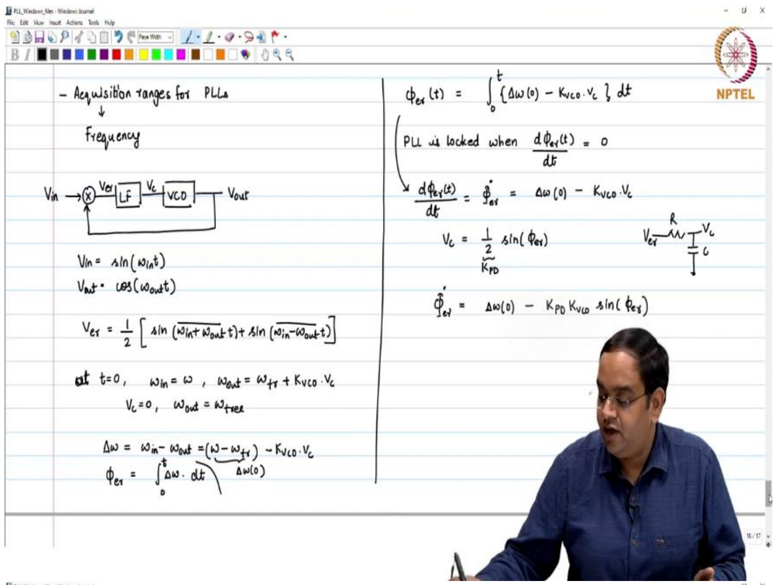
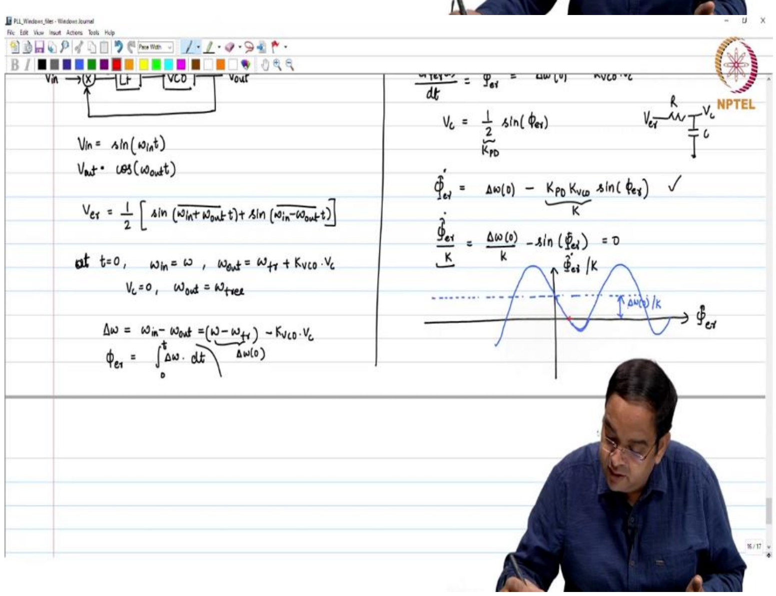
Phase-Locked Loops
Dr. Saurabh Saxena
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture – 11
Frequency Acquisition in Type-I PLLs

(Refer Slide Time: 00:15)





Hello. Welcome to this session on PLLs. In the previous session, we talked about the acquisition ranges for PLLs. We discussed about hold-in range, lock-in range and pull-in range. Now, it is time to work out and see that how we can find the different ranges for the frequency acquisition in

PLLs. By the way, when we are talking about the acquisition here, it is referring to frequency acquisition.

So, we will continue with our simple PLL example, for the PLL which had the mixer followed by loop filter which was followed by VCO. We have been studying this example from the beginning. So, it is always good to figure out all the things related to that PLL. So here, V_{in} and V_{out} are as shown. I am repeating these things so many times that now you know by just looking at the model.

So, we have,

$$V_{in} = \sin(\omega_{in} t)$$

$$V_{out} = \cos(\omega_{out} t)$$

We are going to find out the different frequency acquisition ranges for this particular PLL. This is V_{er}, this is V_c. Now, in this case, we have worked out earlier that the error voltage is given by,

$$V_{er} = \frac{1}{2}\left[\sin\big((\omega_{in} + \omega_{out})t\big) + \sin\big((\omega_{in} - \omega_{out})t\big)\right]$$

Now, in this case, to begin with, we need to find out the frequency acquisition ranges. So, we have, let us say,

$$\text{At } t = 0,\ \omega_{in} = \omega,\ \omega_{out} = \omega_{fr} + K_{VCO}.V_c$$

$$\text{Since } V_c = 0, \text{ we get, } \omega_{out} = \omega_{fr}$$

If that is the case, we have a frequency error between ω_{in} and ω_{out}. So, I will write that as,

$$\Delta\omega = \omega_{in} - \omega_{out} = \left(\omega - \omega_{fr}\right) - K_{VCO}.V_c$$

This is what we have. Now, given this frequency error, we know that the phase error at any given time is given by,

$$\varphi_{er}(t) = \int_0^t \{\Delta\omega(0) - K_{VCO}.V_c\}\, dt$$

where, $\omega - \omega_{fr} = \Delta\omega(0)$

Now, in this particular example, you know what the phase error for this is. We know that the PLL will be locked when the rate of change of phase error with respect to time is equal to zero. Thus, we have,

$$\frac{d\varphi_{er}(t)}{dt} = 0$$

Now, we have,

$$\frac{d\varphi_{er}(t)}{dt} = \dot{\varphi_{er}} = \Delta\omega(0) - K_{VCO}.V_c$$

And what is V_c in our case? You are using a loop filter. So, if we are using a low pass filter with R and C like this, where the filter bandwidth is larger than the frequency error which we have, then in that case, I can write V_c at any given time as follows:

$$V_c = \frac{1}{2}\sin(\varphi_{er})$$

where, $\frac{1}{2} = K_{PD}$, the phase error detector gain.

So, writing it again from the phase error derivative equation, we get,

$$\dot{\varphi_{er}} = \Delta\omega(0) - K_{PD}K_{VCO}\sin(\varphi_{er})$$

So, your PLL is going to lock only when you have a solution for this equation. So, what I will do is that I will try to find the solution of this equation. Now, you see, here we are taking the derivative of the phase error and phase error is in the argument also.

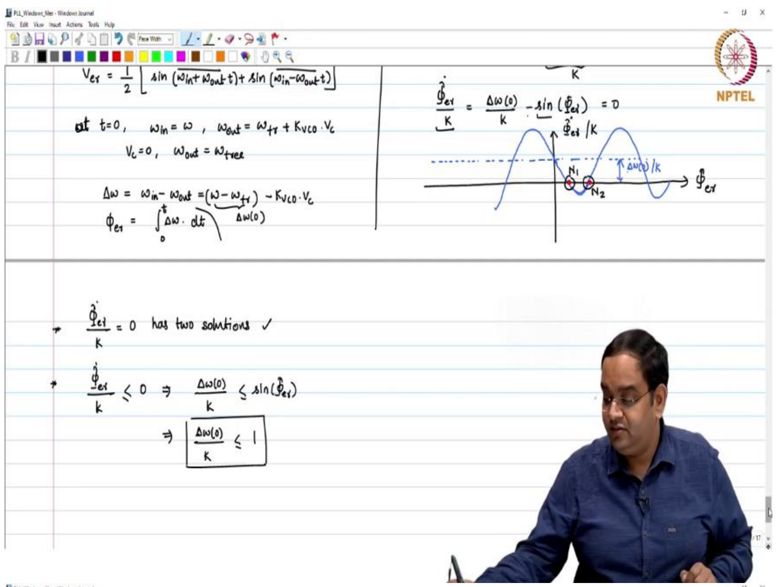
So, to find the solution, one easy way will be that if I look at the metric, $\frac{\dot{\varphi_{er}}}{K}$, and equate it to zero, which is given as,

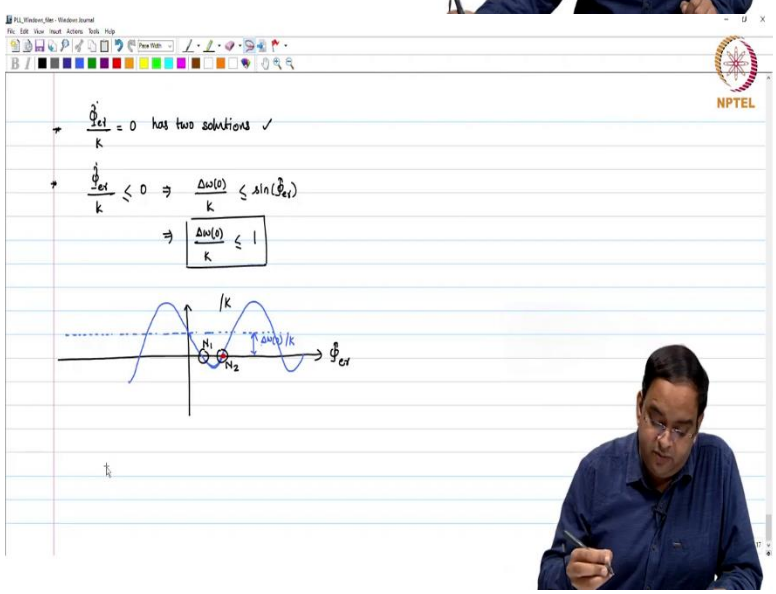
$$\frac{\dot{\varphi_{er}}}{K} = \frac{\Delta\omega(0)}{K} - \sin(\varphi_{er}) = 0$$

If I find the solution for the phase error, then I can say I have a solution. Here, $K_{PD}K_{VCO} = K$. So now, just see if you have a solution. So, I am going to plot $\frac{\dot{\varphi_{er}}}{K}$ with respect to the phase error.

So, with respect to phase error, $\frac{\dot{\varphi_{er}}}{K}$, you see that you are having $-\sin(\varphi_{er})$ here. So, $\Delta\omega(0)$ is the initial frequency error which you have, and K is constant. So, this is quite simple. So, what you will see here is, you will have inverted sine shifted by $\frac{\Delta\omega(0)}{K}$. So, what you are going to see here is, my plot may not be the exact sin wave but that is what you are going to see. Here, whatever shift you are seeing, this shift is nothing but $\frac{\Delta\omega(0)}{K}$. So, in order to find the solution for the above equation, we need to just know wherever this variable is equal to zero and we see that you are seeing two such points, one is this, the other is this.

(Refer Slide Time: 09:20)





So, the equation $\frac{\dot{\varphi_{er}}}{K} = 0$ has two solutions as shown in the plot. I call this as N_1, the left one, and the right one, I call as N_2. We are not making any approximations in the solution here. So, there are two solutions N_1 and N_2, nulls N_1 and N_2. We need to find out which is the stable solution. So, whenever you solve for some roots, you always validate whether the roots are correct in any given equation or whether the solutions hold. So, in this particular case, you see that there are two solutions, N_1 and N_2. The other thing which you also observe is that the two solutions which you are seeing exist only when $\frac{\Delta\omega(0)}{K}$ is some particular value. You may not have solutions always. If you take this up, the shift is on the upside, then you will know that $\sin(\varphi_{er})$ is going to limit the nulls at this point. You are going to have solutions only when,

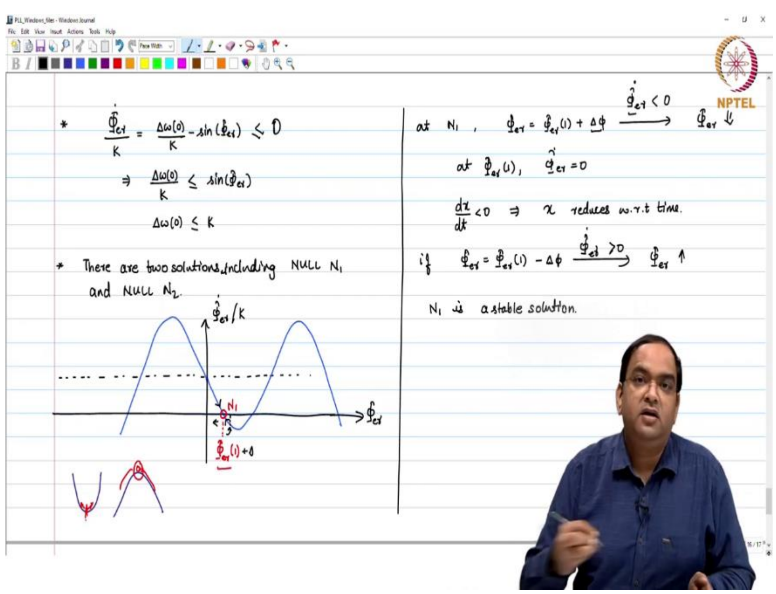
$$\frac{\dot{\varphi_{er}}}{K} \leq 0 \Rightarrow \frac{\Delta\omega(0)}{K} \leq \sin(\varphi_{er})$$

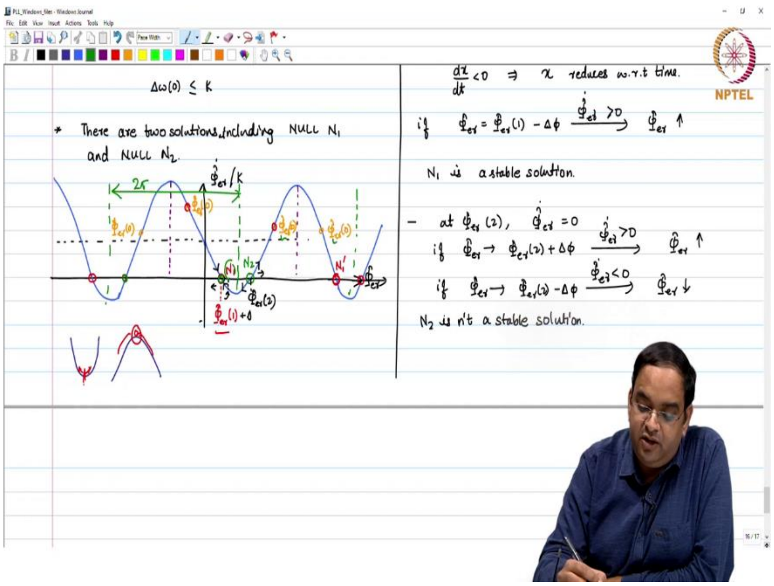
At max, $\sin(\varphi_{er})$ can be equal to 1. So, we get,

$$\frac{\Delta\omega(0)}{K} \leq 1$$

So, a solution exists only under this condition, but there are two solutions. So, the question is, which solution is the correct solution here or both are correct or one of them is correct. If one of them is correct, then, which one is correct. To find that out, what you need to do is, you need to look at this curve. So, let us look at it, one by one.

(Refer Slide Time: 11:52)





From our solutions, we see that there are two things which we need to find out which we need to make sure that a solution exists for the above plot. The first thing is given as,

$$\frac{\dot{\varphi_{er}}}{K} = \frac{\Delta\omega(0)}{K} - \sin(\varphi_{er}) \leq 0$$

Only then, you will find that there are solutions. And this will happen only if,

$$\Rightarrow \frac{\Delta\omega(0)}{K} \leq \sin(\varphi_{er})$$

This means that there is an upper limit on the frequency error which you can have, which is K. $\sin(\varphi_{er})$ at maximum will be equal to 1. So, we get,

$$\Delta\omega(0) \leq K$$

A similar limit exists on the other side of ω input. The other thing which we see from here is that there are two solutions including null N_1 and null N_2. Which of these solutions is a stable solution? That is something which we need to figure out. So now, let me just plot what we have seen before, in a little bit amplified manner. So, I have this thing here, I will just plot it here. So, what I am plotting is, as you see in the previous page, this is $\frac{\dot{\varphi_{er}}}{K}$.

There are two solutions as I told, one is null N_1 and the other is null N_2. So, let us look at the solution, null N_1. So, for this particular phase error, I call this as $\varphi_{er}(1)$, for this particular phase error, you see that the derivative of the phase error is equal to 0. So, you may think that this is a solution. Now, a solution is a stable solution only if you make any disturbance to the system at this particular phase error, the system returns back to N_1.

If you are looking at the stability, so I will just give you another example that this particular system here, let us say, if you have a ball in the valley and you move this on either side, it will come back to the center position. So, the ball is actually stable in the valley at this point. Similarly, if you have a hill and you keep a ball here and you make a small disturbance in the system, it will either roll this way or roll the other way. So, the ball here is not in equilibrium, it is not stable whereas a ball here, in this case, is stable. A similar analogy will apply here.

If N_1 is a stable solution or $\varphi_{er}(1)$ is a stable solution, then in this particular case, we make a small disturbance at N_1 position. So, how are we going to make a small disturbance at N_1? At N_1, I apply the following:

$$\varphi_{er} = \varphi_{er}(1) + \Delta\varphi$$

So, you make a Δ change in the phase, you can apply that disturbance. If you apply the disturbance, then looking at this particular curve, what you see is that the derivative of the phase error is negative, that is, $\dot{\varphi_{er}} < 0$. So, you increase the phase error, but the slope of the phase error is actually negative with respect to the phase error. $\dot{\varphi_{er}}$ is actually negative with respect to time. So, phase error will reduce. I hope you understand this part. At null N_1, if this is a solution, then you know,

$$\text{At } \varphi_{er}(1),\, \dot{\varphi_{er}} = 0$$

But at $\varphi_{er} > \varphi_{er}(1)$, it is like you make a disturbance, you add this $\Delta\varphi$, what happens is, because the derivative of the phase error is negative or the slope is negative, you will reduce the phase error.

So, if I have a variable, x, and let us say, we have,

$$\frac{dx}{dt} < 0$$

It implies that x reduces with respect to time and the slope is negative. Similarly, I made a change to the phase error or I applied a small disturbance at N_1. But, because the slope at $\varphi_{er}(1) + \Delta\varphi$ is negative, so, the phase error will drop. If the phase error drops, what will happen? It will come back to N_1. Similarly, if you make a small disturbance on the negative side, and $\varphi_{er} = \varphi_{er}(1) - \Delta\varphi$, since $\dot{\varphi_{er}} > 0$, the phase error will increase. The phase error increases which means you come back to N_1. So, N_1 is a stable solution here because if you make any disturbance in this particular case, it moves back to the same point.

Now, on the other side, you look at N_2. If you look at N_2, then I call this solution as $\varphi_{er}(2)$.

$$\text{At } \varphi_{er}(2),\ \dot{\varphi_{er}} = 0$$

So, it appears to be a solution. But if φ_{er} changes from $\varphi_{er}(2)$ to $\varphi_{er}(2) + \Delta\varphi$, so, I made a small change on this side, the slope of the phase error is actually greater than zero. You move from zero, the slope is positive. If the slope is positive, what will happen? The phase error will increase. So, you move away from node N_2. Similarly, if the phase error changes from $\varphi_{er}(2)$ to $\varphi_{er}(2) - \Delta\varphi$, because $\dot{\varphi_{er}} < 0$ on this side, the phase error will reduce. That means, you go away from N_2. So, N_2 is not a stable solution.

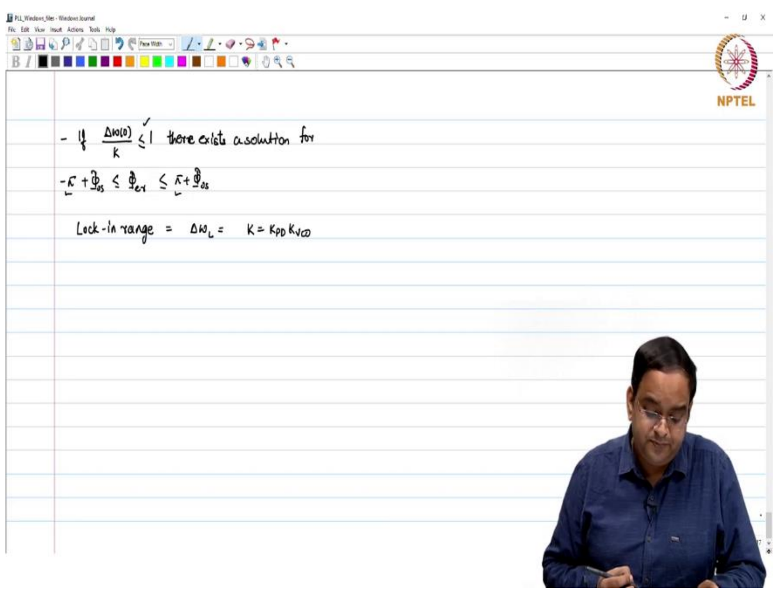
So, what you see here is that two solutions exist. Two phase error values exist which give you the derivative of phase error equal to zero or you can say two solutions exist, even when you start with a frequency error in the PLL. But out of those two solutions, only one solution is stable. So, you may start your PLL, it may start with zero or some other point. Finally, it will come to N_1, in this particular scenario.

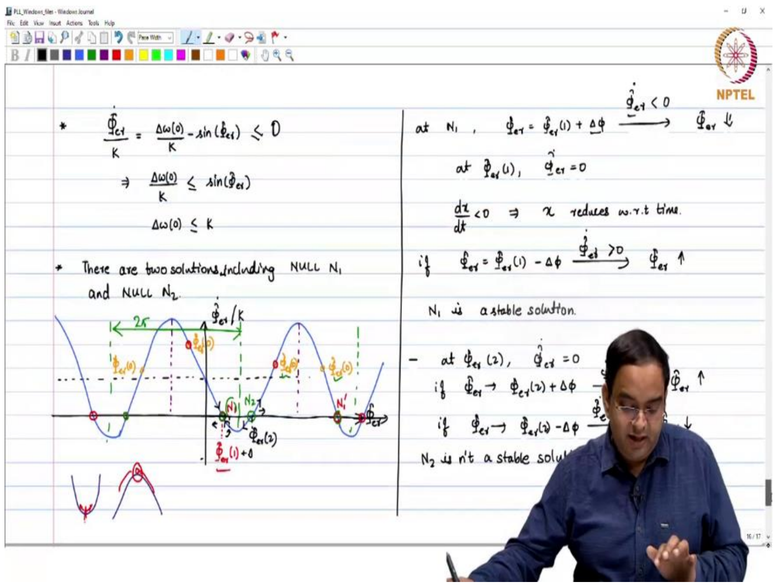
Now, there is one other important thing. I have not drawn the full sine wave but if you look at it, you will see a sine wave, because I can plot, there is nothing which is restricting me on how much of the phase error value I can plot. So, I can plot for many values. Here you see that these solutions, N_1 and N_2, they do not exist only at two particular points in this plot, but there are many such points.

Sine wave is periodic in nature and you see many solutions. An important observation here is that between every two peaks, there exist two solutions, given by N_1 and N_2. Out of those two solutions, only one is stable. Why is it important? It is important that when you start the PLL, you can start from any phase error. I can start from here, I may start from here or I may start from only this point, maybe sometimes I start from here. These are the initial phase offsets which you can start with. So, if I call this as $\varphi_{er}(0)$, this can also be $\varphi_{er}(0)$. Well, this can also be $\varphi_{er}(0)$ and this can also be $\varphi_{er}(0)$.

So, you can start from any point. The important thing here is that no matter which point you start from, your signal will change in the PLL and finally, you will find a solution at N_1. If you start from this particular point, then you will again find a solution at this point, I call this as a N_1'. So, within every 2π phase difference, that is, within every two peaks, you see that there exists a solution. So, this span is 2π and within this 2π span, you see there are two such nulls and the stable null is this particular one. You look for the other one, if you start anywhere between, if at this point or this point, you have a solution here.

(Refer Slide Time: 24:30)





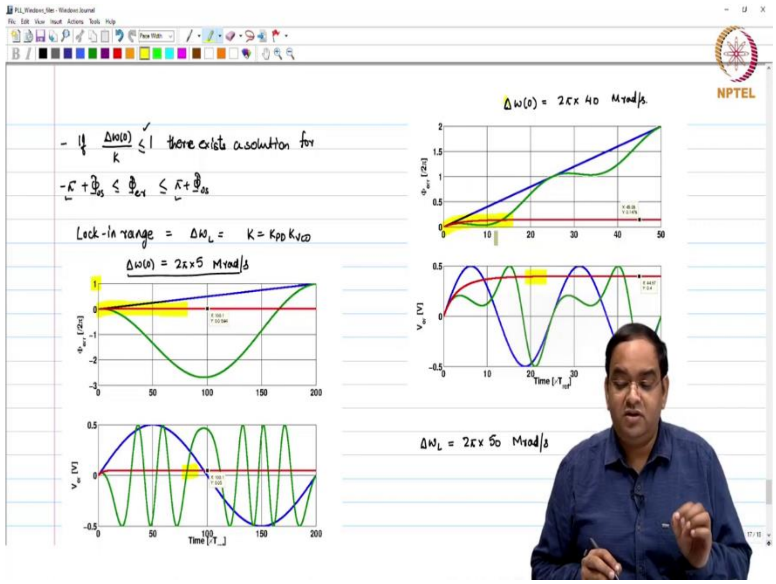
What does it mean? It means that if $\frac{\Delta\omega(0)}{K} \leq 1$, there exists a solution for $-\pi + \varphi_{os} \leq \varphi_{er} \leq \pi + \varphi_{os}$. So, there is always a solution in this range of 2π. So, if you start your PLL with initial frequency error and a solution exists, then your PLL will always lock and it will lock with phase error without exceeding 2π.

So, this is important. In this Type-I PLL which we have been discussing, given the frequency error, if there exists a solution which will be limited by $\frac{\Delta\omega(0)}{K} \leq 1$, then, the PLL will always lock without the phase error exceeding 2π. This means that the phase error does not exceed 2π and the PLL locks. So, the lock-in range is given as,

$$\Delta\omega_L = K = K_{PD}\, K_{VCO}$$

This is the lock-in range of the PLL. I will demonstrate this with the help of simulations which you have also seen before but not in this particular context directly. So, let me show what we mean here.

(Refer Slide Time: 26:51)



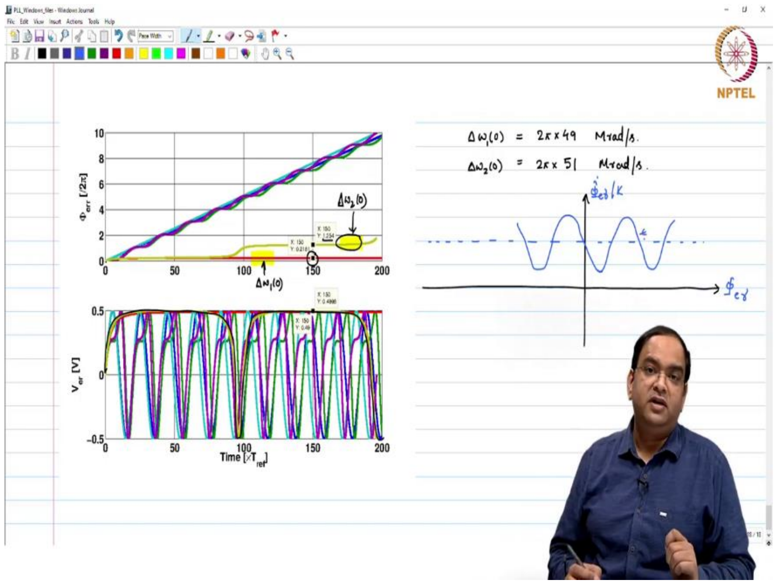
In the first case, we have the initial frequency error as given by $\Delta\omega(0) = 2\pi \times 5$ Mrad/s. So, the important part here is that the phase error is small and in this particular example, I was using the lock-in range, $\Delta\omega_L = 2\pi \times 50$ Mrad/s. So, this was the lock-in range given by $K_{PD}\ K_{VCO}$.

So, what you see here is that the phase error starts from zero, this is the phase error, it starts from zero and it settles to a value. It does not go close to 2π at all. The phase error does not even reach 2π. It settles because the frequency error was small.

Then, when I increase the frequency error, what happens? In this case, I increase the frequency error from $10\ \pi$ Mrad/s to $80\ \pi$ Mrad/s. So, this is $2\pi \times 40$ Mrad/s. I increase the frequency error to this value. And what you see here is that the phase error increases and it actually settles. This also does not increase more than 2π.

You have seen the error voltage in both the cases earlier, but the important point which you now observe is that the PLL locks without the phase error exceeding 2π. Now, what happens if the phase error exceeds 2π? Let me show you.

(Refer Slide Time: 28:57)



So, I am showing you for two different cases as given below.

$$\Delta\omega_1(0) = 2\pi \times 49 \text{ Mrad/s}$$

$$\Delta\omega_2(0) = 2\pi \times 51 \text{ Mrad/s}$$

This is the frequency error, the one which you see here is for the red one, where you see that this particular red one is the case for $\Delta\omega_1(0)$ and this one is the case for $\Delta\omega_2(0)$, these are the phase errors. So, this one is for $\Delta\omega_1(0)$ and this one is for $\Delta\omega_2(0)$. So, what you see here is that in closed loop, for the first case, $\Delta\omega_1(0)$, the phase error starts from zero because the initial offset was zero and it reaches to this value which is $0.2181 \times 2\pi$. It does not reach 2π at all, whereas in the second case which you see, the phase error value is $1.25 \times 2\pi$. So, the phase error exceeds 2π because the phase error is larger than the lock-in range.

So, there were two conditions which we initially told that $\frac{\Delta\omega(0)}{K} \leq 1$ and that there always exists a solution within the phase error of 0 to 2π. If you exceed that, the PLL will not even lock. So, that is something which we saw here using three different examples. The first example was of a low frequency error, then you had a little larger frequency error and then very close to the lock-in range.

So, for the case in which the frequency error is very close to the lock-in range, you see that the phase error exceeds 2π and the PLL does not lock. How will you make sure that the PLL does not lock? The corresponding error voltage is shown at the bottom and you see that the corresponding error voltage does not settle, it just keeps on repeating. Why is it doing this kind of weird behavior is something which we will check out.

You would want to understand the case in which there exists no solution. Well, you have already seen the plot where $\frac{\Delta\omega(0)}{K} > 1$, then, in that particular case, you may see the same plot, $\frac{\dot{\varphi_{er}}}{K}$, as something like this. You do not see this particular plot crossing φ_{er} axis at all. So, there exists no solution. Thank you.