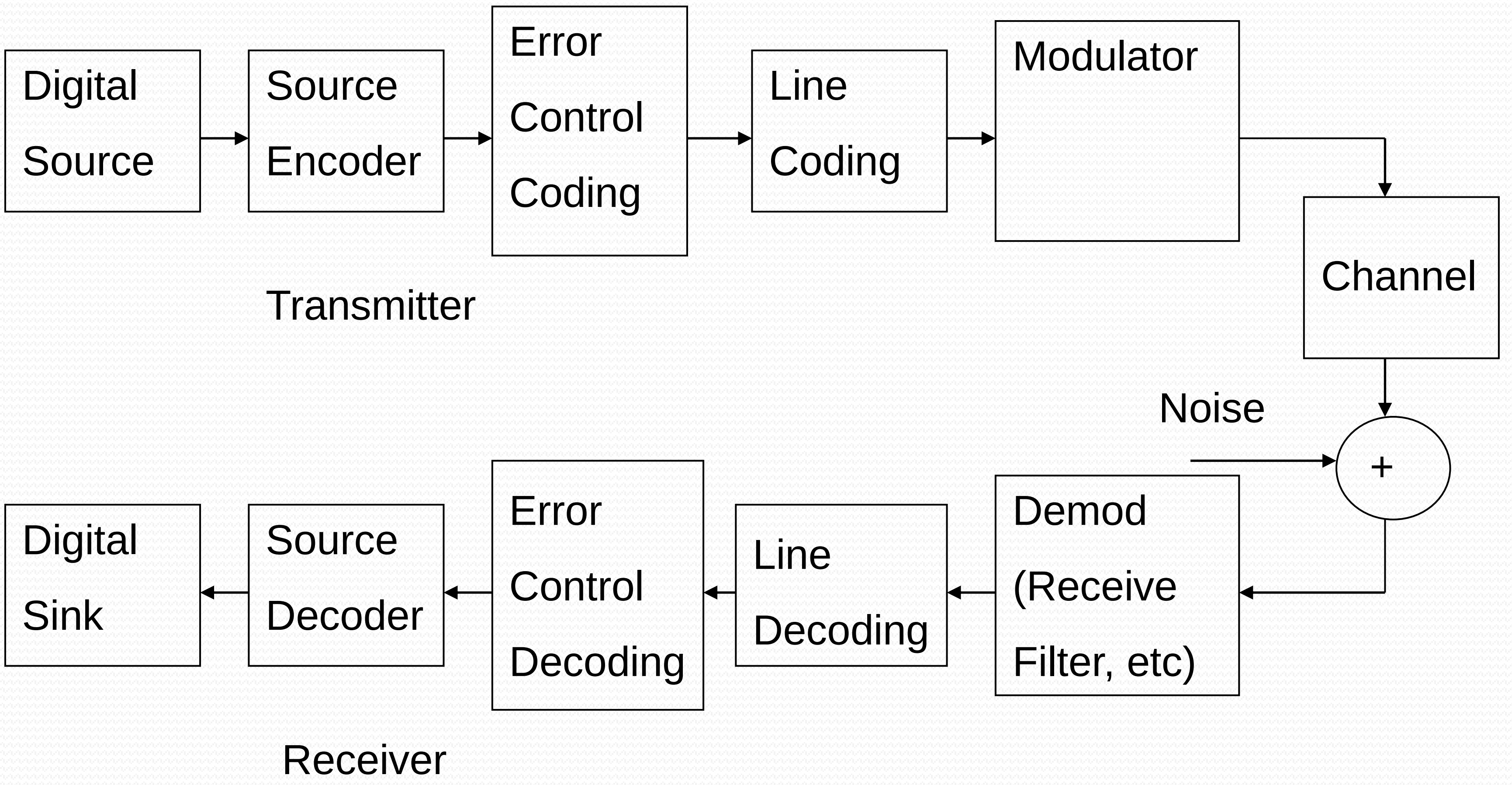
INFORMATION THEORY AND CODING

Error Control Coding-Channel Coding

- When data are transmitted through a physical channel, errors are bound to occur from time to time.
- It is desirable to enable the receiver to detect or even correct some of the errors.
- This is the purpose of **error control coding or channel coding**.

Error Control Coding



Types of Errors

- **Types of Error**

1, Single Bit Error : Only 1 bit of a given data unit is changed from 1 to 0 or from 0 to 1.

Single bit errors are the least likely type of error in serial data transmission.

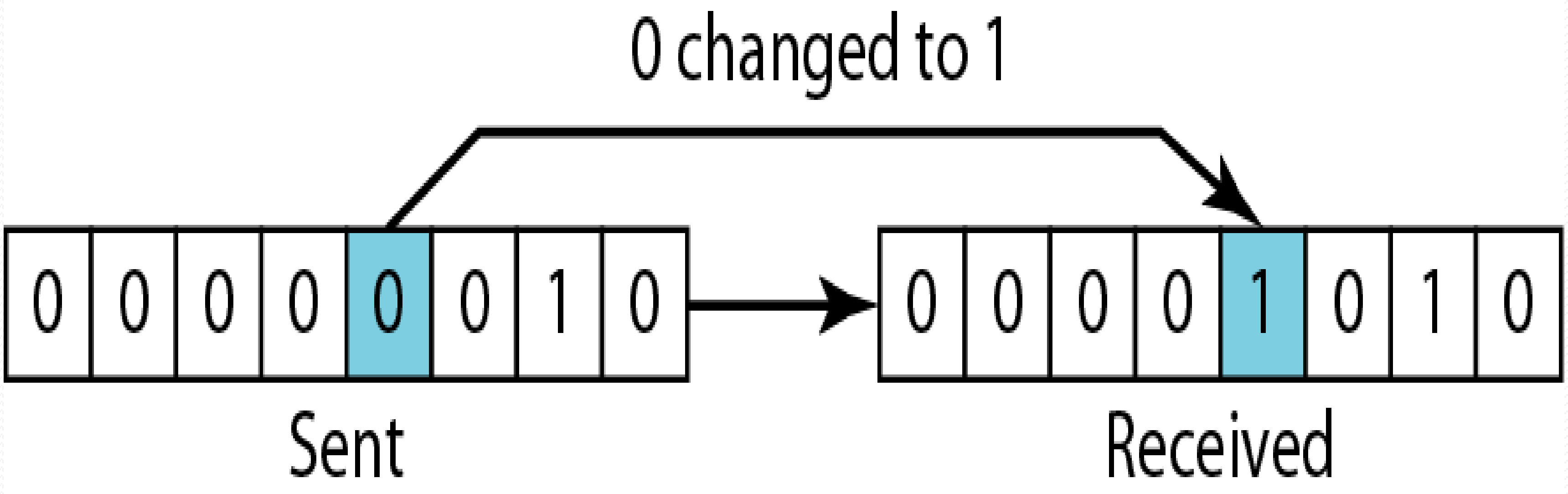
2, Burst Error : 2 or more bit of a given data unit is changed from 1 to 0 or from 0 to 1.

The length of the burst is measured form the first corrupted bit to last corrupted bit.

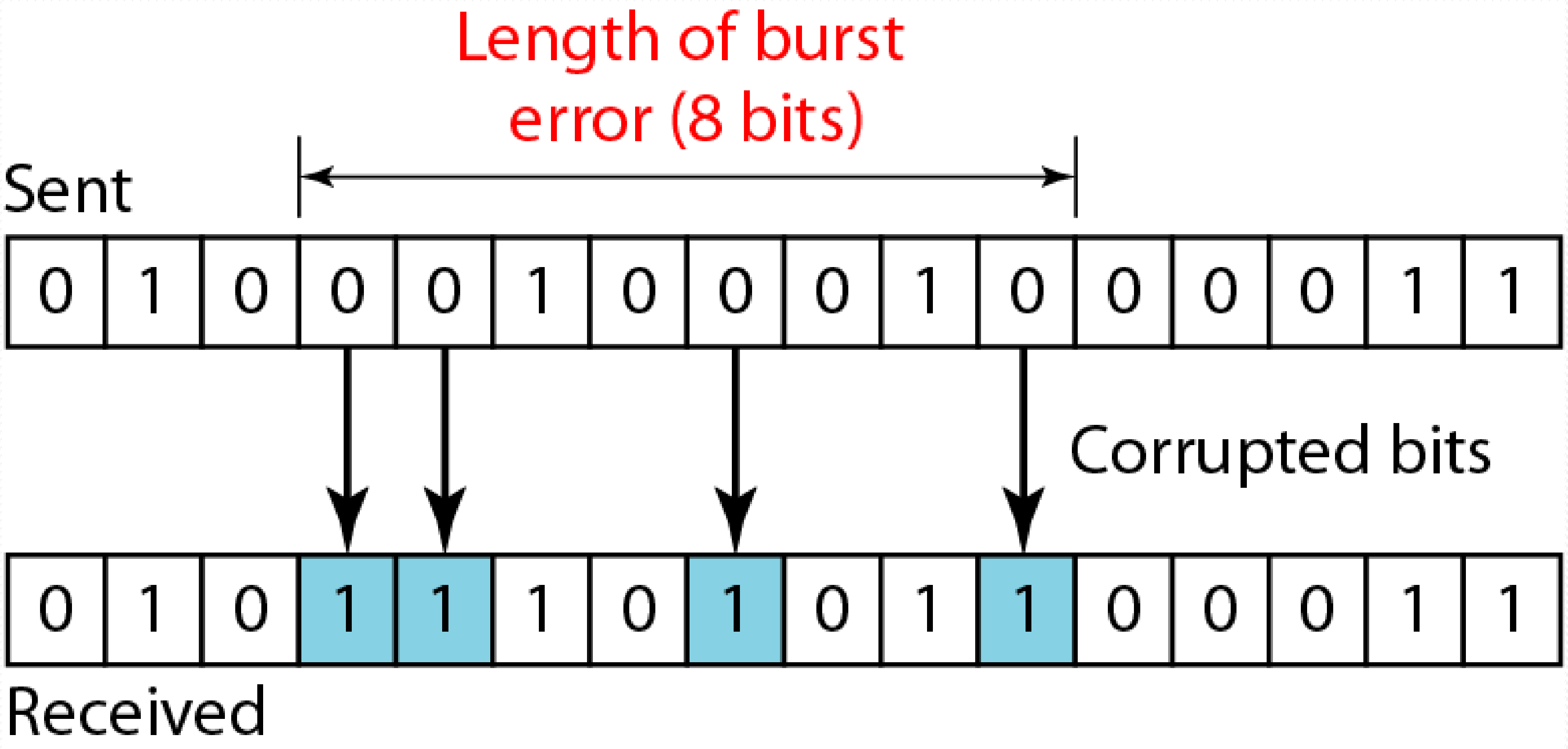
Burst Errors

- A burst error is more likely to occur than a single bit because the duration of noise is normally longer than the duration of 1 bit.
- The number of bits affected depends on **the data rate and duration of noise**. If we are sending data at 1 kbps, a noise of 1/100 second can affect 10 bits, if we are sending data at 1 Mbps, the same noise can affect 10,000 bits.

Types of Errors



Types of Errors



Types of Errors

- What is the maximum effect of a 2 ms burst of noise on data transmission at the following rates?

  (1) 2000 bps (2) 200 kbps

Methods of Controlling Errors

Redundancy :

- To detect or correct errors, we need to send extra (redundant) bits with data .
- In **error detection** , we are looking only to see if any error has occurred.
- In **error correction**, we need to know the exact number of bits that are corrupted and their location in the message.

Two methods of error correction :

1, **Forward error correction** : Receiver tries to guess the message by using redundant bits.

2, **Retransmission** : Receiver detects the occurrence of an error and asks the sender to resend the message.

- **Redundancy is achieved through various coding schemes.**
- The sender adds redundant bits through process that creates a relationship between the redundant bits and the actual data bits. The receiver checks the relationship between the two sets of bits to detect or correct errors.

Types of Codes

- Two broad categories of coding scheme

  1, **Linear Block Coding** 2, **Cyclic Coding**

- In Linear block coding, we divide our message into blocks, each of k bits, called dataword We add m redundant bits to each block to make the length n = k + m. The resulting n-bit blocks are called codeword.
- *Such scheme is called (n,k) linear block code in ehich k data digits are transmitted by a codeword of n digits, the no of check digits are m=n-k.*
- *The code efficiency is k/n. A total of 2^n codewords is available to assign to 2^k datawords.*

Block Coding

Hamming Distance:

- The hamming distance between two words (of the same size) is the **number of differences between the corresponding bits in the words**.
- Hamming distance is the number of bit positions in which the two bits are different.
- Can easily found by applying the XOR operation on the two words and count the no of 1s in the result.
- The minimum hamming distance is the smallest Hamming distance between all possible pairs in a set of words.

Block Coding

- The Hamming distance d(000, 011) is 2 because

$$000 \oplus 011 \text{ is } 011 \text{ (two 1s)}$$

- The Hamming distance d(10101, 11110) is 3 because

$$10101 \oplus 11110 \text{ is } 01011 \text{ (three 1s)}$$

Block Coding

- Find the <u>minimum Hamming distance</u> of the coding scheme in following Table

Datawords	*Codewords*
00	000
01	011
10	101
11	110

Block Coding

- Any coding scheme needs to have at least three parameters:
- The codeword size **n** , The dataword size **k**, and the minimum Hamming distance $\mathbf{d_{min}}$

To guarantee the *detection* of up to t errors in all cases, the minimum Hamming distance between all valid codewords must be $d_{min} = t + 1$

To guarantee correction of up to *t* errors in all cases, the minimum Hamming distance in a block code must be $d_{min} = 2t + 1$.

Hamming Codes

- These codes were originally designed with $d_{min} = 3$, which means that **they can detect up to two errors or correct one single error.**

- E.g. (7,4) hamming code, (6,3) hamming code

- These are **binary, signle error correcting, linear block** codes.

Problem:

- (6,3) hamming code
- Linear block code with minimum hamming distance of 3.
- It can correct 1 bit error and detect up to 2 bit error.

Error Correcting Codes

ERROR-CORRECTING CODES

Code efficiency

$\eta = k/n$

Table 16.1
Some examples of error correcting codes

	n	k	Code	Code Efficiency (or Code Rate)
Single-error correcting, $t = 1$	3	1	(3, 1)	0.33
	4	1	(4, 1)	0.25
Minimum code separation 3	5	2	(5, 2)	0.4
	6	3	(6, 3)	0.5
	7	4	(7, 4)	0.57
	15	11	(15, 11)	0.73
	31	26	(31, 26)	0.838
Double-error correcting, $t = 2$	10	4	(10, 4)	0.4
	15	8	(15, 8)	0.533
Minimum code separation 5				
Triple-error correcting, $t = 3$	10	2	(10, 2)	0.2
	15	5	(15, 5)	0.33
Minimum code separation 7	23	12	(23, 12)	0.52

Hamming Codes (7,4)

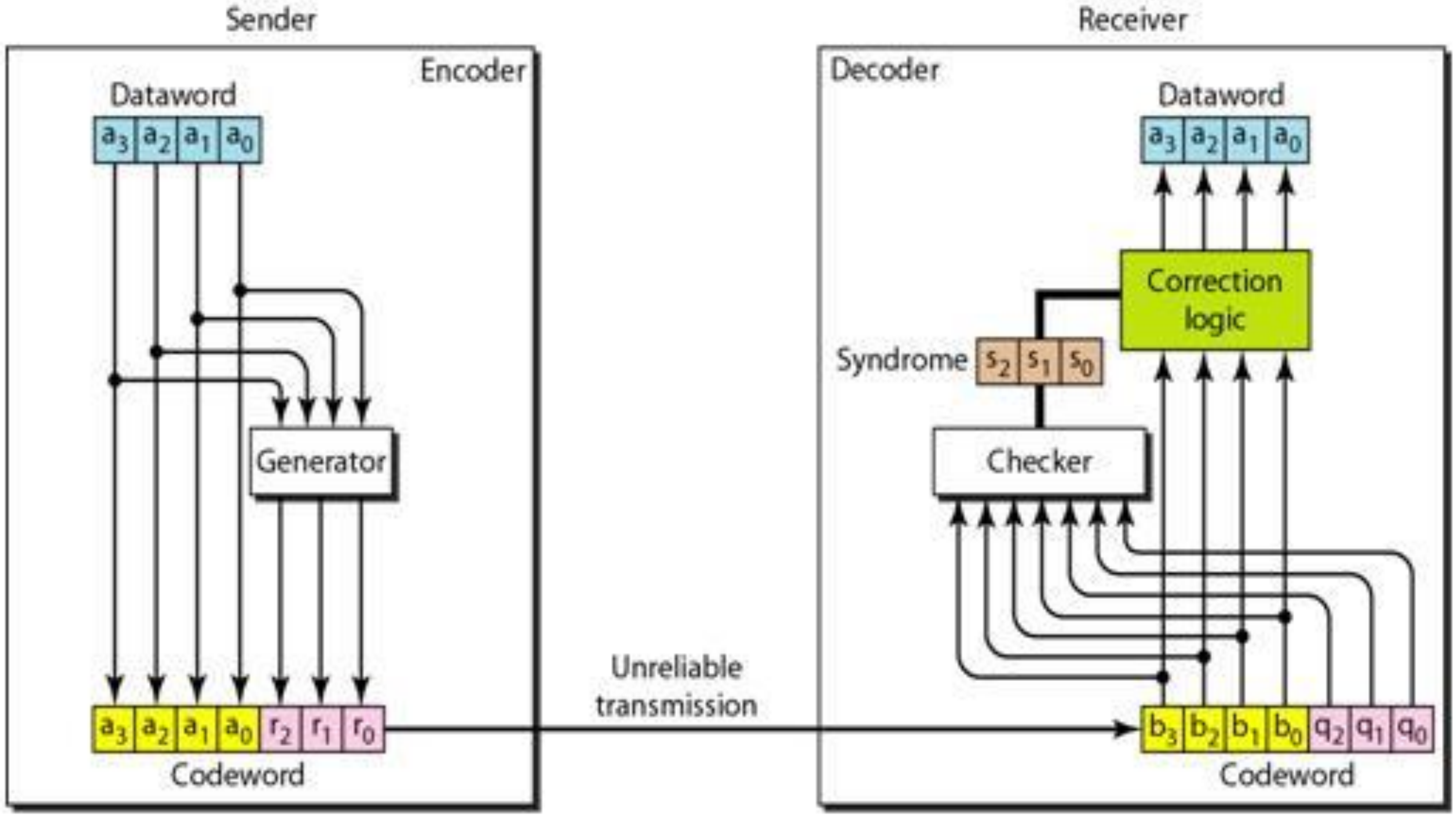
- $k = n - m$
- If $m = 3$ then $n=7$ and $k = 4$.
- This is Hamming code C(7,4) with $d_{min} = 3$
- *Hamming code C(7, 4) can :*
- *detect up to 2-bit error* $(d_{min} - 1)$
- *can correct up to 1 bit error* $(d_{min} - 1)/2$

Que. For a (7,4) linear block code:

$$\begin{matrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{matrix}$$

Obtain the parity check matrix and check whether the received code word 1110111 is correct. Also, find the corresponding decoding table.

Hamming Code (7,4)



CYCLIC CODES

A CRC code with C(7, 4)

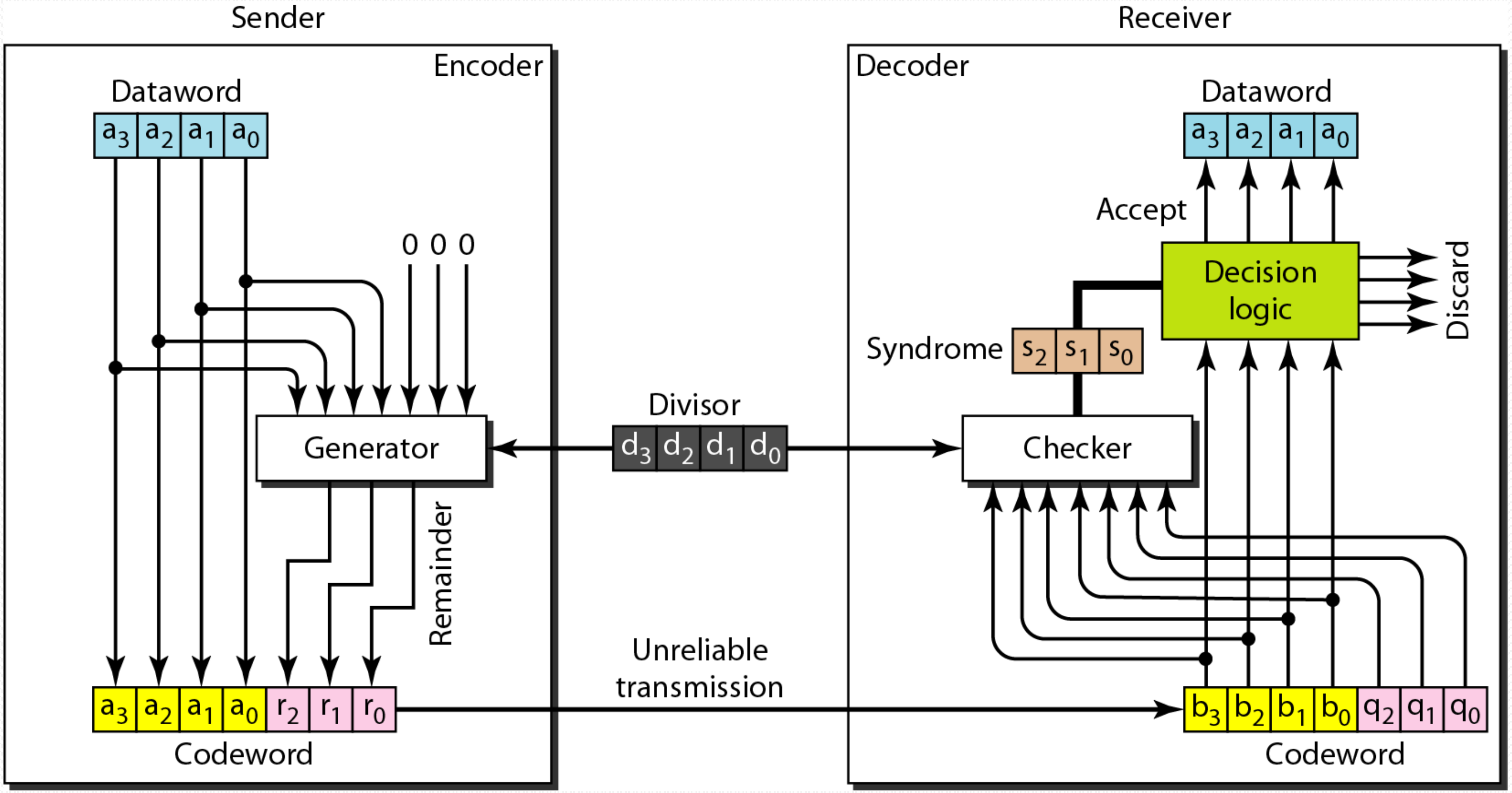
Dataword	*Codeword*	*Dataword*	*Codeword*
0000	0000000	1000	1000101
0001	0001011	1001	1001110
0010	0010110	1010	1010011
0011	0011101	1011	1011000
0100	0100111	1100	1100010
0101	0101100	1101	1101001
0110	0110001	1110	1110100
0111	0111010	1111	1111111

Prove that code in above table is cyclic, use only two tests to partially prove the fact.

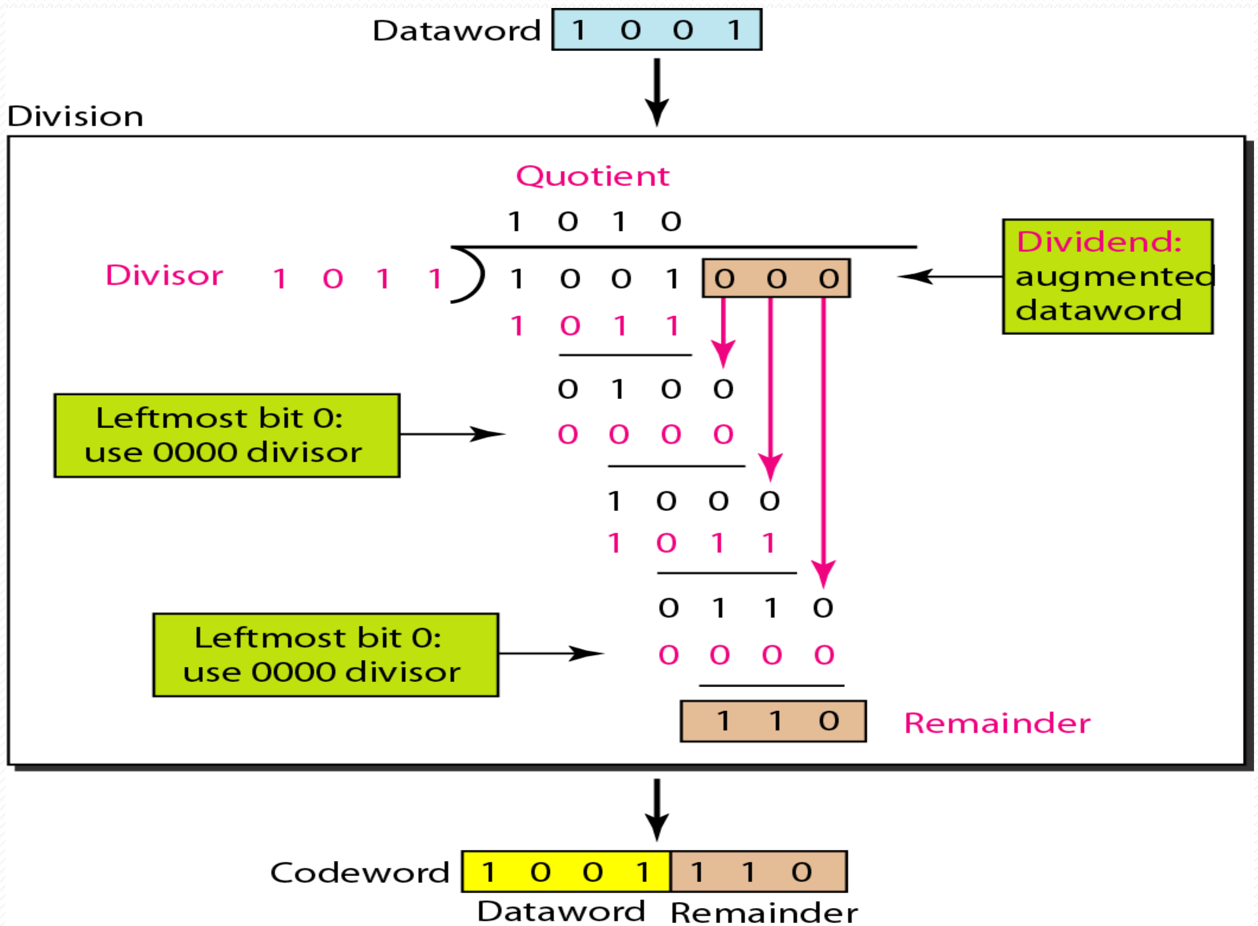
Cyclic Codes-encoding

- **CRC** (Cyclic Redundancy Check) is a category of cyclic codes that is used in LANs and WANs.
- **In the encoder the size of dataword is augmented by adding n-k zeroes to the right hand side of the word.**
- Generator divides the augmented word by the divisor(modulo-2 division). The divisor is predefined and agreed upon.
- **The quotient of the division is discarded and the remainder ($r_2r_1r_0$) is appended to the data word to create codeword.**

CYCLIC CODES



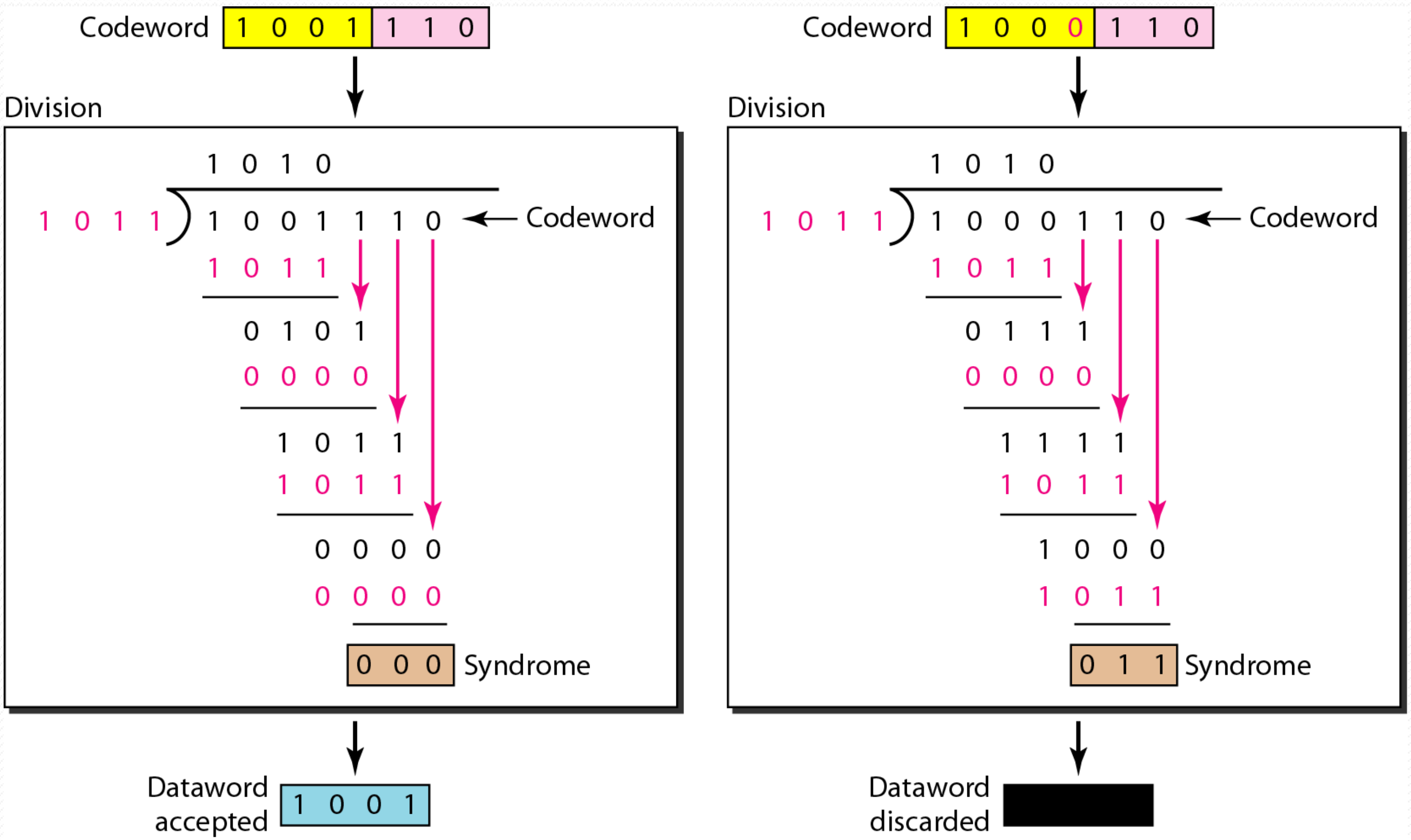
(7,4) CYCLIC CODE



Cyclic Codes-decoding

- In decoder, a copy of all n bits is fed to the checker which does the same division process as the generator.
- **The remainder produced by the checker is a syndrome of n-k bits which is fed to decision logic analyzer.**
- If the syndrome bits are all 0s, the 4 left most bits of the codeword are accepted as dataword; otherwise, the 4 bits are discarded.

CYCLIC CODES-decoding



Numerical-Cyclic Code

- Given the dataword 1010011010 and the divisor 10111

  (a) Show the generation of the codeword at sender site

  (b) Show the checking of the codeword at the receiver site (assume no error)