Stop and Wait Protocol

- Frame number to be included
- What is the minimum of bits required?
- ambiguity between m and m+1
  - 1 bit sequence number
- sender: knows which frame to send next

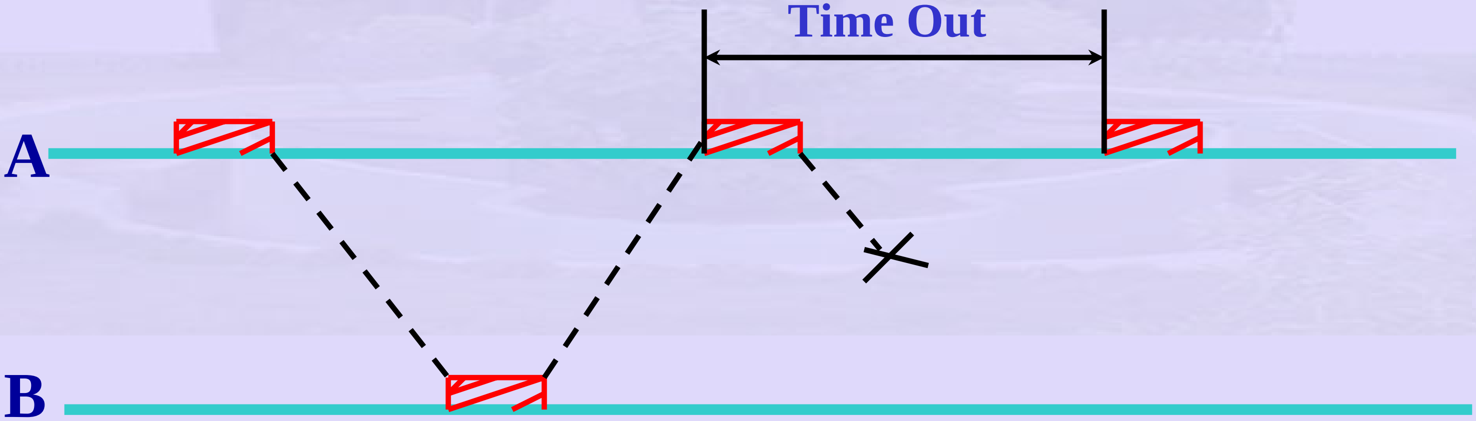
Stop and Wait Protocol

- receiver: knows which frame to expect next
- counters: incremented modulo 2
- Sending process:
- if event = DLL Send then
  - increment next FrameNo modulo 2

Stop and Wait Protocol

- Receiver Process:
- if event = DLLRecv then
- if recv.Seqnum = expected Seqnum then
- DLL State = receiving
- getFrameFrom PhysLayer(r, buffer)
- Sent To NWL(buffer)
- increment NextFrame Expected modulo 2

Stop and Wait: Timing diagram



ThroughPut

- Error Free Case: Throughput is :

$$U = \frac{T_f}{T_t}$$

T_f - Time take to transmit a frame

T_t - Total time engaged in the transmission of a frame

$$T_t = T_f + T_{prop} + T_{ack} + T_{proc} + T_{prop}$$

Example

- Error free case:
  - Frame size = 10 KB
  - RTT = 100ms = 0.1s
  - Bandwidth = 1 Mbps

$$T_f = 10 \times 8 \times 1024 / (10^6)$$
$$= 0.08192$$
$$T_f + 0.1 = 0.18912$$
$$U = \frac{0.08192}{0.18912} = 0.43$$
$$Throughput = 430 \; kbps$$

Errors in transmission

- Let N_r = E [number of retransmissions]

$$U = \frac{T_f}{N_r T_t}$$

Stop and Wait: Analysis

T_{prop} -- is propagation delay

T_{ack} -- time take for acknowledgement

T_{proc} -- time taken for processing at the receiver

If T_{ack}, T_{proc} are negligible then

$$U = \frac{1}{1+2a}, a = T_f / T_p$$

Expected Number of Retransmissions

$$N_r = \sum_{i=1}^{\infty} iP_r[i \text{ transmissions}]$$

$$= \sum_{i=1}^{\infty} iP^{i-1}(1-P)$$

$$= \frac{1}{1-P}$$

$$U = \frac{(1-P)}{1+2a}$$

where P is the probability of a frame being in error

Error Analysis

Let p be the probability that a bit is in error

Let F be the number of bits in a frame

$$P = 1 - (1 - p)^F$$