Coding and Error Control

Coping with Transmission Errors

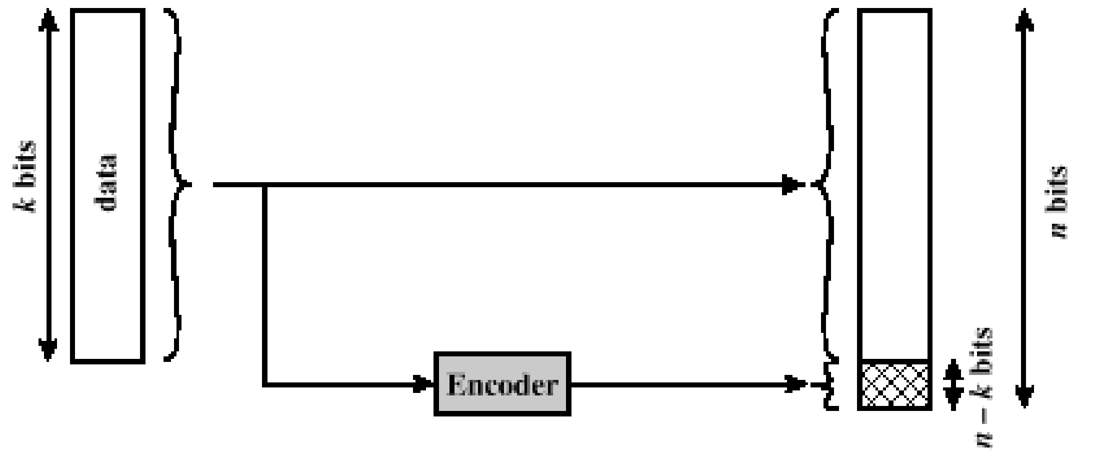
- Error detection codes
  - Detects the presence of an error
- Error correction codes, or forward correction codes (FEC)
  - Designed to detect and correct errors
  - Widely used in wireless networks
- Automatic repeat request (ARQ) protocols
  - Used in combination with error detection/correction
 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

Error Detection Probabilities

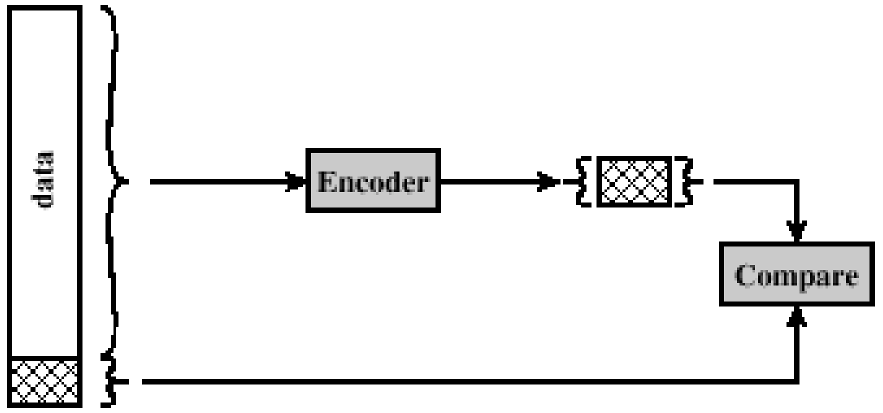
- Probability of single bit error (BER)
- Probability that a frame arrives with no bit errors = $(1 - \text{BER})^F$
- Probability that a frame arrives with undetected errors (residual error rate)
- Probability that a frame arrives with one or more detected bit errors

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* to 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
  - o Dummy variable *X* with binary coefficients

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

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

Error Detection using CRC

- All single bit errors, if P(X) has more than one non-zero term
- All double bit errors, as long as P(X) has a factor with at least 3 terms
- All odd errors, as long as P(X) contains X+1 as a factor
- Any burst error of length at most n-k

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

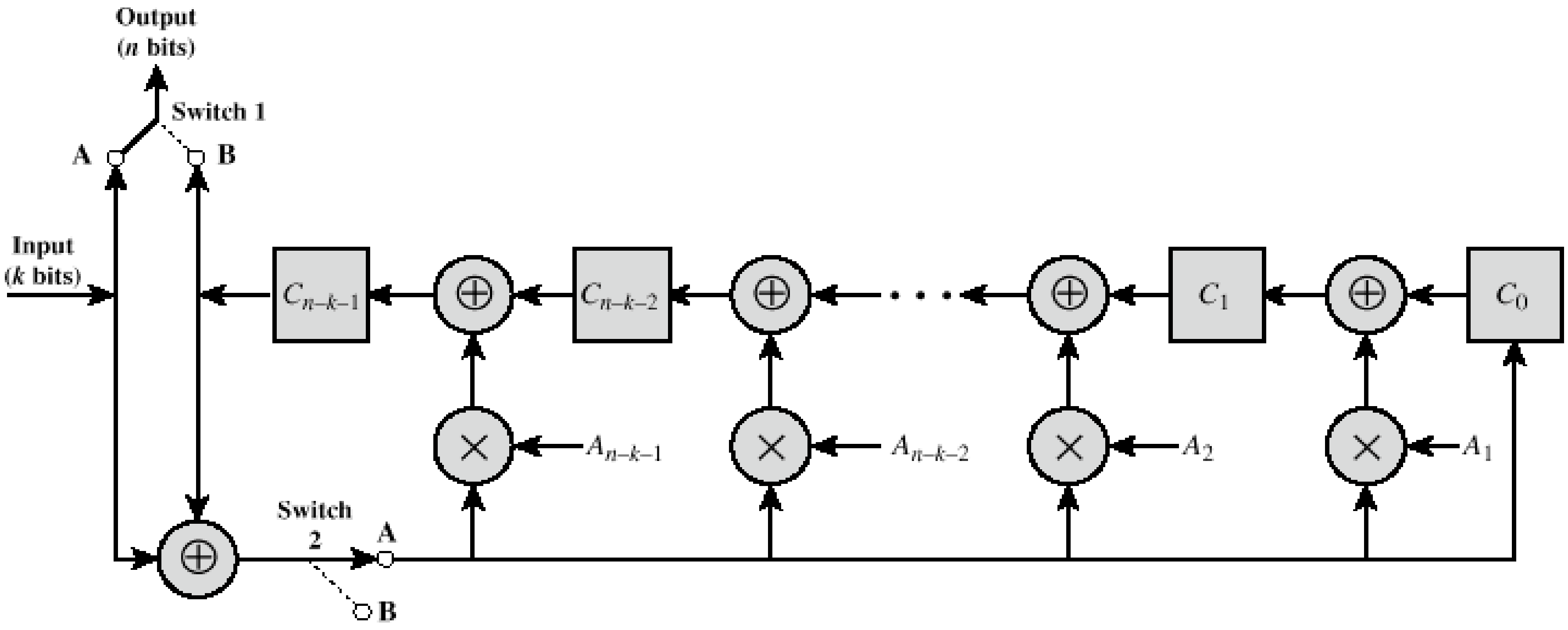


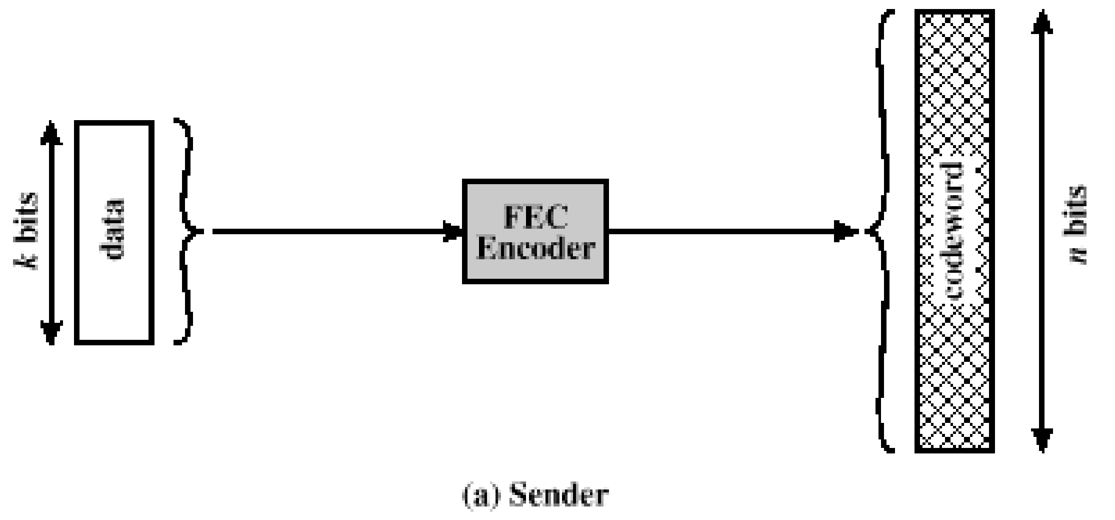
Figure 8.4 General CRC Architecture to Implement Divisor
$$1 + A_1X + A_2X^2 + \ldots + A_{n-1}X^{n-k-1} + X^{n-k}$$

Wireless Transmission Errors

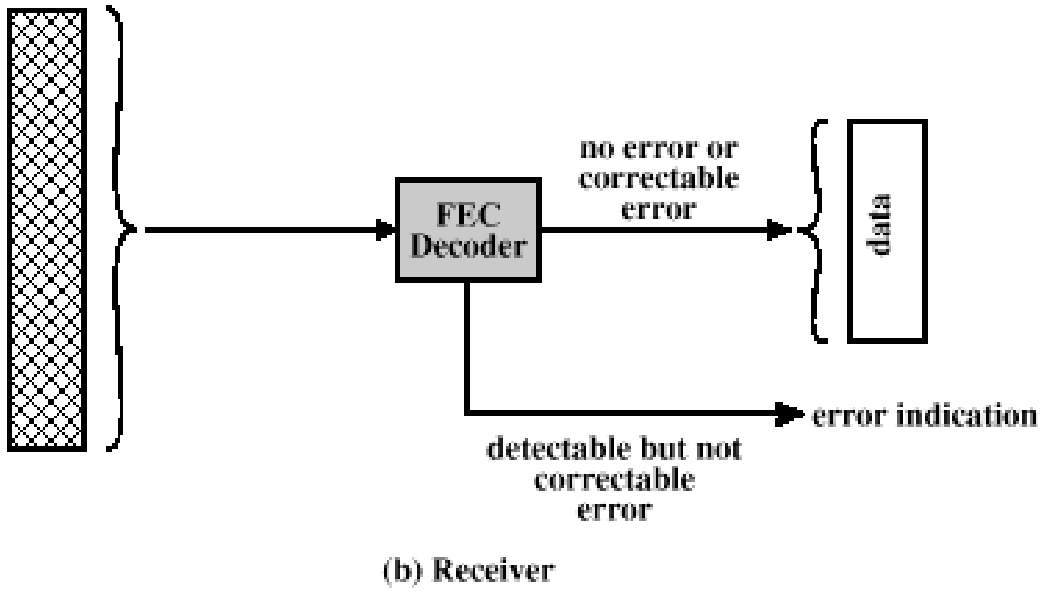
- Error detection requires retransmission
- Detection inadequate for wireless applications
  - Error rate on wireless link can be high, results in a large number of retransmissions
  - Long propagation delay compared to transmission time

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



(a) Sender


FEC Decoder
no error or correctable error
data
error indication
detectable but not correctable error

(b) Receiver

Figure 8.5 Forward Error Correction Process

FEC Decoder Outcomes

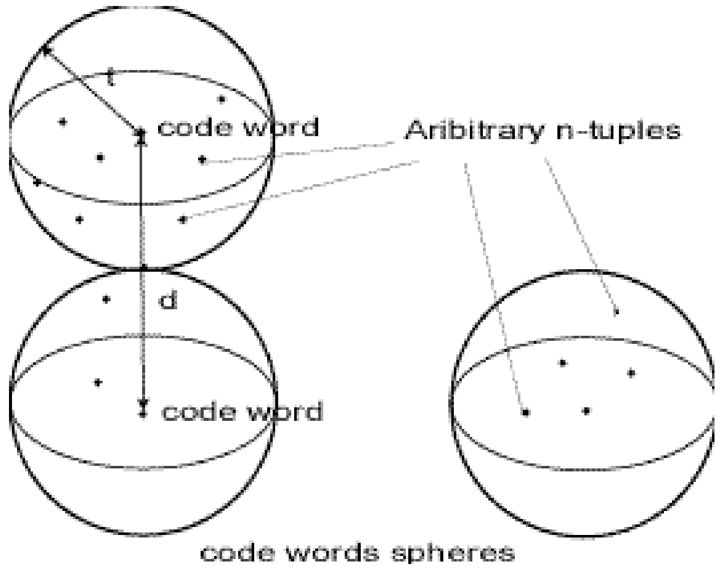
- No errors present
  - Codeword produced by decoder matches original codeword
- Decoder detects and corrects bit errors
- Decoder detects but cannot correct bit errors; reports uncorrectable error
- Decoder detects no bit errors, though errors are present

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
  - BER refers to rate of uncorrected errors

Block Codes

- The Hamming distance *d* of a Block code is the minimum distance between two code words
- Error Detection:
  - o Up to *d*-1 errors
- Error Correction:
  - o Up to $\left\lfloor \frac{d-1}{2} \right\rfloor$



Coding Gain

- Definition:
  - The coding gain is the amount of additional SNR or E_b/N_0 that would be required to provide the same BER performance for an uncoded signal
- If the code is capable of correcting at most t errors and P_{UC} is the BER of the channel without coding, then the probability that a bit is in error using coding is:

$$P_{CB} \cong \frac{1}{n}\sum_{i=t+1}^{n} i\binom{n}{i}P_{UC}^{i}(1-P_{UC})^{n-i}$$

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 Block Codes

- Definition:
  - An (n, k) linear code C is called a *cyclic code* if every cyclic shift of a code vector in C is also a code vector
  - Codewords can be represented as polynomials of degree n. For a cyclic code all codewords are multiple of some polynomial $g(X)$ modulo X^n+1 such that $g(X)$ divides X^n+1. $g(X)$ is called the generator polynomial.
- Examples:
  - Hamming codes, Golay Codes, BCH codes, RS codes
  - BCH codes were independently discovered by Hocquenghem (1959) and by Bose and Chaudhuri (1960)
  - Reed-Solomon codes (non-binary BCH codes) were independently introduced by Reed-Solomon

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

Cyclic Block Codes

- A cyclic Hamming code of length 2^m-1 with m>2 is generated by a primitive polynomial $p(X)$ of degree m
- Hamming code (31, 26)
  - $g(X) = 1 + X^2 + X^5$, $l = 3$
- Golay Code:
  - cyclic code (23, 12)
  - minimum distance 7
  - generator polynomials: either $g_1(X)$ or $g_2(X)$

$$g_1(X) = 1 + X^2 + X^4 + X^5 + X^6 + X^{10} + X^{11}$$
$$g_2(X) = 1 + X + X^5 + X^6 + X^7 + X^9 + X^{11}$$

BCH Codes

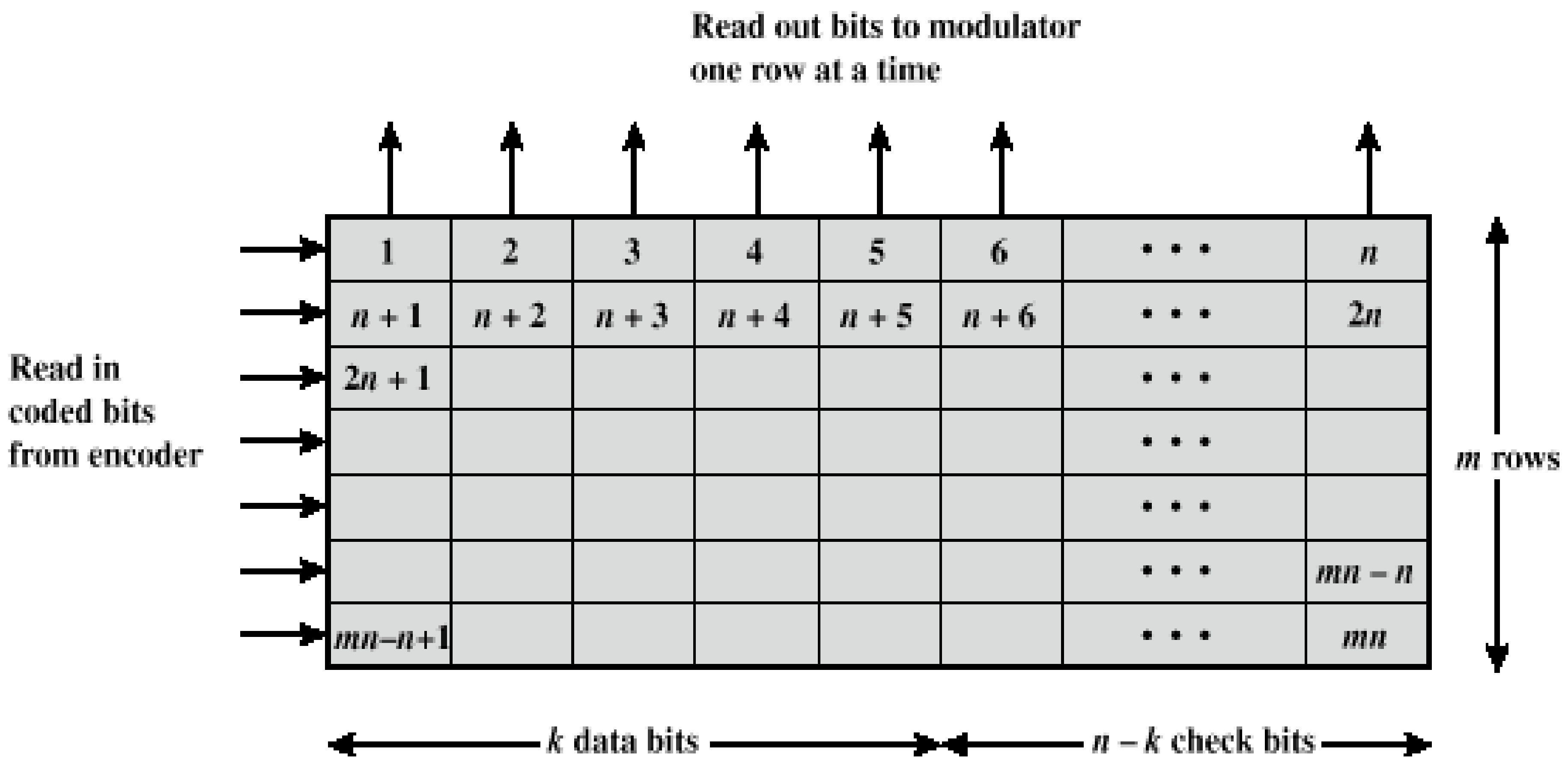
- For positive pair of integers *m* and *t*, a (*n*, *k*) BCH code has parameters:
  - Block length: $n = 2^m - 1$
  - Number of check bits: $n - k <= mt$
  - Minimum distance: $d_{\min} >= 2t + 1$
- Correct combinations of *t* or fewer errors
- Flexibility in choice of parameters
  - Block length, code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

Reed-Solomon Codes

- Subclass of non-binary 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

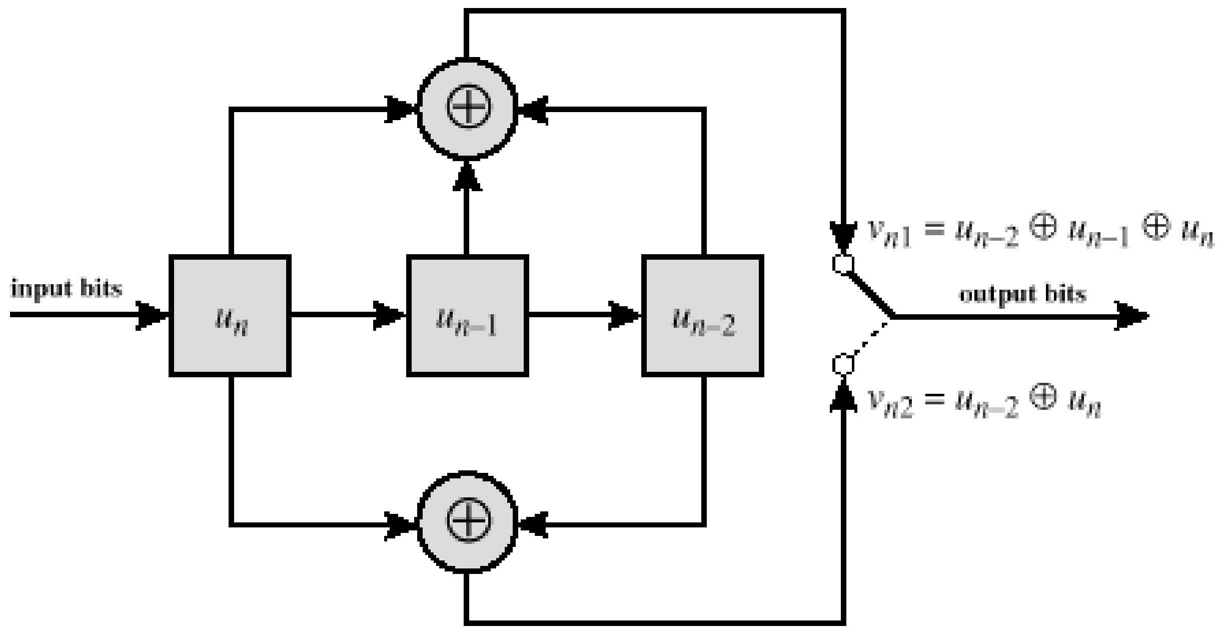


Note: The numbers in the matrix indicate the order in which bits are read in.
Interleaver output sequence: 1, $n + 1$, $2n + 1$, …

Figure 8.8 Block Interleaving

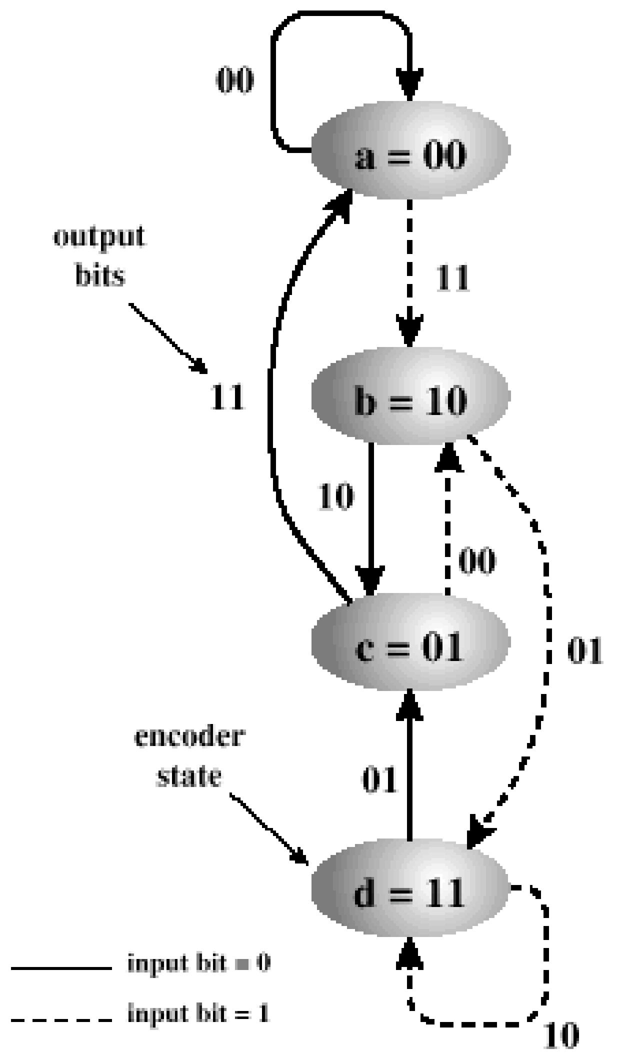
Convolutional Codes

- 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



input bits
u_n
u_{n-1}
u_{n-2}
$v_{n1} = u_{n-2} \oplus u_{n-1} \oplus u_n$
output bits
$v_{n2} = u_{n-2} \oplus u_n$

(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

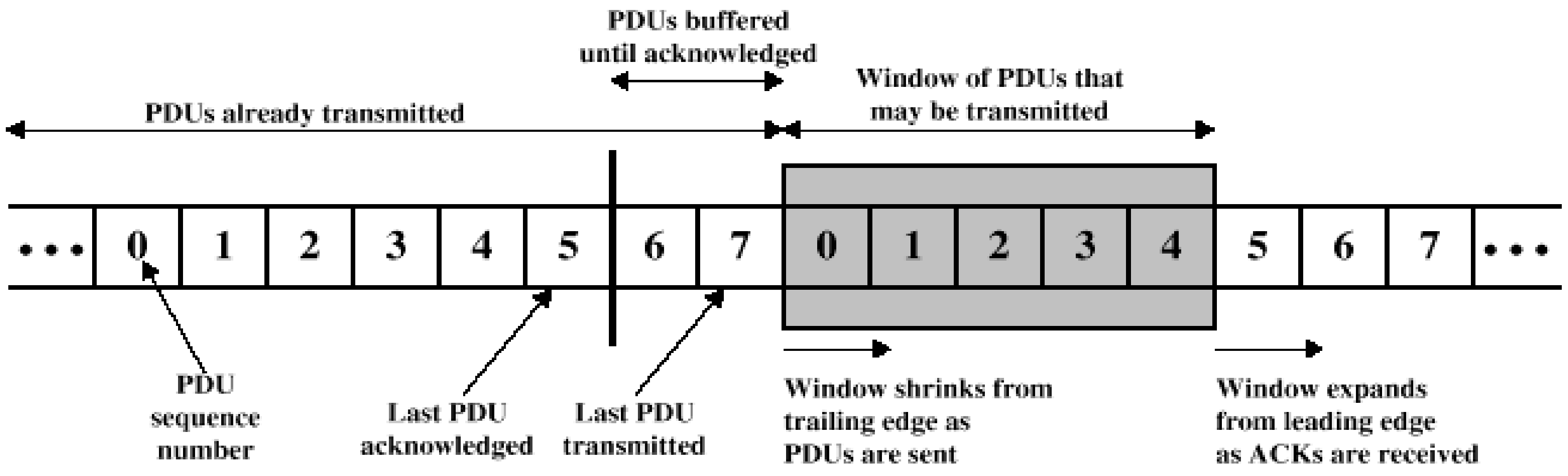
- Mechanism used in data link control and transport protocols
- Relies on use of an error detection code (such as CRC)
- Flow Control
- Error Control

Flow Control

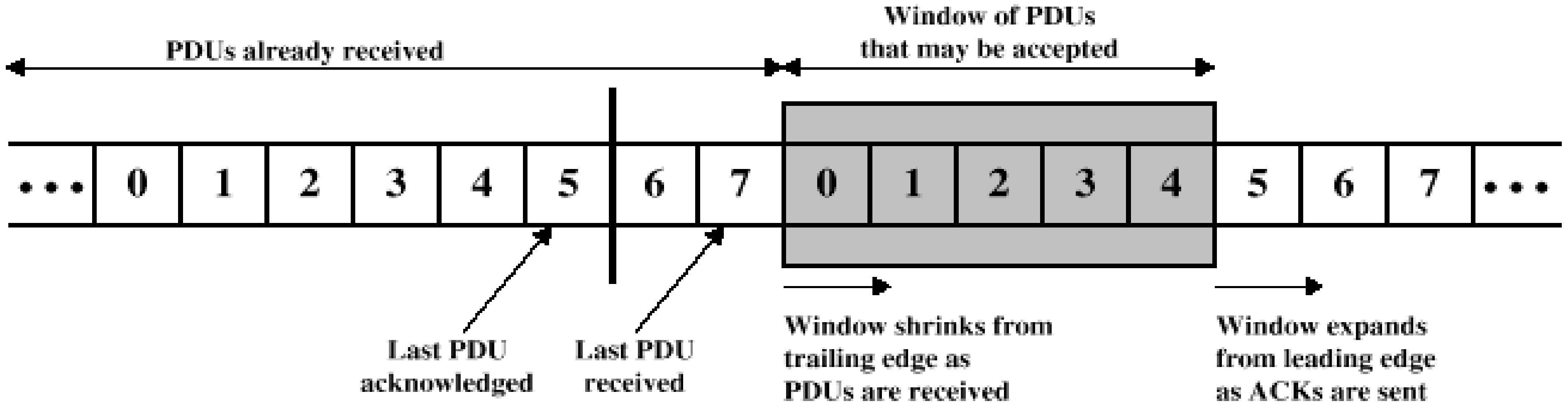
- ❑ Assures that transmitting entity does not overwhelm a receiving entity with data
- ❑ Protocols with flow control mechanism allow multiple PDUs in transit at the same time
- ❑ PDUs arrive in same order they're sent
- ❑ Sliding-window flow control
  - o Transmitter maintains list (window) of sequence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


PDUs buffered until acknowledged
PDUs already transmitted
Window of PDUs that may be transmitted
• • • 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 • • •
PDU sequence number
Last PDU acknowledged
Last PDU transmitted
Window shrinks from trailing edge as PDUs are sent
Window expands from leading edge as ACKs are received

(a) Sender's perspective


PDUs already received
Window of PDUs that may be accepted
• • • 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 • • •
Last PDU acknowledged
Last PDU received
Window shrinks from trailing edge as PDUs are received
Window expands from leading edge as ACKs are sent

(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
- Techniques:
  - Timeouts
  - Acknowledgments
  - Negative acknowledgments

Hybrid ARQ

- Combining error correction and error detection
  - Chase combining
  - Incremental redundancy
- Chase combining (Type I)
  - At receiver, decoding done by combining retransmitted packets
- Incremental redundancy (Type II/III)
  - First packet contains information and selected check bits
  - Subsequent packets contain selected check bits
  - Receiver decodes by combining all received bits
- Commonly used in wireless networks