



Department of Electrical Engineering

EE-370

Communication Engineering I

Laboratory Manual

January 2019

PREFACE

Welcome to the EE370: Communication Engineering - 1 Laboratory. As course & Lab instructors, we do hope that you will enjoy performing the experiments. The set of experiments consists of software simulations and hardware experiments based on Telecoms Trainer. In the software experiments you will do a combination of MATLAB programming and SIMULINK modeling to simulate a communication block set so as to grasp and understand the main concepts of communication engineering. In the hardware experiments you will wire functional blocks of telecom trainer to build and test analog and digital communication systems. You will connect an oscilloscope and a function generator to the Telecoms Trainer as needed. These experiments are setup such that to relate what you are learning in class with what you carry out in the lab. The following outlines the general policies for the lab:

Grading Policy:

- 8% for Lab Reports
- 2% for Pre-labs
- 4% for Lab performance
- 6% for Lab Final

which will add-up to 20% for the Lab grade.

Lab Report:

- You must submit the report at the beginning of the new following lab.
- The report must have a cover page.
- It must have the following:
 1. Lab objective.
 2. Clear theoretical concepts & equations used.
 3. Description of the lab results or data collected and plotted.
 4. Conclusions.
 5. Appendices showing your source code and results of the simulations or plotted data.

Note: You can follow the sample report included below.

Important Notes:

- 1) Always bring an empty USB drive with you.
 - 2) Each unexcused absence will result in a grade of ZERO for that experiment.
 - 3) As per university rules, **three** unexcused absences will cost the student a DN grade.
 - 4) No make-ups are allowed.
 - 5) Any kind of cheating in is forbidden and will be treated according to the university rules. Just do your work independently.
-

University of Hail
Electrical Engineering Department

EE370
Communication Engineering I

Lab Report for Experiment No#

Place the title of the experiment over here

Prepared for

(Place the name of your lab instructor)

By

(Place your) Name:

ID No.

Section No.

(Place the date of the lab experiment)

- Objective: *Place the objective of the experiment here*
- Equations, general concepts and theoretical background: *Place the equations if they exist & your general fundamentals of the experiment here*
- Observations & discussion:
Write your comments independently regarding the simulations you 've done or the results you 've obtained through the whole experiment in details. They should be very clear, well written & logically convincing the reader of the report. You have to refer to the simulations or plots accordingly
- Conclusion:
Write a brief conclusion stating the main lessons that you've learnt from running the experiment.
DON'T REPEAT THE OBJECTIVE HERE OR DESCRIBE THE EXPERIMENT AGAIN!
- Appendix I: *(put Matlab Code here, or other relevant material)*
- Appendix II: *(put Matlab simulation figures, or other lab plots and results)*

LIST OF EXPERIMENTS

Experiment Number	Experiment Title	Page
1	Fourier Series & Fourier Transforms (Analysis and Synthesis)	
2	The Telecoms Trainer Hardware	
3	Amplitude Modulation – MATLAB Simulation	
4	DSB - SC Modulation and Demodulation - Hardware	
5	Amplitude Modulation (AM) and Demodulation - Hardware	
6	Frequency Modulation (FM) - MATLAB Simulation	
7	Frequency Modulation Using the Telecoms Trainer	
8	Sampling & Quantization - MATLAB Simulation	
9	Pulse Amplitude Modulation (PAM) and Demodulation - Hardware	
10	Pulse Code Modulation and Time Division Multiplexing - Hardware	
11	Communication Channel Effects	
12	Delta Modulation	
Appendix	LABORATORY REGULATIONS AND SAFETY RULES	

Experiment

1

Fourier Series & Fourier Transforms

MATLAB/SIMULINK Simulation

Objectives

Fourier analysis plays an important role in communication theory. The main objectives of this experiment are:

- 1) To gain a good understanding and practice with Fourier series and Fourier Transform techniques, and their applications in communication theory.
- 2) Learn how to implement Fourier analysis techniques using MATLAB/SIMULINK.

Pre-Lab Work

You are expected to do the following tasks in preparation for this lab:

- a) MATLAB is a user-friendly, widely used software for numerical computations. You learned MATLAB programming language in EE207. You should have a quick review of the basic commands and syntax for this software. The following exercises will also help in this regard.

Note: it is important to remember that MATLAB is vector-oriented. That is, you are mainly dealing with vectors (or matrices).

- b) SIMULINK is a GUI based block level system modeling and simulation package within MATLAB environment.

MATLAB exercise

- 1) Consider the following code: $Y=3+5j$
 - a. How do you get MATLAB to compute the magnitude of the complex number Y ?
 - b. How do you get MATLAB to compute the phase of the complex number Y ?
- 2) Vector manipulations are very easy to do in MATLAB. Consider the following:

```
xx=[ones(1,4), [2:2:11], zeros(1,3)]
xx(3:7) length(xx)
xx(2:2:length(xx))
```

Explain the result obtained from the last three lines of this code. Now, the vector \mathbf{xx} contains 12 elements. Observe the result of the following assignment:

```
xx(3:7)=pi*(1:5)
```

Now, write a statement that will replace the odd-indexed elements of \mathbf{xx} (i.e., $\mathbf{xx}(1)$, $\mathbf{xx}(3)$, etc) with the constant -77 . Use vector indexing and vector replacement.

- 3) Consider the following file, named example.m:

```
f=200; tt=[0:1/(20*f):1];
z=exp(j*2*pi*f*tt);
subplot(211) plot(real(z)) title('REAL PART OF z')
subplot(212) plot(imag(z)) title('IMAGINARY OF z')
```

- How do you execute the file from the MATLAB prompt?
- Suppose the file name was "example.cat". Would it run? How should you change it to make it work in MATLAB?
- Assuming that the M-file runs, what do you expect the plots to look like? If you're not sure, type in the code and run it.

Introduction to SIMULINK

The MATLAB® and Simulink® environments are integrated into one entity, and thus we can analyze, simulate, and revise our models in either environment at any point. Simulink can be started from within MATLAB.

To run Simulink, we must first start MATLAB. Make sure that Simulink is installed in your system. In the Command Window, type: simulink

Alternately, we can click on the Simulink icon in the toolbar (on the top bar on MATLAB's Command Window). Upon execution of the Simulink command, a window with the Commonly Used Blocks as shown in Fig. 1 will appear. In this window the left side is referred to as the Tree Pane and displays all Simulink libraries installed. The right side is referred to as the Contents Pane and displays the blocks that reside in the library currently selected in the Tree Pane.

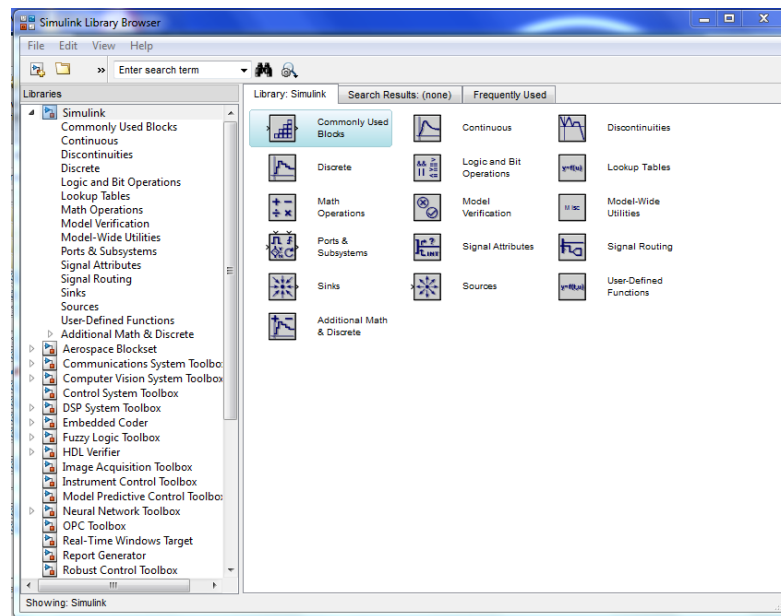


Figure 1.1: Simulink library browser.

A new Simulink model window as in Fig. 2 can be started in one of following the three methods.

- On the Simulink Library Browser, click on the leftmost icon shown as a blank on the top title bar.
- On the menu bar, click on **File** tab, select **New** and then select **Model**.
- Press Ctrl+N.

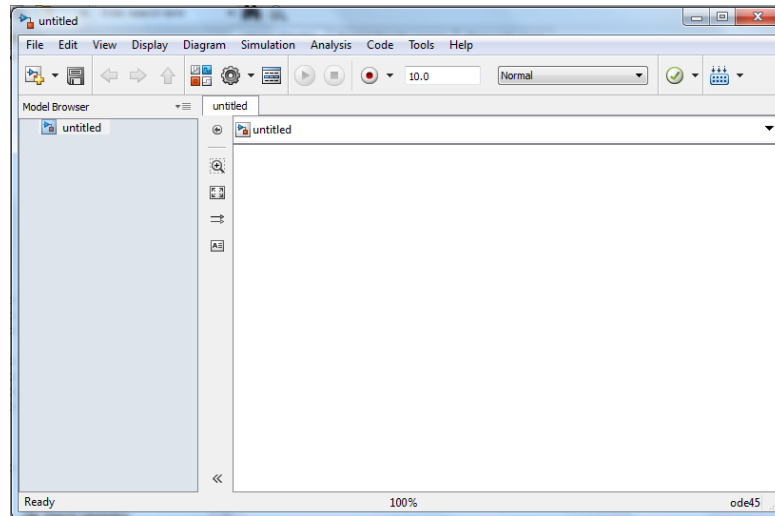


Figure 1.2: Model window.

You can build, simulate, and analyze a system on this model window by dragging and dropping relevant functional blocks from the Simulink library browser. It has the title 'untitled'. The title can be changed by saving with another name of your choice.

Review of Theory

Recall from what you learned in EE207 that the input-output relationship of a linear time-invariant (LTI) system is given by the convolution of the input signal with the impulse response of the LTI system. Recall also that computing the impulse response of LTI systems when the input is an exponential function is particularly easy. Therefore, it is natural in linear system analysis to look for methods of expanding signals as the sum of real sinusoidal or more compactly as the sum of complex exponentials. Fourier series and Fourier transforms are mathematical techniques that do exactly that!, i.e., they are used for expanding signals in terms of real sinusoidal signals or complex exponentials. When real sinusoidal signals are used we obtain a single sided Fourier series or transform. A double sided Fourier series or transform results with complex exponentials.

Fourier Series:

A Fourier series is the orthogonal expansion of periodic signals with period T_o when the signal set $\left\{e^{j2\pi n t/T_o}\right\}_{n=-\infty}^{\infty}$ is employed as the basis for the expansion. With this basis, any given periodic signal $x(t)$ with period T_o can be expressed as:

$$x(t) = \sum_{n=-\infty}^{\infty} x_n e^{j2\pi n t/T_o}$$

where the x_n s are called the Fourier series coefficients of the signal $x(t)$. These coefficients are given by:

$$x_n = \frac{1}{T_o} \int_0^{T_o} x(t) e^{-j2\pi n t/T_o} dt$$

This type of Fourier series is called the exponential Fourier series. The frequency $f_o=1/T_o$ is called the fundamental frequency of the periodic signal. The n^{th} harmonic is given by $f_n = n f_o$. If $x(t)$ is a real valued periodic signal, then the conjugate symmetry property is satisfied. This basically states that $x_{-n} = x_n^*$ where $*$ denotes the complex conjugation. That is, one can

compute the negative coefficients by only taking the complex conjugate of the positive coefficients. Based on this result, it is obvious to see that:

$$|x_n| = |x_{-n}|$$

$$\angle x_n = -\angle x_{-n}$$

Fourier Transforms:

The Fourier transform is an extension of the Fourier series to arbitrary signals. As you have seen in class, the Fourier Transform of a signal $x(t)$ denoted by $X(f)$, is defined

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt$$

On the other hand, the inverse Fourier Transform is given by:

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df$$

If $x(t)$ is a real signal, then $X(f)$ satisfies the following conjugate symmetry property:

$$X(-f) = X^*(f)$$

In other words, the magnitude spectrum is even while the phase spectrum is odd. There are many properties satisfied by the Fourier Transform. These include Linearity, Duality, Scaling, Time Shift, Modulation, Differentiation, Integration, Convolution, and Parseval's relation.

Lab Work

Part A: Fourier Series (With Simulink and MATLAB Programing)

- 1) Fourier synthesis with SIMULINK. In MATLAB, go to the command window and type SIMULINK. This will bring up the Simulink library browser. Open a new model window and build the system shown in Fig. 1.3 to synthesize example periodic waveforms using sinusoidal signals. Follow the steps given below.

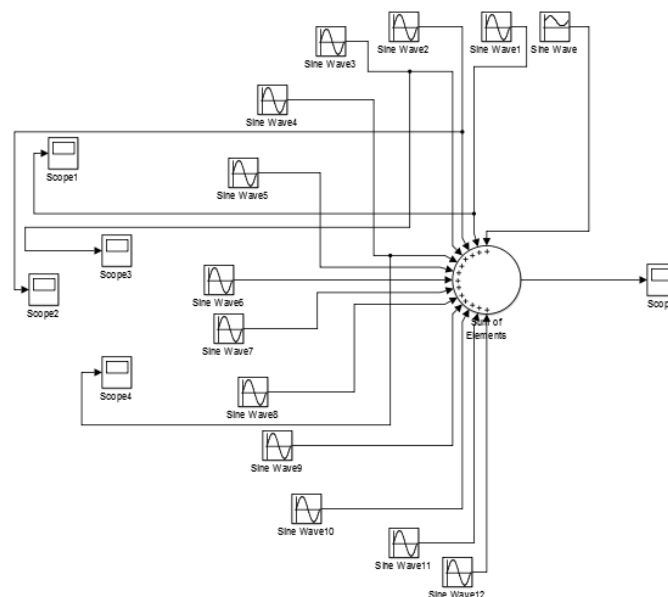


Figure 1.3: Simulink Model for Fourier Synthesis.

- a) Consider the periodic square waveform as shown in Figure 1.4. Show that the compact Fourier series coefficients are given by $C_n = \frac{\sin(n\pi/2)}{n\pi/2}$.

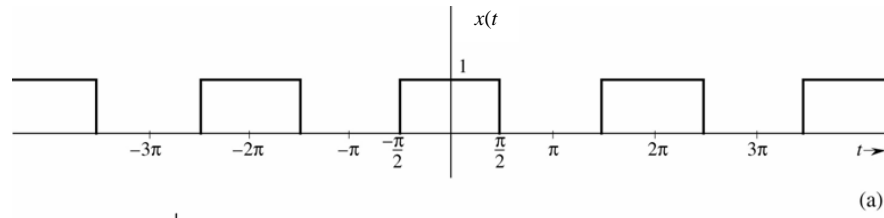


Figure 1.4 A square pulse periodic signal.

- b) Start with one Fourier component. Observe the output of summing block while adding more Fourier components (harmonics) to the input. The inputs and output can be observed by using oscilloscopes as you prefer.
- c) Repeat (a) and (b) for sawtooth waveform and fully rectified sine waveform.
- d) Report your observations. In particular, explain why the Fourier series for the square and sawtooth waves require many more harmonics than the rectified sine waves in order to get a close match between the FS and the original function?
- 2) Now consider a periodic signal $x(t)$ constructed by repeating the pattern $x(t) = e^{-t/2}$ where $t \in [0, \pi]$ (Refer to Example 2.3 of the textbook). Compute and plot the discrete magnitude and phase. Fast Fourier Transform (FFT) function in MATLAB (refer to the notes below for more details). For the expansion of the signal $x(t)$, the number of harmonics N_o to be used should be 32, the period T_o is π , and the step size is $t_s = T_o/N_o$. The output should be in two figure windows. The first window should contain $x(t)$ while the second window should contain both the magnitude and phase spectra versus a vector of harmonics indices (for example, n). You also need to include labels and titles in all plots. What can you observe from these plots?

Notes: In MATLAB, Fourier series computations are performed numerically using the Discrete Fourier Transform (DFT), which in turn is implemented numerically using an efficient algorithm known as the Fast Fourier Transform (FFT). Refer to the textbook (Sec. 3.9) for more theoretical details. You should also type: **help fft** at the MATLAB prompt and browse through the online description of the **fft** function.

Because of the peculiar way MATLAB implements the FFT algorithm, the **fft** MATLAB function will provide you with the positive Fourier coefficients including the coefficient located at 0 Hz. You need to use the even amplitude symmetry and odd phase symmetry properties of the Fourier series for real signals (see the introduction to Fourier series of this experiment) in order to find the coefficients for negative harmonics.

As an illustration, the following code shows how to use **fft** to obtain Fourier expansion coefficients. **You can study this code, and further enhance it to complete your work.**

```
Xn = fft(x,No)/No;
Xn = [conj(Xn(No:-1:2)), Xn];
Xnmag = abs(Xn);
Xnangle = angle(Xn);
k=-No/2+1:No/2-1
stem(k, Xnmag(No/2+1:length(Xn)-No/2))
stem(k,Xnangle(No/2+1:length(Xn)-No/2))
```

Useful MATLAB Functions: exp, fft(x,No), length(), conj, abs, angle, stem, figure, xlabel, ylabel,title.

Part B: Fourier Transform (with MATLAB Programming)

1) Consider the signals $x_1(t)$ and $x_2(t)$ described as follows:

$$x_1(t) = \begin{cases} t+1, & -1 \leq t \leq 0 \\ 1, & 0 < t \leq 1 \\ 0, & \text{elsewhere} \end{cases}$$

$$x_2(t) = \begin{cases} t, & 0 \leq t \leq 1 \\ 1, & 1 < t \leq 2 \\ 0, & \text{elsewhere} \end{cases}$$

Plot these signals and their relative spectra in MATLAB. What do you conclude from the results you obtained? Are there any differences?

You need to plot both time signals in one figure window. Similarly, you need to plot the magnitude and phase spectra for both signals in one figure window, i.e., overlapping each other. For the phase, display small values by using the **axis** command. You also need to normalize the magnitude and phase values, and you should include the labels, titles, grid, etc. Assume the x-axis to work as a ruler of units. Each unit contains 100 points and let the starting point to be at -5 and the last point to be at 5.

Notes: Similar to Fourier series, Fourier transform computations in MATLAB are easily implemented using the **fft** function. The following code illustrates that. Notice in particular the function **fftshift** is very useful for presenting the Fourier spectrum in an understandable format. The internal algorithm used in MATLAB to find the FFT points spreads the signal points in the frequency domain at the edges of the plotting area, and the function **fftshift** centers the frequency plots back around the origin. Notice the line 7 of the code is to compute the frequencies in Hz. The sampling frequency $fs=1/ts$ where ts is the sampling interval in seconds.

```
X = fft(x);
X = fftshift(X);
Xmag = abs(X);
Xmag = Xmag/max(Xmag); %Normalization
Xangle = angle(X);
Xangle = Xangle/max(Xangle);
F = [-length(X)/2:(length(X)/2)-1]*fs/length(X);
plot(F,Xmag), plot(F,Xangle);
```

2) Instructor will provide you a MATLAB data file named **Exp1Part4.mat**. You need to load that file as follows:

```
load Exp1Part4.mat
```

After you successfully loaded the file, go to the command window and type **whos** and press **Enter**. You will notice three stored variables fs (sampling frequency or $1/ts$), t (time axis vector) and m (speech signal). These correspond to a portion of speech recording.

The next step is to plot the speech signal versus the time vector t . In the same figure window and a second window panel, display the magnitude spectrum of m (call it M). What is the bandwidth of the signal? What can you notice in terms of the speech signal? In order to play the signal properly, make sure that the speakers are turned on and write the following MATLAB statement:

```
sound(m,fs)
```

Experiment
2

The Telecoms Trainer

Hardware Experimentation

Objectives

The objective of this lab is to give the students a first introduction to the Telecoms Trainer, which will be extensively used in subsequent experiments.

- The Lab will enable the students to identify the different modules of the Telecoms Trainer, and understand their respective functions.
- The configuration of the different function blocks to implement analog and digital communication systems is introduced.
- Some testing and experimentation with the Telecom Trainer, Function Generator and Oscilloscope equipment is also included.

Pre-Lab Work

In preparation for this lab, do the following tasks:

- 1) Consider a linear time-invariant (LTI) system such as a filter, etc. Explain how to characterize the input-output relationship for such a system. Give the results both in time-domain and in frequency-domain (using convolution, frequency response, etc).
- 2) The function generator that you will be using in the lab generates many special functions such as sine waves, rectangular pulses, triangular signals, etc. How do you classify these signals: as “energy” signals or “power” signals? Explain why.
- 3) Assume now for simplicity that these signals are truly periodic, with a given period T_0 . Give the Fourier series for each type of the basic waveforms: sinusoidal, rectangular and triangular.

Overview

The conventions of signal input and output of the Telecoms Trainer board is as explained in Figure 2.1. Study this carefully.

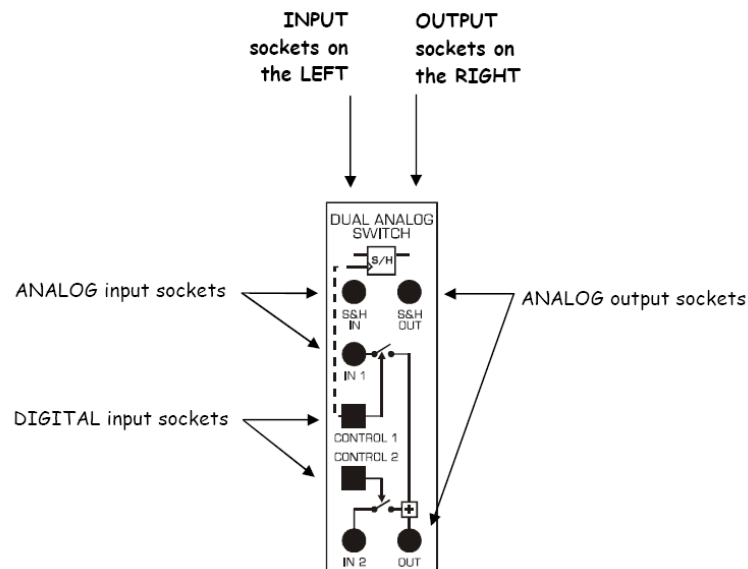


Figure 2.1 Signal input and output conventions of the modules of the Telecoms Trainer.

The Telecoms Trainer includes several modules, including the followings. Refer to Figure 2.2.

1). **ADDER** - There are two adders. The first adder can be used to add two analog input signals $A(t)$ and $B(t)$ with adjustable gains G and g . The resulting sum $GA(t)+gB(t)$ is presented at the output. The gains G and g are in the range $(0 - 2)$ and are inverting. The bandwidth is approximately 600kHz. The second adder has input with fixed gains of unity ($\times 1$).

2). **MULTIPLIER** - This module is used to multiply two signals in real-time. There are three multipliers. The first MULTIPLIER includes both AC and DC coupled inputs. AC coupled inputs remove the DC components of the inputs before multiplication. Two analog input signals $X(t)$ and $Y(t)$ may be multiplied together. The resulting product is scaled by a factor of approximately 1. The other two MULTIPLIER modules have DC coupled inputs.

3). **TWIN PULSE GENERATOR** - This module accepts a digital-level clock signal and outputs two consecutive digital pulses. The width of the pulses and the time delay between the two pulses can be controlled by the user via the front panel WIDTH and DELAY control knobs. The pulses repeat at the occurrence of each CLK input signal. The pulse WIDTHs can be varied from 5us to 40us. The delay between Q1 and Q2 can be varied from 40us to 350us.

4). **DUAL ANALOG SWITCH** - Two identical analog switches are controlled by digital, 0 to 5V level signals. The outputs of the two switches are added internally and presented at the output of the module. This module also provides an additional and independent function called "S&H", or Sample-and-Hold". The Sample-and-Hold is the first step in converting "analog" signals to "digital" signals.

5). **NOISE GENERATOR** module provides a wide source of electrical noise. Three different amplitudes of noise level are provided. Electrical noise is usually added to signals before they pass through the CHANNEL MODULE to model the effects on a signal due to real-world disturbances. Examples of disturbances are other nearby signals, static, nearby electrical machines, electrical transformers and so on. Characteristics: "white" noise in the frequency range 10Hz to 240kHz, Maximum level approximately 4.8Vrms, Attenuator steps 0dB (approx 4.8Vrms), -6dB (approximately 2.4Vrms) and -20dB (approximately 0.48Vrms).

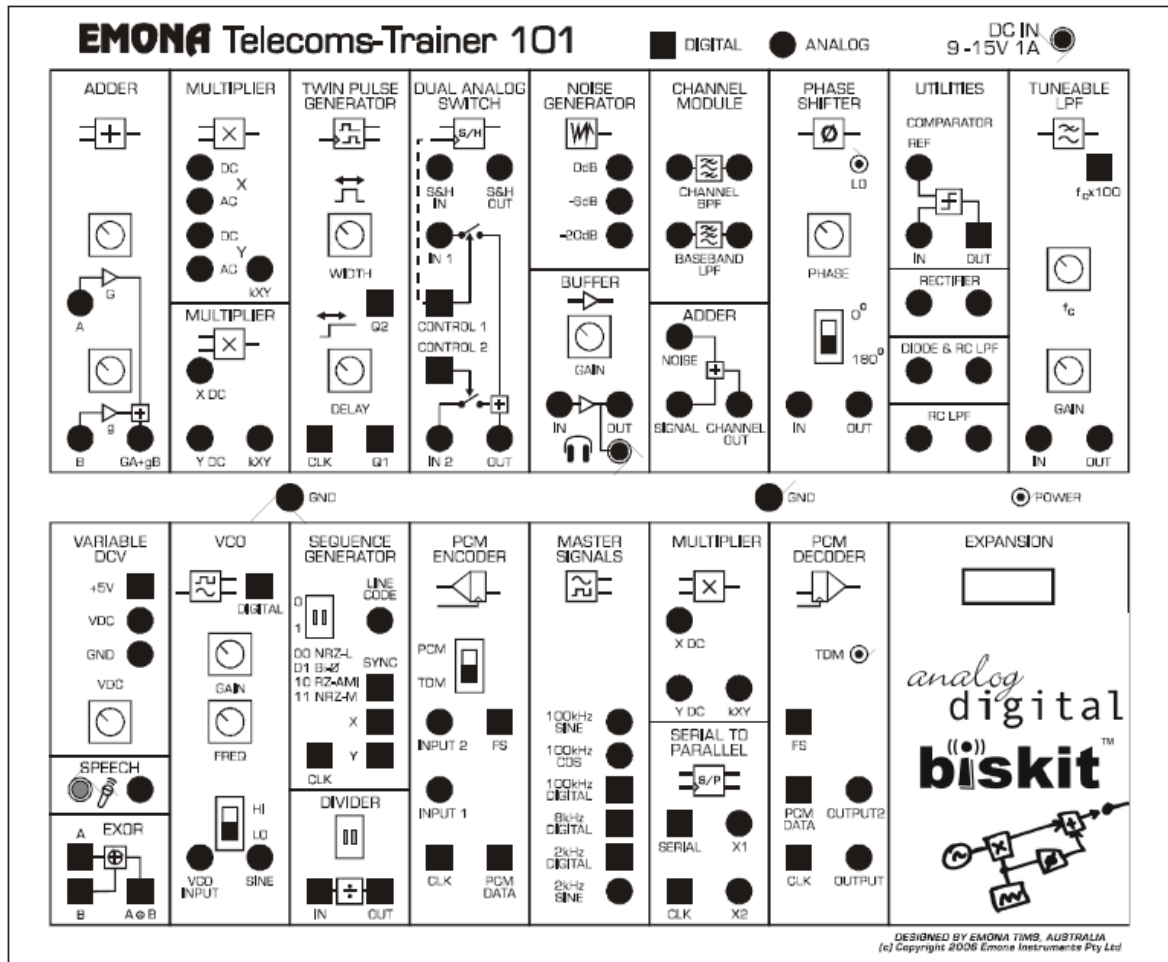


Figure 2.2 Layout of the Telecoms Trainer Board.

6). BUFFER - a variable gain amplifier: Bandwidth DC to approximately 600kHz, Gain 0.2 to 10. The BUFFER may be used to amplify (increase) small signals or attenuate (reduce) large signals. The BUFFER has a gain, (or amplitude), control on the front panel labeled GAIN. Care should be taken to ensure that later modules are not overloaded due to excessive gain. Overload will not cause any damage but it means non-linear operation (distortion), which is to be avoided in analog systems. If overload occurs, turn the gain control counter clockwise.

7). CHANNEL MODULE - There are two channel models in this module. Both channel modules take signals which are at the analog signal levels of 4Vpp (peak to peak voltage).

(a) A BANDPASS wireless channel such as a satellite link, radio or TV channel using a Band Pass Filter (BPF). This is a 6th order Chebychev filter with 0.1dB ripple. The BANDPASS channel is used for signals modulated at the carrier frequency of 100kHz. Thus the filter is centered at 100kHz with a bandwidth of 24kHz (approximately in the range 88kHz - 112kHz) with mid band gain of 1. Stopband at -35dB has a bandwidth of 140kHz (approximately in the range 30kHz - 170kHz);

(b) A base band channel such as a wire in a telephone link, cable in a modem link, or hardwired link between computer networks is modeled using a Low Pass Filter (LPF). BASEBAND channels modeled using a LPF is used for data signals which are not modulated. The cut-off frequency of the filter is 1.6 kHz. The mid band gain is 0.9. This is a 4th order Butterworth filter.

8). PHASE SHIFTER - introduces a phase shift (delay in time) between its input and output. The amount of phase shift is controlled by the user, via the front panel PHASE knob and 0/180 degree switch. This variable PHASE SHIFTER is capable of varying the magnitude of the phase shift through 340 degrees in two steps. The 180 degree switch selects the step or region of interest. If the input is $\text{COS}(\omega(t))$, then the output is $\text{COS}(\omega(t) - F)$, where F lies between 0 and 180 degrees. Although the PHASE SHIFTER will operate from a few hertz up to 200kHz it has been optimized to operate in the neighborhood of two frequencies: around 100kHz in the HI range and around 2kHz in the LO range. Frequency range selection is automatically detected and displayed by the LED. The gain through the PHASE SHIFTER is essentially unity for all phases, but note that the amount of phase shift is a function of frequency. This is NOT a wideband phase changer: thus all the frequency components of a complex signal's spectra are not shifted by the same phase.

9). UTILITIES - This module houses four independent functional blocks:

(a) A signal COMPARATOR with digital-level output. The COMPARATOR is used for squaring analog waveforms. The COMPARATOR's threshold level may be set as required by applying a DC voltage to the REF input terminal. **Important:** the REF input must never be left unconnected. The REF input may be connected to GROUND, VARIABLE DC or any other signal source.

Operating Range > 100kHz, TTL Output Risetime 500nsec (typically).

(b) Precision half-wave RECTIFIER. Bandwidth DC to 100kHz (approximately).

(c) Simple diode and single pole, audio range, Resistor-Capacitor (RC) Low Pass Filter (LPF).

-3 dB bandwidth is 2.6 kHz (approximately).

(d) Single pole, audio range, RC Low Pass Filter (LPF). -3dB bandwidth is 2.6 kHz (approximately).

10). TUNEABLE LPF - This module allows the user to vary which frequencies are passed by adjusting the front panel f_c control knob. f_c is the cutoff frequency of the Low Pass Filter (LPF). The cutoff range is from 300 Hz to 16 kHz. The amplitude of the output signal can be controlled via front panel GAIN control.

Note: A digital-level square wave signal is output which has a frequency 100 times higher than the selected cutoff frequency. Measuring the frequency of the digital signal and dividing the frequency by 100 will give the user an instantaneous reading of the TUNEABLE LPF cutoff frequency.

BASIC SPECIFICATIONS: Filter Range 600 Hz to 12 kHz, Filter Order 8th order, Elliptic Stopband Attenuation > -50dB at $1.4 f_c$ and Passband Ripple < 0.5dB, Gain Control 0 to x 1.6.

11). VARIABLE DC module - a stable, bipolar DC voltage source and a stable +5V DC output. The DC voltage output varies from about -2.5V when the control is fully counter clockwise through zero to +2.5V when control is turned fully clockwise. If greater resolution or wider range is required, then the BUFFER module can be used in conjunction with the VARIABLE DC module. The +5V output serves as a logical HIGH for digital inputs, such as the EXOR. **The +5V output is a signal output only and is not to be used as a power supply.**

BASIC SPECIFICATIONS: VDC Terminal Voltage Range $\pm 2.4V$ DC (+/-10%), Output Current < 5mA, +5V

12). Voltage Controlled Oscillator (VCO) - a variable output frequency signal source, where the user can set the value of the output frequency via the FREQ front panel control knob. The VCO module's output frequency can also be controlled by an external voltage applied to the VCO INPUT terminal. The sensitivity of the VCO INPUT signal is controlled by the GAIN front panel control. The VCO outputs both an analog-level and digital-level signal of the same frequency. The VCO module has two operating frequency ranges, HI and LO. HI covers the

60kHz to 140kHz range – the trainer board’s “carrier” frequency range. LO covers the 1kHz to 15kHz range – the message or “baseband” range.

BASIC SPECIFICATIONS: Frequency Ranges $1\text{kHz} < \text{LO} < 17\text{kHz}$; sinewave and digital-level $60\text{kHz} < \text{HI} < 140\text{kHz}$; sinewave and digital-level, Input Voltage $-3\text{V} < \text{VCO INPUT} < 3\text{V}$, GAIN $G.V_{in} : 1 < G < 2$

13). SEQUENCE GENERATOR - outputs typical “digital” messages. The advantage of using a SEQUENCE GENERATOR as a digital message source is that it outputs a stable and repeating set of digital data, which means that it is easier to observe the data in experiments. The SEQUENCE GENERATOR module outputs two different digital sequences – digital sets of data – labeled “X” and “Y”. The SYNC terminal outputs a pulse at the start of the X data sequence only. The data needs to be clocked by a signal at the CLK input. Typically this would be either the 8kHz or 2kHz DIGITAL signals from MASTER SIGNALS module, the DIGITAL output of the VCO module.

CLK: An external clock signal, CLK, must be provided to operate the SEQUENCE GENERATOR. This must be a digital-level signal. CLK only accepts digital-level signals.

SYNC: The SYNC terminal outputs a digital pulse at the beginning of each X sequence. SYNC is a digital-level signal.

X (short) and Y (long): The length of sequence X is 31 bits. The length of sequence Y is 255 bits. X and Y are digital level-signals.

BASIC SPECIFICATIONS: Input Clock Range TTL 1Hz to 100kHz, Number of Sequences 2, X and Y, Sequence Lengths $X = 31$ bits, $Y = 255$ bits, Sync indicates start of sequence X.

14). The DIVIDER is used to divide digital logic clock signals. Four division factors are available which can be set by two switches. The DIVIDER only accepts digital logic-level signals, of 0 to 5V.

Two switches are used to select the division factors of -1 (inversion), divide-by 2, divide-by 4 and divide-by 8. See the table below:

Table 2.1

DIVIDER	SWITCH	
	LEFT	RIGHT
-1 (inversion)	UP	UP
Divide by 2	UP	DOWN
Divide by 4	DOWN	UP
Divide by 8	DOWN	DOWN

BASIC SPECIFICATIONS: Input digital signals, 0 to 5V; Output digital signals, 0 to 5V; Maximum Input 1MHz

15). Pulse Code Modulation (PCM) ENCODER is the fundamental building block in telecommunications which converts sound, music and speech into digital data. The heart of a PCM ENCODER is known as an analog-to-digital converter. The Telecoms Trainer PCM ENCODER is an audio frequency analog-to-digital converter which outputs the digitized data in serial, digital-level PCM format. Since the digital data comes out in a serial format – one bit after another from a single output – another signal, known as the FS or FRAME SYNCHRONIZATION pulse is required. The FS pulse marks the start of each byte of digital data. The PCM ENCODER has two modes which can be selected at the PCM/TDM front panel switch: PCM mode digitizes only one signal at a time, presented at the INPUT 1 socket. TDM

mode (Time Division Multiplex) digitizes two signals simultaneously, presented at the INPUT 1 and INPUT 2 sockets. The PCM ENCODER always requires a digital clock signal, either the 100 kHz DIGITAL or 8 kHz DIGITAL signals, or the VCO module's DIGITAL output socket. (More on this module appears at a later experiment.)

The PCM ("Pulse Code Modulation") DECODER performs the reverse function of the PCM ENCODER. The PCM DECODER module converts digital data back to sound, music or speech. The heart of a PCM DECODER is known as a digital-to-analog converter.

16). MASTER SIGNALS - provides a set of useful analog and digital signals. Three of the signals have a frequency of 100kHz, which in the context of the Telecoms Trainer is equivalent to radio or wireless frequencies used in telecommunications. The other signals are lower in frequency and referred to in telecommunications as "baseband". In the context of the Telecoms Trainer this means that they can be heard with the human ear. These baseband signals can be used in place of a human voice or music in modulation and coding experiments.

- 100kHz - CARRIER or WIRELESS signals: The Telecoms Trainer models telecommunications systems at frequencies that are lower than real-world telecommunications systems, so that students can easily and safely view the advanced modulation and coding schemes used in modern telecommunications. The "carrier" frequency chosen is 100kHz. A 100kHz sinusoidal and a 100kHz cosine signal are provided. A digital-level, 0V to 5V, square wave signal is also provided to allow more complex experiments to be implemented.
- 8kHz – SAMPLE CLOCK signal: In modern telecommunications, analog (voice, music, etc) signals need to be converted to digital signals using a process called "sampling". The Telecoms Trainer allows students to investigate the principles of sampling, hence an 8kHz digital, square wave signal is provided for this purpose.
- 2kHz – ANALOG MESSAGE signal: An analog signal at a frequency of 2kHz is within the range of the human voice and can be heard by the human ear. Therefore it is useful to provide an analog signal which can be used in modulation and coding experiments. This signal is easily viewable and provides very repeatable displays for student investigation. A digital, square wave version of the 2kHz signal is also provided. This helps students in contrasting analog and digital signals, and also assists in broadening the range of possible fundamental telecommunications experiments.

SYNCHRONIZATION of MASTER SIGNALS – It is important to note that a special feature of the MASTER SIGNALS module's signals is that they are synchronized so that students will always obtain clear and stable textbook-like displays.

BASIC SPECIFICATIONS - Output Frequencies carrier: 100kHz in quadrature and a third digital signal sample clock 8.333kHz (sub-multiple of the carrier) message: 2.083kHz sinusoidal and digital, (sub-multiple of the carrier), Output Levels 4V pp, analog (+/-5%), Digital level, 0V to 5V

17). LINE CODER - Line codes are digital data which has been converted to bipolar voltage levels in a specific manner, following the rules of the line code selected. Line codes are utilized to assist in minimizing distortion and errors due to external effects. For example, before digital data is transmitted along a computer network cable, the digital, 0V to 5V, data is line code encoded to bipolar, say -2V to +2V, data which is optimized for that environment. The Telecoms Trainer offers 4 Line Codes as part of the SEQUENCE GENERATOR module. The LINE CODE ENCODER converts the "X" data stream of the SEQUENCE GENERATOR module into one of the selected line codes.

The line codes are selected via front panel switch. NRZ-L : Non Return to Zero – Level; Bi- ϕ : Bi-phase Line, otherwise know as Manchester; RZ-AMI : Return to Zero – Alternate Mark Invert; NZR-M : Non Return to Zero – Mark

Table 2.2

LINE CODE	SWITCH	
	LEFT	RIGHT
NRZ-L	UP	UP
Bi- ϕ	UP	DOWN
RZ-AMI	DOWN	UP
NRZ-M	DOWN	DOWN

17). SERIAL TO PARALLEL performs a basic digital logic function, breaking a single serial digital data stream into two parallel streams of data. This module is a fundamental building block used primarily in digital communications experiments. The two inputs, SERIAL data and digital CLK, are digital level signals. The two outputs X1 and X2 are square wave signals at bipolar voltages of -2V and +2V.

SERIAL input accepts any serial digital data stream such as the X or Y output of the SEQUENCE GENERATOR module. A digital-level CLK signal must always be provided. If the SEQUENCE GENERATOR module's X or Y data signals are being used, then the SEQUENCE GENERATOR module's CLK must also be patched to the SERIAL TO PARALLEL module's CLK input. Outputs X1 and X2 are digital signals at bipolar voltage levels of -2V and +2V. OPERATION Alternate bits from the SERIAL input are presented sequentially to outputs X1 and X2. This is also known as a "multiplexing" action.

BASIC SPECIFICATIONS: Inputs SERIAL digital-level data; CLK is the digital-level clock signal; Maximum CLK Rate approx 100kHz; Outputs bipolar parallel data output

Lab Work

- Use the oscilloscope to measure the range of amplitudes and frequencies that are available at the analog and digital output terminals of the VCO in LO and HI positions.
 - Measure the output voltage (V_{pp} or V_{amp}).
 - Turn the GAIN control knob and observe the output. Does the output waveform varies?
 - Using the frequency control knob ((FREQ) measure the minimum and maximum frequencies available at the analog as well as the digital output terminals. Record the results in Table 2.3.

RANGE		LO	HI
Minimum Frequency (kHz)	Analog		
	Digital		
Minimum Frequency (kHz)	Analog		
	Digital		

Table 2.3

- Use the benchtop Function Generator to apply a sinusoidal signal with frequency 300 KHz and 2V p-p to the input of the BUFFER. Use the appropriate equipment to measure the voltage gain and phase shift of the resulting output signal.
 - Set the FG output to 1Vamp, ,300kHz. Measure the minimum and maximum Gain and Phase Shift.
 - Vary the input frequency from 0.1kHz
- In this part, you are expected to measure the amplitude transfer function $H(f) = V_{out}/V_{in}$ of the BANDPASS and BASEBAND channels that are modeled respectively by BPF and a LPF. The procedure is as follows. First apply a sinusoidal signal from the external function generator to the input port of the filter. Vary the frequency of this signal in the range of each filter. Use a step size of 5KHz for BPF and a step size of 0.1kHz for LPF. Measure V_{out} at each frequency (make sure the input signal amplitude is at the fixed value in order to simplify the computation of $H(f)$). You should also reduce the frequency step size around the cut off frequencies of the filters.

4. From the measured transfer function $H(f)$, determine the center frequency of the filters and their 3-dB bandwidths. Recall that the 3dB cutoff frequency is defined as the frequency for which the output amplitude gain drops to $1/\sqrt{2}$ of its maximum value.

Note: Recall that the signal power is proportional to the squared amplitude. Hence, a $1/\sqrt{2}$ -factor in signal amplitude corresponds to a $1/2$ -factor in signal power, which in terms of dB units corresponds to 3dB. This explains why we use the term 3dB bandwidth in the above part.

5. Now, change the input signal shape to apply a rectangular pulse stream from the external signal generator. Set the frequency of this signal in the range of the central frequency with $V_{pp} = 1V$, and observe the output signal. Record your observations and explain your results. What happens if you change the rectangular signal frequency?
(Hint: refer to the pre-lab exercise related to the Fourier series expansion of the rectangular signal).

Homework Questions

- Q1.** If a message signal $\cos(2,000 \pi t)$ and a carrier signal $\cos(20,000\pi t)$ are fed to a mixer (multiplier with a BPF in cascade), what is the resulting output signal? Assume that the mixer passes the higher frequency term and suppresses the lower one.
- Q2.** Consider Part 5 above and imagine the case where the input rectangular pulse train had twice the central frequency. Explain, in a simple and precise way, the kind of output you expect to get from this filter.

Experiment

3

Amplitude Modulation

MATLAB Simulation

Objectives

The main objectives of this experiment are:

- 1) To gain a clearer understanding about double-side band suppressed carrier (DSB-SC) and amplitude modulation (AM)
- 2) To learn how to simulate modulation/demodulation systems for DSB-SC and AM using MATLAB for synthetic & real signals (such as speech).

Pre-Lab Work

- 1) Read the relevant material in your textbook (Chapter 4).
- 2) Using MATLAB, perform the following:
 - a. $X=[0,1,2,3,4,5]$
 - b. $Y=[1,2,3,4,5,6]$

Now multiply X by Y using two ways. The first one is the usual MATLAB multiplication (star $X*Y$) and the other one is what is called as point-wise array multiplication (a dot followed by a star $X.*Y$). What is the difference between the two?

- 3) Using MATLAB, generate a vector $t = [0:0.001:1]$. Then generate $m = \cos(2*\pi*t)$ and $v = \cos(4*\pi*t)$. Plot m , v , and the product $x=mv$ versus t . Are you going to use multiplication between matrices or vectors that are representing functions?

Introduction

Amplitude modulation (AM) is the family of modulation schemes in which the amplitude of a sinusoidal carrier is changed as a function of the modulating message signal. This type of modulation schemes includes many variants, such as double-sideband suppressed carrier (DSB-SC), single-sideband (SSB), conventional AM, and vestigial-sideband (VSB). Refer to your textbooks for ample details on amplitude modulation techniques. In this lab, we focus in particular on DSB-SC and conventional AM.

DSB-SC AM:

In DSB-SC AM, the amplitude of the modulated signal is proportional to the message signal. The time-domain representation of this scheme is given by:

$$y(t) = A_c x(t) \cos(2\pi f_c t)$$

where $A_c \cos(2\pi f_c t)$ is the carrier signal with a carrier frequency f_c , and $x(t)$ is the message signal. The transmission bandwidth is twice the bandwidth of the message signal.

Conventional AM:

AM is similar to DSB-SC, but it also includes a pure carrier (non-modulated) component in the transmitted signal. The message signal $x(t)$ is replaced by $[1 + \mu x_n(t)]$, where $x_n(t)$ is the normalized message signal and μ is the modulation index. Therefore the AM signal will be:

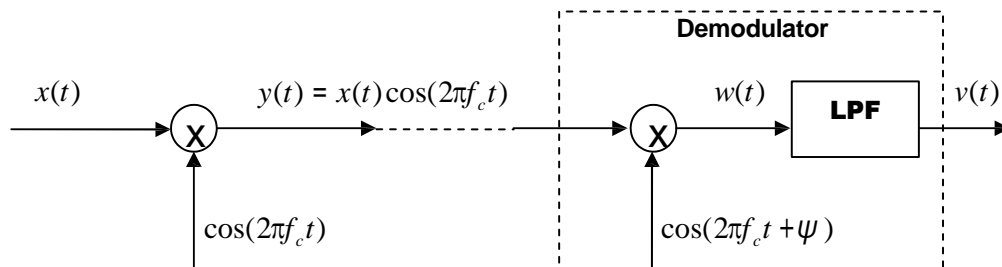
$$y(t) = A_c [1 + \mu x_n(t)] \cos(2\pi f_c t)$$

The existence of the sinusoidal component makes the AM scheme less economical in terms of power utilization as compared to the DSB-SC scheme. However, the demodulation of AM signals is much cheaper than the demodulation process of DSB-SC signals. The conventional AM demodulation process is simply done by employing envelope detectors.

For the bandwidth, the AM signal has the same transmission bandwidth as the DSB-SC transmission bandwidth.

Lab Work

- 1) Use MATLAB to simulate the following block diagram



Assume $\psi = 0$ and let $x(t) = \cos(2\pi 2000t)$. Use a carrier frequency of $f_c = 20$ kHz. Plot $x(t)$, $y(t)$, $w(t)$, and $v(t)$, and their magnitude spectrums each

in a two-panel figure. Define the time vector t as $[0:200]*ts$ where ts is the step size given by $ts=1/(10*fc)$.

At the receiver end, you need to design a Low Pass Filter (LPF). In MATLAB, you can use a type of filters known as *Butterworth* filters. For example, you can design a given filter with some order n and cut-off frequency f_c which is typically normalized in Matlab and given by $2f_c t_s$ (where t_s is the sampling period). To obtain the filter coefficients, the statement will be: `[num,den]=butter(n, 2*f_c*t_s)`, where **num** and **den** are the numerator and denominator coefficients of the rational function representing the analog filter. You can use $n = 5$ for example. Once you obtained these coefficients, you can use the Matlab function **filter** to filter the signal $w(t)$ using the designed LPF. That is, `v = filter(num,den,w)`.

Refer to the additional notes below for further discussion on how to use filters in Matlab.

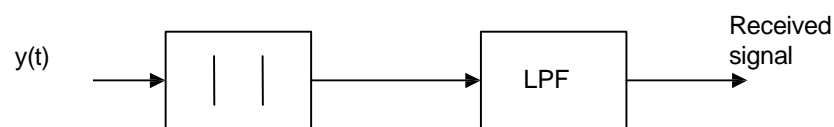
Useful MATLAB Functions: `cos`, `fftshift(fft())`, `butter`, `filter`, `abs`, `plot`, `subplot`, `figure`, `xlabel`, `ylabel`, `title`.

- 2) Repeat Part 1 with 2 different values for the receiver phase offset: $\psi = \pi / 2$ & π .

What do you notice at the receiver end? Is there any difference between the recovered signal here and the one obtained in Part 1? Why is that? And what is the solution to this problem?

Repeat Part 1 by making $y(t) = A_c (1 + \mu x_n(t)) \cos(2\pi f_c t)$ where μ is the modulation index of the AM wave, A_c is the carrier amplitude (set it equal to 4), and $x_n(t)$ is the normalized version of $x(t)$. Set it to be 0.5 (50% modulation).

- 3) At the demodulator, you can implement the functionality of the simple envelope detector that you studied in class (built with a diode, a capacitor and resistor) by using simple MATLAB code to produce full-wave rectification (absolute value function), followed by low-pass filtering. This is illustrated in the following block diagram. You need to think about setting the appropriate cutoff frequency for the LPF. In addition, you can also add a mechanism to remove the DC component from the signal



- 4) Repeat Part 3 by letting the modulation index equal to 1.2. What will happen to the received signal? Explain.

- 5) Load the file called `Exp3Part5.mat`. This data file contains:

- A vector called **ms**, which is a speech signal sampled with $t_s = 1/96E3$ s.
- A vector called **t** that represents time.

With a carrier of 24kHz, transmit and receive `ms` using the AM system in part 3 with

$\mu = 0.5$ & 1.5 . For both cases show the following:

- In one figure with two panels, the time and frequency domain representations of the modulated waves
- Listen to **ms** by typing (**sound(ms,96E3)**, **pause**, then press **Enter** to continue). Also listen to the received signal **v** by typing **sound(v,96E3)**. Comment on the differences between the two signals.

Note: in order to be able to see the spectrum of the signal **ms**, after plotting the magnitude spectrum of **ms**, (denoted by **Ms**) vs. **f**, type:

axis([-4E3,4E3,0,max(|Ms|)])

Useful MATLAB Functions: load, sound, pause, axis.

Additional Notes (Filters in MATLAB):

To further understand how to use filters in Matlab, recall from EE207 that the rational transfer function of a filter can be expressed in terms of the Laplace variable s by:

$$H(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0}$$

where the a 's and b 's define the transfer function coefficients. These coefficients completely characterize the filter response. Matlab returns these two vectors as a result of designing a filter with a certain type (e.g., Butterworth, etc.) and cutoff frequency. For example, with $\text{num}=[b_m, b_{m-1}, \dots, b_0]$ and $\text{den}=[a_n, a_{n-1}, \dots, a_0]$, the filter design is done by: $[\text{num}, \text{den}] = \text{butter}(n, 2 f_c t_s)$.

Notice also that the filter order n is important to specify. For example, it is shown that for these Butterworth-type filters, as the filter order increases the filter response will approach that of an ideal "brick-wall" response.

Experiment
4

DSB - SC Modulation and Demodulation

Hardware Experimentation

Objectives

The goal of this lab is to allow the students to experiment with DSB-SC Modulation & demodulation using Telecoms Trainer hardware. The detailed objectives include the following:

- 1) Implementing DSB-SC modulator and demodulator with coherent detection.
- 2) Investigation of the effect of amplitude on DSB-SC modulation and demodulation.
- 3) Investigation of phase and frequency mismatch on the demodulation.

Pre-Lab Work

Answer the following questions:

1. Explain the difference between coherent and non-coherent detection of AM signals
2. Can DSB-SC be detected non-coherently with an envelope detector? Explain why.

Overview

Refer to your textbook and to the previous experiment for an overview of DSB-SC modulation and the related mathematical expressions. Also refer to Experiment 2 for a description of the Telecoms Trainer.

Basically, Amplitude Modulation (whether it is DSB-SC or AM with a carrier component) is the process of varying the carrier signal in accordance with the message signal. The AM receiver can use coherent or non-coherent detection. In the first case, the receiver is able to extract a synchronized local version of the transmitted carrier (by using carrier synchronization techniques such as a Costas loop for example, or through other methods).

In the case of this experiment, you will be “artificially” testing coherent detection by feeding the same carrier signal used at the transmitter directly to the receiver.

In the second case of non-coherent detection, the receiver doesn't have access to a synchronized local replica of the carrier, so other techniques are necessary. For example, AM (in contrast with DSB-SC) is specifically devised to allow non-coherent detection by transmitting a pure un-modulated carrier component in addition to the one modulated by the message signal. This mechanism allows the receiver to perform simple non-coherent envelope detection (as you studied in class). Notice that it is also possible to detect AM coherently (just like DSB-SC!).

Lab Work

Part 1: DSB-SC Modulation Set-up

- Measure the amplitude of the 2 kHz and 100 kHz sine outputs of the Master Signal Module using oscilloscope and record the results.
- Connect the set-up as shown in Figure 4.1.

$$\text{DSBSC} = 4\text{Vp-p } 2\text{kHz sine} \times 4\text{Vp-p } 100\text{kHz sine}$$
- Set the Trigger Source Control of Oscilloscope to CH1 (INT) position.
- Apply 2kHz sinusoidal signal (message) from Master signal generator to Y input of the multiplier module. Apply 100kHz sinusoidal signal to X input. There are three multipliers. You may choose any of these. Either DC or AC coupling can be used.
- Adjust the oscilloscope's Timebase Control to view about two cycles of the Master Signal Module's 2kHz SINE output.
- Set the scope's Channel 1 Vertical Attenuation Control to the 1V/DIV position and the Channel 2 Vertical Attenuation Control to the 2V/DIV position.
- Observe and draw the two waveforms

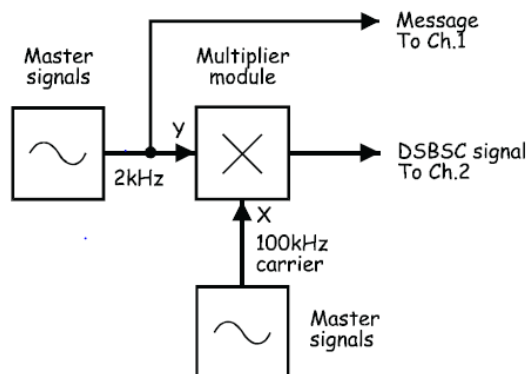


Figure 4.1 DSB-SC Modulation.

Part 2: DSB-SC Demodulation

DSB-SC signals are demodulated using a circuit called a product detector. Product detector or product demodulator is a synchronous detector. The product detector uses multiplication. The incoming DSBS signal is multiplied by a pure sinewave that must be at the same frequency as the DSBSC signal's suppressed carrier. This sinewave is generated by the receiver and is known as the local carrier.

DSBSC demodulator's output = the DSBSC signal x the local carrier

- Modify your connection as shown in Figure 4.2. Use a second multiplier to implement the demodulator.
- Locate the Tuneable Low-pass Filter module and set its Gain control to about the middle of its travel.
- Turn the Tuneable Low-pass Filter module's soft Cut-off Frequency Adjust control fully clockwise.
- Draw the demodulated DSBSC signal to scale below the modulated signal.

Question 1

Why must a product detector be used to recover the message?

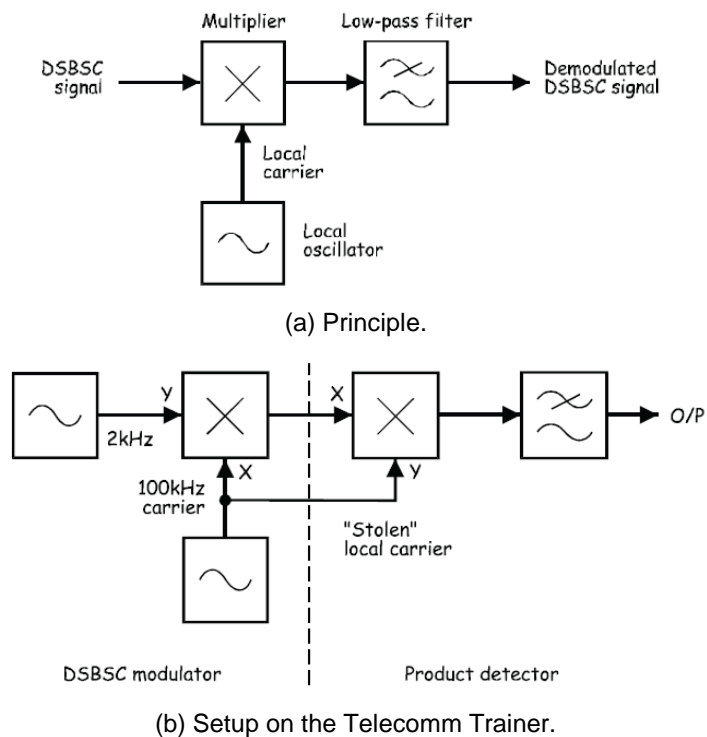


Figure 4.2 DSB-SC Demodulation (Product Demodulator).

Part 3: Investigating the message's amplitude on the recovered message from DSB-SC signal

- Disconnect the plug to the Master Signal module's 2 kHz SINE output.
- Locate the Buffer module and turn its Gain control to about a quarter of its travel.
- Use the Buffer module to modify the set-up as shown in Figure 4.3 below. Do not disconnect the demodulator section.
- Vary the message signal's amplitude up and down a little (by turning the Buffer module's Gain control left and right a little) while watching the demodulated signal.

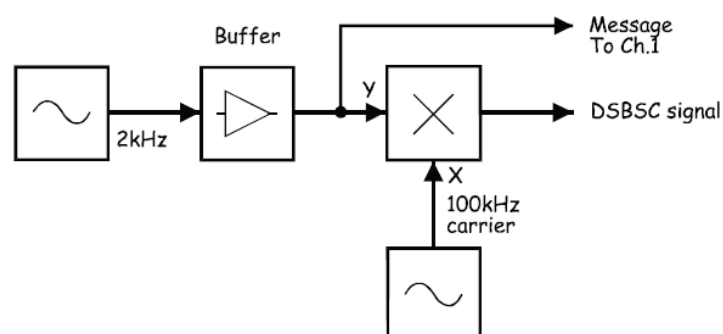


Figure 4.3 Investigation of the effect of amplitude variation in signal input.

Question 2

What is the relationship between the amplitudes of the two message signals?

e. Slowly increase the message signal's amplitude until the demodulated signal begins to distort.

Question 3

What do you think causes the distortion of the demodulated signal? If you are not sure, connect the oscilloscope's Channel 1 input to the DSBSC modulator's output and set its Trigger Source control to the Channel 2 position.

Part 4: Carrier Synchronization

A. Effect of phase mismatch

Crucial to the correct operation of a DSBSC communications system is the synchronization between the modulator's carrier signal and the product detector's local carrier. Any phase or frequency difference between the two signals adversely affects the system's performance.

- a. Turn the Buffer module's Gain control fully anti-clockwise again.
- b. Return the oscilloscope's Timebase control to about the 0.1ms/DIV position.
- c. Locate the Phase Shifter module and set its Phase Change control to the 180° position.
- d. Set the Phase Shifter module's Phase Adjust control to about the middle of its travel.
- e. Modify the set-up as shown in Figure 4.4 below.

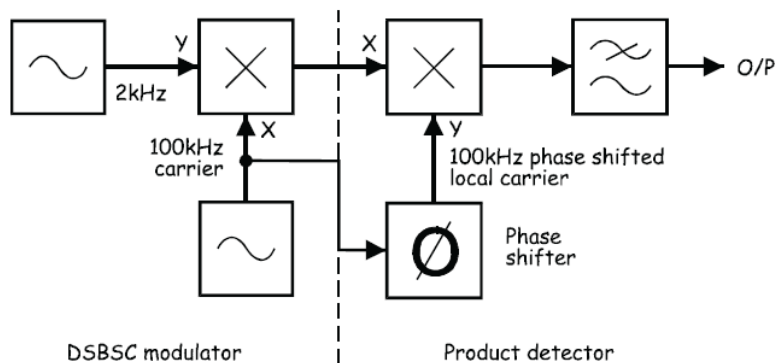


Figure 4.4 Investigation of the effect of phase mismatch.

- f. Slowly increase the Buffer module's gain until you can comfortably hear the demodulated 2 kHz tone (use the headphone).
- g. Vary the Phase Shifter module's Phase Adjust control left and right while watching and listening to the effect on the recovered message.

h. Adjust the Phase Shifter module's Phase Adjust control until the recovered message is biggest.

Question 4

Given the size of the recovered message's amplitude, what is the likely phase error between the two carriers?

i. Adjust the Phase Shifter module's Phase Adjust control until the recovered message is smallest.

Question 5

Given the size of the recovered message's new amplitude, what is the likely phase error between the two carriers?

B. Effect of frequency mismatch

When there is a frequency mismatch (error) between the DSBSC signal's carrier and the product detector's local carrier, there is a corresponding frequency error in the two products that usually coincide. One is the message frequency - error and the other is at the message frequency + error. If the error is small (say 0.1 Hz) the two signals will alternatively reinforce and cancel each other which can render the message periodically inaudible but otherwise intelligible. If the frequency error is large (say 5Hz) the message is reasonably intelligible but fidelity is poor. When frequency errors are large, intelligibility is seriously affected.

- a. Locate the VCO module and set its Range control to the HI position.
- b. Set the VCO module's Frequency Adjust control to about the middle of its travel.
- c. Modify the set-up as shown in Figure 4.5 below.

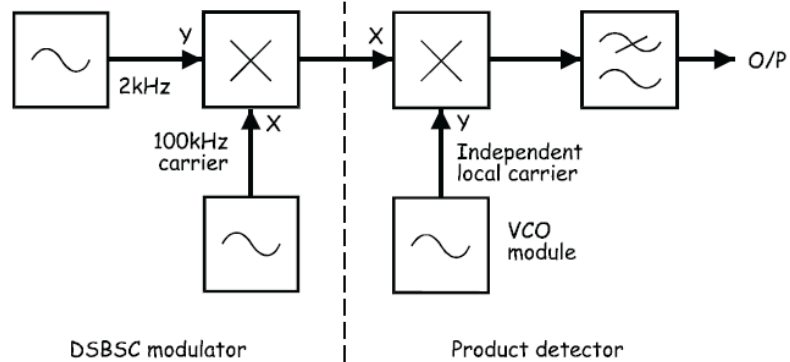


Figure 4.4 Investigation of the effect of frequency mismatch.

- d. Vary the VCO module's Frequency Adjust control left and right and observe the effect on the recovered message.
- e. If you are not doing so already, listen to the recovered message using the headphones.
- f. Use the display of the recovered message on the oscilloscope and your hearing to try to get VCO module's frequency as close as possible to the transmitter's carrier frequency.

Tip: If you cannot remember what 2 kHz sounds like, disconnect the plug to the VCO module's output and connect it to Master Signals module's 100 kHz SINE output for a couple of seconds. This will mean that the two carriers are the same again and the message will be recovered.

g. Disconnect the plugs to the Master Signals module's 2kHz SINE output and connect them to the Speech module's output.

h. Talk, sing, etc. into the microphone while varying the VCO module's Frequency Adjust control and listen to the effect of an unsynchronized local carrier on speech.

Experiment 5

Amplitude Modulation (AM) and Demodulation

Hardware Experimentation

Objectives

The goal of this lab is to allow the students to experiment with DSB-SC Modulation & demodulation using Telecoms Trainer hardware. The detailed objectives include the following:

- 1) Implementing AM modulator and demodulator with non-coherent detection.
- 2) Experimental evaluation of modulation index with tone modulation.
- 3) Understanding the ripple characteristics of the demodulated signal with respect to the low pass filter cut off frequency.

Pre-Lab Work

Answer the following questions:

1. Explain the difference between coherent and non-coherent detection of AM signals
2. Can DSB-SC signal be detected non-coherently with an envelope detector? Explain why.

Overview

Basically, Amplitude Modulation (whether it is DSB-SC or AM with a carrier component) is the process of varying the carrier signal in accordance with the message signal. The AM receiver can use coherent or non-coherent detection. The AM with carrier modulation is known as Amplitude Modulation (AM). In AM as in DSBSC modulation, a base band signal such as speech converted into electrical signal using a microphone is used to vary the amplitude of a carrier (a pure sine wave). The carrier usually has a frequency that is much higher than the message's frequency. An AM signal can be demodulated using non-coherent detectors. This is enabled by the added unmodulated carrier component in the modulated signal. This mechanism allows the receiver to perform simple non-coherent envelope detection (as you studied in class). Notice that it is also possible to detect AM coherently (just like DSB-SC!)

The mathematical model defining the AM signal is:

$$AM = (DC + message) \times \text{the carrier}$$

When the message is a simple sine wave, the above AM signal consists of three sine waves:

1. One at the carrier frequency

2. One with the frequency equal to the sum of the carrier and message frequencies (upper side band)
3. One with the frequency equal to the difference between the carrier and message frequencies (lower side band)

AM is also known as double side band with carrier modulation.

Lab Work

In this experiment you will use the Telecoms-Trainer to generate a real AM signal by implementing its mathematical model. This means that you will add a DC component to a pure sinewave to create a message signal and then multiply it with carrier (another sinewave at a higher frequency). You will examine the AM signal using the scope and compare it to the original message. You will do the same with speech for the message instead of a simple sinewave.

Following this, you will vary the message signal's amplitude and observe how it affects the modulated carrier. You will also observe the effects of modulating the carrier too much. Finally, you will measure the AM signal's depth of modulation using a scope.

Part 1: AM with carrier Modulation Set-up

1. Set the scope to ensure that:
 - the Trigger Source control is set to the CH1 (on INT) position.
 - the Mode control is set to the CH1 position.
2. Set the Oscilloscope's CH1 and CH2 Coupling controls to GND position and move the display traces (horizontal lines) to the center of the display.
3. Set the oscilloscope's CH1 Input Coupling control to the DC position.
4. Locate the Adder module and turn its G and g controls fully anti-clockwise.
5. Locate the Variable DCV module and turn its DC Voltage control almost fully anti-clockwise.
6. Connect the set-up shown in Figure 5.1 below.

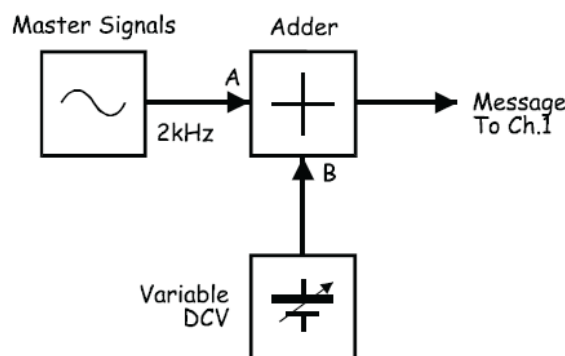


Figure 5.1 Adding DC to message signal

7. While watching the Adder module's output on the scope, turn its g control clockwise until the DC level is 1V.
8. While watching the Adder module's output on the scope, turn its G control clockwise to obtain a 1Vpp sinewave.

Question 1

In what way is the Adder module's output now different to the signal out of the Master Signals module's 2kHz SINE output?

7. Modify the set-up as shown in Figure 5.2 below.

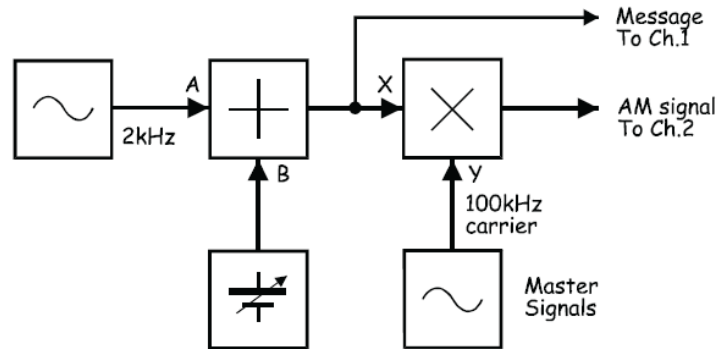


Figure 5.2 AM with carrier modulation

The output of the multiplier is of the form $AM = (DC + message) \times carrier$. With values, the equation is: $AM = (1V_{DC} + 1V_{pp} \text{ 2kHz sine}) \times 4V_{pp} \text{ 100kHz sine}$.

8. Set the oscilloscopes Volt/DIV and Time/DIV settings appropriately to observe the two waveforms and record the waveforms.

Question 2

What feature of the Multiplier module's output suggests that it is an AM signal?

Question 3

The AM signal is a complex waveform consisting of more than one signal. Is one of the signals a 2kHz sinewave? Explain.

Question 4

For the given inputs to the Multiplier module, how many sinewaves does the AM signal consists of, and what are their frequencies?

Part 2: AM with carrier signal Demodulation

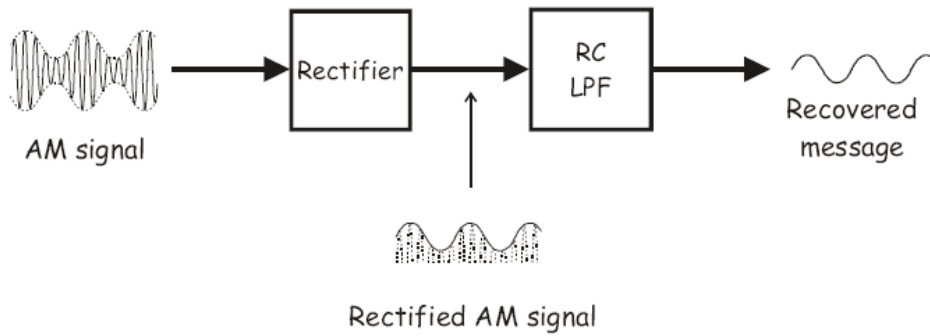


Figure 5.3 Demodulation of AM signal by Envelope Detection (non-coherent detection)

Do not disconnect the modulator set-up in Part 1. Proceed with implementing the demodulator.

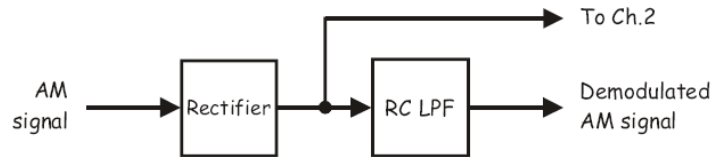


Figure 5.4 Envelope Detector set-up

1. Modify the setup as shown in Figure 5.4.
2. Adjust the scope's Vertical Attenuation controls to appropriate settings for the signals.
3. Observe and record the two waveforms.
4. Disconnect the scope's Channel 2 input from the Rectifier's output and connect it to the RC LPF's output instead.
5. Observe and record the demodulated AM signal to scale in the space that you left on the graph paper.

Question 1

What is the relationship between the original message signal and the recovered message?

6. Vary the message signal's amplitude up and down a little (by tuning the Adder module's G control left and right a little) while watching the demodulated signal.

Question 2

What is the relationship between the amplitudes of the two signals?

7. Slowly increase the message signal's amplitude to maximum while observing the demodulated signal.

Question 3

What do you think causes the heavy distortion of the demodulated signal? Tip: If you are not sure, connect the scope's Channel 1 input to the AM modulator's output and set its Trigger Source control to the CH2 position.

Question 4

Why does over-modulation cause distortion?

Part 3: Transmitting and recovering speech using AM

1. If you moved the scope's CH 1 input to help answer Questions in Part 2, reconnect it to the Adder module's output and return the scope's Trigger Source control to the Ch1 position.
2. Adjust the message signal's amplitude back to 1V_{pp} (by turning the Adder module's G control anti-clockwise).
3. Modify the set-up as shown in Figure 5.5 below.

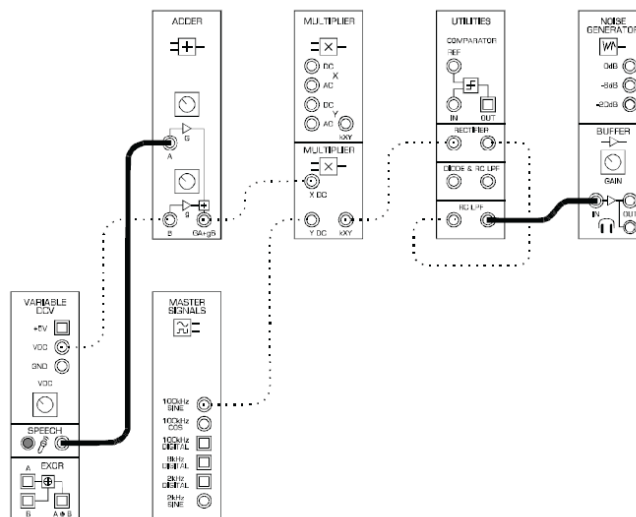


Figure 5.5 Connections for speech transmission and recovery by AM.

4. Set the scope's Timebase control to the 2ms/DIV position
5. Turn the Buffer module's Gain control fully anti-clockwise.
6. Without wearing the headphone, plug them into the Buffer module's headphone socket.
7. Put the headphone on.
8. As you perform the next step, set the Buffer module's Gain control to a comfortable sound level.
9. Talk, sing or hum while watching the scope's display and listening on the headphones.

Frequency Modulation (FM)

MATLAB Simulation

Objectives

The main objectives of this experiment are:

- 1) To gain a good understanding of Frequency Modulation.
- 2) To learn how to implement FM modulation & demodulation in software.
- 3) To demodulate an FM signal using the Differentiation (or Slope) technique.

Pre-Lab Work

- 1) Integration can be used to find the area under a given curve defined by some function $f(u)$. Our concern in this experiment is to learn how to numerically integrate a function $f(u)$ up to some point t (i.e. to get a function of t , say $m(t)$). To do so consider the following formula (i and k are indices in MATLAB):

$$m(k) = \frac{1}{\text{length}(f)} \sum_{i=1}^k f(i); \quad k = 1, 2, \dots, \text{length}(f)$$

Generate $\mathbf{f} = \cos(2*\pi*t)$ for $\mathbf{t} = 0:0.001:1$. Find \mathbf{m} . In order to speed up the time of computation, avoid using **for**-loops. Think of the functions: `sum` or `cumsum`. You also need to plot \mathbf{f} vs. \mathbf{t} and \mathbf{m} vs. \mathbf{t} .

- 2) How to differentiate a function in MATLAB? First, generate a function f in MATLAB like the one in Part 1. Use the function `diff(f)` and store it as a vector called \mathbf{df} . Now plot \mathbf{f} vs. \mathbf{t} and \mathbf{df} vs. \mathbf{t} . What will happen? Type `whos` in the MATLAB command window to see the dimensions of \mathbf{t} , \mathbf{f} , and \mathbf{df} , what do you notice? What do you need to be able to plot \mathbf{df} vs. \mathbf{t} ? Think about the length of the \mathbf{t} and \mathbf{df} vectors. Read carefully how `diff(f)` works by typing `help diff` in the MATLAB command window.

Overview

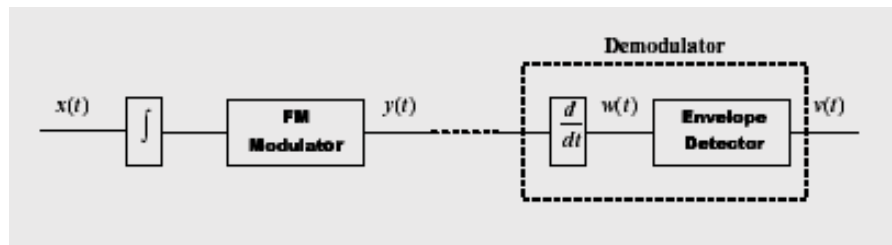
Angle modulation includes both Frequency and Phase modulation schemes (FM and PM), which are characterized by their superior performance (compared to AM) in the presence of noise at the expense of higher bandwidth requirements. As you studied in class, FM and PM are very similar. In fact, an FM signal can be interpreted as PM signal and vice-versa (refer to your textbooks). As such, our focus in this lab will be on FM modulation exclusively.

An FM-modulated signal has its instantaneous frequency that varies linearly with the amplitude the message signal. For example, a message signal $x(t)$ causes the frequency of the FM signal $y(t)$ to vary linearly around a central carrier frequency f_c . The following formula describes this relationship:

$$y(t) = A_c \cos\left(2\pi f_c t + k_f \int_0^t x(\xi) d\xi\right)$$

Here k_f is known as the sensitivity factor, and represents the frequency deviation rate as a result of message amplitude change. In practice, FM modulation is implemented by controlling the instantaneous frequency of a voltage-controlled oscillator (VCO). The amplitude of the input signal voltage controls the oscillation frequency of the VCO output signal.

FM signals can be demodulated using different techniques (refer to your textbook for details). Our focus in this experiment will be on the Slope Method, which uses a cascaded differentiator with an envelope detector circuit as illustrated in the diagram below. The differentiator basically produces an AM-like signal that is then demodulated by the envelope detector block. Refer to your textbooks for the mathematical details of this method.



Lab Work

- 1) Implement the FM Modulator function for the message signal $x(t) = \sin(2\pi t)$. Use a carrier frequency of $f_c = 100$ Hz and a frequency sensitivity factor $k_f = 160\pi$.
- 2) Plot $x(t)$ and $y(t)$. Also, plot the magnitude spectrum for $y(t)$.
- 3) Next, implement the FM demodulator part.
- 4) Plot $w(t)$, and $v(t)$ both in Figure 3. You need to present only one period of the message signal with $t_s = 1/(10f_c)$.

Useful MATLAB Functions: `fftshift(fft())`, `cumsum`, `diff`, `plot`, `subplot`, `figure`, `xlabel`, `ylabel`, `title`.

- 5) Repeat the above for a sinc message signal $x(t) = \text{sinc}(100\pi t)$ with $f_c=250\text{Hz}$ and $k_f=200\pi$, and where $t = -0.1:0.1$. What is the bandwidth of $x(t)$ and $y(t)$?

Homework Questions

- Q1. What is the effect of changing the value of the sensitivity factor k_f ?
- Q2. List and compare all the other methods you know about for demodulating an FM signal?

Experiment 7

Frequency Modulation Using the Telecoms Trainer

Hardware Experimentation

Objectives

The main objectives of this experiment are:

- 1) To build on the introduction to FM in the previous lab (based on MATLAB simulation), and to gain a clearer understanding of its concepts using hardware experimentation with the ACB.
- 2) To perform FM bandwidth computations for different message waveforms.
- 3) To experiment demodulation scheme for FM, using a method known as zero-crossing Detection.

Pre-Lab Work

- 1) Consider the case of FM modulation with a sinusoidal message signal (tone modulation). Give a simple validation of Carson's formula for bandwidth estimation based on the use of mathematical derivations involving *Bessel Functions* (refer to your textbook for relevant details).
- 2) As a preparation for the bandwidth computation in this lab, review Fourier series techniques for rectangular and triangular signals. Give the Fourier series for these signals assuming some signal period T .

Overview

Refer to your textbook and to the previous lab for a general overview and mathematical expressions of FM-modulated signals. This hardware experiment will be based on the Telecoms Trainer board, where we use the VCO for FM signal generation, and the Zero Crossing Detector for FM signal demodulation.

Part 1 of the work in this lab includes FM bandwidth evaluation. A useful expression for the computation of FM bandwidth is given by Carson's rule:

$$B_T = 2(1 + \beta)W$$

where W is the message signal bandwidth, while the modulation index is given by

$$\beta = \frac{k_f \max|x(t)|}{W}$$

In this experiment, the FM modulator is the VCO module.

Lab Work

In this experiment you will generate a real FM signal using the VCO module on the Telecoms Trainer. First you will set up the VCO module to output an unmodulated carrier at a known frequency. Then you will observe the effect of frequency modulating its output with a square wave then speech. You will also use a sinewave to observe the spectral composition of an FM signal (in the time domain). Finally, you will use the speech signal to demonstrate the effect that a message's amplitude has on an FM modulator.

1) Frequency modulating a square wave

- Set up the scope to ensure that the Trigger control is set to the CH1 (or INT) position and the Mode control is set to the CH1 position.
- Locate the VCO module and turn its Gain control to about two thirds of its travel (about the position of the number 2 on a clock face)
- Set the VCO module's Frequency Adjust control to about the middle of its travel.
- Set the VCO module's Range control to the LO position.
- Connect the output of VCO to CH1 of the scope

Note: Insert the oscilloscope probe's black plug into a GND socket.

- If necessary, tweak the VCO module's Gain control unit you obtain an output from the VCO and record the amplitude and frequency.
- Set the scope's Timebase control to obtain 20 μ s/DIV position.
- Adjust the VCO module's Frequency Adjust control so that one cycle of its output is exactly 5 divisions. (This sets the VCO modules' free running frequency to 10 kHz)
- Set the scope's Timebase control to obtain 0.1ms/DIV position. (This will show about ten cycles of the VCO module's SINE output)
- Modify the set up as shown in Figure 7.1 below and adjust the setting if necessary to obtain waveforms as in Figure 7.2.

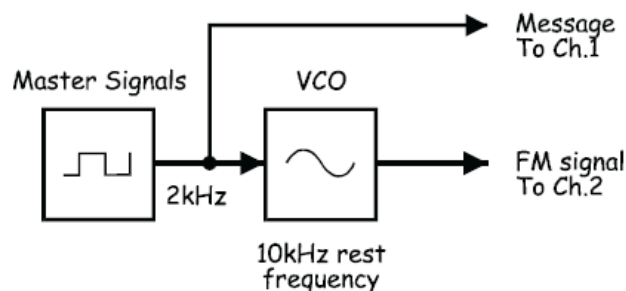


Figure 7.1 Frequency modulating a square wave (set-up)

(k) Measure and record the amplitude and frequency (2 kHz) of the message signal.

Question 1

How does the frequency of the modulated carrier change?

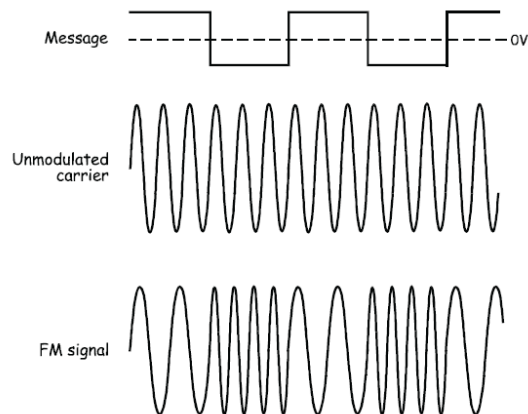


Figure 7.2 Frequency modulating a square wave

(l) Use the scope's CH1 Vertical Position control to overlay the message with the FM signal and compare them.

(m) Calculate and record the two frequencies of the FM waveform. Use the oscilloscope's Volts/DIV and Time/DIV sensitivity values for your calculations.

Question 2

Estimate the FM sensitivity factor (constant of modulation) K_f .

2) FM Bandwidth:

In practice, the spectral properties of modulated signals are normally observed on a spectrum analyzer (which you don't have access to at present). In this part, you will resort to theoretical expressions (supported by lab measurements) to estimate FM signal bandwidth.

For the 2 kHz square wave signal used in Part 1, what is the bandwidth of the message signal, and what is the maximum frequency deviation of the FM modulated signal? You will need to use the K_f factor measured in Part 1.

- Change the message signal from square waveform to sinusoidal waveform (2 kHz).
- Observe the FM signal waveform.
- Estimate the bandwidth using Carson's rule in this case.
- Now change the message signal to 2 kHz SINE and observe the modulated signal and the demodulated signal.
- Use Carson's Rule and the above results to estimate the bandwidth of the FM signal modulated by the 2 kHz sinusoidal message signal.

3) FM Demodulation:

There are many methods of demodulating an FM signal as there are of generating one. Examples include: the slope detector, the Foster-Seeley discriminator, the ratio detector, the phase-locked loop (PLL), the quadrature FM demodulator and the zero-crossing detector. It is possible to implement several of these methods using the Telecoms Trainer but, for an introduction to the principles of FM demodulation, only the zero-crossing detector is used in this experiment.

The zero-crossing detector is a simple yet effective means of recovering the message from FM signals. The block diagram is shown in Figure 7.3.

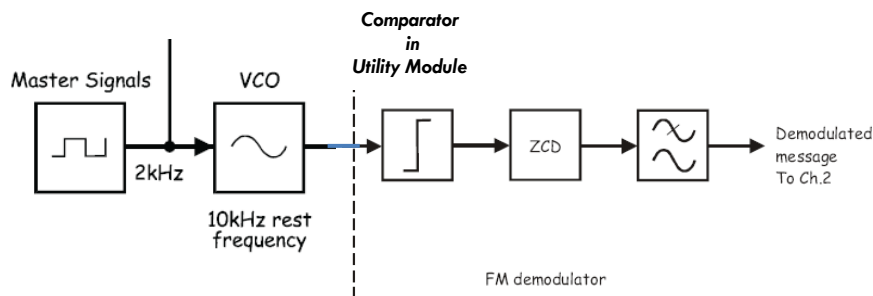


Figure 7.3 Zero-crossing detection FM signals

- Keep the modulator set-up unchanged
- Locate the Twin Pulse Generator module and turn its Width control fully anti-clockwise.
- Set the Twin Pulse Generator module's Delay control fully anti-clockwise.
- Connect the set-up as shown in Figure 7.3. Use the comparator available in the utility module followed by the ZCD.

Question 3

What type of waveform does the comparator output?

Question 4

What does this tell us about the DC component of the comparator's output?

- Set the scope's Timebase control to the $2\mu\text{s}/\text{DIV}$ position.

(f) Adjust the Twin Pulse Generator module's Width control for an output pulse that is $12\mu\text{s}$ long.

(g) observe the tunable low pass filter output. Adjust the filter cut-off frequency until the output resembles the input message.

(h) repeat (a-g) for 2k Hz SINE and square wave as message signal.

4) FM Modulation and Demodulation of Speech Signal:

Disconnect the master signal output and connect the microphone output instead. Observe the message signal, modulated signal, and demodulated signal. Use your skills and intuitions to carry out necessary adjustments as necessary to obtain reasonable waveforms.

Homework Questions

- Q1.** In Part 1 above, specify the expression for the instantaneous frequency of the modulated signal.
- Q2.** Consider two message signals, $m_1(t) = A \cos(2 \pi f_m t)$ and $m_2(t) = A \cos(4 \pi f_m t)$, with the same amplitude but m_2 having double the frequency. Would the two modulated FM signals (by m_1 and m_2) have different frequency deviations? What about their bandwidth? Explain.
- Q3.** Re-do the bandwidth computation of Part 2 with the triangular and rectangular signals. Note: you need to use Fourier techniques to estimate the bandwidth of the message signals (you can truncate to a few harmonics as appropriate).

Experiment 8

Sampling & Quantization

MATLAB Simulation

Objectives

The main objectives of this experiment are:

- 1) To gain a clearer understanding of sampling theory and uniform quantization.
- 2) To learn how to simulate these operations using MATLAB.

Pre-Lab Work

- Use the MATLAB function **pulstran** to generate and plot three periodic pulse trains with different shapes each with three different pulse durations. Read the help of that function and try to take advantage of the example given is at the end of the help for **pulstran**. However, do not copy it!
- Given a vector $x=[1.1,2.6,-3.5,-0.001,0.768]$, apply the MATLAB functions **fix**, **round**, **floor**, & **ceil** on x . State the differences between the resulting vectors.

Overview

An analog signal is characterized by the fact that its attributes (like: amplitude, frequency and phase) can take any value over a continuous range. On the other hand, digital signals can take only discrete and finite values. One can convert an analog signal to a digital signal by sampling and quantizing (collectively called analog-to-digital conversion, or ADC).

It is typically more efficient to process the resulting discrete signals by digital signal processors. The processed signals are then converted back into analog signals using a reconstruction or interpolation operation (called digital-to-analog conversion, or DAC).

Sampling:

To sample a continuous-time signal $x(t)$ is to represent $x(t)$ at a discrete number of points, $t = nT_s$, where T_s is the sampling period. The sampling theorem states that a band-limited signal $x(t)$ with a bandwidth W can be reconstructed from its sample values $x(n) = x(nT_s)$ if

the sampling frequency $f_s = 1/T_s$ is greater than twice the bandwidth W of $x(t)$. Otherwise, aliasing would result in $x(t)$. The minimum sampling rate of $2f_s$ for an analog band-limited signal is called the Nyquist rate.

Quantization :

In order to process the sampled signal digitally, the sample values have to be quantized to a finite number of levels, and each value can then be represented by a string of bits. For example, if the signal is quantized to N different levels, then $\log_2(N)$ bits per sample are required.

Notice that to quantize a sample value is to round it to the nearest point among a finite set of permissible values. Therefore, a distortion will inevitably occur. This is called quantization noise (or error).

Quantization can be classified as uniform and non-uniform. In the case of uniform quantization, the quantization regions are chosen to have equal length. However, in non-uniform quantization, regions of various lengths are allowed. Non-uniform quantization can be implemented through compression-expansion (or companding) of the signal, and this is commonly used (as in telephony) to maintain a uniform signal-to-quantization noise ratio over the full dynamic range of the signal (refer to your textbook for more details).

Lab Work

- 1) Generate a time vector \mathbf{t} from -0.5 to 0.5 with a step size of 0.001. Implement the function $x(t) = e^{-|t|/\tau}$ where $\tau = 0.1$. In order to simulate an A/D converter, perform the following tasks:
 - a. Plot \mathbf{x} and its magnitude spectrum in a two-panel figure window. What is the bandwidth of \mathbf{x} ?
 - b. Set the sampling frequency to twice the bandwidth of the signal \mathbf{x} (which is approximately 25Hz). Generate a rectangular pulse train starting at -2 to 2 where the step size is $1/(\text{sampling frequency})$ and with a duration $1E-10$. In a two-panel figure window, plot the pulse train and its magnitude spectrum. What is the relation between the time and frequency domain representation of the pulse train?
 - c. Sample \mathbf{x} using the pulse train and plot the resulting sampled version of \mathbf{x} , (say \mathbf{xs}). Also, plot its magnitude spectrum. What can you observe from both plots?

Now, to be able to simulate a D/A converter, perform the following tasks:

- d. Let \mathbf{h} be a vector of forty ones (40). Plot its magnitude spectrum. In order to correctly plot $H(\mathbf{f})$, use the `fft` function with a number of FFT points equal to the length of \mathbf{x} .

e. Use the function **filter** to form the reconstructed signal (say **xd**) from **xs** using the filter **h**, i.e, **xd=filter(h,1,xs)**. In a 4-panel figure window, show the plots of **x**, **xd**, and there magnitude spectrums. Comment on your findings.

Useful MATLAB functions: pulstran

- 2) Repeat 1 using a sampling frequency equal to 1.5 times the bandwidth of $x(t)$.
- 3) Repeat 1 using a sampling frequency equal to 10 times the bandwidth of $x(t)$.
- 4) Generate a sine signal (call it **x**) with frequency equal to $1/2\pi$ over a time interval of three periods. The step size should be equal to $1E-4$. The quantized signal will be equal to $x_q = \Delta x \text{round}(x/\Delta)$, where Δ is the quantization step size. You can compute it using the formula $\Delta = 2 \max(|x|)/L$ where $L = 2^b$ and b is the number of quantization bits. Use 2 bits to quantize **x** and show the following:
 - a. In one figure plot **x** vs. time and the quantized signal vs. time.
 - b. Compute the Signal-to-Quantization Noise Ratio (SQNR) defined by the formula: $SQNR = 10 \log_{10} \left(\frac{\sum x^2}{\sum (x - x_q)^2} \right)$ in dB.
- 5) Repeat Part 4 using 3, 4 and 5 bits per sample.
- 6) Plot the SQNR vs. the number of bits per sample for 2, 3, 4, and 5 bits. What is your conclusion?

Additional Questions:

- Q1. Can you process real-world analog signal by digital computers? List possible applications where sampling and quantization are necessary.
- Q2. From the above results, what can you conclude about the improvement in SQNR with every additional quantization bit?

Experiment

9

Pulse Amplitude Modulation

Hardware Experimentation

Objectives

The main objectives of this lab are:

- 1) To gain a practical understanding of the concepts of analog signal sampling, their usefulness and limitations.
- 2) To learn about Pulse Amplitude Modulation (PAM) as a transmission scheme.
- 3) To experiment with the realization of PAM modulation & demodulation using the DCB.

Pre-Lab Work

- 1) Explain the main result of the Sampling Theorem, and give a simple graphical proof for it (there is no need for full analytical details).
- 2) Discuss why is the sampling of analog signals quite useful in many applications?

Overview

An analog message signal, representing voice for example, has continuous amplitude and frequency values that vary with time. Analog communication systems transmit the complete analog waveform. But instead of transmitting the analog signal, it is possible to transmit discrete pulses (or samples) that represent some parameter of the message signal's waveform at regular intervals in time.

In the case of pulse amplitude modulation (PAM), the amplitude of the analog waveform is sampled at discrete (i.e., discontinuous) time instants, and transmitted as a sequence of pulses. Notice, however, that this sampling produces amplitudes that can still take *any value*, hence the scheme is not fully digital (unlike PCM which will be studied in a subsequent lab, and which uses quantization to represent the discrete samples by a finite number of bits).

As long as the sampling of the analog signal is taken with a sufficiently high frequency (higher than the minimum Nyquist rate of twice the signal largest frequency), it can be shown that there is *no loss* in information as a result of taking discrete samples. In fact, the PAM receiver

simply passes the received samples through a low-pass interpolating filter to fully recover the analog waveform. Notice also that the results of the sampling theorem, which are derived for the case of ideal sampling with Dirac delta functions, are equally applicable for practical sampling with pulses that have a finite duration (refer to your textbook for more details).

Different types of sampling can be distinguished. With “natural” PAM signals, the amplitude of each pulse follows the amplitude of the message signal for the duration of the pulse. Another type of sampling results in “flat-top” PAM signals where a sample&hold circuit holds the sampled amplitude at a constant level between the sampling times, thus resulting in a staircase PAM signal shape. This is sometimes used at the receiver end prior to low-pass filtering since it helps increase the amplitude of the reconstructed signal.

There are a number of advantages for transmitting PAM pulses rather than complete analog signals. For example, if the duration of the PAM pulse is small, the energy required to transmit the pulses is much less than the energy required to transmit the full analog signal. For a sampling pulse train with a duty cycle fraction PW/T (i.e., the sampling pulse duration is PW , and its period is T), it can be shown that the power of the sampled PAM signal P_p is only a fraction of the total analog signal power P_s , given by: $P_p = (PW/T) \times P_s$.

Another advantage of PAM has to do with the ability to multiplex (or combine) several different signals and transmit them on the same communication channel. Although this is also possible with analog communication (using for example frequency division multiplexing-FDM), it is much simpler and more economical to implement with digital or discrete systems like PAM using what’s known as time division multiplexing (TDM). Since PAM sends amplitude pulses of a given signal at discrete periodic time slots (or intervals), it is then possible to assign the remaining time slots for other signals, thereby maximizing the utilization of the channel. There is obviously a need for maintaining adequate synchronization at the transmitter and receiver levels to distinguish the different signals. This type of TDM transmission is very efficient and widely used in practical communication systems such as telephone networks, particularly in combination with Pulse Code Modulation-PCM (which is similar to PAM, but uses additional amplitude quantization as will be seen in a following lab).

Lab Work

2) PAM Signal Generation

- Use a two-post connector to apply the M2 message signal to the SAMPLER input. Measure the amplitude and frequency of this signal on the oscilloscope.
- Check the sampling pulse frequency on port SP of the DCB using the oscilloscope (it should be 8KHz). Is this sampling rate sufficient for the given input signal?
- Apply the input analog signal and resulting PAM signal from the SAMPLER output to channels 1 & 2 of the oscilloscope. Observe and sketch both signals. Is the reconstruction quality good?
- What type of sampling is being used: natural sampling, or sample & hold?

3) Power Calculation

- Use the oscilloscope to measure the pulse width PW of the PAM signal, and the period of the sampling pulse train T_p .

- Calculate the pulse duty cycle PW/T_p .
- Calculate the power of the analog sinusoidal signal and the discrete PAM signal across a hypothetical $1\text{ k}\Omega$ resistor. Comment on the efficiency of the PAM scheme.

4) PAM Signal Demodulation

- First, you need to measure the receiver filter characteristics. Use the external signal generator to apply sinusoidal signals with 2V pk-pk and increasing frequency to obtain a rough sketch of the filter frequency response.

Hint: The filter 3dB cutoff frequency should be around 2.6 kHz.

- Disconnect the M2 input, and use the external function generator to apply a new sinusoidal input signal to the SAMPLER. Set the frequency to 2 kHz and the amplitude to 4V pk-pk.
- Now, use a 2-post connector to apply the sampled PAM signal at the output of the SAMPLER to the FILTER input. Observe the reconstructed signal at the FILTER output, and sketch it. Comment on the quality of the reconstructed signal.
- Let the input signal frequency now be changed in the range 3 to 3.5 kHz. Is the Nyquist criterion still met (given that SP is 8kHz)? Observe the demodulated signal at the FILTER output. Explain why it is still strongly attenuated (although the sampling frequency is sufficient!)

5) Effect of Aliasing

- When the sampling rate is 8kHz, what is the theoretical input frequency at which aliasing starts to appear?
- Select an input message signal frequency of 5kHz. Apply the original analog input and PAM demodulated signals on both channels of the oscilloscope. Sketch the resulting demodulated waveforms and comment.
- Sketch a frequency-domain description of the problem that you just observed in time-domain on the oscilloscope. Notice that it possible to observe things in frequency-domain too (using spectrum analyzers, which are currently not available in sufficient numbers for the EE370 lab requirements).

Additional Questions

- Q1.** Explain why it is sufficient to use a low-pass filter at the receiver to recover the original transmitted analog signal? Hint: you can assume that sampling is ideal (i.e., with delta pulses), then use frequency-domain diagrams to show the effect of sampling on the signal spectrum.
- Q2.** Consider the sampling of a signal which has some maximum frequency f_m , but that may be corrupted by high-frequency noise. What should you do first? Explain why.

Experiment

10

Pulse Code Modulation and Time Division Multiplexing

Hardware Experimentation

Objectives

PCM and time-division multiplexing are widely used in modern telecommunication networks. The main objectives of this experiment are to use to DCB board in order to:

1. Gain a good understanding and hands-on experience with Pulse Code Modulation
2. Learn about non-uniform quantization and its applications (e.g., in telephony)
3. Understand and experiment with the concept of Time Division Multiplexing.

Pre-Lab Work

1. Read the relevant material in your textbook.
2. Describe the use of PCM and TDM in modern telephony: give the basic architecture of modern telephony networks from the user phone to the central office switch, discuss T1/E1 multiplexing, etc.
3. Discuss and compare the advantages & disadvantages of TDM vs. FDM (frequency-division multiplexing).

Overview

Pulse-code Modulation (PCM), like PAM, is a digital communication technique that sends samples of the analog signal taken at a sufficiently high rate (higher than the Nyquist rate). In addition, PCM differs than PAM in that it quantizes the samples by constraining them to only take a limited number of values, and then converts each value into a binary string of bits that are transmitted on the communication line. Typically, in digital telephony where PCM is widely used, the sampling rate is 8 kHz (higher than twice the voice band), and the quantization uses 256 levels (i.e., each sample is mapped into an 8-bit PCM code).

In practice, PCM is typically combined with Time Division Multiplexing (TDM), which is the process of combining many PCM signals representing different messages and transmitting them over the same channel on a time-sharing basis. Each PCM signal is assigned a time period called a slot on the transmission line, and slots are arranged in groups called frames. The main advantages of PCM transmission are: lower cost, ease of multiplexing and switching, and better noise immunity. Its main disadvantage is the stringent timing and synchronization requirements. Nowadays, PCM-TDM systems form the backbone for all digital telephony networks worldwide (refer to your textbooks for more details).

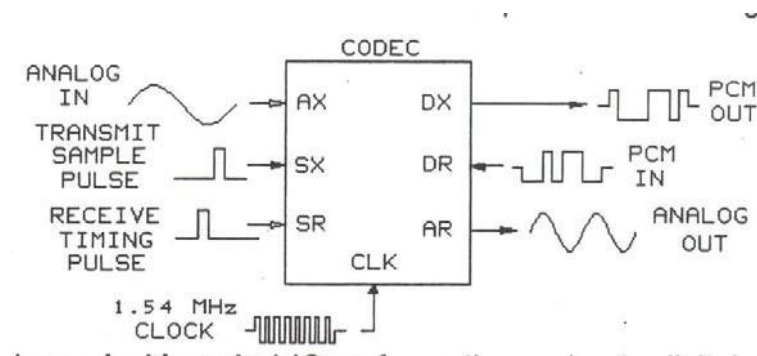
In this experiment, you will be using the PCM block in the DCB board, which is centered on a pair of voiceband CODECs (a codec is a Coder-Decoder) for PCM transmission and reception. These CODECs are IC (integrated circuit) chips that incorporate the major PCM-TDM functionalities. The basic operation of the CODEC is described below.

First, the analog signal at the AX input port is converted into a digital PCM signal at the DX output port. There is a timing signal SX giving an 8 kHz transmit pulse that occurs one clock cycle before the assigned time slot. Each pulse enables the codec to sample/encode the message signal to be transmitted in the following time slot.

At the receiver codec, the PCM signal at the DR port is recovered into an analog signal at the AR port. Similar to the transmitter codec, the receiver has an 8 kHz clock signal SR that enables the operation of the codec. In order for the receiver to decode a transmitted PCM code in a given time slot, the SX and SR pulses must be synchronized.

The transmission rate between codecs, i.e., the PCM line rate, is governed by a 1.54 MHz clock (typically referred to as T1 rate). Notice that this is much faster than the bit stream generated from a single user (which is $8\text{bits} \times 8\text{ kHz}$, or 64 kbps). This is because the PCM line actually carries a number of PCM signals (a T1 line corresponds to 24 PCM signals) in a TDM fashion, as explained above, and therefore needs to run at a much higher speed than a single 64 kbps PCM stream.

The actual architecture of the codecs used in the DCB board comprises many circuit blocks. On the transmitter (encoder) side, there is a voice-band anti-aliasing filter (limited to [0.2 kHz, 3.5 kHz]), followed by an 8 kHz sample & hold circuit, an ADC (analog-to-digital converter) that implements 8-bit quantization and compressor (μ -law or A-law), and finally a parallel-to-serial converter and output register. On the receiver (decoder) path, there is an input register with serial to parallel conversion, a sample & hold circuit, an expander (to remove the transmitter compression), and a receiver interpolation filter.



Lab Work

1) Part 1: PCM Signal Modulation & Demodulation

- Use the external function generator to apply a sinusoidal signal with frequency 1 kHz and amplitude 2 V peak-to-peak to Input AX of CODEC 1.
- Connect CODEC 1 to CODEC 2 with a two-post connector. CODEC 1 is the transmitter, and CODEC 2 is the receiver.
- Use the oscilloscope to simultaneously observe the AX signal of CODEC 1 and the AR signal of CODEC 2. Is the demodulation working properly?
- Observe the signal DX of CODEC 1 (or similarly, DR of CODEC 2) on the oscilloscope. What does this signal correspond to?
- If you keep increasing the amplitude of the input AX signal, what type of distortion do you get? What is the reason for that?
- Now, try to change the input frequency for AX. If the frequency is outside the range [0.2kHz, 3.5kHz], what happens to the received AR signal? What is the reason for that?

2) Part 2: PCM Timing Signals

- Disconnect the external function generator, and use a 2-post connector to apply the signal M1 to Input AX of CODEC 1. Record its frequency and amplitude.
- Observe the input sampling signal SX of CODEC 1 on the oscilloscope. Measure its period and frequency. What is the role of this SX signal?
- Check the SR signal of CODEC 2 on the oscilloscope. Is it in synch with the SX signal? What is the reason for that (i.e., will PCM decoding work if not)?
- Observe the relative timings of the signal SX and DX of CODEC 1 (same for RX and DX of CODEC 2). Apply these signals to channels 1&2 of the oscilloscope, and record your observations. When does DX start relative to SX? Explain why?
- Now, observe the characteristics of the DX signal (or RX) on the oscilloscope (which corresponds to PCM codes). How many bits per PCM code are there? What is the duration of one PCM code? Try displaying a few different code words on the oscilloscope. Are they having the same string of bits? Why?
- Using the oscilloscope, try to measure the PCM stream bit rate. Compare this with the theoretical bit rate you expect to see.

3) Part 3: Companding and Non-Uniform Quantization

- Launch the WinFACET software, and go through the material covering the overview and description of μ -law and A-law companding.
- Explain the meaning of non-uniform quantization. What is its main advantage?

- What is the difference between μ -law and A-law companding?

4) Part 4: Time-division Multiplexing (optional part)

- Launch WinFACET and go to Exercise 2 (Time Division Multiplexing).
- Go over the overview, then start the procedure, and execute all steps up to Step 26, and report your observations and answers to questions.

Additional Questions

Q1. How is PCM different than PAM?

Q2. Does a digital PCM stream of bits representing analog voice consume more or less bandwidth than the original analog signal?

Q3. What is the main advantage of PCM-TDM?

Q4. What is the difference between simplex and full-duplex communication links?

Experiment

11

Communication Channel Effects

Hardware Experimentation

Objectives

The main objectives of this experiment are:

1. To obtain a better understanding of the impact of channel bandwidth limitation on the performance of communication systems.
2. To learn how to measure channel bandwidth.
3. To experiment with channel distortion impact on PAM and PCM transmission using the DCB platform.

Pre-Lab Work

You should read the relevant material in your textbook, and answer the following questions:

1. By referring back to basic Fourier theory results, explain why the time duration of a pulse signal and its bandwidth are inversely proportional, i.e., if time duration is large, bandwidth is small, and vice-versa.
2. What is Inter-symbol Interference (ISI)? Is it caused by channel bandwidth limitation, external noise interference, or other factors? Explain.

Overview

A typical communication system consists of 3 main parts: the transmitter, the receiver, and the communication channel. This channel is a physical propagation medium used to carry signals between the transmitter and the receiver. For example, a propagation channel can be a twisted copper pair, an optical fiber, a coaxial cable, or a wireless RF link.

The propagation channel has many characteristics that can introduce distortion and cause a severe limitation to the proper reception of the transmitted signal. Some of these limitations include:

- Bandwidth limitation: where, for instance, a channel with poor high frequency characteristics (i.e., similar to a low-pass filter) will affect a digital pulse rise and fall times. Thus, limited bandwidth increases the time a pulse needs to attain the full amplitude required by the receiver. Also, if the channel has poor low-frequency characteristics, it will affect the top and bottom of the transmitted pulses, which no longer maintain a stable voltage, and become more difficult to detect.
- Additive noise: coming in the form of undesirable, random-like interference that corrupts the transmitted signal, and affects the ability of the receiver to detect it properly.
- Timing variations: also known as timing jitter, which is due to random variations in the channel time delay. As a result, the received signal is delayed by a varying amount of time from one signal period to the next, and become therefore difficult to track properly.
- Channel resonance: if the channel contains reactive elements, then oscillation can arise and cause amplitude ringing effects, with overshoots and undershoots than can lead to signal detection errors at the receiver.

In this experiment, we focus on bandwidth-related aspects of digital communication channels. For a digital transmitted pulse to be detected at the receiver, the significant frequency components of the pulse must pass undistorted through the channel. But since practical channels have limited bandwidth, only the frequency components inside the channel passband will be received properly. The elimination of some frequency components will change the shape of the received pulse, and may cause loss of information at the receiver.

To get a basic idea of how to assess channel bandwidth, consider for example the case where the channel can be approximated by a simple low-pass filter of 1st order (e.g., an R-C circuit). It is then found that the 3-dB pass-band of this channel (which can taken as a measure of bandwidth) is given by $BW = 1/(2 \pi RC)$ or $1/(2 \pi T)$, where T is the time constant of the circuit. Recall also from electrical circuit theory that one RC time constant is required for the output voltage to reach 63% of the input voltage (when the input is a stair-case pulse).

The concept of *rise time* can be linked to signal bandwidth as well. Rise time, denoted by T_r , is the time required for a pulse to rise from 10 to 90% of its final value. It can be shown that bandwidth is given by: $BW = 1/(\pi T_r)$. Notice that for a pulse to reach its full amplitude at the receiver, the pulse duration (or period) must be greater than the rise time imposed by the channel. This clearly shows that the pulse rise time (or, equivalently the channel bandwidth) sets a practical limit on the minimum signal pulse duration and, therefore, the maximum permissible symbol transmission rate.

In this experiment, you will be using the DCB board with the PAM, PCM and Channel Simulator blocks. The Channel Simulator is simply a configurable filter that emulates the response of a real communication channel. Additionally, the DCB channel block is also capable of including additive noise, but this is outside our scope at this time.

Lab Work

1) Part 1: PAM Transmission through a band-limited channel

- Launch WinFACET software and select the “Channel Effects” experiment under Digital Communications 1.
- Go through the overview section, then start Exercise 1: Channel Bandwidth. Notice that this lab does not cover Exercise 2 (with noise effects).

Important Note: Do NOT follow the procedures for using the PAM-TDM block (bottom left of the DCB board) to connect this experiment setup. Instead, use the PAM block (top left of the DCB). Make sure you observe the output PAM signal from the channel (and *not* from the RX filter).

- Apply the sinusoidal input signal M2 from the PAM block to the Sampler input and follow the rest of the procedure steps up to Step 19 (on Slide 12). Notice also that you will be using the receive filter of the PAM block (and not the Filter 1 in the PAM-TDM block).
- Follow all steps, and use the software control for toggling the CM switch to change the channel simulator bandwidth. Record all your observations, and report your answers to the questions.
- As an additional question, when the channel bandwidth is not limited, does the channel still have some attenuation in its pass-band (i.e., within 80 kHz)? Give measurements and explain whether this is considered as distortion or not?

2) Part 2: PCM Transmission through a band-limited channel

- Disconnect the previous setup, and proceed from Step 21 (slide 14) in the Exercise 1 procedure using the PCM block of the DCB board.
- Now, use CODEC 1 as a transmitter and CODEC 2 as a receiver, with the channel simulator in between. Connect the M1 signal to Input AX of CODEC 1, the PCM output DX to the channel IN, and the Channel OUT to DR of CODEC 2. The analog recovered signal is obtained at AR of CODEC 2.
- Follow all steps, and use the software control for toggling the CM switch to change the channel simulator bandwidth. Write your observations and answers to the questions.
- Finally, go through the five review questions after finishing the experimental procedure, and report your answers.

Additional Questions

- Q1.** In the lab procedure you went through, can you reliably transmit the PCM signal through a channel with 80 kHz bandwidth? Why? What should be the required channel bandwidth (approximately) to allow adequate reception?
- Q2.** Can you suggest a typical method for controlling the bandwidth of a transmitted signal? Explain how. Hint: think of the Fourier transform of various pulse shapes!

Experiment
12

Delta Modulation (DM)

MATLAB Simulation & Hardware Experimentation

Objectives

The main objectives of this experiment are:

1. To obtain a better understanding of Delta Modulation (DM) theory.
2. To learn how to simulate this digital transmission scheme using MATLAB.
3. To experiment with DM hardware implementation using the DCB platform.

Pre-Lab Work

You should read the relevant material in your textbook, and answer the following questions:

1. What is Delta Modulation comparable to (AM, FM, PCM, DPCM)? Explain.
2. Explain the meaning of "Delta" in the definition of Delta Modulation.
3. What are the advantages of Delta Modulation?

Overview

Like PCM and DPCM, Delta Modulation (DM) is a form of digital signal encoding. DM can be considered as a 1-bit differential PCM (DPCM) system. It converts each message signal amplitude change to a 1-bit sample: Logic 1 (or +E) for an increase, or Logic 0 (or -E) for a decrease. Therefore, the transmitted DM signal is a continuous series of bits indicating an increase or decrease in the message signal's amplitude.

In a typical realization of DM, the transmitter circuit contains a digital sampler which generates the DM logic pulses. The digital sampler is a compare/hold circuit that compares the message signal's amplitude with a reconstructed amplitude from a feedback integrator. Ideally, the feedback amplitude should be a close approximation of what the message-signal amplitude was at the previous clock cycle.

The DM receiver contains another integrator (similar to the transmitter's feedback integrator) and a low-pass filter which smoothes the distorted recovered signal out of the integrator.

DM transmission introduces some noise (or distortion) into the signal. There are different sources of distortion, including quantization noise and slope overload (refer to your textbook for a more in-depth description of these problems). DM relies on a very high sampling rate (typically much more than the minimum Nyquist rate of twice the signal bandwidth) to minimize the effects of distortion and to be able to produce an accurate re-construction of the signal.

In the hardware part of this lab, you will be using the DCB which contains a DELTA circuit block. In this block you will identify a comparator and a D-type flip flop which constitute the digital sampler in the DM transmitter. The comparator output feeds the D flip-flop input which latches (i.e., holds) the input value at its Q output (giving the DM transmitted bit sequence) until the next rising edge of the clock signal. The clock frequency is adjustable by control switches SW3&4, and can take any value among 8, 16, 32 and 64 kHz.

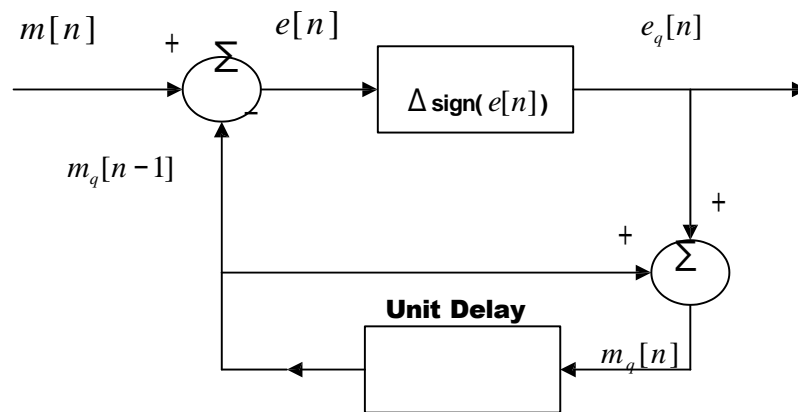
The transmitter feedback integrator is implemented by the INTEGRATE 1 circuit. The REF voltage determines how closely the feedback signal resembles the message signal. This REF value can be adjusted by the positive supply voltage potentiometer on the base unit. An RC time constant sets the positive and negative slope of the integrator output voltage. Each logic 1 DM pulse cause the integrator output voltage to increase by a fixed amount, and each logic 0 pulses causes it to decrease by a similar amount. Notice also the definition of the Baud rate for the transmitter. This simply means the number of signal changes per second. In the case of DM, this Baud rate is simply the bit rate out of the transmitter (true for binary modulation schemes only).

The DM receiver circuit is quite simple. It consists of a receive integrator and a low-pass filter. On the DCB, the receive integrator is the INTEGRATE2 circuit, and the LPF is the FILTER 1 circuit in the PAM-TDM circuit block. This filter recovers an approximate of the original message signal by smoothing out the high-frequency distortion in the integrated signal out of the INTEGRATE 2 circuit. Notice also that clock and synchronization circuits are not required in the demodulation process, which is another advantage of delta modulation.

Lab Work

Part I - MATLAB Simulation:

Consider the signal $m(t) = 2e^{-t/4} \cos(2\pi f_m t)$ where t is between 0 and 20 seconds with a step size equal to $1/\text{the sampling frequency}$. The sampling frequency should be large compared to the signal bandwidth. For example, let it be equal to 40 times the f_m where $f_m = 0.4$ Hz. In order to specify the quantization step size use the formula: $\Delta = \max |dm(t)/dt| / 5$. $m(t)$ based on the following block diagram.



You need first to write down the equations of $e_q[n]$ and $m_q[n]$. Then use a **for**-loop to create $m_q[n]$. That is, implement the formula:

$$m_q[n] = \sum_{i=1}^{\text{length}(m[n])} e_q[i]$$

Note: before you start the **for**-loop, you need to initialize the value of $m_q[1]$ to be zero.

I-1) In figure 1, **plot** $m[n]$ vs. t , then **hold** that figure and use **stairs** function to plot $m_q[n]$ vs. t also. Finally, use **hold** again to disable that figure.

I-2) Generate figure 2 with two panels where the first panel plot $e_q[n]$ vs. t using **stem** and in the second panel, **plot** $(m[n] - m_q[n])$ vs. t . What are your observations? Indicate on the figures by using **gtext** command the types of noise you encountered in the quantized signal.

I-3) The above system described so far is the DM Transmitter. Using your knowledge about DM theory, propose a receiver block diagram, and specify its building blocks and how to simulate them in MATLAB.

Useful MATLAB Functions: sign, for, end, stairs, stem, plot, gtext, xlabel, ylabel, title.

Part II - Hardware Experimentation:

II.1) DM Transmission & Reception:

1. On the DELTA and PAM-TDM blocks, use three 2-post connectors to connect the M1 signal to the Digital Sampler input, the output of the Sampler to the input of INTEGRATE 2, and the output of INTEGRATE 2 to the input of FILTER 1 (in PAM-TDM). Verify the frequency & amplitude of the signal M1. Also check the frequency of the clock signal.

2. On the oscilloscope, observe and comment on the analog input M1 and the output signal of FILTER 1. **Note:** you should adjust the reference voltage for INTEGRATE 1 so that the output signal from FILTER 1 is maximized and not distorted. This is done by using the positive supply knob on the base unit.
3. Observe the output of the Digital Sampler on the oscilloscope. What does it correspond to? Measure the DM baud rate (or bit rate). Why is it equal to the clock frequency?
4. Use the oscilloscope to simultaneously display the input message signal and the fed-back signal from INTEGRATE 1. Sketch both signals, and explain why INTEGRATE 1 output closely tracks the input analog signal.

II.2) DM Performance Factors:

1. Observe the output of INTEGRATE 1 (or INTEGRATE 2) on the oscilloscope. Is it similar to the “accumulator” integrator used in the Matlab software part of the lab? Explain why they produce “saw-tooth” signals, and not stair-case signals. Hint: think about the typical response of 1st order, RC & Op-Amp integrator.
2. Delta Modulation introduces **quantization noise**. Explain the impact of the integrator’s RC time constant on this noise. Hint: think of the relation between the time constant RC, the integrator slope, and the amount of voltage increase or decrease in a given clock period (i.e., the quantization step size).

Slope overload noise occurs when the input signal varies too fast for the DM bits to track closely. To study this, disconnect M1 and use the external function generator to apply a sinusoidal input signal of variable frequency. Study the impact of increasing the input signal frequency. To do so, observe the DM bits at the output of the sampler, the signal from INTEGRATE 1 (compared to the analog input) and the receiver output from Filter 1 as well. **Note:** you should also keep in mind that Filter 1 is a low-pass filter with 3dB cutoff frequency around 2.6 kHz.

Homework Questions

- Q1.** Explain the role of the receive filter (FILTER 1) in the operation of the DM receiver?
- Q2.** Discuss how the quantization step size affects quantization noise and slope overload noise and why?

Appendix

(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 house keeping 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.