

**GIF-1001 Ordinateurs: Structure et Applications**  
**Solutions : Les entrées-sorties**

---

1. Donnez les avantages et les inconvénients de chaque méthode d'E/S présentée dans le cours (E/S programmée, E/S avec interruptions et DMA)?

**Solution:** E/S programmée : Lent, très exigeant pour le CPU, très simple  
E/S avec interruptions : Assez rapide, requiert des interventions ponctuelles du CPU, assez simple  
DMA : Rapide, requiert du matériel additionnel, requiert peu de temps de CPU

2. Pourquoi un module d'E/S a-t-il besoin de registres de donnée ou d'une structure similaire (tampon pour mémoriser des données provenant du CPU ou allant vers le CPU)?

**Solution:** Parce le CPU et les périphériques ne communique pas à la même vitesse.

3. Décrivez le fonctionnement du DMA.

**Solution:** Le CPU demande un transfert de donnée d'un périphérique à la mémoire (ou vice-versa) au contrôleur de DMA. Ensuite, le contrôleur de DMA demande le contrôle des bus au CPU. Ce dernier accorde le contrôle et se déconnecte des bus (haute impédance). Ensuite, le contrôleur de DMA fait le transfert de données en contrôlant les bus (le bus d'adresse est contrôlé par un compteur). Après le transfert de données, le contrôle des bus est redonné au CPU.

4. Quel est le taux de bit maximum pouvant être transmis sur un bus synchrone ayant 12 lignes de contrôle, 13 lignes d'adresses et 14 lignes de données, une horloge de 10MHz, une méthode d'arbitrage décentralisée et supportant le DMA?

**Solution:**  $14 \text{ bits} * 10\text{MHz} = 140\text{Mbits/s}$

5. Quelles sont les principales fonctions des modules d'entrées-sorties? En d'autres mots, à quoi sert le matériel entre le bus et le périphérique?

**Solution:** Il sert d'abord d'interface entre les bus de données, d'adresse et de contrôle. Avec des tampons de données, il ajuste la vitesse entre le bus et le périphérique. Le matériel gère et contrôle le périphérique.

6. Pourquoi les interruptions des périphériques sont-elles gérées par le système d'exploitation?

**Solution:** Il y a plusieurs raisons. Par exemple:

- Le système d'exploitation permet au programmeur de concevoir ses programmes sans se soucier de la nature des périphériques.
- Le système d'exploitation gère le temps de l'ordinateur. Il est naturel qu'il gère les interruptions matérielles.
- Quand un périphérique termine une opération, le système d'exploitation le sait. Il peut ajuster les états des processus en conséquence.