



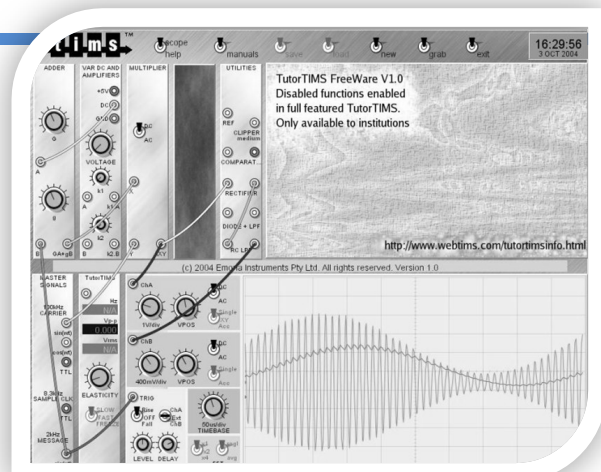
King Fahd University of Petroleum & Minerals
Electrical Engineering Department

EE370 Communications Engineering

LAB Manual

Dr. Maan A. Kousa & Dr. Ali H. Muqaibel

August 2010



Contents

INTRODUCTION TO “COMMUNICATION ENGINEERING I” LABORATORY	3
EXP 1: GETTING FAMILIAR WITH THE LABORATORY EQUIPMENT.....	7
EXP 2: SIMULATION OF COMMUNICATION SYSTEMS USING MATLAB	11
EXP 3: REPRESENTATION OF SIGNALS & SYSTEMS.....	15
EXP 4: SPEECH SIGNALS.....	19
EXP 5: DSBSC MODULATION & DEMODULATION	23
EXP 6: AM AND QAM	27
EXP 7: FM MODULATION.....	31
EXP 8: FM DEMODULATION	35
EXP 9: PCM ENCODING.....	39
EXP 10: PCM DECODING.....	43
EXP 11: LINE CODING	47
EXP 12: DIGITAL MODULATION: FSK.....	51
APPENDIX A: LABORATORY REGULATIONS AND SAFETY RULES	55
APPENDIX B: SAMPLE REPORT	56

Introduction to “Communication Engineering I” Laboratory

Purpose of “Communication Engineering I” Laboratory

The goals of the communication laboratory are:

1. to allow you to perform experiments that demonstrate the theory of signals and communication systems that are discussed in course,
2. to introduce you to some of the electronic blocks that make up communication systems (which may not be discussed in the lecture course because of time limitations), and
3. to familiarize you with proper laboratory procedure, including precise record-keeping, logical troubleshooting, safety, and learning about the capabilities and limitations of your equipment.

Introduction

This document contains the laboratory experiments to accompany the course EE 370 “Communications Engineering I”, offered by Electrical Engineering Department, KFUPM. The document contains twelve experiments, four on basic and general background, four on analog modulation, and four on digital modulation. The four basic experiments cover introduction to the laboratory equipment, simulation of communication systems using MATLAB, time- and frequency-domain representation of signals, and processing of speech signals. The analog modulation part covers the generation and detection of Double-Side Band Suppressed Carrier (DSBSC) modulation, Double-Side Band With Carrier (also known as AM) modulation, Quadrature Amplitude Modulation (QAM), and Frequency Modulation. The digital modulation experiments include PCM encoding and decoding, line codes and digital carrier modulation (ASK and FSK).

Each experiment, whenever applicable, contains the following sections:

Objectives: where the expected achievements by the end of the experiment are stated.

Introduction: where the theory of the subject is reviewed. The introduction is kept brief, assuming the student has covered the material in detail in class, or can refer to his textbook for further reading.

System Modules: where the main new modules to be used in the experiment are described.

Lab Work: leading the student on how to run the experiment. The lab work is organized in parts in order to have a clear and integrated structure of the work.

Post-Lab Work: extra questions and tasks for the student to carry after the lab, and include in the lab report.

General Laboratory Procedure

While there is no specific document to be submitted at the beginning of the Lab –unless your instructor advises you otherwise-, you are expected to read the experiment fully before you come to the laboratory. Interestingly, you can even try parts of the experiment at home. Here is a list of programs that will equip you with a virtual lab at your home:

Tool	Function	Link
TutorTIMS®	Virtual Lab (Modules,..etc)	http://www.webtims.com/
Picoscope®	Oscilloscope & Spectrum Analyzer	http://www.picotech.com/download.html
Matlab®	Simulation Tool	http://www.mathworks.com/

In addition to the experiment write up, a **Lab Sheet** has been prepared for every experiment. The Lab Sheet is a working document, designed to help students record all lab activities (measurements, observations, answers to questions in the lab manual, ...). The student must have his instructor sign the sheet before he leaves. The material in the sheet shall be utilized in writing the report. Plots from the PC-based oscilloscope and spectrum analyzer may be saved on a storage media (or student file-box if network is available) to reproduce them later in the report. The lab sheets for the 12 experiments are collected in one booklet separate from this document.

A set of *Laboratory Regulations and Safety Rules* are attached in *Appendix A*. All students have to observe them carefully.

MATLAB will be frequently invoked as part of the post-lab work, mainly in the form of designing a simulation counterpart for the experimental work. Such exercise will improve the student programming skills, and acquaint him with the most frequently-encountered functions and techniques for simulating communication systems. It is the sole responsibility of the student to learn the basics of MATLAB.

Every student should submit a report on each experiment. The report must be self-contained, and can be read independent from the lab manual. All axes in all graphs should be clearly labeled. If there is more than one trace in the plot, they should be clearly labeled. A sample report is attached in *Appendix B*.

Troubleshooting

Things will not always go as expected; this is the nature of the learning process. While testing a communication block, if the output signal is not what you expect, don't just try things at random, i.e replacing wires, rotating knobs, and toggling switches, hoping to get lucky. Rather, **think before you do anything**. If you do so you will avoid wasting time going down dead-end streets.

Be logical and systematic. First, look for obvious errors that are easy to fix. Is your measuring device correctly set and connected? Are you looking at the proper scale? Is the power supply set for the correct voltage? Is the signal generator correctly set and connected? And so on. Next, check for obvious misconnections or broken connections, at least in simple circuits.

As you work through your circuit, use your lab sheet to record tests and changes that you make as you go along; don't rely on your memory for what you have tried. Identify some test points in the system at which you know what the signal should be, and work your way backwards from the output through the test points until you find a good signal. Now you have a section of the system to focus your efforts on. Here is where a little thought about laying out your board before connecting it up will pay off; if your system looks like a jungle, it is going to be very hard to troubleshoot, but if it is well organized and if the wires are short, it is going to make your job a lot easier.

Final remark: if you do discover a bad module or wire, do not just throw it back in the box. Tell your instructor or the lab technician about it.

Neatness

When you have finished for the day, return all modules to their proper storage bins, return all test leads and probes to their storage racks, return all equipment to its correct location, and clean up the lab station. If appropriate switch off the unneeded equipments.

We hope you an enjoyable learning experience!

This page is intentionally blank. All Experiments start with odd pages for double-sided printing

Exp 1: Getting Familiar with the Laboratory Equipment

Objectives

- Learn the various components and conventions of the lab equipment from TIMS.
- Use the data sheets to learn about the operation, parameters and limitations of system modules.
- Explore the features and capabilities of the PC-based oscilloscope and spectrum analyzer.
- Perform basic modeling using TIMS.

TIMS Overview

Throughout the course, we will be using the laboratory equipment 301C PC-based from TIMS® to complement and demonstrate the theoretical part of the course. We will devote this experiment to introduce the equipment and get familiar with its usages.

TIMS is a telecommunications modeling system that models block diagrams representing telecommunications systems. Physically, TIMS is a dual rack system; the upper rack accepts up to 12 plug-in cards, or modules; the lower rack houses a number of fixed modules, as well as the system power supply.



Figure 1: TIMS 301-C System Unit

The modules are simple electronic circuits, which serve as basic communications building blocks. Each module, fixed or plug-in, has a specific function; functions fall into three categories:

1. Signal Generation - oscillators, variable DC, etc
2. Signal Processing - multipliers, filters, etc
3. Signal Measurement - frequency counter, PC-based instrument inputs.

Some of those modules are classified as basic modules while others are advanced modules. The fixed modules are all basic. They include: BUFFER AMPLIFIERS, FREQUENCY AND EVENT COUNTER, HEADPHONE AMPLIFIER, MASTER SIGNALS, TRUNK PANEL, VARIABLE DV and PC-BASED INSTRUMENT INPUT. The list of available plug-in modules is shown in the table below.

#	Module	Type	#	Module	Type
1	AUDIO OSCILLATOR	Basic	12	VOLTAGE CONTROL OSCILLATOR 1	Basic
2	ADDER	Basic	13	VOLTAGE CONTROL OSCILLATOR 2	Basic
3	DUAL ANALOG SWITCH	Basic	14	60KHz LOW PASS FILTER	Basic
4	MULTIPLIER	Basic	15	QUADRATURE UTILITIES	Advanced
5	PHASE SHIFTER	Basic	16	LINE CODE ENCODER	Advanced
6	QUADRATURE PHASE SHIFTER	Basic	17	LINE CODE DECODER	Advanced
7	SEQUENCE GENERATOR	Basic	18	100KHz CHANNEL FILTER	Advanced
8	UTILITIES	Basic	19	PCM ENCODER	Advanced
9	TUNEABLE LOW PASS FILTER 1	Basic	20	PCM DECODER	Advanced
10	TUNEABLE LOW PASS FILTER 2	Basic	21	BIT CLOCK GENERATOR	Advanced
11	TWIN PULSE GENERATOR	Basic	22	SPEECH MODULE	Advanced

A data sheet for each module describing its input(s), output(s), configurable parameters and function can be found in the *User Manuals (Basic and Advanced)* available in the lab bench drawers. A soft copy is also available on all laboratory computers' desktop.

All TIMS modules conform to the following conventions:

- Signal interconnections are made via front panel sockets
- Sockets on the left hand side are for module inputs.
- Sockets on the right hand side are for module outputs.
- Yellow sockets are for analog signals.
- Red sockets are for digital signals.
- Analog signals are held near the level of 4V p-p.
- Digital signals are TTL level, 0 to 5 V.
- The green socket is the system Ground.
- Any plug-in module may be placed in any of the 12 positions of the upper rack.
- All modules use the back plane bus to obtain power supply.
- The modules can be plugged-in or removed without turning off the power.

It is important to note that:

- The plug-in modules are not firmly locked in the rack, and need to be held in position while interconnecting leads are removed.
- When removing the leads, hold them from their solid heads and DO NOT PULL them from the flexible segment, in order not to damage the wires.
- There are 22 plug-in modules. Make sure you leave them in sequence in the storage shelves.



Oscilloscope and Spectrum Analyzer

TIMS is equipped with a fixed module, PC-BASED INSTRUMENT INPUTS¹, that provides interface with display devices, namely oscilloscope and spectrum analyzer. Either one can be physical stand-alone equipment or soft PC based. The connection to physical display devices is provided by coaxial cords, whereas the connection to the soft devices is provided through USB connection (already connected from the back panel). The application that runs the soft oscilloscope and spectrum analyzer in our lab is called picoscope®, and can be started from the shortcut on the PC.

The DISPLAY INTERFACE module can take up to 4 signals on channels A1, A2, B1 and B2, but allows 2 of them (one from A and one from B) to be viewed simultaneously. The channels can be selected by means of two mechanical switches on the front panel of the module.

If the displayed signal seems to be sliding left and right or changing too fast, then the oscilloscope has to be *triggered*. Triggering is some form of synchronization that provides a reference point for a periodic waveform. Without triggering, each sweep starts from a different instant of the period, resulting in unstable display. It is important to consider which of the many signals present will be used to trigger the oscilloscope. Use a periodic signal with the longest period from among the displayed signals, or use an external signal if needed. External triggering is connected to Channel-E of the DISPLAY INTERAFCE module.

You have been exposed to the oscilloscope before, but the spectrum analyzer may be new to you. The spectrum analyzer is a device that displays the frequency composition of the signal. The horizontal axis represents the frequency whereas the vertical axis represents the magnitude. Because of the large variation of the magnitude spectrum, the vertical axis is usually set to dB scale. Note that $X_{dB} = 20 \log(X)$. For example, if A_{dB} is 20 dB below B_{dB} , then $B/A = 100$. The decibel symbol is often qualified with a suffix that indicates which reference quantity has been used. For example, dBm indicates that the reference quantity (0 dBm) is one milliwatt, while dBu indicates that the reference quantity (0 dBu) is one microwatt.

When observing the signal spectrum on the spectrum analyzer, you will notice a lot of “noise” all over the frequency axis. This is due to the circuit components. However, the noise level is extremely low, in the range of -60 dB or even less, compared to the signal level (i.e. one millionth of the signal level); it can therefore be neglected.

You have many options to plot the results you see on the picoscope. One option is to save the data in *.mat or *.csv. In this case you can import the data to MATLAB or MS Excel and reproduce the plot. You may, alternatively, save the plot directly as *.gif.

You can download a fully functioning demo version of PICOSCOPE (PICOSCOPE 3204) from the following site:

<http://www.picotech.html/software.html>

In this experiment, we will introduce the fixed modules in addition to the ADDER plug-in module.

¹ The name of this module is not intuitive. We will instead refer to it as DISPLAY INTERFACE module.

Lab Work

1. Read the data sheet of the ADDER in the TIMS Manuals-Basic Modules. Which of the following equations can be implemented using the ADDER and which cannot? Write your answers in the Lab Sheet.
 - 2 cos(2π 2x106t) - 1.5 cos(2π 2x105t);
 - 1.3 cos(2π 2x104t)x(t) – 0.5 sin(2π 2x103t);
 - 2.5 cos(2π 2x104t)x(t) – 10.5 sin(2π 2x103t);
 - 1.3 cos(2π 2x104t)x(t) + 0.5 sin(2π 2x103t).
2. Use the FREQUENCY COUNTER module to verify the frequencies of the following four signals from the MASTER SIGNALS: 100 kHz sine, 8.3 kHz Clock, 2 kHz TTL and 2 kHz sinusoid. Note down the values.



Warning: The FREQUENCY COUNETR module accepts TTL and analog inputs. ONLY ONE OF THEM SHOULD BE CONNECTED AT A TIME, otherwise you may get erroneous measurement.

3. Connect the previous four signals of the MASTER SIGNALS module to the four inputs of the DISPLAY INTERFACE. Use the switches to display them on the oscilloscope (picoSCOPE). Measure the amplitude of each signal and note them down in the Lab Sheet.
4. Use the VARIABLE DC, BUFFER AMPLIFIERS and ADDER modules to generate the signal $3\cos(2\pi \times 2 \times 10^3 t) + 6$ V. Draw the modules and show the connections. Let your instructor verify the waveform.
5. Observe and plot the spectra of each of the four signals of the MASTER SIGNALS module.
 - a. Do the spectra plots coincide with your expectations? Explain.
 - b. How far is the noise level below the signal level?
6. Using a 2 kHz sinusoidal signal on one channel and 8.33 kHz digital signal on the other channel, familiarize yourself with the picoSCOPE by exploring the following features.

Feature
Switch between oscilloscope and frequency analyzer on the same view
Display one or both channels on the same view (window)
Separate the two channels on the same view so that they are non-overlapping (do it manually and auto)
Change the setting of the axes.
Take a snap shot or continuous scan
Zoom in a specific segment of the graph
Display measurements of DC value, frequency, period, ...
Use horizontal and vertical markers
Set the oscilloscope on external triggering
Create time view and spectrum view and save them

Exp 2: Simulation of Communication Systems Using MATLAB

Objectives:

The main objective of this session is to learn the basic tools and concepts for simulating communication systems using MATLAB.

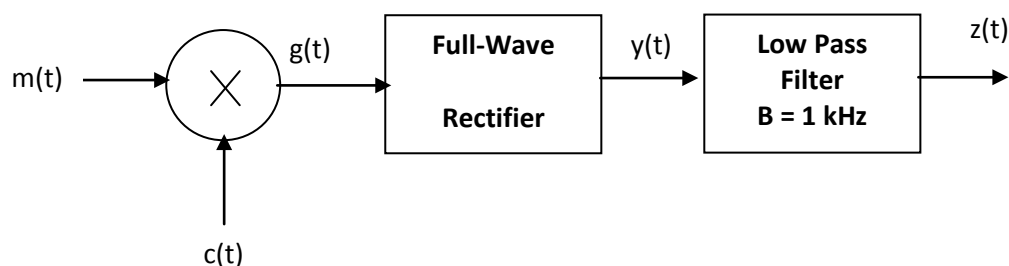
Introduction

MATLAB is a user-friendly, widely used software for numerical computations. MATLAB is vector-oriented, that is, it mainly deals with vectors (or matrices). It is assumed that you have used MATLAB before, and you can do simple operations, as well as create and run *.m files. Some useful tutorials can be found on the EE 370 course/lab website. If you need help on how to start working on MATLAB, we advise you to read [Matlab Primer](#) available in the internet.

Our focus in this session will be on using MATLAB for simulating communication systems. Instead of going in the traditional approach of explaining items individually, we will work through one complete example, and introduce the application as we go.

Case Study:

Write a MATLAB program to simulate the following system



where $m(t) = \exp(-100|t|)$; $c(t) = \cos(2\pi 10^3 t)$

m-File:

% Define the time interval

```
ts=0.00001;
t= -0.1:ts:0.1;
```

% Define the functions m(t) and c(t)

```
m=exp(-100*abs(t));
c=cos(2*pi*1000*t);
```

```
% Performe the multiplication
```

```
g=m.*c;
```

```
% Perform full-wave rectification
```

```
y=abs(g);
```

```
% Create the filter
```

```
cutoff=1000;
[a b]=butter(5,2*cutoff*ts);
```

```
% Get the output after the filter;
```

```
z=filter(a,b,y);
```

```
% Plot the input and output on the same graph
```

```
figure (1)
plot(t,m,t,z);
legend('Input Signal','Output Signal')
xlabel ('time')
ylabel('amplitude')
title ('Case Study')
```

```
% Finding the FT of the signals
```

```
M=abs(fftshift(fft(m)));
G=abs(fftshift(fft(g)));
Y=abs(fftshift(fft(y)));
Z=abs(fftshift(fft(z)));
```

```
% Creating the vector for the frequency axis
```

```
f=[-length(t)/2:length(t)/2-1]/(length(t)*ts);
```

```
% Plotting all FT on one sheet, in a 2x2 matrix format
```

```
figure (2)
subplot (221)
plot(f,M)
subplot(222)
plot(f,G)
subplot (223)
plot(f,Y)
subplot(224)
plot(f,Z)
```

Discussion

```
% Define the time interval
```

This is usually the first step in any simulation. There are three parameters to define: the beginning of the interval, the step size, the end of the interval. The beginning and end of the

interval are intuitive; for periodic signals you want to cover 3-5 periods; for non-periodic signals, you usually want to cover the non-zero part of the signal.

The selection of the step size is crucial for the accuracy of the simulation. You need enough sample points to represent the signal. Usually, the step size is taken to be of the order of one hundredth of the smallest period in the program (Or, the sampling frequency $f_s = 1/t_s$ should be 100 times the frequency of the signal). In our example, since we are having $c(t)$ of frequency 1000 Hz, we selected $f_s = 100000$, or $t_s = 0.00001$.

% Define the functions $m(t)$ and $c(t)$

This is a straightforward step. The function **abs** stands for $| \cdot |$, while $\pi = \pi$. Note that the signals **m** and **c** are now vectors of the same size as **t**.

% Perform the multiplication

This is also a straightforward step. However note the dot after **m**. Why this is necessary here? What would happen if you remove the dot?

% Perform full-wave rectification

This is again a straightforward step, provided you recognize that full-wave rectification is mathematically equivalent to taking the absolute value.

% Create the LPF

This operation is frequently encountered in simulating communication systems. A LPF is defined by one parameter, the cutoff frequency.

A filter in MATLAB is represented by its transfer function. The transfer function is in general in the form of the division of two polynomials. The filter is completely defined by the coefficients of the polynomial at the numerator and the polynomial at the denominator. These are the vectors **a** and **b** respectively in the program.

There are many realizations for designing filters. One common realization is Butterworth, which is the one used here, hence the function name **butter**.

The **butter** function has two arguments. The first argument is the order of the filter. The larger the order the sharper the filter (closer to ideal), but more processing is required. For most of our applications an order of 3-5 should be sufficient.

The second argument is a coefficient related to the cutoff frequency. Without going into the details of the derivation, to design a LPF filter of cutoff frequency W , the argument should be set to $2*W*t_s$, where t_s is the time step size of the program. For more details about the command **butter**, type:

```
>> help butter ; in the MATLAB prompt
```

How many arguments would a BPF require? What are they?

% Get the output after the filter;

In the previous step we have only created the filter. To apply the filter to a given signal, we use the function **filter**. This function has three parameters: the coefficients of the filter **a** and **b**, and the vector to be filtered. Note that although we think of the filter operation in frequency domain, the **filter** function operates on a time-domain vector. The output should as well be taken as a time-domain vector.

% Finding the FT of the signals

The Fourier Transform of signals can be found in MATLAB using the function **fft**. It can be used with a single argument, which is the time-domain vector. The **fft** function yields only the positive side of the spectrum. To get the double-sided spectrum, augment **fft** by **fftshift**. Finally, if you are only interested in the amplitude spectrum, augment all by the function **abs**. The resulting frequency-domain vector will have size one less than the size of the input time-domain vector.

% Creating the vector for the frequency axis

To plot the frequency spectrum as a function of frequency, you need to create the frequency axis. The available range of frequencies depends on t_s , and is given by the relation:

```
f=[-length(t)/2:length(t)/2-1]/(length(t)*ts);
```

% Plotting

We leave this step to the student to explore. Use the help command to read about **plot subplot, figure, legend, xlabel, ylabel, title and axis** commands

Lab Work

1. Create and run the m-file above, and produce Figure (1) and (2).
2. Change $m(t)$ to $2 + \sin(2\pi 1000t)$ and $c(t)$ to $\cos(2\pi 10^4 t)$ and the cutoff frequency of the filter to 2 kHz. Redo part 1.

Post-Lab Work

1. Include the m-file and the figures for the work you did in the lab in your report.
2. Using MATLAB, add the signals $m(t) = \exp(-100|t|)$ and $c(t) = \cos(2\pi 10^3 t)$ then separate them by means of filtering only (LPF and BPF). Provide the m-file and a plot of the sum in time and frequency, and of each of the recovered signals in time and frequency.

Exp 3: Representation of Signals & Systems

Objective:

By the end of this experiment, the student should be able to:

- verify experimentally the relation between frequency and time domain representation of signals.
- observe some of none idealities related to noise floor and harmonics.
- measure the transfer function of a given system (filter) using narrow pulses.

Introduction

A signal is a function that symbolizes a physical variable of interest. Signals can be represented in time or frequency domains (Remember this is only representation). The two representations are related by Fourier Transformation. In this experiment we are going to examine some Fourier Transform properties, namely:

Property	Time	Frequency
Fourier transform of sinusoids	$\cos \omega_0 t$	$\pi[\delta(\omega - \omega_0) + \delta(\omega + \omega_0)]$
Linearity	$a_1 g_1(t) + a_2 g_2(t)$	$a_1 G_1(\omega) + a_2 G_2(\omega)$
Modulation	$g(t)\cos(\omega_0 t)$	
Time Scaling	$g(at)$	

Fill in the missing blocks in the table (See Lab Sheet).

A system, on the other hand, is a combination and interconnection of several components to perform a desired task. Systems can be characterized by their impulse responses in time domain, or transfer functions in frequency domain. For a subclass of systems, the linear systems, the impulse response (or transfer function) provides a very convenient and straightforward relation between the input and out of the system.

One type of system that is frequently-encountered in communications is the filter. A filter is a frequency-selective device that allows a certain frequency band to pass (with high gain), and blocks other bands. Depending on which band it passes, a filter can be classified as low pass (LPF), band pass (BPF) or high pass (HPF).

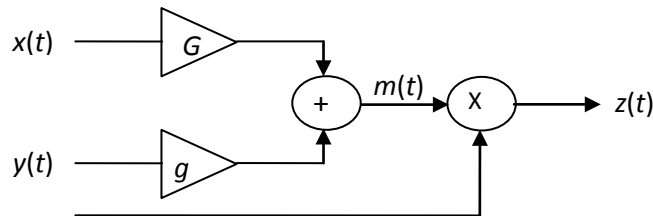
Lab Work

There are three parts in this experiment. In part I, we verify some of Fourier transform properties. In the second part, we study the effect of filtering on periodic signals. The last part is devoted to identify unknown systems by measuring their impulse response and transfer function.

To conduct the experiment, the following modules are needed: TUNABLE LPF, AUDIO OSCILLATOR, TWIN PULSE GENERATOR, 100 kHz CHANNEL FILTER.

Part I: Verification of Fourier Transform Properties

1. Select and connect the proper modules to implement the following block diagram:

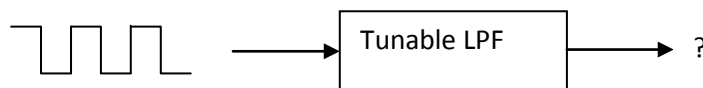


Draw the equivalent Modules and show their interconnection.

2. Using the frequency counter, set the AUDIO OSCILLATOR module to produce a 5-kHz sinusoidal signal and connect it to the system as $x(t)$.
3. Set $y(t)$ as a sinusoidal signal of frequency 2 kHz from the MASTER SIGNALS module.
4. Set f in the above block to 100 kHz.
5. Set g to zero (full counter clockwise) and G to maximum.
6. Obtain the plot of $m(t)$ from both the spectrum analyzer and the scope and compare with your theoretical expectations. *Comment on the noise level and harmonics.*
7. Vary the frequency of $x(t)$ and observe the impact on both frequency and time domain. Describe what you observe in light of the time scaling property.
8. Re-adjust the frequency of $x(t)$ to 5 kHz and increase g gradually. Observe the change in $m(t)$ on the spectrum analyzer and the oscilloscope. When g is maximum, obtain plots of $m(t)$ waveform and spectrum. *What is the property we are trying to prove?*
9. Plot the waveform and the spectrum of $z(t)$.
10. Zoom the spectrum of $z(t)$ around 100 kHz and observe its contents.
11. Compare the spectrum of $m(t)$ and $z(t)$ and comment on the modulation property.

Part II: Filtering of Periodic Signals

In this part we verify the Fourier Series representation of periodic signals, and examine the effect of filtering on the signal's shape and spectrum.



1. Apply a square wave signal with frequency of 2 kHz to the TUNABLE LPF module. Set the TUNE and GAIN knobs on the module to maximum (full clockwise), and set the toggle switch to WIDE. Observe the input and the out of the filter in both time and frequency domain. Are they similar? Why?

2. Turn the TUNE knob to minimum (full counter clockwise). Observing the output, gradually increase the cutoff frequency to allow one harmonic, then two harmonics, then three, and so on.
3. Obtain time and frequency plots for three cases (one harmonic, two harmonics, max filter bandwidth). Adjust the axes to zoom-in the important data and get clear plots.
4. Explain the effect of the filter cutoff frequency on the output waveform.

Part III: System Identification

Systems, in general, are characterized by their impulse responses or transfer functions. An impulse is a non-realizable function. However, it can be approximated from a train of square pulses by making the pulse width as narrow as possible and the period as large as possible. This technique will be used to characterize different filters.

1. Use a TWIN PULSE GENERATOR module and clock it at 2 kHz using the AUDIO OSCILLATOR module (TTL output). Observe the output from **Q1** on the scope and adjust the pulse width to minimum.
2. Connect the pulse train, **Q1**, to the input of the TUNABLE LPF.
3. Adjust the cutoff frequency of the TUNABLE LPF to a mid value. Plot the impulse response and the spectrum of the signal at the output of TUNABLE LPF.
4. Vary the cutoff frequency of the TUNABLE FILTER and verify that (the envelope of) the output spectrum approximates the transfer function of the filter. Get your instructor approval.
Why the spectrum consists of spectral lines and not a continuous curve? What controls the spacing between the spectral lines? Test your hypothesis.
5. Use the above approximate transfer function measurement method to discover the filter characteristics of the 100 kHz CHANNEL FILTER module. Obtain the transfer function for the three settings (1, 2, 3). What is the type of the filter in each case? Estimate the 3dB bandwidth of the filters. Use the horizontal markers of the oscilloscope to trap the 3 dB drop, then the vertical markers to measure the bandwidth.

Post-Lab Work

Use MATLAB to plot and compare the transfer function of

- (1) Butterworth LPF of cutoff frequency 1 kHz, and order 1, 3, 5.
- (2) Butterworth BPF of cutoff frequencies 5 kHz and 8 kHz, order 1, 3, 5.

You can generate an impulse using the function `rectpulse(t,W)`. You can make W as narrow as time step size ts in order to get an excellent approximation of an impulse.

This page is intentionally blank. All Experiments start with odd pages for double-sided printing

Exp 4: Speech Signals

Objectives

- Understand the features and characteristics of speech signals.
- Get acquainted with the SPEECH module from TIMS.
- Perform simple processes on speech signals (filtering, frequency translation), and examine their effect on the sound.

Introduction

Speech is the most frequently encountered message in communication systems. Throughout the lab work we will use a real speech message, whenever appropriate. In order to be prepared, we devote this experiment to study the basic characteristics of speech signals and get acquainted with the SPEECH module of TIMS.

We generate speech, or voice in general, by virtue of the vibration of our vocal cords. The sounds we produce are composed of many harmonics, or pitches. Typically, the significant part of human voice occupies the range from 300 Hz – 3 kHz. This can be seen from the spectrum of the voice signal. The low-end of the spectrum represents the low-pitch sounds, while the high-end of the spectrum represents the high or sharp pitches.

We can hear sounds over a much wider frequency range than the ones we produce. These sounds are called audible signals. A healthy human being can hear frequencies up to 15-20 kHz. This is another proof that we are created to hear more than we talk!

The following plug-in modules will be needed for this experiment: SPEECH, TUNABLE LPF, MULTIPLIER, in addition to external signal generator.

The SPEECH MODULE

The SPEECH module allows speech and audio signals to be recorded and replayed. Three independent channels are provided: CHANNEL 1, CHANNEL 2 and LIVE. The module includes a built-in microphone. An EXTERNAL input is also provided for recording externally generated signals. The module front panel looks like that in Figure 1.

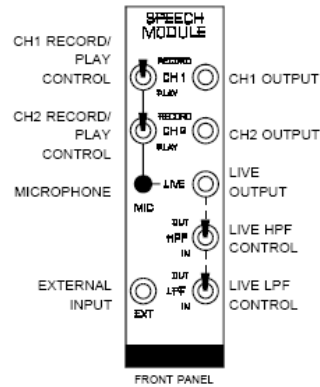


Figure 1: Speech Module

Channels 1 and 2 can each record up to 32 seconds of speech from the common **MIC**rophone input. To record speech or other sounds on either channel, set the front panel switch to **RECORD** and speak clearly into the microphone. The length of your message may vary from a few seconds up to 32 seconds. As soon as you have finished your message, set the switch to the **PLAY** position. The recorded content will automatically repeat upon switching to **PLAY**. Note that the length of the recorded message will only be the length of time the switch was in the **RECORD** position. A third non-recordable channel, **LIVE**, is also provided where the sound at the **MIC**rophone is continuously output as an electrical signal.

A pair of headphones is provided to allow the user to listen to the recorded messages by patching any one of the **SPEECH** module's outputs to the **HEADPHONE AMPLIFIER** in the **TIMS** System Unit.



WARNING: DO not put the headphone on if you are not having sound yet. A sudden high-volume sound may harm your inner ear.

Lab Work

This experiment consists of four parts. We start by measuring the audible range of our hearing system! Then, in part II, we record few different voice signals and observe the variations in their spectra. In part III, we examine the effect of filtering on the sound quality. Finally we listen to the effect of slight frequency translation, or modulation.

Part I: Audible Range of our Hearing System

1. Set an external power supply to sinusoidal signal of frequency 10 Hz, and peak value 2 V.
2. Connect the signal to the input of the **HEADPHONE AMPLIFIER** module. Make sure you connect the ground of the external signal generator to the ground of **TIMS**.
3. On the **HEADPHONE AMPLIFIER**, keep the gain knob to a low setting, and set the **LPF SELECT** to **OUT** (i.e. no filtering).



As we go on with the experiment the sound will become sharp and loud, and you may not feel comfortable to hear it. Therefore keep the headphone aside, but near enough to hear the sounds.

4. Increase the frequency of the input gradually. Note the change in the sound. Record the range (lowest frequency and highest frequency) over which you can hear the sound.
5. Let your partner do the same. Who of you has a wider hearing range?

Part II: Spectrum of Speech Signals

1. Put the switch of CHANNEL 1 of SPEECH module to RECORD. Speak clearly to the microphone for few seconds. Speak continuously and avoid silence periods as much as possible. Put the switch to PLAY position.
2. Observe the spectrum of the signal on the spectrum analyzer. Estimate the bandwidth (Estimate the noise floor, and consider the spectrum above the noise floor). Save a representative snapshot and plot it.
3. Now we want to record a continuous sound with high pitches, something like “whistle” or sharp ring tone from your mobile. Prepare something and have it ready.
4. Start the high-pitch sound, then switch CHANNEL 2 to RECORD for few seconds before you switch back to PLAY.
5. Observe the spectrum of the signal on the spectrum analyzer. How is the spectrum different from that of CHANNEL 1? Save a representative snapshot and plot it.

Part III: Filtering the Speech Signal

1. Feed the output of CHANNEL 2 to the TUNABLE LPF. Toggle the switch to NORM and set the gain to maximum.
2. Observe the spectrum of the signal before and after the filter on the same view, but separate the two plots (right click >> choose auto arrange).
3. Start from the highest possible cutoff frequency (full clockwise) and go down. Comment on what you see on the spectrum analyzer and what you hear from the headphone?

Part IV: Frequency Translation (modulation)

The spectrum of the signal may be translated on the frequency axis by multiplying the signal with a sinusoid (modulation property).

1. Connect the output of CHANNEL 1 of the SPEECH module to one input of the MULTIPLIER module.
2. Connect the external signal generator to the other input of the MULTIPLIER. Set the toggle switch of the MULTIPLIER to DC.
3. Make the arrangements on the picoscope to observe the signal before and after the multiplier simultaneously.
4. Start from the lowest frequency of the signal generator and increase it gradually. Observe the spectrum and listen to the voice before and after the MULTIPLIER. Comment on what you see and hear.
5. Take a snapshot of the translated spectrum when the signal from the external generator is set to 5 kHz.

Post-Lab Work

Using MATLAB:

- Record a speech message for 5 seconds. (use **wavrecord** command)
- Play the message to confirm the recording. (use **wavplay** command). You can use the **pause** and **disp** command to help you control beginning of recording.
- Calculate and plot the spectrum of the message.
- Translate the spectrum by 5 kHz.
- Plot the translated spectrum.
- Submit the m file and the plots.

Exp 5: DSBSC Modulation & Demodulation

Objectives:

By the end of this experiment, the student should be able to:

- demonstrate the modulation and demodulation process of DSBSC.
- realize the real-life difficulties and challenges in designing coherent demodulators.
- examine the implications of the lack of perfect coherence on the recovered signal, and distinguish the different forms of distortion.

Introduction

Double Side Band Suppress Carrier (DSBSC) is one type of Amplitude Modulation. The modulation process is straightforward: the message is multiplied by a high-frequency carrier. The modulated signal occupies double the bandwidth of the baseband signal.

Recovering the message signal from the demodulated signal is performed coherently. That is, the demodulated signal is multiplied by a high-frequency sinusoid in perfect synchronization (in phase and frequency) with the incoming carrier. This requirement poses a challenge on the design of the demodulator circuit, as it would then require a part for carrier-recovery. Failing to accomplish perfect synchronization will result in phase mismatch or frequency mismatch, leading to some form of distortion in the recovered signal.

Multiplying the modulated signal with a local carrier will produce a baseband signal as well as a signal modulated at double the carrier frequency. Therefore, a LPF is needed at the far end of the demodulator to recover the baseband signal.

The following plug-in modules will be needed to run this experiment: AUDIO OSCILLATOR, QUADRATURE UTILITIES, TUNABLE LPF, PHASE SHIFTER and VCO.

Lab Work

This experiment consists of four parts. In Part I we generate the DSBSC signal using single-tone message signal. In Part II we demodulate the signal, assuming perfect synchronization of incoming and local carriers. We also examine the effect of improper filtering. In part III and IV, we examine the effect of phase and frequency mismatch, respectively.

Part I: Generation of DSBSC

1. Sketch the module diagram to generate DSBSC.
2. Generate a DSBSC signal where:
 - The message is sinusoid, $f = 2$ kHz (from the MASTER SIGNALS module)
 - The carrier has a frequency of 9 kHz (use the AUDIO OSCILLATOR)
 - Use one multiplier from the QUADRATURE UTILITIES module.

3. Plot the spectrum of the DSBSC signal.

Part II: DSBSC Demodulation

1. Borrow the same carrier of Part I, and use the second multiplier of QUADRATURE UTILITIES and a TUNABLE LPF to demodulate the DSBSC generated in Part I.
2. Observe the signal in time and frequency domains before and after the LPF simultaneously. Is the spectrum before the filter what you expected? Explain.
3. Vary the cutoff frequency of the LPF, and find the range of acceptable values for best recovery of the message.

(Note: You can measure the 3-dB cutoff frequency of the LPF by connecting the TTL output of the filter to the TTL input of FREQUENCY COUNTER, and divide the reading by 100. While taking the measurement, make sure nothing is connected to the analog input of the counter).

4. Plot, in time and frequency, the best recovered signal you can obtain.
5. Increase the cutoff frequency of the LPF beyond the range of good recovery. What happens to the recovered signal? Why?

Part III: Effect of Phase Mismatch

In this part we use the PHASE SHIFTER module to introduce a phase error between the carrier at the transmitter and the carrier at the receiver.

1. Set the cutoff frequency of the LPF in the demodulation circuit to any value in the good range for recovery.
2. Instead of borrowing the carrier from the transmitter, feed the carrier of the transmitter to the PHASE SHIFTER module and take the output to the multiplier of the demodulator circuit.
3. Observe the original message signal and the recovered signal simultaneously in time domain. Vary the phase shift, and describe the effect on the recovered signal.

Part IV: Effect of Frequency Mismatch

Of course, no one is interested in making frequency mismatch intentionally. But in real life you cannot borrow the carrier from the transmitter. (Otherwise you could have borrowed the message itself and saved all the hassle of communication!). One will do his best to reproduce a carrier at the same frequency used at the transmitter, but they cannot be 100% identical. In this part, we use a different source to generate the carrier for the demodulator circuit.

1. Generate an independent 9 kHz signal for the receiver circuit. For that you can use the Voltage-Controlled Oscillator (VCO) module.

(VCO module has many applications to come later in the course. For now, we will use it to generate a sinusoidal signal. Make sure that the on board switch is set to

VCO mode. Use the FREQUENCY COUNTER to measure the frequency. Make sure nothing is connected to the TTL input of the counter).

2. Observe, simultaneously, the original signal and the recovered signal, in time and frequency. Describe the effect of frequency mismatch.
3. Try to eliminate the frequency mismatch by fine tuning either oscillator.
4. Replace the 2 kHz message with a speech signal. Increase and decrease the frequency mismatch and describe the effect on the sound quality.

Post-Lab Work

1. You noticed that there is a wide range for the design of the LPF cutoff in order to recover the demodulated signal. Is there a particular value you prefer? Why?
2. You have seen that constant phase mismatch results in no shape distortion of the signal, only magnitude reduction. Prove this mathematically.
3. If the phase mismatch was not constant but time-varying, how would it affect the sound?
4. Implement DSBSC modulation and demodulation (perfectly coherent) in MATLAB for the signal and carrier frequencies used in the experiment. Submit the *.m file, plot of modulated signal and recovered signal in time and frequency. What is the difference between spectra plots in MATLAB simulation and the ones observed on the picoscope.

This page is intentionally blank. All Experiments start with odd pages for double-sided printing

Exp 6: AM and QAM

Objectives

- Demonstrate the modulation and demodulation process of AM signals.
- Implement the modulation and demodulation of Quadrature Amplitude Modulation (QAM).
- Examine the sensitivity of QAM to phase errors.

Introduction

AM is the term given to Double Side Band with Carrier modulation. The main advantage of AM over DSBSC is the simplicity of the demodulator circuit: the envelope detector. For that to work, the message signal has to be always positive. Therefore, the message is DC shifted before modulation. This gives rise to the model

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

For the envelope of $s(t)$ to be a true representation of $m(t)$, $A > m_p$ where m_p is the max negative value of the message. The ratio m_p/A defines the modulation index, which varies from 0 to 1.

The demodulation of AM signal can be achieved by a simple circuit of a diode, a resistor and a capacitor. The simplicity of the demodulator is the main attraction of AM modulation. However, this is done at the cost of less power efficiency. In terms of bandwidth, AM requires the same band for transmission as DSBSC.

Quadrature Amplitude Modulation, on the other hand, allows for twice the bandwidth efficiency of DSBSC or AM. In this scheme two messages are modulated with carries that have the same frequency but at quadrature to each other (i.e. have a phase shift of 90°). The two modulated signals are added and transmitted over the same channel. It is easy to show that each of the messages can be perfectly recovered with coherent demodulation at the receiver. The whole system is summarized in Figure 1. Such a scheme, however, is very sensitive to phase errors. Any error will result in one message leaking to the other.

The following plug-in modules will be needed to run this experiment: AUDIO OSCILLATOR, ADDER, MULTIPLIER, QUADRATURE UTILITIES, PHASE SHIFTER and UTILITIES.

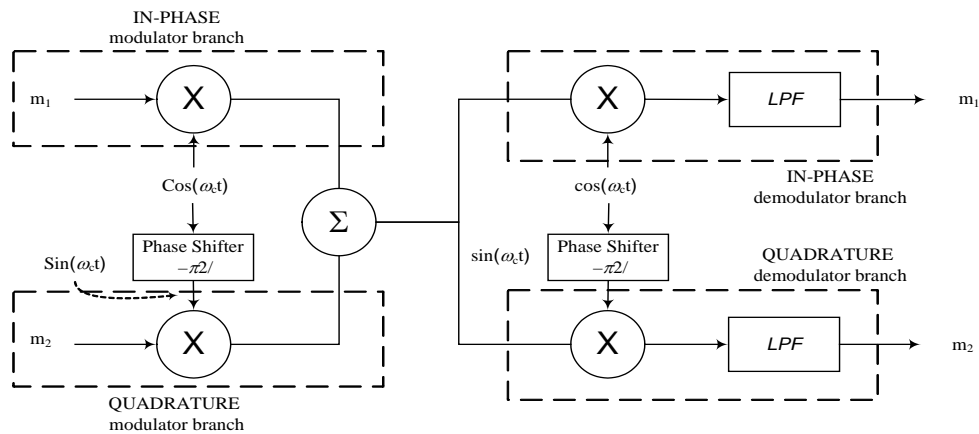


Figure 1: QAM Modulator / Demodulator

Lab Work

This experiment consists of three parts. In Part I, we generate and demodulate an AM signal using single-tone message as well as a real speech signal. In part II, we implement a QAM modulator and demodulator. In part III, we study the effect of phase error on the operation of QAM.

Part I: AM Modulation and Demodulation

1. Generate an AM signal where:

- The message is sinusoid, $f = 2$ kHz (from the MASTER SIGNALS module)
- The carrier has a frequency of 100 kHz (from the MASTER SIGNALS)
- Modulation index = 0.75
- Peak value of modulated signal 2 V

(Tune the DC VARIABLE and the gains of the ADDER G and g until you get the desired settings. Make sure the MULTIPLIER coupling is set to DC)

Hint: $\text{modulation index} = (A-B)/(A+B)$ (2)

where A and B as shown in the figure below.

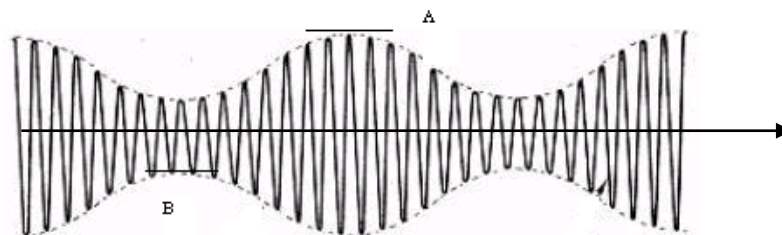


Figure 2: Designing modulation index

2. Plot the waveform and the spectrum of the AM signal
3. Demodulate the signal using the envelope detector (DIODE+ LPF circuit on the UTILITIES module).
4. Plot the demodulated signal. Is it perfectly recovered?
5. Feed the demodulated signal to the HEADPHONE AMPLIFIER module and plot the signal at the output of the module. Remember that this module has a built in LPF. Is the recovery better? Why?

Now we want to examine if the level of improvement achieved in step 5 compared to step 4 is significant to a speech signal.

6. Replace the single-tone message by a speech signal.
7. Listen to the signal with and without the LPF of the HEADPHONE AMPLIFIER module (use the toggle switch of the LPF SELECT). Do you notice any difference in quality? What is the lesson learned?

Part II: QAM Modulation and Demodulation

Set up the block diagram of QAM modulator and demodulator as shown in Figure 1.

1. Use the QUADRATURE UTILITIES module (two multipliers and one adder) for the modulator part of the system. Use the upper multiplier to modulate message 1, and the lower multiplier to modulate message 2. , according to the following steps:
 - Message 1 is fed from CHANNEL 1 of SPEECH module.
 - Message 2 is fed from CHANNEL 2 of SPEECH module
 - Prepare a carrier of 9 kHz sinusoid using the AUDIO OSCILLATOR.
 - Use the in-phase component of the oscillator, $\cos\omega t$, as the carrier to message 1, and the quad-phase component, $\sin\omega t$, as the carrier to message 2.
2. To demodulate the signal, use the MULTIPLIER module followed by the HEADPHONE AMPLIFIER
 - One input of the multiplier should come from the modulated signal.
 - For the other input, connect the in-phase component of the carrier. Can you hear message 1 clearly?
 - Connect the quad-phase carrier. Can you hear message 2 clearly?
 - Ask you instructor to check your system and sign the lab sheet.

Part III: Effect of Phase Error on QAM

1. Feed the carrier of message 2 through PHASE SHIFTER module before connecting it to the modulator circuit.
2. Vary the phase shift and keep listening while demodulating message 1. What do you notice?

3. Vary the phase shift and keep listening while demodulating message 2. What do you notice?

Post-Lab Work

1. Prove Equation (2).
2. Show mathematically that when the carries are not perfectly in quadrature with each other, the two messages leak in one another at the demodulator output.
3. Implement Part I of the experiment (AM Modulation and Demodulation) in MATLAB. Submit the *.m file, and the plots of modulated signal and recovered signal in time and frequency.

Exp 7: FM Modulation

Objective:

The objectives of this experiment are to:

- generate FM signals using VCO.
- understand the modulator sensitivity and linear range of operation.
- examine the factors affecting the shape of the spectrum and the bandwidth of FM signals.

Introduction

A simple and direct method of generating an FM signal is by the use of a voltage controlled oscillator -VCO. The frequency of such an oscillator can be varied by the magnitude of an input (control) voltage. The block diagram of VCO-FM generator is shown in Figure 1(a). Figure 1(b) shows a snap shot of an FM signal, together with the message from which it was derived². Note particularly that there are no amplitude variations - the envelope of an FM waveform is a constant.

For the VCO to work as a frequency modulator, it has to manifest a linear relation between the magnitude of the input signal and the output oscillation. Large signal amplitude may take the system out of its linear range of operation. Therefore a careful design of the deviation sensitivity of the VCO is required to ensure linear operation over the full range of input signal amplitudes.

Unlike Amplitude modulation, the bandwidth of FM signals is not determined by the message bandwidth only, but also by message (maximum) amplitude and deviation sensitivity. The product of the last two factors yields frequency deviation. The bandwidth of FM signal can be approximated by (Carson's rule):

$$\text{Bandwidth of FM signal} = 2 \times (\text{message bandwidth} + \text{frequency deviation})$$

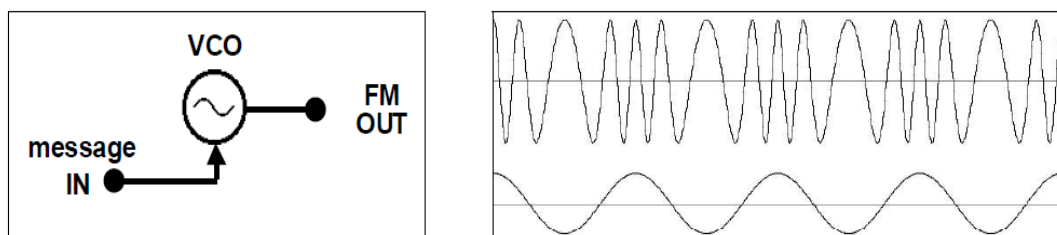


Figure 1: FM by VCO (a), and resulting output (b).

²In this figure the frequency deviation is comparable to the carrier for the objective of visualizing the frequency change. In real signals, the frequency deviation is very small compared to the carrier.

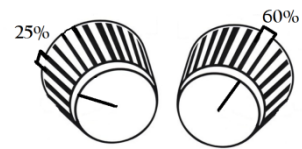
Lab Work

This experiment has four parts. The first part studies sensitivity and the range of linear operation of the voltage controlled oscillator (VCO). In preparation for FM generation in the third part, part II addresses designing the frequency deviation ratio for the modulator. Spectrum analysis and bandwidth estimation will be the subject of the last part. The following modules are needed to complete the experiment: AUDIO OSCILLATOR, VCO.

Part I: Sensitivity and Linearity of VCO

The output frequency of the VCO varies with the input voltage, V_{in} . The amount of variation (Hz/volt) can be controlled by the deviation sensitivity (GAIN). Before generating an FM waveform it is required to set the deviation sensitivity to a value that ensures linearity of the VCO over the whole range of message amplitudes.

1. Plug in the VCO and make sure that the on-board switch SW2 is set to 'VCO'.
2. Use the front panel ' f_0 ' control to set the output frequency (sin ωt) close to 100 kHz.
3. Connect the VARIABLE DC voltage to the input (V_{in}) of the VCO.
4. The deviation sensitivity can be set with the front panel GAIN control. Set this to about 25% of its fully clockwise rotation. *See the figure for approximate setting.*
5. Vary the VARIABLE DC from -2 V to +2 V in steps of 0.5 V and measure the output frequency. You may use the measurement facility in picoscope to measure the DC value. Fill in the first row in the table in the Lab Sheet.
6. With the variable DC on its minimum value, set the GAIN control (sensitivity) of the VCO to about 60% (make sure that you do not overload the VCO, the LED should not light up). Redo step 5 and fill the other row in the table.
7. Plot the output frequency versus the input voltage for each setting.
 - Which of the above settings results in a more linear performance in the given range of V_{in} ?
 - Determine the linear range for the second case (60% setting)
 - Using the table only, estimate the frequency of the VCO when the DC input is 1.75 V for both settings? Which setting results in easier interpolation? Why?



Part II: Setting the Frequency Deviation

The frequency deviation is equal to the product of $V_{in,max}$ and GAIN. Our objective is to design the GAIN that yields frequency deviation of ± 10 kHz, for an input signal of 4 volts peak-to-peak. This can be done as follows:

1. Set a DC voltage of -2 V as input to VCO.
2. Set the GAIN control fully anti-clockwise and the output frequency to 100 kHz.
3. Advance the GAIN control until the frequency changes by 10 kHz.
4. Change to VARIABLE DC to +2V and confirm that the deviation is about 10 kHz in the other direction. Record the measured frequency.

Part III: FM Generation

1. Replace the DC voltage source with the output from an AUDIO OSCILLATOR. The frequency deviation will now be about ± 10 kHz, since the AUDIO OSCILLATOR output is about 2 volt peak. Why the frequency counter is still at 100 kHz?
2. Observe the generated FM on the oscilloscope. Adjust the range and zoom-in to optimize the view. Try $20 \mu\text{s}/\text{div}$.
3. Vary the frequency of the AUDIO AOSCILLATOR. Explain the change in the modulated signal.
4. Vary the GAIN of VCO. Explain the change in the modulated signal.

Part IV: Spectrum Analysis and Bandwidth Estimation

Many interesting observations can be made regarding the FM spectrum.

1. Fix the message frequency from the AUDIO OSCILLATOR to 2 kHz, and the VCO gain to about 25%. Plot the spectrum, zooming in the frequency range (40,180 kHz).
2. Vary the message frequency and describe the impact on the spectrum of the FM signal. Plot the spectrum of the FM signal at the minimum and maximum frequencies of the AUDIO OSCILLATOR.
3. Reset the frequency of the message to 2 kHz, and vary the deviation ratio (by varying the GAIN in the VCO). Describe the effect on the spectrum of the FM signal (make sure you do not overload the VCO). Plot the spectrum at the minimum value and maximum GAIN setting (before overload).
4. Explain the obtained spectra in light of Carson's Rule for bandwidth estimation.

Post-Lab Work

We would like to verify our results in "Part IV: Spectrum Analysis and Bandwidth Estimation" using MATLAB:

1. Use MATLAB to generate an FM signal $y(t)$, let the message signal be $x(t)=2\cos(2000(2\pi)t)$. Use a carrier frequency of $f_c=100$ kHz and design the sensitivity factor to get a frequency deviation of 10 kHz.

Hint: use the *fmod* command available from the Communication Toolbox™.

2. Plot $x(t)$ and $y(t)$.
3. Plot the magnitude spectrum for $y(t)$.
4. Change the message frequency to 4 kHz, observe the spectrum and quantify the effect on the bandwidth
5. Change the frequency deviation to 15 kHz (keep message frequency at 2 kHz), observe the spectrum and quantify the effect on the bandwidth
6. Compare Simulation results with the experimental ones.

This page is intentionally blank. All Experiments start with odd pages for double-sided printing

Exp 8: FM Demodulation

Objective:

There are two main objectives for this experiment:

- to implement the phase locked loop (PLL) for FM demodulation.
- to implement frequency discriminator method for demodulating FM.

Introduction

The block diagram of a phase locked loop (PLL) is shown in Figure 1. The principle of operation is simple. Suppose there is an unmodulated carrier at the input. If the VCO was tuned precisely to the frequency of the incoming carrier, ω_o , then the instantaneous output would be a DC voltage, of magnitude depending on the phase difference between the output of the VCO and the incoming carrier. Now suppose the that the incoming carrier started to drift slowly in frequency. Depending upon which way it drifts, the output voltage will vary accordingly. If the incoming carrier is frequency modulated by a message, the output of the PLL will follow the message.

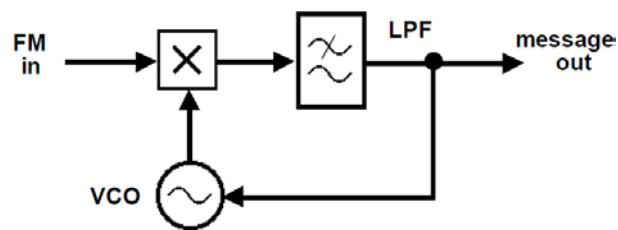


Figure 1: the PLL

FM can be demodulated as well by using a differentiator or a frequency discriminator. Frequency discrimination can be achieved by applying the FM signal to the linear part (transition region) of a BPF as depicted in Figure 2. The output of the discriminator is both FM and AM modulated. The message can be recovered by applying the discriminator output to an envelope detector followed by LPF.

The BPF of the 100 kHz CHANNEL FILTERS module has close-to-linear pattern in the band 80-90 kHz.

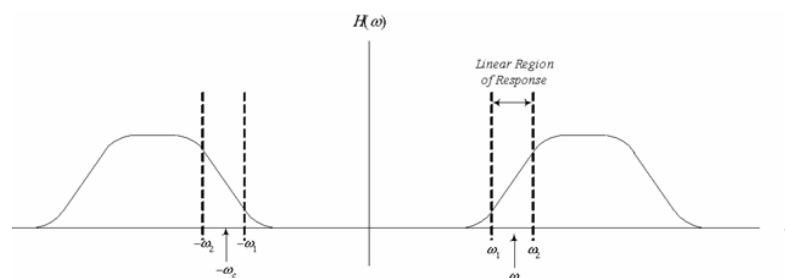


Figure 2: Band Pass Filter as a frequency discriminator

The following modules are needed to complete the experiment: AUDIO OSCILLATOR, ADDER, MULTIPLIER, UTILITIES, 100 kHz CHANNEL FILTERS, VCO (2 modules, one for the modulator and the second is used for the PLL demodulator).

Lab Work

Part I: FM Demodulation Using PLL

1. Reconstruct the FM modulator as in the previous experiment (FM Modulation). Let the message be 2 kHz from the AUDIO OSCILLATOR, the carrier 100 kHz from VCO, and the modulator VCO GAIN about 25%.
2. Model the PLL demodulator illustrated in Figure 1. For the filter use RC LPF provided in the UTILITIES Module. In the MULTIPLIER module set the toggle switch to AC. Draw the corresponding module diagram.
3. Set the VCO in the demodulator to 100 kHz. Set the GAIN control to mid-range position.
4. Connect the output of the modulator to the input of the demodulator.
5. The PLL may or may not lock on to the incoming FM signal. Tune the GAIN (and if necessary the center frequency) of the PLL-VCO until you obtain lock. Examine the output of the PLL VCO and compare it with the original message.
6. Replace the message from the AUDIO OSCILLATOR by a speech signal and make sure that you can hear the recorded message correctly. Study the effect of varying the frequency f_0 and GAIN of the PLL-VCO on the quality of the received speech.

Part II: Frequency Discriminator

1. Set the VCO to generate an FM signal with carrier frequency 85 kHz and GAIN around 25%.
2. Connect the FM signal to the BPF (Use the 100 kHz CHANNEL FILTER MODULE, set CHANNEL SELECT to 3).
3. Perform envelope detection by connecting the BPF output to the DIODE+LPF in the UTILITIES module.
4. Connect the output of the envelope detector to the HEADPHONE AMPLIFIER.
5. Apply a speech signal to the FM modulator (VCO), and listen to the demodulated signal.
6. Tune the VCO carrier frequency slightly around the 85 kHz until you get the best output (BPF modules may have slightly different characteristics). Get the approval of your instructor for this step.

Post-Lab Work:

- 1) Show mathematically the operation of the frequency discriminator.
- 2) Comment on the pros and cons of the two demodulation techniques covered in this experiment.

- 3) Use the FM signal generated in the LAST experiment post-lab work and write a MATLAB code to demodulate the signal. Plot the transmitted and recovered signal and note down your observations.

Useful MATLAB Function: *fmdemod* command available from the Communication Toolbox™

This page is intentionally blank. All Experiments start with odd pages for double-sided printing

Exp 9: PCM ENCODING

Objectives:

- Recognize the various processes of PCM encoding.
- Realize the structure of the PCM stream.
- Understand the operation of linear and non-linear quantizers.

Introduction

Pulse Code Modulation (PCM) is a method of converting an analog signal into a digital signal (A/D conversion). This is achieved by a PCM encoder via three operations in sequence: sampling, quantization and coding. A step-by-step description of the operations of a standard PCM encoder is as follows:

1. The encoder is driven by a TTL clock.
2. The input analog message is sampled periodically. The sample rate is determined by the external clock.
3. Each sample amplitude is compared with a finite set of amplitude levels, called quantization levels. These are distributed within the range $\pm V$ volts.
4. Each sample is assigned a digital (binary) codeword representing the number associated with the quantizing level which is closest to the sample amplitude. The number of bits ' n ' in the digital codeword and the number of quantizing levels L are related by the equation $L = 2^n$.
5. The codeword is assembled into a time frame, together with other bits as may be required. In many commercial systems, a single extra bit is added in the least significant bit position. This is alternately a "0" or a "1". These bits are used by subsequent decoders for frame synchronization. The frame is transmitted serially.

A typical operation that takes place when performing A/D conversion of speech signals is *companding*. Companding stands for signal compression at the encoder and expansion at the decoder. In the encoder, compression makes the quantizing levels for small input amplitudes closer than those for large amplitudes, in a logarithmic proportion. At the decoder the 'reverse action' is performed to restore the original amplitude distribution. Companding is particularly advantageous when the message has high peak-to-average amplitude characteristic, as in speech signals.

The following modules will be used: PCM ENCODER and AUDIO OSCILLATOR.

PCM ENCODER Module of TIMS

In TIMS, A/D conversion is performed by the PCM ENCODER module, Figure 1. The input to the PCM ENCODER is an analog message. The sampling rate of the module is defined by (*but not equal to*) the CLK input, which sets a limit on the maximum allowable message bandwidth, according to Nyquist Sampling Theorem. The dynamic range of the quantizer is

designed for the range ± 2.0 volts, therefore the input message amplitude must be held within this range.

The technical details of the module are described in the TIMS *Advanced Modules User Manual*. We go briefly over each of the input and output connections which will be used in this experiment.

- DIGITISING SCHEME SELECT: a three-position toggle switch which selects the 4-bit or 7-bit linear encoding scheme; or the 4-bit companding scheme.
- FS: frame synchronization, a signal indicating the end of each data frame.
- V_{in} : the analog signal to be encoded.
- PCM DATA: the output data stream.
- CLK: a TTL input serves as the MASTER CLOCK for the module. Clock rate of this module must be 10 kHz or less (manufacture limitation). For this experiment we will use the 8.333 kHz TTL signal from the MASTER SIGNALS module.

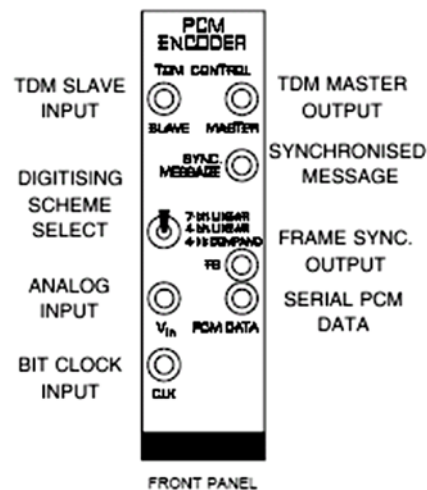


Figure 1: The front panel of the PCM ENCODER

Each binary word is arranged in a *time frame*. The time frame contains eight *slots* of equal length, and is eight clock periods long. The slots, from first to last, are numbered 7 through 0. These slots contain the bits of a binary word. The least significant bit (LSB) is contained in slot 0. See Figure 2.

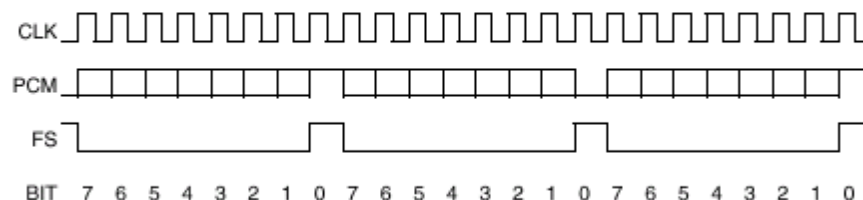


Figure 2: PCM ENCODER timing frame

The LSB consists of alternating *ones* and *zeros*. These are placed (embedded) in the frame by the encoder itself, and cannot be modified by the user. They are used by subsequent decoders to determine the location of each frame in the data stream, and its length.

The remaining seven slots are available for the bits of the binary codeword. Thus the system is capable of a resolution of a maximum of seven bits (128 levels). This resolution, for purposes of experiment, can be reduced to four bits (by front panel toggle switch). The 4-bit mode uses only five of the available eight slots - one for the embedded frame synchronization bits, and the remaining four for the binary codeword (in slots 4, 3, 2, and 1). See Figure 3.

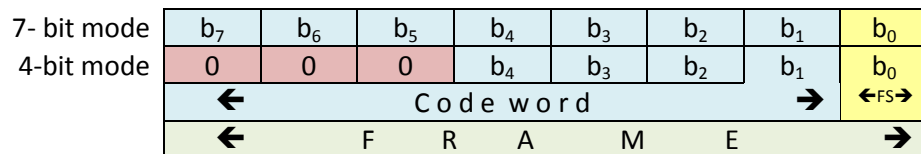


Figure 3: Frame structure for 4-bit and 7-bit words.

Lab Work

The experiment consists of four parts. In Part I, we set up the module and observe the time frame structure. In Part II, we identify the quantization levels by examining PCM of a DC input using 4-bit linear quantizer. A DC input ensures completely stable oscilloscope displays, and enables easy identification of the quantizing levels. In Part III, we switch to 4-bit companding mode and observe the difference in quantizer behavior to the linear mode. Finally, in Part IV, we consider a more meaningful input; an AC signal.

Part I: PCM Frame Structure

1. Plug in the PCM ENCODER into the TIMS shelf.
2. Patch the 8.333 kHz TTL CLOCK from the MASTER SIGNALS module to the CLK input of the PCM ENCODER.
3. On one of the oscilloscope channels display the frame synchronization signal FS. Adjust the sweep speed to show three frame markers. These mark the end of each frame.
4. On the second oscilloscope channel display the CLK signal.
5. Place the two waveform on the top of each other so that they can be easily compared, and plot them.
6. What is the frame duration? Bit duration? Codeword duration?
7. What is the sampling rate of the PCM ENCODER? Is it appropriate to sample a speech signal? Why?

Part II: Quantizing levels for 4-bit linear encoding

1. Set the toggle switch to 4-BIT LINEAR. Though standard PCM uses 7 bits, selecting the 4-bit encoding scheme will reduce the number of quantization levels to be examined to 16 only.

2. Feed a DC voltage level from the VARIABLE DC module to V_{in} of the PCM ENCODER. Turn the knob of the variable DC voltage to the least (negative) value (full counterclockwise).
3. Display on the oscilloscope both the FS signal and the PCM DATA output. Describe the binary sequence of the PCM DATA for three consecutive frames. Give reasons for such sequence.
4. Vary the DC voltage slowly back and forth over its complete range, and note how the data pattern changes in discrete jumps.
5. Adjust V_{in} to its maximum negative value. Record the DC voltage. You should be getting all zeros for the 4-bit binary number for this DC setting.
6. Gradually increase the amplitude of the DC input signal until you notice a change to the PCM output. Record the binary sequence of the new digital word, and the input amplitude at which the change occurred.
7. Continue this process over the full range of the DC supply.
8. Sketch the input-output characteristics of the quantizer. This is a staircase plot, with the input voltage on the x-axis and the output level (labeled as a binary sequence) on the y-axis.
9. Connect the corners of the stair steps with the best fit. Is the curve linear?

Part III: Non-Linear Quantization

Set the toggle switch to 4-bit companding and repeat steps 5-9 of Part II. Were you able to observe all 16 levels? Why?

Part IV: Time-Varying Messages

1. Set the toggle switch to 7-BIT LINEAR.
2. Connect the AUDIO OSCILLATOR output to V_{in} of the PCM ENCODER.
3. Observe the PCM DATA output over consecutive frames. How it is different from the DC input case?

Post-Lab Work

Write a MATLAB code to implement uniform quantization. The quantizer should accept a sampled signal and generate the quantized one. The dynamic range of the quantizer is $-V$ to $+V$. The number of bits should be controllable. Demonstrate your program by two examples.

Exp 10: PCM Decoding

Objective:

The objectives of this experiment are to:

- implement PCM decoding and understand its operation.
- appreciate the importance of data structure and synchronization.
- evaluate the effect of companding.

Introduction

In the previous experiment, the PCM encoder was examined. To complete the picture, the PCM decoder will be investigated in this experiment. Upon reception of a PCM sequence, the PCM decoder:

1. extracts a frame synchronization signal FS from the data itself (from the embedded alternate ones and zeros in the LSB position), or uses an FS signal borrowed from the transmitter .
2. extracts the binary number, which is the coded (and quantized) amplitude of the sample.
3. identifies the quantization level which this number represents.
4. generates a voltage proportional to this amplitude level.
5. presents this voltage to the output V_{out} . The voltage appears at V_{out} for the duration of the sampling period.

Note that, *it is not possible to recover a distortionless message from these samples. They are flat top, rather than natural samples.* The decoder itself has introduced no distortion of the received signal, but the signal from the PCM encoder is already an **inexact** version of the signal at the input of the encoder. Message reconstruction can be improved by low pass filtering.

To complete the experiment the following modules are needed: PCM DECODER, PCM ENCODER, SPEECH MODULE, TUNABLE LPF.

PCM Modules

In this experiment we will use the PCM ENCODER and DECODER modules.

The decoder is driven by an external clock, borrowed from, and so synchronized to, that of the encoder. Frame synchronization may be achieved either by extracting the FS signal from the **embedded** information in the received data, or by borrowing it **externally** from the encoder. A toggle switch (FS SELECT) in the PCM DECODER allows switching between the two modes.

Though an external signal can be used for demonstrating the PCM operation, the PCM encoder is equipped with test periodic signals (SYNC MESSAGE) of frequencies which are fraction of the clock rate. These signals are synchronized with the clock and, therefore,

provide improved triggering. The frequency of these signals can be set from a dip switch SW2 on the board of the PCM ENCODER. Note that these signals are not pure sinusoids.

You can read more about the PCM DECODER module in the *TIMS Advanced Modules User Manual*.

Lab work

This experiment consists of four parts. In part I, we set up the PCM encoder and decoder. In parts II to IV, we examine quantization, signal recovery, and companding, respectively.

Part I: System Setup

The PCM Encoder

1. Set up the PCM ENCODER for a clock rate of 8.3kHz and coding scheme 4-bit LINEAR.
2. Choose a 'large' negative DC for the message (from the VARIABLE DC module).
3. Connect the output of the PCM ENCODER to CH-A1 of the oscilloscope and confirm that the corresponding codeword is '0000', so only the embedded alternating '0' and '1' bits (for remote FS) in the LSB position should be seen. They should be 1920 ms apart. Verify by measurement and calculation.
4. Verify the operation of the PCM encoder by varying the DC source like you did in the previous experiment.

The PCM DECODER

1. Plug in the PCM DECODER module.
2. Use the front panel toggle switch to match the transmitter encoding scheme.
3. Clock the PCM DECODER by the same clock of the encoder.
4. Borrow the frame synchronization signal FS from the transmitter by connecting it to the frame synchronization input FS of the receiver (and check that the FS SELECT toggle switch is set to EXT FS).
5. Connect CH-A1 to the input to the PCM ENCODER module.
6. Connect CH-B1 to the sample-and-hold output, V_{out} , of the PCM DECODER.
7. Ensure that both channels of the oscilloscope are set to accept DC to calibrate the ground of both channels to the zero level. Fix the voltage axis to view -5 V to +5 V..

Part II: Quantization Effects

The effect of number of quantization levels will be investigated with the help of a DC message.

1. **Slowly** vary the DC output from the VARIABLE DC module back and forth over its complete range. Notice that as the input to the encoder moves continuously, the output from the decoder moves in discrete jumps. Explain this behavior.
2. Reset the coding scheme on both modules to 7-bit. Sweep the input DC signal over the complete range as before. Notice the 'granularity' in the output is almost unnoticeable compared with the 4-bit case. *Comment*.

Part III: Signal Recovery

It was not possible, when examining the PCM ENCODER in the previous experiment, to see the sample-and-hold waveform within the *encoder*. But, assuming perfect decoding, it is available at the output of the *decoder*.

1. Connect an external triangular signal of amplitude 4 V (peak-to-peak). Adjust the frequency to 20 Hz. Do not forget to connect the ground of the source to the GND of TMS.³
2. Set the Encoder/DECODER to 4-bit LINEAR.
3. Slow down the oscilloscope sweep speed to 10 ms/div.
4. Observe the decoder output signal. Plot both the input to the PCM ENCODER and the output of the PCM DECODER.
5. Measure the delay between the input and the output.
6. How many samples are taken in one period of the input signal?
7. Using the Picoscope >tools>math channels>A-B, obtain a plot of the error (output - input). Explain how the delay affects the error.
8. Change the coding scheme from 4-bit to 7-bit. Redo step 7.

Message reconstruction may be improved using careful low pass filtering.

9. Connect the output of the PCM DECODER to a TUNABLE LPF. Now observe the original signal (input to the PCM ENCODER) together with the output of the filter and start adjusting the cutoff frequency.
10. Using the FREQUENCY COUNTER, find the cutoff frequency of the TUNABLE LPF for best possible recovery. (Recall that the cutoff frequency = frequency counter reading/100). Plot the input and output waveforms.
11. Adjust the cutoff frequency of the TUNABLE LPF to 50Hz, then 500Hz. Plot the waveforms and comment on the type of distortion in the output signal.

Part IV: Companding

In this part, a speech signal is used to demonstrate the effect of companding.

1. Record your own message using a SPEECH MODULE (when you speech include some silence to be able to evaluate the background noise) and play it. Keep the PCM ENCODER and DECODER at 7-bit LINEAR.
2. At the receiver side use the headphone to judge the quality of the PCM audio signal. Make sure that the signal amplitude is not high, in order not to hurt your ears.
3. Examine the effect of companding by switching both the encoder and decoder to 4-bit COMPAND. Comment on the quality of the decoder signal compared to 4 bit-LINEAR.
4. Examine all the possible combinations of quantizer settings at the encoder and decoder in terms of quality, recoverable/non-recoverable, loud/low with reference

³ (Note: if no external source is available, the SYNC MESSAGE of the PCM ENCODER can be used after setting its peak-to-peak voltage between -2 and +2 using a BUFFER AMPLIFIER. The frequency of the SYNCH MESSAGE is determined by the SW2 on board, Dip switches on board SW2: set left hand toggle UP, right hand toggle DOWN use 65 Hz).

to the case of 4-bit LINEAR in both modules. First guess the output, and then verify your guess experimentally.

Post-Lab Work

- When testing the overall PCM encoding decoding system, we examined a DC signal with both 4-bit and 7 bit-linear quantizers (Part II). Describe the behavior if 4 bit-nonlinear (compand) was used?
- Using MATLAB, consider a sinusoidal signal with amplitude of V and frequency 1 kHz, sampled at 100 kHz. With the help of the Quantizer you designed in the post-lab work of the previous experiment, plot the quantized output in two cases: 4-bit and 8-bit quantizers. Compare the input with the output and plot the error in the two cases. *Comment.*

Exp 11: Line Coding

Objective:

The objectives of this experiment are to:

- get familiar with definitions and properties of commonly used line codes.
- evaluate the spectrum content for different line codes.
- demonstrate the advantage of differential encoding through signal inversion test.
- measure the delay associated with practical digital systems.

Introduction

Digital data can be transmitted by various *Line codes*. Line codes are waveform patterns of voltage or current used to represent the 1s and 0s. Each line code has its advantages and disadvantages. Among other desirable properties, a line code is preferred to have the following:

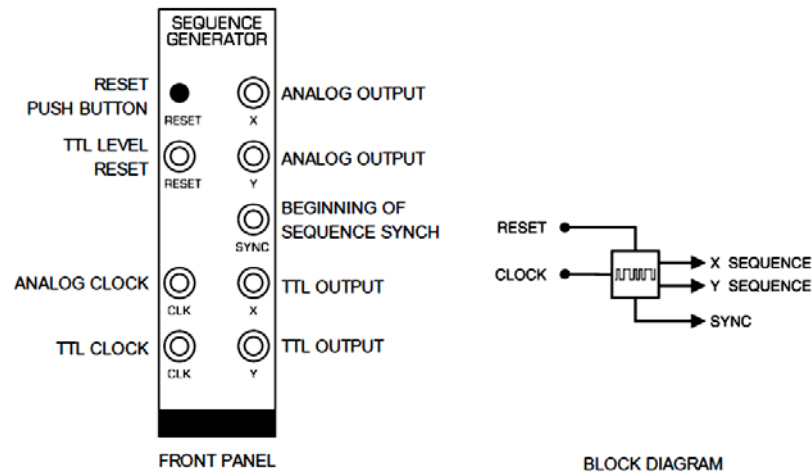
- *Bandwidth efficiency*; the possibility of transmitting at a higher rate than other schemes over the same bandwidth.
- *Power efficiency*: For a given bandwidth and quality, the transmitted power should be as small as possible.
- *No DC component*; this allows AC coupling (capacitor or transformer) between stages (as in telephone lines).
- *Spectrum shaping*; this is important in telephone line applications, for example, where the transfer characteristic has heavy attenuation below 300 Hz.
- *Synchronization*; where bit clock recovery can be simplified.
- *Error detection capabilities*; It should be possible to detect some patterns of errors.

System Modules

To complete the experiment the following modules are needed: SEQUENCE GENERATOR, LINE-CODE ENCODER, and LINE-CODE DECODER.

SEQUENCE GENERATOR

Using a common external clock signal, the SEQUENCE GENERATOR produces two independent pseudorandom sequences **X** and **Y**. In this experiment we need only one output. The SEQUENCE GENERATOR will be clocked by B.CLK from the LINE-CODE ENCODER module. A **SYNC** output is provided which is coincident with the start of the sequences. The sync out from the SEQUENCE GENERATOR can be used to trigger the oscilloscope. The sequences may be stopped and restarted at any time via front panel controls. Sequences **X** and **Y** are available as either standard TTL or analog level output. The SEQUENCE GENERATOR is a basic module and you can read more about it in *TIMS 301 User Manual*. The module and its block diagram are shown in the figure below.



LINE ENCODER and LINE DECODER Modules

In a digital transmission system line encoding is the final digital processing performed on the signal before it is connected to the analog channel, although there may be simultaneous bandlimiting and wave shaping. Thus in TIMS the LINE-CODE ENCODER accepts a TTL input, and produce an output that is suitable for transmission via an analog channel. The TIMS LINE-CODE DECODER decodes it back to the binary TTL format.

The LINE-CODE ENCODER serves as a source of the system bit clock. It is driven by a *master clock (M.CLK)* at 8.3 kHz (from the TIMS MASTER SIGNALS module). The LINE-CODE ENCODER module divides **M.CLK** by a factor of four, in order to derive some necessary internal timing signals at a rate of 2.083 kHz (**B.CLK**). The latter becomes a convenient source for use as the system bit clock. The reason we are using a slower clock (clock/4) is that the encoder requires some cycles to provide the proper output and hence the data should arrive at a slower rate.

Because the LINE-CODE DECODER has some processing to do, it introduces a time delay. To allow for this, it provides a re-timed clock (STROBE) if required by any further digital processing circuits (eg, for decoding, or error counting modules).

For a TTL input signal the following output formats are available from the LINE-CODE ENCODER: NRZ-L, NRZ-M, UNI-RZ, BIP-RZ, RZ-AMI, Bi \emptyset -L (*Manchester*), DICODE-NRZ. Rather than defining each of the previous codes, you will find what they mean experimentally.

Resetting the LINE-CODE ENCODER and the LINE-CODE DECODER is recommended every time after the clock is connected, or interrupted

You can read more about the LINE ENCODER and LINE DECODER Modules in the *Advanced Modules Manuals*.

Lab Work

This experiment has three parts. The first part takes you through defining the available line codes and observing their spectra. The second part verifies the overall operation of the line encoding-decoding system. Differential encoding and signal inversion test are examined in part III.

Part I: Line Codes Definitions and Spectra

1. Plug in the LINE-CODE ENCODER module and clock it with 8.3 kHz from MASTER SIGNALS.
2. Plug in the SEQUENCE GENERATOR module and clock it with **B.CLK** 2.083 kHz clock provided from the LINE ENCODER.
3. Connect the generated TTL sequence (X) to the LINE ENCODER module input data.
4. Using two channels, display the generated sequence (X) and the selected line code. Adjust the scope setting to display few bits (about 10 bits).
5. Observe the different line codes on the scope. Explain what each code does (the law of transformation).
6. Observe and save the spectra of the following two line codes: NRZ-L and Bi \emptyset -L (Manchester). Based on the spectra of the two codes, compare them in terms of DC content and first null bandwidth. How is this BW related to the time domain waveform?

Part II: Line Decoding

1. Connect the LINE DECODER module and make sure it is clocked with the clock received from the LINE ENCODER (**B.CLK**).
2. Connect any of the outputs of the LINE-ENCODER module to the corresponding input of the LINE-DECODER module. Press RESET on both modules.
3. Observe the generated sequence (at the output of the SEQUENE GENERATOR) together with the recovered sequence (at the output of the LINE_CODE DECODER) on the two channels of the oscilloscope. Are the two waveforms identical?
4. Plot the two waveforms and measure the introduced delay.

Part III: Signal Inversion Test

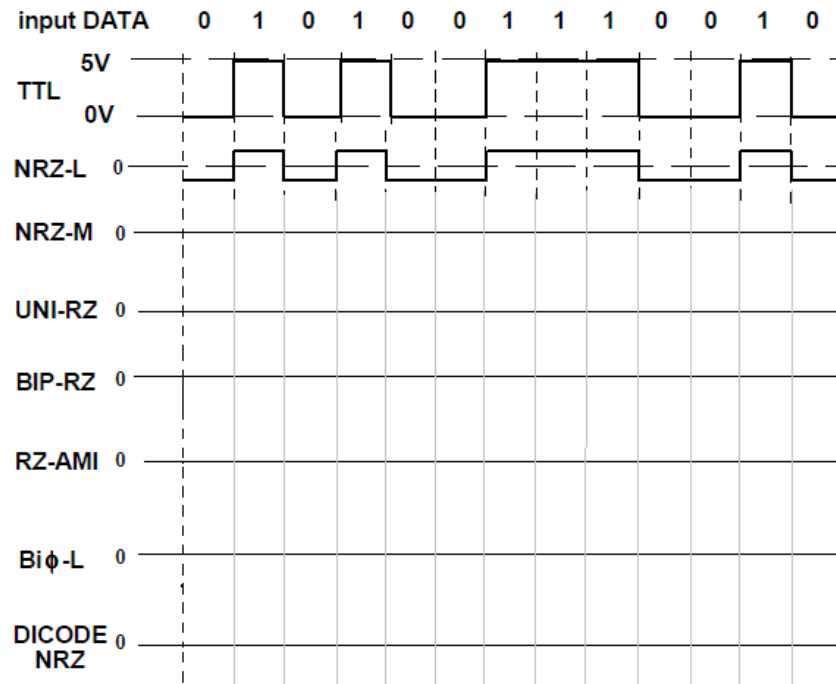
Some Line codes are differential in nature and are insensitive to the signal inversion. This feature is important for many communication systems as the signal can get inverted during transmission, for example, opposite connection of wires or inverting amplifiers results in an inverted signal.

To check the sensitivity of the specific line code to signal inversion:

1. Select a particular code and verify that the message is successfully decoded.
2. Insert a BUFFER AMPLIFIER (which is an inverting amplifier) in the transmission path.
3. Adjust the GAIN of the BUFFER AMPLIFIER. Note that the LINE-CODE DECODER requires, for successful decoding, an input signal of amplitude near ± 2 volt PP.
4. Patch the inverting amplifier -in and out- to find out if the code is sensitive to signal inversion or not (If there is no change to the message output, then the code is insensitive to polarity reversals).
5. Using the above test, identify the codes that are insensitive to channel inversion.

Post-Lab Work

1. Based on your experimental understanding of the law of transformation of the line codes, complete the figure below.
2. Comment on the DC component, synchronization capability and error detection capability of each code.



Exp 12: Digital Modulation: FSK

Objective:

By the end of this experiment, the student should be able to:

- modulate binary digital signals using FSK.
- understand the design issue related to choosing the frequencies in the FSK signal.
- differentiate continuous phase FSK (CPFSK) from general FSK.
- realize ASK and PSK as special cases of FSK.

Introduction

Digital modulation techniques include: amplitude shift keying (ASK), frequency shift keying (FSK), and phase shift keying (PSK). In this experiment we will concentrate on FSK. The other techniques can be related to it. The block diagram of Figure 1 illustrates the operation of an FSK generator.

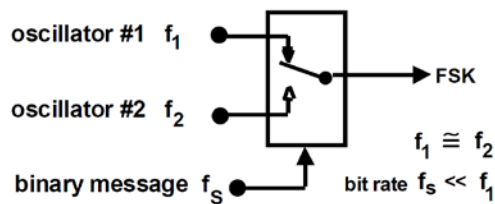


Figure 1: an FSK transmitter

In principle the three frequencies f_1 , f_2 , and f_s are independent. In practice this is often not so - there are certain advantages in having them related in some way (eg, as submultiples). Oscillator #1 and #2 can be taken from the same source (say a VCO), whose frequency is changed by the message, leading to a *continuous phase* output (CPFSK). This is illustrated in Figure 2, which shows a VCO as the source of f_1 and f_2 , and the corresponding CPFSK output waveform.

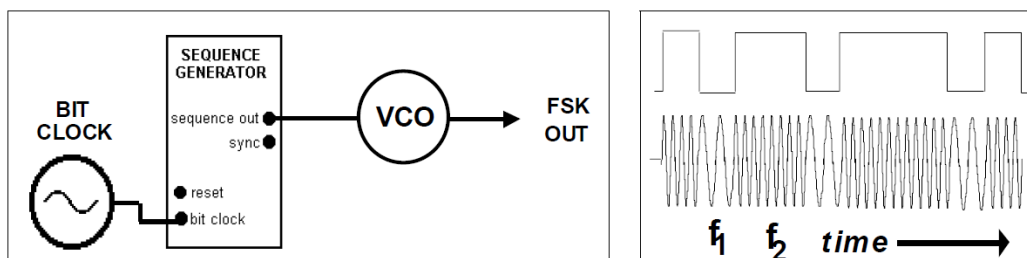


Figure 2: CPFSK generation and output waveform

To complete this experiment the following modules are needed: ADDER, AUDIO OSCILLATOR, DUAL ANALOG SWITCH, SEQUENCE GENERATOR, and VCO.

System Modules

VCO as FSK generator

The VCO module was used before for FM generation. It can also be used for FSK generation. In this case it must be switched to the FSK mode using the on-board switch SW2. A +5V DC to the DATA input of the VCO module allows the setting of f_1 with RV8, while a 0V allows the setting of f_2 using RV7. These frequencies will be in the audio range with the front panel switch set to LO, or near 100 kHz when set to HI. The two other front panel controls have no influence in this FSK mode. See the *TIMS User Manual* for details of FSK mode for the VCO.

DUAL ANALOG SWITCH

Two identical analog switches are controlled by digital TTL level signals. The outputs of the two switches are added internally and presented at the output of the module. See Figure 3.

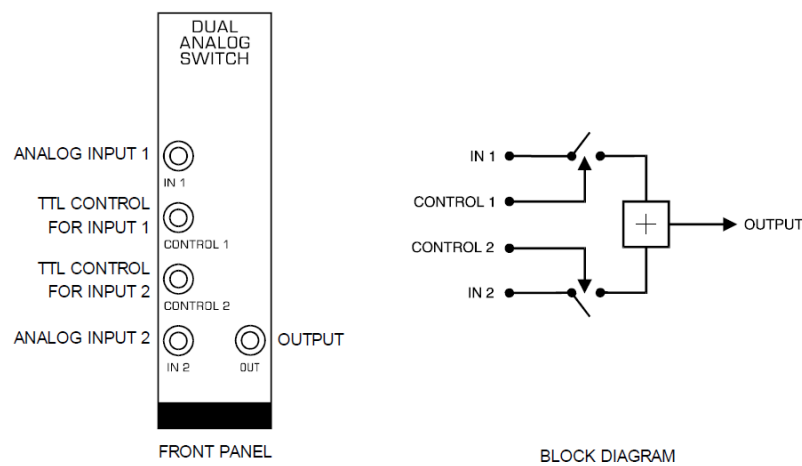


Figure 3. DUAL ANALOG Switch Module and Block Diagram

Lab Work

This experiment has two parts. The former examines the generation of a continuous phase frequency shift keyed (FSK) signal, while the latter demonstrates a general FSK modulator with higher degree of freedom.

Part I: Continuous Phase FSK, (CPFSK)

1. Set up the circuit for CPFSK modulation as shown in Figure 4. Make sure that the on board switch is set to FSK. Set the toggle switch to LO.
2. Plot the data sequence and the FSK modulated signal for a duration of 5 to 8 bits. Is the phase at the bit-transition continuous?
3. Plot the spectrum of the Continuous Phase FSK (CPFSK). Find f_1 and f_2 .
4. Set the toggle switch to HIGH and find f_1 and f_2 .

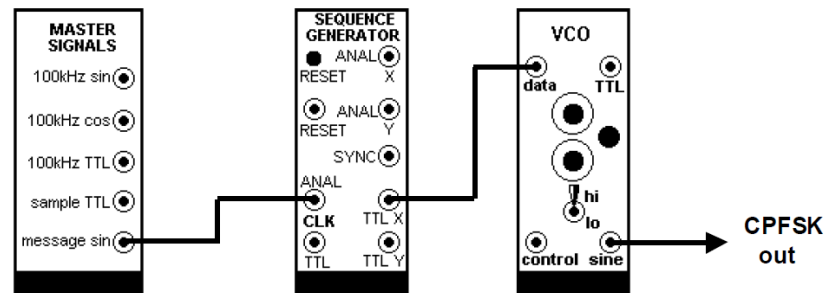


Figure 4: CPFSK generation

Part II: General Method of FSK Generation

A general method of FSK generation with higher degree of freedom is shown in Figure 5.

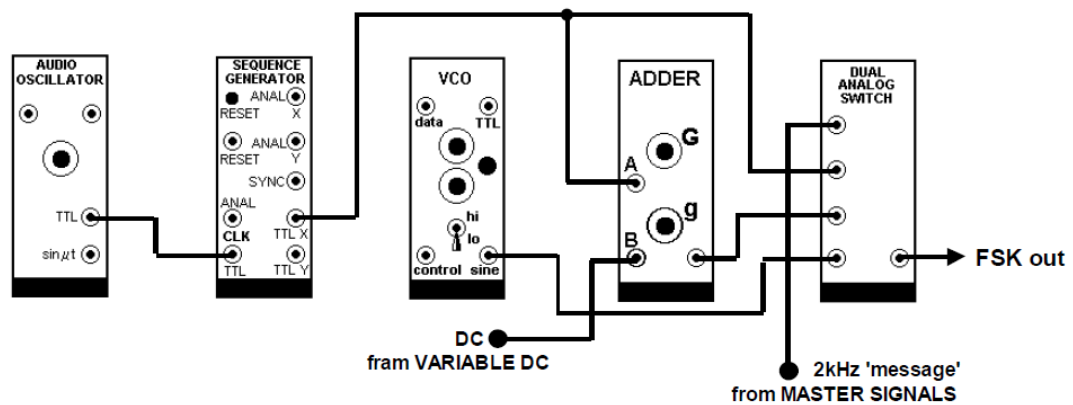


Figure 5: General Method of FSK Generation

1. Connect the circuit in Figure 5.
 - f_1 is 2 kHz obtained from the MASTER SIGNALS modules.
 - f_2 is tunable and obtained from VCO. Make sure that the on board switch SW2 is set to **VCO mode**. Also set the toggle switch to LO.
 - f_s , the bit rate of the message, is determined by the AUDIO OSCILLATOR.
 - The SEQUENCE GENERATOR produces a TTL (0, 5V) sequence of rate f_s .
 - DC shift of around -2.5V is needed to adjust the TTL level to $\pm V$ in order to generate FSK. Set the DC source to full counter clockwise (about -2.5V).

The system generates two carriers at f_1 and f_2 . The DUAL ANALOG SWITCH will alternate between the two according to the message sequences.

2. Vary f_2 . How does that affect the waveform and the spectrum of the modulated signal?
3. Set f_s to minimum then to maximum. Which setting generates a better modulated signal? Why?

4. Zoom in to a transition between any two bits. Is the phase continuous? Why? How does that affect the bandwidth?
5. Set the VARIABLE DC to zero. What type of modulated signal is generated? Why?
6. How can you use the above system to generate PSK?

Appendix A: LABORATORY REGULATIONS AND SAFETY RULES

The following Regulations and Safety Rules must be observed in all concerned laboratory locations.

1. It is the duty of all concerned parties who use any electrical laboratory to take all reasonable steps to safeguard the HEALTH and SAFETY of themselves and all other users and visitors.
2. Be sure that all equipment is properly working before using them for laboratory exercises. Any defective equipment must be reported immediately to the Lab. Instructors or Lab. Technical Staff.
3. Students are allowed to use only the equipment provided in the experiment manual or equipment used for senior project laboratory.
4. Power supply terminals connected to any circuit are only energized with the presence of the Instructor or Lab. Staff.
5. Students should keep a safe distance from the circuit breakers, electric circuits or any moving parts during the experiment.
6. Avoid any part of your body to be connected to the energized circuit and ground.
7. Switch off the equipment and disconnect the power supplies from the circuit before leaving the laboratory.
8. Observe cleanliness and proper laboratory housekeeping of the equipment and other related accessories.
9. Wear proper clothes and safety gloves or goggles required in working areas that involves fabrications of printed circuit boards, chemicals process control system, antenna communication equipment and laser facility laboratories.
10. Double check your circuit connections specifically in handling electrical power machines, AC motors and generators before switching "ON" the power supply.
11. Make sure that the last connection to be made in your circuit is the power supply and first thing to be disconnected is also the power supply.
12. Equipment should not be removed, transferred to any location without permission from the laboratory staff.
13. Software installation in any computer laboratory is not allowed without the permission from the Laboratory Staff.
14. Computer games are strictly prohibited in the computer laboratory.
15. Students are not allowed to use any equipment without proper orientation and actual hands on equipment operation.
16. Smoking and drinking in the laboratory are not permitted.

All these rules and regulations are necessary precaution in Electrical Laboratory to safeguard the students, laboratory staff, the equipment and other laboratory users.

Appendix B: Sample Report

Name: Smart & Neat ID: 200XXXX EE 370-5X Date: 22 August 20XX

Experiment # 3: Basic Operations on Sinusoids

Objectives:

- 1- Perform basic operations on sinusoids in the lab.
- 2- Demonstrate time and frequency domain representations of signals.
- 3- Realize practical limitations.

Lab Work and Results:

Part I: The Sum of Two Sinusoids:

The following block diagram was implemented in the lab

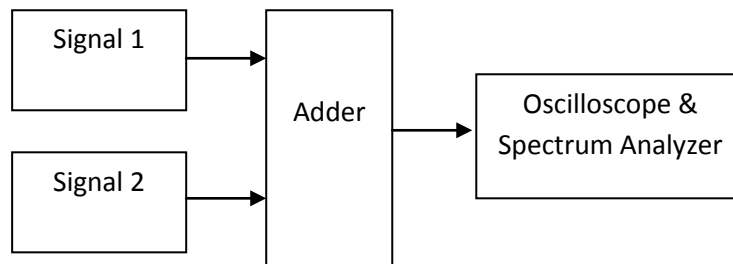


Figure 1: Block Diagram of Sinusoids Addition

Plots:

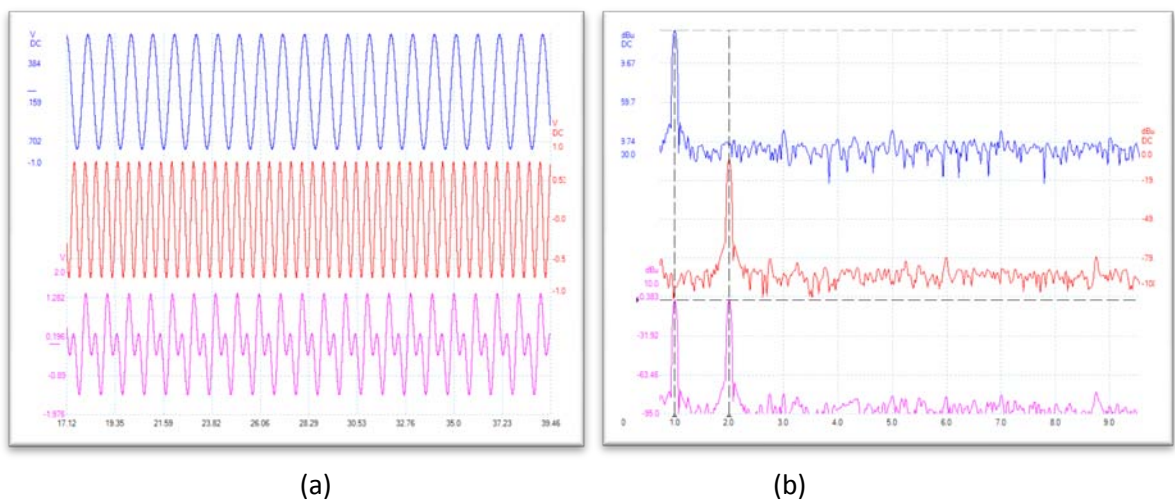


Figure 2: (a) Waveform and (b) Spectrum of the sum of 2 kHz and 3 KHz sinusoid

Discussion and Comments (Answers to questions posed in the lab manual and/or Lab Sheets):

- (1) The sum is period with period 1 msec.
- (2) The noise level in the spectrum plot is -40 dB
- (3) The sinusoids appeared in the spectrum as spikes at 2 and 3 kHz.
- (4)
- (5)

Part II: The Product of Two Sinusoids

The following block diagram was implemented in the lab

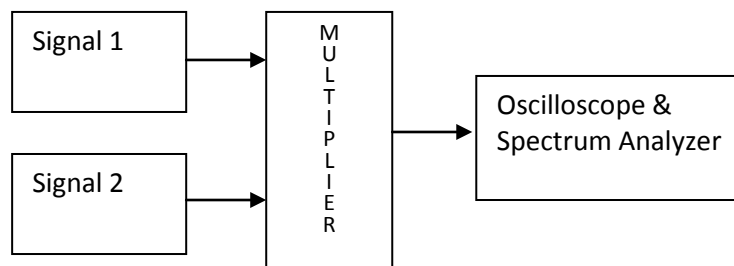


Figure 3. Block Diagram of Sinusoids Multiplier

Plots:

Place Waveform Here	Place Spectrum Here
---------------------	---------------------

Figure 4: (a) Waveform and (b) Spectrum of the product of 2 kHz and 3 KHz sinusoid

Discussion and Comments (Answers to questions posed in the lab manual and/or Lab Sheets):

- (1) The product is period with period 1/3 msec.
- (2) The noise level in the spectrum plot is -40 dB. No change to Part I
- (3) The product appeared in the spectrum as spikes at 1 and 5 kHz.
- (4)
- (5)

Post-Lab Work: (Answer to Questions in the manual)

Q1: If the difference is used instead of sum. How does that affect the spectrum? Why?

Answer: There will be no change to the magnitude spectrum. This is because

Q2: Write a MATLAB file to do the sum of two sinusoids. Evaluate the spectrum of the sum.

Answer:

```
% Name , ID
t= -1:0.01:1;
x=sin(2*pi*2000*t);
y=sin(2*pi*300*t);
z=x+y;
Z=abs(fft(fftshift(z)));
figure (1)
Plot(2)
figure (2)
plot(f,Z)
```

Conclusion:

In this experiment we implemented the basic operations on sinusoids. The experiment was smooth, and all results were as expected. This is my first time to see and quantify the noise of the electrical and electronic circuit. I had a little difficulty in working on the picoscope, but hopefully with time I will be better.