Instruction Pipelining and Hazards

Daniel Sanchez
Computer Science and Artificial Intelligence Laboratory
M.I.T.

http://www.csg.csail.mit.edu/6.823
Princeton Microarchitecture
Datapath & Control for 2-cycles-per-instruction
Princeton Microarchitecture
Datapath & Control for 2-cycles-per-instruction
The same (mux not shown)

Only one of the phases is active in any cycle
⇒ a lot of datapath is not in use at any given time
Princeton Microarchitecture
Overlapped execution

fetch phase

execute phase
Princeton Microarchitecture
Overlapped execution

Can we overlap instruction fetch and execute?
Princeton Microarchitecture
Overlapped execution

Can we overlap instruction fetch and execute?
Yes, unless IR contains a Load or Store
Can we overlap instruction fetch and execute?

Yes, unless IR contains a Load or Store

Which action should be prioritized?
Princeton Microarchitecture
Overlapped execution

Can we overlap instruction fetch and execute?

Yes, unless IR contains a Load or Store

Which action should be prioritized?  Execute
Princeton Microarchitecture
Overlapped execution

Can we overlap instruction fetch and execute?
Yes, unless IR contains a Load or Store

Which action should be prioritized? Execute

What do we do with Fetch?
Princeton Microarchitecture
Overlapped execution

Can we overlap instruction fetch and execute?
Yes, unless IR contains a Load or Store

Which action should be prioritized? Execute

What do we do with Fetch? Stall it
Princeton Microarchitecture
Overlapped execution

Can we overlap instruction fetch and execute?

Yes, unless IR contains a Load or Store

Which action should be prioritized?  Execute

What do we do with Fetch?  Stall it

How?
Stalling the instruction fetch

Princeton Microarchitecture

fetch phase

execute phase

stall?
Stalling the instruction fetch
Princeton Microarchitecture

When stall condition is indicated
Stalling the instruction fetch
Princeton Microarchitecture

When stall condition is indicated
• *don’t* fetch a new instruction and *don’t* change the PC
When stall condition is indicated
  • don’t fetch a new instruction and don’t change the PC
Stalling the instruction fetch
Princeton Microarchitecture

When stall condition is indicated
- don’t fetch a new instruction and don’t change the PC
- insert a nop in the IR
Stalling the instruction fetch

Princeton Microarchitecture

When stall condition is indicated
- don’t fetch a new instruction and don’t change the PC
- insert a nop in the IR
- set the Memory Address mux to ALU (not shown)
When stall condition is indicated
- don’t fetch a new instruction and don’t change the PC
- insert a nop in the IR
- set the Memory Address mux to ALU (not shown)

What if IR contains a jump or branch instruction?
Need to stall on branches

Princeton Microarchitecture

fetch phase

execute phase
Need to stall on branches

Princeton Microarchitecture

When IR contains a jump or branch-taken
- no "structural conflict" for the memory
Need to stall on branches
Princeton Microarchitecture

When IR contains a jump or branch-taken
• no "structural conflict" for the memory
• but we do not have the correct PC value in the PC
Need to stall on branches

Princeton Microarchitecture

When IR contains a jump or branch-taken

- *no “structural conflict” for the memory*
- *but we do not have the correct PC value in the PC*
- *memory cannot be used – Address Mux setting is irrelevant*
**Need to stall on branches**

**Princeton Microarchitecture**

- When IR contains a jump or branch-taken:
  - *no “structural conflict” for the memory*
  - *but we do not have the correct PC value in the PC*
  - *memory cannot be used – Address Mux setting is irrelevant*
  - *insert a nop in the IR*

---

**fetch phase**

**execute phase**
Need to stall on branches

Princeton Microarchitecture

When IR contains a jump or branch-taken

- no “structural conflict” for the memory
- but we do not have the correct PC value in the PC
- memory cannot be used – Address Mux setting is irrelevant
- insert a nop in the IR
- insert the nextPC (branch-target) address in the PC
Need to stall on branches

Princeton Microarchitecture

When IR contains a jump or branch-taken

• no “structural conflict” for the memory
• but we do not have the correct PC value in the PC
• memory cannot be used – Address Mux setting is irrelevant
• insert a nop in the IR
• insert the nextPC (branch-target) address in the PC

fetch phase

execute phase
Pipelined Princeton Microarchitecture
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUi</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUiu</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LW</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ(z=0)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ(z=1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAAddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td></td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUi</td>
<td></td>
<td>sE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUiu</td>
<td></td>
<td>uE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LW</td>
<td></td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td></td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td></td>
<td>sE₁₆</td>
<td>* 0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td></td>
<td>sE₁₆</td>
<td>* 0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAddSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LW</td>
<td></td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td></td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZₜ=₀</td>
<td></td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZₜ=₁</td>
<td></td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LW</td>
<td></td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td></td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td></td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td></td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAaddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>IR Src</th>
<th>MA addr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>LW</td>
<td></td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td></td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;_z=0&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;_z=1&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ;  WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAaddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE\textsubscript{16}</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE\textsubscript{16}</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td></td>
<td>sE\textsubscript{16}</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td></td>
<td>sE\textsubscript{16}</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ\textsubscript{z=0}</td>
<td></td>
<td>sE\textsubscript{16}</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ\textsubscript{z=1}</td>
<td></td>
<td>sE\textsubscript{16}</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAAddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ;  WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;_z=0&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;_z=1&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
</tr>
<tr>
<td>BEQZ₀</td>
<td></td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ₁</td>
<td></td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ;  WBSrc = ALU / Mem / PC;  IRSrc = nop/mem;  MAaddr Src = pc/ALU
RegDst = rt / rd / R31;  PCSrc1 = pc+4 / br / rind / jabs;  PCSrc2 = pc/nPC
### Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZₜ=₀</td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZₜ=₁</td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**BSrc** = Reg / Imm  ;  **WBSrc** = ALU / Mem / PC;  **IRSrc** = nop/mem;  **MAaddr Src** = pc/ALU  
**RegDst** = rt / rd / R31;  **PCSrc1** = pc+4 / br / rind / jabs;  **PCSrc2** = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td>br</td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td></td>
<td></td>
<td>jabs</td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td></td>
<td></td>
<td></td>
<td>jabs</td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td>rind</td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td></td>
<td></td>
<td></td>
<td>rind</td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>nos</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAddSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE_{16}</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE_{16}</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE_{16}</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE_{16}</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ_{z=0}</td>
<td>sE_{16}</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ_{z=1}</td>
<td>sE_{16}</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ;  WBSrc = ALU / Mem / PC;  IRSrc = nop/mem;  MAddrSrc = pc/ALU  
RegDst = rt / rd / R31;  PCSrc1 = pc+4 / br / rind / jabs;  PCSrc2 = pc/nPC
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAddSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td></td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAaddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;_z=0&lt;/sub&gt;</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td>npc</td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;_z=1&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAaddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td></td>
<td></td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;_z=0&lt;/sub&gt;</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td>npc</td>
<td>nop</td>
<td></td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;_z=1&lt;/sub&gt;</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAaddrSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
# Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JAL</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JALR</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
<td></td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ;  WBSrc = ALU / Mem / PC;  IRSrc = nop/mem;  MAaddrSrc = pc/ALU  
RegDst = rt / rd / R31;  PCSrc1 = pc+4 / br / rind / jabs;  PCSrc2 = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
<td>ALU</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE₁₆</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE₁₆</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZₜ₀</td>
<td>yes</td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>BEQZₜ₁</td>
<td>sE₁₆</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>JAL</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>JR</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>JALR</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ;  WBSrc = ALU / Mem / PC;  IRSrc = nop/mem;  MAaddr Src = pc/ALU
RegDst = rt / rd / R31;  PCSrc1 = pc+4 / br / rind / jabs;  PCSrc2 = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>ALUiU</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
</tr>
<tr>
<td>J</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>JAL</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>JR</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>JALR</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm  ;  WBSrc = ALU / Mem / PC;  IRSrc = nop/mem;  MAaddrSrc = pc/ALU
RegDst = rt / rd / R31;  PCSrc1 = pc+4 / br / rind / jabs;  PCSrc2 = pc/nPC
### Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>OpSel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>IR Src</th>
<th>MAAddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>*</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE16</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE16</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE16</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE16</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop ALU</td>
</tr>
<tr>
<td>BEQZz=0</td>
<td>yes</td>
<td>sE16</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>BEQZz=1</td>
<td>no</td>
<td>sE16</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
<tr>
<td>J</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>JAL</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>JR</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>JALR</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td>npc</td>
<td>nop</td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAAddr Src = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC
## Pipelined Princeton: Control Table

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>Ext Sel</th>
<th>B Src</th>
<th>Op Sel</th>
<th>Mem W</th>
<th>Reg W</th>
<th>WB Src</th>
<th>Reg Dst</th>
<th>PC Src1</th>
<th>PC Src2</th>
<th>IR Src</th>
<th>MAaddr Src</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALU</td>
<td>no</td>
<td>Reg</td>
<td>Func</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rd</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
<td></td>
</tr>
<tr>
<td>ALUi</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</td>
<td>uE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>Op</td>
<td>no</td>
<td>yes</td>
<td>ALU</td>
<td>rt</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>LW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>no</td>
<td>yes</td>
<td>Mem</td>
<td>rt</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>Imm</td>
<td>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&lt;/sub&gt;</td>
<td>yes</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>br</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=1&lt;/sub&gt;</td>
<td>no</td>
<td>sE&lt;sub&gt;16&lt;/sub&gt;</td>
<td>*</td>
<td>0?</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
<tr>
<td>J</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>jabs</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>JAL</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>jabs</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>JR</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>rind</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>JALR</td>
<td>yes</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td>rind</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>NOP</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</td>
<td>npc</td>
<td>mem</td>
<td>pc</td>
</tr>
</tbody>
</table>

BSrc = Reg / Imm ; WBSrc = ALU / Mem / PC; IRSrc = nop/mem; MAaddr Src = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PCSrc2 = pc/nPC

* stall & IRSrc columns are identical
Pipelined Princeton Architecture

Clock: \[ t_{C-Princeton} > t_{RF} + t_{ALU} + t_{M} + t_{WB} \]

CPI: \( (1- f) + 2f \) cycles per instruction
where \( f \) is the fraction of instructions that cause a stall
Pipelined Princeton Architecture

Clock: \[ t_{C-Princeton} > t_{RF} + t_{ALU} + t_{M} + t_{WB} \]

CPI: \( (1 - f) + 2f \) cycles per instruction where \( f \) is the fraction of instructions that cause a stall

What is a likely value of \( f \)?
An Ideal Pipeline

- All objects go through the same stages
- No sharing of resources between any two stages
- Propagation delay through all pipeline stages is equal
- The scheduling of an object entering the pipeline is not affected by the objects in other stages

*These conditions generally hold for industrial assembly lines.*

*But what about an instruction pipeline?*
Pipelined Datapath
Pipelined Datapath

Diagram showing the pipelined datapath, including blocks labeled as PC, addr, rdata, IR, Imm Ext, ALU, and Memory with data paths and signal connections.
Pipelined Datapath

fetch phase

decode & Reg-fetch phase

execute phase

memory phase

write-back phase
Clock period can be reduced by dividing the execution of an instruction into multiple cycles

\[ t_C > \max\{t_{\text{IM}}, t_{\text{RF}}, t_{\text{ALU}}, t_{\text{DM}}, t_{\text{RW}}\} \ (= t_{\text{DM}} \text{ probably}) \]

However, CPI will increase unless instructions are pipelined.
How to divide the datapath into stages

Suppose memory is significantly slower than other stages. In particular, suppose

\[ t_{IM} = 10 \text{ units} \]
\[ t_{DM} = 10 \text{ units} \]
\[ t_{ALU} = 5 \text{ units} \]
\[ t_{RF} = 1 \text{ unit} \]
\[ t_{RW} = 1 \text{ unit} \]

Since the slowest stage determines the clock, it may be possible to combine some stages without any loss of performance.
Alternative Pipelining

\[ t_C > \max \{ t_{IM}, t_{RF}, t_{ALU}, t_{DM}, t_{RW} \} = t_{DM} \]
Alternative Pipelining

\[ t_C > \text{max} \{ t_{IM}, t_{RF}, t_{ALU}, t_{DM}, t_{RW} \} = t_{DM} \]
Alternative Pipelining

\[ t_C > \max \{ t_{IM}, t_{RF} + t_{ALU}, t_{DM}, t_{RW} \} = t_{DM} \]
Alternative Pipelining

$t_C > \max \{t_{IM}, t_{RF} + t_{ALU}, t_{DM}, t_{RW}\} = t_{DM}$

Write-back stage takes much less time than other stages. Suppose we combined it with the memory phase.
Alternative Pipelining

\[ t_C > \max \{ t_{IM}, t_{RF} + t_{ALU}, t_{DM}, t_{RW} \} = t_{DM} \]

Write-back stage takes much less time than other stages. Suppose we combined it with the memory phase
Alternative Pipelining

Write-back stage takes much less time than other stages. Suppose we combined it with the memory phase
Alternative Pipelining

\[ t_C > \max \{ t_{IM}, t_{RF} + t_{ALU}, t_{DM} + t_{RW} \} = t_{DM} + t_{RW} \]

⇒ increase the critical path by 10% 

Write-back stage takes much less time than other stages. Suppose we combined it with the memory phase.
## Maximum Speedup by Pipelining

<table>
<thead>
<tr>
<th>Assumptions</th>
<th>Unpipelined $t_C$</th>
<th>Pipelined $t_C$</th>
<th>Speedup</th>
</tr>
</thead>
</table>

February 18, 2014
Maximum Speedup by Pipelining

Assumptions

1. $t_{IM} = t_{DM} = 10,$
   $t_{ALU} = 5,$
   $t_{RF} = t_{RW} = 1$

4-stage pipeline

<table>
<thead>
<tr>
<th>Unpipelined $t_C$</th>
<th>Pipelined $t_C$</th>
<th>Speedup $t_C$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Maximum Speedup by Pipelining

<table>
<thead>
<tr>
<th>Assumptions</th>
<th>Unpipelined $t_C$</th>
<th>Pipelined $t_C$</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. $t_{IM} = t_{DM} = 10$, $t_{ALU} = 5$, $t_{RF} = t_{RW} = 1$</td>
<td>$t_C$</td>
<td>$t_C$</td>
<td>27</td>
</tr>
</tbody>
</table>
# Maximum Speedup by Pipelining

## Assumptions

1. $t_{IM} = t_{DM} = 10$, 
   $t_{ALU} = 5$, 
   $t_{RF} = t_{RW} = 1$

4-stage pipeline

<table>
<thead>
<tr>
<th>Unpipelined</th>
<th>Pipelined</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_C$</td>
<td>$t_C$</td>
<td>10</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

27

10
### Maximum Speedup by Pipelining

<table>
<thead>
<tr>
<th>Assumptions</th>
<th>Unpipelined $t_C$</th>
<th>Pipelined $t_C$</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. $t_{IM} = t_{DM} = 10$, $t_{ALU} = 5$, $t_{RF} = t_{RW} = 1$</td>
<td>27</td>
<td>10</td>
<td>2.7</td>
</tr>
</tbody>
</table>

4-stage pipeline
## Maximum Speedup by Pipelining

### Assumptions

1. \( t_{IM} = t_{DM} = 10, \)
   \( t_{ALU} = 5, \)
   \( t_{RF} = t_{RW} = 1 \)
   4-stage pipeline

2. \( t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5 \)
   4-stage pipeline

### Unpipelined vs. Pipelined Speedup

<table>
<thead>
<tr>
<th></th>
<th>Unpipelined ( t_{C} )</th>
<th>Pipelined ( t_{C} )</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>27</td>
<td>10</td>
<td>2.7</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Maximum Speedup by Pipelining

<table>
<thead>
<tr>
<th>Assumptions</th>
<th>Unpipelined $t_C$</th>
<th>Pipelined $t_C$</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. $t_{IM} = t_{DM} = 10, t_{ALU} = 5, t_{RF} = t_{RW} = 1$</td>
<td>27</td>
<td>10</td>
<td>2.7</td>
</tr>
<tr>
<td>2. $t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$</td>
<td>25</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Maximum Speedup by Pipelining

## Assumptions

1. \( t_{IM} = t_{DM} = 10, \)
   \( t_{ALU} = 5, \)
   \( t_{RF} = t_{RW} = 1 \)
   4-stage pipeline

<table>
<thead>
<tr>
<th>Unpipelined</th>
<th>Pipelined</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_C )</td>
<td>( t_C )</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>10</td>
<td>2.7</td>
</tr>
</tbody>
</table>

2. \( t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5 \)
   4-stage pipeline

<table>
<thead>
<tr>
<th>Unpipelined</th>
<th>Pipelined</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_C )</td>
<td>( t_C )</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>
# Maximum Speedup by Pipelining

<table>
<thead>
<tr>
<th>Assumptions</th>
<th>Unpipelined $t_C$</th>
<th>Pipelined $t_C$</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. $t_{IM} = t_{DM} = 10$, $t_{ALU} = 5$, $t_{RF} = t_{RW} = 1$</td>
<td>27</td>
<td>10</td>
<td>2.7</td>
</tr>
<tr>
<td>2. $t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$</td>
<td>25</td>
<td>10</td>
<td>2.5</td>
</tr>
</tbody>
</table>
## Maximum Speedup by Pipelining

**Assumptions**

1. \( t_{\text{IM}} = t_{\text{DM}} = 10, \)
   \( t_{\text{ALU}} = 5, \)
   \( t_{\text{RF}} = t_{\text{RW}} = 1 \)
   4-stage pipeline

   \[
   27 \quad 10 \quad 2.7
   \]

2. \( t_{\text{IM}} = t_{\text{DM}} = t_{\text{ALU}} = t_{\text{RF}} = t_{\text{RW}} = 5 \)
   4-stage pipeline

   \[
   25 \quad 10 \quad 2.5
   \]

3. \( t_{\text{IM}} = t_{\text{DM}} = t_{\text{ALU}} = t_{\text{RF}} = t_{\text{RW}} = 5 \)
   5-stage pipeline

   \[\text{Speedup} = \frac{t_{\text{C}}}{t_{\text{C}}}\]
# Maximum Speedup by Pipelining

**Assumptions**

1. $t_{IM} = t_{DM} = 10$, 
   $t_{ALU} = 5$, 
   $t_{RF} = t_{RW} = 1$ 
   4-stage pipeline

   Unpipelined $t_C$ | Pipelined $t_C$ | Speedup
   | 27 | 10 | 2.7 |

2. $t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$ 
   4-stage pipeline

   Unpipelined $t_C$ | Pipelined $t_C$ | Speedup
   | 25 | 10 | 2.5 |

3. $t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$ 
   5-stage pipeline

   Unpipelined $t_C$ | Pipelined $t_C$ | Speedup
   | 25 |     |     |
# Maximum Speedup by Pipelining

The table below shows the maximum speedup by pipelining for different assumptions and pipeline stages.

<table>
<thead>
<tr>
<th>Assumptions</th>
<th>Unpipelined $t_C$</th>
<th>Pipelined $t_C$</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_{IM} = t_{DM} = 10$, $t_{ALU} = 5$, $t_{RF} = t_{RW} = 1$ for 4-stage pipeline</td>
<td>27</td>
<td>10</td>
<td>2.7</td>
</tr>
<tr>
<td>$t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$ for 4-stage pipeline</td>
<td>25</td>
<td>10</td>
<td>2.5</td>
</tr>
<tr>
<td>$t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$ for 5-stage pipeline</td>
<td>25</td>
<td>5</td>
<td>-</td>
</tr>
</tbody>
</table>

Assumptions:
- $t_{IM}$: Instruction Memory Access Time
- $t_{DM}$: Data Memory Access Time
- $t_{ALU}$: ALU Operation Time
- $t_{RF}$: Register File Access Time
- $t_{RW}$: Register Write Time
# Maximum Speedup by Pipelining

## Assumptions

1. \( t_{IM} = t_{DM} = 10, \)
   \( t_{ALU} = 5, \)
   \( t_{RF} = t_{RW} = 1 \)
   4-stage pipeline

<table>
<thead>
<tr>
<th>Unpipelined</th>
<th>Pipelined</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{C} )</td>
<td>27</td>
<td>10</td>
</tr>
</tbody>
</table>

2. \( t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5 \)
   4-stage pipeline

<table>
<thead>
<tr>
<th>Unpipelined</th>
<th>Pipelined</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{C} )</td>
<td>25</td>
<td>10</td>
</tr>
</tbody>
</table>

3. \( t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5 \)
   5-stage pipeline

<table>
<thead>
<tr>
<th>Unpipelined</th>
<th>Pipelined</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t_{C} )</td>
<td>25</td>
<td>5</td>
</tr>
</tbody>
</table>
# Maximum Speedup by Pipelining

<table>
<thead>
<tr>
<th>Assumptions</th>
<th>Unpipelined $t_C$</th>
<th>Pipelined $t_C$</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. $t_{IM} = t_{DM} = 10$,</td>
<td>27</td>
<td>10</td>
<td>2.7</td>
</tr>
<tr>
<td>$t_{ALU} = 5$,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$t_{RF} = t_{RW} = 1$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4-stage pipeline</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. $t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$</td>
<td>25</td>
<td>10</td>
<td>2.5</td>
</tr>
<tr>
<td>4-stage pipeline</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. $t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$</td>
<td>25</td>
<td>5</td>
<td>5.0</td>
</tr>
<tr>
<td>5-stage pipeline</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What seems to be the message here?
## Maximum Speedup by Pipelining

**Assumptions**

1. \( t_{IM} = t_{DM} = 10, \)  
   \( t_{ALU} = 5, \)  
   \( t_{RF} = t_{RW} = 1 \)  
   4-stage pipeline

\[
\begin{array}{ccc}
\text{Unpipelined} & \text{Pipelined} & \text{Speedup} \\
\hline
27 & 10 & 2.7 \\
25 & 10 & 2.5 \\
25 & 5 & 5.0 \\
\end{array}
\]

2. \( t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5 \)  
   4-stage pipeline

3. \( t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5 \)  
   5-stage pipeline

**What seems to be the message here?**

*One can achieve higher speedup with more pipeline stages.*
5-Stage Pipelined Execution

Instruction Flow Diagram

I-Fetch (IF)

Decode, Reg. Fetch (ID)

Execute (EX)

Memory (MA)

Write-Back (WB)
5-Stage Pipelined Execution

Instruction Flow Diagram

- **I-Fetch (IF)**: PC → Addr, rdata → Inst. Memory
- **Decode, Reg. Fetch (ID)**: Addr, rdata → IR
- **Execute (EX)**: rs1, rs2, rd1, ws, wd, rd2, GPRs → Imm, Ext, ALU
- **Memory (MA)**: addr, Data, Memory, wdata
- **Write-Back (WB)**: we, rdata, rd1, GPRs

Time:
- t0: I-Fetch
- t1: Decode, Reg. Fetch
- t2: Execute
- t3: Memory
- t4: Write-Back
- t5, t6, t7, ...
5-Stage Pipelined Execution

**Instruction Flow Diagram**

- **I-Fetch (IF)**
- **Decode, Reg. Fetch (ID)**
- **Execute (EX)**
- **Memory (MA)**
- **Write-Back (WB)**

**I-Fetch (IF)**
- PC
- Addr
- Rdata
- Inst. Memory

**Decode, Reg. Fetch (ID)**
- IR
- Decode
- Reg. Fetch
- GPRs
- Imm Ext

**Execute (EX)**
- ALU
- We
- Rs1
- Rs2
- Rd1
- We
- Wd
- Rd2
- GPRs
- Addr
- Rdata
- Inst.
- Memory
- Data
- Wdata

**Memory (MA)**
- We
- Addr
- Data
- Memory
- Wdata

**Write-Back (WB)**
- Write-Back

**Time**
- t0
- t1
- t2
- t3
- t4
- t5
- t6
- t7
- . . . .

**Instruction**
- Instruction1

**Add**
- 0x4
5-Stage Pipelined Execution

Instruction Flow Diagram

I-Fetch (IF)  Decode, Reg. Fetch (ID)  Execute (EX)  Memory (MA)  Write-Back (WB)

<table>
<thead>
<tr>
<th>Time</th>
<th>I-Fetch (IF)</th>
<th>Decode, Reg. Fetch (ID)</th>
<th>Execute (EX)</th>
<th>Memory (MA)</th>
<th>Write-Back (WB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>t0</td>
<td>IF(_1)</td>
<td>ID(_1)</td>
<td>EX(_1)</td>
<td>MA(_1)</td>
<td>WB(_1)</td>
</tr>
<tr>
<td>t1</td>
<td>IF(_2)</td>
<td>ID(_2)</td>
<td>EX(_2)</td>
<td>MA(_2)</td>
<td>WB(_2)</td>
</tr>
<tr>
<td>t2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . .</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Instruction 1
- Addr
- Rdata
- Instruction
- Memory

Instruction 2
- Addr
- Rdata
- Instruction
- Memory
5-Stage Pipelined Execution

Instruction Flow Diagram

- **I-Fetch (IF)**
- **Decode, Reg. Fetch (ID)**
- **Execute (EX)**
- **Memory (MA)**
- **Write-Back (WB)**

**time**
- instruction1
- instruction2
- instruction3

**time**
- $t_0$
- $t_1$
- $t_2$
- $t_3$
- $t_4$
- $t_5$
- $t_6$
- $t_7$

**I-Fetch (IF)**
- $IF_1$
- $IF_2$
- $IF_3$

**Decode, Reg. Fetch (ID)**
- $ID_1$
- $ID_2$
- $ID_3$

**Execute (EX)**
- $EX_1$
- $EX_2$
- $EX_3$

**Memory (MA)**
- $MA_1$
- $MA_2$
- $MA_3$

**Write-Back (WB)**
- $WB_1$
- $WB_2$
- $WB_3$
5-Stage Pipelined Execution

Instruction Flow Diagram

\begin{itemize}
\item \textbf{I-Fetch (IF)}
\item \textbf{Decode, Reg. Fetch (ID)}
\item \textbf{Execute (EX)}
\item \textbf{Memory (MA)}
\item \textbf{Write-Back (WB)}
\end{itemize}

\begin{itemize}
\item \textit{time}
\item \textit{instruction1}
\item \textit{instruction2}
\item \textit{instruction3}
\item \textit{instruction4}
\end{itemize}

\begin{itemize}
\item \textit{t0} \textit{IF}_1
\item \textit{t1} \textit{ID}_1
\item \textit{t2} \textit{EX}_1
\item \textit{t3} \textit{MA}_1
\item \textit{t4} \textit{WB}_1
\item \textit{t5} \textit{IF}_2
\item \textit{t6} \textit{ID}_2
\item \textit{t7} \textit{EX}_2
\item \textit{t8} \textit{MA}_2
\item \textit{t9} \textit{WB}_2
\item \textit{t10} \textit{IF}_3
\item \textit{t11} \textit{ID}_3
\item \textit{t12} \textit{EX}_3
\item \textit{t13} \textit{MA}_3
\item \textit{t14} \textit{WB}_3
\item \textit{t15} \textit{IF}_4
\item \textit{t16} \textit{ID}_4
\item \textit{t17} \textit{EX}_4
\item \textit{t18} \textit{MA}_4
\item \textit{t19} \textit{WB}_4
\end{itemize}
5-Stage Pipelined Execution

Instruction Flow Diagram

I-Fetch (IF)

- Addr
- Rdata
- Inst. Memory

Decode, Reg. Fetch (ID)

- PC
- Addr
- Rdata
- IR

Execute (EX)

- Decode, Reg. Fetch (ID)
- Execute (EX)

Memory (MA)

- Memory (MA)
- Write-Back (WB)

Write-Back (WB)

- Write-Back (WB)

Time

- t0
- t1
- t2
- t3
- t4
- t5
- t6
- t7

Instructions

- Instruction 1
- Instruction 2
- Instruction 3
- Instruction 4
- Instruction 5
5-Stage Pipelined Execution

Instruction Flow Diagram

```
<table>
<thead>
<tr>
<th></th>
<th>t0</th>
<th>t1</th>
<th>t2</th>
<th>t3</th>
<th>t4</th>
<th>t5</th>
<th>t6</th>
<th>t7</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF1</td>
<td>IF2</td>
<td>ID2</td>
<td>EX2</td>
<td>MA2</td>
<td>WB2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IF3</td>
<td>IF4</td>
<td>ID4</td>
<td>EX4</td>
<td>MA4</td>
<td>WB4</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

I-Fetch (IF)   Decode, Reg. Fetch (ID)   Execute (EX)   Memory (MA)   Write-Back (WB)
5-Stage Pipelined Execution

Resource Usage Diagram

Resources

I-Fetch (IF)

Decode, Reg. Fetch (ID)

Execute (EX)

Memory (MA)

Write - Back (WB)

Sanchez & Emer
5-Stage Pipelined Execution

Resource Usage Diagram

- **I-Fetch (IF)**: PC, addr, rdata
- **Decode, Reg. Fetch (ID)**: IR, rs1, rs2
- **Execute (EX)**: ALU, addr, rdata, rdata
- **Memory (MA)**: we, addr, Data, Memory
- **Write-Back (WB)**: we, wdata

Resources:
- PC
- IR
- ALU
- Memory

Time:
- t0, t1, t2, t3, t4, t5, t6, t7, ...

Diagrams and resource usage are visualized in the image.
5-Stage Pipelined Execution
Resource Usage Diagram

I-Fetch (IF)

Decode, Reg. Fetch (ID)

Execute (EX)

Memory (MA)

Write - Back (WB)

Resources

<table>
<thead>
<tr>
<th>Time</th>
<th>IF 1</th>
<th>IF 2</th>
<th>IF 3</th>
<th>IF 4</th>
<th>IF 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>t0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## 5-Stage Pipelined Execution

### Resource Usage Diagram

#### Resources
- **IF**: I1, I2, I3, I4, I5
- **ID**: I1, I2, I3, I4, I5

#### Time Stamps
- t0, t1, t2, t3, t4, t5, t6, t7, ...

#### Stages
- **I-Fetch (IF)**
- **Decode, Reg. Fetch (ID)**
- **Execute (EX)**
- **Memory (MA)**
- **Write-Back (WB)**
5-Stage Pipelined Execution

Resource Usage Diagram

- **I-Fetch (IF)**: 0x4 Add, PC, addr, rdata, IR
- **Decode, Reg. Fetch (ID)**: \( w, rs1, rs2, rd1, ws, wdata, rd2 \), GPRs, Imm Ext
- **Execute (EX)**: ALU, \( we, addr, rdata, wdata \)
- **Memory (MA)**: Memory, Data, \( we, addr, wdata \)
- **Write-Back (WB)**: Write Back

**Resources**
- **time**: t0, t1, t2, t3, t4, t5, t6, t7, ...
- **IF**: \( I_1, I_2, I_3, I_4, I_5 \)
- **ID**: \( I_1, I_2, I_3, I_4, I_5 \)
- **EX**: \( I_1, I_2, I_3, I_4, I_5 \)

**Notes**: This diagram illustrates the flow of instructions through the pipeline stages, highlighting the resource usage and timing. Each stage processes different instructions and data, with connections showing how information is transferred between stages.
5-Stage Pipelined Execution

Resource Usage Diagram

- **I-Fetch (IF)**
- **Decode, Reg. Fetch (ID)**
- **Execute (EX)**
- **Memory (MA)**
- **Write-Back (WB)**

Resources:
- **IF**
- **ID**
- **EX**
- **MA**

Time:
- t0
- t1
- t2
- t3
- t4
- t5
- t6
- t7

Instructions:
- I1
- I2
- I3
- I4
- I5

Operations:
- Addr
- Rdata
- PC
- Inst. Memory
- IF
- EX
- MA
- ALU
- Write-Back

Resources:
- Write-Back (WB)
- I-Fetch (IF)
- Decode, Reg. Fetch (ID)
- Execute (EX)
- Memory (MA)
5-Stage Pipelined Execution

Resource Usage Diagram

- **I-Fetch (IF)**
- **Decode, Reg. Fetch (ID)**
- **Execute (EX)**
- **Memory (MA)**
- **Write-Back (WB)**

**Resources**
- **IF**
- **ID**
- **EX**
- **MA**
- **WB**

**Time**
- $t_0$ I_1
- $t_1$ I_2
- $t_2$ I_3
- $t_3$ I_4
- $t_4$ I_5
- $t_5$ I_6
- $t_6$ I_7
- $t_7$ I_8

**Add**
- PC
- PC
- PC
- PC
- PC

**Resources**
- **Addr rdata**
- **Inst. Memory**
- **Addr rdata**
- **Inst. Memory**
- **Addr rdata**
- **Inst. Memory**
- **Addr rdata**
- **Inst. Memory**

**Write-Back (WB)**
- **Addr rdata**
- **Inst. Memory**
- **Addr rdata**
- **Inst. Memory**
- **Addr rdata**
- **Inst. Memory**
- **Addr rdata**
- **Inst. Memory**

**ALU**
- Imm Ext
- Imm Ext
- Imm Ext
- Imm Ext
- Imm Ext

**Memory**
- Data Memory
- Data Memory
- Data Memory
- Data Memory
- Data Memory

**Resources**
- **addr rdata**
- **Inst. Memory**
- **addr rdata**
- **Inst. Memory**
- **addr rdata**
- **Inst. Memory**
- **addr rdata**
- **Inst. Memory**
5-Stage Pipelined Execution

Resource Usage Diagram

- **I-Fetch (IF)**
- **Decode, Reg. Fetch (ID)**
- **Execute (EX)**
- **Memory (MA)**
- **Write-Back (WB)**

**Resources**
- **IF**: Inst. Memory
- **ID**: I-Fetch
- **EX**: Decode, Reg. Fetch
- **MA**: Execute
- **WB**: Memory

**Time**
- **t0**: I1
- **t1**: I2
- **t2**: I3
- **t3**: I4
- **t4**: I5
- **t5**: I1
- **t6**: I2
- **t7**: I3
- **t8**: I4

**Write-Back (WB)**
- **Addr**: 0x4
- **Data**: Add

**Memory**
- **Addr**: rs1, rs2
- **Data**: rd1, ws, wdata
Pipelined Execution: ALU Instructions

Diagram showing the pipeline stages for ALU instructions, including memory access, instruction fetch, and ALU operation.
Pipelined Execution: ALU Instructions

Not quite correct!
Pipelined Execution: ALU Instructions

Not quite correct!

We need an Instruction Reg (IR) for each stage
Pipelined Execution: ALU Instructions

Not quite correct!

We need an Instruction Reg (IR) for each stage
Pipelined Execution: ALU Instructions

Not quite correct!

We need an Instruction Reg (IR) for each stage
Not quite correct!

We need an Instruction Reg (IR) for each stage
Pipelined MIPS Datapath
without jumps

What else is needed?
Pipelined MIPS Datapath

without jumps

What else is needed?

Control Points Need to Be Connected
Pipelined MIPS Datapath

without jumps

What else is needed?

Control Points Need to Be Connected
Pipelined MIPS Datapath without jumps

What else is needed?

Control Points Need to Be Connected
Pipelined MIPS Datapath

without jumps

Control Points Need to Be Connected

What else is needed?

Sanchez & Emer February 18, 2014
Pipelined MIPS Datapath
without jumps

What else is needed?

Control Points Need to Be Connected
Pipelined MIPS Datapath

without jumps

What else is needed?

Control Points Need to Be Connected
How instructions can interact with each other in a pipeline
How instructions can interact with each other in a pipeline

- An instruction in the pipeline may need a resource being used by another instruction in the pipeline → structural hazard
How instructions can interact with each other in a pipeline

- An instruction in the pipeline may need a resource being used by another instruction in the pipeline → **structural hazard**

- An instruction may depend on something produced by an earlier instruction
  - Dependence may be for a data calculation → **data hazard**
  - Dependence may be for calculating the next PC → **control hazard (branches, interrupts)**
Data Hazards

... 
r1 ← r0 + 10 
r4 ← r1 + 17 
...
Data Hazards

... r1 ← r0 + 10
r4 ← r1 + 17
...

February 18, 2014
Data Hazards

\[ r1 \leftarrow r0 + 10 \]
\[ r4 \leftarrow r1 + 17 \]
Data Hazards

... 
r1 ← r0 + 10 
r4 ← r1 + 17 
...

r1 is stale.Oops!
Resolving Data Hazards

Strategy 1: *Wait for the result to be available by freezing earlier pipeline stages* → **interlocks**

Strategy 2: *Route data as soon as possible after it is calculated to the earlier pipeline stage* → **bypass**

Strategy 3: *Speculate on the dependence*

  Two cases:
Resolving Data Hazards

Strategy 1: *Wait for the result to be available by freezing earlier pipeline stages* → \textit{interlocks}

Strategy 2: *Route data as soon as possible after it is calculated to the earlier pipeline stage* → \textit{bypass}

Strategy 3: *Speculate on the dependence*

\textit{Two cases:}

\begin{itemize}
  \item \textit{Guessed correctly} → do nothing
\end{itemize}
Resolving Data Hazards

Strategy 1: Wait for the result to be available by freezing earlier pipeline stages → interlocks

Strategy 2: Route data as soon as possible after it is calculated to the earlier pipeline stage → bypass

Strategy 3: Speculate on the dependence
Two cases:
   - Guessed correctly → do nothing
   - Guessed incorrectly → kill and restart
Resolving Data Hazards (1)

Strategy 1:

Wait for the result to be available by freezing earlier pipeline stages → interlocks
Feedback to Resolve Hazards

- Later stages provide dependence information to earlier stages which can stall (or kill) instructions.
Feedback to Resolve Hazards

- Later stages provide dependence information to earlier stages which can *stall (or kill) instructions*
Feedback to Resolve Hazards

- Later stages provide dependence information to earlier stages which can stall (or kill) instructions.
Feedback to Resolve Hazards

- Later stages provide dependence information to earlier stages which can *stall (or kill) instructions*
Feedback to Resolve Hazards

- Later stages provide dependence information to earlier stages which can *stall (or kill) instructions*
Feedback to Resolve Hazards

Later stages provide dependence information to earlier stages which can stall (or kill) instructions.
Feedback to Resolve Hazards

Later stages provide dependence information to earlier stages which can stall (or kill) instructions.
Feedback to Resolve Hazards

- Later stages provide dependence information to earlier stages which can **stall (or kill) instructions**
Feedback to Resolve Hazards

- Later stages provide dependence information to earlier stages which can stall (or kill) instructions.

- Controlling a pipeline in this manner works provided the instruction at stage \( i+1 \) can complete without any interference from instructions in stages 1 to \( i \) (otherwise deadlocks may occur).
Interlocks to resolve Data Hazards

... 
\[ r1 \leftarrow r0 + 10 \]
\[ r4 \leftarrow r1 + 17 \]
...
Interlocks to resolve Data Hazards

... r1 ← r0 + 10
r4 ← r1 + 17
...

February 18, 2014

Sanchez & Emer
Interlocks to resolve Data Hazards

... 
r1 ← r0 + 10
r4 ← r1 + 17
...

February 18, 2014
Interlocks to resolve Data Hazards

... r1 ← r0 + 10
r4 ← r1 + 17
...

Stall Condition
Stalled Stages and Pipeline Bubbles
Stalled Stages and Pipeline Bubbles

\[ \text{time} \]
\[ t_0 \quad t_1 \quad t_2 \quad t_3 \quad t_4 \quad t_5 \quad t_6 \quad t_7 \quad \ldots \]
Stalled Stages and Pipeline Bubbles

\begin{equation}
(I_1) \ r_1 \leftarrow (r_0) + 10 \ IF_1 \ ID_1 \ EX_1 \ MA_1 \ WB_1
\end{equation}
# Stalled Stages and Pipeline Bubbles

<table>
<thead>
<tr>
<th>time</th>
<th>t0</th>
<th>t1</th>
<th>t2</th>
<th>t3</th>
<th>t4</th>
<th>t5</th>
<th>t6</th>
<th>t7</th>
<th>. . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>(I₁) r₁ ← (r₀) + 10</td>
<td>IF₁</td>
<td>ID₁</td>
<td>EX₁</td>
<td>MA₁</td>
<td>WB₁</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(I₂) r₄ ← (r₁) + 17</td>
<td>IF₂</td>
<td>ID₂</td>
<td>ID₂</td>
<td>ID₂</td>
<td>ID₂</td>
<td>EX₂</td>
<td>MA₂</td>
<td>WB₂</td>
<td></td>
</tr>
</tbody>
</table>
Stalled Stages and Pipeline Bubbles

\( \text{time} \)
\[
\begin{array}{cccccccc}
  t0 & t1 & t2 & t3 & t4 & t5 & t6 & t7 \\
\end{array}
\]

(I_1) \ r1 \leftarrow (r0) + 10 \quad \text{IF}_1 \quad \text{ID}_1 \quad \text{EX}_1 \quad \text{MA}_1 \quad \text{WB}_1

(I_2) \ r4 \leftarrow (r1) + 17 \quad \text{IF}_2 \quad \text{ID}_2 \quad \text{ID}_2 \quad \text{ID}_2 \quad \text{EX}_2 \quad \text{MA}_2 \quad \text{WB}_2

(I_3) \quad \text{IF}_3 \quad \text{IF}_3 \quad \text{IF}_3 \quad \text{IF}_3 \quad \text{ID}_3 \quad \text{EX}_3 \quad \text{MA}_3 \quad \text{WB}_3
Stalled Stages and Pipeline Bubbles

<table>
<thead>
<tr>
<th>time</th>
<th>t0</th>
<th>t1</th>
<th>t2</th>
<th>t3</th>
<th>t4</th>
<th>t5</th>
<th>t6</th>
<th>t7</th>
<th>....</th>
</tr>
</thead>
<tbody>
<tr>
<td>(I_1) r1 (\leftarrow) (r0) + 10</td>
<td>IF_1</td>
<td>ID_1</td>
<td>EX_1</td>
<td>MA_1</td>
<td>WB_1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(I_2) r4 (\leftarrow) (r1) + 17</td>
<td>IF_2</td>
<td>ID_2</td>
<td>ID_2</td>
<td>ID_2</td>
<td>ID_2</td>
<td>EX_2</td>
<td>MA_2</td>
<td>WB_2</td>
<td></td>
</tr>
<tr>
<td>(I_3)</td>
<td>IF_3</td>
<td>IF_3</td>
<td>IF_3</td>
<td>IF_3</td>
<td>IF_3</td>
<td>ID_3</td>
<td>EX_3</td>
<td>MA_3</td>
<td>WB_3</td>
</tr>
<tr>
<td>(I_4)</td>
<td>IF_4</td>
<td>ID_4</td>
<td>EX_4</td>
<td>MA_4</td>
<td>WB_4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Stalled Stages and Pipeline Bubbles

\[
\begin{align*}
&\text{time} & t0 & t1 & t2 & t3 & t4 & t5 & t6 & t7 & \ldots \\
& (I_1) \ r1 \leftarrow (r0) + 10 & \text{IF}_1 & \text{ID}_1 & \text{EX}_1 & \text{MA}_1 & \text{WB}_1 & \\
& (I_2) \ r4 \leftarrow (r1) + 17 & \text{IF}_2 & \text{ID}_2 & \text{ID}_2 & \text{ID}_2 & \text{EX}_2 & \text{MA}_2 & \text{WB}_2 & \\
& (I_3) & \text{IF}_3 & \text{ID}_3 & \text{IF}_3 & \text{ID}_3 & \text{EX}_3 & \text{MA}_3 & \text{WB}_3 & \\
& (I_4) & & \text{IF}_4 & \text{ID}_4 & \text{EX}_4 & \text{MA}_4 & \text{WB}_4 & \\
& (I_5) & & & \text{IF}_5 & \text{ID}_5 & \text{EX}_5 & \text{MA}_5 & \text{WB}_5 & \\
\end{align*}
\]
Stalled Stages and Pipeline Bubbles

\[ \begin{align*}
  (I_1) & \quad r_1 \leftarrow (r_0) + 10 \\
  (I_2) & \quad r_4 \leftarrow (r_1) + 17 \\
  (I_3) & \\
  (I_4) & \\
  (I_5) & \\
\end{align*} \]
Stalled Stages and Pipeline Bubbles

$t0 \quad t1 \quad t2 \quad t3 \quad t4 \quad t5 \quad t6 \quad t7 \quad \ldots$

(I$_1$) $r1 \leftarrow (r0) + 10$

(I$_2$) $r4 \leftarrow (r1) + 17$

(I$_3$)

(I$_4$)

(I$_5$)

$stalled \ stages$
Stalled Stages and Pipeline Bubbles

\[ (I_1) \ r_1 \leftarrow (r_0) + 10 \quad \text{IF}_1 \]
\[ (I_2) \ r_4 \leftarrow (r_1) + 17 \quad \text{IF}_2 \]
\[ (I_3) \]
\[ (I_4) \]
\[ (I_5) \]

Resource Usage
Stalled Stages and Pipeline Bubbles

\[ (I_1) r_1 \leftarrow (r_0) + 10 \]
\[ (I_2) r_4 \leftarrow (r_1) + 17 \]

Resource Usage
Stalled Stages and Pipeline Bubbles

\[ (I_1) \text{ r}1 \leftarrow (r0) + 10 \]
\[ (I_2) \text{ r}4 \leftarrow (r1) + 17 \]
\[ (I_3) \]
\[ (I_4) \]
\[ (I_5) \]

Resource Usage
Stalled Stages and Pipeline Bubbles

\( (I_1) \ r_1 \leftarrow (r_0) + 10 \)
\( (I_2) \ r_4 \leftarrow (r_1) + 17 \)
\( (I_3) \)
\( (I_4) \)
\( (I_5) \)

Resource Usage

| Time | t0 | t1 | t2 | t3 | t4 | t5 | t6 | t7 | ...
|------|----|----|----|----|----|----|----|----|------
| IF   | \( I_1 \) | \( I_2 \) | \( I_3 \) | \( I_3 \) | \( I_3 \) | \( I_3 \) | \( I_4 \) | \( I_5 \) | ...
| ID   | \( I_1 \) | \( I_2 \) | \( I_2 \) | \( I_2 \) | \( I_2 \) | \( I_3 \) | \( I_4 \) | \( I_5 \) |
Stalled Stages and Pipeline Bubbles

\begin{align*}
\text{time} & \quad t_0 \quad t_1 \quad t_2 \quad t_3 \quad t_4 \quad t_5 \quad t_6 \quad t_7 \quad \ldots \\
(I_1) & \quad r_1 \leftarrow (r_0) + 10 \quad \text{IF}_1 \quad \text{ID}_1 \quad \text{EX}_1 \quad \text{MA}_1 \quad \text{WB}_1 \\
(I_2) & \quad r_4 \leftarrow (r_1) + 17 \quad \text{IF}_2 \quad \text{ID}_2 \quad \text{ID}_2 \quad \text{ID}_2 \quad \text{ID}_2 \quad \text{EX}_2 \quad \text{MA}_2 \quad \text{WB}_2 \\
(I_3) & \quad \text{IF}_3 \quad \text{ID}_3 \quad \text{ID}_3 \quad \text{ID}_3 \quad \text{ID}_3 \quad \text{IF}_3 \quad \text{EX}_3 \quad \text{MA}_3 \quad \text{WB}_3 \\
(I_4) & \quad \text{IF}_4 \quad \text{ID}_4 \quad \text{ID}_4 \quad \text{ID}_4 \quad \text{ID}_4 \quad \text{ID}_4 \quad \text{EX}_4 \quad \text{MA}_4 \quad \text{WB}_4 \\
(I_5) & \quad \text{IF}_5 \quad \text{ID}_5 \quad \text{ID}_5 \quad \text{ID}_5 \quad \text{ID}_5 \quad \text{ID}_5 \quad \text{EX}_5 \quad \text{MA}_5 \quad \text{WB}_5
\end{align*}

\begin{align*}
\text{Resource Usage} \\
\text{IF} & \quad I_1 \quad I_2 \quad I_3 \quad I_3 \quad I_3 \quad I_4 \quad I_5 \quad \ldots \\
\text{ID} & \quad I_1 \quad I_2 \quad I_2 \quad I_2 \quad I_2 \quad I_3 \quad I_4 \quad I_5 \quad \ldots \\
\text{EX} & \quad I_1 \quad \text{nop} \quad \text{nop} \quad \text{nop} \quad I_2 \quad I_3 \quad I_4 \quad I_5
\end{align*}
### Stalled Stages and Pipeline Bubbles

**time**

| time  | t0 | t1   | t2 | t3   | t4 | t5   | t6 | t7 | .....
|-------|----|------|----|------|----|------|----|----|-------
| (I₁)  | r₁ | (r₀) | +10|      |    |      |    |    |       
| (I₂)  | r₄ |      |    |      |    |      |    |    |       
| (I₃)  |    |      |    |      |    |      |    |    |       
| (I₄)  |    |      |    |      |    |      |    |    |       
| (I₅)  |    |      |    |      |    |      |    |    |       

**Resource Usage**

- **IF**: I₁, I₂, I₃, I₄, I₅
- **ID**: I₁, I₂, I₃, I₄, I₅
- **EX**: I₁, I₁, I₁, I₁
- **MA**: I₁, I₁, I₁, I₁

**Stalled Stages**

- IF₁
- ID₁
- EX₁
- MA₁
- WB₁

- IF₂
- ID₂
- ID₂
- ID₂
- ID₂

- IF₃
- ID₃
- ID₃
- ID₃
- ID₃

- IF₄
- ID₄
- ID₄
- ID₄
- ID₄

- IF₅
- ID₅
- ID₅
- ID₅
- ID₅

- EX₂
- MA₂
- WB₂

- EX₃
- MA₃
- WB₃

- EX₄
- MA₄
- WB₄

- EX₅
- MA₅
- WB₅
Stalled Stages and Pipeline Bubbles

\[
\begin{array}{cccccccc}
\text{time} & t0 & t1 & t2 & t3 & t4 & t5 & t6 & t7 \\
(I_1) & r1 & \leftarrow & (r0) + 10 & \uparrow & \downarrow & \downarrow & \downarrow & \downarrow \\
(I_2) & r4 & \leftarrow & (r1) + 17 & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
(I_3) & & & & & & & & \\
(I_4) & & & & & & & & \\
(I_5) & & & & & & & & \\
\end{array}
\]

\[
\begin{array}{cccccccc}
\text{IF} & I_1 & I_2 & I_3 & I_3 & I_3 & I_4 & I_5 \\
\text{ID} & I_1 & I_2 & I_2 & I_2 & I_3 & I_4 & I_5 \\
\text{EX} & I_1 & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} \\
\text{MA} & I_1 & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} \\
\text{WB} & I_1 & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} \\
\end{array}
\]

Resource Usage
Stalled Stages and Pipeline Bubbles

\[ (I_1) \ r_1 \leftarrow (r_0) + 10 \]
\[ (I_2) \ r_4 \leftarrow (r_1) + 17 \]
\[ (I_3) \]
\[ (I_4) \]
\[ (I_5) \]

Resource Usage

<table>
<thead>
<tr>
<th>IF</th>
<th>ID</th>
<th>EX</th>
<th>MA</th>
<th>WB</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_1</td>
<td>I_1</td>
<td>I_1</td>
<td>I_1</td>
<td>I_1</td>
</tr>
</tbody>
</table>

Stalled stages

\[ \text{time} \]
\[ t0 \ t1 \ t2 \ t3 \ t4 \ t5 \ t6 \ t7 \ldots \]
\[ (I_1) \ r_1 \leftarrow (r_0) + 10 \]
\[ (I_2) \ r_4 \leftarrow (r_1) + 17 \]

Stalled stages

\[ \text{time} \]
\[ t0 \ t1 \ t2 \ t3 \ t4 \ t5 \ t6 \ t7 \ldots \]
\[ (I_1) \ r_1 \leftarrow (r_0) + 10 \]
\[ (I_2) \ r_4 \leftarrow (r_1) + 17 \]
Stalled Stages and Pipeline Bubbles

(time
t0 t1 t2 t3 t4 t5 t6 t7 . . .
(I₁) r₁ \leftarrow (r₀) + 10
(IF₁)
(ID₁)
(EX₁)
(MA₁)
(WB₁)
(stalled stages)

(IF₂)
(ID₂)
(ID₂)
(IF₃)

(IF₃)

(IF₄)

(IF₅)

(ID₄)

(ID₅)

(EX₄)

(MA₄)

(WB₄)

(IF₆)

(ID₆)

(EX₅)

(MA₅)

(WB₅)

Resource Usage

IF I₁ I₂ I₃ I₄ I₅ . . .
ID I₁ I₂ I₃ I₄ I₅ . . .
EX I₁ nop nop I₂ I₃ I₄ I₅ . . .
MA I₁ nop nop I₂ I₃ I₄ I₅ . . .
WB I₁ nop nop I₂ I₃ I₄ I₅ . . .

nop \Rightarrow \text{pipeline bubble}
Interlock Control Logic

Compare the source registers of the instruction in the decode stage with the destination register of the uncommitted instructions.
Interlock Control Logic

Compare the source registers of the instruction in the decode stage with the destination register of the uncommitted instructions.
Interlock Control Logic

Compare the source registers of the instruction in the decode stage with the destination register of the uncommitted instructions.
Compare the source registers of the instruction in the decode stage with the destination register of the uncommitted instructions.
Interlock Control Logic

Compare the source registers of the instruction in the decode stage with the destination register of the uncommitted instructions.