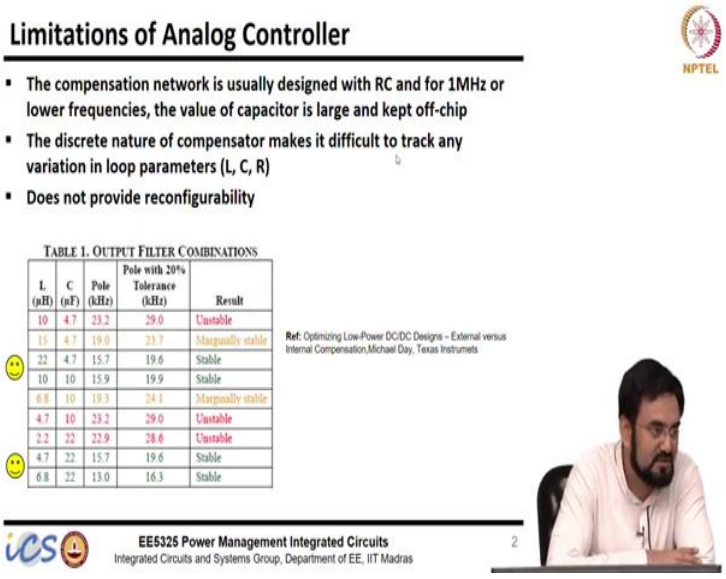
Power Management Integrated Circuits
Dr. Qadeer Ahmad Khan
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture – 92
Digital Control of DC-DC Converters, ADC Architectures



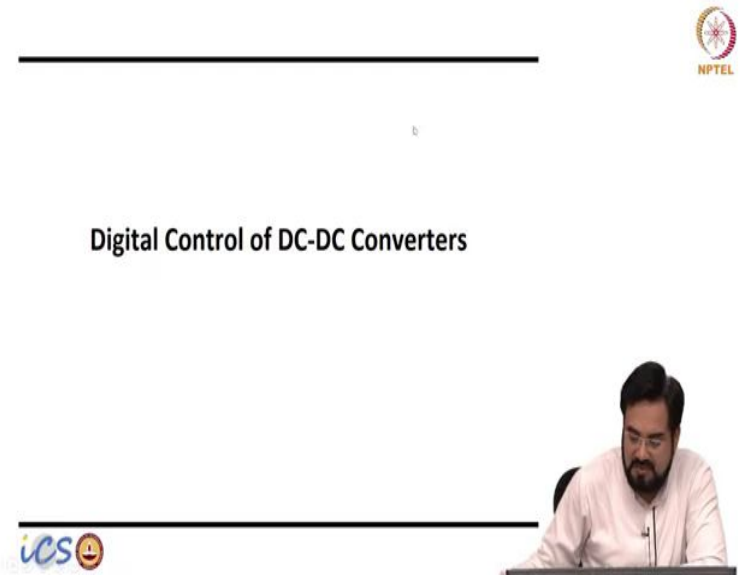
There are limitations in using the analog controller. The compensation network requires capacitors and for lower switching frequencies the value of the compensation capacitor is high which will take a lot of on-chip area or we have to off-chip capacitor.

The discrete nature of the compensator makes it difficult to track any variation in loop variation. We can use R-DAC to change the value of resistances and to change the value of the capacitor we will require large switches because we do not want to add ESR to the capacitor. So, changing the capacitor is not a good option, we can change R, but there will be some limitations with R as well. We can not independently tune zeroes and gain in an analog compensator.

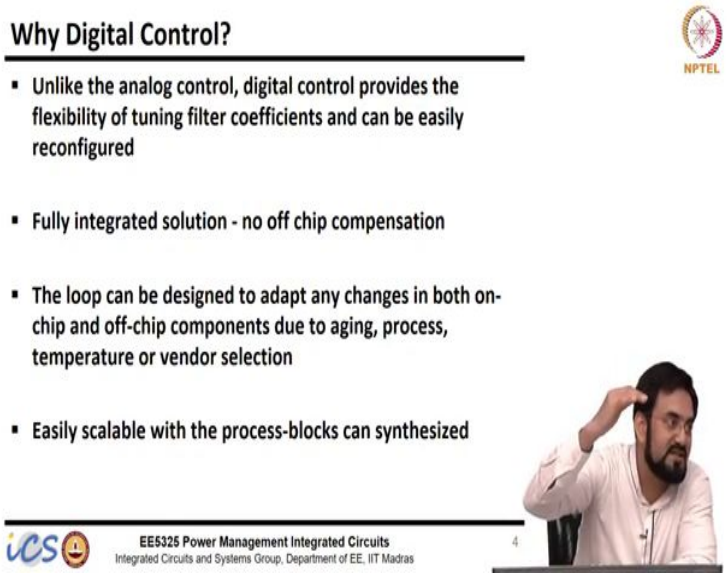
Analog compensator does not provide reconfigurability i.e. it does not support multiple values of inductor and capacitor in the converter because we have designed the compensator

for a particular value of LC frequency and converter may become unstable on changing the value of L and C.

So mostly when we design, we simply look at the tolerance which is plus-minus 10 to 20 percent and we design for that. But if we have to cater to 200 percent, 300 percent kind of a variation in L and C, then it will not work unless we change the compensation parameters which is the gain and location of zeros. In table 1 in the above image, we can see the stability of the converter for different values of L and C when we have designed the converter for the poles values of around 15kHz. We can see that the converter is marginally stable for poles values of around 19kHz i.e there will be some ringing in the output during settling and the converter is unstable around when the poles are around 24kHz. Analog controller can tolerate the variation of 10 to 20 percent but when the variation goes above 50 percent then the converter will become unstable.



To address the limitations of the analog controller, we go to digital control DC-DC converters.



Unlike analog control, digital control can provide the flexibility of tuning filters because everything is digital here, we can program easily. The coefficients can be programmed and we can easily change the gain and the location of zeros.

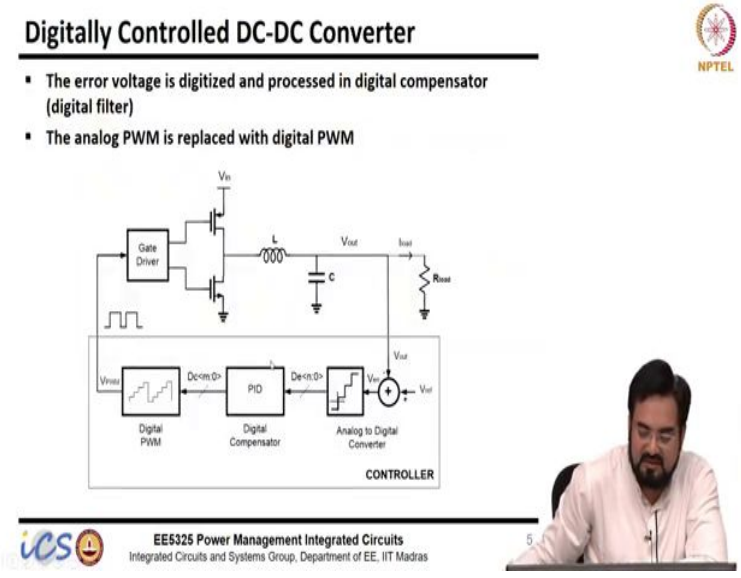
In power electronics, the switching frequency of the dc-dc converter is in order of 100s of kHz because Power FETS are very large because their current requirement is very high. So they cannot operate at the order of megahertz. In that case, they mostly require off-chip capacitors because the capacitor may be in the order of nanofarads and it is difficult to put it on-chip.

They simply use a controller chip and put a compensation network around that. We can assume that the op-amp is inside the chip but the RC network around that will be off-chip. Based on the values of L and C they choose the components value for the compensation network. So with the external components, we can change the gain and location of zeros but then we have to pay something because off-chip means cost is involved here and it will take area on the board also.

But if we use digital control, then it is fully integrated and does not require any off-chip compensation, only off-chip component will be your L and C filter. Obviously, we cannot integrate that microhenry and microfarad kind of inductor and capacitor on-chip. So it has to be off-chip.

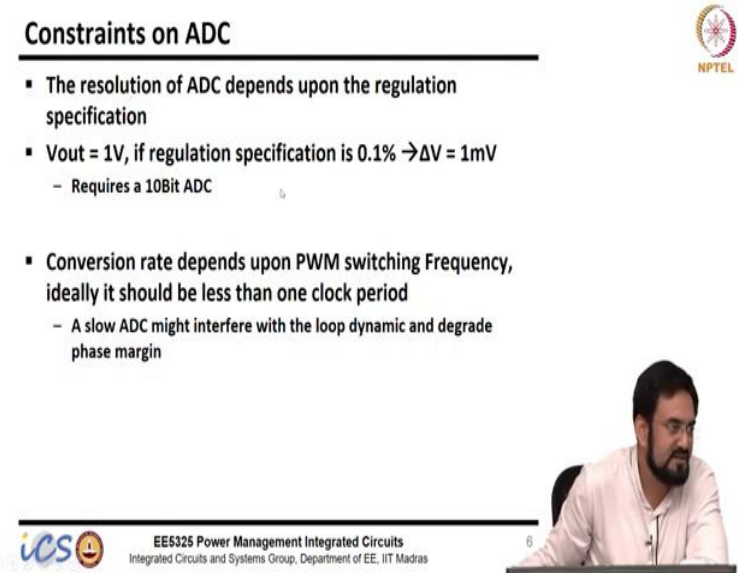
In digital control, the loop can be designed to adapt any changes in both off-chip and on-chip components due to aging, process, temperature, or vendor selection. For example, an inductor can have tolerances of ±20%, and inductance also reduces on changing the load current. So if we consider everything variation maybe is easily 50 percent or so and if we have not overdesigned or over-compensated our DC-DC converter then it may become unstable. And one thing we have to remember whenever we overcompensate anything it will be slow in nature because now we are looking for a higher phase margin and that we can only achieve by reducing the bandwidth. But in digital, we can measure all the parameters digitally and tune our compensation parameters accordingly.

Another advantage of digital control is that it is easily scalable with the process. In analog control, if we change the process then we have to redesign the whole compensator again because for the same size of the transistor all the values in another technology for example gm, rds, etc. will change. There will also be a change in sheet resistivity and capacitor per unit area. Redesigning the whole compensator again is a time-consuming process. But in the case of digital control, we just have to take the Verilog code or the VHDL code and synthesize it. It does not matter what process we use, the Verilog code will not change because the Verilog code is nothing but behavioral. And when you synthesize it simply picks the logic gates from that particular process library and it will synthesize and convert your code into the circuit.



In digital control, we have error voltage just like in analog, now you have to digitize it which means we have to use ADC. Now we get the digital code which is nothing but an error that is represented in binary. And then we have the same PID, but it is digital now. Earlier, we had analog which is continuous time, now we have digital compensator. And then this will give us again the control code which we were calling V_{ctrl} in analog and comparing with ramp.

Now we need digital PWM here. So that is a difference. So everything is digital here, with no analog component except ADC. The moment we get the digital code, everything has to be processed in digital. And then this will eventually give us V_{PWM}. So, from V_{err} to V_{PWM} there should not be any difference between analog and digital, in between how we process that is if we look at a black box. So as long as our end results are the same, we do not care whether it's implemented in digital or analog.

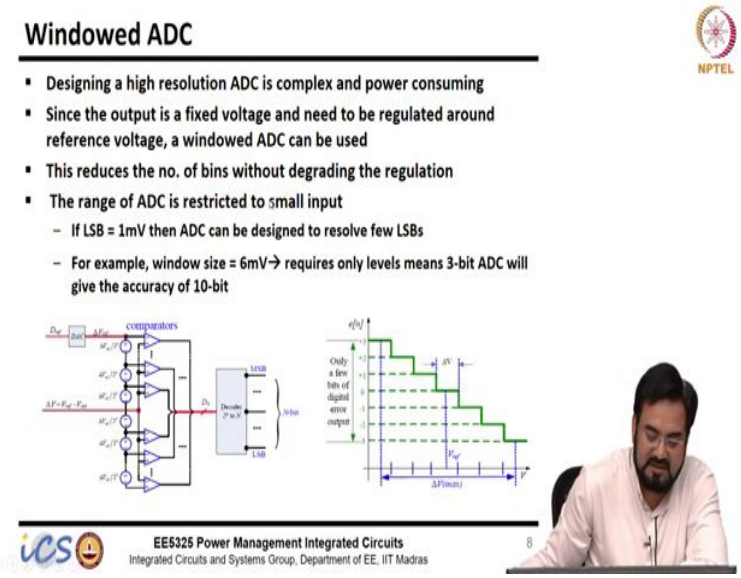


There are certain constraints on ADC. The main thing is the resolution here. For example, if V_{out} is 1V and regulation specification is 0.1% which means we require accuracy of 1 millivolt then we will need ADC of 10-bit, 10-bit corresponds to 0.1 percent.

The conversion rate depends on the PWM switching frequency. Digital code should be ready when the next PWM cycle appears then ADC should operate at switching frequency. Or in most of the time in order to cater to different transient responses, we usually design

asynchronous control where it can respond right away during the transient but in steady-state, it will be synchronized with PWM.

Suppose our switching frequency is 1MHz. So, designing a 10-bit ADC at 1MHz or beyond frequency is not easy. It's going to take a lot of area and a lot of power. The main thing to go to digital is that we want to save area and power. If we are not able to save area and power then it's not benefiting us and does not make sense to go to digital. If we make a slower ADC then we are forced to reduce our bandwidth and the pole which will be occurring due to slower ADC will start interfering with our loop and phase margin will degrade.

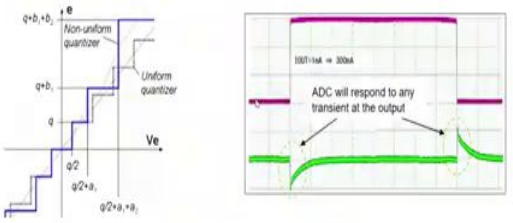


There are different ADC architectures that we will try to overcome these limitations where we require a high resolution ADC.

Ultimately, what we require is Δv of 1mV and it does not mean that we will always require 10-bit ADC. So when the transient occurs, deviation of V_{out} from its original value is a fraction of its original value for example 50mV to 100mV. So we do not need a 1V dynamic range here. The maximum undershoot and overshoot we are expecting we only need to resolve that. So, our range is limited here. If we truncate the range from 1V to 100mV then we will only require a 7-bit ADC.

Non-Uniform Quantization

- A windowed ADC saturates under any transient at the output hence loop goes into non-linear – poor transient response
- A non-uniform quantize can be used. The quantization level is small around zero error (Vout = Vref) and increased gradually as Vout move away

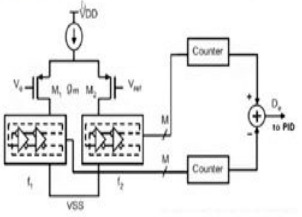


Ref: Haitao Hu et al, *"Nonuniform A/D Quantization for Improved Dynamic Responses of Digitally Controlled DC–DC Converters"*, IEEE Transactions on Power Electronics, July 2008

We also do not need to keep these steps linear. Whenever output voltage is near to 1 volt, we can reduce step size to 1mV and when the output voltage is away from 1V then we can increase the step size because that time we are responding to transient and accuracy requirement is in steady state. So in a steady state, we are regulated at 1V. For example, if we have to cater to a range of ±50mV, then we may require 7-bit ADC but if we use non-linear steps then we may be able to do in 3 or 4 bits ADC only.

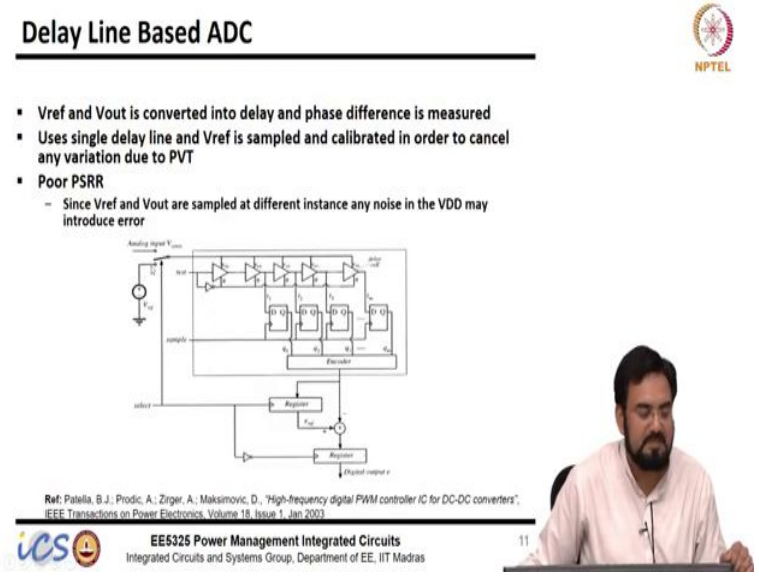
ADC Architectures - VCO Based ADC

- The VCO is used as the Active load of transcunductor
- The two VCOs operate at same frequency if Vout = Vref
- Any error in the output voltage causes diffrence in the two frequencies
- Digital error can be generated by counting the no. of VCO clock cycles
- Overall gain = $A_{gm} \cdot K_{vco}$ → provides good DC regulation



Ref: Xiao, J.; Peterchev, A.; Zhang, J.; Sanders, S. *"A 4μA-quiescent-current dual-mode buck converter IC for cellular phone applications"* IEEE International Solid-State Circuits Conference, 2004. Digest of Technical Papers. ISSCC-2004

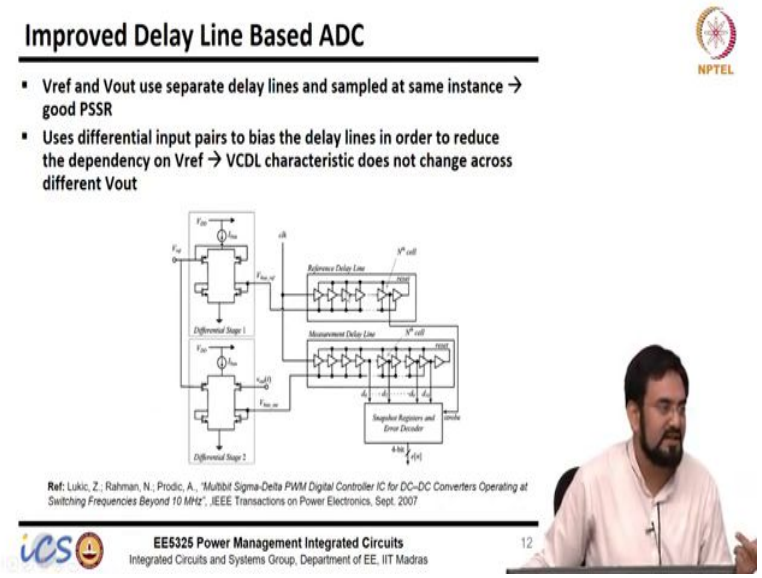
Then another architecture of ADC is based on VCO. The circuit of ADC is in the above image and used ring oscillators as VCO. In the above circuit, the difference between V_{out} and V_{ref} will generate a difference in f_1 and f_2. Correspondingly, we will get a difference in counter output and their difference will give us the digital error. This is a very simple ADC that we can design in very low power and it does not require any capacitor and resistor. The only problem is the linearity because these VCOs are not perfectly linear but we are looking only for the ±50 millivolt kind of a range and within that range, we may get decent linearity.



Another architecture of ADC based on the delay line is shown in the above image. Delay line ADC is similar to flash ADC. In flash ADC where we have multiple comparators and each comparator compares the input voltage with the references and we get an output code. For all the comparators where the input reference is below the input voltage or output, we will get the output as one. Similarly in Delay line-based ADC, the delay is a function of input voltage, and if input voltage changes then the delay will change and we are sampling this delay using the D-flip flop. Each flip flop will give output one when delay crosses a particular value, similar to the flash ADC comparators. D flip flops will give different codes for different input voltage and we can decode this code to binary using an encoder.

The same delay line is sampling both reference and input and in one cycle we will register the reference voltage digital code and in another cycle, we will register the code for input voltage, and then we will get the difference between both to get the error. Digital code for

reference voltage is not going to change but the reason we want to sample every time because the delays might change because of PVT and we will get a different error for the same value of input voltage. So it will keep calibrating basically and will keep updating the reference code based on any variations in the delay cell. That is why we are sampling both input and reference.



The problem with the previous delay line ADC is that we were using the common delay line and let us say we have sampled the reference and next time when we are sampling the input, if the delay is changed between those 2 samples then it will give an error. But here, we are assuming that when we sampled the reference and after that when we sample the input voltage, the delay is not changed. But in between the time when we sampled the V_{ref} and then input voltage, if there is any change happens let us say due to temperature. In that case, we will get an error.

If we keep continuously sampling the V_{ref} then this problem would not happen and for that, we will need two delay lines. The above image shows the architecture of the improved delay line ADC. So what we are doing here, one delay line is sampling the V_{ref} and another delay line is sampling the input voltage. The output digital code of both delay lines is used to calculate the error code using snapshot registers and error decoder. We have used differential pairs so that we can get the same common-mode voltage.