

Contrôle continu Programmation des Systèmes

Exercice 1:

1. Décrivez ce qui est représenté sur la figure 1.
2. Dites ce qui se passe lorsque le processeur exécute l'instruction *MOVS pc,lr*.
3. Comment est provoqué un changement de mode du processeur en mode interrupt (depuis user)?
4. Décrivez les actions exécutées par le processeur lors d'un changement en mode interrupt.

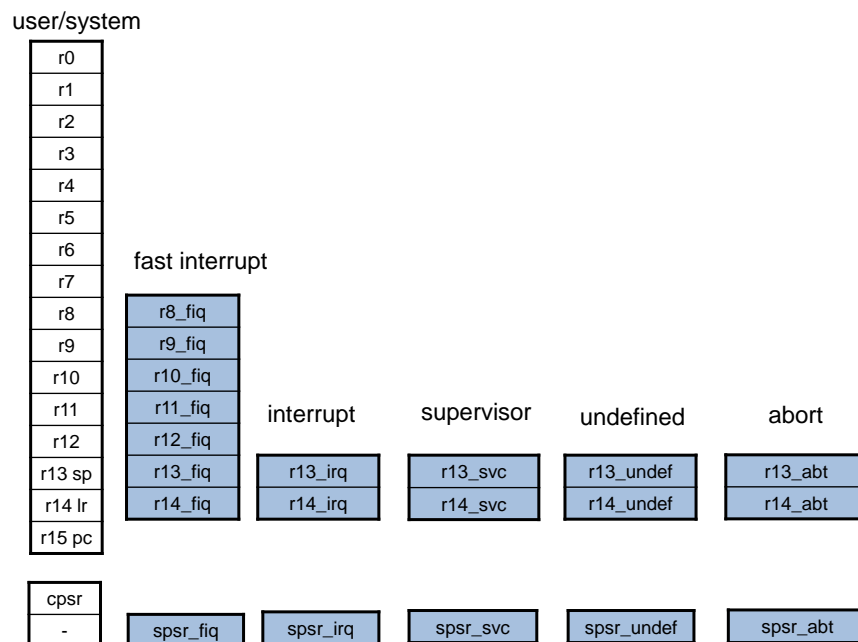


Figure 1:

Exercice 2:

1. Enumérez les 4 types de piles.
2. Ecrivez les instructions assembleurs qui implémentent les instructions *push* et *pop* pour chacun des types de pile.
3. Pour chacune des instructions du point précédent, dites quel mode d'adressage du processeur convient le mieux pour l'implémenter.

Exercice 3:

1. Décrivez la séquence d'actions exécutés par le processeur pour exécuter l'instruction *SWI #123*.
2. Expliquez comment faire pour que, suite à l'exécution de l'instruction *SWI*, le processeur exécute le handler d'interruption ci-dessous.
3. Décrivez les actions de chacune des instructions du handler.

```
SWI_handler  STMFD    sp!, {r0-r12, lr}
             LDR      r10, [lr, #-4]
             BIC      r10, r10, #0xff000000
             BL       routine
             LDMFD    sp!, { r0-r12, pc}^
```

Exercice 4:

1. Ecrivez une routine *unsgdiv*: qui prend comme arguments deux nombres non-signés codés sur 32 bits dans les registres r_0 et r_1 et retourne dans r_0 la division entière de r_0 par r_1 (r_0/r_1). Implémentez un algorithme simple qui compte le nombre de fois que vous pouvez soustraire r_1 à r_0 . Votre routine ne modifie pas d'autres registres de la procédure appelante que r_0 .
2. Ecrivez une routine *sgnsgdiv*: qui calcule la division entière de r_0 par r_1 pour r_0 et r_1 des nombres signés sur 32 bits. Calculez le signe et ensuite appelez la routine *unsgdiv*.
3. Ecrivez un programme qui calcule la division entière de deux nombres signés de 16 bits déclarés statiquement par les directives de compilation *num1: .hword 0x...* et *num2: .hword 0x....*. Votre programme appelle votre routine *sgnsgdiv*.

Exercice 5: Les drapeaux du cpsr sont placés après l'exécution de l'instruction *CMP r0,r1*.

1. Montrez que la condition $Z=1$ est bien équivalente à $r_0 == r_1$.
2. Montrez que la condition $C=0$ est bien équivalente à $r_0 < r_1$ non-signé.
3. Montrez que la condition $N=V$ est bien équivalente à $r_0 \geq r_1$ signé.

Exercice 6: Pour chacune des instructions ci-dessous, décrivez l'exécution par le processeur et indiquez le mode d'adressage.

1. *ldr r0, [r1, #4]!*
2. *ldrsb r0, [r1], #1*
3. *movs r0, r1, asr #1*