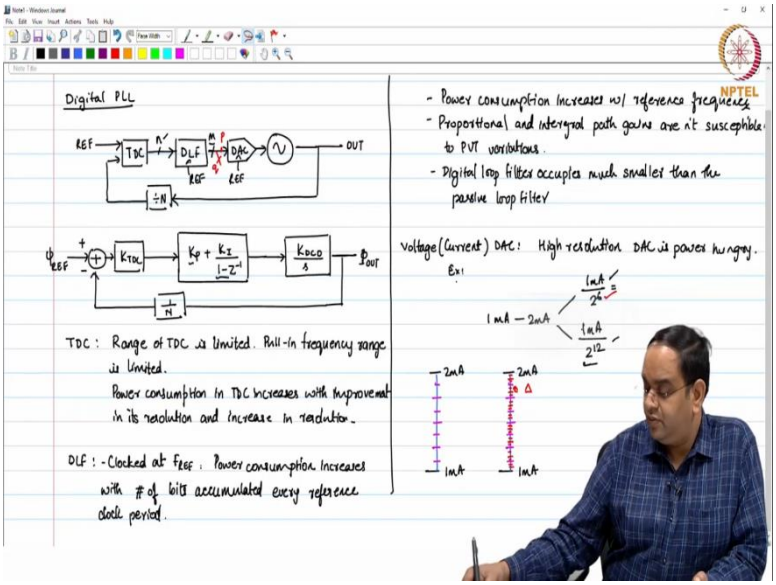
Phase-Locked Loops

Dr. Saurabh Saxena
Department of Electrical Engineering
Indian Institute of Technology Madras

Lecture – 62
Noise Analysis in Digital PLL

(Refer Slide Time: 00:15)



Hello everyone. Welcome to this session. In the previous session, we looked at the block diagram of the digital PLL and the equivalent small signal model. We also found the loop gain, the unity gain frequency and the phase margin of the digital PLL. So, today, in this session, we are going to look at the apparent problems which can come up with the digital PLL and how we are going to resolve those.

So, you have the block diagram of the digital PLL which we have been using, it is like this. You have a time-to-digital converter whose output goes to digital loop filter. So, I will, better I can just write that as DLF, Digital Loop Filter. The output of the digital loop filter goes to a DAC. DAC controls the VCO. The output of the VCO is fed back with a divider here and this is your reference clock.

Now, the small signal model of this particular PLL was shown like this. You have error for the TDC defined by the gain is K_{TDC}. The output of this TDC goes to the digital loop filter whose gain is in z-domain. It was $K_P + \frac{K_I}{1-z^{-1}}$. Here $z = e^{sT}$, where T is the reference period. This

goes through your DAC and then the DAC combined with the VCO gives you a transfer function like $\frac{K_{DCO}}{s}$. The output of this comes back with a feedback ratio $\frac{1}{N}$. This is what we had.

So, in terms of phase, I can write this as φ_{REF} and this is φ_{OUT}. So, in the previous session, we looked at the loop gain and the phase margin and we know this is like a second order system with two zeros and one pole and it is stable. Stability is not an issue. So, let us look at each of these blocks and what do they do in terms of the noise which they add at the output or if there is any limitation.

So, time-to-digital converter, we know, it measures the phase error between the reference and the feedback signal. And the range of phase error which it can compute, it is limited by the delay line or the time for which the delay line is tuned which is used in the TDC. So, quite often the range of TDC is limited.

If the range of TDC is limited, then the phase error which it can measure is limited. And if the phase error which it measures is limited, then the pull-in frequency range is also going to be limited. This is what you will see. Now, what stops us in extending this range? Well, at one time, we want the resolution of the TDC or the smallest period which the TDC measures as the phase error, that should be as small as possible. Now, if we make that error small and you want to extend the range or even if you want to have the range $\pm 2\pi$, the power consumption, it is clocked at the reference period, power consumption in TDC increases with improvement in its resolution and increase in range.

So, this is a problem. It is not straightforward that TDC measures the phase error exactly what your PFD was doing. No, it comes with issues. Now, on top of that, the TDC output if you are looking at, the TDC output depending on how many bits you are going to measure. This TDC output will come with n bit. In the digital loop filter, digital loop filter is clocked at your f_{ref}. If the number of bits which it is using to accumulate, proportional path gain is one thing, accumulator is the other block.

So, when you have a digital accumulator, if you have like 20-bit accumulator in comparison to 12-bit accumulator, the power consumption of the 20-bit accumulator will be more. So, here the thing is that for the digital loop filter, power consumption increases with number of bits accumulated every reference clock period. It is like that.

And it also increases with the reference frequency. Power consumption increases with the reference frequency. This is in contrast to the passive loop filter. The passive loop filter itself does not consume any power because that is made up of passive resistor and capacitor. The only power consumption was in the charge-pump.

But here the loop filter which you are implementing, that will consume power because it is accumulating every reference period and the number of bits it accumulates, that accumulator which you are implementing in digital form, that is going to consume power proportional to the reference frequency.

The good part of this is that you can say the proportional and integral path gains are not susceptible to any process, voltage or temperature variations, PVT variations. You can change them as you like. So, earlier, if there is a process variation in case for the resistor or capacitor, these resistor and capacitor will vary. They will change your zero location.

Here, K_P is just a digital gain which you are implementing digitally. It will not change with the process corners. So, this is the good part. Surely, the digital loop filter occupies much smaller area as compared to the RC filter. So, these are all the good things which you know about the digital PLL.

Now, when you get a certain number of bits at the output of the digital loop filter, the bits are like m and you cannot drive the DAC. So, this, it is like, it can be voltage or current DAC depending on how you are controlling your oscillator. So, voltage or current DAC, this is also clocked by the way at the reference period. This is clocked at the reference period. TDC is anyways clocked at the reference period.

So, DAC converts your DLF output to the control voltage or current for the oscillator. It cannot implement the number of bits which you are having. If you increase the gain, you increase the number of bits in the accumulator. You will have lot many bits and implementing large number of DAC cells, it will face linearity issues. So, you can say high resolution DAC is power hungry. So, what it means is, let us say, this just an example.

Let us say, I want to control the current from something as 1 mA to 2 mA. This is the range of the current variation. I can do this in two ways. One, I can have a resolution of 1 mA. This is

that the $\Delta = 1$ mA. So, I can have a resolution in my current DAC as proportional to $\frac{1\text{ mA}}{2^6}$. I can also have $\frac{1\text{ mA}}{2^{12}}$.

So, the resolution here is much coarse in the first case as compared to the other. But you need more bits, more current cells, you want to maintain the linearity. As you want to maintain the linearity, a high resolution DAC is going to be much more power hungry. So, what happens normally is that these m bits which you are looking here, these m bits which you have, these m bits are dropped here, m bits are not fed directly to the DAC. You have p bits here and you have q bits here.

So, you have a gain factor involved here and when you have this gain factor involved while implementing the DAC. So, it is like you want to control in steps of $\frac{1\text{ mA}}{2^{12}}$. But you are not controlling $\frac{1\text{ mA}}{2^{12}}$, you are controlling by $\frac{1\text{ mA}}{2^6}$ in that resolution. So, there is a quantization which happens while the range will still remain the same.

Earlier, you can say 12 bits span your 1 mA. Now, maybe 6 bits span 1 mA. So, when that happens, there is a quantization involved and this quantization leads to the quantization error in the DAC which is going to change the frequency of your oscillator in a much more coarse step which is an error.

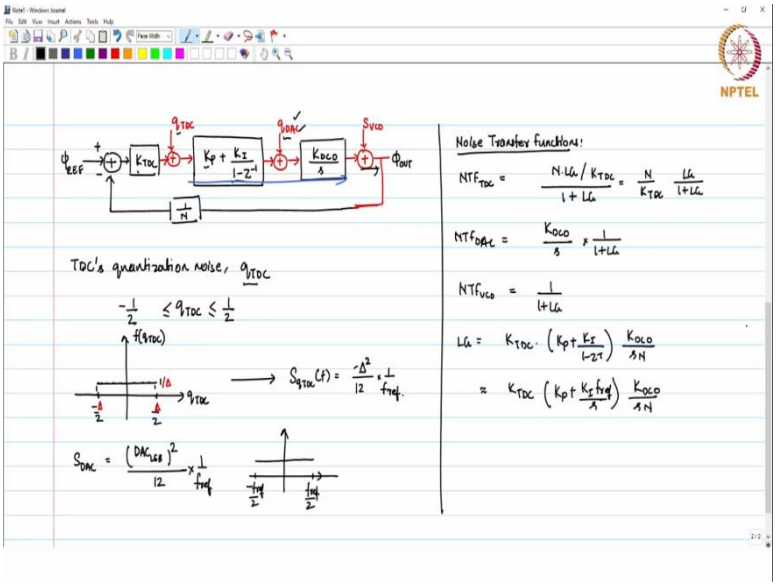
So, to be clear, this is you just try to understand this. You have 1 mA and 2 mA. So, this is the range you want to cover to control the oscillator frequency which is coming from your number of bits from the digital loop filter. In one case, you are having much coarse step like this, this is just an example.

In the other case, you have even fine steps like this. This is only half of it. So, I can have something in between. So, when you have so many steps in between, what happens is that the Δ between each step is much smaller which is like a lesser quantization error you are going to have when you convert the digital word from the loop filter to the control voltage or control current.

The input of the DAC you add the quantization while controlling the frequency of the oscillator. So, that is the quantization error which your DAC is going to see. So, now, given this, well, it is power hungry high resolution, so, you can have a lower resolution, you can adopt some other ways. We can look at it later.

But that is the problem which you will see. So, now, what we are going to do here is we are going to find that what are the noise sources in this digital PLL similar to the noise sources in the analog PLL and try to see how they get filtered by the PLL. So, this is your small signal model of the digital PLL.

(Refer Slide Time: 15:27)



And we will find out or we will identify the noise sources. So, the first noise source which you remember with the TDC because TDC is some finite bit TDC, you are going to add quantization noise at the TDC output. Similarly, at the DAC input, you are going to add quantization error for the DAC, q_{DAC}.

And then the VCO itself whatever noise you were having, phase noise of the VCO, that still remains. So, you have VCO phase noise, S_{VCO}. So, this is something which you can always write. It is always there, no matter whether it is analog or digital. Now, this quantization noise, now, there are certain assumptions here that quantization TDC quantization noise. We know that at any given time, q_{TDC} is limited and q_{TDC}, the quantization noise is going to be limited between your $\pm\frac{1}{2}$ LSB of the TDC.

So, that is what we have. So, if you remember from the last session, you have a phase error. K_{TDC} get that convert that phase error into number of bits and then you have $\pm\frac{1}{2}$ LSB which actually adds to the output of the TDC.

So, this is what you have. At any given time, the quantization noise is limited. So, here if I would like to call it, I will say this quantization noise under the assumption that you have noise in the system, the noise of the VCO which you have and any other current or thermal noise, q_{TDC} has uniform distribution. So, it is like if you look at q_{TDC} at one instant of time, it may have, it may lie anywhere between the probability of it lying between $-\frac{1}{2}$ and $\frac{1}{2}$ always remains the same. So, the probability distribution function of your q_{TDC} is uniform between $-\frac{1}{2}$ and $\frac{1}{2}$.

And if it is like this, then you can very well say this is 1 here. It is like if this happens to be, remember this, if this happens to be $-\frac{\Delta}{2}$ and $\frac{\Delta}{2}$, then what you have is $\frac{1}{\Delta}$. That is the density of the quantization noise. The same assumption will be made for the DAC quantization noise that the actual number is something and your quantization noise will always lie somewhere in between your $-\frac{1}{2}$ LSB to $\frac{1}{2}$ LSB.

So, based on this assumption that the distribution function of the quantization noise is like this, so, the total power, if you take this quantization noise over a long period of time, you calculate the power in that quantization noise, what you will find is that the mean square power or the total power which you will find will be equal to $\frac{\Delta^2}{12}$.

And the power spectral density of the quantization noise is given by,

$$S_{q_{TDC}}(f) = \frac{\Delta^2}{12} \times \frac{1}{f_{ref}}$$

It is this whole power is distributed, is sampled at f_{ref}. This power is distributed over the frequency f_{ref} when we are taking power from $\frac{-f_{ref}}{2}$ to $\frac{f_{ref}}{2}$. So, this is what you will see in terms of the quantization noise for the TDC.

Similarly, you will have for the DAC and quantization noise for the DAC is going to be same way. It is given by,

$$S_{DAC} = \frac{(DAC_{LSB})^2}{12} \times \frac{1}{f_{ref}}$$

That is what I am writing. So, see what I am doing is I am writing here is Δ. Now, we can define in different ways. Here, Δ= 1. Because it is going between the error is limited between $-\frac{1}{2}$ LSB to $\frac{1}{2}$ LSB.

But if at any point of time, if you want to convert that in the absolute number, it is like the error is between -10ps to +10ps. Then Δ will just change to 20 ps because -10ps to +10ps. So, this is $\frac{(DAC_{LSB})^2}{12}$ and this is again going to be f_{ref}. This power spectral density is assumed between $\frac{-f_{ref}}{2}$ to $\frac{f_{ref}}{2}$. Under the assumption, the noise is white, like this. So, both the noise sources are white, and this approximation is all quite valid.

Now, if you have such noise sources in the system, then you look at how these noise sources will get filtered at the output. The way we did for the noise transfer functions in case of the analog PLL. The noise transfer functions which we see, this is going to be NTF_{TDC}. NTF_{TDC} is equal to your, you see, look at the gain.

The gain is actually, the forward path gain is this and then you have the feedback gain or you can say that you have whatever loop gain you have, so, we get,

$$NTF_{TDC} = \frac{N.LG/K_{TDC}}{1 + LG}$$

Similarly, NTF_{DAC} is given by,

$$NTF_{DAC} = \frac{K_{DCO}}{s} \times \frac{1}{1 + LG}$$

and NTF_{VCO} is something we all know from the previous case. This is given by,

$$NTF_{VCO} = \frac{1}{1 + LG}$$

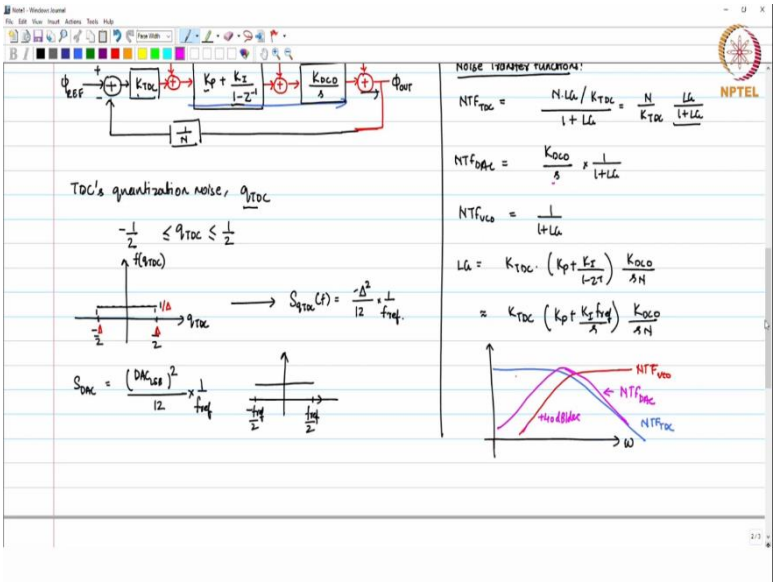
And here loop gain of this PLL is given by,

$$LG = K_{TDC}.\left(K_P + \frac{K_I}{1 - z^{-1}}\right)\frac{K_{DCO}}{sN}$$

We have approximated this earlier as,

$$LG = K_{TDC}.\left(K_P + \frac{K_I f_{ref}}{s}\right)\frac{K_{DCO}}{sN}$$

(Refer Slide Time: 24:17)



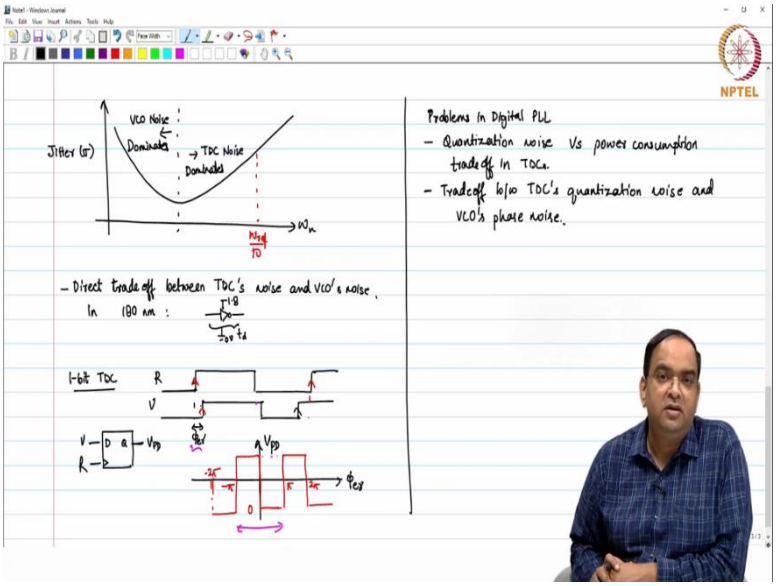
So, given this loop gain and the noise transfer functions, one thing we know, if we just plot rather than going and just thinking about these functions, it is easier to just plot them once. So, what we look at it is the following. You have this ω here and the noise transfer functions.

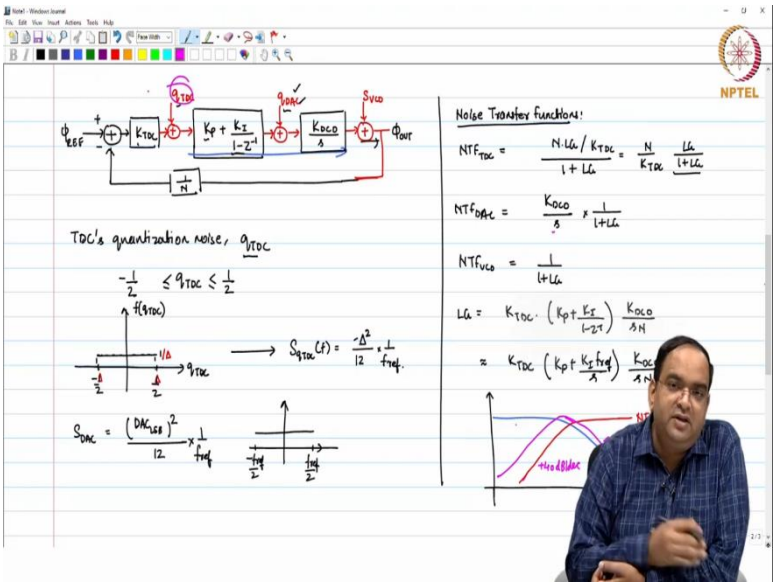
This is $\frac{LG}{1+LG}$ which is going to be low pass. $\frac{N}{K_{TDC}}$ factor is there, that is fine. This is low pass. This I can write this as NTF_{TDC}. NTF_{VCO} has been high pass and that is $\frac{1}{1+LG}$ which is going to be something like this. Do not look at the absolute value of these curves.

This is just the nature, high pass or low pass, is something which we are interested. This is NTF_{VCO} and this is a second order transfer function as you see at the end. And the other one which you see is NTF_{DAC}. That appears to be a bandpass transfer function. It is combining your low pass and high pass. So, what you will see for NTF_{DAC} at $s = 0$? So, $\frac{1}{1+LG}$ is this. This is changing at the rate your frequency is like s^2. This is like +40 dB/dec, $\frac{1}{s}$ is your -20 dB/dec.

So, overall you will have the suppression. So, you will see some kind of transfer function like this. Again, we are looking at the nature of the waveform, not the exact waveform. This will be NTF_{DAC}. In this case, most of the time the noise is going to be dominated by either your noise from the TDC or noise from the VCO. DAC noise is still lesser. Not like you can neglect it but most of the time the TDC noise or the VCO noise will dominate.

(Refer Slide Time: 26:53)





So, what happens is when we look at the, given this, given an example that you have the VCO phase noise, you have a certain TDC, what you will see is that if I go ahead and plot jitter versus the bandwidth of the PLL, σ which is the integrated jitter versus the bandwidth which I choose, ω_u for the PLL. You are going to find that jitter contribution will be something like this. And you will see beyond some optimum frequency in this region, TDC noise dominates, and in this region, your VCO noise dominates.

So, if you look at it, we got rid of the loop filter. At the same time, we also got rid of charge-pump noise and resistor noise. But we added two noise sources in the PLL and these two noise sources are the TDC noise and quantization noise. We cannot eliminate them for now. And this TDC noise is a real kill here which is like because of the TDC noise, most of the time you

would like to reduce the TDC noise, most of the time you would like to keep your bandwidth may be much smaller than $\frac{\omega_{ref}}{10}$. And the reason is you would like to suppress the TDC noise.

So, you keep the bandwidth smaller. You want to suppress the TDC noise. And it can also happen that because you reduce the bandwidth of the PLL to reduce the TDC noise, you have to burn more power in the oscillator to actually reduce the noise contribution of the VCO because the TDC requires lower bandwidth and lower bandwidth increases the phase noise of the VCO.

So, there is a direct trade-off. So, we see a direct trade-off between the noise of the TDC and the noise of the VCO. And this can only be, if you have no other option, what you can do is you can only reduce it by increasing, the overall noise you can only reduce by having a lower bandwidth for the TDC and increased power consumption for the VCO.

Now, one can say, if this is a problem, then what we can do is we can reduce the TDC noise itself. But reducing the noise of the TDC is not possible in any given technology. But for a given technology because the minimum delay which you can get which defines your resolution of the TDC, that minimum delay is limited by the technology itself.

So, in 180 nm technology, the minimum delay which you can realize, minimum CMOS delay which you can realize is like you design a simple inverter with the maximum supply voltage which is 1.8V and 0V. And look at the delay from input to output, what is the t_d? So, whatever is this t_d, this is the minimum delay which you can have in this process. And this is going to limit the quantization noise of the TDC.

Similarly, as you go down to a lower technology, you will find this delay reduces. But still this delay is much higher than the final output jitter which you would like to have. So, the delays will be like in ps, several 10s of ps in 180 nm and then it may reduce to 10 to 20 ps in 65nm, may be much lesser in the order of few ps in the lower technology nodes. But the jitter numbers which you want at the end, they will be in the order of sub-ps or few 100 fs. So, this is a problem which we would like to look at.

The other thing is that quite often because the TDC becomes power hungry, we actually reduce the number of bits in the TDC so that the power consumed in the TDC is lesser and we may be able to extend the range. But that sees a direct trade-off with q_{TDC}. So, this is a trade-off which

we have. The worst case among the TDC will be or worst case is like you use a 1-bit TDC. 1-bit TDC means your quantization error is very high. So, what is a 1-bit TDC?

So, let me just first explain it to you. 1-bit TDC is like I am having a clock like this. This is my R and I have my V signal something like this. I just need to tell among the R and V signal, whether my V signal is after R or before R, that is like 1-bit TDC. So, that can easily be done by using a simple D flip-flop.

You can have your V signal or let us say, you use a simple D flip-flop with a clock and the input and this is Q. So, this is your reference and this is your feedback. So, if your reference like in this particular case, whenever you get the rising edge on the reference, you look at what value of V you have. Here, the value of V in this example is 0. So, what you will sample is 0.

In the other case, if I just try to create this waveform, then when you are going to sample R here, in the next case this is going to be 1. So, in this way, if your phase error, if I just plot, if this is my PD output, I call this as V_{PD}, then depending on the phase error which you have, I look at V_{PD} depending on this R and V here, what we have is, if my phase error, if you call this as the phase error, if phase error is positive in this case, so, here if you look at it, if V is coming after R, then the value which R samples is lesser.

So, that is like $V_{PD} = 0$. I am just making this as 0 by the way so that it just looks little bit different. And when in the other case, if your R is coming after V where the phase error is negative, at that time you are going to see this kind of waveform. And this is going to, by the way this is going to flip at phase error of π and 2π.

So, the range of this 1-bit TDC, this is $-\pi$ to π. This is 2π here. This is -2π. So, the range of the phase error, here what you are getting is only the sign of the phase error. In this case, what we are seeing here is just the opposite value. That is okay. That is not a problem. The only thing is for phase error 0 to π, you sample it as 0.

And for phase error $-\pi$ to 0, you sample it as 1. So, the range of 1-bit TDC is limited. This is the worst TDC which one can use. But this is the worst TDC in terms of quantization noise. But this is actually the best TDC in terms of power consumption because you use only a single flip flop. There are no delay cells. There is nothing. You only check whether your phase error is greater or lesser.

So, now, what you find here is that TDC is one of the main culprits in degrading the performance of the digital PLLs. Either they have a very large power consumption and good resolution or they have a very poor resolution and good power consumption. So, overall it becomes a problem. So, if we just summarize the problems with the digital PLL, quantization noise versus power consumption trade-off in TDCs. This is one big problem.

So, either you lose on power or you lose on noise. The other one is the trade-off between the quantization noise of the TDC and the phase noise of the VCO. These are the two problems which we have. Now, in the presence of these problems, what we are going to do, we need to look at that next. Thank you.