

TD 2

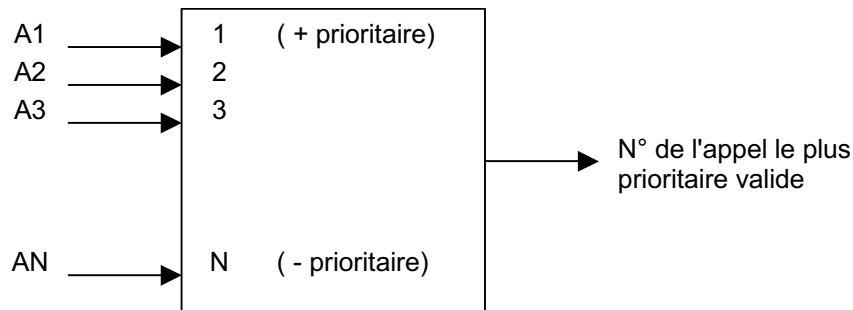
Architecture des Calculateurs

A. SYSTEMES D'INTERRUPTIONS

L'unité centrale peut à tout moment recevoir des appels de l'extérieur. Ces appels correspondent à des demandes d'exécution de programmes.

Exemple : appel d'alarme, appel du clavier ... etc.

L'unité centrale contient un arbitre combinatoire qui reçoit les appels sur ses entrées et les classe par ordre de priorité selon l'entrée sur laquelle se trouve l'appel.



Besoin :

- Suspendre un programme pour en lancer un autre
- Pouvoir faire cela n'importe quand
- Pouvoir revenir au programme suspendu

Armement et Masquage des niveaux :

On peut armer ou désarmer un niveau (programmation dans un registre).

On peut masquer ou démasquer un niveau (programmation dans un registre).

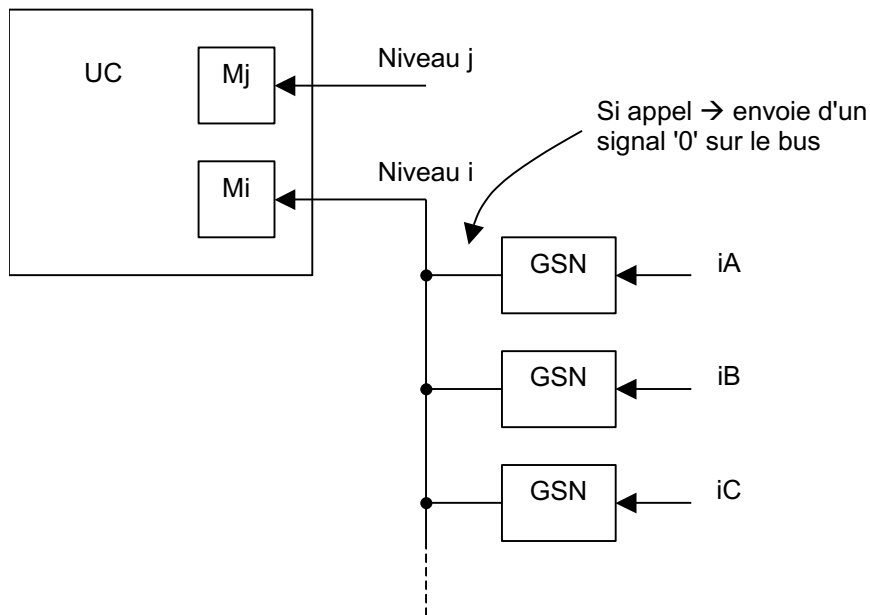
Principes fondamentaux :

- P1 : Un appel ne doit pas être ignoré sauf si le niveau est désarmé.
- P2 : Un appel reste mémorisé tant qu'il n'a pas été exécuté.
- P3 : Un niveau masqué est réceptif à l'appel mais n'est pas servi.

A chaque appel d'un programme + prioritaire, il faut sauver le contexte du programme en cours, par exemple dans une pile.

Décentralisation des interruptions :

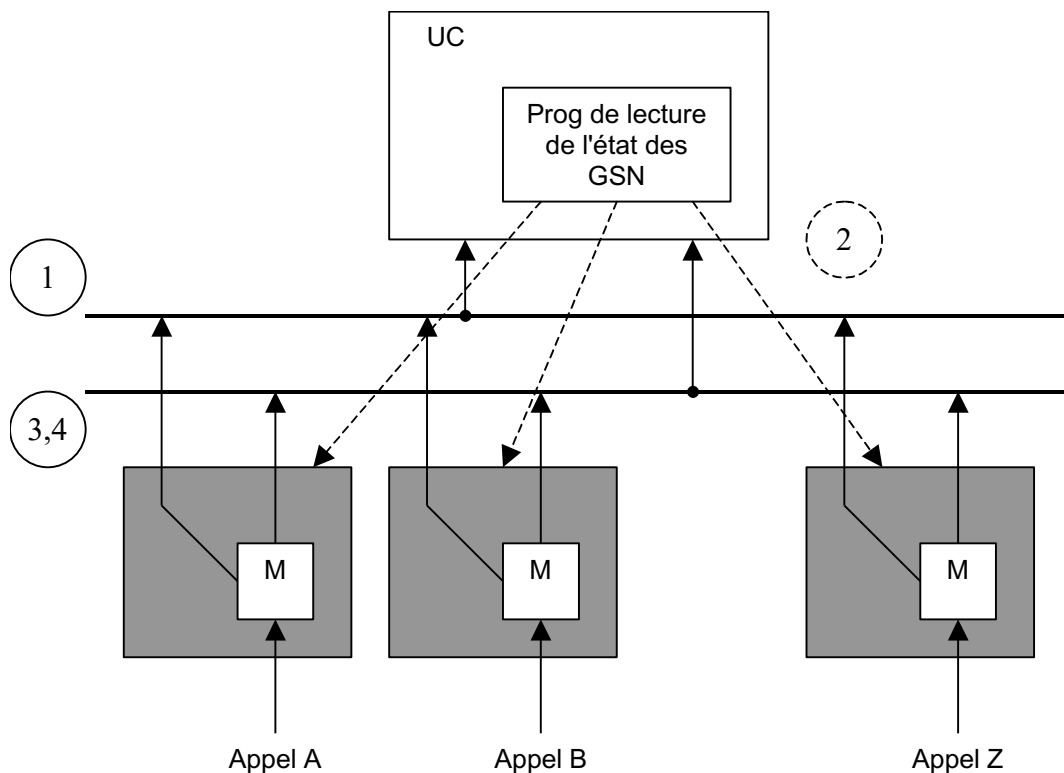
But : Augmenter le nombre d'appels possibles sans augmenter le nombre de connexions de l'unité centrale. Pour cela, on utilise des gestionnaires de sous-niveau.



Comment reconnaître d'où vient l'appel :

- Par scrutation : MOTOROLA
- Par vecteur : INTEL

1. Reconnaissance par scrutation



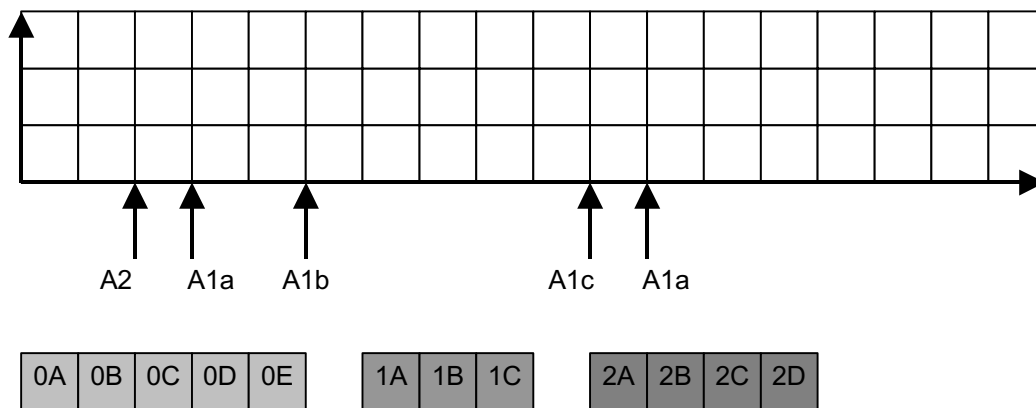
- 1- L'appel déclenche un programme.
- 2- Ce programme vient lire successivement tous les GSNs pour connaître leur état (Appel ou non). Leur lecture RAZ l'appel.

- 3- Si le GSN a un appel, le programme se branche sur une branche contenant le programme P_n demandé par l'appel n.
Sinon le programme passe à la lecture du GSN_{n+1} suivant.
- 4- Quand il les a tous explorés, il se termine par une instruction "Fin d'Interruption".
- 5- Si un GSN a été appelé après, l'unité centrale repart en interruption.

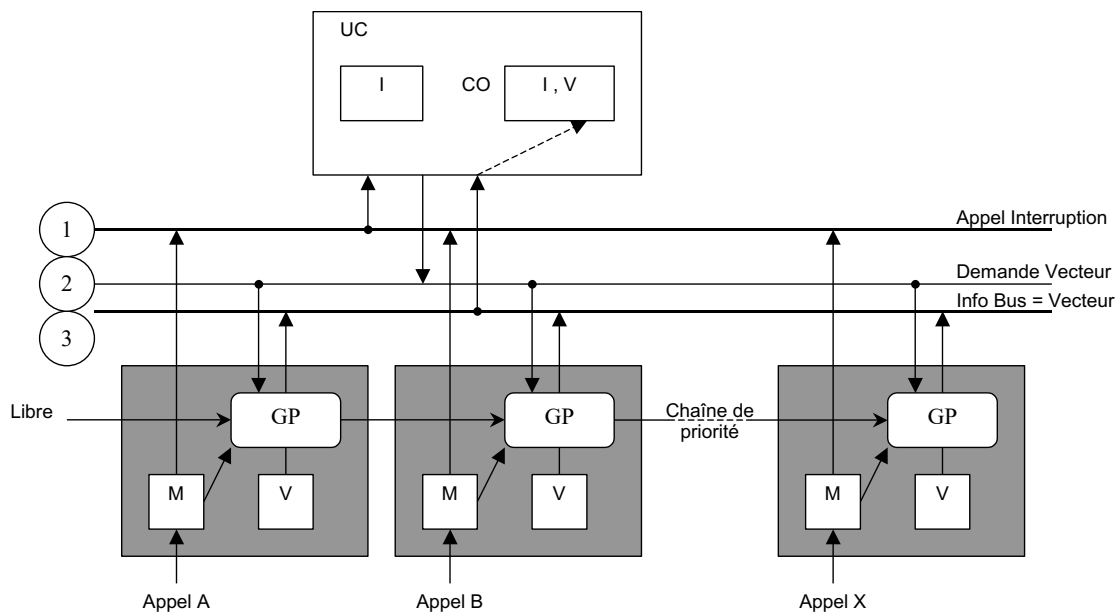
Exercice 1 :

Remplir le chronogramme suivant sachant que :

- Le niveau le plus prioritaire est le niveau 2.
- Le programme de niveau 0 commence à l'instant t₀ et dure 5 unités de temps.
- Le programme de niveau 1 dure 3 unités de temps.
- Celui du niveau 2, 4 unités de temps.



2. Reconnaissance par vecteurs

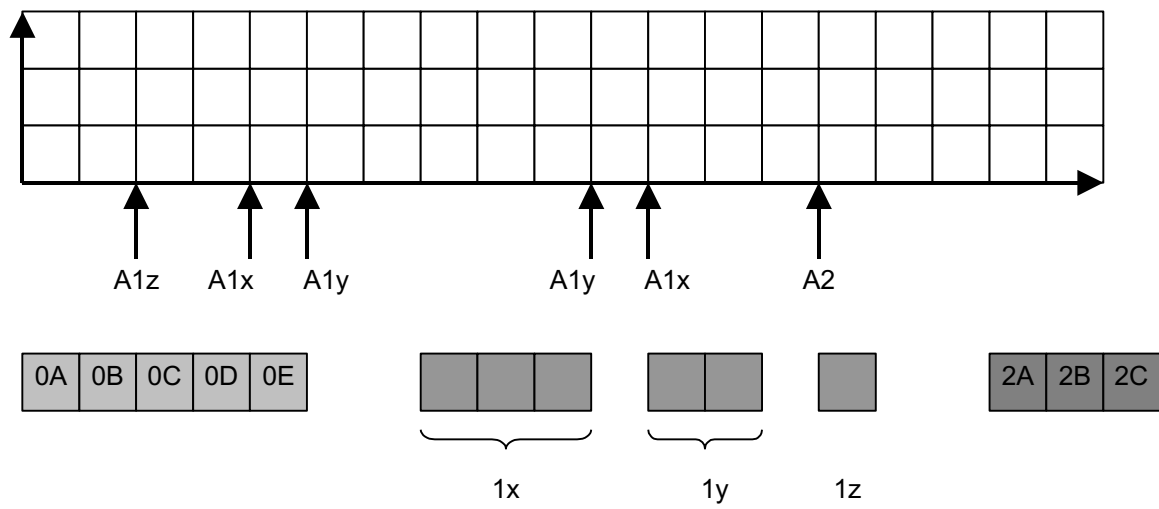


- 1- Un appel arrive sur un GSN. Il met la "valeur active" sur la ligne 1 commune, ce qui appelle le niveau N de l'unité centrale.
- 2- L'unité centrale envoie un signal 2 de demande de vecteur de GSN appelant, à tous les GSN.
- 3- Le GSN appelant présente sur le bus d'information 3, son vecteur. Le vecteur est lu par l'unité centrale.
- 4- L'unité centrale concatène "I" et "V" pour constituer l'adresse du programme qui va être lancé. C'est ce programme qui est appelé par l'appel du sous-niveau.
- 5- Que se passe-t-il si plusieurs GSN reçoivent un appel en même temps ?
Un seul GSN doit envoyer son vecteur (sinon il y aurait conflit sur le bus). Cette sélection d'un GSN se fait par la chaîne de priorité.

Exercice 2 :

Remplir le chronogramme suivant sachant que :

- Le niveau le plus prioritaire est le niveau 2.
- Le programme de niveau 0 commence à l'instant t_0 et dure 5 unités de temps.
- Le programme de niveau 1 appelé par le GSN_x dure 3 unités de temps
- ... GSN_y 2 ...
- ... GSN_z 1 ...
- ... GSN_y 2 ...
- Le programme de niveau 2 dure 3 unités de temps
- La chaîne de priorité impose l'ordre suivant : $1_x > 1_y > 1_z$



Exercice 3 :

On considère une machine possédant trois niveaux d'interruptions.

Elle doit exécuter un programme de niveau 0 et des programmes de niveaux 1, 2 et 3. Le niveau 3 est le plus prioritaire. Les figures 1A et 1B donnent deux cas de répartition des appels dans le temps.

La figure 2 donne la longueur de chaque programme P0, P1, P2 et P3, dans chaque niveau, ainsi que les parties qui masquent ou démasquent des niveaux.

Le programme de niveau 0, P0, par exemple, dure 4 unités de temps et masque le niveau 2 dans la 2^{ème} unité de temps. Le programme de niveau 3, P3, dure 3 unités de temps et démasque le niveau 2 pendant la 2^{ème} unité de temps.

Remplissez les figures 1A et 1B en répartissant dans le temps les programmes P0, P1, P2 et P3.

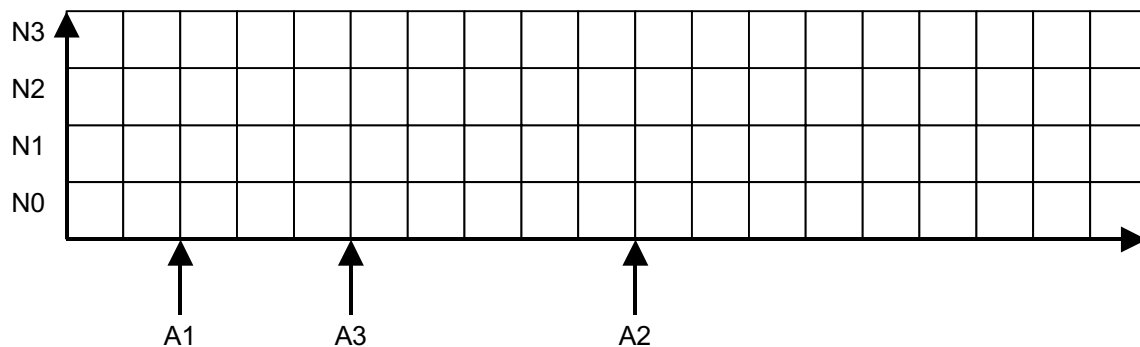


Figure 1A

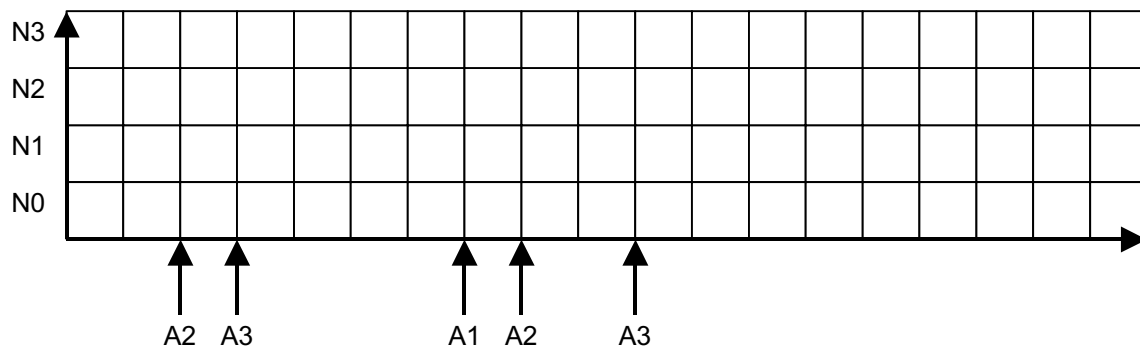


Figure 1B

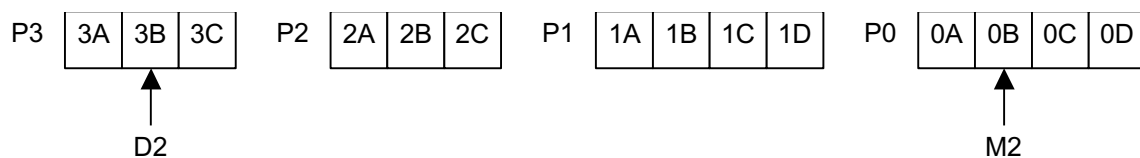


Figure 2

Légende :

- An : Appel du niveau n
- Mn : Masquage du niveau n
- Dn : Démasquage du niveau n