

# Bootloader et hal\_cpu : abstraction du processeur par le noyau

1. [1. Objectifs](#)
2. [2. Séance 1](#)
3. [3. Séance 2](#)

## Objectifs

Le but de ces séances est de comprendre comment le noyau voit le processeur.

## Séance 1

## Séance 2

La gestion des contextes et l'entrée dans le noyau ont été ajoutées. Vous allez expérimenter les deux.

- Répondez aux questions présentes dans les fichiers hal\_cpu\_code.h et hal\_cpu\_code.c.
- Nous n'avons pas encore de threads, mais nous avons des contextes qui permettent d'exécuter plusieurs "tâches" en pseudo parallélisme sur un seul core.
  - ◆ Créez 2 contextes sur le processeur 0
  - ◆ Chaque contexte utilisera un tty différent (le 1 et le 2) pour faire un écho (ce qui est taper au clavier s'écrit à l'écran). Puisqu'on n'a pas encore les drivers vous utiliserez les tty directement comme vous le faites pour le tty0.
- Pour expérimenter les interruptions, vous allez autoriser les interruptions issues du tty3 et faire un echo. L'ISR du tty3 sera directement la fonction do\_interrupt (on n'a pas encore les pilotes)