LABORATORY WORK BOOK

For Academic Session _____

Semester _____

DIGITAL COMMUNICATION AND INFORMATION THEORY

(TC-311)

For

TE (TC)

Name: _____

Roll #: _____

Batch: _____

Group: _____

Department of Electronic Engineering
NED University of Engineering & Technology, Karachi

LABORATORY WORK BOOK

For The Course

TC-311 DIGITAL COMMUNICATION & INFORMATION THEORY

Prepared By:

Sunila Akbar (Assistant Professor)

Reviewed By

Dr. Muhammad Imran Aslam (Associate Professor)

Approved By:

Board of Studies of Department of Electronic Engineering

INTRODUCTION

Digital Communication and Information Theory Practical Workbook covers broad range of practicals that describes how to set up the digital communication experimental system, and provides explanations of principles and methods. We tried our best to relate what you are learning in class with what you are taking in the lab.

The practicals of this manual are arranged on modern trainer boards and the objectives are clearly defined. Some of the practicals are based on MATLAB to give an insight of the system design, analysis and simulation in the communication area.

You are required to go through the lab notebook and the suggested reading before coming to the lab. Make sure you bring with you the assignments and lab tasks, neat and tidy presented and enough explanatory for the marks to be awarded failing to which shall result in the deduction of marks. It must have the following:

- Objective
- Clear theoretical concepts & equations used, source code and results of the simulations or plotted data.
- Description of the results or data collected and plotted.
- Conclusions

You are supposed to fill in the observation and the result field at the time of the performance of the experiment and submit it then, get it duly signed and marked.

Each lab task is due until the next lab session.

Recommended readings for each laboratory session:

[1] Haykin, S, *Communication Systems*, John Wiley & Sons, 4th Edition.
[2] Proakis, John G., *Digital Communications*, New York, McGraw-Hill, 3rd Edition.
[3] Sklar, Bernard, *Digital Communications: Fundamentals and Applications*, Englewood Cliffs, N.J., Prentice-Hall,2nd Edition.
[4] Tomasi, W, *Advanced Electronic Communications Systems*, Prentice-Hall International, 5th Edition.

CONTENTS

		at the receiver.		
10		**Quadrature Phase Shift Keying** • To observe QPSK modulation & demodulation.	48-53	
11		**Differential Phase Shift Keying** • To observe DPSK Modulation and Demodulation.	54-57	
12		**Optimal Receiver Design** • To implement Correlation Receiver and Matched Filter using simulation tool.	58-59	
13		**Equalization** Implement 5-tap transversal equalizer using: • Zero-forcing solution • Minimum MSE solution	60	
14		**Channel Capacity and Coding** • To plot the capacity of binary symmetric channel • To plot the mutual information of the binary non symmetric channel • To simulate a Block Code of (6, 3) & (63, 57) order using MATLAB Communication Toolbox	61	
15		**Digital Signal Processors (DSPs)** • Design a very simple project for the Texas Instruments C6000 family of DSP devices using MATLAB® and Simulink®. • Run the project on the Texas Instruments DSK6713.	62-69	

LAB SESSION 01
Sampling

OBJECTIVE:

To examine 'Pulse Amplitude Modulation' and 'Sample and Hold' methods of sampling and to observe the Aliasing effect.

EQUIPMENT REQUIRED:

Digital Data Formatting Work board 53-150 which comprises the following blocks:

- Signal Generator
- Compressor
- Filter
- PAM Generator
- Sample and Hold Circuit
- Analogue to Digital Converter
- NRZ, RZ, BPRZ, SP, AMI Links
- Synchronisation Circuits
- Digital to Analogue Converter
- Expander

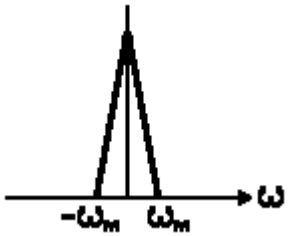
THEORY:

Sampling:

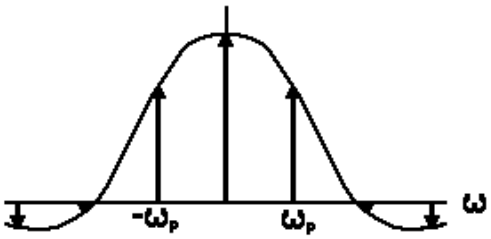
Pulse Amplitude Modulation is achieved by multiplying the signal waveform with a square
wave of a higher frequency.

This results in a waveform similar to the one below.

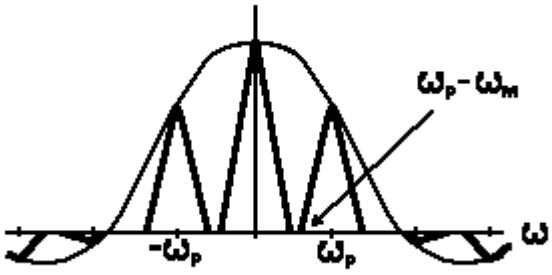
If, in the frequency domain, the source signal looks something like this:



and the square wave looks like this ...



Then the spectrum of the PAM signal will look like this:



From this we can see that the original signal can be recovered exactly from the PAM signal simply by filtering out the higher frequency components of the signal with a low pass filter.

However, there is a condition that must be satisfied for this to be possible. The filtering will not work if the harmonic components of the modulated waveform are not sufficiently far from the baseband component that the two do not overlap.

i.e., $(\omega_p - \omega_m) > \omega_m$ must be true.

This leads to the conclusion that $\omega_p > 2\omega_m$

In other words, for the original signal to be correctly recovered by low pass filtering, the sampling square wave frequency must be greater than twice the highest frequency component in the signal that is being sampled.

The value of $2\omega_m$ is called the Nyquist frequency.

If the signal is sampled at a frequency less that the Nyquist frequency for that signal, the baseband and harmonics overlap, and aliasing occurs.

It is interesting to note that it is only the frequency of the sampling square wave that effects the recoverability of the original signal, not the mark:space ratio.

The 'on' time of the square wave will only effect the size of the signal after the low pass filter - the shorter the 'on' time, the smaller the recovered signal.

Pulse Amplitude Modulation:

A Pulse Amplitude Modulated signal is a signal that has had a proportion of its waveform removed at regular intervals leaving behind a series of pulses whose amplitudes describe the original waveform.

The original modulated waveform can be extracted from the PAM signal by simply passing it through a low pass filter which smoothes out the pulses.

Pulse Amplitude Modulation is produced by multiplying a signal with a square wave of constant amplitude.

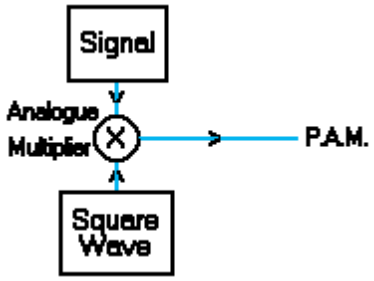


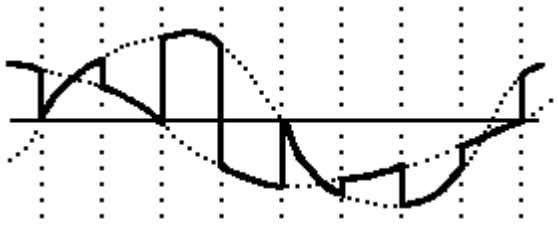
Figure 1.1: PAM Sampling

At the points where the square wave has zero amplitude, the resultant PAM signal will also have zero amplitude.

At the points where the square wave peaks, the PAM signal will have the shape of the original signal, with an amplitude that is dependent on the instantaneous value of the signal.

In the portions of the PAM waveform where there is zero amplitude, there is no information about the waveform carried.

These gaps in the waveform can be filled by the information carrying portions of another completely different PAM waveform. This is a technique called multiplexing.



The single waveform made up of two PAM signals can be transmitted over a communications link and split up again, or demultiplexed, at the receiver into the two separate PAM waveforms.

These waveforms can then be filtered to reproduce the two original signals, having been transmitted using the space required for just one of them.

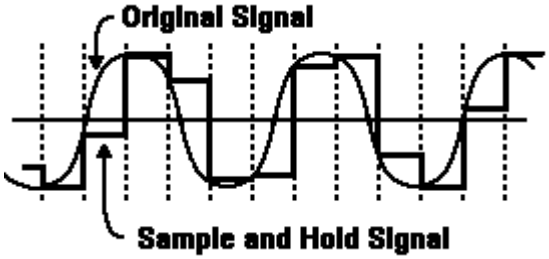
A sine wave signal is multiplied by a square wave sampling signal using an analogue

multiplier circuit to produce a PAM signal. The frequency of the original signal can be altered. The sampling signal has been arranged to be at the frequency needed to chop up the signal into eight pulses per cycle of its sine wave.

Sample and Hold:

A waveform can be represented by a sequence of pulses, `snapshots` of the waveform at equally spaced intervals. These pulses are known as samples. Provided that there are enough samples, i.e., that the sample frequency is high enough, the original signal can be completely recovered from its sampled equivalent.

To restore a PAM signal, it is only necessary to sample the PAM waveform when it is non-zero, and then filter the sampled waveform to reproduce the original signal. To make the recovered signal less vulnerable to noise it is useful to hold the last sample until the next one is taken. This is known as sample and hold.



A signal is sampled by connecting it quickly to a capacitor via a switch. While the signal is connected, the capacitor is charged until it soon reaches the level of the signal. The time constant of the charging circuit is made as small as possible so that the time taken to reach the signal level is minimal.

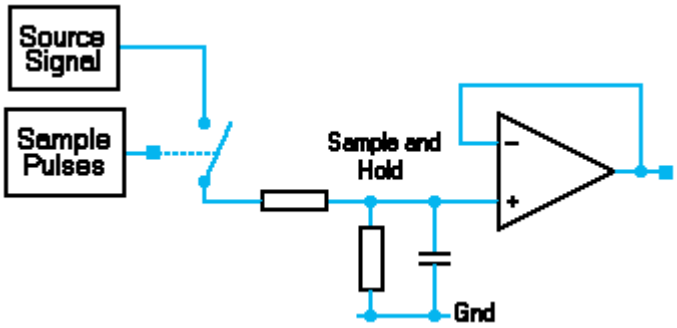


Figure 1.2: Sample and Hold Circuit

When the switch disconnects the signal from the capacitor, the level that the capacitor had reached at that point is held due to the high impedance input of the buffer amplifier.
The waveform that is seen at the output of the buffer amplifier resembles a series of steps, the leading edges of which are rounded off due to the capacitor being charged to the next level at this point.

In the practical you will examine a PAM signal which is sampled by a sample and hold circuit. This circuit is driven by a periodic sample pulse at the sample frequency. The sample frequency is matched automatically to the frequency of the PAM signal's pulses. The PAM signal can be changed in frequency, and the length of the sample pulse can be

altered.

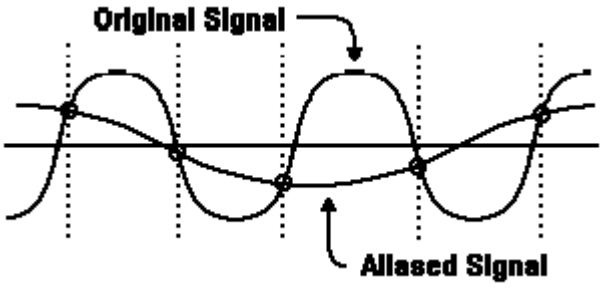
Aliasing:

A waveform can be represented by a sampled waveform made up of samples of the original signal taken at equally spaced intervals. Provided that there are enough samples, ie. that the sample frequency is high enough, the original signal can be completely recovered from its sampled equivalent.

If the samples are not taken at a high enough rate, the original waveform cannot be recovered because aliasing occurs.
To ensure that the sampled signal contains enough information to enable the original signal to be regenerated without distortion, the frequency at which samples are taken needs to be at least twice that of the highest frequency component in the original signal.

This minimum sampling frequency is called the Nyquist frequency.

If samples are taken at slightly less than the Nyquist frequency, aliasing occurs. The result of this is that the recovered signal appears to be one of much lower frequency, as shown below.



PROCEDURE:

Pulse Amplitude Modulation:

Set all of the potentiometer controls to their mid positions. Observe the analogue source signal monitor point **16**, the sampling square wave monitor point 2 and the Pulse Amplitude Modulated signal monitor point **18**. Use the oscilloscope and spectrum analyser to compare these signals. Alter the analogue source, **Signal frequency (2)** and **Signal level (1)**, controls and observe the results.

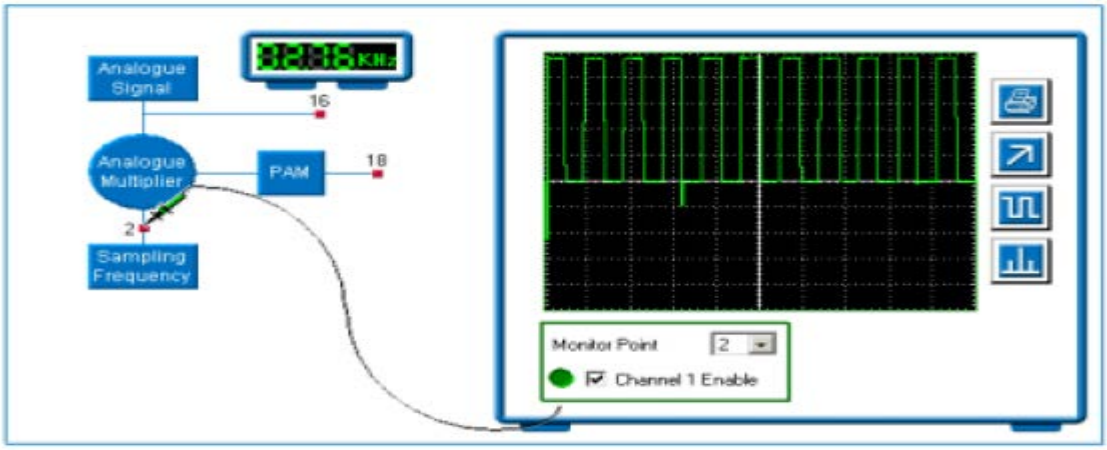


Figure 1.3

OBSERVATIONS AND RESULTS:

Set the **Signal frequency (2)** and **Signal level (1)** controls to their mid positions.

(a) Use the large display to compare the frequency spectrums of the input sine wave, the sampling signal and the PAM signal. Can you identify any similarities between them?
(b) Why are there two peaks on the PAM spectrum at the positions where there are single peaks on the sampling square wave spectrum?
(c) What would happen to the pairs of peaks on the PAM spectrum if the sampling frequency remained the same, and the source signal frequency was increased?

Sample and Hold:

Observe the PAM signal monitor point 18 using the oscilloscope. Vary the **Signal frequency (2)** and the **Signal level (1)** controls. Observe the Sample Pulses monitor point **3**. Investigate the effect of altering the **Sample time (4)** control.

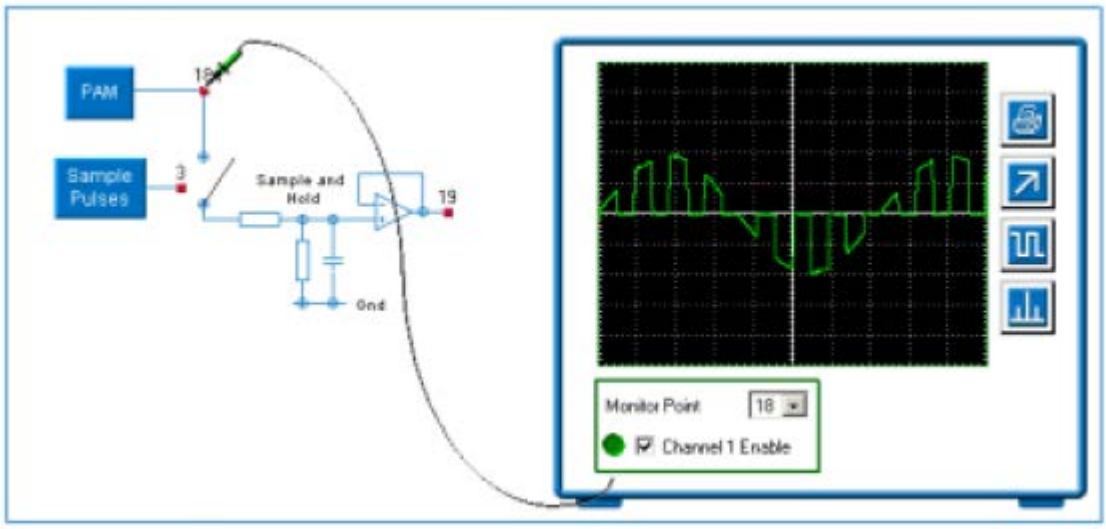


Figure 1.4

OBSERVATIONS AND RESULTS:

(a) Why is there a buffer amplifier at the output of the sample and hold circuit?
(b) What happens when the sample time is very short?
(c) Why does this happen?
(d) What happens when the sample time is long and the frequency of the PAM signal is high?
(e) Why do you think this happens?
(f) What can you say about the ideal requirements for the sample and hold circuit in terms of its sample time and its charging time constant?

Aliasing:

Set all of the potentiometer controls to their mid positions. Observe the analogue source signal monitor point **16** , the Sampled Signal monitor point **19** and the Filter output monitor point **23** with the oscilloscope. Adjust the **Signal frequency (2)** control over its whole range and examine the resulting sampled signals.

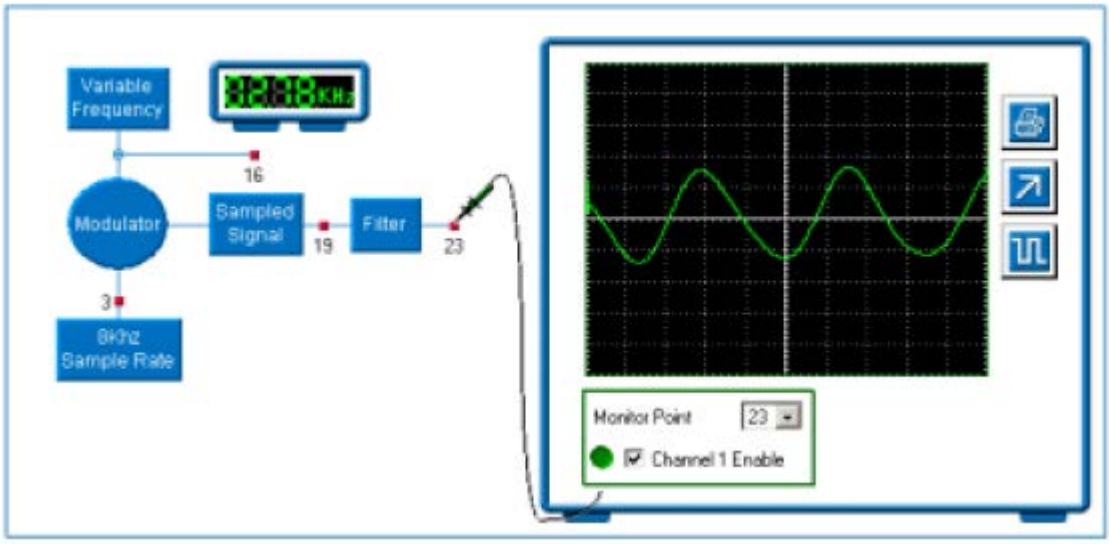


Figure 1.5

OBSERVATIONS AND RESULTS:

(a) Slowly sweep the frequency control for the input signal to the modulator from minimum to maximum and observe the sampled output signal. Describe what happens to the output signal.
(b) When adjusting the frequency of the input signal the sampled output signal appears to become a square wave at a certain frequency. What is the frequency of the input signal when this occurs?
(c) What do you think is happening at this point?

LAB SESSION 02
Pulse Code Modulation

OBJECTIVE:

- To examine the operation of 12-bit and 4-bit linear PCM coders and decoders
- Examine quantization noise
- To observe the eye pattern in codec communication system

EQUIPMENT REQUIRED:

- Module T20B
- +/- 12V dc power supply
- Oscilloscope

THEORY:

Introduction:

According to the sampling theorem, an analog signal s(t) can be converted in a train of pulses, sampling instantaneous voltage values at constant intervals equal to the sampling (or frame) period $\tau = 1/(2.f_M)$, where f_M indicates the maximum frequency of the signal s(t). The PAM signal is obtained this way.

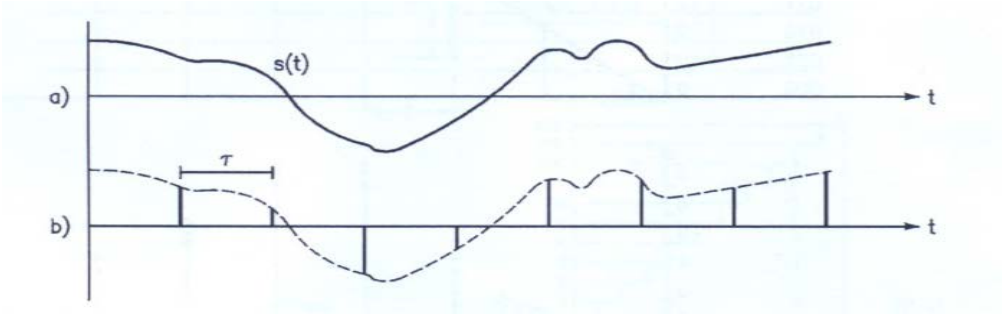


Figure 2.1 PAM waveform

The amplitude information contained in each PAM sample is converted into a binary value of fixed length. Figure 2.2 shows a single channel PCM communication system,

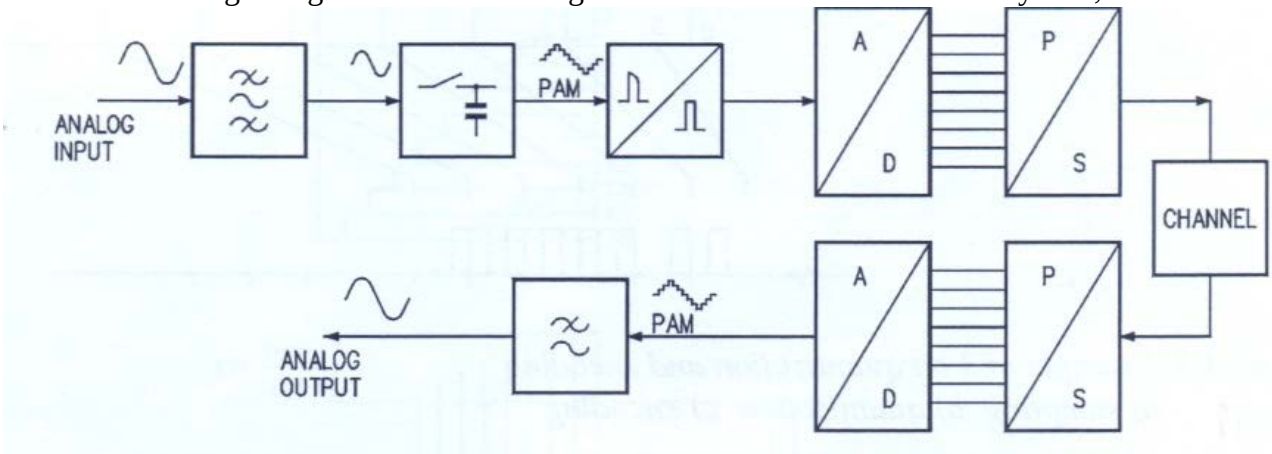


Figure 2.2 PCM transmission system

The input analog signal goes through a low pass anti aliasing filter and gets to the sampler. The quantiser assigns one voltage value to the pulses whose amplitudes are included within the given interval. Then the quantized pulses are applied to an A/D converter which carries out binary encoding of each pulse. The parallel A/D output is converted into a serial output by the next parallel/serial converter. Each bit is represented in an NRZ way. The duration of each bit equals the frame period τ, divided by the bit number n with which A/D conversion is performed. For instance if τ=125μ and n=8, the duration of each bit is 125/8=15.625μs.

The PCN serial signal is transmitted across a transmission channel and gets to a serial/parallel converter whose output digital word is converted into an analog value by the following D/A converter. The converter output is a step signal which approximates the starting analog signal. The next low pass KHz filter cleans the signal and supplies the starting waveform.

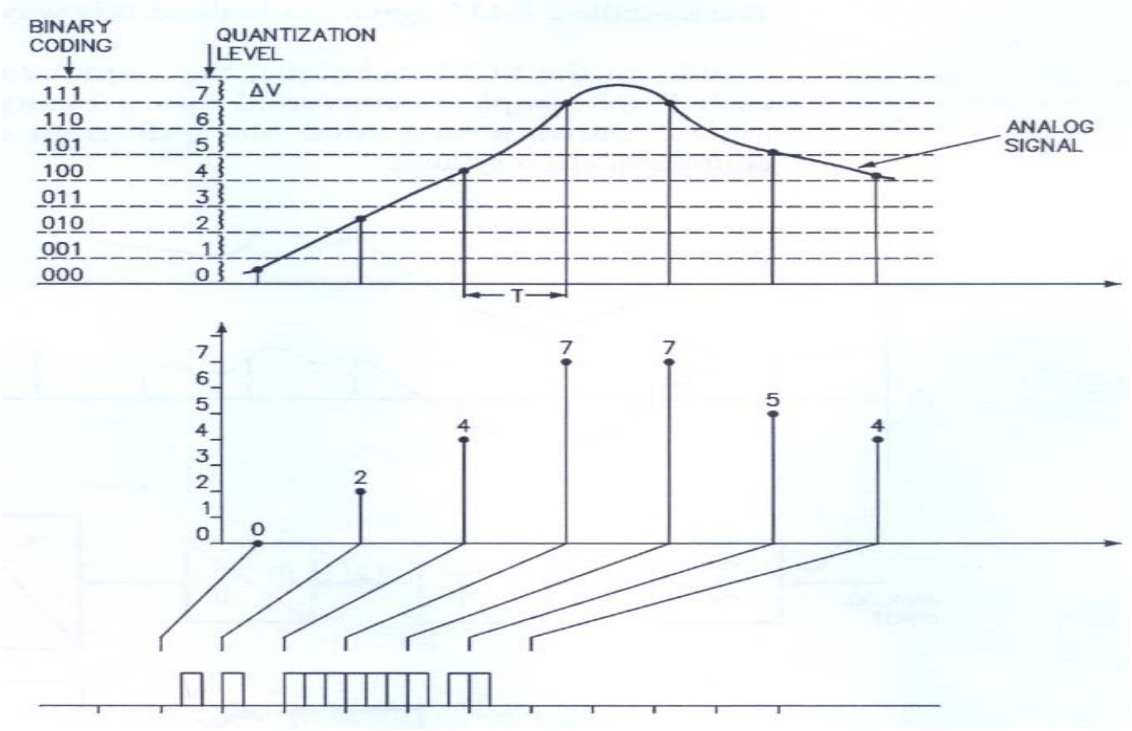


Figure 2.3 3-bit Quantization and Encoding
a. Sampling b. Quantization c. Encoding

Sampling and Quantization

Sampling consists in picking the instantaneous values of the analog signal with a repetition period depending on the signal itself. In case of the telephone signal, the sound channel maximum frequency is 4 KHz, and the sampling period or the frame interval is 125μs.

Sampling supplies PAN pulses of variable amplitude in continuous way. Quantization assigns a well defined voltage value. A finite number of discrete values for the next A/D encoding stage is thus obtained. Such values are called *Quantization levels*.

In case of *linear quantization* the difference between the two adjacent levels is uniform

all along the input signal. Figure 2.4 shows the quantization curve related to 256 levels. The number of levels N depends on the number of bits n of the signal encoded according to the relation $N=2^n$

Voltages valve differing in value:

$$\Delta V=V_{max}/128$$

Quantization Noise

Quantization noise is the difference between the analog signal and the corresponding quantized value. The relation between the signal S and the quantization noise N_Q depends on the signal amplitude as the ΔV value is uniform all along the input signal. This implies that strong signals show a better S/ N_Q ratio than weak signals. In order to obtain a uniform S/ N_Q all along the signal compression technique is used which leads to *non-linear PCM encoding.*

Non-Linear Encoding

Analog Compression and Expansion

When an analog signal is transmitted it is compressed before it gets to the PCM coder proper. When the signal is received it is expanded in the PCM decoder output, according to a law which is complementary to the compression one.
The compression law used in telephony are the law A=87.6 and μ=255 as show in figure 2.6. The former is adopted as a European standard and the latter is an American standard.

Digital Compression and Expansion

Analog compression was used in first PCM systems and was subsequently replaced by digital techniques. Digital compression is performed after PCM conversion and is at a digital signal level. Likewise digital expansion is performed before PCM decoding. Today's digital compression PCM systems use 12-bit code linear compression. This amount of bits is subsequently compressed down to 8.

Differential PCM encoding

When analog signals are PCM encoded(voice, images etc) many samples in a row may show the same level of quantization. Consequently many identical PCM codes are transmitted, which is redundant in received signal reproduction. Differential PCM exploits the redundancy between adjacent samples.

DPCM allows to transmit the code concerning the difference between the two subsequent samples and not the code related to the current sample. Since the amplitude of the difference between between samples is lower than the amplitude of the sample itself, DPCM requires a lower bit than the conventional PCM.

There are several varieties of DPCM, in terms of functions. Some systems uses *adaptive* encoding in order to generate a more accurate and efficient prediction of the input signal. Adaptive encoding may be performed as follows:

- More previous samples are used to generate an analog signal

- Each sample is assign a variable weight which depends on the average power level of the input signal
- According to the input level the quantization step amplitude of the A/D and D/A coders is also adjusted.

Intersymbol Interference

Consider a digital signal with bit interval T. The transmission channel attenuates and broadens each transmitted symbol. When the symbol is received, it occupies not only its interval but partially also the interval of the other ones. This effect is called ISI which indicates the interference caused by each symbol on the following ones. High ISI can lead to incorrect recognition of the received signal.

Eye Diagram

ISI can be detected with an oscilloscope which results in figure called *eye diagrams.* The time axis is synchronized by the symbol frequency, where as the signal is applied to the vertical axis. Owing to the persistence of vision a considerable amount of superimposed pulses is noticed.

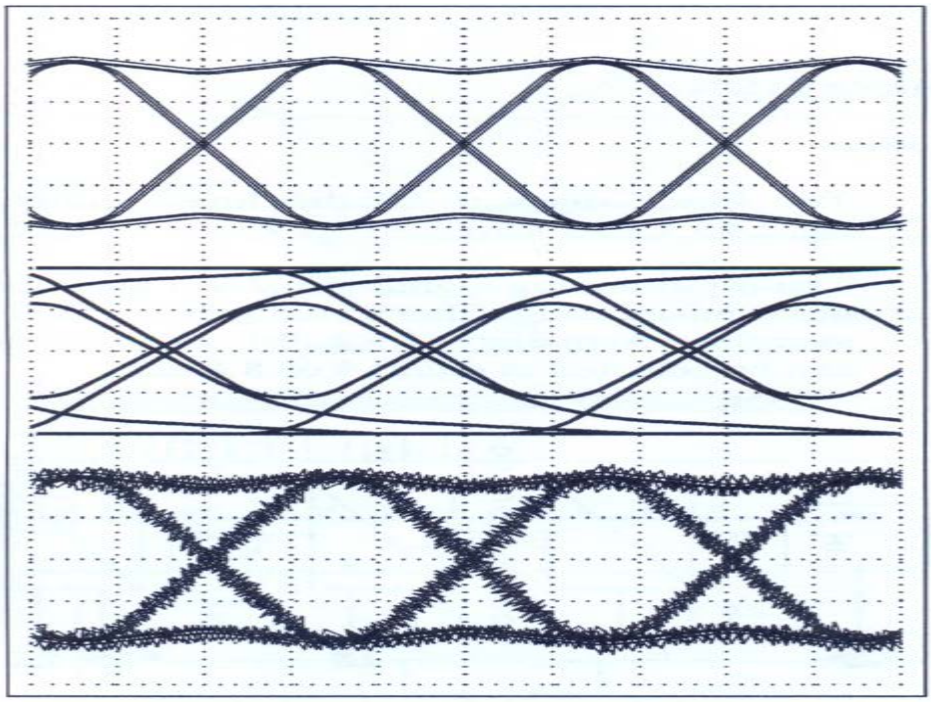


Figure 2.4 Eye diagrams
(a) Zero intersymbol interference (b) accentuated intersymbol interference (c) noise

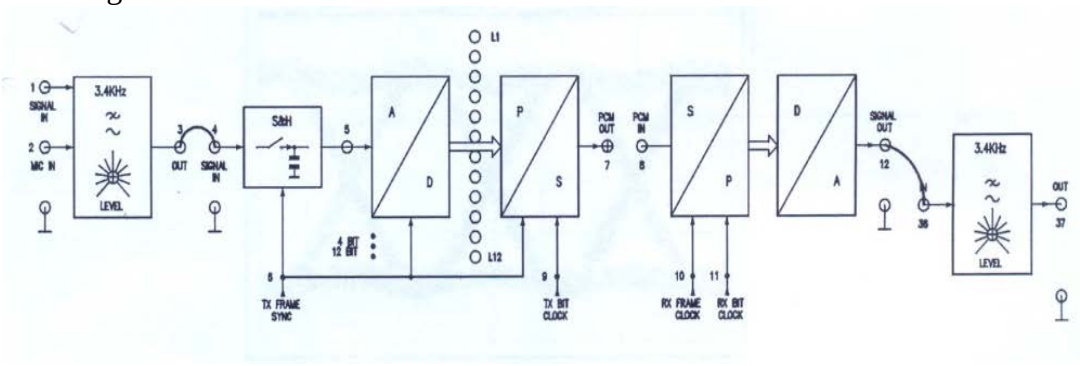
Procedure:

The input analog signal goes though a 3.4 KHz LPF (anti aliasing filter) and gets to the sampler (Sample & Hold). The sampling frequency is determined in the TIMING section and can be selected either 4 or 8 KHz. The sampled signal is applied to the ADC which carries a 12 or 4 bit encoding selectable by jumper. The converter range goes from -5V dc to +5V dc (all 1s). The parallel output of ADC is converted into serial output by

following parallel to serial converter. Considering f_s sampling frequency and the amount of bits N per sample, the PCM bit transmission speed is $v=N.\ f_s$ (With an 8KHz and 12bits/sample sampling frequency, a 96kbits/sec is obtained.

a) PCM signal coding and decoding

- Prearrange the circuit as follows



- Sampling frequency: 8KHz
- 12-bit encoding
- Supply the +/- 12V
- Connect TP38-TP1; TP3-TP4; TP7-TP8 AND TP12-TP36. Adjust the generator and the transmission filter LEVEL in order to obtain a signal with an amplitude of $5V_{pp}$ in TP4.
- Connect the oscilloscope to TP4 and TP37. Adjust the reception filter level so as to obtain signals of the same amplitudes. Notice that there is a phase shift between the transmitted signal and the received signal. This is due to the delays caused by the modulation and demodulation process.
- Move the oscilloscope from TP4 to TP12, notice the effect of the reception filter.
- Repeat the same procedure for 4-bit operation. What did you observe?

b) Quantization noise

- Prearrange the circuit as follows
- Sampling frequency: 8KHz, 12-bit encoding
- Supply the +/- 12V
- From TP38 take a signal with amplitude of $5V_{pp}$ and apply it to TP4.
- Connect TP4-TP13; TP14-TP15 and TP15-TP16
- Connect the oscilloscope to TP15 and TP17, and adjust the PHASE ADJ and GAIN in order to obtain two signals in phase and of the same amplitude
- Connect the oscilloscope to TP18, the difference between the analog and the quantized signal is detected, i.e. the quantization noise
- Turn the sampling frequency to 4 KHz and examine the signal again, adjust PHASE ADJ. describe how quantization noise changes.

c) Waveforms and eye diagram

- Perform the following connection: TP43-TP44; TP45-TP56; TP57-TP46; TP47-TP48; TP51-TP52. See figure 1.8
- Preset the line on 100 KHz and bring attenuation to its minimum value. Turn noise to zero

- Supply +/- 12V
- From TP38 take a signal of $2V_{pp}$ amplitude and apply it to TP40
- Examine the waveforms all along the PCM signal;
  TP43-Codec NRZ output, TP45- Transmission filter output, TP57- Line output, TP47-Reception filter output, TP49-reception amplifier output, TP50-Decision making element output
- Synchronize the oscilloscope with TP53 (RX BIT CLOCK), turn the time base to 5µs/div and examine the waveform in TP49. An eye diagram is obtained
- Lower the line band pass(jumper to 32KHz) and notice the eye diagram
- Remove the transmission and reception filter and observe the eye diagram, turn line attenuation to maximum
- Restore the previous connections, connect the oscilloscope to TP40 and TP55, and adjust the PHASE ADJ of the TIMING section so the output signal TP55 is same as input signal TP40

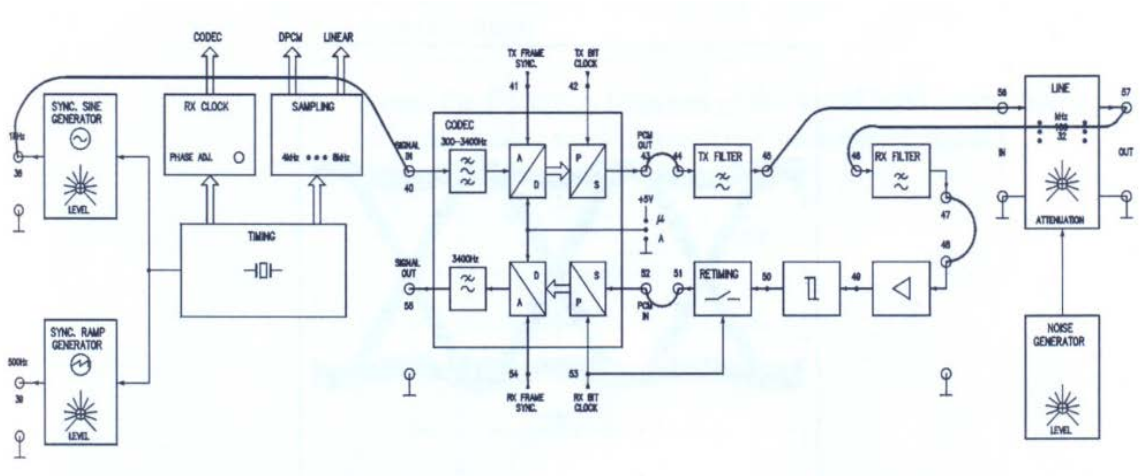


Figure 2.5 PCM communication system

OBSERVATIONS AND RESULTS:

LAB SESSION 03
Binary Signaling Format

OBJECTIVE:

To examine RZ (Return to zero), NRZ (Not Return to zero), MANCHESTER and AMI digital data formats.

EQUIPMENTS REQUIRED:

Modulation and Keying Workboard 53-160 which comprises the following blocks:

- Data Generator
- ASK, FSK, PSK, QPSK, DPSK, DQPSK Links

THEORY:

A binary coded waveform can be represented by a number of different data formats using either unipolar or bipolar signals. Different data formats can be distinguished by the bandwidth required and the characteristics available for channel.

Different data formats have differing minimum bandwidths associated with them. The maximum number of bits that can be transmitted with a single High-Low cycle is two for NRZ, and one for RZ, Bi-Phase, Bipolar RZ and AMI i.e., NRZ can transmit 2 bps/Hz (bits per second per Hertz). The other formats can transmit 1 bps/Hz.

The bps/Hz term is called the bandwidth efficiency.

For a binary PCM system using *n* quantizing levels the minimum required bandwidth is*:*
B
>= 1/2 (log2n / T) Where 1/T is the sample rate.

Binary data formats

(a) NRZ

An NRZ coded signal is high for '1' code bits and low for '0' code bits

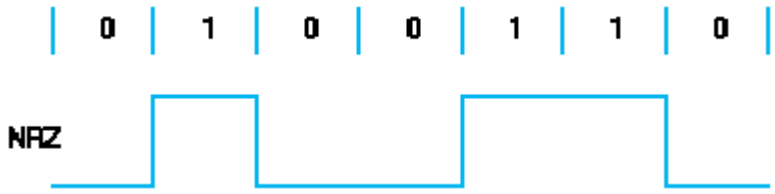


Figure 3.1 NRZ encoding

This is a very simple coding, as nothing needs to be done at all to the basic binary serial data.

Bit Clock Recovery

The NRZ signal contains no periodic component at the bit clock frequency. The rising and falling edges of the data are used to generate pulses. The pulses trigger a bit clock monostable which has an 'on' period approximately half that of the required bit clock frequency. The bit rate must be a known frequency, in this case 64kBits/sec.

The monostable pulses drive a Phase Locked Loop, the VCO of which has its range centred on the required bit clock frequency. The pulses keep the VCO locked in phase and frequency to the data. The PLL acts like a flywheel, keeping a stable, regular bit clock that is largely immune to jitter in the recovered signal that drives one of the phase comparator inputs.

The output of the VCO is the recovered bit clock.

Data Recovery

The received data stream is integrated after it has been squared up by the threshold comparator. This is to minimize the effects of noise by summing the data over the whole bit period.

Just before the end of the bit period, the integrator output is sampled at a time determined by the recovered bit clock. This squared sampled data stream is the final recovered serial data.

Problems associated with NRZ

The NRZ data format is not very good because:

- Its level drifts when it is used on an AC coupled link with data that has varying proportions of '1's and '0's in it. The threshold comparator in the receiver may mistakenly translate '1's as '0' or '0's as '1's if the signal's DC offset drifts too far.
- It does not have a regular level transition at the bit clock frequency, so a more complicated bit clock recovery process involving a bit clock monostable is required.
- For long streams of '1's or '0's, there are no level transitions in the data, so no monostable pulses are generated. This means that the PLL will receive no control signal for long periods and may drift out of lock, resulting in corruption of data.

The complexity of the bit clock and data recovery circuitry is important. Many receivers may be required not only at the end of the communications link, but also along its length, regenerating and retransmitting the weakened signal periodically. If the recovery processes require expensive or over complicated and hence more unreliable circuits then they will be less suitable for use as 'repeaters', which ideally should be cheap and require little maintenance.

PROCEDURE:

Set all controls to mid position. Select the User Data Word or ADC Data using the button and ***adjust dc (5)*** control to create a serial data word. Observe this at monitor point ***4*** . Select an AC or DC coupled link. Observe the coded data word before and after the link monitor points ***21*** & ***20.*** Note the signal offset for different data patterns for AC at monitor point ***20***.

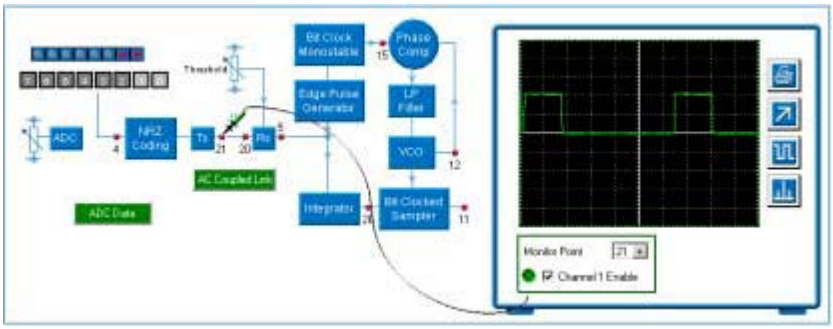


Figure 3.2

View the data at monitor point ***5*** . Adjust the ***Receiver threshold (8)*** control for a stable signal. View the signals in the bit clock recovery circuit at monitor points ***15*** & ***12*** , adjust the ***Synchronous bit clock (9)*** control for PLL lock. Examine the signals in the data recovery path at monitor points ***26*** & ***11.*** Answer the questions provided. You will need to make some observations in order to answer them.

OBSERVATIONS AND RESULTS:

(d) For an AC coupled link, what happens to the signal at monitor point ***20*** for various data words?
(e) Why is this a problem?
(f) What happens to the recovered bit clock for patterns of 00000000 and 11111111?
(g) Why is this a problem?

(b) RZ

A RZ coded signal is high for '1' code bits and low for '0' code bits, but its level returns low half way through the bit period.

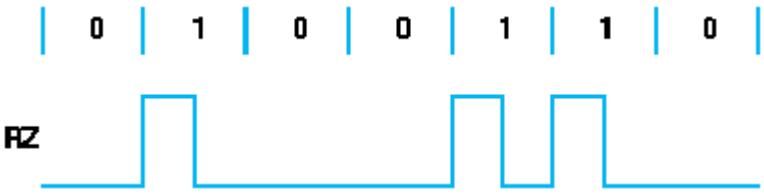


Figure 3.3 RZ encoding

This is a fairly simple coding. The basic binary serial data is AND'ed with the transmitter bit clock to produce RZ data. This means that the coded data contains a component at bit clock frequency which simplifies bit clock recovery.

Bit Clock Recovery

The squared data drives a Phase Locked Loop, the VCO of which has its range centered on the required bit clock frequency. The pulses keep the VCO locked in phase and frequency to the data. The PLL acts like a flywheel, keeping a stable, regular bit clock that is largely immune to jitter in the recovered signal that drives one of the phase comparator inputs. The output of the VCO is the recovered bit clock.

Data Recovery

The received data stream is integrated after it has been squared up by the threshold comparator. This is to minimize the effects of noise by summing the data over the whole bit period. Just before the end of the middle of the bit period, the integrator output is sampled at a time determined by the recovered bit clock. This squared sampled data stream is the final recovered serial data.

Problems associated with RZ

The RZ data format is not very good because:

- Its level drifts when it is used on an AC coupled link with data that has varying proportions of '1's and '0's in it. The threshold comparator in the receiver may mistakenly translate '1's as '0' or '0's as '1's if the signals DC offset drifts too far.
- For long streams of '0's, there are no level transitions in the data. This means that the PLL will receive no control signal for long periods and may drift out of lock, resulting in corruption of data.

PROCEDURE:

Set all controls to mid position. Select User Data Word or ADC Data using the button and adjust ***dc (5)*** control to create a serial data word. Observe this at monitor point ***4***. Select an AC or DC coupled link. Observe the coded data word before and after the link monitor points ***21*** & ***20.*** Note the signal offset for different data patterns for ac at monitor point ***20***.

View the data at monitor point ***5.*** Adjust the ***Receiver threshold (8)*** control for a stable signal. View the signals in the bit clock recovery circuit at monitor points ***15*** & ***12, adjust*** the ***Synchronous bit clock (9)*** control for PLL lock. Examine the signals in the data recovery path at monitor points ***26*** & ***11.*** Answer the questions provided. You will need to make some observations in order to answer them.

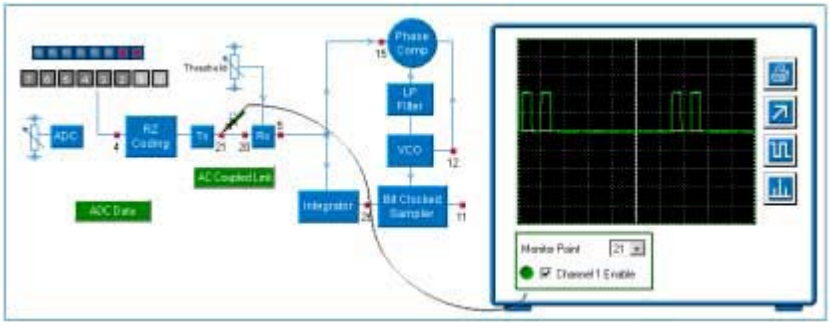


Figure 3.4

OBSERVATIONS AND RESULTS:

1. On an AC coupled link, what happens to the signal at monitor point ***20*** for different data word patterns?
2. What happens to the ***bit clock*** for ***data words*** of:
   a. 11111111?
   b. 00000000?

(c) Bi-phase (Manchester)

A Bi-Phase coded signal is high for half a bit period then low for half a bit period for '1' code bits, and low for half a bit period then high for half a bit period for '0' code bits.

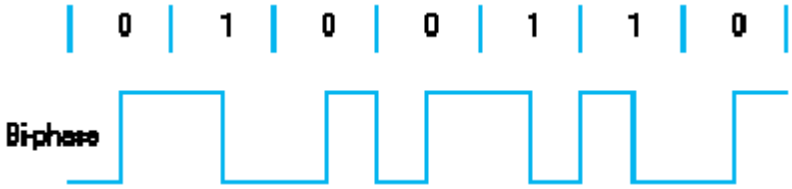


Figure 3.5 Manchester encoding

This is also a fairly simple coding; the basic binary serial data is XNOR'ed with the transmitter bit clock.

Bit Clock Recovery

The Bi-Phase signal has a periodic component at the bit clock frequency for most data patterns. However, a pattern of 10101010 will result in a square wave coded signal with a frequency of half the bit rate. To avoid problems, the rising and falling edges of the data are used to generate pulses for a bit clock monostable.

The pulses trigger the monostable which has an 'on' period approximately half that of the required bit clock frequency. The bit rate must be a known frequency, in this case 64kBits/sec.

The monostable pulses drive a Phase Locked Loop, the VCO of which has its range centred on the required bit clock frequency. The pulses keep the VCO locked in phase and frequency to the data. The PLL acts like a flywheel, keeping a stable, regular bit clock that is largely immune to jitter in the recovered signal that drives one of the phase comparator inputs.

The output of the VCO is the recovered bit clock.

Data Recovery

After it has been squared up by the threshold comparator, the received data stream is XOR'ed with the receiver bit clock. It is then integrated to minimize the effects of noise by summing the data over the whole bit period.

Just before the end of the bit period, the integrator output is sampled at a time determined by the recovered bit clock. This squared sampled data stream is the final recovered serial data.

Problems associated with Bi-Phase Coding

The Bi-Phase data format is not so bad, although:

- It does not have a regular level transition at the bit clock frequency, so a more complicated bit clock recovery process involving a bit clock monostable is required.

PROCEDURE:

Set all controls to mid position. Select User Data Word or ADC Data using the button and adjust ***dc (5)*** control to create a serial data word. Observe this at monitor point ***4***. Select an ac or dc coupled link. Observe the coded data word before and after the link monitor points ***21*** & ***20***. Note the signal offset for different data patterns for AC at monitor point ***20.***

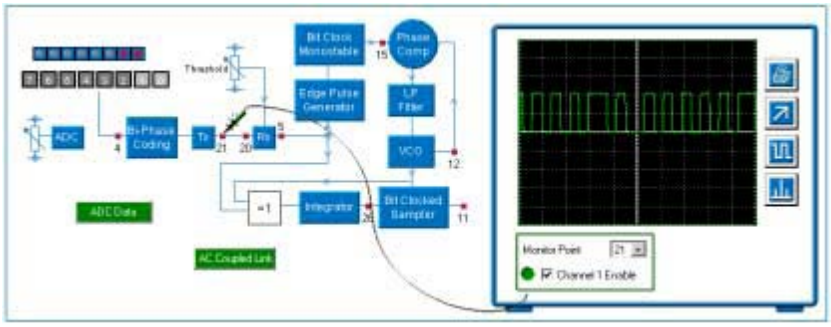


Figure 3.6

View the data at monitor point ***5.*** Adjust the ***Receiver threshold (8)*** control for a stable signal. View the signals in the bit clock recovery circuit at monitor points ***15*** & ***12*** , adjust the ***Synchronous bit clock (9)*** control for PLL lock. Examine the signals in the data recovery path at monitor points ***26*** & ***11.*** Answer the questions provided. You will need to make some observations in order to answer them.

OBSERVATIONS AND RESULTS:

1. Why is there no change in the level of the signal at monitor point ***20*** for dc and ac coupled links with different data words?
2. What happens to the signals at monitor points ***12*** and ***15*** for data words of:
   a. 11111111?
   b. 00000000?
   c. 10101010?

(d) AMI (Ternary)

An AMI coded signal is low for '0' code bits and alternately positive or negative for half a bit period then zero for half a bit period for '1' code bits.

Figure 3.7 AMI encoding

Bit Clock Recovery

The AMI signal contains a component at the bit clock frequency. The receiver has positive and negative threshold outputs for the positive and negative '1's. These two outputs are XORed to produce a stream of pulses at the bit clock rate. However, there will be gaps in the waveform for '0's in the data stream. The stream of pulses drives a Phase Locked Loop, the VCO of which has its range centered on the required bit clock frequency. The pulses keep the VCO locked in phase and frequency to the data. The PLL acts like a flywheel, keeping a stable, regular bit clock that is largely immune to jitter in the recovered signal that drives one of the phase comparator inputs. The output of the VCO is the recovered bit clock.

Data Recovery

The two '1's thresholded outputs from the receiver are XOR'ed together and then integrated. This is to minimize the effects of noise by summing the data over the whole bit period. Just before the end of the middle of the bit period, the integrator output is sampled at a time determined by the recovered bit clock. This squared sampled data stream is the final recovered serial data.

Problems associated with AMI

The AMI data format is very good however:

- For long streams of '0's, there are no level transitions in the data. This means that the PLL will receive no control signal for long periods and may drift out of lock, resulting in corruption of data.

PROCEDURE:

Set all controls to mid position. Select User Data Word or ADC Data using the button and adjust ***dc (5)*** control to create a serial data word. Observe this at monitor point ***4*** . Select an ac or dc coupled link. Observe the coded data word before and after the link monitor points ***21*** & ***20***. Note the signal offset for different data patterns for ac at monitor point ***2***

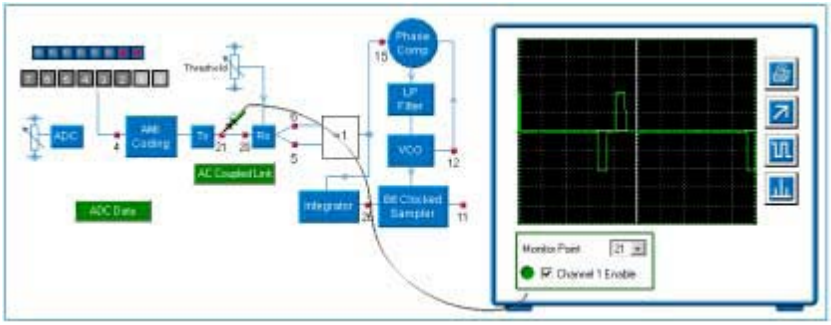


Figure 3.8

View the data at monitor point ***5***. Adjust the ***Receiver threshold (8)*** control for a stable signal. View the signals in the bit clock recovery circuit at monitor points ***15*** & ***12***; adjust the ***Synchronous bit clock (9)*** control for PLL lock. Examine the signals in the data recovery path at monitor points ***26*** & ***11***. Answer the questions provided. You will need to make some observations in order to answer them.

OBSERVATIONS AND RESULTS:

1. Are there any problems when an ac link is introduced?
2. Are there any problems for ***data words*** of:
   a. 11111111?
   b. 00000000?
   c. 10101010?

TASKS:

1. Given the binary sequence b = f1;0;1;0;1;1g; sketch the waveforms representing the sequence by using the following line codes:
   a. Unipolar & Bipolar NRZ
   b. Unipolar & Bipolar RZ
   c. Manchester

   Assume unit pulse amplitude and use binary data rate Rb = 1.5 kbps.
   Write MATLAB codes for the above set of waveforms.

2. For the above set of line codes determine which will generate a waveform with no dc component regardless of binary sequence represented. Why is the absence of a dc component of any practical significance for the transmission of waveforms?

3. For a baseband data communications channel with usable bandwidth of 10 kHz, what is the maximum binary data rate for each of the line codes examined in part 1.

LAB SESSION 04
Digital one way Telephone Link

OBJECTIVE:

To produce a digital link one way telephone using ADC (Analog to Digital converter) and DAC (Digital to Analog Converter).

EQUIPMENT REQUIRED:

- DCS297A Data source
- DCS297H Data receiver
- DCS297J Low pass filter
- DCS297K Audio module
- DCS297M Power supply
- Function generator
- Oscilloscope

THEORY:

Using microphone at the transmitting end and speaker at the receiving end a digitally linked one way telephone can be established. Analog input to data source will be transmitted as a binary coded signal by ADC to data receiver module, where DAC will reconstruct the original waveform from the sequence of discrete steps.

PROCEDURE:

- Connect modules, function generator and oscilloscope as shown in Figure 4.2.
- On data source module set the format switch to ‘8data bits’ and data source switch to ‘ADC’.
- Set function generator to produce a sine waveform, 4Vp-p at 500Hz.
- Switch audio module to ‘speaker’ and level control fully anti-clockwise.
- Adjust oscilloscope to obtain traces of both function generator sine wave input and data receiver output.
- Note data receiver output is not a continuous sine wave but rises and falls in steps of equal time duration.
- Adjust level control on audio module to obtain a tone of comfortable volume.
- Note that steps visible on output waveform donot produce audible distortion at 500Hz. This is due to limited response of speaker to frequencies above 5kHz.
- Decrease input frequency to 100Hz and increase in steps per cycle of waveform.
- Increase input frequency steadily and note that number of steps per cycle decreases until around 5kHz output waveform is no longer recognizable. Also pitch of audio tone ceases to rise above this frequency.
- Disconnect function generator from input of data source and connect audio module as ‘microphone’ and level control clockwise to mid position.

- Turn speaker level control clockwise until a howling noise is heard, turn control anticlockwise until howling stops. System is now acting as one way telephone.

OBSERVATIONS AND RESULTS:

- Observe the signal's waveform at the output of each block and compare.
- Observe the clarity of speech at the receiving end speaker.
- Which unwanted effects are monitored at the speaker output at different levels of control? And Why?

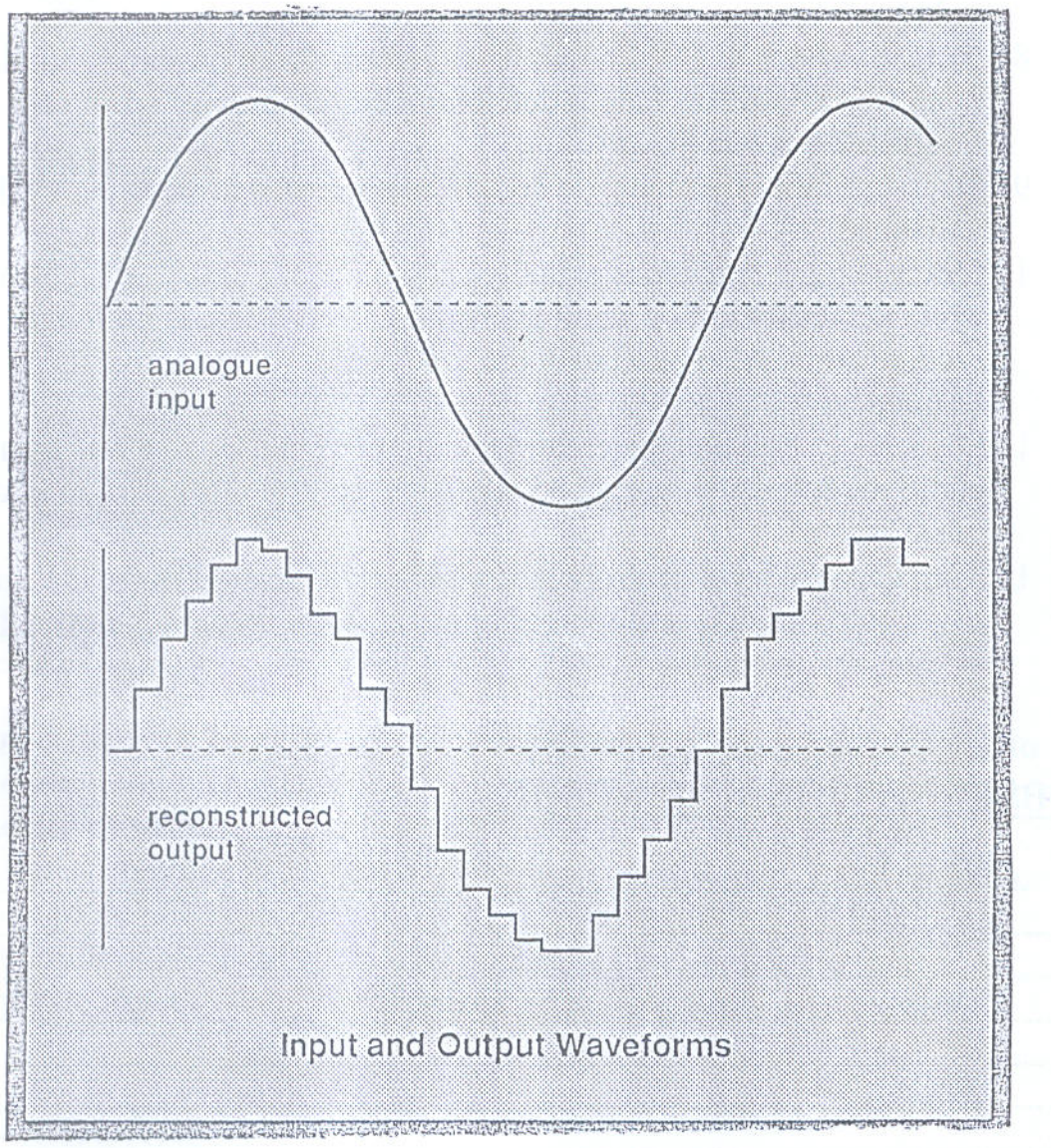


Figure 4.1: Input and Output waveforms of ADC

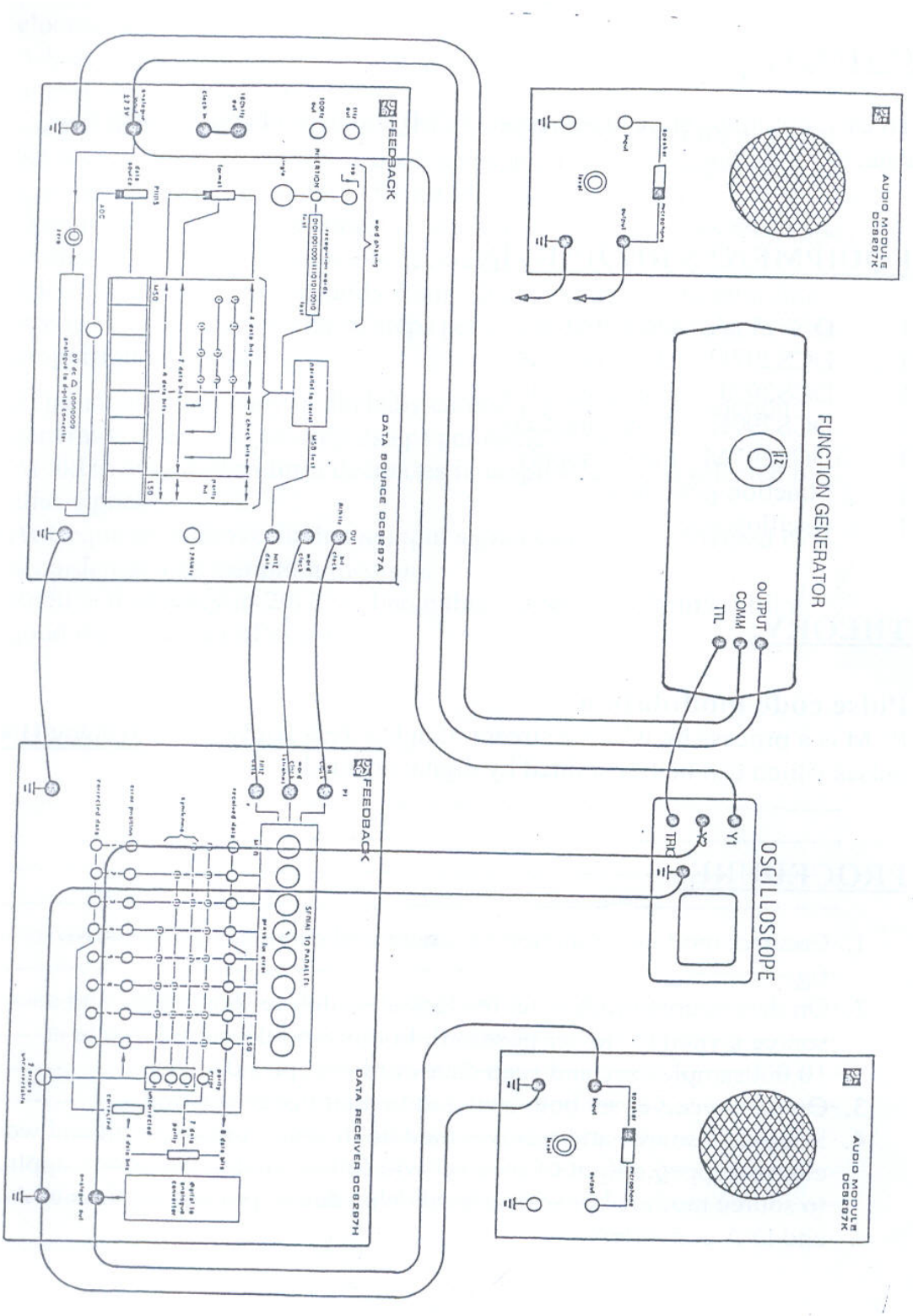

FEEDBACK
DATA SOURCE DCG827A
AUDIO MODULE DCG827K
FUNCTION GENERATOR
OUTPUT
COMM
TTL
OSCILLOSCOPE
Y1
TRIG
FEEDBACK
DATA RECEIVER DCG827H
AUDIO MODULE DCG827K

Figure 4.2

LAB SESSION 05
Pulse Width Modulation & Pulse Position Modulation

OBJECTIVE:

- To examine PWM and PPM modulator operation.
- To examine how PWM and PPM receiver work.

EQUIPMENT REQUIRED:

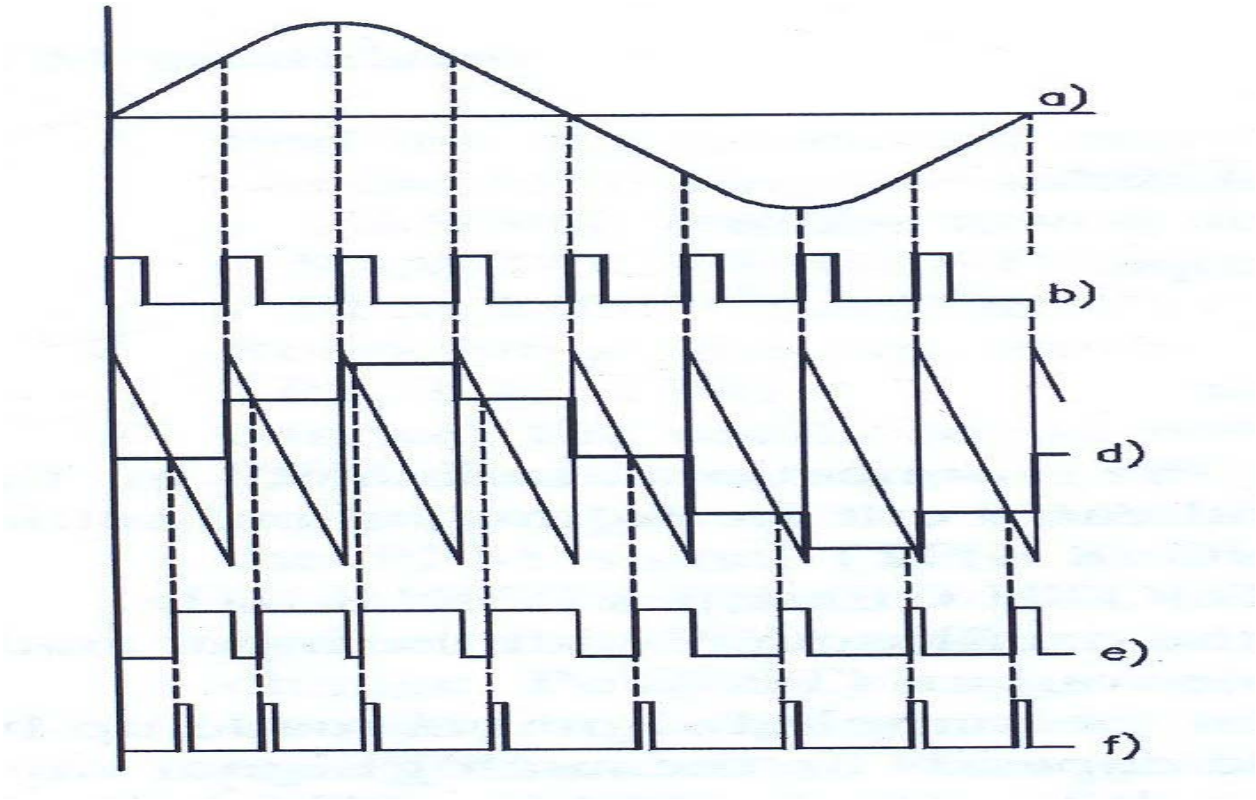
- Modul e T20A
- Power supply
- Oscill oscop e

THEORY:

A Pulse carrier can be modulated as concerns its amplitude or its timing.

This second case is usually defined as pulse Time Modulation (PTM): two instances of PTM are Pulse Width Modulation (PWM) and Pulse Position Modulation (PPM).

A ***PWM signal*** is a pulse signal whose pulse width is proportional to the modulating analog signal amplitude. The PWM signal is also used to generate the ***PPM signal***. This is a pulse signal whose pulse position is proportional to the modulating analog signal amplitude. The PPM pulses are usually generated by the descending front of the PWM pulses.



Modulator Waveforms: a) analog signal b) sampling pulses c) ramp d)PAM e) PWM f) PPM

Modulation:

PWM Modulator

The block diagram of the PWM modulator mounted on the module is shown in fig. The PMW modulator proper includes a stage comparator, which compares the respective amplitude of:

- a PAM signal obtained by sampling the input analog signal
- a sampling-pulse-synchronous ramp signal.

The comparator switch the output when the PAM signal amplitude exceeds the ramp signal amplitude: this results into a pulse signal whose pulse duration depends on the amplitude of the input analog signal. from the modulator waveforms indicated in fig. notice that the PWM pulse trailing edge corresponds to the sampling pulses, whereas the (variable) leading edge corresponds to the comparator switching.

PPM Modulator

The block diagram of the PPM modulator mounted on the module is shown fig. The PPM signal is obtained form the PWM signal, by generating fixed-duration pulse which correspond to the leading edges of the PWM signal. This result into a train of pulse whose position depends on the input analog signal.

Demodulation:

Just like the PAM signal, PPM signal can also be demodulated with a low-pass filter. In fact, the average PWM pulse width and the average position of PPM pulses are

proportional to the modulating analog signal amplitude. The low- pass filter extracts this component from the PWM/PPM signals and provides a demodulated signal which corresponds to the original modulating signal.

This (direct) demodulation method can be used both for PWM and for PPM. In the case of PPM, the demodulated signal shows a very low amplitude, for PPM pulses are vary narrow and much spaced out. A more effective PPM demodulation is performed by converting the PPM signal into a PWM one, with subsequent filtering through a low-pass filter.

PWM Receiver

The block diagram of the PWM receiver mounted on the module is shown in fig. The PWM signal coming from the transmitter is amplified and then directly applied to the low-pass filter which extracts the modulating signal.

PPM Receiver

The block diagram of the PPM receiver mounted on the module is shown in fig. the PPM signal coming from the transmitter is amplified and subsequently applied to two section: the sampling-pulse regenerator and the PPM PWM converter output is filtered through a low-pass filter which supplies the demodulated analog signal.

Sampling-pulse regeneration for the demodulator is performed as follows.
The amplified PPM signal passes the limiting circuit which reduces the signal amplitude variation. The next band-pass filter (adjusted at 8 or 12 KHz according to the sampling frequency adopted for multiplexing) separates the sampling-frequency component. This component gets to the PLL circuit which generates synchronous pulse signal with the pulses of the received PPM signal. The next circuit allows to phase -adjust the PLL-generated pulses so that, with no modulation going on, the PPM pulses are in the middle of the synchronization pulses.

The PPM/PWM converter comprises a bistable circuit (flip-flop) and work as follows.

- The synchronism pulse cases low-output-level switching, whereas the PPM pulse determines high-level switching
- Since the position of the PPM pulse varies, pulses with variable duration are obtained at the flip-flop circuit output

The PWM signal obtained through PPM conversion is filtered again by the low-pass filter which extracts the modulating signal.

PROCEDURE AND OBSERVATIONS:

Modulation:

1. Perform the connections. Supply the ±12V power and carry out the following pre-setting.
   -TIMING: 8KHz , SAWTOOTH GENRATOR: 8KHz
2. Connect the oscilloscope with the input analog signal (TP1) and with the sampler

output (TP5).

3. Verify that the sampled signal is made up by a series of steps whose amplitude depends on the analog signal waveform.
4. Move the probe from TP1 to TP , check that the SAWTOOTH GENERATOR supplies an approximate ramp of +3V : -3V for each sampling interval.
5. Move the probe form TP6 to TP8 (PWM modulator output). Synchronise the oscilloscope with the PAM signal (TP5) and verify the following:
   - The trailing edge of the pulses corresponds to the sampling pulses
   - The leading edge – and consequently the duration of the PWM pulses – varies according to the PAM signal amplitude and corresponds to the instant in which the PAM exceeds the ramp signal.
6. Vary the amplitude of the modulating analog signal and notice the corresponding variation of the PWM signal.
7. Perform the necessary connections, then carry out the following per-settings: - TIMING: 8 KHz, SAWTOOTH GENERATOR: 8 KHz, PPM MODULATOR: Pulse width completely turned clockwise
8. Re-examine the waveforms related to the PWM modulator (TP1,TP5,TP6,TP8)
9. Connect the oscilloscope with the PWM signal (TP8) and with the PPM modulator output (TP9). Synchronise the oscilloscope with the trailing edges of the PWM signal 9TP8). It is possible to verify that the PPM signal (TP9) is made up by a train of generated pulses which correspond to the leading edges of the PWM pulses. Also notice that PPM pulses have affixed duration and their position change. Position change according to the modulating analog signal can also be emphasised by examining the sampling pulses (TP4) and the PPM pulses (TP9) together.
10. Change the amplitude of the modulating analog signal and notice the corresponding variation of the PPM signal.

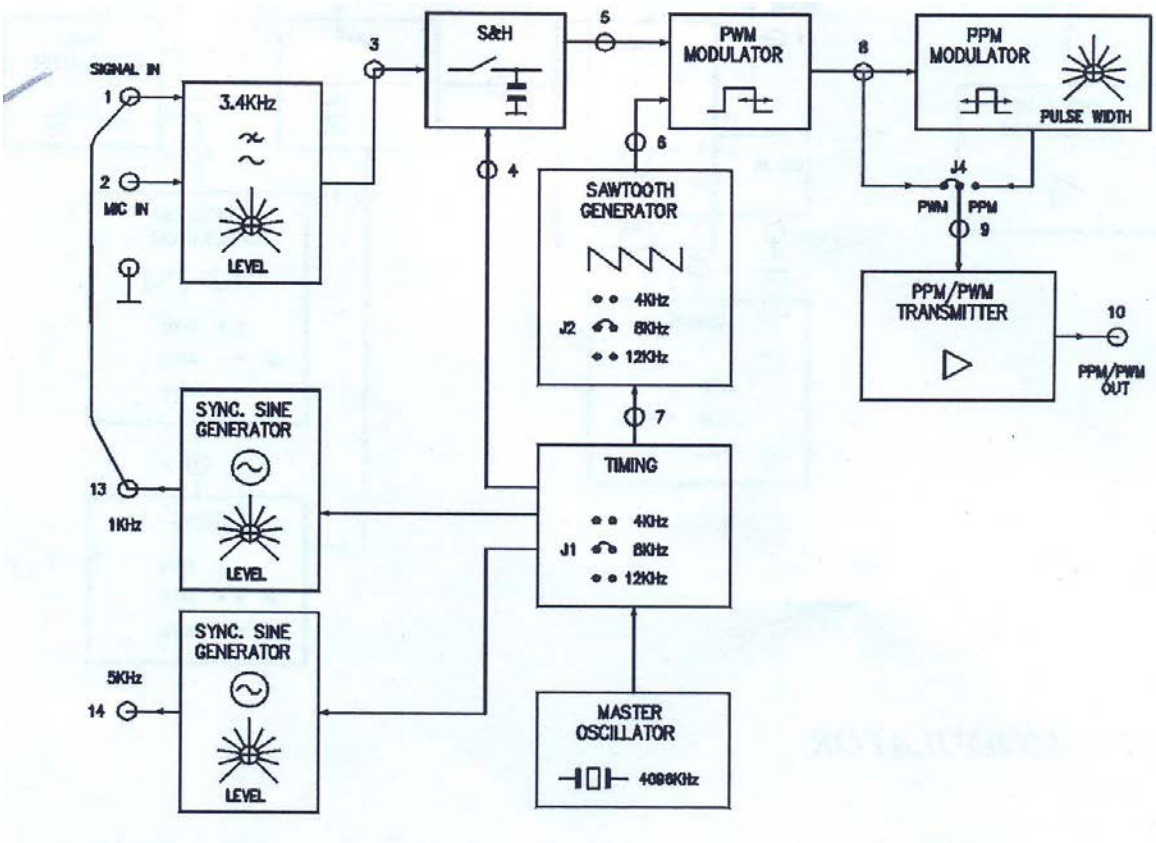


Figure 5.1: PWM Modulation

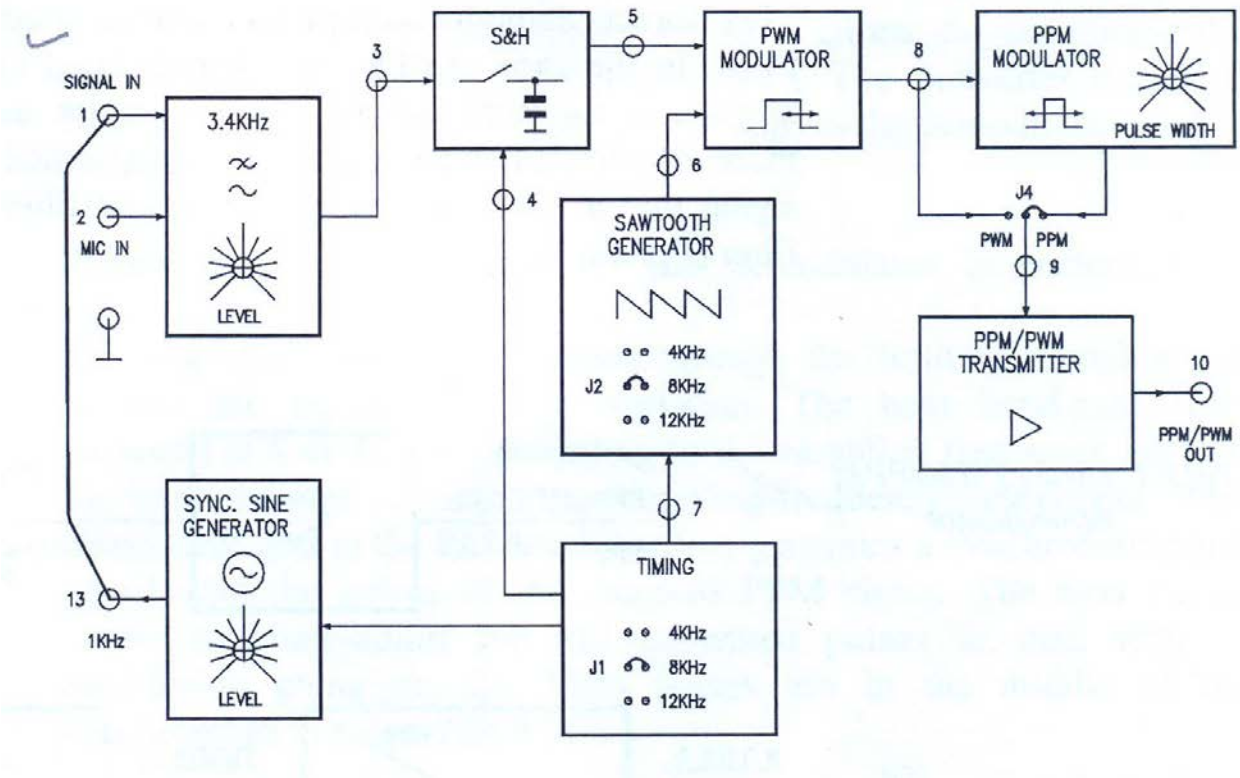


Figure 5.2: PPM Modulation

Demodulation:

PWM Demodulation

1. Generate a PWM signal. Regulate the input signal amplitude in order to obtain approximately 0.5 V_{pp} inTP3.
2. Connect the transmitter output (TP10) with the line input (TP15) and the line output (TP16) with the receiver input (TP17). Bring the line attenuation to the minimum and remove the jumper which selects the line band-pass in the receiver, put jumper J8 in the PWM position.
3. In the TP26, examine the waveform of the reconstructed signal. Verify That this one shown a slight distortion, due to inadequate suppression of the sampling frequency (8KHz) and of the different frequencies in the PWM signal.
4. Cascade-connect the 5 KHz low-pass filter with the 3.4 KHz one (connect TP26 with TP25) in order to increase the overall filter selectivity.
5. In TP27, examine the waveform of the reconstructed signal and verify that distortion almost disappears.
6. Change the line attenuation and observe how the reconstructed signal amplitude changes. Explain the reason why this happens.

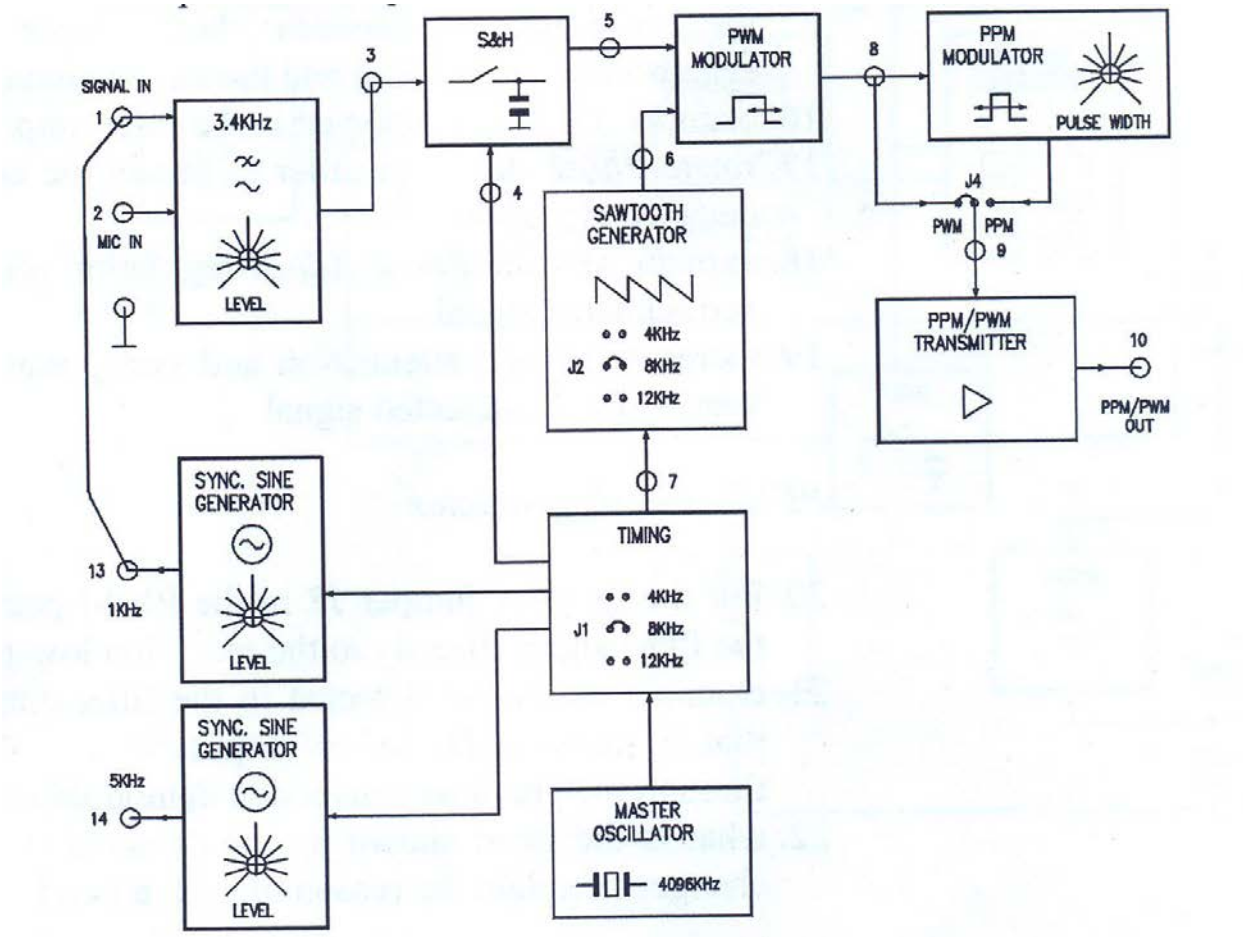


Figure 5.3: PWM Demodulation

PPM Demodulation

Synchronisation pulse regenerator

1. Generate a PPM signal.
2. Connect the transmitter output (TP10) with the line input (TP15) and the line output (TP16) with the receiver input (TP17). Bring the line attenuation to the minimum and remove the jumper which selects the line band-pass
3. Examine the waveform at the amplifier input and output (TP17) and TP18). Output pulses have a wider amplitude and are slightly distorted (having a sharper shape). Distortion is due to the slight low-pass amplifier response. The can effectively reduce the effect of line noise superimposed on the signal
4. In TP20 (filter output) an almost sinusoidal waveform is obtained, with the same frequency as the PPM pulses at the receiver input
5. In the PLL output (TP21), if the PLL is locked (a bright light appears on the LOCK led) a square waveform is obtained. This shown the same frequency as the PPM pulses in the receiver input

PPM conversion demodulator

6. Put the modulating signal to zero. Jointly examine the PPM signal in the pulse-generator output (TP23) and the synchronization pulses coming out of

the phase adjust circuit. Verify that PPM pulses can be equally spaced out compared to synchronisation pulses through the phase adjust potentiometer

7. Examine the PPM/PWM converter-related signal (TP24) and check the relationship between both input signals (PPM and synchronization signals) and the output signal (PWM)
8. Examine the signal detected in the filter output (TP26)
9. Rotate Phase Adjust in order to obtain the correct waveform of the detected signal.

PPM direct demodulator

10. Put the receiver jumper J8in the PWM position, in order to apply the PPM signal directly to the reception low-pass filter.
11. Examine the signal detected in the filter output (TP26), and verify that it shows a far lower amplitude than the amplitude obtained through the previous conversion demodulation
12. What is the trend shown by the detected signal as line attenuation changes?
    a. Explain the reason of such a trend.

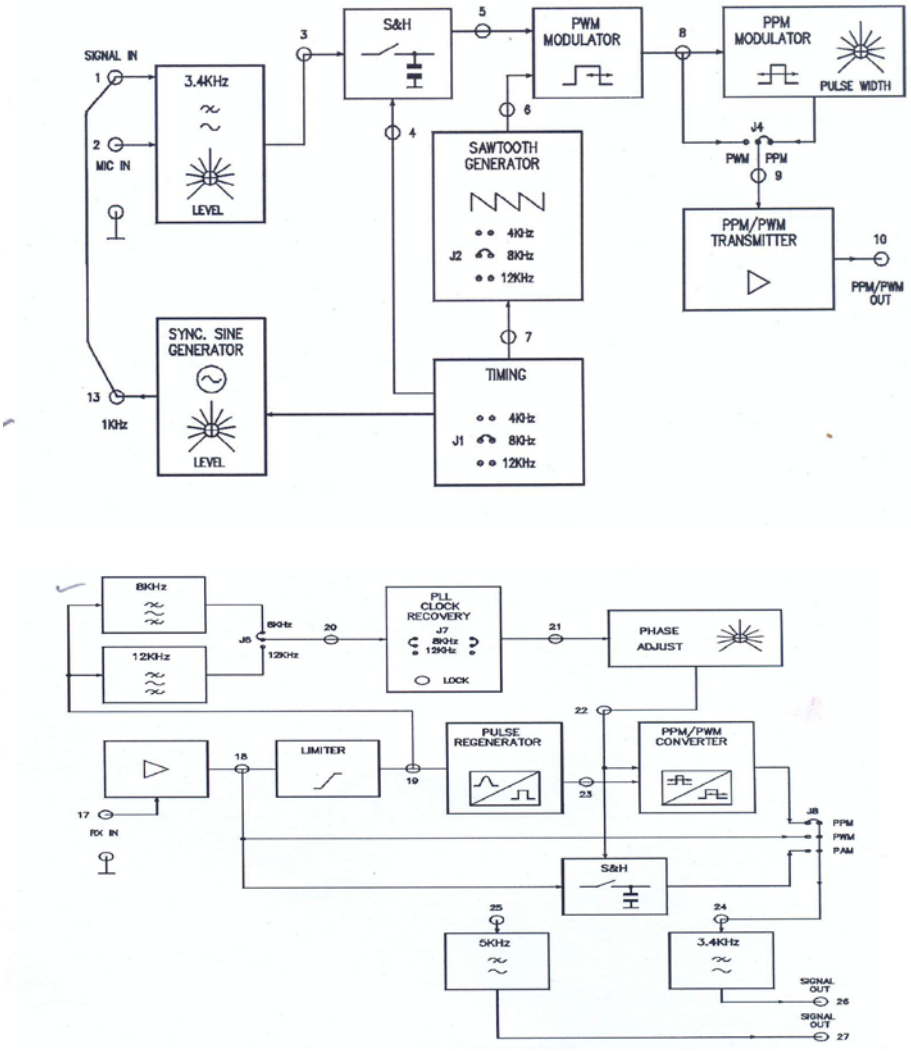


Figure 5.4: PPM Demodulation

LAB SESSION 06
Amplitude shift keying

OBJECTIVE:

- To perform Amplitude Shift Keying (ASK) modulation and demodulation.
- To examine the effect of noise and attenuation on ASK systems.

EQUIPMENT REQUIRED:

- Power unit PSU
- Module holder base
- Experiment module MCM31
- Oscilloscope.

THEORY:

Amplitude shift keying –ASK

In a digital communications system the modulating waveform will be in the form of a square wave, or pulse train. ASK is a form of amplitude modulation where the carrier is modulated by the pulse train.

In this form of modulation the sine carrier takes 2 amplitude values, determined by the binary data signal. In its simplest form the carrier is 'keyed', that is switched on and off for set periods to conform with the bit pattern of the modulating signal. Usually the modulator transmits the carrier when the data bit is “1”. It completely removes when the bit is”0”. This is known as on-off ASK, or on-off keying (OOK). There are also ASK shapes called multi-level where the amplitude of the modulated signal takes more than 2 values.

If the modulating waveform is not a sinusoid but is a square wave, or a pulse train, using on-off keying the equation becomes: $v(t) = A\cos \omega ct$ during the on period and $= 0$ elsewhere.

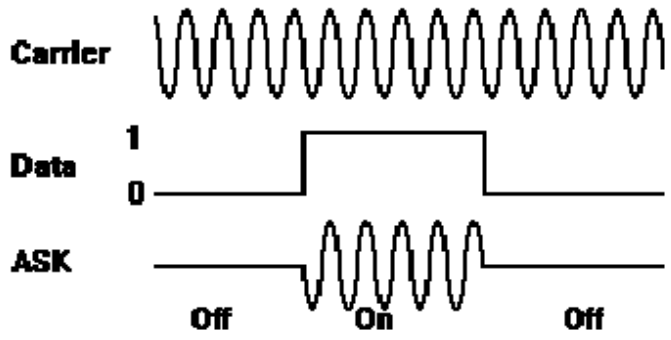


Figure 6.1: Message signal (above) and corresponding ASK signal (below)

The demodulation can be coherent or non coherent. In the first case, more complex as

concern the circuit but more effective as against the noise effect, a product demodulator multiplies the ASK signal by the locally generated carrier. In the second case the envelope of the ASK signal is detected via diode. In both cases the detector is followed by a low pass filter which removes the residual carrier component and a threshold circuit which squares the data signal.

Bit Error rate- B.E.R

The B.E.R is the ratio of the error bits to the total received bits. Practically it tells the user how accurate the received data is.

BER = (No. of error Bits) / (Total No of received bits)

ASK Modulator

The block diagram of ASK modulator is shown in figure 1. The sine carrier (1200 or 1800Hz) is applied to an input of the balanced modulator 1; a data signal is connected to the other circuit which multiplies the two signals connected at the input.

ASK Demodulator

The ASK demodulator consist of the following sections:

- Full wave envelope detector
- A low pass filter
- A threshold circuit (in case of asynchronous data which are not retimed)
- A clock extraction and data re-timing circuit, in case of synchronous data.

PROCEDURE:

Modulation

- Power on module.
- Set the circuit in ASK mode, with 24-data bit source and without data coding (connect J1c-J3d-J4-J5-J6a ; set SW2=normal, SW3=24 bit, SW4=1200 ,SW6=ASK, SW8=BIT and ATT=min, NOISE=min
- Set an alternate data sequence 00/11 and push START
- Connect the oscilloscope to TP6 and TP16 so to display the data signal and ASK signal wave form. See Figure 5.3.
- Adjust the phase of the carrier to make the zero of sine wave correspond to the starting of the bit intervals.

De-modulation

- Keep the last condition (J1c-J3d-J4-J5-J6a; SW2=normal, SW3=24 bit SW4=1200, SW6=ASK, SW8=BIT and ATT=min, NOISE=min
- Set an alternate data sequence 00/11 and push START
- Connect the oscilloscope to TP16 and TP20 to examine the ASK signal before and after the communication channel. Note the readings at TP23, TP24, TP29

- Note the effect of the communication channel on the ASK signal.

Bit Error Rate

- Set the jumpers as follows: J1d-J3d-J4-J5-J6a.
- Set Switches as per the following SW2=Normal, SW3=64 bit, SW4=1200Hz, SW6=ASK, SW8=BIT, **SW9=STOP.**
- Set NOISE at 50 % of maximum value. Set SW9=READ and Push RESET (to initialize counter to zero). Let the counter progress for 60 seconds after which set SW9=STOP and note counter reading.
- Repeat steps and note error reading for NOISE at 100 %.
- The received bits are 18000 per minute. (300 bits/s times 60 seconds)

OBSERVATIONS:

- Draw the block diagram of DCS and draw the output of each block.

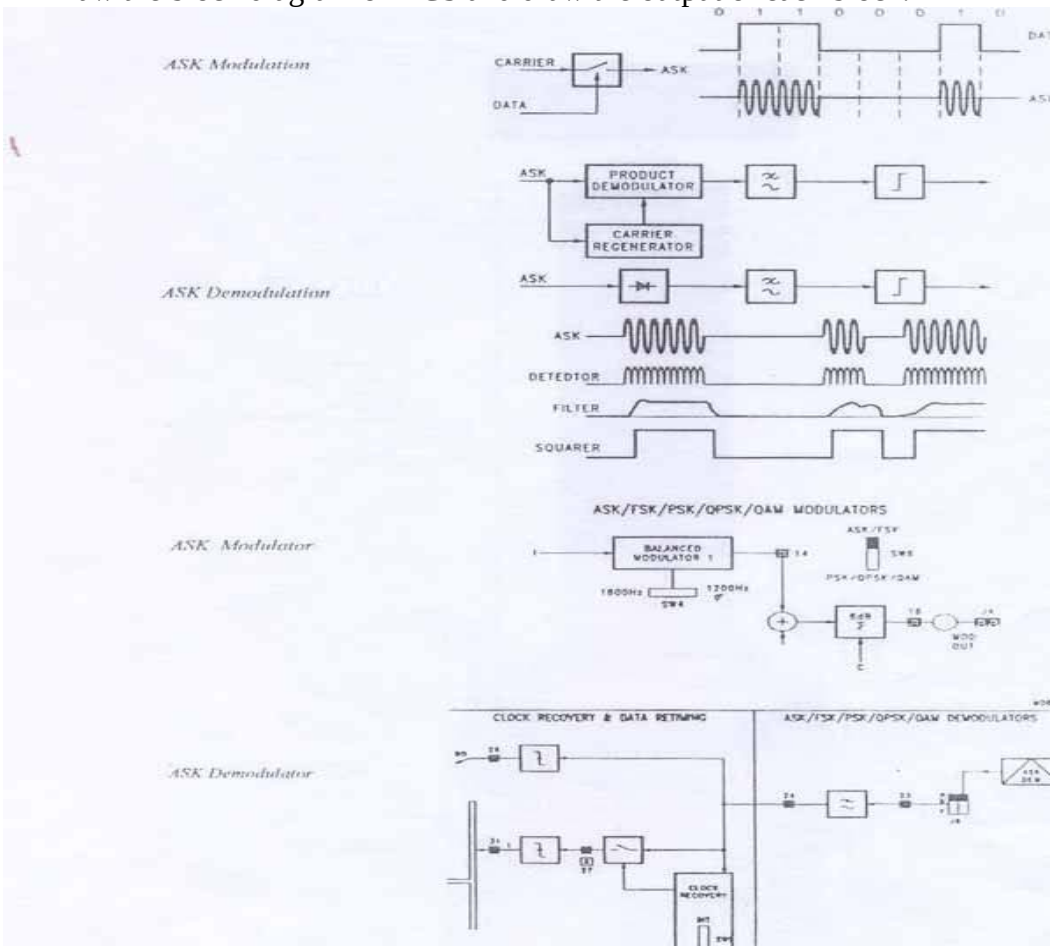


Figure 6. 2

BLOCK DIAGRAM:

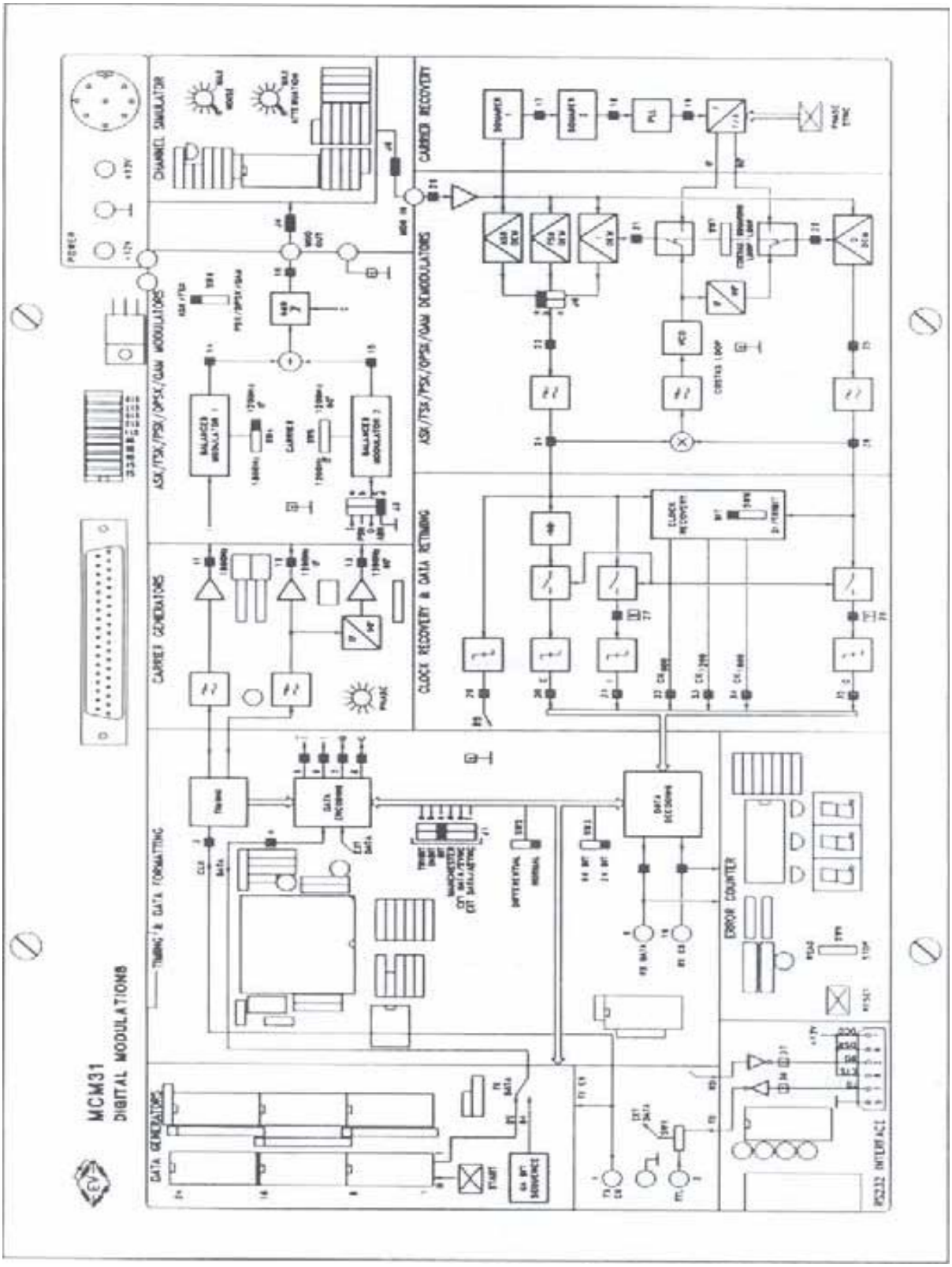


Figure 6.3

RESULTS & CONCLUSION:

- Effect of Attenuation

- Effect of Noise

LAB SESSION 07
Binary Frequency Shift Keying

OBJECTIVE:

- To observe Frequency Shift Keying (FSK) modulation and demodulation.
- To examine the effect of noise and attenuation on FSK systems.

EQUIPMENT REQUIRED:

- Power unit PSU
- Module holder base
- Experiment module MCM3 1
- Oscilloscope

THEORY:

Frequency shift keying –FSK

In this modulation the sine carrier takes 2 frequency values, determined by the binary data signal. The modulator can be carried out in different ways:

- A voltage controlled oscillator (VCO): a single voltage-controlled oscillator may be used with its frequency altered by the modulating signal voltage.
- A system transmitting one of the 2 frequencies as function of the data signal: The two frequencies may be produced by two oscillators and their outputs switched by the modulating signal.
- A frequency divider controlled by the data signal.

The most used demodulation techniques are the one using a PLL circuit. The FSK signal across the PLL input takes two frequency values. The error voltages supplied by the phase comparator follows such variations, and so, it constitutes the NRZ binary representation (high and low level) of the FSK input signal. The PLL demodulator is followed by a low pass filter, which removes the residual carrier components and a squarer circuit which forms the proper data signal.

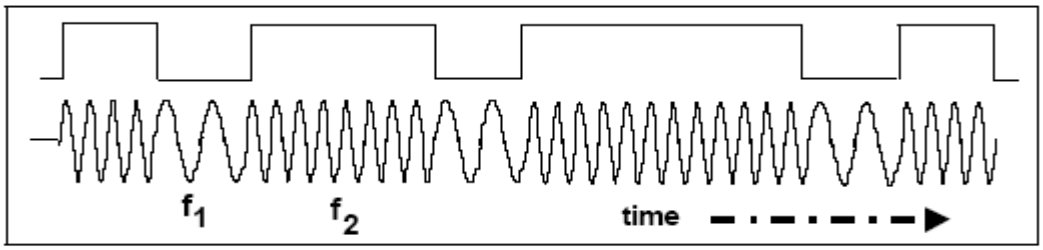


Figure 7.1: Frequency Shift Keying

Bit Error rate- B.E.R

The B.E.R is the ratio of the error bits to the total received bits. Practically it tells the user how accurate the received data is.

BER = (No. of error Bits) / (Total No of received bits)

FSK Modulator

The FSK Signal is generated by means of 2 ASK modulator, whose outputs are combined together with an adder. The two sine carriers 1200 and 1800 Hz are applied separately to the 2 modulators. The data reach one of the two modulators in direct form, the other in negated form. In this way a modulator supply a sin wave when the datum is “1”, the second when the datum is “0”. The FSK signal is obtained by adding the two outputs.

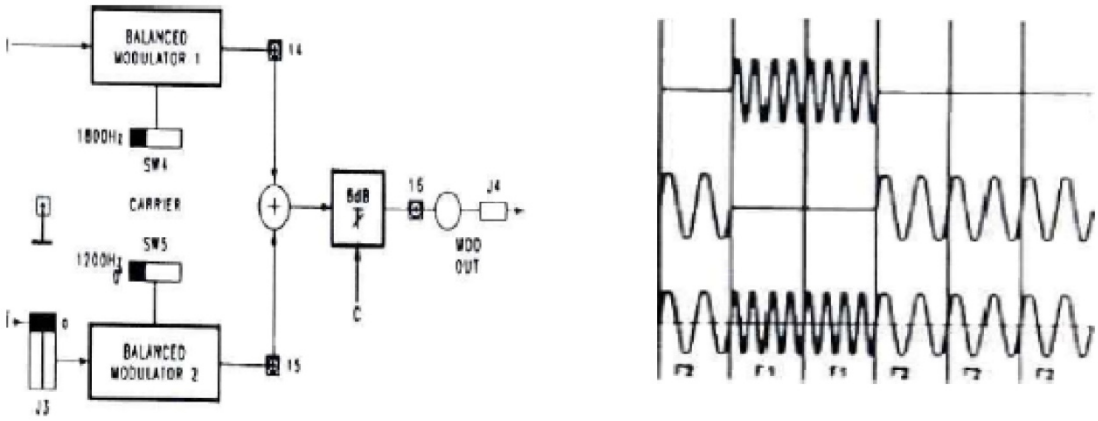


Figure 7.2: FSK modulator

FSK Demodulator

The FSK demodulator consists of an FM detector made with PLL circuit, A low of filter, A squarer circuit (with output at TP29). In case of asynchronous data, which are not retimed), A circuit for clock extraction and data re-timing in case of synchronous data (data output on TP31, clock on TP32).

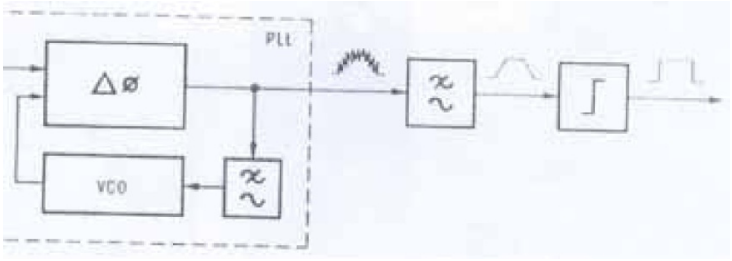


Figure 7.3: FSK Demodulation with PLL

PROCEDURE:

Modulation

- Power the module
- Set the circuit in FSK mode, with 24-bit data source and without data coding (connect J1c-J3a-J5-J6b; set SW2=normal, SW3=24bit, SW4=1 800, SW5=1200/0^{o}, SW6=FSK, SW8=BIT , ATT=min, NOISE=min)
- Set an alternate data sequence 00/11 and push START
- Connect the oscilloscope to TP6, TP 14, TP 15, TP16 and examine the data signal and FSK signal, adjust the phase (PHASE) of the 1200-Hz carrier to get continuity of FSK signal in the passage between the two frequencies (this kind of modulation is known as minimum frequency shift

keying)

Demodulation

- Keep the last condition (J1c –J3a-J4-J5-J6b; SW2=Normal ,SW3= 24bit, SW4=1 800, SW=5=1200/0^0 , SW6=FSK, SW8=BIT , ATT=Min, NOISE =Min
- Set a alternated data sequence 00/11 and push START
- Connect the oscilloscope to TP16 and TP20, to examine the FSK signal before and after the communication channel. Connect oscilloscope to TP23, TP24 and TP29. Note down observations.
- Increase noise & note result then increase attenuation and note result.

Bit Error Rate

- Set the jumpers as follows: J1d-J3d-J4-J5-J6a.
- Set Switches as per the following SW2=Normal, SW3=64 bit, SW4=1200Hz, SW6=ASK, SW8=BIT, SW9=STOP**.**
- Set NOISE at 50 % of maximum value. Set SW9=READ and Push RESET (to initialize counter to zero). Let the counter progress for 60 seconds after which set SW9=STOP and note counter reading.
- Repeat steps and note error reading for NOISE at 100 %.
- The received bits are 18000 per minute. (300 bits/s times 60 seconds).

OBSERVATION:

- Draw the block diagram of DCS and draw the output of each block.

RESULTS & CONCLUSION:

- Effect of Attenuation

- Effect of Noise

TASK:

1. You are required to simulate a digital communication system, the bit steam is to be BFSK modulated, and channel impairments are to be added to the modulated signal, non-coherent detection is to be performed and verified whether the decision is right or wrong.

LAB SESSION 08
Binary Phase Shift Keying

OBJECTIVE:

- To observe the 2 PSK (Phase Shift Keying) modulation and demodulation.

EQUIPMENT REQUIRED:

- Power unit PSU or PS1
- Module holder base
- Experiment module MCM 31
- Oscilloscope

THEORY:

Phase shift keying -PSK

In this kind of modulation the sine carrier takes 2 or more phase values, directly determined by the binary data signal (2-phase modulation) or by the combination of certain number of bits of the same data signal (n-phase modulation) 2 phase PSK modulation is also called 2 PSK or binary PSK (BPSK) or phase reversal keying (PRK). The sine carrier takes 2 Phase valued determined by the binary data signal, modulation techniques is the one using a balanced modulator is the direct or inverted input carrier as function of the data signal.

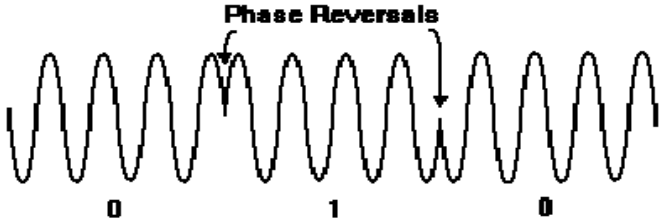


Figure 8.1 A PSK waveform

2PSK Modulator

The sine carrier (1200 Hz) is applied to an input of the balanced modulator 1; a data signal (indicated with I) is applied to the other input. The circuit operates as balanced modulator, and multiplies the two signal applied to the inputs.
Across the output, the sine carrier is direct when the data signal is to low level (bit"0"), inverted (shift 180°) when the bit is "1" The 2-PSK signal then enters the adder used for FSK/QPSK/QAM modulation and exits via separator stage.

2PSK Demodulator:

The demodulation is carried out via a product demodulator which is reached by the PSK signal and a locally regenerated carrier. This must have the same frequency and phase of the one used in transmission (it must be coherent with the received signal), and is taken from the PSK signal.

PROCEDURE:

Modulation

- Power on module
- Set the circuit in PSK mode, to get wave form of the PSK modulator and demodulator, with 24-bit data source and without data coding (connect J1c-J3b-J4-J5-J6c; set SW2=normal, SW3=24_bit, SW4= 1800, SW5=1200/0^{o}, SW6=PSK, SW8=BIT, ATT=min, NOISE=min)
- Set an alternate data sequence 00/11 and push START
- Connect the oscilloscope to TP6 and TP16, and examine the data signal and PSK signal.
- Adjust the phase (PHASE) to invert the phase of carrier in correspondence to 0.

Demodulation

- Connect the oscilloscope to TP24 and TP29. To examine the PSK signal before and after the communication channel.
- Observe the effect of the communication channel one PSK signal. As the communication channel is limited band, the phase transition of the output PSK signal is slightly beveled.

OBSERVATION:

- Draw the block diagram of DCS and draw the output of each block.

RESULT & CONCLUSION:

- Effect of Attenuation

- Effect of Noise

TASK:

1. Write MATLAB programs and display signal constellations for BPSK, QPSK, 8 PSK and 16PSK.

LAB SESSION 09
The Costas Loop Demodulator

OBJECTIVE:

- To observe the Costas Loop for coherent detection at the receiver.

EQUIPMENT REQUIRED:

Modulation and keying Workboard 53-160 which comprises the following blocks:

- Data Generator
- ASK, FSK, PSK, QPSK, DPSK, DQPSK Link

THEORY:

Demodulation of PSK using a Costas Loop

The Costas Loop provides an alternative way of demodulating PSK transmissions. It uses a phase locked loop to produce a carrier frequency reference of constant phase which is then multiplied by the incoming PSK signal to produce a demodulated data output.

The block diagram below shows the three multipliers (modulators) and the VCO which form the Costas Loop.

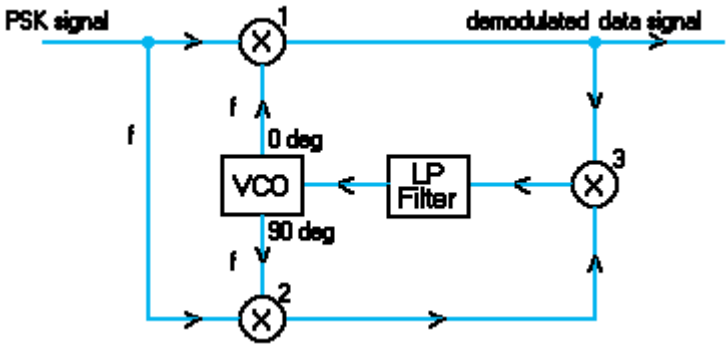


Figure 9.1 Costas Loop

If the VCO is locked to the incoming carrier then:

$$\omega_{vco} = \omega_c$$

only a small phase difference, $ø_e$ will be present.
Let the two outputs from the VCO be: $2\cos \omega_c t$ in phase with the carrier $2\sin \omega_c t$ in quadrature.
The PSK signal input is:

$$s(t) = A \cos [\omega_c t + ø]$$

where ø is 0 or π depending on whether the state of the digital input d is 1, or -1.

So, if d(t) is the state of the digital input, this signal expression can be written:

$$s(t) = A\, d(t) \cos \omega_c t$$

The multiplier outputs are the products of the two inputs to each. Thus these outputs are:

$$[A\ d(t) \cos \omega_c t][2\cos \omega_c t] \text{ and}$$
$$[A\ d(t) \cos \omega_c t][2\sin \omega_c t] .$$

The reference channel output is used; i.e.,

$$v_{out} = [A\ d(t) \cos \omega_c t][2\cos \omega_c t]$$
$$= 2A\ d(t) \cos 2\ \omega_c t$$

Now, $\cos 2x = 0.5[1 + \cos 2x]$, so the expression for v_{out} becomes:

$$v_{out} = 2A\ d(t)\ [0.5 + 0.5\cos 2\omega_c t]$$
$$= A\ d(t) + A\ d(t) \cos 2\omega_c t$$

This expression has two components: a dc component dependant on the phase of the digital input data and a component at twice the carrier frequency.

This double-frequency component can be removed by a post detection filter.

When the loop is in lock in Figure 8.1, the VCO will be phase-locked by modulators (2) and (3), causing it to produce an output from its f90 terminal that leads the incoming signal by 90 degrees. Since the VCO produces outputs which differ by 90 degrees, the reference signal from the f0 output will be in phase with the incoming PSK signal for, say, binary 1 and 180 degrees out of phase for binary 0.

The multiplying action of modulator (1) will then produce a positive dc level when the received and reference signals are in phase and a negative level when they are in anti-phase.
Subsequent data recovery circuits convert the bipolar output from the Costas Loop demodulator to unipolar NRZ data.
It should be noted that when the incoming signal changes state, the sign of both inputs to modulator (3) change simultaneously so that its output remains constant and the VCO will be locked in at constant phase. If there is no information which specifies which of its phase values is +90 degrees and which is -90 degrees with respect to the reference, the demodulated data could be inverted (binary 1 and binary 0 interchanged).

This is a difficulty that cannot be resolved unless we know more about the signal than just that it is +/-90 degrees PSK. This ambiguity can be resolved by extending the use of a sync word pattern which is transmitted at regular intervals.

The pattern is chosen, not only to provide synchronisation information, but to provide a bit sequence that can be identified as either correct, or inverted. When an inverted bit pattern is detected the logic circuits in the receiver re-invert the data to restore the correct sense.

PROCEDURE:

- Connect the modulation and keying work board.
- Power the module. The carrier is being phase modulated by the data waveform. The phase shift of the PSK is +/-90 degrees, but can also be varied
- Set all the potentiometer controls to their mid position.
- Set the **MS bits switch (7)** to A and the **LS bits switch (8)** to A.
- Observe the signals around the circuit using the oscilloscope.

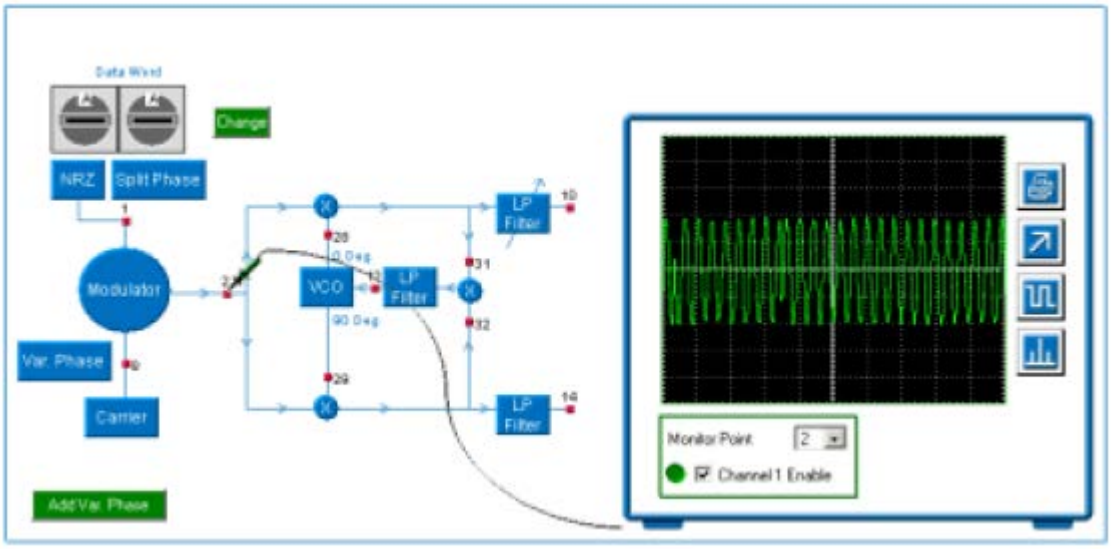


Figure 9.2

OBSERVATIONS & RESULTS:

1. Do the waveforms at monitor point 1 correspond with the formats selected?
2. Do the waveforms at monitor point 2 show PSK?
3. Select NRZ format and ensure that Var. Phase is not selected. Is the required demodulated output present at monitor point 10?
4. Look at monitor point 31.
   a. Is the required demodulated output present at this monitor point?
   b. What needs to be done to provide the required output?
5. Go to monitor point 10 again and vary the PDF control (9). What effect does this have on the demodulated output?
6. Look at the VCO control voltage at monitor point 12. How does this vary in response to the incoming PSK? (You can decrease the selectivity of the LP Filter by turning the PLL filter control (6) towards minimum to see the effect better).
7. Set both the MS bits switch (7) and the LS bits switch (8) to 0. Does the demodulator output correspond to the data input?
8. Set both of the data switches to 1.
   a. Does the output correspond now?
   b. What can you say about the Costas Loop demodulator compared with the PLL demodulator as regards to NRZ formatted data demodulation?
9. Repeat above procedure with Split Phase format?

LAB SESSION 10
Quadrature Phase Shift Keying

OBJECTIVE:

- To observe QPSK modulation & demodulation.

EQUIPMENT REQUIRED:

Modulation and keying Workboard 53-160 which comprises the following blocks:

- Data Generator
- ASK, FSK, PSK, QPSK, DPSK, DQPSK Link

THEORY:

Modulation of QPSK

Quadrature Phase Shift Keying (QPSK) in an extension of the simple PSK method of keying investigated in the Phase Shift Keying Assignment.

In QPSK the signal can take up one of four possible phase angles, mutually in quadrature, each corresponding to a particular data input condition.

Consider NRZ formatted data in which each word is divided into bit pairs instead of individual bits.

There are four possible ways of pairing binary 1 and 0. These are:

00
01
10
11

Any data word with an even number of bits may be represented by a combination of these bit pairs. One of the four phase angles is assigned to each of these bit pairs.

QPSK offers twice as many data bits per carrier phase change than Binary Phase Shift Keying (BPSK), and hence finds wide application in high-speed carrier-modulated data transmission systems.

For example, if the data transfer rate is 9600 bits per second the transmission line signaling rate will have 4800 bit pairs per second and thus will be at 4800 baud.

Typically, the four phases chosen foe QPSK are +/-45 degrees and +/-135 degrees. Each of these is assigned a bit pair (dibit).

The diagram shows a possible dibit pattern, often referred to as a constellation because of its star shape.

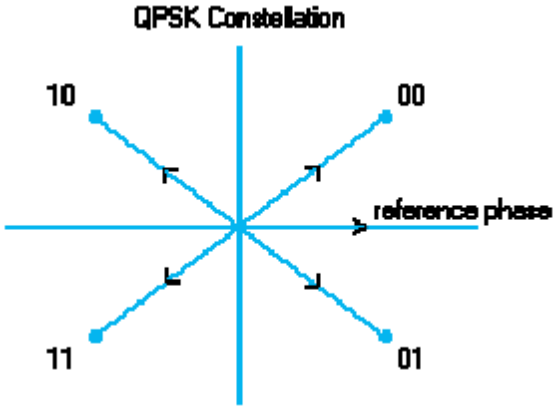


Figure 10.1 QPSK constellation Diagram

In practice, the generation of the bit pairs may be done in a number of different ways. The simplest method is to store two bits, read off the combination and generate the required carrier phase shift and then store the next two bits, etc.
A block diagram of such a system is shown below:

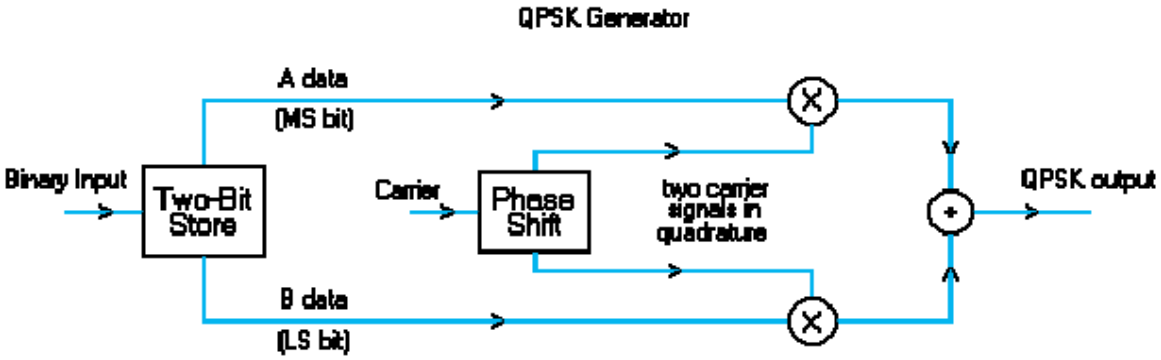


Figure 10.2

The expression for this QPSK will then be:

$$s(t) = \cos(\omega ct + \varphi)$$

Where:
$\varphi = + \pi/4, +3\pi/4, -3\pi/4.$

By trigonometric expansion, this can be written:

$$s(t) = a \cos \omega ct + b \sin \omega ct$$

Where a and b are given values corresponding to the four possible angles.

For the angles chosen, these will be:
$(2a1/2, 2b1/2) = (1, 1), (-1, 1), (-1, -1), (1, -1)$

The transmitted signal is therefore the sum of two waveforms in quadrature.

Demodulation of QPSK

The received QPSK signal must be demodulated to produce the two components of the single transmitted. This is to combat the problems of phase ambiguity (these have already been met in the assignment for PSK).

A phase detector is a mixer, whose action is one of multiplication. In a phase detector the two signals to be multiplied have the same frequency, thus the output of the detector will contain sum and difference frequency components. The sum component will be at double the carrier frequency and the difference component will be at dc.

That is, if the inputs to the detector are cos (ωct + φ) and cos ωct the output will be:

$$\cos(\omega ct + \varphi) \times \cos \omega ct = 0.5 \cos[(\omega ct + \varphi) + \omega ct] + 0.5 \cos[(\omega ct + \varphi) - \omega ct]$$
$$= 0.5 \cos 2(\omega ct + \varphi) + 0.5 \cos \varphi$$

Where φ is the transmitted data phase.

A low-pass filter is used to attenuate the second harmonic (double frequency) term, leaving:

$$Vo = 0.5 \cos \varphi$$

φ is the modulated phase shift (+/-π/4, +/-3 π/4) and Vo is the output dc voltage representing the appropriate dibit.

Now 0.5 cos (+/-45 degrees) = +0.35, and therefore the detector does not know if +45 degrees, or -45 degrees was sent.

Similarly, 0.5 cos (+/-135 degrees) =-0.35, and the same ambiguity exists.

In order to resolve these ambiguities, a second detector operating in quadrature is required. This can be achieved from the double Costas Loop type circuit, as shown below

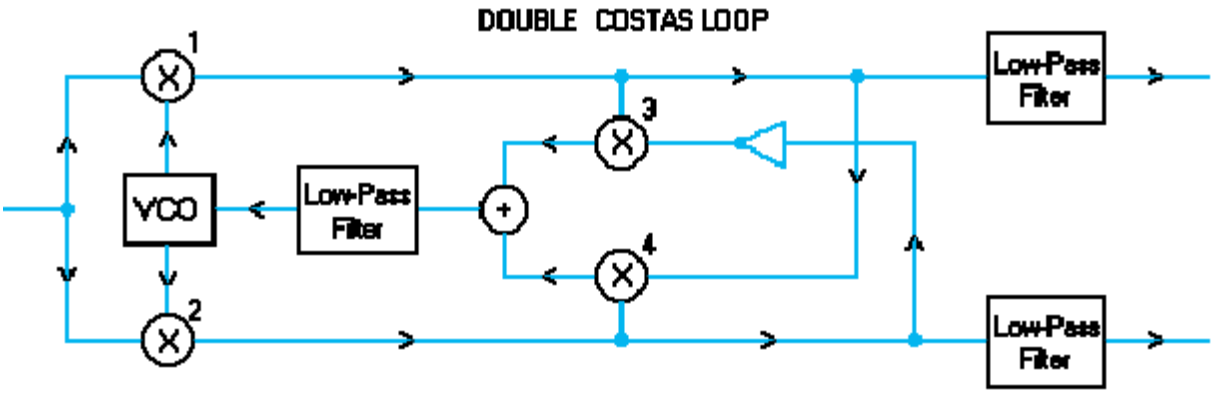


Figure 10.3

The inputs to the second detector are cos (ωct + φ) and sin ωct and the output will be:

$$\cos(\omega ct + \varphi) \times \sin \omega ct = 0.5 \sin[(\omega ct + \varphi) + \omega ct] + 0.5 \sin[(\omega ct + \varphi) - \omega ct]$$
$$= 0.5 \sin(2\omega ct + \varphi) - 0.5 \sin \varphi$$

Where φ is the transmitted data phase.

Again, a low-pass filter iss used to attenuate the second harmonic (double frequency) term, leaving:

$$Vo = 0.5 \sin \varphi$$

φ is the modulated phase shift (+/- π/4, +/– 3 π/4) and Vo is the output DC voltage representing the appropriate dibit.

The outputs of the two modulators are multiplied together to produce the VCO control signal, giving:
0.5[cos (2ωct + φ) + cos φ] x 0.5 [sin (ωct + . φ) – sin φ]
= 0.25 [cos (2ωct + φ)] [sin (2ωct + φ)] – [cos (2ωct + φ)] [sin φ] + [sin (2ωct + φ)] [cos φ]-[cos φ] [sin φ]

Expanding these terms, using the identity:
cos A. sin B= 0.5 [sin (A + B) – sin (A – B)]
= 0.25[0.5 sin 2(2ωct + φ) – 0.5 [sin (2ωct + φ) – sin 2ωct] + 0.5 [sin (2ωct + 2φ) – sin 2ωct] -0.5 sin 2φ]

Which simplifies to:
= 0.125 [0.5 sin 2(2ωct + φ) – 0.5 sin 2φ
= 0.125 [sin (4ωct + φ) – sin 2 φ]

This contains a 4ωc term, which will be filtered out, leaving a dc term proportional to the phase shift which is used to control the VCO.

PROCEDURE:

Modulation

- Connect the modulation and keying work board.
- Power the module.
- Set all the potentiometer controls to their mid position.
- Set the **MS bits switch (7)** and the **LS bits switch (8)** to 0.
- Monitor the signal before modulation at monitor point 2.See Figure 6.4

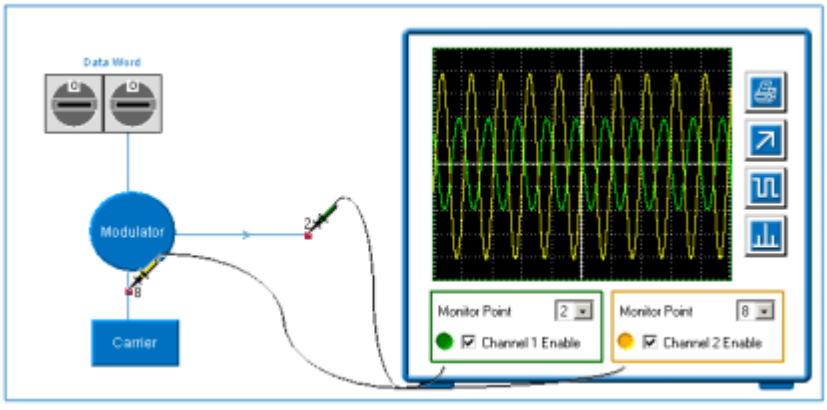


Figure 10.4

OBSERVATIONS & RESULTS:

- What are the four di-bits?

- Look at the monitor point 2 with the oscilloscope, The yellow trace is the carrier and the green trace is the modulated output.
  1. Are the two waveforms in phase?
  2. What is the phase shift of the output relative to the carrier?

- Repeat the above set of observations by:
  Setting the **MS bits switch (7)** and the **LS bits switch (8)** to F
  Setting the **MS bits switch (7)** to 0 and the **LS bits switch (8)** to F
  Setting the **MS bits switch (7)** to F and the **LS bits switch (8)** to 0

  1. Do the four possible combinations of dibits give the four phase shifts?
  2. Can you see the phase shifts in the output waveform?

Demodulation

- Set all the potentiometer controls to their mid position.
- Set the **MS bits switch (7)** and the **LS bits switch (8)** to 0.
- Observe the signal after demodulation at monitor points 10 & 14 using the oscilloscope. See Figure 6.5

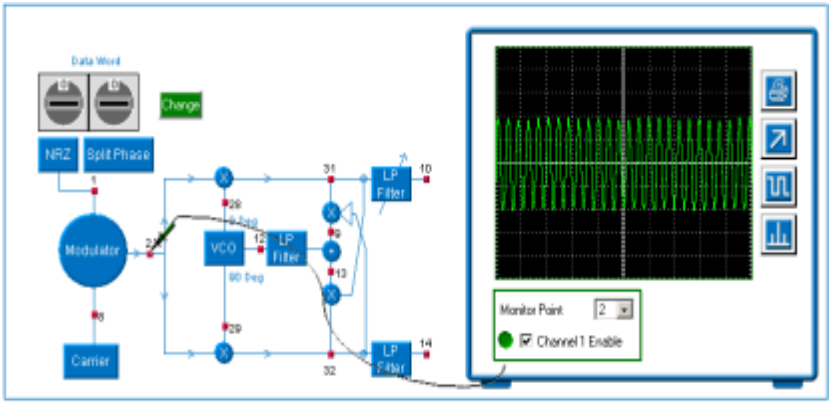


Figure 10.5

OBSERVATIONS & RESULTS:

- Set the **MS bits switch (7)** to 0 and the **LS bits switch (8)** to 5.
  1. What dibit pattern does this give?
  2. What QPSK output pattern do these settings give?

- Look at monitor point ***2*** with the large oscilloscope display. Can you see the 180 degree phase changes between the dibits?

- Look at the two outputs of the double Costas Loop (monitor pints ***10*** and ***14***).
  1. Are the waveforms the same?
  2. Ignoring the small variations present on one of the outputs, do they correspond to the originating two data bit patterns?

- Monitor the output that corresponds to the LS bits pattern. Turn the Carrier level control (5) to minimum and back to maximum a number of times and observe the output.

1. Does the output always come up the same?
2. Does it always correspond to the LS bits pattern?

- Turn the ***Carrier level control (5)*** up and down until the output is in a state that does not correspond with the ***LS bits*** pattern. Look at the other output.
    1. Does this now correspond with the LS bits pattern?
    2. Is there ambiguity as to which output corresponds to which bit in each dibit? How can this be overcome in practice?

TASK:

1. Build the QPSK Modulator and Demodulator using Simulink.

LAB SESSION 11
Differential Phase Shift Keying

OBJECTIVE:

- To observe DPSK Modulation and Demodulation.

EQUIPMENTS REQUIRED:

Modulation and Keying Workboard 53-160 which comprises the following blocks:

- Data Generator
- ASK, FSK, PSK, QPSK, DPSK, DQPSK Links

THEORY:

Modulation of DPSK

With simple PSK there is a reference phase about which the phase of the transmitted wave shifts as it is modulated.

With this type of system, both the transmitter and the receiver have to maintain an absolute phase reference against which the received signal is compared.

With Differential Phase Shift Keying (DPSK) the data is transmitted in the form of discrete phase shifts, where the phase reference is the phase of the previously transmitted signal phase.

The advantage of this technique is that an absolute phase reference does not have to be maintained.

A simple way to produce DPSK is shown in the diagram below:

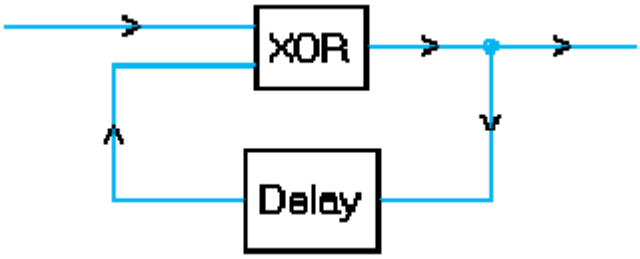


Figure 11.1

Consider the data word 10010110 and assume that the initial state of the delayed input to the exclusive-OR gate is 0. The data word can be written:

1 0 0 1 0 1 1 0
Do D1 D2 D3 D4 D5 D6 D7

Taking the data word bit by bit gives:

Do Excl-OR 0 (the initial value of the delayed input) = 1
D1 Excl-OR 1 (the resulting value from above) = 1

D2 Excl-OR 1 = 1
D3 Excl-OR 1 = 0
D4 Excl-OR 0 = 0 .....etc
Giving a resulting output word from the Excl-OR gate of:
1 1 1 0 0 1 0 0

This word is now used to modulate a binary phase shift modulator.

Let the binary 0 state correspond to an output phase of 0 radians (0 degrees) and a binary 1 to a phase of π radians,(180 degrees) giving:
π π π 0 0 0 π 0 0
as the DSPK output phase.

Demodulation of DPSK

The diagram below shows the form of a typical DPSK demodulator:

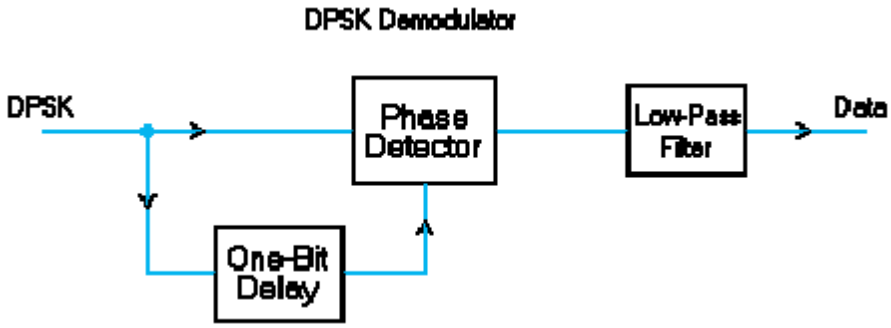


Figure 11.2

The inputs to the phase detector are the received signal and that signal delayed by one bit period.

The phase detector produces a negative output voltage when the input phases are the same and a positive voltage when they are in antiphase.

Thus, for the example of DPSK generation shown before, the inputs to the phase detector will be:

Direct input: π π π 0 0 π 0 0

Delayed input: 0 π π π 0 0 π 0
giving an input of:
- + + - + - - +
which, when put through a comparator, gives:
1 0 0 1 0 1 1 0
which is the original data word.

PROCEDURE:

- Connect the modulation and keying work board.
- Power the module.

Modulation

- The data is applied to an Exclusive-OR gate and delay circuit to differentially pre-code it
- before being applied to a phase modulator.
- Set all the potentiometer controls to their mid positions.
- Set the **MS bits switch (7)** and the **LS bits switch (8)** to 0.
- Monitor the signal before and after modulation at monitor points 17, 21 & 2 .See Figure

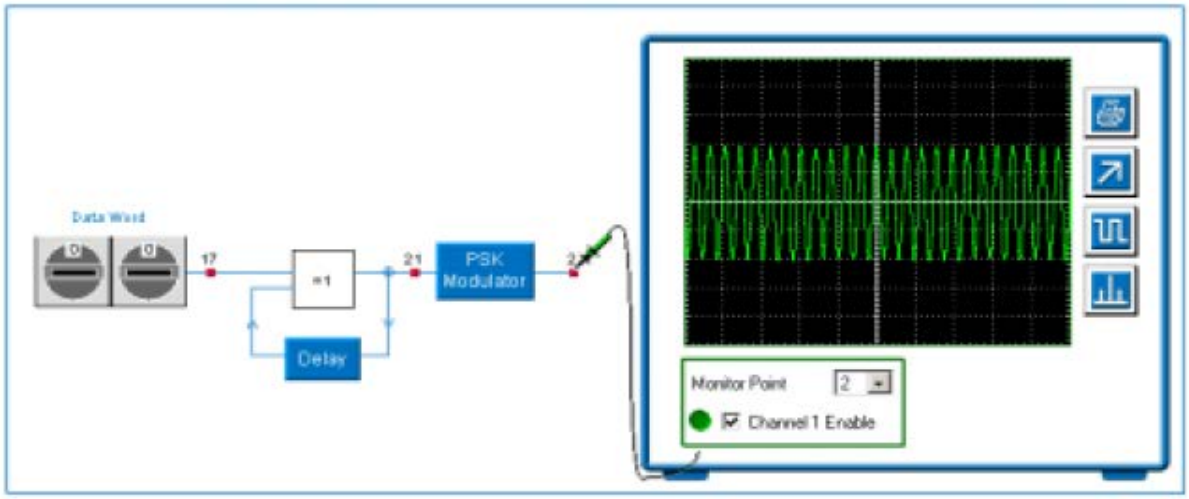


Figure 11.3

OBSERVATIONS AND RESULTS:

- Look at the monitor point 17 with the oscilloscope:
  1. Does the data input to the DPSK modulator changes with time?
  2. What binary data does this represent?

- Now look at monitor point 21 with the oscilloscope. Does the state of Exclusive-OR output changes with time?

- Now look at the monitor point 2 with the oscilloscope:
  1. Does the phase of modulated output changes with time?
  2. How is the phase of the modulated output related to the data input?

- Repeat these set of observations by setting the **MS bits switch (7)** and the **LS bits switch (8)** to F.

Demodulation

- Observe the signal after demodulation at monitor points 31 & 10 using the oscilloscope. See Figure.

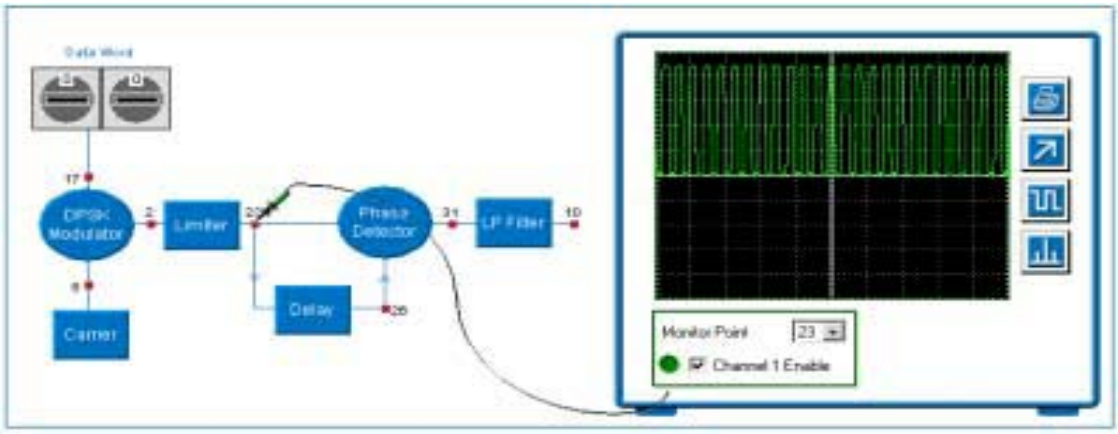


Figure 11.4

OBSERVATIONS AND RESULTS:

Ensure that the **MS bits switch (7)** and the **LS bits switch (8)** to 0.

- Look at monitor point *2* with the large oscilloscope display. Does the DPSK waveform change phase as you would expect?

- Look at the output of phase detector, monitor point 31, with the oscilloscope. Does the waveform corresponds to the originating data?

- Try a number of the **MS bits switch (7)** and the **LS bits switch (8).** Do the waveforms corresponds to the originating data?

LAB SESSION 12
Baseband Digital Transmission

OBJECTIVE:

- To implement Correlation Receiver and Matched Filter using simulation tool.

EXERCISE:

a)

Suppose that two orthogonal signals as shown in figure 8.1, are used to transmit binary information through an AWGN channel. The received signal in each bit interval of duration *Tb* is given by:

$$\mathbf{r(t) = si(t)+n(t)} \text{ for } i = 0;1 \text{ and } 0 \leq t \leq Tb.$$

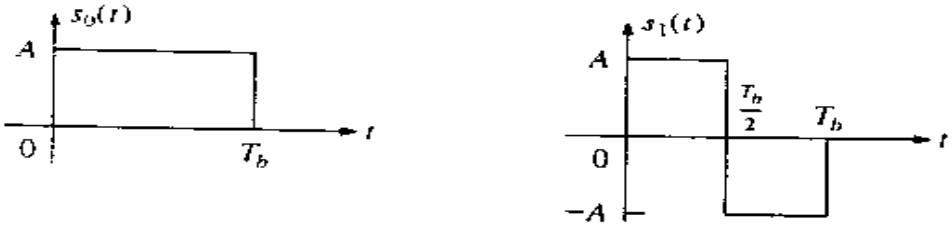


Figure 12.1 Signal waveform s_0t and s_1t for binary communication system

Suppose that the received waveform is sampled at a rate of 10/*Tb*-that is, at ten samples per bit interval. Hence, in discrete time, the signal waveform $s_0(t)$ with amplitude *A* is represented by the ten samples (*A; A;…..;A*), and the signal waveform $s_1(t)$ is represented by the ten samples (*A; A; A; A; A; -A;-A;-A;-A;-A*).

Consequently, the sampled version of the received sequence when $s_0(t)$ is transmitted it is:

$$\mathbf{rk = A + nk \qquad k = 1; 2;……; 10}$$

and when $s_1(t)$ is transmitted it is:

$$\mathbf{rk} = \begin{cases} \mathbf{+A + nk,} & \mathbf{1 \leq k \leq 5} \\ \mathbf{-A + nk,} & \mathbf{6 \leq k \leq 10} \end{cases}$$

where the sequence {nk} is i.i.d. zero mean, Gaussian with each random variable having the variance σ^2.

Write a MATLAB routine that generates the sequence {rk} for each of the two possible received signals, and perform a discrete-time correlation of the sequence {rk} with each of the two possible signals $s_0(t)$ and $s_1(t)$ represented by their sampled versions for different values of the additive Gaussian noise variance σ^2=0; σ^2= 0.1; σ^2 = 1.0;and σ^2 = 2.0. The signal amplitude may be normalized to *A* = 1. Plot the correlator's outputs at time instants *k* = 1; 2; 3;…..; 10.

b)

In this exercise, the objective is to substitute two matched filter in place of two correlators. The condition of generating signals is identical.

Write a MATLAB routine that generates the sequence {rk} for each of the two possible received signals, and perform a discrete-time matched filtering of the sequence {rk} with each of the two possible signals $s_0(t)$ and $s_1(t)$ represented by their sampled versions for different values of the additive Gaussian noise variance $\sigma^2=0$; $\sigma^2= 0.1$; $\sigma^2 = 1.0$;and $\sigma^2 = 2.0$. The signal amplitude may be normalized to $A = 1$. Plot the correlator's outputs at time instants $k = 1; 2; 3;\ldots..; 10$.

Reference:
Contemporary Communication System using MATLAB, John G. Proakis Masoud Salehi
Chapter 5, Baseband Digital Transmission

LAB SESSION13
Digital Transmission through Band limited Channel

OBJECTIVE:

Implement 11-tap transversal equalizer using

- Zero forcing solution
- Minimum MSE solution

EXERCISE:

a)

Consider a channel distorted pulse x(t) at the input to the equalizer given by the expression:

$$x(t) = \frac{1}{1 + (2t/T)^2}$$

The pulse is sampled at a rate of 2/T and equalized by zero forcing equalizer with 2K+1=11taps. Write a MATLAB program to solve for the coefficient of zero-forcing equalizer. Evaluate and plot the output of this equalizer for 50 sampled values.

b)

Write a general MATLAB program for computing the tap coefficients of an FIR filter equalizer of arbitrary length 2K+1 based on MSE criterion, given as input the sampled value of the pulse x(t) taken at the symbol rate and the spectral density of the additive noise N_o. Use the program to evaluate the coefficients of an 11-tap equalizer when the sampled value of x(t) are

$$x(nT) = \begin{cases} 1, & n = 0 \\ 0.5, & n = \pm 1 \\ 0.3, & n = \pm 3 \\ 0.1, & n = \pm 4 \end{cases}$$

N_o =0.01 and N_o= 0.1. Also evaluate the minimum MSE for the optimum equalizer coefficients.

Reference:
Contemporary Communication System using MATLAB, John G. Proakis Masoud Salehi
Chapter 6, Digital Transmission through Bandlimited Channel

LAB SESSION 14
Channel Capacity and Coding

OBJECTIVE:

- To plot the capacity of binary symmetric channel
- To plot the mutual information of the binary non symmetric channel
- To simulate a Block Code of (6,3) & (63, 57) order using MATLAB Communication Toolbox.

EXERCISE:

a)
Write a MATLAB script to plot the capacity of a binary symmetric channel with crossover probability *p* as a function of *p* for $0 \leq p \leq 1$. For what value of *p* is the capacity minimized, and what is the maximum value?

b)
A binary nonsymmetric channel is characterized by the conditional probabilities p(0|1)=0.2 and p(1|0)=0.4. Plot the mutual information I(X;Y) between the input and the output of this channel as a function of p=P(X=1). For what value of p is the mutual information maximized and what is the value of the maximum?

c)
Write a MATLAB script that generates (6, 3) & (63, 57) Hamming Code. Compute the BER for a (6, 3) Hamming code over a BSC with $\varepsilon = 0.005$. Choose N = 10,000. In your lab report, compare this result with the un-coded BER over a BSC with $\varepsilon = 0.005$.

Reference:
Contemporary Communication System using MATLAB, John G. Proakis Masoud Salehi
Chapter 8, Channel Capacity and Coding

LAB SESSION 15

This laboratory is intended to provide you hands-on experience on the Texas Instruments C6000 family of Digital Signal Processors (DSPs).

OBJECTIVES:

- Design a very simple project for the Texas Instruments C6000 family of DSP devices using MATLAB® and Simulink®.
- Run the project on the Texas Instruments DSK6713.

HARDWARE AND SOFTWARE REQUIREMENTS:

This laboratory was originally developed using the following hardware and software:

- MATLAB R2006b with Embedded Target for TI C6000 and Signal
- Processing Blockset.
- Code Composer Studio (CCS) v3.1
- Texas Instruments DSK6713 hardware.
- Microphone and computer loudspeakers/headphones.

PROCEDURE:

Starting a New Project

- There are several stages to follow in order to implement a project. These are now itemized.

Connecting the DSK6713

For MATLAB and Simulink to run correctly, you need to run the DSK6713.

- Connect the C6713 hardware to a USB port of the computer and turn on the supply to the board. If the board is powered, he green LEDs start flashing on the board during the self-test.
- Start Code Composer Studio for DSK6713 and use Debug -> Connect Figure 1

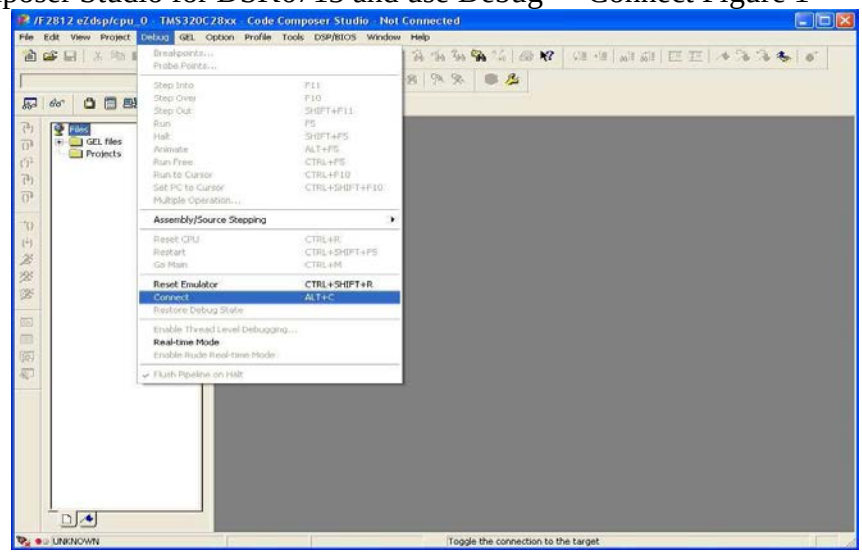


Figure 15.1 – Startup Screen for Code Composer Studio (CCS)

Creating a new MATLAB Model

- Start MATLAB 7.3.0 R2006b. In the top left hand corner of the MATLAB screen, select File -> New -> Model.

Saving the New Model

- Save the new model as “my_first_C6000_model.mdl”

Opening the Library Browser

- Select View-> Library Browser

The Simulink Library Browser

- A list of available blocks will appear. We are particularly interested in the “Embedded Target for TI C6000 DSP”. Click on this selection.

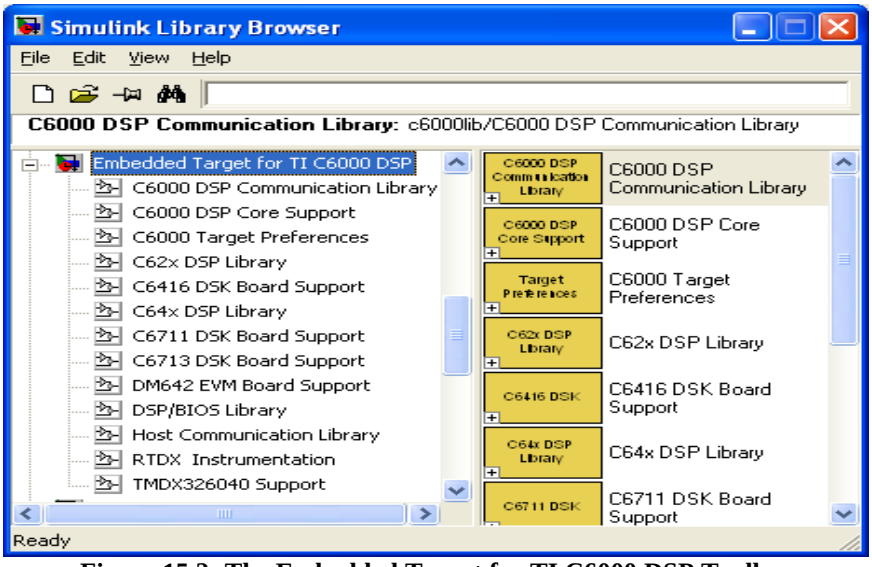


Figure 15.2- The Embedded Target for TI C6000 DSP Toolbox

Selecting Target Preferences

From “C6000 Target Preferences”, select “DSK6713”. Drag-and-drop the icon onto the new model.

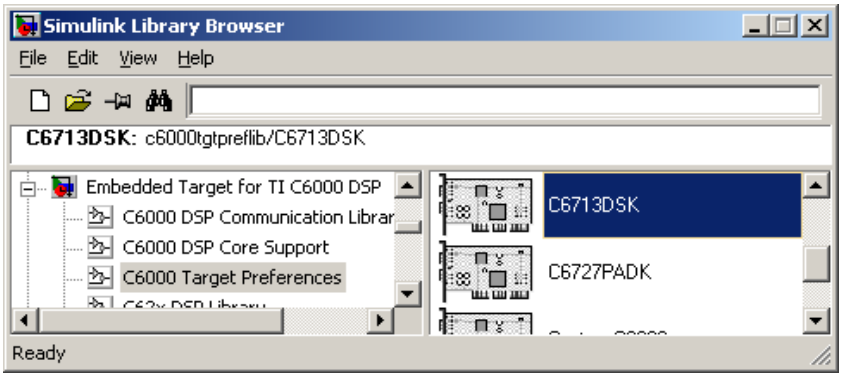


Figure 15.3 – Selecting Target Preferences

- Select “Yes” when the following screen appears:

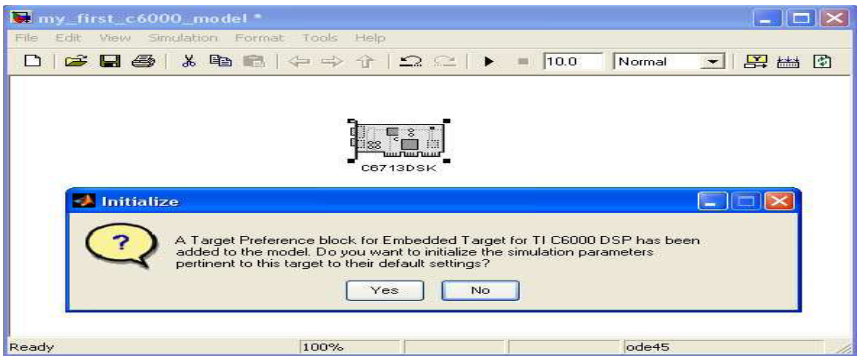


Figure 15.4 -Initializing Simulation Parameters

Adding and Configuring the Analog to Digital Converter(ADC)

- Return to the Simulink Library Browser. Select "DSK6713 Board Support". This lists the inputs and outputs to the DSK6713. Select the ADC. Dragand- drop the ADC icon onto your model. This controls the audio input.

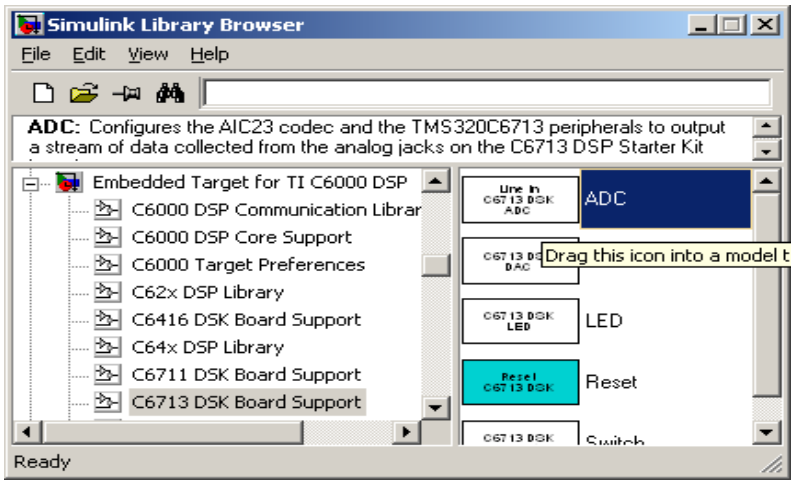


Figure 15.5 – The ADC Block

- For an input from a microphone, use "Mic In". For an input from a CD player or IPOD, use "Line In". We will use a microphone.

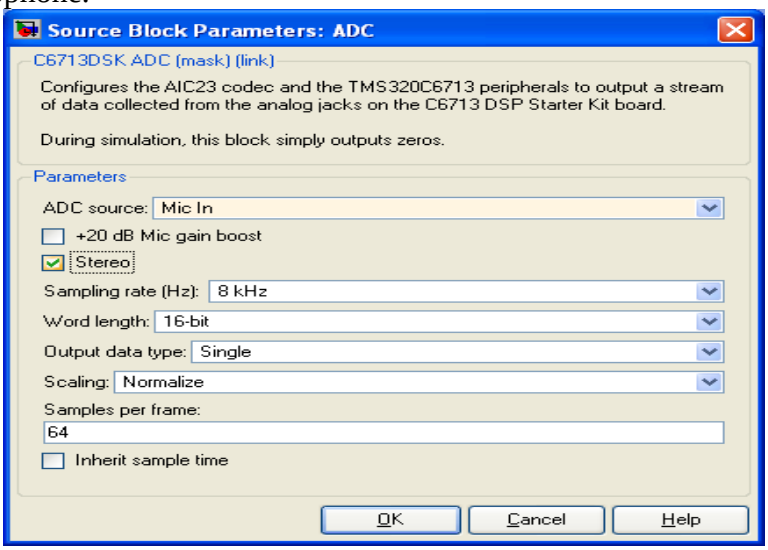


Figure 15.6 – Source Block Parameters for ADC

- For telephone quality, use "8kHz". For a CD data stream, use "44.1 kHz". For professional audio, use "48kHz". We will use "8kHz".

- If your microphone has low output, select "+20dB Mic gain boost". Set the Samples per frame to 1. This means that the data will be processed immediately. Click on "OK".

Adding and Configuring the Digital to Analog Converter(DAC)

- Return to the Simulink Library Browser. Select "DSK6713 Board Support".
- This lists the inputs and outputs to the DSK6713. Select the DAC. Drag-and-drop the DAC icon onto your model. This controls the audio output.

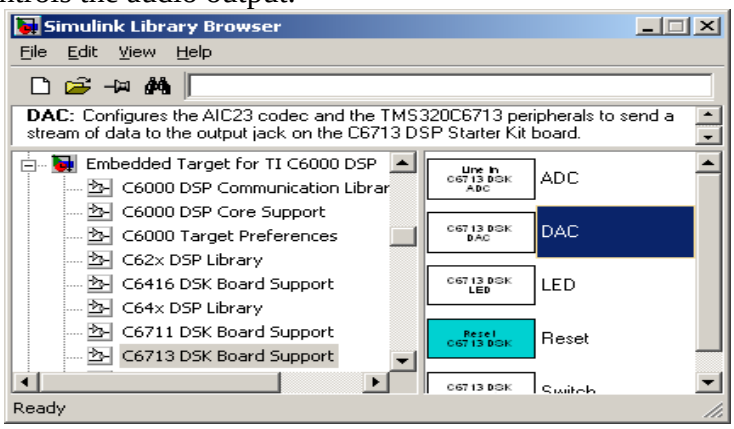


Figure 15.7 – The DAC Block

- Double click on the DAC block. The sampling rate should be the same as the input sampling rate.

- Set the Overflow mode to "Saturate". This prevents large positive numbers becoming negative. Click on "OK".

Concluding the First Model

- Click on the right hand side of the ADC block; drag the cursor to the DAC block. This will connect the two blocks together.

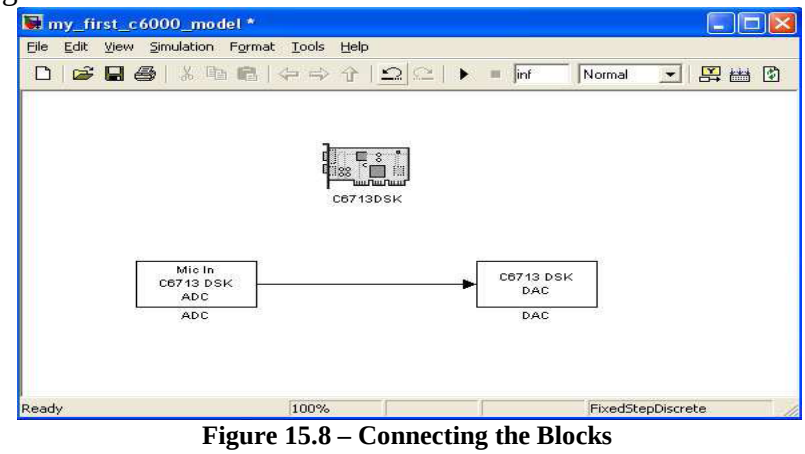


Figure 15.8 – Connecting the Blocks

The first model is now complete. This will feed an audio input from the microphone directly to the computer loudspeakers / headphones.

Building the Model

- Select Tools -> Real-Time Workshop -> Build Model.

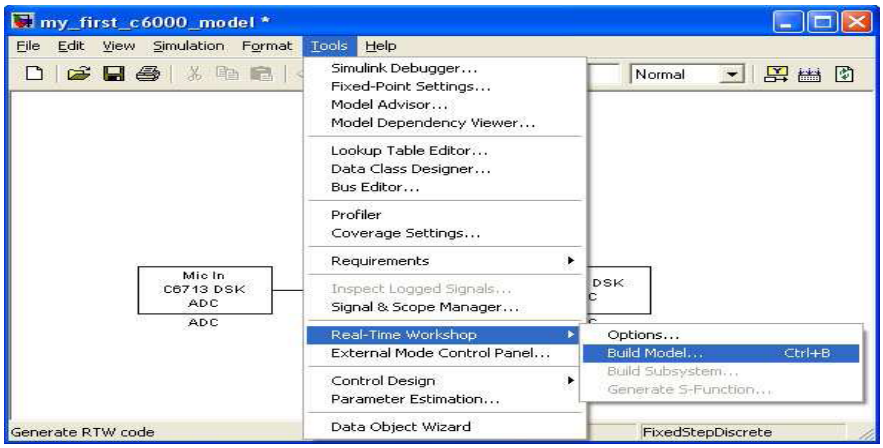


Figure 15.9 – Building the Model

Build in Progress

- MATLAB will then create a new project in Code Composer Studio and generate the code for it.

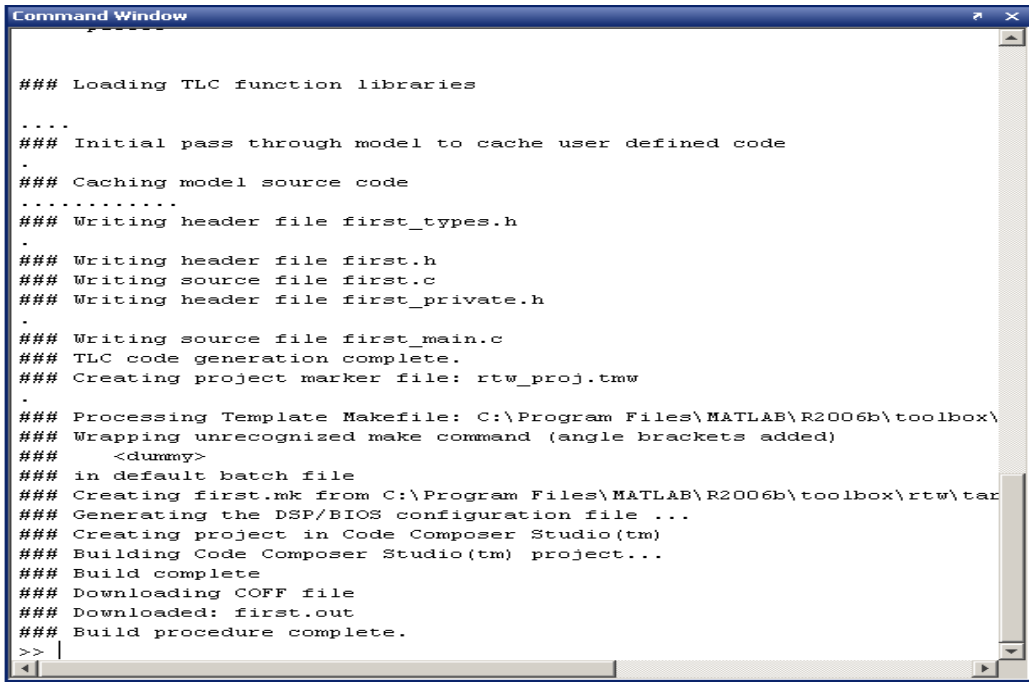


Figure 15.10 – Building the Model

The model has now been built and is running on the DSK6713. If you speak into the microphone, you will hear the words from the loudspeakers / headphones.

Checking the Generated Code

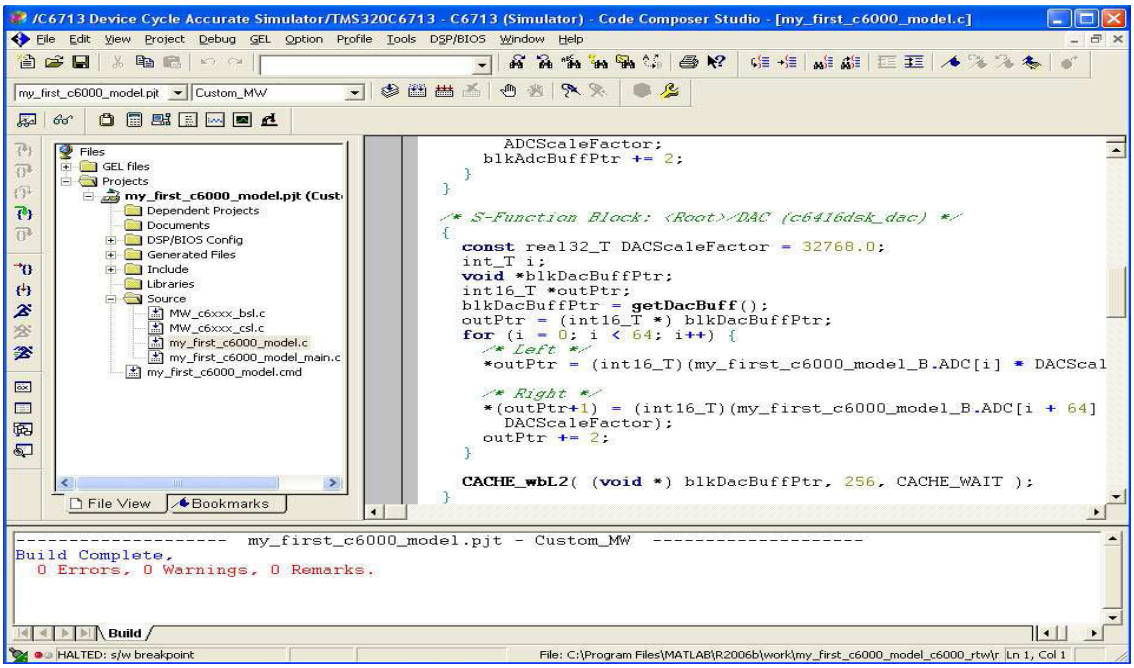


Figure 15.11 – Checking the Generated Code

You may wish to inspect the C code that has been generated by MATLAB. Return to Code Composer Studio (CCS). Click on the file "my_first_6000_model.c" to select the source file.

Adding Some Signal Processing

So far, we have fed the microphone directly through to the loudspeakers / headphones. This is not very interesting. We shall now add some digital signal processing.

Adding a Sine Wave Generator

The Signal Processing Blockset

- In the MATLAB environment, select View -> Simulink Library Browser. Select Signal Processing Blockset then highlight "Signal Processing Sources".

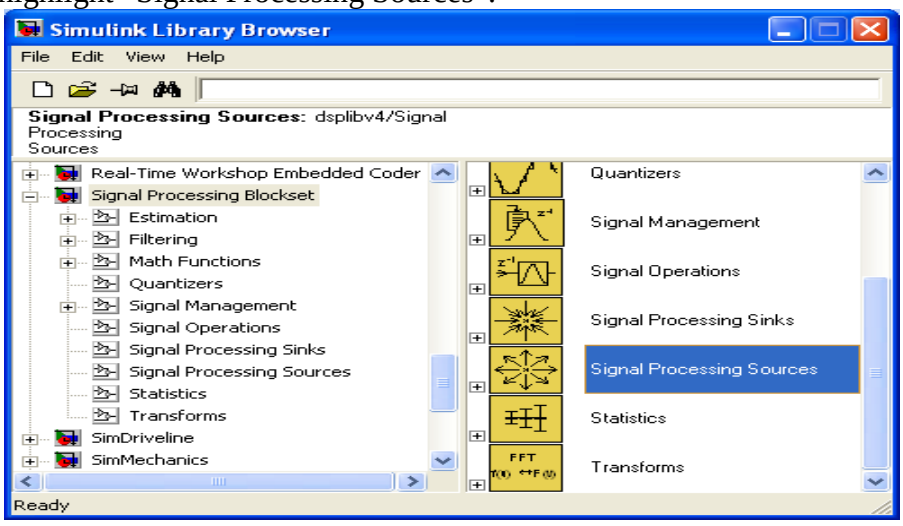


Figure 15.12 – Signal Processing Blockset

- From the Signal Processing Sources, drag-and-drop the "Sine wave" block onto the model.
-

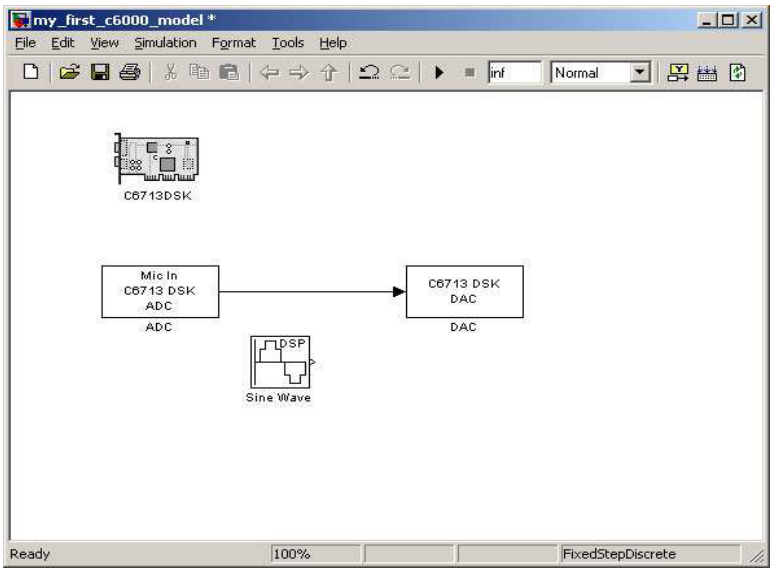


Figure 15.13 – Adding a Sine Wave Generator

- Double-click on the Sine Wave block.
- Set the Frequency to 500 Hz.
- Set the Sample time to 1/8000. This is the same rate as the ADC and DAC.

Adding a Product

- Select View -> Simulink Library Browser -> Commonly Used Blocks.
- Highlight the “Product” block, and drag-and-drop it into the “product” block onto the model.

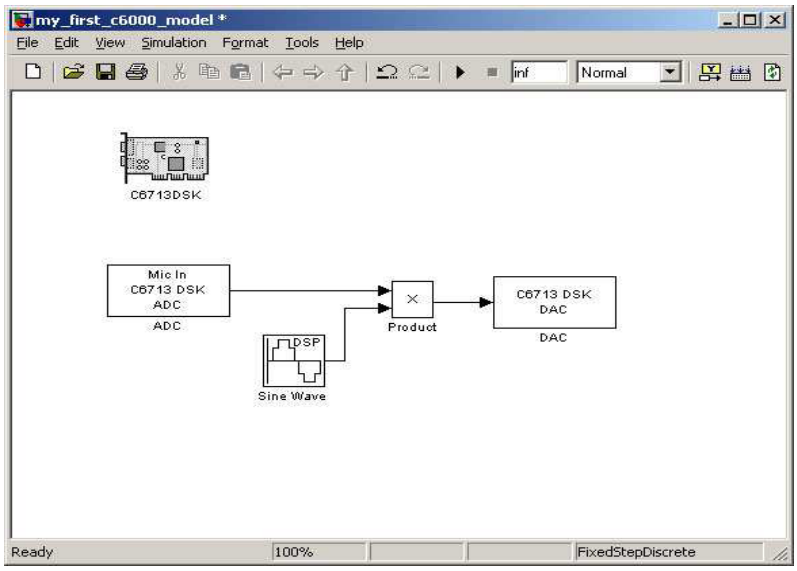


Figure 15.14 – Connecting the Blocks

- Connect the blocks as shown in the diagram.

Adding a Title

- Double click on the white background then type in an appropriate title.

Building the Model

- Select Tools -> Real-Time workshop -> Build Model

Speak into the Microphone

You have now implemented your first DSP project. The algorithm you have just implemented is known as "ring modulation". It is widely used in television and films to generate "alien voices".

How Ring Modulation Works

The multiplication of two sinusoidal waves, with frequencies ***f1*** and ***f2***, generatesa new signal with frequency components at the sum (***f1***+***f2***) and the difference (***f1***-***f2***) of the original frequencies. This can mean that the output bears no resemblance to the inputs!

Suppose we multiply 500 Hz by 2000 Hz, the resultant frequencies will be 2000 - 500 = 1500 Hz and 2000 + 500 = 2500 Hz. The output is as shown below:

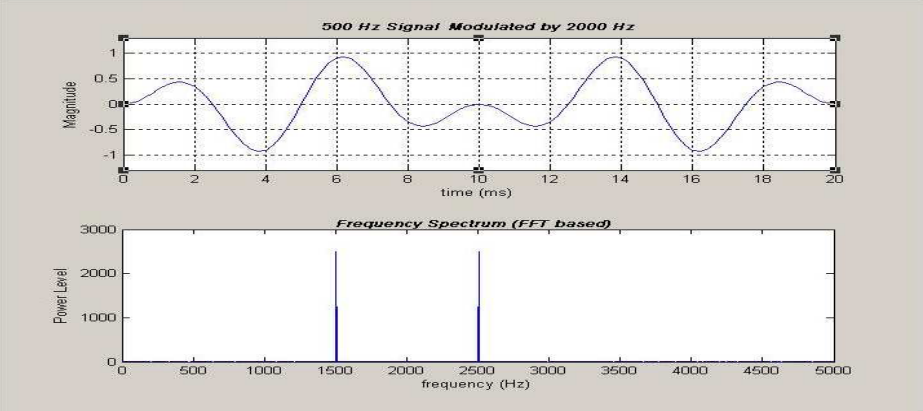


Figure 15.15 – Ring Modulation to produce Sum and Difference Frequencies

Things to Try

Change Sampling Rate

Go into the ADC and DAC blocks and change the sampling rates. What is the difference in audio quality between 8 kHz and 48 kHz sampling rates?

Change Frequency of Signal Generator

Change the frequency of the signal generator to values between 100 Hz and 15 kHz.

Change Signal Generator

Change the signal generator from "Sine Wave" to "Chirp".