

假设有 N 条指令.

$$\begin{aligned}
 5. \text{CPI}_A &= \frac{0.85N \times 1 + 0.15N \times 90\% \times 90\% \times 1 + 0.15N \times 90\% \times 10\% \times 4 + 0.15N \times 10\% \times 3}{N} \\
 &= 0.85 + 0.15 \times 0.9 \times 0.9 + 0.15 \times 0.9 \times 0.1 \times 4 + 0.15 \times 0.1 \times 3 \\
 &= 1.0705 \\
 \text{CPI}_B &= \frac{0.85N \times 1 + 0.15N \times 2}{N} \\
 &= 1.15
 \end{aligned}$$

12. (1) `int a0, a4, a1, a3, a2;`

`a0 = 0; a4 = 1000;`

`for (a1 = 0; a1 < 10000; a1++)`

`{`

`a3 = a0 + 2;`

`a2 = a1 % a3;`

`if (a2 == a0)`

`code A`

`a5 = a0 + 5;`

`a2 = a1 % a3;`

`if (a2 == a0)`

`code B`

`a1++;`

`}`

(2) a_1 是偶数时

`bne a2, a0, rem2` 不跳转.

'1' 跳转比例为 50%

a_1 是 5 的倍数时

`bne a2, a0, End` 不跳转

'1' 跳转比例为: 80%

$a_1 \neq a_4$ 则 $\text{brz } a_1, a_4, \text{loop}$ 跳转
 跳转比例为: ~~99.99%~~ 从 $a_1 = 0$ 到 $a_1 = 1$ 仅最后一次不跳转 99.99%

- (3) B_1 : 正确率 50%
 B_2 : 正确率 80%
 B_3 : 正确率 0.01%

13. (1) 2

(2) 2

- (3) B_1 : 50% 稳态: $\begin{matrix} 00 & 01 & 00 & 01 & 00 & 01 \end{matrix}$
 B_2 : 80% 稳态: $\begin{matrix} 11 & 10 & 11 & 11 & 11 & 10 & 11 \end{matrix}$
 B_3 : 100% 稳态: $\begin{matrix} 11 & 11 & 11 & 11 & \dots \end{matrix}$

14.

4位

稳态时 B_1 的分支历史为 0101010101
 B_2 的分支历史为 01111011101111
 B_3 的分支历史为 11111111

若用3位, 则 B_2 111 索引的预测表应有2种可能值, 不能保证每次都正确 (1111 与 1110)

但用4位可以保证每个索引有唯一对应值
 最小4位

15. GHR 稳态时的历史为:

00111101111011010111101111

若用11位进行索引.

分支历史中有: 1111011110 1111011110 两种可能值, 不能保证每次都正确.

用12位时可保证每个索引对应单一确定值. $\therefore M$ 最小为12

16.

| | | | | |
|--------|-------|-------|---------|---------|
| | $j=0$ | $j=1$ | \dots | $j=a-1$ |
| A: 预测器 | 0 | 1 | \dots | 1 |
| | X | ✓ | \dots | X |

| | | | | |
|--|-------|-------|---------|---------|
| | $j=0$ | $j=1$ | \dots | $j=a-1$ |
| | 0 | 1 | \dots | 1 |
| | X | ✓ | \dots | X |

正确率: $\frac{a-2}{a}$

B: 稳态时 100% 预测正确.

由于所有计数器初值为 0, 即第一次用分支历史索引预测器时都不跳转, 则只正确 1 个

正确率: $\frac{1 + (P-1)Q}{PQ}$

$$\frac{a-2}{a} > \frac{1 + (P-1)Q}{PQ} \Rightarrow a - 2P > 1$$

17. (1) 假设预测器初值都为 0:

| | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|
| B ₁ : | 00 | 00 | 01 | 00 | 01 | 00 | 01 | 00 |
| | ✓ | X | ✓ | X | ✓ | X | ✓ | X |

$a_1=8-1=7, a_1=6, a_1=5, a_1=4, a_1=3, a_1=2, a_1=1, a_1=0$

| | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|
| B ₂ : | 00 | 01 | 10 | 11 | 11 | 11 | 11 | 11 |
| | X | X | ✓ | ✓ | ✓ | ✓ | ✓ | X |

总共循环 8 次.

B₁ 正确率: 50% 发生 4 次错误

B₂ 正确率: 62.5% 发生 3 次错误 共发生 4+3=7 次

若整个预测处于稳态 则:

| | | | | |
|------------------|----|----|----|----|
| B ₁ : | 00 | 01 | 00 | 01 |
| | X | ✓ | X | ✓ |
| B ₂ : | 10 | 11 | 11 | 11 |
| | ✓ | ✓ | ✓ | ✓ |

正确率仍为 50% 发生 4 次错误

11 11 11 10 正确率 87.5%

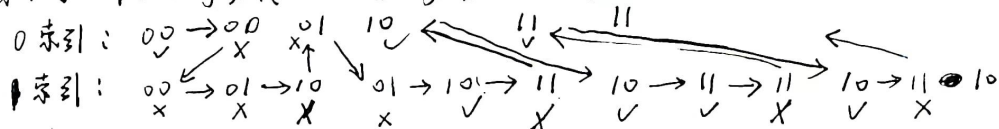
发生 1 次错误

共发生 4+1=5 次.

(2) 全局历史: 011101110110110

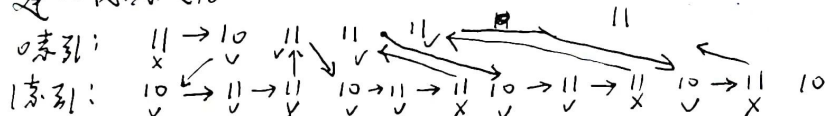
若用 1 位全局分支历史, 则只用 1 位进行索引.

当未建立稳态时, 假设 2 bit 预测器初值都为 0



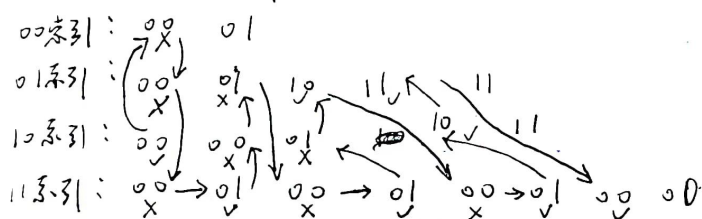
共预测错误: 9 次

建立稳态之后

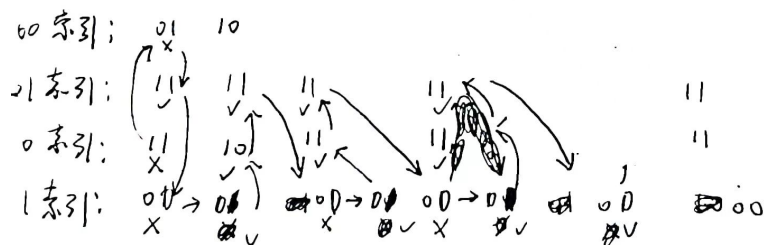


共预测错误: 5 次

(3) 用 2 位全局分支历史, 即用 2 位进行索引, 未建立稳态时:



共预测错误 8 次



共预测错误 5 次

当 00 索引的值为 10 或 11 时达到稳态. 此时 00 索引的预测器正确.

∴ 错误次数为 4 次 (稳态).

(4) 全局分支历史的位置越多, 预测的准确率就越高,

当 n 非常大时

若仅考虑稳态, 则 3 种预测器准确率相当

但考虑到稳态的建立过程, 所以 第一种方案最好

(5) PC 在 0 和 1 之间均等概率取随机值时,

第三种预测器最好

因为 2 位全局历史是根据最近 2 次跳转情况进行预测的, 而 PC 取 0 和 1 等概率, 所以会减少错误概率。

18、顺序流水线中, 指令虽然顺序发射但是乱序执行, 所以异常会乱序发生。

重排序缓存 (ROB) 可用来支持精确异常。

如果指令发生异常, 在指令提交阶段会触发异常处理机制。完成异常处理后, 处理器会从发生异常的指令位置重新执行。ROB 是顺序提交指令的, 可以保证实现精确异常。

出现分支预测错误时, ROB 将错误分支后的所有 ROB 表项清空, 而该分支之前 ROB 表项仍有效, 处理器从正确分支处提取指令, 完成恢复操作。

处理器发生异常, 一般在相应指令准备提交时才会对异常进行处理。发现异常后, 相应指令之后 ROB 全清空。异常处理完成后再从正确位置继续执行程序。

20.

| (1) | Decode | Issue | WB | Committed | 操作码 | 目标 | 源1 | 源2 |
|----------------|--------|-------|----|-----------|--------|----------------|----------------|----------------|
| I ₁ | 0 | 1 | 2 | 3 | fld | T ₀ | a ₀ | — |
| I ₂ | 1 | 3 | 13 | 14 | fmul.d | T ₁ | T ₀ | f ₀ |
| I ₃ | 2 | 14 | 16 | 17 | fadd.d | T ₂ | T ₁ | f ₀ |
| I ₄ | 3 | 4 | 6 | 18 | addi | a ₀ | a ₀ | — |
| I ₅ | 4 | 6 | 7 | 19 | fld | T ₃ | a ₀ | — |
| I ₆ | 5 | 8 | 18 | 20 | fmul.d | T ₄ | T ₃ | T ₃ |
| I ₇ | 6 | 19 | 21 | 22 | fadd.d | T ₅ | T ₄ | T ₂ |

| (2) | Decode | Issue | WB | Committed | 操作码 | 目标 | 源1 | 源2 |
|----------------|--------|-------|----|-----------|--------|----------------|----------------|----------------|
| I ₁ | 0 | 1 | 2 | 3 | fld | T ₀ | a ₀ | — |
| I ₂ | 1 | 3 | 13 | 14 | fmul.d | T ₁ | T ₀ | f ₀ |
| I ₃ | 4 | 14 | 16 | 17 | fadd.d | T ₂ | T ₁ | f ₀ |
| I ₄ | 15 | 16 | 18 | 19 | addi | a ₀ | a ₀ | — |
| I ₅ | 18 | 19 | 20 | 21 | fld | T ₃ | a ₀ | — |
| I ₆ | 20 | 21 | 31 | 32 | fmul.d | T ₄ | T ₃ | T ₃ |
| I ₇ | 22 | 32 | 34 | 35 | fadd.d | T ₅ | T ₄ | T ₂ |