



UNIVERSITI
TEKNOLOGI
MALAYSIA
www.utm.my

SCSR1013 DIGITAL LOGIC

MODULE 1: INTRODUCTORY CONCEPTS

FACULTY OF COMPUTING



CONTENTS

MODULE 1: DIGITAL LOGIC OVERVIEW

Digital and Analog Quantities

Binary Digits, Logic Levels and Digital Waveform

Introduction to Logic Operations

Overview of Logic Functions

Fixed-Function IC

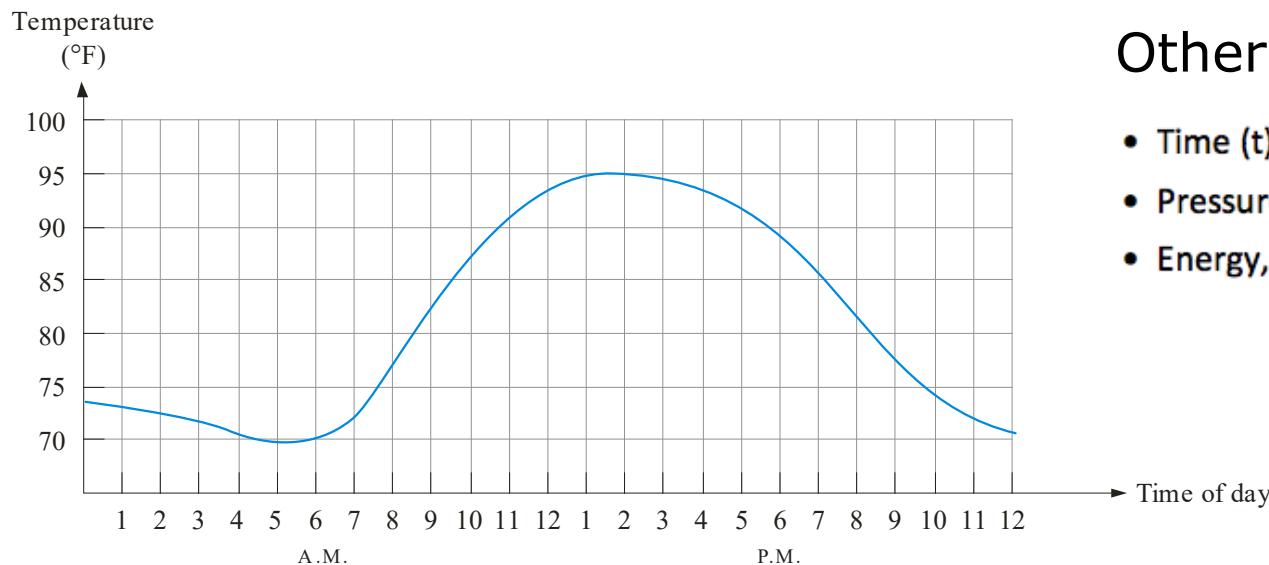
Programmable Logic Devices (PLD)



Digital and Analog Quantities

Analog quantities

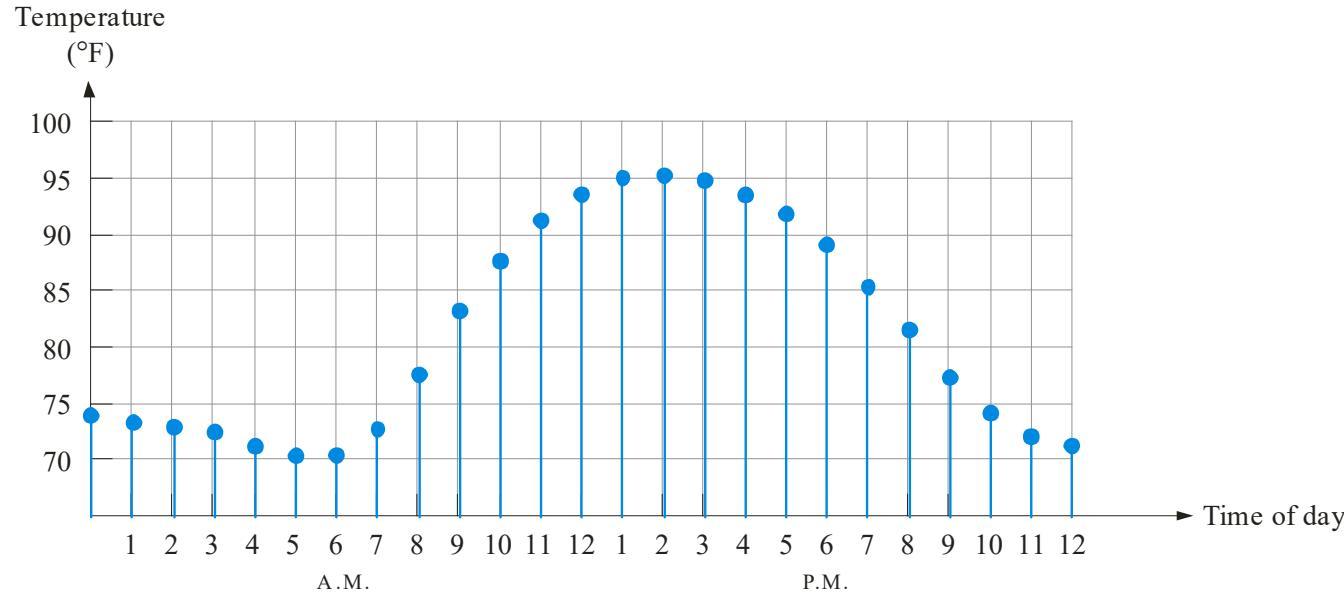
- Most natural quantities that we see are **analog** and vary continuously.
- Analog systems can generally handle higher power than digital systems.



Other examples:

- Time (t) = 10.16s (second)
- Pressure (P) = 220.10KPa (Kilo Pascal)
- Energy, Power = 100.5KW (Kilo Watts)

Digital quantities



- Digital systems can **process, store, and transmit** data more efficiently but can only assign discrete values (**discontinuous**) to each point.

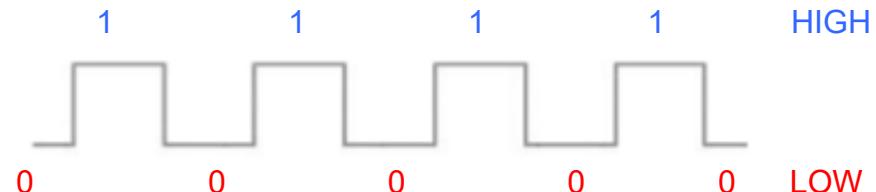
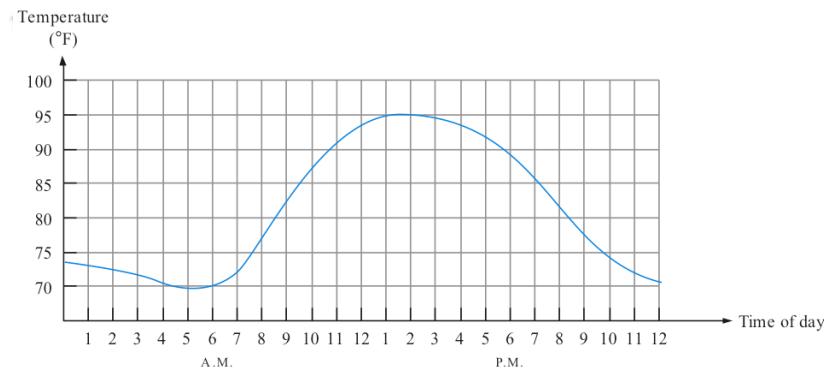
Analog vs Digital

Analog

- Use base 10 (decimal)
- Represented by 10 different level: 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9.
- Analog system: A combination of devices that manipulate values represented in analog form

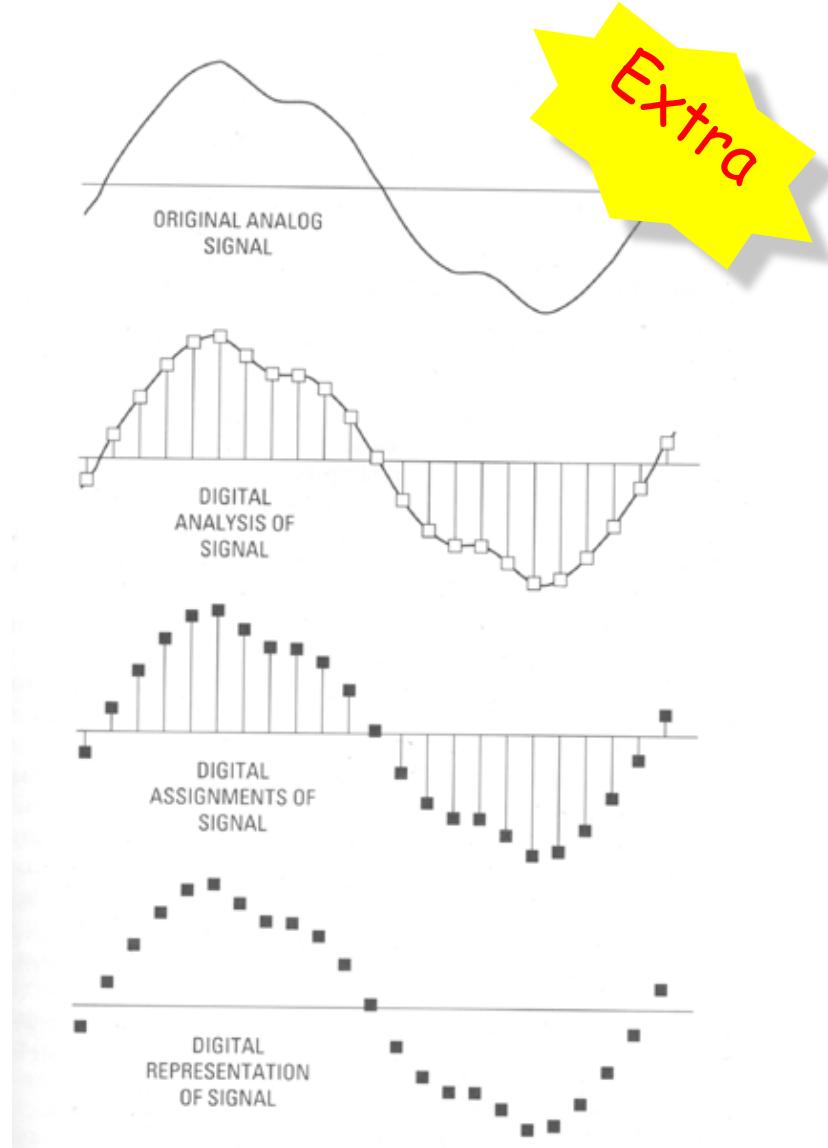
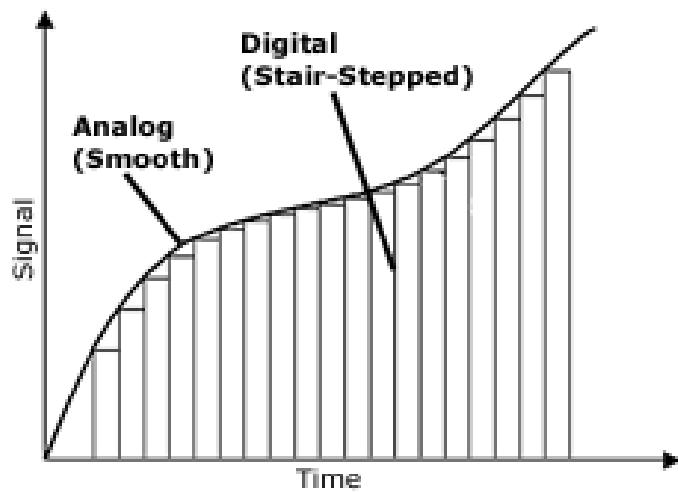
Digital

- Use base 2 (binary)
- Represented by 2 different level: 0 and 1 or low and high.
- Digital system: A combination of devices that manipulate values represented in digital form.

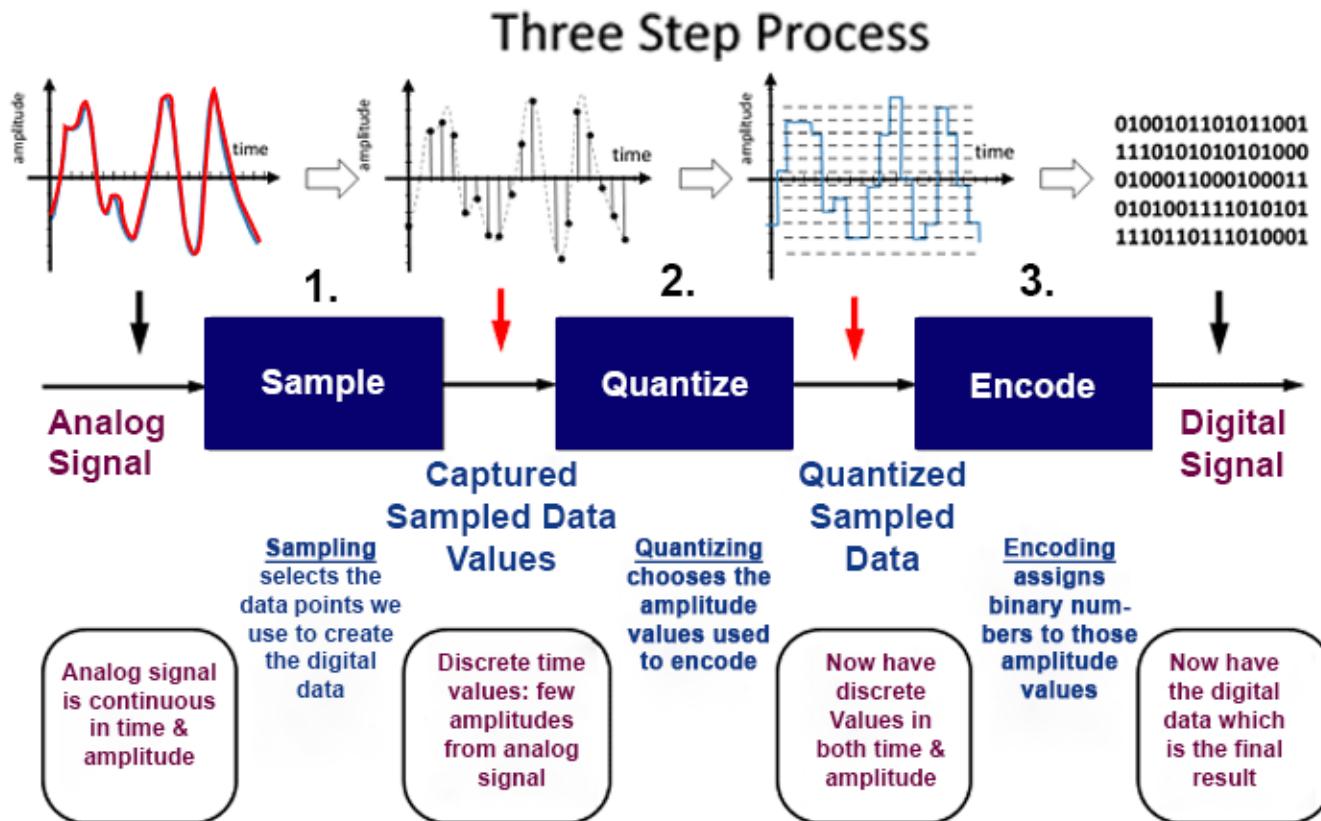


Nyquist's theorem for digitization

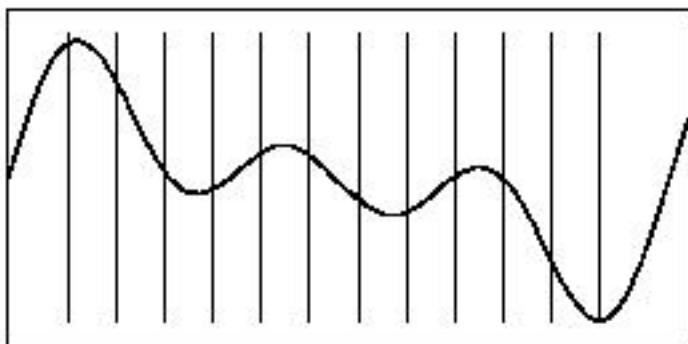
Example of sampling analog-to-digital (frequency at least 2 times higher than analog)



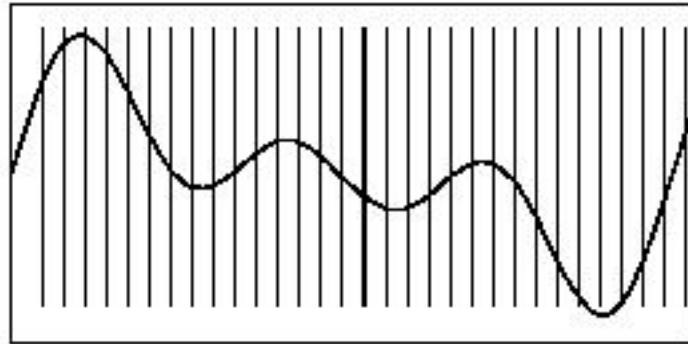
http://www.geardiary.com/2011/04/01/music-diary-notes-the-brave-new-world-of-digital-music/digital_sampling/



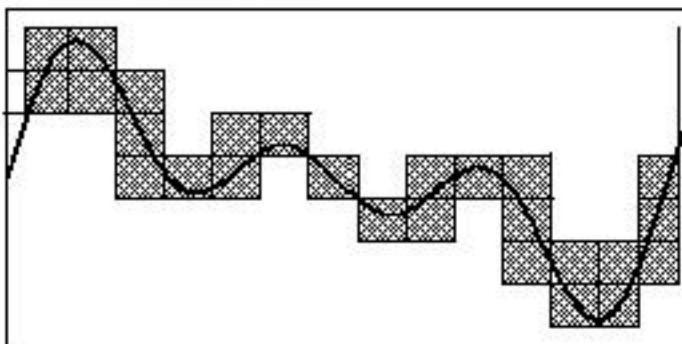
Extra



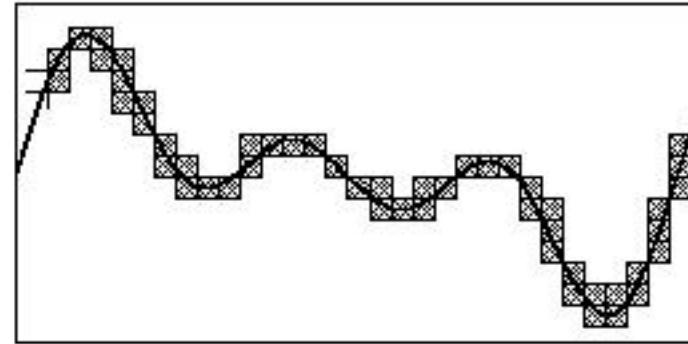
lower sample rates take fewer snapshots
of the waveform



faster sample rates take more
snapshots....



resulting in a rough recreation of the
waveform.



resulting in a smoother and more detailed
recreation of the waveform.



Short Answer

Extra

Self-Test: Which of the following belong to analog system?



(a)



(b)



(c)



(d)



(e)



(f)

- Digital technology is relatively new compared to analog technology, but a lot of analog systems has been changed to a digital systems, Examples:
 - Computers
 - Manufacturing systems
 - Medical Science
 - Transportation
 - Entertainment
 - Telecommunications



*DSL-2320B (ADSL Modem)



Short Answer

Digital



(b)



(e)



(a)

Exercise: Match the picture to which digital application system it belongs to.

- (a) Computers
- (b) Manufacturing systems
- (c) Medical Science
- (d) Transportation
- (e) Entertainment
- (f) Telecommunications



(d)



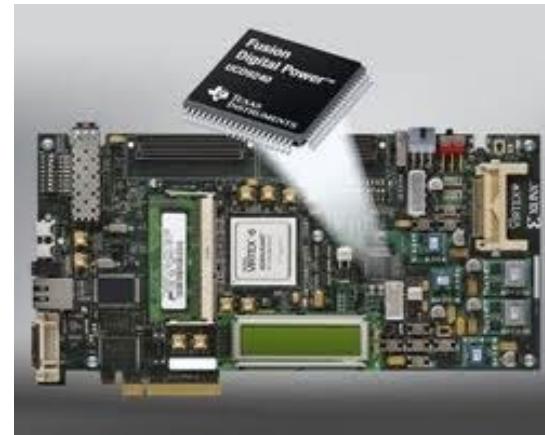
(f)



(c)

The Digital Advantages

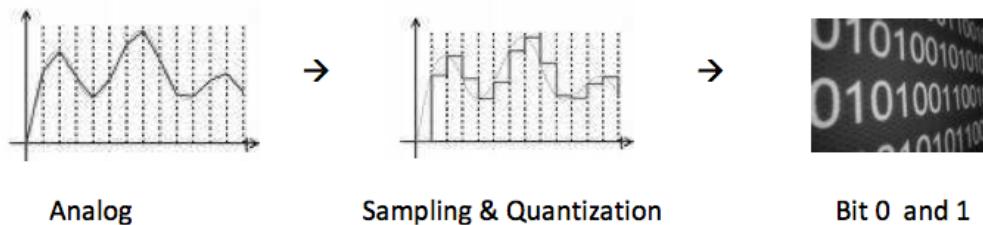
- Ease of design
- Ease of storage
- Accuracy and precision are easier to maintain
- Programmable operation
- Less affected by noise
- Ease of fabrication on IC chips
 - Thus, the digital systems is more efficient and reliable for:
 - Data Processing
 - Data Transmission
 - Data Storage



Digital Disadvantages

- Greater bandwidth
- Sampling error

Sampling Error (Quantization Error): is derived from Analog to Digital Conversion Process:



- Compatibility with existing analog systems
- Short product half life

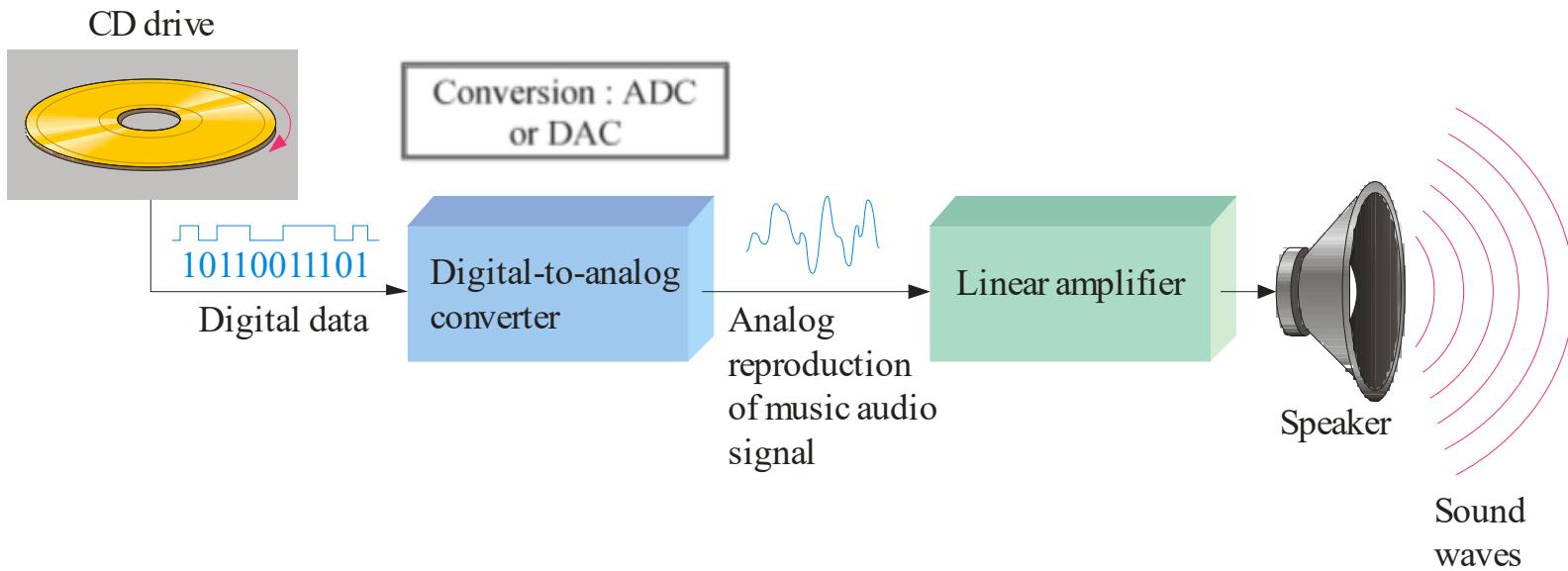


Analog and Digital Systems

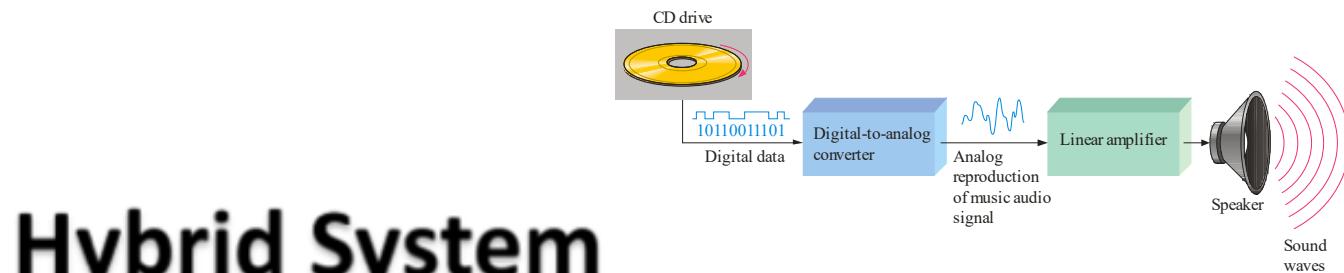
- Many systems use a **mix** of analog and digital electronics to take advantage of each technology.
- A typical CD player accepts digital data from the CD drive and converts it to an analog signal for amplification.



<http://www.it-echo.com/2009/11/14/bose-wave-music-system-and-multi-cd-changer-bundle.html>
<http://cdn-static.zdnet.com/i/story/61/18/006128/31929466-2-440-overview-1.gif>



1. Convert digital sound (CD) to analog
2. Process (amplify) the analog information
3. Convert the analog signal to sound



Hybrid System

- The audio CD is a typical hybrid (Analog & Digital) system.
 - Analog sound is converted into analog voltage using a microphone.
 - Analog voltage is changed into digital through an ADC in the recorder.
 - Digital information is stored on the CD .
 - At playback the digital information is changed into analog by a DAC in the CD player.
 - The analog voltage is amplified and used to drive a speaker that produces the original analog sound.

Conversion:



Analog to Digital Converter (ADC):

- Convert analog signal into digital signal using process such as sampling, quantization process and digital conversion.
- Error will occur during the sampling and quantization, hence loss of information can happen.



Digital to Analog Converter (DAC):

- Needed if the speaker is using analog system.
- Need to convert the digital data to analog signal in order for the speaker works properly and the sound can be heard by human.





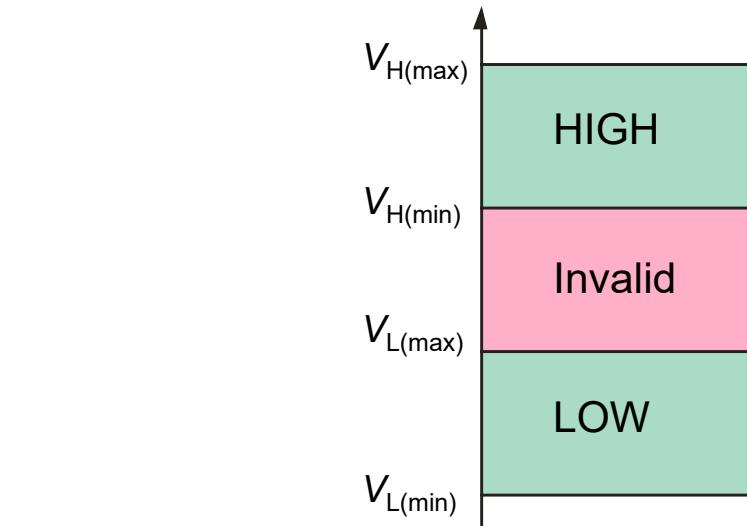
Digits, Logic Levels and Digital Waveform

Binary digits and logic levels

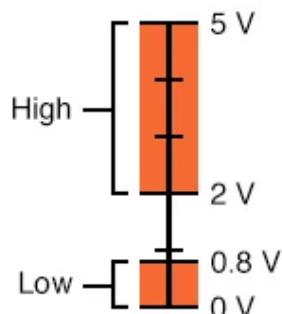
Digital electronics uses circuits that have two states, which are represented by two different voltage levels:

- HIGH (bit 1)
- LOW (bit 0)

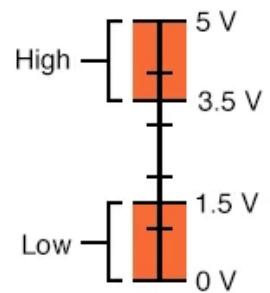
A bit can have the value of either a 0 or a 1, depending on if the voltage is **HIGH** or **LOW**.



Acceptable TTL Gate Input Signal Levels

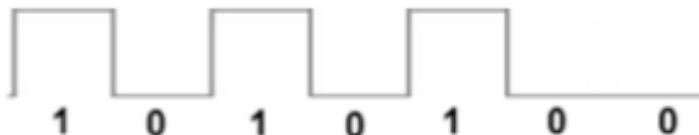


Acceptable CMOS Gate Input Signal Levels



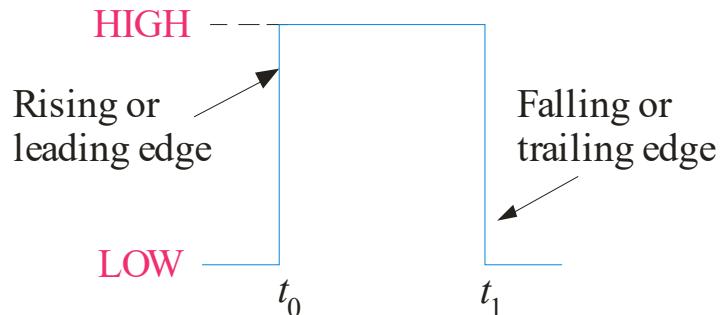
Digital Waveforms

- Digital systems usually uses a square wave to represent binary value of High and Low voltage
- Digital waveforms are made up of a series of pulses (changes of High & Low)

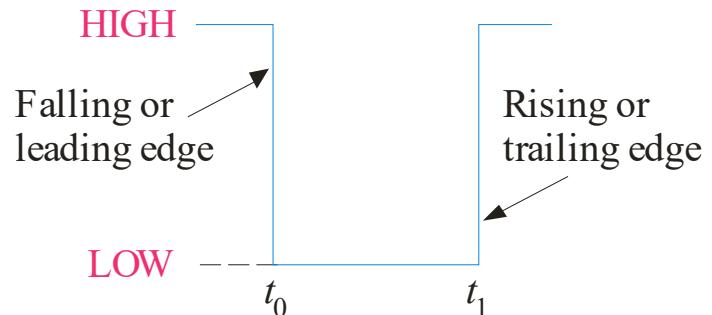


1010100 represented as electrical signal

Anatomy of a Pulse



(a) Positive-going pulse



(b) Negative-going pulse

Positive Logic (active high)

High = 1 (Bit 1)
Low = 0 (Bit 0)

Negative logic (active low)

High = 0
Low = 1

A positive going pulse is one that goes from a normally **LOW** logic level to a **HIGH** level and then back again.

A negative going pulse is one that goes from a normally **HIGH** logic level to a **LOW** level and then back again.

Active-High – The "Intuitive" Convention

- the system most people naturally assume. It **directly maps** "high" to "true."
- **HIGH** Voltage Level = **Logic 1**
 - (interpreted as TRUE, ON, ACTIVE)
- **LOW** Voltage Level = **Logic 0**
 - (interpreted as FALSE, OFF, INACTIVE)
- The Analogy: The Standard Light Switch
 - Imagine a simple light switch on the wall.



Active-Low – The "Safety-First" Convention

- This is where we **flip** the **interpretation**, not the physics.
- The active or "important" event is now triggered by a LOW voltage.
- **LOW Voltage Level = Logic 1**
 - (interpreted as TRUE, ON, ACTIVE)
- **HIGH Voltage Level = Logic 0**
 - (interpreted as FALSE, OFF, INACTIVE)

Active-Low – The "Safety-First" Convention

- The Analogy: The Emergency Stop Button
- Think of the big red emergency stop button on a piece of heavy machinery.



Active-Low – The "Safety-First" Convention

- The Emergency Stop Button at fuel pump



<https://www.mymesra.com.my/about-us/news-press-releases/clarification-on-the-proper-use-of-emergency-button-at-petronas-stations>

Active-Low – The "Safety-First" Convention

- Under normal, **safe operating** conditions, the circuit is complete, and the signal is at a **HIGH** voltage.
- The system **interprets** this as "inactive" or "everything is okay."
- When someone **presses** the **button**, it breaks the circuit and pulls the signal **voltage to LOW** (0V).
- This **LOW** signal is the **trigger**, (the ACTIVE signal) that tells the machine to **immediately** shut down.
- In this **critical safety system**, the action, the important event, happens when the **signal goes LOW**.



Why is Active-Low safer?

www.utm.my

- **Noise Immunity:**

- In an electrically noisy environment (like a factory floor), stray **voltage spikes** (noise) could **accidentally** create a **false HIGH signal**.
- If HIGH meant "activate," the machine could start or stop randomly.
- A **LOW signal** (being connected to Ground) is much **more stable** and **less prone** to noise.
- It's much **harder** to accidentally create a connection to Ground.



Why is Active-Low safer?

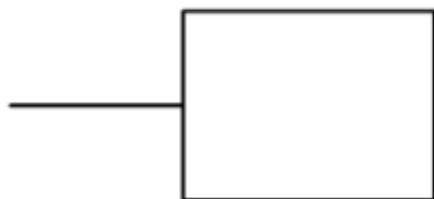
www.utm.my

- **Fail-Safe Design:**

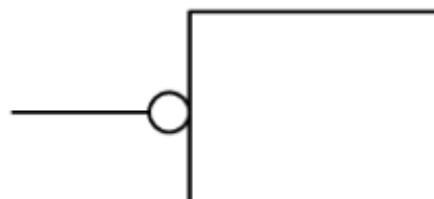
- What happens if a **wire gets cut** in our emergency stop button circuit?
- The **signal will float or go LOW**.
- In an **active-low** system, a cut wire **causes** the machine to **stop**, the **safest** possible outcome.
- If it were an **active-high** system, a cut wire would be **interpreted** as the "**OFF**" signal, and you would **lose** the ability to **stop** the machine, which is **dangerous**.

Generic Circuit Symbols

Symbols to show the input state of “active high” and “active low”:



“active high”



“active low”



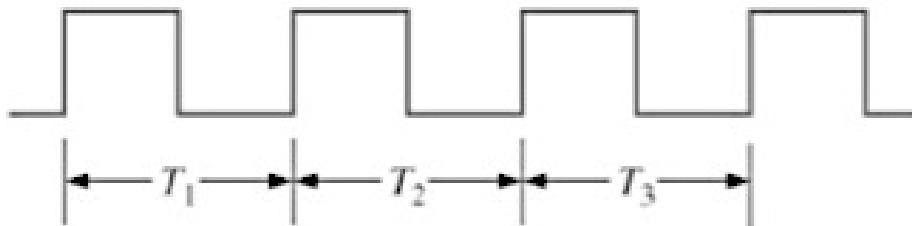
- There are 2 types of waveforms:
 - Periodic
 - Non-periodic

Digital Waveforms

- Two type of squarewave

- Periodic

- The signal keep on repeating after a period of time



$$\text{Period} = T_1 = T_2 = T_3 = \dots = T_n$$

$$\text{Frequency} = \frac{1}{T}$$

- Non-Periodic / Aperiodic

- Doesn't have a period



Periodic Signal Parameter

- Frequency (f) is the rate at which the signal repeat itself at a fixed interval. Is measured in cycles per second or Hertz (Hz)

$$f = \frac{1}{T} \text{ Hz}$$

- Period (T) is the time from the edge of one pulse to the corresponding edge of the next pulse. Is measured in second

$$T = \frac{1}{f} \text{ seconds}$$

■ Example:

■ clock frequency : $f = 100\text{Hz}$,

so, period : $T = 1/100\text{Hz} = \underline{0.01\text{s}} = 0.01\text{ s} \times 1000 = 10\text{ ms}$

$$\begin{aligned}s &\rightarrow \text{ms } (\times 10^3) \\ \text{ms} &\rightarrow s \quad (\times 10^{-3})\end{aligned}$$

Some examples of periodic signal display on the oscilloscope:



(a) Square waveform

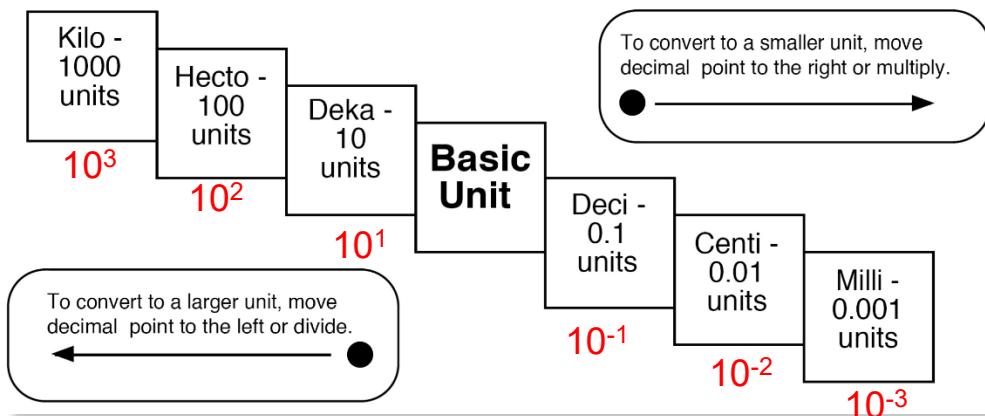


(b) Sinusoid waveform

Unit Conversion

- ◆ Kilo (K)= 10^3
- ◆ Mega (M)= 10^6
- ◆ Giga (G)= 10^9
- ◆ Tera (T)= 10^{12}
- ◆ Mili (m)= 10^{-3}
- ◆ Micro (μ) = 10^{-6}
- ◆ Nano (n)= 10^{-9}
- ◆ Piko (p) = 10^{-12}

Metric Conversion Chart



- Mili (m) = 10^{-3}
- Micro (μ) = 10^{-6}
- Nano (n) = 10^{-9}
- Piko (p) = 10^{-12}

Example : $f = 100\text{KHz}$, So

$$T = 1/f$$

$$= 1/(100 * 10^3 \text{Hz})$$

$$= 0.01 * 10^{-3} \text{s}$$

$$= 0.01 \text{ms}$$

$$= 10 \mu\text{s}$$

$$\begin{aligned}&= (0.01 * 10^{-3}) \text{s} \times 10^3 \\&= (0.01 * 10^{-3+3}) \text{ms} \\&= (0.01 * 10^0) \text{ms} \\&= 0.01 \text{ms}\end{aligned}$$

$$\begin{aligned}&= (0.01 * 10^{-3}) \text{s} \times 10^6 \\&= (0.01 * 10^{-3+6}) \mu\text{s} \\&= (0.01 * 10^3) \mu\text{s} \\&= 10 \mu\text{s}\end{aligned}$$

Exercise 1.1 : Calculate the frequency of signals if time period are given as the following:

a) $10ms = \underline{\hspace{2cm}} Hz$

b) $100\mu s = \underline{\hspace{2cm}} KHz$

c) $100ns = \underline{\hspace{2cm}} MHz$

d) $1000ps = \underline{\hspace{2cm}} GHz$

Exercise 1.2 : Calculate the time (period) of signals if the frequencies are given as the following:

a) $1000\text{KHz} = \underline{\hspace{2cm}} \mu\text{s}$

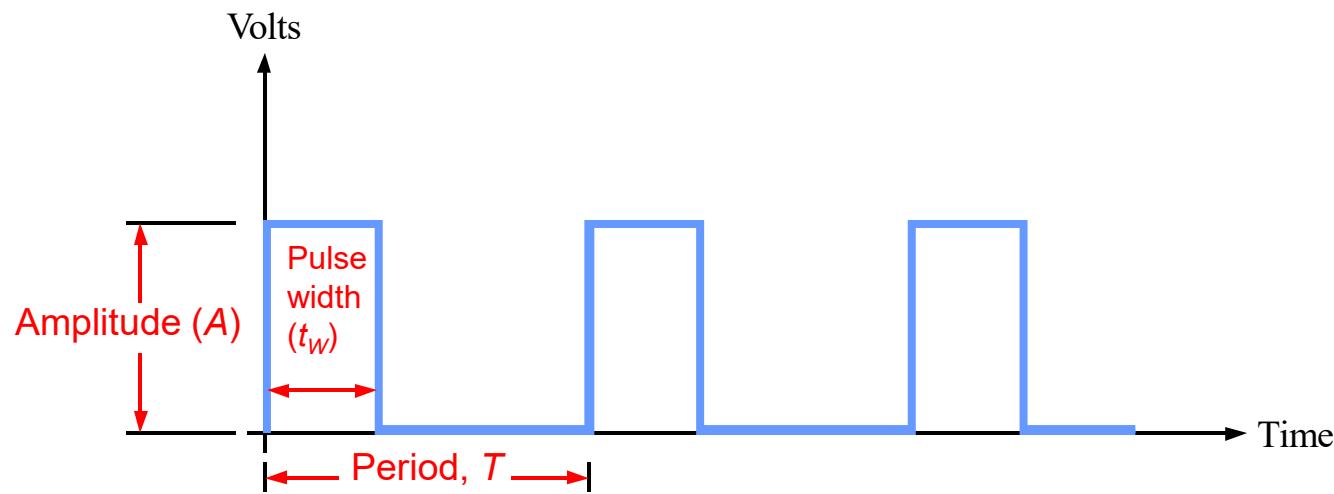
b) $100\text{MHz} = \underline{\hspace{2cm}} \text{ns}$

c) $1000\text{GHz} = \underline{\hspace{2cm}} \text{ps}$

d) $100\text{THz} = \underline{\hspace{2cm}} \text{ps}$

Repetitive Pulse Waveform

- In addition to frequency and period, repetitive pulse waveforms are described by the **amplitude (A)**, **pulse width (t_W)** and **duty cycle**.
- Duty cycle is the ratio of t_W to T .



Duty Cycle

- Duty cycle is the fraction of time that a system is in an "active" state (operated), defined as

$$\text{Duty cycle} = (t_w/T)100\%$$

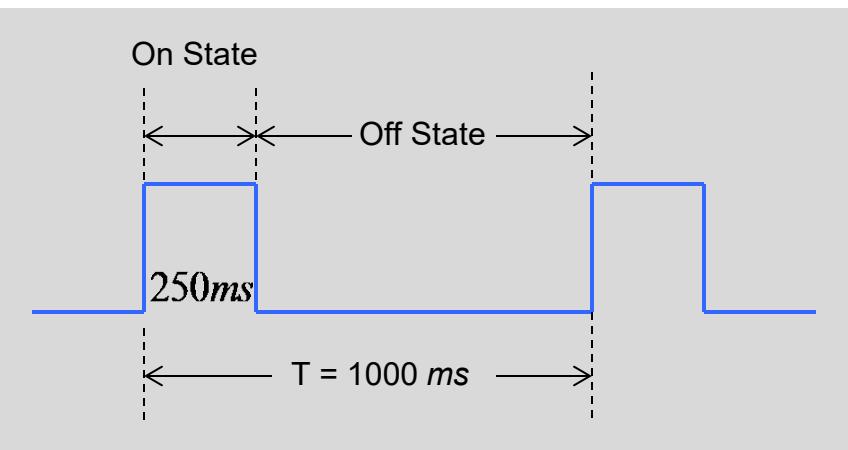


Duty cycle is the ratio of t_w to T .

Example : a periodic digital waveform has a pulse width (t_w) 1ms and period time (T) 10ms, calculate duty cycle?

$$\text{Duty cycle} = 1\text{ms}/10\text{ms} * 100\% = 10\%$$

Exercise 1.3: Given the duration or period of a system is 1000ms, determine the *on state* and *off state* of the system that operate with the ratio of duty cycle is 25%. Show your works.



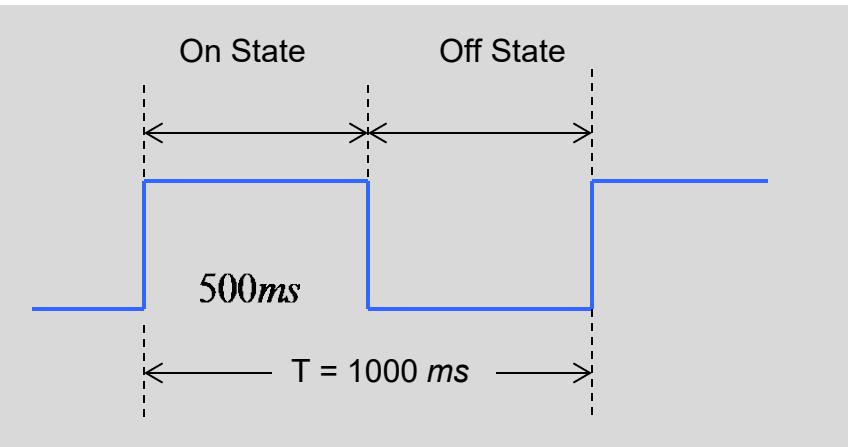
Solution 1.3:

Duty cycle → on state

$$= 25\% \times 1000 \text{ ms} = \frac{25}{100} \times 1000 \text{ ms} = \frac{1}{4} \times 1000 \text{ ms} = 250 \text{ ms}$$

$$\text{Off state : } = 1000 \text{ ms} - 250 \text{ ms} = 750 \text{ ms}$$

Exercise 1.4: Given the duration or period of a system is 1000ms, determine the *on state* and *off state* of the system that operate with the ratio of duty cycle is 50%. Show your works.



Solution 1.4:

Duty cycle → on state

$$= 50\% \times 1000 \text{ ms} = \frac{50}{100} \times 1000 \text{ ms} = \frac{1}{2} \times 1000 \text{ ms} = 500 \text{ ms}$$

$$\text{Off state : } = 1000 \text{ ms} - 500 \text{ ms} = 500 \text{ ms}$$

Exercise 1.5:

Given the *duty cycles* of a system is 30% for a duration of 650ms.

- a) Calculate the pulse width of the system.
- b) Determine the *off state* of the system that operate with the ratio of duty cycle.

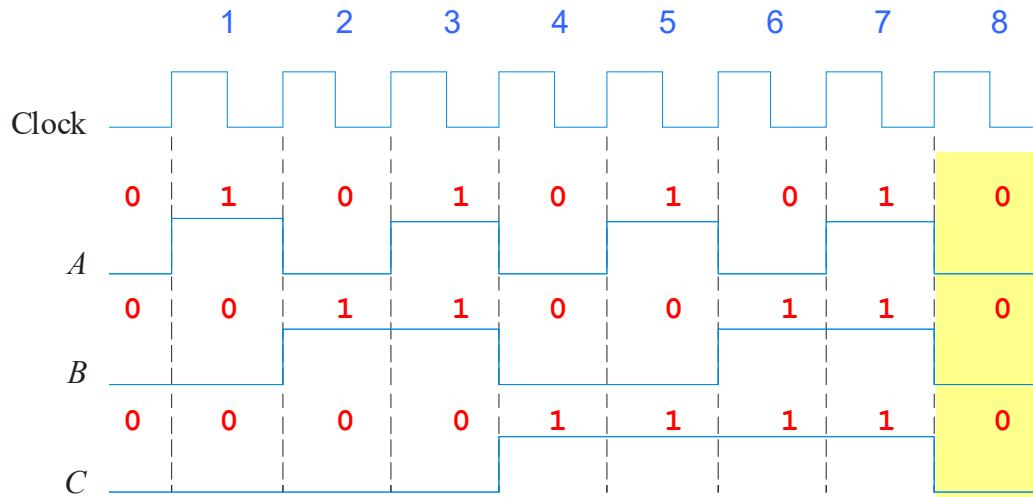
Show your works.



Image Upload

Timing diagram

A timing diagram is used to show the relationship between two or more digital waveforms,

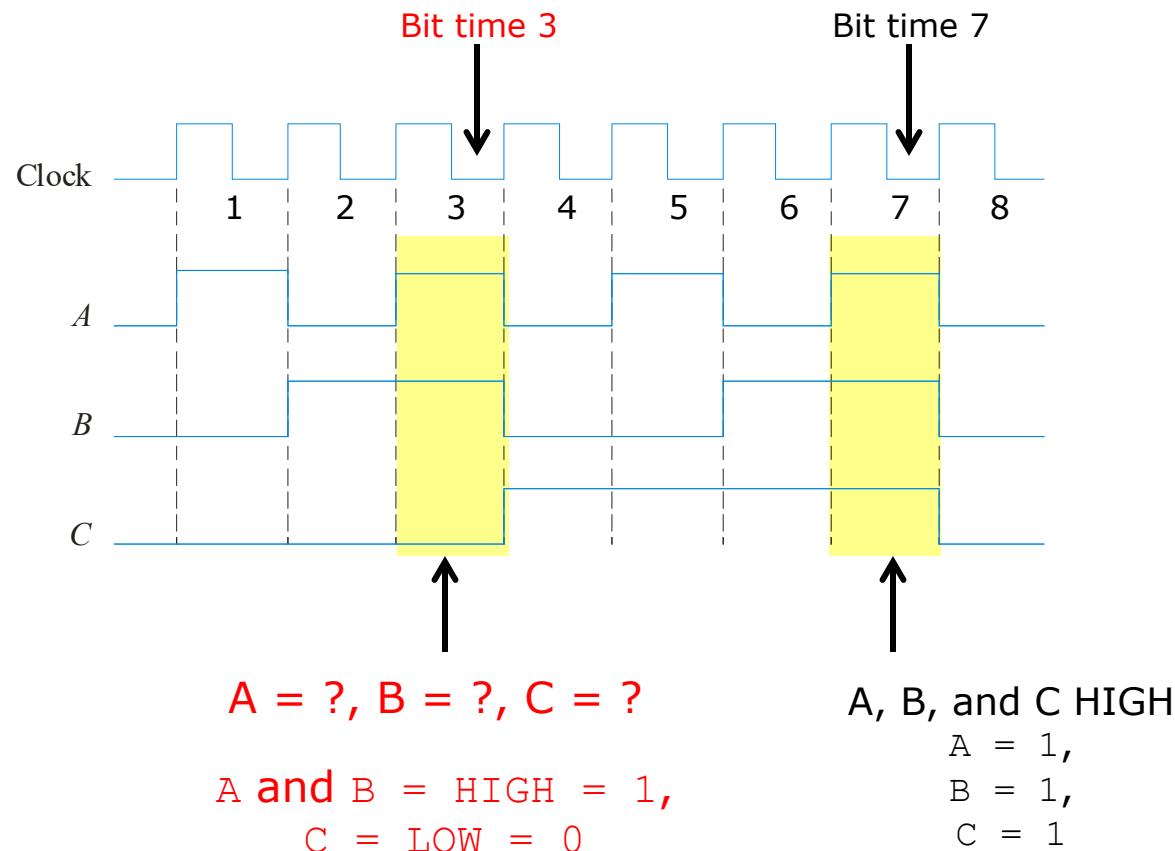


At time 8, all
A, B, and C **LOW**

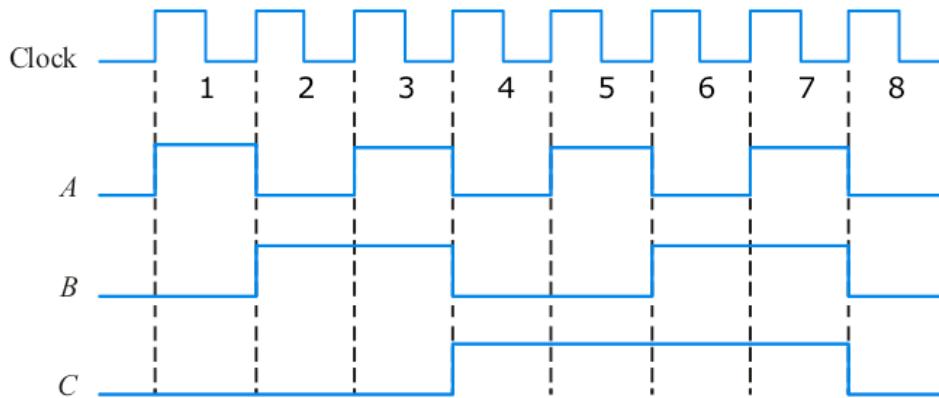


A diagram like this can be observed directly on a logic analyzer.

Example: Timing Diagram



Example: Timing Diagram



Exercise: Complete the truth table.

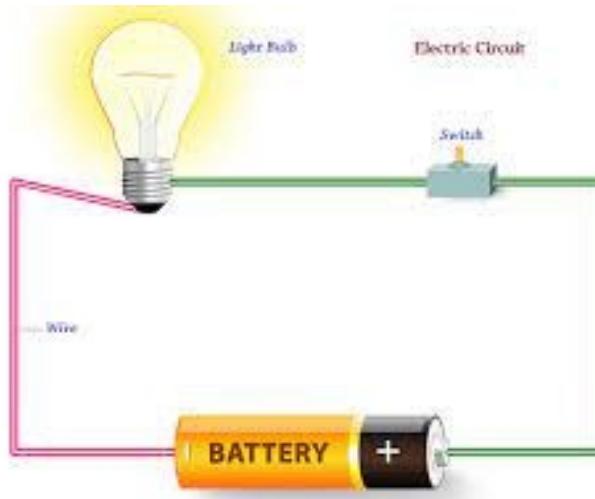
Clock (↑)	Input		Output
	A	B	
1	1	0	0
2	0	1	0
3	1	1	0
4	0	0	1
5	1	0	1
6	0	1	1
7	1	1	1
8	0	0	0



Introduction to Logic Operations

Introduction

- What is logic?
 - is something related to human reasoning,
 - where input is provided to a system or mathematical operation and outputs are provided based on those inputs





Introduction

- 1854
 - George Boole (Irish mathematician) developed mathematical proofs regarding logic.
- 1938
 - Claude Shannon, applied George Boole's 'Boolean Algebra' to the analysis of electrical circuits
 - and that's kind of where the digital electronics started

AND

True only if **all** input conditions are true.



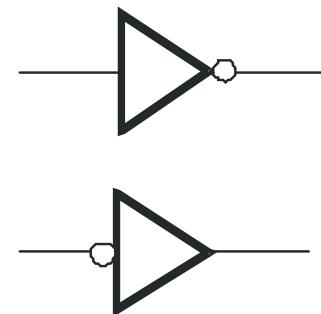
OR

True only if **one or more** input conditions are true.

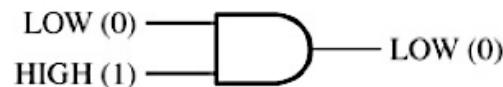


NOT

Indicates the **opposite** condition (inverter).



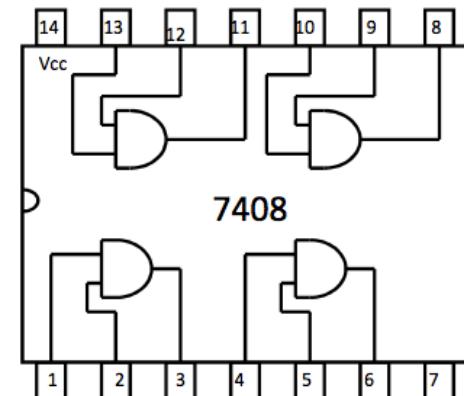
Logic Gates: AND



AND operation

Truth Table AND

X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1



7408 IC four (Quad) AND gates

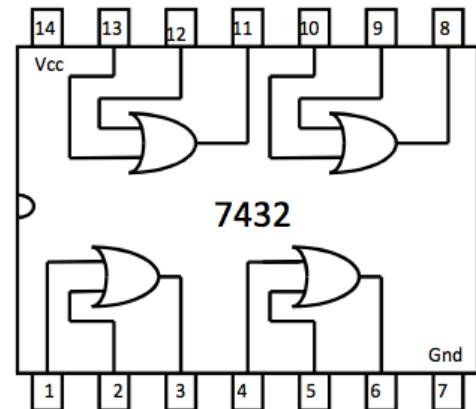
Logic Gates: OR



OR operation

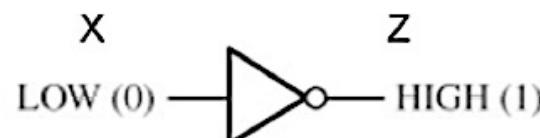
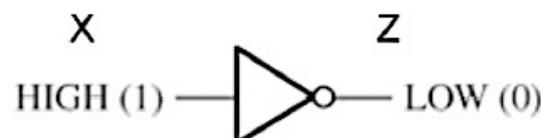
X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

Truth Table OR



7432 integrated circuit provides four (Quad) two-inputs OR gates

Logic Gates: NOT

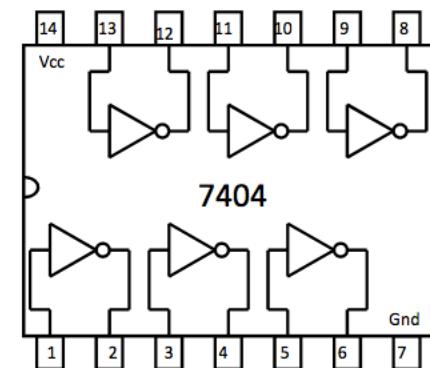


NOT operation

Truth table shows the relationship between output and the input.

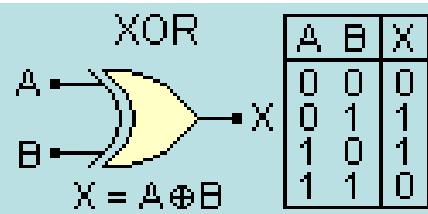
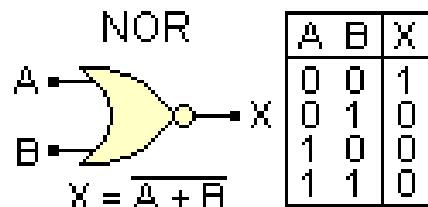
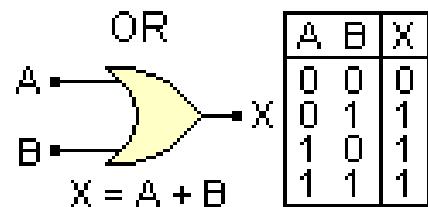
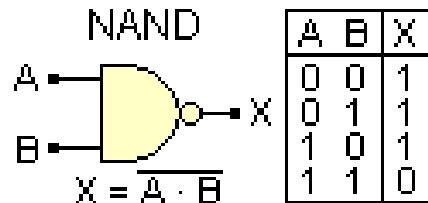
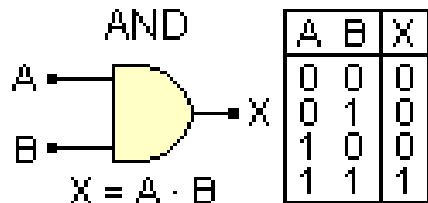
Truth Table for NOT

X	Z
0	1
1	0



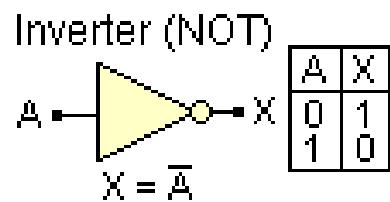
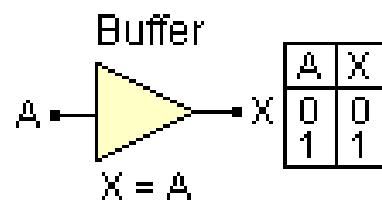
7404 IC six inverters

Logic Gates: Summary



→ **XNOR**

$$X = \overline{A \oplus B}$$

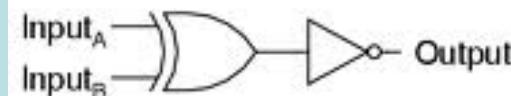


Exclusive-NOR gate



A	B	Output
0	0	1
0	1	0
1	0	0
1	1	1

Equivalent gate circuit



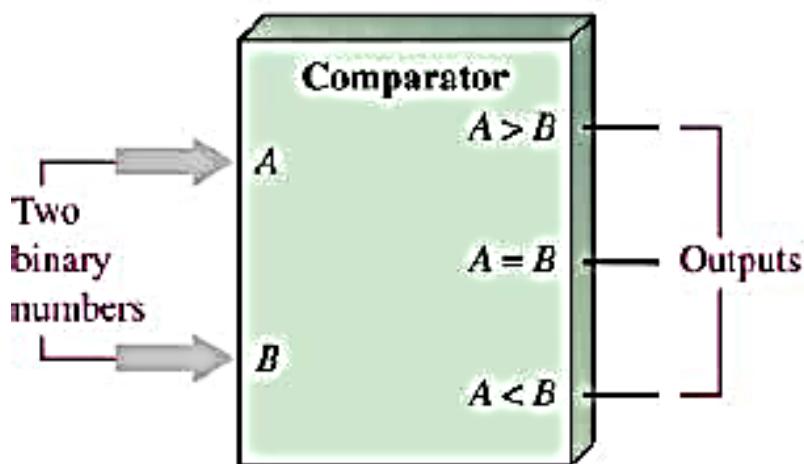


Overview of Logic Functions

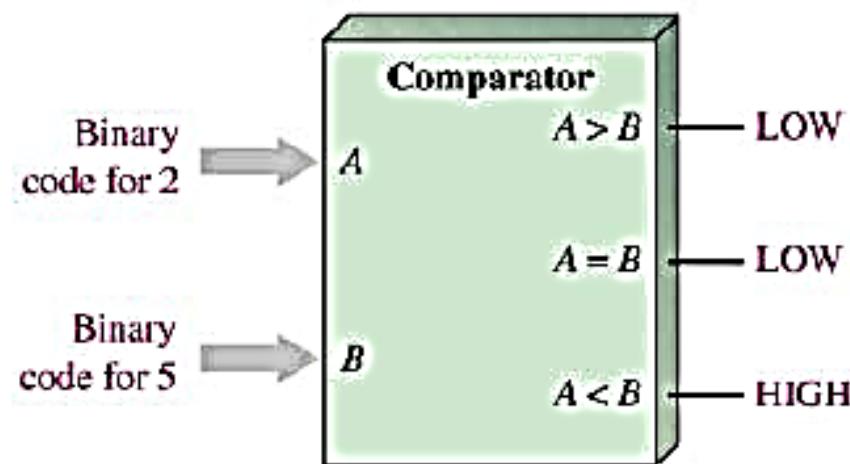
Basic Logic Functions

- Any digital systems has one or more of the following function.
 - This functions are built from the basic gates.
 - Comparison Function
 - Arithmetic Functions
 - Code conversion function
 - Encoding function
 - Decoding function
 - Data selection function
 - Data storage function
 - Counting function

Comparison Function



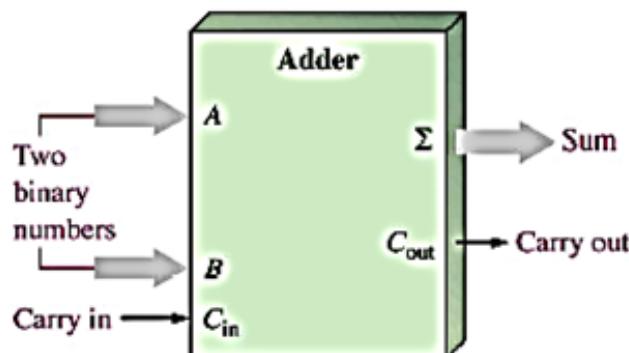
(a) Basic magnitude comparator



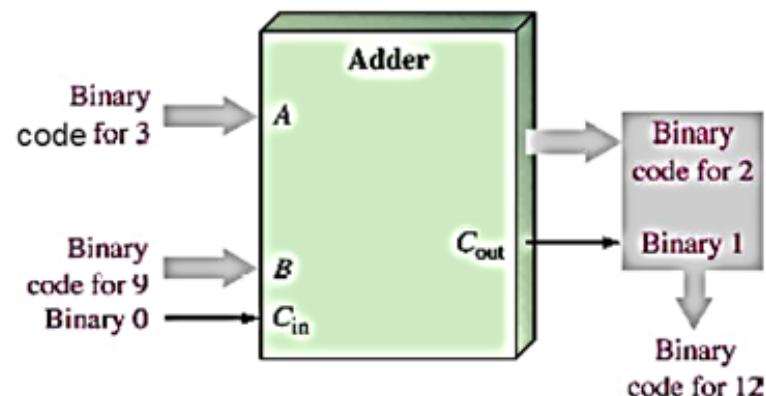
(b) Example: A is less than B ($2 < 5$) as indicated by the HIGH output ($A < B$)

Arithmetic Functions

- Adder



(a) Basic adder



(b) Example: *A* plus *B* ($3 + 9 = 12$)

- Subtractor
- Multiplier
- Division

All the other arithmetic operations can be derived from adder:

- Subtraction is and addition of negative number such as $A-B = A+(-B)$
- Multiplication is a repeated addition such as $A*3=A+A+A$
- Division is a repeated subtraction which is a repeated addition such as
 $6/3=6-3-3=6+(-3)+(-3)$
 - subtract until the remainder = 0
 - total number of subtraction = 2 which is the answer

Code Conversion Function

- A code is a set of bits arranged in a unique pattern and used to represent specified information.
 - Examples : BCD, ASCII
- The usage of codes allow a faster and more efficient data processing.



http://www.ehow.com/how_7162480_

只要去探索
只 - 听不在

[symbols.com](#)



<http://facebooksmileysinfo.com/wp-content/uploads/2012/04/Smiley-Facebook-emoticons.jpg>

و ذ خ ه ت
ج ص م ظ
ك ح و ئ ك
ي ع ل ا ئ
ا ي س ل ب ش

<http://depositphotos.com/2746252/stock-illustration-Arabic-alphabet.html>



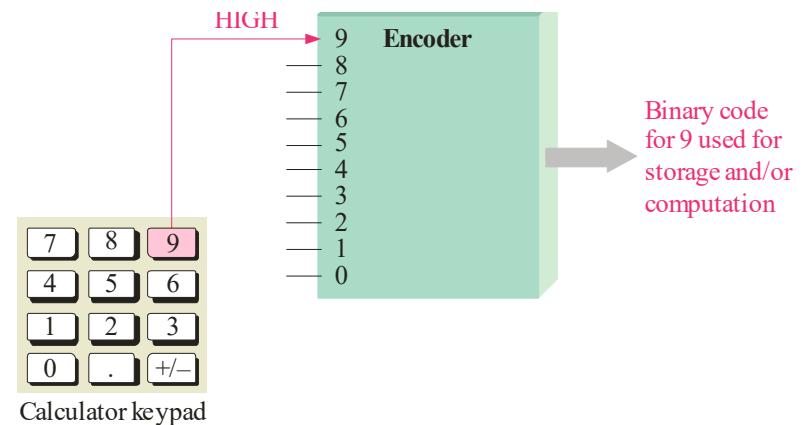
<http://allenmathblog.files.wordpress.com/2012/01/integers.jpg>

Ctrl	Dec	Hex	Char	Code	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
^@	0	00		NUL	32	20	!	64	40	@	96	60	'
^A	1	01		SOH	33	21	"	65	41	À	97	61	a
^B	2	02		STX	34	22	#	66	42	È	98	62	b
^C	3	03		ETX	35	23	\$	67	43	Ç	99	63	c
^D	4	04		EOT	36	24	%	68	44	Ð	100	64	d
^E	5	05		ENQ	37	25	&	69	45	Ѐ	101	65	e
^F	6	06		ACK	38	26	,	70	46	Ѡ	102	66	f
^G	7	07		BEL	39	27	(71	47	Ѿ	103	67	g
^H	8	08		BS	40	28)	72	48	Ѿ	104	68	h
^I	9	09		HT	41	29	*	73	49	ѿ	105	69	i
^J	10	0A		LF	42	2A	+	74	4A	ѿ	106	6A	j
^K	11	0B		VT	43	2B	`	75	4B	ѿ	107	6B	k
^L	12	0C		FF	44	2C	-	76	4C	ѿ	108	6C	l
^M	13	0D		CR	45	2D	.	77	4D	ѿ	109	6D	m
^N	14	0E		SO	46	2E	/	78	4E	ѿ	110	6E	n
^O	15	0F		SI	47	2F	0	79	4F	ѿ	111	6F	o
^P	16	10		DLE	48	30	1	80	50	ѿ	112	70	p
^Q	17	11		DC1	49	31	2	81	51	ѿ	113	71	q
^R	18	12		DC2	50	32	3	82	52	ѿ	114	72	r
^S	19	13		DC3	51	33	4	83	53	ѿ	115	73	s
^T	20	14		DC4	52	34	5	84	54	ѿ	116	74	t
^U	21	15		NAK	53	35	6	85	55	ѿ	117	75	u
^V	22	16		SYN	54	36	7	86	56	ѿ	118	76	v
^W	23	17		ETB	55	37	8	87	57	ѿ	119	77	w
^X	24	18		CAN	56	38	9	88	58	ѿ	120	78	x
^Y	25	19		EM	57	39	:	89	59	ѿ	121	79	y
^Z	26	1A		SUB	58	3A	;	90	5A	ѿ	122	7A	z
^[27	1B		ESC	59	3B	<	91	5B	[123	7B	{
^`	28	1C		FS	60	3C	=	92	5C	\	124	7C	
^]	29	1D	▲	GS	61	3D	>	93	5D]	125	7D	}
^~	30	1E	▼	RS	62	3E	?	94	5E	~	126	7E	~
^-	31	1F		US	63	3F		95	5F	Ѡ	127	7F	Ѡ

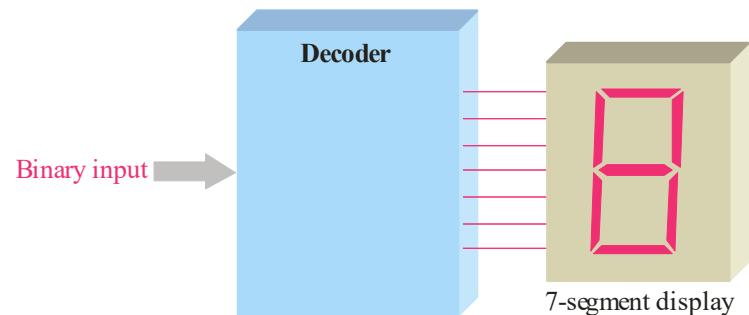
* ASCII code 127 has the code DEL. Under MS-DOS, this code has the same effect as ASCII 8 (BS). The DEL code can be generated by the CTRL + BKSP key.

Encoding & Decoding Function

The encoding function



The decoding function



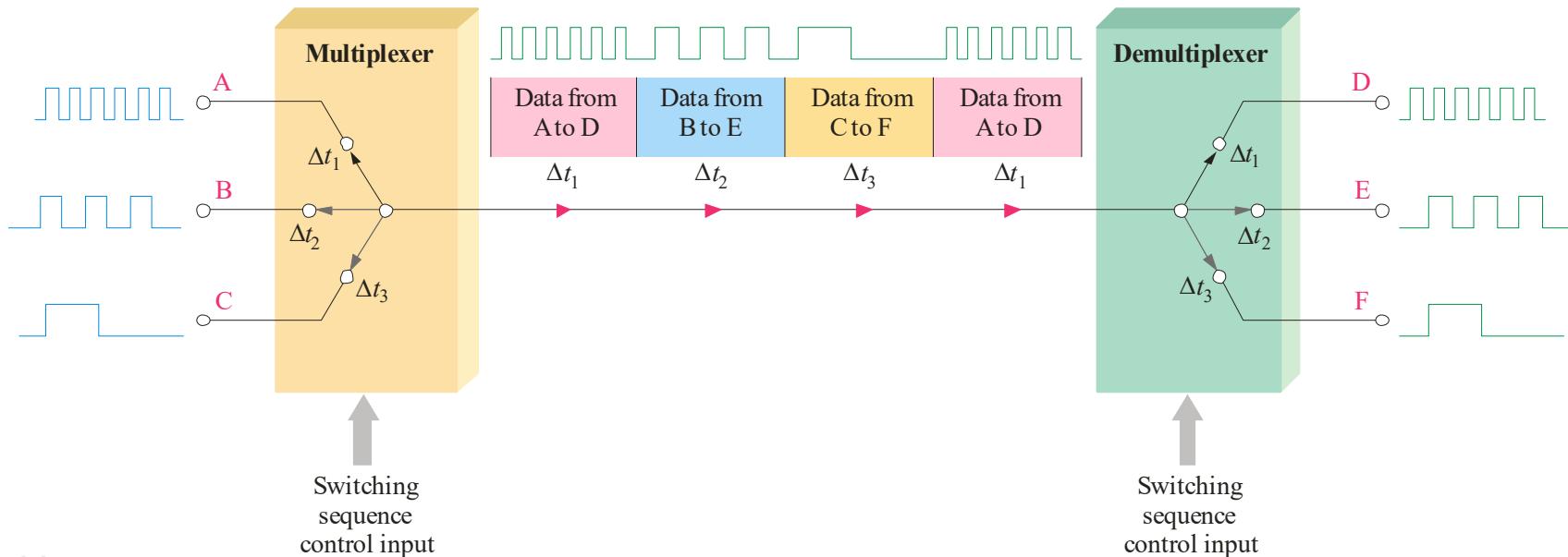
Solution:

MUX: select and permit only one device can use the line and transfer its data at one time.

Data in the transmission line would be arranged as A, B, C

DEMUX: select and route the data to their originate destination

$A \rightarrow D, B \rightarrow E, C \rightarrow F$

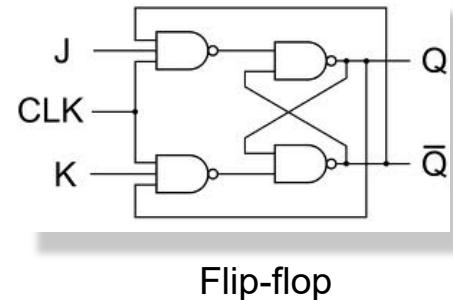


Problem:

Many inputs (e.g. A, B and C) wanted to use a single transmission line for their data transmission. How to make sure the data is transferred in a proper manner (issue of cost, synchronization, conflict , crash, loss?)

Source (A, B, C) and Destination (D, E, F)

$A \rightarrow D, B \rightarrow E, C \rightarrow F$



Data Storage Function

- Flip-flop stores a 1 or 0 only
- Registers
 - Formed by combining several flip-flops
 - 8-bit register → from 8 flip-flops
- Semiconductor Memories
 - e.g. RAM, ROM, Flash
- Magnetic/Optical Memories
 - For mass storage → e.g. hard disk, tape, DVD, Blu-Ray



Semiconductor
Memories



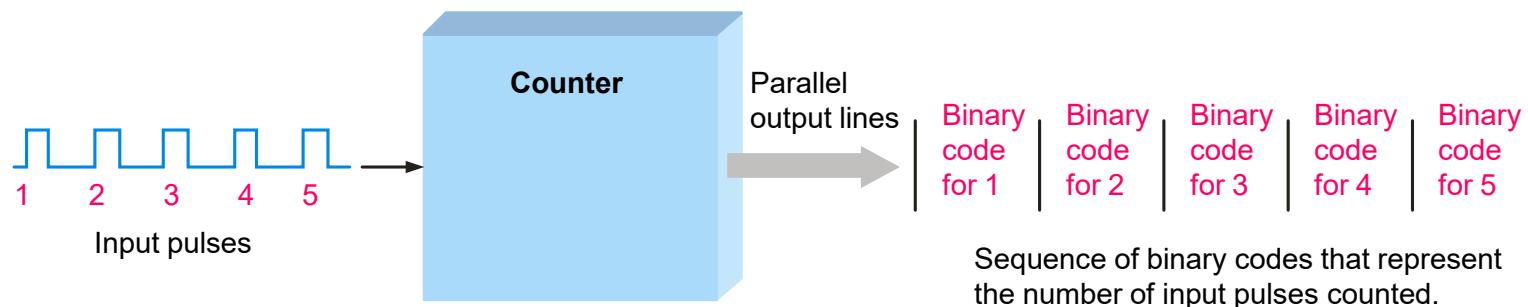
Optical Memories

Counting Function

Examples:

- Traffic light
- Washing machine
- Vending machine
- Xerox machine
- ATM machine
- etc.

- Counter
 - To count the occurrence at the input.
 - to initiate a controller after a certain count (period).

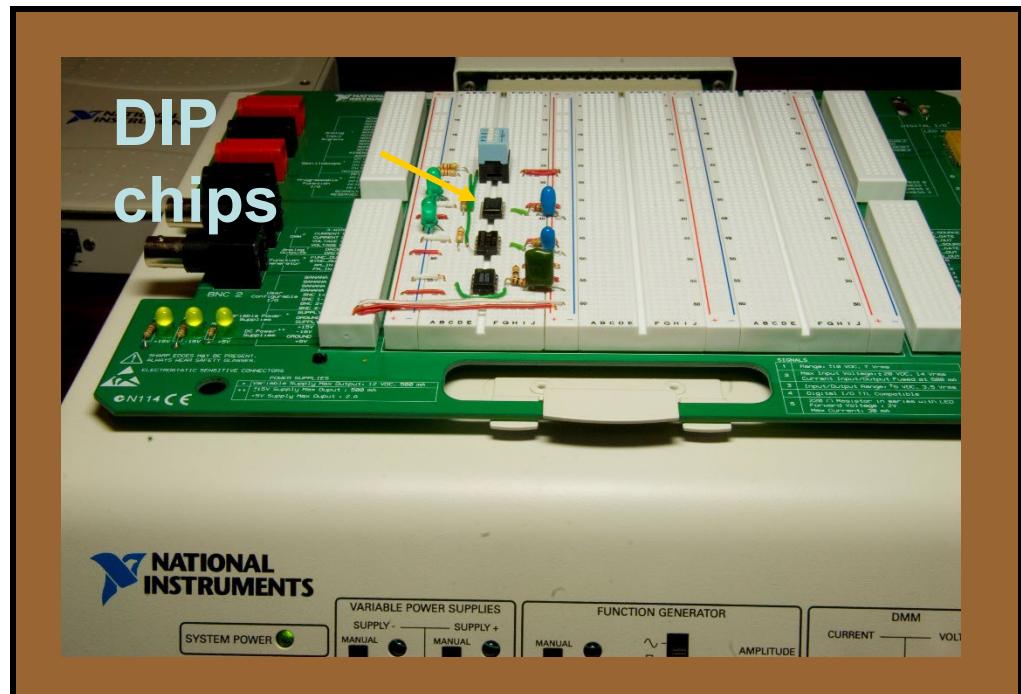




Fixed-Function Integrated Circuit (IC)

An example of laboratory prototyping is shown. The circuit is wired using DIP chips and tested.

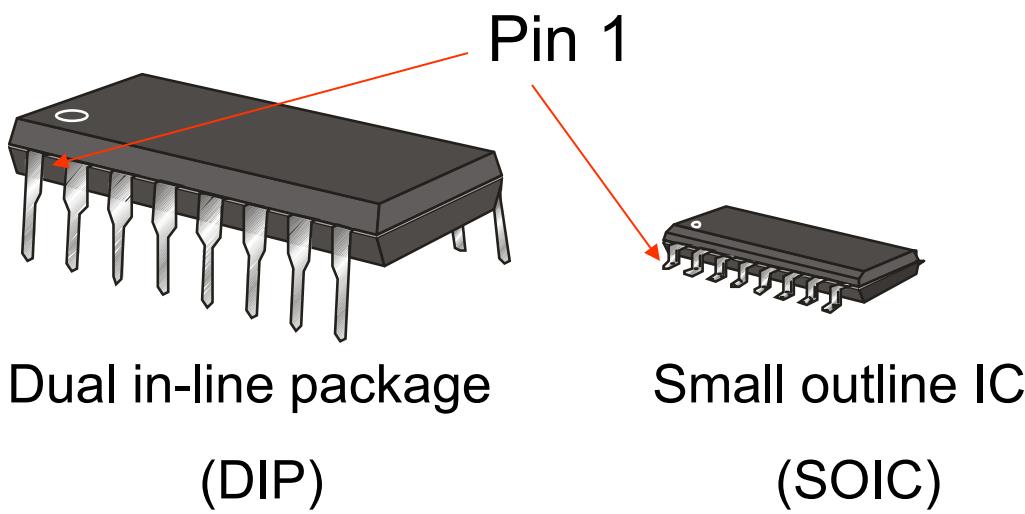
In this case, testing can be done by a computer connected to the system.



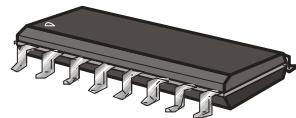
(Dual In-line Package)

IC Packages

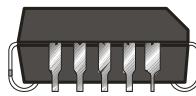
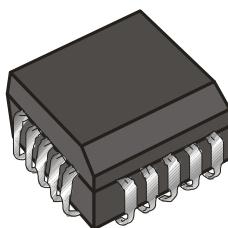
DIP chips and surface mount chips



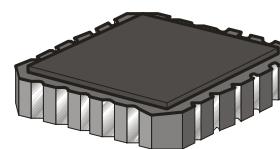
Other surface mount technology (SMT) packages:



SOIC
(Small-outline IC)

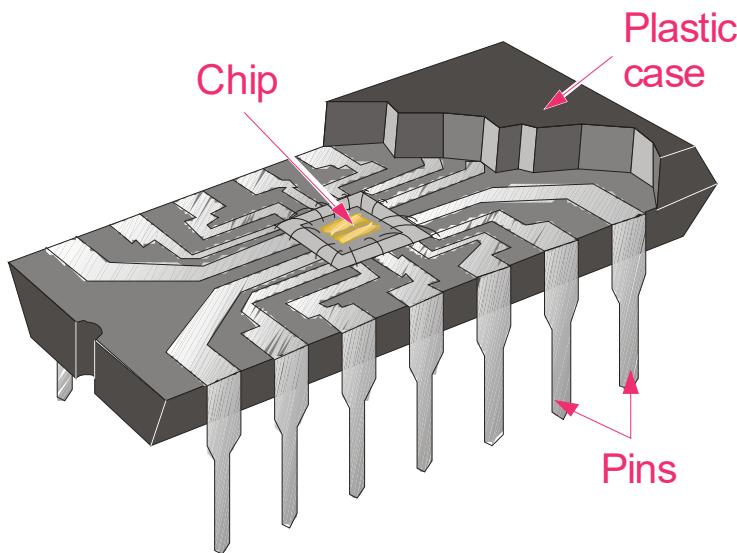


PLCC
(Plastic Leaded
Chip Carrier)



LCCC
(Leadless Ceramic
Chip Carrier)

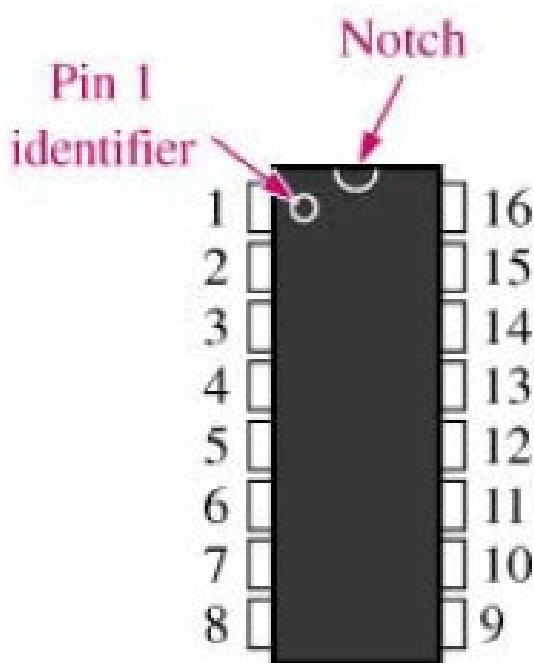
Cutaway view of DIP (Dual-In-line Pins) chip:



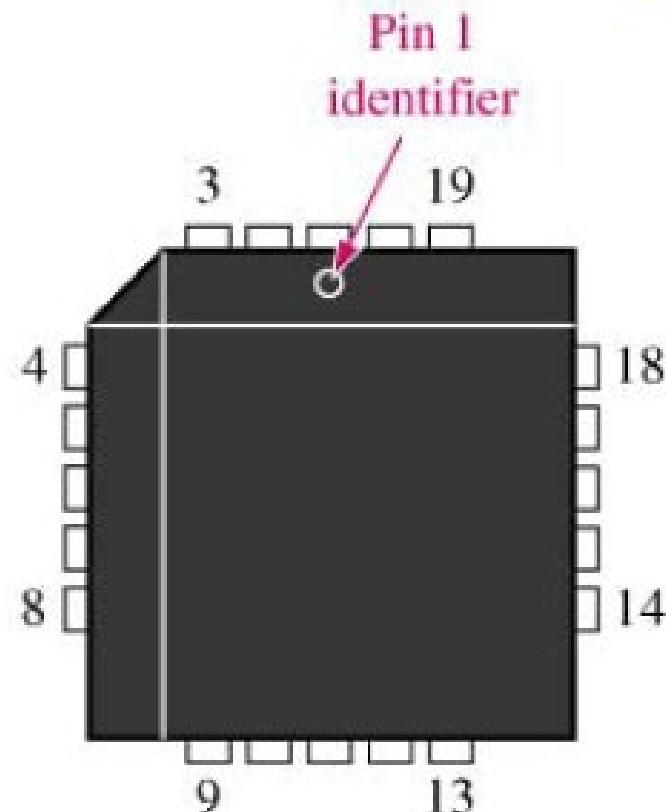
IC Packaging: Why we need packaging?

- To protect the IC (circuit)
- Have a pin system so that can connect to other circuit

Pin Numbering



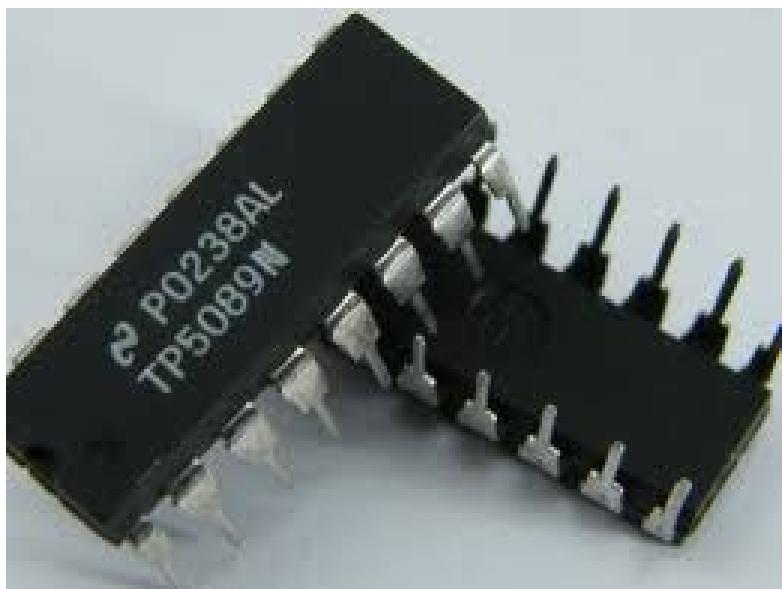
(a) DIP or SOIC



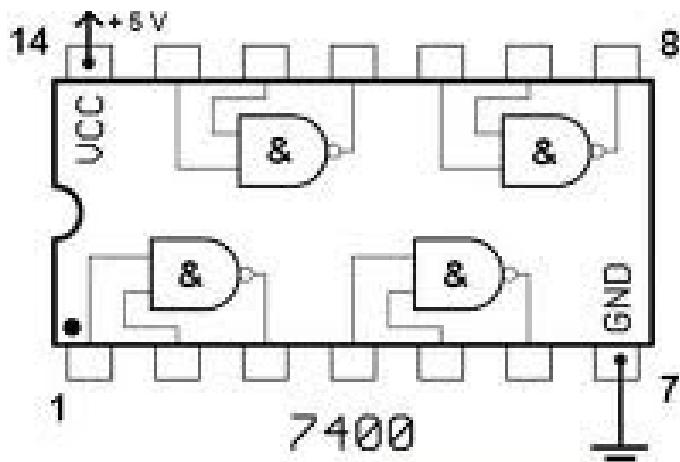
(b) PLCC or LCCC



<http://www.rkonlinestore.co.uk/556-dual-timer-ic-16-pin-dip-pack-of-4-391-p.asp>



<http://www.ebay.com/itm/10pcs-IC-TP5089N-DIP-16-PIN-TP5089-/310306081949>



<http://electroschematics.com/6529/7400-datasheet/>

Complexity Classifications for Fixed-Function ICs

- Small-scale integration (SSI)
have up to 12 gates on a single chip
- Medium-scale integration (MSI)
have from 12-99 gates on a single chip
- Large-scale integration (LSI)
have from 100-9999 gates on a single chip
- Very large-scale integration (VLSI)
have from 10,000-99,999 gates on a single chip
- Ultra large-scale integration (ULSI)
have from 100,000 and greater equivalent gates on a single chip



http://www.visual6502.org/images/263P_SI_263P_8404_chip1_package_top.jpg



<http://www.nysemagazine.com/lscorp>

Integrated Circuit Technologies

Some examples of IC technologies:

- TTL (*Transistor-transistor Logic*)
- ECL (*Emitter-Coupled Logic*)
- CMOS (*Complementary Metal–Oxide–Semiconductor*)
- NMOS (*N-Type Metal–Oxide–Semiconductor*)
- BiCMOS (*Bipolar and Metal–Oxide–Semiconductor*)



CMOS –

<http://www.creativeplanetnetwork.com/dcp/news/cmos-technology-primer/40995>



Programmable Logic Devices (PLD)

Overview of PLD

□ Fixed function

- A specific logic function is contained in the IC (hardwired) and can never be changed.

□ PLD

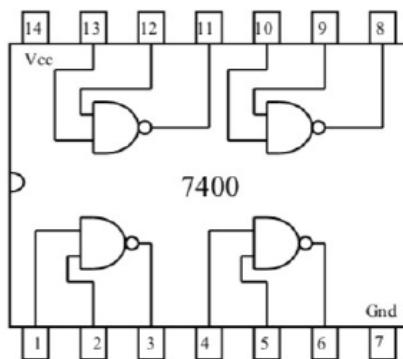
- Logic function programmed by the user.
 - Some, can be reprogrammed many times.

■ Advantage

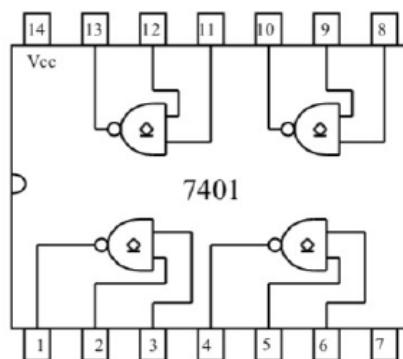
- More logic circuit can be ‘stuffed’ into much smaller area.
- Certain PLD, design can be changed without rewiring or replacing components.
- Can be implemented faster once the required programming language is mastered.

Example of fixed function ICs

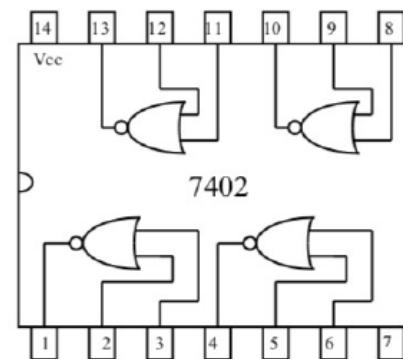
www.utm.my



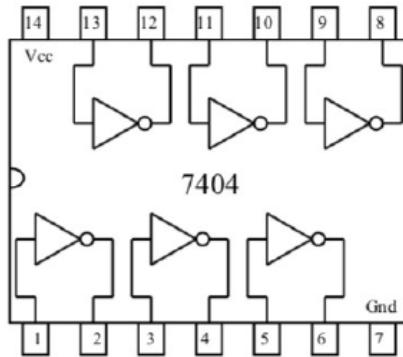
7400 Quad 2-input NAND



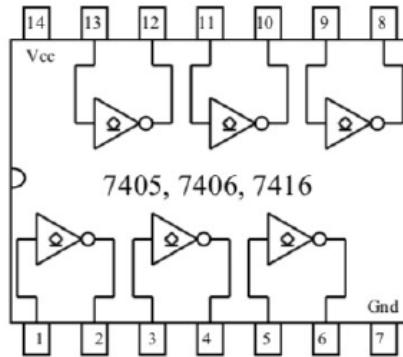
7401 Quad 2-input NAND
(open-collector outputs)



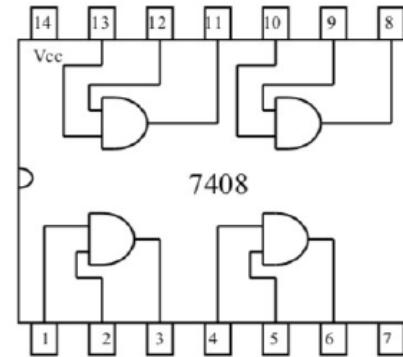
7402 Quad 2-input NOR



7404 Hex Inverter



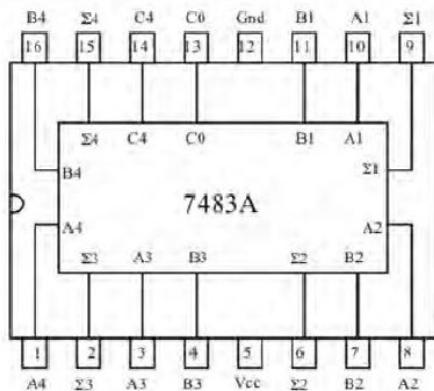
7405, 7406, 7416 Hex Inverter
(open-collector outputs)



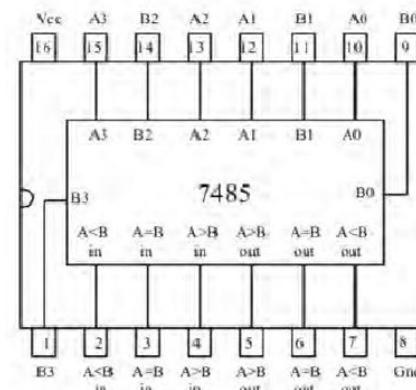
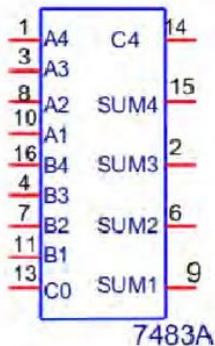
7408 Quad 2-input AND

Example of fixed function ICs

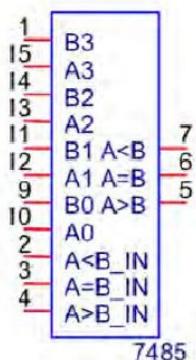
www.utm.my



7483A 4-Bit Adder: Pinout and Logic Symbol



7485 4-Bit Magnitude Comparator: Pinout and Logic Symbol



Types of PLD

3 major types (SPLD, CPLD, FPGA)

1. Simple Programmable Logic Devices (SPLD)

- Can replace several fixed-function SSI or MSI
- First type available
- A few categories
 - PAL (programmable Array Logic)
 - GAL (Generic Array Logic)
 - PLA (Programmable Logic Array)
 - PROM (Programmable Read-Only memory)

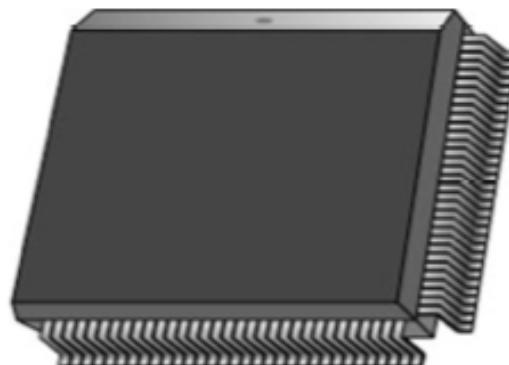
Types of PLD: CPLD

2. Complex Programmable Logic Devices (CPLD)

- Much higher capacity than SPLD (2-64 SPLD)
 - More complex logic circuits can be programmed
 - Typically in 44 – 160 pin package



(a) 84-pin PLCC package



(b) 128-pin PQFP package



Resource: http://upload.wikimedia.org/wikipedia/commons/thumb/a/a3/Altera_MAX_7128_2500_gate_CPLD.jpg/300px-Altera_MAX_7128_2500_gate_CPLD.jpg

Types of PLD: FPGA

- 3. Field-Programmable Gate Arrays (FPGA)
 - Different internal organization than SPLD and CPLD
 - Greatest logic capacity
 - Consist of 64-thousands logic block (logic gate groups)
 - Classes
 - Fine grain (smaller logic block)
 - Coarse grain (large logic block)

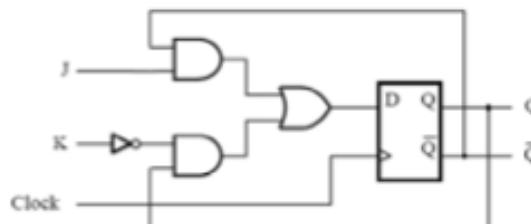


Resource: http://upload.wikimedia.org/wikipedia/commons/thumb/f/fa/Altera_StratixIVGX_FPGA.jpg/300px-Altera_StratixIVGX_FPGA.jpg

PLD Programming

- Logic circuit entered using 2 basic method

- Graphical entry
 - schematic diagram



- Text-based entry (language based entry)
 - Using Hardware Description Language (HDL)
 - Eg . ABEL, CUPL, WinCUPL
 - Becoming widely used especially for CPLD and FPGA
 - VHDL
 - Verilog

```
MODULE decoder
TITLE 'decoder'
A,B,C,D      pin    1,2,3,4;
W,X,Y,Z      pin    14,15,16,17;
equations
W=!B & C # !B & D # C & D # A;
X=!A & D # B # C;
Y=!A & !B & D # C;
Z=IB & C # D;
```