Ch3: **1: Services Provided to the Network Layer, Framing (Byte count Flag bytes with byte stuffing, Flag bits with bit stuffing, Physical layer coding violations), Error control, Flow control.** **Error Control**: Ensuring all frames are eventually delivered:{To the network layer at the destination, In the proper order}; Ensures reliable, connection-oriented service; Requires acknowledgement frames and timers. **Flow control**: Controlling the sending of transmission frames at a faster pace than they can be accepted; Feedback-based flow control: {Receiver sends back information to the sender giving it permission to send more data, Or receiver tells the sender how the receiver is doing}; Rate-based flow control: {Protocol has a built-in mechanism, Mechanism limits the rate at which senders may transmit data, No feedback from the receiver is necessary}.**2: Error-Correcting Codes: (Hamming codes, Binary convolutional codes, Reed-Solomon codes, Low-Density Parity Check codes), Error-Detecting Codes: (Parity, Checksums, Cyclic Redundancy Checks)** **3: Three simplex link-layer protocols: (Utopia: No Flow Control or Error Correction, Adding Flow Control: Stop-and-Wait, Adding Error Correction: Sequence Numbers and ARQ).** **Elementary Data Link Protocols** Assumptions underly the communication model: Three simplex link-layer protocols{Utopia: No Flow Control or Error Correction, Adding Flow Control: Stop-and-Wait, Adding Error Correction: Sequence Numbers and ARQ}**4: Piggybacking, three bidirectional sliding window protocols: (One-bit sliding window, go-back-n, selective repeat)** Bidirectional transmission: piggybacking {Use the same link for data in both directions, Interleave data and control frames on the same link, Temporarily delay outgoing acknowledgements so they can be hooked onto the next outgoing data frame}; Piggybacking advantages: {A better use of the available channel bandwidth, Lighter processing load at the receiver}; Piggybacking issue{Determining time data link layer waits for a packet to piggyback the acknowledgement}; /* Protocol 1 (utopia) provides for data transmission in one direction only, from sender to receiver. The communication channel is assumed to be error free, and the receiver is assumed to be able to process all the input infinitely fast. Consequently, the sender just sits in a loop pumping data out onto the line as fast as it can. // Protocol 2 (stop-and-wait) also provides for a one-directional flow of data from sender to receiver. The communicaiton channel is once again assumed to be error free, as in protocal 1. However, this time, the receiver has only a finite buffer capacity and a finite processing speed, so the protocol must explicitly prevent the sender from flooding the receiver with data faster than it can be handled. // Protocol 3 (par) allows unidirectional data flow over an unreliable channel // Protocol 4 (sliding window) is bidirectional and is more robust than protocol 3. // Protocol 5 (Go-back-n) allows multiple outstanding frames. The sender may transmit up to MAX_SEQ frames without waiting for an ack. In addition, unlike the previous protocols, the network layer causes a network_layer_ready event when there is a packet to send // Protocol 6 (Selective repeat) accepts frames out of order but passes packets to the network layer in order. Associated with each outstanding freme is a timer. When the timer expires, only that frame is retransmitted, not all the outstanding frames, as in protocol 5. */In go-back-n, if a frame in the middle of a long stream is damaged or lost, the receiver just discards all subsequent frames, sending no acknowledgements for the discarded frames. This strategy corresponds to a receive window of size 1. In other words, the data link layer refuses to accept any frame except the next one it must give to the network layer. If the sender's window fills up before the timer runs out, the pipeline will begin to empty. Eventually, the sender will time out and retransmit all unacknowledged frames in order, starting with the damaged or lost one. This approach can waste a lot of bandwidth if the error rate is high. An alternative strategy, the selective repeat protocol, is to allow the receiver to accept and buffer correct frames received following a damaged or lost one. When it is used, a bad frame that is received is discarded, but any good frames received after it are accepted and buffered. When the sender times out, only the oldest unacknowledged frame is retransmitted. If that frame arrives correctly, the receiver can deliver to the network layer, in sequence, all the frames it has buffered. Selective repeat corresponds to a receiver window larger than 1. This approach can require large amounts of data link layer memory if the window is large.

<u>5: Packet over SONET, ADSL</u> Data Link Protocols in Practice: {Packet over SONET, ADSL (Asymmetric Digital Subscriber Loop)}