**Department of Computer Engineering**



**Cairo University**

**Faculty of Engineering**

**ELC 325B – Spring 2023**

**Digital Communications**

**Assignment #1**

**Quantization**

**Submitted to**

Dr.

Dr.

Eng.

**Submitted by**

|  |  |  |
| --- | --- | --- |
| **Name** | **Sec** | **BN** |
| **Abdelaziz Salah Mohammed Abdou** | **1** | **37** |
| **Adham Ali Abdelaal** | **1** | **12** |

Contents

[**Part 1:…………** 4](#_Toc129905380)

[**Comment:** 4](#_Toc129905381)

[**Part 2:…………** 5](#_Toc129905382)

[**Comment:** 5](#_Toc129905383)

[**Part 3:…………** 6](#_Toc129905384)

[**Comment:** 6](#_Toc129905385)

[**Part 4:…………** 7](#_Toc129905386)

[**Comment:** 7](#_Toc129905387)

[**Part 5:…………** 8](#_Toc129905388)

[**Comment:** 8](#_Toc129905389)

[**Part 6:…………** 9](#_Toc129905390)

[**Comment:** 9](#_Toc129905391)

[**Index:** 10](#_Toc129905392)

Figures

[Figure 1 Fig Mid-Raise Staircase 4](#_Toc129905445)

[Figure 2 Fig Mid-Tread Staircase 5](#_Toc129905446)

[Figure 3 Fig 6](#_Toc129905447)

[Figure 4 Fig 7](#_Toc129905448)

[Figure 5 Fig SNR of uniform quantizer with non-uniform input signals 8](#_Toc129905449)

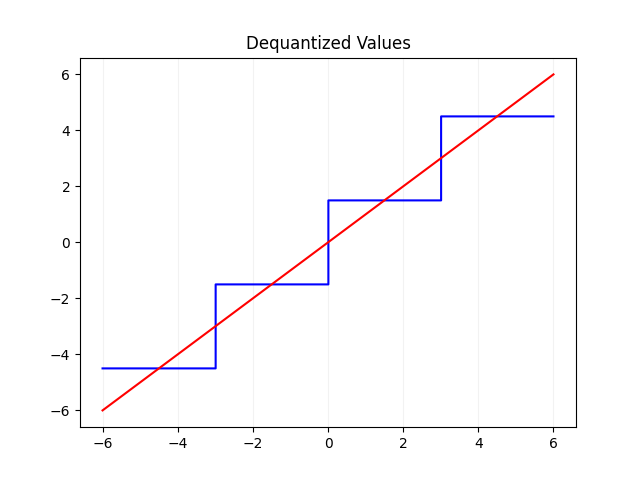
[Figure 6 Fig SNR of non-uniform quantizer with different u values 9](#_Toc129905450)

# **Part 1: Implementing a uniform scalar quantization function.**

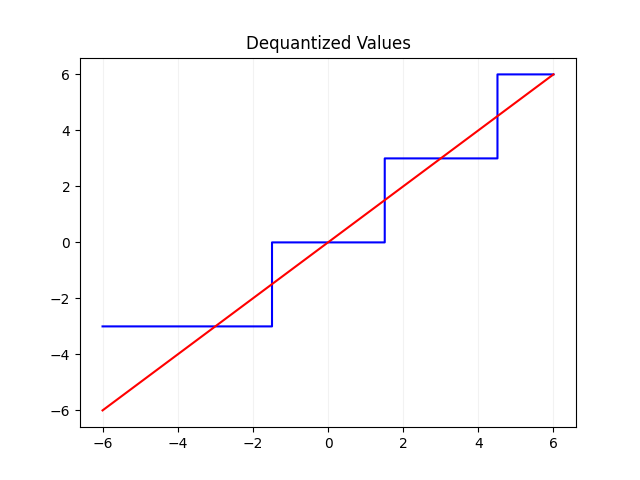
Since there are two types of uniform quantization, we have implemented both, which are Mid-Rise quantization and Mid-Tread quantization.

## **Comment:**

Run Mid-Rise Quantizer by setting m=0 and set the number of bits used for quantization 2 so there are 4 levels. And because of the quantization used is Mid-Rise, the origin lies in the middle of rise.



# **Part 2: Implementing a uniform scalar quantization function with Mid-Tread.**



## **Comment:**

Run Mid-Tread Quantizer by setting m=1 and set the number of bits used for quantization 2 so there are 4 levels. And because of the quantization used is Mid-Tread, the origin lies in the middle of tread. But the Mid-Tread has odd number of levels above or below zero level but we have only 4 levels so we need to map values from -6:-4.5 to zero and map -4.5:6 normally.

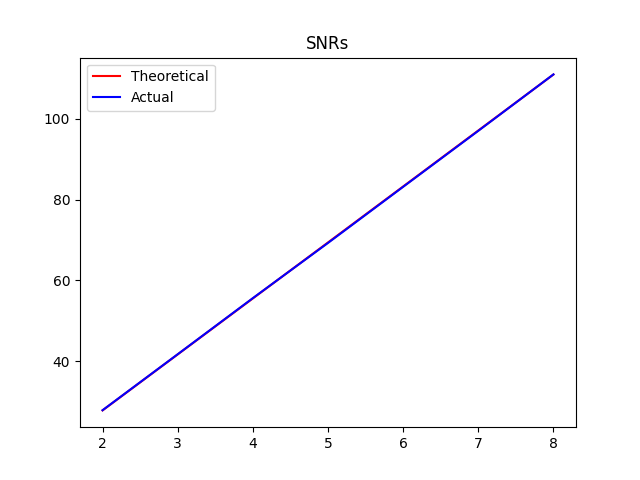
# **Part 3(1):**

As the data generated are uniformly distributed so we can use uniform quantization using Mid-Rise.

We notice that the actual and theoretical SNR increase as the number of bits used for sampling increases because the error decreases.

So the two lines are almost identical.(the red line is behind the blue line)

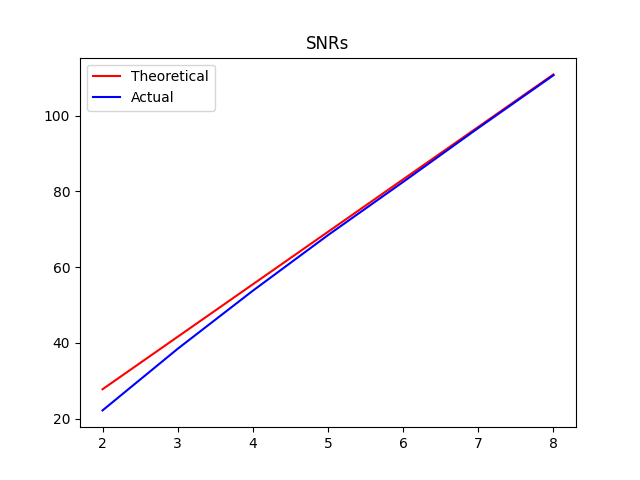
## **Comment:**



# **Part 4(2):**

The same as the previous part but the error is relatively higher because the approximation due to the odd number of levels above and below zero. So the difference between actual SNR and Theoretical is higher in case of small number of bits.

## **Comment:**



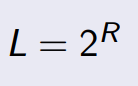
# **Part 5: Testing the uniform quantizer on a non-uniform random input:**

Chart, line chart

Description automatically generated

Figure Fig

## **Comment:**

* This output is plotted in dB so it is linear, while if it was not in dB it should take a curvature form.
* Here we can see that as the number of bits increases the resolution increases so the SNR will also increase.
* That is because the probability of error will decrease and even when there is an error, the difference between it and the nearest level to it will be small because we have increased the number of levels.
* That is according to the equation which defines the number of levels which is:
* Where R is the number of bits.
* And we can also see, that as we increase the number of bits

The theoretical SNR which can be evaluated from this equation:

Text

Description automatically generatedwhere:

* + L is the number of levels
  + mp is the absolute value of the maximum value.
  + P is the signal power.
* Will be close to the actual SNR which will be evaluated from evaluating the difference between the quantized samples and the input sample and dividing the signal power by the noise power.
* And this also happens because of the same reason as the resolution increases so the probability of the error decreases.

# **Part 6: Finally, we now need to use a non-uniform quantizer (u-law) on our quantization functions.**

**Chart, line chart

Description automatically generated**

Figure Fig

## **Comment:**

* Since to be able to apply this method, we need to use the Compander system.
* This happens by using two utility functions.
  + Compressor:
    - It is responsible for compressing the input, by applying a log function to be able minimize the differences between the input values
  + Expander:
    - After applying the quantization we have to return it to its original shape, and this may happen by using the inverse function for the compressor system.
* So, in our case we have used this equation as our compressor:
  + A picture containing diagram

    Description automatically generatedY here is the input to our quantizer function
  + Ln is the natural logarithmic function.
  + (m^) is our input signal but after

Normalization

* + u is a constant which we design our system on it to be able to compress the data as we want, and in this experiment, we have applied different values of u to be able to observe what happen when we change its value.
* After that we insert Y to the quantizer functions.
* Then we apply this function as the Expander function
* Table

  Description automatically generated with low confidenceAfter we got the result, we have evaluated the SNR

For different values of u and get the results shown

In Fig (6).

* And the results are logically correct,
* Since as we increase the value of U the quantization output will increase since we have a better resolution during quantizing the samples, because we decreases the value of the error, so the signal power will increase and the noise power will decreases, and since SNR = (Signal Power / Noise Power), so SNR should also increase.

# **Index:**