Name: Nikhil Wora

HW1: Representing Data and Instructions

The total value of this assignment is 50 points. There are 12 questions that are each worth 4 points, except for Question 12 that is worth 6 points. Homework assignments are to be complete individually. You are allowed to use whatever resources at your disposal with the exception that you cannot ask another human for the answer, e.g. posting on forums, chat, or messaging.

SECTION 0: NUMBER SYSTEMS

1. What is 0x0021 in decimal? What value does the 2 represent?

- 0x0021 in decimal is 33.
- The value 2 represents two 16s in hexadecimal.

2. How many bytes are represented by 4 digits in hexadecimal (e.g. 0xB00A)?

- The number of bytes represented by 4 digits in hexadecimal is 2 bytes.

3. What is 0x8C in binary? What value is C converted to decimal?

- 0x8C in binary is 10001100.
- 12 is the value that of C being converted to decimal.

SECTION 1: INSTRUCTIONS

The table with the instructions from the lab is provide on the last page of this document.

4. Identify the opcode and operand for the instruction: 0x105

   One-bit opcode and 2-bit operand is being used for 0x105. 0x1: Opcode and 0x05: Operand.

5. Explain what the instruction 0x105 does.
   - The instruction allows us to comprehend how operand and opcodes work together.
   - The first bit is the opcode and the next two bits are operands.
   - 0x1: Opcode: Add the contents of the memory address to the accumulator.
   - 0x05: Operand
   - 0x105 adds the contents of memory location 0x05 to the accumulator.

6. How would we write this using mnemonics?

   In mnemonics, we would write this as ADD 0x05 with operand.

SECTION 2: UNDERSTANDING CODE

The following section will be asking questions about the code in the following table. The Instruction Set is defined in the table on the last page of this assignment. The Address column represents the Memory Address. The Contents column represents the contents of the memory at the corresponding address. The Mnemonic column provides the "human readable" format of the program. The comments column is there for your notes if you want them.

Address	Contents	Mnemonic	Comments
0x00	0x901	INP	N/A This section is for note taking.
0x01	0x310	STA 0x10	
0x02	0x901	INP	
0x03	0x311	STA 0x11	
0x04	0x513	LDA 0x13	
0x05	0x110	ADD 0x10	
0x06	0x313	STA 0x13	
0x07	0x511	LDA 0x11	
0x08	0x212	SUB 0x12	
0x09	0x311	STA 0x11	
0x0A	0x804	BRP 0x04	
0x0B	0x513	LDA 0x13	
0x0C	0x210	SUB 0x10	
0x0D	0x313	STA 0x13	
0x0E	0x902	OUT	
0x0F	0x000	HLT	
0x10	0x000	DAT 0x00	
0x11	0x000	DAT 0x00	
0x12	0x001	DAT 0x01	
0x13	0x000	DAT 0x00	

7. Fill-in the missing entries at address 0x05, 0x08, 0x0A, 0x0C: DID THAT ABOVE
8. What addresses in this program act like variables?

The addresses in the program will only act like variables if they contain the specific address and value. The variables in this case are 0x10, 0x11, 0x12, and 0x13 as they are all data contents.

9. This program calculates the product of the two inputs. Describe the steps this program performs to calculate the returned value.

The program takes two user inputs and stores each one in a memory address which then adds the first number to itself in a loop. This leads to each time the number is added back to itself, then one is subtracted from the second number. This will allow the program to continue until the second number hits 0 which then will provide us with a product.

SECTION 3: DEFINING A NEW INSTRUCTION

As seen in Section 2, multiplication requires a good amount of code. Imagine you are working with the hardware engineers to make the next version of the CPU and are adding multiplication to the instruction set. The only requirement is that you do not remove an already existing instruction, so you are limited to replacing an instruction that has no operation defined.

10. Complete the entry for your new multiplication instruction.: DID THAT BELOW

Instruction	Mnemonic	Opcode	Operand	Description
Multiply	MLA	0xA	0xXX	This will multiply all the contents of the memory address.

11. Incorporate your new instruction into the program below to make it calculate the product of two inputs and output the result.: DID THAT BELOW

Address	Contents	Mnemonic	Comments
0x00	0x901	INP	N/A This section is for note taking.
0x01	0x306	STA 0x06	
0x02	0x901	INP	
0x03	0xA06	MLA 0x06	
0x04	0x902	OUT	
0x05	0x000	HLT	
0x06	0x000	DAT 0x00	

12. (6pts) Explain how this program is executed by the CPU using the instruction cycle.

The program is executed by the CPU using the instruction cycle by taking two inputs and multiplying them and outputting the product. This reveals that the CPU will fetch the instruction, then decode the instruction so it can read the address from the memory. This will allow us to execute the problem to obtain an output.

INSTRUCTION SET

This is modified version is based on the work by Dr. Stuart Madnick ("Little Man Computer."). The opcode is 1 hexadecimal digit; the operand is 2 hexadecimal digits.

Instruction	Mnemonic	Opcode	Operand	Description
Halt	HLT	0x0	0x00	Stop running
Add	ADD	0x1	0xXX	Add the contents of the memory address to the Accumulator
Subtract	SUB	0x2	0xXX	Subtract the contents of the memory address from the Accumulator
Store	STA	0x3	0xXX	Store the contents in the Accumulator in the memory address
Load	LDA	0x5	0xXX	Load Accumulator with contents of memory address
No Operation	NOP	0x4	0x00	No operation defined
Branch always	BRA	0x6	0xXX	Set the PC to the value in the operand
Branch if zero	BRZ	0x7	0xXX	Set the PC to the operand if the Accumulator is zero
Branch if zero or positive	BRP	0x8	0xXX	Set the PC to the operand if the contents of the Accumulator is zero or positive
Input	INP	0x9	0x01	Retrieve user input and store in the Accumulator
Output	OUT	0x9	0x02	Output the contents of the Accumulator
No Operation	NOP	0xA	0x00	No operation defined
No Operation	NOP	0xB	0x00	No operation defined
No Operation	NOP	0xC	0x00	No operation defined
No Operation	NOP	0xD	0x00	No operation defined
No Operation	NOP	0xE	0x00	No operation defined
No Operation	NOP	0xF	0x00	No operation defined
Data storage	DAT			Label for a memory address; also can have a contents specified

XX refers to a memory address.

"Little Man Computer." In *Wikipedia*, January 27, 2020. https://en.wikipedia.org/w/index.php?title=Little_man_computer&oldid=937810035.