Pre lab exp 10

1201738 Aya Dahbour

1201505 Noor Makhalfa

sec 2

Line #	Code	Description of what is done
6	**reg [15:0] Memory [0:63]**	Memory 64*16
7	**reg [2:0] state**	To manage instruction cycle steps
13	**Memory [10] = 16'h3020**	load data at address 20Hex into AC
14	**Memory [11] = 16'h7021**	add data at address 21Hex into AC
15	**Memory [12] = 16'hB014**	store data into address 14Hex
18	**Memory [32] = 16'd7**	store at address 20Hex or 32 decimal data = 7 decimal
19	**Memory [33] = 16'd5**	store at address 21Hex or 33 decimal data = 5 decimal
22	**PC = 10; state = 0**	the program should start from the address 10Decimal = 0A Hexadecimal
29	**MAR <= PC**	The address on pc sends it to MAR
30	**state = 1**	Next state, MAR Has the data
33	**IR <= Memory [MAR]**	Send data saved on MAR address to IR
34	**PC <= PC + 1**	Increment pc
35	**state = 2**	Next state, data now on IR
38	**MAR <= IR [11:0]**	Take the address of the operand saved at IR to MAR
39	**state = 3**	Next state, MAR and IR has the data
42	**state = 4**	Next state,
43	**case (IR [15:12])**	case for the various R-type instructions
44	**load: MBR <= Memory [MAR]**	Load data at the memory in MAR Memory [MAR] to MBR
45	**add: MBR <= Memory [MAR]**	Add data at the memory in MAR Memory [MAR] to one in MBR
46	**store: MBR <= AC**	Store data from AC into MBR
52	**AC <= AC + MBR**	If instruction was ADD then AC = AC + MBR
56	**AC <= MBR**	If instruction was LOAD, then AC = MBR
60	**Memory [MAR] <= MBR**	If instruction was STORE, then Memory [MAR] = MBR