PART 1

(1) ASSUME THAT WE ONLY CONSIDER THE FOLLOWING LATENCIES FOR ELEMENTS IN A DATAPATH AS IN FIGURE 4.17 OF THE NOTEBOOK. ANSWER THE FOLLOWING QUESTIONS AND SHOW YOUR CALCULATION TO GET FULL CREDIT.

| I-MEM | REGS | ALU | D-MEM | SIGN-EXTEND |
|---|---|---|---|---|
| 400 ps | 200 ps | 150 ps | 450 ps | 60 ps |

(1.1) HOW LONG DOES IT TAKE TO COMPLETE THE EXECUTION OF A BEQ INSTRUCTION?

BEQ INSTRUCTION USES I-MEM, REGS AND ALU DATAPATH ELEMENTS ONLY, SO, THE TOTAL TIME IS:

EXECUTION TIME = 400 ps + 200 ps + 150 ps = 750 ps

(1.2) HOW LONG DOES IT TAKE TO COMPLETE THE EXECUTION OF A SUB INSTRUCTION?

SUB INSTRUCTION USES I-MEM, REGS, ALU AND SIGN-EXTEND DATAPATH ELEMENTS ONLY, SO, THE TOTAL TIME IS:

EXECUTION TIME = 400 ps + 200 ps + 150 ps + 60 ps = 810 ps

(1.3) HOW LONG DOES IT TAKE TO COMPLETE THE EXECUTION OF AN LW INSTRUCTION?

LW INSTRUCTION USES ALL DATAPATH ELEMENTS, SO, THE TOTAL TIME IS:

EXECUTION TIME = 400 ps + 200 ps + 150 ps + 450 ps + 60 ps = 1260 ps

tilibra

1.4 SUPPOSE THAT WE ONLY CONSIDER BEQ, SUB AND LW INSTRUCTIONS AND USE THE SAME CLOCK CYCLE FOR ALL THREE. IF WE CAN REDUCE THE LATENCY OF ONLY ONE GIVEN DATAPATH ELEMENT BY 25%, WHICH ELEMENT SHOULD WE PICK? WHY?

WE SHOULD PICK THE D-MEM ELEMENT. BECAUSE IT HAS THE BIGGEST LATENCY, IF WE REDUCE THIS LATENCY IN 25% WE GET $450(1 - 0.25) = 450(0.75) = 337.5$ ps OF LATENCY, SO, THE BIGGEST LATENCY WILL BE 400 ps FROM I-MEM AND MUST BE TAKEN AS THE CLOCK CYCLE TIME BECAUSE THIS WILL BE THE STAGE THAT TAKES MORE TIME TO EXECUTE, IF WE TAKE ANOTHER BELOW THE 400 ps, THAT TIME WON'T BE ENOUGH TO EXECUTE THE I-MEM STAGE, AND THE TIME THAT AN INSTRUCTION TAKES TO EXECUTE WILL BE $5 \times 400 = 2000$ ps. IF WE DON'T REDUCE THE LATENCY OF D-MEM STAGE, WE WILL NEED TO TAKE 450 ps AS THE TIME TO THE CLOCK CYCLE TIME, BECAUSE THAT WILL BE THE STAGE THAT TAKES MORE TIME TO EXECUTE AND AND INSTRUCTION WILL EXECUTE IN $5 \times 450$ ps = 2250 ps RESULTING IN MORE TIME INSTRUCTION EXECUTION

2) GIVEN THE FOLLOWING SEQUENCE OF INSTRUCTIONS TO BE EXECUTED ON A 5-STAGE PIPELINED DATAPATH AS DESCRIBED IN OUR TEXTBOOK:

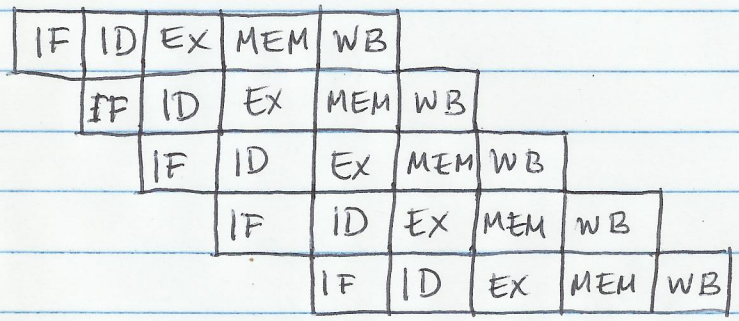
I0: ADD $8, $9, $10
I1: ADD $11, $11, $8
I2: LW $8, 0($9)
I3: OR $8, $8, $10
I4: SW $11, 0($8)

tilibra

/ /

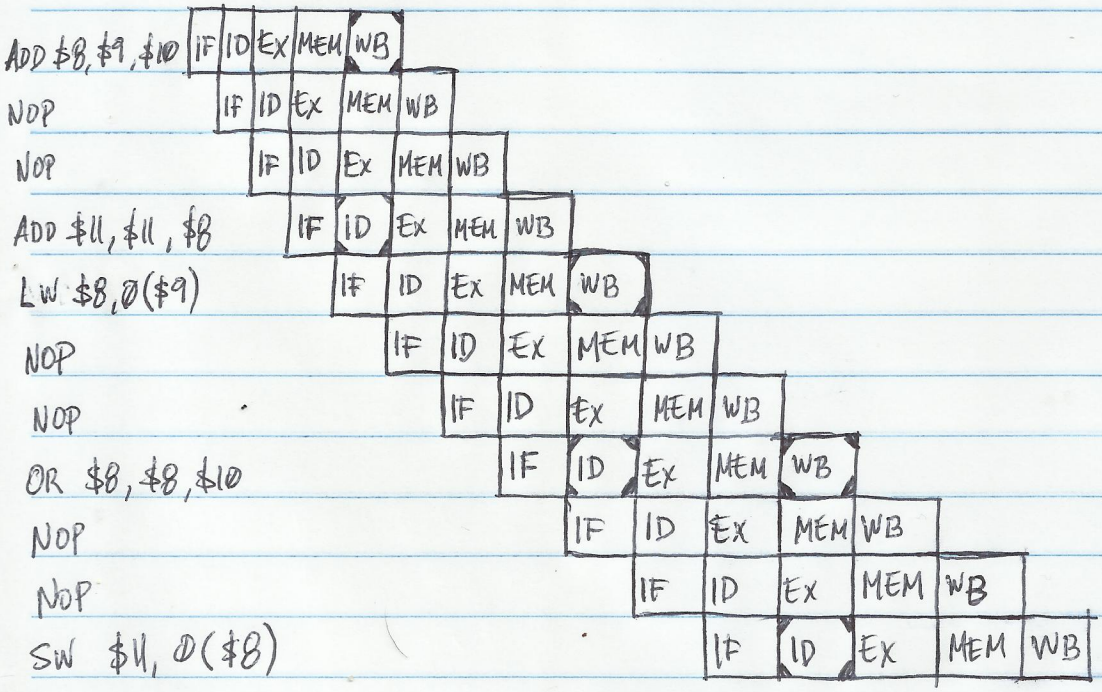
(2.1) LIST TRUE DEPENDENCIES IN THE GIVEN SEQUENCE IN THE FORMAT OF (REGISTER_INVOLVED, PRODUCER_INSTRUCTION, CONSUMER_INSTRUCTION). USE LABELS TO INDICATE INSTRUCTIONS

I0: ADD $8, $9, $10
I1: ADD $11, $11, $8
I2: LW $8, 0($9)
I3: OR $8, $8, $10
I4: SW $11, 0($8)




($8, I0, I1)
($8, I2, I3)
($8, I3, I4)
($11, I1, I4)

(2.2) IF THERE IS NO FORWARDING AND NO REORDERING, INSERT NOPS TO ENSURE CORRECT EXECUTION. SHOW THE SEQUENCE OF EXECUTION WITH NOPS

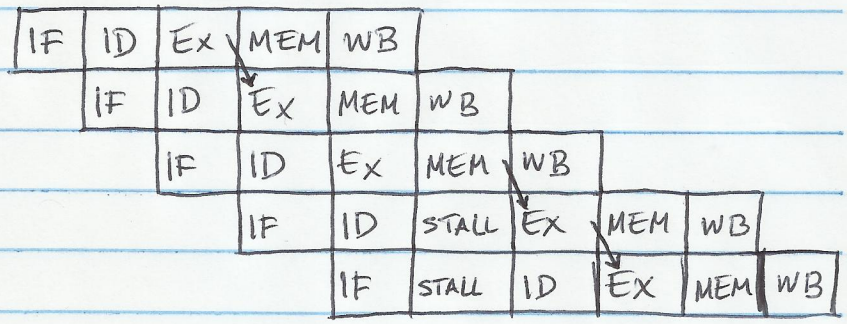



tilibra

1 1

(2.3) IF THERE IS FULL FORWARDING SUPPORT, DRAW MULTIPLE-CYCLED PIPELINE DIAGRAM (LIKE FIGURE 4.44) TO SHOW THE EXECUTION OF THE SEQUENCE. USE ARROWS TO MARK FORWARDINGS CLEARLY IN YOUR DIAGRAM. EACH ARROW SHOULD POINT FROM INSTRUCTION/STAGE HANDING OFF THE DATA → INSTRUCTION/STAGE RECEIVING THE DATA. ALSO MARK THE NECESSARY PIPELINE STALLS.

I0: ADD $8, $9, $10
I1: ADD $11, $11, $8
I2: LW $8, 0($9)
I3: OR $8, $8, $10
I4: SW $11, 0($8)




(3) THIS EXERCISE EXAMINES THE ACCURACY OF VARIOUS BRANCH PREDICTORS

(3.1) CONSIDER THE BRANCH SEQUENCE: NT, T, T, NT, T, T. WHAT IS THE ACCURACY OF ALWAYS-NOT-TAKEN PREDICTOR FOR THIS SEQUENCE?

NT, T, T, NT, T, T
NT, NT, NT, NT, NT, NT
✓ X X ✓ X X
TOTAL X: 4
TOTAL ✓: 2

ACCURACY:

$$\frac{\text{TOTAL} \checkmark}{\text{TOTAL} \checkmark + \text{TOTAL} \times} = \frac{2}{2+4} = \frac{2}{6} = \frac{1}{3} = 0{,}3333$$

ACCURACY = 33,33%

(3.2) FILL IN THE TABLE BELOW TO SHOW THE STATUS TRANSITION/PREDICTION OF A TWO-BIT PREDICTOR FOR THE SAME BRANCH SEQUENCE. ASSUME THAT THE PREDICTOR STARTS OFF IN THE TOP RIGHT STATE FROM FIGURE 4.63 ((WEAK) PREDICT TAKEN)

| BRANCH BEHAVIOR | NT | T | T | NT | T | T |
|---|---|---|---|---|---|---|
| PREDICTOR STATUS | WEAK T | WEAK NT | WEAK T | T | WEAK T | T |
| PREDICTION | T | NT | T | T | T | T |
| CORRECT PREDICTION? | NO | NO | YES | NO | YES | YES |

tilibra

3.3 WHAT IS THE ACCURACY OF THIS TWO-BIT PREDICTOR FOR THE GIVEN SEQUENCE OF 6 BRANCHES BASED ON YOUR TABLE ABOVE? WHAT IS THE ACCURACY IF THE SAME BRANCH SEQUENCE REPEATS FOREVER? SHOW YOUR CALCULATION

ACCURACY FOR THE FIRST 6 BRANCHES:

THERE ARE 3 YES AND 3 NO TO THE CORRECT PREDICTION ANSWER, SO, THE ACCURACY IS 50%

IF THE BRANCH SEQUENCE REPEATS FOREVER:

THE NEXT TABLE FOLLOWS THE 3.2 QUESTION TABLE

| BRANCH BEHAVIOR | NT | T | T | NT | T | T | NT | T | T | NT | T | T |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PREDICTOR STATUS | T | WT | T | T | WT | T | T | WT | T | T | WT | T |
| PREDICTION | T | T | T | T | T | T | T | T | T | T | T | T |
| CORRECT PREDICTION? | NO | YES | YES | NO | YES | YES | NO | YES | YES | NO | YES | YES |

THIS REPEATS

ACCURACY IF THE BRANCH SEQUENCE REPEATS FOREVER:

THE TABLE ABOVE SHOWS THAT PREDICTION IS CORRECT 2 OF 3 TIMES, EVERY 3 PREDICTIONS, 2 ARE CORRECT. THEN, IF THE BRANCH SEQUENCE REPEATS FOREVER, THE ACCURACY WILL BE NEAR TO THE 66,67 %

tilibra