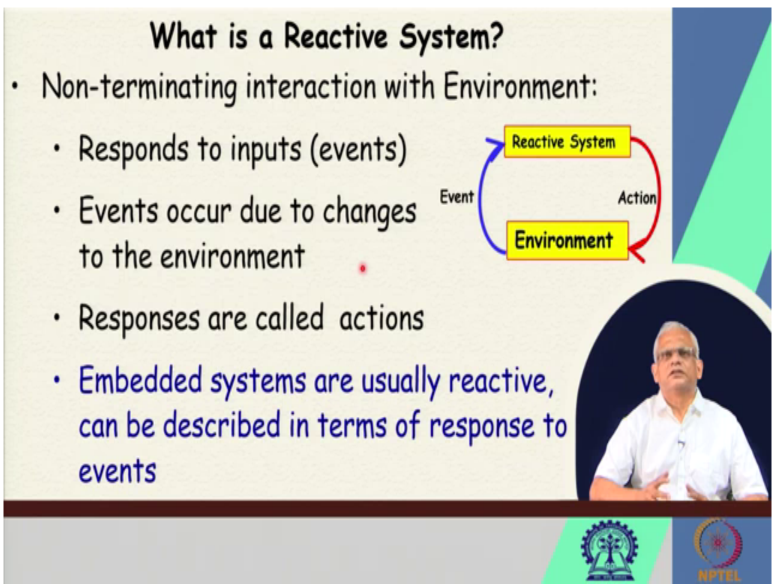
Real Time Systems
Professor Rajib Mall
Department of Computer Science and Engineering
Indian Institute of Technology Kharagpur
Lecture 02
Introduction – II

Welcome to this lecture. In the last lecture, we had a very brief introduction to real time systems, what is real time, what are these real time systems, where are they being used and so on. Let us continue from there. We are still in our introductory session. Let us look at some of the characteristics of these real time systems.

(Refer Slide Time: 00:38)



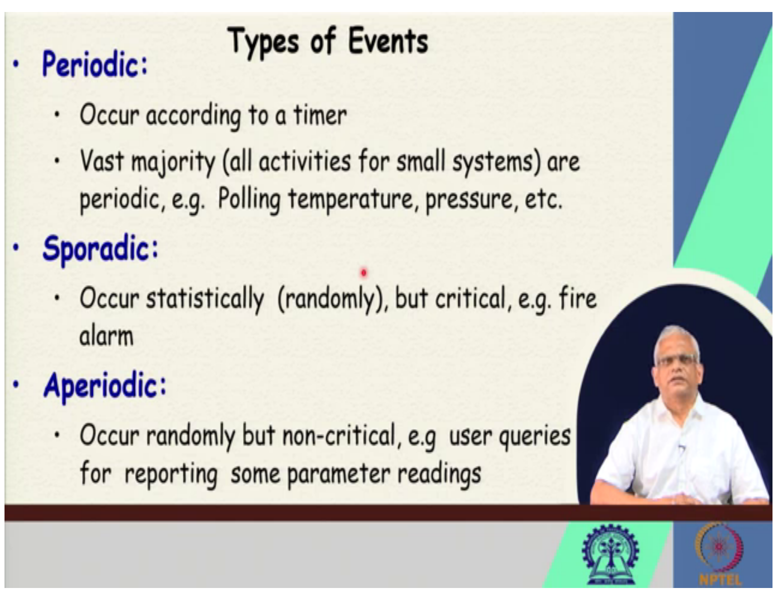
One important thing about these real time systems is that they are reactive systems. But then what exactly is a reactive system? Let us look at the concept of a reactive system and why are these real time systems reactive in nature? A reactive system is one where there is a non-terminating continuous interaction with the environment.

Let us look at this diagram. See here that we have this environment and there is the system which is taking actions on this environment. It is getting events from here and deciding what actions to take. Think of your cruise control. The last example that we discussed in the class, last class is about a cruise control system. In the cruise control system, the reactive system or the embedded system that samples the current speed of the vehicle and based on that is it going too fast compared to what was preset or is it going slow based on that it has to take action. So, we can say that that is a reactive system.

Similarly, an MPFI system, it gets various parameter reading from sensors including the current accelerator press, the current speed of the vehicle, temperature and so on. And based on all these events, it decides on which inlet at what time how much fuel is to be injected or which valve is to be opened, at what time for what duration. So, most of these embedded systems are reactive systems. They have a non-terminating interaction with the environment. They sample the input, read the events and then they compute certain things and then take action based on these computations.

And all these reactive systems if we want to describe their behavior, we need to describe, what are the events that the sample, what are the events they take action based on, they compute certain things based on which events and what are the actions they produced based on that. And that keeps on repeating, because it is a reactive system, it just keeps on repeating. They just keep on sampling those events, computing certain things based on that and then taking action.

(Refer Slide Time: 04:15)



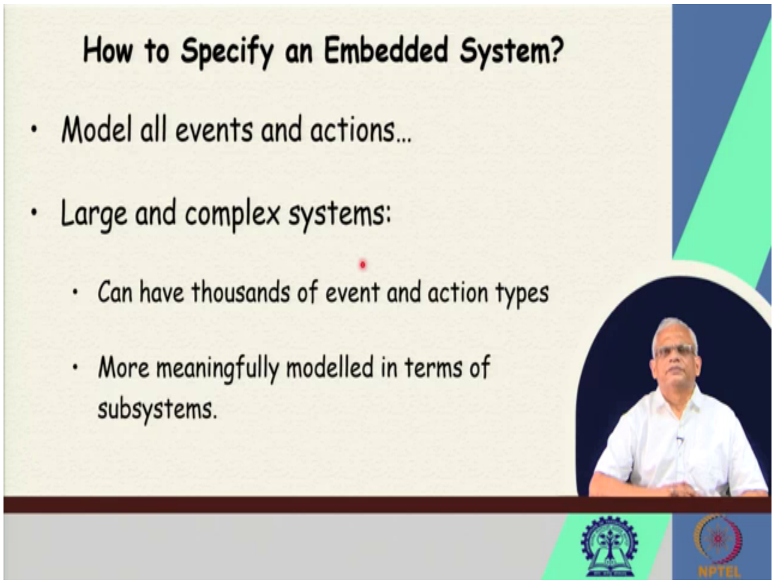
Now, let us look at the types of events in which these systems are interested, these reactive systems, what sort of events they are interested. One of the predominant or large portion of the events, large number of the percentage of these events are periodic in nature. Maybe 90, 95 percent of the events in a typical reactive system are periodic. Overwhelmingly, large part of the events are periodic in nature.

In the periodic events there is a timer inside the system and the timer keeps on giving a timeout after a fixed time interval and whenever the timeout occurs the system samples or in other words, we can say that systems samples its environment at a fixed rate which we call as the sampling rate. And as I was saying that a vast majority of all real time systems are periodic in nature and involve polling or sampling temperature, pressure, speed, humidity, etc, etc.

Another type of event are sporadic. The periodic events they occur at fixed time intervals, whereas sporadic events occur randomly or statistically according to maybe some distribution and often these are critical. For example, a fire alarm, never know when a fire alarm occurs. When fire conditions will occur nobody can predict, occur randomly. Similarly, an intrusion detection do not know when intrusion will take place. So, these are some examples of sporadic events. They occur randomly and the system needs to take action based on that.

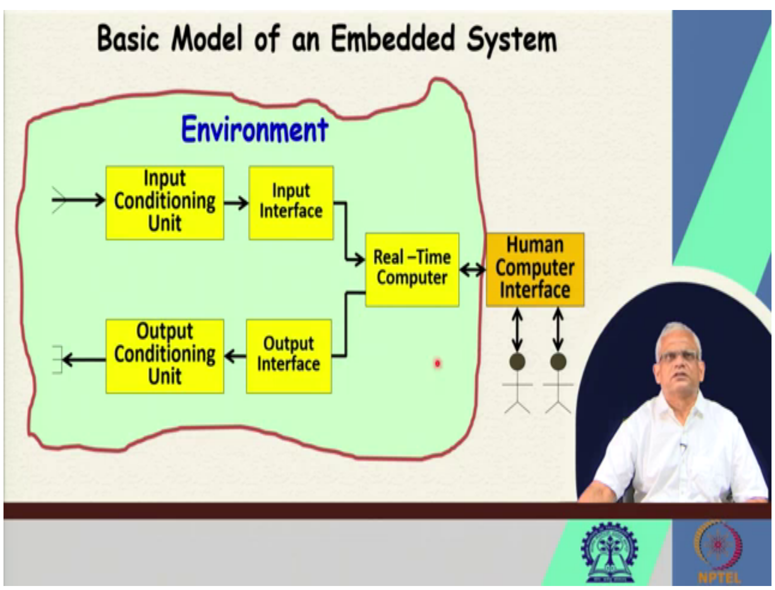
Another category of events are the aperiodic events. The aperiodic events are again random in nature. But, unlike the sporadic events, these are not critical. The system does not need to handle these events with the time constraint. For example, let us say a user queries that, what is the system parameter regarding to certain module. The user queries can occur randomly. But then the system does not have to handle it in time constrained manner as it would due to a fire signal.

(Refer Slide Time: 07:47)



Now, let us briefly look at how to specify an embedded system. An embedded system can be specified by listing all the events and actions. We do not normally list it, we model it, because the state of the system changes and we need to model it and these are often large and complex and can have thousands of events and action types. In such cases, we specify the subsystems and in terms of the subsystems we specify the full system.

(Refer Slide Time: 08:28)



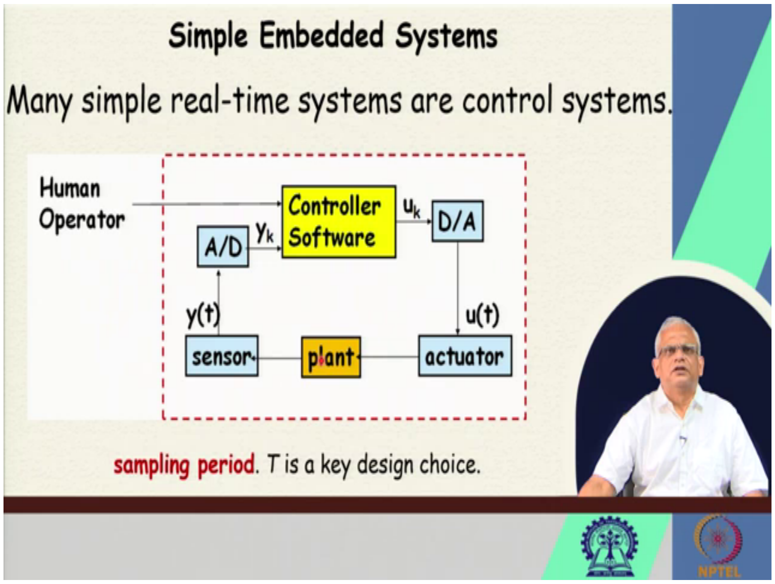
Now, we are still in our introductory lecture, and let us look at a very basic model of an embedded real time system. So, here, you can see the embedded real time computer here, which is part of its environment, maybe a chemical plant, large chemical plant and here that is a small computer which is controlling its actions. How does it control?

It has some sensors and the sensors pass on signal to the input conditioning unit, because the sensor signals are too feeble and in different form to be recognized by the computer. So, the input conditioning is used to make it suitable to be input to a computer through the input interface. The real time computer computes some functionality based on these and then throw an output interface and the output conditioning unit it drives some actuators, because these actuators might require large current voltage and so on.

The computer cannot generate analog commands to these. It just generate some digital commands which are taken care by this output conditioning unit to drive the actuators. And the human computer interface here is used by the human operators who can set parameters on the real time computer, monitor its performance, fine tune, and so on. So, this is a broad model of various real time systems.

We have the sensors, conditioning unit, input interface, and which report the events to the real time computer that computes and takes action based on an output interface and output conditioning unit and drive the actuators here. And this human computer interface to the real time computer using which the operators can monitor and fine tune various parameters of the real time computer.

(Refer Slide Time: 11:05)

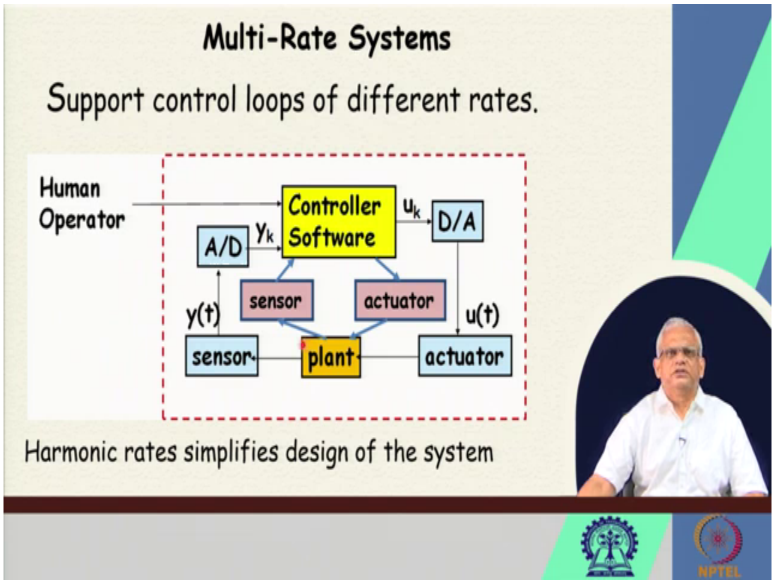


The simplest real time systems, we have this human operator and we have this control software running on some hardware and which produces digital output and that is converted to analog, because most of these actuators are analog. And then this is the plant environment like a chemical plant and this control it like opened a valve or opened coolant shower and so on. And then the plant condition is being sensed by the sensor which through an analog to digital converter, input digital signals to the computer and there is this human operator who controls the settings of the parameters.

But here the sensor senses at a constant time period, which we call as the sampling period, let us say it is T. And in one time period the plant condition is sampled. It is input to the control software through the analog to digital converter, the ADC, and the controller here it computes what actions to take and it gives commands which are taken by this digital to analog converter and which drives the actuator and the actuator acts on the plant.

So, every time period T, T may be in the range of few milliseconds or for some systems, it may be in the microseconds. But that is the typical range is a millisecond. And in let us say 5 millisecond is the sampling period then in 5 millisecond the sensor reading input to the computer, computation in the computer based on what action to take and the action itself all must complete within that 5 millisecond. So, this is the simplest embedded system. But many embedded systems are not like this.

(Refer Slide Time: 13:42)

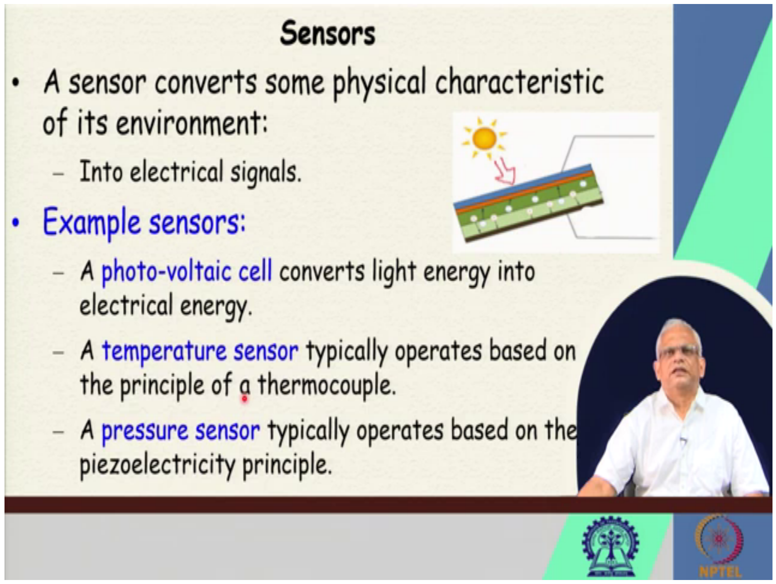


For example, we might have a multi-rate system. Here, it is not just 5 second, 5 millisecond that you need to take action, there are some parts of the plant which needs to be acted upon, let us say, in 10 millisecond and some parts maybe need to be acted upon in 2 milliseconds. So, those are the multi-rate system. Just see here that we have this same computer software.

The controller is taking action. One is it is driving this actuator every 5 millisecond. And this actuator is being driven every let us say 2 milliseconds and another set of actuators are being handled at let us say 10 millisecond and in that the sensor has to read once every 10 millisecond and then the computations occur and the actuator takes action on the plant.

So, a multi-rate system is a more complex system than a simplest system when there is a single rate. But then as we are saying that many times the events are not really periodic. There are many other types of events. So, the systems become more sophisticated if they have to handle even those events.

(Refer Slide Time: 15:18)

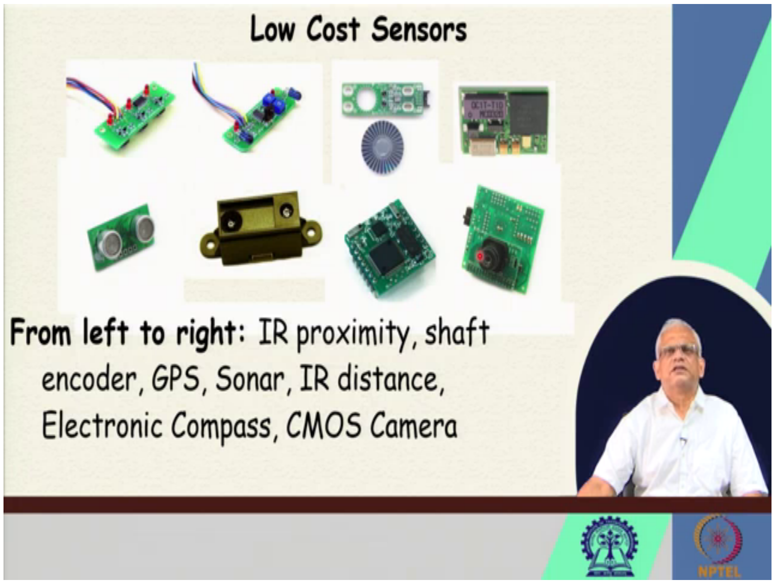


Now let us look at the sensors, because these are one important component of these real time embedded systems. To be able to know these real time systems, we should have some idea about the sensors. A sensor convert some physical characteristics of its environment, that is for example, a plant, a chemical plant or something. It converts it into electrical signals which can be converted to a form which can be read by the computer.

Let us look at some example sensors. One example is a photovoltaic cell. Let us say we want to measure the amount of light, we can use a photovoltaic cell as a sensor, which measures, which based on the light it falls on it, it generates electrical energy. Here based on the light, let us say we want to have a control system, a real time control of switching on and off light, when it becomes dark, the lights will be on and when it is let us say the light intensity is high, the electric lights should be switched off. So, here we can use the sensor as a photovoltaic cell.

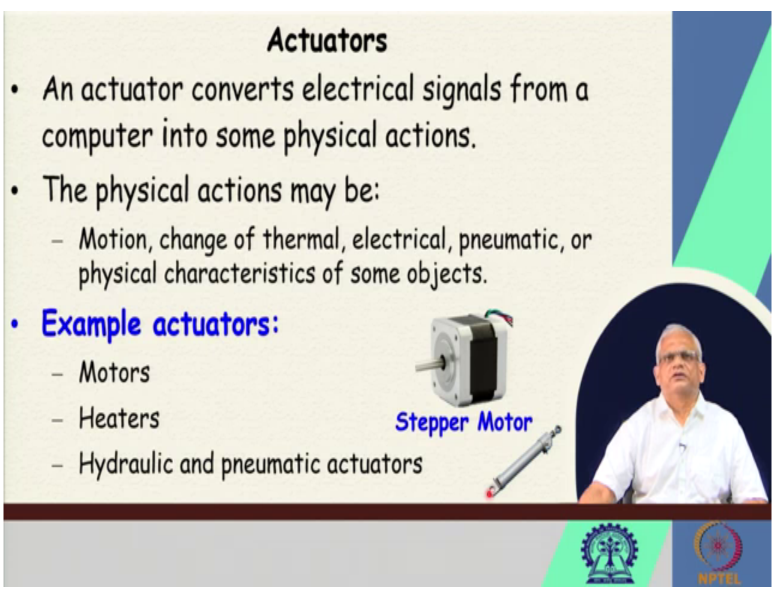
We can have temperature sensors. There are many types of temperature sensors. And one prominent type of temperature sensor works on the principle of a thermocouple. So, when we have this material placed at different temperatures thermocouple it generates electricity. We might need a pressure sensor. For example, we might have pressure sensors based on the piezoelectricity principle. When pressure is applied on some material, they develop electricity.

(Refer Slide Time: 17:36)



These are some examples of low-cost sensors. From left to right, we have the IR proximity, so proximity sensors, infrared based proximity sensors, soft encoders, GPS, IR distance, electronic compass and CMOS camera. So, these are some of the sensors that might be used in embedded application. But then there are a large number of sensors, just given here a few examples of low-cost sensors.

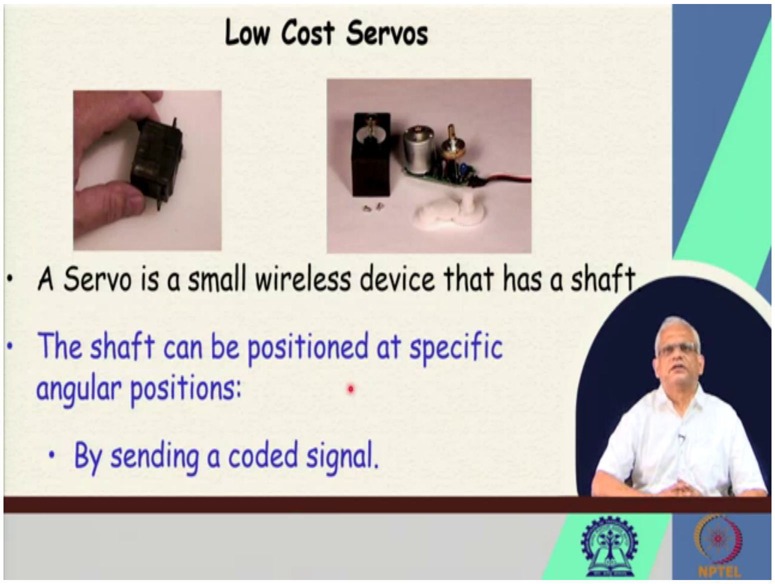
(Refer Slide Time: 18:17)



Now, what about the actuators? There are also many types of actuators which are used in these embedded applications. The actuator says we have been saying that they take the electrical signal from the computer and perform some physical action. For example, the physical action can be in terms of motion or change of thermal characteristic, heating, let us say, or cooling, it may be change of electrical characteristic, it may be pneumatic or physical characteristic of the object.

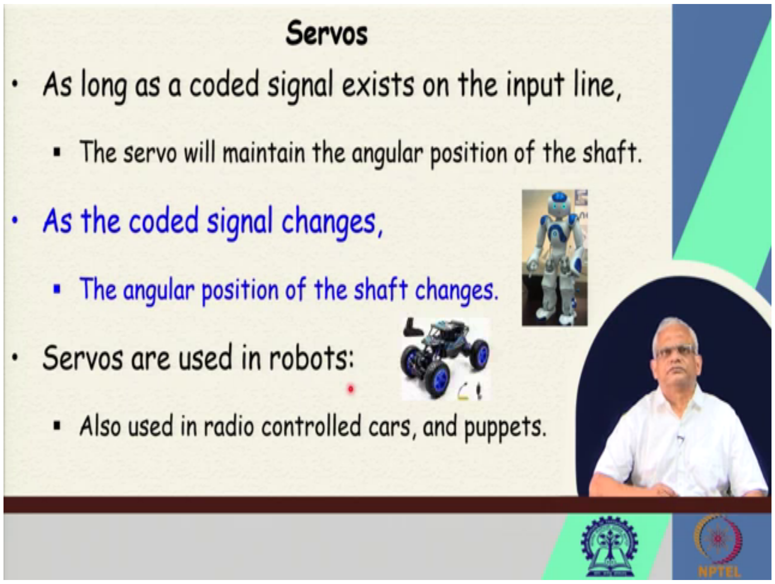
Let us look at some example actuator. A motor is actuator which can result in a motion, for example, a robot. We might use a stepper motor where we might control the movement of the robot, the direction and the steps and so on. We might have heaters as actuators which change the temperature of the environment. The environment and the plant we are using it synonymously. There may be hydraulic and pneumatic actuators. For example, look at here this is a hydraulic actuator. So, here it translates the pressure, input pressure to output, maybe it magnifies that pressure through hydraulic means or pneumatic means.

(Refer Slide Time: 20:10)



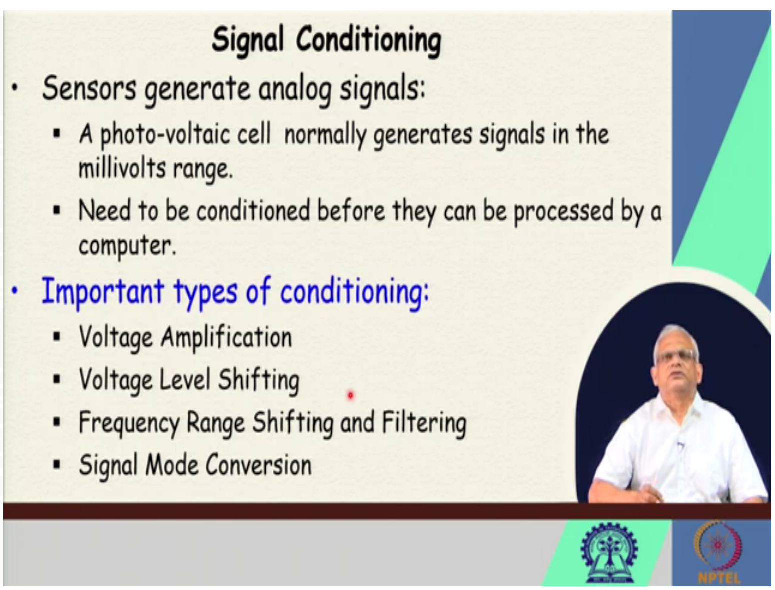
This is another type of actuator are the servos, very popular in embedded applications. A servo is basically a small wireless device that has a shaft. Look at here the shaft here. So, the shaft can be positioned at specific angular position based on some commands. The commands are basically coded signals given by the computer and the computer can change the angular position of the shaft and these are very popular in robot applications.

(Refer Slide Time: 20:44)



Based on the computation by the computer, it may want to change the direction of the robot, for example, so it will give a command, a coded command to the servo which will change its angular position and based on which the robot will appear to change its direction of movement. And the servos are also used in radio-controlled cars, puppets and so on.

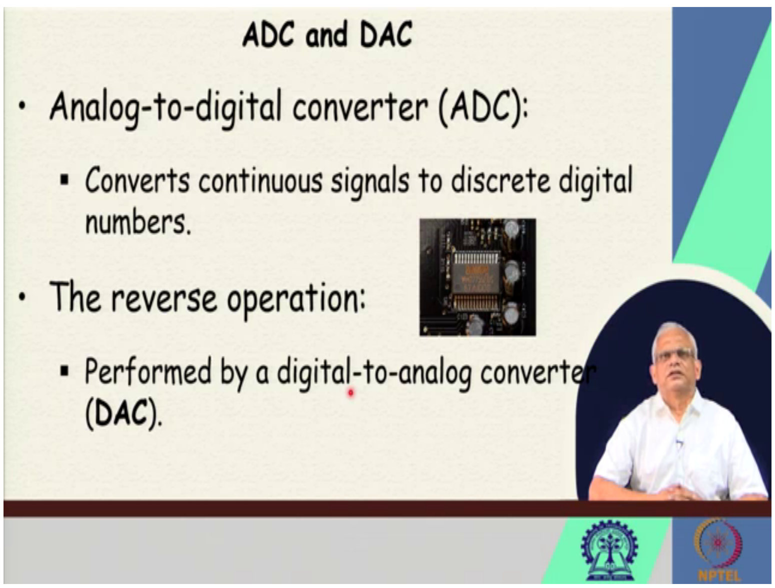
(Refer Slide Time: 21:25)



The sensor signals are sometimes very feeble. Think of the photovoltaic cell, the signal may not be interpreted by a computer, because these are too feeble, maybe need to be amplified. And also, this signal maybe analog need to be converted to digital. So, those are the things that we call as signal conditioning. The sensor signal needs to be conditioned before it can be read by the embedded computer. Many of the sensors they generate analog signals, for example, photovoltaic cell, millivolts range, whereas the computer needs to read in few volts. So, they need to be conditioned.

The important type of conditioning is voltage amplification, voltage level sifting, frequency range sifting and filtering and the signal mode conversion, for example, we need to change the analog signal to digital and vice versa.

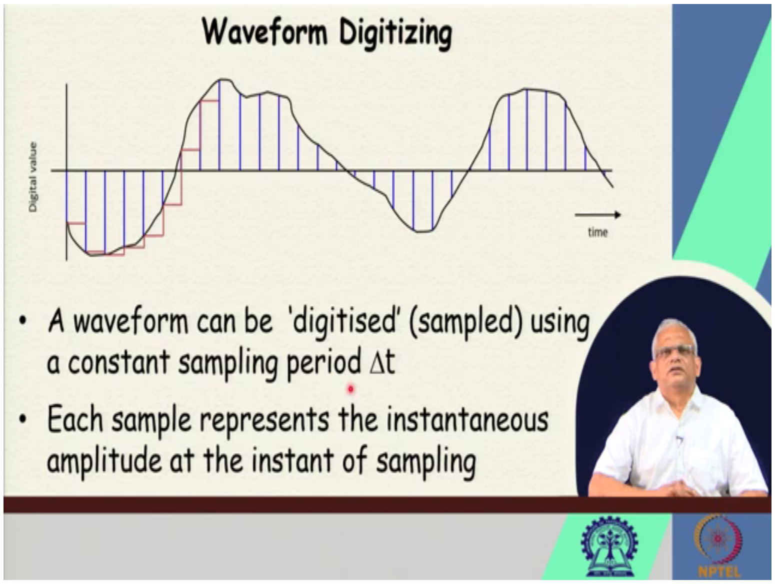
(Refer Slide Time: 22:42)



Now, let us look at how to do the analog to digital conversion and digital to analog conversion very basic topic. Maybe many of you know this. But just for completeness and especially for those who do not know, we just have one or two minutes on analog to digital converter and digital to analog converter.

So, here we convert a continuous signal or the analog signals into digital numbers. The reverse operation is that the computer generates the digital signals and we want to change that to an analog signal to be again conditioned by a signal conditioning unit before it can drive an actuator. So, the ADC and the DAC are frequently used in many embedded applications. And we have now chips available which do this work convert analog to digital signals and digital to analog signals.

(Refer Slide Time: 23:56)

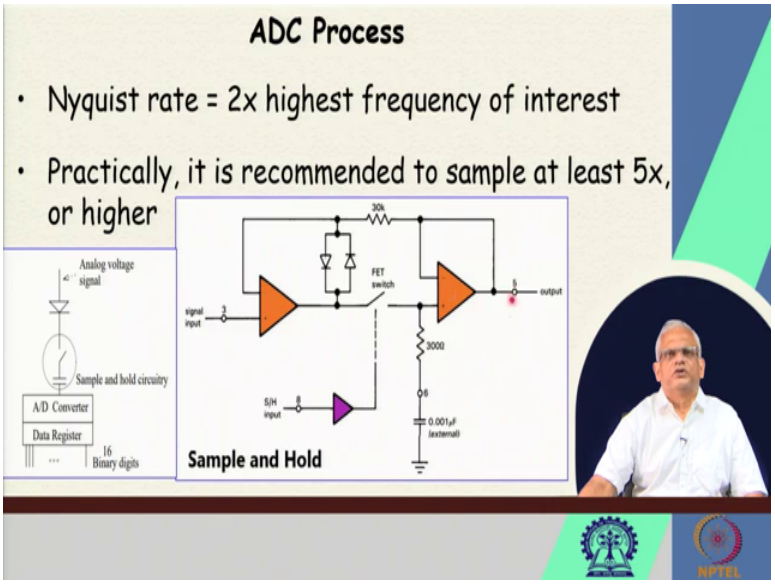


Just to get slightly more insight into this, look at this waveform here. Look at this waveform. It is an analog signal maybe generated by a sensor. And let us say we want to digitize it. Typically, we sample it at a constant rate, which we call as the sampling period, which is a very important parameter. Let us say delta t is the time with which we sample this.

Now, in sampling, we read the amplitude of the signal at that time. And then we assume that it remains same until we sample it again. So, if we sample it at a fast rate, we will get almost the exact analog waveform. But if we sample it at a very slow rate, then it will appear more and more distorted. So, what is the acceptable sampling rate?

We do not want to really sample too fast, because that will be too time consuming. It will occupy our embedded device and it will also create other difficulties like it may generate large number of data. So, what is an optimal sampling period?

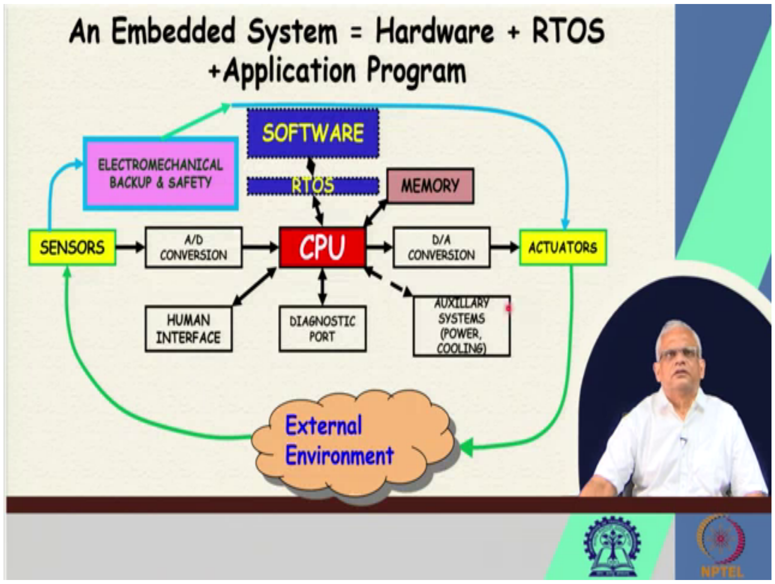
(Refer Slide Time: 25:43)



The optimal sampling period is called as a Nyquist rate, which is twice the highest frequency of interest. But normally, we do not just do twice the highest frequency of interest, we sample at least five times or higher. Now, let us look at the analog to digital conversion. This is the reverse process of digital to analog conversion. And here as you can see in this scheme, as the analog signal comes, we sample that and hold it. It does not change.

We just sample it once and it is held in this capacitor and then this analog to digital converter process where we convert the sample into the digital form which is stored in a data register. So, this one that we have this analog voltage and we want to get the sample this is the one which is used the circuit here the sample and hold. So, here the switch samples and this op amp it sets itself to the output, which is the sampled value and this can be changed to digital form.

(Refer Slide Time: 27:19)



Now, we are almost at the end of this lecture, but then we will briefly look at what is an embedded system. An embedded system as per our definition consists of some hardware like digital to analog conversion, analog to digital conversion, the CPU, memory, and so on, some the real time operating system and some software which controls this and we have few other software like human computer interface and so on.

So, here we have a electromechanical backup. We will see why it is needed. Because once the system fails, we do not want any accident to occur. We should have a fail-safe operation. We will look at these concepts in the next class, the fail-safe operation. These are the different hardware components the actuator, digital to analog conversion, sensor, analog to digital conversion, CPU, memory, etc. some software, some the real time operating system and some the application software which controls this. So, we will stop here and we will continue from this point in the next lecture. Thank you.