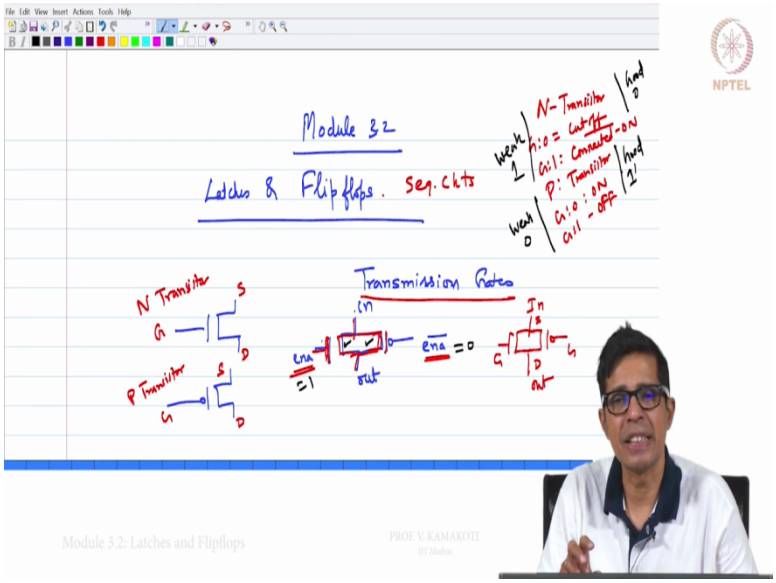
Foundations To Computer Systems Design
Professor V. Kamakoti
Department of Computer Science and Engineering
Indian Institute of Technology Madras
Module 3.2
Latches and Flip-flops

(Refer Slide Time: 0:16)



So welcome to module 3.2 in the previous module 3.1 we have basically explain you the need for sequential circuits. Why there is a necessity to remember something that has happened in the previous clock cycle? So we did explain the need but we did not give you the circuit tree which will basically ensure that whatever is computed in the previous cycle.

So previous clock cycle is basically remembered, so what is the circuit tree that can remember this? Find, so let me explain to basic elements that forms what we call as the sequential circuits and their latches and flip-flops. Latches and flip-flops can remember the values for across a clock cycle for example, right? So these are elements which can remember the bit values.

A latch and a flip-flop can remember one bit, right? And what is a latch? What is a flip-flop? What is the difference? How can we construct them? These are some of the things that we will see in this particular model. First and foremost we have seen to types of transistors, the N transistor and the P transistor. The N transistor is shown here we have explained it very earlier in this course itself.

Now there is for all these transistors there is a gate there is a source there is a brain, here also there is a gate, source and drain. When the gate is 0 the N transistor is cut-off, when gate is 0

it is cut-off, when gate is one it is connected, we say this is on and this is off. When gate is 0 the source is electrically disconnected from the drain it becomes something like an open.

Then gate is one the source is connected to the drain, the P transistor works in the other way that is when gate is 0 it is ON and gate is one it is off opposite. Now what we have done is, we are deciding now something called a transmission gate where we have put a P transistor and N transistor back-to-back if you see here, right? There is a P transistor, there is N transistor connected back-to-back.

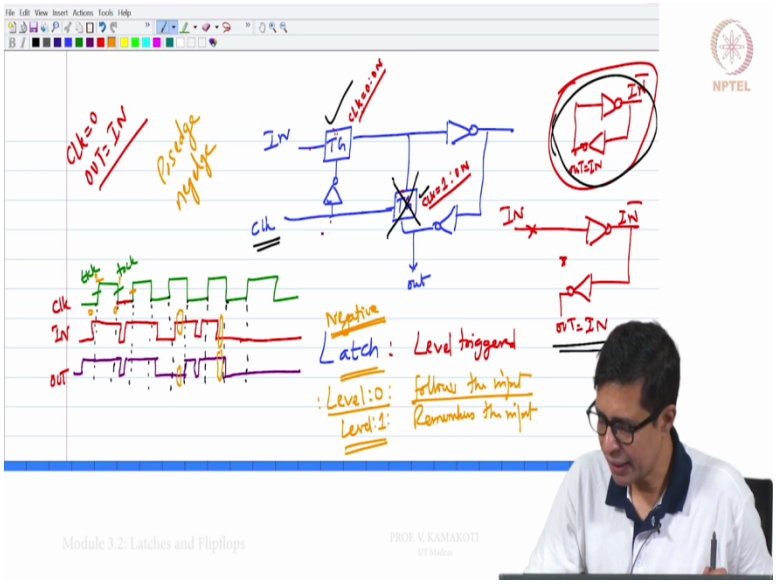
They are connected here and this is P and this is N, right? And this is the source and drain for both these are the gate and of course I give some input here and I collect and output there. So I connect the signal enabled here and a signal bar here, right? So when a enable is 0 please note that this N transistor is cut-off since enable is 0, enable power is one, so this P transistor is also cut-off.

So when enable is 0 IN is electrically disconnected from out when enable is 1 what happens? When enable is one, enable bar becomes 0 this N transistor is on also the P transistor is on, now in is connected to out when enable is one. Interestingly the P transistor if you give a 1 on the source it will give you a strong one on the output. So P transistor is a good carrier of 1.

While N transistor is a good carrier of 0 and it's a weak, N transistor is a weak carrier of 1 while P transistor is a weak carrier of 0, right? So when both the transistors are on while enable is one enable bar is 0, so P is on and N is on when I give a 0 as the input the N transistors will make it a good 0 though the P transistors makes it a weak 0, good plus we will be good, so I get a good 0 as output.

Similarly when in is one the P transistor will make it as a good one N will make it as a weak one good plus weak is again good, so I get a strong one. So this transmission gate will take when the enabled is one it will take a 0 and produce a strong 0, it will take in a one and it can produce a strong one, right? So this is how the transmission gate works and importantly when the enable is 0, the in and out are electrically disconnected. So that is the most important aspect of the transmission gate.

(Refer Slide Time: 5:37)



Now with this transmission gate we will tell you how a latch is constructed. What is a latch and how is it constructed? Now note that this is a TG transmission gate this is also a transmission gate, this is the enable and these are inverters and this is a clock, right? Now this circuit that you are seeing is called a latch, right? And what is a latch? Latch is 1, so let us see the clock how does the clockwork?

So let the clock go from, so this is how the clock goes, so we may say that this is tick tock, tick tock tick tock, so this is actually the terminology will be using in our thing. So how does this latch work? So let me say that we will just see this. So this is a clock and this is the input.

So let me say that the input became one here and so on, okay. Right? Now let me just draw also the basic this is the clock pulse rate, this one. Okay, now let us see how the output is going to be here? Please note that when the clock is 0 this TG is on, when the clock is one this TG is ON, right? Because this is connected when the clock is 0 this not of 0 is 1, so clock 0 means this TG is ON, clock one means this PG is ON.

And vice versa when clock is one this TG is off, clock is 0 this is off when this TG is ON whatever comes from the input when this TG is on that is when the clock is 0 this is on, this thing is off, right? When the clock is 0 this TG is on while this is off, so this circuit will look like an inverter with an inverter and this is the output, right? So this TG is ON while this TG is off, right?

Because now this is ON means in is connected to the out, so this looks like a short and it comes here and output, so whatever I gave as an input will get this, will become IN bar and again IN, so out equal to IN. So when clock is equal to 0 out equal to IN and this out equal to IN will happen till clock is 0. When the clock comes 1 what will happen? When the clock becomes one, note that this is 0.

So when the clock becomes one this gets cut-off while this TG gets cut-off whichever I mark here and this TG, the 2nd TG actually gets ON. So when the clock becomes one it is something like the circuit is something like a feedback circuit because this is a short the TG that I am showing in red here is a short when clock is equal to 1 while this… So what happens is just before the clock became one.

Just before the clock became 1 whatever was the value that would have been at this place that out should be equal to that IN. So when the clock actually becomes one than that value IN will be fed here this will become IN bar again IN bar is in, so this will be a stable loop, right? So IN bar IN, IN bar IN, IN bar IN, right? So when the clock becomes one what will happen?

Whatever was there just before it became one will be retained and it will be going in circles in this particular U circuit as I am showing in black here, right? It will be just going inside, again when the clock become 0 then what will happen? When the clock again becomes 0 this will become ON and this will actually become off and then the fresh invaluable start out will become IN.

So let us look at this, now how will out behave? Let us see this now, so the clock is 0, so when the clock is 0 so this is enabled when the clock is 0 this is enabled, so this will behave, out will behave exactly like IN when the clock is 0. When the clock becomes one when the clock becomes one, right? so this will remain the same thing, okay.

(()) (12:50) input like this, so when the clock is 0 the out will behave exactly like IN. When the clock is one it will retain the value, now the clock is 0 here again the clock becomes 0 here, so out will just behave like this. Now the clock actually became one, so this will be… Now when the clock is 0 here again, so this will come like this, right? Now the clock became one here, note that this will be like this.

Now here please note that here in this one but that will not be reflected on the out because when clock is one the previous value will be retained, right? When the clock is 0 whatever is happening on the IN will come to the out but when the clock is one whatever just happened

before the clock became one that is what will be retained, so you will be seeing is 0 here, right?

And again the clock become 0, so this will go up you will see all these things and then this will remain now the clock becomes is one, so it will go till this, now the clock become 0 it will fall here and then it will remain, okay. So you see a change in behavior at this path as I mark in orange and also in this path as we have marked again in orange. So now this is an element, the latch is an element which will retain the value for half a clock cycle, right?

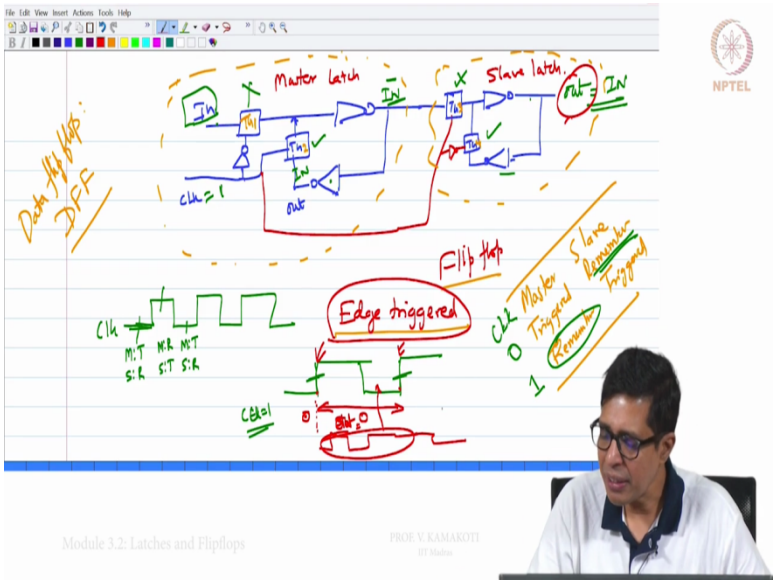
So the clock cycle is between 1 full cycle is between we call it as a positive edge that is weighing from 0 to 1 is a positive edge, 1 to 0 is a negative edge, positive edge of the clock or pasedge and this is negative edge of the clock. So one cycle of the clock is between 1 positive edge and another positive edge, so how does this particular latch behave? Between when the clock is 0 its input follows output when the clock is one whatever was that before it became one whatever the input was there before it became one that value it remembers.

So half the clock cycles this changes and half the clock cycle it remembers, so this is one element which can remember value over a period of time, right? So this is a memory lane, so this element whether it is remembering or not remembering that is dependent on the level of the clock.

Level equal to 0 it actually follows the input and level equal to 1 it remembers the input, so the latch is triggered meaning it starts accepting new inputs only when level is 0, so this is called a negative latch. Negative latch means it is enabled when level is 0 and it remembers when level equal to 1. So latch is a level triggered storage element it depending on the level, it rather remembers or it follows the input, right?

So this is what we call as a latch which is a level triggered element and the latch is basically built using inverter and transmission gates, right?

(Refer Slide Time: 17:16)



Now let us go to the next part of this what we call as the flip-flop. So this is what we are going to see is called the data flip-flop or DFF. Now how does this flip-flop look, if you carefully look at this there is one latch, the latch that we have seen in the previous slide and there is another latch, 1 latch is feeding another latch, right? The left one is called the master latch while the right one is called the slave latch, right?

Now master latch the 1st transmission gates here gets the inverter clock the 2nd transmission, let me call it as TG1 TG2 let me call it as TG 3 and TG4. TG one gets the negation of the clock so while TG 2 gets the clock itself. While on the other slave latch, TG3 is the counterpart of TG 1 in the slave, right? So when master latch is triggered slave will remember?

While the master remembers slave will be triggered, right? So when master slave they work exactly in the opposite when master is triggered slave will remember and when slave is triggered master will remember, right? So this is very clear, now so let us take, so this is the clock pulse, so let us mark this area this is let us say, this is 0 of the clock. When 0 of the clock the master is triggered slave is remembering.

At the one of the clock the master is remembering slave is triggered again at the 0 of the clock master is triggered while slave is remembering and so on. So what will happen is, when the clock is 0 whatever is there on the input it will come as IN bar here. So when the clock is 0 whatever is given as the input that will come as IN bar and when the clock is 0 note that this is cut-off, this TG 3 is cut-off, so slave latch will be remembering something.

We will come to that what is that something very shortly. Now when the clock is 0 IN comes here it is IN bar and of course this is going to be IN again, right? Now when clock becomes one please note that when clock actually becomes one, now this TG 1, right? While TG 3 is ON TG 4 gets cut-off TG2 is on, right? When clock becomes one, now this is what was just before the club becoming one.

The value just before clock becoming one that will that let us call it in that IN bar will be here, as you see here and that IN bar will become in through this not get which am marking in green again through TG2 to come and that will be remembered. So what will the master latch remember? As that we have just showed in the previous slide, just before the clock becoming one whatever was there in the input that will be remembered by the master latch, right?

And the negation of that IN bar will be fetched to the slave latch, right? And what will be the output of the slave latch this is the output, so this IN bar will flow here, so the out will be equal to IN, right? Okay so when the clock actually becomes one then what was there just before the clock becoming one in this input IN as I am marking in green here will now come out here, right?

Now when the clock actually becomes 0 again that value will be remembered by the slave latch, right? Because when the clock becomes 0 this becomes ON TG 4 is ON TG 3 is off this latch remembers that value, so that IN is basically remembered here. The most important thing is when the clock actually becomes one when the clock actually becomes one note that this is cut-off and this is ON.

When the clock is becoming one whatever is happening on the input will not come inside, right? So this master latch is basically not transmitting whatever is happening on the input, so whatever just happened before clock becoming to one that is what is here, so when the clock is actually one whatever is happening on the input will never come to the output. Whenever the clock is 0 whatever is happening on the master latch will not affect the slave latch.

So this essentially, so please understand, so when the clock is one, when the clock is 0 whatever is happening on the master latch the slave latch is not affected. Slave only will remember what happened before. When the clock becomes one? Whatever is happening on the input will not even go inside the master latch.

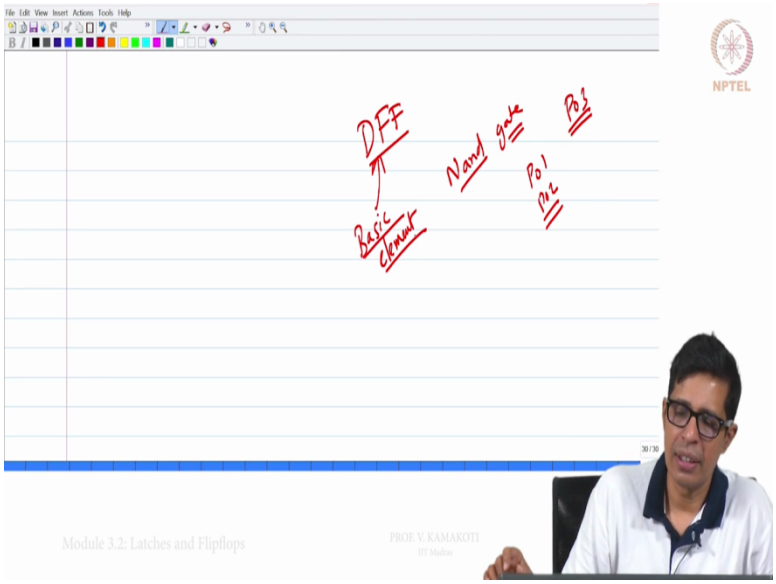
So the master only will remember what happened before the clock became one, so what will essentially reach the slave whatever was the value in the input just before clock becoming 0 that is a value which will go to the slave and that value will be remembered till the next pasedge, right? so the slave will remember what is the IN value between this Pasedge and this Pasedge.

Suppose before this Pasedge the IN value was 0 just before this Pasedge during the entire duration of this clock cycle between this Pasedge and this Pasedge the slave will remember, slave will have the value the output whatever we are seeing here this output will have the value 0 intermediately between this edge and this edge if your input changes or something happened on your input like this nothing will basically go to the output, the output will remain 0.

So just before the positive edge of the clock whatever is the input that will basically get reflected as an output, correct? And in between in any of those periods between the 2 positive edges as we are seeing here if there is any change in your input that will not get reflected at the output. So this is how flip-flop works note that the output of the flip-flop changes based on the value of the input at the positive edge of the clock.

So this flip-flop is called an edge triggered storage element, right? So this is an edge triggered storage element. So this has the capability of remembering the value between 2 positive edges and this is basically constructed using 2 latches one is a master latch and slave latch where in latch is a level triggered element which we explained earlier.

(Refer Slide Time: 26:47)



So as a part of this project the sequential element that you will be using is this D flip flop whose structure I have defined earlier which is constructed using basic transistors but like how you assume that the NAND gate, right? We assume that a NAND gate has been the basic element for company still element combination substitute they are saying that you have done as a part of your project 1 and project2.

Now for this D flip flop which is part of your project 3, now for the sequential circuit which is part of your project 3 we will use the flip-flop as a basic element, right? So D flip-flop becomes a basic element, we will be using this D flip-flop as a basic element to construct lot more sequential circuits. So this is what we like to cover in this module 3.2 on latches and flip-flops.

So to sum up what is a transmission gate? We have seen what is a latch, we have now we also saw what is a flip-flop, D flip-flop and we also stated that we need not construct a D flip-flop as a part of this project but assume that the DFF is available and based on this DFF we will go and construct many more circuits, thank you.