6 pages **CMPT 295: Test 1A SOLUTION** **Febreary 5, 2016** **30 marks**

Answer all questions on the test paper. Use the backs of the pages for rough work, if necessary. Be sure your name and student number are on all pages. A VHDL summary sheet is provided and can be detached from this paper. No calculators, cell phones, or laptops are to be used.

CAUTION: In accordance with the Academic Honesty Policy (S10.01), academic dishonesty in any form will not be tolerated.

1. Suppose you are given the binary sequence 01110010110110. Determine what the sequence represents if it is decoded according to each of the following encoding schemes:

   (a) Natural binary. (i.e., as an unsigned base-2 number. Express your answer as a sum of powers of 2.) **(1 mark)**

   $$01110010110110 \quad = \quad 2^1 + 2^2 + 2^4 + 2^5 + 2^7 + 2^{10} + 2^{11} + 2^{12}$$

   (b) Hexadecimal (i.e., each digit is encoded using the hexadecimal encoding scheme.) **(1 mark)**

   $$01110010110110 \quad = \quad 1CB6$$

   (c) 7-bit ASCII. (i.e., each digit is encoded using the 7-bit ASCII encoding scheme. The character code for a digit is obtained by prefixing the BCD codeword for the digit with 011. **(1 mark)**

   01110010110110 = ”96”

2. When writing programs in a typical programming language, the programmer must be aware of the maximum value that can be stored in a variable. The “largest value” usually depends on the size of a word of memory and on the encoding scheme adopted for representing integers.

   Complete the following table indicating the value of the largest value that can be represented using the encoding scheme given and assuming the word size as indicated: **(6 marks)**

encoding scheme	word size 8-bits	10 bits
unsigned base-2	$2^8 - 1 = 255$	$2^{10} - 1 = 1023$
BCD	99	399
floating point	m=3 $\quad 1.111 \times 2^7$	m=4 $\quad 1.1111 \times 2^{15}$

   NOTE: You may leave your answers expressed as powers of 2 or 10 (as appropriate). For example, $2^8 + 1$ can be left “as is” or evaluated to “257”.

3. Express the number -56.625 as indicated:

   (a) As a base-2 number. **(2 marks)**

$$\begin{aligned} -56 &= -111000 \\ .625 &= .101 \\ -56.625 &= -111000.101 \end{aligned}$$

   (b) As a hexadecimal number. **(2 marks)**

$$\begin{aligned} -56.625 &= = -111000.1010 \\ &= -38.A \end{aligned}$$

   (c) In 12 bit floating point format with a fraction field of 5 bits **(2 marks)**

$$\begin{aligned} -56.625 &= -111000.101 \\ &= -1.11000101 \times 2^5 \\ &= 110010011000 \text{ or } 110010011001 \end{aligned}$$

.

4. Evaluate each of the following C expressions, giving the resulting values in hexadecimal:

   (a) `0xabcd & 0x0ff0 << 8` **(1 mark)**

   CORRECT PRECEDENCE:

$$\begin{aligned} 0xabcd\&0x0ff0 << 8 &= 1010101111001101\&(0000111111110000 << 8) \\ &= 1010101111001101\&(1111000000000000 \\ &= 1010000000000000 \\ &= 0xA000 \end{aligned}$$

   INCORRECT PRECEDENCE BUT ACCEPTED:

$$\begin{aligned} 0xabcd\&0x0ff0 << 8 &= (1010101111001101\&(0000111111110000) << 8 \\ &= 0000101111000000 << 8 \\ &= 101111000000000000000 \\ &= 0xBC000 \end{aligned}$$

   (b) `0x12034) | (0x5 << 20)` **(1 mark)**

$$\begin{aligned} 0x12034|(0x5 << 20) &= 000000010010000000110100|(000000000000000000000101 << 20) \\ &= 000000010010000000110100|010100000000000000000000 \\ &= 010100010010000000110100 \\ &= 0x512034 \end{aligned}$$

(c) `(0xff >> 3) << 3` **(1 mark)**

$$\begin{aligned} (0xff >> 3) << 3 &= (11111111 >> 3) << 3 \\ &= (00011111) << 3 \\ &= 11111000 \\ &= 0xf8 \end{aligned}$$

(d) `(0xab00 | (0xff << 4) ) & 0xa0` **(1 mark)**

$$\begin{aligned} &(0xab00|(0xff << 4))\&0xa0 \\ &\quad = (1010101100000000|(11111111 << 4))\&10100000 \\ &\quad = (1010101100000000|(11110000))\&10100000 \\ &\quad = 1010101111110000\&10100000 \\ &\quad = 0000000010100000 \\ &\quad = 0xA0 \end{aligned}$$

5. **Instruction Set Architecture** A proposed instruction set architecture consists of 60 instructions. The CPU will incorporate an internal memory of 8 registers, R[0] , ..., R[7], and employ a 2-operand machine model. The CPU for the instruction set will be interfaced with a $2^{20} \times 16$ external memory unit.

The syntax and semantics for a set of four instructions are as follows:

opcode	syntax		semantics
1A	ADD	Ri, Rj	$R[i] \leftarrow R[i] + R[j]$; $i, j \in \{0,...,7\}$
1B	SUB	Ri, Rj	$R[i] \leftarrow R[i] - R[j]$; $i, j \in \{0,...,7\}$
2A	LD	R1, addr	$R1 \leftarrow M[addr]$
2B	STO	R1, addr	$M[addr] \leftarrow R1$

(a) Identify how many bits are required to represent each of the following components in the machine instructions:

i. The opcode **(1 mark)**

ANSWER: 6 bits

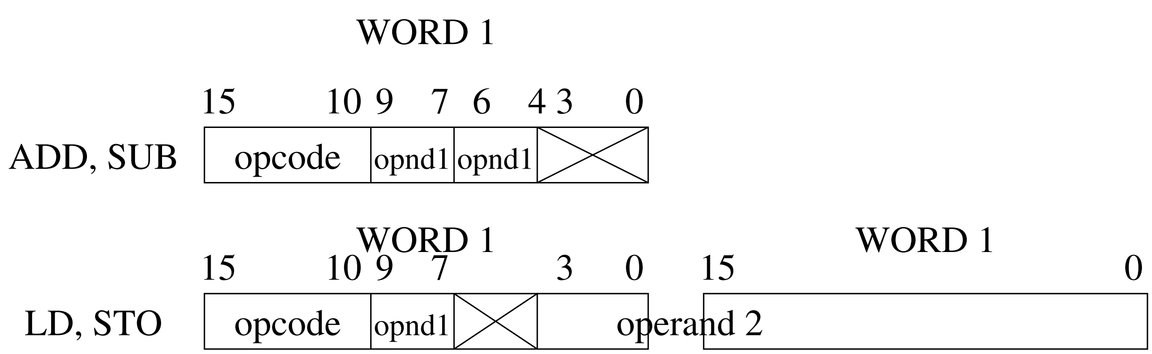
ii. The "addr" operand: **(1 mark)**

ANSWER: 20 bits

iii. The "Ri" operand: **(1 mark)**

ANSWER: 3 bits

(b) Provide physical formats indicating the location of each operand and how many words of memory would be needed to store each instruction in memory. **(4 marks)**



(c) The following program segment is written using the instructions provided above:

```
LD    R1, A
LD    R2, B
ADD   R1, R2
ADD   R1, R1
STO   R1, SUM
```

(d) What assignment statement is evaluated by this program? **(1 mark)**

ANSWER: SUM = 2*(A+B)

(e) Express in binary the following assembly language instructions. The value of `A` is stored at address `A0010` (hexadecimal). **(6 marks)**

i. `LD R1, A`

$2A_{16}\&01_2\&A0010_{16}$ =101010 001 1010 0000 0000 0001 0000_2

ii. `ADD R1, R2`

$1A_{16}\&001_2\&010_2\&0000_2$ =011010 001 010 0000_2

(f) How many memory accesses are required to both fetch the instructions and execute them in order to execute the program segment. **(2 marks)**

Each LD and STO instruction (3) occupies 2 words and makes one external memory access during execution.

Each ADD instruction (2) occupies 1 word and makes no external memory accesses during execution.

Total number of fetches (3*2 + 2*1 = 8) plus total number of execute accesses (3) = 11 memory accesses.