Problématiques Software

# Différences entre uClinux et le Linux “standard”

Il existe un certain nombre de différences, autant au niveau du noyau que au niveau du système entier.

## Noyau Linux vs uClinux

La différence principale entre les deux noyaux est que le noyau Linux standard nécessite un processeur avec une Memory Managment Unit(MMU). La MMU permet de faire en sorte qu’une adresse donnée soit mise en correspondance avec un emplacement physique selectionnable. Dans un OS, ça permet entre autre de faire en sorte que chaque processus voit son propre espace d’adressage, avec chacun son adresse 0.

Les cortex-m ne possèdent pas de MMU. Ils ont seulement une MPU qui permet d’interdire l’accès à certaines zone.

Le noyau uClinux ne nécessite pas de MMU ni de MPU d’ailleurs. Certaines fonctionnalités sont abscentes (pas possible de forker un processus comme sous linux, pas de swap).

Par ailleurs, comme ce sont des changement invasifs, ils sont difficiles à maintenir, ce qui explique que le noyau uClinux soit basé sur un noyau Linux 2.6 et que les noyaux 3.X n’aient pas été adaptés.

## Système basé sur uClinux

Un système de fichier lecture seule a été fait pour uClinux: romfs. Il est simple, et garantie que les fichiers sont alloués sans “trous” (le fichier est en un seul morceau, il n’est pas fragmenté). C’est indispensable pour le eXecute In Place(XIP), vu que le fichier est mappé en mémoire (les octets du fichier sont accessibles directement comme la RAM). En fait, avec le FMC, toute les mémoires raccordées dessus sont mappées dans l’espace d’adressage du processeur, de la même manière que la RAM ou les registres. Les Cortex-M sont des architectures de Von Neumann dans le sens où un unique espace d’adressage est disponible (code, données, registres).

Un format de binaire a aussi été fait spécialement pour uClinux: le BFLT. Le format normal sous Linux est l’ELF. Le BLFT est beaucoup plus basique, probablement pour prendre moins de place. Mais à priori, on peut quand même utiliser l’ELF.

# Nécessités hardware pour supporter uClinux

Peu de choses sont indispensables:

* Une UART
* De la mémoire permanente (visiblement, il faut compter 64Mo pour être totalement tranquille, même si 32Mo peuvent suffire)
* De la RAM externe, vu que la RAM interne n’est pas assez grande
* Un moyen de debug externe, car ça marchera pas du premier coup