ERROR CORRECTION

Any time an error is detected in an exchange, specified frames are retransmitted. This process is called automatic repeat request (ARQ). *Error control in the data link layer is based on automatic repeat request, which is the retransmission of data.*

Three versions of ARQ have been standardized:

Stop-and-wait ARQ
Go-back-N ARQ
Selective-reject ARQ

Stop-and-wait ARQ:

Stop-and-wait ARQ is based on the stop-and-wait flow-control technique outlined previously and is depicted in Figure alongside.

- The source station transmits a single frame and then must await an acknowledgement (ACK). No other data frames can be sent until the destination stations reply arrives at the source station.
- The sending device keeps a copy of the last frame until it receives an acknowledgement for that frame. Keeping a copy allows the sender to retransmits lost or damaged frames until they are received correctly.
- For identification purposes, both data frames and acknowledgement frames (ACK) are numbered o & 1. A data 0 frame is acknowledged by and ACK1 frame, indicating that the receiver has received data frame 0 and is now expecting data frame 1.
- The sender starts a timer when it sends a frame. If an acknowledgement is not received within an alloted time period, the sender assumes that the frame was lost or damage and resends it.

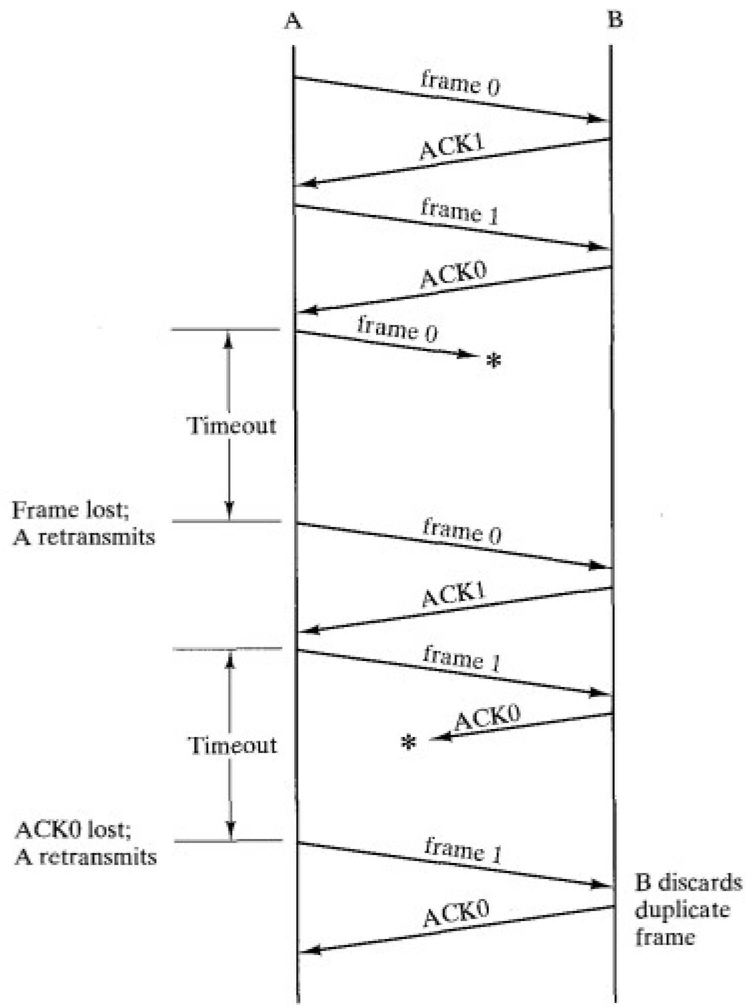

Fig.Stop and Wait ARQ

- The receiver sends only +ve ACK for frame received safe and sound. It is silent about the frames damaged or lost. The acknowledgement number always define the number of next expected frame. If frame 0 is received, ACK1 is sent; if frame 1 is received ACK 0 is sent.

Bidirectional Transmission:

The stop-and-wait mechanism we have discussed is unidirectional. However, we can have bi-directional transmission if the two parties have two separate channels for the full-duplex transmission or share the same channel for half-duplex transmission.

Piggybacking: is a method to combine a data frame with an acknowledgement. For example, stations A and B both have data to send. Instead of sending separate data and ACK frames, Station A sends a data frame that includes an ACK , station B behaves in a similar manner.

Piggybacking can save BW because the overhead from a data frame and ACK frame (addresses, CRC, etc) can be combined into just one frame.

Go-Back-N ARQ:

- The form of error control based on sliding-window flow control that is most commonly used is called go-back-N ARQ.

- In go-back-N ARQ, a station may send a series of frames sequentially numbered modulo some maximum value.

- The number of unacknowledged frames out-standing is determined by window size, using the sliding-window flow control technique.

- While no errors occur, the destination will acknowledge (RR = receive ready) incoming frames as usual.

- If the destination station detects an error in a frame, it sends a negative acknowledgment (REJ = reject) for that frame. The destination station will discard that frame and all future incoming frames until the frame in error is correctly received. Thus, the source station, when it receives an REJ, must retransmit the frame in error plus all succeeding frames that were transmitted in the interim.

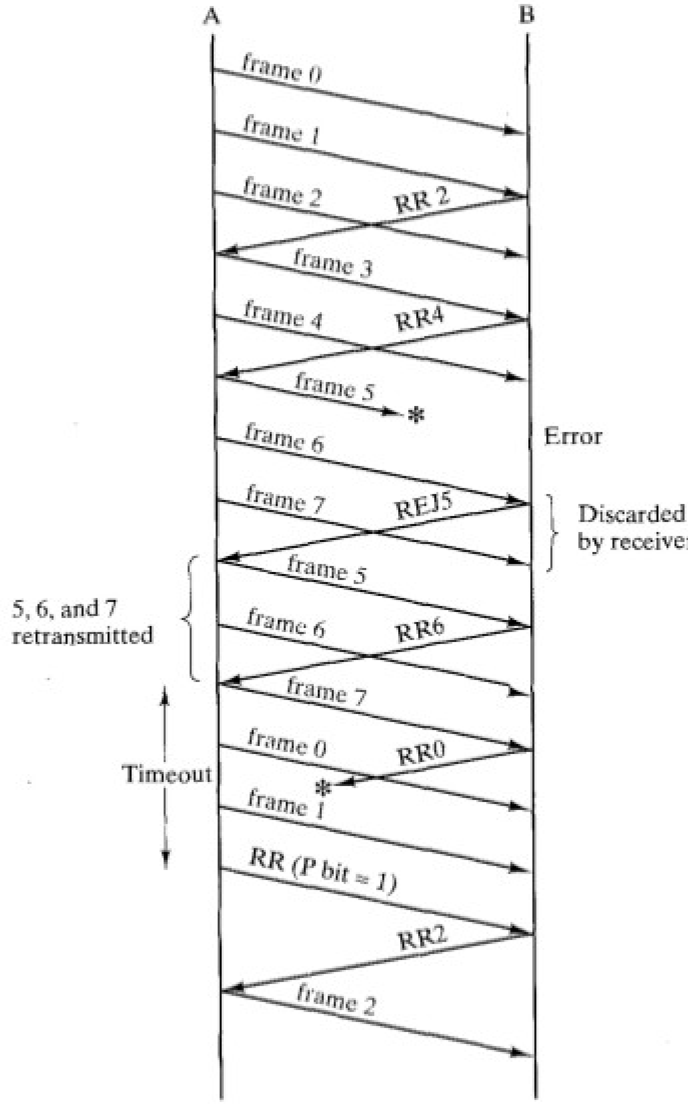

Fig:Go-Back-N ARQ

Consider that station A is sending frames to station B. After each transmission, A sets an acknowledgment timer for the frame just transmitted. The go-back-N technique takes into account the following contingencies:

1. Damaged frame. There are three sub-cases:

a) A transmits frame i. B detects an error and has previously successfully received frame (i - 1). B sends REJ i, indicated that frame i is rejected. When A receives the REJ, it must retransmit frame i and all subsequent frames that it has transmitted since the original transmission of frame i.

b) Frame i is lost in transit. A subsequently sends frame (i + 1). B receives frame (i + 1) out of order and sends an REJ i. A must retransmit frame i and all subsequent frames.

c) Frame i is lost in transit, and A does not soon send additional frames. B receives nothing and returns neither an RR nor an REJ. When A's timer expires, it transmits an RR frame that includes a bit known as the P bit, which is set to 1.B interprets the R R frame with a P bit of 1as a command that must be acknowledged by sending an R R indicating the next frame that it expects. When A receives the RR, it retransmits frame i.

2. Damaged RR. There are two sub-cases:

a) B receives frame i and sends R R (i + I), which is lost in transit. Because acknowledgments are cumulative (e.g., RR 6 means that all frames through 5 are acknowledged), it may be that A will receive a subsequent R R to a subsequent frame and that it will arrive before the timer associated with frame i expires.

b) If A's timer expires, it transmits an RR command as in Case lc. It sets another timer, called the P-bit timer. If B fails to respond to the RR command, or if its response is damaged, then A's P-bit timer will expire. At this point, A will try again by issuing a new RR command and restarting the P-bit timer. This procedure is tried for a number of iterations. If A fails to obtain an acknowledgment after some maximum number of attempts, it initiates a reset procedure.

3. Damaged REJ. If an REJ is lost, this is equivalent to Case lc.

Selective-reject ARQ

With selective-reject ARQ, the only frames retransmitted are those that receive a negative acknowledgment, in this case called SREJ, or that time-out. This would appear to be more efficient than go-back-N, because it minimizes the amount of retransmission. On the other hand, the receiver must maintain a buffer large enough to save post-SREJ frames until the frame in error is retransmitted, and it must contain logic for reinserting that frame in the proper sequence. The transmitter, too, requires more complex logic to be able to send a frame out of sequence. Because of such complications, select-reject ARQ is much less used than go-back- N ARQ.

Source : http://dayaramb.files.wordpress.com/2011/03/computer-network-notes-pu.pdf