

Tom Pillot

Rapport : Développement d'un mini système d'exploitation pour PC x86

Console

L'affichage des caractères de contrôle fonctionne. On peut changer la couleur d'écriture ou de fond avec les variables `terminal_color_fg` et `terminal_color_bg`. Pour connaître l'emplacement du curseur, j'utilise des variables globales qui contiennent la ligne et la colonne du curseur. Le scroll fonctionne aussi.

Interruptions

L'interruption 50 et celle du timer fonctionnent.

Timer

Je conserve le temps dans une variable globale. Pour l'affichage, je le fais que lorsque c'est vraiment nécessaire, c'est-à-dire lors d'un changement du passage à la seconde suivante ou lors d'un scroll. L'affichage du timer en haut à droite de la console fonctionne.

Appels système

J'ai implémenté les appels système `shutdown` et `write` qui marchent correctement.

Processus

Pour les processus, j'ai implémenté un système collaboratif, ce sont les processus eux-mêmes qui rendent la main en appelant `schedule()`. L'ordonnancement est de type Round-robin : on parcourt la liste des processus à partir du processus actif pour trouver un processus prêt, puis on fait un changement de contexte vers ce nouveau processus.

Pour tester les processus, j'ajoute 5 processus simples (ils incrémentent un compteur et l'affichent) à ma liste des processus et je lance le premier. On peut voir au lancement que l'ordonnancement et le changement de contexte fonctionnent car on boucle bien entre les 5 processus et le contexte est effectivement sauvegardé car les compteurs sont bien incrémentés.

Remarques

J'ai essayé de bien commenter mon code si vous voulez plus de détails. Aussi j'ai enlevé les `malloc` que j'utilisais pour être sûr que ça marche avec votre version de qemu.