



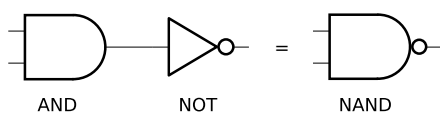
Introduction to Computer 2022

U1525
National Taipei University

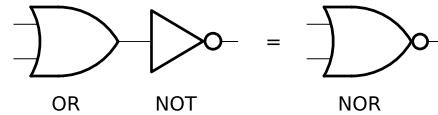


Homework Assignments 1 [Chapter 1]

- We usually use a **NAND** gate to represent an **AND** gate followed by a **NOT** gate (Fig. 1a), and a **NOR** gate to represent a **OR** gate followed by a **NOT** gate (Fig. 1b).



(a) NAND gate



(b) NOR gate

Based on the definitions of the **NAND** and the **NOR** gates, what Boolean operations do the following circuits compute?

(a) Fig. 2 (5%)

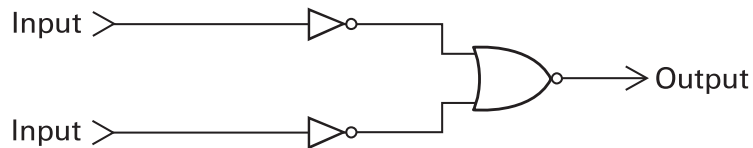


Figure 2: Question 1-(a)

(b) Fig. 3 (5%)

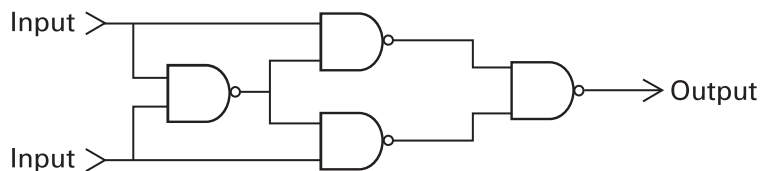


Figure 3: Question 1-(b)

- How many cells can be in a computer's main memory if each cell's address can be represented by **two** hexadecimal digits? (5%) What if **four** hexadecimal digits are used? (5%)
- Convert each of the following excess-32 (6-bit) representations to its equivalent base 10 representation:
 - 100110 (5%)
 - 111000 (5%)

4. Convert each of the following base 10 representations to its equivalent excess-16 (5-bit) representation:
 - (a) -8 (5%)
 - (b) 10 (5%)
5. Convert each of the following two's complement representations to its equivalent base 10 representation:
 - (a) 010101 (5%)
 - (b) 111001 (5%)
6. Convert each of the following base 10 representations to its equivalent two's complement representation in which each value is represented in 8 bits:
 - (a) 21 (5%)
 - (b) -18 (5%)
7. Perform each of the following additions assuming the bit strings represent values in two's complement notation. Identify each case in which the answer is incorrect because of overflow.
 - (a) $01111 + 00001$ (5%)
 - (b) $00111 + 01100$ (5%)
8. Decode the following bit patterns using the floating-point format shown in Fig. 4:
 - (a) 10101100 (5%)
 - (b) 00111001 (5%)
9. Encode the following values using the 8-bit floating-point format described in Fig. 4. Indicate each case in which a truncation error occurs.
 - (a) $7/32$ (5%)
 - (b) $31/32$ (5%)
10. What is the best approximation to the value one-tenth that can be represented using the 8-bit floating-point format described in Fig. 4? (10%)

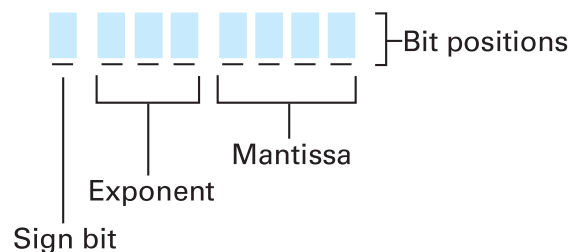


Figure 4: 8-bit floating point format