CompSci 150 Homework 2

Theo Koss

April 2024

1. (a) Similar: They both temporarily store data. They both can be accessed by the CPU, and they are both "volatile", meaning they lose their data when the computer's power is turned off.

   (b) Difference: They differ in capacity, and access speed. Registers - fast but limited in size, main memory larger capacity but slower. They also differ in their role, registers serve as storage for data being changed by the CPU, when a program is excecuted. Main memory cells serve as the main storage for programs.

2. (a) $0x2304$:

   $$0010\ 0011\ 0000\ 0100$$

   (b) Opcode of $0xB2A5$:

   $$B = 1011$$

   (c) Operand of $0xB2A5$:

   $$2A5 = 0010\ 1010\ 0101$$

3. Start: $0x98 = 152_{10}$ end: $0xA2 = 162_{10}$.
   There are 11 memory cells in this block. The addresses are counting

up in hex from $0x98$ to $0xA2$:

$$0x98$$
$$0x99$$
$$0x9A$$
$$0x9B$$
$$0x9C$$
$$0x9D$$
$$0x9E$$
$$0x9F$$
$$0xA0$$
$$0xA1$$
$$0xA2$$

4. Value of the program counter after $0xB0CD$ is the operand, CD.

6. (a) First, load each value from memory into its own register.
   (b) Add together the registers containing x and y (only 2 at a time).
   (c) Add the result to the register containing z.
   (d) Store the result in memory.

   For $(2x) + y$, load each value into its own register, multiply the register containing x by 2, then add the result to the register containing y. Then store the result in memory.

7. Translate to English:

   (a) $0x7123$: OR the bit patterns in registers 2 and 3, placing the result in register 1.
   (b) $0x40E1$: MOVE the bit pattern in register E to register 1.
   (c) $OxA304$: ROTATE the bit pattern in register 3 to the right, 4 times.
   (d) $OxB100$: JUMP to the instruction at address 00 if register 1 has the same bit pattern as register 0. Otherwise, continue.

(e) $0x2BCD$: LOAD register B with bit pattern CD.

9. Translate from English to Vole:

   (a) $0x2677$

   (b) $0x1677$

   (c) $0xBA24$

   (d) $0xA403$

   (e) $0x81E2$

11. Classify whether the instruction changes the contents of the memory cell at location $0x3C$, retrieves the contents of $0x3C$, or is independent of $0x3C$.

   (a) $0x353C$: Changes.

   (b) $0x253C$: Retrieves.

   (c) $0x153C$: Retrieves.

   (d) $0x3C3C$: Changes.

   (e) $0x403C$: Independent.

12. (a) Translate first instruction: $0x2655$ = LOAD register 6 with the bit pattern 55.

   (b) If PC started containing $0x00$, what bit patterns is in register $0x6$ when the machine halts?: Bit pattern: 55.

13. (a) First instruction: $0x1221$ = LOAD register 2 with the bit pattern found in 21.

   (b) First instruction starting with $0x01$: $0x2134$ = LOAD register 1 with bit pattern 34.

15. (a) When the machine halts, the bit pattern found in register A will be in address $0x00$.
   Register A contains: ADD bit patterns of register B and C. B contains: bit pattern 03. C also contains bit pattern 03. So A is the sum of $0x03$ and $0x03$.

   (b) Program counter has bit pattern $0x0A$.

18. Value in register $0x0$: $0010 + 0010 = 0100 = 4_{10}$.

23. Write a prograam in Vole to perform:

    (a) Move the value at memory location $0xD8$ to memory location $0xB3$.

Address	Contents
$0x00$	$0x11$
$0x01$	$0xD8$
$0x02$	$0x31$
$0x03$	$0xB3$
$0x04$	$0xC0$
$0x05$	$0x00$

    (b) Interchange the values stored at memory locations $0xD8$ and $0xB3$.

Address	Contents
$0x00$	$0x11$
$0x01$	$0xD8$
$0x02$	$0x12$
$0x03$	$0xB3$
$0x04$	$0x31$
$0x05$	$0xB3$
$0x06$	$0x32$
$0x07$	$0xD8$
$0x08$	$0xC0$
$0x09$	$0x00$

    (c) If the value stored in $0x44$ is $0x00$, then place the value $0x01$ in memory location $0x46$, otherwise put the value $0xFF$ in memory

location $0x46$.

Address	Contents	
$0x00$	$0x11$	
$0x01$	$0x44$	(Load reg. 1 with value of mem. loc. 0x44)
$0x02$	$0x20$	
$0x03$	$0x00$	(Load reg. 2 with bit pattern 0x00)
$0x04$	$0x23$	
$0x05$	$0x01$	(Load reg. 3 with bit pattern 0x01)
$0x06$	$0xB1$	
$0x07$	$0x0E$	

(Check if reg. 1 is equal to reg. 0, if true, jump to 0x0E. If false, continue.)

$0x08$	$0x24$	
$0x09$	$0xFF$	(Load reg. 4 with bit pattern 0xFF)
$0x0A$	$0x34$	
$0x0B$	$0x44$	(Store reg. 4 to mem. loc. 0x44)
$0x0C$	$0xC0$	
$0x0D$	$0x00$	(Halt)
$0x0E$	$0x33$	
$0x0F$	$0x44$	(Store reg. 3 to mem. loc. 0x44)
$0x10$	$0xC0$	
$0x11$	$0x00$	(Halt)

34. (a)

111001
AND 101001
=101001

c.

$$\begin{array}{r} 001110 \\ \text{AND } 010101 \\ =000100 \end{array}$$

i.

$$\begin{array}{r} 111001 \\ \text{XOR } 101001 \\ =010000 \end{array}$$

(P.S. A link between math and CS: "AND" is multiplication in the boolean ring $\mathbb{Z}/2\mathbb{Z}$, and "XOR" is addition (mod 2))

$$0 \cdot 1 = 0 \quad 1 \cdot 1 = 1 \quad 0 \cdot 0 = 0$$

$$0 + 1 = 1 \quad 1 + 1 = 0 \quad 0 + 0 = 0$$

and:

$$1 \cdot (1 + 0) = (1 \cdot 1) + (1 \cdot 0) = 1 + 0 = 1$$

(Multiplication distributes over +)

Corresponds to the truth tables for AND and XOR.

35. Identify mask and logical operation needed:

    (a) Put 1s in the upper 4 bits of an 8-bit pattern without disturbing the other bits:

    $$\text{Mask} = 11110000$$

    $$\text{Operation} = \text{OR}$$

    (b) Complement the most significant bit of an 8-bit pattern without changing the other bits:

    $$\text{Mask} = 10000000$$

    $$\text{Operation} = \text{XOR}$$

d. Put a 0 in the least significant bit of an 8-bit pattern without disturbing the other bits:

$$\text{Mask} = 11111110$$

$$\text{Operation} = \text{AND}$$

(Essentially, multiply every bit by 1, except least significant bit which gets multiplied by 0)