ECE 4730: Embedded Systems II

Project 4: Linux Compilation

Objectives

---

* Create and utilize a Makefile

* Create and run a kernel module

* Use virtual addresses to access physical addresses/locations.

Deliverables

---

*	Multiply.ko	6
*	Ironman.ko	9
*	Answers to the questions	10
Total:		25

Tips

Save a copy of all files on the SD card in an easily-accessible place on your computer. If there are mounting problems, format it and replace the files.

Procedure

Part 1- Exploration

1. Boot Linux on the Zybo Z7-10 and connect to it on PuTTY.

2. Once inside the Zybo terminal, run the following command:
mount /dev/mmcblk0p1 /mnt/

The above command mounts mmcblk0p1, the SD card, attaching it to /mnt

3. Change to the newly-mounted device (/mnt/mmcblkop1) and test its write capabilities.

4. Change back to / directory, then run the following command:
mount /dev/mmcblk0p1 /mnt/
The above command unmounts the device in /mnt. ALWAYS do this before ejecting the SD   card or powering off.

5. Reset the processor using the SRST button. Then mount the SD card and ls its files. Did the files survive reboot?

Part 2- Your First Kernel Module (hello.ko)

1. Create a new folder (lab5modules) on the server and copy hello.c, multiply.c, and Makefile (from lab5files) into it. (make sure there is a tab in all indented spaces)

2. There may be something interesting about sourcing /brockxilinx/settings64.sh, just a note for future bugs.

3. Type the following command while in the newly-created directory.
        Make ARCH=arm CROSS_COMPILE=arm-none-eabi-
    'hello.ko' should appear in that directory upon completion.

4. Place 'hello.ko' on your SD card.

5. Eject it, then place the SD card in the Zybo and mount it.

6. While in the SD card's directory (the same directory as hello.ko), run the following command:
        Make ARCH=arm CROSS_COMPILE=arm-none-eabi-
    'hello.ko' should appear in that directory upon completion.

7. Run 'dmesg | tail' to see the kernel's loading output. Screenshot the output and attach it to your report.

8. Run 'lsmod' and verify that 'hello' is listed.

9. Run the following commands to remove the module:
        /lib/modules/`uname -r`   The `'s are not the apostrophe. They're the char above tab
        rmmod hello   Removes the module.
        lsmod   Verifies module removal.
    Uname returns name and version of current kernel. It is necessary for safe module dismantling.

Part 3- Your turn (multiply.ko, memory mapping)

1. Create a new folder to create your new multiply module (multiply), and place multiply.c and Makefile in it.

2. Verify that the path to the Linux kernel directory (in the newly copied Makefile) remains unchanged, and modify it to compile multiply.c.

3. In your sdk folder (Xilinx, project 4), find 'xparameters.h' and 'xparameters_ps.h' and copy them to your newly-made multiply module directory.

4. Edit the code in multiply.c to compile and run correctly. Consult [Linux Device Drivers, 3rd Edition](#) for tips.

5. Once compiled, run the kernel module 'multiply.ko' on the Zybo screenshot the output 'win+shift+s' and place it in the report.

6. What is the mount point for the SD card on your machine?

7. If we changed the name of 'hello.c' to 'goodbye.c', what would change in the Makefile?