QATAR UNIVERSITY
College of Engineering
Dept. of Computer Science & Engineering

Homework #3
10 Points

Computer Architecture and Organization I: CMPE 263
Spring 2023 Semester

Due: April 06, 2023 (*@end of the day*)

Name:_____ **QUID:** _____

Instructions

1. You can print this document and use a pen or pencil to write your solutions. When you have solved all problems, then you can open the online HW2 folder and enter your solutions on BB directly. It may be possible that order of questions / solutions in the online BB version will be different. **Only one online attempt is permitted**.
2. Please note that the submitted work must represent your own thinking and effort. *Copying the work of others will not be tolerated*.
3. Please submit within the due time (see above). Late submissions are not accepted.

Chapter 13 & Chapter 14

Some of the questions related to the textbook questions, but need some alterations for some data values. Each questions marked (0.4)

1. Convert the following arithmetic expressions from infix to reverse Polish notation (RPN).

$A + B * [\ C * (D - E) + \frac{F}{G}\ ]$

- D E – C * F G / B * A +
- D E C - * F G / B * A +
- D E* C – F / G B * A +
- D E – C * F / G B A * +

2. Convert the following arithmetic expressions from infix to reverse Polish notation (RPN).

$\frac{A * B + C * [\ D + (\ E * F\ )\ ]}{G + H/K}$

- A B * E * F * D + C * + H / K G + /
- A B * E F * D C + * + H K / G + /
- A B * E F * D + C * + H K / G + /
- A B E F * * D + C * + H K / G + /

3. Convert the following arithmetic expressions from reverse Polish notation (RPN) to infix notation. A B C * + D / E F + *
   - $\frac{(B * C) + A}{D} * (E + F)$
   - $\frac{(B + C) * A}{D} * (E + F)$
   - $\frac{(B * C) * A}{D} + (E + F)$
   - $\frac{(B + C) + A}{D} * (E * F)$

4. Convert the following arithmetic expressions from reverse Polish notation (RPN) to infix notation. A B C D E F G + * + * + *
   - (((F * G) + E + D) + C + B) * A
   - (((F + G) * E + D) * C * B) + A
   - (((F * G) * E * D) * C * B) * A
   - (((F + G) * E + D) * C + B) * A

5. In _____ the operand is specified in the instruction itself.
   - Immediate addressing
   - Register mode
   - Implied addressing
   - Register Indirect

6. In which mode is the operand placed in one of the 8-bit or 16-bit general-purpose registers??
   - Immediate addressing
   - Register mode
   - Implied addressing
   - Register Indirect

7. Zero address instructions are designed with implied addressing mode.
   - True
   - False
   - Can be true or false
   - None of them

8. The addressing mode(s), which uses the PC instead of a general-purpose register is …..
   - Indexed with offset
   - Direct addressing mode
   - Relative mode
   - Both are Indexed with offset and direct

9. The instruction is stored at location 300 with its address field at location 301. The address field has a value of 400. A processor register R1 contains the number 200. Evaluate the effective address if the addressing mode of the instruction is direct;

   This figure will be used for questions from 9 – 13

   - Direct mode: 400
   - Direct mode: 301
   - Direct mode: 600
   - Direct mode: 701

		Memory
	PC →300	Opcode Mode
R1 = 200	301	400
	302	Next instruction

10. The instruction is stored at location 300 with its address field at location 301. The address field has a value of 400. A processor register R1 contains the number 200. Evaluate the effective address if the addressing mode of the instruction is immediate;

    - Immediate: 302
    - Immediate: 201
    - Immediate: 301
    - Immediate: 300

11. The instruction is stored at location 300 with its address field at location 301. The address field has a value of 400. A processor register R1 contains the number 200. Evaluate the effective address if the addressing mode of the instruction is Relative Addressing mode;
    - Relative: 300 + 200 = 500
    - Relative: 302 + 200 = 502
    - Relative: 301 + 400 = 701
    - Relative: 302 + 400 = 702

12. The instruction is stored at location 300 with its address field at location 301. The address field has a value of 400. A processor register R1 contains the number 200. Evaluate the effective address if the addressing mode of the instruction is Register Indirect mode;
    - Register Indirect: 200
    - Register Indirect: 300
    - Register Indirect: 301
    - Register Indirect: 302

13. The instruction is stored at location 300 with its address field at location 301. The address field has a value of 400. A processor register R1 contains the number 200. Evaluate the effective address if the addressing mode of the instruction is Index Register mode;
    - Indexed Register: 300 + 400 = 700
    - Indexed Register: 301 + 400 = 701
    - Indexed Register: 302 + 400 = 702
    - Indexed Register: 200 + 400 = 600

14. An 8-bit computer has a register R. Determine the values of status bits C, S, Z, and V as shown in the figure below after each of the following instructions. The initial value of register R is hexadecimal 72. The digits below are also in hexadecimal. Add immediate operand C6 to R=?.

This figure will be used for Q14-Q18

Mnemonic	Branch condition	Tested condition
BZ	Branch if zero	Z = 1
BNZ	Branch if not zero	Z = 0
BC	Branch if carry	C = 1
BNC	Branch if no carry	C = 0
BP	Branch if plus	S = 0
BM	Branch if minus	S = 1
BV	Branch if overflow	V = 1
BNV	Branch if no overflow	V = 0
Unsigned compare conditions (A - B)		
BHI	Branch if higher	A > B
BHE	Branch if higher or equal	A ≥ B
BLO	Branch if lower	A < B
BLOE	Branch if lower or equal	A ≤ B
BE	Branch if equal	A = B
BNE	Branch if not equal	A ≠ B
Signed compare conditions (A - B)		
BGT	Branch if greater than	A > B
BGE	Branch if greater or equal	A ≥ B
BLT	Branch if less than	A < B
BLE	Branch if less or equal	A ≤ B
BE	Branch if equal	A = B
BNE	Branch if not equal	A ≠ B

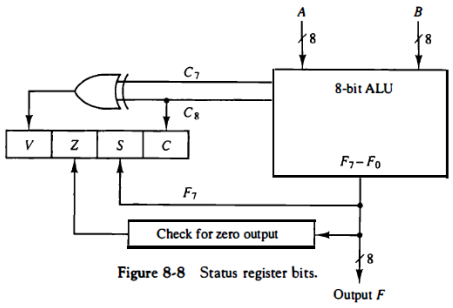


Figure 8-8 Status register bits.

- $(72 + C6 = 138)_{16}$: 01110110 + 11000110 = 00111100: Flags: C = 1, S = 1, Z = 0, V = 0
- $(72 + C6 = 128)_{16}$: 01110000 + 11000010 = 00111010: Flags: C = 1, S = 0, Z = 0, V = 1
- $(72 + C6 = 138)_{16}$: 01110010 + 11000110 = 00111000: Flags: C = 1, S = 0, Z = 0, V = 0
- $(72 + C6 = 138)_{16}$: 01110010 + 11010110 = 01111000: Flags: C = 1, S = 0, Z = 1, V = 0

15. An 8-bit computer has a register R. Determine the values of status bits C, S, Z, and V as shown in the figure below after each of the following instructions. The initial value of register R is hexadecimal 72. The digits below are also in hexadecimal. Add immediate operand 1E to R=?.

- $(72 + 1E = 80)_{16}$: 01110010 + 00011110 = 10011000: Flags: C = 0, S = 1, Z = 0, V = 1
- $(72 + 1E = 90)_{16}$: 01110010 + 00011110 = 10010000: Flags: C = 0, S = 1, Z = 0, V = 1
- $(72 + 1E = 70)_{16}$: 01110010 + 00011110 = 10010100: Flags: C = 0, S = 1, Z = 0, V = 1
- $(72 + 1E = 85)_{16}$: 01110010 + 00011110 = 10010101: Flags: C = 0, S = 1, Z = 0, V = 1

16. An 8-bit computer has a register R. Determine the values of status bits C, S, Z, and V as shown in the figure below after each of the following instructions. The initial value of register R is hexadecimal 72. The digits below are also in hexadecimal. Subtract immediate operand 9A from R?.

- $(72 - 9A = D8)_{16}$: 01110010 + 01100110 = 11011010: Flags: C = 1, S = 1, Z = 0, V = 1 (Borrow = 1
- $(72 - 9A = D8)_{16}$: 01110010 + 01100110 = 11011100: Flags: C = 0, S = 1, Z = 1, V = 1 (Borrow = 1
- $(72 - 9A = D8)_{16}$: 01110010 + 01100110 = 11011001: Flags: C = 0, S = 1, Z = 0, V = 0 (Borrow = 1
- $(72 - 9A = D8)_{16}$: 01110010 + 01100110 = 11011000: Flags: C = 0, S = 1, Z = 0, V = 1 (Borrow = 1

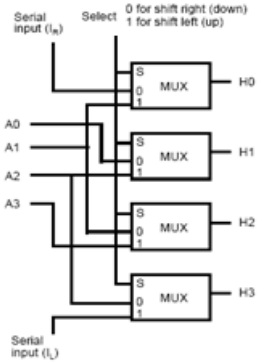
17. An 8-bit computer has a register R. Determine the values of status bits C, S, Z, and V as shown in the figure below after each of the following instructions. The initial value of register R is hexadecimal 72. The digits below are also in hexadecimal. *AND immediate operand 8D to R?*.

- (72 • 8D = 00)16: 01110010 + 10001100 = 00000000: Flags: C = 0, S = 0, Z = 1, V = 0
- (72 • 8D = 00)16: 01110010 + 10001100 = 00110000: Flags: C = 0, S = 0, Z = 1, V = 0
- (72 • 8D = 00)16: 01110010 + 10001100 = 11000000: Flags: C = 0, S = 0, Z = 1, V = 0
- (72 • 8D = 00)16: 01110010 + 10001100 = 00000011: Flags: C = 0, S = 0, Z = 1, V = 0

18. An 8-bit computer has a register R. Determine the values of status bits C, S, Z, and V as shown in the figure below after each of the following instructions. The initial value of register R is hexadecimal 72. The digits below are also in hexadecimal. Exclusive-OR R with R?.

- (72 ⊕ 72 = 01)16: 01110010 + 01100110 = 00010001: Flags: C = 1, S = 0, Z = 1, V = 0
- (72 ⊕ 72 = 00)16: 01110010 + 01100110 = 00000000: Flags: C = 0, S = 0, Z = 1, V = 0
- (72 ⊕ 72 = 10)16: 01110010 + 01100110 = 00100010: Flags: C = 0, S = 0, Z = 1, V = 1
- (72 ⊕ 72 = 11)16: 01110010 + 01100110 = 00110011: Flags: C = 0, S = 1, Z = 1, V = 0

19. The initial value of register R = 11001101 and assume that $SI_l = SI_r = 0$. Write the new register value after logical shift left instruction.



This figure will be used for questions 19 - 24

- R = 10011001
- R = 11011000
- R = 10011010
- R = 01001100

20. The initial value of register R = 11001101 and assume that $SI_l = SI_r = 0$. Write the new register value after logical shift right instruction.

- R = 01100110
- R = 00110110
- R = 01101110
- R = 01101101

21. The initial value of register R = 11001101 and assume that $SI_l = SI_r = 0$. Write the new register value after arithmetic shift left instruction.

- R = 10011000
- R = 10011001
- R = 00011010
- R = 10011010

22. The initial value of register R = 11001101 and assume that $SI_l = SI_r = 0$. Write the new register value after arithmetic shift right instruction.

- R = 11100110
- R = 11000110
- R = 11011100
- R = 10100110

23. The initial value of register R = 11001101 and assume that $SI_l = SI_r = 0$. Write the new register value after rotate shift left instruction.

- R = 10011001
- R = 10011010
- R = 10011011
- R = 11011011

24. The initial value of register R = 11001101 and assume that $SI_l = SI_r = 0$. Write the new register value after rotate shift right instruction.

- R = 11000110
- R = 11100110
- R = 11000111
- R = 10100110

25. Using conditional branch instruction evaluates X – Y, where X= 11110000 and Y= 00010100, using the following table, write down the flag status after the tested condition.

Mnemonic	Branch condition	Tested condition
BZ	Branch if zero	Z = 1
BNZ	Branch if not zero	Z = 0
BC	Branch if carry	C = 1
BNC	Branch if no carry	C = 0
BP	Branch if plus	S = 0
BM	Branch if minus	S = 1
BV	Branch if overflow	V = 1
BNV	Branch if no overflow	V = 0
Unsigned compare conditions (A - B)		
BHI	Branch if higher	A > B
BHE	Branch if higher or equal	A ≥ B
BLO	Branch if lower	A < B
BLOE	Branch if lower or equal	A ≤ B
BE	Branch if equal	A = B
BNE	Branch if not equal	A ≠ B
Signed compare conditions (A - B)		
BGT	Branch if greater than	A > B
BGE	Branch if greater or equal	A ≥ B
BLT	Branch if less than	A < B
BLE	Branch if less or equal	A ≤ B
BE	Branch if equal	A = B
BNE	Branch if not equal	A ≠ B

- X - Y = 11110000 + 11101101 = 11011101: Flags status C = 1, S = 0, V = 0, Z = 0
- X - Y = 11110000 + 11001100 = 11010110: Flags status C = 1, S = 1, V = 1, Z = 0
- X - Y = 11110000 + 11101100 = 11011100: Flags status C = 1, S = 1, V = 0, Z = 0
- X - Y = 11110000 + 11101011 = 11011101: Flags status C = 0, S = 1, V = 0, Z = 1