

## **CEG 3536 – Architecture d'ordinateur II**

### **Labo 5 – La conversion analogique numérique**

#### **Objectifs:**

Explorer l'utilisation d'un convertisseur analogique numérique et le principe de l'A/D.

#### **Préparation:**

- Lire la documentation au sujet de la conversion analogique numérique (Chapitre 17 du texte Cady et Manuel de référence du 9s12DG256 - ATD\_10B8C Block User Guide, V02.12)
- Complétez le travail de préparation selon les consignes données à la fin de ce document avant d'arriver au laboratoire.
- Faites la démonstration de votre système opérationnelle à la fin de la session de labo ou la semaine suivante.
- Soumettez votre code dans un fichier zip et votre rapport de laboratoire en format Word.

#### **Équipement utilisé:**

- Windows PC
- Carte Dragon-12

#### **Composantes utilisées sur le Dragon-12**

- Afficheurs à 4 segments
- Clavier
- Commutateur DIL (branché au port H)
- Afficheur LCD
- Haut-parleur de PC (PC Speaker)

#### **Autres composantes utilisées**

- 1 seulement – DEL rouge
- 1 seulement – Résistance de 4.7 KOhm

## Mesurer la température avec les thermistances

Les thermistances sont des appareils à semi-conducteur sensibles à la température, i.e., leur résistance varie avec la température. La carte Dragon-12 contient une puce thermistance linéaire active (Active Linear Thermistor IC), le LM45. La figure suivante montre l'appareil installé sur la carte Dragon-12 (pour de l'information complète, consultez la fiche technique fournie)

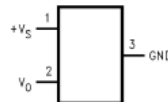


Figure 1. SOT-23  
Top View  
Package Number DBZ0003A

Le LM45 a été conçu pour être branché directement à un convertisseur numérique. Il possède les caractéristiques suivantes :

### DESCRIPTION

The LM45 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM45 does not require any external calibration or trimming to provide accuracies of  $\pm 2^\circ\text{C}$  at room temperature and  $\pm 3^\circ\text{C}$  over a full  $-20$  to  $+100^\circ\text{C}$  temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM45's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with a single power supply, or with plus and minus supplies. As it draws only  $120\ \mu\text{A}$  from its supply, it has very low self-heating, less than  $0.2^\circ\text{C}$  in still air. The LM45 is rated to operate over a  $-20^\circ$  to  $+100^\circ\text{C}$  temperature range.

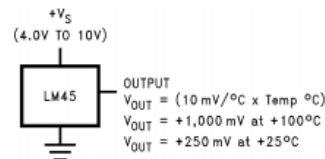
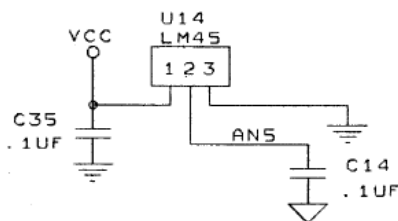


Figure 2. Basic Centigrade Temperature Sensor (+2.5°C to +100°C)

Sur la carte Dragon-12,  $V_S$  du LM45 est branché à 5 V et la sortie  $V_O$  (broche 2) à la broche AN5(PAD5) du microcontrôleur (voir le circuit ci-dessous). En plus,  $V_{HREF}/V_{DDA}$  sont branchés à 5V et  $V_{LREF}/V_{SSA}$  sont mise à terre (0V).



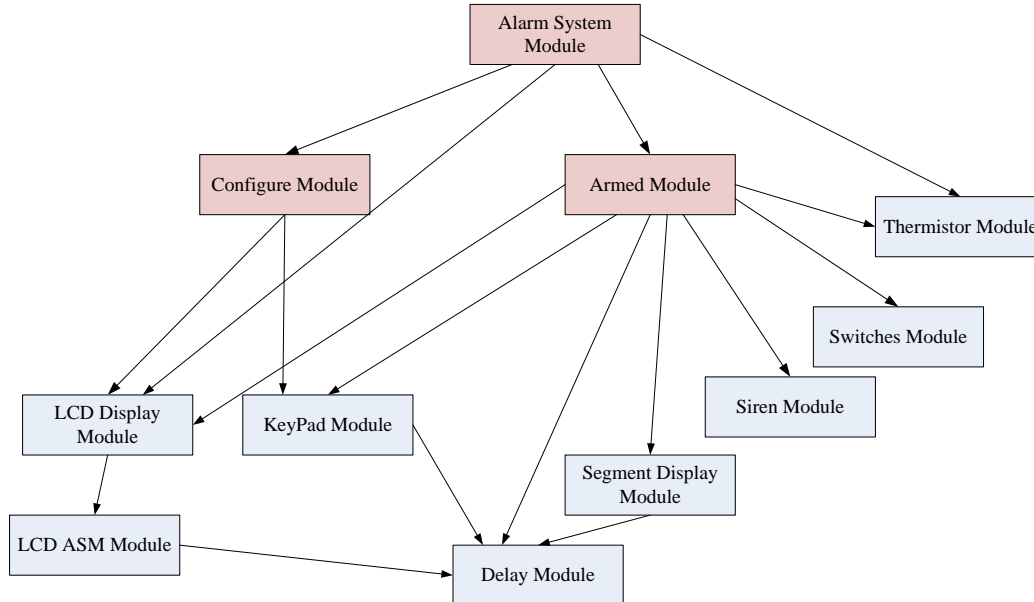
Votre tâche dans ce labo est de créer un module Thermistor qui permettra au système d'alarme de surveiller la température avec le LM45 et l'ATD. La température sera affichée sur les afficheurs à 7-segment (une valeur à 3 chiffres représentant la température au  $10^{\text{ième}}$  de degré prêt). En plus, vous utiliserez une DEL qui s'allument lorsque la température est au-dessus de  $27.0^\circ\text{C}$  (ce qui sera considéré un état dangereux).

**À FAIRE :** Utilisez l'information fournie pour faire un schéma électronique qui montre la configuration physique de l'ATD, le LM45, et la DEL (vous brancherez la DEL de la maquette à la broche 4 du Port P). Aussi développez des équations qui donnent les relations entre la température, le voltage appliqué à la broche PAD05, et la valeur produit par le ATD, étant donnée une conversion de 10-bits. Assurez-vous d'inclure votre schéma de circuit et vos équations dans votre rapport.

## Le système d'alarme

L'objectif principal du labo est d'ajouter un module au projet de système d'alarme du labo 4; le nouveau module utilise l'ATD pour surveiller la température. Une thermistance sur la carte Dragon-12 agit comme transducteur pour traduire la température à un signal électrique. Le nouveau module, le module Thermistor, offre le moyen de mettre à jour régulièrement une variable qui reflète la température mesurée. Ce module est intégré dans le projet de système d'alarme pour émuler un détecteur de fumé qui sonne l'alarme lorsqu'une condition de feu existe.

### Modules



Les modules Alarm System et Armed du labo 4 ont été modifiés de la façon suivante pour faire la surveillance de la température :

- 1) La température est surveillée continuellement, même lorsque le système est désarmé. Lorsque la température dépasse une valeur prédéfinie (27 °C pour rendre le labo pratique), la sirène est activée en appelant la fonction `triggerAlarm()` retrouvée dans le module Armed (la sirène est fermée en entrant un code valide sur le clavier). Il y a deux endroits où l'alarme est sonnée par une haute température : lorsque le système est désarmé (i.e. dans la fonction `main` du module Alarm System qui attend une commande) et lorsque le système est armé (après le délai de 10 secondes quand un code valide est tapé pour armer le système). En tout autre temps (durant les délais ou durant la configuration) l'alarme ne sera pas sonnée si la température augmente. Ceci est assez logique car durant ces moments, une personne est présente à la maison à la console d'alarme. Pour le labo, cette approche garde le code du projet plus simple.
- 2) La température est affichée continuellement sur les afficheurs à 7-segments en appelant les fonctions fournies dans le module Segment Display. Seulement durant le décompte lorsque le système est armé ou désarmé, les afficheurs ne montrent pas la température. Pour simplifier les modules, un programme de service d'interruption est responsable de lire la température du module Thermistor (avec `getTemp()`) et de mettre à jour les afficheurs en appelant les fonctions disponibles dans le module Segment Display. La variable `displayTempFlag` contrôle l'affichage de la température : si vrai (`TRUE` - 1), affiche la température et si faux (`FALSE` - 0) ne pas afficher la température. Le DEL est fonctionnel en tout temps.

## Modules nouveaux et mises à jour

Pour ce labo, vous devez modifier le module Segment Display et créer le module Thermistor selon les directives suivantes.

**Module Segment Display:** Ce module est une extension de celui du labo 4 ayant deux fonctions additionnelles pour gérer les points décimaux sur les afficheurs:

- `turnOnDP(int dNum)`: Cette fonction allume le point décimal de l’afficheur identifié par `dNum`. Notez que l’ajout d’un caractère à l’afficheur ne doit pas affecter le point décimal.
- `turnOffDP(int dNum)`: Cette fonction éteint le point décimal de l’afficheur identifié par `dNum`. Notez que l’ajout d’un caractère à l’afficheur ne doit pas affecter le point décimal.
- Rappelez-vous que la broche 4 du port P est utilisée pour le contrôle du DEL externe. Puisque le module Segment Display utilise les broches 0 à 3 du port P pour contrôler les afficheurs à 7-segments, vous devez possiblement revoir comment le module Segment Display manipule le port P pour ne pas affecter le DEL.

**Module Thermistor:** Utilisez le principe suivant de conception pour ce module: à tous les 100 ms générez une interruption qui démarre une conversion du signal de température et à la fin de la conversion générez une autre interruption à quel point, la valeur binaire de conversion est sauvegardée dans une variables. Le module offre deux points d’entrées (fonctions) :

- `void initThermistor(void)`: Cette fonction configure un canal de la minuterie pour générer une séquence d’interruption pour permettre à tous les 100 ms l’initiation de la conversion du signal de température sur le ATD. Notez qu’avec un tic de minuterie de 1.333  $\mu$ s, il n’est pas possible de générer une interruption à tous les 100 ms. Vous devez concevoir une logique qui permet d’utiliser de plus petits délais entre interruptions, et toutefois démarrer une conversion ATD à tous les 100ms.

Cette fonction initialise aussi le périphérique ATD pour convertie le signal de température reçu de la thermistance à une valeur de 10 bits et générer une interruption lorsque la conversion est complétée.

- `int getTemp(void)`: Cette fonction lit la valeur de conversion de 10 bits retrouvée dans une variable de conversion (voir les ISRs ci-dessous) et la traduit à une valeur entière entre 000 et 700 qui représente les 10<sup>ième</sup> de degrés (i.e. entre 00.0 °C et 70.0 °C). Cette valeur traduite est retournée.

Deux ISRs sont utilisés dans le module:

- ISR de la minuterie: Cet ISR démarre la conversion de l’ATD qui devra être configuré pour faire une seule conversion du signal de la broche PAD05. Tel qu’indiqué ci-dessus, la logique de l’ISR démarre la conversion à tous les 100 ms même si le ISR est exécuté plusieurs fois à l’intérieur de 100 ms.
- ISR de l’ATD: Cet ISR est exécuté à la fin d’une conversion et met à jour la variable de conversion (un emplacement de 16-bits en mémoire) avec la valeur de conversion de 10 bits présent dans l’ATD.

## Information additionnelle

**Projet test:** Un projet test est fourni pour tester les modules Thermistor et Segment Display. Le module Prog5Test utilise le canal 6 de la minuterie avec interruptions pour afficher la température à tous les 200 ms (ceci est la même approche utilisé dans le projet de système d’alarme). Le ISR met à jour les caractères à afficher sur les afficheurs à 7-segments (avec `setCharDisplay()` du module Segment Display) selon la valeur Celsius du module Thermistor (retournée par `getTemp()`). En plus, le DEL est

allumé lorsque la température est égale à ou dépasse 27 °C. Notez que le DEL est géré par ce module. La broche 5 du port P est utilisée pour contrôler le DEL (rappelez-vous que les broches 0 à 3 du port P sont utilisées par le module Segment Display).

**Les canaux de la minuterie:** Le canal 5 est utilisé pour contrôler le haut-parleur PC, et le module Alarm System utilise le canal 6 pour faire afficher la température sur les afficheurs 7-segments. Ceci vous laisse les canaux 0 à 4 pour répondre aux besoins des modules Delay, Segment Display, et Thermistor.

**Horloge ATD:** Utilisez une fréquence de 500 KHz pour l'horloge ATD avec un temps maximum d'échantillonnage. Il est évident que le temps et la fréquence d'échantillonnage n'est pas critique pour cette application.

## **À faire**

### **Préparation**

- a. Complétez une ébauche du rapport de laboratoire en Word qui comprend le suivant:
  - Dessinez un schéma électronique montrant les composantes additionnelles de la carte Dragon-12 utilisées dans la version du système d'alarme du labo 5, c'est-à-dire, comment la thermistance (montez aussi le branchement de  $V_{SSA}$ ,  $V_{DDA}$ ,  $V_{LREF}$ , et  $V_{HREF}$ ) est branchée au microcontrôleur et comment le DEL est branché au port P.
  - Conception des modules modifiés et nouveau. Ceci comprend le module modifié Segment Display, et le nouveau module Thermistor.
- b. Complétez le code projet avec CodeWarrior et assurez-vous que le projet compile sans erreurs.
- c. Vous devez montrer votre préparation complète à la session de laboratoire.

### **Au laboratoire**

- a. Ajoutez le DEL sur votre maquette et branchez-le au Port P.
- b. Chargez et roulez votre programme sur la carte Dragon-12. Déboguez votre logiciel. Montrez à votre AE le système opérationnelle. Une partie de votre note du laboratoire est affectée à la démonstration des fonctions suivantes:
  - i. La température est affichée sur les afficheurs à 7-segments, à l'exception du décompte des délais lorsque le système est armé ou désarmé.
  - ii. La sirène d'alarme est activée lorsque la température arrive à ou dépasse 27 degrés C si le système est désarmé (attend un 'a' ou 'c') ou si le système est armé (après un délai de 10 sec.).
  - iii. Aux autres temps (délais et la configuration de codes d'alarme), la température est affichée sur les afficheurs à 7-segments et le DEL montre une haute température, mais l'alarme n'est pas sonnée.
  - iv. Vous pouvez faire augmenter la température de la thermistance en tenant celle-ci entre votre pouce et doigt.

### **Le rapport de labo**

- a. Fournissez votre matériel de conception : circuit, conception de logiciel, etc.
- b. Fournissez aussi votre code source dans un fichier zip séparé.