ECE 281

Lesson 1 Notes

Objectives:

- Be able to represent an unsigned (positive) number in any number base, 2 to N, but particularly in decimal, binary and hexadecimal
- Know the limitations of a number system (e.g. range)

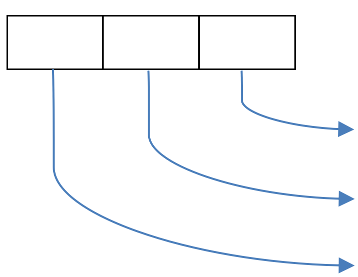
Lesson Notes:

Numerical Bases: As a review from CS 210, we will be discussing number bases. In CS 210 we primarily dealt with decimal (base-10), binary (base-2) or hexadecimal (base-16). Today we will add one more numerical base to the mix… octal (base-8). Numerical bases aren't all that complex once you understand what is really going on here.

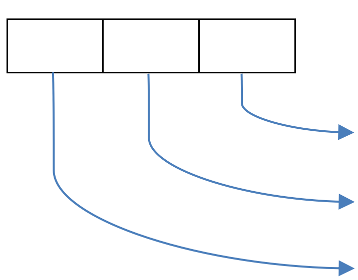
- **Nomenclature:**
- If we are trying to write a number with a specific base, we will generally put the number followed by a subscript indicating the base we are using.
- **Ex)** If I want to write 104 as a base-10 or decimal number, I would put:
    -
- **Ex)** If I want to write 104 as a base-8 or octal number, I would put:
    -

- **Conversions between bases:** the computer really only works with the binary system, but we will often work with each of the bases mentioned above for convenience sake. For that reason, you need to understand how to easily transition between bases. So lets now explore what is going on with each of the digits of a number.
- In order to understand what a number is really saying, we must understand the significance of essentially three separate things. You learned this at a young age for decimal and now accomplish it without even thinking for base-10 numbers. However, lets dig a little deeper for the other important bases for this class:8887315396

Ex) Lets again consider the number 104 as a familiar base-10 number, and break it down digit by digit:



- **Ex)** Now lets do the same thing, but consider the number 104 as an octal or base-8 number, and break it down digit by digit:



Decimal (base 10)	Binary (base 2)	Octal (base 8)	Hexadecimal (base 16)
00	0000	00	0
01	0001	01	1
02	0010	02	2
03	0011	03	3
04	0100	04	4
05	0101	05	5
06	0110	06	6
07	0111	07	7
08	1000	10	8
09	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

- Let's now work an example where we go from decimal to binary. Once you are in binary, it is very easy to transition to octal or hexadecimal. If I don't have a computer available, the easiest technique I know of is the divide by two method.

- **Ex)** Use the divide by two method to convert 104_{10} to binary:

- **Ex)** Now use the chart above to convert your answer to Octal and Hexadecimal as well (Get in the habit of using proper notation):

Range: today we are only going to talk about unsigned (i.e. zero or positive) numbers. Next lesson we will talk about signed integers such as 2's complement. For the various number bases, each digit can take on a value from 0 -> N-1 where N is the base:

Ex) what is the range for each digit in the following numbering systems:

Binary – base 2:

Octal – base 8:

Decimal – base 10:

Hexadecimal – base 16:

Based on what we learned previously, about the power associated with each digits position, we can deduce that the range of any N bit number is as follows:

Ex) 5 digit decimal

4 digit octal

6 digit hexadecimal

8 digit binary

Summary: a numbering system can use any number for a base from 2 to N. The most common bases for our applications will be 10 (decimal), 2 (binary), 16 (hexadecimal) and occasionally 8 (octal). However, any base greater than or equal to 2 is a valid base.

Additional Notes: