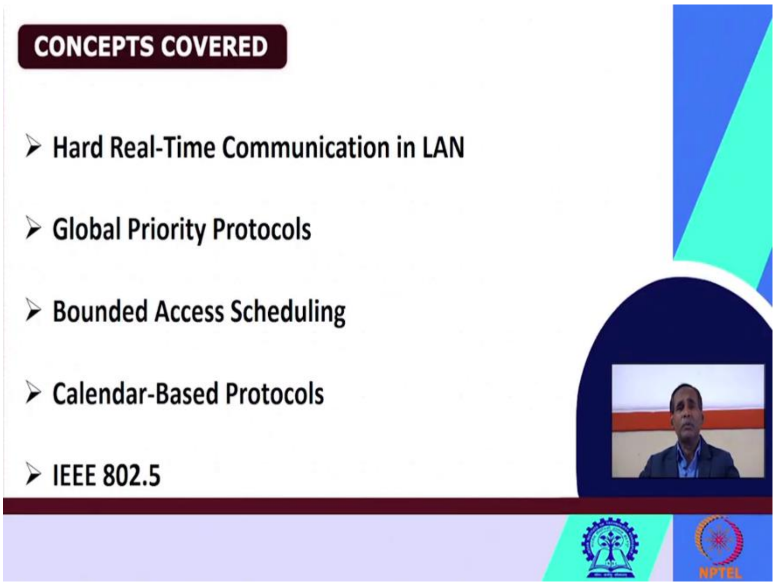
Real Time Systems
Professor Durga Prasad Mohapatra
Department of Computer Science and Engineering
National Institute of Technology Rourkela
Lecture 51
Real - Time Communication in a LAN

Good afternoon to all of you. Now, let us start the next portion, we will go to a next topic, well today we will discuss about the real time communication, how does it occur in LAN.

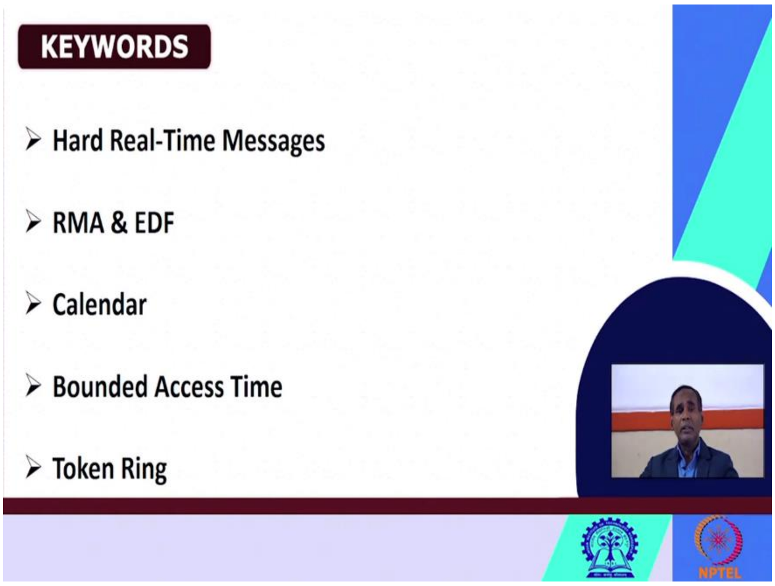
(Refer Slide Time: 00:31)



So basically, I have told you that there are different types of real time communication in LAN such as hard real time communication LAN, soft real time communication LAN or say you can say non real time communication LAN. So today will discuss mainly the hard team communication in LAN. Next class we will discuss about these soft real time communication LAN.

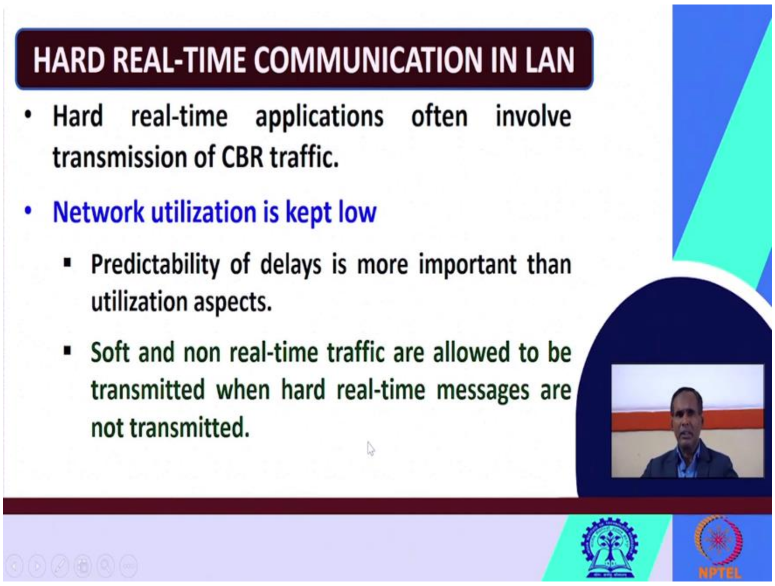
We will discuss some of the protocols which will be used for hard real time communication in LAN, such as global priority protocols, bounded access protocols, calendar-based protocols and then we will also see some particular examples of global priority protocols such as countdown protocol, virtual time protocol, IEEE 802.5, window-based protocols, these are all examples. They are coming under global priority protocols that we will see.

(Refer Slide Time: 01:19)



We will use some keywords like hard real time messages, soft real time messages, non-real time messages, RMA EDF you have already known earlier, calendar, what is the bounded access time, token ring etcetera, these things we will see.

(Refer Slide Time: 01:33)



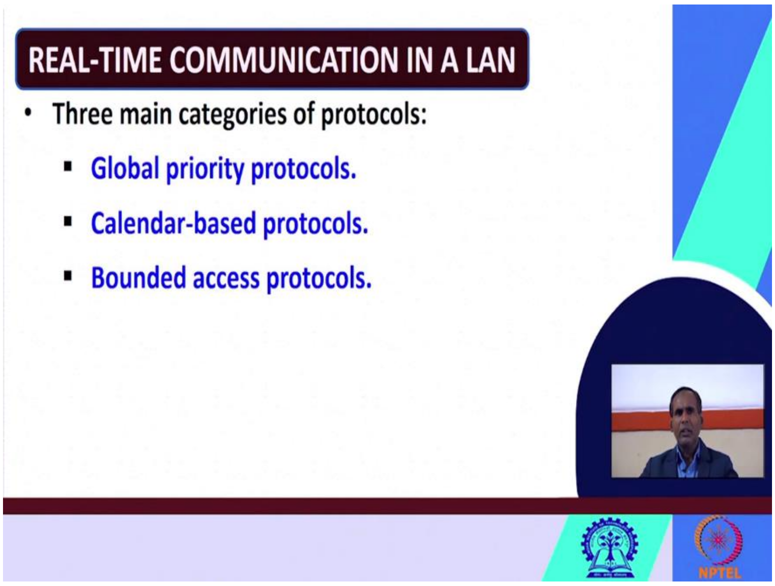
Today we will discuss only about the hard real time communication in LAN and some protocols used for this purpose. Next class we will discuss about the soft real time communication in LAN. Normally, the hard real time applications they often involve transmission of CBR traffic. So, what is CBR traffic, what VBR traffic we have discussed already in the last class. CBR traffic stands for the constant bitrate traffic, is not it?

So, the hard real time applications they often involve what? They often involve transmission of the CBR traffic constant bitrate traffic, here the network utilization is kept low, in case of hard real time communication, the network utilization is kept low.

The predictability of the delays is more important than utilization aspect. Why this network utilization is kept low? Because in hard real time communication more importance is given to the predictability of delays, how the delays can be predictable in a more correct manner. Rather than giving emphasis utilization aspects. So that is why the network utilization is kept low.

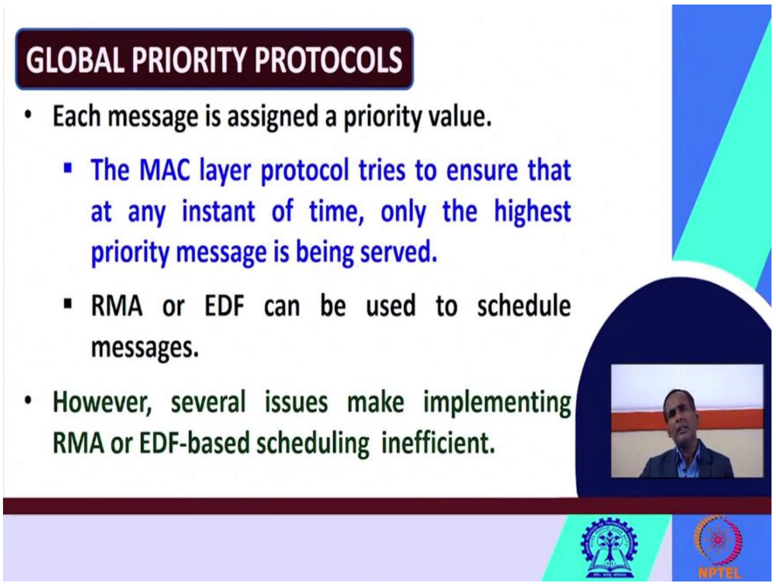
Of course, in hard real time applications, it will normally allow the hard real time traffic or hard real time messages. But besides that, what it also allowed soft and non-real time messages when the soft and non-real time topic they are allowed to be transmitted, when hard real time messages are not transmitted that means there are no hard time messages to be transmitted, at those times soft real time messages and non-real time messages they can be transmitted.

(Refer Slide Time: 03:07)



We will see here three important categories of protocols which will be used for real time communication in a LAN. Those are global priority protocols, calendar-based protocols, bounded access protocols. Let us first see, the preliminary concepts of these three then we will take simple examples from every category.

(Refer Slide Time: 03:27)

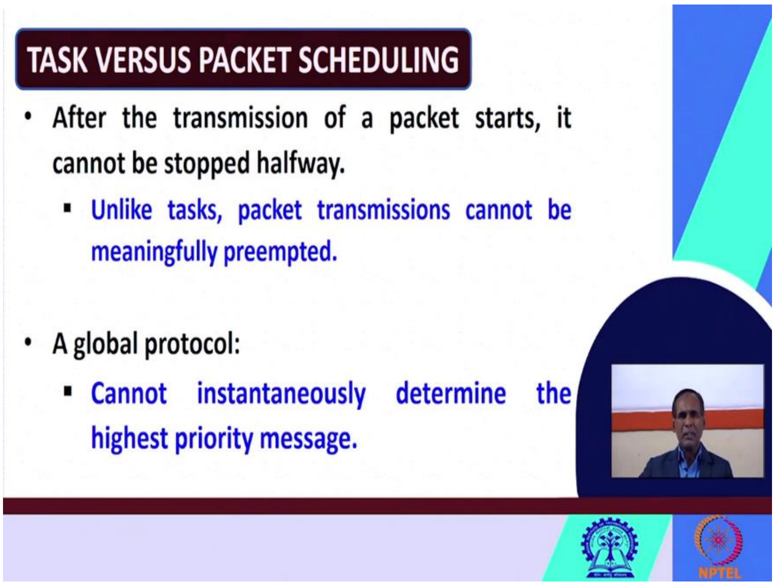


Let us, see first global priority protocols. In global priority protocols, each message is assigned a priority value. In global priority protocol what happens each message is assigned a unique priority value. We have already discussed about MAC layer protocols in the last class. In this case, the MAC layer protocol tries to ensure that at any point of time, at any instant of time only the highest priority message is being served.

So, in global priority protocols are these names suggest every message is assigned with a priority value and the MAC layer protocol, it ensures that at any point of time, only the highest priority message that means the message having the highest priority that is only being served that will be ensured by the MAC layer protocol.

RMA or EDF can be used to schedule the messages. So, in order to schedule the messages, you can either use RMA or EDF. However, what is the difficulty. However, there are several issues which make implementing either the RMA or EDF based scheduling algorithms much more difficult and inefficient. Even if it is mentioned that you can use RMA or EDF to schedule the messages, but there are several issues which will make implementing the RMA or EDF based scheduling much more difficult and it will make them inefficient.

(Refer Slide Time: 04:55)

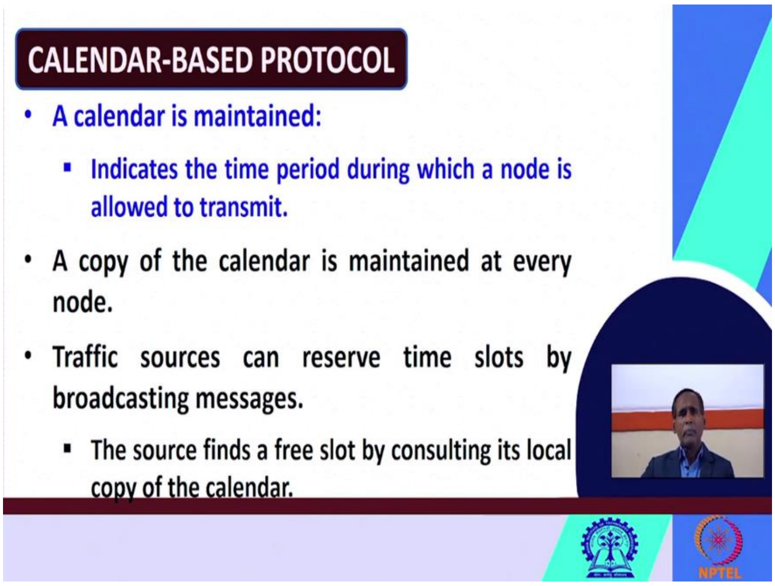


Here, I want to say one more difference like tasks versus packet scheduling. What is the difference? You know, whenever the transmission of a packet occurs, what will happen? After the transmission of a packet starts it cannot be stopped halfway. Once it is started restarted means, once the packet, transmission of a packet starts means it has to be completed it cannot be stopped halfway. Unlike the task packet transmissions cannot be meaningfully pre-empted.

We have already seen in task scheduling when a low priority task is running and the high priority task is arriving, then the low priority tasks can be pre-empted, that we have seen in the about task scheduling, but just unlike the tasks in packet transmissions or the packet transmissions, they cannot be pre-empted meaningfully, please remember that.

In the global priority protocol also the same thing the packets or the packet transmissions cannot be pre-empted. Why? Because in a global priority protocol, it cannot instantaneously determine the highest priority message the in a global priority protocol it cannot instantaneously determine which one is the highest priority message and hence theses packet transmissions they cannot be pre-empted.

(Refer Slide Time: 06:16)

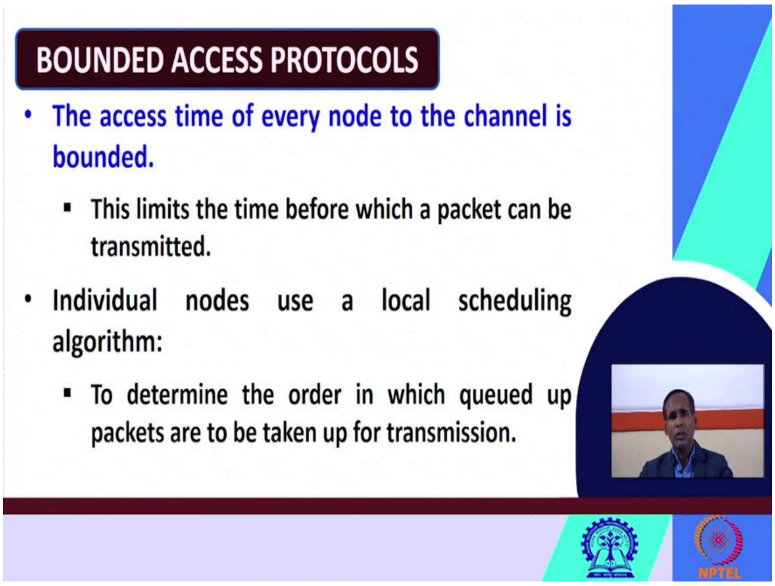


Now, let us see about the calendar-based protocol as its name suggests here a calendar is maintained. Where and why, what does it indicate? In calendar-based protocol a calendar is maintained which indicates that the time period during which a node is allow to transmit. As its name suggests, calendar-based protocol a calendar is used, this calendar indicates the time period during which a node is allowed to transmit the packets.

Then a copy of the calendar is maintained in every node, how many nodes are there? In every node a copy of the calendar maintained, a copy of the calendar is maintained at every node. The traffic sources can reserve time slots by broadcasting messages, the different traffic sources, they can reserve a time slot how by broadcasting the messages.

The source finds a free slot; the traffic source can find your free slot. How? By consulting its local copy of the calendar because I have already told you at every node there is a copy of the calendar is available. So, the traffic source, it can find a free slot by just consulting by just looking at its local copy of the calendar. So, what is the calendar present in this node by looking at that by consulting that local copy of the calendar it can find out free slot.

(Refer Slide Time: 07:38)

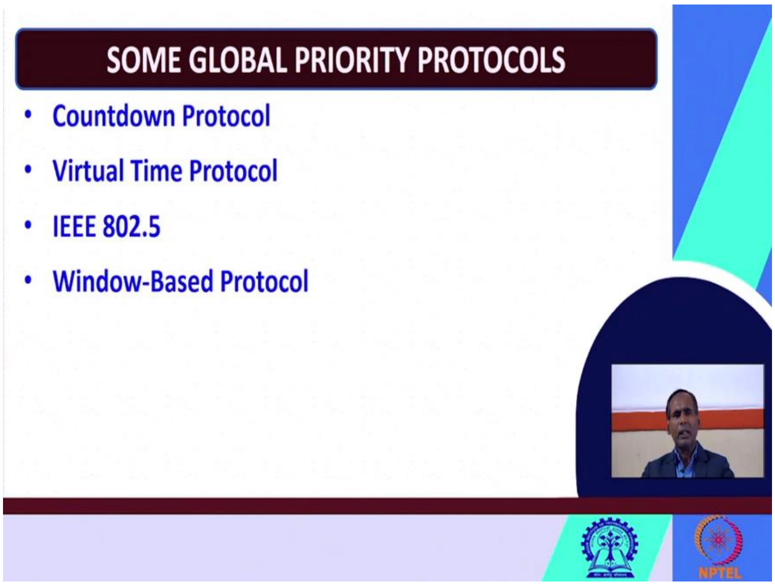


Then the next category of protocol is bounded access protocol. So, in bounded access protocols the access time of every node, access time to what? The access time of every node to the channel that is bounded that is why the name is bounded access protocol. In bounded access protocols, the access time of every node to the channel is bounded by some value. So, this happens in the bounded access protocol. The access time of every node to the channel is bounded.

So, since the access time is bounded, this limits the time before which a packet can be transmitted it imposes limits, it imposes constraints on the time before which a packet can be transmitted before that you cannot transmit the packet. So, in this way, the access time of every node to the channel is bounded and hence it puts limits on the time before which a packet can be transmitted.

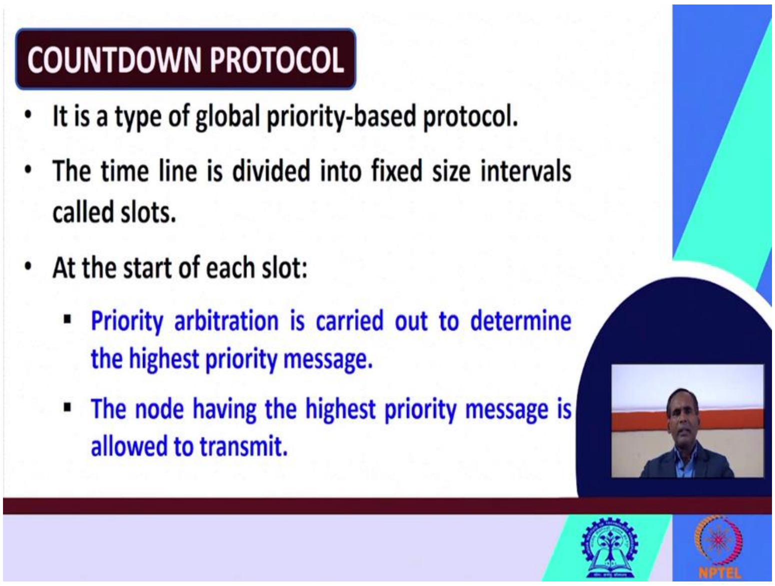
So, the individual nodes they use a local scheduling algorithm. So, every node or the individual nodes the use a local scheduling algorithm to determine the exact order the exact sequence in which the packets which are putting a queue, they have to be taken up for transmission. The individual nodes, they use a local scheduling algorithm through determine the sequence or the order in which the packets which are put in a queue which are kept in a queue they are to be taken up for further transmission. This is what is happened in bounded access protocols.

(Refer Slide Time: 09:08)



Today, we will discuss some of the important global priority protocols. We will start with countdown protocol, then we will discuss virtual time protocol, then we will see an IEEE standard call as IEEE 802.5, then we will see the window-based protocol.

(Refer Slide Time: 09:23)

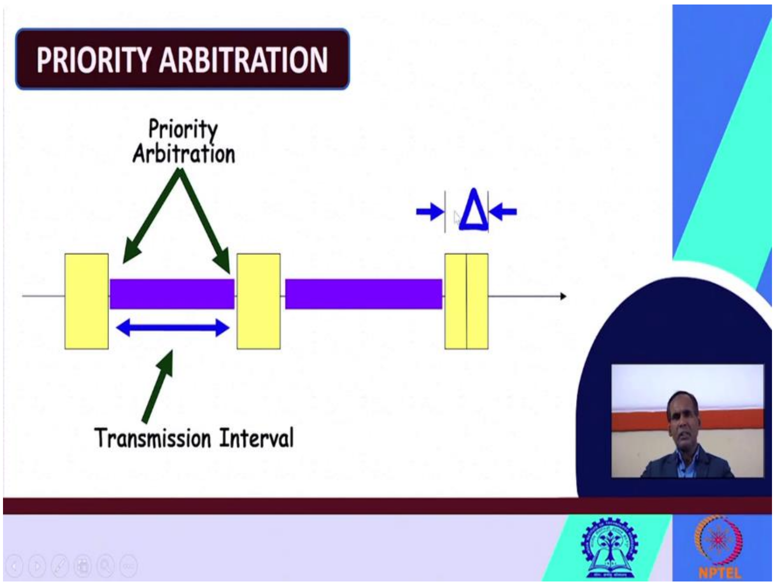


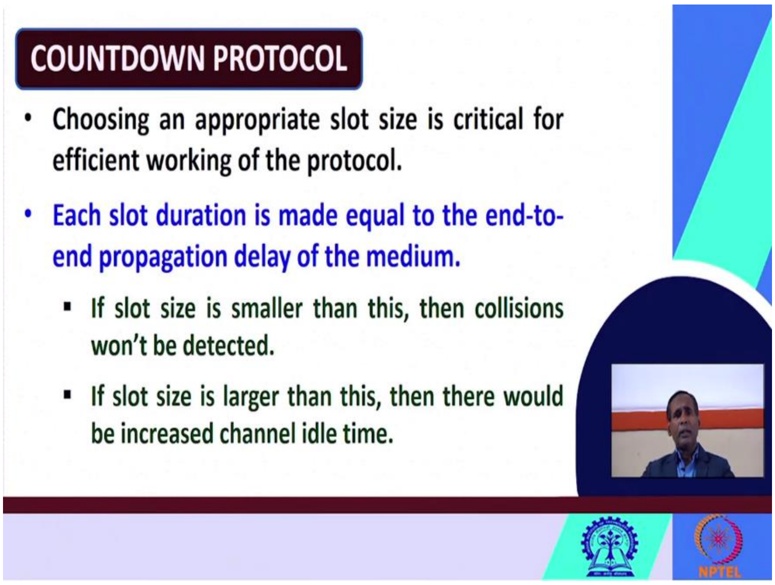
Now, let us start with the countdown protocol. This protocol is a global priority-based protocol. In this protocol, the timeline is divided into fixed size intervals called slots. So, in countdown protocol, the timeline is divided into some fixed size intervals, which are called as slots. Then let us see at the beginning of the start of this slot what happens.

At the start of each slot, priority arbitration is carried out. So, at the beginning of every slot priority arbitration is carried out for what to determine the highest priority message. So, in order to know the highest priority message a process called as priority arbitration process is carried out at the beginning of each slot.

Then once knowing what is the highest priority message which node contents that highest priority message then that node will be allowed to transmit. The node having a node containing the highest priority message is then allowed to transmit the packets or to transmit the message. This is how the countdown protocol starts.

(Refer Slide Time: 10:24)





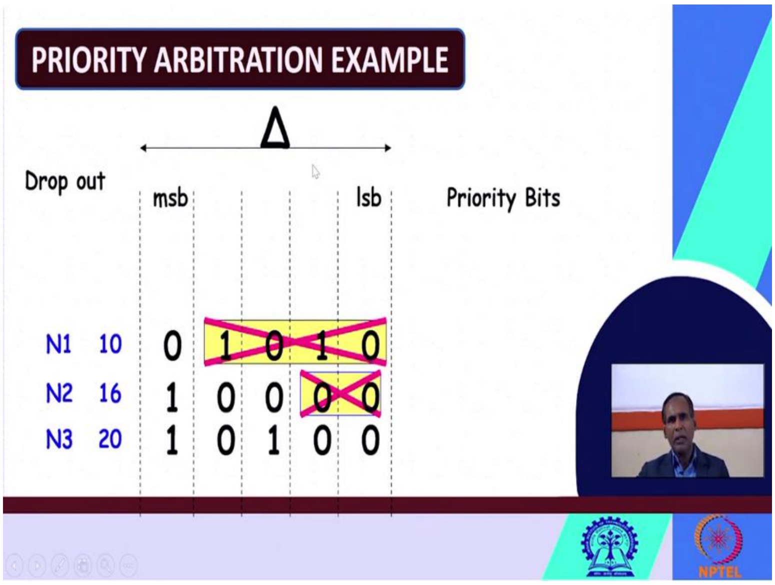
This I have shown in a figure, say these are the slots, this is a slot and this is a slot like this. At the beginning of this slot priority arbitration starts. So, these are two nodes, it will see by using this priority arbitration message you can know you can determine which node is having the highest priority message and that node will be allowed to transmit that message. So, this is called as this is the transmission interval. So, then at the start of every slot this priority arbitration process starts. So, this period we call them as this delta t.

Now, let us see some of the design issues in the countdown protocol choosing an appropriate slot size is very much critical for efficient working of the protocol. So, this is the slot size transmission interval this is the slot size. So, how much should be the slot size. So, choosing an appropriate or choosing a suitable slot size is very much critical, is very much important for efficient working of the protocol. Each slot duration or the size of each slot is made equal to the end-to-end propagation delay of the medium.

Please remember so, the slot each slot duration or the size of the slot is made equal to the end-to-end propagation delay of the medium. Why? because if the slot size will be smaller than this value, that means the end-to-end propagation delay, then what will happen then collisions cannot be detected.

And if this slot size will be larger than this what will happen then there would be increase channel idle time, the channel will remain much time as idle. So, that is why, each slot duration is made equal to the end-to-end propagation delay of the medium. So, this is a very important factor this is a very critical factor for working of the protocol. So, choosing an appropriate slot size is very much critical is very much important for efficient working of this countdown protocol.

(Refer Slide Time: 12:18)



Now, let us explain how does this priority arbitration occurs through an example. So, suppose there are 3 nodes are there N1, N2, N3, N1 is having a message with priority is 10, N2 is having a message with whose priority 16, N3 having a message whose priority is 20 and see those priorities are represented using binary numbers this one is the low least significant bit, this is the most significant bit LSB and MSB.

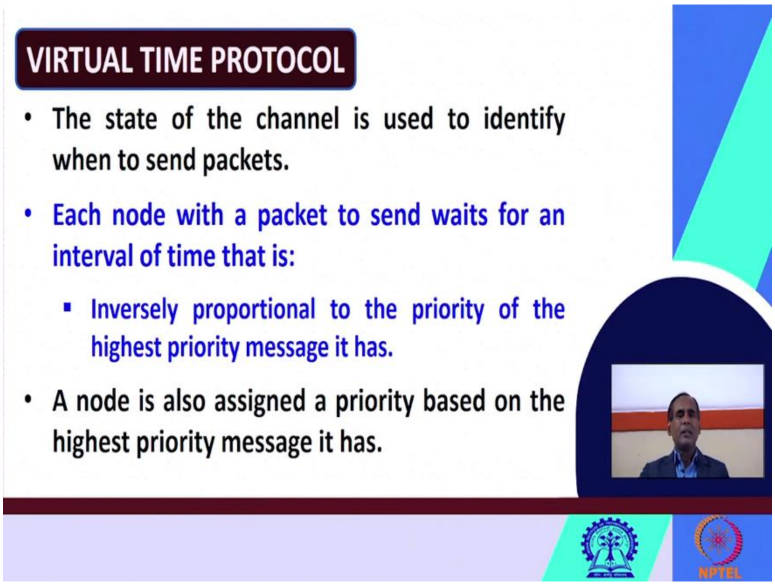
Now, what happens we call this as priority bits. When the system it reads this N1, this node 1 this message is read, what will happen? First bit is 0 will be read. So, when this will read 0 but it will listen 1, is not it? Because N2 is having MSB 1. So, when it will node 1 will read 0 and it will listen to 1 because next N2 is 1. What will happen? Obviously 1 is greater than 0.

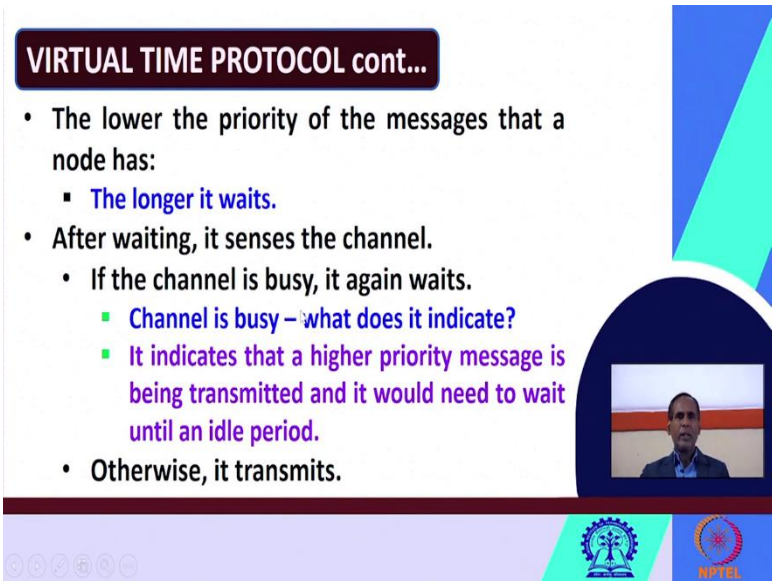
So, node 1 will know that there is another node which is having a message having high priority. So, what it will do automatically, the remaining bits will be dropped out. It will drop out the remaining bits. So, that is why we have made cross. So, now N1 will be dropped out from the contention and now N2 will be there. So, now N2 these values will be read. So, the priority values will be read. So, N2 first it will read 1, then 0, then 0. So, when the third bit is read it is reading third bit 0, but it listens what? It listens 1 because N3 is this third bit you can see it is there is a 1.

So, it will know that N2 will know that there is another node having high priority. Because see 1 is greater than 0. So, N2 will know that there is another node which is has a message having higher priority than this message, then what it will do, it will drop out the remaining bits. So, this 0, 0 will be dropped out and N2 will be dropped down it will be dropped out from the contention. So, only N3 will there.

So, when N3 the bits of the these N3 this message is read you see there is no collision occurs it will come up to the last bit then you see that no collision occurs then this ensures that there are no other nodes having any highest priority message. So, N3 now it will be allowed, it will transmit the message in this way priority arbitration takes place among the nodes during start of this slot, during start of the interval. This is how priority arbitration takes place.

(Refer Slide Time: 15:02)





We will see the next protocol which is coming under this global priority protocol that is virtual time protocol. Here the state of the channel is used to identify when to send packets. So, how to decide when to send the packets for that the state of the channel is used. In virtual time protocol the state of the channel is used to identify to determine when to send packets.

The state of the channel there can be two things either it is busy or it is ideal, is not it. these two states of the channel they are possible. So, each node with a packet to send, it waits for an interval of time. Every node who is having a packet which has to be sent, it will wait for some time. For how much time it will wait? It will wait for a time which is inversely proportional to the priority of the highest priority message it has.

Please see every node which is has a packet to send it has to wait for an interval of time for a period of time and that time is inversely proportional to what it is inversely proportional to the priority of the highest priority message that it has. A node is also assigned a priority based on the highest priority message it has. So, every node in virtual time protocol, it is also assigned a priority based on the highest priority message it has.

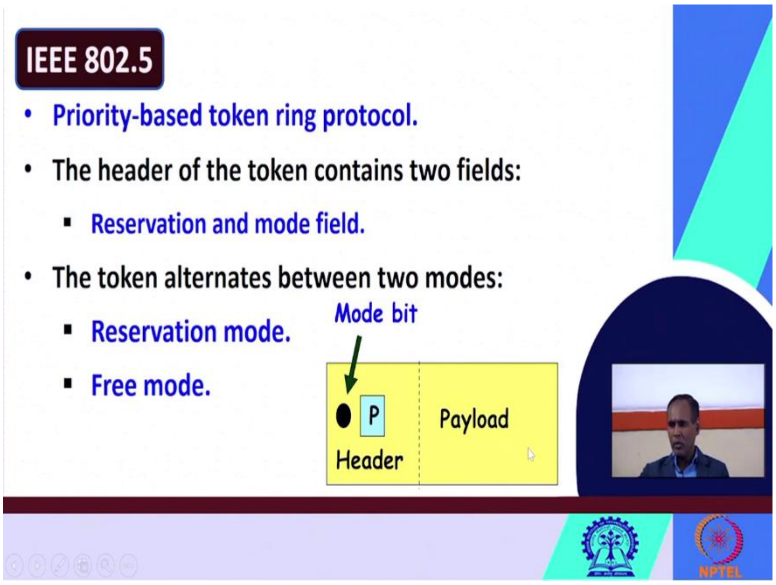
Now, let us see you can see the lower the priority of the message obviously, the node has to wait for a long time, is not it? The lower the priority of the messages that a node has, ultimately the longer time it waits, it has to wait for a longer time. After waiting, so, it has to wait and by what time I have already told you the what, this waiting time is inversely proportional to the priority of the highest priority message it has. After waiting, the node it is senses the channel.

So, I have already told you that here a node has to wait, then after waiting what will happen? After waiting, the node it senses the channel, I have already told you this is based on what the state of the channel by knowing the state of the channel, you can determine when to send packets. The state of the channel can be what two possible states either it is busy or it is ideal.

So, after waiting the node it is senses the channel, the state of the channel. If the channel busy then obviously they are going to wait and otherwise it is free and if it is free it can transmit. Now, what do I mean by channel is busy. If channel is busy, what is it indicate? If the channel is busy, it indicates that a high priority message is being transmitted and it would need to wait until an idle period.

So, if a channel is busy, it means that a higher priority message is being transmitted and it would need to wait until an idle period is there. So, whenever the channel will become idle, then only the node can transmit the message or the data. If the channel is busy then the node has to wait otherwise the channel is idle and it can node can transmit.

(Refer Slide Time: 18:32)

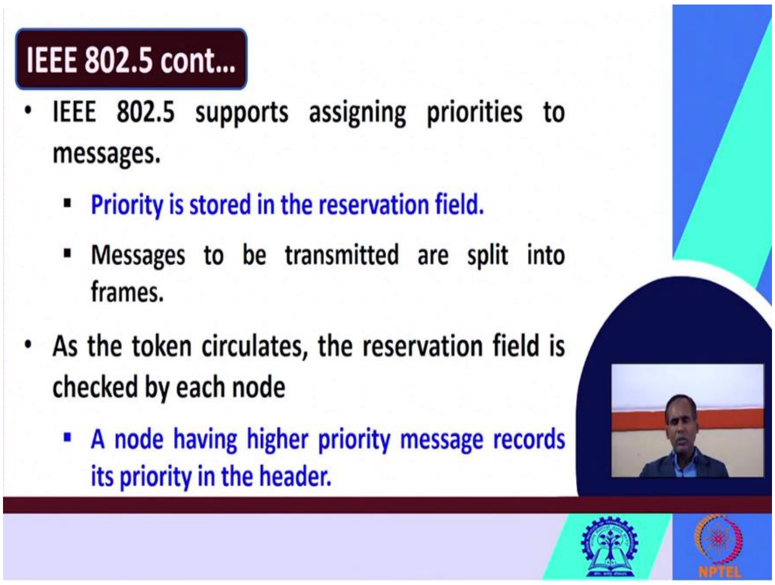


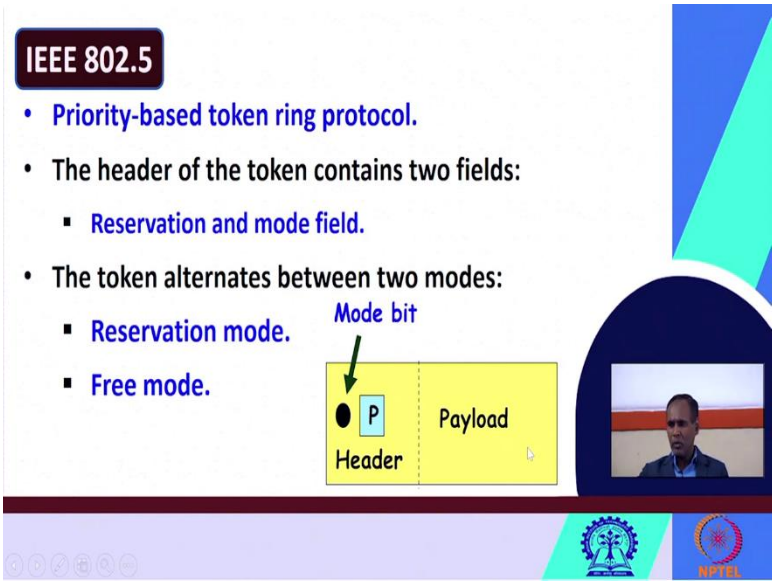
Next protocol is IEEE 802.5. I have already you again, this is a priority-based token ring protocol. Here the header of the token contents two fields. So, in this protocol, it uses a token and this token contains two fields. The header of the token contains two fields. One is the reservation field; another is the mode field. The token, the mode field has two possible values. Let us, see what are the two possible values.

This token actually alternates between two modes one is reservation mode, another is free mode. So, this mode field has two possible values, it can have or it can take the value reservation mode or it can be free mode. The token alternates between what two modes, the reservation mode or the and the free mode. You can see here this is the mode bit here and in this mode bit, I have already told you, there are two bits two fields reservation field and mode field.

So, the header divided into two fields this is the mode fields, this is the reservation field and this token it alternates between two modes that means mode can have two values reserved or free. So, accordingly this mode field this will change and reservation mode this contains a priority, reservation mode this is the, sorry, this reservation field this contains a value called as this priority and beside this the header part the rest of part is the payload.

(Refer Slide Time: 20:02)





So, IEEE 802.5 supports assigning priorities to the messages. So, in IEEE 802.5 it assigns priorities to the messages. And where the priority is stored? The priority is stored in the reservation field I have already told you, this black one is this mode field it can be either in free mode or reserved mode and this field this square is the reservation field reservation this priority it is stored where in the reservation field.

I have already told you in IEEE 802 5 priorities can be assigned to the messages, this priority can be stored in reservation field here the P is the priority it can be stored in the reservation field, the messages to be transmitted are split into frames.

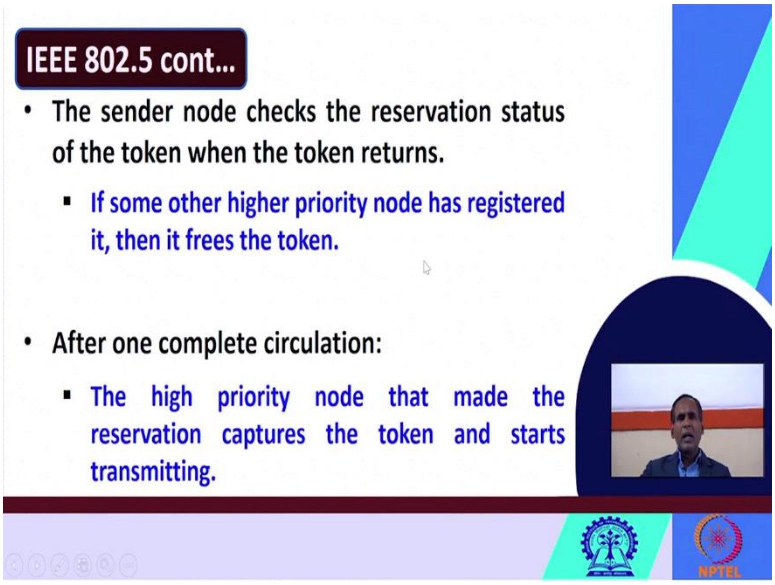
So, what are the messages they have to be transmitted? They are splitted into several frames. Now, as is name being, I have already told you this is a token ring protocol that means a token it will move across the ring. So, as the token circulates, what will happen? So, it will circulate where? It will circulate inside the ring.

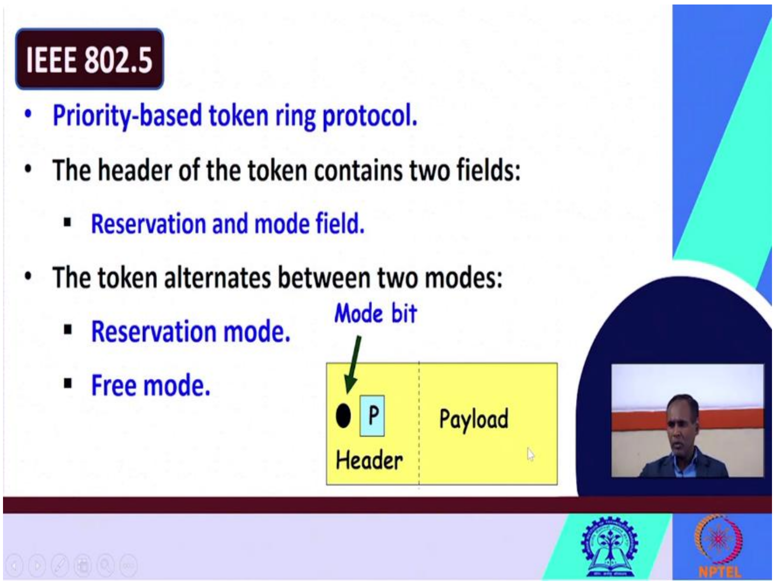
So, as the token circulates inside the ring the reservation field is checked by each node. So, you have already seen. So, this token has a header part in the header there is a mode part either this can be free or this can be reserved. So, when this token it moves along among the nodes in the ring. So, each token so as the token circulates when it is reached at a node the node checks this reservation field or the reservation field is check by each node, every node will check the reservation field of this token.

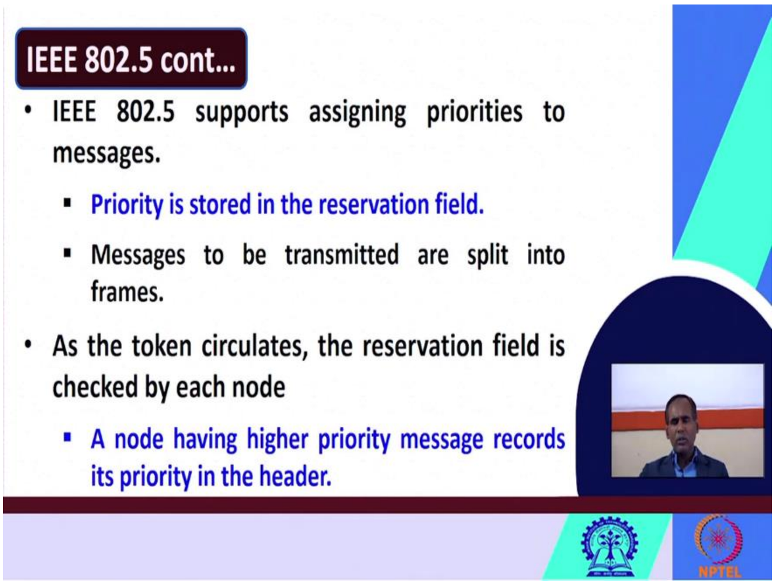
Now, a node having higher priority message records each priority in the header. Please see in the header already one priority is there. Now, if you know when this token is coming to a node, a node first examines what is the priority that is mentioned in this reservation field. If this priority mentioned in the reservation field, it is less than the priority of its message or in other words, if this node has a message which is having higher priority, then this reservation field then this priority. What it will do? It will record its priority in the header.

It will replace this old priority value and then it will put its priority value that means the, because the message it is having it is has more priority it will replace this old priority with the priority of its message because it is higher than this header priority, that is what I am saying. So, as the token circulates among the nodes in the ring, each node checks the reservation field if the node is having a message having higher priority than this reservation field then it will record or it will update this priority in this header.

(Refer Slide Time: 23:00)







Now, the sender node check reservation status of the token, when the token returns now in the token it will move across all the nodes and it will come to the sender. Now, the sender will check what this priority field reservation field. Initially it has sent some value P. Now, if it is replaced by some other node because they are having what highest priority messages, then what will happen? This initial or the sender node it will simply drop out.

Because it will free the token and the what node having the highest priority it will occupy the token and it will transmit. Else if the sender node sees that, this priority value it has already set it is not changed by any node because it is the highest priority message it is holding, then it can transmit this what message and other messages or other nodes they will wait till their turn comes.

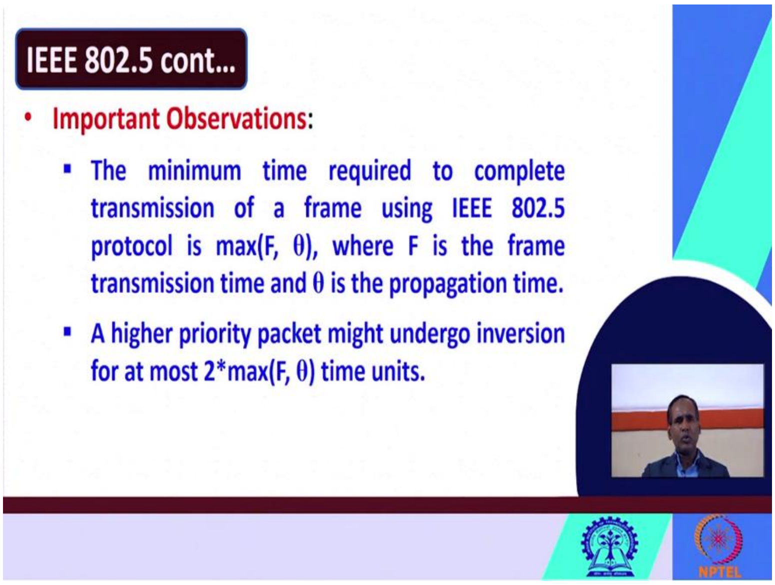
So, the sender node checks the reservation status of the token, when the token returns back to the sender node. If by that time some other priority higher priority node has updated or registered it has changed this priority value field. Then the sender node it will free the token, it will free the token and this token will be now occupied or it will be taken the control will be taken by this node having the higher priority. And now what will happen? After one complete circulation in the ring the high priority node which made the reservation captures the token and starts the transmitting.

I am repeating again, first the sender node, it puts the priority value that means what is the priority value of message it puts it. Then this token it will be circulated among all the nodes present in the ring. Then if a node sees that the message of or the priority of the message it is holding is in larger than this, it will update this value and again that token only circulate.

Then when it will come to this what sender node and it is sees that some other node has replaced the priority has changed the priority because it has having highest priority message. Then the sender node will free this token. Else, if no other node has changed this original priority value that means the sender node has highest priority. So, the sender node will be allowed to send the packet.

Else the sender node will know that, there is some another node which is having a higher priority message. So, it will free the token, then after one complete round of circulation in the highest priority node which has made the reservation which has updated the priority value then it will capture the token and then it can start transmitting the data or the packet in this way IEEE 802.5 it works.

(Refer Slide Time: 26:06)

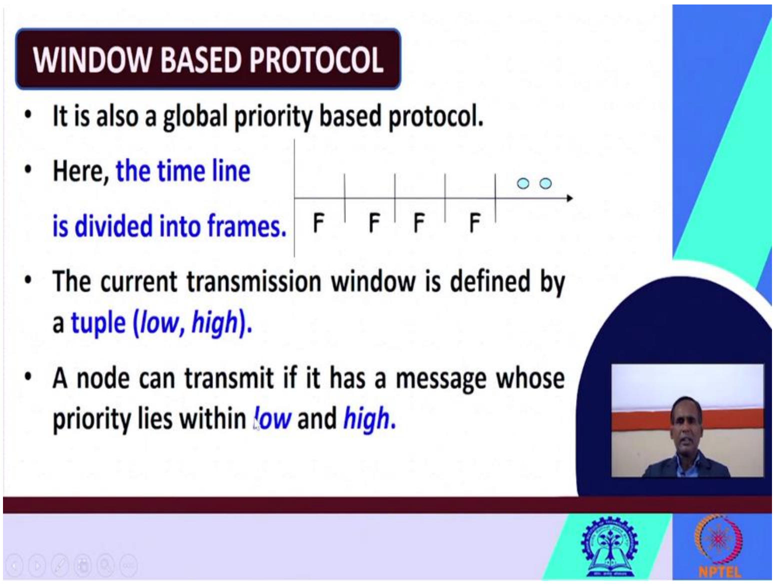


Two important observations we can see in IEEE 802.5 protocol. First the minimum time required to complete the transmission of a frame using this protocol is a maximum of F and theta. Please remember the minimum time required to complete the transmission of a frame using IEEE 802.5 protocol is a maximum of F and theta. Where F is the frame transmission time and theta are the propagation time.

So, this is the which one is maximum either F or theta, where F is the frame transmission time and theta is propagation time, that value will be considered as the minimum time required to complete the transmission of a frame. The proof is there in your book; we will not go through the proof those who are interested they can see the book for this proof. Second observation is that a higher priority packet might undergo inversion for at most 2 into maximum of F theta time units.

I hope you have already known inversion in resource sharing. So, like protocols PIP, PCP etcetera you have known. So, a higher priority packet it might undergo inversion in IEEE 802.5 protocol. For what duration of time? For at most 2 into maximum of F theta time units. Again, the proof is there in your book, we will not discuss, those who are interested they can see the proof from the book.

(Refer Slide Time: 27:29)

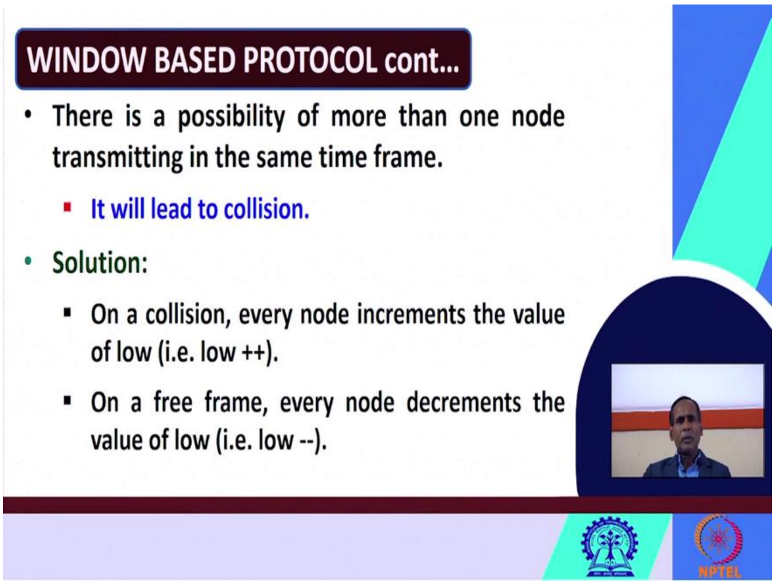


Now, we will see the last protocol that is the window-based protocol. I have already told you it is also a global priority-based protocol. Here the timeline is divided into some frames in window-based protocol as its name suggests window based, here the timeline is divided into some frames. The current transmission window is defined by a tuple low and high. What is the current transmission window? It is defined by a tuple low and high.

So, now let us see how the timeline is divided into some frames. So, this is the timeline you can see it is divided into some frames. The current transmission window it is defined by a tuple that is low and high, low is the minimum value high is the maximum value. A node can transmit if it as a message whose priority lies within low and high.

So, I have already told you a tuple two values low value and high value. Any node or a node can transmit the packets or if can transmit the message if it has a message whose priority is more than the low value and less than this high value that means the priority if it lies, if it lies within low and high then the node can transmit the message.

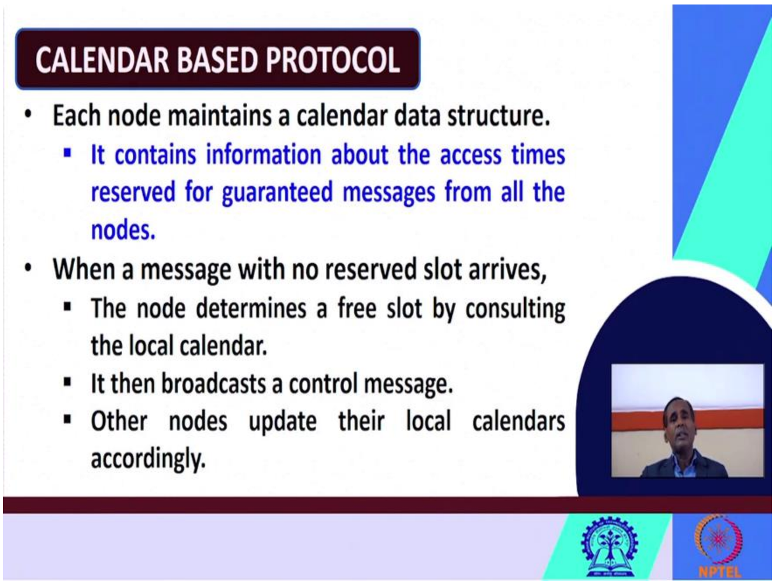
(Refer Slide Time: 28:45)



In window-based protocol there is every possibility of more than one node transmitting the same time frame. It is possible that in the same time frame two nodes they can transmit the messages. Obviously, it will lead to collision, collisions may occur. Then, what is the solution? The solution is as follows on a collision that means when a collision happens when a collision occurs, every node they increment the value of low the value of low is incremented like low will become now low++.

Similarly, when the frame is free, on a free frame every node decrements the value of low that is it becomes now low = low--. So, in this way by following this procedure, these collisions can be avoided. On a collision every node increments the value of low and similarly on a free frame every node decrements the value of low, in this way the collisions can be avoided.

(Refer Slide Time: 29:42)



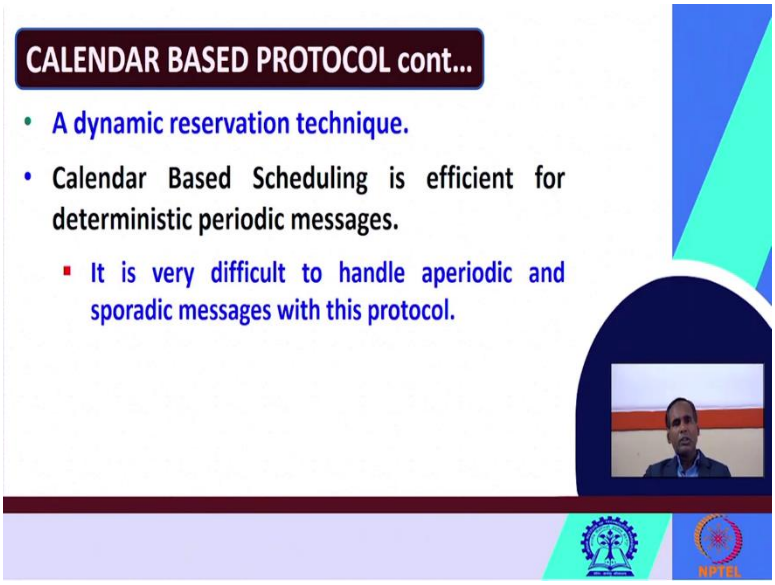
So, we will see the next protocol that is the calendar-based protocol, I have already told you in calendar-based protocol each node maintains a calendar data structure and this calendar contains what? This calendar contains the information about the access times reserved for guaranteed messages from all the nodes.

This calendar, it will contain information about what? About the access times reserved for the guaranteed messages from all the nodes when a message with no reserve slot arrives. So, now, suppose a message is there it is having no reserve slot, that means it has now reserved a slot, how can it can reserve the slot? I have already told you it can reserve a slot by consulting what with its local copy of the calendar.

So, when a message with no reserve slot arrives, what it does? The node it determines a free slot, identifies a free slot. How? By consulting the local calendar or the local calendar present in its. Every message, every node has a local calendar by consulting its local calendar it can determine the free slot. It then broadcasts a control message.

So, after identifying the free slot it the node can broadcast a control message. Other nodes update their local calendars accordingly. So, once the node it gets a free slot, it will start broadcasting the control message, then the other nodes they will update their local calendars accordingly. Because this now calendar at this node which is transmitting the message now it will change. So, the other nodes will update their local calendars accordingly.

(Refer Slide Time: 31:18)



This calendar-based protocol it a dynamic reservation technique. Here you cannot actually determine the free slot in advance. So, looking at the current status, you can see that which slot is free the node can find out a free slot and then it can transmit the message. So, it is a dynamic reservation technique, calendar-based scheduling is very much efficient for deterministic periodic messages.

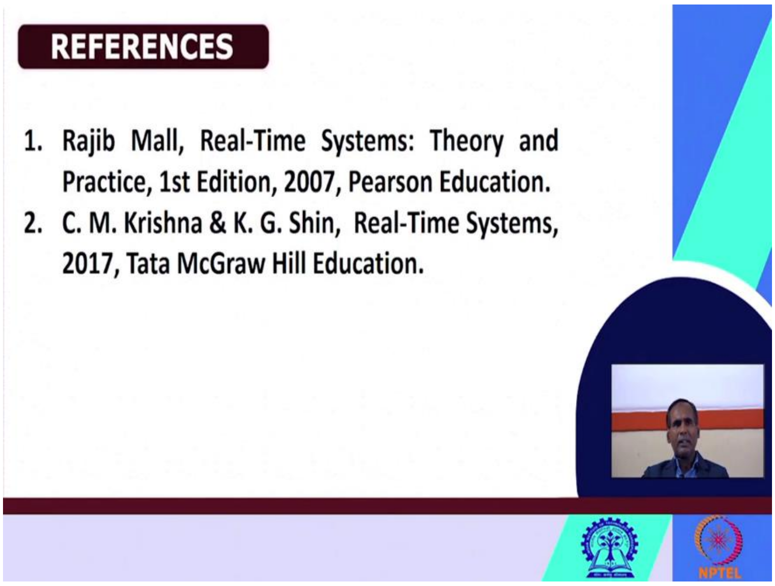
So, this protocol is very much efficient for the deterministic periodic messages, it is very difficult to handle aperiodic and sporadic messages with this protocol. By using this calendar-based scheduling or by using the calendar-based protocol it is very much difficult to handle the aperiodic and the sporadic message. It is very much suitable for handling the deterministic periodic messages only.

(Refer Slide Time: 32:09)

So, today we have discussed about the hard real time communication in LAN. We have seen three important protocols global priority protocols, bounded access protocols and calendar-based protocols. We have seen the functioning of priority-based token ring protocol. We have seen perhaps four important types of global priority protocols such as the countdown protocol, virtual time protocol, IEEE 802.5 protocol and the window-based protocols.

These things we have seen today; we have also seen the fundamentals of calendar-based protocol. In the next class we will see the remaining one that is bounded access protocol. Some examples of bounded access protocols we will discuss in the next class.

(Refer Slide Time: 32:51)



We have taken this from these two books. Thank you very much for your patience while hearing.