

CS 250 Spring 2017 Homework 04

Due 11:58pm Wednesday, February 08, 2017

Submit your typewritten file in PDF format to Blackboard.

- 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	0	0
00010001	9	5
0x81	81	1
0xFF	error	-15

- 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 110000000000000000000000	-1.11×2^2	-0.7×10^1
0x81000000	-1.0×2^{-125}	-0.1×10^{-126}
0xFFFFFFFF	NaN	error

1100000000

- Complete the table to show in hexadecimal notation, the IEEE 754 representation of the given numbers and symbols.

Given value	IEEE 754 representation in 0x notation
-2.0 (base 10)	0x80400000
768 (base 10)	0x600000
$+\infty$	0x3FC00000

- Using congruence modulo 2^{10} , a circuit maps 32-bit unsigned integer addresses for a main memory onto the addresses of a smaller memory having only 2^{10} locations.

- a. How many addresses are mapped onto each of the locations in the small memory? Express your answer in both exponential form and in weighted positional notation that incorporates the appropriate suffix from the set {kilo-, mega-, giga-, tera-, exa-, and peta-} to yield the simplest result.

4,194,304

4 Megabit, 194 Kilobit, 304 bits

- b. Draw a schematic of a circuit that takes as input a single 32-bit address, performs the congruence mapping computation, and then outputs the intended address for the smaller memory. Label the input address $A_{31}A_{30} \dots A_1A_0$. Label the output address bits $B_iB_{i-1} \dots B_1B_0$.
5. Over time processor circuits have become faster than memory circuits. What aspect of the Harvard Architecture gives it an advantage in comparison with the Von Neumann Architecture for a computer processor designer trying use parallelism to speed up the Fetch-Execute Cycle?

The main advantage of the Harvard architecture versus the Von Neumann architecture is that the Harvard architecture allows for program memory and general memory to be fetched at the same time. With this a designer doesn't have to spend time waiting for the processor to finish fetching some data before continuing the application flow. Instead of waiting, the processor can execute application data while waiting for the general data to be fetched.

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			
AND	00	Input A Pointer 8-bit	Input B Pointer 8-bit	Output Pointer 8-bit

OR	01	Input A Pointer 8-bit	Input B Pointer 8-bit	Output Pointer 8-bit
NOT	10	Input Pointer 8-bit	Unused	Output Pointer 8-bit
XOR	11	Input A Pointer 8-bit	Input B Pointer 8-bit	Output Pointer 8-bit

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.

16

Each value (16bit int, 32bit fp, etc) has its own set of opcodes. Each set has four opcodes, being the add, subtract, multiply, and divide operations. 4 sets of 4 opcodes would be 16 opcodes.