CSE 321a: Computer Organization (I)
Third Year, Computer & Systems Engineering

Assignment #3

Due date: **Thursday, December 21st, 2017**

1. Consider an instruction set architecture with the following characteristics:
   - Memory: byte-addressable, 8-bit addresses.
   - Numbers: unsigned or signed (sign-and-magnitude).
   - Registers (8-bit): PC, SP and four general-purpose registers (R0, R1, R2 and R3).
   - Instructions (variable-length): one-operand or two-operand.
     - One-operand instructions (16-bit): 6-bit opcode followed by one 10-bit operand.
     - Two-operand instructions (24-bit): 4-bit opcode followed by two 10-bit operands.
     - Operand (10-bit): 2-bit addressing mode (AM) followed by 8-bit value/address (VA).
     - Addressing-mode (2-bit): 00 → immediate, 01 → register, 10 → indexed, 11 → PC-relative (where register number is encoded by the least significant two bits of VA whenever is needed).
     - Some of the supported operations:

Operation	Mnemonic	Opcode (Decimal)	Description
Move	MOV	10	First operand is destination. Second operand is source.
Arithmetic Shift left	ASL	11	First operand specifies value to be shifted. Second operand specifies number of bit positions.
Branch If Negative	BRN	13	First operand specifies value to be tested. Second operand specifies target address using PC-relative addressing mode.
Branch	BR	59	Operand specifies target address using PC-relative addressing mode.
Decrement	DEC	61	Operand specifies value to be decremented.

   (a) Translate the following C-language snippet to the assembly language of this instruction set:

```
for(i=2; i>=0; i--)
  x[i] = 8 * x[i]; /* x is array of single-byte signed integers */
```

   Use the symbol **x** to represent the address of the first element of the array and register R1 to represent the variable **i**. Your assembly program must start with the following instruction: "**MOV R1,#2**".

   (b) Assume your assembly program (from part 1) was compiled to machine language and stored in memory starting at location 5B. Fill up the following table with the rest of the machine program assuming that array **x** was stored in memory starting at location 2E:

Address (Hexadecimal)	Contents (Hexadecimal)
5B	A4
5C	04
5D	02
5E	...
…	...

   (c) Show, using the table below, the execution trace of the machine program (from part 2) by filling in the contents of every register and memory location **after the execution** of every instruction.

All values are in hexadecimal.

Instruction	PC	R1	Memory Locations		
			2E	2F	30
Initially	5B	C5	0F	85	03
A40402	5E	02	0F	85	03
...	...	...	...	...	...

2. A CPU implements a stack-based (*i.e.*, zero-address) instruction set. Every register in that CPU has a dedicated connection with every other register and every ALU input. Write in your answer sheet the sequence of micro-operations (and show the steps) required for that CPU to execute the following instructions:
   (a) **`NEGATE`**
   (b) **`EXCHANGE`** /* ***Hint****: Assume CPU has special-purpose registers for holding temporary data* */
   (c) **`PUSH 100(R1)+`**

3. A pipelined processor has the following features:
   - Four pipeline stages: fetch instruction (FI), decode instruction (DI), calculate and fetch memory operands (FO), and execute instruction and store memory operands (EI). Register operands are accessed (for read/write) by EI stage.
   - Memory ports: two read ports and one write port.
   - Branch history table: two history bits in each entry. Initial prediction is "Weakly Not-Taken".

   Suppose the processor executes the following loop 100 times:

```
LOOP: INC R1   /* R1 ← [R1] + 1 */
      JNZ (R1), LOOP   /* Jump to LOOP if [[R1]] ≠ 0 */
```

   (a) Draw a timing diagram to show instruction pipelining during the first three iterations of the loop. ***Hint:*** *The processor executes the first three iterations of the loop in 15 clock cycles.*
   (b) How many clock cycles are needed to execute this program (*i.e.*, the 100 iterations of the loop)?