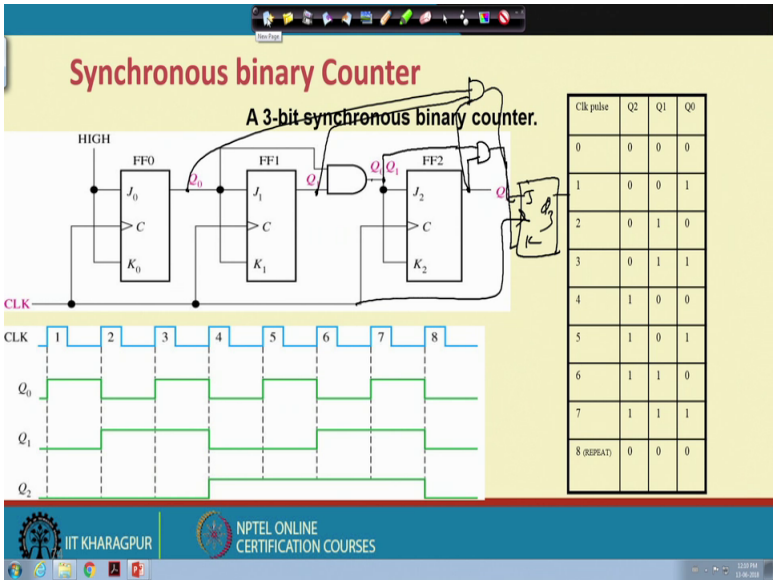
Digital Circuits
Prof. Santanu Chattopadhyay
Department of Electronics and Electrical Communication Engineering
Indian Institute of Technology, Kharagpur

Lecture - 34
Sequential Circuits (Contd.)

Problem with synchronous counters is that many times, you need more amount of logic.

(Refer Slide Time: 00:20)



For example, say this one here we have got a 3 bit counter, 3 bit synchronous counter, but this 3 bit synchronous counter. So, this flip flop 0, flip flop 1 and flip flop 2. So, this flip flop 2 is the most significant bit flip flop 1 is the next bit and flip flop 0 is the least significant bit.

Now, you see that in the connection pattern. So, what we need is that, every clock pulse this flip flop 0 should toggle ok. So, this should toggle flip flop one should toggle if flip flop 0 was equal to 1 like if we look into this counting sequence. So, this whenever this flip flop 0 it toggles at every clock fine flip flop 1 Q 1 it changes the state on only when Q 0 was equal to 1 at the previous time.

You see whenever this is equal to 1. So, it makes a transition similarly this is equal to 1, it makes a transition, but if Q 0 is equal to 0, in that case, it is not making any transition like here also, it is not making any transition. So, it is coming from one to one only and

as far as Q 2 is concerned. So, Q 2 should make a transition when Q 0 and Q 1 both are equal to 1. So, here Q 0 Q 1 both are equal to 1. So, at the next time, it makes a transition to one and they become 0 that is the different thing.

So, while drawing the corresponding logic circuit. So, what is required is that for this J and for the flip flop 0. So, they will it will transit at every clock pulse. So, this is tied high directly. Now for this flip flop 1 this J and K are shorted and it is connected to this Q 0 line. So, whenever Q 0 equal to 1 at the next clock, this flip flop 1 will transit then this flip flop 2, it is connected again J and K are shorted, but it is connected via a an AND gate where this Q 0 and Q 1, they are anded and then this is connected to flip flop 2.

So, this flip flop 2 will transit when Q 0 and Q 1 both the outputs are equal to 1. So, this way you can have a 3 bit synchronous counter. Now if you have more number of stages like if you have trying to have a 4 bit synchronous counter, then you see that this you will be needing another flip flop here ok. So, you will be needing another flip flop here with J and K the clock signal will be common clock signal there is no problem.

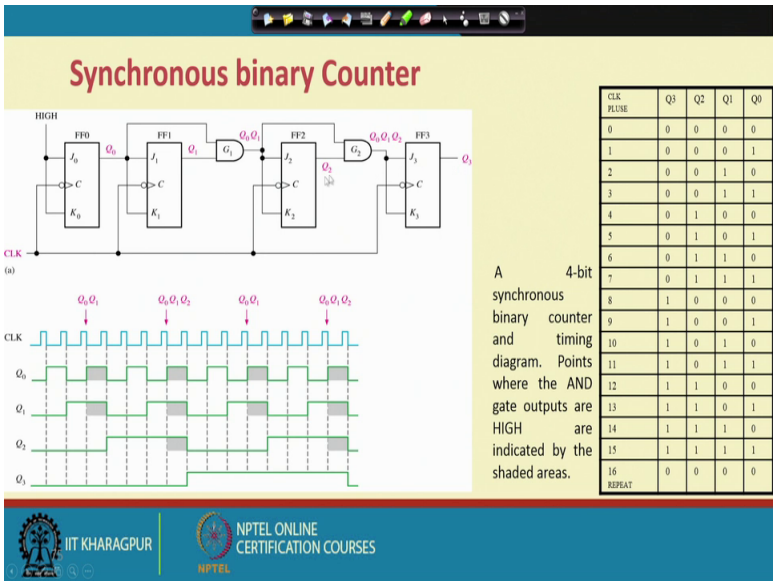
Now this J and K are to be tied together and then that value should come from another and gate. Now this AND gate should have input from this line ok, it should have input from this line and also. So, this Q Q 3, if you say that this line is say Q 3, then Q 3 will transit only when all this Q 0 Q 1 and Q 2 all of them are 1 ok. So, that is this line should be 1 and this Q 0, Q 1. So, this line should also be equal to 1. So, you can make a connection like this and then you will be getting this thing.

So, this way, I can have another stage of a counter we can get a 4 bit synchronous counter, but the problem that you can visualize is that you see the farther stage is that I am connecting. So, the and gate is a forming a chain of and gates. So, the first this flip flops says the sees the delay of 1 and gate, but this flip flop sees the delay of these 2 and gates 2 get the correct value.

So, if you connect another state will see delay of 3 and gates in the process or you use and gates of higher number of input. So, one another possible way of connecting, it is that you do this thing and then you take this output this Q this Q output, this Q output and this Q output and connect it together and then feed it here. So, that may be other possibility, but thing that case the AND gate is becoming more complex.

So, you are recording AND gate of higher number of inputs. So, that is the other difficulty. So, this way this synchronous design, they have got this type of problem the circuit becomes complex anyway. So, many times depending upon the situation, we choose between synchronous counter asynchronous counter.

(Refer Slide Time: 04:42)

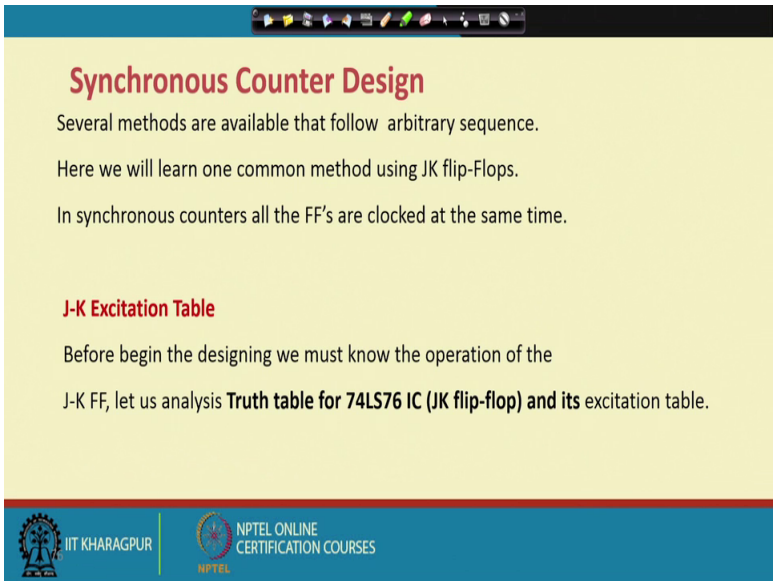


CLK PLUSE	Q3	Q2	Q1	Q0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
16 REPEAT	0	0	0	0

So, here is actually it is shown that is we have got this 4 bit synchronous counter and this G 1 is feeding this flip flop 2 by anding Q 0, Q 1, then this G 2 is feeding flip flop 3 by anding of this Q 0, Q 1 with Q 2.

And then a 4 bit synchronous binary counter and timing diagram have been shown. So, the when the AND gate outputs are equal to high. So, they are shown by these shaded portions. So, essentially, it will got through this sequence starting with all 0 going up to all one and then from 16 cycle, it will be repeating. So, it will be again starting with all 0 and continue like this. So, that is a synchronous binary counter type structure that we have.

(Refer Slide Time: 05:32)

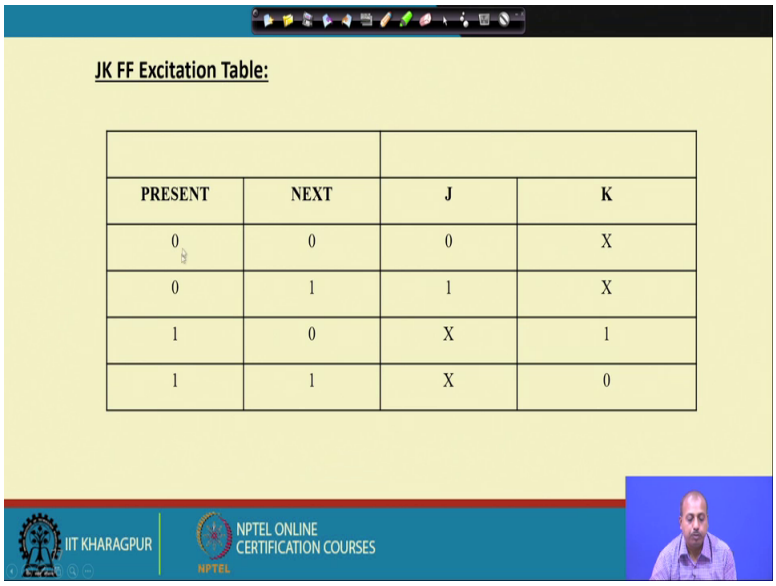


So, how do we design a synchronous counter? So, the example that I have shown you so, it seems that all on a sudden. So, we could land into this particular design, but how do we come to that. So, for that purpose there are several methods that are available for that that can follow arbitrary sequence of counters. So, we will be we will first; we will see some J K flip flop based implementation and since, it is synchronous design. So, all flip flops are clocked at the same times. So, flip flop clocking is we do not have any opportunity to manipulate the clock signal.

So, clock signal is common and it is fed from the same source for all the flip flops. So, all you can do is that you can control this J and K inputs for this flip flops to get your desired sequence.

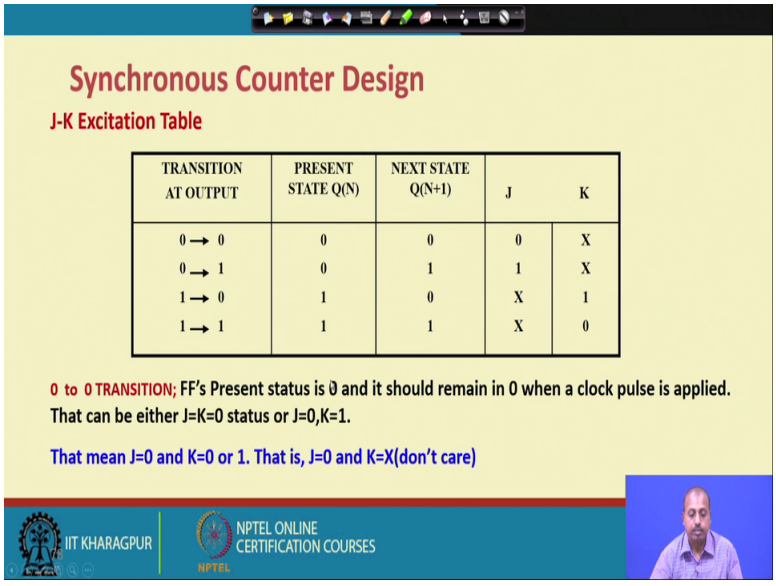
So, the J K excitation table; so, it is known to us just to recapitulate.

(Refer Slide Time: 06:24)



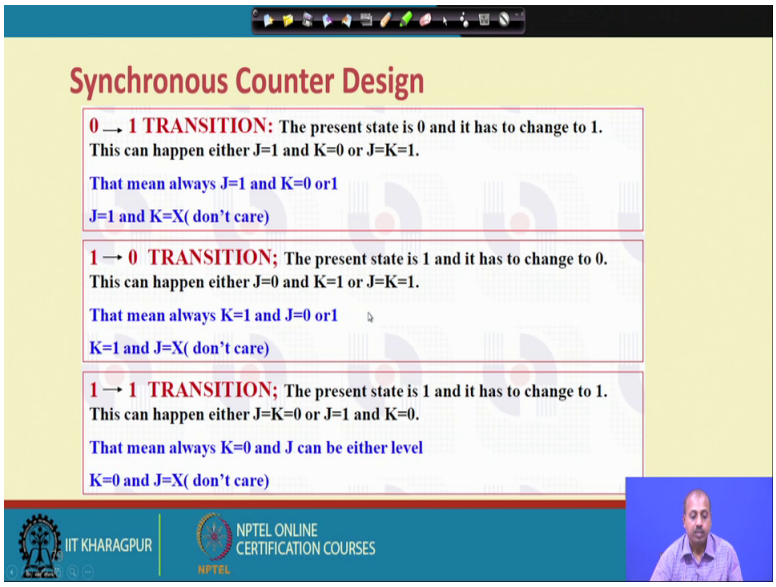
So, if the present value is 0 and the next value is also 0, so, what we have to do is this J should be 0 and K can be do not care. Similarly from 0 to 1, if the present value of this flip flop Q output is 0 and the next output we expect to be 1. So, for that I need to give J equal to 1 and K can be do not care. Similarly, from 1 to 0, K should be equal to 1 and J is do not care and for 1 1, I should this J K value must be equal to 0 and this J value can be a do not care. So, this is the J K excitation table. So, that is known to us.

(Refer Slide Time: 07:03)



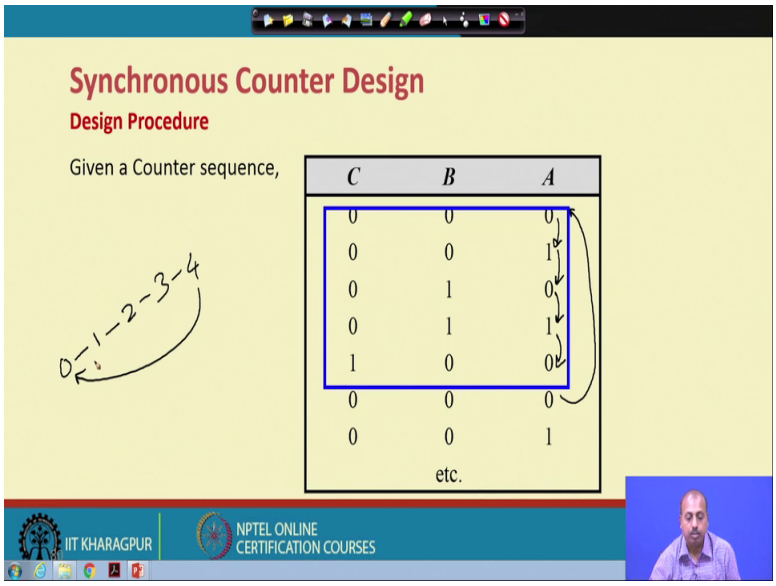
Now, so, for this is again for J K excitation table from 0 to 0; so, present state is 0 and the next state is 0 and J is 0. So, we already explained. So, 0 to 0 transition flip flop present status is 0 and it should remain 0 when a clock pulse is applied that can be either J equal to K equal to 0 or J equal to 0 K equal to 1. So, it is J equal to 0 K equal to 0 or 1. So, that is J equal to 0 and K equal to do not care. So, that explains this table you have already seen that.

(Refer Slide Time: 07:37)



Now, for synchronous counter design; so, you have to see when these transitions are occurring like when this 0 to 1 transition is occurring present state is 0 and it has to change to 1 with J equal to 1, K equal to 0 or J equal to 1, K equal to 1. Similarly from 1 to 0 transition, I can get by K equal to 1 and J equal to 0 or 1 and one to one transition K must be equal to 0 and J can be either of the value 0 or 1.

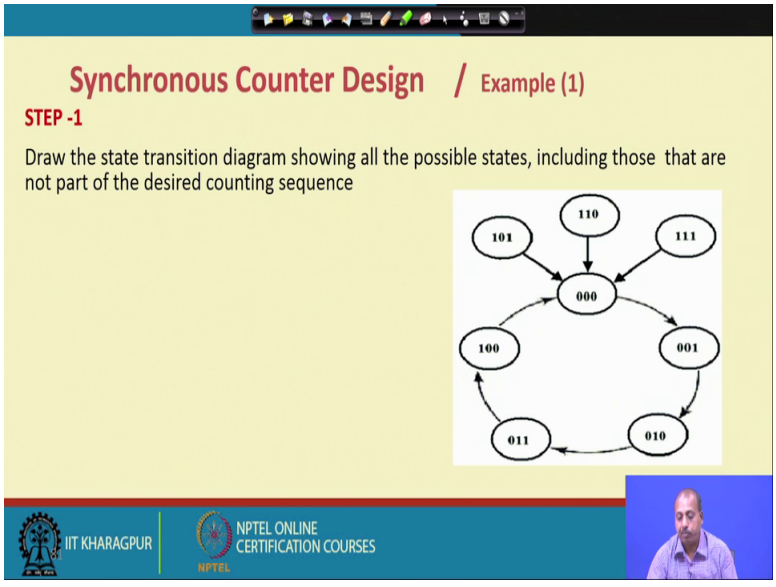
(Refer Slide Time: 08:06)



So, suppose we are asked to design a counter which goes like this. So, this is the sequence that is given to us that is the it says that it is a 3 bit counter, it will start at 0 0 0 and the next time it will go to 0 0 1, then it will go to 0 1 0, then 0 1 1, then 1 0 0 and after that it goes back to again 0 0 0, suppose, this is the sequence that I need to generate. So, in terms of decimal value; so, the sequence that I need is 0, 1, 2, 3 4 and then again 0 ok.

So, how do we design such a counter? So, that is what will try to see.

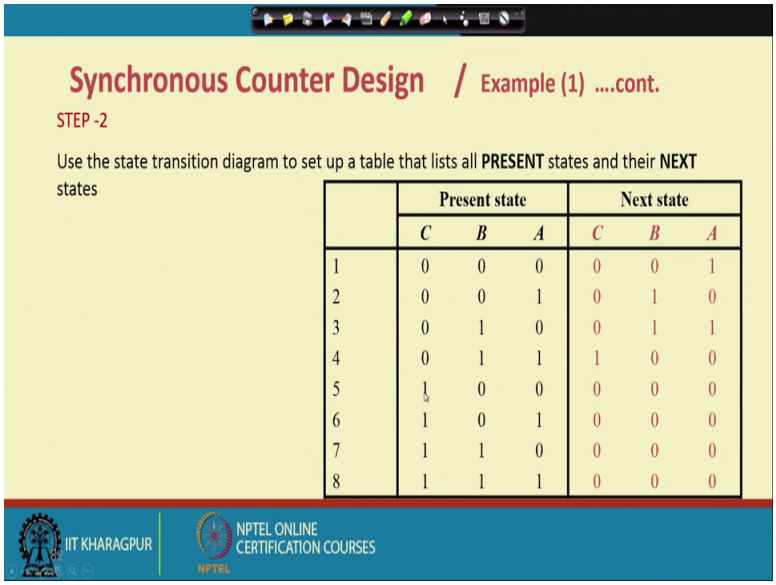
(Refer Slide Time: 08:54)



So, the first point first step for designing such a counter is to draw something called the state transition diagram for the counter. So, it is like this the initially. So, this counter value is 0 0 0, then it goes to 0 0 1 and the next clock pulse, then 0 1 0, 0 1 1, 1 0 0 and going back to 0 0 0, it does not tell anything about what are you going to do for the other states. So, may be ah. So, if you if you assume like this that if you start at any other state if if, then if the by chance the counter reaches any of these states 1 1 1, 0 1 0, 1 1 0, 1 1 1 0 or 1 1 1, then at the next clock pulse. So, it will come back to this state all 0.

So, it will be starting from because that is the. So, this is the, this is actually the drawing the state transition diagram showing all the possible states including those that are not part of the desired counting sequence.

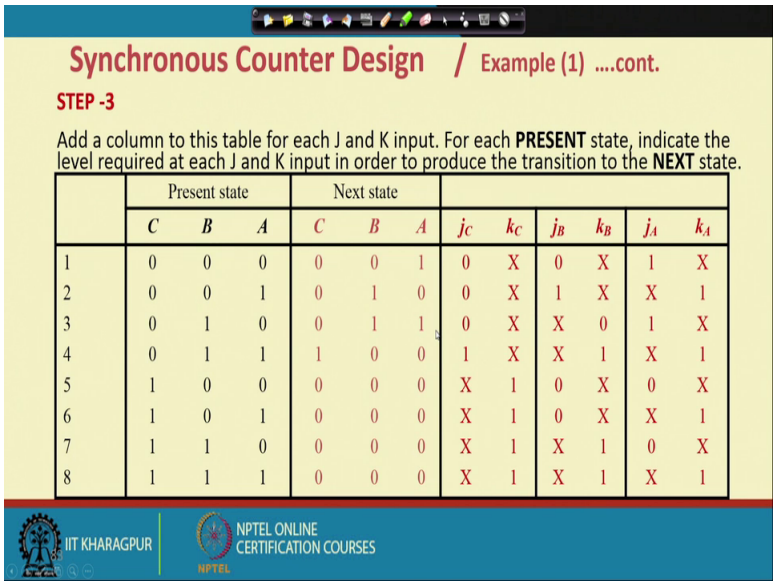
(Refer Slide Time: 09:50)



So, once we are done we are done with this. So, the next step is to use the state transition diagram to set up a table to list all present and next states. So, you see from 0 0 0, if this is the current state then from this diagram, we know the next state should be 0 0 1. So, that is written here from 0 0 0 to 0 0 1 from 0 0 1 to 0 1 0. So, this is sorry; this is coming directly from this table from this state transition diagram particular. So, when we reach this 1 0 0, then the next state is 0 0 0.

So, from 1 0 0, the next state is 0 0 0 and if from other value like 1 0 1, the next state should be 0 0 0 for 1 1 0, next state should be 0 0 0. So, like that. So, this is actually capturing these 3 state transitions. So, 1 0 1, 1 1 0, 1 1 1 to 0 0 0.

(Refer Slide Time: 10:47)

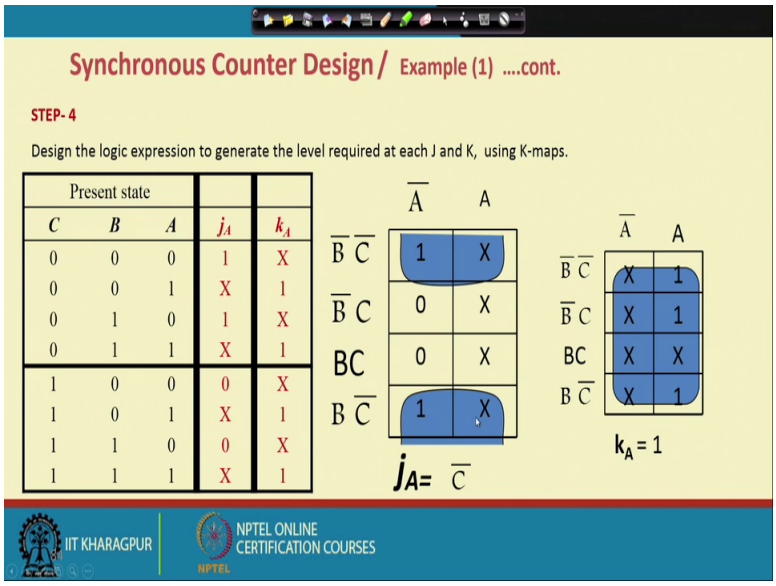


	Present state			Next state								
	C	B	A	C	B	A	j_C	k_C	j_B	k_B	j_A	k_A
1	0	0	0	0	0	1	0	X	0	X	1	X
2	0	0	1	0	1	0	0	X	1	X	X	1
3	0	1	0	0	1	1	0	X	X	0	1	X
4	0	1	1	1	0	0	1	X	X	1	X	1
5	1	0	0	0	0	0	X	1	0	X	0	X
6	1	0	1	0	0	0	X	1	0	X	X	1
7	1	1	0	0	0	0	X	1	X	1	0	X
8	1	1	1	0	0	0	X	1	X	1	X	1

Now, so, that table is drawn and since there will be realized by means of J K flip flop. So, each of these bits A, B and C, they will be realized by J K flip flop. So, we try to see what should be the corresponding values of J and K. So, see for this for this one; so, 0 to 0; so, this J bit must be 0 K bit can be do not care from the JK excitation table. So, we know that similarly from say 0 this also 0 to 0. So, this J bit is 0 and K bit is do not care.

So, at this point at this level. So, we have got 0 to 1 transition for that J bit equal to 1 and K bit is do not care, similarly, if you take any other points say the this at the transition the a bit. So, 0 to 1. So, for that this J A that is a input of the J input of the a flip flop that should be equal to 1 and K input can be do not care. So, this way, we fill up this part. So, this present state next state. So, this is coming from the counting sequence and once the we know that once we know the counting sequence. So, you can derive what should be the control values that should be fed to the J and K inputs of individual flip flops to get the appropriate state transition for the flip flops.

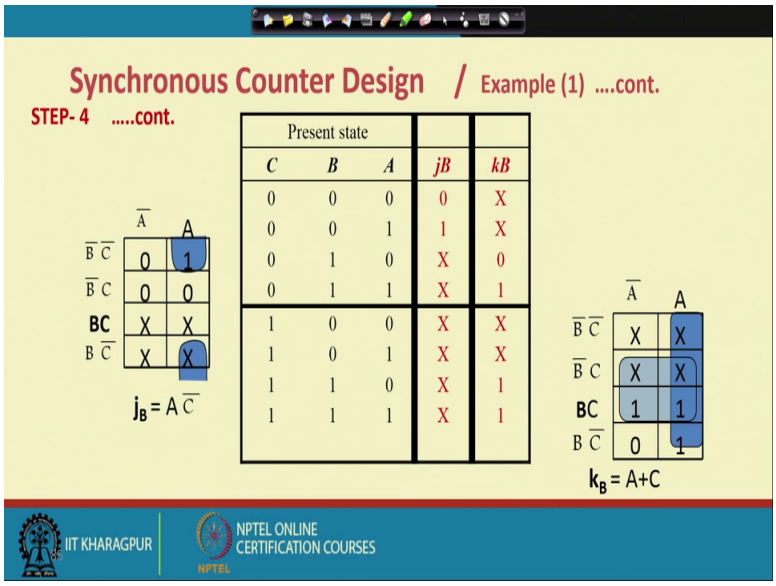
(Refer Slide Time: 12:11)



After that the step is you have to make the corresponding combinational logic. So, if the current state is 0 0 0, then in 1 that J A should be one and K A should be do not care. So, that way you write down the truth table ok. So,. So, 0 0 0 to 1 1 1 and then what is the corresponding expected value of J A and K A so, that we can write down and then you put it into a Karnaugh map do minimization and we see that the function that we getting for J A is C bar and K A is always equal to 1 ok.

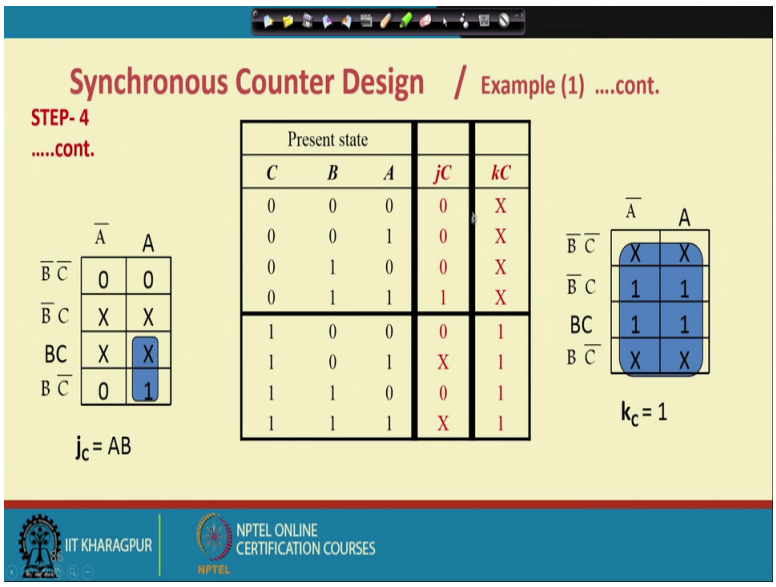
So, J A equal to C bar and K A equal to 1 that is the truth table that is that is the you logic function for the a bit.

(Refer Slide Time: 13:01)



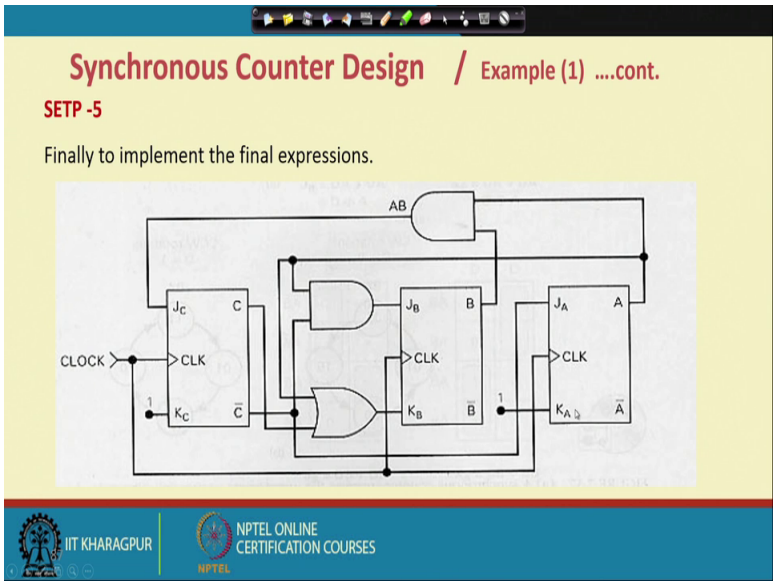
Similarly, for the B bit, so, J B and K B are to be derived. So, here also we do the same thing that from that previous table, we find out from actually from this big table, you just take out this present state this 3 bits and this J B K B columns. So, that way, we just we are just rewriting it here sorry, here J B K B column and then you do a logic minimization by means of Karnaugh map. So, you see that the function that we are getting is J B equal to A C bar and K B equal to A plus C and the same thing; we do for the C flip flop ok.

(Refer Slide Time: 13:40)



And then we again take out the portions of present state and this J C K C columns and accordingly draw the truth table and make the Karnaugh map and do minimization and J C turns out to be A B and K C turns out to be one once this is done. So, we are now ready to draw the circuit.

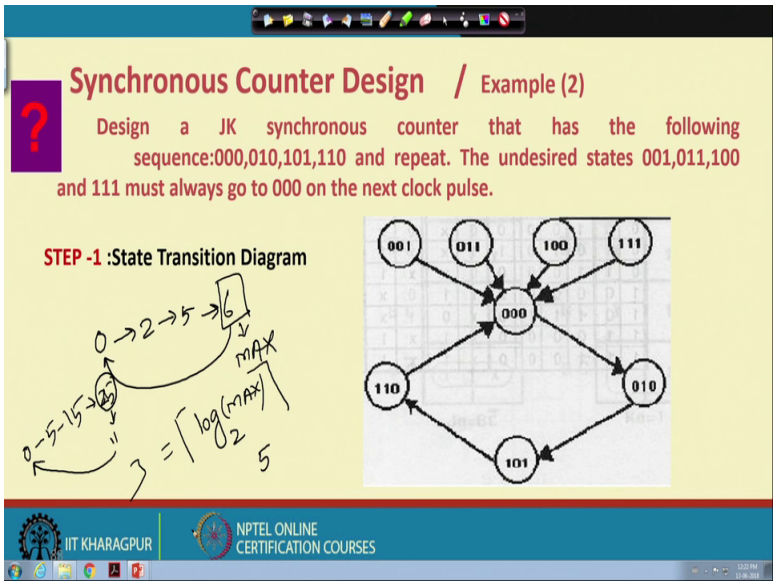
(Refer Slide Time: 14:02)



You see that this; if you look into any of them say K C should be equal to 1. So, where is K C. So, this K C is tied high, similarly, this J C should be equal to A B. So, this J C this a and B this 2 bits have been taken and they are ANDed and this A B is fed to J C. Now if you go back, see this K B equal to A plus C. Now, you see this K B. So, we are ORing this C output and A output. So, these 2 flip flop output are ORed and it is fed K B. So, this circuit that we are getting so, you see the clock is common to all the 3 flip flops and they are coming from the same source. So, this is the synchronous design and. So, if you start it with some initial value ok. So, it will make transitions accordingly.

So, it will follow this particular sequence that we have. So, starting at any of these values; so, it will be producing this value and if you start at any of these 3 states, it will be first transiting into 0 0 0, then it will continually making transitions in this sequence. So, that way we can design some synchronous counters.

(Refer Slide Time: 15:16)



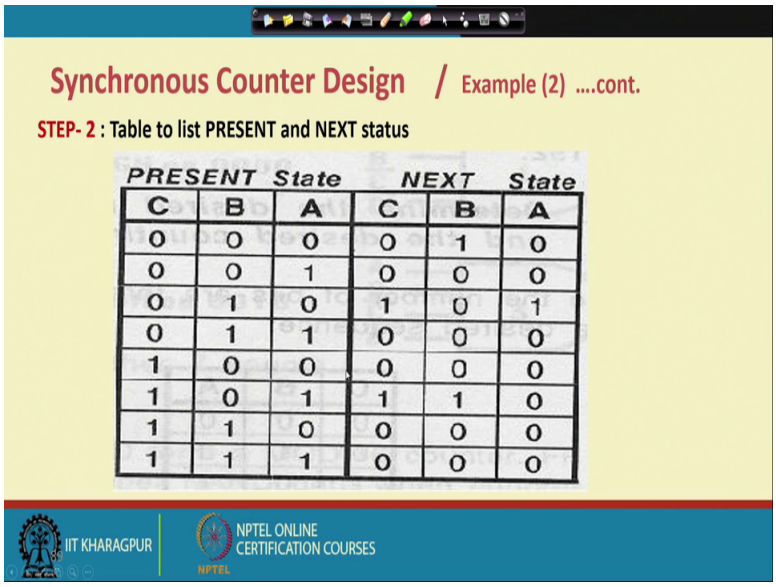
We can modify the designer bit. So, we want to design a sequence where the sequence is 0, then 2, then 5 and then 6 ok. So, the sequence in terms of decimal, what I am looking for is 0, followed by 2, followed by 5, followed by 6 and then coming back to 0. Now how do you say like how many flip flops will be needed. So, that is determined by this number the maximum count value that you have. So, this is it is 6 for representing 6 in a binary notation we need at least 3 bits. So, that is given by. So, if I say that this is the max value that the counter should produce. So, the number of flip flops needed is given by this log of this max log of this max to the base 2 the ceiling of that.

So, in this case, this max value is equal to 6. So, this turns out to be equal to 3. So, if you are designing a sequence, if I tell you that you have to design a sequence like this the 0 5, 15, 25, 11 and then 0, then for deriving how many flip flops will be needed. So, you have to look into this value 25 and this max equal to 25. So, you will need a 5 bit, you will need 5 flip flops for designing ok.

So, that way this log value of this max. So, it will tell you how many flip flops will be needed for designing the counter now in this particular case. So, we have got we have we have we have to design this particular sequence. So, 0 0 0 followed by 0 1 0, 1 0 1 and 1 1 0 and repeat the undesired states 0 0 1, 0 1 1, 1 0 0 and 1 1 1, they must always go to 0 0 0 on the next clock pulse. So, that is expected. So, it will go to 0 0 0 0 in the next clock pulse.

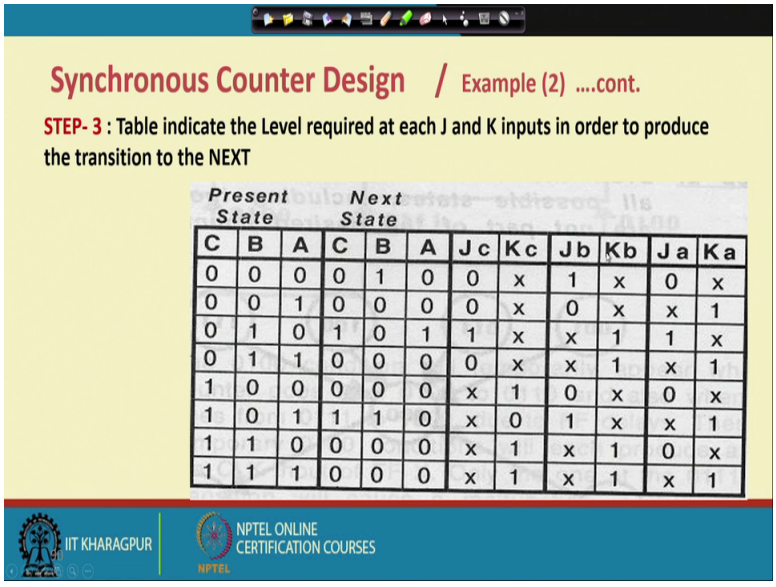
So, this state transition diagram; so, it will be like this.

(Refer Slide Time: 17:18)



And then you can just draw the present state next state type of thing. So, it is 0 0 0 0 to 0 1 1 0 0 0 1 to 0 0 0 0. So, that way you fill up this table. So, as per this sequence from 0 0 0 to 0 1 0 like that so, based on that you fill up this table.

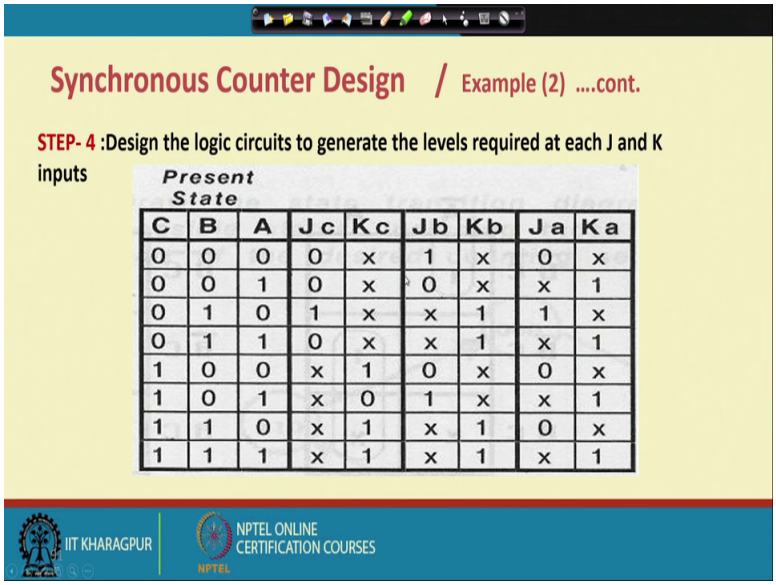
(Refer Slide Time: 17:36)



And then indicate the level required at each J and K inputs. So, that is similar to what we have done previously this J C and K C for the C flip flop J B and K B for the B flip flop

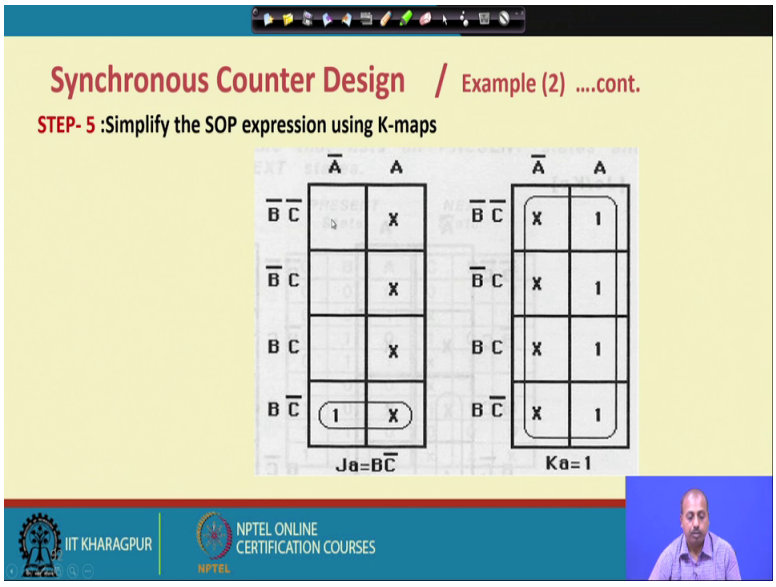
and this J A and K A for the K flip flop. So, that way we derive the, we write down the transition table.

(Refer Slide Time: 17:58)



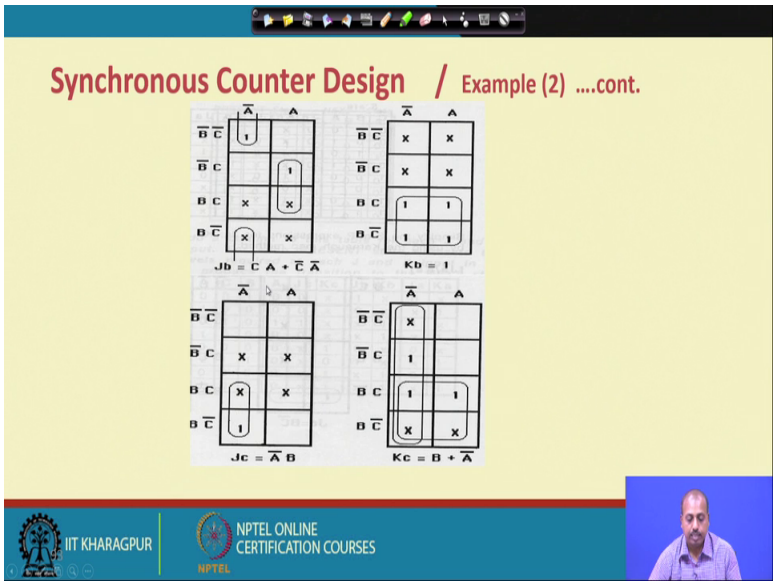
And then we have to go for the logic circuit. So, for the, for this C B A values to be this, this, this, then these 3 values. So, what are the values of J C K C, J B K B and J A K A. So, that part if you write down then you can make Karnaugh map and do a minimization for each of this J C K C, J B K B and all.

(Refer Slide Time: 18:20)



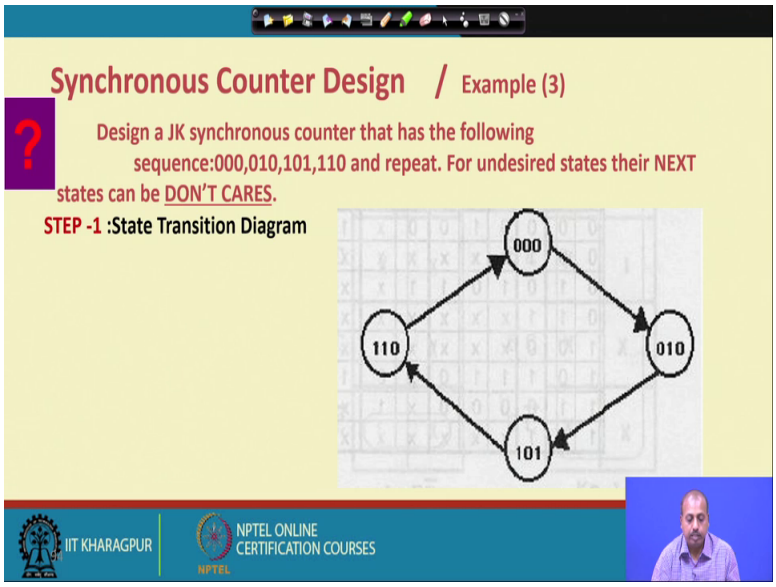
And so, for J J A turns out to be B C bar K A turns out to be 1.

(Refer Slide Time: 18:27)



Then this J B turns out to be C A plus C bar A bar K B turns out to be 1 J for J C turns out to be A bar B and K C turns out to be B plus A bar. So, now, you can final circuit is not shown here. So, that is just another step. So, you have got 3 counters and then you have got appropriate logic connecting to connected to the J and K inputs of individual flip flops.

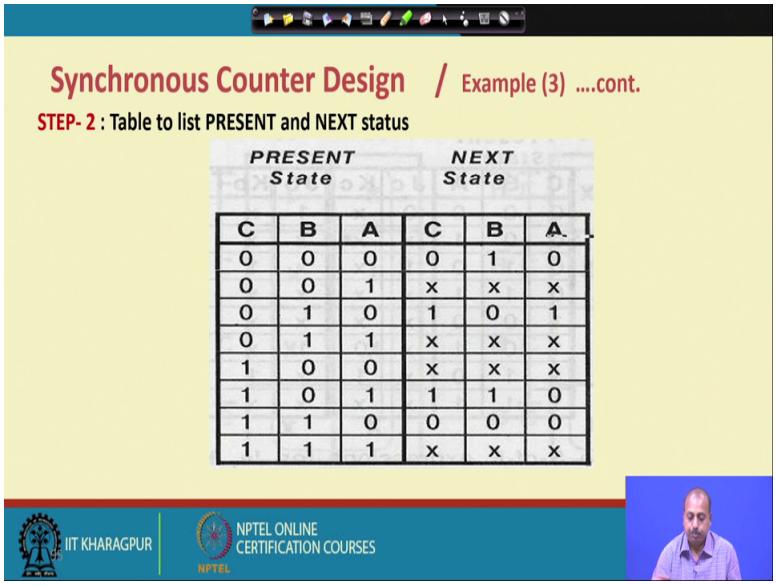
(Refer Slide Time: 18:53)



Next we look into another example where this is again a J K based synchronous counter that counts in the sequence 0 0 0, 0 0 1, 1 0 1, 1 0 1, 1 1 0 and repeat, but all the

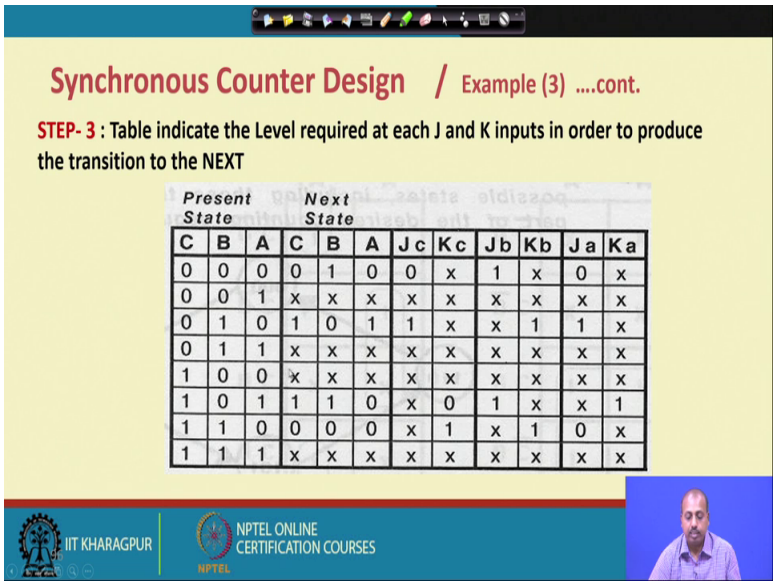
undesirable states undesirable states the next state can be do not care. So, we do not bother like if we if they start at any of the state which are not stated here then what the system will behave. So, that is not to be a concerned ok. So, that is the user will ensure that those undesirable states will never be reached by this counter; they it will never start at those state.

(Refer Slide Time: 19:35)



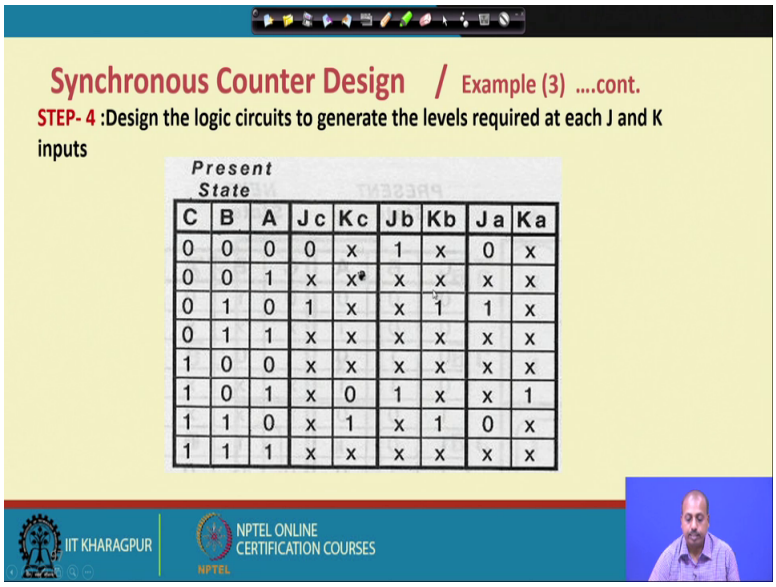
So, this C B A, it can go from 0 0 0, all the 8 combinations can occur, but out of that we are bothered about only these patterns 0 0 0, 0 1 0, 1 0 1 and 1 1 0 for example, we are not bothered about the state 0 0 1. So, at if it if the system is at count value is 0 0 1, what will be the next count value that is we do not bother. So, they are do not cares similarly if the if the current count value is 0 1 1, then what is the next count value. So, that is not relevant. So, that is do not care. So, this way you can introduce a number or do not cares into your design and after that you can do a minimization ok.

(Refer Slide Time: 20:12)



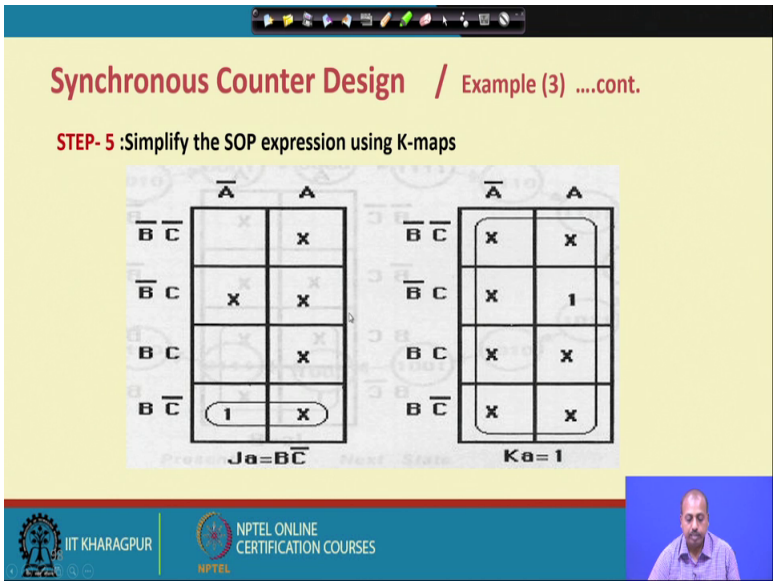
So, so, if it is 0 0 1 and it is do not care, then all these values are they will turn out to be do not cares. Similarly, we can have these thing, we can have this 1 0 0 and this is do not care so, they are all going to be do not cares. So, this way we can have this do not cares put into this J a, J b, J c K a, K c, K b columns and accordingly we can do minimization and get the final circuit.

(Refer Slide Time: 20:45)



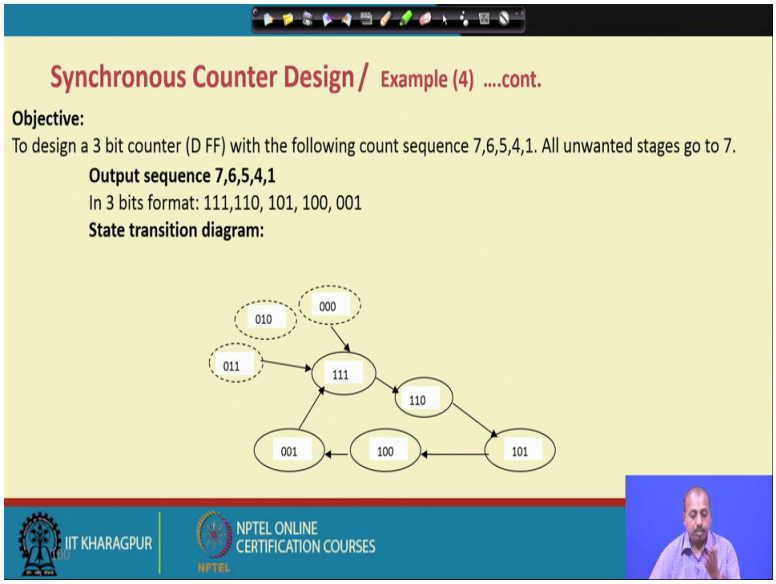
So, you can design the logic circuit for this. So, this is we just write down this table again.

(Refer Slide Time: 20:55)



And then do minimization as we have done previously and then we can make the final circuit.

(Refer Slide Time: 21:01)

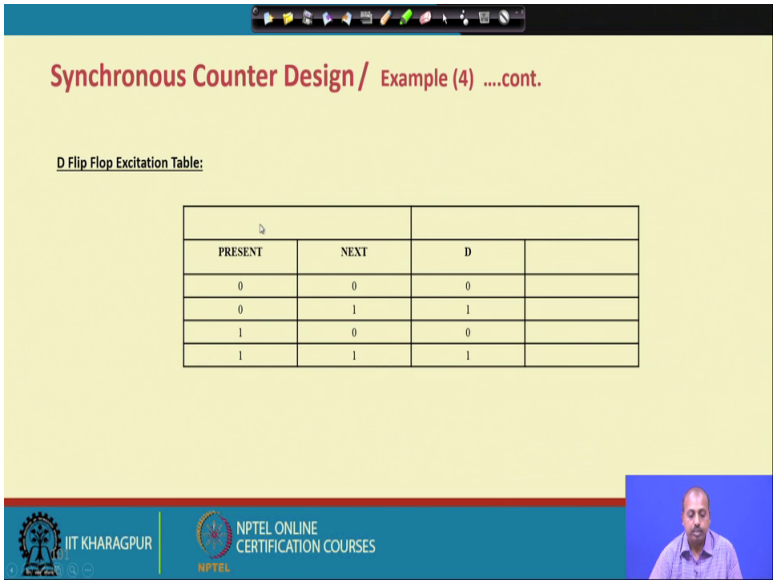


Another example that we take is to design a 3 bit counter with D flip flop with the following sequence. So, this is not based on J K, but it is based on the D flip flop. So, it will count in the sequence 7 6 5 4 1 all unwanted states will go to state 7 and this output sequence is 7 6 5 4 1, then it is like this. So, it is starting at 7, then 6 5 4 1, then again 7,

it goes like this and if its starts at other states all unwanted states like 0 0 0 0 1 0 and 0 1 1, they go to the state 1 1 1 that is 7 ok.

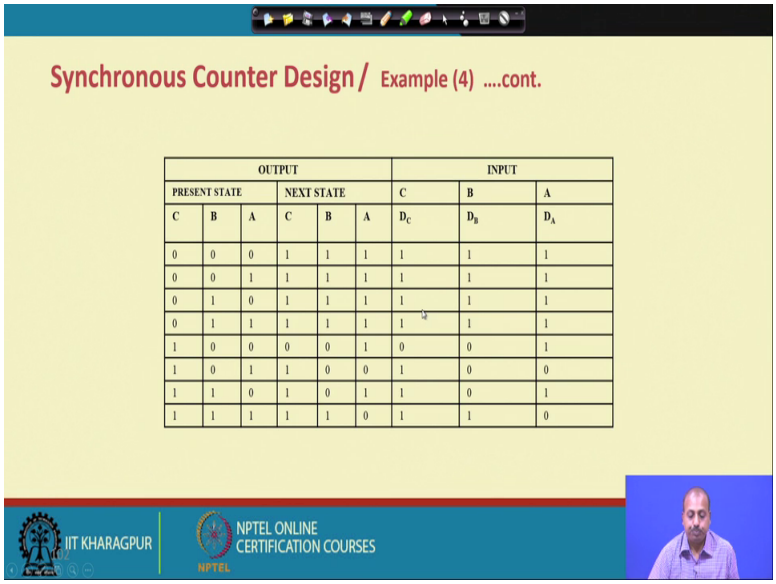
So, we can so, this is the specification of the problem.

(Refer Slide Time: 21:46)



So, if you want to design it first of all, we need to recapitulate the D flip flop excitation table. So, D flip flop table is very simple if the present state is 0 and the next state is 0. So, D should be 0. So, it is depicted by the next state value that you need. So, accordingly D value should be set.

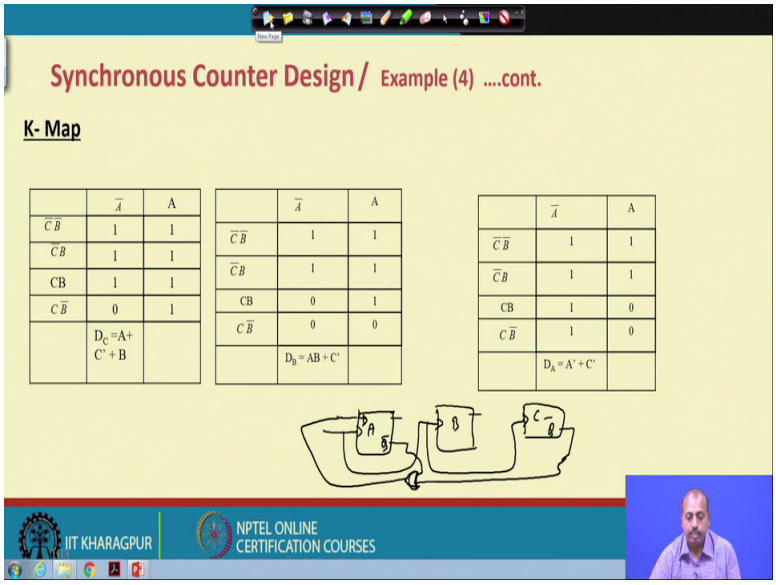
(Refer Slide Time: 22:05)



So, so, as per our specification from 0 0 0, the next state is 1 1 1. So, this D A, D B, D C. So, that is they are going to be 1 1 1 similar. So, 0 1 so far those state. So, you see here for 0 0 0 and this 0 0 1 in both the cases, the next state is 1 1 1. So, that is done here from 0 0 0, it will go to 1 1 1 from 0 0 0 1 also, it will go to 1 1 1 accordingly, this D A, D B, D C, they should be set to 1. So, it goes like this.

So, this way, we can think about this, we can make this table just we were doing it for this J K based counter. So, we can make this, we can make this D flip flop based counter and get the reason.

(Refer Slide Time: 22:57)

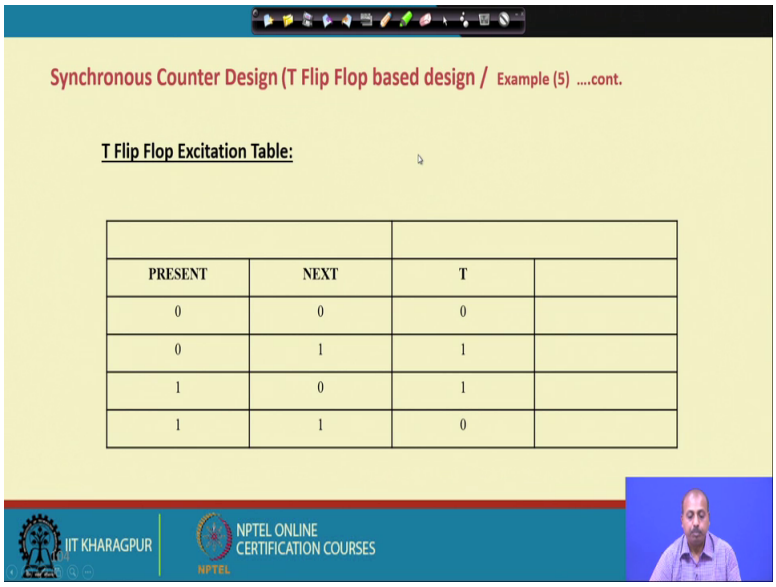


Then we take it to the Karnaugh map and in the Karnaugh map, we can have this minimization like this we can have this minimization for D C, D B and D A. So, they will turn out to be like this D A is A bar plus C bar, D B is A B plus C bar and D C is A plus B plus C bar. So, once we do this minimization. So, you can draw the corresponding circuit. So, that circuit is not shown here, but essentially, what will happen is that you will have this 3 D flip flops. So, these are the 3 D flip flops and the clock will be common to all of them, the same clock will be go to each of them, but this D input. So, this is this is the D input for the a flip flop and A for the a flip flop, it say it is A bar plus C bar.

So, this is this Q bar line from here and Q bar line for this is the A this is B and this is C. So, this Q bar line and A line. So, they should be ORed and then they should be

connected here, they should be ORed and connected here. So, in this way you can you can have the, you can have the other logic circuit other portion like say. So, for say D B, it is a B plus C bar. So, this A and B are output they should be ANDed and ORed with this Q bar output of C that should be connected. So, whatever circuit the way we are drawn the previous circuit. So, we can also make this circuit and we can get the realization in terms of D flip flop.

(Refer Slide Time: 24:42)



So, you can also design counters using T flip flop. So, the for example, this T flip flop excitation table is like this from 0 to for 0 to 0 transition, T should be 0 from one to one transition also, T should be 0 for 0 to 1 or 1 to 0 transition, T input should be equal to 1. So, that is the excitation table out T flip flop so.

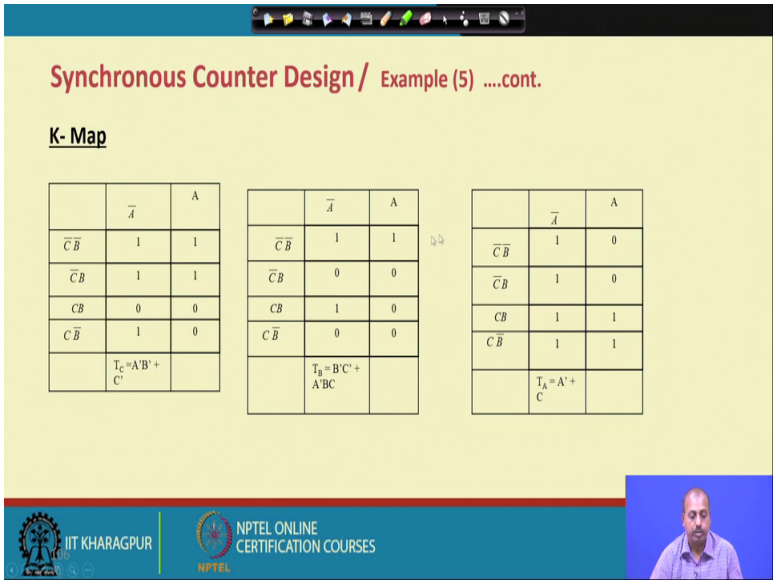
(Refer Slide Time: 25:01)

OUTPUT						INPUT		
PRESENT STATE			NEXT STATE			C	B	A
C	B	A	C	B	A	T_C	T_B	T_A
0	0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1	0
0	1	0	1	1	1	1	0	1
0	1	1	1	1	1	1	0	0
1	0	0	0	0	1	1	0	1
1	0	1	1	0	0	0	0	1
1	1	0	1	0	1	0	1	1
1	1	1	1	1	0	0	0	1

Now, if you go back look into the our counter specification then for 0 0 0, it has to go to 1 1 1. So, this T C, T B and T A; all the input should be equal to 1. Similarly, from 0 0 1 also, you want to go to 1 1 1. Now in this case, you see that A should not transit. So, T A should be equal to 0 T C and T B they should be equal to 1. So, it should go like this, fine.

So, once this is done. So, rest of the thing is simple. So, from this present state, we want to apply T A, T B, T C like this.

(Refer Slide Time: 25:35)



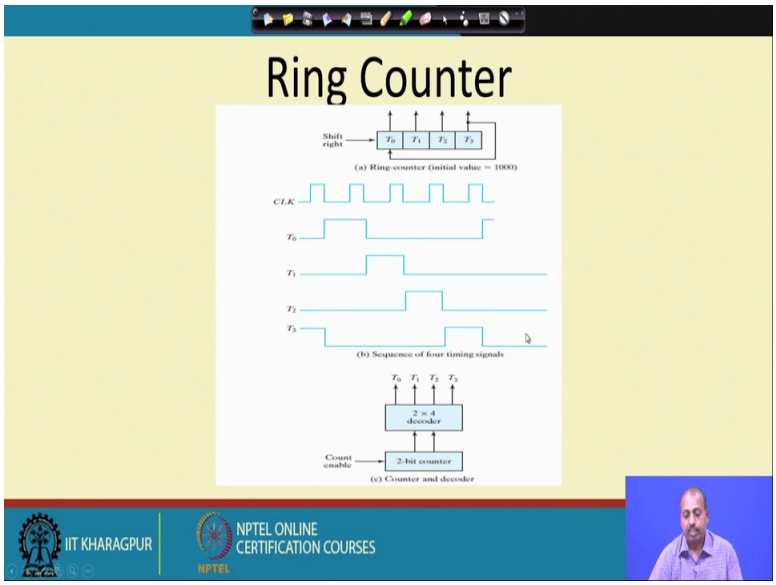
	$\bar{A}$	A
$\bar{C}\bar{B}$	1	1
$\bar{C}B$	1	1
CB	0	0
$C\bar{B}$	1	0
	$T_C = A'B' + C'$	

	$\bar{A}$	A
$\bar{C}\bar{B}$	1	1
$\bar{C}B$	0	0
CB	1	0
$C\bar{B}$	0	0
	$T_B = B'C' + A'BC$	

	$\bar{A}$	A
$\bar{C}\bar{B}$	1	0
$\bar{C}B$	1	0
CB	1	1
$C\bar{B}$	1	1
	$T_A = A' + C$	

So, you have to derive the corresponding functions and then. So, this TC turns out to be A bar B bar plus C bar, T B will turn out to be B bar C bar plus A bar BC and T A will turn out to be A bar plus C. So, that way we can have this Karnaugh map for this T flip flop and accordingly, we can make the circuit for the T flip flop based realization.

(Refer Slide Time: 26:00)

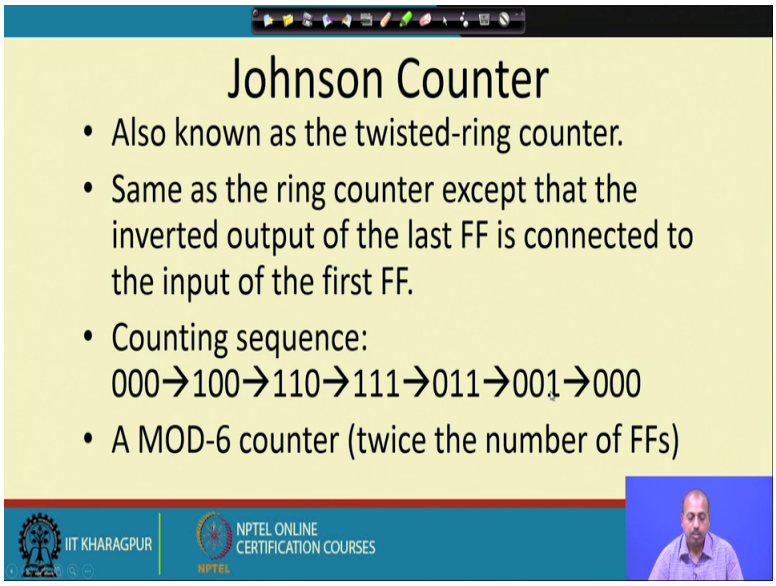


So, another important type of counter that we have is the ring counter. So, in ring counter we have got say this initial it is realized by means of T flip flop then this initial value of the ring counter is say 1 0 0 0. So, this is T value is. So, this T 3 value is one and these values are 0 0 0, then since this is a T flip flop and this input is 1. So, the next clock pulse ok. So, this T 0 will be making a transition. So, at the falling edge T 0 makes a transition and it becomes equal to 1, but rest of the otherwise this is a shift register.

So, this that way this is this values are shifted. So, this the T 2 value. So, that was initially 0. So, it comes to this and then you can have this type of shifting ok. So, this is it will be it will be generating the count in the sense it will first 0 0 0 1 and then 1 0 0 0, then 0 0 0 1, 0 0 1 0, 0 1 0 0 and 1 0 0 0. So, you can do the same thing by means of this 2 bit counter and a 2 2 to 4 decoder what so, this 2 bit counter is a standard counter.

So, it will go from 0 0 0, 1 1 1, 0 and 1 1 when 0 0 0 is given as input. So, T 0 is made high when 0 1 is given as input this decoder will make T one equal to high. So, it will go like this. So, that way we can get a decode counter and we can realizing.

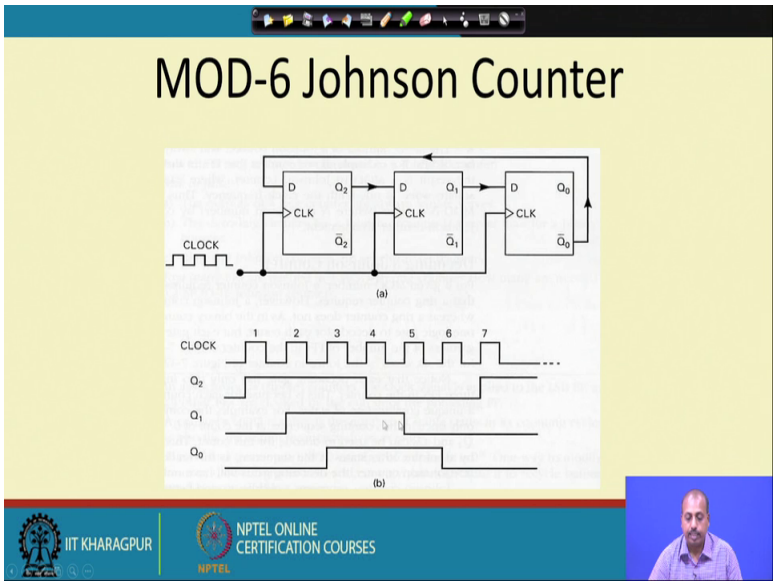
(Refer Slide Time: 27:40)



Another counter is known as Johnson counter. So, this is also known as twisted ring counter because it requires less number of flip flops. So, it is counting in the sequence like say 0 0 0 to 1 0 0, 1 1 1, 0 1 1 1. So, it generates all the 6 patterns, it goes in a mod 6 fashion. So, the advantage is that you number of states that you are getting is twice the number of flip flop. So, we have got 3 flip flops, but using 3 flip flops you can get 6 different patterns ok.

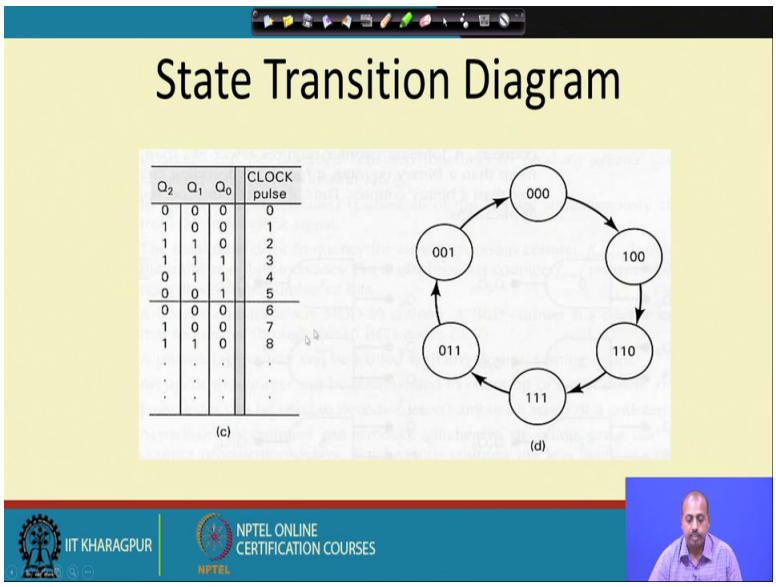
So, that type of this is known as Johnson counter, it counts in this particular sequence 0, then 4, then 6, then 7, then 3, then 1, then 0.

(Refer Slide Time: 28:30)



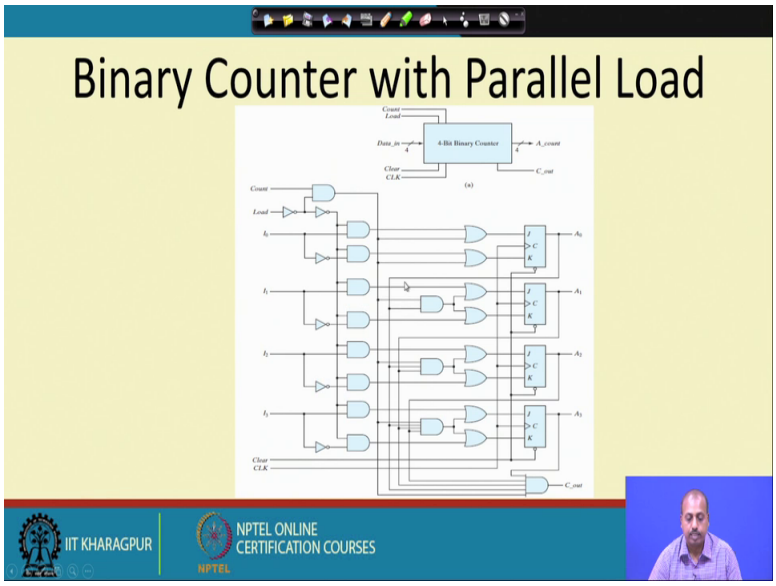
So, this is the corresponding circuit of the Johnson counter. So, you can just trace through this circuit and see that it really generates count in that sequence ok.

(Refer Slide Time: 28:41)



So, the state transition diagram is like this and you can get the corresponding circuitry from there.

(Refer Slide Time: 28:45)



So, finally, in a 4 bit binary counter. So, we can have generic binary counter may have this load facility count of facility clear facility which will clear the counter we can load the counter with some initial value and some when this load input is given and this data input is coming. So, this that is loaded into this 4 bit counter and it goes like that. So, this way, we can design a generic counter which has got the parallel load facility also, this similar to this shift register loading and with that counting facility is added.