Using a Commercial MIPS Soft-Core in Computer Architecture Education

Sarah L. Harris, University of Nevada, Las Vegas
Overview

• Introduction
• MIPSfpga Overview
• Labs
• Example Courses and Applications
• Conclusion
Introduction

• What is MIPSfpga?
  – MIPSfpga is an unobfuscated soft-core commercial MIPS processor available from Imagination Technologies for academic use.

• MIPSfpga supporting material is provided in three packages:
  – MIPSfpga Getting Started Guide
  – MIPSfpga Labs
  – MIPSfpga SoC
MIPSfpga Materials

• **MIPSfpga Getting Started Guide**
  – Overall MIPSfpga system, setup, and tools
  – Verilog files for the MIPSfpga core and system

• **MIPSfpga Labs**
  – 25 hands-on labs for experimenting with, analyzing, and modifying the MIPSfpga system

• **MIPSfpga SoC**
  – MIPSfpga as the core of a system-on-chip that runs Linux
MIPSfpga System

- **Soft-core** available in Verilog
- **System modules** available in Verilog and VHDL
MIPSfpga: microAptiv Core

- Commercial microAptiv core
  - 5-stage pipeline
  - Set-associative I & D caches
  - MMU (memory management unit) with TLB
  - Performance counters
  - No DSP, Coprocessor 2, or shadow registers
  - Interfaces:
    - AHB-Lite bus
    - EJTAG programmer/debugger
    - CorExtend for user-defined instructions
How to Run Programs on MIPSfpga

• **In Simulation:**
  – ModelSim
  – XSIM: Vivado’s built-in simulator

• **In Hardware:**
  – Load program into memory:
    • at synthesis
    • using EJTAG interface (using Bus Blaster Probe)
    • using UART interface (i.e., using Nexys4 DDR’s existing programming cable or USB-UART FTDI board)
FPGA Boards

- Support for Nexys4 DDR and DE2-115 boards
- Instructions on how to port it to other FPGA boards (see Lab 1)
System Setup: Nexys4 DDR + BusBlaster
Overview

• Introduction
• MIPSfpga Overview
• Labs
• Example Courses and Applications
• Conclusion
MIPSfpga Labs

• Hands-on learning in areas including:
  – System-on-chip (SoC) design
  – Assembly language and C programming
  – Instruction set architectures (ISAs)
  – Design with hardware description languages (HDLs)
  – Computer architecture
  – Memory systems
  – Memory-mapped I/O and interfacing with peripherals
  – Interrupts
  – Performance Counters

WCAE 2017
MIPSfpga Labs

• Organized into four sections:
  – **Intro** (Labs 1-4)
  – **I/O** (Labs 5-13)
  – **Core** (Labs 14-19)
  – **Memory** (Labs 20-25)
# MIPSfpga Labs

<table>
<thead>
<tr>
<th>Section</th>
<th>Lab #</th>
<th>Lab Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intro</td>
<td>1</td>
<td>Vivado project for MIPSfpga</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>C programming on MIPSfpga</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Assembly programming</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>More programming practice (optional)</td>
</tr>
<tr>
<td>I/O</td>
<td>5</td>
<td>7-segment displays</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Reaction timer</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>Audio</td>
</tr>
<tr>
<td></td>
<td>8/9</td>
<td>SPI light sensor / LCD</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>Interrupts</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>Direct memory access (DMA)</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>DES encryption</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>Performance counters</td>
</tr>
</tbody>
</table>
# MIPSfpga Labs

<table>
<thead>
<tr>
<th>Section</th>
<th>Lab #</th>
<th>Lab Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core</td>
<td>14</td>
<td>Instruction flow: ADD</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>Instruction flow: AND</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>Instruction flow: LW</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>Instruction flow: BEQ</td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>Hazard logic</td>
</tr>
<tr>
<td></td>
<td>19</td>
<td>CorExtend: Adding user-defined instructions</td>
</tr>
<tr>
<td>Memory</td>
<td>20</td>
<td>Basic caching</td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>Cache structure</td>
</tr>
<tr>
<td></td>
<td>22</td>
<td>Cache controller: Hit &amp; miss management</td>
</tr>
<tr>
<td></td>
<td>23</td>
<td>Cache controller: Content management</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>Cache controller: Store and fill buffers</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>Scratchpad RAM</td>
</tr>
</tbody>
</table>
Overview

- Introduction
- MIPSfpga Overview
- Labs
- **Example Courses and Applications**
- Conclusion
Example Courses and Applications

• 4th year Integrated Systems Architecture Course
• 4th year/Master’s level Processor and I/O Systems Lab
• Hackathon

MIPSfpga is best used after learning the basics of Digital Design, HDLs, and Computer Architecture.
Ex. 1: Systems Architecture Course

• At UCM (Universidad Complutense de Madrid) 2017

• Student background
  – Digital & HDL design (VHDL)
  – Computer Organization (MIPS ISA, single/multi-cycle processors, I/O systems)
  – Programming (C++)
Course Contents

• Module 1:
  – Review: MIPS ISA, single/multi-cycle processors, I/O systems

• Module 2:
  – Pipelined processors

• Module 3:
  – Caches

• Module 4:
  – SoC and embedded system design
Course Materials

• **Textbook:**

• **Slides:**
  – Extended versions of the slides provided with the book and with MIPSfpga Labs

• **Labs:**
  – MIPSfpga materials (MIPSfpga Labs and MIPSfpga SoC)

• **Exercises:**
  – Worksheets adapted from textbook and labs
Course Schedule

• **Lectures:** 24 1.5-hour lessons, 2 per week

• **Lab sessions** 12 2-hour sessions, 1 per week
  
  – **Module 1: MIPS ISA & Hardware (5 weeks)**
    • Install MIPSfpga tools and build MIPSfpga in Vivado (MIPS GSG & Lab 1)
    • Review MIPS ISA: assembly & C programming (Labs 2-4)
    • Review: I/O Systems (Lab 5: Add 7-segment displays)

  – **Module 2: Pipelined Processor (3 weeks)**
    • Labs 13-18 (ADD, AND, LW, BEQ instruction flow, Hazard Unit)

  – **Module 3: Cache Hierarchy (3 weeks)**
    • Labs 22A and 22B (Cache hits, Cache misses)

  – **Module 4: SoC and Embedded System Design (1 week)**
    • MIPSfpga SoC Starter Tutorial
Ex. 2: Processor & I/O Systems Lab

• At TUD (Technical University of Darmstadt, Germany) in 2016

• Student background
  – Digital & HDL design (VHDL)
  – Computer Organization (MIPS ISA, single/multi-cycle processors, I/O systems)
  – Programming
Course Materials

• **Textbook:**
  - Digital Design and Computer Architecture, 2\textsuperscript{nd} Edition (Harris & Harris, Elsevier © 2012)

• **Slides:**
  - Extended versions of the slides provided with the book and MIPSfpga Labs

• **Labs:**
  - Textbook and MIPSfpga Labs

• **Lab kit cost:**
  - Buzzer & LCD ($14) + Nexys4 DDR board ($159) + Bus Blaster probe ($50). Total: $223
## Course Contents

<table>
<thead>
<tr>
<th>Lab</th>
<th>MIPSfpga Lab #</th>
<th>Description</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lab 3</td>
<td>MIPS Assembly Programming (using simulator)</td>
<td>1 week</td>
</tr>
<tr>
<td>2</td>
<td>Lab from book</td>
<td>MIPS Single-Cycle Processor</td>
<td>1 week</td>
</tr>
<tr>
<td>3</td>
<td>Lab from book</td>
<td>MIPS Pipelined Processor</td>
<td>2 weeks</td>
</tr>
<tr>
<td>4</td>
<td>Lab 1</td>
<td>MIPSfpga Tutorial</td>
<td>1 week</td>
</tr>
<tr>
<td>5</td>
<td>Lab 7</td>
<td>MIPSfpga Memory-mapped I/O: Buzzer</td>
<td>1 week</td>
</tr>
<tr>
<td>6</td>
<td>Lab 9</td>
<td>MIPSfpga Memory-mapped I/O: LCD</td>
<td>2 weeks</td>
</tr>
<tr>
<td>7</td>
<td>Lab 11</td>
<td>MIPSfpga DMA Engine</td>
<td>2 weeks</td>
</tr>
<tr>
<td>8</td>
<td>Lab 12</td>
<td>MIPSfpga DES Encryption with DMA</td>
<td>2 weeks</td>
</tr>
</tbody>
</table>
Ex. 3: Hackathon

• Seminars in Russia, Ukraine and Kazakhstan in 2015/2016
• MIPSfpga integrated into courses / labs
• Culminated in Hackathon where students interfaced MIPSfpga to peripherals using SPI, I²C, etc. (see MIPSfpga I/O Labs)

http://store.digilentinc.com/pmod-expansion-modules/pmod-boards/
# MIPSfpga Comparisons

<table>
<thead>
<tr>
<th>Name</th>
<th>Comparisons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nios, Nios II, Microblaze, Cortex M0 Design Start</td>
<td>+ Soft-core</td>
</tr>
<tr>
<td></td>
<td>- Obfuscated</td>
</tr>
<tr>
<td>OpenSPARC, Leon</td>
<td>+ Soft-core</td>
</tr>
<tr>
<td></td>
<td>+ Open-source</td>
</tr>
<tr>
<td></td>
<td>- Few teaching materials</td>
</tr>
<tr>
<td></td>
<td>- ISA not commonly used in academia</td>
</tr>
<tr>
<td>RISC-V, openRISC</td>
<td>+ Soft-core</td>
</tr>
<tr>
<td></td>
<td>+ Open-source</td>
</tr>
<tr>
<td></td>
<td>- Not a commercial core</td>
</tr>
<tr>
<td></td>
<td>- Few teaching materials</td>
</tr>
</tbody>
</table>

**MIPSfpga:**
- Unobfuscated soft-core
- Commercial core (e.g., used in Microchip’s PIC32MZ)
- Commonly taught ISA in academia
- Robust teaching materials
Overview

• Introduction
• MIPSfpga Overview
• Labs
• Example Courses and Applications
• Conclusion
Conclusion

• MIPSfpga **bridges the gap** between toy processors and **industry-level processors**.

• MIPSfpga is an **excellent resource for courses** in:
  – Digital design, computer architecture, embedded systems, Memory systems, VLSI design, SoC design

• MIPSfpga offers **robust teaching materials** best used in upper-division undergraduate or master’s-level courses.
MIPSfpga Materials

- MIPSfpga materials available on the Imagination University Program Website under Teaching Materials:
  
  https://community.imgtec.com/university/resources/

  MIPSfpga 2.0 GSG & Labs available July 1, 2017
MIPSfpga Support

• MIPSfpga Forum (technical support):
  http://community.imgtec.com/forums/cat/mips-insider/mipsfpga/

• Other Forums
  – MIPS Tech Support (general questions):
    http://community.imgtec.com/forums/cat/mips-insider/
  – Imagination University Programme (curriculum discussions, IUP questions, etc. – no tech support):
    http://community.imgtec.com/forums/cat/university/
Acknowledgements

- Robert Owen
- Sarah Harris
- Daniel Chaver-Martinez
- David Money Harris
- Yuri Panchul
- Bruce Ableidinger
- Enrique Sedano
- Zubair Kakakhel
- Kent Brinkley
- Chuck Swartley
- Akhilesh Sandeep Thakur
- Christian White
- Sean Raby
- Rick Leatherman
- Matthew Fortune
- Munir Hasan
- Sachin Sundar
- Michael Alexander
- Sam Bobrowicz
- Cathal McCabe
- Roy Kravitz
- Alexey Pereverzev
- Nicholas Beser
- Larissa Swanland
- Clint Cole
- Students and faculty at UCL
- Ian Oliver
- Steve Kromer
- Parimal Patel
- Jason Wong
- Zhe Yang
- Professor Dai
- Zhongzheng (Jason) Wang
- Bin He
- Dennis Pinto
Thank you!
Any questions?