INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

NPTEL
ONLINE CERTIFICATION COURSE

On Industrial Automation and Control

By Prof. S. Mukhopadhyay
Department of Electrical Engineering
IIT Kharagpur

Topic Lecture – 24
Sequence Control. Scan Cycle. Simple RLL Programs

Good afternoon and welcome to lesson number 19 where we shall discuss.

(Refer Slide Time: 00:29)

Indian Institute of Technology, Kharagpur

Lesson 19

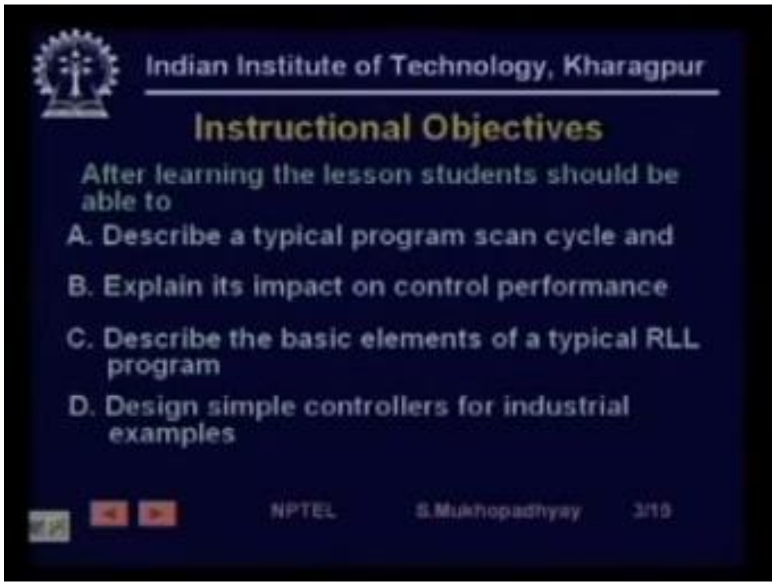
Sequence Control .

Scan Cycle
Simple RLL Programs

NPTEL S.Mukhopadhyay 2/19

We shall continue to discuss, we shall continue to discuss sequence control. Today we will discuss a scan cycle that is a 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: 01:02)



Indian Institute of Technology, Kharagpur

Instructional Objectives

After learning the lesson students should be able to

A. Describe a typical program scan cycle and

B. Explain its impact on control performance

C. Describe the basic elements of a typical RLL program

D. Design simple controllers for industrial examples

NPTEL S.Mukhopadhyay 3/15

So as usual we view the instructional objectives first after today's lesson one should be able to describe a typical program scan cycle that is described 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 responds to changes in inputs with changes in 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. And finally we will take two of the simplest examples and see how exactly a relay ladder logic program segment could look like and also would learn to interpret a particular RLL program that is after we, if you are given an RLL program then we have to know what it means and how the inputs and outputs will change with time so we will do that. So that is the agenda today.

(Refer Slide Time: 02:33)

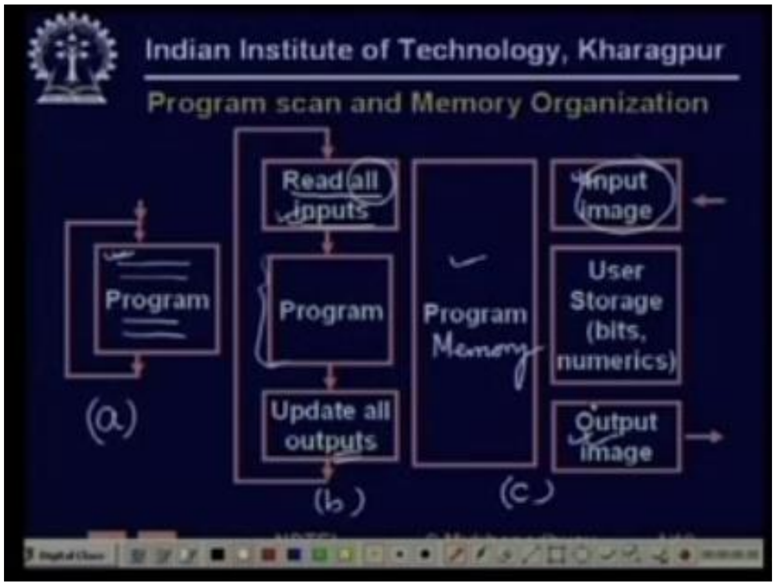
Indian Institute of Technology, Kharagpur

Program scan and Memory Organization

NPTEL S.Mukhopadhyay 4/19

Having said that we continuing on the same note from the last lesson if you recall in the last lesson we had end at a point when we had listed some of the hardware elements of a PLC system namely backplane, CPU, memory, the different IO modules, wiring, programmer, man-machine interface, etc. So today we take a software focus and start looking at how the basic program execution goes in a PLC. So this one obviously in like in most control programs a PLC program also works cyclically.

(Refer Slide Time: 03:33)


Indian Institute of Technology, Kharagpur
Program scan and Memory Organization
Program
(a)
Read all inputs
Program
Update all outputs
(b)
Program Memory
(c)
Input image
User Storage (bits, numerics)
Output image

So as it happens so basically these are actually so this is one view say, view A, in view A the we are indicating that the program simply work cyclically. So this program which is a set of statements execute cyclically that is the execution begins here, then next statement and so on till the last statement and then again the first statement will be executed. Now what happens within this overall program box, that is what are the functions which are carried out in the program.

So as we all know that PLCs are real-time embedded systems they are they are reactive in the sense that they accept inputs from the external world and produce outputs which go to the external world namely the machines. So obviously the program has to be always aware of what changes are taking place in the external world namely the various variables that are being controlled in the machines.

So a typical program cycle includes three steps first is so this is a an expanded view of a namely B which shows that typically three activities take place during a program execution, so for the first one is read all inputs mark the stress on all which means that it is not that so initially it means that initially all the inputs are read one after the other and their values are stored and then the execution of the logic begins so this is a major difference with a with any standard program which you could write in a microprocessor in a microprocessor you could write a you could write a program where the various processing statements addition, subtraction, multiplication, comparison whatever you have.

You could intersperse them with the input and the output statements but typically in a in PLC execution that does not take place so in PLC because presumably because of the fact that initially the PLC programs you know there is a good there is always a stress on keeping things simple so that errors do not occur, so at the cost of some flexibility of programming we are intending to achieve some simplicity and that is the sound strategy provided performance is not affected and in this case it happens that you know typical micro processors statement execution times run in microseconds.

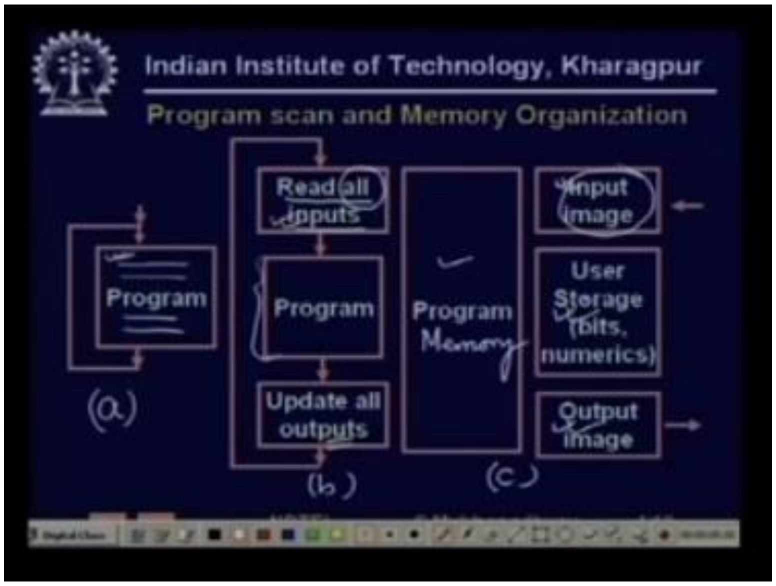
While with PLCs the kind of things that we are trying to control are either mechanical or thermal or chemical, so the time constants are usually so large that this shuttle difference between reading all the inputs at one time makes hardly any effect so therefore typically for PLCs all inputs are read at one shot in the beginning at the beginning of this can cycle one cycle of these three steps would be called a scan cycle after all inputs are read the program logic gets executed.

Now during the execution of the program logic various outputs are produced so it so happens that as outputs are produced they are written to some temporary memory but just like inputs are read at the same time similarly it is not it is not that whenever some outputs are produced and some output values are stored in memory they would be going to the external world that does not happen so after all the logic of the program is executed at the end all the outputs are output values which go to the external world in the sense of you know various switches motor starters lamps indications they are all updated at the end of the cycle and then the cycle repeats namely the next cycle of inputs are read.

So this is the this is the typical way that a PLC programs can takes place and so c is a kind of shows the kind of memories that are used in the PLC system so you have four kinds of memories first you have what is known as program memory where the logic program is stored that is the first one then you have what is known as an input image, so when the inputs are read they are for the program logic evaluation these input values will have to be read from time to time during the evaluation of the logic.

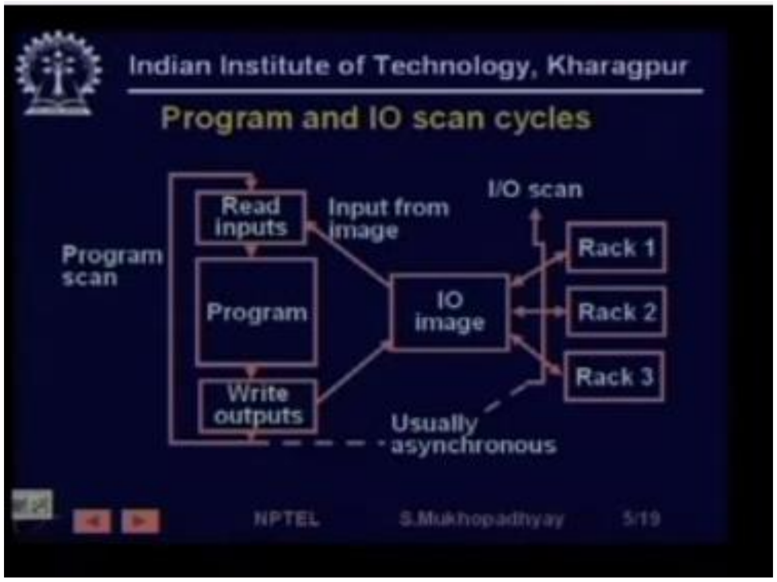
So therefore they have to be necessarily stored in memory so where the input values are stored is called an input image so this is the second use of the memory similarly as outputs are produced as the logic gets computed outputs are produced and they have to be stored in the time that they can be transferred to the external world so for that the output image memory is used and as we know that for all computations we need some we need certain in addition to the input and output variables we need certain temporary variables which are, you know so for storing these things we need.

(Refer Slide Time: 10:05)


Indian Institute of Technology, Kharagpur
Program scan and Memory Organization
Program
(a)
Read all inputs
Program
Update all outputs
(b)
Program Memory
(c)
Input image
User Storage (bits, numerics)
Output image

Some other additional memory so this shows how the overall memory available in a PLC system is used right. Having understood that we have to understand that this kind of a program execution.

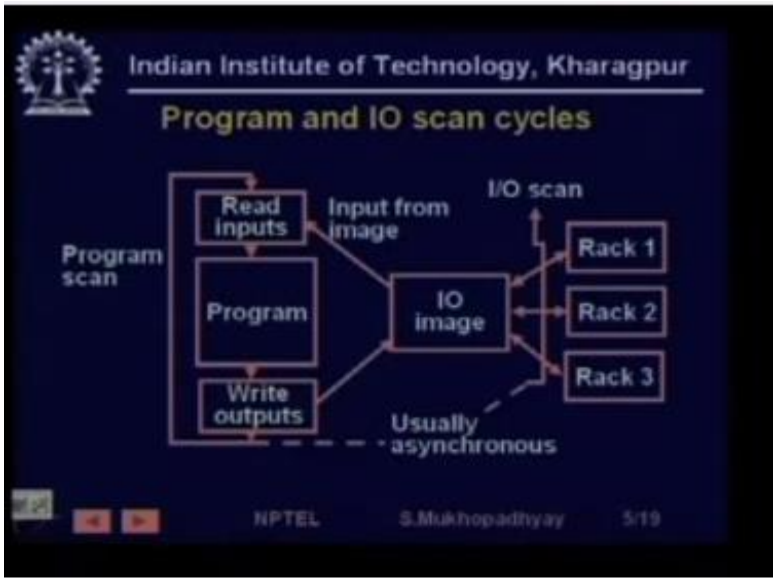
(Refer Slide Time: 10:29)


Indian Institute of Technology, Kharagpur

Program and IO scan cycles

We are in this in this figure we are what we are trying to show is that many times it will happen that it will happen that the if you let us first look at the if you understand if you have to understand this diagram.

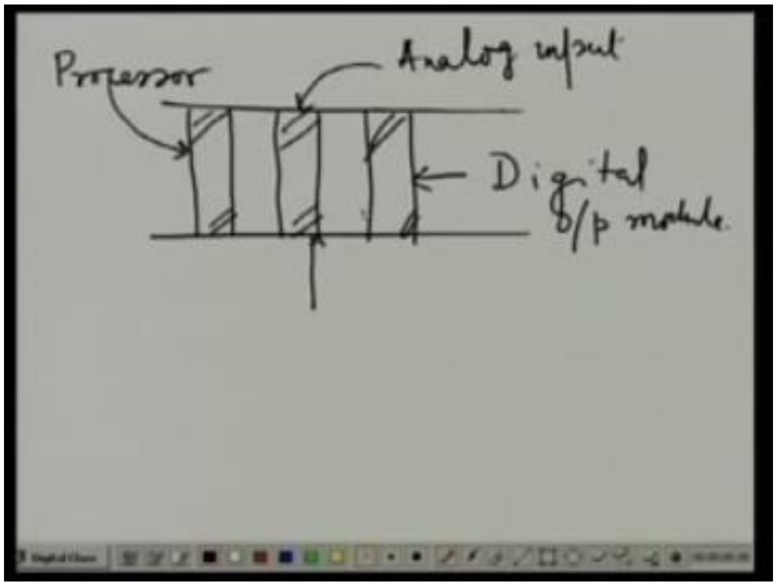
(Refer Slide Time: 10:51)



Indian Institute of Technology, Kharagpur

Program and IO scan cycles

(Refer Slide Time: 10:54)

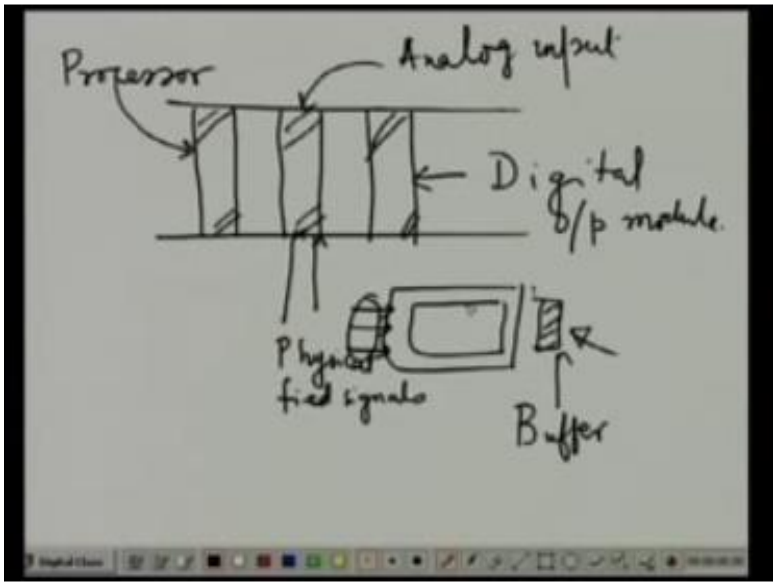


Before that we have to understand a typical you know PLC hardware organization so what happens is that in a PLC the various units this we did not discuss in the earlier lesson so the various units are actually organized as modules so you have what is known as you know racks so this is one module similarly there could be another module by the side of it so this is one module this is another module.

Similarly so this could be a this could be the main processor module, similarly this could be an analog i/o module input module that is, similarly you can have digital input digital out various kinds of modules so what happens often is that so now you see this processor the external variables that is the various signals get interface to these modules and similarly this could be a digital output module.

So what happens is that these external signals are these modules are again self independent there are such independent circuit boards, so they often have processors on them.

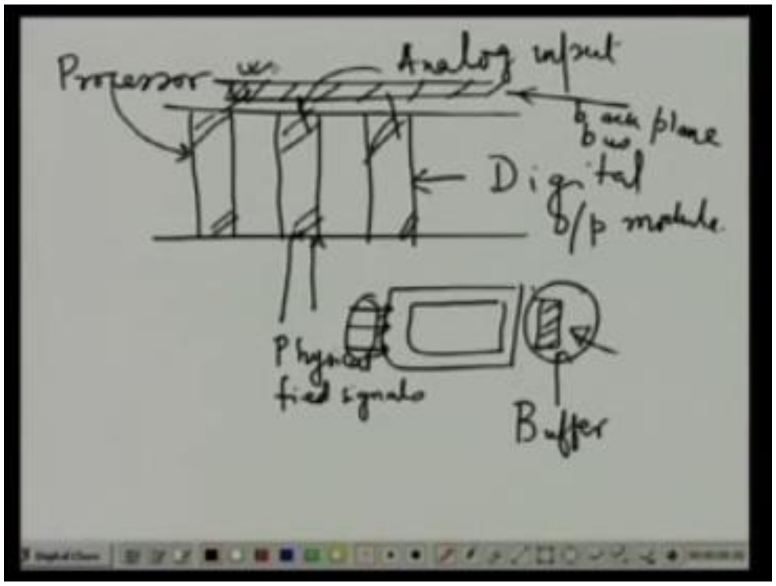
(Refer Slide Time: 12:35)



So what happens is that the processor here the processor here actually what happens is that you know here is a board suppose I we are taking so you have all the circuitry here in this side the physical signals are interfaced physical or field signals and on this side the this so these are physical signals and on this side symbolically maybe in a buffer this is a buffer the values of these signals are stored.

So when the processor module reads an input it is not that it reads these physical signals it actually reads these buffer values which are stored which are continuously kept in the buffers by the i/o cards, right. So and this I/O cards often work under the control of the processor it may work at the control of the processor or it may continuously or it may work independently cyclically itself, right.

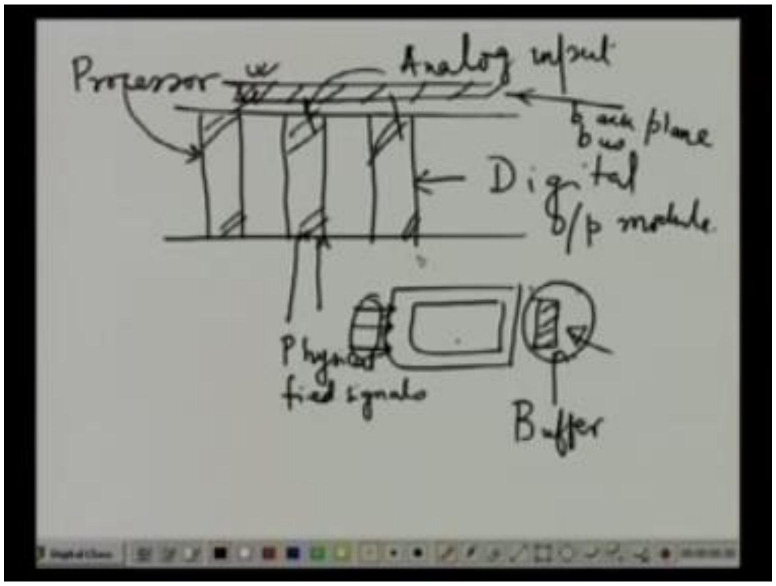
(Refer Slide Time: 13:56)



So all these i/o cards are actually there is a there is what is known as that is what we refer to in here in our earlier lecture as a back plane or a bus, back plane or bus which is nothing but a set of conductors to which all of these are connected, so when I say that up that a that a processor reads an input it actually reads these buffer values over the bus, so actually two activities keep on taking place.

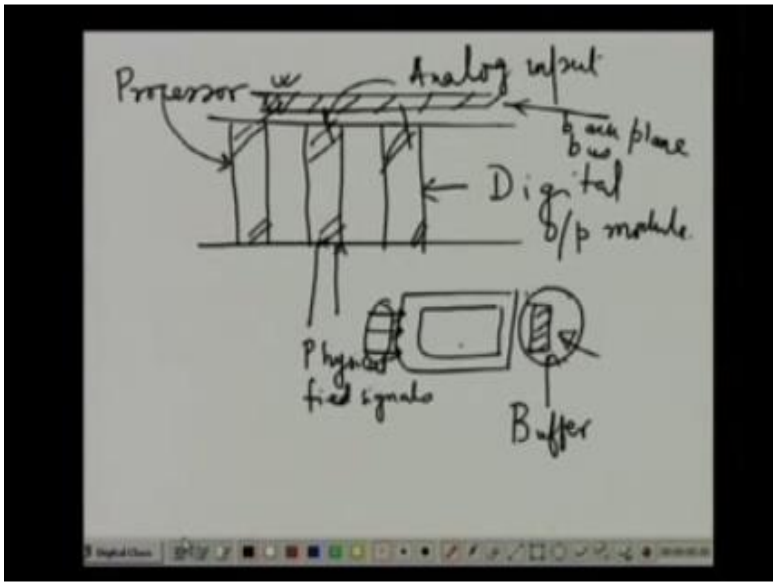
The first one is that the physical signals keep getting transformed from either to these buffers and this these buffers could be.

(Refer Slide Time: 14:46)



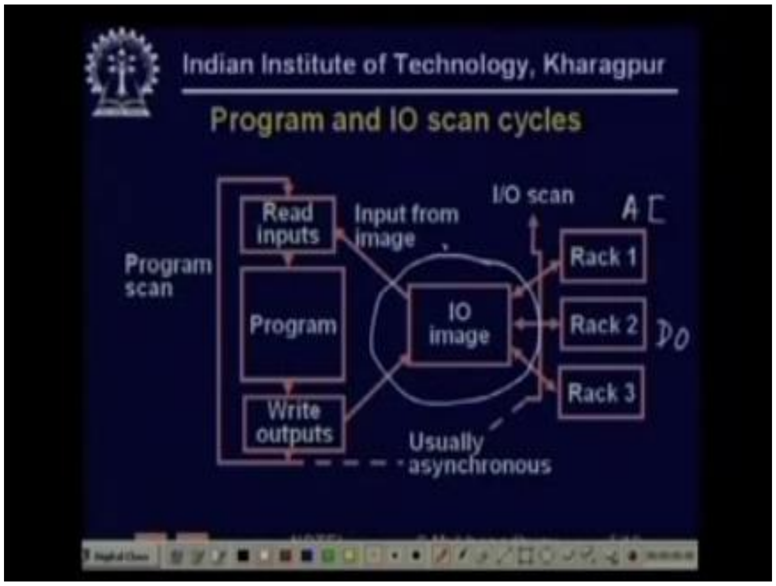
Physically situated either on the card or it could even be situated in the processor memory so it may so happen those who have studied micro processors will understand this that it may so happen that these cards every few units of time convert these values and then using some interrupt mechanism they will transfer these values to some part of the processor memory so what I am trying to stress is that here there are two kinds of cycles going on one kind of cycle continuously cyclically samples the physical inputs and transforms them to a part of the memory which we call the i/o image and the processor where it reads it actually reads from that image right.

(Refer Slide Time: 15:37)



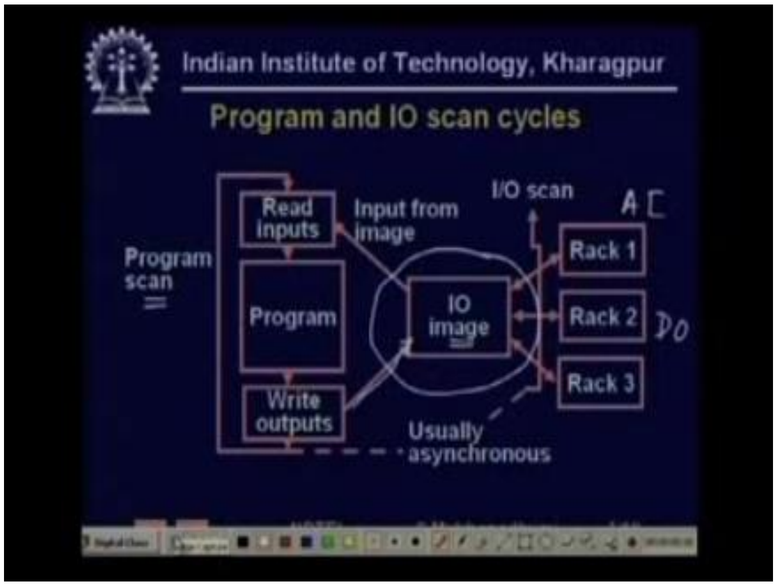
Now so having understood this so what is happening.

(Refer Slide Time: 15:45)



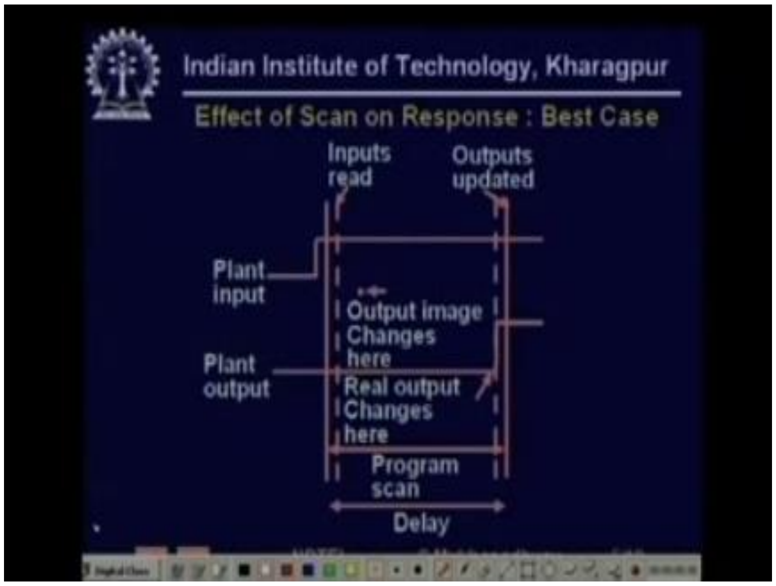
Is that so you see these are the various racks these are the various racks so these are the IO racks this could be an analog in this could be and this could be an analog input similarly this could be a digital output various kinds of things so there is a scanning mechanism which goes on which continuously updates either reads from the racks depending if it is an input rack or rights to a lag from a part of the memory this memory can be at the scanner can be at the cards at the various places while the one the main PLC program scanning.

(Refer Slide Time: 16:30)



Actually reads these IO images from this IO image it actually stores reads the inputs and whenever it actually produces outputs it also writes to this IO image and so it keeps writing to this IO image and then finally they get transferred to the racks so this is how the basic activities go on in a PLC so now we have to see that what is the result of these activities this kind of scan scanning what kind of response what kind of impact it has on the various input and output updating typically we are interested to know how much of delay we can expect in responding to an input change.

(Refer Slide Time: 17:26)

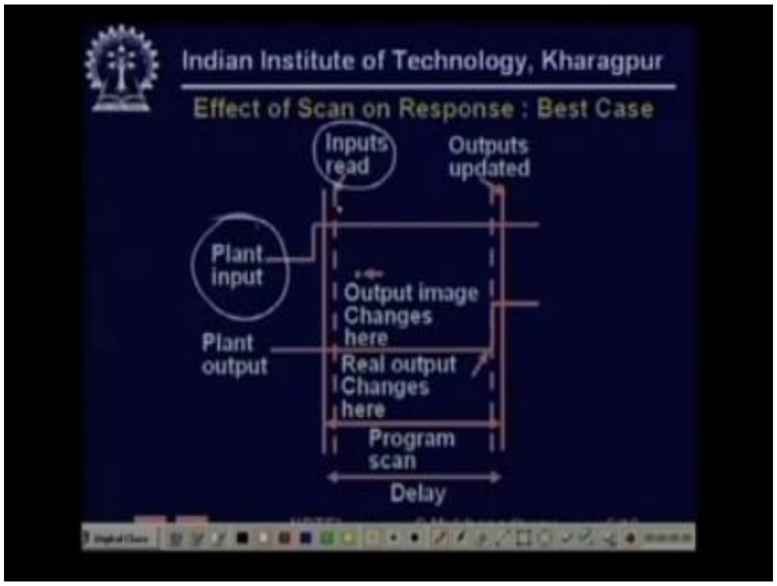


Indian Institute of Technology, Kharagpur

Effect of Scan on Response : Best Case

So imagine that this is a there could be a best case and there could be a worst case so we are first looking at the best case so what could happen is that.

(Refer Slide Time: 17:41)

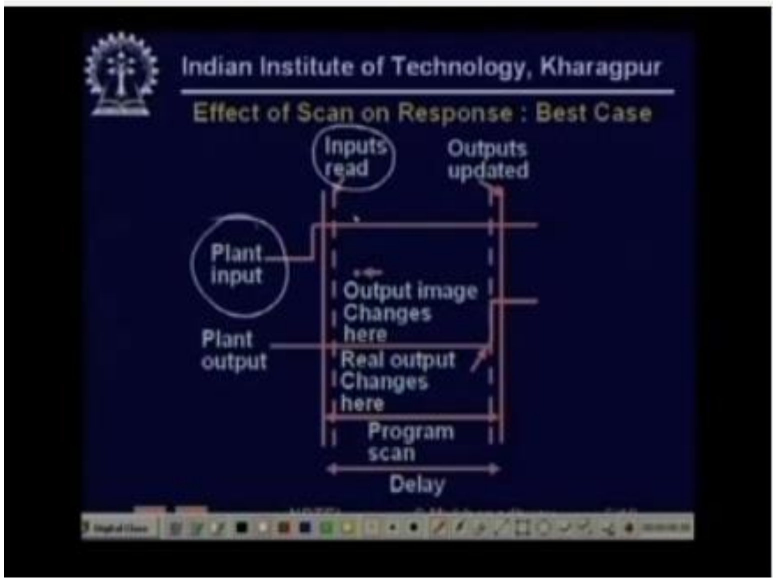


Indian Institute of Technology, Kharagpur

Effect of Scan on Response : Best Case

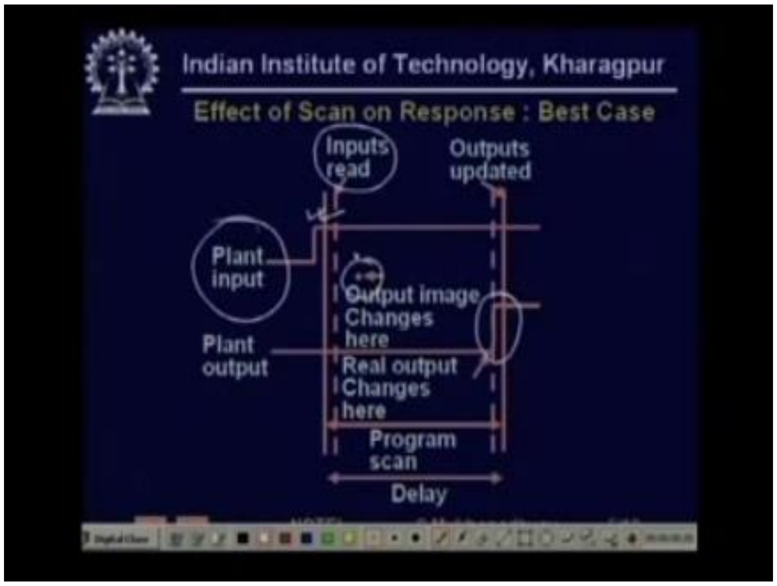
We remember that the plant input can change at any arbitrary period of time we cannot control that right so suppose the plant input change is just before and in the inputs are red so then immediately after it changes it will be read that is this change will be reflected and will come to the processor IO image very quickly.

(Refer Slide Time: 18:11)


Indian Institute of Technology, Kharagpur
Effect of Scan on Response : Best Case
Inputs read
Outputs updated
Plant input
Output image Changes here
Plant output
Real output Changes here
Program scan
Delay

Then the while the program will be well while when the program will be executed then in computing the output value the latest change of the input value will be taken care of will be accounted for so possibly due to this input change.

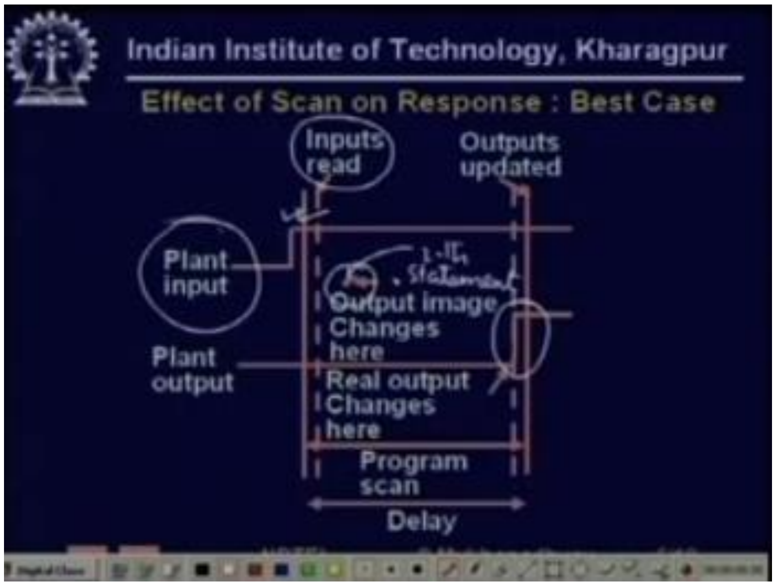
(Refer Slide Time: 18:32)


Indian Institute of Technology, Kharagpur

Effect of Scan on Response : Best Case

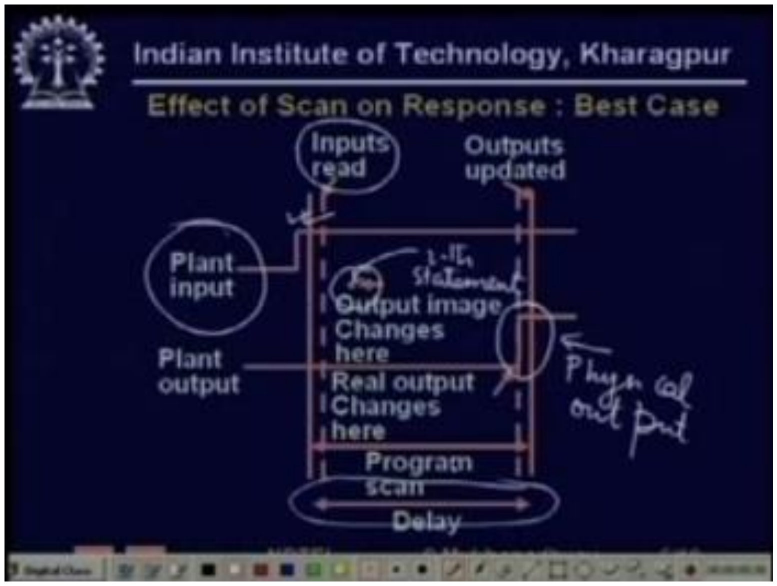
At the end of the execution there will be an output change now since it may actually happen suppose this is the IF statement this is the IF statement that is the effect of these of this input is used to compute some output at the IH logical statement so then what will happen is the during execution.

(Refer Slide Time: 19:02)


Indian Institute of Technology, Kharagpur
Effect of Scan on Response : Best Case
Inputs read
Outputs updated
Plant input
i-th statement
Output image Changes here
Plant output
Real output Changes here
Program scan
Delay

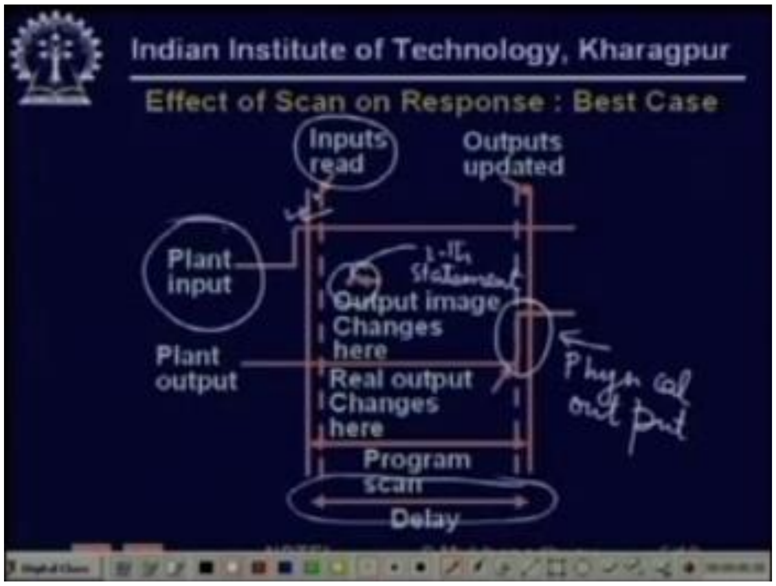
Here is the, this is the point where the ith statement gets executed. So actually in the processor memory the output changes at this point of time. However, it does not go to the physical world till the whole program execution has completed as we have said, so therefore the output.

(Refer Slide Time: 19:29)



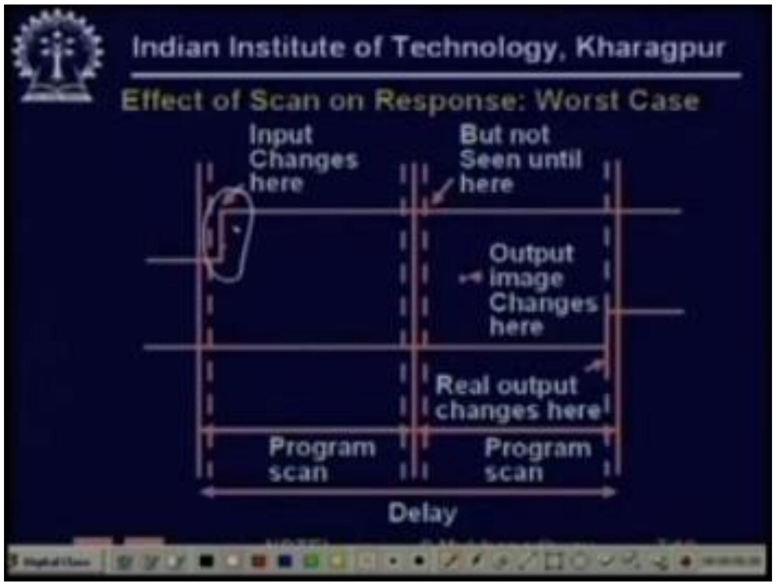
The physical output which goes to the process gets affected only at that time and there is a, this is the amount of delay that is bound to occur. So there is a the best-case response time is the delay of one scan which depends on many factors like the speed of the PLC execution also the length of the RL program extra. This is the best case it is called the best case because the input was.

(Refer Slide Time: 20:08)



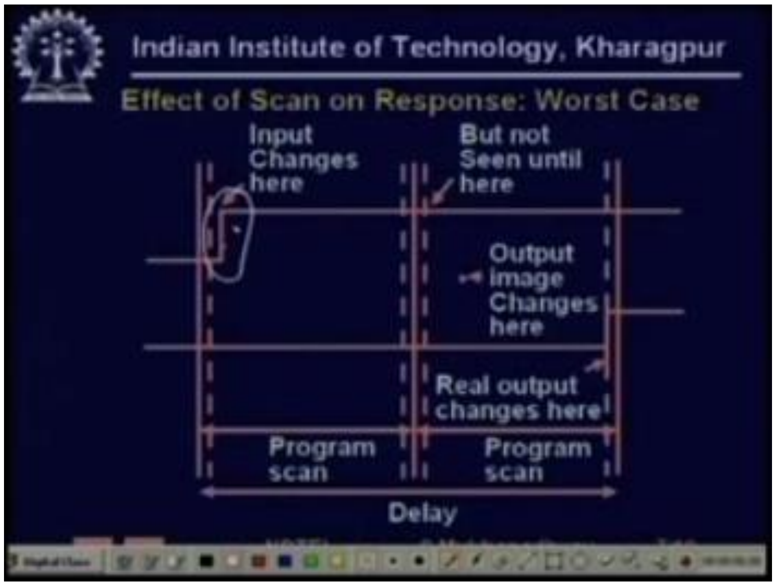
Because the input was read immediately after it changed so there was no delay in sensing the change in the input. In the next case as we shall show that we shall see the worst case, where this is just the opposite.

(Refer Slide Time: 20:29)


Indian Institute of Technology, Kharagpur
Effect of Scan on Response: Worst Case
Input Changes here
But not Seen until here
Output image Changes here
Real output changes here
Program scan
Program scan
Delay

So this is the opposite, this is the worst case so what is happening here, here unfortunately the input has changed just after a reading cycle was completed. So therefore, this change of the input was not sensed by the PLC in this scan cycle.

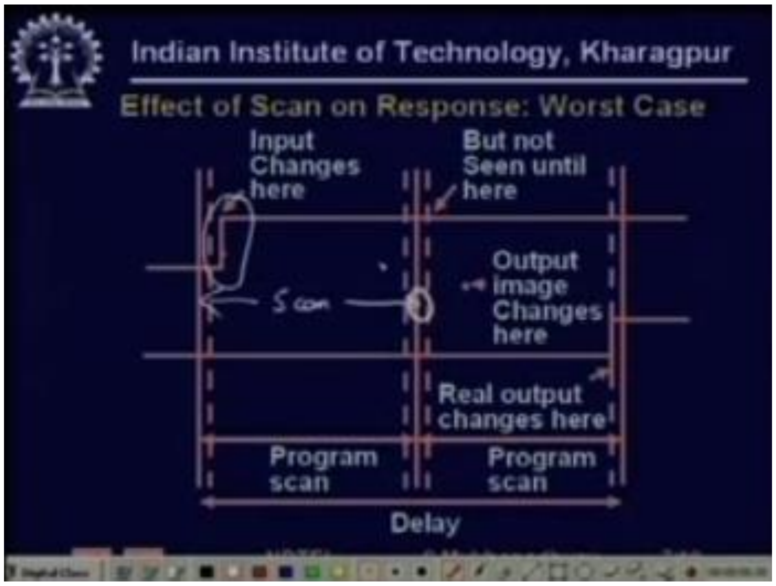
(Refer Slide Time: 20:58)


Indian Institute of Technology, Kharagpur

Effect of Scan on Response: Worst Case

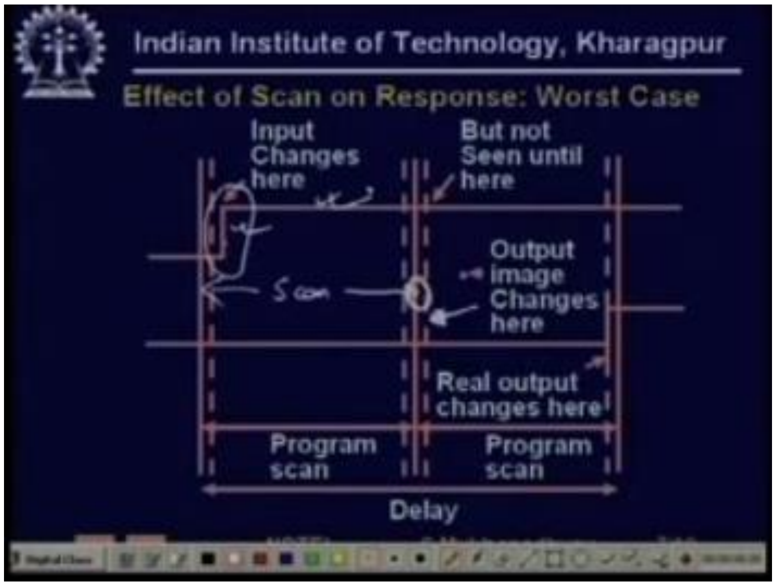
In this scan cycle it could, it did not sense that the input has changed, so therefore it computed by this time.

(Refer Slide Time: 21:11)


Indian Institute of Technology, Kharagpur
Effect of Scan on Response: Worst Case
Input Changes here
But not Seen until here
Scan
Output image Changes here
Real output changes here
Program scan
Program scan
Delay

It had computed and sent out the outputs once, but however those outputs did not reflect the new value of input which occurred just after the input in the cycle took place, so that is why it is the worst case. So it occurred so there is a maximum possible delay.

(Refer Slide Time: 21:35)

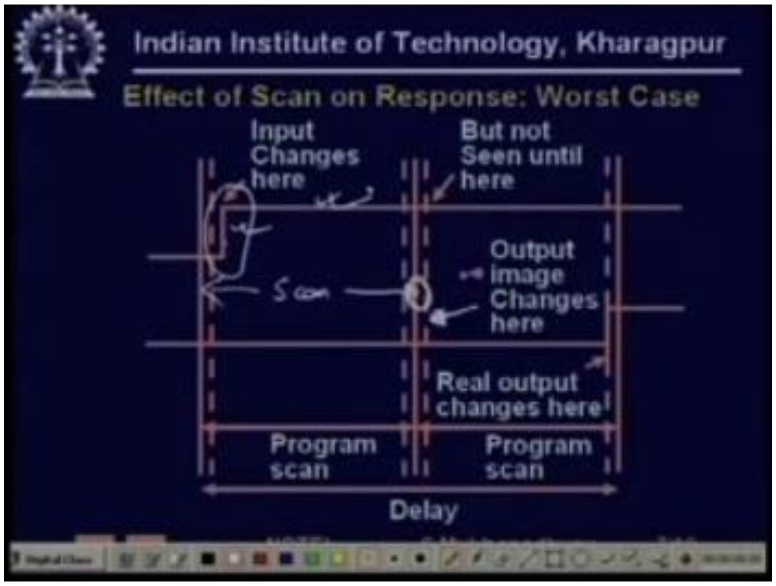


Indian Institute of Technology, Kharagpur

Effect of Scan on Response: Worst Case

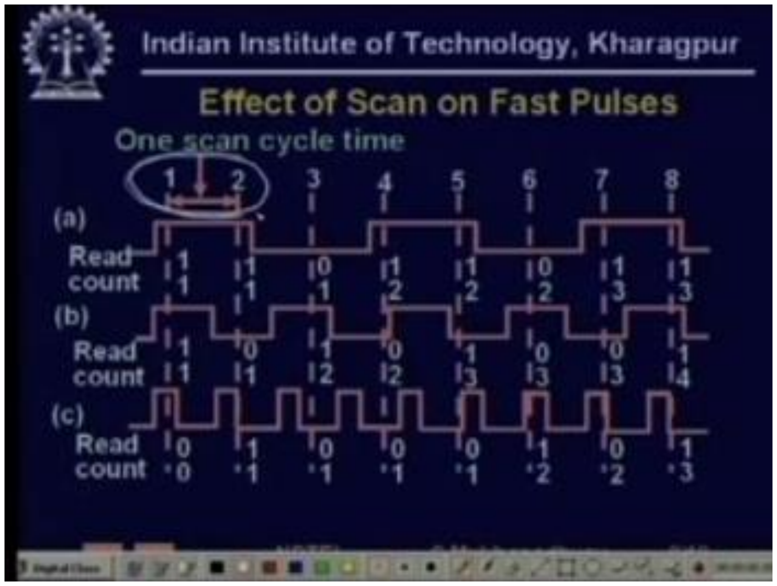
However, at this point in the second scan cycle inputs will be read again and this time this change is going to be sensed and now after this after having sense this there will again be one cycle of delay.

(Refer Slide Time: 21:49)


Indian Institute of Technology, Kharagpur
Effect of Scan on Response: Worst Case
Input Changes here
But not Seen until here
Scan
Output image Changes here
Real output changes here
Program scan
Program scan
Delay

So therefore the worst case of delay is two scan cycles, so that is why depending on the kind of response time one has to decide based on an application whether such delay times are acceptable or not. So we shall see that in certain cases these, such delay times may not be acceptable in which case you have to take additional measure in the sense that you might put additional cards or additional hardware for taking care of them. So this is the case where you get very fast pulses and one has to count those pulses.

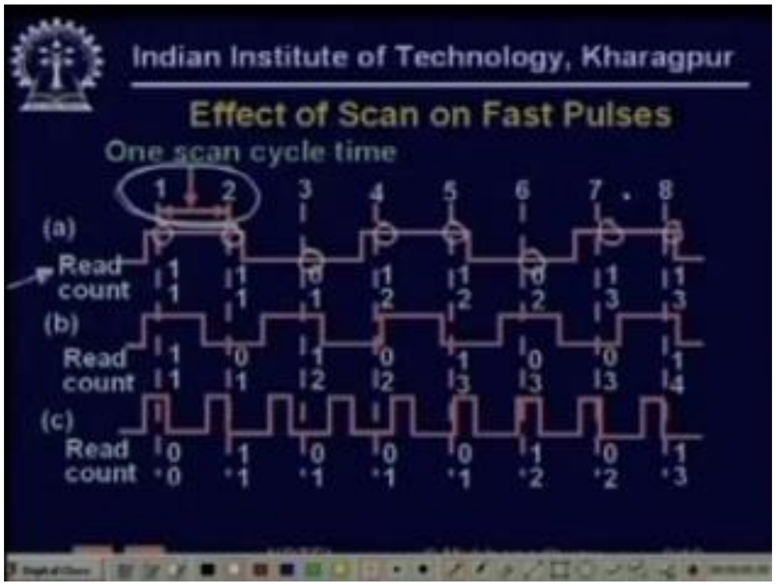
(Refer Slide Time: 22:37)


Indian Institute of Technology, Kharagpur

Effect of Scan on Fast Pulses

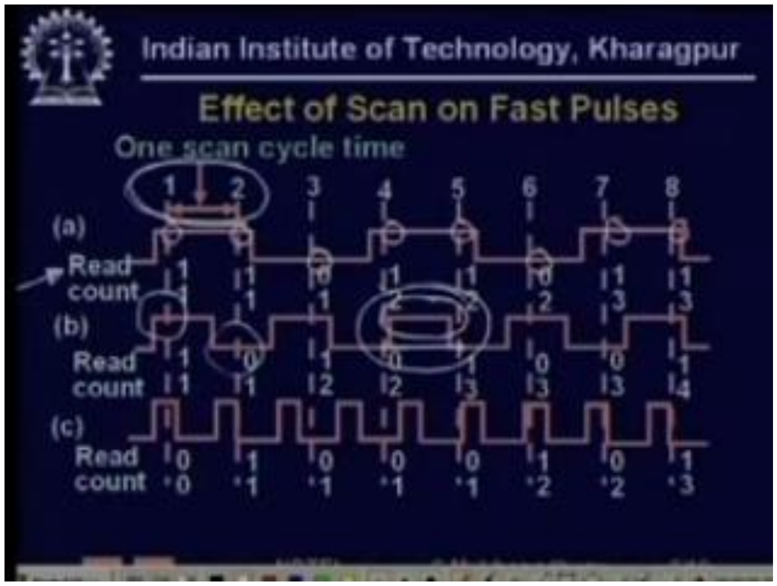
So in this case what we see is that suppose the scans the one scan time is this much, if we have to count the number of pulses which are arriving on a pulse train typically such pulses come from shaft angle encoders. If we have to count them using the PLC itself then.

(Refer Slide Time: 23:07)



If that we can see that if the pulse trains are slow enough there is a pulse width are sufficient then for this for this width of the pulse train all the level changes will be sensed so we are essentially going to count the number of pulses by counting the number of level transitions so then our count is going to be correct and if we compute the speed based on that the value of speed we get is going to be correct and so will be our control.

(Refer Slide Time: 23:53)

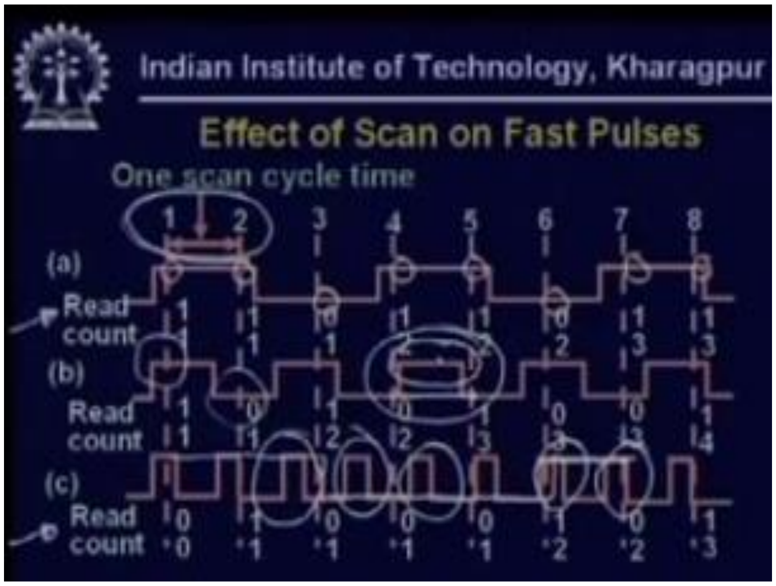

Indian Institute of Technology, Kharagpur

Effect of Scan on Fast Pulses

On the other hand if we increase the speed in which case the pulses will be of shorter duration and they will arrive faster so we will see that most of them are being still being sensed while suddenly one pulse is missed so you see that the program was canned for this pulse the program when the program is candidate the value is zero and when the programs can did next the value was a gain zero.

And so the program assumed that that it remains zero so one that it went up to one in between and came down is not will be the program will be unaware of that and therefore there then we perhaps small delay in the computation of the speed if you have faster pulses coming out.

(Refer Slide Time: 24:44)


Indian Institute of Technology, Kharagpur
Effect of Scan on Fast Pulses
One scan cycle time
(a)
Read count
(b)
Read count
(c)
Read count

Still faster than many such pulses will be based for example this one this one this one so it will be more of a chance to see when for example this pulse will be counted this pulse will be counted so you will get you know you will get drastic for example in this pulse actually we inferred when it went was sensed here it was one when it is sensed here it is also one so therefore it will be assumed that it continued at one.

So the inferred pulse is going to be in this case the pulse count will be severely wrong so it will assume that it has continued here and then it is continuing here and then it is continuing here and then it is continuing at this point is rising so the count will be a small fraction of the real speed and we are going to be totally out it is actually for this reason that for counting fast pulses you cannot use the PLC itself but rather use a special card.

So that card does not have the purpose of that card is solely to count the pulses coming from such a fast shaft angle encoder it is not it is not loaded with any other task so therefore it can really sample that line very fast and therefore can compute the speed and the speed can be made available to the PLC in terms of a value so rather than the PLC counting the pulses somebody else will count the pulse and we will convert it to a value which the PLC will read from time to time.

So it is for such reasons that sometimes you know special functional modules like as I as we mentioned in the earlier class that sometimes special you know high speed counter cards are needed it is for this purpose.