FastCPU

Arshad Hussain
Lixin Su
Jun Tang
What is FastCPU?

- 6 stage pipeline RISC processor
- 16 bit instructions set
- 16 bit internal data bus
- 8 bit I/O bus
- 8-bit CLA adder, 8bit bi-dir Barrel Shifter, 4 bit Multiplier
- **Tap Controller** for serial scan of PC out on a pin for debugging.
- Total number of transistors **8500** (without Pad Frame)
- Fastest clock speed: **42 MHz**
- **Status:**
  - Design and testing of all blocks are **complete**
  - IRSim testing from pads is **complete!**
  - Mission accomplished!
Our Pipeline

1. **Stall** is inserted to avoid the structural hazard (Due to the pin number limitation, the data and instruction memory accesses share the same 8 pins).

2. There must be an ALU operation before conditional branches. The target address of the next instruction can always be resolved in the ID stage of the current instruction.

3. Data forwarding is used in the ID stage when needed.
# Instruction Set

<table>
<thead>
<tr>
<th>operation</th>
<th>op code (15-12)</th>
<th>destination register (11-9)</th>
<th>source register (7-5)</th>
<th>target register (4-2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADD</td>
<td>1000</td>
<td>Rd</td>
<td>Rs</td>
<td>Rt</td>
</tr>
<tr>
<td>SUB</td>
<td>1001</td>
<td>Rd</td>
<td>Rs</td>
<td>Rt</td>
</tr>
<tr>
<td>MULT</td>
<td>1010</td>
<td>Rd</td>
<td>Rs</td>
<td>Rt</td>
</tr>
<tr>
<td>AND</td>
<td>1011</td>
<td>Rd</td>
<td>Rs</td>
<td>Rt</td>
</tr>
<tr>
<td>OR</td>
<td>1100</td>
<td>Rd</td>
<td>Rs</td>
<td>Rt</td>
</tr>
<tr>
<td>NOT</td>
<td>1101</td>
<td>Rd</td>
<td>Rs</td>
<td>N/A</td>
</tr>
<tr>
<td>SLL</td>
<td>0101</td>
<td>Rd</td>
<td>Rs</td>
<td>Rt</td>
</tr>
<tr>
<td>SRL</td>
<td>1111</td>
<td>Rd</td>
<td>Rs</td>
<td>Rt</td>
</tr>
<tr>
<td>JR</td>
<td>0000</td>
<td>N/A</td>
<td></td>
<td>Imm Addr</td>
</tr>
<tr>
<td>JC</td>
<td>0001</td>
<td>N/A</td>
<td></td>
<td>Imm Addr</td>
</tr>
<tr>
<td>JZ</td>
<td>0011</td>
<td>N/A</td>
<td></td>
<td>Imm Addr</td>
</tr>
<tr>
<td>LW</td>
<td>0100</td>
<td>Rd</td>
<td></td>
<td>Imm Addr</td>
</tr>
<tr>
<td>SW</td>
<td>0010</td>
<td>Rs</td>
<td></td>
<td>Imm Addr</td>
</tr>
<tr>
<td>LWI</td>
<td>0101</td>
<td>Rd</td>
<td></td>
<td>Imm Data</td>
</tr>
<tr>
<td>NOP</td>
<td>0110</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>END</td>
<td>0111</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Timing Analysis (1)
Timing Analysis (3)
Main PLA

- 8 inputs
- 21 outputs
- 17 states, 5 status bits
Test Access Port Controller – for CPU Debugging

- **halt_en**
- **tap_en**

**Main Controller**

- giving control to the tap controller

**TAP Controller**

**State diagram for TAP**

- `idle`
- `CAP`
- `UFG`

- TDI
- TDO
Scan Cell For TAP

WIN

SIN

MUX

CAP_SEL

LA

CAP_EN

LB

UPG_EN

SOUT

SOUT
Register File
Carry Look-Ahead Adder
Carry Look-Ahead Adder
Multiplier
Barrel Shifter
ALU
Floor Plan
IRSIM Test Code Sequences

We have four testing code sequences

LW R0 #129;  LW R4 #168;  LWI R0, #18;  LW R2 #129;
LW R1 #130;  LWI R5 #3;  LW R1 #154;  LWI R3 #70;
ADD R0 R1 R0;  MULT R4 R4 R5;  AND R2 R0 R1;  (HALT)
ADD R0 R0 R1;  STORE R4 #168;  JR #32;  SUB R2 R2 R3;
JC #114;  OR R3 R0 R1;  NOT R4 R0;  S LL R5 R0 R1;
                                 SRL R6 R0 R1;  ADD R7 R0 R1;
                                 SW R6 #145;
Memory Access Testing
Memory Access Testing Result

![Memory Access Testing Diagram](image-url)
TAP Testing Result

LW R2 #129;
LWI R3 #70;
(HALT)
SUB R2 R2 R3;
Testing (JZ)

LW R2 #129;
LWI R3 #70;
SUB R2 R2 R3;
JZ #85;
Testing (JC)

LW R0 #129;
LW R1 #130;
ADD R0 R1 R0;
ADD R0 R0 R1;
JC #114;
General ALU Testing

LWI R0, #18;
LW R1 #154;
AND R2 R0 R1;
JR #32;
OR R3 R0 R1;
NOT R4 R0;
SLL R5 R0 R1;
SRL R6 R0 R1;
ADD R7 R0 R1;
SW R6 #145;
Spice Analysis (Barrel Shifter)
Spice Analysis (Multiplier)

Longest Path Delay: 12ns
Full Chip

Timing Analysis with Crystal and Spice Analysis result:

- The longest path 12ns delay VbSa signal through 4x4 multiplier
- Smallest clock cycle period 24ns for pipeline.
- Max frequency 42 MHz.
- 8500 transistors (excluding the pad frame)
What We Have Learned

• Coordination and cooperation!
• Always keep yourself on schedule!
• Learn fast and apply what you learn fast!
The End

Go, Elec422!
Go, AMD!