IT301: Data Communication & Computer Network(DCCN)

Class: B. Tech (CS) Sec A
Semester : V
Teacher: Dr. Amritanjali

Week 6

Module III

- Digital Data Communication Techniques
  – Synchronization
  – Types of Errors
  – Error Detection
  – Error Correction
- Data Link Control Protocols
  —Flow Control
  —Error Control
  —High-Level Data Link Control

Digital Data Communication Techniques

Types of Errors

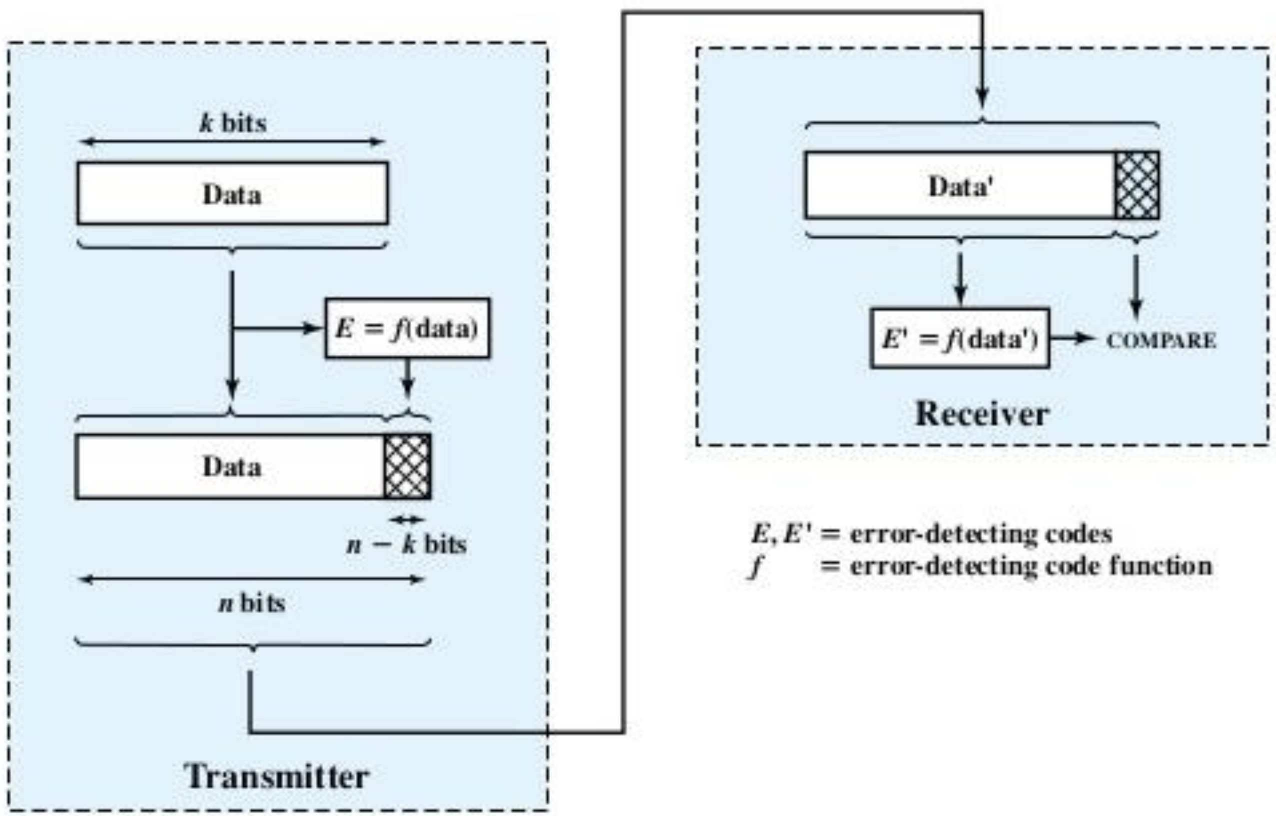
- Bits get altered due to transmission errors
- **Single Bit Error-** occurs in the presence of white noise
- **Burst Error-** when a contiguous sequence of bits gets affected or a cluster of bits with a number of error bits in the cluster
- Burst errors are usually caused by impulse noise and its effects are more at higher data rates

Error Detection

Error Detection Codes (Check Bits or Checksum or Frame Check Sequence)

- Additional bits for error detection added to a given frame of bits
- Code is calculated as a function of transmitted bits
- Receiver performs error detection code calculation on the data bits and compares the result with the received check bits
- If there is mismatch then there is error, otherwise either there in no error or error is undetected

Error Detection Process



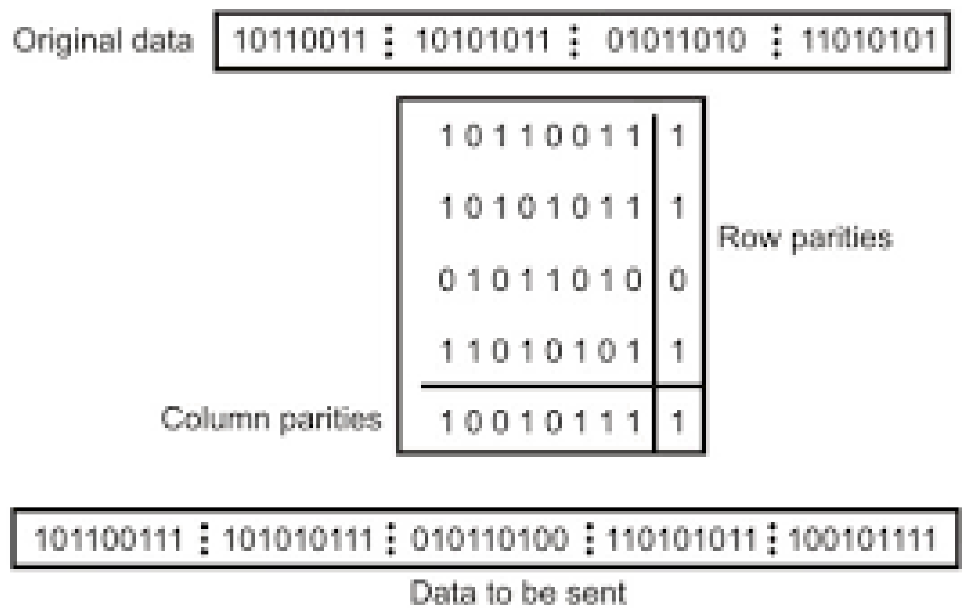
Parity Check

- **Even Parity-** A parity bit is added to the end of data block to make the total number of 1's even
- **Odd Parity-** A parity bit is added to the end of data block to make the total number of 1's odd

Block parity can be used

for sending frames with

multiple characters



Checksum

- In checksum error detection scheme, the data is divided into k segments each of m bits.
- In the sender's end the segments are added using 1's complement arithmetic to get the sum.
- The sum is complemented to get the checksum.
- The checksum segment is sent along with the data segments.
- At the receiver's end, all received segments are added using 1's complement arithmetic to get the sum. The sum is complemented. If the result is zero, the received data is accepted; otherwise discarded

Checksum

Example:

```
k=4,   m=8
10110011
10101011
--------
01011110
       1
--------
01011111
01011010
--------
10111001
11010101
--------
10001110
       1
--------
Sum :      10001111
--------
Checksum   01110000
```

Example: Received data

```
10110011
10101011
--------
01011110
       1
--------
01011111
01011010
--------
10111001
11010101
--------
10001110
       1
--------
10001111
01110000
--------
Sum:  11111111
Complement = 00000000
Conclusion = Accept data
```

Image Source: nptel.ac.in

Cyclic Redundancy Check

Cyclic Redundancy Check (CRC)

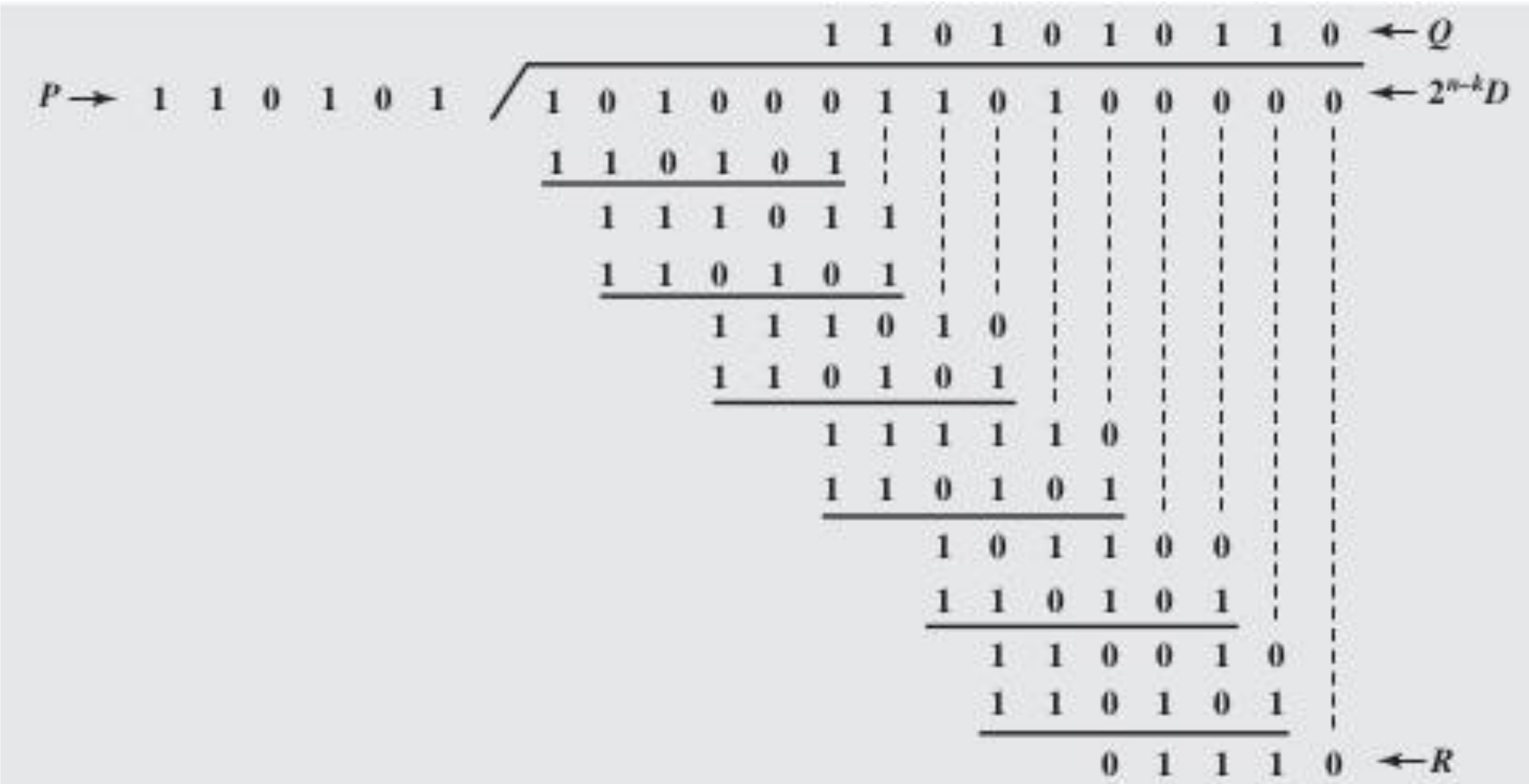
- One of the most powerful error detection codes
- To the k bits of data (n-k) bits frame check sequence (FCS) is appended, so that the resulting n bits is exactly divisible by a predetermined divisor
- To detect error, receiver divides the received frame of n bits by the same divisor
- Assumes no error if the remainder is 0

Frame Check Sequence

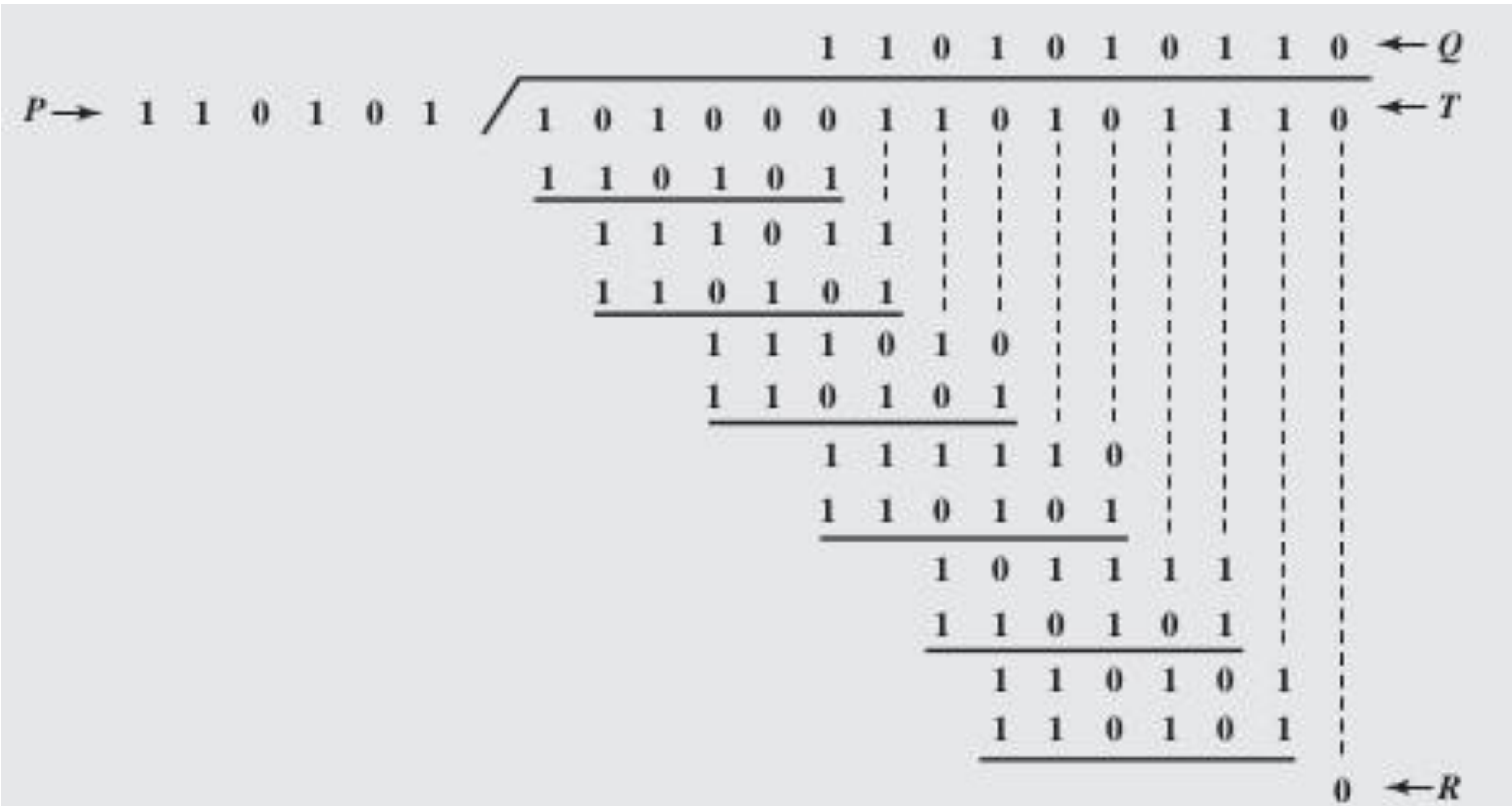
FCS Calculation

- Left Shift k-bits of data by (n-k) bits
- Divide by P, predetermined divisor of (n-k)+1 bits
- Remainder R of length (n-k) bits is FCS

FCS Calculation by Sender



Error Detection by Receiver



Polynomial Notation

- A pattern of 0s and 1s can be represented as a polynomial with coefficient of 0 and 1.
- Power of each term shows the position of the bit and the coefficient shows the values of the bit.

For example,

Binary pattern is 100101

Polynomial representation is $x^5 + x^2 + 1$

FCS using Polynomial Division

$$
\begin{array}{rl}
 & X^9 + X^8 + X^6 + X^4 + X^2 + X \quad \leftarrow Q(X) \\
P(X) \rightarrow X^5 + X^4 + X^2 + 1 \,\big/ & X^{14} \quad X^{12} \quad X^8 + X^7 + \quad X^5 \quad \leftarrow X^5 D(X) \\
 & \underline{X^{14} + X^{13} + \quad X^{11} + \quad X^9} \\
 & X^{13} + X^{12} + X^{11} + \quad X^9 + X^8 \\
 & \underline{X^{13} + X^{12} + \quad X^{10} + \quad X^8} \\
 & X^{11} + X^{10} + X^9 + \quad X^7 \\
 & \underline{X^{11} + X^{10} + \quad X^8 + \quad X^6} \\
 & X^9 + X^8 + X^7 + X^6 + X^5 \\
 & \underline{X^9 + X^8 + \quad X^6 + \quad X^4} \\
 & X^7 + \quad X^5 + X^4 \\
 & \underline{X^7 + X^6 + \quad X^4 + \quad X^2} \\
 & X^6 + X^5 + \quad X^2 \\
 & \underline{X^6 + X^5 + \quad X^3 + \quad X} \\
 & X^3 + X^2 + X \leftarrow R(X)
\end{array}
$$

Divisor Polynomials

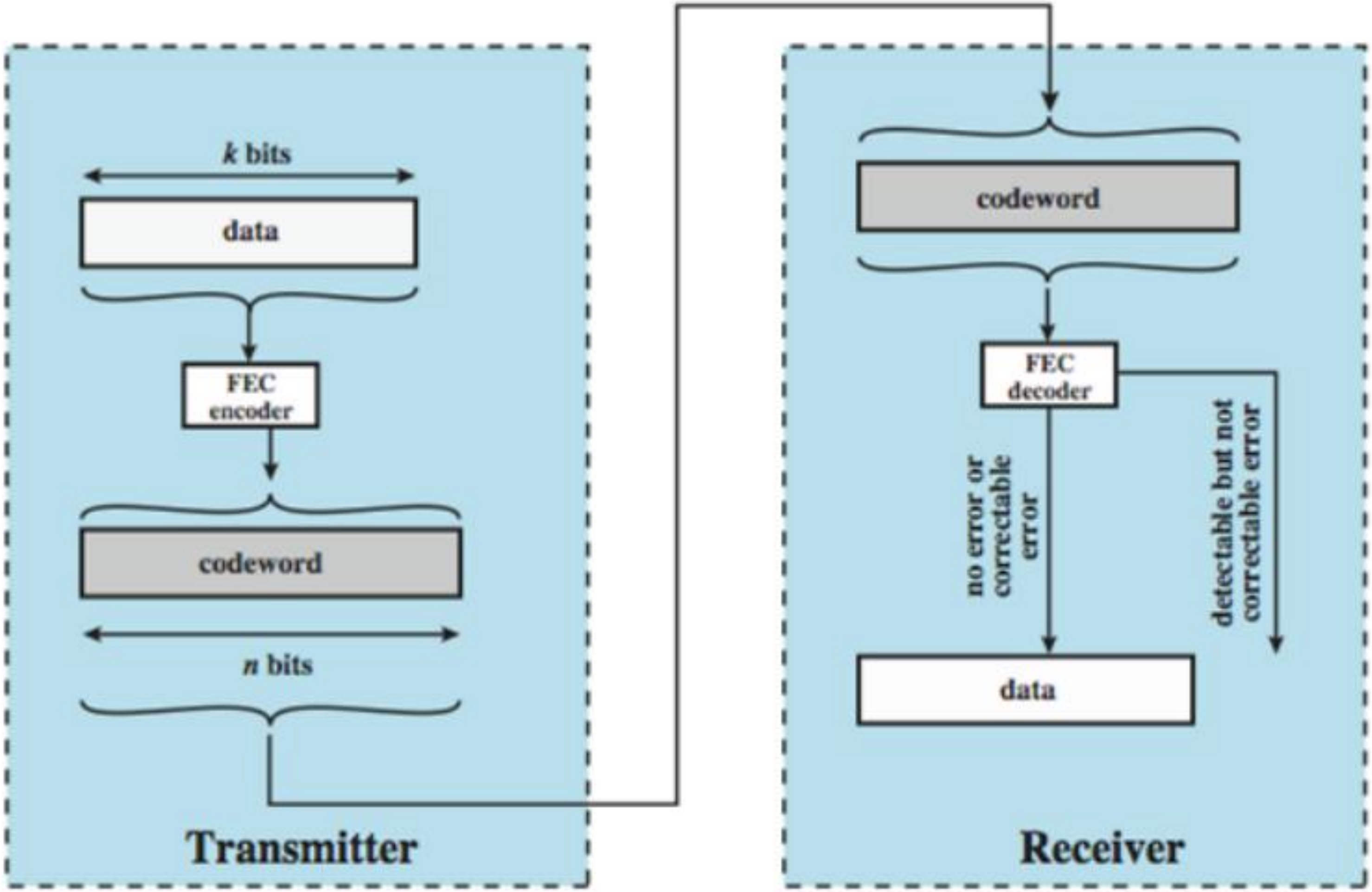
- All single-bit errors, if $P(X)$ has more than one nonzero term
- All double-bit errors, as long as $P(X)$ is a special type of polynomial, called a primitive polynomial, with maximum exponent L, and the frame length is less than $2^L - 1$.
- Any odd number of errors, as long as $P(X)$ contains a factor $(X + 1)$
- Any burst error for which the length of the burst is less than or equal to $n - k$; that is, less than or equal to the length of the FCS

$$
\begin{aligned}
\text{CRC-12} &= X^{12} + X^{11} + X^3 + X^2 + X + 1 \\
\text{CRC-16} &= X^{16} + X^{15} + X^2 + 1 \\
\text{CRC-CCITT} &= X^{16} + X^{12} + X^5 + 1 \\
\text{CRC-32} &= X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} \\
&\quad + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1
\end{aligned}
$$

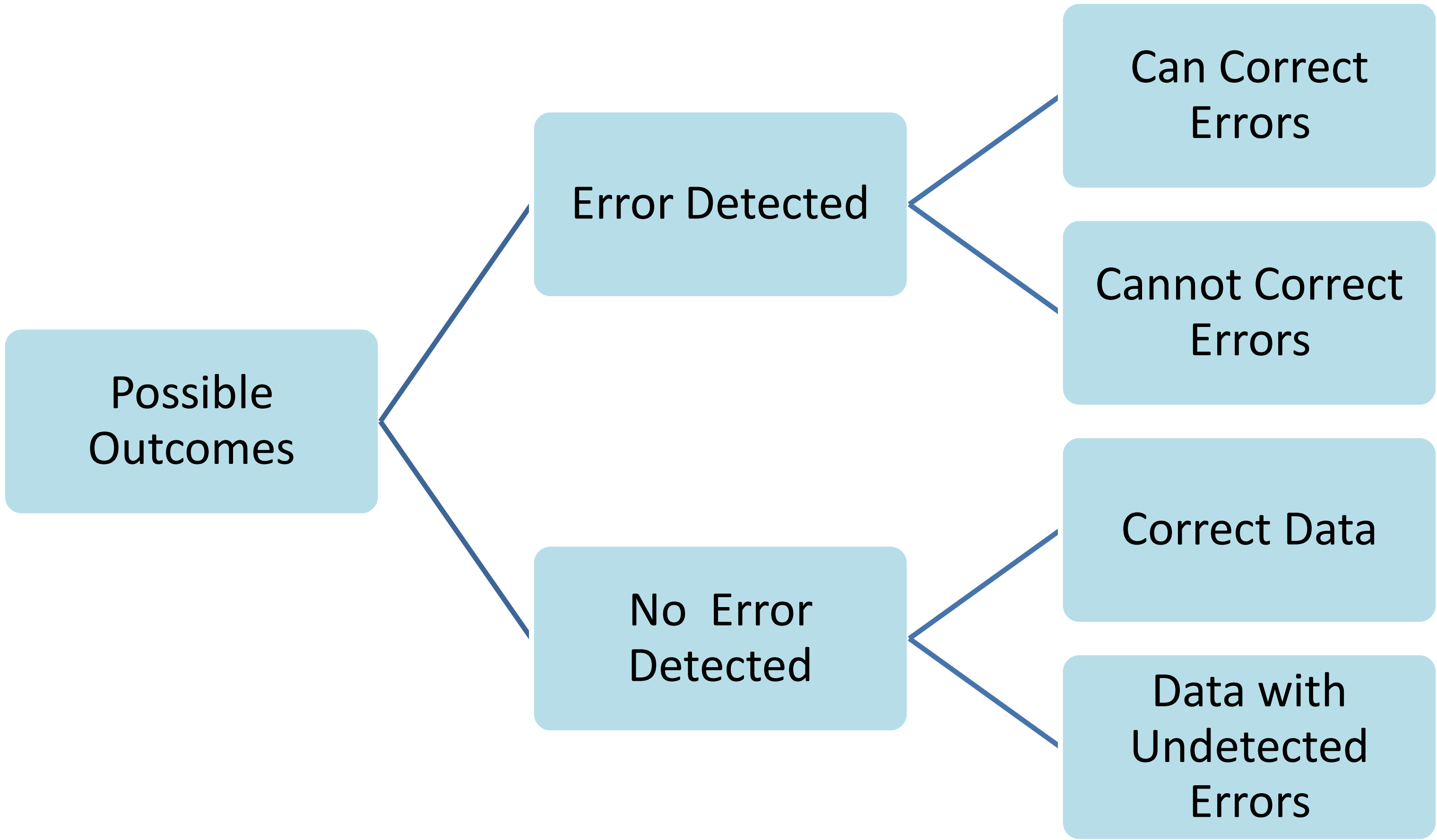
Error Correction

- Error correction codes are useful for wireless transmission where retransmissions is costly
- Before transmission, each k-bit data is mapped to a n-bit codeword using FEC encoder
- At receiver, the bit string obtained after demodulation is passed through a FEC decoder, for error detection and possible correction

Error Correction



Error Correction



Block Code Principles

- Hamming distance between two codewords is the number of bit positions in which they differ
- For k-bit data 2^k codewords of length n are selected with minimum hamming distance of dm. They are called as valid codewords.
- Data is transmitted by substituting with corresponding codeword
- Error is detected if the received codeword is invalid
- Error is corrected if the received codeword is closest to a unique codeword

Example

Data Block	Codeword
00	00000
01	00111
10	11001
11	11110

Let the codeword received is 00100

1. Check if the error can be detected?
2. If yes, then check if the error be corrected?

Error Detection and Correction using block code

- If the minimum Hamming distance between codeword c_1 and any other valid codeword is d_M then decoder could detect up to $d_M - 1$ errors
- To design a code that can correct m single bit errors, a minimum distance of (2m + 1) is required
- n should be selected such that (n-k) in neither too large nor too small, to balance between bandwidth consumption and error rate
- For a given value of n and k, codewords should be selected such that dm is maximised