



## Systèmes Embarqués 1 & 2

Classes T-2/I-2 // 2018-2019

### p.01 – Interruptions

#### Exercices

##### Exercice 1

Si le processeur se trouve en mode superviseur (svc), écrivez les instructions qui réalisent les opérations suivantes

- (a) Passer du mode superviseur en mode système (sys)
- (b) Placer la valeur 0x8020'0000 dans le pointeur de pile système (sys)
- (c) Lire le long mot (32 bit) placé au sommet de la pile système
- (d) Autoriser les interruptions IRQ, sans modifier les autres bits du registre d'état
- (e) Passer du mode système au mode utilisateur (usr)
- (f) Passer du mode utilisateur au mode superviseur

##### Exercice 2

Si le µP se trouve en mode superviseur, écrivez le segment de code qui démarre un programme chargé à l'adresse 0x8000'1000. Le programme s'exécute en mode utilisateur. La pile utilisateur débute à l'adresse 0x8030'0000.

##### Exercice 3

Indiquez l'effet des instructions suivantes, en respectant l'ordre chronologique des instructions

```
msr    cpsr_cxsf, #0x93
eors    r0, r0
msr    cpsr_c, #0x93
msr    cpsr_c, #0x13
msr    spsr_cxsf, #0x93
movs    pc, lr
msr    cpsr_c, #0x91
msr    cpsr_c, #0x92
msr    cpsr_c, #0x90
msr    cpsr_c, #0x93
```

##### Exercice 4

Soit un mini système d'exploitation possédant deux jeux de 5 routines utilitaires. Ces routines sont accessibles par un programme utilisateur via l'instruction SVC #1 pour le 1<sup>er</sup> jeu et SVC #5 pour le 2<sup>e</sup>. Le programme utilisateur place dans le registre de donnée R0, le numéro d'identification (utility ID) avant l'appel au système d'exploitation. Ecrivez le segment de code qui implémente une telle approche du côté système d'exploitation et du côté utilisateur.

##### Exercice 5

Concevez un programme qui utilise les interruptions pour mesurer le temps qui s'écoule entre deux



pressions de touche successives. La valeur minimale, la valeur maximale et le nombre de pressions de touche doivent être calculées.

Deux périphériques sont à disposition:

- (a) Une horloge temps réel (real-time clock).  
Ce périphérique émet une interruption à intervalle régulier
- (b) Un clavier ou une souris.  
Le périphérique utilisé émet une interruption à chaque pression d'une touche

La routine de service de l'horloge temps réel incrémente, à chaque interruption, une variable globale clock\_count.

#### Exercice 6

Quel est le résultat de l'instruction ci-dessous si l'interruption « irq\_h » survient lors de son exécution ?

```
int len = 0;

len += 2;    // irq_h survient ici ?

void irq_h()
{
    len += 4;
}
```

#### Exercice 7

Le  $\mu$ P est en train d'exécuter une application dans le mode utilisateur et les interruptions FIQ et IRQ sont autorisées. Que se passe-t-il et/ou quel est l'état du processeur, si

- (a) une instruction SVC est appelée par l'application
- (b) une interruption IRQ est levée puis une interruption FIQ est levée
- (c) une interruption FIQ est levée puis une interruption IRQ est levée

#### Exercice 8

Expliquer la différence dans le traitement d'une interruption vectorisée et le traitement d'une interruption non-vectorisée. Quels sont les avantages et désavantages de ces deux types de traitement.