Solutions to Exam One

CS130 - Computer Organization and Assembly Language
Drake University - Fall, 2004

Directions: Do all problems. Show all work. Please work first on problems with which you are more comfortable.

Problem 1 - Von Neumann achitecture. (15 points)

Explain in some detail what happens during a single fetch-decode-execute cycle in a modern computer system. Use terminology correctly wherever appropriate.

Solution: (You should discuss the program counter, a.k.a. instruction pointer, and the instruction queue when giving details about the fetch phase. Everything else seemed fine on this one.)

Problem 2 - Number systems. (25 points)

(a) Convert the decimal number 12345 to binary.

Solution: 11000000111001

(b) Convert the hexadecimal number 3E5F to decimal.

Solution: 15967

(c) What decimal number is represented using 8-bit the two's complement system by the bit pattern 11001010?

Solution: -54

(d) What range of values are represented using the 12-bit two's complement number system?

Solution: -2048 to 2047

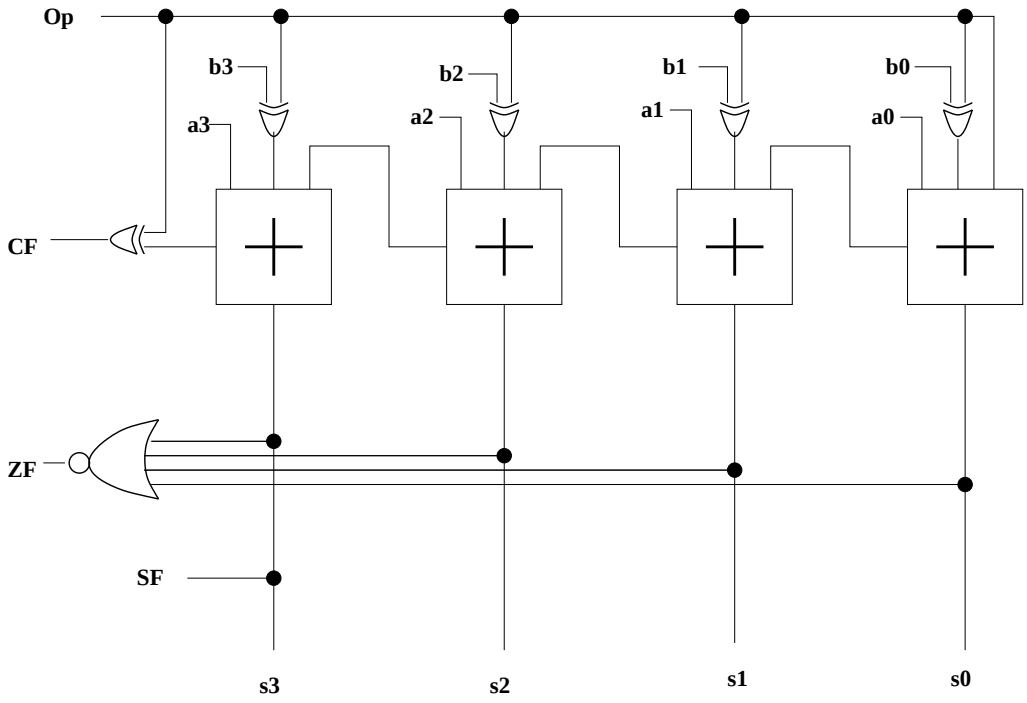
(e) If I understand correctly, some edition of the Oxford English dictionary contains 616,500 words (see http://www.wordorigins.org/number.htm for details). How many bits are required to represent (encode) 616,500 different things so that each thing corresponds to a different bit pattern?

Solution: 20 bits

Problem 3 - 4-bit adder/subtractor circuitry. (20 points) For each of the following problems, show how EVERY line in the circuit gets set (0 or 1), and how the OF bit gets set. Then indicate whether or not the computer's answer is valid.

(*Don't have the patience to put in solutions here. Let me know if you have questions.*)

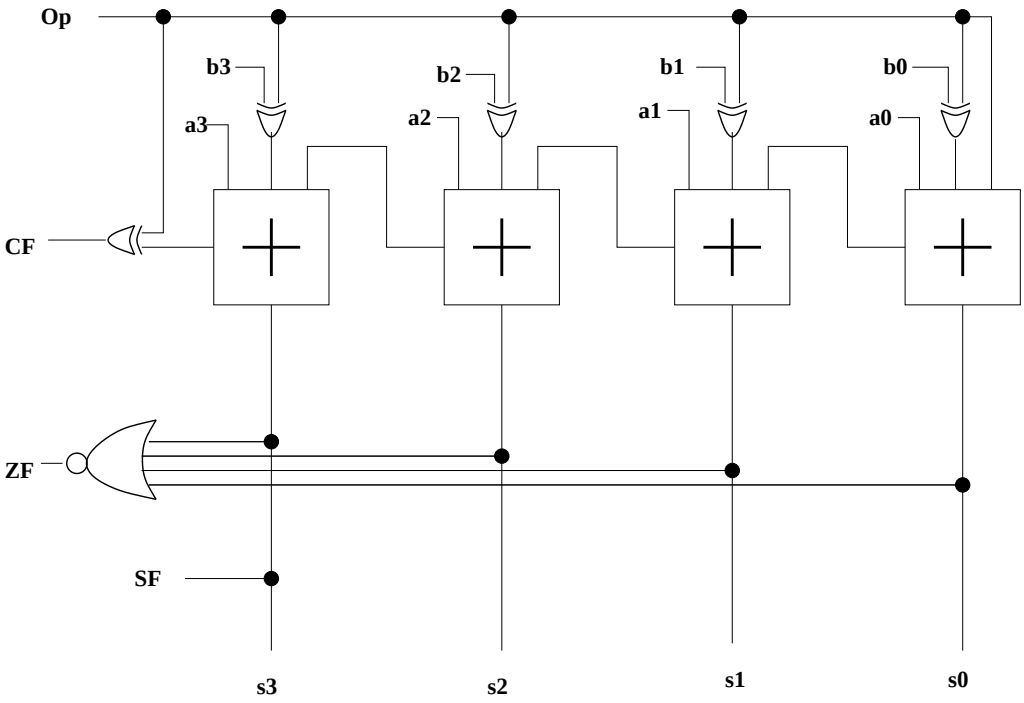
(a) Do the unsigned addition 3 plus 5.



Adder/Subtractor Circuit (part of ALU)

Note: the OF bit should be included here. It depends on Op, a3, b3 and s3. We'll deal with it elsewhere.

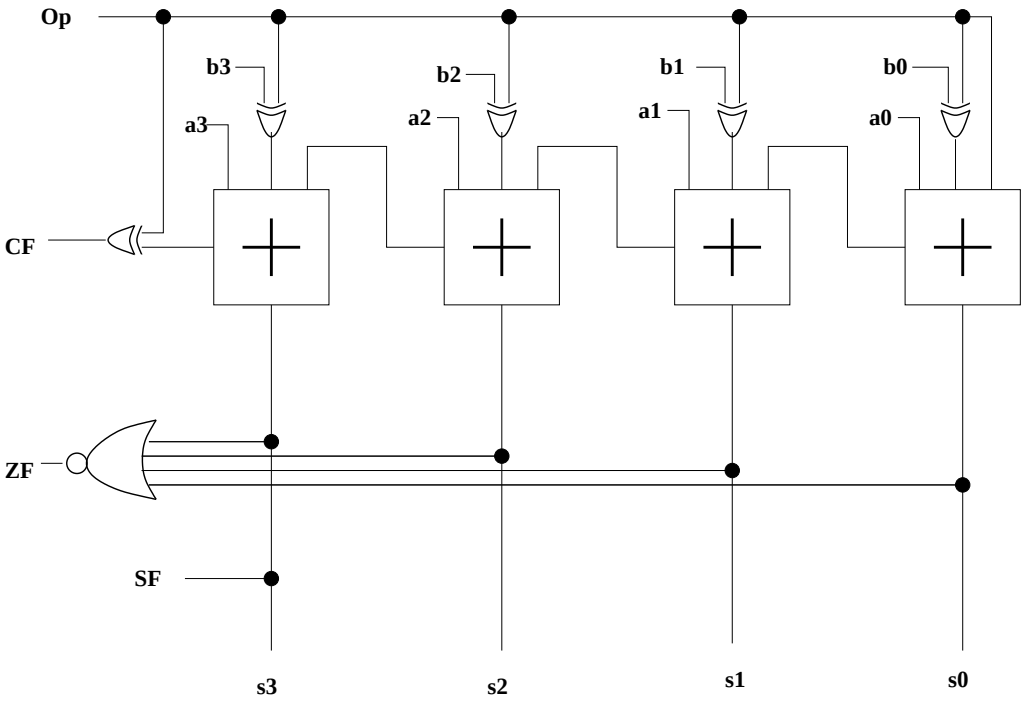
(b) Do the signed addition -4 plus -6.



Adder/Subtractor Circuit (part of ALU)

Note: the OF bit should be included here. It depends on Op, a3, b3 and s3. We'll deal with it elsewhere.

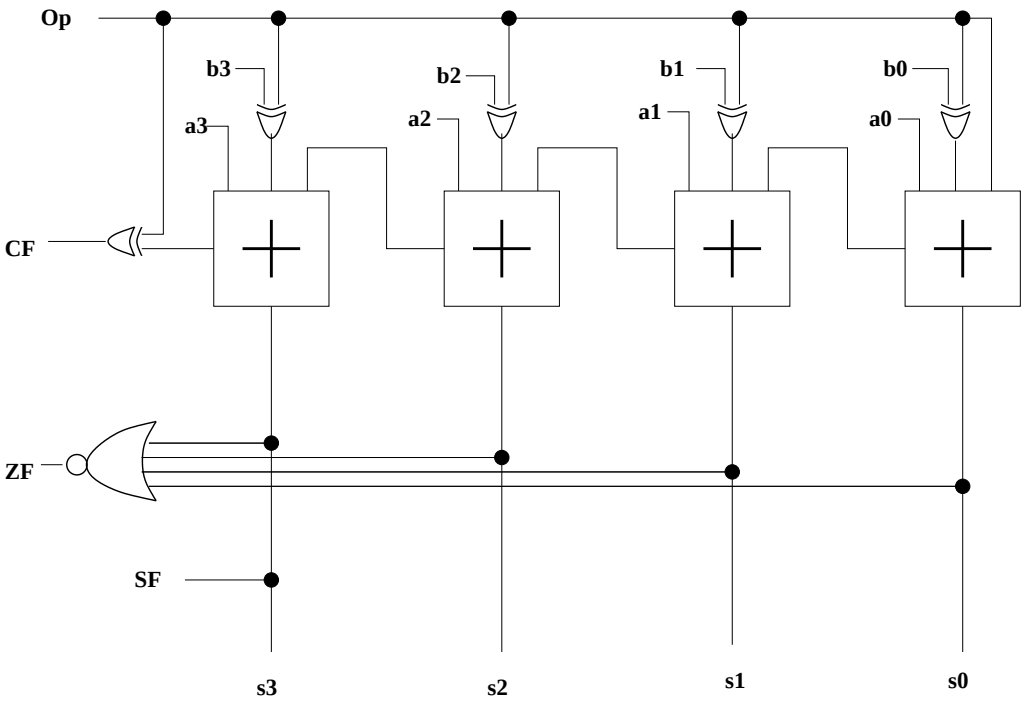
(c) Do the unsigned subtraction 3 minus 5



Adder/Subtractor Circuit (part of ALU)

Note: the OF bit should be included here. It depends on Op, a3, b3 and s3. We'll deal with it elsewhere.

(d) Do the signed subtraction -1 minus -1



Adder/Subtractor Circuit (part of ALU)

Note: the OF bit should be included here. It depends on Op, a3, b3 and s3. We'll deal with it elsewhere.

Problem 4 - MASM source code. (25 points) Go through the following MASM source code and explain the meaning of each line, distinguishing between directives and assembly language instructions. You can skip the lines that contain a semicolon. However, be sure to clearly show that you understand the rest.

(*Don't have the patience to put in solutions here. Let me know if you have questions.*)

```
          .MODEL SMALL ;

          .STACK 1024 ;

          NUM_LOOPS = 3

          .DATA

DUCK      WORD 11B

DUCK2     WORD 11H

GOOSE     WORD 11

          .CODE

MAIN      PROC

          MOV AX, @DATA ;

          MOV DS, AX ;

          MOV CX, NUM_LOOPS

          MOV AX, DUCK

          MOV DUCK2, AX

L1:       MOV AX, GOOSE

          MUL DUCK

          MOV DUCK, AX

          ADD GOOSE, AX

          LOOP L1

          MOV AX, 4CH ;

          INT 21H ;

MAIN      ENDP

END       MAIN
```

Problem 5 - Protected mode addressing. (15 points)

Show your reasoning in the following. Assume that the contents of the active descriptor table are as follows, with each descriptor being a quad word represented here in hexadecimal, and with higher memory being higher on this page, and with a question mark signifying a hex digit that I don't care about:

```
.................
.................
.................
4D3567EF72901???
78AB300C39284???
24201A9DEF003???
92991101004E9???
559DE93F03422???
639A702B34F58???
38847473EF2F1???
93FD80A9AB333???
2203EF5699E25???
```

Assume that the granularity bit of each descriptor is set (to one). Assume that the `CS` register has hex value 0020.

(a) What is the base address of the code segment?

Solution: Going 20 bytes (locations) into the descriptor table, we pull out the descriptor 559DE93F03422???h. The high-order half of this is 559DE93Fh. This is the base address of the code segment.

(b) What is the size of the code segment?

Solution: The "limit" field of the descriptor is 03422h. Since the granularity bit is set, the size of the code segment is 03422000h.

(c) Using hex, indicate one possible valid value for the `EIP` register, and one invalid value? Indicate which is which and explain your answer.

Solution: Anything less than 03422000h (as unsigned integer) would be value. Anything greater than or equal to 03422000h would be invalid.