Umeå University
Department of Computing Science

Operating Systems 7.5 p
5DV171

Assignment 1

Submitted	2017-09-26
Author:	Lorenz Gerber (`dv15lgr@cs.umu.se` `lozger03@student.umu.se`)
Instructor:	Jan Erik Moström / Adam Dahlgren Lindström

1 Introduction

2 Step by Step Description

2.1 Preparing Raspberry Pi

First a recent version of raspbian was installed on the SD card. Here we used 'Raspian Stretch Lite', Version September 2017, release 2017-09-07 with initial Kernel version 4.9.

2.2 Setting up Toolchain

Here it was decided to cross-compile the kernel from a Ubunut Linux machine. Hence the toolchain had to be set up for cross compiling. The toolchain was obtained from `git clone https://github.com/raspberrypi/tools`. In the instructions (`https://www.raspberrypi.org` it was advised to copy the toolchain to a common location and add the directory `/tools/arm-bcm2708/gcc-linar` to the `.bashrc` path.

2.3 Getting the source

The Linux kernal sources for Raspberry Pi Raspian were obtained from `git clone --depth=1 https://git`

2.4 Test Building the Kernel

```
cd linux
KERNEL=kernel7
make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- bcm2709_defconfig
make -j 4 -ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- zImage modules dtbs
```

Instead of setting the `.bashrc` path, it is also possible to indicate the full path for `arm-linux-gnueabihf`. In this case, this was `/usr/local/tools/arm-bcm2708/gcc-linaro-arm-linux-gnu` The `-j 4` flag chooses the number of processers (times 1.5) to use.

Now the SD card of the Raspberry is plugged into the Linux machine. Within the Linux Kernel repo base directory, two mountpoints for the SD card partitions are created (`mkdir -p mnt/fat32` and `mkdir -p mnt/ext4`). The SD card is mounted (`sudo mount /dev/sdb1 mnt/fat3` and `sudo mount /dev/sdb2/ mnt/ext4`. Then standing at the kernel repo base directory, the modules are installed by: `sudo make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- INSTALL_MOD_P` Again, the full path to `arm-linux-gnueabihf-` was needed. Then the kernel and device tree blobs were copied onto the SD card, also backing up the old kernel:

```
sudo cp mnt/fat32/$KERNEL.img mnt/fat32/$KERNEL-backup.img
sudo cp arch/arm/boot/zImage mnt/fat32/$KERNEL.img
sudo cp arch/arm/boot/dts/*.dtb mnt/fat32/
sudo cp arch/arm/boot/dts/overlays/*.dtb* mnt/fat32/overlays/
sudo cp arch/arm/boot/dts/overlays/README mnt/fat32/overlays/
```

Then the SD card can be umounted and booted in the Raspberry Pi.