

CSE 260

Digital Logic Design


Text Books:

Core Texts:

- ❑ **Digital Logic And Computer Design, Morris Mano**

Reference material:

- ❑ Digital Design Principles and Practices, John F. Wakerly

The background of the slide is a detailed, stylized illustration of a printed circuit board (PCB). It features a complex network of thin, light green lines representing the circuit traces, which are interconnected by numerous small, circular green solder pads. The overall pattern is dense and organic, typical of modern electronic designs. The text is centered over this background.

CSE 260

Digital Logic Design

Number System & Calculations

Objective

- Distinguish between analog and digital system
- Understand the advantage and limitation of digital system
- Understand the meaning of digital logic

Analog vs. Digital

- Analog data can vary over a continuous range of values. Example: speedometer
- Digital quantities can take on only discrete values (0 and 1, high and low). Example: Digital Computer, Decimal Digits, Alphabets

Digital System

- A digital system is a combination of devices designed to manipulate physical quantities or information that are represented in digital form.
- “A discreet information processing system”
- Signals: Discreet information

Digital Technology

Advantage

- Greater accuracy or precision
- Easier to design (generality)
- Easier information storage
- Programmability (instructions)
- Speed
- Economical

Limitation

- The real world is mainly analog

How to Overcome

- Convert the real world analog input data into digital one
- Process this digital data
- Then again convert into analog form

Digital logic

- Design logic is a term used to denote the **design** and **analysis** of digital system
- Digital logic is concerned with the interconnection among digital components and modules
- Digital logic design is engineering and engineering means problem solving

Number systems and codes

Digital Systems are built from circuits that process binary digits. BUT very few real-life problems are based on binary numbers.

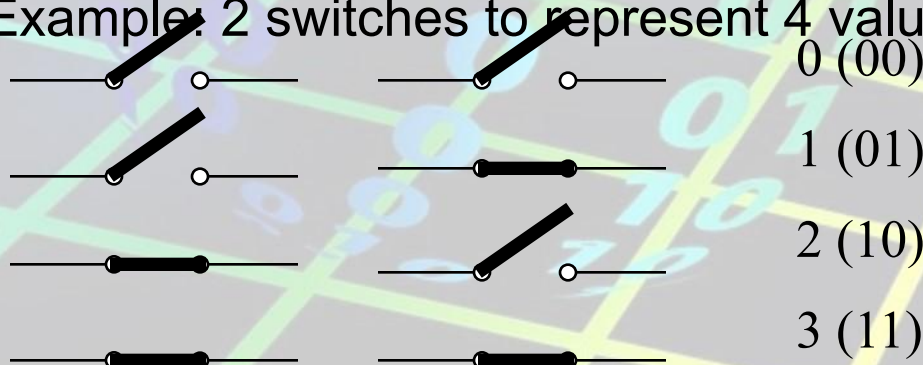
SO a digital system designer must establish some **correspondence between the binary digits processed by digital circuits and real-life numbers**, events and conditions.

Information representation

- Human decisions tends to be binary i.e. Yes or No
- Elementary storage units inside computer are *electronic switches*. Each switch holds one of two states: *on* (1) or *off* (0).



- We use a *bit* (*binary digit*), 0 or 1, to represent the state.
- Storage units can be grouped together to cater for larger range of numbers. Example: 2 switches to represent 4 values.



Information representation

- In general, N bits can represent 2^N different values.
- For M values, $\lceil \log_2 M \rceil$ bits are needed.

1 bit → represents up to 2 values (0 or 1)

2 bits → rep. up to 4 values (00, 01, 10 or 11)

3 bits → rep. up to 8 values (000, 001, 010. ..., 110, 111)

4 bits → rep. up to 16 values (0000, 0001, 0010, ..., 1111)

32 values → requires 5 bits

64 values → requires 6 bits

1024 values → requires 10 bits

40 values → requires 6 bits

100 values → requires 7 bits

Positional Notations

- Decimal number system, symbols = $\{ 0, 1, 2, 3, \dots, 9 \}$
- Position is important
- Example: $(7594)_{10} = (7 \times 10^3) + (5 \times 10^2) + (9 \times 10^1) + (4 \times 10^0)$
- In general, $(a_n a_{n-1} \dots a_0)_{10} = (a_n \times 10^n) + (a_{n-1} \times 10^{n-1}) + \dots + (a_0 \times 10^0)$
- $(2.75)_{10} = (2 \times 10^0) + (7 \times 10^{-1}) + (5 \times 10^{-2})$
- In general, $(a_n a_{n-1} \dots a_0 . f_1 f_2 \dots f_m)_{10} = (a_n \times 10^n) + (a_{n-1} \times 10^{n-1}) + \dots + (a_0 \times 10^0) + (f_1 \times 10^{-1}) + (f_2 \times 10^{-2}) + \dots + (f_m \times 10^{-m})$

Other Number Systems

- **Binary** (base 2): weights in powers-of-2.
Binary digits (bits): **0,1**.
- **Octal** (base 8): weights in powers-of-8.
Octal digits: **0,1,2,3,4,5,6,7**
- **Hexadecimal** (base 16): weights in powers-of-16. Hexadecimal digits: **0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F**

Note: when base is r , coefficient values range from 0 to $r-1$.

Other Number System

- **Binary** (base 2): weights in powers-of-2. Binary digits (bits): **0,1**.
- **Octal** (base 8): weights in powers-of-8. Octal digits: **0,1,2,3,4,5,6,7**
- **Hexadecimal** (base 16): weights in powers-of-16. Hexadecimal digits: **0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F**

Binary	Octal	Decimal	Hexadecimal
0000	0	0	0
0001	1	1	1
0010	2	2	2
0011	3	3	3
0100	4	4	4
0101	5	5	5
0110	6	6	6
0111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F

Base-R to Decimal Conversion

***Formula= $\sum \text{digit} * \text{source_base}^{\text{position}}$

$$\begin{aligned} \blacksquare (1101.101)_2 &= 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-3} \\ &= 8 + 4 + 1 + 0.5 + 0.125 \\ &= (13.625)_{10} \end{aligned}$$

$$\begin{aligned} \blacksquare (572.6)_8 &= 5 \times 8^2 + 7 \times 8^1 + 2 \times 8^0 + 6 \times 8^{-1} \\ &= 320 + 56 + 2 + 0.75 = (378.75)_{10} \end{aligned}$$

$$\begin{aligned} \blacksquare (2A.8)_{16} &= 2 \times 16^1 + 10 \times 16^0 + 8 \times 16^{-1} \\ &= 32 + 10 + 0.5 = (42.5)_{10} \end{aligned}$$

$$\begin{aligned} \blacksquare (341.24)_5 &= 3 \times 5^2 + 4 \times 5^1 + 1 \times 5^0 + 2 \times 5^{-1} + 4 \times 5^{-2} \\ &= 75 + 20 + 1 + 0.4 + 0.16 = (96.56)_{10} \end{aligned}$$

Decimal to Base-R Conversion

- **Decimal to base-R**

- ❖ Whole numbers: repeated division-by-R
- ❖ Fractions: repeated multiplication-by-R

Repeated Division-by-2 Method

- To convert a **whole number** to binary, use **successive division by 2** until the quotient is 0. The remainders form the answer, with the first remainder as the *least significant bit (LSB)* and the last as the *most significant bit (MSB)*.

2	43		
2	21	rem 1	← LSB
2	10	rem 1	
2	5	rem 0	
2	2	rem 1	
2	1	rem 0	
	0	rem 1	← MSB

$$(43)_{10} = (101011)_2$$

Repeated Multiplication-by-2 Method

- To convert **decimal fractions** to binary, **repeated multiplication by 2** is used, until the fractional product is 0 (or until the desired number of decimal places). The carried digits, or *carries*, produce the answer, with the first carry as the MSB, and the last as the LSB.

$$(0.3125)_{10} = (.0101)_2$$

	Carry	
$0.3125 \times 2 = 0.625$	0	←MSB
$0.625 \times 2 = 1.25$	1	
$0.25 \times 2 = 0.50$	0	
$0.5 \times 2 = 1.00$	1	←LSB

Note: difference between conversion of integer and fraction

Integer

- Division by base of target no. system
- Remainders are accumulated
- By division we obtain LSB to MSB

Fraction

- Multiplication by base of target no. system
- Integers are accumulated
- By multiplication we obtain MSB to LSB

Binary-Octal/Hexadecimal Conversion

- **Binary → Octal**: Partition in groups of 3

$$(\underbrace{10}_1 \underbrace{111}_2 \underbrace{011}_3 \underbrace{001}_4 . \underbrace{101}_5 \underbrace{110}_6)_2 = (2731.56)_8$$

- **Octal → Binary**: reverse

$$(2731.56)_8 = (10 \ 111 \ 011 \ 001 . 101 \ 110)_2$$

- **Binary → Hexadecimal**: Partition in groups of 4

$$(\underbrace{101}_1 \underbrace{1101}_2 \underbrace{1001}_3 . \underbrace{1011}_4 \underbrace{1000}_5)_2 = (5D9.B8)_{16}$$

- **Hexadecimal → Binary**: reverse

$$(5D9.B8)_{16} = (101 \ 1101 \ 1001 . 1011 \ 1000)_2$$

Binary-Octal/Hexadecimal Conversion

- $(1100100.10)_2 = (144.4)_8$
 $= (64.8)_{16}$

Exercise:

(1) Try converting this to

$(10110001101011.111100000110)_2$

a) octal b) hexadecimal

(2) Try converting these to binary

a) $(673.124)_8$

b) $(306.D)_{16}$

Answers:

(1) a) $(26153.7406)_8$

(1) b) $(2C6B.F06)_{16}$

(2) a) $(110\ 111\ 011\ .\ 001\ 010\ 100)_2$

(2) b) $(0011\ 0000\ 0110\ .\ 1101)_2$

Binary operations: Addition: Addition Rules w/Carries

For 2 bit

- $0 + 0 = 00$ (0 with a 0 carry)
- $0 + 1 = 01$ (1 with a 0 carry)
- $1 + 0 = 01$ (1 with a 0 carry)
- $1 + 1 = 10$ (0 with a 1 carry)

For 3 bit

- $0+0+0 = 000$ (0 WITH 0 CARRY)
- $0+0+1 = 01$ (1 WITH 0 CARRY)
- $0+1+1 = 10$ (0 WITH 1 CARRY)
- $1+1+1 = 11$ (1 WITH 1 CARRY)

Adding Binary Numbers

$$\begin{array}{r} 28 \\ + 43 \\ \hline 71 \end{array} \quad \begin{array}{l} \longrightarrow \\ \longrightarrow \end{array} \begin{array}{r} 0111000 \\ 00011100 \\ + 00101011 \\ \hline 01000111 \end{array}$$

Exercise

3(a) Add $(101101)_2$ with $(100111)_2$

Solution

- 3(a) $(1010100)_2$

Working:

Augend: 101101

Addend: +100111

Sum: 1010100

Addition of base-r

- Example:

$$(34)_5 + (41)_5 + (24)_5$$

$1+3+4+2=10$
 $10\%5=0$
 $10/5=2$ (carry)

21

$$(34)_5$$

$$\begin{array}{r} (41)_5 \\ + (24)_5 \\ \hline \end{array}$$

$$(204)_5$$

$$4+1+4=9$$

$$9\%5=4$$

$$9/5=1$$
 (carry)

Exercise

- Try:

$$2 \text{ (a) } (FF)_{16} + (F1)_{16}$$

$$2 \text{ (b) } (66)_7 + (55)_7$$

Solution

- $2 \text{ (a) } (1F0)_{16}$

- $2 \text{ (b) } (154)_7$

Binary Multiplication

- The multiplication of two binary numbers can be carried out in the same manner as the decimal multiplication.
- Unlike decimal multiplication, only two values are generated as the outcome of multiplying the multiplication bit by 0 or 1 in the binary multiplication. These values are either 0 or 1.
- The binary multiplication can also be considered as repeated binary addition. Therefore, the binary multiplication is performed in conjunction with the binary addition operation.

Exercise

6 (a) Multiply 1011 with 101

Multiplicand

Multiplier

Partial Products

Product

$$\begin{array}{r} 1011 \\ \times 101 \\ \hline 1011 \\ 0000 \\ 1011 \\ \hline 110111 \end{array}$$

Multiplication with base-r

- $2A3C$
 $\times B7$

$127A4$
 $1D094$

$1E30E4$

Working

$$7 * C = 7 * 12 = 84 = (\text{write } 4, \text{ carry } 5)$$

$$7 * 3 + 5 = 26 = 1A \text{ (write } A, \text{ carry } 1)$$

$$7 * A + 1 = 71 = 0x47 \text{ (write } 7, \text{ carry } 4)$$

$$7 * 2 + 4 = 18 = 0x12$$

This completes the $7 * 2A3C = 127A4$ partial product.

$$B * C = 11 * 12 = 132 = (\text{write } 4, \text{ carry } 8)$$

$$B * 3 + 8 = 11 * 3 + 8 = 41 = (\text{write } 9, \text{ carry } 2)$$

$$B * A + 2 = 11 * 10 + 2 = 112 = (\text{write } 0, \text{ carry } 7)$$

$$B * 2 + 7 = 11 * 2 + 7 = 29 = 1D$$

This completes the $B[0] * 2A3C = 1D094[0]$ partial product, where I'm noting the $[0]$ digits to remind us this is in the 16s column.

Adding the partial products: $127A4 + 1D0940$

$$4 + 0 = 4$$

$$A + 4 = E$$

$$7 + 9 = 16 = 0x10 \text{ (write } 0, \text{ carry } 1)$$

$$2 + 0 + 1 = 3$$

$$1 + D = E$$

$$1 = 1$$

Exercise

7 (a) Multiply $(34)_5$ with $(42)_5$

7 (b) Multiply $(25)_9$ with $(36)_9$

Solution

7 (a) $(3133)_5$

7 (b) $(1033)_9$

Binary operations:

Subtraction Rules w/Carries

- For 2 bit
 - $0 - 0 = 00$ (0 with a 0 carry)
 - $1 - 1 = 00$ (0 with a 0 carry)
 - $1 - 0 = 01$ (1 with a 0 carry)
 - $0 - 1 = ?$ (???)

Subtracting Binary Numbers:

$$\begin{array}{r} 2 \\ - 1 \\ \hline 1 \end{array}$$

$$\begin{array}{r} 4 \\ - 1 \\ \hline 3 \end{array}$$

$$\begin{array}{r} \xrightarrow{\quad} 10 \\ \xrightarrow{\quad} - 01 \\ \hline 01 \end{array}$$

$$\begin{array}{r} \xrightarrow{\quad} 100 \\ \xrightarrow{\quad} - 001 \\ \hline 011 \end{array}$$

Exercise

5(b) Subtract $(100111)_2$ from $(101101)_2$

Solution

5 (b) $(000110)_2$

Working:

5(b) Subtraction

Minuend: 101101

Subtrahend: -100111

Difference: 000110

Subtraction of Base-r

$$\begin{array}{r} ^3 ^{16} \\ (4 \text{ A } 6)_{16} \\ - (1 \text{ B } 3)_{16} \\ \hline (2 \text{ F } 3)_{16} \end{array}$$

$$\begin{array}{r} ^4 ^6 \\ (54)_6 \\ - (35)_6 \\ \hline (15)_6 \end{array}$$

Exercise

$$4 \text{ (a) } (71)_8 - (56)_8$$

$$4 \text{ (b) } (21)_3 - (12)_3$$

Solution

- 4 a) $(13)_8$
- 4 b) $(2)_3$

Division of Base-r

■ Dividend: $x_1x_2x_3...x_m$; Divisor: $y_1y_2...y_n = y$;

Perform $101_2 / 10_2$; Dividend = 101 and Divisor = 10

Step1: Start with the first digit of the dividend. (x_1)

Step2: Compare it with the divisor;

If (it is smaller, than the divisor): ($x_1 < y$)

2.a append 0 to the quotient.

else: ($x_1 \geq y$)

proceed with division.

2.a Find the largest multiple of the divisor that fits.

2.b Append the multiplier to the quotient.

Step3: Subtract and bring down the next digit.

Step4: From now on compare the newly formed number in step3 with the divisor while repeating step2.

Step5. Repeat this step 2, 3 & 4 until all digits of the dividend ($x_1x_2x_3...x_m$) are processed.

Iteration-1: ($1 < 10$)

$$\begin{array}{r|rr} 10 & 101 & 0 \\ & 0 & \\ \hline & 10 & \end{array}$$

Iteration-2: ($10 == 10$)

$$\begin{array}{r|rr} 10 & 101 & 01 \\ & 0 & \\ \hline & 10 & \\ & -10 & \\ \hline & 0 & \end{array}$$

Iteration-3: ($01 < 10$)

$$\begin{array}{r|rr} 10 & 101 & 010 \\ & 0 & \\ \hline & 10 & \\ & -10 & \\ \hline & 01 & \\ & -0 & \\ \hline & 1 & \end{array}$$

Example - 1

Perform $1011101_2 / 110_2$; Find the quotient and remainder.

Dividend

Divisor 110

Quotient

Remainder

$$\begin{array}{r}
 \text{Dividend} \\
 \begin{array}{r}
 \downarrow \downarrow \downarrow \downarrow \downarrow \downarrow \downarrow \\
 1011101 \\
 \hline
 0 \\
 \hline
 10 \\
 -0 \\
 \hline
 101 \\
 -0 \\
 \hline
 1011 \\
 -110 \\
 \hline
 1011 \\
 -110 \\
 \hline
 1010 \\
 -110 \\
 \hline
 1001 \\
 -110 \\
 \hline
 011
 \end{array}
 \end{array}$$

$110 \times 0 = 0$
 $110 \times 1 = 110$

$0001111 = 1111$

011

$$\begin{array}{r}
 1111 \\
 110 \overline{) 1011101} \\
 \underline{-110} \\
 1011 \\
 \underline{-110} \\
 1010 \\
 \underline{-110} \\
 1001 \\
 \underline{-110} \\
 11
 \end{array}$$

Example - 2

Perform $234102_5 / 321_5$; Find the quotient and remainder.

Dividend

Divisor 321

Quotient

Remainder

234102

0

23

-0

234

-0

2341

2334

00020

00000

202

000

202

$$321 \times 0 = 0$$

$$321 \times 1 = 321$$

$$321 \times 2 = 1142$$

$$321 \times 3 = 2013$$

$$321 \times 4 = 2334$$

Exercise

4 (a) $(71)_8 / (56)_8$

4(b) $(21)_3 / (12)_3$

Solution

- 4 a) Quotient = $()_8$, Remainder = $()_8$
- 4 b) Quotient = $()_3$, Remainder = $()_3$