Prototyping Architectural Support for Program Rollback Using FPGAs

Radu Teodorescu
Josep Torrellas

http://iacoma.cs.uiuc.edu
University of Illinois
Summary

• Problem
  – Production software is hard to debug

• Solution
  – Always-on, lightweight debugger
  – Collect info about bug circumstances
  – Hardware support

• FPGA platform
  – Rapid prototyping
  – Design validation
Debugging Production Code

• Processor runs in two possible modes:
  – Normal
  – Speculative
    • Rollback capability

• Transition controlled by software
  – Special instructions
Debugging Production Code

num=a+b;
...
begin_spec();
p1=m[a[*x]]+a[m[&y]];
p2=&p1;
foo(p2);
...
if (rlbk_state) {
    collect[0]=&p1;
    collect[1]=&y;
}
end_spec(flag);
num=num+c;
...

non-speculative code

begin speculation

error-prone code

collect info on re-execution

end speculation

non-speculative code
# Hardware Support

<table>
<thead>
<tr>
<th>New hardware</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speculative cache</td>
<td>Buffer speculative data</td>
</tr>
<tr>
<td>Register checkpointing</td>
<td>Restore processor state</td>
</tr>
<tr>
<td>ISA support</td>
<td>Instructions to mark the speculative section</td>
</tr>
<tr>
<td>Performance counters</td>
<td>Feedback</td>
</tr>
</tbody>
</table>
Experimental Infrastructure

<table>
<thead>
<tr>
<th>Baseline processor</th>
<th>SPARC V8 compliant</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>In-order, single-issue, 5 stage pipeline</td>
</tr>
<tr>
<td></td>
<td>Open source VHDL, Gaisler Research</td>
</tr>
<tr>
<td>Caches</td>
<td>1-4 set associative, 1-64KB/set</td>
</tr>
<tr>
<td>System on a Chip</td>
<td>PCI, Ethernet, serial interfaces</td>
</tr>
<tr>
<td>Development board</td>
<td>Xilinx Virtex II XC2V3000, 64 Mbytes SDRAM</td>
</tr>
<tr>
<td>Operating System</td>
<td>Linux embedded</td>
</tr>
</tbody>
</table>
FPGA System Architecture

- Processor Bit File
- FPGA Programming Tool
- Binaries
- Communication & Control App
- Output Terminal
- PCI
- Serial
- JTAG
- Boot PROM
- Config. PROM
- Xilinx Virtex-II FPGA
- SDRAM

Development Board
Ongoing Work

• Operating system support
  – Extend speculation coverage
  – Kernel debugging

• Compiler support
  – Analysis and instrumentation

• Extend monitoring counters
  – Collect information that can help debugging
Conclusions

- We implemented a hardware prototype for software-controlled speculation
- Use it to debug production software
- FPGA platform
  - Validate the design
  - Experiment with real applications, including the Linux kernel
  - Evaluate hardware overhead
Prototyping Architectural Support for Program Rollback Using FPGAs

Contact:
Radu Teodorescu and Josep Torrellas
http://iacoma.cs.uiuc.edu