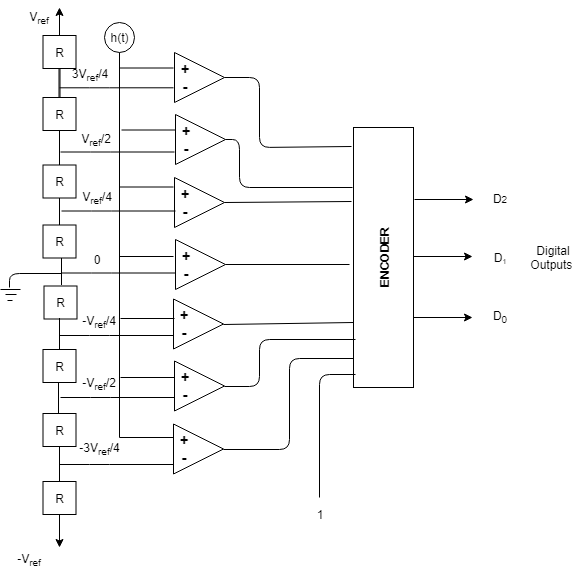
**ECSE 421 – Assignment 2**

**Question 1:**



Bit Mapping: Input on the left, output on the right.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| I7 | I6 | I5 | I4 | I3 | I2 | I1 | I0 | D2 | D1 | D0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Question 2:** Refer to the appendix A for the Matlab code associated with finding the function.

Figure 1: The input function and its quantization function

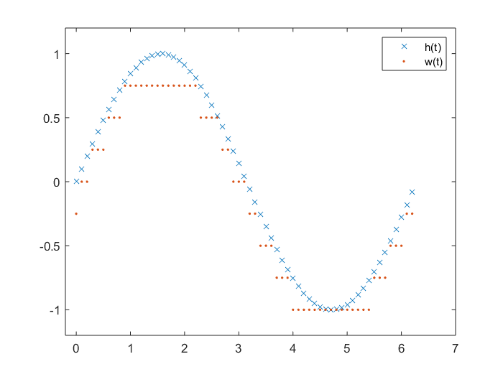
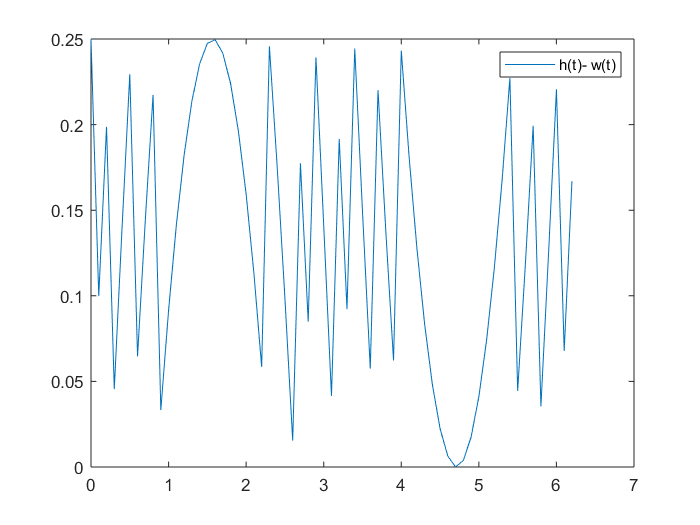


Figure 2: The quantization noise function associated with Figure 1



**Question 3:**

Harvard architecture runs with separate data and instruction buses/cache (parallel), that allow data transfers and instruction fetching to operate simultaneously. Therefore, the CPU has access to 2 busses (address, data) where the data bus is used to carry contents of a memory cell to a destination and the address bus takes the memory address from CPU to memory.

On the other hand, von Neumann memory architecture has only one bus for both the data structures and instruction buses, therefore, both the data transfers and instruction fetching has to coordinate with one another by working sequentially with a fixed sequence of operations. In this case the devices generally have one separate storage which contains the RAM and memory.

1. In this case the Von Neumann memory architecture would be much more suitable because of its single bus system. The single bus system will allow the system to use less power in the system since it only needs to operate one bus and one storage. Additionally, since there is lower computational demand then the bottleneck effect will be lowered because of the lower throughput in the system.
2. Due to the high computational demand of this system the Harvard architecture would be preferred thanks to its higher throughput from its two bus system. The two bus system will allow the system to work in parallel and limit the amount of waiting in between instruction. However, the draw back of the higher throughput is the higher energy/power usage which is not a limiting factor in this scenario.

**Question 4:**

**Part I:**

1. Function is linear if

Given the following function:

Solve:

**AND**

Show: is equal to

The reduced equation is not equal to each other and this can be proven by plugging in numbers. Ie. **AND** . Plugging this in gives the following:

**In conclusion the function is not linear.**

1. A time invariant function is determined as the following:

For: , then

Given the following function:

Solve:

**In conclusion the final functions are equal with a shift in , so it is time invariant.**

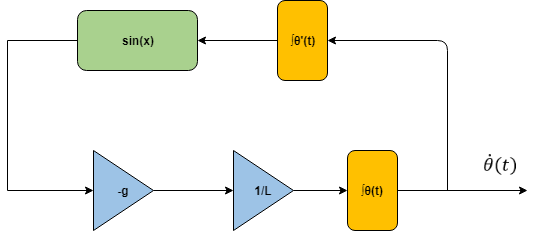
1. We define a memoryless system by the following:

S: [ X → Y] → [ X → Y]

∀ *x* ∈ *X* and ∀ *u* ∈ [*X → Y*], (*S*(*u*))(*x*) = *f*(*u*(*x*)).

**By this definition our function is indeed memoryless because it does not rely on any past values and only depends on the current value of .**

**Part II:**

****

**Using the same definitions for memoryless as above, it is indeed memoryless again because it doesn’t rely on any part values of t.**

**Part III:**

1. The equation is approximated as the following:
2. Function is linear if

Given the following function:

Solve:

**AND**

Show: is equal to

The reduced equation is now equal to one another, this can be proven by plugging in numbers. Ie. **AND** . Plugging this in gives the following:

**In conclusion the function is linear.**

**Part IV:**

**Question 5:**

Since we know the smallest pressure is 20μPa and we know the range is 100 decibels which is derived from a ratio of .

We know the value of the smallest pressure so P1­ = 20μPa.

**Question 6:**

1. In order to measure the bias, we want to make sure the effect of gravity is not accelerating the measured axis; this ensures that the measured axis should in theory be 0 however the value shown will be the bias. Therefore, the value of the bias should be the value when the effect of gravity on the axis is 0.
2. Now, we’re looking for when acceleration is 1g compared to at 0g, so we will orient the accelerometer so that the full force of gravity is being measured along the same axis we found the bias for in part a). Then to calculate a, we just need to subtract b.
3. We can define the affine function model as:

Of course, the model will have sources of uncertainty. Since, it can only take integer values, from the given set {0, …, 2A − 1}. This will lead to the quantization errors we explored in question 2. Additionally, if the acceleration is outside the measurable range then the acceleration will be saturated by the minimum or maximum ranges.

**Appendix**

**Appendix A:**

syms h(t)

% where Vref = 1V

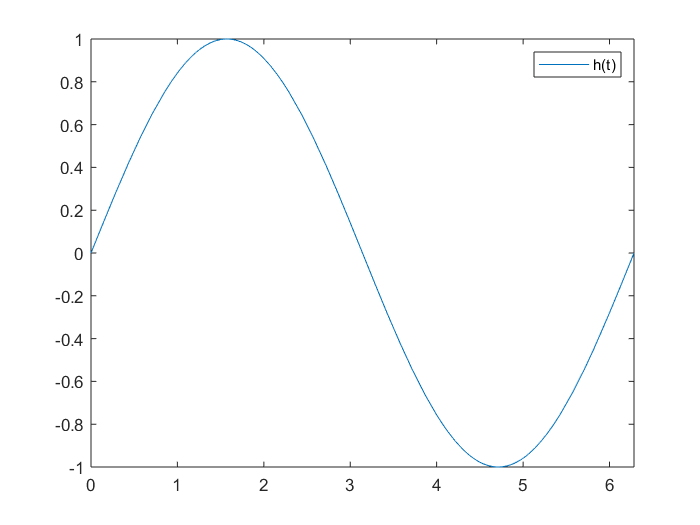
h(t) = sin(t);

% plot the sine function

fplot(h(t), [0, 2\*pi])

legend({'h(t)'},'Location','northeast')

hold off



% find the w(t) function and the quant noise

t = [0:.1:2\*pi]; % Times at which to sample the sine function

sig = sin(t); % Original signal, a sine wave

partition = [-1:.25:1]; % Length 11, to represent 12 intervals

codebook = [-1.25:.25:1]; % Length 12, one entry for each interval

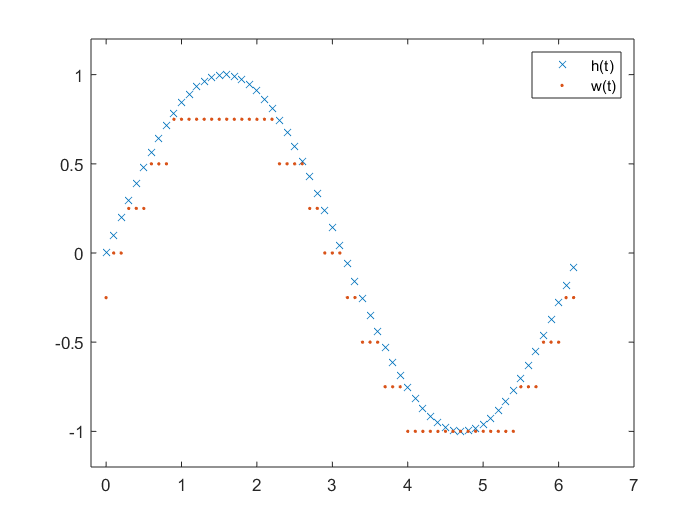
[index,quants] = quantiz(sig,partition,codebook); % Quantize.

plot(t,sig,'x',t,quants,'.')

legend('h(t)','w(t)');

axis([-.2 7 -1.2 1.2])

hold off;



qnoise = sig - quants;

t = [0:0.1:2\*pi];

plot(t, qnoise)

legend({'h(t)- w(t)'},'Location','northeast')

