

UNIVERSITY OF BAHRAIN

COLLEGE OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

EENG 373: DIGITAL COMMUNICATIONS

EXPERIMENT NO. 1

QUANTIZATION

Objective

The main objectives of this experiment are:

- PART I. To perform scalar quantization using MATLAB m-file.
- PART II. To learn how to perform quantization using MATLAB SIMULINK model.

Laboratory Procedure

Part I. Quantizing a signal using MATLAB commands

QUANTIZATION or SIGNAL FORMATTING, is a way of processing data to reduce redundancy or prepare it for later processing. Analog-to-digital conversion and data compression are two categories of source coding.

Source coding divides into two basic procedures: SOURCE ENCODING and SOURCE DECODING. Source encoding converts a source signal into a digital signal using a quantization method. The symbols in the resulting signal are nonnegative integers in some finite range. Source decoding recovers the original information from the source-coded signal.

Representing Quantization Parameters

Scalar quantization is a process that maps all inputs within a specified range to a common value. It maps inputs in a different range of values to a different common value. In effect, scalar quantization digitizes an analog signal. Two parameters determine a quantization: a partition and a codebook. This section describes how blocks represent these parameters.

Partitions

A quantization partition defines several contiguous, nonoverlapping ranges of values within the set of real numbers. To specify a partition as a parameter, list the distinct endpoints of the different ranges in a vector.

For example, if the partition separates the real number line into the sets

$\{x: x \leq 0\}$

$\{x: 0 < x \leq 1\}$

$\{x: 1 < x \leq 3\}$

$\{x: 3 < x\}$

then you can represent the partition as the three-element vector

$[0,1,3]$

The length of the partition vector is one less than the number of partition intervals.

Codebooks

A codebook tells the quantizer which common value to assign to inputs that fall into each range of the partition. Represent a codebook as a vector whose length is the same as the number of partition intervals. For example, the vector $[-1,0.5,2,3]$

is one possible codebook for the partition $[0,1,3]$.

Assignment I

Q 1. Scalar Quantization

The code below shows how the quantiz function uses partition and codebook to map a real vector, samp, to a new vector, quantized, whose entries are either -1, 0.5, 2, or 3.

```
partition = [0,1,3];
codebook = [-1, 0.5, 2, 3];
samp = [-2.4, -1, -.2, 0, .2, 1, 1.2, 1.9, 2, 2.9, 3, 3.5, 5];
[index,quantized] = quantiz(samp,partition,codebook);
plot(samp,'*-')
hold on
plot(quantized,'r.-')
legend('Original signal','Quantized signal')
```

Result: (write your output and comment)

Q 2. Try changing the partition and codebook in order to get better quantized signal.

Result: (write your output and comment)

Q3. This example illustrates the nature of scalar quantization more clearly. After quantizing a sampled sine wave, it plots the original and quantized signals. The plot contrasts the \mathbf{x} 's that make up the sine curve with the dots that make up the quantized signal. The vertical coordinate of each dot is a value in the vector codebook.

```
t = [0:.1:2*pi]; % Times at which to sample the sine function
sig = sin(t); % Original signal, a sine wave
partition = [-1:.2:1]; % Length 11, to represent 12 intervals
codebook = [-1.2:.2:1]; % Length 12, one entry for each interval
[index,quants] = quantiz(sig,partition,codebook); % Quantize.
plot(t,sig,'x',t,quants,')
legend('Original signal','Quantized signal');
axis([-2 7 -1.2 1.2])
```

Result: (write your output and comment)

Q 4. Try changing the partition and codebook in order to get better quantized sine wave.

Result: (write your output and comment)

Assignment II

Q1. Quantize a cosine curve with peak to peak value of 6 V with quantization levels separation =0.3 V.

Q2. Plot the original signal along with the quantized signal on the same graph for 2 cycles of the original signal.

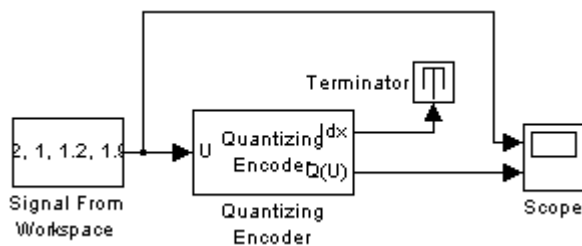
Q3. Show (in two different plots) the effect of changing the codebook to include peak values (maximum and minimum) of the original signal into the quantized signal.

PART II Quantizing a Signal USING SIMULINK MODEL

Assignment III

This section shows how the [Quantizing Encoder](#) and [Quantizing Decoder](#) blocks use the partition and codebook parameters. The examples here are analogous to [Scalar Quantization Example 1](#) and [Scalar Quantization Example 2](#) in the Communications Toolbox documentation.

Q 1. Scalar Quantization The figure below shows how the [Quantizing Encoder](#) block uses the partition and codebook as defined above to map a real vector to a new vector whose entries are either -1, 0.5, 2, or 3. In the Scope window, the bottom signal is the quantization of the (original) top signal.

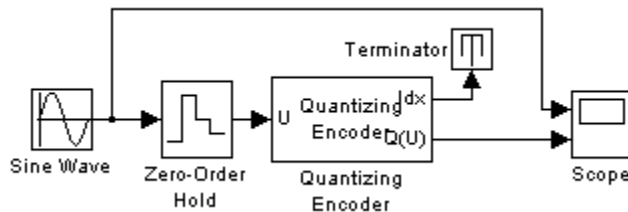


- [Signal From Workspace](#), in the Signal Processing Sources library
Set **Signal** to [-2.4,-1,-.2,0,.2,1,1.2,1.9,2,2.9,3,3.5]'
- [Quantizing Encoder](#)
Set **Quantization partition** to [0, 1, 3].
Set **Quantization codebook** to [-1, 0.5, 2, 3].
- [Terminator](#), in the Simulink Sinks library
- [Scope](#), in the Simulink Sinks library
After double-clicking the block to open it, click the **Parameters** icon and set **Number of axes** to 2.

Connect the blocks as shown in the figure. From the model window's **Simulation** menu, select **Configuration parameters**. In the Configuration Parameters dialog box, set **Stop time** to 12. Run the model to get the wave form in scope.

Result and comments :

Q2. This example, shown in the figure below, illustrates the nature of scalar quantization more clearly. It samples and quantizes a sine wave and then plots the original (top) and quantized (bottom) signals. The plot contrasts the smooth sine curve with the polygonal curve of the quantized signal. The vertical coordinate of each flat part of the polygonal curve is a value in the Quantization codebook vector.



To open the completed model, click [here](#) in the MATLAB Help browser. To build the model, gather and configure these blocks:

[Sine Wave](#), in the Simulink Sources library (NOT the Sine Wave block in the Signal Processing Sources library)

[Zero-Order Hold](#), in the Simulink Discrete library

Set **Sample time** to 0.1.

[Quantizing Encoder](#)

Set **Quantization partition** to [-1.:2:1].

Set **Quantization codebook** to [-1.1.:2:1.1].

[Terminator](#), in the Simulink Sinks library

[Scope](#), in the Simulink Sinks library

After double-clicking the block to open it, click the **Parameters** icon and set **Number of axes** to 2. Connect the blocks as shown in the figure. From the model window's **Simulation** menu, select **Configuration parameters**. In the Configuration Parameters dialog box, set **Stop time** to 2π . Run the model to get the output

Answer: (write your results and comments)

Assignment IV

1. Repeat assignment II using SIMULINK

Discussion and Conclusion