

Les Interruptions et Le PIC-8259

1 Introduction.

Nous étudions dans cette partie les interruptions matérielles (ou externes), c'est à dire déclenchées par le matériel (hardware) extérieur au processeur. Les interruptions permettent au matériel de communiquer avec le processeur. On peut distinguer deux sortes d'interruptions.

Les interruptions matérielles et les interruptions logicielles. Les interruptions internes, dues à l'exécution du programme (division par zéro, dépassement de capacité d'un registre, tentative d'accès à une zone mémoire protégée, ..) sont considérées comme interruptions logicielles. Ces interruptions sont souvent appelées exceptions. Les interruptions ainsi que les exceptions sont toutes traitées de la même façon. La différence vient surtout de la source d'enclenchement de l'interruption.

En effet les interruptions matérielles et les exceptions peuvent se produire n'importe quand, tandis que les interruptions logicielles se produisent à un endroit précis du code où se trouve une des instructions `int` ou `into` ou `trap` (ça dépend du microprocesseur).

De plus, les interruptions matérielles peuvent être masquées (interdites ou autorisées) à l'aide de l'indicateur IF (Registre d'état). Une interruption externe est signalée au processeur par un signal électrique sur une des ses pattes prévues pour ceci.

Lors de la réception de ce signal, le processeur "traite" l'interruption dès la fin de l'instruction en cours d'exécution. Ce traitement consiste à exécuter un programme qui est appelé automatiquement lorsque l'interruption survient. L'adresse de début du programme doit être préalablement stockée dans une table dite des vecteurs d'interruptions. Lorsque la routine d'interruption est exécutée une instruction `IRET` permet au processeur de reprendre l'exécution à l'endroit où il avait été interrompu. Il est aussi possible d'ignorer l'événement et passer normalement à l'instruction suivante dans le cas des interruptions masquables. Il est en effet parfois nécessaire de pouvoir ignorer les interruptions pendant un certain temps, pour effectuer des traitements plus urgents. Lorsque le traitement est terminé, le processeur démasque les interruptions et les prend alors en compte. La procédure d'interruption est donc une tâche associée à un événement externe ou interne.

2 Les interruptions matérielles.

Cas des processeurs de la famille INTEL.

Les interruptions matérielles sont générées par les périphériques : souris, clavier, disque, horloge temps réel, etc. À la différence des interruptions logicielles, elles peuvent être autorisées ou interdites au moyen de l'indicateur IF du registre d'état «EFLAGS ». Comme le Pentium n'a que deux entrées d'interruption matérielle, NMI et INTR, on doit lui adjoindre un contrôleur d'interruption programmable afin de pouvoir disposer de plusieurs sources d'interruption avec une gestion de priorité. C'est le PIC (Programmable Interrupt Controller 8259A).

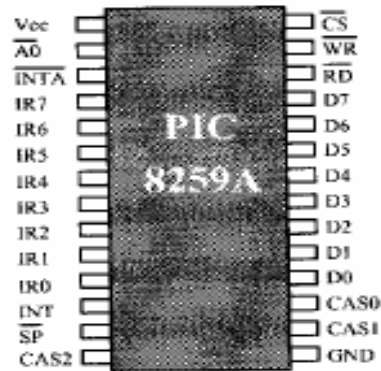


Figure 3.5 Le contrôleur d'interruption PIC 8259A.

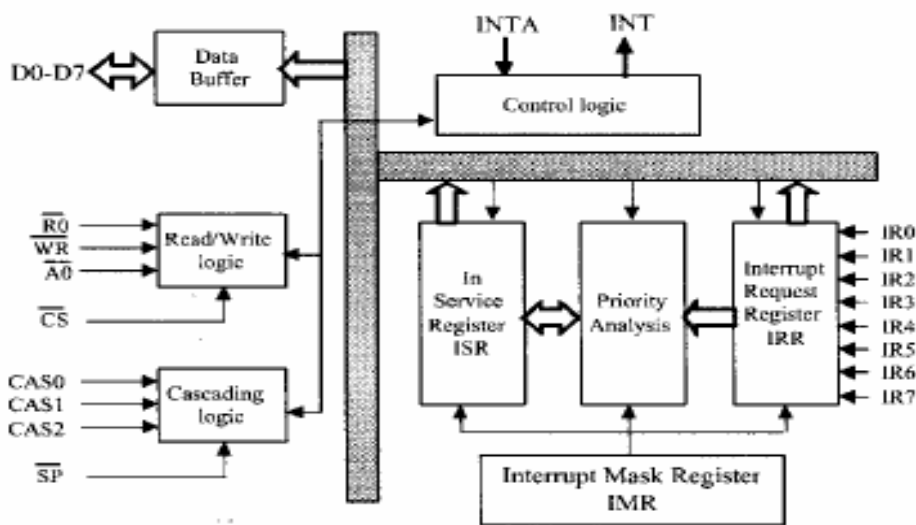


Figure 3.6 Schéma interne du PIC 8259.

3. Le 8259A peut accepter les interruptions de 8 sources externes, et on peut gérer jusqu'à 64 sources différentes en cascader plusieurs 8259A. Il gère la priorité entre les interruptions simultanées, interrompt le processeur et lui passe un code pour identifier la source d'interruption. Une source d'interruption est connectée à chacune des 8 entrées IR0 à IR7. Selon sa priorité, et s'il n'y a pas d'autre interruption en cours, le PIC décide s'il peut transmettre l'interruption au CPU. Si oui, il affirme la ligne INT, qui est connectée à l'entrée INTR (Interrupt Request) du CPU. Si le CPU est prêt à accepter l'interruption, il répond au PIC via la ligne INTA (Interrupt Acknowledge). Le PIC répond à son tour en envoyant le numéro d'interruption sur les lignes D0 à D7. Ce numéro est un index dans la table des vecteurs d'interruption. Le CPU est maintenant prêt à appeler le sous-programme de traitement d'interruption approprié. Quand le sous-programme de traitement d'interruption a terminé son exécution, il en avertit le PIC pour qu'il puisse permettre à d'autres interruptions d'atteindre le CPU. Les interruptions matérielles servent à une gestion efficace des périphériques d'entrée/sortie. Dans un ordinateur moderne, il y a continuellement des interruptions matérielles. Le temporisateur, l'horloge temps réel, les touches du clavier, les mouvements et les clics de la souris, le modem, l'imprimante, les disques durs et souples, le CDROM, sont tous des sources d'interruptions. Les circuits contrôleurs de périphériques contiennent plusieurs registres d'interface avec le CPU. Il y a habituellement un registre de contrôle, un registre d'état, et un ou plusieurs registres de données. Pour

connaître l'état d'un périphérique, le CPU peut interroger le registre d'état. L'approche des drapeaux (flags) consiste à interroger de façon répétitive le registre d'état, pour savoir où le périphérique est rendu dans le transfert des données. A-t-il reçu une nouvelle donnée ? A-t-il terminé la transmission de la dernière donnée envoyée ? etc. Cette approche consomme trop de temps de la part du processeur. L'approche interruption est beaucoup plus performante.

*Le périphérique envoie une interruption matérielle au processeur quand il a quelque chose à signaler.

*Le processeur interrompt alors la tâche en cours, enregistre en mémoire l'état de la machine, et vient interroger le registre d'état du périphérique, pour connaître la cause de l'interruption.

* Il effectue ensuite le traitement approprié et élimine la source de l'interruption. Ce traitement consiste, par exemple, à lire la donnée reçue dans le registre de réception et à l'inscrire en mémoire, ou à lire en mémoire la prochaine donnée à transmettre et à l'inscrire dans le registre de transmission du périphérique.

*Le processeur retourne ensuite à la tâche interrompue après avoir restauré l'état de la machine qu'il avait enregistré au moment de l'interruption.

Le sous-programme de traitement a donc 4 tâches à exécuter : -Sauvegarder l'état de la machine en empilant les registres susceptibles d'être modifiés dans le sous-programme de traitement d'interruption (ISR). Ceci inclut EFLAGS (registre d'état);

-Interroger le registre d'état du périphérique pour savoir quelle opération effectuer ;

-Éliminer la source de l'interruption en effectuant l'opération d'entrée-sortie ;

-Restaurer l'état de la machine et retourner à la tâche interrompue en dépilant les registres empilés.

4. Interruptions dans le PC :

Depuis le premier 8088 qui a équipé les PC, les interruptions sont normalisées pour les périphériques standard. La dernière composante d'un système à microprocesseur est l'accès direct à la mémoire, le DMA (Direct Memory Access). Cette fonction permet de demander au processeur de se déconnecter des bus de données, adresse et contrôle pendant qu'un périphérique prend le contrôle de la mémoire. L'arrivée du pseudo Plug & Play de Windows 95/98 et Win2000 couplé avec le plug & Play du Bios facilite dans certains cas le paramétrage des interruptions, mais il est parfois nécessaire de reprendre les configurations manuellement lorsque le nombre de périphériques augmente. Un périphérique est donc déterminé de manière hardware par une plage d'adresses (parfois une seule adresse), une interruption et éventuellement un canal DMA. Le 8088-8086 utilisait un 8259 d'INTEL, avec 8 niveaux d'interruptions notées de IRQ0 à IRQ7. Le 80286 jusqu'aux 486 utilisaient 2 contrôleurs 8259 chaînés ce qui permet d'avoir 16 sources d'interruptions. Ils sont notés de IRQ0 à IRQ15, mais avec l'IRQ9 réservée pour le chaînage. Depuis les Pentium (et un nouveau circuit gestionnaire d'interruptions), cette IRQ9 est accessible aux périphériques, mais ceci explique que peu de périphériques ne se branchent par défaut sur ce niveau d'interruption. L'interruption non masquable (NMI) est également utilisée pour des erreurs de parité ou erreurs de connecteurs entrées / sorties. Le tableau ci-dessous reprend l'ensemble des adresses et interruptions normalisées du PC, mélangées avec les interruptions utilisées par défaut actuellement. Souvent les interruptions 14 et 15 sont utilisées par le contrôleur de disque dur (une par port IDE). L'interruption 12 est utilisée par la souris PS2. Le port USB utilise également par défaut cette interruption 12. Néanmoins, le port USB passe en 11 si une souris PS2 est connectée. Ces interruptions peuvent être vérifiées par les outils fournis par les systèmes d'exploitation et pour certaines, elles sont affichées lors du démarrage de la majorité des PC. La vérification dans les PC sous DOS passe par le programme MSD.exe situé en c:\DOS. Dans le cas de Windows cet utilitaire est situé dans le panneau de configuration. Il permet les mêmes vérifications, mais de manière plus étendues. Depuis Win98, un nouvel utilitaire, Outils systèmes Microsoft, situé dans le dossier Outils Système du menu Démarrer permet une meilleure connaissance sur le système en cours. En regardant ce tableau, si un PC manque d'interruptions libre, il est possible d'en récupérer en supprimant des ports de communication. Si vous n'utilisez pas de ports série (uniquement USB), vous pouvez rendre inactif

COM 1 et Com 2 (disabled) et récupérer les interruptions 3 et 4. Idem pour le port parallèle. Si vous utilisez 2 ports parallèles, vous pouvez mettre l'interruption de LPT2 en 5 (selon la norme) ou en 7. Si vous utilisez des disques SCSI, pourquoi pas rendre inactif les ports IDE 1 et IDE2 dans le BIOS (récupération des INT 14 et 15), ... Comme les systèmes

Fonction	IRQ	Adresse	Commentaires
Horloge système	0		Ne peut être modifié
Clavier	1		Ne peut-être modifié
Contrôleur d'IRQ programmé	2		Ne peut être modifié
Com 2	3	2F8-2FF	Modifiable avec prudence
Com 4		2E8 - 2EF	
Com 1	4	3F8-3FF	Modifiable avec prudence
Com 3		3E8-3EF	
Libre	5		Généralement utilisé par la carte son, également LPT2 en 278
Contrôleur lecteur de disque	6		Modifiable avec prudence
LPT 1	7	378 -37A	Modifiable avec prudence. On peut mettre LPT1 et LPT2 sur la même interruption, mais pas d'impressions simultanées
LPT 2		(pour les produits de	

Tableau. Ensemble des interruptions normalisées du PC

5. Les Interruptions Logiciel

Comme nous l'avons écrit plus haut, les interruptions logicielles, sont aussi appelées trappes ou déroutements. Elles incluent aussi les fautes et les arrêts. Une faute se produit quand le processeur détecte une erreur durant le traitement d'une instruction. Par exemple, division par 0, opcode invalide, etc. En mode réel (Mode 8086), quand le processeur rencontre une instruction telle que `int immed8` (`immed8` : constante sur un octet), il va lire la table des vecteurs d'interruption IVT (Interrupt Vector Table). Cette table de 1 Ko est située à l'adresse 0000:0000. Chaque entrée de la table contient le numéro de segment de 16 bits et l'offset de 16 bits pour l'adresse d'un sous- programme de traitement d'interruption (Interrupt Service Routine ou ISR). `Immed8` est utilisé comme indice dans cette table (on doit le multiplier par 4 pour avoir l'adresse physique correspondante). De plus, le registre d'état «`FLAGS` » et les registres de segment `CS` et d'instructions `IP` sont empilés dans cet ordre exact, puis les indicateurs `TF` et `IF` sont mis à 0. Le sous-programme d'interruption devra se terminer par l'instruction `IRET`, pour dépiler correctement ces

paramètres. L'exécution se poursuit ensuite à l'adresse contenue dans la table. Par exemple, int 8 ira lire le vecteur situé à l'adresse 32 (0x20) et branchera à l'adresse qui y est contenue. A cette adresse doit débiter un sous-programme de traitement d'interruption. Certaines ISR font partie du système d'exploitation et sont déjà définies, comme celles qui correspondent aux INT 0x21 de MS-DOS. Cette trappe y sert de mécanisme d'appel au système d'exploitation. Si vous voulez définir votre propre sous-programme de traitement d'interruption, il suffit d'aller écrire son adresse à l'endroit approprié dans la table des vecteurs d'interruption avant de l'utiliser.