Points missed: _____ Student's Name: _____

Total score: _____/100 points

East Tennessee State University
Department of Computer and Information Sciences
CSCI 2150 (Tarnoff) – Computer Organization
TEST 3 for Spring Semester, 2004

Section 001 & 002

Read this before starting!

- The total possible score for this test is 100 points.
- This test is closed book and closed notes.
- **All** answers **must** be placed in space provided. Failure to do so may result in loss of points.
- **1 point** will be deducted per answer for missing or incorrect units when required. **No** assumptions will be made for hexadecimal versus decimal, so you should always include the base in your answer.
- If you perform work on the back of a page in this test, indicate that you have done so in case the need arises for partial credit to be determined.
- ***Calculators are not allowed.*** Use the tables below for any conversions you may need. Leaving numeric equations is fine too.

Binary	Hex
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

Binary	Hex
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

Power of 2	Equals
2^3	8
2^4	16
2^5	32
2^6	64
2^7	128
2^8	256
2^9	512
2^{10}	1K
2^{20}	1M
2^{30}	1G

"Fine print"

Academic Misconduct:

Section 5.7 "Academic Misconduct" of the East Tennessee State University Faculty Handbook, June 1, 2001:

"Academic misconduct will be subject to disciplinary action. Any act of dishonesty in academic work constitutes academic misconduct. This includes plagiarism, the changing of falsifying of any academic documents or materials, cheating, and the giving or receiving of unauthorized aid in tests, examinations, or other assigned school work. Penalties for academic misconduct will vary with the seriousness of the offense and may include, but are not limited to: a grade of 'F' on the work in question, a grade of 'F' of the course, reprimand, probation, suspension, and expulsion. For a second academic offense the penalty is permanent expulsion."

NAME: _____

DEC - Decrement
Usage: DEC dest
Modifies flags: AF OF PF SF ZF
Description: Unsigned binary subtraction of one from the destination.

INC - Increment
Usage: INC dest
Modifies flags: CF AF OF PF SF ZF
Description: Adds one to destination unsigned binary operand.

Jxx - Jump Instructions Table

Mnemonic	Meaning	Jump Condition
JA	Jump if Above	CF=0 and ZF=0
JE	Jump if Equal	ZF=1
JG	Jump if Greater (signed)	ZF=0 and SF=OF
JGE	Jump if Greater or Equal (signed)	SF=OF
JL	Jump if Less (signed)	SF != OF
JMP	Unconditional Jump	unconditional
JNB	Jump if Not Below	CF=0
JNE	Jump if Not Equal	ZF=0
JNG	Jump if Not Greater (signed)	ZF=1 or SF != OF
JNL	Jump if Not Less (signed)	SF=OF
JZ	Jump if Zero	ZF=1

MOV - Move Byte or Word
Usage: MOV dest,src
Modifies flags: None
Description: Copies byte or word from the "src" operand to the "dest" operand.

NOT - One's Compliment Negation (Logical NOT)
Usage: NOT dest
Modifies flags: None
Description: Inverts the bits of the dest operand forming the 1s complement.

POP - Pop Word off Stack
Usage: POP dest
Modifies flags: None
Description: Transfers word at the current stack top (SS:SP) to the destination then increments SP by two to point to the new stack top. CS is not a valid destination.

PUSH - Push Word onto Stack
Usage: PUSH src
Modifies flags: None
Description: Decrements SP by the size of the operand (two or four, byte values are sign extended) and transfers one word from source to the stack top (SS:SP).

SAL/SHL - Shift Arithmetic Left / Shift Logical Left
Usage: SAL dest,count SHL dest,count
Modifies flags: CF OF PF SF ZF (AF undefined)
Shifts the destination left by "count" bits with zeroes shifted in on right. The Carry Flag contains the last bit shifted out.

SAR - Shift Arithmetic Right
Usage: SAR dest,count
Modifies flags: CF OF PF SF ZF (AF undefined)
Shifts the destination right by "count" bits with the current sign bit replicated in the leftmost bit. The Carry Flag contains the last bit shifted out.

Answer questions 1 through 10 based on the following settings of the 8086 registers.

AX = 1234h	BP = 1212h	CS = A101h
BX = 8721h	SP = 3434h	DS = B101h
CX = 5678h	DI = 5656h	SS = C101h
DX = 8765h	SI = 7878h	ES = D101h

1. What is the value in the register AL? (2 points) **34h**

2. What is the physical address pointed to by ES:BP? (3 points)

   D101h:1212h = D1010h + 1212h = D2222h

3. True or false: The physical address of the next instruction to be executed by the processor can be calculated from the above data? (2 points)

   The next instruction to execute is pointed to by CS:IP. CS is in the above list of registers, but IP is not. Therefore, the answer is false.

4. True or false: The physical address of the top of the stack can be calculated from the above data? (2 points)

   The top of the stack is pointed to by SS:SP. Since both of these registers are available above, the answer is true.

5. What is the value of CX after the execution of the instruction **NOT CX**? (2 points)

   CX = 5678h = 0101 0110 0111 1000_2

   Looking at the list of instructions on page 1 shows us that NOT takes the 1's complement, i.e., it inverts all of the bits. Inverting all of the bits of CX gives us:

   ~0101 0110 0111 1000_2 = **1010 1001 1000 0111_2**

6. What is the value of SP after the execution of the instruction **POP AX**? (2 points)

   Looking at the list of instructions on page 1 shows us that POP retrieves the value store on the stack, then, "increments SP by two to point to the new stack top." This means that the value in SP will have two added to it.

   SP + 2 = 3434h + 2 = **3436h**.

7. What is the value of SS after the execution of the instruction **POP AX**? (2 points)

   POP does not affect SS, therefore, it stays the same: SS = **C101h**.

8. Assume that the instruction **`INC CL`** is executed. How would the following flags be set? ***Write "N/A" if the flag was not affected.*** (3 points)

First check the instruction set to see which flags are affected by INC. It turns out that ZF, CF, and SF are all affected. Next, see what value results in CL after an INC CL command. CL contains 78h = 0111 1000_2, so incrementing it will make it 79h = 0111 1001_2. The MSB is 0, so it is a positive number, i.e., SF = 0, there was no carry out, so CF = 0, and the result does not equal zero, so ZF = 0.

ZF = __0___ CF = ___0__ SF = ___0__

9. Assume that the instruction **`SAL DH,3`** is executed. How would the following flags be set? ***Write "N/A" if the flag was not affected.*** (3 points)

First check the instruction set to see which flags are affected by SAL. It turns out that ZF, CF, and SF are all affected. Next, examine how the instruction SAL affects the register DH. It turns out that the value that is in DH will be shifted left 3 bit positions with zeros filling in from the right. The last bit shifted out the left side of DH will be placed in the carry flag, CF. (DH = 87h = 1000 0111_2)

1 ← 0000 1110 First shift

0 ← 0001 1100 Second shift

0 ← 0011 1000 Third shift

ZF = __0___ CF = ___0__ SF = ___0__

10. Assume that the instruction **`SAR BH,3`** is executed. What would the new value of BH be? (2 points)

Begin by examining how the instruction SAR affects the register BH. It turns out that the value that is in BH will be shifted right 3 bit positions with the sign bit replicating itself in the MSB position. The last bit shifted out the right side of BH will be placed in the carry flag, CF. (BH = 87h = 1000 0111_2)

1100 0011 → 1 First shift

1110 0001 → 1 Second shift

1111 0000 → 1 Third shift

Therefore, the final value that is in BH will be **F0h**.

11. Circle ***ALL*** of the registers affected by the instruction **`JMP ADD_LOOP`**. (2 points)

a.) BP b.) CS (c.) IP d.) SS e.) SP f.) None of these

12. Circle ***ALL*** of the registers affected by the instruction **`POP AX`**. (2 points)

(a.) AL b.) CS (c.) IP d.) SS (e.) SP f.) None of these

13. List the two benefits of segmented addressing. (3 points)

Being able to create a 20-bit address from two 16-bit registers and allowing the O/S to relocate the code where it needs too.

14. Assume the register BX contains the value 1500h and the table to the right represents the contents of a short portion of memory. Indicate what value AL contains after each of the following MOV instructions. (2 points each)

Address	Value
DS:14FF	F5h
DS:1500	E4h
DS:1501	D3h
DS:1502	C2h
DS:1503	B1h
DS:1504	A0h

MOV AL, DS:[1503h] AL = **B1h**
MOV AL, DS:[BX] AL = **E4h**
MOV AX, BX AL = **00h**
MOV AL, DS:[BX-1] AL = **F5h**

15. If a processor takes 3 cycles to execute any instruction (fetch, decode, execute), how many cycles would a non-pipelined processor take to execute 8 instructions? (3 points)

8 × 3 = 24 cycles

16. If a processor takes 3 cycles to execute any instruction (fetch, decode, execute), how many cycles would a pipelined processor take to execute 8 instructions assuming no jumps? (3 points)

2 + 8 = 10 cycles

17. Of the following jump instructions, indicate which ones will jump to the address LOOP, which ones will simply execute the next address (i.e., not jump), and which ones you don't have enough information to tell.

Instruction		Current Flags	Jump to LOOP	Not jump to LOOP	Cannot be determined	
JA	LOOP	SF=0, OF=1, CF=1	❑	☑	❑	(2 points)
JNL	LOOP	SF=1, ZF=0, OF=0	❑	☑	❑	(2 points)
JMP	LOOP	SF=0, ZF=0, OF=0	☑	❑	❑	(2 points)
JL	LOOP	ZF=0, SF=0, OF=1	☑	❑	❑	(2 points)

18. Assume AX=1000h, BX=2000h, and CX=3000h. After the following code is executed, what would AX, BX, and CX contain? (3 points)

```
PUSH CX
PUSH BX
PUSH AX
POP BX
POP CX
POP AX
```

Place your answers in space below:

AX = **3000h**

BX = **1000h**

CX = **2000h**

19. Using an original value of 10101010_2 and a mask of 11110000_2, calculate the results of a bitwise AND, a bitwise OR, and a bitwise XOR for these values. (2 points each)

Original value	Bitwise operation	Mask	Result
10101010_2	AND	11110000_2	**10100000_2**
10101010_2	OR	11110000_2	**11111010_2**
10101010_2	XOR	11110000_2	**01011010_2**

20. List two of the five benefits discussed in class for using glass as a substrate for a hard drive disk. (3 points)

- improved surface uniformity which increases reliability;
- lower amount of surface defects which reduces read/write errors;
- better stiffness;
- better resistance to shock; and
- ability to have the read/write mechanism ride closer to the surface allowing for better data density.

21. Which method of laying out the sectors on a hard disk allows for a simpler controller, constant angular velocity or multiple zoned recording? (2 points)

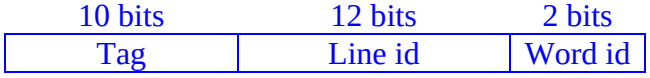
Constant angular velocity

22. True or false: A Winchester head actually comes to rest on the hard drive disks when the disks are not spinning. (2 points)

23. Assume the table below represents a small section of a cache. The complete cache has 4K lines and 4 bytes per block. A block containing the address 0x3514a5 is not contained in the cache. When loaded, which row (a through f) and column (0 through 3) will its value be stored in? (4 points)

Line number (dec/binary)	Tag (binary)	Word within block				
		00	01	10	11	
1318_{10} = 010100100110	0101001101	0x23	0x56	0xaf	0x3d	row a
1319_{10} = 010100100111	0100110110	0x12	0xc4	0xa1	0x00	row b
1320_{10} = 010100101000	1010110111	0x34	0x3a	0x09	0x03	row c
1321_{10} = 010100101001	0110101111	0x58	0xa5	0x56	0x76	row d
1322_{10} = 010100101010	1011010101	0x67	0xa2	0xff	0xf4	row e
1323_{10} = 010100101011	1111000100	0x9a	0xa3	0xf2	0xf3	row f
		col 0	col 1	col 2	col 3	

A cache with 4K = $2^2 \times 2^{10} = 2^{12}$ lines uses 12 bits to identify the line number. To address a word within a block of four words, two bits are needed. The remaining bits are the tag. Therefore, the figure below shows the allocation of the address bits to the different purposes of storing data within a cache.



Now, if we convert 0x3514a5 to binary, and use the figure above to divide it into its components, we get:

$$3514A5_{16} = 0011\ 0101\ 0001\ 0100\ 1010\ 0101_2$$

The first ten bits, 0011010100, are the tag, the next 12 bits, 010100101001, are the line number, and the last two bits, 01, identify the word. Since $010100101001 = 1321_{10}$ = row d (identified from the table), and 01 = column 1, the answer is:

Row d, column 1

24. True or False: A block from main memory could possibly be stored in ***any*** line of a cache using direct mapping (the mapping method we talked about in class). (2 points)

Direct mapping associates exactly one line of the cache with any particular block from main memory.

The next 2 questions are based on the following breakdown of address bits for a direct mapping cache RAM.

7 tag bits	10 line id bits	3 word id bits

25. If each address contains a byte, how many bytes are in a block of memory? (2 points)

The number of words (in this case bytes) allocated to a block is equivalent to the number of unique patterns of ones and zeros possible with the bits of the word id. Therefore, the number of bytes to a block = $2^3 = 8$.

26. If each line in the cache contains a block, how many lines are in the cache? (2 points)

The number of lines is determined by the number of line id bits. Since there are 10, then there are 2^{10} = 1K lines in the cache.

27. In the Ethernet frame format discussed in class, what is the purpose of the 7 bytes of alternating ones and zeros sent at the beginning of the frame? (2 points)

Allows the receiver to synchronize its timing with the sending device.

28. The 8th byte (start delimiter) of the Ethernet frame discussed in class is similar to those described in the previous problem. Write the binary value of this 8-bit number in the space below. (2 points)

10101011

29. True or false: The two-byte ***length field*** of the Ethernet frame discussed in class contains the length of the entire message including preamble and start delimiter. (2 points)

The length field only shows the length of the data field.

30. True or false: Even though the length field could contain values up to 65,536, the maximum length of the data field of an Ethernet frame is 1500 bytes. (2 points)
31. True or false: The minimum length of the data field of an Ethernet frame is 46 bytes. (2 points)
32. True or false: All devices can see all of the messages that pass across their Ethernet network, even the ones not meant for them. (2 points)

33. True or false: If two devices try to transmit at the same time a collision occurs meaning that only one message got through and one device will have to retransmit. (2 points)

    In the case of a collision, ***both*** devices must retransmit.

34. Classify each of the following characteristics as RS232 serial (R), USB (U), Firewire(F), GPIB (G), or SCSI (S). (2 points each)

    ___**G**___ Primarily used for scientific instrumentation
    ___**U**___ Uses two data connection speeds, one for slow devices & one for fast, on same network
    ___**R**___ Serial point-to-point (only 2 devices) communications
    ___**F**___ Developed by Apple for the transmission of video and audio