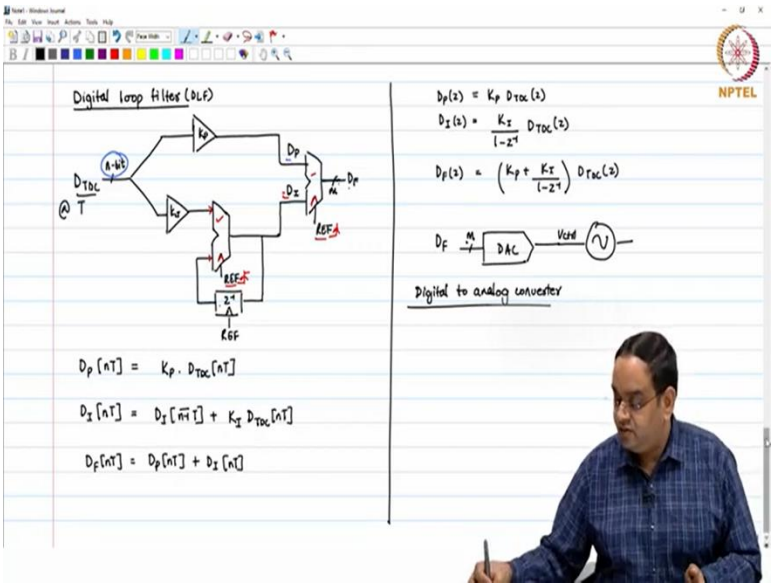
Phase-Locked Loops
Dr. Saurabh Saxena
Department of Electrical Engineering
Indian Institute of Technology Madras

Lecture – 61
Small Signal Analysis of Digital PLL

Hello, welcome to this session. So, in the previous session, we looked at the simple design of time-to-digital converter and understood how we can convert the phase error into its digital equivalent.

(Refer Slide Time: 00:35)



Now, we will look at what is this digital loop filter, how we are going to implement it and what is the transfer function for that. So, as we saw earlier, the digital loop filter is fed by your D_{TDC} coming from the TDC. So, this can be n bit depending on how you design it. Then you have the gain K_P, gain K_I. So, you have an accumulator. This accumulator is in general clocked at your reference. You are using a digital accumulator, so it is going to change at every reference clock period and then you have this as feedback.

When I implement this as z^{-1}, it is actually having a delay of one clock period. So, you get this and then these two things are again added up with an accumulator like this. So, this is another reference clock. Wherever I use z^{-1}, it is actually clocked at reference. This is D_P, I can call this digital word coming from the proportional path, this is the digital word coming from the integral part and this is the digital control word coming out of the loop filter D_F.

So, here if you try to write the small signal block diagram or you want to first write that how multiplication is happening. So, D_{TDC} output you get at reference clock period. At every T_{REF}, you may see a change in the D_{TDC}. So, we have,

$$D_P[nT] = K_P . D_{TDC}[nT]$$

Similarly, we get,

$$D_I[nT] = D_I[(n-1)T] + K_I D_{TDC}[nT]$$

Then, we have,

$$D_F[nT] = D_P[nT] + D_I[nT]$$

I will remove T from all this, writing nT every time is a problem. It is not a problem for technical reasons. It is just that there is no point in writing it again and again when we know that behind what is happening is at every reference clock period.

Now, there are a few things if you think about it that this is an accumulator. It is a digital block. These digital blocks work at the rising edge. Whenever you have a rising edge on the clock, what will happen is whatever the previous value you have, so, when you get a clock, whatever the value you see here, this value is going to be added with the K_I times whatever the present value is and the output will change.

Normally, it takes some time to compute and get a certain output. So, there is some delay involved from reference clock to the output value. So, sometimes it may be done that you do not wait like here you will say when you have a reference clock period here and you get a reference clock here. So, you are seeing the rising edge on both of them, what we can say is we can combine even these two functions and say that you get the accumulated value at the same rising edge.

So, the idea is that when you get a rising edge on reference, from this rising edge to D_I, there is a certain delay involved. And because D_I changes after the reference edge, so, whatever you are getting at this reference edge, whether you are going to process that previous D_I or you are going to process the current D_I, it depends on the implementation.

In this case, the way I have written is that the current value or at the rising edge of n times T clock, whatever the previous value of the integrator value you have. So, if this is this, whatever value you

have at the rising edge, you will get the previous value here and this previous value is added with respect to the current value. So, given these transfer functions, what we can say here is that,

$$D_P(z) = K_P D_{TDC}(z)$$

Now we have changed into z-domain transfer function. Similarly, we have,

$$D_I(z) = \frac{K_I}{1 - z^{-1}} D_{TDC}(z)$$

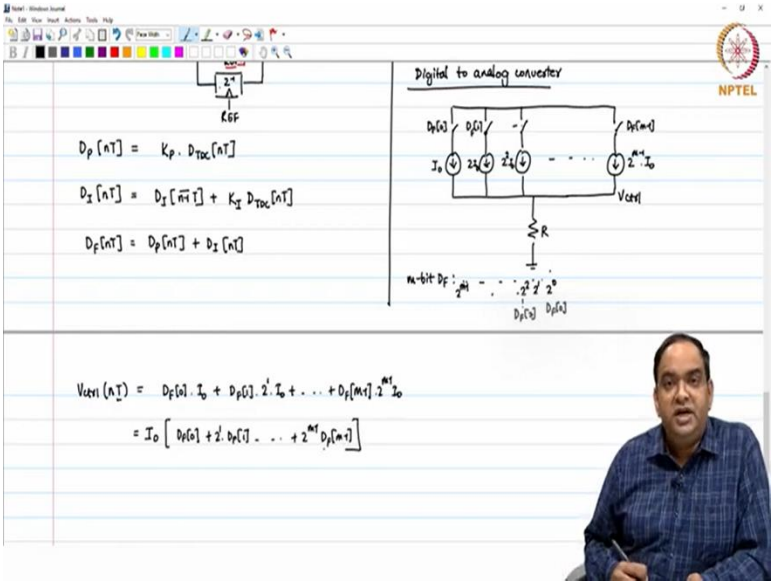
$$D_F(z) = \left(K_P + \frac{K_I}{1 - z^{-1}}\right) D_{TDC}(z)$$

So, this is the transfer function in z-domain. If you implement this loop filter in some other manner where you have delays involved at any given point, you can take those delays into account.

Now, as you see that this particular value is a n bit number and you are adding n bit number with D_P and it keeps on getting accumulated. So, the final value which you may, the D_F which you are getting, this particular D_F, it can be m bit number, some other value, it is m bit number. So, you see that the digital bits which you are getting, these digital bits accumulate and it gives you m bit. It is the same way you got the phase error, phase error got converted and then you integrate using the charge-pump. The same way you got the phase error representation in D_{TDC}, D_{TDC} gets accumulated here and you get the equivalent value and this is going to be m bit here.

Now, this m bit number, whatever you get, this m bit number has to control your oscillator, voltage controlled oscillator. So, what we have now is using the loop filter output, we need to control, so, D_F has to control your oscillator and oscillator works with a control voltage. So, this D_F now has to convert to control voltage, digital-to-analog converter. So, whatever digital word you have, you will convert that digital word to the voltage. So, here if I am getting m bit from here, these m bits have to convert to the respective control voltage. So, there will be gain involved that what does LSB mean here.

(Refer Slide Time: 10:20)



So, let me just try to show you one particular example of this m bit digital-to-analog converter. So, let us say these m bits are converted to control voltage using current sources. So, I am using current source like this and all these current sources are because this is just a simple implementation, you can do it in many different ways and all these current sources, they go to a resistor R. This current source I am because this is a m-bit number.

So, I will use let us say the current I_0, $2I_0$, $2^2 I_0$ and similarly I can have, if this is m-bit number I can ideally have $2^{m-1} I_0$. So, you see the 2^0 going to 2^{m-1} for m-bit DAC. And all these switches, this particular thing you can say is controlled by $D_F[0]$. $D_F[0]$ here means the 0^{th} bit of the digital control word. This is not like at t^{th} instant of time, these are just the bits. So, $D_F[1]$ and so on and this is controlled by $D_F[m-1]$.

So, what I am doing here is for m-bit D_F, I am just writing as, so if m-bit D_F, you know each bit which you have, that will have a place value of 2^0, 2^1, 2^2 and so on to 2^{m-1}. So, when I say $D_F[0]$, $D_F[0]$ means the bit at 2^0, that place value. Similarly, 2^2 means $D_F[2]$. That is how I am controlling. You can have any nomenclature you prefer.

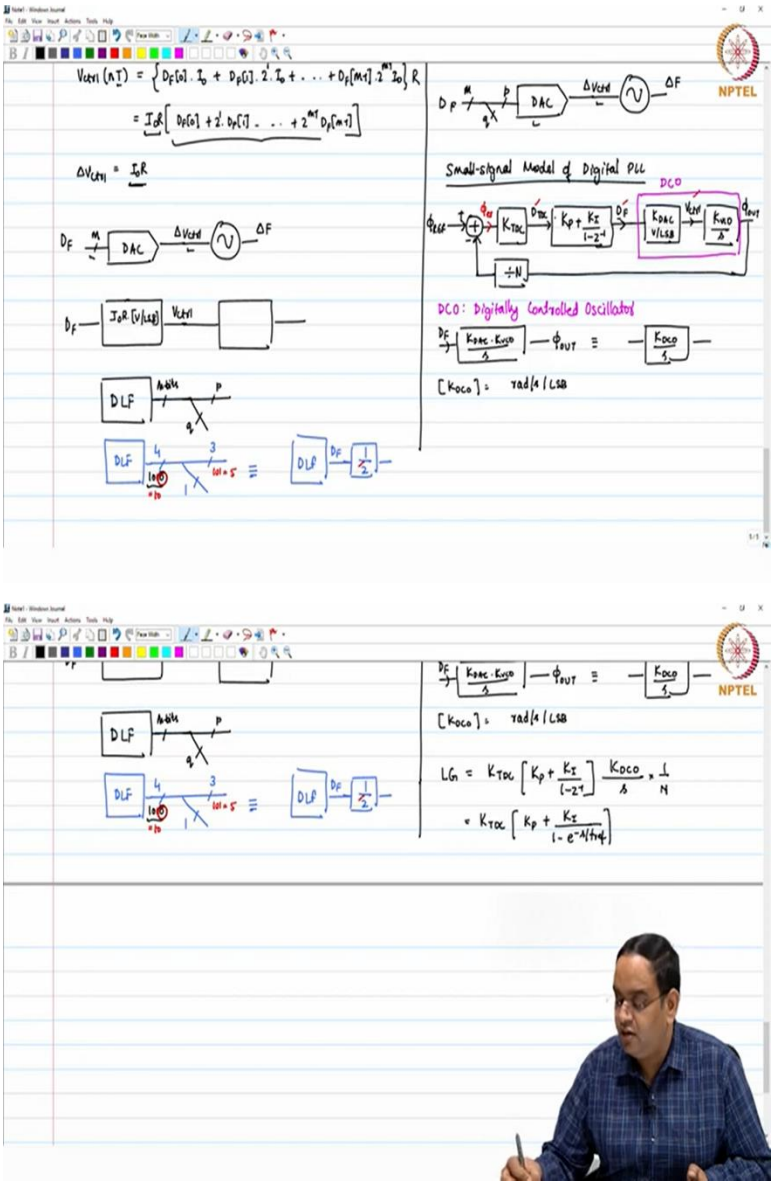
So, here V_{ctrl} which you are looking at, this particular V_{ctrl} value, one thing is that this V_{ctrl} value is going to change only when your digital control word is going to change from the loop filter and that happens at every reference clock period. So, I can write,

$$V_{ctrl}(nT) = \{D_F[0].I_0 + D_F[1].2^1.I_0 + \cdots + D_F[m-1].2^{m-1}.I_0\}R$$

$$V_{ctrl}(nT) = I_0R\left[D_F[0] + 2^1.D_F[1] + \cdots + 2^{m-1}.D_F[m-1]\right]$$

So, this is your control voltage at the given time instant.

(Refer Slide Time: 14:22)



So, this can be written as, by the way if you look at it, this is nothing but the decimal equivalent of your digital loop filter output. So, whatever your digital loop filter output you have, so, your voltage which you are seeing here, your voltage is changing with $\Delta V_{ctrl} = I_0R$. So, the control voltage is actually changing in steps of I_0R, whatever you are using.

So, now the thing is that your digital control word, it is whatever value you get, it can change the control when you are converting from digital to analog, depending on what resolution you choose or what I_0 and R values you are choosing. These I_0 and R values will decide the step of the change in the control voltage of the oscillator. So, it changes only in this step.

So, with respect to your digital loop filter and your V_{ctrl}, I can very well keep a gain here and say I can model this particular part with a gain block, DAC with a gain block, whatever digital word you are getting. But when we are doing the analysis, you can say that it is finally converted into its decimal equivalent, from binary to decimal equivalent and that decimal equivalent value is then multiplied by I_0R to give you the voltage. So, this is normally written as, by the way, it is written as I_0R V/LSB, this gives you the control voltage here, this is your D_F.

If you make 2 LSB change here, your output voltage will change by $2I_0R$. So, the gain of the DAC is I_0R V/LSB. When I write the LSB at the bottom, it appeared like D_F may go into the denominator but these are actually the units. So, this is D_F. So, this is how you are going to change your control voltage and once you change your control voltage, from the control voltage to the oscillator, you very well know what the transfer function you are going to get.

One more thing here that if you want to change your frequency, well, if you want to change your frequency by a small amount, if you want to change the gain, you need to choose a smaller I_0 or a smaller R, but often that is not the possibility in your design because you need to cover a certain range of the VCO frequency also.

So, whatever ΔF output you want with respect to ΔV_{ctrl}, based on ΔV_{ctrl} and the maximum digital word which you have a limit, you are going to decide your I_0R. It is not possible to choose any I_0 or any R. It is constrained by your accumulator output, your frequency range which you want, what kind of step in the frequency tuning you want.

One more thing here is that we have seen all this digital accumulator and so on. One easy thing about the digital part is that whenever we want to increase or decrease the gain of the blocks, increasing or decreasing the gain of the blocks can be easily done by dropping the bits or by adding a bit or by shifting the bits. So, just as an example, I will show you that if I am getting my digital loop filter, whatever digital loop filter I have, I am getting m bits here.

Now, in these m bits, what I do is I drop q LSBs. So, it is normally shown by a sign like this, drop q LSBs and just take p MSBs. So, for example, what I am going to do is just I will show you one example that I have let us say 4 bits to begin with, I drop 1 bit and I take 3 bits. This, in your analysis term, is equivalent to saying that the DLF output is actually multiplied by gain $\frac{1}{2}$.

Because you drop the last bit, so when you drop the last bit, you actually divide by $\frac{1}{2}$. So, it is like just think about it, if the number which I have here is 10. So, this is the digital bit, I dropped LSB which means this particular bit I dropped. So, this bit is dropped. When this bit is dropped, what you get here is 101. What is 101? Decimal equivalent 5, what was the decimal equivalent here? 10. If the number would have been 11, 11 will also become 5 after division by 2 here and 10 will also become 5 after division by 2 but for all practical purposes, we can say that we have divided the number by $\frac{1}{2}$. The error is $\pm\frac{1}{2}$ LSB in any case.

So, this kind of division or this kind of dropping of the bits to implement the division is quite common and it is more common in the case of your V_{ctrl}. So, the reason is when you actually do the accumulator, you want to have good phase error, you want to have a larger integrated value range, integrated output. Quite often it is not possible to implement the DAC which you are having, see having 2^{m-1} bits, if m is 10, having a 10 bit DAC is not that easy, let me put it that way.

So, these m bits which you are getting from D_F, you often do a division. You take only p bits. If you do this, you will implement a gain of $\frac{\frac{1}{2}}{q}$. So, this often happens. Now, given all these things, let us look at the small signal model of the digital PLL now. We have seen the gain of each block of the digital PLL, so, beginning with TDC, you have phase reference, you have feedback, the output goes to TDC with a gain K_{TDC}. I am not adding the quantization error here right now. This goes to loop filter whose transfer function is $K_P + \frac{K_I}{1-z^{-1}}$, the output of that, this is D_{TDC}.

The output of that is D_F and this D_F goes to DAC whose gain happens to be you can say K_{DAC} V/LSB. K_{DAC} is like the voltage gain per LSB, this is the gain. Then this becomes your control voltage, it goes to VCO whose gain is $\frac{K_{VCO}}{s}$ and the output is fed back, this is what you have. This is your φ_{OUT} at the final value. So, one thing you can first make sure are the units. So, you get φ_{ER}

here, K_{TDC} has units of LSB per phase error, it is just the digital word in both input and output for the loop filter, you get the control voltage, you get K_{VCO}.

So, now these things, this block together is often written not as a VCO but as a DCO. DCO stands for Digitally Controlled Oscillator. So, what you can say is that the combined K_{DCO} or the digital block gain is from D_F to output. So, the gain from D_F to φ_{OUT}, if I write this gain from D_F to φ_{OUT}, this is D_F, this is φ_{OUT}, it is $\frac{K_{DAC}.K_{VCO}}{s}$. So, this is written as in general as $\frac{K_{DCO}}{s}$ where K_{DCO} has units of rad/s/LSB.

So, that is the gain K_{DCO}. Whether you use as a K_{DCO} or you use as K_{VCO}, it is one and the same thing till the time your total loop gain remains same. Now, this is the digital loop. So, we have to analyze this loop gain, what is the unity gain frequency and so on. So, the loop gain of the digital loop is, I need to just find the gain as I go around the loop, it is given by,

$$LG = K_{TDC}\left[K_P + \frac{K_I}{1 - z^{-1}}\right]\frac{K_{DCO}}{s} \times \frac{1}{N}$$

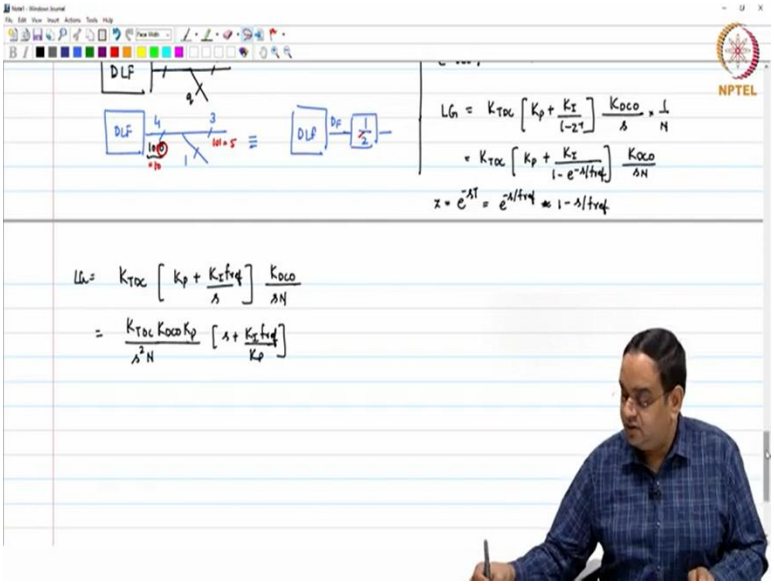
Now, this is interesting here that a part of the transfer function is in z-domain and a part of the transfer function we are using is still in s-domain.

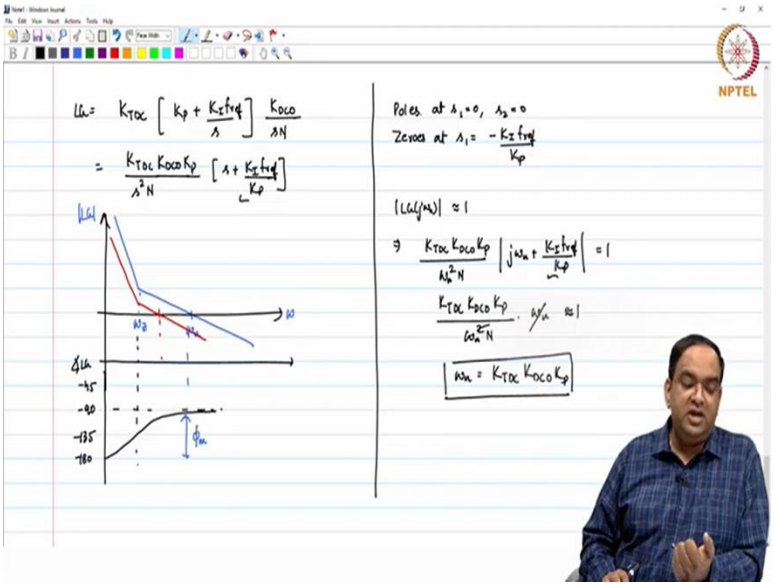
So, it will be better either we convert the whole of it in z-domain or the whole of this transfer function in s-domain. So, we will convert this complete transfer function in s-domain and we are going to do that using this analysis that $z = e^{sT}$, where T is your reference period. So, what you can do here is if I go ahead and write it that way. So, we get,

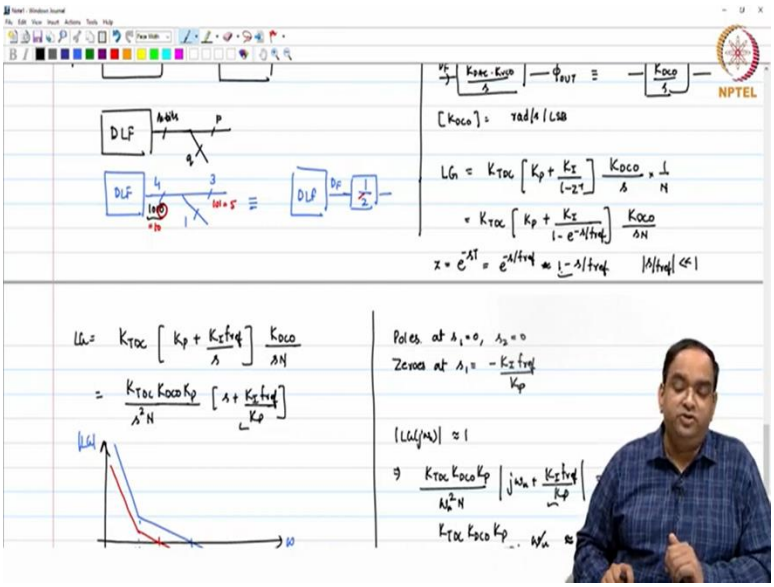
$$LG = K_{TDC}\left[K_P + \frac{K_I}{1 - e^{\frac{-s}{f_{ref}}}}\right]\frac{K_{DCO}}{sN}$$

So, either you write like this or you write $\frac{s}{f_{ref}}$, both the things are possible.

(Refer Slide Time: 28:26)







So, what I have is,

$$z = e^{sT} = e^{\frac{s}{f_{ref}}} \approx 1 + \frac{s}{f_{ref}}$$

So, you do this, what you get here is,

$$z^{-1} = e^{-sT} = e^{\frac{-s}{f_{ref}}} \approx 1 - \frac{s}{f_{ref}}$$

$$LG = K_{TDC}\left[K_P + \frac{K_I f_{ref}}{s}\right]\frac{K_{DCO}}{sN}$$

$$LG = \frac{K_{TDC}K_{DCO}K_P}{s^2 N}\left[s + \frac{K_I f_{ref}}{K_P}\right]$$

It is like you have two poles and one zero which we know is a stable system.

So, here I will just plot this loop gain and the angle. So, you have s^2, so it comes by -40 dB/dec. At this zero location, which is actually ω_z, zero is at $-\frac{K_I f_{ref}}{K_P}$ and $\omega_z = \frac{K_I f_{ref}}{K_P}$. This is your ω_u, magnitude of loop gain with respect to ω. And the angle of the loop gain is, it starts from -180°, it goes to -45°, so it is -45°, -90°, -135°, -180°. So, it starts from -180°, it goes to -135° at ω_z and after some time, it may reach -90°. It never goes above -90° and it does not come down also.

So, this is the phase margin of the PLL. So, here just to summarize it, we can say this digital PLL has two poles, poles at $s_1 = 0$ and $s_2 = 0$. Zeroes at $s_1 = -\frac{K_I f_{ref}}{K_P}$, this is what we have. Now, given this, zeroes and poles and the phase margin close to 90° quite often and if not 90°, you can always, your system is always going to be stable anyways because it is two poles and one zero. So, you may have, well, you can say I have, depending on the loop gain, I can have something like this where you will have a lower phase margin but stability is there.

Now, the unity gain frequency, so, to find the unity gain frequency, you need to just say that,

$$|LG(j\omega_u)| \approx 1$$

So, we get,

$$\frac{K_{TDC}K_{DCO}K_P}{\omega_u^2 N}\left|j\omega_u + \frac{K_I f_{ref}}{K_P}\right| \approx 1$$

$$\frac{K_{TDC}K_{DCO}K_P}{\omega_u^2 N}.\omega_u \approx 1$$

This is because $\omega_z \ll \omega_u$, that is an assumption here.

So, we get,

$$\omega_u \approx \frac{K_{TDC}K_{DCO}K_P}{N}$$

So, the unity gain frequency is equal to the gain in the proportional path, $K_{TDC}K_{DCO}K_P$, they are all the gain of the blocks in the proportional path, so that is what we get. This analysis is surely valid under the assumption that for the frequencies which I am considering, $\frac{s}{f_{ref}} \ll 1$.

And this much less than is quite often actually this value has $\frac{s}{f_{ref}} \ll \frac{1}{10}$. So, this is under this assumption. If this assumption is not valid, I cannot approximate this. So, this is an approximation. If the frequencies at which you are considering or the frequencies become quite close to your reference frequency, then you cannot approximate, you have to go with a more accurate analysis. Thank you.