ELEC6234
Embedded Processor Design

Soft processor cores in embedded systems
Microprocessors in embedded systems

• Microprocessors are everywhere
• A modern car has around 40 to 100 microprocessors
  – Engine control, brake systems, active suspension
    • Need powerful, 32-bit processors
  – Even motorised window control has a microprocessor
• An embedded system:
  • a system that utilizes custom hardware and software to carry out specific tasks

• Embedded Hardware:
  • Microprocessor
  • Application-specific hardware generally used for accelerating time-critical tasks

• Embedded software running on the microprocessor
Almost every modern embedded system contains at least one microprocessor

Designers face many challenges
  - Which parts of a system on chip should be implemented as programmable, combinational and sequential logic?
  - Which parts should be implemented as programs on a microprocessor core?
  - What processor architectures should be used?
    - Little point in using a 32-bit MIPS with double precision floating point if one is designing a burglar alarm
Core-based embedded system design

• Designers re-use pre-designed hardware components called “Intellectual Property (IP) Cores”
• This approach reduces design time at the expense of performance, area etc
• Soft-core embedded processors – Complete microprocessors described in a hardware description language (SystemVerilog, Verilog, VHDL) that can be customised for specific tasks
‘Hard’ Processor Cores

• A hard core processor is an embedded dedicated processor, e.g. ARM
  – On an FPGA it is surrounded by FPGA’s memory and programmable logic
  – On an ASIC, a ‘hard’ core is surrounded by custom logic

• The picoMIPS you are designing is a soft core.

• Hard cores are less configurable but have higher performance than soft cores
Hard processor cores

Altera offers ARM cores in some of their FPGAs, e.g. Cyclone V:
Hard processor cores - cont

Xilinx also offers ARM or PowerPC

Two PowerPC hard cores on a Xilinx Virtex II die
ARM Cortex A9 on Altera Cyclone V FPGA
Soft cores

• Use existing FPGA logic elements
• Flexible, feature rich, reconfigurable
• Customized memory size, data path width, ALU functionality, number and types of peripherals
• Typically have slower clock rates
• Are said to consume more power than their hard-core counterparts
  — Nb. I doubt if this is always true
Advantages of soft cores

• Higher level of abstraction – easier to understand
• More flexible – designers can change the core by editing source code or selecting parameters (more on that later)
• Platform independent – can be synthesized for any IC technology, including FPGAs, ASICs, etc.
• – More immune to obsolescence
Examples of soft cores

• Altera: NIOS II – one of the most popular
• Xilinx: MicroBlaze, PicoBlaze
• Lattice: LatticeMico8
• ARM Cortex M1
  – a third party core offered to FPGA vendors
• There are many open source soft cores, e.g.
  – SPARC - GPL licence, (e.g. www.simplyrisc.com)
  – OpenRISC – also GPL (e.g opencores.org)
  – DSPuva16 – from University of Valladolid, Spain
Development tools

• A dedicated CAD tool, specific to the particular core, is used to specify processor parameters
  – Register file size, hardware arithmetic, floating point, interrupts, I/O hardware, custom functions
  – User programs, typically written in C or C++, are compiled by a custom compiler provided by the CAD tool and included in the synthesis

• The tool outputs a synthesizable HDL model
  – Normally the HDL code is protected and not viewable

• After any user-specified hardware is added, the design is synthesised using a standard synthesis tool
  – NB. Altera Quartus integrates specification for the NIOS processor into a single tool used for general purpose synthesis
  – Altera also provide a SOPC (System on Programmable Chip) Builder for advanced embedded system design
Comparison of some soft core processors

<table>
<thead>
<tr>
<th>Category</th>
<th>Nios II (Fast Core)</th>
<th>MicroBlaze</th>
<th>Xtensa XL</th>
<th>OpenRISC 1200</th>
<th>LEON3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum MHz</td>
<td>200 (FPGA)</td>
<td>200 (FPGA)</td>
<td>350 (ASIC)</td>
<td>300 (ASIC)</td>
<td>400/125 (ASIC/FPGA)</td>
</tr>
<tr>
<td>ASIC/FPGA Technology</td>
<td>/Stratix and Stratix II</td>
<td>/Virtex-4</td>
<td>0.13 μm/ –</td>
<td>0.18 μm/ –</td>
<td>0.13 μm/Not given</td>
</tr>
<tr>
<td>Reported DMIPS</td>
<td>150 DMIPS</td>
<td>166 DMIPS</td>
<td>–</td>
<td>250 DMIPS</td>
<td>85 DMIPS</td>
</tr>
<tr>
<td>ISA</td>
<td>32-bit RISC</td>
<td>32-Bit RISC</td>
<td>32-bit RISC</td>
<td>32-bit RISC</td>
<td>32 or 64-bit RISC</td>
</tr>
<tr>
<td>Cache Memory (I/D)</td>
<td>Up to 64 KB</td>
<td>Up to 64 KB</td>
<td>Up to 32 KB (1)</td>
<td>Up to 64 KB</td>
<td>Up to 256 KB</td>
</tr>
<tr>
<td>Floating Point Unit (optional)</td>
<td>IEEE-754</td>
<td>IEEE-754</td>
<td>IEEE-754</td>
<td>As peripheral</td>
<td>IEEE-754</td>
</tr>
<tr>
<td>Pipeline</td>
<td>6 Stages</td>
<td>3 Stages</td>
<td>5 Stages</td>
<td>5 Stages</td>
<td>7 Stages</td>
</tr>
<tr>
<td>Custom Instructions</td>
<td>Up to 256 Instructions</td>
<td>None</td>
<td>Unlimited</td>
<td>Unspecified limit</td>
<td>None</td>
</tr>
<tr>
<td>Register File Size</td>
<td>32</td>
<td>32</td>
<td>32 or 64</td>
<td>32</td>
<td>2 to 32</td>
</tr>
<tr>
<td>Implementation</td>
<td>FPGA</td>
<td>FPGA</td>
<td>FPGA, ASIC</td>
<td>FPGA, ASIC</td>
<td>FPGA, ASIC</td>
</tr>
<tr>
<td>Area</td>
<td>700-1800 LEs</td>
<td>1269 LUTs</td>
<td>0.26 mm²</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

source: Tong J.G e.a. “Soft core processors for embedded systems”, 18th Int Conf on Microelectronics (ICM), 2006
Alterna NIOS II Soft Core Processor

- A general purpose RISC with a Harvard memory architecture.
- Features a 32-bit Instruction Set Architecture (ISA), 32 general-purpose registers, single-instruction 32x32 multiply and divide operations, and dedicated instructions for 64-bit and 128-bit products of multiplication.
- Nios II has a performance of more than 150 Dhrystone MIPS (DMIPS) on Stratix family of FPGAs.
- Comes in three versions: economy, standard and fast core.
- Each core version has a different number of pipeline stages, instruction and data cache memories and hardware components for multiply and divide operations.
- Each core implementation varies in size and performance depending on the features that are selected.
- Adding peripherals with the Nios II Processors is done through the Avalon Interface Bus which contains the necessary logic to interface the processor with off-the-shelf IP cores or custom designed peripherals.
NIOS II architecture

source: altera.com
NIOS II Scalability

Low-Cost Embedded Solution

Processor?

Cyclone Series & Nios II Economy
- CPU < 20% of Device
- 20 DMIPS
- As Low as 35¢

Complex Embedded System-on-a-Chip

How Many Processors?
179,400 Logic Elements

Stratix II EP2S180 & Nios II Fast
- CPU 1% of Device

source: altera.com
NIOS II Scalability – cont.

- 32-Bit Pipelined RISC Architecture
  - 16-Bit Instructions
  - Most Instructions Take 1 Clock
- Large Internal Register File
- Configurable Data Path
  - 16-bit (1100 LEs)
  - 32-bit (1700 LEs)
- Dynamic Bus Sizing
- 30 to 80 MIPS Performance

source: altera.com
## NIOS II cache and RAM performance

<table>
<thead>
<tr>
<th>Memory</th>
<th>I-Cache</th>
<th>D-Cache</th>
<th>Normalised Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDRAM</td>
<td>No</td>
<td>No</td>
<td>40.2%</td>
</tr>
<tr>
<td>SDRAM</td>
<td>No</td>
<td>Yes</td>
<td>55.2%</td>
</tr>
<tr>
<td>SDRAM</td>
<td>Yes</td>
<td>No</td>
<td>64.3%</td>
</tr>
<tr>
<td>SDRAM</td>
<td>Yes</td>
<td>Yes</td>
<td>96.4%</td>
</tr>
<tr>
<td>OnChip</td>
<td>No</td>
<td>No</td>
<td>100.0%</td>
</tr>
<tr>
<td>OnChip</td>
<td>No</td>
<td>Yes</td>
<td>98.0%</td>
</tr>
<tr>
<td>OnChip</td>
<td>Yes</td>
<td>No</td>
<td>110.2%</td>
</tr>
<tr>
<td>OnChip</td>
<td>Yes</td>
<td>Yes</td>
<td>105.6%</td>
</tr>
</tbody>
</table>

Performance relative to on chip RAM with no Cache running dhry.c modified for unbuffered I/O

source: altera.com
NIOS II windowed register file

- Common Technique Used by High-Performance CPUs
  - Provides Fast Subroutine Calls

- Up to 512 General-Purpose Registers

- Movable Window With Access to 32 Registers
  - 24 Register Window (Movable)
  - 8 Global Registers (Fixed)

- Automatically Used by C Compiler

source: altera.com
NIOS II Multiplier option

- MSTEP – hardware option 1
  - Sequential, one-bit per clock,
  - cheap but less performance
- MUL – hardware option 2
  - Combinational, 16x16 bit multiplication in 2 clocks

<table>
<thead>
<tr>
<th>Multiplier</th>
<th>Additional Logic Elements Used</th>
<th>Clock Cycles 16x16&gt;32</th>
<th>Clock Cycles 32x32&gt;32</th>
</tr>
</thead>
<tbody>
<tr>
<td>None (Software)</td>
<td>0</td>
<td>80</td>
<td>250</td>
</tr>
<tr>
<td>MSTEP</td>
<td>+200</td>
<td>18</td>
<td>80</td>
</tr>
<tr>
<td>MUL</td>
<td>+400</td>
<td>2</td>
<td>16</td>
</tr>
</tbody>
</table>

source: altera.com
Typical applications of NIOS

• Many Altera recommended standard HDL modules for Altera FPGA development systems are based on NIOS, e.g.
  – USB port
  – Ethernet
  – UART
  – Parallel I/O port
  – SPI (Serial Peripheral Interface) ports
  – Timers
NIOS II application example: Ethernet controller

- 10/100-Mbps SMSC LAN91C111 single-chip Ethernet controller interface
- Software support provided by the NicheStack TCP/IP Stack - Nios II Edition
- Included with the Nios II development kits

source: altera.com
Open source soft cores

• There are a great number of soft-cores freely available from open-source communities and commercial vendors

• Some examples
  – LEON by Gaisler Research
  – OpenRISC 1200 from opencores.org
  – RISC V from riscv.org
  – Ensilica, both open and commercial cores, [www.ensilica.com](http://www.ensilica.com)
  
  • Some Ensilica open source projects:
    – FT816Float - Floating point accelerator
    – Versatile FIFO
    – SPI Verilog Master & Slave modules
    – AVR Core
Summary

• Popularity of embedded soft-core and hard-core processors increases
  – It is difficult to imagine a complex system-on-chip design without at least one embedded processor core

• Most popular embedded processors use RISC/Harvard architectures

• ARM Cortex seems to be the most popular hard core

• Altera NIOS II – most popular soft core

• Some good videos on youtube:
  – An Altera man explains NIOS: http://www.youtube.com/watch?v=GbG0NyQ1zzU
  – Another Altera man advocates soft MIPS: http://www.youtube.com/watch?feature=player_embedded&v=TWWS3ckJKw
  – A student demonstrates his dual core soft MIPS on FPGA: http://www.youtube.com/watch?v=PJTNvYhVhks, see also: www.realmb.com/portfolio/