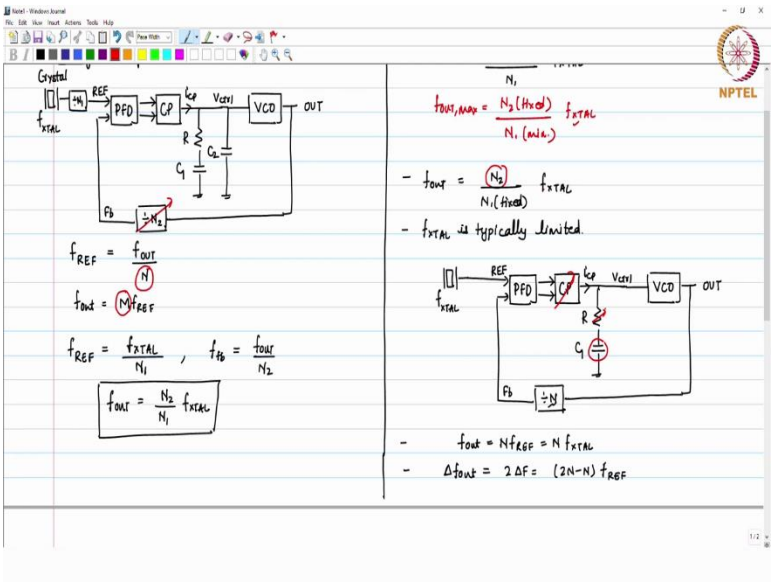
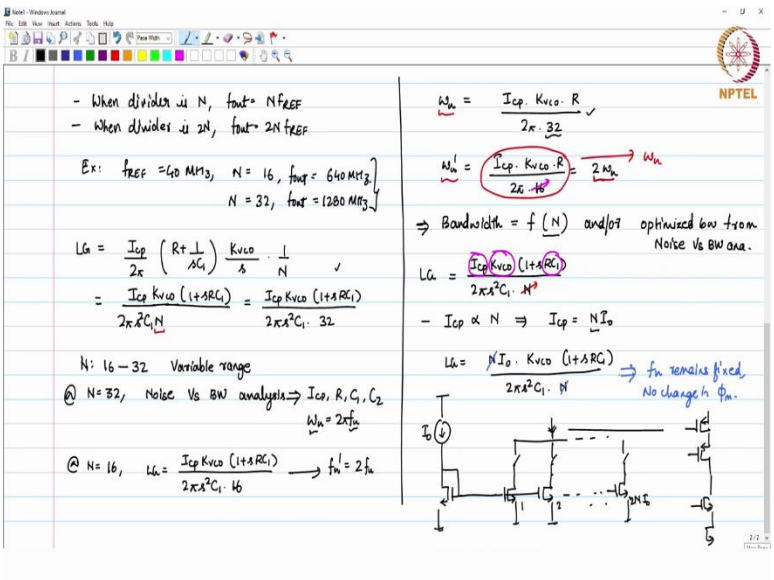
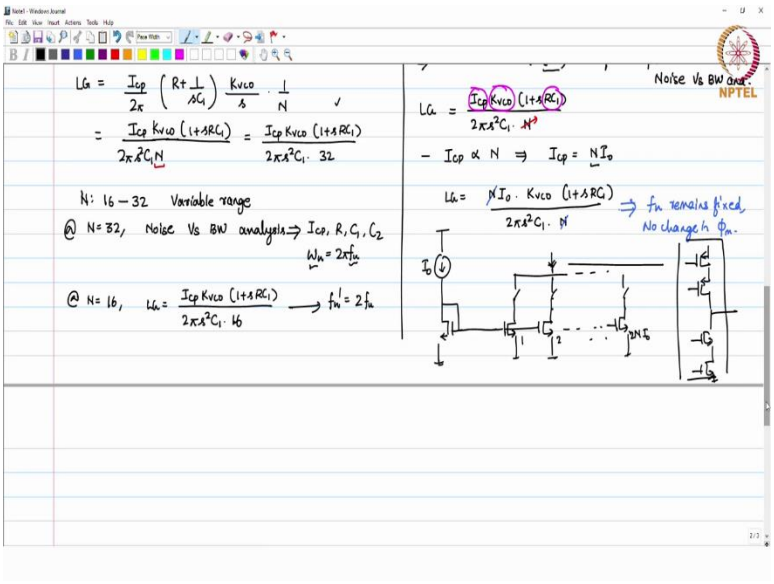
Phase-Locked Loops
Dr. Saurabh Saxena
Department of Electrical Engineering
Indian Institute of Technology Madras

Lecture – 58
Techniques for Wide Frequency Range Clock Multiplier

(Refer Slide Time: 00:15)







So, so far, we have seen the design of a charge-pump PLL and we were focused on designing this PLL for one single output frequency. Now, as we go ahead and we employ this PLL in a given application, it is often required that you may need to change the output frequency for the same PLL. So, let us just look at the PLL block diagram which we have been studying in-depth. So, a charge-pump PLL, this is what you are looking at.

We have a PFD followed by charge-pump where you get UP and DN pulses. And then output of the charge-pump goes to a loop filter like this and then you have your capacitors C_1 and C_2 and here you have a VCO and then you have a divider in between. So, I will write this as N here and this is your PLL. So, you have a reference clock and from the VCO, you have output clock.

Now, in most of the applications, what you will see is that your reference frequency in general remains the same because you are going to use a crystal as a reference. So, this is the symbol of the crystal I am using, this is crystal and this feeds into your PLL. The crystal can be a piezo-electric crystal or it can be temperature compensated crystal oscillator (TCXO) or voltage compensated. Here you have this i_{cp} and V_{ctrl}. So, in our case, what we have is this crystal which gives us the reference signal here, the frequency of the reference is f_{REF} and we know that it is given by,

$$f_{REF} = \frac{f_{OUT}}{N}$$

What we want is that,

$$f_{OUT} = Mf_{REF}$$

where, M is variable. How do we make this M variable? That is something which is for us to decide. So, in the PLL loop, what you see here is that the feedback loop will always enforce that the feedback signal at the input of the PFD and your reference signal, both these signals should have the same frequency. So, in order to change the output frequency, we can have two possible methods. One, because the feedback loop is going to enforce that, so I can have a divider here let us say $\div N_1$ or a divider in the feedback which is $\div N_2$.

So, now the reference which goes to your PFD is this and this is your feedback signal. So, f_{REF} in this new case is equal to your crystal frequency. So, you can say,

$$f_{REF} = \frac{f_{XTAL}}{N_1}$$

and your feedback frequency is given by,

$$f_{fb} = \frac{f_{OUT}}{N_2}$$

The PLL loop is going to enforce that your reference frequency is equal to your feedback frequency. So, we get,

$$f_{OUT} = \frac{N_2}{N_1} f_{XTAL}$$

So, if you want to change the output frequency, you can change the output frequency in this particular manner. You look one by one. If I have f_{OUT}, I will keep one of it as fixed, so N_2 is fixed for example and N_1 is varying. In that case, the maximum frequency which I can have in my system is given by,

$$f_{OUT,max} = \frac{N_2(fixed)}{N_1(min)} f_{XTAL}$$

The minimum value of divider is 1, so, you do not divide. Whatever your crystal frequency you have, your $N_2(fixed)$ times that crystal frequency is something which you will get. Similarly, if I want to keep my N_1 as fixed and make my N_2 variable. Then, we have,

$$f_{OUT} = \frac{N_2}{N_1(fixed)} f_{XTAL}$$

So, in this case, whatever you have your N_2, depending on your N_2, you will get the output frequency.

The problem with having a divider in the crystal path is that f_{XTAL} is typically limited. For a good phase noise at the crystal output, the crystal frequency is typically limited. And if you want a really good high frequency crystal, the size will be large, the power consumption will be more and your cost will also be high. So, crystal frequency is typically limited.

So, dividing the crystal frequency in general is not an option. It is not that technically it is not possible. It is very much possible but quite often what we use is we vary only the divider part here. We vary the division factor in the feedback path to vary the output frequency. So, if we do that, let us say you have both the options but most of the time we use this one only in the feedback.

So, given this PLL where you are only varying the input frequency and this is your feedback. Now I will just use a divider N here rather than N_2 to make sure that you have only one divider in the PLL. So, what are the problems or the challenges which you have when you want to have a large tuning range? So, here for simplicity, I will just remove the C_2 for now. You can add C_2, just to avoid any complicated expressions for now.

So, in this example, let us say, we have,

$$f_{OUT} = Nf_{REF} = Nf_{XTAL}$$

And I want to have a wide range for this output frequency. So, I will just choose one factor that,

$$\Delta f_{OUT} = 2\Delta f = (2N - N)f_{REF}$$

You can say this division factor is varying from N to $2N$. That is how you want to have.

So, when your division factor was N, so let us just write it here. So, when your division factor is N, when divider is N, in that case, $f_{OUT} = Nf_{REF}$ and you are able to change the divider ratio from N to $2N$ in the PLL. So, $f_{OUT} = 2Nf_{REF}$. So, the range which we cover is 2x.

As an example, just to make it look better, so let us say your $f_{REF} = 40$ MHz and $N = 16$. In that case, $f_{OUT} = 640$ MHz. And I am able to change this value to 32 and the PLL still operates fine. This is equal to your 1280 MHz. So, this range if you look at it, this is like a 2x range which you have. That is what we call as a 2x range. So, for this 2x range, if we just change the divider N here, what are the problems which we will face? Is there any issue or there is no problem at all?

So, when we want to extend the output range of the PLL, in that case, let us write down the loop gain expression of the PLL. The loop gain is given by,

$$LG = \frac{I_{CP}}{2\pi}\left(R + \frac{1}{sC_1}\right)\frac{K_{VCO}}{s}\frac{1}{N}$$

$$LG = \frac{I_{CP}K_{VCO}(1 + sRC_1)}{2\pi s^2 C_1 . N}$$

So, here we have seen this multiple times. So, when we design our PLL, we design our PLL in general for the highest frequency which we have to minimize the jitter at the output.

So, that is what we are going to do and then we did all our noise transfer function analysis and bandwidth and so on. Everything was done perfectly and in our example, let us say we want to change N, here N which you are seeing, N can be varied from 16 to 32. It is not only like 16 and 32, but 16, 17, 18 or whatever N you have, this is the variable range which you want.

So, if I keep everything else constant and I vary this particular N, there are many assumptions which we can make to say that our design is a good design here or an optimized design, these assumptions may or may not be valid.

So, first one is at $N = 32$, so we have the loop gain as,

$$LG = \frac{I_{CP}K_{VCO}(1 + sRC_1)}{2\pi s^2 C_1 . 32}$$

Let us say, you did the noise versus bandwidth analysis. The noise versus bandwidth analysis gave you I_{CP}, R, C_1 for a given phase margin. If you had C_2, then you will also get C_2. For expression, I am not writing C_2 but assume it is there. So, you get all these component values and $\omega_u = 2\pi f_u$. So, your unity gain frequency is optimized for this case.

Now, suddenly I go ahead and change N to 16. What happens if I change from $N = 32$ to $N = 16$? My loop gain expression is given by,

$$LG = \frac{I_{CP}K_{VCO}(1 + sRC_1)}{2\pi s^2 C_1.16}$$

Now, for this loop gain expression, if you do not do anything, whatever unity gain frequency you had in the previous case, the unity gain frequency in the second case is you can say $f_u{}' = 2f_u$.

How did we get that? Well, from the loop gain expression, assuming your unity gain frequency is much larger than your zero frequency. So, you can find that out. It is given by,

$$\omega_u = \frac{I_{CP}K_{VCO}R}{2\pi.32}$$

This is what you had. Now, as soon as I change my N factor, this is the N factor which you get. So, we get,

$$\omega_u{}' = \frac{I_{CP}K_{VCO}R}{2\pi.16} = 2\omega_u$$

So, your unity gain frequency just changes and if your unity gain frequency changes, then your noise analysis is also going to change or the noise at the output, total jitter at the output changes. Now, previously ω_u might be the optimized bandwidth. Now $\omega_u{}'$ is not at all the optimized bandwidth. It is just you can say a by-product of whatever you did to change the divider.

So, this is something which may or may not be good for the design. This implies that in order to have an optimized design, bandwidth should be some function of the divider N. Now, you can say whether it should, so typically if it is the case that I want to cancel, the oscillator noise is dominant and the maximum bandwidth which we can have is $\frac{f_{REF}}{10}$, so if you want to keep the bandwidth

constant across the output frequency, then you can say I would like to keep the bandwidth directly proportional to N.

So, here bandwidth is, in this case what you are seeing is the bandwidth is some value, here the bandwidth becomes twice ideally. I have to do something to this expression such that the bandwidth for this still remains ω_u. It should not increase to $2\omega_u$. Yes, I know that this particular divider value changes, that is something which you need to fix.

So, if you look at the loop gain expression which we had before, so, I will just write the expression here which we had earlier. So, we have,

$$LG = \frac{I_{CP}K_{VCO}(1 + sRC_1)}{2\pi s^2 C_1.N}$$

I know that in this loop gain expression, my N changes as per the required frequency. If N changes as per the required frequency, I would like to change some other parameter.

So, let us say you optimize everything at the highest frequency where you need the minimum jitter and then you would like to keep the bandwidth fixed because bandwidth is decided by the reference frequency, the maximum bandwidth, you cannot increase it more than a certain value. So, you would like the bandwidth as a function of N and/or let me say the optimized bandwidth from noise versus bandwidth analysis.

So, you would like to do that. Either thing is something which you will utilize. Now, given the PLL when you have this whole block diagram of the PLL or this thing, there are a few things which you cannot change. You cannot change the gain of the PFD. For the charge-pump, you have the charge-pump current that is something which is in your control. For R and C, changing the capacitor C is typically hard because capacitor has a large value and large area and adding tunability to the capacitor is difficult to implement or you can say it has a lot of cost in terms of area.

Resistance is the other thing which you would, you can try to change. VCO is not that much in your control, but we will see how we can control the K_{VCO}. So, we look at each of these parameters and see whether we can change it or not. So, I_{CP}, K_{VCO} and this parameter. So, there are a few parameters which if you change, you may not find, few parameters may be easy to change without

changing phase margin and other values. It will just help in retaining the unity gain frequency. A few parameters are difficult to change, but there is nothing which stops you from doing that.

So, in this case, let us look at it, if I make $I_{CP} \propto N$, for example, if such is the case, I can write,

$$I_{CP} = NI_0$$

When I am getting this bias current, I make this I_{CP} proportional to the divider ratio, then in my loop gain expression, I am going to have,

$$LG = \frac{NI_0 K_{VCO}(1 + sRC_1)}{2\pi s^2 C_1 . N}$$

So, what you see is that your loop gain transfer function is actually independent of N, your unity gain bandwidth will be retained and all your noise transfer functions, they are going to be, so noise transfer function depends on loop gain, so only the thing is the charge-pump transfer function will vary but your loop gain will remain constant. Charge-pump noise transfer function has your $\frac{I_{CP}}{2\pi}$ in the expression other than the loop gain which will vary.

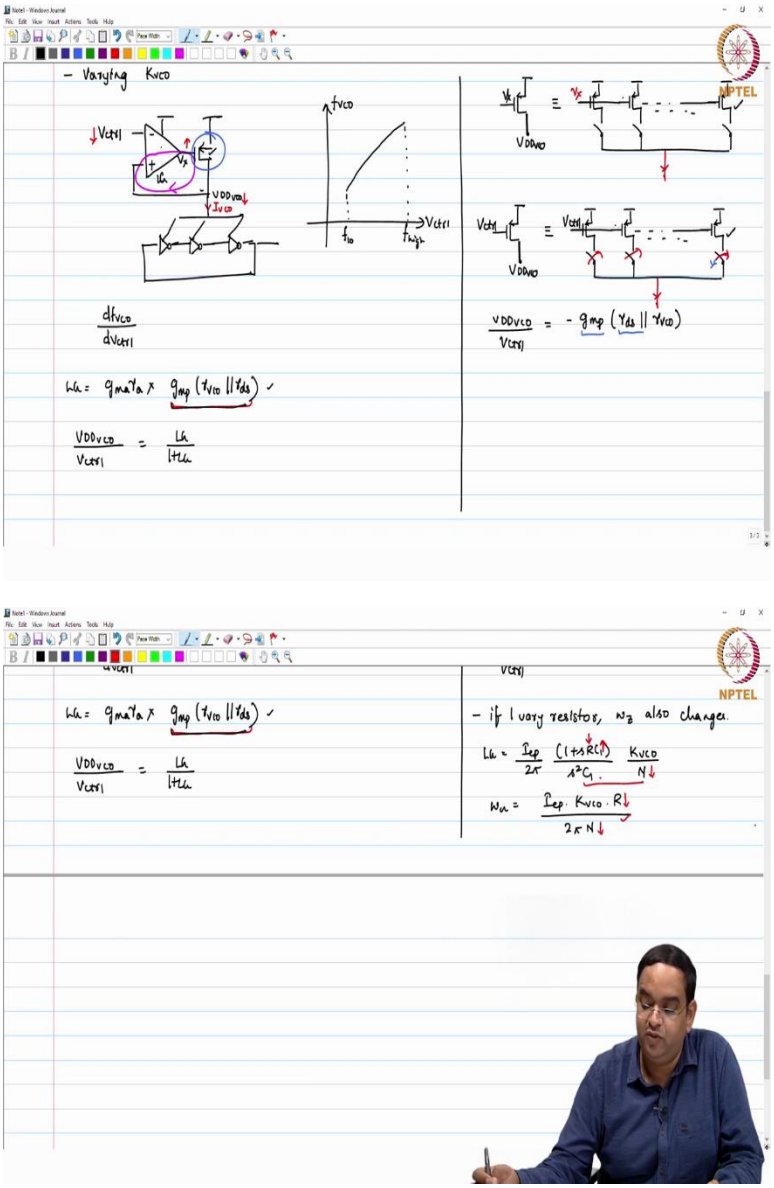
Other than that, your unity gain bandwidth f_u remains fixed, there is no change in the phase margin. Now, you can ask a question that how am I going to change I_{CP} with respect to N? Well, you know what divider ratio you are going to use. So, based on that, you can have a bias current which is switchable and from that bias current, you can change the charge-pump current. So, a very simple implementation you can think about it that you have an external current which is given to a diode, just an example, this is I_0.

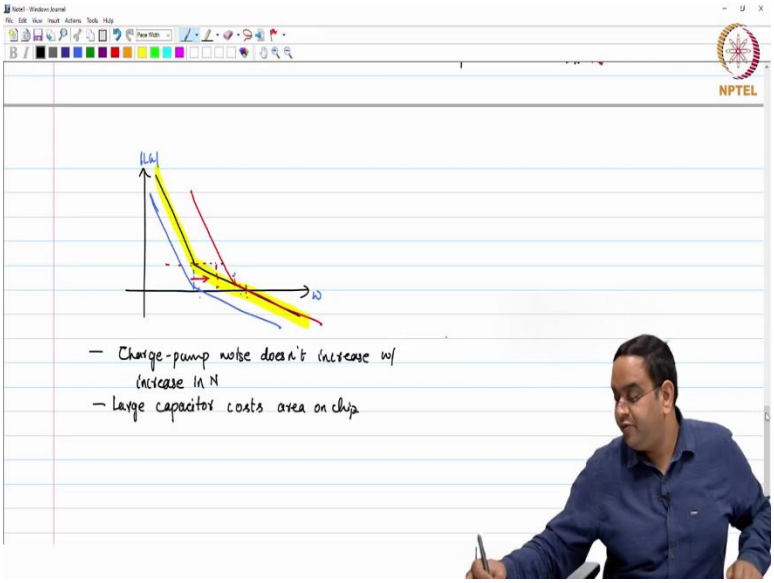
And if I want to take this current as NI_0, maybe I can do this. So, I can, this is just one example, do not think that this is the only way you can do it. So, you can have switches here and these switches are going to finally give your bias current to your charge-pump. So, if I want this particular thing to be tunable from I_0, I can keep this 1, 2, to $2NI_0$ if I want that.

So, based on how much current you are switching in, this particular current is then mirrored in the charge-pump somehow. The basic charge-pump which you would like to use, you can mirror this current in the charge-pump. I am not showing the exact circuit but that is what you can do. So, in this way, you can control your charge-pump current. Now, this is one way to deal with this.

Another thing is the K_{VCO} variation. So, now normally we do not vary K_{VCO} or we do not change the K_{VCO} that much but if you like, if the design needs, that is also possible. The reason here is that as you change the I_{CP} current, your unity gain bandwidth remains fixed, but because you change your I_{CP} current and you are reducing your I_{CP} current, so the noise from the charge-pump current will, overall noise from the charge-pump current will increase. And if the overall noise from the charge-pump current increases, it may lead to a large jitter as compared to what you think. So, that is something which may go against while changing your I_{CP} with respect to your N factor.

(Refer Slide Time: 24:51)





- Charge-pump noise doesn't increase w/ increase in N
- Large capacitor costs area on chip

- When divider is N, $f_{out} = N f_{REF}$
- When divider is 2N, $f_{out} = 2N f_{REF}$

Ex: $f_{REF} = 40\ MHz$, $N = 16$, $f_{out} = 640\ MHz$
$N = 32$, $f_{out} = 1280\ MHz$

$$LG = \frac{I_{cp}}{2\pi}\left(R + \frac{1}{sC_1}\right)\frac{K_{VCO}}{s}\cdot\frac{1}{N} \quad \checkmark$$

$$= \frac{I_{cp} K_{VCO}(1+sRC_1)}{2\pi s^2 C_1 N} = \frac{I_{cp} K_{VCO}(1+sRC_1)}{2\pi s^2 C_1 \cdot 32}$$

N: 16 – 32 Variable range

@ N = 32, Noise Vs BW analysis ⇒ I_{cp}, R, C_1, C_2

$$\omega_n = 2\pi f_n$$

@ N = 16, $LG = \frac{I_{cp} K_{VCO}(1+sRC_1)}{2\pi s^2 C_1 \cdot 16} \longrightarrow f_n' = 2 f_n$

$$\omega_n = \frac{I_{cp} \cdot K_{VCO} \cdot R}{2\pi \cdot 32}$$

$$\omega_n' = \frac{I_{cp} \cdot K_{VCO} \cdot R}{2\pi \cdot 16} = 2\omega_n \longrightarrow \omega_n$$

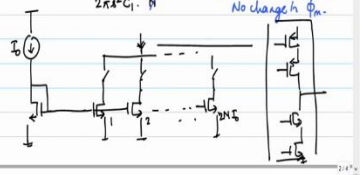
⇒ Bandwidth = f (N) and/or optimized bw from Noise Vs BW ana.

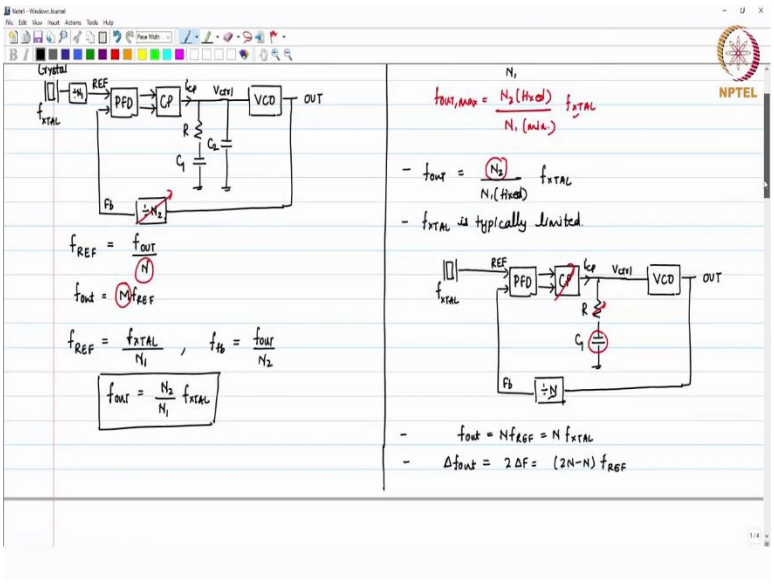
$$LG = \frac{I_{cp} K_{VCO}(1+sRC_1)}{2\pi s^2 C_1 \cdot N^2}$$

- $I_{cp} \propto N \Rightarrow I_{cp} = N I_0$

$$LG = \frac{N I_0 \cdot K_{VCO}(1+sRC_1)}{2\pi s^2 C_1 \cdot N}$$

⇒ f_n remains fixed, No change in ϕ_m.





Then you look at varying K_{VCO}. So, the case which you have seen earlier is that to control the oscillator, you had this particular current source which feeds into your ring oscillator, for example. So, you have this ring oscillator whose output voltage is something which you are controlling, so this is your V_{ctrl} and this is your VDD_{VCO}.

So, here we know the K_{VCO} is defined as follows:

$$K_{VCO} = \frac{df_{VCO}}{dV_{ctrl}}$$

If you go and look at the exact curve on f_{VCO} versus your V_{ctrl}, you may find something kind of this curve. It may or may not be linear depending on the range which you want. So, this may be the lowest frequency and this may be the highest frequency, f_{lo} and f_{high}, let me write it like this.

So, what you would like, so, when you are, think about it, when you are changing your frequency from 1280 MHz, 1.28 GHz to 640 MHz, what happens is this control voltage drops. Because of this drop in the control voltage, this voltage increases and this voltage also drops. That is how the oscillator frequency is controlled. So, this device in this whole of your transfer function, what you are looking at is the following.

So, I am just writing let us say the DC gain of this loop, the closed loop gain is 1. We know that what we are seeing here is from your V_{ctrl}, you change your PMOS gate voltage which is V_x, V_x

changes VDD_{VCO}. So, exactly you change the same voltage. And this particular loop, you have a certain loop gain and the loop gain is given by,

$$LG = g_{ma}r_a \times g_{mp}(r_{VCO}||r_{ds})$$

So, this is the DC gain we have seen and at DC, these two control voltages are same. Now, what I want to do is I know that when the frequency of the oscillator reduces, at the same time the current of the oscillator will also reduce. So, in this particular case, if you want to change the K_{VCO}, what can you do? By the way, the gain which we are having here, this gain may not be large.

So, there are a couple of ways in which you can do it. You can split this current source. So, earlier the current source which I had, a single current source like this with this node as V_x and this particular value as VDD_{VCO}. I can change this maybe just for an example, I can have V_x and a switch. So, you can have this here. Similarly, you can have multiple of these. So, you connect all of these and this is connected.

So, by doing so, what you are actually doing is you are changing your K_{VCO} and how will you change your K_{VCO}? Because you will change this g_{mp} and r_{ds}. g_{mp} and r_{ds} are the values which you will change based on your current source. So, that is something which you will do and based on this, your K_{VCO} will be changed. Now, here the difference is that the closed loop gain, if I just look at it, the closed loop gain, DC gain $\frac{VDD_{VCO}}{V_{ctrl}}$ is given by,

$$\frac{VDD_{VCO}}{V_{ctrl}} = \frac{LG}{1+LG}$$

So, ideally this is equal to 1 and depending on how much contribution you are having from here, this is going to change the K_{VCO}. Another way will be that if in case you are not using the amplifier and you only have this one directly as V_{ctrl}, connected to V_{ctrl}, so, let me just. So, if this node happens to be V_{ctrl}, then you think about it, your $\frac{VDD_{VCO}}{V_{ctrl}}$ is given by,

$$\frac{VDD_{VCO}}{V_{ctrl}} = -g_{mp}(r_{ds}||r_{VCO})$$

r_{VCO} will also change based on the current which you are having in the oscillator.

So, this is the gain which you, $\frac{VDD_{VCO}}{V_{ctrl}}$, this is the gain which you have. Now, by switching these current sources, in one case, you have all of them on and in the other case, you can think let us say few of them are on and the others are open. So, this is directly changing your g_{mp} and r_{ds} when r_{VCO} is also changed.

So, in this way also, you can actually control the K_{VCO} of the oscillator. All these things you can work out based on your design requirement. So, then the other thing which we would like to change is the resistor. Resistor and capacitor are the two other parameters which you can change. The problem while changing the resistor is, so one thing is the resistor itself is susceptible to process variation. So, if I vary resistor, ω_z also changes.

So, the problem which you will see is that if you vary the resistor, because ω_z changes, so your phase margin will also change. So, one way is the unity gain bandwidth is only proportional to R. So, you look at this expression here. So, unity gain bandwidth is this.

If the unity gain bandwidth is only proportional to R, you want to compensate for the unity gain bandwidth, but you would not like to change the zero location because that is going to compromise your phase margin. Then you can say that given the following two expressions:

$$LG = \frac{I_{CP}}{2\pi}\frac{(1+sRC_1)}{s^2C_1}\frac{K_{VCO}}{N}$$

$$\omega_u = \frac{I_{CP}K_{VCO}R}{2\pi N}$$

So, let us say if N reduces, I want to reduce my N to keep the unity gain bandwidth constant. If I do that here, it is not sufficient because if I do this, reduce R, my zero frequency changes, so I have to increase C_1. Now, if you increase C_1, you see your loop gain again drops. So, this method of changing the resistor to compensate for the change in N is not that fruitful. Because as N reduces, you reduce R, to keep the same zero frequency, you increase C. So, your bandwidth remains fixed but what happens is your overall loop gain changes. So, that is what you will see.

So, what we are doing here is the following. Let us just look at it. So, in one case, we had this particular you have s^2, so I will just write it like this. So, when N drops only, what you have is your loop gain, this is the magnitude of loop gain with respect to ω. N drops, your loop gain

changes like this. So, this is going to do this. So, effectively you can say your unity gain frequency changes.

So, how did you restore the unity gain frequency? You reduced R. So, reducing R actually shifts your, reducing R, if you just take reducing R, then what it does? Reducing R, your unity gain frequency goes high. There are two changes. One, your ω_z changes, $\omega_z = \frac{1}{RC_1}$. So, I changed my ω_z, if I change my ω_z, what I am doing is I am just shifting this curve.

So, what I am doing here is I am just shifting the zero frequency, this is what I do. I shifted my zero frequency such that my unity gain frequency remains same. But what this has done is this has actually changed our unity gain frequency. So, then if my unity gain frequency has changed, then what I would like to do is I would like to increase my C_1 to bring it back and if I increase my C_1 here, what I have done is, so I bring it back and I increase C_1 such that my zero frequency remains same, so unity gain frequency to keep the same phase margin, that is what I need to do.

Well, the curve is not exact, but what we are doing is I hope you got that idea, let me just remove this. So, by increasing C_1, effectively you can say that my N reduces, my C_1 increases, I retain the same curve. So, this curve is retained by not changing I_{CP} but by reducing R and increasing C_1. The benefit of this is that the charge-pump noise does not increase with increase in N. But what is the cost? Large capacitor is the problem, large capacitor costs area on chip.

So, what you see here is that every trick here has some trade-off associated with it. It is for the designer to decide which trade-off he or she would like to entertain. So, this is how we vary the bandwidth, we vary the output frequency or make the PLL, analog PLL, wide bandwidth. Thank you.