Bilkent University

CS Department

CS 224 - Digital Design and Computer Architecture

Preliminary Design Report Lab 04

Section 04

Okan Şen 21202377

28/11/2019

b)

hex	assembly	location
32'h20020005	addi $v0, $zero, 5	0x00000000
32'h2003000c	addi $v1, $zero, 12	0x00000004
32'h2067fff7	addi $a3, $v1, -9	0x00000008
32'h00e22025	or $a0, $a3, $v0	0x0000000C
32'h00642824	and $a1, $v1, $a0	0x00000010
32'h00a42820	add $a1, $a1, $a0	0x00000014
32'h10a7000a	beq $a1, $a3, 10	0x00000018
32'h0064202a	slt $a0, $v1, $a0	0x0000001C
32'h10800001	beq $a0, $zero, 1	0x00000020
32'h20050000	addi $a1, $zero, 0	0x00000024

32'h00e2202a	slt $a0, $a3, $v0	0x00000028
32'h00853820	add $a3, $a0, $a1	0x0000002C
32'h00e23822	sub $a3, $a3, $v0	0x00000030
32'hac670044	sw $a3, 0x0044, $v1	0x00000034
32'h8c020050	lw $v0, 0x0050, $zero	0x00000038

32'h08000011	j 0x0000011	0x0000003C
32'h20020001	addi $v0, $zero, 0x0001	0x00000040
32'hac020054	sw $v0, 54($zero)	0x00000044
32'h08000012	j 0x0000012	0x00000048

c)

sw+

IM[PC]

DM[R[rs] + SingExt(Imm) <- R[rt]]

R[rs] <- R[rs] + 4

PC <- PC + 4

bge

IM[PC]

if (RF[rs] < RF[rt])

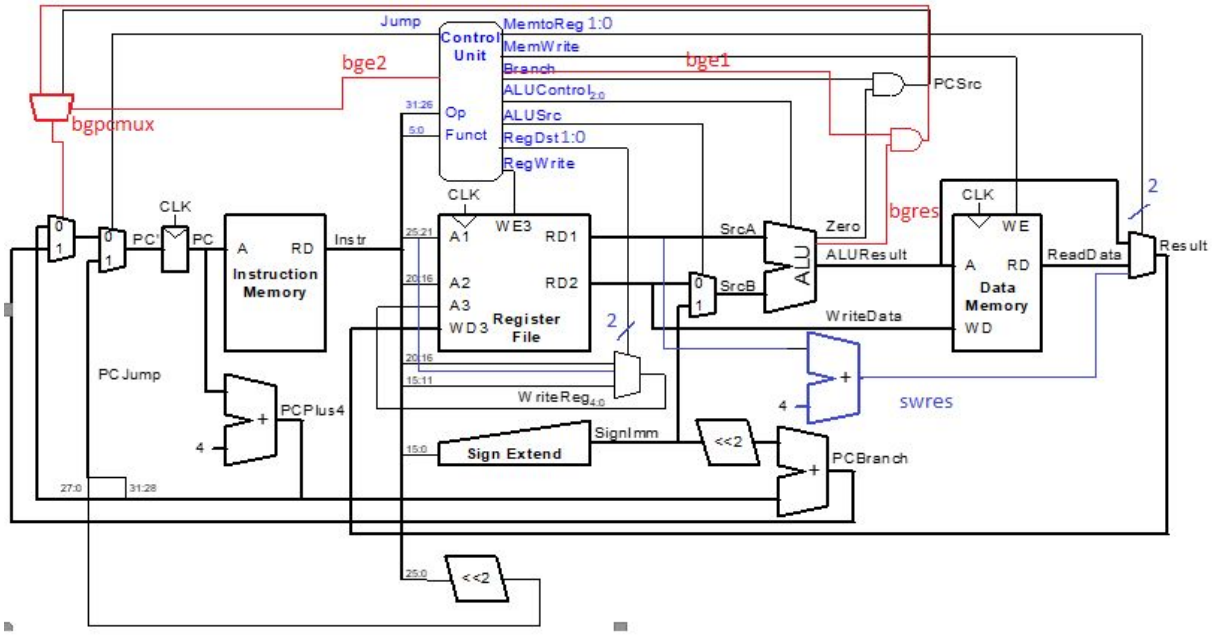
PC <- PC + 4

else

PC <- BTA

d) sw+ is in blue. MemtoReg and RegDst are increased to 2 bits to handle mux3's that are converted from mux2's. They are the selection inputs.

Bge is in red. Bgpcmux is the name of the new added mux2 on the top left are, selection input is named bge2.



e)

Only main decoder truth table was edited, the alu decoder truth table stays the same.

Instruc-tion	$Op_{5:0}$	Reg Write 1:0	Reg Dst	Alu Src	Branch	Mem Write 1:0	Mem toReg	ALU $Op_{1:0}$	Jump	bge1	bge2
R-type	000000	01	1	0	0	00	0	1x	0	0	0
lw	100011	01	0	1	0	00	1	00	0	0	0

sw	101011	00	X	1	0	01	X	00	0	0	0
beq	000100	00	X	0	1	00	X	01	0	0	0
j	000010	00	X	X	X	00	X	XX	1	0	0
addi	001000	01	0	1	0	00	0	00	0	0	0

Instruc-tion	$Op_{5:0}$	Reg Write	Reg Dst 1:0	Alu Src	Branch	Mem Write	Mem toReg 1:0	ALU $Op_{1:0}$	Jump	bge1	bge2
sw+	111110	1	10	1	0	1	10	00	0	0	0
bge	111111	0	00	0	0	0	00	01	0	1	1

f)
Hexes of sw+ and bge, respectively;
0 x FA 08 00 00
0 x FD 09 00 04

Mips code, from hex conversions:

```
        .text
        .globl __start

__start:
        sdc2 $t0, 0($s0)
        sd $t1, $t0, label
```

```
        add $t5, $t5, $zero

label:

        li $v0,10
        syscall
```