EXERCISES on REORDER BUFFER (V2022)

ACA2020_EXAM_PART2_25June2020

Question 1 (format open text)

Let's consider the following code:

```
LOOP: LD $F0, 0 ($R1)
      MULTD $F4, $F0, $F2
      SD $F4, 0 ($R1)
      SUBI $R1, $R1, 8
      BNEZ $R1, LOOP
```

*executed by the **Speculative Tomasulo** architecture with:*
2 load buffers (Load1, Load2),
2 multiply RS (Mult1, Mult2),
2 integer RS (Int1, Int2)
and 8-entry ReOrder Buffer.
Let's consider the following situation when the ROB is FULL:

ROB#	Instruction	Dest.	Ready	SPECUL.	
ROB0	LD $F0, 0 ($R1)	$F0	No (in exec. Dcache miss)	No	**HEAD**
ROB1	MULTD $F4, $F0, $F2	$F4	No (issued)	No	
ROB2	SD $F4, 0 ($R1)	MEM	No (issued)	No	
ROB3	SUBI $R1, $R1, 8	$R1	No (in exec.)	No	
ROB4	BNEZ $R1, LOOP (pred. taken)	--	No (issued	No	
ROB5	LD $F0, 0 ($R1) (2^ iter.)	$F0	No (issued)	**Yes**	
ROB6	MULTD $F4, $F0, $F2 (2^ iter.)	$F4	No(issued)	**Yes**	
ROB7	SD $F4, 0 ($R1) (2^ iter.)	MEM	No (issued)	**Yes**	

Rename Table:

$F0	ROB5
$F2	
$F4	ROB6

Let's explain what happens in the ROB and in the Rename Table if the branch (allocated in ROB4) is mispredicted.

Open text answer: (max 50 words)

Solution:

Let's assume the value of $R1 has been initialized at 0 and $F2 at 20, this is the situation when the ROB is full:

Reservation Stations:

Load1	LD $F0, 0 ($R1) 1st iter.	0	0	ROB0
Load2	LD $F0, 0 ($R1) 2nd iter.	0	ROB3	ROB5

Mult1	MULTD $F4, $F0, $F2 1st iter.	ROB0	20	ROB1
Mult2	MULTD $F4, $F0, $F2 2nd iter.	ROB5	20	ROB6

Int1	SUBI $R1, $R1, 8	0	8	ROB3
Int2	BNEZ $R1, LOOP	ROB3	0	ROB4

Rename Table:

$F0	ROB5
$F2	
$F4	ROB6

*When the 1^ branch is mispredicted, the entries **ROB5, ROB6 and ROB7 must be flushed.***

Reservation Stations:

Load1	LD $F0, 0 ($R1) 1st iter.	0	0	ROB0
Load2				

Mult1	MULTD $F4, $F0, $F2 1st iter.	ROB0	20	ROB1
Mult2				

Int1	SUBI $R1, $R1, 8.	0	8	ROB3
Int2	BNEZ $R1, LOOP	ROB3	0	ROB4

Therefore, the Rename Table is updated as follow: $F0 points to ROB0 and $F4 points to ROB1:

Rename Table:

$F0	~~ROB5~~ ROB0
$F2	
$F4	~~ROB6~~ ROB1

See also L07: Reorder Buffer & Speculation

ACA2020_EXAM_PART1_16July2020

Question 2 (format Multiple Choice – Single answer) ReOrder Buffer

Let's consider the following code:

```
LOOP: LD $F0, 0 ($R0)
      MULTD $F4, $F0, $F2
      SD $F4, 0 ($R0)
      SUBI $R0, $R0, 8
      BNEZ $R0, LOOP
```

*executed by the **Speculative Tomasulo** architecture with 2 load buffers (Load1, Load2), 2 multiply RS (Mult1, Mult2), 2 integer RS (Int1, Int2) and 6-entry ROB which is FULL in the following situation:*

ROB#	Instruction	Dest.	Ready	SPECUL.	
ROB0	MULTD $F4, $F0, $F2	$F4	No (issued)	**Yes**	
ROB1	MULTD $F4, $F0, $F2	$F4	No (in exec.)	No	**HEAD**
ROB2	SD $F4, 0 ($R0)	MEM	No (issued)	No	
ROB3	SUBI $R0, $R0, 8	$R0	No (issued)	No	
ROB4	BNEZ $R0, LOOP (pred. taken)	--	No (issued)	No	
ROB5	LD $F0, 0 ($R0)	$F0	No (issued)	**Yes**	

*In the Rename Table, what are the pointer values used for $F0 and $F4? **(SINGLE ANSWER)***

Answer 1: ROB5 for $F0; ROB0 for $F4 **(TRUE)**
Answer 2: ROB5 for $F0; ROB1 for $F4
Answer 3: ROB5 for $F0; ROB2 for $F4

Feedback:

Rename Table:

$F0	ROB5
$F2	
$F4	ROB0

In the Rename Table, $F0 points to ROB5 because the second LD has been issued speculatively while $F4 points to ROB0 because the second MULTD has been issued speculatively. See also L07: Reorder Buffer & Speculation

ACA2020_EXAM_PART1_31Aug2020

Question 5 (format Multiple Choice – Single answer)

Let's consider the following code:

I0: LD $F6, VECTA ($R6)
I1 ADDD $F4, $F0, $F2
I2 SUBD $F8, $F4, $F6
I3 MULD $F4, $F4, $F8
I4 SD $F4, VECTA ($R6)
I5: LD $F6, VECTB ($R6)

*executed by the **Speculative Tomasulo** architecture with 2 load buffers (Load1, Load2), 2 MUL FP RS (Mult1, Mult2), 2 ADD FP RS (ADD1, ADD2) and 6-entry ROB which is FULL in the following situation:*

ROB#	Instruction	Dest.	Ready	SPECUL.	
ROB0	I5: LD $F6, VECTB ($R6)	$F6	No (issued)	No	
ROB1	I0: LD $F6, VECTA ($R6)	$F6	No (in exec.)	No	**HEAD**
ROB2	I1: ADDD $F4, $F0, $F2	$F4	No (issued)	No	
ROB3	I2: SUBD $F8, $F4, $F6	$F8	No (issued)	No	
ROB4	I3: MULD $F4, $F4, $F8	$F4	No (issued)	No	
ROB5	I4: SD $F4, VECTA ($R6)	MEM	No (issued)	No	

*In the Rename Table, what are the pointers used for $F6 and $F4? **(SINGLE ANSWER)***

Answer 1: ROB0 for $F6; ROB4 for $F4 **(TRUE)**
Answer 2: ROB1 for $F6; ROB2 for $F4
Answer 3: ROB0 for $F6; ROB5 for $F4
Answer 4: ROB1 for $F6; ROB5 for $F4

Feedback:

Rename Table:

$F0	--
$F2	--
$F4	ROB4
$F6	ROB0
$F8	ROB3

In the Rename Table, $F6 points to ROB0 because the second LD has been issued by solving the WAW, while $F4 points to ROB4 because the MULTD has been issued by solving the WAW. See also L07: Reorder Buffer & Speculation

ACA2020_EXAM_PART2_11Jan2021

Question 1 (format Multiple Choice – Single answer)

Let's consider the following code:

I0: LD $F6, VECTA ($R1)
I1 SUBD $F6, $F4, $F6
I2 SD $F6, VECTA ($R1)
I3 LD $F8, VECTB ($R1)
I4 SUBD $F8, $F4, $F8
I5 SD $F8, VECTB ($R1)

*executed by the **Speculative Tomasulo** architecture with 2 load buffers (Load1, Load2), 2 FP RS (FP1, FP2) and 6-entry ROB which is FULL in the following situation:*

ROB#	Instruction	Dest.	Ready	SPECUL.	
ROB0	I0: LD $F6, VECTA ($R1)	$F6	No (in exec.)	No	**HEAD**
ROB1	I1: SUBD $F6, $F4, $F6	$F6	No (issued)	No	
ROB2	I2: SD $F6, VECTA ($R1)	MEM	No (issued)	No	
ROB3	I3: LD $F8, VECTB ($R1)	$F8	No (issued)	No	
ROB4	I4: SUBD $F8, $F4, $F8	$F8	No (issued)	No	
ROB5	I5: SD $F8, VECTB ($R1)	MEM	No (issued)	No	

*In the Rename Table, what are the pointers used for $F6 and $F8? **(SINGLE ANSWER)***

Answer 1: ROB1 for $F6; ROB4 for $F8 **(TRUE)**
Answer 2: ROB0 for $F6; ROB3 for $F8
Answer 3: ROB2 for $F6; ROB5 for $F8

Feedback:

Rename Table:

$F0	--
$F2	--
$F4	--
$F6	ROB1
$F8	ROB4

In the Rename Table, $F6 points to ROB1 because I1 has been issued by solving the WAW with I0; $F8 points to ROB4 because I4 has been issued by solving the WAW with I3. See also L07: Reorder Buffer & Speculation

ACA2020_EXAM_PART2_8Feb2021

Question 1 (format Multiple Choice – Single answer)

Let's consider the following code:

I0: LD $F6, VECTA ($R1)
I1 LD $F4, VECTB ($R1)
I2 ADDD $F6, $F6, $F8
I3 SUBD $F4, $F4, $F8
I4 SD $F6, VECTA ($R1)
I5 SD $F4, VECTB ($R1)

*executed by the **Speculative Tomasulo** architecture with 2 load buffers (Load1, Load2), 2 FP RS (FP1, FP2) and 6-entry ROB which is FULL in the following situation:*

ROB#	Instruction	Dest.	Ready	SPECUL.	
ROB0	I0: LD $F6, VECTA ($R1)	$F6	No (in exec.)	No	**HEAD**
ROB1	I1: LD $F4, VECTB ($R1)	$F4	No (issued)	No	
ROB2	I2: ADDD $F6, $F6, $F8	$F6	No (issued)	No	
ROB3	I3: SUBD $F4, $F4, $F8	$F4	No (issued)	No	
ROB4	I4: SD $F6, VECTA ($R1)	MEM	No (issued)	No	
ROB5	I5: SD $F4, VECTB ($R1)	MEM	No (issued)	No	

*In the Rename Table, what are the pointers used for $F4 and $F6? **(SINGLE ANSWER)***

Answer 1: ROB3 for $F4; ROB2 for $F6 **(TRUE)**
Answer 2: ROB1 for $F4; ROB0 for $F6
Answer 3: ROB5 for $F4; ROB4 for $F6

Feedback:

Rename Table:

$F0	--
$F2	--
$F4	ROB3
$F6	ROB2
$F8	

In the Rename Table, $F4 points to ROB3 because I3 has been issued by solving the WAW with I1; $F6 points to ROB2 because I2 has been issued by solving the WAW with I0. See also L07: Reorder Buffer & Speculation

ACA2020_EXE_SESSION_03May2021

Question ROB (format OPEN TEXT)

Let's consider the following code:

```
I0:    LOOP: LD $F6, VECTA ($R1)
I1:          SUBD $F6, $F4, $F6
I2:          SD $F6, VECTA ($R1)
I3:          SUBI $R1, $R1, 8
I4:          BNEZ $R1, LOOP
```

*executed by the **Speculative Tomasulo** architecture with 2 load buffers (Load1, Load2), 2 FP RS (FP1, FP2), 2 integer RS (Int1, Int2) and 6-entry ROB.*
Let's consider this starting situation:

ROB#	Instruction	Dest.	Ready	SPECUL.	
ROB0					
ROB1	I1: SUBD $F6, $F4, $F6	$F6	No (In exec.)	No	**HEAD**
ROB2	I2: SD $F6, VECTA ($R1)	MEM	No (Issued)	No	
ROB3					**TAIL**
ROB4					
ROB5					

Rename Table:

$F0	--
$F2	--
$F4	--
$F6	ROB1
$F8	--

Please complete the ROB entries and the RENAME TABLE after I1: SUBD has been committed, while I2: SD is in execution and the next instructions have been issued until the ROB is FULL;

Feedback:

ROB#	Instruction	Dest.	Ready	SPECUL.	
ROB0	I1: SUBD $F6, $F4, $F6	$F6	No (Issued)	Yes	
ROB1	I2: SD $F6, VECTA ($R1)	MEM	No (Issued)	Yes	
ROB2	**I2: SD $F6, VECTA ($R1)**	**MEM**	**No (In exec.)**	**No**	**HEAD**
ROB3	I3: SUBI $R1, $R1, 8	$R1	No (Issued)	No	
ROB4	I4: BNEZ $R1, LOOP (pred. taken)	--	No (Issued)	No	
ROB5	I0: LD $F6, VECTA ($R1)	$F6	No (Issued)	Yes	

Rename Table:

$F0	--
$F2	--
$F4	--
$F6	~~ROB1 ROB5~~ ROB0
$F8	

In the Rename Table, $F6 points to ROB0 because I1 at the 2nd iteration has been issued by solving the WAW with I0 at the 2nd iteration issued in ROB5.
See also L07: Reorder Buffer & Speculation

ACA2021_EXAM_PART2_10Jan2022

Question 3 (format open text MAX 15 ROWS)

Let's consider the following code:

```
I0:     LD.D $FP0, 0($R1)
I1:     ADDI.D $FP0, $FP0, 4
I2:     SD.D $FP0, 4($R1)

IF:     BEQ $R1, $R2, ELSE     /* branch predicted as NOT TAKEN */

THEN:  LD.D $FP6, 8($R1)
T1:     SD.D $FP6, 8($R6)
T2:     J EXIT

ELSE:  LD.D $FP2, 8($R1)
E1:     ADD.D $FP2, $FP2, $FP0
E2:     MUL.D $FP4, $FP2, $FP2
E3:     SD.D $FP2, 12($R1)

EXIT: LD.D $FP0, 4($R5)
```

executed by the **Speculative Tomasulo** architecture with 5 load buffers (Load0, Load1, Load2, Load3, Load4), 2 FP RS (FPU1, FPU2), 2 integer RS (Int1, Int2).
*Assuming the first LOAD instruction I0 is in execution to manage a long cache data cache miss until the ROB becomes **FULL**.*
Please complete the 8-entry ROB table and the Rename Table:

ROB Table:

ROB#	Instruction	Dest.	Ready	SPECUL.	HEAD
ROB0	I0: LD.D $FP0, 0($R1)	$FP0	No (in exec)	No	Yes
ROB1	I1: ADDI.D $FP0, $FP0, 4	$FP0	No (issued)	No	No
ROB2					
ROB3					
ROB4					
ROB5					
ROB6					
ROB7					

Rename Table:

$FP0	~~ROB0~~ ROB1
$FP2	
$FP4	
$FP6	

Open text answer (please copy/paste the two tables and complete them)
2 points

Feedback:

ROB Table:

ROB#	Instruction	Dest.	Ready	SPECUL.	HEAD
ROB0	**I0: LD.D $FP0, 0($R1)**	**$FP0**	**No (in exec)**	**No**	**Yes**
ROB1	**I1: ADDI.D $FP0, $FP0, 4**	**$FP0**	**No (issued)**	**No**	**No**
ROB2	I2: SD.D $FP0, 4($R1)	MEM	No (issued)	No	No
ROB3	IF: BEQ $R1, $R2, ELSE	None	No (issued)	No	No
ROB4	THEN: LD.D $FP6, 8($R1)	$FP6	No (issued)	Yes	No
ROB5	T1: SD.D $FP6, 8($R6)	MEM	No (issued)	Yes	No
ROB6	T2: J EXIT	None	No (issued)	Yes	No
ROB7	EXIT: LD.D $FP0, 4($R5)	$FP0	No (issued)	Yes	No

Rename Table:

$FP0	~~ROB0~~ ~~ROB1~~ ROB7
$FP2	None
$FP4	None
$FP6	ROB4

ACA2021_EXAM_PART2_25Aug2021

Question 6 (format open text)

Let's consider the following code:

```
I0:     LD.D $FP0, 0($R1)
I1:     ADDI.D $FP0, $FP0, 4
I2:     SD.D $FP0, 4($R1)

IF:     BEQ $R1, $R2, ELSE    /* branch predicted as TAKEN */

THEN:  LD.D $FP6, 8($R1)
T1:     ADD.D $FP6, $FP6, $FP0
T2:     MUL.D $FP6, $FP6, $FP0
T3:     SD.D $FP6, 12($R1)

ELSE:  LD.D $FP2, 8($R1)
E1:     ADD.D $FP2, $FP2, $FP0
E2:     MUL.D $FP4, $FP2, $FP2
E3:     SD.D $FP2, 12($R1)
```

executed by the **Speculative Tomasulo** architecture with 2 load buffers (Load1, Load2), 4 FP RS (FPU1, FPU2, FP3, FP4), 2 integer RS (Int1, Int2).

Please complete the 8-entry ROB table and the Rename Table up to when the ROB becomes FULL:

ROB Table:

ROB#	Instruction	Dest.	Ready	SPECUL.	HEAD
ROB0	I0: LD.D $FP0, 0($R1)	$FP0	No (in exec)	No	Yes
ROB1	I1: ADDI.D $FP0, $FP0, 4	$FP0	No (issued)	No	No
ROB2					
ROB3					
ROB4					
ROB5					
ROB6					
ROB7					

Rename Table:

$F0	~~ROB0~~ ROB1
$F2	
$F4	
$F6	

Open text answer (please copy/paste the two tables and complete them)
2 points

Feedback:

ROB Table:

ROB#	Instruction	Dest.	Ready	SPECUL.	HEAD
ROB0	**I0: LD.D $FP0, 0($R1)**	**$FP0**	**No (in exec)**	**No**	**Yes**
ROB1	**I1: ADDI.D $FP0, $FP0, 4**	**$FP0**	**No (issued)**	**No**	**No**
ROB2	I2: SD.D $FP0, 4($R1)	MEM	No (issued)	No	No
ROB3	IF: BEQ $R1, $R2, ELSE	None	No (issued)	No	No
ROB4	ELSE: LD.D $FP2, 8($R1)	$FP2	No (issued)	Yes	No
ROB5	E1: ADD.D $FP2, $FP2, $FP0	$FP2	No (issued)	Yes	No
ROB6	E2: MUL.D $FP4, $FP2, $FP2	$FP4	No (issued)	Yes	No
ROB7	E3: SD.D $FP2, 12($R1)	MEM	No (issued)	Yes	No

Rename Table:

$F0	~~ROB0~~ ROB1
$F2	~~ROB4~~ ROB5
$F4	ROB6
$F6	None