Error and Flow Control

Go-Back-N Protocol

- Focus on which frames to retransmit
- Pipeline: send up to N frames before receiving and acknowledgment

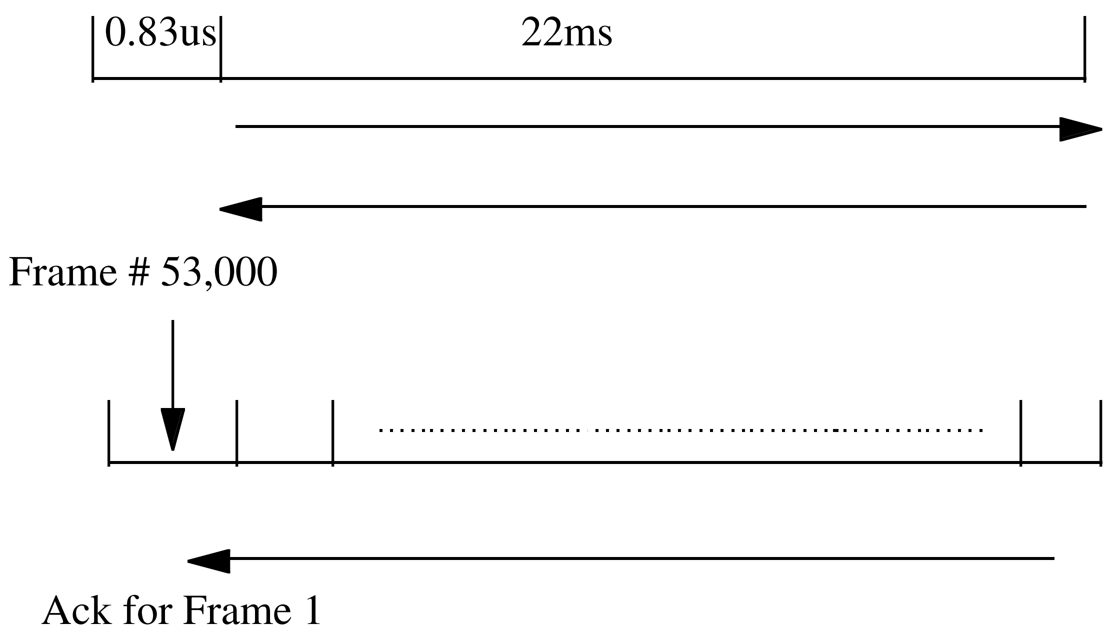
Error and Flow Control

Go-Back-N Protocol

- Example:
- Distance between nodes = 6600 km
- Frame length = 1000 bits
- Rate = 1.2Gb/s
  - Case 1: Stop and Wait
  - Frame transmission time = 0.83us
    Propagation time = 22 ms
  - Transmit frame at t=0, then at 0.83us + 22 ms frame received and
  - at 0.83us + 44ms the acknowledgment is received, therefore
  - the effective transmission rate is ~22.7kb/s or 0.002% efficient
  - Case 2: Pipeline 53,000 frames
  - Note with N=53,000 the first acknowledgment arrives at the transmitter just in time for the next frame to be transmitted. The transmitter is never blocked. The protocol is 100% efficient

Error and Flow Control

Go-Back-N Protocol



Error and Flow Control

Go-Back-N Protocol

- Problem:

  If there is an error or lost frame then what rules are used to determine the frames to retransmit.
- Go-back-N
  - Retransmit all frames transmitted after the errored frame
  - The receiver ignores all out-of sequence frames

  Example:

  Transmit 1,2,3,4,5 and

  frame 2 is in error then

  3, 4, and 5 are received out of sequence and

  retransmit 2,3,4,5

Error and Flow Control

Selective Repeat

- Receiver accepts out of sequence frames
- Requires buffers in receiver and transmitter
- Requires extra processing to deliver packets in order to the Network Layer

Error and Flow Control

Other Enhancements

- Negative Acknowledgment
  - When an out-of-sequence frame is received the receiver sends a **NAK** frame to the transmitter, the **NAK** frame contains the sequence number of the expected data frame.
  - **NAK** enables faster error recovery, without a **NAK** time-out must be used to learn about errors.

Error and Flow Control

Other Enhancements

- Acknowledgment timer
  - Reverse traffic is used to Piggyback **ACKS**
  - If there is light (or no) reverse traffic then **ACKS** may not be sent.
  - An acknowledgment timer is used to insure **ACKS** are sent.
  - Upon receipt of a frame an *AckTimer* is started. If reverse traffic arrives before the *AckTimer* fires then piggyback the **ACK**. If the *AckTimer* fires then send a supervisory ACK frame.