Hexadecimal # System Base 16 Digit values: 0, 1, ..., 8, 9, A, B, C, D, E, f

(A = 10, f = 15)

0x BEEf

0x identifies following as a hexadecimal #

16^3 16^2 16^1 16^0

B E E f

↳11 ↳14 ↳14 ↳15

Hex → Dec

$0xBEEf = 11 \times 16^3 + 14 \times 16^2 + 14 \times 16 + 15$

$= 48879$

Dec → Hex

48879 / 16

3054.9375

3054 R 15 → f

3054 / 16

190 R 14 → E

190 / 16

11 R 14 → E

11 / 16

0 R 11 → B

→ 0x BEEf

Number Systems

Decimal # System

Each digit is 10 one-of characters 0-9

Base 10

10^3	10^2	10^1	10^0
1,000s	100s	10s	1's
0	4	6	2

$= 0 \times 1000 + 4 \times 100 + 6 \times 10 + 2 \times 1$

Binary # System → Base 2

Each digit: 0 or 1

False True

128	64	32	16	8	4	2	1
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	1	0	0	1	1	0	1

0b ← identifies as binary #

$= 0 \times 128 + 1 \times 64 + 0 \times 32 \ldots + 1 \times 8 + 4 + 1 = 77$

↑ Binary to Decimal Conversion.

Decimal → Binary
Dec → Bin

128	64	32	16	8	4	2	1
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
... 0	0	1	0	0	1	1	0	1

77
−64
13
−8
5
−4
1
−1
0

Method 1: See if it fits

We'll see this (kind of) later in the semester...

Dec → Bin Conversion

Method 2: Divide-By-The-Base

77 ÷ 2

38 R 1

19 R 0

9 R 1

4 R 1

2 R 0

1 R 0

0 R 1

0 R 0

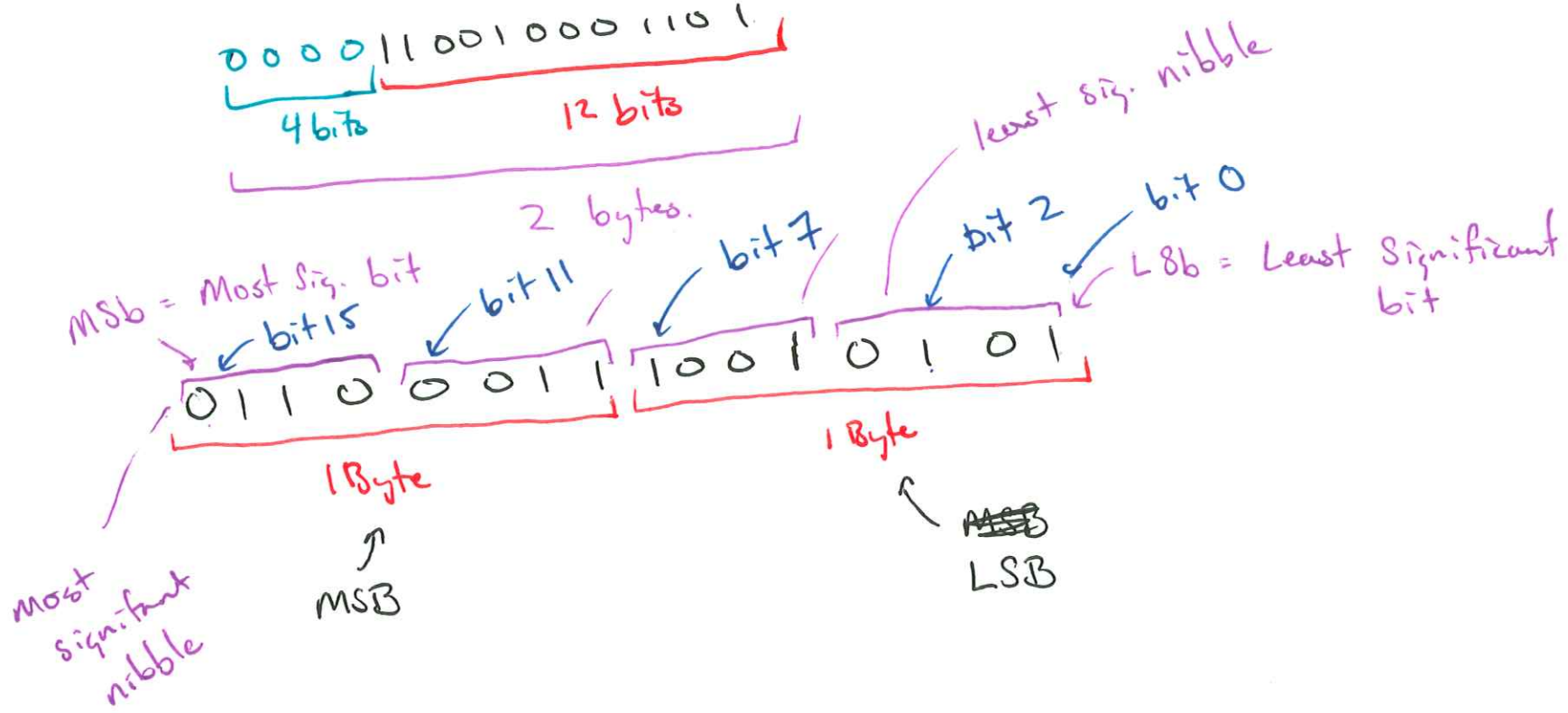
0 1 0 0 1 1 0 1

Binary Terminology

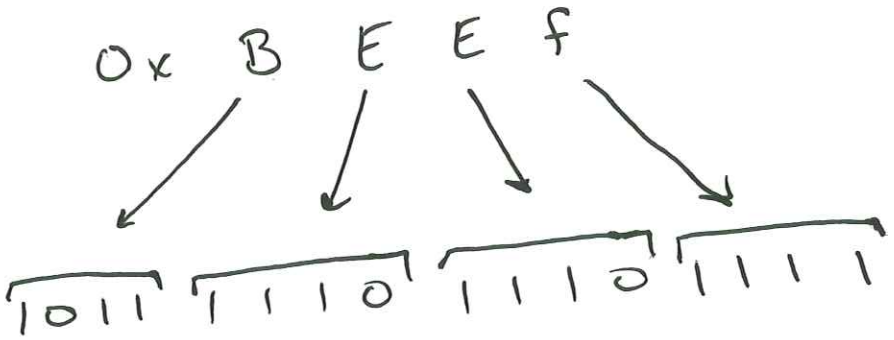
1 Binary Digit = 1 Bit.

8 Bits = 1 Byte

Whenever we represent Binary data in this class do so using full bytes. for ex:



Hex → Bin:



	Dec	Hex
0000	= 0	= 0
0001	= 1	= 1
⋮		
1110	= 14	= E
1111	= 15	= F

0b 01 001 000 110 100 111 010 011 0

4 8 D 3 A 6

↳ 0x 48D3A6 → 3 bytes, 6 nibbles, 24 bits.

Ex Change above # Bit 19 to 1

Digit 8 must have bits 19-16

↘ 19 18 17 bit16

1 0 0 0