**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**

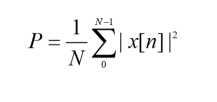
## Lab Objective(s):

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:



**Problem Analysis:**

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

**Algorithm:**

* Write code
* Execute Code
* Record Results

**Code:**

clc;

clear;

close all;

n = 0:1:19; % Discrete time values

P = 20; % Period of the signal

f = 1/P; % Frequency of the signal

xn = cos(2\*pi\*f\*n);

stem(n, xn);

xlabel('n');

ylabel('Amplitude');

title('Discrete Time Cosine Signal');

power\_xn = sum(abs(xn).^2) / (length(n) + 1);

fprintf('Power of the signal: %.4f\n', power\_xn);

**Output / Graphs / Plots / Results:**

**A screen shot of a computer screen

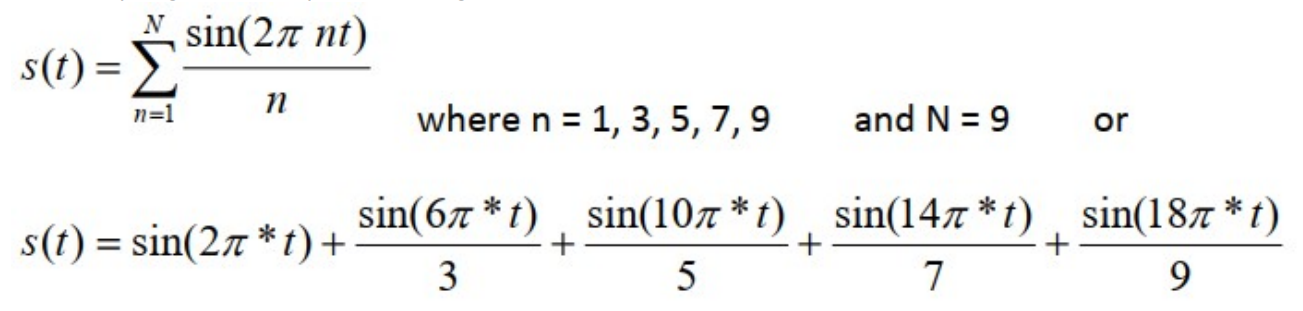
Description automatically generated with low confidence**

**Discussion and Conclusion:**

We can make desired signals in MATLAB

## Task # 02:

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



**Problem Analysis:**

To add harmonics and analyze the square wave function.

**Algorithm:**

* Write code
* Execute Code
* Record Results

**Code:**

clc

clear

close all

t = 0:0.0001:8;

f = 1;

y = sin(2\*pi\*f\*t);

for k = 3:2:9

fh = k \* f;

harmonic = (sin(2\*pi\*fh\*t)) / k;

y = y + harmonic;

end

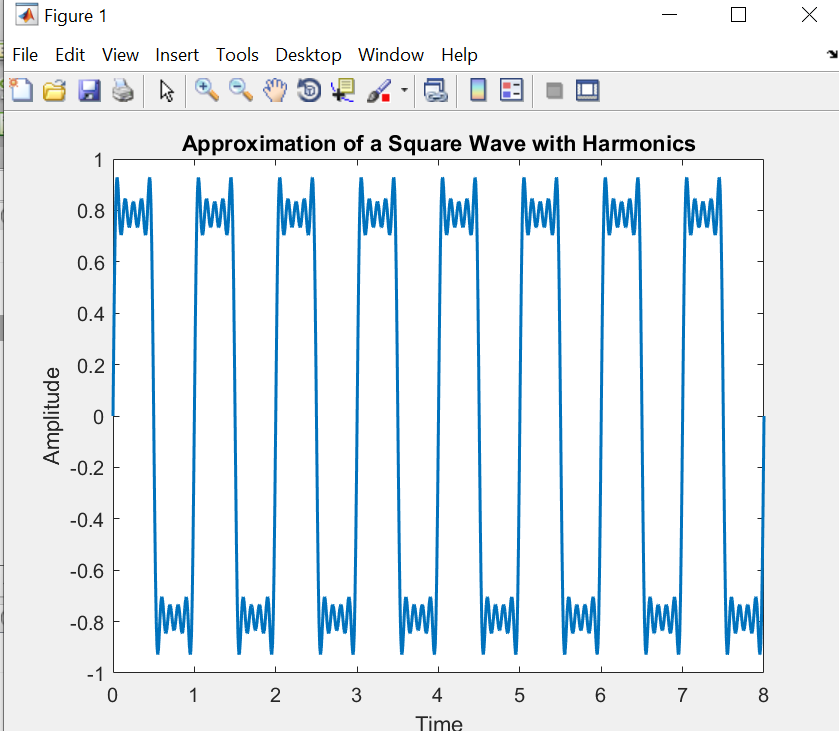
plot(t, y, 'linewidth', 1.5);

title('Approximation of a Square Wave with Harmonics');

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:**

clc

clear

close all

t = 0:0.0001:8;

f = 1;

y = sin(2\*pi\*f\*t);

for k = 3:2:100

fh = k \* f;

harmonic = (sin(2\*pi\*fh\*t)) / k;

y = y + harmonic;

end

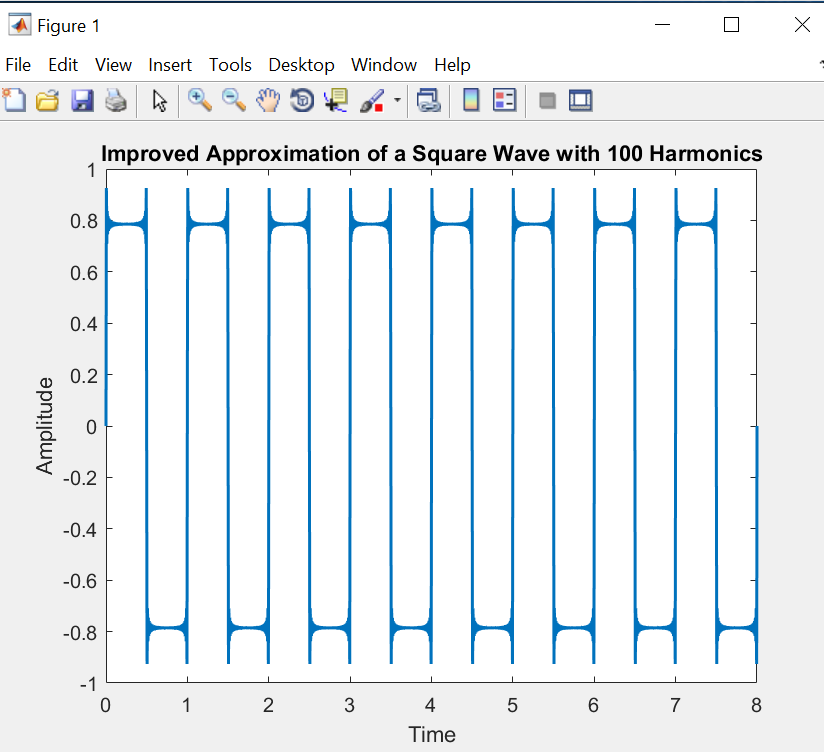
plot(t, y, 'linewidth', 1.5);

title('Improved Approximation of a Square Wave with 100 Harmonics');

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 summary, the provided tasks demonstrate the utilization of Fourier series to generate square waveforms with different numbers of harmonics. Task 2 introduces the concept by constructing a square wave with specific harmonics, while Task 3 expands upon it by incorporating a larger number of harmonics for improved waveform accuracy.