2. However, using k bit sequence numbers, we can generate 2^k sequences.

So, we can keep our window size to be 2^k.

But it can cause issue while error control.

Say, sender sends frame 0 and gets back RR 1

Then it sends frame 1, 2, ... $2^k - 1$ and frame 0.

Then it gets another ~~RR 0~~ RR 1.

But sender will not be able to know whether all the 2^k frames were recieved correctly or all of these were lost because in both the cases RR 1 will be the acknowledgment recieved.

In 1st case, cumulative one & in 2nd case, the repeated one.

This can be solved by restricting the window size to be $2^k - 1$.

As now it cannot send 2^k messages at a time,

Thus it will get different acknowledgements in case of error and cumulative acknowledgement.

Ex- In previous case,

i) If all the $2^k - 1$ message get lost, ~~the~~ RR1 will be ~~the~~ sent.

ii) If all $2^k - 1$ messages are recieved, i.e. from frame 1 to frame 2^k, The cumulative acknowledgement will be RR0.

Hence no ambiguity.

That is the reason window size is $2^k - 1$.

3. Here we will have window size of 2^{k-1} for k-bit sequence numbers.

We don't want window of reciever to overlap with that of sender if there is any loss of acknowledgement ~~as in~~ because Selective-Reject ARQ unlike of Go-back-N accepts any frame that is in window of reciever & buffers it.

So, we must need a ~~be~~ window size of 2^{k-1}.

As, if we take ~~a~~ more than this, say $2^{k-1}+1$.

Then, first ~~recie~~ sender sends frame 0, 1, ... 2^{k-1} which gets accepted by reciever & reciever sends an acknowledgement but it ~~gest~~, gets lost,

By then reciever has advanced its window to recieve from frame $2^{k-1}+1$ upto frame 0.

Now, reciever after timer again sends from frame 0. ~~upto frame 2^{k-1}~~

Here frame 0 will be accepted and buffered.

But this should not be the case.

Hence window size of 2^{k-1}.

4. For Stop & wait flow control,

$$U = \frac{1}{2a+1} \quad \text{where } a = \frac{t_{prop}}{t_{frame}}$$

Given, $U \geqslant 0.5$

$\Rightarrow 1+2a \leq 2$

$\Rightarrow 2a \leqslant 1$

$\Rightarrow a \leqslant 1/2$

Thus, $\dfrac{t_{prop}}{t_{frame}} \leqslant 1/2$

$\Rightarrow t_{frame} \geqslant 2 \cdot t_{prop}$

$\Rightarrow t_{frame} \geqslant 40\ ms$

$\Rightarrow \dfrac{\text{frame size}}{\text{data rate}} \geqslant 40\ ms$

$\Rightarrow$ frame size $\geq 40\ ms \times 4\ kbps$

$\Rightarrow$ frame size $\geqslant 0.16\ kb$

5. Say, prob of error of 1 bit is p.

Then prob of no error at 1 pos $= (1-p)$.

If recieved frame has no error,
then all 4 bits should be errorless.

Hence, prob $= (1-p)^4$

$$= \left(1 - 1/1000\right)^4$$

$$\approx 0.996.$$

(b) Prob of atleast 1 error $= 1 -$ prob of no errors

$$= 1 - 0.996$$

$$\approx 0.004$$

(c) We can have 2 cases,

(i) Parity bit has error.

(ii) Parity bit is correct.

Total prob that error is not detected

~~= prob of error at parity~~

= prob (error in parity bit) × prob$_1$

+

prob (no error in parity bit) × prob$_2$.

Here prob 1 & prob 2 represents the respective probabilities such that error in not detected.

In case (1), parity bit has error, hence there should be 1 or 3 number of errors in rest of 4 bits.

If even parity → Ex → $\boxed{01100}$ → $\boxed{\quad\quad\quad 1}$ ~~Here~~

Actual Erroneous.

Here, now we should have odd number of 1's in

~~as parity is changed to odd.~~

the rest 4 bits to maintain even parity.

which results in 1 or 3 num of errors.

Thus prob 1 $= {}^4C_3(1-p)^3 p + {}^4C_1(1-p)\,p^3$

Similarly in Case II,

Parity bit has no error, hence there should be even number of bit changes (errors) to accept the erroneous message i.e. 2 or 4 number of bits.

Thus prob 2 $= {}^4C_2(1-p)^2 p + {}^4C_4 p^4$

Total ans $= \frac{1}{1000} \times$ prob 1 $+ \frac{999}{1000} \times$ prob 2

$$\approx \frac{1}{1000} \times 0.004 + \frac{999}{1000} \times 6 \times 10^{-6}$$

$$\approx \frac{0.004 + 0.006}{1000}$$

$$\approx 10^{-5}$$

6) First we will append M with size(P) - 1 number of 0s.

So M = 1110001100000

Now,

```
          10110110
110011 ) 1110001100000
         110011
         ------
          010111
          000000
          ------
           101111
           110011
           ------
            111000
            110011
            ------
             010110
             000000
             ------
              101100
              110011
              ------
               011110
               110011
               ------
                011010
                000000
                ------
                 11010
                 -----
```

Codeword : $\boxed{11100011}\boxed{11010}$

7. $D(x) = x^{10} + x^7 + x^4 + x^3 + x + 1$

$P(x) = x^4 + x + 1$

Now, we will multiply $D(x)$ by x^4 (highest degree of P(x

$$
\begin{array}{r|l}
 & x^{10} + x^6 + x^4 + x^2 \\
\hline
x^4+x+1 & x^{14} + x^{11} + x^8 + x^7 + x^5 + x^4 \\
 & x^{14} + x^{11} + x^{10} \\
\hline
 & x^{10} + x^8 + x^7 \\
 & x^{10} + x^7 + x^6 \\
\hline
 & x^8 + x^6 + x^5 + x^4 \\
 & x^8 + x^5 + x^4 \\
\hline
 & x^6 \\
 & x^6 + x^3 + x^2 \\
\hline
 & x^3 + x^2 \leftarrow R(x)
\end{array}
$$

So codeword in polynomial

$= x^{14} + x^{11} + x^8 + x^7 + x^5 + x^4 + \underbrace{x^3 + x^2}_{R(x)}$

which is

$\boxed{100100110111100}$

b) Following the error pattern, recieved code will be.

D = 000110110111100

$$D(x) = x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{3} + x^{2}$$

$$P(x) = x^4 + x + 1$$

$$x^7 + x^6 + x^3 + x^2 + x$$

$$x^4 + x + 1 \,\big)\, x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{3} + x^{2}$$

$$x^{11} + x^{8} + x^{7}$$

$$x^{10} + x^{5} + x^{4}$$

$$x^{10} + x^{7} + x^{6}$$

$$x^{7} + x^{6} + x^{5} + x^{4} + x^{3}$$

$$x^{7} + x^{4} + x^{3}$$

$$x^{6} + x^{5} + x^{2}$$

$$x^{6} + x^{3} + x^{2}$$

$$x^{5} + x^{3}$$

$$x^{5} + x^{2} + x$$

(Non Zero). $x^3 + x^2 + x$

As Remainder is non, zero. It will not be accepted.

(c)

$D = 000010110111100$

$D(x) = x^{10} + x^8 + x^7 + x^5 + x^4 + x^3 + x^2$

$P(x) = x^4 + x + 1$

$$\begin{array}{r l}
 & x^6 + x^4 + x^2 \\
x^4 + x + 1 \,) & x^{10} + x^8 + x^7 + x^5 + x^4 + x^3 + x^2 \\
 & x^{10} + x^7 + x^6 \\
\hline
 & x^8 + x^6 + x^5 + x^4 \\
 & x^8 + x^5 + x^4 \\
\hline
 & x^6 + x^3 + x^2 \\
 & x^6 + x^3 + x^2 \\
\hline
 & 0 \\
\hline
\end{array}$$

In this case, Remainder is 0

Hence, this erraneous codeword will be accepted.