

Les interruptions

Ce mécanisme a été imaginé pour permettre à un dispositif extérieur d'interrompre le déroulement normal du processeur pour lui faire exécuter un traitement spécifique. Un programme en arrière plan peut ainsi être interrompu pour permettre d'activer un travail en avant plan; il sera ensuite poursuivi à la fin de ce dernier. Un périphérique peut signaler la fin de l'opération d'entrées-sorties demandée précédemment, permettant ainsi au système de temps partagé de reprendre l'exécution du programme qui avait demandé cette opération.

Le mécanisme est obtenu en modifiant légèrement le fonctionnement du décodeur-séquenceur du processeur, par introduction d'une troisième phase, et qui devient le suivant:

- lecture mémoire à l'adresse indiquée par le compteur ordinal, et rangement du résultat dans le registre instruction,
- décodage de cette instruction pour en exécuter les différentes phases,
- s'il y a une demande d'interruption, alors la prendre en compte.

La prise en compte de l'interruption peut se faire de différentes façons. Puisqu'il s'agit d'interrompre le déroulement normal des instructions, il "suffit" de modifier le compteur ordinal. Cependant comme on désire reprendre ultérieurement le programme interrompu, il faut aussi "sauvegarder" la valeur de ce compteur. Dans la plupart des processeurs, ce n'est pas jugé tout à fait suffisant. En général, le constructeur définit un *PSW (Program Status Word)*, ou *mot d'état du programme*, contenant en particulier le compteur ordinal ainsi que certains indicateurs sur l'état courant du processeur. La prise en compte d'une interruption par le processeur consiste alors à ranger en mémoire ce PSW dans un emplacement déterminé ou repéré par un registre particulier utilisé comme un pointeur de pile, et à charger un nouveau PSW soit fixe, soit obtenu depuis la mémoire.

Pour éviter de ranger inutilement en mémoire des registres, ce qui serait coûteux en temps, le processeur ne range que le minimum lors de la prise en compte de l'interruption. Si on désire pouvoir reprendre ultérieurement le programme interrompu, les instructions qui constituent le traitement de l'interruption que nous appellerons *sous-programme d'interruption*, doit ranger en mémoire le contenu de l'ensemble des registres du processeur qui ne l'ont pas été par la prise en compte de l'interruption, pour permettre cette reprise ultérieure. C'est ce que l'on appelle la *sauvegarde du contexte*. Dans certains cas, on pourra ne faire qu'une sauvegarde partielle des seuls registres modifiés par le sous-programme d'interruption. Lorsque le contexte est sauvegardé, le sous-programme analyse la cause de l'interruption, et effectue le traitement approprié. Il peut ensuite restituer le contexte du programme interrompu (ou celui d'un autre programme lorsque l'interruption correspond à la fin d'une opération d'entrées-sorties pour celui-ci). Il se termine par une instruction spéciale qui demande au processeur de restituer le PSW du programme interrompu, entraînant sa reprise effective (figure 2.7).

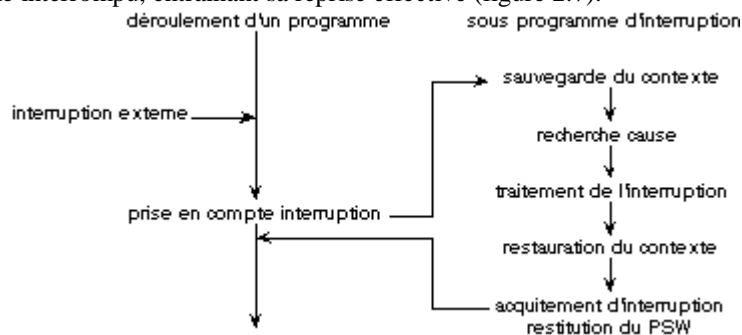


Fig. 2.7. Déroulement d'un sous-programme d'interruption.

Le schéma de fonctionnement du mécanisme d'interruption ainsi décrit met en avant une difficulté si une nouvelle interruption survient alors que le traitement de la précédente n'est pas terminée. Si on ne prend pas de précaution, le sous-programme d'interruption est de nouveau appelé avant sa fin, entraînant la sauvegarde du contexte du sous-programme d'interruption et la perte du contexte du programme initialement interrompu. Ce problème est résolu par le biais du *masquage des interruptions*, qui interdit au processeur la prise en compte d'une interruption si elle est masquée. Cela n'annule pas la demande d'interruption proprement dite, qui n'est pas considérée par le processeur. Lorsque, ultérieurement, une instruction sera exécutée par le processeur pour démasquer cette interruption, celui-ci pourra alors la prendre en compte. Cela permet ainsi au logiciel (en général le système d'exploitation) de contrôler les moments où une telle prise en compte pourrait perturber son bon fonctionnement. En particulier, le masquage automatique des interruptions par le processeur lors de la prise en compte d'une interruption évite de le réexécuter une nouvelle fois avant qu'il ne soit terminé, à moins que le

programmeur n'ait lui-même demandé le démasquage des interruptions dans le sous-programme après avoir pris les précautions nécessaires à cette nouvelle interruption (*sous-programme réentrant*).

Certains processeurs ont plusieurs "fils d'interruption" qui sont masquables individuellement. Ils correspondent alors à des interruptions hiérarchisées, c'est-à-dire qu'à chaque fil est attribué un numéro appelé *niveau de priorité d'interruption*, permettant au processeur de savoir laquelle il y a lieu de prendre en compte si plusieurs sont présentes au même moment. Ce dispositif est complété par un registre spécialisé du processeur (qui fait partie du PSW) qui contient le niveau de priorité de l'interruption en cours de traitement. Le processeur ne prend en compte une interruption que si elle est non masquée, et de priorité supérieure au niveau de priorité courant. Cela permet de satisfaire les demandes plus urgentes que celle que l'on est en train de traiter tout en évitant les problèmes de réentrance évoqués plus haut.

1 Notion d' appel système

1.1 Mode maître-esclave

Le chapitre précédent a introduit la nécessité de contrôler les actions d'un programme vis à vis de son environnement direct. En général, il est nécessaire d'empêcher le programme d'un utilisateur de perturber le fonctionnement global du système ou les programmes des autres utilisateurs. Cela implique qu'il n'ait pas accès à l'ensemble des ressources de la machine, mais seulement à celles qui lui ont été allouées en propre. La solution la plus couramment adoptée est de distinguer deux modes de fonctionnement du processeur, le mode *maître* et le mode *esclave*. Dans le mode maître, le processeur a accès à toutes les ressources de la machine. Dans le mode esclave, certaines instructions lui sont interdites. Pour cela, le fonctionnement du décodeur-séquenceur du processeur est légèrement modifié, par introduction d'une phase de contrôle intermédiaire:

- lecture mémoire à l'adresse indiquée par le compteur ordinal, et rangement du résultat dans le registre instruction,
- si le processeur est en mode esclave, vérification que l'instruction est autorisée,
- décodage de cette instruction pour en exécuter les différentes phases,
- s'il y a une demande d'interruption, alors la prendre en compte.

L'indicateur du mode de fonctionnement maître/esclave fait partie du mot d'état programme. De cette façon, ce mode est lié au programme en cours d'exécution. En particulier, la prise en compte d'une interruption chargeant un mot d'état programme spécifique à l'interruption, le sous programme correspondant s'exécutera dans le mode défini par ce mot d'état et non par celui du programme interrompu. La reprise de ce dernier restaurera le mode qui lui est attaché.

1.2 Déroutement

Si le programme d'un utilisateur fonctionne en mode esclave, sous un jeu d'instructions réduit, néanmoins, certaines actions nécessaires à son bon fonctionnement nécessitent de disposer du jeu complet, et doivent donc être exécutées en mode maître. Evidemment le changement de mode doit être contrôlé. En général, ceci se fait au moyen d'une instruction spéciale, qui va changer le mot d'état programme du processeur, selon un principe voisin de la prise en compte des interruptions, c'est-à-dire, rangement du mot d'état courant et chargement du mot d'état lié à la fonction demandée et restauration du mot d'état initial à la fin de la fonction. C'est pourquoi, ce mécanisme prend parfois la dénomination d'*interruption programmée*. On dit encore *déroutement* ou *appel superviseur*.

Un déroutement est aussi très voisin d'un appel de sous programme : sauvegarde du compteur ordinal, dont la valeur est remplacée par l'adresse du sous programme, et restauration de la valeur sauvegardée à la fin du sous programme. Il y a, cependant, deux différences essentielles :

- Le déroutement modifiant le mot d'état programme permet un changement de mode, un masquage des interruptions, ou un changement du niveau de priorité courant du processeur.
- L'adresse du sous programme exécuté n'est pas contenue dans l'instruction de déroutement elle-même, mais dans le mot d'état programme chargé. Ceci oblige le passage par des points de contrôles qui vérifieront les valeurs des paramètres de la fonction avant de l'exécuter. De plus, le programme n'a pas à connaître les adresses qui sont à l'intérieur du système et aux quelles il n'a pas accès.

2 Conclusion

- + Un ordinateur est constitué d'un bus qui relie les divers organes, d'un processeur, d'une mémoire centrale et de périphériques.
- + Un processeur est constitué d'une unité arithmétique et logique, de registres et d'un décodeur-séquenceur qui le contrôle. Il lit en mémoire l'instruction située à l'adresse définie par le compteur ordinal, l'exécute, puis recommence. Son langage est binaire.
- + Les entrées-sorties programmées sont obtenues par un programme qui effectue le transfert des informations entre la mémoire et un périphérique, une par une.
- + L'accès direct à la mémoire est un dispositif qui assure le transfert des informations entre la mémoire et un périphérique par vol de cycle.
- + La prise en compte d'une interruption par le processeur consiste à ranger en mémoire le mot d'état programme et à le remplacer par un nouveau. Le compteur ordinal fait partie de ce mot d'état programme. Le reste du contexte doit être sauvegardé par le sous-programme d'interruption, pour permettre sa restitution en fin d'exécution de ce sous-programme.
- + Le masquage des interruptions est le mécanisme qui permet au logiciel de contrôler la prise en compte des interruptions par le processeur.
- + Le processeur peut travailler selon le mode maître ou le mode esclave dans lequel certaines instructions sont interdites. Des instructions spéciales d'appel superviseur permettent de changer de mode de façon contrôlée.
- + Les périphériques de dialogue homme-machine, lorsqu'ils sont simples, peuvent être gérés par des transferts programmés, éventuellement régis par interruption. Cependant lorsque leur nombre augmente sur une même machine, la charge de traitement peut nécessiter l'utilisation de processeurs spécialisés appelés frontaux. Par ailleurs, dans les stations de travail, l'affichage est le résultat de modifications du contenu de la mémoire partagée par le dispositif de balayage vidéo.
- + Les périphériques magnétiques, bandes ou disques, nécessitent l'utilisation soit de l'accès direct à la mémoire, soit d'un processeur spécialisé d'entrées-sorties. Ce dernier a l'avantage de permettre de décharger le processeur principal d'une partie de la gestion du périphérique.
- + La communication entre machines utilise des protocoles normalisés, qui permettent de s'affranchir des contraintes matérielles et prendre en compte la complexité du réseau d'interconnexion. L'information est découpée par l'émetteur en paquets indépendants, et reconstruite par le récepteur à l'aide du protocole.
- + L'une des fonctions du système est de décharger le programmeur du souci de la diversité des périphériques, par la fourniture d'un nombre limité d'interfaces standards et uniformes.