Reading 2

2.2 Unsigned binary numbers

- A number in base 10 is called a decimal number.
- A number is base 2 is called a binary number.
- Each digit in a binary number is called a bit.

- An unsigned binary number can only represent non-negative values.

- Counting up in binary:
  - ↳ Increment Rightmost bit
  - ↳ if its already 1, reset to 0 and carry
  - ↳ keep going

- Decimal to binary Example (4 bit binary numbers)

5 to binary → Starting from left:

1 _ _ _	8 is too much
0 1 _ _	4 is less
0 1 1 _	6 is too much
0 1 0 1	4+1=5 ; is equal

- Adding binary numbers is identical to adding decimal by hand.

Ex 0111 + 0110

$$\begin{array}{r} {\scriptstyle 1\ 1\ \ \ } \\ 0111 \\ +\,0110 \\ \hline 1101 \end{array}$$

1+0 = 1

1+1 → $\begin{array}{r} {\scriptstyle 1\ \ } \\ +\,1 \\ \hline 10 \end{array}$

1+1+1 = $\begin{array}{r} {\scriptstyle 1\ 1\ } \\ +\,1 \\ \hline 11 \end{array}$

- Overflow occurs when the result of a binary operation is too large to fit in allowed number of bits.
  - ↳ Ex 1111 + 0001 = 10000

2.3 Binary, hexadecimal, and octal

- hexadecimal (or hex) means base 16 number.
  - ↳ 16 symbols are needed: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

corresponding decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

- When converting binary to hexadecimal use groups of 4 bits starting from the rightmost group of 4 (use table for conversions)

- Hex to Decimal
  - multiply each digits decimal value by the digit's weight and sum
  - Ex $A7F = 10 \times 16^2 + 7 \times 16^1 + 15 \times 16^0 = 2687$

- **Decimal to Hex**
  - find the highest hex digit where a 1 doesn't exceed the decimal value, incrementing as much as possible without exceeding the decimal value, and repeating for lower digits.

Ex 20 decimal to Hex

↳ $1 \times 16^2 = 256$ is too much

↳ $1 \times 16^1 = 16$ is OK.

↳ $2 \times 16^1 = 32$ is too much. 4 remains.

↳ $4 \times 16^0 = 4$

↳ so **14**

Ex 40 decimal to Hex

$40/16 = 2$ rem 8 ← least significant hex digit

$2/8 = 0$ rem 2 ← next hex digit

(28)

Ex 258 decimal to Hex

$258/16 = 16$ rem 2

$16/16 = 1$ rem 0

$1/16 = 0$ rem 1

(102)

- **Octal** means a base 8 number.

2.4 ASCII and UniCode

- A character is a letter (a, b, ... z, A, B, ..., Z), symbol (!, @, #, ...), or a single-digit number (0, 1, ..., 9)

- ASCII is a popular code for characters.

- 7 bits for a character
- these 7 bits are stored as one "chunk" of memory

2.5 Signed numbers in binary

- Signed numbers involve both positive and negative numbers.
- In binary, a signed-magnitude representation uses the left bit for the sign
  - ↳ 0 means positive
  - ↳ 1 means negative
  - ↳ These are rarely used

- A complement of an N-digit number is another number that yields a sum of 100...00 (with N zeros) and can be used to represent the negative of that number.

Ex 5 − 3 in binary

- ↳ 5 = 0101
- ↳ 3 = 0011
- → 3 has complement $(0011)' + 1 = 1100 + 1 = 1101$

and then do 5 + complement(3) so

$$\begin{array}{r} {}^{1}0^{1}0^{1}1 \leftarrow 5 \\ +\,1101 \leftarrow (-3) \\ \hline +0010 \end{array}$$

ignore carry

this is 2

- The above is called the two's complement representation, which inverts every bit and adds 1.

Ex Assuming two's complement representation, what base 10 number does 1111 represent?

↳ Left bit is 1 so negative

↳ Complement to find positive

↳ 0110 + 1 = 0111

↳ Magnitude is 7. Negate to yield -7.

↳ -7 is answer

- Adding two positives, or adding two negatives, may yield a value that can't be represented in the given number of bits, a situation known as <u>overflow</u>.

↳ Adding two positives may overflow

↳ Adding two negatives may overflow

↳ Adding a positive and negative <u>CANNOT</u> overflow

↳ Overflow if numbers' sign bits are 0's but sum's is 1

↳ Overflow if numbers' sign bits are 1s but sum's is 0.