Analog Circuits and Systems through SPICE Simulation
Prof. Mrigank Sharad
Department of Electronics and Electrical Communication Engineering
Indian Institute of Technology, Kharagpur

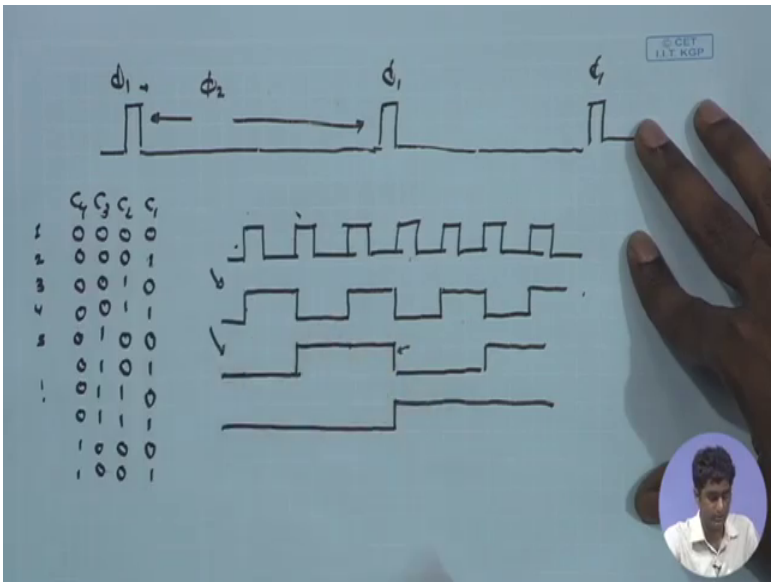
Lecture – 46
Flip – flop

Welcome back. And today let us start our discussion on the digital component of our e to D converter. We have been discussing the front end amplifier and the filter blocks along with that we have looked into the ADC where we started discussion with the individual blocks of our single slope ADC, we looked into the comparator design, we looked into the ramping circuit the sampling circuit and finally, dealt with some of the non idealities of the comparator like offset cancellation. And now we also discuss the overall scheme of clocking and how to use the counters scheme to implement the A to D converter.

Now today we will be looking into the overall design for the digital component of course, at the core of it we expect it to be a digital counter. And will first of all describe the block level implementation of the digital component which can implement our functionality of A to D converter. Basically the functionality we are targeting is trying to digital converter. So, you are having a time measurement you are trying to measure the time after which the comparator makes a transition and you are trying to convert that into a digital word. Of course, therefore, counter is definitely very logical you need to be used. We are going to look at the design of such a counter at block level. And trying to see how are we going to generate the control schemes using that counter for controlling the switches like the sampling switches and the switches controlling the ram circuit etcetera, and once we have done the block level description. We will go ahead and look into the transistor level design of individual blocks constituting the digital module whatever logic creates and the flip flops are required we will looking into the transistor level design and try to understand some of the intricacies and the issues related with the transistor level design of the digital units.

Let us get started. So, remember the overall operation that we have.

(Refer Slide Time: 02:33)



You have the 2 phases, the fibrin phase is being used for sampling and it is also being used for offset cancellation. Whereas, the phi 2 phase is being used for the comparator operation and overall counting operation. So, this is the overall scheme we had remember during phi one we are sampling the data at the same time we are doing the offset cancellation or offset sampling of the comparator. The data is sampled and the offset of the comparator the samples and the phi 2 phase the comparator operation is initiated. So, the ramp circuit starts charging the capacitor voltage starts rising and the counter also starts counting.

So, at the core of it of course, we have a counter at hand with the help of which we are going to implement this overall operation. So, based on the counter we can see how to implement this control loop, how to obtain the phi one pulses and likewise, how to control the transitions of phi one and phi 2 pulses. For that we first of all need to look at the block level implementation of counter. And before that we need to look into the constituents of the counter which is ultimately going to be latches and flip flops.

So, going to the basics of the first try to look at the definition of latches. And flip flops and then with the help of that we will try to see how to construct counters. Will try to see 2 counts of kinds of counters one is asynchronous or repel counter which is going to be good enough for our particular application. I thought I try to discuss synchronous counter which may not be required in our case, but only increase the operating frequency is

larger and the clock frequency is larger for certain other applications you may be required to put a synchronous counter try to distinguish these 2 cases. And then we will pick up the asynchronous or the ripple counter. So, called ripple counter and try to look into the block and transistor level designs for each of the modules in the ripple counter.

So, if we look at the overall operation of counting. So, ultimately we have save 7 bits. Suppose I am just looking at the last 4 bits. And the overall operation that each of these bits perform you have the LSB the least significant bit starting from 0 it goes to 1 0 1 0. So, it alternates every clock cycle. Likewise if I look at the second LSB. Here we have 0 0 1 1 0 0 1 1 something like that. So, it alternates every 2 cycles. So, I can extend, likewise the third one it alternates every 4 cycles and so on. The 4th one will be alternating after 8 cycles.

So, will be 0 0 and so on, it will be remaining one for the rest of 8 cycles. So, this is the LSB least significant bit alternating every clock cycle first clock edge first clock cycle if it is 0 next 1 0. Likewise second one alternating every 2 clock cycles third one after 4 4th one after 5. So, this is the operation of a counter. So, if these are the 4 bits coming out of my counter. I expect this behavior from each of those 4 bits. So, if I look at the operation that each of these outputs are doing with respect to the applied clock. So, if you have a clock we say that this transitions of the LSB are happening at every clock edge or every rising edge of the clock. So, if you have the input clock we say that the I can call it c 1 c 2 c 3 c 4.

So, these are the c 1 is the first bit LSB e c 2 is the second third 4th bit and so on. So, here for the c 1, but we see that it is going to start with 0. And at the first edge suppose it goes to 1, at the second edge of the clock after one period it goes to 0 again. After the third edge of the clock it goes to 1 again and it keeps repeating and so on. So, this is the first clock edge second clock edge third 4th fifth and so on. So, at the just after the first clock edge I can say the c 1 is turning 1, likewise if I look at the c 2 behavior c 2 is starting from 0 and after say the after the c 1 c 2 is going down c 1 is supposed to go up.

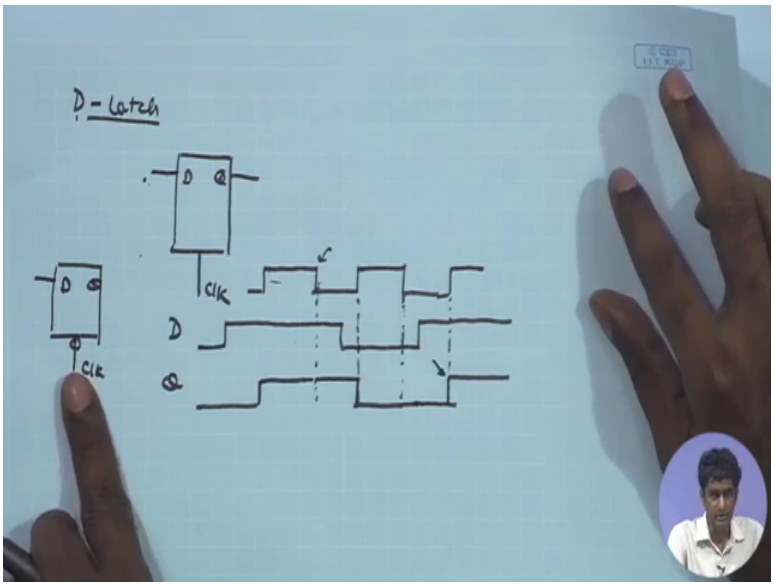
So, we have after cone goes down c 2 goes up and so on. And then once again at the next transition of c 2, when the c 1 goes up and once again we have the c 1 going down once again we have the c 2 waveform propagating. So, if I look at the waveforms we effectively see that the c 1 c 2 and so on they are having frequencies which is divided by

2 of the previous stage. So, if this is the original clock we can see that c 1 is also behaving like a clock with frequency half that of the input clock. Likewise c 2 is also going to behave like clock with frequency one by 4th of the into clock and so on. And they are triggered at the transition of the previous clock pulses. So, if I assume that the c 1 is triggered by the clock wherever the clock is making the transition even is making a transition.

So, here I have assumed the positive transition positive clock trigger transition. And whenever c 2 is making a transition high to low sorry, the c 1 is making a transition high to low c 2 is making transition. Like y is assumed that c 3 whenever c 1 whenever c 2 makes a negative going transition if I assume that c 1 is going c 3 is going to make a transition see the waveform will be looking like this. So, at the negative transitions of the c 2 c 3 is going up. So, subsequently I have the frequencies of the clock getting divided by 2. So, this is the operation we would like to implement using the digital circuitry. So, the counters or the circuitry incrementing the counter ultimately should be able to implement this divided by 2 operation starting from the original clock going on dividing the clock by 2. So, and the other important consideration is that the overall operation of each of these bits is going to be edge triggered.

So, the transition happens at the edges of the previous clock or the previous phase. So, we need to look at the edge triggered operation and therefore, we are going to look at the flip flop. So, for sequential circuit is which can give us clock driven operation can be level sensitive or edge sensitive. So, let us see the definition of level sensitive and edge sensitive blocks and then look at the definition of flip flop which is basically cascade of 2 opposite phase level sensitive latches. And with the help of that we can see how we are going to construct such as such a counter which is performing overall counting operation.

(Refer Slide Time: 10:54)



So, one of the common Latch topology can be identified as the D latch or the data latch, which can which is driven by a clock. And you have that D input and the Q output. And here the functionality is that when you have the clock high then the latch is transparent. So, during the high phase of the clock the latches transparent; that means, whatever data is available at the D input the same data will be transferred to the Q output. So, if I assume there is an arbitrary data waveform coming in, suppose this is the data this is the D input and I want to see the Q output. So, when the clock is high the data is going to be transferred to the Q. So, during the high phase of the clock whatever data D is available Q will be copying that. And during the negative phase of the clock the connection is cut off and whatever data was stored at Q at the falling edge of the clock it will remain there as it is.

So, during this duration the connectivity between D and Q is broken. And Q is going to store the level that was available just before the falling edge of the clock. So, during the high clock D and Q are you can assume that they are directly shorted or you are having direct connection between D and Q whatever data is available at due D the same data is transferred to Q, but once the clock goes down then it is no longer transparent. Whatever data was available at Q it remains stored and therefore, any change in D during the negative clock phase will not be transferred to Q. So, Q is retaining the D level which was available before the negative edge of the clock. And then once again when the positive phase of the clock comes then when the clock becomes active again and

therefore, D and Q becomes connected again or the latch becomes transparent, again we can say that once again the new D value is going to be available at Q. So, at this point once again the Q will make a transition.

Now while the latches on or while the clock is high? The D is getting copied to Q and once again as the clock goes 0 once again the transparency is lost and therefore, this point onward the Q will be retaining whatever D value was available in this positive phase. And once again after the clock goes high again the Q will be once again connected to D once again it will be copying the value that is available at D. So, this is the overall latch operation, latch becomes transparent when the clock is high and it be it disconnects the D and Q, and the Q retains the level that was available at the following edge of the clock. This is the operation of a latch.

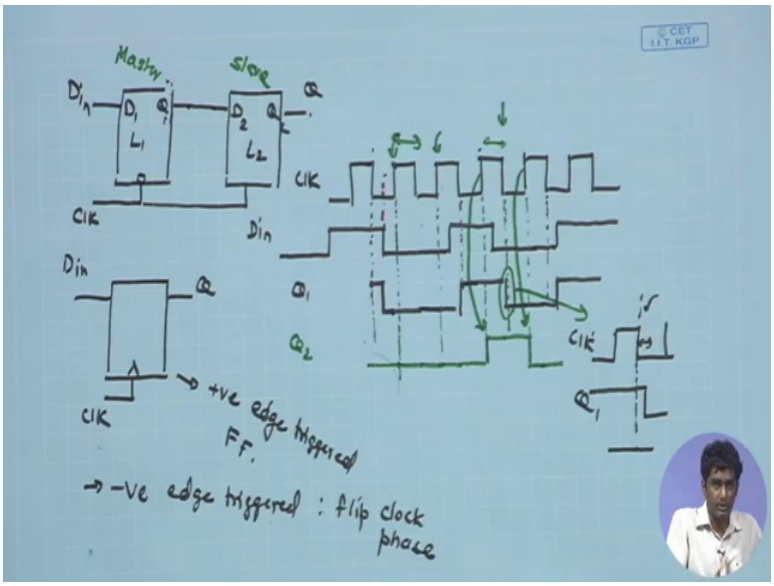
So, why do we call it a latch because it is latching on to the data which was available at the falling edge of the clock. So, during the high phase of the clock it just copied the data directly. So, it remain transparent, but when the clock went low it latched on to the data that was available at that falling edge of the clock. For example, if this is the falling edge of the clock whatever data was available before the falling edge of the clock it is retained for that entire low phase. So, even though the data is making a transition during the falling edge of the clock the Q output of the latch is not changing. So, this is positive level triggered latch or it is transferring or it is becoming transparent when the clock is high. You can also have latches which are negative level sensitive or they are going to be transparent when the clock is negative. So, for that I can put a bubble over here. And you can say it is an active low latch. So, when the clock is low then it will be transparent and the clock is high then it will be latching the data or storing the data at the that was available at the rising edge of the clock.

So, both kind of latches can be constructed. Now the operation that we want for the counter operation as we have seen we need the edge triggered operation it should not be level sensitive, but it should be edge sensitive. So, the data transition should take place only at the edges of the clock either the positive edge of the clock or the negative edge of the clock. So, here we can see of course, the data transition the cue transition can take place even during the positive phase of the clock at any point during the positive phase of the clock if the data is changing cue will also change. But for the counter operation we need an edge sensitive block where the transition is going to be confined only to the

transitions of the clock whenever there is a transition in the clock 0 to high transition only at that instance the data should be latched to cue or the cue should be changing the state.

So, we can construct a flip flop using such latches, which are triggered by opposite clock phases. So, high level sensitive latch and a low level sensitive latch if we cascade it together we can realize a an edge sensitive block which is can be considered visually (Refer Time: 17:29) called a flip flop. Let us see the operation of a flip flop when we cascade 2 such blocks one block which is high level sensitive another block which is low level sensitive. And then see how this is the edge triggered operation comes into picture.

(Refer Slide Time: 17:47)



Let us call it D 1 Q 1 D 2 Q 2. And you have the output of Q 1 connected to the input of the second latch. So, I can call this latch 1, L 1 let us call this latch to latch 1 is low level sensitive. So, this becomes transparent when the clock is low, that is denoted by this bubble. And we have the latch 2 which is high level sensitive.

So, this becomes transparent when the clock is high the same clock is being fed to both of them and the D the input D in irravailable over here. And we have the final output Q over here. So, I want to get the waveforms of Q corresponding to the change in D in and the clock. And let me this is the Q 1 and the 2. So, let me first of all other clock waveforms and see the overall behavior. This is my clock. And suppose I have some random data it may be making transition at arbitrary places. This is your D in. Which is

coming from some previous stage. And it can make transition anywhere and then I need to look at the waveforms of Q 1 and Q 2 which is our final output. So, how does the Q 1 change. So, Q 1 is negative level sensitive or active loop latch. So, when the clock level is 0 then the Q 1 will be copying D in.

So, if I start from this phase the clock is low here. Therefore, during this duration the Q 1 will be copying the data D in which is high. So, I am writing the reform of Q 1. So, during this low phase of the clock the low value of the high value of D in will be transferred to Q 1. And as soon as the data is going low during this phase the Q 1 also goes low. And before the positive going edge of the clock the data is remaining low and therefore, during this entire positive phase of the clock the Q 1 will be remaining low. This is the operation we discuss for Q 1. Once again for the next negative phase of the clock when the clock is low data is remaining 0 therefore, Q 1 will also remain 0. Once again during the positive phase of the clock the L 1 is not transparent. Therefore, during the positive fills the clock whatever transition happens in D the Q 1 will be opaque to that. Therefore, during the positive phase of the clock once again there is no transition. At this point; however, when the clock has gone low you have the D in changing. So, D in has changed at this point from 0 to 1 in the next low phase of the clock the Q 1 receives the D in which is high therefore, the Q 1 will be making a transition over here. And during this entire low phase the D in remains 1.

So, therefore, the Q will be remaining one during this entire phase and before the clock goes high against the b inverse low D in was high as a result during this entire high phase the Q 1 will be remaining high. Again when the next negative phase of the clock comes the din has changed as the result once again the low value of din will be copied to Q 1 therefore, at this point once again you have the Q 1 going low. During the positive phase once again the transparency is lost therefore, it will retain the same value and as we see when the clock goes again low once again the data till this point is 0 therefore, it will remain zero, but the moment the data din goes high during this negative phase of the clock the Q 1 will be copying that and therefore, will be making a transition. And once again in the negative phase it will be retaining the data. This is the overall waveform and of course, we can see that it is not edge sensitive it is the level sensitive operation. So, it is changing the state during the negative or the low phase of the clock provided the data is changing the state.

So, whenever during the low phase of the clock the data changes it is state the Q 1 also change state. So, you can see during this period the data is changing the state as a result of Q 1 being transparent to it is also changing the state. Likewise another zone over here where the latch 1 is transparent data is changing a straight from 0 to 1 as a result Q 1 is also immediately changing the state, but if the data transition happens during the positive phase of the clock. During that period Q 1 is not going to change because during that period it is opaque it is not transparent to the changes in d in. So, during the positive phase of the clock whatever value was available in Q 1 as the rising edge of the clock it will be retained for example, in this phase once again we can see that in the positive phase the data has changed in the state, but Q 1 will not change unless the clock goes low. Again this is the operation of Q 1 that we just discussed the latch 1.

Now this is the input going to the Q 2 or l 2. So, this Q 1 waveform is the output of the first latch in input the second latch. And therefore, it is going to determine what is going to be the output Q 2 at the positive edges of the at the positive phases of the clock. So, Q 2 is positive level sensitive. So, when the clock is high then Q 2 will be copying the data D 2. So, let us see what is the situation for Q 2. So, the data is the data is going high the clock is going high at this particular location if you see. And therefore, the Q 2 is going to be transparent in this period. And during this period we can say that the Q 1 is low or the input to the l 2 is low, as the result we will have the Q to make remaining low at this point. So, the this entire duration it remains low Q 1 is low therefore, (Refer Time: 25:25) it will be going low. And then when the clock becomes low again definitely it is going to lose transparency, and whatever data was available at the falling edge of the clock it will retain that.

Now, if I look at the next positive phase of the clock once again the Q 1 is 0. So, once again after this instance once again the l 2 will becoming transparent, Q 1 input is remaining 0. Another result it will also retain the same value. Now once again till the positive edge of the clock once again after this low phase when the next positive phase of the clock comes. The l 2 is seen high level at Q 1 therefore, during this positive phase of the clock the input to the D 2 is high, as a result of which the Q 1 the will make a 0 to 1 transition at this point. And once again during the during this entire positive phase of the clock it will remain constant it will remain retain it is state.

Now, during this negative phase therefore, the level of Q 2 remains as it is it remains constant. If you look at this point this can be a crucial juncture to observe at this point as the clock is going low Q 1 is making a transition from 0 to 1 sorry, a 1 to 0. And just before this falling edge of the clock the Q 1 was high, as a result during the positive phase of the clock Q 2 was also high because it was transparent to Q 1. Right at the falling edge of the clock Q 1 make transition; however, for the Q 2 if you look at it whatever value was available for Q 1 just before the falling edge of the clock that will be retained for Q 2.

So, the definition of the latches the latching operation will retain the value of the data which was available immediately before the falling edge of the clock in case of a positive level sensitive latch. So, here we are seeing that at this particular time instants the clock phase has gone low immediately before that whatever was the data available at the input of l 2 that will be retained at the output Q 2. So, although just after this clock edge the Q 1 has changed it is data Q 1 has changed it is level, but the Q 1 the Q 2 will be retaining the state or the input which was available just before the falling edge of the clock. Because if you look at if I if I magnify this if I magnify this instance for Q 1. If a look at realistic circuit is will see that when the when the clock makes a transition over here and corresponding to that the Q 1 is making a transition at the falling edge of the clock because it is negative level sensitive. It will have certain delay it is not ideal it will have certain delay for that transition. So, with respect to the falling edge of the clock Q 2 will have certain delay. And as a result just as the falling edge of the clock the Q 2 is receiving a high level Q 1 is remaining high at the positive at the negative transition of the clock.

So, for Q 2 as the clock is turning low at the falling edge it is receiving a high level the Q 1 is remaining high and it is going to go low it is going to have a transition at the negative edge of the clock. Another result if I assume that Q 2 is going to be is Q 2 is going to become opaque to the changes in Q 1 just after this negative edge of the clock; that means, it will retain whatever data of Q 1 was available at this falling edge. So, it will basically retain that data for this entire duration till the next positive edge of the clock comes. Therefore, here we are seeing that for the Q 2 latch whatever data level was available just before the falling edge of the clock that will be retained. And hence that is the value of Q 1 which was available before the following edge of the clock.

So, that value is going to be retained over here, till the next positive into the clock comes. And at this point once again you have the positive transition of the Q 2 happening Q 2 becomes transparent again as a result once again it will be let copying the value of Q 1 that is available to the input during this positive phase. And therefore, the Q 2 makes the transition once again at this positive edge, because it is getting transparent only during the positive phase therefore, it will make transitions only when the positive edge of the clock comes and; however, Q 1 will make transition when you having transition from high to low. And as a result I have if I look at the overall operation of Q 2 we can see that it is always going to make transition at the positive edges of the clock.

So, we can see at the positive edges of the clock only the transition is Q 2 is going to be seen. So, the in the summary the operation is that the latch Q 1 being negative level sensitive it is making transitions during the negative phase of the clock. So, it is transparent during the negative phase of the clock. Whenever the clock is low it becomes transparent and whatever data transition is happening it will be copying that. And just before the clock makes 0 to high transition, the L 1 becomes it loose the transparency it, becomes opaque and therefore, just before the positive edge of the clock it becomes opaque and therefore, holds the data latches the data. And in the other case if I look at the l 2 operation it is just doing the opposite. It is becoming transparent during the high phase. And therefore, whatever data is available whatever data is available to the input of D 2 during the high phase it will be copying that to Q 2, but remember that Q 1 is not making any transition during the high phase of the clock. Q 1 is remaining stationary during the high phase of the clock.

And therefore, when the clock makes a positive transition whatever state of Q 1 was available at that positive h that is retained for the entire period of the clock. So, during this particular during the positive phase of the clock the Q 1 output did not change it was stationary whereas, the l 2 latch was transparent therefore, during this phase although l 2 can be transparent to the data over here in the any change over here can be conveyed to the output, but Q 1 is not changing during this phase therefore, that ensures that the D 2 input is not changing during this phase and therefore, whatever D 2 input was available over here or in other words whatever Q 1 was available over here that is being transferred to Q 2. And as soon as the clock goes low the l 2 becomes opaque and as a result it is no longer going to transfer any changes in Q 1.

So, during this phase once again there is no scope of transferring any change in Q 1 to Q 2. So, this is the overall 2 phase operation and we call the first latch as the master latch and the second one as a slave latch. And the master slave operation overall achieves the edge sensitive transition for the second stage l 2 only when there is a transition in the clock edge in this case a positive transition in the clock edge then only the Q 2 is going to change it is state. So, in other words slave effectively is latching on to the data provided by the master at the positive edge of the clock. So, whenever there is a positive transition in the clock the slave is able to copy the data that was available over here. Stable over here during the positive phase. So, during the positive phase the master does not change as a result whatever data was made available by the slave at this positive edge of the clock that is transferred to the output.

So, this is the positive edge triggered clock edge triggered flip flop and we can denoted by a symbol were at the place of the clock we put small triangle like this which is going to denote that it is edge sensitive latches, we do not put this triangle; however, for flip flop we put this triangle to denote that it is a edge sensitive blocker the flip flop and not a latch. And then we have din and then you have the Q of the flip flop. So, these 2 can be captured in the form of this block. And you have the clk how do we make a negative edge triggered flip flop we just need to reverse the polarity of the clock. So, this becomes clock and this becomes clock bar then the Q 2 will be transitioning at the negative edge of the clock. So, this is the positive edge triggered flip flop, in order to make a positive edge triggered negative edge triggered flip flop, flip the clock polarity

So, in that case this becomes negative less level sensitive this becomes positive level sensitive. And as a result the transition over here the Q 2 transition will be happening only when the clock is going from high to low because only during this low. Phase this will be becoming transparent as a result only when the clock goes to low it will be transferring the data Q 1 to Q 2. And during the high phase it is anyway opaque. So, that will be a negative issue sensitive flip flop. So, this is the overall concept of latches and how do we construct edge sensitive flip flops with the help of latches operating with opposite clock phases, master and slave. And with the help of this we are going to see we are how going to construct the counter and from there we look into the control signals that we are going to generate with the help of that counter. And finally, we will go into the transistor level implementation of these latches and flip flops and try to see what are

the important timing constraints that we need to address in transistor level implementation of this flip flops.

So, the control scheme is not is going to be relatively straightforward if we stick to our original designs that we have discussed. Of course however, if you try to make the control more sophisticated add more features to it as we as you see the control logic will becoming a little bit more sophisticated. So, that so, we can stop here for a short while and see there is any question before we take a small break.