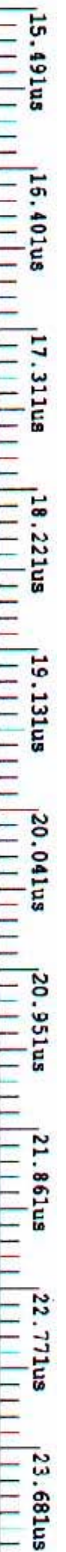


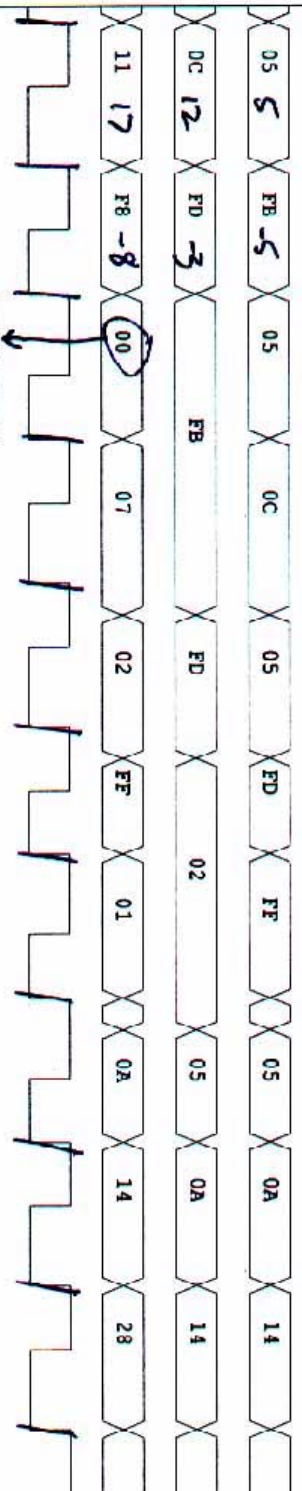
reg values
B A
ALU
output

B	U1.OP2.. (hex) #3
B	U1.A7... (hex) #8
B	U1.B7... (hex) #8
B	U1.OUT7. (hex) #8
i	U2.CLK.....
o	U1.ZERO -
B	U2.DEST1. (hex) #
B	U2.SRCAL. (hex) #
B	U2.SRCB1. (hex) #
i	U2.WRITE.....

Initial Values:
R0: 5
R1: 12
R2: -5
R3: -3



opcode 0 = add



3	0	1	R1	3	0	reg write address
3	0	0	R0	3	0	reg read address A
2	1	3	R3	1	0	reg read address B

don't write dest reg

write dest reg

c1: ^(R0) 5 + ^(R1) 12 = 17

c2: -5 + -3 = -8

c3: 5 + -5 = 0
zero flag set!

c4: 12 + -5 = 7

(regs now being written)
c5: R3 = R0(5) + R3(-3)
(2)

c6: R3 = R3(-3) + R1(now 2!)
(-1)
c6: R3 = R3(now -1) + R1(2)

c7: R0 = R0(5) + R0

R0 = R0(now 10) + R0
= 20

R0 = 20 + 20

Demonstrating ALU
only 'add'. You'll demonstrate all ops)

Demonstrating reg reading and writing