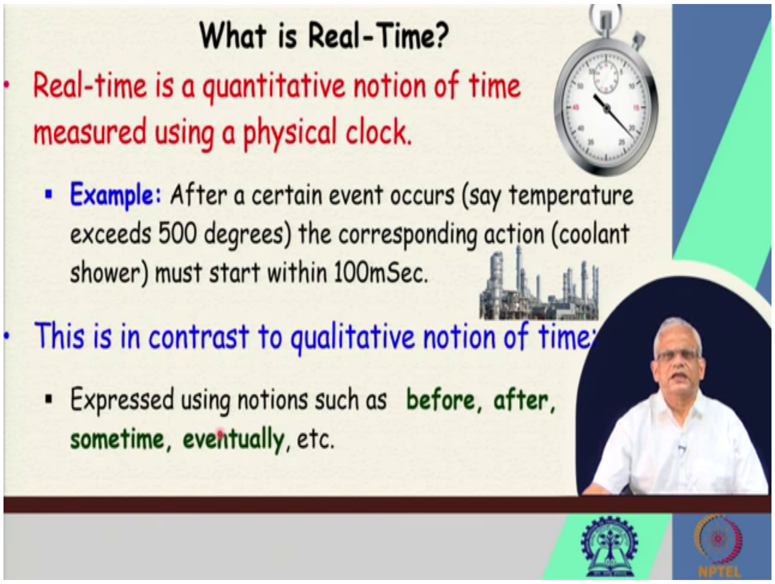
Real Time Systems
Professor Rajib Mall
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur
Lecture 01
Introduction - I

(Refer Slide Time: 00:20)

Welcome to this first lecture of Real Time Systems. This course will be handled by myself. My name is Rajib Mall, and also by Professor Durga Prashad Mohapatra. We will be sharing this course and as is with all other courses, we will start with a brief introduction. We will discuss what is real time, what is a real time system, where are these used, why is it important, how do we go about learning it, what are the different systems which we encounter and which you do not normally encounter, where do we anticipate that this will be used and so on. So, without wasting much time, let's get started with a brief introduction to what is real time.

(Refer Slide Time: 01:15)



A real time in the context of a real time system, means a quantitative notion of time measured using a physical clock. For the real time systems, we need to specify their behavior by using a physical clock. For non-real time systems, we don't do that. But for real time systems, we need to deal with this quantitative notion of time which will be measured using a physical clock.

Now, just to give one example, the behavior of a chemical plant like this maybe that after a certain event occurs, let us say, the temperature of the reaction chamber exceeds 500° C, then the corresponding action which happens to starting the coolant shower must occur within 100 milliseconds. So, this 100 millisecond we are saying with respect to some physical clock that when the event occurs, that is the temperature exceeding 500° C, the start of the action i.e., starting the coolant shower must occur within 100 milliseconds. Here, we are dealing with real time. We can contrast this with what happens in the traditional notion of time where it is a qualitative notion.

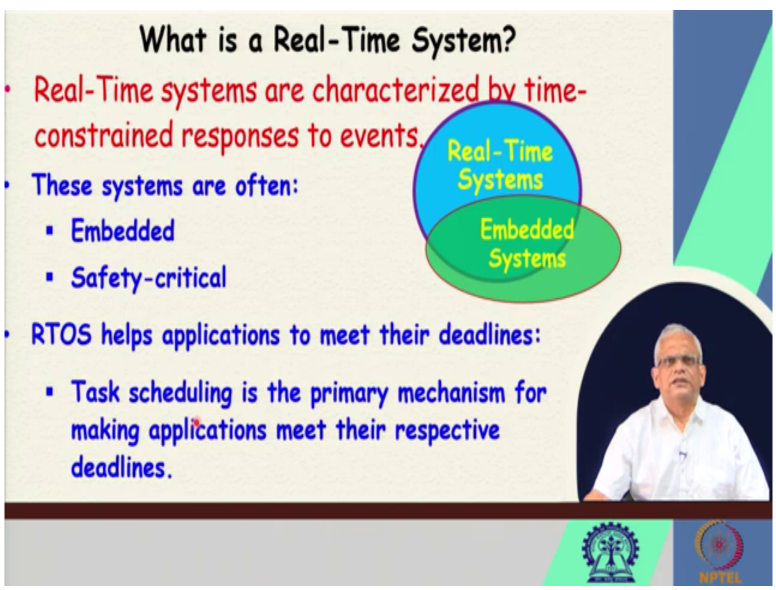
In most other systems which we encounter, which you have dealt with, studied there use qualitative notion of time. So, there the behavior of the system is expressed using notions like before something, after something, sometimes something will occur, eventually an event will occur and so on.

Just to give an example, let's say, we are aware of a library information system, we can say that a user can issue a book only after he has registered as a member of the library. So, here it is after can be any time. He may become a member and then instantly he may issue a book, he may issue after a month. So, there is no restriction with respect to a physical clock. It is only that he has to just issue book after he has registered as a member.

In most systems, the notion of time is qualitative and we do not need a physical clock to express the behavior. We just express the behavior like we want to specify a system. We specify by saying after, before and so on. But if we want to specify the behavior of a real time system, we have to have this clock: a physical clock and the actions will have deadlines specified with respect to this clock.

Once the event occurs, we will specify the time of reaction using the reading on this physical clock. So, that is one very basic thing about real time systems is that the notion of time is quantitative here and expressed using a physical clock that is part of the system.

(Refer Slide Time: 5:46)



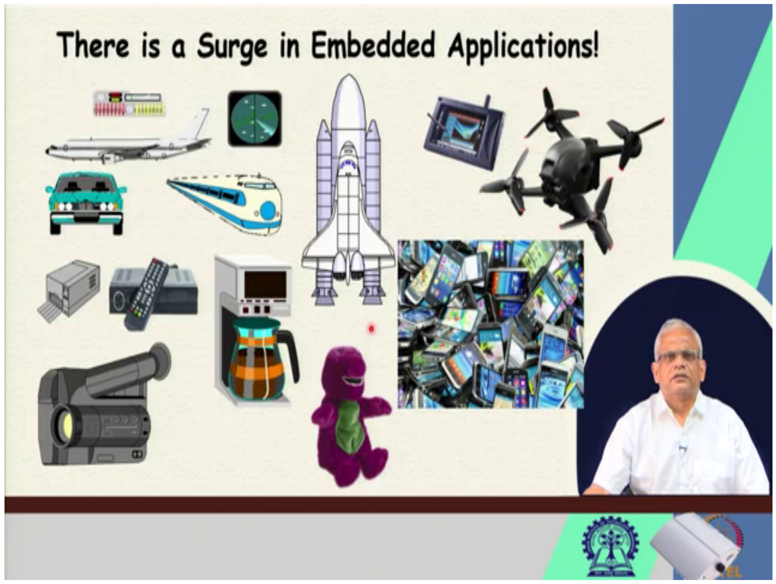
Now, what is a real time system? A real time system is one whose behavior is described by time constraint response to events. The real time systems respond to events in a time constraint manner. We just looked at the chemical plant, where we had a time constraint reaction to the temperature exceeding 500° C. But these systems are overwhelmingly embedded systems, a large percentage of real time systems happened to be embedded and also many of these are safety critical.

So, as we show here in this diagram, many of the embedded systems are real time. But we can argue that not all embedded systems require real time behavior. And similarly, not all real time systems are embedded. Here, one of the primary things in a real time system is that the system has to meet time constraint behavior, but then the question comes that how does it meet? If we restrict the system to respond within certain time how can one make it, meet its deadline?

The answer is that, the real time operating system is the one which if used properly can help meet the deadline. It should now become obvious that for a real time system one of the most important thing is the operating system, because that is the crucial element which helps the system to meet its time constraint behavior. But the next question that comes to mind is that how does the real time operating system do it? How does it help the tasks to meet their deadline?

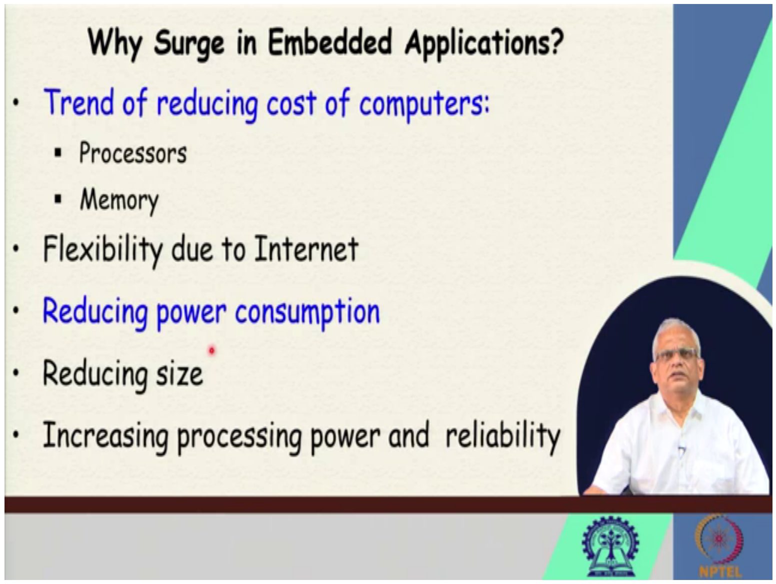
The answer is that, the task scheduler which is part of the real time operating system is the one whose primary responsibility is to make applications meet their respective deadline. So, the tasks scheduler will see that it knows somehow, it determines which tasks are critical, which are going to miss the deadline and start running them. And somehow, we will see that it's an interesting and large topic by itself, the task schedulers for real time systems and they make these systems meet their deadlines.

(Refer Slide Time: 09:39)



As you might have already imagined that since many of the embedded real time systems are real time and there is a surge in embedded application, naturally the application of the real time systems has increased tremendously and is growing at a very fast rate. Look at all these different gizmos, appliances and so on, starting with transportation, a car, an aero plane, train, a laser printer, a video recorder, a drone, mobile phones, toys, rockets, nuclear power plants and so on. The applications are huge and they are growing. So, this is a very important topic. The real time system area is a very important topic and that is the focus of this course.

(Refer Slide Time: 11:12)



Now, let us try to just for a second reflect on why the embedded applications are growing at such a fast pace. One is that, of course, which anybody can guess is that the cost of computers has come down drastically. From millions of dollar, it has become a dollar, something like that. The cost of processor, memory and so on have fallen dramatically. And it has become affordable to automate applications which were once thought to be very difficult to automate given the cost of automation.

The other factor is that of Internet. The flexibility due to Internet has caught the fancy of the users. Internet-enabled devices have large demand and the flexibility really sells, and therefore, there is a popular demand on these embedded applications which are Internet enabled. Some time back power consumption was a big issue. You develop a system, sell it and then the users find that they are disappointed. The battery wears out. It is hardly usable. But now with reducing power consumption rates and increasing capability of the processors, flexibility due to Internet has caught the consumer fancy, and in addition to that, reduced size.

Most of these embedded systems are hidden inside a large system to such an extent that you can hardly recognize that there is embedded system inside it, there is a real time system which is controlling it. You can't see it. It is become so small. Just think of a mouse with a computer you are operating the mouse and just thought that you are operating the same mechanical mouse that we used to operate 20 or 30 years back but nothing can be far from truth.

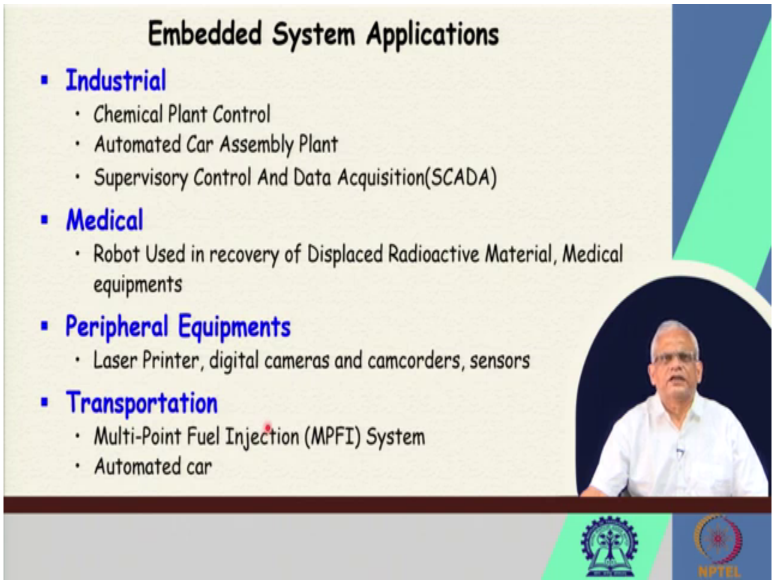
Open the mouse see that there is a processor memory. It is an embedded system. It is through laser light. It is reading the trackball. Earlier the mechanical device is gone, now we have this laser. And the movement with respect to the laser light is tracked by the processor. It is computing the displacement and transmitting, very accurately.

There are hundreds of examples where you in the everyday life think that the system is same just like I gave an example of a mouse, but then it is not the same system, inside there is an embedded system which has appeared quite some years back and which is controlling its behavior, it is not the older mechanical device.

Just think of the car, 30 years back there were no processors, no embedded system inside a car. Now, a car has hundreds of processors. And if it doesn’t have the basic processing power inside it, it doesn’t give even certification. It is not roadworthy. So, we will just come to that topic, because that is an important application of real time systems. In our introductory lecture, we will just see how these embedded systems are used in transportation, namely a car, and of course, the other transportation devices as well.

And the computers now have large processing power and also the reliability has increased tremendously. 20, 30 years back, they used to break down, the computer used to hang, necessitating reboot and so on. Now, those things are gone, very fast processor and very reliable. So, these are the key elements which are driving the increased use of embedded applications, and with that, the real time systems.

(Refer Slide Time: 17:07)



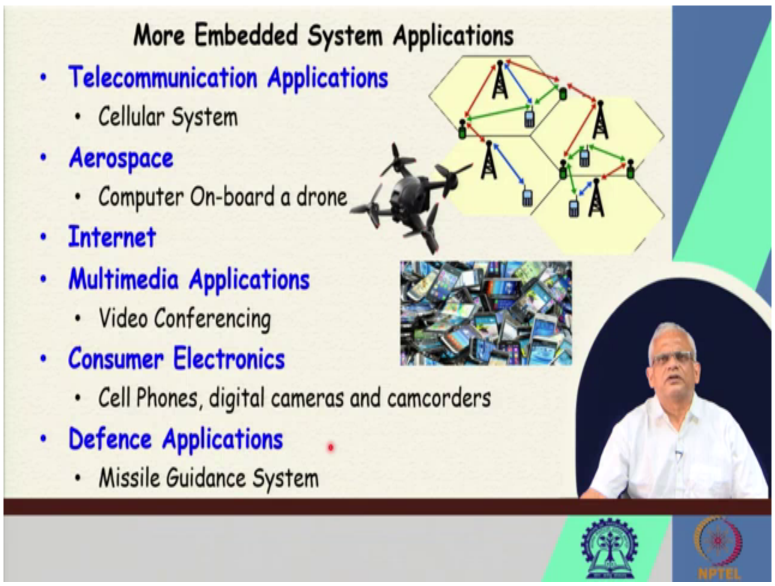
Now, let us just try to identify few of these embedded real time applications. As I was saying that there are millions of applications, according to some estimate, the number of processors manufactured worldwide about 85 to 90 percent of those processors are used in these embedded real time applications and only that 10 percent of the processors worldwide are the ones which are used in terms of conventional computers like laptops and desktops or servers.

Now, let us look at few of the embedded applications. We will look at industrial, medical, peripheral equipment and transportation and few others in the next slide. In the industrial application, we had seen a chemical plant controlled by a real time embedded system. An automated car assembly plant, you go into the car assembly plant and find that hardly there is any worker. It is only the robots which are working. There are robot stations. The partially assembled car is passing through these robot stations and they are quickly completing the car manufacturing. The supervisory control and data acquisitions, the SCADA applications are a large application area of the real time embedded systems.

In the medical area, there are many uses. For example, a robot assistance surgery, a robot used in recovery of displaced radio material, other medical equipments and so on. These are also present in large numbers in peripheral equipments, for example, laser printers, digital cameras, camcorders, sensors, and so on. In transportation, starting from an aero plane to a train to a automobile, the use of embedded applications have become prevalent.

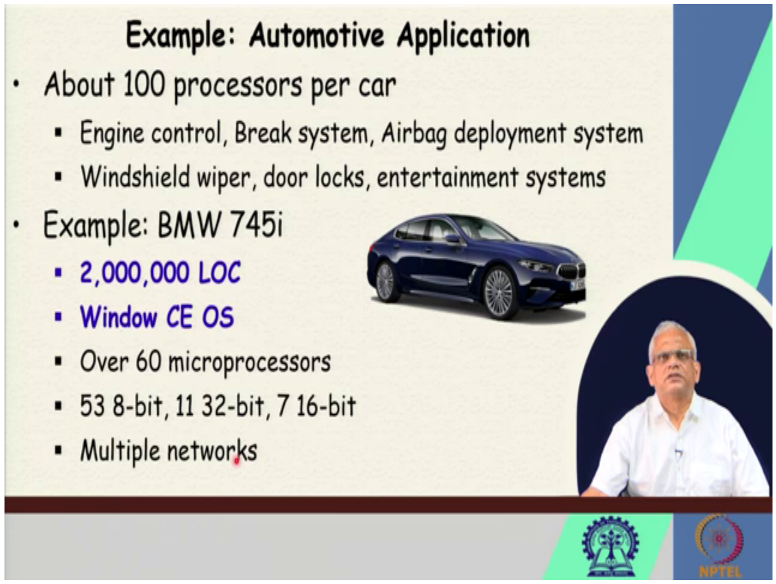
We will just look at a small example of an automated car and we will look at the multipoint fuel injection system MPFI which is part of every car now. Unless you have the MPFI system, your car is not roadworthy according to the Bharat VI, BS VI standard.

(Refer Slide Time: 20:07)



The telecommunication domain, we have the cellular systems, VoIP and so on. This is the cellular system where we have these embedded real time systems used abundantly. The aerospace, a computer onboard, a drone are some examples. In the Internet, working of the internet, routers and so on, internet devices, we need these real time embedded systems. Multimedia applications like video conferencing, we need these real time embedded systems. Consumer electronics like cell phone, digital cameras, camcorders, we need these embedded real time systems. In defense applications, like a missile guidance system is one of the original areas, the defense application, where real time systems have been used for a long time.

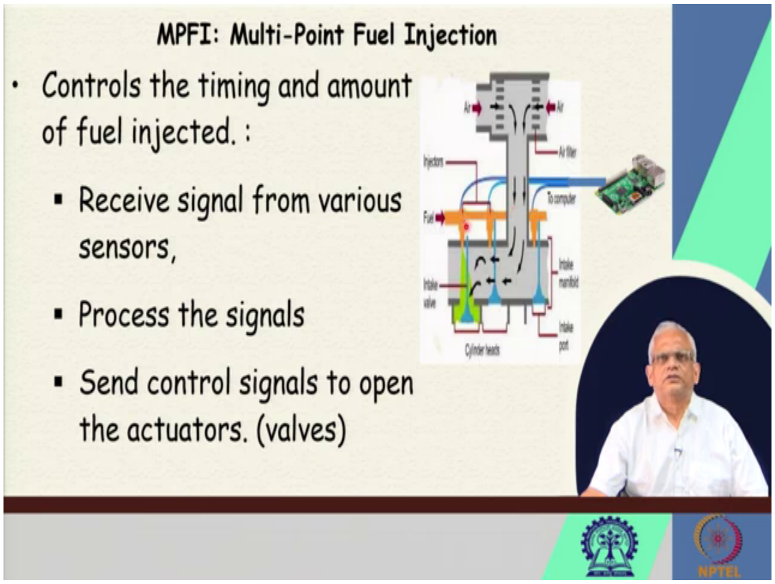
(Refer Slide Time: 21:23)



Now, let us look at an automotive application. Where are the real time systems? We can see here the car is there in the given figure. But we do not see the hundreds of application of embedded systems inside the car. Where are they? At present about 100 embedded processors inside a car. They do engine control, the braking, airbag deployment, windshield wiper, door lock, entertainment systems and so on.

Just look at this example of the BMW car. The applications amount to 2 million lines of code. These embedded real time applications use the Windows CE operating system which is a real time operating system over 60 microprocessors. Many of them are 8 bit. There are several 32 bit and some 16-bit processors which are used for these embedded applications. There are many networks, many of these are real time networks. As part of this course, we will also look at real time communication.

(Refer Slide Time: 22:53)



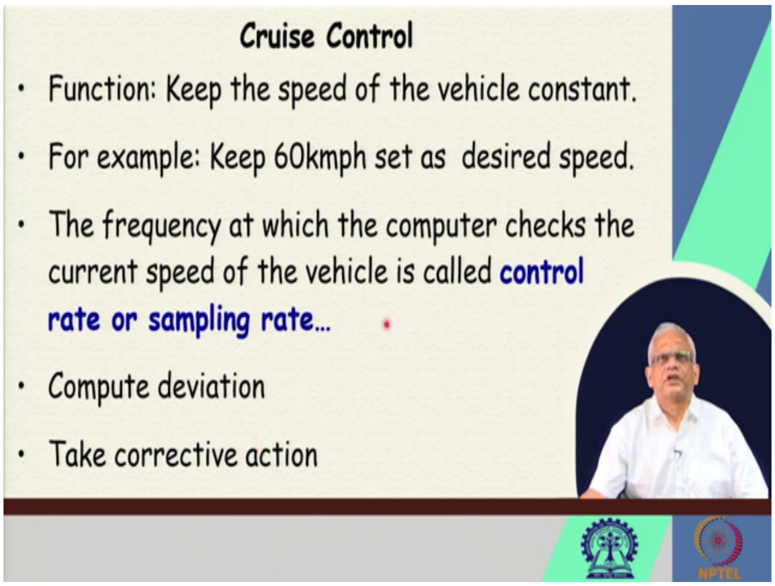
Now, let’s look at one typical example of a real time system is this multi-point fuel injection. As I was saying that this MPFI is now a mandatory requirement for all cars. The conventional mechanical fuel injection carburetor system is inefficient. And therefore, to make the cars efficient, reduce pollution level, the MPFI has become mandatory.

Here the timing of the fuel injection and the amount of fuel injection is monitored and controlled by a computer in MPFI. And here the fuel is injected through multiple fuel inlets and therefore, it’s quite complicated. A carburetor system, a mechanical system of fuel injection will be very difficult to handle multiple fuel injections and the amount of fuel injection. You do not really need to go and tune your system every now and then because it is being controlled by a computer.

So, here the embedded system that is handling the MPFI receive signals from various sensors, processes these signals and then controls the exact amount of fuel to be injected through the valves and the timing of this based on this computer. You can see here this computer. So, here let us look at these fuel inlets here. And all these are being controlled by these computers.

And there is this air inlet through an air filter as shown in the figure above and the air to the intake valves, so they are controlled by the computer and the timing and the amount of fuel is controlled, so that the combustion is complete and the efficiency is maximum, and that is possible because of this computer here which is controlling many parameters, which would be very difficult to control using a mechanical device. It gets all these parameters based on that makes decision which valve for how much time do be opened.

(Refer Slide Time: 26:02)



Another example is a cruise control. Here we keep the vehicle at a constant speed, for example, let's say 60 kilometer per hour. So, here the computer needs to sample the vehicle speed and then if it is different from 60 kilometer per hour, it needs to take some corrective actions. So, the crucial thing here is the control rate or the sampling rate based on which it will try to control and keep it at 60 kilometer per hour. It will compute the deviation and take corrective actions in terms of fuel injection and so on. So, the cruise control system is again another example of an embedded application.

In this introductory lecture, we just looked at what is real time, a real time system and what are some of the applications, because these have large number of applications, just looked at what are the some of the many applications, prominent areas of application and we are just trying to look at one or two applications in some depth. We are almost at the end of this lecture. We will stop here and we will continue in the next lecture. Thank you.