Bryant Chapter 4

4.13

Stage	Generic `irmovq` V, rB	Specific `irmovq $128, %rsp`
Fetch	icode:ifun ← M_1[PC]	icode:ifun ← M_1[0x016] = 0x3:0x0
	rA:rB ← M_1[PC+1]	rA:rB ← M_1[0x017] = 0xF:0x4
	valC ← M_8[PC+2]	valC ← M_8[0x018] = 0x80
	valP ← PC + 10	valP ← 0x016 + 0xA = 0x020
Decode		
Execute	valE ← 0 + valC	valE ← 0x0 + 0x80 = 0x80
Memory		
Write back	R[rB] ← valE	R[%rsp] ← 0x80
PC update	PC ← valP	PC ← valP = 0x020

4.14

Stage	Generic `popq` rA	Specific `popq %rax`
Fetch	icode:ifun ← M_1[PC]	icode:ifun ← M_1[0x02C] = 0xB:0x0
	rA:rB ← M_1[PC+1]	rA:rB ← M_1[0x02D] = 0x0:0xF
	valP ← PC + 2	valP ← 0x02C + 0x2 = 0x02E
Decode	valA ← R[%rsp]	
	valB ← R[%rsp]	
Execute	valE ← valB + 8	
Memory	valM ← M_8[valA]	
Write back	R[%rsp] ← valE	
	R[rA] ← valM	
PC update	PC ← valP	

4.43

Bryant Chapter 6

6.12

6.17

6.18