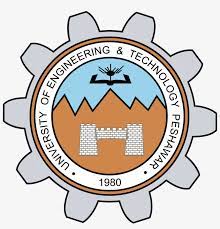
**“MODELING FREQUENCY DIVISION MULTIPLEXING/DE-MULTIPLEXING”**

**LAB # 09**



**FALL 2023**

**DCSE Digital Signals Processing Lab**

**Submitted By: Esha Rizwan**

**Registration No: 21PWCSE2010**

**Section: C**

**“On my honor, as a student of University of Engineering and Technology, I have neither given not received unauthorized assistance on this academic work.”**

**Submitted To:**

**Sir Yasir Saleem**

**January 8th, 2024**

**Department of Computer Systems Engineering**

**University of Engineering and Technology, Peshawar**

**Lab Objectives:**

* To understand and learn the concept of multiplexing.
* To understand the concept of de-multiplexing.
* To learn to use MATLAB for multiplexing and de-multiplexing.

**MULTIPLEXING:**

Multiplexing is a technique used in telecommunications and data transmission to combine multiple signals into a single signal that can be transmitted over a shared medium, such as a cable or a communication channel, and then separated back into individual signals at the receiving end.

Types of multiplexing are:

* Time Division Multiplexing (TDM)
* Frequency Division Multiplexing (FDM)
* Wavelength Division Multiplexing (WDM)
* Code Division Multiplexing (CDM)

Multiplexing significantly increases the efficiency of communication channels by allowing multiple signals to share the same medium, reducing costs and maximizing bandwidth utilization.

**DE-MULTIPLEXING:**

De-multiplexing is the process of separating a combined signal, which contains multiple individual signals that were multiplexed together, back into its original constituent signals at the receiving end.

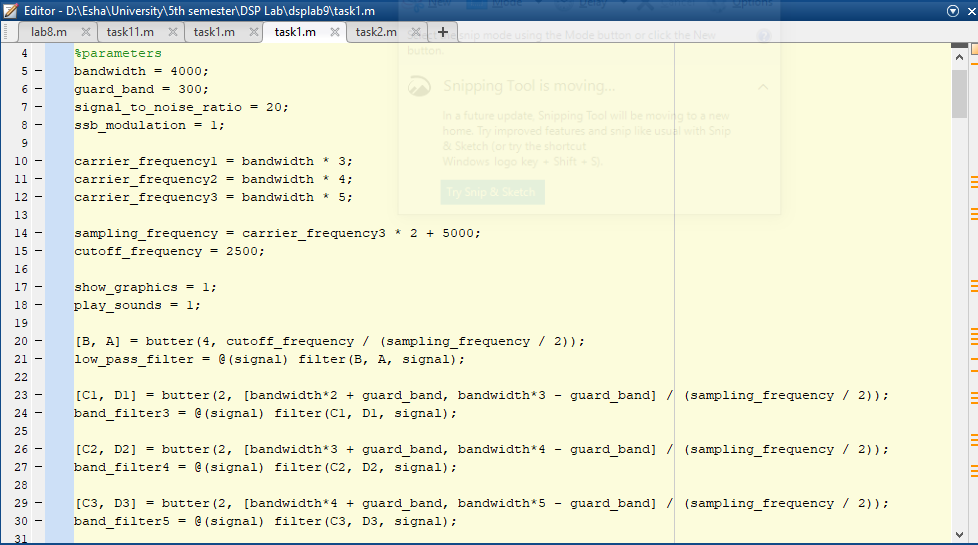
When multiple signals are multiplexed together for transmission over a shared medium, such as in techniques like Time Division Multiplexing (TDM), Frequency Division Multiplexing (FDM), or others, they are combined into a single signal for transmission. At the receiving end, the de-multiplexing process is employed to separate and extract each individual signal from the combined signal.

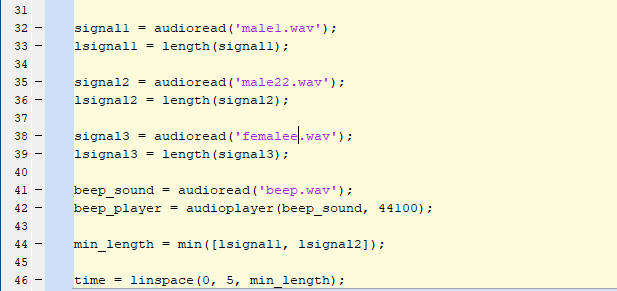
**LAB TASKS:**

Implement the following steps in MATLAB to multiplex three input voice signals at the transmitter end and de-multiplex and play them back at the Receiver end. Add random noise to the signal while propagating via the channel.

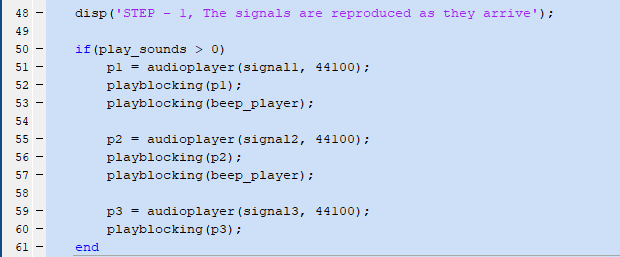
**SETTING UP THE PARAMETERS.**

**CODE:**





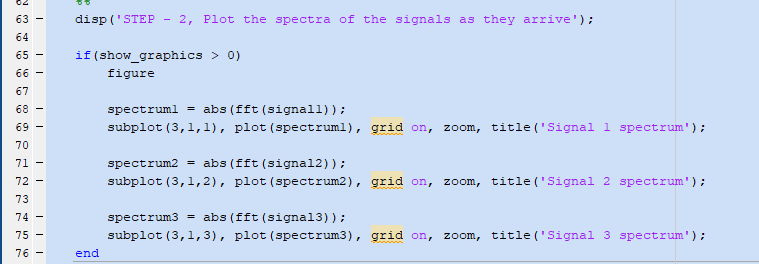
**THE SIGNALS ARE REPRODUCED AS THEY ARRIVE.**

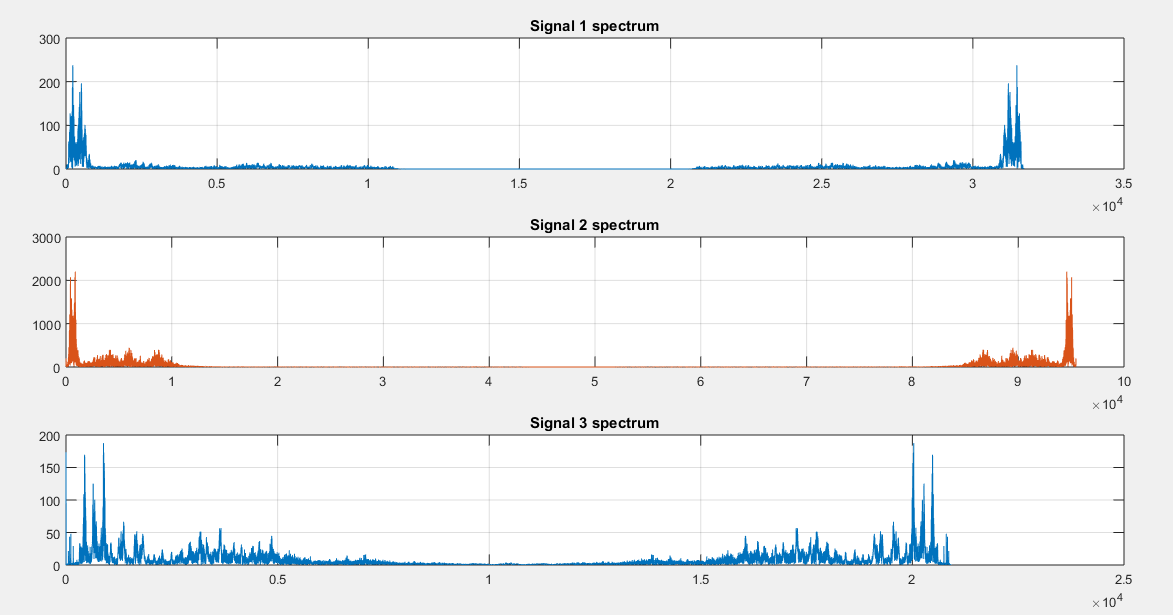
**CODE:**

**PLOT THE SPECTRA OF THE SIGNALS AS THEY ARRIVE (USE FFT AND**

**DSP.SPECTRUMANALYZER FOR COMPARISON).**

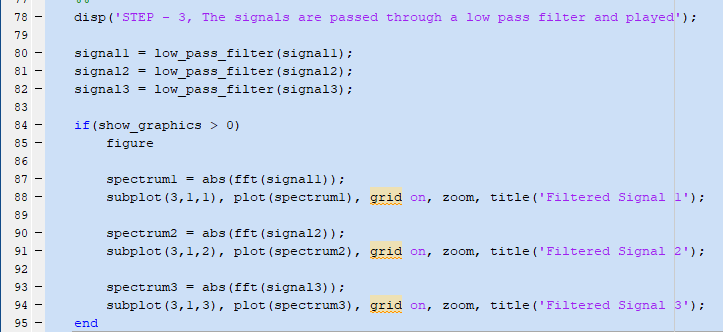
**CODE:**

  
**PLOT:**

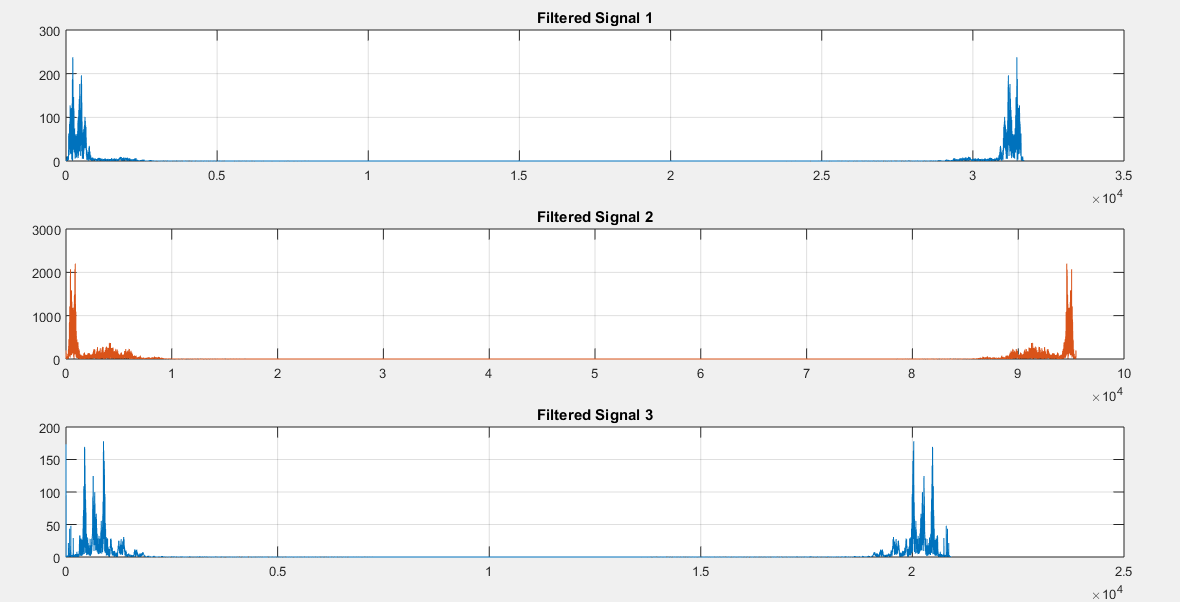


**THE SIGNALS ARE PASSED THROUGH A LOW PASS FILTER AND PLOTTED.**

**CODE:**

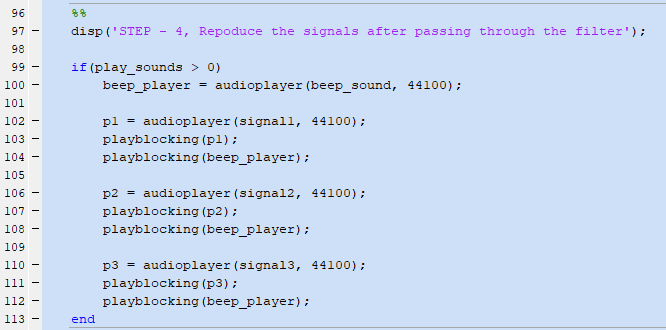


**PLOT:**



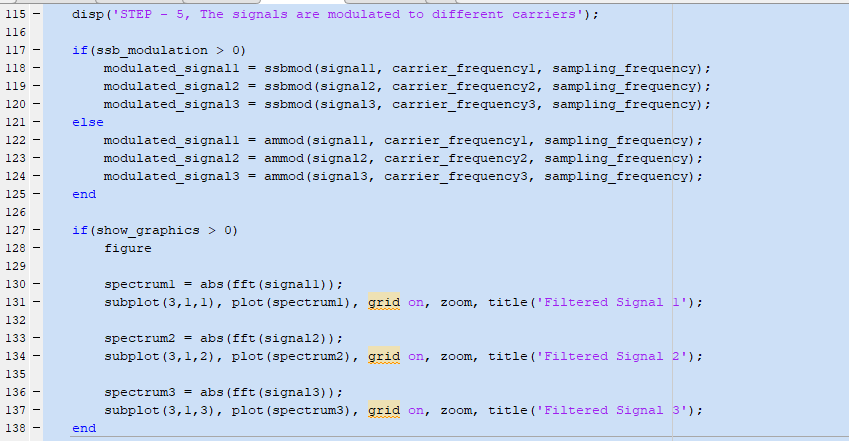
**REPRODUCE THE SIGNALS AFTER PASSING THEM THROUGH THE FILTER.**

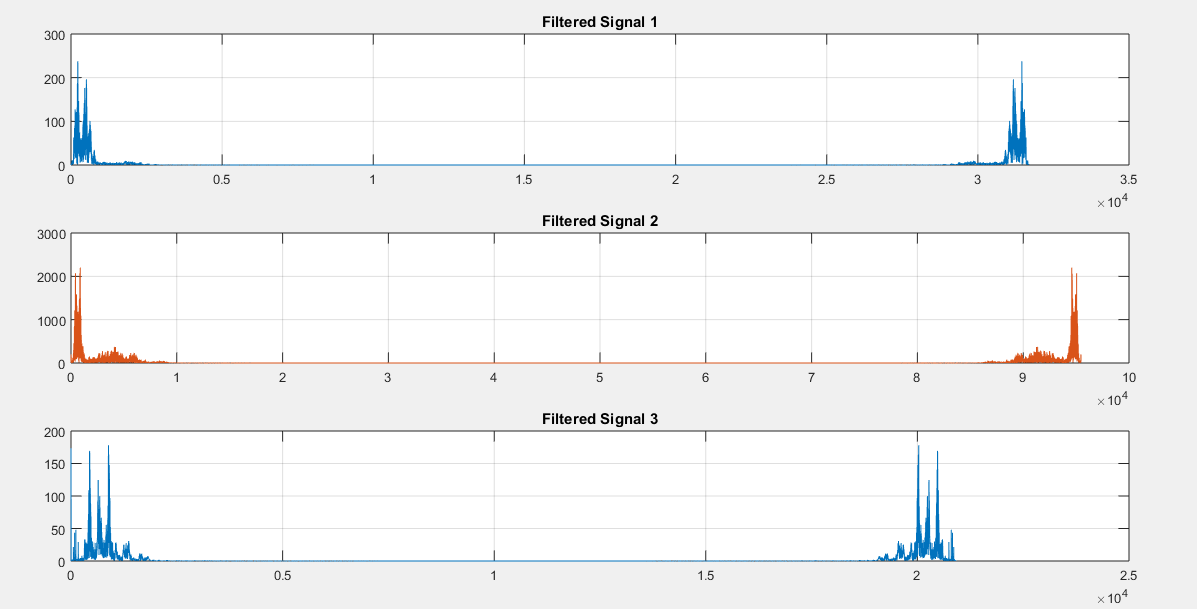
**CODE:**



**THE SIGNALS ARE MODULATED TO DIFFERENT CARRIERS.**

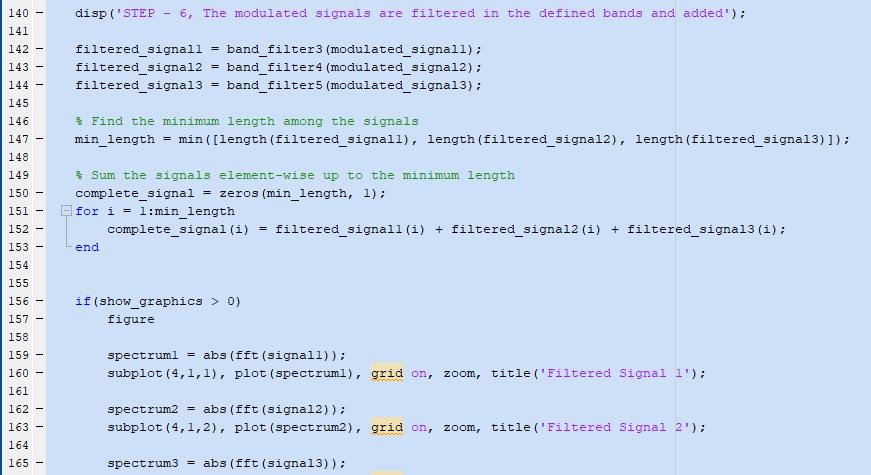
**CODE:**

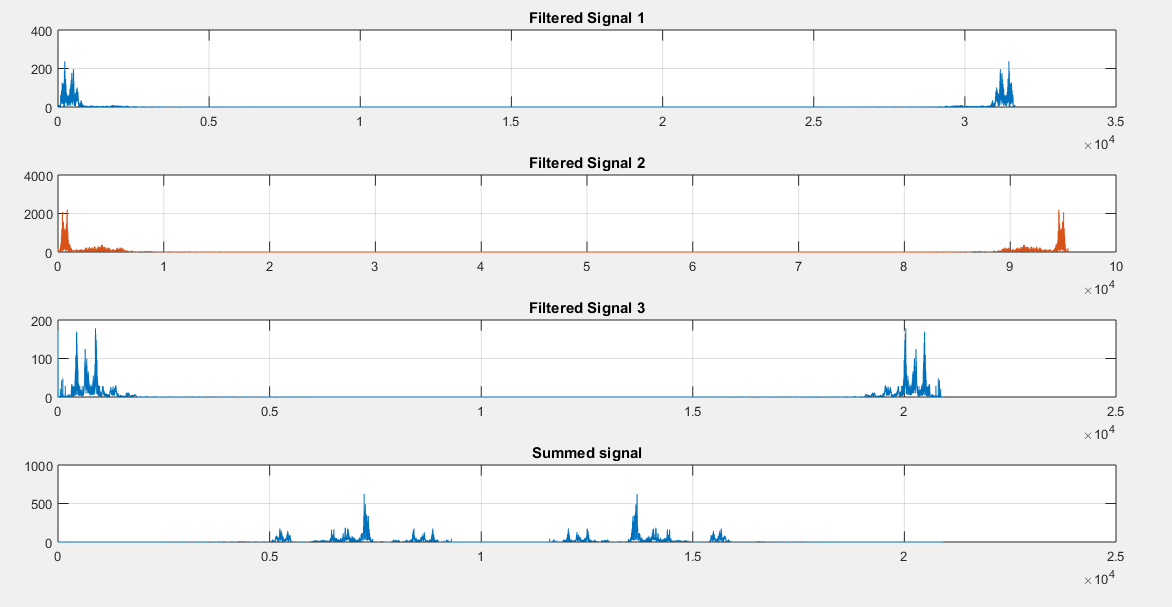
  
**PLOT:**



**THE MODULATED SIGNALS ARE FILTERED IN THE GIVEN BAND AND ADDED TOGETHER.**

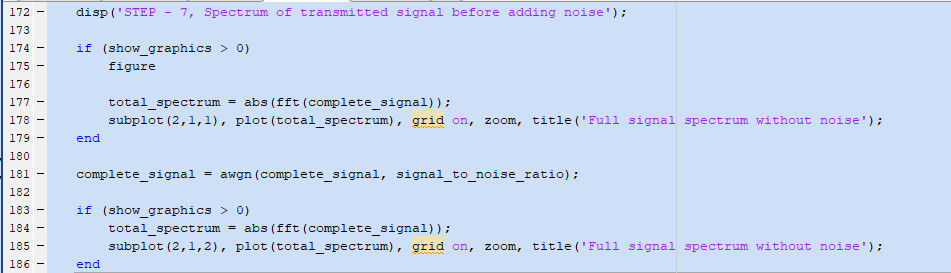
**CODE:**

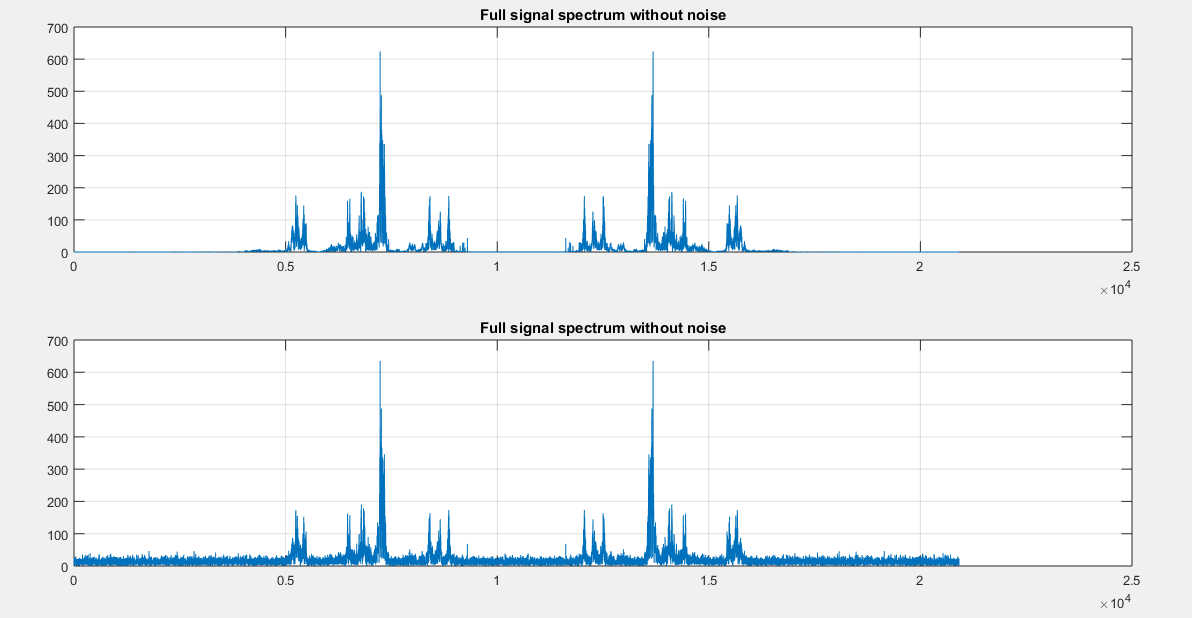
  
**PLOT:**



**SOME NOISE IS ADDED TO THE TRANSMITTED SIGNAL.**

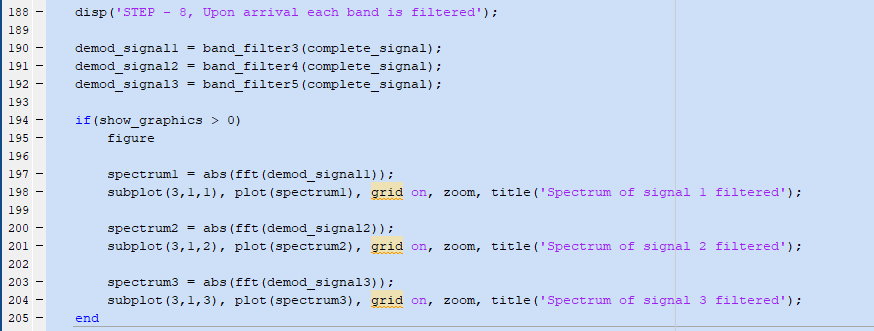
**CODE:**

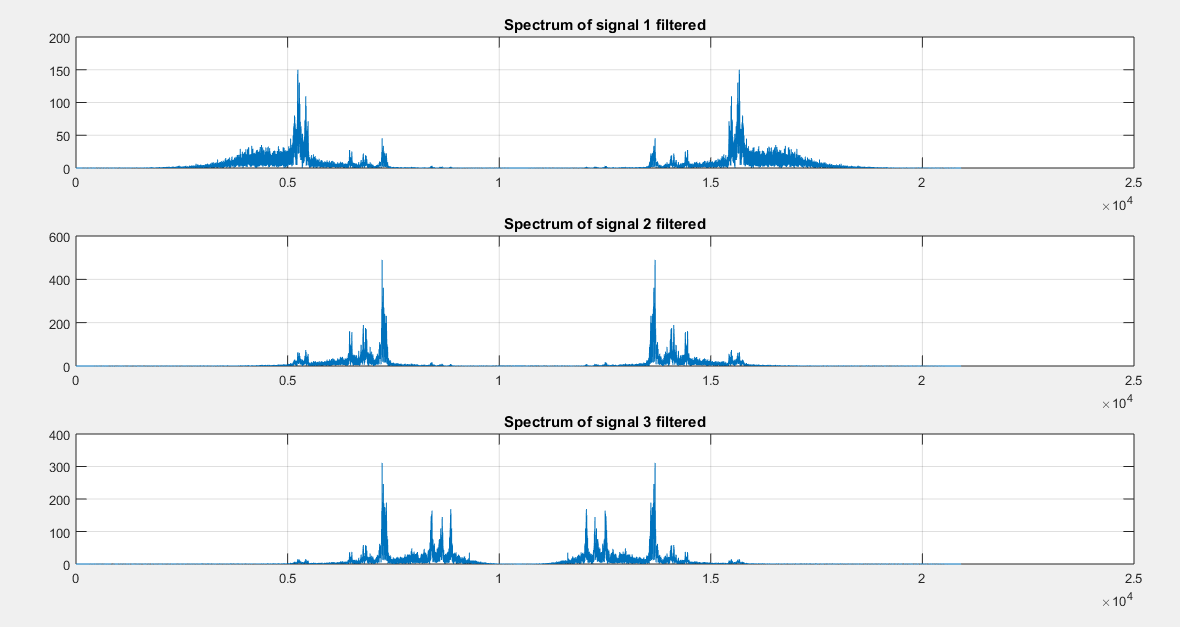
  
**PLOT:**



**UPON ARRIVAL EACH BAND IS FILTERED.**

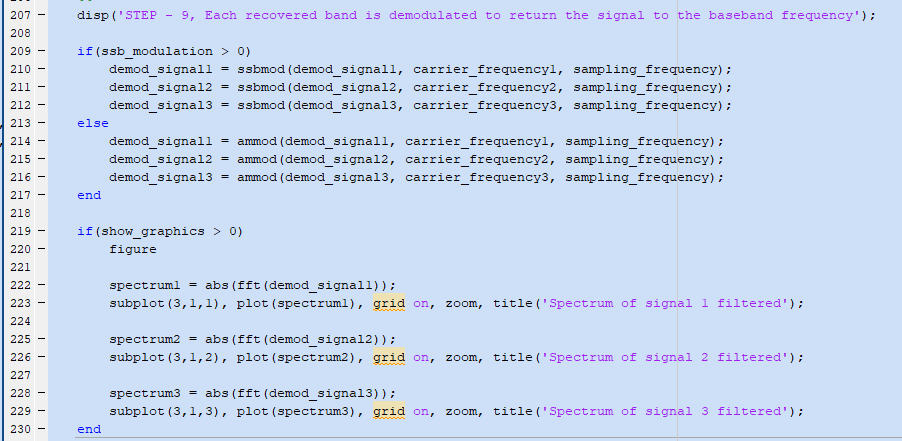
**CODE:**

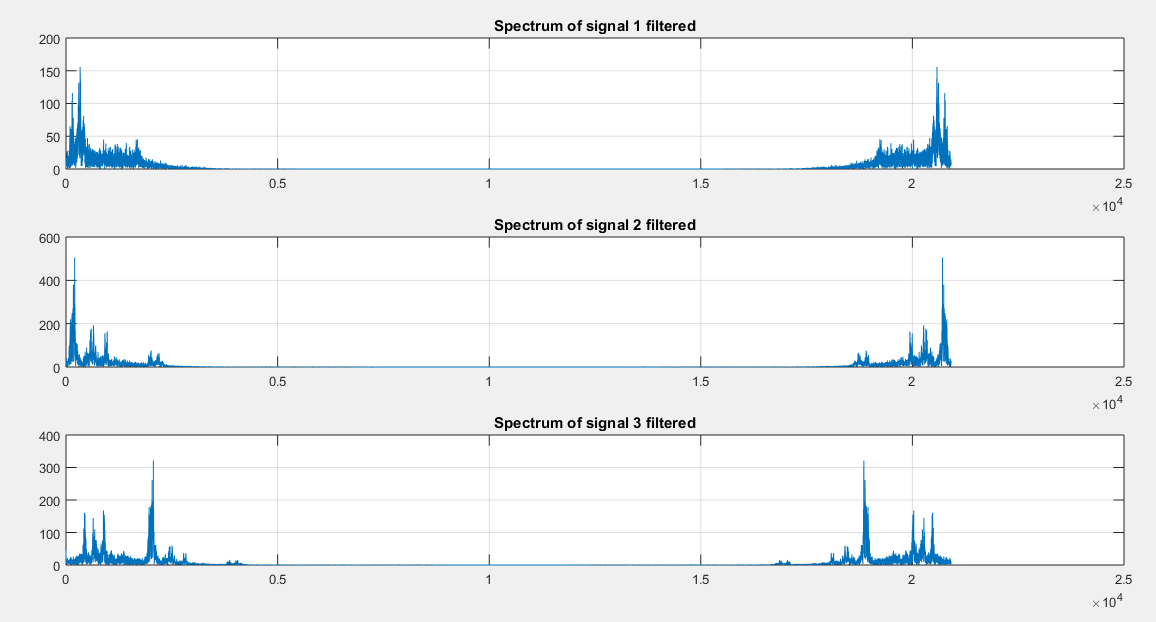
  
**PLOT:**



**EACH RECOVERED BAND IS DEMODULATED TO RETURN THE SIGNAL AT THE INDICATED FREQUENCY.**

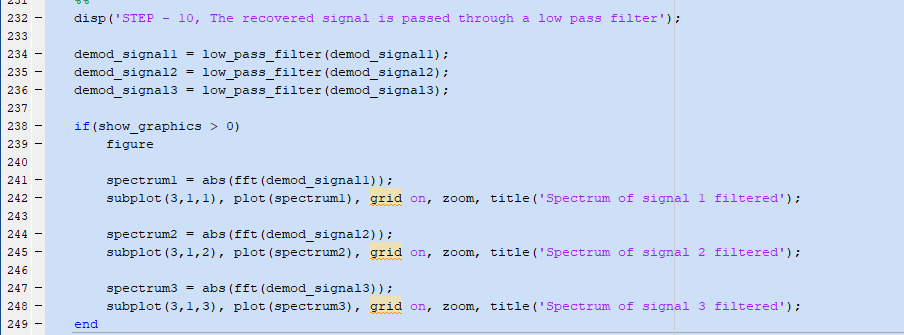
**CODE:**

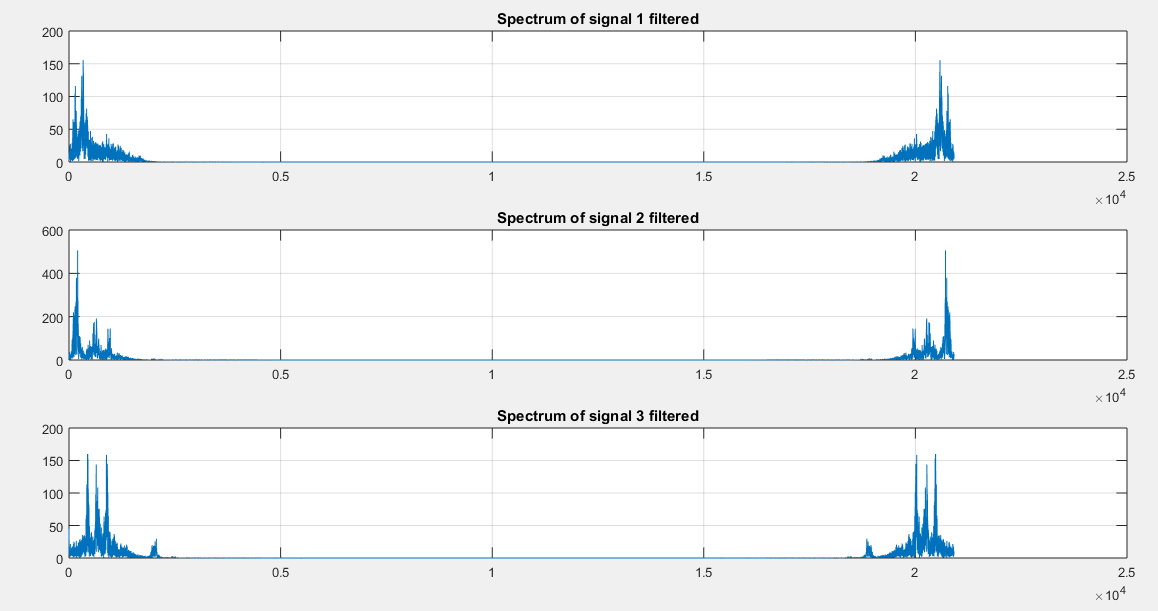
  
**PLOT:**



**THE RECOVERED SIGNAL IS PASSED THROUGH A LOW PASS FILTER.**

**CODE:**

  
**PLOT:**



**SIGNAL REPRODUCED AFTER TRANSMISSION.**

**CODE:**

