DIGITAL COMMUNICATIONS LABORATORY

LIST OF EXPERMENTS

1. TIME DIVISION MULTIPLEXING AND DEMULTIPLEXING
2. PULSE CODE MODULATION AND DEMODULATION
3. DIFFERENTIAL PULSE CODE MODULATION & DEMODULATION
4. DELTA MODULATION AND DEMODULATION
5. FREQUENCY SHIFT KEYING MODULATOR AND DEMODULATOR
6. PHASE SHIFT KEYING MODULATOR AND DEMODULATOR
7. DIFFERENTIAL PHASE SHIFT KEYING MODULATOR AND DEMODULATOR
8. LINEAR BLOCK CODE
9. COMPANDING
10. BINARY CYCLIC CODE
11. CONVOLUTION CODE
12. SORCE ENCODER AND DECODER

EXP NO: 01

TIME DIVISION MULTIPLEXING DEMULTIPLEXING

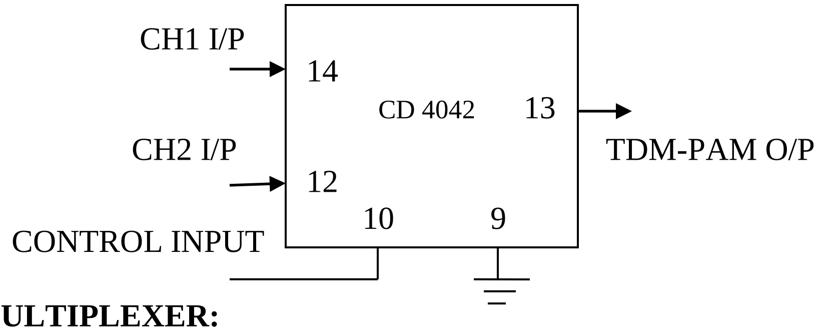
AIM: To study the Time Division Multiplexing and Demultiplexing.

APPARATUS:

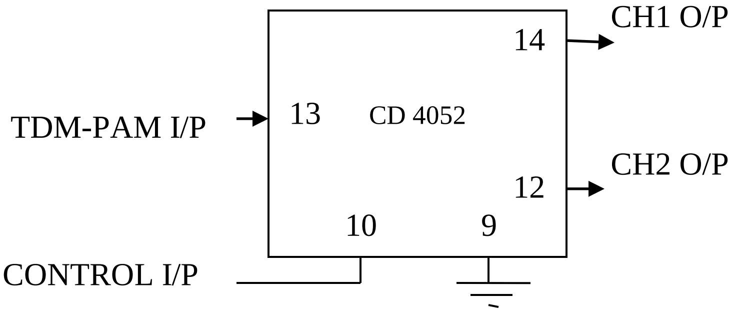
1. TDM trainer kit
2. Set of patch chords.
3. CRO

CIRCUIT DIAGRAM:

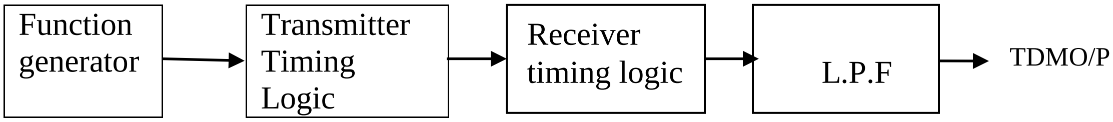
TDM MULTIPLEXER:



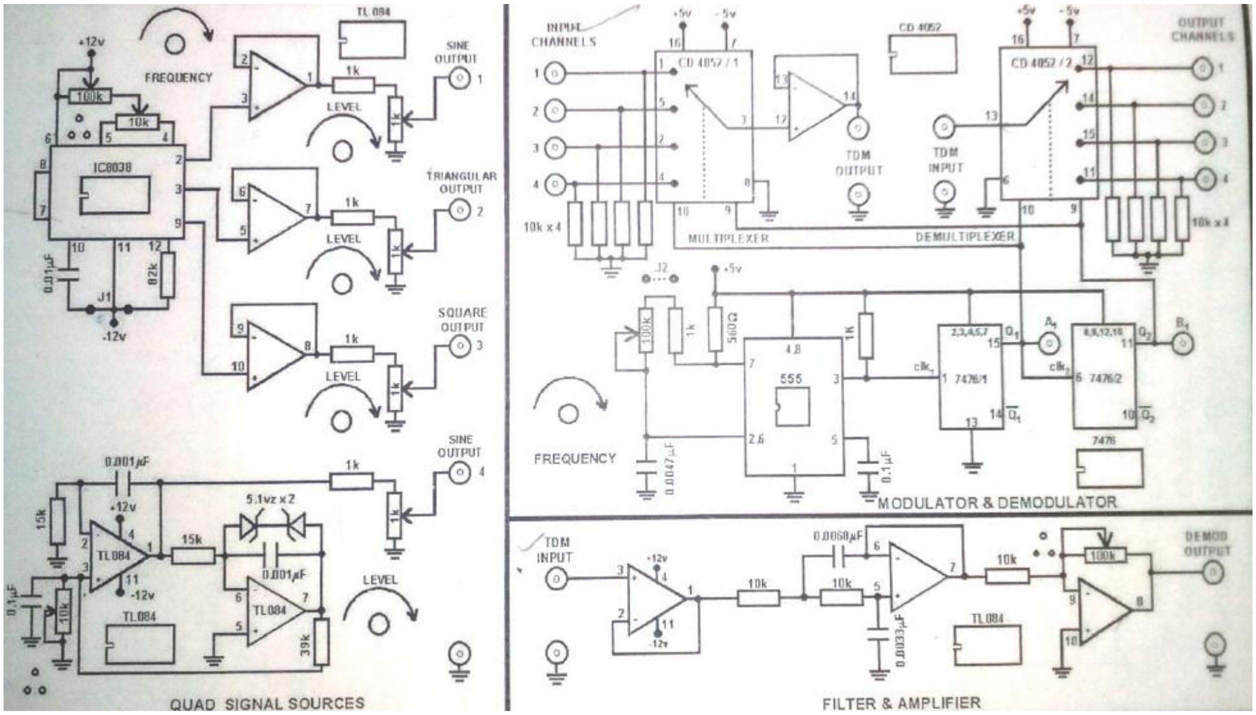
TDM DEMULTIPLEXER:



BLOCK DIAGRAM:



Circuit diagram:



THEORY:

A time division multiplex system enables the joint utilization of a common communication channel by a plurality of independent message sources without mutual interference among them. The block diagram as shown in figure illustrates the concept of TDM. Each input signal is first restricted in bandwidth by a low pass anti aliening filter to remove the frequencies that are non-essential to an adequate signal representation. The low pass filter outputs are then applied to commentator, which is usually implanted using electronic switching circuitry the function of the commutator, is twofold. To take a narrow sample of each of the N input messages at rate fs that are slightly higher than 2w where W is the cutoff frequency of the anti-aliening. To sequentially interleave there N samples inside the sampling interval TS

In deed this later function is the essence of the time division multiplexing operation following the communication process the multiplied signal is applied to pulse modulator, the purpose of which is to transform the multiplied signal into a form suitable for transmission over the communication channel it is clear that the use of time division multiplying introduces a band width expansion factor N because the scheme must squeeze N samples derived from N independent message sources into a time slot equal to one sampling interval at the receiving end of the system, the receive signal is applied to pulse demodulator, which performs the reverse operation of the pulse modulator.

The narrow samples produced at the pulse demodulator output are distributed to the appropriate low pass reconstruction filter by means of a dissimulators which operates in synchronism with the commutator in the transmitter the is synchronization is essential for a satisfactory operation of the system. The way this synchronization is implemented depends naturally on the method of pulse modulation use to transmit the multiplied sequence of samples.

The TDM systems are highly sensitive to dispersion in the common channel .so accurate equalization of both magnitude3 and phase response of the channel is necessary to ensure a satisfactory operation of the system.

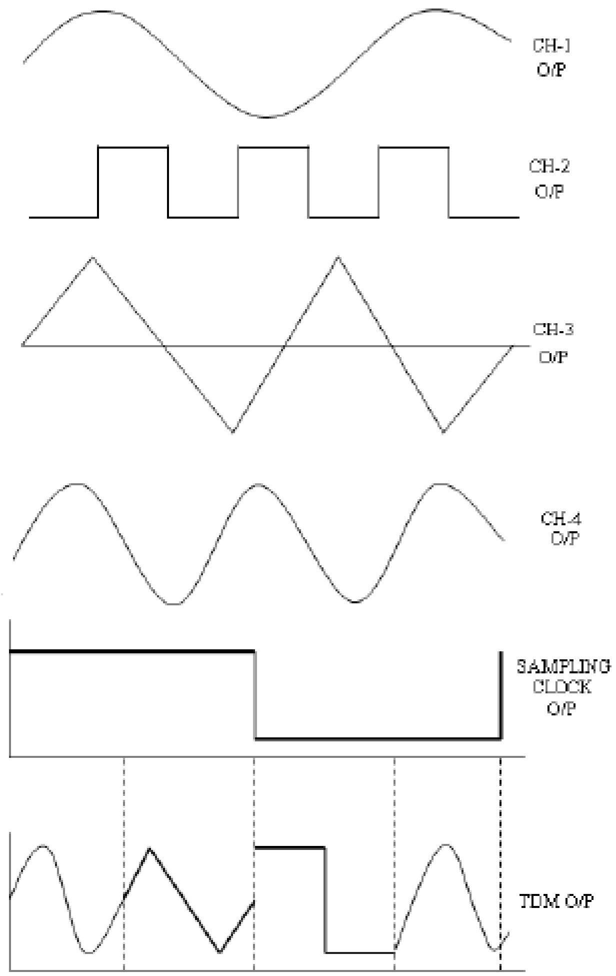
PROCEDURE:

1. Connections were made as per the circuit diagram
2. The bit A_2 of the address generator is applied to one channel of CRO & triggered the CRO with respect to the same channel.
3. The output of 8 to 1 line multiplexer is observed on second channel of the CRO.
4. A high (+5v) signal is applied to the 8 multiplexing inputs one by one & multiplexer output is observed and the total time division by each channel is observed with respect to the address generator.
5. Now , the 8 to 1 line multiplexer output is connected to the 1 to 8 line demultiplexer.
6. The demultiplexer's outputs are inverted as per the logic design of the IC.
7. Any data available from the data generator is given to any multiplexing input and the output is observed at the corresponding demultiplexer output.
8. Now, different data is connected to different outputs which are inverted are observed and compared

PRECAUTIONS:

1. Avoid loose and wrong connections.
2. The output waveforms of multiplexer & demultiplexer are observed & noted without any parallax errors.

MODEL GRAPHS



APPLICATIONS OF TDM:

- The plesiochronous digital hierarchy (PDH) system, also known as the PCM system, for digital transmission of several telephone calls over the same four-wire copper cable (T-carrier or E-carrier) or fiber cable in the circuit switched digital telephone network
- The synchronous digital hierarchy (SDH)/synchronous optical networking (SONET) network transmission standards that have replaced PDH.
- The Basic Rate Interface and Primary Rate Interface for the Integrated Services Digital Network (ISDN).
- The RIFF (WAV) audio standard interleaves left and right stereo signals on a per-sample basis
- The left-right channel splitting in use for stereoscopic liquid crystal shutter glasses

RESULT:

The response of the time division multiplexing and demultiplexing system is observed & compared

VIVA QUESTIONS :

1. What is multiplexing?
2. What are the types of multiplexing?
3. Define time division multiplexing?
4. What are the advantages of Time division multiplexing?
5. What are the Types of TDM?

Ans: Synchronous TDM and asynchronous TDM or statical TDM.In Synchronous Time Division Multiplexing, multiplexer allocates exactly the same time slot to each device at all times. In Asynchronous means something different in multiplexing in the area of data communication.It means flexible or not fixed.

6. What are the applications of synchronous TDM?
   Ans: T-1 multiplexing (the classic)
   ISDN (Integrated Services Digital Networks) multiplexing
   SONET (Synchronous Optical NETwork)
7. What are the Advantages of synchronous TDM ?
   Digital signals
   Relatively simple
   Commonly used with ISDN (Integrated Services digital network)
8. What are the Disadvantages of synchronous TDM?
9. What is the difference between TDM & FDM?

GRAPH

GRAPH

EXPNO: 02

PULSE CODE MODULATION AND DEMODULATION

AIM: To study the operation and working of the pulse code modulation and demodulation.

APPARATUS:

PCM modulator and demodulator kit.
Set of patch chords.
CRO

THEORY:

Pulse Code Modulation (PCM) is different from Amplitude Modulation (AM) and Frequency Modulation (FM) because those two are continuous forms of modulation. PCM is used to convert analog signals into binary form. In the absence of noise and distortion it is possible to completely recover a continuous analog modulated signals. But in real time they suffer from transmission distortion and noise to an appreciable extent. In the PCM system, groups of pulses or codes are transmitted which represent binary numbers corresponding to modulating signal voltage levels. Recovery of the transmitter information does not depend on the height, width, or energy content of the individual pulses, but only on their presence or absence. Since it is relatively easy to recover pulses under these conditions, even in the presence of large amounts of noise and distortion, PCM systems tend to be very immune to interference and noise. Regeneration of the pulse reroute is also relatively easy, resulting in system that produces excellent result for long distance communication.

PCM ENCODING:

The encoding process generates a binary code number corresponding to modulating signal voltage level to be transmitted for each sampling interval. Any one of the codes like binary, ASCII etc, may be used as it provides a sufficient number of different symbols to represent all of the levels to be transmitted. Ordinary binary number will contain a train of'1' and '0' pulses with a total of log2N pulses in each number. (N is no of levels in the full range). This system is very economical to realize because it corresponds exactly to the process of analog – to – digital (A/D) conversion.

QUANTIZATION:

The 1st step in the PCM system is to quantize the modulating signal. The modulating signal can assume an infinite no. of different level between the two limit values, which define the range of the signal. In PCM a coded no is transmitted for each level sampled in the modulating signal. If the exact no corresponding to the exact voltage were to be transmitted for every sample, an infinitely large no of different code symbols would be needed. Quantization has the effect of reducing this infinite no of levels to a relatively small number, which can be coded without difficulty. In the quantization process, the total range of the modulating signal is divided into a no of small sub ranges. The number will depend on the nature of the modulating signal and will form as few as 8 to as many as 128 levels. A number that is an integer power of two is usually chosen because of the ease of generating binary codes. The result is stepped waveform, which follows the counter of the original modulating signal with each step synchronized to the sampling period.

The quantized staircase waveform is an approximation to the original waveform. The difference between the two-wave form amounts to “noise” added to the signal by the quantizing circuit. The mean square quantization noise voltage has a value of

$$E_{np}^2 = \frac{S^2}{12}$$

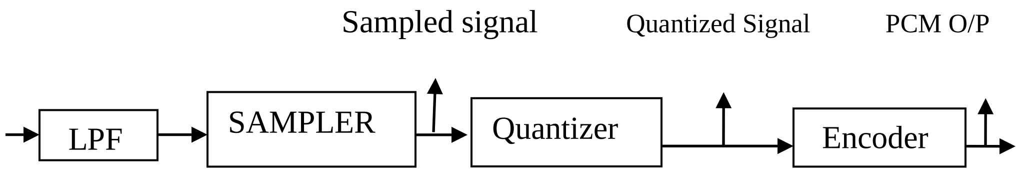
Where S is the voltage of each step. As a result the number of quantization levels must be kept high in order to keep the quantization noise below some acceptable limit given by the power signal-to-noise ratio, which is the ratio of average noise power.

DECODING:

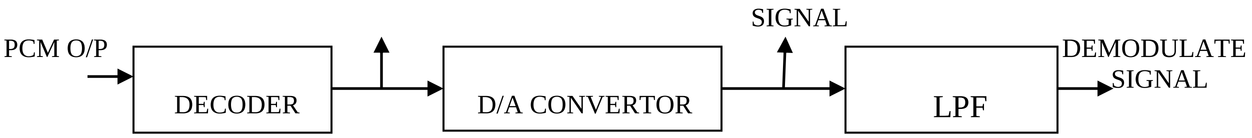
The decoding process reshapes the incoming pulses and eliminates most of the transmission noise. A serial to parallel circuit passes the bits in parallel groups to a digital to analog converter (D/A) for decoding. Thus decoded signal passes through a sample and hold amplifier, which maintains the pulse level for the duration of the sampling period, recreating the staircase waveform approximation of the modulation signal. A low-pass filter may be used to reduce the quantization noise.

<u>BLOCK DIAGRAM</u>:

MODULATOR:



DEMODULATOR:



CIRCUIT DIAGRAM:

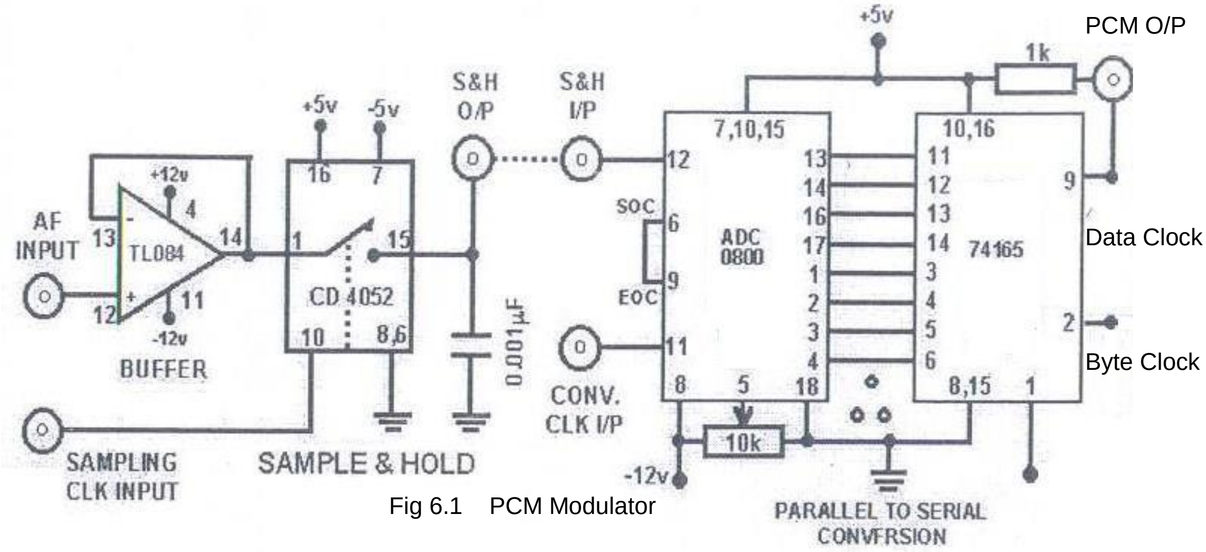


Fig 6.1 PCM Modulator

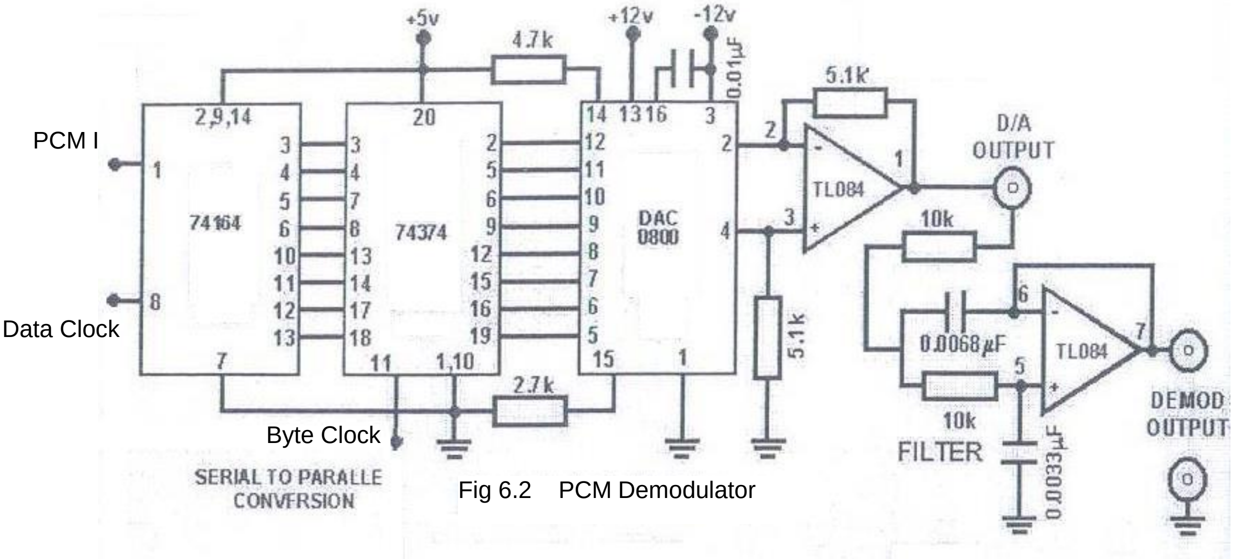


Fig 6.2 PCM Demodulator

PROCEDURE:

1. AC adaptor is connected to the mains and the other side to the experimental trainer kit.
2. PCM modulator and demodulator systems are switched ON.
3. The frequency of the clock is measured.
4. The DC variable voltage is applied as the modulating signal for visual convenience.
5. The clock is connected to the timing and control circuit.
6. The binary wore from LED’s is noted i.e.; LED ON represents‘1’and OFF represents ‘0’.
7. Sampling clock is connected to the CH1 of CRO and the serial data is observed by connecting it to CH2 of CRO and the binary data is observed.
8. The same word, which is at the transmitter, is observed at the receiver end.
9. Now, again the modulating signal is applied at the input.
10. The quantized level output waveform is observed at the output of D/A converter and the demodulated output is observed.

PRECAUTIONS:

1. Precautions are taken to avoid loose connections.
2. The output waveforms are observed without any parallax errors.

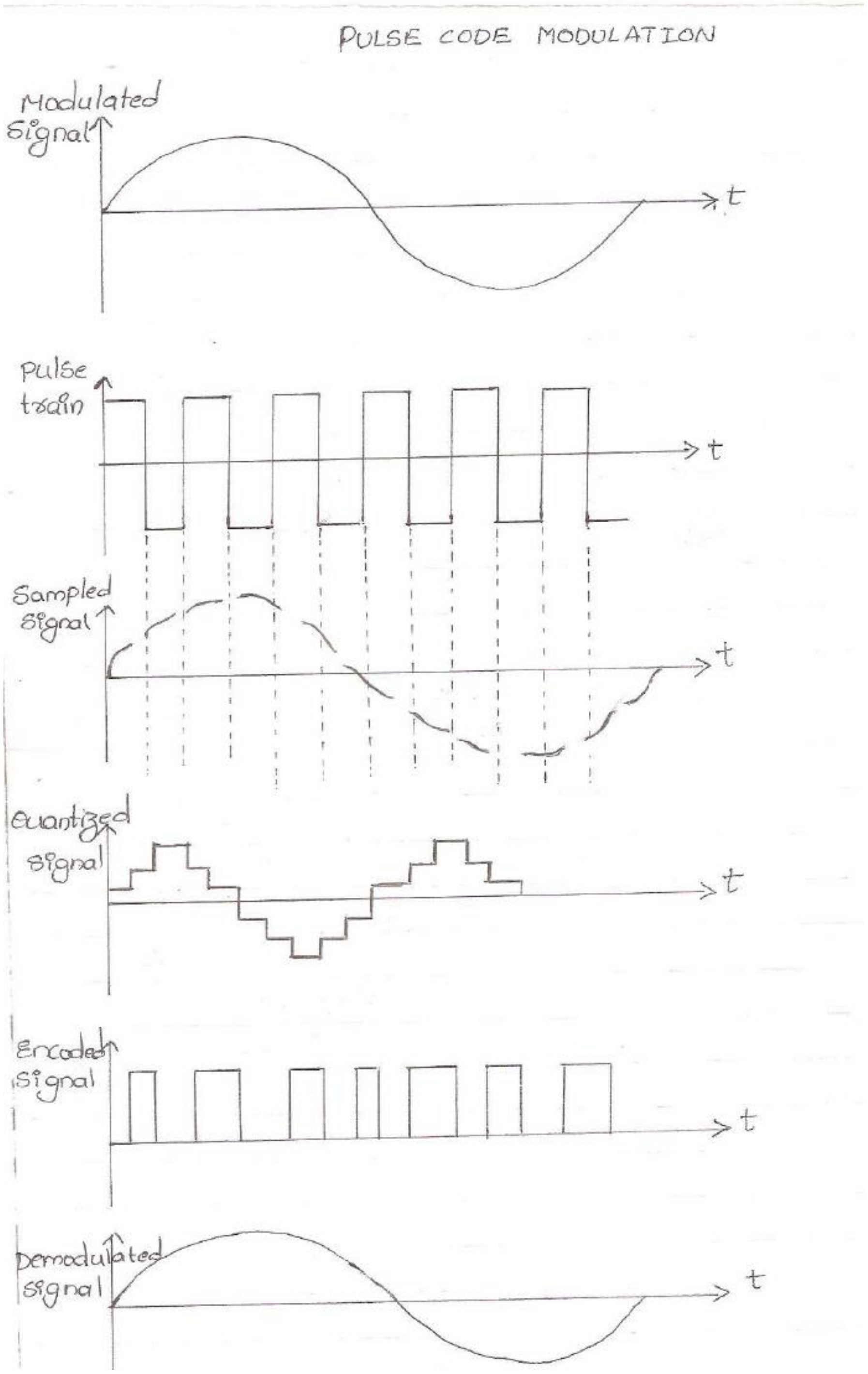
APPLICATIONS OF PCM:

1. PCM is used to sound-signal distribution in a broadcasting network.
2. PCM is used with T-1 and T-3 carrier systems. These carrier systems combine the PCM signals from many lines and transmit them over a single cable or other medium.
3. PCM is also the usual digital method used for music audio playback of music CDs. While supported by DVDs, DVDs have a greater volume so they use Linear PCM, which has a higher sampling rate – up to 24-bit at a sampling rate of 96 kHz.

VIVA QUESTIONS :

1. What is PCM draw the block diagram of a typical PCM system?
2. How can an analog voice signal be digitized?
3. What is quantization? How is this error rrinimized?
4. Explain sampling theorem for a band pass signal?
5. A message signal is band limited to a frequency range of 70— 12 kHz, find the minimum sampling frequency?
6. How messages signal is 12econstructed from the sampled values?
7. What is PCM draw the block diagram of a typical PCM system?
8. State and explain sampling theorem
9. What is Nyquist rate of sampling.
10. In a PCM system the minimum SNR required is 40dB, how many bits of quantization is required.

MODEL GRAPHS:



RESULT: The PCM modulated and demodulated waveforms are observed and recorded.

GRAPH

EXPNO: 03

DEFFERENTIAL PULSE CODE MODULATION AND DEMODULATION

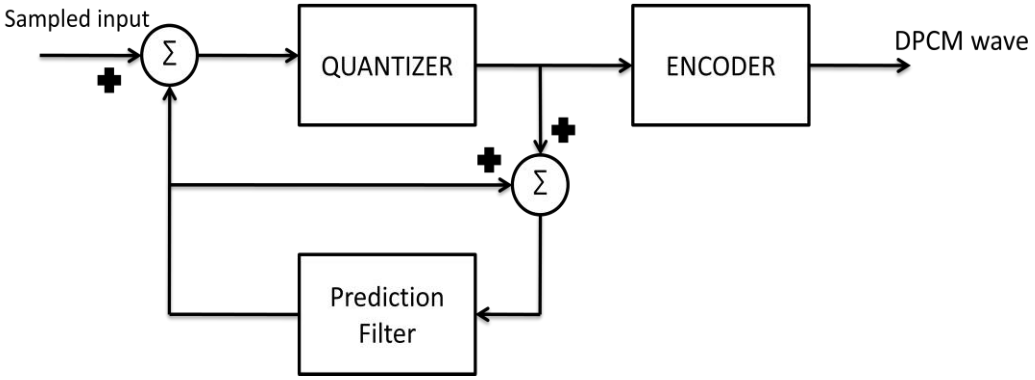
AIM: To study the differential pulse code modulation and demodulation by sending variable frequency sine wave variable DC signal input.

APPARATUS:

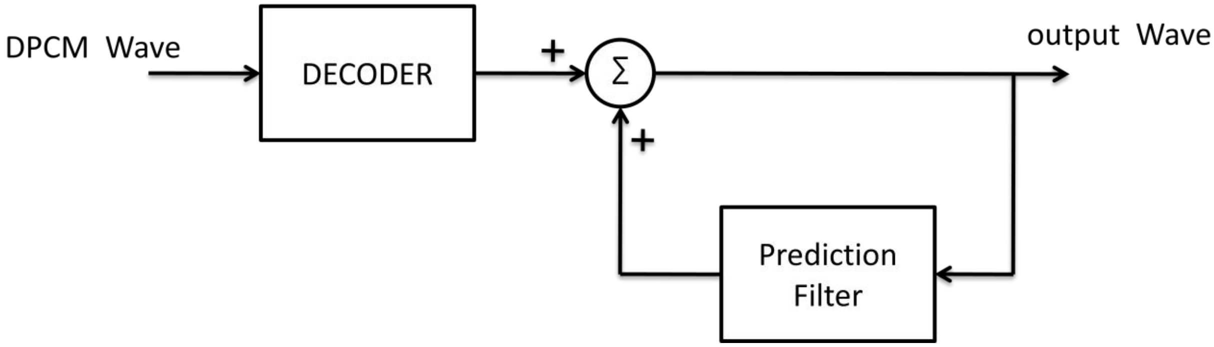
1. DPCM modulator and demodulator kit.
2. Set of patch chords.
3. CRO.

BLOCK DIAGRAM:

DPCM MODULATOR:



DPCM DEMODULATOR:



THEORY:

Pulse Code Modulation (PCM) is different from Amplitude Modulation (AM) and Frequency Modulation (FM) because those two are continuous forms of modulation. PCM is used to convert analog signals into binary form. In the absence of noise and distortion it is possible to completely recover a continuous analog modulated signals. But in real time they suffer from transmission distortion and noise to an appreciable extent. In the PCM system, groups of pulses or codes are transmitted which represent binary numbers corresponding to modulating signal voltage levels. Recovery of the transmitter information does not depend on the height, width, or energy content of the individual pulses, but only on their presence or absence. Since it is relatively easy to recover pulses under these conditions, even in the presence of large amounts of noise and distortion, PCM systems tend to be very immune to interference and noise.

Regeneration of the pulse reroute is also relatively easy, resulting in system that produces excellent result for long distance communication. Differential PCM is quite similar to ordinary PCM. However, each word in this system indicates the difference in amplitude, positive or negative, between this sample and the previous sample. Thus the relative value of each sample is indicated rather than the absolute value as in normal PCM. In this each amplitude is related to the previous amplitude, so that large variations from one sample to the next are unlikely. This being the case, it would take fewer bits to indicate to indicate the size of the amplitude change than the absolute amplitude, and so a smaller bandwidth would be required for the transmission. The differential PCM system has not found wide acceptance because complications in the encoding and decoding process appear to out weigh advantages gained.

DPCM ENCODING:

DPCM Encoding is similar to the PCM encoding, except that initial stage employs Delta modulation after that PCM encoding is following. The encoding process generates a binary code number corresponding to modulating signal voltage level to be transmitted for each sampling interval. Any one of the codes like binary, ASCII etc, may be used as it provides a sufficient number of different symbols to represent all of the levels to be transmitted. Ordinary binary number will contain a train of'1' and '0' pulses with a total of log2N pulses in each number. (N is no of levels in the full range). This system is very economical to realize because it corresponds exactly to the process of analog – to – digital (A/D) conversion. The 1st step in the PCM system is to quantize the modulating signal. The modulating signal can assume an infinite no.of different level between the two limit values, which define the range of the signal. In PCM a coded no is transmitted for each level sampled in the modulating signal. If the exact no corresponding to the exact voltage were to be transmitted for every sample, an infinitely large no of different code symbols would be needed. Quantization has the effect of reducing this infinite no of levels to a relatively small number, which can be coded without difficulty.

In the quantization process, the total range of the modulating signal is divided into a no of small sub ranges. The number will depend on the nature of the modulating signal and will form as few as 8 to as many as 128 levels. A number that is an integer power of two is usually chosen because of the ease of generating binary codes. The result is stepped waveform which follows the counter of the original modulating signal with each step synchronized to the sampling period. The quantized staircase waveform is an approximation to the original waveform. The difference between the two-wave form amounts to “noise” added to the signal by the quantizing circuit. The mean square quantization noise voltage has a value of

$$E_{np}^2 = \frac{S^2}{12}$$

Where S is the voltage of each step. As a result the number of quantization levels must be kept high in order to keep the quantization noise below some acceptable limit given by the power signal-to-noise ratio, which is the ratio of average noise power.

DECODING:

The decoding process reshapes the incoming pulses and eliminates most of the transmission noise. A serial to parallel circuit passes the bits in parallel groups to a digital to analog converter (D/A) for decoding. Thus decoded signal passes through a sample and hold amplifier, which maintains the pulse level for the duration of the sampling period, recreating the staircase waveform approximation of the modulation signal. A low-pass filter may be used to reduce the quantization noise.

CIRCUIT DIAGRAM:

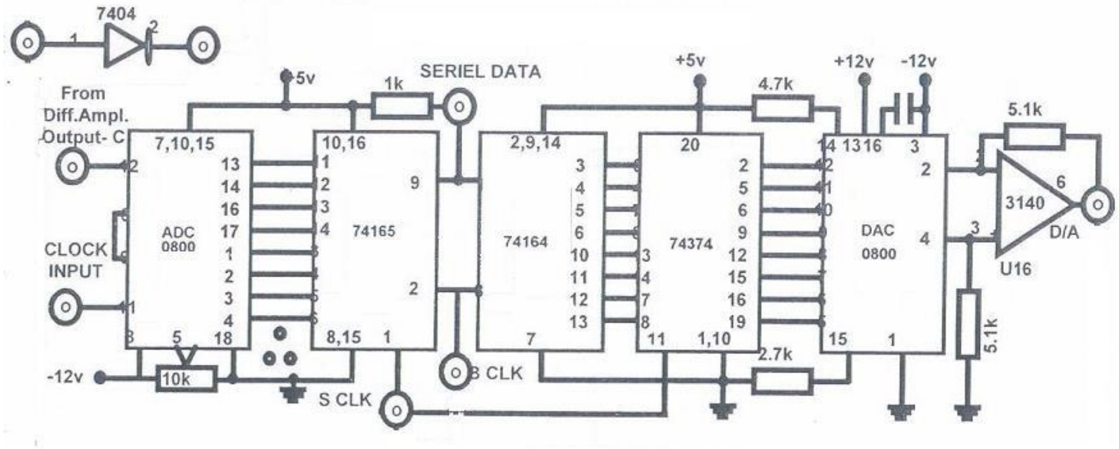


Fig 7.1 DPCM Modulator

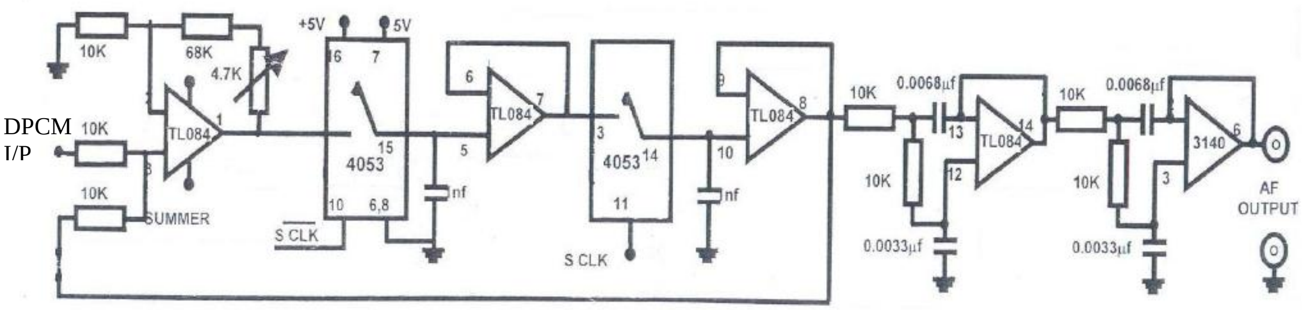
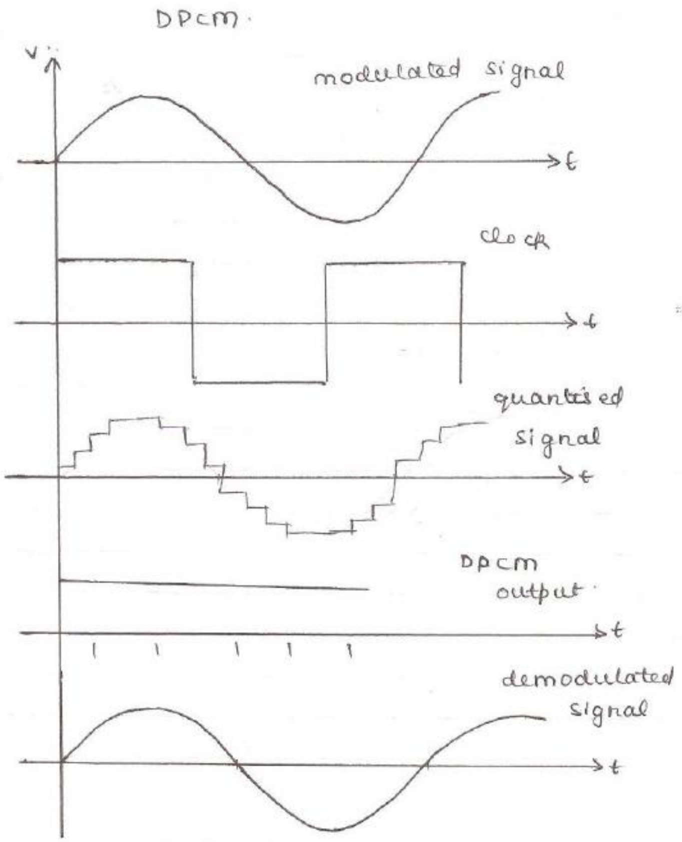


Fig. 7.2 DPCM Demodulator

PROCEDURE:

1. The AC adaptor is connected to the mains and the other side to the experimental trainer kit.
3. The power supply is switched ON.
4. The variable DC signal is applied to the input terminals of the DPCM modulator
5. The sampling signal output is observed on one channel of the CRO.
6. By adjusting the DC voltage potentiometer, the DPCM output waveform is observed.
7. Now, by disconnecting the DC voltage, AF oscillator output is applied to the input of DPCM modulator and the output of the conditioning amplifier and DPCM output's in synchronization with the sampling signal is observed.
8. During demodulation, the DPCM is applied to the demodulator output and DPCM demodulator output is observed.

MODELGRAPHS:



APPLICATIONS:

1. DPCM is usually used with lossy compression techniques, like coarser quantization of differences can be used, which leads to shorter code words.
2. This is used in JPEG and in adaptive DPCM (ADPCM), a common audio compression method.

VIVA QUESTIONS:

1. Define DPCM?
2. What is the effect of reducing bit resolution?
3. Does the low pass filter remove all the quantization error?

RESULT:

The DPCM modulated and demodulated waveforms are observed and studied.

GRAPH

EXPNO: 04

DELTA MODULATION AND DEMODULATION

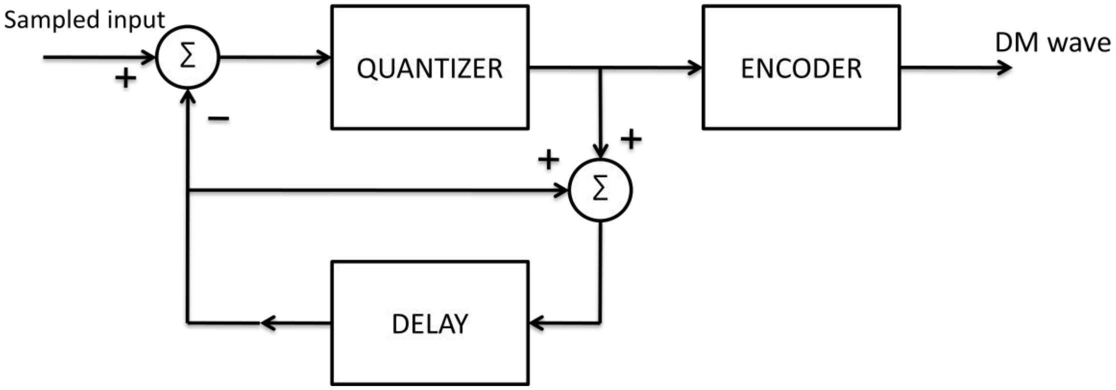
AIM: To study the modulation and demodulation by comparing the present signal with the previous signal of the given modulating signal.

APPARATUS:

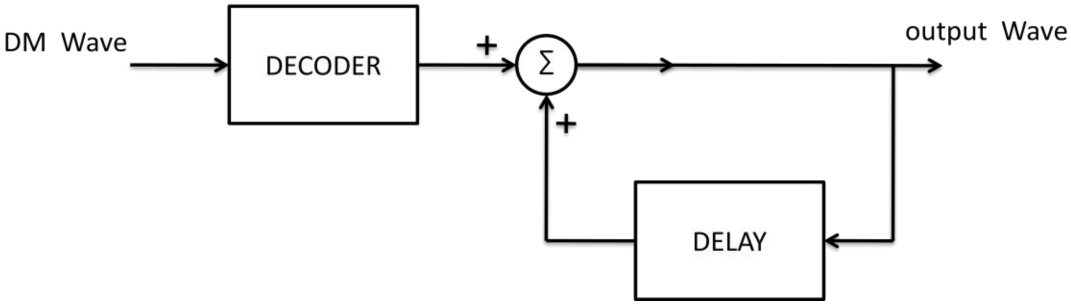
1. Delta modulator and demodulator trainer kit.
2. Set of patch chords.
3. CRO.

BLOCK DIAGRAM:

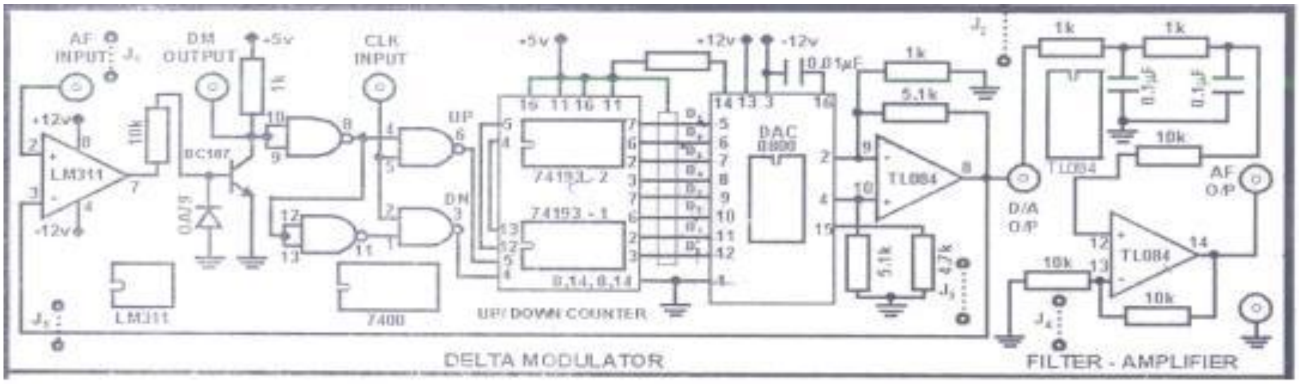
DM MODULATOR:



DM DEMODULATOR:

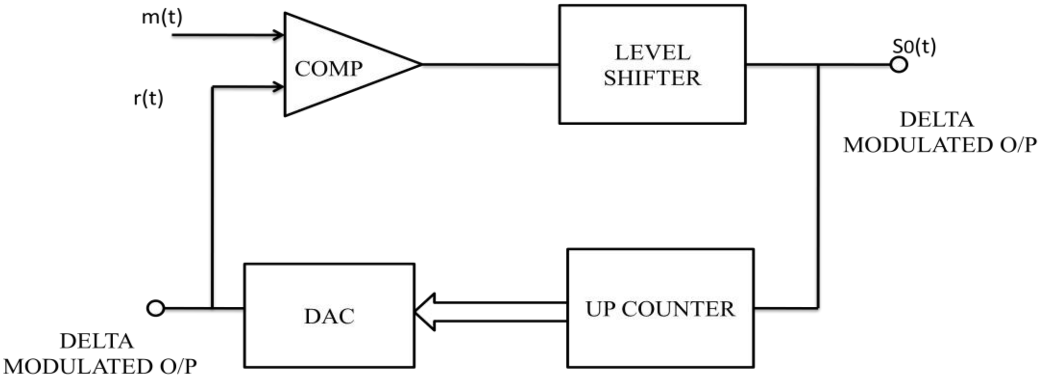


CIRCUIT DIAGRAM:



THEORY:

The block diagram of the delta modulation is also known as linear delta modulator. The signal m(t) is the analog input signal. While r(t) is a reconstructed signal which is same as the quantized input signal with 1 bit delay. The signal r (t) tries to follow the input signal m(t) with one bit period delay.



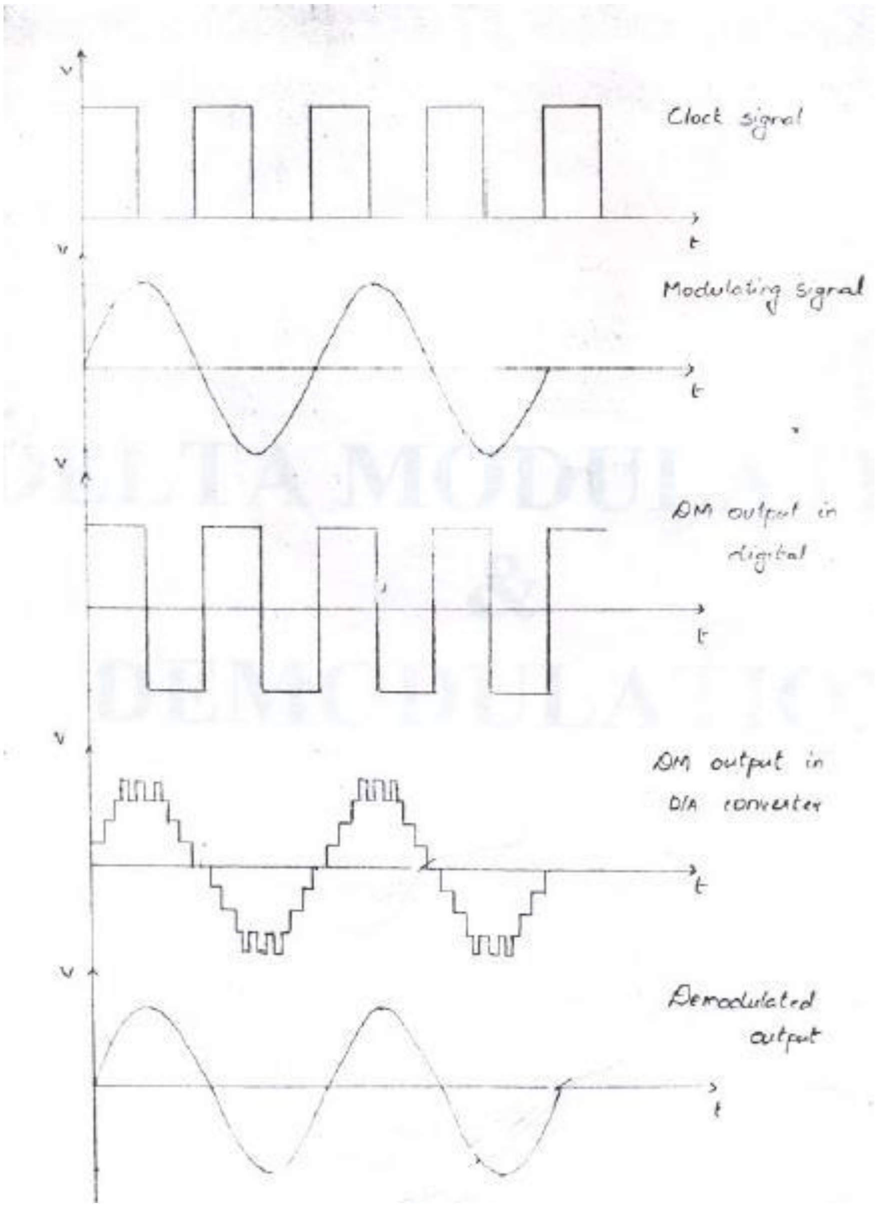
The process of encoding is as follows. The comparator compares the I m(t) > r(t) a logic 1 is generated at the output of the comparator, otherwise a logic 0 is generated. The value of logic 1 or logic 0 turned as current to generate So(t), the delta modulated output.

This output So(t) is fed to the 8 bit binary up/down counter to control it’s count direction. Logic 1 at the mode control input increases the count value by one and logic ‘0’ decrements the count value by one. All the 8 outputs of the counter are given to DAC to reconstruct the original signal. In essence the counter & decoder forms the delta modulator in the feedback loop of the comparator. Thus if the input signal is higher than the reconstructed signal the counter increments at each step so as to enable the DAC output to reach to the input signal values. Similarly if the input signal m(t) is lower than the reconstructed signal r(t), the counter decrements at each step, and the DAC output gets reduced to reach a value to that
input signal m(t) and r(t).

PROCEDURE:

1. The AC adaptor is connected to the mains and the other side to the experimental trainer.
2. Clock signal is applied to the delta modulator circuit.
3. Modulating signal is applied to the modulating input of the delta modulator and it is observed on one channel of cRO.
4. The delta modulator and it is observed on the other channel of CRO.
5. Now, the delta modulation output and clock signal is applied to the delta demodulator and the output is observed.
6. The demodulator output is observed with the without RC filter on CRO.

MODEL GRAPHS:



RESULT:

The delta modulated and demodulated waveforms are observed and studied.

APPLICATIONS:

1. Transimission of voice information where quality is not of primary importance.
2. Telecommunications.
3. Secure communications.
4. Delta Modulation was used by Satellite Business Systems or SBS for its voice ports to provide long distance phone service to large domestic corporations with a significant inter-corporation communications

VIVA QUESTIONS :

1. Define Delta modulation?
2. What do you mean by granular noise?
3. What do you mean by slope overload error?
4. What was the effect on output noise and distortion of an increase of step size and sampling rate?

EXPNO: 05

FREQUENCY SHIFT KEYING MODULATION AND DEMODULATION

AIM: To study the frequency shift keyed output and also to modulate the FSK output.

APPARATUS:

1. FSK modulator and de-modulator kits.
2. PCM modulator kit
3. CRO (32 MHz)
4. Patch chords.

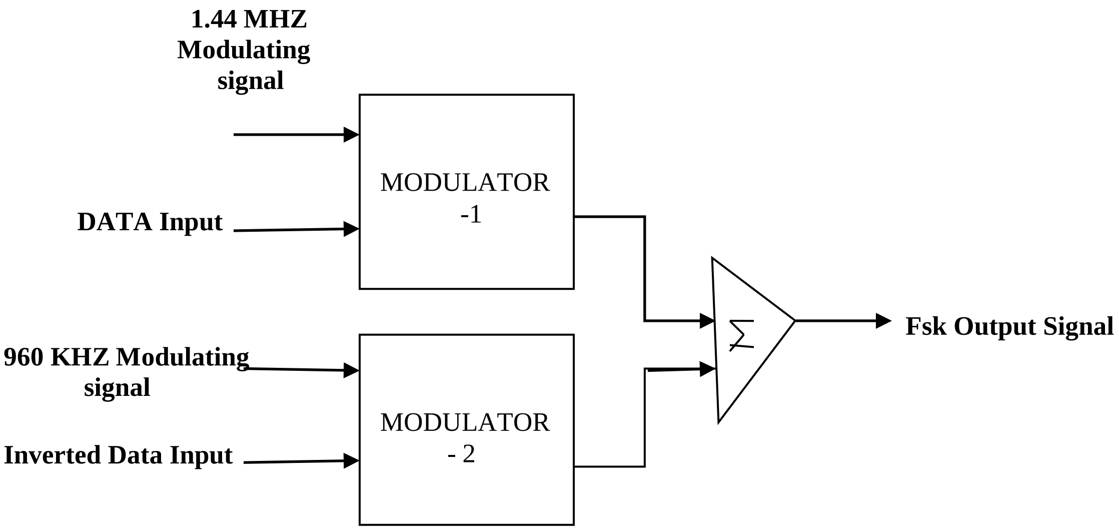
THEORY:

FSK is a system of frequency modulation, in it the nominal unmodulated carrier frequency corresponds to the mark condition, and a space is represented by a downward frequency shift. The mount was 850 Hz in the original wideband FSK system designed for HF radio. For transmission by line or broadband systems, the current shift is 60Hz. This is known as narrow band FSK, or Frequency modulated voice frequency telegraph (FM VFT). FSK is still often used for HF radio transmissions, with a frequency shift that is commonly 170Hz. As with other forms of FM, the main advantage of the wide band system is greater noise immunity, while the narrow band systems are used to conserve the allocated frequency spectrum. Note that FSK may be thought of as an FM system in which the carrier frequency is midway between the mark and space frequencies, and modulation is by a square wave. In practice, of course, only the fundamental frequency of the square wave is transmitted, and regeneration takes place in the receiver.

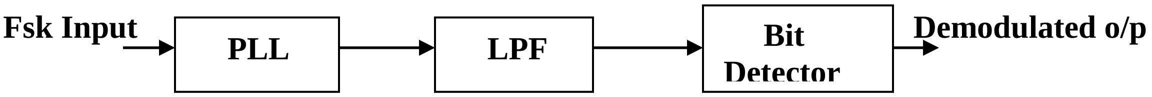
Basically a 555 IC is connected in Astable Multivibrator mode, generates a clock pulse of frequency determined by the values of RT and CT. This clock signal is given to a divided by 16 counter (74163 IC) which generates divided by 2,4,8 & 16 outputs of the input clock signal. In this system, divided by 2 & 8 outputs are taken as two carrier frequencies. So these are given to a FSK modulator constructed by using NAND gates. Divided by 16 output is given to a decade counter (IC 7490) which generates the modulating data signals. So depending on the level of the modulating data signal given to the FSK modulator, either divided by 2 or divided by 8 frequency outputs of the IC 74163 are transmitted to the output of the FSK modulator. In the demodulator section, the FSK output is given to a high - Q tuned filter which is tuned to any frequency either divided by 2 or 8 of the outputs of 74163 counter. So the filter passes one frequency and stops the other frequency. This filter is constructed with operational amplifier TL 084 IC. The passed frequency is given to a full wave rectifier Using operational amplifier TL084 IC. Which rectifies and its output is given to an envelope detector, which acts as a peak detector during pulse period. Then the output of the envelop' detector is given to a comparator, its output is equivalent to the modulating data given at the input of the FSK modulator.

BLOCKDIAGRAM:

MODULATOR:



DEMODULATOR:



CIRCUIT DIAGRAM:

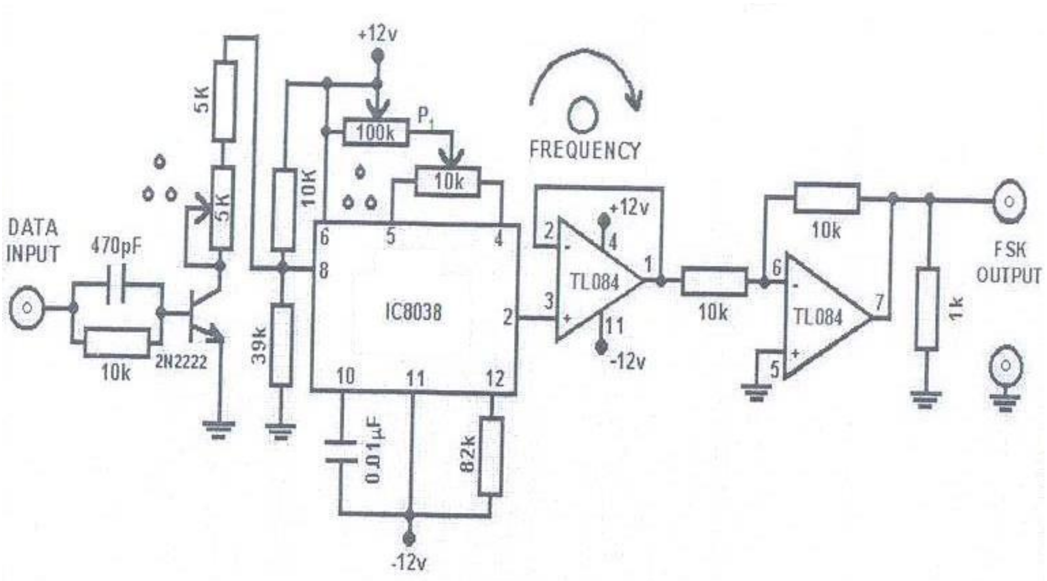


Fig. FSK Modulator

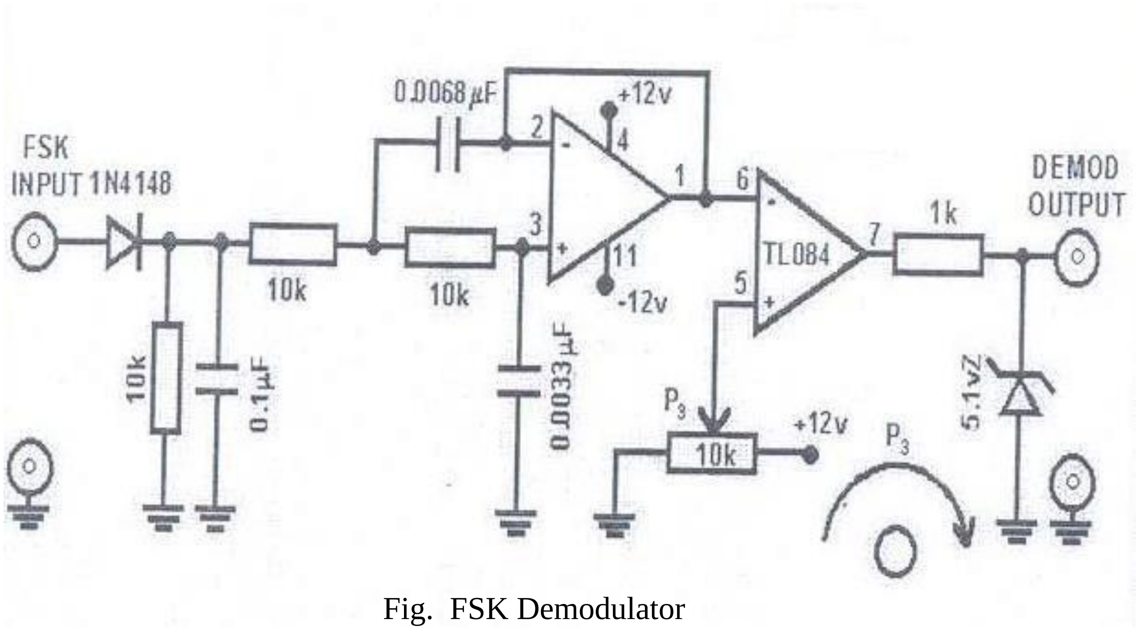
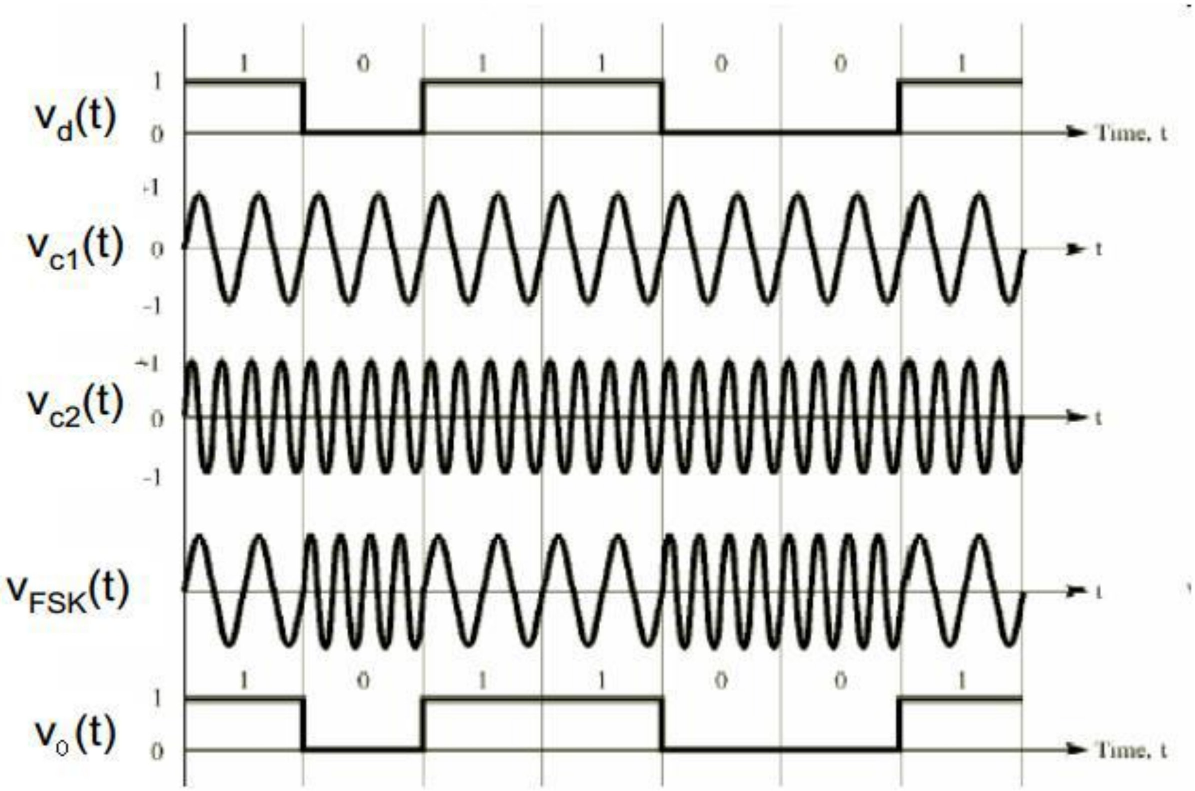


Fig. FSK Demodulator

PROCEDURE:

1. Connect the AC adaptor to the mains and the other side to the experimental trainer kit. Switch ON the power
2. Connect Data Input socket to ground
3. Connect the FSK output to the CH-1
4. Set the frequency adjust potentiometer so that the output is around 30kHz
5. Set the switches for required word pattern
6. Connect the Data input to ground , measure the frequency
7. Connect the data output to data input
8. Observe the data output on CH1 and FSK output on CH2
9. Observe that at each negative transition the carrier switches from high to low and every positive transition, the frequency switches from low to high.
10. Connect FSK output to FSK input of the demodulator.
11. Adjust P_3 to regenerate the data correctly
12. Compare the demodulated output to the data output which are identical in nature.

MODEL GRAPH:



APPLICATIONS:

1. AFSK is still widely used in amateur radio, as it allows data transmission through unmodified voice band equipment.
2. Radio control gear uses FSK.
3. AFSK is also used in the United States' Emergency Alert System to transmit warning information.
4. It is used at higher bit rates for Weather copy used on Weather radio by NOAA in the U.S.
5. Used in point to point military communication

RESULT:

The response of modulated and demodulated output of FSK system are observed and recorded.

VIVA QUESTIONS:

1. Define FSK?
2. What is the minimum modulation index that can be taken in FSK?
3. What is binary FSK? Why is it called so?

GRAPH

EXPNO: 06

PHASE SHIFT KEYING MODULATION AND DEMODULATION

AIM: To study the operation of phase shift keying modulation and demodulation.

APPARATUS:

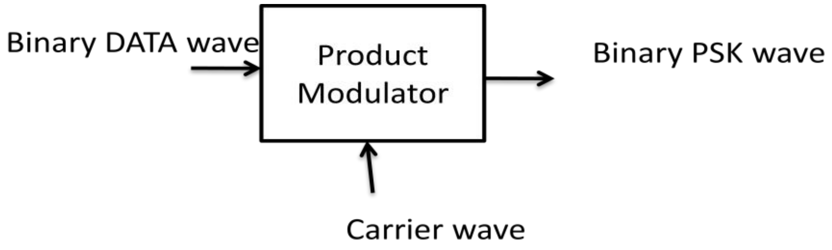
1. AC adapter,
2. PSK modulation and demodulation trainer kit.
3. CRO,
4. Probes,
5. Patch chords.

THEORY:

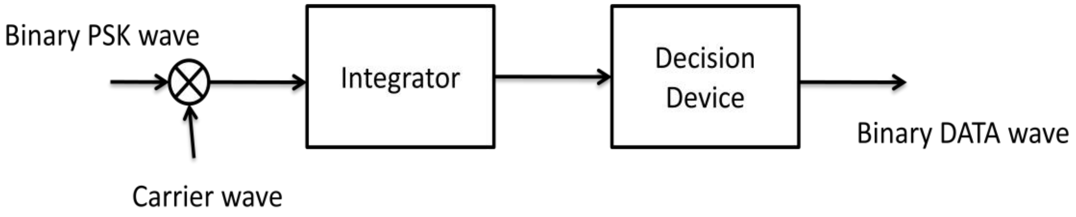
In this carrier Generator is designed around a wein bridge' oscillator using 741. The sinusoidal output has a frequency of around 10KHz. Square wave clock at the same Frequency is generated by using TL084 Op Amp in comparator Mode. A transistor inverter using BC 107 improves the clock shape with sharp, rising and falling edges. This square wave is used as a clock input to a decade counter (IC7490), which generates the modulating data outputs. IC CD4051 is an Analog multiplexer to which carrier is applied with and without 180^0 phase shift to the two multiplex inputs of the IC. Modulating data input is applied to its control input. Depending upon the level of the control signal, carrier signal applied with or without Phase shift is steered to the output. The 180^0 phase shift to the carrier signal created by an operational amplifier using 741 IC. During the demodulation, the PSK signal is converted into a +5 volts square wave signal using a transistor and is applied to one input of an EX-ORgate. To the second input of the gate, carrier signal is applied after conversion into a +5 volts signal. So the EX-OR gate output is equivalent to the modulating data signal.

BLOCK DIAGRAM:

MODULATOR:



DEMODULATOR:



CIRCUIT DIAGRAM:

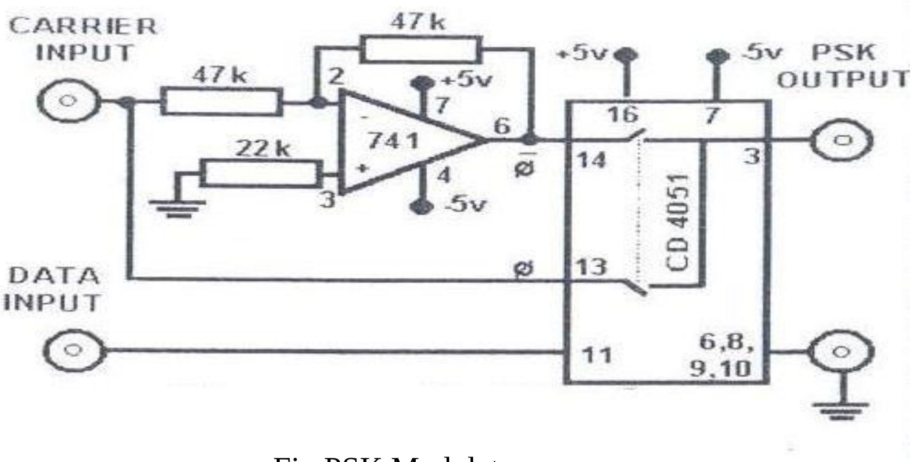


Fig PSK Modulator

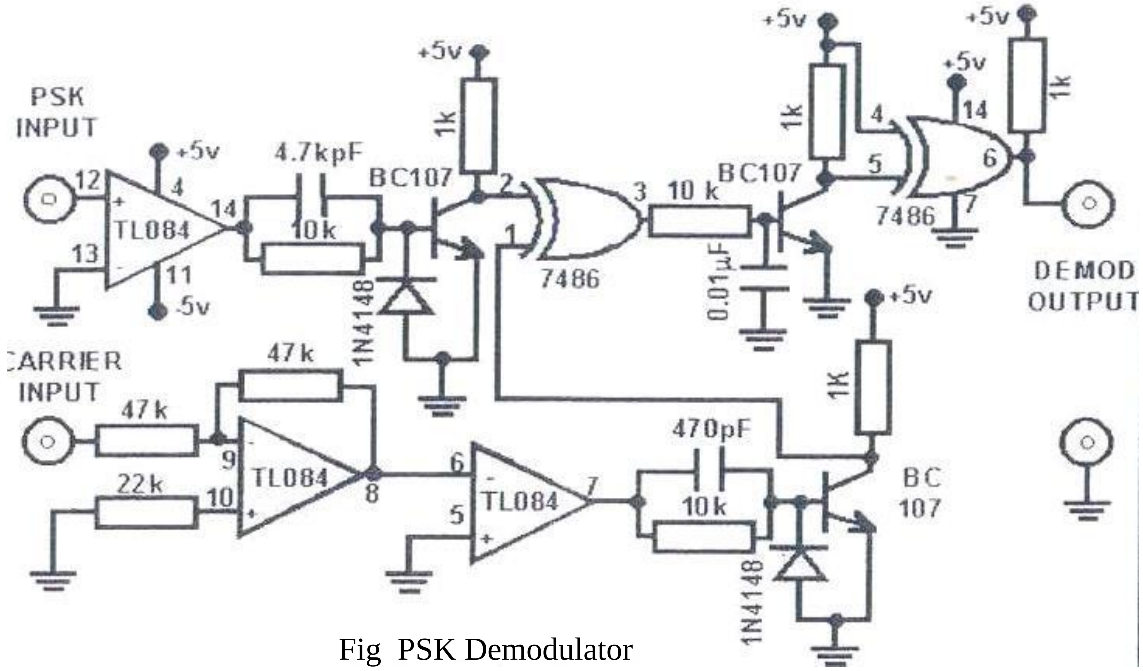


Fig PSK Demodulator

PROCEDURE:

1. AC adaptor is connected to the mains and the other side to the experimental trainer kit.
2. Carrier signal is applied to the input of the modulator.
3. The modulating data signal is applied to the modulator input and it is observed on one channel of the CRO.
4. The output of the PSK modulator is observed on the channel two of the CRO.
5. The PSK output is applied as input to the PSK demodulator and also the carrier input is applied to it.
6. The demodulator output is observed and it is identical to the modulating data signal applied to system.

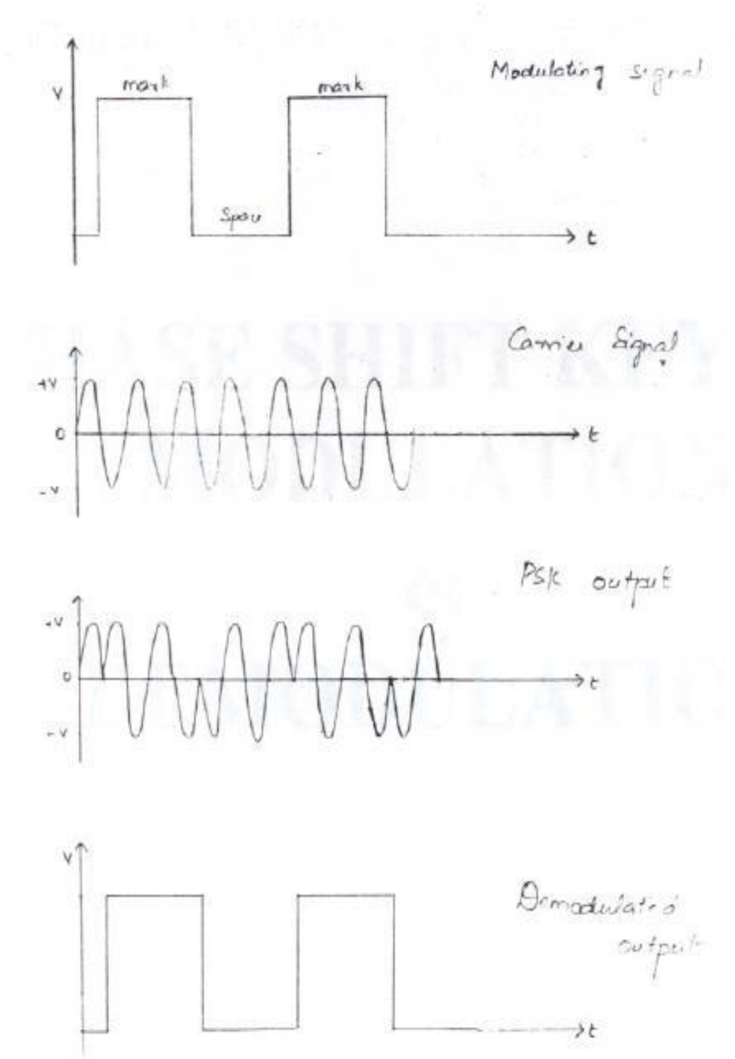
APPLICATIONS:

- The wireless LAN standard, IEEE 802.11b-1999 uses a variety of different PSKs depending on the data-rate required.
- Because of its simplicity BPSK is appropriate for low-cost passive transmitters, and is used in RFID standards such as ISO/IEC 14443 which has been adopted for biometric passports, credit cards such as American Express's ExpressPay, and many other applications

VIVA QUESTIONS

1. Define phse shift keying?
2. Draw the block diagram for generating psk?
3. Draw the constellation diagram for bpsk?
4. What is the bit error rate of bpsk?
5. For the binary data 1011 draw PSK modulated waveforms.
6. What is QPSK modulation?

MODEL GRAPHS:



RESULT:

The response of PSK modulator and demodulator for a given modulating data is observed.

GRAPH

EXPNO: 07

DIFFERENTIAL PHASE SHIFT KEYING MODULATION AND DEMODULATION

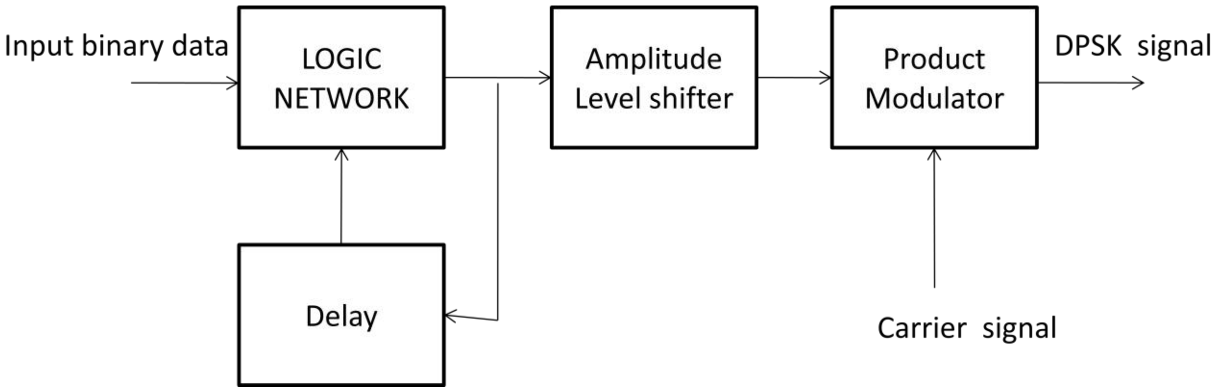
AIM: To study the various steps involved in generating the differential binary signal and Differential phase shift keying signal at the modulator and recovering the binary signal from the received DPSK signal.

APPARATUS:

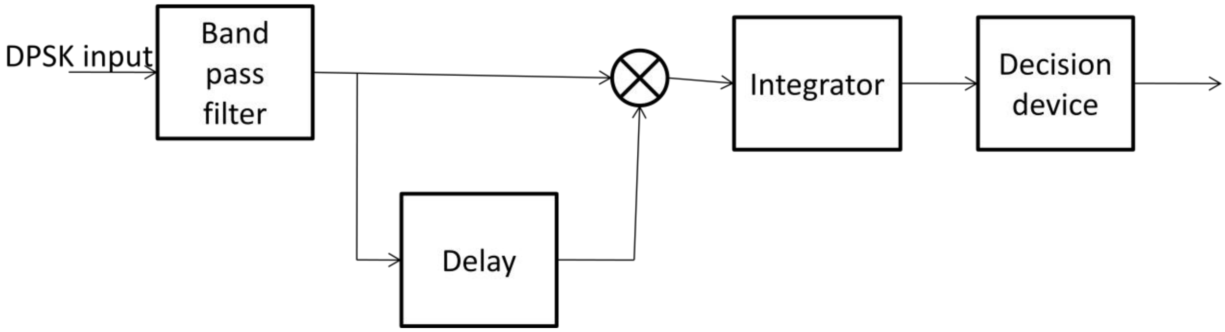
1. AC adapter
2. DPSK trainer kit
3. CRO
4. Probes
5. Patch cards

BLOCK DIAGRAM:

MODULATOR:



DEMODULATOR:



CIRCUIT DIAGRAM:

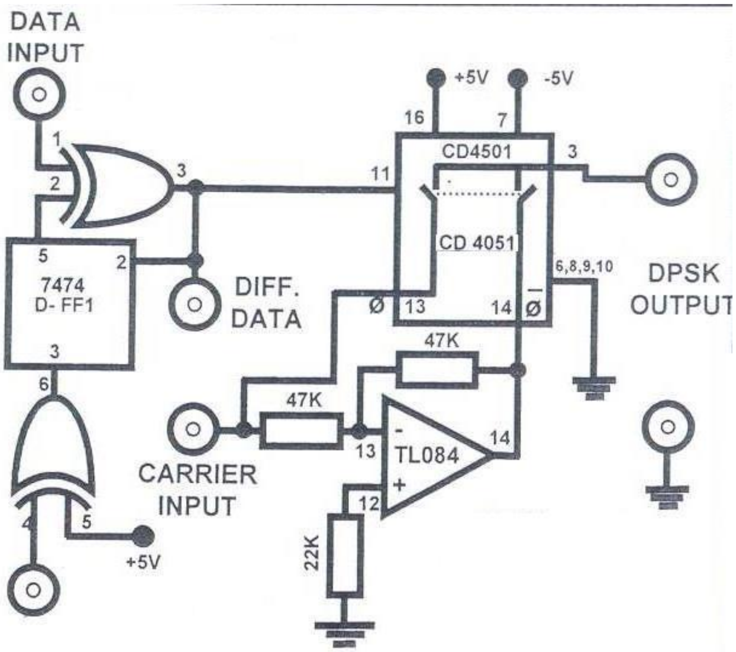


Fig DPSK modulator

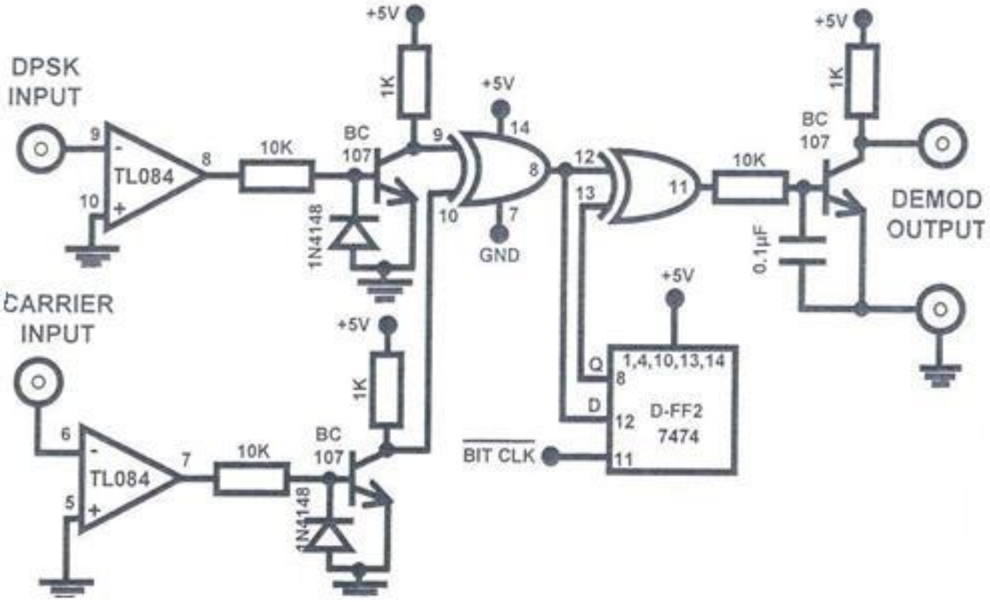


Fig DPSK Demodulator

THEORY:

Digital communications became important with the expansion of the use of computers and data processing, and have continued to develop into a major industry providing the interconnection of computer peripherals and transmission of data between distant sites. Phase shift keying (PSK) is a relatively new system, in which the carrier may be phase shifted by + 90 degrees for a mark, and by -90 degrees for a space. PSK has a number of similarities to FSK in many aspects, as in FSK frequency of the carrier is shifted according to the modulating square wave shows the circuit diagram of Differential phase Shift Key Modulation & Demodulation. In this IC 8038 is a basic waveform generator, which generates Sine, Square, and Triangle waveforms. The sine wave generated by this 8038 IC is used as carrier signal to the system. The Square Wave generated by 80381C is at +/- 12V level. So this is converted into a +5V signal with the help of a transistor and diode as shown in fig-1. This Square wave is used as a clock input to a decade counter (IC 7490), which generates the modulating data outputs.

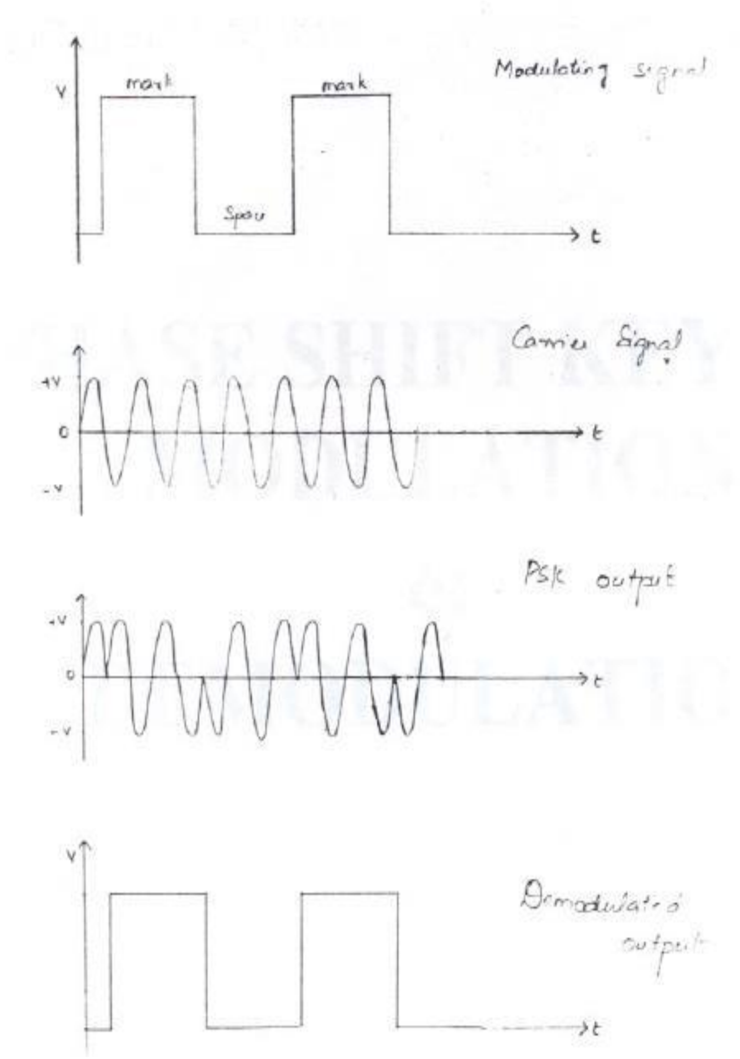
MODULATION: The Differential signal to the modulating signal is generated using an Exclusive -OR gate and a 1-bit delay circuit (It is shown in fig-1). CD 4051 is an analog multiplexer to which carrier is applied with and without 1800 degrees Phase shift (created by using an operational amplifier connected in inverting amplifier mode) to the two inputs of the IC741. Differential signal generated by Ex-OR gate (IC7486) is given to the multiplexer's control signal input. Depending upon the level of the control signal, carrier signal applied with or without phase shift is steered to the output. Using a D-flip-flop (IC747) creates 11bit delay generation of differential signal to the input.

DEMODULATION: During the demodulation, the DPSK signal is converted into a +5V Square Wave signal using a transistor and is applied to one input of an EX-OR gate. To the second input of the gate, carrier signal is applied after conversion into a +5V signal. So the EX-OR gate output is equivalent to the differential signal of the modulating data. This differential data is applied to one input of an Exclusive -OR gate and to the second input. After 1-bit delay the same signal is given. So the output of this Ex-OR gate is modulating signal.

PROCEDURE:

1. AC adaptor is connected to the mains and the other side to the experimental trainer kit.
2. The carrier signal and data generator’s outputs are observed on CRO.
3. The carrier signal is applied to the carrier input of the DPSK modulator and bit clock pulse is also applied, and data input from data generator is also applied to the data input terminals of DPSK modulator.
4. The output of the DPSK modulator output is observed on one channel of CRO, with respect to the data generator signal on other channel of CRO.
5. Now, the output of DPSK modulator and also the bit clock is applied as input to the demodulator and also carrier input and also applied as input to the DPSK demodulator circuit.
6. The demodulator output is observed with respect to the generator signal which is the modulating signal.

MODEL GRAPHS:



APPLICATIONS:

- DPSK is appropriate for low-cost passive transmitters, and is used in RFID standards such as ISO/IEC 14443 which has been adopted for biometric passports, credit cards such as American Express's ExpressPay, and many other applications.

RESULT:

The response of DPSK modulator and demodulator for a given modulating data is observed.

VIVA QUESTIONS:

1. Define DPSK?
2. What is the principle behind PSK?
3. What do you mean by bit error rate?
4. Draw the DPSK output waveform for the message following 10110

GRAPH

EXPT NO:8
COMPANDING

AIM: Implementation of Mu-law companding and expansion of signal

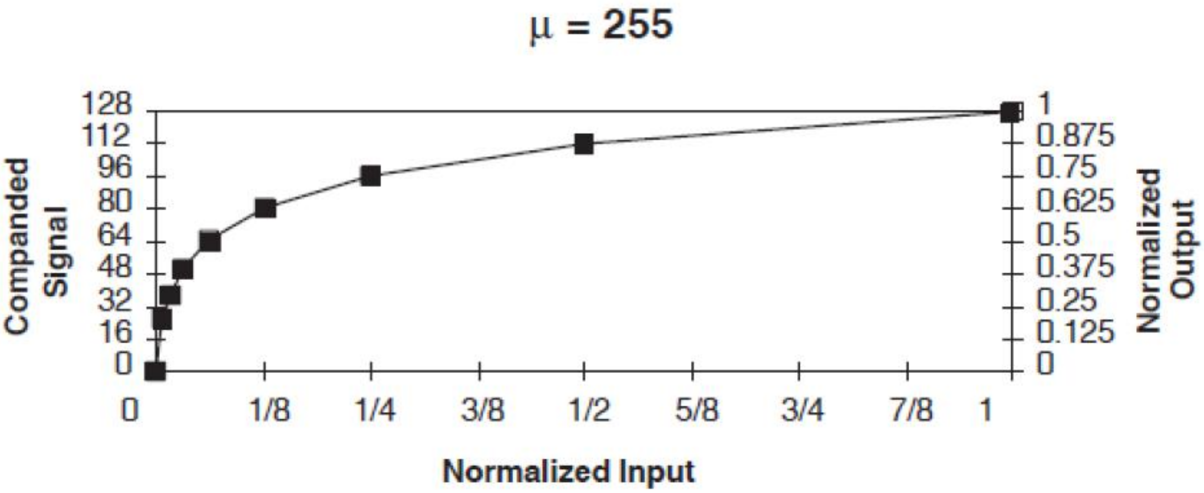
APPARATUS:

1. DSP Processor kit
2. Computer
3. Power supply
4. Code Composer Studio

THEORY:

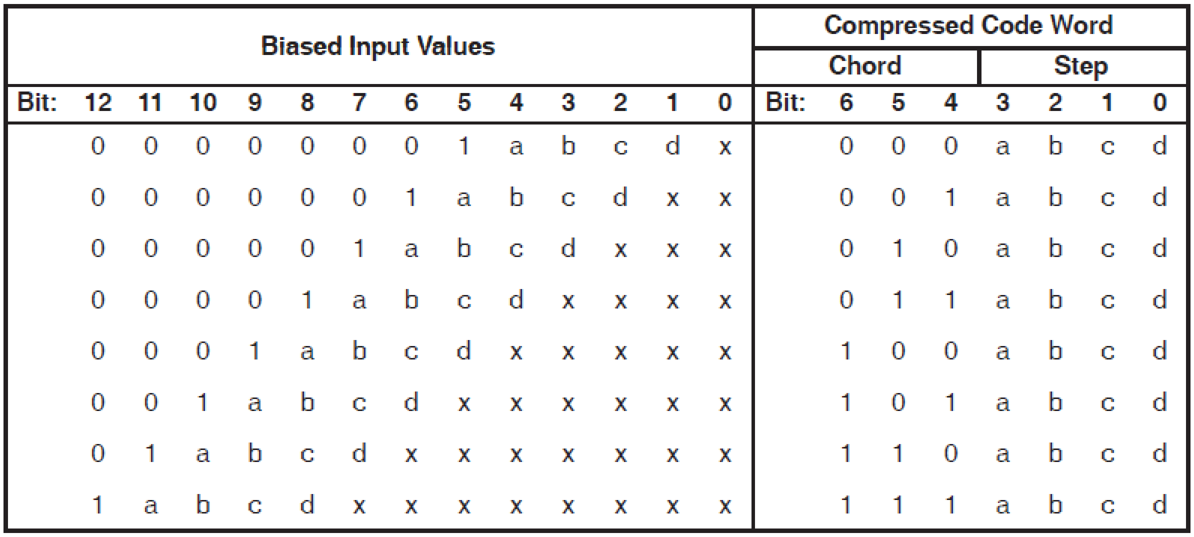
The United States and Japan support m-law companding. Limiting sample values to 13 magnitude bits, m-law compression can be defined mathematically by the following continuous equation: F(x) = sgn(x) ln(1 + m |x|) / ln (1 + m) - formula(1)

Where m is the compression parameter (m=25510 for the U.S. and Japan), and x is the normalized integer to be compressed. Following Figure illustrates a piecewise linear approximation to this compression equation.



The least significant bits of large amplitude values are discarded during compression. The number of deleted bits is encoded into a field of the encoded word, called the segment. Each segment of this piece-wise linear approximation is equally divided into quantization levels. The segment size between adjacent codeword's is doubled for each succeeding segment. Moreover, the most significant bit of the codeword contains the sign of the original integer. An 8-bit m-255 codeword is comprised of one sign bit, concatenated with a 3-bit segment, concatenated with a 4-bit quantization value. Prior to transmission, all the bits are inverted so a positive value will have a sign bit of "1" (one). Prior to segment determination, the sign of the original integer is set aside and a bias of 3310 is added to the absolute value (magnitude) of the integer. The bias limits the maximum allowable input to 815910, and reduces the minimum step size to 2/815910. The bias simplifies the calculation by making the endpoints of each segment powers of two. Locating the segment is determined by detecting the most significant “1” of the biased magnitude, while the quantization value is comprised of the four bits following it. The translation from linear to m-law compression is illust rated in following Table of the compressed codeword, bits 0– 3 represent the quantization and bits 4–6 represent the segment. The sign of the compressed codeword is left out for simplicity.

µ-Law Binary Encoding Table



Biased Input Values														Compressed Code Word							
														Chord				Step			
Bit:	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit:	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	1	a	b	c	d	x		0	0	0	a	b	c	d
	0	0	0	0	0	0	1	a	b	c	d	x	x		0	0	1	a	b	c	d
	0	0	0	0	0	1	a	b	c	d	x	x	x		0	1	0	a	b	c	d
	0	0	0	0	1	a	b	c	d	x	x	x	x		0	1	1	a	b	c	d
	0	0	0	1	a	b	c	d	x	x	x	x	x		1	0	0	a	b	c	d
	0	0	1	a	b	c	d	x	x	x	x	x	x		1	0	1	a	b	c	d
	0	1	a	b	c	d	x	x	x	x	x	x	x		1	1	0	a	b	c	d
	1	a	b	c	d	x	x	x	x	x	x	x	x		1	1	1	a	b	c	d

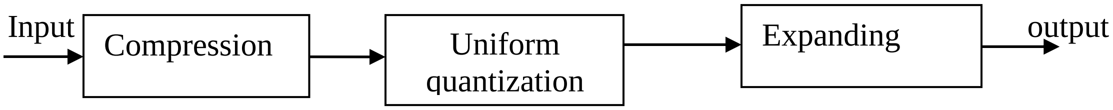
The entire m-law codeword is inverted prior to transmission. The inversion is performed because low amplitude signals occur more frequently than large amplitude signals. Consequently, inverting the bits increases the positive pulse density on the transmission line, which improves system performance. m-law expansion can be defined mathematically by the following continuous equation: F -1(y) = sgn(y) (1 / m) [(1+ m)|y| – 1] - formula(2)

Prior to expansion, the m-law codeword is inverted again during expansion, the least significant bits are discarded but are approximated by the median interval, to reduce the loss in accuracy. For example, if five of the least significant bits of the original integer were discarded during compression, 100002 will approximate them during expansion. The translation from m-law to linear expansion is illustrated in following table. Again, the sign bits are left out for simplicity. After decoding the m-law codeword, the bias is removed and the sign bit is applied to obtain the final linear value**.**

µ-Law Binary Decoding Table

Compressed Code Word								Biased Input Values													
Chord				Step																	
Bit:	6	5	4	3	2	1	0	Bit:	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	a	b	c	d		0	0	0	0	0	0	0	1	a	b	c	d	1
	0	0	1	a	b	c	d		0	0	0	0	0	0	1	a	b	c	d	1	0
	0	1	0	a	b	c	d		0	0	0	0	0	1	a	b	c	d	1	0	0
	0	1	1	a	b	c	d		0	0	0	0	1	a	b	c	d	1	0	0	0
	1	0	0	a	b	c	d		0	0	0	1	a	b	c	d	1	0	0	0	0
	1	0	1	a	b	c	d		0	0	1	a	b	c	d	1	0	0	0	0	0
	1	1	0	a	b	c	d		0	1	a	b	c	d	1	0	0	0	0	0	0
	1	1	1	a	b	c	d		1	a	b	c	d	1	0	0	0	0	0	0	0

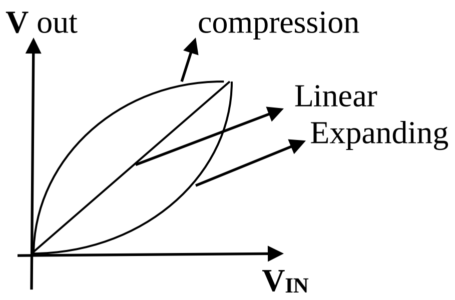
BLOCK DIAGRAM:

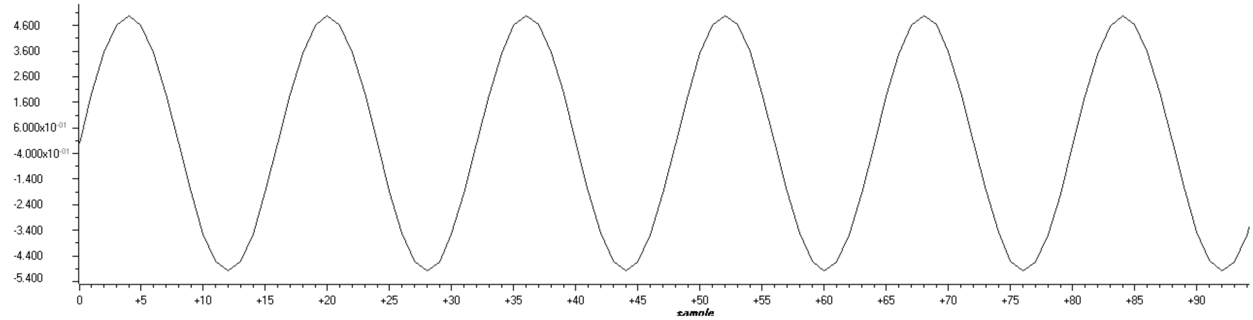


PROCEDURE:

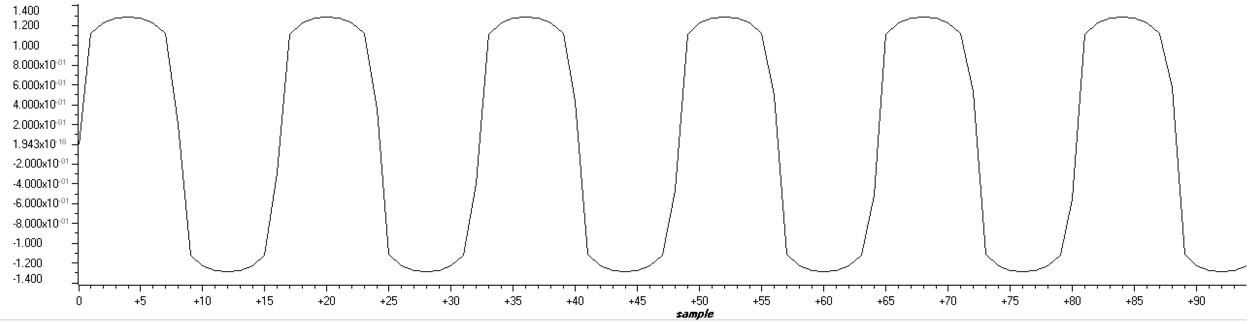
1. Open Code Composer Studio, make sure the DSP kit is turned on.
2. Load program following location.
3. PATH: DSP320PROGRAMS \ Mu_law
4. Then run program from debug.

MODEL GRAPHS:

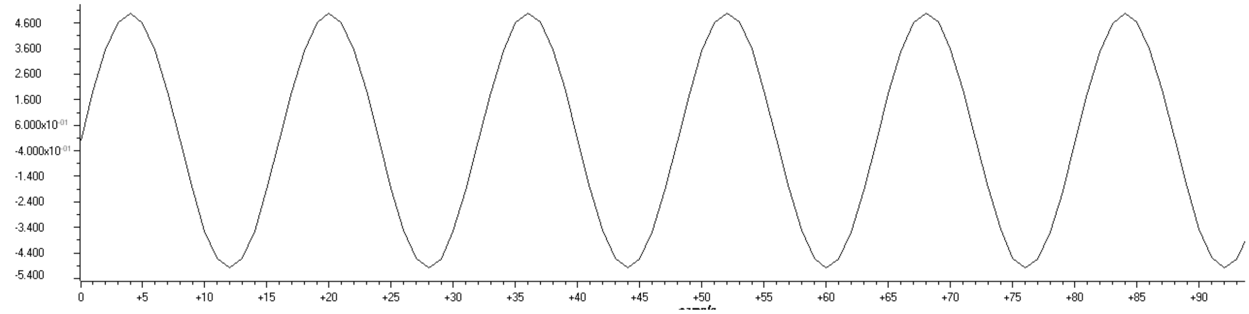




INPUT



COMPRESSED OUPUT



EXPANDED OUTPUT

PROGRAM:

```
#include<stdio.h>
#include<math.h>
#define Mu 255

float original[100], x[100],y[100], com[100], ex[100];

void main()
{
            int i;
            float amp;
            int sgn[100], sgn_y[100];
            FILE *fp;
```

```
fp=fopen("D:\\Mu-law_companding.txt","wr");
fprintf(fp,"\nINPUT SINE\tCOMPANDING\tOUTPUT SINE\n");
printf("Enter amplitude level = ");
scanf("%f",&amp);
for(i=0; i<100; i++)
{
    x[i]=amp*sin(2*3.14*i*1000/16000);
    original[i]=x[i];
    if(x[i] > 0.0)
    sgn[i] = 1;
    else
    {
            if(x[i] < 0.0)
            {
            sgn[i] = -1;
            x[i]=x[i]*(-1);
            }
            else
            if(x[i] == 0.0)
            sgn[i] = 0;
    }

}

for(i=0; i<100; i++)
{
    com[i] = (sgn[i]*log(1+(Mu*x[i])))/log(Mu+1);
    y[i] = com[i];
    if(com[i] > 0.0)
    sgn_y[i] = 1;
    else
    {
            if(com[i] < 0.0)
            {
            sgn_y[i] = -1;
            y[i]=com[i]*(-1);
            }
            else
            if(com[i] == 0.0)
            sgn_y[i] = 0;
    }
}
for(i=0; i<100; i++)
{
    ex[i] = (sgn_y[i]*(pow((1+Mu),y[i])-1)/Mu);
}
```

```
            for(i=0; i<100; i++)
            {
                fprintf(fp,"%f\t%f\t%f\n", original[i],com[i],ex[i]);
            }


}
```

APPLICATIONS:

1. Companding is used in digital telephony systems, compressing before input to an analog-to-digital converter, and then expanding after a digital-to-analog converter.
2. This method is also used in digital file formats for better signal-to-noise ratio (SNR) at lower bit rates.
3. Compandors are used in concert audio systems and in some noise reduction schemes such as dbx and Dolby NR .
4. Another application of companding involves professional wireless microphones, which have a larger dynamic range than is possible through radio transmission.

RESULT: Hence we studied companding characteristics.

VIVA QUESTIONS:

1. Define Companding?
2. What is the principle of companding?
3. What is the main advantage of companding?
4. Explain µ-law and A-law companding characteristics?

GRAPH

EXPT NO: 9

SOURCE ENCODER AND DECODER

AIM : Image compression using Huffman coding.

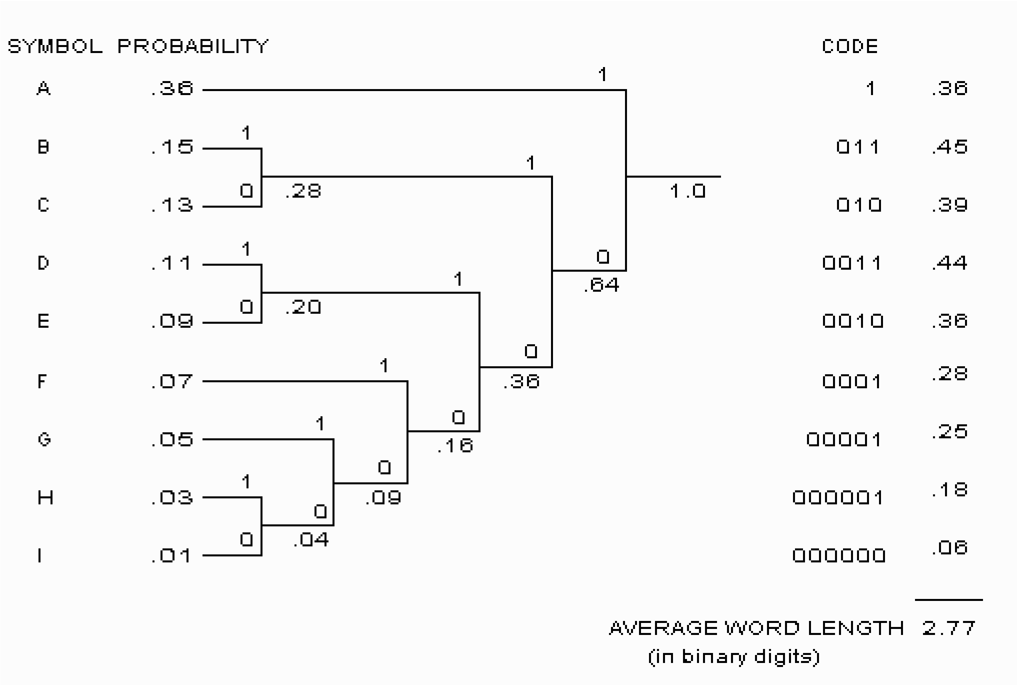
APPARATUS:

1. DSP Processor kit
2. Computer
3. Power supply
4. Code Composer Studio

THEORY: Huffman coding

Huffman encoding, an algorithm for the lossless compression of files based on the frequency of occurrence of a symbol in the file that is being compressed. The Huffman algorithm is based on statistical coding, which means that the probability of a symbol has a direct bearing on the length of its representation. The more probable the occurrence of a symbol is, the shorter will be its bit-size representation. In any file, certain characters are used more than others. Using binary representation, the number of bits required to represent each character depends upon the number of characters that have to be represented. Using one bit we can represent two characters, i.e., 0 represents the first character and 1 represents the second character. Using two bits we can represent four characters, and so on.

Unlike ASCII code, which is a fixed-length code using seven bits per character, Huffman compression is a variable-length coding system that assigns smaller codes for more frequently used characters and larger codes for less frequently used characters in order to reduce the size of files being compressed and transferred. The basic idea in Huffman coding is to assign short codeword to those input blocks with high probabilities and long codeword to those with low probabilities. This concept is similar to that of the Morse code. A Huffman code is designed by merging together the two *least probable* characters, and repeating this process until there is only one character remaining. A code tree is thus generated and the Huffman code is obtained from the labeling of the code tree. An example of how this is done is shown below.

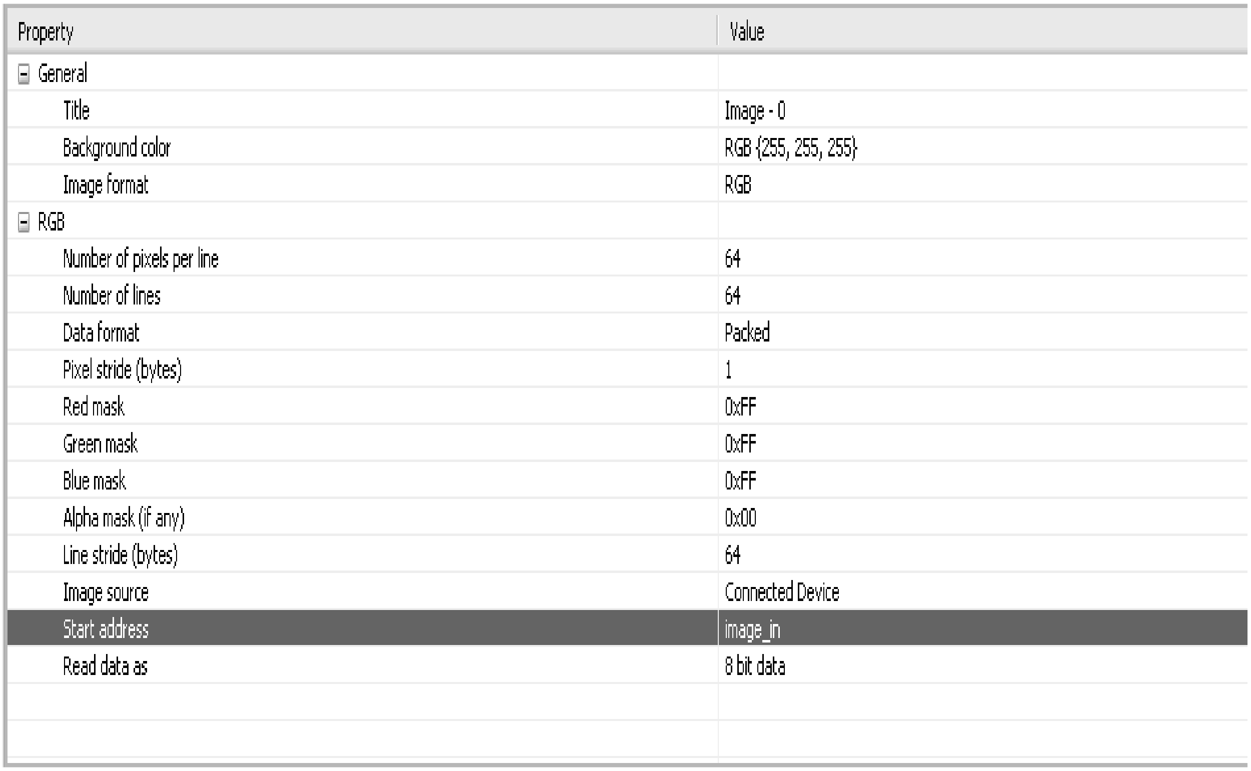


PROCEDURE:

1. Open Code Composer Studio, make sure the DSP kit is turned on.
2. Import program using ‘Project_import Existing ccs Eclipe project. Which is saved in DVD at following location **PATH: DSP320_PROGRAMS\ HUFFMAN_CODING**
3. Then debug and run the program

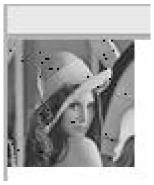
OUTPUT:

1. Here we have generated pixel values from MATLAB for image of Lena
2. Then we calculate frequency of same pixel value & asine new values from 0 to 256 as per occurance.
3. To view Original image Select **Tool →Image Analyzer**

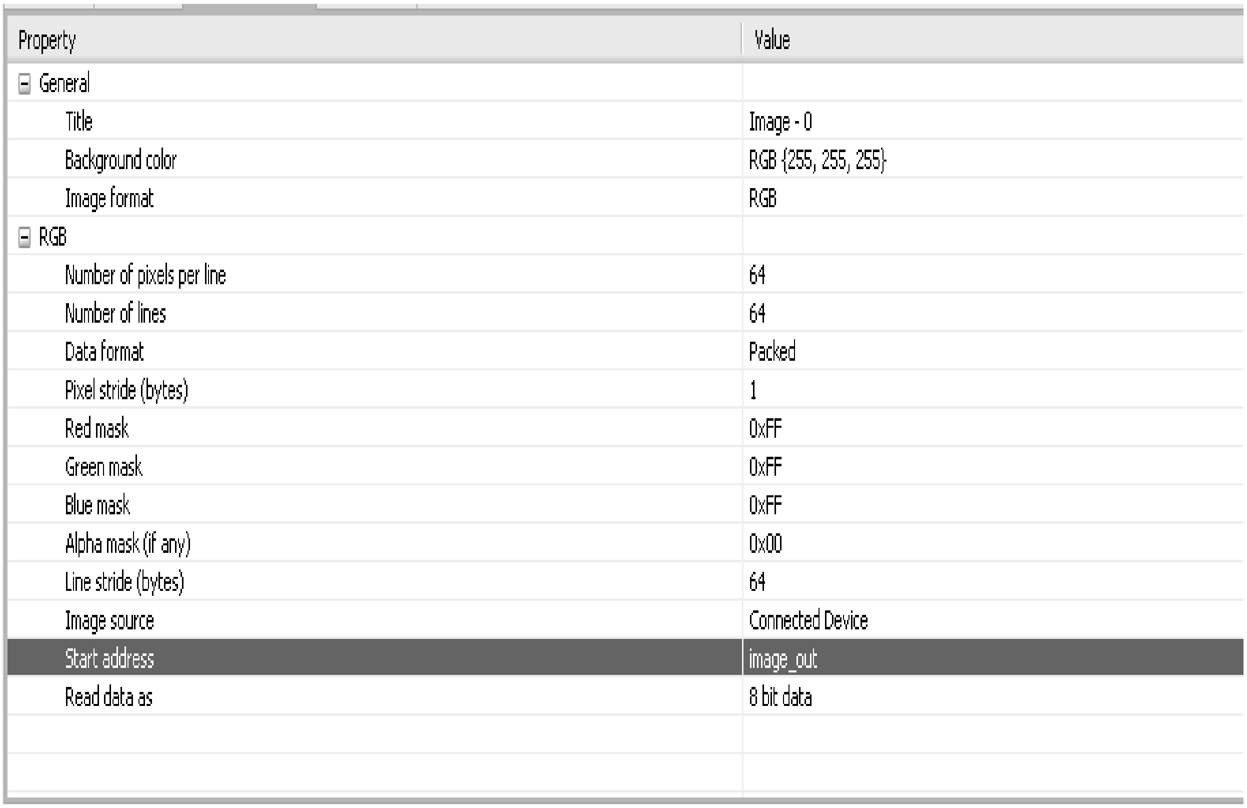


Property	Value
⊟ General	
Title	Image - 0
Background color	RGB {255, 255, 255}
Image format	RGB
⊟ RGB	
Number of pixels per line	64
Number of lines	64
Data format	Packed
Pixel stride (bytes)	1
Red mask	0xFF
Green mask	0xFF
Blue mask	0xFF
Alpha mask (if any)	0x00
Line stride (bytes)	64
Image source	Connected Device
Start address	image_in
Read data as	8 bit data

After completing property go on image → right click → refersh In image



To view compressed image
Select **Tool →Image Analyzer**



Property	Value
⊟ General	
Title	Image - 0
Background color	RGB {255, 255, 255}
Image format	RGB
⊟ RGB	
Number of pixels per line	64
Number of lines	64
Data format	Packed
Pixel stride (bytes)	1
Red mask	0xFF
Green mask	0xFF
Blue mask	0xFF
Alpha mask (if any)	0x00
Line stride (bytes)	64
Image source	Connected Device
Start address	image_out
Read data as	8 bit data

After completing property go on image →right click →refersh

RESULT:

Thus we have studied image compression by using Huffman code.

VIVA QUESTIONS:

1. What are the types of source coding techniques?
2. What are the advantages of huffman coding technique?
3. How to calculate efficiency of huffman code?

EXPT NO: 10

LINEAR BLOCK CODE-ENCODER AND DECODER

AIM:

To study of error detection and correction codes using hamming code

APPARATUS:

Experimental kits DCL-03 and DCL-04
Connecting cards
Power supply
20MHz Dual Trace Oscilloscope

THEORY:

The linear block codes in detail and obtain various mathematical expressions related to these codes. We consider an (n,k) linear block code in which k number of bits are identical to message sequence which is to be transmitted.

The remaining (n-k) number of bits is called as the generalized parity check bits or simply parity bits, these parity bits are computed from the message bits. According to prescribed encoding rule which devices the mathematical structure of code. A code word consists of k message bits which are denoted by m0,m1,m2…………m(k-1) and (n-k) parity bits denoted by c0,c1,c2…..c(n-k) these sequence of message bits are applied to a linear block encoder to produce an n-bit code word. The elements of this code word are x0,x1,x2………x(n-1). A block code generator the parity vector required to be added to message bits to generate the code words.

PROCEDURE:

Hamming code parity

1. Refer to the block diagram & carry out the following connections.
2. Connect power supply in proper polarity to the kits DCL-03 and DCL -04 and switch it on.
3. Connect DC input signal DC1 to the input CH0 and CH1 of the sample and hold logic.
4. Set the speed selection switch SW1 to FAST mode.
5. select parity selection switch to HAMMING mode on both the kit DCL-03 and DCL-04 as shown in switch setting diagram(fig A)
6. Connect TXDATA, TXCLK and TXSYNC of the transmitter section DCL-03 to the corresponding RXDATA, RXCLK and RXSYNC of the receiver section DCL-04.
7. Vary the amplitude of input DC signal from 0v to 4.96v and observe the variation on LED on the transmitter and receiver as mention below.
8. create a single bit fault in any one of the 4-MSB data bit by putting switch in below position of SF1 and observe the status of PARITY ERROR.

BLOCK DIAGRAM:

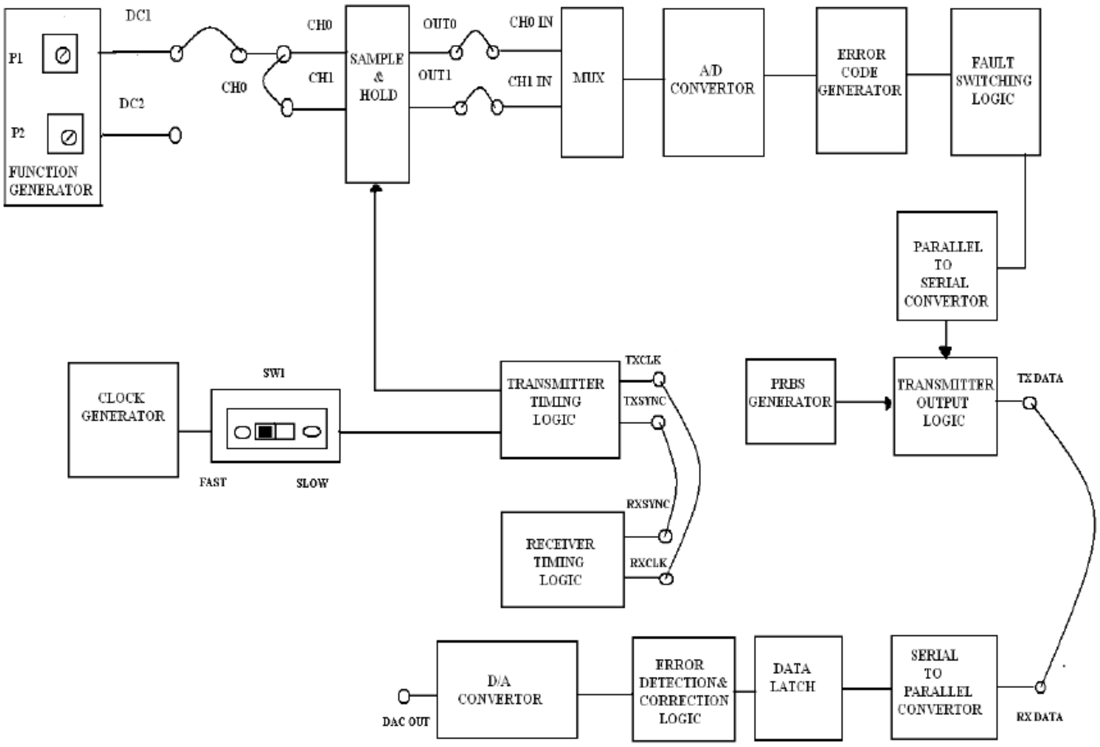
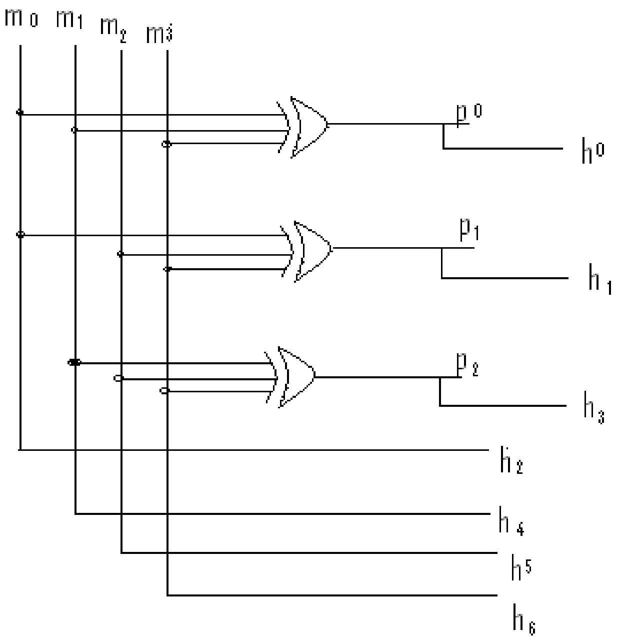


FIG . 1

THEORETICAL:



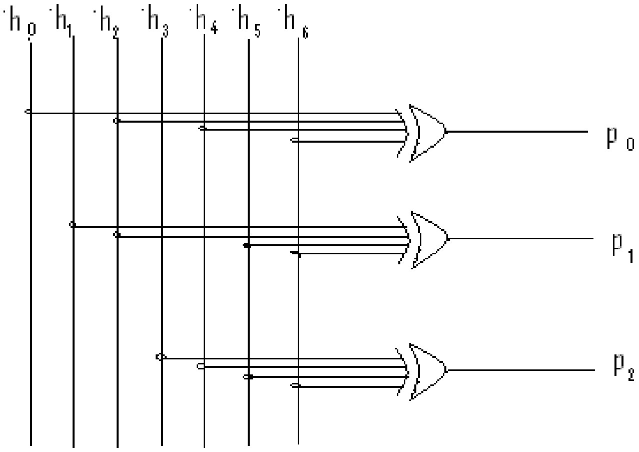
Hamming code circuit diagram

$P0=m0 \oplus m1 \oplus m3$

$P1=m0 \oplus m2 \oplus m3$

$P2=m1 \oplus m2 \oplus m3$

m0=h2
m1=h4
m2=h5
m3=h6



ERROR DETECTION

$P0=m1 \oplus m2 \oplus m4 \oplus m6$

$P1=m1 \oplus m2 \oplus m5 \oplus \mathbf{m6}$

$P2=m3 \oplus m4 \oplus m5 \oplus \mathbf{m6}$

Name	Value	
m[3:0]	0101	0101
h[6:0]	0101101	0101101
p[2:0]	101	101

2^p >= n+p+1

2^3 >= 4=3=1

p=3,n=4

111	110	101	100	011	010	001
7	6	5	4	3	2	1
m3	m2	m1	p2	m0	p1	p2
0	1	0	1	1	0	1
h6	h5	h4	h3	h2	h1	h0

$P0=m0 \oplus m1 \oplus m3 =1$

$P1=m0 \oplus m2 \oplus m3 =0$

$P2=m1 \oplus m2 \oplus m3 =1$

H = 0101101

HAMMING CODE PROGRAM:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity hamingcodedecoder is
Port ( m : in  STD_LOGIC_VECTOR (3 downto 0);
h : out  STD_LOGIC_VECTOR (6 downto 0));
end hamingcodedecoder;
architecture Behavioral of hamingcodedecoder is
signal p: STD_LOGIC_VECTOR (2 downto 0);
begin
p(0) <= m(0) xor m(1) xor m(3);
p(1) <= m(0) xor m(2) xor m(3);
p(2) <= m(1) xor m(2) xor m(3);
h(0) <= p(0);
h(1) <= p(1);
h(2) <= m(0);
h(3) <= p(2);
```

```
h(4) <= m(1);
h(5) <= m(2);
h(6) <= m(3);
end Behavioral;
```

ERROR DETECTION:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity errordetection is
Port ( h : in  STD_LOGIC_VECTOR (6 downto 0);
errorlocation : out  STD_LOGIC_VECTOR (2 downto 0));
end errordetection;
architecture Behavioral of errordetection is
begin
process(h)
begin
if((h(0) xor h(2)) xor (h(4) xor h(6))) = '1' then
errorlocation(0)<='1';
else
errorlocation(0)<='0';
end if;
if((h(1) xor h(2)) xor (h(5) xor h(6))) = '1' then
errorlocation(1)<='1';
else
errorlocation(1)<='0';
end if;
if((h(3) xor h(4)) xor (h(5) xor h(6))) = '1' then
errorlocation(2)<='1';
else
errorlocation(2)<='0';
end if;
end process;
end Behavioral;
```

PRACTICAL VALUES

Hamming code = 0100101

Error position = 100

THEORITICAL VALUES

h6	h5	h4	h3	h2	h1	h0
0	1	0	0	1	0	1

P1-> 1 ⊕ 3 ⊕ 5 ⊕ 7 -> 0 ⊕ 0 ⊕1 ⊕1 ->0
P1-> 2 ⊕ 3 ⊕ 6 ⊕ 7 -> 0 ⊕ 1 ⊕1 ⊕0 ->0
P1-> 1 ⊕ 5 ⊕ 6 ⊕ 7 -> 0 ⊕ 1 ⊕0 ⊕0 ->1

RESULT:

Hence we studied the hamming code generation and error detection.

VIVA QUESTIONS:

1. What do you mean by Linear block codes?
2. How are these codes useful in error detection?
3. What do you mean by hamming distance?
4. Define parity check.

EXPTNO: 11

BINARY CYCLIC CODE – ENCODER & DECODER

AIM:

To study the cyclic redundancy code encoding & decoding

APPARATUS:

1 ADCL-08 board and its power supply

THEORY:

A cyclic redundancy check (CRC) is an error-detecting code commonly used in digital networks and storage devices to detect accidental changes to raw data. Blocks of data entering these systems get a short check value attached, based on the remainder of a polynomial division of their contents; on retrieval the calculation is repeated, and corrective action can be taken against presumed data corruption if the check values do not match.

CRCs are so called because the check (data verification) value is a redundancy (it adds no information to the message) and the algorithm is based on cyclic codes. CRCs are popular because they are simple to implement in binary hardware, easy to analyze mathematically, and particularly good at detecting common errors caused by noise in transmission channels. Because the check value has a fixed length, the function that generates it is occasionally used as a hash function. The CRC was invented by W. Wesley Peterson in 1961; the 32-bit polynomial used in the CRC function of Ethernet and many other standards is the work of several researchers and were published during 1975. Typically an n-bit CRC applied to a data block of arbitrary length will detect any single error burst not longer than n bits and will detect a fraction 1−2−n of all longer error bursts.

- ❖ To compute an n-bit binary CRC, line the bits representing the input in a row, and position the (n+1)-bit pattern representing the CRC's divisor

(Called a "polynomial") underneath the left-hand end of the row.

Start with the message to be encoded:

11010011101100

This is first padded with zeroes corresponding to the bit length n of the CRC. Here is the first calculation for computing a 3-bit CRC:

```
11010011101100 000  ← input right padded by 3 bits
1011← divisor (4 bits) = $x^3+x+1$
------------------
01100011101100 000 ← result
```

If the input bit above the leftmost divisor bit is 0, do nothing. If the input bit above the leftmost divisor bit is 1, the divisor is XORed into the input (in other words, the input bit above each 1-bit in the divisor is toggled). The divisor is then shifted one bit to the right, and the process is repeated until the divisor reaches the right-hand end of the input row. Here is the entire calculation:

```
11010011101100 000 ← input right padded by 3 bits
1011 ← divisor
01100011101100 000 ← result
1011 ← divisor...
00111011101100 000
1011
00010111101100 000
1011
00000001101100 000
1011
00000000110100 000
1011
00000000011000 000
1011
00000000001110 000
1011
00000000000101 000
101 1
-----------------
00000000000000 100 ←remainder (3 bits)
```

Since the leftmost divisor bit zeroed every input bit it touched, when this process ends the only bits in the input row that can be nonzero are the nbits at the right-hand end of the row. These n bits are the remainder of the division step, and will also be the value of the CRC function (unless the chosen CRC specification calls for some post processing).

The validity of a received message can easily be verified by performing the above calculation again, this time with the check value added instead of zeroes. The remainder should equal zero if there are no detectable errors.

11010011101100 100 ←input with check value
1011 ← divisor
01100011101100 100 ←result
1011 ← divisor...
00111011101100 100
......
00000000001110 100
1011
00000000000101 100
101 1

---

0 ← reminder

BLOCK DIAGRAM OF CRC ENCODER & DECODER:

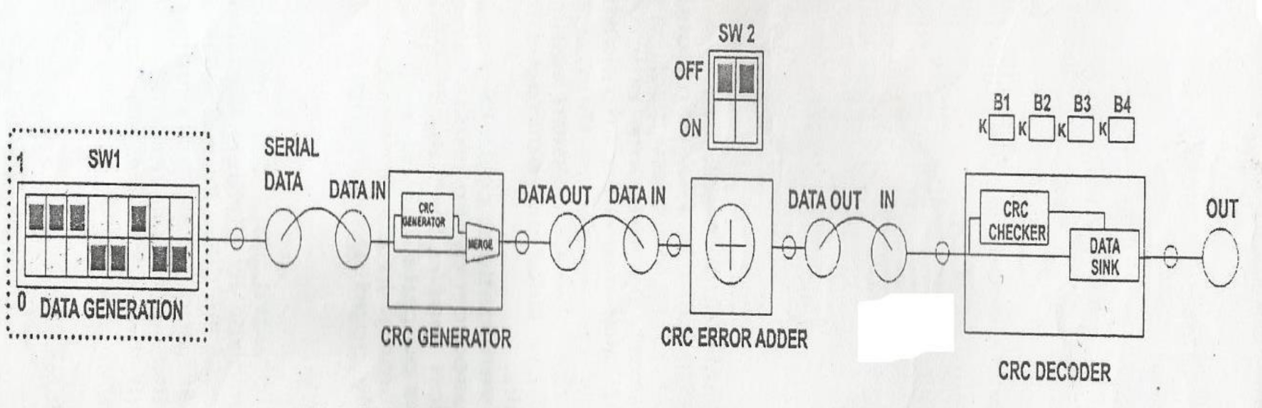


FIG 2. BLOCK DIAGRAM FOR CRC ENCODING AND DECODING

EXAMPLE:

- Determine the encoded message signal for following 8 bit data code using CRC

G(x) = $p^4 + p^3 + 1$ and M(x) = 11001100.

Sol: Given data

G (p) = $p^4 + p^3 + 1$

M (p) =$p^7 + p^6 + p^3 + p^2$

Now, we know that check bits $C(p) = Reminder[p^q * \frac{M(p)}{G(p)}]$

q=highest degree of the polynomial i.e., q=4

$$p^q * M\ (p) = p^4(p^7 + p^6 + p^3 + p^2) = p^{11} + p^{10} + p^7 + p^6$$

$$p^4 + p^3 + 1 \quad \Big| \quad P^{11} + p^{10} + p^7 + p^6 \quad \Big| \quad p^7 + p^2 + p + 1$$

$$\underline{p^{11} + p^{10} + p^7 \qquad\qquad\qquad\qquad}$$

$$p^6$$

$$\underline{P^6 + P^5 + P^2}$$

$$P^5 + P^2$$

$$\underline{p^5 + p^4 + p}$$

$$p^4 + p^2 + p$$

$$\underline{p^4 + p^3 + 1}$$

$$p^3 + p^2 + p + 1$$

$$C(p) = p^3 + p^2 + p + 1$$

$\therefore$ C= (1111)

$\therefore$ Code word X= [M:C]=[110011001111]

Now we have to find that there is any error presented or not, for that we have

$$\text{R}eminder\left[\frac{X(p)}{G(p)}\right] = 0$$

If Re=0 then there is no error presented in the received codeword

$$X(p) = \ p^{11} + p^{10} + p^7 + p^6 + p^3 + p^2 + p + 1$$

$$p^4 + p^3 + 1 \quad \Big| \quad p^{11} + p^{10} + p^7 + p^6 + p^3 + p^2 + p + 1 \quad \Big| p^7 + p^2 + p + 1$$

$$\underline{p^{11} + p^{10} + p^7 \qquad\qquad\qquad\qquad\qquad}$$

$$p^6 + p^3 + p^2 + p + 1$$

$$\underline{p^6 + p^5 + p \qquad\qquad\qquad}$$

$$p^5 + p^3 + p + 1$$

$$\underline{p^5 + p^4 + p \qquad\qquad}$$

$$p^4 + p^3 + 1$$

$$\underline{p^4 + p^3 + 1 \qquad\quad}$$

$\underline{0}$

$\therefore\ \text{R}eminder \left[\frac{X(p)}{G(p)}\right] = 0$, so there is no error presented in the received information

PROCEDURE:

1. Do the connections as per the block diagram shown in figure.
2. Connect the power supply to the kit and switch it on.
3. Set the data pattern as shown in block diagram using SW1 observe the 8-bit serial data at SERIAL DATA post.
4. Observe CRC encoded Signal at DATA OUT POST of CRC GENERATOR.
5. Connect DATA OUT to DATA IN post of CRC ERROR ADDER BLOCK to introduce 2-bit manual error. Introduce error by switch SW2.
6. To detect the signal connect DATA OUT to DATA IN post of CRC DECODER BLOCK.
7. Observe CRC decoded signal and correct the signal at outpost of CRC DECODER. Calculated CRC at receiver end is displayed on led B1 to B4.

APPLICATIONS:

1. A cyclic redundancy check (CRC) is an error-detecting code commonly used in digital networks and storage devices to detect accidental changes to raw data
2. A CRC-enabled device calculates a short, fixed-length binary sequence, known as the *check value* or improperly *the CRC*, for each block of data to be sent or stored and appends it to the data, forming a *codeword*. When a codeword is received or read, the device either compares its check value with one freshly calculated from the data block, or equivalently, performs a CRC on the whole codeword and compares the resulting check value with an expected *residue* constant. If the check values do not match, then the block contains a*data error*. The device may take corrective action, such as rereading the block or requesting that it be sent again. Otherwise, the data is assumed to be error-free

RESULT:

Hence we studied the cyclic redundancy code of encoding, decoding & generated Re=0 means there is no error in the code word.

VIVA QUESTIONS:

1. What do you mean by redundancy?
2. Define cyclic redundancy check
3. Define syndrome

GRAPH

EXPT NO: 12

CONVOLUTION CODE – ENCODER AND DECODER

AIM : To study convolution encoding and hard decision viterbi decoding for k=7 and rate=1/2.

APPARATUS:

1. Experimental kit ADCL-06
2. Patch chords
3. Power supply
4. 16 channel logic analyzer

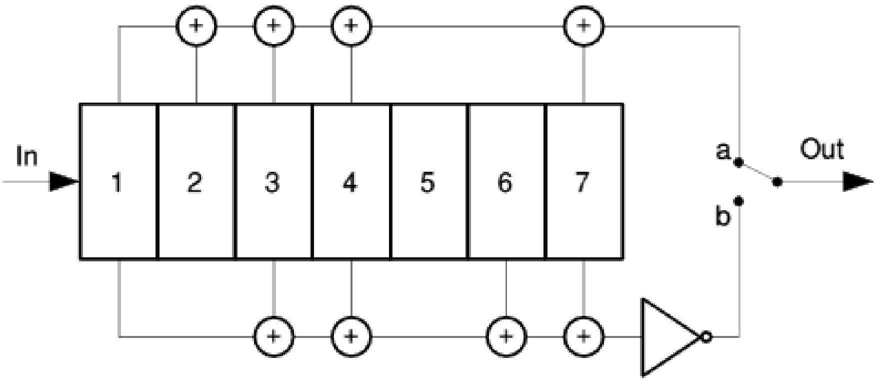
THEORY:

A convolution code works by adding some structured redundant information to the user's data and then correcting errors using this information. A conventional encoder is a linear system. A binary convolutional encoder can be represented as a shift register. The output of the encoder are modulo2 sums of the values in the certain register's cells. The input to the encoder is either the uuencoded sequence or the unencoded sequence added with the values of some register's cells.

The convolution encoder used in ADCL-06 supports INTELSAT standard. As per the standard the generating polynomials for k=7,R=1/2 are

$$G_o(x)=1+x^2+x^3+x^5+x^6$$

$$G_1(x)=1+x+x^2+x^3+x^6$$



K=7, R=½ Convolutional Encoder

2 Symbols are output for each Bit input

i.e $G_o(x)$ =133$_{(octal)}$ & $G_1(x)$=171$_{(octal).}$ The implementation depicted below and is used in conjuction with an R=1/2,k=7 hard decision viterbi decoder. The intent of this experiment is to help clarify the terms used to define the conventional encoding and vetrbi decoding takes place theoretically and to observe and verify the results practically.

We can approach the encoder in terms of its impulse response i.e. the response of the encoder to a single "one" bit that moves through it. Consider the contents of the register in fig (a)below

Register contents	Branch word	
	U1	U2
1 0 0 0 0 0 0	1	1
0 1 0 0 0 0 0	0	1
0 0 1 0 0 0 0	1	1
0 0 0 1 0 0 0	1	1
0 0 0 0 1 0 0	0	0
0 0 0 0 0 1 0	1	0
0 0 0 0 0 0 1	1	1

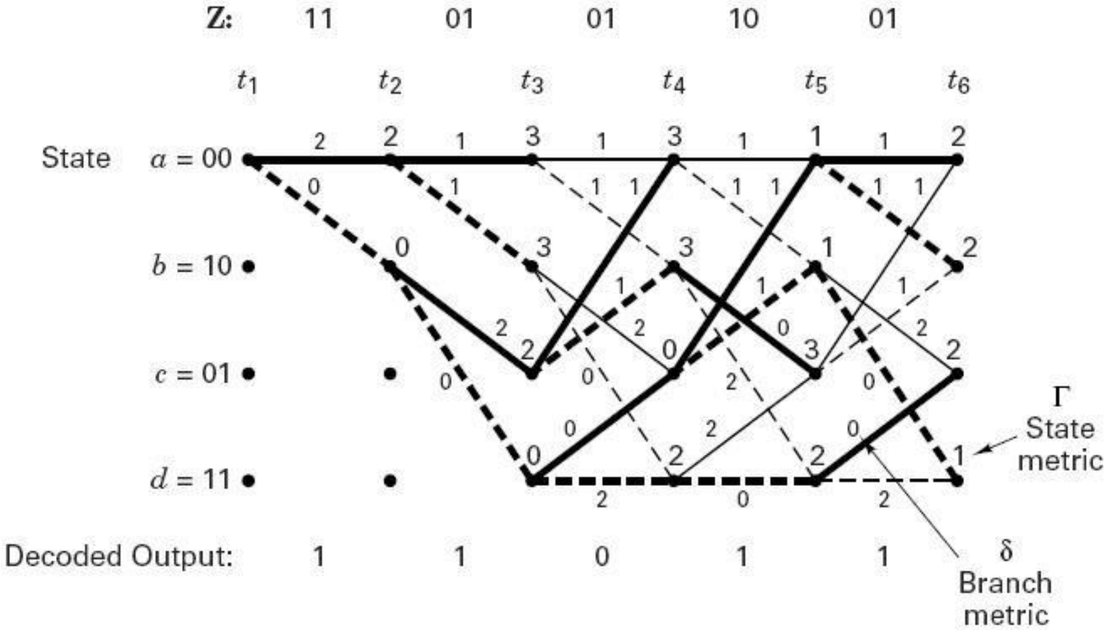
Input sequence : 1 0 0 0 0 0 0 Output sequence : 11 01 11 11 00 10 11 The output sequency for the input "one" is called the impulse response of the encoder. Then for the input sequencye m= 1 1 1 1 1 1 1. The output may be found by the superposition or the linear addition of the time shifted input "impulses" as follows figure below

Input(m)													output
1	11	01	11	11	00	10	11						
1		11	01	11	11	00	10	11					
1			11	01	11	11	00	10	11				
1				11	01	11	11	00	10	11			
1					11	01	11	11	00	10	11		
1						11	01	11	11	00	10	11	
1							11	01	11	11	00	10	11
o/p	11	10	01	10	10	10	00	11	11	00	01	01	11

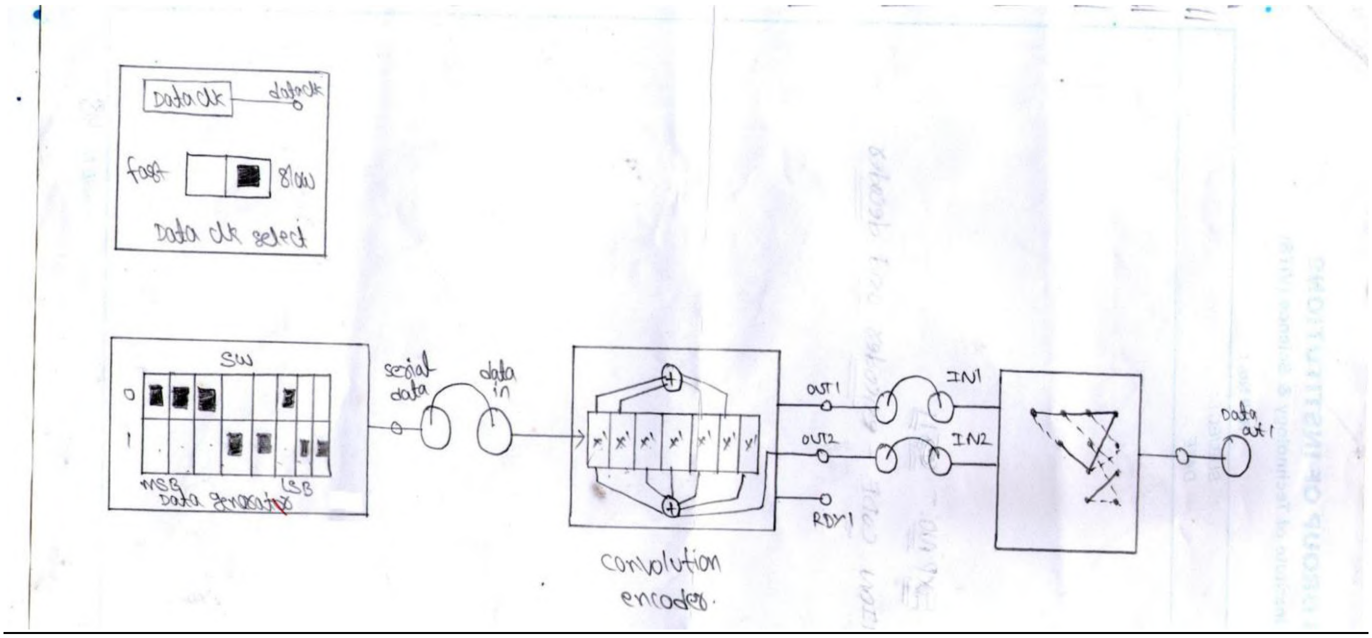
Convolution codes are linear. It is from this property of generating the output by the linear addition of time shifted impulses, or the convolution of the input sequence with the impulse response of the encoder, that we derive the name convolution encoder.

Viterbi convolution decoding:

Figure below shows the viterbi algorithm for decoding the original data.



BLOCK DIAGRAM:



From fig:

By using convolution code we will find code rate ie.., code rater=k/n = 1/2 (n=no of o/p's)
Constraint length =k=3(k=no of shifts)
Dimension of the code(n,k)=(2,1)

PROCEDURE:

1. Connect power supply in proper polarity to the kit “adcl-06” and switch it on.
2. Keep the data clk select switch sw2 towards slow position
3. Select data pattern using select switch sw1 in the data generator block
4. Connect serial data generator on board to data in of convolution encoder.
5. Observe RDY1 pin, convolutionally encoded data will be observed at out1 and out2 post. The convolutionally encoded data are valid from the instant when RDY1 goes high
6. Connect out1 and out2 post of convolution encoder block in1 and in2 of hard decision viterbi decoder block
7. Observe the decoded data at the data out1 post of hard decision viterbi decoder block
8. Repeat the proceure by keeping the data clk select switch towards fast position.

RESULT:

Thus we have studied convolution encoding and hard decision viterbi decoding technique of serial data and also observed how a serial data is convolutionally encoded and when it is passed through an error free medium how it is decoded using hard decision viterbi decoder.

APPLICATIONS:

1. Convolutional codes are used extensively in numerous applications in order to achieve reliable data transfer, including digital video, radio, mobile communication, and satellite communication.

2. Convolution code is a type of error correcting in telecommunications

VIVA QUESTIONS:

1. Define constraint length of a convolution code
2. Explain code tree for a convolution code
3. Explain the trellis diagram for convolution code
4. Explain viterbi algorithm for decoding convolution code

GRAPH