

Médian IF40

I Comptage du nombre de valeurs paires et impaires (7 points)

Le tableau TABLE déclaré à l'aide de la directive `.bss` contient 1000 entiers signés de 16 bits. On désire déterminer le nombre de valeurs paires et de valeurs impaires que contient ce tableau. Les résultats seront stockés dans les variables de 16 bits PAIRE et IMPAIRE déclarées également à l'aide de la directive `.bss`.

- I.1 Indiquer comment déclarer TABLE, PAIRE et IMPAIRE à l'aide de la directive `.bss`.
- I.2 Donner l'organigramme PARITE déterminant le nombre de valeurs paires et impaires contenues dans TABLE et stockant ces résultats dans les variables respectives PAIRE et IMPAIRE.
- I.3 Ecrire en assembleur le sous-programme PARITE traduisant l'organigramme ci-dessus.

II Gestion d'un clavier 16 touches (13 points)

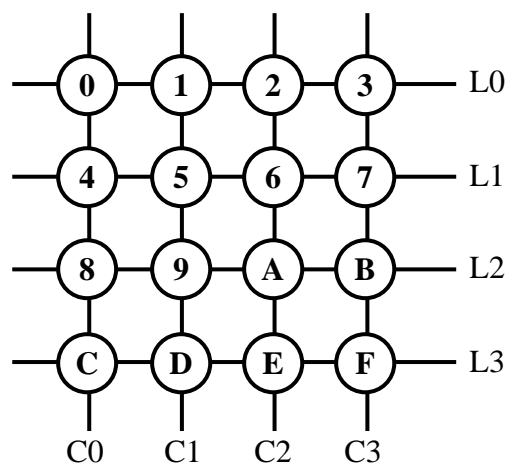
Un clavier 16 touches comme celui représenté ci-contre est constitué d'une matrice 4x4 de touches. Son principe consiste à mettre à la masse (état logique 0) la connexion de ligne (L0 à L3) et de colonne (C0 à C3) correspondant à la touche enfoncée.

Par exemple l'appui sur la touche 9 met L2 et C1 à l'état logique 0. Toutes les autres lignes et colonnes restent à l'état logique 1.

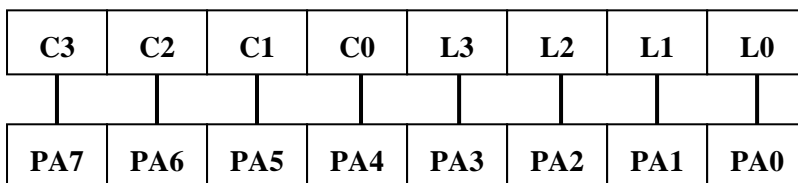
On peut alors facilement déterminer le numéro de la touche enfoncée par la relation :

$$\text{touche} = 4 * (\text{n}^\circ \text{ de ligne à } 0) + \text{n}^\circ \text{ de colonne à } 0$$

Dans l'exemple précédent on a bien : $\text{touche} = 4 * 2 + 1 = 9$



La détection de la touche enfoncée par lecture de l'état des lignes L0 à L3 et C0 à C3 se fait par l'intermédiaire du port A du DSP contrôleur TMS320LF2407, qui est relié au clavier de la manière suivante :



D'autre part, les 8 broches du port B sont reliées à des LEDs afin de permettre un affichage en binaire de la dernière touche enfoncée. L'état du port B doit donc représenter en permanence la valeur binaire de la dernière touche enfoncée.

- II.1 Indiquer comment déclarer la variable *touche* occupant 1 mot de 16 bits
- II.2 Ecrire en assembleur le sous-programme INITPA initialisant le port A uniquement.
- II.3 Ecrire en assembleur le sous-programme INITPB initialisant le port B uniquement.
- II.4 Donner l'organigramme CLAVIER réalisant une lecture du clavier et plaçant dans la variable *touche* la valeur de la touche enfoncée (valeur de 0 à F). Si aucune touche n'est enfoncée, la variable *touche* ne doit pas être modifiée.
- II.5 Ecrire en assembleur le sous-programme CLAVIER correspondant.
- II.6 Donner l'organigramme AFFIC réalisant l'affichage en binaire de la valeur *touche* sur les LEDs reliées au port B.
- II.7 Ecrire en assembleur le sous-programme AFFIC correspondant.
- II.8 Ecrire en assembleur le programme principal qui initialise les ports, puis appelle en boucle les sous-programmes CLAVIER puis AFFIC.