Zainab Hussein Lab 7 Prelab 4-4-2017

Cycle	reset	pc	instr	branch	srca	srcb	aluout	zero	pcsrc	writedata	memwrite	read data
1	1	00	addi $2,$0,5 (0x20020005)	0	0	5	5	0	0	0	0	x
2	0	04	addi $3,$0,12 (0x2003000c)	0	0	c	c	0	0	0	0	x
3	0	08	addi $7,$3,-9 (0x2067fff7)	0	c	-9	3	0	0	0	0	x
4	0	0C	Or $4,$7, $2 (0x00e22025)	0	3	5	7	0	0	0	0	X
5	0	10	And $5, $3, $4 (0x00642824)	0	C	7	4	0	0	0	0	X
6	0	14	Add $5, $3, $4 (0x00a42820)	0	4	7	b	0	0	0	0	X
7	0	18	Beq $5, $7, end (0x10a7000a)	1	B	3	8	0	0	0	0	X
8	0	1C	Slt $4, $3, $4 (0x0064202a)	0	C	7	0	1	0	0	0	X
9	0	20	Beq $4, $0, loop (0x10800001)	1	0	0	0	1	1	0	0	X
10	0	24	addi $5,$0, 0 (nop) (0x20050000)									
10	0	28	Slt $4, $7, $2 (0x00e2202a)	0	3	5	1	0	0	0	0	X
11	0	2C	Add $7, $4, $5 (0x00853820)	0	1	8	9	0	0	0	0	X
12	0	30	Sub $7, $7, $2 (0x00e23822)	0	9	5	4	0	0	0	0	X
13	0	34	Sw $7, 68($3) (0xac670044)	0	C	68	80	0	0	4	1	X
14	0	38	Lw $2, 80($0) (0x8c020050)	0	0	80	80	0	0	0	0	X
15	0	3C	J end (0x08000011)	0	X	X	X	X	0	0	0	X

16	0	40	Addi $2,$0, 1 (not excecute after jump) (0x20020001)									
16	0	44	Sw $2, 84($0) (0xac020054)	0	0	84	84	0	0	80	1	X

Table 1.

First sixteen cycles of executing mipstest.asm

Extended functionality. Main Decoder:

Instruction	$Op_{5:0}$	RegWrite	RegDst	AluSrc	Branch	MemWrite	MemtoReg	$ALUOp_{1:0}$	Jump	Bne
R-type	000000	1	1	0	0	0	0	10	0	0
lw	100011	1	0	1	0	0	1	00	0	0
sw	101011	0	X	1	0	1	X	00	0	0
beq	000100	0	X	0	1	0	X	01	0	0
addi	001000	1	0	1	0	0	0	00	0	0
j	000010	0	X	X	X	0	X	XX	1	0
ori	001101	1	0	10	0	0	0	11	0	0
bne	000101	0	x	00	1	0	x	01	0	1

Extended functionality. ALU Decoder:

$ALUOp_{1:0}$	Meaning
00	Add
01	Subtract
10	Look at funct field
11	ori