James Stallkamp

ECE 375 HW2

1 Give the sequence of micro operations for the fetch and execute cycles of the call label.

Fetch:

1. MAR<-PC
2. MDR<-M(MAR), PC<-PC+1
3. IR(high) <-MDR
4. MAR<-PC
5. MDR<-M(MAR), PC<-PC+1
6. IR(low) <-MDR
7. MAR<-PC
8. MDR<-M(MAR), PC<-PC+1
9. TEMP(high)<-MDR
10. MAR<-PC
11. MDR<-M(MAR), PC<-PC+1
12. TEMP(low)<-MDR

Execute:

1. MDR<-PC(high)
2. MAR<-SP
3. M(MAR)<-MDR, SP<-SP-1
4. MDR<-PC(low)
5. MAR<-SP
6. M(MAR)<-MDR, SP<-SP-1
7. PC<-TEMP

2 Give the sequence of micro operations for the fetch and execute cycles of the ret label.

Fetch:

13. MAR<-PC
14. MDR<-M(MAR)
15. IR(high) <- MDR, PC<-PC+1
16. MAR<-PC
17. MDR<-M(MAR)
18. IR(low) <- MDR, PC<-PC+1

Execute:

1. SP<-SP+1

2. MAR<-SP, MDR<-M(MAR)
3. PV(high)<-MDR, SP<-SP+1
4. MAR<-SP, MDR<-M(MAR)
5. PC(low)<-MDR

3Examine code from lab 5 and answer questions.

a) The two 16 bit values being multiplied are: 0108 and 1414
b) After lines 11-25 the first time
    a. LAddr:20
    b. LAddr+1:00
    c. LAddr+2:00
    d. LAddr+3:00
c) After lines 11-25 the second time
    a. LAddr:20
    b. LAddr+1:04
    c. LAddr+2:00
    d. LAddr+3:00
d) After lines 11-25 the third time
    a. LAddr:20
    b. LAddr+1:04
    c. LAddr+2:32
    d. LAddr+3:00
e) After lines 11-25 the fourth time
    a. LAddr:20
    b. LAddr+1:04
    c. LAddr+2:32
    d. LAddr+3:04

4 Write an avr assembly subroutine that implements XOR.

Xor = ~(~ p and ~ q) and ~(p and q)

XOR:

```
Mov r3, r1;
```

```
Mov r4, r2;

Mov r5, r1;

Mov r6, r2;


Com r3

Com r4

And r3, r4

Com r3


And R5, r6

Com r5

And r3 r5

Ret
```

4a. The stack will contain the return address after rcall is done. The return address is placed on the stack with low byte first

0xff
0x01