BPU Examples

E. Sanchez

Politecnico di Torino
Dipartimento di Automatica e Informatica

BHT Example

Considering a 2-bit saturating counter BHT of 1K entries; and assuming that the processor executes the following code fragment, determine the BHT final state and calculate the misprediction ratio for the presented case.

```
for(i=1;i<=100;i++){
  read_data();
  if (aa==2)
     aa = 0;
  if (bb==2)
     bb = 0;
  if (aa == bb)
  {…}
}
```

BHT Example

Considering a 2-bit saturating counter BHT of 1K entries; and assuming that the processor executes the following code fragment, determine the BHT final state and calculate the misprediction ratio for the presented case.

```
for(i=1;i<=100;i++){
  read_data();
  if (aa==2)
     aa = 0;
  if (bb==2)
     bb = 0;
  if (aa == bb)
  {…}
}
```

```
L0:   …
      DADDUI  R3, R1, #-2
      BNEZ  R3, L1
      DADD  R1, R0, R0
L1:   DADDUI  R3, R2, #-2
      BNEZ  R3, L2
      DADD  R2, R0, R0
L2:   DSUB  R3, R1, R2
      BNEZ  R3, L3
      …
L3:   …
      DADDI R4, R4, #-1
      BNEZ R4, L0
      …
```

BHT Example

Considering a 2-bit saturating counter BHT of 1K entries; and assuming that the processor executes the following code fragment, determine the BHT final state and calculate the misprediction ratio for the presented case.

Case 1:

-Program inputs for `aa` and `bb` are always different than 2.

Iteration 1

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	0	NT	
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	0	NT	
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	0	NT	
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	0	NT	
0x0040		…	0	NT	

Iteration 1

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	1	NT	1
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	0	NT	
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	0	NT	
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	0	NT	
0x0040		…	0	NT	

Iteration 1

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	1	NT	1
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	1	NT	1
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	0	NT	
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	0	NT	
0x0040		…	0	NT	

Iteration 1

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	1	NT	1
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	1	NT	1
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	1	NT	1
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	1	NT	1
0x0040		…	0	NT	

Iteration 2

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	2	NT	2
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	2	NT	2
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	2	NT	2
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	2	NT	2
0x0040		…	0	NT	

Iteration 3

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	3	T	2
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	3	T	2
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	3	T	2
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	3	T	2
0x0040		…	0	NT	

Iteration 4

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	3	T	2
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	3	T	2
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	3	T	2
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	3	T	2
0x0040		…	0	NT	

Iteration 100

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	3	T	2
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	3	T	2
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	3	T	2
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	2	T	3
0x0040		…	0	NT	

BHT Example

Considering a 2-bit saturating counter BHT of 1K entries; and assuming that the processor executes the following code fragment, determine the BHT final state and calculate the misprediction ratio for the presented case.

Case 1:

-Program inputs for `aa` and `bb` are always different than 2.

Misprediton ratio =

Number of mispredicted branches / Total branches

Misprediton ratio = 9 / 400

Misprediton ratio = 2.25%

BHT Example - Case 2

Considering a 2-bit saturating counter BHT of 1K entries; and assuming that the processor executes the following code fragment, determine the BHT final state and calculate the misprediction ratio for the presented case.

Case 2:

-Program inputs for `aa` and `bb` alternate values different than 2 and equal to 2.

Case 2 - Iteration 1

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	0	NT	
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	0	NT	
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	0	NT	
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	0	NT	
0x0040		…	0	NT	

Case 2 - Iteration 1

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	1	NT	1
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	1	NT	1
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	1	NT	1
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	1	NT	1
0x0040		…	0	NT	

Case 2 - Iteration 2

```
aa = 2
bb = 2
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	0	NT	1
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	0	NT	1
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	0	NT	1
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	2	NT	2
0x0040		…	0	NT	

Case 2 - Iteration 3

```
aa = 4
bb = 3
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	1	NT	2
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	1	NT	2
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	1	NT	2
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	3	T	2
0x0040		…	0	NT	

Case 2 - Iteration 4

```
aa = 2
bb = 2
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	0	NT	2
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	0	NT	2
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	0	NT	2
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	3	T	2
0x0040		…	0	NT	

Case 2 - Iteration 100

```
aa = 2
bb = 2
```

address	Instruction		BHT	Prediction	misP. counter
0x0000	L0:	…	0	NT	
…		…	0	NT	
0x0010		DADDUI R3, R1, #-2	0	NT	
0x0014		BNEZ R3, L1	0	NT	50
0x0018		DADD R1, R0, R0	0	NT	
0x001C	L1:	DADDUI R3, R2, #-2	0	NT	
0x0020		BNEZ R3, L2	0	NT	50
0x0024		DADD R2, R0, R0	0	NT	
0x0028	L2:	DSUB R3, R1, R2	0	NT	
0x002C		BNEZ R3, L3	0	NT	50
0x0030		…	0	NT	
0x0034	L3:	…	0	NT	
0x0038		DADDI R4, R4, #-1	0	NT	
0x003C		BNEZ R4, L0	2	T	3
0x0040		…	0	NT	

BHT Example - Case 2

Considering a 2-bit saturating counter BHT of 1K entries; and assuming that the processor executes the following code fragment, determine the BHT final state and calculate the misprediction ratio for the presented case.

Case 2:

-Program inputs for `aa` and `bb` alternate values different than 2 and equal to 2.

Misprediton ratio =

Number of mispredicted branches / Total branches

Misprediton ratio = 153 / 400

Misprediton ratio = 38.25%

(2,2) Example

Considering a (2,2) correlating predictor of 1K entries; and assuming that the processor executes the following code fragment, determine the (2,2) final state and calculate the misprediction ratio for the presented case.

```
for(i=1;i<=100;i++){
  read_data();
  if (aa==2)
     aa = 0;
  if (bb==2)
     bb = 0;
  if (aa == bb)
  {…}
}
```

```
L0:   …
      DADDUI  R3, R1, #-2
      BNEZ  R3, L1
      DADD  R1, R0, R0
L1:   DADDUI  R3, R2, #-2
      BNEZ  R3, L2
      DADD  R2, R0, R0
L2:   DSUB  R3, R1, R2
      BNEZ  R3, L3
      …
L3:   …
      DADDI R4, R4, #-1
      BNEZ R4, L0
      …
```

(2,2) Example

Considering a (2,2) correlating predictor of 1K entries; and assuming that the processor executes the following code fragment, determine the (2,2) final state and calculate the misprediction ratio for the presented case.

Case 1:

-Program inputs for `aa` and `bb` alternate values different than 2 and equal to 2.

(2,2) - Iteration 1

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	00	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	0	0	0	0		
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	0	0	0		
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	0		
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	0	0	0	0		
0x0040	…	0	0	0	0		

(2,2) - Iteration 1

```
aa = 3
bb = 4
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	00	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	0	0	0	01	1
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	0	0	0		
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	0		
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	0	0	0	0		
0x0040	…	0	0	0	0		

(2,2) - Iteration 1

```
aa = 3
bb = 4
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	00	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	0	0	0	01	1
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	0	11	1
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	0		
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	0	0	0	0		
0x0040	…	0	0	0	0		

(2,2) - Iteration 1

```
aa = 3
bb = 4
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	00	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	0	0	0	01	1
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	0	11	1
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	1	11	1
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	0	0	0	1	11	1
0x0040	…	0	0	0	0		

(2,2) - Iteration 2

```
aa = 2
bb = 2
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	11	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	0	0	0	10	1
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	0	00	1
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	1	00	1
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	1	0	0	1	01	2
0x0040	…	0	0	0	0		

(2,2) - Iteration 3

```
aa = 3
bb = 4
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	01	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	1	0	0	11	2
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	1	11	2
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	2	11	2
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	1	0	0	2	11	3
0x0040	…	0	0	0	0		

(2,2) - Iteration 4

```
aa = 2
bb = 2
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	11	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	1	0	0	10	2
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	1	00	2
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	2	00	2
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	2	0	0	2	01	4
0x0040	…	0	0	0	0		

(2,2) - Iteration 5

```
aa = 3
bb = 4
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	01	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	2	0	0	11	3
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	2	11	3
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	3	11	2
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	2	0	0	3	11	4
0x0040	…	0	0	0	0		

(2,2) - Iteration 6

```
aa = 2
bb = 2
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	11	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	2	0	0	10	3
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	2	00	3
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	3	00	2
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	3	0	0	3	01	4
0x0040	…	0	0	0	0		

(2,2) - Iteration 7

```
aa = 3
bb = 4
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	01	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	3	0	0	11	3
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	3	11	3
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	3	11	2
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	3	0	0	3	11	4
0x0040	…	0	0	0	0		

(2,2) - Iteration 8

```
aa = 2
bb = 2
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	11	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	3	0	0	10	3
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	3	00	3
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	3	00	2
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	3	0	0	3	01	4
0x0040	…	0	0	0	0		

(2,2) - Iteration 100

```
aa = 2
bb = 2
```

address	Instruction	00	01	10	11	2-b sft reg	misP. counter
0x0000	L0: …	0	0	0	0	11	
…	…	0	0	0	0		
0x0010	DADDUI R3, R1, #-2	0	0	0	0		
0x0014	BNEZ R3, L1	1	3	0	0	10	3
0x0018	DADD R1, R0, R0	0	0	0	0		
0x001C	L1: DADDUI R3, R2, #-2	0	0	0	0		
0x0020	BNEZ R3, L2	0	1	0	3	00	3
0x0024	DADD R2, R0, R0	0	0	0	0		
0x0028	L2: DSUB R3, R1, R2	0	0	0	0		
0x002C	BNEZ R3, L3	0	0	0	3	00	2
0x0030	…	0	0	0	0		
0x0034	L3: …	0	0	0	0		
0x0038	DADDI R4, R4, #-1	0	0	0	0		
0x003C	BNEZ R4, L0	2	0	0	3	00	5
0x0040	…	0	0	0	0		

(2,2) Example

Considering a (2,2) correlating predictor of 1K entries; and assuming that the processor executes the following code fragment, determine the (2,2) final state and calculate the misprediction ratio for the presented case.

Case 1:

-Program inputs for `aa` and `bb` alternate values different than 2 and equal to 2.

Misprediction ratio =

Number of mispredicted branches / Total branches

Misprediton ratio = 13 / 400

Misprediton ratio = 3.25%