Instruction Pipelining

Ryan Lee
Computer Science and Artificial Intelligence Laboratory
M.I.T.
Announcements
Announcements

- Lab 1 released later today
  - Designing 3 different cache models using Pin
  - Due Oct. 1
Announcements

• Lab 1 released later today
  – Designing 3 different cache models using Pin
  – Due Oct. 1

• Please view the Pin tutorial video posted on the course website (under Recitation tab)
Announcements

- Lab 1 released later today
  - Designing 3 different cache models using Pin
  - Due Oct. 1

- Please view the Pin tutorial video posted on the course website (under Recitation tab)

- Contact me if you cannot get access to lab machines
Reminder: Harvard-Style Single-Cycle Datapath for MIPS

PCSrc  br  rind  jabs  pc+4

0x4  Add

addr  inst  Inst. Memory

clk

RegWrite

MemWrite  WBSrc

Add

clk

we  rs1  rs2  rd1  ws  wd  rd2  GPRs

Imm  Ext

ALU

z

ALU Control

we  addr  Data Memory wdata

clk

OpCode  RegDst  ExtSel  OpSel  BSrc  zero?
Reminder: Princeton Microarchitecture
Datapath & Control for 2 cycles-per-instruction

March 4, 2021
The same (mux not shown)

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

Overlapped execution

fetch phase

execute phase
Can we overlap instruction fetch and execute?
Can we overlap instruction fetch and execute?

Yes, unless IR contains a Load or Store
Princeton Microarchitecture

Overlapped execution

Can we overlap instruction fetch and execute?

Yes, unless IR contains a Load or Store

Which action should be prioritized?
Can we overlap instruction fetch and execute?

Yes, unless IR contains a Load or Store

Which action should be prioritized? Execute
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
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

fetch phase

execute phase
When stall condition is indicated

- don’t fetch a new instruction and don’t change the PC
Stalling the instruction fetch

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
- insert a nop in the IR
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)
Stalling the instruction fetch

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

When IR contains a jump or taken branch
- no "structural conflict" for the memory
When IR contains a jump or taken branch

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

- 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 taken branch

• 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
Need to stall on branches
Princeton Microarchitecture

When IR contains a jump or taken branch

- *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*
When IR contains a jump or taken branch

- 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
Pipelined Princeton Microarchitecture

March 4, 2021

MIT 6.823 Spring 2021

L06-9
## 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></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=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>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>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; 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></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></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></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALUiu</td>
<td></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></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=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></td>
<td>br</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>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>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; 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>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>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></td>
<td></td>
<td></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></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LW</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>
<td></td>
</tr>
<tr>
<td>SW</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>
<td></td>
</tr>
<tr>
<td>BEQZ_{z=1}</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>
<td></td>
</tr>
<tr>
<td>BEQZ_{z=0}</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>
<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>ALUi u</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=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>br</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>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; 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></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></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></td>
</tr>
<tr>
<td>LW</td>
<td></td>
<td>sE₁₆</td>
<td>Imm</td>
<td></td>
<td>+</td>
<td>no</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>+</td>
<td>yes</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</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>br</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>pc+4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></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>*</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>*</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>*</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>
<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>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></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_{z=1}</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_{z=0}</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></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>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>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_{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>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></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>br</td>
<td></td>
<td></td>
<td></td>
<td></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>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  ;  WB Src = ALU / Mem / PC; IR Src = nop/mem; MAaddr Src = pc/ALU
Reg Dst = 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>BSrc</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></td>
<td>pc</td>
<td></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></td>
<td>pc</td>
<td></td>
</tr>
<tr>
<td>BEQZ²=1</td>
<td>sE₁₆</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>BEQZ²=0</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>no</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>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td></td>
<td>jabs</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>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>no</td>
<td>yes</td>
<td>PC</td>
<td>R31</td>
<td></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_{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>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_{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></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></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>br</td>
<td></td>
<td></td>
<td></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>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>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=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>*</td>
<td>br</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>*</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>*</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>
<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>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>yes</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>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>SW</td>
<td>yes</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>pc</td>
<td>nop</td>
<td>ALU</td>
</tr>
<tr>
<td>BEQZ\textsubscript{z=1}</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>
<td></td>
</tr>
<tr>
<td>BEQZ\textsubscript{z=0}</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>
<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>*</td>
<td>no</td>
<td>no</td>
<td>*</td>
<td>*</td>
<td>pc+4</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_{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>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_{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=1}</td>
<td>sE_{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>
<td></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>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></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>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_{z=1}</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_{z=0}</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></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>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>R31</td>
</tr>
<tr>
<td>ALUi</td>
<td>no</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>
<td></td>
</tr>
<tr>
<td>ALUiu</td>
<td>no</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>
<td></td>
</tr>
<tr>
<td>LW</td>
<td>yes</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>
<td></td>
</tr>
<tr>
<td>SW</td>
<td>yes</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>
<td></td>
</tr>
<tr>
<td>BEQZz=1</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>BEQZz=0</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>
<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>*</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
Reg Dst = 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=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>*</td>
<td>br</td>
<td></td>
<td></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>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>*</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>
<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>*</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>
<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; 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>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=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>br</td>
<td>npc</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>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>*</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
<table>
<thead>
<tr>
<th>Opcode</th>
<th>Stall</th>
<th>ExtSel</th>
<th>BSrc</th>
<th>OpSel</th>
<th>MemW</th>
<th>RegW</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>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_{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=1}</td>
<td>yes</td>
<td>sE_{16}</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_{z=0}</td>
<td>yes</td>
<td>sE_{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>
<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;  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>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=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>br</td>
<td>npc</td>
<td>nop</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>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>*</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; 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>ExtSel</th>
<th>BSrc</th>
<th>OpSel</th>
<th>MemW</th>
<th>RegW</th>
<th>WBSrc</th>
<th>RegDst</th>
<th>PCSrc1</th>
<th>PCSrc2</th>
<th>IRSrc</th>
<th>MAddrSrc</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=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>br</td>
<td>npc</td>
<td>nop</td>
<td>*</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>pc+4</td>
<td>pc</td>
<td>nop</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; MAddSrc = pc/ALU
RegDst = rt / rd / R31; PCSrc1 = pc+4 / br / rind / jabs; PC Src2 = 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=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>br</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;_z=0&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>
<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>*</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>
<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>*</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>
<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>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=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>br</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&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; 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&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=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>br</td>
<td>npc</td>
<td>nop</td>
<td>*</td>
</tr>
<tr>
<td>BEQZ&lt;sub&gt;z=0&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; MAddrSrc = 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

- **PC**
- **Addr**
- **Rdata**
- **Inst. Memory**
- **IR**
- **Add**
- **0x4**

**Memory**
- **Addr**
- **Rdata**

**ALU**
- **Rs1**
- **Rs2**
- **Rd1**
- **Ws**
- **Wd**
- **Rd2**
- **GPRs**

**Data Memory**
- **Addr**
- **Rdata**
- **Wdata**

**Imm Ext**
Pipelined Datapath

fetch phase

decode & Reg-fetch phase

execute phase

memory phase

write-back phase

PC

addr rdata

Inst. Memory

IR

ALU

addr rdata

GPRs

rs1 rs2

rd1 ws wd rd2

Imm Ext

Data Memory

we wdata

write
Pipelined Datapath

Clock period can be reduced by dividing the execution of an instruction into multiple cycles

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

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

Suppose memory is significantly slower than other stages. For example, suppose

\[
\begin{align*}
t_{IM} & = 10 \text{ units} \\
t_{DM} & = 10 \text{ units} \\
t_{ALU} & = 5 \text{ units} \\
t_{RF} & = 1 \text{ unit} \\
t_{RW} & = 1 \text{ unit}
\end{align*}
\]

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

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

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

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

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

\[ \Rightarrow \text{increase the critical path by 10\%} \]

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 $t_C$</th>
</tr>
</thead>
</table>

March 4, 2021
# 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>Assumptions</th>
<th>Unpipelined $t_C$</th>
<th>Pipelined $t_C$</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unpipelined</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pipelined</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Maximum Speedup by Pipelining

## Assumptions

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

   4-stage pipeline

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

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>27</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></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>
</tbody>
</table>
Maximum Speedup by Pipelining

<table>
<thead>
<tr>
<th>Assumptions</th>
<th>Unpipelined</th>
<th>Pipelined</th>
<th>Speedup</th>
</tr>
</thead>
</table>
| 1. \( t_{\text{IM}} = t_{\text{DM}} = 10, \)  
\( t_{\text{ALU}} = 5, \)  
\( t_{\text{RF}} = t_{\text{RW}} = 1 \)  
4-stage pipeline | 27          | 10        | 2.7     |
| 2. \( t_{\text{IM}} = t_{\text{DM}} = t_{\text{ALU}} = t_{\text{RF}} = t_{\text{RW}} = 5 \)  
4-stage pipeline |             |           |         |
## Maximum Speedup by Pipelining

### Assumptions

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

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

## Assumptions

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

### Assumptions

1. \( t_{IM} = t_{DM} = 10, \quad t_{ALU} = 5, \quad t_{RF} = t_{RW} = 1 \)
   - 4-stage pipeline
   - Unpipelined: \( t_C = 27 \)
   - Pipelined: \( t_C = 10 \)
   - Speedup: 2.7

2. \( t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5 \)
   - 4-stage pipeline
   - Unpipelined: \( t_C = 25 \)
   - Pipelined: \( t_C = 10 \)
   - Speedup: 2.5
# 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 10</td>
<td>2.7</td>
<td></td>
</tr>
<tr>
<td>2. (t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5)</td>
<td>25 10</td>
<td>2.5</td>
<td></td>
</tr>
<tr>
<td>3. (t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5)</td>
<td>25 10</td>
<td>2.5</td>
<td></td>
</tr>
</tbody>
</table>

Notes:
- \(t_{IM}\) = Instruction Memory Time
- \(t_{DM}\) = Data Memory Time
- \(t_{ALU}\) = ALU Time
- \(t_{RF}\) = Register File Time
- \(t_{RW}\) = Register Write Time
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>
<tr>
<td>3. $t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$</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 ( t_C )</th>
<th>Pipelined ( t_C )</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<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 ( t_C )</th>
<th>Pipelined ( t_C )</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>10</td>
<td>2.5</td>
</tr>
</tbody>
</table>

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

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

## Assumptions

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

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

## Assumptions

<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>$t_{IM} = t_{DM} = 10$, $t_{ALU} = 5$, $t_{RF} = t_{RW} = 1$</td>
<td>27</td>
<td>10</td>
</tr>
<tr>
<td>2.</td>
<td>$t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$</td>
<td>25</td>
<td>10</td>
</tr>
<tr>
<td>3.</td>
<td>$t_{IM} = t_{DM} = t_{ALU} = t_{RF} = t_{RW} = 5$</td>
<td>25</td>
<td>5</td>
</tr>
</tbody>
</table>

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)

Fetch (ID)

Execute (EX)

Memory (MA)

Write - Back (WB)

PC

Inst. Memory

IR

ALU

GPRs

Imm Ext

Add

addr
rdata

rs1
rs2

rd1
ws
wd
rd2

we

rdata

addr

Memory

Addr

wdata

Write - Back (WB)
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)**
  - time
  - instruction

- **Decode, Reg. Fetch (ID)**
  - t0
  - IF

- **Execute (EX)**
  - t1
  - ID
  - t2
  - EX

- **Memory (MA)**
  - t3
  - t4
  - MA
  - t5
  - WB

- **Write - Back (WB)**
  - t6
  - t7
  - . . .
5-Stage Pipelined Execution

Instruction Flow Diagram

$t_{0-7}$

- $t_0$: Fetch (IF) -> pc $\rightarrow$ memory
- $t_1$: Decode, Register Fetch (ID) -> $\rightarrow$ IR, rs1, rs2
- $t_2$: Execute (EX) -> $\rightarrow$ ALU, addr, rdata, we
- $t_3$: Memory (MA) -> $\rightarrow$ rd1, GPRs, Memory
- $t_4$: Write Back (WB) -> $\rightarrow$ wdata

$I$-Fetch (IF)

Decode, Reg. Fetch (ID)

Execute (EX)

Memory (MA)

Write Back (WB)
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>Instruction 1</th>
<th>Instruction 2</th>
<th>Instruction 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>t0</td>
<td>IF₁</td>
<td>ID₁</td>
<td>EX₁</td>
</tr>
<tr>
<td>t1</td>
<td>IF₂</td>
<td>ID₂</td>
<td>EX₂</td>
</tr>
<tr>
<td>t2</td>
<td>IF₃</td>
<td>ID₃</td>
<td>EX₃</td>
</tr>
<tr>
<td>t3</td>
<td>MA₁</td>
<td>WB₁</td>
<td></td>
</tr>
<tr>
<td>t4</td>
<td>MA₂</td>
<td>WB₂</td>
<td></td>
</tr>
<tr>
<td>t5</td>
<td>MA₃</td>
<td>WB₃</td>
<td></td>
</tr>
<tr>
<td>t6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>. . .</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
5-Stage Pipelined Execution

Instruction Flow Diagram

- **I-Fetch (IF)**
  - Fetch
  - Decode
  - Fetch

- **Decode, Reg. Fetch (ID)**
  - Decode
  - Register Fetch

- **Execute (EX)**
  - ALU
  - Execute

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

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

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

- **Instructions**
  - instruction1
  - instruction2
  - instruction3
  - instruction4

- **Register Values**
  - rs1
  - rs2
  - rd1
  - ws
  - wd
  - rd2

- **Memory Access**
  - addr
  - wdata
  - rdata

- **Data Memory**
  - Memory
  - addr
  - wdata

- **IR**
  - Instruction Register

- **PC**
  - Program Counter

- **Adder**
  - 0x4

- **Immediate Extension**
  - Ext

- **ALU**
  - ALU

- **Write-Back (WB)**
  - write-back
5-Stage Pipelined Execution

Instruction Flow Diagram

I-Fetch (IF)

Decode, Reg. Fetch (ID)

Execute (EX)

Memory (MA)

Write-Back (WB)

\[ \begin{align*}
\text{time} &: t0, t1, t2, t3, t4, t5, t6, t7, \ldots \\
\text{instruction1} &: \text{IF}_1, \text{ID}_1, \text{EX}_1, \text{MA}_1, \text{WB}_1 \\
\text{instruction2} &: \text{IF}_2, \text{ID}_2, \text{EX}_2, \text{MA}_2, \text{WB}_2 \\
\text{instruction3} &: \text{IF}_3, \text{ID}_3, \text{EX}_3, \text{MA}_3, \text{WB}_3 \\
\text{instruction4} &: \text{IF}_4, \text{ID}_4, \text{EX}_4, \text{MA}_4, \text{WB}_4 \\
\text{instruction5} &: \text{IF}_5, \text{ID}_5, \text{EX}_5, \text{MA}_5, \text{WB}_5 \\
\end{align*} \]
5-Stage Pipelined Execution

Instruction Flow Diagram

I-Fetch (IF)

Decode, Reg. Fetch (ID)

Execute (EX)

Memory (MA)

Write-Back (WB)

Time:
- $t_0$: Instruction 1
- $t_1$: Instruction 2
- $t_2$: Instruction 3
- $t_3$: Instruction 4
- $t_4$: Instruction 5
- $t_5$ to $t_7$: ...
5-Stage Pipelined Execution

Resource Usage Diagram

---

I-Fetch (IF)

Decode, Reg. Fetch (ID)

Execute (EX)

Memory (MA)

Write-Back (WB)

---

Resources
5-Stage Pipelined Execution

Resource Usage Diagram

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

Resources

- PC
- Add
- Addr
- Rdata
- IR
- Inst. Memory
- Imm Ext
- ALU
- GPRs
- Memory
- Write-Back

Time:
- t0
- t1
- t2
- t3
- t4
- t5
- t6
- t7
- . . .
5-Stage Pipelined Execution

Resource Usage Diagram
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>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>IF</td>
<td>I1</td>
<td>I2</td>
<td>I3</td>
<td>I4</td>
<td>I5</td>
<td>I6</td>
<td>I7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>I1</td>
<td>I2</td>
<td>I3</td>
<td>I4</td>
<td>I5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
5-Stage Pipelined Execution

Resource Usage Diagram

- **IF (I-Fetch)**: PC → Inst. Memory → IR
- **ID (Decode, Reg. Fetch)**: IR → Decode, Reg. Fetch → ALU
- **EX (Execute)**: ALU → Execution Unit
- **MA (Memory)**: Memory → Memory
- **WB (Write-Back)**: Memory → Write-Back

**Resources**

- **Time**: t0, t1, t2, t3, t4, t5, t6, t7, ...
- **IF**: I1, I2, I3, I4, I5
- **ID**: I1, I2, I3, I4, I5
- **EX**: I1, I2, I3, I4, I5

**Example Instruction**

- **Add 0x4**: addr, inst. Memory, IR, Decode, Reg. Fetch, Execute, Memory, Write-Back

**Symbols**

- **Addr**: Address
- **Rdata**: Data
- **Inst. Memory**: Instruction Memory
- **IR**: Instruction Register
- **Mem**: Memory
- **ALU**: Arithmetic Logic Unit
- **Mem**: Memory
- **Write-Back**: Write-Back

**Data Flows**

- **Addr**: Address
- **Rdata**: Data
- **Inst. Memory**: Instruction Memory
- **IR**: Instruction Register
- **Mem**: Memory
- **ALU**: Arithmetic Logic Unit
- **Mem**: Memory
- **Write-Back**: Write-Back

**Processor Resources**

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

<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><strong>IF</strong></td>
<td>I₁</td>
<td>I₂</td>
<td>I₃</td>
<td>I₄</td>
<td>I₅</td>
<td>t₆</td>
<td>t₇</td>
<td>....</td>
<td></td>
</tr>
<tr>
<td><strong>ID</strong></td>
<td>I₁</td>
<td>I₂</td>
<td>I₃</td>
<td>I₄</td>
<td>I₅</td>
<td>I₅</td>
<td>I₄</td>
<td>I₅</td>
<td></td>
</tr>
<tr>
<td><strong>EX</strong></td>
<td>I₁</td>
<td>I₂</td>
<td>I₃</td>
<td>I₄</td>
<td>I₅</td>
<td>I₅</td>
<td>I₄</td>
<td>I₅</td>
<td></td>
</tr>
<tr>
<td><strong>MA</strong></td>
<td>I₁</td>
<td>I₂</td>
<td>I₃</td>
<td>I₄</td>
<td>I₅</td>
<td>I₅</td>
<td>I₄</td>
<td>I₅</td>
<td></td>
</tr>
</tbody>
</table>
5-Stage Pipelined Execution

Resource Usage Diagram

- **I-Fetch (IF)**
  - **IF**
  - **I1**
  - **I2**
  - **I3**
  - **I4**
  - **I5**

- **Decode, Reg. Fetch (ID)**
  - **I1**
  - **I2**
  - **I3**
  - **I4**
  - **I5**

- **Execute (EX)**
  - **I1**
  - **I2**
  - **I3**
  - **I4**
  - **I5**

- **Memory (MA)**
  - **I1**
  - **I2**
  - **I3**
  - **I4**
  - **I5**

- **Write-Back (WB)**
  - **I1**
  - **I2**
  - **I3**
  - **I4**
  - **I5**

**Resources**
- **time**
- **t0**
- **t1**
- **t2**
- **t3**
- **t4**
- **t5**
- **t6**
- **t7**
- **...**

**PC**
- **Addr**
- **Inst. Memory**

**Write-Back (WB)**
- **Write-Back (WB)**
- **Addr**
- **Data Memory**
- **rdata**
- **wdata**

**Instruction Memory**
- **Addr**
- **IR**

**Decode**
- **Imm Ext**
- **rs1**
- **rs2**
- **rd1**
- **ws**
- **wr**
- **rd2**

**ALU**
- **addr**
- **rdata**
- **Ext**

**Memory (MA)**
- **Addr**
- **Data Memory**
- **rdata**
- **wdata**

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

March 4, 2021

MIT 6.823 Spring 2021
5-Stage Pipelined Execution
Resource Usage Diagram

I-Fetch (IF)

Decode, Reg. Fetch (ID)

Execute (EX)

Memory (MA)

Write-Back (WB)

<table>
<thead>
<tr>
<th>Resources</th>
<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>IF</td>
<td></td>
<td>I₁</td>
<td>I₂</td>
<td>I₃</td>
<td>I₄</td>
<td>I₅</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td></td>
<td></td>
<td>I₁</td>
<td>I₂</td>
<td>I₃</td>
<td>I₄</td>
<td>I₅</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX</td>
<td></td>
<td>I₁</td>
<td>I₁</td>
<td>I₂</td>
<td>I₃</td>
<td>I₄</td>
<td>I₅</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MA</td>
<td></td>
<td>I₁</td>
<td>I₁</td>
<td>I₂</td>
<td>I₃</td>
<td>I₄</td>
<td>I₅</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WB</td>
<td></td>
<td>I₁</td>
<td>I₁</td>
<td>I₂</td>
<td>I₃</td>
<td>I₄</td>
<td>I₅</td>
<td>I₅</td>
<td>I₅</td>
<td></td>
</tr>
</tbody>
</table>
Pipelined Execution

**ALU Instructions**

![Pipelined Execution Diagram]
Pipelined Execution

ALU Instructions

Not quite correct!
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
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 MIPS Datapath
without jumps

What else is needed?

March 4, 2021
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

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
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 a value 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**

\[
\begin{align*}
\text{r1} &\leftarrow r0 + 10 \\
r4 &\leftarrow r1 + 17 \\
\end{align*}
\]
Data Hazards

... 
\[ r1 \leftarrow r0 + 10 \] 
\[ r4 \leftarrow r1 + 17 \] 
...

March 4, 2021
Data Hazards

\[ r_1 \leftarrow r_0 + 10 \]
\[ r_4 \leftarrow r_1 + 17 \]
Data Hazards

... 
\[ r_1 \leftarrow r_0 + 10 \] 
\[ r_4 \leftarrow r_1 + 17 \] 
...

\[ r_1 \text{ is stale. Oops!} \]
Resolving Data Hazards

Use strategy from Princeton Pipeline:

Wait for the result to be available by freezing earlier pipeline stages $\rightarrow$ stall
Feedback to Resolve Hazards

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

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

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

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

Later stages provide dependence information to earlier stages which can *stall instructions*.
Feedback to Resolve Hazards

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

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

- Later stages provide dependence information to earlier stages which can stall 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).
Resolving Data Hazards by Stalling

... 
\( r1 \leftarrow r0 + 10 \)
\( r4 \leftarrow r1 + 17 \)
...

March 4, 2021
Resolving Data Hazards by Stalling

... 

r1 ← r0 + 10 

r4 ← r1 + 17 

...
Resolving Data Hazards by Stalling

... 
\[ r_1 \leftarrow r_0 + 10 \]
\[ r_4 \leftarrow r_1 + 17 \]
...

March 4, 2021
Resolving Data Hazards by Stalling

Stall Condition

... r1 ← r0 + 10
r4 ← r1 + 17
...

March 4, 2021
Stalled Stages and Pipeline Bubbles
Stalled Stages and Pipeline Bubbles

\( time \)
\( t_0 \quad t_1 \quad t_2 \quad t_3 \quad t_4 \quad t_5 \quad t_6 \quad t_7 \quad \ldots \ldots \)
Stalled Stages and Pipeline Bubbles

\[
\begin{align*}
time \\
t0 & \quad t1 & \quad t2 & \quad t3 & \quad t4 & \quad t5 & \quad t6 & \quad t7 & \ldots \\
(I_1) \quad r1 \leftarrow (r0) + 10 & \quad IF_1 & \quad ID_1 & \quad EX_1 & \quad MA_1 & \quad WB_1
\end{align*}
\]
Stalled Stages and Pipeline Bubbles

\[
\begin{array}{cccccccc}
\text{time} & t0 & t1 & t2 & t3 & t4 & t5 & t6 & t7 \\
(I_1) & r1 & \leftarrow & (r0) + 10 & \text{IF}_1 & \text{ID}_1 & \text{EX}_1 & \text{MA}_1 & \text{WB}_1 \\
(I_2) & r4 & \leftarrow & (r1) + 17 & \text{IF}_2 & \text{ID}_2 & \text{ID}_2 & \text{ID}_2 & \text{EX}_2 & \text{MA}_2 & \text{WB}_2
\end{array}
\]
Stalled Stages and Pipeline Bubbles

\(r_1 \leftarrow (r_0) + 10\) 
\(r_4 \leftarrow (r_1) + 17\)

\(I_1\) 
\(I_2\) 
\(I_3\)

\(t0\) \(t1\) \(t2\) \(t3\) \(t4\) \(t5\) \(t6\) \(t7\) ....

IF\(_1\) ID\(_1\) EX\(_1\) MA\(_1\) WB\(_1\)
IF\(_2\) ID\(_2\) ID\(_2\) ID\(_2\) ID\(_2\) EX\(_2\) MA\(_2\) WB\(_2\)
IF\(_3\) IF\(_3\) IF\(_3\) IF\(_3\) ID\(_3\) EX\(_3\) MA\(_3\) 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>
</table>

(I\_1) \( r_1 \leftarrow (r_0) + 10 \)

(I\_2) \( r_4 \leftarrow (r_1) + 17 \)

(I\_3)

(I\_4)

\[ \begin{array}{cccccccc}
\text{IF}_1 & \text{ID}_1 & \text{EX}_1 & \text{MA}_1 & \text{WB}_1 \\
\text{IF}_2 & \text{ID}_2 & \text{ID}_2 & \text{ID}_2 & \text{ID}_2 & \text{EX}_2 & \text{MA}_2 & \text{WB}_2 \\
\text{IF}_3 & \text{ID}_3 & \text{IF}_3 & \text{IF}_3 & \text{IF}_3 & \text{ID}_3 & \text{EX}_3 & \text{MA}_3 & \text{WB}_3 \\
\text{IF}_4 & \text{ID}_4 & \text{EX}_4 & \text{MA}_4 & \text{WB}_4 \\
\end{array} \]
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</td>
<td>←</td>
<td>(r0)</td>
<td>+</td>
<td>10</td>
<td>IF_1</td>
<td>ID_1</td>
<td>EX_1</td>
<td>MA_1</td>
<td>WB_1</td>
</tr>
<tr>
<td>(I_2) r4</td>
<td>←</td>
<td>(r1)</td>
<td>+</td>
<td>17</td>
<td>IF_2</td>
<td>ID_2</td>
<td>ID_2</td>
<td>ID_2</td>
<td>ID_2</td>
</tr>
<tr>
<td>(I_3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>IF_3</td>
<td>ID_3</td>
<td>IF_3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(I_4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(I_5)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

March 4, 2021
MIT 6.823 Spring 2021

L06-26
Stalled Stages and Pipeline Bubbles

\[ \text{time} \]
\[
\begin{array}{cccccccc}
  t_0 & t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 & \ldots \\
\end{array}
\]

(I_1) \quad r_1 \leftarrow (r_0) + 10

(I_2) \quad r_4 \leftarrow (r_1) + 17

(I_3)

(I_4)

(I_5)

\text{IF}_1 \quad \text{ID}_1 \quad \text{EX}_1 \quad \text{MA}_1 \quad \text{WB}_1

\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

\text{IF}_3 \quad \text{IF}_3 \quad \text{IF}_3 \quad \text{IF}_3 \quad \text{IF}_3 \quad \text{IF}_3 \quad \text{EX}_3 \quad \text{MA}_3 \quad \text{WB}_3

\text{IF}_4 \quad \text{ID}_4 \quad \text{EX}_4 \quad \text{MA}_4 \quad \text{WB}_4

\text{IF}_5 \quad \text{ID}_5 \quad \text{EX}_5 \quad \text{MA}_5 \quad \text{WB}_5
Stalled Stages and Pipeline Bubbles

Stalled stages

\[ \text{time} \]
\[ t_0 \quad t_1 \quad t_2 \quad t_3 \quad t_4 \quad t_5 \quad t_6 \quad t_7 \quad \ldots \]

(I) \( r_1 \leftarrow (r_0) + 10 \)

(II) \( r_4 \leftarrow (r_1) + 17 \)

stalled stages

\[ r_1 \]

\[ r_4 \]

\[ r_5 \]

\[ r_6 \]

\[ r_7 \]
Stalled Stages and Pipeline Bubbles

Resource Usage
Stalled Stages and Pipeline Bubbles

Resource Usage

March 4, 2021
Stalled Stages and Pipeline Bubbles

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

 IF 1  IF 1  EX 1  MA 1  WB 1  IF 2  IF 2  IF 2  IF 3  IF 3  EX 2  MA 2  WB 2  IF 4  ID 4  ID 4  ID 4  ID 4  EX 4  MA 4  WB 4  IF 5  ID 5  EX 5  MA 5  WB 5

(I_1) r1 ← (r0) + 10
(I_2) r4 ← (r1) + 17
(I_3)
(I_4)
(I_5)

Resource Usage

L06-26
Stalled Stages and Pipeline Bubbles

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

\( (I_3) \)
\( (I_4) \)
\( (I_5) \)

\textbf{Resource Usage}

\textit{time}
\begin{tabular}{cccccccc}
  t0 & t1   & t2 & t3 & t4 & t5 & t6 & t7 \\
  \text{IF} & \( I_1 \) & \( I_2 \) & \( I_3 \) & \( I_3 \) & \( I_3 \) & \( I_3 \) & \( I_4 \) & \( I_5 \) \\
  \text{ID} & \( I_1 \) & \( I_2 \) & \( I_2 \) & \( I_2 \) & \( I_2 \) & \( I_3 \) & \( I_4 \) & \( I_5 \) \\
\end{tabular}

\textbf{stalled stages}

\( \text{IF}_1 \rightarrow \text{ID}_1 \rightarrow \text{EX}_1 \rightarrow \text{MA}_1 \rightarrow \text{WB}_1 \)
\( \text{IF}_2 \rightarrow \text{ID}_2 \rightarrow \text{ID}_2 \rightarrow \text{ID}_2 \rightarrow \text{ID}_2 \rightarrow \text{EX}_2 \rightarrow \text{MA}_2 \rightarrow \text{WB}_2 \)
\( \text{IF}_3 \rightarrow \text{ID}_3 \rightarrow \text{ID}_3 \rightarrow \text{ID}_3 \rightarrow \text{ID}_3 \rightarrow \text{EX}_3 \rightarrow \text{MA}_3 \rightarrow \text{WB}_3 \)
\( \text{IF}_4 \rightarrow \text{ID}_4 \rightarrow \text{ID}_4 \rightarrow \text{EX}_4 \rightarrow \text{MA}_4 \rightarrow \text{WB}_4 \)
\( \text{IF}_5 \rightarrow \text{ID}_5 \rightarrow \text{EX}_5 \rightarrow \text{MA}_5 \rightarrow \text{WB}_5 \)
Stalled Stages and Pipeline Bubbles

Time:
- $t_0$, $t_1$, $t_2$, $t_3$, $t_4$, $t_5$, $t_6$, $t_7$, ...

Instructions:
- $I_1$: $r_1 \leftarrow (r_0) + 10$
- $I_2$: $r_4 \leftarrow (r_1) + 17$

Stalled stages:
- $I_1$, $I_2$, $I_3$, $I_4$, $I_5$

Resource Usage:
- IF: $I_1$, $I_2$, $I_3$, $I_3$, $I_3$, $I_4$, $I_5$, ...
- ID: $I_1$, $I_2$, $I_2$, $I_2$, $I_2$, $I_3$, $I_4$, $I_5$, ...
- EX: $I_1$, nop, nop, nop, $I_2$, $I_3$, $I_4$, $I_5$, ...

MIT 6.823 Spring 2021
March 4, 2021
Stalled Stages and Pipeline Bubbles

\[
\begin{array}{cccccccc}
time & t0 & t1 & t2 & t3 & t4 & t5 & t6 & t7 & \ldots \ldots \\
\text{(I}_1\text{)} & r1 & \leftarrow & (r0) & + & 10 & \text{IF}_1 & \text{ID}_1 & \text{EX}_1 & \text{MA}_1 & \text{WB}_1 \\
\text{(I}_2\text{)} & r4 & \leftarrow & (r1) & + & 17 & \text{IF}_2 & \text{ID}_2 & \text{ID}_2 & \text{ID}_2 & \text{EX}_2 & \text{MA}_2 & \text{WB}_2 \\
\text{(I}_3\text{)} & & & & & & & \text{ID}_3 & \text{ID}_3 & \text{ID}_3 & \text{EX}_3 & \text{MA}_3 & \text{WB}_3 \\
\text{(I}_4\text{)} & & & & & & & \text{ID}_4 & \text{ID}_4 & \text{ID}_4 & \text{EX}_4 & \text{MA}_4 & \text{WB}_4 \\
\text{(I}_5\text{)} & & & & & & & \text{ID}_5 & \text{ID}_5 & \text{ID}_5 & \text{EX}_5 & \text{MA}_5 & \text{WB}_5 \\
\end{array}
\]

stalled stages

Resource Usage

\[
\begin{array}{cccccccc}
\text{time} & t0 & t1 & t2 & t3 & t4 & t5 & t6 & t7 & \ldots \ldots \\
\text{IF} & \text{I}_1 & \text{I}_2 & \text{I}_3 & \text{I}_3 & \text{I}_3 & \text{I}_3 & \text{I}_3 & \text{I}_4 & \text{I}_4 & \text{I}_4 & \text{I}_4 & \text{I}_5 & \text{I}_5 \\
\text{ID} & \text{I}_1 & \text{I}_2 & \text{I}_2 & \text{I}_2 & \text{I}_2 & \text{I}_2 & \text{I}_2 & \text{I}_3 & \text{I}_3 & \text{I}_3 & \text{I}_3 & \text{I}_5 & \text{I}_5 \\
\text{EX} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} & \text{nop} \\
\text{MA} & \text{I}_1 & \text{I}_2 & \text{nop} & \text{I}_2 & \text{I}_3 & \text{I}_4 & \text{I}_5 & \text{I}_5 & \text{I}_5 & \text{I}_5 & \text{I}_5 & \text{I}_5 & \text{I}_5 \\
\end{array}
\]
Stalled Stages and Pipeline Bubbles

Time

t0    t1    t2    t3    t4    t5    t6    t7    . . .

(I_1) r1 ← (r0) + 10

(I_2) r4 ← (r1) + 17

(I_3)

(I_4)

(I_5)

Stalled stages

Resource Usage

Time

t0    t1    t2    t3    t4    t5    t6    t7    . . .

IF   I_1   I_2   I_3   I_3   I_3   I_4   I_5

ID   I_1   I_2   I_3   I_3   I_3   I_4   I_5

EX   I_1   nop   nop   nop   I_2   I_3   I_4   I_5

MA   I_1   nop   nop   nop   nop   I_2   I_3   I_4   I_5

WB   I_1   nop   nop   nop   nop   nop   I_2   I_3   I_4   I_5
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>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>IF</td>
<td>I_1</td>
<td>I_2</td>
<td>I_1</td>
<td>I_1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td></td>
<td>I_1</td>
<td>I_2</td>
<td>I_2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>n0p</td>
<td>n0p</td>
<td>n0p</td>
<td>n0p</td>
<td></td>
</tr>
<tr>
<td>WB</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>n0p</td>
<td>n0p</td>
<td>n0p</td>
<td>n0p</td>
<td></td>
</tr>
</tbody>
</table>

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

\( (I_3) \)
\( (I_4) \)
\( (I_5) \)
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 ← (r0) + 10</td>
<td>IF_1</td>
<td>ID_1</td>
<td>EX_1</td>
<td>MA_1</td>
<td>WB_1</td>
<td>IF_2</td>
<td>ID_2</td>
<td>ID_2</td>
<td>ID_2</td>
</tr>
<tr>
<td>(I_2) r4 ← (r1) + 17</td>
<td>IF_2</td>
<td>ID_2</td>
<td>ID_2</td>
<td>ID_2</td>
<td>IF_3</td>
<td>ID_3</td>
<td>EX_2</td>
<td>MA_2</td>
<td>WB_2</td>
</tr>
<tr>
<td>(I_3)</td>
<td>IF_3</td>
<td>ID_3</td>
<td>ID_3</td>
<td>IF_3</td>
<td>EX_3</td>
<td>MA_3</td>
<td>WB_3</td>
<td>IF_4</td>
<td>ID_4</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>IF_5</td>
<td>ID_5</td>
<td>EX_5</td>
<td>MA_5</td>
</tr>
<tr>
<td>(I_5)</td>
<td>IF_5</td>
<td>ID_5</td>
<td>EX_5</td>
<td>MA_5</td>
<td>WB_5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Resource Usage**

<table>
<thead>
<tr>
<th>IF</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</td>
<td>I_1</td>
<td>I_2</td>
<td>I_3</td>
<td>I_3</td>
<td>I_3</td>
<td>I_3</td>
<td>I_4</td>
<td>I_5</td>
<td>. . .</td>
</tr>
<tr>
<td>I_2</td>
<td>I_2</td>
<td>I_2</td>
<td>I_2</td>
<td>I_2</td>
<td>I_2</td>
<td>I_2</td>
<td>I_4</td>
<td>I_5</td>
<td>. . .</td>
</tr>
<tr>
<td>I_3</td>
<td>I_3</td>
<td>I_3</td>
<td>I_3</td>
<td>I_3</td>
<td>I_3</td>
<td>I_3</td>
<td>I_5</td>
<td>I_5</td>
<td>. . .</td>
</tr>
<tr>
<td>I_5</td>
<td>I_5</td>
<td>I_5</td>
<td>I_5</td>
<td>I_5</td>
<td>I_5</td>
<td>I_5</td>
<td>I_5</td>
<td>I_5</td>
<td>. . .</td>
</tr>
</tbody>
</table>

**nop** \(\Rightarrow\) **pipeline bubble**
Stall Control Logic

Compare the source registers of the instruction in the decode stage with the destination register of the uncommitted instructions.
Stall 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.
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.
Should we always stall if the rs field matches some rd?
Stall Control Logic
ignoring jumps & branches

Should we always stall if the rs field matches some rd?

not every instruction writes a register ⇒ we
Should we always stall if the rs field matches some rd?

- not every instruction writes a register ⇒ we
- not every instruction reads a register ⇒ re
Should we always stall if the rs field matches some rd?

not every instruction writes a register ⇒ we
not every instruction reads a register ⇒ re
Stall Control Logic

ignoring jumps & branches

Should we always stall if the rs field matches some rd?

not every instruction writes a register \implies we

not every instruction reads a register \implies re
Stall Control Logic
ignoring jumps & branches

Should we always stall if the rs field matches some rd?

not every instruction writes a register ⇒ we
not every instruction reads a register ⇒ re
Should we always stall if the rs field matches some rd?

not every instruction writes a register ⇒ we
not every instruction reads a register ⇒ re
Thank you!