EEE104 – Digital Electronics (I)

Lecture 3

Dr. Ming Xu

Dept of Electrical & Electronic Engineering

XJTLU

In This Session

- Binary Arithmetic
- Hexadecimal Numbers.
- Binary Coded Decimal (BCD)

Binary Arithmetic

1's Complement

- This is to change all 1s to 0s and all 0s to 1s in a binary number.
- It is important to the representation of negative numbers.

1 0 1 1 0 0 1 0	Binary number
↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓	
0 1 0 0 1 1 0 1	1's complement

Binary Arithmetic

2's Complement

- This is to add 1 to the 1's complement.
- It is important to the representation of negative numbers.

10110010	Binary number
01001101	1's complement
+ 1	Add 1
01001110	2's complement

Hexadecimal Numbers

- Long binary numbers are difficult to read and write.
- So **hexadecimal number system** is introduced as a compact way of writing binary numbers.
- It is widely used in computers and microprocessors.

Hexadecimal Numbers

- The hexadecimal number system has 16 digits: 10 numeric digits (0-9) and 6 alphabetic characters (A-F).
- Each digit represents a 4-bit binary number.
- A hexadecimal number may have a subscript 16 or be followed by an "h".

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Hexadecimal Numbers

Counting in Hexadecimal

- Once you get to F, add another digit and continue.

  0, 1, ……, 9, A, B, C, D, E, F

  10, 11, ……, 19, 1A, 1B, 1C, 1D, 1E, 1F

  ……

  F0, F1, ……, F9, FA, FB, FC, FD, FE, FF

  100, 101, ……109, 10A, 10B, 10C, 10D, 10E, 10F

Hexadecimal Numbers

Binary-to-Hexadecimal Conversion

- Starting at the right-most bit, break the binary number into 4-bit groups.
- Replace each 4-bit group with the equivalent hexadecimal symbol.

$$\underbrace{1100}_{\downarrow\ C}\underbrace{1010}_{\downarrow\ A}\underbrace{0101}_{\downarrow\ 5}\underbrace{0111}_{\downarrow\ 7} = \mathbf{CA57}_{16}$$

Hexadecimal Numbers

Hexadecimal-to-Binary Conversion

- Replace each hexadecimal symbol with the appropriate 4 bits.

1 0 A 4 → 1000010100100

C F 8 E → 1100111110001110

Hexadecimal Numbers

Hexadecimal-to-Decimal Conversion

- The weights of hexadecimal digits are increasing powers of 16 (from right to left).

16^3	16^2	16^1	16^0
4096	256	16	1

- Multiply the decimal value of each hexadecimal digit by its weight and then take the sum of these products.

$$\begin{aligned} B2F8_{16} &= (B \times 4096) + (2 \times 256) + (F \times 16) + (8 \times 1) \\ &= (11 \times 4096) + (2 \times 256) + (15 \times 16) + (8 \times 1) \\ &= 45{,}056 + 512 + 240 + 8 = \mathbf{45{,}816_{10}} \end{aligned}$$

Hexadecimal Numbers

Decimal-to-Hexadecimal Conversion

- Divide a decimal number or the previous quotient by 16. The remainder is a digit in the hexadecimal number.
- The first remainder is the LSD.
- Repeat this process until the whole number quotient becomes zero.

	quotient	remainder	
$\frac{650}{16}$	40	10 = A	$650 = 28A_{16}$
$\frac{40}{16}$	2	8 = 8	
$\frac{2}{16}$	0	2	

Hexadecimal Numbers

Hexadecimal Addition

- If the sum of two digits is 15_{10} or less, bring down the corresponding hexadecimal digit.
- If the sum of these two digits is greater than 15_{10}, bring down the amount of the sum that exceeds 16_{10} and carry a 1 to the next column.

$$\begin{array}{r} 58_{16} \\ +\ 22_{16} \\ \hline \mathbf{7A}_{16} \end{array}$$

right column: $8_{16} + 2_{16} = 8_{10} + 2_{10} = 10_{10} = A_{16}$

left column: $5_{16} + 2_{16} = 5_{10} + 2_{10} = 7_{10} = 7_{16}$

$$\begin{array}{r} DF_{16} \\ +\ AC_{16} \\ \hline \mathbf{18B}_{16} \end{array}$$

right column: $F_{16} + C_{16} = 15_{10} + 12_{10} = 27_{10}$

$27_{10} - 16_{10} = 11_{10} = B_{16}$ with a 1 carry

left column: $D_{16} + A_{16} + 1_{16} = 13_{10} + 10_{10} + 1_{10} = 24_{10}$

$24_{10} - 16_{10} = 8_{10} = 8_{16}$ with a 1 carry

Binary Coded Decimal (BCD)

- **Binary coded decimal** (BCD) is an easy way to express decimal digits with a binary code.
- The BCD system has only 10 code groups.
- It is mainly used in user interface such as keypads and digital displays.
- The **8421 code** is a type of BCD, where the weights of the four bits are 8, 4, 2 and 1.

Decimal Digit	0	1	2	3	4	5	6	7	8	9
BCD	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

Binary Coded Decimal (BCD)

Decimal-to-BCD Conversion

- Replace each decimal digit with the appropriate 4-bit.

$$\begin{array}{ccc} 1 & 7 & 0 \\ \downarrow & \downarrow & \downarrow \\ \underbrace{\mathbf{0001}} & \underbrace{\mathbf{0111}} & \underbrace{\mathbf{0000}} \end{array}$$

BCD-to-Decimal Conversion

- Start at the right-most bit and break the code into groups of four bits.
- Write the decimal digit for each 4-bit group.

$$\begin{array}{ccc} \underbrace{0011} & \underbrace{0101} & \underbrace{0001} \\ \downarrow & \downarrow & \downarrow \\ \mathbf{3} & \mathbf{5} & \mathbf{1} \end{array}$$

Binary Coded Decimal (BCD)

BCD Addition

- Add the two BCD number using the rules for binary addition.
- If a 4-bit sum is equal to or less than 9, it is a valid BCD number.
- If a 4-bit sum is greater than 9, or if a carry out of the 4-bit group is generated, it is an invalid result. Add 6 (0110) to the 4-bit sum to skip the six invalid states.

$$\begin{array}{rr} 0010 & 0011 \\ +\,0001 & 0101 \\ \hline \mathbf{0011} & \mathbf{1000} \end{array} \qquad \begin{array}{r} 23 \\ +\,15 \\ \hline 38 \end{array}$$

Binary Coded Decimal (BCD)

BCD Addition

$$\begin{array}{rrl} & 1001 & \\ & +\,0100 & \\ \hline & 1101 & \text{Invalid BCD number } (>9) \\ & +\,0110 & \text{Add 6} \\ \hline \underbrace{\mathbf{0001}} & \underbrace{\mathbf{0011}} & \text{Valid BCD number} \\ \downarrow & \downarrow & \\ 1 & 3 & \end{array} \qquad \begin{array}{r} 9 \\ +\,4 \\ \hline 13 \end{array}$$

$$\begin{array}{rrl} & 1001 & \\ & +\,1001 & \\ \hline 1 & 0010 & \text{Invalid because of carry} \\ & +\,0110 & \text{Add 6} \\ \hline \underbrace{\mathbf{0001}} & \underbrace{\mathbf{1000}} & \text{Valid BCD number} \\ \downarrow & \downarrow & \\ 1 & 8 & \end{array} \qquad \begin{array}{r} 9 \\ +\,9 \\ \hline 18 \end{array}$$