Cyclic codes

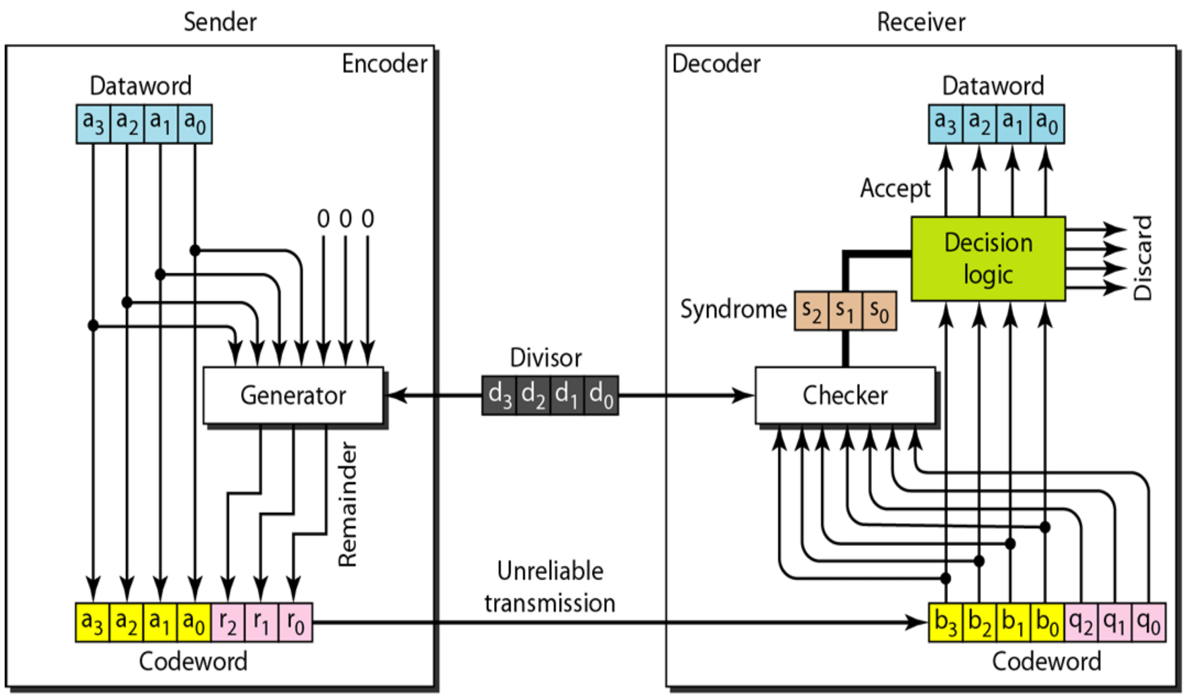
Cyclic codes are special linear block codes with one extra property that if a codeword is cyclically shifted (rotated), the result is another codeword.

For example, if 1011000 is a codeword and we cyclically left-shift (rotate instruction as studied in CAO), then 0110001 is also a codeword.

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

Now here we can check by picking any code word and try to rotate it left of right any number of bits we will again get a valid code word.

Best and most widely used example of cyclic code is CRC (cyclic redundancy check) Now we will try to generate the code word from the given data word of the above table.



In the encoder, the dataword has k bits (4 here); the codeword has n bits (7 here). The size of the dataword is augmented by adding n-k (3 here) 0s to the right-hand side of the word. The n-bit result is fed into the generator. The generator uses a divisor of size n-k+1 (4 here), predefined and agreed upon. The generator divides the augmented dataword by the divisor (modulo-2 division as described in next figure i.e. in place of subtraction XOR is done). The quotient of the division is discarded; the remainder ($r_2r_1r_0$) is appended to the dataword to create the codeword. The decoder receives the possibly corrupted codeword. A copy of all n bits is fed to the checker which is a replica of the generator. The remainder produced by the checker is a syndrome of n-k (3 here) bits, which is fed to the decision logic analyzer. The analyzer has a simple function. If the syndrome bits are all 0s, the 4 leftmost bits of the codeword are accepted as the dataword (interpreted as no error); otherwise, the 4 bits are discarded (error).

Now here if syndrome is 0 then the received codeword is accepted. But if syndrome is 0 then still there are possibility of errors. Let's take an example and suppose: -

Dataword: *d(x),* Syndrome: *s(x),* sent Codeword: *c(x),* Error: *e(x)*, Generator: *g(x)*

$$Received\ Codeword = c(x) + e(x)$$

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

Now here we are writing it as division but actually we are checking the reminder generated after division

So according to procedure done in CRC LHS of above equation must be 0 only then the received code word is accepted.

And in RHS *c(x)/g(x)* is always 0(acc. to definition) and *e(x)/g(x)* must be 0 only then the received code word is accepted.

And now 2 cases are present

a) *e(x)* is actually 0 then CRC error detection process is successful
b) *e(x)* is not 0 but still *e(x)/g(x)* is 0 i.e., the error which has occurred is divisible by *g(x)* hence here CRC error detection process is unsuccessful (but probability of such errors is very less).

Process of CRC using polynomials

Let’s take an example Single

dataword is 1001 and divisor is 1011 and we need to find the codeword

The dataword 1001 is represented as *x3 + 1*. The divisor 1011 is represented as x^2+x+1. To find the augmented dataword, we have left-shifted the dataword 3 bits (multiplying by x^3) The result is x6 + x3. Division is straightforward.

We divide the first term of the dividend, x6, by the first term of the divisor, x3. The first term of the quotient is then x6/x3, or x3. Then we multiply x3 by the divisor and subtract (subtraction here is special and same power terms are cancelled and different power terms are added) the result from the dividend. The result is x^4, with a degree greater than the divisor's degree; we continue to divide until the degree of the remainder is less than the degree of the divisor.

Key thing in CRC is the divisor i.e. the generator polynomial and it must be chosen in such a way that it can detect errors with very high probability. And sender as well as receiver must know the devisor in advance. Some of the standard CRC divisors are shown in the following table.

Name	Polynomial	Application
CRC - 8	x^8+x^2+x+1	ATM header
CRC - 10	$x^{10}+x^9+x^5+x^4+x^2+1$	ATM AAL
CRC – 16	$x^{16}+x^{12}+x^5+1$	HDLC
CRC - 32	$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$	LANs

Advantage of CRC

Easy to implement, fast and error detection probability is very high.

Checksum

Like linear and cyclic codes, the checksum is based on the concept of redundancy. Several protocols still use the checksum for error detection as we will see while studying higher layers (checksum is not used in DLL protocols but we are still studying it here just to complete the discussion of error detection)

However, the tendency is to replace it with a CRC because it is not as good as CRC. (This means that the CRC is also used in layers other than the data link layer).

General idea

Suppose our data is a list of five 4-bit numbers that we want to send to a destination. In addition to sending these numbers, we send the sum of the numbers. For example, if the set of numbers is (7, 11, 12, 0, 6), we send (7, 11, 12,0,6,36), where 36 is the sum of the original numbers. The receiver adds the five numbers and compares the result with the sum. If the two are the same, the receiver assumes no error, accepts the five numbers, and discards the sum. Otherwise, there is an error somewhere and the data are not accepted.

We can make the job of the receiver easier if we send the negative (complement) of the sum, called the checksum. In this case, we send (7, 11, 12,0,6, -36). The receiver can add all the numbers received (including the checksum). If the result is 0, it assumes no error; otherwise, there is an error.

The previous example has one major drawback. All of our data can be written as a 4-bit word (they are less than 15) except for the checksum. One solution is to use one's complement arithmetic. In this arithmetic, we can represent unsigned numbers between 0 and 2n - 1 using only n bits. If the number has more than n bits, the extra leftmost bits need to be added to the n rightmost bits (wrapping). In one's complement arithmetic, a negative number can be represented by inverting all bits (changing a 0 to a 1 and a 1 to a 0).

Internet protocol (IP) uses 16-bit checksum for its header only

However, it is impractical to use of size of data is very large. For example if we have 16 bit checksum and size of data is 1MB then we have approx 512000 groups of 16 bit data and then we have to add all these which needs hell lots of calculations.

Disadvantages of checksum

The main disadvantage is if data is large then this method is time consuming and if one of the data element is incremented and other is decremented then the total sum of data remain constant hence it will never be detected by checksum.

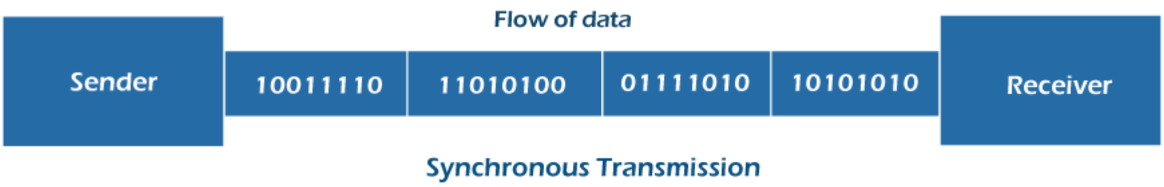
Synchronous And Asynchronous Transmission

Before starting the topic difference between ***synchronous*** and ***asynchronous transmission***, you must know about the transmission. The action of transferring data or anything from one place to other is referred to as transmission. It is a method of sharing data between two devices linked by a network, also known as communication mode. Synchronous and asynchronous transmissions are the two main types of transmission used in computer networking.

In both synchronous and asynchronous transmission, data is sent between the transmitter and the receiver based on a clock pulse utilized for synchronization. These serial data transmission techniques are both known as synchronous transmission.

What is Synchronous Transmission?

Synchronous transmission is an effective and dependable method of sending huge amounts of data. The data travels in a full-duplex method in the type of frames or blocks in Synchronous Transmission. The transmitter and receiver must be synced so that the sender knows where to start the new byte. As a result, every data block is marked with synchronization characters, and the receiving device obtains the data until a certain ending character is found.



It also allows connected devices to interact in real time. Synchronous transmission can be seen in chat rooms, video conferencing, telephonic talks, and face-to-face interactions. It utilizes the broad-band and voice band channels because they enable quicker speeds of up to ***1200 bps*** and meet the objective of high data transfer speed.

Advantages and Disadvantages of Synchronous Transmission

There are various advantages and disadvantages of synchronous transmission. Some advantages and disadvantages of synchronous transmission are as follows:

Advantages

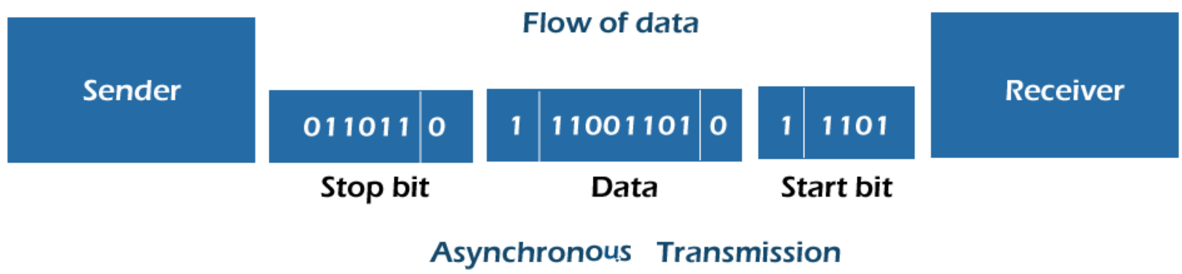
1. It aids the user in transferring a huge amount of data.
2. Every Byte is sent without a pause before the next.
3. It also helps to reduce timing errors.
4. It allows connected devices to communicate in real-time.

Disadvantages

1. The sender and receiver must operate at the same clock frequency simultaneously.
2. The accuracy of the received data is determined by the receiver's capacity to count the received bits precisely.

What is Asynchronous Transmission?

Asynchronous transmission is also referred to as start and stop transmission. It sends data from the transmitter to the receiver using the flow control approach and synchronizes data between the source and destination without utilizing a clock.



This transmission technique sends ***8 bits*** or one letter at a time. In this system, each character transmits the start bit before the transmission process begins, and it also transmits the stop bit when the character is sent. The total number of bits is ***10***, including the character, start, and stop bits.

It employs character-based synchronization for the receiving terminal to synchronize with receiving data on a character. It is easy, quick, and inexpensive and doesn't need two-way communication. Asynchronous transmission is demonstrated via letters, televisions, emails, forums, and radios.

Asynchronous transmission makes use of voice-band channels that are narrow and operate at a slower speed. In this case, the transmitting device operates manually or intermittently.

Advantages and Disadvantages of Asynchronous Transmission

There are various advantages and disadvantages of Asynchronous transmission. Some advantages and disadvantages of Asynchronous transmission are as follows:

Advantages

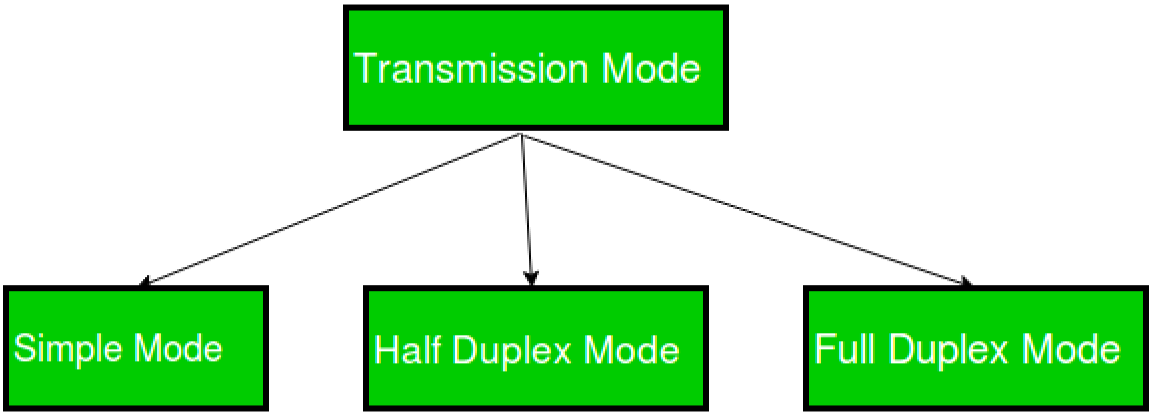
1. It doesn't require synchronizing the receiver and transmitter.
2. It is a very flexible technique of data transmission.
3. This kind of transmission is simple to implement.
4. It allows users to send signals from sources with varying bit rates.
5. When the data byte transmission is complete, the data transmission may be resumed.

Disadvantages

1. The timing errors may occur because synchronization is difficult to determine.
2. These bits could be mistakenly recognized due to the noise on the channel.
3. The start and stop bits are extra bits that must be utilized in asynchronous transmission.
4. It transmits information at a slower rate.

Transmission Modes in Computer Networks (Simplex, Half-Duplex and Full-Duplex)

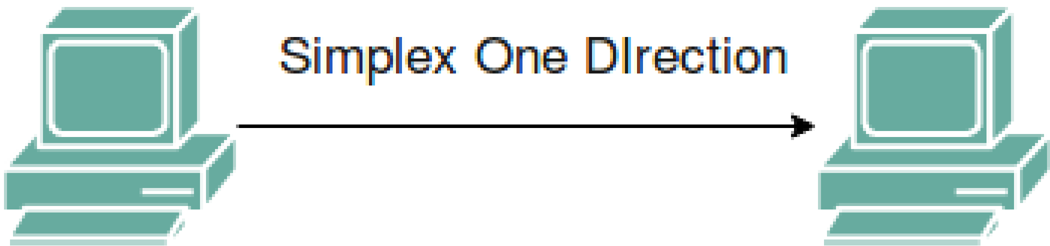
Transmission mode means transferring data between two devices. It is also known as a communication mode. Buses and networks are designed to allow communication to occur between individual devices that are interconnected.



1. Simplex Mode –

In Simplex mode, the communication is unidirectional, as on a one-way street. Only one of the two devices on a link can transmit, the other can only receive. The simplex mode can use the entire capacity of the channel to send data in one direction.

Example: Keyboard and traditional monitors. The keyboard can only introduce input, the monitor can only give the output.



Advantages:

- Simplex mode is the easiest and most reliable mode of communication.
- It is the most cost-effective mode, as it only requires one communication channel.
- There is no need for coordination between the transmitting and receiving devices, which simplifies the communication process.
- Simplex mode is particularly useful in situations where feedback or response is not required, such as broadcasting or surveillance.

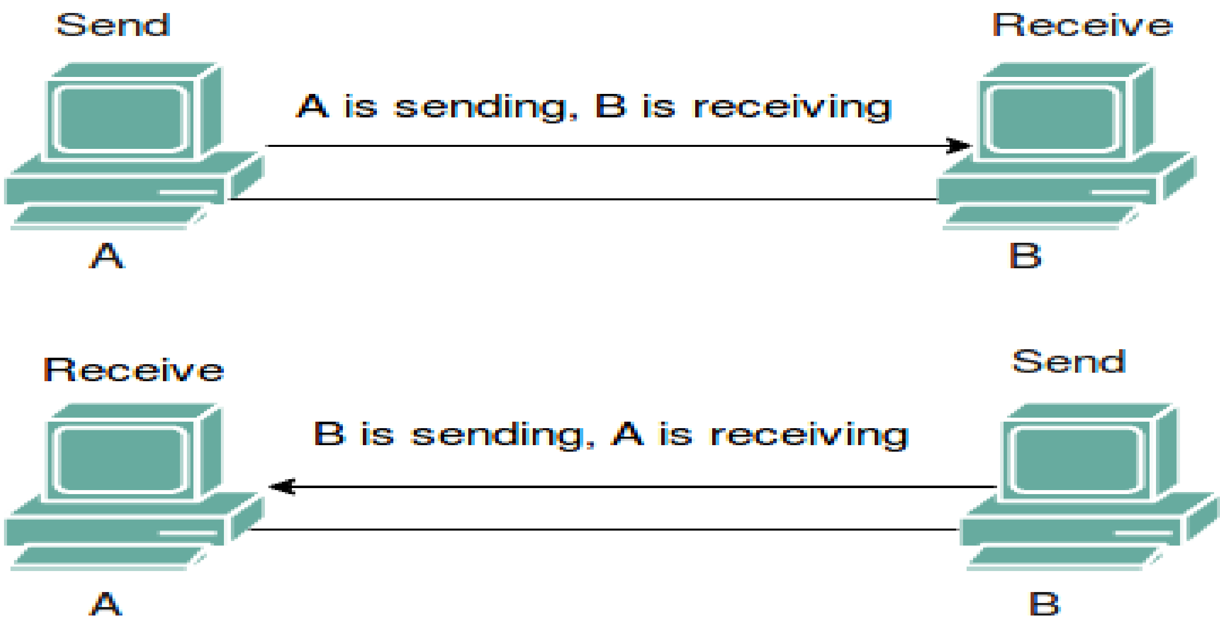
Disadvantages:

- Only one-way communication is possible.
- There is no way to verify if the transmitted data has been received correctly.
- Simplex mode is not suitable for applications that require bidirectional communication.

2. Half-Duplex Mode –

In half-duplex mode, each station can both transmit and receive, but not at the same time. When one device is sending, the other can only receive, and vice versa. The half-duplex mode is used in cases where there is no need for communication in both directions at the same time. The entire capacity of the channel can be utilized for each direction.

Example: Walkie-talkie in which message is sent one at a time and messages are sent in both directions.



Advantages:

Half-duplex mode allows for bidirectional communication, which is useful in situations where devices need to send and receive data.

It is a more efficient mode of communication than simplex mode, as the channel can be used for both transmission and reception.

Half-duplex mode is less expensive than full-duplex mode, as it only requires one communication channel.

Disadvantages:

Half-duplex mode is less reliable than Full-Duplex mode, as both devices cannot transmit at the same time.

There is a delay between transmission and reception, which can cause problems in some applications.

There is a need for coordination between the transmitting and receiving devices, which can complicate the communication process.

3. Full-Duplex Mode –

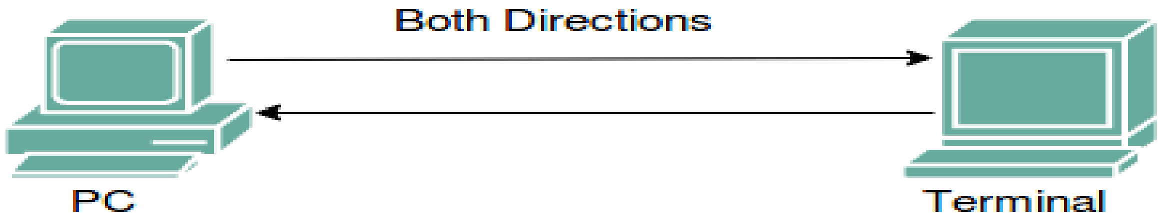
In full-duplex mode, both stations can transmit and receive simultaneously. In full_duplex mode, signals going in one direction share the capacity of the link with signals going in another direction, this sharing can occur in two ways:

Either the link must contain two physically separate transmission paths, one for sending and the other for receiving.

Or the capacity is divided between signals traveling in both directions.

Full-duplex mode is used when communication in both directions is required all the time. The capacity of the channel, however, must be divided between the two directions.

Example: Telephone Network in which there is communication between two persons by a telephone line, through which both can talk and listen at the same time.



Advantages:

Full-duplex mode allows for simultaneous bidirectional communication, which is ideal for real-time applications such as video conferencing or online gaming.

It is the most efficient mode of communication, as both devices can transmit and receive data simultaneously.

Full-duplex mode provides a high level of reliability and accuracy, as there is no need for error correction mechanisms.

Disadvantages:

Full-duplex mode is the most expensive mode, as it requires two communication channels.

It is more complex than simplex and half-duplex modes, as it requires two physically separate transmission paths or a division of channel capacity.

Full-duplex mode may not be suitable for all applications, as it requires a high level of bandwidth and may not be necessary for some types of communication.