# 16.317: Microprocessor Systems Design I

Summer 2013

Lecture 2: Key Questions
July 11, 2013

1. Describe the general characteristics of the x86 architecture.

2. Briefly describe the x86 registers.

3. Describe the different memory spaces in the x86 architecture.

4. Explain the basic concept of memory segmentation.

5. Describe the specifics of x86 memory segmentation.

6. Describe how x86 real mode addresses are generated, including the components of the address and the actual calculations performed.

7. **Example:** Given the following register values:

- CS = 0x1000
- SS = 0x2000
- DS = 0x3000
- ES = 0x4000
- IP = 0x0100
- ESP = 0x0002FF00
- EBP = 0x0000F000
- ESI = 0x0001000E
- EBX = 0xABCD1234

What linear addresses correspond to the following logical addresses?

- CS:IP

- SS:SP

- SS:BP

- DS:SI

- ES:BX

8. Describe the different addressing modes specific to the x86 architecture.

9. **Example:** Compute the linear address for the specified operand in each of the following instructions. The register contents and variables are as follows:
   - (CS) = $0A00_{16}$
   - (DS) = $0B00_{16}$
   - (ESI) = $00000100_{16}$
   - (EDI) = $00000200_{16}$
   - (EBX) = $00000300_{16}$

a. Destination operand in: MOV [DI], AX

b. Source operand in: MOV DI, [SI]

c. Destination operand in: MOV [BX+0400H], CX

d. Destination operand in: MOV [DI+0400H], AH

e. Destination operand in MOV [BX+DI+0400H], AL

10. Describe the basic structure of an assembly language statement.

11. What information is typically encoded in an instruction?

12. What is the benefit of having fixed-length instructions? Variable-length instructions?

13. Describe how the x86 registers are accessed as 8-bit, 16-bit, and 32-bit values. Include the answer to the example provided in the slides (EAX = 1A2B3C4DH).

14. Describe how to determine the number of bytes being accessed from memory in an x86 instruction.

15. Describe the use of the MOV instruction.

16. The example program below shows the initialization of internal registers with immediate data and address information, using MOV instructions. Show the state of all affected registers. Also, explain why AX is used to initialize segment registers.

```
MOV  AX,2000H
MOV  DS, AX
MOV  ES, AX
MOV  AX,3000H
MOV  SS,AX
MOV  AX,0H
MOV  BX,AX
MOV  CX,0AH
MOV  DX,100H
MOV  SI,200H
MOV  DI,300H
```

17. Describe the operation of the MOVSX/MOVZX instructions. How/when are these instructions useful?

18. Assume: AX = 0100H, DX = 8100H, (DS:100H) = 00H, (DS:101H) = FFH. What are the results of the following instructions?

a. MOVSX EBX, AX

b. MOVSX EBX, DX

c. MOVZX EBX, DX

d. MOVSX EBX, BYTE PTR [100H]

e. MOVSX EBX, WORD PTR [100H]

19. Explain the operation of the XCHG instruction. What restrictions are placed on this instruction?

20. Explain the operation of the LEA instruction.

21. Explain the operation of the instructions used for loading a full address pointer (LDS, LSS, LES, LFS, LGS).

22. Show the results of running the following program if DATA_SEG_ADDR = 1200H, assuming the memory contents shown:

| | | |
|---|---|---|
| DATA_SEG_ADDR:0000 | | |
| | | |
| DATA_SEG_ADDR:INIT_TABLE | 11 | 22 |
| | 33 | 44 |
| | 55 | 66 |
| | 77 | 88 |
| | 99 | AA |
| | BB | CC |
| | DD | EE |
| | FF | 16 |
| | 03 | 17 |

```
MOV  AX, DATA_SEG_ADDR
MOV  DS,AX
MOV  SI,[INIT_TABLE]
LES  DI,[INIT_TABLE+02H]
MOV  AX,[INIT_TABLE+06H]
MOV  SS,AX
MOV  AX,[INIT_TABLE+08H]
MOV  BX,[INIT_TABLE+0AH]
MOV  CX,[INIT_TABLE+0CH]
MOV  DX,[INIT_TABLE+0EH]
```