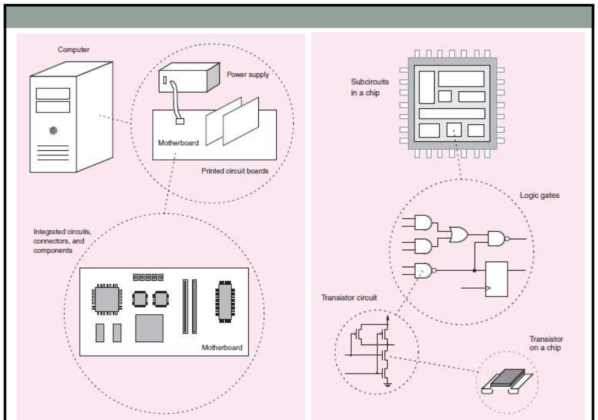
# DIGITAL DESIGN

CS/ECE/EEE/INSTR F215

Lecture slides
Sarang Dhongdi

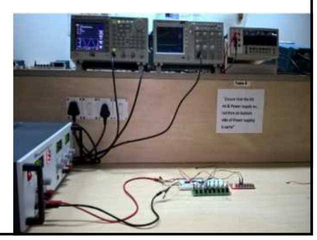
## Introduction

- Digital Systems and Binary numbers
- Boolean Algebra and Logic Gates
- Gate Level minimization
- Combinational Logic
- Synchronous Sequential logic
- Asynchronous Sequential logic
- Registers, counters, memory  and programmable logic
- Digital Integrated Circuits
- Design of Digital Systems




## Learning in the lab components

- Hardware Lab-
  - Usage of breadboard, Digital IC, LED panels, Switch boards
  - AFG (Arbitrary Signal Generator)
  - DSO (Digital Storage Oscilloscope)
- Software Lab
  - Verilog tool



## Text books

- M.Moris Mano, “Digital Design”, Pearson, 4th Edition, 2009.
- Brian Holdsworth, Clive Woods, “Digital Logic Design”, Elsevier, 4th Edition, 2008

## Reference books

- John.M.Yarbrough, “Digital Logic Design”, Cengage Learning, 2009.
- Ronald.J.Tocci, Neal.S.Widmer, Gregory.L.Moss, “Digital Systems”, 2007.
- Stephen Brown, Zvonko Vranesic, “Digital Logic with VHDL Design”, McGraw Hill,2013

## Evaluation components

| Component | Duration | Maximum Marks | Date | Remarks |
|---|---|---|---|---|
| | | Theory | | |
| Mid-Term Test | 90 Min | 60 | 11/10/2018<br>11.00 am to 12.30 pm | CB |
| Quiz-I | 30 Min | 10 | 28/08/2018<br>6.00 pm to 6.30 pm | CB |
| Quiz-II | 30 Min | 10 | 25/09/2018<br>6.00 pm to 6.30 pm | OB |
| Quiz-III | 30 Min | 10 | 23/10/2018<br>6.00 pm to 6.30 pm | CB |
| Quiz-IV | 30 Min | 10 | 20/11/2018<br>6.00 pm to 6.30 pm | OB |
| Comprehensive Examination | 3 Hrs | 100 | 08/12/2018 (FN)<br>9.00 am to 12.00 noon | CB/OB |
| | | Lab | | |
| H/w Lab Evaluation | | 50 | Regularly | OB |
| Verilog Evaluation-I | | 10 | 23/09/2018<br>10.00 am to 5.00 pm | OB |
| Verilog Evaluation-II | | 10 | 18/11/2018<br>10.00 am to 5.00 pm | OB |
| H/w Lab Comprehensive | | 30 | 01/11-21/11 | OB |

## Positional number representation

- Decimal number system – Has base or radix 10 because it uses 10 digits and the coefficients are multiplied by the power of 10.
- In general, for decimal number $a_3a_2a_1a_0.a_{-1}a_{-2}a_{-3}$, the value is calculated as
  $=10^3a_3+10^2a_2+10^1a_1+10^0a_0+10^{-1}a_{-1}+10^{-2}a_{-2}+10^{-3}a_{-3}$
- Ex. $842.45 = 10^2\times8+10^1\times4+10^0\times2+10^{-1}\times4+10^{-2}\times5$

## Binary number system

- Base or radix of 2. Only 2 possible values 0 and 1.
- Coefficients are multiplied with power of 2.
- Number 101.11 =
  $= 2^2\times1 + 2^1\times0 + 2^0\times1 + 2^{-1}\times1 + 2^{-2}\times1 = 5.75$
- In general, a number is represented in radix r (or base-r system) as follows:
- $r^na_n+r^{n-1}a_{n-1}+\ldots+r^2a_2+r^1a_1+r^0a_0+r^{-1}a_{-1}+r^{-2}a_{-2}+\ldots+r^{-m}a_{-m}$
- Here, coefficients are multiplied by power of r and coefficients range is from 0 to r-1.

## Other numbering systems

- Octal numbering system
  - Base 8
  - Numbers from 0 to 7 (Digits 8,9 do not appear)
  - Number $(123.4)_8$ =
    $= 8^2\times1 + 8^1\times2 + 8^0\times3 + 8^{-1}\times4 = (83.5)_{10}$
- Hexadecimal numbering system
  - Base 16
  - Numbers from 0 to 9 along with letters A to F
  - Number $(A3)_{16}$ =
    $= 16^1\times10 + 16^0\times3 = (163)_{10}$

## Different numbering systems

| Decimal (base 10) | Binary (base 2) | Octal (base 8) | Hexadecimal (base 16) |
|---|---|---|---|
| 00 | 0000 | 00 | 0 |
| 01 | 0001 | 01 | 1 |
| 02 | 0010 | 02 | 2 |
| 03 | 0011 | 03 | 3 |
| 04 | 0100 | 04 | 4 |
| 05 | 0101 | 05 | 5 |
| 06 | 0110 | 06 | 6 |
| 07 | 0111 | 07 | 7 |
| 08 | 1000 | 10 | 8 |
| 09 | 1001 | 11 | 9 |
| 10 | 1010 | 12 | A |
| 11 | 1011 | 13 | B |
| 12 | 1100 | 14 | C |
| 13 | 1101 | 15 | D |
| 14 | 1110 | 16 | E |
| 15 | 1111 | 17 | F |

## Number-Base conversion

- Convert $(857)_{10}$ to binary – Integer quotient Remainder
  - $857 \div 2 = 428$ 1 LSB
  - $428 \div 2 = 214$ 0
  - $214 \div 2 = 107$ 0
  - $107 \div 2 = 53$ 1
  - $53 \div 2 = 26$ 1
  - $26 \div 2 = 13$ 0
  - $13 \div 2 = 6$ 1
  - $6 \div 2 = 3$ 0
  - $3 \div 2 = 1$ 1
  - $1 \div 2 = 0$ 1 MSB
  - Result is $(1101011001)_2$

## Number-Base conversion

| | | |
|---|---|---|
| 2 | 100 | 0 |
| 2 | 50 | 0 |
| 2 | 25 | 1 |
| 2 | 12 | 0 |
| 2 | 6 | 0 |
| 2 | 3 | 1 |
| 2 | 1 | 1 |
| | 0 | |

$(100)_{10} = (1100100)_2$

| | | |
|---|---|---|
| 8 | 100 | 4 |
| 8 | 12 | 4 |
| 8 | 1 | 1 |
| | 0 | |

$= (144)_8$

| | | |
|---|---|---|
| 16 | 100 | 4 |
| 16 | 6 | 6 |
| | 0 | |

$= (64)_{16}$

## Convert $(0.265)_{10}$ into binary, octal and hex

| Binary | Octal | Hex |
|---|---|---|
| $.265 \times 2$ | $.265 \times 8$ | $.265 \times 16$ |
| $0.530 \times 2$ | $2.120 \times 8$ | $4.240 \times 16$ |
| $1.060 \times 2$ | $0.960 \times 8$ | $3.840 \times 16$ |
| $0.120 \times 2$ | $7.680 \times 8$ | $D.440 \times 16$ |
| $0.240 \times 2$ | $5.440 \times 8$ | $7.040 \times 16$ |
| $0.480$ | $3.520$ | $0.640$ |
| $(0.265)_{10} = (0.0100)_2$ | $(0.20753)_8$ | $(0.43D70)_{16}$ |

Expressed upto 4 digits of accuracy. OR continue till the fraction part becomes zero.

## Octal or hexadecimal to Binary

| Octal | Binary |
|---|---|
| 0 | 000 |
| 1 | 001 |
| 2 | 010 |
| 3 | 011 |
| 4 | 100 |
| 5 | 101 |
| 6 | 110 |
| 7 | 111 |

| HD | Binary | HD | Binary |
|---|---|---|---|
| 0 | 0000 | 8 | 1000 |
| 1 | 0001 | 9 | 1001 |
| 2 | 0010 | A | 1010 |
| 3 | 0011 | B | 1011 |
| 4 | 0100 | C | 1100 |
| 5 | 0101 | D | 1101 |
| 6 | 0110 | E | 1110 |
| 7 | 0111 | F | 1111 |

## Octal to Binary

$$(110\ \ 001\ \ 011\ \ 100)_2 = (6\ \ 1\ \ 3\ \ 4)_8$$

$$(100\ \ 001\ \ 010\ \ 100\ \ .010) = (4\ \ 1\ \ 2\ \ 4\ \ .2)$$

To find the octal representation of a string of binary digits it is divided into groups of three, starting from the binary point and proceeding to left and to right. Corresponding octal digit is then assigned to the group.

$$(4\ \ 3\ \ 2\ \ .7)_8 = (100\ \ 011\ \ 010\ \ .111)_2$$

Octal numbers can also be converted to binary by replacing each octal digit with the corresponding three binary digits from the conversion table.

## Hexadecimal to Binary

$$(1011\ \ 1010\ \ 0011\ \ .0010)_2 = (B\ \ A\ \ 3\ \ .2)_{16}$$

Grouping of 4 bits

$$(4\ \ F\ \ C\ \ 2)_{16} = (0100\ \ 1111\ \ 1100\ \ 0010)_2$$

## Binary addition and subtraction

| Augend | Addend | Sum | Carry |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

| Minuend | Subtrahend | Difference | Borrow |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |

| | | $2^3$ | $2^2$ | $2^1$ | $2^0$ | |
|---|---|---|---|---|---|---|
| Augend | | 1 | 0 | 1 | 1 | 11 |
| Addend | | 0 | 1 | 1 | 1 | +7 |
| Sum | 1 | 0 | 0 | 1 | 0 | 18 |
| Carries | 1 | 1 | 1 | 1 | | |

| | | $2^3$ | $2^2$ | $2^1$ | $2^0$ | |
|---|---|---|---|---|---|---|
| Minuend | | 1 | 1 | 0 | 0 | 12 |
| Subtrahend | | 0 | 0 | 1 | 1 | −3 |
| Difference | | 1 | 0 | 0 | 1 | +9 |
| Borrows | | | 1 | 1 | | |

| | | $2^3$ | $2^2$ | $2^1$ | $2^0$ | |
|---|---|---|---|---|---|---|
| Minuend | | 0 | 0 | 1 | 1 | 3 |
| Subtrahend | | 1 | 1 | 0 | 0 | −12 |
| Difference | | 0 | 1 | 1 | 1 | −9 |
| Borrows | 1 | 1 | | | | |

## Complements

- Two types of complements:
  - Diminished Radix Complement ((r – 1)'s complement)
  - Radix Complement (r's complement)

## Complements

- Diminished Radix Complement ((r – 1)'s complement)
- In base-r, for a number N having n digits, (r – 1)'s complement is **($r^n$ – 1) – N.**
- Radix Complement (r's complement)
- In base-r, for a number N having n digits, r 's complement is **$r^n$ – N.**

## Complements

- 9's complement of 654 is 999-654 = 345.
- 10's complement of 654 is then 346.
- 1's complement of 0110 is 1001
- 2's complement is then 1010.
- Similarly complements for octal and hexadecimal numbers.

## Subtraction using complements – Unsigned numbers

- The subtraction of two n-digit unsigned numbers M-N in base r
- - Add the minuend M to the r's complement of subtrahend N.
- - If M ≥ N, then the sum will produce end-carry – discard it.
- If M<N, the sum does not produce end-carry. Take r's complement of sum and place negative sign in front of it.

## Using (r – 1)'s complement

- In (r-1)'s complement –
- In case of end carry – remove it and add 1 to the sum.
  - - End-around carry
- In case of no carry, take (r-1)'s complement and put negative sign in front of the result.

## Signed binary numbers

- Negative binary numbers can be represented as -
  - Signed magnitude representation
  - Signed 1's complement representation
  - Signed 2's complement representation

## Signed magnitude representation

- In "unsigned binary numbers", symbol "+" or "-" is used for representing positive or negative numbers.
- In "Signed binary numbers" leftmost bit is used to represent the positive number (bit 0) or negative number (bit 1).
- For example, in unsigned binary
- Number 01001 is +9 and number 11001 is +25.
- Whereas, in signed binary number system
- Number 01001 is +9 and number 11001 is -9.

## Signed magnitude representation

| Decimal | Signed magnitude |
|---|---|
| +7 | 0111 |
| +6 | 0110 |
| +5 | 0101 |
| +4 | 0100 |
| +3 | 0011 |
| +2 | 0010 |
| +1 | 0001 |
| +0 | 0000 |

| Decimal | Signed magnitude |
|---|---|
| -7 | 1111 |
| -6 | 1110 |
| -5 | 1101 |
| -4 | 1100 |
| -3 | 1011 |
| -2 | 1010 |
| -1 | 1001 |
| -0 | 1000 |

## Signed complement form

- Negative numbers are represented by complement.
- Signed 1's complement and signed 2's complement

| Decimal | Signed 1's |
|---|---|
| -7 | 1000 |
| -6 | 1001 |
| -5 | 1010 |
| -4 | 1011 |
| -3 | 1100 |
| -2 | 1101 |
| -1 | 1110 |
| -0 | 1111 |

| Decimal | Signed 2's |
|---|---|
| -8 | 1000 |
| -7 | 1001 |
| -6 | 1010 |
| -5 | 1011 |
| -4 | 1100 |
| -3 | 1101 |
| -2 | 1110 |
| -1 | 1111 |

## Binary arithmetic – signed complement

- Addition and subtraction in the 2's complement system are both carried out as additions.
- Subtrahends are regarded as negative numbers and are converted to their 2's complement form. They are then added to the positive minuend.
- When adding two negative numbers they are both converted to their 2's complement form before addition takes place.

## Binary codes for decimal digits

| Decimal Digit | BCD 8421 | 2421 | Excess-3 | 8, 4, −2, −1 |
|---|---|---|---|---|
| 0 | 0000 | 0000 | 0011 | 0000 |
| 1 | 0001 | 0001 | 0100 | 0111 |
| 2 | 0010 | 0010 | 0101 | 0110 |
| 3 | 0011 | 0011 | 0110 | 0101 |
| 4 | 0100 | 0100 | 0111 | 0100 |
| 5 | 0101 | 1011 | 1000 | 1011 |
| 6 | 0110 | 1100 | 1001 | 1010 |
| 7 | 0111 | 1101 | 1010 | 1001 |
| 8 | 1000 | 1110 | 1011 | 1000 |
| 9 | 1001 | 1111 | 1100 | 1111 |
| | 1010 | 0101 | 0000 | 0001 |
| Unused | 1011 | 0110 | 0001 | 0010 |
| bit | 1100 | 0111 | 0010 | 0011 |
| combi- | 1101 | 1000 | 1101 | 1100 |
| nations | 1110 | 1001 | 1110 | 1101 |
| | 1111 | 1010 | 1111 | 1110 |

## ASCII Character Code

| | $b_7b_6b_5$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $b_4b_3b_2b_1$ | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| 0000 | NUL | DLE | SP | 0 | @ | P | ` | p |
| 0001 | SOH | DC1 | ! | 1 | A | Q | a | q |
| 0010 | STX | DC2 | " | 2 | B | R | b | r |
| 0011 | ETX | DC3 | # | 3 | C | S | c | s |
| 0100 | EOT | DC4 | $ | 4 | D | T | d | t |
| 0101 | ENQ | NAK | % | 5 | E | U | e | u |
| 0110 | ACK | SYN | & | 6 | F | V | f | v |
| 0111 | BEL | ETB | ' | 7 | G | W | g | w |
| 1000 | BS | CAN | ( | 8 | H | X | h | x |
| 1001 | HT | EM | ) | 9 | I | Y | i | y |
| 1010 | LF | SUB | * | : | J | Z | j | z |
| 1011 | VT | ESC | + | ; | K | [ | k | { |
| 1100 | FF | FS | , | < | L | \ | l | \| |
| 1101 | CR | GS | – | = | M | ] | m | } |
| 1110 | SO | RS | . | > | N | ∧ | n | ~ |
| 1111 | SI | US | / | ? | O | – | o | DEL |

## Gray Code

| Gray Code | Decimal Equivalent |
|---|---|
| 0000 | 0 |
| 0001 | 1 |
| 0011 | 2 |
| 0010 | 3 |
| 0110 | 4 |
| 0111 | 5 |
| 0101 | 6 |
| 0100 | 7 |
| 1100 | 8 |
| 1101 | 9 |
| 1111 | 10 |
| 1110 | 11 |
| 1010 | 12 |
| 1011 | 13 |
| 1001 | 14 |
| 1000 | 15 |