

## CS 250 Spring 2017 Homework 04

Due 11:58pm Wednesday, February 08, 2017

Submit your typewritten file in PDF format to Blackboard.

1. Complete the table to show how the given binary strings are interpreted in each data representation. For numerical representations write your answer in the form of a decimal number. Use care when writing a decimal number equivalent to show a sign when there is ambiguity if a sign is not shown. If a binary string is not valid for a given representation, write “error” in the table.

Given binary string	Packed BCD	Binary signed digit
00000000	0000	0
00010001	0001 0111	5
0x81 (Converted to binary and worked from there. If that is not allowed than this is ‘error’ as it is not a valid binary string, it is hex)	0001 0010 1001	1
0xFF (Same as above)	0010 0101 0101	1

2. Complete the table to show how the given binary strings are interpreted in IEEE 754 (Standard for Floating Point Arithmetic). Lecture 6-7 v5 Slide 56 will be quite helpful in answering this question and Question 3, next.
- Scientific notation in base 2 is the form  $\pm 1.b \dots b \times 2^n$  where  $b$  denotes a non-zero bit unless all the bits in the fractional weighted positions are zero in which case a single  $b$  bit equal to zero should be shown, and where  $n$  denotes a sign-magnitude decimal integer.
- Base 10 integers have the form  $\pm d \dots d$  where  $d$  represents a significant decimal digit.

Given binary string	Value when interpreted per IEEE 754 and written in base 2 scientific form	Value when further converted to base 10 integer; if this conversion is not valid then write “error”
1 10000001 11000000000000000000000000000000	$-1^1 * 1.11 * 2^2$	-7
0x81000000 (Converted to decimal and worked from there. If that is not allowed	$-1^0 * 1.0000001 * 2^{31}$	2164260864



6. Design a simple, single instruction format, in the style shown in Figure 5.1, and machine language for a computer with the following characteristics. Use the table below as a framework for your answer.

The instructions for the ISA of this computer include only the Boolean operations And, Or, Not, and Exclusive Or. An instruction bit string specifies the location of each operand to be read from memory by holding a pointer to the desired memory location in a designated operand field. The same strategy is used to specify the location for the result to be written. Each operand and the result for each instruction is an 8-bit binary string. The computer memory has  $2^{10}$  storage locations. Each memory location holds a one-byte bit string.

Using Figure 5.1 as a guide, define this four-instruction ISA by creating a tabular diagram with four rows, one for each of the instructions. Clearly label the fields of your instruction format, and let the leftmost field be the Operation Code, or Opcode, field, followed by all operand fields, and ending with the result field. If a field is not needed, give that field the name "Unused." Next, using the minimum possible number of bits sufficient for each field to perform its function, specify how long each field in your instruction format is. Finally, for any field in the format for which the bit string is must be known before writing any programs for this computer, use your Designer's Prerogative to choose the bit string that will fill that field and display your choice in your format for each instruction.

Answer:

Operation	Opcode (2 bits)	Operand1	Operand2	Total Bits
AND	00	1 bit	1 bit	4 bits
OR	01	1 bit	1 bit	4 bits
NOT	10	1 bit	Unused	4 bits
XOR	11	1 bit	1 bit	4 bits

*The result of each operation is also 1 bit.*

In this system (using the minimum possible number of bits sufficient for each field), each operation would be a bitwise (bit by bit) operation. The Opcode + Operand1 + Operand2 = Total Bits which is 2 bits + 1 bit + 1 bit = 4 bits. The hardware ignores the single unused bit in the Operand2 field of the NOT Operation.

7. If a computer can add, subtract, multiply, and divide 16-bit integers, 32-bit integers, 32-bit floating-point values, and 64-bit floating-point values, how many unique opcodes will be needed? Hint: an opcode defines not only the operation to be performed but also the data types of its operand(s) and result.

4 operations x 4 data types = 16 opcodes