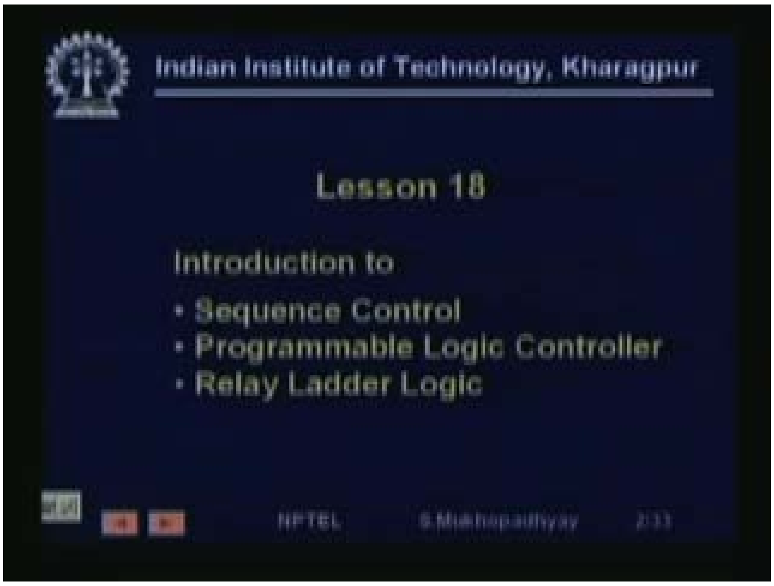
Industrial Automation and Control
Prof. S. Mukhopadhyay
Department of Electrical Engineering
Indian Institute of Technology, Kharagpur

Lecture - 18
Introduction to: Sequence Control Programmable Logic Controller Relay Ladder Logic

(Refer Slide Time: 01:03)

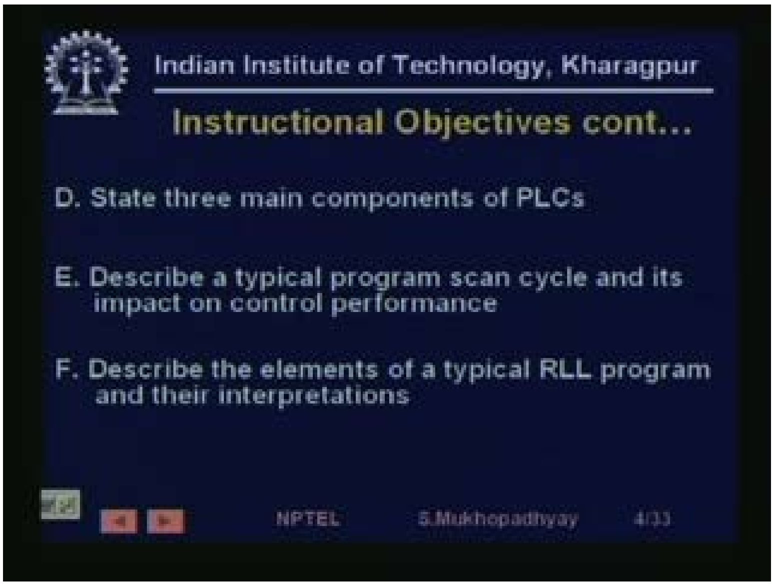


Good after noon, we are going to start lessons eighteen which is on new subject it is a new module on sequence control. So, we are going have our first introduction today on sequence control on the kinds of controllers that are used for the sequence control namely programmable logic controllers or PLCs. We are going to take a first look at the programming languages the languages, which are used to specify control programs for PLCs, so as usual we will look at the instructional objectives.

(Refer Slide Time: 01:35)

So, after learning the lesson, the student should be able to state three differences between analog and sequence control. This is important because traditionally an electrical or manufacturing or chemical engineering student is typically exposed to analog control, while the kind of control that we are considering here at different in nature. So, it is important to understand the differences between analog control and sequence control. Obviously, we should know he will be she should able to do give an example of industrial sequence control and actual practical example. The programmable logic controllers are controllers which are primarily used in the industry to implement sequence control, he should be able know three of its major functions the major functional characteristics.

(Refer Slide Time: 02:47)



We should also know the major components the major structural components of the PLC and he should be able to describe a typical program execution cycle that takes place in PLCs and how these program execution cycles affects the control performance. Finally, he should be able to know what are the typical elements of a of a of an relay ladder logic program, which is one of the programming languages which is used for programming a PLC. We know the various interpretations of the symbol used in this language, so first of all let us try to understand what is sequence or logic control.

(Refer Slide Time: 03:31)

So, as I said it is fundamentally different not fundamentally may be very significantly different from analog controls on various aspects. So, we need to understand what it is, so I constructed this is my definition words are mine, so it is a class of control problems for systems where inputs outputs and feedbacks are discrete set value. Actually, I would rather say that there are for the purpose of control it is taken that they are set values that it is the intermediate values are not considered. So, for example, a switch it stays 99.999 percent of time in on state or off state, but obviously, it goes from the on state to the on to the off state and that timing is so short that we chose to ignore it. So, we say that a disk switch is a system which stays in either one of the one of two states. So, it is either on or off we actually chose to ignore the times when it is going from on to off because it is too small and of no consequence for us.

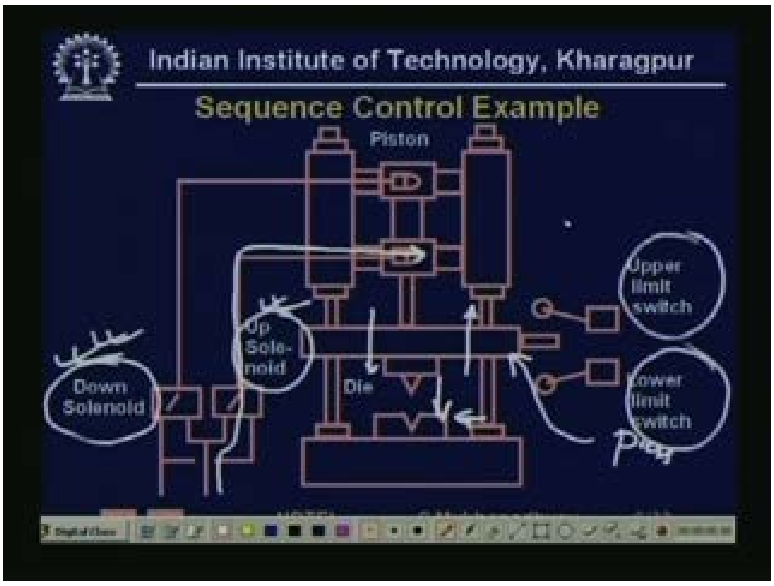
So, in that sense the inputs and outputs and feedbacks are discrete set valued like on off and the problem is to cause or prevent occurrences off either particular values of outputs or combinations of outputs. So, when the switch one is off switch one should not switch two should not be on, so you know giving a give an example from daily life if let us take I mean daily life example of a railway crossing. So, if the incoming train coming sensor is on then the gate should be off, so some particular combinations are allowed and some odd some particular combinations could be prohibited. So, the purpose of control is to ensure that in all future behavior of the system those combinations of outputs either must occur or cannot occur as is specified.

Similarly, we could also have timing restrictions, so you can say that after the train coming sensor becomes on within one minute the gate close sensor should become on. So, now you are giving some timing restriction, so you must ensure that the system that the gate becomes closed within one minute of the train coming sensor becoming on. We might also like some given sequences of output. So, for example, suppose let us see, let us take a buffer where some machine is producing parts and then putting them in the buffer and the buffer from time to time gets emptied. So, these states through which the buffer goes, so may be the buffer capacity is 2, so every time the machine is dropping a part where the buffer is getting filled.

So, the buffer should never go to the overflowing states, so it should go to part count 0 part count 1 part count 3, then empty part state and then again come to part count 0. So, the buffer should only go through certain states and should not go through go to some other states. So, only some particular sequences of output or events should be allowed or there could be some given orders between various outputs. So, after this occurred that must occur, so these are you know sequencing orders sometimes also called interlocks if this has already occurred that cannot occur. So, basically what we are trying to say is that if the system states are assumed to be discrete these kinds of statements are made about the desired behavior of the system.

Then, the problem of control is to realize them by exercising the inputs to the process, so this is typically a sequence of logic control problem why they are called sequence of logic control because it is a finite state thing. So, it is described more by logical methods, then by numerical methods before we look at them in more conceptual ways, let us take an example.

(Refer Slide Time: 08:17)



So, you know look at this is what is what it is suppose to be, it is suppose to be dye, it suppose to be stamping press you know. So, if you look at this, then this is the stamp, this is the press, so there are two solenoids, up solenoid and down solenoid. So, if the up solenoid is energies, these are these are these are hydraulic presses, so if the up solenoid press is energized, then the pressurized fluid flows through this path and goes into the machine. So, the machine is constructed in such a manner that this press goes up, similarly, if the down solenoid is energized, then this press goes down and this, so this is your metal sheet, it comes down and stamps on it.

So, it keep stamping retracting stamping retracing and creating a particular shape of sheet metal you know this things are typical used in making, let us say car bodies. So, it is a hydraulic stamp press and obviously, after the there are, so what happens is that after it goes up the up solenoid may be de energized. So, whether it is reached the top most position or the down most position this needs to be sensed. So, there are two sensors this is the upper limit switch and lower limit switch and these are two actuators the down solenoid and the up solenoid are the actuators and this thing is the is our system. So, this is a typical example of a sequence control problem, where we are interested in only seeing the system is everything is analog if you look fine enough.

So, obviously the piston cannot the system cannot instantaneously come down from the up to down state, but what we mean is that for the control of this system we are not interested or the sequence control in this of this system. We are not interested to know the intermediate states we want to only ensure that if this machine is suppose to stamp if that command is given then it should periodically come down and then go up come down and go up. So, only that we are only interested in that is our control problem is only to ensure that it periodically goes to the up down up down up down sequence that is all. So, in this sense it is a sequence control problem, so there are n number of such other problems, for example there are many other application.

(Refer Slide Time: 11:48)

For example, a plant start up shut down sequence, so when a plant is started up initially you have to bring up levels to start initially automatic control may not be immediately applied. So, first we have to you know may be fill some chambers, so then first open the valve, then you start the pump then once. So, the levels start filling then you then you see when it is reached some sufficient level, then you switch off the pump then you switch off the valve. So, basically and then you start the heater, so let the temperature come up then may be the automatic control will start.

So, initially for these start up shut down sequences nobody I mean you may not be interested in doing an analog control, but rather the system is actually operated based on discrete controls. Similar things happen for shut down sequences, so once things have to be shut down, there are certain such you know some discrete control action have to be taken.

So, for such things typically sequence or logic controllers are used that is conveyors are very widely used material handling systems. So, in conveyor you do not bother about what is going to be what is going to be the motor speed maybe you won't bother about that. So, you only have to start the conveyor or stop the conveyor maybe when you detect a part on the conveyor you will you will start it. Then, after the part goes out from the other end, then you then you stop, so conveyors are typical examples where a lot of you know very simple PLC controls take place sequence control. Similarly, there could be automated assembly operations, so for example, you have a pick and place robots.

So, the robot you know it goes to certain states, so maybe it is situated somewhere and it is going to the picking up item x from the left hand side of the work space. Then, putting it on the machine then operating the machine then again picking it up then and then putting it on the bean which is on the right side of the works space. So, the overall sequence through which a robot goes or may be picks a pick an electronic component places it on the on the PCB. So, such operations are typically described and controlled by a sequence controllers, now we must understand that this is actually a robot when is for its movement, it has very sophisticated controls, it has server motors in its various joints, it has it has analog controls on them.

Here, we are we are thinking of designing the controls at a different level where we are not interested in that that speed control loop is actually in place and it is working, but at a lower level. So, only this sequencing tells us that what set point to give at what time, so it is a kind of you know set point sequencing if you if you look at it from analog control point of view. So, based on the sequence of the digital controller or the discrete event controller corresponding analog set points are being down loaded on the on the analog controllers and the analog controllers are actually taking place and controlling the motion of the robot and then taking them through the final states.

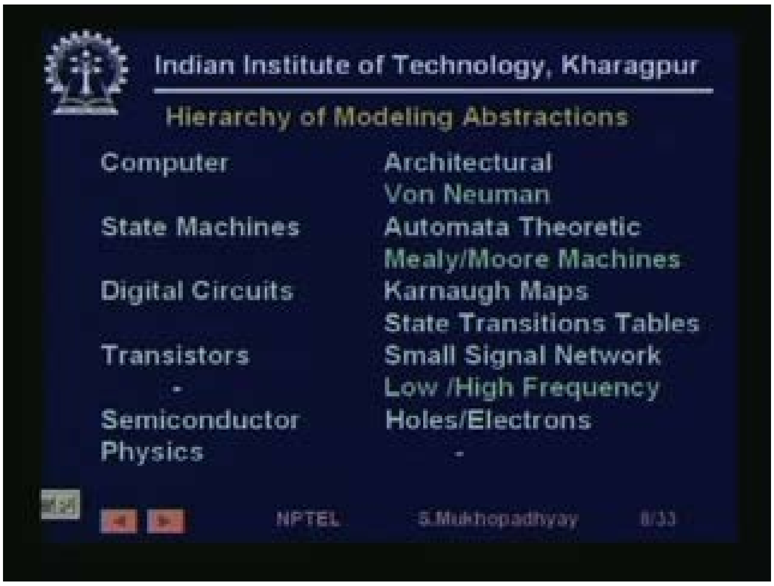
So, as far as the sequence control problem is concerned we are not interested in knowing or in describing or in solving how a robot arm is moving from one point x y to another point x 1 y 1 we are not interested in that. We are just saying we are just saying we are just making our mechanism by which the proper sequence of commands can be specified to the analog controllers, then the analog controller will be expected to take care of that motion.

So, in this sense it is actually a supervisory control problems, so sequence are logic problems are often applied at a supervisory level to provide set points to the analog controllers, which are working below it at an at an automatic controller. So, similarly sequence control in CNC machines where depending on the geometry of the part to be cut depending on the machine configurations. Sometimes, we have to we have to move the bed this way, sometime we have to move the bed that way or sometimes we have to extract this spindle up you may you may have to change the tool you may have to switch on some lubricant flow.

So, various kinds of you know again supervisory control actions must be specified, in fact in PLC there are CNC machine controls, then you can very clearly see that there are two different control systems usually on two different controlled processes. So, you have generally have a high speed microprocessor or a DSP sometimes generally microprocessors which take care of the analog controls, which are which are much faster actually control the various motors and you have a PLC. Generally, I whatever I mean, some make a PLC which we will give, which will basically control these various motions and may be also monitor the machine.

So, supervisory control of robots as have told sequence, that is why PLC are used hugely in this kind of decretive and controlled problems are very common in actually in industrial processes and I mean you will find I mean literally hundreds of them. Now, we must realize since we are I thought that this is kind of control problem and you know we are the same motor or the same robot, which we learnt from an analog control point of view. We are still using them, but now trying to construct completely different types of models for them and trying to solve different types of control problems. It is useful to understand that a model of a process is actually, you will construct for a process entirely depends on what it is, which aspect of the system behavior you actually want to capture.

(Refer Slide Time: 18:22)



So, it is not always. So, for example, let us say imagine that if you if you are trying to model a computer then there are various architectural models for a computer. For example there may be a Von Neumann architecture there may be a reduced instruction set architecture there are several you know well know architecture of computers I mean models around which computer are constructed. Now, you know computers are typically you know very large a digital circuits I mean sequential digital circuits which are sometimes thought of as a as you know state machines. So, state machines are again have there are there are there are there are various kinds of state machines.

So, if you are seeing the computer at the state machine level, then you are using models like you know Mealy machine Moore machine basically automata theoretic points. So, again this state machines are made of what there are actually made of digital circuits, so digital circuits how do we describe digital circuits? We know that we described combinational digital circuits using Karnaugh maps or you know some of product Boolean formula or if there is sequentially circuits, then we describe them as state transition tables. So, now what are digital circuits made of they are made of transistors now transistors? Now, transistors are actually the movement you come to a transistor there are there you have come down to analog there, so there you start describing voltages currents and their time behavior.

So, you have various kinds of you know small signal network models either low frequency models or high frequency we have we have learnt several kinds of model, again transistors are made up of semiconductors. So, semiconductors have you know physical models you know electrons holes energy bands. So, you see that, but you obviously, will not describe a state machine by a state of transistors. So, there is a whole hierarchy of you know abstractions, so as you go up and up you consider more and more things. You consider them in more and more abstracted ways you know, but sometimes it happens that the same device is being look at as something by 1 percent and by another thing as another point.

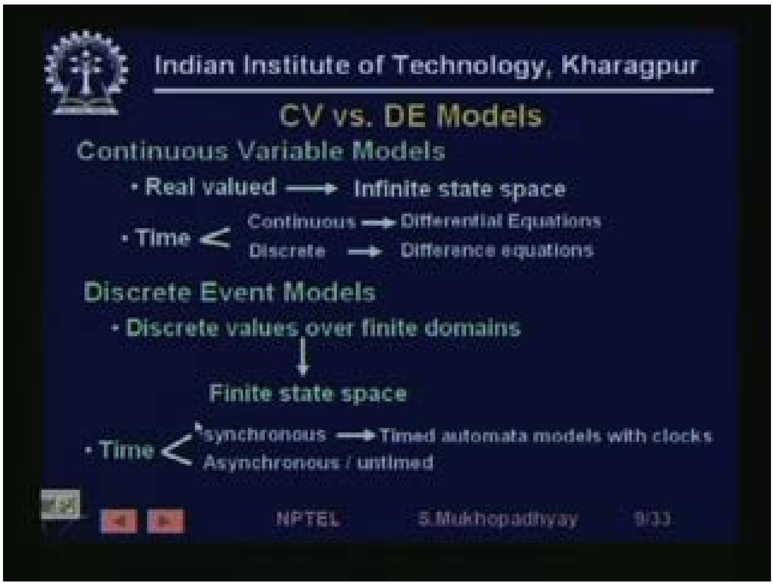
For example, say if you are a if you are a if you are the user of a digital circuit, then you will use a model like you know I mean either of a NAND gate or AND gate or flip flop. So, you use one kind of models, but imagine the semiconductor imagine the gate designer I mean the VLSI designer of the gate. So, why you construct why you look at a gate not as transistors, but as a digital circuit because you know that the circuit is such that for most of the time just like the switch example I gave you, it will either stay in a 5 volt state or a 1 volt state. So, it is either 0 or 1, so it suffices to describe when it will go to 0 and when it go to 1, it is not important at all to describe the transitions of how exactly the voltage signal will rise from 0 to 1 that is not important at all.

On the other hand, the person who the look at the VLSI designer, who wants to design the gate. So, you know the gate circuit is such that it will automatically go from 0 to 1 to 1 or 1 to 1. The beauty of design or the or the whole credit of design goes into understanding that how exactly how fast you can make it rise how much less power can be dissipated and things like that. So, on the other hand the circuit designer is only concern mostly concern about the transition part from 0 to 1. So, he obviously, cannot take a view of the same gate as that of an as that of a NAND gate model. He as to take a view of the same device as a non-linear transistor circuit because his attention is focused only on the transition or on the analog behavior.

So, it is important to understand that, therefore this is what I am saying that even for industrial processes there is obviously, analog behavior everything is the analog. If we know that the analog behavior can be taken care of by the analog controls which existed at lower level.

Then the problem can be simplified if we assume the existence of suitable analog controllers then the problem can be simplified to that only specifying certain discrete commands and how those commands will be realized is taken care of by something else. So, we are not interested in modeling there, therefore we will simplify the model and use different model and then solve sequence control problem that is that that is a whole idea, so it is very important to understand that.

(Refer Slide Time: 23:27)



If we look at continuous variable models, there this signals are real valued they have they have the actually you have an infinite number of possibilities of values and time is also time is sometimes continuous and times discrete. So, when we have continuous time models, then we have time continuous varying, but for all practical implementations especially using computers we must discretize time that is we must consider values only at certain instance of time. Now, you must realize that we are discretizing the time axis we are not discretizing the variable value variable value is axis is still continuous variable value can be anything, so but the time axis is sometimes discretized.

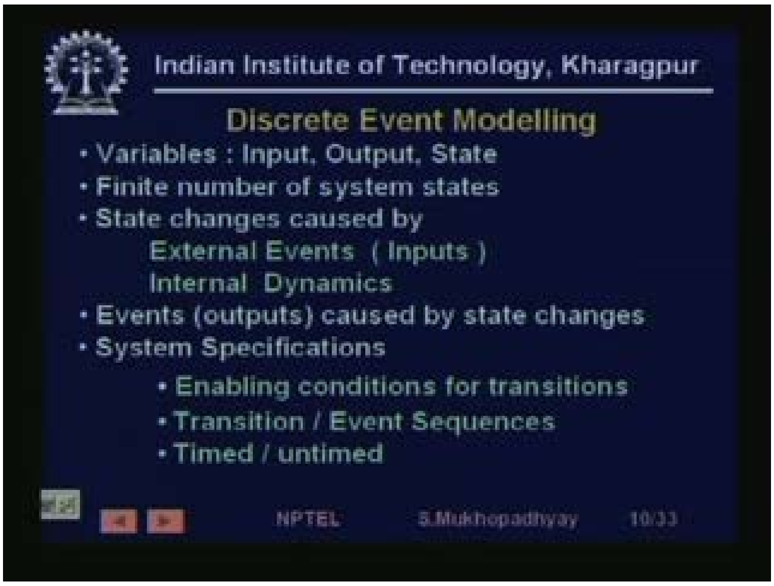
So, that is why it is called a continuous variable models where variables can take any value contrary to this in discrete event models which are which we are going for sequential logic controls problems. Typically the values of variables that we will consider will be will be considering finite. So, let us say I mean typical example is on off you know some tank level low medium high.

So, we are not interested in knowing what exactly how many meters we are only considering we are only thinking that the tank is in the tank level could take one of three symbolic values. So, each of those symbol is actually represent some kind of an analog level value range. So, if it is 0 to 2 meters, we just call it low we indicate it by a symbol low if it is 2 to 4 meters, we call it medium and if it is 4 to 6 meters we call it high. So, we may define a control problem where we are not interested in knowing the value, but we are only interested to see that the movement the water level falls from low to from high to medium start them.

So, we are interested is you know specifying such things, so a discrete values over finite domains and therefore, we have we have we have what is known as finite state space. So, the combinations of variables that can exist in a system are going to finite, similarly you know we have two kinds of models this we are not going to see too much, but there are two models. It is called synchronous or an asynchronous where you know that we assume that all variable changes also occur at in along with some cloths, you know just like a just like a master slave flip flop is synchronous device.

So, sometimes we have asynchronous, so at anytime certain variables can change, so it is like you know time discretization here, so if the time is if variables can change from one value to another value at any point of time we call it asynchronous. If it only happens at certain periods of clock ticks, we call it synchronous, so once we quantize, once we create a discrete, you know this kind of you know discrete finite state description of the system. Then, it is easy to construct the then we can construct the behavior of this system.

(Refer Slide Time: 27:00)



So, typically we have in such systems three kinds of variables they are the input variables input variables are something which are applied externally. There are output variables and then there are state variables, so output variables are something you know which we which we can say which we can may be sense or see and there may be some other variables which we cannot see, but the, but the system goes through them. So, typically we consider systems and in most of the industrial control problems, we have we have only a finite number of systems. Now, this state's what is behavior is the is the pattern by which states change from I mean system continuously moves evolves from one state to another.

Now, why the state change should could change either by an external event, so your room was in dark state, now you somebody comes and gives an input or change is an input variable say a switch from on to off to on. So, as the consequence of this the system the room state goes from dark to lighten, so this is state change caused by an external events. Sometimes system also go by because of their internal dynamics without any other external input given it can it might it might change the state. So, there may be some arrangements in the system by which suppose after it is made on it, let us look at a let us look at a mono short timer.

You know we has common electric examples, so there may be something that once you press it some bulb glows up and then its dynamics is such that that again after some time by itself without any input being applied it will come down.

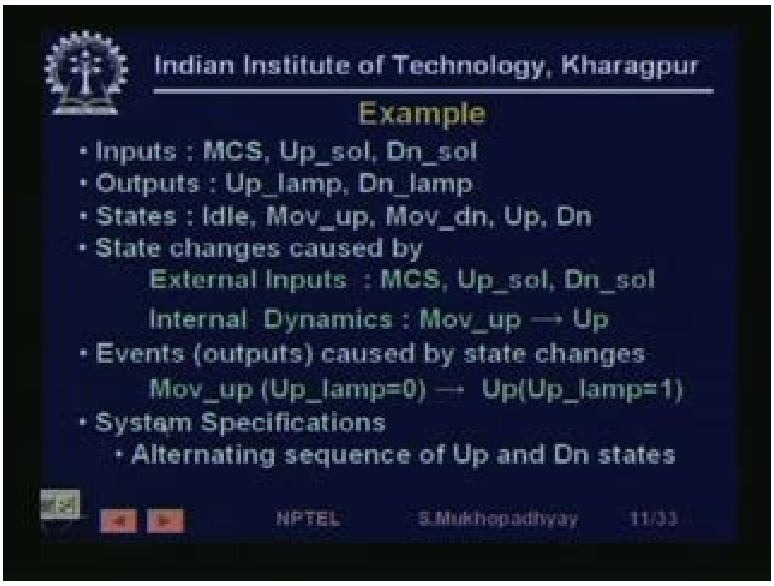
So, state changes are sometimes caused by external events and sometimes caused because of you know their own internal dynamic state they might come down. Similarly, state changes also at every state there are there are some outputs, which are exerted by the system. So, as states change as a system goes from one state to another the outputs are also change and these output changes are perceived in the external world and they are sometimes called events. So, you could have events or outputs caused by state changes, so this is when we have to describe a system we will presently see an example we have.
Basically whenever we want to model a system for the purpose of logic or sequence control. We have to find out to which are variables, which are the input variable which are the output variables, how those change what input change what input exertion cause what state change.

So, basically if we can describe these then we have got the system model, now in any control problem there is just like in in analog control you have a open loop plan and you have a closed loop plan. So, you have a planned by itself which is called uncontrolled or open loop and you also have a closed loop plan which is controlled. So, you know the controlled plan is actually has does not exhibit the all possible behavior of the open loop plan, but exist, but exhibits only a certain subsets.

So, that is the job they control we have to when we are taking about the controlling design problem. Then, we there are two things that we have to do the first thing is that we have to say what this system does by itself what it is capable of doing what it is capable of doing that we can get from its model. Then, we have to give a specification that is of all these behavior we do not want sudden some behavior. So, we want only some of them, so what is the behavior that we want for the controlled system that is the called the specification.

So, these specifications could be again stated in various ways for example, by transient transition event sequences which could be either timed or untimed. You know saying that giving the enabling condition for some transitions that is we may like to cause some transitions in in certain ways in the system. So, in various ways system specification will be given and then the job of the controller design is to ensure that this system actually behaves like the specification system. That is the problem of discrete event on logic or sequence controller, so seems a bit abstract perhaps, so let us see an example.

(Refer Slide Time: 32:00)



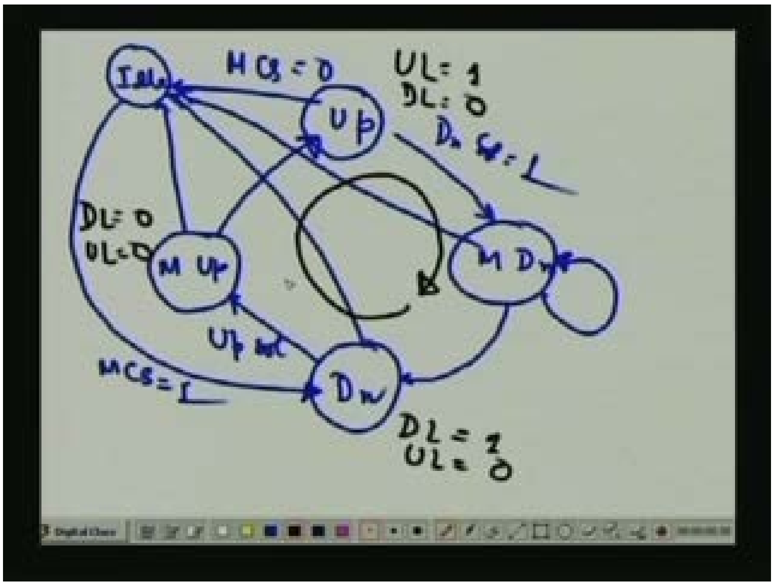
So, in our old example what are the external inputs, so depends on what you have you have I mean you actually have to see. So, may be that you have in this system we may assume that it has three external inputs one is the up solenoid another is a down solenoid and the third one is the master control switch. So, these are three input variables there are all external to the system and each one of them for example, the up solenoid and the down solenoid may be a may take two values either 0 or 1. So, what could be outputs for example, we could say that corresponding to limit switches we have some lamps. So, whenever our limit switch is made we have an up lamp or and a and a down lamp, so they are like indicators.

So, corresponding to the limit switches we are having some outputs up lamp and down lamp. Similarly, the machine we assume is existing in three kind of states you know basically they are the motion states of the system. So, the system could be either be in the idle mode in which it is doing nothing it is not in the active mode or among. So, we can say that this this where whenever the master control switch is 1, what is the effect of this external input. That will make the system from idle to any of this and if it is off then from these any one of this it will actually come to the idle state. This is what we are trying to describe the behavior of the system, so otherwise among the active states the system could be either in move up moving up.

So, it is moving up it is neither up nor down it is it is the position of the press is neither up nor down neither down nor up it is somewhere in the middle, but it is moving up.

Similarly, it could be moving down or it could be not moving, but is in the up position or in the down position. So, we have chosen to model the system in such a manner, so we assume that the system goes through such states. Now, now state changes what are the typical changes in the system, for example we say that the external inputs are MCS up solenoid and down solenoid. So, if MCS is in exercise system comes from idle to any one of these may be it come from idle to down and if the up solenoid is pressed and if the system is the is in the down state from there is the up solenoid state input is exerted. Then, it will go from the down state to the moving up state, so these are cases where a state change is caused by input we will we will we will present a draw a diagram, similarly internal diagram for example, let us draw this diagram.

(Refer Slide Time: 35:40)



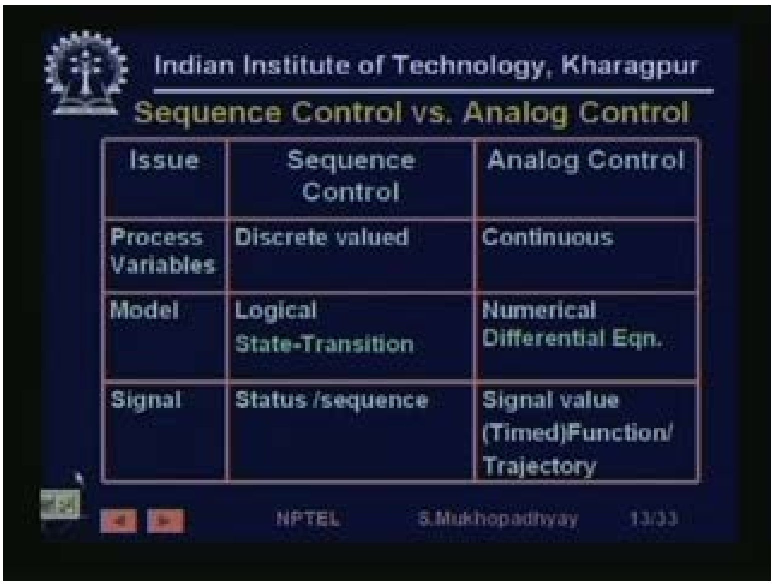
So, this system we are saying exists in five states, so maybe this is up this is down this is moving up and this is moving down and somewhere you have idle, so how does the system change states? So, you have from idle to down when MCS equal to 1 you are tend to describe the system behavior and then from down if you give a give an up solenoid. Then, it goes to the moving up state from there by internal dynamics after sometime it will go to the up state. In the up state, if you give it a down solenoid equal to 1 input, then it will moving down state and then by itself it will come to the down state in between sometimes you know self-loops. You can you can write that in between if moving down is in this position, it will be in the moving down state.

It might take quite some time to for it to come down from up to down state, so then we can say that anywhere if you go to moving up then it goes to I and if MCS becomes 0 at any point of time, then it is goes to the idle side. So, you know that we are we are basically graphically we are we are describing the how the system goes from one state to another because of various external inputs or because of internal dynamics. Similarly, we can say that for example, we can have we can have the various outputs, so if we have we have the two outputs. So, here we will have down lamp is equal to one and up lamp is equal to 0, similarly here we have DL equal to zero and UL equal to 0, both are 0 here UL to equal 1 and DL equal to 0.

So, these are the output, so at the various states these are the outputs which are actually exercised and which may be also which may also be sensed by the sensor for the purpose of the controller. Now, this is the behavior of the intrinsic plant, so now, one has to ensure one as to ensure somehow that the specification that you give for the system that is actually first up solenoid is exercised. It is held till it goes to the up state and then when it goes to the up state then then the down solenoid is exercise. So, this actually this is the this is this is the desirable cycle of the of the machine, so one has to exercise some device as to actually compute this when to apply this up solenoid and when to apply this down solenoid and it is the controller which is going to do that.

So, as we say that events are caused by state changes because this we have already for example, in moving up lamp is equal to 0, when there is a state change from moving up to up lamp becomes 1. So, state changes also cause outputs and assistant specification as we said that is the objective of the machine. The objective of the machine is that when the control switch is on at that time it must come down go up come down go up this it must do a certain number of times till the MCS switch is put off. If the MCS switch is put off immediately it from there it must go the idle state which we can describe what is the position of the place at that time whatever it is. So, this this system specification for this system could be an alternating sequence of up and down states, so it must continuously go from up to down and down to up states.

(Refer Slide Time: 40:41)



So, having done this example we can take a look at some basic differences between sequence control and analog. So. Firstly, the process variables in the sequence control are discrete valued in analog they particular valued. The model as a consequence the model here is logical typically you know state transition kind of models while in analog control they are numerical. So, you use either differential equation or you use difference equation, so use some equation numerical equations. Similarly, the signals generally the signals indicate some status, so something is on off may be a traffic light is either red or amber or green or we can we are interested in or they are sometimes you know signals are in this case sequence of symbols.

So, it goes from up down up down up down up down or may be a may be a conveyor for a buffer another part another part another part buffer empty another part another part. So, this these are you know sequences, so another part is an event occurring whenever another part comes into the buffer and in between certain other event call buffer empty take places. So, the signals by signals we mean this sequence of this events in the case of sequence control. On the other hand, we know that in the case of analog controls we want to specify actual trajectories we want to specify properties of trajectories rise time should be less than something, maximum overshoot should be less than something. So, what are trying to say we are trying to say that we are trying to describe some properties of continues trajectories in terms of their values, while in the sequence control, we are interested in either status of some signals and there their sequences overtime.

(Refer Slide Time: 42:59)

Similarly, if you look at control here also can be open loop or closed loop closed loop control is possible even for even for a discrete event or logical control for example, in our in our dye press example we had those two limits switches. So, any controller that will work can take feedback of this state of the process using the limit switches and then decide the controlling.

So, in that sense logic control can also be feedback control right, but it is generally on off or logical kind of control and they are generally supervisory in nature that is there. They are used to decide in many cases they are used to decide the command sequences, which are again realized in term in turn by the automatic controller.

While the in the automatic controller we could have linear or we could have non-linear controllers and I mean which are which are which are actual working at an automatic control level. One very interest design is of course you know since you have so much simplified the states space or the values that that I mean signals can take into 2, 3 or 4 n codes, I mean codes. So, if the models are much simpler say for example, very linear and non-linear differential equations can then finite finally, give rise to rise to some you know 5 6 7 8 or may be 20 state finite state machine which is a much simpler model compared to the compared to the nonlinear differential equation. So, generally design is order of magnitude simpler than continuous automatic control designs.

One interesting thing is that tuning is generally not needed because the models are generally not physical there are actually you know abstract you know in information oriented models. So, unless you change the strategy of operating the plant it, I mean the controller need not be tuned while in the case of automatic control. You know systems system heat transfer coefficient will change pipes will develop slugs inside I mean valve characteristics will change reactor characteristics will change.

So, therefore, automatic controllers continuously required tuning from time to time, but these sequence control controllers they generally do not require any training any tuning frequently because processers. I mean till you do not have a different automatic control system, which will take different input or you have want to change the whole operating mode of the process, there is no tuning of the sequential logic control law to change.

(Refer Slide Time: 45:44)

So, now we have we have understood the discrete event control problem, now this is not new it is to be this kind of problems have been handled in the industry long time even before the even before the microprocessor came into existence. So, previously what people is to do is that people is to use relays and contractors to actually realize this kind of logical functions you know. So, that is why previously such logical functions is to be called relay logic, in fact the name relay logic diagram relay logic ladder. These are basically legacies of that past which are still being still being used of course there are gradually getting change and probably after 5 10 years relay ladder logic may not be. So, relevant people will find other graphical and other ways of describing this programs.

It is a fact that PLC is have actually say as they started to replace the real logic and so why they why they are better than relay logic that we can see. For example relay logic is hardwired actually things are things are put relays contractor there are actually connected, while PLCs the logic is actually in a program. So, while if you want to change the logic little bit then you have to dismantle real logic say at least some parts and then again install a new one. In the case of PLCs, it is a microprocessor device all it takes is to change the program, therefore they these are you know relay logic is actually difficult to upgrade and maintain while PLCs are very modular.

You can just take out one and you can put it another put another, you can all the time put you know you can you can expand your system very easily. You can put by new input output modules, you can then simple plug in plug them into the racks without any problems. So, these are much more industry friendly to upgrade and maintain real logic is obviously, much more limited in terms of size and complexity because you have to construct them using physical relays. In while PLCs, you can you can you can latterly put I mean thousands and thousands of such logical functions mainly in mainly in computer memories.

So, there in terms of speed size complexity and even reliability there they are beating there relay counter. They have already beaten no question of beating there they have beaten their relay counter parts hands down. So, very old technology actually nowadays hardly anybody uses any, in fact all the time anywhere you go to any factor you will find tones of PLCs all around.

(Refer Slide Time: 49:03)

So, what is a PLC what does it do we have understood that it a actually you know I look at the Internet and try to locate some definitions of the PLC and these are some of definitions which I found on the internet, but somehow I did not like them. So, somebody says it is a solid state device design to perform logic functions solid state device the term is actual not very commutative because it again because it describe everything is in solid state device even a small transistor is also a solid state device. So, solid state device design to perform logical functions what previously accomplished by electro mechanical relays this is something which mentions the history.

Devices used for control operation and integration of manufacturing processes, yes this is I like it more of manufacturing process equipment and machinery, but it still it does not say what kind of device. It is very cryptic on the device side an assembly of digital logic elements this word element is very confusion what is meant by digital logic elements design to make logical decisions. So, you know design to make logical decisions and provide outputs, so I thought that I will find out my own engine own definitions.

(Refer Slide Time: 50:31)

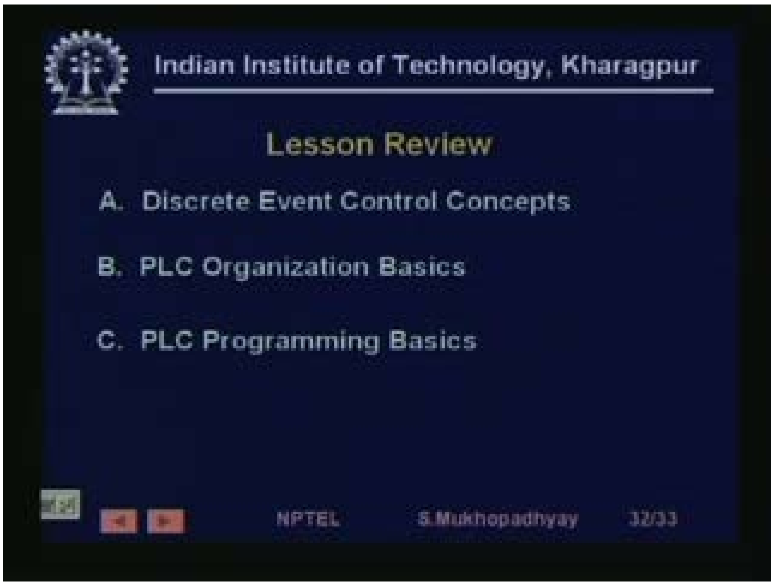
So, I am calling it an industrial computer it is actual a computer which is used for industrial function. What kind of functions, which will accept from digital or analog sensors which will execute logic for sequence control or analog control. In fact, modern PLCs is in in many cases also contain capabilities for doing analog controls on the same device after all it is microprocessor. So, you can have several it its speed is very high it has lot of memory. So, why not also once since you are buying a device why not put the analog controls also. Otherwise, you will have to put buy another device just for an analog controls, so while logically they are different physically speaking one single PLC you know box can house even analog controls in facts they do. They could drive actuators or indicators or they could communicate with other computers.

(Refer Slide Time: 51:31)

So, what are its components its components are typical of computers, so you can have you know the sort of you know the back plane or you can have a power supply or you can have CPU memory or you have IO cards various kinds of IO cards are used. So, you can you can digital or analog modules, you can have some special purpose function modules like you know high speed counters which measure you know shaft angle encoder which measure rotational speeds. Certain very fast and precise positioning commands can be generation modules PWM generation modules etcetera. So, various kinds of cards it uses, now obviously, there is wiring and then you need two other separate devices with PLC mainly for you know for interacting with it.

One of them is a programmer using which you can download the PLC or you can see how variables are getting changed and the other is a man machine interface which can displace on the variables nicely. So, that people can see how things are being controlled, so actually what I we will make a I have little change of plan in this thing we since time is nearly up. So, I would skip the programming in this lesser and then we will include it in the next lessons. So, I am going to skip the next few slides and then come to the come directly to the lessons review. So, we are going to see all these things in the next class which I thought I will include here, but there is no time.

(Refer Slide Time: 53:32)

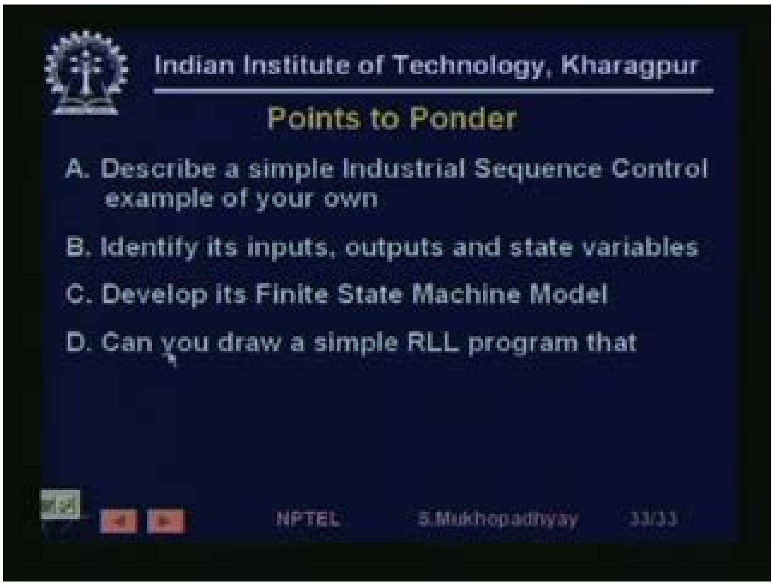


So, we have the lessons review, so what have we seen in this lecture we are primarily tried to impress upon the nature of discrete event controlled problems. You know there is basically two thing that there are some problems where analog controls are not required because they are too simple. If you start a motor for a for a conveyor, you do not need to control its speed just put full voltage whatever speed it will rotate, let it rotate on will move. We are not interested in controlling the speed in a analog manner these are situations where analog logic control is used. There are another kind of situations where you a time to design the control at a supervisory level.

So, there is a whole set of sophisticated analog controllers existing and the moment you give them command you know that they will they will realize them. How they will realize is not your concern as long as you are designing at the supervising level. So, because you are designing at the supervisory level you controlled problem is abstracted out. So, in that sense it is it is discretion, so we have seen that many such problems arise in the context of industry and we are going to in this module we are going to look take a look at how we are we will handle them.

So, we also define first introduce the programmable controller we saw its we gave its functional definition and we saw what it contains. It is basically a microprocessor based device with a lot of you know interfacing capabilities, and this PLC programming basic we actual did not consider in this lecture and we will be considering in the next one.

(Refer Slide Time: 55:19)

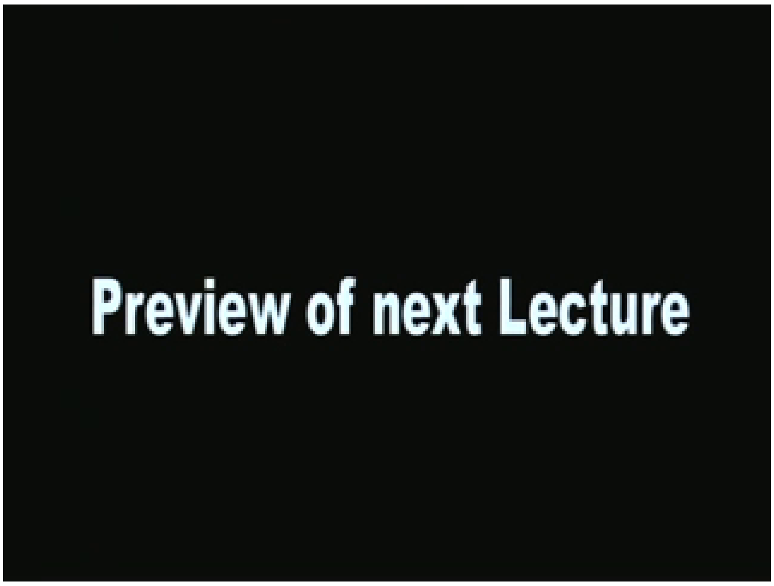


So, you have you could do several things for example, you could find a typical industrial example take a take a take a let us see for example, you can have see things like I will give you one example myself and you can think of other. For example think of how to automate the overhead tank in your house using may be some level sensors and a motor and a pump such that the water level never goes below certain level. So, you are you are not you are never surprised in the kitchen, so try to model the system identify its input output and state variables develop its finite state machine model.

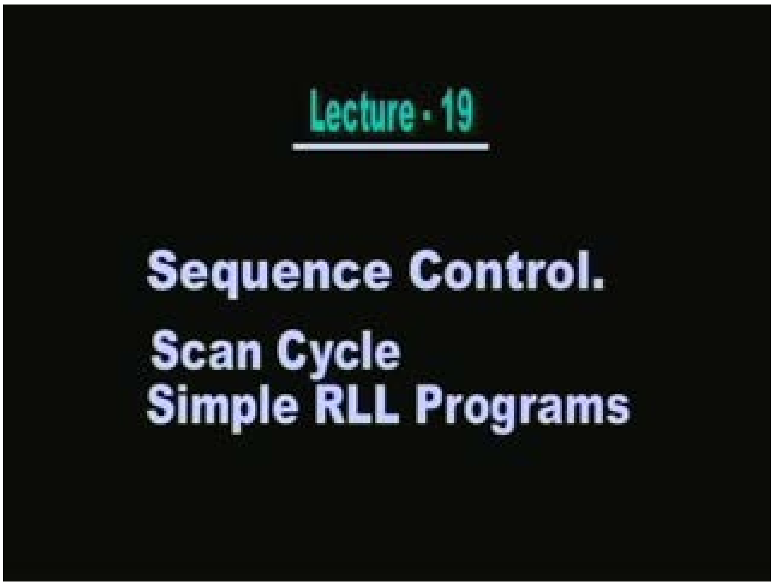
Then, finally can you draw you can do this now, but you can you can, but you can perhaps figure out that what when to how based on the feedback you are going to apply the different kinds of controls. So, construct an example of your own that is the assignment and that is all for today, thank you very much, from the next class we will study on we will start with the programming aspects of PLCs.

Thank you.

(Refer Slide Time: 56:38)

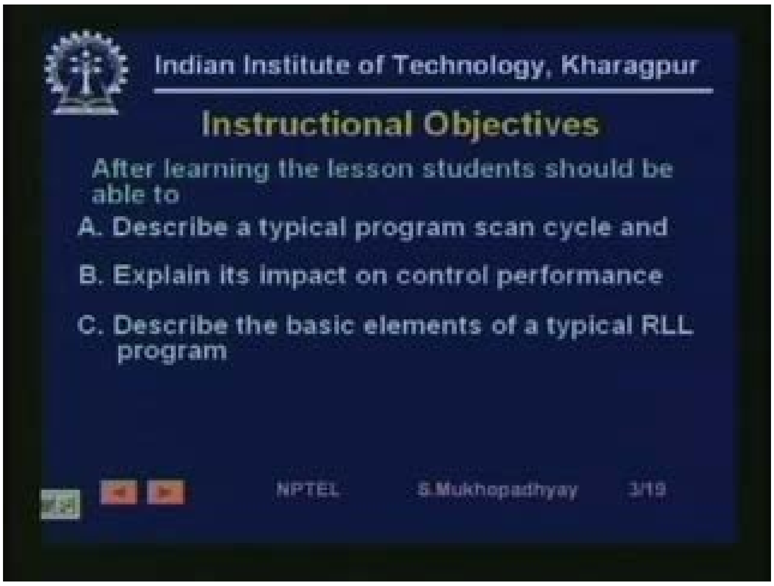


(Refer Slide Time: 56:43)



Good after noon, and welcome to lesson number nineteen where we shall discuss, so continue to discuss sequence control. Today we will discuss a scan cycle that is scan is a program execution. So, we will discuss how programs are cyclically executed and we will discuss the nature of its impact on performance and we will also discuss how to write some simple relay ladder logic programs.

(Refer Slide Time: 57:32)



So, as usual we view the instructional objectives first, after today's lessons one should be able to describe a typical program scan cycle that is describe what takes place how the program works cyclically would be able to explain the impact of this kind of execution on the control performance typically on the fastness of response. That is how fast the programmable logic controller response to change is inputs with change is output. Then, we will go on to describe the basic elements of a typical relay ladder logic program, today we will look at the very simple elements of the program and in future lectures we shall go on to see more advanced programming elements.