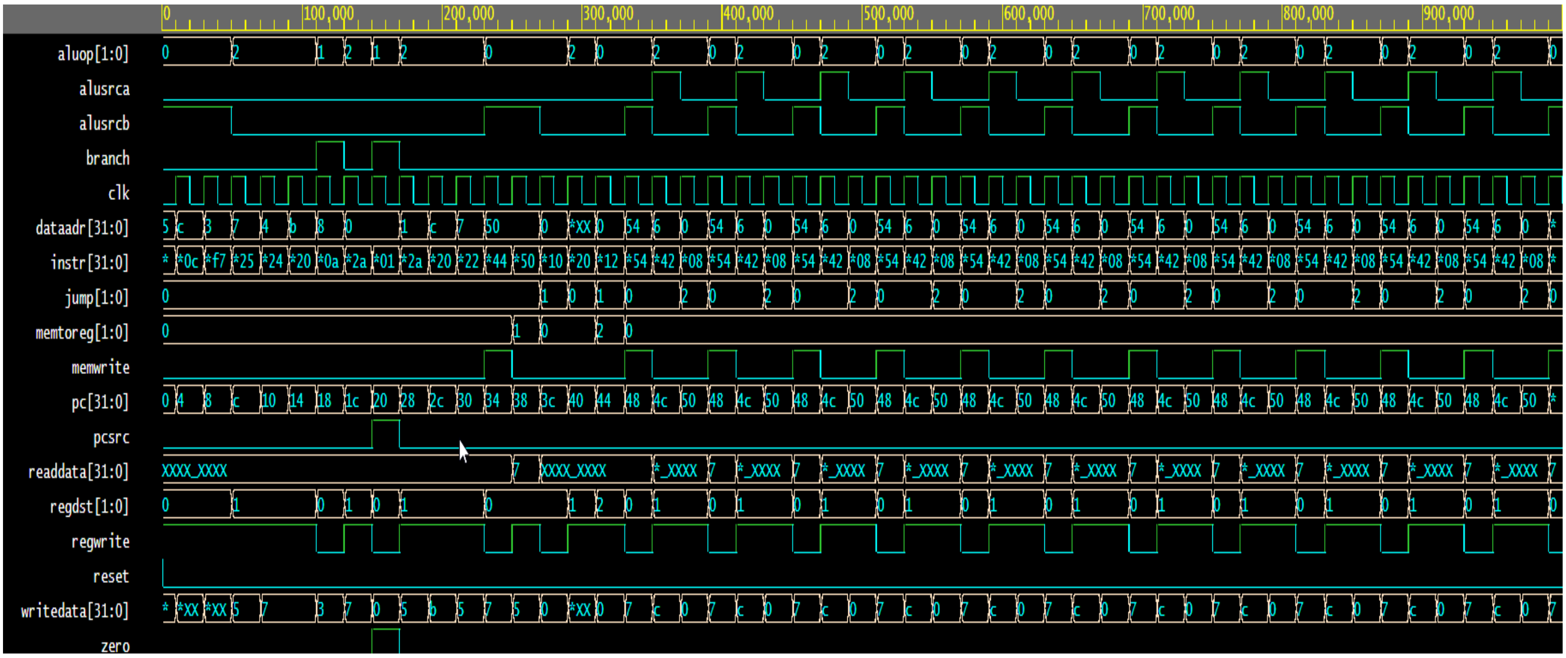
CS224 SPRING 2021 PRELIM REPORT

CS224
Lab No. 04
Section No. 05
Your Full Name Muhammed Ikbal Doğan
Bilkent ID 21702990

Part 1.a

ADRESS	MACHINE INSTRUCTION	ASSEMBLY EQUIVILENT
00	0x20020005	addi $v0,$zero,0x0005
04	0x 2003000c	addi $v1,$zero,0x000C
08	0x 2067fff7	addi $a3,$v1,0xFFF7
0c	0x 00e22025	or $a0,$a3,$v0
10	0x 00642824	and $a1,$v1,$a0
14	0x 00a42820	add $a1,$a1,$a0
18	0x 10a7000a	beq $a1,$a3,0x000A
1c	0x 0064202a	slt $a0,$v1,$a0
20	0x 10800001	beq $a0,$zero,0x0001
24	0x 20050000	Addi $a1,$zero,0x0000
28	0x 00e2202a	slt $a0,$a3,$v0
2c	0x 00853820	add $a3,$a0,$a1
30	0x 00e23822	sub $a3,$a3,$v0
34	0x ac670044	sw $a3,0x0044($v1)
38	0x 8c020050	lw $v0,0x0050($zero)
3c	0x 08000010	j 0x0000010
40	0x 001f6020	add $t4,$zero,$ra
44	0x 0c000012	jal 0x0000012
48	0x ac020054	sw $v0,0x0054($zero)
4c	0x 00039042	srl $s2, $v1, 1
50	0x 03E00008	jr $ra

Part 1.e



Part 1.f

i) Write data corresponds to RD2.In R type write data corresponds to RF[rt]

ii) Because in the I type instrucions, ALUSrcB becomes 1 and write data is not used, signimm is used.

iii) Because read data is used when loading Word from the memory (lw) therefore it is used when lw instruction called only.

iv) it corresponds to alu result of rs and rt but dataaddr is not used in r type instructions.

v)If there is a jump instruction or j-type instruction, dataadr is not being used therefore it is undefined

Part 1.g

i) There is no need fort he data path because it has the path already but it should be added to main decoder

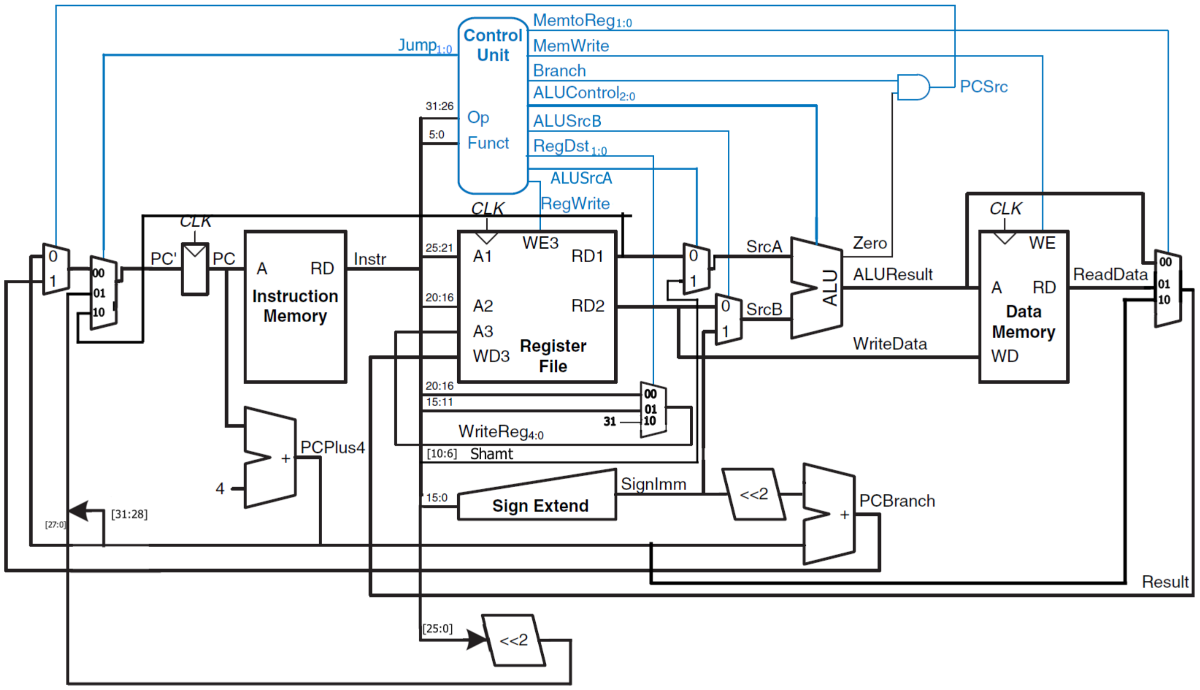
ii) I would set wire rd2 to left shifter and and connect it into the multiplexer which is goingto a3. Then i would define 11 to regDst to select it

Part 2.a

RTL for bge

```
IM[PC]
 if RF[rs] >= RF[rt]
        PC <= PC+4 + (SignExt(imm16) x4)
else
PC <= PC+4
```

Part 2.b



There is no need to add additional components or change to datapath because bge works with the zero and branch signal perfectly therefore by using them, we can perform bge.

Part 2.c

Instruction	OpCode	RegWrite	RegDst	ALUSrcA	ALUSrcB	Branch	MemWrite	MemtoReg	ALU Op	jUMP
R-type	000000	1	01	0	0	0	0	00	10	00
Srl	000000	1	01	1	0	0	0	00	10	00
Lw	100011	1	00	0	1	0	0	01	00	00
Sw	101011	0	X	0	1	0	1	XX	00	00
Beq	000100	0	X	0	0	1	0	01	01	00
Addi	001000	1	00	0	1	0	0	00	00	00
J	000010	0	X	X	X	X	0	XX	XX	01
Jal	000011	1	10	X	X	X	0	10	XX	01
Jr	000000	1	01	0	0	0	0	00	10	10
bge	111111	0	X	0	0	1	0	X	10	00