

**Exercice 1 (8 points)**

Répondre aux questions suivantes (5 ou 6 lignes par réponse au maximum) :

1. Qu'est ce que le contexte d'un programme. Donnez un exemple.
2. Un programme principal, écrit en langage C, appelle une procédure (sous-programme écrit en assembleur) avec un seul paramètre. Le sous-programme (procédure) utilise et préserve les registres R5, R6 et R8. Ecrivez le début et la fin du sous-programme en assembleur.
3. L'instruction ARM7TDMI est codée sur 32 bits, quelle est la technique utilisée par ARM pour permettre de réaliser l'instruction « LDR R1,=0x12345678 » en 32 bits ?
4. Expliquez les avantages et les inconvénients des instructions MOV et LDR. Illustrez par des exemples.

**Exercice 2 (6+6 points)**

1. Ecrire en assembleur ARM, une fonction qui compte tous les caractères ASCII visibles d'une chaîne autres que les chiffres décimaux (0 à 9). Le paramètre est passé par la pile et le nombre de caractères est retourné dans R0. La chaîne de caractères est définie par :

```
ChaîneCar    DC8 "Asm_ZZ1-2011";
```

2. Ecrire un programme en langage assembleur permettant de détecter la position du premier bit à '1' d'un mot de 32 bits (ETAT) en balayant de droite à gauche ( $\leftarrow$ ). La position du premier bit à '1' doit être sauvegardée dans un registre à la fin du programme (exemple : ETAT=0x01230014=B0000 0001 0010 0011 0000 0000 0001 0100), 2 est la position du premier bit à '1' car la position du bit poids fort est de 31 et celle du bit poids faible est de 0 (Figure 1).

31

0

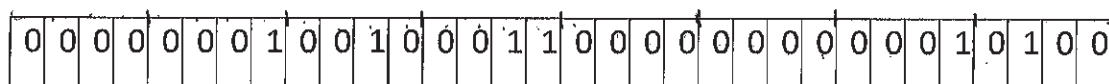


Figure 1 : Les positions des bits du mot ETAT