**Faculty of Engineering – Cairo University Computer Engineering Department**

 

**ELC 325B – Spring 2025**

**Digital Communications**

**Assignment #1**

**Quantization**

**Submitted to**

 Dr. Mai Badaway

Dr. Hala Abdelkader

Eng. Mohammed Khaled

**Submitted by**

|  |  |  |
| --- | --- | --- |
| **Name** | **Sec** | **BN** |
| **Ahmed Hamdy Mohammed** | **1** | **4** |
| **Abdelrahman Mostafa Gomaa** | **1** | **31** |

# Contents

[Requirement – 1 Explanation 3](#_Toc193097523)

[Requirement – 2 Explanation 3](#_Toc193097524)

[Requirement – 3 Explanation 4](#_Toc193097525)

[Requirement – 4 Explanation 5](#_Toc193097526)

[Requirement – 5 Explanation 6](#_Toc193097527)

[Requirement – 6 Explanation 7](#_Toc193097528)

[Index 8](#_Toc193097529)

[Requirement – 1 Code 8](#_Toc193097530)

[Requirement – 2 Code 8](#_Toc193097531)

[Requirement – 3 Code 9](#_Toc193097532)

[Requirement – 4 Code 10](#_Toc193097533)

[Requirement – 5 Code 11](#_Toc193097534)

[Requirement – 6 Code 12](#_Toc193097535)

# Figures

[Figure 1: Uniform Quantizer Function 3](#_Toc193097261)

[Figure 2: Uniform De-Quantizer Function 3](#_Toc193097262)

[Figure 3: Midrise Plot 4](#_Toc193097263)

[Figure 4: Midtread Plot 4](#_Toc193097264)

[Figure 5: Response to a random i.i.d signal 5](#_Toc193097265)

[Figure 6: Response to a non-uniform random input signal 6](#_Toc193097266)

[Figure 7: Responses of different μ-values 7](#_Toc193097267)

[Figure 8: Compressor, and expansion function 7](#_Toc193097268)

[Figure 9: Requirement - 1 Flow 8](#_Toc193097269)

[Figure 10: Requirement - 2 Flow 8](#_Toc193097270)

[Figure 11: Requirement - 3 Flow 9](#_Toc193097271)

[Figure 12: Requirement - 4 Flow 10](#_Toc193097272)

[Figure 13: Requirement - 5 Flow 11](#_Toc193097273)

[Figure 14: Requirement - 6 Flow 12](#_Toc193097274)

# Requirement – 1 Explanation

A computer code with colorful text

Description automatically generated

Figure : Uniform Quantizer Function

* The **UniformQuantizer** function performs uniform quantization on input samples as follows:

1. It calculates the quantization interval width using the formula:
2. It uses the formula: to quantize the input samples correctly

# Requirement – 2 Explanation

A computer screen shot of a number

Description automatically generated with medium confidence

Figure : Uniform De-Quantizer Function

* The **UniformDequantizer** function gets dequantized values from quantized ones as follows:

1. It calculates the dequantization level using the formula:
2. It reconstructs the values using the formula:

# Requirement – 3 Explanation

A graph with blue and orange lines

Description automatically generated

Figure : Midrise Plot

A graph with blue and orange lines

Description automatically generated

Figure : Midtread Plot

* As we see that the maximum quantization error in both cases for the midrise, and the midtread is
* In midtread quantization, the input value at zero corresponds to 0, whereas in midrise quantization, it rises instantly from to

# Requirement – 4 Explanation

A graph with a line

Description automatically generated

Figure : Response to a random i.i.d signal

* A uniform quantizer was applied to a uniform random variable that we had created. We discovered that the uniform quantizer's output Signal-to-Noise Ratio (SNR) matched the theoretical SNR exactly. This shows that the quantizer is accurate because its performance roughly matches the predicted theoretical values

# Requirement – 5 Explanation

A graph with a line

Description automatically generated

Figure : Response to a non-uniform random input signal

* A non-uniform random variable was created in this section and sent into the uniform quantizer and de-quantizer. After examination, we found that the quantizer's output signal's SNR values were below the theoretical values, particularly at lower ***n\_bit*** values. This was not the case in the preceding part, where SNR values nearly matched the theoretical values due to the use of a uniform random variable.

# Requirement – 6 Explanation

A graph with colored lines and numbers

Description automatically generated

Figure : Responses of different μ-values

A computer screen shot of text

Description automatically generated

Figure : Compressor, and expansion function

* To handle the previous problem, we used nonuniform μ quantizer. We started with a μ compressor, then used a uniform quantizer, de-quantizer, and expander. We compared the Signal-to-Noise Ratio (SNR) to theoretical values for various µ values.

# Index

This is our implementation ^\_^:

## Requirement – 1 Code

A computer code with colorful text

Description automatically generated

Figure : Requirement - 1 Flow

## Requirement – 2 Code

A computer screen shot of a number

Description automatically generated with medium confidence

Figure : Requirement - 2 Flow

## Requirement – 3 Code

A screenshot of a computer program

Description automatically generated

Figure : Requirement - 3 Flow

## Requirement – 4 Code

A screen shot of a computer program

Description automatically generated

Figure : Requirement - 4 Flow

## Requirement – 5 Code

A screen shot of a computer program

Description automatically generated

Figure : Requirement - 5 Flow

## Requirement – 6 Code

A screen shot of a computer code

Description automatically generated

Figure : Requirement - 6 Flow