Chapter -4

Encoding & Modulating

We must transform data into signals to send them from one place to another.

Data stored in a computer are in the form of 0s and 1s. To be carried from one place to another. (inside or outside the computer), data are usually converted to digital signals.

→ This is called digital to digital conversion. or encoding digital data into a digital signal.

→ Sometimes, we need to convert an analog signal (such as voice in a telephone conversation) into a digital signal for several reasons. such as to decrease the effect of noise. This is called analog to ~~analog~~ digital conversion or digitizing an analog signal

→ At other times, we want to send a digital signal coming out of a computer through a medium designed for an analog signal. For Example to send data from one place to another using public telephone line, the digital signal produced by the computer should be converted to an analog signal. This is called as digital to analog conversion or modulating a digital signal.

→ Often an analog signal is sent over long distances using analog media. For. Eg. voice or music from radio station, which is naturally an analog signal, is transmitted through air. However the frequency of the voice or music is not appropriate for this kind of transmission, the signal should be carried by a higher frequency signal.

This is called analog to analog conversion or modulating an analog signal.

Conversion Methods

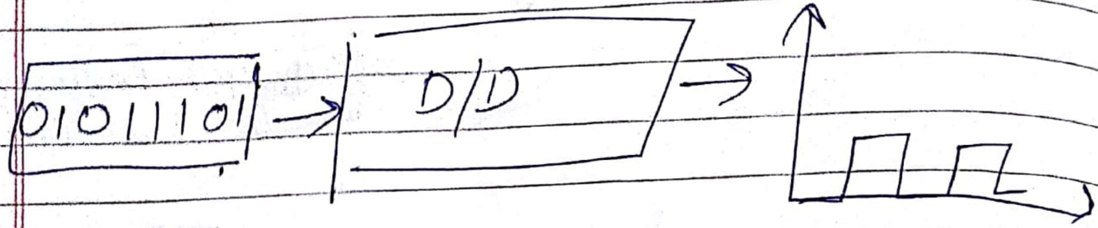
- Digital / Digital
- Analog / Digital
- Digital / Analog
- Analog / Analog

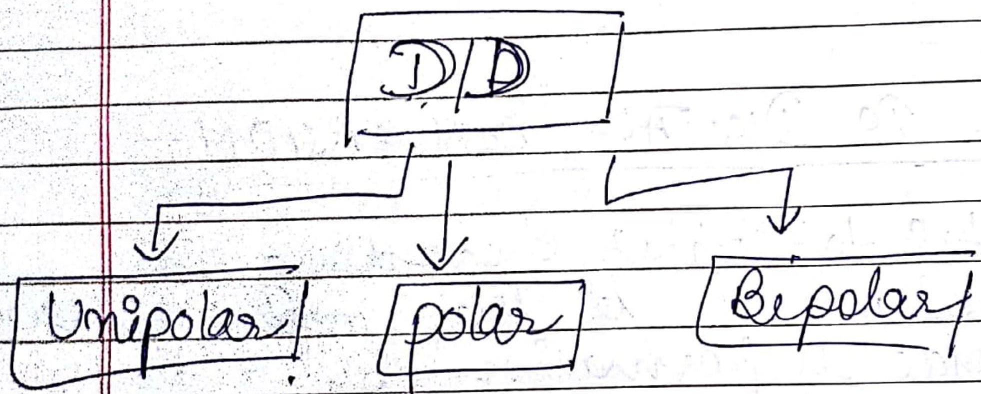
DIGITAL TO DIGITAL CONVERSION —

Digital to Digital Encoding or Conversion is the representation of digital information by a digital signal.

For Eg. when you transmit data from your computer to your printer, both original data and the transmitted data are digital.

In this type of conversion, the binary 1s and 0s generated by a computer are translated into a sequence of voltage pulses that can be propogated over a ~~wire~~ wire





Unipolar → very simple & very primitive.

In most types of encoding, one voltage level stands for binary 0 and another level stands for binary 1.

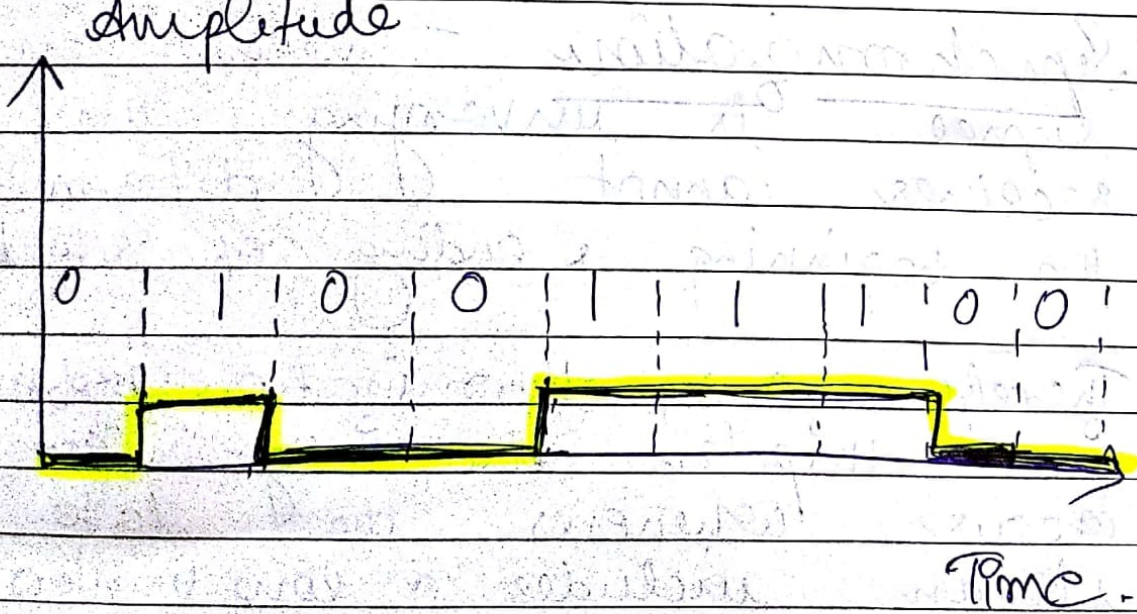
The polarity of pulse refers to whether it is positive or negative

Unipolar Encoding is so named because it uses only one polarity.

This polarity is assigned to one of the two binary states, usually 1.

The other state usually 0, is represented by zero voltage.

* Unipolar Encoding uses only one level of value



However, unipolar encoding has two problems —

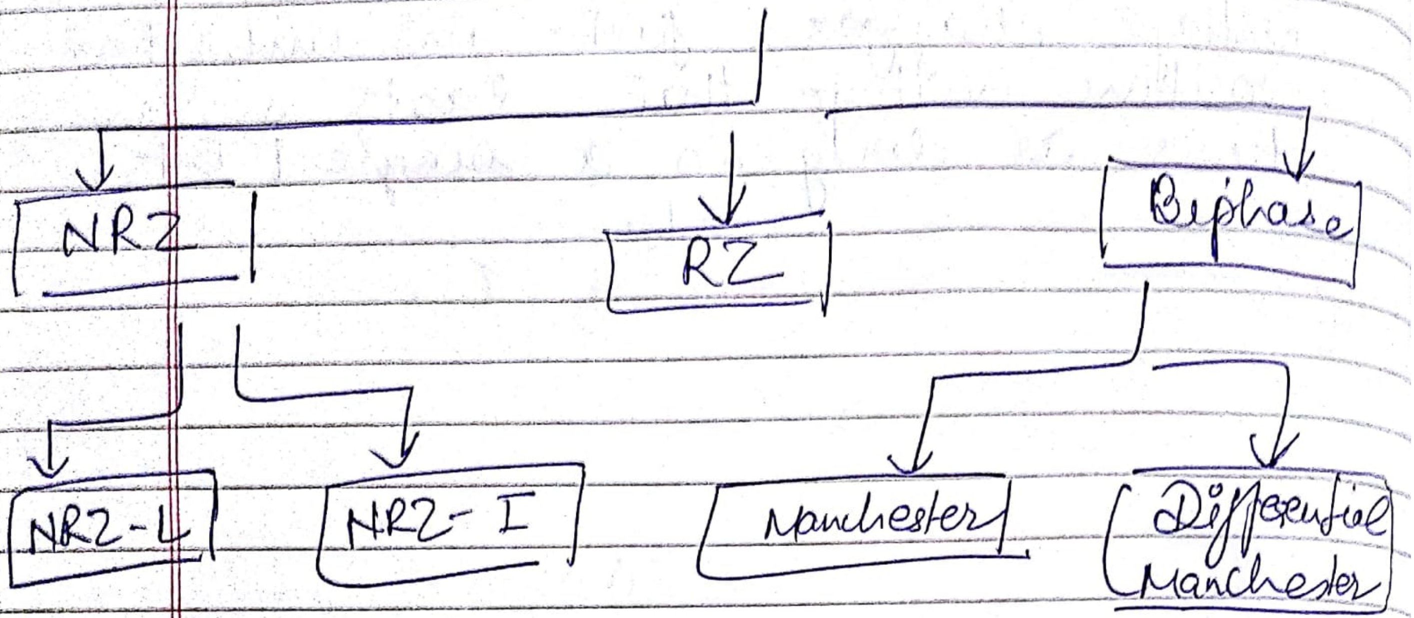
2. Synchronization - when the signal is unvarying, the receiver cannot determine the beginning & ending of each bit

Therefore a synchronization problem in unipolar encoding can occur whenever the data stream includes a long uninterrupted series of 1s & 0s.

Digital Encoding scheme ~~uses~~ use changes in voltage level to indicate change in bit type.

Polar –

Polar encoding uses two levels (positive & negative) of amplitude



NRZ :- Non Return to Zero.

In NRZ, encoding, the level of the signal is always either positive or negative

NRZ – L :- In NRZ-L encoding, the level of signal depends on the type of bit it represents.

A positive voltage usually means the bit is 0, and a negative voltage means the bit is 1. (or vice versa)

Thus the level of signal is dependent upon the state of bit

A problem of synchronization can arise when there is a long stream of 0s or 1s in the data.

The receiver receives a continuous voltage & should determine how many bits are sent by relying on its clock, which may or may not be synchronized with the sender clock.

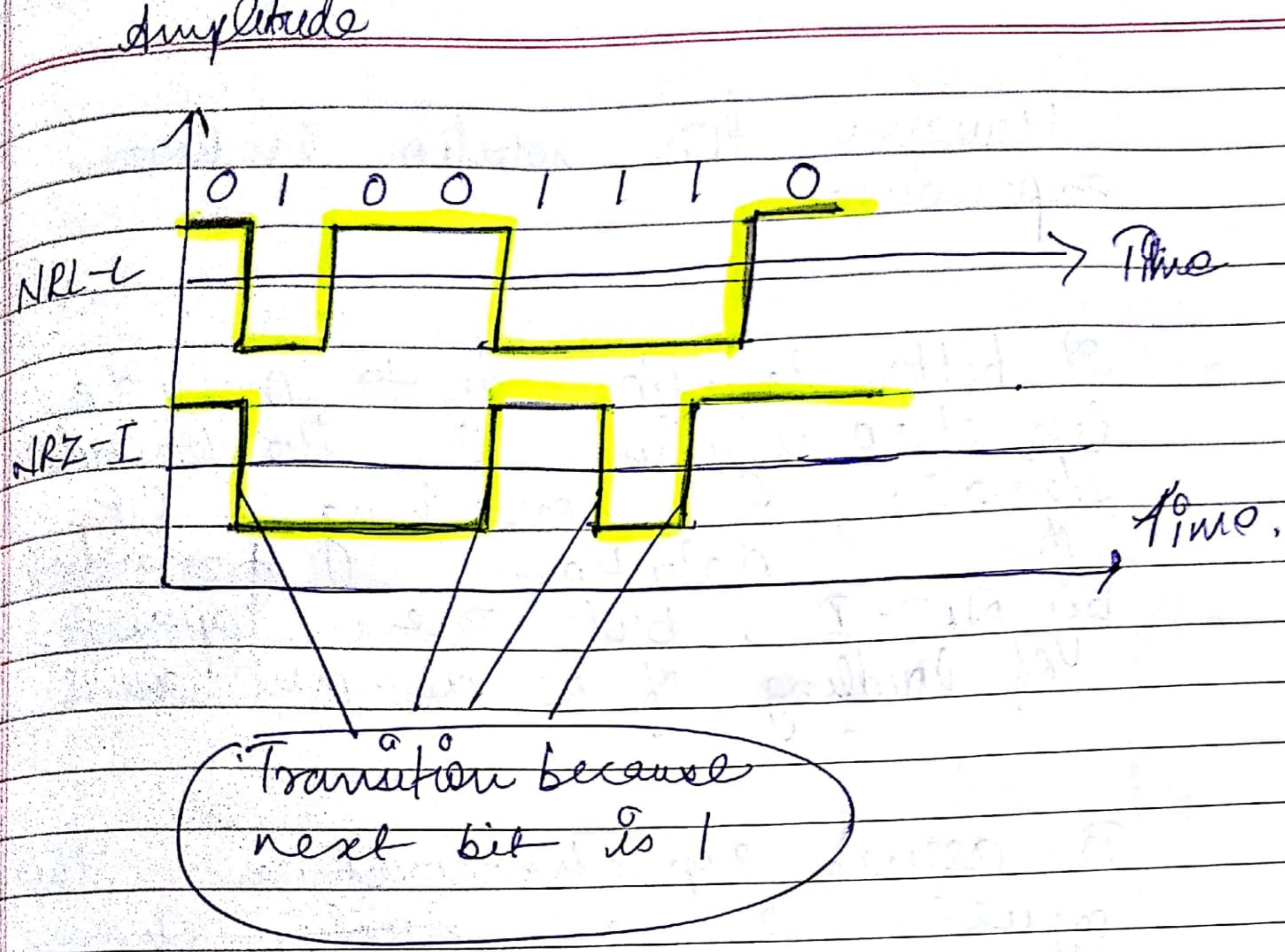
NRZ - I :- In NRZ - I, an inversion of the voltage level represents a 1 bit. It is transition between a positive & a negative voltage, not the voltages themselves, that represents a 1 bit.

Date
Page

A 0 bit is represented by no change.

NRZ - I is superior to NRZ-L due to synchronization provided by the signal change each time a 1 bit is encountered.

In NRZ - I, the signal is inverted if a 1 is encountered



Return to Zero (RZ) :— As you can see, anytime the original data contain strings of consecutive 1s or 0s, the receiver can lose its place.

As we mentioned in our discussion of unipolar encoding, one way to assure synchronization is to send a separate timing signal on separate channel.

However this solution is ~~both~~, expensive.

A better solution is to include synchronization in the encoded signal, something like the solution provided by NRZ-I, but one capable of handling of 0s as well as 1s.

To assure synchronization, there must be a ~~single change~~ signal change for each bit.

As we saw, NRZ-I accomplishes this for the sequences of 1s.

But to change with every bit, we need more than just two values

One Solution is return to zero (RZ) encoding, which uses three values - positive, Negative + zero.

In RZ, the signal changes not between bits but during each bit.

like NRZ-L, a positive voltage means 1 and a negative voltage means 0.

But unlike NRZ-L, halfway through each bit interval, the signal returns to zero.

A 1 bit is actually represented by positive to zero (⅂) and

0 bit by negative to zero (⅃)

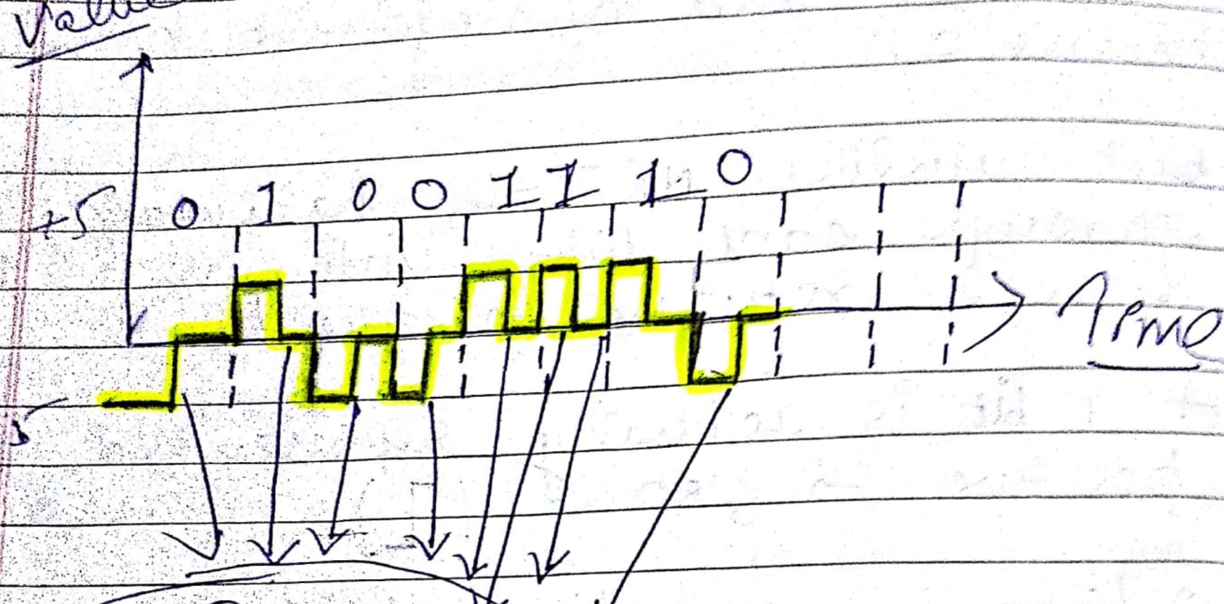
rather than by positive & negative alone.

Disadvantage -

The main disadvantage of RZ encoding is that it requires two signals changes to encode one bit and ∴ occupies more bandwidth.

Advantage :- It is most effective

A good encoded digital signal must contain a provision for synchronization



These transitions can be used for synchronizations.

Biphase:- In this method, the signal changes at the middle of the bit interval but does not return to zero.

Instead it continuous to opposite pole

Biphase Encoding

Manchester Encoding | Differential Manchester

Manchester Encoding - Manchester encoding uses the inversion at the middle of each bit interval for both synchronization & bit representation

-ve to +ve transition → binary 1

+ve to -ve transition → binary 0

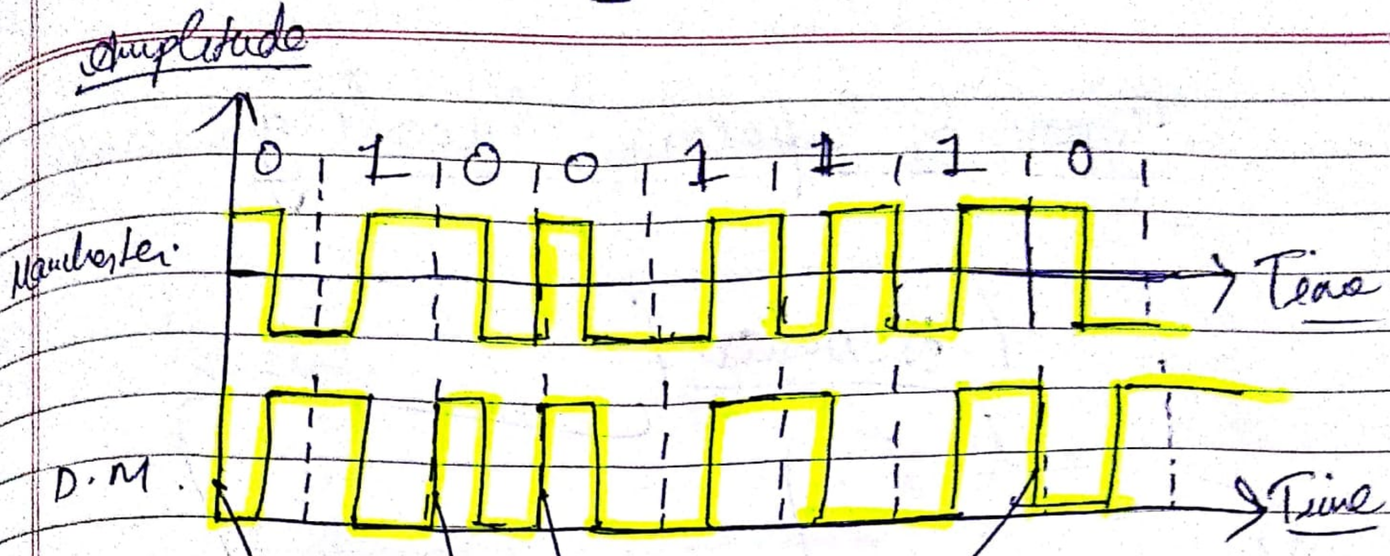
By using single transition for dual purpose, Manchester encoding achieves the same level of synchronization as RZ but with only two levels of amplitude

Differential Manchester:- the inversion at the middle of the bit interval is used for synchronization, but presence & absence of an additional transition at the beginning of the interval is used to identify the bit.

A transition means → binary 0
No transition means → binary 1.

Differential Manchester requires two signals changes to represent binary 0 but only one to represent binary 1.

Zero is ⌊ one & ⌈



Presence of transition at the begining of bit time means zero.

Bipolar encoding

In Bipolar encoding we use three levels of polarity — positive, negative & zero

Types of Bipolar Encoding

Bipolar

AMI | B8ZS | HDB3

AMI :- Bipolar Alternate marke inversion

The word mark comes from telegraphy & means 1.

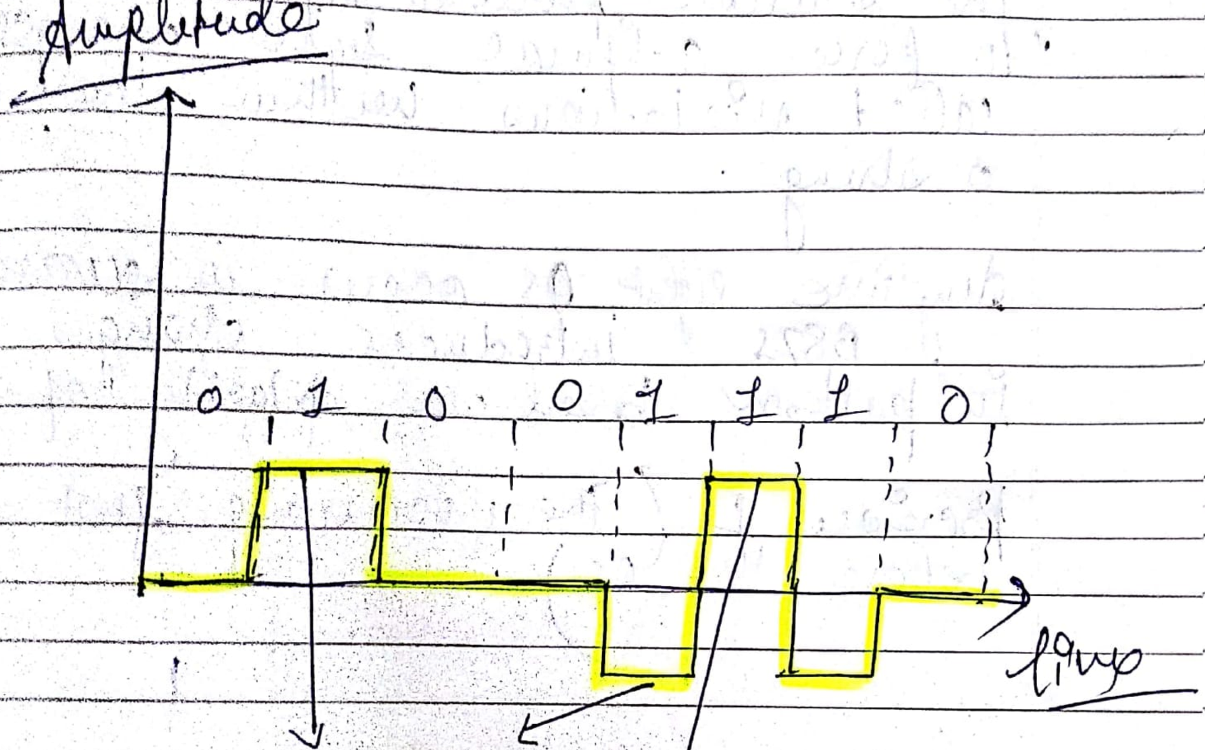
So AMI means alternate 1 inversion.

A neutral, zero voltage represents binary 0.

$\boxed{\text{0 voltage} \rightarrow \text{binary zero}}$

1s are represented by alternating positive & negative voltages.

binary 1 → alternate +ve & -ve voltage



The 1s are +ve & -ve alternatively

By inverting on each occurance of 1, bipolar AMI accomplishes one thing i.e a long sequence of 1s stays synchronized.

Polarity of previous bit

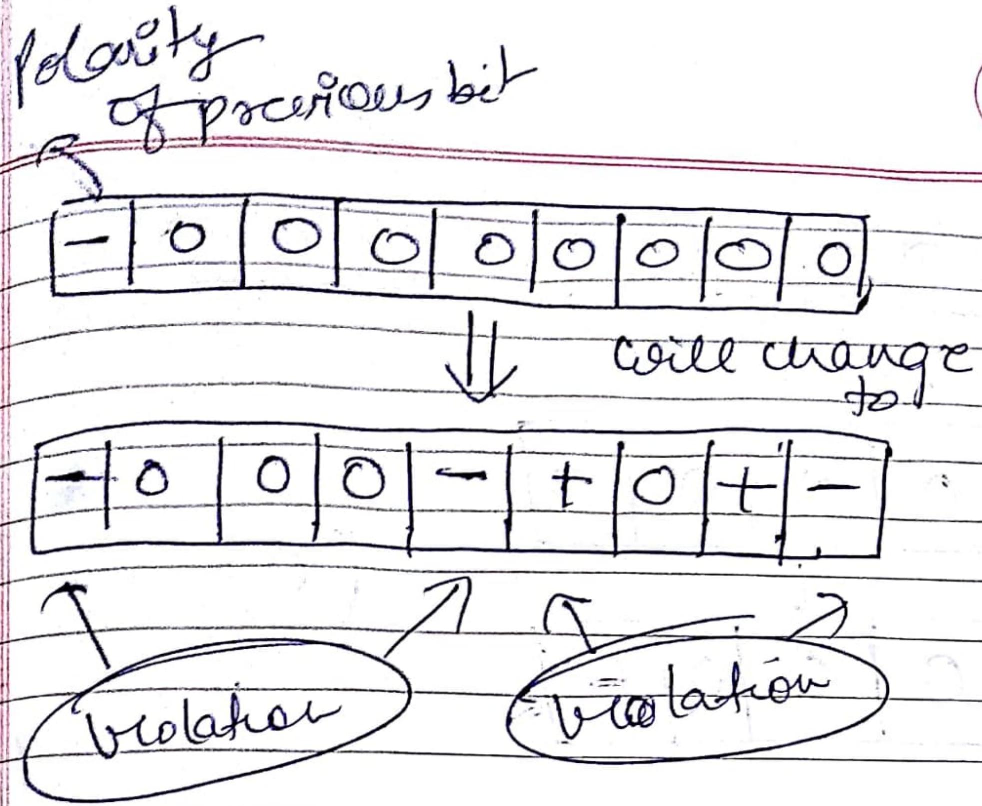
+	0	0	0	0	0	0	0	0

$\Downarrow$ will change to

+	0	0	0	+	−	0	−	+

violation

violation



In B8ZS, if eight 0s come one after another, we change the pattern in one of the two way based on the polarity of previous 1.

High Density Bipolar 3 (HDB3)

+	0	0	0	0

↓

+	0	0	0	+

(a)

−	0	0	0	0

↓

−	0	0	0	−

(b)

(a) & (b) are applied if the number of 1s since the last substitution is odd.

+	0	0	0	0

↓

+	-	0	0	-

(c)

-	0	0	0	0

↓

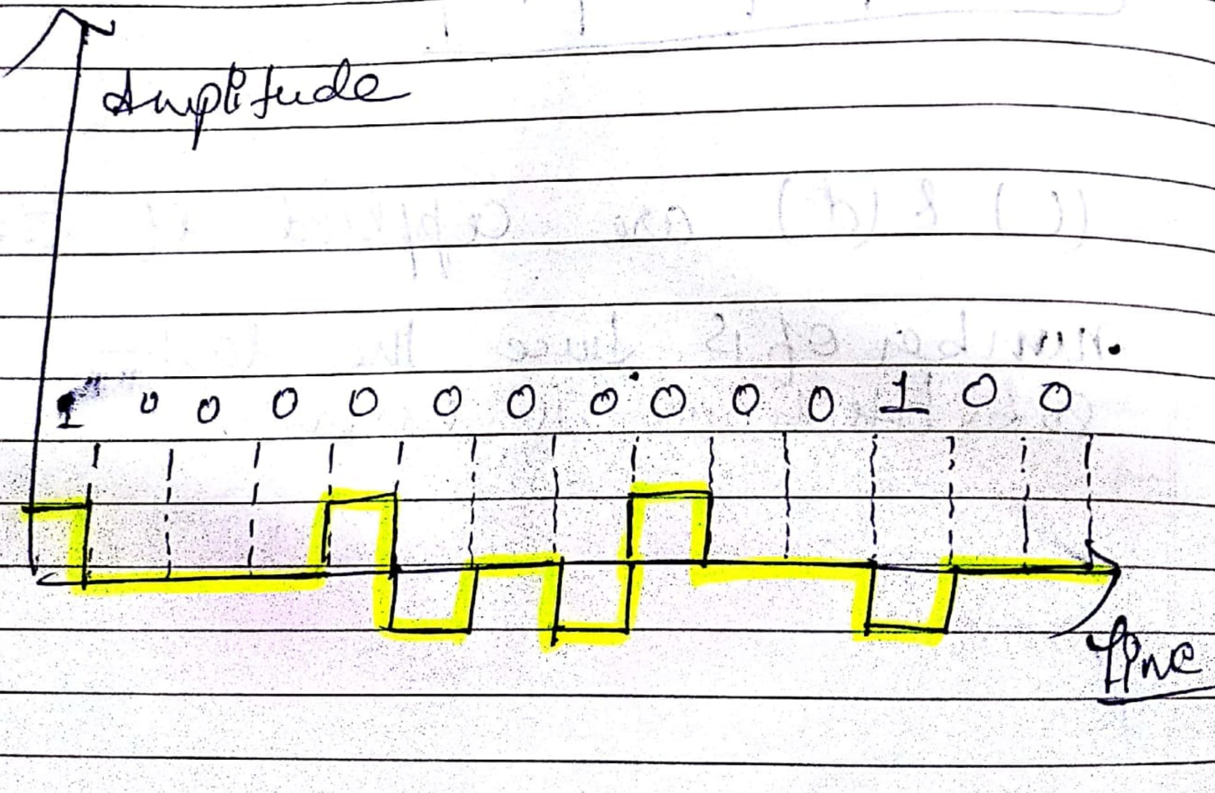
-	+	0	0	+

(d)

(c) & (d) are applied if the number of 1s since the last substitution is even.

In HDB3, if four 0s come one after the other, we change the pattern in one of four ways based on the polarity of previous 1 & the no. of 1s since the last substitution

Q Using B8ZS, encode the bit stream 10000000000100. Assume the polarity of 1st 1 is +ve.



Q. Using HDB3, encode the bit stream 10000000000100. Assume that the number of 1s so far is odd & 1st 1 is +ve.