INTRODUCTION TO FOURIER SERIES

LAB # 09

Spring 2023

CSE301L Signals & Systems Lab

Submitted by: **Ali Asghar**

Registration No. : **21PWCSE2059**

Class Section: **C**

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Submitted to:

Engr. Sumayyea Salahuddin

Date:

June 5, 2023

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Lab Objective(s):

Objectives of this Lab are;

- Power of Continuous & Discrete time Signals •
- Application of Fourier Series
- Synthesis of Square Wave
- Synthesis of Triangular Wave

Task # 01:

Calculate the power of discrete-time cosine signal with period 20, defined over interval 0:19 using the following formula:

$$P = \frac{1}{N}\sum_{0}^{N-1} | x[n] |^2$$

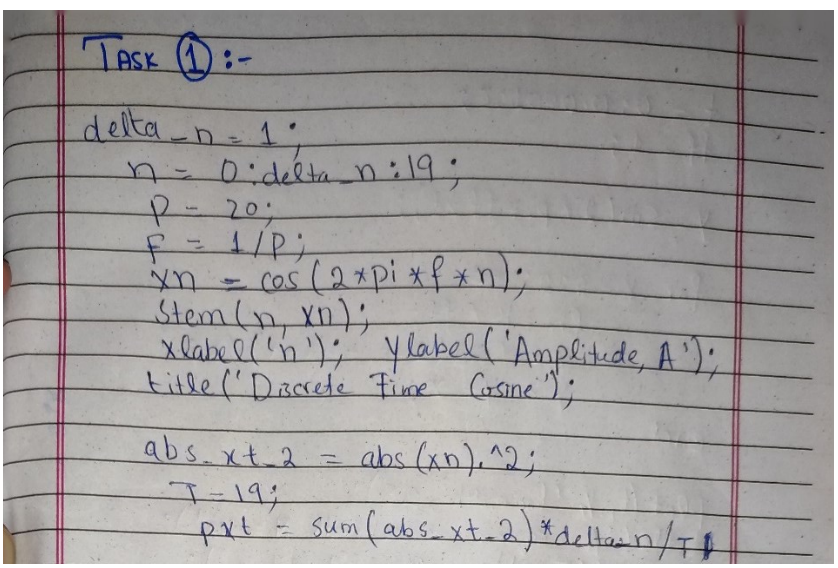
Problem Analysis:

We have to make a signal ranging in a specific time interval

Algorithm:

- Write code
- Execute Code
- Record Results

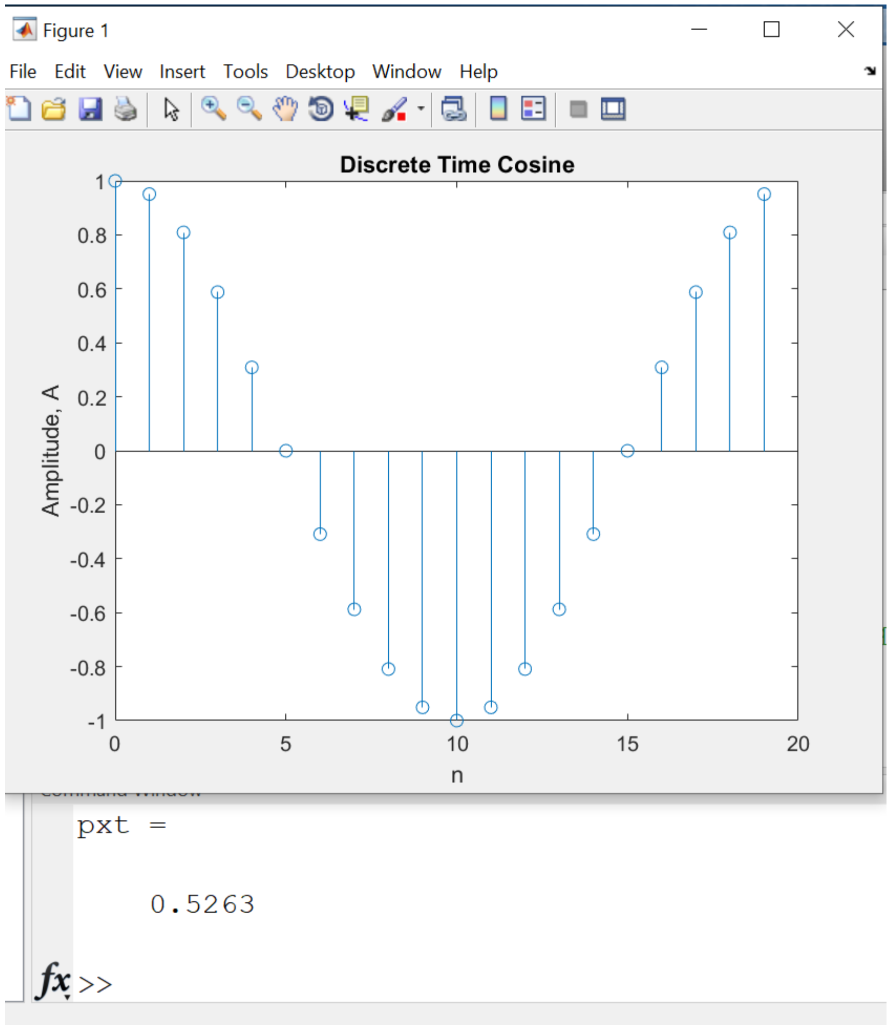
Code:



TASK 1 :-

```
delta_n = 1;
n = 0:delta_n:19;
p = 20;
f = 1/p;
xn = cos(2*pi*f*n);
stem(n, xn);
xlabel('n'); ylabel('Amplitude, A');
title('Discrete Time Cosine');

abs_xt_2 = abs(xn).^2;
T = 19;
pxt = sum(abs_xt_2)*delta_n/T
```

Output / Graphs / Plots / Results:



Discussion and Conclusion:
We can make desired signals in MATLAB

Task # 02:

Write a program that plots the signal s(t).

$$s(t) = \sum_{n=1}^{N} \frac{\sin(2\pi\, nt)}{n}$$ where n = 1, 3, 5, 7, 9 and N = 9 or

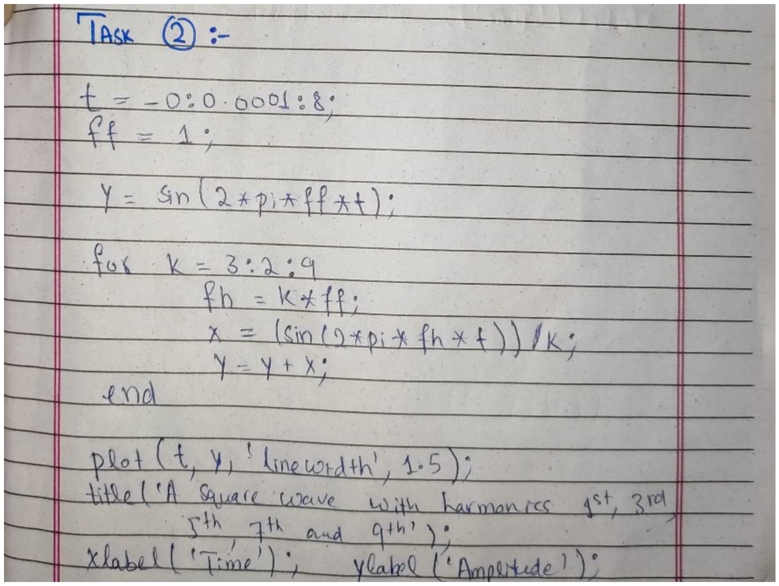
$$s(t) = \sin(2\pi * t) + \frac{\sin(6\pi * t)}{3} + \frac{\sin(10\pi * t)}{5} + \frac{\sin(14\pi * t)}{7} + \frac{\sin(18\pi * t)}{9}$$

Problem Analysis:
To add harmonics and analyze the square wave function.

Algorithm:

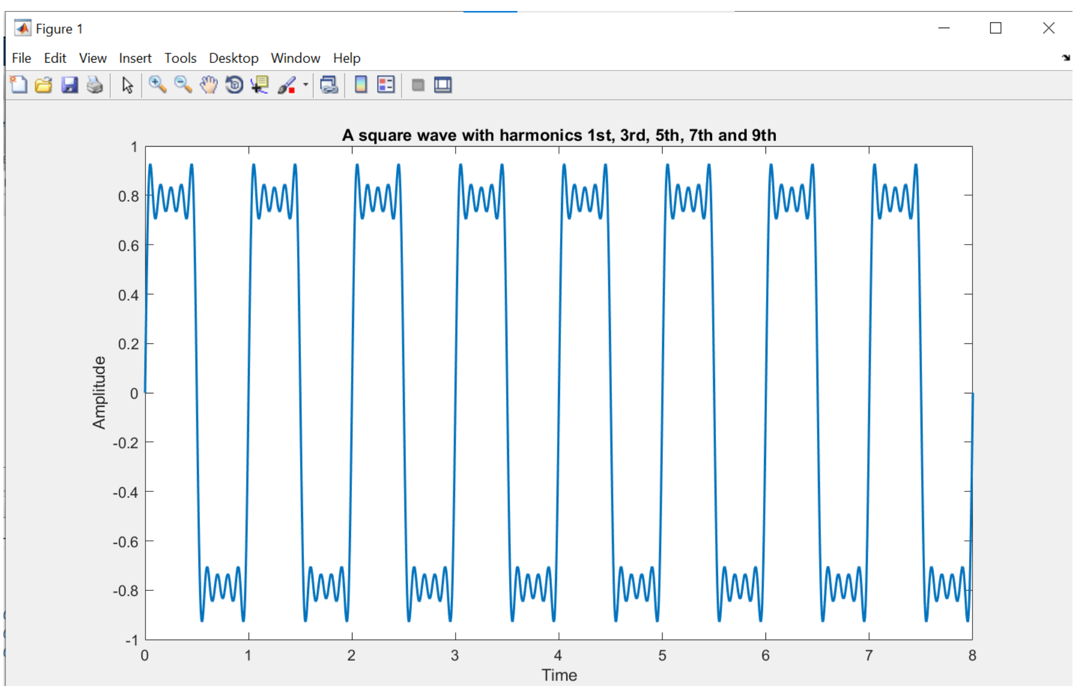
- Write code
- Execute Code
- Record Results

Code:



```
Task (2) :-

t = 0:0.0001:8;
ff = 1;

y = sin(2*pi*ff*t);

for k = 3:2:9
        fh = k*ff;
        x = (sin(2*pi*fh*t))/k;
        y = y + x;
end

plot(t, y, 'linewidth', 1.5);
title('A square wave with harmonics 1st, 3rd,
      5th, 7th and 9th');
xlabel('Time');     ylabel('Amplitude');
```

Output / Graphs / Plots / Results:



Discussion and Conclusion:

We can add more and more harmonics to a square wave under gibbs effect.

Task # 03:

Write a program that plots the signal s(t) but with N = 100.

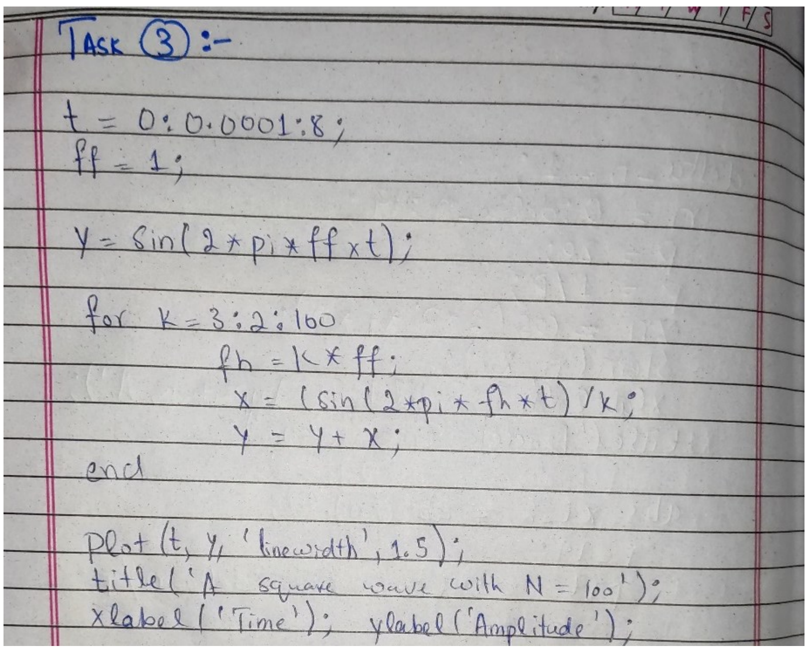
Problem Analysis:

To create a signal with above values.

Algorithm:

- Write code
- Execute Code
- Record Results

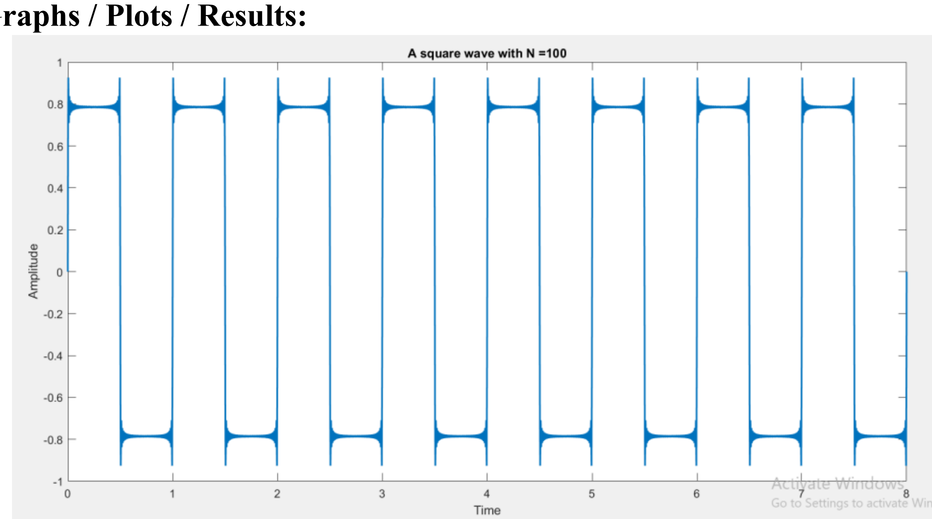
Code:



TASK (3):-

```
t = 0:0.0001:8;
ff = 1;

y = sin(2*pi*ff*t);

for k = 3:2:100
        fh = k*ff;
        x = (sin(2*pi*fh*t)/k;
        y = y + x;
end

plot(t, y, 'linewidth', 1.5);
title('A square wave with N = 100');
xlabel('Time'); ylabel('Amplitude');
```

Output / Graphs / Plots / Results:



Discussion and Conclusion:
We can make new signals with our desired values.

Task # 04:

What do you conclude from TASKS 2 & 3?

Answer:

In Task 2, I generated a square wave by summing specific odd harmonics (1st, 3rd, 5th, 7th, and 9th) using the Fourier series. It starts with a fundamental sine wave and then adds subsequent sine waves with frequencies that are multiples of the fundamental frequency. The resulting waveform approximates a square wave with varying accuracy depending on the number of harmonics included.

In Task 3: I extended the concept of Task 1 by generating a square wave with a larger number of harmonics (100 in this case). By including more harmonics, the waveform becomes more accurate in approximating a square wave. The additional harmonics help refine the shape of the waveform, reducing the distortion present in the approximation.

In conclusion, the provided tasks highlight the application of Fourier series in generating square waveforms with different numbers of harmonics. Task 2 demonstrates the fundamental concept by constructing a square wave using a few specific odd harmonics. It showcases how the addition of harmonics progressively shapes the waveform towards a square-like pattern. Task 3 expands upon this idea by including a larger number of harmonics, resulting in a more accurate square wave approximation. By incorporating more harmonics, the waveform becomes smoother and better resembles the desired square wave shape. These tasks illustrate the power of Fourier series in decomposing complex waveforms into simpler sine wave components and synthesizing them to reconstruct the original waveform.