

16.216: ECE Application Programming

Practice Problems for Exam 2

Note that these problems do not cover all material to be tested on the second exam. However, any topic that is not covered by at least one practice problem will only be tested in a multiple choice question (if at all).

1. Assume the state of the 80386DX's registers and memory are:

- (EAX) = 00005555H
- (EBX) = 00000010H
- (ECX) = 00000010H
- (EDX) = 0000AAAAH
- (ESI) = 00000100H
- (EDI) = 00000200H
- (DS:100H) = 0FH
- (DS:101H) = F0H
- (DS:110H) = 00H
- (DS:111H) = FFH
- (DS:200H) = 30H
- (DS:201H) = 00H
- (DS:210H) = AAH
- (DS:211H) = AAH
- (DS:220H) = 55H
- (DS:221H) = 55H
- (DS:300H) = AAH
- (DS:301H) = 55H

Also, assume all flags (ZF, CF, SF, PF, OF) are initialized to 0.

For each instruction sequence shown below, list all changed registers and/or memory locations and their new values, as well as all changed flags from the list above. Note that the registers and memory have the same starting values at the beginning of each sequence, but a value changed by one instruction in a sequence can affect the results of all other instructions in the same sequence.

- a. BT AX, 4
SETC [100H]
BTS AX, 5
SETC [101H]
BTR AX, 6
SETC [110H]
BTC AX, 7
SETC [111H]
- b. BSF AL, WORD PTR [BX+SI]
BSR AH, WORD PTR [BX+SI]
CMP AL, AH
JG S
MOV DX, [200H]
JMP E
S: MOV DX, [210H]
E: MOV [BX+DI+10H], DX

1 (cont.) Assume the state of the 80386DX's registers and memory are:

- (EAX) = 00005555H
- (EBX) = 00000010H
- (ECX) = 00000010H
- (EDX) = 0000AAAAH
- (ESI) = 00000100H
- (EDI) = 00000200H
- (DS:100H) = 0FH
- (DS:101H) = F0H
- (DS:110H) = 00H
- (DS:111H) = FFH
- (DS:200H) = 30H
- (DS:201H) = 00H
- (DS:210H) = AAH
- (DS:211H) = AAH
- (DS:220H) = 55H
- (DS:221H) = 55H
- (DS:300H) = AAH
- (DS:301H) = 55H

Also, assume all flags (ZF, CF, SF, PF, OF) are initialized to 0.

```
c.    CMP    AL, 56H
      JL     L1
      JG     L2
      MOV    AH, BL
      JMP    E
L1:   MOV    AH, CH
      JMP    E
L2:   MOV    AH, DL
E:    SETL   [DI]
```

```
d.    MOV    AX, 0001H
      MOV    CX, 0004H
ST:   SHL    AX, CX
      LOOP  ST
```

```
e.    MOV    AX, 8000H
ST:   SAR    AX, 1
      CMP    AX, [BX+SI]
      LOOPNE ST
```

2. As noted in class, the SETcc instruction can be used to combine multiple conditions together to create a compound conditional test. For example, the code below tests the condition $((A < B) \&\& (C < D))$, storing the result in DL:

```
MOV    AX, A
CMP    AX, B
SETL   DL
MOV    AX, C
CMP    AX, D
SETL   DH
AND    DL, DH
```

For each part of this problem, assume A, B, C, D, E, and F refer to signed integers stored in memory.

- a. What compound condition is tested by each of the code sequences below?

i.

```
MOV    AX, A
CMP    AX, B
SETLE  BL
CMP    AX, E
SETGE  BH
OR     BL, BH
```

iii.

```
MOV    AX, A
SUB    AX, B
CMP    AX, C
SETGE  BL
MOV    AX, D
ADD    AX, E
SUB    AX, F
SETNZ  BH
OR     BL, BH
```

ii.

```
MOV    AX, C
CMP    AX, A
SETE   BL
MOV    AX, B
CMP    AX, A
SETNE  BH
AND    BL, BH
CMP    AX, C
SETL   BH
AND    BL, BH
CMP    AX, A
SETZ   BH
OR     BL, BH
```

- b. Write a sequence of instructions that tests each of the following compound conditions.

- i. $((A > B) \ || \ (A < C)) \ \&\& \ ((A \neq D) \ || \ (A == E))$
ii. $((A - B > 0) \ \&\& \ !C)$
iii. $((B \geq A + C) \ || \ (D \leq C + A))$

3. Assume CS = 1010H, IP = 1A00, and EBX = 20AAFE00. What is the starting address of each subroutine accessed by the CALL instructions below? (In other words, what is the target address of the CALL?)
- i. CALL 0100H
 - ii. CALL FFF0H
 - iii. CALL 411ABE00
 - iv. CALL BX
 - v. CALL EBX
4. Assume the 80386 is running in protected mode with the state given below (all values in hex); note that each memory location shown contains a descriptor about a particular segment:

GDTR = 00200000001F
LDTR = 000B

DS = 0017
SS = 0018
ESI = 00001000
EBX = 0001120

Memory	Address
Base = 030010F0 Limit = 020F	00200000
Base = 00200020 Limit = 0017	00200008
Base = 00200038 Limit = 0010	00200010
Base = 1200C000 Limit = FFFF	00200018
Base = 12340000 Limit = 00FF	00200020

Memory	Address
Base = 01000010 Limit = 1127	00200028
Base = 03170200 Limit = 03F7	00200030
Base = 1A000000 Limit = 01FF	00200038
Base = 06B01000 Limit = 0F07	00200040
Base = 05000120 Limit = 000F	00200048

- a. What is the base address and limit of the global descriptor table? How many descriptors does this table contain?
- b. What is the base address and limit of the current local descriptor table? How many descriptors does this table contain?
- c. What are the starting and ending addresses for the current data and stack segments?
- d. What address is accessed by each of the following instructions?
 - i. MOV AX, [0100H]
 - ii. ADD DX, [SI]
 - iii. MOV AX, SS:[SI+EF00]
 - iv. SUB SS:[A200], CX
 - v. MOV DX, [BX+SI]
 - vi. MOV CX, [BX+SI+1EH]