Online Instructor's Manual
to accompany

The x86 PC: Assembly Language, Design, and Interfacing

5th Edition

Muhammad Ali Mazidi

Janice Gillispie Mazidi

Danny Causey

Prentice Hall

Boston Columbus Indianapolis New York San Francisco Upper Saddle River

Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montreal Toronto

Delhi Mexico City Sao Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo

10 9 8 7 6 5 4 3 2 1

PEARSON

ISBN-13: 978-0-13-504081-2
ISBN-10: 0-13-504081-7

CHAPTER 0: INTRODUCTION TO COMPUTING

SECTION 0.1: NUMBERING AND CODING SYSTEMS

1.
(a) 1210 = 11002
(b) 12310 = 0111 10112
(c) 6310 = 0011 11112
(d) 12810 = 1000 00002
(e) 100010 = 0011 1110 10002
2.
(a) 1001002 = 3610
(b) 10000012 = 6510
(c) 111012 = 2910
(d) 10102 = 1010
(e) 001000102 = 3410
3.
(a) 1001002 = 2416
(b) 10000012 = 4116
(c) 111012 = 1D16
(d) 10102 = 0A16
(e) 001000102 = 2216
4.
(a) 2B916 = 0010 1011 10012, 69710
(b) F4416 = 1111 0100 01002, 390810
(c) 91216 = 1001 0001 00102, 232210
(d) 2B16 = 0010 10112, 4310
(e) FFFF16 = 1111 1111 1111 11112, 6553510
5.
(a) 1210 = 0C16
(b) 12310 = 7B16
(c) 6310 = 3F16
(d) 12810 = 8016
(e) 100010 = 3E816
6.
(a) 1001010 = 0011 0110
(b) 111001 = 0000 0111
(c) 10000010 = 0111 1110
(d) 111110001 = 0000 1111
7.
(a) 2C+3F = 6B
(b) F34+5D6 = 150A
(c) 20000+12FF = 212FF
(d) FFFF+2222 = 12221

8. (a) 24F-129 = 12616
(b) FE9-5CC = A1D16
(c) 2FFFF-FFFFF = 3000016
(d) 9FF25-4DD99 = 5218C16

9. (a) Hex: 30, 31, 32 , 33, 34, 35, 36, 37, 38, 39
(b) Binary: 11 0000, 11 0001, 11 0010, 11 0011, 11 0100, 11 0101, 11 0110, 11 0111, 11 1000, 11 1001.

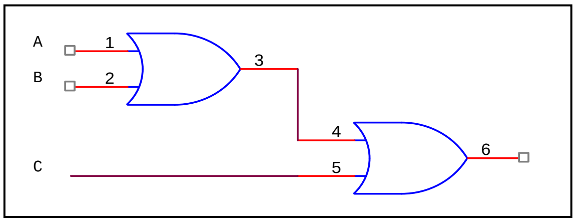
ASCII(hex)	Binary		
0	30	011	0000
1	31	011	0001

2	32	011	0010
3	33	011	0011
4	34	011	0100
5	35	011	0101
6	36	011	0110
7	37	011	0111
8	38	011	1000
9	39	011	1001

10.
```
000000 22 55 2E 53 2E 41 2E 20 69 73 20 61 20 63 6F 75    "U.S.A. is a cou
000010 6E 74 72 79 22 0D 0A 22 69 6E 20 4E 6F 72 74 68    ntry".."in North
000020 20 41 6D 65 72 69 63 61 22 0D 0A                   America"..
```

SECTION 0.2: DIGITAL PRIMER

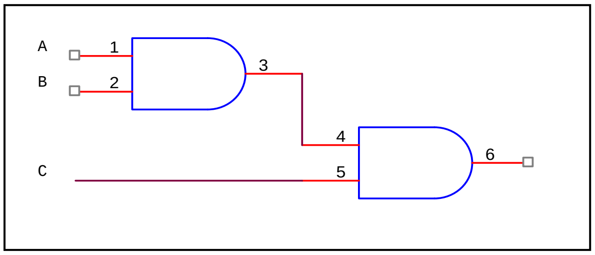
11.



12.

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

13.

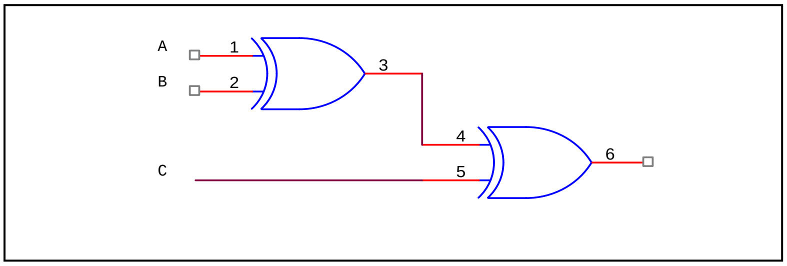


14.

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0

1	1	0	0
1	1	1	1

15.



A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

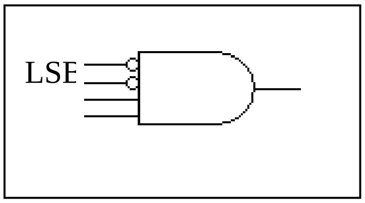
16.

A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

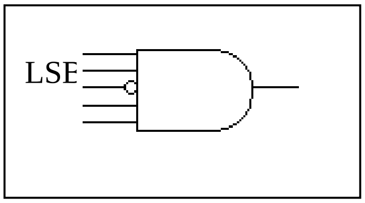
17.

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

18.



19.



20.

CLK	D	Q
No	X	NC
Yes	0	0
Yes	1	1

SECTION 0.3: INSIDE THE COMPUTER

21. (a) 4

(b) 4

(c) 4

(d) 1 048 576, 220

(e) 1024K

(f) 1 073 741 824, 230

(g) 1 048 576 K

(h) 1024M

(i) 8388608, 8192K

22. Disk storage capacity / size of a page = (2*230) / (25*80) = 1 million pages

23. (a) 9FFFFh – 10000h = 8FFFFh = 589 824 bytes

(b) 576 kbytes

24. 232 – 1 = 4 294 967 295

25. (a) FFh, 255

(b) FFFFh, 65535

(c) FFFF FFFFh, 4 294 967 295

(d) FFFF FFFF FFFF FFFFh, 18 446 744 073 709 551 615

26. (a) 216 = 64K

(b) 224 = 16M

(c) 232 = 4096 Mega, 4G

(d) 248 = 256 Tera, 262144 Giga, 268435456 Mega

27. Data bus is bidirectional, address bus is unidirectional (exit CPU).

28. PC (Program Counter)

29. ALU is responsible for all arithmetic and logic calculations in the CPU.

30. Address, control and data

CHAPTER 1: THE x86 MICROPROCESSOR

SECTION 1.1: BRIEF HISTORY OF THE x86 FAMILY

1. 8086

2. the internal data bus of the 386SX is 32 bits, whereas
   the internal data bus of the 286 is 16 bits

3. terms such as "16-bit" or "32-bit" microprocessors refer
   to the internal data bus and register size of the microprocessor

4. yes

5. upward compatibility means that any program written
   for a lower (earlier) system will run on more advanced (later) systems

6. the 8088 has an 8-bit external data bus but the 8086
   has a 16-bit external data bus

7. the 8088 has a 4-byte queue, the 8086 has a 6-byte queue

SECTION 1.2: INSIDE THE 8088/86

8. more efficient internal architecture such as pipelining
   and wider registers

9. the BIU (bus interface unit) fetches instructions into
   the CPU and the EU (execution unit) executes the instruction

10. (a) 8-bit registers are: AH, AL, BH, BL, CH, CL, DH,CL
    (b) 16-bit registers are: AX, BX, CX, DX

11. (a) CS (c) DS (d) SS (h) SI (i) DI

SECTION 1.3: INTRODUCTION TO ASSEMBLY PROGRAMMING

12. (b) is illegal since the value is too large
    (c) is illegal since immediate addressing is not allowed for segment registers
    (f) is illegal since immediate addressing is not allowed for segment registers
    (i) is illegal since the operand types do not match
    (j) is illegal since the value is too large for the register
    (k) is illegal since the register sizes do not match
    (l) is illegal since the operand sizes do not match

SECTION 1.4: INTRODUCTION TO PROGRAM SEGMENTS

13. CS is the code segment register and holds the segment address for the code section
    DS is the data segment register and holds the segment address for the data section
    SS is the stack segment register and holds the segment address for the stack section
    ES is the extra segment register and holds the segment address for the extra segment
    which is used for many string operations

14. (a) 3499:2500 (b) 36E90 (c) 34990 to 4498F

15. (a) 1296:0100 (b) 12A60 (c) 12960 to 2295F

16. (a) 38949 (b) 3499:3FB9 (c) 34990 to 4498F

17. (a) 1A648 (b) 1298:7CC8 (c) 12980 to 2297F

18. 0042:004C

19. no, because the upper range of the code segment would be 36FFF
CS should be 3777

20.

12B0:0170	12C70	B0
12B0:0171	12C71	76
12B0:0172	12C72	B7
12B0:0173	12C73	8F
12B0:0174	12C74	00
12B0:0175	12C75	C7
12B0:0176	12C76	80
12B0:0177	12C77	C7
12B0:0178	12C78	7B
12B0:0179	12C79	88
12B0:017A	12C7A	FB
12B0:017B	12C7B	00
12B0:017C	12C7C	C3

21.

12B0:0100	12C00	B0
12B0:0101	12C01	00
12B0:0102	12C02	02
12B0:0103	12C03	06
12B0:0104	12C04	00
12B0:0105	12C05	02
12B0:0106	12C06	02
12B0:0107	12C07	06
12B0:0108	12C08	01
12B0:0109	12C09	02
12B0:010A	12C0A	02
12B0:010B	12C0B	06
12B0:010C	12C0C	02
12B0:010D	12C0D	02
12B0:010E	12C0E	02
12B0:010F	12C0F	06
12B0:0110	12C10	03
12B0:0111	12C11	02
12B0:0112	12C12	02
12B0:0113	12C13	06
12B0:0114	12C14	04
12B0:0115	12C15	02

SECTION 1.5: THE STACK

22. (b)

23. (c)

24. decremented, incremented

25. (b)

26. the stack is slower than registers, since the stack is a section of RAM

27. (a) 24578 (b) 2000:4578 (c) 20000 (d) 2FFFF

28. 24FB

29. after "PUSH AX", the stack pointer = FF2C and the stack is as follows:

logical address	stack contents
SS:FF2C	91
SS:FF2D	32

after "PUSH BX", the stack pointer = FF2A and the stack is as follows:

logical address	stack contents
SS:FF2A	3C
SS:FF2B	F4
SS:FF2C	91
SS:FF2D	32

after "PUSH CX", the stack pointer = FF28 and the stack is as follows:

logical address	stack contents
SS:FF28	09
SS:FF29	00
SS:FF2A	3C
SS:FF2B	F4
SS:FF2C	91
SS:FF2D	32

30. at the conclusion of Problem 28, SP = FF28,

```
POP CX       ;then SP = FF2A
POP BX       ;then SP = FF2C
POP AX       ;then SP = FF2E
```

31. (a) SS (b) DS (c) CS (d) DS
(e) SS (f) DS

32. (a) SS overrides default register DS
(b) SS overrides default register DS
(c) DS overrides default register SS

SECTION 1.6: FLAG REGISTER

33. (a) CF = 1 indicating a carry occurred
PF = 1 indicating even parity
AF = 1 indicating a carry from bit 3
ZF = 1 indicating the result is zero
SF = 0 indicating a positive result

(b) CF = 0 indicating no carry

PF = 0 indicating odd parity
AF = 0 indicating no carry from bit 3
ZF = 0 indicating that the result is not zero
SF = 1 indicating negative result

(c) CF = 0 indicating no carry
PF = 1 indicating even parity
AF = 1 indicating a carry from bit 3
ZF = 0 indicating the result is not zero
SF = 0 indicating positive result

SECTION 1.7: x86 ADDRESSING MODES

34. (a) location 24000 (20000 + 4000) contains FF
(b) location 2A088 (20000 + 4000 + 6080 + 8) contains 25
(c) location 26080 (20000 + 6080) contains FF
location 26081 contains 25
(d) location 25006 (20000 + 5000 + 6) contains 80
location 25007 contains 60
(e) location 2B0A8 (20000 + 5000 + 6080 + 28) contains 91
location 2B0A9 contains 87
(f) location 34010 (30000 + 4000 + 10) contains 99
location 34011 contains 12
(g) location 23600 (20000 + 3600) contains FF
location 23601 contains 25
(h) location 260B0 (20000 + 6080 + 30) contains 99
location 260B1 contains 12
(i) location 37200 (30000 + 7000 + 200) contains FF
location 37201 contains 25
(j) location 3B100 (30000 + 7000 + 4000 + 100) contains 80
location 3B101 contains 60
(k) location 24050 (20000 + 4000 + 50) contains 25
(l) location 2C100 (20000 + 5000 + 7000 + 100) contains FF
location 2C101 contains 25

35. (a) register (b) immediate
(c) direct (d) register
(e) register indirect (f) register indirect
(g) based index (h) register
(i) based (j) based index
(k) index (l) based index

36. (a) DS:1450 contains 9F DS:1451 contains 12
(b) DS:2348 contains 63 DS:2349 contains 8C