In electronics, a flip-flop or latch is a circuit that has two stable states and can be used to store state information. A flip-flop is a bistable multivibrator. The circuit can be made to change state by signals applied to one or more control inputs and will have one or two outputs. It is the basic storage element in sequential logic. Flip-flops and latches are fundamental building blocks of digital electronics systems used in computers, communications, and many other types of systems.

Flip-flops and latches are used as data storage elements. A flip-flop stores a single bit (binary digit) of data; one of its two states represents a "one" and the other represents a "zero".

SR Flip-Flop

The **SR flip-flop**, also known as a *SR Latch*, can be considered as one of the most basic sequential logic circuit possible. This simple flip-flop is basically a one-bit memory bistable device that has two inputs, one which will "SET" the device (meaning the output = "1"), and is labelled S and another which will "RESET" the device (meaning the output = "0"), labelled R.

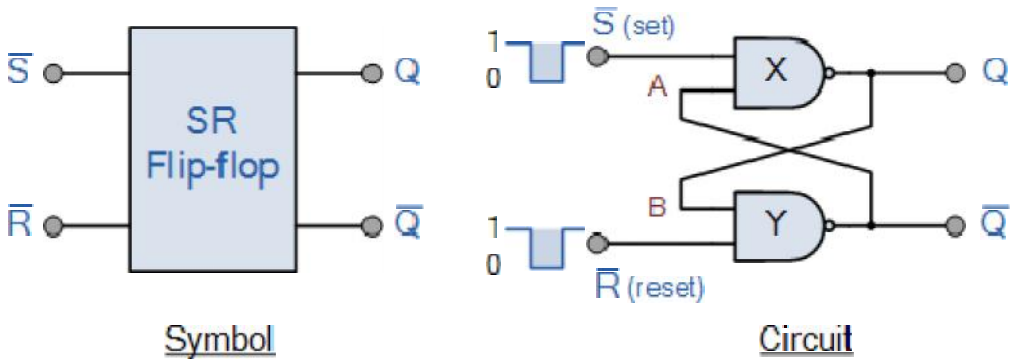
Then the SR description stands for "Set-Reset". The reset input resets the flip-flop back to its original state with an output Q that will be either at a logic level "1" or logic "0" depending upon this set/reset condition.

A basic NAND gate SR flip-flop circuit provides feedback from both of its outputs back to its opposing inputs and is commonly used in memory circuits to store a single data bit. Then the SR flip-flop actually has three inputs, Set, Reset and its current output Qrelating to it's current state or history. The term "Flip-flop" relates to the actual operation of the device, as it can be "flipped" into one logic Set state or "flopped" back into the opposing logic Reset state.

The NAND Gate SR Flip-Flop

The simplest way to make any basic single bit set-reset SR flip-flop is to connect together a pair of cross-coupled 2-input NAND gates as shown, to form a Set-Reset Bistable also known as an active LOW SR NAND Gate Latch, so that there is feedback from each output to one of the other NAND gate inputs. This device consists of two inputs, one called the *Set*, S and the other called the *Reset*, R with two corresponding outputs Q and its inverse or complement Q (not-Q) as shown below.

The Basic SR Flip-flop



Symbol Circuit

The Set State

Consider the circuit shown above. If the input R is at logic level "0" (R = 0) and input S is at logic level "1" (S = 1), the NAND gate *Y* has at least one of its inputs at logic "0" therefore, its output Q must be at a logic level "1" (NAND Gate principles). Output Q is also fed back to input "A" and so both inputs to NAND gate *X* are at logic level "1", and therefore its output Q must be at logic level "0".

Again NAND gate principals. If the reset input R changes state, and goes HIGH to logic "1" with S remaining HIGH also at logic level "1", NAND gate *Y* inputs are now R = "1" and B = "0". Since one of its inputs is still at logic level "0" the output at Q still remains HIGH at logic level "1" and there is no change of state. Therefore, the flip-flop circuit is said to be "Latched" or "Set" with Q = "1" and Q = "0".

Reset State

In this second stable state, Q is at logic level "0", (not Q = "0") its inverse output at Q is at logic level "1", (Q = "1"), and is given by R = "1" and S = "0". As gate *X* has one of its inputs at logic "0" its output Q must equal logic level "1" (again NAND gate principles). Output Q is fed back to input "B", so both inputs to NAND gate *Y* are at logic "1", therefore, Q = "0".

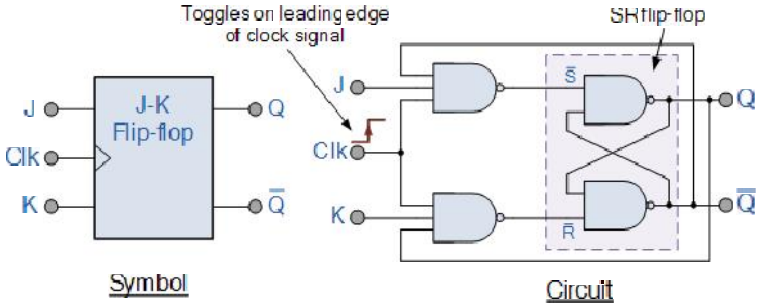
If the set input, S now changes state to logic "1" with input R remaining at logic "1", output Q still remains LOW at logic level "0" and there is no change of state. Therefore, the flip-flop circuits "Reset" state has also been latched and we can define this "set/reset" action in the following truth table.

Truth Table for this Set-Reset Function

State	S	R	Q	Q	Description
Set	1	0	0	1	Set Q » 1
	1	1	0	1	no change
Reset	0	1	1	0	Reset Q » 0
	1	1	1	0	no change
Invalid	0	0	1	1	Invalid Condition

It can be seen that when both inputs S = "1" and R = "1" the outputs Q and Q can be at either logic level "1" or "0", depending upon the state of the inputs S or R BEFORE this input condition existed. Therefore the condition of S = R = "1" does not change the state of the outputs Q and Q.
However, the input state of S = "0" and R = "0" is an undesirable or invalid condition and must be avoided. The condition of S = R = "0" causes both outputs Q and Q to be HIGH together at logic level "1" when we would normally want Q to be the inverse of Q. The result is that the flip-flop looses control of Q and Q, and if the two inputs are now switched "HIGH" again after this condition to logic "1",

JK Flip Flop



Both the S and the R inputs of the previous SR bistable have now been replaced by two inputs called the J and K inputs, respectively after its inventor Jack Kilby. Then this equates to: J = S and K = R.

The two 2-input AND gates of the gated SR bistable have now been replaced by two 3-input NAND gates with the third input of each gate connected to the outputs at Q and Q. This cross coupling of the SR flip-flop allows the previously invalid condition of S = "1" and R = "1" state to be used to produce a "toggle action" as the two inputs are now interlocked.

If the circuit is now "SET" the J input is inhibited by the "0" status of Q through the lower NAND gate. If the circuit is "RESET" the K input is inhibited by the "0" status of Qthrough the upper NAND gate. As Q and Q are always different we can use them to control the input. When both inputs J and K are equal to logic "1", the JK flip flop toggles as shown in the following truth table.

Truth Table for the JK Function

	Input		Output		Description
	S	R	Q	Q	
Same as for the SR Latch	0	0	0	0	Memory no change
	0	0	0	1	
	0	1	1	0	Reset Q » 0
	0	1	0	1	
	1	0	0	1	Set Q » 1
	1	1	0	1	
Toggle action	1	1	0	1	Toggle
	1	1	1	0	

Then the JK flip-flop is basically an SR flip flop with feedback which enables only one of its two input terminals, either SET or RESET to be active at any one time thereby eliminating the invalid condition seen previously in the SR flip flop circuit.

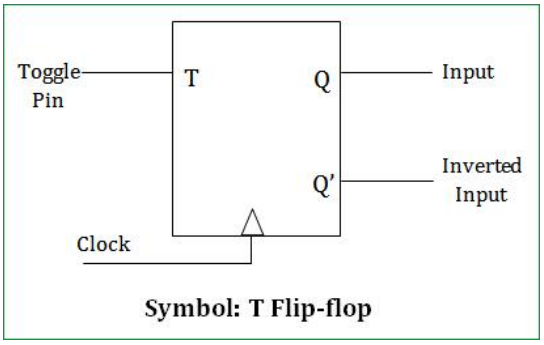
Also when both the J and the K inputs are at logic level "1" at the same time, and the clock input is pulsed "HIGH", the circuit will "toggle" from its SET state to a RESET state, or visa-versa. This results in the JK flip flop acting more like a T-type toggle flip-flop when both terminals are "HIGH".

Although this circuit is an improvement on the clocked SR flip-flop it still suffers from timing problems called "race" if the output Q changes state before the timing pulse of the clock input has time to go "OFF". To avoid this the timing pulse period (T) must be kept as short as possible (high frequency).

T Flip flop

The name T flip-flop is termed from the nature of toggling operation. The major applications of T flip-flop are counters and control circuits. T flip flop is modified form of JK flip-flop making it to operate in toggling region.

Whenever the clock signal is LOW, the input is never going to affect the output state. The clock has to be high for the inputs to get active. Thus, T flip-flop is a controlled Bi-stable latch where the clock signal is the control signal. Thus, the output has two stable states based on the inputs



Symbol: T Flip-flop

Truth Table of T Flip Flop:

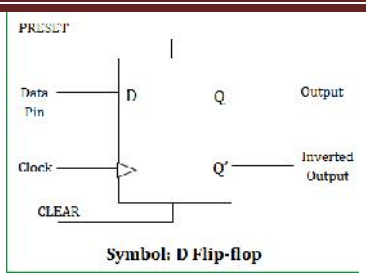
Clock	INPUT		OUTPUT	
	RESET	T	Q	Q'
X	LOW	X	0	1
HIGH	HIGH	0	No Change	
HIGH	HIGH	1	Toggle	
LOW	HIGH	X	No Change	

The T-Flip flop is the modified form of JK flip flop. The Q and Q' represents the output states of the flip flop. According to the table, based on the input the output changes its state. But the important thing to consider is all these can occur only in the presence of the clock signal. This works unlike SR flip flop & JK flip flop for the complimentary inputs. This only has the toggling function.

D Flip-flop

D Flip-flops are used as a part of memory storage elements and data processors as well. D flip-flop can be built using NAND gate or with NOR gate. Due to its versatility they are available as IC packages. The major applications of D flip-flop are to introduce delay in timing circuit, as a buffer, sampling data at specific intervals. D flip-flop is simpler in terms of wiring connection compared to JK flip-flop. Here we are using **NAND gates** for demonstrating the D flip flop

Whenever the clock signal is LOW, the input is never going to affect the output state**.** The clock has to be high for the inputs to get active. Thus, D flip-flop is a controlled Bi-stable latch where the clock signal is the control signal. Again, this gets divided **into** positive edge triggered D flip flop and negative edge triggered D flip-flop**.**



Symbol: D Flip-flop

Clock	INPUT	OUTPUT	
	D	Q	Q'
LOW	x	0	1
HIGH	0	0	1
HIGH	1	1	0

The D(Data) is the input state for the D flip-flop. The Q and Q' represents the output states of the flip-flop. According to the table, based on the inputs the output changes its state. But, the important thing to consider is all these can occur only in the presence of the clock signal. This, works exactly like SR flip-flop for the complimentary inputs alone.

REGISTERS

When a number of flip flops are connected in series, this arrangement is called a Register. The stored information can be transferred within the registers; these are called as 'Shift Registers'. A shift register is a sequential circuit which stores the data and shifts it towards the output on every clock cycle.

Basically shift registers are of 4 types. They are

Serial In Serial Out shift register

Serial In parallel Out shift register

Parallel In Serial Out shift register

Parallel In parallel Out shift register

Serial in Serial Out Shift Register

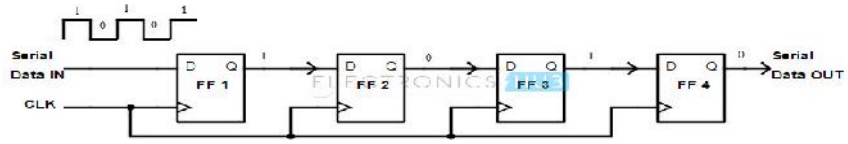
The input to this register is given in serial fashion i.e. one bit after the other through a single data line and the output is also collected serially. The data can be shifted only left or shifted only right. Hence it is called Serial in Serial out shift register or a SISO shift register.

As the data is fed from right as bit by bit, the shift register shifts the data bits to left. A 4-bit SISO shift register consists of 4 flip flops and only three connections.

The registers which will shift the bits to left are called "Shift left registers".

The registers which will shift the bits to right are called "Shift right registers".

As the clock signal is connected to all the 4 flip flops, the serial data is connected to the left most or right most flip flop. The output of the first flip flop is connected to the input of the next flip flop and so on.

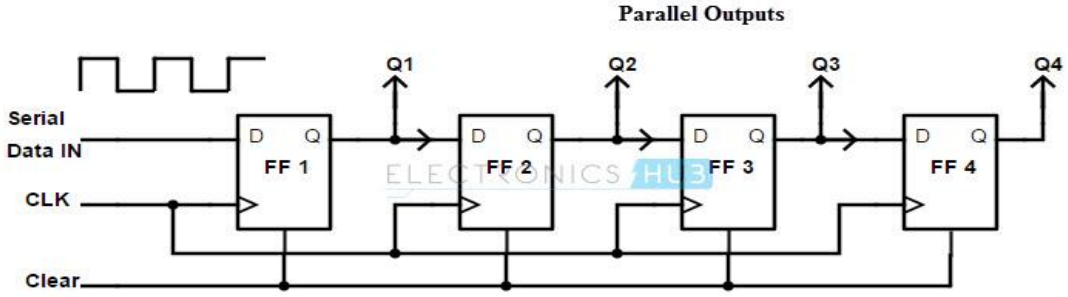


In the above diagram, we see the shift right register; feeding the serial data input from the left side of the flip flop arrangement.

In this shift register, when the clock signal is applied and the serial data is given; only one bit will be available at output at a time in the order of the input data. The use of SISO shift register is to act as temporary data storage device. But the main use of a SISO is to act as a delay element.

Serial in Parallel Out shift register

The input to this register is given in serial and the output is collected in parallel.The clear (CLR) signal is connected in addition to clock signal to all the 4 flip flops in order to RESET them and the serial data is connected to the flip flop at either end (depending on shift left register or shift right register). The output of the first flip flop is connected to the input of the next flip flop and so on. All the flip flops are connected with a common clock.

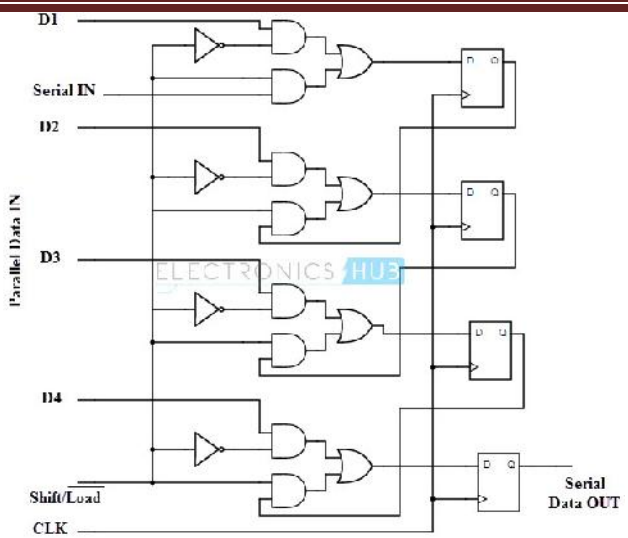


Unlike the serial in serial out shift registers, the output of Serial in Parallel out (SIPO) shift register is collected at each flip flop. Q_1, Q_2, Q_3 and Q_4 are the outputs of first, second, third and fourth flip flops, respectively.

Parallel in Serial out shift register

The input to this register is given in parallel i.e. data is given separately to each flip flop and the output is collected in serial at the output of the end flip flop.

The clock input is directly connected to all the flip flops but the input data is connected individually to each flip flop through a mux (multiplexer) at input of every flip flop. Here D1, D2, D3 and D4 are the individual parallel inputs to the shift register. In this register the output is collected in serial.

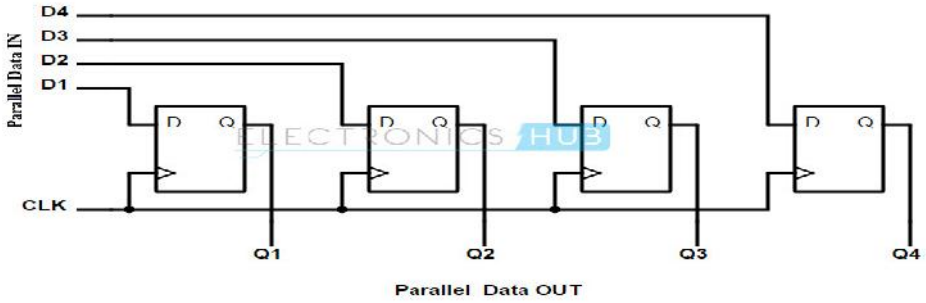


The output of the previous flip flop and parallel data input are connected to the input of the MUX and the output of MUX is connected to the next flip flop. A Parallel in Serial out (PISO) shift register converts parallel data to serial data. Hence they are used in communication lines where a number of data lines are multiplexed into single serial data line.

Parallel in Parallel out shift register

In this register, the input is given in parallel and the output also collected in parallel. The clear (CLR) signal and clock signals are connected to all the 4 flip flops. Data is given as input separately for each flip flop and in the same way, output also collected individually from each flip flop.

In the 4 stage parallel in parallel out register. Qa, Qb, Qc and Qd are the parallel outputs and Pa, Pb, Pc and Pd are the individual parallel inputs. There are no interconnections between any of the four flip flops.

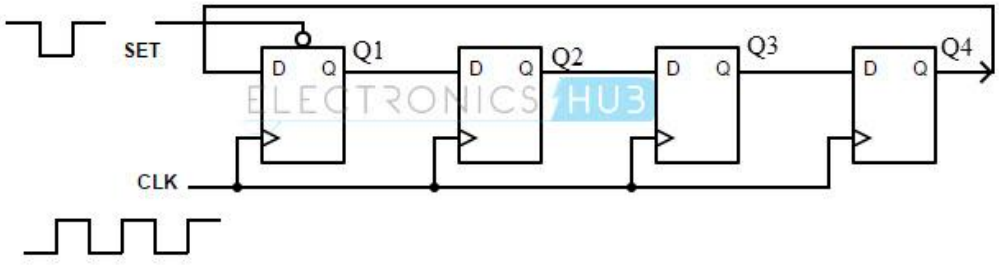


A Parallel in Parallel out (PIPO) shift register is used as a temporary storage device and also as a delay element similar to a SISO shift register.

COUNTERS

Ring Counter

It is designed by connecting the output of the first flip flop to its next one and the output of last flip flop is connected again to the first one as input, like a feedback path. So this is called “Ring Counter”.

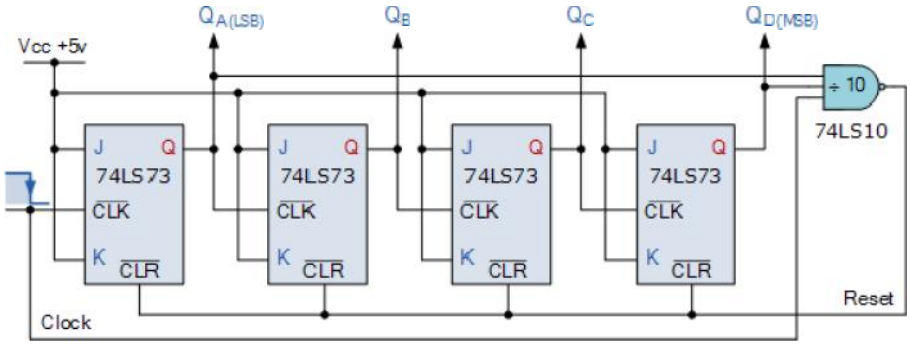


Clock Pulse	Q1	Q2	Q3	Q4
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1

The first flip flop is connected to high input i.e. its input is preset with logic 1 and the output of the first flip flop is connected to input of second flip flop and so on.

Finally, the output of last flip flop is fed back as input to first flip flop. When we apply the first clock pulse to the arrangement; the second stage input changes to 1 and rest inputs are 0. In this way, the input 1 is rotated around the ring.

Asynchronous Decade Counter



This type of asynchronous counter counts upwards on each trailing edge of the input clock signal starting from 0000 until it reaches an output 1001 (decimal 9). Both outputs QA and QD are now equal to logic “1”. On the application of the next clock pulse, the output from the 74LS10 NAND gate changes state from logic “1” to a logic “0” level.

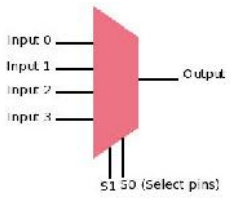
As the output of the NAND gate is connected to the CLEAR (CLR) inputs of all the 74LS73 J-K Flip-flops, this signal causes all of the Q outputs to be reset back to binary 0000 on the count of 10. As outputs QA and QD are now both equal to logic “0” as the flip-flop’s have just been reset, the output of the NAND gate returns back to a logic level “1” and the counter restarts again from 0000. We now have a decade or Modulo-10 up-counter.

Decade Counter Truth Table

Clock Count	Output bit Pattern				Decimal Value
	QD	QC	QB	QA	
1	0	0	0	0	0
2	0	0	0	1	1
3	0	0	1	0	2
4	0	0	1	1	3
5	0	1	0	0	4
6	0	1	0	1	5
7	0	1	1	0	6
8	0	1	1	1	7
9	1	0	0	0	8
10	1	0	0	1	9
11	Counter Resets its Outputs back to Zero				

Multiplexer

Multiplexer is a device that has multiple inputs and a single line output. The select lines determine which input is connected to the output, and also to increase the amount of data that can be sent over a network within certain time. It is also called a data selector.



Multiplexer

The single pole multi-position switch is a simple example of non-electronic circuit of multiplexer, and it is widely used in many electronic circuits. The multiplexer is used to perform high-speed switching and is constructed by electronic components.

Multiplexers are capable of handling both analog and digital applications. In analog applications, multiplexers are made up of of relays and transistor switches, whereas in digital applications, the multiplexers are built from standard logic gates. When the multiplexer is used for digital applications, it is called a digital multiplexer.

Multiplexer Types

Multiplexers are classified into four types:

2-1 multiplexer (1select line)

4-1 multiplexer (2 select lines)

8-1 multiplexer(3 select lines)

16-1 multiplexer (4 select lines)

Multiplexer is one of the basic building units of a computer system which in principle allows sharing of a common line by more than one input lines. It connects multiple input lines to a single output line. At a specific time one of the input lines is selected and the selected input is passed on to the output line.

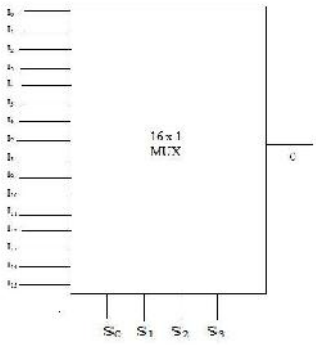
Relation between multiple Input lines and Selection lines

Input lines $16 = 2^4$ i.e. 4 Selection lines

Input lines will be I_0 - I_{15}

Selection lines will be S_0 - S_3

Block Diagram:

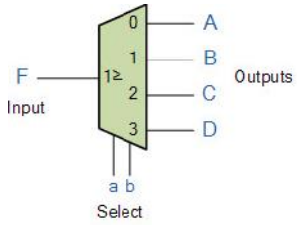


Constructed Diagram:

The diagram will be same as of the block diagram of 16-to-1 line multiplexer in which 8-to-1 line multiplexer Selection lines will be S_0 - $S_2$and S_3will be connected to 2-to-1 line multiplexer Selection and First 8-to-1 line multiplexer Input lines will be I_0 - $I_7$and Second8-to-1 line multiplexer Input lines will be I_8 - I_{15}

Demultiplexer (16:1 and 1:16 description and truth table verification)

De-multiplexer is also a device with one input and multiple output lines. It is used to send a signal to one of the many devices. The main difference between a multiplexer and a de-multiplexer is that a multiplexer takes two or more signals and encodes them on a wire, whereas a de-multiplexer does reverse to what the multiplexer does.



Types of De multiplexer

De-multiplexers are classified into four types

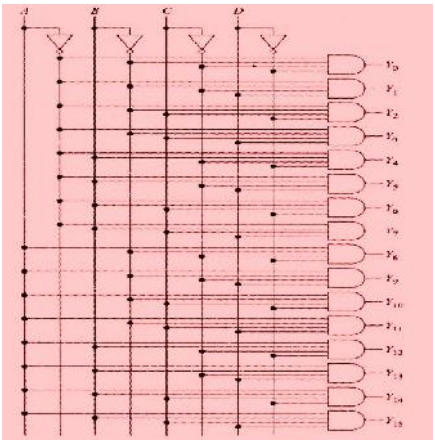
1-2 demultiplexer (1 select line)

1-4 demultiplexer (2 select lines)

1-8 demultiplexer (3 select lines)

1-16 demultiplexer (4 select lines)

The only input are the control input ABCD. In this logic circuit only 1 of the 16 output lines is high and therefore it is called 1 of 16 demultiplexer.



1 of 16 demultiplexer

Let control input ABCD=0001, so only Y_1 AND has all input high, therefore only Y_1 is high. The subscript of Y_1 is 1, means means when you conver ABCD=0001 binary nibble into decimal answer will be 1.Similarly when ABCD=0100, only Y4 AND gate has all high input , as a result Y4 goes high. The subscript of Y_4 is 4, so when you convert ABCD=0100 into decimal, it will be 4.

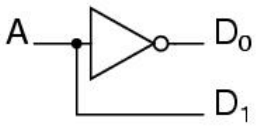
Decoders

A decoder is a circuit that changes a code into a set of signals. It is called a decoder because it does the reverse of encoding, but we will begin our study of encoders and decoders with decoders because they are simpler to design.

A common type of decoder is the line decoder which takes an n-digit binary number and decodes it into 2^ndata lines. The simplest is the 1-to-2 line decoder. The truth table is

A	D_1	D_0
0	0	1
1	1	0

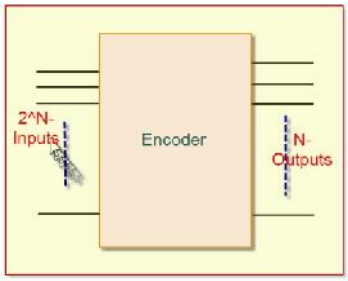
A is the address and D is the data line. D_0 is NOT A and D_1 is A. The circuit looks like



Encoders (Definitions, Seven segment decoder, decimal to BCD encoder)

An encoder is a device, that converts information from one format to another. The purpose of encoder is standardization, speed, security, or saving space by shrinking size. Encoders are combinational logic circuits and they are exactly opposite of decoders. They accept one or more inputs and generate a multi bit output code.

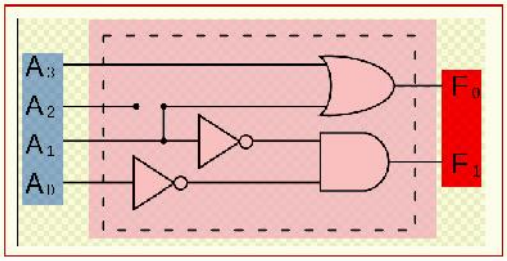
Encoders perform exactly reverse operation than decoder. An encoder has M input and N output lines. Out of M input lines only one is activated at a time and produces equivalent code on output N lines. If a device output code has fewer bits than the input code has, the device is usually called an encoder.



Encoder

Truth Table of The Encoder

The decoders and encoders are designed with logic gate such as an OR-gate. There are different types of encoders and decoders like 4 , 8, and 16 encoders and the truth table of encoder depends upon a particular encoder chosen by the user. Here, a 4-bit encoder is being explained along with the truth table. The four-bit encoder allows only four inputs such as A0, A1, A2, A3 and generates the two outputs F0, F1, as shown in below diagram.



Simple Encoder

I_3	I_2	I_1	I_0		O_1	O_0
0	0	0	1		0	0
0	0	1	0		0	1
0	1	0	0		1	0
1	0	0	0		1	1

Encoder Truth Table

Possible Questions

3 Marks

1) Define flip-flop. What are the types of it? Explain clocked RS flip-flop.

2) What is racing in JK flip-flop? How it is solved in JK master-slave flip-flop?

3) Explain the shift left shift right shift registers.

4) Explain mod 16 counter.

5) With a neat diagram construct and explain the working of D flip-flop.

6) What is called as multiplexer? Draw the circuit for an 8 input multiplexer and explain it.

7) Discuss about 2 to 4 and 3 to 8 decoder.

8) Discuss the working of a multiplexer in detail.

9) With a neat circuit explain BCD to seven segment decoder..

10) What is an encoder? Give the truth table for a octal to binary encoder.

10 Marks

(1) Design a 16-inpt multiplexer sing two 8-inpts multiplexers having active low strobe input.

(2) (i) An 8 bit DAC has an output voltage range of 0-2.55 V. Define its resolution in two ways.

(ii) The digital input for a 4-bit DAC is 0110. Calculate its final output voltage.

(iii) An 8 bit DAC has resolution of 20 mV/LSB. Find V_{OFS} and V_o if the input is $(10000000)_2$.

Questions	opt 1	opt 2	opt 3	opt 4	Answer
UNIT-1					
Two cross coupled NAND gates make	SR Latch	RS flipflop	D flipflop	master slave flipflop	SR Latch
The Basic building block of sequential circuit is called…….	Binary Number	Output	Input	Flip Flop	Flip Flop
D flipflop is a circuit having	2NAND gates	3NAND gates	4NAND gates	5NAND gates	4NAND gates
SR latch is made by two cross coupled	AND gates	OR gates	not change	NOR gates	NOR gates
D flipflop is constructed with	AND	NOT	NAND gates	OR	C. NAND gates
The interval between two pulses is called……..	Period	Frequency	Wavelength	Flip Flpo	Period
Latches constructed with NOR and NAND gates tend to remain in the latched condition due to which configuration feature?	asynchronous operation	low input voltages	cross coupling	gate impedance	asynchronous operation
…….. edge triggered flip flop responds during the rising edge of the clock pulse.	Negative	Positive	Clock Pulse	toggle	Positive
When both inputs of a J-K flip-flop cycle, the output will:	toggle	be invalid	not change	change	not change
Which statement BEST describes the operation of a negative-edge-triggered D flip-flop?	The logic level at the *D* input is transferred to *Q* on NGT of *CLK*.		The *Q* output is ALWAYS identical to the *D* input when *CLK* = PGT.	The *Q* output is ALWAYS identical to the *D* input.	The logic level at the *D* input is transferred to *Q* on NGT of *CLK*.
The truth table for an S-R flip-flop has how many VALID entries?	2	4	3	1	3
For J = 1 and K=1 the output changes state only once the each pulse this condition is called ……..	Toggle	High	Low	Clock Pulse	Toggle
What is one disadvantage of an S-R flip-flop?	It has only a single output.	It has a RACE condition.	It has no clock input.	It has no Enable input.	It has a RACE condition.
What is the significance of the J and K terminals on the J-K flip-flop?	There is no known significance in their designations.	The J represents "jump," which is how the Q output reacts whenever the clock goes HIGH and the J input is also HIGH.	All of the other letters of the alphabet are already in use.	The letters represent the initials of Johnson and King, the co-inventors of the J-K flip-flop	There is no known significance in their designations.
Which of the following is correct for a gated D-type flip-flop?	The output complement follows the input when enabled.	The Q output is either SET or RESET as soon as the D input goes HIGH or LOW.	Only one of the inputs can be HIGH at a time.	The output toggles if one of the inputs is held HIGH.	The Q output is either SET or RESET as soon as the D input goes HIGH or LOW.
A bubble and triangle included at the clock input indicates that the flip flop is a ……	Positive Edge Trigger	Negative Edge Trigger	Clock Pulse	Toggle	Negative Edge Trigger
One example of the use of an S-R flip-flop is as a(n):	transition pulse generator	switch debouncer	astable oscillator	racer	switch debouncer
Which of the following describes the operation of a positive edge-triggered D-type flip-flop?	The output will follow the input on the leading edge of the clock.	The input is toggled into the flip-flop on the leading edge of the clock and is passed to the output on the trailing edge of the clock.	If both inputs are HIGH, the output will toggle.	When both inputs are LOW, an invalid state exists.	The output will follow the input on the leading edge of the clock.

For normal operation both Preset and Clear input must be kept in …….	1	0	Low	Toggle	1
Propagation delay time, t_{PLH}, is measured from the _____.	triggering edge of the clock pulse to the LOW-to-HIGH transition of the output	triggering edge of the clock pulse to the HIGH-to-LOW transition of the output	preset input to the LOW-to-HIGH transition of the output	clear input to the HIGH-to-LOW transition of the output	triggering edge of the clock pulse to the LOW-to-HIGH transition of the output
How is a *J-K* flip-flop made to toggle?	$J = 0, K = 0$	$J = 1, K = 0$	$J = 0, K = 1$	$J = 1, K = 1$	$J = 1, K = 1$
A register is simply a group of …….. that can be used to store binary data.	Flip Flop	Toggle	Counter	Multiplexer	Flip Flop
How many flip-flops are in the 7475 IC?	1	2	4	8	4
A decimal counter has	5 states	10 states	15 states	20 states	10 states
Special type of registers are	latch	flipflop	counters	memory	counters
A register which is used to shift the binary data either to the right or left is called ……	Shift Register	Counter	Flip Flop	Decoder	Shift Register
A group of flip-flops make	clocked sequential circuit	sequential circuit	clocked combinational circuit	combinational circuit	clocked sequential circuit
How many outputs are on a BCD decoder?	4	16	8	10	10
Which digital system translates coded characters into a more useful form?	encoder	display	counters	decoder	decoder
MOD-8 counter require …….. JK Flip flops	2	3	4	8	4
What is the function of an enable input on a multiplexer chip?	to apply V_{cc}	to connect ground	to active the entire chip	to active one half of the chip	to active the entire chip
The expansion inputs to a comparator are used for expansion to a(n):	4-bit system	8-bit system	BCD system	counter system	8-bit system
If the output Q and Q are equal to 1 this state is called ……	Forbidden	Flip Flop	Clock Pulse	Period	Forbidden
A basic multiplexer principle can be demonstrated through the use of a:	single-pole relay	DPDT switch	rotary switch	linear stepper	rotary switch
How many inputs will a decimal-to-BCD encoder have?	4	8	10	16	10
4 to 1 mux would have	2 inputs	3 inputs	4 inputs	5 inputs	4 inputs
Two input mux would have	1 select line	2 select line	4 select line	3 select line	1 select line
A combinational circuit that selects one from many inputs	encoder	decoder	demux	mux	mux
4 to 1 mux would have	1 output	2 output	3 output	4 output	1 output
A principle regarding most IC decoders is that when the correct input is present, the related output will switch:	active-HIGH	to a high impedance	to an open	active-LOW	active-LOW
What control signals may be necessary to operate a 1-line-to-16 line decoder?	flasher circuit control signal	a LOW on all gate enable inputs	input from a hexadecimal counter	a HIGH on all gate enable circuits	a LOW on all gate enable inputs
One multiplexer can take the place of:	several SSI logic gates	combinational logic circuits	several Ex-NOR gates	several Ex-OR gates	several Ex-NOR gates
How many exclusive-NOR gates would be required for an 8-bit comparator circuit?	4	6	8	10	8
How many inputs are required for a 1-of-10 BCD decoder?	4	8	10	1	4
A BCD decoder will have how many rows in its truth table?	10	9	8	3	10
How many possible outputs would a decoder have with a 6-bit binary input?	16	32	64	128	64

Most demultiplexers facilitate which type of conversion?	decimal-to-hexadecimal	single input, multiple outputs	ac to dc	odd parity to even parity	single input, multiple outputs
The inputs/outputs of an analog multiplexer/demultiplexer are:	bidirectional	unidirectional	even parity	binary-coded decimal	bidirectional
Why can a CMOS IC be used as both a multiplexer and a demultiplexer?	It cannot be used as both.	CMOS uses bidirectional switches.	CMOS uses unidirectional switches.	CMOS uses multidirectional switches.	CMOS uses bidirectional switches.
One application of a digital multiplexer is to facilitate:	data generation	serial-to-parallel conversion	parity checking	data selector	data selector
Why is a demultiplexer called a data distributor?	The input will be distributed to one of the outputs.	One of the inputs will be selected for the output.	The output will be distributed to one of the inputs.	All the above	The input will be distributed to one of the outputs.
What is the status of the inputs S_0, S_1, and S_2 of the 74151 eight-line multiplexer in order for the output Y to be a copy of input I_5?	$S_0 = 0, S_1 = 1, S_2 = 0$	$S_0 = 0, S_1 = 0, S_2 = 1$	$S_0 = 1, S_1 = 1, S_2 = 0$	$S_0 = 1, S_1 = 0, S_2 = 1$	$S_0 = 1, S_1 = 0, S_2 = 1$
One way to convert BCD to binary using the hardware approach is:	with MSI IC circuits	with a keyboard encoder	with an ALU	UART	with MSI IC circuits
It is possible for an enable or strobe input to undergo an expansion of two or more mux ICs to the digital multiplexer with the proficiency of large number of _____.	inputs	outputs	selection lines	all of the above	inputs
Which method of combination circuit implementation is widely adopted with maximum output functions and minimum requirement of ICs?	Multiplexer Method	Decoder Method	encoder method	parity generator method	Decoder Method
A binary code that progresses such that only one bit changes between two successive codes is:	nine's-complement code	8421 code	excess-3 code	Gray code	Gray code
For normal operation clear input for each Flip Flop must be …….	0	1	1010	1001	1
How many inputs are required for a 1-of-16 decoder?	2	4	8	16	4
A 4 input multiplexer will be disabled if enabled input is ……	One	Two	Three	Four	Two
A truth table with output columns numbered 0–15 may be for which type of decoder IC?	hexadecimal 1-of-16	dual octal outputs	binary-to-hexadecimal	hexadecimal-to-binary	hexadecimal 1-of-16
……. Is a 16 input multiplexer will active low strobe	High	Low	Toggle	Forbidden	Low
In a BCD-to-seven-segment converter, why must a code converter be utilized?	to convert the 4-bit BCD into 7-bit code	to convert the 4-bit BCD into 10-bit code	to convert the 4-bit BCD into Gray code	No conversion is necessary.	to convert the 4-bit BCD into 7-bit code
How can the active condition (HIGH or LOW) or the decoder output be determined from the logic symbol?	A bubble indicates active-HIGH	A bubble indicates active-LOW	A square indicates active-HIGH	A square indicates active-LOW	A bubble indicates active-LOW
Demultiplexer is …….. of multiplexer.	IC 74150	IC 74154	IC74151	IC 74151	IC 74150
If two inputs are active on a priority encoder, which will be coded on the output?	the higher value	the lower value	neither of the inputs	both of the inputs	the higher value
1 line to 2 line Demultiplexer has …… output.	One	Two	Three	Four	Two
How many 74184 BCD-to-binary converters would be required to convert two complete BCD digits to a binary number?	8	4	2	1	2
How many select lines would be required for an 8-line-to-1-line multiplexer?	2	3	4	8	3

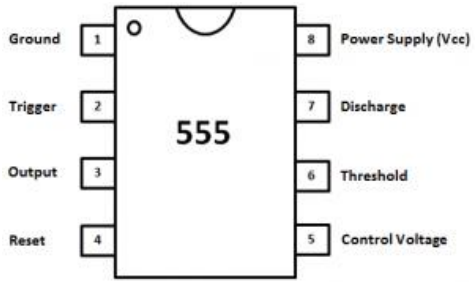
What is the normal operating condition of decoder corresponding to input & output states?	E= 0 & Outputs at '0' logic state	E = 1 & Outputs at '1' logic state	E= 0 & Outputs at '1' logic state	E= 1 & Outputs at '0' logic state	E= 0 & Outputs at '0' logic state
he characteristic equation of D-flipflop implies that _____.	the next state is dependent on previous state	the next state is dependent on present state	the next state is independent of previous state	the next state is independent of present state	the next state is independent of present state
Which circuit is generated from D-flipflop due to addition of an inverter by causing reduction in the number of inputs?	Gated JK- latch	Gated SR- latch	Gated T- latch	Gated D- latch	Gated JK- latch
What is/are the directional mode/s of shifting the binary information in a shift register?	Up-Down	Left - Right	Front - Back	All of the above	Left - Right
Which time interval specify the shifting of overall contents of the shift registers?	Bit time	shift time	word time	code time	word time
A counter is fundamentally a _____ sequential circuit that proceeds through the predetermined sequence of states only when input pulses are applied to it.	register	memory unit	flipflop	arithmetic logic unit	register

Special Function Ics: Timer Ic 555 (Block Diagram, Pin Description)

It is a monolithic timing circuit that gives precise and highly stable delays of time or oscillation. These types of ICs are very cheap and reliable in cost when we compared with the OP-Amp applications in the same areas. These ICs are used as an astable and monostable multivibrators in digital logic probes, DC-DC converters, tachometers, analog frequency meters, voltage regulators, temperature controlled and measurement devices.

The designing of IC 555 timers can be done by using various electrical and electronic components like transistors, resistors, diodes and a flip flop. The operating range of this IC ranges from 4.5V -15V DC supply. The functional parts of the 555 timer IC include flip-flop, voltage divider and a comparator. The main function of this IC is to generate an accurate timing pulse. In the monostable mode, the delay of this IC is controlled by the external components like a resistor and capacitor. In the astable mode, both the duty cycle & frequency are controlled by two external resistors and one capacitor.

The pin configuration of this IC is shown below.



Pin Configuration of 555 Timer IC

GND Pin

Pin-1 is a GND pin which is used to supply a zero voltage to the IC.

Trigger Pin

Pin-2 is a trigger pin which is used to convert the FF from set to RST (reset). The output of the timer depends on the amplitude of the external trigger pulse that is applied to the trigger pin.

Output Pin

Pin-3 is an output pin.

Reset Pin

Pin-4 is a RST pin. When the negative pulse is applied to this pin to disable or reset, and false triggering can be neglected by connecting to VCC.

Control Voltage Pin

Pin-5 is the control voltage pin used to control the pulse width of the output waveform and also the levels of threshold and trigger. When an external voltage is applied to this pin, then the output waveform will be modulated

Threshold Pin

Pin-6 is the threshold pin, when the voltage is applied to threshold pin, then it contrasts with a reference voltage. The set state of the FF can be depends on the amplitude of this pin.

Discharge Pin

Pin-7 is the discharge pin, when the output of the open collector discharges a capacitor between the intervals, then it toggles the output from high to low.

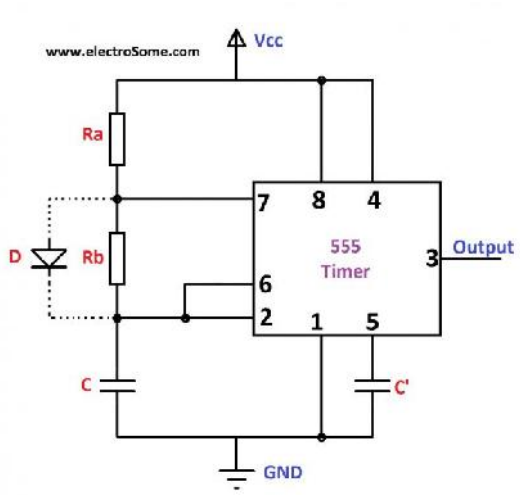
Supply Terminal

Pin-8 is the voltage supply pin which is used to supply the voltage to the IC with respect to the ground terminal.

Application As Astable Multivibrator

Astable Multivibrator using 555 An Astable Multivibrator is an oscillator circuit that continuously produces rectangular wave without the aid of external triggering. So Astable Multivibrator is also known as Free Running Multivibrator. Astable Multivibrator using 555 Timer is very simple, easy to design, very stable and low cost. It can be used for timing from microseconds to hours. Due to these reasons 555 has a large number of applications and it is a popular IC among electronics hobbyists.

Astable Multivibrator using 555 – Circuit



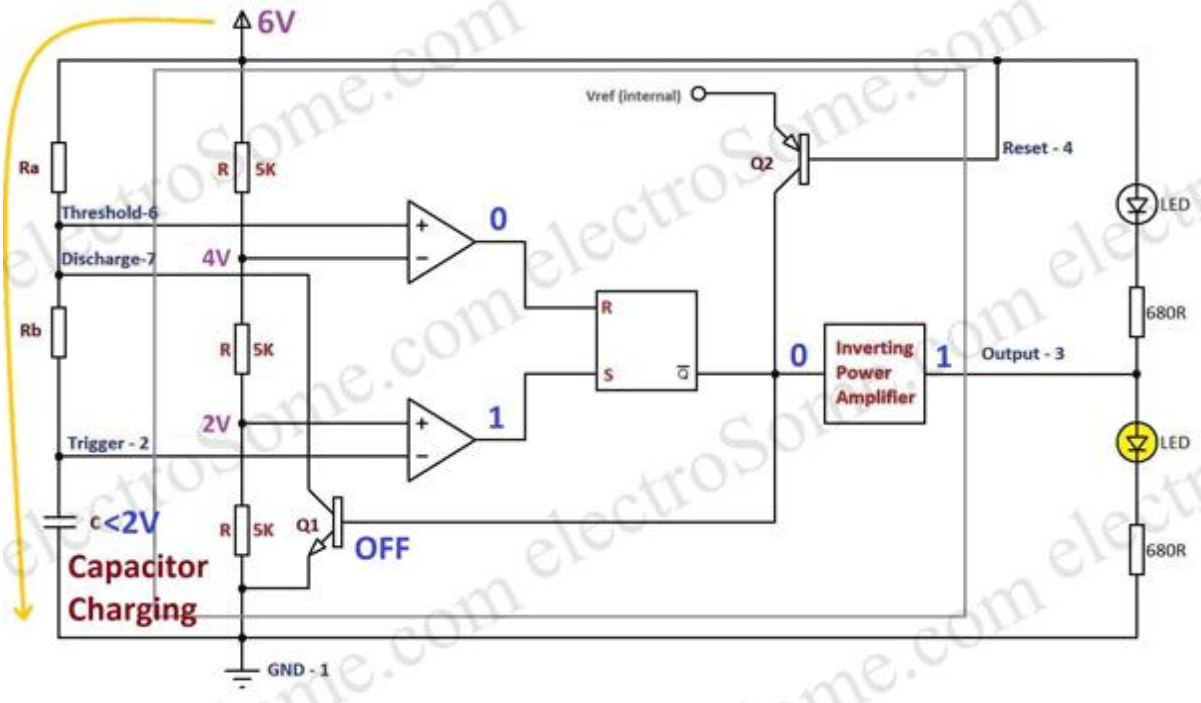
Astable Multivibrator using 555 Timer Circuit Diagram

The circuit diagram of a 555 Timer wired in Astable Mode. 8th pin and 1st pin of the IC are used to give power, Vcc and GND respectively. The 4th pin is RESET pin which is active low and is connected to Vcc to avoid accidental resets. 5th pin is the Control Voltage pin which is not used. So to avoid high frequency noises it is connected to a capacitor C' whose other end is connected to ground. Usually C' = 0.01μF. The Trigger (pin 2) and Threshold (pin 6) inputs are connected to the capacitor which determines the output of the timer. Discharge pin (pin 7) is connected to the resistor Rb such that the capacitor can discharge through Rb. Diode D connected in parallel to Rb is only used when an output of duty cycle less than or equal to 50% is required.

Since the Control Voltage (pin 5) is not used the comparator reference voltages will be 2/3 Vcc and 1/3 Vcc respectively. So the output of the 555 will set (goes high) when the capacitor voltage goes below 1/3 Vcc and output will reset (goes low) when the capacitor voltage goes above 2/3 Vcc.

Working

- When the circuit is switched ON, the capacitor (C) voltage will be less than 1/3 Vcc. So the output of the lower comparator will be HIGH and of the higher comparator will be LOW. This SETs the output of the SR Flip-flop.
- Thus the discharging transistor will be OFF and the capacitor C starts charging from Vcc through resistor Ra & Rb.
- When the capacitor voltage will become greater than 1/3 Vcc (less than 2/3 Vcc), the output of both comparators will be LOW and the output of SR Flip-flop will be same as the previous condition. Thus the capacitor continuous to charge.



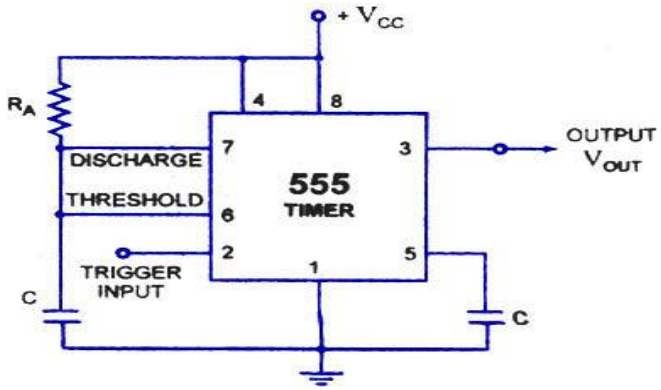
Astable Multivibrator using 555 Timer – Working

- When the capacitor voltage will becomes slightly greater than 2/3 Vcc the output of the higher comparator will be HIGH and of lower comparator will be LOW. This resets the SR Flip-flop.
- Thus the discharging transistor turns ON and the capacitor starts discharging through resistor Rb.
- Soon the capacitor voltage will be less than 2/3 Vcc and output of both comparators will be LOW. So the output of the SR Flip-flop will be the previous state.
- So the discharging of capacitor continuous.
- When the capacitor voltage will become less than 1/3 Vcc, the output SETs since the output of lower comparator is HIGH and of higher comparator is LOW and the capacitor starts charging again.
- This process continuous and a rectangular wave will be obtained at the output.

Monostable Multivibrator

A monostable multivibrator (MMV) is often called a pulse generator circuit in which the duration of the pulse is determined by the R-C network, connected externally to the 555 timer. Here one state of output is stable while the other is unstable to make stable state energy is stored by an externally connected capacitor C.

The time taken in storage determines the pulse width. The schematic of a 555 timer in monostable mode of operation is shown in the figure



Circuit of The Timer 555 as a Monostable Multivibrator

Monostable Multivibrator Circuit details

Pin 1 is grounded. Trigger input is applied to pin 2. The input is kept at + VCC. To obtain transition of output from stable state to unstable state, a negative-going pulse and amplitude of greater than + 2/3 VCC is applied to pin 2. Output is taken from pin 3. Pin 4 is usually connected to + VCC Pin 5 is grounded through a 0.01 u F capacitor to avoid noise problem. Pin 6 (threshold) is shorted to pin 7. A resistor RA is connected between pins 6 and 8. At pins 7 a discharge capacitor is connected while pin 8 is connected to supply VCC.

WORKING

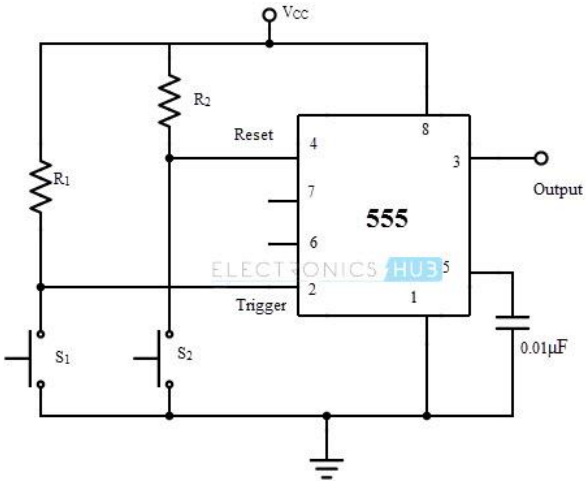
when the output at pin 3 is low i.e. the circuit is in a stable state, the transistor is on and capacitor- C is shorted to ground. When a negative pulse is applied to pin 2, the trigger input falls below +1/3 VCC, the output of comparator goes high which resets the flip-flop and consequently the transistor turns off and the output at pin 3 goes high. This is the transition of the output from stable to quasi-stable state, as shown in figure. As the discharge transistor is cut off, the capacitor C begins charging toward +VCC through resistance RA with a time constant equal to RAC. When the increasing capacitor voltage becomes slightly greater than +2/3 VCC, the output of comparator 1 goes high, which sets the flip-flop. The transistor goes to saturation, thereby discharging the capacitor C and the output of the timer goes low, as illustrated in figure.

Bistable Multivibrator

A Bistable multivibrator is a type of circuit which has two stable states (high and low). It stays in the same state until and unless an external trigger input is applied.

Generally, a bistable multivibrator stays low until a trigger signal is applied and it stays high until a reset signal is applied. Bistable multi vibrators are also called as flip-flops or latches. The term flip-flop is used because it 'flips' to one state and stays there until a trigger is applied and once the trigger is applied it 'flops' back to the original state.

The circuit for a bistable multivibrator using the 555 timer is shown below



A bistable multivibrator is one of the easiest circuits that can be built using a 555 timer. It doesn't require a capacitor as the RC charging unit is not responsible for the generation of the output. The generation of high and low outputs is not dependent on the charging and discharging of the capacitor in the RC unit but rather it is controlled by the external trigger and reset signals.

The explanation of the bi stable mode of operation of the 555 timer is as follows. The trigger and reset pins (pins 2 and 4 respectively) are connected to the supply through two resistors R_1 and R_2 so that they are always high. In all the previous cases, the reset pin is not used and in order to avoid any accidental reset, it is simply connected to VCC.

Two switches are connected between these pins and ground in order to make them go low momentarily. The switch at the trigger input will act as S (SET) input for the internal flip-flop. The switch at the reset input will act as reset for the internal flip-flop.

When the switch S_1 is pressed, the voltage from VCC will bypass the trigger terminal and is shorted to ground through the resistor R_1. Hence, the trigger pulse will momentarily go low and the output of the timer at pin 3 will become HIGH. The output stays HIGH because there is no input from the threshold pin (pin 6 is left open or better if connected to ground) and the output of the internal comparator (comparator 1) will not go high.

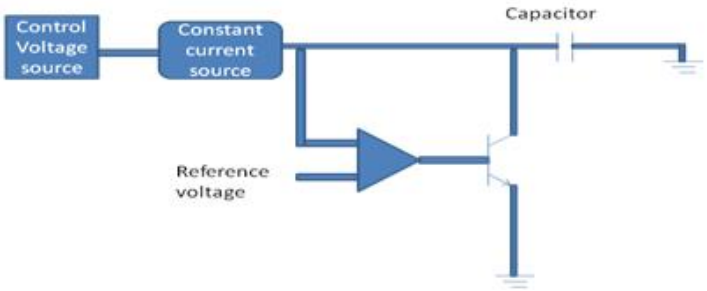
When the switch S_2 is pressed, the voltage from VCC will bypass the reset terminal and is shorted to ground through the resistor R_2. This pin is internally connected to the RESET terminal of the flip-flop. When this signal goes low for a moment, the flip-flop receives the reset signal and RESETs the flip-flop.

Voltage Controlled Oscillator

Vco Ic 566 (Block Diagram And Pin Description)

A Voltage controlled oscillator is an oscillator with an output signal whose output can be varied over a range, which is controlled by the input DC voltage. It is an oscillator whose output frequency is directly related to the voltage at its input. The oscillation frequency varies from few hertz to hundreds of GHz. By varying the input DC voltage, the output frequency of the signal produced can be adjusted

Working



For a Voltage controlled oscillator generating a saw tooth waveform, the main component is the capacitor who's charging and discharging actually decides the formation of the output waveform. The input is given in form a voltage which can be controlled. This voltage is converted to a current signal and is applied to the capacitor. As the current passes through the capacitor, it starts charging and a voltage starts building across it. As the capacitor charges and the voltage across it increases gradually, the voltage is compared with a reference voltage using a comparator.

When the capacitor voltage exceeds the reference voltage, the comparator generates a high logic output which triggers the transistor and the capacitor is connected to ground and starts discharging. Thus the output waveform generated is the representation of the charging and discharging of the capacitor and the frequency is controlled by the input dc voltage.

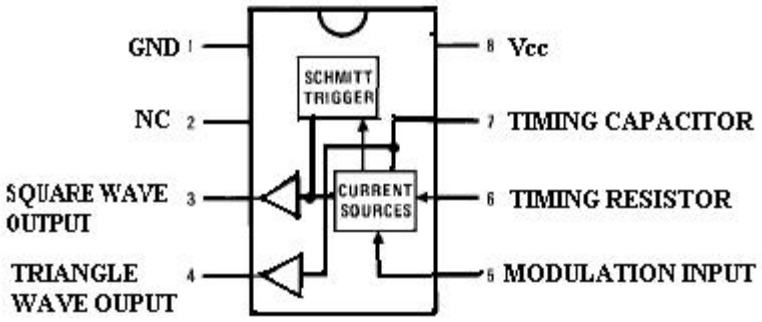
Applications of VCO

- Electronic jamming equipment.
- Function generator.
- Production of electronic music, for production of different types of noise.
- Phase locked loop.
- Frequency synthesizers, used in communication circuits.
-

Voltage Controlled Oscillator (VCO – LM566)

A practical example of a voltage controlled oscillator (VCO) is the LM566. The LM566 is a general purpose VCO which may be used to generate square wave and triangular waveforms as a function input voltage.

The LM566 is specified for operation over 0˚C to 70˚C temperature range. The frequency of which is a linear function of a controlling voltage. The frequency is also controlled by an external resistor and capacitor, whose values control the free running frequency.



Pin Description:

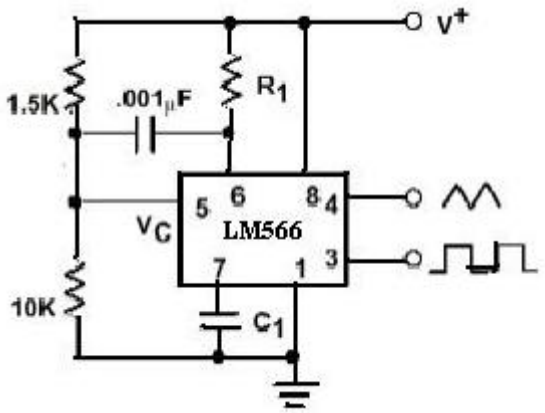
- Pin 1: Ground (GND)
- Pin 2: No connection (NC)
- Pin 3: Square wave output
- Pin 4: Triangular wave output
- Pin 5: Modulation input
- Pin 6: Timing resistor
- Pin 7: Timing capacitor
- Pin 8: Vcc

Applications:

- Function generator
- Tone generator
- FM modulation
- Frequency shift keying
- Clock generator

Working of LM566:

Figure shows that the LM566 IC contains current sources to charge and discharge an external capacitor at a rate set by an external resistor R1 and the modulating dc input voltage V. A 0.001µF capacitor is connected to pin 5 and pin 6. A Schmitt trigger circuit is used to switch the current sources between charging and discharging the capacitor and the triangular voltage produced across the capacitor and square wave from the Schmitt trigger are provided as outputs through buffer amplifiers. Both the output waveforms are buffered so that the output impedance of each is 50 f2. The typical magnitude of the triangular wave and the square wave are 2.4Vpeak to peak and 5.4Vpeak to peak. The free running or center-operating frequency, f0 is



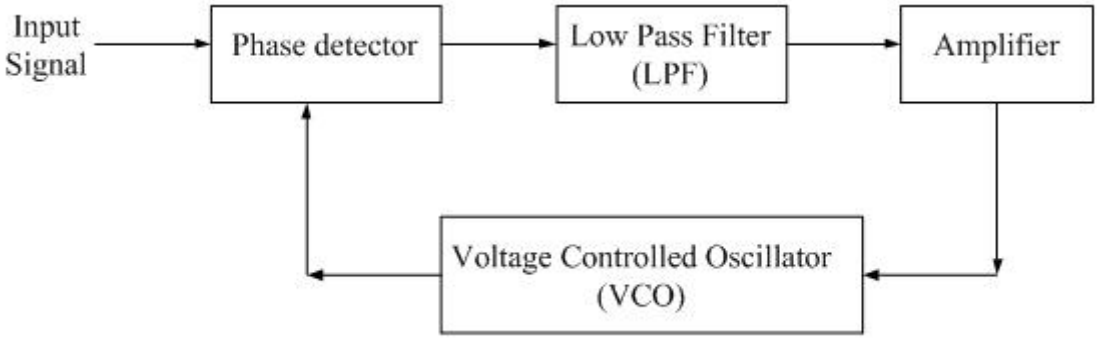
$$f_O = \frac{2\ (V^+ - V_C)}{R_1 C_1 V^+}$$

Phase Locked Loop

PLL Ic 565 (Block Diagram And Pin Description)

The Phase Locked Loop concept was first developed in 1930. Since then it is used in communication systems of different types, particularly in satellite communication system. Before the invention of IC PLL, systems were very complex and costly for use in most consumer & industrial systems. Now PLL ICs are fabricated at a very low cost. Therefore their use has become attractive for many applications such as FM demodulator, Stereo demodulators, Tone detectors, frequency synthesizers etc. Bellow figure shows the block

diagram of PLL which consists of phase detector, LPF, error amplifier and Voltage controlled oscillator.



The phase detector block has two inputs, external signal and oscillator signal. Phase detector detects the phase between two signals and develops an output voltage proportional to phase difference. This block also called as phase detector or phase comparator. Output of phase detector is passed through a LPF. It removes high frequency signal and passes only low frequencies. This filter is also called as loop filter because the PLL system is a closed loop system. Output of filter is amplified by amplifier. This amplified output is applied as input for VCO. This input voltage adjusts the frequency of VCO such that the VCO frequency is equal to signal frequency i.e. VCO converts input voltage into frequency i.e. acts as voltage to frequency converter.

When the signal frequency and VCO frequency is same the loop gets locked. The loop gets locked by detecting the phase difference between two inputs so called Phase Locked Loop (PLL). Without application of any external signal, VCO has some frequency called as free running frequency or centre frequency. In this initial condition loop is not locked i.e. in open condition.

When external signal is applied its frequency is either less or greater than VCO frequency so there is a phase difference between them. Phase detector detects the phase difference between two inputs and generates an error voltage. This is passed through LPF. After amplification it is given as a controlled voltage. This adjusts the frequency of VCO such that input frequency is equal to VCO frequency and forms locked condition.This process of locking the loop is called Capture effect.

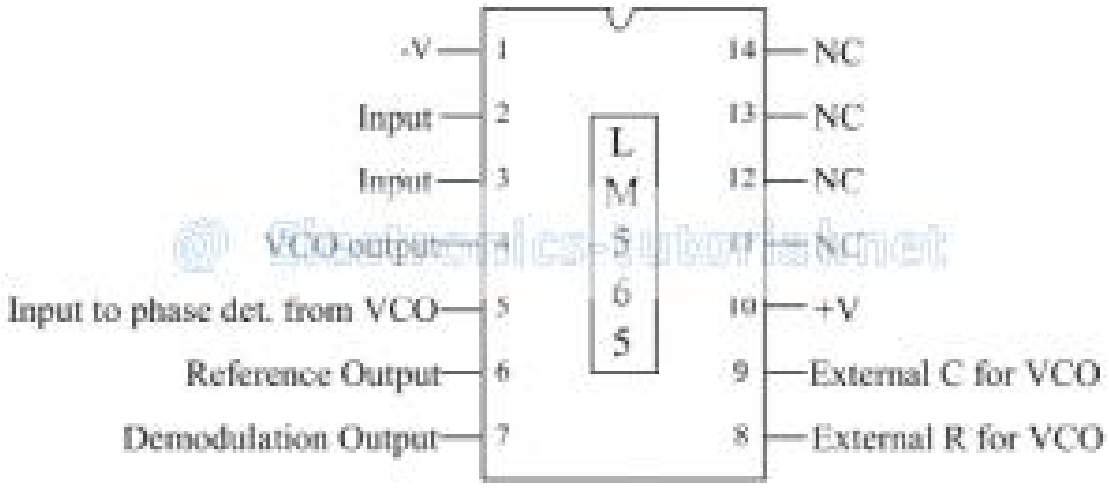
The time required for VCO to adjust its frequency with signal frequency is called capture time. It depends on the internal parameters of system. There is some limit for input signal for which system can acquire a locked condition. This range of frequency between which the system can goes into locked condition is called capture range. This range is symmetrical about centre frequency. This capture range depends upon filter and amplifier characteristics.

If system acquires a locked condition then even if the signal frequency changes the loop remains in locked condition. The range of input frequency over which the locked condition maintained is called locked range. This also depends on amplifier and filter characteristics. Capture range is always less than lock range or almost equal; but capture range is never greater than locked range.

Phase Locked Loop

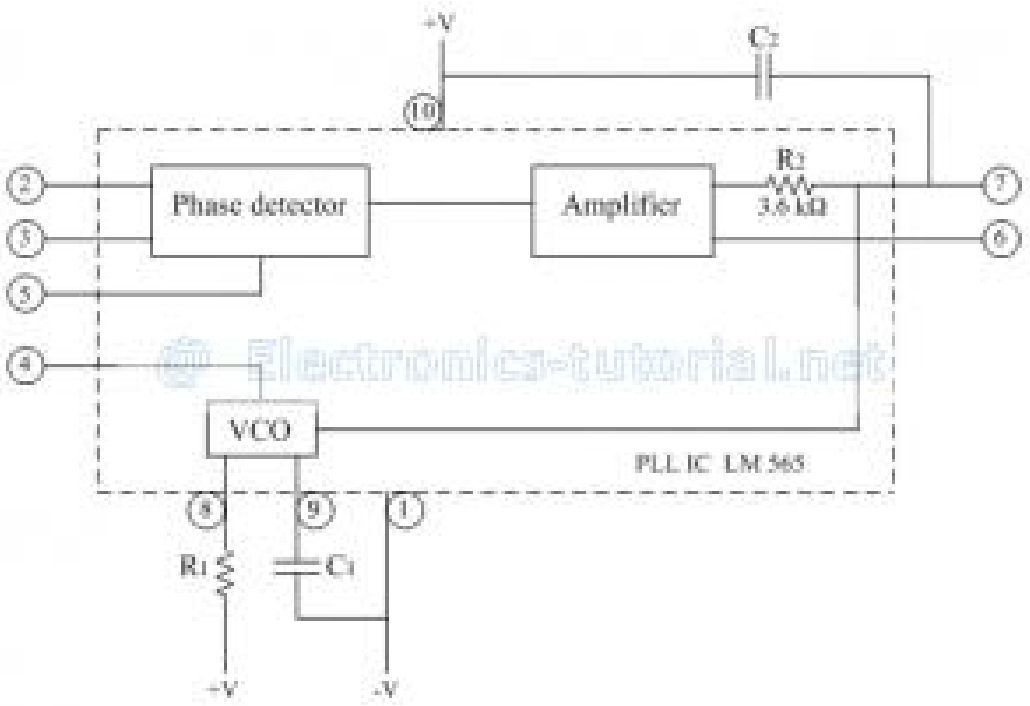
PLL IC 565

The PLL IC 565 is usable over the frequency range 0.1 Hz to 500 kHz. It has highly stable centre frequency and is able to achieve a very linear FM detection. The output of VCO is capable of producing TTL compatible square wave The dual supply is in the range of ±6V to ±12V. The IC can also be operated from single supply in the range 12V to 24V. The following figure shows the pin-out and the internal block schematic of PLL IC LM 565.



It is a 14 pin IC, operated from a dual power supply +V (at pin no. 10) and –V (at pin no. 1).

PIN	Description
Pin no 2 & 3	Signal input for phase detector.
Pin no 4	VCO output is available
Pin no 4 & 5	Shorted externally so that VCO output is applied for phase detection. In some applications PLL loop is broken and some circuit is to be connected between pin no 4 and 5.
Pin no 6	Reference dc voltage is available
Pin no 7	Demodulated output.If input signal between pin no 2 and 3 is FM signal then at pin no 7 we get FM demodulation output.
Pin no 8 and 9	External R1 and C1 for VCO (determines free running frequency of VCO) Internal resistance R2 and external capacitor C2 forms a LPF. The value of internal resistance R2 is 3.6kΩ.



Features of IC 565:

1) Extreme stability of centre frequency typically 200ppm.
2) Wide range of operating voltage ±6V to ±12V.
3) Centre frequency of VCO is programmable by means of resistor, capacitor or voltage.
4) TTL compatible square wave output.
5) Highly linear triangular wave output available at pin no.9
6) Loop can be broken between pin no.4 and 5 and external circuit can be added.
7) Frequency adjustable over the range 1:10 with single capacitor.

Design Equations:

1. Centre Frequency (Free running freq./ output freq./oscillator freq.)

$$fo=0.3/(R1\ C1\ )$$

2. Lock range

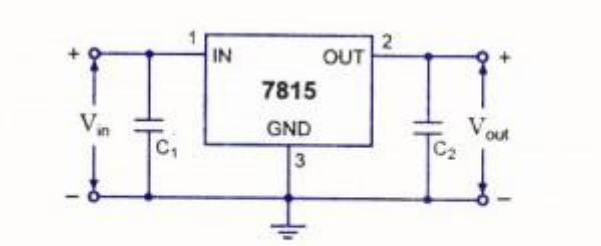
$$fL=(8fo)/V$$

where V=|+V|+|-V|……..(addition of two power supplies)

3. Capture range

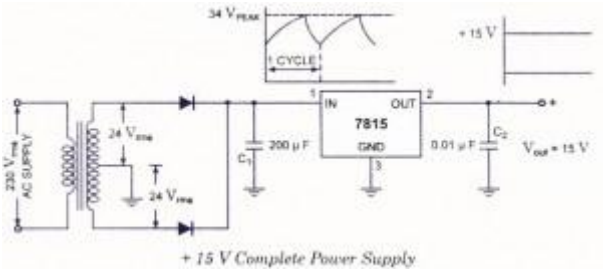
$$fc=\pm 1/2\pi\ \sqrt{((2\pi f\_L)/(R\_2\ C\_2\ ))}$$

Fixed Voltage Regulator Ics 7800 And 7900 Series
Voltage Regulators using 78XX series IC



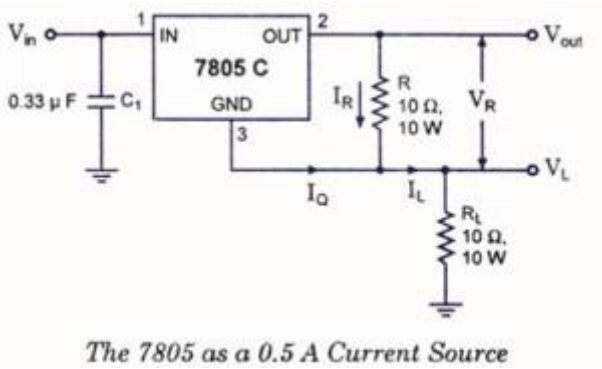
Connection of 7815 Voltage Regulator

7815 voltage regulator (15 volts Power Supply)



+ 15 V Complete Power Supply

The series 7800 regulators provide eight voltage options, ranging from 5 to 24 V. These ICs are designed as fixed voltage regulators and with adequate heat sinking can deliver output currents in excess of 1 A. Although these devices do not require any external component, such components can be employed for providing adjustable voltages and currents. These ICs also have internal thermal overload protection and internal short-circuit current limiting. Figure illustrates how one such IC, a 7815, is connected to provide voltage regulation with output of + 15 V dc from this unit. An unregulated, input voltage V_{in} is filtered by capacitor C, and connected to the pin .1 (IN terminal) of IC. The pin 2 (OUT terminal) of the IC provides a regulated + 15 V which is filtered by capacitor C_2 (mostly for any high frequency noise). The third pin (GND terminal) of the IC is connected to ground. While the input voltage may vary over some permissible voltage range, and the output load may vary over some acceptable range, the output voltage remains constant within specified voltage variation limits. These limitations are mentioned in the manufacturer's specification sheet. In addition, the difference between input and output voltages (V_{in}– V_{out}), callec the dropout voltage, must be typically 20 V, even during the low point on the input ripple voltage. Furthermore, the capacitor C_1, is required if the regulator is located an appreciable distance from a power supply filter. Even though C_2 is not required, it may be used to improve the transient response of the regulator.



The 7805 as a 0.5 A Current Source

IC 7805 as current source

The connection of a 7815 in a complete supply is shown in figure. The ac line voltage is stepped down to 24 V_{rms} across each half of the centre-tapped transformer. A full-wave rectifier and capacitor filter then provides an unregulated dc voltage with ac ripple of a few volts as input to the voltage regulator. The 7815 IC then provides an output of + 15 V dc.

The 7800 regulators can also be employed as current sources. A typical connection diagram of 7805 IC as a 0.5 A current source is depicted in figure.

The current supplied to the load is given as

$$\mathbf{IL = V_R / R + I_Q}$$

when Iq is quiescent current in amperes (4.3 m A typically for the 7805 IC)

In figure, $\mathbf{V_R = V_{23} = 5\ V}$ **and R = 10 ohms**

$$\mathbf{So\ I_L = 5/10 = 0.5A}$$

The output voltage with respect to ground is

$$\mathbf{V_{out} = V_R + V_L}$$

The load resistance, R_L = 10 Ohms, therefore V_L = 5 V

Thus $V_{out} = V_R + V_L = 5 + 5 = 10$ V Minimum input voltage required,

$V_{in} = V_{out}$ + dropout voltage = 10 + 2=12V

Specifications. The specification sheet of fixed positive voltage regulators of 7800 series is given below. Some considerations of a few of the more important parameters should be considered.

Absolute Maximum Ratings

Input voltage: 40 V
Continuous total power dissipation: 2 W
Operating free-air temperature range : – 65 to 150° C

Output Voltage.

The specification for the 7812 indicates that its nominal output voltage is 12 V but could be as low as 11.5 V or as high as 12.5 V.

Input or Line Regulation.

The input or line regulation is seen to be typically 3 mV, to a maximum of 120 mV.

Output or Load Regulation.

The output or load regulation is seen to be typically 4 mV to a maximum of 100 mV (for output currents from 0.25 to 0.75 A). It means that the output voltage can typically vary only 4 mV from the rated 12 V dc.

Short-circuit Output Current.

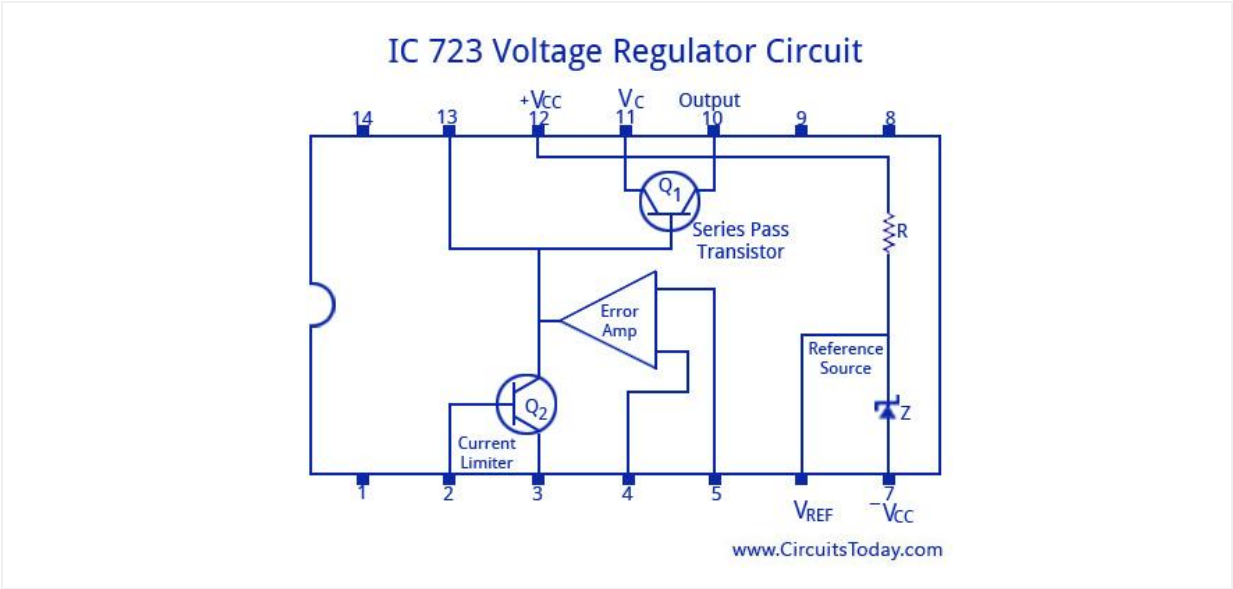
The amount of current is limited to 350 m A if the output were to be short-circuited (may be by accident or by another faulty component). Peak Output Current. The typical peak output current that might be drawn from the supply is 2.2 A against rated maximum current of 1.5 A. It indicates that though the IC is rated as capable of providing 1.5 A, but somewhat more current can be drawn (possibly for a short duration of time).

Drop out Voltage.

The dropout voltage, typically 2 V, is the minimum amount of voltage across the input-output terminals that is required to be maintained if the IC is to operate as a regulator. In case the input voltage falls too low or the output rises so that at least 2 V is not maintained across the input-output terminals of IC, the IC will no longer provide voltage regulation. So input voltage is maintained large enough to ensure that the dropout voltage is provided.

VOLTAGE REGULATOR IC 723 (DESCRIPTION, DESIGNING FOR LOW AND HIGH VOLTAGE)

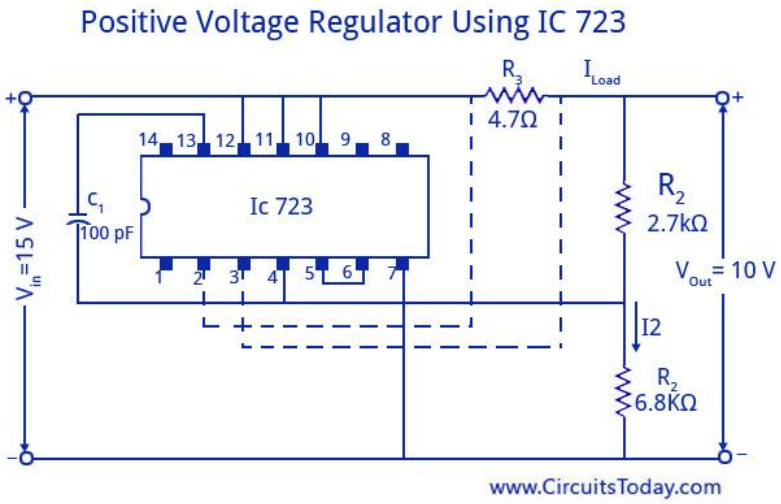
The functional diagram of the voltage regulator is shown below. It consists of a voltage reference source (Pin 6), an error amplifier with its inverting input on pin 4 and non-inverting input on pin 5, a series pass transistor (pins 10 and 11), and a current limiting transistor on pins 2 and 3. The device can be set to work as both posistive and negaive voltage regulators with an output voltage ranging from 2 V to 37 V, and output current levels upto 150 m A. The maximum supply voltage is 40 V, and the line and load regulations are each specified as 0.01%.



IC 723 Voltage Regulator Circuit

The 723 Voltage Regulator IC. The functional diagram of the voltage regulator is shown below. It consists of a voltage reference source (Pin 6), an error amplifier with its inverting input on pin 4 and non-inverting input on pin 5, a series pass transistor (pins 10 and 11), and a current limiting transistor on pins 2 and 3. The device can be set to work as both posistive and negaive voltage regulators with an output voltage ranging from 2 V to 37 V, and output current levels upto 150 m A. The maximum supply voltage is 40 V, and the line and load regulations are each specified as 0.01%.

The figure shown below is a positive voltage regulator with an IC 723. The output voltage can be set to any desired positive voltage between (7-37) volts. 7 volts is the reference starting voltage. All these variations are brought with the change of values in resistors R1 and R2 with the help of a potentiometer. A darlington connection is made by the transistor to Q1 to handle large load current. The broken lines in the image indicate the internal connections for current limiting. Even foldback current limiting is possible in this IC. A regulator output voltage less than the 7 V reference level can be obtained by using a voltage divider across the reference source. The potentially divided reference voltage is then connected to terminal 5.



Positive Voltage Regulator Using IC 723

The IC is provided with voltage at the lowest point on the ripple waveform, should be at least 3 V greater than the output of the regulator and greater than V_{ref}. If it is not so a high-amplitude output ripple is possible to occur.

Possible Questions

3 Marks

1) Explain the pin diagram of IC 555 timer.
2) Explain the operation of IC 566 with the help of block diagram.
3) Explain the block diagram of IC 555 timer.
4) Explain the pin configuration of IC 565.
5) Draw the circuit of a bistable multivibrator using 555 timer and explain its working.
6) Draw the circuit of a astable multivibrator using 555 timer and explain its working.
7) Explain fixed voltage regulator IC 7800
8) Draw the circuit of a monostable multivibrator using 555 timer and explain its working.
9) Discuss about phase locked loop.

10 Marks

1) Draw the circuit diagram of an astable multivibrator to generate the output signal with frequency of 1kHz and the duty cycle of 75%.

2) Design a timer, which should turn ON heater immediately after pressing a push button and should hold heater in ON-state for 5 seconds.

UNIT-II

The 555 timer can be used in which of the following configurations?	astable, monostable	monostable, bistable	bistable, tristable	astable, toggled	astable, monostable
What is another name for a one-shot?	monostable	bistable	astable	tristable	monostable
An Astable Multivibrator or a Free Running Multivibrator is the multivibrator which has no----------	Stable state	Astable State	Toggle State	Forbidden State	Stable state
An astable multivibrator is a circuit that:	is free-running	has two stable states	produces a continuous output signal	is free-running and produces a continuous output signal	produces a continuous output signal
he timing network that sets the output frequency of a 555 astable circuit contains _____.	three external resistors are used	two external resistors and an external capacitor are used	an external resistor and two external capacitors are used	no external resistor or capacitor is required	two external resistors and an external capacitor are used
How to overcome mistriggering on the positive pulse edges in the monostable circuit?	Connect a RC network at the input	Connect an integrator at the input	Connect a differentiator at the input	Connect a diode at the input	Connect a differentiator at the input
A monostable multivibrator has R = 120kΩ and the time delay T = 1000ms, calculate the value of C?	0.9μF	1.32μF	7.5μF	2.49μF	7.5μF
Which among the following can be used to detect the missing heart beat?	Monostable multivibrator	Astable multivibrator	Schmitt trigger	None of the mentioned	Monostable multivibrator
A 555 timer in monostable application mode can be used for	Pulse position modulation	Frequency shift keying	Speed control and measurement	Digital phase detector	Speed control and measurement
When the circuit is switched on, one of the ----------will be more conducting than the other due imbalance in the circuit.	Transistor	Diode	Resistor	Capacitor	Transistor
How can a monostable multivibrator be modified into a linear ramp generator?	Connect a constant current source to trigger input	Connect a constant current source to trigger output	Replace resistor by constant current source	Replace capacitor by constant current source	Replace resistor by constant current source
How does a monostable multivibrator used as frequency divider?	Using square wave generator	Using triangular wave generator	Using sawtooth wave generator	Using sine wave generator	Using square wave generator
What is the function of the comparators in the 555 timer circuit?	to compare the output voltages to the internal voltage divider	to compare the input voltages to the internal voltage divider	to compare the output voltages to the external voltage divider	to compare the input voltages to the external voltage divider	to compare the input voltages to the internal voltage divider
The _____ is defined as the time the output is active divided by the total period of the output signal.	on time	off time	duty cycle	active ratio	duty cycle
What does the discharge transistor do in the 555 timer circuit?	charge the external capacitor to stop the timing	charge the external capacitor to start the timing over again	discharge the external capacitor to stop the timing	discharge the external capacitor to start the timing over again	discharge the external capacitor to start the timing over again
Pulse stretching, time-delay, and pulse generation are all easily accomplished with which type of multivibrator circuit?	astable	monostable	multistable	bistable	monostable
The internal circuitry of the 555 timer consists of _____, an *R-S* flip-flop, a transistor switch, an output buffer amplifier, and a voltage divider.	a comparator	a voltage amplifier	two comparators	a peak detector	two comparators
With most monostable multivibrators, what is the *Q* output when no input trigger has occurred?	low	5 volt	set	high	low
An astable multivibrator requires:	balanced time constants	a pair of matched transistors	no input signal	dual *J-K* flip-flops	no input signal
A Voltage controlled oscillator is ----------with an output signal whose output can be varied over a range, which is controlled by the input DC voltage			an oscillator	Frequency	an oscillator
What is the difference between an astable multivibrator and a monostable multivibrator?	The astable is free running.	The astable needs to be clocked	The monostable is free running	The astable needs to be anti-clocked	The astable is free running.
The series regulators provide eight voltage options, ranging from 5 to 24 V	7800	7801	7814	7885	7800
The output of the astable circuit _____.	constantly switches between two states	is LOW until a trigger is received	is HIGH until a trigger is received	floats until triggered	constantly switches between two states
What controls the output pulse width of a one shot?	the clock frequency	the width of the clock pulse	an *RL* time constant	an *RC* time constant	an *RC* time constant

Frequency Wavelength Period

In a typical IC monostable multivibrator circuit, at the falling edge of the trigger input, the output switches HIGH for a period of time determined by the _____	value of the *RC* timing components	amplitude of the input trigger	frequency of the input trigger	magnitude of the dc supply voltage	value of the *RC* timing components
A monostable 555 timer has the following number of stable states:	0	1	2	3	1
What is the difference between a retriggerable one shot and a nonretriggerable one shot?	The nonretriggerable can only be triggered once	The retriggerable can be triggered many times	The output pulse can be stretched with a nonretriggerable	The output pulse can be stretched with a retriggerable	The output pulse can be stretched with a retriggerable
Triggering a retriggerable one shot during pulse generation will	time out the original pulse	extend the pulse to this trigger width	have no effect	double the original pulse width	extend the pulse to this trigger width
The monostable multivibrator circuit is not an oscillator because _____	its output switches between two states	it requires a trigger to obtain an output signal	it requires a sine wave input signal	the circuit does not require a dc power supply	it requires a trigger to obtain an output signal
A retriggerable one shot has a pulse of 10 ms. 3 ms after being triggered, another trigger pulse is applied. The resulting output pulse will be _____ ms	3	7	10	13	13
What is another name for a bistable multivibrator?	an on-off switch	an oscillator	a flip-flop	none	a flip-flop
Which mode of operation is being used when a 555 timer chip has two external resistors and an external capacitor?	monostable	pulse stretching	Schmitt triggering	astable	astable
For a PLL IC 565 with timing resistor & timing capacitor of about 15 kΩ & 0.02µF respectively, what would be the value of output frequency (f0)?	433.33 Hz	833.33 Hz	1000 Hz	2500 Hz	833.33 Hz
n VCO IC 566, the value of charging & discharging is dependent on the voltage applied at _____.	Triangular wave output	Square wave output	Modulating input	All of the above	Modulating input
According to transfer characteristics of PLL, the phase error between VCO output & incoming signal must be maintained between _____ in order to maintain a lock	0 & Pi	0 & pi/2	0 & 2pi	pi & 2pi	0 & Pi
Which characteristic of PLL is defined as the range of frequencies over which PLL can acquire lock with the input signal?	Free-running state	Pull-in time	Lock-in range	Capture range	Capture range
In PLL, the capture range is always _____ the lock range.	greater than	equal to	less than	zero	less than
Once the phase is locked, the PLL tracks the variation in the input frequency. This indicates that _____	Output frequency changes by same amount as that of input frequency	Output frequency does not change as that of input frequency	There is no relation between input & output frequencies	output remain constant	Output frequency changes by same amount as that of input frequency
In the locked state of PLL, the phase error between the input & output is _____.	maximum	moderate	minimum	zero	minimum
Basically, PLL is used to lock _____	Its output frequency	Phase to the frequency	Phase of the input signal	All of the above	All of the above
Which device is used for diagnostic purposes and for recording?	Low pass filter	Monolithic PLL	Voltage Controlled Oscillator	high pass filter	Voltage Controlled Oscillator
Determine the value of current flow in VCO, when the NE566 VCO external timing resistor RT =250Ω and the modulating input voltage Vc=3.25V.(Assume Vcc=+5v).	3 mA	12 mA	7 mA	10 mA	7 mA
calculate the value of external timing capacitor, if no modulating input signal is applied to VCO. Consider fo=25 kHz and RT=5 kΩ.	6 nF	100 nF	2 nF	10 nF	2 nF
What is the advantage of using filter?	High noise immunity	Reduce the bandwidth of PLL	Provides dynamic range of frequencies	increase the bandwidth of PLL	High noise immunity
Calculate the voltage to frequency conversion factor, where fo=155Hz and Vcc=10V.	130	124	134	116	124
The pulse width out of a one-shot multivibrator increases when the:	supply voltage increases	timing resistor decreases	UTP decreases	timing capacitance increases	supply voltage increases
If the resistor in the Schmitt trigger astable multivibrator is a variable resistor, what part of the output voltage waveform will change when the resistance is changed?	the shape of the waveform	the amplitude of the waveform	the period of the waveform	the period of the wavenumber	the period of the waveform
The 7812 regulator IC provides _____.	5 volt	(- 5 volt)	12 V	(-12 volt)	12 V
The 7805 regulator IC provides _____	5 volt	(- 5 volt)	12 V	(-12 volt)	5 volt

The 7912 regulator IC provides _____	5 volt	(- 5 volt)	12 V	(-12 volt)	(-12 volt)
When the transistor switches to ……..a large voltage will be induced across the inductor	Cut -Off	High Voltage	Current	watt	Cut -Off
Calculate the output frequency in a frequency multiplier if, f_{in} = 200Hz is applied to a 7 divide by N-network	1.2 kHz	1.6 kHz	1.4 kHz	1.9 kHz	1.4 kHz
What must the typical value of n for a frequency multiplication / division?	n<12	n>11	n<10	n=7	n=7
For what kind of input signal, the frequency divider can be avoided frequency multiplier?	Triangular waveform	Square waveform	Saw tooth waveform	Sine waveform	Triangular waveform
Which filter is chosen to remove the carrier component in the frequency shift keying?	3 stage filter	two stage filter	single stage filter	four stage filter	3 stage filter
The voltage range of 566 VCO is	10 to 24 mV	10 to 24 V	10 to 50 V	10 to 50 mV	10 to 24 V
566 VCO provide _output pins	1	2	3	n	2
The PLL was first introduced in early _____	1900s	1830s	1930s	1830s	1930s
IC 565 has _____pin	8	10	16	14	14
Maximum operating voltage of 565 PLL is	10 V	20 V	26 V	32 V	26 V
Bandgap voltage of PLL is	1 mV	1.3 V	1.3mV	1 V	1.3 V

MICROPROCESSOR:

8085 is pronounced as "eighty-eighty-five" microprocessor. It is an 8-bit microprocessor designed by Intel in 1977 using NMOS technology.

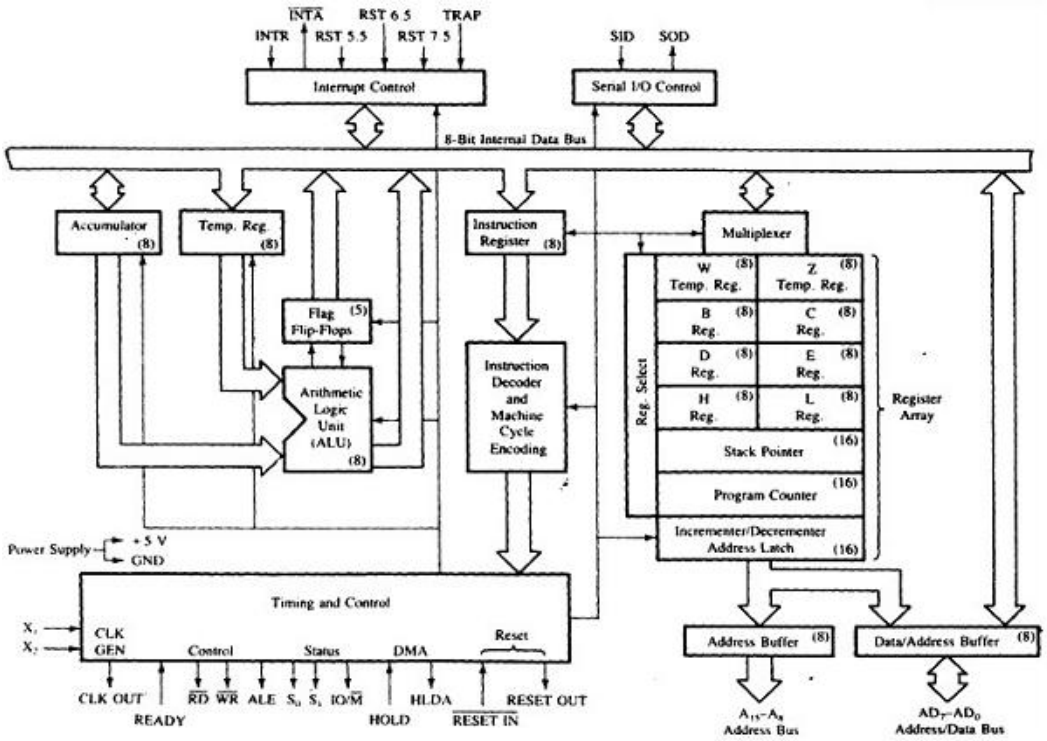
It has the following configuration −

- 8-bit data bus
- 16-bit address bus, which can address upto 64KB
- A 16-bit program counter
- A 16-bit stack pointer
- Six 8-bit registers arranged in pairs: BC, DE, HL
- Requires +5V supply to operate at 3.2 MHZ single phase clock

It is used in washing machines, microwave ovens, mobile phones, etc.

MICROPROCESSOR ARCHITECTURE,

The architecture of 8085 with this following image −



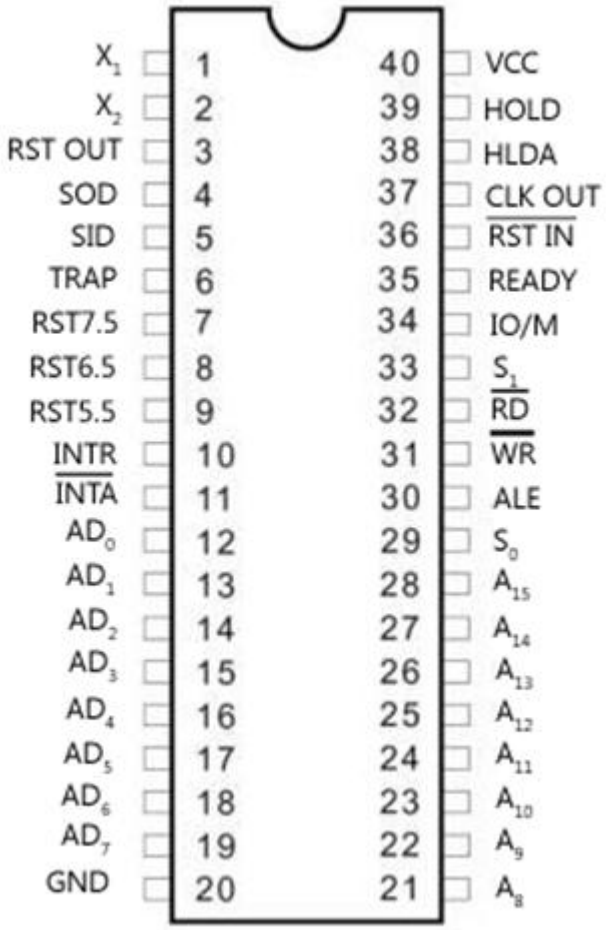


Fig: pin diagram of 8085

8085 up is an 8-bit general purpose microprocessor capable of addressing 64Kb of memory. The device has 40 pins,+5 V power supply, operate on 3 to 5 MHZ frequency single phase clock.

PIN OUT CONFIGURATION OF 8085-

All the signals can be classified in 8085 up pin diagram into six groups –

1) Address Bus: in this 16 signals lines. These lines are splits into two segments -

a) A_{15}-A_8 – unidirectional and used for the higher order address (MSB).

b)AD_7-AD_0 – Dual purpose such as data bus as well as lower address data bus(LSB).

2) Control and status signals: these signals are used to identify the nature of operation.

Three control signals that are-

RD – it is a active low signal. Which indicate that the selected IO or Memory device is to be read and data is available on the data bus.

WR-it is a active low signal which indicate that the data on the data bus are to be written into a selected memory or IO location.

ALE- it is a +ve going pulse generated everytime the 8085 begins an operation (machine cycle): which indicate that the bits on AD_7-AD_0 are address bits.

Three status signals that are –

IO/M- this is a status signal used to differentiate between IO and Memory operations.when it is hign then IO operation and When it is low then Memory operation.

S1 and S0- status signals,similar to IO/M,can identify various operations.that are rarely used in the systems.

3) Power supply:

VCC : +5 V VSS : Ground

4) Clock Frequency:

X1, X2**:** A crystal (RC,LC N/W) is connected at these two pins. this frequency is internally divided by 2.

CLK OUT: clock output this signal can be used as the system clock for the other devices.

5) Externally initiated signals: In this

Five interrupt signals: TRAP, RST 7.5, RST 6.5, RST 5.5 ,INTR.

INTA: interrupt acknowledge

RESET IN: It is a active low signal. When active program counter is set to zero.

RESET OUT: This signal indicates that the MPU is being reset, the signal can be used to reset other devices.

READY: If ready is high during a read or write cycle, it indicates that the memory or peripheral is ready to send or receive data. If ready is low, the CPU will wait for ready to go high before completing the read or write cycle.

HOLD: this signal indicate that another master is requesting the use of the address and data buses.

HLDA: HOLD Acknowledge indicates that the CPU has received the Hold request and that it will relinquish the buses I the next clock cycle. HLDA goes low after the HOLD request is removed. The CPU takes the buses one half clock cycle HLDA goes low.

6) Serial I/O ports:

SOD: serial output data line. The output SOD is set or reset as specified by the SIM instruction.

SID: Serial input data line, the data on this line is loaded into accumulator whenever a RIM instruction is executed.

Three control signals that are-

RD – it is a active low signal. Which indicate that the selected IO or Memory device is to be read and data is available on the data bus.

WR-it is a active low signal which indicate that the data on the data bus are to be written into a selected memory or IO location.

ALE- it is a +ve going pulse generated every time the 8085 begins an operation (machine cycle): which indicate that the bits on AD_7-AD_0 are address bits.

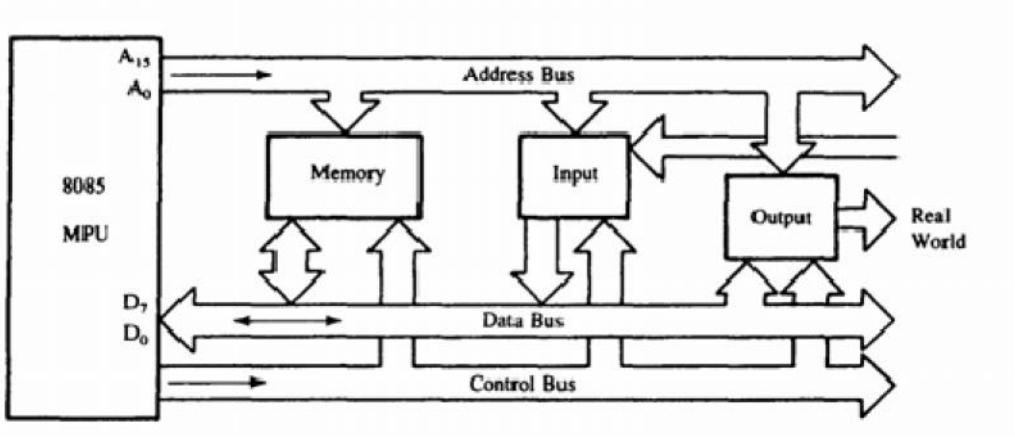
Three status signals that are –

IO/M- this is a status signal used to differentiate between IO and Memory operations. when it is high then IO operation and When it is low then Memory operation.

S1 and S0- status signals ,similar to IO/M, can identify various operations that are rarely used in the systems.

BUS ORGANIZATION AND TIMINGS

There are three buses in Microprocessor:
1.Address Bus
2.Data Bus
3.Control Bus



1.Address Bus:-Genearlly, Microprocessor has 16 bit address bus. The bus over which the CPU sends out the address of the memory location is known as Address bus. The address bus carries the address of memory location to be written or to be read from.
The address bus is unidirectional. It means bits flowing occurs only in one direction, only from microprocessor to peripheral devices.
We can find that how much memory location it can using the formula 2^N. where N is the number of bits used for address lines.

Here, 2^16 = 65536bytes or 64Kb
So we can say that it can access upto 64 kb memory location.

Q.>If a processor has 4 GB memory then how many address lines are required to access this memory?

Ans: 4GB= 4 * 1GB

4 = 2^2

1GB = 2^30

4GB = 2^2 * 2^30 = 2^32

So 32 address lines are required to access the 4 GB memory.

2.Data Bus:-8085 Microprocessor has 8 bit data bus. So it can be used to carry the 8 bit data starting from 00000000H(00H) to 11111111H(FFH). Here 'H' tells the Hexadecimal Number. It is bidirectional. These lines are used for data flowing in both direction means data can be transferred or can be received through these lines. The data bus also connects the I/O ports and CPU. The largest number that can appear on the data bus is 11111111.

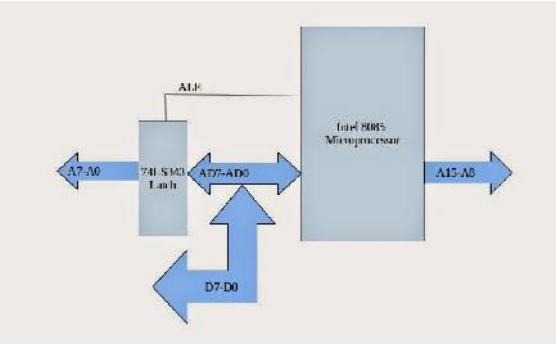
It has 8 parallel lines of data bus. So it can access upto 2^8 = 256 data bus lines.

3.Control Bus:-The control bus is used for sending control signals to the memory and I/O devices. The CPU sends control signal on the control bus to enable the outputs of addressed memory devices or I/O port devices.

Some of the control bus signals are as follows:
1.Memory read
2.Memory write
3.I/O read
4.I/O write.

MULTIPLEXING ADDRESS/DATA BUS AND CONTROL AND STATUS SIGNAL,

In Intel 8085 microprocessor lower order address bus is multiplexed with data bus. This is done to reduce the size of microprocessor. Because we dont require address and data bus at the same time, we can have a common bus for address and data. To select a memory location, we need address first, when the location is selected after that we have to transfer the data with that selected location.



In the first T-state of a machine cycle we need the address bus because we have to address to memory. At this time ALE (Address latch enable) signal becomes active and it enables the latch.

When the latch is enabled, output of the latch becomes equal to the output of AD7-AD0, hence demultiplexing of AD7-AD0 to lower order address bus(A7-A0) is achieved. The higher order address bus is already available as A15-A8.

The data bus contains unspecified data and is in no use because the processor is just addressing at this time, no data transfer could take place.

After first T-state, ALE becomes Low, hence disables the latch. So the output of latch gets disconnected from the microprocessor.

INTERRUPTS: MASKABLE AND NON-MASKABLE INTERRUPT (CONCEPT),

Interrupt is a signal send by an external device to the processor, to the processor to perform a particular task or work. Mainly in the microprocessor based system the interrupts are used for data transfer between the peripheral and the microprocessor.

When a peripheral is ready for data transfer, it interrupts the processor by sending an appropriate signal to the interrupt pin of the processor. If the processor accepts the interrupt then the processor suspends its current activity and executes an interrupt service subroutine to complete the data transfer between the peripheral and processor. After executing the interrupt service routine the processor resumes its current activity. This type of data transfer scheme is called interrupt driven data transfer scheme.

TYPES OF INTERRUPTS

The interrupts are classified into software interrupts and hardware interrupts.

- The software interrupts are program instructions. These instructions are inserted at desired locations in a program. While running a program, lf a software interrupt instruction is encountered, then the processor executes an interrupt service routine (ISR).

- The hardware interrupts are initiated by an external device by placing an appropriate signal at the interrupt pin of the processor. If the interrupt is accepted, then the processor executes an interrupt service routine (ISR).

INTERRUPTS OF 8085

The software interrupts are program instructions. When the instruction is executed, the processor executes an interrupt service routine stored in the vector address of the software interrupt instruction. The software interrupts of 8085 are RST 0, RST 1, RST 2, RST 3, RST 4, RST 5, RST 6 and RST 7.

The vector addresses of software interrupts are given in table below.

Interrupt	Vector address
RST 0	0000_H
RST 1	0008_H
RST 2	0010_H
RST 3	0018_H
RST 4	0020_H
RST 5	0028_H
RST 6	0030_H
RST 7	0038_H

Interrupt	Vector address
RST 7.5	$003C_H$
RST 6.5	0034_H
RST 5.5	$002C_H$
TRAP	0024_H

The software interrupt instructions are included at the appropriate (or required) place in the main program. When the processor encounters the software instruction, it pushes the content of PC (Program Counter) to stack. Then loads the Vector address in PC and starts executing the Interrupt Service Routine (ISR) stored in this vector address. At the end of ISR, a return instruction - RET will be placed. When the RET instruction is executed, the processor POP the content of stack to PC. Hence the processor control returns to the main program after servicing the interrupt. Execution of ISR is referred to as servicing of interrupt.

8085 INTERRUPT.

- Peripheral device activates interrupt by activating the respective pin.
- In response to the interrupt request, microprocessor completes the current instruction execution in main program and transfer program control to interrupt service routine.
- In ISR routine, required task is completed. Task may be to read data, to write data, to update the status, to update the counter etc.
- After completing the task, the program control is transferred back to the main program.These types of interrupts where the microprocessor pins are used to receive interrupt requests are called hardware interrupts.
- The microprocessor 8085 has five hardware interrupts. They are TRAP, RST 7.5, RST 6.5, RST 5.5 and INTR.

TRAP:

- It is a non-mask-able edge and level triggered interrupt.
- It is unaffected by any mask or interrupt enable.
- The TRAP signal must make a LOW to HIGH transition and remain HIGH until acknowledged. This avoids false triggering due to noise or glitches.
- It has the highest priority among all interrupts.
- This interrupt transfers the microprocessor's control to location 0024 H.

RST 7.5:

- It is mask-able, edge triggered interrupt request input line. This interrupt is triggered at the rising edge of the signal.

- It has highest priority among all mask-able interrupts and second priority among all interrupts.
- The interrupt vector location for this interrupt is 003C H.

RST 6.5 and RST 5.5:

- These are level triggered, mask-able interrupt request input lines.
- RST 6.5 transfers microprocessor's control to location 0034 H while RST 5.5 transfers microprocessor's control to location 002C H.

INTR:

- It is level triggered, mask-able interrupt request input line.
- This interrupt works in conjunction with RST N or CALL instruction.

Software interrupts-

1) In case of software interrupts the cause of the interrupt is the execution of the instruction.
2) The microprocessor 8085 has eight instructions. These eight instructions are RST 0 to RST 7. Such interrupts are called as software interrupts.

3) They allow the microprocessor to transfer program control from the main program to the subroutine program i.e. predefined service routine addresses.
4) Predefined service routine is also referred to as ISR.

After completing the subroutine program, the program control returns back to the main program.

The vector locations for RST N instruction are as follows-

Instruction	Address of ISR
RST 0	0000H (8X0)=0000H
RST 1	0008H (8X1)=0008H
RST 2	0010H (8X2)=0010H
RST 3	0018H (8X3)=0018H
RST 4	0020H (8X4)=0020H
RST 5	0028H (8X5)=0028H
RST 6	0030H (8X6)=0030H
RST 7	0038H (8X7)=0038H

Possible Questions

3 Marks

1) Explain the pin out configuration of 8085.

2) What is an interrupt? Explain different 8085 interrupt?

3) Give a note on address and data bus.

4) Discuss about the interrupts of 8085.

5) Discuss the architecture of INTEL 8085 with a neat diagram.

6) What are the functions of pin 7, 8, 9, 10, 11, 35, 37, 38 and 39 of 8085?

7) Explain the architecture of 8085 microprocessor.

8) Explain software and hardware interrupts of 8085.

9) What is the necessity of interrupts? What are its types? Explain it.

10 Marks

1) Explain INTR of 8085 using 8 to 3 encoder.

2) Explain the Microprocessor Architecture of 8085 .

UNIT-III					
Which interrupt has the highest priority	INTR	TRAP	RST 6.5	RST 4.5	RST 6.5
What is the RST for the TRAP?	RST 5.5	RST 4.5	RST 6.5	RST 4	RST 4.5
Which are software interrupts?	RST 0-7	RST 5.5-7.5	RST 6.5	INTR, TRAP	RST 0-7
How many buses are connected as part of the 8085A microprocessor?	2	3	4	8	3
How many bits are used in the data bus?	7	8	9	16	8
The items that you can physically touch in a computer system are called:	software	firmware	hardware	software and hardware	hardware
When was the first 8-bit microprocessor introduced?	1969	1974	1979	1985	1974
I/O mapped systems identify their input/output devices by giving them a(n) _____.	8-bit port number	16-bit port number	8-bit buffer number	8-bit instruction	8-bit port number
The register in the 8085A that is used to keep track of the memory address of the next op-code to be run in the program is the:	stack pointer	program counter	instruction pointer	accumulator	program counter
The microprocessor 8085 has _____ basic instructions and _____ opcodes	80,246	70,346	80,346	70,246	80,246
What does the microprocessor speed depends on ____	clock	data bus width	address bus width	sixe of register	address bus width
The number of software interrupts in 8085 is _____	5	8	9	10	8
Which is a 8 bit microprocessor?	Intel 4040	Pentium	8088	Motorala MC 6801	Motorala MC 6801
In an 8085 based system, the maximum number of input output devices can be connectedusing I/0 mapped I/O method is	64	512	256	65536	512
Intel Itanium processor are designed for __	servers and personal computers	servers only	personal computers only	calculators	servers only
The TRAP is one of the interrupts available its INTEL 8085. Which one of the followingstatements is true of TRAP?	It is level triggered	It is negative edge trigge red	It is positive edge triggered	It is both positive edge triggered and level triggered	It is both positive edge triggered and level triggered
The 8085A is a(n):	16-bit parallel CPU	8-bit serial CPU	8-bit parallel CPU	16-bit serial CPU	8-bit parallel CPU
The status that cannot be operated by direct instructions is	Cy	Z	P	AC	AC
Which bus is a bidirectional bus?	address bus	data bus	address bus and data bus	control bus	data bus
Which of the following buses is primarily used to carry signals that direct other ICs to find out what type of operation is being performed?	address bus	data bus	control bus	address decoder bus	control bus
What kind of computer program is used to convert mnemonic code to machine code?	debug	assembler	C++	fortran	assembler
Which of the following are the three basic sections of a microprocessor unit?	operand, register, and arithmetic/logic unit (ALU)	control and timing, register, and arithmetic/logic unit (ALU)	control and timing, register, and memory	arithmetic/logic unit (ALU), memory, and input/output	control and timing, register, and arithmetic/logic unit (ALU)
Intel 8085 is a _____ bit microprocessor.	4 bit	8 bit	16 bit	32 bit	8 bit

In 8085 are of the following statements is not true	Co processor is interfaced in max mode	Co processor is interfaced in min mode	Co processor is interfaced in max/min mode	Supports pipelinig	Co processor is interfaced in min mode
_____ and _____ are treated as a 16 bit unit for stack operation.	PSW and ACC	CS and P	Z and S	PC and SP	PSW and ACC
The width of address bus and data bus in 8085 are respectively ….	16, 8	8,16	8,8	16,16	16, 8
_____ memory locations can be addressed directly by Intel 8085.	34 K	44K	54 K	64 K	64 K
Identify the non makeable interrupt in the following	RST4.5	RST5.5	RST6.5	RST 7.5	RST4.5
In response to RST 7.5 interrupt, the execution of control transfers to memory location...	0000H	002CH	0034H	003CH	003CH
The second part of the instruction is the data to be operated on, and it is called _____	opcode	operand	instruction cycle	fetch cycle	operand
The first part of an instruction which specifies the task to be performed by the computer is called _____	opcode	operand	instruction cycle	fetch cycle	opcode
Which of the following is a one-byte instruction?	MVI B, 05	LDA 2500H	IN 01	MOV A,B	MOV A,B
Which of the following is a two-byte instruction?	MVI B, 05	LDA 2500H	IN 01	both a and c	both a and c
The necessary steps carried out to perform the operation of accessing either memory or I/O Device, constitute a _____	fetch operation	execute operation	machine cycle	instruction cycle	machine cycle
The status of S0 and S1 pins for memory write is.	0, 0	0,1	1,0	1,1	1,0
The status of S0 and S1 pins for memory fetch is.	0, 0	0, 1	1,0	1,1	1,1
The interrupt vector address for RST 6.5 is	0000H	0034H	0018H	002CH	0034H
The interrupt vector address for RST 5.5 is	0000H	0034H	0018H	002CH	002CH
The difference between memory and storage is that the memory is_____ and storage is_____	Temporary, permanent	Permanent, temporary	Slow, fast	None of the above	Temporary, permanent
Which of the Following holds the ROM, CPU, RAM and expansion cards.	Hard disk	Floppy disk	Mother board	None of the above	Mother board
The language that the computer can understand and execute is called _____	Machine language	Application software	System program	None of the above	Machine language
Actual execution of instructions in a computer takes place in	ALU	Control Unit	Storage unit	None of the above	ALU
Execution of two or more programs by a single CPU is known as:	Multiprocessing	Time sharing	Multiprogramming	None of the above	Multiprogramming
Operating system is_____	A collection of hardware components	A collection of software routines	A collection of input-output devices	none of the above	A collection of software routines
The part of machine level instruction, which tells the central processor what was to be done is	Operation code	Address	Operand	None of the above	Operation code
The communication line between the CPU, memory and peripherals is called a	Bus	line	media	none of these	Bus
The language that the computer can understand and execute is called	Machine language	Application software	System program	None of the above	Machine language
The First Microprocessor was_____	Intel 4004	8080	8085	4008	Intel 4004
The address bus flow in _____	bidirection	unidirection	Mulidirection	Circular	unidirection
Status register is also called as _____	Accumulator	Stack	Counter	flags	flags

The external system bus architecture is created using from _____ architectur	Pascal	Dennis Ritchie	Charles Babbage	Von Neumann	Von Neumann
16 bit address bus can generate _____ addressess	32767	25652	65536	65767	65536
_____ a subsystem that transfer data between computer components inside a computer or between computer	chip	register	processor	bus	Bus
How many bit stored by status register	1 bit	4 bit	6 bit	8 bit	1 bit
What are level triggering interrupts?	INTR	INTR & TRAP	RST 6.5 & RST 5.5	RST 7.5 & RST 5.5	RST 6.5 & RST 5.5
CS stands for	code segment	coot segment	cost segment	counter segment	code segment
DS stands for	data segment	direct segment	declare segment	divide segment	data segment
IP stands for	instructions purpose	instructions paints	instruction program	instruction pointer	instruction pointer
The 16 bit wide accumulator is called	AX	AH	AL	DL	AX
The upper 8 bit register is	BH	BL	AH	CH	AH

PROGRAMMING MODEL OF 8085 :
INSTRUCTION SET-DATA TRANSFER ARITHMETIC, LOGICAL AND BRANCH INSTRUCTION

The 8086 microprocessor supports 8 types of instructions −

- Data Transfer Instructions
- Arithmetic Instructions
- Bit Manipulation Instructions
- String Instructions
- Program Execution Transfer Instructions (Branch & Loop Instructions)
- Processor Control Instructions
- Iteration Control Instructions
- Interrupt Instructions

Let us now discuss these instruction sets in detail.

Data Transfer Instructions

These instructions are used to transfer the data from the source operand to the destination operand. Following are the list of instructions under this group −

Instruction to transfer a word

- **MOV** − Used to copy the byte or word from the provided source to the provided destination.
- **PPUSH** − Used to put a word at the top of the stack.
- **POP** − Used to get a word from the top of the stack to the provided location.
- **PUSHA** − Used to put all the registers into the stack.
- **POPA** − Used to get words from the stack to all registers.
- **XCHG** − Used to exchange the data from two locations.
- **XLAT** − Used to translate a byte in AL using a table in the memory.

Instructions for input and output port transfer

- **IN** − Used to read a byte or word from the provided port to the accumulator.
- **OUT** − Used to send out a byte or word from the accumulator to the provided port.

Instructions to transfer the address

- **LEA** − Used to load the address of operand into the provided register.
- **LDS** − Used to load DS register and other provided register from the memory
- **LES** − Used to load ES register and other provided register from the memory.

Instructions to transfer flag registers

- **LAHF** − Used to load AH with the low byte of the flag register.
- **SAHF** − Used to store AH register to low byte of the flag register.
- **PUSHF** − Used to copy the flag register at the top of the stack.
- **POPF** − Used to copy a word at the top of the stack to the flag register.

Arithmetic Instructions

These instructions are used to perform arithmetic operations like addition, subtraction, multiplication, division, etc.

Following is the list of instructions under this group −

Instructions to perform addition

- **ADD** − Used to add the provided byte to byte/word to word.
- **ADC** − Used to add with carry.
- **INC** − Used to increment the provided byte/word by 1.
- **AAA** − Used to adjust ASCII after addition.
- **DAA** − Used to adjust the decimal after the addition/subtraction operation.

Instructions to perform subtraction

- **SUB** − Used to subtract the byte from byte/word from word.
- **SBB** − Used to perform subtraction with borrow.
- **DEC** − Used to decrement the provided byte/word by 1.
- **NPG** − Used to negate each bit of the provided byte/word and add 1/2's complement.
- **CMP** − Used to compare 2 provided byte/word.
- **AAS** − Used to adjust ASCII codes after subtraction.
- **DAS** − Used to adjust decimal after subtraction.

Instruction to perform multiplication

- **MUL** − Used to multiply unsigned byte by byte/word by word.
- **IMUL** − Used to multiply signed byte by byte/word by word.
- **AAM** − Used to adjust ASCII codes after multiplication.

Instructions to perform division

- **DIV** − Used to divide the unsigned word by byte or unsigned double word by word.
- **IDIV** − Used to divide the signed word by byte or signed double word by word.
- **AAD** − Used to adjust ASCII codes after division.
- **CBW** − Used to fill the upper byte of the word with the copies of sign bit of the lower byte.
- **CWD** − Used to fill the upper word of the double word with the sign bit of the lower word.

Bit Manipulation Instructions

These instructions are used to perform operations where data bits are involved, i.e. operations like logical, shift, etc.

Following is the list of instructions under this group −

Instructions to perform logical operation

- **NOT** − Used to invert each bit of a byte or word.
- **AND** − Used for adding each bit in a byte/word with the corresponding bit in another byte/word.
- **OR** − Used to multiply each bit in a byte/word with the corresponding bit in another byte/word.
- **XOR** − Used to perform Exclusive-OR operation over each bit in a byte/word with the corresponding bit in another byte/word.
- **TEST** − Used to add operands to update flags, without affecting operands.

Instructions to perform shift operations

- **SHL/SAL** − Used to shift bits of a byte/word towards left and put zero(S) in LSBs.
- **SHR** − Used to shift bits of a byte/word towards the right and put zero(S) in MSBs.
- **SAR** − Used to shift bits of a byte/word towards the right and copy the old MSB into the new MSB.

Instructions to perform rotate operations

- **ROL** − Used to rotate bits of byte/word towards the left, i.e. MSB to LSB and to Carry Flag [CF].
- **ROR** − Used to rotate bits of byte/word towards the right, i.e. LSB to MSB and to Carry Flag [CF].
- **RCR** − Used to rotate bits of byte/word towards the right, i.e. LSB to CF and CF to MSB.
- **RCL** − Used to rotate bits of byte/word towards the left, i.e. MSB to CF and CF to LSB.

String Instructions

String is a group of bytes/words and their memory is always allocated in a sequential order.

Following is the list of instructions under this group −

- **REP** − Used to repeat the given instruction till CX ≠ 0.
- **REPE/REPZ** − Used to repeat the given instruction until CX = 0 or zero flag ZF = 1.
- **REPNE/REPNZ** − Used to repeat the given instruction until CX= 0 or zero flag ZF = 1.
- **MOVS/MOVSB/MOVSW** − Used to move the byte/word from one string to another.
- **COMS/COMPSB/COMPSW** − Used to compare two string bytes/words.
- **INS/INSB/INSW** − Used as an input string/byte/word from the I/O port to the provided memory location.
- **OUTS/OUTSB/OUTSW** − Used as an output string/byte/word from the provided memory location to the I/O port.
- **SCAS/SCASB/SCASW** − Used to scan a string and compare its byte with a byte in AL or string word with a word in AX.

- **LODS/LODSB/LODSW** − Used to store the string byte into AL or string word into AX.

Program Execution Transfer Instructions (Branch and Loop Instructions)

These instructions are used to transfer/branch the instructions during an execution. It includes the following instructions −

Instructions to transfer the instruction during an execution without any condition −

- **CALL** − Used to call a procedure and save their return address to the stack.
- **RET** − Used to return from the procedure to the main program.
- **JMP** − Used to jump to the provided address to proceed to the next instruction.

Instructions to transfer the instruction during an execution with some conditions −

- **JA/JNBE** − Used to jump if above/not below/equal instruction satisfies.
- **JAE/JNB** − Used to jump if above/not below instruction satisfies.
- **JBE/JNA** − Used to jump if below/equal/ not above instruction satisfies.
- **JC** − Used to jump if carry flag CF = 1
- **JE/JZ** − Used to jump if equal/zero flag ZF = 1
- **JG/JNLE** − Used to jump if greater/not less than/equal instruction satisfies.
- **JGE/JNL** − Used to jump if greater than/equal/not less than instruction satisfies.
- **JL/JNGE** − Used to jump if less than/not greater than/equal instruction satisfies.
- **JLE/JNG** − Used to jump if less than/equal/if not greater than instruction satisfies.
- **JNC** − Used to jump if no carry flag (CF = 0)
- **JNE/JNZ** − Used to jump if not equal/zero flag ZF = 0
- **JNO** − Used to jump if no overflow flag OF = 0
- **JNP/JPO** − Used to jump if not parity/parity odd PF = 0
- **JNS** − Used to jump if not sign SF = 0
- **JO** − Used to jump if overflow flag OF = 1
- **JP/JPE** − Used to jump if parity/parity even PF = 1
- **JS** − Used to jump if sign flag SF = 1

Processor Control Instructions

These instructions are used to control the processor action by setting/resetting the flag values.

Following are the instructions under this group −

- **STC** − Used to set carry flag CF to 1
- **CLC** − Used to clear/reset carry flag CF to 0
- **CMC** − Used to put complement at the state of carry flag CF.
- **STD** − Used to set the direction flag DF to 1

- **CLD** − Used to clear/reset the direction flag DF to 0
- **STI** − Used to set the interrupt enable flag to 1, i.e., enable INTR input.
- **CLI** − Used to clear the interrupt enable flag to 0, i.e., disable INTR input.

Iteration Control Instructions

These instructions are used to execute the given instructions for number of times. Following is the list of instructions under this group −

- **LOOP** − Used to loop a group of instructions until the condition satisfies, i.e., CX = 0
- **LOOPE/LOOPZ** − Used to loop a group of instructions till it satisfies ZF = 1 & CX = 0
- **LOOPNE/LOOPNZ** −Used to loop a group of instructions till it satisfies ZF=0 &CX=0
- **JCXZ** − Used to jump to the provided address if CX = 0

Interrupt Instructions

These instructions are used to call the interrupt during program execution.

- **INT** − Used to interrupt the program during execution and calling service specified.
- **INTO** − Used to interrupt the program during execution if OF = 1
- **IRET** − Used to return from interrupt service to the main program

ADDRESSING MODES

16 BIT DATA TRANSFER AND MEMORY RELATED INSTRUCTIONS-

Addressing Modes in 8085

These are the instructions used to transfer the data from one register to another register, from the memory to the register, and from the register to the memory without any alteration in the content. Addressing modes in 8085 is classified into 5 groups −

Immediate addressing mode

In this mode, the 8/16-bit data is specified in the instruction itself as one of its operand. **For example:** MVI K, 20F: means 20F is copied into register K.

Register addressing mode

In this mode, the data is copied from one register to another. **For example:**MOV K, B: means data in register B is copied to register K.

Direct addressing mode

In this mode, the data is directly copied from the given address to the register. **For example:** LDB 5000K: means the data at address 5000K is copied to register B.

Indirect addressing mode

In this mode, the data is transferred from one register to another by using the address pointed by the register. **For example:** MOV K, B: means data is transferred from the memory address pointed by the register to the register K.

Implied addressing mode

This mode doesn't require any operand; the data is specified by the opcode itself. **For example:** CMP.

STACK AND SUBROUTINE INSTRUCTIONS.

The STACK

The stack is one of the most important things you must know when programming. Think of the stack as a deck of cards. When we put a card on the deck, it will be the top card. Then we put another card, then another. When we remove the cards, we remove them from backwards, the last card first and so on. The stack works the same way, we put (push) words (addresses or register pairs) on the stack and then remove (pop) them backwards. That's called LIFO, Last In First Out.

The 8085 uses a 16 bit register to know where the stack top is located, and that register is called the SP (Stack Pointer). There are instructions that allow you to modify it's contents but you should NOT change the contents of that register if you don't know what you're doing!
PUSH & POP

Push and pop "pushes" bytes on the stack and then takes them off. When you push something, the stack counter will decrease with 2 (the stack "grows" down, from higher addresses to lower) and then the register pair is loaded onto the stack. When you pop, the register pair is first lifted of the stack,

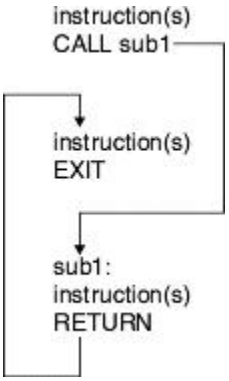
SUBROUTINE INSTRUCTIONS

Subroutine is a sequence of program instructions that perform a specific task, packaged as a unit. This unit can then be used in programs wherever that particular task should be performed.

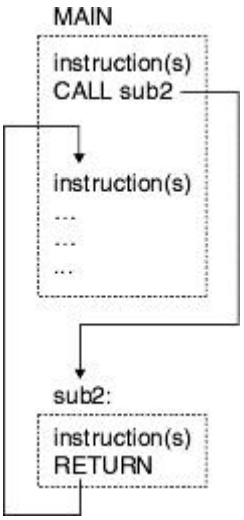
A subroutine is often coded so that it can be started (called) several times and from several places during one execution of the program, including from other subroutines, and then branch back (return) to the next instruction after the call, once the subroutine's task is done.

The CALL instruction interrupts the flow of a program by passing control to an internal or external subroutine. An internal subroutine is part of the calling program. An external subroutine is another program. The RETURN instruction returns control from a subroutine back to the calling program and optionally returns a value. For more detailed information on the CALL and RETURN instructions, see sections CALL and RETURN.

When calling an internal subroutine, CALL passes control to a label specified after the CALL keyword. When the subroutine ends with the RETURN instruction, the instructions following CALL are processed.



When calling an external subroutine, CALL passes control to the program name that is specified after the CALL keyword. When the external subroutine completes, you can use the RETURN instruction to return to where you left off in the calling program.



SIMPLE PROGRAM: 8 BIT ADDITION

ADDITION OF TWO 8 BIT NUMBERS

AIM:

To perform addition of two 8 bit numbers using 8085.

ALGORITHM:

1) Start the program by loading the first data into Accumulator.
2) Move the data to a register (B register).
3) Get the second data and load into Accumulator.
4) Add the two register contents.
5) Check for carry.
6) Store the value of sum and carry in memory location.
7) Terminate the program.

PROGRAM:

```
MVI C, 00        'Initialize C register to 00
LDA 4150         'Load the value to Accumulator.
MOV B, A         'Move the content of Accumulator to B register.
LDA 4151         'Load the value to Accumulator.
ADD B                'Add the value of register B to A
```

```
JNC LOOP        'Jump on no carry.
INR C           'Increment value of register C
LOOP: STA 4152  'Store the value of Accumulator (SUM).
MOV A, C        'Move content of register C to Acc.
STA 4153        'Store the value of Accumulator (CARRY)
HLT             'Halt the program.
```

OBSERVATION:

Input: 80 (4150)
80 (4251)
Output: 00 (4152)
01 (4153)

RESULT:

Thus the program to add two 8-bit numbers was executed.

SUBTRACTION-

SUBTRACTION OF TWO 8 BIT NUMBERS

AIM:

To perform the subtraction of two 8 bit numbers using 8085.

ALGORITHM:

1. Start the program by loading the first data into Accumulator.
   Move the data to a register (B register).
2. Get the second data and load into Accumulator.
3. Subtract the two register contents.
4. Check for carry.
5. If carry is present take 2's complement of Accumulator.
6. Store the value of borrow in memory location.
7. Store the difference value (present in Accumulator) to a memory
8. location and terminate the program.

PROGRAM:

```
MVI C, 00I Initialize C to 00
LDA 4150   Load the value to Acc.
MOV B, A   Move the content of Acc to B register.
LDA 4151   Load the value to Acc.
SUB B
JNC LOOP   Jump on no carry.
CMA        Complement Accumulator contents.
INR A      Increment value in Accumulator.
INR C      Increment value in register C
LOOP: STA 4152  Store the value of A-reg to memory address.
MOV A, C   Move contents of register C to Accumulator.
```

STA 4153 Store the value of Accumulator memory address.
HLT Terminate the program.

OBSERVATION:

Input: 06 (4150)
02 (4251)
Output: 04 (4152)
01 (4153)

RESULT:

Thus the program to subtract two 8-bit numbers was executed.

MULTIPLICATION

MULTIPLICATION OF TWO 8 BIT NUMBERS

AIM:

To perform the multiplication of two 8 bit numbers using 8085.

ALGORITHM:

1) Start the program by loading HL register pair with address of memory location.
2) Move the data to a register (B register).
3) Get the second data and load into Accumulator.
Add the two register contents
4) Check for carry.
Increment the value of carry.
5) Check whether repeated addition is over and store the value of product and carry in memory location.
6) Terminate the program.

PROGRAM:

MVI D,00 Initialize register D to 00
MVI A,00 Initialize Accumulator content to 00
LXI H,4150
MOV B,M Get the first number in B - reg
INX H
MOV C,M Get the second number in C- reg.
LOOP: ADD B Add content of A - reg to register B.
JNC NEXT Jump on no carry to NEXT.
INR D Increment content of register D
NEXT: DCR C Decrement content of register C.
JNZ LOOP Jump on no zero to address
STA 4152 Store the result in Memory
MOV A, D
STA 4153 Store the MSB of result in Memory

HLT Terminate the program.

OBSERVATION:

FF (4150)

Input:

FF (4151)

01 (4152)

Output:

FE (4153)

RESULT:

Thus the program to multiply two 8-bit numbers was executed.

FINDING LARGEST

LARGEST NUMBER IN AN ARRAY OF DATA

AIM:

To find the largest number in an array of data using 8085 instruction set.

ALGORITHM:

1) Load the address of the first element of the array in HL pair
2) Move the count to B – reg.
3) Increment the pointer
4) Get the first data in A – reg.
5) Decrement the count.
6) Increment the pointer
7) Compare the content of memory addressed by HL pair with that of A - reg.
8) If Carry = 0, go to step 10 or if Carry = 1 go to step 9
9) Move the content of memory addressed by HL to A – reg.
10) Decrement the count
11) Check for Zero of the count. If ZF = 0, go to step 6, or if ZF = 1 go to next step.
12) Store the largest data in memory.
13) Terminate the program.

PROGRAM:

```
        LXI H,4200 Set pointer for array
        MOV B,M    Load the Count
        INX H      Set 1st element as largest data
        MOV A,M
        DCR B      Decrements the count
LOOP:   INX H
        CMP M      f A- reg > M go to AHEAD
        JNC AHEAD
        MOVA,M     Set the new value as largest
AHEAD:  DCR B
        JNZ LOOP   Repeat comparisons till count = 0
        STA 4300   Store the largest value at 4300
        HLT
```

OBSERVATION:

Input: 05 (4200) ----- Array Size

Output: 0A (4201)
F1 (4202)
1F (4203)
26 (4204)
FE (4205)
FE (4300)

RESULT:
Thus the program to find the largest number in an array of data was executed

AND SMALLEST NUMBER,

SMALLEST NUMBER IN AN ARRAY OF DATA
AIM:
To find the smallest number in an array of data using 8085 instruction set.

ALGORITHM:
1) Load the address of the first element of the array in HL pair
2) Move the count to B – reg.
3) Increment the pointer
4) Get the first data in A – reg.
5) Decrement the count.
6) Increment the pointer
7) Compare the content of memory addressed by HL pair with that of A - reg.
8) If carry = 1, go to step 10 or if Carry = 0 go to step 9
9) Move the content of memory addressed by HL to A – reg.
10) Decrement the count
11) Check for Zero of the count. If ZF = 0, go to step 6, or if ZF = 1 go to next step.
12) Store the smallest data in memory.
13) Terminate the program.

PROGRAM:

```
        LXI H,4200 Set pointer for array
        MOV B,M    Load the Count
        INX  H     Set 1st element as largest data
        MOV A,M
        DCR B      Decremented the count
LOOP:   INX H
        CMP M      If A- reg < M go to AHEAD
        JC AHEAD
        MOV A,M    Set the new value as smallest
AHEAD:  DCR B
        JNZ LOOP   Repeat comparisons till count = 0
        STA 4300   Store the largest value at 4300
        HLT
```

OBSERVATION:

05 (4200) ----- Array Size

Input:

0A (4201)
F1 (4202)
1F (4203)
26 (4204)
FE (4205)
0A (4300)

Output:

RESULT:

Thus the program to find the smallest number in an array of data was executed

ASCENDING

Arrange Numbers in an Ascending Order

Write a program to arrange first 10 numbers from memory address 3000H in an ascending order.

```
MVI B, 09        :"Initialize counter"
START            :"LXI H, 3000H: Initialize memory pointer"
MVI C, 09H       :"Initialize counter 2"
BACK: MOV A, M   :"Get the number"
INX H            :"Increment memory pointer"
CMP M            :"Compare number with next number"
JC SKIP          :"If less, don't interchange"
JZ SKIP          :"If equal, don't interchange"
MOV D, M
MOV M, A
DCX H
MOV M, D
INX H            :"Interchange two numbers"
SKIP:DCR C       :"Decrement counter 2"
JNZ BACK         :"If not zero, repeat"
DCR B            :"Decrement counter 1"
JNZ START
HLT              :"Terminate program execution"
```

DESCENDING ORDER

An Assembly Language Program to arrange an array of data in descending order using 8085?

ALGORITHM

1) Initialize HL pair as memory pointer.
2) Get the count at 4200 in to C register.

3) Copy it in D register.
4) Get the first vale in Accumulator.
5) Compare it with the value at next location.
6) If they are out of order, exchange the contents of accumulator and memory.

7) Decrement D register's content by 1.
8) Repeat steps 5 and 7 till the value in D register become zero.
9) Decrement C register's content by 1.
10) Repeat steps 3 to 9 till the value in C register becomes zero.
11) Terminate the program.

Program

4400 LXI H,4200 21 Load the array size to the HL pair
4401 00
4402 42
4403 MOV C,M 4E Copy the array size to C register
4404 DCR C 0D Decrement C by 1
4405 REPEAT MOV D,C 51 Copy content of C to D register
4406 LXI H,4201 21 Load the first data to the HL pair
4407 01
4408 42
4409 LOOP MOV A,M 7E Copy the data to the accumulator
440A INX H 23 Increment memory by 1
440B CMP M BE Compare accumulator and memory content
440C JNC SKIP D2 Jump on no carry to the label SKIP
440D 14
440E 44
440F MOV B,M 46 Copy memory content to B register
4410 MOV M,A 77 Copy accumulator content to memory
4411 DCX H 2B Decrement memory by 1
4412 MOV M,B 70 Copy B register's content to memory
4413 INX H 23 Increment memory by 1
4414 SKIP DCR D 15 Decrement D by 1
4415 JNZ LOOP C2 Jump on non-zero to the label LOOP
4416 09
4417 44
4418 DCR C 0D Decrement C by 1
4419 JNZ REPEAT C2 Jump on non-zero to the label REPEAT
441A 05
441B 44
441C HLT 76
Program ends

OBSERVATION

Input at 4200 : 05H ---------- Array Size
4201 : 01H
4202 : 02H
4203 : 03H
4204 : 04H
4205 : 05H
Output at 4200 : 05H ---------- Array Size
4201 : 05H
4202 : 04H
4203 : 03H
4204 : 02H
4205 : 01H

16 BIT ADDITION,

8085 Program To Add Two 16 Bit Numbers
Let us suppose that the two 16 bit numbers are stored in 9000H and 9002H. These two 16 but numbers are to be added and the result is to be stored in memory location 9004 and the carry (if generated) is to be stored in memory location 9006H.

ALGORITHM:

1. Start.
2. Initialize register C for using it as a counter for storing carry value.
3. Load data into HL register pair from one memory address (9000H).
4. Exchange contents of register pair HL with DE.
5. Load second data into HL register pair (from 9002H).
6. Add register pair DE with HL and store the result in HL.
7. If carry is present, go to 8 else go to 9.
8. Increment register C by 1.
9. Store value present in register pair HL to 9004H.
10. Move content of register C to accumulator A.
11. Store value present in accumulator(carry) into 9006H.
12. Terminate the program.

PROGRAM CODE:

```
MVI C, 00H
LHLD 9000H
XCHG
LHLD 9002H
DAD D
JNC AHEAD
INR C
AHEAD:
SHLD 9004H
MOV A, C
STA 9006H
HLT
```

The register C is initialized to 00 to store the carry value. The first 16 bit number is loaded into HL register pair. 9000H stores the lower significant 8 bits and 9001H stores the upper significant 8 bits meaning that if 1234 is the 16 bit number stored from memory location 9000H then the 16 bit numbers are stored in memory as:
9000H - 34
9001H - 12

The first 16 bit number is then stored in DE register pair with the help of XCHG instruction. The second 16 bit number (stored is 9002H and 9003H) is then loaded into the HL register pair. The contents of DE register pair is then added with the contents of HL register pair and the result is stored in HL register pair. If carry is generated during the addition then register C is incremented by 1. But if no carry is present then the program skips the INR C instruction. The contents of HL register pair is then stored in memory location 9004H which is the sum value. Then the contents of register C is moved into the accumulator and then stored in memory location 9006H which is the carry value.

Output:

Address	Data
9000	34
9001	12
9002	34
9003	12
9004	68
9005	24
9006	00

Here the first 16 bit number is stored in 9000H and 9001H and the second 16 bit number is stored in 9002H and 9003H. The addition takes place as:

1234
+1234

2468

The sum value is stored in memory location 9004H and 9005H with lower 2 significant bits at 9004H and the upper 2 significant bits at 9005H. Since no carry is generated so 00 is stored in 9006H.

Possible Questions

3 Marks

1) Write a short note on arithmetic instructions of 8085.

2) Write a 8085 program to add two 16 bit numbers.

3) Write a short note on conditional branching instructions in 8085.

4) Write an 8085 program to multiply two 8-bit numbers.

5) Explain logical instructions of 8085.

6) Write a μp 8085 program for a set of numbers to arrange in ascending order.

7) Discuss about 8085 data transfer instructions.

8) Write an 8085 program to add and subtract two 8-bit numbers.

9) Discuss the addressing modes of 8085.

10) Write an 8085 program to find the largest and smallest numbers

10 Marks

1)Write a microprocessor program to find largest and smallest number.

2)Write program to find the largest and smallest number.

UNIT-IV

What is the output of the following code: PUSH AL	Decrement SP by 2 & push a word to stack	Increment SP by 2 & push a word to stack	Decrement SP by 2 & push a AL to stack	Illegal	Illegal
What is SIM?	Select interrupt mask	Sorting interrupt mask	Set interrupt mask	Softer interrupt mask	Set interrupt mask
CMC is ____ byte instructions.	1	2	3	4	1
When DI instructions is executed all the interrupts except _____ are disabled.	RST 4.5	RST 5.5	TRAP	RST 6.5	TRAP
Which of the following is a one-byte instruction?	MVI B, 05	LDA 250 H	IN 01	MOV A,B	MOV A,B
Which of the following is not a data copy/transfer instruction?	MOV	PUSH	DAS	POP	DAS
In PUSH instruction, after each execution of the instruction, the stack pointer is	incremented by 1	decremented by 1	ncremented by 2	decremented by 2	decremented by 2
The instructions that are used for reading an input port and writing an output port respectively are	MOV, XCHG	MOV, IN	IN, MOV	IN, OUT	IN, OUT
The instruction that loads effective address formed by destination operand into the specified source register is	LEA	LDS	less than	LAHF	LEA
The instruction that loads the flag register completely from the word contents of the memory location is	PUSH	POP	PUSHF	POPF	POPF
The instruction that loads the AH register with the lower byte of the flag register is	SAHF	AH	LAHF	PUSHF	LAHF
The mnemonic that is placed before the arithmetic operation is performed is	AAA	AAS	AAM	AAD	AAD
The Carry flag is undefined after performing the operation	AAA	ADC	AAM	AAD	AAD
The instruction that performs logical AND operation and the result of the operation is not available is	AAA	AND	TEST	XOR	TEST
In the RCL instruction, the contents of the destination operand undergoes function as	carry flag is pushed into LSB & MSB is pushed into carry flag	carry flag is pushed into MSB & LSB is pushed into carry flag	auxiliary flag is pushed into LSB & MSB is pushed into carry flag	parity flag is pushed into MSB & LSB is pushed into carry flag	carry flag is pushed into LSB & MSB is pushed into carry flag
The instruction that is used as prefix to an instruction to execute it repeatedly until the CX register becomes zero is	SCAS	REP	CMPS	STOS	REP
The instructions that are used to call a subroutine from a main program and return to the main program after execution of called function are	CALL, JMP	JMP, IRET	CALL, RET	JMP, RET	CALL, RET
The instruction that unconditionally transfers the control of execution to the specified address is	CALL	JMP	RET	IRET	JMP
NOP instruction introduces	address	delay	memory location	none	delay
Which of the following is not a machine controlled instruction?	HLT	CLC	LOCK	ESC	CLC
The conditional branch instructions JNS performs the operations when if	ZF=0	SF=0	PF=0	CF=0	SF=0
HLT is	1 byte instruction	2 byte instructions	4 byte instructions	3 byte instructions	1 byte instruction
8085 has _____ basic instructions	256	246	70	74	74
Number of opcodes used in 8085 instructions is _____	256	246	70	74	246
Maximum number of I/O that can be addressed by the INTEL 8085 is	65536	285	512	256	256
After the execution of CMP A instruction	ZF is set and CY is reset	ZF is set and CY is unchanged	ZF is reset and CY is set	ZF is reset and CY is unchanged	ZF is set and CY is reset
The ALE line of 8085 microprocessor is used to	latch the output of an I/O instruction into an external latch	deactivate the chip-select signal from memory device	latch the 8-bit of address lines AD0-AD7 into an external latch	find the interrupt enable status of the TRAP interrupt	latch the 8-bit of address lines AD0-AD7 into an external latch

The load instruction is mostly used to designate a transfer from memory to a processor register known as____.	Accumulator	Instruction Register	Program counter	Memory address Register	Accumulator
MIMD stands for _____.	Multiple instruction multiple data	Multiple instruction memory data	Memory instruction multiple data	Multiple information memory data	Multiple instruction multiple data
Logic gates with a set of input and outputs is arrangement of_____.	Computational circuit	Logic circuit	Design circuits	Register	Computational circuit
The BSA instruction is_____.	Branch and store accumulator	Branch and save return address	Branch and shift address	Branch and show accumulator	Branch and save return address
Instructions are classified into___ groups	2	3	4	5	5
Instructions to load H and L registers directly from memory is	LDA	LDAX	LHLD	LLHD	LHLD
Which of the following is not an arithmetic instruction?	ADD	INR	CMA	DCX	CMA
MVI M, d8 is _____	1 byte instruction	two byte instruction	four byte instruction	three byte instruction	two byte instruction
The circuits in the 8085A that provide the arithmetic and logic functions are called the:	CPU	ALU	I/O	NONE	ALU
Instructions performing actions are called	imperative statements	declarative statemenets	directive statements	none	imperative statements
number of machine cyle required for RET instructions in 8085 microprocessor is	1	2	3	5	3
How many memory locations required to store the instructions LXIH,0800H in an 8085 assembly language program?	1	2	3	4	2
The insttruction DEC N inform the assembler to	decrement the content of N	decrement the data addressing by N	convert signed decimal number to binary	none	decrement the content of N

INTERFACING PERIPHERALS AND APPLICATIONS: INTERFACING CONCEPTS-

Peripheral Interfacing is considered to be a main part of Microprocessor, as it is the only way to interact with the external world. The interfacing happens with the ports of the Microprocessor.

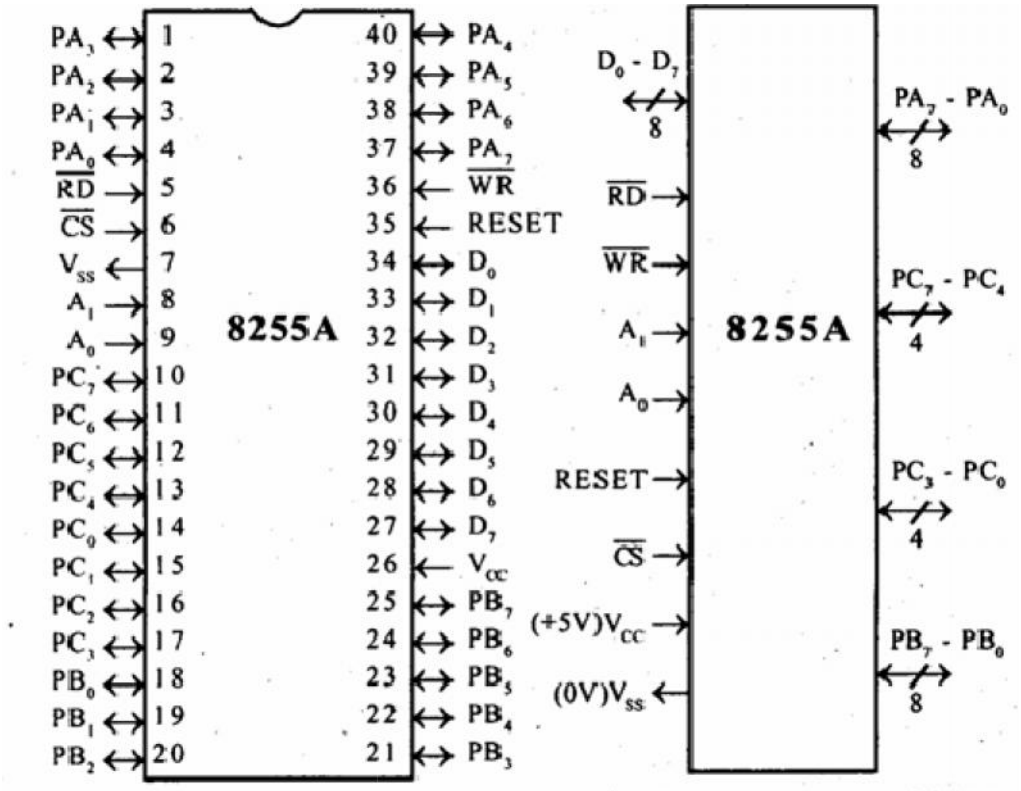
- The main IC's which are to be interfaced with 8085 are:

1. 8255 PPI
2. 8259 PIC
3. 8251 USART
4. 8279 Key board display controller
5. 8253 Timer/ Counter
6. A/D and D/A converter interfacing.

PROGRAMMABLE PERIPHERAL INTERFACE - INTEL 8255

Pins, Signals and internal block diagram of 8255:

- It has 40 pins and requires a single +5V supply.



- The INTEL 8255 is a device used to parallel data transfer between processor and slow peripheral devices like ADC, DAC, keyboard, 7-segment display, LCD, etc.

- The 8255 has three ports: Port-A, Port-B and Port-C.

- Port-A can be programmed to work in any one of the three operating modes mode-0, mode-1 and mode-2 as input or output port.
- Port-B can be programmed to work either in mode-0 or mode-1 as input or output port.
- Port-C (8-pins) has different assignments depending on the mode of port-A and port-B.
- If port-A and B are programmed in mode-0, then the port-C can perform any one of the following functions.
- As 8-bit parallel port in mode-0 for input or output.
- As two numbers of 4-bit parallel ports in mode-0 for input or output.
- The individual pins of port-C can be set or reset for various control applications.
- If port-A is programmed in mode- 1/mode-2 and port-B is programmed in mode-1 then some of the pins of port-C are used for handshake signals and the remaining pins can be used as input/ output lines or individually set/reset for control applications.
- The read/write control logic requires six control signals. These signals are given below.

1. RD (low): This control signal enables the read operation. When this signal is low, the microprocessor reads data from a selected I/O port of the 8255A.

2. WR (low): This control signal enables the write operation. When this signal goes low, the microprocessor writes into a selected I/O port or the control register.

3. RESET: This is an active high signal. It clears the control register and set all ports in the input mode.

4. CS (low), A0 and A1: These are device select signals. They are,

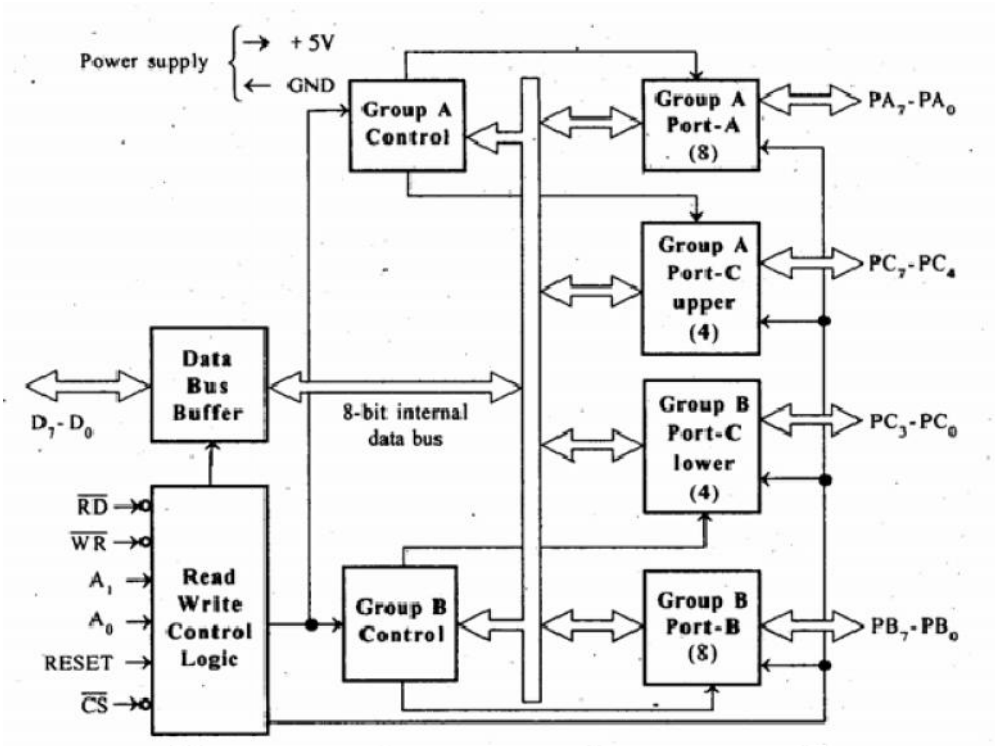
Interfacing of 8255 with 8085 processor:

- A simple schematic for interfacing the 8255 with 8085 processor is shown in fig.

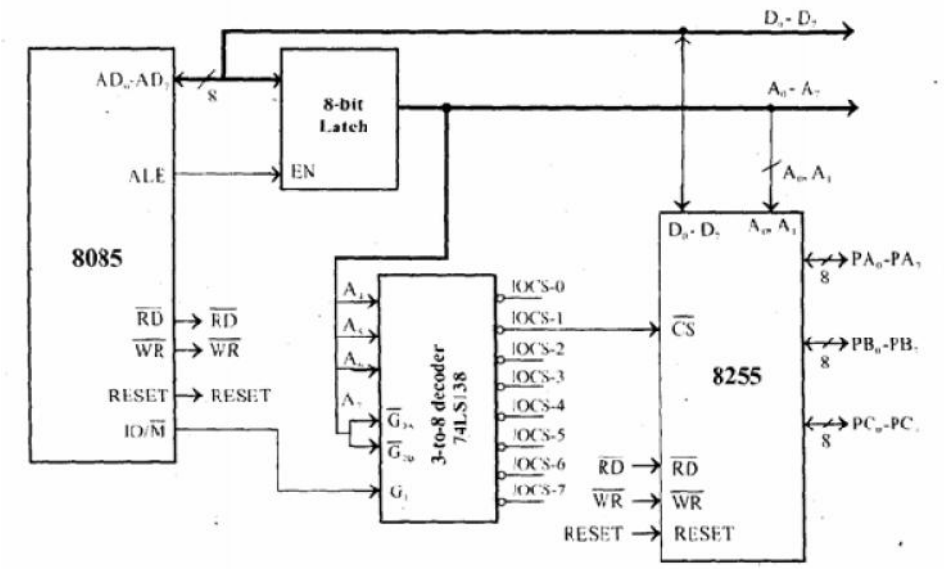
PIN DESCRIPTION - INTEL 8255

Pin	Description
D_0 - D_7	Data lines
RESET	Reset input
$\overline{CS}$	Chip select
$\overline{RD}$	Read control
$\overline{WR}$	Write control
A_0, A_1	Internal address
PA_7 - PA_0	Port-A pins
PB_7 - PB_0	Port-B pins
PC_7 - PC_0	Port-C pins
V_{CC}	+5V
V_{SS}	0V (GND)

- The internal block diagram of 8255 is shown in fig:



Internal Devices	A_1	A_0
Port A	0	0
Port B	0	1
Port C	1	0
Control Register	1	1



- The 8255 can be either memory mapped or I/O mapped in the system. In the schematic shown in above is I/O mapped in the system.
- Using a 3-to-8 decoder generates the chip select signals for I/O mapped devices.
- The address lines A4, A5 and A6 are decoded to generate eight chip select signals (IOCS-0 to IOCS-7) and in this, the chip select IOCS- 1 is used to select 8255.
- The address line A7 and the control signal IO/M (low) are used as enable for the decoder.
- The address line A0 of 8085 is connected to A0 of 8255 and A1 of 8085 is connected to A1 of 8255 to provide the internal addresses.
- The data lines D0-D7 are connected to D0-D7 of the processor to achieve parallel data transfer.
- The I/O addresses allotted to the internal devices of 8255 are listed in table.

Internal Device	Binary Address								Hexa Address
	Decoder input and enable				Input to address pins of 8255				
	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	
Port-A	0	0	0	1	x	x	0	0	10
Port-B	0	0	0	1	x	x	0	1	11
Port-C	0	0	0	1	x	x	1	0	12
Control Register	0	0	0	1	x	x	1	1	13

Note : *Don't care "x" is considered as zero.*

PERIPHERAL I/O INSTRUCTIONS-

Memory Interfacing

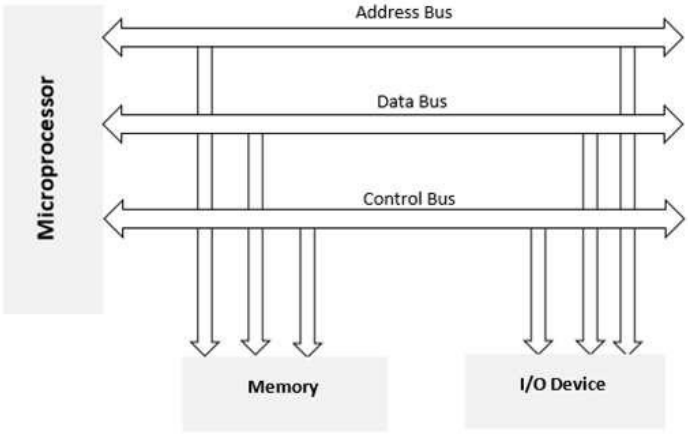
When we are executing any instruction, we need the microprocessor to access the memory for reading instruction codes and the data stored in the memory. For this, both the memory and the microprocessor requires some signals to read from and write to registers.

The interfacing process includes some key factors to match with the memory requirements and microprocessor signals. The interfacing circuit therefore should be designed in such a way that it matches the memory signal requirements with the signals of the microprocessor.

I/O Interfacing

There are various communication devices like the keyboard, mouse, printer, etc. So, we need to interface the keyboard and other devices with the microprocessor by using latches and buffers. This type of interfacing is known as I/O interfacing.

Block Diagram of Memory and I/O Interfacing



8085 Interfacing Pins

Following is the list of 8085 pins used for interfacing with other devices

- A_{15} - A_8 (Higher Address Bus)
- AD_7 - AD_0(Lower Address/Data Bus)
- ALE
- RD
- WR
- READY

There are two ways of communication in which the microprocessor can connect with the outside world.

- Serial Communication Interface
- Parallel Communication interface

Serial Communication Interface − In this type of communication, the interface gets a single byte of data from the microprocessor and sends it bit by bit to the other system serially and vice-a-versa.

Parallel Communication Interface − In this type of communication, the interface gets a byte of data from the microprocessor and sends it bit by bit to the other systems in simultaneous (or) parallel fashion and vice-a-versa.

INTERFACING PROGRAMS

LED INTERFACING,

Light Emitting Diodes (**LED**) is the most commonly used components, usually for displaying pins digital states. Typical uses of LEDs include alarm devices, timers and confirmation of user input such as a mouse click or keystroke.

INTERFACING LED

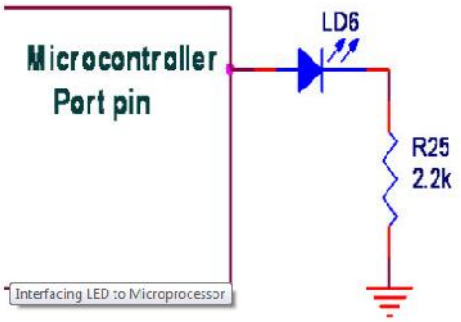


Fig. 1 Interfacing LED to Microprocessor

Above Fig. 1 shows how to interface the LED to microprocessor. As you can see the Anode is connected through a resistor to GND & the Cathode is connected to the Microprocessor pin. So when the Port Pin is HIGH the LED is OFF & when the Port Pin is LOW the LED is turned ON.

ASSEMBLY PROGRAM TO ON AND OFF LED USING 8085

CNTRL : 0X23H PORT B : 0X21H

ADDRESS	OPCODE	MNEMONICS
8500	3E 80	MVI A, 80H
8502	D3 0F	OUT CNTRL
8504		**START:**
8504	3E 00	MVI A, 00H
8506	D3 0D	OUT PORTB
8508	CD 15 85	CALL DELAY
850B	3E FF	MVI A, FFH
850D	D3 0D	OUT PORTB
850F	CD 15 85	CALL **DELAY**
8512	C3 04 85	JMP **START**
8515		**DELAY:**
8515	06 05	MVI B, 05
8517	0E FF	L1: MVI C, FFH
8519	0D	L2: DCR C
851A	C2 19 85	JNZ L2
851D	05	DCR B
8520	C2 17 85	JNZ L1
8521	C9	RET

STEPPER MOTOR INTERFACING,

A stepper motor is a widely used device that translates electrical pulses into mechanical movement. In applications such as disk drives, dot matrix printers, and robotics, the stepper motor is used for position control. Stepper motors commonly have a permanent magnetrotor surrounded by a stator. There are also steppers called variable reluctance stepper motors that do not have a PM rotor. The most common stepper motors have four stator windings that are paired with a center-tapped. This type of stepper motor is commonly referred to as *a.* four-phase or unipolar stepper motor. The center tap allows a change of current direction in each of two coils when a winding is grounded, thereby resulting in a polarity change of the stator.

INTERFACING STEPPER MOTOR

We can see the stepper moter is connected with microprocessor output port pins through aULN2803A array. So when the microprocessor is giving pulses with particular frequency the motor is rotated in clockwise or anticlockwise.

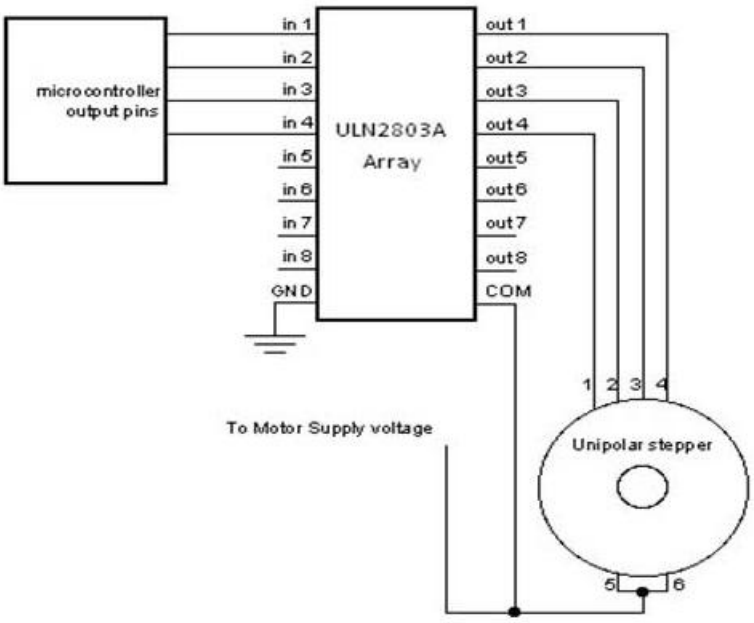


Fig. 2Interfacing Stepper Motor to Microprocessor

ASSEMBLY PROGRAM TO INTERFACE STEPPER MOTOR WITH 8085

Title : **Program to rotate the stepper motor in clockwise direction**

ADDRESS	OPCODE	MNEMONICS
9100	3E 80	MVI A,80
9102	D3 23	OUT 23
9104	21 00 92	START: LXI H,9200
9107	06 04	MVI B,04
9109	7E	REPET : MOV A,M
910A	D3 20	OUT 20
910C	21 02 02	LXI H,0202
910F	00	DELAY: NOP
9110	1B	DCX D
9111	7E	MOV A,E
9112	B2	ORA D
9113	C2 0F 91	JNZ DELAY

```
9116        23                  INX H
9117        05                  DCR B
9118        C2 09 91            JNZ  REPET
911B        C3 04 91            JNZ  START
```

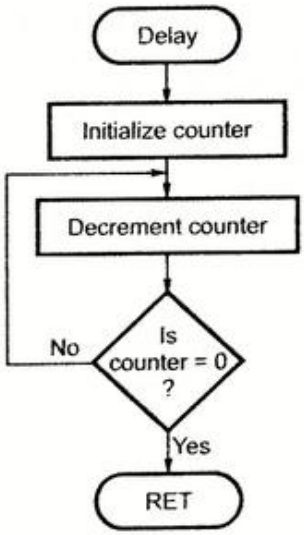
LOOK UP TABLE

```
            DATA
9200        DB 03H,06H,0CH,09H (CLOCK WISE)

9204        END
```

HEX KEYBOARD INTERFACING.

Keyboard interface(64-key-matrix-keyboard) Flow Chat



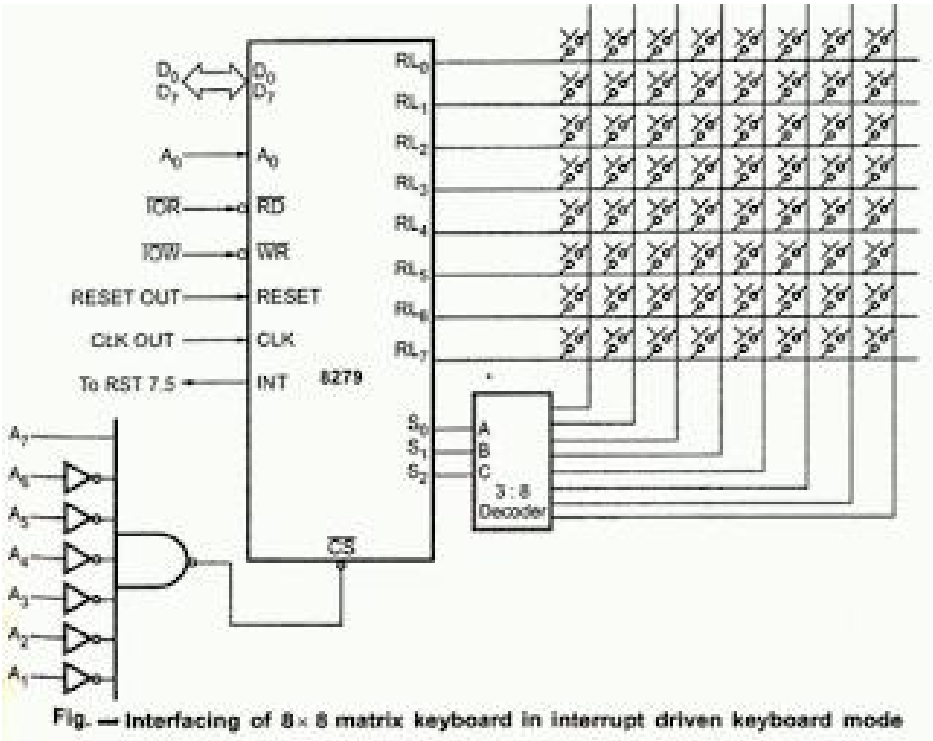


Fig. — Interfacing of 8 × 8 matrix keyboard in interrupt driven keyboard mode

Hardware For Matrix Keyboard Interface

Fig. shows a matrix keyboard with 64 keys connected to the 8085 microprocessor using 8255. A matrix keyboard reduces the number of connections, thus the number of interfacing lines. In this example, the keyboard with 64 keys, is arranged in 8 x 8 (8 rows and 8 columns) matrix. This requires sixteen lines from the microprocessor to make all the connections instead of 64 lines if the keys are connected individually. The interfacing of matrix keyboard requires two ports: one input port and other output port. Rows are connected to the input port, port A and columns are connected to the output port, port B

Software For Matrix Keyboard Interface

```
Source program
MVI A, 90H          : Initialize Port A as input and
OUT CR              : Port B as Output
START: MVI A, 00    : Make all scan lines zero
OUT PB
BACK: IN PA
CPI FF              : Check for key release
JNZ BACK            : If not, wait for key release
CALL DELAY          : Wait for key debounce
BACK 1: IN PA
CPI FF              : Check for key press
JZ BACK 1           : If not, wait for key press
CALL DELAY          : Wait for key debounce
MVI L, 00H          : Initialize key counter
MVI C, 08H
MVI B, FEH          : Make one column low
NEXTCOL: MOV A, B
OUT PB
MVI D, 08H          : Initialize row counter
IN PA               : Read return line status
NEXTROW: RRC        : Check for one row
JNC DISPLAY         : If zero, goto display else continue
INR L               : Increment key counter
DCR D               : Decrement row counter
JNZ NEXTROW         : Check for next row
MOV A, B
RLC                 : Select the next column
MOV B, A
DCR C               : Decrement column count
JNZ NEXTCOL         : Check for last column if not repeat
JMP START           : Go to start
INTERFACING SCHEME
```

Delay subroutine:

```
Delay: LXI D, Count
Back: DCX D
MOV A, D
ORA E
JNZ Back
RET
```

Possible Questions

3 Marks

1) Explain LED interfacing using ports.

2) Draw a flowchart and explain hex keyboard interfacing.

3) Explain any one conversion technique of analog to digital.

4) Write a program to interface stepper motor.

5) Construct a block diagram of 8255 and explain it.

6) With a neat diagram explain R/2R ladder D/A converter.

7) Draw the circuit for keyboard and display interfacing using 8279.

8) Discuss about single slope ADC

9) Discuss about the features of ADC family.

10) Write a program to interface LED.

10 Marks

1) Explain the working of Stepper motor interface.

2) Explain LED interface.

UNIT-V					
Programmable peripheral input-output port is other name for	serial input-output port	parallel input-output port	serial input port	parallel output port	parallel input-output port
Port C of 8255 can function independently as	input port	output port	either input or output ports	both input and output ports	either input or output ports
All the functions of the ports of 8255 are achieved by programming the bits of an internal register called	data bus control	read logic control	control word register	none	control word register
The data bus buffer is controlled by	control word register	read/write control logic	data bus	none	read/write control logic
The popular technique that is used in the integration of ADC chips is	successive approximation	dual slope integration	successive approximation and dual slope integration	none	successive approximation and dual slope integration
The procedure of algorithm for interfacing ADC contain	ensuring stability of analog input	issuing start of conversion pulse to ADC	reading digital data output of ADC as equivalent digital output	All of the above	All of the above
Which is the ADC among the following?	AD 7523	74373	74245	ICL7109	ICL7109
The conversion delay in successive approximation of an ADC 0808/0809 is	100 milliseconds	100 microseconds	50 milliseconds	50microseconds	100 microseconds
The number of inputs that can be connected at a time to an ADC that is integrated with successive approximation is	4	2	8	16	8
ADC 7109 integrated by Dual slope integration technique is used for	low cost option	slow practical applications	low complexity	all of the mentioned	all of the mentioned
In the signal integrate phase, the differential input voltage between IN LO(input low) and IN HI(input high) pins is integrated by the internal integrator for a fixed period of	256 clock cycles	1024 clock cycles	2048 clock cycles	4096 clock cycles	2048 clock cycles
DAC (Digital to Analog Converter) finds application in	digitally controlled gains	motor speed controls	programmable gain amplifiers	all of the mentioned	all of the mentioned
To save the DAC from negative transients the device connected between OUT1 and OUT2 of AD 7523 is	p-n junction diode	zener	FET	BJT	zener
An operational amplifier connected to the output of AD 7523 is used	to convert current output to output voltage	to provide additional driving capability	as current-to-voltage converter	all of the mentioned	all of the mentioned
The DAC 0800 has a settling time of	100 milliseconds	100 microseconds	50 milliseconds	50microseconds	100 milliseconds
The device that is used to obtain an accurate position control of rotating shafts in terms of steps is	DC motor	AC motor	stepper motor	servo motor	stepper motor
The internal schematic of a typical stepper motor has	1 windings	2 windings	3 windings	4 windings	4 windings
The number of pulses required for one complete rotation of the shaft of the stepper motor is equal to the	number of internal teeth on a rotor	number of internal teeth on a stator	number of internal teeth on a rotor and stator	number of external teeth on a stator	number of internal teeth on a rotor
A simple scheme for rotating the shaft of a stepper motor is called	rotating scheme	shaft scheme	wave scheme	none	wave scheme
The firing angles of thyristors are controlled by	pulse generating circuits	relaxation oscillators	microprocessor	all of the mentioned	all of the mentioned
The Isolation transformers are generally used for	protecting low power circuit	protecting low power circuit and isolation	isolation	protecting high power circuit and isolation	protecting low power circuit and isolation

ADC, DAC, Hex-keyboard are using _____	parallel data transfer	serial data transfer	any data transfer	direct memory access data transfer	parallel data transfer
Which of the following is not the clasification of programmed data transfer?	synchronous data transfer scheme	asynchronous data transfer scheme	interrupt driven data transfer scheme	direct memory access data transfer	direct memory access data transfer
Which of the following is not the clasification of DMA?	cycle stealing	synchronous	block or burst mode	demand transfer mode	synchronous
Example of DMA are _____	ADC	DAC	CRT controller	7 segment LED	CRT controller
Example of parallel data transfer is _____	CRT controller	hard disk	floppy disk	ADC	ADC
The handshake data transfer without interrupt _____ is an example of synchronous data transfer	8085	8125	8215	8155	8155
The timer section of 8155 has two _____ registers	2-bit	4-bit	8-bit	16-bit	8-bit
The timer output of 8155 at mode-0 is	square wave	high	low	continuous wave	high
In 8155, Mode-3 is similar to _____	Mode-0	Mode-1	Mode-2	Mode-4	Mode-2
In simple I/O ports no handshake signals are exchanged between ____	input port	output port	I/O ports	I/O ports and I/O devices	I/O ports and I/O devices
The peripheral devices not interfaced to 8085 system are	INTEL 8254	INTEL 8251	INTEL 8086	INTEL 8237	INTEL 8086
How many 7447 is needed for 7-segment LED?	1	2	3	4	1
If the full scale output voltage is 10.2 V then by second definietion the resolution for an 8-bit DAC is	40 mV/LSB	0.4 mV/LSB	40 V/LSB	4 mV/LSB	40 mV/LSB
For a 8-bit DAC resolution is	216	256	236	266	256
Calculate the change in the output voltage if input changes by 5 V with FRR for S/H circuit is 80 dB	0.5 V	5 V	0.5mV	5 mV	0.5 mV
The leakage current causes voltage of the ____ to drop down	capacitor	resistor	diode	thyristor	capacitor
IC _____ is 3 to 8 decoder	74LS180	74LS148	74LS128	74LS138	74LS138
IC _____ is a 1 to 16 demultiplexer	74154	74144	74124	74134	74154
IC _____ is a 16 input multiplexer	74144	74150	74124	74134	74150
IC _____ ia 8 input multiplexer	74150	74124	74134	74151	74151
Which of the following has 2^n input and one output?	demultiplexer	decoder	encoder	multiplexer	multiplexer
IC _____ is quad 2 line to 1 line multiplexers	74157	74124	74134	74151	74157
INTEL 8212 is a ____pin I/O device	10	8	20	24	24
Which among the following can be used as a latch or a tri-state buffer?	8285	8212	8280	8279	8212
When strobe of 8155 ia LOW, BF is reset to _____	LOW	HIGH	ZERO	toggle	LOW
Intel 8255 has ____ ports	2	3	4	5	3
INTEL 8355 is a ROM and I/O port chip that can be used in ____ microprocessor	8085	8085 and 8086	8086 and 8088	8085A and 8088	8085A and 8088
In INTEL 8755, EPROM portion is organized as __ words by 8 bits.	256	2560	2048	248	2048
The keyboard section has ___ FIFO RAM	8 x 8	4 x 4	2 x 2	16 x 16	8 x 8
The time taken by the ADC from the active edge of SOC(start of conversion) pulse till the active edge of EOC(end of conversion) signal is called	edge time	conversion time	conversion delay	time delay	conversion delay
Which of the following is not one of the phase of total conversion cycle?	autozero phase	conversion phase	signal integrate phase	deintegrate phase	conversion phase

Which of the following phase contain feedback loop in it?	autozero phase	signal integrate phase	deintegrate phase	none	autozero phase
The port that is used for the generation of handshake lines in mode 1 or mode 2 is	port A	port B	port C LOWER	port C UPPER	port C UPPER
If A1=0, A0=1 then the input read cycle is performed from	port A to data bus	port B to data bus	port C to data bus	CWR to data bus	port B to data bus
The sensor RAM acts as 8-byte first-in-first-out RAM in	keyboard mode	strobed input mode	keyboard and strobed input mode	scanned sensor matrix mode	keyboard and strobed input mode
The registers that holds the address of the word currently being written by the CPU from the display RAM are	control and timing register	control and timing register and timing control	display RAM	display address registers	display address registers
The FIFO status word is used to indicate the error in	keyboard mode	strobed input mode	keyboard and strobed input mode	scanned sensor matrix mode	keyboard and strobed input mode