

MI43 – TP2 (3 séances)

L'objectif de ce TP est d'implanter une application en mémoire sur une carte cible. L'application consiste à faire clignoter une led connectée à un port d'entrée sortie d'un microcontrôleur NXP LPC2478 à une fréquence variable.

1. Chargement d'une application et débogage

Vous disposez pour démarrer d'un projet uVision configuré permettant d'allumer une led. Ce projet a été configuré pour le microcontrôleur et la carte cible. Il comprend :

- La configuration mémoire pour l'édition de lien, permettant à l'éditeur de lien de créer un fichier image adapté à cette configuration mémoire : code et données en SDRAM ou code en flash et données en SDRAM
- Un fichier de « boot », permettant l'initialisation des vecteurs d'exception, l'initialisation des horloges de manière à ce que le processeur fonctionne sur le main oscillateur avec la PLL réglée pour une fréquence de travail à 72MHZ, l'initialisation de la pile et du tas pour l'application
- Un fichier comprenant l'application principale
- Un fichier de script permettant de configurer le microcontrôleur (horloges interne essentiellement) afin de pouvoir implanter l'application en mémoire et que cette dernière puisse être lancée directement sans effectuer de « reboot ». Ceci est particulièrement intéressant pour implanter le programme en SDRAM (phase de test) et le déboguer.

Copiez le projet dans votre répertoire personnel. Ouvrez le. Sélectionnez en « target » la SDRAM. Les autres target sont la « flash », la NAND flash et le simulateur. Ces autres cible ne seront pas utilisées.

Ouvrez le fichier contenant l'application.

La led notée P2.10 est reliée au port P2.10 du microcontrôleur, ainsi que le bouton poussoir juste en dessous. Cette application permet d'allumer cette led. Vous noterez que la led est allumée lorsque le port P2.10 est à l'état bas. Vous devez bien comprendre ce qui est fait dans ce programme.

Compiler le projet et lancer une session de débogage. La session de débogage télécharge le code et les données dans la SDRAM de la carte cible au travers de la liaison JTAG. La session de débogage s'effectue également à l'aide de cette liaison. Exécutez en pas à pas le programme.

2. Modification de l'application

2.1. Clignotement

Modifiez le programme principal de manière à faire clignoter la led à une fréquence visible à l'œil. On utilisera pour la fréquence de clignotement une variable globale. Cette variable globale sera fixe dans cette partie mais elle sera amenée à devenir variable sur les exercices suivants.

2.2. Lecture d'un port

Cette fois ci vous allez faire clignoter la led mais la fréquence pourra être sélectionnée par l'utilisateur à l'aide du bouton « joystick » de la carte. Le principe consiste à partir d'une fréquence de départ quelconque. À chaque impulsion vers le haut du bouton joystick, la fréquence doit augmenter. À chaque impulsion vers le bas, la fréquence doit diminuer.

Le bouton joystick comprend 5 contacteurs (centre, haut, bas, droite, gauche) reliés aux ports P2.22 à P2.27.

2.3. Utilisation des interruptions

Dans cette partie, vous allez utiliser les interruptions pour modifier la fréquence de clignotement. Le principe décrit au 2.2 reste le même. L'interface GPIO doit être configurée pour que lorsque le « joystick » est déplacé vers le haut ou vers le bas, une requête interruption soit générée vers le processeur au travers du contrôleur d'interruption VIC.

Interruption avec le GPIO

Les deux pattes du GPIO doivent être configurées pour générer une requête d'interruption sur un front descendant (falling edge). Cela est configuré au travers du registre IO2IntEn (P2EF) en positionnant un 1 sur le bit correspondant. Lorsqu'une requête d'interruption est en attente pour le GPIO2, le bit N°2 du registre « overall Interrupt Status register (IOIntStatus) » est à 1. Si un ou plusieurs des bits du port ont effectivement généré cette requête, les bits correspondant dans le registre « Interrupt Satus for falling edge register » (IO2IntstatF) est positionné à 1. Pour acquitter la ou les interruptions correspondant, il faut écrire un 1 dans le registre « Interrupt Clear register » (IO2IntClr) sur le bit correspondant à la patte qui a été la source de la requête d'interruption.

VIC

Les lignes d'interruption des GPIO2 et de l'interruption externe 3 sont partagées : EINT3 (N°17) . Pour un fonctionnement correct du système il convient de configurer correctement les différents registre du VIC (p114 et 115) :

- Charger l'adresse de la routine de traitement des interruption (ISR) dans le registre VICVectAddr17
- définir (optionnel) la priorité de cette interruption en mémorisant une valeur entre 0 et 15 dans le registre VICVectPriority17
- autoriser les interruptions de cette source en mettant à 1 le bit 17 du registre VICIntEnable
- autoriser les interruptions de ce niveau de priorité : registre VICSWPriorityMadrk. Par défaut tous les niveaux de priorité sont autorisés
- spécifier que cette interruption est de type IRQ plutôt que FIQ en plaçant le bit 17 du registre VICIntSelect à 0

Faites vérifier votre travail et votre code