HW 4

ID: 111062107

Name: 鄧弘鈞

1.

(a) sd 、ld 、beq
(since they need immediate)

(b) add 、beq

(c) Branch, MemRead, RegWrite) 0 when run sd X_{13} 0(X_{12})

If Branch = 1, PC+4 → PC + imm, 且 ∵ imm = 0, PC = PC + 0 導致無限循環。If MemRead = 1, 則 MemRead、MemWrite 都 = 1, 則 memory 會同時讀寫, 根據 X12 + 0 位置輸出 Readdata 且 ∵ RegWrite = 1, Readdata 會寫入 Register File, 但是 ∵ ins[11:7] = 0, 且 X_0 不允許寫入, 所以不會有影響。

(b) 820 ps

since minimum clock period should be able to execute every instruction, and the longest instruction which is "ld" takes 820 ps, so the clock period should be 820 ps.

2.

$0|EEA5z3_{16}$

$= \underbrace{0000\ 000}_{\text{im[11:5]}}\underbrace{1|1110}_{\text{rs2}}\underbrace{1110}_{\text{rs1}}\underbrace{|0|0}_{\text{fnc3}}\underbrace{0|0|}_{\text{im[4:0]}}\underbrace{00|0\ 00|1}_{\text{opcode}}{}_2$

im[11:5] → 0; rs2 → 30; rs1 → 29; fnc3 → SW; im[4:0] → 10

sw X_{30}, 10(X_{29})

(a) Branch: 0
MemRead: 0
MemtoReg: X
ALUOp: 00
MemWrite: 1
ALUSrc: 1
RegWrite: 0

(b) Reg[29], 10_{10} (= 0000 0000 0000 000A $_{hex}$)

3.

(a) "add" is the least time consuming one, it need 600 ps.

add: 25 + 235 + 130 + 15 + 190 + 15 + 10
PC Read, I-Mem, Register File, MUX, ALU, Writing Mux, Register Setup
= 600 ps

ld: 25 + 235 + 130 + 190 + 235 + 15 + 10
PC Read, I-Mem, Register File, ALU, D-Mem, MUX, Register Setup
= 820 ps

sd: 25 + 235 + 130 + 190 + 235 = 795 ps
PC Read, I-Mem, Register File, ALU, D-Mem

beq: 25 + 235 + 130 + 190 + 15 + 15 + 5 + 10
PC Read, I-Mem, Register File, ALU, Mux, Mux, Single Gate, Register Setup
= 605 ps

4.

Original time

$$\frac{1200 \times 4}{4 \times 10^9} \text{ (sec)}$$

$$= 12 \times 10^{-7} \text{ (sec)}$$

improved time

# of clocks = 1200 + 4
= 1204

$$\frac{1204}{2 \times 10^9} = 602 \times 10^{-9}$$

$$\text{speedup} = \frac{12 \times 10^{-7}}{602 \times 10^{-9}}$$

$$\approx 0.0199 \times 10^2$$

$$= 1.99 \ \#$$

5.

(a) RAW

① add X_{29}, X_6, X_7
sub X_{30}, X_{28}, X_{29}

RAW

② sub X_{30}, X_{28}, X_{29}
sd X_{30}, 0(X_{11})

load-use

③ ld X_{28}, 4(X_5)
....
sub X_{30}, X_{28}, X_{29}

(b) No since load-use hazard will not occur.

since ld: IF ID EX MEM WB
add: " " " " "
sub: " " EX " "

so no hazard that will require the pipeline to stall.

(c)

ld X28, 4(X5)	IF	ID	EX	MEM	WB			
add X29, X6, X7		IF	ID	EX	MEM	WB		
stall			X	X	X	X	X	
stall				X	X	X	X	X
sub X30, X28, X29					IF	ID	EX	MEM	WB
stall						X	X	X	X	X
stall							X	X	X	X
sd X30, 0(X11)								IF	ID	EX

4 NOPs #

6.

(a)

always - not - taken

$$\text{accuracy} = \frac{5}{8} = 0.625 = 62.5\ \%$$

always - taken

$$\text{accuracy} = \frac{3}{8} = 0.375 = 37.5\ \%$$

(b) starts at T

Ground truth	NT	T	T	NT	NT	NT	NT	T
State	T	NT	T	T	NT	NT	NT	NT
Decision	T	NT	T	T	NT	NT	NT	NT
Correctness	X	X	✓	X	✓	✓	✓	X

$$\text{accuracy} = \frac{4}{8} = 50\ \%$$

(c)

Ground truth	NT	T	T	NT	NT	NT	NT	T
State	ST	WT	ST	ST	WT	WN	SN	SN
Decision	T	T	T	T	T	NT	NT	NT
Correctness	X	✓	✓	X	X	✓	✓	X

ST : strongly predict taken
WT : weakly predict taken
WN : weakly predict not taken
SN : strongly predict not taken

$$\text{accuracy} = \frac{4}{8} = 50\ \%$$

7.

(7+4)

(a) 11 clock cycles, NO #

```
beq  X11, X12, Label
sd   X15, 0(X23)
ld   X15, 0(X24)
add  X11, X6, X12
NOP
NOP
sub  X11, X13, X12
```

若新增 NOP 指令，因 NOP 位於記憶體內，故在 fetch NOP 時，也會有 structural hazard。

∴ NO #

(b)

```
beq  X11, X12, Label
add  X11, X6, X12
sub  X11, X13, X12
sd   X15, 0(X23)
ld   X15, 0(X24)
```

9 cycles (5 + 4) #

(c) $\frac{(4+4+4)}{(2+4+4)} = 1.2$ #

if determined in ID, should add 1 NOP between beq, sd
if " " MEM, " " 3 NOPs " beq, sd

(d) The original result takes 5+5-1 = 9 cycles.
The new result with only 4 stages takes 5+4-1 = 8 cycles. speedup = 9/8 = 1.125 #

(e) The clock period of original pipelined process is 220.
The new pipelined process has a clock period of max(220, 180) + 25 = 245. The speedup is $\frac{220 \times 9}{245 \times 8}$

≈ 1.0102 #

8.

	IF	ID	EX	MEM	WB
Cycle 1	add X12, X6, X5	-	-	-	-
Cycle 2	sub X10, X11, X12	add X12, X6, X5	-	-	-
Cycle 3	beq X11, X12, label	sub X10, X11, X12	add X12, X6, X5	-	-
Cycle 4	sd X11, 0(X12)	beq X11, X12, label	sub X10, X11, X12	add X12, X6, X5	-
Cycle 5	first instruction of the exception handler	NOP	NOP	sub X10, X11, X12	add X12, X6, X5