

Entrées/Sorties

Une des tâches fondamentales du système d'exploitation est le contrôle des périphériques d'entrée/sortie et la gestion des flux de données en entrée ou en sortie.

Il doit :

- Émettre des commandes vers ses périphériques
- Intercepter les interruptions
- Gérer les erreurs

I - L'aspect matériel

Une des principales difficultés vient de la grande diversité des périphériques (clavier, disque, ...), or l'interface doit autant que possible tous les englober avec le même traitement.

1) Les différents périphériques

Du point de vue du traitement de l'information, on peut séparer les périphériques d'entrée/sortie en deux classes :

- Les périphériques par blocs, en général de 512o - 32Ko (disques)
- Les périphériques par caractères appelés aussi périphériques alpha-numérique qui traitent un flot sans structure fixe (clavier, souris, imprimante)

Il reste des périphériques à part qui n'entrent pas dans ces catégories comme l'horloge et l'écran avec mappage de mémoire... Mais cette classification peut suffire comme base d'interface entre le logiciel du système d'exploitation et un périphérique d'entrée/sortie autonome.

Une des difficultés vient des grandes différences de débit des flux de données suivant les périphériques. A titre d'exemples :

Périphérique	Débit de données
Clavier	10 o/s
Souris	100 o/s
Imprimante laser	100 Ko/s
Scanner	400 Ko/s
USB (Universal Serial Bus) 1.1	1,5 Mo/s
Disque IDE (Integrated Drive Electronics)	5 Mo/s
CD-ROM 40x	6 Mo/s
Clé USB 2.0 en écriture	8 Mo/s
Bus ISA (Industry Standard Architecture)	16,7 Mo/s
Disque EIDE (Extended IDE)	16,7 Mo/s
FireWire (IEEE 1394) (bus série)	50 Mo/s
USB 2.0 (taux maximal)	60 Mo/s
Écran XGA	60 Mo/s
Disque SCSI (Small Computer System Interface)	80 Mo/s
Bus PCI (Peripheral Component Interconnect)	528 Mo/s

2) Contrôleur de périphérique

Une unité d'entrée/sortie est en général constituée de deux parties :

- Un composant mécanique : le périphérique lui-même
- Un composant électronique : le contrôleur de périphérique (puce placée sur une carte qui contrôle physiquement le périphérique appelé aussi unité d'échange)

C'est le contrôleur qui reçoit les ordres du système d'exploitation, comme lire une donnée par exemple. Le contrôleur doit fournir une interface simplifiée au système d'exploitation : par exemple l'ordre du S.E. « lire secteur 345 du disque 2 » doit être traduit en numéro de cylindre/numéro de secteur/numéro de tête par le contrôleur. Le contrôleur doit ensuite vérifier que le secteur n'est pas défectueux, connaître la position du bras, lui ordonner de se déplacer de tant de cylindres, attendre que le secteur recherché soit sous la tête de lecture, ordonner la lecture, convertir dans un tampon le flot binaire série en bloc d'octets, détecter d'éventuelles erreurs. Enfin si le bloc complet est déclaré sans erreur, il peut ensuite être copié en mémoire principale.

Le système d'exploitation ne voit que l'interface avec le contrôleur de périphérique, pas avec le périphérique mécanique. Le contrôleur possède un petit nombre de registres qui servent à communiquer avec le système d'exploitation (registre d'état, registre de commande, registre de données). Chaque type de contrôleur est spécifique, il faut donc un logiciel spécial pour communiquer avec le contrôleur, il est appelé pilote de périphérique (device driver) et doit être adapté au système d'exploitation (par exemple pilote HP laser jet pour Windows XP pro, pilote HP laser jet pour UNIX, ...).

En général les pilotes sont placés dans le système d'exploitation pour fonctionner en mode noyau (protection).

3) Entrées/sorties mappées en mémoire

Par le biais de l'écriture dans les registres du contrôleur, le système d'exploitation commande au périphérique de transmettre des données, de s'activer ou de se désactiver... En lisant ces registres le système d'exploitation peut également savoir dans quel état est le périphérique, s'il est prêt, occupé, ... De plus le périphérique possède souvent un tampon de données dans lequel le système d'exploitation peut lire ou écrire (les pixels affichés à l'écran par exemple proviennent de la RAM vidéo).

Il existe trois méthodes de communication du processeur avec les registres du contrôleur et les tampons de données :

- Dans la première, chaque registre a un numéro de port d'entrée/sorties (entier 1 ou 2 octets), les instructions sont du genre :

- In Registre_Proc n°_Port (pour lire une donnée dans le registre n°_Port et la stocker dans le registre processeur Registre_Proc)
- Out n°_Port Registre_Proc (pour écrire le contenu du registre processeur Registre_proc dans le registre du périphérique n°_Port)

- Dans la seconde méthode, appelée entrées/sorties projetées/mappées en mémoire, on attribue à chaque registre du contrôleur une adresse dans l'espace mémoire (cet espace étant exclusivement réservé à ce registre). Les adresses assignées se trouvent en général en haut de l'espace d'adressage. L'avantage est que les instructions utilisent un seul type d'adressage et que la protection consiste juste à ne pas allouer à un programme utilisateur l'espace mémoire réservées aux registres du contrôleur.

- Une troisième méthode consiste en un système mixte : la partie du tampon de données mappées en mémoire, en plus des ports d'entrées/sorties (c'est le cas du Pentium).

4) Accès direct à la mémoire

Quelque soit la méthode, le processeur peut ensuite demander les données au contrôleur octet par octet, mais cela prend beaucoup de temps processeur. Une autre méthode est l'accès direct à la mémoire (DMA : Direct Memory Acces) qui ne peut se faire que si le matériel est équipé d'un contrôleur DMA. Le contrôleur DMA possède une zone tampon et plusieurs registres, dont un d'adressage mémoire, un de décompte d'octets, plusieurs de contrôle, avec le port d'entrée/sortie, la direction du transfert (lecture/écriture), l'unité de transfert (un ou plusieurs octets à la fois). Le transfert classique des données du tampon à la mémoire principale se fait octets par octets, alors que dans le DMA, le transfert se fait d'abord vers le tampon du contrôleur DMA, ensuite le DMA travaille de façon autonome, libérant ainsi le processeur.

II - L'aspect logicielle

1) Les grands principes

Le premier principe est celui d'indépendance par rapport au matériel, le programme doit accéder à n'importe quel périphérique d'entrée/sorties sans que l'on ait à préciser le type de périphérique.

Le second principe est celui de désignation universelle, le nom d'un fichier ou d'un périphérique doit être une chaîne simple et ne pas dépendre du type de périphérique.

Le troisième principe concerne la gestion des erreurs, elle doit être aussi près du matériel que possible. Si le contrôleur détecte une erreur de lecture, il essaie d'abord de la corriger lui-même, s'il ne peut pas, c'est le pilote qui la gère (en redemandant la lecture par exemple dans le cas d'une poussière...), si le pilote échoue, il en informe la couche logicielle supérieure qui gère alors l'interruption.

On distingue les transferts synchrones, bloquant le processeur, et les transferts asynchrones qui libèrent le processeur dès que le transfert est commencé. En général, un programme utilisateur doit être suspendu jusqu'à ce que sa requête d'entrée/sortie soit réalisée, c'est le système d'exploitation qui assure que les opérations engendrées par les interruptions paraissent bloquantes pour les programmes utilisateurs.

Un autre principe des logiciels d'entrées/sorties est l'usage d'une mémoire tampon pour stocker les données, pour examiner de façon globale le transfert (pour décider où le placer par exemple), pour des contraintes de temps réel (son)...

Enfin, on distingue les périphériques partageables (disque), et les périphériques dédiés ou non partageables (imprimante), le système d'exploitation doit pouvoir gérer les périphériques non partageables sans interblocage.

2) Les entrées/sorties programmées

Il existe plusieurs façons de construire des entrées/sorties. La plus simple consiste à laisser le processeur traiter toute l'opération, c'est ce qu'on appelle les entrées/sorties programmées. Le processeur gère toutes les étapes du transfert et dialogue continuellement avec le contrôleur de périphérique pour vérifier s'il est en état de continuer. Le processeur n'est libéré que lorsque l'entrée/sortie est terminée (cela ne correspond pas nécessairement à l'exécution complète, l'action du processeur se limitant souvent à copier les données dans un tampon interne).

3) Les entrées/sorties pilotées par les interruptions

Pour ne pas monopoliser le processeur, on fait appel aux procédures d'interruptions. Si le processeur doit imprimer un caractère et que l'imprimante n'est pas libre, une interruption suspend le processus d'impression, fait appel à l'ordonnanceur pour charger un autre processus, lorsque l'imprimante est libre, elle envoie un signal qui génère une interruption, si le processus d'impression est prioritaire par rapport au processus en cours, celui-ci est suspendu pour reprendre l'impression.

4) Les entrées/sorties avec DMA

Les procédures d'interruption prennent beaucoup de temps, une alternative aux entrées/sorties gérées par les interruptions est l'utilisation de la DMA. Lors d'une impression par exemple, au lieu d'avoir une interruption par caractère, on aura une interruption par tampon imprimé.

5) La structure en couches

En général, on organise les logiciels d'entrées/sorties suivant quatre couches :

partie logicielle d'entrée/sortie au niveau utilisateur
partie logicielle du S.E. indépendante du périphérique
pilotes du périphérique
matériel

Dans la partie logicielle du système d'exploitation indépendante des périphériques, on a les fonctionnalités suivantes : interfaçage uniforme avec les pilotes, mise en mémoire tampon de données, rapport d'erreurs (erreurs de programmation comme écrire sur un disque inexistant, erreurs qui n'ont pu être résolues par le pilote, ...), allocation et libération des périphériques dédiés (i.e. non partageables, en gérant une file d'attente par exemple...), fourniture d'une taille de bloc indépendante du périphérique.

Dans la partie logicielle d'entrée/sortie au niveau de l'utilisateur, on a les fonctionnalités suivantes : émission des appels d'entrées/sorties, formatage des entrées/sorties, utilisation du spool. Le système de spoule (spool) sert à la gestion des périphériques non partageables comme l'imprimante. Pour éviter qu'un processus ne bloque le périphérique, on crée un répertoire spécial, appelé répertoire de spoule, et un processus spécial, appelé démon (daemon), qui sera le seul à pouvoir écrire les fichiers dans ce répertoire.

III - Quelques périphériques

1) Les disques

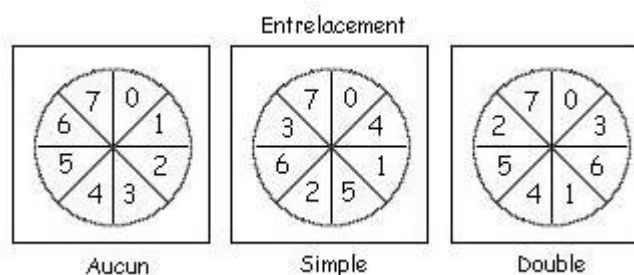
Les disques magnétiques (disque dur, disquette) sont organisés en cylindres, pistes, secteurs. La vitesse de rotation du lecteur (vitesse angulaire) est constante. Sur les anciens disques le nombre de secteurs par piste est le même quelque soit le cylindre. Sur les disques récents, il y a plus de secteurs sur les pistes externes que sur les pistes internes. Certains logiciels fonctionnent comme s'il y avait le même nombre de secteurs quelque soit la piste avec des numéros virtuels de cylindres, têtes, secteurs, et le contrôleur remappe ensuite sur les numéros réels (dans ce cas, la géométrie physique du disque ne correspond pas à sa géométrie virtuelle). Certains contrôleurs de disques peuvent effectuer des recherches sur plusieurs disques simultanément, cependant un seul transfert entre le contrôleur et la mémoire est possible. La technologie RAID (Redundant Array Inexpensive Disk) consiste à recopier les données de façon redondante sur plusieurs disques pour accélérer les temps de lecture et augmenter la sécurité (secteurs défectueux).

Les disques optiques (CD, DVD) sont des disques de polycarbonate contenant des creux et des méplats (la transition creux/méplat ou méplat/creux symbolisant 1, son absence symbolisant 0) éclairés par une diode laser et lus par un détecteur photosensible. La lecture s'effectue à une vitesse uniforme donc la vitesse angulaire diminue lorsqu'on se rapproche du centre.

La première opération à effectuer est le formatage, il consiste à organiser les secteurs (souvent 512 octets) de chaque piste en trois parties :

- Le préambule qui contient une séquence particulière de bit identifiant le début du secteur, le numéro du cylindre, le numéro du secteur.
- Bloc réservé aux données
- Le champs ECC (Error Correction Code) qui contient des informations servant à la récupération en cas d'erreur.

La numérotation des secteurs peut être linéaire ou entrelacée :



Lorsque plusieurs requêtes d'accès au disque se présentent, il peut être plus rapide de ne pas les traiter dans l'ordre. Il existe plusieurs algorithmes de déplacement du bras du disque :

- FCFS (First Come First Served) prend dans l'ordre d'arrivée
- SSF (Shortest Seelc (ou Seelc, voir avec l'écrivain...) First) ou SSTF (Shortest Seelc Titre First), on prend le plus proche à partir de la position où l'on est
- L'algorithme de l'ascenseur ou SCAN balayage, le logiciel conserve un bit de direction lorsqu'une requête se termine, ensuite il poursuit sa requête dans cette même direction. Il modifie alors le bit de direction et reprend les recherches dans l'autre sens.

On suppose par exemple que le bras est positionné (en lecture) sur le cylindre 12, que les requêtes 4, 13, 7, 32, 10 arrivent, et qu'avant d'être sur le cylindre 12, le bras était sur le cylindre 9.

Avec FCFS, on aura 4, 13, 7, 32, 10

Avec SSF, on aura 13, 10, 7, 4, 32

Avec l'algorithme de l'ascenseur, on aura 13, 32, 10, 7, 4

Ces algorithmes ne sont évidemment valables que si la géométrie réelle du disque est la même que la géométrie virtuelle.

N.B. : On appelle temps de positionnement du bras, le temps nécessaire pour que le bras portant la tête de lecture vienne sur la bonne piste, on appelle temps de latence, le temps pour qu'une fois placée sur la bonne piste, la tête de lecture soit sur le bon secteur.

2) Horloge

Les horloges sont constituées d'un oscillateur à quartz qui génère des signaux périodiques, d'un compteur et d'un registre de maintien. Elles sont le plus souvent programmables et possèdent deux modes de fonctionnement.

En mode non répétitif (one shot), l'horloge copie la valeur du registre de maintien dans le compteur qui se décrémente à chaque impulsion. Lorsque le compteur est arrivé à zéro, il engendre une interruption et s'arrête, jusqu'à ce qu'il soit de nouveau démarré.

En mode répétitif (square wave), lorsque le compteur est arrivé à zéro et a engendré une interruption, le registre de maintien est de nouveau copié dans le compteur, et le même enchaînement se répète, les interruptions périodiques sont appelées tops d'horloge.

Les fonctions du pilote d'horloge peuvent varier selon le système d'exploitation, elles se résument en :

- Maintenir l'heure
- Empêcher les processus de s'exécuter plus longtemps que le temps alloué
- Rendre compte de l'utilisation au processeur
- Gérer l'appel système qui demande l'heure
- Fournir des horloges pour certaines parties du système
- Garder certaines données de durée pour établir des statistiques, profiler le système

3) Clavier

Chaque action sur une touche provoque une interruption, le numéro de code de la touche est transmis au pilote du clavier via un port d'entrée/sortie, le numéro sera ensuite traduit selon la configuration du clavier.

4) Écran

Il existe deux types d'écrans : les écrans vectoriels, auxquels on peut transmettre des commandes telles que dessiner une ligne, une figure géométrique, et les écrans matriciels, qui ont un affichage point par point, appelés pixels (1024x768 par ex.).

Les écrans matriciels sont gérés par la carte vidéo, qui contient une RAM vidéo où est stockée l'image de l'écran. En mode texte, chaque octet ou couple d'octets de la RAM vidéo correspond à un caractère à afficher. En mode bitmap, chaque pixel de l'écran est représenté dans la RAM vidéo avec un bit pour l'affichage d'un pixel noir/blanc, ou plusieurs bits (24 par exemple) pour l'affichage d'un pixel en couleur. La carte vidéo contient également le contrôleur vidéo qui lit les données, la RAM vidéo et génère les signaux vidéo adéquats.

5) Interface graphique

Avant la plupart des PC fonctionnaient avec une interface textuelle MS-DOS. La plupart des PC actuels utilisent une interface graphique ou GUI (Graphical User Interface).

Une GUI contient quatre éléments : fenêtres, icônes, menus, pointage, on les note aussi WINP (Windows, Icons, Menus, Pointing). La partie logicielle de la GUI peut être implémentée au niveau du code utilisateur (comme dans UNIX) ou au niveau système d'exploitation (comme dans Windows).

6) Terminaux réseau

Les terminaux réseau permettent à un utilisateur de se connecter par le biais d'un terminal à un ordinateur distant appelé alors serveur. Il y a plusieurs conceptions de terminaux. Un terminal minimaliste peut se contenter d'afficher les pixels à l'écran et de transmettre les actions effectuées par l'utilisateur (enfoncement de touches, déplacement de souris). Un terminal intelligent contiendra un processeur puissant, beaucoup de mémoire et travaillera activement avec le serveur.

7) Alimentation

Une étude faite sur la consommation électrique des différentes parties d'un ordinateur portable donne les résultats suivants :

- écran : entre 39 % et 68 %
- processeur : entre 12 % et 18 %
- disque dur : entre 12 % et 20 %
- périphérique sonore : $\simeq 2\%$
- mémoire : entre 0.5 % et 1 %

Le système d'exploitation peut intervenir dans la gestion de la consommation électrique, en se mettant en veille, en hibernation (veille prolongée) ou en réduisant la définition de l'écran par exemple (fonctionnement en mode dégradé).