Homework 2: CS425

Computer Networks

Abhishek Pardhi

200026

B.Tech students

apardhi20@iitk.ac.in

Indian Institute of Technology
Kanpur

Department of Computer Science & Engineering

February 16, 2023

Contents

1 Problem 1

Just run the code attached to this report. The results will be printed out in the terminal. Here, D is the original frame, P is the CRC pattern, T is the CRC code, and T_{error} is the received frame.

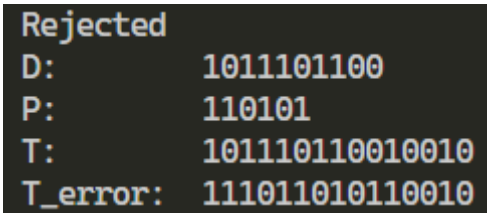

Figure 1: CRC

2 Problem 2

In the Go-Back-N ARQ mechanism using k-bit sequence numbers, the window size is limited to $2^k - 1$ and not 2^k because the sequence numbers are represented using k bits, which means that there are 2^k possible sequence numbers.

Now consider a situation where the window size is 2^k and the sender sends frame 0 and gets back an $RR1$. Then it sends frame $1, 2, 3, \ldots, 2^{k-1}, 0$ and gets another $RR1$. Now after receiving the $RR1$ call, the sender might think that all the 2^k frames were received correctly and the $RR1$ is a cumulative acknowledgment, or it can also think that all the 2^k frames were lost in transit and the receiving station is repeating its previous $RR1$ call.

To avoid this scenario, we take the maximum window size to be 2^{k-1} because then the RR calls will be different in the two cases mentioned above.

3 Problem 3

In the Selective-Reject ARQ mechanism that uses k-bit sequence numbers, the maximum window size that can be used is 2^{k-1}.

Let's consider an SRARQ mechanism using 3-bit sequence numbers with a window size of 3. First, the sender sends frame $0, 1, 2$ to the receiver. Now, suppose the frame 0 got lost, so the sender will send frame 0 again to the receiver along with the frames of the next window that are $3, 0, 1$. But this will lead to duplicated transmission of frame 0. To avoid such cases, we limit the window size to half the number of distinct frames available because doing so will ensure that the next cycle of frames will be different from any of the frames from the previous cycle.

Therefore, the maximum window size that can be used in the Selective-Reject ARQ mechanism that uses k-bit sequence numbers is 2^{k-1}.

4 Problem 4

Efficiency $U = \frac{1}{1+2a}$ where $a = \frac{Propagation\ Time}{Transimission\ Time}$ and $Transimission\ Time = \frac{Frame\ size}{Data\ rate}$
Plugging in the values, we get:
$\Rightarrow \frac{1}{1+2\times20\times10^{-3}\times\frac{4\times10^3}{FS}} \geq 0.5$
$\Rightarrow FS \geq 160$
Therefore, Frame size needs to be greater than or equal to $\boxed{160}$ bits.

5 Problem 5

Let $p = 10^{-3}$ be the probability of getting a bit error and the frame size is given to be 4.

1. Probability that the received frame contains no error is $\boxed{(1-p)^4 = 0.996}$
2. Probability that the received frame contains at least one error is $\boxed{1-(1-p)^4 = 0.004}$
3. Now that one parity bit is added to the frame, it can also get an error along with the original frame with a probability of p. Consider two cases for a given input stream:
   - If the parity bit is not flipped, then any even number of bit flips in the first four bits will make the error go undetected.
   - If the parity bit is flipped, then any odd number of bit flips in the first four bits will make the error go undetected.
   - Alternatively, if there are an even number of bit flips in the 5-bit stream then the error won't get detected. This will occur with a probability of
     $\Rightarrow \mathcal{P} = \binom{5}{2}p^2(1-p)^3 + \binom{5}{4}p^4(1-p) = \boxed{9.97\times10^{-6}}$

6 Problem 6

$P = 110011$ and $M = 11100011$
$\Rightarrow$ *Remainder* $= 11010 \Rightarrow$ CRC $= \boxed{11100011}\boxed{11010}$

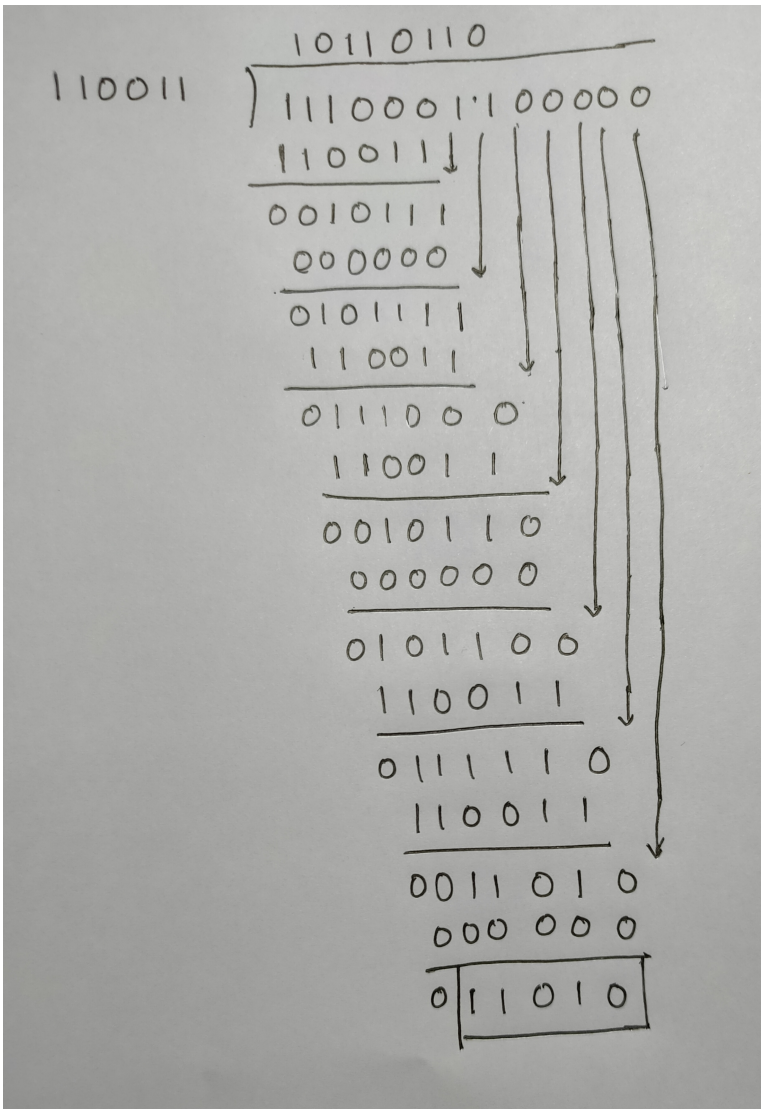

Figure 2: CRC representation using modulo 2 arithmetic

7 Problem 7

$P(X) = X^4 + X + 1$ and let the frame be $D(X) = X^{10} + X^7 + X^4 + X^3 + X + 1$

1. $R(X) = X^3 + X^2$. Therefore the CRC encoding will be $\boxed{10010011011}\boxed{1100}$

2. Received frame will be 000110110111100. Since the remainder $R(X) = X^3+X^2+X$ is not 0, the receiver detects an error.

3. Since the remainder $R(X) = 0$ the receiver couldn't detect error. The reason behind this is based on the property that if $a \oplus b = 0$ and $a \oplus c = 0$ then $a \oplus (b - c) = 0$, meaning that $c = b + \lambda a$. Therefore if the receiver receives the original signal plus finite times the remainder, then the corrupted frame will go undetected by the receiver.

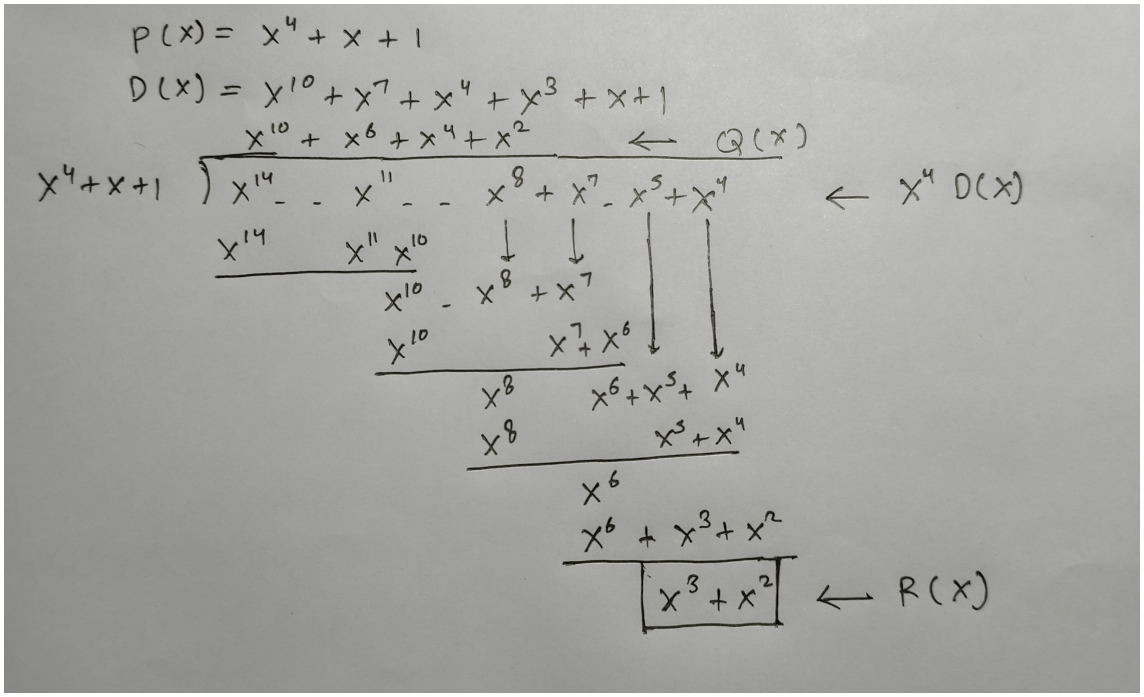

Figure 3: CRC representation using polynomials

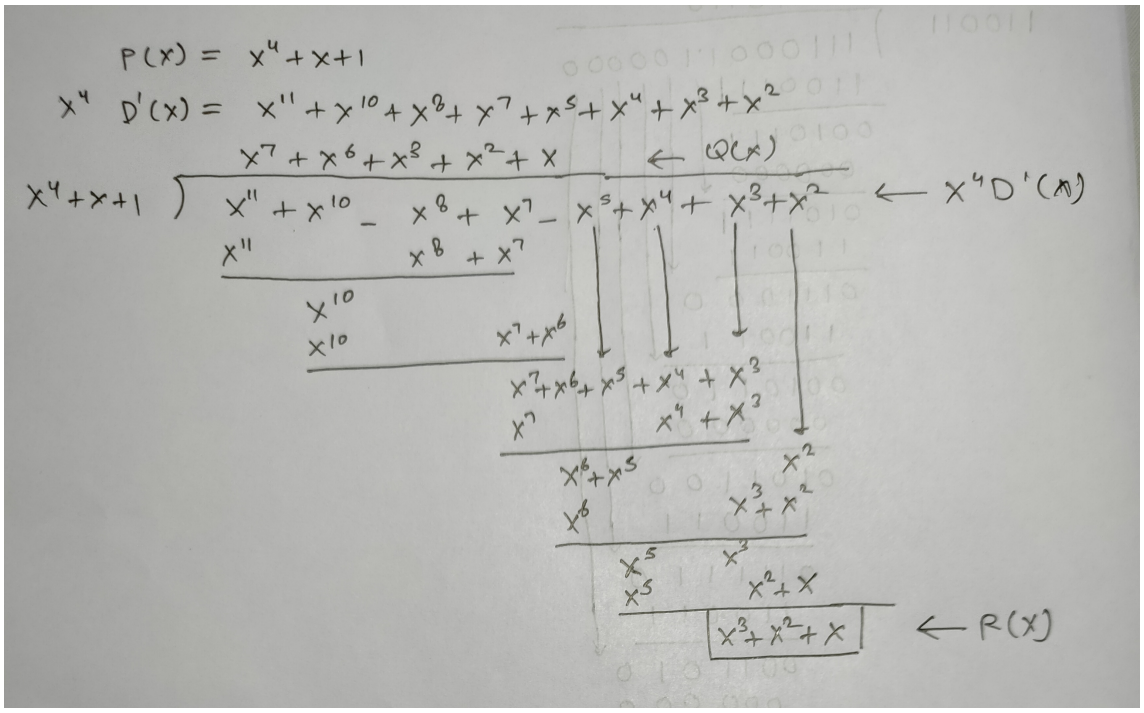

Figure 4: Error in received frame got detected

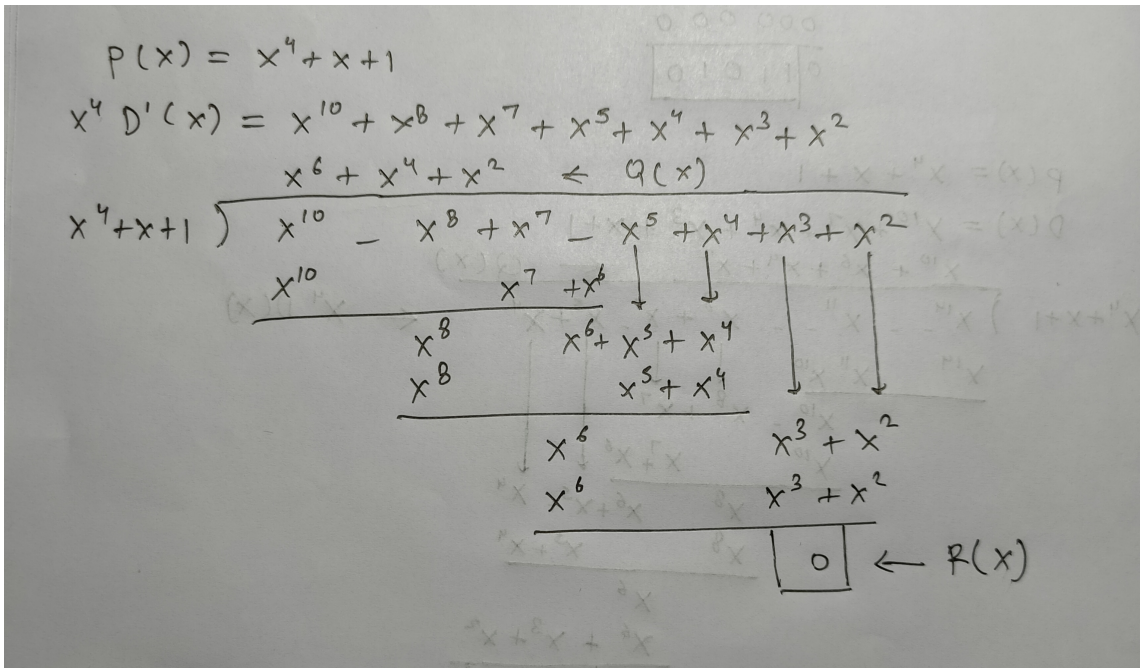

Figure 5: Error in received frame got undetected