**1) Overview**
  a) (c)
  b) (a)
  c) (d)
  d) (b)
  e) (d)
  f) (b)
  g) (a)
  h) (d)
  i) (b)
  j) (a)

**2) Nyquist and Shannon Limits Reconsidered**
  a)  We take two previous bit and current bit because each bit lasts for 3 usec.

| Combination | Voltage | Output Bit |
|---|---|---|
| 000 | 0 | 0 |
| 001 | 4 | 1 |
| 010 | 2 | 0 |
| 011 | 6 | 1 |
| 100 | 1 | 0 |
| 101 | 5 | 1 |
| 110 | 3 | 0 |
| 111 | 7 | 1 |

  b)  Rule: If voltage > 3.5, output 1; else output 0
  c)  (I assume given the above rule, even with ISI, we can still keep sending bit at every 1 usec and still correctly detect the current bit)
  d)  If the noise amplitude is 0.5V, one can encode 'symbols' of 3 bits as in table above into 8 different levels as in Shannon and send a symbol at every 3 usec.

**3) CRCs**
  a)  The size of burst error of the frame is 7. Three consecutive errors in the Line 0 cause errors on the 1st, 4th, and 7th bits in the frame. Thus, the size of the burst error in the frame is 7.

b) Since the size of the frame burst error is 7 which is smaller than the size of CRC-8, the error will be always detected. In other words, there is always remainder after dividing with the CRC-8 generator.

c) The four consecutive bit errors in the Line 0 will make the error polynomial in the frame as $x^9 + x^6 + x^3 + 1$.
$(x^9 + x^6 + x^3 + 1) / (x^8 + x^6 + x^4 + x^2 + 1) = x +$ (Remainder: $x^7 + x^6 + x^5 + x + 1$).
Remainder is 11100011. (Not divisible) Thus, the answer is 'Yes'. The receiver will catch the error using CRC-8.

## 4) Error Recovery

a) Change A(1) in first case to A(1: 2) (i,e. I need 1, but have received 2) and then the second Ack to A(1; 2, 3) ( Need 1 but have received 2 and 3).  Also, buffer at receiver

b) Yes, it solves the single packet loss case without retransmitting the entire window but it is very slow to react to a loss of multiple packets in a window.  If an entire window W is lost one needs W roundtrips to recover.  So improved throughtput for 1 error case compared to go-back N but much worse latency for multiple errors than go-back N

c) The advantage of the new scheme is that it greatly reduces latency for a single error case (as soon as you get 3 duplicate acks instead of waiting for a timeout) 'while being as good as Go-Back N in terms of throughput and latency for the multiple error case