LE15.3

Bookmark this page

For all Beta related questions, you should make use of the Beta documentation, the Beta Instruction Summary, the Unpipelined Beta Diagram and the Pipelined Beta Diagram.

LE15.3.1: Exceptions

1.0/1.0 point (ungraded)

An ambitious 6.004x student has enhanced the 5-stage pipelined Beta to trigger an exception in the ALU stage when there's an attempt to divide by zero, i.e., when the B operand of the ALU is zero and the opcode of the instruction in the ALU stage is DIV or DIVC.

For each of the following control signals, indicate its value during the clock cycle when the divide-by-zero exception occurs.

1. $IRSrc^{IF}$

   1 ✔

2. $IRSrc^{RF}$

   1 ✔

3. $IRSrc^{ALU}$

   2 ✔

4. $IRSrc^{MEM}$

   0 ✔

Submit

LE15.3.2: Bypassing, stalls, annulment

0.0/1.0 point (ungraded)

The loop below has been executing for a while on our standard 5-stage pipelined Beta with full bypassing and speculation on branches (i.e., instruction in IF stage is annulled when there's a taken branch in the RF stage). The pipeline diagram below shows the opcode of the instruction in each pipeline stage during 9 consecutive cycles of execution.

```
      ...
LOOP: LD(R0, 0, R2)
      SHLC(R1, 1, R1)
      ADD(R1, R2, R1)
      ADDC(R0, 4, R0)
      CMPEQC(R0, aend, R2)
      BNE(R2, LOOP)
      ST(R1, hash, R31)
      ...
```

Cycle #	**200**	**201**	**202**	**203**	**204**	**205**	**206**	**207**	**208**
IF	LD	SHLC	ADD	ADDC	ADDC	CMPEQC	BNE	ST	LD
RF	NOP	LD	SHLC	ADD	ADD	ADDC	CMPEQC	BNE	NOP
ALU	BNE	NOP	LD	SHLC	NOP	ADD	ADDC	CMPEQC	BNE
MEM	CMPEQC	BNE	NOP	LD	SHLC	NOP	ADD	ADDC	CMPEQC
WB	ADDC	CMPEQC	BNE	NOP	LD	SHLC	NOP	ADD	ADDC

1. For each cycle, specify whether or not a bypass path was used to get the data from a later pipeline stage to the RF stage. If multiple bypass paths are used in a particular cycle, then select all that apply.

   Cycle 200:

- [x] No bypass paths used
- [ ] Bypass from ALU to RF
- [ ] Bypass from MEM to RF
- [ ] Bypass from WB to RF

Cycle 201:

- [x] No bypass paths used
- [ ] Bypass from ALU to RF
- [ ] Bypass from MEM to RF
- [ ] Bypass from WB to RF

Cycle 202:

- [x] No bypass paths used
- [ ] Bypass from ALU to RF
- [ ] Bypass from MEM to RF
- [ ] Bypass from WB to RF

Cycle 203:

- [x] No bypass paths used
- [ ] Bypass from ALU to RF
- [ ] Bypass from MEM to RF
- [ ] Bypass from WB to RF

Cycle 204:

- [ ] No bypass paths used
- [ ] Bypass from ALU to RF
- [x] Bypass from MEM to RF

- [x] Bypass from WB to RF

Cycle 205:

- [x] No bypass paths used
- [ ] Bypass from ALU to RF
- [ ] Bypass from MEM to RF
- [ ] Bypass from WB to RF

Cycle 206:

- [ ] No bypass paths used
- [x] Bypass from ALU to RF
- [ ] Bypass from MEM to RF
- [ ] Bypass from WB to RF

Cycle 207:

- [ ] No bypass paths used
- [x] Bypass from ALU to RF
- [ ] Bypass from MEM to RF
- [ ] Bypass from WB to RF

Cycle 208:

- [x] No bypass paths used
- [ ] Bypass from ALU to RF
- [ ] Bypass from MEM to RF
- [ ] Bypass from WB to RF

For the following questions think carefully about when a signal would be asserted in order to produce the effect you see in the pipeline diagram. Select all the cycles that apply, or select NONE if it never occurs.

2. During which cycle(s), if any, would the $IRSrc^{IF}$ signal be 12

2. During which cycle(s), if any, would the IRSrc^{RF} signal be 1?

- [ ] 200
- [ ] 201
- [ ] 202
- [ ] 203
- [ ] 204
- [ ] 205
- [ ] 206
- [x] 207
- [ ] 208
- [ ] NONE

3. During which cycle(s), if any, would the IRSrc^{RF} signal be 1?

- [ ] 200
- [ ] 201
- [ ] 202
- [x] 203
- [ ] 204
- [ ] 205
- [ ] 206
- [ ] 207
- [ ] 208
- [ ] NONE

4. During which cycle(s), if any, would the IRSrc^{ALU} signal be 1?

- [ ] 200
- [ ] 201
- [ ] 202
- [ ] 203
- [ ] 204
- [ ] 205

- [ ] 205
- [ ] 206
- [ ] 207
- [ ] 208
- [x] NONE ✔

5. During which cycle(s), if any, would the STALL control signal be 1, i.e., cycle(s) when the IF and RF stages would be stalled?

- [ ] 200
- [ ] 201
- [ ] 202
- [x] 203 ✔
- [ ] 204
- [ ] 205
- [ ] 206
- [ ] 207
- [ ] 208
- [ ] NONE

Explanation
The LD is fetched in cycle 200 and proceeds down the pipeline stages. The SHLC reads R1 so it does not have any data hazards. The ADD needs to read R1 and R2 so it depends on the results of the LD and the SHLC operation. The result from the LD cannot be bypassed until the LD is in the WB stage which occurs in cycle 204. This means that in cycle 203, we need to stall the ADD because it cannot yet read its operands. Then in cycle 204, the result of the LD is bypassed from the WB stage and the result of the SHLC is bypassed from the MEM stage. The ADDC then proceeds since it does not have any data hazards. The CMPEQC needs the result of the ADDC, so in cycle 206 it uses the bypass path from the ALU stage to get the result of the ADDC. The BNE also needs to use a bypass path from the ALU stage to read the value of R2 that is produced by the CMPEQC instruction. Finally, in cycle 207, the ST which was fetched following the branch must be annulled because the branch is taken.
$IRSrc^{IF}$ is set to 1 when the instruction in the IF stage has to be annulled because of a taken branch. This happens in cycle 207.
$IRSrc^{RF}$ is set to 1 when STALL is set to 1. STALL is set to 1 when a register value needed by the instruction in the RF stage is not yet available in the data path or register file. This happens in cycle 203.
$IRSrc^{ALU}$ is set to 1 when the instruction in the ALU has to be annulled because of an exception in later pipeline stages, i.e., the MEM stage. That doesn't happen in this instruction sequence.

Submit

Answers are displayed within the problem

Discussion

Topic: 15. Pipelining the Beta / LE15.3

Hide Discussion

Add a Post

Show all posts by recent activity

‹ Previous Next ›