CSE 2021
Computer Organization

Hugh Chesser, CSEB 1012U

Agenda

Topics:

1. Control Hazards – complete
2. Study Suggestions

Patterson: 4.8

Control Hazards

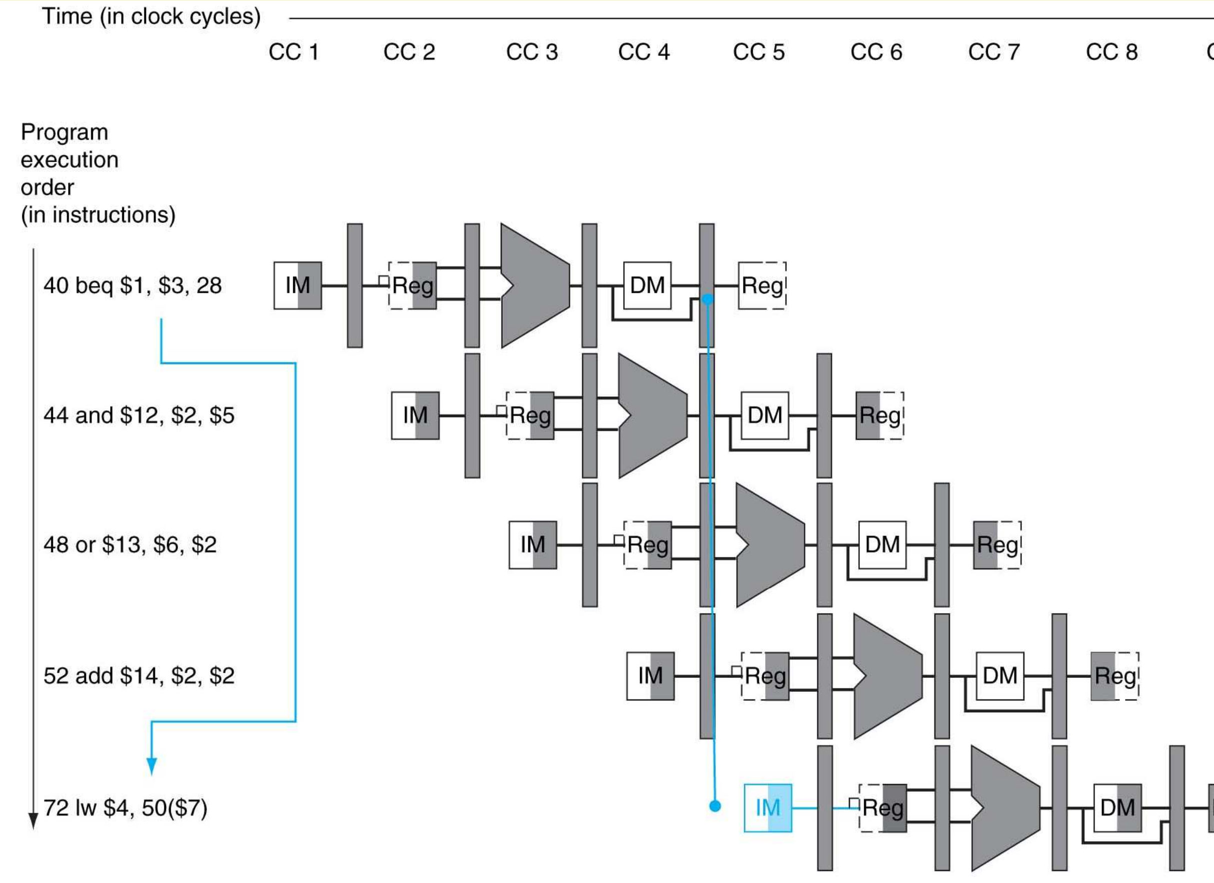
Three methods to minimize stalls

- Always assume branch not taken – must “flush” results if the branch IS taken
- Reduce branch delays – separate branch adder to calculate branch target address, move execution (test) earlier (“equality unit” – XOR)
- Dynamic branch prediction – remember if the branch was taken the last time the branch instruction was executed – branch history table

Assume Not Taken

Proceed with instructions as if branch not there

- "Flush" instructions in the pipeline if taken
- Effectively this creates a stall if branch is taken

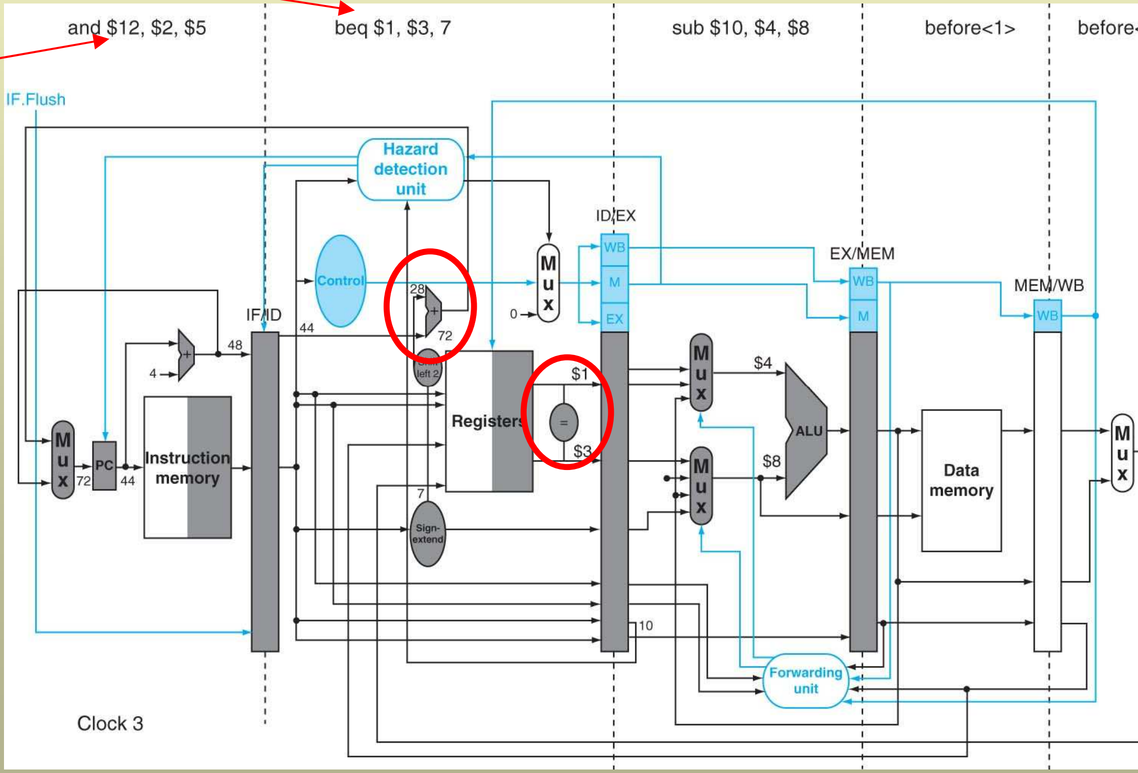


Assume Branch Not Taken – Minimize Stall

```
36 sub $10, $4, $8

40 beq $1, $3, 7

44 and $12, $2, $5

…

72 lw  $4, 50($7)
```

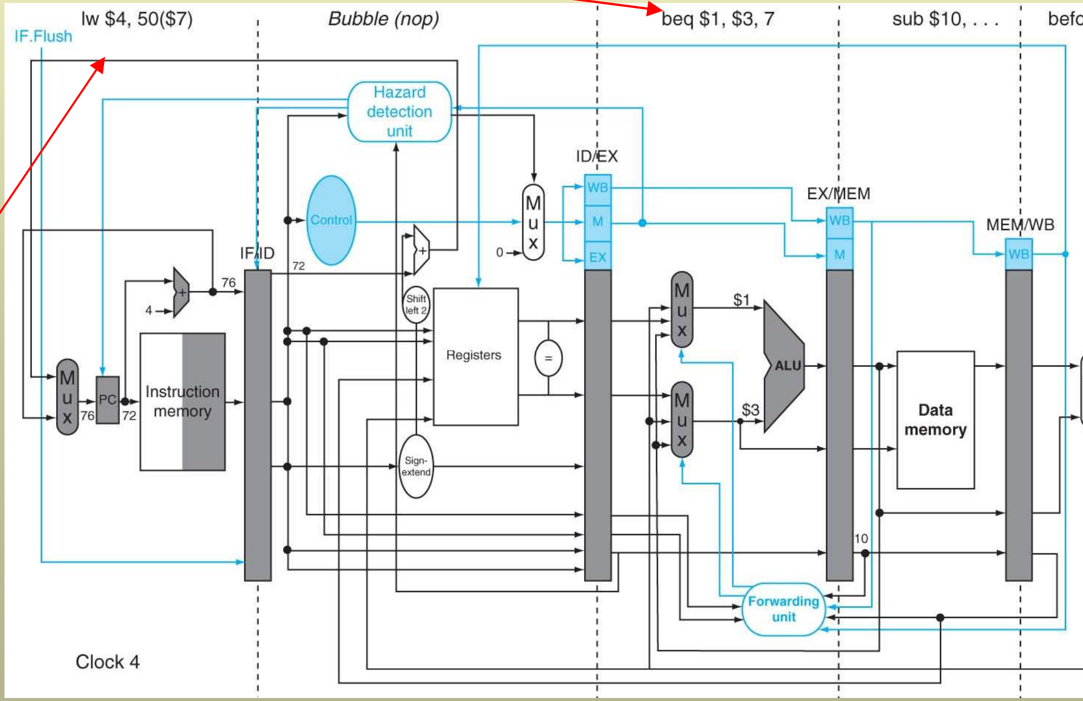
$PC \leftarrow PC + 4 + \text{BranchAddr} = 40 + 4 + 7 \cdot 4 = 72$

Branch adder and decision logic added to minimize branch stall – branch decided during ID phase



Branch to be taken – Flush current pipeline instructions

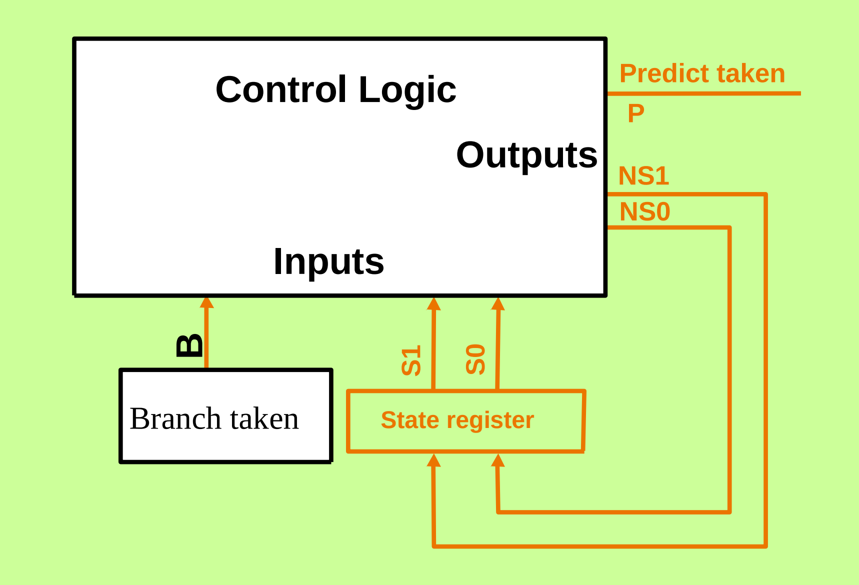
```
36 sub $10, $4, $8

40 beq $1, $3, 7

44 and $12, $2, $5

…

72 lw  $4, 50($7)
```

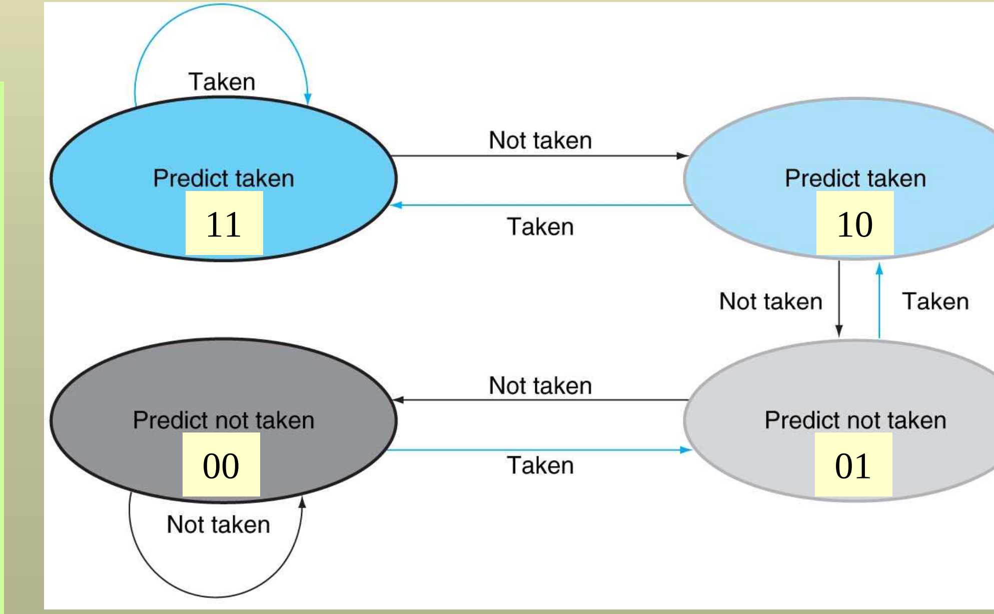


Dynamic Branch Prediction

Prediction of branches while the program is executing

- A portion of memory is utilized which indicates whether or not the branch was taken last time the instruction was executed
- FSM implementation is…

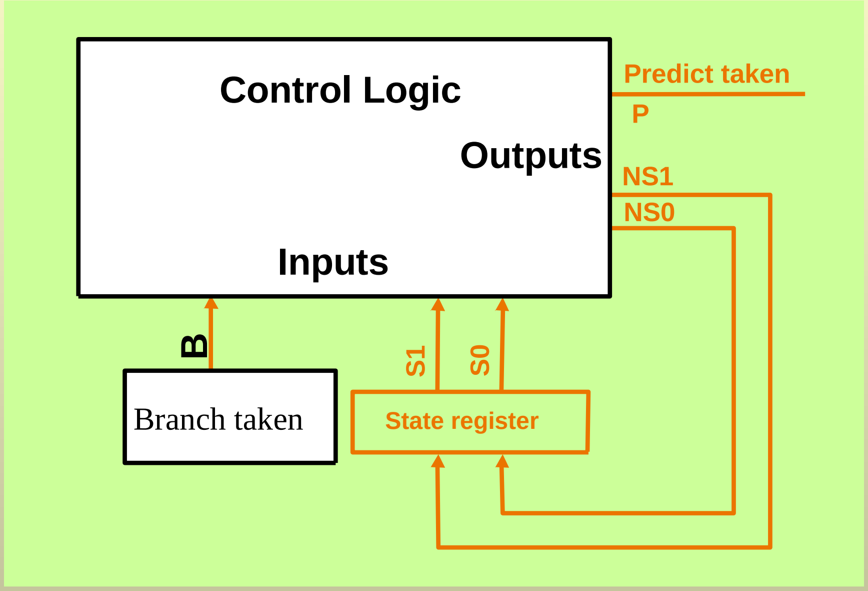




Finite State Machine

Control of Branch Prediction

Branch Taken	*State*	*Next State*	*Predict taken*
0	*00*	*00*	*0*
1	*00*	*01*	*0*
0	*01*	*00*	*0*
1	*01*	*10*	*0*
0	*10*	*01*	*1*
1	*10*	*11*	*1*
0	*11*	*10*	*1*
1	*11*	*11*	*1*



Exam Study Suggestions

Do practice questions! Do NOT simply read the textbook

Questions are available in the back of the chapters that are on the exam:

- Chapters 1, 2, 3, 4 (up to and including section 4.8), Appendix B, C, D – I have some solutions if stuck
- Practice with spim and iVerilog