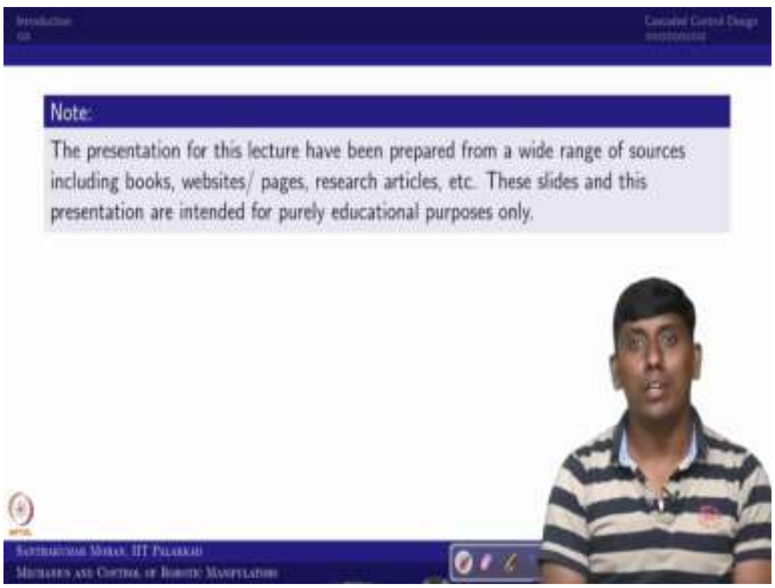
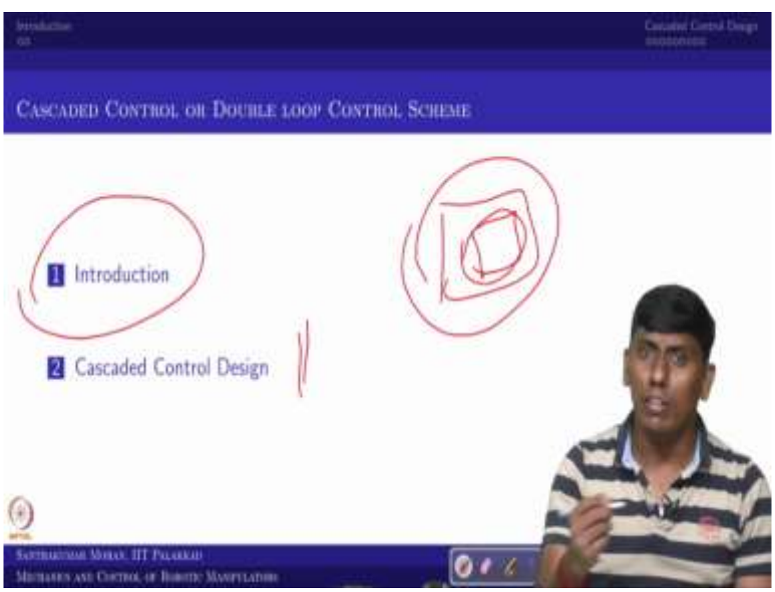
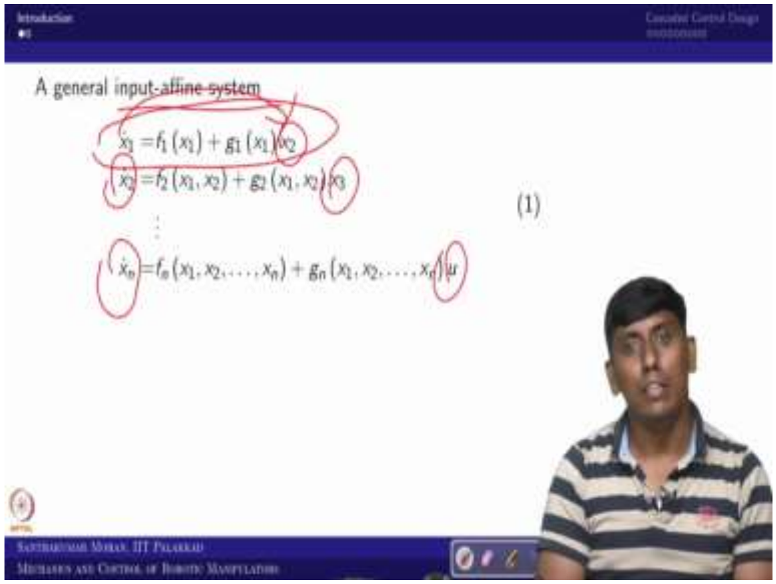
Mechanics and Control of Robotic Manipulators
Professor Santhakumar Mohan
Department of Mechanical Engineering
Indian Institute of Technology, Palakkad
Lecture: 41
Simulations Related to Dynamic Control Schemes Using MATLAB

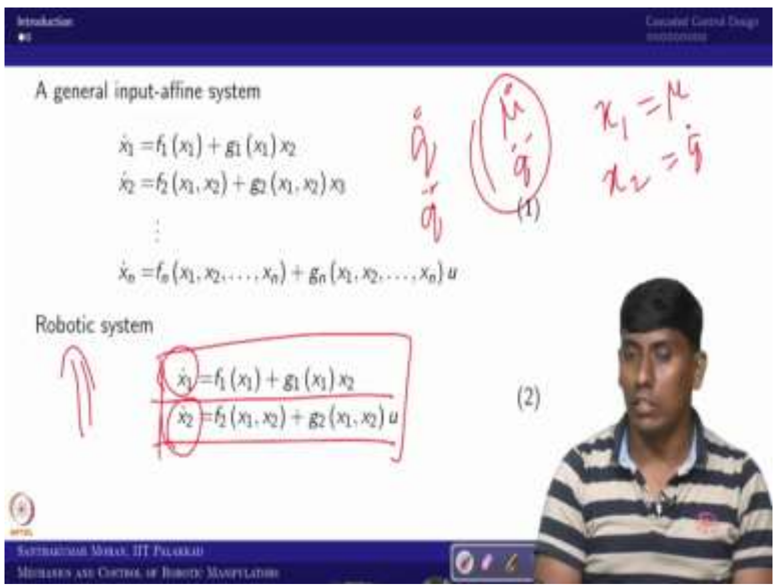
Welcome back to Mechanics and Control of Robotic Manipulator. So, this particular lecture we are going to see the extension of the dynamic control. So, in fact it is like integration of kinematic and dynamic control so that is why people call a dual loop.

(Refer Slide Time: 00:28)









So, we will see like how the dual loop can be incorporated. In that dual loop what we can have advantage, so, we can control the kinematic level control on the task space, or you can say the task space position can be controlled directly then we will go to the dynamics in the joint space. So, what we get advantage that tau we can directly get with the combination of mu and q.

So, that is what we are trying to see. So, this some people call cascaded because it is a dual loop are why we call cascaded because, the velocity we are like controlling based on the position control output. So, that is why it is called cascaded. So, now it is like a two combined thing. So, the inner and the outer, so would be controlling that is why it is called a cascaded.

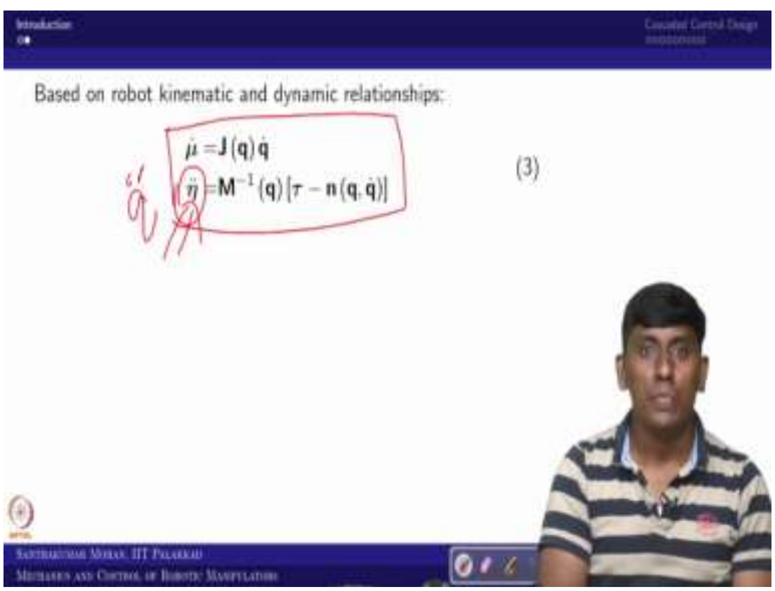
So, we will see even little more detail. So, far that one important condition that the system what we call nonlinear system should be having in the form of input affine. So, what that mean, so, it should be any subsystem you take. So, that should be assigned with one you can say output or input in other way around.

So, this output is like affine with some input. So, for example, x2 dot is having x3 as the you can say independent input. So, when you call x dot n, so that is like having an independent variable called u. So, if this is a form, we call input affine system. Fortunately, our mobile robot or even our manipulator has this kind of advantage.

So, what that mean, so, our robotic manipulator dynamic model and kinematic model can be written in this form. So, that is what we are like trying to write. So, since a robotic system is a second order system, so, we should have it two you can say subsystem one is x1 dot the other one is x2 dot so the x1 dot can be written as q dot and x2 dot can be written as q double dot or we can take mu dot and q double dot so like that we can take.

So, we are going to take this mu dot and q double dot in the sense so, x1 is like mu x1 and x2 is like q dot so this is what we are going to take. So, for that we use robot kinematic model in the sense differential kinematic model as first subsystem and the dynamic model in joint space is the second subsystem.

(Refer Slide Time: 02:58)



Based on robot kinematic and dynamic relationships:

$$\begin{aligned} \dot{\mu} &= \mathbf{J}(\mathbf{q})\,\dot{\mathbf{q}} \\ \ddot{\eta} &= \mathbf{M}^{-1}(\mathbf{q})\left[\tau - \mathbf{n}(\mathbf{q},\dot{\mathbf{q}})\right] \end{aligned} \tag{3}$$

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{f}_1(\mathbf{x}_1) + \mathbf{g}_1(\mathbf{x}_1)\,\mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= \mathbf{f}_2(\mathbf{x}_1,\mathbf{x}_2) + \mathbf{g}_2(\mathbf{x}_1,\mathbf{x}_2)\,\mathbf{u} \end{aligned} \tag{4}$$

Introduction

Cascaded Control Design

Based on robot kinematic and dynamic relationships:

$$\begin{aligned} \dot{\mu} &= \mathbf{J}(\mathbf{q})\,\dot{\mathbf{q}} \\ \ddot{\eta} &= \mathbf{M}^{-1}(\mathbf{q})\left[\tau - \mathbf{n}(\mathbf{q},\dot{\mathbf{q}})\right] \end{aligned} \tag{3}$$

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{f}_1(\mathbf{x}_1) + \mathbf{g}_1(\mathbf{x}_1)\,\mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= \mathbf{f}_2(\mathbf{x}_1,\mathbf{x}_2) + \mathbf{g}_2(\mathbf{x}_1,\mathbf{x}_2)\,\mathbf{u} \end{aligned} \tag{4}$$

where

$\mathbf{x}_1 = \mu,$

Introduction

Cascaded Control Design

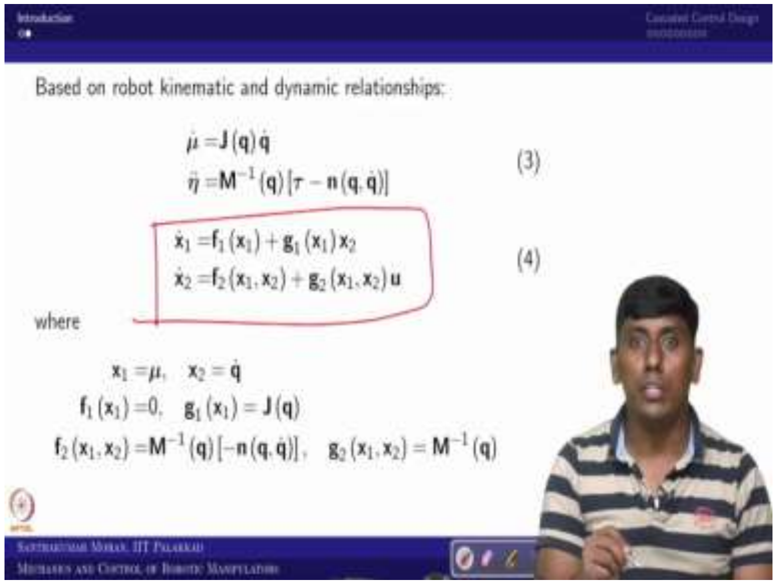
Based on robot kinematic and dynamic relationships:

$$\begin{aligned} \dot{\mu} &= \mathbf{J}(\mathbf{q})\,\dot{\mathbf{q}} \\ \ddot{\eta} &= \mathbf{M}^{-1}(\mathbf{q})\left[\tau - \mathbf{n}(\mathbf{q},\dot{\mathbf{q}})\right] \end{aligned} \tag{3}$$

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{f}_1(\mathbf{x}_1) + \mathbf{g}_1(\mathbf{x}_1)\,\mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= \mathbf{f}_2(\mathbf{x}_1,\mathbf{x}_2) + \mathbf{g}_2(\mathbf{x}_1,\mathbf{x}_2)\,\mathbf{u} \end{aligned} \tag{4}$$

where

$\mathbf{x}_1 = \mu,\ \mathbf{x}_2 = \dot{\mathbf{q}}$

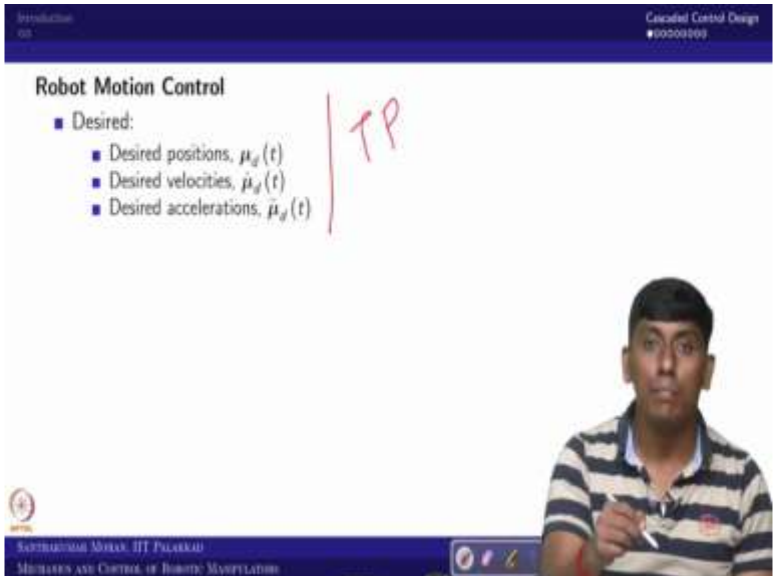


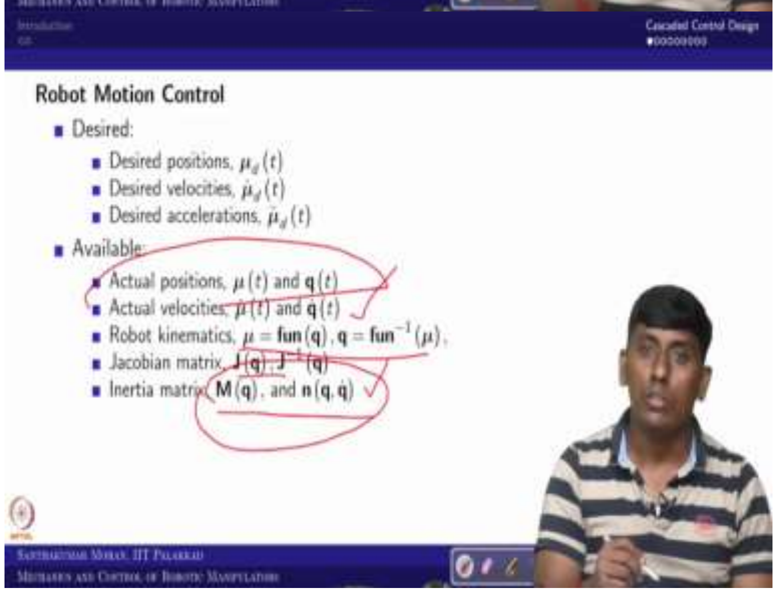
So, now, you can like see it, so, this way. So, here like we can like rewrite this eta double dot as q double dot so in order to give consistency for us. So, because this is the most of the robotic community people call eta double dot so what we have used from conventionally q double dot so we can use it that. So, in that case what we can see.

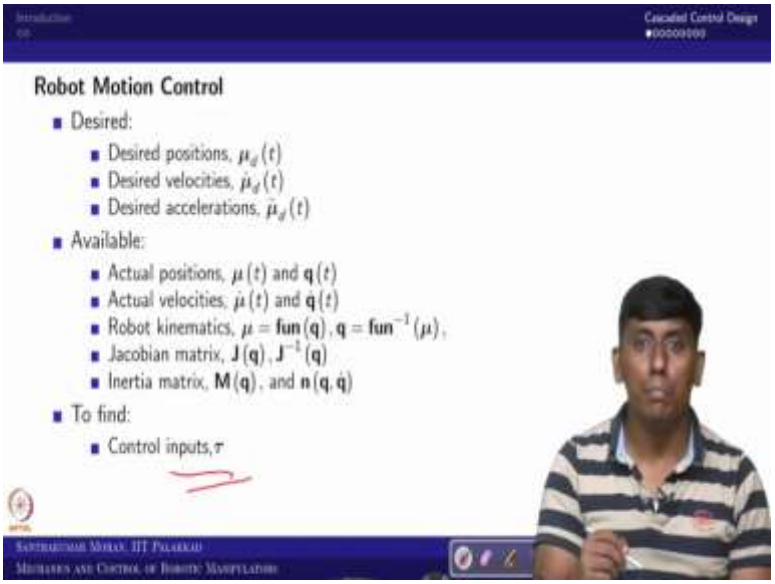
So, we can like get it x1 dot and x2 dot in this form. So, now we can see like here for f1 of x1 so in this case there is nothing because this is related to x2. So, in the sense so, you are f of x1 is not any having any value. So, I assume that x1 is mu and x2 is like q dot so then I can like find the other variables like this.

So, now based on this what I can do, I can go to the. What you call this cascaded control design then I can like bring back what is supposed to be tau. So, in that sense, I am like taking only these two are the equations. Based on this I am trying to derive the control design. So, in that sense, what we are trying to do it.

(Refer Slide Time: 04:18)



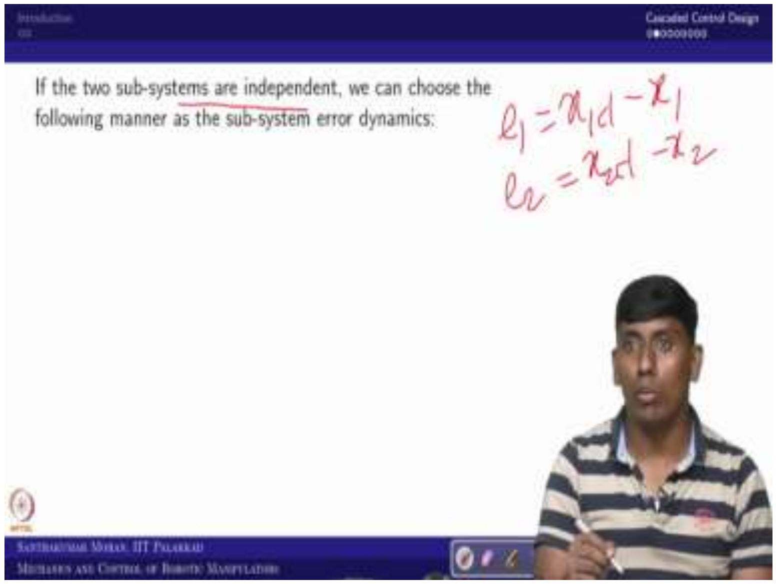


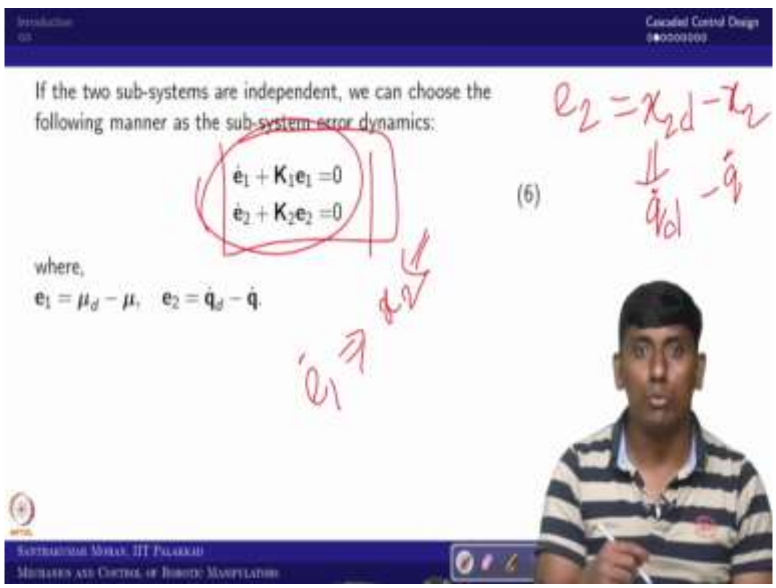


So, we are like trying to take the motion control. So, what we assume that the mu desired mu desired dot mu double dot desired all obtainable which is like from the trajectory planner. So, then we are like assuming that the actual position and you can see both task space and joint space are available, both task space velocity and joint space velocities are available.

So, the forward and inverse dynamic model and you can see kinematic model are available in the sense J of q and mu and the q function inverse of mu and J inverse are available. Further if you want to do it in model-based control then the inertia matrix and other effects are available. Since we are talking about cascade control design that is a modified version of computer torque control you can take it. So, in that meaning, so the M of q and n of q comma q dot we are going to use it. So, now we are trying to find out the control input.

(Refer Slide Time: 05:21)

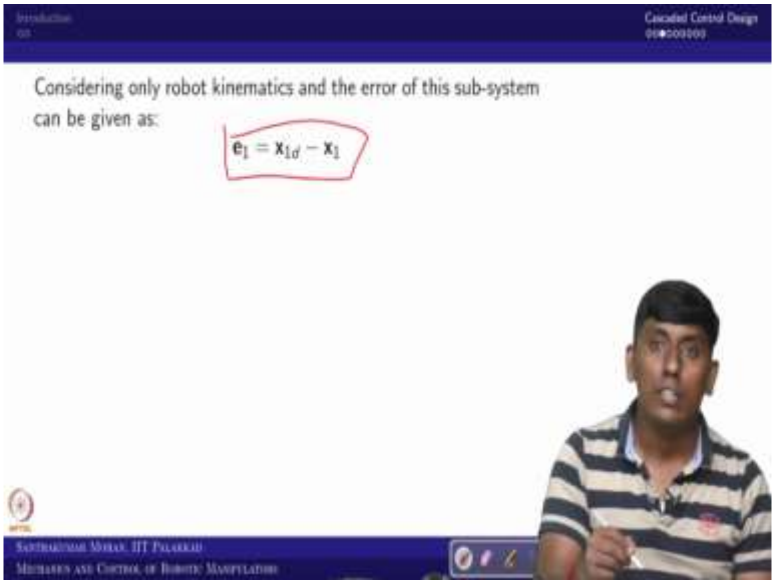


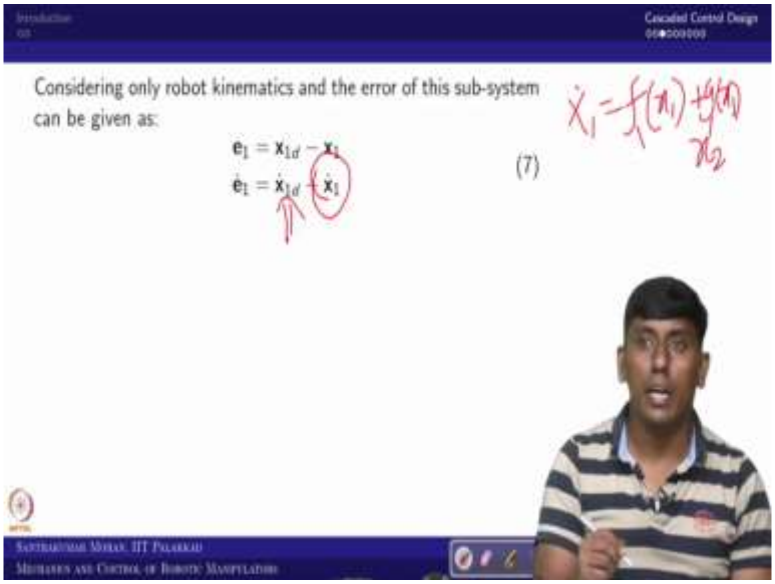


So, for that first we are taking two subsystems are independent. So, what subsystems we have taken x1 dot and x2 dot we have taken these two subsystems are independent then what we can say so, e1 is like x1 desired minus x1 and e2 is like x2 desired minus x2. So, these two error dynamics are like supposed to be stable.

So, in that sense what I can say so, e1 dot plus K1 e1 equal to 0 and e2 dot plus K2 e2 equal to 0 which we have learned in the kinematic control the same thing so, if it is a two independent system this is possible. But what happened the e2 which is like you are writing x2 desired minus x2 which is like you can see this is. So, q dot desired minus q dot whereas where as so, you can see like the e1 dot we are like getting the input which is like x2. So, in that case so, x2 desired I cannot give if I assume that these two systems are combined. So, then what I can do it so, if these two are not independent subsystem, then I have to bring that error also into case.

(Refer Slide Time: 06:41)





Considering only robot kinematics and the error of this sub-system can be given as:

$$\mathbf{e}_1 = \mathbf{x}_{1d} - \mathbf{x}_1$$
$$\dot{\mathbf{e}}_1 = \dot{\mathbf{x}}_{1d} - \dot{\mathbf{x}}_1 \qquad (7)$$

where

$$\dot{\mathbf{x}}_1 = \mathbf{J}(\mathbf{q})\,\dot{\mathbf{q}}$$
$$= \mathbf{g}_1(\mathbf{x}_1)\,\mathbf{x}_2 \qquad (8)$$

Considering $\mathbf{x}_2$ as an input to this sub-system and make the $\mathbf{e}_1$ tends to zero when the time t, tends to infinity.

Considering only robot kinematics and the error of this sub-system can be given as:

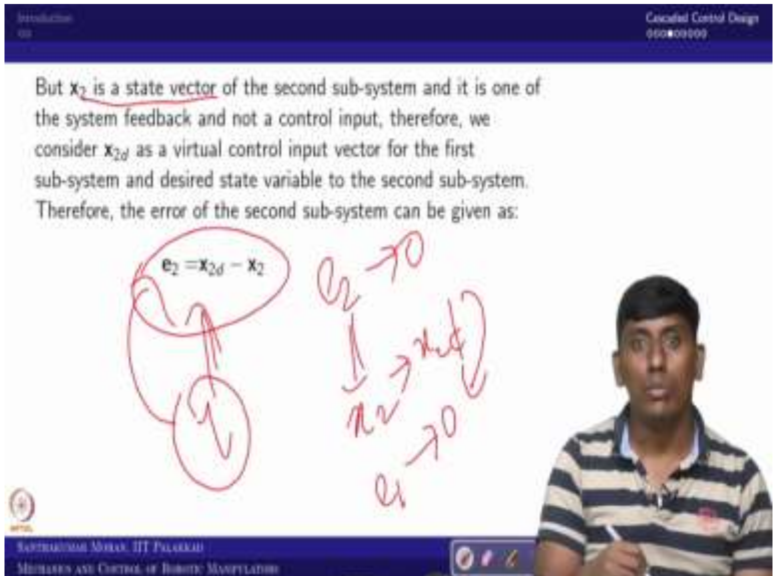
$$\mathbf{e}_1 = \mathbf{x}_{1d} - \mathbf{x}_1$$
$$\dot{\mathbf{e}}_1 = \dot{\mathbf{x}}_{1d} - \dot{\mathbf{x}}_1 \qquad (7)$$

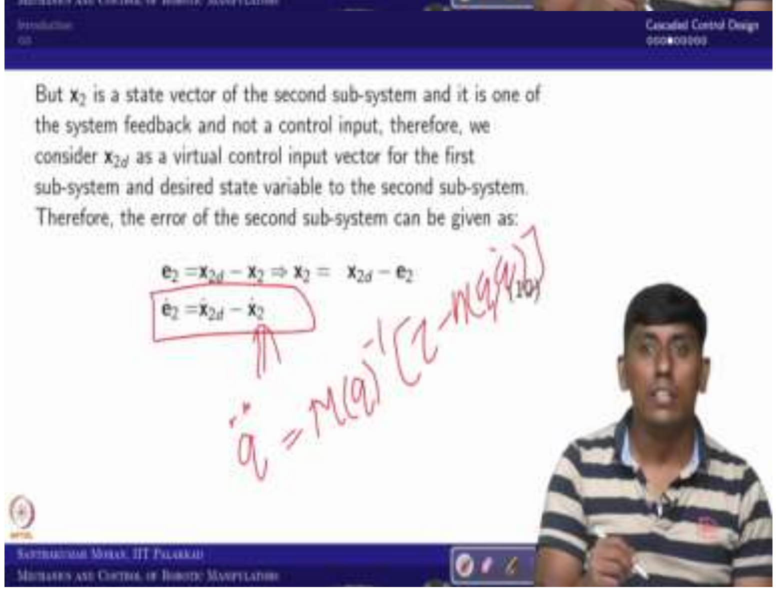
where

$$\dot{\mathbf{x}}_1 = \mathbf{J}(\mathbf{q})\,\dot{\mathbf{q}}$$
$$= \mathbf{g}_1(\mathbf{x}_1)\,\mathbf{x}_2 \qquad (8)$$

Considering $\mathbf{x}_2$ as an input to this sub-system and make the $\mathbf{e}_1$ tends to zero when the time t, tends to infinity.

$$\mathbf{x}_2 = \mathbf{g}_1^{-1}(\mathbf{x}_1)\left[\mathbf{K}_1(\mathbf{x}_{1d} - \mathbf{x}_1) + \dot{\mathbf{x}}_{1d}\right] - \mathbf{f}_1(\mathbf{x}_1)$$
$$= \mathbf{g}_1^{-1}(\mathbf{x}_1)\left[\mathbf{K}_1\mathbf{e}_1 + \dot{\mathbf{x}}_{1d}\right] - \mathbf{f}_1(\mathbf{x}_1) \qquad (9)$$





So, that is what we are trying to do it here. So, the e1 is like this. So, e1 dot I am taking it here. So, now thus x1 dot desired these like what I have it from the trajectory planner, but this x1 dot so the x1 dot I can write it as. So, function of x1 plus g1 of x1 into x2. So, that way if I relate so, what I will get. So, this I can like substitute it here. So, what I will get.

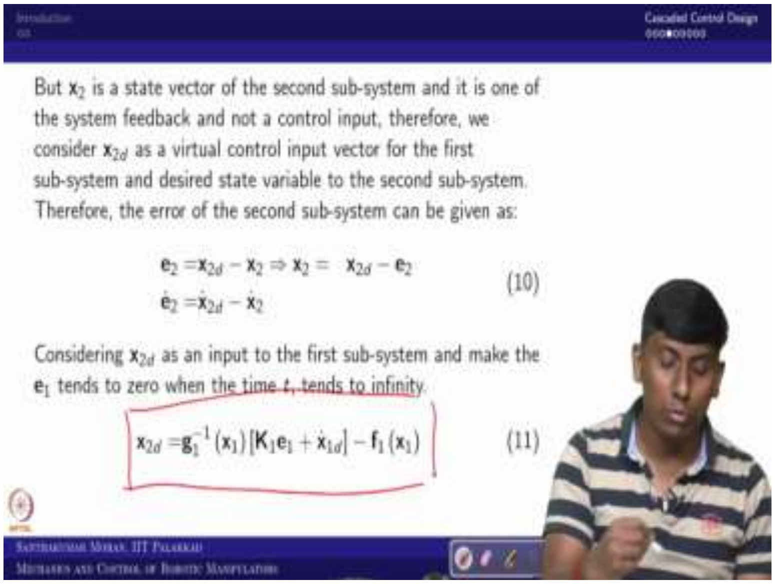
So, the x2 I will get as an input, but like this is like obtainable where the e1 tends to 0 when t tends to infinity only if so, x2 equal to x2 desired what you are planning, or you can say x2 is like independent state. But like this is like coupled system because the system is not having two subsystems, it is a single system.

So, then we have to like to bring that error. So, this is what the x2 desired supposed to be. So, what we are assuming this is the x2, but, if this is not controllable that is what I have written it here if the x2 is the state vector of a second subsystem, it is one of the system feedbacks not a control input then what we can do so, we can consider instead of what we have derived here, so, this x2 I can consider as x2 desired. So, then there would be error will come.

So, this error if I converge in the seconds of system where tau is the input if I choose a proper tau in such a way that this e2 tends to 0 then what happened so, x2 tends to x2 desired. So, then what happened e1 tends to 0. So, like that it is coupled, and it is like you can see like it is coming backward. So, that is why it is called backstepping?

So, that is what we are like trying to see. So, now based on this what we can see the x2 we can give this form. So, now, if I substitute this into the, you can see the first equation the equation would be slightly different. So, now, I am taking the derivative of this time derivative. So, e1 we know already so, the e2 dot is x2 dot desired minus x2 dot but what is x2 dot so x2 dot is nothing but, so, q double dot so if it is q double dot this we can write as so, M of q inverse into tau minus n of q comma q dot. So, that I can use then I can like make it. So, that is what we are trying to do it.

(Refer Slide Time: 09:34)



We know that, the second sub-system is as:

$$\dot{\mathbf{x}}_2 = \mathbf{f}_2(\mathbf{x}_1, \mathbf{x}_2) + \mathbf{g}_2(\mathbf{x}_1, \mathbf{x}_2)\mathbf{u} \tag{12}$$

By choosing a proper $\mathbf{u}$, the error $\mathbf{e}_2$ tends to zero when t tends to infinity.

$$\mathbf{u} = \mathbf{g}_2^{-1}(\mathbf{x}_1, \mathbf{x}_2)\left[\mathbf{K}_2\mathbf{e}_2 + \dot{\mathbf{x}}_{2d}\right] - \mathbf{f}_2(\mathbf{x}_1, \mathbf{x}_2) \tag{13}$$

Introduction
Cascaded Control Design

We know that, the second sub-system is as:

$$\dot{\mathbf{x}}_2 = \mathbf{f}_2(\mathbf{x}_1, \mathbf{x}_2) + \mathbf{g}_2(\mathbf{x}_1, \mathbf{x}_2)\mathbf{u} \tag{12}$$

By choosing a proper $\mathbf{u}$, the error $\mathbf{e}_2$ tends to zero when t tends to infinity.

$$\mathbf{u} = \mathbf{g}_2^{-1}(\mathbf{x}_1, \mathbf{x}_2)\left[\mathbf{K}_2\mathbf{e}_2 + \dot{\mathbf{x}}_{2d}\right] - \mathbf{f}_2(\mathbf{x}_1, \mathbf{x}_2) \tag{13}$$

where

$$\dot{\mathbf{x}}_{2d} = \mathbf{g}_1^{-1}(\mathbf{x}_1)\left[\mathbf{K}_1\dot{\mathbf{e}}_1 + \ddot{\mathbf{x}}_{1d}\right] - \dot{\mathbf{f}}_1(\mathbf{x}_1) + \dot{\mathbf{g}}_1^{-1}(\mathbf{x}_1)\left[\mathbf{K}_1\mathbf{e}_1 + \dot{\mathbf{x}}_{1d}\right] \tag{14}$$

Santhakumar Mohan, IIT Palakkad
Mechanics and Control of Robotic Manipulators

Introduction
Cascaded Control Design

However, in order to prove the closed-loop stability of the system, the Lyapunov's direct method can be applied here.
Let us choose a Lyapunov's candidate function as follows:

$$V(\mathbf{e}_1, \mathbf{e}_2) = \frac{1}{2}\left[\mathbf{e}_1^T\mathbf{e}_1 + \mathbf{e}_2^T\mathbf{e}_2\right] \tag{15}$$

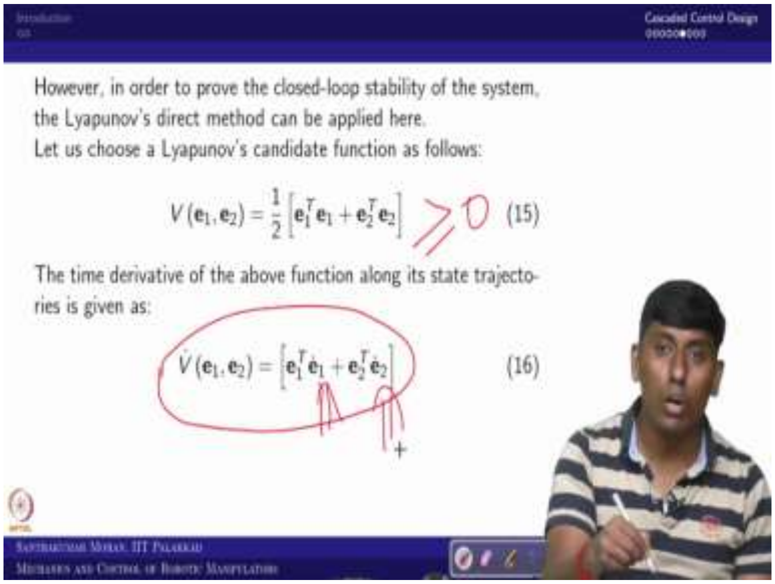
Santhakumar Mohan, IIT Palakkad
Mechanics and Control of Robotic Manipulators

So, now, we can see that the first sub system has like so, e1 to 0 when t tends to infinity then for that you need to have e2 suppose to be 0, when this e2 supposed to be 0, where the tau properly you choose so that it can be converged. So for that we are like taking it this the x2 desired. So, which we have taken so, then we are like bringing it. So, now we are like substituting this equation. So, from there we can like find that u can be one of the control inputs. So, this is what we can choose. So, in order to like make it this more clear so, we will go with a Lyapunov or you can see control stability proof.

So, where we can like see that this is the x2 dot desired we can find. But here you can see that there is a small hiccup will come because x2 dot desired minus x2 are not same there would be error that the error may like prolong somewhere. So, because, so, x2 desired is not obtainable by x2 because this is t tends to infinity only this will be obtainable.

So, in that sense there would be a residue in order to make that residue go out we can like modify this control law slightly different. So, for that we are taking a Lyapunov direct method where we are taking a Lyapunov candidate function as in this form. So, where half e1 transpose e1 plus e2 transpose e2. So, now, this is like a positive definite. So, when this would be 0 when e1 equal to 0 and e2 equal to 0 then only the V would be 0 otherwise it would be positive throughout.

(Refer Slide Time: 11:29)



However, in order to prove the closed-loop stability of the system, the Lyapunov's direct method can be applied here.
Let us choose a Lyapunov's candidate function as follows:

$$V(\mathbf{e}_1, \mathbf{e}_2) = \frac{1}{2}\left[\mathbf{e}_1^T\mathbf{e}_1 + \mathbf{e}_2^T\mathbf{e}_2\right] \tag{15}$$

The time derivative of the above function along its state trajectories is given as:

$$\dot{V}(\mathbf{e}_1, \mathbf{e}_2) = \left[\mathbf{e}_1^T\dot{\mathbf{e}}_1 + \mathbf{e}_2^T\dot{\mathbf{e}}_2\right] \tag{16}$$

But as per the Lyapunov's method, the closed-loop system is asymptotically stable only if $\dot{V}(\mathbf{e}_1, \mathbf{e}_2) \geq 0$

$$\begin{aligned}\dot{\mathbf{e}}_1 &= \dot{\mathbf{x}}_{1d} - \dot{\mathbf{x}}_1 \\ &= \dot{\mathbf{x}}_{1d} - (\mathbf{f}_1(\mathbf{x}_1) + \mathbf{g}_1(\mathbf{x}_1)\mathbf{x}_2)\end{aligned} \tag{17}$$

$$\begin{aligned}\Rightarrow \mathbf{x}_2 &= \mathbf{x}_{2d} - \mathbf{e}_2 \\ &= \mathbf{g}_1^{-1}(\mathbf{x}_1)\left[\mathbf{K}_1\mathbf{e}_1 + \dot{\mathbf{x}}_{1d}\right] - \mathbf{f}_1(\mathbf{x}_1) - \mathbf{e}_2\end{aligned} \tag{18}$$

$$\begin{aligned}\dot{\mathbf{e}}_1 &= \dot{\mathbf{x}}_{1d} - \left(\mathbf{f}_1(\mathbf{x}_1) + \mathbf{g}_1(\mathbf{x}_1)\left[\mathbf{g}_1^{-1}(\mathbf{x}_1)\left[\mathbf{K}_1\mathbf{e}_1 + \dot{\mathbf{x}}_{1d}\right] - \mathbf{f}_1(\mathbf{x}_1) - \mathbf{e}_2\right]\right) \\ &= -\mathbf{K}_1\mathbf{e}_1 + \mathbf{g}_1(\mathbf{x}_1)\mathbf{e}_2\end{aligned} \tag{19}$$

$$\begin{aligned}\dot{\mathbf{e}}_2 &= \dot{\mathbf{x}}_{2d} - \dot{\mathbf{x}}_2 \\ &= \mathbf{g}_1^{-1}(\mathbf{x}_1)\left[\mathbf{K}_1\dot{\mathbf{e}}_1 + \ddot{\mathbf{x}}_{1d}\right] - \dot{\mathbf{f}}_1(\mathbf{x}_1) \\ &\quad + \dot{\mathbf{g}}_1^{-1}(\mathbf{x}_1)\left[\mathbf{K}_1\mathbf{e}_1 + \dot{\mathbf{x}}_{1d}\right] \\ &\quad - \left[\mathbf{f}_2(\mathbf{x}_1, \mathbf{x}_2) + \mathbf{g}_2(\mathbf{x}_1, \mathbf{x}_2)\mathbf{u}\right]\end{aligned} \tag{20}$$

$$\begin{aligned}\mathbf{u} &= \mathbf{g}_2^{-1}(\mathbf{x}_1, \mathbf{x}_2)\left[\mathbf{K}_2\mathbf{e}_2 + \dot{\mathbf{x}}_{2d}\right] - \mathbf{f}_2(\mathbf{x}_1, \mathbf{x}_2) \\ &\quad + \dot{\mathbf{g}}_1^{-1}(\mathbf{x}_1)\left[\mathbf{K}_1\mathbf{e}_1 + \dot{\mathbf{x}}_{1d}\right] \\ &\quad + \mathbf{g}_2^{-1}(\mathbf{x}_1, \mathbf{x}_2)\mathbf{g}_1^T(\mathbf{x}_1)\mathbf{e}_1\end{aligned} \tag{21}$$

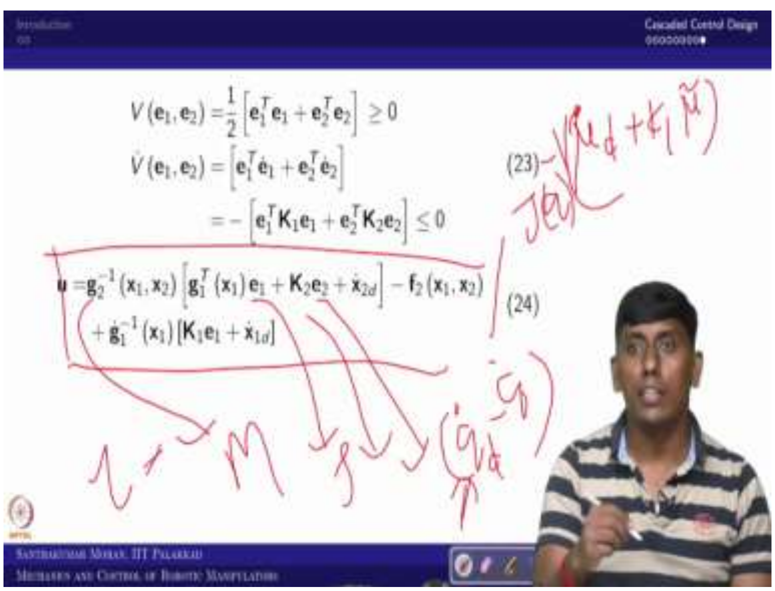
So, that is what we are like taking it. So, now this is like a positive definite. So, this would be equal to 0 when e1 and e2 both are 0. So, then I can take the time derivative of above function along its state trajectory. So, or you can say Lyapunov function derivative along with the state trajectory. So, then it comes like this.

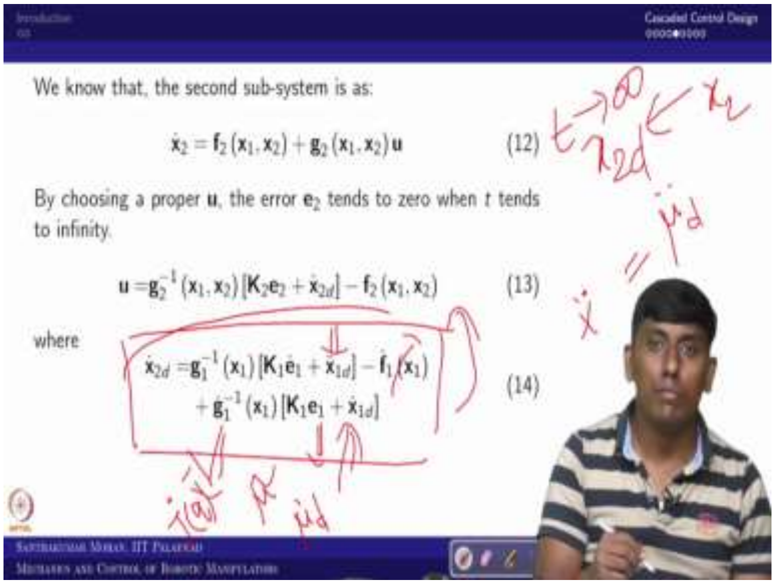
So, we already know e1 dot and e2 dot so we can substitute that. So, this is like fulfilled. So, for making a stability proof so, the V dot supposed to be negative definite so, it should be 0 when e1 and e2 are 0 otherwise it should be negative. So, that is what we are trying to prove. So, for that what we are trying to take e1 dot and e2 dot also like we can get it.

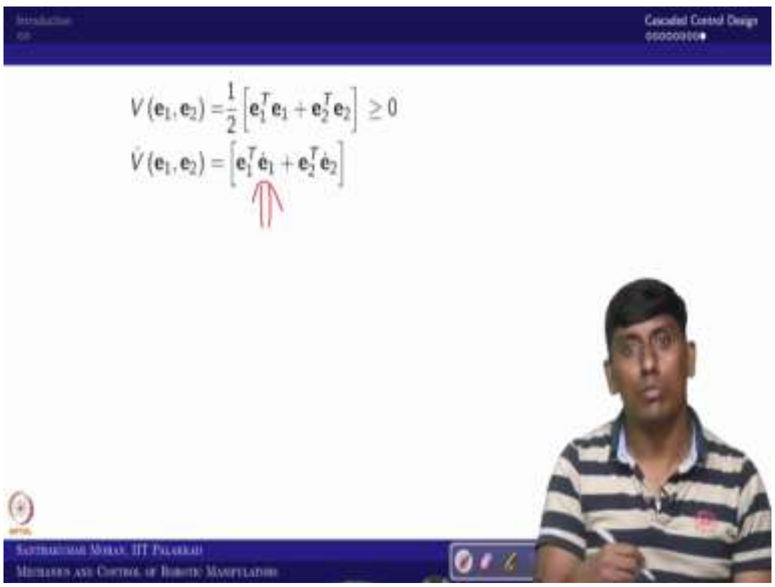
So, now you can see the e1 dot by substituting this error. So, you will get some residue down here. If it is independent, this is what is going to come since so, it is not independent so, there is a residue value. So, this value needs to be compensated now coming to e2 so, e2 dot come like this. So, we substitute everything.

So, then what we can see we if we choose u in this way so, what we can see the e2 dot would be coming minus k2 e2 plus some residue term. So, this residue term and other residue term what we obtained here these two would be you can say cancel each other when we substitute into the Lyapunov function.

(Refer Slide Time: 13:07)







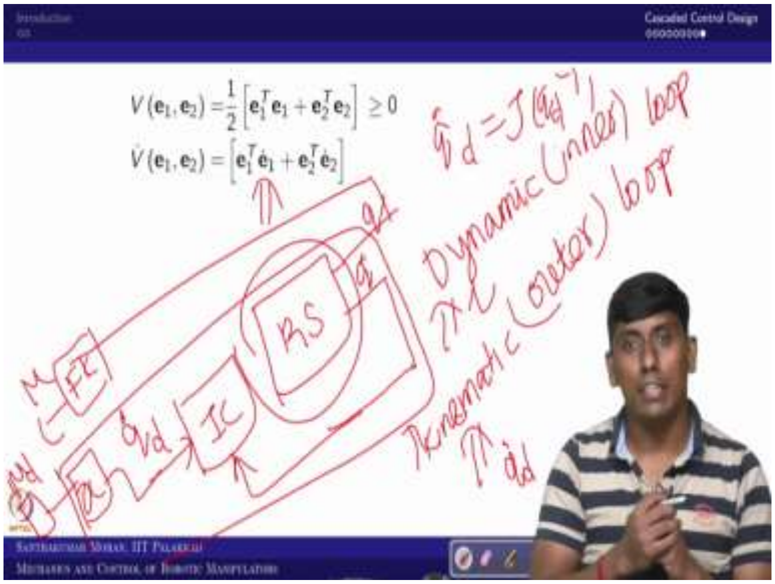
So, when we substitute here so when we substitute it here so you can see like automatically these two terms will go out and then you will end up with this, so this equation ensure that it is negative definite. So, this would be 0 when e1 and e2 are 0 only if e1 and e2 are 0 similar case to the week. So, in that sense the stability is proved.

So, based on that if you take u so, this is what the final control input. So, now you can convert into tau because u equal to tau and this become as M, and this become as a J and this become as a K2 and this e2 become q dot desired minus q dot so this q dot desired we can write as so mu dot desired plus so K1 mu tilde multiply with the J of q inverse that you can keep going.

So, like that you can make it so, for example, I just wanted to show that. So, where you can say so, what is x1 what is x2 these all we can get it here. So, you can see. So, this is x2 dot desired so, you can see this is so, x1 double dot means, so, that would be mu double dot desired. So, this becomes 0 because we have so function of x1 is no value. And this is mu desired dot.

And this is a mu tilde and this is J you can say dot of so, inverse. So, like that you can keep substitute and use it. So, in order to get this more clear, so we will do a MATLAB simulation in upcoming lecture. So, with that, we can close this particular content, then we will see some numerical problem from there we can close this entire portion.

(Refer Slide Time: 15:16)



So, I hope you are clear on. What is the cascade control design or what you call dual loop so, why it is called dual loop because it is having for example, I call this is a robotic system the robotic system will give two things. So, q and q dot this q dot would be compensated with the inner control so, where here q desired dot would be calculated.

So, this would be a calculated based on you can say forward kinematics and you have mu, and you will take mu desired, and you will compare and you take outer loop control. So, that is why it is a two stage. So, the outer loop control will give the q dot desired. So, that is why it is called cascaded, but the dual loop because the kinematic loop on the outer so kinematics or outer loop and dynamic or inner loop is inside.

So, this is controlling tau, and this is controlling q dot desired. So, now if you use the old computing torque or computer torque control the q dot desired, we can take it J you can say q desired inverse of mu desired dot so we can see like the dynamic which is inner loop so, that is making a small patch up there the robotic system the outer loop is control in task space that is the beneficial.

So, that you can do a proper motion control however, the inner loop which is controlling the what you call dynamics that directly given in the task space so the actuator can be controlled straight forward. So, I hope so you are clear on this cascaded control design. So, we will see the numerical example along with MATLAB simulation there you will be getting more clarity. So, with that, see you then thank you bye.