Laboratory Exercise #4

Linux boot-up on ZYBO Z7-10 board via SD Card

Name: Daniel Horan
UIN: 527005307

ECEN 449-504

I. Introduction

In this lab, we will integrate Linux on the ZYBO Z7-10 board using Vivado and the Zynq Processing System. Our primary objective is to design a microprocessor system optimized for Linux. Subsequently, we'll utilize PetaLinux tools to compile the Linux kernel for this custom system. This exercise deepens our understanding of embedded OS integration on FPGA platforms and emphasizes the critical relationship between OS-driven software and dedicated hardware.

II. Procedure

In the initial phase, a Vivado project was initiated for the ZYBO Z7-10 board, focusing on constructing a Zynq (ARM Cortex A9) based microprocessor system optimized for a Linux environment. Key components, including the SD Card, DDR3 controller, and timer, were integrated to enhance system functionality.

Post hardware configuration, the process shifted to the PetaLinux environment. Within this context, the primary task was to compile the Linux kernel, ensuring alignment with the bespoke microprocessor system design. This necessitated the generation of Zynq's boot image along with vital Linux boot files.

Upon successful compilation, the Linux boot was executed on the ZYBO Z7-10 board via an SD card. Throughout, the 'picocom' serial console application provided a lens for real-time output analysis.

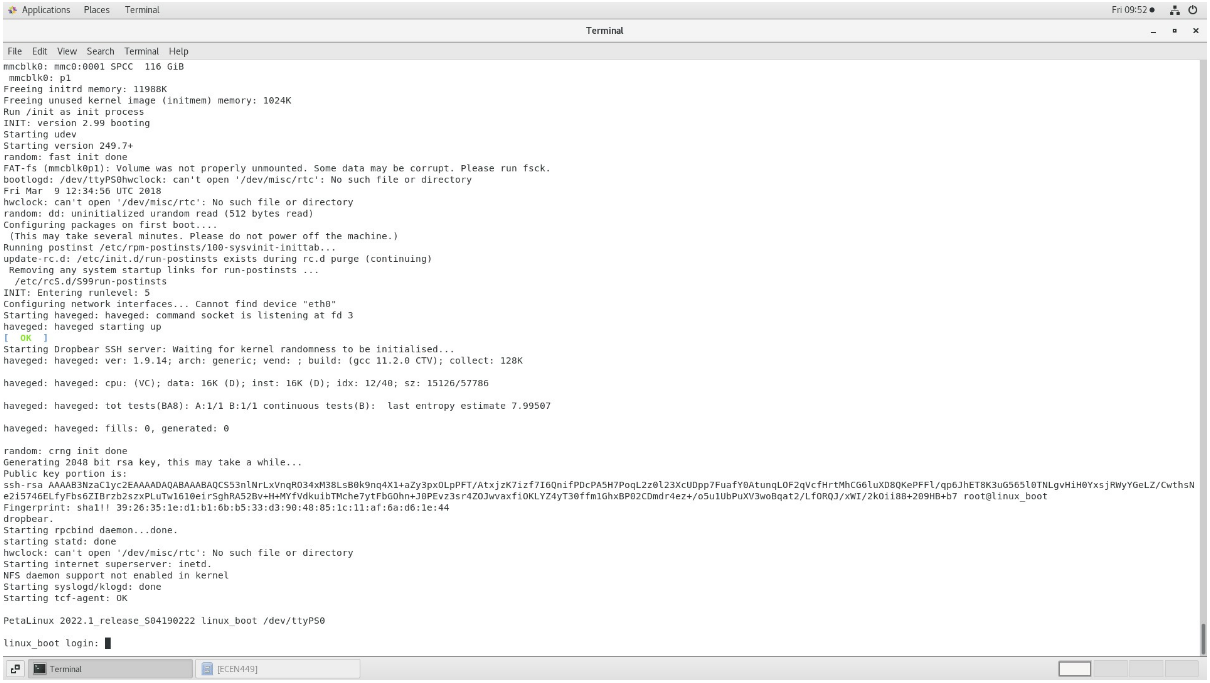
The procedure culminated with a demonstration of Linux OS functioning seamlessly on the tailored hardware platform. Throughout the procedure, strategic hints were provided to navigate potential challenges and enhance system efficiency.

III. Results

In this laboratory session, the central objective was the configuration and optimization of a Zynq-based microprocessor system for a Linux environment utilizing Vivado. This task demanded rigorous integration of essential components, culminating in the execution of the Linux boot on the ZYBO Z7-10 board.

Throughout the implementation phase, the majority of the procedures were executed without significant setbacks. However, nuances associated with the USB drive formatting presented minor technical challenges. These were swiftly addressed, ensuring minimal disturbance to the project's progression.

Following the completion of the setup, a comprehensive demonstration was conducted for the TA, underscoring the effective operation of the Linux OS within the tailored hardware framework, thereby confirming the successful attainment of the prescribed laboratory aims.



```
mmcblk0: mmc0:0001 SPCC  116 GiB
 mmcblk0: p1
Freeing initrd memory: 11988K
Freeing unused kernel image (initmem) memory: 1024K
Run /init as init process
INIT: version 2.99 booting
Starting udev
Starting version 249.7+
random: fast init done
FAT-fs (mmcblk0p1): Volume was not properly unmounted. Some data may be corrupt. Please run fsck.
bootlogd: /dev/ttyPS0hwclock: can't open '/dev/misc/rtc': No such file or directory
Fri Mar  9 12:34:56 UTC 2018
hwclock: can't open '/dev/misc/rtc': No such file or directory
random: dd: uninitialized urandom read (512 bytes read)
Configuring packages on first boot....
 (This may take several minutes. Please do not power off the machine.)
Running postinst /etc/rpm-postinsts/100-sysvinit-inittab...
update-rc.d: /etc/init.d/run-postinsts exists during rc.d purge (continuing)
 Removing any system startup links for run-postinsts ...
  /etc/rcS.d/S99run-postinsts
INIT: Entering runlevel: 5
Configuring network interfaces... Cannot find device "eth0"
Starting haveged: haveged: command socket is listening at fd 3
haveged: haveged starting up
[  OK  ]
Starting Dropbear SSH server: Waiting for kernel randomness to be initialised...
haveged: haveged: ver: 1.9.14; arch: generic; vend: ; build: (gcc 11.2.0 CTV); collect: 128K

haveged: haveged: cpu: (VC); data: 16K (D); inst: 16K (D); idx: 12/40; sz: 15126/57786

haveged: haveged: tot tests(BA8): A:1/1 B:1/1 continuous tests(B):  last entropy estimate 7.99507

haveged: haveged: fills: 0, generated: 0

random: crng init done
Generating 2048 bit rsa key, this may take a while...
Public key portion is:
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCS53nlNrLxVnqRO34xM38LsB0k9nq4X1+aZy3pxOLpPFT/AtxjzK7izf7I6QnifPDcPA5H7PoqL2z0l23XcUDpp7FuafY0AtunqLOF2qVcfHrtMhCG6luXD8QKePFFl/qp6JhET8K3uG565l0TNLgvHiH0YxsjRWyYGeLZ/CwthsN
e2i5746ELfyFbs6ZIBrzb2szxPLuTw1610eirSghRA52Bv+H+MYfVdkuibTMche7ytFbGOhn+J0PEvz3sr4ZOJwvaxfiOKLYZ4yT30ffm1GhxBP02CDmdr4ez+/o5u1UbPuXV3woBqat2/LfORQJ/xWI/2kOii88+209HB+b7 root@linux_boot
Fingerprint: sha1!! 39:26:35:1e:d1:b1:6b:b5:33:d3:90:48:85:1c:11:af:6a:d6:1e:44
dropbear.
Starting rpcbind daemon...done.
starting statd: done
hwclock: can't open '/dev/misc/rtc': No such file or directory
Starting internet superserver: inetd.
NFS daemon support not enabled in kernel
Starting syslogd/klogd: done
Starting tcf-agent: OK

PetaLinux 2022.1_release_S04190222 linux_boot /dev/ttyPS0

linux_boot login:
```

IV. Conclusion

During this lab, we utilized Vivado to establish a Linux environment on the ZYBO Z7-10 board. The majority of the process was executed seamlessly, with a minor challenge encountered during the USB drive formatting phase. Addressing this issue highlighted the importance of meticulous attention to detail in engineering tasks. After resolving the concern, the environment was successfully demonstrated to the TA, confirming the lab's successful completion. This endeavor further emphasized the value of practical experience in understanding theoretical concepts, and the necessity of precision in both overarching strategies and granular implementations.

V. Questions

a) The function of local memory is to provide faster, temporary storage closer to the processing unit for frequently accessed data, improving processing speed. Yes, this "local memory" exists on a standard motherboard and is known as CPU cache. It's typically found directly on the CPU chip, with levels labeled as L1, L2, and sometimes L3 caches. The L1 cache is closest to the CPU core, ensuring the quickest access times.

b) Directories with 'w' in permissions are writable. Use "ls -l" to check and touch <filename> to test. If a file is created in volatile memory like RAM, it's lost upon rebooting the ZYBO Z7-10 board due to memory not retaining data without power. If saved in non-volatile storage like an SD card, it remains after reboot.

c) If you add another peripheral after compiling the kernel, you'd need to:

    1. Update the hardware design in Vivado.
    2. Export the updated design.
    3. Recompile the kernel to recognize the new peripheral.

   This ensures the kernel manages the new hardware component effectively.