# DELTA MODULATION DEMODULATION

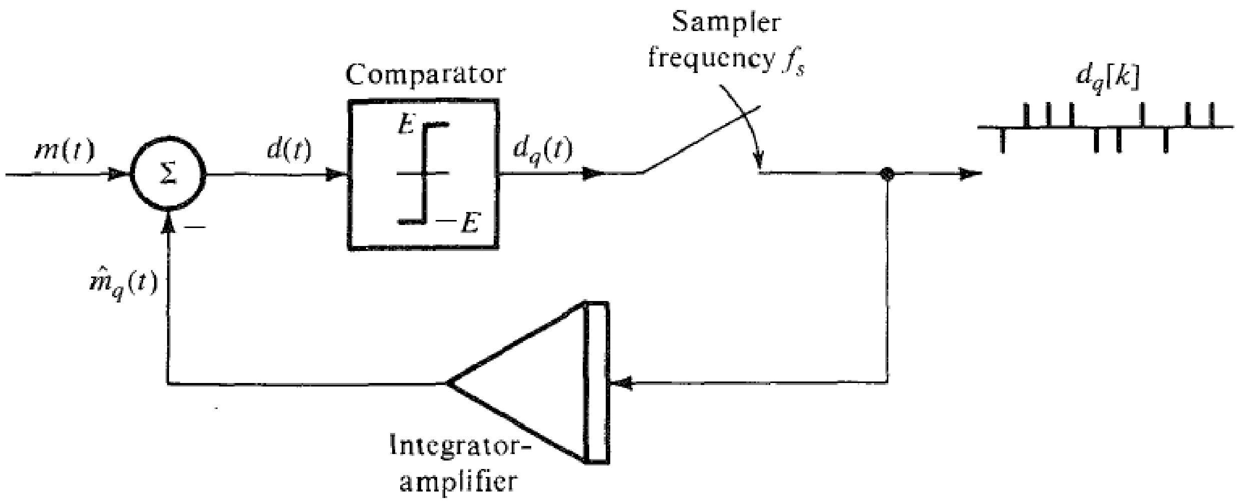
**Experiment No:** **Date:**

**Aim: Write a MATLAB code to modulate and demodulate the given signal by Delta Modulation Technique.**

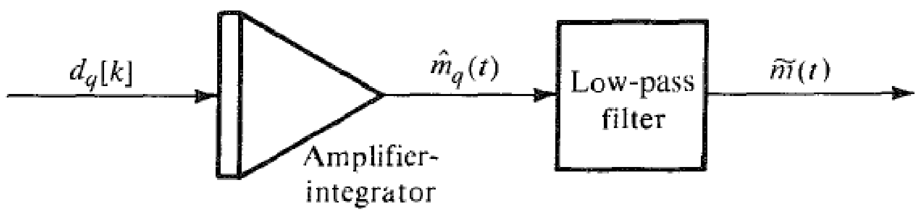
**Brief Theory/Equations:**

A delta modulation (DM or Δ-modulation) is an analog-to-digital and digital-to-analog signal conversion technique used for transmission of voice information where quality is not of primary importance. DM is the simplest form of differential pulse-code modulation (DPCM) where the difference between successive samples is encoded into n-bit data streams. In delta modulation, the transmitted data are reduced to a 1-bit data stream. Its main features are:

- The analog signal is approximated with a series of segments.
- Each segment of the approximated signal is compared to the preceding bits and the successive bits are determined by this comparison.
- Only the change of information is sent, that is, only an increase or decrease of the signal amplitude from the previous sample is sent whereas a no-change condition causes the modulated signal to remain at the same 0 or 1 state of the previous sample.
- To achieve high signal-to-noise ratio, delta modulation must use oversampling techniques, that is, the analog signal is sampled at a rate several times higher than the Nyquist rate.



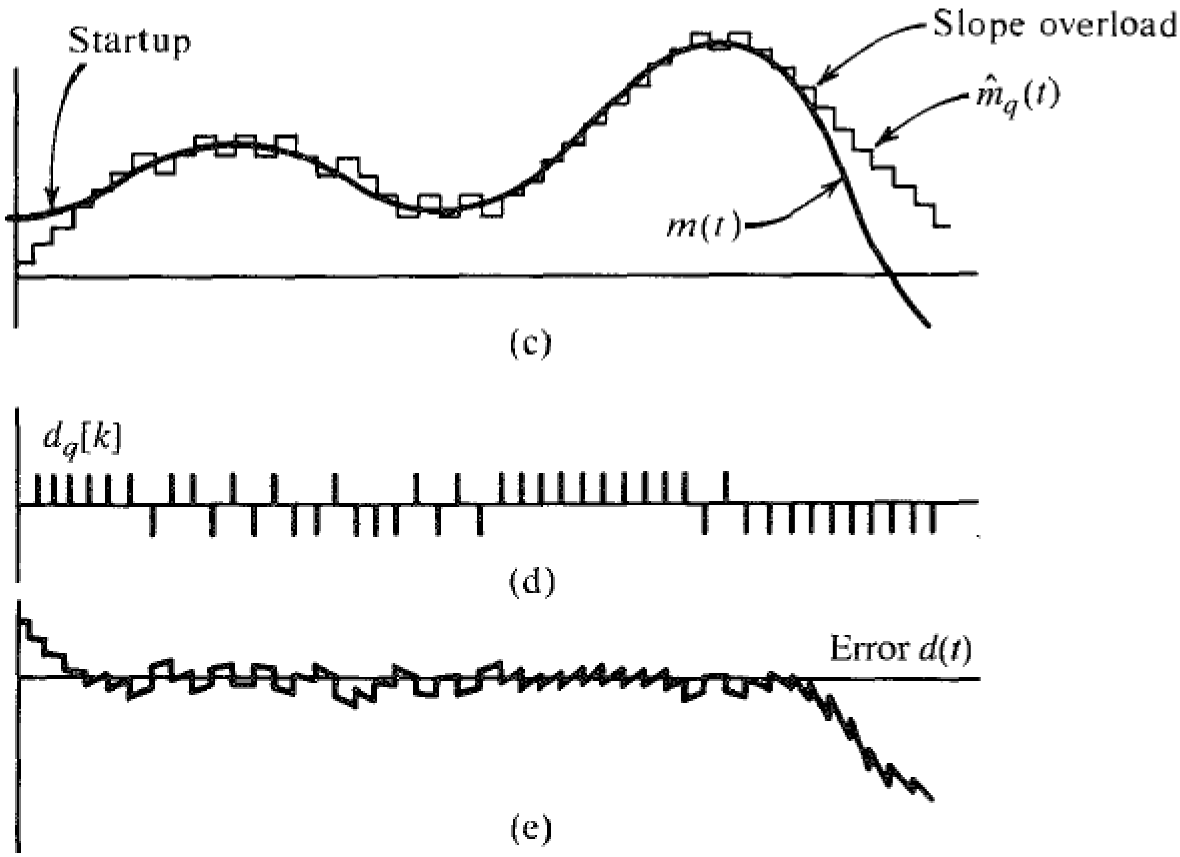

(a) Delta modulator




(b) Delta demodulator

The modulator consists of comparator and sampler in the direct path and an integrator amplifier in the feedback path. The analog signal $m(t)$ is compared with the feedback signal $\widehat{m}_q(t)$. The error signal $d(t) = m(t) - \widehat{m}_q(t)$ is applied to the comparator. If $d(t)$ is positive, the comparator output is of constant amplitude E and if $d(t)$ is negative, the comparator output is of constant value –E. Thus, the difference is the binary signal that is needed to generate 1 bit DPCM (Differential Pulse Code Modulation). The comparator output is sampled by a sampler at sampling frequency $f_s$, samples per second where $f_s$ is much higher than the nyquist sampling rate. The sampler thus produces train of narrow pulses $d_q(k)$ (to simulate impulses) with a positive pulse when $m(t) > \widehat{m}_q(t)$ and negative pulse when $m(t) < \widehat{m}_q(t)$. Each sample is coded by a single binary pulse. The pulse train $d_q(k)$ is delta modulated pulse train. The modulated signal $d_q(k)$ is amplified and integrated in the feedback path to generate $\widehat{m}_q(t)$, which tries to follow $m(t)$.

Each pulse in $d_q(k)$ at the input of the integrator gives rise to a step function (positive or negative depending on the polarity of pulse) in $\widehat{m}_q(t)$. If for example, $m(t) > \widehat{m}_q(t)$, a positive pulse is generated in $d_q(k)$, which gives rise to a positive step in $\widehat{m}_q(t)$ trying to equalize $\widehat{m}_q(t)$ to $m(t)$ in small steps at every sampling instant as shown in figure below. It can be seen that $\widehat{m}_q(t)$ is kind of staircase approximation of $m(t)$. When $\widehat{m}_q(t)$ is passed through a low pass filter, the coarseness of staircase in $\hat{m}_q(t)$ is eliminated and we get smoother and better approximation of $m(t)$. The demodulator at the receiver consists of an amplifier-integrator (identical to that of feedback path of modulator) followed by a low pass filter.




In DM, the modulated signal carries information not about the signal samples but about the difference between successive samples. If the difference is positive or negative, a positive or negative pulse is generated in the modulated signal $d_q(k)$. Basically, therefore, DM carries the information about the derivative of $m(t)$, hence the name delta modulation. This can also be seen from the fact that integration of delta modulated signal yields $\hat{m}_q(t)$, which is an approximation of $m(t)$. The information of difference between successive samples is transmitted by a 1 bit code word.

**Threshold of Coding and Overloading**

Threshold and overloading effects can be seen in the figure c. Variations in $m(t)$, smaller than the step value (threshold of coding) are lost in DM. Moreover, If $m(t)$ is too fast, derivative of it $\dot{m}(t)$ is too high, $\hat{m}_q(t)$ cannot follow $m(t)$ and overloading occurs. This is known as slope overloading, which gives rise to the slope overload noise. This noise is one of the basic limiting factors in the performance of DM. We should expect slope overload rather than amplitude overload in DM, because DM basically carries the information about $m(t)$. The granular nature of the output signal gives rise to the granular noise similar to the quantization noise. The slope overload noise can be reduced by increasing E (the step size). This unfortunately increases the granular noise. There is an optimum value of E, which yields the best compromise giving the minimum overall noise. This optimum value of E depends on the sampling frequency $f_s$ and the nature of the signal.

The slope overload occurs when $\widehat{m}_q(t)$ cannot follow $m(t)$. During the sampling interval $T_s$, $\widehat{m}_q(t)$ is capable of changing by $\sigma$, where $\sigma$ is the height of the step (amplitude). Hence, the maximum slope that $m(t)$ can follow is $\frac{\sigma}{T_s}$ , or $\sigma f_s$, where $f_s$, is the sampling frequency. Hence, no overload occurs if

$$|\dot{m}(t)|^{\cdot} \geq \sigma f_s$$

Consider the case of tone modulation (meaning a sinusoidal message):

$$m(t) = A \cos \omega t$$

The condition for no overload is

$$|\dot{m}(t)|_{max} = \omega A < \sigma \dot{f}_s$$

Hence, the maximum amplitude $A_{max}$ of this signal that can be tolerated without overload is given by

$$A_{max} = \frac{\sigma f_s}{\omega}$$

The overload amplitude of the modulating signal is inversely proportional to the frequency $\omega$. For higher modulating frequencies, the overload occurs for smaller amplitudes. For voice signals, which contain all frequency components up to (say) 4 kHz, calculating $A_{max}$ by using$\omega = 2\pi \times 4000$ in above equation will give an overly conservative value. It has been shown that $A_{max}$ for voice signals can be calculated by using $\omega_r \approx 2\pi \times 800$

$$[A_{max}]_{voice} \sim \frac{\sigma f_s}{\omega_r}$$

Thus, the maximum voice signal amplitude $A_{max}$ that can be used without causing slope overload in DM is the same as the maximum amplitude of a sinusoidal signal of reference frequency $f_r$ that can be used without causing slope overload in the same system

Fortunately, the voice spectrum (as well as the television video signal) also decays with frequency and closely follows the overload characteristics. For this reason, DM is well suited for voice (and television) signals. Actually, the voice signal spectrum (curve b) decreases as $\frac{1}{\omega}$ up to

2000 Hz, and beyond this frequency, it decreases as $\frac{1}{\omega^2}$ . If we had used a double integration in the feedback circuit instead of a single integration, $A_{max}$ would be proportional to $\frac{1}{\omega^2}$ . Hence, a better match between the voice spectrum and the overload characteristics is achieved by using a single integration up to 2000 Hz and a double integration beyond 2000 Hz. Such a circuit (the double integration) responds fast but has a tendency to instability, which can be reduced by using some low-order prediction along with double integration. A double integrator can be built by placing in cascade two low-pass RC integrators with time constants R1 C1 = l /200pi and R2C2 = 1/4000pi, respectively. This results in single integration from 100 to 2000 Hz and double integration beyond 2000 Hz.

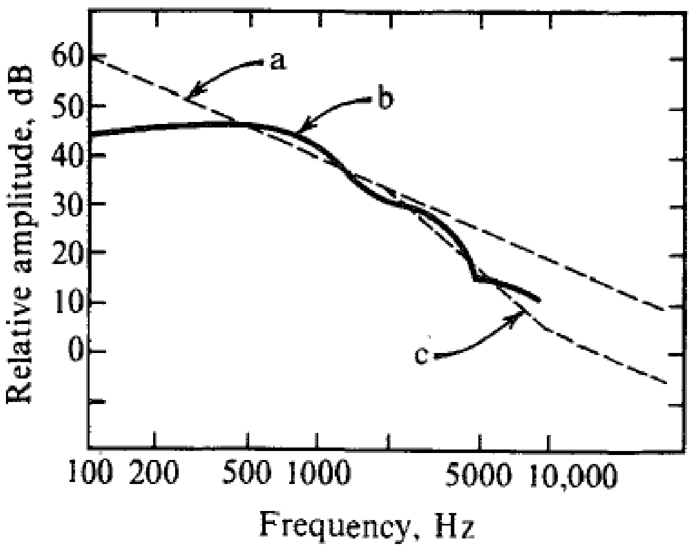



**Figure 6.21** Voice signal spectrum.

**Delta Modulation : A special case of DPCM**

Sample correlation used in DPCM is further exploited in ***delta modulation (DM)*** by oversampling (typically four times the Nyquist rate) the baseband signal. This increases the correlation between adjacent samples, which results in a small prediction error that can be encoded using only one bit (L = 2). Thus, DM is basically a 1-bit DPCM, that is, a DPCM that uses only two levels (L = 2) for quantization of $m[k] - m_q[k]$. In comparison to PCM (and DPCM), it is a very simple and inexpensive method of *A/D* conversion. A 1-bit codeword in DM makes word framing unnecessary at the transmitter and the receiver. This strategy allows us to use fewer bits per sample for encoding a baseband signal.

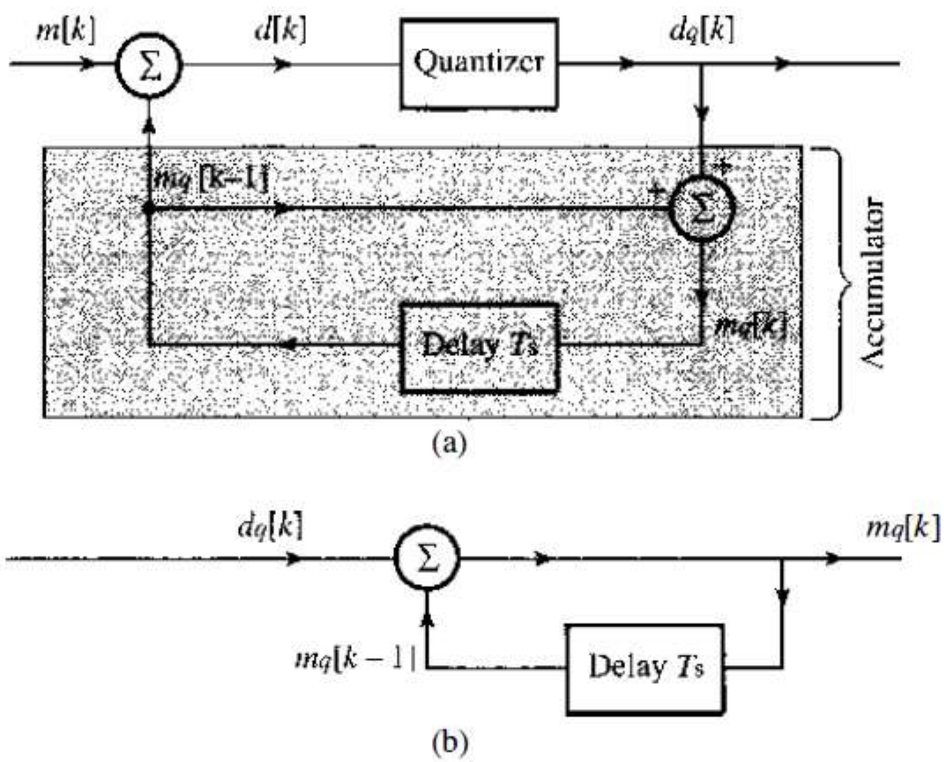
In DM, we use a first-order predictor, which, as seen earlier, is just a time delay of $T_s$, (the sampling interval). Thus, the DM transmitter (modulator) and receiver (demodulator) are identical to those of the DPCM in Fig. below, with a time delay for the predictor, as shown in Fig, from which we can write

$$m_q[k] = m_q[k-1] + d_q[k]$$

Hence,

$$m_q[k-1] = m_q[k-2] + d_q[k-1]$$

**Figure 6.30** Delta modulation is a special case of DPCM.




Or we can write,

$$m_q[k] = m_q[k-2] + d_q[k-1] + d_q[k]$$

Proceeding in this manner, assuming zero initial condition, i.e. $m_q[0]$=0, we write

$$m_q[k] = \sum_{m=0}^{k} d_q[m]$$

This shows that the receiver (demodulator) is just an accumulator (adder). If the output $d_q[k]$ is represented by impulses, then the accumulator (receiver) may be realized by an integrator because its output is the sum of the strengths of the input impulses ( sum of the areas underthe impulses). We may also replace with an integrator the feedback portion of the modulator (which is identical to the demodulator). The demodulator output is $m_q[k]$ , which when passed through a low-pass filter yields the desired signal reconstructed from the quantized samples.

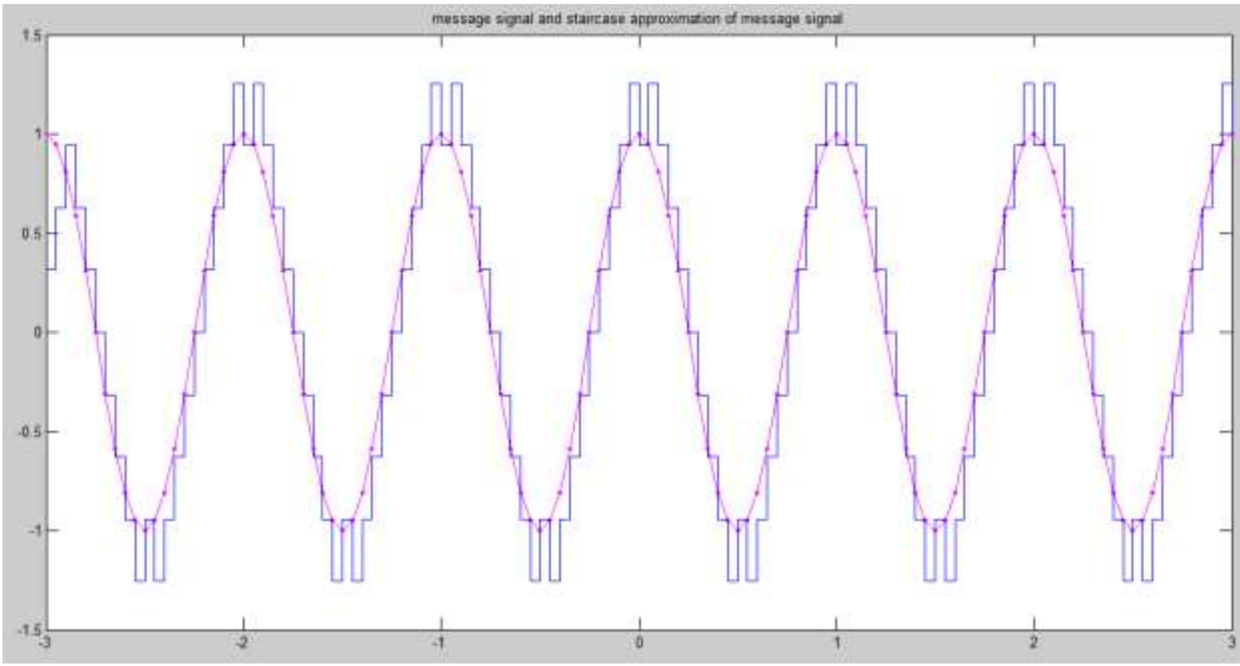
**Algorithm:**

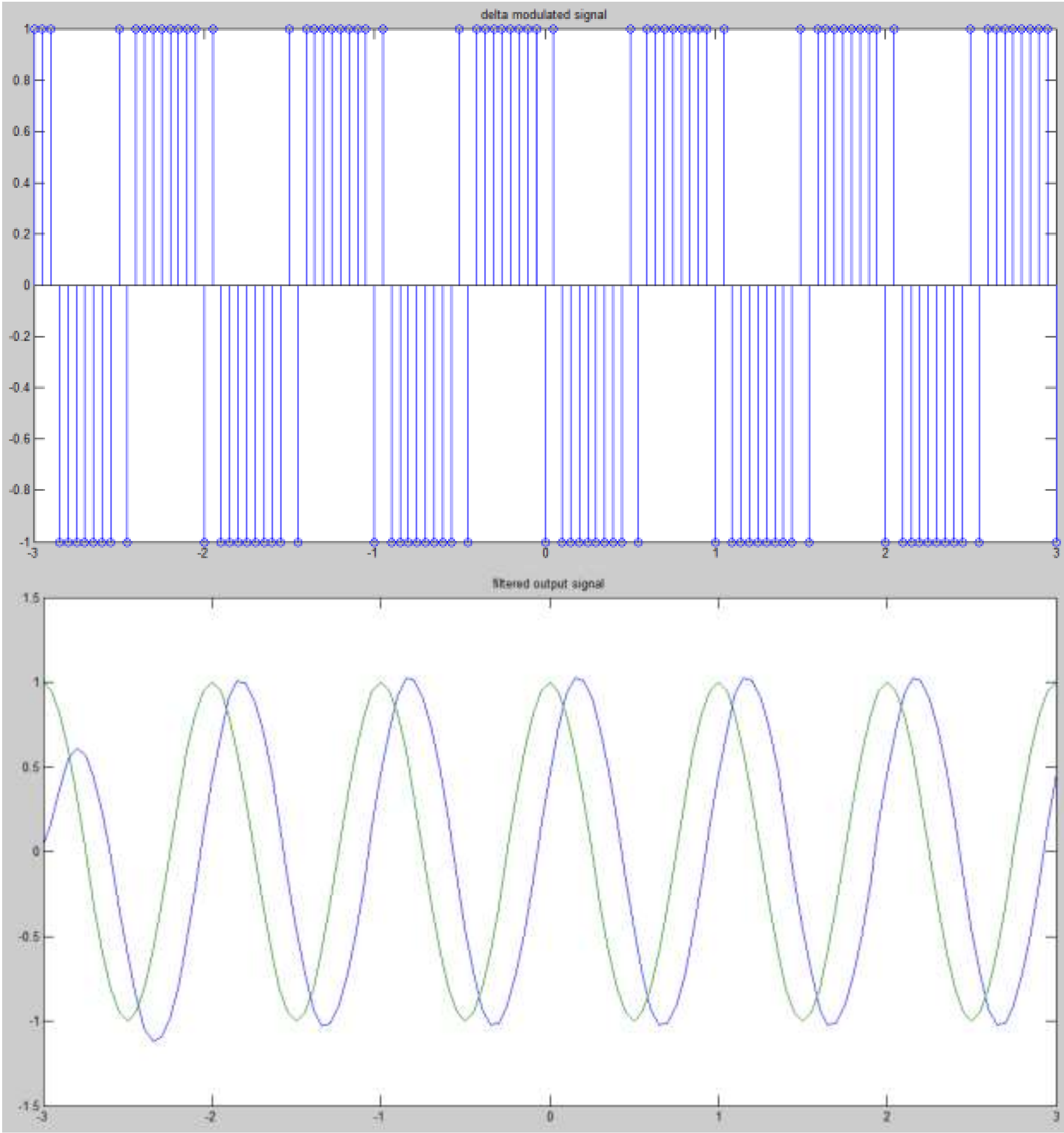
- Impement the block diagram of DM as a special case of DPCM (Figure 6.30, page 295, "Modern Analog and Digital Communication" by B.P. Lathi $4^{th}$ edition)
- Consider the input/message signal as sinusoidal `m=Am*cos(2*pi*fm*t)`, with parameters, Am=1V, fm=1Hz.
- Define the time range with sampling frequency fs=20*fm (oversampling), hence, t can be defined as `t=-3:1/fs:3;`
- Define the step size del for the delta modulator which should satisfy the condition

$$\Delta \leq \frac{2\pi A_m f_m}{f_s}$$

  Hence, `del=(2*pi*fm*Am)/fs;`
- Choose the index i=1:length(t) ; length(t)=max no. of columns in t.
  - If i=1, then mq=0.
    - So, the difference signal `d(i)=m(i);`
    - Use the sign function of MATLAB to determine whether d is +ve or –ve
    - Determine the approximate difference value dq by applying hard limiting operation i.e. by multiplying sign(d) with del.
    - Approximated message signal mq=dq, for i=1.
  - Else
    - The difference signal, `d(i)=m(i)-mq(i-1);`
    - The approximated difference operation will be same as in case for i=1.
    - Approximated message signal (staircase approximation), `mq(i)=dq(i)+mq(i-1);`
- Figure1: plot the message signal and staircase approximation signal the same window. Use the command hold on. And use command stairs(t,mq) for approximated message signal.
- Figure2: plot the delta modulated signal, consider the modulated output x to be +1 if dq>0 else x will be -1.
  - Use stem command for plotting as it is a discrete time signal
- For demodulation, pass the approximated message signal via low pass filter. (See FM demodulation algorithm for filtering logic).
- Figure 3: plot filtered output signal and original input signal in the same window.
  - `plot(t,2*y,t,m);`

## Expected Ouput Waveforms:






## **Code:**

**Output Waveforms:**

**Conclusion:**

**Remarks:** **Signature:**

## References:


- NPTEL digital communication systems lectures
  http://www.digimat.in/nptel/courses/video/108102096/L02.html
- Modern Analog and Digital Communication by B.P. Lathi ($3^{rd}$ or $4^{th}$ edition)
- Communication Systems by Simon Haykin ($4^{th}$ edition).
- Delta modulation/demodulation in MATLAB
  https://www.youtube.com/watch?v=XHHrh-vyhcE