1. The following character encoding is used in a data link protocol: A: 01000111 B: 11100011 FLAG: 01111110 ESC: 11100000 Show the bit sequence transmitted(in binary) for the four-character frame A B ESC FLAG when each of the following framing methods is used:
   (a) Byte count.
   (b) Flag bytes with byte stuffing.
   (c) Starting and ending flag bytes with bit stuffing.

Solution:
(a) 00000101 01000111 11100011 11100000 01111110
(b) 01111110 01000111 11100011 11100000 11100000 11100000 01111110 01111110
(c) 01111110 01000111 110100011 111000000 011111010 01111110

2. Hamming code is an effective way for error correcting. Show that the # of check bits(i.e. r) in the Hamming codes described in the textbook(e.g., Fig.3-6) (almost) achieves the low bound of Eq (3-1).

Solution:
Assuming that the number of check bits is r.
According to hamming codes, the number of data bits m is between 2^(r-1)+1-r bits and 2^r – r-1 bits.
So 2^(r-1)+2<r+m+1<2^r . The number of check bits in the hamming codes (almost) achieves the low bound of Eq(3-1).
In Figure 3.6, 4+7+1<2^4, so 4 bits are enough.

3. Suppose you have the following 12-bit message: 010100111111
   (a) Numbering bits from right to left (ie least-significant bit on the right), insert check bits according to to Hamming's 1-bit error correction system. Indicate which bits are check bits and which are message bits.
   (b) Hamming's scheme only corrects 1-bit errors. Since it' s a distance 3 code, it could also be used to detect 2-bit errors. Describe a 3-bit error (3 *1-bit errors) in the above codeword affecting only message bits (not check bits) that would be undetected (and of course uncorrected). Be sure to describe how and why the algorithm fails.

Solution:
(a) First, how many checkbits do we need: to satisfy m+r+1<=2^r , with m=12 we need r=5 checkbits.

bit	17	**16**	15	14	13	12	11	10	9	**8**	7	6	5	**4**	3	**2**	**1**
value	0	**0**	1	0	1	0	0	1	1	**0**	1	1	1	**1**	1	**1**	**0**

Checkbits above are shown in bold, and are calculated:

bit 3:	0 0 0 1 1
bit 5:	0 0 1 0 1
bit 6:	0 0 1 1 0
bit 7:	0 0 1 1 1
bit 9:	0 1 0 0 1
bit 10:	0 1 0 1 0
bit 13:	0 1 1 0 1
bit 15:	0 1 1 1 1
sum mod-2:	0 0 1 1 0

（b）. In order for the error to be undetected the check bits must remain valid. Thus the 3 bits flipped must be such that the sum of their bit positions (mod-2) is 0. There are many such triplets; e.g., flipping bit 3, 5 and 6:

bit 3:	0	0	0	1	1
bit 5:	0	0	1	0	1
bit 6:	0	0	1	1	0
sum mod-2:	0	0	0	0	0

4. Consider an original frame 110111011011. The generator polynomial x4+x+1, show the converted frame after appending the CRC.

Solution:

First append 4 zeros to original frame, then we get 1101110110110000

Generator is 10011, using modulo 2 subtraction

The answer is 110110110110101

5. A 3000-km-long T1 trunk (with data rate 1.536Mbps) is used to transmit 64-byte frames. How many bits should the sequence numbers be for protocol 5 and protocol 6 respectively? The propagation speed is 6usec/km.

Solution:

To operate efficiently, the sequence space (actually, the sender's window size) must be large enough to allow the transmitter to keep transmitting until the first ack has been received. The propagation is 18 msec. At T1 speed, which is 1.536Mbps(excluding the 1 header bit), a 64-byte frame takes 0.300 msec. Therefore, the first frame fully arrives 18.3ms after transmission was started. The ack takes another 18ms to get back, plus a small (negligible) time for the ack to arrive fully. In all, this time is 36.3 msec. A frame takes 0.3ms, so it takes 121 frames to fill the pipe.

For protocol 5, 7-bit sequence number are needed.

For protocol 6, 8-bit sequence number are needed.

6. Frames of 1000 bits are sent over a 1-Mbps channel using a geostationary satellite whose propagation time from the earth is 270 msec. Acknowledgements are always piggybacked onto data frames. The headers are very short. Three-bit sequence numbers are used. What is the maximum achievable channel utilization for

   (a) Stop-and-wait?

   (b) Protocol 5?

   (c) Protocol 6?

Solution:

Let t =0 denote the start of transmission. At t =1 msec, the first frame has been fully transmitted. At t =271 msec, the first frame has fully arrived. At t =272 msec, the frame acknowledging the first one has been fully sent. At t =542 msec, the acknowledgement-bearing frame has fully arrived. Thus, the cycle is 542 msec. A total of k frames are sent in 542 msec, for an efficiency of k/542. Hence, for

(a) k = 1, efficiency = 1/542 = 0.18%.

(b) k = 7, efficiency = 7/542 = 1.29%.

(c) k = 4, efficiency = 4/542 = 0.74%.

7. Compute the fraction of the useful data bandwidth for protocol 6 on a heavily loaded 50-kbps satellite channel with data frames consisting of 40 header and 3960 data bits. Assume that the signal propagation time from the earth to the satellite is 270 msec. ACK frames never occur. NAK frames are 40 bits. The error rate for data frames is 1%, and the error rate for NAK frames is negligible. The sequence numbers are 3 bits.

Solution:

The sequence number are 3 bits, then the windows size is 4

4000 bits frame takes 80ms and 40 bits NAK takes 0.8ms.

The utilization is 4*80/(270*2+80+0.8)=51.5%

The number of retransmissions per frame is about 0.01. Each good frame wastes 40 header bits, plus 1% of 4000 bits (retransmission), plus a 40-bit Nak once every 100 frames. The total overhead is 80.4 bits per 3960 data bits, so the wasted on overhead is 80.4/(3960+80.4=1.99%

So the fraction of the useful data bandwidth is 51.5%*(1-1.99%)=50.5%