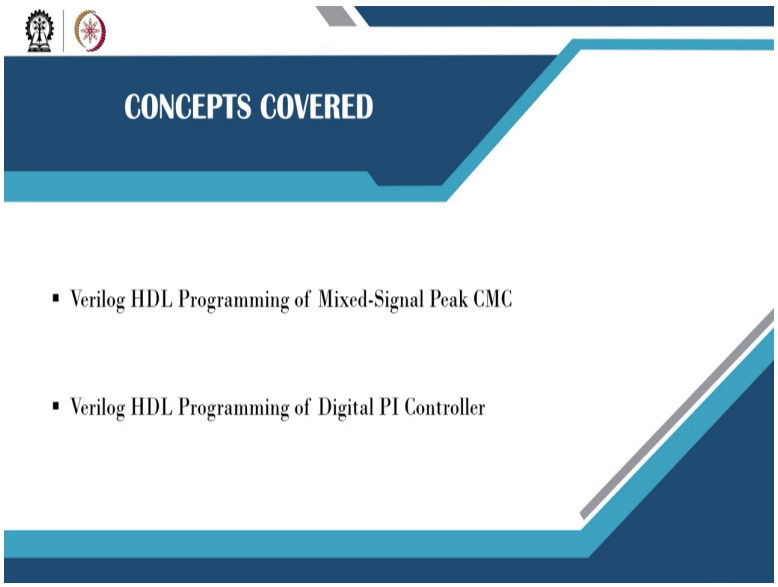
Digital Control in Switched Mode Power Converters and FPGA-based Prototyping

Prof. Santanu Kapat
Department of Electrical Engineering
Indian Institute of Technology, Kharagpur

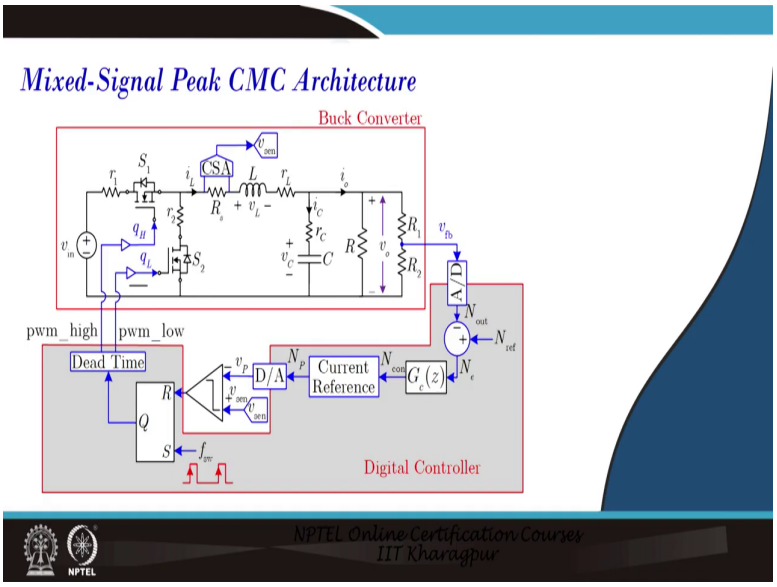
Module - 08
Digital Controller Implementation using Fixed-Point Arithmetic and Verilog HDL
Lecture - 77
Verilog HDL-based Digital PI Control Implementation of Mixed-Signal CMC

So, this is a continuation of the previous lecture, and here we are going to just see the Digital PI Control Implementation and Verilog HDL-based Implementation. We have already learned digital PID controllers. So, PI control will be much simpler.

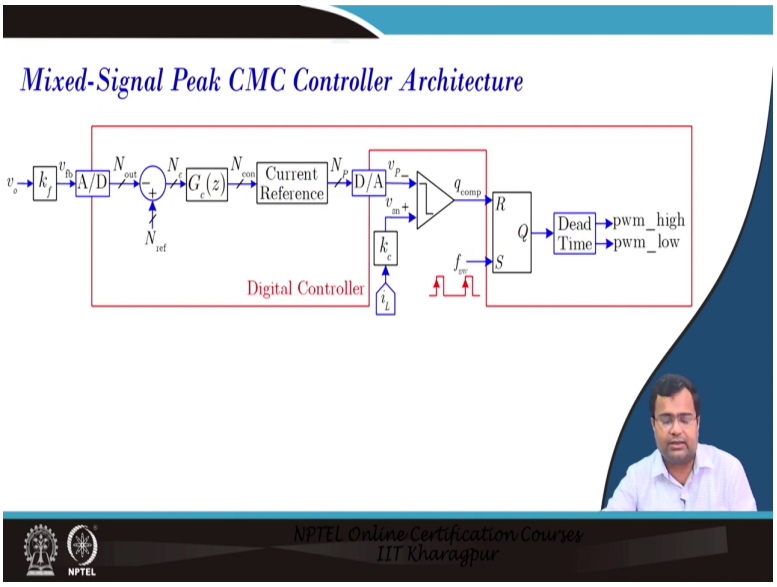
(Refer Slide Time: 00:39)



(Refer Slide Time: 00:44)



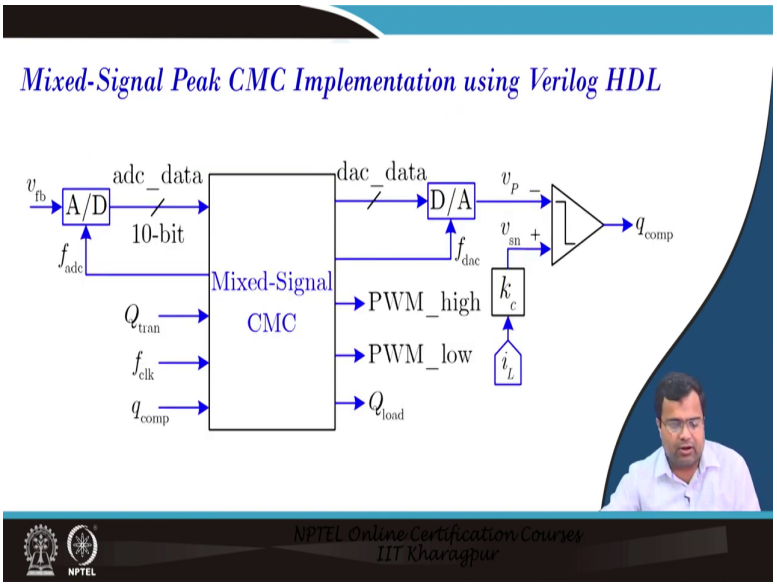
(Refer Slide Time: 00:51)



So, in this lecture, we are first going to talk about the mixed signal current mode control and digital PI controller. So, this architecture has been, we have discussed multiple times; and then, we have also discussed, in the previous two lectures regarding, what are we going to do. Which part is digital? And which part is analog?

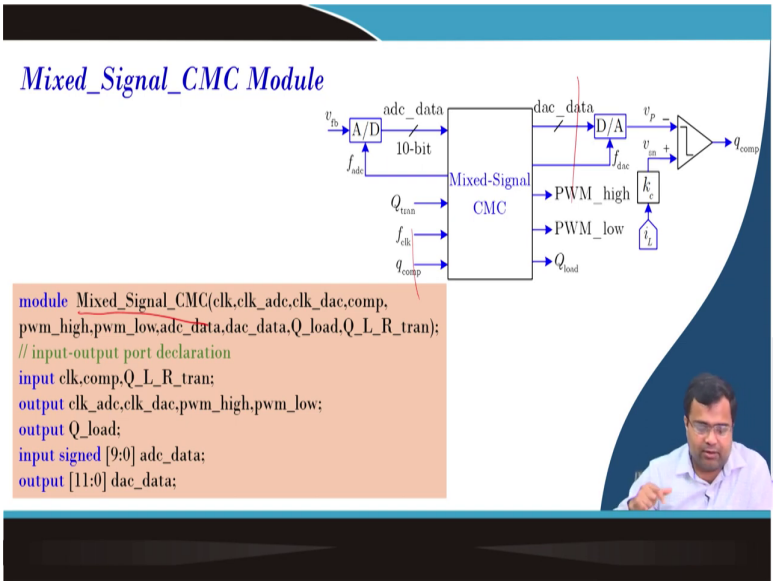
So, this red curve, which is indicating, these are all digital data.

(Refer Slide Time: 01:04)



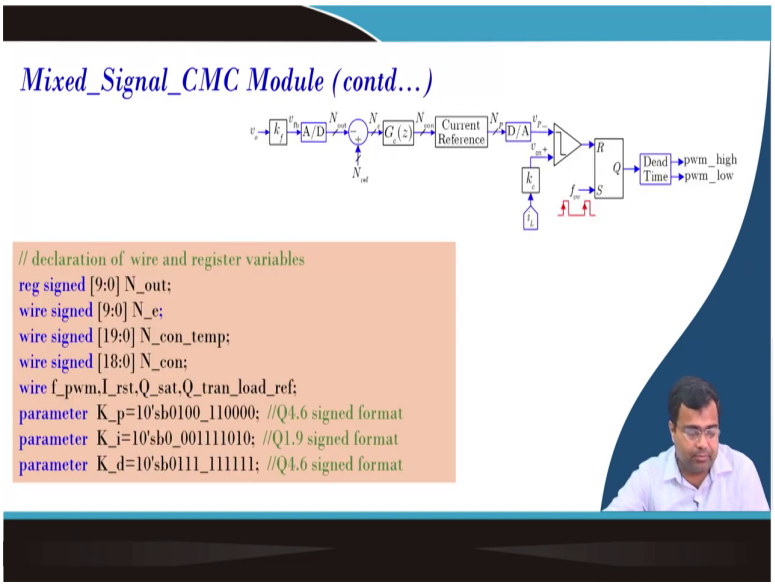
And what we are going to do? We are in FPGA, and we have to implement, in the Verilog module, we want to create this mixed signal current mode control block, which we have also discussed.

(Refer Slide Time: 01:15)



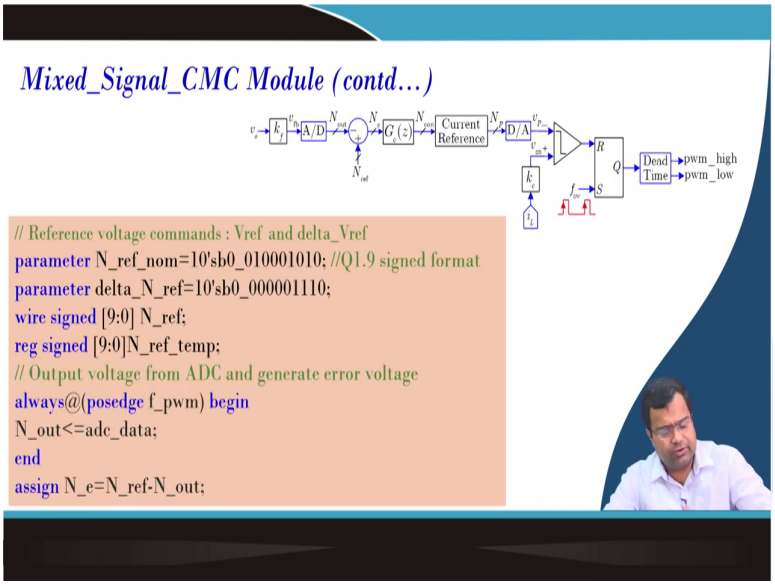
Then, we have also discussed the main module, which consists of this interface, input and output scalar as well as vector input and vector output, and scalar output also.

(Refer Slide Time: 01:29)

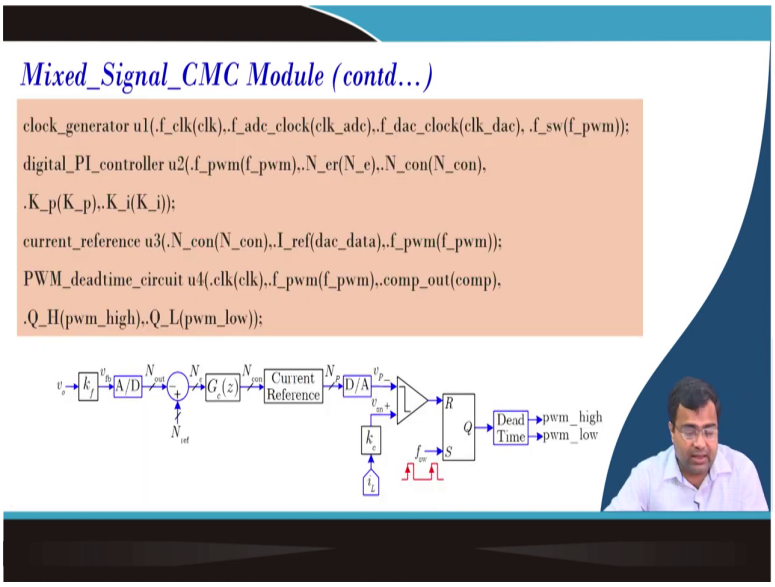


And we have also discussed, that what is the parameter that we are taking. The PID controller is not needed. So, you can remove this; we are just using the PI controller; because, it is the crrent mode control.

(Refer Slide Time: 01:40)



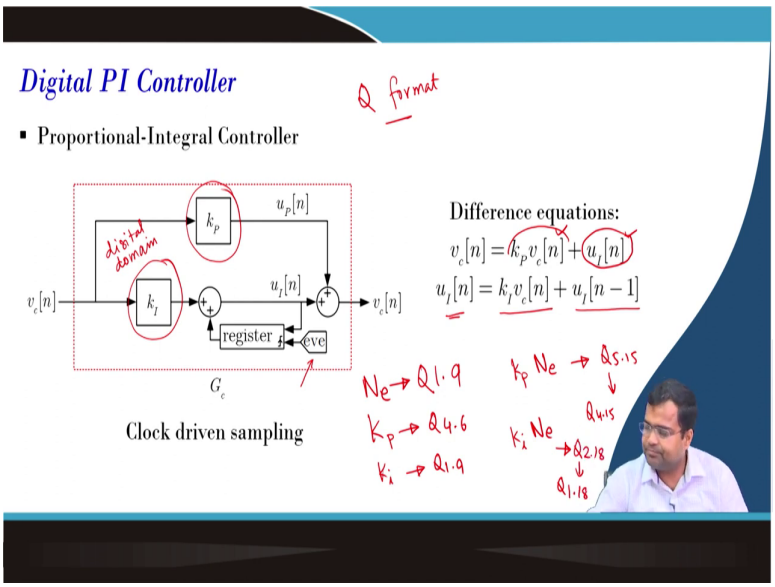
(Refer Slide Time: 01:44)



We have discussed that we are capturing the output voltage and we are instantiation clock generator digital PI controller current reference PWM. So, we have explained this, we have explained this in the previous lecture, we have explained this earlier, in the context of digital voltage mode controller.

Now, we just want to highlight, the digital PI controller, which is in the generic form of a PID controller is also discussed.

(Refer Slide Time: 02:05)



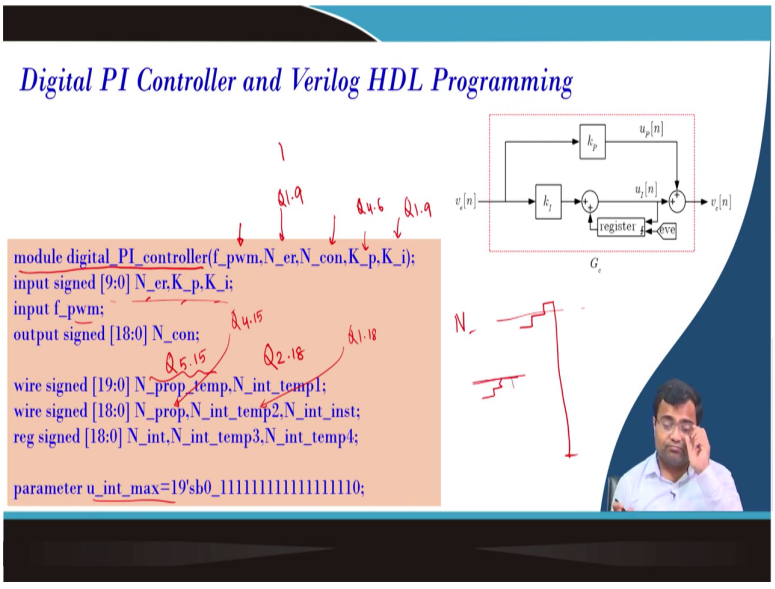
So, PI controller, if you look at, it can be you know we want to update the PI controller with respect to a clock, which can be a switching frequency clock or any other clock. Here, we are not talking about even based PI controller, but we can also implement even based PI controller, which we did in the MATLAB simulation.

So, here we need a proportional gain and the integral gain; and this is all in the discrete domain; it is all in the discrete or I would say digital domain; I would say present digital domain; that means, there will be a number, digital domain and this will be simply number and this we need to know, the Q format, which we have also discussed ok.

So, if we see; that means, this will be overall integral control, this block plus you; this is a proportional control, this is an integral control. The integral control has two parts the history term, plus the instantaneous gain factor ok. So that means, we will talk about Q format; and the error signal, that we will be talking about, is Q 1 dot 9 and which we call an N e; because it is a digital number ok, and K p, if you go back to the previous thing, that K p you are taking 4 dot 6 formats, K i 1 dot 9; that means, we are talking about K p, Q 4 dot 6 and K i, Q 1 dot 9.

So, when you multiply K p into V N e, then as we have discussed, it will be 5 dot 15, but we can resize it to Q 4 dot 15 by dropping 1 MSB; this is the sign bit. Then, we also know, K p; so, this I think better. So, write N e, So, N e into P I, K i. So, this will be, in which format Q 2 dot 18 and which can be scaled to 2 dot 1 dot 18 by dropping 1 MSB.

(Refer Slide Time: 04:34)



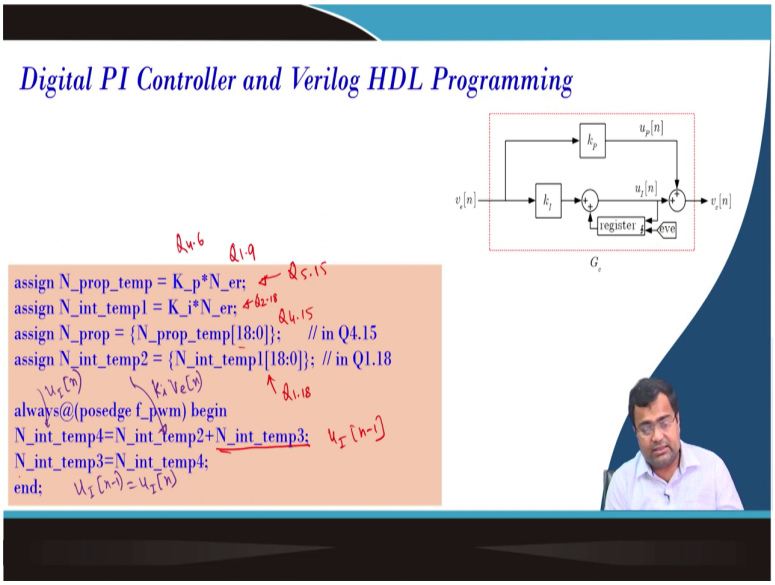
So, this is what, the module name, digital PI controller; here, we need a switching frequency clock, because we need to compute the integral with respect to the clock synchronism, the error signal it is taking, then controller output; then K p, K i; and input as a sign N e r, K p, K i though they are all 10-bit sign, they have a different format.

So, what is N e r? This is Q 1 dot 9; what is K p? This is Q 4 dot 6, and this is Q 1 dot 9, and then, input this is a scalar signal output; what is the N con format? That we are going to discuss. So, first, the proportional temporary is 9; so that means, it is, we are talking about 5 dot 15, 20 bit. Integral; so, this is the proportional case, integral, it is 2 dot 18.

Why it is coming? Because, what is N? I mean, we will discuss this; and then, after resizing, this will be Q 4 dot 15 and we will discuss this, Q 1 dot 18. And, we are setting an upper limit; because, integral saturation is one of the major problems, and we want to make sure that the integral, if it goes below 0 and odd 1, then will become negative; that means, integral; suppose, the maximum value, if you further increase, there can be wrapping error, and it can go all the way down; because, it is a 2's complement number. So, you have to be very careful.

So, we are setting, a limit of the integral; that means, if the integral value, output goes some higher, it will simply hold there, ok.

(Refer Slide Time: 06:35)

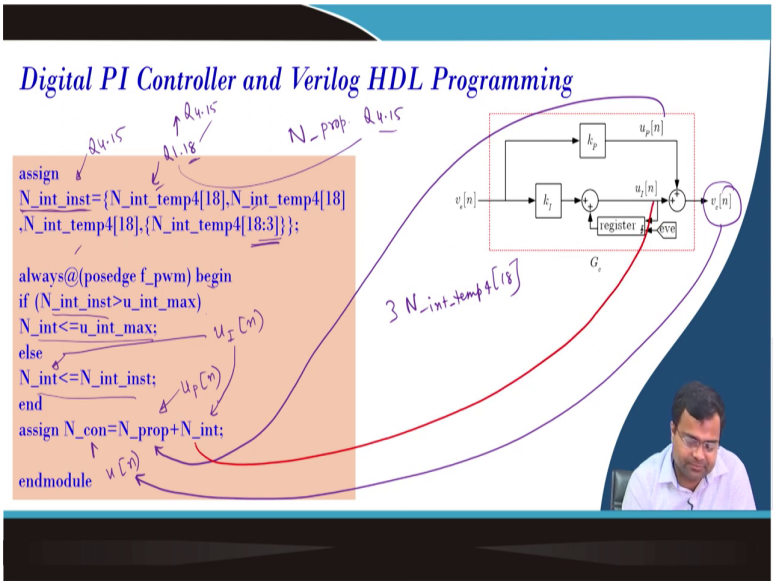


So now, this K p you see, this is Q, 4 dot 6, this is Q 1 dot 9. So, naturally, this will be Q 5 dot 15 and after resizing, this is coming to be; so, after resizing, so, here we are dropping 1

MSB, it is Q 4 dot 15. Integral, this will be Q, 2 dot 18, after resizing, it is coming Q 1 dot 18.

Now, inside the integral, you have that one history term. So, here this is the history term; it is like u, n minus 1; this corresponds. And, this corresponds to like this corresponds to what? So, this particular term is coming from here; which is a resizing of K i. So, it is linked with K i into V error n, right; and this is nothing but your u I n, but we need certain resizing; and here, it is like, u, n minus 1 is equal to u I n. So, the typical standard.

(Refer Slide Time: 07:59)



But, this is just a representation; we need resizing, and then, this is the resizing is happening. So, the instantaneous value, since we have, n this one, was Q 1 dot 18; but, the proportional that we got, for the proportional term; that means, n proportional, we got Q 4 dot 15. So, we need to take this into this format. So, we need to match this format.

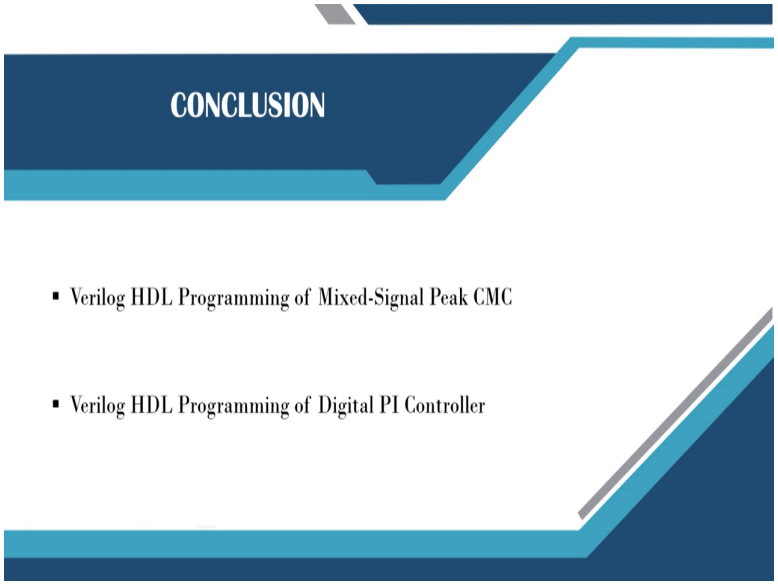
So, we need to convert this Q 4 dot 15, to match this. So, for 4, the, this bit, we can simply discard 3 bits; and this is what we are discarding; 3 LSB, we are discarding. And, for another 3 MSB, we can do a sign extension and this is what we are doing. So, we are manually extending 3 times, but it can be done in 3, 3, 3 times that N initial temp 4.

But somehow, this was not working. So, we just make it, 3 times repeatedly. So, this is your instantaneous value, and this now becomes Q 4 dot 15 ok; but, this is not the end of the story; because, we still have to consider, the saturation limit; that is why it is going inside and it is

checking, whether this instantaneous value is greater than max; then it will be set to max, or it will pass the current value.

So, now this one is integral. So, this corresponds to u I n, and this corresponds to u P n. So, this is your u I n and this is your now, u of n that is your contour output; which we typically consider. So, this is the one, which is this one; and this is the one which is, this one and this integral term, this is the one which is or maybe we will use a different color, we will use, this term is here ok.

(Refer Slide Time: 10:13)



So, in summary, we have discussed the Verilog HDL program, we have mixed signal, and peak current mode control. We have discussed, Verilog HDL programming of digital PI controller. So, in the next lecture, we want to show some experimental results, of this mixed signal current mode control. That is it for today.

Thank you very much.