Bootloader, Linux kernel, Root filesystem

Purpose

Laern how to built Linux with Busybox and install it on a embedded device.

Steps

1. `git clone git://git.busybox.net/buildroot`
2. `cd buildroot/`
3. `make raspberrypi3_defconfig`
4. `sudo dnf install perl-ExtUtils-MakeMaker`
5. `make source`
6. `make menuconfig`
7. Filesystem images [Enter]
8. Scroll down to "tar the root filesystem" [y]
9. Compression method() [Enter]
10. Choose gzip or bzip2
11. Save and exit
12. make -j 7
13. `cd output/images`
14. `df -a` to find the SD card
15. `sudo dd if=sdcard.img of=/dev/mmcblk0`

Usage of files in `output/images/rpi-firmware`

1. `bootcode.bin`: 第二階段的 bootloader, 啟用 SDRAM
2. `cmdline.txt`: 開機時 Linux kernel 所用到的參數
3. `config.txt`: 剛開機時所用的設定檔，取代傳統 BIOS
4. `fixup.dat`: 處理 SDRAM 的記憶體分配
5. `start.elf`: GPU 的韌體，會讀取config.stxt, cmdline.txt, kernel.img，並讓 GPU 有能力把 CPU 啟動

What’s inside `output/images/rpi-firmware/cmdline.txt`, and what does those arguements mean?

- root=/dev/mmcblk0p2: `/` 掛載位置
- rootwait: 等待 root 掛載完成再進一步執行
- console=tty1 console=ttyAMA0,115200: 設定 console