Course Name: _Comp Arch Lab_

Course Number and Section: 14:332:333:xx

Experiment: Experiment # 1 — Introduction and Number Representation

Lab Instructor: Dian Jin

Date Performed: September 15, 2021

Date Submitted: September 27, 2021

Submitted by: Qadis Chaudhry, 202001055

Lab 1
Introduction and Number Representation

Qadis Chaudhry

September 29, 2021

1 Unsigned Integers

Conversions

a. Convert the following from their initial radix to the other two common radices:

0b011000101:

$2^7 + 2^6 + 2^2 + 2^0 = \boxed{(198)_{10}}$

$(1100)_2(0101)_2 = (C)_{16}(5)_{16} = \boxed{(C5)_{16}}$

0xF54E:

$15 \times 16^3 + 5 \times 16^2 + 4 \times 16 + 14 = \boxed{(62798)_{10}}$

$(F)_{16}(5)_{16}(4)_{16}(E)_{16} = (1111)_2(0101)_2(0100)_2(1110)_2 = \boxed{(1111010101001110)_2}$

312:

$$312 \div 2 = 156 \ R0$$

$$156 \div 2 = 78 \ R0$$

$$78 \div 2 = 39 \ R1$$

$$39 \div 2 = 19 \ R1 \qquad\qquad 312 \div 16 = 19 \ R8$$

$$19 \div 2 = 9 \ R1 \qquad\qquad 19 \div 16 = 1 \ R3$$

$$9 \div 2 = 4 \ R1 \qquad\qquad 1 \div 16 = 0 \ R1$$

$$4 \div 2 = 2 \ R0 \qquad\qquad \boxed{(138)_{16}}$$

$$2 \div 2 = 1 \ R0$$

$$1 \div 2 = 0 \ R1$$

$$\boxed{(100111100)_2}$$

0b10110111:

$$2^7 + 2^5 + 2^4 + 2^2 + 2^1 + 2^0 = \boxed{(183)_{10}}$$

$$(1011)_2(0111)_2 = \boxed{(B7)_{16}}$$

0x6BCC:

$$6 \times 16^3 + 11 \times 16^2 + 12 \times 16 + 12 = \boxed{(27596)_{10}}$$

$$(6)_{16}(B)_{16}(C)_{16}(C)_{16} = (0110)_2(1011)_2(1100)_2(1100)_2 = \boxed{(110101111001100)_2}$$

87:

$$87 \div 2 = 43 \ R1$$

$$43 \div 2 = 21 \ R1$$

$$21 \div 2 = 10 \ R1$$

$$87 \div 16 = 5 \ R7$$

$$10 \div 2 = 5 \ R0$$

$$5 \div 16 = 0 \ R5$$

$$5 \div 2 = 2 \ R1$$

$$2 \div 2 = 1 \ R0$$

$$\boxed{(57)_{16}}$$

$$1 \div 2 = 0 \ R1$$

$$\boxed{(1010111)_2}$$

b. Write the following using IEC prefixes:

2^{29} : $\boxed{512 \text{ Mi}}$, 2^{54} : $\boxed{16 \text{ Pi}}$, 2^{16} : $\boxed{64 \text{ Ki}}$, 2^{65} : $\boxed{32 \text{ Ei}}$, 2^{33} : $\boxed{8 \text{ Gi}}$, 2^{42} : $\boxed{4 \text{ Ti}}$

c. Write the following using SI prefixes:

10^5 : $\boxed{100 \text{ K}}$, 10^{16} : $\boxed{10 \text{ P}}$, 10^{10} : $\boxed{10 \text{ G}}$, 10^{19} : $\boxed{10 \text{ E}}$, 10^{27} : $\boxed{1000 \text{ Y}}$, 10^4 : $\boxed{10 \text{ K}}$

d. Write the following with powers of 10:

21 Z : $\boxed{21 \times 10^{21}}$, 9 Y : $\boxed{9 \times 10^{24}}$, 13 K : $\boxed{13 \times 10^3}$

e. Write the following with powers of 2:

14 Mi : $\boxed{14 \times 2^{20}}$, 12 Ei : $\boxed{12 \times 2^{60}}$, 27 Ki : $\boxed{27 \times 10^{10}}$

2 Signed Integers

Two's Complement Exercises

1. What is the largest integer that can be represented with 16 bits? How many bits do you need to represent the largest integer plus 1?

With Two's Complement, the range is from -2^{16-1} to $2^{16-1} - 1$, which is from -32,768 to 32,767. This makes the largest number able to be represented by Two's Complement, 32,767. To represent the largest integer plus 1, you would need one more bit as this adds to the overall range making the new range from, -65536 to 65535.

With unsigned numbers, the largest number that can be represented is $2^{16} - 1 = 65535$. To represent the largest number plus one, once again, you would need 17 bits.

2. How do you represent the numbers 100, 6, and -5 (assume the numbers are 8 bits)?

100:

For positive integers, unsigned and Two's Complement are exactly the same, so for 100, using the division technique,

$$100 \div 2 = 50 \ R0$$

$$50 \div 2 = 25 \ R0$$

$$25 \div 2 = 12 \ R1$$

$$12 \div 2 = 6 \ R0$$

$$6 \div 2 = 3 \ R0$$

$$3 \div 2 = 1 \ R1$$

$$1 \div 2 = 0 \ R1$$

The number $(100)_{10}$ in 8 bit unsigned and Two's Complement notation would be $\boxed{01100100.}$

6:

Using the same technique as before and the same principle,

$$6 \div 2 = 3 \ R0$$

$$3 \div 2 = 1 \ R1$$

$$1 \div 2 = 0 \ R1$$

the 8 bit representation for the decimal number 6 in Two's Complement as well as unsigned integers, would be $\boxed{00000110.}$

-5:

Since this number is negative, it is signed and therefore cannot be represented using the unsigned integers. In this case, the number must be converted to binary and then using Two's Complement, made negative.

$$5 \div 2 = 2 \ R1$$

$$2 \div 2 = 1 \ R0$$

$$1 \div 2 = 0 \ R1$$

Taking this number in 8 bits, $(00000101)_2$ and converting it to negative using Two's Complement, first we flip all the bits yielding, $(11111010)_2$ and then we add 1 to the least significant bit, 11111011.

3. How do you represent 264 and -264 (assume the numbers are 8 bits)?

Finding the range for the maximum possible integer for 8 bit numbers, for unsigned we find that it is from 0 to $2^8 - 1 = 255$, and for Two's Complement it ranges from -2^{8-1} to $2^{8-1} - 1$ which is from -128 to 127. In both of these cases, it can be seen that the value of 264 resides outside of the range for 8 bit numbers and therefore cannot be represented using either notation.

4. What is the range of decimal (presented with powers of 2) that 32-bit Two's Complement can represent? Please explain why.

The range of numbers would be from -2^{32-1} to $2^{32-1} - 1$, which would result in a range from -2147483648 to 2147483647. Represented using powers of 2 this would be, from -2 Gi to 2 Gi - 1. This happens to be the range since Two's Complement has the ability to represent both positive and negative numbers and adding two opposite numbers together would have to result in zero. For this reason, since there are only a limited number of bits, in this case 32, the range will be half that of unsigned 32-bit integers. The maximum unsigned 32-bit integer can be represented as 2^{32} which is 4294967296. This number is cut in half, and the negative integers can be represented up to -2147483648, and the maximum positive integer will be 2147483647 since we have to account for zero.

5. Why do computers use bytes for address representation?

Using bits would likely be too costly to be beneficial. Bytes are used for addressing as this allows for more addresses to be defined with a large number of bits.

6. How much data can we store on a 8 K-byte memory chip?

8 bytes will be equal to 8000 bytes, and since there are 8 bits in one byte, there are 64000 bits that can be stored on the chip. Since each of the registers in the chip can hold 32 bits, this leaves 2000 addresses in which information can be stored.

3 Counting

1. How many bits do we need to represent a variable that can only take on the values 10, 20, 30, or 40?

Since the variable can only have four possible values, 2^2, 2 bits would be required to cover the four states.

2. If the only value a variable can take on is 98.65, how many bits are needed to represent it?

Since there is only one value for this variable to take on the lowest possible number of bits would be 1 bit since it can hold two values and we cannot go lower than that.

3. If we need to address 4 Mi-Byte of memory and we want to address every byte of memory, how long does an address need to be?

4 Mi-Bytes would be equal to $4 * 2^{20}$ Bytes, which equals 2^{22} Bytes. Using the equation $\log_2(n)$ which gives the number of bits needed to address some n number of bytes, we get that 22 bits are needed for each address.