An analog message m(t) = 3cos(100πt) + 4sin(50πt). Using Python Script(s)

1. Plot m(t) in both time and frequency domain.
2. Sample m(t) using a sampling frequency of 1000Hz, quantize it using round off technique with a quantization level of L=256 & convert it into a PCM signal by assigning necessary

bits to each quantization level. Plot the sampled (ms[n]) and quantized signal (mq[n]) in time and frequency domain. Also plot the PCM bit-stream (mb[n]) as a stair-plot in time

domain.

1. Modulate the PCM bit-stream using M-ary QAM modulation. Take M=16 and carrier frequency, fc = 10 kHz. Plot the QAM modulated signal in both time and frequency domain.
2. Add additive white Gaussian noise (AWGN) of -3dB with the QAM modulated signal. Plot the noisy QAM signal in both time and frequency domain. Also, plot the constellation diagram for this noisy QAM signal.
3. Now, design a QAM detector to de-modulate the above noisy QAM signal and recover the PCM bit-stream. Plot the demodulated bit-stream m^b[n] as a stair-plot and compare it with the actual PCM bit-stream mb[n] found in (2). Also, report the probability of bit error (i.e. mean(|mb[n] - m^b[n]|)).
4. From the recovered PCM bitstream, recover the quantized form of the analog message (m^q[n]). Compare it with the actual quantized message (mq[n]) and plot the error signal in time domain. Also, plot the recovered quantized signal, m^q[n] in time and frequency domain.