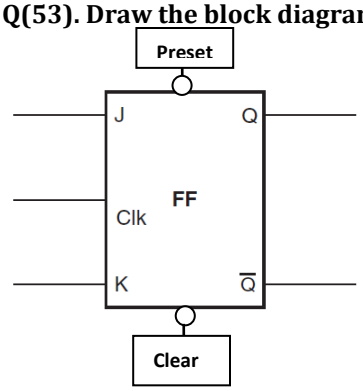
Q(53). Draw the block diagram of Master-Slave J-K flip flop showing 'Preset' & 'Clear' terminals?.



CLK	Preset	Clear	ACTION
1	1	1	Normal flip flop
0	0	1	Flip Flop will Set
0	1	0	Flip Flop will Reset

Q(54). Give the applications of flip flops?.

(i). As a frequency divider, (ii). In timer circuit, (iii). In shift registers, (iv). In counters, (v). As a memory device.

Q(55). Give the operation of flip flop as a frequency divider?.

Please refer the question number (50).

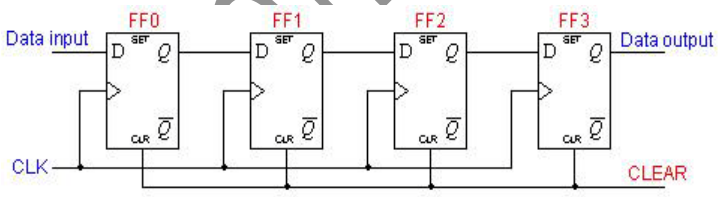
Q(56). Explain operating principle of shift register & give its types?.

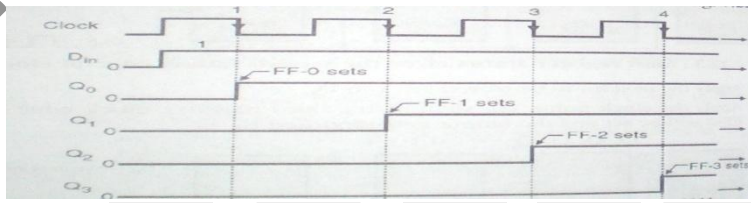
The shift register sequential circuit which is nothing but the collection of flip flops. It is used for storing temporary data or digital data. There are group of flip-flops connected in a chain so that the output from one flip-flop becomes the input of the next flip-flop. Most of the registers possess no characteristic internal sequence of states. All the flip-flops are driven by a common clock pulses, and all are set or reset simultaneously. For 2-bit shift register we required 2 number of flip flops, For 4-bit shift register we required 4 number of flip flops & For 8-bit shift register we required 8 number of flip flops & so on. Hence shift registers are used for data storage, data transfer & for performing arithmetic & logical operations. Following are the types of flip flops,

(i).SISO:- Serial Input Serial Output, (ii).SIPO:- Serial Input Parallel Output,

(iii).PISO:- Parallel Input Serial Output, (iv).PIPO:- Parallel Input Parallel Output.

Q(57). Draw the circuit & timing diagram (waveforms) of 4-bit SISO shift register with working?.





Stored Word:- 0000 | 1000 | 1100 | 1110 | 1111

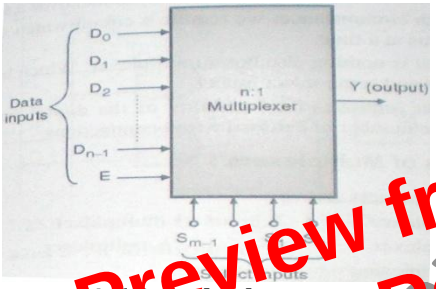
The SISO (Serial Input Serial Output) shift register (Right shift mode) is shown in above figure. Assume Q0 is the output for FF-0, assume Q1 is the output for FF-1, assume Q2 is the output for FF-2, assume Q3 is the output for FF-3, Following are the operations are possible,

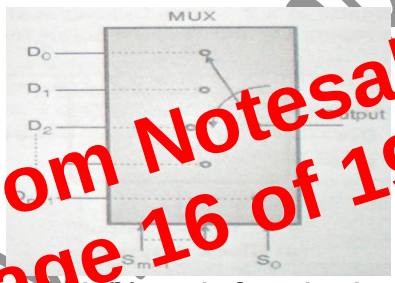
A *ripple counter* is a cascaded arrangement of flip-flops. A ripple counter is constructed using clocked JK-flip flops. In a ripple counter, also called an *asynchronous counter* or a *serial counter*, In this counter all the flip flops are not under the control of a single clock. The clock input is applied only to the first flip-flop, also called the input flip-flop, in the cascaded arrangement. The clock input to any subsequent flip-flop comes from the output of its immediately preceding flip-flop. For instance, the output of the first flip-flop acts as the clock input to the second flip-flop, the output of the second flip-flop feeds the clock input of the third flip-flop and so on.

That is, the second flip-flop would change state a certain time delay after the occurrence of the input clock pulse owing to the fact that it gets its own clock input from the output of the first flip-flop and not from the input clock. This time delay here equals the sum of propagation delays of two flip-flops, the first and the second flip-flops. In general, the nth flip-flop will change state only after a delay equal to n times the propagation delay of one flip-flop. The term 'ripple counter' comes from the mode in which the clock information ripples through the counter.

It is also called an 'asynchronous counter' as different flip-flops comprising the counter do not change state in synchronization with the input clock. In a counter like this, after the occurrence of each clock input pulse, the counter has to wait for a time period equal to the sum of propagation delays of all flip-flops before the next clock pulse can be applied. The propagation delay of each flip-flop, of course, will depend upon the logic family to which it belongs.

Q(60). Draw the circuit diagram of multiplexer (data selector) & also explain its operation?.

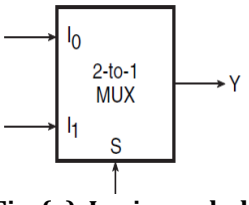


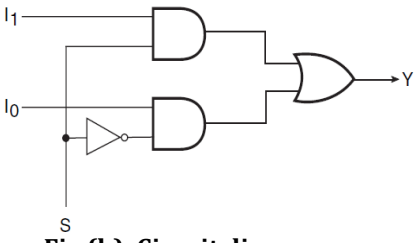


Fig(a). Multiplexer, Fig(b). Equivalent circuit.

A Multiplexer (MUX) is a combinational device having multiple data inputs, select inputs & only one output. The select input lines decide the state of output. Hence, a multiplexer can take many data bits and put them, one at a time, on a single output data line in a particular sequence. Hence also called as digitally controlled single pole multiple way switches. i.e. output line can be attached with only one input data line at the same time only. The multiplexer is also used for transforming *parallel* data to *serial* data. The relation between select inputs (i.e. 'm') & data inputs (i.e. 'n') can be stated as, $2^m=n$. the E input is called as Probe/Enable terminal which should active for performing required operation. Hence by this way multiplexer circuit reduces the number of wired connections & produces the proper digital code on single wire.
Following are the types of multiplexers,
(i). 2:1 multiplexer, (ii). 4:1 multiplexer, (iii). 8:1 multiplexer, (iv). 16:1 multiplexer.

Q(61). Draw logic symbol, circuit diagram & truth table of 2:1 multiplexer?.

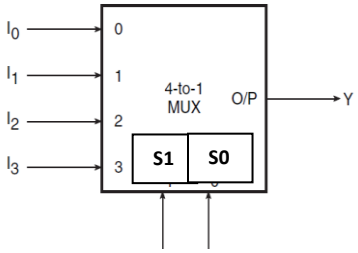


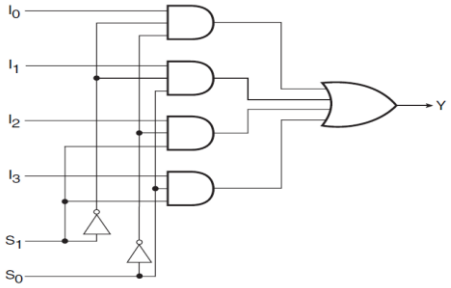


E	S	Y
1	0	Io
1	1	I_1

Fig. (a). Logic symbol, Fig.(b). Circuit diagram, Fig.(c). Truth table

Q(62). Draw logic symbol, circuit diagram & truth table of 4:1 multiplexer?.

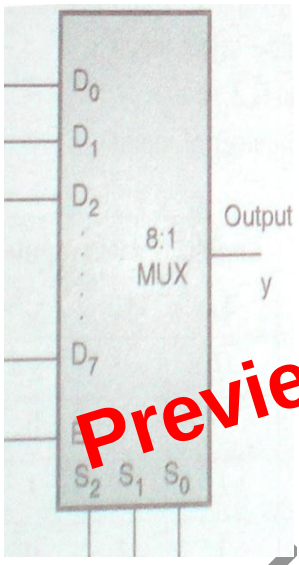


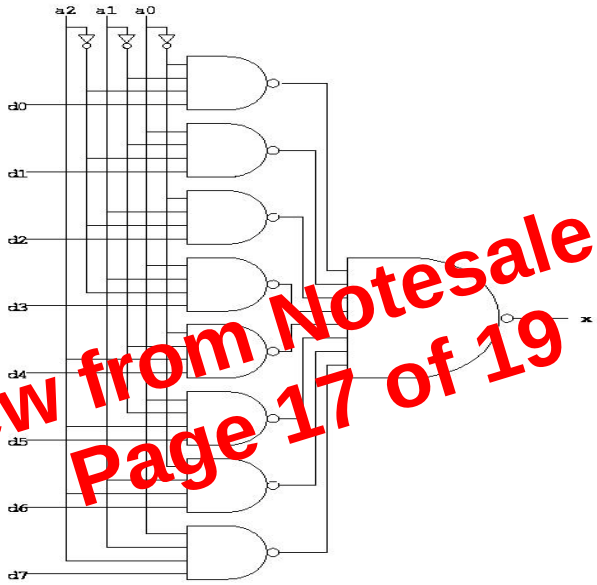


E	S1	S0	Y
1	0	0	Io
1	0	1	I_1
1	1	0	I_2
1	1	1	I_3

Fig. (a). Logic symbol, **Fig.(b). Circuit diagram,** **Fig.(c). Truth table**

Q(63). Draw logic symbol, circuit diagram & truth table of 8:1 multiplexer?.





E	S2	S1	S0	Y
1	0	0	0	Do
1	0	0	1	D_1
1	0	1	0	D_2
1	0	1	1	D_3
1	1	0	0	D_4
1	1	0	1	D_5
1	1	1	0	D_6
1	1	1	1	D_7

For working please refer question no.(60).

Q(64). Draw the circuit diagram of demultiplexer & also explain its operation?.

A Demultiplexer (DEMUX) is a combinational device having multiple data outputs, select inputs & only one input. The select input lines decide the state of output. Hence, a multiplexer can take only data bits and put them, one at a time, on a multiple output data lines in a particular sequence. Hence also called as digitally controlled multiple pole single way switch. i.e output line can be attached with only one input data line at the same time only. The Demultiplexer is also used for transforming *serial* data to *parallel* data. The relation between select inputs (i.e. 'm') & outputs (i.e. 'n') can be stated as, $2^m=n$. The E input is called as Probe/Enable terminal which should active for performing required operation. Hence by this way demultiplexer circuit increases the number of wired connections & produces the proper digital code on multiple wires.
Following are the types of demultiplexers,
(i). 1:2 demultiplexer, (ii). 1:4 demultiplexer, (iii). 1:8 demultiplexer, (iv). 1:16 demultiplexer.