affichages sur le lcd

Pour améliorer l'affichage, on va utiliser les fonctions de la librairie xlcd100 pour écrire sur l'afficheur lcd.

Modifiez votre projet pour utiliser la librairie xlcd100 et affichez la valeur du compteur sur la première ligne de l'écran lcd sous la forme `cpt = XX`

5/ Voltmètre

Convertir la tension sur la broche RA0. Afficher le résultat (0-1023) sur la deuxième ligne de l'écran lcd.

Vous constatez sans doute que l'affichage n'est pas très joli et la boucle est ralentie. Cela est du au curseur qui n'arrête pas de changer de ligne et au temps d'affichage.

Pour y remédier, il ne faut faire un nouvel affichage que lorsque la valeur lue à changée.

Ajouter à droite de la valeur brute, la valeur de la tension en milivolts.
(Utiliser le format long pour éviter un dépassement de capacité)

Remplacer la valeur en mV par une valeur décimale en Volts.

(Bien qu'ils ne soient pas nécessaires dans cet exercice, notez que C18 peut calculer en réel mais printf ne peut pas les afficher)