PART I: PCM QUANTIZATION

Consider the system shown in Fig. 1

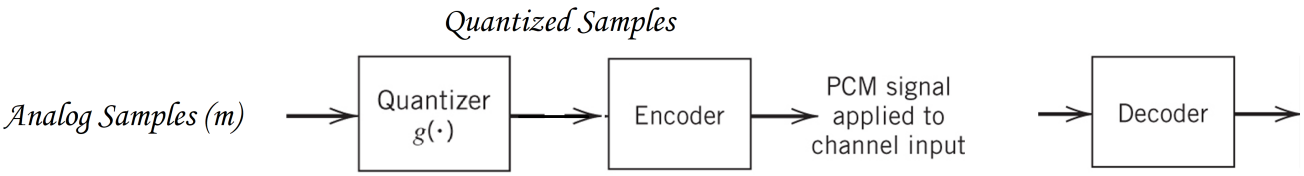


Fig. 1: PCM System with Discrete Input

You are required to **write software programs** and **implement a GUI** to help users **quantize**[1] an analog discrete signal.

GUI Description

Your 'Quantizer' function and GUI should have the option that **the user chooses between:**

1) **Uniform quantizer**, where the user specifies the number of levels, L, the peak quantization level, m_p, and whether the quantizer in mid-rise or mid-tread
2) **Non-Uniform μ-Law quantizer**, where the user specifies μ, L and m_p

The GUI should **allow the user to input a signal** to be quantized. That signal will be in the form of two vectors, a time vector and an amplitude vector.

The GUI should also display the following:

1) A figure showing the input signal and the quantized signal, on the same plot, with proper legend. **Note:** Display the input signal as a continuous signal, and display the quantized signal as a continuous staircase signal.
2) The value of the mean square quantization error, i.e. $\mathcal{E}\{(m-\nu)^2\}$

[1] You will implement the Quantizer only

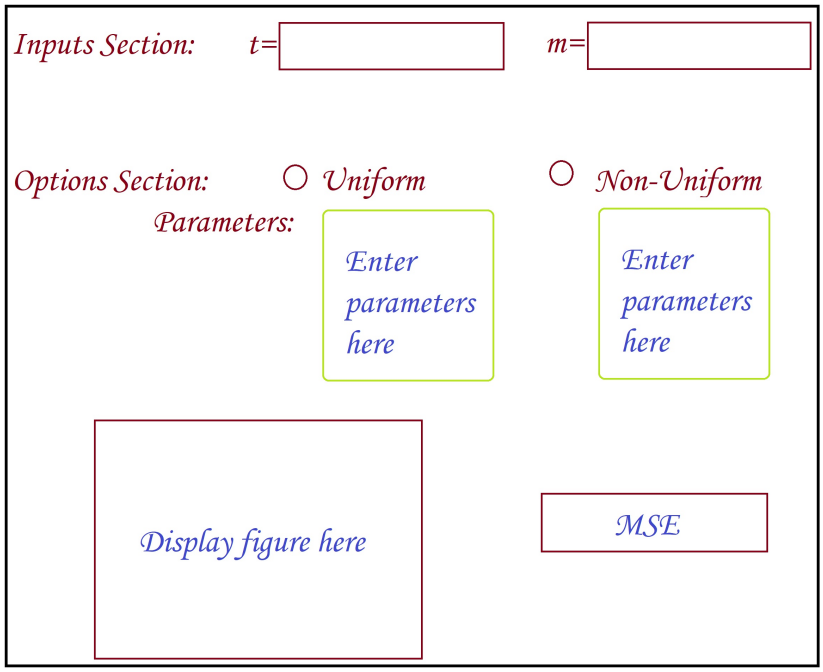


Fig. 2: Sample GUI

Testing your Simulator and GUI

Test your quantizer GUI for the input signal $m[k]$ for the following cases

$$m[k] = 5\cos\left(2\pi f_m k\right), \qquad \text{where } f_m = 10\ Hz, \quad \text{for one complete cycle of the signal}$$

	Case 1	Case 2	Case 3	Case 4
Sampling Frequency	$f_s = 40\ Hz$	$f_s = 20\ Hz$	$f_s = 15\ Hz$	$f_s = 20\ Hz$
Quantizer	$\mu = 0, L = 8, m_p = 5$	$\mu = 0, L = 32, m_p = 5$	$\mu = 0, L = 16, m_p = 5$	$\mu = 100, L = 32, m_p = 5$

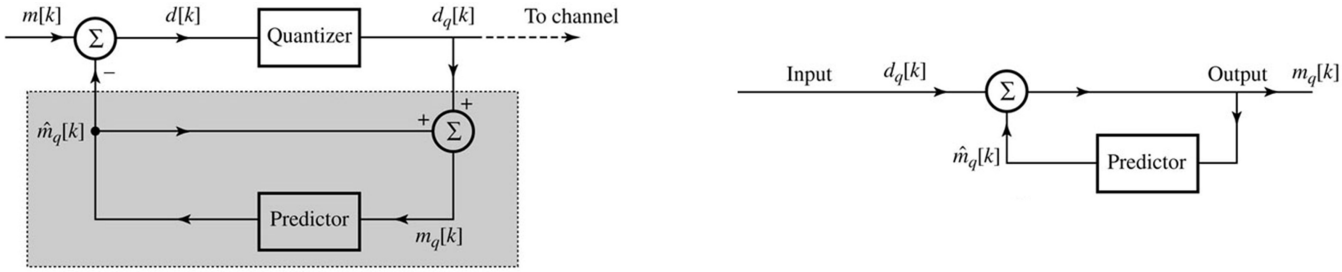
Deliverable - Part I

Deliver the following in a .zip file

1) The GUI files.
   This will be used to test your system with arbitrary parameters and for arbitrary input signals
2) Source codes (.m files) of functions and main files.
3) Source code of main script used for the 4 test cases.
4) Screenshots of the GUI's output for the 4 test cases.
5) For each of the 4 cases, make a brief comment on your findings
6) A single .pdf project report with a cover page.

PART II: DPCM ENCODER AND DECODER

Consider the system shown in Fig. 3



DPCM Encoder *DPCM Decoder*

Fig. 3: DPCM Encoder and Decoder

You are required to **write software programs** and **implement a GUI** to **encode** and **decode** an analog discrete signal using DPCM.

GUI Description

Your 'DPCM' encoder and decoder functions as well as the GUI should have the option that **the user specifies** the following:

1) The step size, Δ and whether the quantizer in mid-rise or mid-tread.
   Note: Only uniform quantizers are used. You may re-use files you created in Part I.
2) The order of the **predictor**:
   - First-order predictor
   - Second-order averaging predictor

The GUI should **allow the user to input a signal** to be encoded. That signal will be in the form of two vectors, a time vector and an amplitude vector.

The GUI should also display the following:

1) A figure with two subplots:
   - **Subplot 1** shows the input signal to the encoder and the output signal of the decoder, on the same plot, with proper legend.
     Note: Display the input signal as a continuous signal, and display the output signal as a continuous staircase signal.
   - **Subplot 2** shows the difference signal $d[k]$ and the quantized signal $d_q[k]$, on the same plot, with proper legend.
     These can be shown as discrete functions (using the **stem** command).
2) The value of the mean square quantization error, i.e. $\mathcal{E}\{(m - m_q)^2\}$

Testing your Simulator and GUI

Test your DPCM GUI for the input signal $m[k]$ as in Part I for Cases $1, 2$ and 3.

Deliverable - Part II

Deliver the following in a .zip file

1) The GUI files.
   This will be used to test your system with arbitrary parameters and for arbitrary input signals
2) Source codes (.m files) of functions and main files.
3) Source code of main script used for the 3 test cases.
4) Screenshots of the GUI's output for the 3 test cases.
5) For each of the 3 cases, make a brief comment on your findings
6) A single .pdf project report with a cover page.

GENERAL INSTRUCTIONS

- You can work this reports in teams of $3 \sim 5$ members per team.
- Write a full report including all the deliverable items.
- Late submissions are not allowed.
- **All team members should expect to be asked about all the report parts.**
- Teams are not expected to have used the same exact FECs, nor similar GUIs.
- Duplicate reports will be penalized by zero grade.
- Grading of the project will depend on:
  - **70%:** Completeness and correctness of the deliverable items (as per the .pdf report)
  - **20%:** Clarity of figures, and proper labeling (as per the .pdf report)
  - **20%:** Report writing and organization.