Computer Network

Lecture-18

Dharmendra Kumar (Associate Professor)
Department of Computer Science and Engineering
United College of Engineering and Research,
Prayagraj

Error Detection and Correction

Cyclic Redundancy Check (CRC)

- Suppose size of dataword is k-bits.
- This technique uses a divisor to find a codeword.
- Suppose size of divisor is m-bits.

At sender end:

The codeword corresponding to dataword is found in the following way:-

1. First we find a word by augmenting m-1 0's to the right end of the dataword.
2. Now, we divide this new word by the divisor and find a remainder.
3. Codeword is obtained by augmenting the remainder to the right end of dataword.

Error Detection and Correction

At receiver end:

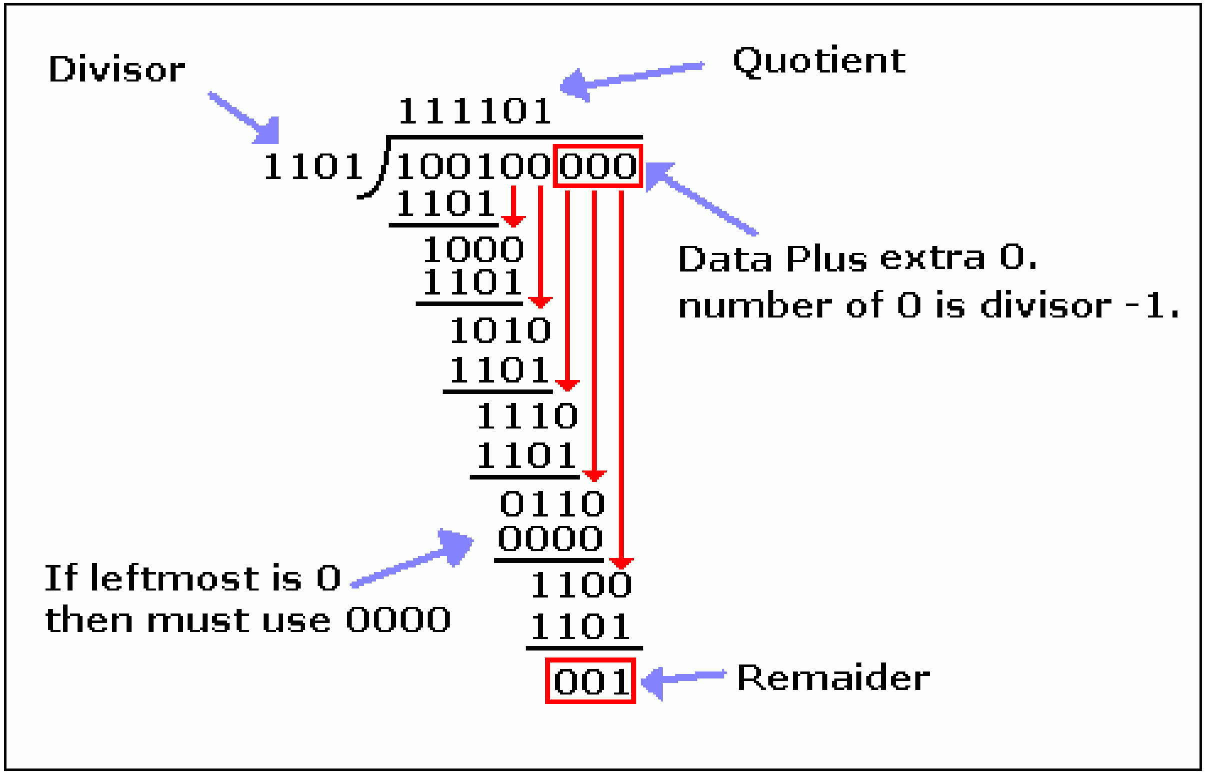
The dataword at the receiver end is found in the following way:-

1. First we divide the received codeword by divisor and find the remainder.
2. Remainder is called syndrome. If remainder is zero, then dataword will be accepted otherwise dataword will be rejected or discorded.
3. If remainder is zero, then dataword will be found by removing m-1 least significant bits of received codeword.

Error Detection and Correction

Example: If divisor is 1101, then find codeword corresponding to the dataword 100100.

Solution:



Therefore codeword = 100100001

Error Detection and Correction

Example:

(1) If Codeword 100100001 is received at receiver end, then find syndrome.

(2) If Codeword 100100101 is received at receiver end, then find syndrome.

CRC in polynomial

- The divisor in CRC is normally called generator.
- We define the following terms:-

Dataword = d(x)   Codeword = c(x)   Generator = g(x)

Syndrome = s(x)   Error = e(x)

Error Detection and Correction

In a cyclic code,

1. If $s(x) \neq 0$, one or more bits is corrupted.
2. If $s(x) = 0$, then either
   a. No bit is corrupted. or
   b. Some bits are corrupted, but the decoder failed to detect them.

Received codeword = c(x) + e(x)

The receiver divides the received codeword by g(x) to get the syndrome.

$$\frac{Received\ codeword}{g(x)} = \frac{c(x)}{g(x)} + \frac{e(x)}{g(x)}$$

$\frac{c(x)}{g(x)}$ $does\ not\ have\ a\ remainder. So\ the\ syndrome\ is\ the\ remainder\ of\ \frac{e(x)}{g(x)}$.

In a cyclic code, those e(x) errors that are divisible by g(x) are not caught.

Error Detection and Correction

Example:

Let dataword $d(x) = x^3+1$, generator $g(x) = x^3+x+1$.

Find codeword.

Solution:

Augmented dataword $= x^6+x^3$

$$
\begin{array}{r l}
x^3+x+1) & x^6+x^3 \quad (x^3+x \\
 & x^6+x^4+x^3 \\
\hline
 & x^4 \\
 & x^4+x^2+x \\
\hline
 & x^2+x
\end{array}
$$

Remainder

Therefore, $c(x) = x^6+x^3+x^2+x$

Error Detection and Correction

Single-Bit Error

If the generator has more than one term and the coefficient of x^0 is 1, then all single bit errors can be caught.

Example:

Which of the following g(x) values guarantees that a single-bit error is caught? For each case, what is the error that cannot be caught?

(a) $x + 1$

(b) x^3

(c) 1

Solution:

(a) No x^i can be divisible by $x + 1$. In other words, $x^i/(x + 1)$ always has a remainder. So the syndrome is nonzero. Any single-bit error can be caught.

Error Detection and Correction

(b) If i is equal to or greater than 3, then x^i is divisible by g(x). The remainder of x^i/x^3 is zero, and the receiver is fooled into believing that there is no error, although there might be one.

Note that in this case, the corrupted bit must be in position 4 or above. All single-bit errors in positions 1 to 3 are caught.

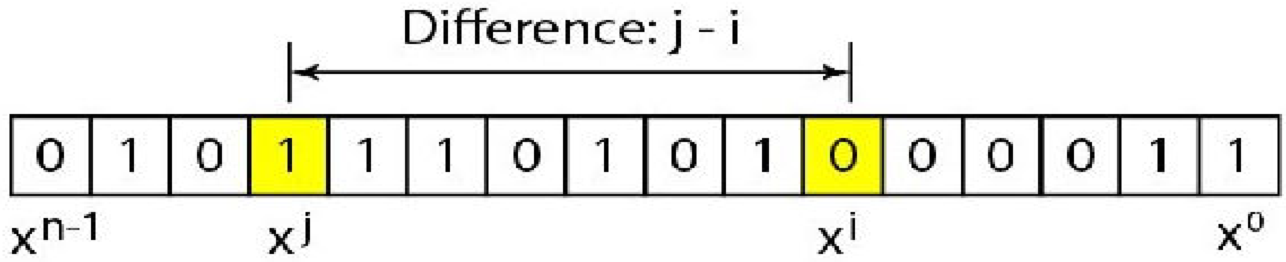
(c) For all values of i, x^i is divisible by g(x). No single-bit error can be caught. In addition, this g(x) is useless because it means the codeword is just the dataword augmented with n-k zeros.

Error Detection and Correction

Two Isolated Single-Bit Errors

$$e(x) = x^j + x^i$$

The values of i and j define the positions of the errors, and the difference j - i defines the distance between the two errors.



- ❖ If a generator cannot divide $x^t + 1$ (t between 0 and n - 1), then all isolated double errors can be detected.

Error Detection and Correction

Example:

Find the status of the following generators related to two isolated, single-bit errors.

(a) $x + 1$

(b) $x^4 + 1$

(c) $x^7 + x^6 + 1$

(d) $x^{15} + x^{14} + 1$

Solution:

(a) This is a very poor choice for a generator. Any two errors next to each other cannot be detected.

(b) This generator cannot detect two errors that are four positions apart. The two errors can be anywhere, but if their distance is 4, they remain undetected.

Error Detection and Correction

(c) This is a good choice for this purpose.

(d) This polynomial cannot divide any error of type $x^t + 1$ if t is less than 32,768. This means that a codeword with two isolated errors that are next to each other or up to 32,768 bits apart can be detected by this generator.

Odd Numbers of Errors

A generator that contains a factor of $x + 1$ can detect all odd-numbered errors.

Some standard generators

(1) CRC-8 = $x^8+x^2+x + 1$

(2) (2) CRC-10 = $x^{10}+x^9+x^5+x^4+x^2+ 1$

(3) CRC-16 = $x^{16} + x^{12} + x^5 + 1$

(4) CRC-32 = $x^{32} + x^{26} +x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$