Kyle Loyka
Lab 4 Report

ECEN 449-503

Due: 21 February 2016

Introduction

The purpose of this lab was to build a microprocessor on the FPGA which was capable of running the Linux operating system.

Procedure

In this lab, a MicroBlaze soft-IP microprocessor was created with the XPS software. Using the Base System Builder, Caches, SDRAM, Compact Flash, and interrupt support was added to the MicroBlaze system. The *multiply* module (created in the previous lab) was also added to this system.

After the hardware description was complete, the Linux source was compiled and configured for our MicroBlaze system. Then the *linux_system.ace* file was generated, and copied to a compact flash card. After inserting the CF card into the FPGA, on startup, the MicroBlaze system would boot the Linux kernel.

Results / Q&A

The project functioned as intended.

(A) The 8KB of local memory on the FPGA stores important start up information and firmware. When the FPGA is first turned on, the instructions in this local memory analyze what devices are connected to the board. These instructions then determine from what drive/location to boot from.
On a standard motherboard, this 'local memory' is known as the BIOS (a type of firmware).

(B) The following Linux directories are writeable: *bin, dev, etc, mnt, sbin, tmp, var.*
The following Linux directories are read-only: *proc, sys.*
Note: in the top directory, there is also a file called *init*.
Changes to these directories are not saved between restarts. That is because every time the kernel boots, it loads from nonvolatile memory. The changes in these directories are written to volatile memory, which is 'erased' once the power is turned off.

(C) If another peripheral was added, a new device tree should be generated. Next, the software libraries would have to be regenerated using **Software → Generate Libraries from BSPs**. Then the device bitstream would have to be updated using **Device Configuration → Update Bitstream**. The *system.dts* file found in the Linux source would have to be updated with the newly generated *xilinx.dts* device tree. The Linux code would then have to be reconfigured and recompiled with this new *xilinx.dts* file.

(D) If *USE_BARREL* was set to 0 in the Linux configuration, the system would still function as intended. The purpose of the barrel shifter is to speed up bit level operations. With *USE_BARREL* set to 0, the hardware would support barrel shifting, but the software (Linux) would not take advantage of this support.

Conclusion

This lab provided a strong foundation for compiling and configuring operating systems to run on embedded devices. Notably, we learned how to get a kernel running on an FPGA board.

Kermit Output of Linux Boot

```
early_printk_console is enabled at 0x84000000
Ramdisk addr 0x00000003, Compiled-in FDT at 0xc0165008
Linux version 2.6.35.7 (kyleloyka@lin06-424cvlb.ece.tamu.edu) (gcc
version 4.1.2) #2 Mon Feb 15 10:17:57 CST 2016
setup_cpuinfo: initialising
setup_cpuinfo: No PVR support. Using static CPU info from FDT
cache: wt_msr
setup_memory: max_mapnr: 0x10000
setup_memory: min_low_pfn: 0x90000
setup_memory: max_low_pfn: 0xa0000
On node 0 totalpages: 65536
free_area_init_node: node 0, pgdat c01dc3e4, node_mem_map c02e6000
  Normal zone: 512 pages used for memmap
  Normal zone: 0 pages reserved
  Normal zone: 65024 pages, LIFO batch:15
Built 1 zonelists in Zone order, mobility grouping on.  Total pages:
65024
Kernel command line: console=ttyUL0 root=/dev/ram
PID hash table entries: 1024 (order: 0, 4096 bytes)
Dentry cache hash table entries: 32768 (order: 5, 131072 bytes)
Inode-cache hash table entries: 16384 (order: 4, 65536 bytes)
Memory: 256668k/262144k available
Hierarchical RCU implementation.
        RCU-based detection of stalled CPUs is disabled.
        Verbose stalled-CPUs detection is disabled.
NR_IRQS:32
xlnx,xps-intc-1.00.a #0 at 0xd0000000, num_irq=3, edge=0x2
xlnx,xps-timer-1.00.a #0 at 0xd0004000, irq=0
microblaze_timer_set_mode: shutdown
microblaze_timer_set_mode: periodic
Calibrating delay loop... 36.45 BogoMIPS (lpj=182272)
pid_max: default: 4096 minimum: 301
Mount-cache hash table entries: 512
bio: create slab <bio-0> at 0
Switching to clocksource microblaze_clocksource
Skipping unavailable RESET gpio -2 (reset)
GPIO pin is already allocated
```

```
msgmni has been set to 501
io scheduler noop registered
io scheduler deadline registered
io scheduler cfq registered (default)
Serial: 8250/16550 driver, 4 ports, IRQ sharing disabled
84000000.serial: ttyUL0 at MMIO 0x84000000 (irq = 1) is a uartlite
console [ttyUL0] enabled
brd: module loaded
xsysace 83600000.sysace: Xilinx SystemACE revision 1.0.12
xsysace 83600000.sysace: capacity: 1957536 sectors
 xsa: xsa1
Xilinx SystemACE device driver, major=254
i2c /dev entries driver
Freeing unused kernel memory: 889k freed
Starting rcS...
++ Creating device points
++ Mounting filesystem
++ Loading system loggers
++ Starting telnet daemon
rcS Complete
/bin/sh: can't access tty; job control turned off
/ #
```