[T]EE2026 DIGITAL DESIGN

TUTORIAL 1: NUMBER SYSTEMS

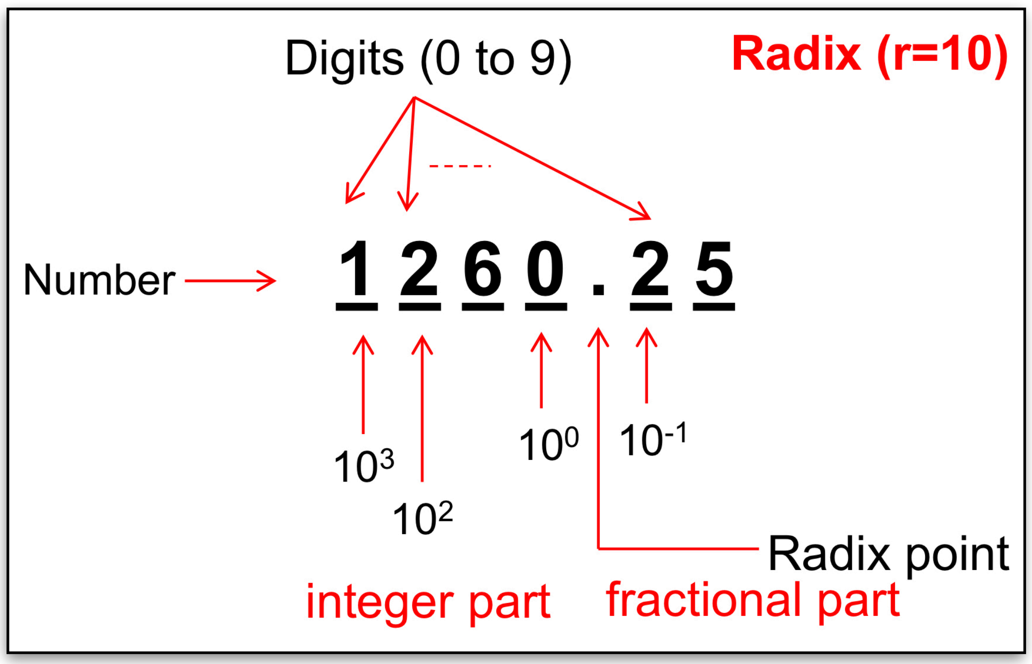
GU JING

ELEGUJI@NUS.EDU.SG

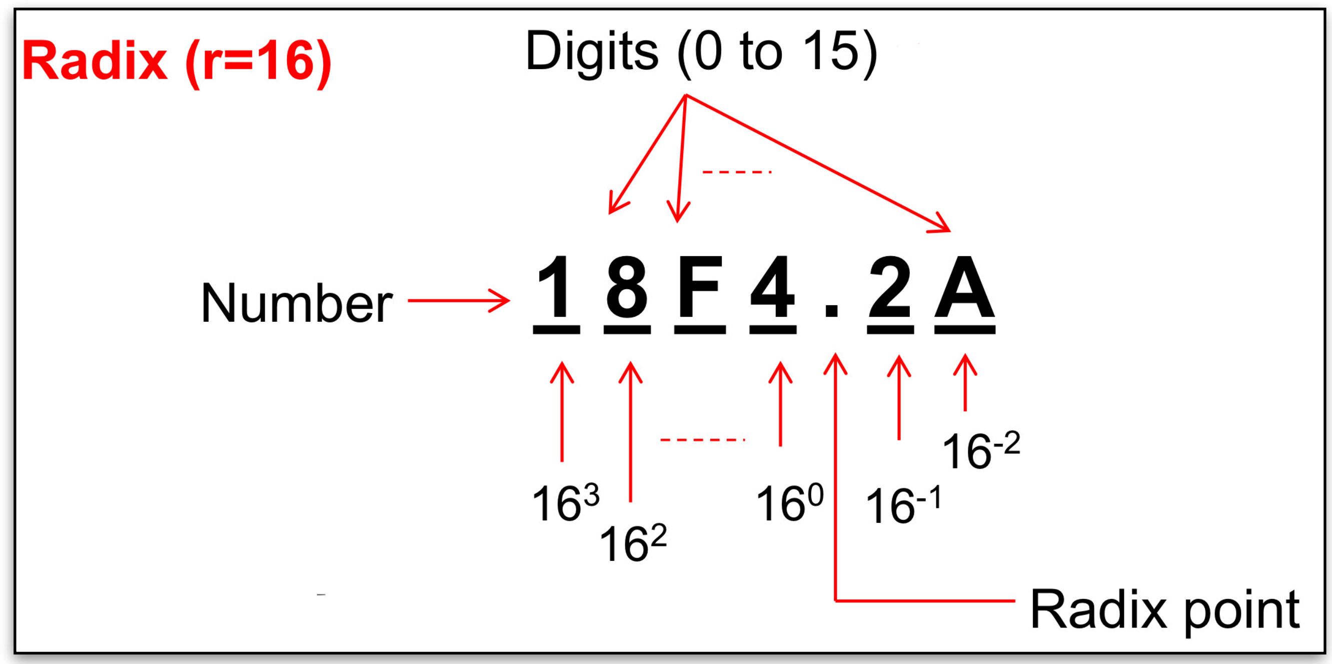
NUMBER SYSTEMS

1. Positional number system (radix 10, 2, 8 and 16)
2. Conversion among decimal, binary, octal and hex.
3. Binary arithmetic
4. Signed binary number representation and Arithmetic (S-M, 1's complement and 2's complement)
5. Binary-coded decimals (BCD) and its Decimal addition

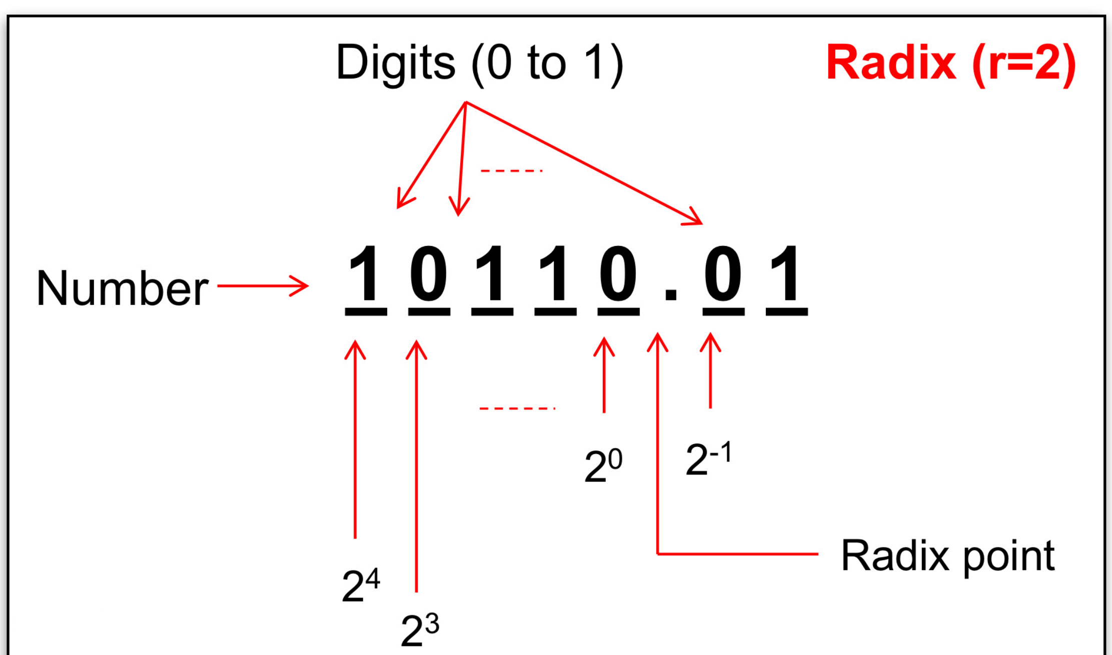
1. POSITIONAL NUMBER SYSTEM



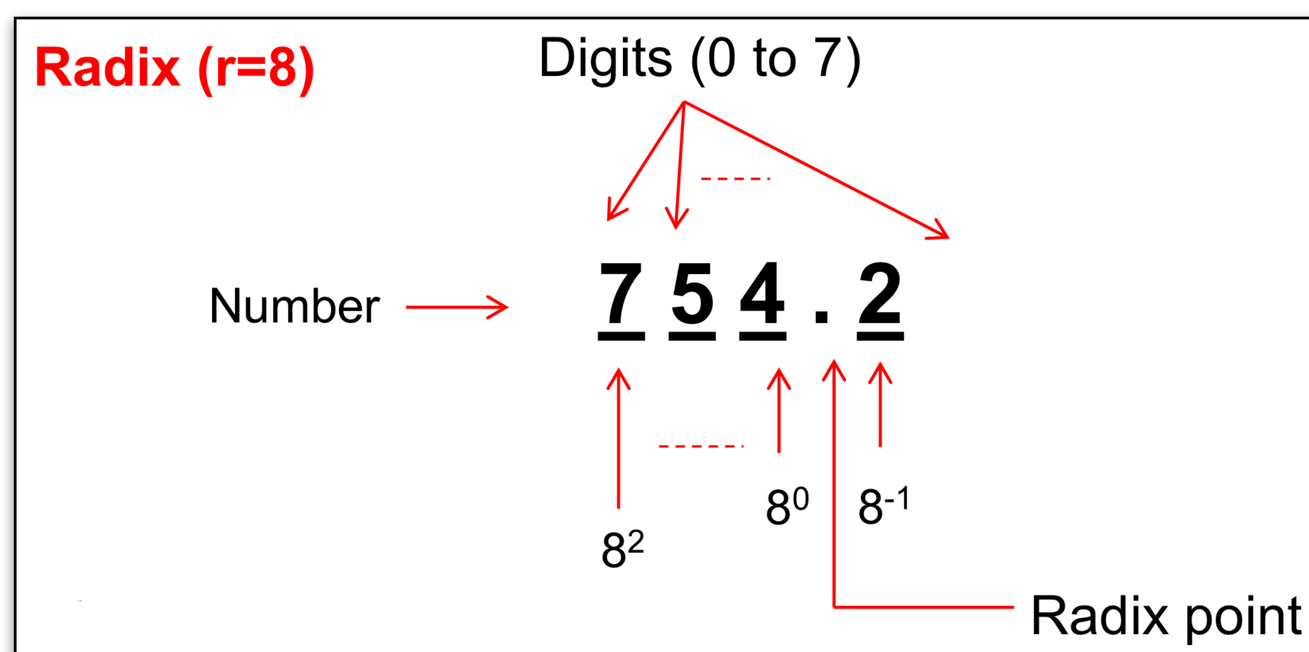
$(1260.25)_{10}$



$(18F4.2A)_{16}$

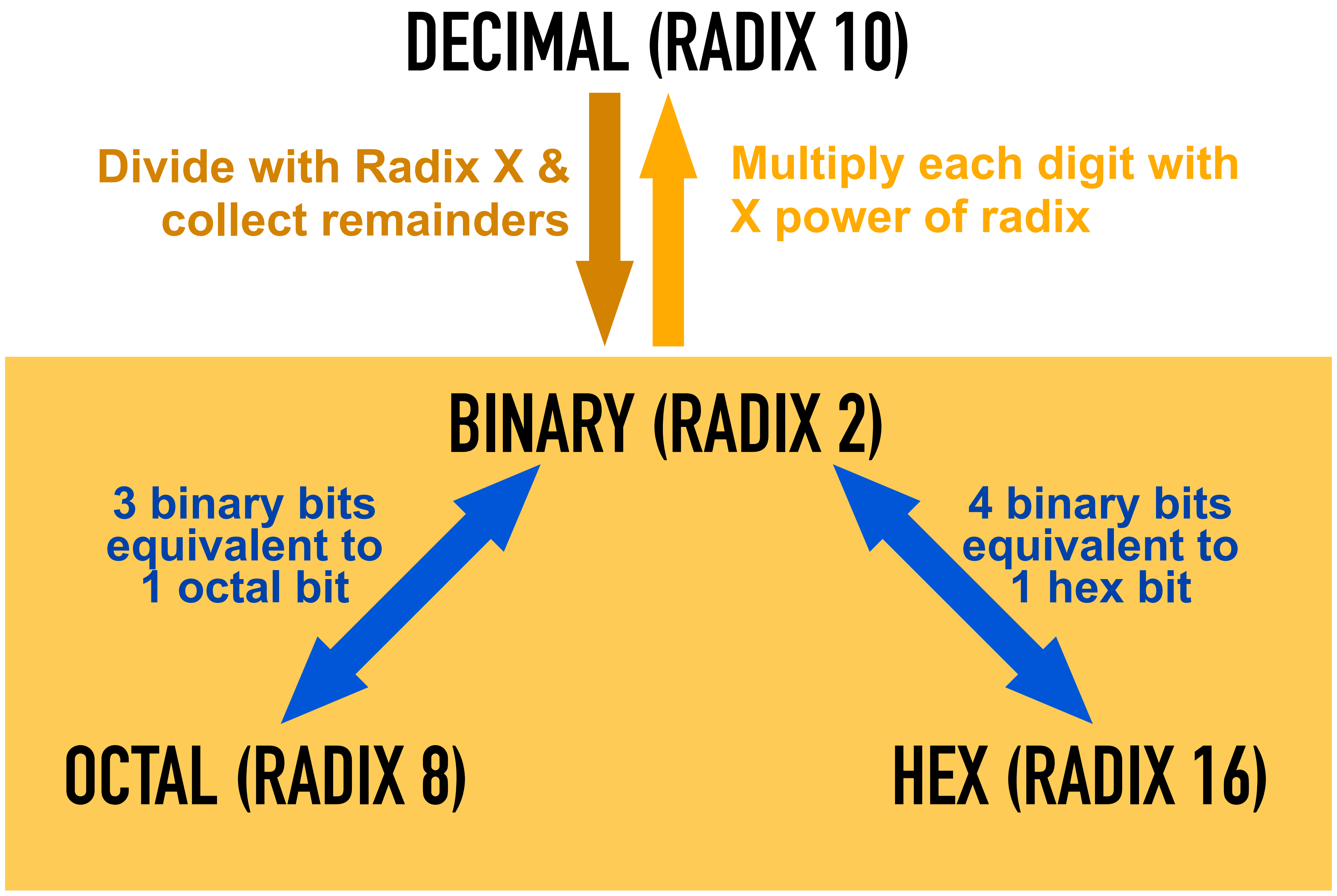


$(10110.01)_2$



$(754.2)_8$

2. NUMBER CONVERSION



2. NUMBER CONVERSION

Q1(a). Convert the decimal number 166.34 into binary.

2 | 166 − 0
2 | 83 − 1
2 | 41 − 1
2 | 20 − 0
2 | 10 − 0
2 | 5 − 1
2 | 2 − 0
1

$0.34 \times 2 = \underline{0}.68$

$0.68 \times 2 = \underline{1}.36$

$0.36 \times 2 = \underline{0}.72$

$0.72 \times 2 = \underline{1}.44$

⋮

$\therefore (166.34)_{10}$

$= (10100110.0101\cdots)_2$

Q1(b). Convert the decimal number 1400.16 to hexadecimal.

16 | 1400 − 8
16 | 87 − 7
5

$0.16 \times 16 = \underline{2}.56$ (2)

$0.56 \times 16 = \underline{8}.96$ (8)

$0.96 \times 16 = \underline{15}.36$ (F)

$0.36 \times 16 = 5.76$ (5)

⋮

$\therefore (1400.16)_{10}$

$= (578.28F5\cdots)_{16}$

2. NUMBER CONVERSION

Q1(c). Convert the binary number $\boxed{101}\boxed{011}\boxed{100}.\boxed{000}\boxed{111}$ into octal.

5 3 4 0 7

$\therefore (534.07)_8$

Q1(d). Convert the hexadecimal number A59.FCE to binary.

1010 0101 1001 1111 1100 1110

$\therefore (\underline{1010}\ \underline{0101}\ \underline{1001}\,.\,\underline{1111}\ \underline{1100}\ \underline{1110})_2$

2. NUMBER CONVERSION

Q1(e). $(62)_\mathbf{x} - (26)_\mathbf{x} = (34)_\mathbf{x}$. Identify **x**.

$x^1 \quad x^0$

$$(6 \cdot x^1 + 2 \cdot x^0) - (2 \cdot x^1 + 6 \cdot x^0) = (3 \cdot x^1 + 4 \cdot x^0)$$

$$6x + 2 - 2x - 6 = 3x + 4$$

$$x = 8$$

$\therefore$ Octal number system.

3. BINARY ARITHMETIC

Addition table:

0 + 0 = 0
0 + 1 = 1
1 + 1 = 10

"1" is the carry to the next higher bit

Example:

10111 + 110 = 11101

```
  1 1        ← Carry
 1 0 1 1 1
+    1 1 0
---------------
 1 1 1 0 1
```

Subtraction table:

0 - 0 = 0
1 - 0 = 1
1 - 1 = 0
0 – 1 = 1 ← with a borrow from the next (higher) bit

Example:

11011 - 110 = 10101

```
  1 1 0 1 1
-     1 1 0
---------------
  1 0 1 0 1
```

Multiplication table:

0 x 0 = 0
0 x 1 = 0
1 x 1 = 1

Example:

10111 x 110 = 10001010

```
       1 0 1 1 1     ← Multiplicand
  x        1 1 0     ← Multiplier
  ---------------
       0 0 0 0 0  }
     1 0 1 1 1    }  ← Partial products
 +  1 0 1 1 1     }
 ----------------------
   1 0 0 0 1 0 1 0   ← Product
```

Multiplication:
→ Shift then Add
→ Only need "add" operation

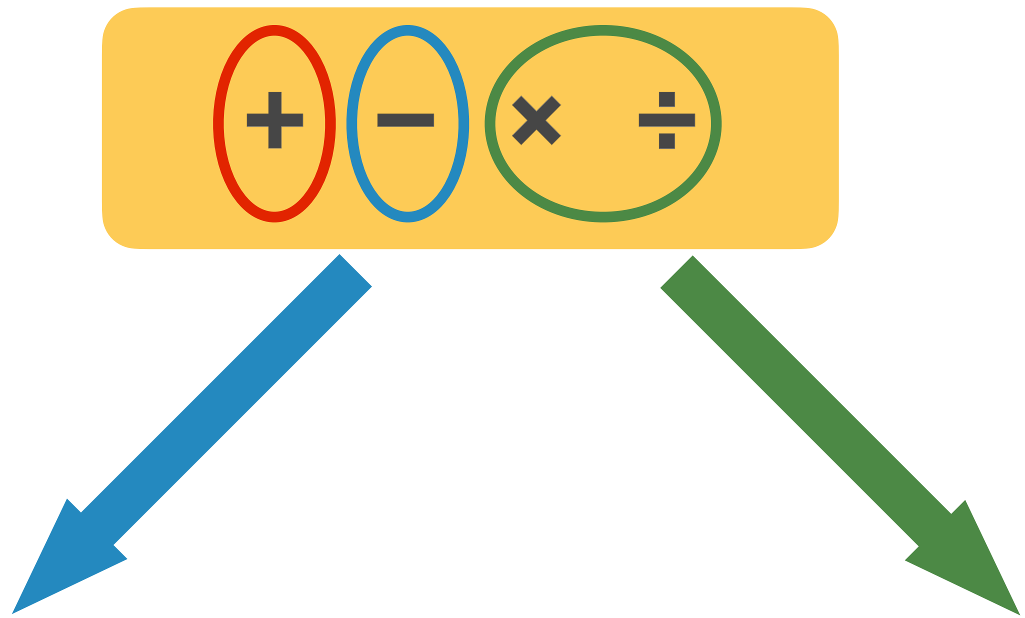
Division

100101/101 = ?

```
        111     ← quotient
101 ) 100101
      101
      ----
       1000
        101
        ---
        111
        101
        ---
         10     ← remainder
```

Division (shift and subtract)
→ Shift then subtraction
→ Only need "subtract" operation

3. BINARY ARITHMETIC: USING COMPUTER



Subtraction: Adding a negative number, A - B = A + (-B)

Multiply: Addition and Shift

Division: Subtraction and Shift

- Thus, a computer may only use **adders** to perform all binary arithmetic operations
- This requires an appropriate representation of the **negative binary** numbers

4. SIGNED BINARY NUMBER REPRESENTATION

- **A. Signed + Magnitude**
- B. 1's Complement
- C. 2's Complement

* **Max magnitude: $(2^{n-1}-1)_{10}$**

* **Range: $-(2^{n-1}-1)_{10} \sim +(2^{n-1}-1)_{10}$**

* **CANNOT be used for addition of two numbers with opposite signs or subtraction when using a simple adder,**

e.g. 011 (3) - 101 (1) = 100 (0)

Decimal	S-M
3	011
2	010
1	001
+0	000
-0	**100**
-1	**101**
-2	**110**
-3	**111**

Note:
Two zeros

Negative numbers

"1" in MSB position for all negative numbers

4. SIGNED BINARY NUMBER REPRESENTATION

Q2. Convert the following decimal numbers into **8-bit signed magnitude** representations:

(a) +127

MSB=0

$(127)_{10}$ 7-bit

$= (111\ 1111)_2$

$\therefore (127)_{10} = (0111\ 1111)_2$

(b) -0

MSB=1

$(0)_{0}$

$= (0000000)_2$

$\therefore (-0)_{0}$

$= (1000\ 0000)_2$

(c) -55

MSB=1

$(55)_{10} = (011\ 0111)_2$

$\therefore (-55)_{10}$

$= (1011\ 0111)_2$

4. SIGNED BINARY NUMBER REPRESENTATION

A. Signed + Magnitude

B. 1's Complement

C. 2's Complement

*** Diminished Radix Complement: $A^* = (2^n - 1) - A$**

*** Known A, find A* : reversing the bits**

E.g. Binary number (n=8): 0 1 0 1 1 1 0 0

1's Complement: 1 1 1 1 1 1 1 1 - 0 1 0 1 1 1 0 0

= 1 0 1 0 0 0 1 1

Bits Reverse

4. SIGNED BINARY NUMBER REPRESENTATION

A. Signed + Magnitude

B. 1's Complement

C. 2's Complement

Decimal	1's Complement
3	011
2	010
1	001
+0	000
-0	**111**
-1	**110**
-2	**101**
-3	**100**

Still two zeros

*** Range: $-(2^{n-1}-1)_{10} \sim +(2^{n-1}-1)_{10}$**

*** It has no problem to perform subtraction, but needs to shift and add the carry.**

3 – 2 = 3+(-2)=1

```
    011
+   101
--------
 (1)000
+     1
--------
    001
```

✓

3 – 1 = 3+(-1) = 2

```
    011
+   110
--------
 (1)001
+     1
--------
    010
```

✓

4. SIGNED BINARY NUMBER REPRESENTATION

Q3. Convert the following signed decimal numbers into **10-bit 1's complement** representations:

(a) +43

2 | 43 − 1
2 | 21 − 1
2 | 10 − 0
2 | 5 − 1
2 | 2 − 0
1

$\therefore (43)_{10} = (0000101011)_2$

(b) -1

1 : 0000000001

↓ flip

$\therefore -1 : (1111111110)_2$

(c) -128

128 : 0010000000

↓ flip

$\therefore -128 : (1101111111)_2$

4. SIGNED BINARY NUMBER REPRESENTATION

A. Signed + Magnitude

B. 1's Complement

C. 2's Complement

*** Radix Complement: $\boxed{A^* = 2^n - A}$ = 1's Complement + 1**

*** Known A, find A* : reversing the bits, then plus 1**

E.g. Binary number (n=8): 0 1 0 1 1 1 0 0

$$= \underbrace{\boxed{1\ 1\ 1\ 0\ 0\ 0\ 1\ 1}}_{\text{Bits Reverse}} + 1$$

2's Complement: = 1 0 1 0 0 1 0 0

*** Know A*, find A: since $\boxed{A = 2^n - A^*}$, just apply 2's complement to A*, you can find A. (See tutorial Q4b)**

4. SIGNED BINARY NUMBER REPRESENTATIO

A. Signed + Magnitude

B. 1's Complement

C. 2's Complement

Decimal	2's Complement
3	011
2	010
1	001
0	000
-1	**111**
-2	**110**
-3	**101**
-4	**100**

Only one zero

*** Range: $-(2^{n-1})_{10} \sim +(2^{n-1}-1)_{10}$**

*** It has no problem to perform subtraction, and no need to shift and add the carry (hardware efficient)**

3 – 2 = 3+(-2)=1

```
   011
+  110
--------
(1)001
```

↑

Carry ignored

3 – 1 = 3+(-1) = 2

```
   011
+  111
--------
(1)010
```

↑

Carry ignored

4. SIGNED BINARY NUMBER REPRESENTATION

A. Signed + Magnitude
B. 1's Complement
C. 2's Complement

Decimal	Signed-2's Complement	Signed-1's Complement	Signed Magnitude
+7	0111	0111	0111
+6	0110	0110	0110
+5	0101	0101	0101
+4	0100	0100	0100
+3	0011	0011	0011
+2	0010	0010	0010
+1	0001	0001	0001
+0	0000	0000	0000
−0	—	1111	1000
−1	1111	1110	1001
−2	1110	1101	1010
−3	1101	1100	1011
−4	1100	1011	1100
−5	1011	1010	1101
−6	1010	1001	1110
−7	1001	1000	1111
−8	1000	—	—

* Positive numbers are same in all 3 signed binary number representations

4. SIGNED BINARY NUMBER REPRESENTATION

Q4. Convert the following **2's complement** numbers to their signed decimal equivalents:

(a) 10000 (b) 10000001

assume 5-bit system is used:

$\because$ MSB=1, it's a -ve no.

10000

$\downarrow$ flip

01111

$\downarrow$ +1

10000, $\therefore$ mag = 16

$\therefore (10000)_{2's} = (-16)_{10}$

assume 8-bit system is used:

$\because$ MSB=1, it's a -ve no.

10000001

$\downarrow$ flip

01111110

$\downarrow$ +1

01111111, $\therefore$ magnitude = 127

$\therefore (10000001)_{2's} = (-127)_{10}$

THE END

For Consultation: eleguji@nus.edu.sg
Office: E4-03-10