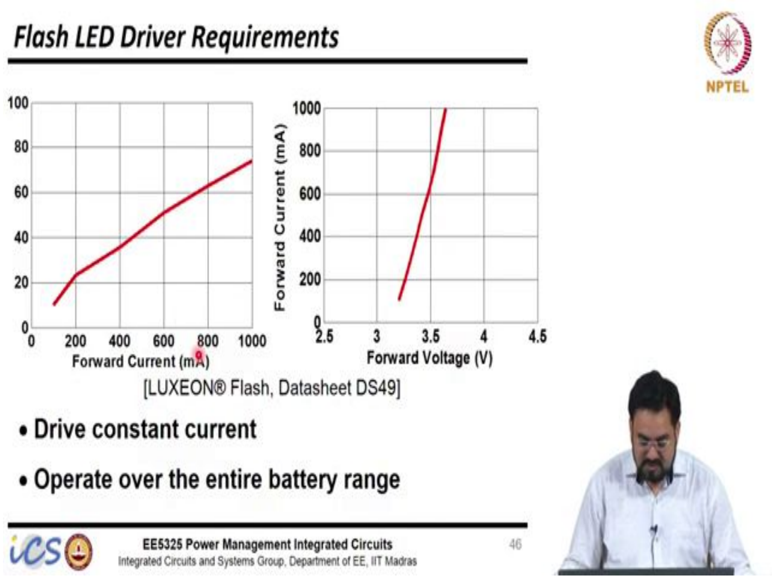
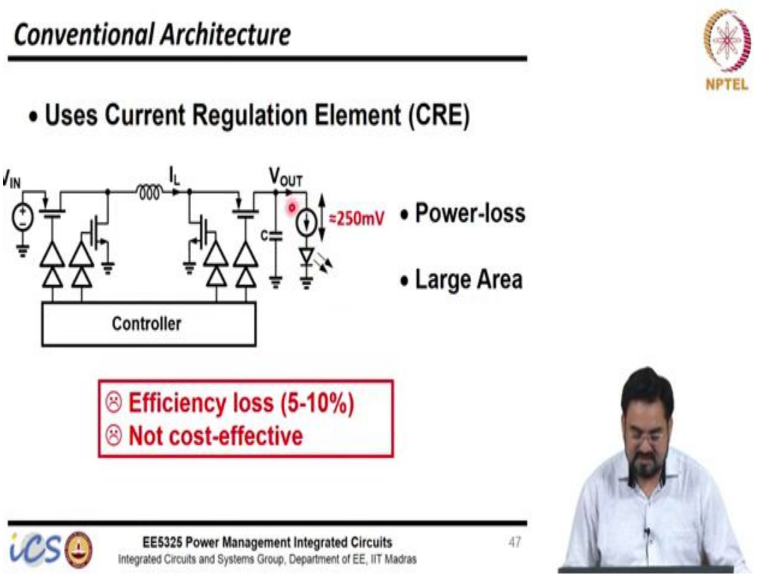
Power Management Integrated Circuits
Dr. Qadeer Ahmad Khan
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture – 101
LED Drivers for Camera Flash



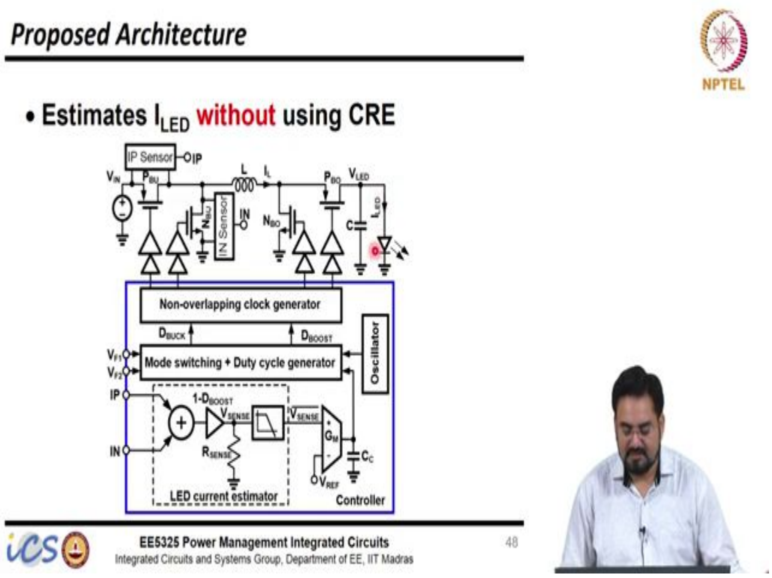
Let us move to LED Drivers for Camera Flash. So again just like we have backlight it requires white LED, but the current required is much higher. If you remember the full scale current in the backlight was 20 milliamp only. Here in this flash, we need much brighter because we are taking a photograph, so the current can go as high as 1 amp, and then we know that it has an exponential relationship with a forward voltage.

At 1 amp current, the forward voltage required will be more than 3.5 volts and it could be slightly higher than 3 volts when we are below 200 milliamps. It drives a constant current and operates over the entire battery range. Most of the time these flashes use 1 or 2 LEDs and you may find even 3 LEDs in very high-end phones, but they are different color LEDs in order to get a more realistic picture. They mix the color and they change the light color when they are flashing.

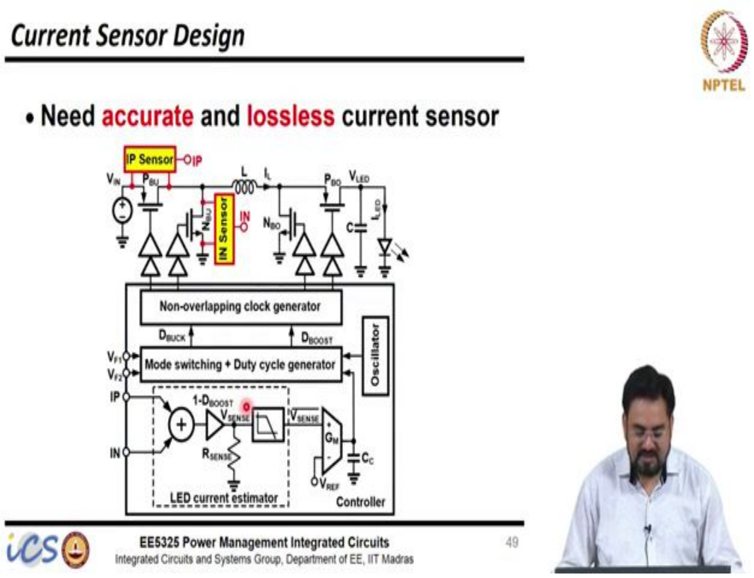


Let us say we are having only a single LED in this flash. So, the way it is done is that we have a constant current just like we had in the backlight. In the backlight, it was a sink and here I am showing a source current. And I already mentioned that you can have either sink or source depending upon if you have the option of using higher voltage, but here we are using a single LED. So, the voltage required won't be like 28 or 30 volts which we had in the backlight, we know the forward voltage is let us say 3.5 to 3.6 max. So, if I drop 250 millivolts here, still this output voltage will not be more than 4 volts or so. So, you can easily use 5-volt devices here and live with it, so there is no issue in using high sight current sources here. But the problem is again we need some dropout voltage for this current source and there will be power loss.

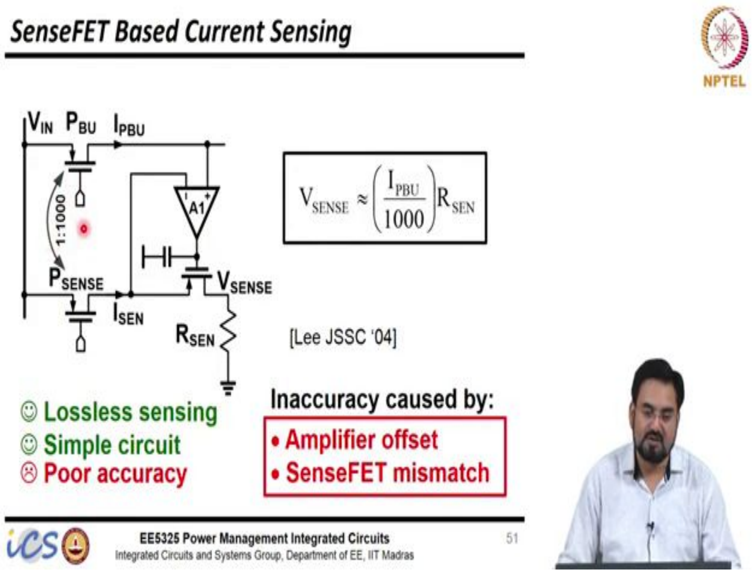
And then again this current source has to supply as high as 1 amp current. So, the device size will be much larger here, the power fets that are used in this current source. We have some efficiency loss here which could be 5 to 10 percent and it is also not very cost-effective because the die size is more and your cost is always measured with die size, so we want to shrink the die area as much as possible.



Instead of having this extra current source, we simply sense the current which is going into the inductor. And we know that the inductor current in the case of the boost converter is I_{LOAD} over 1 minus D and in the case of buck, your inductor current is the same as I_{LOAD}. If I can sense the inductor current, then I can regulate the load current as well. So, all we need to do is just to sense the inductor current and which can be easily done by sensing the current in these MOSFETs, the high side and low side and we can estimate the average current based on that.

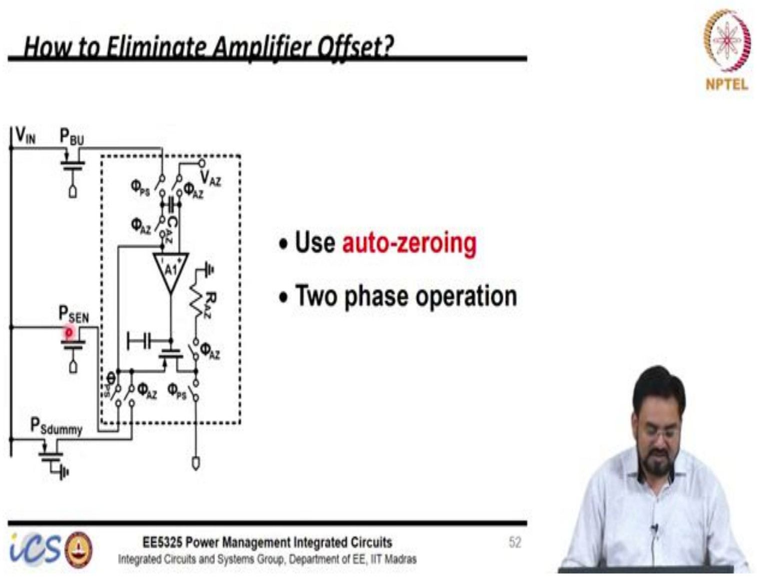


We have two sensors here and once we sense both the currents we can add up the 2 and that will give me the inductor current. And if I am in buck mode then it will be the same as your load current. If I am in the boost mode, then I have to multiply this by 1 minus D in order to get the load current because this will be boosted by 1 over 1 minus D.

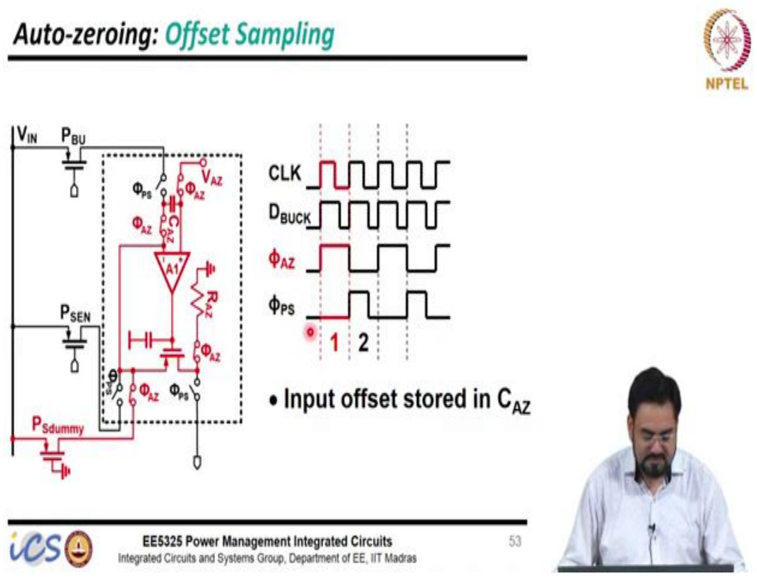


We already talked about different current sensing techniques. If you remember, we talked about lossless SenseFET where you have a smaller FET. And so this is 1 is to 1000 ratio, so if this is 1000 times then this will be only 1 time of that. These devices are linear, we know when they are turned on and if we do not match the V_{ds} then you cannot mirror the current. So, instead of 1 is to 1000 dies scale ratio, it will be something different. So, you won't get the real current number so we have to match both V_{gs} and V_{ds} in order to get the exact 1 is to 1000 ratio. You can dump this current into a resistor and this V_{SENSE} will have the information on your current. Your V_{SENSE} will be nothing, but whatever the current flowing into the high side FET divided by 1000 and multiplied by R_{SEN}.

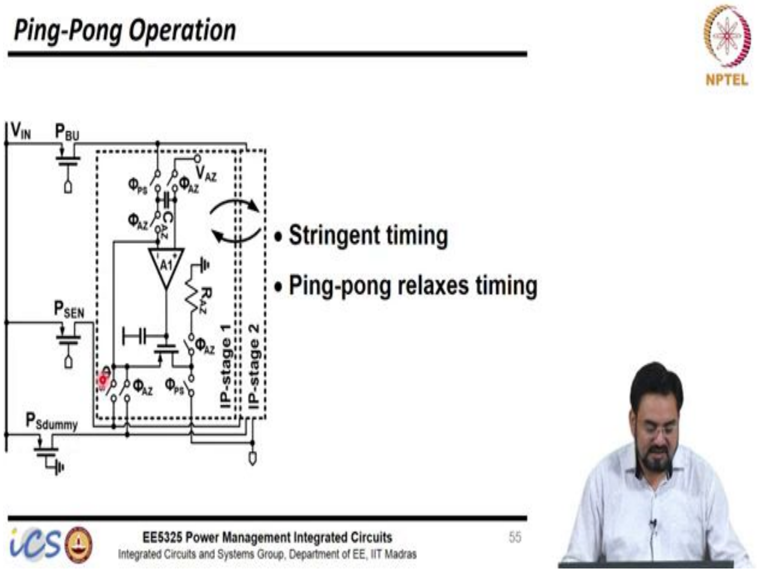
This is nothing but a regulated cascode in order to match the V_{ds} of these two FETs but the problem is that it is highly sensitive to the offset and the mismatch between these SenseFETs. And that is why the accuracy may not be as good as what you achieve with the external sense resistor or when you have a series sense resistor and it also requires a very high bandwidth amplifier because it has to track the ramp up and ramp down of the current. It is a triangular current here not the dc current here.



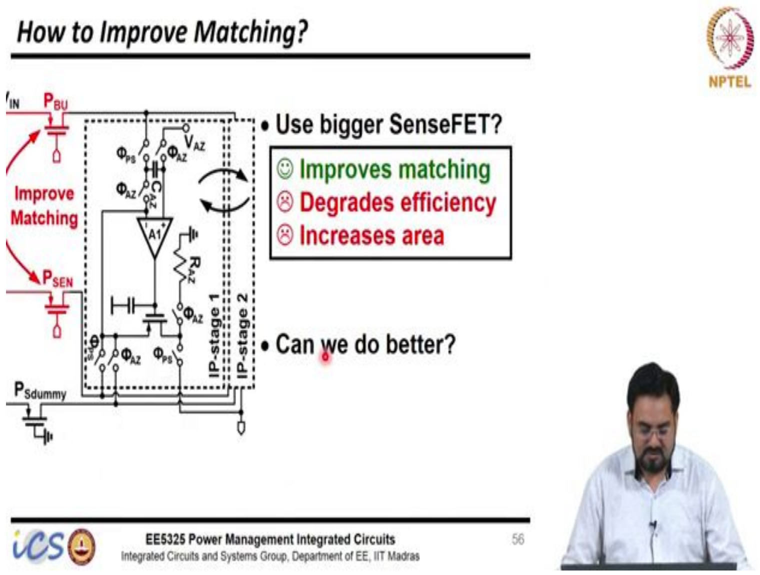
So, in order to eliminate the offset, we use the auto-zero technique and we already talked about how you do auto-zero in zero cross detect. You connect in the unity feedback in order to sample the offset and after that, you cancel it and subtract that from your input and it will get canceled out, so it is a two phase operation actually.



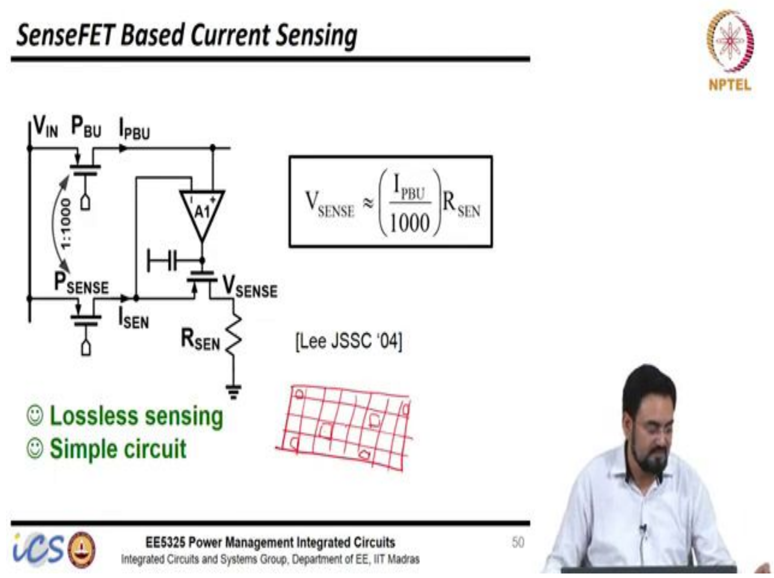
And as I mentioned your input offset is stored in this capacitor(C_{AZ}) and in the next phase, it will get canceled out.



The problem here is you have to do everything within a very short time. So, for the one period of your switching period, you have to do everything. and the next cycle comes then again you have to do everything, it is cycle by cycle current sensing. So, which means we hardly get any time to settle the output, it is quite possible that your output may not get settled and your next PWM cycle will come again. We use ping pong operation in this case. In ping pong, we use two units of this. So, while one is sensing the current then the other you put in the auto zero mode. One is sensing the current and the other is canceling the offset. You need continuous monitoring of the current here and the offset cancellation happens in one phase and in another phase you sample. But here we require continuous monitoring of the current, so you cannot put this in the auto-zero mode. Otherwise, you will miss the current.

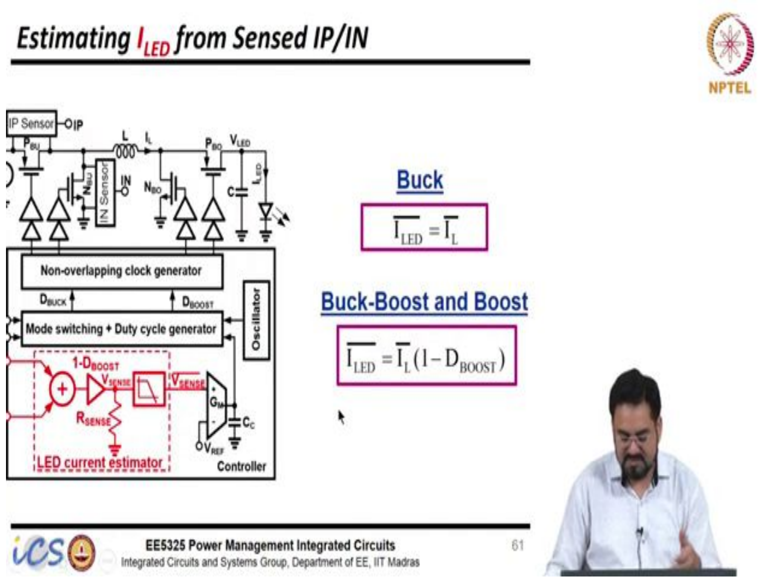


Offset cancellation also improves the matching because we are canceling the offset so accuracy will be higher. One more thing you need to remember is that the ratio of these FETs is 1 is to 1000. If you use let us say 1 is to 100 instead of 1 is to 1000, then the matching between these two devices will be better. But SenseFET has to carry more current in case of 1 is to 100 ratio and this current is completely lost because you are not utilizing this current. and is getting dumped into the resistor. In order to improve accuracy, we have to sacrifice the efficiency here by increasing the size of this SenseFET.

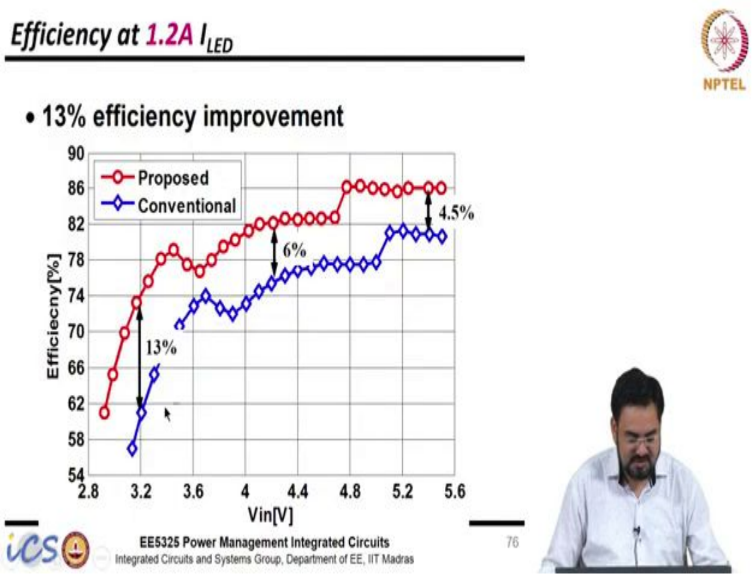


Let us see how we can improve accuracy without degrading efficiency. Let us say I have 10 devices of the same size for SenseFET. If I connect all the 10 in parallel then I will get 1 is to 100 ratio, but instead of doing that what I will do, I will turn on only one unit at a time. But in one period I will move across all the FETs. Let us say your 1 clock period is 1 megaHertz frequency which is a time period of 1 microsecond.

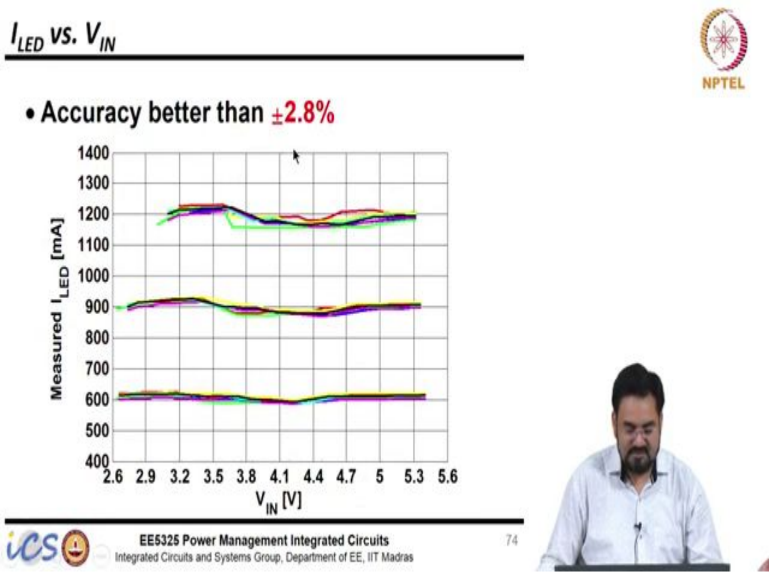
And I switched these FETs on and off 10 times and I have 10 of these. So, in 1 clock period, I can cover all the 10 FETs, so I turn on one SenseFET for one tenth of your period, then next one tenth period you connect the next one. Similarly, you rotate through all the FETs. So, what you are doing here is that you are moving through all the FETs, and in order for better matching they are scattered or spread all over the places within your bigger power FET. So, ultimately, You are improving the matching and you will get much better accuracy compared to what you get with the single unit of 1 is to 1000. And this is done without increasing the current because I am turning on only one FET at a time.



After sensing how you scale, you are multiplying by 1 minus D_{boost} ratio.



So, the main thing I want to show here is an improvement in efficiency because we do not have a current source. You can see 13 percent of improvement inefficiency.



This is accuracy plus minus 2.8 percent at different current levels. Most of the time the accuracy requirement is less than 5 percent or so. So, it is within the range.