SE311: Design of Digital Systems

Lecture 1: Introduction to Digital Systems

Dr. Samir Al-Amer

(Term 041)

Design of Digital Systems

- Grading policy
- Course Outlines
- Introduction to digital Systems
- Numbering systems

Grading Policy

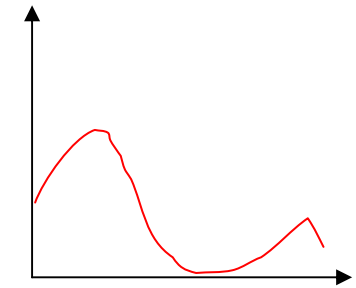
Major Exam	25
Quizzes	17
Lab and projects	20
HW& Attendance	8
Final	30

- Quizzes:
  - Pop quizzes: Short (0.5 % each)
  - Regular (three 5% each) Announced
- Attendance: -1 % for each unexcused absence.
- HW:
  - submit in class
  - Late submissions are not accepted
- Major Exam: Date will be announced

Course Outlines

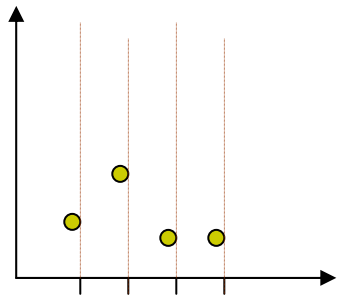
- Number Systems
- Binary Logic
- Gate-Level Minimization
- Combinatorial Logic
- Synchronous Sequential logic
- Registers and Counters
- Memory and programming Logic

Classification of Signals

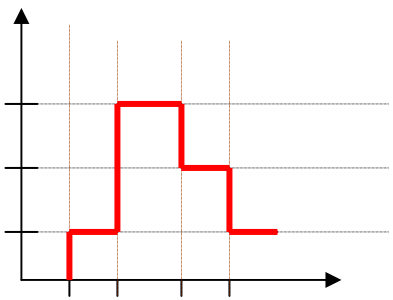


Continuous-time,nonquantized

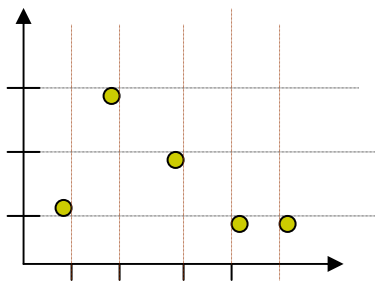
(Analog signal)



Discrete-time,nonquantized



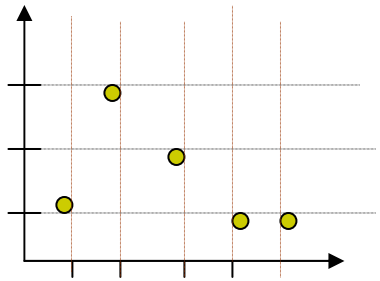
Continuous-time,quantized

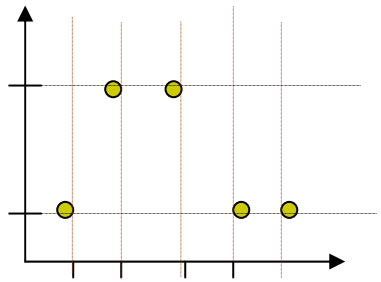


Discrete-time,quantized

(Digital Signal)

Classification of Signals
Digital Signals



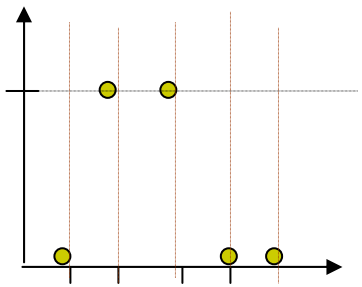


Digital Signal: is a discrete-time, quantized signal

Binary Signal: It is a digital signal with two possible values (1 / 0, ON/OFF)

Classification of Signals

Binary Signals

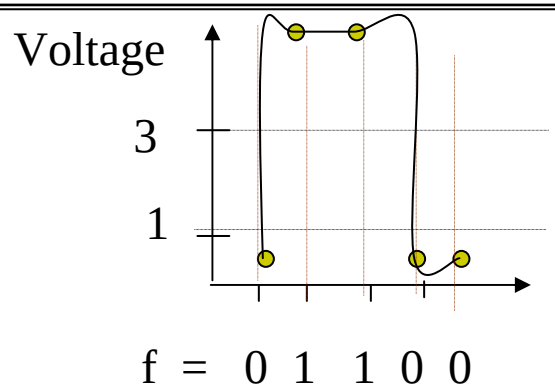


ON	1	High
OFF	0	Low

Binary Signals: They have two possible values

Classification of Signals

Binary Signals



ON	1	High
OFF	0	Low

The signals in a typical electronic circuit are *analog signals* but they are coded in order to use them as *binary signals*:

f =1 (ON) if voltage > 3

f = 0 (OFF) if voltage < 1

Examples of Analog and Digital systems

Digital Watch

Analog Watch

Digital Systems

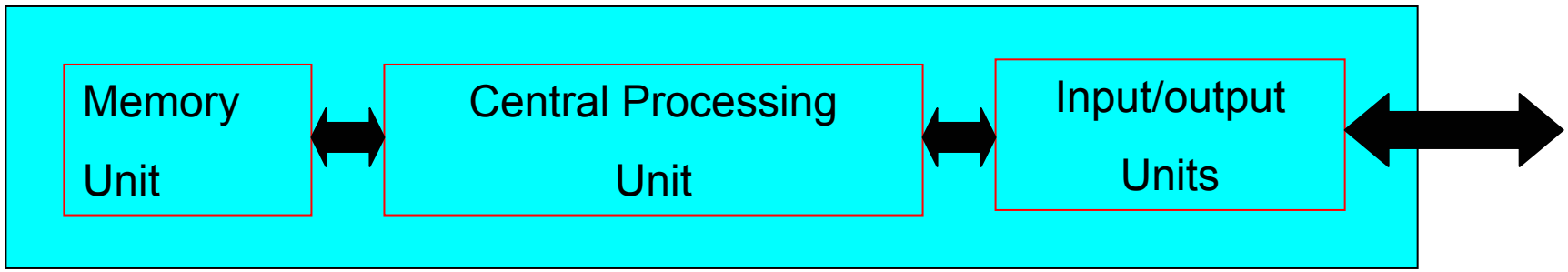
A digital system is a system that manipulates discrete elements of information that are represented internally in binary form

Examples:

- Computers
- Calculators
- Digital cammera

Example of Digital System

The computer



- **Memory Unit**: stores the programs and data
- **Central Processing unit (CPU):** Performs the operations specified by the program
- **Input/output Unit (I/O unit)** : communicate with the external devices.

Advantages of Digital System

- Digital systems are easier to design
- Information storage is easy
- Accuracy and precision are greater
- Flexibility (operations can be programmed)
- Low cost
- Reliability (error detection and correction is easy)

Example

Problem: Design a binary adder that adds two 8-bit numbers

To be able to solve this problem you should be able to do:

- Add two Binary numbers
- Develop the logic that produces the result
- Simplify the resulted logic
- Select the required Hardware
- Implement and verify the circuit

Numbering System

Four Numbering system

- Decimal
- Binary
- Octal
- Hexadecimal

Numbering System

Decimal

Base 10

Digits: 0,1,2,3,4,5,6,7,8,9

Binary

Base 2

Digits: 0,1

Octal

Base 8

Digits: 0,1,2,3,4,5,6,7

Hexadecimal

Base 16

Digits: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

Decimal Numbering System

Base =10

Digits: {0,1,2,3,4,5,6,7,8,9}

$X = (3\ 5\ 6.7\ 2)_{10} = 3x10^{2} + 5x10^{1} + 6x10^{0} + 7\ x10^{-1} + 2x10^{-2}$

Digits	3	5	6	7	2
Weight	100	10	1	0.1	0.01
Value	300	50	6	0.7	0.02

Binary Numbering System

Base =2

Digits: {0,1}

$$Y = (1\ 0\ 1.1\ 1)_2 = 1x\ 2^2 + 0x\ 2^1 + 1x\ 2^0 + 1\ x\ 2^{-1} + 1x\ 2^{-2}$$

$$= 5.75$$

Digits	1	0	1	1	1
Weight	4	2	1	0.5	0.25
Value	4	0	1	0.5	0.25

Octal Numbering System

Base =8

Digits: {0,1,2,3,4,5,6,7}

$X = (2\ 4\ 6.3)_8 = 2x8^2 + 4x8^1 + 6x8^0 + 3\ x8^{-1} = 166.375$

Digits	2	4	6 .	3
Weight	64	8	1	0.125
Value	128	32	6	0.375

Hexadecimal Numbering System

Base =16

Digits: {0,1,2,3,4,5,6,7,8,9, A, B, C, D, E, F}

$X = (2\ A\ 6.E)_{16} = 2x16^2 + 10x16^1 + 6x16^0 + 14\ x16^{-1}$

Digits	2	A	6 .	E
Weight	256	16	1	0.0625
Value	512	160	6	0.875

r - Numbering System

Base = r

Digits: {0,1,…,r-1}

$X = (2\ 1\ 0.1)_r = 2x\ r^2 + 1x\ r^1 + 0x\ r^0 + 1\ x\ r^{-1}$

$$\left(b_m\ b_{m-1} b_{m-2}\ \ldots b_0 . b_{-1}\ b_{-2} \ldots b_{-n}\right)_r = \sum_{k=-n}^{m} b_k\ \ r^k$$

Notation

Bit: a binary digit

Byte: a group of eight bits

K Byte (kilo bytes) $= 2^{10}$ bytes = 1024 byte

M Byte (mega bytes) $= 2^{20}$ bytes ≈ Million bytes

G Byte (gega bytes) $= 2^{30}$ bytes ≈ Billion bytes

Converting (Binary, octal, hexadecimal) to decimal

$$\left(b_m\ b_{m-1} b_{m-2}\ \ldots b_0 . b_{-1}\ b_{-2} \ldots b_{-n}\right)_r = \sum_{k=-n}^{m} b_k\ \ r^k$$

$r = 2$ $(Binary)$

$r = 8$ $(Octal)$

$r = 16$ $(Hexadecimal)$