Designing a Control Application by Using a Specialized Multi-Core Soft Microprocessor

Bernd Däne, Alexander Pacholik, Stephan Zschäck, Wolfgang Fengler, Christoph Ament, Tobias Braune

Ilmenau University of Technology
Germany

www.tu-ilmenau.de/ra
Topics

1. Motivation and prerequisites
2. Approach and concept
3. Processor architecture
4. Design flow and tools
5. Experimental results
6. Conclusion and further work
Motivation

- Computationally intensive algorithms
e.g. digital filters and controllers
- Most important requirement: Low latency
- Realization using FPGA
  widespread hardware platform,
tightly coupled I/O
- Case study: Control algorithm for measuring machine
Control Algorithm

- For 6 axes: Approx. 300 MAC per step
- Double precision floating point calculation
- **Goal:** Latency (execution time) below 3 µs
Prerequisites

From surrounding project:

- Modular system: *cRIO* (National Instruments)
- FPGA included: *Virtex-5 LX110* (Xilinx)
- Design tool: *LabView* (National Instruments)
Approach

- Direct implementation of dataflow graph
  → High resource consumption, low utilization
- Same, but multiple use of hardware blocks
  → Timing hard to optimize
- Existing soft microprocessors
  → Not adapted to specific problem, rare DOUBLE support

Therefore:
Designing a problem-specific soft microprocessor
Architectural Concept

- Harvard style RISC processor
- Configurable instruction set
- Omitting unneeded features
- Simple pipeline: No dependency logic
- No branches and loops
- Predicated instructions
- Flattened memory architecture
Consequences

- All optimizations at compile time
- No wait states
- Statically defined timing
- Reduced resource consumption
Processor Core Structure

MUX

Operator 1
... ...
Operator n

In
Read 1
on-the-fly
Pre-Operation 1
Instruct.
Pre-Operation 2
Addr. 1
Addr. 2
...
...
Addr.
Operation
Address
Out
Write

Instruction Decoder

PDES 2013 Designing a Control Application by Using a Specialized Multi-Core Soft Microprocessor
On-the-fly-operations: e.g. negation, absolute value (optionally)
Operators: e.g. add, multiply, square root, type conversion
Processor Core Structure

Multiplexer: Selection of required result
Instruction decoder: Separates opcodes and addresses
Processor Core Structure

Interface: For memory blocks and input/output ports

Diagram showing the processor core structure with inputs, operators, and outputs.
Single-Core Setup

- Counter
- Program Memory
- Instructions
- Core
- Write Address
- Read 1
- Addr. 1
- Read 2
- Addr. 2
- Data Mem. 1
- Data Mem. 2
- In
- Out

PDES 2013
Designing a Control Application by Using a Specialized Multi-Core Soft Microprocessor
Single-Core Setup

Instructions: Continuously (no branches, no stalls)
Write: Identically into two memory blocks
Single-Core Setup

Read: One operand per memory block
Single-Core Setup

Input and output: Addressable registers
Multi-Core Structures

- Raising instruction level parallelism
- VLIW architecture (Very Long Instruction Word)
- Shared data memory space (register set)
- Problem: Memory ports limited
  → Further duplication of memory
Dual-Core Setup

- Write
- Read 1
- Read 2

Core 1

- Memory 1a
- Memory 1b

- Memory 2a
- Memory 2b

- Memory 3a
- Memory 3b

- Memory 4a
- Memory 4b

Core 2

- Write
- Read 1
- Read 2
Write: 4x identically

Dual-Core Setup

Write: 4x identically
Dual-Core Setup

Write: 4x identically
Read: Two operands per core
PDES 2013
Designing a Control Application by Using a Specialized Multi-Core Soft Microprocessor

Dual-Core Setup

Read: Two operands per core
Memory Consumption

- For $n$-core configuration:
  - $2 \cdot n$ identical copies
  - $2 \cdot n^2$ individually addressable blocks
- Tested so far:
  - Quad-core, logical memory space: 4,096 cells
  - Physical memory consumption: 262,144 bytes
  - „Block RAM“ consumption for LX110: 46 %
All components use *LabView* design tool

Except floating point operators: VHDL library

Fully integrated design process in preparation
Example Models

- Code Converter detail
PDES 2013 Designing a Control Application by Using a Specialized Multi-Core Soft Microprocessor
Example Models

- Processor Core detail
Designing a Control Application by Using a Specialized Multi-Core Soft Microprocessor
Example Models

- Testbench front panel
PDES 2013 Designing a Control Application by Using a Specialized Multi-Core Soft Microprocessor
## Experimental Results

<table>
<thead>
<tr>
<th>6-Axes Control Algorithm</th>
<th>Instruction Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADD, SUB</td>
<td>361</td>
</tr>
<tr>
<td>MUL</td>
<td>296</td>
</tr>
<tr>
<td>DIV</td>
<td>3</td>
</tr>
<tr>
<td>Type conversions</td>
<td>26</td>
</tr>
<tr>
<td>Transports</td>
<td>180</td>
</tr>
</tbody>
</table>
## Experimental Results

<table>
<thead>
<tr>
<th>6-Axes Control Algorithm</th>
<th>Instruction Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>866</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of cores</th>
<th>1</th>
<th>2</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock cycle count</td>
<td>993</td>
<td>578</td>
<td>321</td>
</tr>
<tr>
<td>Execution time [µs]</td>
<td>8.28</td>
<td>4.82</td>
<td>2.68</td>
</tr>
</tbody>
</table>

Double-precision floating point, clock frequency 120 MHz
Experimental Results

<table>
<thead>
<tr>
<th>Ellipse regression, 4 channels</th>
<th>Instruction Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADD, SUB</td>
<td>432</td>
</tr>
<tr>
<td>MUL</td>
<td>460</td>
</tr>
<tr>
<td>DIV</td>
<td>44</td>
</tr>
<tr>
<td>SQR</td>
<td>8</td>
</tr>
<tr>
<td>Type conversions</td>
<td>32</td>
</tr>
<tr>
<td>Transports</td>
<td>480</td>
</tr>
</tbody>
</table>
# Experimental Results

<table>
<thead>
<tr>
<th>Ellipse regression, 4 channels</th>
<th>Instruction Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>1456</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of cores</th>
<th>1</th>
<th>2</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock cycle count</td>
<td>1725</td>
<td>1142</td>
<td>938</td>
</tr>
<tr>
<td>Execution time [µs]</td>
<td>14.38</td>
<td>9.52</td>
<td>7.82</td>
</tr>
</tbody>
</table>

Double-precision floating point, clock frequency 120 MHz
Conclusion

- Case study about problem-specific soft microprocessor: Workflow and implementation investigated
- Model-based design process
- Experimental results for multi-core configurations

Funded by Deutsche Forschungsgemeinschaft (German Research Foundation)
Further Work

- Architectural improvements
- More experiments with various algorithms
- Model based source code design
- Model based configuration decisions e.g. instruction set definition
- Integrated optimization process
Thank you for your kind attention