Further Experiences Teaching an FPGA-based Embedded Systems Class

Stephen A. Edwards

Columbia University

WESE, Torino, Italy, October 4, 2018
What is an FPGA?

A Field Programmable Gate Array:
A configurable circuit; not a stored-program computer

LUT: 16-element lookup table

Source: http://evergreen.loyola.edu/dhhoe/www/HoeResearchFPGA.htm
Why FPGAs?

Students can explore hardware/software boundary
Easy to put a stored-program computer on an FPGA
Easy to add custom peripherals

<table>
<thead>
<tr>
<th>FPGA</th>
<th>Single-Board Computer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flexible</td>
<td>Cheaper</td>
</tr>
<tr>
<td>Custom peripherals</td>
<td>Stock peripherals</td>
</tr>
<tr>
<td>Commercially uncommon</td>
<td>Mainstream</td>
</tr>
<tr>
<td>Digital logic design w/o soldering</td>
<td>Software-only</td>
</tr>
<tr>
<td>Demands very wide range of skills</td>
<td>More narrow</td>
</tr>
</tbody>
</table>
# Basic Class Structure

<table>
<thead>
<tr>
<th>Lab Type</th>
<th>Weeks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware Lab</td>
<td>1–2</td>
</tr>
<tr>
<td>Software Lab</td>
<td>3–4</td>
</tr>
<tr>
<td>Hardware-Software Lab</td>
<td>5–6</td>
</tr>
<tr>
<td>Project</td>
<td>7–14</td>
</tr>
</tbody>
</table>

Labs come with skeletons; Lab 3 typical project skeleton
2003–2006: XESS XSB-300E (Xilinx Spartan IIE)
2003–2006: XESS XSB-300E (Xilinx Spartan IIE)

Microblaze soft processor

Plenty of peripherals:
VGA, video in, audio I/O, Ethernet, USB, PS/2, SRAM, DRAM, Flash

Pin limit forced bussed peripherals

Hard to use more than one

SRAM usually needed; never enough FPGA RAM

Used VHDL

[Edwards, WESE 2005]
XSB Projects

Raycasting game

Internet video camera

Audio FFT

Video Effects
2007–2013: Terasic DE2 (Altera Cyclone II)
2007–2013: Terasic DE2 (Altera Cyclone II)

Nios II soft processor

Similar peripherals to XSB

Dedicated pins per peripheral

No operating system

VGA easy;
Ethernet hard;
USB impossible

Still used VHDL
DE2 Projects

- Pac-Edwards
- Digital Picture Frame
- Encrypted video
- Real-time ray tracer
2014–2016: Terasic SoCKit (Altera Cyclone V)
2014–2016: Terasic SoCKit (Altera Cyclone V)

2 ARM9 Hard Processor Cores

Linux: TCP/IP, USB

Students write device drivers

Switched to System Verilog

Boot, run diskless (PXE server)

No external SRAM, video input, 7-segment displays

Fragile micro-USB connectors
SoCKit Projects

Inverse Kinematics Accel.

Pottery Game

Auto-parking car

Game of Life
Open Questions

Next generation of boards: DE10-Standard or DE1-SoC or ?

How much “friction” to apply to the students? What should they be given vs. what should they develop themselves?

Should students be allowed to specialize? Is it OK that students take the class and merely sharpen their existing hardware or software skills?

Is the “dynamic range” of skills I require unrealistic?