Data Link Layer Protocols

Data link layer protocols

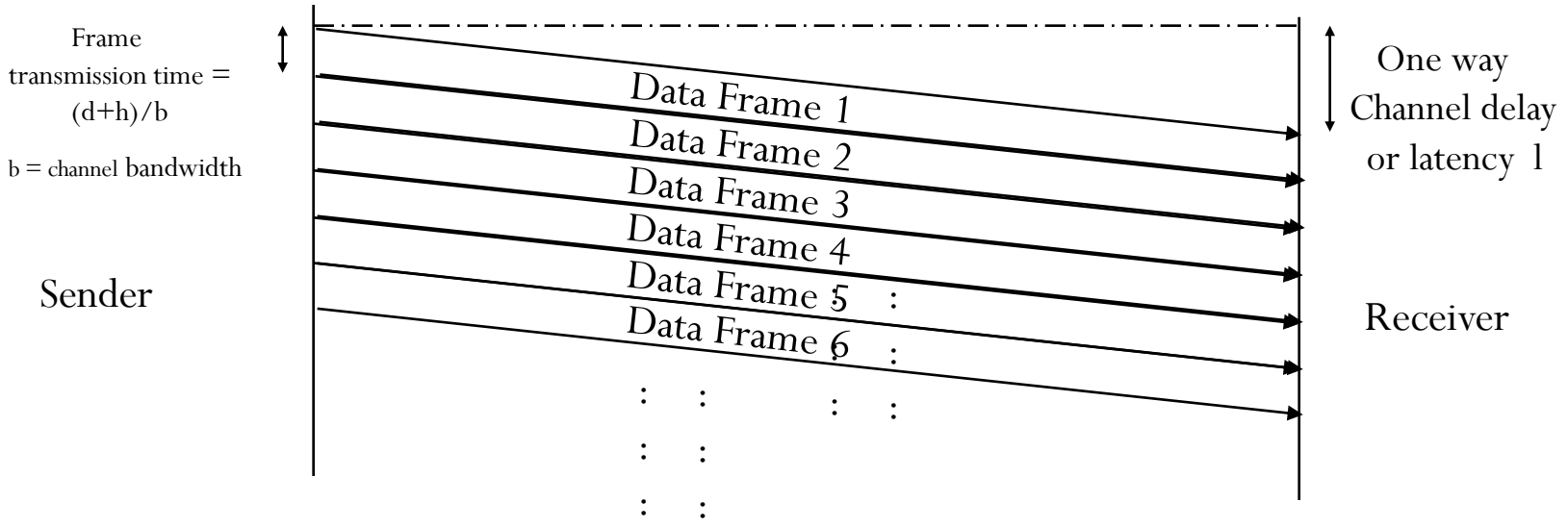
- Assumptions
- process of physical, datalink and network layers are independent.
- physical and data link layer process running in network I/O chip and network layer in main CPU.
- machine A want to transmits long stream of data to machine B using reliable and connection oriented service.
- machines don't crashes. → errors with protocols only.
-

Protocol 1: An Unrestricted Simplex Protocol

- **Data Transmission in one direction and processing time is ignored.**
- **The receiver is always ready to receive the next frame (has infinite buffer storage).**
- **Error-free communication channel. → unrealistic → utopia protocol**
- **No acknowledgments or retransmissions used.**
- **If frame has d data bits and h overhead bits, channel bandwidth b bits/second:**

  maximum channel utilization = data size/frame size = $d/(d+h)$

  maximum data throughput = $d/(d+h)$ * channel bandwidth = $d/(d+h)$ * b



Protocol - 2

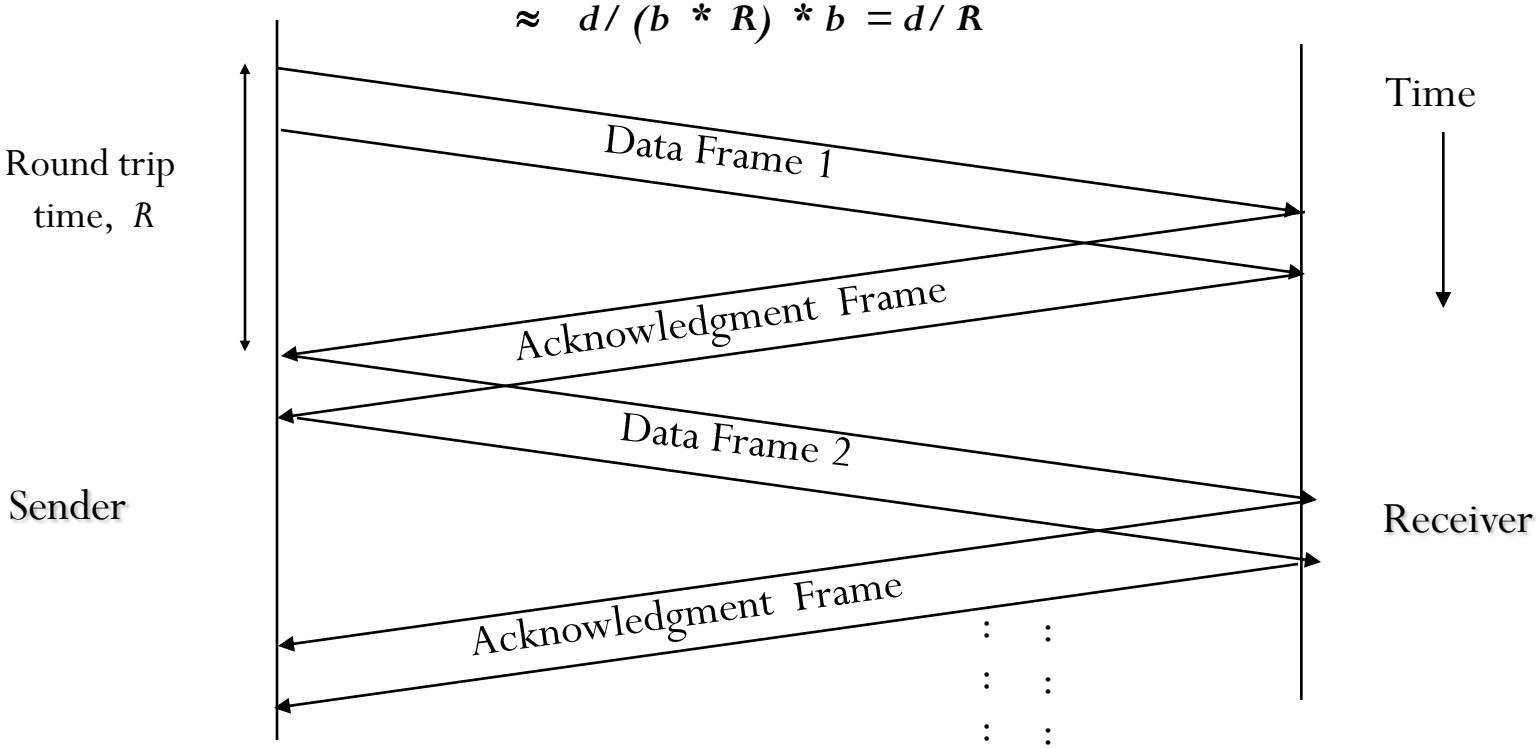
- Assumptions :
- drops the ability to receive large amount of data @ receiver side → both at data link layer and network layer.
- communication channel is error free.
- data traffic is simplex.
- problem : how to prevent the sender from flooding the receiver with data faster than the latte is able to process them.
- just put the delay at sender side of protocol – 1 → more time delay → inefficient. b/w utilization far from optimization. → not good idea.
- feedback from receiver to sender about data transmission.
- so sender sends one frame and then waits for an acknowledgment before processing is called STOP – AND – WAIT protocol.

Protocol #2: A Simplex Stop-and-Wait Protocol

- **The receiver may not be always ready to receive the next frame (finite buffer storage). Feedback .**
- **Receiver sends a positive acknowledgment frame to sender to transmit the next data frame.**
- **Half duplex channel is used.**
- **Maximum channel utilization ≈ (time to transmit frame /round trip time) * $d/(d + h)$**

  ≈ $d/(b * R)$
- **Maximum data throughput ≈ channel utilization * channel bandwidth**

  ≈ $d/(b * R) * b = d/R$



Protocol-3 → noisy channel

- normal communication channel → errors.
- frames either damaged or lost completely
- checksum used to detect the errors in frames. If it is unable to find the error, then this protocol fails.
- variation of protocol – 2:
- adding timer → for acknowledgment.
- sender waits after transmits frame → receiver sends ack if frame is ok, otherwise it will not send any ack.
- so timer expires @ sender and retransmits frame.
- problem → ack is lost → duplicate of frames @ receiver.

Protocol 3: A Simplex Positive Acknowledgment with Retransmission (PAR) Protocol or ARQ

- **Receiver sends a positive acknowledgment frame to sender to transmit the next data frame. Any frame has a sequence number, either 0 or 1**
- **Maximum utilization and throughput similar to protocol 2 when the effect of errors is ignored.**