

Questions / réponses

Raed Abdennadher et Steven Liatti, groupe 7

1. Dans quel ordre avez-vous initialisé les différents points ci-dessus dans votre kernel ? Justifiez.

L'initialisation de l'écran, de la gdt, de l'idt et du pic sont interchangeables, car ils ne dépendent pas l'un de l'autre. En revanche, il faut initialiser le timer après idt pour utiliser le handler. Finalement on finit par démasquer les interruptions matérielles.

2. Pourquoi remappe-t-on les IRQ 0 à 7 aux interruptions 32 à 39 ? Que se passerait-il si on ne le faisait pas ?

Il y aurait un conflit entre les exceptions CPU et les interruptions.

3. Comment pouvez-vous vérifier que votre gestionnaire d'interruptions pour les exceptions fonctionne correctement ?

Faire une division par 0, accéder à une zone mémoire non appropriée.

4. Quelle taille de buffer clavier avez-vous choisie et pour quoi ?

Une taille de 20. Nous sommes partis du principe qu'il fallait une taille de buffer adaptée aux doigts humains. Pourquoi 20 ? Dans le cas où on jouerait à un jeu à deux joueurs et qu'ils appuyeraient simultanément sur le clavier avec tous leurs doigts. Pour un usage courant, une taille de 10 est largement suffisante.

5. Comment pouvez-vous causer une situation de buffer plein, même pour un buffer de grande capacité ?

En faisant un très long sleep et en tapant beaucoup de texte au clavier.

6. Est-ce que votre fonction sleep possède une limitation ? Si oui, laquelle et pour quoi ?

Oui, comme ticks est une valeur sur 32 bits, la valeur maximale de ticks = $4'294'967'296$. À une fréquence de 1000 Hz, on pourra attendre 4'294'967 secondes au maximum (environ 50 jours). Imaginons que le système est allumé depuis 49 jours et qu'une instruction de code demande de faire un sleep de 5 jours, l'attente ne sera pas effective car le timer ne pourra "compter" encore que pour 1 jour.