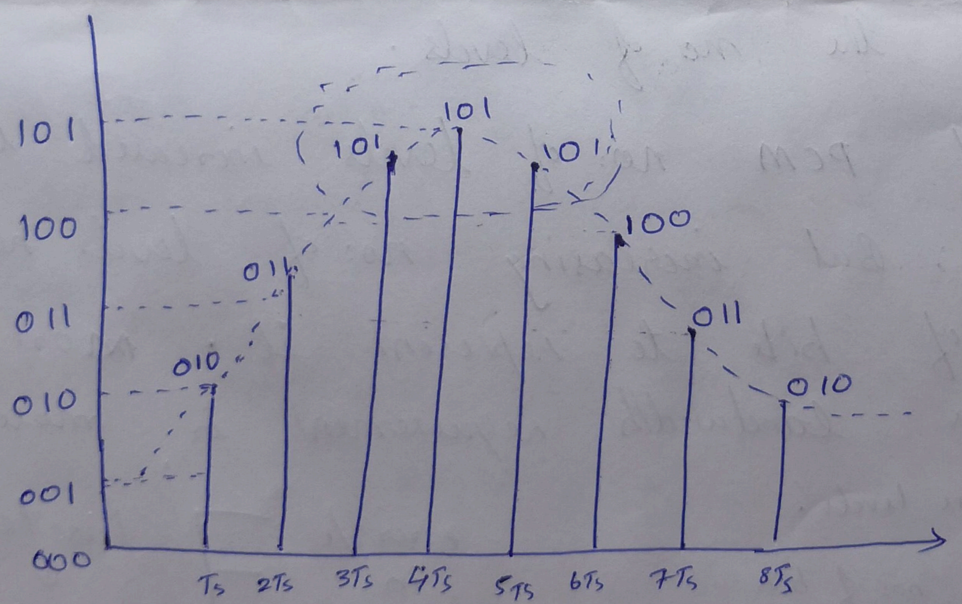
Differential pulse code modulation (DPCM)

Inorder to overcome the demerits of pulse code modulation we use differential pulse code modulation. we know that is pulse code modulation digital representation of an analog signal is obtained by taking the samples of the amplitude of the signal at regular intervals and sampled data is rounding off to a level by quantization and then represent these quantised data by binary digits.

By doing this there is a chance of transmission of redundant data and this will cause wastage of bandwidth for example;



3 successive samples are aproximated to same quantization level These samples can be carried by a single code. ie; same information is carried at 3 different ~~samp~~ sampling instant. This is nothing but redundant data

if the redundancy is reduced bit rate also can reduce.

If we look in the aspect of quantisation error ;

$$|Q_e|_{max} = \frac{\Delta}{2}$$

from the equation we can know that in order to reduce Quantization error we have to reduce the stepsize Δ; where $\Delta = \dfrac{V_{max} - V_{min}}{2^n}$ $\left[\underbrace{2^n = L}_{\text{no. of levels}} \right.$

So inorder to reduce stepsize Δ either we have to reduce the dynamic range $(V_{max} - V_{min})$ or we have to increase the no of levels.

In case of PCM no: of levels increased to reduce the error ; But increasing no: of levels needs more no: of bits to represent it. more no. of bits means bandwidth requirement is more.

PCM — more levels → more no: of bits per level

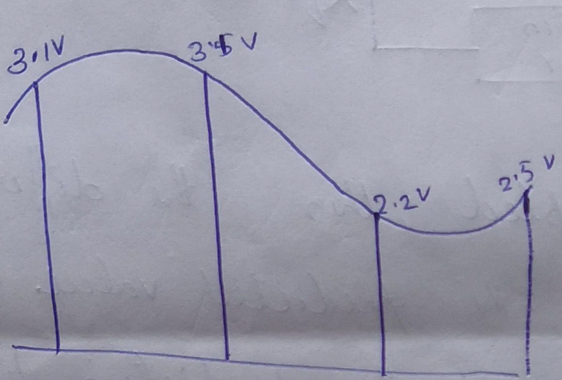
errors s/l in DPCM — less levels → less no: of bits per level.

In differential pulse code modulation quantisation error is decreased without effecting channel bandwidth by reducing the dynamic range.

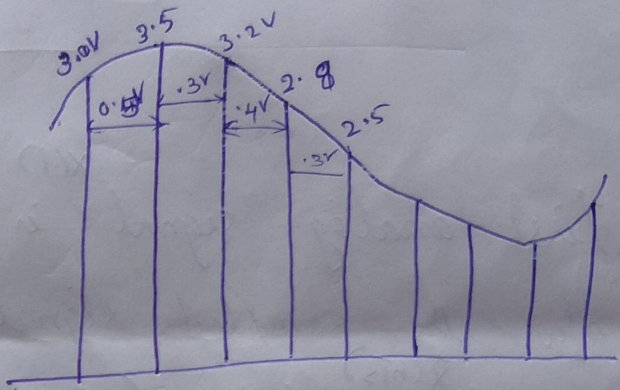
In DPCM to reduce the redundant information and to achieve more compression only the difference b/w the successive samples are transmitted. In DPCM the input analog signal is sampled at a rate higher than the Nyquist rate; so that the successive samples get more correlated.

$f_s = 5W$

$f_s = 2W$



PCM

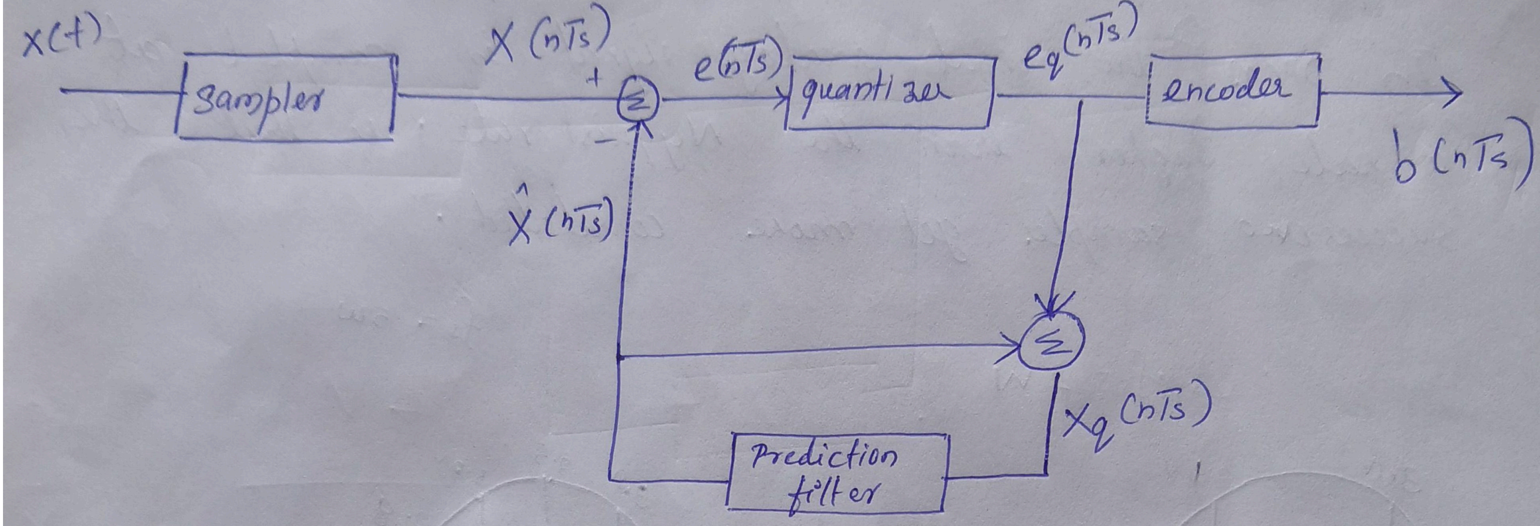


DPCM

In DPCM when we take the difference in amplitude BW two consecutive samples the dynamic range is reduced or we can say that the signal is compressed for this compressed s/c there is a less memory requirement for encoding.

DPCM is an analog to digital signal conversion technique which works on the principle of prediction. In this a prediction filter predicts nearby future values of the signal; by analysing past behaviour of the signal to sufficient time extend.

DPCM Transmitter



First analog signal $x(t)$ is sampled; then the difference b/w the sampled signal $x(nT_s)$ and the predicted value of the signal $\hat{x}(nT_s)$ is taken (Initially consider the prediction filter o/p as zero). This is done by a comparator. This results in an error signal $e(nT_s)$; which is known as prediction error.

$$e(nT_s) = x(nT_s) - \hat{x}(nT_s)$$

This prediction error is given as the inputs to the quantizer. Then we get a quantized error s/l ($e_q(nT_s)$) It is a combination of prediction error and quantized error signal.

$$e_q(nT_s) = e(nT_s) + q(nT_s)$$

This is given to the encoder which encodes the quantized s/l. At the o/p of the encoder we get the DPCM o/p → $b(nT_s)$; which is nothing but a binary data.

The quantized error s/l $e_q(nT_s)$ is given to an adder and the another input to this adder is $\hat{x}(nT_s)$. The o/p of this adder is then given to the prediction filter; which is represented by $x_q(nT_s)$

So the prediction filter i/p is expressed as

$$x_q(nT_s) = e_q(nT_s) + \hat{x}(nT_s)$$

$$= e(nT_s) + q(nT_s) + \hat{x}(nT_s)$$

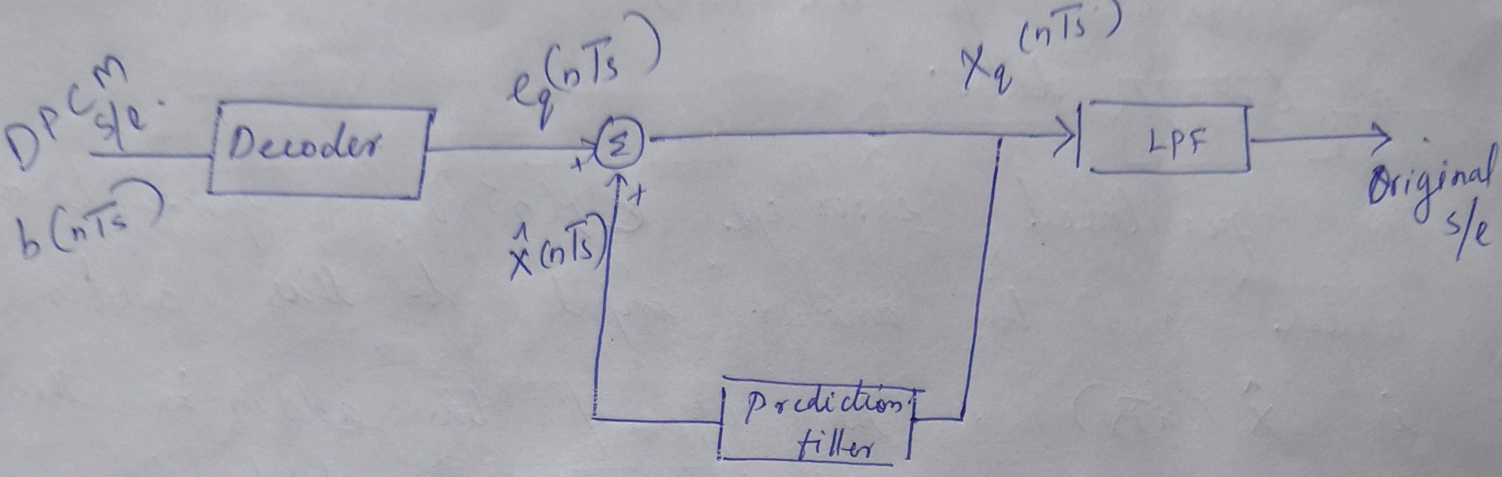
$$\because \left[e(nT_s) = x(nT_s) + \hat{x}(nT_s) \right]$$

$$= x(nT_s) - \cancel{\hat{x}(nT_s)} + q(nT_s) + \cancel{\hat{x}(nT_s)}$$

$$x_q(nT_s) = x(nT_s) + q(nT_s)$$

Hence the quantized o/p of the DPCM depends on the input sampled s/l and quantization error signal. But it doesn't depends upon any characteristics of the prediction filter.

DPCM Reciever.



$$X_q(nT_s) = e_q(nT_s) + \hat{x}(nT_s)$$

$X_q(nT_s)$ is a staircase o/p which we pass it through a LPF ; and at the o/p LPF it will be generating ~~the~~ original s/l.

From the fig: we can know that the decoder first reconstructs the quantized error signal from incoming binary signal the prediction filter output ($\hat{x}(nT_s)$) and quantized error signals are summed up to give the quantised version of the original s/l Thus the s/l at the reciever differs from actual s/l by quantization error $q(nT_s)$, which introduced permanently in the reconstructed s/l.