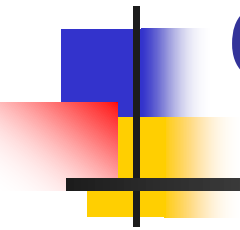
FaaDoOEngineers.com
Engaging Peers, Inspiring Careers!

India's No.1 website for:

- **IIT-JEE/AIEEE preparation resources**
- **Coaching centre packages**
- **Engineering Major and Minor projects**
- **Seminar reports**
- **Paper presentations**
- **EBOOKS**
- **Resumes/CVs, and so much more...**



Coding and Error Control

Chapter 8

g with Data Transmission Errors

- Error detection codes
  - Detects the presence of an error
- Automatic repeat request (ARQ) protocols
  - Block of data with error is discarded
  - Transmitter retransmits that block of data
- Error correction codes, or forward correction codes (FEC)
  - Designed to detect and correct errors

Error Detection Probabilities

- Definitions
  - P_b : Probability of single bit error (BER)
  - P_1 : Probability that a frame arrives with no bit errors
  - P_2 : While using error detection, the probability that a frame arrives with one or more undetected errors
  - P_3 : While using error detection, the probability that a frame arrives with one or more detected bit errors but no undetected bit errors

Error Detection Probabilities

- With no error detection

$$P_1 = (1 - P_b)^F$$

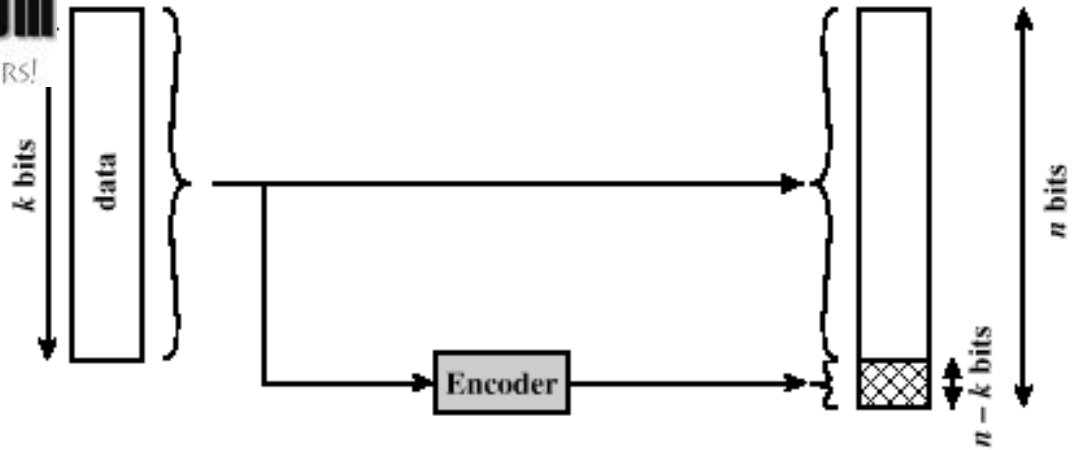
$$P_2 = 1 - P_1$$

$$P_3 = 0$$

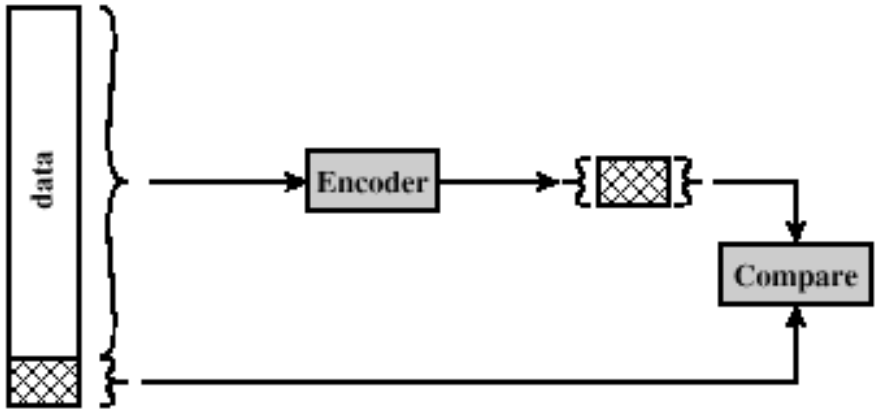
  - F = Number of bits per frame

Error Detection Process

- Transmitter
  - For a given frame, an error-detecting code (check bits) is calculated from data bits
  - Check bits are appended to data bits
- Receiver
  - Separates incoming frame into data bits and check bits
  - Calculates check bits from received data bits
  - Compares calculated check bits against received check bits
  - Detected error occurs if mismatch



(a) Sender



(b) Receiver

Figure 8.1 Error Detection Process

Parity Check

- Parity bit appended to a block of data
- Even parity
  - Added bit ensures an even number of 1s
- Odd parity
  - Added bit ensures an odd number of 1s
- Example, 7-bit character [1110001]
  - Even parity [11100010]
  - Odd parity [11100011]

Cyclic Redundancy Check (CRC)

- Transmitter
  - For a *k*-bit block, transmitter generates an (*n*-*k*)-bit frame check sequence (FCS)
  - Resulting frame of *n* bits is exactly divisible by predetermined number
- Receiver
  - Divides incoming frame by predetermined number
  - If no remainder, assumes no error

CRC using Modulo 2 Arithmetic

- Exclusive-OR (XOR) operation
- Parameters:
  - T = n-bit frame to be transmitted
  - D = k-bit block of data; the first k bits of T
  - F = $(n - k)$-bit FCS; the last $(n - k)$ bits of T
  - P = pattern of $n–k+1$ bits; this is the predetermined divisor
  - Q = Quotient
  - R = Remainder

CRC using Modulo 2 Arithmetic

- For T/P to have no remainder, start with

$$T = 2^{n-k} D + F$$

- Divide $2^{n-k}D$ by P gives quotient and remainder

$$\frac{2^{n-k} D}{P} = Q + \frac{R}{P}$$

- Use remainder as FCS

$$T = 2^{n-k} D + R$$

CRC using Modulo 2 Arithmetic

- Does R cause T/P have no remainder?

$$\frac{T}{P} = \frac{2^{n-k}D + R}{P} = \frac{2^{n-k}D}{P} + \frac{R}{P}$$

- Substituting,

$$\frac{T}{P} = Q + \frac{R}{P} + \frac{R}{P} = Q + \frac{R+R}{P} = Q$$

  - No remainder, so T is exactly divisible by P

CRC using Polynomials

- All values expressed as polynomials
  - Dummy variable X with binary coefficients

$$\frac{X^{n-k}D(X)}{P(X)} = Q(X) + \frac{R(X)}{P(X)}$$

$$T(X) = X^{n-k}D(X) + R(X)$$

CRC using Polynomials

- Widely used versions of *P(X)*
  - CRC–12
    - $X^{12} + X^{11} + X^3 + X^2 + X + 1$
  - CRC–16
    - $X^{16} + X^{15} + X^2 + 1$
  - CRC – CCITT
    - $X^{16} + X^{12} + X^5 + 1$
  - CRC – 32
    - $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$

CRC using Digital Logic

- Dividing circuit consisting of:
  - XOR gates
    - Up to $n - k$ XOR gates
    - Presence of a gate corresponds to the presence of a term in the divisor polynomial $P(X)$
  - A shift register
    - String of 1-bit storage devices
    - Register contains $n - k$ bits, equal to the length of the FCS

Digital Logic CRC

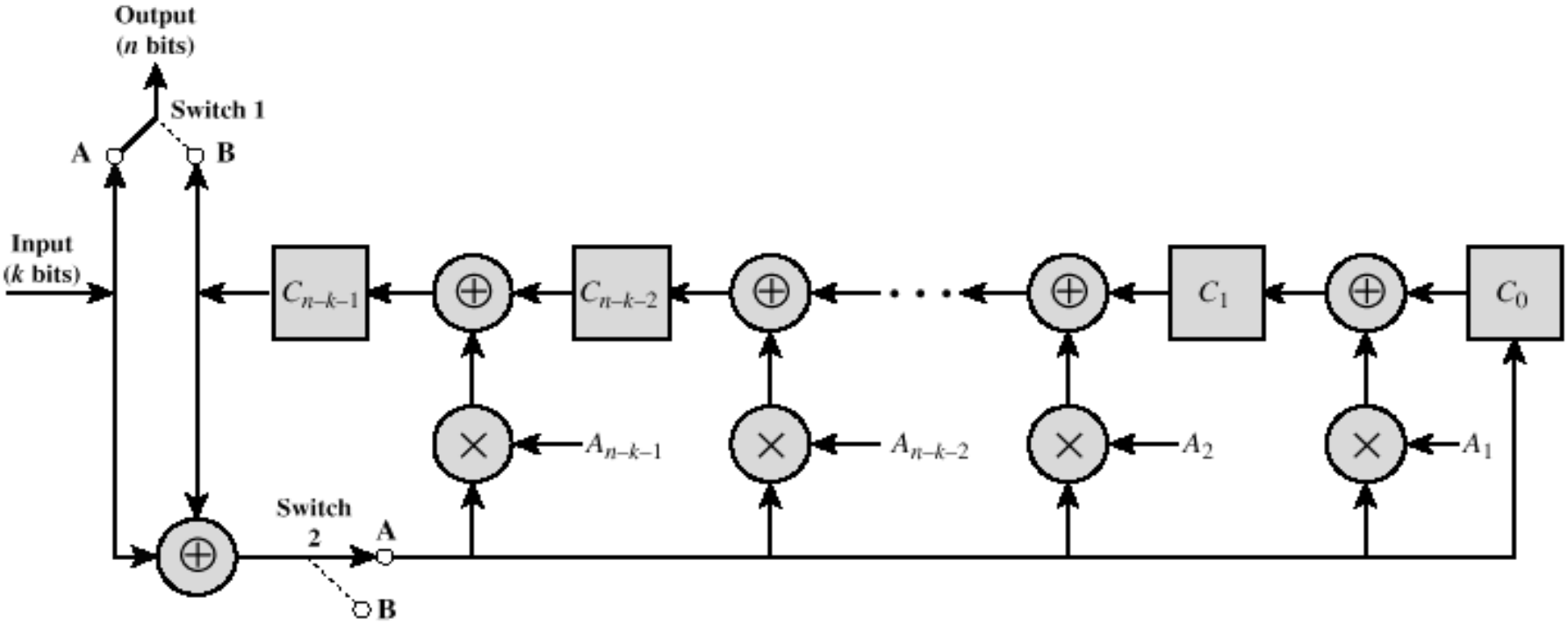


Figure 8.4 General CRC Architecture to Implement Divisor
$1 + A_1X + A_2X^2 + \ldots + A_{n-1}X^{n-k-1} + X^{n-k}$

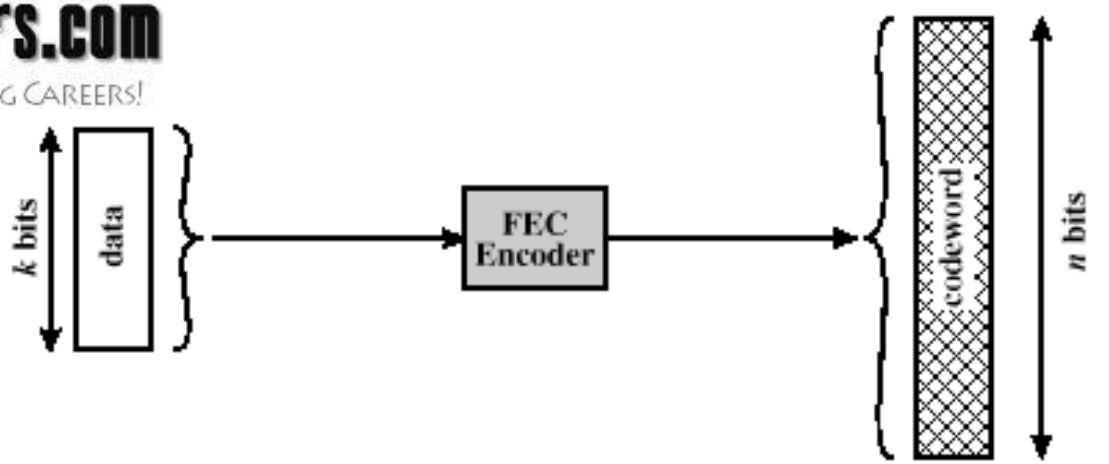
Wireless Transmission Errors

- Error detection requires retransmission
- Detection inadequate for wireless applications
  - Error rate on wireless link can be high, results in a large number of retransmissions
  - Long propagation delay compared to transmission time

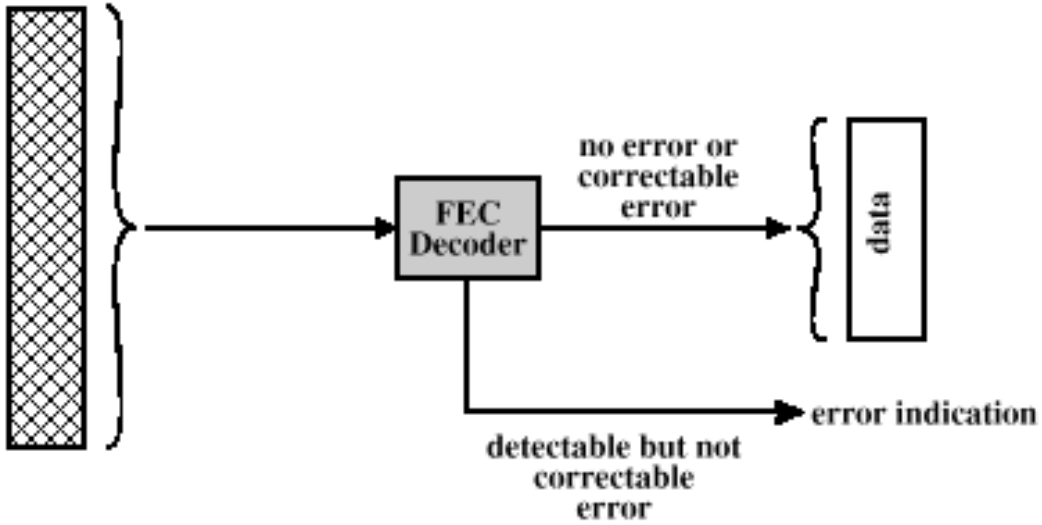
Block Error Correction Codes

- Transmitter
  - Forward error correction (FEC) encoder maps each k-bit block into an n-bit block codeword
  - Codeword is transmitted; analog for wireless transmission
- Receiver
  - Incoming signal is demodulated
  - Block passed through an FEC decoder

SS



(a) Sender



(b) Receiver

Figure 8.5 Forward Error Correction Process

FEC Decoder Outcomes

- No errors present
  - Codeword produced by decoder matches original codeword
- Decoder detects and corrects bit errors
- Decoder detects but cannot correct bit errors; reports uncorrectable error
- Decoder detects no bit errors, though errors are present

Block Code Principles

- Hamming distance – for 2 *n*-bit binary sequences, the number of different bits
  - E.g., v_1=011011; v_2=110001; d(v1, v_2)=3
- Redundancy – ratio of redundant bits to data bits
- Code rate – ratio of data bits to total bits
- Coding gain – the reduction in the required E_b/N_0 to achieve a specified BER of an error-correcting coded system

Hamming Code

- Designed to correct single bit errors
- Family of (n, k) block error-correcting codes with parameters:
  - Block length: $n = 2^m - 1$
  - Number of data bits: $k = 2^m - m - 1$
  - Number of check bits: $n - k = m$
  - Minimum distance: $d_{\min} = 3$
- Single-error-correcting (SEC) code
  - SEC double-error-detecting (SEC-DED) code

Hamming Code Process

- Encoding: k data bits + (n -k) check bits
- Decoding: compares received (n -k) bits with calculated (n -k) bits using XOR
  - Resulting (n -k) bits called *syndrome word*
  - Syndrome range is between 0 and $2^{(n-k)}$-1
  - Each bit of syndrome indicates a match (0) or conflict (1) in that bit position

Cyclic Codes

- Can be encoded and decoded using linear feedback shift registers (LFSRs)
- For cyclic codes, a valid codeword (c_0, c_1, …, c_{n-1}), shifted right one bit, is also a valid codeword (c_{n-1}, c_0, …, c_{n-2})
- Takes fixed-length input (k) and produces fixed-length check code (n-k)
  - In contrast, CRC error-detecting code accepts arbitrary length input for fixed-length check code

BCH Codes

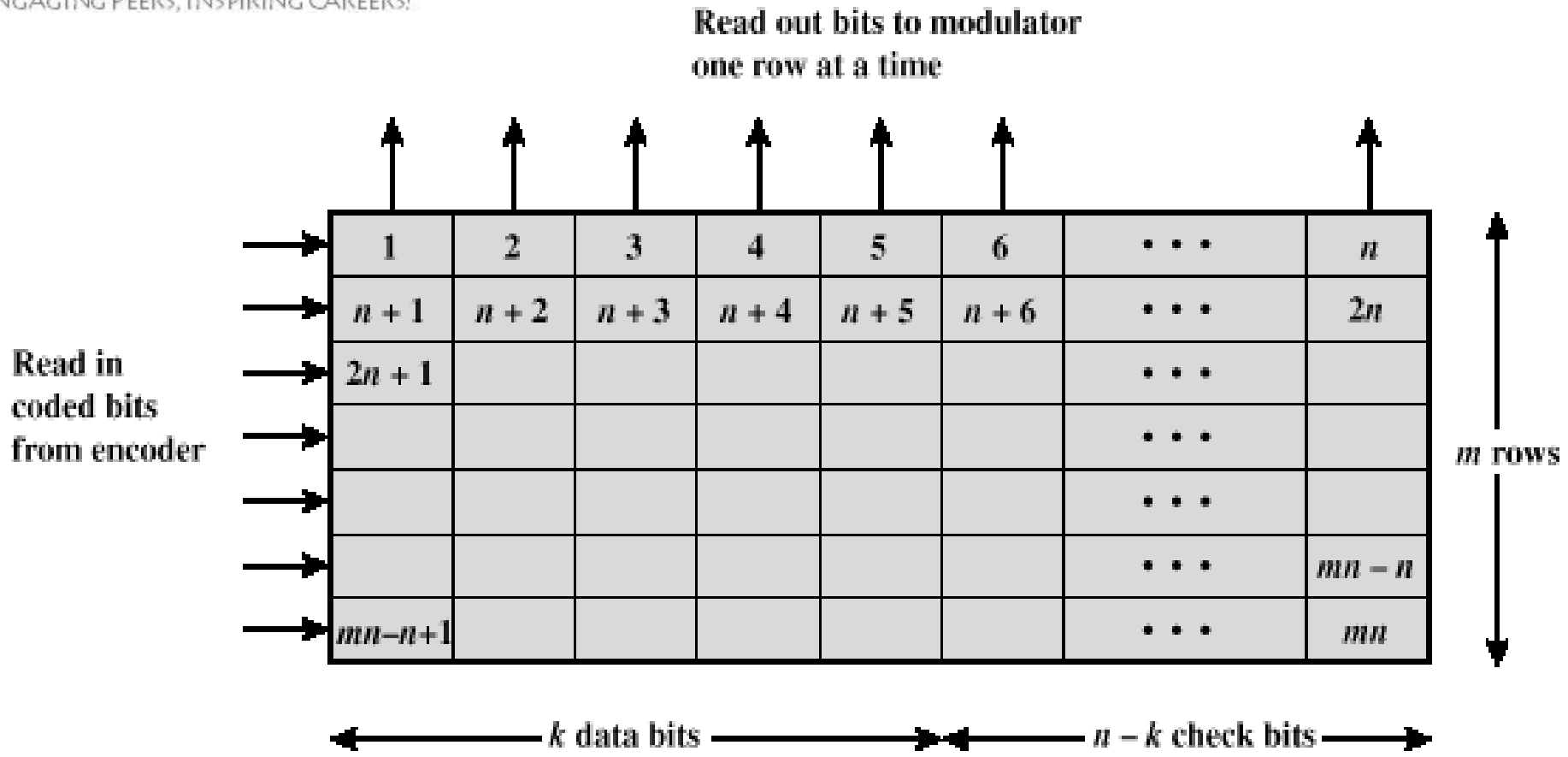
- For positive pair of integers m and t, a (n, k) BCH code has parameters:
  - Block length: $n = 2^m - 1$
  - Number of check bits: $n - k \leq mt$
  - Minimum distance: $d_{\min} \geq 2t + 1$
- Correct combinations of t or fewer errors
- Flexibility in choice of parameters
  - Block length, code rate

Reed-Solomon Codes

- Subclass of nonbinary BCH codes
- Data processed in chunks of m bits, called symbols
- An (n, k) RS code has parameters:
  - Symbol length: m bits per symbol
  - Block length: $n = 2^m - 1$ symbols = $m(2^m - 1)$ bits
  - Data length: k symbols
  - Size of check code: $n - k = 2t$ symbols = $m(2t)$ bits
  - Minimum distance: $d_{\min} = 2t + 1$ symbols

Block Interleaving

- Data written to and read from memory in different orders
- Data bits and corresponding check bits are interspersed with bits from other blocks
- At receiver, data are deinterleaved to recover original order
- A burst error that may occur is spread out over a number of blocks, making error correction possible

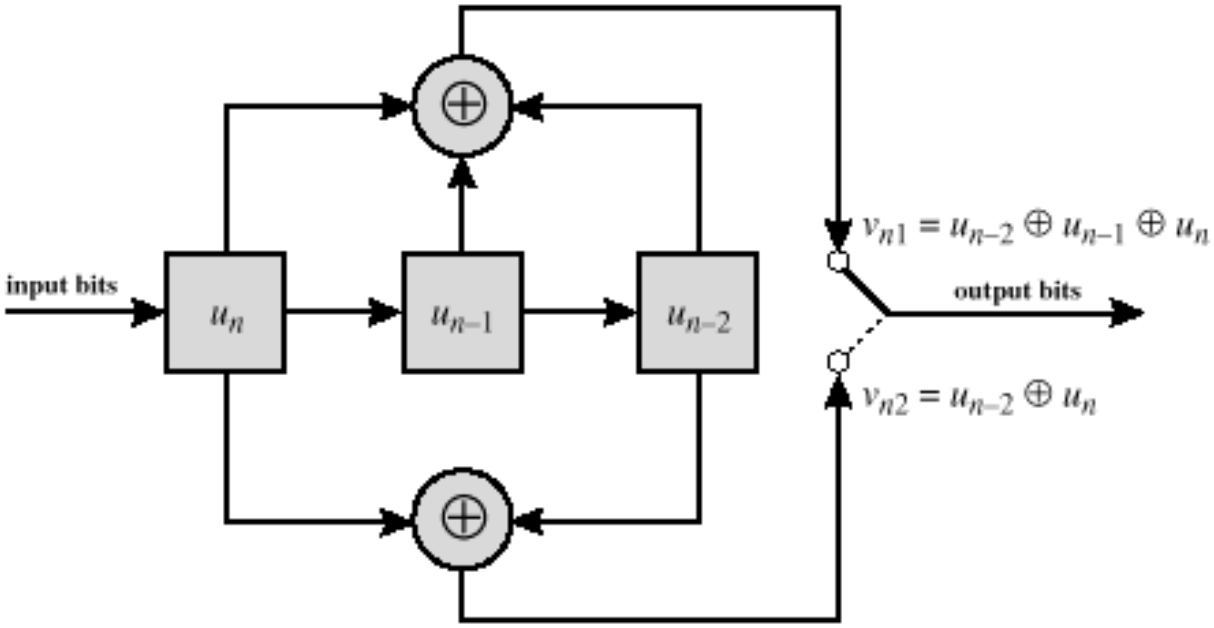


Note: The numbers in the matrix indicate the order in which bits are read in.
Interleaver output sequence: 1, $n + 1$, $2n + 1$, ...

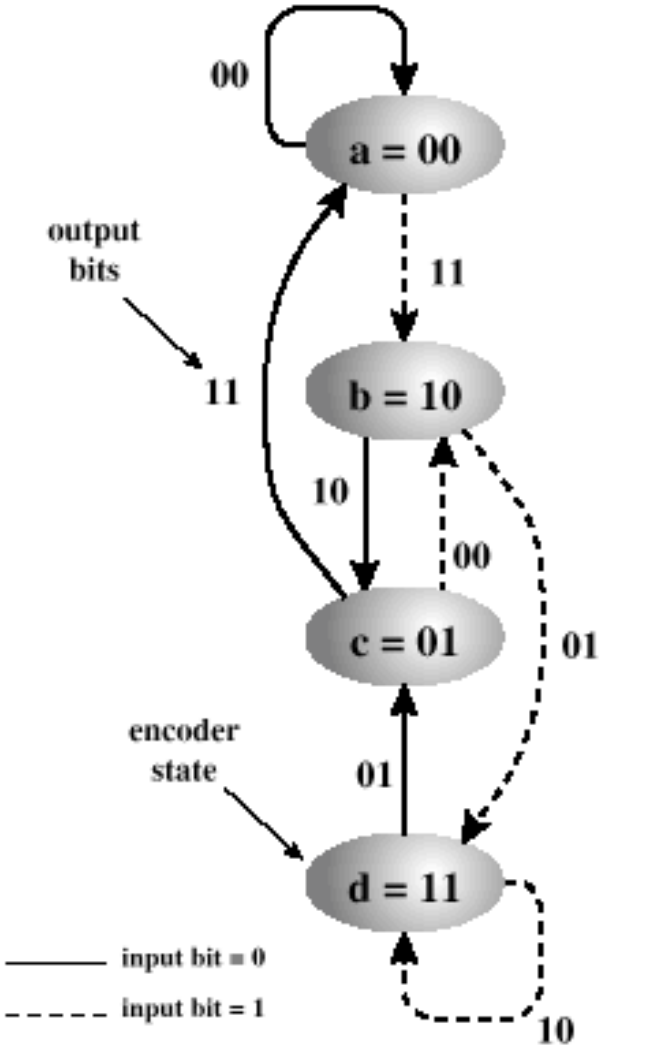
Figure 8.8 Block Interleaving

Convolutional Codes

- Generates redundant bits continuously
- Error checking and correcting carried out continuously
  - (n, k, K) code
    - Input processes k bits at a time
    - Output produces n bits for every k input bits
    - K = constraint factor
    - k and n generally very small
  - n-bit output of (n, k, K) code depends on:
    - Current block of k input bits
    - Previous K-1 blocks of k input bits



(a) Encoder shift register



(b) Encoder state diagram

Figure 8.9 Convolutional Encoder with $(n, k, K) = (2, 1, 3)$

Decoding

- Trellis diagram – expanded encoder diagram
- Viterbi code – error correction algorithm
  - Compares received sequence with all possible transmitted sequences
  - Algorithm chooses path through trellis whose coded sequence differs from received sequence in the fewest number of places
  - Once a valid path is selected as the correct path, the decoder can recover the input data bits from the output code bits

Automatic Repeat Request

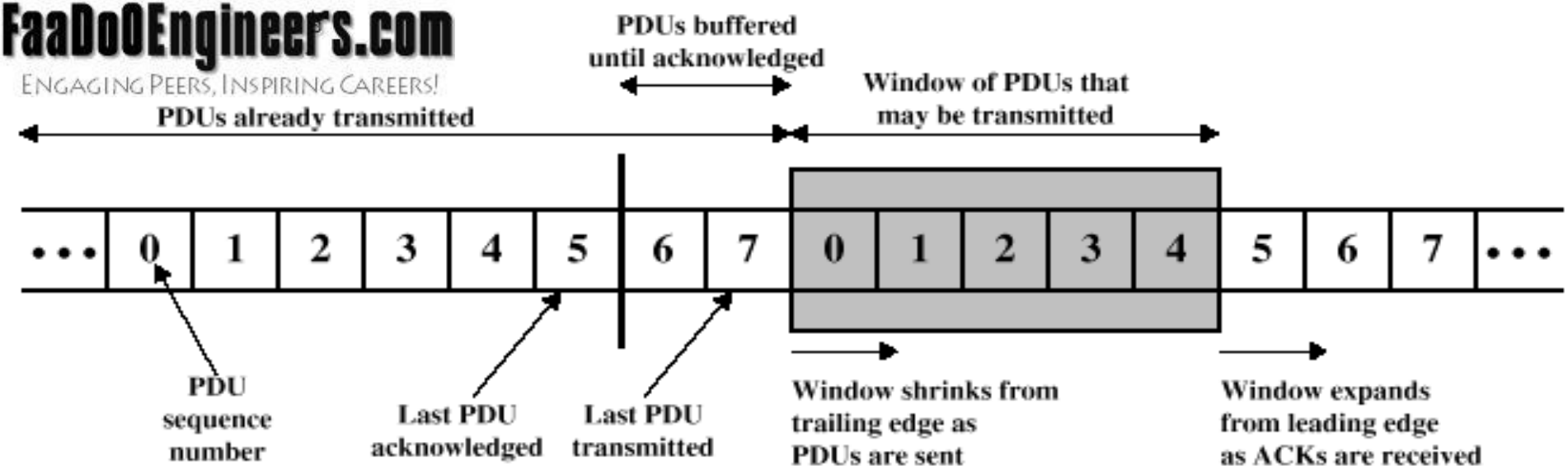
- Mechanism used in data link control and transport protocols
- Relies on use of an error detection code (such as CRC)
- Flow Control
- Error Control

Flow Control

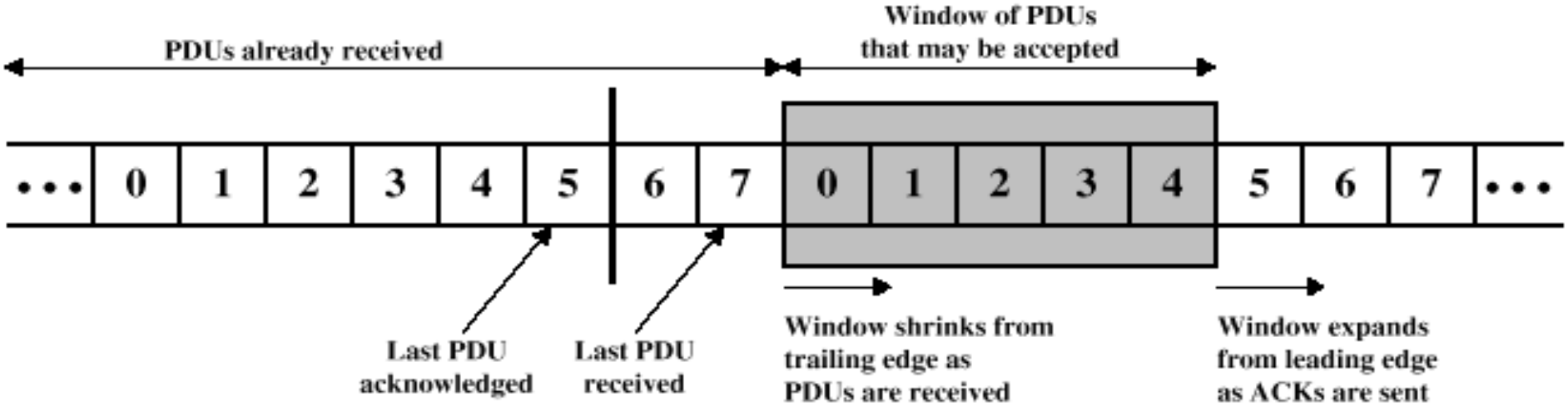
- Assures that transmitting entity does not overwhelm a receiving entity with data
- Protocols with flow control mechanism allow multiple PDUs in transit at the same time
- PDUs arrive in same order they're sent
- Sliding-window flow control
  - Transmitter maintains list (window) of sequence numbers allowed to send
  - Receiver maintains list allowed to receive

Flow Control

- Reasons for breaking up a block of data before transmitting:
  - Limited buffer size of receiver
  - Retransmission of PDU due to error requires smaller amounts of data to be retransmitted
  - On shared medium, larger PDUs occupy medium for extended period, causing delays at other sending stations



(a) Sender's perspective



(b) Receiver's perspective

Figure 8.17 Sliding-Window Depiction

Error Control

- Mechanisms to detect and correct transmission errors
- Types of errors:
  - Lost PDU : a PDU fails to arrive
  - Damaged PDU : PDU arrives with errors

Error Control Requirements

- Error detection
  - Receiver detects errors and discards PDUs
- Positive acknowledgement
  - Destination returns acknowledgment of received, error-free PDUs
- Retransmission after timeout
  - Source retransmits unacknowledged PDU
- Negative acknowledgement and retransmission
  - Destination returns negative acknowledgment to PDUs in error

Go-back-N ARQ

- Acknowledgments
  - RR = receive ready (no errors occur)
  - REJ = reject (error detected)
- Contingencies
  - Damaged PDU
  - Damaged RR
  - Damaged REJ

FaaDoOEngineers.com
Engaging Peers, Inspiring Careers!

India's No.1 website for:

- **IIT-JEE/AIEEE preparation resources**
- **Coaching centre packages**
- **Engineering Major and Minor projects**
- **Seminar reports**
- **Paper presentations**
- **EBOOKS**
- **Resumes/CVs, and so much more...**