Digital Electronic Circuits

Prof. Goutam Saha
Department of E & EC Engineering
Indian Institute of Technology, Kharagpur

Lecture – 37
Shift Register: SIPO, PISO and Universal Shift Register

Hello everybody, this week we have started discussion on shift register. So, in the last class we have seen parallel in parallel output option for register as well as serial in and serial out ok. So, in today’s class we shall look at serial in parallel out SIPO and parallel in serial out PISO how are they used and how they look like and also universal shift register ok.

(Refer Slide Time: 00:47)

CONCEPTS COVERED

Concepts Covered:

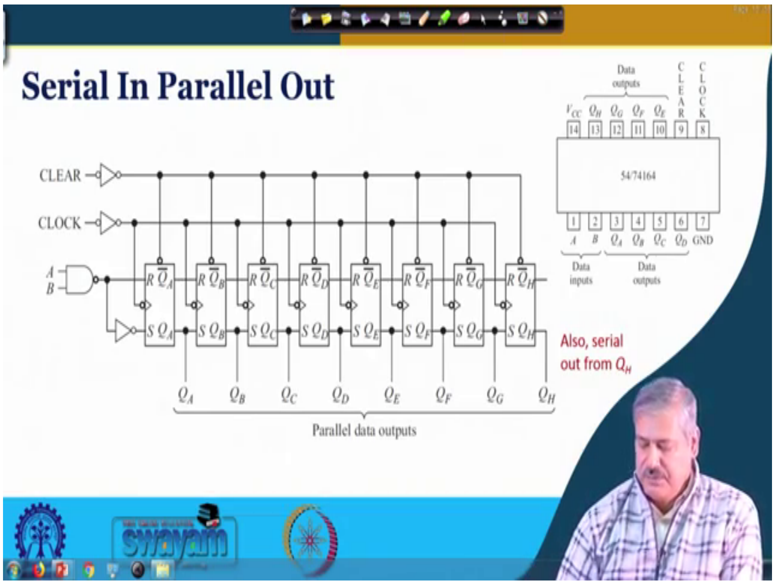
❑ Serial In Parallel Out (SIPO)

❑ Parallel In Serial Out (PISO)

❑ Universal Shift Register

So, this is what we are going to discuss today.

(Refer Slide Time: 00:49)



Serial In Parallel Out

CLEAR
CLOCK
A
B
R $\overline{Q}_A$ R $\overline{Q}_B$ R $\overline{Q}_C$ R $\overline{Q}_D$ R $\overline{Q}_E$ R $\overline{Q}_F$ R $\overline{Q}_G$ R $\overline{Q}_H$
S Q_A S Q_B S Q_C S Q_D S Q_E S Q_F S Q_G S Q_H
Q_A Q_B Q_C Q_D Q_E Q_F Q_G Q_H
Parallel data outputs
Data outputs
CLEAR
CLOCK
V_{CC} Q_H Q_G Q_F Q_E
14 13 12 11 10 9 8
54/74164
1 2 3 4 5 6 7
A B Q_A Q_B Q_C Q_D GND
Data inputs
Data outputs

Also, serial out from Q_H

So, we start with a practical circuit which is used in the IC packages IC 74164 which is a serial in parallel out shift register I mean register ok. Register is a group of flip flops which stores binary information that you have seen before and if it uses shifting option from one flip flop to another where input of one is made output of one is made as input of the other so that is shift register operation ok.

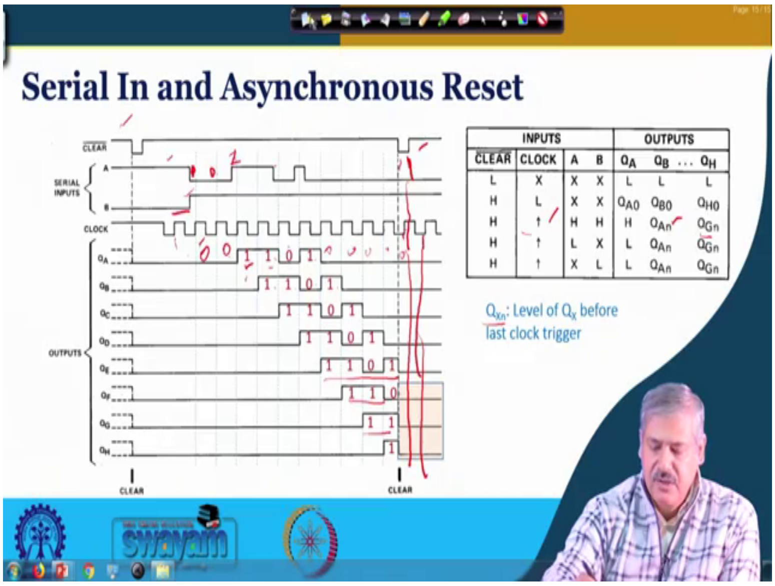
So, if we look at this circuit it is a 14 pin package ok. So, you have got 1, 2, 3, 4, 5, 6, 7, 8 so 8 flip flops. So, 8 serial in parallel out so this Q A to Q H, A, B, C, D, E, F, G, H so 8 outputs are there. So, it can you can read it as many times as you want there is nothing like destructive reading and all through where serial out reading through serial out is there and these, but the data is written serially through serial input ok. So, in one you see that these are the two inputs. So, one of them you can consider as a control input say B. So, in this is low right. So, the out this particular input over here will always be low ok.

So, S input we always get low and R is always inversion inverted version of R ok. So, that is the way we usually store the data right effectively this one is D flip flop, it is two of them are SR flip flop we do not need additional inverter is not it to convert, it to D because directly we can connect and then internally and we can get the shift register action performed right ok. So, and A is where B is 1 say the control input is 1 then, A whatever is the value of A the data that you want to put in the registers that is group of flip flops that will be passed to A and will get to the get it get to those flip flops through clock trigger and this A and B they can interchange their role one can become the control input the other can become data input and we have got asynchronous reset which is active low and common one we shall say how it works in the and also a common clock ok.

So, this is the way the circuit behaves and now you can understand that, while this is your serial in say in through one of them and you can read the data out other than parallel data output through Q H also as a serial data out because it is internally connected as a shift register one is feeding the other. So, if you keep you know triggering the clock so whatever is Q H is immediately after the Q G will become available here, after one more clock Q F will be available here, after another clock Q E will be available here ok.

So, that way also you can read all of them by triggering of the clock, but it will require more number of clocks and you already have the option of parallel reading ok, but to know that it is also possible to serially read it through Q H fine.

(Refer Slide Time: 04:45)



Serial In and Asynchronous Reset

INPUTS				OUTPUTS		
$\overline{\text{CLEAR}}$	CLOCK	A	B	Q_A	Q_B ...	Q_H
L	X	X	X	L	L	L
H	L	X	X	Q_{A0}	Q_{B0}	Q_{H0}
H	↑	H	H	H	Q_{An}	Q_{Gn}
H	↑	L	X	L	Q_{An}	Q_{Gn}
H	↑	X	L	L	Q_{An}	Q_{Gn}

Q_{Xn}: Level of Q_X before last clock trigger

So, let us see how this serial in and asynchronous reset work in this particular IC. So, we have given a example here. So, this is your clear right. So, whenever this is asynchronous reset, so whenever you are giving this clear so it become all set ok. So, clock start triggering over here and at every positive edge of the clock; every positive edge of the clock you see from the truth table, it is positive edge triggered and also you can see from the circuit that this positive edge triggered.

So, every positive edge of the clock it gets a trigger ok, but as long as you know B is 0, we consider B as a control input right then, even if A is high the that is not getting transferred to the output ok. So, the at this positive edge whatever is the value it will remain in a same so that there is no change.

When B becomes high ok, the control input becomes high then, whatever is the change in a that can go to the flip flops serially one after another. So, you can see this is one positive edge right and when after you know clock wise become control input B has become high and at the time at this edge this valuable over here is 0 for A ok, so Q A will get 0 that is what here it has got.

So, next clock trigger before this clock trigger again A is 0 so Q A will get 0 and before this clock trigger you see that value is 1 so now, after that clock trigger Q A has become 1 is it fine ok. So, similarly over here this becomes 1 and just before clock trigger you see it has become 0, so it becomes 0, before the clock trigger this is 1, after that it is all 0 so there are all 0.

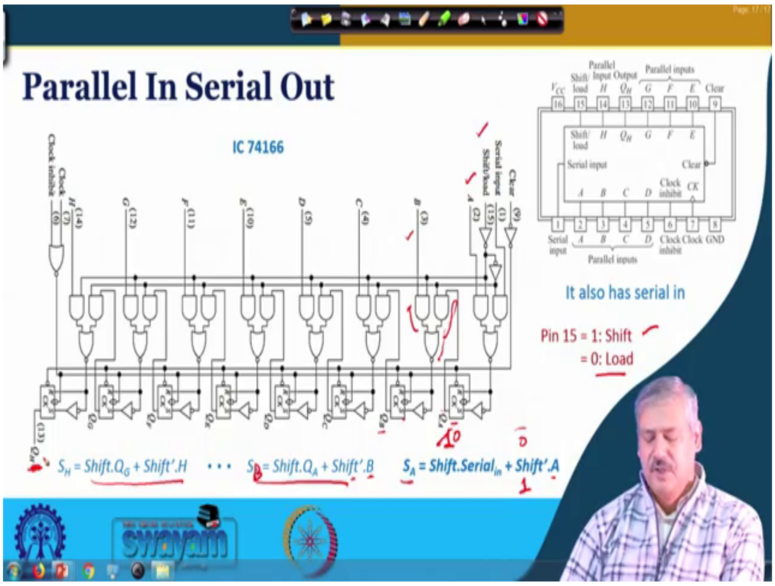
So, this is how it would look like till the next you know reset comes otherwise it is 0 it is there reset comes asynchronous reset immediately irrespective of the clock trigger, you see the clock trigger comes over here it was here again it comes here, but it become it will become immediately 0 this effect is not very pronounced, but will see for the other cases ok.

Now, what happens to Q B? So, Q A is fade as output of Q you know A is paid as input of B right. So, this 1 up to this, this is 0, 0 so 1 will come over here, 1 will come over here, 0 will come over here and 1 will come over is it ok. So, this is where it will proceed and finally, you can see that this is 1101 for Q E and after that Q F becomes 1100 like this and this becomes 1 1 and it could have been you know 0 and 1 unless this asynchronous reset was; reset was there and similarly for this one.

So, whenever this asynchronous reset occurs all of them becomes 0 after that when a clock trigger comes like this, the reset has moved so as long as it is reset this line will remain 0 ok, after the reset is moved so it is become you know inactive right it has become high so next clock trigger is here. So, at this clock trigger; at this clock trigger all the inputs are 0 right so the outputs are 0 is it.

So, this is how the serial data incomes and output you can read from Q A to Q H parallelly directly and in the truth table also it is mention like this Q xn whatever is the value so Q B takes the previous value of Q A in the with the clock trigger ok, Q H takes the previous value of Q G just before the last practical whatever the value was that is taken. So, this is how to read the truth table and corresponding timing diagram would appear in this manner and also you understand the how asynchronous reset works.

(Refer Slide Time: 09:39)



Parallel In Serial Out

IC 74166

It also has serial in

Pin 15 = 1: Shift
= 0: Load

$S_H = Shift.Q_G + Shift'.H$ ••• $S_B = Shift.Q_A + Shift'.B$ $S_A = Shift.Serial_{in} + Shift'.A$

Now, let us look at the other variety parallel in serial out ok. So, again we take up an actual circuit practical circuit which is in use which is IC 74166 ok. So, in this what you see it looks bit complicated slowly your moving into more complicated circuit. The first one is parallel in parallel out, but then you see that because it has it provides little bit of you know additional control operation. So, what is it? So, this particular block if you look at it so there is a inverter here and there is inverter over there. So, if you put them into together basically it is a SOP realization that your having ok.

So, these SOP realization, you look at it for S A so this is shift here ended with serial in, this is yours serial in do you get it so shift here ended with serial in I consider this shift forget about the lower part of it and then shift prime, so this is coming over here the inverter ended with A, that part is understood so what does it mean?

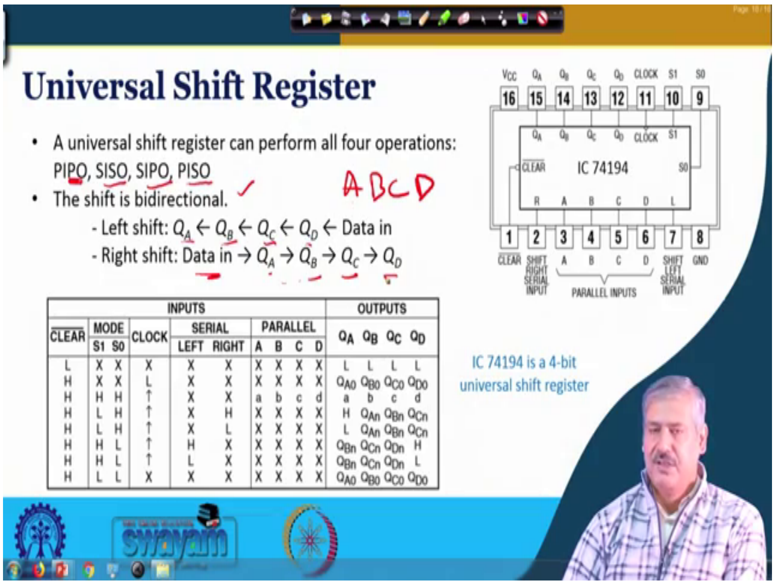
When a clock trigger comes; when a clock trigger comes at the input of the S flip flop over here; SR flip flop over here and R is just taking the inverts invert of R S if shift is equal to 1 serial in will be going as the input of S A and if shift is equal to 0 so 0 bar so, this is 1, so A will be going as input of S A is it clear this part you understand so; that means, with clock trigger with shift is equal to 1, serial in we come here and clock is equal to 0 the value A will get loaded.

Now come to flip flop B so this is S B ok. So, what is happening at the input of it shift whenever it is 1 so, this is the shift is equal to 1 so this thing ok. So, then Q A is coming, Q A is coming to the input of S A and when shift is equal to 0, B is coming to the input of S A ok.

So, what does it mean whatever was the output of Q A becomes output of Q B after the clock trigger. So, it is getting shifted, value of a getting shifted to B ok, flip flop A value is getting shifted to flip flop B is it fine and if shift is equal to 0 then this is getting loaded by the parallel in that is there which is B. So, similarly over here you can see is it ok. So, this parallel in serial out and serial out is coming you are taking from finally, Q H ok.

So, you can put it parallel in just by simply you know loading it using this input control input to be 0 and then for reading it, since there is no parallel out available because there are 8 flip flops then you will require another 8 output pins which is not there ok. So, you have to read it from here, by triggering the clock with S h will be destructive, but if you have a feedback and another mechanism that we shall discuss before, then it will be non destructive reading is it fine understand how it works.

(Refer Slide Time: 13:59)



Universal Shift Register

- A universal shift register can perform all four operations: PIPO, SISO, SIPO, PISO
- The shift is bidirectional. ✓
  - Left shift: $Q_A \leftarrow Q_B \leftarrow Q_C \leftarrow Q_D \leftarrow$ Data in
  - Right shift: Data in $\rightarrow Q_A \rightarrow Q_B \rightarrow Q_C \rightarrow Q_D$

ABCD

INPUTS										OUTPUTS			
$\overline{\text{CLEAR}}$	MODE		CLOCK	SERIAL		PARALLEL				Q_A	Q_B	Q_C	Q_D
	S1	S0		LEFT	RIGHT	A	B	C	D				
L	X	X	X	X	X	X	X	X	X	L	L	L	L
H	X	X	L	X	X	X	X	X	X	Q_{A0}	Q_{B0}	Q_{C0}	Q_{D0}
H	H	H	↑	X	X	a	b	c	d	a	b	c	d
H	L	H	↑	X	H	X	X	X	X	H	Q_{An}	Q_{Bn}	Q_{Cn}
H	L	H	↑	X	L	X	X	X	X	L	Q_{An}	Q_{Bn}	Q_{Cn}
H	H	L	↑	H	X	X	X	X	X	Q_{Bn}	Q_{Cn}	Q_{Dn}	H
H	H	L	↑	L	X	X	X	X	X	Q_{Bn}	Q_{Cn}	Q_{Dn}	L
H	L	L	X	X	X	X	X	X	X	Q_{A0}	Q_{B0}	Q_{C0}	Q_{D0}

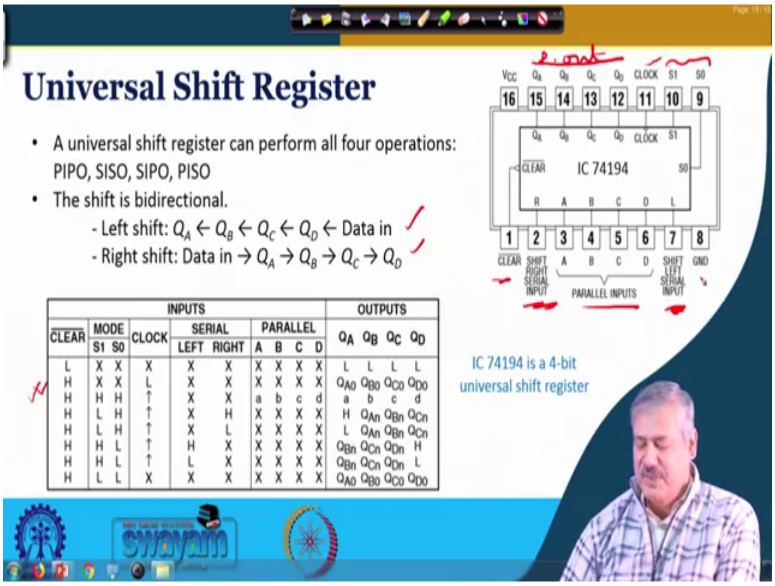
IC 74194 is a 4-bit universal shift register

Now, we look at what is called universal shift register. So, we had seen in a 4 varieties parallel in parallel out ok. So, this is the one then serial in serial out in last class and in today's class serial in parallel out and parallel in serial out ok. So, in the universal shift register, you have all these options available right and of course, when you make all these options available the sacrifies will be the number of bits you can handle within a particular package ok.

So, that is another part of it, but if you require a versatile device to do which can perform any different operations shift right, shift left and all those things yes, in this connection you need to remember the universal shift register offers both left shift and right shift ok, so that means, data can have a left shift; that means, if it is A, B, C, D this is the way the flip flops are organized. So, data can come to Q D, then Q C, Q B and Q A this is way it can be shifted with clock or it can have right shift where data is coming to Q A this and then Q A, goes to Q B, then Q C and Q D is it ok.

So, this is the bidirectional shift that is possible in universal shift register. So, let us look at again a practical circuit which will help us in understanding how it works and we shall look at inside circuit as well. So, you can see this is IC 741919194 right. It has got S1 and S0 two control inputs right not one earlier shift load there is only one control input. So, this two control inputs between them offers 4 possible you know options 0001 10 and 1 1 ok. So, for 0 0 we expect sudden operation 0 1 something else 1 0 and 1 1 right and this is depicted in this truth table we shall have more compact and easy to understand version of it in the next slide which I shall will discuss shortly ok.

(Refer Slide Time: 16:35)



Universal Shift Register

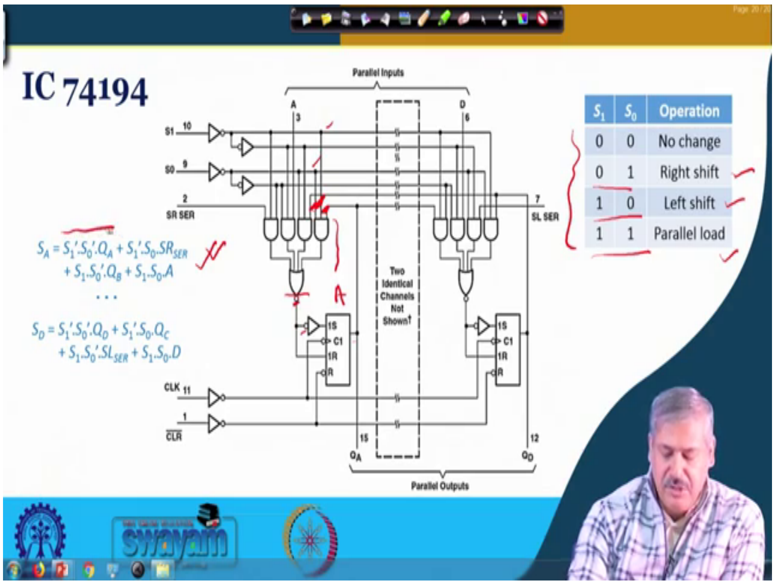
- A universal shift register can perform all four operations: PIPO, SISO, SIPO, PISO
- The shift is bidirectional.
  - Left shift: $Q_A \leftarrow Q_B \leftarrow Q_C \leftarrow Q_D \leftarrow$ Data in
  - Right shift: Data in $\rightarrow Q_A \rightarrow Q_B \rightarrow Q_C \rightarrow Q_D$

INPUTS										OUTPUTS			
$\overline{\text{CLEAR}}$	MODE		CLOCK	SERIAL		PARALLEL				Q_A	Q_B	Q_C	Q_D
	S1	S0		LEFT	RIGHT	A	B	C	D				
L	X	X	X	X	X	X	X	X	X	L	L	L	L
H	X	X	L	X	X	X	X	X	X	Q_{A0}	Q_{B0}	Q_{C0}	Q_{D0}
H	H	H	↑	X	X	a	b	c	d	a	b	c	d
H	L	H	↑	X	H	X	X	X	X	H	Q_{An}	Q_{Bn}	Q_{Cn}
H	L	H	↑	X	L	X	X	X	X	L	Q_{An}	Q_{Bn}	Q_{Cn}
H	H	L	↑	H	X	X	X	X	X	Q_{Bn}	Q_{Cn}	Q_{Dn}	H
H	H	L	↑	L	X	X	X	X	X	Q_{Bn}	Q_{Cn}	Q_{Dn}	L
H	L	L	X	X	X	X	X	X	X	Q_{A0}	Q_{B0}	Q_{C0}	Q_{D0}

IC 74194 is a 4-bit universal shift register

So, what else is there? So, parallel in and parallel out, you can see this is parallel out because that option also need to be made available it is universal shift register all possible things, then shift left serial input and shift right serial input both the things need to be that can have bidirectional shift and other than that you have got clock and a synchronous reset active look clear and of course, VCCN ground and it is a 16 pin package is it right.

(Refer Slide Time: 17:13)



IC 74194

$$S_A = S_1'.S_0'.Q_A + S_1'.S_0.SR_{SER} + S_1.S_0'.Q_B + S_1.S_0.A$$

. . .

$$S_D = S_1'.S_0'.Q_D + S_1'.S_0.Q_C + S_1.S_0'.SL_{SER} + S_1.S_0.D$$

S_1	S_0	Operation
0	0	No change
0	1	Right shift
1	0	Left shift
1	1	Parallel load

So, let us see the circuit and how the truth table looks like in a compact manner. So, S1 S naught being 00 then you no change S1 no change means whatever is the value it will be the same right. If there is a it is 01 it will be right shift, 1 0 this is left shift and 11 parallel load because parallel in that is possible and parallel out is always there because the (Refer Time: 17:45) Q A, Q B, Q C, Q D outputs are there.

So, this is the way we control inputs play into the circuit. So, we look at this circuit not all 4 bits are shown to ones are shown, but from that you can understand how it works. So, first of all again the way we have done it for another shift register another IC. So, if you look at this output. So, this is your now that is there is a bubble there is a bubble so effectively it is a AND or SOP you know realization that you see here and if you try to write it in the form of Boolean equation for A so, this is your Q A so this is flip flop A right.

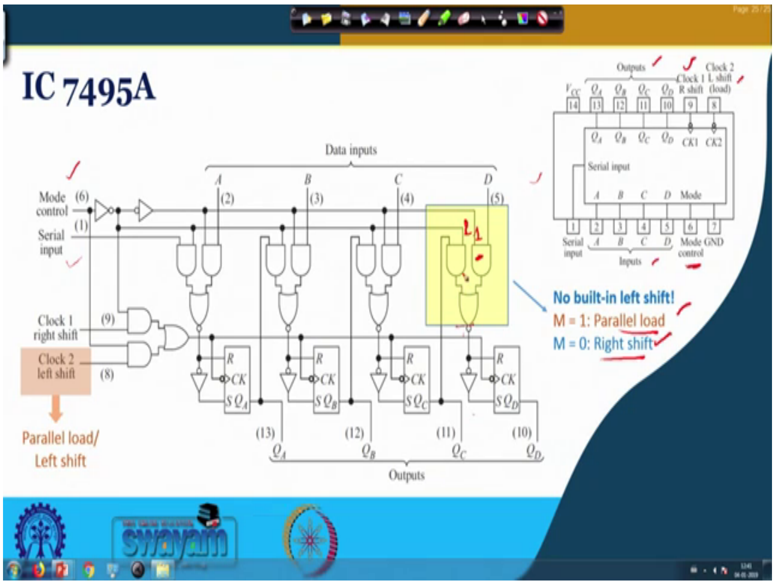
So, this is how it would look like right. So, if you look at this way. So, S1 prime S naught prime so, S1 is here. So, S1 prime is this one this line is S1 prime and S naught prime ok. So, this line is S naught prime and this is your Q A, so this is S1 S naught prime, this is a S1 prime and this is your Q A these are the 3 inputs ok. So, that is what you see as the AND output which is odd with rest of the inputs ok. Similarly S1 prime and S naught you can see, this is your S1 prime you can see you know dropdown from here and S naught there are two inverters over here.

So, this is your S naught and that is ended with serial in shift right serial in and then S1 is 1 and S naught is 0 ok; S naught is 0. So, this is the case it happens S1 is 1 here, S naught is 0 this is the case this two points that is coming downward over here so, this is this AND gate we are talking about ok.

So, in this the input is coming from the previous stage. So, that is Q A so, it is previous stage is Q B so it is coming from Q B and finally, here S1 and S0 both are 1. So, this is the case this is the AND gate so in this case A is coming over here is it clear right. So, if this is for happening for S A for S D you know the other case you can see the first one is self loading I mean this its previous value is loaded so that is there no change case. So, S1 and S0 both being 00, the second case a shift right.

So, earlier it was serial in this case it will be Q C it is coming from you know it is right shift. So, Q C will be the input of this one right and for S1 being 1 and S naught being 0 right this was Q B here so, the it is a left shift operation from Q B it was coming to Q A, now there is no other flip flop after this so, basically then you have to consider serial data in which is for left shift that was there in the IC one of the input ok. So, that is what is connected here and finally, when both of them are 1, D is directly getting loaded parallel load is there is it fine ok. So, this is replicated in the rest of the cases with the appropriate values of here other things are similar right and this is asynchronous clear and you know and clock is common to all of them fine.

(Refer Slide Time: 22:03)



IC 7495A

No built-in left shift!

M = 1: Parallel load

M = 0: Right shift

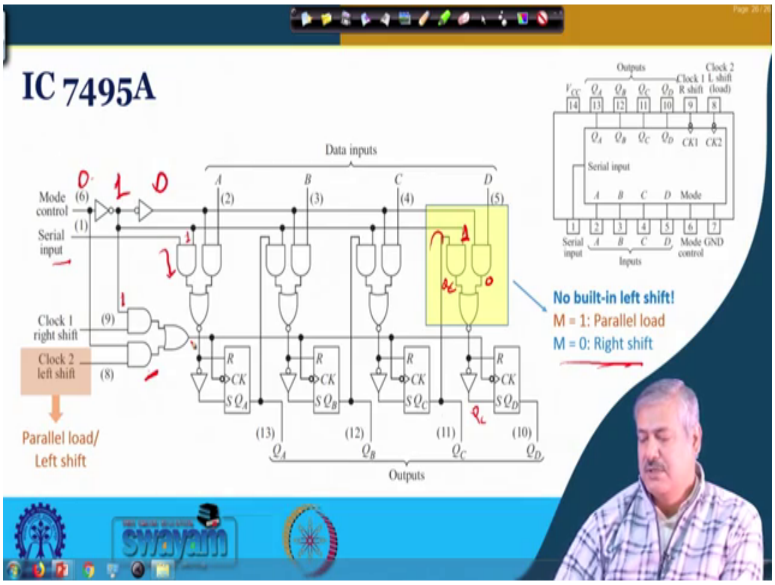
So, now we understand how this universal shift register works, but 1 IC that often are used in the lab is IC 7495 was 7495 a and we it is not a universal shift register if you look at the built-in things that are available un like the previous one that we just discussed ok, but we can make it work like a universal shift register by external connection since it is often used in the lab and so I have taken a consider this as a topic of the discussion so let us see how it actually looks like and so this is a 14 pin IC ok.

So, 4 bit in you know information can be stored. So, A, B, C, D is the parallel in and Q A, Q B, Q C, the Q D are the parallel output that you can see and here you have got 1 control input remote control ok. So, when mode is equal to 1 it is parallel load and when mode is equal to 0 it is right shift it is right shift ok.

So, this is the way it has been configured and you can see one two clocks are there one clock is for right shift that is there, the other clock over here you see it is written left shift within bracket load. So, the other clock is actually used for this parallel load which can be also used for left shift operation. So, how that left shift which is not built in we shall see how to get it done ok. So, this is basically what is there and now we look at the circuit ok.

So, this is again made through a SR flip flop and so this is the remote control right, this is serial input and if you look at just one of them over here. So, when mode is equal to 1 ok. So, 1 comes here. So, this is your 1 ok, this is 1, this is 0 so another inverter, so this is 1. So, this split this AND gate is now active they are AND gate input will be 0. So, this will be 0. So, this is again a SOP kind of thing because of two bubbles two NOT gates are there. So, it is AND or SOP realization. So, at the time this is 1 right and D, so D comes as the input.

(Refer Slide Time: 24:55)



IC 7495A

No built-in left shift!

M = 1: Parallel load

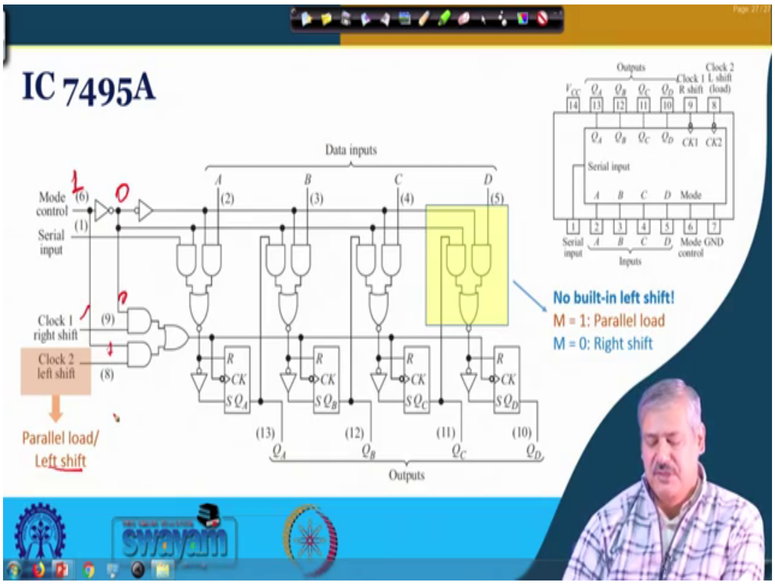
M = 0: Right shift

Parallel load/ Left shift

So, it is parallel load that is happening and for the other case when your mode is equal to 0, mode is equal to 0 ok. So you can see that this is your 1 and this is your 0 so, this AND gate output will be 0 and this is this AND gate, the other AND gate it is taken from here this inverter output. So, this is your 1 and then at that time Q C is coming over here.

So, this is Q C right. So, Q C will be over here so Q C output will go to the Q D. So, basically it is a Q C is going to Q D. So, similarly if you look at the others Q B will go to Q C, Q A to Q B and for Q A there will be serial input in. So, this is the serial input in for mode is equal to 0 ok. So, this becomes 1 is it right. So, that is your right shift operation and at that time when mode is equal to 0 right mode is equal to 0 you can see that this clock is deformed ok; this clock is deformed right at that time this is 1, so this clock is right shift clock is functional ok, so right shift clock is functional.

(Refer Slide Time: 26:11)


IC 7495A

No built-in left shift!

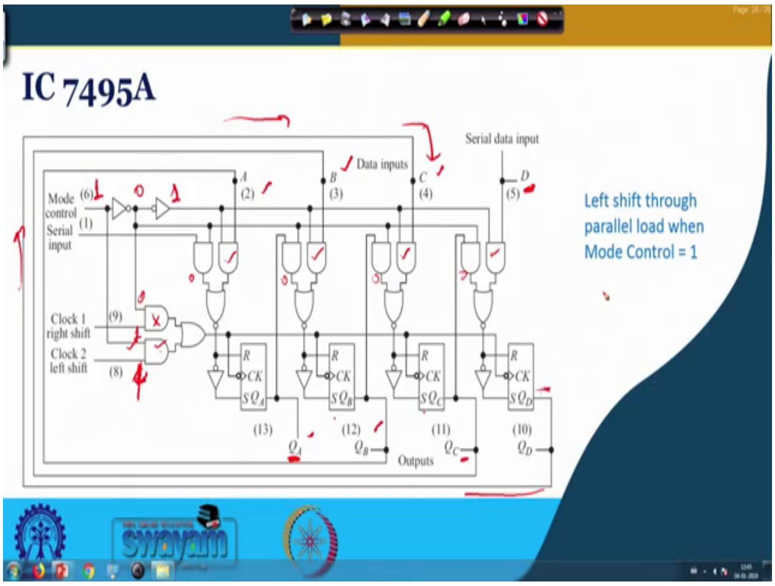
M = 1: Parallel load

M = 0: Right shift

Parallel load/
Left shift

And when mode is equal to 1 what is equal to 1. So, this is 0; so this is 0. So, this right shift clock is nonfunctional ok. So, at that in this is 1 so this so called left shift or parallel load clock. So, right now we have seen parallel load only do not worry about the left shift part of it ok. So, it is just doing parallel load through this clock at the trigger of this clock is it.

(Refer Slide Time: 26:43)


IC 7495A

Left shift through parallel load when Mode Control = 1

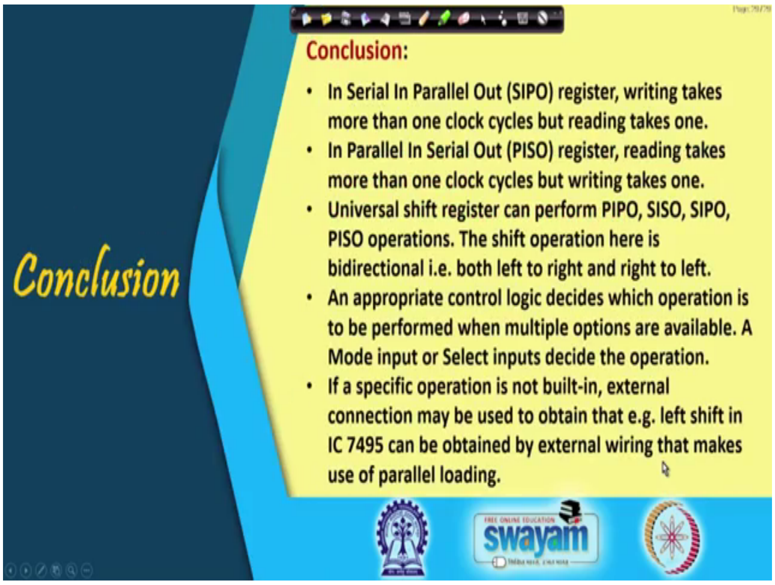
Now, let us see this popular IC 7495 how it can be used for left shift also ok. So, for this we need external wiring ok. So, you see how the external wiring is done. So, you see this thing, this is external wire. So, Q D; Q D is connected this wire is going here, it is coming here and then over here as input to C. So, instead of C, so C is the output parallel load you know this is input parallel load input was their C Q C.

So, Q D output is now connected as at the input of C externally you connect this wire ok. Similarly Q C is connected to B input. So, parallel load input is there for the flip flop so B. So, Q B is connected to A right and Q A is there as the final output ok. So, serial data in will be coming through D and serial data output will be through Q A ok.

So, how it is happening? So, the mode control part we have already seen. So, basically when it is 1; when it is 1 right so, then this flip flop this AND gate, this AND gate ok, this AND gate and this AND gate. So, they are acting and the other AND gates these outputs are all 0 this outputs are all 0 because it is coming from here and this is your 1 ok. So, at that time at this particular clock so left shift clock so which is coming here this is 1 and this is the clock we are talking about so, this is 1, so this is the clock and the other case this is 0 so, this is not working. So, this is this AND gate output is always 0. So OR gate output will be taking this AND gate output.

So, this clock will be the 1 in consideration. So, at that time you are having your this Q D getting loaded through C to flip flop C. Q C is getting loaded through B to flip flop B and Q B to flip flop A and Q S is available as the final output ok. So, this is the way the left shift operations takes place ok. So, it is not built in, but it works in that manner ok.

(Refer Slide Time: 29:33)



Conclusion

Conclusion:

- In Serial In Parallel Out (SIPO) register, writing takes more than one clock cycles but reading takes one.
- In Parallel In Serial Out (PISO) register, reading takes more than one clock cycles but writing takes one.
- Universal shift register can perform PIPO, SISO, SIPO, PISO operations. The shift operation here is bidirectional i.e. both left to right and right to left.
- An appropriate control logic decides which operation is to be performed when multiple options are available. A Mode input or Select inputs decide the operation.
- If a specific operation is not built-in, external connection may be used to obtain that e.g. left shift in IC 7495 can be obtained by external wiring that makes use of parallel loading.

So, we got an idea of the register and you know the parallel in parallel out and different kinds you know shift registers that is a SIPO, PISO and SISO in today's class as well as the previous class as well as what is in the making of universal shift register. So, last week we saw flip flops as individuals and in this two classes we have seen them as a group storing binary information and we have seen how to read and how to write for this different configuration and with this we move to next class we shall see the applications of shift register other applications of shift register.

Thank you.