

### **Laboratoire 6, MINI-PROJET** **Application embarquée**

Ce projet de session vaut 8% de la note totale du cours. À faire individuellement ou en équipe de 2, il sera corrigé sur un total de 10 points. Le projet est à remettre au plus tard mardi le 18 décembre 2016 à 16h30. La note sera 0% après, à moins d'entente avec l'enseignant, un assistant du cours ou avec la direction du programme de GEL/GIF.

#### **OBJECTIFS :**

Le mini-projet vise les objectifs suivants :

- Maîtriser les notions rencontrées dans les laboratoires précédents.
- Intégrer plusieurs interfaces ensembles pour réaliser un système complexe.
- Approfondir les connaissances en électronique numérique.
- Approfondir les connaissances en programmation embarquée.

#### **ÉNONCÉ DU PROJET :**

Le projet de session vous amènera à développer un système original basé sur des circuits d'interfaces. Cela vous introduira au développement de solutions embarquées alliant électronique et informatique tel que rencontré en industrie. De plus, il vous fera réaliser les implications et les difficultés rencontrées dans la conception d'un système complexe à plusieurs fonctions.

Le sujet du projet de session est libre. Néanmoins, il devrait respecter certains critères de base :

- Votre projet devrait interfacer *au moins* deux circuits ou périphériques externes. L'un de ces circuits ou périphériques doit être différent de ceux rencontrés dans les laboratoires.
- Votre projet devrait réaliser une fonction utile tout en représentant un certain défi. Activer une DEL comme « lampe de poche » est un projet trivial qui sera rejeté. Toutefois, soyez réalistes dans vos objectifs.
- Votre projet devrait réutiliser l'un des modules que vous avez conçu au laboratoire 3, 4 ou 5. Cela vous fera réaliser l'importance de concevoir un bon design pour maximiser sa réutilisation dans le futur.
- Votre projet devrait être relié à un thème du cours, c'est-à-dire à au moins un aspect précis des systèmes microprocesseurs et leurs interfaces.

Dans tous les cas, votre projet doit inclure un microprocesseur et de la programmation embarquée! **Le projet doit également être approuvé par l'enseignant ou par un assistant du cours.**

#### **EXIGENCES :**

Voici quelques exigences reliées à la remise du projet :

- Vous devez remettre les plans de votre circuit numérique (peut être fait à la main, mais doit être propre).
- Vous devez remettre un texte court (maximum deux pages) décrivant comment utiliser vos fichiers et les ressources utilisées par vos modules/pilotes.
- Vous devez fournir le code source de vos applications.
- Vous devez faire une courte présentation du système à un assistant du cours ou à l'enseignant et montrer votre code à l'occasion.

#### **PRÉPARATION :**

Du matériel est disponible au service technique de GEL/GIF pour les fins de ce projet. Alternativement, vous pouvez commander des pièces chez des détaillants d'électronique en ligne comme DigiKey et Mouser.

## **IDÉES :**

Le kit de développement des laboratoires comporte beaucoup de fonctionnalités intégrées qui n'ont pas été explorés dans les laboratoires, comme le bus CAN, le support USB, la génération de signaux PWM, l'exploitation d'un CNA, un encodeur de quadrature pour les capteurs de rotation, etc. De plus, vous pouvez utiliser des broches en GPIO pour interfacer directement un circuit utilisant une interface ou un protocole propriétaire.

Si vous êtes à court d'idées, voici quelques idées de pièces ou périphériques que vous pouvez interfacer pour votre projet :

- Potentiomètres
- Capteur de températures, de poids ou d'humidité
- Capteur de vitesse ou de rotation
- Photo détecteurs
- Moteurs pas à pas, moteur DC contrôlé par PWM ou autres moteurs
- Microphones et haut-parleurs
- GPS
- Inclinomètre
- Mémoires de toute sorte

Par ailleurs, plusieurs aspects logiciels n'ont pas été abordés dans les laboratoires, comme le chargement de programmes, les systèmes d'exploitation, le DMA, la protection de la mémoire... Vous pouvez combiner des parties de vos laboratoires à ces aspects.

## **PROJETS COMMUNS APPROUVÉS :**

Les projets suivants sont approuvés par défaut:

- Utiliser un RTOS commercial et accéder à au moins 3 périphériques.
- Monter un système d'exploitation préemptif avec le cœur ARM Cortex M4 et les exemples de code disponibles dans les notes de cours. Le projet est réussi si vous exécutez trois tâches avec des « `while(1){...}` » à tour de rôle.
- Coder un algorithme d'allocation dynamique de mémoire similaire à celui de M. Doug Lea (dlMalloc). Le projet est réussi si vos fonction New et Free fonctionnent correctement avec un tas alloué par segment et des « bins » pour accélérer la recherche du segment libre à utiliser.
- Créer une interface avec un LCD TFT (voir laboratoire 6 de 2015, facile par SPI)
- Créer un générateur d'onde avec du DMA (voir laboratoire 6 de 2014)

## QUELQUES CONSEILS POUR VOUS AIDER :

Voici une série de conseils pour vous donner un coup de pouce :

- Pour déverminer une application embarquée, les étapes suivantes sont habituellement recommandées :
  - Vérifier le matériel :
    - Une inspection visuelle sérieuse permet généralement de trouver la cause de 95% des problèmes matériels.
    - Vérifier les signaux d'alimentations
    - Vérifier les signaux critiques du circuit (horloge, reset et plus...)
    - Vérifier la fonctionnalité de tous les sous-systèmes du système.
  - Vérifier le logiciel
    - Le programme décolle-t-il?
    - Mes interruptions se produisent-elles comme attendu?
    - Est-ce que chaque branchement du code a été testé?
  - **Un oscilloscope ne ment jamais.** Pour savoir si une erreur est logicielle ou matérielle, utiliser un oscilloscope en combinaison avec du déverminage logiciel est une stratégie payante, surtout lors de programmation d'interfaces.
  - Si vous interfacez un module contenant plusieurs signaux (SPI, I2C, CAN), **un analyseur de logique peut grandement vous aider.** Cet appareil peut réaliser un *timing diagram* de votre circuit, ce qui permet de trouver les problèmes de synchronisation entre les différents signaux. Quelques analyseurs de logique sont disponibles au laboratoire d'électronique PLT-3103.
- Vous avez avantage à bien lire les *datasheets* de votre matériel et du microprocesseur AVANT d'acheter vos pièces et de commencer. Vous devriez aussi faire du design AVANT de commencer.
- Les modules Ethernet et USB sont des modules complexes qui peuvent demander beaucoup de travail. Bien que cela puisse vous permettre d'acquérir une expérience très précieuse pour votre future carrière, nous vous conseillons fortement de vous inspirer des exemples disponibles si vous utilisez ces interfaces ou de vous abstenir.
- Plusieurs exemples d'applications existent pour le cours. Vous pouvez vous inspirer d'un de ces exemples (ou d'une combinaison!) afin de réaliser votre projet.

Bon projet,

Etienne Tremblay