# EE 200- Digital Logic Circuit Design
# 1.7 Binary Codes

Dr. Muhammad Mahmoud

جامعة الملك فهد للبترول والمعادن
King Fahd University of Petroleum & Minerals

September 10, 2013

# Entry Question

- What dose code mean?
- Can you give example of codes?

# Objectives

1. Binary Codes
   - Decimal Codes
   - Gray Code
   - ASCII Code
   - Error-Detecting Code

# Why Coding?

- Digital systems and circuits can only store one of two states, “0” and “1”.
- One bit can represent two elements only!
- With n bits, we can produce $2^n$ different combinations.
- To represent m elements, we need n bits, where $2^n \geq m$.

# Coding

Example:

- How many bits are needed to represent the primary and secondary compass points? (e.g., N, S, E, W, NE, SE, NW, SW)

| | |
|---|---|
| N | 000 |
| NE | 001 |
| E | 011 |
| SE | 010 |
| S | 110 |
| SW | 111 |
| W | 101 |
| NW | 100 |

# Binary-Coded Decimal Code

- How many bits do we need to code the decimal digits?
- What if we dedicated four bits for every decimal digit (BCD)?

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

- $64 \rightarrow$ 0110 0100 instead of $(1000000)_2$
- $185 \rightarrow$ 0001 1000 0101 instead of $(10111001)_2$

# BCD Addition

- If $0 \leq$ Sum $\leq 9$ then, sum in BCD $=$ sum in binary.
- If $10 \leq$ Sum $\leq 19$ then, sum in BCD consists of 8 bits which is not equal to the sum in binary. Corrected by adding 0110 to the binary sum.

| | | | |
|---|---|---|---|
| $4 \rightarrow$ | 0100 | $4 \rightarrow$ | 0100 |
| $+5 \rightarrow$ | 0101 | $+8 \rightarrow$ | 1000 |
| $9 \rightarrow$ | 1001 | $12 \rightarrow$ | 1100 |
| | | | $+0110$ |
| | | | 0001 0010 |

# Decimal Arithmetic

- Similar to binary, the sign in signed BCD is represented by the most significant digit.
- "0000" $\rightarrow$ +ve, "1001" $\rightarrow$ -ve.
- Arithmetic operation using 10's complement.
- Solve this in BCD : (+375)+(-240).

  the 10's complement of (-)240 is (9)760.

$$\begin{array}{rr} 0 & 375 \\ +9 & \underline{760} \\ \cancel{1}0 & 135 \end{array}$$

# Decimal Codes

| Decimal Digit | BCD 8421 | Excess-3 | 84-2-1 | 2421 |
|---|---|---|---|---|
| 0 | 0000 | 0011 | 0000 | 0000 |
| 1 | 0001 | 0100 | 0111 | 0001 |
| 2 | 0010 | 0101 | 0110 | 0010 |
| 3 | 0011 | 0110 | 0101 | 0011 |
| 4 | 0100 | 0111 | 0100 | 0100 |
| 5 | 0101 | 1000 | 1011 | 1011 |
| 6 | 0110 | 1001 | 1010 | 1100 |
| 7 | 0111 | 1010 | 1001 | 1101 |
| 8 | 1000 | 1011 | 1000 | 1110 |
| 9 | 1001 | 1100 | 1111 | 1111 |

# Gray Code

- To represent analog data that have been converted to digital.
- Switching between two consecutive numbers requires one bit switching only.

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

# Example

| Direction | Gray Code | Decimal Code |
|---|---|---|
| N | 000 | 000 |
| NE | 001 | 001 |
| E | 011 | 010 |
| SE | 010 | 011 |
| S | 110 | 100 |
| SW | 111 | 101 |
| W | 101 | 110 |
| NW | 100 | 111 |

# ASCII Code

- **A**merican **S**tandard **C**ode for **I**nformation **I**nterchange.
- 26 UPPAER + 26 lower case letters + 10 numerals + 32 special characters + 34 control characters = 128 $=2^7$
- $\overbrace{b_7\,b_6\,b_5}^{\text{column}}\ \overbrace{b_4\,b_3\,b_2\,b_1}^{\text{row}}$

# ASCII 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 |

# Error-Detecting Code

- A method for error detection in binary numbers transmission $\rightarrow$ parity bit.
- An extra bit included with the binary number to make the total number of 1's transmitted either odd or even.
- Example:

| Sequence | With Even Parity | With Odd Parity |
|---|---|---|
| 1000001 | 01000001 | 11000001 |
| 1010100 | 11010100 | 01010100 |

# Summary

# Next Lecture

Binary Logic