Computer Networks
Prof: Sujoy Ghosh
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur
Lecturer Name - 19
Ethernet – CSMA/CD

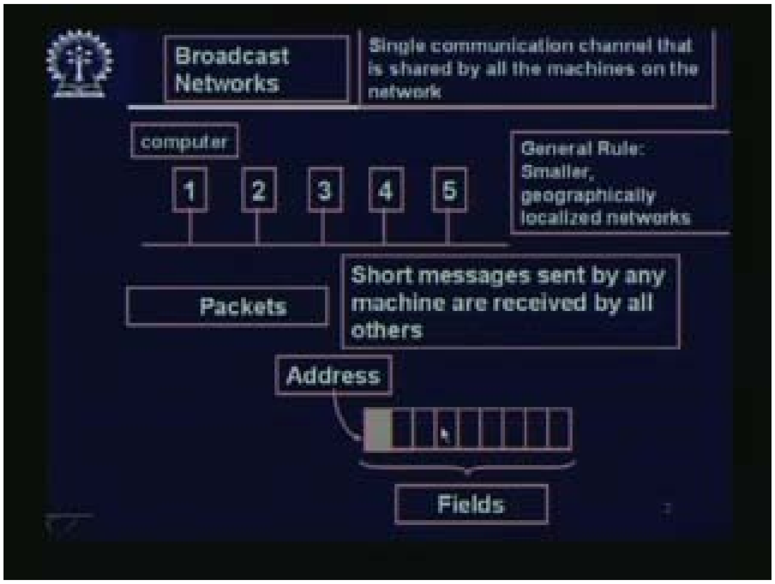
Good day. Today, we will talk about Ethernet. This is possibly the most ubiquitous LAN technology today. As a matter of fact, when computer network developed, there were a number of competing LAN technologies but today Ethernet has come to dominate LAN almost totally, excepting for the newly emerging world spot also, which we will discuss later. Not only LAN, nowadays people are talking about Ethernet in the MAN, i.e., Metropolitan Area Network also. Ethernet as you understand is very important so we will talk about Ethernet.

(Refer Slide Time: 01:37)

Ethernet – CSMA/CD

It is a dominating data link layer technology and the multiple access scheme of Ethernet is CSMA/CD. We will talk about Ethernet and CSMA/CD; this will be our first lecture on this. We will continue on this topic in the next lecture also. What is the broad outline of the Ethernet? First of all, it was conceived as a broadcast network, so single communication channel that is shared by all the machines on the network. This would be the kind of medium we will have. We have a shared medium where everybody broadcasts. It is supposedly one of the common broadcasts in the early days, one of the common physical organizations of Ethernet where some machines were connected to a coaxial cable. This is a LAN technology unlike the satellite technology that we have talked about – remember that in satellite technology, which a shared medium, we of send data is also and then detect collision. This is further advancement.

(Refer Slide Time: 02:00)



Broadcast Networks

Single communication channel that is shared by all the machines on the network

computer

General Rule: Smaller, geographically localized networks

Packets

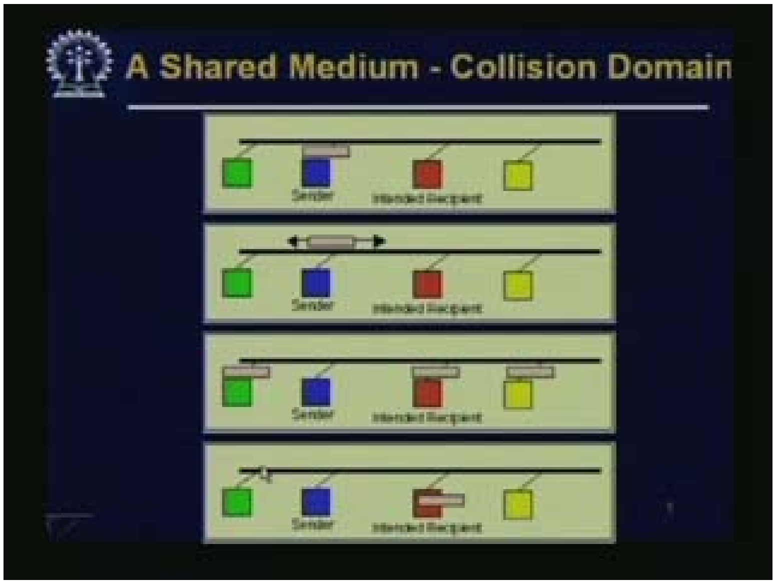
Short messages sent by any machine are received by all others

Address

Fields

The multiple access technique of Ethernet is just some refinement over that but this is essential in a LAN environment, meaning it is for a small geographical area. We have packets; that means packets are sent over this, we will see later on how it frames them. So short messages sent by any machine are received by all others. Since this is a broadcast network, a message which is sent by any machine is received by all networks and so naturally you have to give the destination address. There are various fields in the Ethernet header and we will look at that later on. One of them has to be the destination address so that we understand because otherwise the recipients will not know for whom this frame is meant.

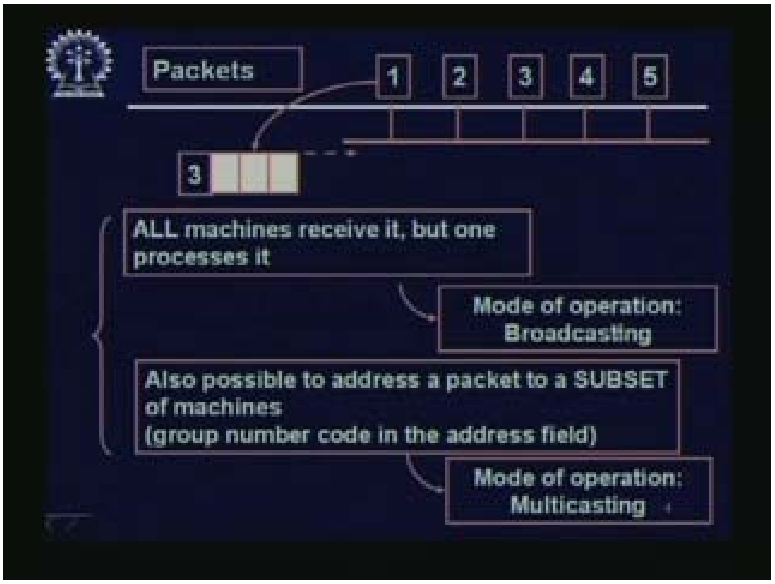
(Refer Slide Time: 04:05)



A Shared Medium - Collision Domain

Since it is a shared medium, all these machines are supposed to be in the so-called same collision domain; because if two machines are sending simultaneously, their packets will collide. Suppose you have the sender over here and sender has a packet to send, it puts it on this common bus. The packet travels both ways and then it is received by all the recipients. All of them copy the frame into their NIC. They would compare the destination address with their own address, if it is not for them, they will simply ignore it. If it is meant for this node, it will absorb it and then send it to the higher layer.

(Refer Slide Time: 04:58)



Packets

ALL machines receive it, but one processes it

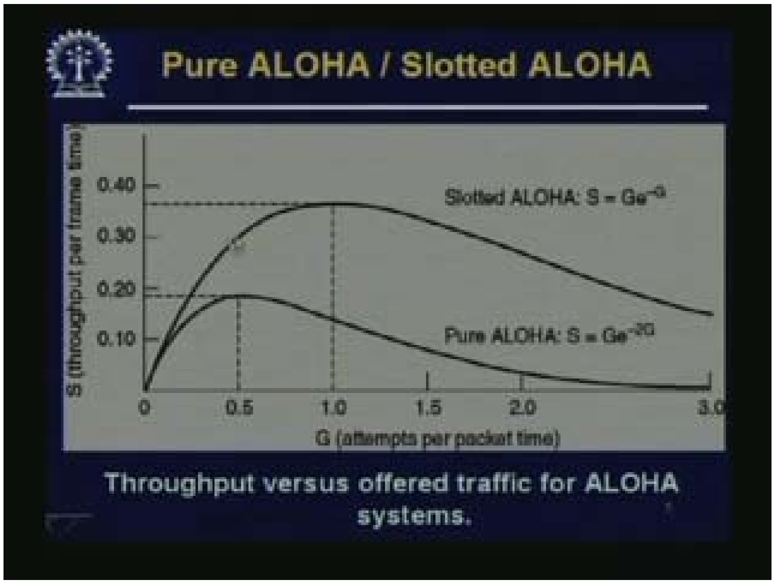
Mode of operation: Broadcasting

Also possible to address a packet to a SUBSET of machines
(group number code in the address field)

Mode of operation: Multicasting

All machines receive the packets but only one would process it; i.e. the mode of operation is broadcasting. It is also possible to address a packet to a subset of machines; this is called multicasting. This mode of operation is called multicasting, meaning we have two ends of the spectrum: one is a point-to-point communication, one sender and one receiver. On the other end of spectrum is a broadcast where one sender and everybody else is a receiver. Somewhere in between is a multicast; meaning there is one sender and several receivers. In order that several receivers can receive it, we have to do something with the address. So we have to give a special kind of address; that is also possible. It is possible to multicast since it is possible to broadcast.

(Refer Slide Time: 05:59)

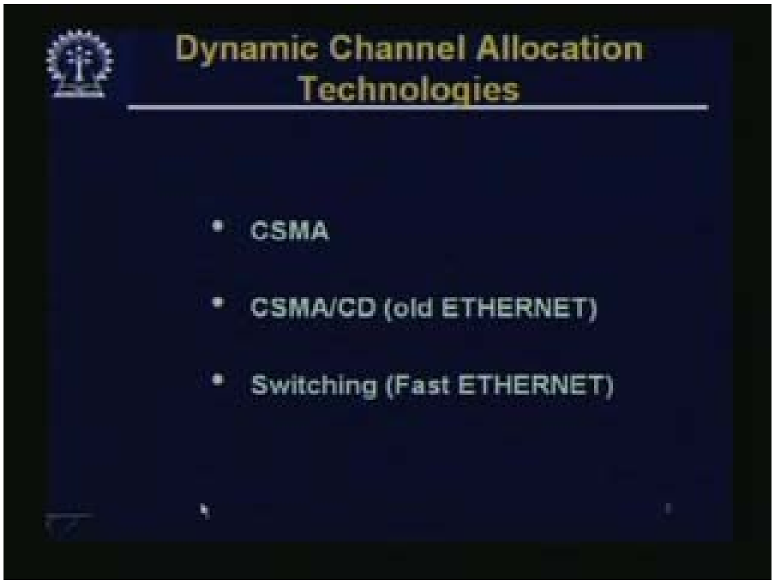


Pure ALOHA / Slotted ALOHA

Throughput versus offered traffic for ALOHA systems.

Just to remind you about our discussion on satellite communication, if you remember, the satellite communication used a MAC protocol called ALOHA. There were two versions of ALOHA and one was the pure ALOHA, whereby if anybody has any data to send he simply sends it. If there is a collision, he would back off and maybe send it again. That is a pure ALOHA scheme and we saw that this is the form of S is equal to G_e^{-2G} where s is the throughput, that means how much you are actually able to communicate successfully, and G is the number of attempts per packet time. This is the s versus gee of pure ALOHA, which gives about 18% of efficiency at the maximum. Slotted ALOHA is where you can send only from the beginning of a slot time; that has considerably better performance then it comes to about 37%. The formula is S is equal to Ge^{-s}.

(Refer Slide Time: 07:23)



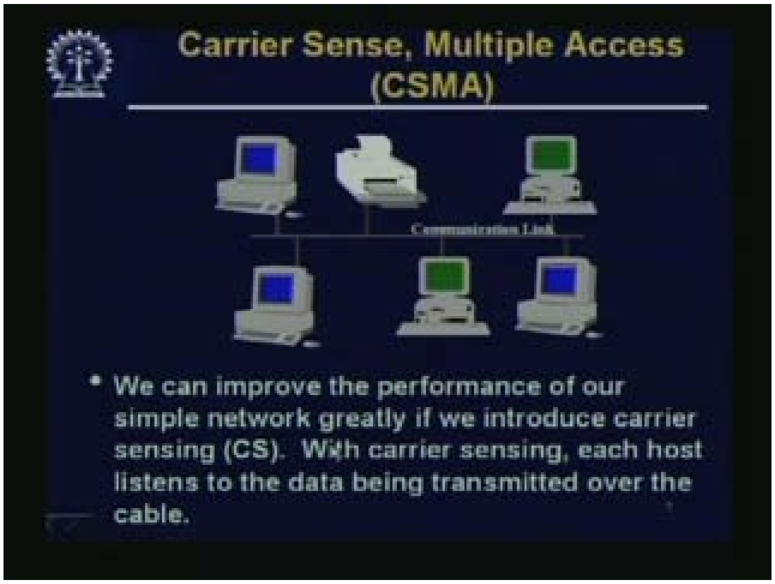
Dynamic Channel Allocation Technologies

- CSMA
- CSMA/CD (old ETHERNET)
- Switching (Fast ETHERNET)

Ethernet is something similar. That means we broadcast and then we detect collision, so it should be the same as some kind of ALOHA. The only thing is that since this is a local area network, a small network, what we can do is that we can sense the carrier; that means we can sense whether somebody is sending. If somebody is sending, we refrain from sending anything; so that is called carrier sensing. We discussed this earlier that it is not possible to do carrier sensing in satellites because the space delay is so high, whatever you are listening to now must have happened may be 250 milliseconds earlier; that is a long time back. But that is not the case in a small LAN. The timeframes are much smaller, so you can actually listen to the medium and find out whether somebody is actually broadcasting something at that particular point of time.

In that case, even if you have something to send, you do not do so. That reduces the collision considerably and that naturally leads to an improvement in the efficiency. So this stream is called carrier sense multiple access and we find that even if you are doing carrier sensing, there can be some collisions although there will be reduced number of collisions compared to pure ALOHA, slotted ALOHA. We have to detect the collision and this CD part. The full MAC protocol that is used by Ethernet is the CSMA CD; this is the dynamic channel allocation technology or the random access MAC for this, used in old Ethernet. Nowadays we have moved to fast Ethernet, where we have reduced collision and switched Ethernet, which we will discuss these in the next lecture.

(Refer Slide Time: 09:40)

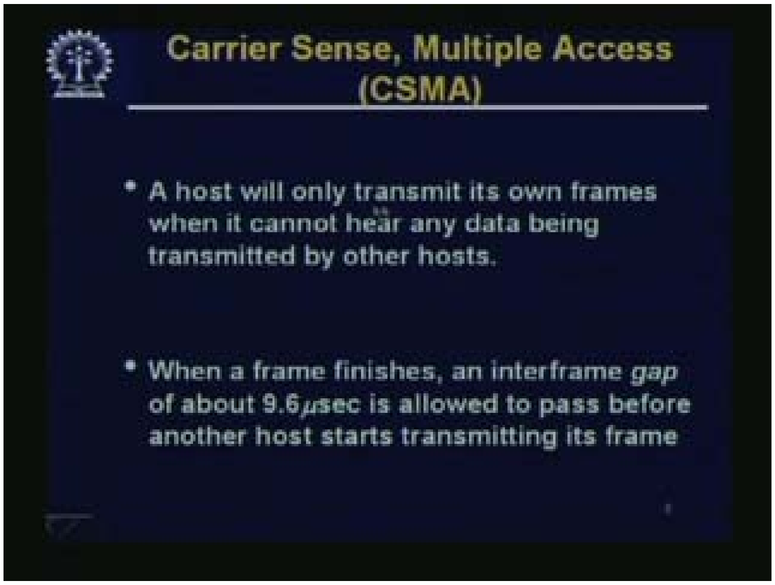


Carrier Sense, Multiple Access (CSMA)

- We can improve the performance of our simple network greatly if we introduce carrier sensing (CS). With carrier sensing, each host listens to the data being transmitted over the cable.

So we have carrier sense multiple access. We improve the performance of our simple network greatly if we introduce carrier sensing; with carrier sensing each hosts listens to the data being transmitted over the cable.

(Refer Slide Time: 09:56)

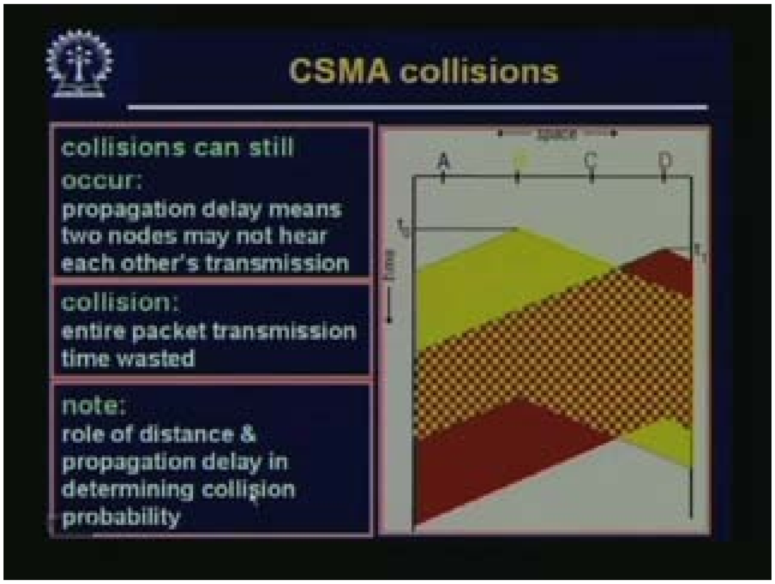


Carrier Sense, Multiple Access (CSMA)

- A host will only transmit its own frames when it cannot hear any data being transmitted by other hosts.
- When a frame finishes, an interframe *gap* of about 9.6μsec is allowed to pass before another host starts transmitting its frame

A host will only transmit its own frames when it cannot hear any data being transmitted by the other hosts. When a frame finishes an inter frame gap of about 9.6 µs is allowed to pass before another host starts transmitting its frame. So he listens to somebody sending and then he naturally waits and then after that frame is over, he gives a gap of about 9.6 µs and then starts transmitting Collisions can still occur mainly because of the propagation delay.

(Refer Slide Time: 10:27)



CSMA collisions

collisions can still occur:
propagation delay means two nodes may not hear each other's transmission

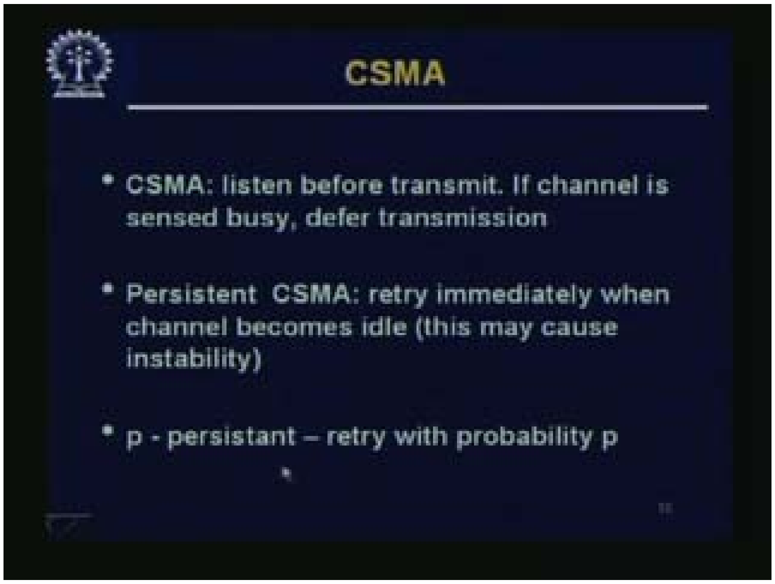
collision:
entire packet transmission time wasted

note:
role of distance & propagation delay in determining collision probability

Propagation delay means two nodes may not hear each other's transmission. As before, if there is a collision, the entire packet transmission time is wasted; and naturally the distance and propagation delay has a strong role to play in the collision probability. Let us look at this. Suppose this node B starts transmission at time t_0. Naturally, the transmission goes both towards A as well as towards C and D. So there is a cone like this – if you plot it against time and distance – over which B's transmission is going. What happens is that C was listening and at this particular point of time, t_0 or even later than that, C did not have any carrier to sense, because B's transmission, although it had started in terms of absolute time, had still not reached C.

So it will reach C only at this particular point of time. During this period, the transmission from B will reach D at a much later time; let us say t_2 or something. If D starts its transmission t_1, what will happen is that somewhere in-between they are going to collide; and these garbled bits will now start propagating on both sides, which is shown in this hashed figure. What happens is that, naturally B's transmission is lost and D's transmission is also lost. So there will be a finite probability of collision and this probability will depend on the time it takes for the frame to reach all the nodes. Because if it had reached some other node, that particular node would not start transmitting; but it takes a finite amount of time, and within that finite amount of time, if somebody else starts transmitting, we have a collision. So we can still have collision; that is why CSMA is not enough; we have to do CD also.

(Refer Slide Time: 13:04)



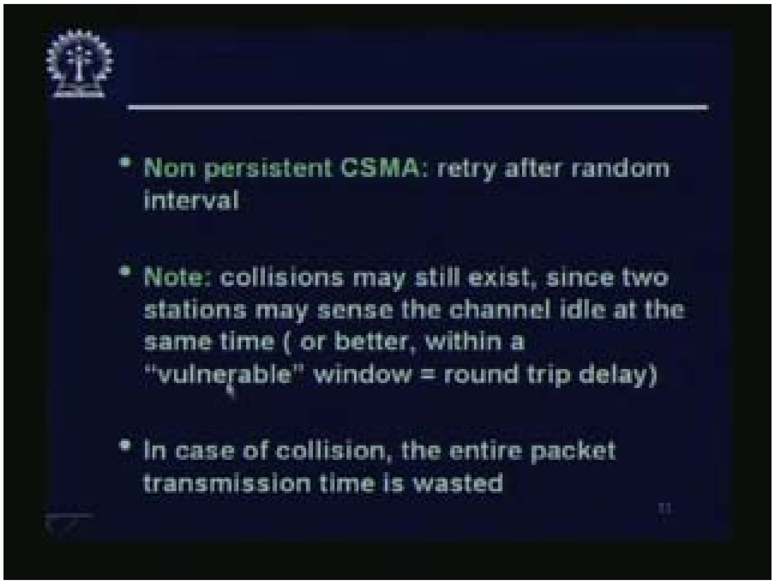
CSMA

- CSMA: listen before transmit. If channel is sensed busy, defer transmission
- Persistent CSMA: retry immediately when channel becomes idle (this may cause instability)
- p - persistant – retry with probability p

So CSMA, as we mentioned, means listen before transmit. If a channel is sensed busy, defer transmission; and persistent CSMA means retry immediately when channel becomes idle – this may cause instability. This is called persistent CSMA. That means retry immediately when channel becomes idle; because what might happen is that during the time when the frame is being sent – that may be a considerable amount of time – more than one node may become ready to transmit. If more than one node is ready to transmit, naturally all of them will pounce on it after giving a 9.6-μs gap and start transmitting, and naturally they are going to collide.

If they collide they know that is bad, so they allow that collision to subside and then they start again. By this time, quite a lot of time has elapsed. Somebody else may become ready and somebody else may again start persisting; so this may grow to some kind of instability. Being totally persistent is not very good, as we will see the performance of each of this; but that is persistent CSMA. We call a strategy as pure persistent; that means, we retry in the next immediate available time with the probability of p, which means, with the probability of 1 − p, we do not persist. That means, we defer still more.

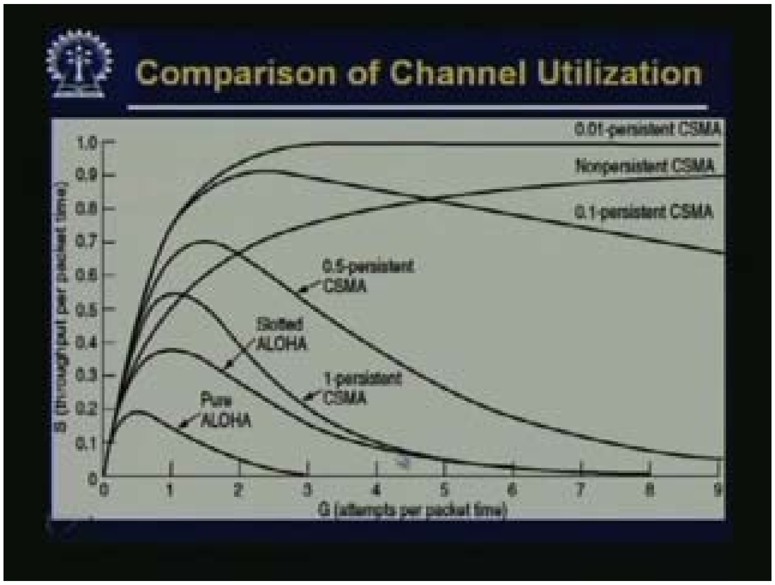
(Refer Slide Time: 14:54)



- Non persistent CSMA: retry after random interval
- Note: collisions may still exist, since two stations may sense the channel idle at the same time (or better, within a "vulnerable" window = round trip delay)
- In case of collision, the entire packet transmission time is wasted

There is a non-persistent CSMA; that is, somebody is ready to send something; finds that it is busy. Then he will retry only after the random interval. He generates some random numbers and decides by himself to wait for this random amount of time. Collisions may still exist, since two stations may sense the channel idle at the same time or within a vulnerable window, which is equal to the round-trip delay. In case of collision, the entire packet transmission time is wasted; so we have pure ALOHA. This is the group of protocols: we have pure ALOHA, then we have slotted ALOHA, persistent CSMA, p persistent CSMA, non-persistent CSMA; and here is how they behave.

(Refer Slide Time: 15:53)

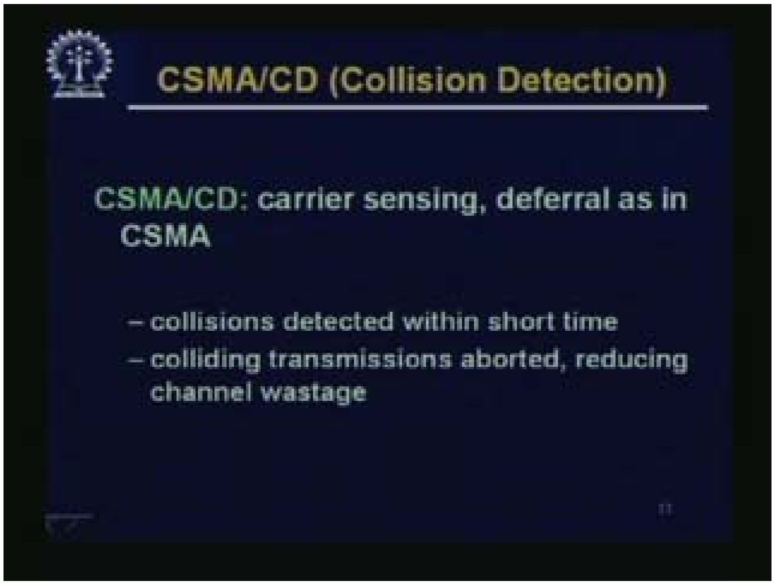


Comparison of Channel Utilization

We have already seen these two curves, pure ALOHA and slotted ALOHA. They are used in satellite communication because we cannot do area sensing over there because of the significant space delay. If you do carrier sensing and if it is persistent, the performance improves somewhat; so it may go slightly above 50% and then comes down. You can have different values of p, remember p is the probability that the node will try immediately when it senses that the channel is available; that probability will be 0.5. At probability 0.5, it will not try at all. At 0.5, persistent CSMA is good, 0.1 persistent CSMA is even better; that means what it will do is with 0.1 probabilities it will try and with 0.9 probabilities it will differ.

If it is non-persistent, this is the 0.01 persistent CSMA and non-persistent CSMA. Non-persistent means it will always go back. If you draw the throughput versus load curve, this is the kind of curve you get. But the exact throughput will depend on how many nodes are there and what their distances are, etc. These are somewhat idealized figures but you get something like this. As you can see, with non-persistent or very low persistent CSMA, we get very high throughput even when the load is going up. So when the load is going up, it is not falling down steeply like ALOHA or slotted ALOHA. This is the channel utilization by the various MAC protocols.

(Refer Slide Time: 17:55)



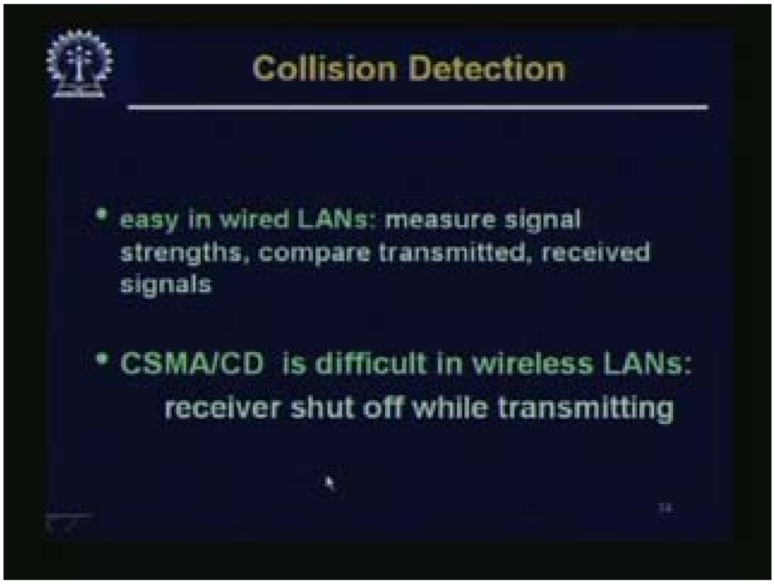
CSMA/CD (Collision Detection)

CSMA/CD: carrier sensing, deferral as in CSMA

- collisions detected within short time
- colliding transmissions aborted, reducing channel wastage

So CSMA/CD is carrier sensing, deferral as in CSMA; collision is detected within a short time; Colliding transmissions are aborted, reducing channel wastage.

(Refer Slide Time: 18:09)

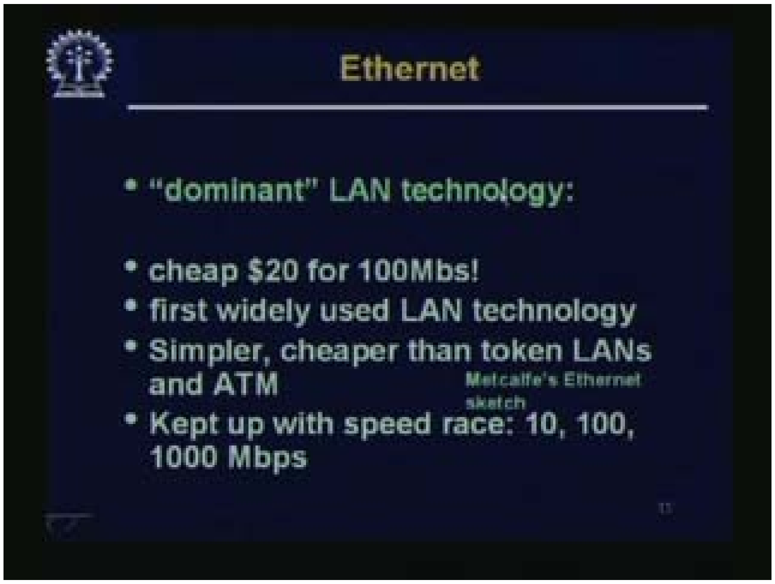


Collision Detection

- easy in wired LANs: measure signal strengths, compare transmitted, received signals
- CSMA/CD is difficult in wireless LANs: receiver shut off while transmitting

Collision detection is easy in wired LAN; it measures signal strengths and compares transmitted and received signals. This is how you detect a collision. It is somewhat more complicated in wireless LANs. We will discuss wireless LANs later on as we move on, but the receiver is usually shut off while transmitting. This is a problem with the wireless LANs, where collision detection is difficult. We go for some other kind of scheme over there, which is called collision avoidance, we will discuss it later.

(Refer Slide Time: 18:54)



Ethernet

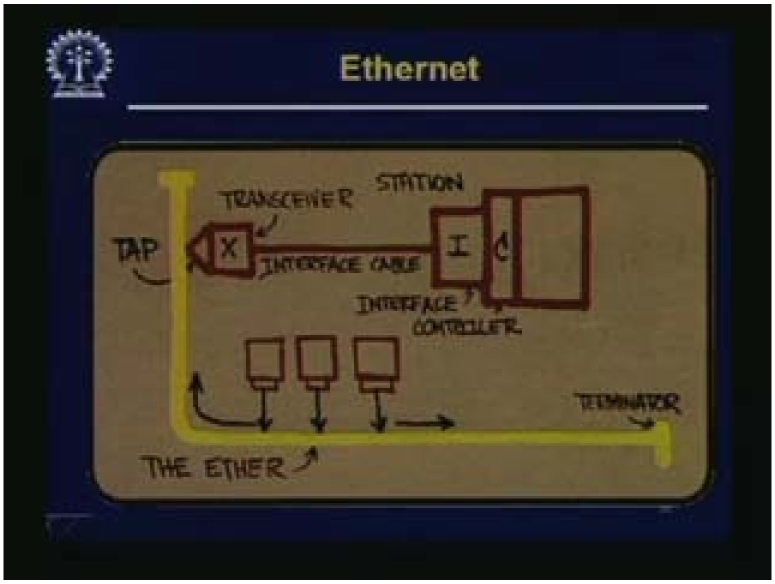
- “dominant” LAN technology:
- cheap $20 for 100Mbs!
- first widely used LAN technology
- Simpler, cheaper than token LANs and ATM
- Kept up with speed race: 10, 100, 1000 Mbps

Metcalfe's Ethernet sketch

As I mentioned, Ethernet is the dominant LAN technology, and since this is dominant, people who followed the Ethernet line had the great advantage of having a huge market.

That really drove down the cost and today Ethernet is cheap. Ethernet network interface card costs in the order of some $20 or something of that order. For a 100 Mbps card, which is quite fast, it is very cheap. It is the first widely used LAN technology; it is simpler and cheaper than token LANs and ATM – so that is a great advantage and it kept up with speed rates of 10,100, or 1000 Mbps. Ethernet started with 10 Mbps, then went on to 100 Mbps, which is a standard today. People are talking about 1000 Mbps or 1 Giga bit Ethernet, which is also making inroads into the backbones of LANs etc.

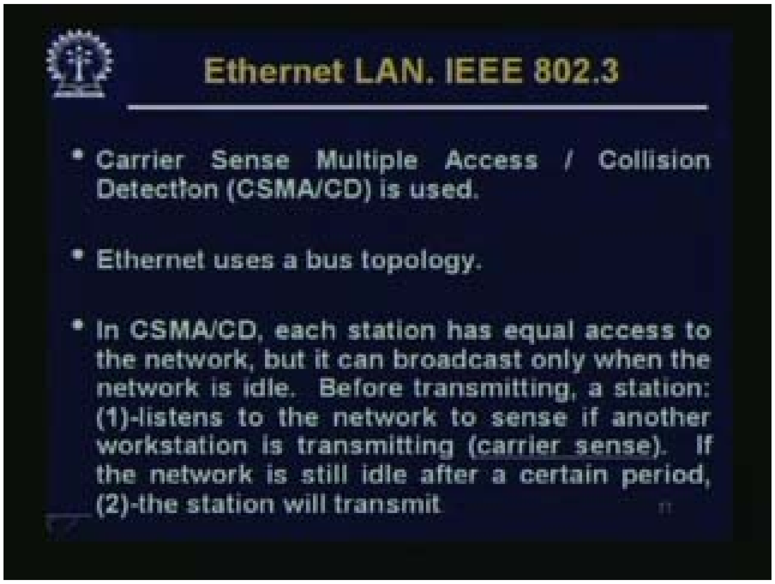
(Refer Slide Time: 20:26)



Ethernet

Maybe in a couple of years' time, it will go into the desktop also, if it has not already reached some desktop. So Ethernet is a very widely used technology. This is a diagram to show you how the original Ethernet looks. You have a cable, which would be terminated on two sides; it would be tapped and then a transceiver will tap into it, to connect it. There will be an interface cable from the transceiver to the interface of the controller. This is basically the NIC, and this is how it will be connected to various machines, which should be tapped into the cable at various points of time, so this is the so-called Ethernet.

(Refer Slide Time: 20:55)

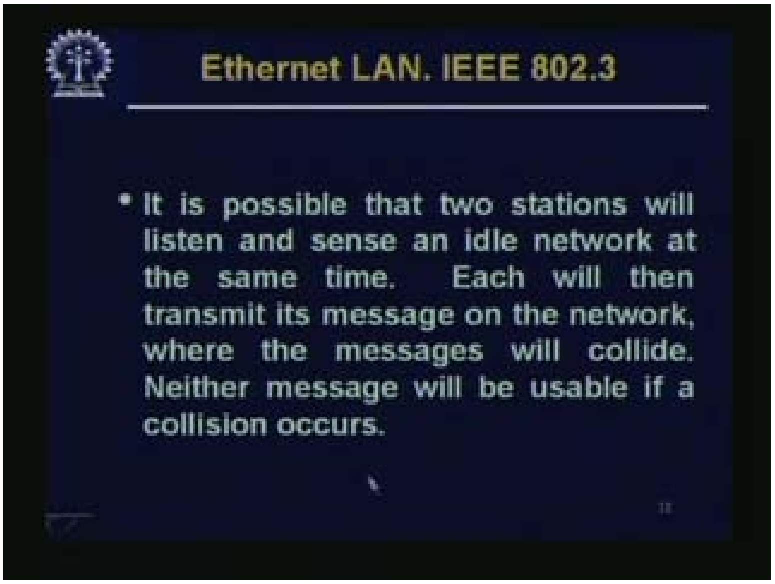


Ethernet LAN. IEEE 802.3

- Carrier Sense Multiple Access / Collision Detection (CSMA/CD) is used.
- Ethernet uses a bus topology.
- In CSMA/CD, each station has equal access to the network, but it can broadcast only when the network is idle. Before transmitting, a station: (1)-listens to the network to sense if another workstation is transmitting (carrier sense). If the network is still idle after a certain period, (2)-the station will transmit

Now, we go into the details of how exactly this is done. We have discussed the basic technology already. Ethernet uses a bus topology; it assumes a bus topology, carrier sense multiple access. In CSMA/CD, each station has equal access to the network but it can broadcast only when the network is idle. Before transmitting, a station: (1) listens to the network to sense if another workstation is transmitting, which is carrier sense. If the network is still idle after a certain period, (2) the station will transmit.

(Refer Slide Time: 21:35)

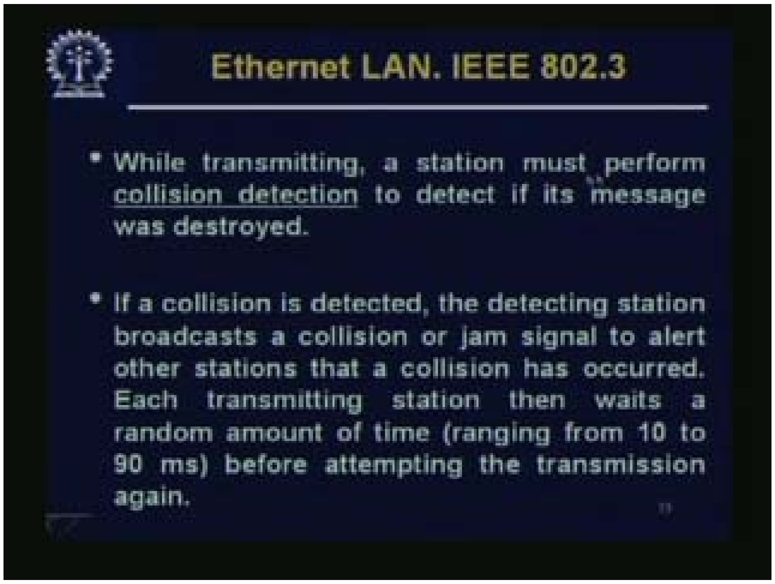


Ethernet LAN. IEEE 802.3

- It is possible that two stations will listen and sense an idle network at the same time. Each will then transmit its message on the network, where the messages will collide. Neither message will be usable if a collision occurs.

It is possible that two stations will listen and sense an idle network at the same time. Each will then transmit a message, which will collide.

(Refer Slide Time: 21:43)

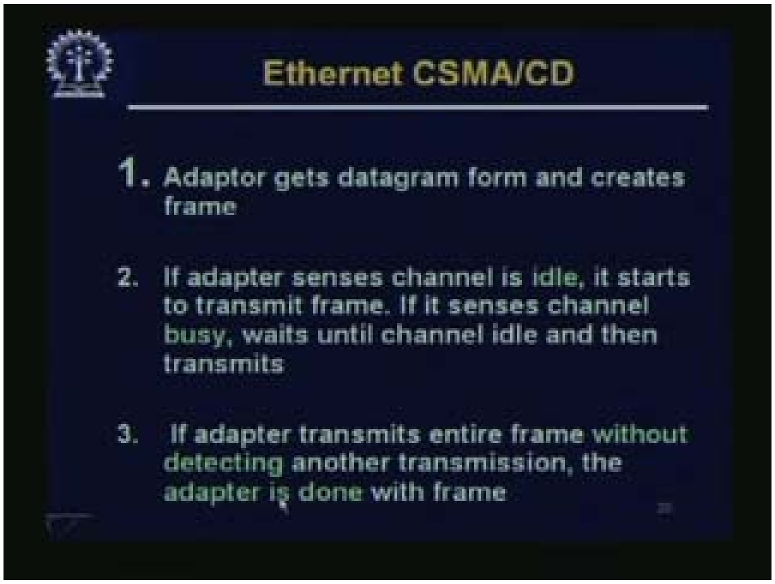


Ethernet LAN. IEEE 802.3

- While transmitting, a station must perform collision detection to detect if its message was destroyed.
- If a collision is detected, the detecting station broadcasts a collision or jam signal to alert other stations that a collision has occurred. Each transmitting station then waits a random amount of time (ranging from 10 to 90 ms) before attempting the transmission again.

While transmitting, a station must perform collision detection to detect if its message was destroyed. If a collision is detected, the detecting station broadcasts a collision or jam signal to alert other stations that a collision has occurred. Each transmission station then waits a random amount of time ranging from 10 to 90 µs before attempting the transmission again.

(Refer Slide Time: 22:06)



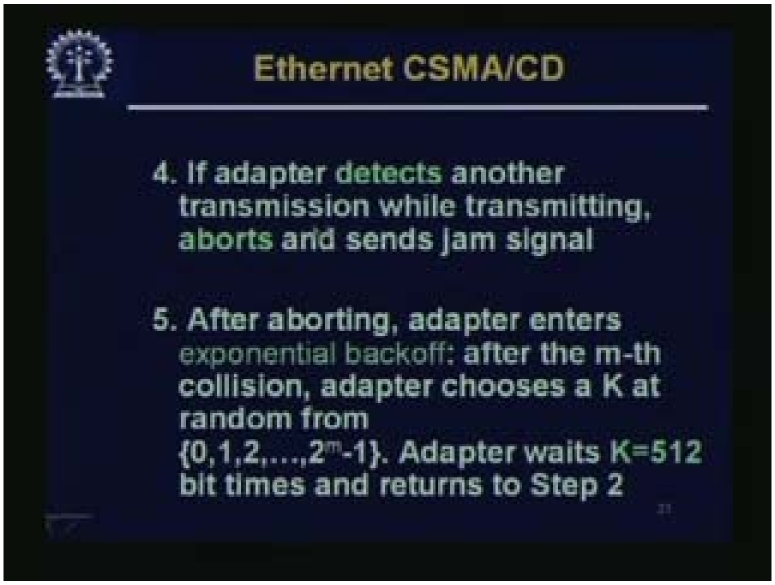
Ethernet CSMA/CD

1. Adaptor gets datagram form and creates frame
2. If adapter senses channel is idle, it starts to transmit frame. If it senses channel busy, waits until channel idle and then transmits
3. If adapter transmits entire frame without detecting another transmission, the adapter is done with frame

So this is the same algorithm step by step; adapter gets datagram from the network layer and creates a frame. That means it adds a header to create a frame. If the adapter senses that the channel is idle, it starts to transmit the frame; if it senses that the channel is busy, it waits until the channel is idle and then transmits.

If the adapter transmits the entire frame without detecting another transmission, the adapter is done with the frame. If the adapter detects another transmission while transmitting, it aborts the transmission and sends a jam signal.

(Refer Slide Time: 22:39)



Ethernet CSMA/CD

4. If adapter detects another transmission while transmitting, aborts and sends jam signal

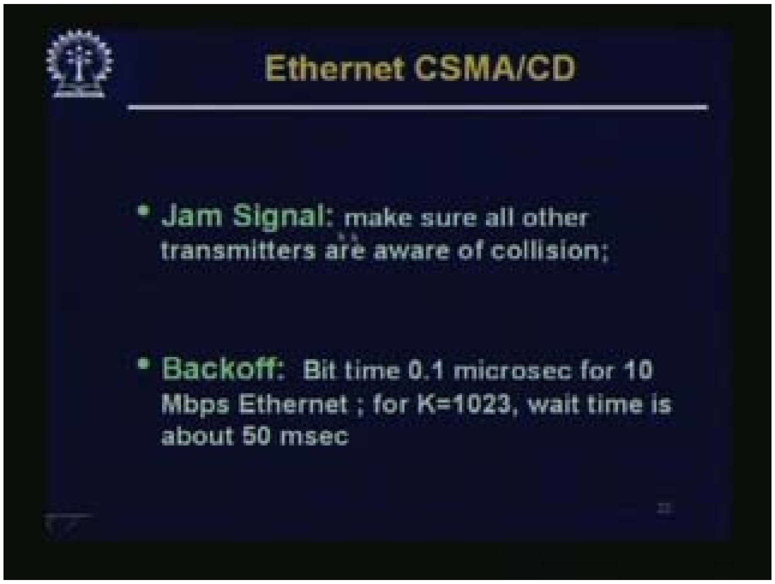
5. After aborting, adapter enters exponential backoff: after the m-th collision, adapter chooses a K at random from $\{0,1,2,\ldots,2^m-1\}$. Adapter waits K=512 bit times and returns to Step 2

This is an interesting point – it needs to detect collision only while it is transmitting. After the transmission is over, there is no detection within that frame transmission time; you can forget about collision. That means just some time after that, you may still detect the collision. That collision would be presumed to be due to another station, not your frame – because if you detect a collision after sending a frame, you have to understand whether it was your frame which collided or your frame went through and somebody nearby started transmitting and they collided. So the idea is that while you are transmitting, that time only should be enough for any collision, any inadvertent collision to happen; we will come back to this point. So if there is a collision, it aborts; after aborting, the adapter enters an exponential back-off algorithm. What is an exponential back-off algorithm? What might happen is that the frame sent might have collided and then it might have backed off for sometime; it may have tried again and it may have collided again; maybe the network is busy.

What is the remedy if the network is very busy? Obviously if the network is very busy, what is going to happen is that more and more people will try to send; they will fail and then they will try again. This may bring down the overall throughput of the system to a large degree. So what is done is that if the network is busy, everybody tries to make the network less loaded, which means that they wait for longer before they try to send it. What they do is that they generate this random number over an exponentially increasing period so that the probability of a higher random number becomes more and more if the network is more loaded, which means that you are backing off for a larger amount of time. That is exponential back-off, so after the mth collision, the adapter chooses a k at random from 0, 1, 2^{-1}. If m is 10, it chooses a random number between 0 to 1023.

The adapter waits k is equal to 512 bit times and then returns to step 2, which means, suppose k is 100, it would be 100 into 512 bit times. It waits 512 bit times and tries again.

(Refer Slide Time: 25:51)

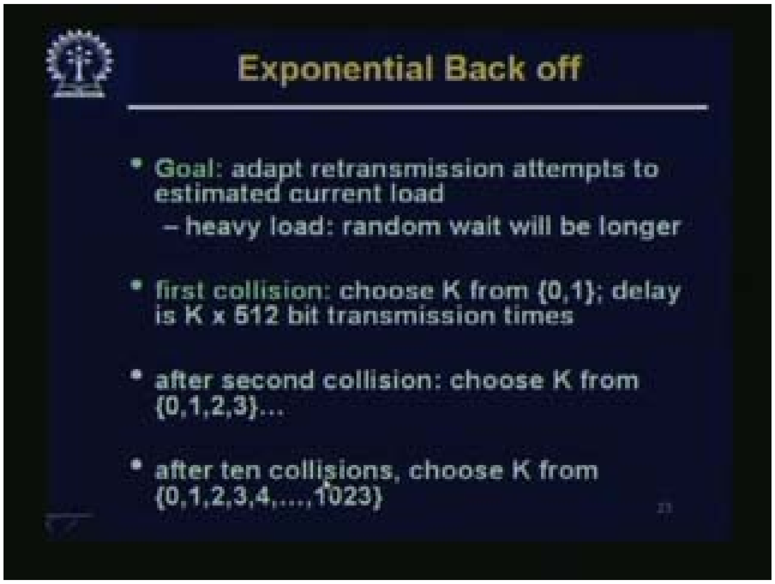


Ethernet CSMA/CD

- Jam Signal: make sure all other transmitters are aware of collision;
- Backoff: Bit time 0.1 microsec for 10 Mbps Ethernet ; for K=1023, wait time is about 50 msec

So jam signal makes sure that all other transmitters are aware of the collision. We send a jam when it detects a collision and backs off. We calculate the amount it backs off as follows: Let us say bit time is 0.1 µs for 10 Mbps Ethernet for k is equal to 1023. If you remember 512 into 0.1, 512 bit time should be about 51.2 µs and 51.2 µs into 1023 will be approximately 50 msec, i.e. it waits for 50 msec, if k has to come out as 1023.

(Refer Slide Time: 26:37)

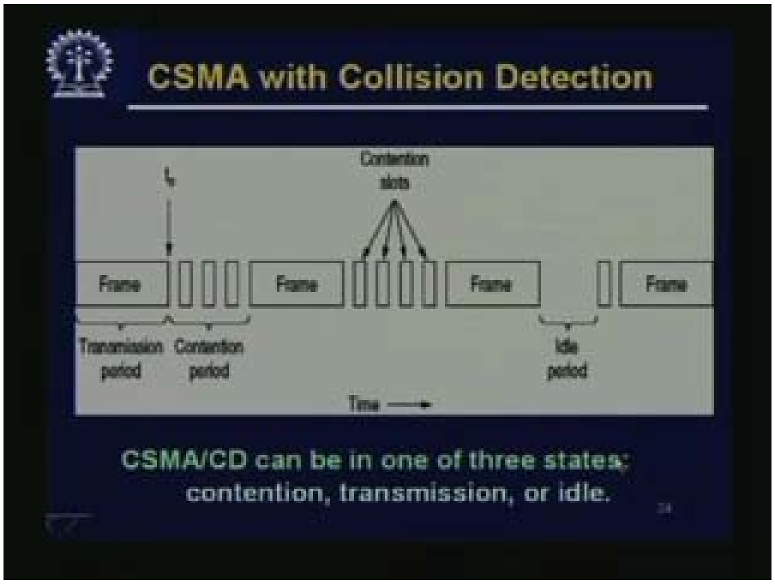


Exponential Back off

- Goal: adapt retransmission attempts to estimated current load
  - heavy load: random wait will be longer
- first collision: choose K from {0,1}; delay is K x 512 bit transmission times
- after second collision: choose K from {0,1,2,3}...
- after ten collisions, choose K from {0,1,2,3,4,...,1023}

The idea of this exponential back off is to adapt retransmission attempts to estimated current load. If it is heavy, the random wait will be longer. Once again, we do this because if the load is heavy at some particular point of time, then people in general should refrain from sending so that the load comes down. If the load is kept high and everybody starts to transmit, there will be even more collision and even less throughput than what you would achieve by a number of people backing off. This backing off is done exponentially, so first collision, choose k from 01, and delay is k into 512 bit transmission time, which means that with 50% probability, we try to transmit immediately, and with 50% probability, we would back off for this k into 512, i.e., 51.2 µs before listening and trying to send again. After the second collision, we choose k from 0123. The aim is increasing; the number of collisions is increasing. After 10 collisions, it is not allowed to increase more than that. This range of k becomes fixed and you always choose from this 0 to 1023. This is the exponential back-off algorithm; for heavier load presumably a node will choose a larger k and wait for longer, reducing the load on the system and increasing the throughput. This CSMA/CD system can be in one of the three states.

(Refer Slide Time: 28:23)

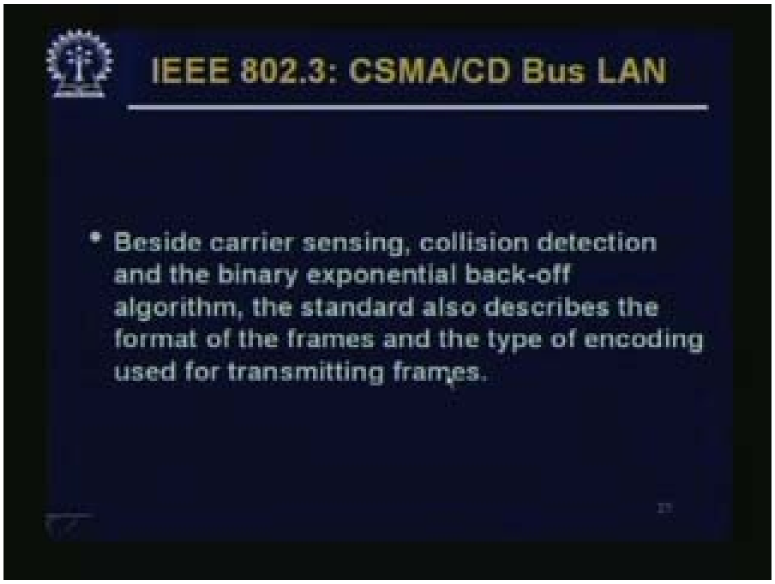


CSMA with Collision Detection

CSMA/CD can be in one of three states: contention, transmission, or idle.

There may be contention, there may be successful transmission, or it may be idle. There may be contention slots or frames or idle period.

(Refer Slide Time: 28:41)

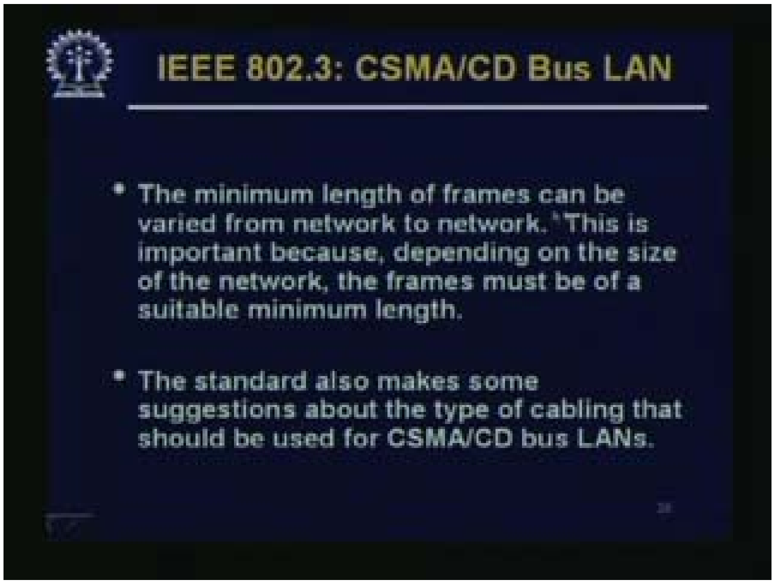


IEEE 802.3: CSMA/CD Bus LAN

- Beside carrier sensing, collision detection and the binary exponential back-off algorithm, the standard also describes the format of the frames and the type of encoding used for transmitting frames.

IEEE 802.3 defines this Ethernet protocol. If you remember 802.2 was LLC and 802.3 was for CSMA/CD and 802.45 etc., like token bus token ring etc. There are a couple of variants of Ethernet: one is IEEE 802.3 and the other is Dixie Ethernet. We will maybe mention it later on, but they are more or less the same, excepting for minor differences. 802.3 is the more dominant one these days; besides, this is the standard. 802.3 is the number of the standard and it describes the format of the frames and the type of encoding used for transmitting frames. It also describes what the frame format is and what kind of encoding is used for transmitting the frame. The minimum length of frames can be varied from network to network. This is important because, depending on the size of the network, the frames must be of a suitable minimum length.

(Refer Slide Time: 29:59)

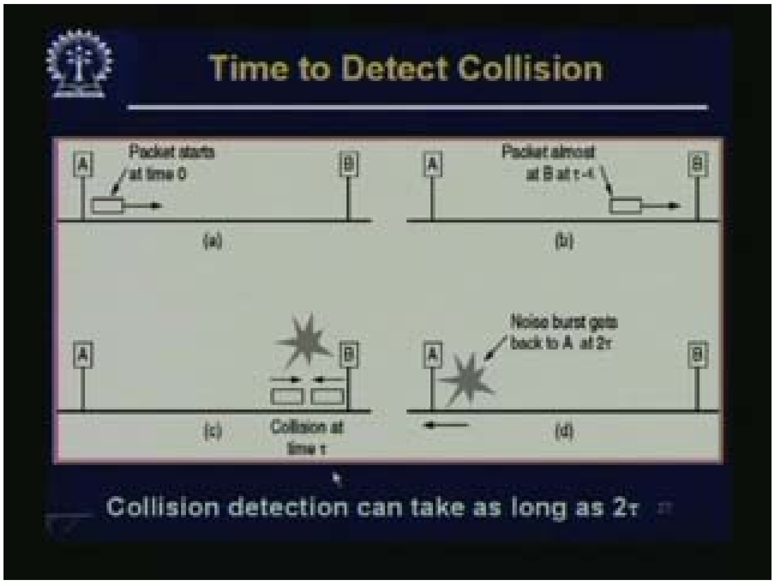


IEEE 802.3: CSMA/CD Bus LAN

- The minimum length of frames can be varied from network to network. This is important because, depending on the size of the network, the frames must be of a suitable minimum length.
- The standard also makes some suggestions about the type of cabling that should be used for CSMA/CD bus LANs.

Remember that a node has to detect collision only during the transmission time. If transmission is over and there has been no collision detected, it assumes that its own frame has reached whatever destination it was supposed to reach. After that, if there is a collision, it is supposed to be between two other nodes because once there is a collision only the sender would know whether his own frame has collided because you cannot really decipher the address field of the collided frame. The collided frames are gone, they are garbled. The standard also makes some suggestions about the type of cabling that should be used for CSMA/CD, bus LANs, etc. This minimum time may vary from system to system; that means, network to network. This is how it is calculated. First of all, let's see how the minimum time would come. Suppose this is the shared bus.

(Refer Slide Time: 31:02)

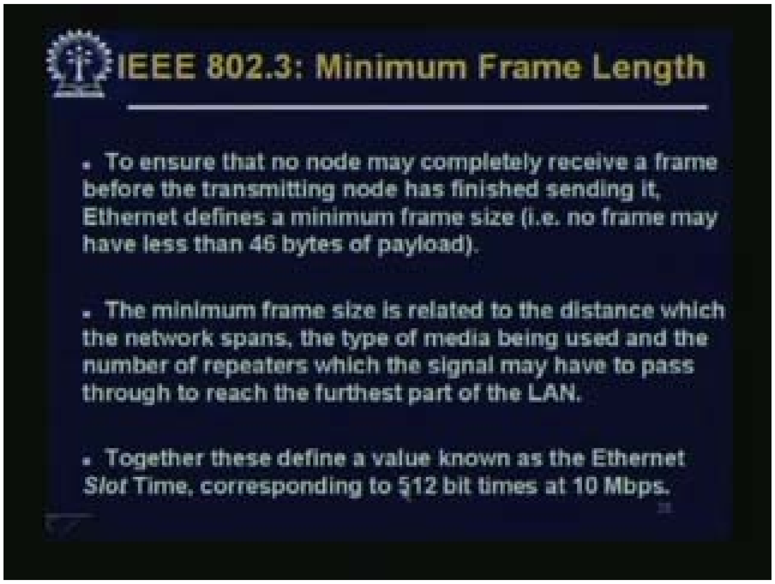


Time to Detect Collision

Collision detection can take as long as 2τ

We put two nodes at the two extreme ends, the packet starts over here at time t = 0, it almost reached B at time (t – ε). Suppose this propagation time of the packet, from one end to the other, is τ, at (τ – ε), it has almost reached B at that point of time. B starts transmitting because while B has been finding that, the bus has been very quiet; there is no signal on it. B can start transmitting and it starts sending. There is a collision at one end of the medium. The collision bursts, starts' traveling back and the jam signal is sent. That starts travelling back, it again takes time τ to reach A. The total time is 2 τ, the collision detection can take as long as 2 τ. The point is that your transmission time has to be greater than this 2 τ. There are a few other issues; we will discuss them later.

(Refer Slide Time: 32:42)



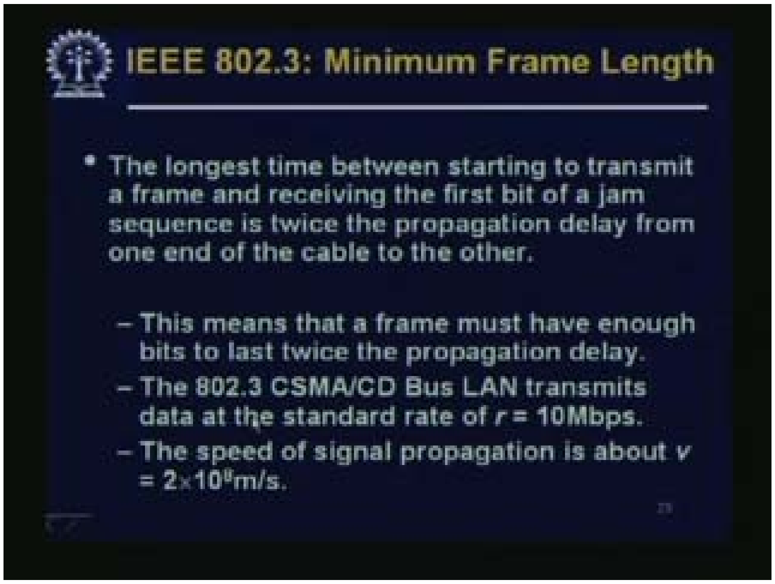
IEEE 802.3: Minimum Frame Length

- To ensure that no node may completely receive a frame before the transmitting node has finished sending it, Ethernet defines a minimum frame size (i.e. no frame may have less than 46 bytes of payload).
- The minimum frame size is related to the distance which the network spans, the type of media being used and the number of repeaters which the signal may have to pass through to reach the furthest part of the LAN.
- Together these define a value known as the Ethernet *Slot* Time, corresponding to 512 bit times at 10 Mbps.

This is what we were discussing, minimum frame length to ensure that no node may completely receive a frame before the transmitting node has finished sending it. Ethernet defines a minimum frame size, i.e. no frame may have less than 46 bytes of payload. Remember payload is the packet, which the data link layer receives from the upper layer. What happens if the packet is much smaller? The application itself is such that you have to send maybe 1 character or 2 characters; what happens to it? Ethernet does not allow less than 46 bytes, so you have to pad it up; we will come to that later. The minimum frame size is related to the distance which the network spans, the type of media being used – because of the time it takes for the signal to travel – and the number of repeaters which the signal may have to pass through to reach the furthest part of the LAN. Repeater if you remember is something, which enhances the signal.

As a signal travels down the transmission line, it tends to get weaker and weaker because of attenuation. At some particular point of time the signal may have to be amplified. A repeater does just that, it takes some incoming signal and amplifies it, maybe some wave shipping or something. There may be a number of repeaters. Each of these repeaters might introduce some delay because the repeater will not go through instantaneously. There will be some delay in this repeater and if there are a number of repeaters, this delay would get added. Remember what we are trying to find out. When the packet being sent from one end of the network, reaches the other end and almost at the other end, there is a collision and that collision comes back. What is this total time? During this whole time, this packet must be transmitting because it will detect the collision only during the packet transmission time. The number of repeaters also is important as are the type of media and distance to the furthest part of the LAN. Together these define a value known as the Ethernet slot time, corresponding to 512 bit times at 10 Mbps.

(Refer Slide Time: 35:20)

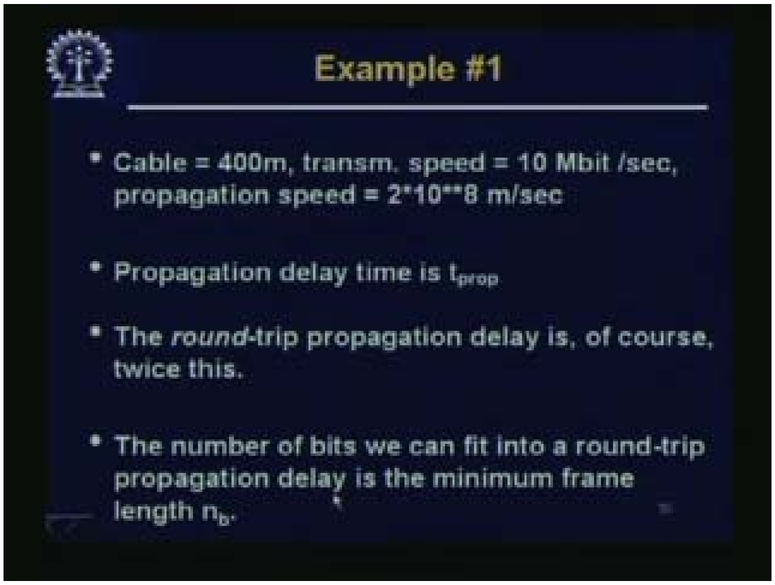


IEEE 802.3: Minimum Frame Length

- The longest time between starting to transmit a frame and receiving the first bit of a jam sequence is twice the propagation delay from one end of the cable to the other.
  - This means that a frame must have enough bits to last twice the propagation delay.
  - The 802.3 CSMA/CD Bus LAN transmits data at the standard rate of r = 10Mbps.
  - The speed of signal propagation is about $v = 2\times10^8$m/s.

The longest time between starting to transmit a frame and receiving the first bit of a jam sequence is twice the propagation delay from one end of the cable to the other. This means that a frame must have enough bits to last twice the propagation delay. The 802.3 CSMA/CD bus LAN transmits data at the standard rate of r is equal to 10 Mbps. This was the earlier standard of 10 Mbps. The speed of signal propagation is about V is equal to 2 into 10^8 µs in this coaxial cable.

(Refer Slide Time: 35:57)

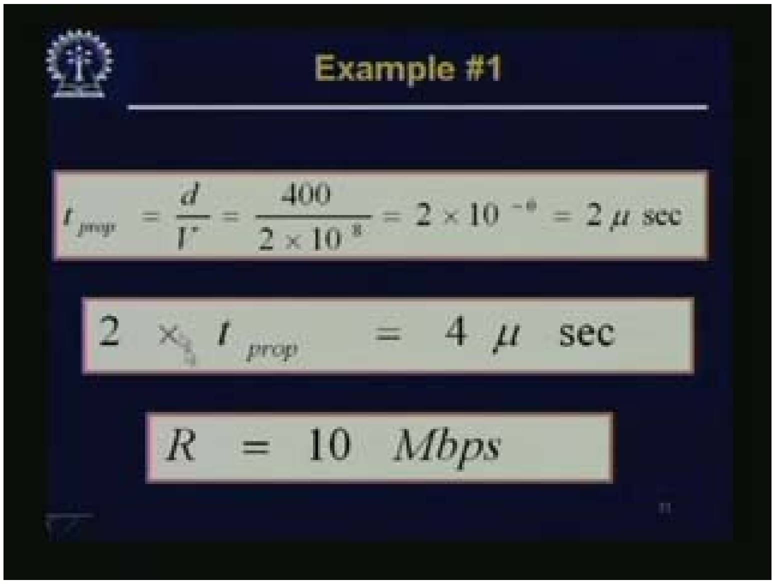


Example #1

- Cable = 400m, transm. speed = 10 Mbit /sec, propagation speed = 2*10**8 m/sec
- Propagation delay time is t_{prop}
- The *round*-trip propagation delay is, of course, twice this.
- The number of bits we can fit into a round-trip propagation delay is the minimum frame length n_b.

The cable maybe of 400 m length, transmission speed is equal to 10 Mbps, so propagation speed is 2*10**8 µs. Let us see what the delay is and what the minimum frame size comes out to be. Propagation delay time is t_{prop}; the round-trip propagation delay is twice this number of bits.

(Refer Slide Time: 36:29)



Example #1

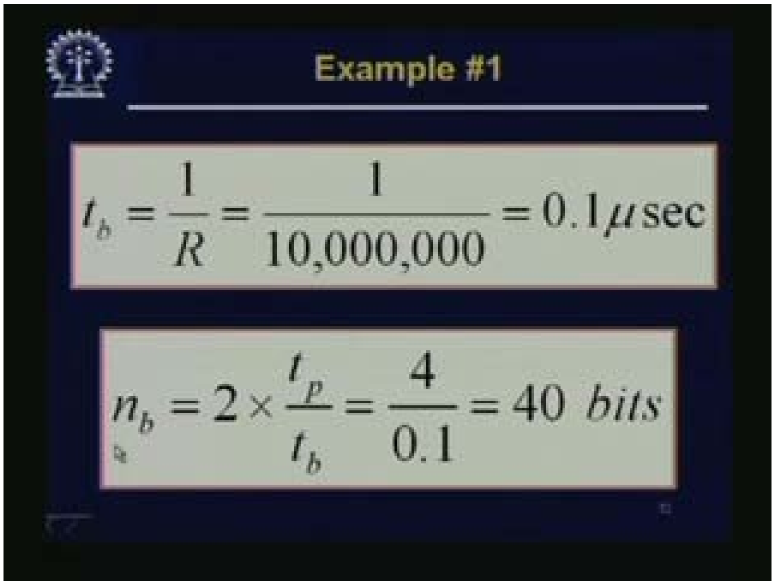
$$t_{prop} = \frac{d}{V} = \frac{400}{2 \times 10^{8}} = 2 \times 10^{-6} = 2\,\mu\text{ sec}$$

$$2 \times t_{prop} = 4\ \mu\ \text{sec}$$

$$R = 10\ Mbps$$

We can fit into a round-trip propagation delay, the minimum frame length n_b. t_{prop} is equal to d/V is equal to 2 into 10^8 and distance is 400 m, which makes it 2 into 10^{-6} sec or rather 2 µs, 2 intot$_{prop}$; that means, for the signal to travel all these 400 m and come back, it takes 4 µs. For the 4 µs, the transmitting station must go on transmitting and the transmitting station is pumping data at the rate of r is equal to 10 Mbps.

(Refer Slide Time: 37:05)



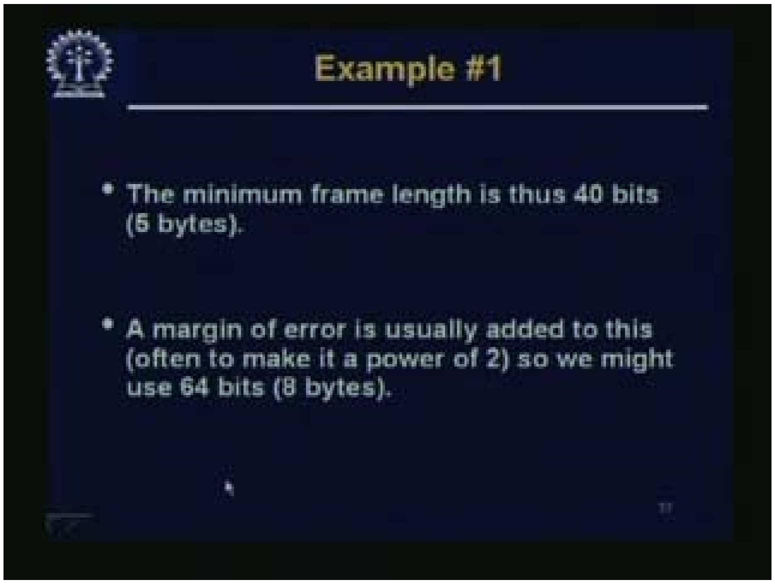
Example #1

$$t_b = \frac{1}{R} = \frac{1}{10,000,000} = 0.1\mu\text{sec}$$

$$n_b = 2 \times \frac{t_p}{t_b} = \frac{4}{0.1} = 40\ bits$$

A bit time, the time to transmit 1 bit is 1byR; it is 1by10 megabits, which is 10 into 10^6 which is 0.1 µs. The minimum number of bits the frame must have is 2 into t_p/t_b. t_p is the propagation delay; 2 into t_p for the round trip propagation delay divided by t_b is equal to 4/0.1 is equal to 40 bits. This calculation brings us to 40 bits. As I said the minimum size is much bigger than 40 bits, and we will see why.

(Refer Slide Time: 37:53)

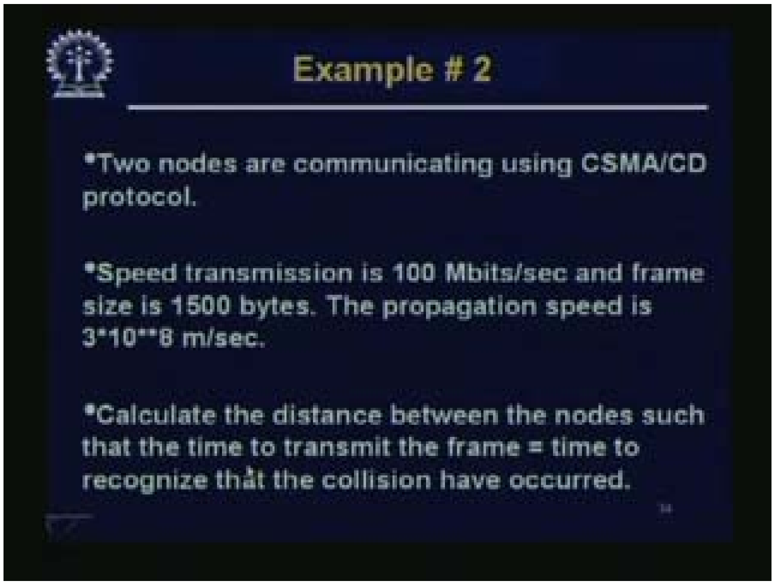


Example #1

- The minimum frame length is thus 40 bits (5 bytes).
- A margin of error is usually added to this (often to make it a power of 2) so we might use 64 bits (8 bytes).

The minimum frame length is thus 40 bits or 5 bytes. A margin of error is usually added to this, often to make it a power of two, so we might use 64 bits or 8 bytes.

(Refer Slide Time: 38:08)



Example # 2

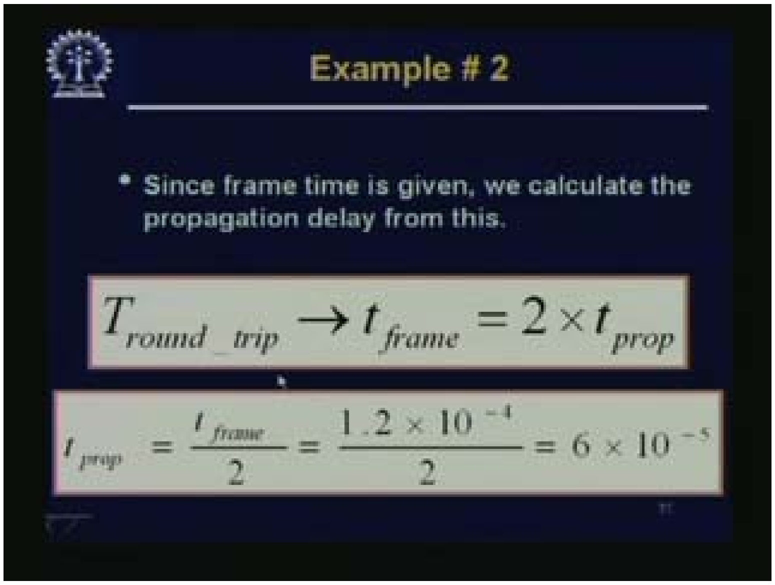
*Two nodes are communicating using CSMA/CD protocol.

*Speed transmission is 100 Mbits/sec and frame size is 1500 bytes. The propagation speed is 3*10**8 m/sec.

*Calculate the distance between the nodes such that the time to transmit the frame = time to recognize that the collision have occurred.

Let us see another example, calculating something else but it is closely related. Two nodes are communicating using CSMA/CD protocol, and the speed transmission is 100 mbps. This is a fast Ethernet and frame size is 1500 bytes. The propagation speed is 3 into 10^8 meters per second – that is the speed of the light. Calculate the distance between the nodes such that the time to transmit the frame equals the time to recognize that the collision has occurred. We want to calculate the distance.

(Refer Slide Time: 38:54)



Example # 2

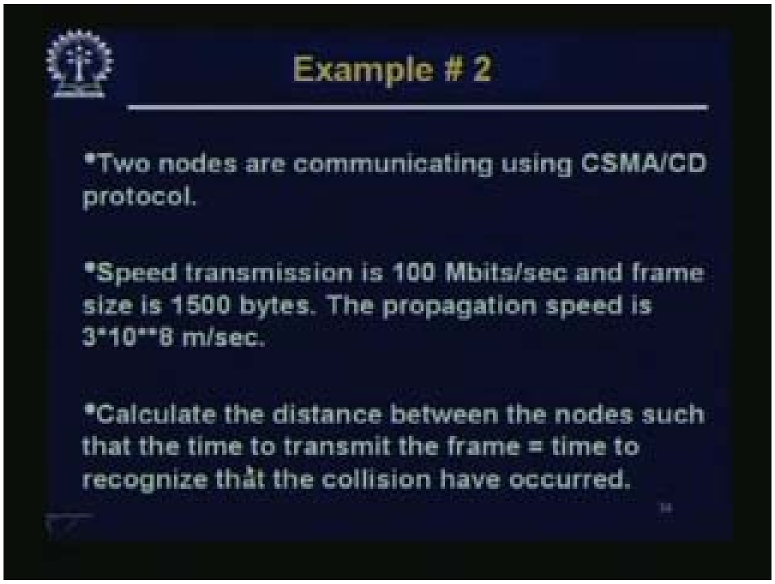
- Since frame time is given, we calculate the propagation delay from this.

$$T_{round\_trip} \rightarrow t_{frame} = 2 \times t_{prop}$$

$$t_{prop} = \frac{t_{frame}}{2} = \frac{1.2 \times 10^{-4}}{2} = 6 \times 10^{-5}$$

Since frame time is given we calculate the propagation delay from this. So $T_{round\ trip}$ is equal to t_{frame} is equal to 2 into t_{prop} so t_{prop} is equal to tframe by 2, and what is the t_{frame}? We have 1500 bytes, which is being pumped at the rate of 100 mbps.

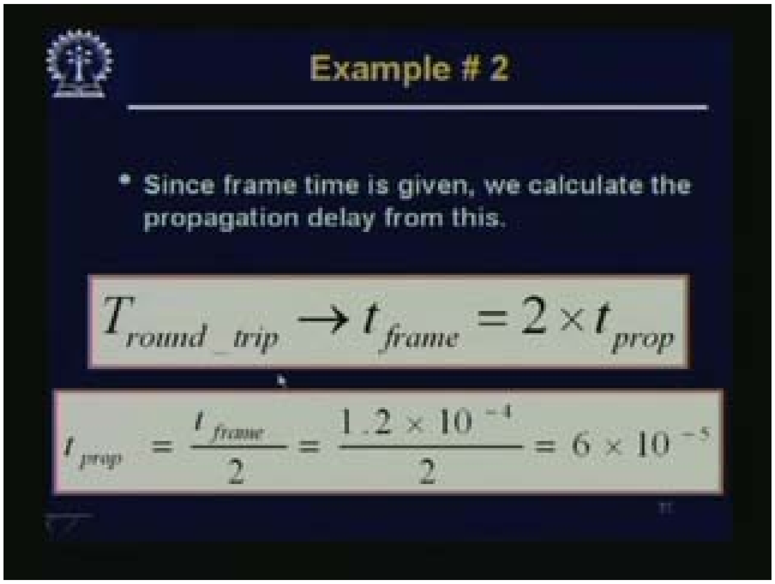
(Refer Slide Time: 39:12)



Example # 2

- Two nodes are communicating using CSMA/CD protocol.
- Speed transmission is 100 Mbits/sec and frame size is 1500 bytes. The propagation speed is 3*10**8 m/sec.
- Calculate the distance between the nodes such that the time to transmit the frame = time to recognize that the collision have occurred.

(Refer Slide Time: 39:23)



Example # 2

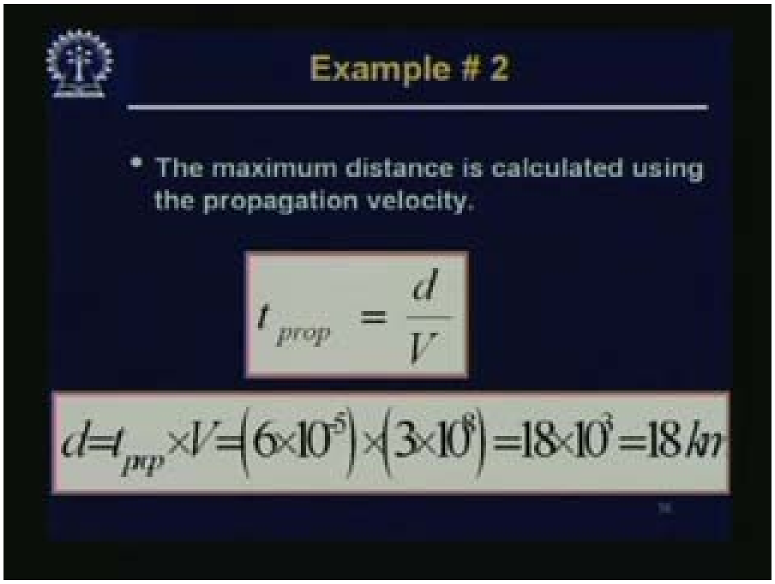
- Since frame time is given, we calculate the propagation delay from this.

$$T_{round\_trip} \rightarrow t_{frame} = 2 \times t_{prop}$$

$$t_{prop} = \frac{t_{frame}}{2} = \frac{1.2 \times 10^{-4}}{2} = 6 \times 10^{-5}$$

If you multiply 1500 bytes into eight, that means how many bits will get 12000 bits which is 1.2 into 10^{-4}, i.e., 6 into 10^{-5} The maximum distance is calculated using the propagation velocity, t_{prop} is equal to d/V is equal to tprop into V is equal to (6 into 10 minus 5) into (3 into 10^8) is equal to 18 into 10^3 is equal to 18 km.

(Refer Slide Time: 39:53)



Example # 2

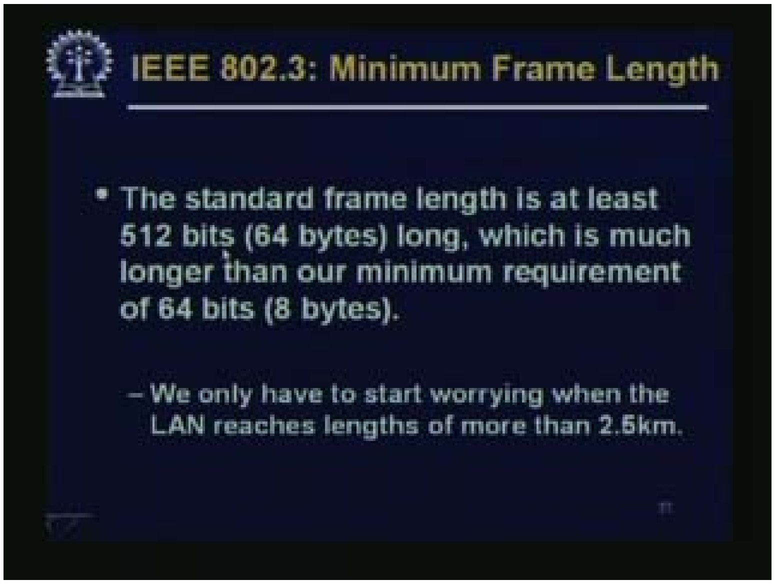
- The maximum distance is calculated using the propagation velocity.

$$t_{prop} = \frac{d}{V}$$

$$d = t_{prp} \times V = \left(6 \times 10^{5}\right) \times \left(3 \times 10^{8}\right) = 18 \times 10^{3} = 18\,km$$

With this kind of minimum frame length, you can go up to 18 km. Remember; the minimum frame size has become 1,500 bytes. This standard frame length is at least 512 bits or 64 bytes long, which is much longer than our minimum requirement of 64 bits which is 8 bytes. We only have to start worrying when the LAN reaches lengths of more 2.5 km.

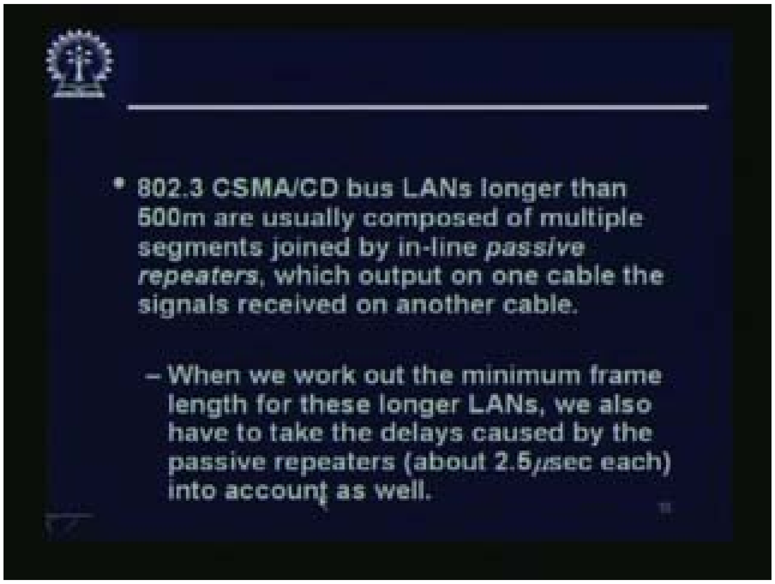
(Refer Slide Time: 40:29)



IEEE 802.3: Minimum Frame Length

- The standard frame length is at least 512 bits (64 bytes) long, which is much longer than our minimum requirement of 64 bits (8 bytes).
  - We only have to start worrying when the LAN reaches lengths of more than 2.5km.

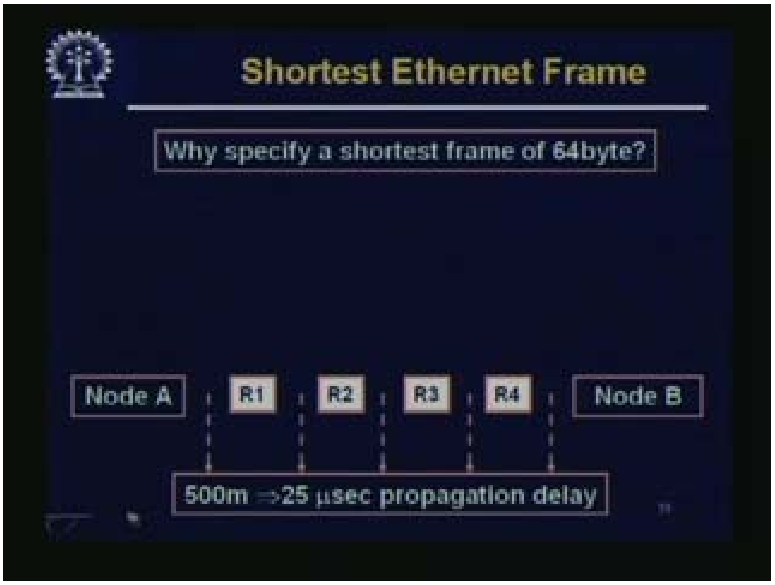
(Refer Slide Time: 40:49)



- 802.3 CSMA/CD bus LANs longer than 500m are usually composed of multiple segments joined by in-line *passive repeaters*, which output on one cable the signals received on another cable.
  - When we work out the minimum frame length for these longer LANs, we also have to take the delays caused by the passive repeaters (about 2.5μsec each) into account as well.

802.3 CSMA/CD bus LANs longer than 500 m are usually composed of multiple segments joined by in-line passive repeaters. As a signal travels down transmission line, what would happen is that it will become weaker, so we will have to put a repeater, and one kind of repeater maybe called a hub. In-line passive repeaters output on one cable the signals received on another cable or we may simply have an amplifier over there. When we work out the minimum frame length for these longer LANs, we also have to take the delays caused by the passive repeater; each passive repeater introduces a delay of about 2.5 µs each, so we have to take this into account as well.

(Refer Slide Time: 41:42)



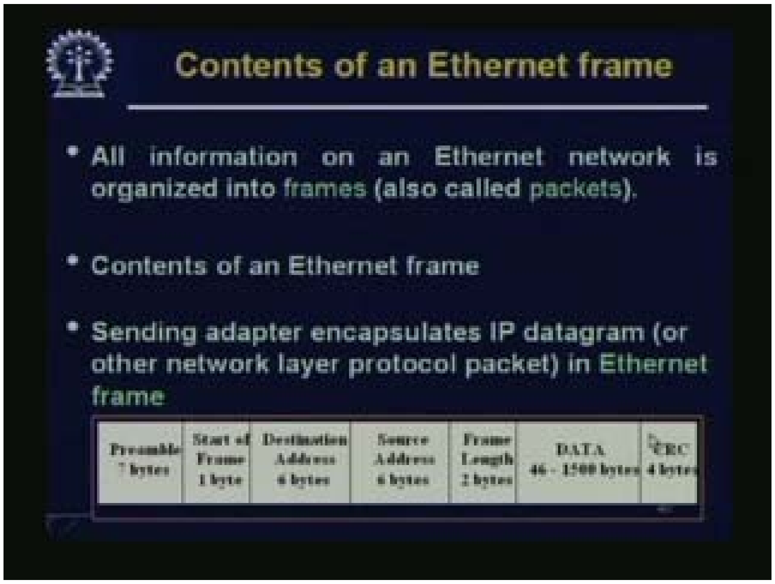
Shortest Ethernet Frame

Why specify a shortest frame of 64byte?

500m ⇒25 µsec propagation delay

Let us say we have 500 m on each segment, we have 4 repeaters, so 5 segments. That is about the maximum you should go. Nowadays we do not use such coaxial cables. What happens is that 64 bytes sent at 10 Mbps would take 64 into 8 is equal to 512 bits; 512 bits at 10 mbps is 51.2 µs and 500 m per segment. Four repeaters between the nodes means there are 5 segments; 2500 m in all. This gives rise to 25 µs propagation delay, because remember our propagation velocity is about 2 into 10^8. If you calculate, it comes to about 12.5 µs from one end to the other, twice which is about 25 µs. The frame should be long enough for the sender to detect the collision in 2 into 25, this is 125 and the other 25 is these 4 repeaters together, about 50 µs; 64 bytes sent at 10 Mbps is 51.2 µs. If you have a frame length of 64 bytes, that is quite good.

(Refer Slide Time: 43:24)



Contents of an Ethernet frame

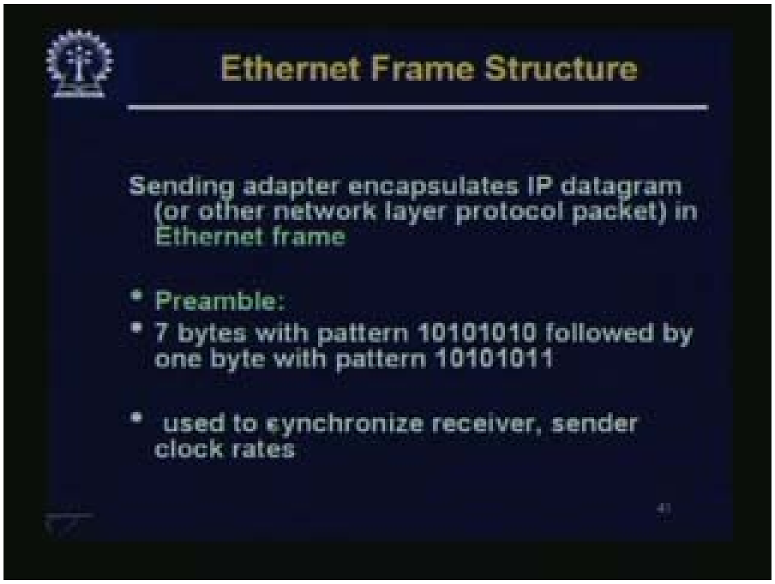
- All information on an Ethernet network is organized into frames (also called packets).
- Contents of an Ethernet frame
- Sending adapter encapsulates IP datagram (or other network layer protocol packet) in Ethernet frame

Preamble 7 bytes	Start of Frame 1 byte	Destination Address 6 bytes	Source Address 6 bytes	Frame Length 2 bytes	DATA 46 - 1500 bytes	CRC 4 bytes

Let us look at the details of the Ethernet frame. All information on an Ethernet network is organized into frames also called packets or may be packet is what the upper layer gives and then this is formed into a frame. contents of an Ethernet frame: Sending adapter encapsulates IP datagram or other network layer protocol packet in Ethernet frame. This is what the Ethernet frame looks like: we have a preamble of 7 bytes, we have a start of frame of 1 byte, we have a destination address of 6 bytes, we have a source address of 6 bytes, we have frame length of 2 bytes and then data may vary from 46 to 1500 bytes and then followed by CRC of 4 bytes.

If you remember CRC is the cyclic redundancy code which Ethernet uses for error detection on the other side. So you have actually about 72 bytes, which is even larger than the 64 bytes. We require 46 bytes the minimum and 1500 bytes is the maximum length of the payload, this is the payload part; to this we have to add this header and trailer part. In the trailer you have the CRC in the header the source address destination address, etc. This frame length is required because you might have padding over here. In some other version of Ethernet, there is a type field over here, which gives the network layer protocol type.

(Refer Slide Time: 45:28)



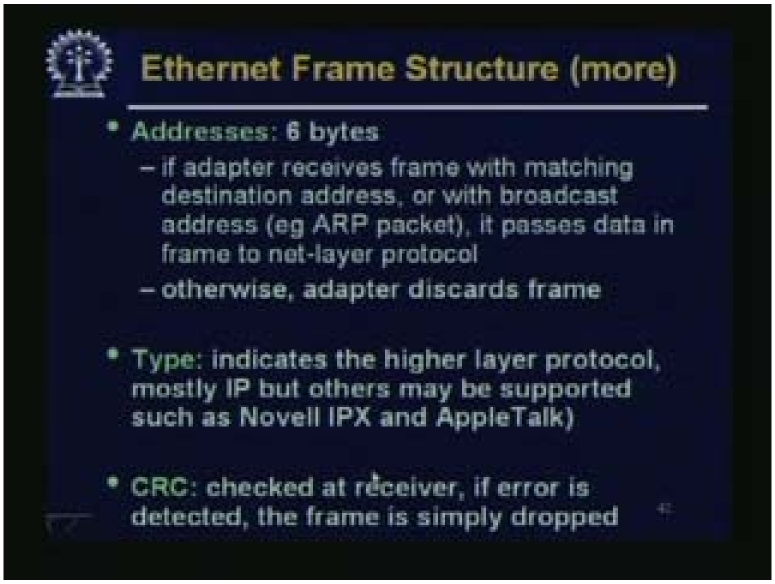
Ethernet Frame Structure

Sending adapter encapsulates IP datagram (or other network layer protocol packet) in Ethernet frame

- Preamble:
- 7 bytes with pattern 10101010 followed by one byte with pattern 10101011
- used to synchronize receiver, sender clock rates

Let us look at the details quickly. Sending adapter encapsulates IP datagram Ethernet frame; preamble has 7 bytes with the pattern 10101010. Please note that 1s and 0s are alternating – and there is a reason for that – followed by 1 byte with pattern 10101011; so these 7 bytes are called the preamble. This 10101011 is the start of frame delimiter, 7 bytes of pattern. This 1 0 is required; it is used to synchronize receiver and sender clock rates. That is why we alternate 1 and 0 etc.

(Refer Slide Time: 46:09)

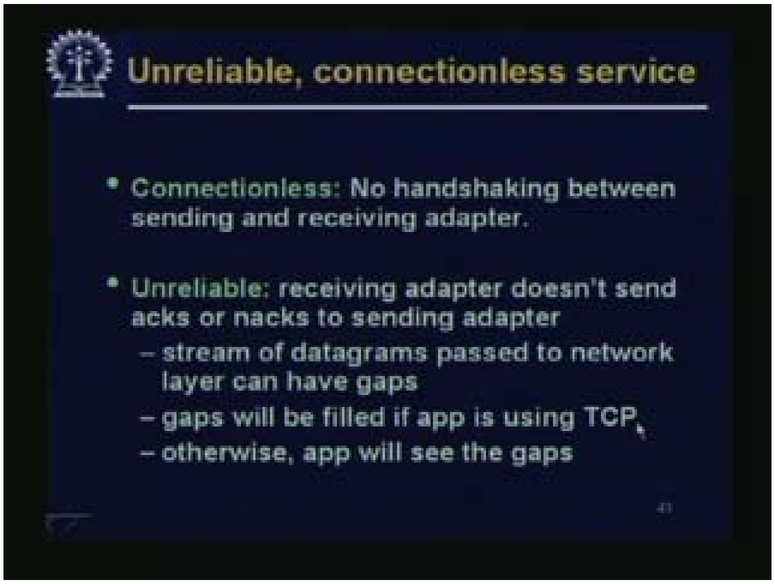


Ethernet Frame Structure (more)

- Addresses: 6 bytes
  - if adapter receives frame with matching destination address, or with broadcast address (eg ARP packet), it passes data in frame to net-layer protocol
  - otherwise, adapter discards frame
- Type: indicates the higher layer protocol, mostly IP but others may be supported such as Novell IPX and AppleTalk)
- CRC: checked at receiver, if error is detected, the frame is simply dropped

We have an address of 6 bytes. If the adapter receives a frame with matching destination address or with broadcast address, for example an ARP packet, it may be broadcast, it may be multicast or it may be unicast. If actually a frame would be broadcast, only 1 would come up. ARP is the address resolution protocol; we will discuss this in the next lecture. It passes data in the frame to the net layer protocol; otherwise the adapter discards the frame. So this address has to be there. The destination address and source address are also put over there. Type: as I said, in some form the type indicates the higher layer protocol, mostly IP, but others are supported, such as Novell IPX and Apple Talk. What type of network layer protocol is it talking to? The same Ethernet – as I mentioned before – the same data link layer protocol may be supporting a number of network layer protocols. In that case, there has to be multiplexing and de-multiplexing. In the receiver side when a packet comes, maybe this data link layer knows that this is for this particular machine because its address is there so it will absorb it but then send it. If there are two different network layer protocols, which are running at the same time, whom will it send it to? That type is mentioned in the field so that it can do a de-multiplexing properly. CRC or cyclic redundancy code is checked at the receiver. If an error is detected, the frame is simply dropped.

(Refer Slide Time: 47:48)



Unreliable, connectionless service

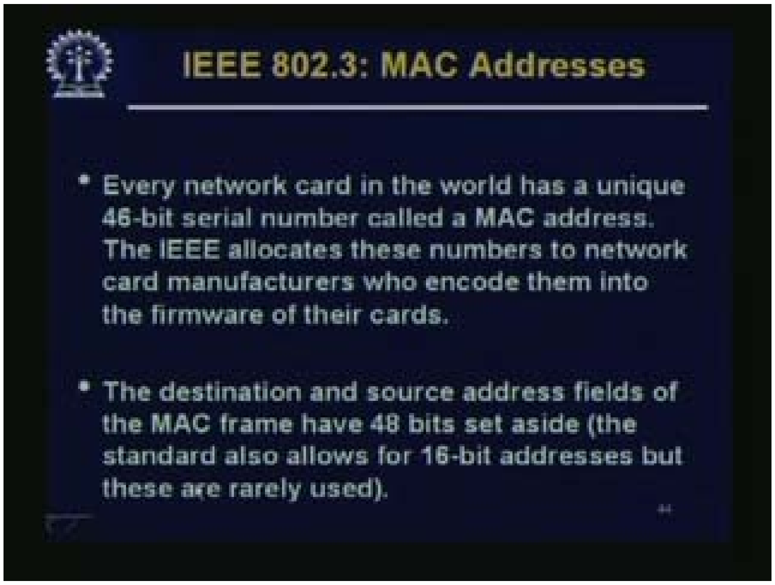
- Connectionless: No handshaking between sending and receiving adapter.
- Unreliable: receiving adapter doesn't send acks or nacks to sending adapter
  - stream of datagrams passed to network layer can have gaps
  - gaps will be filled if app is using TCP
  - otherwise, app will see the gaps

41

The other thing to understand is that this – Ethernet gives you an unreliable connectionless service. It’s connectionless, and there is no handshaking between sending and receiving adapters; so it is connectionless. The sender simply sends some packet and there is no handshaking between the two. The actual reliability, whether there will be error or not, will depend on the quality of the cable, the quality of connections, whether all standards have been followed very meticulously, and the quality of distance and the ambient noise and these kinds of things. That is not what we are talking about, it is unreliable in the sense that whatever physical layer parameters are there, they are given to the data link layer, that is, the Ethernet layer. But the Ethernet does not really try to take any special care to see that the communication is reliable. It has a CRC for sure, so on the other side if the receiver sees that the CRC shows some error, it may drop the packet. But the receiver does not send any acknowledgement to the sender. The sender does not know about it. If that packet is dropped that frame is dropped.

If that has to be handled, it can only be handled at a higher layer. So it is unreliable in that sense, receiving adapter does not send ACKs or negative acknowledgements to the sending adapter. The stream of datagram's passed to network layer can have gaps; that means some of the frames may have been dropped. Gaps will be filled if the application is using TCP. This is one kind of transmission protocol, which we will discuss later, which tries to take care of such errors. Otherwise, the application will have to handle these gaps.

(Refer Slide Time: 49:54)



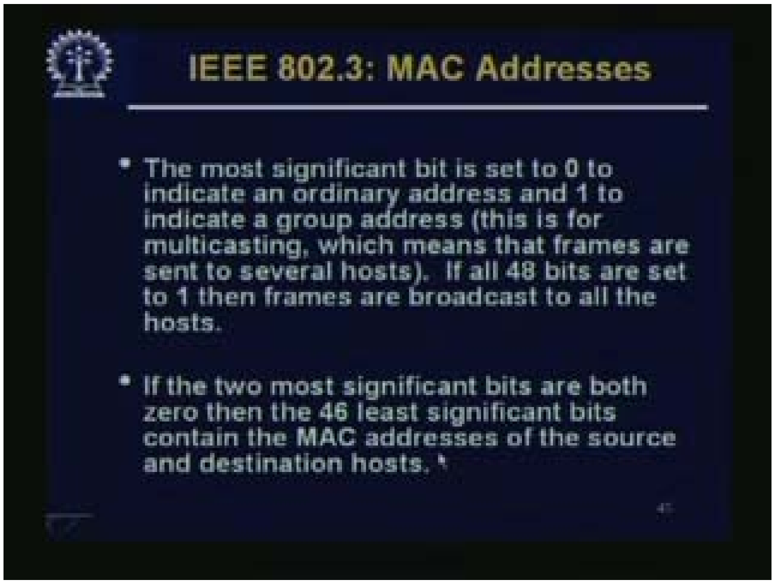
IEEE 802.3: MAC Addresses

- Every network card in the world has a unique 46-bit serial number called a MAC address. The IEEE allocates these numbers to network card manufacturers who encode them into the firmware of their cards.
- The destination and source address fields of the MAC frame have 48 bits set aside (the standard also allows for 16-bit addresses but these are rarely used).

Just a couple of words about the addresses – remember there are 6 byte addresses. Every network card in the world has a unique 46-bit serial number, called a MAC address. We are talking about the Ethernet, there is 46 bit; 2^{46} is about 64 trillion, which is a very large number and these numbers or these addresses are distributed by IEEE. A manufacturer who manufactures Ethernet NIC – that means Ethernet network interface cards – will apply to IEEE and get a whole chunk of addresses, whole block of addresses and will put these addresses one by one into the network cards that are produced and then sell in the market. What happens is that one of the network cards may have gone to India and the next network card may be used in Beijing.

It may so happen that the network cards we have and not the network addresses in the LAN will have very contiguous addresses. They may not; anyway, what is guaranteed is that since this is distributed centrally from IEEE, no two addresses, i.e. no two MAC addresses of two network interface cards or two Ethernet cards are going to be the same. They are all going to be distinct, because we have this pool of 64 trillion addresses being distributed by IEEE. IEEE allocates these numbers to network card manufacturers, who encode them into the firmware on their cards. It is almost in the hardware, encoded in the firmware in the card, the destination and source address fields of the MAC frame have 48 bits set aside. Remember 6 bytes is 48 bits. There is a 2-bit discrepancy. These 2 bits are used for something, the standard also allows for a 16-bit address but they are rarely used.

(Refer Slide Time: 52:21)



IEEE 802.3: MAC Addresses

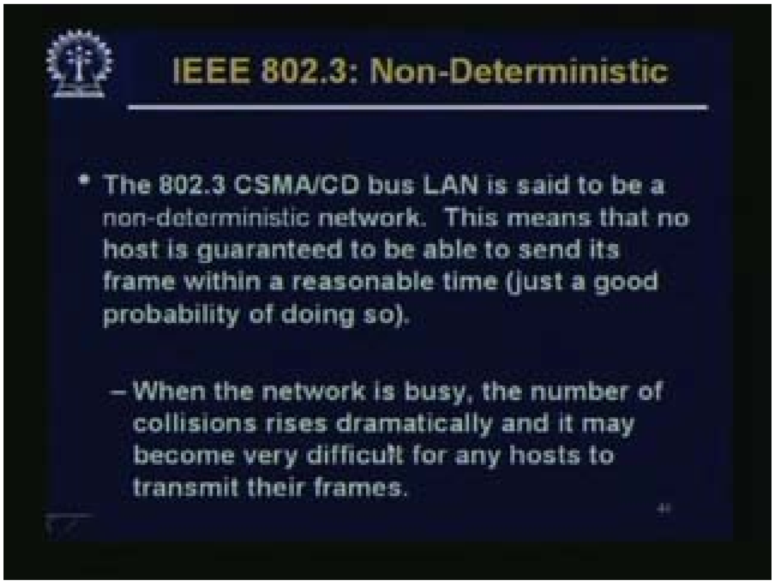
- The most significant bit is set to 0 to indicate an ordinary address and 1 to indicate a group address (this is for multicasting, which means that frames are sent to several hosts). If all 48 bits are set to 1 then frames are broadcast to all the hosts.
- If the two most significant bits are both zero then the 46 least significant bits contain the MAC addresses of the source and destination hosts.

45

In these 2 bits, the most significant bit is set to 0 to indicate an ordinary address, and 1 is to indicate a group address. This is for multicasting, which means that frames are sent to several hosts. Remember, in the beginning of the lecture we said that although the medium is broadcast, usually there will be 1 sender and 1 receiver, i.e., it is a unicast; that means 1 sender 1 receiver. Sometimes you may have a broadcast; that means it is meant for everybody, and sometimes you may have a multicast; that means it is meant for a few of the hosts in the network, not for all. How do you indicate the address? You indicate a very special address starting with a 1 for multicasting. If all 48 bits are set to 1, frames are broadcast. So for the destination field if all bits are 1, there is a very special address, not the address of any particular machine, which is meant to match with any Ethernet address on that particular LAN. If 2 most significant bits are both 0, the 46 least significant bits contain the MAC addresses of the source and destination hosts. This is how you make out.

Also this is connectionless and unreliable in the sense that it does not take any special care for reliability. This is also non-deterministic network; this means that no host is guaranteed to be able to send its frame within a reasonable time, just a good probability of doing so. Theoretically what might happen is that you may try to go and find it busy. Again you come back and then go; it might collide next time you come back and this might happen an indefinite number of times. Although the probability of its happening in a large number of times becomes lower and lower, with a very good probability, we will be able to send it but there is no guarantee that within this time you will definitely be able to send your frame. When the network is busy, the number of collisions rises dramatically and it may become very difficult for any host to transmit its frames.

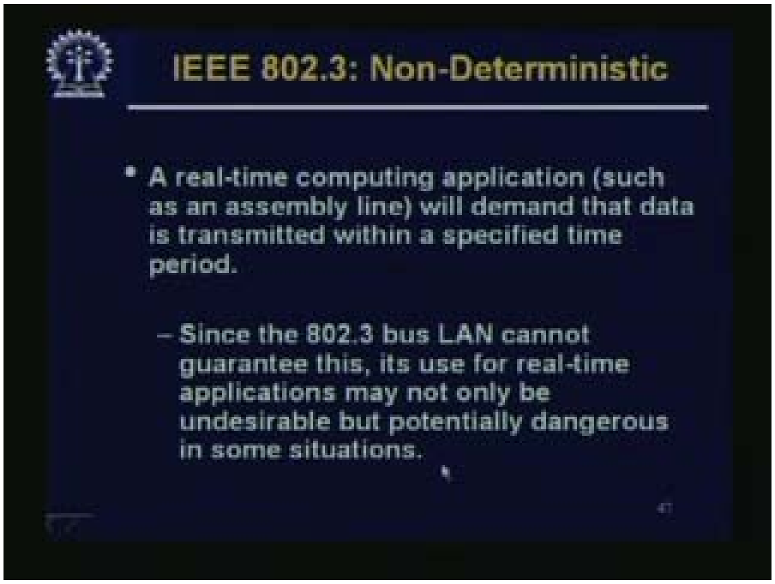
(Refer Slide Time: 53:53)



IEEE 802.3: Non-Deterministic

- The 802.3 CSMA/CD bus LAN is said to be a non-deterministic network. This means that no host is guaranteed to be able to send its frame within a reasonable time (just a good probability of doing so).
  - When the network is busy, the number of collisions rises dramatically and it may become very difficult for any hosts to transmit their frames.

(Refer Slide Time: 54:50)

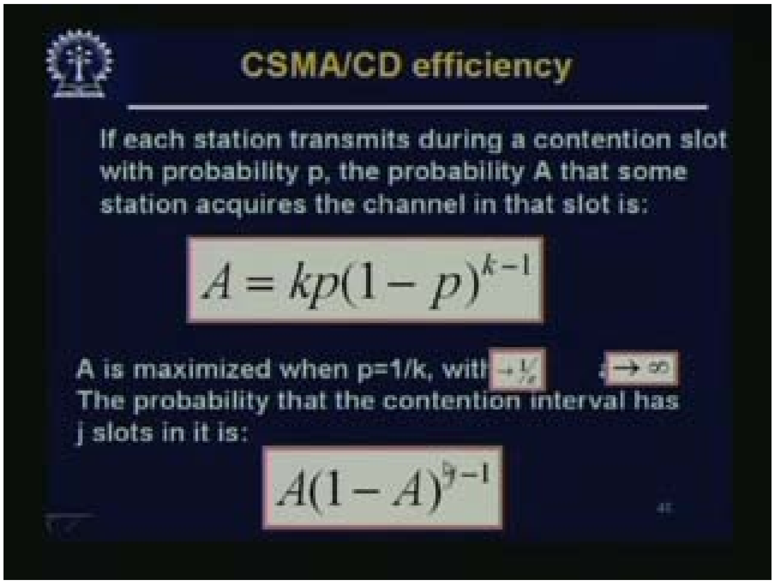


IEEE 802.3: Non-Deterministic

- A real-time computing application (such as an assembly line) will demand that data is transmitted within a specified time period.
  - Since the 802.3 bus LAN cannot guarantee this, its use for real-time applications may not only be undesirable but potentially dangerous in some situations.

This makes it not suitable for a real-time computing application. At least this bus type network may not be suitable for a real-time computing application such as an assembly line, which will demand that data be transmitted within a specified time period. Since the 802.3 bus LAN cannot guarantee this, its use for real-time applications may not only be undesirable but sometimes potentially dangerous. There are other ways of very high-speed network when this is low; then the chance of not being able to send at all becomes very low indeed.

(Refer Slide Time: 55:36)



CSMA/CD efficiency

If each station transmits during a contention slot with probability p, the probability A that some station acquires the channel in that slot is:

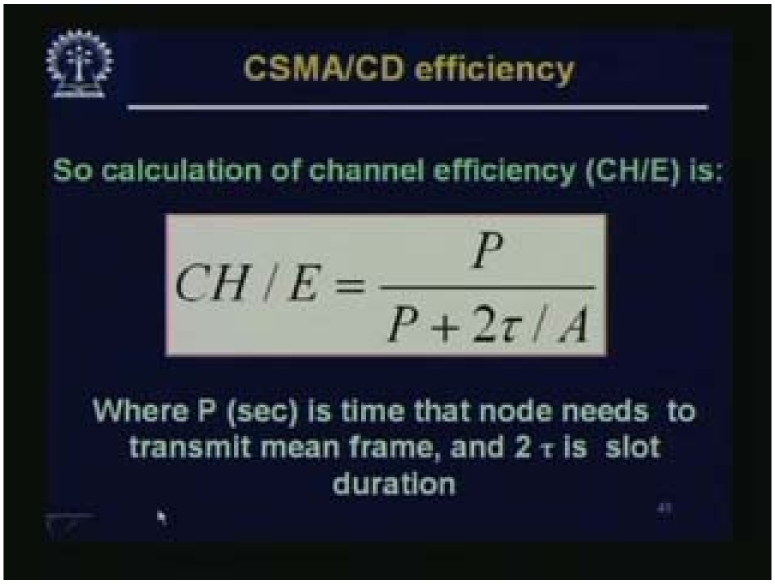
$$\boxed{A = kp(1-p)^{k-1}}$$

A is maximized when p=1/k, with $\to 1/e$ $\to \infty$
The probability that the contention interval has j slots in it is:

$$\boxed{A(1-A)^{j-1}}$$

Just one last word about the CSMA/CD efficiency. We will not go to the details of this discussion. Suppose each station transmits during a contention slot with probability p. The probability A that some station acquires the channel in that slot is A is equal to kp $(1 - p)^{k-1}$. So p is the probability that a particular contention slot is taken by one node and 1 − p is the probability that all the other k minus 1 nodes have not taken and that a particular slot is being used, i.e. kp $(1 - p)^{k-1}$. So A is maximized when p is equal to 1/k with a tending to 1/ε as k tends to ∞. The probability that the contention interval has j slots in it is A (1 minus A)$^{j-1}$.

(Refer Slide Time: 56:43)



CSMA/CD efficiency

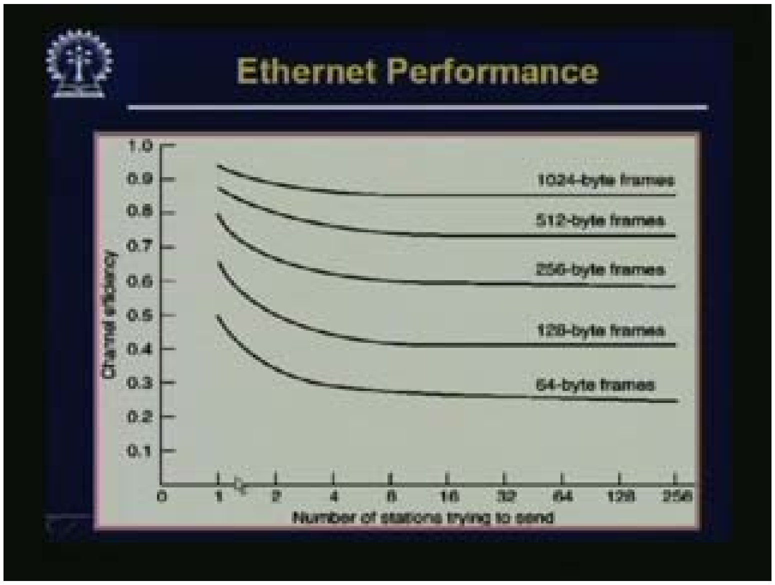
So calculation of channel efficiency (CH/E) is:

$$\boxed{CH / E = \frac{P}{P + 2\tau / A}}$$

Where P (sec) is time that node needs to transmit mean frame, and 2 τ is slot duration

The calculation of channel efficiency would be p, where p is the time that a node needs to transmit mean frame and 2 τ is slot duration so p/p plus 2τ/A means that if p is large, that means if we are sending a large frame, the channel efficiency increases.

(Refer Slide Time: 57:03)



Ethernet Performance

As you will see in this slide, with 1024-byte frames, the channel efficiency is very high. If you send in small frames, the channel efficiency tends to be smaller. This plot is with a number of stations trying to send. With this, we come to the end of this lecture. In the next lecture, we will look at the modern versions of Ethernet like switched Ethernet and how different networks are connected together. Thank you.