Department Of Communication Systems

Benha National University

CSE207: Signal Analysis and Systems

ECG System using MATLAB

Group Members:

Hossam Ahmed	23020079
Kirolos Tamer	23020161
Abdelrahman Nayef	23020115
Mennatullah elbassir	23020252

Under supervision:

Dr. Heba Allah Adly

ECG System using MATLAB

Table of contents:

Abstract:

In this project, we used MATLAB to work on five important topics in signal processing: time and amplitude manipulation, convolution, Fourier Series, Fourier Transform, and Laplace Transform. The main idea was to see how each of these operations affects a signal and how we can better understand signals in both the time and frequency domains. We used MATLAB to write the code, run simulations, and create plots that helped us visualize how they work.

1. Introduction

Signal processing plays a crucial role in many areas of engineering, especially in communications, where signals are used to carry information. We focused on five main topics that form the foundation of signal analysis: time and amplitude manipulation, convolution, Fourier series, Fourier transform, and Laplace Transform.

Each of these concepts is essential in understanding how signals behave. For example, convolution shows how a system responds to inputs, and both Fourier and Laplace transforms enable us to analyze signals in the frequency domain.

To observe how they work, we used MATLAB to code each topic and visualize the results. We created plots and observed how signals changed with each operation. This made the theoretical concepts we took in lectures clearer and provided a much better understanding of signal processing.

2. Theoretical basis

2.1 Signal Manipulation:

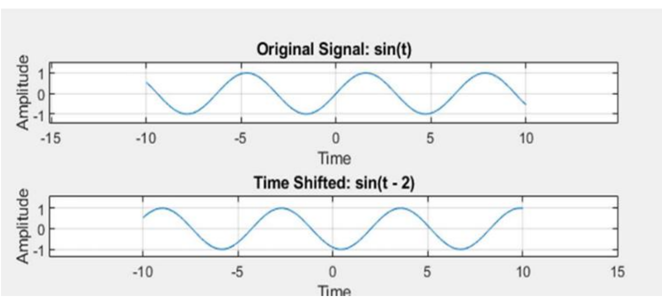
- Signals can be represented using mathematical functions, which means we can apply operations like addition, multiplication, or scaling to create new signals from existing ones.

- One common way to modify a signal is by changing its time characteristics. This involves adjusting how the signal behaves along the time axis.

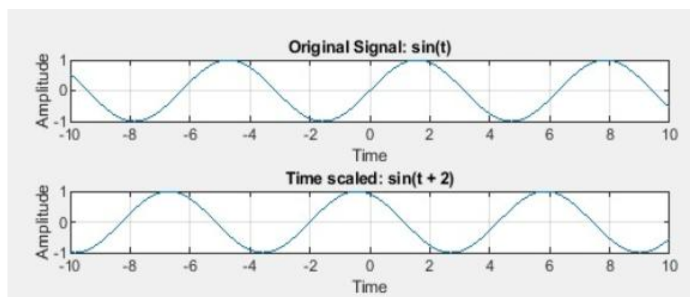
- There are four major time-based transformations:

Time shift (can be delayed or advanced)

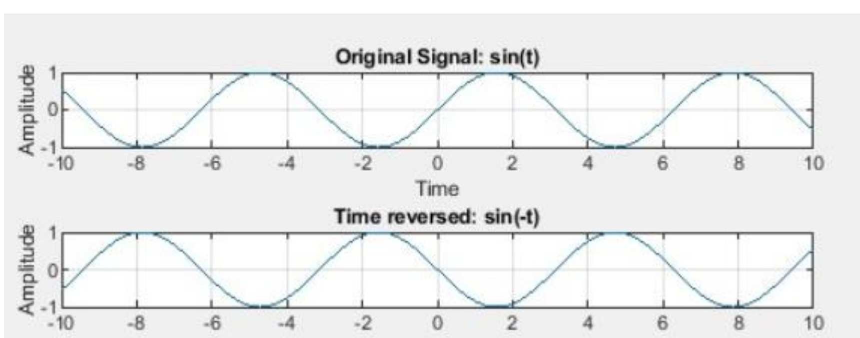
Delayed signal



Advanced signal

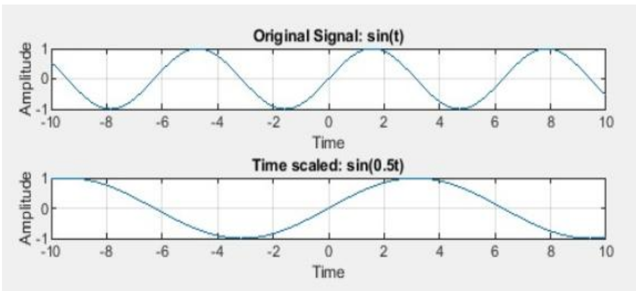


Time reversal

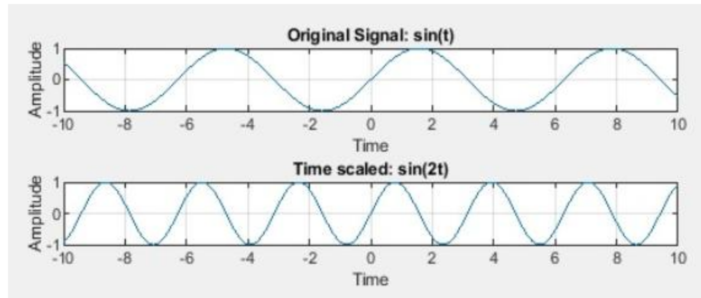


Time scaling (Compression or Expansion)

Expansion

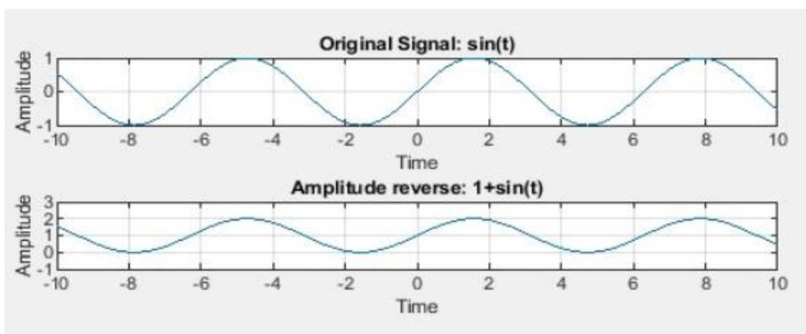


compression



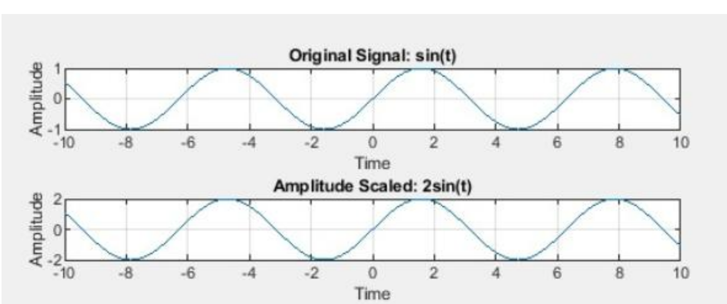
In addition to time transformations, amplitude manipulation is also an essential operation. It involves:

Amplitude shift

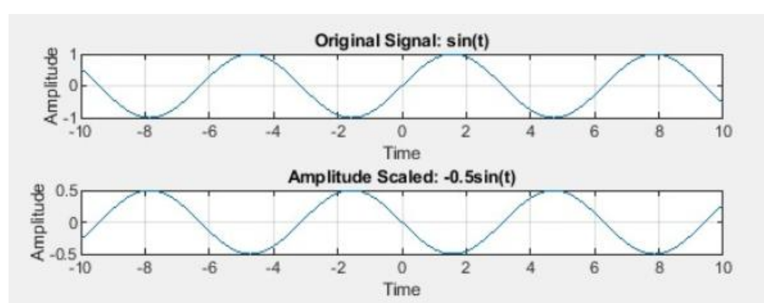


Amplitude scale (Attenuation or Amplification)

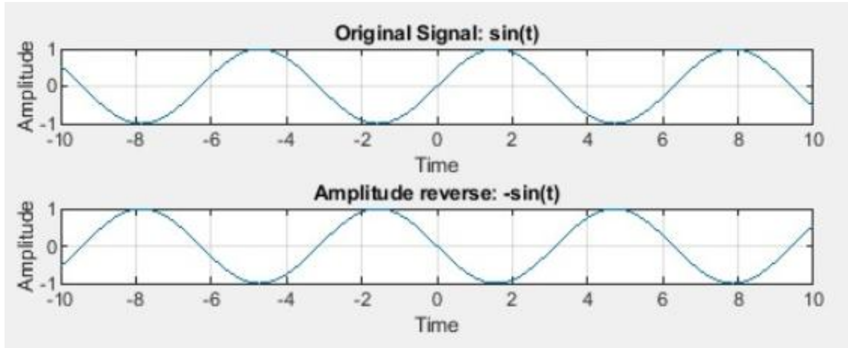
Amplification



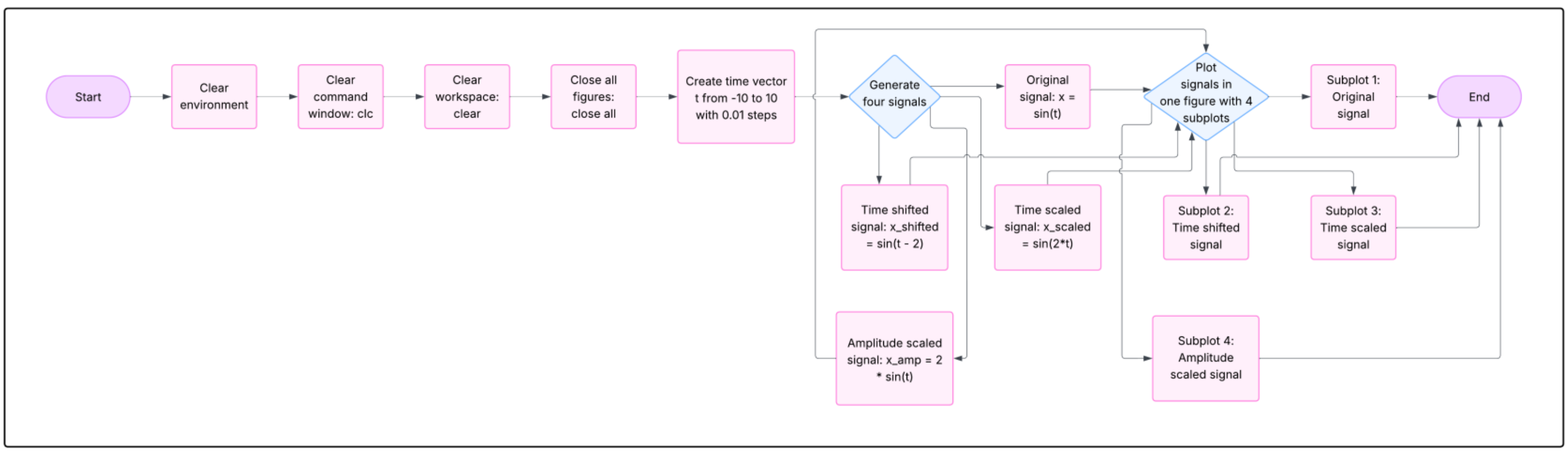
Attenuation



Amplitude reverse



Flowchart:



2.2 Convolution:

Convolution is a method used to understand how a system responds to an input signal. It is fundamental to understand how linear time-invariant (LTI) systems respond to inputs.

Output signal= Input signal$*$Impulse response

It is used in:

- Designing and analyzing filters
- Image processing
- Audio processing
- Communication systems.

Steps to Solve Convolution Problems:

Step 1: Identify the Input Signal x(t)

Choose the signal that enters the system.

Step 2: Identify the Impulse Response h(t)

This represents how the system responds to a unit impulse.

Step 3: Flip the Impulse Response

For continuous-time: replace t with $-\tau \rightarrow h(-\tau)$

Step 4: Shift the Flipped Signal

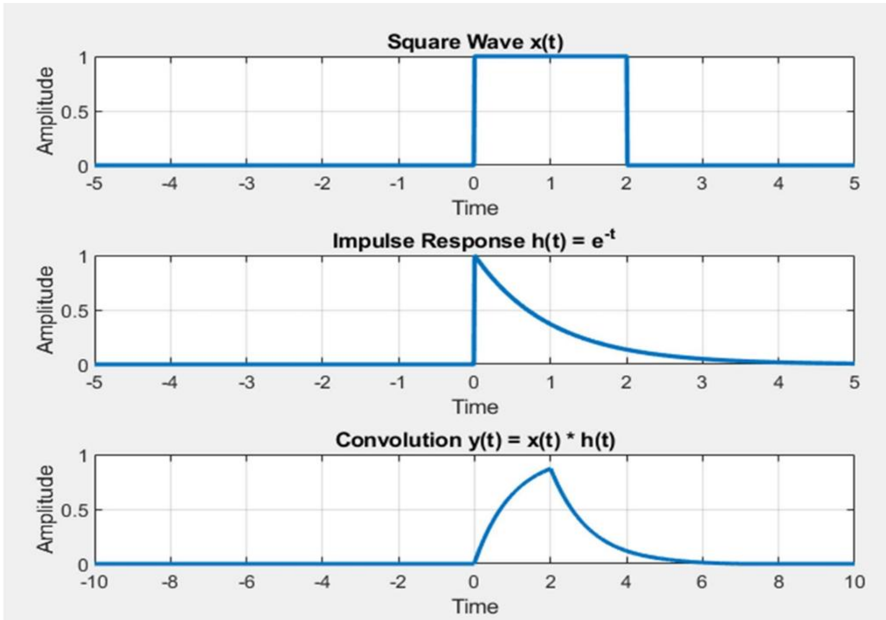
Replace τ with $t - \tau$

Step 5: Multiply the Shifted Signal with the Input Signal

Point-by-point multiply: $x(\tau) \cdot h(t-\tau)$

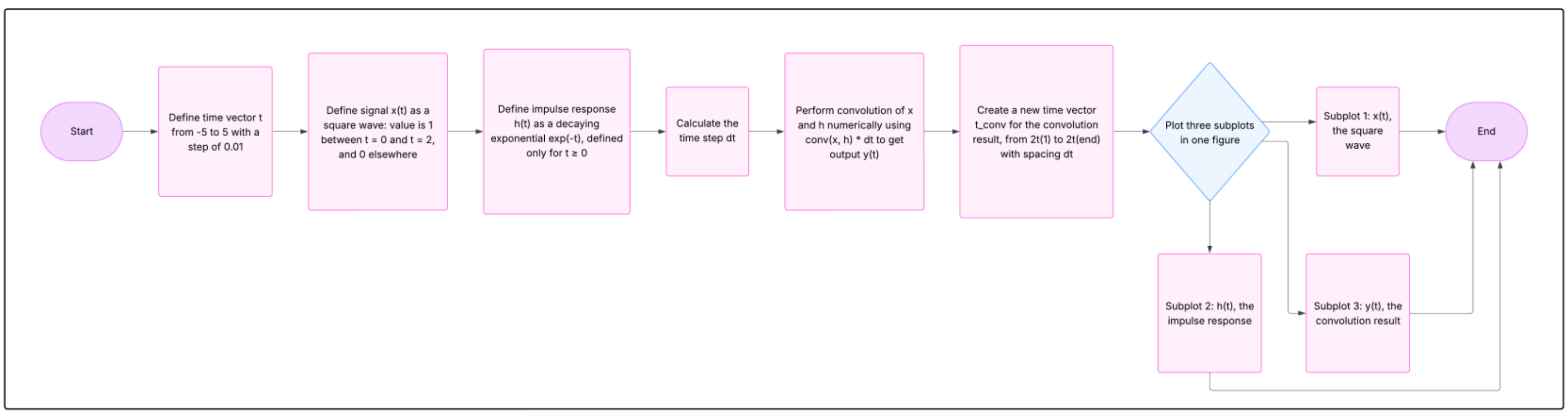
Step 6: Integrate or Sum the Result

$$x(t) = \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau$$



Square Wave x(t)
Amplitude
Time
Impulse Response h(t) = e^-t
Amplitude
Time
Convolution y(t) = x(t) * h(t)
Amplitude
Time

Flowchart:



Start
Define time vector t from -5 to 5 with a step of 0.01
Define signal x(t) as a square wave: value is 1 between t = 0 and t = 2, and 0 elsewhere
Define impulse response h(t) as a decaying exponential exp(-t), defined only for t ≥ 0
Calculate the time step dt
Perform convolution of x and h numerically using conv(x, h) * dt to get output y(t)
Create a new time vector t_conv for the convolution result, from 2t(1) to 2t(end) with spacing dt
Plot three subplots in one figure
Subplot 1: x(t), the square wave
Subplot 2: h(t), the impulse response
Subplot 3: y(t), the convolution result
End

2.3 Application from Convolution:

Signal Filtering in Communication Systems: In communication systems, signals transmitted over a channel often suffer from noise, distortion, or interference. To recover the original signal at the receiver, a filter is applied.

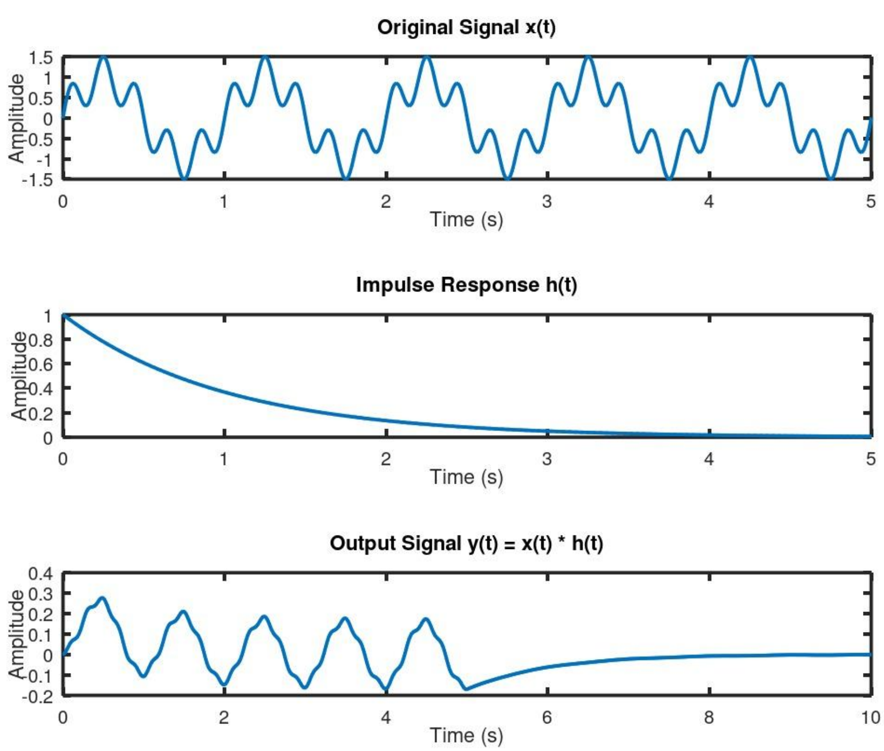
Convolution is widely used in filtering unwanted frequency components from a signal. For example, consider an input signal composed of multiple frequency components:

$$x(t) = \sin(2\pi t) + 0.5 \sin(10\pi t)$$

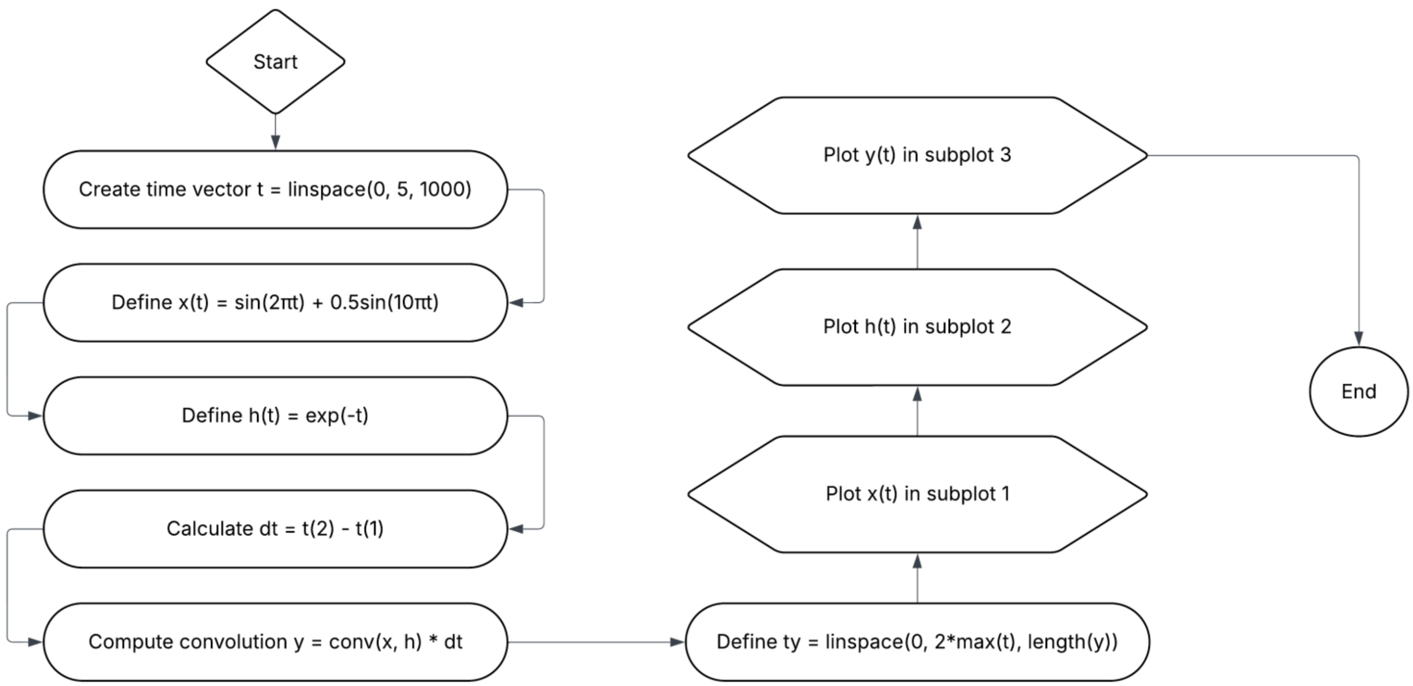
This signal contains both low and high frequencies. To isolate the low-frequency part, we can apply a low-pass filter with the impulse response:

$$h(t) = e^{-t}$$

By convolving x(t) with h(t), the high-frequency components are attenuated, and the output becomes smoother. This demonstrates how convolution can model real-world systems that suppress noise or unwanted signal parts.



Flowchart:



Start
Create time vector t = linspace(0, 5, 1000)
Define x(t) = sin(2πt) + 0.5sin(10πt)
Define h(t) = exp(-t)
Calculate dt = t(2) - t(1)
Compute convolution y = conv(x, h) * dt
Define ty = linspace(0, 2*max(t), length(y))
Plot x(t) in subplot 1
Plot h(t) in subplot 2
Plot y(t) in subplot 3
End

2.4 Fourier Series

One of the most frequent features of natural phenomena is periodicity. Many systems in nature exhibit periodic behavior, such as the audible hum of a mosquito, the symmetry of crystal structures, the vibrations of musical instruments, and the propagation of electromagnetic waves. In all these cases, there is a pattern of displacement that repeats itself over time or space. These patterns can be simple, like a sine wave, or quite complex, as seen in real-world signals.

In signal processing, most of the signals we work with are periodic or can be approximated as periodic over a given interval. The Fourier series is a powerful tool in electrical engineering, communications, control systems, and many other fields where understanding signal behavior is crucial. Its main idea is that any non-sinusoidal periodic signal can be expressed as a sum of simple sine and cosine waves. These sinusoids are mathematically well-behaved and orthogonal, making them ideal building blocks for analyzing and synthesizing complex signals. This decomposition simplifies the analysis, filtering, and compression of signals by focusing on their frequency components.

There are three forms to represent it:

1) Trigonometric form

$$x(t) = a_o + \sum_{n=1}^{\infty} a_n \cos(\omega_o n t) + \sum_{n=1}^{\infty} b_n \sin(\omega_o n t)$$

Where:

$$a_o = \frac{1}{T_o}\int_{T_o} x(t)\, dt$$

$$a_n = \frac{2}{T_o}\int_{T_o} x(t)\cos(\omega_o n t)\, dt$$

$$b_n = \frac{2}{T_o}\int_{T_o} x(t)\sin(\omega_o n t)\, dt$$

$$\omega_o = 2\pi f_o$$

<u>Special cases:</u>

1) If it is even function:

$$b_n = zero$$

2) if it is an odd function:

$$a_o\,, a_n = zero$$

2) Compact trigonometric

$$x(t) = C_o + \sum_{n=1}^{\infty} C_n \cos(\omega_o nt + \theta_o)$$

Where:

$C_o = a_o$

$C_n = \sqrt{{a_n}^2 + {b_n}^2}$

$\theta_o = -\tan^{-1}(\frac{b_n}{a_n})$

3) Complex exponential

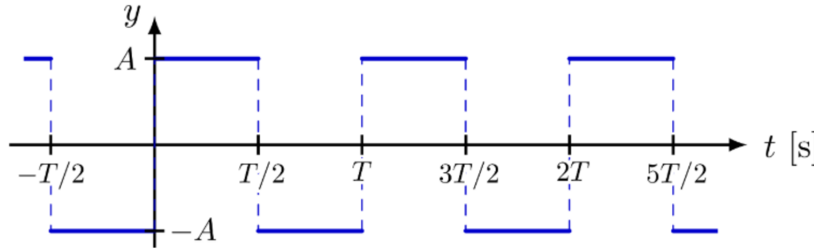
$$x(t) = \sum_{n=1}^{\infty} c_n e^{jn\omega_o t} + c_{-n} e^{-jn\omega_o t}$$

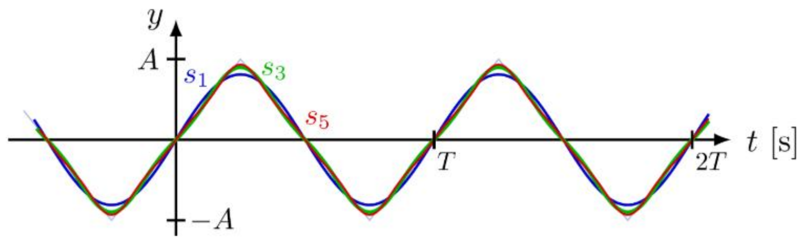
$$x(t) = \sum_{n=-\infty}^{\infty} C_n\, e^{jn\omega_o t}$$

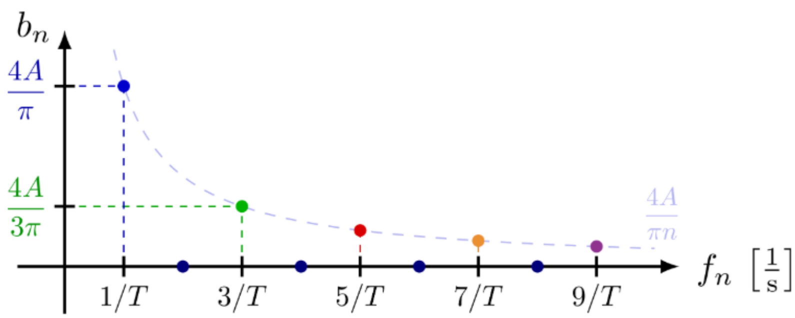
Where:

$c_n = \frac{1}{2}(a_n - jb_n)$

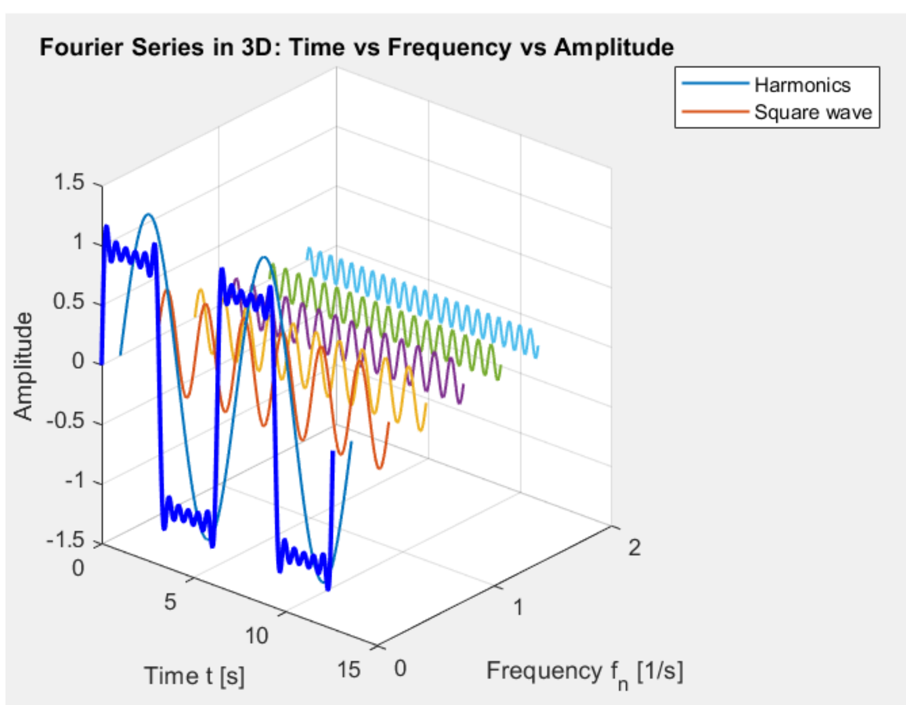
$c_{-n} = \frac{1}{2}(a_n + jb_n)$


y
A
−A
t [s]
−T/2
T/2
T
3T/2
2T
5T/2

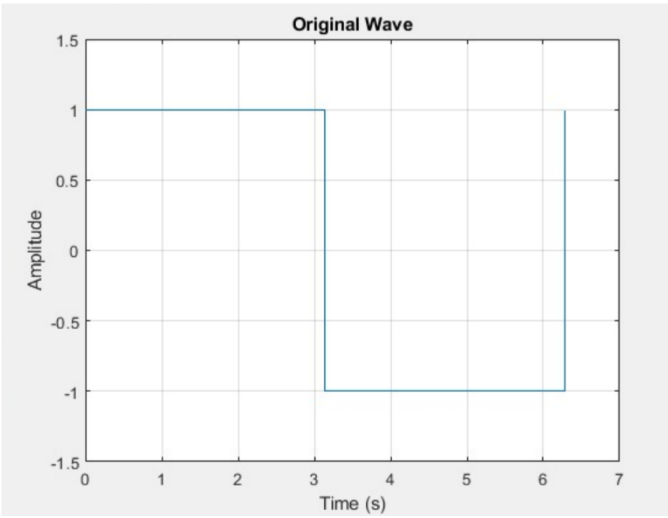

y
A
−A
s1
s3
s5
t [s]
T
2T


bn
4A/π
4A/3π
4A/πn
fn [1/s]
1/T
3/T
5/T
7/T
9/T

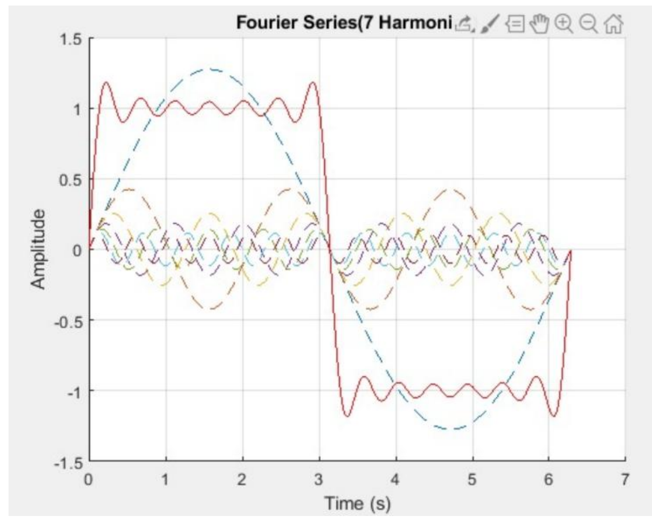
In 3D


Fourier Series in 3D: Time vs Frequency vs Amplitude
Harmonics
Square wave
Amplitude
Time t [s]
Frequency fn [1/s]
1.5
1
0.5
0
-0.5
-1
-1.5
0
5
10
15
0
1
2

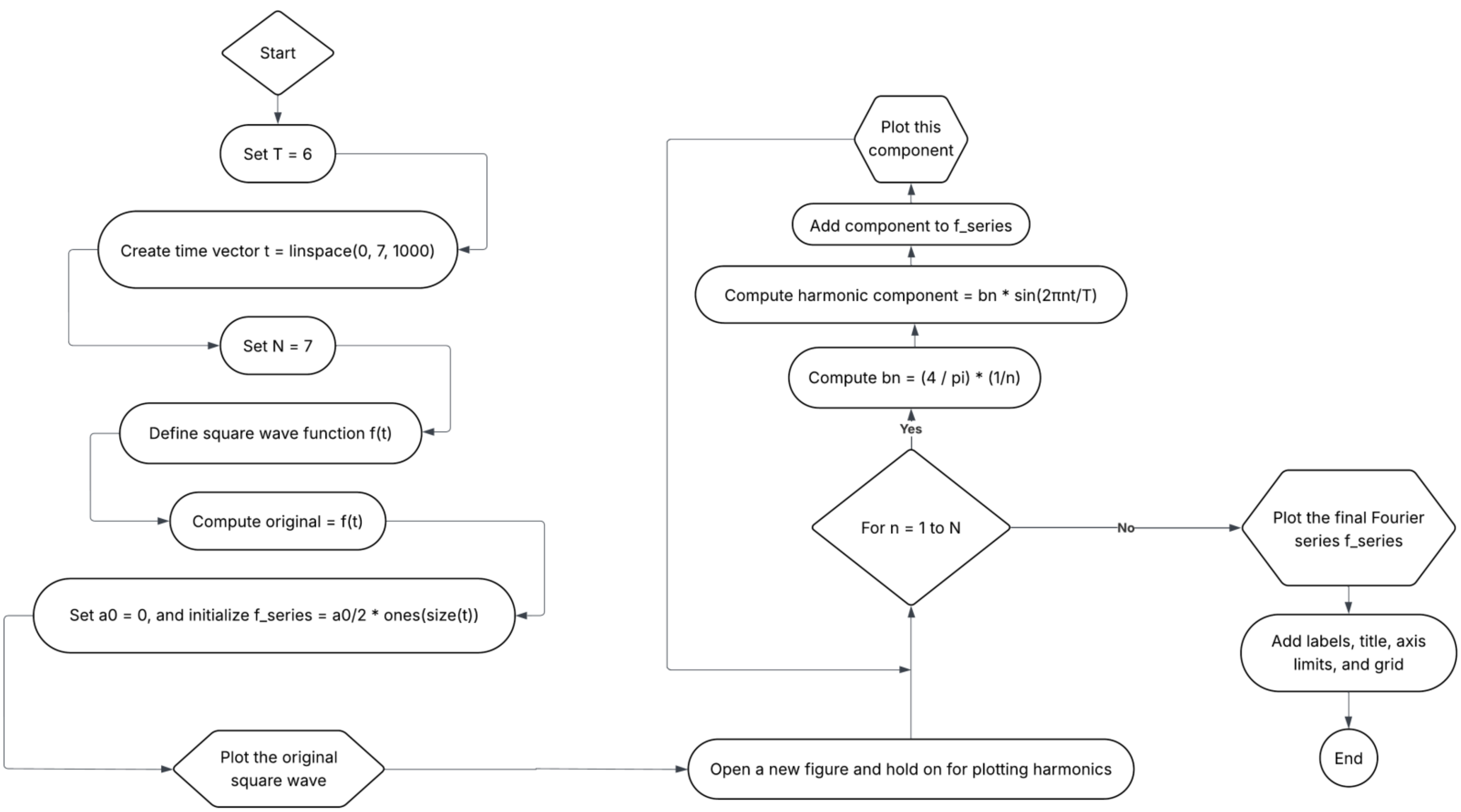
Original square wave



Expressed square wave using Fourier series



Flowchart:



2.5 Application from Fourier Series

Amplitude Modulation (AM) is a method for transmitting information — such as sound or data—by varying the strength (amplitude) of a radio wave. The radio wave, called the carrier, remains at a constant frequency, but its amplitude fluctuates based on the signal you want to transmit. Imagine the carrier wave as a steady, fast-moving wave. When you add your voice or music (the modulating signal), the peaks of the carrier wave grow taller or shorter to match the sound. If you connect these peaks, you obtain a shape known as the envelope, which resembles the original signal.

In short, AM works by generating a strong radio wave that follows the shape of the sound signal, maintaining the same frequency but adjusting its amplitude to carry the information.

When the sound or data signal you're sending is periodic, it can be broken down into a combination of sine and cosine waves using the Fourier series. This helps engineers understand what frequencies are present in the signal and how they affect the AM wave.

In AM, the carrier wave combines with each of these sine waves (the Fourier components of the modulating signal), creating new frequencies called sidebands. These sidebands, situated just above and below the carrier frequency, carry all the information.

Using the Fourier series, we can:

- Predict the frequencies present in the transmitted signal.
- Design filters to remove unwanted parts.
- Calculate bandwidth.
- Rebuild the original signal at the receiver.

The AM signal can be mathematically represented as:

$$s(t) = [1 + k_a m(t)].\cos(2\pi f_c t)$$

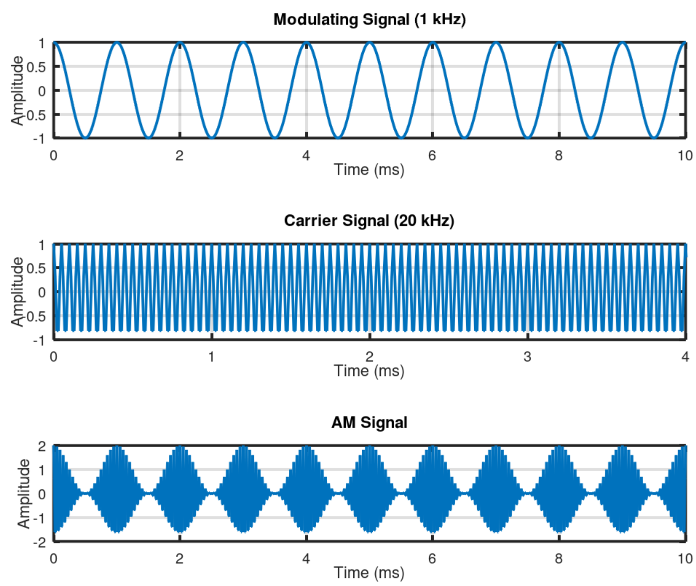
Where:

$s(t)$ is the amplitude-modulated signal

$m(t)$ is the modulating signal

f_c is the carrier frequency

k_a is the amplitude sensitivity

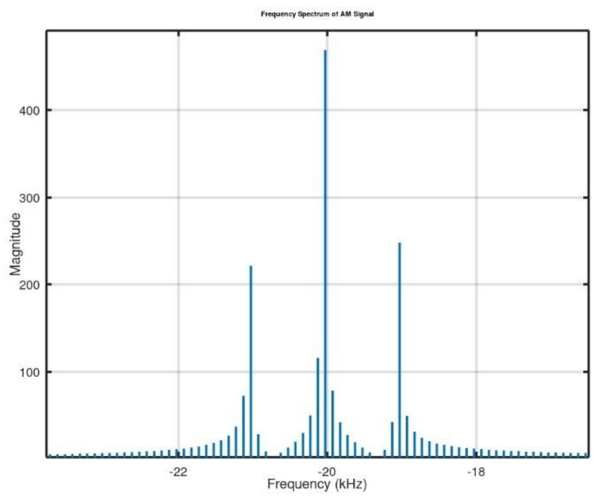


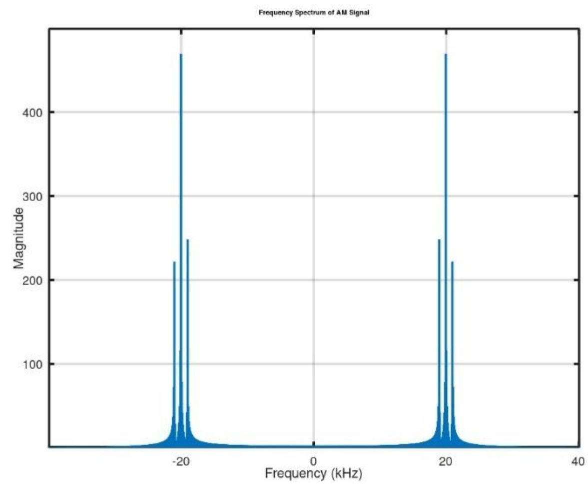
1) Predict the frequencies present in the transmitted signal

In Amplitude Modulation (AM), when a carrier signal is modulated by a periodic baseband signal (like a sine wave), the resulting waveform contains new frequency components. These frequencies can be predicted using Fourier series, especially if the modulating signal is periodic.

So, the resulting frequencies are:

- Carrier frequency f_c
- Upper sideband $f_c + f_m$
- Lower sideband $f_c - f_m$





This is a well-known outcome derived from analyzing a modulated signal using the Fourier series or transform.

When the modulating signal is not perfectly periodic like real speech or music we use the Fourier Transform instead of the Fourier Series. The Fourier Transform allows us to analyze the full spectrum of any signal, even if it doesn't repeat. It helps visualize the energy content across all frequencies and is widely used in digital signal processing to analyze and filter AM signals in both transmitters and receivers.

2) Calculate the Bandwidth of an AM Signal

The bandwidth of an AM signal is the range of frequencies the signal occupies. From the previous analysis:

If the modulating signal has a maximum frequency f_m,

Then the AM signal will have significant frequency components between:

$f_c - f_m$ to $f_c + f_m$

So, the total bandwidth is:

$BW = 2f_m$

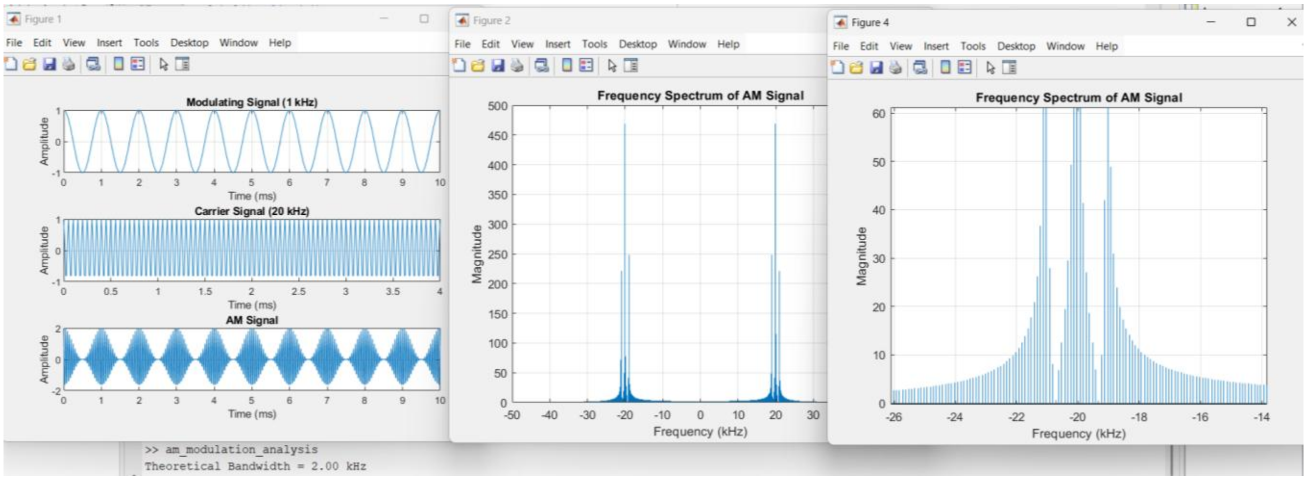
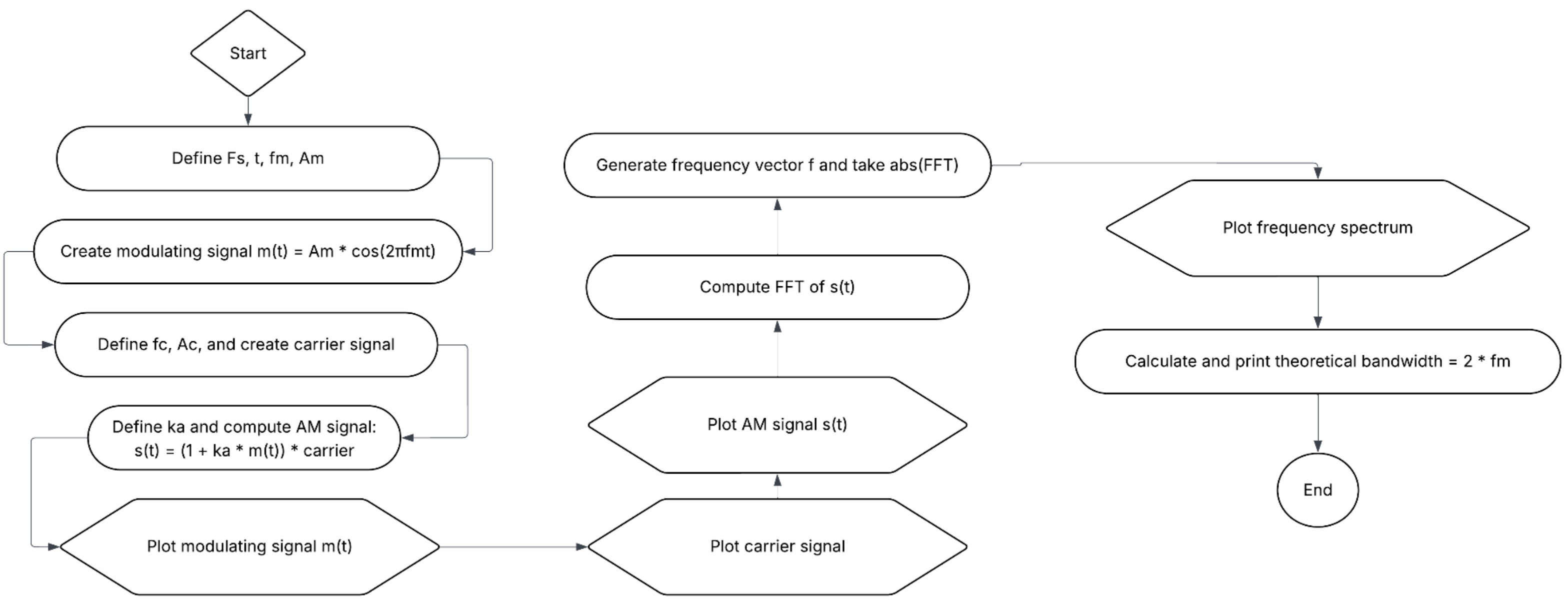


Figure 1
File Edit View Insert Tools Desktop Window Help
Modulating Signal (1 kHz)
Amplitude
Time (ms)
Carrier Signal (20 kHz)
Amplitude
Time (ms)
AM Signal
Amplitude
Time (ms)
Figure 2
Frequency Spectrum of AM Signal
Magnitude
Frequency (kHz)
Figure 4
Frequency Spectrum of AM Signal
Magnitude
Frequency (kHz)
>> am_modulation_analysis
Theoretical Bandwidth = 2.00 kHz

Flow Chart:


Start
Define Fs, t, fm, Am
Create modulating signal m(t) = Am * cos(2πfmt)
Define fc, Ac, and create carrier signal
Define ka and compute AM signal:
s(t) = (1 + ka * m(t)) * carrier
Plot modulating signal m(t)
Plot carrier signal
Plot AM signal s(t)
Compute FFT of s(t)
Generate frequency vector f and take abs(FFT)
Plot frequency spectrum
Calculate and print theoretical bandwidth = 2 * fm
End

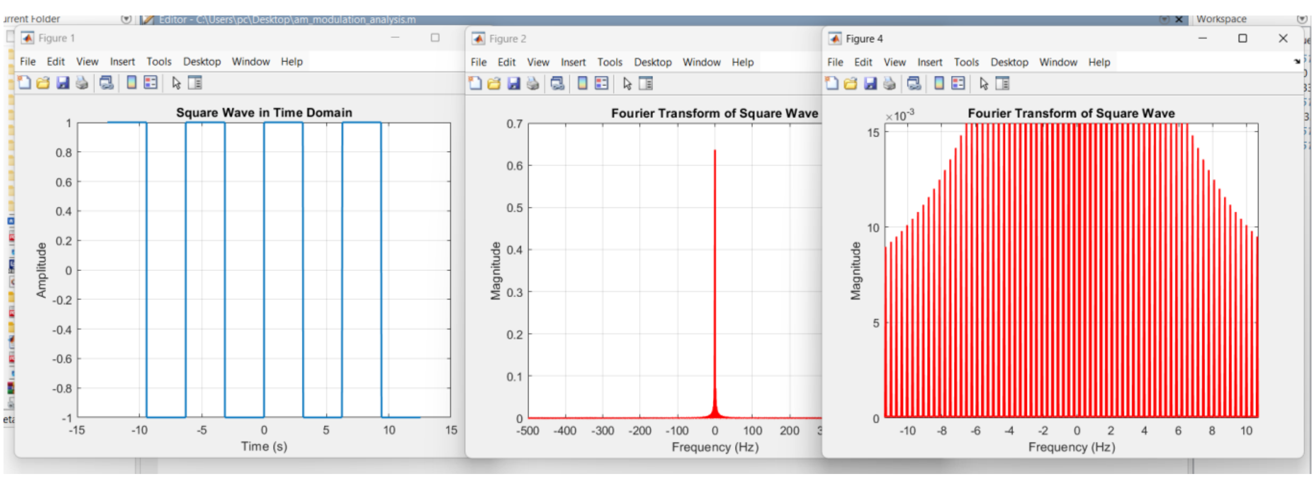
2.6 Fourier Transform

Unlike the Fourier series, which is designed for periodic signals, the Fourier Transform allows us to analyze non-periodic and real-world signals. While the Fourier series expresses a repeating signal as a sum of sinusoids, the Fourier transform generalizes this concept to handle signals that do not repeat. It provides a way to convert a time-domain signal f(t) into its frequency-domain representation $F(\omega)$ revealing how much of each frequency is present in the signal.

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \times e^{-j\omega t} dt$$

With the inverse Fourier Transform, we can also reconstruct the original time-domain signal from its frequency-domain components. This two-way transformation makes the Fourier Transform a powerful tool in signal processing, communications, image analysis, and many areas of engineering

$$f(t) = \frac{1}{2\pi} \times \int_{-\infty}^{\infty} F(\omega) \times e^{j\omega t}\, d\omega$$



2.7 Application from Fourier Transform

Digital Communication is used for the transfer of messages all over the globe using features such as email, instant messaging, and video conferencing. ▪ Electronics and communication systems make use of Digital Communication in domains like pattern detection, robotics, and image enlargement. ▪ Digital Communication is used in signal compression, such as video compression, to save bandwidth during transmission. ▪ Fields in electronics engineering, like digital signal processing, make use of digital communication to understand signal processing. ▪ Digital Communication is used in spacecraft for space-to-Earth communication, utilizing signals.

The same thing that happens in systems like:

application	Description
Modems	When you connect to the Internet via a telephone line, the device converts the 1's and 0's into wave signals (ASK or other).
RFID / NFC	Smart ID cards transmit digital data in the form of waves.
Communication between satellites	They use the same principle but at higher frequencies.
Bluetooth and Wi-Fi	Different modulation types are used, but the idea is the same: representing bits as waves.
Remote control systems	Data is sent in the form of wave pulses (similar to ASK).

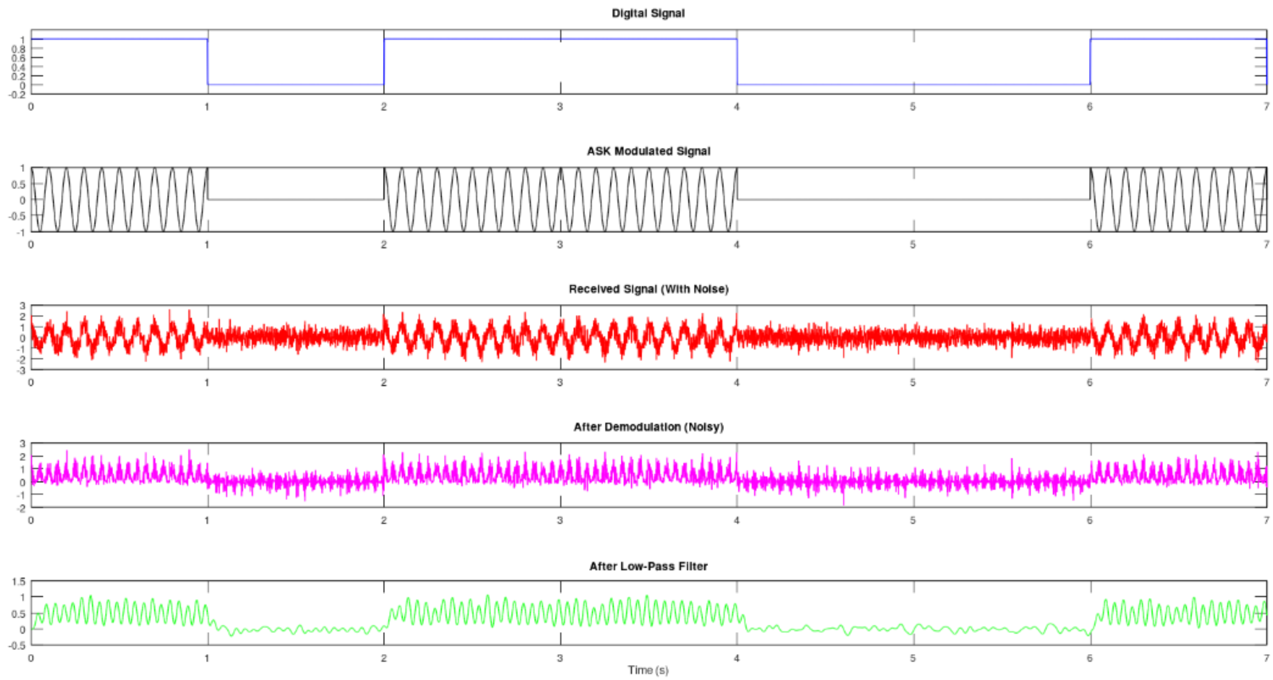
In modern digital communication systems, signals often undergo several transformations to ensure reliable transmission and reception. One of the most essential tools used to analyze and understand these signals is the Fourier Transform. It allows us to move from the time domain — where signals are defined by their amplitude over time — to the frequency domain, where we can study how much of each frequency exists in the signal.

This is particularly useful in modulation techniques like Amplitude Shift Keying (ASK), where the amplitude of a carrier wave changes based on the digital input data. By applying Fourier analysis, engineers can visualize how the signal's frequency content changes during transmission, how noise affects those frequencies, and how to design filters that help recover the original data.

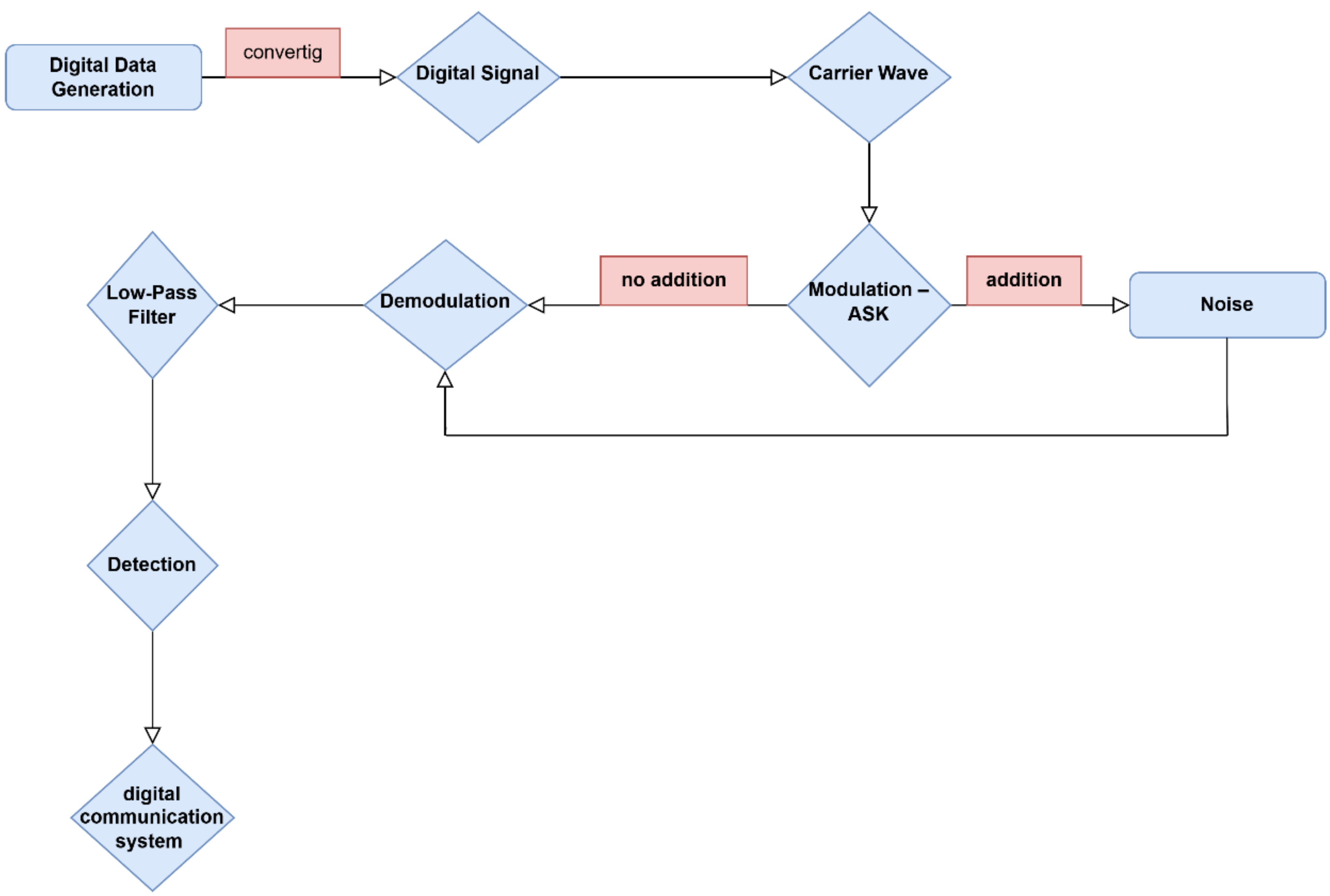
Fourier Transform isn't just a mathematical tool — it's a practical foundation for understanding and improving real-world communication systems. Whether it's in wireless data transmission, satellite links, or even audio compression, the ability to analyze signals in the frequency domain helps make digital communication faster, more efficient, and more reliable.

The same concept is applied in various systems such as:

- Wireless communication, where signal spectra are managed to avoid interference.
- Radar and sonar, where reflected signals are analyzed using Fourier methods.
- Audio and speech processing, for separating noise from useful information.
- Biomedical signal processing, like ECG and EEG analysis, which rely on Fourier analysis for diagnosing abnormalities.


Digital Signal
ASK Modulated Signal
Received Signal (With Noise)
After Demodulation (Noisy)
After Low-Pass Filter
Time (s)

Flow Chart:


Digital Data Generation
convertig
Digital Signal
Carrier Wave
Modulation – ASK
addition
Noise
no addition
Demodulation
Low-Pass Filter
Detection
digital communication system

2.8 Laplace Transform

One of the most powerful mathematical tools used in analyzing dynamic systems especially in engineering fields like control systems, electronics, and signal processing is the Laplace Transform, While the Fourier series focuses on representing periodic signals in the frequency domain, the Laplace Transform is more general: it allows us to analyze both transient and steady-state behavior of systems, even when signals are non-periodic or impulsive in nature.

Think of it like this: if the time domain is a messy battlefield full of differential equations and sudden spikes (like turning on a switch), the Laplace domain is the calm strategic overview where solving those equations becomes algebraic and easier to visualize.

This tool converts time-domain functions (which are usually hard to deal with, especially when they involve differential equations) into s-domain functions, where differentiation becomes simple multiplication and integration becomes division.

Why it is important?

- Solving Differential Equations: Laplace simplifies solving linear time-invariant (LTI) systems by converting them to algebraic equations

- Control Systems: It's foundational for designing controllers like PID, and analyzing system stability and transfer functions.

- Signal Processing: It's useful for system response, filtering, and modeling analog circuits.

- Initial & Final Value Theorems: Helps predict long-term and short-term behavior of a system without fully solving it.

Mathematical Definition:

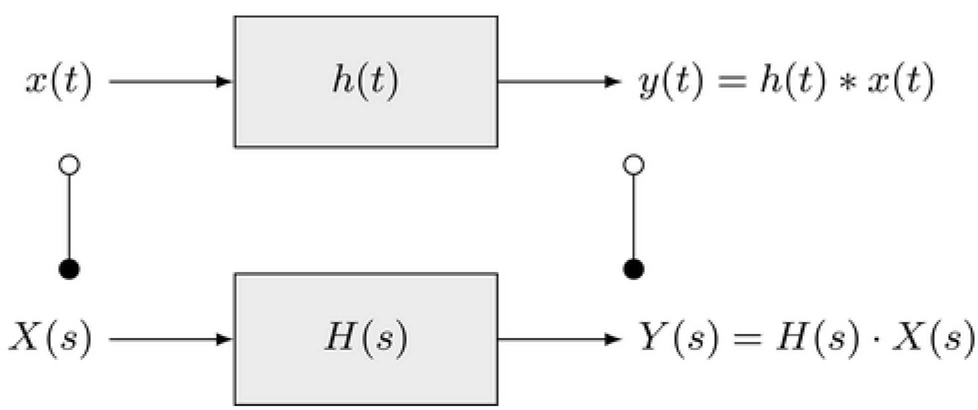
$$L\{f(t)\} = F(s) = \int_{0}^{\infty} f(t)e^{-st}dt$$

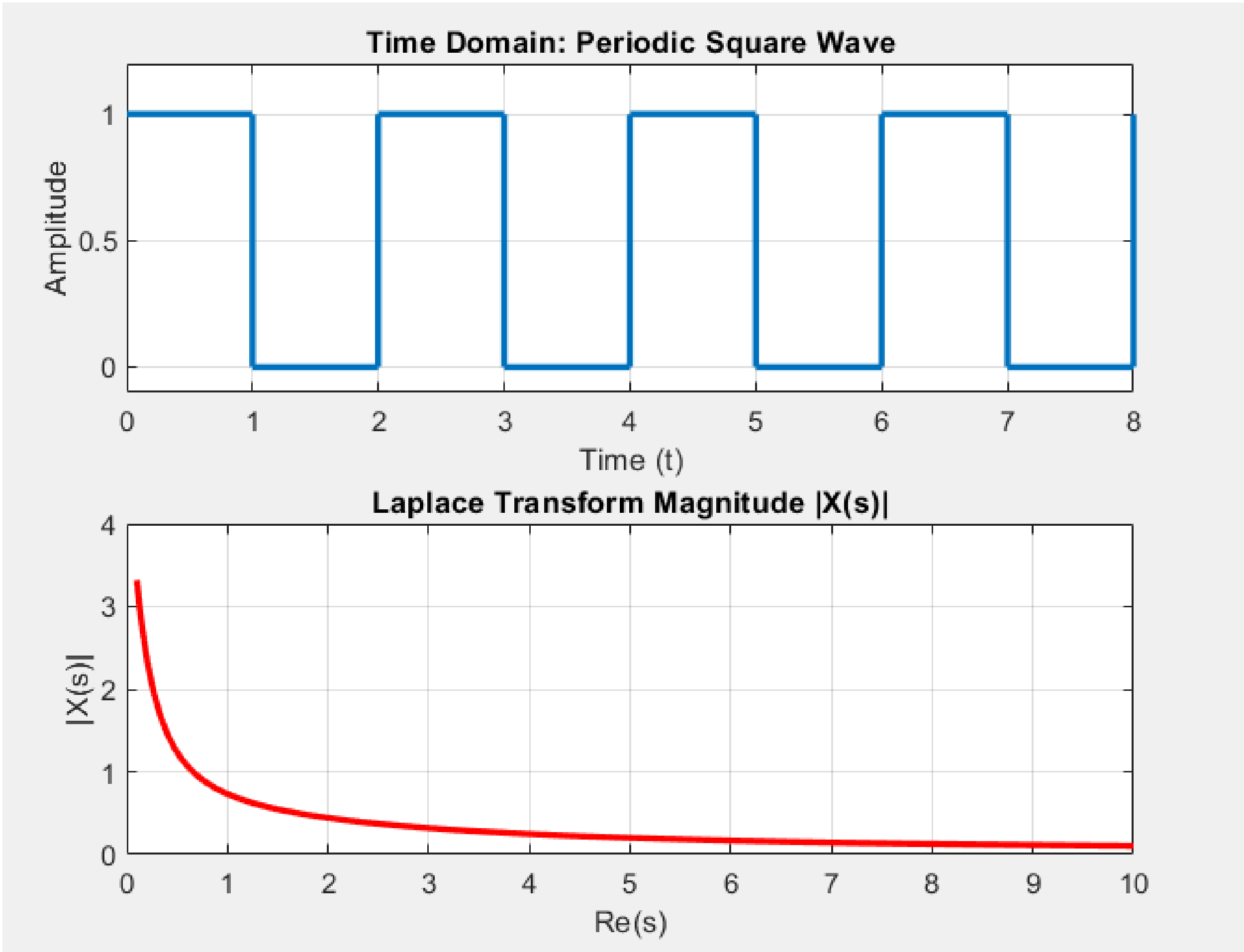
Where:

- f(t): time-domain function
- F(s): Laplace domain (s-domain) function
- s: complex variable s = σ + j*ω

Properties of the Laplace Transform:

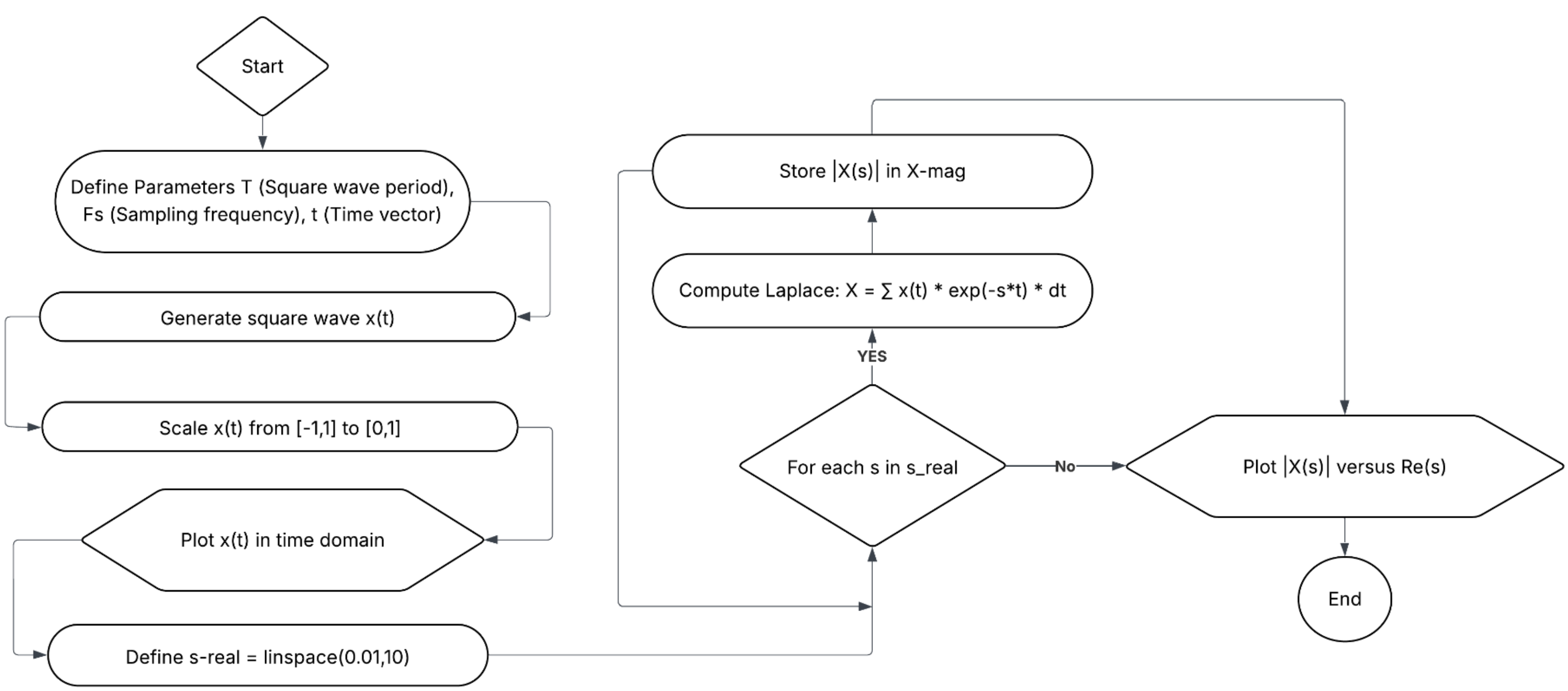
I. Linearity: $L\{af(t) + bg(t)\} = aF(s) + bG(s)$
II. Time Shifting: $L\{f(t - t_0)u(t - t_0)\} = e^{st_0}F(s)$
III. Differentiation: $L\{f'(t)\} = sF(s) - f(0)$
IV. Integration: $L\left\{\int_0^t f(\tau)d\tau\right\} = \frac{1}{8}F(s)$
V. Convolution (in time domain becomes multiplication): $L\{f(t) * g(t)\} = F(s)G(s)$





Time Domain: Periodic Square Wave
Amplitude
Time (t)
Laplace Transform Magnitude |X(s)|
|X(s)|
Re(s)

Flow Chart:

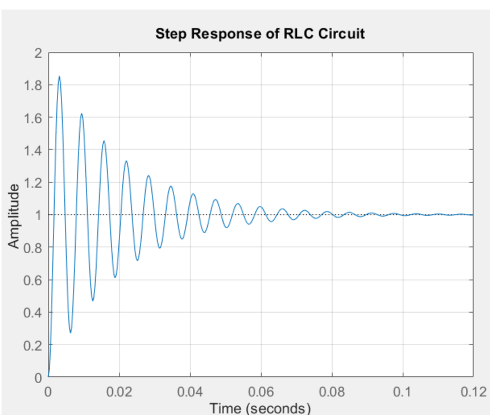


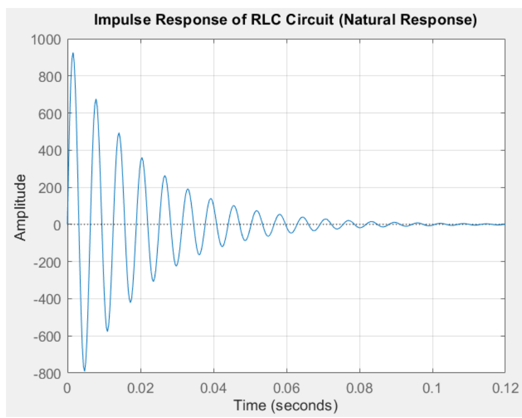
Start
Define Parameters T (Square wave period), Fs (Sampling frequency), t (Time vector)
Generate square wave x(t)
Scale x(t) from [-1,1] to [0,1]
Plot x(t) in time domain
Define s-real = linspace(0.01,10)
For each s in s_real
YES
Compute Laplace: X = ∑ x(t) * exp(-s*t) * dt
Store |X(s)| in X-mag
No
Plot |X(s)| versus Re(s)
End

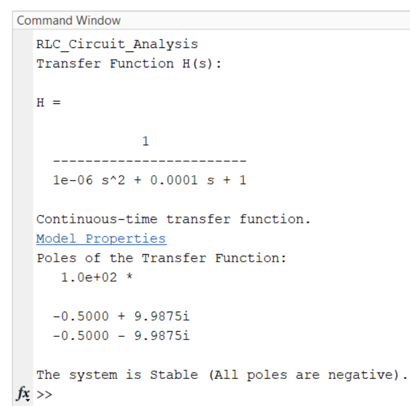
2.9 Application from Laplace Transform

In the field of Signals and Systems, we often encounter complex electrical circuits like the RLC circuit (Resistor-Inductor-Capacitor), which are used in control systems, resonance, and signal processing. This application demonstrates how to use Laplace Transform to analyze an RLC circuit, obtain its Transfer Function, and study its response to an input signal.

- The circuit is a Series RLC Circuit composed of:
  - Resistor R
  - Inductor L
  - Capacitor C
- The input voltage $V_{in}(t)$ is applied to the circuit, and the output voltage $V_{out}(t)$ is measured across the capacitor.







1. Deriving the Differential Equation Using KCL (Kirchhoff's Voltage Law):

$$V_{in}(t) = V_R(t) + V_L(t) + V_C(t)$$

Where:

- $V_R(t) = R \times i(t)$
- $V_L(t) = L \times \frac{di(t)}{dt}$
- $V_C(t) = \frac{1}{C}\int i(t)dt$

$$V_{in}(t) = [R \times i(t)] + \left[L \times \frac{di(t)}{dt}\right] + \left[\frac{1}{C}\int i(t)dt\right]$$

2. Applying Laplace Transform:
   - $V_{in}(t) => V_{in}(S)$
   - $i(t) => I(S)$
   - $\frac{di(t)}{dt} => sI(S)$
   - $\int i(t)dt => \frac{I(s)}{s}$

I. The equation becomes:

$$Vin(s) = RI(s) + LsI(s) + \frac{1}{sC}I(s)$$

II. Solve for I(S):

$$I(s) = \frac{Vin(s)}{R + Ls + \frac{1}{sC}}$$

III. $V_{out}(s)$ is the voltage across the capacitor:

$$V_{out}(s) = \frac{1}{sC}I(s)$$

IV. Substituting I(s) into the equation:

$$H(s) = \frac{Vin(s)}{Vout(s)} = \frac{1}{s^2LC + sRC + 1}$$

3. Transfer Function Analysis:

$$H(s) = \frac{1}{LC} \times \frac{1}{s^2 + \frac{R}{L}s + \frac{1}{LC}}$$

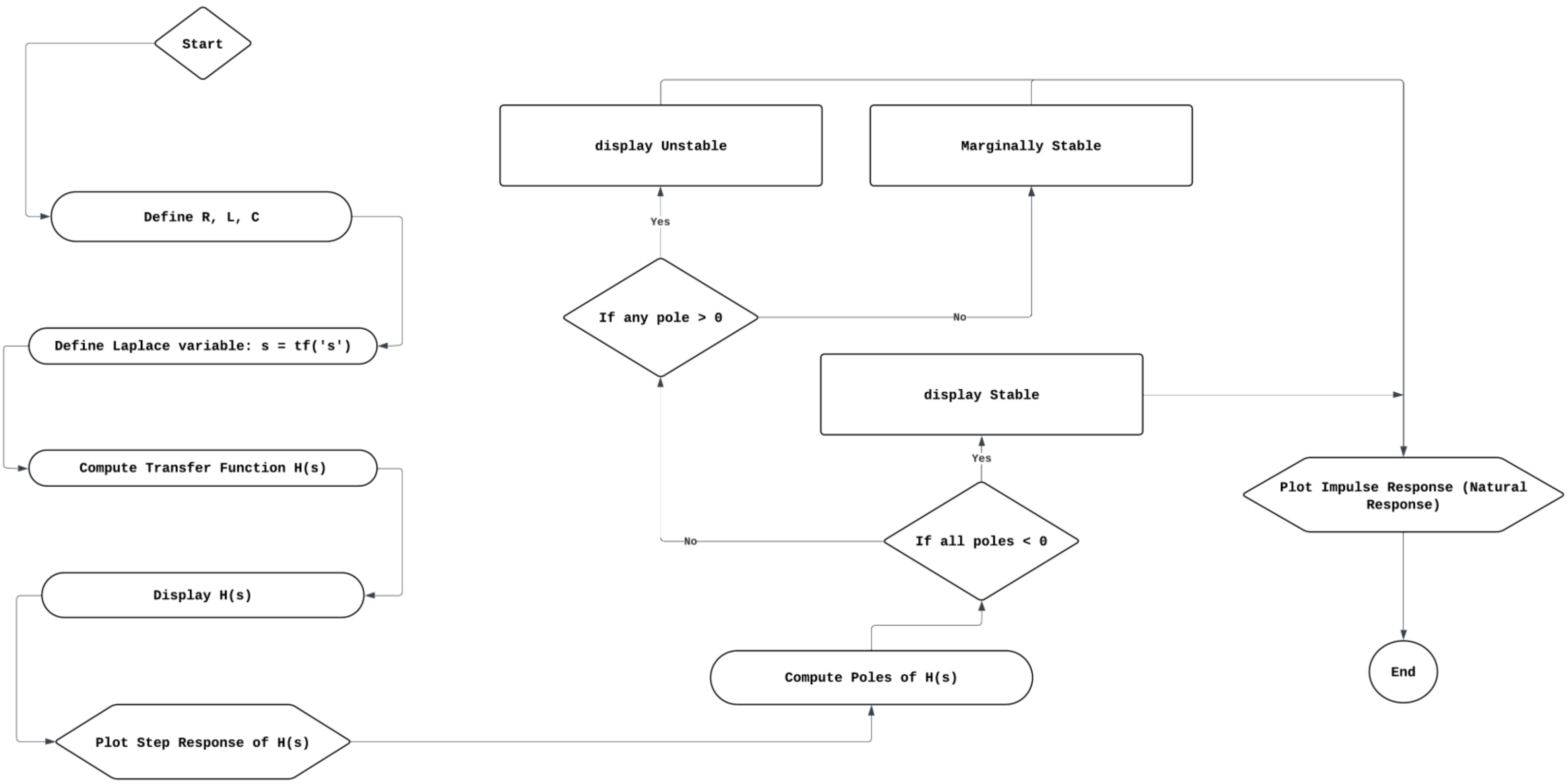
This is a second-order system, where the nature of the response depends on the Roots:

$$s = \frac{-R^2}{2L} \pm \sqrt{\left(\frac{R}{2L}\right)^2 - \frac{1}{LC}}$$

Depending on the Roots:

- Real Negative Poles (Roots): System is Stable.
- Complex Poles (Roots): System has oscillatory response.
- Real Positive Poles (Roots): System is Unstable.

Flow Chart:



Start
Define R, L, C
Define Laplace variable: s = tf('s')
Compute Transfer Function H(s)
Display H(s)
Plot Step Response of H(s)
Compute Poles of H(s)
If all poles < 0
Yes
display Stable
No
If any pole > 0
Yes
display Unstable
No
Marginally Stable
Plot Impulse Response (Natural Response)
End