

# ASCII Table

Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char	Dec	Hex	Oct	Char
0	0	0		32	20	40	[space]	64	40	100	@	96	60	140	'
1	1	1		33	21	41	!	65	41	101	A	97	61	141	a
2	2	2		34	22	42	"	66	42	102	B	98	62	142	b
3	3	3		35	23	43	#	67	43	103	C	99	63	143	c
4	4	4		36	24	44	\$	68	44	104	D	100	64	144	d
5	5	5		37	25	45	%	69	45	105	E	101	65	145	e
6	6	6		38	26	46	&	70	46	106	F	102	66	146	f
7	7	7		39	27	47	'	71	47	107	G	103	67	147	g
8	8	10		40	28	50	(	72	48	110	H	104	68	150	h
9	9	11		41	29	51	)	73	49	111	I	105	69	151	i
10	A	12		42	2A	52	*	74	4A	112	J	106	6A	152	j
11	B	13		43	2B	53	+	75	4B	113	K	107	6B	153	k
12	C	14		44	2C	54	,	76	4C	114	L	108	6C	154	l
13	D	15		45	2D	55	-	77	4D	115	M	109	6D	155	m
14	E	16		46	2E	56	.	78	4E	116	N	110	6E	156	n
15	F	17		47	2F	57	/	79	4F	117	O	111	6F	157	o
16	10	20		48	30	60	0	80	50	120	P	112	70	160	p
17	11	21		49	31	61	1	81	51	121	Q	113	71	161	q
18	12	22		50	32	62	2	82	52	122	R	114	72	162	r
19	13	23		51	33	63	3	83	53	123	S	115	73	163	s
20	14	24		52	34	64	4	84	54	124	T	116	74	164	t
21	15	25		53	35	65	5	85	55	125	U	117	75	165	u
22	16	26		54	36	66	6	86	56	126	V	118	76	166	v
23	17	27		55	37	67	7	87	57	127	W	119	77	167	w
24	18	30		56	38	70	8	88	58	130	X	120	78	170	x
25	19	31		57	39	71	9	89	59	131	Y	121	79	171	y
26	1A	32		58	3A	72	:	90	5A	132	Z	122	7A	172	z
27	1B	33		59	3B	73	:	91	5B	133	[	123	7B	173	{
28	1C	34		60	3C	74	<	92	5C	134	\	124	7C	174	
29	1D	35		61	3D	75	=	93	5D	135	]	125	7D	175	}
30	1E	36		62	3E	76	>	94	5E	136	^	126	7E	176	~
31	1F	37		63	3F	77	?	95	5F	137	-	127	7F	177	

EEE 174 Spring 2015

- 1) Given: 0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D  
0B0E:0210 62 6C 79 20 4C 61 6E 67-75 67 61 65 00 00 00 00  
An ASCII message begins at memory location 0200, what is the message?  
 ⇒ Welcome to Assembly Language
- 2) For the instruction sequence below, determine the contents of the register AL after this program is executed:

Program Listing

MOV AL,83h

ADD AL,68h

DAA

⇒ 51h

- 3) The letters "NO" labelled on relays and PLCs means which of the following?  
 ⇒ Normally open
- 4) A microprocessor with a 33-bit address bus could access how much memory?  
 ⇒ 8GB
- 5) What is the number, 1010.0101(2) in decimal?  
 ⇒ 10.31
- 6) What is -96.2697 converted to single precision FP?  
 ⇒ C2 C0 8A 16

- 7) In MASM, with a “MOV CX,24h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
⇒ 36
- 8) If the SP is F00F, what is the SP value after a “POP CX” instruction?  
⇒ F011
- 9) On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh? => 0
- 10) Which of the following would be used to set the TRISA register to control the direction of the PIC18 port to input? =>1
- 11) The instruction MOV CX,SI is what addressing mode?  
⇒ Immediate
- 12) Question 14
- 

When should we use pre-compiler statements? Select one:

- a. Create Constants
- b. All of the answers
- c. Create simple functions like #define getmax(a,b) ((a)>(b)?(a):(b))
- d. Remove blocks of code for debugging

What is the voltage for IO output for an “OFF” signal?

Select one:

- a. Between 3.3V to 5V
- b. Around 1V
- c. Between 2.5V to 3.3V
- d. all of the answers
- e. None of the answers
- f. Between 0V to 5V

The Intel x86 architecture is based on which of the following computer architecture(s)?

Select one:

- a. Von Neumann
- b. Berkley
- c. MIT
- d. None of the answers
- e. Stanford
- f. Yale

g. Harvard

What is the RS232C specification voltage range for Logic 1 output?

Select one:

- a. -12v
- b. -3v to -25v
- c. +5 to +15v
- d. +3v to +25v
- e. -5v to -15v

How many bytes are in an INT?

Select one:

- a. 2 byte
- b. 4 byte
- c. a or b
- d. None of the answers

What is  $-32.75_{10}$  in a base two number system?

100000.11000

Which of the following is(are) setback(s) of the Arduino platform?

Select one:

- a. Structure of functions may not be as efficient
- b. all of the answers
- c. Scaling to larger design
- d. Cost
- e. None of the answers

What is C2 C0 8A 16 Hex single precision FP converted to Decimal?

Select one:

- a. -9.781250
- b. 0.078125
- c. 9.7500
- d. 152.1875

**e. -96.2697**

What is the resolution of and 10bit ADC operating at 5 volts?

Select one:

- a. 5 mV
- b. 5 nV
- c. 0 V
- d. 5 V
- e. 5 uV
- f. None of the answers

In the MicroChip PIC what bit mask would need to be applied to configure bit 4 of Port D to input? Assume ORing bit operation is applied.

Select one:

- a. 0xA8
- b. 0xFF**
- c. 0x0A
- d. 0x08
- e. None of the answers
- f. 0xA0

Which of the following is a setback(s) of the Arduino platform?

Select one:

- a. Rapid prototyping and ready to use**
- b. Scaling to larger design**
- c. None of the answers
- d. Large Open Source Community
- e. all of the answers

In the x86 which register in the segmented group is used for code?

Select one:

- a. AX
- b. SI
- c. IP
- d. All of the answers
- e. CX
- f. DS
- g. SP
- h. DI
- i. BX
- j. DX
- k. None of the answers
- l. CS**
- m. SS

In the MicroChip PIC with a bit mask = 0xFF, what bit mask operation would be used to configure Port D to input?

Select one:

- a. ORing
- b. XNORing
- c. ANDing**
- d. XORing

On the Analog Discovery, to examine the data from a I2C bus what tool would you use?

Select one:

- a. Network
- b. Logic**
- c. Scope
- d. None of the answers
- e. All of the answers
- f. Logger

In the MicroChip PIC what bit mask would need to be applied to configure Port D to input? Assume ORing bit operation is applied.

Select one:

- a. 0x0A
- b. 0xF0
- c. None of the answers
- d. 0xA0
- e. 0x0F
- f. 0xFF**

102-

In the x86 which register is used for current stack location?

Select one:

- a. AX
- b. IP
- c. SS**
- d. CS
- e. DI
- f. DX
- g. BX
- h. SI
- i. DS
- j. SP
- k. CX
- l. All of the answers
- m. None of the answers

**In adding 10+6 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:**

Select one:

- a. OF = 0, CF = 0**
- b. OF = 1, CF = 1
- c. OF = 0, CF = 1
- d. OF = 1, CF = 0
- e. OF = 0, CF = 0, ZF = 0

**1010**

**0110**

**1110**

In the x86 which register is used most often and has the one less byte in its machine code?

Select one:

- a. DX
- b. SI
- c. SS
- d. BX
- e. DS
- f. CX
- g. None of the answers
- h. DI
- i. AX
- j. CS
- k. All of the answers

What is the resolution of a 10bit ADC operating at 5 volts?

I don't know how to do this one

Select one:

- a. 5 uV
- b. 5 nV
- c. 5 mV
- d. None of the answers
- e. 5 V
- f. 0 V

---

1010 0110 in 2's complement equals in base 10 → **-90**

A microprocessor with a 32-bit address bus could access how much memory → **4GB**

A6 in 2's complement equals in base 10 → **-90**

A “POP” instruction: → **Increments the SP**

A “PUSH” instruction → **decrements the SP**

A “NOP” instruction in a program will → **Perform a No Operation**

AND’ing 1FH and 02H will result in which of the following → **02**

AND’ing 10H and 2FH will result in which of the following → **0**

A “pull down” resistor is used in digital circuits to do what? → To keep the signal line “tied” low until the line is active (goes high)

A “pull up” resistor is used in digital circuits to do what → To keep singal “tied” high until the line is active (goes low)

Determine the contents of register BL after the following instructions have been executed: → E2H

MOV BL, E2H

MOV CL, 1000b

ROL BL, CL ; rotate BL by 8 to left = same value

For the instruction sequence below, determine the contents of the registers AL after this program → 51H

MOV AL, 73h ; 0111 0011

ADD AL, 78h ; 0111 1000 => 1110 1011 => EA

DAA

if low nibble of AL > 9 or AF = 1 then:

AL = AL + 6

AF = 1

if AL > 9Fh or CF = 1 then:

AL = AL + 60h

CF = 1

For the instruction sequence below, determine the contents of the register AL after this program is executed → 51H

MOV AL, 83h ; 1000 0011

ADD, AL, 68h ;0110 1000 = 1110 1011 = EA

DAA

Given the short code, what is the value in AX after the program is run → 0100

Mov BX, 0100  
PUSH BX  
MOV AX, 0500  
POP AX

MOV BX, 0100

MOV BX, 0001

Given the short code, what is the value in AX after the program is run → 0001

PUSH BX  
MOV AX, 0500  
POP AX

GIVEN: IP = 0106 Flags: NV UP EI NG NZ NA PE NC Instruction: JMP 011F

What will the IP value be after “t” command is executed in DOS Debug? → 011FH (Unconditional Jump)

GIVEN: IP = 0109 Flags: OV UP EI PL NZ NA PO CY Instruction: JGE 0118

What will the IP value be after a “t” command is executed in DOS Debug → 010B

(0109 + 0010 add two bytes)

GIVEN: IP=FFE0 Flags: OV UP EI PL NZ NA PO NC Instruction: JGE 0116: ID72:010D 7D09

How many bytes will the processor jump if the condition for a jump were met? → 9

GIVEN: AX= FFF0 IP = 0109 FALGS: OV UP EI PL NZ NA PO CY ID72: 010F **7D18** Instruction: JGE 0118. What is the signed decimal value of the number in the AX register? → -16

Covert the number 7D18 into decimal.

GIVEN: IP= 010F Flags: NV UP EI NG NZ NA PO NC Instruction: JNL 0115.

How many bytes in decimal will the processor jump if the conditions for a jump were met → 24

GIVEN: 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D

62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message? → Welcome to Assembly Language

Here is a short sequence of code: 7413 EBA3 CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The third instruction operator is → INT

Here is a short sequence of code: 7413 A3EB CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The fourth instruction operator is → JGE

Here is a short sequence of code: B400 CD16 3C4A 7404 BC6A 7513. All of the instructions are two bytes long. The sixth instruction operator is → JNZ

How many cores does the propeller microcontroller have → 8

How many bits(s) is/are required to represent a range of decimal numbers from 0 to 15 → 4

How many bits(s) is/are required to represent a range of decimal numbers from 0 to 63 → 6

How many bits(s) is/are required to represent a range of decimal numbers from 0 to 127 → 7

How many bits(s) is/are required to represent a range of decimal numbers from 0 to 255 → 8

How many bytes are there in this short sequence of code B4 00 CD 16 4C CD 20 → 7

How many nibbles are there in this short sequence of code B4 00 CD 16 3C 4A 74 04 3C 6A 75 13 → 24

How many bytes are in double precision IEEE floating point format numbers → 8

How many nibbles are in double precision IEEE floating point format numbers → 16

How many address lines would be required to address 128 MB directly → 27 (128 x 1048576 = 134217728 and  $2^{27} = 134217728$ )

How many address lines would be required to address 64 MB directly → 26 ( $64 \times 1048576 = 67108864$  and  $2^{26} = 67108864$ )

If CX is 0000, what will CX be after a “LOOP” instruction → FFFF

If CX is 0003, what will CX be after a “LOOPNZ” instruction → 0002

If the SP is F00F, what is the SP value after a “PUSH CX” instruction → F00D

If the SP is F00F, what is the SP value after a “POP CX” instruction → F011

If the SP is F00F, what is the SP value after a “POP SP” instruction → F011

In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be → OF = 1, CF = 0

In x86 architecture, BIU stands for which of the following → Bus Interface Unit

In x86 architecture, ALU stands for which of the following → Arithmetic Logic Unit

In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”? → 020E

In MASM, with a “MOV CX, 24h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal → 36

In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, how many times will the program loop in decimal → 24

In MASM, with a “MOV CX, 12h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal → 18

In the Hello MASM lab in the original code, what is the address of the string to start the message “Hello World 0” → 0200

In the PIC18 with TRISD = 0b10000000, what is the configuration of the Port D → Bit 7 of port D is set to input

In the PIC18 with TRISD = 0b01111111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs → Bit 7 of port D is set to output (because the first bit is zero = output)

In the PIC18 with TRISD = 0b00001111, what is the configuration of the Port D → A0 (First 4 are outputs and last four are inputs)

In the PIC18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs → 0A

In the Propeller microcontroller, the command “dira[9..4] := %000000” would cause the processor to do which of the following → Sets the propeller pin P4 through P9 as output pins

In the Propeller microcontroller, the command “dira[9..4] := %111111” would cause the processor to do which of the following → Sets the propeller pin P4 through P9 as output pins

In the propeller microcontroller, the command “waitcnt(clkfreq\*3 + cnt)” would cause the processor to do which of the following → A 3 second delay

In the propeller microcontroller, the command “waitcnt(clkfreq\*2 + cnt)” would cause the processor to do which of the following → A 2 second delay

In the Propeller microcontroller, the term “Method” is (are) which of the following → A block of executable commands that has variables, can receive parameters, and returns a value.

Int 10h uses what function code to write a character to the screen and advance the cursor by one character position → 0Eh

Int 21h, Function 09h requires three things set up before calling in order to correctly print a string:

**DS=SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h.**



What is -32.75 in a base two number system → **-100000.110000**

What is 14.4375 in binary → **001110.0110**

What is 16.4375 in binary → **010000.01110**

What is the binary value of decimal 12.875 → **1100.1110**

What is 16.4375 in binary → **010000.01110**

What is number, 1011.0101 (2) in decimal? → **11.31**

What is the **numeric sequence** of the key pad columns on the PPE board → **37,2F,1F**

What is the decimal value of C5 5A 57 00 in IEEE single precision FP format → **-3493.4375**

What of the following instruction would be used to set the LED to light on the Arduino platform → **digitalWrite(ledPin, HIGH);**

What type of program is this → **EXE**

**0115**, 1376:0115 0100      ADD [BX+SI], AL DS:0000=CD

IP =

What type of program is this → **COM**

0100, 1376:0100 0100      ADD [BX+SI], AL DS:0000=CD

IP =

Which of the following DOS Debug instructions would set a break point at memory location 010C → **G = 100 10C**

Which of the following would be used to set the **TRISA** register to control the direction of **PIC18** port to **input** → **1** and for **output** its → **0**

Which of the following DOS Debug instructions would be used to change the IP register to 110 → **RAX = 0110**

Which of the following will cause a program with a LOOP instruction to loop 48 times (decimal) → **CX=30h**

Which of the following is a valid x86 command for multiplying a number → **MUL BX**

Which of the following is **not a valid** command for a number into a register in MASM → **MOV AX, BADH**

With a POP BX instruction, what will be order off the accumulator, base, count, and data registers restored from the stack → **BX**

With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack → **BCDA**

You are trying to rebuild a HELLO program project in MASM and you get the following error: "ERROR 4 line 1". What is the cause of the error? → **Not known—this error by itself isn't a problem, press the enter key to clear the error.**

You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK : warning L4021: no stack segment". What would be the reason for the such an error → **No project template for COM was selected.**

You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK : fatal error L1089: HELLO.lrf: cannot open response file". What would be the reason for the such an error → **No source file is identified(no .asm file)**

#### **PPE Row Column Scan decoding**

D7 D6 D5 D4 D3 D2 D1 D0

S7 S6 S5 S4 S3 S2

^ ^ ^

3 2 1

0 0 0 0 1 0 0 0 = 08h

0 0 0 1 0 0 0 0 = 10h

0 0 1 0 0 0 0 0 = 20h

0 0 1 1 1 1 1 1 = 3Fh -> Nothing pressed

0 0 1 1 0 1 1 1 = 37h -> Number 1 pressed

0 0 1 0 1 1 1 1 = 2Fh -> Number 2 pressed  
 0 0 0 1 1 1 1 1 = 1Fh -> Number 3 pressed

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	Ø	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	\$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(	72	48	H	104	68	h
9	09	Horizontal tab	41	29	)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	Ø	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

#### S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:  
**DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36**

2. Moore's Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?

**Doubling every 18 – 24 months**

3. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
 1D72:0109 7D06 JNL 0118

What will the IP value be after a "t" command is executed in DOS Debug?

**010B**

4. A "NOP" instruction will:

### **Perform a No Operation**

5. Given:  
AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72:0106 EB0F JMP 0118

What will the IP value be after a “l” command is executed in DOS Debug?

**0118h**

6. In x86 architecture, BIU stands for which of the following?

**Bus Interface Unit**

7. Determine the contents of register BL after the following instructions have been executed:

**2EH**

7 Determine the contents of register BL after the following instructions have been executed:

**2EH Program Listing**

M  
O  
V

B  
L  
,

E  
2  
H  
M  
O  
V

C  
L  
,

1  
0  
0  
0  
b  
R  
O  
L

B  
L  
,

C  
L

8. The number of bytes in a double word are:

**4**

9. In x86 architecture, ALU stands for which of the following?

**Arithmetic Logic Unit**

10. The “LOOPNE” instruction is equivalent to which of the following instructions?

**DEC CX, JNE**

11. The instruction MOV CX, DADD is what addressing mode?

**Immediate**

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

**INT**

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The forth instruction operator is:

**JGE**

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The fifth instruction operator is:

**CMP**

13. Which of the following DOS Debug instructions would set a break point at memory location 010C?

**G = 100 10C**

14. How many bytes are there in this short sequence of code? B400CD164CCD21

**7**

15. How many address lines would be required to address 128 MB directly?

**27**

16. What are the contents of CX after this program has been run:

**D800h**

17. 1010 0110 in 2's complement equals \_\_\_\_ in base 10.

**-90**

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

**01D8**

19. What is 11.4375<sub>10</sub> in binary?

**001011.01110**

20. If CX is 0000 what will CX be after a "LOOP" instruction?

**FFFF**

21. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?

**7DF2**

22. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO NC  
1D72:010F 7D08 JGE 0115

How many bytes will the processor jump if the conditions for a jump were met?

**7 if not an option pick 8**

23. What command in DEBUG would be used to execute interrupts?

**P**

24. What is the advantage of C Language over Assembly Language?

**C is transportable to other microprocessor architectures**

25. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

**OF = 1, CF = 0**

26. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?  
**36**

27. Given:

AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JNL 0116

**-10**

28. AND'ing 1FH and 20H will result in which of the following?

**0**

29. The letters "NO" labeled on relays and PLCs means which of the following?

Normally open

13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

**72**

30. A microprocessor with a 31-bit address bus could access how much memory?

**2 GB**

31. For the instruction sequence below, determine the contents of the register AL after this program is executed:

**51H**

32. Given:

AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

**-256**

33. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

**32, 32, 13, 10, 36**

What is -96.2697 converted to single precision FP?

**C2 C0 8A 16**

What is C2 C0 8A 16 Hex single precision FP converted to Decimal

**-96.2697**

### Ieee converter

<http://www.h-schmidt.net/FloatConverter/IEEE754.html>

### S2010 MT2

1. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 99?

**7**

2. If CX is 0003, what will CX be after a “LOOP” instruction?

**0002**

3. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?

**A 5 second delay**

4. What is the number 1011.0101<sub>2</sub> in decimal?

**11.31**

5. This section of memory represents a stack. What type of program is this?

BEEF : FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00

BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F

BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

**COM program**

6. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

**BDCAB**

7. Determine the contents of register BL after the following instruction have been executed:

**2EH**

8. What Hex values must be sent to address the key pad rows on the PPE board?

**1, 2, 4, 8**

9. What is the number 32.4375<sub>10</sub> in binary?

**100000.01110**

10. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

**24**

11. The acronym PWM used for motor control, is defined as which of the following?

**Pulse Width Modulation**

12. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

**Bit 7 of port D is set to input**

12. In the PIC18 with TRISD = 0b11111111, what is the configuration of the Port D?

**Bit 8 of port D is set to input**

13. Given the short code, what is the value in AX after the program is run?

**0500**

13. Given the short code, what is the value in AX after the program is run?

**0100**

14. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?

**ZF**

15. How many nibbles are in double precision IEEE floating point format number?

**16**

16. How many nibbles are in extended precision IEEE floating point format numbers (80bit)?

**20**

17. What type of program is this?

```
AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC
1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD
```

**EXE**

18. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?

**F00D**

19. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

**A0**

20. What is the numeric sequence of the key pad columns on the PPE board?

**37, 2F, 1F**

21. What is -130 decimal in 2’s complement (8bits)?

**01111110**

22. Which of the following is a valid x86 command for multiplying a number?

**MUL BX**

23. The number of bits in single precision IEEE floating point format are:

**32**

24. You are trying to rebuild a HELLO project program in MASM and you get the following error:  
“LINK : warning L4021: no stack segment”.

What would be the reason for such an error?

**No project template for COM was selected**

25. A “pull down” resistor is used in digital circuits to do what?

**To keep the signal line “tied” low until the line is active (goes high)**

26. A “POP” instruction:

**increments the SP**

27. AND’ing 10H and 2FH will result in which of the following?

**0**

28. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

**Sets the Propeller pins P4 through P9 as output pins**

29. What commands in MASM-CodeView would be used to step through a program line by line?

**T (F8)**

30. If the SP is F00F, what will the SP value be after a “POP SP” instruction?

**F011**

31. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?

**0**

32. In the Propeller microcontroller, the term “Method” is(are) which of the following?

**A block of executable commands that has variables, can receive parameters, and returns a value.**

33. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?

**MOV CX, 30H or MOV CX, 48**

34. A “pull up” resistor is used in digital circuits to do what?

**To keep the signal “tied” high until the line is active (goes low)**

EEE-174 MT1

1. What is the hexadecimal encoding for "JNL" for a jump back 8 bytes?

**OF F8**

2. In X86 architecture, ALU stands for which of the following?

**Arithmetic Logic Unit**

3. The number of nibbles in a word are:

**16/4 = 4**

4. The instruction MOV BX, [2BAD] is what addressing mode?

**Direct**

5. What is the hexadecimal encoding for adding DX with BX and storing the result in BX?

**01CB**

6. A microprocessor with a 32-bit address bus could access how much memory?

 **$2^{32}$ =4GB**

7. You add 7+6 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:

**0111 1****OF=1, the sign bit has changed****+0110****CF=1, there is a carryon of bit 7****0|1101**

8. Which of the following DOS Debug instructions would be used change the IP register to 100?

RIP

Convert 129.C hexadecimal into decimal

297.75 \*note after decimal divide by 16

8.1 What is the RS232C, specification voltage range for the Logic 0 output?

+3v to +25v.

8.1 What is the RS232C, specification voltage range for the Logic 1 output?

-3v to -25v.

9. What are the contents of CX after this program has been run:

Memory location	Contents
5514	24
5513	D8
5512	00
5511	21

```
MOV DX, 11h  
MOV CX, [5512]  
MOV BX, 5511h  
SUB DX, [BX]
```

0000h

\*\*\*What are the contents of DX after this program has been run:

FFF0h

9. What are the contents of BX after this program?

**What are the contents of BX after this program?**

Program	Memory location	Contents
MOV BX, 2024h	2026	F2
MOV CX, 4eh	2025	59
DEC BX	2024	39
AND CX, [BX]	2023	4E

Select one:

- a. 2023h
- b. F239h
- c. 2420h
- d. 2024h
- e. 394Eh

Assuming DS = 1000h, the instruction sequence in the listing below takes the last byte in the transfer from memory at:

**Assuming DS = 1000h, the instruction sequence in the listing below takes the last byte in the transfer from memory at:**

STD
MOV CX, 500H
MOV DX, 100H
MOV SI, 250H
A1: LODSB
OUT DX, AL
LOOP A1

Select one:

- a. 10500h
- b. 10250h
- c. 1750h
- d. 10750h
- e. 0FD50h

A microprocessor with a 26-bit address bus could access how much memory?

64MB

10. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

-48

11. How many bytes are there in this short sequence of code? B815B400CD16CD20

8

12. In using INT 10h to set the video code to 640x350, what value must be in the AH register?

AX  
AH|AL  
00

00h

13. Moore's law has accurately predicted the growth rate in the number of transistors per die for the last 25 years. What is that rate?

Doubling every 18-24 months

14. INT 21h, Function 09h requires three things set up before calling in order to correctly print a string.

Hello\_msg. They are:

DS=SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h

15. Given

13A7:0110 CD 20 32 20 54 68 69 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 75 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in decimal?

50

An input buffer is at memory location 0118, what is the size of the buffer in decimal?

32 if 0118 is 50 then answer is 80

16. Here is a short sequence of code: 7413CD16EB157D213C04EBF0A3C6. All of the instructions are two bytes long. The third instruction operator is: EB15

JMP

17. What is the hexadecimal encoding for loading DX with a word (value) from memory location 0820h?

8B162008

18. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV UP EI PL NZ NA PO NC  
1D72: 010D 7D09 JGE 0116

How many bytes will the processor jump if the condition for a jump were met?

9

19. F6 in 2's complement equals \_\_\_\_\_ in base 10.

-10

21. Determine the contents of register BH after the following instructions have executed:  
70h MOV [0202], AX

22. Given:

AX=2247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72: 0106 EB0F JMP 0118

What will be the IP value be after a "t" command is executed in DOS Debug?

0118h

23. Which of the following DOS Debug instructions would set a break point at memory location 010E?  
G=100 10E

24. Given:

0B0E: 0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D  
0B0E: 0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message?

Welcome to Assembly Language

25. Given:

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0104 NV UP EI NG NZ NA PE NC  
1D72: 0104 7002 JO 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0106h

26. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO CY  
1D72: 0109 7D06 JNL 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0110

27. The instruction MOV BX, 2BAD is what addressing mode?

Immediate

28. What is the hexadecimal encoding for "JNL" for a jump back 10 bytes?

7DF4

29. How many address lines would be required to address 64MB directly?

26 / $2^{26}$ =67mb/

30. The number of nibbles in a double word are: 8

31. In using INT 10h to move the screen cursor to return on the same line, what value must be in the AX register?

0E0Dh

32. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV U EI PL NZ NA PO NC  
1D72: 010D 7D09 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug?

010Fh

33. Which of the following DOS Debug instructions would be used change the AX register?  
RAX

34. How many Bytes are there in this short sequence of code? B815B400CD168A3CCD20  
10

35. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
1D72: 0111 EB08 JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug?

0119h

36. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI NG NZ NA PO CY

1D72: 010E 7D06 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0118

37. How many bytes decimal will the program jump for JNB instruction, given the following?

AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1376 ES=1376 SS=1376 CS=1376 IP=0100 NV UP EI PL NZ NA PO NC

1376:0100 73E0 JNB 00E2

-32

38. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV UP EI PL NZ NA PO NC

1D72: 010D 7F09 JNLE 0118

What is the decimal value of the signed number in the AX register?

-0032

39. How many address lines would be required to address 512 MB directly?

29 / $2^9=29$ /

40. What is the hexadecimal encoding for adding BX with CX and storing the result in BX?

03CB or 03D9 -- 02CB or 02D9 - these are the only options

41. The binary number, 0111 1110, represents what values; in Hex, and as a BCD number?

7E, 7 invalid

42. In using INT 10h to set the video mode to 640 X 200, what value must be in the AX register?

0006h

43. what is the hexadecimal encoding for loading AH with a word from memory location 0520h?

8B262005 (choose 8B162005 if that is only option)

44. What is the hexadecimal encoding for "JGE" for a jump back 10 bytes?

7DF6

45. Given:

13A7:0110 CD 20 48 20 54 68 69 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0120 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in bytes in decimal?

72

46. Given:

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC

1D72:0104 7002 JNO 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0118

47. Determine the contents of register AH after the following instruction have executed:

MOV BX,BA70H

MOV AX,47E7H

MOV BL, 0FH

47H

AND AL,BL

MOV [0202], AX

48. In using INT 10h to move the screen cursor to return to the beginning of the line, what value must be in the AX register?

0E0Dh

49. Given:

1376:0110 48 61 76 65 20 61 20 67-72 65 61 74 20 53 70 72

1376:0120 69 6E 67 20 52 65 63 65-73 73 21 00 00 00 00 00

An ASCII message begins at memory location 0110, what is the message?

Have a great Spring Recess!

50. You add 1+8 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:

OF=0, CF=0

## Page 16

51. What is the advantage of Assembly Language over C Language?

The Assembler creates much faster executable code

52. In The X86 lab 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence "Hello World 0"?

020E

53. Which of the following DOS Debug instructions would set a break point at memory location 010C?

G = 100 10C

54. If CX is 0000 what will CX be after a "LOOP" instruction?

FFFF

55. Given:

AX=0353 BX=0534 CX=0000 DX=0180

DS= 1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 (OV=1,NV=1 ZR=1, NZ=0)  
010B

56. How many core does the propeller microcontroller have?

8

57. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?

7DF2 (marked right) though maybe 7DF4

58. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. instructions are a word long. The third instruction operator is:

INT

59. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?

24

60. The ASCII codes for space , space, carriage return, line feed, end of string in decimal are:

32,32,13,10,36

61. A "NOP" instruction in a program will:

Perform a No Operation

62. How many address lines would be required to address 64MB directly?

26

63. What command in DEBUG would be used to execute interrupts?

P

64. What high level language is the propeller programmed in?

Spin

65. Which of the following DOS Debug instruction would be used to change the IP register to 010C?

RIP

66. The acronym PWM used in the Parallax Propeller and MicroChip PIC18, is defined as:

Pulse Width Modulation

67. Which command would you use to execute another core in the propeller microcontroller?

Cognew

68. Given: IP=0111 NV UP EI NG NZ NA PO CY  
1D72:0111 JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug?

0119h

69. The instruction MOV CX, DADD is what addressing mode?

Immediate

69. The instruction MOV CX, [DADD] is what addressing mode?

Direct

70. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

A0

71. In the Propeller microcontroller, the command "dira[4..9] := %111111" would cause the processor to do with of the following?

Sets the Propeller pins P4 through P9 as output pins

72. In the Propeller microcontroller, the command "waitcnt(clkfreq\*10 + cnt)" would cause the processor to do with of the following?

Create 10 second delay

73. The "LOOPNE" instruction is equivalent to which of the following instructions?

DEC CX, JNE

74. On the Arduino platform, what is the programming language used?

C

EEE 174 Midterm Study Guide

S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:

a. DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36

Or the other option in hex is below

b. DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h

2. Moore's Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?
- a. **Doubling every 18 – 24 months**
3. Given:  
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118
- What will the IP value be after a “t” command is executed in DOS Debug?
- a. **010B**
4. A “NOP” instruction will:
- a. **Perform a No OPeration**
5. Given:  
AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72:0106 EB0F JMP 0118
- What will the IP value be after a “t” command is executed in DOS Debug?
- a. **0118h**
6. In x86 architecture, BIU stands for which of the following?
- a. **Bus Interface Unit**
7. 7. Determine the contents of register BL after the following instructions have been executed:
- a. **2EH**
8. 8. The number of bytes in a double word are:
- a. **4**
9. In x86 architecture, ALU stands for which of the following?
- a. **Arithmetic Logic Unit**
10. The “LOOPNE” instruction is equivalent to which of the following instructions?
- a. **DEC CX, JNE**
11. The instruction MOV CX, DADD is what addressing mode?
- a. **Immediate**
12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:
- a. **INT**
13. Which of the following DOS Debug instructions would set a break point at memory location 010C?
- a. **G = 100 10C**
14. How many bytes are there in this short sequence of code? B400CD164CCD21
- a. **7**
15. How many address lines would be required to address 128 MB directly?
- a. **27**
16. What are the contents of CX after this program has been run:
- a. **D800h**
17. 1010 0110 in 2's complement equals\_\_\_\_ in base 10.  
a. **-90**
18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?
- a. **01D8**
19. What is 11.4375<sub>10</sub> in binary?  
a. **001011.01110**
20. If CX is 0000 what will CX be after a “LOOP” instruction?  
a. **FFFF**

21. Given:

```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO NC  
1D72:010F 7D08 JGE 0115
```

How many bytes will the processor jump if the conditions for a jump were met?

a. 8

22. What command in DEBUG would be used to execute interrupts?

a. P

23. What is the advantage of C Language over Assembly Language?

a. C is transportable to other microprocessor architectures

24. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

a. OF = 1, CF = 0

25. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

a. 36

26. Given:

```
AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JNL 0116
```

a. -10

27. AND'ing 1FH and 20H will result in which of the following?

a. 0

28. Given:

```
13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 00
```

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

a. 72

29. A microprocessor with a 31-bit address bus could access how much memory?

a. 2 GB

30. For the instruction sequence below, determine the contents of the register AL after this program is executed:

a. 51H

31. Given:

```
AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118
```

What is the signed decimal value of the number in the AX register?

a. -256

32. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

a. 32, 32, 13, 10, 36

### Spring 2012

33. What is the advantage of assembly language over C language?

a. The assembler creates much faster executable code

34. How many lines would be required to address 128 mb directly?

a. 27

35. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

a. 020E

36. Given

```
W e l c o m e _ t o _ A s s e m  
0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D  
0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00  
b l y _ L a n g u a g e
```

An Ascii message begins at memory location 0200, what is the message?

- a. Welcome to Assembly Language

37. Which of the following DOS Debug instructions would be used to change the IP register to 0110?

- a. RIP

38. Moore's law has accurately predicted the growth rate in the number of transistors per ide for the last 40 years. What is the rate?

- a. Doubling every 18 – 24 months

39. Which of the following DOS debug instructions would set a break point at memory location 010C?

- a. G = 1000 10C

40. AND'ing 1FH and 02H will result in which of the following?

- a. 02

41. If CX is 0000 what will CX be after a "LOOP" instruction?

- a. FFFF

42. The number of bits in a word are:

- a. 16

43. In x86 architecture, ALU stands for which of the following?

- a. Arithmetic Logical Unit

44. Given:

```
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JGE 0118 1 0
```

What will the IP value be after a "t" command is executed in DOS DEBUG?

- a. 010B

45. What is the number 1010.0101<sub>2</sub> in decimal?

- a. 10.31

46. How many cores does the propeller microcontroller have?

- a. 8

47. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?

- a. 7DF2

48. What command in debug would be used to step through a program line by line?

- a. T

49. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long The third instruction operator is:

- a. INT

50. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?

- a. 24

51. Given:

```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 0DF6 JNL 0116 0 0 0 0
```

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- a. 10

52. Determine the contents of the register BL after the following instructions have been executed:

- a. E2H

Program Listing
MOV BL, E2H
MOV CL, 08H
ROL BL, CL

53. The ASCII codes for space, space, carriage return, line feed, end of string in hex are:

- a. 20, 20, 0D, 0A, 24

54. AND'ing 10H and 2FH will result in which of the following?

- a. 0

55. A "NOP" instruction in a program will

- a. Perform a No Operation

56. Given:

```
AX=FFF0  BX=3534  CX=0000  DX=0180  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1D72  ES=1D72  SS=1D72  CS=1D72  IP=0109  OV UP EI PL NZ NA PO CY
1D72:0109 7D06      JGE    0118
```

What is the signed decimal value of the number in the AX register?

- a. -16

57. How many address lines would be required to address 64 MB directly?

- a. 26

58. What command in debug would be used to execute interrupts?

- a. P

59. What high level language is the propeller programmed in?

- a. Spin

60. Which of the following DOS debug instructions would be used to change the IP register to 010C?

- a. RIP

61. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:

- a. Pulse width modulation

62. Which command would you use to execute another core in the propeller microcontroller?

- a. Cognew

63. Given:

```
AX=FFD0  BX=3534  CX=0000  DX=0180  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1D72  ES=1D72  SS=1D72  CS=1D72  IP=0111  NV UP EI NG NZ NA PO CY
1D72:0111 EB08      JMP    0119
```

What will the IP value be after the "t" command is executed in DOS debug?

- a. 0119h

64. The instruction MOV CD, DADD is what addressing mode?

- a. Immediate

65. How many byte in decimal will the processor jump if the conditions for a jump are met?

- a. 24

66. In the Propeller microcontroller, the command "dira[4..9] := %111111" would cause the processor to do which of the following?

- a. Sets the Propeller pins P4 through P9 as output pins

67. What are the contents of DX after this program has been run:

- a. FFF0h.

```
MOV DX, 11h
MOV CX, [5512]
MOV BX, 5511h
SUB DX, [BX]
AND BX, FFFF
```

Memory Location	Contents
5514	24
5513	D8
5512	00
5511	21
5510	00

68. The number of nibbles in a word are:

- a. 4

What are the TTL logic level voltages for a logic 0 and logic 1?

0V to +5V

What is 152.875 converted to double precision fp?

40 63 06 00 00 00 00

What is 152.1875 converted to single precision fp?

**Question 123**

Answer saved

Marked out of 1

Flag question

**What is 152.1875 Converted to single precision FP?**

Select one:

- a. 3D A0 00 00
- b. C1 1C 80 00
- c. 43 18 30 00
- d. C2 C0 8A 3D
- e. 40 63 06 00
- f. 41 1C 00 00

69. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?
- a. Create 10 second delay
70. The “LOOPNE” instruction is equivalent to which of the following instructions?
- a. DEC CX JNE
71. Given
- ```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI NG NZ NA PO NC
1D72:010D EB07           JMP    0114
```
- How many bytes will the processor jump if the conditions for a jump were met?
- a. 7
72. On the Arduino platform, what is the programming language used?
- a. C
73. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?
- a. A12008
74. Which of the following would be used to set the TRISA register to control the direction of PIC18 Port to input?
- a. 1
75. The acronym ADC in microcontrollers stands for which of the following?
- a. Analog to Digital converter
76. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:
- a. OF = 1, CF = 0

Compare and contrast the Harvard architecture with the von Neumann

**Question 99**

Answer saved

Marked out of 1

Flag question

**Compare and contrast the Harvard architecture with the von Neumann architecture.**

Select one:

- a. The von Neumann architecture uses big endian addressing.
- b. The Harvard architecture uses programmed I/O.
- c. The von Neumann architecture uses separate program and data memory.
- d. The Harvard type uses a separate program and data memory
- e. The Harvard architecture uses big endian addressing.

**von Neumann vs. Harvard**

- von Neumann
  - Same memory holds data, instructions.
  - A single set of address/data buses between CPU and memory
- Harvard
  - Separate memories for data and instructions.
  - Two sets of address/data buses between CPU and memory

**S2010 MT2**

1. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?
  - a. 7
2. If CX is 0003, what will CX be after a “LOOP” instruction?
  - a. 0002
3. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?
  - a. A 5 second delay
4. What is the number 1011.0101<sub>2</sub> in decimal?
  - a. 11.31
5. This section of memory represents a stack. What type of program is this?  
 BEEF : FF00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  
 BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
 BEEF : FFFF 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF
6. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?
  - a. BDCA
7. Determine the contents of register BL after the following instruction have been executed:
  - a. 2EH
8. What Hex values must be sent to address the key pad rows on the PPE board?
  - a. 1, 2, 4, 8
9. What is the number 32.4375<sub>10</sub> in binary?
  - a. 100000.01110
10. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

**a. 24**

11. The acronym PWM used for motor control, is defined as which of the following?

**a. Pulse Width Modulation**

12. Given the short code, what is the value in AX after the program is run?

**a. 0500**

13. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?

**a. ZF**

14. How many nibbles are in double precision IEEE floating point format number?

**a. 16**

15. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC  
1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD

**a. EXE**

16. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?

**a. F00D**

17. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

**a. A0**

18. What is the numeric sequence of the key pad columns on the PPE board?

**a. 37, 2F, 1F**

19. What is -130 decimal in 2's complement (8bits)?

**a. 01111110**

20. Which of the following is a valid x86 command for multiplying a number?

**a. MUL BX**

21. The number of bits in single precision IEEE floating point format are:

**a. 32**

22. You are trying to rebuild a HELLO project program in MASM and you get the following error:

“LINK : warning L4021: no stack segment”.

What would be the reason for such an error?

**a. No project template for COM was selected**

23. A “pull down” resistor is used in digital circuits to do what?

**a. To keep the signal line “tied” low until the line is active (goes high)**

24. A “POP” instruction:

**a. increments the SP**

25. AND'ing 10H and 2FH will result in which of the following?

**a. 0**

26. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

**a. Sets the Propeller pins P4 through P9 as output pins**

27. What commands in MASM-CodeView would be used to step through a program line by line?

**a. T (F8)**

28. If the SP is F00F, what will the SP value be after a “POP SP” instruction?

**a. F011**

29. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?

**a. 0**

30. In the Propeller microcontroller, the term “Method” is(are) which of the following?

**a. A block of executable commands that has variables, can receive parameters, and returns a value.**

31. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?

**a. MOV CX, 30H      or      MOV CX, 48**

32. A “pull up” resistor is used in digital circuits to do what?

- a. To keep the signal “tied” high until the line is active (goes low)**
33. With a PUSHA instruction, what will be the order of the register (registers A ~ D) contents on the stack?
- a. ACDB**
34. The LOOPNE instruction performs which of the following?
- a. Decrements CX, tests the ZF flag, if it is not zero jumps to address specified.**
35. What must the value be and in what register, prior to executing a LOOPNE instruction, to discontinue looping?
- a. CX =1**
36. With a POPA instruction, what will be the order of the registers (registers A~ D) restored from stack?
- a. BDCA**
37. What flags does the “LOOPNZ” instruction look at to determine whether to loop or not?
- a. ZF**
38. If the SP is F00F, what is the SP value after a “POP BX” instruction:
- a. F011**
39. If CX is 0001, what will CX be after a “LOOPNZ” instruction:
- a. 0000**
40. -11.25 in decimal converted to binary would be:
- a. -1011.0100**
41. What is 31.4375 in binary?
- a. 011111.0111**
42. Using debug which command should be used to change the flag setting?
- a. RF**
43. What is 14.4375 in binary?
- a. 001110.0110**

EEE 174 Midterm Study Guide

S2010 MT 1

2. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:
- b. DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36**
77. Moore’s Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?
- a. Doubling every 18 – 24 months**
78. Given:  
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118
- What will the IP value be after a “t” command is executed in DOS Debug?
- a. 010B**
79. A “NOP” instruction will:
- a. Perform a No OPeration**
80. Given:  
AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72:0106 EB0F JMP 0118
- What will the IP value be after a “t” command is executed in DOS Debug?
- a. 0118h**
81. In x86 architecture, BIU stands for which of the following?
- a. Bus Interface Unit**
82. 7. Determine the contents of register BL after the following instructions have been executed:
- a. 2EH**

83. 8. The number of bytes in a double word are:

- a. 4**

- 84.** In x86 architecture, ALU stands for which of the following?
- a. **Arithmetic Logic Unit**
- 85.** The “LOOPNE” instruction is equivalent to which of the following instructions?
- a. **DEC CX, JNE**
- 86.** The instruction MOV CX, DADD is what addressing mode?
- a. **Immediate**
- 87.** Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:
- a. **INT**
- 88.** Which of the following DOS Debug instructions would set a break point at memory location 010C?
- a. **G = 100 10C**
- 89.** How many bytes are there in this short sequence of code? B400CD164CCD21
- a. **7**
- 11.** How many bytes are there in this short sequence of code? B400CD16CD20
- a. **6**
- 90.** How many address lines would be required to address 128 MB directly?
- a. **27**
- 91.** What are the contents of CX after this program has been run:
- a. **D800h**
- 
- 92.** 1010 0110 in 2's complement equals \_\_\_\_ in base 10.
- a. **-90**
- 93.** What is the hexadecimal encoding for adding AX with BX and storing the result in AX?
- a. **01D8**
- 94.** What is 11.4375<sub>10</sub> in binary?
- a. **001011.01110**
- 95.** If CX is 0000 what will CX be after a “LOOP” instruction?
- a. **FFFF**
- 
- 96.** Given:
- ```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO NC
1D72:010F 7D08 JGE      0115
```
- How many bytes will the processor jump if the conditions for a jump were met?
- a. **8**
- 97.** What command in DEBUG would be used to execute interrupts?
- a. **P**
- 98.** What is the advantage of C Language over Assembly Language?
- a. **C is transportable to other microprocessor architectures**
- 99.** In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:
- a. **OF = 1, CF = 0**
- 100.** In MASM, with a “MOV CX, 24h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
- a. **36**
- 101.** Given:
- ```
AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC
1D72:010D 7DF6 JNL      0116
```
- a. **-10**

102. AND'ing 1FH and 20H will result in which of the following?  
 a. **0**
103. Given:  
 13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20  
 13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
 13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00
- An input buffer is at memory location 0115, what is the size of the buffer in decimal?  
 a. **72**
104. A microprocessor with a 31-bit address bus could access how much memory?  
 a. **2 GB**
105. For the instruction sequence below, determine the contents of the register AL after this program is executed:  
 a. **51H**
106. Given:  
 AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
 1D72:0109 7D06 JNL 0118
- What is the signed decimal value of the number in the AX register?  
 a. **-256**
107. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:  
 a. **32, 32, 13, 10, 36**
- Spring 2012**
108. What is the advantage of assembly language over C language?  
 a. The assembler creates much faster executable code
109. How many lines would be required to address 128 mb directly?  
 a. 27
110. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?  
 a. 020E
111. Given  
 W e l c o m e \_ t o \_ A s s e m  
 0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D  
 0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00  
 b l y \_ L a n g u a g e
- An Ascii message begins at memory location 0200, what is the message?  
 a. Welcome to Assembly Language
112. Which of the following DOS Debug instructions would be used to change the IP register to 0110?  
 a. RIP
113. Moore's law has accurately predicted the growth rate in the number of transistors per ide for the last 40 years.  
 What is the rate?  
 a. Doubling every 18 – 24 months
114. Which of the following DOS debug instructions would set a break point at memory location 010C?  
 a. G = 1000 10C
115. AND'ing 1FH and 02H will result in which of the following?  
 a. 02
116. If CX is 0000 what will CX be after a “LOOP” instruction?  
 a. FFFF
117. The number of bits in a word are:  
 a. 16
118. In x86 architecture, ALU stands for which of the following?  
 a. Arithmetic Logical Unit

119. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JGE 0118

What will the IP value be after a “t” command is executed in DOS DEBUG?

- a. 010B
- 120. What is the number 1010.0101<sub>2</sub> in decimal?
  - a. 10.31
- 121. How many cores does the propeller microcontroller have?
  - a. 8
- 122. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?
  - a. 7DF2
- 123. What command in debug would be used to step through a program line by line?
  - a. T
- 124. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:
  - a. INT
- 125. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
  - a. 24

126. Given:

AX=FFEO BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JNL 0116

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- a. 10
- 127. Determine the contents of the register BL after the following instructions have been executed:
  - a. E2H

| Program Listing |
|-----------------|
| MOV BL, E2H     |
| MOV CL, 08H     |
| ROL BL, CL      |

- 128. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
  - a. 20, 20, 0D, 0A, 24
- 129. AND’ing 10H and 2FH will result in which of the following?
  - a. 0
- 130. A “NOP” instruction in a program will
  - a. Perform a No Operation
- 131. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JGE 0118

What is the signed decimal value of the number in the AX register?
  - a. -16
- 132. How many address lines would be required to address 64 MB directly?
  - a. 26
- 133. What command in debug would be used to execute interrupts?
  - a. P
- 134. What high level language is the propeller programmed in?
  - a. Spin
- 135. Which of the following DOS debug instructions would be used to change the IP register to 010C?
  - a. RIP
- 136. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:
  - a. Pulse width modulation
- 137. Which command would you use to execute another core in the propeller microcontroller?

- a. Cognex

138. Given:

```
AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY
1D72:0111 EB08 JMP 0119
```

What will the IP value be after the “t” command is executed in DOS debug?

- a. 0119h

139. The instruction MOV CD, DADD is what addressing mode?

- a. Immediate

140. How many byte in decimal will the processor jump if the conditions for a jump are met?

- a. 24

141. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

- a. Sets the Propeller pins P4 through P9 as output pins

142. What are the contents of DX after this program has been run:

- a. FFF0h.

| Memory Location | Contents |
|-----------------|----------|
| 5514            | 24       |
| 5513            | D8       |
| 5512            | 00       |
| 5511            | 21       |
| 5510            | 00       |

143. The number of nibbles in a word are:

- a. 4

144. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?

- a. Create 10 second delay

145. The “PPE board E” instruction is equivalent to which of the following instructions?

- a. DEC CX JNE

146. Given

```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI NG NZ NA PO NC
1D72:010D EB07 JMP 0114
```

How many bytes will the processor jump if the conditions for a jump were met?

- a. 7

147. On the Arduino platform, what is the programming language used?

- a. C

148. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?

- a. A12008

149. Which of the following would be used to set the TRISA register to control the direction of PIC18 Port to input?

- a. 1

150. The acronym ADC in microcontrollers stands for which of the following?

- a. Analog to Digital converter

151. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

- a. OF = 1, CF = 0

S2010 MT2

2. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?  
    **b. 7**
44. If CX is 0003, what will CX be after a “LOOP” instruction?  
    **a. 0002**
45. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?
- a. A 5 second delay**
46. What is the number 1011.0101<sub>2</sub> in decimal?  
    **a. 11.31**
47. This section of memory represents a stack. What type of program is this?  
    BEEF : FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  
    BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
    BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF
- a. COM program**
48. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?  
    **a. BDCA**
49. Determine the contents of register BL after the following instruction have been executed:  
    **a. 2EH**
50. What Hex values must be sent to address the key pad rows on the PPE board?  
    **a. 1, 2, 4, 8**
51. What is the number 32.4375<sub>10</sub> in binary?  
    **a. 100000.01110**
52. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
    **a. 24**
53. The acronym PWM used for motor control, is defined as which of the following?  
    **a. Pulse Width Modulation**
54. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?  
    **a. Bit 7 of port D is set to output**
55. Given the short code, what is the value in AX after the program is run?  
    **a. 0500**
56. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?  
    **a. ZF**
57. How many nibbles are in double precision IEEE floating point format number?  
    **a. 16**
58. What type of program is this?  
    AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
    DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC  
    1376:0015 0100 ADD [BI + SI], AL                                                  DS : 0000=CD  
    **a. EXE**
59. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?  
    **a. F00D**
60. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

- a. A0**
61. What is the numeric sequence of the key pad columns on the PPE board?  
**a. 37, 2F, 1F**
  62. What is -130 decimal in 2's complement (8bits)?  
**a. 0111110**
  63. Which of the following is a valid x86 command for multiplying a number?  
**a. MUL BX**
  64. The number of bits in single precision IEEE floating point format are:  
**a. 32**
  65. You are trying to rebuild a HELLO project program in MASM and you get the following error:  
“LINK : warning L4021: no stack segment”.  
What would be the reason for such an error?  
**a. No project template for COM was selected**
  66. A “pull down” resistor is used in digital circuits to do what?  
**a. To keep the signal line “tied” low until the line is active (goes high)**
  67. A “POP” instruction:  
**a. increments the SP**
  68. AND’ing 10H and 2FH will result in which of the following?  
**a. 0**
  69. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?  
**a. Sets the Propeller pins P4 through P9 as output pins**
  70. What commands in MASM-CodeView would be used to step through a program line by line?  
**a. T (F8)**
  71. If the SP is F00F, what will the SP value be after a “POP SP” instruction?  
**a. F011**
  72. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?  
**a. 0**
  73. In the Propeller microcontroller, the term “Method” is(are) which of the following?  
**a. A block of executable commands that has variables, can receive parameters, and returns a value.**
  74. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?  
**a. MOV CX, 30H      or      MOV CX, 48**
  75. A “pull up” resistor is used in digital circuits to do what?  
**a. To keep the signal “tied” high until the line is active (goes low)**

## SECOND DOCUMENT

### S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:  
**DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36**
2. Moore’s Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?

#### Doubling every 18 – 24 months

3. Given:  
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118  
What will the IP value be after a “t” command is executed in DOS Debuge range of decimal numbers range of decimal numbers g?  
**010B**
4. A “NOP” instruction will:  
**Perform a No OPeration**
5. Given:  
AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72:0106 EB0F JMP 0118

What will the IP value be after a “t” command is executed in DOS Debug?

**0118h**

6. In x86 architecture, BIU stands for which of the following?

**Bus Interface Unit**

7. Determine the contents of register BL after the following instructions have been executed:

**2EH**

8. The number of bytes in a double word are:

**4**

9. In x86 architecture, ALU stands for which of the following?

**Arithmetic Logic Unit**

10. The “LOOPNE” instruction is equivalent to which of the following instructions?

**DEC CX, JNE**

11. The instruction MOV CX, DADD is what addressing mode?

**Immediate**

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

**INT 16**

13. Which of the following DOS Debug instructions would set a break point at memory location 010C?

**G = 100 10C**

14. How many bytes are there in this short sequence of code? B400CD164CCD21

**7**

15. How many address lines would be required to address 128 MB directly?

**27**

16. What are the contents of CX after this program has been run:

**D800h**

17. 1010 0110 in 2's complement equals \_\_\_\_ in base 10.

**-90**

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

**01D8**

19. What is 11.4375<sub>10</sub> in binary?

**001011.01110**

20. If CX is 0000 what will CX be after a “LOOP” instruction?

**FFFF**

22. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO NC  
1D72:010F 7D08 JGE 0115

How many bytes will the processor jump if the conditions for a jump were met?

**8**

23. What command in DEBUG would be used to execute interrupts?

**P**

24. What is the advantage of C Language over Assembly Language?

**C is transportable to other microprocessor architectures**

25. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

**OF = 1, CF = 0**

26. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

**36**

27. Given:

AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JNL 0116

**-10**

28. AND'ing 1FH and 20H will result in which of the following?

**0**

29. Given:

13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

**72**

30. A microprocessor with a 31-bit address bus could access how much memory?

**2 GB**

31. For the instruction sequence below, determine the contents of the register AL after this program is executed:

**51H**

32. Given:

AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

**-256**

33. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

**32, 32, 13, 10, 36**

S2010 MT2

1. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?

**7**

2. If CX is 0003, what will CX be after a “LOOP” instruction?

**0002**

3. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?

**A 5 second delay**

4. What is the number 1011.0101<sub>2</sub> in decimal?

**11.31**

5. This section of memory represents a stack. What type of program is this?

BEEF : FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00

BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F

BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

**COM program**

6. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

**BDCAB**

7. Determine the contents of register BL after the following instruction have been executed:

**2EH**

8. What Hex values must be sent to address the key pad rows on the PPE board?

**1, 2, 4, 8**

9. What is the number 32.4375<sub>10</sub> in binary?

**100000.01110**

10. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

**24**

11. The acronym PWM used for motor control, is defined as which of the following?

**Pulse Width Modulation**

12. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

**Bit 7 of port D is set to output**

13. Given the short code, what is the value in AX after the program is run?

**0500**

14. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?

**ZF**

15. How many nibbles are in double precision IEEE floating point format number?

**16**

17. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC

1376:0015 0100 ADD [BX + SI], AL DS : 0000=CD

**EXE**

18. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?

**F00D**

19. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

**A0**

20. What is the numeric sequence of the key pad columns on the PPE board?

**37, 2F, 1F**

21. What is -130 decimal in 2’s complement (8bits)?

**01111110**

22. Which of the following is a valid x86 command for multiplying a number?

**MUL BX**

23. The number of bits in single precision IEEE floating point format are:

**32**

24. You are trying to rebuild a HELLO project program in MASM and you get the following error:

“LINK : warning L4021: no stack segment”.

What would be the reason for such an error?

**No project template for COM was selected**

25. A “pull down” resistor is used in digital circuits to do what?

**To keep the signal line “tied” low until the line is active (goes high)**

26. A “POP” instruction:

**increments the SP**

27. AND’ing 10H and 2FH will result in which of the following?

**0**

28. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

**Sets the Propeller pins P4 through P9 as output pins**

29. What commands in MASM-CodeView would be used to step through a program line by line?

**T (F8)**

30. If the SP is F00F, what will the SP value be after a “POP SP” instruction?

**F011**

31. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?

**0**

32. In the Propeller microcontroller, the term “Method” is(are) which of the following?

**A block of executable commands that has variables, can receive parameters, and returns a value.**

33. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?

**MOV CX, 30H or MOV CX, 48**

34. A “pull up” resistor is used in digital circuits to do what?

**To keep the signal “tied” high until the line is active (goes low)**

How many bit(s) is/are required to represent a range of decimal numbers from 0 to 9?

a) 4

A “POP” instruction:

a) increments the Stack Pointer, SP

The instruction MOV CX, SI is what addressing mode?

a) Register Addressing - direct

The instruction MOV CX, [SI] is what addressing mode?

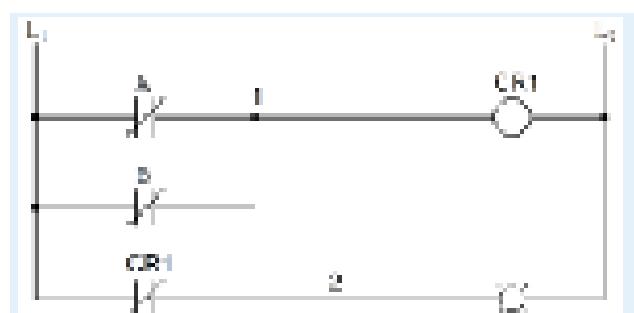
a) Register Indirect

Which of the following will cause a program with a LOOP instruction to loop 48 times (decimal)?

a) CX = 30h

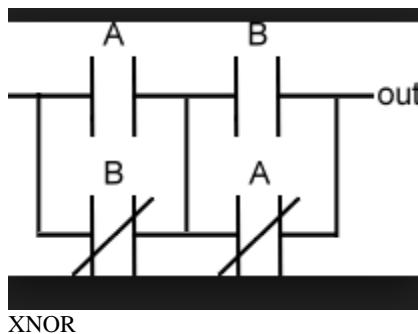
A “PUSH” instruction:

a) Decrements the SP/increments the IP



The Ladder Logic diagram would represent which of the following?

a) NOR



If you want to use a INT software interrupt function to print a string out to the screen, what is the function code, start pointer, termination character, and interrupt you need to use?

Select one:

- a) ah = 09h, ds:dx, "\$", 21h

You are trying to rebuild a HELLO project program in MASM and you get the following error: "LINK: fatal error L1089: HELLO.lrf: cannot open response file".

What would be the reason for such an error?

Select one:

- a) No source file is identified (no.asm file)

The instruction MOV SI, 4DAD is what addressing mode?

Select one:

- a) Immediate

How many nibble are in double precision IEEE floating point format numbers (64bit)?

Select one:

- a) 16

What flag(s) does the "LOOPNE" instruction look at to determine whether to loop or not?

- a) ZF

What is the numeric sequence of the key pad columns on the PPE board?

Select one:

- a) 37,2F,1F

18 hexadecimal would be what value in decimal?

- a) 24

In the PIC 18 with TRISD = 0b01000000, what is the configuration of the Port D?

Select one:

- a) Bit 6 of port D is set to input

In the PIC 18 with TRISD = 0b10000000, what is the configuration of the Port D?

Select one:

- a) Bit 7 of port D is set to input

What register(s) does the "LOOPNE" instruction look at to determine how many times to loop?

Select one:

- a) CX

If CX is 0003, what will CX be after a “LOOPNZ” instruction?

Select one:

- a) 0002

What type of program is this?

```
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1476 ES=1576 SS=1676 CS=1376 IP=0015 NV UP EI PL NE NR PO NC
1376:0015 0100 ADD [BX+SI],AL DS:0000=CD
```

Select one:

- a) EXE program

Determine the content of register BL after the following instruction have been executed:

- a) E2H

```
MOV BL, 2EH
MOV CL, 0100b
ROL BL, CL
```

#### OTHER

A “pull down” resistor is used in digital circuits to do what?

- b) To keep the signal line “tied” low until the line is active (goes high)

A “pull up” resistor is used in digital circuits to do what?

- c) To keep the signal “tied” high until the line is active (goes low)

Double precision IEEE FP standard uses \_\_\_\_\_ nibbles to represent data:

- a) 16

How many bytes are in double precision IEEE floating point format numbers?

- b) 8

You are trying to rebuild a Hello project program in MASM and you get the following error: “LINK : warning L4021: no stack segment”.

- c) No project template for COM was selected

How many double words are in double IEEE floating point format numbers?

- d) 16

In x86 architecture, BIU stands for which of the following?

- e) Bus Interface Unit

Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is that rate?

b) Doubling every 18 – 24 months

NMI stands for what?

- a) Non Maskable Interrupt – it means it cannot be blocked

The letters “NC” labeled on relays and PLCs means which of the following?

- b) Normally Closed

The number of bits in single precision IEEE floating point format are:

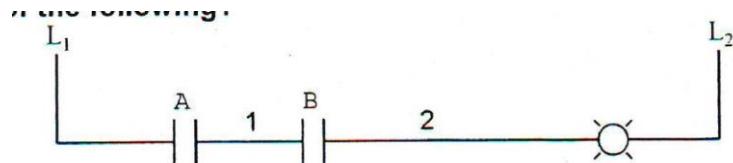
- c) 32

The number of bytes in extended precision IEEE floating point format are:

- b) 10

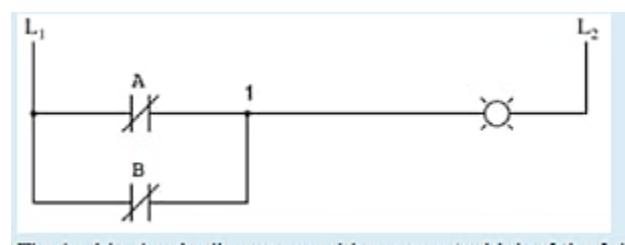
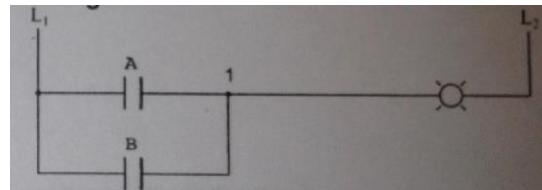
The Ladder Logic diagram would represent which of the following?

- a) XOR
- b) OR
- c) NAND
- d) AND
- e) OPEN CIRCUIT



The Ladder Logic diagram would represent which of the following?

- a) XOR
- b) OR
- c) NAND
- d) AND
- e) OPEN CIRCUIT



**NOR**

What is the decimal value of C1 1C 00 00 in IEEE single precision FP format?

- d) -9.75

What is the advantage of Assembly Language over C Language?

- a) The assembler creates much faster executable code

What is(are) the advantage(s) of C Language over Assembly Language?

- a) C is transportable to other microprocessor architectures

What is the decimal value of C5 5A 57 00 in IEEE single precision FP format?

- b) -3493.4375

PPE

On the PPE board, what number(s) on the key pad is(are) processed for an output port value of 08h and an input port value of 2Fh?

- a) 0

On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 01h and an input port value of 1Fh?

- b) 3

On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 02h and an input port value of 2fh?

- e) 5

On the PPE board, what number(s) on the key pad is(are) pressed for an output port of 04h and an input port value of 2Fh?

- c) 8

What Hex values must be sent to address the key pad rows on the PPE board?

- c) 1,2,4,8

On the raspberry pi platform, what is the programming language used?

Various open source languages

What is the numeric sequence of the key pad columns on the PPE board?

- d) 37, 2F, 1F

What is the numeric sequence to address the key pad rows on the PPE board used in the lab?

- a) 1,2,4,8

Arduino

Which of the following instruction would be used to set the LED to light on the Arduino Platform?

- e) LED = 1

On the Arduino platform, what is the programming language used?

- d) C

Microchip PICKIT

How many bits does the PIC 18 microcontroller used in the PICkit 3 Debug Express have?

- b) 16

In the PIC18 with TRISD = 0b10000000, what is the configuration of the Port D?

- a) Bit 7 of port D is set to input

In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

- b) A0

In the PIC18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

e) 0A

In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

c) Bit 7 of port D is set to output

Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?

d) 1

A “POPstruction:  
Increments the SP

With a “POPAX” instruction, what will the order of the accumulator, base, count, and data registers restored from the stack?  
ax

DAS used for BCD operation stand for which of the following?

Decimal adjust for subtraction

What is -1011.0101 in decimal?

-11.31

what is -32.75 in binary?

-100000.11000

what command in debug would be used to change the IP value?

RIP

What is -130 decimal in 2's complement (8 bits)?

01111110

what is the decimal value of C5 5A 57 00 in IEEE single precision FP format?

-3493.4375

What is -34 decimal in 2's complement?

1101 1110

EEE 174 Midterm Study Guide

S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:
  - a. **DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36**
2. Moore's Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?
  - a. **Doubling every 18 – 24 months**
3. Given:  
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

What will the IP value be after a “t” command is executed in DOS Debug?

a. **010B**

4. A “NOP” instruction will:

**a. Perform a No Operation**

5. Given:

```
AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72:0106 EB0F JMP 0118
```

What will the IP value be after a “t” command is executed in DOS Debug?

**a. 0118h**

6. In x86 architecture, BIU stands for which of the following?

**a. Bus Interface Unit**

7. 7. Determine the contents of register BL after the following instructions have been executed:

**a. 2EH**

8. 8. The number of bytes in a double word are:

**a. 4**

9. In x86 architecture, ALU stands for which of the following?

**a. Arithmetic Logic Unit**

10. The “LOOPNE” instruction is equivalent to which of the following instructions?

**a. DEC CX, JNE**

11. The instruction MOV CX, DADD is what addressing mode?

**a. Immediate**

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

**a. INT**

13. Which of the following DOS Debug instructions would set a break point at memory location 010C?

**a. G = 100 10C**

14. How many bytes are there in this short sequence of code? B400CD164CCD21

**a. 7**

15. How many address lines would be required to address 128 MB directly?

**a. 27**

16. What are the contents of CX after this program has been run:

**a. D800h**

17. 1010 0110 in 2's complement equals \_\_\_\_ in base 10.

**a. -90**

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

**a. 01D8**

19. What is 11.4375<sub>10</sub> in binary?

**a. 001011.01110**

20. If CX is 0000 what will CX be after a “LOOP” instruction?

**a. FFFF**

21. Given:

```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO NC  
1D72:010F 7D08 JGE 0115
```

How many bytes will the processor jump if the conditions for a jump were met?

**a. 8**

22. What command in DEBUG would be used to execute interrupts?

**a. P**

23. What is the advantage of C Language over Assembly Language?

- a. C is transportable to other microprocessor architectures**
24. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:  
**a. OF = 1, CF = 0**
25. In MASM, with a “MOV CX, 36” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
**a. 36**
26. Given:  
AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JNL 0116  
**a. -10**
27. AND’ing 1FH and 20H will result in which of the following?  
**a. 0**
28. Given:  
13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00  
An input buffer is at memory location 0115, what is the size of the buffer in decimal?  
**a. 72**
- An input buffer is at memory location 0114, what is the size of the buffer in decimal?  
**a. 84**
29. A microprocessor with a 31-bit address bus could access how much memory?  
**a. 2 GB**
30. For the instruction sequence below, determine the contents of the register AL after this program is executed:  
**a. 51H**

31. Given:

AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

**a. -256**

32. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:  
**a. 32, 32, 13, 10, 36**

### Spring 2012

33. What is the advantage of assembly language over C language?  
**a. The assembler creates much faster executable code**
34. How many lines would be required to address 128 mb directly?  
**a. 27**
35. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?  
**a. 020E**
36. Given  
W e l c o m e \_ t o \_ A s s e m  
0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D  
0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00  
b l y \_ L a n g u a g e
- An Ascii message begins at memory location 0200, what is the message?  
**a. Welcome to Assembly Language**
37. Which of the following DOS Debug instructions would be used to change the IP register to 0110?

- a. RIP
38. Moore's law has accurately predicted the growth rate in the number of transistors per ide for the last 40 years. What is the rate?
- a. Doubling every 18 – 24 months
39. Which of the following DOS debug instructions would set a break point at memory location 010C?
- a. G = 1000 10C
40. AND'ing 1FH and 02H will result in which of the following?
- a. 02
41. If CX is 0000 what will CX be after a "LOOP" instruction?
- a. FFFF
42. The number of bits in a word are:
- a. 16
43. The number of bits in a double word are:
- a. 32
44. In x86 architecture, ALU stands for which of the following?
- a. Arithmetic Logical Unit
45. Given:

```

AX=0353  BX=0534  CX=0000  DX=0180  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1D72  ES=1D72  SS=1D72  CS=1D72  IP=0109  OV UP EI PL NZ NA PO CY
1D72:0109 7D06    JGE   0118

```

What will the IP value be after a "t" command is executed in DOS DEBUG?

- a. 010B
46. What is the number 1010.0101<sub>2</sub> in decimal?
- a. 10.31
47. How many cores does the propeller microcontroller have?
- a. 8
48. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?
- a. 7DF2
49. What command in debug would be used to step through a program line by line?
- a. T
50. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:
- a. INT
51. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?
- a. 24
52. Given:

```

AX=FFEO  BX=3534  CX=0000  DX=0180  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1D72  ES=1D72  SS=1D72  CS=1D72  TP=010D  NV UP EI PL NZ NA PO NC
1D72:010D 7DF6    JNL   011F

```

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- a. 10
53. Determine the contents of the register BL after the following instructions have been executed:
- a. E2H

| Program Listing |
|-----------------|
| MOV BL, E2H     |
| MOV CL, 08H     |
| ROL BL, CL      |

54. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- a. 20, 20, 0D, 0A, 24
55. AND'ing 10H and 2FH will result in which of the following?
- a. 0
56. A "NOP" instruction in a program will
- a. Perform a No Operation
57. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 NV UP EI PL NZ NA PO CY  
 1D72:0109 7D06 JGE 0118

What is the signed decimal value of the number in the AX register?

- a. -16
58. How many address lines would be required to address 64 MB directly?
- a. 26
59. What command in debug would be used to execute interrupts?
- a. P
60. What high level language is the propeller programmed in?
- a. Spin
61. Which of the following DOS debug instructions would be used to change the IP register to 010C?
- a. RIP
62. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:
- a. Pulse width modulation
63. Which command would you use to execute another core in the propeller microcontroller?
- a. Cognew
64. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
 1D72:0111 EB08 JMP 0119

What will the IP value be after the “t” command is executed in DOS debug?

- a. 0119h
65. The instruction MOV CD, DADD is what addressing mode?
- a. Immediate
66. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?
- a. F0
67. How many byte in decimal will the processor jump if the conditions for a jump are met?
- a. 24
68. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?
- a. Sets the Propeller pins P4 through P9 as output pins

69. What are the contents of DX after this program has been run:
- a. FFF0h.

| Memory Location | Contents |
|-----------------|----------|
| 5514            | 24       |
| 5513            | D8       |
| 5512            | 00       |
| 5511            | 21       |
| 5510            | 00       |

70. The number of nibbles in a word are:
- a. 4
71. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?
- a. Create 10 second delay
72. The “LOOPNE” instruction is equivalent to which of the following instructions?
- a. DEC CX JNE

73. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI NG NZ NA PO NC  
 1D72:010D EB07 JMP 0114

How many bytes will the processor jump if the conditions for a jump were met?

- a. 7
74. On the Arduino platform, what is the programming language used?
- a. C
75. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?
- a. A12008
76. Which of the following would be used to set the TRISA register to control the direction of PIC18 Port to input?

- a. 1
77. The acronym ADC in microcontrollers stands for which of the following?  
 a. Analog to Digital converter
78. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:  
 a. OF = 1, CF = 0
79. What is the advantage of Assembly Language over C Language?  
 a. **The Assembler creates much faster executable code**
80. How many address lines would be required to address 128 MB directly?  
 a. 27
81. In the x86 lab part Hello MASM program in the original code, what is the address of the byte used to start number in the sequence “Hello World 0”?  
 a. **020E**
82. Given:  
 0B0E : 0200        57 65 6C 63 F 6D 65 20-74 6F 20 41 73 73 65 6D  
 0B0E: 0210        62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00  
 An ASCII message begins at memory location 0200, what is the message?
- a. **Welcome to Assembly Language**
83. Which of the following DOS Debug instructions would be used to change the IP register to 0110?  
 a. **RIP**
84. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?  
 a. **Doubling every 18-24 months**
85. Which of the following DOS Debug instructions would set a break point at memory location 010C?  
 a. **G = 100 10C**
86. AND’ing 1FH and 02H will result in which of the following?  
 a. **02**
87. If CX is 0000 what will CX be after a “LOOP” instruction?  
 a. **FFFF**
88. The number of bits in a word are:  
 a. **16**
89. In x86 architecture, ALU stands for which of the following?  
 a. **Arithmetic Logic Unit**
90. Given:  
 AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 0V UP EI PL NZ NA PO CY  
 1D72:0109 7D06            JGE                      0118  
 What will the IP value be after a “t” command is executed in DOS Debug?  
 a. **010B**
91. What is the number, 1010.0101<sub>2</sub> in decimal?  
 a. **10.31**
92. How many cores does the propeller microcontroller have?  
 a. **8**
93. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?  
 a. **7DF2**
94. What command in DEBUG would be used to step through a program line by line?  
 a. **T**
95. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:  
 a. **INT**
96. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
 a. **24**
97. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JN1 0116

How many bytes in decimal will the processor jump if the conditions for a jump were met?

a. **10**

98. Determine the contents of register BL after the following instructions have been executed:

a. **E2H**

99. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

a. **32, 32, 13, 10, 36**

100. AND'ing 10H and 2FH will result in which of the following?

a. **0**

101. A “NOP” instruction in a program will:

a. **Perform a No Operation**

102. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72  
SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
ID72:0109 7D06 JGE 0118

What is the signed decimal value of the number in the AX register?

• **-16**

103. How many address lines would be required to address 64 MB directly?

• **26**

104. What command in DEBUG would be used to execute interrupts?

• **P**

105. What high level language is the propeller programmed in?

• **Spin**

106. Which of the following DOS Debug Instructions would be used to change the IP register to 010C?

• **RIP**

107. The acronym PWM used in the Parallax Propeller and MicroChip PIC18, is defined as:

• **Pulse Width Modulation**

108. Which command would you use to execute another core in the propeller microcontroller?

• **Cognew**

109. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
1D72:0111 ED08 JMP 0119

What will the IP value be after a “t” command is executed in DOS Debug?

• **0119h**

2. The instruction MOV CX, DADD is what addressing mode?

• **Immediate**

110. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

• **A0**

111. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010F NV UP EI NG NZ NA PO NC  
ID72:010F 7D18 JGE 0128

How many bytes in decimal will the processor jump if the conditions for a jump are met?

• **24**

112. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

• **Sets the Propeller pins P4 through P9 as output pins**

113. What are the contexts of DX after this program has been

```

MOV DX, 11h
MOV CX, [5512]
MOV BX, 5511h
SUB DX, [BX]
AND BX, FFFF

```

| Memory Location | Contents |
|-----------------|----------|
| 5514            | 24       |
| 5513            | D8       |
| 5512            | 00       |
| 5511            | 21       |
| 5510            | 00       |

run:

- **FFF0h**

114. The number of nibbles in a word are:

- **4**

115. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?

- **Create 10 second delay**

116. The “LOOPNE” instruction is equivalent to which of the following instructions?

- **DEC CX, JNE**

117. Given:

```

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI NG NZ NA PO NC
1D72:010D EB07      JMP          0114

```

How many bytes will the processor jump if the conditions for a jump were met?

- **7**

118. On the Arduino platform, what is the programming language used?

- **C**

119. What is the hexadecimal encoding for loading AX with a word (value) from the memory location 0820h?

- **A12008**

120. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?

- **1**

121. The acronym ADC in microcontrollers stands for which of the following?

- **Analog to Digital Converter**

122. In adding 5+5 through a 4 bit integer unit. The state of the OF and the CF flags after the add instruction would be:

- **OF=1, CF=0**

123. Given:

```

0B0E:0200      57 65 6C 63 6F 6D 65 20-74 6F 20 74 68 65 20 66
0B0E:0210      69 72 73 74 20 64 61 79-20 6F 66 20 74 68 65 20
0B0E:0220      72 65 73 74 20 6F 66 20-79 6F 75 72 20 6C 69 66
0B0E:0230      65 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

An ASCII message begins at memory location 0200, what is the message?

- **Welcome to the first day of the rest of your life**

124. The instruction MOV DX, BADD is what addressing mode?

- **Immediate**

125. Which of the following is the hexadecimal encoding for adding BX with CX and storing the result in CX?

- **03CB**

126. What is the advantage of Assembly Language over C Language?

- **The Assembler creates much faster executable code**

127. What is 18.4375<sub>10</sub> in binary?

- **010010.01110**

128. For the instruction sequence below, determine the contents of the register AL after this program is executed:

- **51H**

129. In x86 architecture, ALU stands for which of the following?

- **Arithmetic Logic Unit**

130. A microprocessor with a 33-bit address bus could access how much memory?

- **8 GB**

131. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?  
**• 7DF2**
132. Given:  
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010F NV UP EI NG NZ NA PO NC  
1D72:010F 7D18 JGE 0128  
How many bytes in decimal will the processor jump if the conditions for a jump are met?  
**a. 24**
133. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years.  
What is that rate?  
**a. Doubling every 18-24 months**
134. The number of bytes in a word are:  
**a. 2**
135. Determine the contents of register BL after the following instructions have been executed:  
**a. 2EH**
136. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?  
**a. 7**
137. What high level language is the propeller programmed in?  
**a. Spin**
138. In the Propeller microcontroller, the command “dira[4..9] := %000000” would cause the processor to do which of the following?  
**a. Sets the Propeller pins P4 through P9 as input pins**
139. Which command would you use to execute another core in the propeller microcontroller?  
**a. Cognew**
140. How many cores does the propeller microcontroller have?  
**a. 8**
141. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?  
**a. Create 10 second delay**
142. The acronym ADC in microcontrollers stands for which of the following?  
**a. Analog to Digital Converter**
143. The acronym PWM used in the Parallax Propeller and Microchip PIC18, is defined as:  
**a. Pulse Width Modulation**
144. How many bits does the PIC18 microcontroller use in the PICkit 3 Debug Express have?  
**a. 8**
145. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?  
**a. 1**
146. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?  
**a. Bit 7 of port D is set to output**
147. In the PIC 18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?  
**a. 0A**
148. On the Arduino platform, what is the programming language used?  
**a. C**
149. Given:  
AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
1D72:0111 EB08 JMP 0119  
What will the IP value be after a “t” command is executed in DOS Debug?  
**a. 0119h**
150. How many bytes are there in this short sequence of code? B4 00 CD 16 4C CD 21 CD 20  
**a. 9**  
**The world's first microprocessor was developed in 1972 by?**  
**Intel**

**RISC stands for**

**Reduce instruction set computer**

151. In x86 architecture, BIU stands for which of the following?
- Bus Interface Unit**
152. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The fifth instruction operator is:
- CMP**
153. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- 32, 32, 13, 10, 36**
154. A “NOP” instruction in a program will:
- Perform a No Operation**
155. Given:  
AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118  
What is the signed decimal value of the number in the AX register?
- 16**
156. Which of the following DOS Debug instructions would set a break point at memory location 010C?
- G = 100 10C**
157. In adding 5+7 through a 4 bit integer unit, the state of the OF and CF flags after the add instruction would be:
- OF = 0, CF = 0, ZF=0**
158. Given:  
AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV UP EI NG ZR NA PO NC  
1D72:010D 7509 JNZ 0116  
What will the IP value be after a “t” command is executed in DOS Debug?
- 010Fh**
159. What are the contents of DX after this program has been run:
- FFF0h**
160. Which of the following DOS Debug instructions would be used to change the IP register to 010C?
- RIP**
161. What is the number, 1010.0101<sub>2</sub> in decimal?
- 10.31**
162. What command in DEBUG would be used to step through a program line by line?
- T**
163. AND’ing 1FH and 02H will result in which of the following?
- 02**
164. How many address lines would be required to address 64 MB directly?
- 26**
- If you want to use a INT software interrupt function to print a string out to the screen....  
Ah = 09h, ds:dx, "\$", 21h

S2010 MT2

165. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?  
a. 7
166. If CX is 0003, what will CX be after a “LOOP” instruction?  
b. 0002  
If CX is 0002, what will CX be after a “LOOP” instruction?  
C. 0001
167. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?  
d. A 5 second delay
168. What is the number 1011.0101<sub>2</sub> in decimal?  
e. 11.31
169. This section of memory represents a stack. What type of program is this?  
BEEF : FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  
BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF
- f. COM program
170. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?  
g. BDCA
171. Determine the contents of register BL after the following instruction have been executed:  
h. 2EH
172. What Hex values must be sent to address the key pad rows on the PPE board?  
i. 1, 2, 4, 8
173. What is the number 32.4375<sub>10</sub> in binary?  
j. 100000.01110
174. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
k. 24
175. The acronym PWM used for motor control, is defined as which of the following?  
l. Pulse Width Modulation
176. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?  
m. Bit 7 of port D is set to output
177. Given the short code, what is the value in AX after the program is run?  
n. 0500
178. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?  
o. ZF
179. How many nibbles are in double precision IEEE floating point format number?  
p. 16
180. What type of program is this?  
AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC  
1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD

- q. EXE**  
 181. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?  
**r. F00D**
182. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?  
**s. A0**
183. What is the numeric sequence of the key pad columns on the PPE board?  
**t. 37, 2F, 1F**
184. What is -130 decimal in 2’s complement (8bits)?  
**u. 01111110**
185. Which of the following is a valid x86 command for multiplying a number?  
**v. MUL BX**
186. The number of bits in single precision IEEE floating point format are:  
**w. 32**
187. You are trying to rebuild a HELLO project program in MASM and you get the following error:  
 “LINK : warning L4021: no stack segment”.  
 What would be the reason for such an error?  
**x. No project template for COM was selected**
188. A “pull down” resistor is used in digital circuits to do what?  
**y. To keep the signal line “tied” low until the line is active (goes high)**
189. A “POP” instruction:  
**z. increments the SP**
190. AND’ing 10H and 2FH will result in which of the following?  
**aa. 0**
191. In the Propeller microcontroller, the command “dira[4..9] := %1111111” would cause the processor to do which of the following?  
**bb. Sets the Propeller pins P4 through P9 as output pins**
192. What commands in MASM-CodeView would be used to step through a program line by line?  
**cc. T (F8)**
193. If the SP is F00F, what will the SP value be after a “POP SP” instruction?  
**dd. F011**
194. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?  
**ee. 0**
195. In the Propeller microcontroller, the term “Method” is(are) which of the following?  
**ff. A block of executable commands that has variables, can receive parameters, and returns a value.**
196. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?  
**gg. MOV CX, 30H        or        MOV CX, 48**
197. A “pull up” resistor is used in digital circuits to do what?  
**hh. To keep the signal “tied” high until the line is active (goes low)**
2. 1010 0110 in 2’s complement equals in base 10 → **\_90**
  - 3.
  4. A microprocessor with a 32-bit address bus could access how much memory → **4GB**
  5. A microprocessor with a 24-bit address bus could access how much memory → **16MB**
  - 6.
  7. A6 in 2’s complement equals in base 10 → **\_90**
  - 8.
  9. A “POP” instruction: → **Increments the SP**
  10. A “PUSH” instruction → **decrements the SP**
  11. A “NOP” instruction in a program will → **Perform a No Operation**
  12. AND’ing 1FH and 02H will result in which of the following → **\_02**
  13. AND’ing 10H and 2FH will result in which of the following → **\_0**

14. A “pull down” resistor is used in digital circuits to do what? → To keep the signal line “tied” low until the line is active (goes high)
- 15.
16. A “pull up” resistor is used in digital circuits to do what → To keep singal “tied” high until the line is active (goes low)
17. Determine the contents of register BL after the following instructions have been executed: → **E2H**
18. MOV BL, E2H
19. MOV CL, 1000b
20. Mov CL, SI is what addressing mode **REGISTER**
21. ROL BL, CL ; rotate BL by 8 to left = same value
- 22.
23. For the instruction sequence below, determine the contents of the registers AL after this program → **51H**
24. MOV AL, 73h ; 0111 0011
25. ADD AL, 78h ; 0111 1000
26. DAA
27. if low nibble of AL > 9 or AF = 1 then:
28. AL = **AL + 6**
29. AF = 1
30. if AL > 9Fh or CF = 1 then:
31. AL = **AL + 60h**
32. CF = 1
- 33.
34. For the instruction sequence below, determine the contents of the register AL after this program is executed → **51H**
- MOV AL, 83h  
ADD, AL, 68h
35. DAA
- 36.
37. Given the short code, what is the value in AX after the program is run → **0100**
- Mov BX, 0100  
PUSH BX  
MOV AX, 0500  
POP AX
- 38.
39. Given the short code, what is the value in AX after the program is run → **0001**
- MOV BX, 0001  
PUSH BX  
MOV AX, 0500  
POP AX
- 40.
- 41.
- 42.
- 43.
44. GIVEN: IP = 0106 Flags: NV UP EI NG NZ NA PE NC Instruction: JMP 011F
45. What will the IP value be after “t” command is executed in DOS Debug? → **011FH** (Unconditional Jump)
- 46.
47. GIVEN: IP = 0109 Flags: OV UP EI PL NZ NA PO CY Instruction: JGE 0118
48. What will the IP value be after a “t” command is executed in DOS Debug → **010B**
49. (0109 + 0010 add two bytes)
- 50.
51. GIVEN: IP=FFE0 Flags: OV UP EI PL NZ NA PO NC Instruction: JGE 0116: ID72:010D 7D**09**
52. How many bytes will the processor jump if the condition for a jump were met? → **2**
- 53.
- 54.
55. GIVEN: AX= FFF0 IP = 0109 FALGS: OV UP EI PL NZ NA PO CY ID72: 010F **7D18** Instruction: JGE 0118.  
What is the signed decimal value of the number in the AX register? → **-16**
56. Covert the number 7D18 into decimal.
- 57.
58. GIVEN: IP= 010F Flags: NV UP EI NG NZ NA PO NC Instruction: JNL 0115.
59. How many bytes in decimal will the processor jump if the conditions for a jump were met → **24**
- 60.
61. GIVEN: 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D
62. 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

63. An ASCII message begins at memory location 0200, what is the message? → Welcome to Assembly Language
- 64.
65. Here is a short sequence of code: 7413 EBA3 CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The third instruction operator is → INT
- 66.
67. Here is a short sequence of code: 7413 A3EB CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The fourth instruction operator is → JGE
- 68.
69. Here is a short sequence of code: B400 CD16 3C4A 7404 BC6A 7513. All of the instructions are two bytes long. The sixth instruction operator is → JNZ
- 70.
- 71.
72. How many cores does the propeller microcontroller have → 8
- 73.
74. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 15 → 4
75. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 63 → 6
76. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 127 → 7
77. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 255 → 8
78. How many bytes are there in this short sequence of code B4 00 CD 16 4C CD 20 → 7
79. How many nibbles are there in this short sequence of code B4 00 CD 16 3C 4A 74 04 3C 6A 75 13 → 24
80. How many bytes are in double precision IEEE floating point format numbers → 8
81. How many nibbles are in double precision IEEE floating point format numbers → 16
82. How many address lines would be required to address 128 MB directly → 27 ( $128 \times 1048576 = 134217728$  and  $2^{27} = 134217728$ )
83. How many address lines would be required to address 64 MB directly → 26 ( $64 \times 1048576 = 67108864$  and  $2^{26} = 67108864$ )
84. If CX is 0000, what will CX be after a “LOOP” instruction → FFFF
85. If CX is 0003, what will CX be after a “LOOPNZ” instruction → 0002
86. If the SP is F00F, what is the SP value after a “PUSH CX” instruction → F00D
87. If the SP is F00F, what is the SP value after a “POP CX” instruction → F011
88. If the SP is F00F, what is the SP value after a “POP SP” instruction → F011
89. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be → OF = 1, CF = 0
90. In x86 architecture, BIU stands for which of the following → Bus Interface Unit
91. In x86 architecture, ALU stands for which of the following → Arithmetic Logic Unit
92. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”? → 020E
93. In MASM, with a “MOV CX, 24h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal → 36
94. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, how many times will the program loop in decimal → 24
95. In MASM, with a “MOV CX, 12h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal → 18
96. In the Hello MASM lab in the original code, what is the address of the string to start the message “Hello World 0” → 0200
97. In the PIC18 with TRISD = 0b10000000, what is the configuration of the Port D → Bit 7 of port D is set to input
98. In the PIC18 with TRISD = 0b01111111 and LATD = 0xAA, what value will be on Port D and shown on the LEDS → Bit 7 of port D is set to output (because the first bit is zero = output)
99. In the PIC18 with TRISD = 0b00001111, what is the configuration of the Port D → A0 (First 4 are outputs and last four are inputs)
100. In the PIC18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDS → 0A
101. In the Propeller microcontroller, the command “dira[9..4] := %000000” would cause the processor to do which of the following → Sets the propeller pin P4 through P9 as output pins
102. In the Propeller microcontroller, the command “dira[9..4] := %111111” would cause the processor to do which of the following → Sets the propeller pin P4 through P9 as output pins

103. In the propeller microcontroller, the command “waitcnt(clkfreq\*3 + cnt)” would cause the processor to do which of the following → A 3 second delay
104. In the propeller microcontroller, the command “waitcnt(clkfreq\*2 + cnt)” would cause the processor to do which of the following → A 2 second delay
105. In the Propeller microcontroller, the term “Method” is (are) which of the following → A block of executable commands that has variables, can receive parameters, and returns a value.
106. Int 10h uses what function code to write a character to the screen and advance the cursor by one character position → 0Eh
107. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string:
108. **DS=SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h.**
- 109.
110. Ladder Logic is used in? → **PLCs**
111. **Moore's law** has accurately predicted the growth rate in the number of transistors per die for the last 40 years.  
What is the rate? → Doubling every 18-24 months
112. On the Arduino platform what is the program language used → **C**
- 113.
114. On the **PPE board**, what numbers(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh → 0
115. On the **PPE board**, what numbers(s) on the key pad is(are) pressed for an output port value of 04h and an input port value of 2Fh → 8
116. The “LOOPNZ” instruction is equivalent to which of the following instructions → DEC CX, JNE
- 117.
118. The acronym PWM used for motor control, is defined as which of the following → Pulse Width Modulation
119. The acronym PLC, is defined as which of the following? → Programmable Logic Controller
- 120.
121. The ASCII codes for space, space, carriage return, line feed, end of string in **decimal** are → 32, 32, 13, 10, 36
122. The ASCII codes for space, space, carriage return, line feed, end of string in **hexadecimal** are: → 20, 20, 0D, 0A, 24
123. The binary number, **1011 0101**, represents what values as a unsigned binary, 8 bit signed binary, odd parity ASCII, and BCD number (in that order) → 181, -76, 5, invalid5
124. The binary number, **1000 0101**, represents what values as a unsigned binary, 8 bit signed binary, odd parity ASCII, and BCD number (in that order) → 133, -123, ENQ, 85
125. The instruction MOV CX, DADD is what addressing mode → Immediate
126. The instruction MOV CX, [DADD] is what addressing mode → Direct
127. The number of bits in single precision IEEE floating point format are → 32
- 128.
- 129.
- 130.
131. This section of memory represents a stack. What type of program is this → EXE PROGRAM  
**BEEF:00D0** 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  
**BEEF:00E0** 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
**BEEF:00F0** 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF
- 132.
133. This section of memory represents a stack. What type of program is this → COM PROGRAM  
**BEEF:FFD0** 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00  
**BEEF:FFE0** 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
**BEEF:FFF0** 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF
- 134.
- 135.
136. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal → MOV CX, 48
137. Which command would you use to execute another core in the propeller microcontroller → Cognew
138. What command in DEBUG would be used to change the code segment → RCS
139. What command in DEBUG would be used to change the IP value → RIP
140. What command in DEBUG would be used to execute interrupts → P
141. What command in MASM-CodeView would be used to step through a program line by line → T(F8)
142. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not → ZF
143. What flag(s) does the “LOOPNE” instruction look at to determine whether to loop or not → ZF
144. What Hex values must be sent to address the key pad rows on the PPE board → 1,2,4,8

145. What is the advantage of C Language over Assembly Language → C is transportable to other microprocessor architectures
146. What is the hexadecimal encoding for “JGE” for a jump back 10 bytes → 7DF2
147. What is the hexadecimal encoding for adding AX with BX and storing the result in AX → 01D8
148. ADD AX, BX 000 00W 11 reg1 reg2
149. What is the hexadecimal encoding for adding BX with DX and storing the result in BX → 01D3
- 150.
151. What is -130 decimal in 2's complement (8bits) → 0111110
152. What is -32.75 in a base two number system → -100000.110000
153. What is 16.4375 in binary → 010000.01110
154. What is the binary value of decimal 12.875 → 1100.1110
- 155.
156. What is 16.4375 in binary → 010000.01110
- 157.
158. What is number, 1011.0101 (2) in decimal? → 11.31

What is the hexadecimal encoding for lading AX with a word (value from memory location 0820h

**Question 121**  
Answer saved  
Marked out of 1  
Flag question

What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?

Select one:

a. 8A262008  
 c. 8B162008 ← *Correct*  
 d. A10820  
 e. A00820

- 159.
160. What is the **numeric sequence** of the key pad columns on the PPE board → 37,2F,1F
161. What is the decimal value of C5 5A 57 00 in IEEE single precision FP format → -3493.4375
162. What of the following instruction would be used to set the LED to light on the Arudino platform → digitalWrite(ledPin, HIGH);
- 163.
164. What type of program is this → **EXE**  
 IP = **0115**, 1376:0115 0100 ADD [BX+SI], AL DS:0000=CD
- 165.
166. What type of program is this → **COM**  
 IP = **0100**, 1376:0100 0100 ADD [BX+SI], AL DS:0000=CD
- 167.
168. Which of the following DOS Debug instructions would set a break point at memory location 010C → G = 100  
10C
- 169.
170. Which of the following would be used to set the **TRISA** register to control the direction of **PIC18** port to **input** → 1 and for **output** its → 0
- 171.
172. Which of the following DOS Debug instructions would be used to change the IP register to 110 → RAX = 0110
173. What is 458752.00 Converted to double precision FP?  
 a. 41 1c 00 00 00 00
174. Which of the following will cause a program with a LOOP instruction to loop 48 times (decimal) → **CX=30h**
- 175.
176. Which of the following is a valid x86 command for multiplying a number → **MUL BX**
177. Which of the following is **not a valid** command for a number into a register in MASM → **MOV AX, BADH**

178. With a POP BX instruction, what will be order off the accumulator, base, count, and data registers restored from the stack → **BX**
179. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack → **BDCA**
180. You are trying to rebuild a HELLO program project in MASM and you get the following error: "ERROR 4 line 1". What is the cause of the error? → **Not known—this error by itself isn't a problem, press the enter key to clear the error.**
- 181.
182. You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK : warring L4021: no stack segment". What would be the reason for the such an error → **No project template for COM was selected.**
- 183.
184. You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK : fatal error L1089: HELLO.lrf: cannot open response file". What would be the reason for the such an error → **No source file is identified(no .asm file)**
- 185.
- 186.
- 187.
- 188.
- 189.
190. **PPE Row Column Scan decoding**
- |                                           |
|-------------------------------------------|
| D7 D6 D5 D4 D3 D2 D1 D0                   |
| S7 S6 S5 S4 S3 S2                         |
| ^ ^ ^                                     |
| 3 2 1                                     |
| 0 0 0 0 1 0 0 0 = 08h                     |
| 0 0 0 1 0 0 0 0 = 10h                     |
| 0 0 1 0 0 0 0 0 = 20h                     |
| 0 0 1 1 1 1 1 1 = 3Fh -> Nothing pressed  |
| 0 0 1 1 0 1 1 1 = 37h -> Number 1 pressed |
| 0 0 1 0 1 1 1 1 = 2Fh -> Number 2 pressed |
| 0 0 0 1 1 1 1 1 = 1Fh -> Number 3 pressed |
- 191.

## Flags

The complete set of possible flag mnemonics in Debug (ordered from left to right) are as follows:

| <i>Set</i>              | <i>Clear</i>             |
|-------------------------|--------------------------|
| OV = Overflow           | NV = No Overflow         |
| DN = Direction Down     | UP = Direction Up        |
| EI = Interrupts Enabled | DI = Interrupts Disabled |
| NG = Sign Flag negative | PL = Sign Flag positive  |
| ZR = Zero               | NZ = Not Zero            |
| AC = Auxiliary Carry    | NA = No Auxiliary Carry  |
| PO = Odd Parity         | PE = Even Parity         |
| CY = Carry              | NC = No Carry            |

## Jump information

| Mnemonic                   | Condition                                             |
|----------------------------|-------------------------------------------------------|
| <i>Signed Operations</i>   |                                                       |
| JG/JNLE                    | Greater/not less or equal $((SF \oplus OF) + ZF) = 0$ |
| JGE/JNL                    | Greater or equal/not less $(SF \oplus OF) = 0$        |
| JL/JNGE                    | Less/not greater or equal $(SF \oplus OF) = 1$        |
| JLE/JNG                    | Less or equal/not greater $((SF \oplus OF) + ZF) = 1$ |
| JO                         | Overflow $(OF = 1)$                                   |
| JS                         | Sign $(SF = 1)$                                       |
| JNO                        | Not overflow $(OF = 0)$                               |
| JNS                        | Not sign $(SF = 0)$                                   |
| <i>Unsigned Operations</i> |                                                       |
| JA/JNBE                    | Above/not below or equal $(CF \oplus ZF) = 0$         |
| JAE/JNB                    | Above or equal/not below $(CF = 0)$                   |
| JB/JNAE                    | Below/not above or equal $(CF = 1)$                   |
| JBE/JNA                    | Below or equal/not above $(CF \oplus ZF) = 1$         |
| <i>Either</i>              |                                                       |
| JC                         | Carry $(CF = 1)$                                      |
| JE/JZ                      | Equal/zero $(ZF = 1)$                                 |
| JP/JPE                     | Parity/parity even $(PF = 1)$                         |
| JNC                        | Not carry $(CF = 0)$                                  |
| JNE/JNZ                    | Not equal/not zero $(ZF = 0)$                         |
| JNP/JPO                    | Not parity/parity odd $(PF = 0)$                      |

Jumps always start with 7...

**Jcc – Jump if Condition is met, (see conditional jump instructions table Ch4 Uffenbeck)**  
 8-bit displacement

|            |             |
|------------|-------------|
| 0111 ttt n | byte offset |
|------------|-------------|

**Full displacement**

|           |            |             |
|-----------|------------|-------------|
| 0000 1111 | 1000 ttt n | word offset |
|-----------|------------|-------------|

| Hex      | t        | t        | t        | n        | Flag Test          | unsigned | signed   | Other |
|----------|----------|----------|----------|----------|--------------------|----------|----------|-------|
| <b>0</b> | <b>0</b> | <b>0</b> | <b>0</b> | <b>0</b> | OF = 1             |          | JO       |       |
| <b>1</b> | <b>0</b> | <b>0</b> | <b>0</b> | <b>1</b> | OF = 0             |          | JNO      |       |
| <b>2</b> | <b>0</b> | <b>0</b> | <b>1</b> | <b>0</b> | CF = 1             | JB, JNAE |          |       |
| <b>3</b> | <b>0</b> | <b>0</b> | <b>1</b> | <b>1</b> | CF = 0             | JNB, JAE |          |       |
| <b>4</b> | <b>0</b> | <b>1</b> | <b>0</b> | <b>0</b> | ZF = 1             | JE, JZ   | JE, JZ   |       |
| <b>5</b> | <b>0</b> | <b>1</b> | <b>0</b> | <b>1</b> | ZF = 0             | JNE, JNZ | JNE, JNZ |       |
| <b>6</b> | <b>0</b> | <b>1</b> | <b>1</b> | <b>0</b> | CF = 1 or ZF = 1   | JBE, JNA |          |       |
| <b>7</b> | <b>0</b> | <b>1</b> | <b>1</b> | <b>1</b> | CF = 0 and ZF = 0  | JNBE, JA |          |       |
| <b>8</b> | <b>1</b> | <b>0</b> | <b>0</b> | <b>0</b> | SF = 1             |          | JS       |       |
| <b>9</b> | <b>1</b> | <b>0</b> | <b>0</b> | <b>1</b> | SF = 0             |          | JNS      |       |
| <b>A</b> | <b>1</b> | <b>0</b> | <b>1</b> | <b>0</b> | PF = 1             |          | JP       |       |
| <b>B</b> | <b>1</b> | <b>0</b> | <b>1</b> | <b>1</b> | PF = 0             |          | JNP      |       |
| <b>C</b> | <b>1</b> | <b>1</b> | <b>0</b> | <b>0</b> | SF ≠ OF            |          | JNGE, JL |       |
| <b>D</b> | <b>1</b> | <b>1</b> | <b>0</b> | <b>1</b> | SF = OF            |          | JGE, JNL |       |
| <b>E</b> | <b>1</b> | <b>1</b> | <b>1</b> | <b>0</b> | ZF = 1 or SF ≠ OF  |          | JNG, JLE |       |
| <b>F</b> | <b>1</b> | <b>1</b> | <b>1</b> | <b>1</b> | ZF = 0 and SF = OF |          | JG, JNLE |       |

Examples:

JMP unconditional jump (same argument)  
 1110 1011: byte displacement

(EB   )

8bit-> +127 forward to -128 bytes  
 backward, expressed in Hex

1110 1001: word displacement      (E9   )

JCXZ jump if CX = 0

1110 0011: byte displacement      (E3   )  
*(to work with ECX use address size prefix)*

LOOP

1110 0010: byte displacement      (E2   )

*Include auto-decrement of the CX register.  
 Jump if CX is not zero after decrement.*

Link to a converter

<http://www.exploringbinary.com/floating-point-converter/>

Q) You add 9+9 through a 4 bit integer unit. This state of the OF and CR flags after the add will be A)01 (OF, CF)

Q) Moore's law has accurately predicted the growth rate in the number of transistors per die for the last 25 years. What is that rate.

A) Doubling every 18-24 months.

Q) In x86 architecture, ALU stands for which of the following?

A) arithmetic Logic Unit

Q) The number of nibbles in a word are:

A) 4

Q) The instruction MOV BX, [2BAD] is what addressing mode?

A) Direct

Q) What is the hexadecimal encoding for adding DX with BX and storing the result in BX? A)03CB

Q) A microprocessor with a 32-bit address bus could access how much memory?

A) 4GB

Q) You add 7+6 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:

A) OF=0, CF = 0

Q) Which of the following DOS Debug instructions would be used change the IP register to 100?

A) RIP

Q) What are the contents of CX after this program has been run: A)D800

Q) Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

What is the signed decimal value of the number in the AX register?

A) -48

Q) How many Bytes are there in this short sequence of code? B815B400CD16CD20 A)8

Q) In using INT 10h to set the video mode to 640x350, what value must be in the AH register?

A) 00h

Q) int 21h, function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:

A) DS = SEG Hello\_msg, DX = OFFSET Hello\_msg, Hello\_msg terminated with 24h

Q) Given:

13A7:0110 CD 20 32 20 54 68 69 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in decimal?

A) 50

Q) Here is a short sequence of code: 7413CD18EB157D213C04EBF0A3C6. All of the instructions are two bytes long. The third instruction operator is:

A) JMP

Q) What is the hexadecimal encoding for loading DX with a word (value) from memory location 0820h?

A) 8B162008

Q) Given:

1D72:010D JGE 0116

How many bytes will the processor jump if the conditions for a jump were met?

A) 9

Q) F6 in 2's complement equals \_\_ in base 10?

A) -10

Q) Given:

OV UP EI NG NZ NA PO NC

1D72:010D JNLE 0116

What is the IP value after a "t" command is executed in DOS debug?

A) 0116h

Q) Determine the contents of register BH after the following instructions have executed: Mov BX, 70BAh

Mov AX, E4E7h Mov BL, F0h  
AND AH, BL Mov [0202],AX

A) 70h

Q) Given:

0106 JMP 0118

What will the IP value be after a "t" command is executed in DOS Debug?

A) 0118h

Q) Which of the following DOS Debug instructions would set a break point at memory location 010E?

A) G=100 10E

Q) Given:

0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D 0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00 A)  
Welcome to Assembly Language

Q) Given:

NV UP EI NG NZ NA PE NC

1D72:0104            JO(Jump if overflow) 0118

A) 0106h

Q) Using DEBUG, which command should be used to change the flag settings?

A) RF

Q) ANDing 20H and 1FH will result in which of the following?

A) 0

Q) In MASM, with a “MOV CX, 12h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

A) 18

Q) What is the numeric sequence to address the key pad rows on the PPE board used in the lab?

A) 1,2,4,8

Q) What is 14,4375 base 10 in binary?

A) 001110.01110

Q) If the SP is F00F, what will the SP value be after a “POP CX” instruction?

A) F011

Q) What type of program is this?

DS=1476 ES=1576 SS=1676 CS=1376 IP=0015

0100     ADD [BX+SI], AL

(Since DS, ES, SS, CS are all diff = EXE)

A) EXE

Q) How many double words are in double IEEE floating point format numbers?

A) 2

Q) How many bit(s) is/are required to represent a range of number from 0 to 255?

A) 8

Q) A “NOP” instruction in a program will:

A) Perform a No Operation

Q) You are trying to rebuild a HELLO project program in MASM and you get the following error; “LINK : warning L4021: no slack segment”.

What would be the reason for such an error?

A) No project template for COM was selected.

Q) A “PUSH” instruction:

A) Decrements the SP

Q) If CX is 0000, what will CX be after a “LOOP” instruction?

A) FFFF

Q) On the PPE board, what number(s) on the key pad is (are) pressed for an output port value of 02h and an input port value of 2Fh?

A) 5

Q) With a POPAX instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

A) AX

Q) This section of memory represents a stack. What type of program is this?

BEEF:FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 BEEF:FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
BEEF:FFEF 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF A) Table 6-3: COM program

Q) Determine the contents of register BL after the following instructions have been executed: Mov BL, E2H

Mov CL, 1000b

ROL BL,CL

A) E2H

Q) What is – 1011.0101 base two in dec?

A) -11.31

Q) Given the short code, what is the value in AX after the program is run? Mov BX, 0500

Push BX

Mov AX, 0100

POP AX

A) 0500

Q) For the instruction sequence below, determine the contents of the register AL after this program is executed:

Mov AL, 82h Add AL,68h

DAA

A) 50H

Q) Which of the following is not a valid command for a number into a register in MASM?

A) MOV AX, AADH

Q) The number of bits in single precision IEEE floating point format are:

A) 32

Q) What is the numeric sequence of the key pad columns on the PPE board used in the lab?

A) 37, 2F, 1F

- Q) What command in DEBUG would be used to execute interrupts?
- A) P
- Q) Which of the following is a valid x86 command for multiplying a number?
- A) MUL BX
- Q) The number of bytes in extended precision IEEE floating point format are:
- A) 10
- Q) With a POP DX, instruction, what will be the order of the accumulator, base count, and data registers.
- A) DX
- Q) What flag does the “LOOPNZ” instruction look at the determine whether to loop or not?
- A) ZF
- Q) DAS used for BCD operations, stands for which of the following? A)Decimal Adjust for Subtraction
- Q) Double-precision IEEE FP standard uses \_\_\_\_ nibbles to represent data:
- A) 16
- Q) A “PUSH” instruction:
- A) decrements the SP
- Q) -10.25 in decimal converted to binary would be:
- A) -1010.0100
- Q) Given the short code, what is the value in AX after the program is run? Mov AX, 0200  
Mov BX, 0300 Push BX POP AX
- A) 0300
- Q) Which of the following is not a valid command for a number in MASM? A)MOV AL, C4H
- Q) What is the binary value of decimal 12.875? A)1100.1110
- Q) If the SP is F00F, what is the SP value after a “POP BX” instruction.
- A) F011
- Q) In MASM, with a “MOV CX, 10h” instruction, and a “LOOP” instruction, how many times will the program loop?
- A) 16
- Q) Determine the contents of register BL after the following instruction have been executed: MOV BL,2EH  
MOV CL, 0100b  
ROL BL,CL
- A) E2H

Q) What are the contents of AL, AH, AX, and EAX after the execution of the instruction, “MOV EAX, [30]” A  
21,43,4321,87654321

Q) If CX is 0001, what will CX be after a “LOOPNZ” instruction:

A) 0000

Q) With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

A) BDCA

Q) The ASCII codes in decimal for space, space, carriage return, line feed, end of string are:

A) 32,32,13,10,36

Q) You are trying to rebuild a HELLO project program in MASM and you get the following error:

“LINK : fatal error L1089: HELLO.lrf: cannot open response file”.

What would be the reason for such an error?

A) No source file is identified (no .asm file)

Q) On the PPE board, what number(s) on the key pad is(are) pressed for an output port value and an input port value of 1Fh?

A) 3

Q) For the instruction sequence below MOV AL, 83h

ADD AL,45h

DAA

A) 28h

Q) In string operations, register \_\_\_ is used to point to the source operand and register \_\_\_ is used to point to the destination operand.

A) SI,DI

Q) A “POPA” instruction:

A) increments the SP

Q) What flag(s) does the “LOOPNE” instruction look at to determine whether to loop or not?

A) ZF

Q) Double-precision IEEE FP standard uses \_\_\_ nibbles to represent data.

A) 16

Q) The “LOOPNE” instruction is equivalent to which of the following instructions?

A) DEX CX, JNZ

Q) In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

A) 24

- Q) The number of bytes in extended precision IEEE floating point format are:
- A) 10
- Q) If CX is 0002, what will CX be after a “LOOP” instruction?
- A) 0001
- Q) ANDing 2FH and 10H will result in which of the following
- A) 0
- Q) With a POPAX instruction, what will be the order of the accumulator, base, count, and data register restored from the stack?
- A) AX
- Q) What are the contents of AL, Ah, AX, and EAX after the execution of the instruction, “MOV EAX, 12345678H”
- A) 78, 56, 5678, 12345678
- Q) On the PPE board, what number(s) on the key pad is (are) pressed for an output port value of 04h and an input port value of 2Fh?
- A) 8
- Q) For the instruction sequence below, determine the contents of the register AL after the program is executed: MOV AL, 72h  
ADD AL, 56h  
DAA
- A) 28H
- Q) What is 31. 4375 base 10 in binary?
- A) 011111.0111
- Q) determine the contents of register BL after the following instruction have been executed: Mov BL, E2H  
Mov Cl, 1000b  
ROL BL, CL
- A) E2H
- Q) Using DEBUG which command should be used to change the flag settings?
- A) RF
- Q) The instruction MOV BX, 2BAD is what addressing mode?
- A) Immediate
- Q) What is the hexadecimal encoding for “JNL” for a jump back 10 bytes?
- A) 7DF4
- Q) How many address lines would be required to address 64 MB directly?

A) 26

Q) The number of nibbles in a double word are:

A) 8

Q) Given: IP = 010E

OV UP EI PL NZ NA PO CY JNL 0118

A) 0110

Q) What are the contents of DX after this program has been run: MOV DX, 11h

MOV CX, [5512] MOV BX, 5511h SUB DX, [BX] 5514 24

5513 D8

5512 00

5511 21

A) FFF0h

Q) In using INT 10h to move the screen cursor to return on the same line, what value must be in the AX register?

A) 0E0Dh

Q) Given: IP = 010D

OV UP EI PL NZ NA PO NC JNLE 0118

what will be the IP value after a “t” command is executed in DOS Debug?

A) 010Fh

Q) Which of the following DOS debug instructions would be used change the AX register? A)RAX

Q) IP = 0111

NV UP EI NG NZ NA PO CY

JMP 0119

after “t”

A) 0119h

Q) Given: IP = 010E

OV UP EI NG NZ NA PO CY JNL 0118

A) 0118

Q) How many bytes will the program jump for JNB instruction, given the following?

AX= 0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1376 ES=1376 SS=1376 CS=1376  
IP=0100 NV UP EI PL NZ NA PO NC

1376:0100 73E0 JNB 00E2

A) -32

Q) Given:

IP=010D

OV UP EI PL NZ NA PO NC JNLE 0118

A) -0032

Q) How many address lines would be required to address 512 MB directly? A)29

Q) What is the hexadecimal encoding for adding BX with CX and storing the result in BX?

A) 01CB

Q) The binary number, 0111 1110, represents what values; in Hex, and as a BCD number?

A) 7E, 7 invalid

Q) In using INT 10h to set the video mode to 640 x 200 what value must be in the AX register? A) 0006h

Q) how many bytes are there in this short sequence of code? CD15B400CD168A3C4A05

A) 10

Q) What is the hexadecimal encoding for loading AH with a word for memory location 0520h?

A) 8A262005

Q) What is the hexadecimal encoding for “JGE” for a jump back 10 bytes?

A) 7DF4

Q) Given:

13A7:0110 CD 30 48 20 54 68 69 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in bytes in decimal? A)72

Look at 48 4\*16 + 8 = 72

Q) Given: IP = 0106

NV UP EI NG NZ NA PE NC JNO 0118

A) 0118

Jump if not overflow.

Q) Refer to listing 5 of instructions and memory map determine the contents of registers AX, BX, and SP after the following instructions have been executed (assume SS=0000);

STD0404 3A

Mov SP, 0400h 0403 DC

POP, AX 0402 6E

POP,BX 0401 5F

A) AX= 5F48, BX= 0C6E, SP=0404

Q) What is -13 in 2's compliment (8 bits)?

A) 11110011

Q) If you want to use a DOS software interrupt function to print a string out to the screen what is the function code, start pointer, termination character, and interrupt you need to use?

A) ah= 09h, ds:dx, "\$", 21h

Q) If CS=2DF6h and IP = 0BADh compute the physical address of the next 8086 instruction fetch.

A) 2EB0Dh

MOV CX, [SI] is what addressing mode?

- a) Direct
- b) Register
- c) Scaled Index
- d) Register Indirect

The IN & OUT instruction can only transfer data between an I/O port and the \_\_\_\_\_ register.

- a) Only al
- b) Only cx
- c) cl, cx, or ecx
- d) al, ax, eax
- e) bl, bx, or ebx

The 80x86 processors have two general-purpose hardware interrupts, called \_\_\_\_\_ and \_\_\_\_\_. Of these, interrupts on \_\_\_\_\_ can be blocked by giving the \_\_\_\_\_ instruction.

- a) INTR, NMI, NMI, STI
- b) NMI, INTR, NMI, CLI
- c) INT0, INT1, INT1, CLI
- d) INTR, NMI, INTR, CLI
- e) INTA, INTB, INTA, STI

What are the contents of CX after this program:

- a) 3C7Ah
- b) 9002h
- c) 0038h

- d) 3038h
- e) F239h

| Listing       | Memory Location | Contents | Contents |
|---------------|-----------------|----------|----------|
| MOV BX, 9002h | 9003            | F2       | F2       |
| MOV CX, 3C7Ah | 9002            | 39       | 39       |
| AND CX, [BX]  | 9001            | 4E       | 4E       |

What is the binary value of -128?

- d) 1000 0000

What is the status of overflow flag, carry flag, and sign flag, after the program is run?

#### Listing Problem 2

MOV AL, FEH

MOV CH, FBH

ADD CH, AL

- A) 0,0,1

In string operation, register \_\_\_\_ is used to point to the source operand and register \_\_\_\_ is used to point to the destination operand.

SI, DI

The \_\_\_\_ flag, bit \_\_\_\_ of the register, is used to tell the CPU whether to increment or decrement pointers in repeated string operations.

- c) Directional flag, bit 11

In the following program segment, what condition will cause the REPNZ to fail?

#### Listing Problem 5

MOV SI, OFFSET DATA1

MOV DI, OFFSET DATA2

MOV CX, LENGTH

REPNZ CMPSE

When CX=0 or the point at which DATA1 and DATA2 are not equal.

What is the numeric sequence to address the key pad rows on the PPE board used in the lab? 1,2,4,8

When using DOS Debug, which command is used to execute INT instructions (to keep from changing the Code Segment)?

P

What must the value be and in what register, prior to executing a LOOPNE instruction, to discontinue looping?

The ASCII codes for carriage return and line feed are:

0Dh, 0Ah

With a POPA instruction, what will be the order of the registers (registers A~D) restored from the stack?

BDCA

What is the 9.75 in Binary?

3D A0 00 00

Double-precision IEEE FP standard uses \_\_\_\_ bit to represent data:

c) 64

What is the decimal value of 41 1C 00 00 in IEEE signal precision FP format?

9.75

What are the contents of BL, BH, BX, and EBX after the execution of the instruction, MOV EBX, 99FF77AAH:

AA, 77, 77AA, 99FF77AA

What are the contents of BX after this program:

| Listing for problem 17 | Memory Location | Contents |
|------------------------|-----------------|----------|
| MOV BX, 8002h          | 8003            | 4E       |
| MOV AX, 3C7Ah          | 8002            | 24       |
| ADD [BX], AX           | 8001            | F2       |
| DAA                    | 8000            | 39       |
|                        |                 | 8002h    |

Determine the contents of register AL after the following instructions have been executed:

Listing for Problem 18

MOV AL, 2EH  
MOV CL, 8H  
ROR AL, CL  
2EH

With a PUSHA instruction, what will be the order of the register (register A~D) contents on the stack?

ACDB

The number of nibbles in a word are:

4

Assuming DS=100h, the instruction sequence in listing takes the last byte in the transfer from memory at:

10250h

The LOOPNE instruction performs which of the following?

Decrements the CX, tests the ZF flag, if it is not zero jumps to address specified

For the instruction sequence below, determine the contents of the register AL after this program is executed:

Listing for Problem 24

MOV AL, 45

ADD AL, 65

DAA

a) 10H

The IN & OUT instruction can only transfer between an I/O port and the \_\_\_\_\_ register.

e) AL, AX, or EAX

What are the contents of BX after this program:

| Problem #1 Program | MEMORY LOCATION | CONTENTS |
|--------------------|-----------------|----------|
| MOV BX, 9002h      | 9003            | F2       |
| MOV CX, 3C7Ah      | 9002            | 39       |

AND CX, [BX] 9001 4E

9002h

The instruction sequence in listing 2 below outputs \_\_\_\_\_ consecutive bytes of memory.

500h

Assuming DS=1000h the instruction sequence in listing 3 below takes the last byte in the transfer OFFDB0h

Referring again to listing 3. The input switches are on port \_\_\_\_\_ and the output display is on port 379h, 378h

Outputs bytes from ES 1001 through ES: 100F to LO port 0A010h

The IN & OUT instructions can only transfer data between an I/O port and the al, ax, or eax register. True

The 80x86 processors have two general-purpose hardware interrupts, called \_\_\_\_\_ and \_\_\_\_\_. Of these, interrupts on \_\_\_\_\_

b) INTR, NMI, INTR, CLI

Assume the interrupt service routine for the NMI input is stored at address 0xA0:0100. What is the physical address and contents of the IVT

What is the fastest possible data transfer rate to main memory for a Pentium III processor with 133 MHz Front Side Bus? Assume you transfer 8 bytes per data transfer and you require a single clock cycle per bus cycle.

1.064 GB/s

IRQ7 has which type code output by the PIC?

0fh

Which of the following techniques does RAMBUS use to exceed DRAM access times:

Narrow bus topology with high clock rate.

Of the memory addresses listed, which one is misaligned on a 486 processor, assuming a doubleword, access? 90094005

Refer to the following figure and determine the contents of registers AX, BX, and SP after the following instructions have been executed (assume SS=0000):

```
MOV SP, 0700h  0704    3A
POP AX  0703   DC
POP BX  0702   6E
    0701   5F
    0700   48
    06FF   29
    06FE   3C
```

```
06FD   34
06FC   12
AX=5F48, BX=DC6E, SP=0704
```

The serial port interrupts on a standard PC (com1 & com2) are routed through the 8259 PIC to IRQ/INT:

IRQ3, IRQ4, Int Bh, Int Ch

The mouse (IRQ12) will be serviced before the floppy drive (IRQ6) if both interrupts reach the 3259

PIC simultaneously:

False

The third step in the DRAM read cycle is:

Turn on sense amps

The number  $1.101 \times 2^{-3}$  is encoded in single precision format for the FPU as:

3E500000h

What is 40000000h in single precision real format? Note that both operands are assumed to also be in single precision real format.

41000000h

What is the opcode for mov bx, [est82]? Assume you're in a 32-bit segment.

668B1C7500000000

Determine the contents of register BL after the following instructions have been executed:

E2H

What is -1011.0101 in decimal?

-11.31

If CX is 0000, what will CX be after a "LOOP" instruction?

e)FFFF

How many bit(s) is/are required to represent a range of numbers from 0 to 255?

8

What is 16.4375 in binary?

010000.01110

In Masm, with a "MOV CX, 12h" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?

What is the binary value of decimal 12.875?

1100.1110

What is the numeric sequence of the key pad columns on the ppe board?

37, 2F, 1F

This section of memory represents a stack. What type of program is this? BEEF: 0FD0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 BEEF: 0FE0 00 010 2 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F BEEF: 0FF0 11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF

EXE

Given the short code, what is the value in AX after the program is run?

Program Listing

MOV BX, 0100

Push BX

MOV AX, 0500

POP AX

0500

What is -32.75 in binary?

100000.11000

What command in DEBUG would be used to change the IP value?

RIP

What is the purpose of this?

AX=0000 BX=0000 CX=0000 DX=0000 SP=F1EE BP=0000

DS=1476 ES=1575 SS=1676 CS=1376 IP=0015 NV UP EI

L375:0015 0100 ADD [BX+SI] AL

EXE

16. What flag(s) does the "LOOPNZ" instruction look at to determine whether to loop or not?  
ZF

17. Which of the following is a valid x86 command for multiplying a number?

c) MUL BX

How many bytes are in double precision IEEE floating point format numbers?

8

What is -130 decimal in 2's complement (8bits)?

a) 01111110

If the SP is F00F, what will the SP value be after a "Pop CX" instruction?

a) F011

What is the decimal value of C5 5A 57 00 in IEEE single precision FP format?

-3493.4375

On the PPE board, what number(s) on the key pad is (are) pressed for an output port value of 04h and an input port value of 2Fh?  
37,2F,1F

23. You are trying to rebuild a HELLO project program in MASM and you get the following error:

"LINK: fatal error L1089: HELLO.lrf: cannot open response file". What would be the reason for such an error?

No source file is identified, no project was setup

Which of the following will cause a program with a LOOP instruction to loop 48 times?

CX=30h

If the SP is F00F, what is the SP value after a "Push CX" instruction?

F00D

What is (are) the advantages of C Language over assembly language?

Hand assembly coding is much faster in C

The number of bits in single precision IEEE floating point format are:

32

Which of the following is not valid command for a number into a register in MASM?

MOV AX, F8ADH

In the Hello MASM lab in the original code, what is the address of the byte used to start the string in the sequence “Hello World0”?

0200

You are trying to rebuild a HELLO project program in MASM and you get the following error. “LINK: warning L4021: no stack segment”.

b) No project template for COM was selected

How many nibbles are in double precision IEEE floating point format numbers?

16

A “pull down” resistor is used in digital circuits to do what?

To keep the signal line “tied” low until the line is active (goes high)

The acronym PWM used for motor control, is defined as which of the following?

Pulse Width Modulation

Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DS=1D72 ES=TD7Z SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA 1D72:01097 D06 JNL 0118

What is the signed decimal value of the number in the AX register?

-16

What are the contents of BX after this program has been run:

Memory location    Contents

|      |    |
|------|----|
| 1103 | 24 |
| 1102 | D8 |

|      |    |
|------|----|
| 1101 | 00 |
| 1100 | 21 |

1100h

What is 16.4375 in binary?

010000.01110

Here is a short sequence of code: 7413A3EBCD167D213C04EBF0EB15.

JGE

Determine the contents of register BL after the following instructions have been executed:

E2H

Program Listing

MOV BL, E2H

MOV CL, 1000b

ROL BL, CL

Which of the following DOS Debug instructions would set a break point at memory location 010E? G=100 10E

The “LOOPNZ” instruction is equivalent to which of the following instructions?

DEC CX, JNE

AX=FFEOB X=3534C X=0000D X=0180S p=FFEEB p=0000 Sr=0000 Dr=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0IOD OV UP EI PL NZ NA PO NC 1D72:010D 7D09 JGE 0116

How many bytes will the processor jump if the conditions for a jump were met?

9

A)(=0353 BX=0534c X=0000 DX=0180S P=FFEEB P.=0000S r=0000 Dr-0000 DS=TD7? ES=1D72 SS=1D72 CS=1D72  
IP=0109 'OV/UP EI PL NZ NA FO CY 1D72:01097 D06 INL 0118

What will the IP value be after a “t” command is executed in DOS Debug?

010B

Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:

DS= SEG Hello\_msg, DX=OFFSET Hello\_msg terminated with 24h

How many address lines would be required to address 64 MB directly?

26

NV UP EI NG NZ NA PE NC

JMP 0118

What will the IP value after a “t” command is executed in DOS Debug?

0118h

In the Hello MASM lab in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

020E

What is the hexadecimal encoding for adding BX CX and storing the result in BX?

01D9

1010 0110 in 2's complement equals \_\_\_\_ in base 10.

-90

An input buffer is at memory location 0114, what is the size of the buffer in the decimal?  
84

A “NOP” instruction in program will:

Perform a No Operation

If CX is 0002 what will CX be after a “LOOP” instruction?  
0001

In x86 architecture, ALU stands for which of the following?  
Arithmetic Logic Unit

The instruction MOV CX, [DADD] is what addressing mode?  
Direct

In adding 5+7 through a 4 bit integer unit. The stat of the OF and CF flags afer the add instruction would be:

OF=0, CF=0

What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?

7DF2

In x86 architecture, BIU stands for which of the following?  
Bus interface Unit

F6 in 2's compliment equals\_\_\_\_ in base 10  
-10

A Microprocessor with a 32-bit address bus could access how much memory?

4GB

AND'ing 10H and 2FH will result in which of the following?  
0

For the instruction sequence below, determine the contents of the register AL after this program 51H

The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

32, 32, 13, 10, 36

In MASM, with a “MOV CX, 18” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

In MASM, with a “MOV CX, 18” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

18

What command in DEBUG would be used to execute interrupts?

P

The number of nibbles in a double word are:

8

Determine the contents of register AH after the following instructions have been executed: MOV BX, BA70h // MOV AX, 47E7h // MOV BL, 0Fh // AND AL, BL // MOV [0202], AX  
47h

In using INT 10h to move the screen cursor to return to the beginning of the line what value must be in the AX register?

OE0Dh

Given: 1376:0110 48 61 76 65 20 61 20 67-72 65 61 74 20 53 70 72  
1376:0120 69 6E 67 20 52 65 63 65-73 73 21 00 00 00 00 00 An  
ASCII message begins at memory location 0110, what is the message?

Have a great Spring Recess!

Here is a short sequence of code: 7404CD15EB213C04EBF0A3C67513. All of the instructions are two bytes long. The third instruction operator is:

JMP

You add 1+8 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:

OF = 0, CF = 0

Which BS:DX pair below is wrong when trying to access 2BAD2h in physical memory:

2900:DAD2

Given: AX=CCD0 BX=3534 CX=0000 DX=018 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72  
SS=1D72 CS=1D72 IP=010C NV UP EI NG NZ NA PO CY 1D72:010C 7B06 JGE 0114

What will the IP value be after a “t” command is executed in DOS Debug?

010E

The number of nibbles in a double word are:

8

1011 1100 in 2's complement equals \_\_\_\_ in base 10.

-68

Which DS:AX pair below is WRONG when trying to access 335Dah in physical memory:

3220:04DA

What is the hexadecimal encoding for “JNZ 15” where 15h is the relative offset?

21

Here is a short sequence of code: B400CD163C4A74043C6A7513. All of the instruction are two bytes long. The sixth instruction operator is:

JNZ

You add 9+9 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:

(First digit is the OF, second is the CF)

11

Moore's law has accurately predicted the growth rate in the number of transistors per die for the last 25 years.

What is that rate?

Doubling every 18-24 months

Which instruction below loads register BX with the word beginning at physical address 30608h. Assume DS = 3050,  
BX=1000

MOV BX, [0308]

Determine the contents of register AL after the following instructions have executed: MOV BL, 0Fh // MOV AL, 6Ch //  
XOR AL, BL  
62h

A microprocessor with a 24-bit address bus could access how much memory?

16 MB

The binary number, 1000 0110, represents what values as an Unsigned Binary, 8-bit signed binary, Odd Parity ASCII, and BCD number (in that order)?

134,-122,ACK,86

The instruction MOV CL, SI is what addressing mode?

Register

A "PUSH" instruction:

decrements the SP

If CX is 0000, what will CX be after a "LOOP" instruction?

FFFF

You are trying to rebuild a HELLO program project in MASM and you get the following error: "ERROR 4 line 1". What is the cause of the error?

Not known--this error by itself isn't a problem, press the enter key to clear the error

In MASM, with a "MOV CX, 18" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?

18

You are trying to rebuild a HELLO project program in MASM and you get the following error: "LINK : warning L4021: no stack segment".

What would be the reason for such an error?

No project template for COM was selected

In the Hello MASM lab in the original code, what is the address of the byte used to start the string in the sequence "Hello World 0"?

0200

What is 14.4375 in binary?

001110.01110

How many bit(s) is/are required to represent a range of numbers from 0 to 63?

6

What type of program is this? AX=0000 BX-

0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1376 ES=1376 SS=1376 CS=1376 IP=0100 NV UP EI PL NZ NA PO NC

1376:0100 0100 ADD [BX+SI],AL BP=0000 SI=0000 DI=0000 NV UP EI PL NZ NA PO NC DS:0000=CD

COM

What command in DEBUG would be used to change the code segment?

RCS

This section of memory represents a stack. What type of program is this? BEEF:FFD0

00 00 00 00 00 00 00-00 00 00 00 00 00 00 BEEF:FFE0 00 01 02 03 04 05 06

07-08 09 0A 0B 0C 0D 0E 0FBEEF:FFF0 11 22 33 44 55 66 77

88-99 AA BB CC DD EE FF

EXE program

Determine the contents of register BL after the following instructions have been executed: MOV BL, E2H // MOV CL, 1000b //ROL BL, CL  
E2H

What Hex values must be sent to address the key pad rows on the PPE board?

1,2,4,8

The ASCII codes for space, space, carriage return, line feed, end of string in hexadecimal are:

20,20,0D,0A24

Which of the following is a valid x86 command for multiplying a number?

MUL BX

What command in MASM-CodeView would be used to step through a program line by line?

T (F8)

17. Given the short code, what is the value in AX after the program is run?

0500

A "POP" Instruction

Increments the SP

A "NOP" instruction in a program will:

Perform a No Operation

What is the numeric sequence of the key pad columns on the PPE board used in the lab?

2,4,6,8

For the instruction sequence below, determine the contents of the register AL after this program is executed: MOV AL,  
93h // ADD AL, 69h // DAA

62H

Which of the following is not a valid command for a number into a register in MASM?

MOV AX, F8ADH

On the PPE board, what number(s) on the key pad is (are) pressed for an output port value of 04h and an input port value of 2Fh?

8

AND'ing 20H and 1FH will result in which of the following?

0

With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

BDCA

If the SP is F00F, what will the SP value be after a "PUSH CX" instruction?

F00D

Int 10h uses what function to write a character to the screen and advance the cursor by one character position?

0Eh

Int 21h Function 09h requires three things set up before calling in order to correctly print a string, msg1. They are:

DS=SEG msg1,DX=OFFSET msg1, msg1 terminated with a 24h

What are the contents of AX after this program: MOV AX, 0010h // MOV BX, 8011h // SUB AX, [BX] Memory  
Locations: 8013: F2 8012: 00 8011: 21

FFEFh

The instruction JGE compares which of the following?

The sign flag and the overflow flag to see if they are equal

What will the following program do?

Takes in the keyboard input, "This is the first Midterm"

Here is a short sequence of code: B400CD163C4A74043C6A7513. All of the instructions are two bytes long. The sixth instruction operator is:

JNZ

How many nibbles are in this short sequence of code: B400CD163C4A74043C6A7513

24

What is the hexadecimal encoding for JGE for a conditional jump back 8 bytes?

7DF8

The binary number, 1011 0101, represents what values as an unsigned binary, 8-bit signed binary, odd parity ASCII, and BCD number (in that order)?

181,-76,5,invalid5

1000 0101 in 2's complement equals \_\_\_ in base 10.

-123

How many bytes will the program jump for “JNL 30” where 30h is the relative offset

48

You add 8+8 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:

OF=1, CF=1

In x86 architecture, ALU stands for which of the following?

Arithmetic Logic Unit

Determine the contents of register BL after the following instructions have executed: MOV DH, 7Eh // MOV AL, E7h  
//MOV BL, F0h // AND AL, BL // MOV [0202], AL  
F0h

What are the contents of CX after this program:

0000h

Which instruction below loads register DX with the word beginning at physical address A0802h? Assume DS=A010,  
DX=FF80  
MOV DX, [0702]

Looking at the following program, how many bytes are in the buffer?

19

In the following program, how many bytes will the buffer hold?

40

In using Int 10h to set the video mode to 80 x 25, what value must be in the A register?

00h

1100 1010 in 2's complement equals \_\_\_\_ in base 10.

-36

Which DS:DX pair below is wrong when trying to access 2BADDh in physical memory:

2000:DADD

How many bytes will the program jump for “JNS 20” where 20h is the relative offset?

32

You add 9+8 through a 4 bit integer unit. The state of the OF and CF flags after the add will be: (First digit is the OF, second is the CF)

11

Which instruction below loads register BX, with the word beginning at physical address 90802h? Assume DS=9020, BX=2800

MOV BX, [0602]

Determine the contents of register AL after the following instructions have executed:

E0h

In x86 architecture, BJU stands for which of the following?

Bus Interface Unit

A microprocessor with a 28-bit address bus could access how much memory?

256 MB

The number of nibbles in a byte are:

2

A microprocessor with a 26-bit address bus could access how much memory?

64 MB

1100 1011 in 2's complement equals \_\_\_\_ in base 10.

-50

Which DS:DX pair below is wrong when trying to access 2BADAh in physical memory:

2000:DADA

How many bytes will the program jump for “JNS 12” where 21h is the relative offset?

18

The instruction MOV BX, [SI] is what addressing mode?

Register Indirect

What will the following program do?

Displays “This is the first Midterm” and returns to DOS.

What is the advantage of Assembly Language over C Language?

- a) The Assembler creates much faster executable code

How many address lines would be required to address 128 MB directly?

- a) 27

In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

- a) 020E

Given:

0B0E:0200        57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D

0B0E:0210        62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

As ASCII message begins at memory location 0200, what is the message?

- a) Welcome to Assembly Language

Which of the following DOS Debug instructions would be used to change the IP register to 0110?

- a) RIP

Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is that rate?

- a) Doubling every 18-24 months

Which of the following DOS Debug instructions would set a break point at memory location 010C?

- a) G = 100 10C

AND’ing 1FH and 02H will result in which of the following?

- a) 02

If CX is 0000 what will CX after a “LOOP” instruction?

- a) FFFF

The number of bits in a word are:

- a) 16

In x86 architecture, ALU stands for which of the following?

a) Arithmetic Logic Unit

Given:

|           |                       |                                 |
|-----------|-----------------------|---------------------------------|
| AX=0353   | BX=0534CX=0000DX=0180 | SP=FFEE BP=0000 SI=0000 DI=0000 |
| DS=1D72   | ES=1D72SS=1D72CS=1D72 | IP=0109 OV UP EI PL NZ NA PO CY |
| 1D72:0109 | 7D06 JGE 0118         |                                 |

a) 010B

What is the number 1010.01012 in decimal?

a) 10.31

How many cores does the propeller microcontroller have?

a) 8

What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?

a) 7DF2

What command in DEBUG would be used to step through a program line by line?

a) T

Here is a short sequence of code: 7413\_EBA3\_CD16\_7D21\_3C04\_EBF0\_EB15 . All of the instructions are a word long. The third instruction operator is:

a) Int

In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

a) 24

Given:

|           |                       |                                 |
|-----------|-----------------------|---------------------------------|
| AX=FFE0   | BX=3534CX=0000DX=0180 | SP=FFEE BP=0000 SI=0000 DI=0000 |
| DS=1D72   | ES=1D72SS=1D72CS=1D72 | IP=010D NV UP EI PL NZ NA PO NC |
| 1D72:010D | 7DF6 JNL 0116         |                                 |

How many bytes in decimal will the processor jump if the conditions for a jump were met?

a) -10

Determine the contents of register BL after the following instructions have been executed:

Program Listing

MOV BL, E2H

MOV CL, 08H

ROL BL, CL

a) E2H

The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

a) 32, 32, 13, 10, 36

AND’ing 10H and 2FH will result in which of the following?

a) 0

A “NOP” instruction in a program will:

a) Perform a No Operation

Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY 1D72:0109 7D06 JGE 0118

a) -16

How many address lines would be required to address 64 MB directly?

a) 26

What command in DEBUG would be used to execute interrupts?

a) P

What high level language is the propeller programmed in?

a) Spin

Which of the following DOS Debug instructions would be used to change the IP register to 010C?

a) RIP

The acronym PWM used in the Parallax Propeller and MicroChip PIC18, is defined as:

a) Pulse Width Modulation

Which command would you use to execute another core in the propeller microcontroller?

a) Cognew

Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY 1D72:0111 EB08 JMP 0119

a) 0119h

The instruction MOV CX, DADD is what addressing mode?

a) Immediate

In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

a) A0

Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010F NV UP EI NG NZ NA PO NC 1D72:010F 7D18 JGE 0128

How many bytes in decimal will the processor jump if the conditions for a jump are met?

- a) 24

In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

- a) Sets the Propeller pins P4 through P9 as output pins

What are the contents of DX after this program has been run:

MOV DX, 11h

MOV CX, [5512]

MOV BX, 5511h

SUB DX, [BX]

AND BX, FFFF

Memory Location    Contents

5514    24  
5513    D8

5512    00

5511    21

5510    00

- a) FFF0h

The number of nibbles in a word are:

- a) 4

In the Propellor microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?

- a) Create 10 second delay

The “LOOPNE” instruction is equivalent to which of the following instructions?

- a) DEC CX, JNE

Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI NG NZ NA PO NC 1D72:010D EB07 JMP 0114

- a) 7

On the Arduino platform, what is the programming language used?

- a) C

What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?

- a) A12008

Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?

- a) 1

The acronym ADC in microcontrollers stands for which of the following?

- a) Analog to Digital Converter

In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add

- a) OF = 1, CF = 0

With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

- a) BDCA

A “pull up” resistor is used in digital circuits to do what?

- a) To keep the signal “tied” high the line is active (goes low)

Which of the following is not a valid command for a number into a register in MASM?

- a) MOV AX, AADDH

Given the short code, what is the value in AX after the program is run?

Program Listing

MOV BX, 0005

PUSH BX

MOV AX, 0100

POP AX

- a) 0005

A “pull down” resistor is used in digital circuits to do what?

- a) To keep the signal line “tied” low until the line is active (goes high)

The Ladder Logic diagram would represent which of the following?

- a) OR

What flag(s) does the “LOOPNE” instruction look at to determine whether to loop or not?

- a) ZF

You are trying to rebuild a HELLO project program in MASM and you get the following error:

“LINK : fatal error L1089: HELLO.lrf : cannot open responsefile”.

What would be the reason for such an error?

- a) No source file is identified (no .asm file)

A “POP” instruction:

- a) Increments the SP

Ladder Logic is used in \_\_\_\_\_

- a) PLCs
- a) 37, 2F, 1F

If CX is 0001, what will CX be after a “LOOPNZ” instruction?

- a) 0000

What command in MASM-CodeView (debugging mode) would be used to step through a program line by line?

- a) T (F8)

The Ladder Logic diagram would represent which of the following?

- a) AND

If the SP is F00F, what is the SP value after a “POP CX” instruction?

- a) F011

In the Propeller microcontroller, the term “Method” is (are) which of the following?

- a) A block of executable commands that has variables, can receive parameters, and returns a value.

For the instruction sequence below, determine the contents of the register AL after this program is executed:

#### Program Listing

MOV AL, 83h

ADD AL, 68h

DAA

- a) 51H

This section of memory represents a stack. What type of program is this?

BEEF:FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 BEEF:FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
BEEF:FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

- a) COM program

On the PPE board, what number(s) on the key pad is (are) pressed for an output port value of 08h and an input port value of 2Fh?

- a) 0

Given

13A7:0110 CD 20 30 20 54 68 69 73-30 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0118, how many bytes are in the buffer (in decimal)?

- a) 32

In the Propeller, how many values does a method return?

- a) 1

Which of the following instruction would be used to set the LED to light on the Arduino platform?

- a) digitalWrite(ledPin, HIGH);

Which of the following will cause a program with a LOOP instruction to loop 48 times (decimal)?

- a) CX = 30h

What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1476 ES=1576 SS=1676 CS=1376 IP=0115 NV UP EI PL NZ NA PO NC  
1376:0115 0100 ADD [BX+SI], AL DS:0000=CD

- a) EXE

The letters "NC" labeled to relays and PLCs means which of the following?

- a) Normally Closed

What is -130 decimal in 2's complement (8 bits)?

- a) 01111110

What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1376 ES=1376 SS=1376 CS=1376  
IP=0115 NV UP EI PL NZ NA PO NC 1376:0115 0100 ADD [BX+SI], AL DS:0000=CD

- a) COM

The acronym PLC, is defined as which of the following?

- a) 32KB

You are trying to rebuild a HELLO project program in MASM and you get the following error:

"LINK : warning L4021 : no stack segment".

What would be the reason for such an error?

- a) No project template for COM was selected

On the PPE board, what number(s) on the key pad is (are) pressed for an output port value of 04h and an input port value of 2Fh?

- a) 8

The Ladder Logic diagram would represent which of the following?

- a) AND

In the PIC18 with TRISD = 0b0111111, what is the configuration of the Port D?

- a) Bit 7 of Port D is set to output

How many nibbles are in double precision IEEE floating point format numbers?

- a) 16

What is the decimal value of C5 5A 57 00 in IEEE single precision FP format?

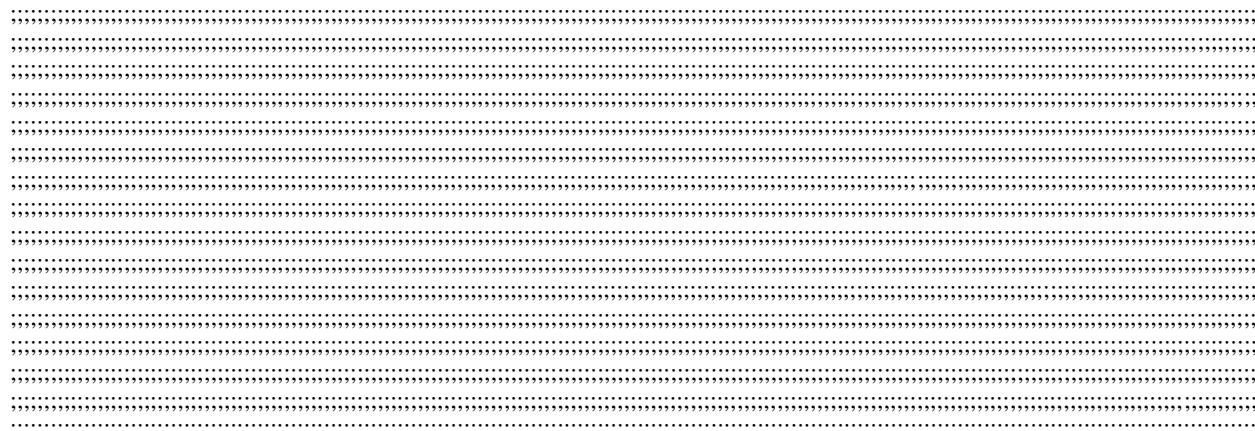
- a) MUL BX

The letters "NO" labeled on relays and PLCs means which of the following?

- a) Normally Open

The instruction MOV CX, [SI] is what addressing mode?

- a) Register Indirect



;Pg47

What are the contents of BX after this program:

| Program #1 Program | Memory Location | Contents |
|--------------------|-----------------|----------|
| MOV BX, 5002h      | 5003            | F2       |
| MOV CX, 3C7Ah      | 5002            | 39       |
| AND CX, [BX]       | 5001            | 4E       |

- a) 5002h

With a PUSHA instruction, what will be the order of the register (registers A – D) contents on the stack?

- a) ACDB

The number of nibbles in a doubleword are:

- a) 8

Determine the contents of register AL after the following instructions have been executed:

### Listing 3

**MOV AL, E2H**

MOV CL, 4H

ROR AL, CL

- a) 2EH

The instruction sequence in listing 2, outputs \_\_\_\_ consecutive bytes of memory.

## Listing 2

STD

MOV CX, 500h

MOV DX, 100h

MOV SL 250h

A1· LODSB

OUT DX AL

LOOP A1

a) 500h

Assuming DS=1000h, the instruction sequence in listing 2 takes the last byte in the transfer from memory at:

a) 0FDB0h

Refer to listing 4. What does this code do?

Listing 4

```
MOV DX, 0F010h  
MOV SI, 1001h
```

```
MOV CX, 0Fh
```

```
CLD
```

```
REP OUTSB
```

a) Outputs bytes from DS:1001 through DS:100F to I/O port 0F010h

1010 0110 in 2's complement equals in base 10 -90

A microprocessor with a 32-bit address bus could access how much memory 4GB

A6 in 2's complement equals in base 10 -90

A "POP" instruction: Increments the SP

A "PUSH" instruction decrements the SP

A "NOP" instruction in a program will Perform a No Operation

AND'ing 1FH and 02H will result in which of the following 02

AND'ing 10H and 2FH will result in which of the following 0

A "pull down" resistor is used in digital circuits to do what? To keep the signal line "tied" low until the line is active (goes high)

A "pull up" resistor is used in digital circuits to do what To keep singal "tied" high until the line is active (goes low)

Determine the contents of register BL after the following instructions have been executed: E2H MOV BL, E2H

MOV CL, 1000b

ROL BL, CL ; rotate BL by 8 to left = same value

For the instruction sequence below, determine the contents of the registers AL after this program 51H

MOV AL, 73h ; 0111 0011

ADD AL, 78h ; 0111 1000

DAA

if low nibble of AL > 9 or AF = 1 then:

AL = AL + 6

AF = 1

if AL > 9Fh or CF = 1 then:

AL = AL + 60h

CF = 1

For the instruction sequence below, determine the contents of the register AL after this program is executed 51H

MOV AL, 83h

ADD, AL, 68h  
DAA

Given the short code, what is the value in AX after the program is run 0100

Mov BX, 0100  
PUSH BX  
MOV AX, 0500  
POP AX

Given the short code, what is the value in AX after the program is run 0001

MOV BX, 0001  
PUSH BX  
MOV AX, 0500  
POP AX

GIVEN: IP = 0106 Flags: NV UP EI NG NZ NA PE NC Instruction: JMP 011F

What will the IP value be after “t” command is executed in DOS Debug? 011FH (Unconditional Jump)

GIVEN: IP = 0109 Flags: OV UP EI PL NZ NA PO CY Instruction: JGE 0118 What will the IP value be after a “t” command is executed in DOS Debug 010B (0109 + 0010 add two bytes)

GIVEN: IP=FFE0 Flags: OV UP EI PL NZ NA PO NC Instruction: JGE 0116: ID72:010D 7D09 How many bytes will the processor jump if the condition for a jump were met? 9

GIVEN: AX= FFF0 IP = 0109 FALGS: OV UP EI PL NZ NA PO CY ID72: 010F 7D18 Instruction: JGE 0118. What is the signed decimal value of the number in the AX register? -16  
Convert the number 7D18 into decimal.

GIVEN: IP= 010F Flags: NV UP EI NG NZ NA PO NC Instruction: JNL 0115.

How many bytes in decimal will the processor jump if the conditions for a jump were met 24

GIVEN: 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message? Welcome to Assembly Language

Here is a short sequence of code: 7413 EBA3 CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The third instruction operator is INT

Here is a short sequence of code: 7413 A3EB CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The fourth instruction operator is JGE

Here is a short sequence of code: B400 CD16 3C4A 7404 BC6A 7513. All of the instructions are two bytes long. The sixth instruction operator is JNZ

How many cores does the propeller microcontroller have 8

How many bits(s) is/are required to represent a range of decimal numbers from 0 to 15 4

How many bits(s) is/are required to represent a range of decimal numbers from 0 to 63 6

How many bits(s) is/are required to represent a range of decimal numbers from 0 to 127

How many bits(s) is/are required to represent a range of decimal numbers from 0 to 255 8

How many bytes are there in this short sequence of code B4 00 CD 16 4C CD 20 7

How many nibbles are there in this short sequence of code B4 00 CD 16 3C 4A 74 04 3C 6A 75 13 2  
How many bytes are in double precision IEEE floating point format numbers 8

How many nibbles are in double precision IEEE floating point format numbers 16

How many address lines would be required to address 128 MB directly 27 ( $128 \times 1048576 = 134217728$  and  $2^{27} = 134217728$ )

How many address lines would be required to address 64 MB directly 26 ( $64 \times 1048576 = 67108864$  and  $2^{26} = 67108864$ )

If CX is 0000, what will CX be after a “LOOP” instruction FFFF If CX is 0003, what will CX be after a “LOOPNZ” instruction 0002

If the SP is F00F, what is the SP value after a “PUSH CX” instruction F00D If the SP is F00F, what is the SP value after a “POP CX” instruction F011 If the SP is F00F, what is the SP value after a “POP SP” instruction F011

In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be OF = 1, CF = 0

In x86 architecture, BIU stands for which of the following Bus Interface Unit

In x86 architecture, ALU stands for which of the following Arithmetic Logic Unit

In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”? 020E

In MASM, with a “MOV CX, 24h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal 36

In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, how many times will the program loop in decimal 24

In MASM, with a “MOV CX, 12h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal 18

In the Hello MASM lab in the original code, what is the address of the string to start the message “Hello World 0” 0200

In the PIC18 with TRISD = 0b10000000, what is the configuration of the Port D Bit 7 of port D is set to input

In the PIC18 with TRISD = 0b01111111 and LATD = 0xAA, what value will be on Port D and shown on

the LEDS Bit 7 of port D is set to output (because the first bit is zero = output)

In the PIC18 with TRISD = 0b00001111, what is the configuration of the Port D A0 (First 4 are outputs and last four are inputs)

In the PIC18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDS 0A

In the Propeller microcontroller, the command “dira[9..4] := %000000” would cause the processor to do which of the following Sets the propeller pin P4 through P9 as output pins

In the Propeller microcontroller, the command “dira[9..4] := %111111” would cause the processor to do which of the following Sets the propeller pin P4 through P9 as output pins

In the propeller microcontroller, the command “waitcnt(clkfreq\*3 + cnt)” would cause the processor to do which of the following A 3 second delay

In the propeller microcontroller, the command “waitcnt(clkfreq\*2 + cnt)” would cause the processor to do which of the following A 2 second delay

In the Propeller microcontroller, the term “Method” is (are) which of the following A block of executable commands that has variables, can receive parameters, and returns a value.

Int 10h uses what function code to write a character to the screen and advance the cursor by one character position 0Eh

Int 21h, Function 09h requires three things set up before calling in order to correctly print a string:  
DS=SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h.

Ladder Logic is used in? PLCs

Moore's law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?  
Doubling every 18-24 months

On the Arduino platform what is the program language used C

On the PPE board, what numbers(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh 0

On the PPE board, what numbers(s) on the key pad is(are) pressed for an output port value of 04h and an input port value of 2Fh 8

The "LOOPNZ" instruction is equivalent to which of the following instructions DEC CX, JNE

The acronym PWM used for motor control, is defined as which of the following Pulse Width Modulation

The acronym PLC, is defined as which of the following? Programmable Logic Controller

The ASCII codes for space, space, carriage return, line feed, end of string in decimal are 32, 32, 13,10, 36

The ASCII codes for space, space, carriage return, line feed, end of string in hexadecimal are: 20,20,0D,0A,24

The binary number, 1011 0101, represents what values as a unsigned binary, 8 bit signed binary, odd parity ASCII, and BCD number (in that order) 181, -76, 5, invalaid5

The instruction MOV CX, DADD is what addressing mode Immediate

The instruction MOV CX, [DADD] is what addressing mode Direct

The number of bits in single precision IEEE floating pint format are 32

This section of memory represents a stack. What type of program is this EXE PROGRAM BEEF:00D0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00

BEEF:00E0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F BEEF:00F0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

BEEF:0FD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 BEEF:0FE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
BEEF:0FF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

This section of memory represents a stack. What type of program is this COM PROGRAM BEEF:FFD0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00

BEEF:FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F BEEF:FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal MOV CX, 48

Which command would you use to execute another core in the propeller microcontroller Cognew

What command in DEBUG would be used to change the code segment RCS

What command in DEBUG would be used to change the IP value RIP

What command in DEBUG would be used to execute interrupts P

What command in MASM-CodeView would be used to step through a program line by line T(F8)

What flag(s) does the "LOOPNZ" instruction look at to determine whether to loop or not ZF What flag(s) does the "LOOPNE" instruction look at to determine whether to loop or not ZF

What Hex values must be sent to address the key pad rows on the PPE board 1,2,4,8

What is the advantage of C Language over Assembly Language C is transportable to other microprocessor architectures

What is the hexadecimal encoding for “JGE” for a jump back 10 bytes 7DF4 (10 bytes + 2 )

What is the hexadecimal encoding for adding AX with BX and storing the result in AX 01D8 ADD AX, BX 000 00W 11 reg1 reg2

What is the hexadecimal encoding for adding BX with CX and storing the result in BX 01CB

What is the hexadecimal encoding for adding BX with DX and storing the result in BX 01D3

What is -130 decimal in 2's complement (8bits) 01111110

What is -32.75 in a base two number system -100000.110000

What is 14.4375 in binary 001110.0110

What is 16.4375 in binary 010000.01110

What is the binary value of decimal 12.875 1100.1110

What is 16.4375 in binary 010000.01110

What is number, 1011.0101 (2) in decimal? 11.31

What is the numeric sequence of the key pad columns on the PPE board 37,2F,1F

What is the decimal value of C5 5A 57 00 in IEEE single precision FP format -3493.4375

What of the following instruction would be used to set the LED to light on the Arudino platform digitalWrite(ledPin, HIGH);

What type of program is this EXE

IP = 0115, 1376:0115 0100

ADD [BX+SI], AL

DS:0000=CD

What type of program is this COM

IP = 0100, 1376:0100 0100

ADD [BX+SI], AL

DS:0000=CD

Which of the following DOS Debug instructions would set a break point at memory location 010C G = 100 10C

Which of the following would be used to set the TRISA register to control the direction of PIC18 port to input 1 and for output its 0

Which of the following DOS Debug instructions would be used to change the IP register to 110 RAX = 0110

Which of the following will cause a program with a LOOP instruction to loop 48 times (decimal) CX=30h

Which of the following is a valid x86 command for multiplying a number MUL BX

Which of the following is not a valid command for a number into a register in MASM MOV AX, BADH

With a POP BX instruction, what will be order off the accumulator, base, count, and data registers restored from the stack BX

With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack BDCA

You are trying to rebuild a HELLO program project in MASM and you get the following error: "ERROR 4 line 1". What is the cause of the error? Not known—this error by itself isn't a problem, press the enter key to clear the error.

You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK :

warring L4021: no stack segment". What would be the reason for the such an error No project template for COM was selected.

You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK : fatal error L1089: HELLO.lrf: cannot open response file". What would be the reason for the such an error No source file is identified(no .asm file)

PPE Row Column Scan decoding

D7 D6 D5 D4 D3 D2 D1 D0

S7 S6 S5 S4 S3 S2

^ ^ ^

3 2 1

0 0 0 0 1 0 0 0 = 08h

0 0 0 1 0 0 0 0 = 10h

0 0 1 0 0 0 0 0 = 20h

0 0 1 1 1 1 1 1 = 3Fh -> Nothing pressed

0 0 1 1 0 1 1 1 = 37h -> Number 1 pressed

0 0 1 0 1 1 1 1 = 2Fh -> Number 2 pressed

0 0 0 1 1 1 1 1 = 1Fh -> Number 3 pressed

EEE 174 Midterm Study Guide

S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:

- DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36

2. Moore's Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?

- Doubling every 18 – 24 months

3. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What will the IP value be after a “t” command is executed in DOS Debug?

- 010B

4. A “NOP” instruction will:

- Perform a No OPeration

5. Given:

AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0106 NV UP EI NG NZ NA PE NC

1D72:0106 EB0F JMP 0118

What will the IP value be after a “t” command is executed in DOS Debug?

- 0118h

6. In x86 architecture, BIU stands for which of the following?

- Bus Interface Unit

7. Determine the contents of register BL after the following instructions have been executed:

- 2EH

8. The number of bytes in a double word are:

- 4

9. In x86 architecture, ALU stands for which of the following?

- Arithmetic Logic Unit

10. The “LOOPNE” instruction is equivalent to which of the following instructions?

- DEC CX, JNE

11. The instruction MOV CX, DADD is what addressing mode?

- Immediate

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

- INT

13. Which of the following DOS Debug instructions would set a break point at memory location 010C

Program Listing

MOV BL, E2H  
MOV CL, 100b

- G = 100 10C
- 14. How many bytes are there in this short sequence of code? B400CD164CCD21
- 7
- 15. How many address lines would be required to address 128 MB directly?
- 27
- 16. What are the contents of CX after this program has been run:
- D800h
- 17. 1010 0110 in 2's complement equals \_\_\_\_ in base 10.
- -90

Memory Location Contents

1103 24

1102 D8

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

- 01D8
- 19. What is 11.437510 in binary?
- 001011.01110
- 20. If CX is 0000 what will CX be after a “LOOP” instruction?
- FFFF

21. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?

- 7DFA
- 22. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010E OV UP EI PL NZ NA PO NC

1D72:010F 7D08 JGE 0115

How many bytes will the processor jump if the conditions for a jump were met?

- 8

23. What command in DEBUG would be used to execute interrupts?
- P
24. What is the advantage of C Language over Assembly Language?
- C is transportable to other microprocessor architectures
25. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:
- OF = 1, CF = 0
26. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
- 36
27. Given:

AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI PL NZ NA PO NC

1D72:010D 7DF6 JNL 0116

- -10
28. AND’ing 1FH and 20H will result in which of the following?
- 0
29. Given:

13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

- 72
30. A microprocessor with a 31-bit address bus could access how much memory?
- 2 GB
31. For the instruction sequence below, determine the contents of the register AL after this program is executed:
- 51H

32. Given:

AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

- -256

33. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

- 32, 32, 13, 10, 36

Spring 2012

34. What is the advantage of assembly language over C language?

- The assembler creates much faster executable code

35. How many lines would be required to address 128 mb directly?

- 27

36. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

- 020E

37. Given

W e l c o m e \_ t o \_ A s s e m b l e r 0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D 0B0E:0210 62 6C 79 20  
4C 61 6E 67-75 61 67 65 00 00 00 00 b l y \_ L a n g u a g e

An Ascii message begins at memory location 0200, what is the message?

- Welcome to Assembly Language

#### Program Listing

MOV AL, 73h

ADD AL, 78h

DAA

38. Which of the following DOS Debug instructions would be used to change the IP register to 0110?

- RIP

39. Moore's law has accurately predicted the growth rate in the number of transistors per ide for the last 40 years. What is the rate?
- Doubling every 18 – 24 months
40. Which of the following DOS debug instructions would set a break point at memory location 010C?
- G = 1000 10C
41. AND'ing 1FH and 02H will result in which of the following?
- 02
42. If CX is 0000 what will CX be after a "LOOP" instruction?
- FFFF
43. The number of bits in a word are:
- 16
44. In x86 architecture, ALU stands for which of the following?
- Arithmetic Logical Unit
45. Given:

What will the IP value be after a "t" command is executed in DOS DEBUG?

- 010B
46. What is the number 1010.01012 in decimal?
- 10.31
47. How many cores does the propeller microcontroller have?
- 8
48. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?
- 7DF2
49. What command in debug would be used to step through a program line by line?
- T
50. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:
- INT
51. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?
- 24
52. Given:

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- -10
53. Determine the contents of the register BL after the following instructions have been executed:
- E2H
54. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- 20, 20, 0D, 0A, 24
55. AND'ing 10H and 2FH will result in which of the following?
- 0
56. A “NOP” instruction in a program will
- Perform a No Operation
57. Given:

What is the signed decimal value of the number in the AX register?

- -16
58. How many address lines would be required to address 64 MB directly?
- 26
59. What command in debug would be used to execute interrupts?
- P
60. What high level language is the propeller programmed in?
- Spin
61. Which of the following DOS debug instructions would be used to change the IP register to 010C?
- RIP
62. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:
- Pulse width modulation
63. Which command would you use to execute another core in the propeller microcontroller?
- Cognew
64. Given:

What will the IP value be after the “t” command is executed in DOS debug?

- 0119h
65. The instruction MOV CD, DADD is what addressing mode?
- Immediate
66. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?
- F0
67. How many byte in decimal will the processor jump if the conditions for a jump are met?
- 24
68. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?
- Sets the Propeller pins P4 through P9 as output pins
69. What are the contents of DX after this program has been run:
- FFF0h.
70. The number of nibbles in a word are:
- 4
71. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?
- Create 10 second delay
72. The “LOOPNE” instruction is equivalent to which of the following instructions?
- DEC CX JNE
73. Given

How many bytes will the processor jump if the conditions for a jump were met?

- 7
74. On the Arduino platform, what is the programming language used?
- C
75. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?
- A12008
76. Which of the following would be used to set the TRISA register to control the direction of PIC18 Port to input?

- 1
77. The acronym ADC in microcontrollers stands for which of the following?
- Analog to Digital converter
78. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:
- OF = 1, CF = 0
79. What is the advantage of Assembly Language over C Language?
- The Assembler creates much faster executable code
80. How many address lines would be required to address 128 MB directly?
- 27

81. In the x86 lab part Hello MASM program in the original code, what is the address of the byte used to start number in the sequence “Hello World 0”?

- 020E
82. Given:

0B0E : 0200 57 65 6C 63 F 6D 65 20-74 6F 20 41 73 73 65 6D

0B0E: 021062 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message?

- Welcome to Assembly Language
83. Which of the following DOS Debug instructions would be used to change the IP register to 0110?
- RIP
84. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?
- Doubling every 18-24 months
85. Which of the following DOS Debug instructions would set a break point at memory location 010C?
- G = 100 10C
86. AND’ing 1FH and 02H will result in which of the following?
- 02
87. If CX is 0000 what will CX be after a “LOOP” instruction?
- FFFF
88. The number of bits in a word are:
- 16
89. In x86 architecture, ALU stands for which of the following?
- Arithmetic Logic Unit

90. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 0V UP EI PL NZ NA PO CY

1D72:0109 7D06 JGE 0118

What will the IP value be after a “t” command is executed in DOS Debug?

- 010B
91. What is the number, 1010.01012 in decimal?

- 10.31
92. How many cores does the propeller microcontroller have?
- 8
93. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?
- 7DF2
94. What command in DEBUG would be used to step through a program line by line?
- T

95. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:

- INT
96. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
- 24
97. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI PL NZ NA PO NC

1D72:010D 7DF6 JN1 0116

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- -10
98. Determine the contents of register BL after the following instructions have been executed:
- E2H
99. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- 32, 32, 13, 10, 36
100. AND’ing 10H and 2FH will result in which of the following?
- 0
101. A “NOP” instruction in a program will:
- Perform a No Operation

102. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72

ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY ID72:0109 7D06 JGE 0118

What is the signed decimal value of the number in the AX register?

- -16
103. How many address lines would be required to address 64 MB directly?
- 26
104. What command in DEBUG would be used to execute interrupts?
- P

105. What high level language is the propeller programmed in?  
 • Spin
106. Which of the following DOS Debug Instructions would be used to change the IP register to 010C?  
 • RIP
107. The acronym PWM used in the Parallax Propeller and MicroChip PIC18, is defined as:  
 • Pulse Width Modulation
108. Which command would you use to execute another core in the propeller microcontroller?  
 • Cognew
109. Given:

AX=FFDO BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
 IP=0111 NV UP EI NG NZ NA PO CY

1D72:0111 ED08 JMP 0119

What will the IP value be after a “t” command is executed in DOS Debug?

- 0119h
2. The instruction MOV CX, DADD is what addressing mode?  
 • Immediate
110. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?  
 • A0
111. Given:
- AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
 IP=010F NV UP EI NG NZ NA PO NC
- ID72:010F 7D18 JGE 0128
- How many bytes in decimal will the processor jump if the conditions for a jump are met?  
 • 24
112. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?  
 • Sets the Propeller pins P4 through P9 as output pins
113. What are the contexts of DX after this program has been run:

- FFFF0h

114. The number of nibbles in a word are:
- 4
115. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?
- Create 10 second delay
116. The “LOOPNE” instruction is equivalent to which of the following instructions?
- DEC CX, JNE
117. Given:
- AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI NG NZ NA PO NC
- 1D72:010D EB07 JMP 0114
- How many bytes will the processor jump if the conditions for a jump were met?
- 7
118. On the Arduino platform, what is the programming language used?
- C
119. What is the hexadecimal encoding for loading AX with a word (value) from the memory location 0820h?
- A12008
120. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?
- 1
121. The acronym ADC in microcontrollers stands for which of the following?
- Analog to Digital Converter
122. In adding 5+5 through a 4 bit integer unit. The state of the OF and the CF flags after the add instruction would be:
- OF=1, CF=0
123. Given:  
0B0E:020057 65 6C 63 6F 6D 65 20-74 6F 20 74 68 65 20 66
- 0B0E:0210 69 72 73 74 20 64 61 79-20 6F 66 20 74 68 65 20
- 0B0E:0220 72 65 73 74 20 6F 66 20-79 6F 75 72 20 6C 69 66
- 0B0E:0230 65 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
- An ASCII message begins at memory location 0200, what is the message?
- Welcome to the first day of the rest of your life
124. The instruction MOV DX, BADD is what addressing mode?
- Immediate
125. Which of the following is the hexadecimal encoding for adding BX with CX and storing the result in CX?

- 03CB
126. What is the advantage of Assembly Language over C Language?
- The Assembler creates much faster executable code
127. What is 18.437510 in binary?
- 010010.01110
128. For the instruction sequence below, determine the contents of the register AL after this program is executed:
- 51H
129. In x86 architecture, ALU stands for which of the following?
- Arithmetic Logic Unit
130. A microprocessor with a 33-bit address bus could access how much memory?
- 8 GB
131. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?
- 7DF2
132. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010F NV UP EI NG NZ NA PO NC

1D72:010F      7D18      JGE      0128

How many bytes in decimal will the processor jump if the conditions for a jump are met?

- 24
133. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is that rate?
- Doubling every 18-24 months
134. The number of bytes in a word are:
- 2
135. Determine the contents of register BL after the following instructions have been executed:
- 2EH
136. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?
- 7
137. What high level language is the propeller programmed in?
- Spin
138. In the Propeller microcontroller, the command “dira[4..9] := %000000” would cause the processor to do which of the following?
- Sets the Propeller pins P4 through P9 as input pins
139. Which command would you use to execute another core in the propeller microcontroller?
- Cognew
140. How many cores does the propeller microcontroller have?

- 8
141. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?
- Create 10 second delay
142. The acronym ADC in microcontrollers stands for which of the following?
- Analog to Digital Converter
143. The acronym PWM used in the Parallax Propeller and Microchip PIC18, is defined as:
- Pulse Width Modulation
144. How many bits does the PIC18 microcontroller use in the PICkit 3 Debug Express have?
- 8
145. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?
- 1
146. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?
- Bit 7 of port D is set to output
147. In the PIC 18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?
- 0A
148. On the Arduino platform, what is the programming language used?
- C
149. Given:  
AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
1D72:0111 EB08 JMP 0119
- What will the IP value be after a “t” command is executed in DOS Debug?
- 0119h
150. How many bytes are there in this short sequence of code? B4 00 CD 16 4C CD 21 CD 20
- 9
151. In x86 architecture, BIU stands for which of the following?
- Bus Interface Unit
152. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The fifth instruction operator is:
- CMP
153. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- 32, 32, 13, 10, 36
154. A “NOP” instruction in a program will:
- Perform a No Operation
155. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

- -16
- 156. Which of the following DOS Debug instructions would set a break point at memory location 010C?  
• G = 100 10C

157. In adding 5+7 through a 4 bit integer unit, the state of the OF and CF flags after the add instruction would be:

- OF = 1, CF = 0

158. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV UP EI NG ZR NA PO NC

1D72:010D 7509 JNZ 0116

What will the IP value be after a “t” command is executed in DOS Debug?

- 010Fh
- 159. What are the contents of DX after this program has been run:  
• FFF0h
- 160. Which of the following DOS Debug instructions would be used to change the IP register to 010C?  
• RIP
- 161. What is the number, 1010.01012 in decimal?  
• 10.31
- 162. What command in DEBUG would be used to step through a program line by line?  
• T
- 163. AND’ing 1FH and 02H will result in which of the following?  
• 02
- 164. How many address lines would be required to address 64 MB directly?  
• 26
- 165.

S2010 MT2

- 166. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?  
• 7
- 167. If CX is 0003, what will CX be after a “LOOP” instruction?  
• 0002
- 168. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?  
• A 5 second delay
- 169. What is the number 1011.01012 in decimal?  
• 11.31

170. This section of memory represents a stack. What type of program is this? BEEF : FFD0 00 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

- COM program

171. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

- BDCA

172. Determine the contents of register BL after the following instruction have been executed:

- 2EH

173. What Hex values must be sent to address the key pad rows on the PPE board?

- 1, 2, 4, 8

174. What is the number 32.437510 in binary?

- 100000.01110

175. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

- 24

176. The acronym PWM used for motor control, is defined as which of the following?

- Pulse Width Modulation

177. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

- Bit 7 of port D is set to output

178. Given the short code, what is the value in AX after the program is run?

- 0500

#### Program Listing

MOV BL, 2E

MOV CL, 10

ROL BL, CL

### Program Listing

MOV BX, 0500

Push BX

MOV AX, 0100

POP AX

179. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?

- ZF

180. How many nibbles are in double precision IEEE floating point format number?

- 16

181. In the Hello MASM lab in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

- 020E

182. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC

1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD

- EXE

183. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?

- F00D

184. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D

and shown on the LEDs?

- A0
185. What is the numeric sequence of the key pad columns on the PPE board?
- 37, 2F, 1F
186. What is -130 decimal in 2's complement (8bits)?
- 0111110
187. Which of the following is a valid x86 command for multiplying a number?
- MUL BX
188. The number of bits in single precision IEEE floating point format are:
- 32
189. You are trying to rebuild a HELLO project program in MASM and you get the following error:
- “LINK : warning L4021: no stack segment”.
- What would be the reason for such an error?
- No project template for COM was selected
190. A “pull down” resistor is used in digital circuits to do what?
- To keep the signal line “tied” low until the line is active (goes high)
191. A “POP” instruction:
- increments the SP
192. AND’ing 10H and 2FH will result in which of the following?
- 0
193. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?
- Sets the Propeller pins P4 through P9 as output pins
194. What commands in MASM-CodeView would be used to step through a program line by line?
- T (F8)
195. If the SP is F00F, what will the SP value be after a “POP SP” instruction?
- F011
196. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?
- 0

197. In the Propeller microcontroller, the term “Method” is(are) which of the following?
- A block of executable commands that has variables, can receive parameters, and returns a value.
198. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?
- MOV CX, 30H      or      MOV CX, 48
199. A “pull up” resistor is used in digital circuits to do what?
- To keep the signal “tied” high until the line is active (goes low)

#### EEE 64 Exams

200. A “pull up” resistor is used in digital circuits to do what?
- To keep the signal “tied” high until the line is active (goes low)
201. How many Flip Flops would be required for a 5 state, State Machine?
- 3
202. 32 decimal would be what value in hexadecimal?
- 20
203. Which gate would be used for the function,  $F = /C/D + CD$ ?
- XNOR
204. In 8 to 1 Multiplexer, show, if  $A = 1, B = 0, C = 0, D = 0$ ; what would

the output be equal too?

- 0
205. How many bits would be required to count from 0 to 511 in binary?
- 9
206. The Ladder Logic diagram would represent which of the following?
- NAND

207. The number of nibbles needed for a 32 bit number are?

- 8

208. The Ladder Logic diagram would represent

which of the following?

- AND

209. XOR'ing 10Hex and 2FHex will result in which of the following numbers?

- 3F

210. The Ladder Logic diagram would represent which of the following?

- NOR

211. What is the difference between a half adder and a full adder?

- The half adder is missing a carry in

212. In the truth table shown; how many Karnaugh maps

would be required to solve the truth table?

- 6

213. In the truth table shown; in mapping values to Karnaugh

map what value would be assigned to the states 10 to 15?

- 0

214. A Moore state machine:

- The output depends on the state; the next state depends on the input and current state.

215. F0 In 2's complement equals \_\_\_\_ in base 10.

- -16

216. In 8 to 1 Multiplexer shown, with A = 0, B = 1, C = 0; what would the

value of output Y be equal to?

- 1

217. A "pull down" resistor is used in digital circuits to do what?

- To keep the signal line "tied" low until the line is active (goes high)

218. ADD'ing 10HEX and 2FHEX will result In which of the following Hex numbers?

- 3F

219. In the truth table shown; how many Karnaugh maps would

be required to solve the truth table?

- 4

220. In the truth table shown; in mapping values into the

Karnaugh map what value would be assigned to the states 10 to 15?

- X or D for don't care

221. A Mealy state machine:

- The output depends on input and the current state; the next state depends on input and current state.

222. For a NAND RS Latch; if both R and S inputs are High, Q output would be in what state?

- Undefined state

223. XNOR'ing 2FHex and 10Hex will result in which of the following Hex numbers?

- 0

224. Given a 4 bit adder with carry out, S4, adding two four bit numbers A and B. If A = 7 and B = 8, what would the values of S4, S3, S2, S1, S0 be?

- 01111

225. Which gate would be used for the function,  $F = /AB + A/B$ ?

- XOR

226. For a NOR RS Latch, if both R and S inputs are High, Q output would be in what state?

- Undefined state

227. In 8 to 1 Multiplexer shown, with A = 0, B = 1, C = 1;

what would the value of output Y be equal to?

- 0

228. What would 6A Hex equal in base 2?
- 0110 1010
229. The number F6Hex in 2's complement equals what number in base 10.
- -10
230. For the Flip Flops in the Circuit diagram notice that set and reset are active low. If S1 = 1, R1 = 0, S2 = 1, R2 = 0, S3 = 0, R3 = 1, S4 = 0, R4 = 1; what is the output of the counter in hexadecimal.
- 3
231. If S and R in their inactive state, how many combination of Qs are available?
- 16
232. For the circuit shown, what is the equation for the next state of Q0,?
- $/Q_1/Q_0 + Q_1Q_0$
233. What will this state machine do?
- Divide by 3
234. If Q1 is 1 and Q0 is 1, what is the next state of Q1 and Q0?
- 01
235. Using the Karnaugh map for minimum gates, would result in which of the following equations:
- AB + CD
236. 36 decimal would be what value in hexadecimal?

- 24
237. NAND'ing 10Hex and 2FHex will result in which of the following Hex numbers?
- FF
238. How many bytes are there in this short sequence of hex numbers? B400CD164CCD20
- 7
239. How many nibbles are there in this short sequence of hex numbers? B400CD164CCD20
- 14
240. Using Karnaugh map for the least gates, would result in

which equation?

- $/B \cdot C/D + /B \cdot /C \cdot D$
241. How many bits would be required to count for 0 to 1024 in binary?
- 10
242. AND'ing 8FHex and 02Hex will result in which of the following Hex numbers?
- 02
243. Which of the following equations were used to map the values into the Karnaugh map?
- $/A \cdot B \cdot C \cdot D + /A \cdot B \cdot C \cdot /D + A \cdot B \cdot C \cdot D + A \cdot B \cdot /C \cdot D + A \cdot /B \cdot C \cdot D + /A \cdot B \cdot /C \cdot D + A \cdot /B \cdot /C \cdot D$
244. How many bits are there in this sequence of hex numbers?
- B400CD16CD20
- 48
245. In 8 to 1 Multiplexer show, if A = 0, B = 1, C = 1; what would the output be equal to?
- Not D ( $/D$ )
246. XOR'ing 1FHex and 09Hex will result in which of the

following Hex numbers?

- 16

247. In the truth table shown; how many Karnaugh maps would be required to solve the

truth table?

- 7

248. In the truth table shown; in mapping values to Karnaugh map what value would be assigned to the states 10 to 15?

- 0

249. For the Karnaugh map shown, what equation was used to maps the ones into the map?

- $/A/C/B/D + /A/BC/D + A/B/C/D + A/BC/D$

250. Which of the following gates would you use to output a one when the inputs have the same value?

- XNOR

251. In 8 to 1 Multiplexer show, if  $A = 0$ ,  $B = 1$ ,  $C = 1$ ; what would the

output be equal to?

- 0

252. How many Flip Flops would be required for a 12 state, state machine?

- 4

253. A “pull down” resistor is used in digital circuits to do what?

- To keep the signal line “tied” low until the line is active (goes high)

254. Using the Karnaugh map for the least number of gates, what

would the final equation be?

- $/D$

255. In 8 to 1 Multiplexer, show, if A = 1, B = 0, C  
= 0; what would the value of output Y be equal to?  
• 1

256. An NAND gate with its inputs inverted would be equivalent to which of the following?  
• OR
257. If S1 = 1, R1 = 0, S2 = 1, R2 = 0, S3 = 0, R3 = 1, S4 = 1, R4 = 0; what is the output of the counter in hexadecimal.  
• 2
258. If S and R are allowed to take on all values, how many combinations of Qs and the Next state of Q, /Qs are available?  
• 16
259. How many Flip Flops would be required for a 9 state, state machine  
• 4
260. If 10Hex is XOR'ing with 2FHex would result in which of the following Decimal numbers?  
• C0
261. For a NOR RS Latch, if both R and S inputs are Low, Q output would be in what state?  
• Inactive state
262. The number of bytes needed for a 32 bit number are:  
• 4

263. How many bits would be required to count from 0 to 1023 in binary?

- 10

264. What is the signed decimal value of the hex number, FFF0

- -16

265. If 10Hex is ADDED to 2FHex would result in which of the following Hex numbers?

- 3F

266. In 8 to 1 Multiplexer, show, if A = 1, B = 0, C = 0, D = 0; what

would the output be equal too?

- 1

267. Given a 4 bit adder with carry out, S4, adding two four bit

numbers A and B. If A = 8 and B = 8, what would the values of S4, S3, S2, S1, S0 be?

- 10000

268. What would 6A Hex equal in base 2?

- 106

269. For the circuit shown, what is the equation for the next state of Q1,?

- Q0/Q1

270. What will this circuit

do?

- Divide by 3

271. If Q1 is 1 and Q0 is 0,

what is the next state of Q1 and Q0?

- 00

272. The number F6Hex in 2's complement equals what number in base 16.

- -A

273. In 8 to 1 Multiplexer show, if A = 1, B = 1, C =

1; what would the output be equal to?

- 1

274. 0F in 2's complement equals (8 bits) \_\_\_\_ in base 10.

- 15

275. For the Flip Flops in the

Circuit diagram notice that set and reset are active low. If S1 = 1, R1 = 0, S2 = 1, R2 = 0, S3 = 0, R3 = 1, S4 = 1, R4 = 0; what is the output of the counter in hexadecimal.

- 2

276. If S and R are in their inactive state, how many possible states

could there be?

- 16

277. Using the Karnaugh map, would result in which of the following equations:

- AB+BC+AC

278. ADD'ing 1FH and 20H will result in which of the following?

- 3F

279. SOP in logic design, would be defined as

- The terms of the equation are expressed in sum of products.

280. The expression (equation) for the circuit

shown would be:

- $/((D+(AB))/(B+C))$

281. Which of the gates would be used for the equation A/C

- AND

282. The equation  $F = ABD + CD + /C/D$  would be equivalent to which of the equations?

- $(B + C + /D)(A + C + /D)(/C + D)$

283. The binary value 1111 0110 in 2's complement equals \_\_\_\_ in base 10.

- -10
284. The number of bits in a Byte are:
- 8
285. Gray code, would be a code where
- Only one bit changes from one number to the next
286. The minterms for  $/A \cdot B \cdot C \cdot D + /A \cdot B \cdot C \cdot D + A \cdot B \cdot C \cdot D$  would be?
- m<sub>2</sub>, m<sub>7</sub>, m<sub>11</sub>, m<sub>12</sub>, m<sub>13</sub>, m<sub>14</sub>, m<sub>15</sub>
287. The equation for F(A,B,C) in the diagram would be
- $/A \cdot C + B \cdot C$
- 
288. Which of the following gates would be used for the equation  $/A \cdot C$
- AND
289. 24 hex would be what value in decimal?
- 36
290. The ASCII codes for space, space, carriage return, line feed, \$ in decimal are:
- 32, 32, 13, 10, 36
291. Using the Karnaugh map (Kmap) for minimum gate reduction, would result in which of the following equations (the “X” are don’t cares):
- $/D$
292. What is 166.437510 in binary
- 10100110.01110
293. The expression for the circuit shown would be:
- $/C \cdot D + A \cdot B \cdot D + C \cdot D$
- 
294. The binary number, 0110 1110, represents what values; in Hex, and as a BCD number?
- 6E, 6 invalid
295. Which of the gates would be used for the equation  $A \cdot C + /A \cdot C$
- XNOR

296. What is the signed decimal value of the hex number, 0F in 8bits?
- 15
297. Minimizing  $/A/B + /AB + AB$  would result in:
- $/A + B$
298. Minimizing the equation  $/A/BCD + /ABCD + ABCD + AB/C/D + A/BCD + AB/CD + ABC/D$  would result in:
- $AB + CD$
299. What would hexadecimal A6 equal in base 2?
- 1010 0110
300. Minimizing the truth table would result in:
- $AC + /A/B + ABD$
301. Which of the gates would be used for the equation  $A + B$ ?
- OR
302. OR'ing 1FHex and 02Hex will result in which of the following hexadecimal numbers?
- 1F
303. The equation used to populate Karnaugh map, would be which of the following equations?
- $AB + CD$
304. The binary number, 0110 1110, would have what values for parity; even, odd, and no parity bits?
- 100
305. Verilog is part of the HDL family, what is the definition of HDL
- Hardware Descriptive Language
306. Which of the gates would be used for the equation  $/A + /B$ ?

- NAND
307. Reducing  $F = /ABC + A/BC + AB/C + ABC$
- $AC+AB+BC$
308. Using the Karnaugh map for minimum gate reduction would result in which of the following equations?
- $/B/D$
309. AND'ing 10H and 2FH will result in which of the following?
- 3F
310. How many bits would be required to count from 0 to 127?
- 7

## EEE-174 MT1

1. What is the hexadecimal encoding for "JNL" for a jump back 8 bytes? answer
2. In X86 architecture, ALU stands for which of the following? Arithmetic Logic Unit
3. The number of nibbles in a word are:  
 $16/4 = 4$
4. The instruction MOV BX, [2BAD] is what addressing mode? Direct
5. What is the hexadecimal encoding for adding DX with BX and storing the result in BX?  
?????
6. A microprocessor with a 32-bit address bus could access how much memory?  $2^4=16MB$  ,  $2^8=256KB$ ,  $2^{32}=4GB$
7. You add 7+6 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:  

|        |   |                                   |
|--------|---|-----------------------------------|
| 0111   | 1 | OF=1, the sign bit has changed    |
| +0110  |   | CF=1, there is a carryon of bit 7 |
| 0 1101 |   |                                   |
8. Which of the following DOS Debug instructions would be used change the IP register to 100?  
RIP
9. What are the contents of CX after this program has been run:

MOV DX, 11h  
MOV CX, [5512]

**MOV BX, 5511h**

SUB DX, [BX]

## Memory Contents

## location

5514 24

5513 D8

5512 00

D800h

\*\*\*What are the contents of DX after this program has been run:

FFF0h

10. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO

CY

1B72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register? -48

11. How many bytes are there in this short sequence of code? B815B400CD16CD20  
8

12. In using INT 10h to set the video code to 640x350, what value must be in the AH register?

Ax

AH|AI

00

00h

13. Moore's law has accurately predicted the growth rate in the number of

transistors per die for the last 25 years. What is that rate?

Doubling every 18-24 months

14 INT 21b Function 09h requires three things set up before calling in order to correctly print a string.

Hello msg They are:

DS=SEG Hello msg DX=OFFSET Hello msg Hello msg terminated with 24h

DS-SEG Hello\_

13A7:0110 CD 20 32 20 54 68 69 73-20 69 73 20 74 68 65 20

13A7:0110 CD 20 32 20 34 68 69 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 75 65 72 6D 0D 24 D9

13A7:0130\_00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in decimal?

16. Here is a short sequence of code: 7413CD16EB157D213C04EBF0A3C6. All of the instructions are two bytes long.  
The third instruction operator is: EB15

JMP

17. What is the hexadecimal encoding for loading DX with a word (value) from memory location 0820h?  
8B162008

18. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV UP EI PL NZ NA PO NC

1D72: 010D 7D09 JGE 0116

How many bytes will the processor jump if the condition for a jump were met? 9

19. F6 in 2's complement equals \_\_\_\_\_ in base 10.  
10

20. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV UP EI NG NZ NA PO NC 1D72: 010D 7D09 JNLE 0116

What will the IP value be after a "t" command is executed in DOS Debug?

??????

21. Determine the contents of register BH after the following instructions have executed:  
70h MOV [0202], AX

22. Given:

AX=2247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC

1D72: 0106 EB0F JMP 0118

What will be the IP value be after a "t" command is executed in DOS Debug? 0118h

23. Which of the following DOS Debug instructions would set a break point at memory location 010E?

G=100 10E

24. Given:

0B0E: 0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D 0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message? Welcome to Assembly Language

25. Given:

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 IP=0104 NV UP EI NG NZ NA PE NC  
1D72:0104 7002 JO 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0106h

26. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

What will the IP value be after a "t" command is executed in DOS Debug? 0110

27. The instruction MOV BX, 2BAD is what addressing mode? Immediate
28. What is the hexadecimal encoding for "JNL" for a jump back 10 bytes? 7DF4
29. How many address lines would be required to address 64MB directly?  
26 / $2^{26}$ =67mb/
30. The number of nibbles in a double word are: 8
31. In using INT 10h to move the screen cursor to return on the same line, what value must be in the AX register?

0E0Dh

32. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV U EI PL NZ NA PO NC 1d72: 010D 7D09 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug? 010Fh

33. Which of the following DOS Debug instructions would be used change the AX register?  
RAX

34. How many Bytes are there in this short sequence of code? B815B400CD168A3CCD20

10

35. Given:

AX=FFD0 DS=1D72

BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG  
NZ NA PO CY

1D72:0111 EB08 JMP 0119

- What will the IP value be after a "t" command is executed in DOS Debug? 0119h  
36. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010E OV UP EI NG NZ NA PO CY 1D72: 010E 7D06 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug? 0118

37. How many bytes decimal will the program jump for JNB instruction, given the following?

AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1376 ES=1376 SS=1376 CS=1376 IP=0100 NV UP EI PL NZ NA PO NC  
1376:0100 73E0 JNB 00E2  
-32

38. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV UP EI PL NZ NA PO NC  
1D72: 010D 7F09 JNLE 0118

What is the decimal value of the signed number in the AX register? -0032

39. How many address lines would be required to address 512 MB directly? 29 / $2^9$ =29/

40. What is the hexadecimal encoding for adding BX with CX and storing the result in BX?  
03D9

41. The binary number, 0111 1110, represents what values; in Hex, and as a BCD number?

7E, 7 invalid

42. In using INT 10h to set the video mode to 640 X 200, what value must be in the AX register?

0006h

43. what is the hexadecimal encoding for loading AH with a word from memory location 0520h?  
8A262005

44. What is the hexadecimal encoding for "JGE" for a jump back 10 bytes?  
7DF6

45. Given:

13A7:0110 CD 20 48 20 54 68 69 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
13A7:0120 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in bytes in decimal:

72

46. Given:

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0106 NV UP EI NG NZ NA PE NC 1D72:0104 7002 JNO 0118

What will the IP value be after a "t" command is executed in DOS Debug? 0118

47. Determine the contents of register AH after the following instruction have executed:  
MOV BX,BA70H

48. In using INT 10h to move the screen cursor to return to the beginning of the line, what value must be in the AX register?

0E0Dh

49. Given:

1376:0110 48 61 76 65 20 61 20 67-72 65 61 74 20 53 70 72  
1376:0120 69 6E 67 20 52 65 63 65-73 73 21 00 00 00 00 00

An ASCII message begins at memory location 0110, what is the message? Have a great Spring Recess!

50. You add 1+8 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:  
OF=0, CF=0

Page 16

51. What is the advantage of Assembly Language over C Language? The Assembler creates much faster executable code

52. In The X86 lab 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence "Hello World 0"?

020E

53. Which of the following DOS Debug instructions would set a break point at memory location 010C?  
G = 100 10C

54. If CX is 0000 what will CX be after a "LOOP" instruction?  
FFFF

55. Given:

AX=0353 BX=0534 CX=0000 DX=0180  
DS= 1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 (OV=1,NV=1 ZR=1, NZ=0)  
010B

56. How many core does the propeller microcontroller have? 8  
57. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?  
7DF2

58. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15.  
instructions are a word long. The third instruction operator is:  
INT

59. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?  
24

60. The ASCII codes for space , space, carriage return, line feed, end of string in decimal are:

32,32,13,10,36

61. A "NOP" instruction in a program will:

62. How many address lines would be required to address 64MB directly? 26

63. What command in DEBUG would be used to execute interrupts?

64. What high level language is the propeller programmed in? Spin

65. Which of the following DOS Debug instruction would be used to change the IP register to 010C?  
RIP

66. The acronym PWM used in the Parrallax Propeller and MicroChip PIC18, is defined as:

Pulse Width Modulation

67. Which command would you use to execute another core in the propeller microcontroller?  
Cognew

68. Given: IP=0111 NV UP EI NG NZ NA PO CY  
1D72:0111 JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug? 0119h

69. The instruction MOV CX, DADD is what addressing mode? Immediate

70. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?  
A0

71. In the Propeller microcontroller, the command "dira[4..9] := %111111" would cause the processor to do with of the following?

Sets the Propeller pins P4 through P9 as output pins

72. In the Propeller microcontroller, the command "waitcnt(clkfreq\*10 + cnt)" would cause the processor to do with of the following?

Create 10 second delay

73. The "LOOPNE" instruction is equivalent to which of the following instructions?  
DEC CX, JNE

74. On the Arduino platform, what is the programming language used?  
C

Moore's Law - the number of transistors in a dense integrated circuit doubles approximately every two years.

Acronyms:

IDE – Integrated Design Environment

BIU – Bus Interface Unit

ALU – Arithmetic Logic Unit

Spring 2014 Exam questions:

1. How much global RAM does the Propeller microcontroller have?

32 kB

2. How many lines would be required to address 128 MB directly?

27

3. If you want to use a INT software interrupt function to print a string out to the screen, what is the function code, start pointer, termination character, and interrupt you need to use?

ah = 0eh, es:dx, “:”, 10h

4. Moore's law has accurately predicted the growth rate in the number of transistors per die in the last 40 years. What is that rate?

Doubling every 18-24 months

5. What command in DEBUG would be used to execute interrupts without stepping through the interrupt instructions?

T – trace command

6. In x86 architecture, ALU stands for which of the following?

Arithmetic Logic Unit

7. What is the hexadecimal encoding for adding BX with CX and storing the result in BX?

8. On the Arduino platform, what is the programming language used?

C Programming Language

9. Which of the following DOS Debug instructions would be used to change the AX register to 010C?

10. Program Listing: MOV BL, E2H: MOV CL, 06H: ROL BL, CL.

E2Hex

11. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

Bit 7 of port D is set to output

12. What high level language(s) is (are) the propeller programmed in?

Spin language

13. A microprocessor with a 26-bit address bus could access how much memory?

64M

14. The instruction MOV CX, [DADD] is what addressing mode?

Direct

15. Given:

0B0E: 0200 57 65 6C 63 6F 60 65 20-74 6F 20 41 73 73 65 60

0B0E: 0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message?

Welcome to Assembly Language

16. How many cores does the propeller microcontroller have?

8 cores or cogs

17. In x86 architecture, BIU stands for which of the following?

Bus Interface Unit

18. In the PIC 18 with TRISD = 0b011110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

AA

19. In the Propeller microcontroller, the term “Method” is (are) which of the following?

a block of executable Spin commands that has a name, access rule, and can optionally create local (temporary) variables, receive parameters, and return a value.

20. What is the number, 1010.01012 in decimal?

10.31

21. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:

Pulse Width Modulation

22. No Q

23. AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA P0 CY 1D72:0111 EB08 JMP 0119

What will the IP value be after a “t” command is executed in the DOS Debug?

0119h

24. In the Propeller microcontroller, the command “dira[4:9] :=%111111” would cause the processor to do which of the following?

Sets the Propeller pins P4 through P9 as output pins

25. The instruction MOV CX, DADD is what addressing mode?

Register

26. If CX is 0000 what will CX be after a “LOOP” x86 instruction?

0000

27. What are the contents of DX after this program has been run:

Memory Location    Contents

MOV DX, 11h    5514    24

MOV CX, [5512]    5513    D8

MOV BX, 5511h    5512    D0  
SUB DX, [BX]  
              5511    21  
AND BX, FFFF  
              5510    00

DX: -860

28. Which command would you use to execute another core in the propeller microcontroller?

Cognew

29. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

30. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 ST=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 0V UP EI PL NZ NA PO CY 1D72:0109 7D06 JGE 0118

What is the signed decimal value of the number in the AX register?

-16

31. How many address lines would be required to address 64 MB directly?

26

32. In the Propeller microcontroller, the command “waintcnt[clkfreq\*10 + cnt]” would cause the processor to do which of the following?

Wait for 10 seconds

33. AX=2247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72  
IP=0106 NV UP EI NG NZ NA PE NC 1D72:0106 EB0F JMP 011F

What will the IP value be after a “t” command is executed in Debug?

011Fh

34. The number of bits in a word are:

32

35. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

32, 32, 13, 10, 36

36. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 IP=010F NV  
UP EI NG NZ NA PO NC 1D72:010F 7D18 JGE 0128

How many bytes in decimal will the processor jump if the conditions for a jump are met?

37. The number of nibbles in a word are:

8 nibbles

38. In the PIC18 with TRISD = 0b01000000, what is the configuration of the Port D?

Bit 7 of port D is set to output

39. AND’ing 1FH and 02H will result in which of the following?

02

40. With this a short sequence of code: 7413EBA3CD167D213C04EBF0EB15, All of the instructions are a word long. The third instruction operator is:

41. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

42. In the Propeller, how many values does a method return?

As many values as there are objects in the method.

43. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?
44. Which of the following Debug instructions would be used to change the IP register to 0110?
45. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY 1D72:0109 7D06 JGE 0118

What will the IP value be after a “t” command is executed in DOS Debug?

46. The “LOOPNE” instruction is equivalent to which of the following instructions?

JNZ, DEC CX

47. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?
48. What command in DEBUG would be used to step through a program line by line?

T – trace command

49. Which of the following is a valid x86 command for multiplying a number?
50. What is the advantage of Assembly Language over C Language?

51. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

AA

52. No question
53. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72 NV  
UP EI NG NZ NA PO NC 1D72:010D EB07 JMP 0114

How many bytes will the processor jump if the conditions for a jump were met?

54. AND’ing 10Hex and 2FHex will result in which of the following Hex numbers?
55. In the PIC18 with TRISD = 0b11111111, what is the configuration of the Port D?
56. AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
57. DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC 1D72:010D 7DF6 JNL 0116

How many bytes in decimal will the processor jump if the conditions for a jump were met?

58. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:
59. Which of the following Debug commands would set a break point at memory location 010C?

G = 100 10C

60. In the PIC18 with TRISD = 0b10000000, what is the configuration of the Port D?

Bit 7 of Port D is set to input

- R ~ r n t r c ~ .

1. A "PUSH" instruction:
- a) increments the SP
  - b) increments the IP
  - @) decrements the SP
  - d) points to the data inputted from the keyboard
  - e) stores the returning address
2. If CX is 0000, what will CX be after a "LOOP" instruction?
- q ::::  
/--3 ;2F  
e) 0003
3. You are trying to rebuild a HELLO program project in MASM and you get the following error: "ERROR 4 line 1". What is the cause of the error?  
4 MASM isn't installed correctly on the computer being used

- b Not known--this error by itself isn't a problem, press the enter key to clear the error No source file is identified (no .asm file)
- d) No project template for COM was selected
  - e) No project was setup
4. In MASM, with a "MOV CX, 18" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?

5. You are trying to rebuild a HELLO project program in MASM and you get the following error: "LINK : warning L4021: no stack segment".

What would be the reason for such an error?

- a) MASM isn't installed correctly on the computer being used b No .mak file specified
- 4? " "
- e) No project was setup

6. In the Hello MASM lab in the original code, what is the address of the byte used to start the string in the sequence "Hello World O"?

- 0200  
b) 020E  
c) 0100  
d) message  
e) Hello

EEE 174  
Midterm 2, F08 Dahlquist Page 2

7. What is 14.437510 in binary?

8. How many bit(s) are required to represent a range of numbers from 0 to 63?

- a) 2

9. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

~s=i376 ~s=i376 ~ ~ 1 1 3 7 cs=i376

I c n a

UP

EI

PL

NA PO

1376:0100

0100

ADD

[BX+SI],AL

DS:0000=CD

- a) MASM
- b) EXE

- g,"&::
- e) ADD

10. What command in DEBUG would be used to change the code segment?

r n ~  
-v RCS  
/ c) p

- d) RIP

- e) T
11. This section of memory represents a stack. What type of program is this?
- |           |             |       |    |    |                         |    |                            |
|-----------|-------------|-------|----|----|-------------------------|----|----------------------------|
| BEEF:FFDO | 00          | 00 00 | 00 | 00 | 00                      | 00 | 00-00 00 00 00 00 00 00 00 |
| BEEF:FFEO | 00          | 01 02 | 03 | 04 | 05                      | 06 | 07-08 09 OA OB OC OD OE OF |
| BEEF:FFFO | 11 22 33 44 | 55    | 66 | 77 | 88-99 AA BB CC DD EE FF |    |                            |
- EXE program with overlapping segments  
 /  
 COM program  
 d) COM program without a PSP  
 COM program with different segments

the contents of register BL after the following instructions have been executed:

- c) 2EH
- d) 4EH
- e) 2H

EEE 174 Page 3  
 Midterm 2, F08 Dahlquist

13. What Hex values must be sent to address the key pad rows on the PPE board?
- a) 1,2,3,4
  - \*I- 1,2,4,8,

- e) 378,379

14. The ASCII codes for space, space, carriage return, line feed, end of string in hexadecimal are:
- a) OD, 30,20,20,24
  - 4b 20,20, DO, A0 24
  - 20,20, OD, OA 24
  - 32, 32, 13, 10,36

- e) SP, SP, CR, LF, \$

15. Which of the following is a valid x86 command for multiplying a number?
- 4        a        )        MULBX

#### Program Listing

```
Mov BX, 0500
Push BX
```

```
Mov AX, 0100
```

```
MUL CL,BL
```

16. What command in MASM-CodeView would be used to step through a program line by line? a T (F8)

17. Given the short code, what is the value in AX after the Fa":program is run?  
\$ A d r o o

18. A "POP" instruction: 8a) increments the SP

increments the IP

c) decrements the SP

d) points to the data inputted from the keyboard  
e) stores the returning address

19. A "NOP" instruction in a program will:

- a) Set a Normal Operating Point
- b) Clear the Overflow flag
- 9c) Perform a No Operation
- ) Reset the IP register
- e) Exit the program

EEE 174  
Midterm 2, F08 Dahlquist Page 4

20. What is the numeric sequence of the key pad columns on the PPE board used in the lab?

- @ 2,4,6,8
- b) 1,2,4,8,
- c) 1,2,3,4
- / -d) 37,2F, IF
- e) 378,379

21. For the instruction sequence below, determine the contents of the register AL after this program is executed:

- a) F2H
- b) -62H
- c) FCH
- 62H - A IDAA I
- e) 162H

22. Which of the following is not a valid command for a number into a register in MASM?

- e) MOVAH, ~ B C H

23. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 04h and an input port value of 2Fh?

- a) 112,418,

24. AND'ing 20H and 1FH will result in which of the following?

20        O o I o ~ s o o  
Door      c c  
\_\_\_\_\_  
0 k - c C~CC,L\DI

25. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

- a)        ABCD

26. If the SP is FOOF, what will the SP value be after a "PUSH CX" instruction?

\$Ffg

- d)        FOOC  
e)        F012

S2010 MT 1

1.        Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:

DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36

2.        Moore's Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?

Doubling every 18 – 24 months

3. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What will the IP value be after a "t" command is executed in DOS Debug?

010B

4.        A "NOP" instruction will:

Perform a No OPeration

5.        Given:

AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0106 NV UP EI NG NZ NA PE NC

1D72:0106 EB0F JMP 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0118h

6.        In x86 architecture, BIU stands for which of the following?

Bus Interface Unit

7. Determine the contents of register BL after the following instructions have been executed:

2EH

8. The number of bytes in a double word are:

4

9. In x86 architecture, ALU stands for which of the following?

Arithmetic Logic Unit

10. The “LOOPNE” instruction is equivalent to which of the following instructions?

DEC CX, JNE

11. The instruction MOV CX, DADD is what addressing mode?

Program Listing MOV BL, E2H MOV CL, 100b ROL BL, CL

Immediate

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

INT

13. Which of the following DOS Debug instructions would set a break point at memory location

010C?

G = 100 10C

14. How many bytes are there in this short sequence of code? B400CD164CCD21

15. How many address lines would be required to address 128 MB directly?

27

16. What are the contents of CX after this program has been run:

D800h

17. 1010 0110 in 2's complement equals\_\_\_\_ in base 10.

#### Memory Location

1103

1102

1101

1100

#### Contents

24

D8

00

21

-90

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

01D8

19. What is 11.437510 in binary?

001011.01110

20. If CX is 0000 what will CX be after a “LOOP” instruction?

FFFF

21. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?

7DFA

22. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010E OV UP EI PL NZ NA PO NC

1D72:010F 7D08 JGE 0115

How many bytes will the processor jump if the conditions for a jump were met?

8

23. What command in DEBUG would be used to execute interrupts?

P

24. What is the advantage of C Language over Assembly Language?

C is transportable to other microprocessor architectures

25. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

OF = 1, CF = 0

26. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

36

27. Given:

AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI PL NZ NA PO NC 1D72:010D 7DF6 JNL 0116

-10

28. AND’ing 1FH and 20H will result in which of the following?

0

29. Given:

13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

72

30. A microprocessor with a 31-bit address bus could access how much memory?

2 GB

31. For the instruction sequence below, determine the contents of the register AL after this program is executed:

51H

32. Given:

AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

-256

33. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

32, 32, 13, 10, 36

Program Listing MOV AL, 73h ADD AL, 78h DAA

S2010 MT2

1. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?

7

2. If CX is 0003, what will CX be after a “LOOP” instruction?

0002

3. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?

A 5 second delay

4. What is the number 1011.01012 in decimal?

11.31

5. This section of memory represents a stack. What type of program is this? BEEF : FFD0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

COM program

6. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

BDCA

7. Determine the contents of register BL after the following instruction have been executed:

2EH

8. What Hex values must be sent to address the key pad rows on the PPE board?

1, 2, 4, 8

9. What is the number 32.437510 in binary?

Program Listing MOV BL, 2E MOV CL, 10

ROL BL, CL

100000.01110

10. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

24

11. The acronym PWM used for motor control, is defined as which of the following?

Pulse Width Modulation

12. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

Bit 7 of port D is set to output

13. Given the short code, what is the value in AX after the program is run?

0500

14. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?

ZF

15. How many nibbles are in double precision IEEE floating point format number?

## Program Listing

MOV BX, 0500

Push BX

MOV AX, 0100

POP AX

16

16. In the Hello MASM lab in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

020E

17. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1476 ES=1576 SS=1376 CS=1D72  
IP=0015 NV UP EI PL NZ NA PO NC

1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD

EXE

18. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?

F00D

19. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

A0

20. What is the numeric sequence of the key pad columns on the PPE board?

37, 2F, 1F

21. What is -130 decimal in 2’s complement (8bits)?

01111110

22. Which of the following is a valid x86 command for multiplying a number?

MUL BX

23. The number of bits in single precision IEEE floating point format are:

32

24. You are trying to rebuild a HELLO project program in MASM and you get the following error:  
“LINK : warning L4021: no stack segment”.

What would be the reason for such an error?

No project template for COM was selected

25. A “pull down” resistor is used in digital circuits to do what?

To keep the signal line “tied” low until the line is active (goes high)

26. A “POP” instruction: increments the SP

27. AND'ing 10H and 2FH will result in which of the following?

0

28. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

Sets the Propeller pins P4 through P9 as output pins

29. What commands in MASM-CodeView would be used to step through a program line by line?

T (F8)

30. If the SP is F00F, what will the SP value be after a “POP SP” instruction?

F011

31. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?

0

32. In the Propeller microcontroller, the term “Method” is(are) which of the following?

A block of executable commands that has variables, can receive parameters, and returns a value.

33. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?

MOV CX, 30H or MOV CX, 48

34. A “pull up” resistor is used in digital circuits to do what?

To keep the signal “tied” high until the line is active (goes low)

```
'1      -'t:'tcpm,  
AKUTIU'f,'lz:uu  weqnesoay, ucroDer'r+, zvvo  
Midterm#1:Opentext Timeallowed:75 minutes   F2009Dahlquist
```

1. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=TD7Z SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:01097D06 JNL 0118

What is the signed decimal value of the number in the AX register?

F r r t ,'  
l j i l ]1)jr \  
 i . - \

2. What are the contents of BX after this program has been run:

- a) 5511h
  - b) FFFoh
  - c) -0010h      Memory location    contents
  - d) 0010h      MOV AX, 11h      1103      a+
  - e) D800h      MOVC(r t11011 ].toz      D6
- 1\*\*mJ      MOV BX, 1100h      1.rol      00  
               suB AX, [ex]      It00      Z L

3. What is 16.437510 in binary?

- a) 100000.11101
- b) 001110.01110

4'3"1"--  
,,,'t  
'i.\T: ;,

tffii  
E)"-aoffrifiurom

)  
, .t \;, I

4. Here is a short sequence of code: znifisr;#;itqtoztbco+herqhBt+ Au of the  
instructions area word long. The forth instruction operator is:

- a) cMP - i i ) ?  
b) INT \  
ffi ( , r1i 1l r t i( i r l r )

e) lNz

5. Determin the contents of register BL after the following instructions have been executed:

- lIJFStl Lz\ - ..1{} - F - sErTEii's  
ProgramListing  
b) 4H | : | ..) f )' MOVB{BL.E2H,  
c) 2EH ? tMoV  
d) 4EH . ' - r MOVCL,1000b  
ROL BL, CL

e) 2H

6. Which of the following DOS Debug instructions would set a breakpoint at memory location 010E?

- a) G 0E10  
ffi  
YJ--:-UIOE:  
e) c [010e]

7. The jLQQPNZ instruction is equivalent to which of the following instructions?

- DECCX,JNE u,  
c) tNccx, JNz /1e,,,'it  
d) JNZ,DECCX 4  
e) JZ, tNCCX {i

/1 ,',,,, i '

'EEE 1 S8. ty - 70,Ll6 c

, 174- CpE185 ,qr/  
Midterm1 F09Dahlquist 741

8. Given:

A)(=FFEOBX=3534CX=0000DX=0180Sp=FFEEBp=0000 Sr=0000 Dr=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=OIOD OV UP EI PL NZ NA PO NC  
1D72:010D7D09 JGE 0116  
How many bytes will the processor jump if the conditions for a jump were met?  
d) 10D  
e) 116

9. Given:  
A)(=0353 BX=0534cX=0000 DX=0180SP=FFEEBP.=0000Sr=0000 Dr-0000  
IP=0109 'OV/UP  
DS=TD7? ES=1D72 SS=1D72 CS=1D72 EI PL NZ NA FO CY  
1D72:01097D06 INL 0118  
What will the value of IP be after a "t" command is executed in DOS Debug?  
Eff  
c) 0110  
d) 0118  
e) LD72

10. Int 21h, Function 09 requires three things set up before calling in order to correctly print a string,  
Hello\_msg. They are:

ES= SEGHello , DX = OFFS\_Ef"ll,9-[f-q  
= sEq Hello-msg, Ex = G, FFSETHello-msg, \_msg terminated with "\$"  
d) DS= SEGHello\_msg, AX = OFFSETHello-msg, Hello  
\_msg terminated with 24h  
e) CS = SEGHello\_msg, DX= OFFSETHello-msg, Hello  
\_msg

11. How many address lines would be required to address 64 MB directly?

a) 5L2 z o - ' zt\6'  
b) 64  
c) zG  
d) 24

IEI--lo\

12. How many bytes are there in this short sequence of code? 1{ \*rl#rfd

w 3 . | , ? , , t i l i ' t ,  
a )

d) t2  
e) 48

13. Given:

IX=2247 BX=0000 CX=0000 DX=0000 SP=FFEEBP=0000 Sr=0000 Dr=0000 DS=1D72 ES=LD7 ZSS=1D72 CS=1D72 RP=0105  
NV UP EI NG NZ NA PE NC  
1D72:0106EB0F JMP 0118

What will the IP value be after a "t" command is executed in DOS Debug?

a) 0107h \ i \  
:i 81?8[ t."rrr'  
nJcc-Yic\ i"FF  
ff

T'rTzitE  
c) 0100 \,  
;i messaseY)  
e) Hello

ffi  
16. 10100110in 2'scomplement equals in base10.  
\*h) 90 ,/lo\o cr \ \|(2 (1  
bi -106 ' o t o i l r \_ 1 ) I  
"1 i 'tr  
17. Given: i . ) i ' ! 1 i i " ) 1 { n

1 " J /  
13A7:0110 cD 20 32 2 }iitAI48 69 73-20 69 73 20 74 68 65 20  
13A7: 0120 66 69 72 73 TTZO +o 69-64 74 65 72 6D OD24 D9  
13A7: 0130 00 c6 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An inputbuffer is at memorylocation0114, what is the size of the buffer in decimal?

- a) 20 'o q \* > '5 r : , - '.
- b ) 5 0 . { , tr , l : r . u -
- c) 32 'i \$ + . i . f 'r . i
- d#

18. A "NOP" instruction in a program will:

- a) Set a Normal OperatingPoint
- b) Glear the Overflow flag

ffi  
"rdl\*Rffi

- e) Exit the program

19. If CX is 0002 what will CX be after a "LOOP" instruction?

- a) 0000
- w
- c) FFFF
- d) 0002
- e) 0003

20. In x86 architecture, ALU stands for which of the following?

- a) A Logic User
  - b) Address Logic Unit .
- ffiiil  
ti-ffiflosie\*Ufrli\*\*
- e) Arithmetic Lining Unit

)l  
1{

EEE174- CpE185

Midterm1 F09Dahlquist      Page 4

21. The instruction MOV CX, IDADDI is what addressing mode?

- a) Scaled Index
- b) RegisterIndirect
- c) Register

d) Immeditate

f ei-Eirectl

1...."---\*\*\*r'

22. In adding 5+7 through a 4 bit integer unit. The state of the OF and CF flags after the add

instruction would be: O\O t

- a ) O F = 0 , C F = 0            1 0lr1
- \ - bffitrl;eF=il
- : . - # - . . \* \* - "
- 'c )        O F = 1 , G F = 0    m ',i i { r , " t
- d )        O F = 1 , C F = 1    u \*

e )        O F = 0 , G F = 0 , Z F = 0

23. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?

- a) TDFA 0            \7
- ffi'                = | lcn"
- >4lh (        o'1
- .f^l        a1 ,    L r l i '
- '{\_M                c : r l , I i , r - . L ;

24. Moore's law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is that rate?

- a) Doubling every 18 - 24 hours
- b) Doubling every 18 - 24 weeks

25. In x86 architecture, BIU stands for which of the following?

- a) BestInterfaceUnit
- b) Bus InteqratedUnit
- ici-ffiSTteffiu-iti
- dl-Bl6gTTiiGiradeTir
- er

e) Best IntegratedUnit

26. F6 in 2's complement equals \_

1 1 1 0 1 1 0 r r r t

b) - 600c,0F)Zi  
rdf:E1 i ----t---  
l u  
e) c t t , p

in base 10.  
O t t O (' ! \ \ 1 o ) i o  
1c' o I da . 1 . ' , t l I ( , D j  
l I I 1 . . r | ) . ' " . ' ; '  
l o 1 c ) . - a . - f )  
t L t > , " r 1  
? . , 4 '

27.\*A,m1clgpfocessorwith a 32-bitaddressbus couldaccesshowmuchmemory?

Td: J <? (^z|r t3o \ /  
b) 1 6 M B e a \ L , ' r ' /  
c) 4 MB /\*.t n.^.,,  
d) 1 2 8 M 8 , \  
e) 256MB

Al EEE174-CpE 185

Midterm 1 F09 Dahlquist

28. AND'ing 10H and 2FH will result in, which of the following?

w C) O o \ A r . : P(X)t()  
c) 16  
d) 45 O O O t : >  
e) 63

Page 5

(xxlt:

\ \ \ \ \

P O I ) O h : D ' {

29. For the instruction sequence below, determine the contents of the register AL after this program

is executed:      AL - \crD      cDll      s ?  
 a)      FIH      +      0)I?"- \uj?  
 TTT-TFII      ProgramListing      !\*L  
 MOV AL, 83h  
 d)      -51H      Ir1{} tt)l\ ADDAL,68h  
 E B      DAA  
 l e )      1 5 1 H

30. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

- a) 0D,30,20,20,247.6 'Lo
- b) 20,20,D0,A024
- @
- ff
- e) SP,SP,CR,LF,\$

31. In MASM, with a "MOVCX,18" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?

- a) 0C
- b) 24
- /
- ff
- ./ d) 12
- e) 36

32. What command in DEBUG would be used to execute interrupts?

- a ) R
- b) Rcs
- rcrFl
- d) RIP
- '
- e ) T

33. The number of nibbles in a doubleword are:

- a) 32      b V4      - 't,-xi\
- b) 16      JnlSJ.      Z \*'t,i',
- w      t"rfA      t-;IL"\*,," ;;il4
- e ) z

, - r ,

.rvrvrttryg1

i V t      - V V v

Midterm#2: Open note, text Time allowed: 75 minutes

F2009 Dahlquist

1. Determine the contents of register BL after the following instructions have been executed:

- b) 4H      Program Listing

- c) zEH MOVBL.2EH
  - d) 4EH MOVCL .0100b e - tl
  - e) 2H ROL BL. CL

2. What Hex values must be sent to address the keypad rows on the PPE board?

- a) 2,4,6,8
  - b) 1,2,3,4
  - ffi
  - d) 37,2F,1F
  - e) 378,379

3. With a POPBX instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

- a) ABCD
  - b) DBAC
  - c) AGDB
  - d)-6nBDCA

F

- |              |                   |                                                                   |                            |
|--------------|-------------------|-------------------------------------------------------------------|----------------------------|
| r'.ii        | FFBS              | I                                                                 | \$ / O \\<br>c, "l\`t:i'\` |
| (            | c ) - 8 . 5       | . -- t                                                            | .                          |
|              | d ) B . 5         | c/                                                                | , 'fi                      |
| u \ ,<br>" " | e ) , 1 1 . 3 1 r | Ii,\`i                                                            | )r                         |
| 5            | - f,, ["] , '1    | if GW is 0000_1, then ill GW has a file named "LQCD" in section 2 |                            |

5. If GX is 0000, what will CX be after "LOOP" instruction?

- a) 0000
  - b) 0001
  - c) 0002
  - d) 0003

- How many bit(s) is/are required to represent a range of numbers?

  - a ) 1
  - b ) 2
  - c ) 4

d) 7

:r

, - { --

y\

7. What is

7. What's 1101001010101010 in binary?

- a) 001110.01110      '1.      'i      I      i ! - .

b) 001111.01010

I

$\{ \dots, i \} \subseteq$

$U * i$

i

I

1

EEE174 fue M\* Page 2

Midterm2, F09Dahlquist

8. fn MASM,with a "MOV CX,12h" instruction, and a "LOOP" instruction, in decimal how many times

will the program loop?

- a) 0C
  - b) 24
  - #
  - l c ) 1 8 i
  - d ) 1 2
  - e) 36

(rrrti

11

\r", "

4

,L

9. What is the binary value of decimal 12.875?

- a) 1011.1110 l. -r.-., .. O I  
 -' b) 1 1 0 0 . 1 1 0 0 1 " 1 t;; ("

d) 1 1 0 0 0 1.10 i {}  
 e) r.

10. What is the numeric sequence of the keypad columns on the PPE board?

- a) 2,4,6,8  
 b) 1,2,4,8  
 cl 1.2.3.4  
 W

11. This section of memory represents a stack. What type of program is this?

BEEF:QFD0000000000000000-0000000000000000

BEEF:QFE0001020304050607-08090A0B0c0D0E0F

BEEF:QFEQJt 22 33 44 55 66 77 88 - 99 AA BB CC DD EE FF

a)- EXEprogram\_overlappingsegments

## Program With

-i 'b)' "EXE program / -i: " ; l - ' ; l - it rrfb

c) COMprogram

d) COM program without a PSP

e) COM program with different segments

12. Given the short code, what is the value in AX after the program is run?

```
(,  
 a) 0000 \{ tI ProgramListing  
 b) oo01  
 c) 0005 MovBX.0100  
 ' d 1 0 1 0 0 ( - L r r r ' { c L PushBX  
 ,/ FII5oT\ Mov AX, 0500  
,/ \-- POPAX
```

13. What is -32.7510 in binary? 4,2\*- . \o0 Oclo

- a) - 1 0 0 0 0 0 . 1 1 0 0 0 E  
( e O 1 i I I )
- b) 1 0 0 0 0 0 . 0 1 1 1  
c) 1 1 1 1 0 0 . 0 1 1 1 . t . - .  
-\*>,:' ! r ,r' r"; i"! I
- d) 0 1 1 1 1 1 . 1 1 0 0 2;""]-

0 0 . 1 1 0 0 0 i' ),'o'l

14. What command in DEBUG would be used to change the IP value?

- a) RcS  
trFEi  
c ) R  
d)r

e ) P

EEE 174 Page 3  
Midterm 2, F09 Dahlquist

15. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0000 SP=F1EE BP=0000 S I = 0 0 0 0 D I = 0 0 0 0

D S = 1 4 7 6 E S = 1 5 7 5 5 5 = 1 6 7 6 C S = 1 3 7 6 I P = 0 0 1 5 N V U P E I P L N Z N A P O N C  
L 3 7 5 : 0 0 1 5 0 1 0 0 ADD [ B x + s I ] 'A1 DS : 0 0 0 0 = C D

A} MASM ! 60"t{'

f6fi-EiFl 1,.\*r^L 1{,'Ln' II2 r>tcn

at"q{,n4#

c) Debug

d) coM

e) ADD

16. What flag(s) does the "LOOPNZ" instruction look at to determine whether to loop or not?

- a) Cf  
b) OF and CF  
c) SF and OF  
d) SF and ZF  
H.n

17. Which of the following is a valid x86 command for multiplying a number?

0c40FH

C1 v11'I  
AX, BADH  
e) MUL10H

18. How many bytes are in double precision IEEE floating point format numbers?

a) 2q

19. What is -130 decimal in 2's complement (8 bits)?

Lel--011i1-1T01

I t O : J

b) 10000010

tODboorc

c) 10011000

d) 111100',11

e) 11'111110

O t \ \ \ I o r

ol f \ \ \ \

D

'4/

.-f |

20. If the SP is F00F, what will be the value of the "PoP cx" instruction?

a) F010 U\* d \*i\,,{ir, ' ' ' ' /t fr  
ffi F00D irii I (

b)  
-dl-Fooc S  
e) F012

1- \ i. fr"uv"  
fr"uv"  
i. irq

f,i,,,;1rl1Pcl.,uo'

precision of floating point format?

21. What is the decimal value of C55A57 00 in IEEE single precision?

p,'  
-r ,\,-,-,1f,'1.- T:1"-?lo-i- roro  
' [i.ro:rifi\*-ffi-e43751 ffi}ie i ,  
I el bl' - :f0ilEiE - S - r(i)D',,vi'|l.,l,: .. t niiolo'r\oi."r  
. -l,t93,tt)1r,, -209.9125 - (t){"t.1.,^"r)!,,r) }r.'\*:, \*r,,+, - '}u,,,,-  
- i,J ^,,.+.\*., ,tJi  
\ - / - ,\ / ^ rjq-,1\*t | . 1otjl)x (- \z /L'''''u'\\*,,-:ol1n . L77

i I O O D , i . )  
{ J C I o I

# n l O I L  
,, b I : .  
l i t o o  
q ; 0 1 1 , r ' D  
' S l r o I 1 ?  
1 i  
| , p o a 1 r  
\ { } C I I i ' - i  
i S { 0 , r l  
' ? t x , i  
t { 0 o , ''  
l x # t ' r , r t r t l . ' i  
  
' l x I f :

q\*/

EEE'174 Page4  
lr/lidterm2, FOgDahlquist

22. On the PPEboard, what number(s) on the keypad is(are) pressed for an output port value of 04h and an input port value of 2Fh?

- a) 0
- b) 1,2,4,8,

rF.) c) 8  
/ [ttfuffi,h  
,/ %rc-7sFru.\*

23. You are trying to rebuild a HELLO project program in MASM and you get the following error: "LINK: fatal error L1089: HELLO.lrf: cannot open response file".

Wha!\_W.guldJettetason.lo\*t\*sJ\$h.an,e;rsm...  
a1"-1to\_9-9\_fi!e"urce.a --!lgE[ified(n9.asmfile)\*i  
b) Nqpleie"e"lwa9  
99t\_r\*)  
ci\* r,ro;. miil; iile'ipidirieu

- d) No listing file was selected (or requested)

e) MASMin't installed correctly on the computer being used

24. Which of the following will cause a program with a LOOP instruction to loop 48 times?

26. What is(are) the advantage(s) of G Language over Assembly Language?

- a) C does not need a compiler to be assembled into an executable program.
  - b) The C assembler creates much faster executable code

l{ ;' 4t:r'0,|.1.i,r:1",..1;.,r,7'.f.lrr'."r | i i at.;  
lrl1i 't

27. The number of bits in single precision IEEE floating point format are: a ) 10

- b) 20

e) 80

28. Which of the following is not a valid command for a number into a register in MASM?

( '1 \$ \ : r '

EEE174

Midterm2, F09Dahlquist

29. In the HelloMASMlab in the original code, what is the address of the byte used to thesequence "HelloWorld0"?

Page 5

stand the string in

- @
- b) 020E
- c) 0100
- d) message
- e) Hello

30. You are trying to rebuild a HELLO project program in MASMA and you get the following error: "LINK: warning L4021: no stacksegment".

What would be the reason for such an error?

a) MASMA isn't installed correctly on the computer being used

was  
-\*Ig\_\_" \*o'Jlrqg\_-tile-'is'ident  
-\*Noproject was set up

31. How many nibbles are in double precision IEEE floating point format numbers?

- a) 2 , x ? p , t r
  - b ) 4
  - c ) 8
- mT-T6-{1  
:,\_\_e\_

32. A "pull down" resistor is used in digital circuits to do what?

- a) To keep Jh-eSgnall tied "high" until the line is active (go "eg-l-ow)- low until the line is active (goes high).  
ULta\_fgef-tlgg\_g\_1\_?!"lle'l'tied'i \
- c) To keep the voltage at 0 Volts
- d) To keep the voltage at 1 Volt
- e) To make sure the digital line is always high

33. The acronym PWM used for motor control, is defined as which of the following?

- a) Parallel Width Manipulation
- b) Parallel Wide Manipulator
- I ci Pulse Width Modulator \
- , r : '
- d i P o o r W o r i d e r M e r a n i p u t a t o r -r/
- ei Parallel width Modulator "

Midterm #1: Open lext Time allowed: 50 minutes S2007 Dahlquist  
i V

1. Given:

AX=FFD0 BX-3534 CX=0000 DX-0180 SP=FFEE BFH3000 SI^OOOO DI-0000  
DS=ID72 EF=ID72 SS=ID72 CS=ID72 IP=010E OV UP EI PL N2 NA PO CY  
1D72:0109 7D06 3NL 0118

What will the IP value be after a "t" command is executed in DOS Debug?

a) 010B

IJU10  
0118 I'V  
e) 1D72

2. The instruction MOV BX, 2BAD is what addressing mode?

a) Direct

b) Register

<-CcKT)m mediate

d) Scaled Index

e) Register Indirect

3. What is the hexadecimal encoding for "JNL" for a jump back 10 bytes?

o

EB16

e) EBF4

4. How many address lines would be required to address 64 MB directly?

a) 512

b) 64 20 ID

1 I  
Mb

e) 20

5. The number of nibbles in a double word are:

a) 32

b) 16

dj

6. In using INT 10h to set the video mode to 640 x 350, what value must be in the AX register?

a) 10h

)\_-1000b

OOh

e) OOIOh

7. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:

a) CS = SEG Hollomsg, DX = OFFSET Hellornsg, Hello\_msg terminated with 24h

b) DS ■ SEG Hellojnsq, AX = OFFSET Hello\_msg, Hello\_msg terminated with "\$"

S = SEG Hello\_msg I BX = OFFSET Hello\_msg, Hello\_msg terminated with "S" S = SEG Hel!o\_msg, DX = OFFSET Helloinsg, Hello\_msg terminated with 24h

e) ES = SEG Hello\_insg, DX = OFFSET Hello^msg, Hello\_msg terminated with "S"

EEE 174

Name V fe.j.

Midterm 1S07 Dahlquisl

Page 4

\ -

21. What are the contents of DX after this program has been run: ?« i m tfl

a) 5511h

cr^JDIOh

d) OOIOh

MOV DX, llh

e) D800h

MOV CX, [5512]

r :\'>'■' \_ fc i c t t'•i MOV BX, 551lh  
SUB DX, [BX],

' ' ' .f""';?a \*S

22. In using INT 10h to move the screen cursor to return on the same line, what value must be in the AX register?

a) EOAh

d) OEOAh

e) 0012h

a

23. Given: \ ,  
AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BRpOOOO SI4OOOj9^ DI=0000

DS=ID72 ES=ID72 SS=ID72 CS=ID72 IP=010D OV UP EI PL NZ NA PO NC

1D72:010D 7D09 3NLE 0118

What will the IP value be after a "t" command is executed in DOS Debug? -s -

c) OIOEh is )'

d) OIODh \

e) 0009h

24. Which of the following DOS Debug instructions would be used change the AX register?

a) G AX

C) RAX = 0110

d) RBX = 100 110

e) RIP = [0110]

\ |2^;U 5 ,G >' V^ ■ I?

25. How many Bytes are there in this short sequence of code? B815B400CD168A3CCD20

a) 6

b) 7

O 8 d) 11 e) 23

EEE 174 Introduction to Microprocessors Name

BRH 101, 12:00 -1:15 pm, Thursday, March 9, 2006 Midterm #1: Open text Time allowed: 50 minutes

Th. Given:

'AX=FFD0 BX=3534 CX=0000 DX=0180 SF=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72  
CS=1D72 IP=0111 HV UP EI NG NZ NA PO CY  
1D72:0111 EB0B JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug?

a) B bytes forward

b) 0008d U,V\C.5'M

cf Olllh

-Q119h-^r.

0113h

The number of nibbles in a double word are:

a) 2 4

S2006 Dahlquist

C

6^

32

Which of the following DOS commands would set a break point at memory location 010E7

b) G = Ee4

B = 010E

^ 100

G 0E01

"I

AX=FFD0 BX=3534 CX=0000 DX=0160 SP=FFEE BPJ=0000 SI=00004 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI NG NZ NA PO CY

1D72:010E 7D06 JHI^ 0118

What will the IP value be after a 'T' command is executed in DOS Debug?

a) 01OE

e) 0110

In x8(i iirchileeturc, ALU stands far which of (he following?

a) A Logic User

b) All Logic Unit

Address Logic Unit  
(f d)] Arithmetic Losi

\ Arithmetic Lining Unit

^ How many^tesjec,imajjv|ll (he program jump)for JNB instruction) given the following? AX=0000 BX=0000 CX=00d'0  
DX^OOOO SP=FFEE BP=O000 SI=0000 DI^OOOO

DS=1376 ES=1376 SS=1376 CS=1376 IP=0100. NV UP EI PL NZ NA PO NC

1376:0100 73E0 JNB OOE2

a) 64

b) 20

or

Wi I^Wv

-64

-a,

^~  
^

1-S b on7  
xi G

\

. Given:

AX=FFEO BX=3534 CX=0000 DX=0180 SP=FFEE BP=OGOO SI=O000 DI=O000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV UP EI PL NZ NR PO KC  
1D72:010D TEG 9 ■ JNLE 0118 ff

What is the decimal value of the signed number in the AX register?

- b) 0020
- c) 0032
- d) -0020

■ instruction MOV 11X, BAD is what addressing mode? -H7—Brect -^ a

- b) Register.

Sealed Index

)—RcgiSTETtml Ire

\flC Moore's law has accurately predicted the growth rate in the number of transistors per die for the last 25 years. What is that rate?

- a) Doubling every 18 years
- b) Doubling every 6 months Doubling every 36 months

JjJoul)lfng~every 18-24 months

Doubling every 30 - 41 weeks

How many address lines would be required to address 512MB directly?

512

- c) 20

yi. What is the hexadecimal encoding for adding BX with CX and storing the result in \$X? |

^ J

The binary number, 0111 1110, represents what values; in Hex, and as a BCD number?

\g) 7E.7-20

bj F5, invalids

fen 7E, 7 invalid

- e) invalid 7, E7

In using INT 10h to set the video mode to 640 x 200, what value must be in the AX register?

- a) 00h
- b) 02h
- c) 0000b
- d) 0E06h

1. Given

AX=FFF0 BX=3534 CX=0000 DX 0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY 1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

- a. -16
- 2. What are the contents of BX after this program has been run

a.

MOV AX, 11H MOV CX, [1101] MOV BX, 110H SUB AX, [BX]

- 3. What is 16.4375<sub>10</sub> in binary
- a. 010000.01110
- 4. Here is a short sequence of code: 7413A3EB CD16 7D21 3C04 EBF0 EB15. All of the instructions are
  - a. word long. The forth instruction operator is
  - a. JGE
- 5. Determine the contents of register BL after the following instructions have been executed:

- a.
6. Which of the following DOS Debug instructions would set a break point at memory location 010E
- a.
7. The “LOOPNZ” instruction is equivalent to which of the following instructions a. DEC CX, JNE
8. Given
- AX=FFE0 BC=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=00000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV UP EI PL NZ NA PO NC 1D72:010D 7D09 JGE 0116
- How many bytes will the processor jump if the condition for a jump were met?
- a. 9
9. Given
- AX=0353 BC=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=00000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY
- 1D72:0109 7D06 JNL 0118
- a. 010B
10. How many address lines would be required to address 64 MB directly?
- a. 512
11. How many bytes are there in this short sequence of code? B400CD16CD204
- a. 6
12. Given
- AX=2247 BC=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=00000 DI=0000
- DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 NV UP EI NG NZ NA PE NC 1D72:0106 EB0F JMP 0118
- What will the IP value be after a “t” command is executed in DOS Debug?
- A. 0118h
13. In the hello MASM lab in the original code, what is the address of the byte used to start the number in the sequence “Hello World”
- a. .data? or MyTitle
14. What is the hexadecimal encoding for adding BX with CX and storing the result in BX?
- a. 01CB
15. 1010 0110 in 2's complement equals \_\_\_\_ in base 10
- a. 166

16. Given:

13a7:0110 CD 20 32 20 54 48 69 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 CD 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0114, what is the size of the buffer in decimal? a.

17. A “NOP” instruction in a program will:

a. Align instruction addresses

18. If CX is 0002 what will CX be after a “LOOP” instruction?

a. 0001

19. In x86 architecture, ALU stands for which of the following?

a. Arithmetic Logic Unit

20. The Instruction MOV CX [DADD] is what addressing mode?

a. Register method

21. In adding 5+7 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

a. 0?

22. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?

a. 7DF2

23. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is that rate?

a. Doubling every 18-24 months

24. In x86 architect, BIU stands for which of the following

a. Bus interfering unit

25. F6 in 2’s complement equals \_\_\_\_\_ in base 10

a.

26. A microprocessor with a 32-bit address bus could access how much memory? a.

27. AND’ing 10H and 2FH will result in which of the following? a. 0

28. For the instructions sequence below, determine the contents of the register AL after this program

a.

29. The ASCII code for space, space, carriage return, line feed, end of string in decimal are

a.

30. What command in DEBUG would be used to execute interrupts?

- a. P
31. The number of nibbles in a double word are:
- a. 8
- 32.

Moore's Law - the number of transistors in a dense integrated circuit doubles approximately every two years.

Acronyms:

IDE – Integrated Design Environment

BIU – Bus Interface Unit

ALU – Arithmetic Logic Unit

Spring 2014 Exam questions:

1. How much global RAM does the Propeller microcontroller have?

32 Kb of RAM

2. How many lines would be required to address 128 MB directly?

27 lines... because 1 Mb can be written as approximately  $2^{20}$  (that's 20 address lines) and so with 128, can be written as  $2^7 + 2^{20} = 2^{27}$ .

3. If you want to use a INT software interrupt function to print a string out to the screen, what is the function code, start pointer, termination character, and interrupt you need to use?

Print out to the screen: int 21; termination character: int 20;

4. Moore's law has accurately predicted the growth rate in the number of transistors per die in the last 40 years. What is that rate?

Every 2 years

5. What command in DEBUG would be used to execute interrupts without stepping through the interrupt instructions?

6. In x86 architecture, ALU stands for which of the following?

Arithmetic Logic Unit

7. What is the hexadecimal encoding for adding BX with CX and storing the result in BX?

Add BX,CX [BX]

8. On the Arduino platform, what is the programming language used?

C/C++

9. Which of the following DOS Debug instructions would be used to change the AX register to 010C?

Mov ax, 010C

10. Program Listing: MOV BL, E2H: MOV CL, 06H: ROL BL, CL.

11. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

12. What high level language(s) is (are) the propeller programmed in?

Spin language or C

13. A microprocessor with a 26-bit address bus could access how much memory?

2<sup>26</sup>=67108864 or 64M

14. The instruction MOV CX, [DADD] is what addressing mode?

Direct

15. Given:

0B0E: 0200 57 65 6C 63 6F 60 65 20-74 6F 20 41 73 73 65 60

0B0E: 0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message?

16. How many cores does the propeller microcontroller have?

8 32-bit

17. In x86 architecture, BIU stands for which of the following?

Bus Interface Unit

18. In the PIC 18 with TRISD = 0b011110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

19. In the Propeller microcontroller, the term “Method” is(are) which of the following?

20. What is the number, 1010.01012 in decimal?

21. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as

Pulse-width modulation

22. No Q

23. AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA P0 CY 1D72:0111 EB08 JMP 0119

What will the IP value be after a “t” command is executed in the DOS Debug?

24. In the Propeller microcontroller, the command “dira[4:9] :=%111111” would cause the processor to do which of the following?

25. The instruction MOV CX, DADD is what addressing mode?

Register addressing

26. If CX is 0000 what will CX be after a “LOOP” x86 instruction?

Decrements cx and jumps the to a specific address

27. What are the contents of DX after this program has been run:

Memory Location    Contents

5513 D8

```
5512 D0
MOV DX, 11h
      5511 21
MOV CX, [5512]
      5510 00
```

MOV BX, 5511h

```
SUB DX, [BX]
AND BX, FFFF
```

28. Which command would you use to execute another core in the propeller microcontroller?

Cognew

29. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

.data? or Mytitle?

30. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 ST=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 0V UP EI PL NZ NA PO CY 1D72:0109 7D06 JGE 0118

What is the signed decimal value of the number in the AX register?

31. How many address lines would be required to address 64 MB directly?

I think it would be 512

32. In the Propeller microcontroller, the command “waintcnt[clkfreq’10 + cnt]” would cause the processor to do which of the following?

33. AX=2247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72  
IP=0106 NV UP EI NG NZ NA PE NC 1D72:0106 EB0F JMP 011F

What will the IP value be after a “t” command is executed in Debug?

34. The number of bits in a word are:

32 or 64 bits

35. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

36. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 IP=010F NV  
UP EI NG NZ NA PO NC 1D72:010F 7D18 JGE 0128

How many bytes in decimal will the processor jump if the conditions for a jump are met?

37. The number of nibbles in a word are:

8 nibbles

38. In the PIC18 with TRISD = 0b01000000, what is the configuration of the Port D?
39. AND'ing 1FH and 02H will result in which of the following?
40. With this a short sequence of code: 7413EBA3CD167D213C04EBF0EB15, All of the instructions are a word long. The third instruction operator is:
41. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
42. In the Propeller, how many values does a method return?
43. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?
44. Which of the following Debug instructions would be used to change the IP register to 0110?
45. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY 1D72:0109 7D06 JGE 0118

What will the IP value be after a “t” command is executed in DOS Debug?

46. The “LOOPNE” instruction is equivalent to which of the following instructions?

Should be LOOPNZ

47. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?
48. What command in DEBUG would be used to step through a program line by line?

The “t” command

49. Which of the following is a valid x86 command for multiplying a number?

Mul command

50. What is the advantage of Assembly Language over C Language?

You get to write code in one-line instruction as opposed to several lines used in C language

51. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?
52. No question
53. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72 NV  
UP EI NG NZ NA PO NC 1D72:010D EB07 JMP 0114

How many bytes will the processor jump if the conditions for a jump were met?

54. AND'ing 10Hex and 2FHex will result in which of the following Hex numbers?
55. In the PIC18 with TRISD = 0b11111111, what is the configuration of the Port D?
56. AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
57. DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC 1D72:010D 7DF6 JNL 0116

How many bytes in decimal will the processor jump if the conditions for a jump were met?

58. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:
59. Which of the following Debug commands would set a break point at memory location 010C?

The DR command

60. In the PIC18 with TRISD = 0b10000000, what is the configuration of the Port D?
  1. What is the hexadecimal encoding for “JNL” for a jump back 8 bytes?  
A: the signal flag and the overflow flag would be the same
  2. In x86 architecture, ALU stand for which of the following?  
A: Arithmetic Logic unit
  3. The number of nibbles in a word are:  
A: 4
  4. The instructions MOV BX [2BAD] is what addressing mode?  
a. Direct
  5. What is the hexadecimal encoding for adding DX with BX and storing the result in BX  
a. Add dx, bx [bx]
  6. A microprocessor with a 32-bit address bus could access how much memory  
a. 4 GB
  7. You add 7+6 through a 4bit integer unit. The state of the OF and the CF flags after the add will be  
a. OF=0, CF=0
  8. Which of the following Dos Debug instructions would be used change the IP register to 100  
a. RIP
  9. What are the contexts of CX after this program has been run: MOV DX, 11H  
MOV CX,[5512] MOV BX, 5511H SUB DX, [BX]

- a. D800H

10. Given

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

- a. -48
11. How many bytes are there in this short sequence if cod B815B400CD16CD20
- a. 8
12. In using INT 10h to set the video mode to 640 x 350, what value must be in the AH register
- a. 00h
13. Moore's law has accurately predicted the growth rate in the number of transistors per die for the last 25 years. What is the rate
- a. Doubling every 18-24 months
14. Int 21h, function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. they are:
- a. DS=SEG Hello\_msg, DX= OFFSET Hello\_msg, Hello\_msg terminated with 24h
15. Given

13A7:0110 CD 20 32 20 54 68 69 73-20 73 20 41 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in decimal a. 50

16. Here is a short sequence of code: 7413CD16EB157D213C04EBF0A3C6, all of the instruction are two bytes long. The third operator is:

- a. JPM
17. What is the hexadecimal encoding for loading DX with a word (value) from memory location 0820h?
- a. 8B162008
18. Given

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI= 0000 DI=0000

DS= 1D72 ES=1D72 SS=1D72 CS=1D72 EP=010D OV UP EI PL NZ NA PO NC

1D72:010D 7D09 JDE 0116

How many bytes will the processor jump if the condition for a jump were met/

- a. 9
19. F6 in 2's complement equals \_\_\_\_ in base 10 a. -10
20. Given

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI= 0000 DI=0000 DS= 1D72 ES=1D72 SS=1D72 CS=1D72 EP=010D OV UP EI NG NZ NA PO NC

1D72:010D 7D09 JNLE 0116

What will the IP value be after a "t" command is executed in DOS Debug?

a.

21. Determine the contents of register BH after the following instruction have executed:

a. 70h MOV[0202] AX

22. Given

AX=2247 BX=0000 CX=000 DX=0000 SP=FFEE BP=0000 SI= 0000 DI=0000

DS= 1D72 ES=1D72 SS=1D72 CS=1D72 EP=0106 NV UP EI NG NZ NA PE NC

1D72:0106 EB0F JMP 0118

23. Which of the following DOS Debug instruction would set a break point at memory location 010E a. G = 100 10E

24. Given

0B0E:0200 57 65 6C 63 6F 6D 65 20 -74 6F 20 41 73 73 65 6D 0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at the memory location 0200, what is the message a. Welcome to Assembly language

25. Given

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI= 0000 DI=0000 DS= 1D72 ES=1D72 SS=1D72 CS=1D72 EP=0104 NV UP EI NG NZ NA PE NC

1D72:0104 7002 J0 0118

What will the IP value be after a “t” command is executed in DOS Debug?

a. 0106h

EEE 174 Midterm Study Guide

S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:

• DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36

2. Moore’s Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?

• Doubling every 18 – 24 months

3. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What will the IP value be after a “t” command is executed in DOS Debug?

• 010B

4. A “NOP” instruction will:

• Perform a No OPeration

5. Given:

AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0106 NV UP EI NG NZ NA PE NC

1D72:0106 EB0F JMP 0118

What will the IP value be after a “t” command is executed in DOS Debug?

- 0118h

6. In x86 architecture, BIU stands for which of the following?

- Bus Interface Unit

7. Determine the contents of register BL after the following instructions have been executed:

- 2EH

8. 8. The number of bytes in a double word are:

- 4

9. In x86 architecture, ALU stands for which of the following?

- Arithmetic Logic Unit

10. The “LOOPNE” instruction is equivalent to which of the following instructions?

- DEC CX, JNE

11. The instruction MOV CX, DADD is what addressing mode?

- Immediate

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

- INT

13. Which of the following DOS Debug instructions would set a break point at memory location 010C?

### Program Listing

MOV BL, E2H

MOV CL, 100b

- G = 100 10C
14. How many bytes are there in this short sequence of code? B400CD164CCD21
- 7
15. How many address lines would be required to address 128 MB directly?
- 27
16. What are the contents of CX after this program has been run:
- D800h
17. 1010 0110 in 2's complement equals\_\_\_\_ in base 10.
- -90

### Memory Location    Contents

1103      24

1102      D8

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

- 01D8
19. What is 11.437510 in binary?
- 001011.01110
20. If CX is 0000 what will CX be after a “LOOP” instruction?
- FFFF
21. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?
- 7DFA
22. Given:
- AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010E OV UP EI PL NZ NA PO NC
- 1D72:010F 7D08 JGE 0115
- How many bytes will the processor jump if the conditions for a jump were met?
- 8
23. What command in DEBUG would be used to execute interrupts?
- P
24. What is the advantage of C Language over Assembly Language?
- C is transportable to other microprocessor architectures
25. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:
- OF = 1, CF = 0
26. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
- 36
27. Given:

AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI PL NZ NA PO NC

1D72:010D 7DF6 JNL 0116

- -10
28. AND’ing 1FH and 20H will result in which of the following?
- 0
29. Given:

13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

- 72
30. A microprocessor with a 31-bit address bus could access how much memory?
- 2 GB
31. For the instruction sequence below, determine the contents of the register AL after this program is executed:
- 51H

32. Given:

AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

- -256
33. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- 32, 32, 13, 10, 36

Spring 2012

34. What is the advantage of assembly language over C language?

- The assembler creates much faster executable code
35. How many lines would be required to address 128 mb directly?
- 27

36. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence "Hello World 0"?

- 020E
37. Given

W e l c o m e \_ t o \_ A s s e m 0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D 0B0E:0210 62 6C 79 20  
4C 61 6E 67-75 61 67 65 00 00 00 00 b l y \_ L a n g u a g e

An Ascii message begins at memory location 0200, what is the message?

- Welcome to Assembly Language

## Program Listing

MOV AL, 73h

ADD AL, 78h

DAA

38. Which of the following DOS Debug instructions would be used to change the IP register to 0110?

- RIP

39. Moore's law has accurately predicted the growth rate in the number of transistors per ide for the last 40 years. What is the rate?

- Doubling every 18 – 24 months

40. Which of the following DOS debug instructions would set a break point at memory location 010C?

- G = 1000 10C

41. AND'ing 1FH and 02H will result in which of the following?

- 02

42. If CX is 0000 what will CX be after a "LOOP" instruction?

- FFFF

43. The number of bits in a word are:

- 16

44. In x86 architecture, ALU stands for which of the following?

- Arithmetic Logical Unit

45. Given:

What will the IP value be after a "t" command is executed in DOS DEBUG?

- 010B

46. What is the number 1010.01012 in decimal?

- 10.31
47. How many cores does the propeller microcontroller have?
- 8
48. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?
- 7DF2
49. What command in debug would be used to step through a program line by line?
- T
50. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:
- INT
51. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
- 24
52. Given:

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- -10
53. Determine the contents of the register BL after the following instructions have been executed:
- E2H
54. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- 20, 20, 0D, 0A, 24
55. AND’ing 10H and 2FH will result in which of the following?
- 0
56. A “NOP” instruction in a program will
- Perform a No Operation
57. Given:

What is the signed decimal value of the number in the AX register?

- -16
58. How many address lines would be required to address 64 MB directly?

- 26
59. What command in debug would be used to execute interrupts?
- P
60. What high level language is the propeller programmed in?
- Spin
61. Which of the following DOS debug instructions would be used to change the IP register to 010C?
- RIP
62. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:
- Pulse width modulation
63. Which command would you use to execute another core in the propeller microcontroller?
- Cognew
64. Given:

What will the IP value be after the “t” command is executed in DOS debug?

- 0119h
65. The instruction MOV CD, DADD is what addressing mode?
- Immediate
66. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?
- F0
67. How many byte in decimal will the processor jump if the conditions for a jump are met?
- 24
68. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?
- Sets the Propeller pins P4 through P9 as output pins
69. What are the contents of DX after this program has been run:
- FFF0h.
70. The number of nibbles in a word are:
- 4
71. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?

- Create 10 second delay
72. The “LOOPNE” instruction is equivalent to which of the following instructions?
- DEC CX JNE
73. Given

How many bytes will the processor jump if the conditions for a jump were met?

- 7
74. On the Arduino platform, what is the programming language used?
- C
75. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?
- A12008
76. Which of the following would be used to set the TRISA register to control the direction of PIC18 Port to input?
- 1
77. The acronym ADC in microcontrollers stands for which of the following?
- Analog to Digital converter
78. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:
- OF = 1, CF = 0
79. What is the advantage of Assembly Language over C Language?
- The Assembler creates much faster executable code
80. How many address lines would be required to address 128 MB directly?
- 27
81. In the x86 lab part Hello MASM program in the original code, what is the address of the byte used to start number in the sequence “Hello World 0”?

- 020E
82. Given:  
0B0E : 0200 57 65 6C 63 F 6D 65 20-74 6F 20 41 73 73 65 6D

0B0E: 021062 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message?

- Welcome to Assembly Language
83. Which of the following DOS Debug instructions would be used to change the IP register to 0110?
- RIP
84. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?
- Doubling every 18-24 months
85. Which of the following DOS Debug instructions would set a break point at memory location 010C?

- G = 100 10C
86. AND'ing 1FH and 02H will result in which of the following?
- 02
87. If CX is 0000 what will CX be after a “LOOP” instruction?
- FFFF
88. The number of bits in a word are:
- 16
89. In x86 architecture, ALU stands for which of the following?
- Arithmetic Logic Unit
90. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 0V UP EI PL NZ NA PO CY

1D72:0109 7D06 JGE 0118

What will the IP value be after a “t” command is executed in DOS Debug?

- 010B
91. What is the number, 1010.01012 in decimal?
- 10.31
92. How many cores does the propeller microcontroller have?
- 8
93. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?
- 7DF2
94. What command in DEBUG would be used to step through a program line by line?
- T
95. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:

- INT
96. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
- 24
97. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI PL NZ NA PO NC

1D72:010D 7DF6 JNI 0116

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- -10
98. Determine the contents of register BL after the following instructions have been executed:
- E2H
99. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

- 32, 32, 13, 10, 36
100. AND'ing 10H and 2FH will result in which of the following?
- 0
101. A “NOP” instruction in a program will:
- Perform a No Operation
102. Given:  
AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72

ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY ID72:0109 7D06 JGE 0118

What is the signed decimal value of the number in the AX register?

- -16
103. How many address lines would be required to address 64 MB directly?
- 26
104. What command in DEBUG would be used to execute interrupts?
- P
105. What high level language is the propeller programmed in?
- Spin
106. Which of the following DOS Debug Instructions would be used to change the IP register to 010C?
- RIP
107. The acronym PWM used in the Parallax Propeller and MicroChip PIC18, is defined as:
- Pulse Width Modulation
108. Which command would you use to execute another core in the propeller microcontroller?
- Cognew
109. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0111 NV UP EI NG NZ NA PO CY

1D72:0111 ED08 JMP 0119

What will the IP value be after a “t” command is executed in DOS Debug?

- 0119h
2. The instruction MOV CX, DADD is what addressing mode?
- Immediate
110. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?
- A0
111. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010F NV UP EI NG NZ NA PO NC

1D72:010F 7D18 JGE 0128

How many bytes in decimal will the processor jump if the conditions for a jump are met?

- 24
112. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?
- Sets the Propeller pins P4 through P9 as output pins
113. What are the contexts of DX after this program has been run:

• FFF0h

114. The number of nibbles in a word are:
- 4

115. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?

- Create 10 second delay
116. The “LOOPNE” instruction is equivalent to which of the following instructions?
- DEC CX, JNE

117. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI NG NZ NA PO NC

1D72:010D EB07 JMP 0114

How many bytes will the processor jump if the conditions for a jump were met?

- 7
118. On the Arduino platform, what is the programming language used?
- C
119. What is the hexadecimal encoding for loading AX with a word (value) from the memory location 0820h?
- A12008
120. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?
- 1
121. The acronym ADC in microcontrollers stands for which of the following?

- Analog to Digital Converter
122. In adding 5+5 through a 4 bit integer unit. The state of the OF and the CF flags after the add instruction would be:
- OF=1, CF=0

123. Given:  
0B0E:020057 65 6C 63 6F 6D 65 20-74 6F 20 74 68 65 20 66

0B0E:0210        69 72 73 74 20 64 61 79-20 6F 66 20 74 68 65 20  
 0B0E:0220        72 65 73 74 20 6F 66 20-79 6F 75 72 20 6C 69 66  
 0B0E:0230        65 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

An ASCII message begins at memory location 0200, what is the message?

- Welcome to the first day of the rest of your life
124. The instruction MOV DX, BADD is what addressing mode?
- Immediate
125. Which of the following is the hexadecimal encoding for adding BX with CX and storing the result in CX?
- 03CB
126. What is the advantage of Assembly Language over C Language?
- The Assembler creates much faster executable code
127. What is 18.437510 in binary?
- 010010.01110
128. For the instruction sequence below, determine the contents of the register AL after this program is executed:
- 51H
129. In x86 architecture, ALU stands for which of the following?
- Arithmetic Logic Unit
130. A microprocessor with a 33-bit address bus could access how much memory?
- 8 GB
131. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?
- 7DF2
132. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
 IP=010F NV UP EI NG NZ NA PO NC

1D72:010F        7D18      JGE      0128

How many bytes in decimal will the processor jump if the conditions for a jump are met?

- 24
133. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is that rate?
- Doubling every 18-24 months

134. The number of bytes in a word are:
- 2
135. Determine the contents of register BL after the following instructions have been executed:
- 2EH
136. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?
- 7
137. What high level language is the propeller programmed in?
- Spin
138. In the Propeller microcontroller, the command “dira[4..9] := %000000” would cause the processor to do which of the following?
- Sets the Propeller pins P4 through P9 as input pins
139. Which command would you use to execute another core in the propeller microcontroller?
- Cognew
140. How many cores does the propeller microcontroller have?
- 8
141. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?
- Create 10 second delay
142. The acronym ADC in microcontrollers stands for which of the following?
- Analog to Digital Converter
143. The acronym PWM used in the Parallax Propeller and Microchip PIC18, is defined as:
- Pulse Width Modulation
144. How many bits does the PIC18 microcontroller use in the PICkit 3 Debug Express have?
- 8
145. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?
- 1
146. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?
- Bit 7 of port D is set to output
147. In the PIC 18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?
- 0A
148. On the Arduino platform, what is the programming language used?
- C
149. Given:  
AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY

1D72:0111 EB08 JMP 0119

What will the IP value be after a “t” command is executed in DOS Debug?

- 0119h
150. How many bytes are there in this short sequence of code? B4 00 CD 16 4C CD 21 CD 20  
• 9
151. In x86 architecture, BIU stands for which of the following?  
• Bus Interface Unit
152. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The fifth instruction operator is:  
• CMP
153. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:  
• 32, 32, 13, 10, 36
154. A “NOP” instruction in a program will:  
• Perform a No Operation
155. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

- -16
156. Which of the following DOS Debug instructions would set a break point at memory location 010C?  
• G = 100 10C
157. In adding 5+7 through a 4 bit integer unit, the state of the OF and CF flags after the add instruction would be:  
• OF = 1, CF = 0
158. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV UP EI NG ZR NA PO NC

1D72:010D 7509 JNZ 0116

What will the IP value be after a “t” command is executed in DOS Debug?

- 010Fh
159. What are the contents of DX after this program has been run:  
• FFF0h
160. Which of the following DOS Debug instructions would be used to change the IP register to 010C?  
• RIP
161. What is the number, 1010.01012 in decimal?  
• 10.31
162. What command in DEBUG would be used to step through a program line by line?

- T
163. AND'ing 1FH and 02H will result in which of the following?
- 02
164. How many address lines would be required to address 64 MB directly?
- 26
- 165.
- S2010 MT2
166. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?
- 7
167. If CX is 0003, what will CX be after a “LOOP” instruction?
- 0002
168. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?
- A 5 second delay
  - 169. What is the number 1011.01012 in decimal?
  - 11.31
170. This section of memory represents a stack. What type of program is this? BEEF : FFD0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF
- COM program
171. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?
- BDCA
172. Determine the contents of register BL after the following instruction have been executed:
- 2EH
173. What Hex values must be sent to address the key pad rows on the PPE board?
- 1, 2, 4, 8
174. What is the number 32.437510 in binary?
- 100000.01110
175. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
- 24
176. The acronym PWM used for motor control, is defined as which of the following?

- Pulse Width Modulation
177. In the PIC18 with TRISD = 0b0111111, what is the configuration of the Port D?
- Bit 7 of port D is set to output
178. Given the short code, what is the value in AX after the program is run?
- 0500

#### Program Listing

MOV BL, 2E

MOV CL, 10

ROL BL, CL

#### Program Listing

MOV BX, 0500

Push BX

MOV AX, 0100

POP AX

179. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or

not?

- ZF

180. How many nibbles are in double precision IEEE floating point format number?

- 16

181. In the Hello MASM lab in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

- 020E

182. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC

1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD

- EXE

183. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?

- F00D

184. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D

and shown on the LEDs?

- A0

185. What is the numeric sequence of the key pad columns on the PPE board?

- 37, 2F, 1F

186. What is -130 decimal in 2’s complement (8bits)?

- 01111110

187. Which of the following is a valid x86 command for multiplying a number?

- MUL BX

188. The number of bits in single precision IEEE floating point format are:

- 32

189. You are trying to rebuild a HELLO project program in MASM and you get the following error:

“LINK : warning L4021: no stack segment”.

What would be the reason for such an error?

- No project template for COM was selected

190. A “pull down” resistor is used in digital circuits to do what?

- To keep the signal line “tied” low until the line is active (goes high)

191. A “POP” instruction:

- increments the SP

192. AND'ing 10H and 2FH will result in which of the following?
- 0
193. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?
- Sets the Propeller pins P4 through P9 as output pins
194. What commands in MASM-CodeView would be used to step through a program line by line?
- T (F8)
195. If the SP is F00F, what will the SP value be after a “POP SP” instruction?
- F011
196. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?
- 0
197. In the Propeller microcontroller, the term “Method” is(are) which of the following?
- A block of executable commands that has variables, can receive parameters, and returns a value.
198. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?
- MOV CX, 30H      or      MOV CX, 48
199. A “pull up” resistor is used in digital circuits to do what?
- To keep the signal “tied” high until the line is active (goes low)
  - 2. 1010 0110 in 2's complement equals in base 10 -90
  - 3.
  - 4. A microprocessor with a 32-bit address bus could access how much memory 4GB
  - 5.
  - 6. A6 in 2's complement equals in base 10 -90
  - 7.
  - 8. A “POP” instruction: Increments the SP
  - 9. A “PUSH” instruction decrements the SP
  - 10. A “NOP” instruction in a program will Perform a No Operation
  - 11. AND'ing 1FH and 02H will result in which of the following 02
  - 12. AND'ing 10H and 2FH will result in which of the following 0
  - 13. A “pull down” resistor is used in digital circuits to do what? To keep the signal line “tied” low until the line is active (goes high)
- 14.
15. A “pull up” resistor is used in digital circuits to do what To keep singal “tied” high until the line is active (goes low)
16. Determine the contents of register BL after the following instructions have been executed: E2H
17. MOV BL, E2H
18. MOV CL, 1000b
19. ROL BL, CL ; rotate BL by 8 to left = same value
- 20.
21. For the instruction sequence below, determine the contents of the registers AL after this program

22. MOV AL, 73h ;0111 0011  
 23. ADD AL, 78h ;0111 1000  
 24. DAA  
 25. if low nibble of AL > 9 or AF = 1 then:  
 26.     AL = AL + 6  
 27.     AF = 1  
 28.     if AL > 9Fh or CF = 1 then:  
 29.         AL = AL + 60h  
 30.         CF = 1
31.  
 32. For the instruction sequence below, determine the contents of the register AL after this program is executed 51H  
**MOV AL, 83h**  
**ADD, AL, 68h**  
 33.     DAA
34.  
 35. Given the short code, what is the value in AX after the program is run 0100 Mov BX, 0100  
**PUSH BX** **MOV AX, 0500**  
**POP AX**
36.  
 37. Given the short code, what is the value in AX after the program is run 0001  
**PUSH BX**  
 39. **MOV AX, 0500**  
 40. **POP AX**
41.  
 42. GIVEN: IP = 0106 Flags: NV UP EI NG NZ NA PE NC Instruction: JMP 011F  
 43. What will the IP value be after “t” command is executed in DOS Debug? 011FH (Unconditional  
 Jump)  
 44.
45. GIVEN: IP = 0109 Flags: OV UP EI PL NZ NA PO CY Instruction: JGE 0118  
 46. What will the IP value be after a “t” command is executed in DOS Debug 010B  
 47. (0109 + 0010 add two bytes)
48.  
 49. GIVEN: IP=FFE0 Flags: OV UP EI PL NZ NA PO NC Instruction: JGE 0116: ID72:010D 7D09  
 50. How many bytes will the processor jump if the condition for a jump were met? 9  
 51.
52.  
 53. GIVEN: AX= FFF0 IP = 0109 FALGS: OV UP EI PL NZ NA PO CY ID72: 010F 7D18 Instruction: JGE 0118. What is  
 the signed decimal value of the number in the AX register? -16
54. Covert the number 7D18 into decimal.  
 55.
56. GIVEN: IP= 010F Flags: NV UP EI NG NZ NA PO NC Instruction: JNL 0115.  
 57. How many bytes in decimal will the processor jump if the conditions for a jump were met 24  
 58.

59. GIVEN: 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D  
60. 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00  
61. An ASCII message begins at memory location 0200, what is the message? Welcome to Assembly Language  
62.
63. Here is a short sequence of code: 7413 EBA3 CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The third instruction operator is INT  
64.
65. Here is a short sequence of code: 7413 A3EB CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The fourth instruction operator is JGE  
66.
67. Here is a short sequence of code: B400 CD16 3C4A 7404 BC6A 7513. All of the instructions are two bytes long. The sixth instruction operator is JNZ  
68.  
69.  
70. How many cores does the propeller microcontroller have 8  
71.  
72. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 15 4  
73. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 63 6  
74. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 127 7  
75. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 255 8  
76. How many bytes are there in this short sequence of code B4 00 CD 16 4C CD 20 7  
77. How many nibbles are there in this short sequence of code B4 00 CD 16 3C 4A 74 04 3C 6A 75 13 24  
78. How many bytes are in double precision IEEE floating point format numbers 8  
79. How many nibbles are in double precision IEEE floating point format numbers 16
80. How many address lines would be required to address 128 MB directly 27 ( $128 \times 1048576 = 134217728$  and  $2^{27} = 134217728$ )
81. How many address lines would be required to address 64 MB directly 26 ( $64 \times 1048576 = 67108864$  and  $2^{26} = 67108864$ )
82. If CX is 0000, what will CX be after a “LOOP” instruction FFFF
83. If CX is 0003, what will CX be after a “LOOPNZ” instruction 0002
84. If the SP is F00F, what is the SP value after a “PUSH CX” instruction F00D  
85. If the SP is F00F, what is the SP value after a “POP CX” instruction F011  
86. If the SP is F00F, what is the SP value after a “POP SP” instruction F011  
87. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be OF = 1, CF = 0
88. In x86 architecture, BIU stands for which of the following Bus Interface Unit  
89. In x86 architecture, ALU stands for which of the following Arithmetic Logic Unit  
90. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”? 020E  
91. In MASM, with a “MOV CX, 24h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal 36
92. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, how many times will the program loop in decimal 24
93. In MASM, with a “MOV CX, 12h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal 18
94. In the Hello MASM lab in the original code, what is the address of the string to start the message “Hello World 0” 0200
95. In the PIC18 with TRISD = 0b10000000, what is the configuration of the Port D Bit 7 of port D is set to input

96. In the PIC18 with TRISD = 0b01111111 and LATD = 0xAA, what value will be on Port D and shown on the LEDS Bit 7 of port D is set to output (because the first bit is zero = output)
97. In the PIC18 with TRISD = 0b00001111, what is the configuration of the Port D A0 (First 4 are outputs and last four are inputs)
98. In the PIC18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDS 0A
99. In the Propeller microcontroller, the command “dira[9..4] := %000000” would cause the processor to do which of the following Sets the propeller pin P4 through P9 as output pins
100. In the Propeller microcontroller, the command “dira[9..4] := %111111” would cause the processor to do which of the following Sets the propeller pin P4 through P9 as output pins
101. In the propeller microcontroller, the command “waitcnt(clkfreq\*3 + cnt)” would cause the processor to do which of the following A 3 second delay
102. In the propeller microcontroller, the command “waitcnt(clkfreq\*2 + cnt)” would cause the processor to do which of the following A 2 second delay
103. In the Propeller microcontroller, the term “Method” is (are) which of the following A block of executable commands that has variables, can receive parameters, and returns a value.
- 115.
104. Int 10h uses what function code to write a character to the screen and advance the cursor by one character position 0Eh
105. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string:
106. DS=SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h.
- 107.
108. Ladder Logic is used in? PLCs
109. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate? Doubling every 18-24 months
110. On the Arduino platform what is the program language used C
- 111.
112. On the PPE board, what numbers(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh 0
113. On the PPE board, what numbers(s) on the key pad is(are) pressed for an output port value of 04h and an input port value of 2Fh 8
114. The “LOOPNZ” instruction is equivalent to which of the following instructions DEC CX, JNE
116. The acronym PWM used for motor control, is defined as which of the following Pulse Width Modulation
117. The acronym PLC, is defined as which of the following? Programmable Logic Controller
- 118.
119. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are 32, 32, 13, 10, 36
120. The ASCII codes for space, space, carriage return, line feed, end of string in hexadecimal are: 20, 20, 0D, 0A, 24

121. The binary number, 1011 0101, represents what values as a unsigned binary, 8 bit signed binary, odd parity ASCII, and BCD number (in that order) 181, -76, 5, invalid

122. The instruction MOV CX, DADD is what addressing mode Immediate

123. The instruction MOV CX, [DADD] is what addressing mode Direct

124. The number of bits in single precision IEEE floating point format are 32

125.

126.

127.

128. This section of memory represents a stack. What type of program is this EXE

PROGRAM BEEF:00D0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 BEEF:00E0 00 01 02 03 04 05 06 07-08 09 0A 0B  
0C 0D 0E 0F BEEF:00F0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

129.

130. BEEF:0FD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 BEEF:0FE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C  
0D 0E 0F BEEF:0FF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

131.

132. This section of memory represents a stack. What type of program is this COM

PROGRAM BEEF:FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 BEEF:FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B  
0C 0D 0E 0F BEEF:FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

133.

134.

135. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal MOV CX, 48

136. Which command would you use to execute another core in the propeller microcontroller Cognew

137. What command in DEBUG would be used to change the code segment RCS

138. What command in DEBUG would be used to change the IP value RIP

139. What command in DEBUG would be used to execute interrupts P

140. What command in MASM-CodeView would be used to step through a program line by line T(F8)

141. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not  
ZF

142. What flag(s) does the “LOOPNE” instruction look at to determine whether to loop or not  
ZF

143. What Hex values must be sent to address the key pad rows on the PPE board 1,2,4,8

144. What is the advantage of C Language over Assembly Language C is transportable to other microprocessor architectures

145. What is the hexadecimal encoding for “JGE” for a jump back 10 bytes 7DF4 (10 bytes

+ 2 )

146. What is the hexadecimal encoding for adding AX with BX and storing the result in AX  
01D8

147. ADD AX, BX 000 00W 11 reg1 reg2

148.

149. What is the hexadecimal encoding for adding BX with CX and storing the result in BX

01CB

- 150.
151. What is the hexadecimal encoding for adding BX with DX and storing the result in BX  
01D3
- 152.
153. What is -130 decimal in 2's complement (8bits) 01111110
154. What is -32.75 in a base two number system -100000.110000
155. What is 14.4375 in binary 001110.0110
156. What is 16.4375 in binary 010000.01110
157. What is the binary value of decimal 12.875 1100.1110
- 158.
159. What is 16.4375 in binary 010000.01110
- 160.
161. What is number, 1011.0101 (2) in decimal? 11.31
- 162.
163. What is the numeric sequence of the key pad columns on the PPE board 37,2F,1F
164. What is the decimal value of C5 5A 57 00 in IEEE single precision FP format - 3493.4375
165. What of the following instruction would be used to set the LED to light on the Arduino platform digitalWrite(ledPin, HIGH);
- 166.
167. What type of program is this EXE  
IP = 0115, 1376:0115 0100 ADD [BX+SI], AL DS:0000=CD
- 168.
169. What type of program is this COM  
IP = 0100, 1376:0100 0100 ADD [BX+SI], AL DS:0000=CD
- 170.
171. Which of the following DOS Debug instructions would set a break point at memory location 010C G = 100 10C
- 172.
173. Which of the following would be used to set the TRISA register to control the direction of PIC18 port to input 1 and for output its 0
- 174.
175. Which of the following DOS Debug instructions would be used to change the IP register to 110 RAX = 0110
- 176.
177. Which of the following will cause a program with a LOOP instruction to loop 48 times (decimal) CX=30h
- 178.
179. Which of the following is a valid x86 command for multiplying a number MUL BX
180. Which of the following is not a valid command for a number into a register in MASM
- MOV AX, BADH
181. With a POP BX instruction, what will be order off the accumulator, base, count, and data registers restored from the stack BX
182. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack BDCA
183. You are trying to rebuild a HELLO program project in MASM and you get the following error: "ERROR 4 line 1". What is the cause of the error? Not known—this error by itself isn't a problem, press the enter key to clear the error.
- 184.
185. You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK : warring L4021: no stack segment". What would be the reason for the such an error

No project template for COM was selected.

186.

187. You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK : fatal error L1089: HELLO.lrf: cannot open response file". What would be the reason for the such an error? No source file is identified(no .asm file)

188.

189.

190.

191.

192.

193. PPE Row Column Scan decoding

D7 D6 D5 D4 D3 D2 D1 D0

S7 S6 S5 S4 S3 S2

^ ^ ^

3 2 1

0 0 0 0 1 0 0 0 = 08h

0 0 0 1 0 0 0 0 = 10h

0 0 1 0 0 0 0 0 = 20h

0 0 1 1 1 1 1 1 = 3Fh -> Nothing pressed

0 0 1 1 0 1 1 1 = 37h -> Number 1 pressed

0 0 1 0 1 1 1 1 = 2Fh -> Number 2 pressed

0 0 0 1 1 1 1 1 = 1Fh -> Number 3 pressed

194.

S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:

DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36

2. Moore's Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?

Doubling every 18 – 24 months

3. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What will the IP value be after a "t" command is executed in DOS Debug?

010B

4. A "NOP" instruction will:

Perform a No OPeration

5. Given:

AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0106 NV UP EI NG NZ NA PE NC

1D72:0106 EB0F JMP 0118

What will the IP value be after a “t” command is executed in DOS Debug?

0118h

6. In x86 architecture, BIU stands for which of the following?

Bus Interface Unit

7. Determine the contents of register BL after the following instructions have been executed:

2EH

8. The number of bytes in a double word are:

4

9. In x86 architecture, ALU stands for which of the following?

Arithmetic Logic Unit

10. The “LOOPNE” instruction is equivalent to which of the following instructions?

DEC CX, JNE

11. The instruction MOV CX, DADD is what addressing mode?

Program Listing MOV BL, E2H MOV CL, 100b ROL BL, CL

Immediate

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

INT

13. Which of the following DOS Debug instructions would set a break point at memory location

010C?

G = 100 10C

14. How many bytes are there in this short sequence of code? B400CD164CCD21

7

15. How many address lines would be required to address 128 MB directly?

27

16. What are the contents of CX after this program has been run:

D800h

17. 1010 0110 in 2's complement equals\_\_\_\_ in base 10.

Memory Location

1103

1102

1101

1100

Contents

24

D8

00

21

-90

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

01D8

19. What is 11.437510 in binary?

001011.01110

20. If CX is 0000 what will CX be after a "LOOP" instruction?

FFFF

21. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?

7DFA

22. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010E OV UP EI PL NZ NA PO NC

1D72:010F 7D08 JGE 0115

How many bytes will the processor jump if the conditions for a jump were met?

8

23. What command in DEBUG would be used to execute interrupts?

P

24. What is the advantage of C Language over Assembly Language?

C is transportable to other microprocessor architectures

25. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

OF = 1, CF = 0

26. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

36

27. Given:

AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI PL NZ NA PO NC 1D72:010D 7DF6 JNL 0116

-10

28. AND’ing 1FH and 20H will result in which of the following?

0

29. Given:

13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

72

30. A microprocessor with a 31-bit address bus could access how much memory?

2 GB

31. For the instruction sequence below, determine the contents of the register AL after this program is executed:

51H

32. Given:

AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

-256

33. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

32, 32, 13, 10, 36

Program Listing MOV AL, 73h ADD AL, 78h DAA

S2010 MT2

1. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?

7

2. If CX is 0003, what will CX be after a “LOOP” instruction?

0002

3. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?

A 5 second delay

4. What is the number 1011.01012 in decimal?

11.31

5. This section of memory represents a stack. What type of program is this? BEEF : FFD0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

COM program

6. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

BDCA

7. Determine the contents of register BL after the following instruction have been executed:

2EH

8. What Hex values must be sent to address the key pad rows on the PPE board?

1, 2, 4, 8

9. What is the number 32.437510 in binary?

Program Listing MOV BL, 2E MOV CL, 10

ROL BL, CL

100000.01110

10. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

24

11. The acronym PWM used for motor control, is defined as which of the following?

Pulse Width Modulation

12. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

Bit 7 of port D is set to output

13. Given the short code, what is the value in AX after the program is run?

0500

14. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?

ZF

15. How many nibbles are in double precision IEEE floating point format number?

Program Listing

MOV BX, 0500

Push BX

MOV AX, 0100

POP AX

16

16. In the Hello MASM lab in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

020E

17. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1476 ES=1576 SS=1376 CS=1D72  
IP=0015 NV UP EI PL NZ NA PO NC

1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD

EXE

18. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?

F00D

19. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

A0

20. What is the numeric sequence of the key pad columns on the PPE board?

37, 2F, 1F

21. What is -130 decimal in 2’s complement (8bits)?

01111110

22. Which of the following is a valid x86 command for multiplying a number?

MUL BX

23. The number of bits in single precision IEEE floating point format are:

32

24. You are trying to rebuild a HELLO project program in MASM and you get the following error:  
“LINK : warning L4021: no stack segment”.

What would be the reason for such an error?

No project template for COM was selected

25. A “pull down” resistor is used in digital circuits to do what?

To keep the signal line “tied” low until the line is active (goes high)

26. A “POP” instruction: increments the SP

27. AND’ing 10H and 2FH will result in which of the following?

0

28. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

Sets the Propeller pins P4 through P9 as output pins

29. What commands in MASM-CodeView would be used to step through a program line by line?

T (F8)

30. If the SP is F00F, what will the SP value be after a “POP SP” instruction?

F011

31. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?

0

32. In the Propeller microcontroller, the term “Method” is(are) which of the following?

A block of executable commands that has variables, can receive parameters, and returns a value.

33. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?

MOV CX, 30H or MOV CX, 48

34. A “pull up” resistor is used in digital circuits to do what?

To keep the signal “tied” high until the line is active (goes low)

EEE-174 MT1

1. What is the hexadecimal encoding for "JNL" for a jump back 8 bytes? answer

2. In X86 architecture, ALU stands for which of the following? Arithmetic Logic Unit

3. The number of nibbles in a word are:

$16/4 = 4$

4. The instruction MOV BX, [2BAD] is what addressing mode? Direct

5. What is the hexadecimal encoding for adding DX with BX and storing the result in BX?  
?????

6. A microprocessor with a 32-bit address bus could access how much memory?  $2^4=16MB$ ,  $2^8=256KB$ ,  $2^{32}=4GB$

7. You add 7+6 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:

0111 1 OF=1, the sign bit has changed  
+0110 CF=1, there is a carryon of bit 7  
0|1101

8. Which of the following DOS Debug instructions would be used change the IP register to 100?

RIP

9. What are the contents of CX after this program has been run:

MOV DX, 11h  
MOV CX, [5512]

MOV BX, 5511h  
SUB DX, [BX]

D800h

Memory Contents

location

|      |    |
|------|----|
| 5514 | 24 |
| 5513 | D8 |
| 5512 | 00 |
| 5511 | 21 |

\*\*\*What are the contents of DX after this program has been run:

FFF0h

10. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO

CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

-48

11. How many bytes are there in this short sequence of code? B815B400CD16CD20 8

12. In using INT 10h to set the video code to 640x350, what value must be in the AH register?

AX

AH|AL

00

00h

13. Moore's law has accurately predicted the growth rate in the number of transistors per die for

the last 25 years. What is that rate? Doubling every 18-24 months

14. INT 21h, Function 09h requires three things set up before calling in order to correctly print a string,

Hello\_msg. They are:

DS=SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h

15. Given

13A7:0110 CD 20 32 20 54 68 69 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 75 65 72 6D 0D 24 D9  
13A7:0130 00 C6 00 00 00 00 00 00-00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in decimal? 50

16. Here is a short sequence of code: 7413CD16EB157D213C04EBF0A3C6. All of the instructions are two bytes long. The third instruction operator is: EB15

JMP

17. What is the hexadecimal encoding for loading DX with a word (value) from memory location 0820h?

8B16008

18. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV UP EI PL NZ NA PO NC

1D72: 010D 7D09 JGE 0116

How many bytes will the processor jump if the condition for a jump were met? 9

19. F6 in 2's complement equals \_\_\_\_\_ in base 10.

10

20. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV UP EI NG NZ NA PO NC 1D72: 010D 7D09 JNLE 0116

What will the IP value be after a "t" command is executed in DOS Debug?

??????

21. Determine the contents of register BH after the following instructions have executed: 70h MOV [0202], AX

22. Given:

AX=2247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0106 NV UP EI NG NZ NA PE NC 1D72: 0106 EB0F JMP 0118

What will be the IP value be after a "t" command is executed in DOS Debug? 0118h

23. Which of the following DOS Debug instructions would set a break point at memory location 010E?

G=100 10E

24. Given:

0B0E: 0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D 0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message?

Welcome to Assembly Language

25. Given:

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 IP=0104 NV  
UP EI NG NZ NA PE NC 1D72:0104 7002 JO 0118

What will the IP value be after a "t" command is executed in DOS Debug?  
0106h

26. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72            ES=1D72 SS=1D72            CS=1D72 IP=010E OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118  
What will the IP value be after a "t" command is executed in DOS Debug?  
0110

27. The instruction MOV BX, 2BAD is what addressing mode? Immediate
28. What is the hexadecimal encoding for "JNL" for a jump back 10 bytes? 7DF4
29. How many address lines would be required to address 64MB directly?  $2^{26} = 67\text{mb}$ /
30. The number of nibbles in a double word are: 8
31. In using INT 10h to move the screen cursor to return on the same line, what value must be in the AX register?

0E0Dh

32. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV U EI PL NZ NA PO NC

1d72: 010D        7D09 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug?

010Fh

33. Which of the following DOS Debug instructions would be used change the AX register?  
RAX

34. How many Bytes are there in this short sequence of code? B815B400CD168A3CCD20 10

35. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0111 NV UP EI NG NZ NA PO CY 1D72:0111 EB08 JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug?

0119h

36. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010E OV UP EI NG NZ NA PO CY

1D72: 010E        7D06 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0118

37. How many bytes decimal will the program jump for JNB instruction, given the following? AX=0000 BX=0000  
CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1376 ES=1376 SS=1376 CS=1376 IP=0100 NV UP EI PL NZ NA PO NC  
1376:0100 73E0 JNB 00E2

-32

38. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D

BP=0000 SI=0000 DI=0000 OV UP EI PL NZ NA PO NC

1D72:010D 7F09 JNLE 0118

What is the decimal value of the signed number in the AX register?

-0032

39. How many address lines would be required to address 512 MB directly?  $29 / 2^9 = 29$ /
40. What is the hexadecimal encoding for adding BX with CX and storing the result in BX? 03D9
41. The binary number, 0111 1110, represents what values; in Hex, and as a BCD number? 7E, 7 invalid
42. In using INT 10h to set the video mode to 640 X 200, what value must be in the AX register?  
0006h
43. what is the hexadecimal encoding for loading AH with a word from memory location 0520h?

8A262005

44. What is the hexadecimal encoding for "JGE" for a jump back 10 bytes?  
7DF6

45. Given:

13A7:0110 CD 20 48 20 54 68 69 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
13A7:0120 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in bytes in decimal:  
72

46. Given:

AX=FF47 BX=0000CX=0000DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC 1D72:0104 7002 JNO 0118

What will the IP value be after a "t" command is executed in DOS Debug?  
0118

47. Determine the contents of register AH after the following instruction have executed:

MOV BX,BA70H  
MOV AX,47E7H  
MOV BL, 0FH 47H  
AND AL,BL  
MOV [0202],AX

48. In using INT 10h to move the screen cursor to return to the beginning of the line, what value must be in the AX register?

0E0Dh

49. Given:

1376:0110 48 61 76 65 20 61 20 67-72 65 61 74 20 53 70 72

1376:0120 69 6E 67 20 52 65 63 65-73 73 21 00 00 00 00 00

An ASCII message begins at memory location 0110, what is the message?

Have a great Spring Recess!

50. You add 1+8 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:  
OF=0, CF=0

Page 16

51. What is the advantage of Assembly Language over C Language? The Assembler creates much faster executable code

52. In The X86 lab 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence "Hello World 0"?

020E

53. Which of the following DOS Debug instructions would set a break point at memory location 010C?

G = 100 10C

54. If CX is 0000 what will CX be after a "LOOP" instruction?

FFFF

55. Given:

AX=0353 BX=0534 CX=0000 DX=0180

DS=1D72 IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 (OV=1,NV=1 ZR=1, NZ=0)

010B

56. How many core does the propeller microcontroller have? 8

57. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?

7DF2

58. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. instructions are a word long. The third instruction operator is:

INT

59. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?

24

60. The ASCII codes for space , space, carriage return, line feed, end of string in decimal are: 32,32,13,10,36

61. A "NOP" instruction in a program will:

62. How many address lines would be required to address 64MB directly? 26

63. What command in DEBUG would be used to execute interrupts?

64. What high level language is the propeller programmed in? Spin

65. Which of the following DOS Debug instruction would be used to change the IP register to 010C?

RIP

66. The acronym PWM used in the Parrallax Propeller and MicroChip PIC18, is defined as:

Pulse Width Modulation

67. Which command would you use to execute another core in the propeller microcontroller? Cognew

68. Given: IP=0111 NV UP EI NG NZ NA PO CY

1D72:0111 JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug?

0119h

69. The instruction MOV CX, DADD is what addressing mode? Immediate

70. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

A0

71. In the Propeller microcontroller, the command "dira[4..9] := %111111" would cause the processor to do with of the following?

Sets the Propeller pins P4 through P9 as output pins

72. In the Propeller microcontroller, the command "waitcnt(clkfreq\*10 + cnt)" would cause the processor to do with of the following?

Create 10 second delay

73. The "LOOPNE" instruction is equivalent to which of the following instructions?

DEC CX, JNE

74. On the Arduino platform, what is the programming language used?

C

EEE 174 Midterm Study Guide

S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:

a. DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36

2. Moore's Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years.

What is the rate?

a. Doubling every 18 – 24 months

3. Given:

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What will the IP value be after a "t" command is executed in DOS Debug?

a. 010B

4. A "NOP" instruction will:

a. Perform a No OPeration

5. Given:

AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0106 NV UP EI NG NZ NA PE NC

1D72:0106 EB0F JMP 0118

What will the IP value be after a "t" command is executed in DOS Debug?

a. 0118h

6. In x86 architecture, BIU stands for which of the following?
  - a. Bus Interface Unit
7. Determine the contents of register BL after the following instructions have been executed:
  - a. 2EH
8. The number of bytes in a double word are:
  - a. 4
9. In x86 architecture, ALU stands for which of the following?
  - a. Arithmetic Logic Unit
10. The “LOOPNE” instruction is equivalent to which of the following instructions?
  - a. DEC CX, JNE
11. The instruction MOV CX, DADD is what addressing mode?
  - a. Immediate

Program Listing MOV BL, E2H MOV CL, 100b ROL BL, CL

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:
  - a. INT
13. Which of the following DOS Debug instructions would set a break point at memory location 010C?
  - a. G = 100 10C
14. How many bytes are there in this short sequence of code? B400CD164CCD21
  - a. 7
15. How many address lines would be required to address 128 MB directly?
  - a. 27
16. What are the contents of CX after this program has been run?
  - a. D800h
17. 1010 0110 in 2's complement equals\_\_\_\_ in base 10.

a. -90

Memory Location    Contents

|      |    |
|------|----|
| 1103 | 24 |
| 1102 | D8 |
| 1101 | 00 |
| 1100 | 21 |

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

a. 01D8

19. What is 11.437510 in binary?

a. 001011.01110

20. If CX is 0000 what will CX be after a “LOOP” instruction?

a. FFFF

21. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?

a. 7DFA

22. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010E OV UP EI PL NZ NA PO NC

1D72:010F 7D08 JGE 0115

How many bytes will the processor jump if the conditions for a jump were met?

a. 8

23. What command in DEBUG would be used to execute interrupts?

a. P

24. What is the advantage of C Language over Assembly Language?

a. C is transportable to other microprocessor architectures

25. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

a. OF = 1, CF = 0

26. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

a. 36

27. Given:

AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D NV UP EI PL NZ NA PO NC

1D72:010D 7DF6 JNL 0116

- a. -10
- 28. AND'ing 1FH and 20H will result in which of the following?

- a. 0

29. Given:

13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

- a. 72
- 30. A microprocessor with a 31-bit address bus could access how much memory?

- a. 2 GB

31. For the instruction sequence below, determine the contents of the register AL after this program is executed: Listing

- a. 51H

32. Given:

AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

- a. -256
- 33. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- a. 32, 32, 13, 10, 36

Spring 2012

MOV AL, 73h

ADD AL, 78h

DAA

- 34. What is the advantage of assembly language over C language?
- a. The assembler creates much faster executable code

35. How many lines would be required to address 128 mb directly?

a. 27

36. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?

a. 020E

37. Given

W e l c o m e \_ t o \_ A s s e m 0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D 0B0E:0210 62 6C 79 20 4C 61 6E  
67-75 61 67 65 00 00 00 00

b l y \_ L a n g u a g e

An Ascii message begins at memory location 0200, what is the message?

a. Welcome to Assembly Language

38. Which of the following DOS Debug instructions would be used to change the IP register to 0110?

a. RIP

39. Moore’s law has accurately predicted the growth rate in the number of transistors per ide for the last 40 years.

What is the rate?

a. Doubling every 18 – 24 months

40. Which of the following DOS debug instructions would set a break point at memory location 010C?

a. G = 1000 10C

41. AND’ing 1FH and 02H will result in which of the following?

a. 02

42. If CX is 0000 what will CX be after a “LOOP” instruction?

a. FFFF

43. The number of bits in a word are:

a. 16

44. In x86 architecture, ALU stands for which of the following?

a. Arithmetic Logical Unit

45. Given:

What will the IP value be after a “t” command is executed in DOS DEBUG?

a. 010B

46. What is the number 1010.01012 in decimal?

- a. 10.31
47. How many cores does the propeller microcontroller have?
- a. 8
48. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?
- a. 7DF2
49. What command in debug would be used to step through a program line by line?
- a. T
50. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:
- a. INT
51. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
- a. 24
52. Given:

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- a. -10
53. Determine the contents of the register BL after the following instructions have been executed:
- a. E2H
54. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- a. 20, 20, 0D, 0A, 24
55. AND’ing 10H and 2FH will result in which of the following?
- a. 0
56. A “NOP” instruction in a program will
- a. Perform a No Operation
57. Given:

What is the signed decimal value of the number in the AX register?

- a. -16
58. How many address lines would be required to address 64 MB directly?
- a. 26

59. What command in debug would be used to execute interrupts?
- a. P
60. What high level language is the propeller programmed in?
- a. Spin
61. Which of the following DOS debug instructions would be used to change the IP register to 010C?
- a. RIP
62. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:
- a. Pulse width modulation
63. Which command would you use to execute another core in the propeller microcontroller?
- a. Cognew
64. Given:

What will the IP value be after the “t” command is executed in DOS debug?

- a. 0119h
65. The instruction MOV CD, DADD is what addressing mode?
- a. Immediate
66. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?
- a. F0
67. How many byte in decimal will the processor jump if the conditions for a jump are met?
- a. 24
68. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?
- a. Sets the Propeller pins P4 through P9 as output pins
69. What are the contents of DX after this program has been run:
- a. FFF0h.
70. The number of nibbles in a word are:
- a. 4
71. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?
- a. Create 10 second delayalBIUOP SP” instruction?  
**F011**

31. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?

0

32. In the Propeller microcontroller, the term "Method" is(are) which of the following?

A block of executable commands that has variables, can receive parameters, and returns a value.

33. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?

MOV CX, 30H or MOV CX, 48

34. A "pull up" resistor is used in digital circuits to do what?

To keep the signal "tied" high until the line is active (goes low)

EEE-174 MT1

1. What is the hexadecimal encoding for "JNL" for a jump back 8 bytes?

OF F8

2. In X86 architecture, ALU stands for which of the following?

Arithmetic Logic Unit

3. The number of nibbles in a word are:

16/4 = 4

4. The instruction MOV BX, [2BAD] is what addressing mode?

Direct

5. What is the hexadecimal encoding for adding DX with BX and storing the result in BX?

01CB

6. A microprocessor with a 32-bit address bus could access how much memory?

$2^{32}$ =4GB ,  $2^8$ =256KB,  $2^{32}$ =4GB

7. You add 7+6 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:

|        |   |                                   |
|--------|---|-----------------------------------|
| 0111   | 1 | OF=1, the sign bit has changed    |
| +0110  |   | CF=1, there is a carryon of bit 7 |
| 0 1101 |   |                                   |

8. Which of the following DOS Debug instructions would be used change the IP register to 100?

RIP

Convert 129.C hexadecimal into decimal

297.75 \*note after decimal divide by 16

8.1 What is the RS232C, specification voltage range for the Logic 0 output?

+3v to +25v.

8.1 What is the RS232C, specification voltage range for the Logic 1 output?

-3v to -25v.

9. What are the contents of CX after this program has been run:

| Memory location | Contents |
|-----------------|----------|
| 5514            | 24       |
| 5513            | D8       |
| 5512            | 00       |
| 5511            | 21       |

MOV DX, 11h  
MOV CX, [5512]  
MOV BX, 5511h  
SUB DX, [BX]

0000h

\*\*\*What are the contents of DX after this program has been run:

FFF0h

9. What are the contents of BX after this program?

**What are the contents of BX after this program?**

| Program       | Memory location | Contents |
|---------------|-----------------|----------|
| MOV BX, 2024h | 2026            | F2       |
| MOV CX, 4eh   | 2025            | 59       |
| DEC BX        | 2024            | 39       |
| AND CX, [BX]  | 2023            | 4E       |

Select one:

- a. 2023h
- b. F239h
- c. 2420h
- d. 2024h
- e. 394Eh

Assuming DS = 1000h, the instruction sequence in the listing below takes the last byte in the transfer from memory at:

**Assuming DS = 1000h, the instruction sequence in the listing below takes the last byte in the transfer from memory at:**

|              |
|--------------|
| STD          |
| MOV CX, 500H |
| MOV DX, 100H |
| MOV SI, 250H |
| A1: LODSB    |
| OUT DX, AL   |
| LOOP A1      |

Select one:

- a. 10500h
- b. 10250h
- c. 1750h
- d. 10750h
- e. 0FD50h

A microprocessor with a 26-bit address bus could access how much memory?

64MB

10. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

-48

11. How many bytes are there in this short sequence of code? B815B400CD16CD20

8

12. In using INT 10h to set the video code to 640x350, what value must be in the AH register?

AX  
AH|AL  
00

00h

13. Moore's law has accurately predicted the growth rate in the number of transistors per die for the last 25 years. What is that rate?

Doubling every 18-24 months

14. INT 21h, Function 09h requires three things set up before calling in order to correctly print a string,

Hello\_msg. They are:

DS=SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h

15. Given

13A7:0110 CD 20 32 20 54 68 69 73-20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69-64 75 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in decimal?

50

An input buffer is at memory location 0118, what is the size of the buffer in decimal?

32 if 0118 is 50 then answer is 80

16. Here is a short sequence of code: 7413CD16EB157D213C04EBF0A3C6. All of the instructions are two bytes long. The third instruction operator is: EB15

JMP

17. What is the hexadecimal encoding for loading DX with a word (value) from memory location 0820h?

8B162008

18. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV UP EI PL NZ NA PO NC

1D72: 010D 7D09 JGE 0116

How many bytes will the processor jump if the condition for a jump were met?

9

19. F6 in 2's complement equals \_\_\_\_\_ in base 10.

-10

21. Determine the contents of register BH after the following instructions have executed:

70h MOV [0202], AX

22. Given:

AX=2247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC

1D72: 0106 EB0F JMP 0118

What will be the IP value be after a "t" command is executed in DOS Debug?

0118h

23. Which of the following DOS Debug instructions would set a break point at memory location 010E?

G=100 10E

24. Given:

0B0E: 0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D

0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message?

Welcome to Assembly Language

25. Given:

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 IP=0104 NV UP EI NG NZ NA PE NC

1D72:0104 7002 JO 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0106h

26. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO CY

1D72:0109 7D06 JNL 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0110

27. The instruction MOV BX, 2BAD is what addressing mode?

Immediate

28. What is the hexadecimal encoding for "JNL" for a jump back 10 bytes?

7DF4

29. How many address lines would be required to address 64MB directly?

26 / $2^{26}$ =67mb/

30. The number of nibbles in a double word are: 8

31. In using INT 10h to move the screen cursor to return on the same line, what value must be in the AX register?  
0E0Dh

32. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV U EI PL NZ NA PO NC  
1D72: 010D 7D09 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug?

010Fh

33. Which of the following DOS Debug instructions would be used change the AX register?

RAX

34. How many Bytes are there in this short sequence of code? B815B400CD168A3CCD20  
10

35. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
1D72: 0111 EB08 JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug?

0119h

36. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI NG NZ NA PO CY  
1D72: 010E 7D06 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0118

37. How many bytes decimal will the program jump for JNB instruction, given the following?

AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1376 ES=1376 SS=1376 CS=1376 IP=0100 NV UP EI PL NZ NA PO NC

1376: 0100 73E0 JNB 00E2

-32

38. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV UP EI PL NZ NA PO NC  
1D72: 010D 7F09 JNLE 0118

What is the decimal value of the signed number in the AX register?

-0032

39. How many address lines would be required to address 512 MB directly?

29 / $2^9$ =29/

40. What is the hexadecimal encoding for adding BX with CX and storing the result in BX?

03CB or 03D9 -- 02CB or 02D9 - these are the only options

41. The binary number, 0111 1110, represents what values; in Hex, and as a BCD number?

7E, 7 invalid

42. In using INT 10h to set the video mode to 640 X 200, what value must be in the AX register?

0006h

43. what is the hexadecimal encoding for loading AH with a word from memory location 0520h?

8B262005 (choose 8B162005 if that is only option)

44. What is the hexadecimal encoding for "JGE" for a jump back 10 bytes?

7DF6

45. Given:

13A7: 0110 CD 20 48 20 54 68 69 73-20 69 73 20 74 68 65 20  
13A7: 0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
13A7: 0120 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in bytes in decimal?

72

46. Given:

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72: 0104 7002 JNO 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0118

47. Determine the contents of register AH after the following instruction have executed:

```
MOV BX,BA70H  
MOV AX,47E7H  
MOV BL, 0FH  
AND AL,BL  
MOV [0202], AX
```

47H

48. In using INT 10h to move the screen cursor to return to the beginning of the line, what value must be in the AX register?

0E0Dh

49. Given:

1376:0110 48 61 76 65 20 61 20 67-72 65 61 74 20 53 70 72

1376:0120 69 6E 67 20 52 65 63 65-73 73 21 00 00 00 00 00

An ASCII message begins at memory location 0110, what is the message?

Have a great Spring Recess!

50. You add 1+8 through a 4 bit integer unit. The state of the OF and CF flags after the ad will be:

OF=0, CF=0

## Page 16

51. What is the advantage of Assembly Language over C Language?

The Assembler creates much faster executable code

52. In The X86 lab 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence "Hello World 0"?

020E

53. Which of the following DOS Debug instructions would set a break point at memory location 010C?

G = 100 10C

54. If CX is 0000 what will CX be after a "LOOP" instruction?

FFFF

55. Given:

AX=0353 BX=0534 CX=0000 DX=0180

DS= 1D72 IP=0109 OV UP EI PL NZ NA PO CY

1D72:0109 7D06 (OV=1,NV=1 ZR=1, NZ=0)

010B

56. How many core does the propeller microcontroller have?

8

57. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?

7DF2 (marked right) though maybe 7DF4

58. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. instructions are a word long. The third instruction operator is:

INT

59. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?  
24

60. The ASCII codes for space , space, carriage return, line feed, end of string in decimal are:

32,32,13,10,36

61. A "NOP" instruction in a program will:

Perform a No Operation

62. How many address lines would be required to address 64MB directly?

26

63. What command in DEBUG would be used to execute interrupts?

P

64. What high level language is the propeller programmed in?

Spin

65. Which of the following DOS Debug instruction would be used to change the IP register to 010C?

RIP

66. The acronym PWM used in the Parrallax Propeller and MicroChip PIC18, is defined as:

Pulse Width Modulation

67. Which command would you use to execute another core in the propeller microcontroller?

Cognew

68. Given: IP=0111 NV UP EI NG NZ NA PO CY

1D72:0111 JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug?

0119h

69. The instruction MOV CX, DADD is what addressing mode?  
 Immediate
69. The instruction MOV CX, [DADD] is what addressing mode?  
 Direct
70. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?  
 A0
71. In the Propeller microcontroller, the command "dira[4..9] := %111111" would cause the processor to do with of the following?  
**Sets the Propeller pins P4 through P9 as output pins**
72. In the Propeller microcontroller, the command "waitcnt(clkfreq\*10 + cnt)" would cause the processor to do with of the following?  
**Create 10 second delay**
73. The "LOOPNE" instruction is equivalent to which of the following instructions?  
 DEC CX, JNE
74. On the Arduino platform, what is the programming language used?  
 C

EEE 174 Midterm Study Guide  
S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:
  - a. DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36  
 Or the other option in hex is below
  - b. DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h
2. Moore's Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?
  - a. Doubling every 18 – 24 months

3. Given:

```
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY
1D72:0109 7D06 JNL 0118
```

What will the IP value be after a "t" command is executed in DOS Debug?

a. **010B**

4. A "NOP" instruction will:

a. **Perform a No Operation**

5. Given:

```
AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC
1D72:0106 EB0F JMP 0118
```

What will the IP value be after a "t" command is executed in DOS Debug?

a. **0118h**

6. In x86 architecture, BIU stands for which of the following?

a. **Bus Interface Unit**

7. 7. Determine the contents of register BL after the following instructions have been executed:

a. **2EH**

8. 8. The number of bytes in a double word are:

a. **4**

9. In x86 architecture, ALU stands for which of the following?

a. **Arithmetic Logic Unit**

10. The "LOOPNE" instruction is equivalent to which of the following instructions?

a. **DEC CX, JNE**

11. The instruction MOV CX, DADD is what addressing mode?

**a. Immediate**

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

**a. INT**

13. Which of the following DOS Debug instructions would set a break point at memory location 010C?

**a. G = 100 10C**

14. How many bytes are there in this short sequence of code? B400CD164CCD21

**a. 7**

15. How many address lines would be required to address 128 MB directly?

**a. 27**

16. What are the contents of CX after this program has been run:

**a. D800h**

17. 1010 0110 in 2's complement equals \_\_\_\_ in base 10.

**a. -90**

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

**a. 01D8**

19. What is 11.4375<sub>10</sub> in binary?

**a. 001011.01110**

20. If CX is 0000 what will CX be after a “LOOP” instruction?

**a. FFFF**

21. Given:

```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO NC  
1D72:010F 7D08 JGE 0115
```

How many bytes will the processor jump if the conditions for a jump were met?

**a. 8**

22. What command in DEBUG would be used to execute interrupts?

**a. P**

23. What is the advantage of C Language over Assembly Language?

**a. C is transportable to other microprocessor architectures**

24. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

**a. OF = 1, CF = 0**

25. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

**a. 36**

26. Given:

```
AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JNL 0116
```

**a. -10**

27. AND’ing 1FH and 20H will result in which of the following?

**a. 0**

28. Given:

```
13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 00
```

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

- a. 72
29. A microprocessor with a 31-bit address bus could access how much memory?  
a. 2 GB
30. For the instruction sequence below, determine the contents of the register AL after this program is executed:  
a. 51H

31. Given:

```
AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY
1D72:0109 7D06 JNL    0118
```

What is the signed decimal value of the number in the AX register?

- a. -256
32. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:  
a. 32, 32, 13, 10, 36

### Spring 2012

33. What is the advantage of assembly language over C language?  
a. The assembler creates much faster executable code
34. How many lines would be required to address 128 mb directly?  
a. 27
35. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?  
a. 020E
36. Given
- |           |                                                                                                                               |
|-----------|-------------------------------------------------------------------------------------------------------------------------------|
| 0B0E:0200 | W e l c o m e _ t o _ A s s e m                                                                                               |
| 0B0E:0210 | 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D<br>62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00<br>b l y _ L a n g u a g e |
- An Ascii message begins at memory location 0200, what is the message?  
a. Welcome to Assembly Language
37. Which of the following DOS Debug instructions would be used to change the IP register to 0110?  
a. RIP
38. Moore’s law has accurately predicted the growth rate in the number of transistors per ide for the last 40 years. What is the rate?  
a. Doubling every 18 – 24 months
39. Which of the following DOS debug instructions would set a break point at memory location 010C?  
a. G = 1000 10C
40. AND’ing 1FH and 02H will result in which of the following?  
a. 02
41. If CX is 0000 what will CX be after a “LOOP” instruction?  
a. FFFF
42. The number of bits in a word are:  
a. 16
43. In x86 architecture, ALU stands for which of the following?  
a. Arithmetic Logical Unit
44. Given:

```
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY
1D72:0109 7D06 JGE    0118
```

What will the IP value be after a “t” command is executed in DOS DEBUG?

- a. 010B

45. What is the number 1010.0101<sub>2</sub> in decimal?  
 a. 10.31
46. How many cores does the propeller microcontroller have?  
 a. 8
47. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?  
 a. 7DF2
48. What command in debug would be used to step through a program line by line?  
 a. T
49. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:  
 a. INT
50. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
 a. 24

51. Given:

|                       |                |                |                |                |                                |                |                |
|-----------------------|----------------|----------------|----------------|----------------|--------------------------------|----------------|----------------|
| <b>AX=FFE0</b>        | <b>BX=3534</b> | <b>CX=0000</b> | <b>DX=0180</b> | <b>SP=FFEE</b> | <b>BP=0000</b>                 | <b>SI=0000</b> | <b>DI=0000</b> |
| <b>DS=1D72</b>        | <b>ES=1D72</b> | <b>SS=1D72</b> | <b>CS=1D72</b> | <b>IP=010D</b> | <b>NV UP EI PL NZ NA PO NC</b> |                |                |
| <b>1D72:010D 0DF6</b> |                |                |                | <b>JNL</b>     | <b>0116</b>                    |                |                |

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- a. 10

52. Determine the contents of the register BL after the following instructions have been executed:  
 a. E2H

| Program Listing |
|-----------------|
| MOV BL, E2H     |
| MOV CL, 08H     |
| ROL BL, CL      |

53. The ASCII codes for space, space, carriage return, line feed, end of string in hex are:

- a. 20, 20, 0D, 0A, 24

54. AND’ing 10H and 2FH will result in which of the following?

- a. 0

55. A “NOP” instruction in a program will

- a. Perform a No Operation

56. Given:

|                       |                |                |                |                |                                |                |                |
|-----------------------|----------------|----------------|----------------|----------------|--------------------------------|----------------|----------------|
| <b>AX=FFF0</b>        | <b>BX=3534</b> | <b>CX=0000</b> | <b>DX=0180</b> | <b>SP=FFEE</b> | <b>BP=0000</b>                 | <b>SI=0000</b> | <b>DI=0000</b> |
| <b>DS=1D72</b>        | <b>ES=1D72</b> | <b>SS=1D72</b> | <b>CS=1D72</b> | <b>IP=0109</b> | <b>OV UP EI PL NZ NA PO CY</b> |                |                |
| <b>1D72:0109 7D06</b> |                |                |                | <b>JGE</b>     | <b>0118</b>                    |                |                |

What is the signed decimal value of the number in the AX register?

- a. -16

57. How many address lines would be required to address 64 MB directly?

- a. 26

58. What command in debug would be used to execute interrupts?

- a. P

59. What high level language is the propeller programmed in?

- a. Spin

60. Which of the following DOS debug instructions would be used to change the IP register to 010C?

- a. RIP

61. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:

- a. Pulse width modulation

62. Which command would you use to execute another core in the propeller microcontroller?

- a. Cognew

63. Given:

|                       |                |                |                |                |                                |                |                |
|-----------------------|----------------|----------------|----------------|----------------|--------------------------------|----------------|----------------|
| <b>AX=FFD0</b>        | <b>BX=3534</b> | <b>CX=0000</b> | <b>DX=0180</b> | <b>SP=FFEE</b> | <b>BP=0000</b>                 | <b>SI=0000</b> | <b>DI=0000</b> |
| <b>DS=1D72</b>        | <b>ES=1D72</b> | <b>SS=1D72</b> | <b>CS=1D72</b> | <b>IP=0111</b> | <b>NV UP EI NG NZ NA PO CY</b> |                |                |
| <b>1D72:0111 EB08</b> |                |                |                | <b>JMP</b>     | <b>0119</b>                    |                |                |

What will the IP value be after the “t” command is executed in DOS debug?

- a. 0119h

64. The instruction MOV CD, DADD is what addressing mode?  
 a. Immediate
65. How many byte in decimal will the processor jump if the conditions for a jump are met?  
 a. 24
66. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?  
 a. Sets the Propeller pins P4 through P9 as output pins
67. What are the contents of DX after this program has been run:  
 a. FFF0h.

| Memory Location | Contents |
|-----------------|----------|
| 5514            | 24       |
| 5513            | D8       |
| 5512            | 00       |
| 5511            | 21       |
| 5510            | 00       |

68. The number of nibbles in a word are:

- a. 4

What are the TTL logic level voltages for a logic 0 and logic 1?

0V to +5V

What is 152.875 converted to double precision fp?

40 63 06 00 00 00 00

What is 152.1875 converted to single precision fp?

**Question 123**

Answer saved

Marked out of 1

▼ Flag question

**What is 152.1875 Converted to single precision FP?**

Select one:

a. 3D A0 00 00

b. C1 1C 80 00

c. 43 18 30 00

d. C2 C0 8A 3D

e. 40 63 06 00

f. 41 1C 00 00

69. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?

- a. Create 10 second delay

70. The “LOOPNE” instruction is equivalent to which of the following instructions?

- a. DEC CX JNE

71. Given

```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI NG NZ NA PO NC
1D72:010D EB07           JMP      0114
```

How many bytes will the processor jump if the conditions for a jump were met?

- a. 7

72. On the Arduino platform, what is the programming language used?

- a. C

73. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?

- a. A12008

74. Which of the following would be used to set the TRISA register to control the direction of PIC18 Port to input?

- a. 1

75. The acronym ADC in microcontrollers stands for which of the following?

- a. Analog to Digital converter

76. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

- a. OF = 1, CF = 0

Compare and contrast the Harvard architecture with the von Neumann

**Question 99**  
Answer saved  
Marked out of 1  
[Flag question](#)

**Compare and contrast the Harvard architecture with the von Neumann architecture.**

Select one:

a. The von Neumann architecture uses big endian addressing.  
 b. The Harvard architecture uses programmed I/O.  
 c. The von Neumann architecture uses separate program and data memory.  
 d. The Harvard type uses a separate program and data memory  
 e. The Harvard architecture uses big endian addressing.

## **von Neumann vs. Harvard**

- **von Neumann**
  - Same memory holds data, instructions.
  - A single set of address/data buses between CPU and memory
- **Harvard**
  - Separate memories for data and instructions.
  - Two sets of address/data buses between CPU and memory

Chenyang Lu

CSE 4675

4

### S2010 MT2

1. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?  
**a. 7**
2. If CX is 0003, what will CX be after a “LOOP” instruction?  
**a. 0002**
3. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?  
**a. A 5 second delay**
4. What is the number 1011.0101<sub>2</sub> in decimal?  
**a. 11.31**
5. This section of memory represents a stack. What type of program is this?  
BEEF : FFD0 00 00 00 00 00 00-00 00 00 00 00 00 00  
BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
BEEF : FFFF 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF  
**a. COM program**
6. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?  
**a. BDCA**
7. Determine the contents of register BL after the following instruction have been executed:

**a. 2EH**

8. What Hex values must be sent to address the key pad rows on the PPE board?  
**a. 1, 2, 4, 8**
9. What is the number  $32.4375_{10}$  in binary?  
**a. 100000.01110**
10. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
**a. 24**
11. The acronym PWM used for motor control, is defined as which of the following?  
**a. Pulse Width Modulation**
12. Given the short code, what is the value in AX after the program is run?  
**a. 0500**
  
13. What flag(s) does the “LOOPENZ” instruction look at to determine whether to loop or not?  
**a. ZF**
14. How many nibbles are in double precision IEEE floating point format number?  
**a. 16**
15. What type of program is this?  
AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC  
1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD  
**a. EXE**
16. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?  
**a. F00D**
17. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?  
**a. A0**
18. What is the numeric sequence of the key pad columns on the PPE board?  
**a. 37, 2F, 1F**
19. What is -130 decimal in 2's complement (8bits)?  
**a. 01111110**
20. Which of the following is a valid x86 command for multiplying a number?  
**a. MUL BX**
21. The number of bits in single precision IEEE floating point format are:  
**a. 32**
22. You are trying to rebuild a HELLO project program in MASM and you get the following error:  
“LINK : warning L4021: no stack segment”.  
What would be the reason for such an error?  
**a. No project template for COM was selected**
23. A “pull down” resistor is used in digital circuits to do what?  
**a. To keep the signal line “tied” low until the line is active (goes high)**
24. A “POP” instruction:  
**a. increments the SP**
25. AND'ing 10H and 2FH will result in which of the following?  
**a. 0**
26. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?  
**a. Sets the Propeller pins P4 through P9 as output pins**
27. What commands in MASM-CodeView would be used to step through a program line by line?

- a. T (F8)**
28. If the SP is F00F, what will the SP value be after a “POP SP” instruction?  
**a. F011**
29. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?  
**a. 0**
30. In the Propeller microcontroller, the term “Method” is(are) which of the following?  
**a. A block of executable commands that has variables, can receive parameters, and returns a value.**
31. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?  
**a. MOV CX, 30H or MOV CX, 48**
32. A “pull up” resistor is used in digital circuits to do what?  
**a. To keep the signal “tied” high until the line is active (goes low)**
33. With a PUSHA instruction, what will be the order of the register (registers A ~ D) contents on the stack?  
**a. ACDB**
34. The LOOPNE instruction performs which of the following?  
**a. Decrements CX, tests the ZF flag, if it is not zero jumps to address specified.**
35. What must the value be and in what register, prior to executing a LOOPNE instruction, to discontinue looping?  
**a. CX =1**
36. With a POPA instruction, what will be the order of the registers (registers A~ D) restored from stack?  
**a. BDCA**
37. What flags does the “LOOPNZ” instruction look at to determine whether to loop or not?  
**a. ZF**
38. If the SP is F00F, what is the SP value after a “POP BX” instruction?  
**a. F011**
39. If CX is 0001, what will CX be after a “LOOPNZ” instruction?  
**a. 0000**
40. -11.25 in decimal converted to binary would be:  
**a. -1011.0100**
41. What is 31.4375 in binary?  
**a. 011111.0111**
42. Using debug which command should be used to change the flag setting?  
**a. RF**
43. What is 14.4375 in binary?  
**a. 001110.01110**

EEE 174 Midterm Study Guide

S2010 MT 1

2. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:  
**b. DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36**
77. Moore’s Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?  
**a. Doubling every 18 – 24 months**
78. Given:  
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118
- What will the IP value be after a “t” command is executed in DOS Debug?  
**a. 010B**
79. A “NOP” instruction will:  
**a. Perform a No OPeration**
80. Given:  
AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72:0106 EB0F JMP 0118

What will the IP value be after a “t” command is executed in DOS Debug?

a. **0118h**

81. In x86 architecture, BIU stands for which of the following?

a. **Bus Interface Unit**

82. 7. Determine the contents of register BL after the following instructions have been executed:

a. **2EH**

83. 8. The number of bytes in a double word are:

a. **4**

84. In x86 architecture, ALU stands for which of the following?

a. **Arithmetic Logic Unit**

85. The “LOOPNE” instruction is equivalent to which of the following instructions?

a. **DEC CX, JNE**

86. The instruction MOV CX, DADD is what addressing mode?

a. **Immediate**

87. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

a. **INT**

88. Which of the following DOS Debug instructions would set a break point at memory location 010C?

a. **G = 100 10C**

89. How many bytes are there in this short sequence of code? B400CD164CCD21

a. **7**

90. How many bytes are there in this short sequence of code? B400CD16CD20

a. **6**

91. How many address lines would be required to address 128 MB directly?

a. **27**

92. What are the contents of CX after this program has been run:

a. **D800h**

93. 1010 0110 in 2's complement equals\_\_\_\_ in base 10.

a. **-90**

94. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

a. **01D8**

95. What is 11.4375<sub>10</sub> in binary?

a. **001011.01110**

96. If CX is 0000 what will CX be after a “LOOP” instruction?

a. **FFFF**

97. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO NC  
1D72:010F 7D08 JGE 0115

How many bytes will the processor jump if the conditions for a jump were met?

a. **8**

98. What command in DEBUG would be used to execute interrupts?

a. **P**

99. What is the advantage of C Language over Assembly Language?

a. **C is transportable to other microprocessor architectures**

100. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

- a. **OF = 1, CF = 0**
100. In MASM, with a “MOV CX, 24h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
 a. **36**
101. Given:  
 AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
 1D72:010D 7DF6 JNL 0116  
 a. **-10**
102. AND’ing 1FH and 20H will result in which of the following?  
 a. **0**
103. Given:  
 13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20  
 13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
 13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00  
 An input buffer is at memory location 0115, what is the size of the buffer in decimal?  
 a. **72**
104. A microprocessor with a 31-bit address bus could access how much memory?  
 a. **2 GB**
105. For the instruction sequence below, determine the contents of the register AL after this program is executed:  
 a. **51H**
106. Given:  
 AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
 1D72:0109 7D06 JNL 0118  
 What is the signed decimal value of the number in the AX register?  
 a. **-256**
107. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:  
 a. **32, 32, 13, 10, 36**
- Spring 2012**
108. What is the advantage of assembly language over C language?  
 a. The assembler creates much faster executable code
109. How many lines would be required to address 128 mb directly?  
 a. **27**
110. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?  
 a. **020E**
111. Given  
 W e l c o m e \_ t o \_ A s s e m  
 0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D  
 0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00  
 b l y \_ L a n g u a g e
- An Ascii message begins at memory location 0200, what is the message?  
 a. Welcome to Assembly Language
112. Which of the following DOS Debug instructions would be used to change the IP register to 0110?  
 a. **RIP**
113. Moore’s law has accurately predicted the growth rate in the number of transistors per ide for the last 40 years.  
 What is the rate?  
 a. Doubling every 18 – 24 months

114. Which of the following DOS debug instructions would set a break point at memory location 010C?  
 a. G = 1000 10C
115. AND'ing 1FH and 02H will result in which of the following?  
 a. 02
116. If CX is 0000 what will CX be after a "LOOP" instruction?  
 a. FFFF
117. The number of bits in a word are:  
 a. 16
118. In x86 architecture, ALU stands for which of the following?  
 a. Arithmetic Logical Unit
119. Given:

```
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY
1D72:0109 7D06 JGE 0118 1 0
```

What will the IP value be after a "t" command is executed in DOS DEBUG?

- a. 010B
120. What is the number 1010.0101<sub>2</sub> in decimal?  
 a. 10.31
121. How many cores does the propeller microcontroller have?  
 a. 8
122. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?  
 a. 7DF2
123. What command in debug would be used to step through a program line by line?  
 a. T
124. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:  
 a. INT

125. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?  
 a. 24
126. Given:

```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 TP=010D NV UP EI PL NZ NA PO NC
1D72:010D (DF6) JNL 0116 0 0
```

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- a. 10
127. Determine the contents of the register BL after the following instructions have been executed:  
 a. E2H

| Program Listing |
|-----------------|
| MOV BL, E2H     |
| MOV CL, 08H     |
| ROL BL, CL      |

128. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:  
 a. 20, 20, 0D, 0A, 24

129. AND'ing 10H and 2FH will result in which of the following?  
 a. 0

130. A "NOP" instruction in a program will  
 a. Perform a No Operation

131. Given:

```
AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY
1D72:0109 7D06 JGE 0118
```

What is the signed decimal value of the number in the AX register?

- a. -16
132. How many address lines would be required to address 64 MB directly?

- a. 26
133. What command in debug would be used to execute interrupts?  
a. P
134. What high level language is the propeller programmed in?  
a. Spin
135. Which of the following DOS debug instructions would be used to change the IP register to 010C?  
a. RIP
136. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:  
a. Pulse width modulation
137. Which command would you use to execute another core in the propeller microcontroller?  
a. Cognew
138. Given:  
**AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
1D72:0111 EB08 JMP 0119**  
What will the IP value be after the “t” command is executed in DOS debug?  
a. 0119h
139. The instruction MOV CD, DADD is what addressing mode?  
a. Immediate
140. How many byte in decimal will the processor jump if the conditions for a jump are met?  
a. 24
141. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?  
a. Sets the Propeller pins P4 through P9 as output pins
142. What are the contents of DX after this program has been run:  
a. FFF0h.

| Memory Location | Contents |
|-----------------|----------|
| 5514            | 24       |
| 5513            | D8       |
| 5512            | 00       |
| 5511            | 21       |
| 5510            | 00       |

143. The number of nibbles in a word are:  
a. 4
144. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?  
a. Create 10 second delay
145. The “PPE board E” instruction is equivalent to which of the following instructions?  
a. DEC CX JNE
146. Given:  
**AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI NG NZ NA PO NC  
1D72:010D EB07 JMP 0114**  
How many bytes will the processor jump if the conditions for a jump were met?  
a. 7
147. On the Arduino platform, what is the programming language used?  
a. C
148. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?  
a. A12008
149. Which of the following would be used to set the TRISA register to control the direction of PIC18 Port to input?  
a. 1
150. The acronym ADC in microcontrollers stands for which of the following?  
a. Analog to Digital converter
151. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:  
a. OF = 1, CF = 0



S2010 MT2

2. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?
  - b. 7
44. If CX is 0003, what will CX be after a “LOOP” instruction?
  - a. 0002
45. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?
  - a. A 5 second delay
46. What is the number 1011.0101<sub>2</sub> in decimal?
  - a. 11.31
47. This section of memory represents a stack. What type of program is this?

BEEF : FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  
BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

  - a. COM program
48. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?
  - a. BDCA
49. Determine the contents of register BL after the following instruction have been executed:
  - a. 2EH
  
50. What Hex values must be sent to address the key pad rows on the PPE board?
  - a. 1, 2, 4, 8
51. What is the number 32.4375<sub>10</sub> in binary?
  - a. 100000.01110
52. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?
  - a. 24
53. The acronym PWM used for motor control, is defined as which of the following?
  - a. Pulse Width Modulation
54. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?
  - a. Bit 7 of port D is set to output
55. Given the short code, what is the value in AX after the program is run?
  - a. 0500
  
56. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?
  - a. ZF
57. How many nibbles are in double precision IEEE floating point format number?
  - a. 16
58. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC  
1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD

  - a. EXE
59. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?
  - a. F00D
60. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

- a. A0**
61. What is the numeric sequence of the key pad columns on the PPE board?  
**a. 37, 2F, 1F**
  62. What is -130 decimal in 2's complement (8bits)?  
**a. 0111110**
  63. Which of the following is a valid x86 command for multiplying a number?  
**a. MUL BX**
  64. The number of bits in single precision IEEE floating point format are:  
**a. 32**
  65. You are trying to rebuild a HELLO project program in MASM and you get the following error:  
“LINK : warning L4021: no stack segment”.  
What would be the reason for such an error?  
**a. No project template for COM was selected**
  66. A “pull down” resistor is used in digital circuits to do what?  
**a. To keep the signal line “tied” low until the line is active (goes high)**
  67. A “POP” instruction:  
**a. increments the SP**
  68. AND'ing 10H and 2FH will result in which of the following?  
**a. 0**
  69. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?  
**a. Sets the Propeller pins P4 through P9 as output pins**
  70. What commands in MASM-CodeView would be used to step through a program line by line?  
**a. T (F8)**
  71. If the SP is F00F, what will the SP value be after a “POP SP” instruction?  
**a. F011**
  72. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?  
**a. 0**
  73. In the Propeller microcontroller, the term “Method” is(are) which of the following?  
**a. A block of executable commands that has variables, can receive parameters, and returns a value.**
  74. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?  
**a. MOV CX, 30H      or      MOV CX, 48**
  75. A “pull up” resistor is used in digital circuits to do what?  
**a. To keep the signal “tied” high until the line is active (goes low)**

## SECOND DOCUMENT

### S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:  
**DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36**
2. Moore’s Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?

#### Doubling every 18 – 24 months

3. Given:  
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118  
What will the IP value be after a “t” command is executed in DOS Debuge range of decimal numbers range of decimal numbers g?  
**010B**
4. A “NOP” instruction will:  
**Perform a No OPeration**
5. Given:  
AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72:0106 EB0F JMP 0118

What will the IP value be after a “t” command is executed in DOS Debug?

**0118h**

6. In x86 architecture, BIU stands for which of the following?

**Bus Interface Unit**

7. Determine the contents of register BL after the following instructions have been executed:

**2EH**

8. The number of bytes in a double word are:

**4**

9. In x86 architecture, ALU stands for which of the following?

**Arithmetic Logic Unit**

10. The “LOOPNE” instruction is equivalent to which of the following instructions?

**DEC CX, JNE**

11. The instruction MOV CX, DADD is what addressing mode?

**Immediate**

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

**INT 16**

13. Which of the following DOS Debug instructions would set a break point at memory location 010C?

**G = 100 10C**

14. How many bytes are there in this short sequence of code? B400CD164CCD21

**7**

15. How many address lines would be required to address 128 MB directly?

**27**

16. What are the contents of CX after this program has been run:

**D800h**

17. 1010 0110 in 2's complement equals \_\_\_\_ in base 10.

**-90**

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

**01D8**

19. What is 11.4375<sub>10</sub> in binary?

**001011.01110**

20. If CX is 0000 what will CX be after a “LOOP” instruction?

**FFFF**

22. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO NC  
1D72:010F 7D08 JGE 0115

How many bytes will the processor jump if the conditions for a jump were met?

**8**

23. What command in DEBUG would be used to execute interrupts?

**P**

24. What is the advantage of C Language over Assembly Language?

**C is transportable to other microprocessor architectures**

25. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:

**OF = 1, CF = 0**

26. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

**36**

27. Given:

AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JNL 0116

**-10**

28. AND'ing 1FH and 20H will result in which of the following?

**0**

29. Given:

13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0115, what is the size of the buffer in decimal?

**72**

30. A microprocessor with a 31-bit address bus could access how much memory?

**2 GB**

31. For the instruction sequence below, determine the contents of the register AL after this program is executed:

**51H**

32. Given:

AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register?

**-256**

33. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

**32, 32, 13, 10, 36**

S2010 MT2

1. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?

**7**

2. If CX is 0003, what will CX be after a “LOOP” instruction?

**0002**

3. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?

**A 5 second delay**

4. What is the number 1011.0101<sub>2</sub> in decimal?

**11.31**

5. This section of memory represents a stack. What type of program is this?

BEEF : FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00

BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F

BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF

**COM program**

6. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?

**B D C A**

7. Determine the contents of register BL after the following instruction have been executed:

**2EH**

8. What Hex values must be sent to address the key pad rows on the PPE board?

**1, 2, 4, 8**

9. What is the number 32.4375<sub>10</sub> in binary?

**100000.01110**

10. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?

**24**

11. The acronym PWM used for motor control, is defined as which of the following?

**Pulse Width Modulation**

12. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

**Bit 7 of port D is set to output**

13. Given the short code, what is the value in AX after the program is run?

**0500**

14. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?

**ZF**

15. How many nibbles are in double precision IEEE floating point format number?

**16**

17. What type of program is this?

AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC

1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD

**EXE**

18. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?

**F00D**

19. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

**A0**

20. What is the numeric sequence of the key pad columns on the PPE board?

**37, 2F, 1F**

21. What is -130 decimal in 2’s complement (8bits)?

**01111110**

22. Which of the following is a valid x86 command for multiplying a number?

**MUL BX**

23. The number of bits in single precision IEEE floating point format are:

**32**

24. You are trying to rebuild a HELLO project program in MASM and you get the following error:

“LINK : warning L4021: no stack segment”.

What would be the reason for such an error?

**No project template for COM was selected**

25. A “pull down” resistor is used in digital circuits to do what?

**To keep the signal line “tied” low until the line is active (goes high)**

26. A “POP” instruction:

**increments the SP**

27. AND’ing 10H and 2FH will result in which of the following?

**0**

28. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

**Sets the Propeller pins P4 through P9 as output pins**

29. What commands in MASM-CodeView would be used to step through a program line by line?

**T (F8)**

30. If the SP is F00F, what will the SP value be after a “POP SP” instruction?

**F011**

31. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?

**0**

32. In the Propeller microcontroller, the term “Method” is(are) which of the following?

**A block of executable commands that has variables, can receive parameters, and returns a value.**

33. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?

**MOV CX, 30H or MOV CX, 48**

34. A “pull up” resistor is used in digital circuits to do what?

**To keep the signal “tied” high until the line is active (goes low)**

How many bit(s) is/are required to represent a range of decimal numbers from 0 to 9?

a) 4

A “POP” instruction:

a) increments the Stack Pointer, SP

The instruction MOV CX, SI is what addressing mode?

a) Register Addressing - direct

The instruction MOV CX, [SI] is what addressing mode?

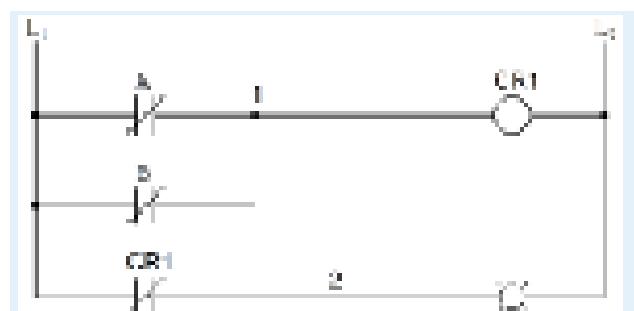
a) Register Indirect

Which of the following will cause a program with a LOOP instruction to loop 48 times (decimal)?

a) CX = 30h

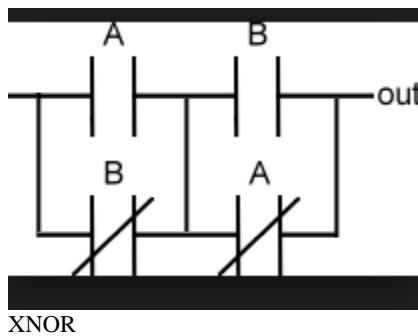
A “PUSH” instruction:

a) Decrements the SP/increments the IP



The Ladder Logic diagram would represent which of the following?

a) NOR



If you want to use a INT software interrupt function to print a string out to the screen, what is the function code, start pointer, termination character, and interrupt you need to use?

Select one:

- a) ah = 09h, ds:dx, "\$", 21h

You are trying to rebuild a HELLO project program in MASM and you get the following error: "LINK: fatal error L1089: HELLO.lrf: cannot open response file".

What would be the reason for such an error?

Select one:

- a) No source file is identified (no.asm file)

The instruction MOV SI, 4DAD is what addressing mode?

Select one:

- a) Immediate

How many nibble are in double precision IEEE floating point format numbers (64bit)?

Select one:

- a) 16

What flag(s) does the "LOOPNE" instruction look at to determine whether to loop or not?

- a) ZF

What is the numeric sequence of the key pad columns on the PPE board?

Select one:

- a) 37,2F,1F

18 hexadecimal would be what value in decimal?

- a) 24

In the PIC 18 with TRISD = 0b01000000, what is the configuration of the Port D?

Select one:

- a) Bit 6 of port D is set to input

In the PIC 18 with TRISD = 0b10000000, what is the configuration of the Port D?

Select one:

- a) Bit 7 of port D is set to input

What register(s) does the "LOOPNE" instruction look at to determine how many times to loop?

Select one:

- a) CX

If CX is 0003, what will CX be after a “LOOPNZ” instruction?

Select one:

- a) 0002

What type of program is this?

```
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1476 ES=1576 SS=1676 CS=1376 IP=0015 NV UP EI PL NE NR PO NC  
1376:0015 0100 ADD [BX+SI],AL DS:0000=CD
```

Select one:

- a) EXE program

Determine the content of register BL after the following instruction have been executed:

- a) E2H

```
MOV BL, 2EH  
MOV CL, 0100b  
ROL BL, CL
```

#### OTHER

A “pull down” resistor is used in digital circuits to do what?

- b) To keep the signal line “tied” low until the line is active (goes high)

A “pull up” resistor is used in digital circuits to do what?

- c) To keep the signal “tied” high until the line is active (goes low)

Double precision IEEE FP standard uses \_\_\_\_\_ nibbles to represent data:

- a) 16

How many bytes are in double precision IEEE floating point format numbers?

- b) 8

You are trying to rebuild a Hello project program in MASM and you get the following error: “LINK : warning L4021: no stack segment”.

- c) No project template for COM was selected

How many double words are in double IEEE floating point format numbers?

- d) 16

In x86 architecture, BIU stands for which of the following?

- e) Bus Interface Unit

Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is that rate?

b) Doubling every 18 – 24 months

NMI stands for what?

- a) Non Maskable Interrupt – it means it cannot be blocked

The letters “NC” labeled on relays and PLCs means which of the following?

- b) Normally Closed

The number of bits in single precision IEEE floating point format are:

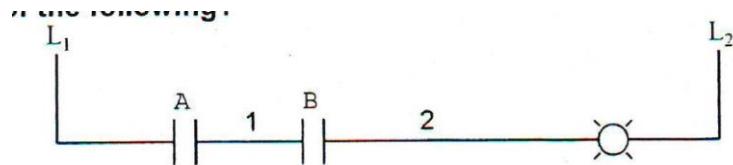
- c) 32

The number of bytes in extended precision IEEE floating point format are:

- b) 10

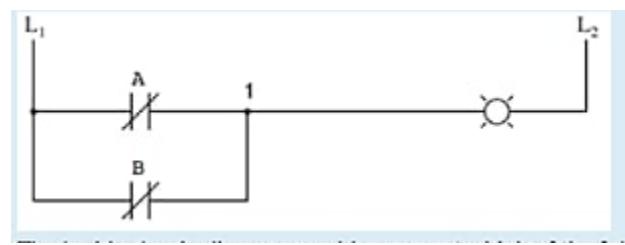
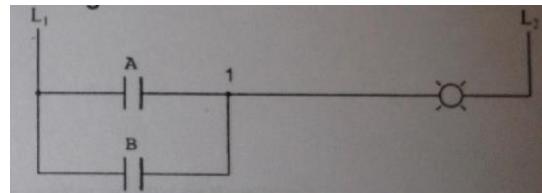
The Ladder Logic diagram would represent which of the following?

- a) XOR
- b) OR
- c) NAND
- d) AND
- e) OPEN CIRCUIT



The Ladder Logic diagram would represent which of the following?

- a) XOR
- b) OR
- c) NAND
- d) AND
- e) OPEN CIRCUIT



**NOR**

What is the decimal value of C1 1C 00 00 in IEEE single precision FP format?

- d) -9.75

What is the advantage of Assembly Language over C Language?

- a) The assembler creates much faster executable code

What is(are) the advantage(s) of C Language over Assembly Language?

- a) C is transportable to other microprocessor architectures

What is the decimal value of C5 5A 57 00 in IEEE single precision FP format?

- b) -3493.4375

PPE

On the PPE board, what number(s) on the key pad is(are) processed for an output port value of 08h and an input port value of 2Fh?

- a) 0

On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 01h and an input port value of 1Fh?

- b) 3

On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 02h and an input port value of 2fh?

- e) 5

On the PPE board, what number(s) on the key pad is(are) pressed for an output port of 04h and an input port value of 2Fh?

- c) 8

What Hex values must be sent to address the key pad rows on the PPE board?

- c) 1,2,4,8

On the raspberry pi platform, what is the programming language used?

Various open source languages

What is the numeric sequence of the key pad columns on the PPE board?

- d) 37, 2F, 1F

What is the numeric sequence to address the key pad rows on the PPE board used in the lab?

- a) 1,2,4,8

Arduino

Which of the following instruction would be used to set the LED to light on the Arduino Platform?

- e) LED = 1

On the Arduino platform, what is the programming language used?

- d) C

Microchip PICKIT

How many bits does the PIC 18 microcontroller used in the PICkit 3 Debug Express have?

- b) 16

In the PIC18 with TRISD = 0b10000000, what is the configuration of the Port D?

- a) Bit 7 of port D is set to input

In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

- b) A0

In the PIC18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

e) 0A

In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

c) Bit 7 of port D is set to output

Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?

d) 1

A “POPstruction:  
Increments the SP

With a “POPAX” instruction, what will the order of the accumulator, base, count, and data registers restored from the stack?  
ax

DAS used for BCD operation stand for which of the following?

Decimal adjust for subtraction

What is -1011.0101 in decimal?

-11.31

what is -32.75 in binary?

-100000.11000

what command in debug would be used to change the IP value?

RIP

What is -130 decimal in 2's complement (8 bits)?

01111110

what is the decimal value of C5 5A 57 00 in IEEE single precision FP format?

-3493.4375

What is -34 decimal in 2's complement?

1101 1110

EEE 174 Midterm Study Guide

S2010 MT 1

1. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string, Hello\_msg. They are:
  - a. **DS = SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 36**
2. Moore's Law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?
  - a. **Doubling every 18 – 24 months**
3. Given:  
AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

What will the IP value be after a “t” command is executed in DOS Debug?

a. **010B**

4. A “NOP” instruction will:

**a. Perform a No Operation**

5. Given:

```
AX=F247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72:0106 EB0F JMP 0118
```

What will the IP value be after a “t” command is executed in DOS Debug?

**a. 0118h**

6. In x86 architecture, BIU stands for which of the following?

**a. Bus Interface Unit**

7. 7. Determine the contents of register BL after the following instructions have been executed:

**a. 2EH**

8. 8. The number of bytes in a double word are:

**a. 4**

9. In x86 architecture, ALU stands for which of the following?

**a. Arithmetic Logic Unit**

10. The “LOOPNE” instruction is equivalent to which of the following instructions?

**a. DEC CX, JNE**

11. The instruction MOV CX, DADD is what addressing mode?

**a. Immediate**

12. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15. All of the instructions are a word long. The third instruction operator is:

**a. INT**

13. Which of the following DOS Debug instructions would set a break point at memory location 010C?

**a. G = 100 10C**

14. How many bytes are there in this short sequence of code? B400CD164CCD21

**a. 7**

15. How many address lines would be required to address 128 MB directly?

**a. 27**

16. What are the contents of CX after this program has been run:

**a. D800h**

17. 1010 0110 in 2's complement equals \_\_\_\_ in base 10.

**a. -90**

18. What is the hexadecimal encoding for adding AX with BX and storing the result in AX?

**a. 01D8**

19. What is 11.4375<sub>10</sub> in binary?

**a. 001011.01110**

20. If CX is 0000 what will CX be after a “LOOP” instruction?

**a. FFFF**

21. Given:

```
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO NC  
1D72:010F 7D08 JGE 0115
```

How many bytes will the processor jump if the conditions for a jump were met?

**a. 8**

22. What command in DEBUG would be used to execute interrupts?

**a. P**

23. What is the advantage of C Language over Assembly Language?

- a. C is transportable to other microprocessor architectures**
24. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:  
**a. OF = 1, CF = 0**
25. In MASM, with a “MOV CX, 36” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
**a. 36**
26. Given:  
AX=0353 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JNL 0116  
**a. -10**
27. AND’ing 1FH and 20H will result in which of the following?  
**a. 0**
28. Given:  
13A7:0110 CD 20 32 20 54 48 39 73-20 69 73 20 74 68 65 20  
13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9  
13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00  
An input buffer is at memory location 0115, what is the size of the buffer in decimal?  
**a. 72**
- An input buffer is at memory location 0114, what is the size of the buffer in decimal?  
**a. 84**
29. A microprocessor with a 31-bit address bus could access how much memory?  
**a. 2 GB**
30. For the instruction sequence below, determine the contents of the register AL after this program is executed:  
**a. 51H**

31. Given:  
AX=FF00 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

- What is the signed decimal value of the number in the AX register?  
**a. -256**
32. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:  
**a. 32, 32, 13, 10, 36**

### Spring 2012

33. What is the advantage of assembly language over C language?  
**a. The assembler creates much faster executable code**
34. How many lines would be required to address 128 mb directly?  
**a. 27**
35. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?  
**a. 020E**
36. Given  
W e l c o m e \_ t o \_ A s s e m  
0B0E:0200 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D  
0B0E:0210 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00  
b l y \_ L a n g u a g e

- An Ascii message begins at memory location 0200, what is the message?  
**a. Welcome to Assembly Language**
37. Which of the following DOS Debug instructions would be used to change the IP register to 0110?

- a. RIP
38. Moore's law has accurately predicted the growth rate in the number of transistors per ide for the last 40 years. What is the rate?
- a. Doubling every 18 – 24 months
39. Which of the following DOS debug instructions would set a break point at memory location 010C?
- a. G = 1000 10C
40. AND'ing 1FH and 02H will result in which of the following?
- a. 02
41. If CX is 0000 what will CX be after a "LOOP" instruction?
- a. FFFF
42. The number of bits in a word are:
- a. 16
43. The number of bits in a double word are:
- a. 32
44. In x86 architecture, ALU stands for which of the following?
- a. Arithmetic Logical Unit
45. Given:

```

AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY
1D72:0109 7D06 JGE 0118

```

What will the IP value be after a "t" command is executed in DOS DEBUG?

- a. 010B
46. What is the number 1010.0101<sub>2</sub> in decimal?
- a. 10.31
47. How many cores does the propeller microcontroller have?
- a. 8
48. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?
- a. 7DF2
49. What command in debug would be used to step through a program line by line?
- a. T
50. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:
- a. INT
51. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?
- a. 24
52. Given:

```

AX=FFEO BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 TP=010D NV UP EI PL NZ NA PO NC
1D72:010D 7DF6 JNL 011F

```

How many bytes in decimal will the processor jump if the conditions for a jump were met?

- a. 10
53. Determine the contents of the register BL after the following instructions have been executed:
- a. E2H

| Program Listing |
|-----------------|
| MOV BL, E2H     |
| MOV CL, 08H     |
| ROL BL, CL      |

54. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- a. 20, 20, 0D, 0A, 24
55. AND'ing 10H and 2FH will result in which of the following?
- a. 0
56. A "NOP" instruction in a program will
- a. Perform a No Operation
57. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 NV UP EI PL NZ NA PO CY  
 1D72:0109 7D06 JGE 0118

What is the signed decimal value of the number in the AX register?

- a. -16
58. How many address lines would be required to address 64 MB directly?
- a. 26
59. What command in debug would be used to execute interrupts?
- a. P
60. What high level language is the propeller programmed in?
- a. Spin
61. Which of the following DOS debug instructions would be used to change the IP register to 010C?
- a. RIP
62. The acronym PWM used in Parallax Propeller and MicroChip PIC18, is defined as:
- a. Pulse width modulation
63. Which command would you use to execute another core in the propeller microcontroller?
- a. Cognew
64. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
 1D72:0111 EB08 JMP 0119

What will the IP value be after the “t” command is executed in DOS debug?

- a. 0119h
65. The instruction MOV CD, DADD is what addressing mode?
- a. Immediate
66. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?
- a. F0
67. How many byte in decimal will the processor jump if the conditions for a jump are met?
- a. 24
68. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?
- a. Sets the Propeller pins P4 through P9 as output pins

69. What are the contents of DX after this program has been run:
- a. FFF0h.

| Memory Location | Contents |
|-----------------|----------|
| 5514            | 24       |
| 5513            | D8       |
| 5512            | 00       |
| 5511            | 21       |
| 5510            | 00       |

70. The number of nibbles in a word are:
- a. 4
71. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?
- a. Create 10 second delay
72. The “LOOPNE” instruction is equivalent to which of the following instructions?
- a. DEC CX JNE
73. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI NG NZ NA PO NC  
 1D72:010D EB07 JMP 0114

How many bytes will the processor jump if the conditions for a jump were met?

- a. 7
74. On the Arduino platform, what is the programming language used?
- a. C
75. What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?
- a. A12008
76. Which of the following would be used to set the TRISA register to control the direction of PIC18 Port to input?

- a. 1
77. The acronym ADC in microcontrollers stands for which of the following?  
 a. Analog to Digital converter
78. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be:  
 a. OF = 1, CF = 0
79. What is the advantage of Assembly Language over C Language?  
 a. **The Assembler creates much faster executable code**
80. How many address lines would be required to address 128 MB directly?  
 a. 27
81. In the x86 lab part Hello MASM program in the original code, what is the address of the byte used to start number in the sequence “Hello World 0”?  
 a. **020E**
82. Given:  
 0B0E : 0200        57 65 6C 63 F 6D 65 20-74 6F 20 41 73 73 65 6D  
 0B0E: 0210        62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00  
 An ASCII message begins at memory location 0200, what is the message?
- a. **Welcome to Assembly Language**
83. Which of the following DOS Debug instructions would be used to change the IP register to 0110?  
 a. **RIP**
84. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is the rate?  
 a. **Doubling every 18-24 months**
85. Which of the following DOS Debug instructions would set a break point at memory location 010C?  
 a. **G = 100 10C**
86. AND’ing 1FH and 02H will result in which of the following?  
 a. **02**
87. If CX is 0000 what will CX be after a “LOOP” instruction?  
 a. **FFFF**
88. The number of bits in a word are:  
 a. **16**
89. In x86 architecture, ALU stands for which of the following?  
 a. **Arithmetic Logic Unit**
90. Given:  
 AX=0353 BX=0534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 0V UP EI PL NZ NA PO CY  
 1D72:0109 7D06            JGE                      0118  
 What will the IP value be after a “t” command is executed in DOS Debug?  
 a. **010B**
91. What is the number, 1010.0101<sub>2</sub> in decimal?  
 a. **10.31**
92. How many cores does the propeller microcontroller have?  
 a. **8**
93. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?  
 a. **7DF2**
94. What command in DEBUG would be used to step through a program line by line?  
 a. **T**
95. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The third instruction operator is:  
 a. **INT**
96. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
 a. **24**
97. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI PL NZ NA PO NC  
1D72:010D 7DF6 JN1 0116

How many bytes in decimal will the processor jump if the conditions for a jump were met?

a. **10**

98. Determine the contents of register BL after the following instructions have been executed:

a. **E2H**

99. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

a. **32, 32, 13, 10, 36**

100. AND'ing 10H and 2FH will result in which of the following?

a. **0**

101. A “NOP” instruction in a program will:

a. **Perform a No Operation**

102. Given:

AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72  
SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
ID72:0109 7D06 JGE 0118

What is the signed decimal value of the number in the AX register?

• **-16**

103. How many address lines would be required to address 64 MB directly?

• **26**

104. What command in DEBUG would be used to execute interrupts?

• **P**

105. What high level language is the propeller programmed in?

• **Spin**

106. Which of the following DOS Debug Instructions would be used to change the IP register to 010C?

• **RIP**

107. The acronym PWM used in the Parallax Propeller and MicroChip PIC18, is defined as:

• **Pulse Width Modulation**

108. Which command would you use to execute another core in the propeller microcontroller?

• **Cognew**

109. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
1D72:0111 ED08 JMP 0119

What will the IP value be after a “t” command is executed in DOS Debug?

• **0119h**

2. The instruction MOV CX, DADD is what addressing mode?

• **Immediate**

110. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

• **A0**

111. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010F NV UP EI NG NZ NA PO NC  
1D72:010F 7D18 JGE 0128

How many bytes in decimal will the processor jump if the conditions for a jump are met?

• **24**

112. In the Propeller microcontroller, the command “dira[4..9] := %111111” would cause the processor to do which of the following?

• **Sets the Propeller pins P4 through P9 as output pins**

113. What are the contexts of DX after this program has been

```

MOV DX, 11h
MOV CX, [5512]
MOV BX, 5511h
SUB DX, [BX]
AND BX, FFFF

```

| Memory Location | Contents |
|-----------------|----------|
| 5514            | 24       |
| 5513            | D8       |
| 5512            | 00       |
| 5511            | 21       |
| 5510            | 00       |

run:

- **FFF0h**

114. The number of nibbles in a word are:

- **4**

115. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?

- **Create 10 second delay**

116. The “LOOPNE” instruction is equivalent to which of the following instructions?

- **DEC CX, JNE**

117. Given:

```

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D NV UP EI NG NZ NA PO NC
1D72:010D EB07      JMP          0114

```

How many bytes will the processor jump if the conditions for a jump were met?

- **7**

118. On the Arduino platform, what is the programming language used?

- **C**

119. What is the hexadecimal encoding for loading AX with a word (value) from the memory location 0820h?

- **A12008**

120. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?

- **1**

121. The acronym ADC in microcontrollers stands for which of the following?

- **Analog to Digital Converter**

122. In adding 5+5 through a 4 bit integer unit. The state of the OF and the CF flags after the add instruction would be:

- **OF=1, CF=0**

123. Given:

|           |                                                       |
|-----------|-------------------------------------------------------|
| 0B0E:0200 | 57 65 6C 63 6F 6D 65 20-74 6F 20 74 68 65 20 66       |
| 0B0E:0210 | 69 72 73 74 20 64 61 79-20 6F 66 20 74 68 65 20       |
| 0B0E:0220 | 72 65 73 74 20 6F 66 20-79 6F 75 72 20 6C 69 66       |
| 0B0E:0230 | 65 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |

An ASCII message begins at memory location 0200, what is the message?

- **Welcome to the first day of the rest of your life**

124. The instruction MOV DX, BADD is what addressing mode?

- **Immediate**

125. Which of the following is the hexadecimal encoding for adding BX with CX and storing the result in CX?

- **03CB**

126. What is the advantage of Assembly Language over C Language?

- **The Assembler creates much faster executable code**

127. What is 18.4375<sub>10</sub> in binary?

- **010010.01110**

128. For the instruction sequence below, determine the contents of the register AL after this program is executed:

- **51H**

129. In x86 architecture, ALU stands for which of the following?

- **Arithmetic Logic Unit**

130. A microprocessor with a 33-bit address bus could access how much memory?

- **8 GB**

131. What is the hexadecimal encoding for “JGE” for a jump back 12 bytes?  
**• 7DF2**
132. Given:  
AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010F NV UP EI NG NZ NA PO NC  
1D72:010F 7D18 JGE 0128  
How many bytes in decimal will the processor jump if the conditions for a jump are met?  
**a. 24**
133. Moore’s law has accurately predicted the growth rate in the number of transistors per die for the last 40 years.  
What is that rate?  
**a. Doubling every 18-24 months**
134. The number of bytes in a word are:  
**a. 2**
135. Determine the contents of register BL after the following instructions have been executed:  
**a. 2EH**
136. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?  
**a. 7**
137. What high level language is the propeller programmed in?  
**a. Spin**
138. In the Propeller microcontroller, the command “dira[4..9] := %0000000” would cause the processor to do which of the following?  
**a. Sets the Propeller pins P4 through P9 as input pins**
139. Which command would you use to execute another core in the propeller microcontroller?  
**a. Cognew**
140. How many cores does the propeller microcontroller have?  
**a. 8**
141. In the Propeller microcontroller, the command “waitcnt(clkfreq\*10 + cnt)” would cause the processor to do which of the following?  
**a. Create 10 second delay**
142. The acronym ADC in microcontrollers stands for which of the following?  
**a. Analog to Digital Converter**
143. The acronym PWM used in the Parallax Propeller and Microchip PIC18, is defined as:  
**a. Pulse Width Modulation**
144. How many bits does the PIC18 microcontroller use in the PICkit 3 Debug Express have?  
**a. 8**
145. Which of the following would be used to set the TRISA register to control the direction of the PIC18 Port to input?  
**a. 1**
146. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?  
**a. Bit 7 of port D is set to output**
147. In the PIC 18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?  
**a. 0A**
148. On the Arduino platform, what is the programming language used?  
**a. C**
149. Given:  
AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY  
1D72:0111 EB08 JMP 0119  
What will the IP value be after a “t” command is executed in DOS Debug?  
**a. 0119h**
150. How many bytes are there in this short sequence of code? B4 00 CD 16 4C CD 21 CD 20  
**a. 9**  
**The world's first microprocessor was developed in 1972 by?**  
**Intel**

**RISC stands for**

**Reduce instruction set computer**

151. In x86 architecture, BIU stands for which of the following?
- Bus Interface Unit**
152. Here is a short sequence of code: 74 13 EB A3 CD 16 7D 21 3C 04 EB F0 EB 15. All of the instructions are a word long. The fifth instruction operator is:
- CMP**
153. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:
- 32, 32, 13, 10, 36**
154. A “NOP” instruction in a program will:
- Perform a No Operation**
155. Given:  
AX=FFF0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118  
What is the signed decimal value of the number in the AX register?
- 16**
156. Which of the following DOS Debug instructions would set a break point at memory location 010C?
- G = 100 10C**
157. In adding 5+7 through a 4 bit integer unit, the state of the OF and CF flags after the add instruction would be:
- OF = 0, CF = 0, ZF=0**
158. Given:  
AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV UP EI NG ZR NA PO NC  
1D72:010D 7509 JNZ 0116  
What will the IP value be after a “t” command is executed in DOS Debug?
- 010Fh**
159. What are the contents of DX after this program has been run?
- FFF0h**
160. Which of the following DOS Debug instructions would be used to change the IP register to 010C?
- RIP**
161. What is the number, 1010.0101<sub>2</sub> in decimal?
- 10.31**
162. What command in DEBUG would be used to step through a program line by line?
- T**
163. AND’ing 1FH and 02H will result in which of the following?
- 02**
164. How many address lines would be required to address 64 MB directly?
- 26**
- If you want to use a INT software interrupt function to print a string out to the screen....  
Ah = 09h, ds:dx, "\$", 21h

S2010 MT2

165. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?  
a. 7
166. If CX is 0003, what will CX be after a “LOOP” instruction?  
b. 0002  
If CX is 0002, what will CX be after a “LOOP” instruction?  
C. 0001
167. IN the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following?  
d. A 5 second delay
168. What is the number 1011.0101<sub>2</sub> in decimal?  
e. 11.31
169. This section of memory represents a stack. What type of program is this?  
BEEF : FFD0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  
BEEF : FFE0 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
BEEF : FFF0 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF
- f. COM program
170. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack?  
g. BDCA
171. Determine the contents of register BL after the following instruction have been executed:  
h. 2EH
172. What Hex values must be sent to address the key pad rows on the PPE board?  
i. 1, 2, 4, 8
173. What is the number 32.4375<sub>10</sub> in binary?  
j. 100000.01110
174. In MASM, with a “MOV CX, 18h” instruction, and a “LOOP” instruction, in decimal how many times will the program loop?  
k. 24
175. The acronym PWM used for motor control, is defined as which of the following?  
l. Pulse Width Modulation
176. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?  
m. Bit 7 of port D is set to output
177. Given the short code, what is the value in AX after the program is run?  
n. 0500
178. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not?  
o. ZF
179. How many nibbles are in double precision IEEE floating point format number?  
p. 16
180. What type of program is this?  
AX=0000 BX=0000 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1476 ES=1576 SS=1376 CS=1D72 IP=0015 NV UP EI PL NZ NA PO NC  
1376:0015 0100 ADD [BI + SI], AL DS : 0000=CD

- q. EXE**  
 181. If the SP is F00F, what is the SP value after a “PUSH CX” instruction?  
**r. F00D**
182. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?  
**s. A0**
183. What is the numeric sequence of the key pad columns on the PPE board?  
**t. 37, 2F, 1F**
184. What is -130 decimal in 2’s complement (8bits)?  
**u. 01111110**
185. Which of the following is a valid x86 command for multiplying a number?  
**v. MUL BX**
186. The number of bits in single precision IEEE floating point format are:  
**w. 32**
187. You are trying to rebuild a HELLO project program in MASM and you get the following error:  
 “LINK : warning L4021: no stack segment”.  
 What would be the reason for such an error?  
**x. No project template for COM was selected**
188. A “pull down” resistor is used in digital circuits to do what?  
**y. To keep the signal line “tied” low until the line is active (goes high)**
189. A “POP” instruction:  
**z. increments the SP**
190. AND’ing 10H and 2FH will result in which of the following?  
**aa. 0**
191. In the Propeller microcontroller, the command “dira[4..9] := %1111111” would cause the processor to do which of the following?  
**bb. Sets the Propeller pins P4 through P9 as output pins**
192. What commands in MASM-CodeView would be used to step through a program line by line?  
**cc. T (F8)**
193. If the SP is F00F, what will the SP value be after a “POP SP” instruction?  
**dd. F011**
194. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh?  
**ee. 0**
195. In the Propeller microcontroller, the term “Method” is(are) which of the following?  
**ff. A block of executable commands that has variables, can receive parameters, and returns a value.**
196. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal?  
**gg. MOV CX, 30H        or        MOV CX, 48**
197. A “pull up” resistor is used in digital circuits to do what?  
**hh. To keep the signal “tied” high until the line is active (goes low)**
2. 1010 0110 in 2’s complement equals in base 10 → **\_90**
  - 3.
  4. A microprocessor with a 32-bit address bus could access how much memory → **4GB**
  5. A microprocessor with a 24-bit address bus could access how much memory → **16MB**
  - 6.
  7. A6 in 2’s complement equals in base 10 → **\_90**
  - 8.
  9. A “POP” instruction: → **Increments the SP**
  10. A “PUSH” instruction → **decrements the SP**
  11. A “NOP” instruction in a program will → **Perform a No Operation**
  12. AND’ing 1FH and 02H will result in which of the following → **\_02**
  13. AND’ing 10H and 2FH will result in which of the following → **\_0**

14. A “pull down” resistor is used in digital circuits to do what? → To keep the signal line “tied” low until the line is active (goes high)
- 15.
16. A “pull up” resistor is used in digital circuits to do what → To keep singal “tied” high until the line is active (goes low)
17. Determine the contents of register BL after the following instructions have been executed: → **E2H**
18. MOV BL, E2H
19. MOV CL, 1000b
20. Mov CL, SI is what addressing mode **REGISTER**
21. ROL BL, CL ; rotate BL by 8 to left = same value
- 22.
23. For the instruction sequence below, determine the contents of the registers AL after this program → **51H**
24. MOV AL, 73h ; 0111 0011
25. ADD AL, 78h ; 0111 1000
26. DAA
27. if low nibble of AL > 9 or AF = 1 then:
28. AL = **AL + 6**
29. AF = 1
30. if AL > 9Fh or CF = 1 then:
31. AL = **AL + 60h**
32. CF = 1
- 33.
34. For the instruction sequence below, determine the contents of the register AL after this program is executed → **51H**
- MOV AL, 83h
- ADD, AL, 68h
35. DAA
- 36.
37. Given the short code, what is the value in AX after the program is run → **0100**
- Mov BX, 0100
- PUSH BX
- MOV AX, 0500
- POP AX
- 38.
39. Given the short code, what is the value in AX after the program is run → **0001**
- MOV BX, 0001
40. PUSH BX
41. MOV AX, 0500
42. POP AX
- 43.
44. GIVEN: IP = 0106 Flags: NV UP EI NG NZ NA PE NC Instruction: JMP 011F
45. What will the IP value be after “t” command is executed in DOS Debug? → **011FH** (Unconditional Jump)
- 46.
47. GIVEN: IP = 0109 Flags: OV UP EI PL NZ NA PO CY Instruction: JGE 0118
48. What will the IP value be after a “t” command is executed in DOS Debug → **010B**
49. (0109 + 0010 add two bytes)
- 50.
51. GIVEN: IP=FFE0 Flags: OV UP EI PL NZ NA PO NC Instruction: JGE 0116: ID72:010D 7D**09**
52. How many bytes will the processor jump if the condition for a jump were met? → **2**
- 53.
- 54.
55. GIVEN: AX= FFF0 IP = 0109 FALGS: OV UP EI PL NZ NA PO CY ID72: 010F **7D18** Instruction: JGE 0118.  
What is the signed decimal value of the number in the AX register? → **-16**
56. Covert the number 7D18 into decimal.
- 57.
58. GIVEN: IP= 010F Flags: NV UP EI NG NZ NA PO NC Instruction: JNL 0115.
59. How many bytes in decimal will the processor jump if the conditions for a jump were met → **24**
- 60.
61. GIVEN: 57 65 6C 63 6F 6D 65 20-74 6F 20 41 73 73 65 6D
62. 62 6C 79 20 4C 61 6E 67-75 61 67 65 00 00 00 00

63. An ASCII message begins at memory location 0200, what is the message? → Welcome to Assembly Language
- 64.
65. Here is a short sequence of code: 7413 EBA3 CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The third instruction operator is → INT
- 66.
67. Here is a short sequence of code: 7413 A3EB CD16 7D21 3C04 EBF0 EB15. All of the instructions are a word long. The fourth instruction operator is → JGE
- 68.
69. Here is a short sequence of code: B400 CD16 3C4A 7404 BC6A 7513. All of the instructions are two bytes long. The sixth instruction operator is → JNZ
- 70.
- 71.
72. How many cores does the propeller microcontroller have → 8
- 73.
74. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 15 → 4
75. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 63 → 6
76. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 127 → 7
77. How many bits(s) is/are required to represent a range of decimal numbers from 0 to 255 → 8
78. How many bytes are there in this short sequence of code B4 00 CD 16 4C CD 20 → 7
79. How many nibbles are there in this short sequence of code B4 00 CD 16 3C 4A 74 04 3C 6A 75 13 → 24
80. How many bytes are in double precision IEEE floating point format numbers → 8
81. How many nibbles are in double precision IEEE floating point format numbers → 16
82. How many address lines would be required to address 128 MB directly → 27 ( $128 \times 1048576 = 134217728$  and  $2^{27} = 134217728$ )
83. How many address lines would be required to address 64 MB directly → 26 ( $64 \times 1048576 = 67108864$  and  $2^{26} = 67108864$ )
84. If CX is 0000, what will CX be after a “LOOP” instruction → FFFF
85. If CX is 0003, what will CX be after a “LOOPNZ” instruction → 0002
86. If the SP is F00F, what is the SP value after a “PUSH CX” instruction → F00D
87. If the SP is F00F, what is the SP value after a “POP CX” instruction → F011
88. If the SP is F00F, what is the SP value after a “POP SP” instruction → F011
89. In adding 5+5 through a 4 bit integer unit. The state of the OF and CF flags after the add instruction would be → OF = 1, CF = 0
90. In x86 architecture, BIU stands for which of the following → Bus Interface Unit
91. In x86 architecture, ALU stands for which of the following → Arithmetic Logic Unit
92. In the x86 lab part 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”? → 020E
93. In MASM, with a “MOV CX, 24h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal → 36
94. In MASM, with a “MOV CX, 24” instruction, and a “LOOP” instruction, how many times will the program loop in decimal → 24
95. In MASM, with a “MOV CX, 12h” instruction, and a “LOOP” instruction, how many times will the program loop in decimal → 18
96. In the Hello MASM lab in the original code, what is the address of the string to start the message “Hello World 0” → 0200
97. In the PIC18 with TRISD = 0b10000000, what is the configuration of the Port D → Bit 7 of port D is set to input
98. In the PIC18 with TRISD = 0b01111111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs → Bit 7 of port D is set to output (because the first bit is zero = output)
99. In the PIC18 with TRISD = 0b00001111, what is the configuration of the Port D → A0 (First 4 are outputs and last four are inputs)
100. In the PIC18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs → 0A
101. In the Propeller microcontroller, the command “dira[9..4] := %000000” would cause the processor to do which of the following → Sets the propeller pin P4 through P9 as output pins
102. In the Propeller microcontroller, the command “dira[9..4] := %111111” would cause the processor to do which of the following → Sets the propeller pin P4 through P9 as output pins

103. In the propeller microcontroller, the command “waitcnt(clkfreq\*3 + cnt)” would cause the processor to do which of the following → **A 3 second delay**
104. In the propeller microcontroller, the command “waitcnt(clkfreq\*2 + cnt)” would cause the processor to do which of the following → **A 2 second delay**
105. In the Propeller microcontroller, the term “Method” is (are) which of the following → **A block of executable commands that has variables, can receive parameters, and returns a value.**
106. Int 10h uses what function code to write a character to the screen and advance the cursor by one character position → **0Eh**
107. Int 21h, Function 09h requires three things set up before calling in order to correctly print a string:
108. **DS=SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h.**
- 109.
110. Ladder Logic is used in? → **PLCs**
111. **Moore's law** has accurately predicted the growth rate in the number of transistors per die for the last 40 years.  
What is the rate? → **Doubling every 18-24 months**
112. On the Arduino platform what is the program language used → **C**
- 113.
114. On the **PPE board**, what numbers(s) on the key pad is(are) pressed for an output port value of 08h and an input port value of 2Fh → **0**
115. On the **PPE board**, what numbers(s) on the key pad is(are) pressed for an output port value of 04h and an input port value of 2Fh → **8**
116. The “LOOPNZ” instruction is equivalent to which of the following instructions → **DEC CX, JNE**
- 117.
118. The acronym PWM used for motor control, is defined as which of the following → **Pulse Width Modulation**
119. The acronym PLC, is defined as which of the following? → **Programmable Logic Controller**
- 120.
121. The ASCII codes for space, space, carriage return, line feed, end of string in **decimal** are → **32, 32, 13, 10, 36**
122. The ASCII codes for space, space, carriage return, line feed, end of string in **hexadecimal** are: → **20, 20, 0D, 0A, 24**
123. The binary number, **1011 0101**, represents what values as a unsigned binary, 8 bit signed binary, odd parity ASCII, and BCD number (in that order) → **181, -76, 5, invalid5**
124. The binary number, **1000 0101**, represents what values as a unsigned binary, 8 bit signed binary, odd parity ASCII, and BCD number (in that order) → **133, -123, ENQ, 85**
125. The instruction MOV CX, DADD is what addressing mode → **Immediate**
126. The instruction MOV CX, [DADD] is what addressing mode → **Direct**
127. The number of bits in single precision IEEE floating point format are → **32**
- 128.
- 129.
- 130.
131. This section of memory represents a stack. What type of program is this → **EXE PROGRAM**  
**BEEF:00D0** 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  
**BEEF:00E0** 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
**BEEF:00F0** 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF
- 132.
133. This section of memory represents a stack. What type of program is this → **COM PROGRAM**  
**BEEF:FFD0** 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00  
**BEEF:FFE0** 00 01 02 03 04 05 06 07-08 09 0A 0B 0C 0D 0E 0F  
**BEEF:FFF0** 11 22 33 44 55 66 77 88-99 AA BB CC DD EE FF
- 134.
- 135.
136. Using MASM, which of the following will cause a program with a LOOP instruction to loop 48 times in decimal → **MOV CX, 48**
137. Which command would you use to execute another core in the propeller microcontroller → **Cognew**
138. What command in DEBUG would be used to change the code segment → **RCS**
139. What command in DEBUG would be used to change the IP value → **RIP**
140. What command in DEBUG would be used to execute interrupts → **P**
141. What command in MASM-CodeView would be used to step through a program line by line → **T(F8)**
142. What flag(s) does the “LOOPNZ” instruction look at to determine whether to loop or not → **ZF**
143. What flag(s) does the “LOOPNE” instruction look at to determine whether to loop or not → **ZF**
144. What Hex values must be sent to address the key pad rows on the PPE board → **1,2,4,8**

145. What is the advantage of C Language over Assembly Language → C is transportable to other microprocessor architectures
146. What is the hexadecimal encoding for “JGE” for a jump back 10 bytes → 7DF2
147. What is the hexadecimal encoding for adding AX with BX and storing the result in AX → 01D8
148. ADD AX, BX 000 00W 11 reg1 reg2
149. What is the hexadecimal encoding for adding BX with DX and storing the result in BX → 01D3
- 150.
151. What is -130 decimal in 2's complement (8bits) → 0111110
152. What is -32.75 in a base two number system → -100000.110000
153. What is 16.4375 in binary → 010000.01110
154. What is the binary value of decimal 12.875 → 1100.1110
- 155.
156. What is 16.4375 in binary → 010000.01110
- 157.
158. What is number, 1011.0101 (2) in decimal? → 11.31

What is the hexadecimal encoding for lading AX with a word (value from memory location 0820h

**Question 121**  
Answer saved  
Marked out of 1  
Flag question

What is the hexadecimal encoding for loading AX with a word (value) from memory location 0820h?

Select one:

a. 8A262008  
 c. 8B162008 ← Correct  
 d. A10820  
 e. A00820

- 159.
160. What is the **numeric sequence** of the key pad columns on the PPE board → 37,2F,1F
161. What is the decimal value of C5 5A 57 00 in IEEE single precision FP format → -3493.4375
162. What of the following instruction would be used to set the LED to light on the Arudino platform → digitalWrite(ledPin, HIGH);
- 163.
164. What type of program is this → **EXE**  
 IP = **0115**, 1376:0115 0100 ADD [BX+SI], AL DS:0000=CD
- 165.
166. What type of program is this → **COM**  
 IP = **0100**, 1376:0100 0100 ADD [BX+SI], AL DS:0000=CD
- 167.
168. Which of the following DOS Debug instructions would set a break point at memory location 010C → G = 100  
10C
- 169.
170. Which of the following would be used to set the **TRISA** register to control the direction of **PIC18** port to **input** → 1 and for **output** its → 0
- 171.
172. Which of the following DOS Debug instructions would be used to change the IP register to 110 → RAX = 0110
173. What is 458752.00 Converted to double precision FP?  
 a. 41 1c 00 00 00 00
174. Which of the following will cause a program with a LOOP instruction to loop 48 times (decimal) → **CX=30h**
- 175.
176. Which of the following is a valid x86 command for multiplying a number → **MUL BX**
177. Which of the following is **not a valid** command for a number into a register in MASM → **MOV AX, BADH**

178. With a POP BX instruction, what will be order off the accumulator, base, count, and data registers restored from the stack → **BX**
179. With a POPA instruction, what will be the order of the accumulator, base, count, and data registers restored from the stack → **BDCA**
180. You are trying to rebuild a HELLO program project in MASM and you get the following error: "ERROR 4 line 1". What is the cause of the error? → **Not known—this error by itself isn't a problem, press the enter key to clear the error.**
- 181.
182. You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK : warring L4021: no stack segment". What would be the reason for the such an error → **No project template for COM was selected.**
- 183.
184. You are typing to rebuild a HELLO project program in MASM and you get the following error: "LINK : fatal error L1089: HELLO.lrf: cannot open response file". What would be the reason for the such an error → **No source file is identified(no .asm file)**
- 185.
- 186.
- 187.
- 188.
- 189.
190. **PPE Row Column Scan decoding**
- |                                           |
|-------------------------------------------|
| D7 D6 D5 D4 D3 D2 D1 D0                   |
| S7 S6 S5 S4 S3 S2                         |
| ^ ^ ^                                     |
| 3 2 1                                     |
| 0 0 0 0 1 0 0 0 = 08h                     |
| 0 0 0 1 0 0 0 0 = 10h                     |
| 0 0 1 0 0 0 0 0 = 20h                     |
| 0 0 1 1 1 1 1 1 = 3Fh -> Nothing pressed  |
| 0 0 1 1 0 1 1 1 = 37h -> Number 1 pressed |
| 0 0 1 0 1 1 1 1 = 2Fh -> Number 2 pressed |
| 0 0 0 1 1 1 1 1 = 1Fh -> Number 3 pressed |
- 191.

## Flags

The complete set of possible flag mnemonics in Debug (ordered from left to right) are as follows:

| <i>Set</i>              | <i>Clear</i>             |
|-------------------------|--------------------------|
| OV = Overflow           | NV = No Overflow         |
| DN = Direction Down     | UP = Direction Up        |
| EI = Interrupts Enabled | DI = Interrupts Disabled |
| NG = Sign Flag negative | PL = Sign Flag positive  |
| ZR = Zero               | NZ = Not Zero            |
| AC = Auxiliary Carry    | NA = No Auxiliary Carry  |
| PO = Odd Parity         | PE = Even Parity         |
| CY = Carry              | NC = No Carry            |

## Jump information

| Mnemonic                   | Condition                                             |
|----------------------------|-------------------------------------------------------|
| <i>Signed Operations</i>   |                                                       |
| JG/JNLE                    | Greater/not less or equal $((SF \oplus OF) + ZF) = 0$ |
| JGE/JNL                    | Greater or equal/not less $(SF \oplus OF) = 0$        |
| JL/JNGE                    | Less/not greater or equal $(SF \oplus OF) = 1$        |
| JLE/JNG                    | Less or equal/not greater $((SF \oplus OF) + ZF) = 1$ |
| JO                         | Overflow $(OF = 1)$                                   |
| JS                         | Sign $(SF = 1)$                                       |
| JNO                        | Not overflow $(OF = 0)$                               |
| JNS                        | Not sign $(SF = 0)$                                   |
| <i>Unsigned Operations</i> |                                                       |
| JA/JNBE                    | Above/not below or equal $(CF \oplus ZF) = 0$         |
| JAE/JNB                    | Above or equal/not below $(CF = 0)$                   |
| JB/JNAE                    | Below/not above or equal $(CF = 1)$                   |
| JBE/JNA                    | Below or equal/not above $(CF \oplus ZF) = 1$         |
| <i>Either</i>              |                                                       |
| JC                         | Carry $(CF = 1)$                                      |
| JE/JZ                      | Equal/zero $(ZF = 1)$                                 |
| JP/JPE                     | Parity/parity even $(PF = 1)$                         |
| JNC                        | Not carry $(CF = 0)$                                  |
| JNE/JNZ                    | Not equal/not zero $(ZF = 0)$                         |
| JNP/JPO                    | Not parity/parity odd $(PF = 0)$                      |

Jumps always start with 7...

Jcc – Jump if Condition is met, (see conditional jump instructions table Ch4 Uffenbeck)  
 8-bit displacement

|            |             |
|------------|-------------|
| 0111 ttt n | byte offset |
|------------|-------------|

Full displacement

|           |            |             |
|-----------|------------|-------------|
| 0000 1111 | 1000 ttt n | word offset |
|-----------|------------|-------------|

| Hex      | t        | t        | t        | n        | Flag Test          | unsigned | signed   | Other |
|----------|----------|----------|----------|----------|--------------------|----------|----------|-------|
| <b>0</b> | <b>0</b> | <b>0</b> | <b>0</b> | <b>0</b> | OF = 1             |          | JO       |       |
| <b>1</b> | <b>0</b> | <b>0</b> | <b>0</b> | <b>1</b> | OF = 0             |          | JNO      |       |
| <b>2</b> | <b>0</b> | <b>0</b> | <b>1</b> | <b>0</b> | CF = 1             | JB, JNAE |          |       |
| <b>3</b> | <b>0</b> | <b>0</b> | <b>1</b> | <b>1</b> | CF = 0             | JNB, JAE |          |       |
| <b>4</b> | <b>0</b> | <b>1</b> | <b>0</b> | <b>0</b> | ZF = 1             | JE, JZ   | JE, JZ   |       |
| <b>5</b> | <b>0</b> | <b>1</b> | <b>0</b> | <b>1</b> | ZF = 0             | JNE, JNZ | JNE, JNZ |       |
| <b>6</b> | <b>0</b> | <b>1</b> | <b>1</b> | <b>0</b> | CF = 1 or ZF = 1   | JBE, JNA |          |       |
| <b>7</b> | <b>0</b> | <b>1</b> | <b>1</b> | <b>1</b> | CF = 0 and ZF = 0  | JNBE, JA |          |       |
| <b>8</b> | <b>1</b> | <b>0</b> | <b>0</b> | <b>0</b> | SF = 1             |          | JS       |       |
| <b>9</b> | <b>1</b> | <b>0</b> | <b>0</b> | <b>1</b> | SF = 0             |          | JNS      |       |
| <b>A</b> | <b>1</b> | <b>0</b> | <b>1</b> | <b>0</b> | PF = 1             |          | JP       |       |
| <b>B</b> | <b>1</b> | <b>0</b> | <b>1</b> | <b>1</b> | PF = 0             |          | JNP      |       |
| <b>C</b> | <b>1</b> | <b>1</b> | <b>0</b> | <b>0</b> | SF ≠ OF            |          | JNGE, JL |       |
| <b>D</b> | <b>1</b> | <b>1</b> | <b>0</b> | <b>1</b> | SF = OF            |          | JGE, JNL |       |
| <b>E</b> | <b>1</b> | <b>1</b> | <b>1</b> | <b>0</b> | ZF = 1 or SF ≠ OF  |          | JNG, JLE |       |
| <b>F</b> | <b>1</b> | <b>1</b> | <b>1</b> | <b>1</b> | ZF = 0 and SF = OF |          | JG, JNLE |       |

Examples:

JMP unconditional jump (same argument)  
 1110 1011: byte displacement

(EB   )

8bit-> +127 forward to -128 bytes  
 backward, expressed in Hex

1110 1001: word displacement      (E9   )

JCXZ jump if CX = 0

1110 0011: byte displacement      (E3   )  
*(to work with ECX use address size prefix)*

LOOP

1110 0010: byte displacement      (E2   )

*Include auto-decrement of the CX register.  
 Jump if CX is not zero after decrement.*

Link to a converter

<http://www.exploringbinary.com/floating-point-converter/>

2. In X86 architecture, ALU stands for which of the following? Arithmetic Logic Unit

3. The number of nibbles in a word are:

$$16/4 = 4$$

4. The instruction MOV BX, [2BAD] is what addressing mode? Direct

5. What is the hexadecimal encoding for adding DX with BX and storing the result in BX?

???

6. A microprocessor with a 32-bit address bus could access how much memory?  $2^4=16MB$ ,  $2^8=256KB$ ,  $2^{32}=4GB$

7. You add 7+6 through a 4 bit integer unit. The state of the OF and CF flags after the add will be:

01111 OF=1, the sign bit has changed

+0110 CF=1, there is a carryon of bit 7

0|1101

8. Which of the following DOS Debug instructions would be used change the IP register to 100?

RIP

9. What are the contents of CX after this program has been run:

MOV DX, 11h

```
MOV CX, [5512]  
MOV BX, 5511h  
SUB DX, [BX]
```

## Memory Contents location

5513 D8

5512 00

5511 21

D800h

\*\*\*What are the contents of DX after this program has been run:

FFF0h

10. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0109 OV UP EI PL NZ NA PO  
CY

1D72:0109 7D06 JNL 0118

What is the signed decimal value of the number in the AX register? --48

11. How many bytes are there in this short sequence of code? B815B400CD16CD20

8

12. In using INT 10h to set the video code to 640x350, what value must be in the AH register?

AX  
AH|AL

00

00h

13. Moore's law has accurately predicted the growth rate in the number of

transistors per die for the last 25 years. What is that rate?

Doubling every 18--24 months

14. INT 21h, Function 09h requires three things set up before calling in order to correctly print a string,

Hello\_msg. They are:

DS=SEG Hello\_msg, DX=OFFSET Hello\_msg, Hello\_msg terminated with 24h

15. Given

13A7:0110 CD 20 32 20 54 68 69 73--20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69--64 75 65 72 6D 0D 24 D9

13A7:0130 00 C6 00 00 00 00 00--00 00 00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in decimal?

50

16. Here is a short sequence of code: 7413CD16EB157D213C04EBF0A3C6. All of the instructions are two bytes long. The third instruction operator is: EB15

JMP

17. What is the hexadecimal encoding for loading DX with a word (value) from memory location 0820h?

8B162008

18. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV UP EI PL NZ NA PO NC

1D72: 010D 7D09

JGE 0116

How many bytes will the processor jump if the condition for a jump were met? 9

19. F6 in 2's complement equals \_\_\_\_\_ in base 10.

10

20. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72  
IP=010D OV UP EI NG NZ NA PO NC 1D72: 010D 7D09 JNLE 0116

What will the IP value be after a "t" command is executed in DOS Debug?

??????

21. Determine the contents of register BH after the following instructions have executed:

70h MOV [0202], AX

22. Given:

AX=2247 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000  
DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC  
1D72: 0106 EB0F JMP 0118

What will be the IP value be after a "t" command is executed in DOS Debug? 0118h

23. Which of the following DOS Debug instructions would set a break point at memory location 010E?

G=100 10E

24. Given:

0B0E: 0200 57 65 6C 63 6F 6D 65 20--74 6F 20 41 73 73 65 6D 0B0E:0210 62 6C 79 20 4C 61  
6E 67--75 61 67 65 00 00 00 00

An ASCII message begins at memory location 0200, what is the message? Welcome to Assembly Language

25. Given:

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72  
IP=0104 NV UP EI NG NZ NA PE NC

1D72:0104 7002 JO 0118

What will the IP value be after a "t" command is executed in DOS Debug?

0106h

26. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI PL NZ NA PO CY  
1D72:0109 7D06 JNL 0118

What will the IP value be after a "t" command is executed in DOS Debug? 0110

27. The instruction MOV BX, 2BAD is what addressing mode? Immediate

28. What is the hexadecimal encoding for "JNL" for a jump back 10 bytes? 7DF4

29. How many address lines would be required to address 64MB directly?

26 / $2^{26}=67\text{mb}$ /

30. The number of nibbles in a double word are: 8

31. In using INT 10h to move the screen cursor to return on the same line, what value must be in the AX register?

0E0Dh

32. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D OV U EI PL NZ NA PO NC 1d72: 010D 7D09 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug? 010Fh

33. Which of the following DOS Debug instructions would be used change the AX register?

RAX

34. How many Bytes are there in this short sequence of code? B815B400CD168A3CCD20

10

35. Given:

AX=FFD0 DS=1D72

BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 ES=1D72 SS=1D72 CS=1D72 IP=0111 NV UP EI NG NZ NA PO CY

1D72:0111 EB08 JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug? 0119h

36. Given:

AX=FFD0 BX=3534 CX=0000 DX=0180 SP=FFEE BP=0000 SI=0000 DI=0000 DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010E OV UP EI NG NZ NA PO CY 1D72: 010E 7D06 JNLE 0118

What will the IP value be after a "t" command is executed in DOS Debug? 0118

37. How many bytes decimal will the program jump for JNB instruction, given the following?

AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1376 ES=1376 SS=1376 CS=1376 IP=0100

NV UP EI PL NZ NA PO NC

1376:0100 73E0 JNB 00E2

--32

38. Given:

AX=FFE0 BX=3534 CX=0000 DX=0180 SP=FFEE

BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=010D

OV UP EI PL NZ NA PO NC

1D72: 010D 7F09 JNLE 0118

What is the decimal value of the signed number in the AX register? --0032

39. How many address lines would be required to address 512 MB directly?  $29 / 2^9 = 29$ /

40. What is the hexadecimal encoding for adding BX with CX and storing the result in BX?

03D9

41. The binary number, 0111 1110, represents what values; in Hex, and as a BCD number?

7E, 7 invalid

42. In using INT 10h to set the video mode to 640 X 200, what value must be in the AX register?

0006h

43. what is the hexadecimal encoding for loading AH with a word from memory location 0520h?

8A262005

44. What is the hexadecimal encoding for "JGE" for a jump back 10 bytes?

7DF6

45. Given:

13A7:0110 CD 20 48 20 54 68 69 73--20 69 73 20 74 68 65 20

13A7:0120 66 69 72 73 74 20 4D 69--64 74 65 72 6D 0D 24 D9

13A7:0120 00 C6 00 00 00 00 00--00 00 00 00 00 00 00 00 00

An input buffer is at memory location 0112, what is the size of the buffer in bytes in decimal:

72

46. Given:

AX=FF47 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=1D72 ES=1D72 SS=1D72 CS=1D72 IP=0106 NV UP EI NG NZ NA PE NC

1D72:0104 7002 JNO 0118

What will the IP value be after a "t" command is executed in DOS Debug? 0118

47. Determine the contents of register AH after the following instruction have executed:

MOV BX,BA70H      MOV  
AX,47E7H

MOV BL, 0FH                          47H

48. In using INT 10h to move the screen cursor to return to the beginning of the line, what value must be in the AX register?

0E0Dh

49. Given:

1376:0110 48 61 76 65 20 61 20 67--72 65 61 74 20 53 70 72  
1376:0120 69 6E 67 20 52 65 63 65--73 73 21 00 00 00 00 00

An ASCII message begins at memory location 0110, what is the message? Have a great Spring Recess!

50. You add 1+8 through a 4 bit integer unit. The state of the OF and CF flags after the ad will be:

OF=0, CF=0

**Page 16**

51. What is the advantage of Assembly Language over C Language? The Assembler creates much faster executable code

52. In The X86 lab 3 Hello MASM program in the original code, what is the address of the byte used to start the number in the sequence "Hello World 0"?

020E

53. Which of the following DOS Debug instructions would set a break point at memory location 010C?

G = 100 10C

54. If CX is 0000 what will CX be after a "LOOP" instruction?

FFFF

55. Given:

AX=0353 BX=0534 CX=0000 DX=0180

DS= 1D72 IP=0109                          OV UP EI PL NZ NA PO CY

1D72:0109 7D06                                  (OV=1,NV=1 ZR=1, NZ=0)

010B

56. How many core does the propeller microcontroller have? 8

57. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes? 7DF2

58. Here is a short sequence of code: 7413EBA3CD167D213C04EBF0EB15.  
instructions are a word long. The third instruction operator is:

INT

59. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, in decimal how many times will the program loop?

24

60. The ASCII codes for space , space, carriage return, line feed, end of string in decimal are:

32,32,13,10,36

61. A "NOP" instruction in a program will:

62. How many address lines would be required to address 64MB directly? 26

63. What command in DEBUG would be used to execute interrupts?

64. What high level language is the propeller programmed in? Spin

65. Which of the following DOS Debug instruction would be used to change the IP register to 010C?

RIP

66. The acronym PWM used in the Parrallax Propeller and MicroChip PIC18, is defined as:

Pulse Width Modulation

67. Which command would you use to execute another core in the propeller microcontroller?

Cognew

68. Given: IP=0111 NV UP EI NG NZ NA PO CY  
1D72:0111 JMP 0119

What will the IP value be after a "t" command is executed in DOS Debug? 0119h

69. The instruction MOV CX, DADD is what addressing mode? Immediate

70. In the PIC18 with TRISD = 0b00001111 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

A0

71. In the Propeller microcontroller, the command "dira[4..9] := %111111" would cause the processor to do with of the following?

Sets the Propeller pins P4 through P9 as output pins

72. In the Propeller microcontroller, the command "waitcnt(clkfreq\*10 + cnt)" would cause the processor to do with of the following?

Create 10 second delay

73. The "LOOPNE" instruction is equivalent to which of the following instructions?

DEC CX, JNE

74. On the Arduino platform, what is the programming language used?

C

---

**1. Given:**

|           |    |    |    |    |    |    |    |       |    |    |    |    |    |    |    |
|-----------|----|----|----|----|----|----|----|-------|----|----|----|----|----|----|----|
| 0BOE:0200 | 57 | 65 | 6C | 63 | 6F | 6D | 65 | 20-74 | 6F | 20 | 74 | 68 | 65 | 20 | 66 |
| 0BOE:0210 | 69 | 72 | 73 | 74 | 20 | 64 | 61 | 79-20 | 6F | 66 | 20 | 74 | 68 | 65 | 20 |
| 0BOE:0220 | 72 | 65 | 73 | 74 | 20 | 6F | 66 | 20-79 | 6F | 75 | 72 | 20 | 6C | 69 | 66 |
| 0BOE:0230 | 65 | 00 | 00 | 00 | 00 | 00 | 00 | 00-00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |

An ASCII message begins at memory location of 0200, what is the message?

- Welcome to the first day of the rest of your life

2. The instruction MOV DX,BADD is addressing what mode?

- Immediate

3. What is the hexadecimal encoding for adding BX with CX and storing the result in CX?

- 03CB

4. What is the advantage of assembly language over C language?

- The assembler creates a much faster executable code

5. What is 18.4375 in binary?

- 010010.01110

6. For the instruction sequence below, determine the contents of the register AL after this program is executed:

| Program Listing |
|-----------------|
| MOV AL, 75h     |
| ADD AL, 76h     |
| DAA             |

- 51H

7. In x86 architecture, what does ALU stand for?

- Arithmetic Logic Unit

8. A microprocessor with a 33-bit address bus could access how much memory?

- 8 GB

9. What is the hexadecimal encoding for "JGE" for a jump back 12 bytes?

- 7DF2

**10. Given:**

|           |         |         |         |         |                         |         |         |
|-----------|---------|---------|---------|---------|-------------------------|---------|---------|
| AX=FFEO   | BX=3534 | CX=0000 | DX=0180 | SP=FFEE | BP=0000                 | SI=0000 | DI=0000 |
| DS=1D72   | ES=1D72 | SS=1D72 | CS=1D72 | IP=010F | NV UP EI NG NZ NA PO NC |         |         |
| 1D72:010F | 7D18    | JGE     | 0128    |         |                         |         |         |

How many bytes in decimal will the processor jump if the conditions for a jump are met?

- 24

11. Moore's law has accurately predicted the growth rate in the number of transistors per die for the last 40 years. What is that rate?

- Doubling every 18-24 months

12. The number of bytes in a word are

- 2

13. Determine the contents of register BL after the following instructions have been executed:

| Program Listing |
|-----------------|
| MOV BL, E2H     |
| MOV CL, 0100b   |
| ROL BL, CL      |

- 2EH

14. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 127?

- 7

15. What high level language is the propeller programmed in?

- SPIN

16. In the Propeller microcontroller, the command "dire[4...9]:=%000000" would cause the processor to do what?

- Sets the propeller pins P4 through P9 as input pins

17. Which command would you use to execute another core in the propeller microcontroller?

- Cognew

18. How many cores does the propeller microcontroller have?

- 8

19. In the Propeller microcontroller, the command "waitcnt(clkfreq\*10cnt)" would cause the processor to do what?

- Create 10 second delay

20. The acronym ADC in microcontrollers stands for what?

- Analog to Digital Converter

21. The acronym PWM used in the Parallax Propeller and MicroChip PIC18, is defined as:

- Pulse Width Modulation

22. How many bits does the PIC18 microcontroller used in the PICkit3 Debug Express have?

- 8

23. What would be used to set the TRISA register to control the direction of the register to control the direction of the PIC18 Port to input?

- 1

24. In the PIC18 with TRISD = 0b01111111, what is the configuration of the Port D?

- Bit 7 of port D is set to output

25. In the PIC18 with TRISD = 0b11110000 and LATD = 0xAA, what value will be on Port D and shown on the LEDs?

- A0

26. On the Arduino platform, what is the programming language used?

- C

**27. Given:**

```
AX=FFD0  BX=3534  CX=0000  DX=0180  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1D72  ES=1D72  SS=1D72  CS=1D72  IP=0111  NV UP EI NG NZ NA PO CY
1D72:0111 EB08      JMP     0119
```

What will the IP value be after a “t” command is executed in DOS Debug?

- 0119h

28. How many bytes are there in this short sequence of code? B400CD164CCD21CD20

- 9

29. In x86 architecture, BIU stands for what?

- Bus Interface Unit

30. Here is a short sequence of code: **7413EBA3CD167D213C04EBF0EB15**. All of the instructions are a word long. The fifth instruction operator is:

- CMP

31. The ASCII codes for space, space, carriage return, line feed, end of string in decimal are:

- 32, 32, 13, 10, 36

32. A “NOP” instruction in a program will:

- Perform a No Operation

**33. Given:**

```
AX=FFF0  BX=3534  CX=0000  DX=0180  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1D72  ES=1D72  SS=1D72  CS=1D72  IP=0109  OV UP EI PL NZ NA PO CY
1D72:0109 7D06      JNL     0118
```

What is the signed decimal value of the number in the AX register?

- -16

34. What DOS Debug instruction would set a breakpoint at memory location 010C?

- G = 100 10C

35. In adding 5+7 through a 4 bit integer unit, the state of the OF and CF flags after the add instruction would be:

- OF = 0, CF = 0

**36. Given:**

```
AX=FFD0  BX=3534  CX=0000  DX=0180  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=1D72  ES=1D72  SS=1D72  CS=1D72  IP=010D  OV UP EI NG ZR NA PO NC
1D72:010D 7509      JNZ     0116
```

What will the IP value be after a “t” command is executed in DOS Debug?

- 010Fh

37. What are the contents of DX after this program has been run:

|                | Memory Location | Contents |
|----------------|-----------------|----------|
| MOV DX, 11h    | 5514            | 24       |
| MOV CX, [5512] | 5513            | D8       |
| MOV BX, 5511h  | 5512            | 00       |
| SUB DX, [BX]   | 5511            | 21       |
| AND BX, FFFF   | 5510            | 00       |

- FFF0h

38. Which of the following DOS Debug instructions would be used to change the IP register to 010C?

- RIP

39. What is the number, 1010.0101<sub>2</sub> in decimal?

- 10.31

40. What command in DEBUG would be used to step through a program line by line?

- T

41. AND’ing 1FH and 02H will result in which of the following?

- 02

42. How many address lines would be required to address 64 MB directly?

- 26

- 
1. Given a short code, what is the value in AX after the program is run?

| Program Listing |
|-----------------|
| Mov BX, 0500    |
| Push BX         |
| Mov AX, 0100    |
| POP AX          |

- a. 0500
2. A “pull down” resistor is used in digital circuits to do what?  
a. To keep the signal line “toed” low until the line is active (goes high)
3. A “POP instruction”  
a. Increments the IP
4. Ladder Logic is used in?  
a. PLCs
5. If CX is 0001, what will CX be after a “LOOPNZ” instruction?  
a. 0000
6. If the SP is F00F, what is the SP value after a “POP CX” instruction?  
a. F011
7. The acronym PWM used for motor control, is defined as which of the following?  
a. Pulse Width Modulation
8. How many bit(s) is/are required to represent a range of decimal numbers from 0 to 9?  
a. 4
9. In the PIC18 with TRISD = 0b01000000, what is the configuration of the Port D?  
a. Bit 7 of port D is set to input
10. Which of the following is not a valid command for a number into a register in MASM?  
a. MOV AX, BADH
11. You are trying to rebuild a HELLO project program in MASM and you get the following error: “LINK : warning L4021 : no stack segment”. What would be the reason for such an error?  
a. No project template for COM was selected?
12. What flag(s) does the “LOOPNE” instruction look at to determine whether to loop or not?  
a. ZF
13. In the propeller microcontroller, the command “dira[4..9] :=%000000” would cause the processor to do which of the following?  
a. Sets the propeller pins P4 through P9 as input pins
14. What command in MASM-CodeView would be used to step through a program line by line?  
a. T (F8)
15. What is the binary value of decimal 12.875?  
a. 1100.1110
16. What is the numeric sequence of the key pad columns on the PPE board  
a. 1,2,4,8
17. In MASM, with a “MOV CX, 18h” instruction , and a “LOOP” instruction, in decimal how many times will the program loop?  
a. 24
18. A “PUSH” instruction  
a. Decrements the SP
19. In the Hello MASM lab in the original code, what is the address of the byte used to start the number in the sequence “Hello World 0”?  
a. 020E
20. In the Propeller microcontroller, the command “waitcnt(clkfreq\*5 + cnt)” would cause the processor to do which of the following  
a. A 5 second delay
21. ADD’ing 10H and 2FH will result in which of the following  
a. 3F
22. A “pull up” resistor is used in the digital circuits to do what?  
a. To keep the signal “tied” high until the line is active (goes low)
23. With POPA instruction, what will be the order of the accumulator, base count, and data registers restored from the stack?  
a. BDCA
24. In the Propeller microcontroller, the term “Method” is(are) which of the following?  
a. A block of executable commands that has variables, can receive parameters, and returns a value

25. If the SP is F00F, what is the SP value after a "PUSH DX" instruction?  
 a. F00D  
 26. The "LOOPNZ" instruction is equivalent to which of the following instructions?  
 a. JNZ, DEC CX

27. What type of program is this?

```
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=1376 ES=1376 SS=1376 CS=1376 IP=0115 NV UP EI PL NZ NA PO NC
1376:0115 0100 ADD [BX+SI],AL DS:0000=CD
```

- a. COM  
 28. What Hex values must be sent to address the key pad rows on the PPE board?  
 a. 37,2F,1F  
 29. What is -34 decimal in 2's complement (8 bits)?  
 a. 1101 1110  
 30. Given:

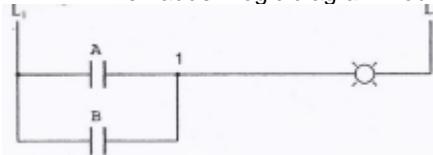
```
13A7:0110 CD 20 30 20 54 68 69 73-20 69 73 20 74 68 65 20
13A7:0120 66 69 72 73 74 20 4D 69-64 74 65 72 6D 0D 24 D9
13A7:0130 00 C6 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00
```

An input buffer is at memory location 0113, how many bytes are in the buffer (in decimal)?

- a. 30  
 31. How many global RAM does the Propeller microcontroller have?  
 a. 46KB  
 32. In the Propeller, how many values does a method return?  
 a. 1  
 33. AND'ing 10H and 2FH will result in which of the following?  
 a. 0  
 34. In the Propeller microcontroller, the command "dira[4..9] := %000000" would cause the processor to do which of the following?  
 a. Sets the Propeller pins P4 through P9 as input pins  
 35. The Ladder Logic diagram would represent which of the following?



- a. AND  
 36. Which of the following is a valid x86 command for multiplying a number?  
 a. MUL BX  
 37. The instruction MOV CX, [SI] is what addressing mode?  
 a. Register Indirect  
 38. On the PPE board, what number(s) on the key pad is(are) pressed for an output port value of 02h and an input port value of 2Fh?  
 a. 5  
 39. In MASM, with a "MOV CX, 24" instruction, and a "LOOP" instruction, how many times will the program loop in decimal?  
 a. 24  
 40. The letters "NC" labeled on relays and PLCs means which of the following?  
 a. Normally Closed  
 41. The Ladder Logic diagram would represent which of the following



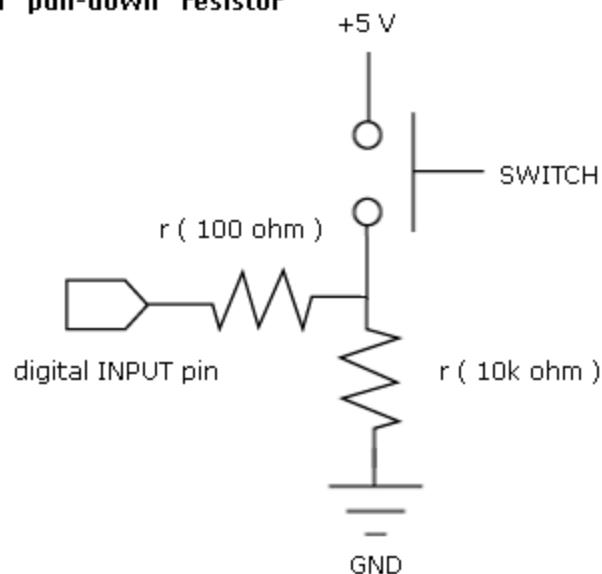
- a. OR  
 42. If you want to use a INT software interrupt function to print a string out to the screen, what is the function code, start pointer, termination character, and interrupt you need to use?  
 a. ah = 09h, ds:dx, "\$", 21h

## Midterm 2 Study Guide

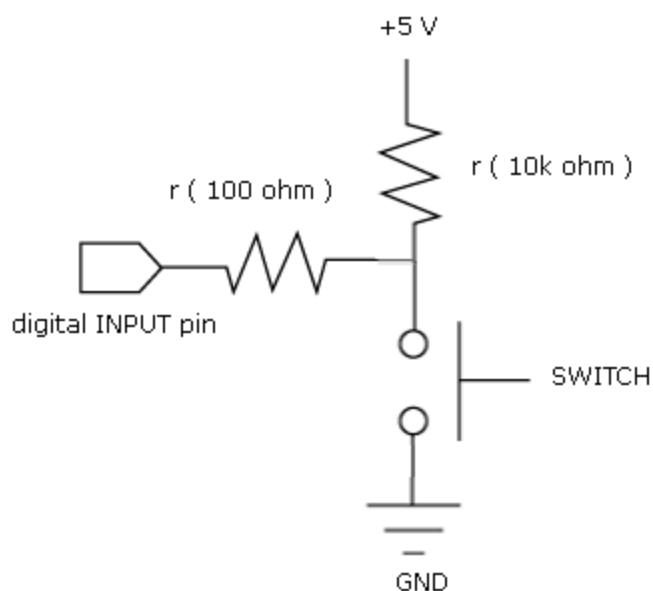
- Loop Instructions
  - Loop
    - decrements ecx and checks if ecx is not zero, if that condition is met it jumps at specified label, otherwise falls through.
  - Loop e
    - decrements ecx and checks that ecx is not zero *and* ZF is set - if these conditions are met, it jumps at label, otherwise falls through.
  - Loop z
    - will decrement CX even if the ZeroFlag is clear (no Zero Result),
  - Loop NE
    - same as LOOPE except that it requires ZF to be not set (i.e be zero) to do the jump.
  - Loop NZ
    -
- Push and Pop Instructions - Source : <http://www.penguin.cz/~literakl/intel/p.html>
  - Push A - Push All Registers onto Stack
    - Pushes all general purpose registers onto the stack in the following order: (E)AX, (E)CX, (E)DX, (E)BX, (E)SP, (E)BP, (E)SI, (E)DI. The value of SP is the value before the actual push of SP.
  - Pop A - Pop All Registers onto Stack
    - Pops the top 8 words off the stack into the 8 general purpose 16/32 bit registers. Registers are popped in the following order: (E)DI, (E)SI, (E)BP, (E)SP, (E)DX, (E)CX and (E)AX. The (E)SP value popped from the stack is actually discarded.
  - Push F - Push Flags onto Stack
    - Transfers the Flags Register onto the stack. PUSHF saves a 16 bit value while PUSHFD saves a 32 bit value.
  - Pop F - Pop Flags off Stack
    - Pops word/doubleword from stack into the Flags Register and then increments SP by 2 (for POPF) or 4 (for POPFD).

Pull up is where you place a resistor between a signal and +V, pull down is pulling it to ground.

### Switch with "pull-down" resistor



### Switch with "pull-up" resistor



- Pullup Resistor - <https://playground.arduino.cc/CommonTopics/PullUpDownResistor>
  - A pull-up resistor weakly "pulls" the voltage of the wire it is connected to towards its voltage source level when the other components on the line are inactive. When the switch on the line is open, it is high-impedance and acts like it is disconnected. Since the other components act as though they are disconnected, the circuit acts as though it is disconnected, and the pull-up resistor brings the wire up to the high logic level. When another component on the line goes active, it will override the high logic level set by the pull-up resistor. The pull-up resistor assures that the wire is at a defined logic level even if no active devices are connected to it.
- Pulldown Resistor
  - A pull-down resistor works in the same way but is connected to ground. It holds the logic signal near zero volts when no other active device is connected.

- The value of a pull down or pull up resistor will vary depending upon your specific devices involved.
  -
- Masm
  - Valid Instructions - Hex in MASM and Decimal
  - Errors
  - Code Viewer
    - Shortcuts
    - Different window
  - Difference between COM and EXE
- Bit Masking
  - Bit masks are used to access specific bits in a byte of data. This is often useful as a method of iteration, for example when sending a byte of data serially out a single pin. In this example the pin needs to change its state from high to low for each bit in the byte to be transmitted. This is accomplished using what are known as bitwise operations and a bit mask.
  - Set
  - Clear
  - Toggle
  - With PIC 32 Resistors
- PIC 32 Registers
  - LAT - to write to a pin
  - TRIS - Sets input or Output (change the direction)
  - PORT - to read from a pin
- Ladder Logic

Each rung of ladder language typically has one coil at the far right. Some manufacturers may allow more than one output coil on a rung.

- Rung Input : Checkers (contacts)
  - —[ ]— Normally open contact, closed whenever its corresponding coil or an input which controls it is energized. (Open contact at rest)
  - —[\ ]— Normally closed ("not") contact, closed whenever its corresponding coil or an input which controls it is not energized. (Closed contact at rest)
- Rung Output: Actuators (coils)
  - —( )— Normally inactive coil, energized whenever its rung is closed. (Inactive at rest)
  - —(\ )— Normally active ("not") coil, energized whenever its rung is open. (Active at rest)

The "coil" (output of a rung) may represent a physical output which operates some device connected to the programmable controller, or may represent an internal storage bit for use elsewhere in the program.

- Basic Logic
  - Truth Table
  - What Devices @ Industries use it ?
    - used for automation of typically industrial electromechanical processes, such as control of machinery on factory assembly lines, amusement rides, or light fixtures. PLCs are used in many machines, in many industries
  - What does NO and NC mean?
    - Normally Open , Normally Close
- Assembly
  - Rotate/Shift
  - MUL, Sub, ADD
  - What result If Lshift
- The Stack LIFO
- Home work
- Lecture example 5.1 5.2 5.3
- **Introduction to Microprocessors and Computer Applications,**

- -
- **Questions?**
- On material covered so far?
- Questions on Strings, "Shopper and Shopping"?
- Strings, example handouts
- Direction register, DF Flag Vol. 1 Intel manual
- CLD instruction clears the D flag, D = 0, UP
- STD instruction sets the D flag, D = 1, DN down
- In Debug / MASM , flag is set to UP (D=0), and DN (D=1).
- For example:
- -r
- AX=0000 BX=0000 CX=0000 DX=0000 SP= FFEE BP=0000 SI=0000 DI=0000
- DS =13A9 ES=13A9 SS=13A9 CS=13A9 IP =0100 NV **DN** EI PL NZ NA PO NC
- 13A9:0100 FD **STD**
- -t
- 
- AX=0000 BX =0000 CX =0000 DX =0000 SP= FFEE BP=0000 SI=0000 DI=0000
- DS =13A9 ES=13A9 SS=13A9 CS=13A9 IP =0101 NV **DN** EI PL NZ NA PO NC
- 13A9:0101 FC **CLD**
- -t
- 
- AX=0000 BX =0000 CX =0000 DX =0000 SP= FFEE BP=0000 SI=0000 DI=0000
- DS =13A9 ES=13A9 SS=13A9 CS=13A9 IP =0102 NV **UP** EI PL NZ NA PO NC
- 13A9:0102 FD **STD**
- -t
- 
- AX=0000 BX =0000 CX =0000 DX =0000 SP= FFEE BP=0000 SI=0000 DI=0000
- DS =13A9 ES=13A9 SS=13A9 CS=13A9 IP =0103 NV **DN** EI PL NZ NA PO NC
- 13A9:0103 FC **CLD**
- -t
- 
- AX=0000 BX =0000 CX =0000 DX =0000 SP= FFEE BP=0000 SI=0000 DI=0000
- DS =13A9 ES=13A9 SS=13A9 CS=13A9 IP =0104 NV **UP** EI PL NZ NA PO NC
- 13A9:0104 FD **STD**
- -t
- 
- AX=0000 BX =0000 CX =0000 DX =0000 SP= FFEE BP=0000 SI=0000 DI=0000
- DS =13A9 ES=13A9 SS=13A9 CS=13A9 IP =0105 NV **DN** EI PL NZ NA PO NC
- 13A9:0105 FC **CLD**
- -
- 
- 
- Information on the Web, <http://gaia.ecs.csus.edu/~eee174>
  - **Review of " Shopper and Shopping "**
  - CX register is not modified by CLD or the Directional Flag.
    - **Review of " Strings Attached " Program**
    - 
    - 1: Page 58, 80
    - 2: Title strings\_attached
    - 3:
    - 4: cseg segment 'code'
    - 5: assume cs : cseg , ds : cseg , es : cseg , ss : cseg
    - 6:
    - 7:
    - 8: org 100h

```

■ 9:
■ 10: start: mov ah,0
■ 11: int 16h
■ 12: mov cx ,ax
■ 13: and cx , 000fh
■ 14: mov ah, 0eh
■ 15: int 10h
■ 16: cld
■ 17: mov ax, 0e0dh
■ 18: int 10h
■ 19: mov ax, 0e0ah
■ 20: int 10h
■ 21: mov si, offset shopa
■ 22: mov di , offset shopb
■ 23: repz cmpsb
■ 24: jz good
■ 25: mov dx , offset notso
■ 26: jmp putout
■ 27:
■ 28: good: mov dx , offset fine
■ 29: putout: mov ah,09
■ 30: int 21h
■ 31: mov ah, 4ch
■ 32: int 21h
■ 33:
■ 34: org 200h
■ 35:
■ 36: shopa db "SHOPPER", 20h, 0dh
■ 37:
■ 38: shopb db "SHOPPING",0dh
■ 39:
■ 40: fine db "EQUAL",0dh, 0ah, "$"
■ 41:
■ 42: notso db "NOT EQUAL", 0dh, 0ah, "$"
■ 43:
■ 44: cseg ends
■ 45: end start
■
■
■

```

- [MASM Labels-Numbers.PDF](#)

- 
- -
- **Lab information**
- 
- Call, Return
- Stack, Pointer Addressing
- String Instructions
- Packed BCD
- 
- Recommendations:
  - \* Write a main routine which call the each of the sub routines
  - \* First set up the sub routines as dummies, just to see if it works
  - \* Test each sub routine as a separate program first then convert the program to a sub routine and test it.
  - \* Finally "glue it" all together as a main and sub routines.
- 
- Additional Suggestions and issues
- \*Focus on getting the program to work, then make it a subroutine.

- \*See "game of NIM " for examples of how to do proper subroutines.
- \*This program is complete enough to make it difficult to "hack your way through it", you need structure - you need to start with a flow chart and "build" it in sections.
- - [Lab information](#)
  - [Course web page](#) updated
  - [Subroutine Flow Charting](#)
  - **Old lab reference**
    - [Password Protection Lab](#)
    - [PPE Lab](#)
    - [Intro SX-Lab](#)
    - [SX-IO Lab](#)
    - [SX using C Lab](#)
    - [Robot - Boebot](#)
    - SEL Relays
    - Propeller
    - [The 8-Core Parallax Processor Really Spins](#)
    - 
    - [Spinning Multicore](#)
      - [PE-Lab-Setup-v1.0](#)
      - [PE-IOTimingBasics-v1.0](#)
  - [Password Protection Lab](#)
    - Hints
      - Sub routines see program 5.3
      - BCD to ASCII see program 5.2
      - Lab 1 [16bit banking program](#) converted to [BCD](#)
      - [8bit banking program](#) using BCD
    - Demo BCD.asm
  - [PPE Board Lab](#)
    - PPE example program ([PPE.ASM](#))
    - PPE sample [program](#) ([PPEtest.ASM](#))
      - Getting the ports to work in NT/2000/ XP
      - Use UserPort
      - available from my FTP site:
      - <ftp://gaia.ecs.csus.edu/dahlquist> / MASM /
      - directly from
      - <ftp://gaia.ecs.csus.edu/dahlquist> / MASM / UserPort.zip
  - 
  - 
  - 
  - Review of Last [week's Lecture](#)
  - [Banking program.exe](#)
    - [PPE](#)
      - PPE sample [program](#) , example of using subroutines ([PPEtest.ASM](#) )
      - PPE example program ([PPE.ASM](#))
      - PPE Row Column [Scan](#) decoding
      - D7 D6 D5 D4 D3 D2 D1 D0
      - S7 S6 S5 S4 S3 S2
      - ^ ^ ^
      - 3 2 1
      - 0 0 0 0 1 0 0 0 = 08h
      - 0 0 0 1 0 0 0 0 = 10h
      - 0 0 1 0 0 0 0 0 = 20h
      - 
      - 0 0 1 1 1 1 1 1 = 3Fh -> Nothing pressed
      - 0 0 1 1 0 1 1 1 = 37h -> Number 1 pressed
      - 0 0 1 0 1 1 1 1 = 2Fh -> Number 2 pressed
      - 0 0 0 1 1 1 1 1 = 1Fh -> Number 3 pressed
      - 
      - Running [PPE](#) without a PPE board
    - [Program 5.7](#) , p208

- 
- Note program development;
- Problem Statement
- Discussion, outline, Flow Chart goes here.
- Pseudo Code
- Code
- Commented Code
- 
- Prefix bytes, **PSP => Program Segment Prefix**
- 
- Program 5.7 Programmable Time Delay
- Why Delay, don't we want the processor to run as fast as it can?
- 
- Review parts of prog . 5.7:
- Mov AL, [0084]
- CMP AL, 0D <- test to see if CR is present (it should be)
- JNZ ; jump if it is not zero (if it is something other than CR)
- 
- Sub word ptr [0082], 3030 ;word ASCII strip
- Mov AL, 0082 ; tens digit
- Mov BL , 0A ; move 10 into BL , prep for multiplication instruction
- Mul BL ; multiply 10XAL and put the result in AX.
- 
- 
- **32 bit mode .386**
- Prefix bytes, 0000 - 0OFF
- Interrupts
- 
- Text pages 90-92: see figure 3.11
- Programming 32 Bit Mode Model, General Purpose Registers
- These are the data, pointer, and index registers,
- to reference 32-bit registers, use the letter "E" (for extended) in front of the 16bit register.
- 
- For example, **EAX** vs AX
- 
- see figure 3.11, pages 91 and 92
- In code view under options>preferences>32 bit registers
- Text pages 130-132: 32-Bit Mode Memory Operands,
- Mov ECX , [0200] ; will move 32 bits into ECX , 4 bytes, double word
- Remember, that the least significant byte of the data is always stored in the starting memory address,
- little endian format, "**the little end of the number is stored first**".
- 
- 32 bit instructions:
- In the assembler MASM , use a ".386" or "USE32"
- INC ECX ; 32 bit increment, 1 + ECX
- Sub EDX , EBX ; 32 bit subtract
- Mov ESI , 12345678H ; 32 bit move immediate, register load
- 
- 
- 
- **Text pages 157-162: 80x86 Interrupts**
- 
- What is an interrupt and what would they be used for?
- 
- An interrupt is an event that causes the processor to suspend its present task and transfer control to a new program, the ISR .
- ISR = Interrupt service routine
- 
- 3 sources of Interrupts; **processor** interrupts, **software** interrupts, and **hardware** interrupts
-

- Each interrupt must supply a type number or vector, which is used by the processor as a pointer into an IVT , interrupt vector table, to determine the address of that interrupt's service routine.
- IVT = interrupt vector table
- See fig 4.15, there are 256 interrupt vectors allowed 00 - FF
- Each entry in the IVT points to the location of its corresponding ISR .
- On boot up the table is empty, it must be loaded after boot.
- Before transferring control to the ISR , the processor, it saves the current program address and the flags on the stack.
- When the ISR finishes with an IRET , return from interrupt, it recovers the flags and the old program address form the stack.
- Many vectors in the IVT are reserved for the processor itself.
- You have already used some of them.
- See table 4.5, pages 190 -191
- **Processor Interrupts** (types 0 -12), these are generated by the processor itself, in response to error conditions.
- "0D" general protection fault, Blue screen of death.
- **Software Interrupts** : these are the interrupts you have been using like,
- Int 20 = terminate program
- Int 10 = BIOS video services, see page 196, table 4.6
- 
- **Hardware Interrupts** : these interrupts are requests initiated by external hardware. Two pins NMI , and INTR
- NMI - nonmaskable interrupt, it cannot be blocked. It would be use for example to shut down the system due to a memory parity error.
- INTR - Interrupt request, allows external hardware to activate any of the 256 interrupt types.
- This is done by placing the type number on the data bus when the processor acknowledges the interrupt request.
- IRQ - Interrupt request, provides physical connections between HWdevices and the interrupt controllers.
- 
- 
- 
- **Menu program in MASM**
- **Demo and discuss BCD.ASM program**
- **Reviewing Handout " Demo: Stack, Nested Subroutines "**
- 
- C:\>debug
- -a100
- 2BAD:0100 call 109
- 2BAD:0103 mov ah, 03
- 2BAD:0105 int 20
- 2BAD:0107 mov ah, 04
- 2BAD:0109 call 110
- 2BAD:010C mov ch, 01
- 2BAD:010E ret

- 2BAD:010F nop
- 2BAD:0110 mov cl ,02
- 2BAD:0112 ret
- 2BAD:0113
- -
- 
- -u100 112
- 2BAD:0100 E80600 CALL 0109
- 2BAD:0103 B403 MOV AH,03
- 2BAD:0105 CD20 INT 20
- 2BAD:0107 B404 MOV AH,04
- 2BAD:0109 E80400 CALL 0110
- 2BAD:010C B501 MOV CH,01
- 2BAD:010E C3 RET
- 2BAD:010F 90 NOP
- 2BAD:0110 B102 MOV CL,02
- 2BAD:0112 C3 RET
- -
- 
- 
- -d100 112
- 2BAD:0100 E8 06 00 B4 03 CD 20 B4-04 E8 04 00 B5 01 C3 90
- 2BAD:0110 B1 02 C3
- -
- 
- -r
- AX=0000 BX =0000 CX =0000 DX =0000 SP= FFEE BP=0000 SI=0000 DI=0000
- DS =2BAD ES=2BAD SS=2BAD CS=2BAD IP =0100 NV UP EI PL NZ NA PO NC
- 2BAD:0100 E80600 CALL 0109
- -d ffe0 ffef
- 2BAD:FFE0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 00 00 00
- .....
- -
- -t
- 
- AX=0000 BX =0000 CX =0000 DX =0000 SP= FFEC BP=0000 SI=0000 DI=0000
- DS =2BAD ES=2BAD SS=2BAD CS=2BAD IP =0109 NV UP EI PL NZ NA PO NC
- 2BAD:0109 E80400 CALL 0110
- -d ffe0 ffef
- 2BAD:FFE0 00 00 00 00 00 00 09 01-AD 2B 32 25 03 01 00 00
- .....+2%....
- -
- -
- -t
- 
- AX=0000 BX =0000 CX =0000 DX =0000 SP= FFEA BP=0000 SI=0000 DI=0000
- DS =2BAD ES=2BAD SS=2BAD CS=2BAD IP =0110 NV UP EI PL NZ NA PO NC
- 2BAD:0110 B102 MOV CL,02
- -d ffe0 ffef
- 2BAD:FFE0 00 00 00 00 10 01 AD 2B-32 25 0C 01 03 01 00 00
- .....+2%.....
- -
- -t
- 
- AX=0000 BX =0000 CX =0002 DX =0000 SP= FFEA BP=0000 SI=0000 DI=0000
- DS =2BAD ES=2BAD SS=2BAD CS=2BAD IP =0112 NV UP EI PL NZ NA PO NC
- 2BAD:0112 C3 RET
- -d ffe0 ffef



- Review parts of prog. 5.7:
- Mov AL, [0084]
- CMP AL, 0D <- test to see if CR is present (it should be)
- JNZ ; jump if it is not zero (if it is something other than CR)
- 
- Sub word ptr [0082], 3030 ;word ASCII strip
- Mov AL, 0082 ; tens digit
- Mov BL, 0A ; move 10 into BL, prep for multiplication instruction
- Mul BL ; multiply 10XAL and put the result in AX.
- 
- 
- **32 bit mode .386**
- Prefix bytes, 0000 - 00FF
- Interrupts
- 
- Text pages 90-92: see figure 3.11
- Programming 32 Bit Mode Model, General Purpose Registers
- These are the data, pointer, and index registers,
- to reference 32-bit registers, use the letter "E" (for extended) in front of the 16bit register.
- 
- For example, EAX vs AX
- 
- see figure 3.11, pages 91 and 92
- 
- In **code view** under options>preferences>32 bit registers
- 
- 
- 
- Text pages 130-132: 32-Bit Mode Memory Operands,
- 
- Mov ECX, [0200] ; will move 32 bits into ECX, 4 bytes, double word
- 
- Remember, that the least significant byte of the data is always stored in the starting memory address,
- 
- little endian format, "the little end of the number is stored first".
- 
- 
- 32 bit instructions:
- 
- In the assembler MASM, use a ".386" or "USE32"
- 
- INC ECX ; 32 bit increment, 1 + ECX
- 
- Sub EDX, EBX ; 32 bit subtract
- 
- Mov ESI, 12345678H ; 32 bit move immediate, register load
- 
- 
- 
- **Text pages 157-162: 80x86 Interrupts**
- 
- What is an interrupt and what would they be used for?
- 
- An interrupt is an event that causes the processor to suspend its present task and transfer control to a new program, the ISR.
- 
- ISR = Interrupt service routine
- 
- 3 sources of Interrupts; processor interrupts, software interrupts, and hardware interrupts
-

- Each interrupt must supply a type number or vector, which is used by the processor as a pointer into an IVT, interrupt vector table, to determine the address of that interrupt's service routine.
- IVT = interrupt vector table
- See fig 4.15, there are 256 interrupt vectors allowed 00 - FF
- Each entry in the IVT points to the location of its corresponding ISR.
- On boot up the table is empty, it must be loaded after boot.
- Before transferring control to the ISR, the processor, it saves the current program address and the flags on the stack.
- When the ISR finishes with an IRET, return from interrupt, it recovers the flags and the old program address from the stack.
- Many vectors in the IVT are reserved for the processor itself.
- You have already used some of them.
- See table 4.5, pages 160 -161
- Processor Interrupts (types 0 -12), these are generated by the processor itself, in response to error conditions.
  - "0D" general protection fault, Blue screen of death.
  - Software Interrupts: these are the interrupts you have been using like,
    - Int 20 = terminate program
    - Int 10 = BIOS video services, see page 162, table 4.6
  - Hardware Interrupts: these interrupts are requests initiated by external hardware. Two pins NMI, and INTR
  - NMI - nonmaskable interrupt, it cannot be blocked. It would be used for example to shut down the system due to a memory parity error.
  - INTR - Interrupt request, allows external hardware to activate any of the 256 interrupt types.
  - This is done by placing the type number on the data bus when the processor acknowledges the interrupt request.
  - IRQ - Interrupt request, provides physical connections between HW devices and the interrupt controllers.
  - 
  - 
  -

#### **Segment Registers, Text pages 82-87:**

- Code, Data, Stack and Extra Segment Registers: each are 64K memory blocks and are used differently by the processor.
- The four segment registers shown in Fig 3.9 page 84, CS, DS, SS, and ES are used to point at location 0, the base address of each segment.
- Remember logical and physical address (see example 3.3 page 85, - 3rd p101), remember offset segment by 4 bits (one nibble). Notice in example 3.3 both logical addresses point to the same physical address.
- This can have disastrous results when the data begins to overwrite the subroutine stack area or vice versa.

- ○ Table 3.1 Segment register assignments, indicates that some memory references can have their segment definitions changed (note the alternate segment column in the table).
- ○ Not everyone likes segmented memory, it can seem confusing, remember each has a special function (like the registers A, B, C, D.).
- ○ One advantage to having separate data and code segments is that one program can work on several different sets of data. Load DS to point to the new data.
- ○ Down side, Seg mem introduces extra complexity in both HW and SW, SW esp with 64K limits per segment.
- ○ ○
- ○ ○ **Demo and discuss BCD.ASM program**
- ○ ○ **x86 Memory Segment Refresher, Text pages 224 (275):**
- ○ ○ Segments may be separate, partially overlapped or fully overlapped.
- ○ ○ Two types of executable files (COM and EXE ) programs use these segments differently.
- ○ ○ COM programs are limited to 64K and consist of a single fully overlapped segment. In such programs, all of the segment registers are typically set to the same value. COM programs do not need a stack segment as DOS will take of it.
- ○ ○ EXE files have no file size restriction and may contain several program segments. This is good programming.
- ○ ○ Complex programs are often written in modular form with several code, data, and stack segments.
- ○ ○ **Comparing COM and EXE files, Text pages 244-246 (293-298):**
- ○ ○ All COM files begin with a 256 byte **Program Segment Prefix** , PSP. Therefore the first line of code in a COM file should include ORG 0100H (don't forget the H or MASM will assume decimal) to ensure that the program is loaded after the PSP.
- ○ ○ EXE files are loaded with 256 byte PSP , but DOS sets the DS , ES, FS, and GS segment registers to point to the base of this area, the PSP in its own data segment.
- ○ ○ EXE files require that a separate stack segment be created, DOS automatically loads register SS with the address of this segment and points register SP at the top of the segment.
- ○ ○ Program instructions are placed in the code segment at offset 0000 ( ie.IP = 0000),
- ○ ○ Table 6.3 page 246, summarize the differences between COM and EXEfiles.
- ○ ○
- ● —
- **Segment Registers , Text pages 82-87:**
- Code, Data, Stack and Extra Segment Registers: each are 64K memory blocks and are used differently by the processor.
- The four segment registers shown in Fig 3.9 page 84, CS, DS , SS, and ES are used to point at location 0, the base address of each segment.
- Remember logical and physical address (see example 3.3 page 85), remember off set segment by 4 bits (one nibble). Notice in example 3.3 both logical addresses point to the same physical address.
-

- This can have disastrous results when the data begins to overwrite the subroutine stack area or vice versa.
- 
- Table 3.1 Segment register assignments, indicates that some memory references can have their segment definitions changed (note the alternate segment column in the table).
- 
- Not everyone likes segmented memory, it can seem confusing, remember each has a special function (like the registers A, B, C, D.).
- 
- One advantage to having separate data and code segments is that one program can work on several different sets of data. Load DS to point to the new data.
- 
- Down side, Seg mem introduces extra complexity in both HW and SW, SW esp with 64K limits per segment.
  - **x86 Memory Segment Refresher, Text pages 224 (275):**
  - Segments may be separate, partially overlapped or fully overlapped.
  - Two types of executable files (COM and EXE ) programs use these segments differently.
  - COM programs are limited to 64K and consist of a single fully overlapped segment. In such programs, all of the segment registers are typically set to the same value. COM programs do not need a stack segment as DOS will take of it.
  - EXE files have no file size restriction and may contain several program segments. This is good programming.
  - 
  - Complex programs are often written in modular form with several code, data, and stack segments.
  - 
  - **Comparing COM and EXE files, Text pages 244-246 (293-298):**
  - 
  - All COM files begin with a 256 byte program segment prefix, PSP . Therefore the first line of code in a COM file should include ORG 0100H (don't forget the H or MASM will assume decimal) to ensure that the program is loaded after the PSP .
  - 
  - EXE files are loaded with 256 byte PSP , but DOS sets the DS , ES, FS, and GS segment registers to point to the base of this area, the PSP in its own data segment.
  - 
  - EXE files require that a separate stack segment be created, DOS automatically loads register SS with the address of this segment and points register SP at the top of the segment.
  - 
  - Program instructions are placed in the code segment at offset 0000 ( ie. IP = 0000),
  - 
  - Table 6.3 page 246, summarize the differences between COM and EXEfiles.
  - 
  - 
  - 
  - 
  - 
  - **Game Program " NIM "**
  - .EXE
  - Segmented code, NIM is an example of how and why you would use .EXE
  - 
  - Template
  - Fig. 6.14 page 260, Program Template (generic form, like wordprocessing ), This is good programing (and Design) practice
  - 
  - Game of NIM Program, Text Pages 248-262
  - There is a lot in this program example.
  - 
  - Note the program layout and flow from pages 250 - 258.
  - 
  - Note the proper way of documenting program, main program and subroutine organization / layout.
  - 
  - Note the use of equates for easier program structure/documentation, and the easier way to modify the program.
  - 
  - **Template**

- What is the game NIM and what does the program output look like?
- See page 298
- This is what you want to include in your program reports.
- 15 sticks, choose only 1, 2, 3, at one time, first to 0 wins.
- 
- **Program Description**
- Program outline page 250, fig. 6.12 -- read through
- Program Header and Equates, fig. 6.13a page 252
- pseudo-op equate operator, these statements allow us to assign a label to program constants, for example cr is equated to 0dh the ASCII code for a carriage return. Using this label whenever we want to place a return in an ASCII message we can simply enter " cr ".
- This only works when the context of this program.
- 
- This is like definition, or DIM dimensioning statements used by high level languages.
- Besides making the program easier to read the equate operator allows global changes to be made to the program with a single line, for example to change DELAY just change it in the equate statement area.
- 
- 
- 
- The Data Segment fig. 6.13 page 253
- All of the ASCII messages required by the game are placed here.
- Each message ends with "\$" (required by DOS function 9)
- You can use actual characters enclosed by quotes or hex (or decimal) values for ASCII.
- You can now use " cr " and "lf" from the equates.
- Notice the choice table, 15 choices are stored corresponding to 0 - 14 sticks remaining.
- 
- 
- 
- The Main Program fig. 6.13c and 6.13d pages 254 and 255
- Note the main program calls to the subroutines.
- call show,
- 
- The procedures fig 6.13e, 6.13f and 6.13g pages 256 ~ 258
- Think\_time, generates a time delay whose value is set by the Delay equate, and uses INT 15h service 86h for time delay (some problems in the lab with this in the past).
- 
- Process, displays the current number of sticks in the pile, notice that if the total is greater than 9, and ASCII "1" is first output, and then the units digit.
- 
- Show, uses INT 21h with AH = 09h, again from the equate statements.
- 
- Watch out for missing "h's in your number statements in your MASM programs, this can still cause you lots of problems!!!
- 
-

- [\*\*MASM Labels\*\*](#)
- 
- —
- **Next time**
- Floating Point Numbers
- Coprocessor
- Pages (5 Antonakos ) 312 - 315
- (2 Irvine) 556 ...
  - **Test Review**
  - 
  - Things to know:
  - Know how to binary coded decimal BCD, operations, adding and subtracting.
  - 
  - Given a set of register values before a set of instructions, be able to show the resulting values.
  - 
  - You may be given a set of register values, and instructions and you will want to be able to find the new register contents.
  - 
  - .Tiny model = requires that all software and data fit into one 64k byte memory segment.
  - .Small model = requires that only one data segment be used with one code segment, for a total of 128K bytes of memory.
  - Other models are available up to Huge model (see MASMDocumentation).
  - 
  - Given Address data (buffer area, for example), register status, and instructions, determine the resulting address data, and register values.
  - 
  - Given segment descriptions from a .LIST file develop a Memory map and determine the number of unused bytes. Page 295
  - 
  - Be able to determine if a program is a .COM or a .EXE . Page 296
  - 
  - Given instructions like [PUSH](#) , Call, [POP](#) , Ret, what happens to the stack.
  - 
  -
- 
- —

[Back](#)

- Previous Exam Questions
- Midterm Study Guide
- **ADC**
    - Analog Digital Converter
    - 0-1024
    - **analogRead()**:

#### Description

Reads the value from the specified analog pin. The Arduino board contains a 6 channel (8 channels on the Mini and Nano, 16 on the Mega), 10-bit analog to digital converter. This means that it will map input voltages between 0 and 5 volts into integer values between 0 and 1023. This yields a resolution between readings of: 5 volts / 1024 units or, .0049 volts (4.9 mV) per unit. The input range and resolution can be changed using [analogReference\(\)](#).

It takes about 100 microseconds (0.0001 s) to read an analog input, so the maximum reading rate is about 10,000 times a second.

#### Syntax

```
analogRead(pin)
```

#### Parameters

pin: the number of the analog input pin to read from (0 to 5 on most boards, 0 to 7 on the Mini and Nano, 0 to 15 on the Mega)

#### Returns

```
int (0 to 1023)
```

#### Note

If the analog input pin is not connected to anything, the value returned by `analogRead()` will fluctuate based on a number of factors (e.g. the values of the other analog inputs, how close your hand is to the board, etc.).

#### Example

```
int analogPin = 3; // potentiometer wiper (middle terminal) connected to analog pin 3
```

```
// outside leads to ground and +5V
```

```
int val = 0; // variable to store the value read
```

```
void setup()
```

```
{
```

```
    Serial.begin(9600); // setup serial
```

```
}
```

```
void loop()
```

```
{
```

```
    val = analogRead(analogPin); // read the input pin
```

```
    Serial.println(val); // debug value
```

```
}
```

- **Arduino**

- **pinMod():**

- Description

- Configures the specified pin to behave either as an input or an output. See the description of [digital pins](#) for details on the functionality of the pins.

- As of Arduino 1.0.1, it is possible to enable the internal pullup resistors with the mode INPUT\_PULLUP. Additionally, the INPUT mode explicitly disables the internal pullups.

- Syntax

- `pinMode(pin, mode)`

- Parameters

- `pin`: the number of the pin whose mode you wish to set

- mode: INPUT, OUTPUT, or INPUT\_PULLUP. (see the [digital pins](#) page for a more complete description of the functionality.)
- Returns
  - None
- Example
  - 
  - int ledPin = 13; // LED connected to digital pin 13
  - 
  - void **setup()**
  - {
  - pinMode(ledPin, OUTPUT); // sets the digital pin as output
  - }
  - 
  - void **loop()**
  - {
  - digitalWrite(ledPin, HIGH); // sets the LED on
  - delay(1000); // waits for a second
  - digitalWrite(ledPin, LOW); // sets the LED off
  - delay(1000); // waits for a second
  - }
- Note
  - The analog input pins can be used as digital pins, referred to as A0, A1, etc.
- **digitalWrite():**
  - Description
    - Write a HIGH or a LOW value to a digital pin.
    - If the pin has been configured as an OUTPUT with [pinMode\(\)](#), its voltage will be set to the corresponding value: 5V (or 3.3V on 3.3V boards) for HIGH, 0V (ground) for LOW.
    - If the pin is configured as an INPUT, [digitalWrite\(\)](#) will enable (HIGH) or disable (LOW) the internal pullup on the input pin. It is recommended to set the [pinMode\(\)](#) to INPUT\_PULLUP to enable the internal pull-up resistor. See the [digital pins tutorial](#) for more information.
    - NOTE: If you do not set the [pinMode\(\)](#) to OUTPUT, and connect an LED to a pin, when calling [digitalWrite\(HIGH\)](#), the LED may appear dim. Without explicitly setting [pinMode\(\)](#), [digitalWrite\(\)](#) will have enabled the internal pull-up resistor, which acts like a large current-limiting resistor.
  - Syntax
    - `digitalWrite(pin, value)`
  - Parameters
    - pin: the pin number
    - value: HIGH or LOW

- Returns
- none
- Example

```

int ledPin = 13;           // LED connected to digital pin 13

void setup()
{
  pinMode(ledPin, OUTPUT); // sets the digital pin as output
}

void loop()
{
  digitalWrite(ledPin, HIGH); // sets the LED on
  delay(1000);             // waits for a second
  digitalWrite(ledPin, LOW); // sets the LED off
  delay(1000);             // waits for a second
}

```

- Sets pin 13 to HIGH, makes a one-second-long delay, and sets the pin back to LOW.

- Note
  - The analog input pins can be used as digital pins, referred to as A0, A1, etc.

#### - **digitalRead():**

- Description
    - Reads the value from a specified digital pin, either HIGH or LOW.
  - Syntax
    - `digitalRead(pin)`
  - Parameters
    - `pin`: the number of the digital pin you want to read (*int*)
  - Returns
    - HIGH or LOW
  - Example
    - Sets pin 13 to the same value as pin 7, declared as an input.
- ```

- int ledPin = 13; // LED connected to digital pin 13
- int inPin = 7;  // pushbutton connected to digital pin 7
- int val = 0;   // variable to store the read value
-
- void setup()
{

```

```

-   pinMode(ledPin, OUTPUT);    // sets the digital pin 13 as output
-   pinMode(inPin, INPUT);     // sets the digital pin 7 as input
-
- }
-
- void loop()
{
-   val = digitalRead(inPin); // read the input pin
-   digitalWrite(ledPin, val); // sets the LED to the button's value
}

```

- Note
  - If the pin isn't connected to anything, digitalRead() can return either HIGH or LOW (and this can change randomly).
  - The analog input pins can be used as digital pins, referred to as A0, A1, etc.
- **Constants:**

Constants are predefined expressions in the Arduino language. They are used to make the programs easier to read. We classify constants in groups:

#### Defining Logical Levels: true and false (Boolean Constants)

There are two constants used to represent truth and falsity in the Arduino language: true, and false.

##### False

false is the easier of the two to define. false is defined as 0 (zero).

##### True

true is often said to be defined as 1, which is correct, but true has a wider definition. Any integer which is non-zero is true, in a Boolean sense. So -1, 2 and -200 are all defined as true, too, in a Boolean sense.

Note that the true and false constants are typed in lowercase unlike HIGH, LOW, INPUT, and OUTPUT.

#### Defining Pin Levels: HIGH and LOW

When reading or writing to a digital pin there are only two possible values a pin can take/be-set-to: HIGH and LOW.

##### HIGH

The meaning of HIGH (in reference to a pin) is somewhat different depending on whether a pin is set to an INPUT or OUTPUT. When a pin is configured as an INPUT with pinMode(), and read with digitalRead(), the Arduino (Atmega) will report HIGH if:

a voltage greater than 3.0V is present at the pin (5V boards);  
a voltage greater than 2.0V is present at the pin (3.3V boards);  
A pin may also be configured as an INPUT with pinMode(), and subsequently made HIGH with digitalWrite(). This will enable the internal 20K pullup resistors, which will *pull up* the input pin to a HIGH reading unless it is pulled LOW by external circuitry. This is how INPUT\_PULLUP works and is described below in more detail.

When a pin is configured to OUTPUT with `pinMode()`, and set to HIGH with `digitalWrite()`, the pin is at:

5 volts (5V boards);

3.3 volts (3.3V boards);

In this state it can *source* current, e.g. light an LED that is connected through a series resistor to ground.

## LOW

The meaning of LOW also has a different meaning depending on whether a pin is set to INPUT or OUTPUT. When a pin is configured as an INPUT with `pinMode()`, and read with `digitalRead()`, the Arduino (Atmega) will report LOW if:

a voltage less than 1.5V is present at the pin (5V boards);

a voltage less than 1.0V (Approx) is present at the pin (3.3V boards);

When a pin is configured to OUTPUT with `pinMode()`, and set to LOW with `digitalWrite()`, the pin is at 0 volts (both 5V and 3.3V boards). In this state it can *sink* current, e.g. light an LED that is connected through a series resistor to +5 volts (or +3.3 volts).

## Defining Digital Pins modes: INPUT, INPUT\_PULLUP, and OUTPUT

Digital pins can be used as INPUT, INPUT\_PULLUP, or OUTPUT. Changing a pin with `pinMode()` changes the electrical behavior of the pin.

### Pins Configured as INPUT

Arduino (Atmega) pins configured as INPUT with `pinMode()` are said to be in a high-impedance state. Pins configured as INPUT make extremely small demands on the circuit that they are sampling, equivalent to a series resistor of 100 Megohms in front of the pin. This makes them useful for reading a sensor.

If you have your pin configured as an INPUT, and are reading a switch, when the switch is in the open state the input pin will be "floating", resulting in unpredictable results. In order to assure a proper reading when the switch is open, a pull-up or pull-down resistor must be used. The purpose of this resistor is to pull the pin to a known state when the switch is open. A 10 K ohm resistor is usually chosen, as it is a low enough value to reliably prevent a floating input, and at the same time a high enough value to not draw too much current when the switch is closed. See the [Digital Read Serial](#) tutorial for more information.

If a pull-down resistor is used, the input pin will be LOW when the switch is open and HIGH when the switch is closed.

If a pull-up resistor is used, the input pin will be HIGH when the switch is open and LOW when the switch is closed.

### Pins Configured as INPUT\_PULLUP

The Atmega microcontroller on the Arduino has internal pull-up resistors (resistors that connect to power internally) that you can access. If you prefer to use these instead of external pull-up resistors, you can use the INPUT\_PULLUP argument in `pinMode()`.

See the [Input Pullup Serial](#) tutorial for an example of this in use.

Pins configured as inputs with either INPUT or INPUT\_PULLUP can be damaged or destroyed if they are connected to voltages below ground (negative voltages) or above the positive power rail (5V or 3V).

#### Pins Configured as Outputs

Pins configured as OUTPUT with pinMode() are said to be in a low-impedance state. This means that they can provide a substantial amount of current to other circuits. Atmega pins can source (provide current) or sink (absorb current) up to 40 mA (milliamps) of current to other devices/circuits. This makes them useful for powering LEDs because LEDs typically use less than 40 mA. Loads greater than 40 mA (e.g. motors) will require a transistor or other interface circuitry.

Pins configured as outputs can be damaged or destroyed if they are connected to either the ground or positive power rails.

#### Defining built-ins: LED\_BUILTIN

Most Arduino boards have a pin connected to an on-board LED in series with a resistor. The constant LED\_BUILTIN is the number of the pin to which the on-board LED is connected. Most boards have this LED connected to digital pin 13.

Reference : [ATMEL / MICROCHIP 328 datasheet](#)

- **Serial:**

Serial communication on pins TX/RX uses TTL logic levels (5V or 3.3V depending on the board). Don't connect these pins directly to an RS232 serial port; they operate at +/- 12V and can damage your Arduino board.

Serial is used for communication between the Arduino board and a computer or other devices. All Arduino boards have at least one serial port (also known as a UART or USART): Serial. It communicates on digital pins 0 (RX) and 1 (TX) as well as with the computer via USB. Thus, if you use these functions, you cannot also use pins 0 and 1 for digital input or output.

You can use the Arduino environment's built-in serial monitor to communicate with an Arduino board. Click the serial monitor button in the toolbar and select the same baud rate used in the call to begin().

The [Arduino Mega](#) has three additional serial ports: Serial1 on pins 19 (RX) and 18 (TX), Serial2 on pins 17 (RX) and 16 (TX), Serial3 on pins 15 (RX) and 14 (TX). To use these pins to communicate with your personal computer, you will need an additional USB-to-serial adaptor, as they are not connected to the Mega's USB-to-serial adaptor. To use them to communicate with an external TTL serial device, connect the TX pin to your device's RX pin, the RX to your device's TX pin, and the ground of your Mega to your device's ground.

The [Arduino Due](#) has three additional 3.3V TTL serial ports: Serial1 on pins 19 (RX) and 18 (TX); Serial2 on pins 17 (RX) and 16 (TX), Serial3 on pins 15 (RX) and 14 (TX). Pins 0 and 1 are also connected to the corresponding pins of the ATmega16U2 USB-to-TTL Serial chip, which is connected to the USB debug port. Additionally, there is a native USB-serial port on the SAM3X chip, [SerialUSB](#).

The Arduino Leonardo board uses Serial1 to communicate via TTL (5V) serial on pins 0 (RX) and 1 (TX). Serial is reserved for USB CDC communication. For more information, refer to the Leonardo [getting started](#) page and [hardware](#) page.

## - **Digital Pins:**

- The pins on the Arduino can be configured as either inputs or outputs. This document explains the functioning of the pins in those modes. While the title of this document refers to digital pins, it is important to note that vast majority of Arduino (Atmega) analog pins, may be configured, and used, in exactly the same manner as digital pins.
- Properties of Pins Configured as INPUT
- Arduino (Atmega) pins default to inputs, so they don't need to be explicitly declared as inputs with `pinMode()` when you're using them as inputs. Pins configured this way are said to be in a high-impedance state. Input pins make extremely small demands on the circuit that they are sampling, equivalent to a series resistor of 100 megohm in front of the pin. This means that it takes very little current to move the input pin from one state to another, and can make the pins useful for such tasks as implementing [a capacitive touch sensor](#), reading an LED as a [photodiode](#), or reading an analog sensor with a scheme such as [RCTime](#).

- This also means however, that pins configured as `pinMode(pin, INPUT)` with nothing connected to them, or with wires connected to them that are not connected to other circuits, will report seemingly random changes in pin state, picking up electrical noise from the environment, or capacitively coupling the state of a nearby pin.
- Pullup Resistors with pins configured as INPUT
- Often it is useful to steer an input pin to a known state if no input is present. This can be done by adding a pullup resistor (to +5V), or a pulldown resistor (resistor to ground) on the input. A 10K resistor is a good value for a pullup or pulldown resistor.
- Properties of Pins Configured as INPUT\_PULLUP
- There are 20K pullup resistors built into the Atmega chip that can be accessed from software. These built-in pullup resistors are accessed by setting the `pinMode()` as INPUT\_PULLUP. This effectively inverts the behavior of the INPUT mode, where HIGH means the sensor is off, and LOW means the sensor is on.
- The value of this pullup depends on the microcontroller used. On most AVR-based boards, the value is guaranteed to be between 20kΩ and 50kΩ. On the Arduino Due, it is between 50kΩ and 150kΩ. For the exact value, consult the datasheet of the microcontroller on your board.
- When connecting a sensor to a pin configured with INPUT\_PULLUP, the other end should be connected to ground. In the case of a simple switch, this causes the pin to read HIGH when the switch is open, and LOW when the switch is pressed.
- The pullup resistors provide enough current to dimly light an LED connected to a pin that has been configured as an input. If LEDs in a project seem to be working, but very dimly, this is likely what is going on.
- The pullup resistors are controlled by the same registers (internal chip memory locations) that control whether a pin is HIGH or LOW. Consequently, a pin that is configured to have pullup resistors turned on when the pin is an INPUT, will have the pin configured as HIGH if the pin is then switched to an OUTPUT with `pinMode()`. This works in the other direction as well, and an output pin that is left in a HIGH state will have the pullup resistors set if switched to an input with `pinMode()`.
- Prior to Arduino 1.0.1, it was possible to configure the internal pull-ups in the following manner:

  - ```
pinMode(pin, INPUT);      // set pin to input
digitalWrite(pin, HIGH); // turn on pullup resistors
```
  - NOTE: Digital pin 13 is harder to use as a digital input than the other digital pins because it has an LED and resistor attached to it that's soldered to the board on most boards. If you enable its internal 20k pull-up resistor, it will hang at around 1.7V instead of the expected 5V because the onboard LED and series resistor pull the voltage level down, meaning it always returns LOW. If you must use pin 13 as a digital input, set its `pinMode()` to INPUT and use an external pull down resistor.
  - Properties of Pins Configured as OUTPUT
  - Pins configured as OUTPUT with `pinMode()` are said to be in a low-impedance state. This means that they can provide a substantial amount of current to other circuits. Atmega pins can source (provide positive current) or sink (provide negative current) up to 40 mA (milliamps) of current to other devices/circuits. This is enough current to brightly light up an LED (don't forget the series

resistor), or run many sensors, for example, but not enough current to run most relays, solenoids, or motors.

- Short circuits on Arduino pins, or attempting to run high current devices from them, can damage or destroy the output transistors in the pin, or damage the entire Atmega chip. Often this will result in a "dead" pin in the microcontroller but the remaining chip will still function adequately. For this reason it is a good idea to connect OUTPUT pins to other devices with  $470\Omega$  or  $1k$  resistors, unless maximum current draw from the pins is required for a particular application.

- **Different Tools With Analog Discovery**

- Logger (Voltmeter) good for measuring spikes in voltage
- Oscilloscope
- Logic Analyzer
- BUS Analyzer - Measure UART
- AWG - , AWGs allow us to generate a number of different types of signals and with these signals we can create test cases for our code to run on.
- PWM - Pulse width modulation (PWM) is a techniques to generate a analog signal with a digital system.
- **MAX 32**
- Interrupts - Pin 3 (INT0), Pin 2 (INT1), Pin 7 (INT2), Pin 21 (INT3), Pin 20 (INT4).
- **Benefits of Arduino**
- Good For rapid Prototyping
- **Setback of Arduino**
- Cost - Very Expensive
- Open Source
- Doesn't Scale
- Efficiency
- **Benefits of Assembly**
- More Efficient
- **How many Bytes in Char, Int, Float, and Doubles**

| Type                 | Storage size        | Value range                                                 |
|----------------------|---------------------|-------------------------------------------------------------|
| <b>char</b>          | <b>1 byte</b>       | <b>-128 to 127 or 0 to 255</b>                              |
| <b>unsigned char</b> | <b>1 byte</b>       | <b>0 to 255</b>                                             |
| <b>signed char</b>   | <b>1 byte</b>       | <b>-128 to 127</b>                                          |
| <b>int</b>           | <b>2 or 4 bytes</b> | <b>-32,768 to 32,767 or -2,147,483,648 to 2,147,483,647</b> |
| <b>unsigned int</b>  | <b>2 or 4 bytes</b> | <b>0 to 65,535 or 0 to 4,294,967,295</b>                    |
| <b>short</b>         | <b>2 bytes</b>      | <b>-32,768 to 32,767</b>                                    |

|                       |                |                                        |
|-----------------------|----------------|----------------------------------------|
| <b>unsigned short</b> | <b>2 bytes</b> | <b>0 to 65,535</b>                     |
| <b>long</b>           | <b>4 bytes</b> | <b>-2,147,483,648 to 2,147,483,647</b> |
| <b>unsigned long</b>  | <b>4 bytes</b> | <b>0 to 4,294,967,295</b>              |

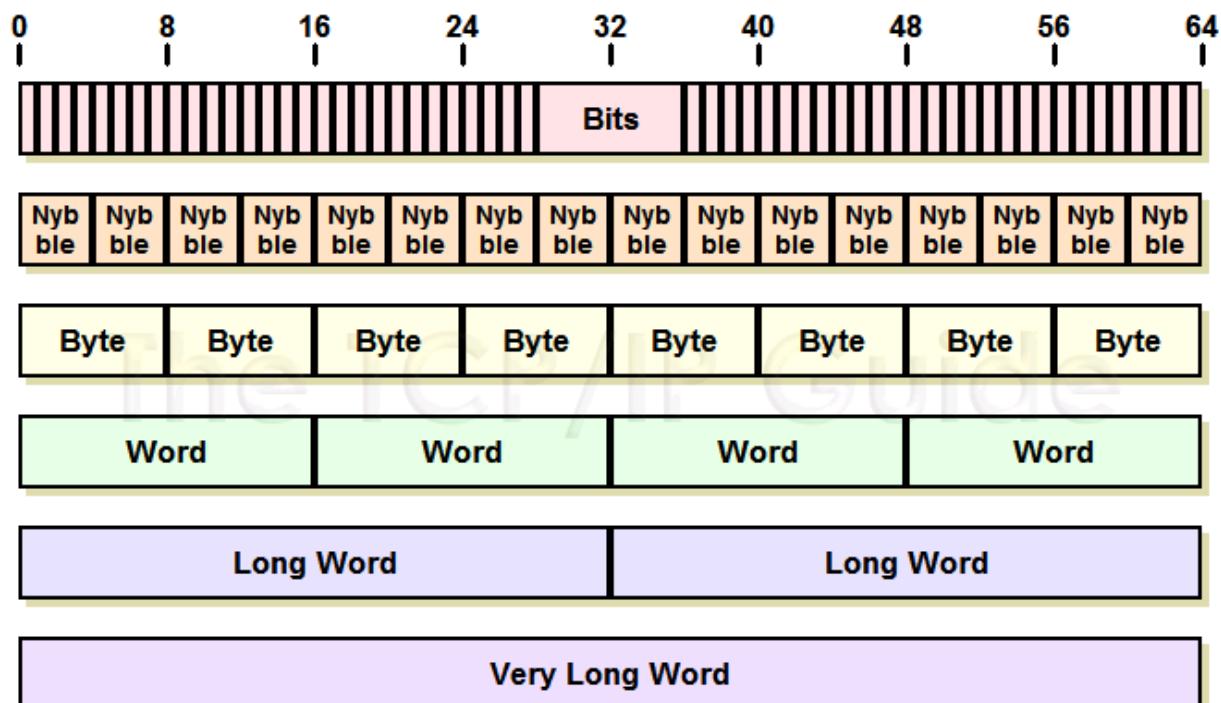
| Type               | Storage size   | Value range                   | Precision                |
|--------------------|----------------|-------------------------------|--------------------------|
| <b>float</b>       | <b>4 byte</b>  | <b>1.2E-38 to 3.4E+38</b>     | <b>6 decimal places</b>  |
| <b>double</b>      | <b>8 byte</b>  | <b>2.3E-308 to 1.7E+308</b>   | <b>15 decimal places</b> |
| <b>long double</b> | <b>10 byte</b> | <b>3.4E-4932 to 1.1E+4932</b> | <b>19 decimal places</b> |

- **Pre - Compiler in C**
    - #include <stdio.h>
    - #define LED\_Pin 3  
[https://gcc.gnu.org/onlinedocs/gcc-2.95.3/cpp\\_1.html](https://gcc.gnu.org/onlinedocs/gcc-2.95.3/cpp_1.html)
  - **Inputs**
    - For Inputs we use the digitalRead() function to receive either a HIGH or LOW value.
  - **Infinite Loop**
    - While(1)
    - for(;;)
    - Loop()
- Reference Arduino: [http://www.learningaboutelectronics.com/Articles/While-\(1\)-embedded-C.php](http://www.learningaboutelectronics.com/Articles/While-(1)-embedded-C.php)
- **When should you use Pointers?**
  - **I2C**
    - A common communication protocol that is found in embedded systems. Used in sensors and communication between devices I2C allows for many devices to share one communication bus and reduces the number of wires on a board.
    - Inter Integrated Circuit
    - 2 Wires
    - Wire Library
  - **Hand Assembly Encoding**
    - Add BX to CX what's the output
  - **Assembly Loop:** [https://www.tutorialspoint.com/assembly\\_programming/assembly\\_loops.htm](https://www.tutorialspoint.com/assembly_programming/assembly_loops.htm)

**Inline Assembly:** <https://software.intel.com/en-us/node/694235>

- **Assembly MASM**
- **Assembly Jmp + Conditional Jump**  
[https://www.tutorialspoint.com/assembly\\_programming/assembly\\_conditions.htm](https://www.tutorialspoint.com/assembly_programming/assembly_conditions.htm)
- **Bit to Byte , Word to Nibble**

| Number of Bits | Common Representation Terms |
|----------------|-----------------------------|
| 1              | Bit / Digit / Flag          |
| 4              | Nibble / Nibble             |
| 8              | Byte / Octet / Character    |
| 16             | Double Byte / Word          |
| 32             | Double Word / Long Word     |
| 64             | Very Long Word              |



- ASCII to HEX

| Dec | Hx     | Oct | Char                               |  | Dec | Hx     | Oct | Html  | Chr          |  | Dec | Hx     | Oct | Html  | Chr      |  | Dec | Hx     | Oct | Html   | Chr        |
|-----|--------|-----|------------------------------------|--|-----|--------|-----|-------|--------------|--|-----|--------|-----|-------|----------|--|-----|--------|-----|--------|------------|
| 0   | 0 000  | 000 | <b>NUL</b> (null)                  |  | 32  | 20 040 | 000 | &#32; | <b>Space</b> |  | 64  | 40 100 | 000 | &#64; | <b>Ø</b> |  | 96  | 60 140 | 000 | &#96;  | '          |
| 1   | 1 001  | 001 | <b>SOH</b> (start of heading)      |  | 33  | 21 041 | 001 | &#33; | !            |  | 65  | 41 101 | 001 | &#65; | <b>A</b> |  | 97  | 61 141 | 001 | &#97;  | <b>a</b>   |
| 2   | 2 002  | 002 | <b>STX</b> (start of text)         |  | 34  | 22 042 | 002 | &#34; | "            |  | 66  | 42 102 | 002 | &#66; | <b>B</b> |  | 98  | 62 142 | 002 | &#98;  | <b>b</b>   |
| 3   | 3 003  | 003 | <b>ETX</b> (end of text)           |  | 35  | 23 043 | 003 | &#35; | #            |  | 67  | 43 103 | 003 | &#67; | <b>C</b> |  | 99  | 63 143 | 003 | &#99;  | <b>c</b>   |
| 4   | 4 004  | 004 | <b>EOT</b> (end of transmission)   |  | 36  | 24 044 | 004 | &#36; | \$           |  | 68  | 44 104 | 004 | &#68; | <b>D</b> |  | 100 | 64 144 | 004 | &#100; | <b>d</b>   |
| 5   | 5 005  | 005 | <b>ENQ</b> (enquiry)               |  | 37  | 25 045 | 005 | &#37; | %            |  | 69  | 45 105 | 005 | &#69; | <b>E</b> |  | 101 | 65 145 | 005 | &#101; | <b>e</b>   |
| 6   | 6 006  | 006 | <b>ACK</b> (acknowledge)           |  | 38  | 26 046 | 006 | &#38; | &            |  | 70  | 46 106 | 006 | &#70; | <b>F</b> |  | 102 | 66 146 | 006 | &#102; | <b>f</b>   |
| 7   | 7 007  | 007 | <b>BEL</b> (bell)                  |  | 39  | 27 047 | 007 | &#39; | !            |  | 71  | 47 107 | 007 | &#71; | <b>G</b> |  | 103 | 67 147 | 007 | &#103; | <b>g</b>   |
| 8   | 8 010  | 010 | <b>BS</b> (backspace)              |  | 40  | 28 050 | 010 | &#40; | (            |  | 72  | 48 110 | 010 | &#72; | <b>H</b> |  | 104 | 68 150 | 010 | &#104; | <b>h</b>   |
| 9   | 9 011  | 011 | <b>TAB</b> (horizontal tab)        |  | 41  | 29 051 | 011 | &#41; | )            |  | 73  | 49 111 | 011 | &#73; | <b>I</b> |  | 105 | 69 151 | 011 | &#105; | <b>i</b>   |
| 10  | A 012  | 012 | <b>LF</b> (NL line feed, new line) |  | 42  | 2A 052 | 012 | &#42; | *            |  | 74  | 4A 112 | 012 | &#74; | <b>J</b> |  | 106 | 6A 152 | 012 | &#106; | <b>j</b>   |
| 11  | B 013  | 013 | <b>VT</b> (vertical tab)           |  | 43  | 2B 053 | 013 | &#43; | +            |  | 75  | 4B 113 | 013 | &#75; | <b>K</b> |  | 107 | 6B 153 | 013 | &#107; | <b>k</b>   |
| 12  | C 014  | 014 | <b>FF</b> (NP form feed, new page) |  | 44  | 2C 054 | 014 | &#44; | ,            |  | 76  | 4C 114 | 014 | &#76; | <b>L</b> |  | 108 | 6C 154 | 014 | &#108; | <b>l</b>   |
| 13  | D 015  | 015 | <b>CR</b> (carriage return)        |  | 45  | 2D 055 | 015 | &#45; | -            |  | 77  | 4D 115 | 015 | &#77; | <b>M</b> |  | 109 | 6D 155 | 015 | &#109; | <b>m</b>   |
| 14  | E 016  | 016 | <b>SO</b> (shift out)              |  | 46  | 2E 056 | 016 | &#46; | .            |  | 78  | 4E 116 | 016 | &#78; | <b>N</b> |  | 110 | 6E 156 | 016 | &#110; | <b>n</b>   |
| 15  | F 017  | 017 | <b>SI</b> (shift in)               |  | 47  | 2F 057 | 017 | &#47; | /            |  | 79  | 4F 117 | 017 | &#79; | <b>O</b> |  | 111 | 6F 157 | 017 | &#111; | <b>o</b>   |
| 16  | 10 020 | 020 | <b>DLE</b> (data link escape)      |  | 48  | 30 060 | 020 | &#48; | 0            |  | 80  | 50 120 | 020 | &#80; | <b>P</b> |  | 112 | 70 160 | 020 | &#112; | <b>p</b>   |
| 17  | 11 021 | 021 | <b>DC1</b> (device control 1)      |  | 49  | 31 061 | 021 | &#49; | 1            |  | 81  | 51 121 | 021 | &#81; | <b>Q</b> |  | 113 | 71 161 | 021 | &#113; | <b>q</b>   |
| 18  | 12 022 | 022 | <b>DC2</b> (device control 2)      |  | 50  | 32 062 | 022 | &#50; | 2            |  | 82  | 52 122 | 022 | &#82; | <b>R</b> |  | 114 | 72 162 | 022 | &#114; | <b>r</b>   |
| 19  | 13 023 | 023 | <b>DC3</b> (device control 3)      |  | 51  | 33 063 | 023 | &#51; | 3            |  | 83  | 53 123 | 023 | &#83; | <b>S</b> |  | 115 | 73 163 | 023 | &#115; | <b>s</b>   |
| 20  | 14 024 | 024 | <b>DC4</b> (device control 4)      |  | 52  | 34 064 | 024 | &#52; | 4            |  | 84  | 54 124 | 024 | &#84; | <b>T</b> |  | 116 | 74 164 | 024 | &#116; | <b>t</b>   |
| 21  | 15 025 | 025 | <b>NAK</b> (negative acknowledge)  |  | 53  | 35 065 | 025 | &#53; | 5            |  | 85  | 55 125 | 025 | &#85; | <b>U</b> |  | 117 | 75 165 | 025 | &#117; | <b>u</b>   |
| 22  | 16 026 | 026 | <b>SYN</b> (synchronous idle)      |  | 54  | 36 066 | 026 | &#54; | 6            |  | 86  | 56 126 | 026 | &#86; | <b>V</b> |  | 118 | 76 166 | 026 | &#118; | <b>v</b>   |
| 23  | 17 027 | 027 | <b>ETB</b> (end of trans. block)   |  | 55  | 37 067 | 027 | &#55; | 7            |  | 87  | 57 127 | 027 | &#87; | <b>W</b> |  | 119 | 77 167 | 027 | &#119; | <b>w</b>   |
| 24  | 18 030 | 030 | <b>CAN</b> (cancel)                |  | 56  | 38 070 | 030 | &#56; | 8            |  | 88  | 58 130 | 030 | &#88; | <b>X</b> |  | 120 | 78 170 | 030 | &#120; | <b>x</b>   |
| 25  | 19 031 | 031 | <b>EM</b> (end of medium)          |  | 57  | 39 071 | 031 | &#57; | 9            |  | 89  | 59 131 | 031 | &#89; | <b>Y</b> |  | 121 | 79 171 | 031 | &#121; | <b>y</b>   |
| 26  | 1A 032 | 032 | <b>SUB</b> (substitute)            |  | 58  | 3A 072 | 032 | &#58; | :            |  | 90  | 5A 132 | 032 | &#90; | <b>Z</b> |  | 122 | 7A 172 | 032 | &#122; | <b>z</b>   |
| 27  | 1B 033 | 033 | <b>ESC</b> (escape)                |  | 59  | 3B 073 | 033 | &#59; | :            |  | 91  | 5B 133 | 033 | &#91; | <b>[</b> |  | 123 | 7B 173 | 033 | &#123; | {          |
| 28  | 1C 034 | 034 | <b>FS</b> (file separator)         |  | 60  | 3C 074 | 034 | &#60; | <            |  | 92  | 5C 134 | 034 | &#92; | <b>\</b> |  | 124 | 7C 174 | 034 | &#124; |            |
| 29  | 1D 035 | 035 | <b>GS</b> (group separator)        |  | 61  | 3D 075 | 035 | &#61; | =            |  | 93  | 5D 135 | 035 | &#93; | <b>]</b> |  | 125 | 7D 175 | 035 | &#125; | }          |
| 30  | 1E 036 | 036 | <b>RS</b> (record separator)       |  | 62  | 3E 076 | 036 | &#62; | >            |  | 94  | 5E 136 | 036 | &#94; | <b>^</b> |  | 126 | 7E 176 | 036 | &#126; | <u>~</u>   |
| 31  | 1F 037 | 037 | <b>US</b> (unit separator)         |  | 63  | 3F 077 | 037 | &#63; | ?            |  | 95  | 5F 137 | 037 | &#95; | <b>_</b> |  | 127 | 7F 177 | 037 | &#127; | <b>DEL</b> |

Source: [www.LookupTables.com](http://www.LookupTables.com)

- **HEX to Binary**

| Hex | Binary |
|-----|--------|
| 0   | 0000   |
| 1   | 0001   |
| 2   | 0010   |
| 3   | 0011   |
| 4   | 0100   |
| 5   | 0101   |
| 6   | 0110   |
| 7   | 0111   |
| 8   | 1000   |

|   |      |
|---|------|
| 9 | 1001 |
| A | 1010 |
| B | 1011 |
| C | 1100 |
| D | 1101 |
| E | 1110 |
| F | 1111 |

- **Microchip**
  - Set = ORing
  - Clear = ANDing
  - Toggle = XORing
- **Pointer:**
  - <https://users.cs.cf.ac.uk/Dave.Marshall/C/node10.html>
- When Should You use it ?
  - To Reference
  - Pass by Reference vs Pass by Value
- Commands
  - <http://ref.x86asm.net/coder32.html#x0FC8>

For reading machine language

<http://ref.x86asm.net/coder32.html#xF4A>

#### Debug:

##### Flags:

[https://moodle2.ecs.csus.edu/pluginfile.php/58444/mod\\_resource/content/6/Lecture03/174lecture03\\_files/irvineappbpage599\\_19.pdf](https://moodle2.ecs.csus.edu/pluginfile.php/58444/mod_resource/content/6/Lecture03/174lecture03_files/irvineappbpage599_19.pdf)  
[https://moodle2.ecs.csus.edu/pluginfile.php/58444/mod\\_resource/content/6/Lecture03/174lecture03\\_files/debugflagstable4-4p152\\_40.pdf](https://moodle2.ecs.csus.edu/pluginfile.php/58444/mod_resource/content/6/Lecture03/174lecture03_files/debugflagstable4-4p152_40.pdf)

##### Conditional Jumps:

[https://moodle2.ecs.csus.edu/pluginfile.php/58444/mod\\_resource/content/6/Lecture03/174lecture03\\_files/jumpcontable4-3page145\\_39.pdf](https://moodle2.ecs.csus.edu/pluginfile.php/58444/mod_resource/content/6/Lecture03/174lecture03_files/jumpcontable4-3page145_39.pdf)  
[https://moodle2.ecs.csus.edu/pluginfile.php/58444/mod\\_resource/content/6/Lecture03/174lecture03\\_files/conditionaljumpflowc1891\\_41.pdf](https://moodle2.ecs.csus.edu/pluginfile.php/58444/mod_resource/content/6/Lecture03/174lecture03_files/conditionaljumpflowc1891_41.pdf)

##### Command List:

###### Debug

Starts the Debug program, which you can use to test and debug executable files.

###### On This Page

[Syntax debug \[\[drive : \]\[path\]filename \[testfile-parameters\]\]](#)  
[Debug: A \(Assemble\)](#)  
[Debug: C \(Compare\)](#)  
[Debug: D \(Dump\)](#)  
[Debug: E \(Enter\)](#)  
[Debug: F \(Fill\)](#)  
[Debug: G \(Go\)](#)  
[Debug: H \(Hex\)](#)  
[Debug: I \(Input\)](#)

[Debug: L \(Load\)](#)  
[Debug: M \(Move\)](#)  
[Debug: N \(Name\)](#)  
[Debug: O \(Output\)](#)  
[Debug: P \(Proceed\)](#)  
[Debug: Q \(Quit\)](#)  
[Debug: R \(Register\)](#)  
[Debug: S \(Search\)](#)  
[Debug: T \(Trace\)](#)  
[Debug: U \(Unassemble\)](#)  
[Debug: W \(Write\)](#)  
[Debug: XA \(Allocate Expanded Memory\)](#)  
[Debug: XD \(Deallocate Expanded Memory\)](#)  
[Debug: XM \(Map Expanded Memory Pages\)](#)  
[Debug: XS \(Display Expanded-Memory Status\)](#)  
[Defrag](#)  
[Del \(Erase\)](#)  
[Deltree](#)  
[Device](#)  
[Devicehigh](#)  
[Dir](#)  
[Diskcomp](#)  
[Diskcopy](#)  
[DISPLAY.SYS](#)  
[Dos](#)  
[Doskey](#)  
[Doshell](#)  
[DRIVER.SYS](#)  
[Drivparm](#)  
[Echo](#)  
[Edit](#)  
[EGA.SYS](#)

Syntax debug [[drive : ][path]filename [testfile-parameters]]

#### Parameters

**[ drive : ][ path ] filename**

Specifies the location and name of the executable file you want to test.

**testfile-parameters**

Specifies any command-line information required by the executable file you want to test.

#### Notes

#### Using the debug command without specifying a file to be tested

If you use the **debug** command without a location and filename, you then type all Debug commands in response to the Debug prompt, a hyphen (-).

#### Debug commands

| Command | Description                            |
|---------|----------------------------------------|
| ?       | Displays a list of the Debug commands. |
| a       | Assembles 8086/8087/8088 mnemonics.    |
| c       | Compares two portions of memory.       |

|   |                                                                                                   |
|---|---------------------------------------------------------------------------------------------------|
| d | Displays the contents of a portion of memory.                                                     |
| e | Enters data into memory starting at a specified address.                                          |
| f | Fills a range of memory with specified values.                                                    |
| g | Runs the executable file that is in memory.                                                       |
| h | Performs hexadecimal arithmetic.                                                                  |
| i | Displays one byte value from a specified port.                                                    |
| l | Loads the contents of a file or disk sectors into memory.                                         |
| m | Copies the contents of a block of memory.                                                         |
| n | Specifies a file for an l or w command, or specifies the parameters for the file you are testing. |
| o | Sends one byte value to an output port.                                                           |
| p | Executes a loop, a repeated string instruction, a software interrupt, or a subroutine.            |
| q | Stops the Debug session.                                                                          |
| r | Displays or alters the contents of one or more registers.                                         |

|    |                                                                                                                                                                          |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| s  | Searches a portion of memory for a specified pattern of one or more byte values.                                                                                         |
| t  | Executes one instruction and then displays the contents of all registers, the status of all flags, and the decoded form of the instruction that Debug will execute next. |
| u  | Disassembles bytes and displays the corresponding source statements.                                                                                                     |
| w  | Writes the file being tested to a disk.                                                                                                                                  |
| xa | Allocates expanded memory.                                                                                                                                               |
| xd | Deallocates expanded memory.                                                                                                                                             |
| xm | Maps expanded memory pages.                                                                                                                                              |
| xs | Displays the status of expanded memory.                                                                                                                                  |

#### Separating command parameters

All Debug commands accept parameters, except the **q** command. You can separate parameters with commas or spaces, but these separators are required only between two hexadecimal values. Therefore, the following commands are equivalent:

```
dcs:100 110
d cs:100 110
d,cs:100,110
```

#### Specifying valid address entries

An *address* parameter in a Debug command specifies a location in memory. *Address* is a two-part designation containing either an alphabetic segment register or a 4-digit segment address, plus an offset value. You can omit the segment register or segment address. The default segment for the **a**, **g**, **I**, **t**, **u**, and **w** commands is CS. The default segment for all other commands is DS. All numeric values are in hexadecimal format.

The following are valid addresses:

```
CS:0100
04BA:0100
```

The colon between the segment name and the offset value is required.

#### Specifying valid range entries

A *range* parameter in a Debug command specifies a range of memory. You can choose from two formats for *range*: a starting address and an ending address, or a starting address and the length (denoted by **l**) of the range.

For example, both of the following syntaxes specify a 16-byte range beginning at CS:100:

```
cs:100 10f
cs:100 l 10
```

### **Related Commands**

The following commands are Debug commands:

| a (Assemble) | p (Proceed)                         |
|--------------|-------------------------------------|
| c (Compare)  | q (Quit)                            |
| d (Dump)     | r (Register)                        |
| e (Enter)    | s (Search)                          |
| f (Fill)     | t (Trace)                           |
| g (Go)       | u (Unassemble)                      |
| h (Hex)      | w (Write)                           |
| i (Input)    | xa (Allocate Expanded Memory)       |
| l (Load)     | xd (Deallocate Expanded Memory)     |
| m (Move)     | xm (Map Extended Memory Pages)      |
| n (Name)     | xs (Display Expanded Memory Status) |
| o (Output)   |                                     |

[Top Of Page](#)

**Debug: A (Assemble)**

Assembles 8086/8087/8088 mnemonics directly into memory.

This command creates executable machine code from assembly-language statements. All numeric values are in hexadecimal format, and you must type them as 1 to 4 characters. You specify a prefix mnemonic in front of the operation code (opcode) to which it refers.

**Syntax**

**a [address]**

**Parameters**

### **address**

Specifies the location where you type assembly-language mnemonics. You use hexadecimal values for **address** and type each value without the trailing "h" character. If you do not specify an address, **a** starts assembling where it last stopped.

### **Notes**

#### **Using mnemonics**

The segment-override mnemonics are **cs:**, **ds:**, **es:**, and **ss:**. The mnemonic for the far return is **retf**. String-manipulation mnemonics must explicitly state the string size. For example, use **movsw** to move word strings (16 bits), and use **movsb** to move byte strings (8 bits).

#### **Assembling jumps and calls**

The assembler automatically assembles a short, near, or far jump or call, depending on byte displacement, to the destination address. You can override such a jump or call by using a **near** or **far** prefix, as the following example shows:

```
-a0100:0500  
0100:0500 jmp 502 ; a 2-byte short jump  
0100:0502 jmp near 505 ; a 3-byte near jump  
0100:0505 jmp far 50a ; a 5-byte far jump
```

You can abbreviate the **near** prefix to **ne**.

#### **Distinguishing word and byte memory locations**

When an operand can refer to either a word memory location or a byte memory location, you must specify the data type with the prefix **word ptr** or the prefix **byte ptr**. Acceptable abbreviations are **wo** and **by**, respectively. The following example shows the two formats:

```
dec    wo [si]  
neg    byte ptr [128]
```

#### **Specifying operands**

Debug uses the common convention that an operand enclosed in brackets ([ ]) refers to a memory location. This is because Debug cannot otherwise differentiate between an immediate operand and an operand that is a memory location. The following example shows the two formats:

```
mov    ax,21      ; load AX with 21h  
mov    ax,[21]     ; load AX with the  
                  ; contents of  
                  ; memory location 21h
```

#### **Using pseudoinstructions**

Two popular pseudoinstructions are available with the **a** command: the **db** opcode, which assembles byte values directly into memory, and the **dw** opcode, which assembles word values directly into memory.

Following are examples of both pseudoinstructions:

```
db 1,2,3,4,"THIS IS AN EXAMPLE"  
db 'THIS IS A QUOTATION MARK: ''  
db "THIS IS A QUOTATION MARK: ""  
dw 1000,2000,3000,"BACH"
```

#### **Examples**

The **a** command supports all forms of register-indirect commands, as the following example shows:

```
add bx,34[bp+2].[si-1]  
pop [bp+di]  
push [si]
```

All opcode synonyms are also supported, as the following example shows:

```
loopz 100  
loope 100  
ja 200  
jnbe 200
```

For 8087 opcodes, you must specify the **wait** or **fwait** prefix, as the following example shows:

```
fwait fadd st,st(3)      ; this line assembles  
                          ; an fwait prefix
```

#### **Related Commands**

For information about entering data into specific bytes, see the Debug **e** (Enter) command.

For information about disassembling bytes, see the Debug **u** (Unassemble) command.

#### [Top Of Page](#)

**Debug: C (Compare)**

Compares two portions of memory.

**Syntax**

**c range address**

**Parameters**

***range***

Specifies the starting and ending addresses, or the starting address and length, of the first area of memory you want to compare. For information about valid *range* values, see the **debug** command.

***address***

Specifies the starting address of the second area of memory you want to compare. For information about valid *address* values, see the **debug** command.

**Note**

If the *range* and *address* memory areas are identical, Debug displays nothing and returns directly to the Debug prompt. If there are differences, Debug displays them in the following format: address1 byte1 byte2 address2

**Example**

The following commands have the same effect:

```
c100,10f 300
c100110 300
```

Each command compares the block of memory from 100h through 10Fh with the block of memory from 300h through 30Fh.

Debug responds to either of the previous commands with a display similar to the following (assuming DS = 197F):

```
197F:0100 4D E4 197F:0300
197F:0101 67 99 197F:0301
197F:0102 A3 27 197F:0302
197F:0103 35 F3 197F:0303
197F:0104 97 BD 197F:0304
197F:0105 04 35 197F:0305
197F:0107 76 71 197F:0307
197F:0108 E6 11 197F:0308
197F:0109 19 2C 197F:0309
197F:010A 80 0A 197F:030A
197F:010B 36 7F 197F:030B
197F:010C BE 22 197F:030C
197F:010D 83 93 197F:030D
197F:010E 49 77 197F:030E
197F:010F 4F 8A 197F:030F
```

Notice that the addresses 197F:0106 and 197F:0306 are missing from the list. This means that the values in those addresses are identical.

**Top Of Page****Debug: D (Dump)**

Displays the contents of a range of memory addresses.

**Syntax**

**d [range]**

**Parameter*****range***

Specifies the starting and ending addresses, or the starting address and length, of the memory area whose contents you want to display. For information about valid *range* values, see the **debug** command. If you do not specify *range*, Debug displays the contents of 128 bytes, starting at the end of the address range specified in the previous **d** command.

**Note**

When you use the **d** command, Debug displays memory contents in two portions: a hexadecimal portion (each byte value is shown in hexadecimal format) and an ASCII portion (each byte value is shown as an ASCII character). Each nonprinting character is denoted by a period (.) in the ASCII portion of the display. Each display line shows the contents of 16 bytes, with a hyphen between the eighth and ninth bytes. Each display line begins on a 16-byte boundary.

**Examples**

Suppose you type the following command:

```
dcs:100 10f
```

Debug displays the the contents of the range in the following format:

```
04BA:0100 54 4F 4D 00 53 41 57 59-45 52 00 00 00 00 00 00 TOM.SAWYER.....
```

If you type the **d** command without parameters, Debug formats the display as described in the previous example. Each line of the display begins with an address that is 16 bytes greater than the address on the previous line (or 8 bytes if you have a 40-column screen).

For each subsequent **d** command you type without parameters, Debug displays the bytes immediately following those last displayed.

If you type the following command, Debug displays the contents of 20h bytes, starting at CS:100:

```
dcs:100 1 20
```

If you type the following command, Debug displays the contents of all bytes in the range of lines from 100h through 115h in the CS segment:

```
dcs:100 115
```

### Related Commands

For information about displaying the contents of registers, see the Debug **r** (Register) command.

For information about disassembling bytes, see the Debug **u** (Unassemble) command.

#### [Top Of Page](#)

Debug: E (Enter)

Enters data into memory at the address you specify.

You can type data in either hexadecimal or ASCII format. Any data previously stored at the specified address is lost.

Syntax

```
e address [list]
```

Parameters

**address**

Specifies the first memory location where you want to enter data.

**list**

Specifies the data you want to enter into successive bytes of memory.

Notes

#### Using the address parameter

If you specify a value for **address** without specifying a value for the optional **list** parameter, Debug displays the address and its contents, repeats the address on the next line, and waits for your input. At this point, you can perform one of the following actions:

- Replace the byte value. To do this, you type a new value after the current value. If the value you type is not a valid hexadecimal value or if it contains more than two digits, Debug does not echo the invalid or extra character.
- Advance to the next byte. To do this, you press the SPACEBAR. To change the value in that byte, type a new value after the current value. If you move beyond an 8-byte boundary when you press the SPACEBAR, Debug starts a new display line and displays the new address at the beginning of the line.
- Return to the preceding byte. To do this, you press the HYPHEN key. You can press the HYPHEN key repeatedly to move back more than 1 byte. When you press HYPHEN, Debug starts a new line and displays the current address and byte value.
- Stop the **e** command. To do this, you press the ENTER key. You can press ENTER at any byte position.

#### Using the list parameter

If you specify values for the **list** parameter, the **e** command sequentially replaces the existing byte values with the values from the list. If an error occurs, no byte values are changed.

**List** values can be either hexadecimal byte values or strings. You separate values by using a space, a comma, or a tab character. You must enclose strings within single or double quotation marks.

#### Examples

Suppose you type the following command:

```
ecs:100
```

Debug displays the contents of the first byte in the following format:

```
04BA:0100 EB._
```

To change this value to 41, type **41** at the cursor, as follows:

```
04BA:0100 EB.41_
```

You can type consecutive byte values with one **e** command. Instead of pressing ENTER after typing the new value, press the SPACEBAR. Debug displays the next value. In this example, if you press the SPACEBAR three times, Debug displays the following values:

```
04BA:0100 EB.41 10. 00. BC._
```

To change the hexadecimal value BC to 42, type **42** at the cursor, as follows:

```
04BA:0100 EB.41 10. 00. BC.42_
```

Now suppose that you decide the value 10 should be 6F. To correct this value, press the HYPHEN key twice to return to address 0101 (value 10). Debug displays the following:

```
04BA:0100 EB.41 10. 00. BC.42-
```

```
04BA:0102 00.-
```

```
04BA:0101 10._
```

Type **6f** at the cursor to change the value, as follows:

```
04BA:0101 10.6f_
```

Press ENTER to stop the **e** command and return to the Debug prompt.

The following is an example of a string entry:

```
eds:100 "This is the text example"
This string will fill 24 bytes, starting at DS:100.
```

#### Related Commands

For information about assembling mnemonics, see the Debug **a** (Assemble) command.

For information about displaying the contents of a portion of memory, see the Debug **d** (Dump) command.

#### [Top Of Page](#)

##### Debug: F (Fill)

Fills addresses in the specified memory area with values you specify.

You can specify data in either hexadecimal or ASCII format. Any data you previously stored at the specified address is lost.

##### Syntax

```
f range list
```

##### Parameters

###### **range**

Specifies the starting and ending addresses, or the starting address and length, of the memory area you want to fill. For information about valid *range* values, see the **debug** command.

###### **list**

Specifies the data you want to enter. *List* can consist of hexadecimal numbers or a string enclosed in quotation marks.

##### Notes

###### Using the range parameter

If *range* contains more bytes than the number of values in *list*, Debug assigns the values in *list* repeatedly until all bytes in *range* are filled.

If any of the memory in *range* is bad or doesn't exist, Debug displays an error message and stops the **f** command.

###### Using the list parameter

If *list* contains more values than the number of bytes in *range*, Debug ignores the extra values in *list*.

##### Example

Suppose you type the following command:

```
f04ba:1001100 42 45 52 54 41
```

In response, Debug fills memory locations 04BA:100 through 04BA:1FF with the values specified. Debug repeats the five values until all the 100h bytes are filled.

#### [Top Of Page](#)

##### Debug: G (Go)

Runs the program currently in memory.

##### Syntax

```
g [=address] [breakpoints]
```

##### Parameters

###### **= address**

Specifies the address in the program currently in memory at which you want execution to begin. If you do not specify *address*, MS-DOS begins program execution at the current address in the CS:IP registers.

###### **breakpoints**

Specifies 1 to 10 temporary breakpoints that you can set as part of the **g** command.

##### Notes

###### Using the address parameter

You must precede the *address* parameter with an equal sign (=) to distinguish the starting address (*address*) from the breakpoint addresses (*breakpoints*).

###### Specifying breakpoints

The program stops at the first breakpoint it encounters, regardless of where you typed that breakpoint in the *breakpoints* list. Debug replaces the original instruction at each breakpoint with an interrupt code.

When the program reaches a breakpoint, Debug restores all breakpoint addresses to their original instructions and displays the contents of all registers, the status of all flags, and the decoded form of the last instruction executed. Debug displays the same information as it would display if you used the Debug **r** (register) command and specified the breakpoint address.

If you do not stop the program at one of the breakpoints, Debug does not replace the interrupt codes with the original instructions.

###### Limitations on setting breakpoints

You can set breakpoints only at addresses containing the first byte of an 8086 operation code (opcode). If you set more than 10 breakpoints, Debug displays the following message:

```
bp Error
```

###### Requirements for the user stack pointer

The user stack pointer must be valid and must have 6 bytes available for the **g** command. This command uses an **iret** instruction to jump to the program being tested. Debug sets the user stack pointer and pushes the user flags, the code segment register, and the instruction pointer onto the user stack. (If the user stack is not valid or is too small, the operating system might fail.) Debug places an interrupt code (0CCh) at the specified breakpoint address(es).

#### **Restarting a program**

Do not attempt to restart a program after MS-DOS displays the following message:

Program terminated normally

To run the program properly, you must reload it by using the Debug **n** (name) and **I** (load) commands.

#### **Examples**

Suppose you type the following command:

**gcs:7550**

MS-DOS runs the program currently in memory up to the breakpoint address 7550 in the CS segment.

Debug then displays the contents of the registers and the status of the flags and stops the **g** command.

The following command sets two breakpoints:

**gcs:7550, cs:8000**

If you type the **g** command again after Debug encounters a breakpoint, execution begins at the instruction after the breakpoint, rather than at the usual starting address.

#### **Related Commands**

For information about executing a loop, a repeated string instruction, a software interrupt, or a subroutine, see the Debug **p** (Proceed) command.

For information about executing one instruction, see the Debug **t** (Trace) command.

#### [Top Of Page](#)

**Debug: H (Hex)**

Performs hexadecimal arithmetic on two parameters you specify.

#### **Syntax**

**h value1 value2**

#### **Parameters**

**value1**

Represents any hexadecimal number in the range 0 through FFFFh.

**value2**

Represents a second hexadecimal number in the range 0 through FFFFh.

#### **Note**

Debug first adds the two parameters you specify and then subtracts the second parameter from the first. The results of these calculations are displayed on one line — first the sum, then the difference.

#### **Example**

Suppose you type the following command:

**h19f 10a**

Debug performs the calculations and displays the following result:

**02A9 0095**

#### [Top Of Page](#)

**Debug: I (Input)**

Reads and displays one byte value from the port you specify.

#### **Syntax**

**i port**

#### **Parameter**

**port**

Specifies the input port by *address*. The address can be a 16-bit value.

#### **Example**

Suppose you type the following command:

**i2f8**

Suppose also that the byte value at the port is 42h. Debug reads the byte and then displays the value, as follows:

**42**

#### **Related Command**

For information about sending the value of a byte to an output port, see the Debug **o** (Output) command.

#### [Top Of Page](#)

**Debug: L (Load)**

Loads a file or contents of specific disk sectors into memory.

To load the contents of the number of bytes specified in the BX:CX registers from a disk file, use the following syntax:

**Syntax I [address]**

To bypass the MS-DOS file system and directly load specific sectors, use the following syntax: **I** address  
drive start number

#### **Parameters**

##### **address**

Specifies the memory location where you want to load the file or the sector contents. If you do not specify *address*, Debug uses the current address in the CS register.

##### **drive**

Specifies the drive that contains the disk from which specific sectors are to be read. This value is numeric: 0 = A, 1 = B, 2 = C, and so on. You use the *drive*, *start*, and *number* parameters only if you want to load the contents of specific sectors rather than load the file specified on the **debug** command line or in the most recent Debug **n** (name) command.

##### **start**

Specifies the hexadecimal number of the first sector whose contents you want to load.

##### **number**

Specifies the hexadecimal number of consecutive sectors whose contents you want to load.

#### **Notes**

##### **Using the I command without parameters**

When you use the **I** command without parameters, the file you specified on the **debug** command line is loaded into memory, beginning at address CS:100. Debug also sets the BX and CX registers to the number of bytes loaded. If you did not specify a file on the **debug** command line, the file loaded is the one you most recently specified by using the **n** command.

##### **Using the I command with the address parameter**

If you use the **I** command with the *address* parameter, Debug begins loading the file or the contents of the specified sectors at the memory location *address*.

##### **Using the I command with all parameters**

If you use the **I** command with all parameters, Debug loads the contents of specific disk sectors instead of loading a file.

##### **Loading the contents of specific sectors**

Each sector in the range you specify is read from *drive*. Debug begins loading with *start* and continues until the contents of the number of sectors specified in *number* have been loaded.

##### **Loading an .EXE file**

Debug ignores the *address* parameter for .EXE files. If you specify an .EXE file, Debug relocates the file to the loading address specified in the header of the .EXE file. The header itself is stripped off the .EXE file before the file is loaded into memory, so the size of an .EXE file on disk differs from its size in memory. If you want to examine a complete .EXE file, rename the file with a different extension.

##### **Opening a hex file**

A *hex file* is a file that uses the Intel hexadecimal format, as described in *The MS-DOS Encyclopedia*. Debug assumes that files with the .HEX extension are hexadecimal-format files. You can type the **I** command with no parameters to load a hex file beginning at the address specified in the hex file. If the **I** command you type includes the *address* parameter, Debug adds the specified address to the address found in the hex file to determine the starting address.

#### **Examples**

Suppose you start Debug and type the following command:

`nfile.com`

You can now type the **I** command to load FILE.COM. Debug loads the file and displays the Debug prompt.

Suppose that you want to load the contents of 109 (6Dh) sectors from drive C, beginning with logical sector 15 (0Fh), into memory beginning at address 04BA:0100. To do this, type the following command:

`104ba:100 2 0f 6d`

#### **Related Commands**

For information about specifying a file for the **I** command, see the Debug **n** (Name) command.

For information about writing the file being debugged to a disk, see the Debug **w** (Write) command.

#### [Top Of Page](#)

##### **Debug: M (Move)**

Copies the contents of a block of memory to another block of memory.

#### **Syntax**

`m range address`

#### **Parameters**

##### **range**

Specifies the starting and ending addresses, or the starting address and the length, of the memory area whose contents you want to copy.

##### **address**

Specifies the starting address of the location to which you want to copy the contents of *range*.

## Notes

### Effects of the copy operation on existing data

If the addresses in the block being copied do not have new data written to them, the original data remains intact. However, if the destination block already contains data (as it might in an *overlapping copy operation*), that data is overwritten. (Overlapping copy operations are those in which part of the destination block overlaps part of the source block.)

### Performing overlapping copy operations

The **m** command performs overlapping copy operations without losing data at the destination addresses. The contents of addresses that will be overwritten are copied first. Thus, if data is to be copied from higher addresses to lower addresses, the copy operation begins at the source block's lowest address and progresses toward the highest address. Conversely, if data is to be copied from lower addresses to higher addresses, the copy operation begins at the source block's highest address and progresses toward the lowest address.

### Example

Suppose you type the following command:

```
mcs:100 110 cs:500
```

Debug first copies the contents of address CS:110 to CS:510, then copies the contents of CS:10F to CS:50F, and so on until it has copied the contents of CS:100 to CS:500. To view the results, you can use the Debug **d** (dump) command, specifying the destination address you used with the **m** command.

### [Top Of Page](#)

### Debug: N (Name)

Specifies the name of an executable file for a Debug **l** (load) or **w** (write) command, or specifies parameters for the executable file being debugged.

#### Syntax

**n** [*drive:][path]**filename*

To specify parameters for the executable file you are testing, use the following syntax:

**n** *file-parameters*

To clear the current specifications, use the following syntax:

**n**

#### Parameters

**[** *drive :][ path ]* **filename**

Specifies the location and name of the executable file you want to test.

**file-parameters**

Specifies parameters and switches for the executable file you are testing.

#### Notes

### The two uses of the n command

You can use the **n** command in two ways. First, you can use it to specify a file to be used by a later **l** or **w** command. If you start Debug without naming a file to be debugged, you must use the command **n***filename* before you can use the **l** command to load the file. The filename is correctly formatted for a file control block at CS:5C. Second, you can use the **n** command to specify command-line parameters and switches for the file being debugged.

### Memory areas

| Memory location | Contents                                 |
|-----------------|------------------------------------------|
| CS:5C           | File control block (FCB) for file 1      |
| CS:6C           | File control block (FCB) for file 2      |
| CS:80           | Length of n command line (in characters) |

CS:81

Beginning of n command-line characters

The first filename you specify for the **n** command is placed in a file control block (FCB) at CS:5C. If you specify a second filename, this name is placed in an FCB at CS:6C. The number of characters typed on the **n** command line (exclusive of the first character, **n**) is stored at location CS:80. The actual characters on the **n** command line (again, exclusive of the letter **n**) are stored beginning at CS:81. Note that these characters can be any switches and delimiters that would be legal in a command typed at the MS-DOS prompt.

#### Examples

Suppose you've started Debug and loaded the program PROG.COM for debugging. You subsequently decide to specify two parameters for PROG.COM and run the program. Following is the sequence of commands for this example:

```
debug prog.com
nparam1 param2
g
```

In this case, the Debug **g** (go) command runs the program as if you had typed the following command at the MS-DOS prompt:

```
prog param1 param2
```

Testing and debugging therefore reflect a typical run-time environment for PROG.COM.

In the following sequence of commands, the first **n** command specifies FILE1.EXE as the file for the subsequent **l** command, which loads FILE1.EXE into memory. The second **n** command specifies the parameters to be used by FILE1.EXE. Finally, the **g** command runs FILE1.EXE as if you had typed **file1 file2.dat file3.dat** at the MS-DOS prompt.

```
nfile1.exe
l
nfile2.dat file3.dat
g
```

Note that you do not use the **l** command after the second form of the **n** command. Also note that if you now use the **w** command, MS-DOS saves FILE1.EXE, the file being debugged, with the name FILE2.DAT. To avoid this result, you should always use the first form of the **n** command immediately before either an **l** or a **w** command.

#### Related Commands

For information about loading the contents of a file or of specific disk sectors into memory, see the Debug **l** (Load) command.

For information about writing the file being debugged to a disk, see the Debug **w** (Write) command.

#### [Top Of Page](#)

#### Debug: O (Output)

Sends the value of a byte to an output port.

##### Syntax

**o port byte-value**

##### Parameters

##### **port**

Specifies the output port by address. The port address can be a 16-bit value.

##### **byte-value**

Specifies the byte value you want to direct to **port**.

##### Example

To send the byte value 4Fh to the output port at address 2F8h, type the following command:

```
o2f8 4f
```

#### Related Command

For information about reading the value of a byte from an input port, see the Debug **i** (Input) command.

#### [Top Of Page](#)

#### Debug: P (Proceed)

Executes a loop, a repeated string instruction, a software interrupt, or a subroutine; or traces through any other instruction.

##### Syntax

**p [=address] [number]**

##### Parameters

##### **= address**

Specifies the location of the first instruction to execute. If you do not specify an address, the default address is the current address specified in the CS:IP registers.

##### **number**

Specifies the number of instructions to execute before returning control to Debug. The default value is 1.

#### Notes

##### Transferring control to the program being tested

When the **p** command transfers control from Debug to the program being tested, that program runs without interruption until the loop, repeated string instruction, software interrupt, or subroutine at the specified address is completed, or until the specified number of machine instructions have been executed. Control then returns to Debug.

##### Limitations on the address parameter

If the *address* parameter does not specify a segment, Debug uses the CS register of the program being tested. If you omit *address*, the program is executed beginning at the address specified by its CS:IP registers. You must precede the *address* parameter with an equal sign (=) to distinguish it from the *number* parameter. If the instruction at the specified address is not a loop, a repeated string instruction, a software interrupt, or a subroutine, the **p** command works the same way as the Debug **t** (trace) command.

##### Messages displayed with the p command

After **p** executes an instruction, Debug displays the contents of the program's registers, the status of its flags, and the decoded form of the next instruction to be executed.

**Caution:** You cannot use the **p** command to trace through read-only memory (ROM).

#### Example

Suppose that the program you're testing contains a **call** instruction at address CS:143F. To run the subroutine that is the destination of **call** and then return control to Debug, type the following command:  
**p=143f**

Debug displays the results in the following format:

```
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=2246 ES=2246 SS=2246 CS=2246 IP=1443 NV UP EI PL NZ AC PO NC
2246:1442 7505          JNZ     144A
```

#### Related Commands

For information about running the program currently in memory, see the Debug **g** (Go) command.

For information about executing one instruction, see the Debug **t** (Trace) command.

#### [Top Of Page](#)

##### Debug: Q (Quit)

Stops the Debug session without saving the file currently being tested.

After you type **q**, control returns to MS-DOS.

Syntax **q**

#### Example

To stop the debugging session, type the following command: **q**

MS-DOS displays the MS-DOS prompt.

#### Related Command

For information about saving a file, see the Debug **w** (Write) command.

#### [Top Of Page](#)

##### Debug: R (Register)

Displays or alters the contents of one or more central-processing-unit (CPU) registers.

Syntax **r [register-name]**

To display the contents of all registers and flags in the register storage area, use the following syntax: **r**

#### Parameter

##### **register-name**

Specifies the name of the register whose contents you want to display.

#### Notes

##### Using the r command

If you specify a register name, MS-DOS displays the 16-bit value of that register in hexadecimal notation and displays a colon as the prompt. If you want to change the value contained in the register, type a new value and press ENTER; otherwise, just press ENTER to return to the Debug prompt.

##### Valid register names

The following are valid values for register-name: **ax**, **bx**, , **dx**, **sp**, **bp**, **si**, **di**, **ds**, **es**, **ss**, **cs**, **ip**, **pc**, and **f**. Both **ip** and **pc** refer to the instruction pointer.

If you specify a register name other than one from the preceding list, MS-DOS displays the following message:

```
br error
```

##### Using the f character instead of a register name

If you type the **f** character instead of a register name, Debug displays the current setting of each flag as a two-letter code and then displays the Debug prompt. To change the setting of a flag, type the appropriate two-letter code from the following table:

| Flag name       | Set            | Clear          |
|-----------------|----------------|----------------|
| Overflow        | ov             | nv             |
| Direction       | dn (decrement) | up (increment) |
| Interrupt       | ei (enabled)   | di (disabled)  |
| Sign            | ng (negative)  | pl (positive)  |
| Zero            | zr             | nz             |
| Auxiliary Carry | ac             | na             |
| Parity          | pe (even)      | po (odd)       |
| Carry           | cy             | nc             |

You can type new flag values in any order. You need not leave spaces between these values. To stop the **r** command, press ENTER. Any flags for which you did not specify new values remain unchanged.

#### Messages displayed with the **r** command

If you specify more than one value for a flag, Debug displays the following message:

**df error**

If you specify a flag code not listed in the preceding table, Debug displays the following message:

**bf error**

In both cases, Debug ignores all settings specified after the invalid entry.

#### Default settings for Debug

When you start Debug, the segment registers are set to the bottom of free memory, the instruction pointer is set to 0100h, all flags are cleared, and the remaining registers are set to zero, except for **sp**, which is set to FFFEH.

#### Examples

To view the contents of all registers, the status of all flags, and the decoded form of the instruction at the current location, type the following command:

**r**

If the current location is CS:11A, the display will look similar to the following:

```
AX=0E00 BX=00FF CX=0007 DX=01FF SP=039D BP=0000 SI=005C DI=0000
DS=04BA ES=04BA SS=04BA CS=04BA IP=011A NV UP DI NG NZ AC PE NC
04BA:011A CD21          INT    21
```

To view only the status of the flags, type the following command:

**rf**

Debug displays the information in the following format:

```
NV UP DI NG NZ AC PE NC - _
```

Now you can type one or more valid flag values, in any order, with or without spaces, as in the following command:

```
nv up di ng nz ac pe nc - pleicy
```

Debug stops the **r** command and displays the Debug prompt. To see the changes, type either the **r** or **rf** command. Debug then displays the following:

```
NV UP EI PL NZ AC PE CY -
```

Press ENTER to return to the Debug prompt.

#### Related Commands

For information about displaying the contents of a portion of memory, see the Debug **d** (Dump) command.

For information about disassembling bytes, see the Debug **u** (Unassemble) command.

#### [Top Of Page](#)

Debug: S (Search)

Searches a range of addresses for a pattern of one or more byte values.

Syntax

```
s range list
```

#### Parameters

**range**

Specifies the beginning and ending addresses of the range you want to search. For information about valid values for the *range* parameter, see the **debug** command.

**list**

Specifies the pattern of one or more byte values or a string you want to search for. Separate each byte value from the next with a space or a comma. Enclose string values in quotation marks.

#### Note

If the *list* parameter contains more than one byte value, Debug displays only the first address where the byte value occurs. If *list* contains only one byte value, Debug displays all addresses where the value occurs in the specified range.

#### Examples

Suppose you want to find all addresses in the range CS:100 through CS:110 that contain the value 41. To do this, type the following command:

```
scs:100 110 41
```

Debug displays the results in the following format:

```
04BA:0104
```

```
04BA:010D
```

-

The following command searches for the string "Ph" in the range CS:100 through CS:1A0:

```
scs:100 1a0 "Ph"
```

#### [Top Of Page](#)

Debug: T (Trace)

Executes one instruction and displays the contents of all registers, the status of all flags, and the decoded form of the instruction executed.

Syntax t [=address] [number]

#### Parameters

**= address**

Specifies the address at which Debug is to start tracing instructions. If you omit the *address* parameter, tracing begins at the address specified by your program's CS:IP registers. For information about valid values for the *address* parameter, see the **debug** command.

**number**

Specifies the number of instructions to be traced. This value must be a hexadecimal number. The default value is 1.

#### Notes

##### Tracing instructions in read-only memory

The **t** command uses the hardware trace mode of the 8086 or 8088 microprocessor. Therefore, you can also trace instructions stored in read-only memory (ROM).

##### Using the address parameter

You must precede the *address* parameter with an equal sign (=) to distinguish it from the *number* parameter.

##### Example

To execute one instruction (pointed to by CS:IP), and then display the contents of the registers, the status of the flags, and the decoded form of the instruction, type the following command:

t

If the position of the instruction in the program were 04BA:011A, Debug might display the following information:

```
AX=0E00 BX=00FF CX=0007 DX=01FF SP=039D BP=0000 SI=005C DI=0000
```

```
DS=04BA ES=04BA SS=04BA CS=04BA IP=011A NV UP DI NG NZ AC PE NC  
04BA:011A CD21 INT 21
```

#### **Related Commands**

For information about executing a loop, a repeated string instruction, a software interrupt, or a subroutine, see the Debug **p** (Proceed) command.

For information about executing the program currently in memory, see the Debug **g** (Go) command.

#### [Top Of Page](#)

##### **Debug: U (Unassemble)**

Disassembles bytes and displays their corresponding source statements, including addresses and byte values. The disassembled code looks like a listing for an assembled file.

##### **Syntax u [range]**

To disassemble 20h bytes (the default number), beginning at the first address after the address displayed by the previous **u** command, use the following syntax: **u**

##### **Parameter**

##### **range**

Specifies the starting and ending addresses, or the starting address and length, of the code you want to disassemble. For information about valid values for the *range* parameter, see the **debug** command.

##### **Examples**

To disassemble 16 (10h) bytes, beginning at address 04BA:0100, type the following command:

```
u04ba:100110
```

Debug displays the results in the following format:

```
04BA:0100 206472 AND [SI+72],AH  
04BA:0103 69 DB 69  
04BA:0104 7665 JBE 016B  
04BA:0106 207370 AND [BP+DI+70],DH  
04BA:0109 65 DB 65  
04BA:010A 63 DB 63  
04BA:010B 69 DB 69  
04BA:010C 66 DB 66  
04BA:010D 69 DB 69  
04BA:010E 63 DB 63  
04BA:010F 61 DB 61
```

To display only the information for the specific addresses 04BA:0100 through 04BA:0108, type the following command:

```
u04ba:0100 0108
```

Debug displays the following:

```
04BA:0100 206472 AND [SI+72],AH  
04BA:0103 69 DB 69  
04BA:0104 7665 JBE 016B  
04BA:0106 207370 AND [BP+DI+70],DH
```

#### **Related Commands**

For information about assembling mnemonics, see the Debug **a** (Assemble) command.

For information about displaying the contents of a portion of memory, see the Debug **d** (Dump) command.

#### [Top Of Page](#)

##### **Debug: W (Write)**

Writes a file or specific sectors to disk.

You must have specified the name of the disk file when you started Debug or in the most recent Debug **n** (name) command. Both of these methods properly format a filename for a file control block at address CS:5C.

To write the contents of the number of bytes specified in the BX:CX registers to a disk file, use the following syntax:

##### **Syntax w [address]**

To bypass the MS-DOS file system and directly write specific sectors, use the following syntax: **w address** **drive start number**

**Caution:** Writing specific sectors is extremely risky because it bypasses the MS-DOS file handler. The disk's file structure can easily be damaged if the wrong values are typed.

##### **Parameters**

##### **address**

Specifies the beginning memory address of the file, or portion of the file, you want to write to a disk file. If you do not specify **address**, Debug starts from CS:100. For information about valid values for the **address** parameter, see the **debug** command.

##### **drive**

Specifies the drive that contains the destination disk. This value is numeric: 0 = A, 1 = B, 2 = C, and so on.

**start**

Specifies the hexadecimal number of the first sector to which you want to write.

**number**

Specifies the number of sectors to which you want to write.

**Notes**

**Resetting BX:CX before using the w command without parameters**

If you have used a Debug **g** (go), **t** (trace), **p** (proceed), or **r** (register) command, you must reset the BX:CX registers before using the **w** command without parameters.

**Writing a modified file to a disk**

If you modify the file but do not change the name, length, or starting address, Debug can still correctly write the file to the original disk location.

**Limitation on the w command**

You cannot write an .EXE or .HEX file with this command.

**Example**

Suppose you want to write the contents of memory, beginning at the address CS:100, to the disk in drive B. You want the data to begin in the disk's logical sector number 37h and continue for 2Bh sectors. To do this, type the following command:

```
wcs:100 1 37 2b
```

When the write operation is complete, Debug displays the Debug prompt again.

**Related Commands**

For information about specifying a file for the **w** command, see the Debug **n** (Name) command.

For information about loading the contents of a file or file sectors into memory, see the Debug **l** (Load) command.

[\*\*Top Of Page\*\*](#)

**Debug: XA (Allocate Expanded Memory)**

Allocates a specified number of pages of expanded memory.

To use expanded memory, you must have installed an expanded-memory device driver that conforms to version 4.0 of the Lotus/Intel/Microsoft Expanded Memory Specification (LIM EMS).

**Syntax**

**xa [count]**

**Parameter**

**count**

Specifies the number of 16-kilobyte pages of expanded memory to allocate.

**Example**

To deallocate handle 0003, type the following command:

```
xd 0003
```

If the command is successful, Debug displays the following message:

```
Handle 0003 deallocated
```

**Related Commands**

For information about other Debug commands that work with expanded memory, see the Debug commands **xd** (Deallocate Expanded Memory), **xm** (Map Expanded-Memory Pages), and **xs** (Display Expanded-Memory Status).

[\*\*Top Of Page\*\*](#)

**Debug: XD (Deallocate Expanded Memory)**

Deallocates a handle to expanded memory.

To use expanded memory, you must have installed an expanded-memory device driver that conforms to version 4.0 of the Lotus/Intel/Microsoft Expanded Memory Specification (LIM EMS).

**xd [handle]**

**Parameters**

**handle**

Specifies the handle you want to deallocate.

**Example**

To deallocate handle 0003, type the following command:

```
xd 0003
```

If the command is successful, Debug displays the following message:

```
Handle 0003 deallocated
```

**Related Commands**

For information about other Debug commands that work with expanded memory, see the Debug commands **xa** (allocate expanded memory), **xm**(map expanded-memory pages), and **xs** (display expanded-memory status).

[\*\*Top Of Page\*\*](#)

## Debug: XM (Map Expanded Memory Pages)

Maps a logical page of expanded memory, belonging to the specified handle, to a physical page of expanded memory.

To use expanded memory, you must have installed an expanded-memory device driver that conforms to version 4.0 of the Lotus/Intel/Microsoft Expanded Memory Specification (LIM EMS).

### Syntax

**xm** [/page] [ppage] [handle]

### Parameters

#### *lpage*

Specifies the number of the logical page of expanded memory that you want to map to physical page *ppage*.

#### *ppage*

Specifies the number of the physical page to which *lpage* is to be mapped.

#### *handle*

Specifies the handle.

### Example

To map logical page 5 of handle 0003 to physical page 2, type the following command:

```
xm 5 2 0003
```

If the command is successful, Debug displays the following message:

```
Logical page 05 mapped to physical page 02
```

### Related Commands

For information about other Debug commands that work with expanded memory, see the Debug commands **xa** (Allocate Expanded Memory), **xd** (Deallocate Expanded Memory), and **xs** (Display Expanded-Memory Status).

### [Top Of Page](#)

## Debug: XS (Display Expanded-Memory Status)

Displays information about the status of expanded memory.

To use expanded memory, you must have installed an expanded-memory device driver that conforms to version 4.0 of the Lotus/Intel/Microsoft Expanded Memory Specification (LIM EMS).

### Syntax

#### **xs**

### Note

The information that Debug displays has the following format:

```
Handle xx has xx pages allocated  
Physical page xx = Frame segment xx  
xx of a total xx EMS pages have been allocated  
xx of a total xx EMS handles have been allocated
```

### Example

To display expanded-memory information, type the following command:

```
xs
```

```
Debug displays information similar to the following:  
Handle 0000 has 0000 pages allocated  
Handle 0001 has 0002 pages allocated  
Physical page 00 = Frame segment C000  
Physical page 01 = Frame segment C400  
Physical page 02 = Frame segment C800  
Physical page 03 = Frame segment CC00  
 2 of a total 80 EMS pages have been allocated  
 2 of a total FF EMS handles have been allocated
```

### Related Commands

For information about other Debug commands that work with expanded memory, see the Debug commands **xa** (Allocate Expanded Memory), **xd** (Deallocate Expanded Memory), and **xm** (Map Expanded-Memory Pages).

### [Top Of Page](#)

## Defrag

Reorganizes the files on a disk to optimize disk performance. Do not use this command when you are running Windows.

### Syntax

```
defrag [drive:] [/f] [/s[:order] [/b] [/skiphigh] [/lcd | /bw | /g0] [/h]  
defrag [drive:] [/u] [/b] [/skiphigh] [/lcd | /bw | /g0] [/h]
```

### Parameter

#### *drive :*

Specifies the drive that contains the disk you want to optimize.

**Switches****/f**

Defragments files and ensures that the disk contains no empty spaces between files.

**/u**

Defragments files and leaves empty spaces, if any, between files.

**/s**

Controls how the files are sorted in their directories. If you omit this switch, **defrag** uses the current order on the disk. The colon (:) is optional. The following list describes each of the values you can use to sort files. Use any combination of the values, and do not separate these values with spaces.

|    |                                                        |
|----|--------------------------------------------------------|
| n  | In alphabetic order by name                            |
| n- | In reverse alphabetic order by name (Z through A)      |
| e  | In alphabetic order by extension                       |
| e- | In reverse alphabetic order by extension (Z through A) |
| d  | By date and time, earliest first                       |
| d- | By date and time, latest first                         |
| s  | By size, smallest first                                |
| s- | By size, largest first                                 |

**/b**

Restarts your computer after files have been reorganized.

**/skiphigh**Loads **defrag** into conventional memory. By default, **defrag** is loaded into upper memory, if upper memory is available.**/lcd**Starts **defrag** using an LCD color scheme.**/bw**Starts **defrag** using a black and white color scheme.**/g0**

Disables the graphic mouse and graphic character set.

**/h**

Moves hidden files.

**Notes****Network and INTERLNK drives**You cannot use **defrag** to optimize network drives or drives created with INTERLNK.**Disk information reported by defrag and chkdsk**

Disk information that **defrag** reports differs from information that **chkdsk** reports. **Defrag** reports hidden and user files as one number; **chkdsk** reports numbers for each type. **Defrag** counts the root as a directory; **chkdsk** does not. **Defrag** does not count the volume label as a file; **chkdsk** does.

#### Start defrag only from MS-DOS

If you start **defrag** from a program such as Microsoft Windows, you may lose data.

#### Defrag exit codes

The following list briefly describes the meaning of each **defrag** exit code (**errorlevel** parameter):

|   |                                                                                                |
|---|------------------------------------------------------------------------------------------------|
| 0 | The defragmentation was successful.                                                            |
| 1 | An internal error occurred.                                                                    |
| 2 | The disk contained no free clusters. To operate, defrag needs 1 free cluster.                  |
| 3 | The user pressed CTRL+C to stop the process.                                                   |
| 4 | A general error occurred.                                                                      |
| 5 | Defrag encountered an error while reading a cluster.                                           |
| 6 | Defrag encountered an error while writing a cluster.                                           |
| 7 | An allocation error occurred. To correct the error, use the chkdsk command with the /f switch. |
| 8 | A memory error occurred.                                                                       |
| 9 | There was insufficient memory to defragment the disk.                                          |

You can use the **errorlevel** parameter on the **if** command line in a batch program to process exit codes returned by **defrag**. For an example of a batch program that processes exit codes, see the **diskcomp** command.

#### Example

To load **defrag** into conventional memory and specify that **defrag** sort files according to the date they were created, from latest created to earliest created, type the following command:

```
defrag c: /f /sd- /skiphigh
```

This example fully optimizes drive C, but slows **defrag**.

[Top Of Page](#)

Del (Erase)

Deletes the files you specify.

**Syntax**

**del** [drive:]pathfilename [/p]  
**erase** [drive:]pathfilename [/p]

**Parameter**

**[ drive :][ path ] filename**

Specifies the location and name of the file or set of files you want to delete.

**Switch**

**/p**

Prompts you for confirmation before deleting the specified file.

**Notes****Using the /p switch**

If you use the **/p** switch, **del** displays the name of a file and prompts you with a message in the following format:

filename, Delete (Y/N) ?

Press Y to confirm the deletion, N to cancel the deletion and display the next filename (if you specified a group of files), or CRTL+C to stop the **del** command.

**Deleting more than one file at a time**

You can delete all the files in a directory by typing the **del** command followed by [drive:]path. You can also use wildcards (\*) and (?) to delete more than one file at a time. However, you should use wildcards cautiously with the **del** command to avoid deleting files unintentionally. Suppose you type the following command:

**del \*.\***

**Del** displays the following prompt:

All files in directory will be deleted!

Are you sure (Y/N) ?

Press Y and then ENTER to delete all files in the current directory, or press N and then ENTER to cancel the deletion.

Before you use wildcards with the **del** command to delete a group of files, you can use the same wildcards with the **dir** command to see a list of the names of all the files included in the group.

**Caution:** Once you delete a file from your disk, you may not be able to retrieve it. Although the **undelete** command can retrieve deleted files, it can do so with certainty only if no other files have been created or changed on the disk. If you accidentally delete a file that you want to keep, stop what you are doing and immediately use the **undelete** command to retrieve the file.

For more information on undeleting files, see the chapter "Managing Your System" in the *MS-DOS 6 User's Guide*.

**Examples**

To delete the CAT.TMP file from the TEST directory on drive C, you can use either of the following commands:

**del c:\test\cat.tmp**  
**erase c:\test\cat.tmp**

To delete all the files in a directory named TEST on drive C, you can use either of the following commands:

**del c:\test**  
**del c:\test\\*.\***

**Related Commands**

For information about retrieving a deleted file, see the **undelete** command.

For information about removing a directory, see the **rmdir** command.

For information about deleting a directory, its files, and all subdirectories and files subordinate to it, see the **deltree** command.

[\*\*Top Of Page\*\*](#)**Deltree**

Deletes a directory and all the files and subdirectories that are in it.

**Syntax**

**deltree [/y] [drive:]path**

**Parameter**

**drive : path**

Specifies the name of the directory you want to delete. The **deltree** command will delete all the files contained in the directory you specify, as well as all subdirectories and files in the subdirectories subordinate to this directory.

**Switch**

**/y**

Carries out the **deltree** command without first prompting you to confirm the deletion.

**Notes****Deltree and Hidden, System, and Read-Only Attributes**

The **deltree** command deletes all files contained in a directory or subdirectory, regardless of attributes.

#### Errorlevel parameters

If **deltree** successfully deleted the directory, it returns an **errorlevel** value of 0.

#### Using wildcards with **deltree**

You can use wildcards with the **deltree** command, but use them with extreme caution. If you specify a wildcard that matches both directory names and filenames, both the directories and files will be deleted. Before specifying wildcards with the **deltree** command, use the **dir** command to view the files and directories you will delete.

#### Example

To delete the TEMP directory on drive C, including all files and subdirectories of the TEMP directory, type the following at the command prompt:

```
deltree c:\temp
```

#### Related Commands

For information about removing a directory, see the **rmdir** command.

For information about deleting files, see the **del** command.

#### [Top Of Page](#)

#### Device

Loads the device driver you specify into memory. You can use this command only in your CONFIG.SYS file.

#### Syntax

```
device=[drive:][path]filename [dd-parameters]
```

#### Parameters

[ **drive :I| path ] filename**

Specifies the location and name of the device driver you want to load.

[ **dd-parameters** ]

Specifies any command-line information required by the device driver.

#### Notes

##### Using standard device drivers

The standard installable device drivers provided with MS-DOS 6 are ANSI.SYS, DISPLAY.SYS, DRIVER.SYS, DBLSPACE.SYS, EGA.SYS, EMM386.EXE, HIMEM.SYS, INTERLNK.EXE, POWER.EXE, RAMDRIVE.SYS, SETVER.EXE, and SMARTDRV.EXE.

The files COUNTRY.SYS and KEYBOARD.SYS are not device drivers. They are data files for the **country** and **keyb** commands, respectively. Do not try to load either of these files with the **device** command. If you do, your system halts, and you cannot restart MS-DOS. For information about loading COUNTRY.SYS, see the **country** command. For information about loading KEYBOARD.SYS, see the **keyb** command.

##### Installing device drivers for other products

When you purchase a mouse, a scanner, or a similar product, the manufacturer usually includes device-driver software. To install a device driver, specify its location and name on a **device** command line.

##### Installing a third-party console driver

If you install both DISPLAY.SYS and a third-party console driver, such as VT52.SYS, the third-party device driver must be installed first. Otherwise, the third-party device driver may disable DISPLAY.SYS.

##### Installing multiple device drivers

Sometimes one installable device driver will require that it be loaded before or after another in your CONFIG.SYS file. For example, EMM386.EXE requires HIMEM.SYS to be loaded first. If the device driver requires that another device driver be loaded before it, make sure the commands are listed in the correct order in your CONFIG.SYS file.

#### Example

If you plan to use an ANSI escape sequence to control the screen and keyboard, you should add the following command to your CONFIG.SYS file (assuming MS-DOS files are in the DOS directory on drive C):

```
device=c:\dos\ansi.sys
```

#### Related Command

For information about loading device drivers into the upper memory area, see the **devicehigh** command.

#### [Top Of Page](#)

#### Devicehigh

Loads device driver you specify into the upper memory area. Loading a device driver into the upper memory area frees more bytes of conventional memory for other programs. If upper memory is not available, the **devicehigh** command functions just like the **device** command.

You can use this command only in your CONFIG.SYS file.

#### Syntax

```
devicehigh [drive:][path]filename [dd-parameters]
```

To specify the region(s) of memory into which to load the device driver, use the following syntax:

```
devicehigh [[/I:region1[,minsize1]][;region2[,minsize2]] [/S]]= [drive:][path]filename [dd-parameters]
```

#### Parameters

**[ drive :][ path ] filename**

Specifies the location and name of the device driver you want to load into the upper memory area.

**dd-parameters**

Specifies any command-line information required by the device driver.

**Switches****/I: region1 [, minsize1 ][; region2 [, minsize2 ]...]**

Specifies one or more regions of memory into which to load the device driver. By default, MS-DOS loads the driver into the largest free upper-memory block (UMB) and makes all other UMBs available for the driver's use. You can use the **/I** switch to load the device driver into a specific region of memory or to specify which region(s) the driver can use.

To load the driver into the largest block in a specific region of upper memory, specify the region number after the **/I** switch. For example, to load the driver into the largest free block in region 4, you would type **/I:4**. (To list the free areas of memory, type **mem /f** at the command prompt.)

When loaded with the **/I** switch, a device driver can use only the specified memory region. Some device drivers use more than one area of memory; for those drivers, you can specify more than one region. (To find out how a particular device driver uses memory, issue the **mem /m** command and specify the device-driver name as an argument.) To specify two or more regions, separate the block numbers with a semicolon (;). For example, to use blocks 2 and 3, you would type **/I:2;3**.

Normally, MS-DOS loads a driver into a UMB in the specified region only if that region contains a UMB larger than the driver's load size (usually equal to the size of the executable program file). If the driver requires more memory while running than it does when loaded, you can use the *minsize* parameter to ensure that the driver will not be loaded into a UMB that is too small for it. If you specify a value for *minsize*, MS-DOS loads the driver into that region only if it contains a UMB that is larger than both the driver's load size and the *minsize* value.

**/S**

Shrinks the UMB to its minimum size while the driver is loading. Using this switch makes the most efficient use of memory. This switch is normally used only by the MemMaker program, which can analyze a device driver's memory use to determine whether the **/S** switch can safely be used when loading that driver. This switch can be used only in conjunction with the **/I** switch and affects only UMBs for which a minimum size was specified.

**Notes****Using the DOS=UMB command**

To use the **devicehigh** command, you must also include the **DOS=UMB** command in your CONFIG.SYS file. If you do not specify this command, all device drivers are loaded into conventional memory, as if you had used the **device** command. For more information, see the DOS command.

**Using MemMaker to optimize upper memory area automatically**

The MemMaker program, included with MS-DOS 6, automatically optimizes your system's memory. MemMaker surveys the upper memory area, analyzes the memory use of your drivers and programs, and determines which drivers and programs fit best into the available UMBs. MemMaker then changes selected **device** commands in your CONFIG.SYS file to **devicehigh** commands and adds **/I** and **/S** switches as necessary. For more information about using MemMaker to optimize your computer's memory, see "Making More Memory Available" in the *MS-DOS 6 User's Guide*.

**Using MS-DOS 5 devicehigh syntax**

The version of **devicehigh** provided with MS-DOS 5 used the following syntax:

```
devicehigh size=hexsize [drive:][path] filename [dd-parameters]
```

Although the MS-DOS 5 **devicehigh** syntax will still work with MS-DOS 6, it is strongly recommended that you use the current **devicehigh** syntax whenever possible.

**Installing HIMEM.SYS and a UMB provider**

To load a device driver into the upper memory area, your computer must have extended memory. You must use the **device** command once to install the HIMEM.SYS device driver and then again to install an upper-memory-block (UMB) provider. These commands must appear before the **devicehigh** command in your CONFIG.SYS file. If your computer has an 80386 or 80486 processor, you can use EMM386.EXE as the UMB provider. If your computer has a different processor, you must supply a different UMB provider.

**If no upper memory area is available**

If there is not enough upper memory area available to load the device driver you specified with the **devicehigh** command, MS-DOS will load it into conventional memory (as if you had used the **device** command).

**Examples**

The following CONFIG.SYS commands make the upper memory area available for running device drivers and programs:

```
device=c:\dos\himem.sys  
device=c:\dos\emm386.exe ram
```

`dos=umb`

The following command directs MS-DOS to load a device driver named MYDRIV.SYS into the upper memory area of an 80386 computer:

`devicehigh=mydriv.sys`

The following CONFIG.SYS command directs MS-DOS to run the MOUSE.SYS driver in the upper memory area and load the driver into upper memory block 2:

`devicehigh=/1:2 C:\drivers\mouse.sys`

The following command loads the MYDRIV.SYS driver into region 1 of upper memory, and also allows the driver to use region 3 if it needs to:

`devicehigh=/1:1;3 C:\util\mydriv.sys`

The following command loads the same driver into upper memory regions 1 and 3, but only if each region is at least 30 bytes in size:

`devicehigh=/1:1,30;3,30 C:\util\mydriv.sys`

### Related Commands

For information about loading programs into the upper memory area, see the **loadhigh** command.

For information about loading device drivers into conventional memory, see the **device** command.

For information about using the MemMaker program to move programs to the upper memory area, see the **memmaker** command.

[Top Of Page](#)

### Dir

Displays a list of the files and subdirectories that are in the directory you specify.

When you use **dir** without parameters or switches, it displays the disk's volume label and serial number; one directory or filename per line, including the filename extension, the file size in bytes, and the date and time the file was last modified; and the total number of files listed, their cumulative size, and the free space (in bytes) remaining on the disk.

### Syntax

`dir [drive:][path][filename] [/p] [/w] [/a[:][attributes]] [/o[:][sortorder]] [/s] [/b] [/l] [/c]`

### Parameters

**[ drive :][ path ]**

Specifies the drive and directory for which you want to see a listing.

**[ filename ]**

Specifies a particular file or group of files for which you want to see a listing.

### Switches

**/p**

Displays one screen of the listing at a time. To see the next screen, press any key.

**/w**

Displays the listing in wide format, with as many as five filenames or directory names on each line.

**/a[:][ attributes ]**

Displays only the names of those directories and files with the attributes you specify. If you omit this switch, **dir** displays the names of all files except hidden and system files. If you use this switch without specifying **attributes**, **dir** displays the names of all files, including hidden and system files. The following list describes each of the values you can use for **attributes**. The colon (:) is optional. Use any combination of these values, and do not separate the values with spaces.

|                 |                               |
|-----------------|-------------------------------|
| <code>h</code>  | Hidden files                  |
| <code>-h</code> | Files that are not hidden     |
| <code>s</code>  | System files                  |
| <code>-s</code> | Files other than system files |

|    |                                                   |
|----|---------------------------------------------------|
| d  | Directories                                       |
| -d | Files only (not directories)                      |
| a  | Files ready for archiving (backup)                |
| -a | Files that have not changed since the last backup |
| r  | Read-only files                                   |
| -r | Files that are not read-only                      |

***/o[: sortorder]***

Controls the order in which **dir** sorts and displays directory names and filenames. If you omit this switch, **dir** displays the names in the order in which they occur in the directory. If you use this switch without specifying *sortorder*, **dir** displays the names of the directories, sorted in alphabetic order, and then displays the names of files, sorted in alphabetic order. The colon (:) is optional. The following list describes each of the values you can use for *sortorder*. Use any combination of the values, and do not separate these values with spaces.

| In alphabetic order by name |                                                        |
|-----------------------------|--------------------------------------------------------|
| -n                          | In reverse alphabetic order by name (Z through A)      |
| e                           | In alphabetic order by extension                       |
| -e                          | In reverse alphabetic order by extension (Z through A) |
| d                           | By date and time, earliest first                       |
| -d                          | By date and time, latest first                         |

|    |                                       |
|----|---------------------------------------|
| s  | By size, smallest first               |
| -s | By size, largest first                |
| g  | With directories grouped before files |
| -g | With directories grouped after files  |
| c  | By compression ratio, lowest first.   |
| -c | By compression ratio, highest first.  |

#### **/s**

Lists every occurrence, in the specified directory and all subdirectories, of the specified filename.

#### **/b**

Lists each directory name or filename, one per line (including the filename extension). This switch displays no heading information and no summary. The **/b** switch overrides the **/w** switch.

#### **/l**

Displays unsorted directory names and filenames in lowercase. This switch does not convert extended characters to lowercase.

#### **/c[h]**

Displays the compression ratio of files compressed using Doublespace, based on an 8K cluster size. The optional **h** switch displays the compression ratio of files compressed using Doublespace, based on the cluster size of the host drive. The **/c[h]** switch is ignored when used with the **/w** or **/b** switch.

#### **Notes**

##### **Using wildcards with dir**

You can use wildcards (\*) and (?) to display a listing of a subset of files and subdirectories. For an example illustrating the use of a wildcard, see the "Examples" section.

##### **Specifying file display attributes**

If you specify the **/a** switch with more than one value in attributes, **dir** displays the names of only those files with all the specified attributes. For example, if you specify the **/a** switch with the **r** and **-h** values for attributes by using either **/a:r-h** or **/ar-h**, **dir** displays only the names of Read-Only files that are not hidden.

##### **Specifying filename sorting**

If you specify more than one **sortorder** value, **dir** sorts the filenames by the first criterion first, then by the second criterion, and so on. For example, if you specify the **/o** switch with the **e** and **-s** values for **sortorder** by using either **/o:e-s** or **/oe-s**, **dir** sorts the names of directories and files by extension, with the largest first, and displays the final result. The alphabetic sorting by extension causes filenames with no extensions to appear first, then directory names, then filenames with extensions.

##### **Setting date and time formats**

The date and time formats used by **dir** depend on the country setting you use in your CONFIG.SYS file. If you don't use the **country** command, the formats are those for the United States.

##### **Using redirection symbols and pipes**

When you use a redirection symbol (>) to send **dir** output to a file or a pipe (|) to send **dir** output to another command, use the **/a:-d** and **/bs** switches to list only the filenames. You can use the **filename** parameter with the **/b** and **/s** switches to specify that **dir** is to search the current directory and its subdirectories for all filenames that match **filename**. **Dir** lists only the drive letter, directory name, filename, and filename extension, one path per line, for each filename it finds.

Before using a pipe for redirection, you should set the TEMP environment variable in your AUTOEXEC.BAT file. Otherwise, the temporary file will appear in the directory listing.

### **Presetting dir parameters and switches**

You can preset **dir** parameters and switches by including the **set** command with the DIRCMD environment variable in your AUTOEXEC.BAT file. You can use any valid combination of **dir** parameters and switches with the **set DIRCMD** command, including the location and name of a file.

For example, to use the DIRCMD environment variable to set the wide display format (**/w**) as the default format, include the following command in your AUTOEXEC.BAT file:

```
set dircmd=/w
```

For a single use of the **dir** command, you can override a switch set by using the DIRCMD environment variable. To do so, you use the same switch on the **dir** command line, but you must also precede the switch letter with a minus sign, as the following example shows:

```
dir /-w
```

You can change the DIRCMD default settings by typing the **set** command at the command prompt with a new parameter or switch after the equal sign (=). The new default settings are effective for all subsequent **dir** commands until you use **set DIRCMD** again on the command line or until you restart MS-DOS.

To clear all default settings, type the following command:

```
set dircmd=
```

You can view the current settings of the DIRCMD environment variable by typing the following command:

```
set
```

MS-DOS displays a list of environment variables and their settings. For more information about setting environment variables, see the **set** command.

### **Examples**

Suppose you want to display all files and directories in a directory, including hidden or system files. To specify this display, type the following command:

```
dir /a
```

Suppose you want **dir** to display one directory listing after another, until it has displayed the listing for every directory on the disk in the current drive. Suppose also that you want **dir** to alphabetize each directory listing, display it in wide format, and pause after each screen. To specify such a display, be sure the root directory is the current directory and then type the following command:

```
dir /s/w/o/p/a:-d
```

**Dir** lists the name of the root directory, the names of the subdirectories of the root directory, and the names of the files in the root directory (including extensions). Then **dir** lists the subdirectory names and filenames in each subdirectory in the directory tree.

To alter the preceding example so that **dir** displays the filenames and extensions but omits the directory names, type the following command:

```
dir /s/w/o/p/a:-d
```

To print a directory listing, type the redirection symbol and **prn** after any form of the **dir** command, as the following example shows:

```
dir > prn
```

When you specify **prn** on the **dir** command line, the directory listing is sent to the printer attached to the LPT1 port. If your printer is attached to a different port, you must replace **prn** with the name of the correct port.

You can also redirect output of the **dir** command to a file by replacing **prn** with a filename. A path is also accepted on the command line. For example, to direct **dir** output to the file DIR.DOC in the RECORDS directory, type the following command:

```
dir > \records\dir.doc
```

If DIR.DOC does not exist, MS-DOS creates it, unless the directory RECORDS also does not exist. In that case, MS-DOS displays the following message:

File creation error

To display a list of all the filenames with the .TXT extension in all directories on drive C, type the following command:

```
dir c:\*.txt /w/o/s/p
```

**Dir** displays, in wide format, an alphabetized list of the matching filenames in each directory and pauses each time the screen fills, until you press a key to continue.

### **Related Commands**

For information about displaying the directory structure of a path or disk, see the **tree** command.

For information about compressing disks, see the **dblspace** command.

### [Top Of Page](#)

#### **Diskcomp**

Compares the contents of two floppy disks.

This command performs a track-by-track comparison. **Diskcomp** determines the number of sides and sectors per track to compare based on the format of the first disk you specify.

#### **Syntax**

**diskcomp** [*drive1*: [*drive2*:]] [/1] [/8]

**Parameters**

***drive1*** :

Specifies the drive containing one of the floppy disks.

***drive2*** :

Specifies the drive containing the other floppy disk.

**Switches**

**/1**

Compares only the first sides of the disks, even if the disks are double-sided and the drives can read double-sided disks.

**/8**

Compares only the first 8 sectors per track, even if the disks contain 9 or 15 sectors per track.

**Notes**

**Invalid drive for diskcomp**

The **diskcomp** command works only with floppy disks. You cannot use **diskcomp** with a hard disk. If you specify a hard disk drive for *drive1* or *drive2*, **diskcomp** displays the following error message:

Invalid drive specification

Specified drive does not exist

or is non-removable

**Diskcomp messages**

If all tracks on the two disks being compared are the same, **diskcomp** displays the following message:

Compare OK

If the tracks are not the same, **diskcomp** displays a message similar to the following:

Compare error on

side 1, track 2

When **diskcomp** completes the comparison, it displays the following message:

Compare another diskette (Y/N) ?

If you press Y, **diskcomp** prompts you to insert disks for the next comparison. If you press N, **diskcomp** stops the comparison.

**Diskcomp** ignores a disk's volume number when it makes the comparison.

**Omitting drive parameters**

If you omit the *drive2* parameter, **diskcomp** uses the current drive for *drive2*. If you omit both drive parameters, **diskcomp** uses the current drive for both. If the current drive is the same as *drive1*, **diskcomp** prompts you to swap disks as necessary.

**Using one drive for the comparison**

If you specify the same floppy disk drive for *drive1* and *drive2*, **diskcomp** does a comparison by using one drive and prompts you to insert the disks as necessary. You might have to swap the disks more than once, depending on the capacity of the disks and the amount of available memory.

**Comparing different types of disks**

**Diskcomp** cannot compare a single-sided disk with a double-sided disk or a high-density disk with a double-density disk. If the disk in *drive1* is not of the same type as the disk in *drive2*, **diskcomp** displays the following message:

Drive types or diskette types not compatible

**Using diskcomp with networks and redirected drives**

**Diskcomp** does not work on a network drive or on a drive created or affected by a **subst** command. If you attempt to use **diskcomp** with a network drive or a drive created by the **subst** command, **diskcomp** displays an error message.

**Comparing an original disk with a copy**

When you use **diskcomp** with a disk that you made with the **copy** command, **diskcomp** may display a message similar to the following:

Compare error on

side 0, track 0

This type of error can occur even if the files on the disks are identical. Although the **copy** command duplicates information, it doesn't necessarily place it in the same location on the destination disk. For more information about comparing individual files on two disks, see the **fc** command.

**Diskcomp exit codes**

The following list shows each exit code and gives a brief description of its meaning:

|   |                         |
|---|-------------------------|
| 0 | The disks are the same. |
|---|-------------------------|

|   |                                              |
|---|----------------------------------------------|
| 1 | Differences were found.                      |
| 2 | The user pressed CTRL+C to stop the process. |
| 3 | A critical error occurred.                   |
| 4 | An initialization error occurred.            |

You can use the **errorlevel** parameter on the **if** command line in a batch program to process exit codes returned by **diskcomp**.

#### **Example**

If your system has only one floppy disk drive, drive A, and you want to compare two disks, type the following command:

```
diskcomp a: a:
```

**Diskcomp** prompts you to insert each disk, as required.

#### **Related Command**

For information about comparing two files, see the **fc** command.

#### [Top Of Page](#)

### **Diskcopy**

Copies the entire contents of one floppy disk to another floppy disk. **Diskcopy** writes over the existing contents of the destination disk as it copies the new information to it.

This command determines the number of sides to copy based on the source drive and disk.

#### **Syntax**

```
diskcopy [drive1: [drive2:]] [/1] [/v]
```

#### **Parameters**

##### **drive1 :**

Specifies the drive containing the source disk.

##### **drive2 :**

Specifies the drive containing the destination disk.

#### **Switches**

##### **/1**

Copies only the first side of a disk.

##### **/v**

Verifies that the information is copied correctly. Use of this switch slows the copying process.

#### **Notes**

##### **Invalid drive for diskcopy**

The **diskcopy** command works only with removable disks, such as floppy disks. You cannot use **diskcopy** with a hard disk. If you specify a hard disk drive for *drive1* or *drive2*, **diskcopy** displays the following error message:

```
Invalid drive specification
Specified drive does not exist
or is non-removable
```

#### **Diskcopy messages**

The **diskcopy** command prompts you to insert the source and destination disks and waits for you to press any key before continuing.

After copying, **diskcopy** displays the following message:

```
Copy another diskette (Y/N) ?
```

If you press Y, **diskcopy** prompts you to insert source and destination disks for the next copy operation. To stop the **diskcopy** process, press N.

If you are copying to an unformatted floppy disk in *drive2*, **diskcopy** formats the disk with the same number of sides and sectors per track as are on the disk in *drive1*. **Diskcopy** displays the following message while it formats the disk and copies the files:

```
Formatting while copying
```

If the capacity of the source disk is greater than that of the destination disk and your computer can detect this difference, **diskcopy** displays the following message:

Drive types or diskette types not compatible

#### Disk serial numbers

If the source disk has a volume serial number, **diskcopy** creates a new volume serial number for the destination disk and displays the number when the copy operation is complete.

#### Omitting drive parameters

If you omit the *drive2* parameter, **diskcopy** uses the current drive as the destination drive. If you omit both drive parameters, **diskcopy** uses the current drive for both. If the current drive is the same as *drive1*, **diskcopy** prompts you to swap disks as necessary.

#### Using one drive for copying

If *drive1* and *drive2* are the same, **diskcopy** prompts you whenever you should switch disks. If you omit both drive parameters and the current disk drive is a floppy disk drive, **diskcopy** prompts you each time you should insert a disk in the drive. If the disks contain more information than available memory can hold, **diskcopy** cannot read all of the information at once. **Diskcopy** reads from the source disk, writes to the destination disk, and prompts you to insert the source disk again. This process continues until the entire disk has been copied.

#### Avoiding disk fragmentation

Because **diskcopy** makes an exact copy of the source disk on the destination disk, any fragmentation on the source disk is transferred to the destination disk. Fragmentation is the presence of small areas of unused disk space between existing files on a disk.

A fragmented source disk can slow down the finding, reading, or writing of files. To avoid transferring fragmentation from one disk to another, use either the **copy** command or the **xcopy** command to copy your disk. Because **copy** and **xcopy** copy files sequentially, the new disk is not fragmented.

#### Copying Startup disks

If you use the **diskcopy** command to copy a startup disk, the copy will also be a startup disk. If you use **copy** or **xcopy** to copy a startup disk, the copy usually will not be a startup disk.

#### Diskcopy exit codes

The following list shows each exit code (**errorlevel** parameter) and gives a brief description of its meaning:

|   |                                              |
|---|----------------------------------------------|
| 0 | The copy operation was successful.           |
| 1 | A nonfatal read/write error occurred.        |
| 2 | The user pressed CTRL+C to stop the process. |
| 3 | A critical error occurred.                   |
| 4 | An initialization error occurred.            |

You can use the **errorlevel** parameter on the **if** command line in a batch program to process exit codes returned by **diskcopy**. For an example of a batch program that processes exit codes, see the **diskcomp** command.

#### Related Commands

For information about copying one or more files, see the **copy** command.

For information about copying directories and subdirectories, see the **xcopy** command.

For information about comparing two disks to see if they are identical, see the **diskcomp** command.

#### [Top Of Page](#)

#### DISPLAY.SYS

Enables you to display international character sets on EGA, VGA, and LCD monitors. This device driver must be loaded by a **device** or **devicehigh** command in your CONFIG.SYS file.

For an introduction to preparing your screen and keyboard for character sets, see the chapter "Customizing for International Use" in the *MS-DOS 6 User's Guide*.

**Syntax**

**device=[drive:][path]display.sys con[:]=(type[,hwcp][,n])**  
**device=[drive:][path]display.sys con[:]=(type[,hwcp][(n,m)])**

**Parameters**

**[ drive : ][ path ]**

Specifies the location of the DISPLAY.SYS file.

**type**

Specifies the display adapter in use. Valid values include EGA and LCD. The EGA value supports both EGA and VGA display adapters. If you omit the *type* parameter, DISPLAY.SYS checks the hardware to determine which display adapter is in use. You can also specify CGA and MONO as values for *type*, but they have no effect because character set switching is not enabled for these devices.

**hwcp**

Specifies the number of the character set that your hardware supports. The following list shows the character sets that MS-DOS supports and the country/region or language for each:

|     |                        |
|-----|------------------------|
| 437 | United States          |
| 850 | Multilingual (Latin I) |
| 852 | Slavic (Latin II)      |
| 860 | Portuguese             |
| 863 | Canadian-French        |
| 865 | Nordic                 |

For more information about character sets, see the appendix "Keyboard Layouts and Character Sets" in the *MS-DOS 6 User's Guide*.

**n**

Specifies the number of character sets the hardware can support in addition to the primary character set specified for the hwcp parameter. Valid values for *n* are in the range 0 through 6. This value depends on your hardware. For EGA display adapters, the maximum value for *n* is 6; for LCD display adapters, the maximum value for *n* is 1.

**m**

Specifies the number of subfonts the hardware supports for each code page. The default value is 2 if *type* is EGA, and 1 if *type* is LCD.

**Notes**

**Using DISPLAY.SYS with monochrome or CGA display adapters**

Because monochrome and CGA display adapters do not support character set switching, using DISPLAY.SYS with either type of adapter has no effect.

**Installing a third-party console driver**

If you install both DISPLAY.SYS and a third-party console driver, such as VT52.SYS, the third-party device driver must be installed first. Otherwise, the third-party device driver may disable DISPLAY.SYS.

**Example**

Suppose you want DISPLAY.SYS to support an EGA display adapter with the United States hardware character set (437) and the potential for two additional MS-DOS character sets. To do this and to specify that DISPLAY.SYS is in the DOS directory on drive C, add the following line to your CONFIG.SYS file:

```
device=c:\dos\display.sys con=(ega,437,2)
```

#### [Top Of Page](#)

##### Dos

Specifies that MS-DOS should maintain a link to the upper memory area, load part of itself into the high memory area (HMA), or both. You can use this command only in your CONFIG.SYS file.

##### Syntax

```
dos=high|low[,umb],noumb  
dos=[high,]low[,umb|noumb]
```

##### Parameters

**umb|noumb**

Specifies whether MS-DOS should manage upper memory blocks (UMBs) created by a UMB provider such as EMM386.EXE. The **umb** parameter specifies that MS-DOS should manage UMBs, if they exist. The **noumb** parameter specifies that MS-DOS should not manage UMBs. The default setting is **noumb**.

**high|low**

Specifies whether MS-DOS should attempt to load a part of itself into the HMA (**high**) or keep all of MS-DOS in conventional memory (**low**). The default setting is **low**.

##### Notes

###### **Must install HIMEM.SYS for dos=umb or dos=high**

You must install the HIMEM.SYS device driver or another extended memory manager before you specify either **dos=umb** or **dos=high**.

##### Using the umb parameter

You must specify the **dos=umb** command in order to load programs and device drivers into the upper memory area. Using the upper memory area frees more space in conventional memory for programs. In addition to using this command, you must install an upper-memory-block (UMB) provider. If your computer has an 80386 or 80486 processor, you can use EMM386.EXE for your UMB provider.

If you specify **dos=umb** and no UMB provider is installed, MS-DOS will not display an error message.

##### Using the high parameter

If you specify the **high** parameter, MS-DOS attempts to load part of itself into the HMA. Loading part of MS-DOS into the HMA frees conventional memory for programs. If you specify **dos=high** and MS-DOS is unable to use the HMA, the following message will appear:

HMA not available

Loading DOS low

##### Combining parameters

You can include more than one parameter on a single DOS command line, using commas to separate them.

For example, the following command lines are valid:

**dos=umb,low**

**dos=high,umb**

You can place the **dos** command anywhere in your CONFIG.SYS file.

##### Related Commands

For information about loading a device driver into the upper memory area, see the **devicehigh** command.

For information about loading a program into the upper memory area, see the **loadhigh** command.

#### [Top Of Page](#)

##### Doskey

Loads the Doskey program into memory. The Doskey program recalls MS-DOS commands and enables you to edit command lines and create and run macros.

Doskey is a memory-resident program. When installed, Doskey occupies about 3 kilobytes of resident memory.

##### Syntax

```
doskey [/reinstall] [/bufsize=size] [/macros] [/history][/insert]/overstrike] [macroname=[text]]
```

To start the Doskey program and use the default settings, use the following syntax:

**doskey**

##### Parameter

**macroname =[ text ]**

Creates a macro that carries out one or more MS-DOS commands (a Doskey macro). *Macroname* specifies the name you want to assign to the macro. *Text* specifies the commands you want to record.

##### Switches

**/reinstall**

Installs a new copy of the Doskey program, even if one is already installed. In the latter case, the **/reinstall** switch also clears the buffer.

**/bufsize=size**

Specifies the size of the buffer in which Doskey stores commands and Doskey macros. The default size is 512 bytes. The minimum buffer size is 256 bytes.

**/macros**

Displays a list of all Doskey macros. You can use a redirection symbol (>) with the **/macros** switch to redirect the list to a file. You can abbreviate the **/macros** switch as **/m**.

**/history**

Displays a list of all commands stored in memory. You can use a redirection symbol (>) with the **/history** switch to redirect the list to a file. You can abbreviate the **/history** switch as **/h**.

**/insert/overstrike**

Specifies whether new text you type is to replace old text. If you use the **/insert** switch, new text that you type on a line is inserted into old text (as if you had pressed the INSERT key). If you use the **/overstrike** switch, new text replaces old text. The default setting is **/overstrike**.

**Notes****Recalling a command**

To recall a command, you can use any of the following keys after loading Doskey into memory:

**UP ARROW**

Recalls the MS-DOS command you used before the one displayed.

**DOWN ARROW**

Recalls the MS-DOS command you used after the one displayed.

**PAGE UP**

Recalls the oldest MS-DOS command you used in the current session.

**PAGE DOWN**

Recalls the most recent MS-DOS command you used.

**Editing the command line**

With the Doskey program, you can edit the current command line. The following list describes the Doskey editing keys and their functions:

**LEFT ARROW**

Moves the cursor back one character.

**RIGHT ARROW**

Moves the cursor forward one character.

**CTRL+LEFT ARROW**

Moves the cursor back one word.

**CTRL+RIGHT ARROW**

Moves the cursor forward one word.

**HOME**

Moves the cursor to the beginning of the line.

**END**

Moves the cursor to the end of the line.

**ESC**

Clears the command from the display.

**F1**

Copies one character from the template to the MS-DOS command line. (The template is a memory buffer that holds the last command you typed.)

**F2**

Searches forward in the template for the next key you type after pressing F2. Doskey inserts the text from the template up to but not including the character you specify.

**F3**

Copies the remainder of the template to the command line. Doskey begins copying characters from the position in the template that corresponds to the position indicated by the cursor on the command line.

**F4**

Deletes characters, beginning with the current character position, up to a character you specify. To use this editing key, press F4 and type a character. Doskey deletes up to, but not including, that character.

**F5**

Copies the current command into the template and clears the command line.

**F6**

Places an end-of-file character (CTRL+Z) at the current position on the command line.

**F7**

Displays all commands stored in memory, with their associated numbers. Doskey assigns these numbers sequentially, beginning with 1 for the first (oldest) command stored in memory.

**ALT+F7**

Deletes all commands stored in memory.

**F8**

Searches memory for a command that you want Doskey to display. To use this editing key, type the first character, or the first few characters, of the command you want Doskey to search for and then press F8. Doskey displays the most recent command that begins with the text you typed. Press F8 repeatedly to cycle through all the commands that start with the characters you specified.

**F9**

Prompts you for a command number and displays the command associated with the number you specify. To display all the numbers and their associated commands, press F7.

**ALT+F10**

Deletes all macro definitions.

**Specifying a default insert mode**

If you press the INSERT key, you can type text on the Doskey command line in the middle of old text without replacing the old text. However, once you press ENTER, Doskey returns your keyboard to replace mode. You must press INSERT again to return to insert mode.

The **/insert** switch puts your keyboard in insert mode each time you press ENTER. Your keyboard effectively remains in insert mode until you use the **/overstrike** switch. You can temporarily return to replace mode by pressing the INSERT key; but once you press ENTER, Doskey returns your keyboard to insert mode.

The cursor changes shape when you use the INSERT key to change from one mode to the other.

**Creating a macro**

You can use the Doskey program to create macros that carry out one or more MS-DOS commands.

You can use the following special characters to control command operations when defining a macro:

**\$G or \$g**

Redirects output. Use either of these special characters to send output to a device or a file instead of to the screen. This character is equivalent to the redirection symbol for output (>).

**\$G\$G or \$g\$g**

Appends output to the end of a file. Use either of these special double characters to append output to an existing file rather than replace the data in the file. These double characters are equivalent to the "append" redirection symbol for output (>>).

**\$L or \$l**

Redirects input. Use either of these special characters to read input from a device or a file instead of from the keyboard. This character is equivalent to the redirection symbol for input (<).

**\$B or \$b**

Sends macro output to a command. Using one of these special characters is equivalent to using the pipe (|) on a command line.

**\$T or \$t**

Separates commands. Use either of these special characters to separate commands when you are creating macros or typing commands on the Doskey command line.

**\$\$**

Specifies the dollar-sign character (\$).

**\$1 through \$9**

Represents any command-line information you want to specify when you run the macro. The special characters **\$1** through **\$9** are batch parameters, which make it possible for you to use different data on the command line each time you run the macro. The **\$1** character in a **doskey** command is similar to the **%1** character in a batch program.

**\$\***

Represents *all* the command-line information you want to specify when you type the macro name. The special character **\$\*** is a replaceable parameter that is similar to the batch parameters **\$1** through **\$9**, with one important difference. Here, *everything* you type on the command line after the macro name is substituted for the **\$\*** in the macro.

For example, to create a macro that performs a quick and unconditional format of a disk, type the following command:

```
doskey qf=format $1 /q /u
```

For information about quick and unconditional formatting, see the **format** command.

You can use the **doskey** command in a batch program to create a macro.

**Running a macro**

To run a macro, type the macro name starting at the first position on the command line. If the macro was defined with **\$\*** or any of the batch parameters **\$1** through **\$9**, use a space to separate parameters.

You could run the **qf** macro created in the previous example to format a disk in drive A quickly and unconditionally. To do so, you would type the following command:

```
qf a:
```

You cannot run a macro from a batch program.

**Creating a macro with the same name as an MS-DOS command**

You might want to create a macro that has the same name as an MS-DOS command. This can be useful, for example, if you always use a certain command with specific switches. To specify whether you want to run the macro or the MS-DOS command, follow these guidelines:

- To run the macro, begin typing the macro name immediately after the command prompt, with no space between the prompt and the command name.
- To carry out the command, insert one or more spaces between the command prompt and the command name.

To delete a macro, type the following command:

```
doskey macroname=
```

### Examples

The **/macros** and **/history** switches are useful for creating batch programs to save macros and commands. For example, to create a batch program named MACINIT.BAT that includes all Doskey macros, type the following command:

```
doskey /macros > macinit.bat
```

To use the MACINIT.BAT file, edit it to include the **doskey** command at the beginning of each macro line.

To create a batch program named TMP.BAT that contains recently used commands, type the following command:

```
doskey /history > tmp.bat
```

To define a macro with multiple commands, use **\$t** to separate commands, as follows:

```
doskey tx=cd\temp$tdir/w $*
```

In the preceding example, the **tx** macro changes the current directory to TEMP and then displays a directory listing, using the wide display format. You can use **\$\*** at the end of the macro to append other switches to the **dir** command when you run **tx**.

The following macro uses a batch parameter for a new directory name. The macro first creates a new directory and then changes to it from the current directory.

```
doskey mc=md $1$tcd $1
```

To use the preceding macro to create and change to a directory named BOOKS, type the following:

```
mc books
```

To create a macro that uses batch parameters for moving a file or group of files, type the following command:

```
doskey mv=copy $1 $2 $t del $1
```

To create a macro that causes the **mem** command to pause after each screen, type the following command:

```
doskey mem=mem $* /p
```

### [Top Of Page](#)

#### Dosshell

Starts MS-DOS Shell, a graphical interface to MS-DOS.

#### Syntax

To start MS-DOS Shell in text mode, use the following syntax:

**dosshell [/t[:res[n]]] [/b]**

To start MS-DOS Shell in graphics mode, use the following syntax:

**dosshell [/g[:res[n]]] [/b]**

#### Parameters

##### **res**

Specifies a screen-resolution category. Valid values are **I**, **M**, and **H** to specify low, medium, and high resolution, respectively. The default value of **res** depends on your hardware.

##### **n**

Specifies a screen resolution when there is more than one choice within a category. For information about the valid values for this parameter, see the "Notes" section. The default value of **n** depends on your hardware.

#### Switches

##### **/t**

Starts MS-DOS Shell in text mode.

##### **/b**

Starts MS-DOS Shell using a black-and-white color scheme.

##### **/g**

Starts MS-DOS Shell in graphics mode.

#### Notes

##### **Running MS-DOS Shell with Microsoft Windows**

Do not start Microsoft Windows from within MS-DOS Shell. If you want to use both Microsoft Windows and MS-DOS Shell, start Windows, and then start MS-DOS Shell from within Windows.

#### **Memory requirement**

To run MS-DOS Shell, you should ensure that your computer has at least 384K of available conventional memory.

#### **Adjusting screen resolution**

Once you have started MS-DOS Shell, you can adjust the screen resolution by using the Display command on the Options menu. A dialog box displays the mode (text or graphics), the number of lines, the resolution category, and the specific number within each category for all possible screen-resolution modes available for your hardware.

#### **The DOSSHELL.INI file**

Your current MS-DOS Shell settings for program items and groups, options, screen resolution, colors, and so on, are stored in a file called DOSSHELL.INI. The DOSSHELL.INI file will be updated whenever you make a change or start a program item, so it must be located on a drive that is not write-protected. You can specify the location of the DOSSHELL.INI file by setting a **dosshell** environment variable in your AUTOEXEC.BAT file.

For example, if the DOSSHELL.INI file is located in the DOS directory on drive C, type the following command in your AUTOEXEC.BAT file:

```
dosshell=c:\dos
```

If you customize MS-DOS Shell to suit your own needs, you should back up this file regularly. If the DOSSHELL.INI file is deleted or corrupted, a new file will be created the next time you start MS-DOS Shell.

#### **Setting the location to store temporary files**

When you run a program from MS-DOS Shell, temporary files are created in the directory where the DOSSHELL.EXE file is located. You can specify that temporary files should be placed elsewhere by setting the TEMP environment variable in your AUTOEXEC.BAT file.

#### **Example**

To start MS-DOS Shell in graphics mode, type the following command:

```
dosshell /g
```

#### [Top Of Page](#)

#### **DRIVER.SYS**

Creates a logical drive that you can use to refer to a physical floppy disk drive. This device driver must be loaded by a **device** or **devicehigh** command in your CONFIG.SYS file.

A logical drive is a pointer to a physical disk drive in your system. The logical drive is associated with a drive letter (for example, A or B). You can specify parameters to describe the disk drive to MS-DOS.

#### **Syntax**

```
device=[drive:][path]driver.sys /d:number [/c] [/f:factor] [/h:heads] [/s:sectors] [/t:tracks]
```

#### **Parameter**

##### **[ drive :][ path ]**

Specifies the location of the DRIVER.SYS file.

#### **Switches**

##### **/d: number**

Specifies the number of the physical floppy disk drive. Valid values for *number* are in the range 0 through 127. The first physical floppy disk drive (drive A) is drive 0; a second physical floppy disk drive is drive 1; a third physical floppy disk drive, which must be external, is drive 2. For a computer with one floppy disk drive, drives A and B are both numbered 0; for a computer with multiple floppy disk drives, drive B is numbered 1.

##### **/c**

Specifies that the physical disk drive can detect whether the drive door is closed (change-line support).

##### **/f: factor**

Specifies the type of disk drive. Valid values for *factor* are as follows:

|  |                               |
|--|-------------------------------|
|  | 160K/180K or 320K/360K        |
|  | 1.2 megabyte (MB)             |
|  | 720K (3.5-inch disk) or other |

|  |                         |
|--|-------------------------|
|  | 1.44 MB (3.5-inch disk) |
|  | 2.88 MB (3.5-inch disk) |

The default value for factor is 2.

Generally, if you use the **/f** switch, you can omit the **/h**, **/s**, and **/t** switches. Check the default values for these switches to make sure they are correct for the type of disk drive you are using. To determine the appropriate values for the disk drive, see the disk-drive manufacturer's documentation.

If you specify the **/h**, **/s**, and **/t** switches, you can omit the **/f** switch.

#### **/h: heads**

Specifies the number of heads in the disk drive. Valid values for *heads* are in the range 1 through 99. The default value is 2. To determine the correct value for your disk drive, see the disk-drive manufacturer's documentation.

#### **/s: sectors**

Specifies the number of sectors per track. Valid values for *sectors* are in the range 1 through 99. The default value depends on the value of **/f:factor**, as follows:

**/f:0 /s:9 /f:1 /s:15 /f:2 /s:9 /f:7 /s:18 /f:9 /s:36**

To determine the correct value for your disk drive, see the disk-drive manufacturer's documentation.

#### **/t: tracks**

Specifies the number of tracks per side on the block device. Valid values for *tracks* are in the range 1 through 999. The default value is 80, unless **/f:factor** is 0, in which case the default value is 40. To determine the correct value for your disk drive, see the disk-drive manufacturer's documentation.

#### **Notes**

##### **Disk-drive change-line support**

The term "change-line support" means that a physical disk drive can detect when the drive door is opened and closed. Change-line support allows faster MS-DOS operation with floppy disks. The **/c** switch indicates to MS-DOS that the physical disk drive can support change-line error detection. To determine whether your disk drive has change-line support, see the disk-drive manufacturer's documentation.

##### **Modifying or redefining a supported physical disk drive**

For information about modifying the parameters of a physical disk drive that is supported by your hardware, see the **drivparm** command. You can also use DRIVER.SYS to redefine a physical floppy disk drive.

##### **Limitations on DRIVER.SYS**

You cannot use DRIVER.SYS with hard disk drives. For information about substituting a logical drive letter for a hard disk drive, see the **subst** command.

##### **Creating a duplicate logical drive**

Suppose you want to use one physical floppy disk drive to copy files from one floppy disk to another. Because you cannot copy from and to the same logical drive by using the **copy** or **xcopy** command, you must assign a second drive letter to that physical drive.

If your system has just one physical floppy disk drive, you do not need to install DRIVER.SYS for this purpose. MS-DOS already assigns both logical drive A and logical drive B to that drive. Just copy files from drive A to drive B and switch disks when MS-DOS prompts you.

If your system has more than one floppy disk drive, then you need to use DRIVER.SYS to assign a second drive letter to the physical floppy disk drive.

##### **Creating a new logical drive with different parameters**

If you use DRIVER.SYS to assign a logical drive that has parameters different from those of the previously assigned logical drive, then the parameters of the previous logical drive will be invalid. Therefore, you should no longer use the drive letter corresponding to the previous logical drive.

#### **Examples**

To add an external 720K drive to your system, add the following line to your CONFIG.SYS file:

```
device=driver.sys /d:2
```

Since no location is specified, MS-DOS searches for DRIVER.SYS in the root directory of your startup drive.

Suppose you want to use a single 1.44-megabyte external disk drive to copy files from one floppy disk to another. To do this, you must add two identical **device** commands for DRIVER.SYS in your CONFIG.SYS file. This procedure assigns two logical drive letters to the same physical drive. You can then swap disks in the same drive during the copying process. The following example shows how to do this:

```
device=driver.sys /d:2 /f:7
```

```
device=driver.sys /d:2 /f:7
```

[Top Of Page](#)

## Drivparm

Defines parameters for devices such as disk and tape drives when you start MS-DOS. You can use this command only in your CONFIG.SYS file.

The **drivparm** command modifies the parameters of an existing physical drive. It does not create a new logical drive. The settings specified in the **drivparm** command override the driver definitions for any previous block device.

### Syntax

**drivparm=/d:*number* [/c] [/f:*factor*] [/h:*heads*] [/i] [/n] [/s:*sectors*] [/t:*tracks*]**

### Switches

#### **/d: *number***

Specifies the physical drive number. Values for *number* must be in the range 0 through 255 (for example, drive number 0 = drive A, 1 = drive B, 2 = drive C, and so on).

#### **/c**

Specifies that the drive can detect whether the drive door is closed.

#### **/f: *factor***

Specifies the drive type. The following list shows the valid values for *factor* and a brief description of each. The default value is 2.

|  |                           |
|--|---------------------------|
|  | 160K/180K or<br>320K/360K |
|  | 1.2 megabyte (MB)         |
|  | 720K (3.5-inch disk)      |
|  | Hard disk                 |
|  | Tape                      |
|  | 1.44 MB (3.5-inch disk)   |
|  | Read/write optical disk   |
|  | 2.88 MB (3.5-inch disk)   |

#### **/h: *heads***

Specifies the maximum number of heads. Values for *heads* must be in the range 1 through 99. The default value depends upon the value you specify for **/f:factor**.

#### **/i**

Specifies an electronically compatible 3.5-inch floppy disk drive. (Electronically compatible drives are installed on your computer and use your existing floppy-disk-drive controller.) Use the **/i** switch if your computer's ROM BIOS does not support 3.5-inch floppy disk drives.

#### **/n**

Specifies a non-removable block device.

#### **/s: *sectors***

Specifies the number of sectors per track that the block device supports. Values for *sectors* must be in the range 1 through 99. The default value depends upon the value you specify for *If:factor*.

***l:* tracks**

Specifies the number of tracks per side that the block device supports. The default value depends upon the value you specify for *If:factor*.

**Notes**

**Using the /i switch**

Use the **/i** switch if your system does not support 3.5-inch floppy disk drives. (Some IBM PC/AT-compatible systems do not have a ROM BIOS that supports 3.5-inch floppy disk drives.)

**Disk drive change-line support**

*Change-line support* means that a physical disk drive can detect whether the drive door is opened and closed. Change-line support improves performance by letting MS-DOS know when one floppy disk has been replaced by another. The **/c** switch allows MS-DOS to make use of change-line support. To find out whether your disk drive has change-line support, see your disk-drive documentation.

**Creating a logical drive**

**Drivparm** modifies the parameters of an existing physical drive and does not create a new logical drive.

**Example**

Suppose your system has an internal tape drive with one head on drive D that is configured at startup to write 20 tracks of 40 sectors per track. To reconfigure this tape drive to write 10 tracks of 99 sectors each, add the following command to your CONFIG.SYS file:

```
drivparm=/d:3 /f:6 /h:1 /s:99 /t:10
```

[Top Of Page](#)

**Echo**

Displays or hides the text in batch programs when the program is running. Also indicates whether the command-echoing feature is on or off.

When you run a batch program, MS-DOS typically displays (echoes) the batch program's commands on the screen. You can turn this feature on or off by using the **echo** command.

**Syntax**

**echo [on|off]**

To use the **echo** command to display a message, use the following syntax:

**echo [message]**

**Parameters**

**on|off**

Specifies whether to turn the command-echoing feature on or off. To display the current **echo** setting, use the **echo** command without a parameter.

**message**

Specifies text you want MS-DOS to display on the screen.

**Notes**

**Using a message with the echo command**

The **echo message** command is useful when **echo** is off. To display a message that is several lines long without displaying other commands, you can include several **echo message** commands after the **echo off** command in your batch program.

**Hiding the command prompt**

If you use the **echo off** command on the command line, the command prompt does not appear on your screen. To redisplay the command prompt, type **echo on**.

**Preventing MS-DOS from echoing a line**

You can insert an at sign (@) in front of a command in a batch program to prevent MS-DOS from echoing that line.

**Echoing a blank line**

To echo a blank line on the screen, you can type **echo** and then a period (**echo.**). There must be no intervening space.

**Displaying pipes and redirection characters**

You cannot display a pipe (|) or redirection character (< or >) by using the **echo** command.

**Examples**

The following example shows a batch program that includes a three-line message preceded and followed by a blank line:

```
echo off
echo.
echo This batch program
echo formats and checks
echo new disks
echo.
```

If you want to turn **echo** off and you do not want to echo the **echo** command itself, include an at sign (@) before the command, as follows:

```
@echo off
```

You can use the **if** and **echo** commands on the same command line, as follows:

```
if exist *.rpt echo The report has arrived.
```

#### Related Command

For information about suspending the execution of a batch program, see the **pause** command.

[Top Of Page](#)

#### Edit

Starts MS-DOS Editor, a text editor you can use to create and edit ASCII text files.

MS-DOS Editor is a full-screen editor that allows you to create, edit, save, and print ASCII text files. Using MS-DOS Editor, you can choose commands from menus and specify information and preferences in dialog boxes. MS-DOS Editor includes extensive online Help about MS-DOS Editor techniques and commands.

#### Syntax

```
edit [[drive:]][path]filename [/b] [/g] [/h] [/nohi]
```

#### Parameter

[ **drive :]**[ **path** ] **filename**

Specifies the location and name of an ASCII text file. If the file does not exist, MS-DOS Editor creates it. If the file exists, MS-DOS Editor opens it and displays its contents on the screen.

#### Switches

**/b**

Displays MS-DOS Editor in black and white. Use this option if MS-DOS Editor isn't displayed correctly on a monochrome monitor.

**/g**

Uses the fastest screen updating for a CGA monitor.

**/h**

Displays the maximum number of lines possible for the monitor you are using.

**/nohi**

Enables you to use 8-color monitors with MS-DOS Editor. Usually, MS-DOS uses 16 colors.

Crash course in x86: [https://sensepost.com/blogstatic/2014/01/SensePost\\_crash\\_course\\_in\\_x86\\_assembly-pdf](https://sensepost.com/blogstatic/2014/01/SensePost_crash_course_in_x86_assembly-pdf)

## x86 Architecture[\[edit\]](#)

The x86 architecture has 8 General-Purpose Registers (GPR), 6 Segment Registers, 1 Flags Register and an Instruction Pointer. 64-bit x86 has additional registers.

### General-Purpose Registers (GPR) - 16-bit naming conventions[\[edit\]](#)

The 8 GPRs are:

1. Accumulator register (AX). Used in arithmetic operations
2. Counter register (CX). Used in shift/rotate instructions and loops.
3. Data register (DX). Used in arithmetic operations and I/O operations.
4. Base register (BX). Used as a pointer to data (located in segment register DS, when in segmented mode).
5. Stack Pointer register (SP). Pointer to the top of the stack.
6. Stack Base Pointer register (BP). Used to point to the base of the stack.
7. Source Index register (SI). Used as a pointer to a source in stream operations.
8. Destination Index register (DI). Used as a pointer to a destination in stream operations.

The order in which they are listed here is for a reason: it is the same order that is used in a push-to-stack operation, which will be covered later.

All registers can be accessed in 16-bit and 32-bit modes. In 16-bit mode, the register is identified by its two-letter abbreviation from the list above. In 32-bit mode, this two-letter abbreviation is prefixed with an 'E' (*extended*). For example, 'EAX' is the accumulator register as a 32-bit value.

Similarly, in the 64-bit version, the 'E' is replaced with an 'R', so the 64-bit version of 'EAX' is called 'RAX'.

It is also possible to address the first four registers (AX, CX, DX and BX) in their size of 16-bit as two 8-bit halves. The least significant byte (LSB), or low half, is identified by replacing the 'X' with an 'L'. The most significant byte (MSB), or high half, uses an 'H' instead. For example, CL is the LSB of the counter register, whereas CH is its MSB.

In total, this gives us five ways to access the accumulator, counter, data and base registers: 64-bit, 32-bit, 16-bit, 8-bit LSB, and 8-bit MSB. The other four are accessed in only three ways: 64-bit, 32-bit and 16-bit. The following table summarises this:

## Segment Registers[edit]

Vikidia

has

related

information

at

[Wikiversity](#)

[Memory](#)

[Segmentation](#)

- Stack Segment (SS). Pointer to the stack.
- Code Segment (CS). Pointer to the code.
- Data Segment (DS). Pointer to the data.
- Extra Segment (ES). Pointer to extra data ('E' stands for 'Extra').
- F Segment (FS). Pointer to more extra data ('F' comes after 'E').
- G Segment (GS). Pointer to still more extra data ('G' comes after 'F').

Most applications on most modern operating systems (like FreeBSD, Linux or Microsoft Windows) use a memory model that points nearly all segment registers to the same place (and uses paging instead), effectively disabling their use. Typically the use of FS or GS is an exception to this rule, instead being used to point at thread-specific data.

#### **EFLAGS Register**[\[edit\]](#)

The EFLAGS is a 32-bit register used as a collection of bits representing Boolean values to store the results of operations and the state of the processor.

The names of these bits are:

|    |    |    |    |    |    |    |    |    |    |    |     |     |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|-----|-----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20  | 19  | 18 | 17 | 16 |
| 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | ID | VIP | VIF | AC | VM | RF |

|    |    |             |    |    |    |    |    |    |   |    |   |    |   |    |   |
|----|----|-------------|----|----|----|----|----|----|---|----|---|----|---|----|---|
| 15 | 14 | 13          | 12 | 11 | 10 | 9  | 8  | 7  | 6 | 5  | 4 | 3  | 2 | 1  | 0 |
| 0  | NT | <b>IOPL</b> | OF | DF | IF | TF | SF | ZF | 0 | AF | 0 | PF | 1 | CF |   |

The bits named 0 and 1 are reserved bits and shouldn't be modified.

**The different use of these flags are:**

0. CF : Carry Flag. Set if the last arithmetic operation carried (addition) or borrowed (subtraction) a bit beyond the size of the register. This is then checked when the operation is followed with an add-with-carry or subtract-with-borrow to deal with values too large for just one register to contain.
2. PF : Parity Flag. Set if the number of set bits in the least significant byte is a multiple of 2.

4. AF : Adjust Flag. Carry of Binary Code Decimal (BCD) numbers arithmetic operations.
6. ZF : Zero Flag. Set if the result of an operation is Zero (0).
7. SF : Sign Flag. Set if the result of an operation is negative.
8. TF : Trap Flag. Set if step by step debugging.
9. IF : Interruption Flag. Set if interrupts are enabled.
10. DF : Direction Flag. Stream direction. If set, string operations will decrement their pointer rather than incrementing it, reading memory backwards.
11. OF : Overflow Flag. Set if signed arithmetic operations result in a value too large for the register to contain.
- 12- IOPL : I/O Privilege Level field (2 bits). I/O Privilege Level of the current process.
14. NT : Nested Task flag. Controls chaining of interrupts. Set if the current process is linked to the next process.
16. RF : Resume Flag. Response to debug exceptions.
17. VM : Virtual-8086 Mode. Set if in 8086 compatibility mode.
18. AC : Alignment Check. Set if alignment checking of memory references is done.
19. VIF : Virtual Interrupt Flag. Virtual image of IF.
20. VIP : Virtual Interrupt Pending flag. Set if an interrupt is pending.

21. ID : Identification Flag. Support for CPUID instruction if can be set.

#### **Instruction Pointer**[\[edit\]](#)

The EIP register contains the address of the **next** instruction to be executed if no branching is done.

EIP can only be read through the stack after a **call** instruction.

#### **Memory**[\[edit\]](#)

The x86 architecture is **little-endian**, meaning that multi-byte values are written least significant byte first. (This refers only to the ordering of the bytes, not to the bits.)

So the 32 bit value  $B3B2B1B0_{16}$  on an x86 would be represented in memory as:

|    |    |    |    |
|----|----|----|----|
| B0 | B1 | B2 | B3 |
|----|----|----|----|

For example, the 32 bits double word  $0x1BA583D4$  (the **0x** denotes hexadecimal) would be written in memory as:

|    |    |    |    |
|----|----|----|----|
| D4 | 83 | A5 | 1B |
|----|----|----|----|

This will be seen as  $0xD4\ 0x83\ 0xA5\ 0x1B$  when doing a memory dump.

#### **Two's Complement Representation**[\[edit\]](#)

Two's complement is the standard way of representing negative integers in binary. The sign is changed by inverting all of the bits and adding one.

|                 |      |
|-----------------|------|
| <i>Start:</i>   | 0001 |
| <i>Invert:</i>  | 1110 |
| <i>Add One:</i> | 1111 |

0001 represents decimal 1

1111 represents decimal -1

#### **Addressing modes**[\[edit\]](#)

The addressing mode indicates the manner in which the operand is presented.

##### **Register Addressing**

(operand address R is in the address field)

**mov ax, bx ; moves contents of register bx into ax**

##### **Immediate**

(actual value is in the field)

**mov ax, 1 ; moves value of 1 into register ax**

or

```
mov ax, 010Ch ; moves value of 0x010C into register ax
```

#### Direct memory addressing

(operand address is in the address field)

```
.data  
my_var dw 0abcdh ; my_var = 0xabcd  
.code  
mov ax, [my_var] ; copy my_var content into ax (ax=0abcd)
```

#### Direct offset addressing

(uses arithmetics to modify address)

```
byte_table db 12,15,16,22,..... ; Table of bytes  
mov al,[byte_table+2]  
mov al,byte_table[2] ; same as the former
```

#### Register Indirect

(field points to a register that contains the operand address)

```
mov ax,[di]
```

The registers used for indirect addressing are BX, BP, SI, DI

#### General-purpose registers (64-bit naming conventions)[\[edit\]](#)

Main page: [X86 Assembly/16 32 and 64 Bits](#)

Main page: [X86 Assembly/SSE](#)

64-bit x86 adds 8 more general-purpose registers, named R8, R9, R10 and so on up to R15. It also introduces a new naming convention that must be used for these new registers and can also be used for the old ones (except that AH, CH, DH and BH have no equivalents). In the new convention:

- R0 is RAX.
- R1 is RCX.
- R2 is RDX.
- R3 is RBX.
- R4 is RSP.
- R5 is RBP.
- R6 is RSI.
- R7 is RDI.
- R8,R9,R10,R11,R12,R13,R14,R15 are the new registers and have no other names.
- R0D~R15D are the lowermost 32 bits of each register. For example, R0D is EAX.
- R0W~R15W are the lowermost 16 bits of each register. For example, R0W is AX.
- R0L~R15L are the lowermost 8 bits of each register. For example, R0L is AL.

As well, 64-bit x86 includes SSE2, so each 64-bit x86 CPU has at least 8 registers (named XMM0~XMM7) that are 128 bits wide, but only accessible through [SSE instructions](#). They cannot be used for quadruple-precision (128-bit) floating-point arithmetic, but they can each hold 2 double-precision or 4 single-precision floating-point values for a SIMD parallel instruction. They can also be operated on as 128-bit

integers or vectors of shorter integers. If the processor supports AVX, as newer Intel and AMD desktop CPUs do, then each of these registers is actually the lower half of a 256-bit register (named YMM0~YMM7), the whole of which can be accessed with AVX instructions for further parallelization.

### Stack[\[edit\]](#)

The stack is a Last In First Out (LIFO) data structure; data is pushed onto it and popped off of it in the reverse order.

```
mov ax, 006Ah  
mov bx, F79Ah  
mov cx, 1124h  
push ax
```

You push the value in AX onto the top of the stack, which now holds the value \$006A.

```
push bx
```

You do the same thing to the value in BX; the stack now has \$006A and \$F79A.

```
push cx
```

Now the stack has \$006A, \$F79A, and \$1124.

```
call do_stuff
```

Do some stuff. The function is not forced to save the registers it uses, hence us saving them.

```
pop cx
```

Pop the last element pushed onto the stack into CX, \$1124; the stack now has \$006A and \$F79A.

```
pop bx
```

Pop the last element pushed onto the stack into BX, \$F79A; the stack now has just \$006A.

```
pop ax
```

Pop the last element pushed onto the stack into AX, \$006A; the stack is empty.

The Stack is usually used to pass arguments to functions or procedures and also to keep track of control flow when the call instruction is used. The other common use of the Stack is temporarily saving registers.

### CPU Operation Modes[\[edit\]](#)

#### Real Mode[\[edit\]](#)

Real Mode is a holdover from the original Intel 8086. You generally won't need to know anything about it (unless you are programming for a DOS-based system or, more likely, writing a boot loader that is directly called by the BIOS).

The Intel 8086 accessed memory using 20-bit addresses. But, as the processor itself was 16-bit, Intel invented an addressing scheme that provided a way of mapping a 20-bit addressing space into 16-bit words. Today's x86 processors start in the so-called Real Mode, which is an operating mode that mimics the behavior of the 8086, with some very tiny differences, for backwards compatibility.

In Real Mode, a segment and an offset register are used together to yield a final memory address. The value in the segment register is multiplied by 16 (shifted 4 bits to the left) and the offset is added to the result. This provides a usable address space of 1 MB. However, a quirk in the addressing scheme allows access past the 1 MB limit if a segment address of 0xFFFF (the highest possible) is used; on the 8086 and 8088, all accesses to this area wrapped around to the low end of memory, but on the 80286 and later, up to 65520 bytes past the 1 MB mark can be addressed this way if the A20 address line is enabled. See: [The A20 Gate Saga](#).

One benefit shared by Real Mode segmentation and by [Protected Mode Multi-Segment Memory Model](#) is that all addresses must be given relative to another address (this is, the segment base address). A program can have its own address space and completely ignore the segment registers, and thus no pointers have to be relocated to run the program. Programs can perform *near* calls and jumps within the same segment, and data is always relative to segment base addresses (which in the Real Mode addressing scheme are computed from the values loaded in the Segment Registers).

This is what the DOS \*.COM format does; the contents of the file are loaded into memory and blindly run. However, due to the fact that Real Mode segments are always 64 KB long, COM files could not be larger than that (in fact, they had to fit into 65280 bytes, since DOS used the first 256 of a segment for housekeeping data); for many years this wasn't a problem.

#### [Protected Mode](#)[edit]

#### [Flat Memory Model](#)[edit]

If programming in a modern operating system (such as Linux, Windows), you are basically programming in flat 32-bit mode. Any register can be used in addressing, and it is generally more efficient to use a full 32-bit register instead of a 16-bit register part. Additionally, segment registers are generally unused in flat mode, and it is generally a bad idea to touch them.

#### [Multi-Segmented Memory Model](#)[edit]

Using a 32-bit register to address memory, the program can access (almost) all of the memory in a modern computer. For earlier processors (with only 16-bit registers) the segmented memory model was used. The 'CS', 'DS', and 'ES' registers are used to point to the different *chunks* of memory. For a small program (small model) the CS=DS=ES. For larger memory models, these 'segments' can point to different locations.

Home Work #1:

22. A binary bit stores a(n) or a(n) \_\_\_\_\_ –

A binary bit stores a 1 or a 0.

23. A computer K (pronounced kay) is equal to bytes.  
1024

24. A computer M (pronounced meg) is equal to K bytes.  
1024K

25. A computer G (pronounced gig) is equal to M bytes.  
1024

26. A computer P (pronounced peta) is equal to T bytes.  
1024

27. How many typewritten pages of information are stored in a 4G-byte memory?  
1,000,000

28. The first 1 M byte of memory in a DOS-based computer system contains a(n) \_\_\_\_\_ – and a(n) area.  
System area and transient program area

29. How large is the Windows application programming area?

2G or 3G for 32-bit mode and currently 8G for 64-bit mode

30. How much memory is found in the DOS transient program area?

640K

31. How much memory is found in the Windows systems area?

1G

32. The 8086 microprocessor addresses bytes of memory.

1M

33. The Core2 microprocessor addresses bytes of memory.

Currently 1T byte using a 40-bit address

34. Which microprocessors address 4G bytes of memory?

80386, 80486, Pentium, Pentium Pro, PH, PIII, P4, and Core2

35. Memory above the first 1M byte is called memory.

Protected memory or extended memory

36. What is the system BIOS?

The basic I/O system

37. What is DOS?

An early operating system called the Disk Operating System

Convert the following binary numbers into decimal:

(a) 1101.01

156.625

(b) 111001.0011

18.375

(c) 101011.0101

83.578125

(d) 111.0001

58.58

58.90625

Convert the following octal numbers into decimal:

(a) 234.5

(b) 12.3

(c) 7767.07

(d) 123.45

(e) 72.72

59.

Convert the following hexadecimal numbers into decimal:

(a) A3.3

(b) 129.C

(c) AC.DC

(d) FAB.3

(e) BB8.0D

60. Convert the following decimal integers into binary, octal, and hexadecimal:

(a) 23

(b) 107

(c) 1238

(d) 92

(e) 173

61. Convert the following decimal numbers into binary, octal, and hexadecimal:

(a) 0.625

(b) .00390625

(c) .62890625

(d) 0.75

(e) .9375

62. Convert the following hexadecimal numbers into binary-coded hexadecimal code (BCH):

(a) 23

(b) AD4

(c) 34.AD

(d) BD32

(e) 234.3

63. Convert the following binary-coded hexadecimal numbers into hexadecimal:

(a) 11000010

(b) 000100001111 110]

(e) IOII 1100

(d) 0001 0000

(e) 10001011 IOIO

64. Convert the following binary numbers to the one's complement form:

- (a) 1000 1000
- (b) OJOI 1010
- c) 01110111
- (d) 10000000

65. Convert the following binary numbers to the two's complement form:

- (a) 1000000 1
- (b) 10101100
- (c) 1010 1111
- (d) 1000 0000

Homework Solutions:

<http://haell.com/~wyrn/works/academia/CSUS/cpe185/>

PLC:

PLC Basics

The Most Unconventional Guide to PLCs on the Internet

Since the late 1960's, the Programmable Logic Controller (or PLC) has become an essential aspect of any automated manufacturing process. In recent times there have been contenders to replace the PLC but the principles and popularity of those early PLC's functions and concepts have continued unabated. It is essential for many people from the technician to the engineer to understand these essential pieces of equipment.

PLCDev has developed this instruction manual "PLC Basics" in hopes it will serve the beginner to the advanced user. Our goal is to make it comprehensive as possible, providing not only the generic principles in all PLCs, but to give practical examples from many different PLC manufacturers. To this end you are welcome to comment on each section by registering as a user and clicking the "add a comment" link at the bottom of the page.

"I'm writing a book. I've got the page numbers done. " --- Steven Wright

PLC Basics is a work in progress. If you wish to be notified of updates then please register for an account and make sure the email notification selection is checked.

Introduction to Programmable Controllers

It's always good to get an overview of where designs have been and were they are going. To do this it's essential to get a bird's eye view of the concepts and processes that make the PLC so valuable in industrial control. Pitting PLCs against other control types will also serve to show the pros and cons for different applications.

Definition of a PLC

What is a PLC?

A Programmable Logic Controller, or PLC for short, is simply a special computer device used for industrial control systems. They are used in many industries such as oil refineries, manufacturing lines, conveyor systems and so on. Where ever there is a need to control devices the PLC provides a flexible way to "softwire" the components together.

The basic units have a CPU (a computer processor) that is dedicated to run one program that monitors a series of different inputs and logically manipulates the outputs for the desired control. They are meant to be very flexible in how they can be programmed while also providing the advantages of high reliability (no program crashes or mechanical failures), compact and economical over traditional control systems.

A Simple Example

Consider something as simple as a switch that turns on a light. In this system with a flick of the switch the light would turn on or off. Beyond that though there is no more control. If your boss came along and said I want that light to turn on thirty seconds after the switch has been flipped, then you would need to buy a timer and do some rewiring. So it is time, labor and money for any little change.

00-01 switch to light

A PLC Saves the Day

Now consider the same device with a PLC in the middle. The switch is fed as an input into the PLC and the light is controlled by a PLC output. Implementing a delay in this system is easy since all that needs to be changed is the program in the PLC to use a delay timer.

00-02 switch to plc to light

This is a rather simple example but in a larger system with many switches and lights (and a host of other devices) all interacting with each other this kind of flexibility is not only nice but imperative. Hopefully a light bulb has now turned on over your head.

How PLCs Work

A programmable logic controller is a specialized computer used to control machines and processes. It therefore shares common terms with typical PCs like central processing unit, memory, software and communications. Unlike a personal computer though the PLC is designed to survive in a rugged industrial atmosphere and to be very flexible in how it interfaces with inputs and outputs to the real world.

The components that make a PLC work can be divided into three core areas.

The power supply and rack

The central processing unit (CPU)

The input/output (I/O) section

PLCs come in many shapes and sizes. They can be so small as to fit in your shirt pocket while more involved controls systems require large PLC racks. Smaller PLCs (a.k.a. "bricks") are typically designed with fixed I/O points. For our consideration, we'll look at the more modular rack based systems. It's called "modular" because the rack can accept many different types of I/O modules that simply slide into the rack and plug in.

PLC angled with doors closed and text

The Power Supply and Rack

So let's start off by removing all our modules which leaves us with a naked PLC with only the power supply and the rack.

PLC empty rack with text

The rack is the component that holds everything together. Depending on the needs of the control system it can be ordered in different sizes to hold more modules. Like a human spine the rack has a backplane at the rear which allows the cards to communicate with the CPU. The power supply plugs into the rack as well and supplies a regulated DC power to other modules that plug into the rack. The most popular power supplies work with 120 VAC or 24 VDC sources.

The CPU

The brain of the whole PLC is the CPU module. This module typically lives in the slot beside the power supply. Manufacturers offer different types of CPUs based on the complexity needed for the system.

The CPU consists of a microprocessor, memory chip and other integrated circuits to control logic, monitoring and communications. The CPU has different operating modes. In programming mode it accepts the downloaded logic from a PC. The CPU is then placed in run mode so that it can execute the program and operate the process.

## SLC500 CPU components

Since a PLC is a dedicated controller it will only process this one program over and over again. One cycle through the program is called a scan time and involves reading the inputs from the other modules, executing the logic based on these inputs and then updating the outputs accordingly. The scan time happens very quickly (in the range of 1/1000th of a second). The memory in the CPU stores the program while also holding the status of the I/O and providing a means to store values.

### CPU operating cycle

### I/O System

The I/O system provides the physical connection between the equipment and the PLC. Opening the doors on an I/O card reveals a terminal strip where the devices connect.

### PLC face with doors open

There are many different kinds of I/O cards which serve to condition the type of input or output so the CPU can use it for its logic. It's simply a matter of determining what inputs and outputs are needed, filling the rack with the appropriate cards and then addressing them correctly in the CPUs program.

### PLC inserting card

### Inputs

Input devices can consist of digital or analog devices. A digital input card handles discrete devices which give a signal that is either on or off such as a pushbutton, limit switch, sensors or selector switches. An analog input card converts a voltage or current (e.g. a signal that can be anywhere from 0 to 20mA) into a digitally equivalent number that can be understood by the CPU. Examples of analog devices are pressure transducers, flow meters and thermocouples for temperature readings

### Outputs

Output devices can also consist of digital or analog types. A digital output card either turns a device on or off such as lights, LEDs, small motors, and relays. An analog output card will convert a digital number sent by the CPU to its real world voltage or current. Typical outputs signals can range from 0-10 VDC or 4-20mA and are used to drive mass flow controllers, pressure regulators and position controls.

### Programming a PLC

In these modern times a PC with specially dedicated software from the PLC manufacturer is used to program a PLC.

The most widely used form of programming is called ladder logic. Ladder logic uses symbols, instead of words, to emulate the real world relay logic control, which is a relic from the PLC's history. These symbols are interconnected by lines to indicate the flow of current through relay like contacts and coils. Over the years the number of symbols has increased to provide a high level of functionality.

The completed program looks like a ladder but in actuality it represents an electrical circuit. The left and right rails indicate the positive and ground of a power supply. The rungs represent the wiring between the different components which in the case of a PLC are all in the virtual world of the CPU. So if you can understand how basic electrical circuits work then you can understand ladder logic.

In this simplest of examples a digital input (like a button connected to the first position on the card) when it is pressed turns on an output which energizes an indicator light.

### Simple ladder logic example

The completed program is downloaded from the PC to the PLC using a special cable that's connected to the front of the CPU. The CPU is then put into run mode so that it can start scanning the logic and controlling the outputs.

## The Birth of the PLC

### The Original Challenge

The early history of the PLC is fascinating. Imagine if you will a fifty foot long cabinet filled with relays whose function in life is to control a machine. Wires run in and out of the system as the relays click and clack to the logic. Now imagine there is a problem or a small design change and you have to figure it all out on paper and then shut down the machine, move some wires, add some relays, debug and do it all over again. Imagine the labor involved in the simplest of changes. This is the problem that faced the engineers at the Hydra-matic division of GM motors in the late 1960's.

Fortunately for them the prospect of computer control was rapidly becoming a reality for large corporations as themselves. So in 1968 the GM engineers developed a design criteria for a "standard machine controller". This early model simply had to replace relays but it also had to be:

A solid-state system that was flexible like a computer but priced competitively with a like kind relay logic system.  
Easily maintained and programmed in line with the all ready accepted relay ladder logic way of doing things.  
It had to work in an industrial environment with all it's dirt, moisture, electromagnetism and vibration.  
It had to be modular in form to allow for easy exchange of components and expandability.

### The Race is On

This was a tall order in 1968 but four companies took on the challenge.

Information Instruments, Inc. (fully owned by Allen-Bradley a year later).

Digital Equipment Corp. (DEC)

Century Detroit

Bedford Associates

Bedford Associates, run by Richard Morley, won the contract and quickly formed a new company around the technology called MODICON after Modular Digital Control. By June of 1969 they were selling the first viable Programmable Controller the "084" (their 84th project) which sold over one thousand units. These early experiences gave birth to their next model the "184" in 1973 which set Modicon as the early leader in programmable controllers.

Not to be outdone, the powerhouse Allen-Bradley (all ready known for it's rheostats, relays and motor controls) purchased Information Instruments in 1969 and began development on this new technology. The early models (PDQ-II and PMC) were deemed to be too large and complex. By 1971 Odo Struger and Ernst Dummermuth had begun to develop a new concept known as the Bulletin 1774 PLC which would make them successful for years to come. Allen-Bradley termed their new device the "Programmable Logic Controller" (patent #3,942,158) over the then accepted term "Programmable Controller". The PLC terminology became the industry standard especially when PC became associated with personal computers.

## Numbers, Codes and a Little Logic

At the heart of any computer system are the numbering systems and digital codes used for instructions and memory storage. Besides the ones and zeros it is important to understand how these bits are packaged into codes like BCD and Gray codes. Understanding these concepts enables the programmer to manipulate the PLC at its most basic level. Don't you want to feel the power at your fingertips!? Now for the most part these types of concepts will be transparent when programming but there will be occasions when you'll be glad you read this.

### Caveman Tech

When the cavemen first started counting they sat around looking at their fingers and stopped at ten. It took man a little longer to figure out the concept of zero but nobody is perfect. So we've ended up with a common way of counting by 10's which in tech jargon is referred to as base-10 or radix-10. Base or radix refers to the number of symbols you have available to count. We'll see that computers make it more difficult for us lazy humans because they like to count by base-2 (binary). In a compromise with computers us humans have developed the base-8 (octal) or base-16 (hexadecimal) systems.

### Binary People: Learning your 1's and 0's

#### Introduction

"There are 10 types of people in the world: Those who understand binary, and those who don't."

If you don't get this joke then hopefully by the end of the chapter you'll get a chuckle out of it. You see, in the binary number system  $10 = 2$ . How? Consider if you had no choice but to count with only a zero or a one. Pretend that the evil goblin of number snatchers had taken every number from two to nine. How would you count anything? You'd have to do it like 0, 1, 10, 11, 100, 101, 110, 111 and so on. You get the point? Therefore, if you had three apples then the third apple would be designated as 10 when in reality you only have three apples to eat.

When it comes to computers and therefore PLCs they can only store in memory a 0 or a 1. That's the beauty of our digital age, it's either "on" or it's "off". Those memory chips in computers are actually made of rows upon rows of circuits that are either on at some voltage or off at some voltage. Therefore a computer at its very basic level can only count using a 0 (off) or a 1 (on).

That's why it's called binary because there are only two numbers like there are only two wheels on a bicycle. The number system we are used to using is called decimal (dec = 10) and therefore we get to use ten numbers from 0 to 9. When you think about it, it's truly arbitrary how we count. You could also use an octal system (by 8) or hexadecimal (by 16) numbering system which we'll talk about a little later.

#### Base 10: An Old Familiar Friend

Let's start off by looking closer at our all too familiar base 10 decimal system and then compare it to binary. Decimal, like all these other number systems, is based on place-value system. This means that the value of a digit depends both on the digit itself and its position within the number. The following figure shows the weights of a decimal number broken down into columns.

#### Place-Value number systems

That much is obvious right? So with a number like 9876 we add it up like this:

#### 01-01 combining digits with column weights

The value of the number is computed by multiplying each digit by the weight of its position and adding up the results.

Now for Binary People: Learning your 1s and 0s.">Binary

Remember that for a base 10 system the weights are 1, 10, 100, 1000 and so on. For a binary system the weights are 1, 2, 4, 8, 16, 32, 64, etc.

#### 01-03 place value number systems for binary

To figure out the value of a binary system we do the same as with the decimal but with different weights.

#### 01-04 combining digits with column weights binary

The base of the number is usually expressed in a subscript so that in our examples  $11111_2 = 3110$  and  $10101_2 = 2110$ . It sure is a little cumbersome to count in binary but at least it's easier than Roman numerals Laughing. Word, Byte and Bit

What do you call a group of binary digits? The geeks who first thought this stuff up decided to call a binary digit a bit (b-inary dig-it). This is not to be confused with Tim-bits. After some time they decided that it would be good to call a group of 8 bits a byte. Funny bunch of geeks that they were the term nibble became used for 4 bits being a subset of byte. Finally, a group of 16 bits are referred to as a word. Here's a picture to drive home the point.

#### 01-05 word bytes bits

#### Cheating with a Calculator

Unless you're a glutton for punishment a standard scientific calculator switches between the different number formats with ease. In fact there's one built into the Windows operating system that typically lives under the Start button | All Programs | Accessories. Pull down the View menu and select "Scientific". I've highlighted the area below where you can switch between hexadecimal, decimal, octal and binary. Very handy even for doing math and logical functions (i.e. AND, OR, etc.).

#### Scientific Calculator

#### Octal? What the Hex?

"Do not worry about your problems with mathematics, I assure you mine are far greater." --- Albert Einstein

You're throbbing head is probably all ready telling you that binary numbers are not easy to read. As a compromise between humans and computers the octal (base 8) and hexadecimal (base 16) are used.

#### Octal

While octal is not as common as its cousin hexadecimal it is still used in various PLCs so it's important to grasp the concept. For instance, when programming an AutomationDirect PLC the memory addresses are in octal. Octal, like an octopus' eight legs, means eight and therefore there are eight numbers to use from zero to seven. The column weights are 1, 8, 64, 512, etc. The weights are derived by taking the base number to the power of the column,  $8^0=1$ ,  $8^1=8$ ,  $8^2=64$ ,  $8^3=512$ , etc. Now we can do the same exercise as in the last chapter to convert an octal number to decimal.

#### 01-06 combining digits with column weights octal

I know this isn't helpful so far. Where it really comes in handy is converting from binary to octal because all you have to do is break down the binary number into chunks of three. This is because 8 is 23.

## 01-07 binary to octal by threes

Most programmable controllers have inputs and output cards grouped in 8 or 16 (and high density of 32 and 64). The reason for this is the way computers like to have things in powers of 2, 4, 8, 16 and so on. So if it is not in octal it is typically in hexadecimal.

## Hexadecimal

Hexadecimal is a little more tricky because it is base 16 and therefore we need something beyond 0 through 9 for symbols and this is done by using the letters 'A' through 'F'. Hexadecimal is used for the same reasons as octal so that we can represent binary in a condensed form and make it easier for conversion. Where octal used 3 bits the hexadecimal system used 4 bits to represent one number.

## 01-08 binary to hex by fours

If you're going to be programming something like a Mitsubishi PLC then you better get used to hexadecimal.

## Conclusion

To wrap things up here's a table below showing the equivalents for each numbering system.

| Decimal | Binary | People: Learning your 1s and 0s.">Binary | Octal | Hex |
|---------|--------|------------------------------------------|-------|-----|
| 0       | 00000  | 0                                        | 0     |     |
| 1       | 00001  | 1                                        | 1     |     |
| 2       | 00010  | 2                                        | 2     |     |
| 3       | 00011  | 3                                        | 3     |     |
| 4       | 00100  | 4                                        | 4     |     |
| 5       | 00101  | 5                                        | 5     |     |
| 6       | 00110  | 6                                        | 6     |     |
| 7       | 00111  | 7                                        | 7     |     |
| 8       | 01000  | 10                                       | 8     |     |
| 9       | 01001  | 11                                       | 9     |     |
| 10      | 01010  | 12                                       | A     |     |
| 11      | 01011  | 13                                       | B     |     |
| 12      | 01100  | 14                                       | C     |     |
| 13      | 01101  | 15                                       | D     |     |
| 14      | 01110  | 16                                       | E     |     |
| 15      | 01111  | 17                                       | F     |     |
| 16      | 10000  | 20                                       | 10    |     |
| 17      | 10001  | 21                                       | 11    |     |
| 18      | 10010  | 22                                       | 12    |     |
| 19      | 10011  | 23                                       | 13    |     |

and so on and so on . . .

## When Being Negative is a Complement

Let's delve deeper into PLC programming by considering again our common word made of 16 bits. If it was all filled up with one's then the decimal value would be 65535. So a range from 0 to 65535 could be represented. Adding binary numbers together would be very similar to addition in decimal. For example  $0 + 1 = 1$  and  $1 + 1 = 10$  (carry the one).

The problem comes when you need to subtract. How do you represent a negative number when you can't just put a minus sign in front of it and say it's good? Remember that the computer can only do a 0 or a 1. To our rescue comes a concept called taking the complement. Complement's are a pretty cool trick and you can learn more about them at Wikipedia. We'll keep it simple here and talk about two's complement which is the most common in computers and

PLCs.

Signed binary numbers are achieved by stealing the 16th bit in a word (the most significant bit) and using that as a sign bit where 0 is positive and 1 is negative.

01-09 negative bit

By doing this we have shifted the range of values from 0 to 65535 to -32767 to +32767. So the high end of our value is decreased but we've made it a whole lot easier to indicate a negative number and do subtraction. Here's how it works. Let's take a number like 30 and perform the two's complement to get -30.

The number 30 is            0000 0000 0001 1110  
The first step is to invert (or flip) the bits       1111 1111 1110 0001  
The second step is to add one       1111 1111 1110 0010

In the PLC then the value of -30 will be represented as 1111 1111 1110 0010. Maybe not what you would expect? The beauty of this is that now all the processor has to worry about is adding the two numbers to get the correct value. See how this magically works in the table below (I'm just going to use 8 bits now to simplify things but it works just the same with however many bits you want).

Decimal Signed Binary People: Learning your 1s and 0s.">Binary  
57

|             |  |
|-------------|--|
| + 30        |  |
| = 87        |  |
| 0011 1001   |  |
| + 0001 1110 |  |
| 0101 0111   |  |
| 57          |  |
| + (-30)     |  |
| = 27        |  |
| 0011 1001   |  |
| + 1110 0010 |  |
| 0001 1011   |  |
| -57         |  |
| + 30        |  |
| = -27       |  |
| 1100 0111   |  |
| + 0001 1110 |  |

1110 0101

-57

+ (-30)

= -87

1100 0111

+ 1110 0010

1010 1001

For the most part this will all work seamlessly in the background while you program away. It's just that every once in awhile you'll need this knowledge to overcome any limitations in the system. Dealing with negative numbers can be pretty tricky but with complements like this you better not let it go to your head.

#### Getting the Point about Floating Point Numbers

##### The Freedom of Floating Point

Floating point numbers (also known as 'real numbers') give a certain freedom in being able to represent both very large and very small numbers in the confines of a 32 bit word (that's a double word in our PLCs). Up until this point the range of numbers we were able to represent with a double word would be from 0 to 4,294,967,295. Floating point on the other hand allows a representation of 0.0000000000000001 as well as +/-1,000,000,000,000. It allows for such large numbers that we can even keep track of the US national debt.

Floating point gives us an easy way to deal with fractions. Before, a word could only represent an integer, that is, a whole number. We'd have to use some tricks to maybe imply a decimal point. For instance, a number like 2300 in a word could be taken to represent 23.00 if the decimal point is "implied" to be in the 1/100th place. This might be all we need but it can get a bit tricky when it comes to math where we want to retain a remainder. The trick is to get some sort of format where the decimal point can "float" around the number.

##### Real Numbers in the Real World

At this point let's deal with an example. In this case we're using an Automation Direct DL250 PLC which conveniently has the ability to handle real numbers (floating point). Our PLC is reading a pressure transducer input whose max reading is 250 psi. In our PLC the max number is represented by 4095 (FFF in hex). So essentially to get our real world reading we would need to divide 4095 by 16.38 (4095 reading / 250 max pressure). This is easily done with real numbers but our reading is in decimal. So the BTOR instruction is used to convert the decimal number to a real number format. Then we use the special DIVR instruction to divide it with a real number and get our reading. The resulting ladder logic would look like below.

##### 01-09b real number example

If you're a complete newbie at this and don't understand the ladder logic then don't worry about that. We'll get into ladder latter. Just understand that when you need to deal in fractions you'll most likely want to turn to real number formats in the PLC instruction set.

If you're still staying afloat in all these concepts and want to understand more then read on...  
Sinking Deeper into Floating Point Numbers

Floating point is basically a representation of scientific notation. Oh yeah? What's scientific notation? Scientific notation represents numbers as a base number and an exponent. For example, 123.456 would be 1.23456 x 10<sup>2</sup>. That 10 with a little 2 above is telling us to move the decimal two spaces to the right to get the real number. Another example, 0.0123 would be 1.23 x 10<sup>-2</sup>. That little -2 indicates we move the decimal point in the opposite direction to

the left. (Just a heads up, in the PLC you may be able to use scientific notation but in a different form like 1.23456E2 which is the same as a first example.) The number 10 here means we're dealing in decimal. We could just as easily do scientific notation in hexadecimal (123.ABC x 16<sup>2</sup>) or even binary ( 1.0101 x 2<sup>2</sup>, this binary one becomes important later on).

### The Format

At some point in history a bunch of geeks got together and agreed upon a certain format or layout for a 32-bit floating point number. Due to a lack of originality, it officially became called "IEEE Standard 754". Here it is in all its glory.

#### 01-09a floating point format

First there is the sign bit. It doesn't get any easier than this. If the bit is 0 then the number is positive but if it is a 1 then it is negative. Flip the bit and you change the sign of the number. What power.

The exponent is the same as our little number above the 10 in scientific notation. It tells us which way the decimal should go so it needs to be positive (go to the right) or negative (go to the left). Here we are again trying to deal with negative numbers but in this case the geeks decided to use what's called a bias or offset of 127. Basically this means that at a value of 127 the exponent is 0. Any number below 127 will cause a negative exponent. Any number above 127 will be a positive exponent. So a stored value of 200 indicates an exponent of 73 (200-127).

The mantissa (or significand, if that is any easier to say) represent the precision bits of the number. In our example above it was the 1.23456 part of the scientific notation.

The final nomenclature in scientific notation would be: (sign) mantissa x baseexponent

Normally the base would be 10 but in this case it will be 2 since we are only dealing in binary. Since it's in base 2 (or binary) there's a little optimization trick that can be done to save one bit. Waste not, want not, you know. The trick comes about by realizing that scientific notation allows us to write numbers in many different ways. Consider how the number five can be

5.00 x 10<sup>0</sup>

0.05 x 10<sup>2</sup>

5000 x 10<sup>-3</sup>

These are all the same number. Floating point numbers are typically in a normalized form with one digit to the left of the decimal (i.e. 5.00 x 10<sup>0</sup> or 4.0 x 10<sup>3</sup>). The exponent is always adjusted to make this happen. In terms of using binary we'll always have a 1 in front (i.e. 1.0 x 2<sup>3</sup>). You wouldn't have 0.1 x 24 as it wouldn't be normalized. So in this case it's always safe to assume that the leading digit is a 1 and therefore we don't have to store it. That makes the mantissa actually 24 bits long when all we have are 23 bits of storage. Ah, what we do to save one bit.

### WARNING: It's Not a Perfect World

With all this power using floating point you are probably thinking, "I'll just use it all the time". There's a problem though as this method can actually lose some precision. In many cases it will be negligible and therefore well worth it to use real numbers. In other cases though it could cause significant errors. So beware.

Consider what would happen if the mantissa part of the floating point format was actually longer than 24 bits? Something has to give and what happens is the end is truncated, that is, it is cut off the end and lost.

Here's an example of a 32-bit number

11110000 11001100 10101010 00001111 which would be 4039944719 in decimal

In floating point with only 24 bits it would have to be

1.1110000 11001100 10101010 x 2<sup>31</sup> which when converted back would be

11110000 11001100 10101010 00000000 and therefore 4039944704 in decimal.

That's a difference of 15. During normal math this might not be of concern but if you are accumulating and totalizing values then that kind of error could really make the bean counters mad. This is simply a case of knowing your limitations.

#### Glutton for Punishment: Further Reading

There's more on this subject concerning things like double precision, overflow, zero and 'not a number' which you can read about in these excellent articles.

[What Every Computer Scientist Should Know About Floating-Point Arithmetic](#)

[IEEE Standard 754 Floating Point Numbers](#)

[Introduction to Floating point calculations and IEEE 754 standard](#)

ASCII ... and ye shall receive

ASCII Chart ">ASCII is an acronym for American Standard Code for Information Interchange. You can see now why they shortened it down to ASCII which is pronounced as 'askey' by us geeks. This is the most common code for the exchange of letters and control characters between computers and their peripheral devices like printers. It is a standard that provides an easy way for the letters on our monitors to get into a binary format that a computer can understand. Thus it also finds its way into our everyday PLC lives in form of serial transmission to a display or printer.

All the assigned ASCII codes can be represented with 7 bits. Remember from our binary chapter that 7 bits of binary represent 128 in decimal. Typical PLCs though have 8 bits of data and therefore the left over bit is used for parity checking just to make sure everything is transmitting correctly. The following chart shows all the assigned letters, numbers and control characters with the computer number on the left and the symbol on the right. Some of these look weird because this stuff goes way back to the days when you had to control teletypes and phone lines. These days I'm typically only concerned with 10 (line feed), 13 (carriage return) and 32 through 126.

[ASCII chart regular small picture](#)

[ASCII Chart in Word format available for download](#)  
[International Codes](#)

In a lot of ways ASCII is very restrictive because the A stands for American and so it basically serves only English. Until recently the computer world has been trying to catch up to the many different languages in the world and the need for computers to be able to use them all. Thus there are many different character sets for other languages. The Alphabet Soup is a great resource to locate a language and it's corresponding character set. The real advance though has come with the advent of Unicode which is one standard representing all the languages of the world. Unicode has and will continue to supersede all these other types of alphanumeric codes because it simplifies the use of multiple languages. The list of possibilities is endless so if you ever need to program in Ugaritic then you got it.

#### The ABCs of BCD

Binary Coded Decimal (BCD) is a number code that makes all those 1s and 0s in the computer easier for humans to read. In BCD the digits 0 through 9 are stored as 4 bits (a.k.a. a nibble). Thus in a 16-bit word you can have four digits (16 bit word / 4 BCD bits = 4 digits). The figure below demonstrates how to convert a binary number to its BCD equivalent.

01-10 binary to BCD by fours

The astute student will see that this is a lot like converting from binary to hexadecimal. In this case though there are no letters like A through F which is part of making it easier to read. In a pure binary sense the 16-bit word can have a value of 0 to 65535 but BCD limits us from 0 to 9999.

What is BCD good for?

Most modern control system components like PLCs and HMIs (for example AutomationDirect) make the use of BCD seamless. That is, you use them like you would decimal numbers without knowing the difference. There comes a time though when you'll need to know the difference (for instance when setting up an I/O card).

The real use of BCD comes in older types of controls like thumbwheel switches, 7-segment data displays and absolute encoders. For example, the thumbwheel switch will have (at least) four outputs for one digit which represent the binary numbers of that digit. Ganging the thumbwheel switches together then makes a word in BCD format.

#### 01-11 BCD component examples

#### The Black and White of Gray Code

Gray code (named after its inventor Frank Gray) is a sequence of binary numbers where only one bit changes at a time. Marching through the integer sequence then only requires flipping one bit at a time which in certain applications drastically reduces any errors. In standard binary many digits can change at once, for instant when going from 7 to 8 (0111 to 1000) there are four bits changing state.

It's a fact of life that a PLC input can turn ON quicker than it can turn OFF. We're talking milliseconds here but with equally fast scan times it can generate a wrong reading when using normal binary methods. With Gray code you can be assured that only one bit is going to change. Anymore than that one bit changing and there is something wrong.

The problem is demonstrated in the figure below. For example, let's say it takes an input signal 300 milliseconds to come ON but it takes 500 milliseconds to go OFF. In binary then when going from 7 to 8 the ON bit will come on first but the bits that are ON are still ON and therefore the decimal value is 15. It takes another 200 milliseconds for the ON bits to go OFF and achieve our proper reading of eight. On the other hand, with Gray code notice that only one bit is changing from OFF to ON and therefore there is no error.

#### 01-14 gray code transition benefit

Here's how the sequence in Gray code starts and you can compare how different it is to standard binary code. I've made it easy by emphasizing the bits that are changing.

| Decimal | Binary | People: Learning your 1s and 0s.">Binary | Gray Code |
|---------|--------|------------------------------------------|-----------|
| 0       | 0000   | 0000                                     |           |
| 1       | 0001   | 0001                                     |           |
| 2       | 0010   | 0011                                     |           |
| 3       | 0011   | 0010                                     |           |
| 4       | 0100   | 0110                                     |           |
| 5       | 0101   | 0111                                     |           |
| 6       | 0110   | 0101                                     |           |
| 7       | 0111   | 0100                                     |           |
| 8       | 1000   | 1100                                     |           |
| 9       | 1001   | 1101                                     |           |
| 10      | 1010   | 1111                                     |           |
| 11      | 1011   | 1110                                     |           |
| 12      | 1100   | 1010                                     |           |
| 13      | 1101   | 1011                                     |           |
| 14      | 1110   | 1001                                     |           |
| 15      | 1111   | 1000                                     |           |

Another neat trick is you'll notice that going from 15 to 0 still preserves our Gray code of only changing one bit. Pretty neat, eh? This is why they call it a cyclic code because it can go around in circles.

What is Gray Code good for?

In automation it's particularly good for position transducers used to measure the angle of a shaft. This application benefits from the cyclic nature of Gray codes, because the first and last values of the sequence are different by only one bit. That is, if you feel like going round and round in circles.

In an optical absolute encoder there is a disk with a Gray code pattern like so...

#### 01-12 Gray code encoder disc

A beam of light, LED or laser, is shot through the disk and the light is picked up on the other side by a gang of phototransistors. This light is converted into electrical signals which can be read as Gray code or converted to BCD. As the shaft (and therefore the disk) rotate the light coming through the holes changes but only slightly based on the Gray code. Thus the controller reading the phototransistors knows the exact position of the shaft.

#### 01-13 optical encoder system

Illustration: Example of an absolute encoder converting light to a Gray code.

Advanced Topic: Using Ladder Logic for Gray code conversion

#### The Decision Makers: AND, OR and NOT

Life is full of decisions. What is true for us is also true of PLCs. We gather information (input) and based on that we make choices that determine our output. All though I've always found computers to be quite a bit more logical then human beings.

For an example of how we use logic in everyday life consider these statements:

If Tommy OR Bob want to play basketball then I'll play too.

It's 6 o'clock AND I'm NOT hungry therefore I'm going to keep playing.

If Mom comes out AND orders me inside OR it get's dark then I'll stop playing.

Now these are pretty simple decisions especially if you're a ten year old boy. You'll notice that they all involve three types of comparisions: AND, OR and NOT. Now we could get more complex but all that we'd be doing is using these simple building blocks.

In the world of automation these types of TRUE or FALSE conditions come down to a device being ON or OFF, CLOSED or OPEN, PRESENT or ABSENT, 24 VOLTS or 0 VOLTS. In the PLC it all boils down to our now familiar binary system of a 1 or a 0. Typically having a bit ON represents a TRUE condition while OFF is FALSE. This is arbitrary though as it may make more sense to use what is called failsafe logic and have an ON bit as a FALSE condition.

Let's turn again to some simple statements but this time using automation examples.

When the button is pressed AND the door is closed then turn on the motor.

If the process is done OR the emergency stop button signal is NOT on then turn off the motor. (This is an example of a failsafe operation as the emergency stop button could be pressed or the wire has been disconnected. In either case we want to check this for safety reasons. Relying on a signal to turn on when a wire has fallen off long ago may cause an awkward moment when we truly have to stop the machine in an emergency.)

If the tank is full OR the button is pressed AND there are no alarms then start the process.

It would be nice to program like this but computers like to be a little bit more structured. A series of graphical objects have been used for years to represent these logic elements and they can be easily converted to a common ladder logic equivalent. These functions are also called gates as they act like gate keepers for different logic.

### The NOT function

The simplest of all logic functions is the NOT gate.

Its sole function in life is to invert or flip the logic state. So an input of 1 will come out as a 0 and visa versa. Shown below is a truth table (it doesn't lie) showing all possible inputs and the resulting logical output.

| Input A | Output |
|---------|--------|
| 0       | 1      |
| 1       | 0      |

The ladder logic equivalent for a NOT function looks like a normal contact but with a slash through it.

### The AND function

The AND gate is associated with the following symbol that can have any number of inputs but only one output.

The truth table below shows that the output is only turned on when all the inputs are true (1). An easy way to remember this is AND works like multiplication.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 0      |
| 1       | 0       | 0      |
| 0       | 1       | 0      |
| 1       | 1       | 1      |

The ladder logic equivalent for an AND function looks like two normal contacts side by side.

### The OR function

Last but not least the OR gate is associated with the following symbol that also can have any number of inputs but only one output.

The truth table below shows that the output is turned on (1) when any of the inputs are true (1). An easy way to remember this is OR works like addition.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 0      |
| 1       | 0       | 1      |
| 0       | 1       | 1      |

1      1      1

The ladder logic equivalent for an OR function looks like two normal contacts on top of each other.

### Combining AND or OR with NOT

The NOT gate might not look like much help if you haven't programmed much but you'll find yourself actually using it frequently. It's very common to use it in combination with AND and OR. So the engineering gods decided to make some symbols for these combinations.

Putting the NOT and AND gates together forms the NAND gate. The truth table below shows that it is simply an inverted output of the AND gate.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 1      |
| 1       | 0       | 1      |
| 0       | 1       | 1      |
| 1       | 1       | 0      |

A little circle (or if you like, a bubble) at the end of a AND gate is used to signify the NAND function. Its symbol and corresponding ladder logic are shown below. Now pay close attention to the ladder logic because the contacts are in parallel and not in series like the AND function.

Putting the NOT and OR gates together forms... you got it... the NOR gate. The truth table below shows that it is simply an inverted output of the OR gate.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 1      |
| 1       | 0       | 0      |
| 0       | 1       | 0      |
| 1       | 1       | 0      |

Again a little circle is placed at the end of an OR gate to signify the NOR function. Its symbol and corresponding ladder logic are shown below. The ladder logic is very different from the regular OR gate.

But wait! Don't order yet... the XOR gate.

So far with our logic gates we've covered almost all possible combinations except for one shown by the truth table below.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 0      |
| 1       | 0       | 1      |
| 0       | 1       | 1      |

1      1      0

The logic to produce this output is called an Exclusive OR gate otherwise known as the XOR gate. It's a specialized form of the OR gate. So if either one of the inputs are on then the output is true, otherwise you're out of luck. The symbol for the XOR gate is shown by adding a curved line to the OR gate symbol.

The ladder logic to implement an XOR gate is a little more complex than the others.

How useful is the XOR logic? You probably use the XOR gate everyday without thinking about it if you have a room with a light that works off two switches. If both switches are in the same position then the light will be off. Therefore just flipping one switch will turn the light on. In the PLC program this can be extremely useful for programming alternating actions or gray codes.

Ok, there is one more logic gate but I promise it is the last one. It makes sense that there is a XNOR gate which is the combination of the NOT and XOR logic. It simply inverts the output of the XOR function.

| Input A | Input B | Output |
|---------|---------|--------|
| 0       | 0       | 1      |
| 1       | 0       | 0      |
| 0       | 1       | 0      |
| 1       | 1       | 1      |

The symbol for the XNOR gate is shown below along with its ladder logic equivalent.

### Hope for the Future

While these terms and symbols may seem a bit esoteric for the PLC beginner they are important in the long run. A good grasp of these essentials will make PLC programming easier, simpler and save memory. The next lesson will make this clearer when we consider Boolean algebra.

### What is the acronym CISC stand for

-Complex instruction set computer

### The MicroChip Atmel AVR architecture is based on which of the following computer architecture(s)

-Harvard

### Compare and contrast the Harvard architecture with the von Neumann architecture.

-The Harvard type uses a separate program and data memory

### Using microcontrollers, GPIO is defined as

- General Program Input and Output

### When should we use pointers

(Unsure of answer)

-all of the answers

What is the definition of DAC?

-Digital Analog Converter

FPU used in Intel Architecture is defined as which of the following

- Floating point unit

The doubling of performance and having of the cost in the electronics industry is which of the following

- Moore's Law