CS M51A
Logic Design of Digital Systems
Winter 2021

Some slides borrowed and modified from:
M.D. Ercegovac, T. Lang and J. Moreno, Introduction to Digital Systems.
D. Patterson and J. Hennessy, Computer Organization and Design

Introduction

- **Instructor:** Prof. Omid Abari
  http://www.cs.ucla.edu/~omid

- Feel free to reach me whenever you need a help. omid@cs.ucla.edu
- Please put CS M51A in the email's subject

Introduction

- **Teaching Assistants**:
  - Sicheng Jia (jsicheng@cs.ucla.edu)
  - Xiaojian Ma (xm@cs.ucla.edu)

- They are expert in this course.

Introduction

- **Lectures:**
  - MW 10:00 am - 11:50 am, Zoom (link is posted on CCLE)

- **Discussions:**
  - F 10:00 am - 11:50 am, Zoom (link is posted on CCLE)

Introduction

- **Office hours (**Sicheng Jia**):**
  - Tue 8:45-9:45am, and Thu 12:00-1:00pm, Zoom (link is posted on CCLE)
- **Office hours (**Xiaojian Ma**):**
  - Tue 4:00 pm - 6:00 pm, Zoom (link is posted on CCLE)
- **Office hours (**Prof. Omid Abari**):**
  - Mon 12:00 pm - 1:00 pm, Zoom (link is posted on CCLE)

Introduction

- **Textbook (Optional):**
  - M.D. Ercegovac, T. Lang and J. Moreno, Introduction to Digital Systems, John Wiley & Sons, New York, 1999. Available as reader at UCLA Bookstore and on the Web.

Introduction

- **Textbook (Optional):**
  - M.D. Ercegovac, T. Lang and J. Moreno, Introduction to Digital Systems, John Wiley & Sons, New York, 1999. Available as reader at UCLA Bookstore and on the Web.

- **Course material:** Lectures, assignments and solutions will be posted on CCLE or/and Gradescope

Introduction

- **Class Communication:**
  - Important class announcements will be done through online class forum on Piazza. If you have any questions regarding class materials, they also need to be asked on Piazza. Please make sure to sign up for CSM51A Piazza forum at http://piazza.com/ucla/winter2021/csM51A

Introduction

- **Grading:**
  - 10 assignments, each is roughly 10% of your final grade. (No midterm, No Final Exam)

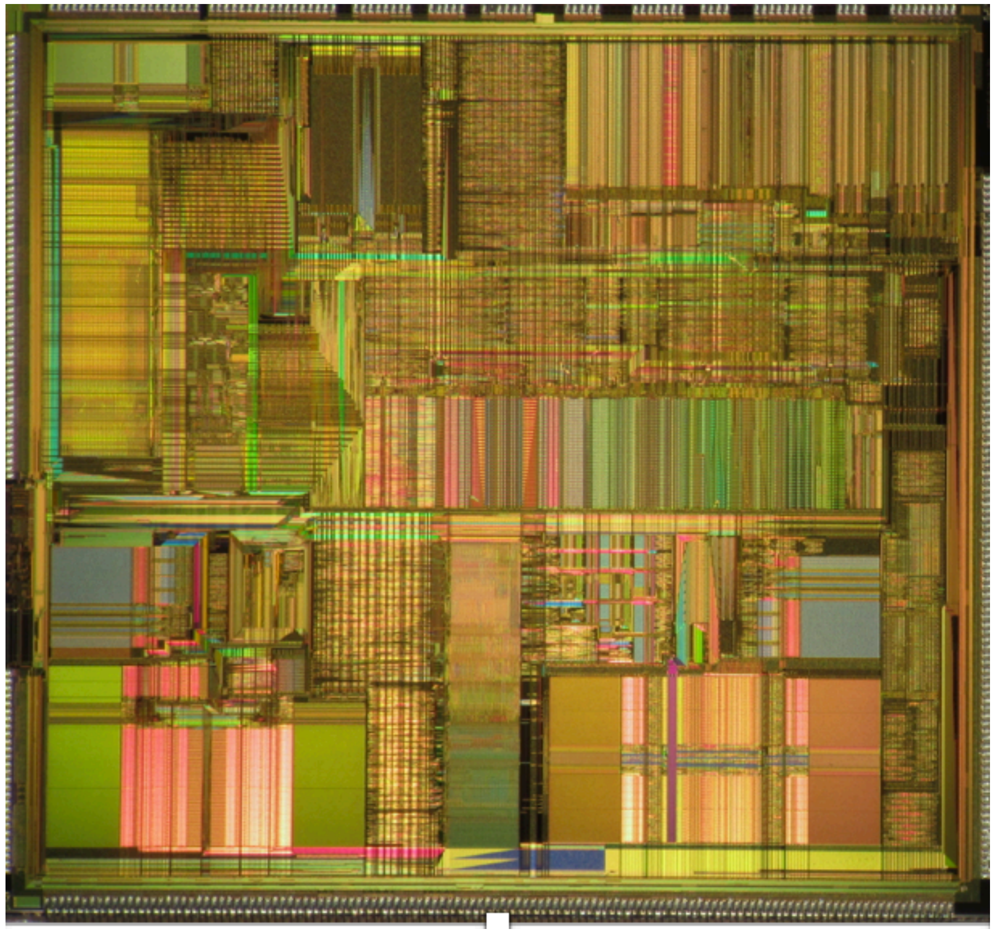
Introduction

- **Assignments:**
  - Assignments are due on **Wednesdays 10 am PT**.
  - Late work will not be accepted. In exceptional circumstances, arrangements must be made in advance of the due date for an extension.
  - You must complete the assignments **entirely on your own**. You are NOT allowed to discuss your solutions with others or see another student's solutions.
  - Gradescope is used to submit assignments.

Introduction

- **Academic Honesty:**
  - We expect all students to follow the UCLA Student Conduct Code. This code prohibits cheating, fabrication, multiple submissions, and facilitating academic dishonesty. You can find further information about this code at the Student Guide to Academic Integrity. The Office of the Dean of Students offers a workshop on academic integrity if you wish to understand UCLA's policies on this issue more thoroughly.

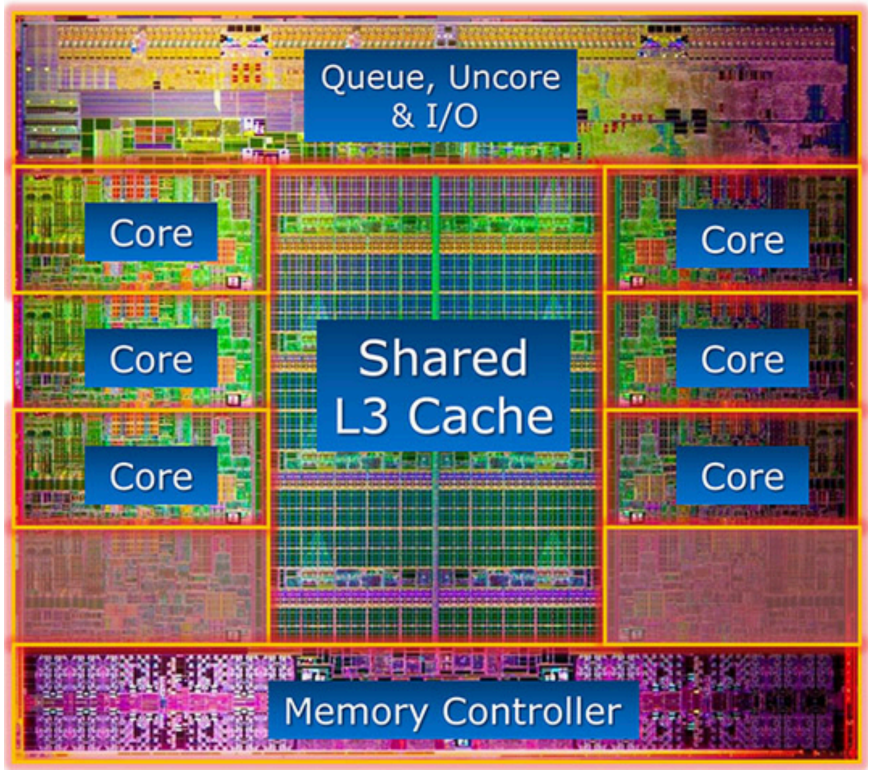
What is this?



a) carpet b) painting c) CPU d) Billions of transistors e) Piece of wood

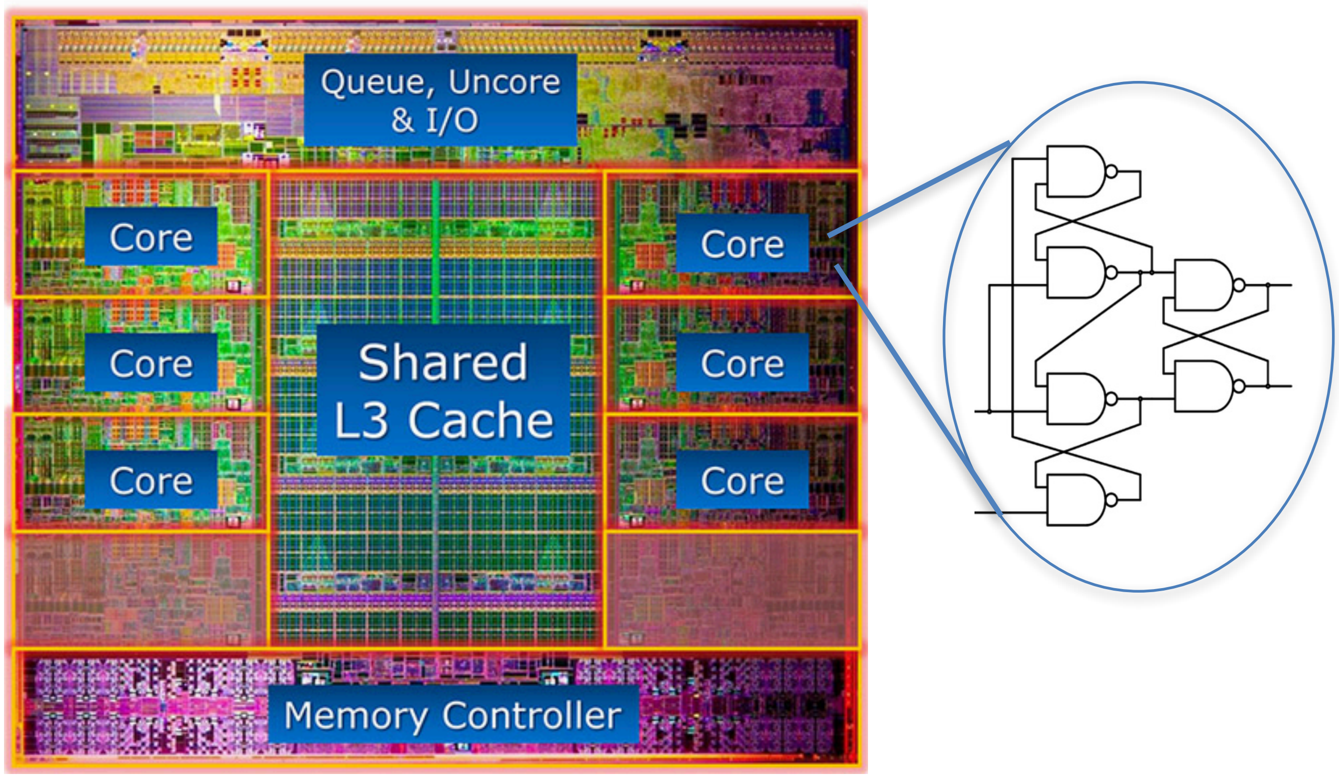
CPU

CPU

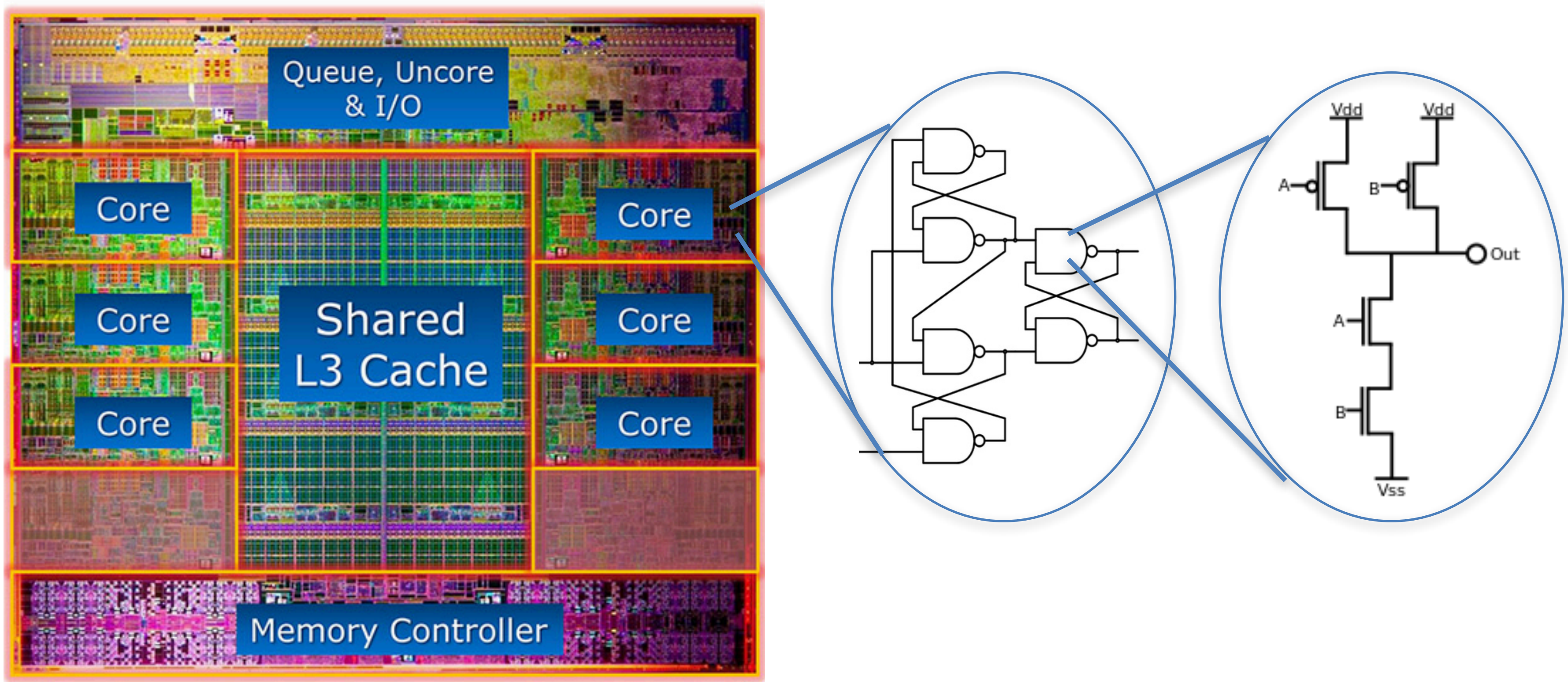


Queue, Uncore & I/O
Core
Core
Core
Shared L3 Cache
Core
Core
Core
Memory Controller

CPU


Queue, Uncore & I/O
Core
Core
Core
Shared L3 Cache
Core
Core
Core
Memory Controller

CPU



Queue, Uncore & I/O
Core
Core
Core
Core
Core
Core
Shared L3 Cache
Memory Controller
Vdd
Vdd
A
B
Out
A
B
Vss

INTRODUCTION TO DIGITAL SYSTEMS

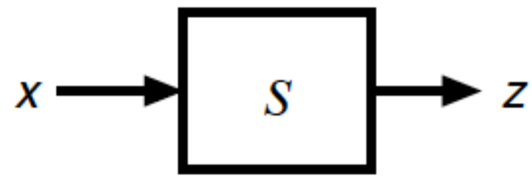
- DESCRIPTION AND DESIGN OF DIGITAL SYSTEMS
- FORMAL BASIS: SWITCHING ALGEBRA
- IMPLEMENTATION: MODULES (ICs) AND NETWORKS
- IMPLEMENTATION OF ALGORITHMS IN “HARDWARE”
- COURSE EMPHASIS: CONCEPTS, ANALYSIS AND DESIGN
- Follow-on courses:
  - Digital Lab
  - Computer Architecture
  - Computer Architecture Lab
  - Digital Design - Advanced Topics

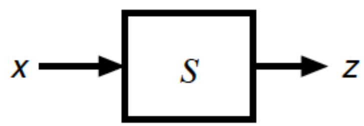
OVERVIEW

- WHAT IS A DIGITAL SYSTEM?
- HOW IT DIFFERS FROM AN ANALOG SYSTEM?
- WHY ARE DIGITAL SYSTEMS IMPORTANT?
- BASIC TYPES OF DIGITAL SYSTEMS:
  COMBINATIONAL AND SEQUENTIAL
- SPECIFICATION AND IMPLEMENTATION OF DIGITAL SYSTEMS
- ANALYSIS AND DESIGN OF DIGITAL SYSTEMS

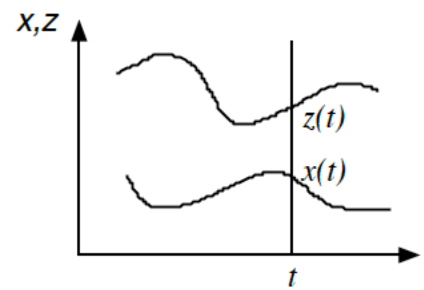
What is a system?

What is a system?

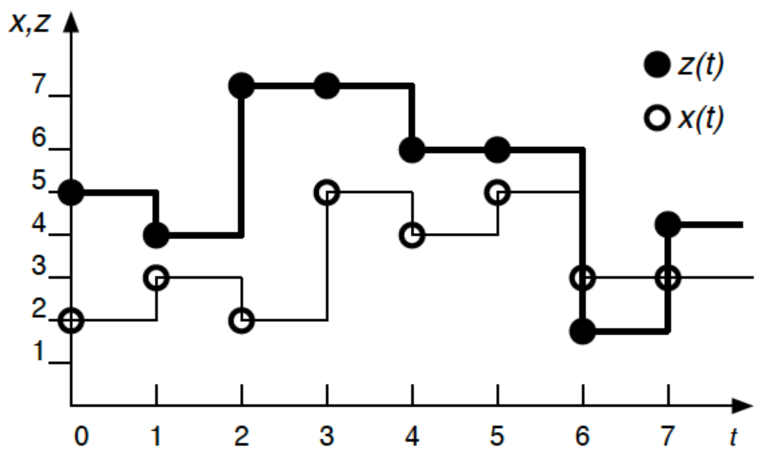




(a)



(b)



t	0	1	2	3	4	5	6	7
$x(t)$	2	3	2	5	4	5	3	3
$z(t)$	5	4	7	7	6	6	2	4

WHAT IS DIGITAL

DIGITAL SYSTEMS

– inputs and outputs:
  finite number of discrete values

ANALOG SYSTEMS

– inputs and output values
  from a continuous (infinite) set

Example: digital vs. analog scale for measuring weights

MAIN USE OF DIGITAL SYSTEMS:

- INFORMATION PROCESSING (text, audio, visual, video)
- TRANSMISSION (communication)
- STORAGE

WHY DIGITAL

1. FOR BOTH NUMERICAL AND NONNUMERICAL INFORMATION PROCESSING

2. INFORMATION PROCESSING CAN USE A GENERAL-PURPOSE SYSTEM (a *computer*)

3. DIGITAL REPRESENTATION:

   – vector of signals with just two values (*binary signals*)

   Example:

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

   – All signals binary

   – Simple devices to process binary signals:

   (SWITCHES with two STATES: open and closed).

WHY DIGITAL (cont.)

4. DIGITAL SIGNALS INSENSITIVE TO VARIATIONS OF COMPONENT PARAMETER VALUES

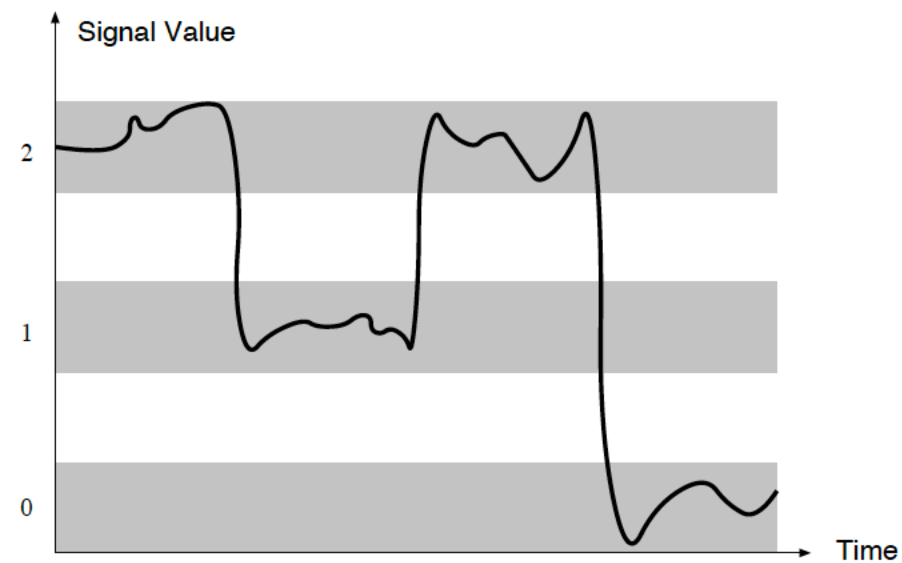


Figure 1.2: Separation of digital signal values.

WHY DIGITAL (cont.)

5. Numerical digital systems can be made MORE ACCURATE by simply increasing the number of digits used in the representation.

6. PHENOMENAL ADVANCES OF MICROELECTRONICS TECHNOLOGY:

   - Possible to fabricate extremely complex digital systems, which are small, fast, and cheap

   - Digital systems built as *integrated circuits* composed of a large number of very simple devices

7. DIFFERENT IMPLEMENTATIONS OF SYSTEMS WHICH TRADE-OFF SPEED AND AMOUNT OF HARDWARE (COST)

Example:

– add two integers represented by six decimal digits

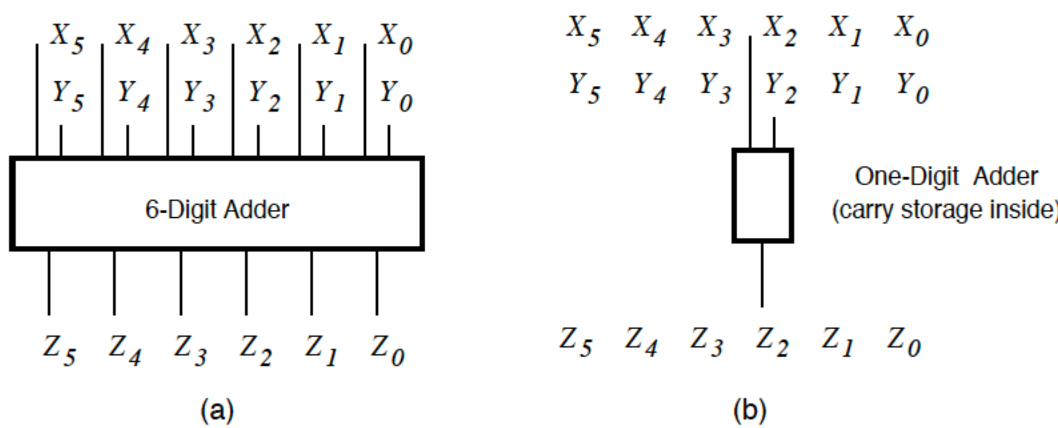


Figure 1.3: Six-digit adder: a) Parallel implementation. b) Serial implementation.

MIXED ANALOG/DIGITAL SYSTEMS

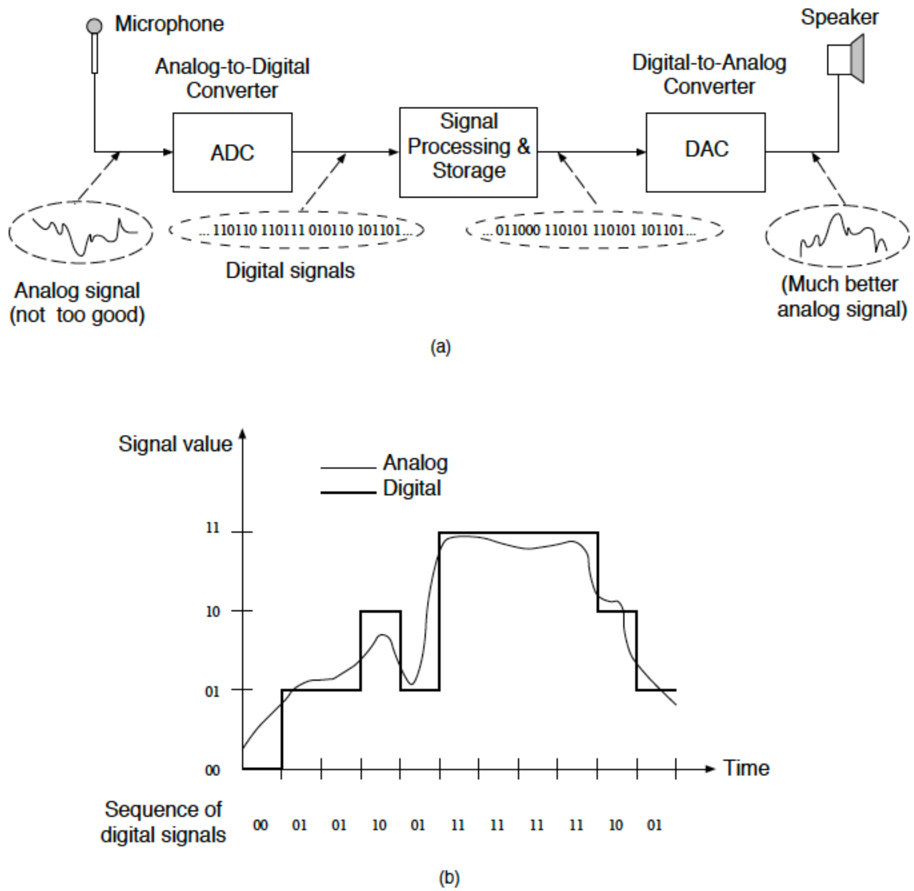


Figure 1.4: a) A system with analog and digital signals. b) Analog-to-digital conversion.

HIGH-LEVEL SPECIFICATION

- SET OF VALUES FOR THE INPUT, *input set;*
- SET OF VALUES FOR THE OUTPUT, *output set;* and
- SPECIFICATION OF THE *input-output function.*

INPUT AND OUTPUT SETS

$$\{UP,\ DOWN,\ LEFT,\ RIGHT,\ FIRE\}$$

$$\{x \mid (5 \leq x \leq 10^4) \text{ and } (x \bmod 3 = 0)\}$$

Examples of vectors

Vector type		Example
Digit	$\underline{x} = (x_{n-1}, x_{n-2}, \ldots, x_0)$ $x_i \in \{0, 1, 2, \ldots, 9\}$	$\underline{x} = (7, 0, 6, 3)$
Character	$\underline{c} = (c_{n-1}, c_{n-2}, \ldots, c_0)$ $c_i \in \{\ , A, B, \ldots, Z\}$	$\underline{c} = (B, O, O, K)$
Set	$\underline{s} = (s_{n-1}, s_{n-2}, \ldots, s_0)$ $s_i \in \{\text{red},\ \text{blue},\ \text{white}\}$	$\underline{s} = (\text{red}, \text{blue}, \text{blue})$
Bit	$\underline{y} = (y_{n-1}, y_{n-2}, \ldots, y_0)$ $y_i \in \{0, 1\}$	$\underline{y} = (1, 1, 0, 1, 0, 0)$ $\underline{y} = 110100$

INPUT-OUTPUT FUNCTION

1. *TABLE*

x	z
A	65
B	66
C	67
D	68
E	69

2. *ARITHMETIC EXPRESSION*

$$z = 3x + 2y - 2$$

3. *CONDITIONAL EXPRESSION*

$$z = \begin{cases} a + b & \textbf{if} \quad c > d \\ a - b & \textbf{if} \quad c = d \\ 0 & \textbf{if} \quad c < d \end{cases}$$

4. *LOGICAL EXPRESSION*

$$
\begin{aligned}
z \;=\; & (SWITCH1 = CLOSED) \textbf{ and } (SWITCH2 = OPEN) \\
& \textbf{or } (SWITCH3 = CLOSED)
\end{aligned}
$$

5. *COMPOSITION OF SIMPLER FUNCTIONS*

$$max(v, w, x, y) = GREATER(v, GREATER(w, GREATER(x, y)))$$

in which

$$GREATER(a, b) = \begin{cases} a \ \textbf{if} \ \ a > b \\ b \ \textbf{otherwise} \end{cases}$$

Example

Inputs: $\underline{x} = (x_3, x_2, x_1, x_0)$,
$x_i \in \{\mathrm{A}, \mathrm{B}, \ldots, \mathrm{Z}, \mathrm{a}, \mathrm{b}, \ldots, \mathrm{z}\}$
$y \in \{\mathrm{A}, \mathrm{B}, \ldots, \mathrm{Z}, \mathrm{a}, \mathrm{b}, \ldots, \mathrm{z}\}$
$k \in \{0, 1, 2, 3\}$

Outputs: $\underline{z} = (z_3, z_2, z_1, z_0)$,
$z_i \in \{\mathrm{A}, \mathrm{B}, \ldots, \mathrm{Z}, \mathrm{a}, \mathrm{b}, \ldots, \mathrm{z}\}$

Function: $z_j = \begin{cases} x_j & \textbf{if} \quad j \neq k \\ y & \textbf{if} \quad j = k \end{cases}$

Input: $\underline{x} = (\mathrm{C,A,S,E})$, $y = \mathrm{R}$, $k = 1$
Output: $\underline{z} = (\mathrm{C,A,R,E})$

COMBINATIONAL AND SEQUENTIAL SYSTEMS

- DIGITAL SYSTEMS - TWO CLASSES:
- *COMBINATIONAL SYSTEMS*

$$z(t) = F(x(t))$$

  – no memory, the output does not depend on previous inputs

- *SEQUENTIAL SYSTEMS*

$$z(t) = F(x(0,t))$$

  $x(0,t)$: input sequence from time 0 to time t

  – $z(t)$ depends also on previous inputs - the system has MEMORY

COMBINATIONAL AND SEQUENTIAL SYSTEMS (cont.)

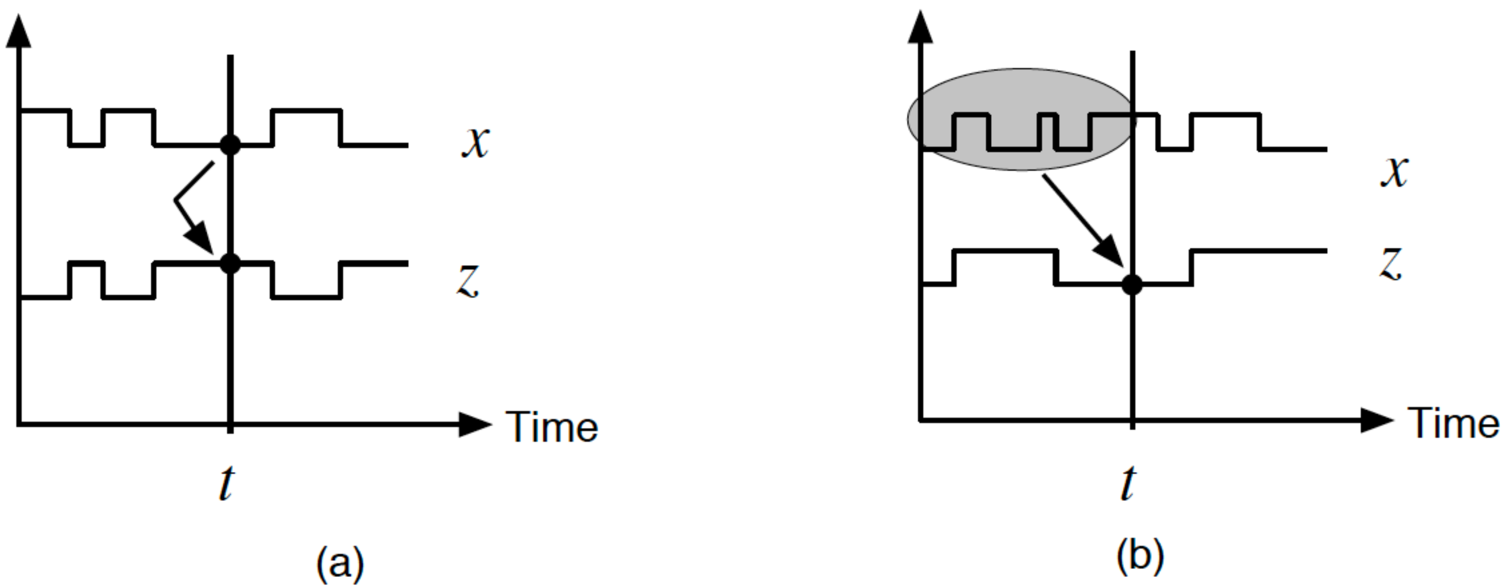


Figure 1.5: Input-output functions for: a) Combinational system; b) Sequential system.

EXAMPLE SEQUENTIAL SYSTEM

- INPUT x with VALUES 0,1, or 2
- OUTPUT z with VALUES 0 or 1
- FUNCTION:

$$z(t) = \begin{cases} 1 & \textbf{if} \;\; (x(0), x(1), \ldots, x(t)) \text{ has} \\ & \quad \text{even } 2'\text{s and odd } 1'\text{s} \\ 0 & \textbf{otherwise} \end{cases}$$

- AN INPUT-OUTPUT PAIR:

t	0	1	2	3	4	5	6	7	8	9	10	11
x	1	2	2	0	1	2	0	0	0	2	1	1
z												

EXAMPLE SEQUENTIAL SYSTEM

- INPUT x with VALUES 0,1, or 2
- OUTPUT z with VALUES 0 or 1
- FUNCTION:

$$z(t) = \begin{cases} 1 & \textbf{if} \quad (x(0), x(1), \ldots, x(t)) \text{ has} \\ & \quad \text{even } 2'\text{s and odd } 1'\text{s} \\ 0 & \textbf{otherwise} \end{cases}$$

- AN INPUT-OUTPUT PAIR:

t	0	1	2	3	4	5	6	7	8	9	10	11
x	1	2	2	0	1	2	0	0	0	2	1	1
z	1	0	1	1	0	0	0	0	0	0	1	0

EXAMPLE COMBINATIONAL SYSTEM

- INPUT $x(t)$ with values from the set of letters (upper and lower case)
- INPUT $y(t)$ with values 0 and 1
- FUNCTION:
  - change $x(t)$ to opposite case when $y(t) = 1$
  - leave it unchanged when $y(t) = 0$
- AN INPUT-OUTPUT PAIR:

t	0	1	2	3	4	5	6
x	E	X	A	M	P	L	E
y	0	1	0	0	0	1	0
z							

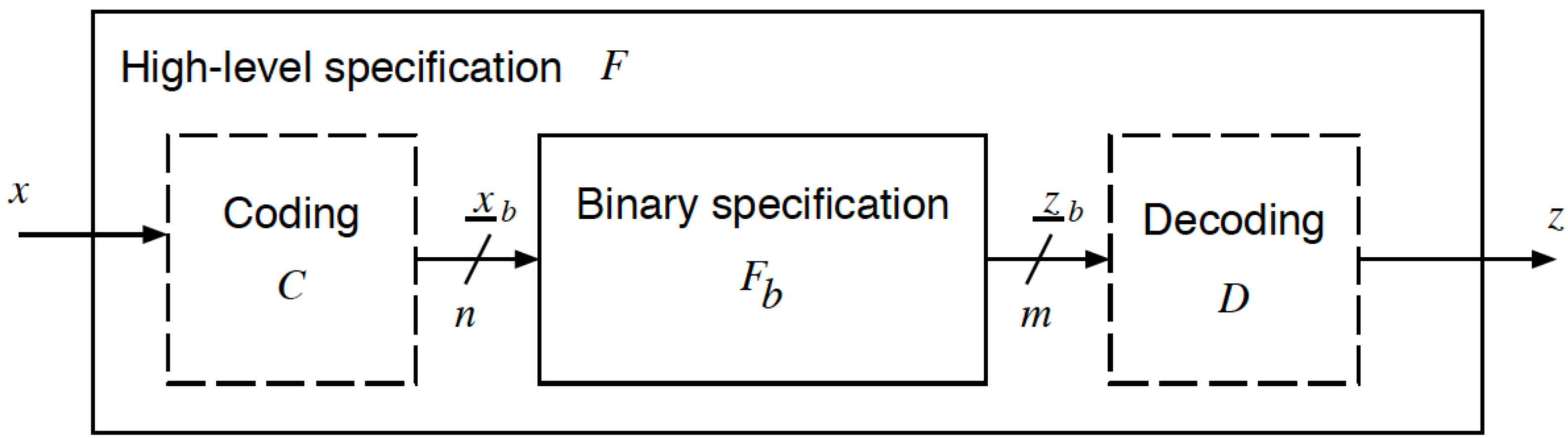
EXAMPLE COMBINATIONAL SYSTEM

- INPUT $x(t)$ with values from the set of letters (upper and lower case)
- INPUT $y(t)$ with values 0 and 1
- FUNCTION:
  - change $x(t)$ to opposite case when $y(t) = 1$
  - leave it unchanged when $y(t) = 0$
- AN INPUT-OUTPUT PAIR:

t	0	1	2	3	4	5	6
x	E	X	A	M	P	L	E
y	0	1	0	0	0	1	0
z	E	x	A	M	P	l	E

BINARY LEVEL

$$\underline{z}_b = F_b(\underline{x}_b)$$



Example

Input: $x \in \{0,1,2,3,4,5,6,7,8,9\}$
Output: $z \in \{0,1,2\}$

Function: F is described by the following table

x	0	1	2	3	4	5	6	7	8	9
$z = F(x)$	0	1	2	0	1	2	0	1	2	0

or by the arithmetic expression

$$z = x \bmod 3,$$

x	0	1	2	3	4	5	6	7	8	9
$\underline{x}_b = C(x)$	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

$\underline{z}_b$	00	01	10
$z = D(\underline{z}_b)$	0	1	2

Input: $\underline{x}_b = (x_3, x_2, x_1, x_0),\ x_i \in \{0,1\}$
Output: $\underline{z}_b = (z_1, z_0),\ z_i \in \{0,1\}$

Function: F_b is described by the following table

$\underline{x}_b$	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001
$\underline{z}_b = F_b(\underline{x}_b)$	00	01	10	00	01	10	00	01	10	00

Data Representation and Manipulation

Data Representation and Manipulation

- How characters and numbers are represented in a typical computer

Characters

- ASCII (American Standard Code for Information Interchange)
- Uses 7 bits to represent 128 different characters

ALPHANUMERIC CODES

Character	Codes	
	ASCII	EBCDIC
A	100 0001	1100 0001
B	100 0010	1100 0010
C	100 0011	1100 0011
⋮	⋮	⋮
Y	101 1001	1110 1000
Z	101 1010	1110 1001
0	011 0000	1111 0000
1	011 0001	1111 0001
2	011 0010	1111 0010
⋮	⋮	⋮
8	011 1000	1111 1000
9	011 1001	1111 1001
blank	010 0000	0100 0000
.	010 1110	0100 1011
(	010 1000	0100 1101
+	010 1011	0100 1110
⋮	⋮	⋮

A	blank	C	A	B	.
01000001	00100000	01000011	01000001	01000010	00101110

REPRESENTATION BY DIGIT-VECTOR

$$\underline{x} = (x_{n-1}, x_{n-2}, \ldots, x_1, x_0)$$

$$x = \sum_{i=0}^{n-1} x_i r^i$$

DIGIT x_i in $\{0, 1, \ldots, r-1\}$, r – THE RADIX

$$\underline{x} = (1, 0, 0, 1, 0, 1)$$

$\Longleftrightarrow$

$$1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = (37)_{10}$$

SET OF REPRESENTABLE VALUES

$$0 \leq x \leq r^n - 1$$

EXAMPLE OF BINARY CODES

Digit Value (Symbol)	Binary $k=1$	Quaternary $k=2$	Octal $k=3$	Hexadecimal $k=4$
	d_0	$d_1 d_0$	$d_2 d_1 d_0$	$d_3 d_2 d_1 d_0$
0	0	00	000	0000
1	1	01	001	0001
2		10	010	0010
3		11	011	0011
4			100	0100
5			101	0101
6			110	0110
7			111	0111
8				1000
9				1001
10 (A)				1010
11 (B)				1011
12 (C)				1100
13 (D)				1101
14 (E)				1110
15 (F)				1111

Unsigned Binary Numbers

- With 4 bits, can represent 0 through 15

$$
\begin{aligned}
1101_2 &= (1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0) \\
&= 13_{10}
\end{aligned}
$$

- With 32 bits, can represent 0 through $2^{32} - 1 = 4,294,967,295$

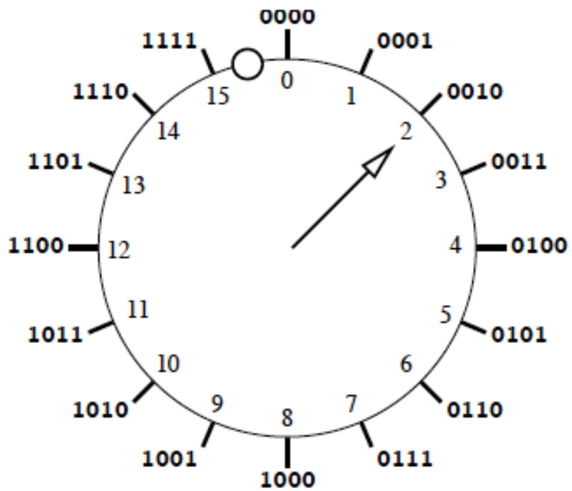
Signed Binary Numbers

- First idea: use MSB as "sign bit" (0 means positive, 1 means negative)
- Called "signed-magnitude" representation
- 4-bit example: 1110 is -6
- With 4 bits, can represent -7 (1111) to $+7$ (0111)
- Problems: two different versions of zero (0000 and 1000), addition is complicated
- Better idea: two's complement representation

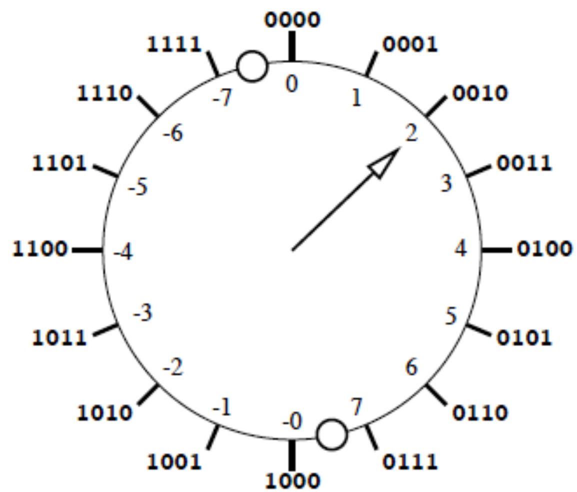
Two's Complement Representation

- Idea: Let MSB represent the negative of a power of 2
- With 4 bits, bit 3 (MSB) represents -2^3
- $1110 = -2^3 + 2^2 + 2^1 = -2$
- With 4 bits, can represent -8 (1000) to $+7$ (0111)
- With 32 bits, can represent $-2,147,483,648$ to 2,147,483,647

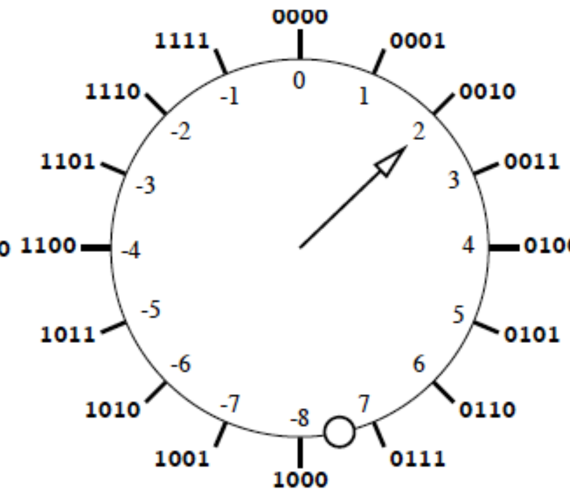
Pictorial Representation, 4 Bits



Unsigned



Signed Magnitude



Two’s Complement

Negating a Two's Complement Number

- For a bit pattern x, let $\overline{x}$ be the result of inverting each bit
- Example: $x = 0110$, $\overline{x} = 1001$
- Since $x + \overline{x} = -1$, $-x = \overline{x} + 1$
- To negate a number in two's complement representation, invert every bit and add 1 to the result

Sign Extension

- With 4 bits, 0110 is +6. With 8 bits, what is +6?
- With 4 bits, 1010 is -6. With 8 bits, what is -6?
- To expand number of bits used, copy old MSB into new bit positions.

Clicker Question

Which one is correct?

- A) 1001 (in signed binary) = 9 (in decimal)
- B) 0101 (in signed binary) = 5 (in decimal)
- C) 1010 (in unsigned binary) = 10 (in decimal)
- D) B and C
- E) none of the above

Clicker Question

2's Complement

What is the 8-bit 2's complement of (the negation of) 0010 0100

a 1101 1011

b 1101 1100

c 1110 1100

d 0020 0200

e None of the above