Computer Systems
- Hardware -

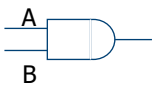
Objectives

- Understand the concept of electric and electronic circuits, which are components of the computer
- Understand the characteristics of typical methods for electronically controlling machines
- Understand the characteristics of components and the important points in performing logic design
- Understand the importance of power consumption in developing embedded devices

Logic Gates

- Logic gates are the basic building blocks of any digital system. It is an electronic circuit having one or more than one input and only one output. The relationship between the input and the output is based on a certain logical operation. Based on this, logic gates are named as AND gate, OR gate, NOT gate, etc.
- Logical operations refer to arithmetic operations that have only two truth values of True (1) and False (0)

AND gate



A	B	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

OR gate

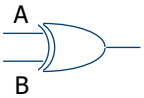
A	B	A OR B
0	0	0
0	1	1
1	0	1
1	1	1

NOT gate

A	NOT A
0	1
1	0

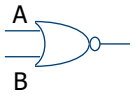
Logic Gates

EOR gate
(Exclusive logical sum operation)



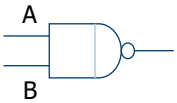
A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

NOR gate
(Negative logical sum operation)



A	B	A NOR B
0	0	1
0	1	0
1	0	0
1	1	0

NAND gate
(Negative logical product operation)



A	B	A NAND B
0	0	1
0	1	1
1	0	1
1	1	0

Combinational Circuit

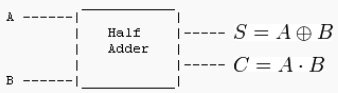
- Combinational circuit is a circuit in which we combine the different gates in the circuit, for example half-adder, full adder, encoder, decoder, multiplexer and demultiplexer. Some of the characteristics of combinational circuits are following −
  - The output of combinational circuit at any instant of time, depends only on the levels present at input terminals.
  - The combinational circuit do not use any memory. The previous state of input does not have any effect on the present state of the circuit.
  - A combinational circuit can have an *n* number of inputs and *m* number of outputs.

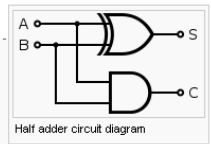
Block diagram

Combinational Circuit

- **Addition circuits**
  - **Half-adder**. It is a combinational logic circuit with two inputs and two outputs. The half adder circuit is designed to add two single bit binary number A and B. It is the basic building block for addition of two single bit numbers. This circuit has two outputs - carry and sum.

Schematic Symbol of Half Adder





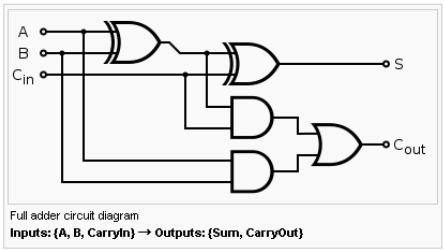
Half adder circuit diagram

S = A ⊕ B
C = A • B

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

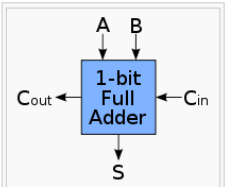
Combinational Circuit

- **Full adder.** It is developed to overcome the drawback of Half Adder circuit. It can add two one-bit numbers A and B, and carry c. The full adder is a three input and two output combinational circuit.



Full adder circuit diagram

Inputs: {A, B, CarryIn} → Outputs: {Sum, CarryOut}



Schematic symbol for a 1-bit full adder

Input			Output	
A	B	C_i	C_o	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

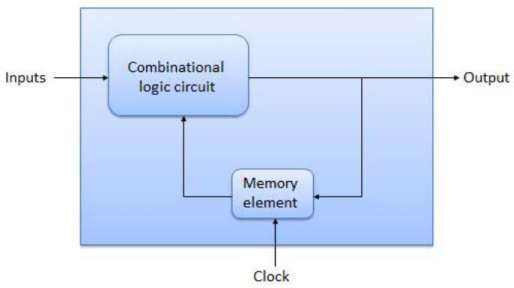
$$S = (A \oplus B) \oplus C_{in}$$

$$C_{out} = (A \cdot B) + (C_{in} \cdot (A \oplus B)) = (A \cdot B) + (C_{in} \cdot B) + (C_{in} \cdot A)$$

Sequential Circuit

- A sequential circuit has memory so output can vary based on input. This type of circuit uses previous input, output, clock, and a memory element.

Block diagram



Sequential Circuit

Key	Synchronous Sequential Circuits	Asynchronous Sequential Circuits
Definition	Synchronous sequential circuits are digital sequential circuits in which the feedback to the input for next output generation is governed by clock signals.	On other hand Asynchronous sequential circuits are digital sequential circuits in which the feedback to the input for next output generation is not governed by clock signals.
Memory Unit	In Synchronous sequential circuits, the memory unit which is being get used for governance is clocked flip flop.	On other hand unclocked flip flop or time delay is used as memory element in case of Asynchronous sequential circuits.
State	The states of Synchronous sequential circuits are always predictable and thus reliable.	On other hand there are chances for the Asynchronous circuits to enter into a wrong state because of the time difference between the arrivals of inputs. This is called as race condition.
Complexity	It is easy to design Synchronous sequential circuits	However on other hand the presence of feedback among logic gates causes instability issues making the design of Asynchronous sequential circuits difficult.
Performance	Due to the propagation delay of clock signal in reaching all elements of the circuit the Synchronous sequential circuits are slower in its operation speed	Since there is no clock signal delay, these are fast compared to the Synchronous Sequential Circuits
Example	Synchronous circuits are used in counters, shift registers, memory units.	On other hand Asynchronous circuits are used in low power and high speed operations such as simple microprocessors, digital signal processing units and in communication systems for email applications, internet access and networking.

Sequential Circuit

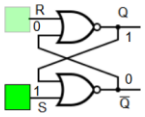
- Flip-flops and latches are used as data storage elements. A flip-flop is a device which stores a single bit (binary digit) of data; one of its two states represents a "one" and the other represents a "zero". Such data storage can be used for storage of state, and such a circuit is described as sequential logic in electronics.

- Flip-flops can be either **level-triggered** (asynchronous, transparent or opaque) or **edge-triggered** (synchronous, or clocked) circuit that store a single bit of data using gates. Recently, some authors reserve the term flip-flop exclusively for discussing clocked circuits; the simple ones are commonly called transparent latches. Using this terminology, a level-sensitive flip-flop is called a transparent latch, whereas an edge-triggered flip-flop is simply called a flip-flop

- Flip-flops can be divided into common types: the **SR** ("set-reset"), **D** ("data" or "delay"), **T** ("toggle"), and **JK**. The behavior of a particular type can be described by what is termed the characteristic equation, which derives the "next" (i.e., after the next clock pulse) output, *Qnext* in terms of the input signal(s), and/or the current output, *Q*.

SR FLIP-FLOP

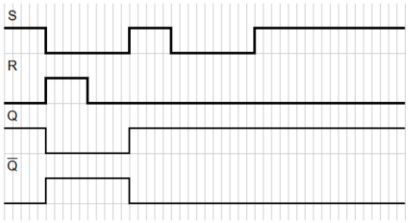
- When using static gates as building blocks, the most fundamental latch is the simple SR latch, where S and R stand for set and reset. It can be constructed from a pair of cross-coupled NOR or NAND logic gates. The stored bit is present on the output marked Q.



TRUTH TABLE

INPUTS		OUTPUTS	
S	R	Q	$\overline{Q}$
0	0	Q_0	$\overline{Q}_0$
0	1	0	1
1	0	1	0
1	1	X	X

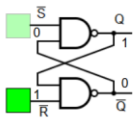
- **SR NOR LATCH**
  - While the R and S inputs are both low, feedback maintains the Q and Q' outputs in a constant state, with Q' the complement of Q. If S (*Set*) is pulsed high while R (*Reset*) is held low, then the Q output is forced high, and stays high when S returns to low; similarly, if R is pulsed high while S is held low, then the Q output is forced low, and stays low when R returns to low.



  - The R = S = 1 combination is called a restricted combination or a forbidden state because, as both NOR gates then output 0s, it breaks the logical equation Q = not Q. The combination is also inappropriate in circuits where both inputs may go low simultaneously (i.e. a transition from restricted to keep). The output would lock at either 1 or 0 depending on the propagation time relations between the gates (a race condition). In certain implementations, it could also lead to longer ringings (damped oscillations) before the output settles, and thereby result in undetermined values.

SR FLIP-FLOP

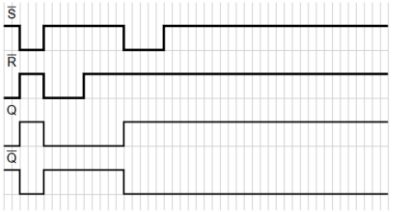
- **SR NAND LATCH**
  - The inputs are generally designated S and R for Set and Reset respectively. Because the NAND inputs must normally be logic 1 to avoid affecting the latching action, the inputs are considered to be inverted in this circuit (or active low).



TRUTH TABLE

INPUTS		OUTPUTS	
$\overline{S}$	$\overline{R}$	Q	$\overline{Q}$
0	0	X	X
0	1	1	0
1	0	0	1
1	1	Q_0	$\overline{Q}_0$

  - The circuit uses feedback to "remember" and retain its logical state even after the controlling input signals have changed. When the S and R inputs are both high, feedback maintains the Q outputs to the previous state.



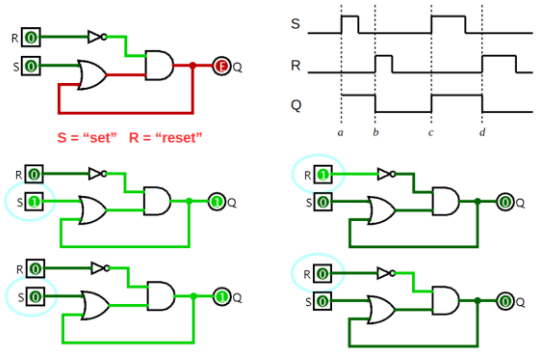
  - The R = S = 0 combination is called a restricted combination or a forbidden state because, as both NAND gates then output 1s, it breaks the logical equation Q = not Q. The combination is also inappropriate in circuits where both inputs may go high simultaneously (i.e. a transition from restricted to keep). The output would lock at either 1 or 0 depending on the propagation time relations between the gates (a race condition). In certain implementations, it could also lead to longer ringings (damped oscillations) before the output settles, and thereby result in undetermined values.

SR FLIP-FLOP

- **SR AND-OR LATCH**
  - The SR AND-OR latch is easier to understand, because both gates can be explained in isolation. When neither S or R is set, then both the OR gate and the AND gate are in "hold mode", i.e., their output is the input from the feedback loop.
  - When input S = 1, then the output of the OR gate becomes 1, regardless of the other input from the feedback loop ("set mode"). When input R = 1, then the output of the AND gate becomes 0, regardless of the other input from the feedback loop ("reset mode"). And since the output Q is directly connected to the output of the AND gate, R has priority over S. Latches drawn as cross-coupled gates may look less intuitive, as the behavior of one gate appears to be intertwined with the other gate.

SR AND-OR latch operation

S	R	Action
0	0	No change; random initial
1	0	Q = 1
X	1	Q = 0

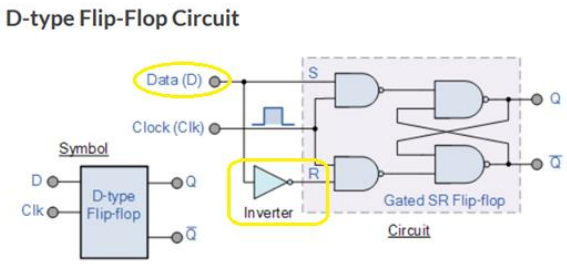


Page 13

D FLIP-FLOP

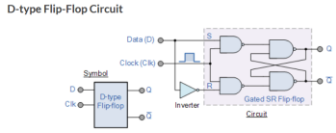
- The D Flip Flop is by far the most important of the clocked flip-flops as it ensures that inputs S and R are never equal to one at the same time. The D-type flip flop are constructed from a gated SR flip-flop with an inverter added between the S and the R inputs to allow for a single D (Data) input.

- This single data input, labelled "D" is used in place of the "Set" signal, and the inverter is used to generate the complementary "Reset" input thereby making a level-sensitive D-type flip-flop from a level-sensitive SR-latch as now S = D and R = not D as shown.



Page 14

D FLIP-FLOP



- Remember that a simple SR flip-flop requires two inputs, one to “SET” the output and one to “RESET” the output. By connecting an inverter (NOT gate) to the SR flip-flop we can “SET” and “RESET” the flip-flop using just one input as now the two input signals are complements of each other. This complement avoids the ambiguity inherent in the SR latch when both inputs are LOW, since that state is no longer possible.

- Thus this single input is called the “DATA” input. If this data input is held HIGH the flip flop would be “SET” and when it is LOW the flip flop would change and become “RESET”. However, this would be rather pointless since the output of the flip flop would always change on every pulse applied to this data input.

- To avoid this, an additional input called the “CLOCK” or “ENABLE” input is used to isolate the data input from the flip flop’s latching circuitry after the desired data has been stored. The effect is that D input condition is only copied to the output Q when the clock input is active. This then forms the basis of another sequential device called a D Flip Flop.

D FLIP-FLOP

- The “D flip flop” will store and output whatever logic level is applied to its data terminal so long as the clock input is HIGH.
- Once the clock input goes LOW the “set” and “reset” inputs of the flip-flop are both held at logic level “1” so, it will not change state and store whatever data was present on its output before the clock transition occurred. In other words, the output is “latched” at either logic “0” or logic “1”.

Truth Table for the D-type Flip Flop

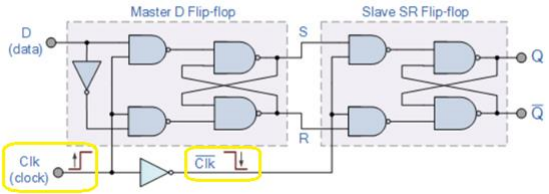
Clk	D	Q	$\overline{Q}$	Description
↓ » 0	X	Q	$\overline{Q}$	Memory no change
↑ » 1	0	0	1	Reset Q » 0
↑ » 1	1	1	0	Set Q » 1

Note that: ↓ and ↑ indicates direction of clock pulse as it is assumed D-type flip flops are edge triggered

D FLIP-FLOP

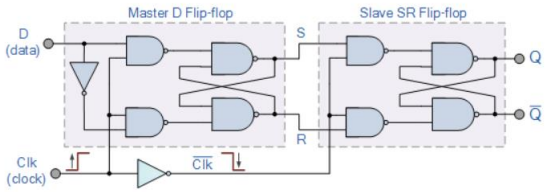
- **The Master-Slave D Flip Flop**
  - The basic D-type flip flop can be improved further by adding a second SR flip-flop to its output that is activated on the complementary clock signal to produce a “Master-Slave D-type flip flop”. On the leading edge of the clock signal (LOW-to-HIGH) the first stage, the “master” latches the input condition at D, while the output stage is deactivated.
  - On the trailing edge of the clock signal (HIGH-to-LOW) the second “slave” stage is now activated, latching on to the output from the first master circuit. Then the output stage appears to be triggered on the negative edge of the clock pulse. “Master-Slave D-type flip flops” can be constructed by the cascading together of two latches with opposite clock phases as shown.

The Master-Slave D Flip Flop Circuit



D FLIP-FLOP

The Master-Slave D Flip Flop Circuit

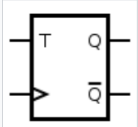


- The leading edge of the clock pulse the master flip-flop will be loading data from the data D input, therefore the master is “ON”. With the trailing edge of the clock pulse the slave flip-flop is loading data, i.e. the slave is “ON”. Then there will always be one flip-flop “ON” and the other “OFF” but never both the master and slave “ON” at the same time. Therefore, the output Q acquires the value of D, only when one complete pulse, i.e., 0-1-0 is applied to the clock input.

T FLIP-FLOP

- The “T” in “T flip-flop” stands for “toggle.” If the T input is high, the T flip-flop changes state ("toggles") whenever the clock input is strobed. If the T input is low, the flip-flop holds the previous value. This behavior is described by the characteristic equation:

$$Q_{\text{next}} = T \oplus Q = T\overline{Q} + \overline{T}Q \text{ (expanding the XOR operator)}$$



A circuit symbol for a T-type flip-flop

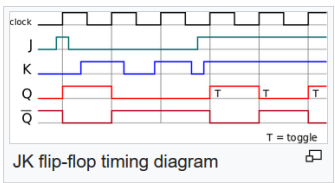
T flip-flop operation

Characteristic table				Excitation table			
T	Q	Q_{next}	Comment	Q	Q_{next}	T	Comment
0	0	0	Hold state (no clock)	0	0	0	No change
0	1	1	Hold state (no clock)	1	1	0	No change
1	0	1	Toggle	0	1	1	Complement
1	1	0	Toggle	1	0	1	Complement

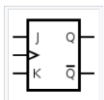
- When T is held high, the toggle flip-flop divides the clock frequency by two; that is, if clock frequency is 4 MHz, the output frequency obtained from the flip-flop will be 2 MHz. This "divide by" feature has application in various types of digital counters. A T flip-flop can also be built using a JK flip-flop (J & K pins are connected together and act as T) or a D flip-flop (T input XOR Q_{previous} drives the D input).

JK FLIP-FLOP

- The JK flip-flop augments the behavior of the SR flip-flop (J: Set, K: Reset) by interpreting the J = K = 1 condition as a "flip" or toggle command.
  - J = 1, K = 0 is a command to set the flip-flop;
  - J = 0, K = 1 is a command to reset the flip-flop;
  - J = K = 1 is a command to toggle the flip-flop, i.e., change its output to the logical complement of its current value
  - J = K = 0 maintains the current state
- To synthesize a D flip-flop, simply set K equal to the complement of J (input J will act as input D). Similarly, to synthesize a T flip-flop, set K equal to J. The JK flip-flop is therefore a universal flip-flop, because it can be configured to work as an SR flip-flop, a D flip-flop, or a T flip-flop.



JK flip-flop timing diagram



A circuit symbol for a positive-edge-triggered JK flip-flop

JK flip-flop operation

Characteristic table				Excitation table				
J	K	Comment	Q_{next}	Q	Q_{next}	Comment	J	K
0	0	Hold state	Q	0	0	No change	0	X
0	1	Reset	0	0	1	Set	1	X
1	0	Set	1	1	0	Reset	X	1
1	1	Toggle	$\overline{Q}$	1	1	No change	X	0

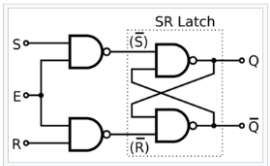
The characteristic equation of the JK flip-flop is:

$$Q_{\text{next}} = J\overline{Q} + \overline{K}Q$$

Gated Latches and Conditional Transparency

- Latches are designed to be transparent. That is, input signal changes cause immediate changes in output. Additional logic can be added to a simple transparent latch to make it non-transparent or opaque when another input (an "enable" input) is not asserted. When several transparent latches follow each other, using the same enable signal, signals can propagate through all of them at once. However, by following a transparent-high latch with a transparent-low (or opaque-high) latch, a master–slave flip-flop is implemented.

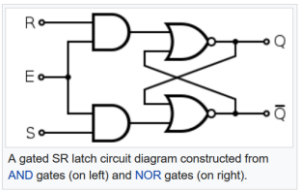
- **GATED SR LATCH**
  - A synchronous SR latch (sometimes clocked SR flip-flop) can be made by adding a second level of NAND gates to the inverted SR latch (or a second level of AND gates to the direct SR latch). The extra NAND gates further invert the inputs so SR latch becomes a gated SR latch (and a SR latch would transform into a gated SR latch with inverted enable).



NAND Gated SR Latch (Clocked SR flip-flop). Note the inverted inputs.

Gated Latches and Conditional Transparency

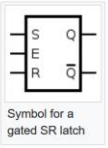
- **GATED SR LATCH**



A gated SR latch circuit diagram constructed from AND gates (on left) and NOR gates (on right).

Gated SR latch operation

E/C	Action
0	No action (keep state)
1	The same as non-clocked SR latch



Symbol for a gated SR latch

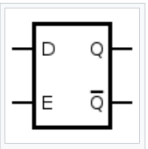
  - With E high (enable true), the signals can pass through the input gates to the encapsulated latch; all signal combinations except for (0, 0) = hold then immediately reproduce on the (Q, Q') output, i.e. the latch is transparent.
  - With E low (enable false) the latch is closed (opaque) and remains in the state it was left the last time E was high.
  - The enable input is sometimes a clock signal, but more often a read or write strobe. When the enable input is a clock signal, the latch is said to be level-sensitive (to the level of the clock signal), as opposed to edge-sensitive like flip-flops

Gated Latches and Conditional Transparency

- **GATED D LATCH**
  - A gated D latch has just two inputs: DATA and ENABLE. When a HIGH is received at the ENABLE input, the DATA input is copied to the output. Even if the ENABLE input then goes low, the output remains unchanged. The output cannot be changed until the ENABLE input goes high.
  - The truth table below shows that when the enable/clock input is 0, the D input has no effect on the output. When E/C is high, the output equals D.

Gated D latch truth table

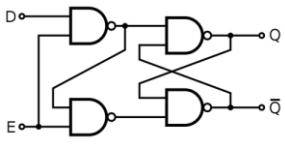
E/C	D	Q	$\overline{Q}$	Comment
0	X	Q_{prev}	$\overline{Q}_{prev}$	No change
1	0	0	1	Reset
1	1	1	0	Set



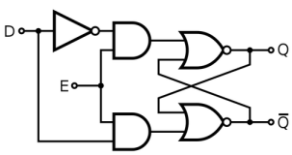
Symbol for a gated D latch

Gated Latches and Conditional Transparency

- **GATED D LATCH**
  - This latch exploits the fact that, in the two active input combinations (01 and 10) of a gated SR latch, R is the complement of S. The input NAND stage converts the two D input states (0 and 1) to these two input combinations for the next SR latch by inverting the data input signal. The low state of the enable signal produces the inactive "11" combination. Thus a gated D-latch may be considered as a *one-input synchronous SR latch*.
  - It is also known as *transparent latch*, *data latch*, or simply *gated latch*. It has a *data* input and an *enable* signal (sometimes named clock, or control). The word *transparent* comes from the fact that, when the enable input is on, the signal propagates directly through the circuit, from the input D to the output Q. Gated D-latches are also **level-sensitive** with respect to the level of the clock or enable signal.



A gated D latch based on an $\overline{SR}$ NAND latch



A gated D latch based on an SR NOR latch

Integrated Circuit

- Normally bipolar junction transistors, diodes and field effect transistors are commonly used electronics component in electronic circuit. These components are interconnected along with required resistors and capacitors to form an electronic circuit. This type of circuit is known as discrete circuit as each of the components can be separated from the circuit as when required. Nowadays there is a new trend of producing electronic circuit where on a semiconductor wafer numbers of diodes, transistors, and capacitors are permanently fabricated.
- As the components in this type of electronic circuit are not separable that is integrated on the semiconductor wafer, this circuit is commonly referred to as an Integrated Circuit. IC is also popularly known as chip or microchip.
- Two main types of integrated circuits:
  1. **Analog IC**. In this type of ICs, the input and output both signals are continuous. The output signal level depends upon the input signal level and the output signal level is a linear function of input signal level. Linear ICs or analog ICs are most commonly used as audio frequency amplifier and radio frequency amplifier. Op amps, voltage regulators, comparators and timers are also well-known examples of linear ICs or analog ICs.
  2. **Digital IC**. The logic Gates, such as AND gate, OR gate, NAND gate, XOR gate, flip flops, counters; microprocessors are some well-known examples of digital ICs. These ICs operate with binary data such as either 0 or 1. Normally in digital circuit, 0 indicates 0 V and one indicate +5 V. Digital ICs are commonly used in many electronics projects, and are often available as added components to the top Arduino starter kits.

Integrated Circuit

- The main components of an IC are transistors. These transistors may be bipolar or field effect depending upon the applications of ICs.
- As the technology is improving day by day, the number of transistors incorporated in a single IC chip is also increasing. Depending upon the number of transistors incorporated in a single chip, the ICs are categorized in five groups. Namely,
  1. Small Scale Integration (SSI) where the number of transistors incorporated in a single IC chip is from 1 to 10.
  2. Medium Scale Integration (MSI) where the number of transistors incorporated in a single IC chip is from 10 to 500.
  3. Large Scale Integration (LSI) where the number of transistors incorporated in a single IC chip is from 500 to 20,000.
  4. Very Large Scale Integration (VLSI) where the number of transistors incorporated in a single IC chip is from 20,000 to 1,000,000.
  5. Ultra Large Scale Integration (ULSI) where the number of transistors incorporated in a single IC chip is from 1,000,000 and more.
- Depending upon the active devices used in ICs, it can be further classified as **bipolar ICs** and **unipolar ICs**. In bipolar ICs, the main components are bipolar junction transistors, whereas in unipolar ICs, the main components are field effect transistors or MOSFETs.

Integrated Circuit

- The advantages of integrated circuits (ICs) include:
  1. It is quite small in size practically around 20,000 electronic components can be incorporated in a single square inch of IC chip.
  2. Many complex circuits are fabricated in a single chip and hence this simplifies the designing of a complex electronic circuit. Also it improves the performance.
  3. Reliability of ICs is high
  4. These are available at low cost due to bulk production.
  5. ICs consume very tiny power.
  6. Higher operating speed due to absence of parasitic capacitance effect.
  7. Very easily replaceable from the mother circuit.

- The disadvantages of integrated circuits (ICs) include:
  1. Because of its small size, IC is unable to dissipate heat in required rate when current in it increased. That is why ICs are often damaged due to over current flowing through them.
  2. Inductors and Transformers cannot be incorporated in ICs.

Integrated Circuit

- **Terminologies**:
  - **Resistors** - a passive component in a circuit which provides resistance to the flow of current.
  - **Capacitor** – In an electrical circuit, it behaves as a charge storage device. It holds the electric charge when we apply a voltage across it, and it gives up the stored charge to the circuit as when required.
  - **Diode** - defined as a two-terminal electronic component that only conducts current in one direction (so long as it is operated within a specified voltage level). An ideal diode will have zero resistance in one direction, and infinite resistance in the reverse direction.
  - **Transistors** - is a semiconductor device used to amplify or switch electronic signals and electrical power.
  - **Semiconductor** - the materials that are neither conductor nor insulator with energy gap of about 1 eV (electron volt).
  - **Operational Amplifier** (Op amps) - is used to denote an amplifier which can be configured to perform various operations like amplification, subtraction, differentiation, addition, integration etc.
  - **MOSFET** - stands for Metal Oxide Semiconductor Field Effect Transistor. The MOSFET is a capacitor operated transistor device. They are widely used for automotive, industrial and communications systems in particular.

Integrated Circuit

- **System on Chip (SoC)**
  - It is an integrated circuit (also known as a "chip") that integrates all or most components of a computer or other electronic system. These components almost always include a central processing unit (CPU), memory, input/output ports and secondary storage – all on a single substrate or microchip, the size of a coin. It may contain digital, analog, mixed-signal, and often radio frequency signal processing functions (otherwise it is considered only an application processor).

The Raspberry Pi uses a system on a chip as an almost fully contained microcomputer. This SoC does not contain any kind of data storage, which is common for a microprocessor SoC.

7-Segment Display

7-segment Display

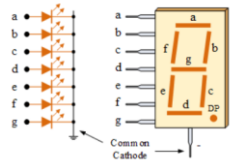
- The 7-segment display, consists of seven LEDs arranged in a rectangular fashion as shown. Each of the seven LEDs is called a segment because when illuminated the segment forms part of a numerical digit (both Decimal and Hex) to be displayed. An additional 8th LED is sometimes used within the same package thus allowing the indication of a decimal point.

- Each one of the seven LEDs in the display is given a positional segment with one of its connection pins being brought straight out of the rectangular plastic package. These individually LED pins are labelled from **a** through to **g** representing each individual LED. The other LED pins are connected together and wired to form a common pin.
- The displays common pin is generally used to identify which type of 7-segment display it is. As each LED has two connecting pins, one called the “Anode” and the other called the “Cathode”, there are therefore two types of LED 7-segment display called: Common Cathode (CC) and Common Anode (CA).

7-Segment Display

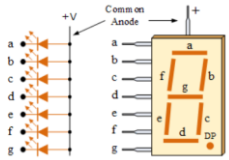
- Common Cathode (CC) – In the common cathode display, all the cathode connections of the LED segments are joined together to logic “0” or ground. The individual segments are illuminated by application of a “HIGH”, or logic “1” signal via a current limiting resistor to forward bias the individual Anode terminals (a-g).

Common Cathode 7-segment Display



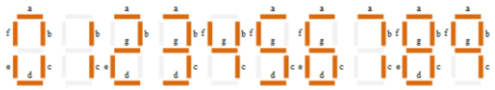
- Common Anode (CA) – In the common anode display, all the anode connections of the LED segments are joined together to logic “1”. The individual segments are illuminated by applying a ground, logic “0” or “LOW” signal via a suitable current limiting resistor to the Cathode of the particular segment (a-g).

Common Anode 7-segment Display



7-Segment Display

7-Segment Display Segments for all Numbers.



7-segment Display Truth Table

Decimal Digit	Individual Segments Illuminated						
	a	b	c	d	e	f	g
0	×	×	×	×	×	×	
1		×	×				
2	×	×		×	×		×
3	×	×	×	×			×
4		×	×			×	×
5	×		×	×		×	×
6	×		×	×	×	×	×
7	×	×	×				
8	×	×	×	×	×	×	×
9	×	×	×			×	×

- For a 7-segment display, we can produce a truth table giving the individual segments that need to be illuminated in order to produce the required decimal digit from 0 through 9 as shown in the image.

- In most practical applications, 7-segment displays are driven by a suitable decoder/driver IC such as the CMOS 4511 or TTL 7447 from a 4-bit BCD (Binary Coded Decimal) input. Today, LED based 7-segment displays have been largely replaced by liquid crystal displays (LCDs) which consume less current.

References

- Internet Sources:
  - https://www.electronics-tutorials.ws/blog/7-segment-display-tutorial.html
  - https://www.tutorialspoint.com/computer_logical_organization/logic_gates.htm
  - https://www.tutorialspoint.com/computer_logical_organization/combinational_circuits.htm
  - https://www.tutorialspoint.com/computer_logical_organization/sequential_circuits.htm
  - https://www.tutorialspoint.com/difference-between-synchronous-and-asynchronous-sequential-circuits
  - http://electronics-course.com/sr-nor-latch
  - http://electronics-course.com/sr-nand-latch
  - https://w3.cs.jmu.edu/lam2mo/cs261_2016_08/files/19_seq_circuits.pdf
  - https://www.electronics-tutorials.ws/sequential/seq_4.html
  - https://www.dummies.com/programming/electronics/diy-projects/electronics-basics-what-is-a-gated-latch/
  - https://www.electrical4u.com/integrated-circuits-types-of-ic/
  - https://en.wikipedia.org/wiki/System_on_a_chip

References

- Videos:
  - 7-segment Displays
    - https://www.youtube.com/watch?v=Irgz_qy27qc
    - https://www.youtube.com/watch?v=k9lDX9_MUUM
  - Latches and Flip-Flops
    - https://www.youtube.com/watch?v=-aQH0ybMd3U&t=297s
    - https://www.youtube.com/watch?v=HxAhOETcvr4
    - https://www.youtube.com/watch?v=y7Zf7Bv_J74
    - https://www.youtube.com/watch?v=8bUKw2cGcGg
    - https://www.youtube.com/watch?v=RP00R9uf004
  - NESO Academy Video Tutorials:
    - LATCHES & FLIP-FLOPS
      - https://www.youtube.com/watch?v=AaN72s5WfOM&list=PLuYnCh-Sh1XdvuSGjQRi2jgUH9_CiVR8J&index=2&t=0s
      - https://www.youtube.com/watch?v=Pi_MHyMoenA
      - https://www.youtube.com/watch?v=wx0NyUfpm48
      - https://www.youtube.com/watch?v=m1QBxTeVaNs
      - https://www.youtube.com/watch?v=mXoQ4WAQ0qk
      - https://www.youtube.com/watch?v=iaIu5SYmWVM&t=290s
    - SR FLIP-FLOPS
      - https://www.youtube.com/watch?v=HZg7fNu-I24
      - https://www.youtube.com/watch?v=uiKKRPZbuXA
      - https://www.youtube.com/watch?v=kt8d3CYWGH4
    - JK FLIP-FLOPS
      - https://www.youtube.com/watch?v=j6krFp511HA
      - https://www.youtube.com/watch?v=lnQD2_M9uDI
    - D FLIP-FLOPS
      - https://www.youtube.com/watch?v=dnfXXpW7tlw
      - https://www.youtube.com/watch?v=4c6z9RKrC8Q
    - T FLIP-FLOPS
      - https://www.youtube.com/watch?v=wcfnEla_Y78
      - https://www.youtube.com/watch?v=9mh-2QwxmF4