WIRELESS FM COMMUNICATION

BETWEEN PC'S

P - 1382

PROJECT REPOR

SUBMITTED F

M. ARUN PRASAD

C. BHARATHIRAJ

C. VASANTHRA

K. R. VINO

GUIDED

Prof. M. RAMASAMY, M.E., MIEEE(US

MISTE, MIE, C.Engg(I), MBME

IN PARTIAL FULFILMENT OF THE REQUIREMEN

FOR THE AWARD OF THE DEGREE

BACHELOR OF ENGINEERING

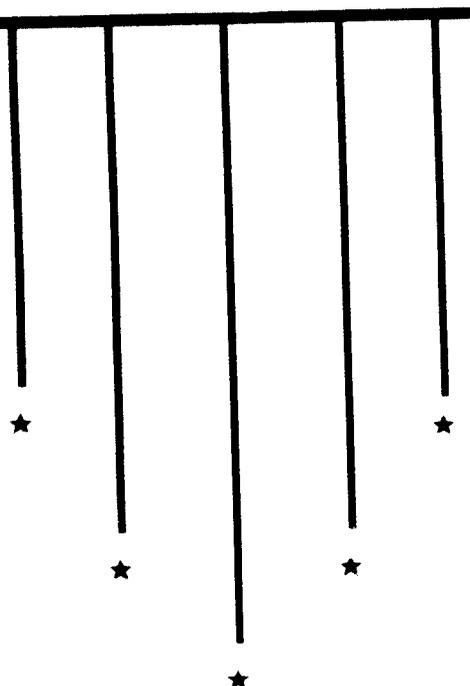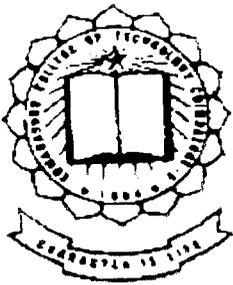**ELECTRONICS & COMMUNICATION ENGINEERI**

OF THE BHARATHIAR UNIVERS

2000 - 2001

Department of Electronics & Communication Engineering

Kumaraguru College of Technology

Coimbatore - 641 006

Kumaraguru College of Technology

Coimbatore - 641006

## Department of Electronics and Communication Engineering

Certificate

This is to certify that this project entitled

Wireless FM Communication between PC's

has been submitted by

_____

In partial fulfillment of the requirements for the award of Degree of Bachelor of

Engineering in the Electronics and Communication Engineering Branch of the Bharathiar

University, Coimbatore - 641046 during the Academic Year 2000-01
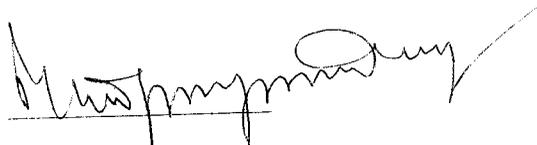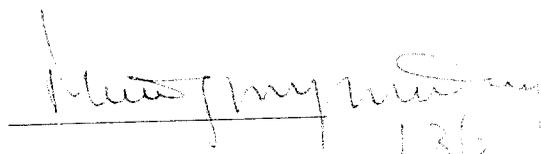
_____
(Guide)

_____
(Head of Department)

Certified that the candidate was examined by us in the Project Work Viva-Voce

Examination held on _____13-3-2001_____

University Register Number

_____

_____
(Internal Examiner)

_____
(External Examiner)

ACKNOWLEDGEMENTS

PREFACE

Computer Communication is unavoidable for ablest one of the three reasons. Firstly to share peripherals. Secondly to access data or application software residing on another computer. Thirdly, to avail special processing capabilities that resides only on another computer. The phenomenal growth of communication technologies in terms of connectivity hardware and software made possible the above necessities happen at a click of a button. It goes without saying that communication is one of the hottest areas in IT world.

Henceforth, we tried for a project that focuses on computer communication and finally prepared to concentrate and work on RS232C STANDARDS and TCP/IP protocol suites, not to mention TCP/IP as the most popular Internet protocol.

Here we use wireless communication so that usage of cable can be reduced which reduces the cost. We can also transmit for very long-range distance. For wireless communication we use FM transmitter and receiver.

CONTENTS

Here we have shown interest to develop an equipment and software, which will replace the cable used for communication between two systems.

We have used FM transmitter and receiver for transmission and reception of data. The transmission is done using serial ports. The software for this interface is written in turbo C.

Using this interface we are performing file transaction between PC's.

The data available in the COM port will be in RS-232 form. This will be converted to TTL form then this is given to FSK modulator to produce an analog signal, which is used to transmit through the FM transmitter.

During the reception the reverse process will take place. The FM receiver after receiving the data passes to the FSK demodulator and the output is given to the TTL to RS-232 converter then given to COM port again.

1.1 COMPUTER COMMUNICATION:

This term refers to sending, receiving and processing of information by electric means modem communication system is first concerned with the sorting of information before its transmission.

Depending on the physical distances separating the components we distinguish two major cases:

INTRASYSTEM communication, which occurs within a single computer system and involves information transfer over distance of not more than a meter or so.

INTERSYSTEM communication, which involves communication over long distance.

1.1.1 MODES OF COMMUNICATION:

There are two modes of communication parallel mode and serial mode. In short distance communication parallel mode is the general choice. In long distance, serial is the suitable mode.

In parallel mode, the interconnection cable is a bus with separate lines for all bits of the bytes, lines for handshaking signals, lines for error checking signals such as parity. Thus the interconnecting cable is expensive.

In serial mode, the interconnecting cable is a bus with two lines for receiving and transmitting the bit, lines for handshaking signals and line error checking signals.

Two types of serial transfer are in existence, asynchronous and synchronous transmission. The asynchronous format is character oriented;

Each character carries the information of the start and stop bits. This is also known as framing. This format is generally used for high-speed transmission.

Serial communication also can also be classified according to the direction and simultaneity.

In simplex transmission data are transmitted in only one direction. In duplex transmission occurs in both directions. However if the transmission goes one way at a time it is called half duplex if it goes in both ways simultaneously it is called full duplex.

The rate at which the bits are transmitted/seconds is called a baud in serial transmission.

A modem (modulator-demodulator) is a circuit that translates digital data into audio tone frequencies for transmission over telephone lines.

And convert audio frequencies into digital data for reception. The modulator technique generally used is called frequency shift keying.

COMPUTER NETWORKS:

A group of computer, user terminals and other system components that are linked together over long distances constitute a computer network.

A network can be a LAN-local area network OR a WAN-wide area network.

We can define a LAN to be a set of independent computers connected together by short and fast communication links. And WAN is founded to be a set of independent computers connected together by long and hence slower communication links. Accessing of resources by LAN network is much faster than by means of WAN. Number of LAN can also be interlined to form a WAN.

1.2 COMPUTER PORTS:

This is an introduction to program the wireless PC's communication ports. The basic way by which the computer communicates with the devices is through ports. On PC's it's basically an 8-bit doorway. The computer communicates with other devices either through serial ports or through parallel ports.

Each method has it's own merits and demerits. A parallel port transfers an entire byte at time where as a serial port transfer only one bit at a time. This is the basic difference between two ports. These serial ports take eight-fold time that of a parallel port to transfer a byte. This makes us to come to a conclusion that parallel transfer is much faster than serial transfer is. In spite of this we have selected serial port as the medium of transfer for this project.

1.3 SERIAL AND PARALLEL TRANSMISSION:

Historically, the need for serial port arose when mainframe become large and fast enough to share time between several program or tasks, rapidly switching between them to give each of them an illusion of exclusive attention. The multitask facilities each user to work on an independent terminal. These works even in cases where independent terminal are at some distances form the main system and terminal as per requirements.

The main drawback of parallel communication, which forced people to use serial communication in cases of long distances, is the high cost due to the need of separate wires for each bit to be transmitted simultaneously.

Another serious drawback in cases of long distances is the speeds at which the bits travel along the different wires, which differs slightly. Thus bits reach the receiver at different times even together. Over long distances these garbles the message.

Serial communication does not require use of expensive cables. The phone line transmits the data over long distances. To extend the communication over thousands of kilometers the usage of modems at both ends are the only additional requirements.

1.3.1 SYNCHRONOUS SERIAL COMMUNICATION:

Synchronous serial communication is efficient when high data transfer rates are required. Here synchronous or bit stream synchronisation that involves sending a group of characters in a continuos bit stream is used. The data transfer is controlled by a timing signal at the originating device. This may originate from the terminal itself or it may be provided by communication components like modems, multiplexers or front-end processor channel. At the receiving end, the timing is derived from a synchronized clock to ensure synchronization. The figure shows the format of synchronous data transfer.

To continue and obtain synchronization, data to be sent is blocked into a group of data bits preceded by the transmission of one or more special character. These synchronising characters have a unique bit configuration of 'zeros' and 'ones' that allows the receivers to recognize and synchronize itself.

Since the data is blocked into a group of character, terminal transmitting and receiving data using this mode must have a buffer for storage of character blocks. More complex circuitry is required since the receiving terminal must remain in step with the data originally transmitted. Considering these aspects, asynchronous serial communication is used in the project. The details of asynchronous serial communication are dealt in the following paragraphs.

1.3.2 ASYNCHRONOUS SERIAL COMMUNICATION:

Almost all computers store and manipulate their data in parallel. This means that when a bytes is send from one part of the PC to another, it is not send one bit at a time, instead it is send several bits at a time over a number of buses running in parallel. Since serial communication from a PC to another device is done serially, meaning that data are sent one bit at a time. A communication interface must be able to take data bits that are received in parallel and send out the individual bits separately.

The data lines in serial communication can only be in either mark (negative voltage) or space (positive voltage) condition. Therefore any transmitted data must first be translate into a sequence of Mark's and Space's. For the purpose of this translation, a MARK represents a '1' and SPACES represents a '0'.

The above mentioned details pertain to serial communication in general and since for communication between PCs any PC can initiate transmission and terminate the transmission at any time and the length of the data transmitted varies widely, asynchronous serial transmission is preferred which does not require the two PCs to be in synchronism.

Three modes of communication are possible simplex, Half Duplex and Full Duplex. In simplex mode, data transmission takes place from one system only. Half-Duplex transmission means that communication can take place

in either direction between two systems, but can occur in one direction at a time. Full duplex means that each system can send and receive at the same time. The half-duplex mode is used in the project.

FRAMING:

In the case of asynchronous serial communications, the bits are representing one byte, which are known as the data bits preceded and followed by start stop and parity bits. This process is known as framing.

The various bits and their meaning are presented below.

START BITS:

A start bit is always added at the beginning of the frame to alert the receiving device that, data are arriving and synchronize the mechanism that separates out the individual bit start is a SPACE or binary zero.

DATA BITS:

These represent the actual data. They can be 7 bits or 8 bits depending upon the data to be transmitted. Text files need only seven data bits per byte of data. These bits are transmitted with least significant bit first.

Binary files need 8- bit data representation.

PARITY BIT:

Parity checking is a method of testing whether the transmission is being received correctly. The sending device adds a parity bit, the value of which (0 or 1) depends on the contents of the data bits. The receiving device checks that the parity bit does indeed bear the correct relationship to the other bits. EVEN, ODD, SPACES or MARK parity schemes can also be used.

STOP BITS:

At the end of each frame, stop bits are sent. This can be 1, 1.5 or 2. More than one-stop bits are generally used when the receiving device requires extra time before it can handle the next incoming character. When the number of data bits is five we should always go for 1.5 stop bits.

2.1 INTERFACING:

Interface is a process where information is being converted from one to another or transported from one machine to another using FM transmitter and receiver.

In the case of computer, interface made through ports.

A port is a gateway of data transformer. On PC's it is normally an 8-bit gateway. The two ways that a computer communicates with other devices are through serial and parallel ports.

The parallel ports can transfer an entire byte at once but a serial port can transfer only bit at a time. Thus parallel transfer much quicker than serial transfers.

One of the disadvantages of parallel communications over long distances is the high cost associated with it. Because interconnecting cable must have separate wires for as many bits as are transmitted separately.

Overlong distances, there is also a problem of line noise, or cross talk between all this data wires. Serial communications at the other ends does not need the use of expensive cables. To communicate over long distances, serial cables are the answers.

The phone line allows us only a simple wire for the flow of data, so only serial transfers can take place over phone lines to communicate with a distance computer.

There are two ways this can be done. Writing a program takes each byte and spools the bits out over a single bit of the port, one at a time, or uses a serial interface.

Bit spooling needs a very precise timing scheme on both transmit and receiving ends and a half-crazed programmer to write the programs. Obviously the serial interface is the solution. The PC uses an RS232 serial interface for serial data transfers. It is the serial interface circuit that does the bit spooling.

The PC sends a byte to a CPU port connected to the serial interface, spools the bits over the modem cable to the modem. The modem converts each bit into a sound frequency on the phone line. That's why we hear that high-pitched squeal. Receiving data is just the reverse process. Using a serial interface standardizes communications and simplifies the programming.

If we have been using a communication program to dial up other computer we have everything we need.

A modem is optional and is necessary only if we wish to communicate over phone lines without a modem, two computers or other device can communicate over a direct cable connection. This is a connection between serial parts of each device.

2.2 RS-232C SERIAL DATA STANDARD:

In 1960's as the use of time-share computer terminals became more widespread, modems were developed. So that terminals could use phone lines to communicate with distant computer. As we stated earlier, modems and other devices used to send serial data transferred to as data communication equipment

or DCE. The terminal or computers that are sending or receiving data referred to as data terminal equipment or DTE.

In response to the need for signal and handshake standards between DTE and DCE, the Electronics Industries Association developed EIA standard RS-232c. This standard describes the function of 25 signal and handshake pins for serial data transfer. It also describes voltage levels, impedance levels, rise and fall times, maximum bit rate and maximum capacitance levels for the signal lines. Before we work our through the 25 pin functions we will take a brief look at some other hardware aspects of RS-232.

RS-232C specifies 25 signal pins and it specifies that the DTE connector should be a male, and the DCE connector should be a female. A specific connector is not given, but the most commonly used connectors are the DB-25S females. When you are writing up these connectors, it is important to note the order in which the pins are numbered.

The voltage levels for all RS-232C signals are as follows. A logic high or mark is a voltage between -3V to -15V under load (-25V no load). A logic low or space is a voltage between +3V and +15v under load (+25V under no load). Voltages such as + or - 12V are commonly used.

2.3 RS-232 INTERFACE:

RS 232 is what is known as a "serial " interface. It transfers data between devices on a single wire, byte by byte being sent one bit at a time.

RS 232 JOBS:

Let us turn attention to basic areas to successfully solve an RS 232-interface problem.

CABLING:

Getting the appropriate connector pins at one end connected to appropriate pins at the other end. This job is the most obvious task associated with RS 232, but is only one third of the journey.

DB 9:

This is the style of connector generally used with RS 232. It is a D shaped and it has 9 pins in two rows. The connector comes in male as well as in female types.

OPTIONS:

Baud is the unit name for measuring speed of communication in bits per second (bps). The data rate of serial channel is equal to the number of characters sent in one second.

CHOICE OF BAUD RATES:

50, 110, 300, 600,1200,2400, 4800, 9600 and 19200 baud. The two communicating devices must agree on this.

Out these 300, 1200, 4800 and 9600 are commonly preferred Baud rates.

NUMBER OF BITS OR DATA LENGTH:

Our choice of data bits will be from 5 to 8, with 8 being the most common for newer equipment and 5 or 6 bits were used with ancient teletypes. Both machine should agree, but many (at least essentially) even without agreement.

PARITY:

Parity is an error-checking scheme, so it is most useful where the communication path is of dubious quality.

Our choice will be none, even or odd. Choose none on both machines if possible, otherwise select an agreeing choice.

Check whether each byte is followed by a parity bit, and if so whether 'even ' or 'odd' parity is to be used.

STOP BITS:

After each byte, the line must be 'at rest' for a pre specified amount of time. This parity to make sure both sides are synchronized and parity to allow slower device

time to recover. This 'stop bit' is a set of 1 or 1.5 or 2 bits time. 1.5 is usually used with 5 -bit character length.

The first is software at each end to equalize this parameter. With PC's we are often provided with the manufacture software for setting up many options including the configuration of the serial ports

2.4 RS-232C SIGNAL PINS:

RS-232C has 25 pins out of which only few pins are used. First note that the signal direction is specified with respect to the DCE. This conversion is part of the standard. We have found it very helpful to put arrowheads on all signal lines as shown in figure.

Next observe that there is both a chassis ground signal ground. To prevent large AC induced currents in the signal grounds this two should be connected together only at the power supply in the terminal or the computer. The TxD, RxD and the handshake the signal shown with common names in figure are the once most often is used for simple systems.

This signal control what is called the primary or forward communication channel of the modem? Some modem communication over secondary or backward channel, which operates in the reverse direction from the forward channel and at a much lower baud rate. Pin 12,13,14,16and 19 are the data and handshake lines for this backward channel pins 15,17,21 and 24 are used for synchronous data communication.

INITIALIZATION OF THE COMMUNICATION PORT

The system will be ready to communicate only after initialization of the communication port.

We use *bioserialcom* instruction of Turbo C to initialize the communication port with the baud rate, parity and stop bits.

This instruction can also be used for receiving and sending data through the serial port. After initializing, the system is ready to communicate.

ALTERNATIVE METHOD TO USE THE PORT:

Using Interrupts directly within the functions, we can set communication portion Turbo C, the int86 and INTR instructions are used for this purpose. For this purpose following data are used.

INITIALISE COMMUNICATION PORT:

Initialize a serial communication port to a desired baud rate, parity, word-length and number of stop bits.

3.1 INT14H FUNCTION 00H:

CALL WITH:

AH = 00H

AL = initialization parameter (see notes)

DX = communication port number (0 =COM1, 1=COM2, etc)

RETURNS:

AH = port status

Bit significance (if set)

7- timed out

6- transmit shift register empty

5- transmit holding register empty

4- break detected

3- framing error detected

2- parity error detected

1- overrun error data detected

0- receive data ready

AL = modem status

Bit significance (if set)

7- receive line signal detect

6- ring indicator

5- data set ready

4- clear to send

3- change in receive line signal detect

2-trailing edge ring indicator

1-change in data set ready status

0- change in clear to send status

NOTES:

The initialization parameter byte is defined as follows:

Baud rate	Parity	Stop bit	Word length
7 6 5	4 3	2	1 0
000 = 110	X0 = none	0 = 1bit	10 = 7bits
001 = 150	01 = odd	1 = 2 bits	11 = 8 bits
010 = 300	11 = even		
011 = 600			
100 = 1200			
101 = 2400			
110 = 4800			
111 = 9600			

Write character to communication port:

Write a character to the specified communication port returning the curr

status of the port.

3.2 INT 14H FUNCTION 01H

CALL WITH:

AH = 01H

AL = Character

DX = COMMUNICATION PORT NUMBER

(0 = COM1, 1 = COM2)

RETURNS:

If function successful

AH BIT 7 = 0

AH BIT 0-6 = PORT STATUS

Bit significance (if set)

6 - transmit shift register empty

5 - transmit holding register empty

4 - break detected

3 - framing error detected

2 - parity error detected

1 - overrun error detected

0 - receive data ready

AL = Character (unchanged)

If function successful (timed out)

AH bit 7 = 1

READ CHARACTER FROM COMMUNICATION PORT:

Read a character from the specified serial communication port, also returning the port status.

3.3 INT 14H FUNCTION 02H

CALL WITH:

AH = 02H

DX = COMMUNICATION PORT NUMBER

(0 = COM1, 1 = COM2)

RETURNS:

If function successful

AH BIT 7 = 0

AH BIT 0-6 = STATUS

Bit significance (if set)

4 - break detected

3 - framing error detected

2 - parity error detected

1 - overrun error detected

If function successful (timed out)

AH bit 7 = 1

GET COMMUNICATION PORT STATUS:

Returns the status of the specified serial two-communication port.

3.4 INT 14H FUNCTION 03H

CALL WITH:

AH = 03H

DX = communication port number

(0 = COM1, 1 = COM2)

RETURNS:

AH = port status

(See INT 14H function 00H)

AL = Modern status

(See INT 14H Function 00H)

EXTENDED INITIALIZE COMMUNICATION PORT:

Initialize a serial communication port to a desired baud rate, parity, word length and number of stop bit. Provides a superset of INT 14H Function 00H capability PS/2 machines.

3.5 INT 14H FUNCTION 04H

CALL WITH:

AH = 04H

AL= Break flag

00H none

01H odd

02H even

03H steak parity odd

04H steak parity even

BL = stop bits

00H 1 stop bit

00H 2 stop bit if word length = 6 to 8 bits

01H 1.5 stop bits if word length = 5 bits

CH = word length

00H 5 bits

01H 6 bits

02H 7 bits

03H 8 bits

CL = Baud rate

00H 110 Baud

01H 150 Baud

02H 300 Baud

03H 600 Baud

04H 1200 Baud

05H 2400 Baud

06H 4800 Baud

07H 9600 Baud

08H 19200 Baud

DX = communication port number

MODULATION AND TRANSMISSION

The digital signal from the computer cannot be transmitted directly, as they do not have much power with them. The RS 232C signals that are coming from the USART are bipolar and have both positive and negative voltage levels. Transmitting such a discrete bipolar signal may cause loss of data. So these signals have to be modulated properly on a carrier before transmission. The fundamental concepts of modulation and demodulation are discussed in this chapter. At the end of the chapter the use of frequency shift keying (FSK-FM) as the modulation scheme for wireless data transmission is justified.

4.1 MODULATION TECHNIQUES:

Before going into the various details of various modulations techniques, let us get acquainted with the concept of modulation. Modulation is the process where in some parameters of a high frequency signal, termed as carrier is varied in accordance with the signal to be transmitted. The signal to be transmitted is known as the modulating signal. Various modulation methods have been developed for transmission of signals with minimum Possible distortion.

The various modulation techniques that are available for signal transmission are given below.

1. Amplitude modulation

2. Frequency modulation

3. Phase modulation

4. Pulse modulation

First three methods come under Analogue Modulation where as in digital modulation a pulse train is used as a carrier.

4.1.1 ANALOGUE MODULATION:

Analogue modulation may be divided into two parts.

1. Amplitude modulation

2. Phase modulation

3. Frequency modulation

AMPLITUDE MODULATION:

The process of Amplitude modulation consists of varying the peak amplitude of the modulating signal in proportion to the instantaneous amplitude of the modulating signal. The Signal to Noise ratio (S/N) is comparatively less in Amplitude modulation. This means that the data transmitted using amplitude modulation is more prone to noise. Also the power required for AM is more prone to noise. Also the power required for AM is more when compared with FM. So we should select a modulation scheme which consumes less power and is less prone to noise.

PHASE MODULATION:

Phase shift keying (PSK) is also called as phase modulation. The signal differs by phase shift instead of frequency or amplitude. Typically, a signal's phase shift is measured relative to the previous signal. In such case, the term differential phase shift keying (DPSK) is often used. Here n bits can be assigned a signal having one of 2^n phase shifts, giving a technique in which the bit rate is n times the baud rate.

FREQUENCY MODULATION:

In Frequency modulation, the frequency of the carrier signal is varied in accordance with the instantaneous amplitude of the modulating signal, without any variations in the amplitude of the carrier wave.

These factors when taken into consideration, gives frequency modulation a pric over AM. We will discuss in detail about the frequency modulation technique in following section.

Why preferred FM to AM?

•FM gives a noiseless reception. Noise being a form of amplitude variations, FM receivers will reject such signals.

•FM operates for a wide bandwidth.

•FM provides high quality reception.

•FM provides adjacent channel or side band rejection by providing a guard band.

4.2 FM-AN ANALYTICAL APPROACH:

We have been talking about the theory of frequency modulation as an analytical approach for the technique that will provide us with more details about its feature. When the modulating frequency is zero, the output frequency equals f_c (center frequency). When the modulating signal reaches its positive peak, the frequency of the modulated signal is maximum and equals $(f_c + f_m)$. At the negative peaks of the modulating signal, the frequency of FM wave becomes minimum $(f_c - f_m)$. Thus, the process of frequency modulation makes the frequency of FM wave to deviate from the centre frequency by an amount (Δf –frequency deviation). The analytical approach is given below.

Assume modulating signal to be represented by:

$e_m = E_m \cos(\omega_m t)$

The carrier wave being represented by

$e_c = E_c \sin(\omega_c t)$

Where $(\omega_c t + \theta)$ represents total phase angle θ at time t and represents the initial phase angle.

Thus,

$\phi = (\omega_c t + \theta)$

The angular velocity may be determined by finding the rate of change of phase angle.

Angular velocity $= d\phi/dt = \omega_c$

After the frequency modulation, angular velocity of the carrier wave varies in proportion to the instantaneous amplitude of the modulating signal. The instantaneous angular velocity ω_i is given by

$$\omega_i = \omega_c + Ke_m$$

$$= \omega_c \pm K\ E_m\cos(\omega_m t)$$

Where K is the constant of proportionality.

Maximum frequency shift or deviation occurs when the cosine term in equation has a value of ± 1. Under this condition, the instantaneous angular velocity is given by

$$\omega_i = \omega_c \pm K\ E_m$$

So that the maximum frequency deviation is given by

$$\Delta f = KE_m/2\pi$$

This gives $KE_m = 2\pi\ .\Delta f$

Therefore, $\omega_i = \omega_c + 2\pi\ .\Delta f \cos(\omega_m t)$

Integrating instantaneous frequency gives the instantaneous phase angle of the FM wave

$$\Phi_I = \int \omega_i\ dt$$

$$= \int(\omega_c + 2\pi\ .\Delta f \omega_c t)\ dt$$

$$= (\omega_c + 2\pi\ .\Delta f/(\omega_m \sin \omega_c t) + \theta)$$

Where θ is constant of integration representing a constant phase angle and may be neglected in the following analysis.

The instantaneous amplitude of the modulated waves is given by

$$e_{mod} = E_c \sin \Phi_i$$

$$= E_c \sin (\omega_c t + \Delta f/\omega_m \sin \omega_m t)$$

The ratio $(\Delta f/\omega_m)$ is termed as the modulating index of the frequency modulated wave and is denoted by f_m. Thus, we find that the instantaneous frequency of the carrier is changed in proportional to the modulating signal amplitude.

4.3 FSK MODULATION:

Frequency shift keying (FSK) is most commonly used method for transmitting digital data over Telecommunication links. In order to use FSK, a modem (MOdulator DEModulator) is needed to translate digital 1's & 0's into their respective frequencies and back again.

In FSK modulation the carrier frequency is shifted in steps or levels corresponding to the levels of digital modulating signal. In case of a binary signal, two carrier frequencies are used, one corresponding to binary '0' other to binary '1'.

Frequency shift keying employs two different carrier frequencies which are switched ON and OFF by the mark and space signal. A mark signal switches ON one carrier while the other carrier remains OFF. Under the space conditions, the first carrier is switched OFF and the second carrier is switched ON. By applying the FSK signal before it is allowed to modulate a high frequency carrier, the noise

problem can be improved drastically. The data communication becomes very reliable, but the data transfer rate comes down if FSK is used because FSK standard supports a maximum baud rate of 1.2 KBPS.

Each frequency change conveys two bits of data; that is, the bit rate is twice the baud rate. In general, n bits can have one of 2^n combinations, and each can be assigned to one of 2^n frequencies. Here, the bit rate is n times the baud rate.

4.4 ANTENNA PROPAGATION:

Antennas are basic component of any electric system and are the connection link between the transmitter, and the receiver. The type of antenna used depends on the frequency of transmission. The frequency of the transmission in this project is 88-108Mhz. As the power of the transmitted waves is small we use a telescopic antenna whose length can be adjusted. The length of the antenna is determined by the frequency of the transmitted wave. Usually the length is about 1/10th of the wavelength.

Electromagnetic waves are oscillation, which propagate with the velocity of light in free space. Electromagnetic waves consist of moving fields of electric and magnetic force. The radio waves transmitted from the antenna may reach the receiving antenna following any of the modes of propagation depending upon several factors like frequency, distance etc.

Up to 2Mhz ground wave propagation takes place. Here the wave is guided along the surface of the earth just as wave-guide or transmission line

29

guides an electromagnetic wave. Between 20 to 30Mhz, sky wave propagation takes place. In this mode of propagation, electromagnetic waves reach the receiving point after reflection from the ionized region in the upper atmosphere called ionosphere. The space wave propagation is of practical importance at VHF bands. As our transmitted wave lies in the FM range, space wave propagation takes place.

PERSONAL COMPUTER → LOGIC LEVEL CONVERTER → FSK MODULATOR → FM MODULATOR → FM TRANSMITTER → ANTENNA

+5V (to LOGIC LEVEL CONVERTER)

+12V (to FSK MODULATOR)

+12V (to FM TRANSMITTER)
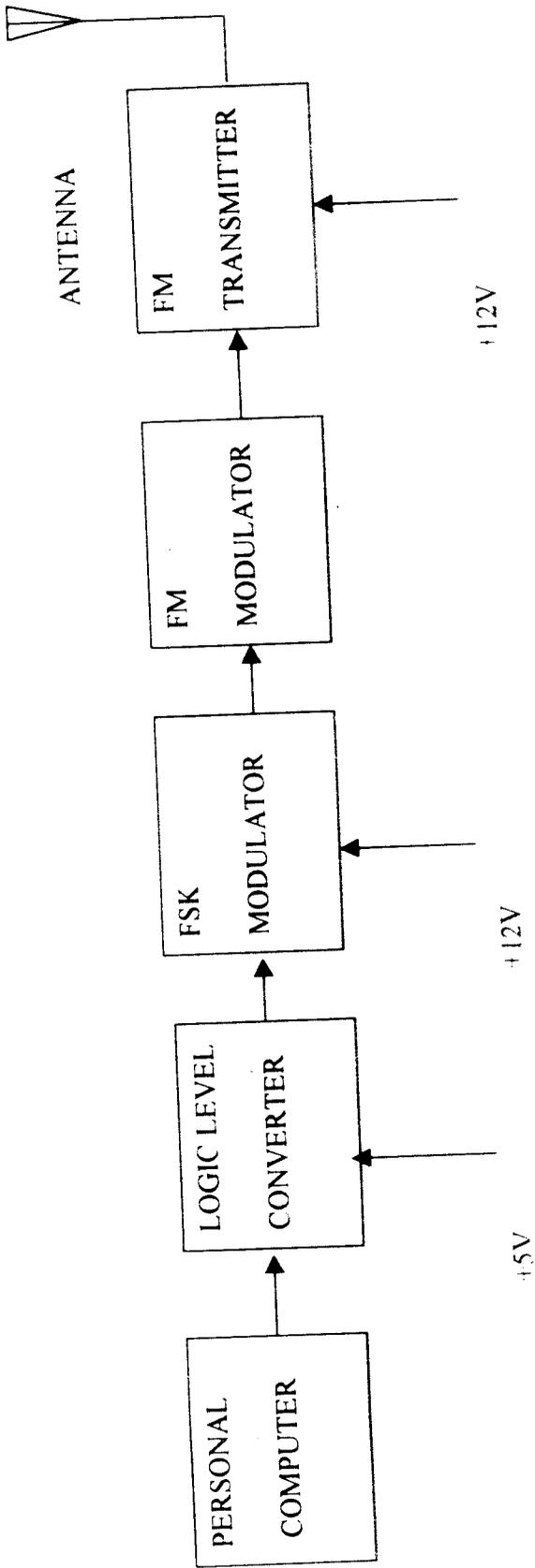
BLOCK DIAGRAM OF FM TRANSMITTER

RECEPTION AND DEMODULATION

The process of recovering the original modulating signal from a modulated wave is known as demodulation or detection. The type of demodulation technique employed depends on the technique used. This project uses FM transmission for transmission of data.

To receive the FM signal the most popularly used technique is super heterodyne reception. The forthcoming section deals with principle of super reception.

5.1 SUPER HETERODYNE RECEPTION:

Super heterodyne is the process of operation on modulated radio waves to obtain similarly modulated waves of a different frequency. In general this process uses the locally generated wave, which determines the change of frequency. The word SUPER stands for SUPER- sonic HET-aerodyne which means that the heterodyning takes place with the resulting output frequency greater than audio. Thus this principle of reception involves one or more changes of frequency before the modulating signal is extracted from the modulated wave. The above principle is used in the reception of FM waves.

ANTENNA

FM
RECEIVER

12V

FSK
DEMODULATOR

12V

LOGIC LEVEL
CONVERTER

5V

PERSONAL
COMPUTER
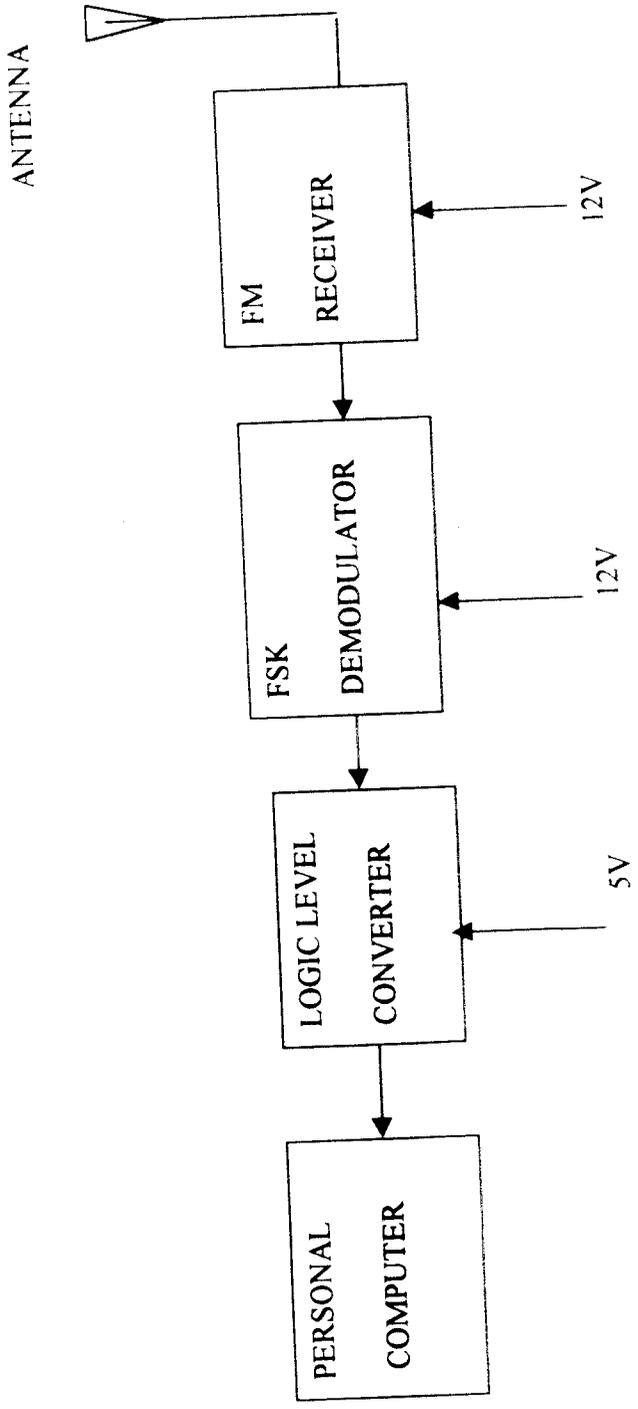
BLOCK DIAGRAM OF FM TRANSMITTER

Super heterodyne reception method has the following advantage over other methods.

1. Improved selectivity.

2. Improved receiver stability.

3. Higher gain per stage because the intermediate frequency (IF) amplifier are operated at a lower frequency.

4. Uniform bandwidth because of fixed intermediate frequency.

5.2 DEMODULATION:

The process of extracting modulating signal from a frequency-modulated carrier is known as frequency modulation or detection. The electronic circuits that perform the demodulation process are called FM detectors. The FM detectors perform the detection in two steps.

i. It converts the frequency modulated signal into its corresponding amplitude modulated signal by using frequency dependent circuits i.e, circuits whose output voltage depends on input frequency from which original modulating signal is detected. Such circuits are called frequency discriminators.

ii. The original modulating signal is recovered from this AM signal (converted from FM to AM in previous step) by using a linear diode envelope detector.

The threshold can be improved by pre-emphasis and de-emphasis circuits

6.1 RS 232 LEVEL CONVERTER:

FEATURES:

- ◆ ESD Protection for RS-232 I/O Pins:

- ◆ ±15kV—Human Body Model

- ◆ ±8kV—IEC1000-4-2, Contact Discharge

- ◆ ±15kV—IEC1000-4-2, Air-Gap Discharge

- ◆ Latchup Free (unlike bipolar equivalents)

- ◆ Guaranteed 120kbps Data Rate—LapLink™ Compatible

- ◆ Guaranteed 3V/µs Min Slew Rate

GENERAL DESCRIPTION:

The MAX232E line drivers/receivers are designed for RS-232 and V.28 communications in harsh environments. Each transmitter output and receiver input is protected against ±15kV electrostatic discharge (ESD) shocks, without latchup. The various combinations of features are outlined in the Selection Guide. The drivers and receivers for all ten devices meet all EIA/TIA-232E and CCITT V.28 specifications at data rates up to 120kbps, when loaded in accordance with the EIA/TIA-232E specification.

CIRCUIT DIAGRAM OF RS-232 LEVEL CONVERTER:
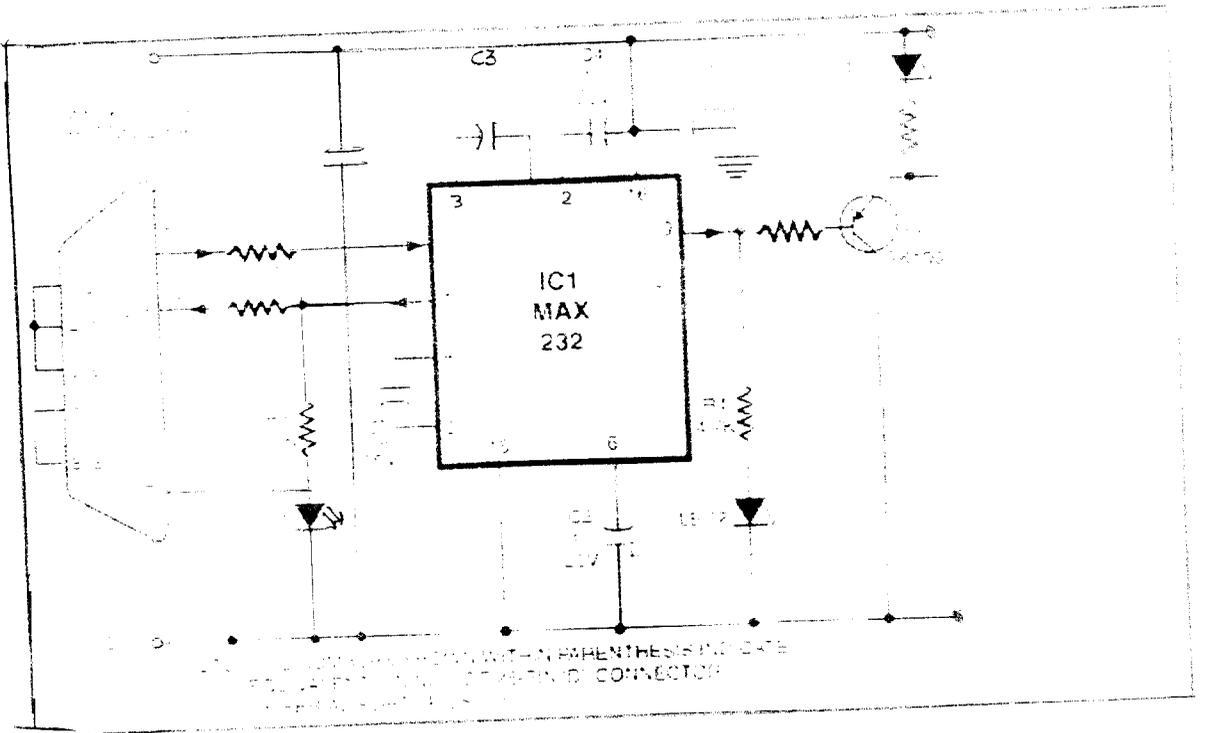
6.2 XR2206 FSK MODULATOR:

FEATURES OF XR 2206:

- Low sine wave distortion   0.5% typical

- Excellent temperature stability   typical 20 PM/Degree

- Wide sweep range   2000:1 typical

- Low supply sensitivity

- Wide supply range   10V to 26V

GENERAL DESCRIPTION:

The XR2206 is comprised of four functional blocks, a voltage controlled oscillator (VCO), an analog multiplier and sine wave shaper, a unity gain buffer amplifier and set of current switches. The VCO actually introduces an output frequency proportional to input current, which is produced by a resistor from the timing terminal to ground. The current to the VCO pin is controlled by an FSK input pin, to produce an output frequency. With two timing pins two discrete output frequencies can be independently produced for FSK generation applications.

The XR2206 can be operated with two separate timing resistors; R1 & R2 connected to pins 7&8 respectively. Thus the output levels can be keyed between two levels F1 & F2 as,

$$F1 = 1/(R1*C) \quad F2 = 1/(R2*C)$$

Mark and space frequencies can be independently adjusted by the choice of timing resistors R1 & R2 respectively.

37

CIRCUIT DIAGRAM OF FSK MODULATOR:

O/P

25 K

50 K

500Ω

5Ki

0.01MF

10MF

5Ki

NC

XR2206

IP

16 K

30 K

10K

20K
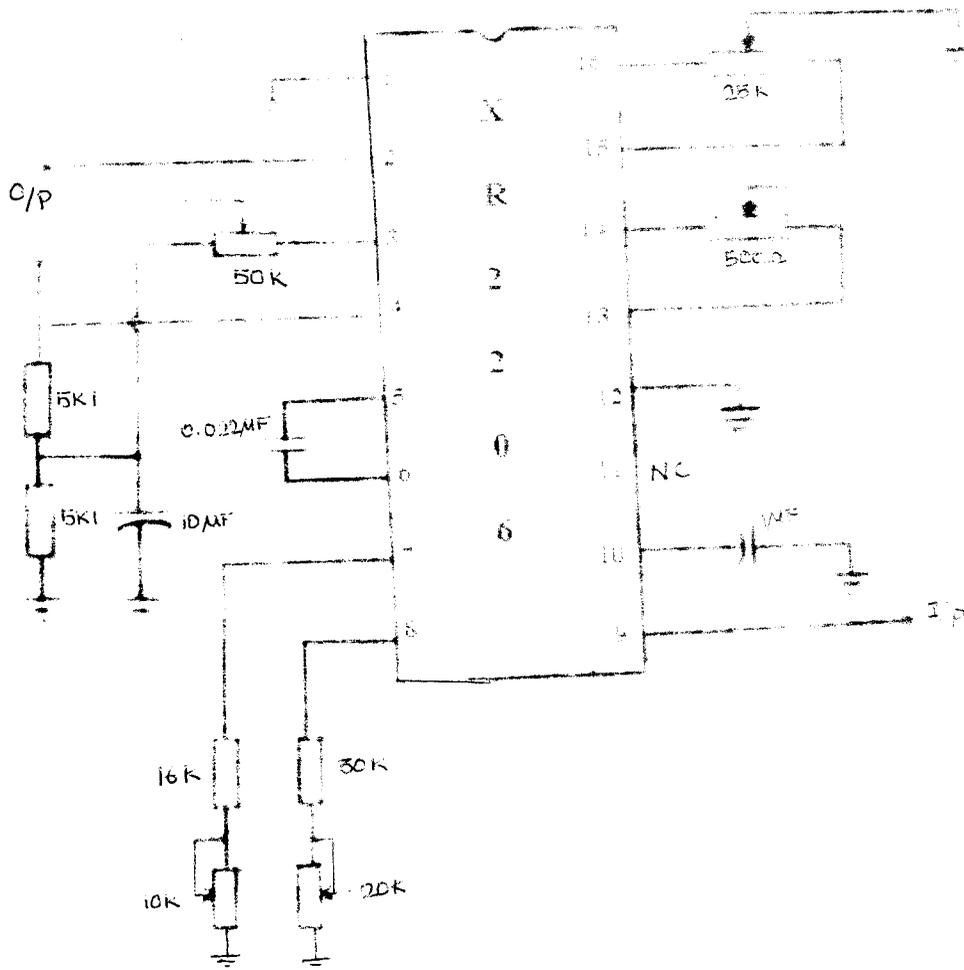
6.3 FM GENERATOR:

In our project Transistor reactance modulator generates FM. In the reactance modulator, quite a common one in use operates on the tank circuit of a Clapp-Gouriet oscillator. Provided that the correct component values are employed, any reactance modulator may be connected across the tank circuitry of any LC oscillator (not crystal of course) with one provision the oscillator used must base turned collector oscillator.

The Hartley and Colpitts (or Clapp-Gouriet) oscillator are most commonly used and each should be isolated with a buffer. The RF chokes are used to isolate various points of the circuit for alternating current while still providing a dc path.

6.4 FM RECEIVER CXA1019:

It comprises of RF amplifier, local oscillator, mixed and the IF amplifier, all built in the same chip. The input from the antenna is fed to pin 13. The crystal filter at pin 18 is used to filter the frequency below 10.7Mhz. The crystal at pin 3 serves as the local oscillator. The capacitors at pins 8,9,10& the inductance serves as LC tuning circuits, to tune the IF mixer frequency.

The LED glows only if it receives a proper signal.

CIRCUIT DIAGRAM OF FM TRANSMITTER:

6.5 XR 2211 DEMODULATOR

FEATURES:

- Wide Frequency Range, 0.01Hz to 300kHz

- Wide Supply Voltage Range, 4.5V to 20V

- HCMOS/TTL/Logic Compatibility

- FSK Demodulation, with Carrier Detection

- Wide Dynamic Range, 10mV to 3V rms.

- Adjustable Tracking Range, +1% to 80%

- Excellent Temp. Stability, +50ppm/°C, max.

GENERAL DESCRIPTION:

The XR-2211 is a monolithic phase-locked loop (PLL) system especially designed for data communications applications. It is particularly suited for FSK modem applications. It operates over a wide supply voltage range of 4.5 to 20V and a wide frequency range of 0.01Hz to 300kHz. It can accommodate analog signals between 10mV and 3V, and can interface with conventional DTL, TTL, and ECL logic families. The circuit consists of a basic PLL for tracking an input signal within the pass band, a quadrate phase detector, which provides carrier detection, and an FSK voltage compurgator, which provides FSK demodulation. External components are used to independently set center frequency, bandwidth, and output delay. An internal voltage reference proportional to the power supply is provided at an output pin. The XR-2211 is available in 14 pin packages specified for military and industrial temperature ranges.

CIRCUIT DIAGRAM OF FSK DEMODULATOR:
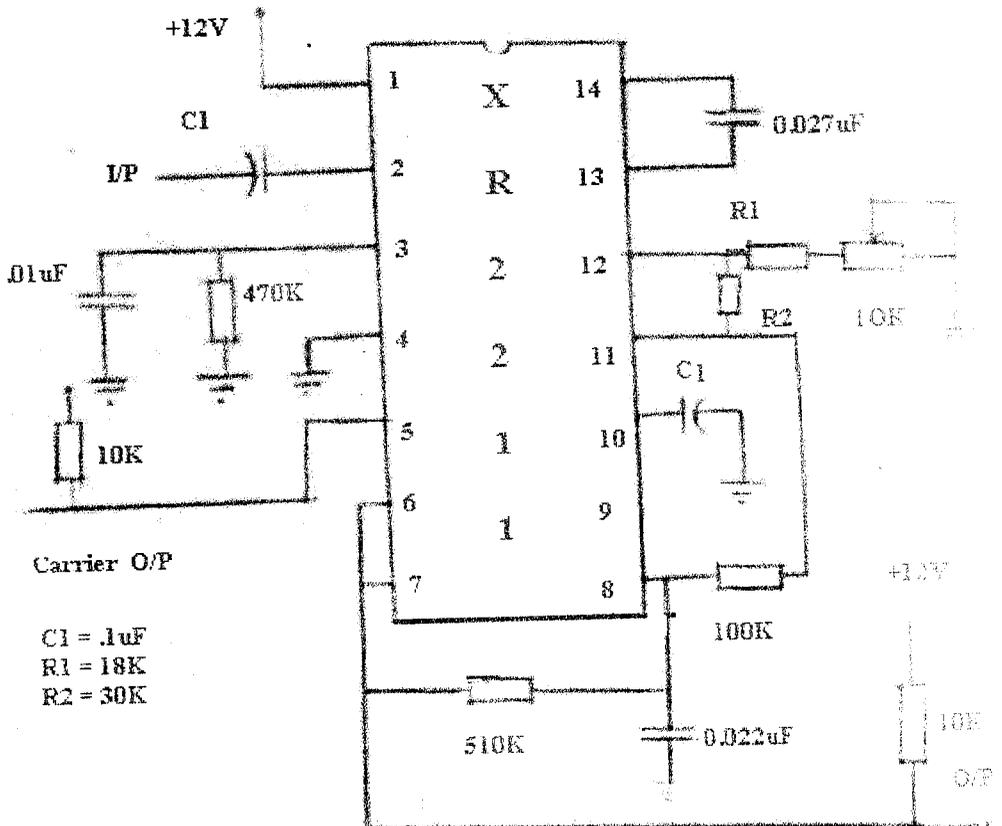
7.1 FLOWCHART:

TRANSMISSION:

RECEPTION:

```
                    ┌─────────────┐
                    │    start    │
                    └──────┬──────┘
                           │
                           ▼◄──────────────────────┐
                ╱─────────────────────╲            │
               ╱  Input  the File Name  ╲           │
              ╱─────────────────────────╲          │
                           │                        │
                           │         YES            │
                           ▼                        │
                      ╱─────────╲                   │
                     ╱ If Exists ? ╲────────────────┘
                      ╲─────────╱
                           │
                           │      NO
                           ▼
                     ╱───────────╲
                    ╱  Is space    ╲            ╭─────╮
                   ╱  available?     ╲──────────►   B   │
                    ╲───────────────╱           ╰─────╯
                           │
                           ▼
              ┌───────────────────────────┐
              │  Create and Open the File  │
              └─────────────┬─────────────┘
                            │
                            ▼
              ┌───────────────────────────┐
              │     Initialise the Port    │
              └─────────────┬─────────────┘
                            │            ╭─────╮
                            ▼◄───────────   C   │
              ┌───────────────────────────┐╰───╯
              │ Receive a Character from the Port │
              └─────────────┬─────────────┘
                            │
                            ▼
                        ╭─────╮
                        │   A   │
                        ╰─────╯
```

```
                    ( A )
                      |
                      v
      +-----------------------------+
      |      Put on the File        |
      +-----------------------------+
                      |
                      v              NO
                   /\      /\
                  /   Is EOF ?  \----------------->  ( C )
                  \            /
                   \/      \/
                      | YES
                      v
       /----------------------------/
      /      Display "RX over"      /
     /----------------------------/
                      |
                      v
      +-----------------------------+
      |       Close the file        |
      +-----------------------------+
                      |                          ( B )
                      v<--------------------------
                 /----------\
                (    Stop    )
                 \----------/
```
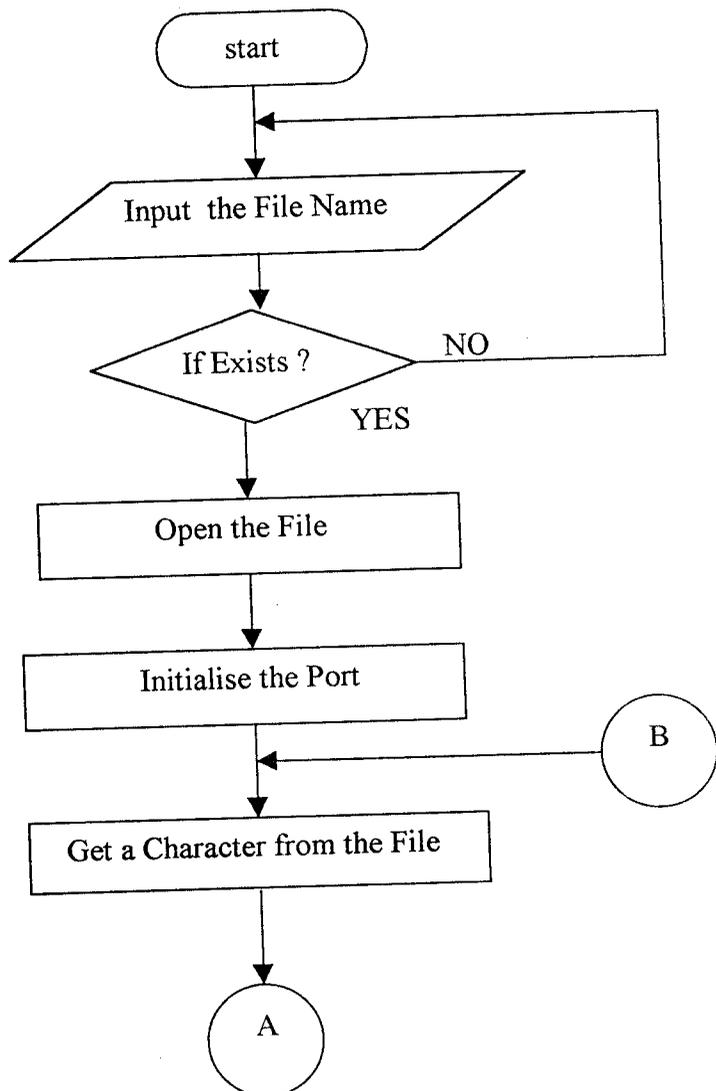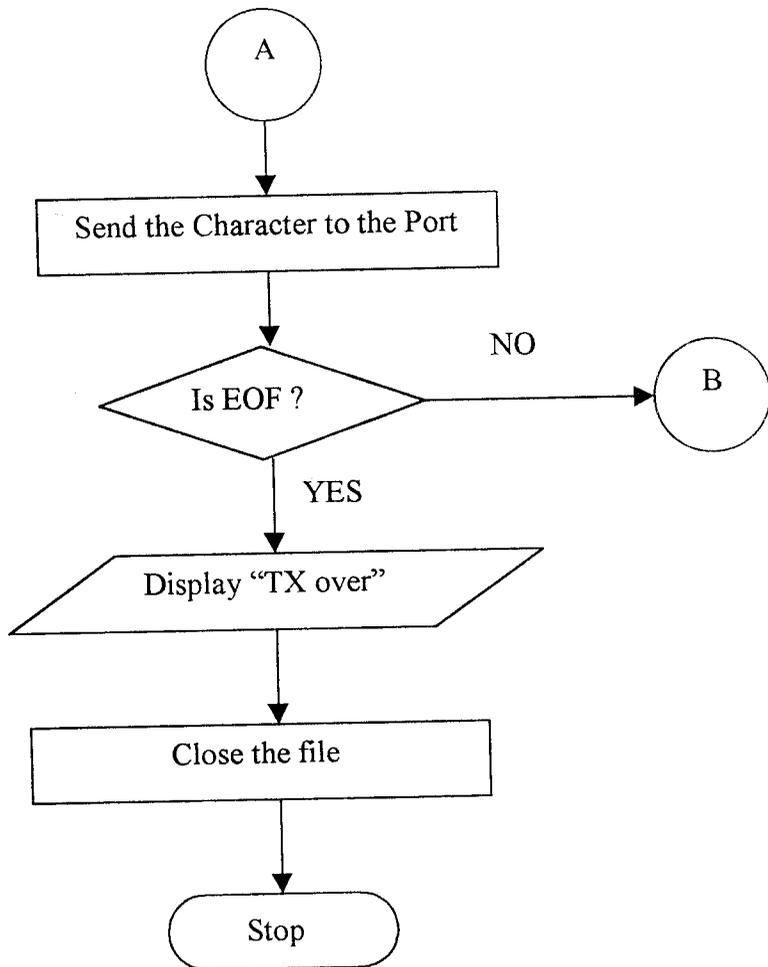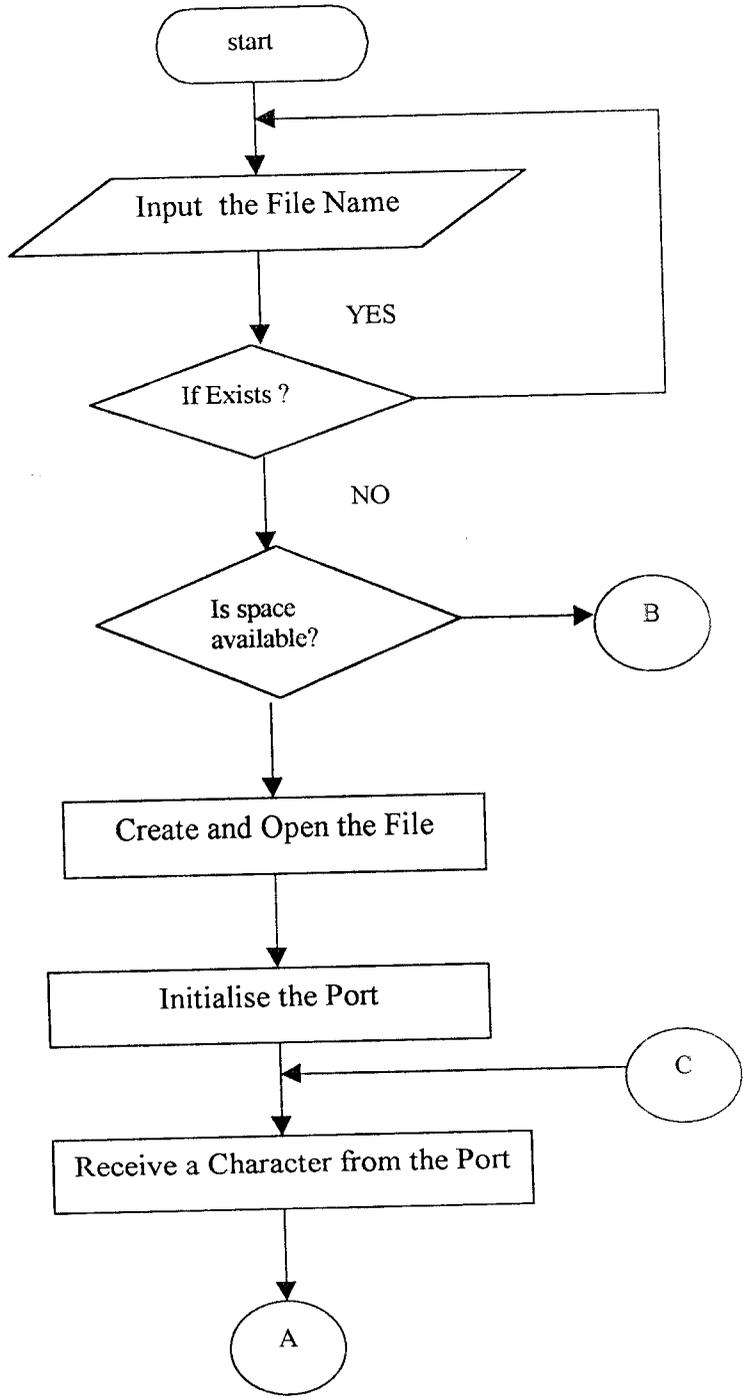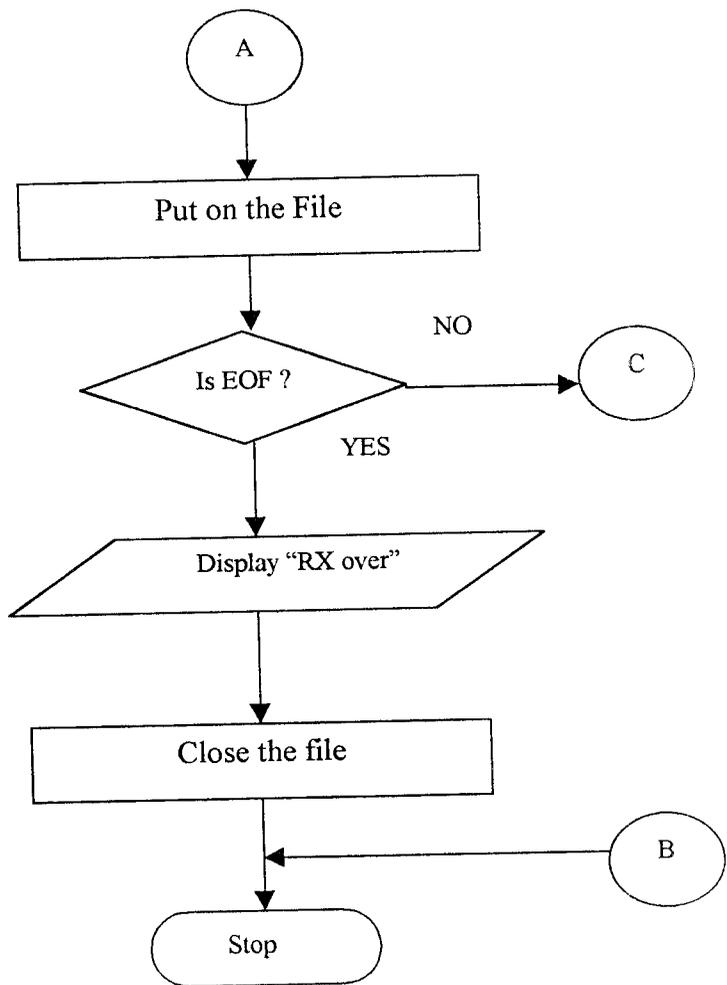
7.2 CODING:

```
/*     PROGRAM LISTING FOR PC-TO-PC COMMUNICATION IN 'C' */
/* PROGRAM FOR WIRELESS COMMUNICATION BETWEEN TWO PCs     */


# include <stdio.h> /*Header Files*/
# include <dos.h>
# include <conio.h>
# include <graphics.h>
# include <stdlib.h>
# include<string.h>
# include<fcntl.h>
# include<io.h>
# include<time.h>
# include<ctype.h>
# define DEL 25 /* Preprocessor - Delay Variable */
# define COM 0x03f8  /*0x02f8 - com2,0x03f8-com 1 */
char gra='Y';   /*Global Variance*/
int flag=0;
union REGS inregs,outregs; /*Union declaration for registers*/
FILE*fp; /*File declaration */
int status;
char temp='/n',t2;
int t1=10;


/* The main Function*/
void main(void)
{
char ch,chr,chs;/*Local Variable*/
clrscr( );
if (flag==0)
```

```c
/* splash() */ /*Calling Splash routine*/
flag++;
textcolor(4);gotoxy(26,6);
cprintf("WIRELESS COMMUNICATION");
gotoxy(34,9);textcolor(10);
cprintf("R");textcolor(7);cprintf("eceive mode");
textcolor(14);gotoxy(35,12);
cprintf("S");textcolor(7);cprintf("end mode");
textcolor(6);gotoxy(37,15);
cprintf("E");textcolor(7);cprintf("xit");
ch = getch( );/*Select Mode*/
switch(toupper(ch))
{
case'R': R:clrscr( );
   textcolor(4);gotoxy(26,6);
    cprintf("WIRELESS COMMUNICATION");
    textcolor(138);gotoxy(3,9);
    cprintf("RECEIVE MODE");
    textcolor(9);gotoxy(33,12);
    cprintf("A");textcolor(7);cprintf("lign device");
    textcolor(11);gotoxy(33,15);
    cprintf("F");textcolor(7);cprintf("ile receive");
    textcolor(6);gotoxy(36,18);
    cprintf("Q");textcolor(7);textcolor(7);cprintf("uit");
    chr = getch( );
    switch(toupper(chr))
                        {
    case 'A' : ralgn();break;
    case 'F' : f_rev( );break;
    case 'Q' : main( );
    default : clrscr( );
```

```c
            printf("Wrong Key Pressed");
            goto R;
    }
      break;
  case 'S' :S:clrscr( );
    textcolor(4);gotoxy(26,6);
    cprintf("WIRELESS COMMUNICATION");
       textcolor(142);
       gotoxy(36,9);
       cprintf("SEND MODE");
       textcolor(9);gotoxy(34,12);
        cprintf("A");textcolor(7);cprintf("lign device");
        textcolor(11);gotoxy(34,15);
        cprintf("T");textcolor(7);cprintf("ransfer file");
        textcolor(6);gotoxy(38,18);
        cprintf("Q");textcolor(7);cprintf("uit");
        chs = getch( );
        switch(toupper(chs))
          {
                              case'A' : salgn( );break;
        case'T' : f_snd( );break;
        case'Q' : main( );
        default :clrscr( );
         printf("Wrong Key Pressed");
         goto S;
         }
          break;
                              case'E' :clrscr( );
        textcolor(143);
        gotoxy(35,13);
        cprintf("GOOD BYE");
```

```c
exit(1);
default : clrscr( );
printf("Wrong Key Pressed");
main( );
return;
}
}
/*Function for receiving (For Device Alignment)/*
ralgn(void)
{
char st = ' '; /*Local variables*/
gotoxy(30,2);
textcolor(10);
cprintf("RECEIVE MODE :");
textcolor(9);cprintf("ALIGN DEVICE");
printf("\n");
initial(); /*Call Initialisation routine */
loop:  if (!kbhit())
                {
            if(st == 0x04) /* Check for end of Transmission */

    {
        clrscr();
                    textcolor(140);
        gotoxy(30,12);
    cprintf("ALIGNED PROPERLY");
    gotoxy(48,24);
    printf("Press any key to quit.");
    getch( );
    main( ); /* Got to main function after aligning properly */

    }
    status = inp(0x3fd); /* Checking status at com 1 port */
```

```c
if((status & 0x01)= =0x00) /* Checking for data Ready */
goto loop;
                          else if(!kbhit( ))
                  {
st = inp(COM); /* Get character from com 1 port till */
printf("%c",st); /* key hit or end of transmission */
goto loop;
                  }
                  else
main();/* call main function if key hit*/
                  }
          return;


/*Function for File Receive*/
f_rev( )
{
int flag=0,bytecount=0,count;  /* Local Variables */
float ot = 0.00,nt = 0.00;
char ch,st[55000],fnm[30];
clrscr();
initial();/* calling initialization routine*/
ot = clock()/18.2;/* calculate exec time in secs from start of the program*/
gotoxy(2,2);
printf("FILE NAME ?");
fp=fopen(gets(fnm),"wb");/* get file name in write mode*/
gotoxy(26,10);
printf("(ready for )RECEIVING DATA");
gotoxy(50,24);
textcolor(138);
cprintf("don't KEY IN may loss data");
loop: nt = clock()/18.2;/* calculate exec time in secs from start of the program*/
```

```c
status = inp(0x3FD);/*get character from com 1 port*/
if ((status & 0x01)==0x00)/* check for data ready*/
{
/* check for no data reception for 5 secs after start of reception if no data is received
continue other process*/
if(( bytecount >0)&&(nt-ot)>5.0)
    {
                            clrscr();
                            for(count=0; count<flag;count++)

    {
                            gotoxy(26,10);
    textcolor(11);
    cprintf("saving data in ");
    gotoxy(43,10);
    textcolor(12);
    cprintf("%s",fnm);
    /* dump the data received in a file*/
    fprintf(fp,"%c",st[count]);
    }
    fclose(fp);
    gotoxy(26,13);
    textcolor(11);
    cprintf("file %s of %d bytes created ",fnm,count);
    gotoxy(50,24);
    textcolor(7);


    cprintf("Press any key to quit.");
    getch();
    main();
                }
```

```c
                    goto loop;
                         }
              else if(!kbhit())
{
   st[flag] = inp(COM);/*Get character from
Com1 port*/
   flag++;
   bytecount++;
   ot = clock()/18.2;/*Calculate exec time during receiving*/
    goto loop;
                     }
                   else
                    {
/*If transmission is cut terminate abnormally */
   clrscr();
    for(count=0;count<flag;count++)
    {
                  gotoxy(26,3);
          textcolor(140);
          cprintf("TERMINATED ABNORMALLY");
          gotoxy(26,10);
          textcolor(11);
          cprintf("Saving data in");
          textcolor(12);
          cprintf("%s",fnm);
          fprintf(fp,"%c",st[count]);
                   }
       fclose(fp);
       gotoxy(26,13);
       textcolor(11);
          cprintf(" File %s of %d bytes created",fnm,count);
```

```c
sleep(5);
main();/*go to main after dumping in file*/
            }
            return 0;

}
/* Function for send align( for device alignment)*/
salgn(void)
{
int flag=0;/*Local Variables*/
char st[127];
clrscr();
initial();
textcolor(14);
cprintf("type the sentence( < 127 chars)");
puts("\n");
gets(st); /* Get string to send*/
loop:status=inp(0X3FD);/*get com1 port status*/
if((status & 0x20) == 0x00)/* Check Transfer holding register empty*/
goto loop;
else
                    do
                    {
    if(!kbhit()) /*Check for key hit*/
                        {
                            outport(COM,0X0D);/*Send carriage return*/
    outport(COM,0X0A);/*send line feed*/
                            if(flag==strlen(st)) /* Check for length of string*/

    {
        printf("\n");
    flag=0;
     outport(COM,0x0D);
```

```
/*send carriage return*/
delay(5);
outport(COM,0X0A);/* Send carriage return*/
                        delay(5);
                    }
                                else
                          {
outport(COM,st[flag]);/* Send character to com1 port*/
printf("%c",st[flag]);
flag++;
 delay(DEL);
                        }
}
                if(!kbhit())/*check key hit*/
                                        {

delay(1);
outport(COM,0x04);/*Send End of transmission*/
main();
                              }
                          }
 while(!kbhit());
                    }
 /*Function for file transfer*/
 f_snd()
 {
 int flag=0,count=0,fl; /* Local Variables */
 char ch,st[55000],fnm[20];
 clrscr();
 initial();/* Calling Initialisation Routine */
 gotoxy(2,2);
 printf("File Name?:");
```

```c
fp=fopen(gets(fnm),"rb");/* Get file name to be sent */
if(fp==NULL)
  {
  clrscr();
  gotoxy(35,13);
  printf("FILE NOT FOUND!");
  delay(1000);
  main();
              }
else
{
  fl=filelength(5); /*Calculate file length*/
  gotoxy(23,20);
  printf("File being transferred has % u bytes",fl);
  do
                                {
     ch = fgetc(fp);
     st[count]=ch;
     count++;
  }
   while(count<=fl);
  }
  fclose(fp);
  loop: status = inp(0X3FD);/*Check com 1 port status */
       if((status & 0x20)==0x00)/* Check Transfer holding register empty */
       goto loop;
       else
                          do
                    {
    if(flag==fl)/*check for file length */
          {
```

```
gotoxy(50,24);
printf("Press any ket to exit!");
getch();
main(); } /*Call mainfunction */
else
{
outport(COM,st[flag]);/*send each character in to file*/
printf("\t%004x",st[flag]);
flag++;
delay(DEL);
}
                }
                        while(!kbhit());/*Check for key hit*/
                        }
                /*initialisation funtion*/
initial()
                {       inregs.h.ah=0;/*Initialisation of  port/*
                        inregs.h.al = 0X63; /* Baudrate , Parity, Databits,Stopbit(s)*/
                        inregs.x.dx = 0;/*Select port COM1*/
                        int86(0x14,&inregs,&outregs);/*Complete
                        Communication service Interrupt*/
                }
/*Function for Splash screen*/
splash(void)
                {
int d=DETECT,m,j,i;
struct palettetype pal;/*structure for palette colour*/
initgraph(&d,&m," " ); /*Initialisation for splash screen*/
getpalette(&pal); /* Structure for palette colours*/
initgraph(&d,&m," ");/*Initialisation for splash screen */
getpalette(&pal); /*get palette colours*/
```

```
for(i=0;i<=pal.size;i++)
setrgbpalette(pal.colors[i],i*5,i*4,i*4);/*Combination of RGB colours*/
setfillstyle(8,8);
setcolor(15);
settextstyle(1,0,4);
setbkcolor(4);
   for(i=0;i<17;i++)  /* Writing text with RGB palatte colors*/
               {
setcolor(i);
         outtextxy(45+i,200+i,"PC to PC wireless Communication");
               }
                  sleep(1);
      cleardevice();
for(i=0;i<17;i++)   /* Writing text with RGB palette colors*/
               {
   setcolor(i);
                     outtextxy(175+i,200+i,"MOSTEC LECTRONICS");
               }
               sleep(1);
               cleardevice();
               for(i= 0;i<17;i++)/* writing text with rgb pallet colors*/
               {
setcolor(i);
 outtextxy(160+i,175+i," KCT");
 }
  sleep(1);
   cleardevice();
    closegraph();
                  }
 /*------end-----*/
```

CONCLUSION

This project gives the advantage of all forms of communication possible between any two systems without using cables and floppy diskettes, which are used for, file transaction. Also this project overcomes the few demerits of the infrared wireless communication method.

Also message of any kind can be sent from one system to another thus allowing session between the users. If printing option is provided files of any type can be directly sent to the printer or hard copy.

This project can also be extended in such a way that many computers may be connected to one another so that they form a network and in such a case the resources available in one system can be shared with others. Each system in the network can act as a server at different times and at different situations.

General Description

The MAX202E, MAX213E, MAX232E/MAX241E line drivers/receivers are designed for RS-232 and V.28 communications in harsh environments. Each transmitter output and receiver input is protected against ±15kV electrostatic discharge (ESD) shocks, without latchup. The various combinations of features are outlined in the Selection Guide. The drivers and receivers for all ten devices meet all EIA/TIA-232E and CCITT V.28 specifications at data rates up to 120kbps, when loaded in accordance with the EIA/TIA-232E specification.

The MAX211E/MAX213E/MAX241E are available in 28-pin SO packages, as well as a 28-pin SSOP that uses 60% less board space. The MAX202E/MAX232E come in 16-pin narrow SO, wide SO, and DIP packages. The MAX203E comes in a 20-pin DIP/SO package, and needs no external charge-pump capacitors. The MAX205E comes in a 24-pin wide DIP package, and also eliminates external charge-pump capacitors. The MAX206E/MAX207E/MAX208E come in 24-pin SO, SSOP, and narrow DIP packages. The MAX232E/MAX241E operate with four 1µF capacitors, while the MAX202E/MAX206E/MAX207E/MAX208E/MAX211E/MAX213E operate with four 0.1µF capacitors, further reducing cost and board space.
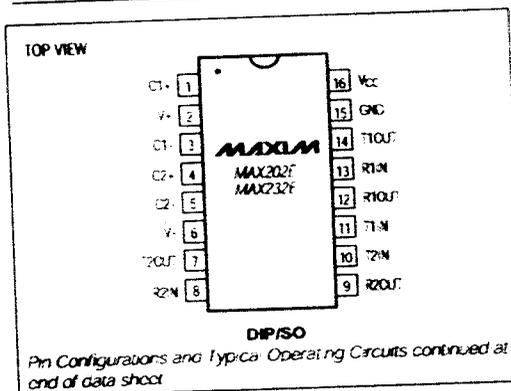
Applications

Notebook, Subnotebook, and Palmtop Computers

Battery-Powered Equipment

Hand-Held Equipment

Ordering Information appears at end of data sheet.

Features

- ♦ **ESD Protection for RS-232 I/O Pins:**
  ±15kV—Human Body Model
  ±8kV—IEC1000-4-2, Contact Discharge
  ±15kV—IEC1000-4-2, Air-Gap Discharge
- ♦ **Latchup Free (unlike bipolar equivalents)**
- ♦ **Guaranteed 120kbps Data Rate—LapLink™ Compatible**
- ♦ **Guaranteed 3V/µs Min Slew Rate**
- ♦ **Operate from a Single +5V Power Supply**

Pin Configurations

TOP VIEW

C1+	1	16	Vcc
V+	2	15	GND
C1-	3	14	T1OUT
C2+	4	13	R1IN
C2-	5	12	R1OUT
V-	6	11	T1IN
T2OUT	7	10	T2IN
R2IN	8	9	R2OUT

MAX202E
MAX232E

DIP/SO

Pin Configurations and Typical Operating Circuits continued at end of data sheet.

Selection Guide

PART	No. of RS-232 DRIVERS	No. of RS-232 RECEIVERS	RECEIVERS ACTIVE IN SHUTDOWN	No. of EXTERNAL CAPACITORS	LOW-POWER SHUTDOWN	TTL THREE-STATE
MAX202E	2	2	0	4 (0.1µF)	No	No
MAX203E	2	2	0	None	No	No
MAX205E	5	5	0	None	Yes	Yes
MAX206E	4	3	0	4 (0.1µF)	Yes	Yes
MAX207E	5	3	0	4 (0.1µF)	No	No
MAX208E	4	4	0	4 (0.1µF)	No	No
MAX211E	4	5	0	4 (0.1µF)	Yes	Yes
MAX213E	4	5	2	4 (0.1µF)	Yes	Yes
MAX232E	2	2	0	4 (1µF)	No	No
MAX241E	4	5	0	4 (1µF)	Yes	Yes

LapLink is a registered trademark of Traveling Software, Inc.

±15kV ESD-Protected, +5V RS-232 Transceivers

ABSOLUTE MAXIMUM RATINGS

VCC .... 0.3V to +6V
V+ .... (VCC 0.3V) to +14V
V- .... -14V to +0.3V
Input Voltages
  T_IN .... 0.3V to (V+ - 0.3V)
  R_IN .... ±30V
Output Voltages
  T_OUT .... (V- 0.3V) to (V+ + 0.3V)
  R_OUT .... 0.3V to (VCC + 0.3V)
Short Circuit Duration T_OUT .... Continuous
Continuous Power Dissipation (T_A = +70°C)
  16 Pin Plastic DIP (derate 10.53mW/°C above +70°C) ... 842mW
  16 Pin Narrow SO (derate 8.70mW/°C above +70°C) ... 696mW
  16 Pin Wide SO (derate 9.52mW/°C above +70°C) ... 762mW
  20 Pin Plastic DIP (derate 11.11mW/°C above +70°C) ... 889mW
  20 Pin SO (derate 10.00mW/°C above +70°C) 800mW
  24 Pin Narrow Plastic DIP
    (derate 13.33mW/°C above +70°C) 1.07W
  24 Pin Wide Plastic DIP
    (derate 14.29mW/°C above +70°C) 1.14W
  24 Pin SO (derate 11.76mW/°C above +70°C) 941mW
  24 Pin SSOP (derate 8.00mW/°C above +70°C) 640mW
  28 Pin SO (derate 12.50mW/°C above +70°C) 1W
  28 Pin SSOP (derate 9.52mW/°C above +70°C) 762mW
Operating Temperature Ranges
  MAX2__EC_ .... 0°C to +70°C
  MAX2__EE_ .... -40°C to +85°C
  MAX2__EE_ .... -55°C to +155°C
Storage Temperature Range .... -55°C to +165°C
Lead Temperature (soldering) 10sec. .... +300°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

(Vcc = +5V ±10% for MAX202E/206E/207E/208E/211E/213E. C1–C4 = 1µF for MAX232E/241E. Vcc = +5V ±5% for MAX203E/205E/207E. C1–C4 = 0.1µF for MAX202E/206E/207E/208E/211E/213E. C1–C4 = 1µF for MAX232E/241E; TA = TMIN to TMAX, unless otherwise noted. Typical values are at TA = +25°C.)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
DC CHARACTERISTICS							
Vcc Supply Current	Icc	No load, TA = +25°C	MAX202E/203E		8	15	mA
			MAX205E–208E			20	
			MAX211E/213E		4	20	
			MAX232E		5	10	
			MAX241E		7	15	
Shutdown Supply Current		TA = +25°C, Figure 1	MAX205E/206E			10	µA
			MAX211E/241E			10	
			MAX213E		15	50	
LOGIC							
Input Pull Up Current		T_IN = 0V (MAX205E–208E/211E/213E/241E)			15	200	µA
Input Leakage Current		T_IN = 0V to VCC (MAX202E/203E/232E)				±10	µA
Input Threshold Low	V_IL	T_IN, EN, SHDN (MAX213E) or EN, SHDN (MAX205E–208E/211E/241E)				0.8	V
Input Threshold High	V_IH	T_IN		2.0			V
		EN, SHDN (MAX213E) or EN, SHDN (MAX205E–208E/211E/241E)		2.4			
Output Voltage Low	V_OL	R_OUT, Iout = 3.2mA (MAX202E/203E/232E) or Iout = 1.6mA (MAX205E/208E/211E/213E/241E)				0.4	V
Output Voltage High	V_OH	R_OUT, Iout = 1.0mA		3.5	VCC - 0.4		V
Output Leakage Current		EN = VCC, EN = 0V, 0V ≤ ROUT ≤ VCC, MAX205E–208E/211E/213E/241E outputs disabled			±0.05	±10	µA

MAXIM

2

ELECTRICAL CHARACTERISTICS (continued)

(VCC = +5V ±10% for MAX202E/206E/208E/211E/213E/232E/241E, VCC = +5V ±5% for MAX203E/205E/207E, C1–C4 = 0.1µF for MAX202E/206E/207E/208E/211E/213E, C1–C4 = 1µF for MAX232E/241E; TA = TMIN to TMAX, unless otherwise noted. Typical values are at TA = +25°C.)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
EIA/TIA-232E RECEIVER INPUTS							
Input Voltage Range				-30		30	V
Input Threshold Low			All parts, normal operation	0.8	1.2		V
		TA = +25°C VCC = 5V	MAX213E, SHDN = 0V EN = VCC	0.6	1.5		
Input Threshold High			All parts, normal operation		1.7	2.4	V
		TA = +25°C VCC = 5V	MAX213E (R4, R5), SHDN = 0V, EN = VCC		1.5	2.4	
Input Hysteresis		VCC = 5V, no hysteresis in shutdown		0.2	0.5	1.0	V
Input Resistance		TA = +25°C, VCC = 5V		3	5	7	kΩ
EIA/TIA-232E TRANSMITTER OUTPUTS							
Output Voltage Swing		All drivers loaded with 3kΩ to ground (Note 1)		±5	±9		V
Output Resistance		VCC = V+ = V- = 0V, VOUT = ±2V		300			Ω
Output Short Circuit Current					±10	±60	mA
TIMING CHARACTERISTICS							
Maximum Data Rate		RL = 3kΩ to 7kΩ, CL = 50pF to 1000pF, one transmitter switching		120			kbps
Receiver Propagation Delay	tPLHR, tPHLR	CL = 150pF	All parts, normal operation		0.5	1.0	µs
			MAX213E (R4, R5), SHDN = 0V, EN = VCC		4	40	
Receiver Output Enable Time		MAX205E/206E/211E/213E/241E normal operation, Figure 2			600		ns
Receiver Output Disable Time		MAX205E/206E/211E/213E/241E normal operation, Figure 2			200		ns
Transmitter Propagation Delay	tPLHT, tPHLT	RL = 3kΩ, CL = 2500pF, all transmitters loaded			2		µs
Transition Region Slew Rate		TA = +25°C, VCC = 5V, RL = 3kΩ to 7kΩ CL = 50pF to 1000pF, measured from -3V to +3V or +3V to -3V, Figure 3		3	6	30	V/µs
ESD PERFORMANCE: TRANSMITTER OUTPUTS, RECEIVER INPUTS							
ESD Protection Voltage		Human Body Model			±15		kV
		IEC1000-4-2, Contact Discharge			±8		
		IEC1000-4-2, Air-Gap Discharge			±15		

Note 1: MAX211EE_ tested with VCC = +5V ±5%.

MAXIM

Table 3. DB9 Cable Connections Commonly Used for EIA/TIAE-232E and V.24 Asynchronous Interfaces

PIN		CONNECTION
1	Received Line Signal Detector (sometimes called Carrier Detect DCD)	Handshake from DCE
2	Receive Data (RD)	Data from DCE
3	Transmit Data (TD)	Data from DTE
4	Data Terminal Ready	Handshake from DTE
5	Signal Ground	Reference point for signals
6	Data Set Ready (DSR)	Handshake from DCE
7	Request to Send (RTS)	Handshake from DTE
8	Clear to Send (CTS)	Handshake from DCE
9	Ring Indicator	Handshake from DCE

Pin Configurations and Typical Operating Circuits (continued)

PIN NUMBERS ON TYPICAL OPERATING CIRCUIT REFER TO DIP/SO PACKAGE. NOT LCC
1.0µF CAPACITORS MAX241 ONLY

MAXIM

14

EXAR
...the analog plus company™

FEATURES

- Low-Sine Wave Distortion, 0.5%, Typical
- Excellent Temperature Stability, 20ppm/°C, Typ.
- Wide Sweep Range, 2000:1, Typical
- Low-Supply Sensitivity, 0.01%V, Typ.
- Linear Amplitude Modulation
- TTL Compatible FSK Controls
- Wide Supply Range, 10V to 26V
- Adjustable Duty Cycle, 1% TO 99%

APPLICATIONS

- Waveform Generation
- Sweep Generation
- AM/FM Generation
- V/F Conversion
- FSK Generation
- Phase-Locked Loops (VCO)

GENERAL DESCRIPTION

The XR-2206 is a monolithic function generator integrated circuit capable of producing high quality sine, square, triangle, ramp, and pulse waveforms of high-stability and accuracy. The output waveforms can be both amplitude and frequency modulated by an external voltage. Frequency of operation can be selected externally over a range of 0.01Hz to more than 1MHz.

The circuit is ideally suited for communications, instrumentation, and function generator applications requiring sinusoidal tone, AM, FM, or FSK generation. It has a typical drift specification of 20ppm/°C. The oscillator frequency can be linearly swept over a 2000:1 frequency range with an external control voltage, while maintaining low distortion.

ORDERING INFORMATION

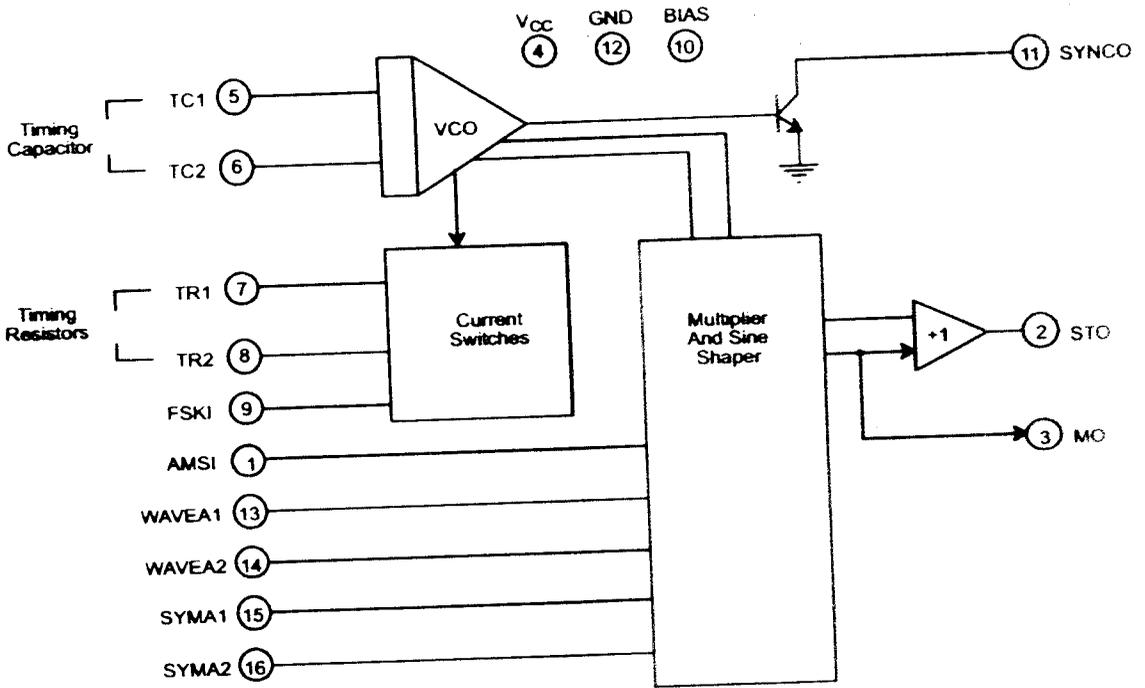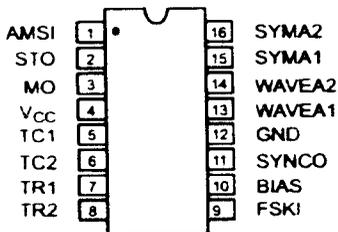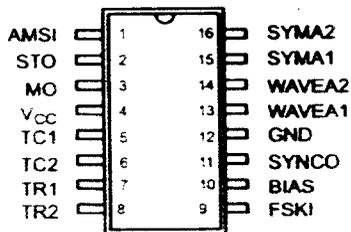| Part No.	Package	Operating Temperature Range
XR-2206M	16 Lead 300 Mil CDIP	-55°C to +125°C
XR-2206P	16 Lead 300 Mil PDIP	-40°C to +85°C
XR-2206CP	16 Lead 300 Mil PDIP	0°C to +70°C
XR-2206D	16 Lead 300 Mil JEDEC SOIC	0°C to +70°C

TOM

Figure 1.   XR-2206 Block Diagram

EXAR

```
AMSI   1      16  SYMA2
STO    2      15  SYMA1
MO     3      14  WAVEA2
Vcc    4      13  WAVEA1
TC1    5      12  GND
TC2    6      11  SYNCO
TR1    7      10  BIAS
TR2    8       9  FSKI
```

16 Lead PDIP, CDIP (0.300")

```
AMSI   1      16  SYMA2
STO    2      15  SYMA1
MO     3      14  WAVEA2
Vcc    4      13  WAVEA1
TC1    5      12  GND
TC2    6      11  SYNCO
TR1    7      10  BIAS
TR2    8       9  FSKI
```

16 Lead SOIC (Jedec, 0.300")

PIN DESCRIPTION

Pin #	Symbol	Type	Description
1	AMSI	I	Amplitude Modulating Signal Input.
2	STO	O	Sine or Triangle Wave Output.
3	MO	O	Multiplier Output.
4	V_{CC}		Positive Power Supply.
5	TC1	I	Timing Capacitor Input.
6	TC2	I	Timing Capacitor Input.
7	TR1	O	Timing Resistor 1 Output.
8	TR2	O	Timing Resistor 2 Output.
9	FSKI	I	Frequency Shift Keying Input.
10	BIAS	O	Internal Voltage Reference.
11	SYNCO	O	Sync Output. This output is a open collector and needs a pull up resistor to V_{CC}.
12	GND		Ground pin.
13	WAVEA1	I	Wave Form Adjust Input 1.
14	WAVEA2	I	Wave Form Adjust Input 2.
15	SYMA1	I	Wave Symetry Adjust 1.
16	SYMA2	I	Wave Symetry Adjust 2.

TQM

XR-2206

XP EXAR

DC ELECTRICAL CHARACTERISTICS

Test Conditions: Test Circuit of *Figure 2* Vcc = 12V, T_A = 25°C, C = 0.01μF, R_1 = 100kΩ, R_2 = 10kΩ, R_3 = 25kΩ
Unless Otherwise Specified. S_1 open for triangle, closed for sine wave.

Parameters	XR-2206M/P Min.	Typ.	Max.	XR-2206CP/D Min.	Typ.	Max.	Units	Conditions
General Characteristics								
Single Supply Voltage	10		26	10		26	V	
Split-Supply Voltage	±5		±13	±5		±13	V	
Supply Current		12	17		14	20	mA	$R_1 \geq$ 10kΩ
Oscillator Section								
Max. Operating Frequency	0.5	1		0.5	1		MHz	C = 1000pF, R_1 = 1kΩ
Lowest Practical Frequency		0.01			0.01		Hz	C = 50μF, R_1 = 2MΩ
Frequency Accuracy		±1	±4		±2		% of f_o	f_o = 1/R_1C
Temperature Stability Frequency		+10	+50		+20		ppm/°C	0°C $\leq T_A \leq$ 70°C $R_1 = R_2$ = 20kΩ
Sine Wave Amplitude Stability[2]		4800			4800		ppm/°C	
Supply Sensitivity		0.01	0.1		0.01		%/V	V_{LOW} = 10V, V_{HIGH} = 20V, $R_1 = R_2$ = 20kΩ
Sweep Range	1000:1	2000:1			2000:1		$f_H = f_L$	f_H @ R_1 = 1kΩ f_L @ R_1 = 2MΩ
Sweep Linearity								
10:1 Sweep		2			2		%	f_L = 1kHz, f_H = 10kHz
1000:1 Sweep		8			8		%	f_L = 100Hz, f_H = 100kHz
FM Distortion		0.1			0.1		%	±10% Deviation
Recommended Timing Components								
Timing Capacitor: C	0.001		100	0.001		100	μF	*Figure 5*
Timing Resistors: R_1 & R_2	1		2000	1		2000	kΩ	
Triangle Sine Wave Output[1]								*Figure 3*
Triangle Amplitude		160			160		mV/kΩ	*Figure 2*, S_1 Open
Sine Wave Amplitude	40	60	80		60		mV/kΩ	*Figure 2*, S_1 Closed
Max. Output Swing		6			6		Vp-p	
Output Impedance		600			600		Ω	
Triangle Linearity		1			1		%	
Amplitude Stability		0.5			0.5		dB	For 1000:1 Sweep
Sine Wave Distortion								
Without Adjustment		2.5			2.5		%	R_1 = 30kΩ
With Adjustment		0.4	1.0		0.5	1.5	%	See *Figure 7* and *Figure 8*

Notes
[1] *Output amplitude is directly proportional to the resistance, R_3, on Pin 3. See Figure 3.*
[2] *For maximum amplitude stability, R_3 should be a positive temperature coefficient resistor.*
***Bold face parameters** are covered by production test and guaranteed over operating temperature range.*

TOM

Output Amplitude:

Maximum output amplitude is inversely proportional to the external resistor, R_3, connected to Pin 3 (see *Figure 3*). For sine wave output, amplitude is approximately 60mV peak per $k\Omega$ of R_3; for triangle, the peak amplitude is approximately 160mV peak per $k\Omega$ of R_3. Thus, for example, $R_3 = 50k\Omega$ would produce approximately 13V sinusoidal output amplitude.

Amplitude Modulation:

Output amplitude can be modulated by applying a dc bias and a modulating signal to Pin 1. The internal impedance at Pin 1 is approximately 100kΩ. Output amplitude varies linearly with the applied voltage at Pin 1, for values of dc bias at this pin, within 14 volts of $V_{CC}/2$ as shown in *Figure 6*. As this bias level approaches $V_{CC}/2$, the phase of the output signal is reversed, and the amplitude goes through zero. This property is suitable for phase-shift keying and suppressed-carrier AM generation. Total dynamic range of amplitude modulation is approximately 55dB.

CAUTION: AM control must be used in conjunction with a well-regulated supply, since the output amplitude now becomes a function of V_{CC}.
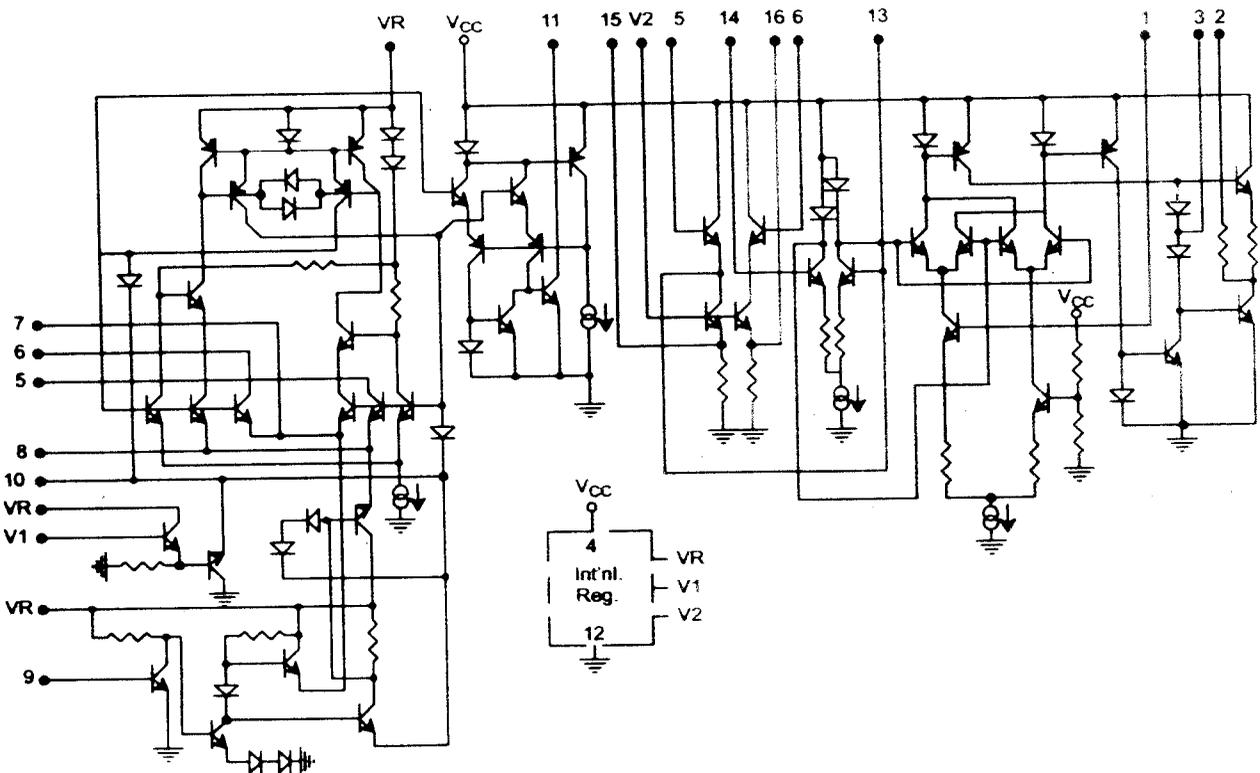

Figure 15. Equivalent Schematic Diagram

XR-2211

FSK Demodulator/Tone Decoder

EXAR *...the analog plus company*[TM]

FEATURES

- Wide Frequency Range, 0.01Hz to 300kHz
- Wide Supply Voltage Range, 4.5V to 20V
- HCMOS/TTL/Logic Compatibility
- FSK Demodulation, with Carrier Detection
- Wide Dynamic Range, 10mV to 3V rms
- Adjustable Tracking Range, ±1% to 80%
- Excellent Temp. Stability, +50ppm/°C, max.

APPLICATIONS

- Caller Identification Delivery
- FSK Demodulation
- Data Synchronization
- Tone Decoding
- FM Detection
- Carrier Detection

GENERAL DESCRIPTION

The XR-2211 is a monolithic phase-locked loop (PLL) system especially designed for data communications applications. It is particularly suited for FSK modem applications. It operates over a wide supply voltage range of 4.5 to 20V and a wide frequency range of 0.01Hz to 300kHz. It can accommodate analog signals between 10mV and 3V, and can interface with conventional DTL, TTL, and ECL logic families. The circuit consists of a basic PLL for tracking an input signal within the pass band, a quadrature phase detector which provides carrier detection, and an FSK voltage comparator which provides FSK demodulation. External components are used to independently set center frequency, bandwidth, and output delay. An internal voltage reference proportional to the power supply is provided at an output pin.

The XR-2211 is available in 14 pin packages specified for military and industrial temperature ranges.

ORDERING INFORMATION

Part No.	Package	Operating Temperature Range
XR-2211M	14 Pin CDIP (0.300")	-55°C to +125°C
XR-2211N	14 Pin CDIP (0.300")	-40°C to +85°C
XR-2211P	14 Pin PDIP (0.300")	-40°C to +85°C
XR-2211ID	14 Lead SOIC (Jedec, 0.150")	-40°C to +85°C

EXAR Corporation, 48720 Kato Road, Fremont, CA 94538 ◆ (510) 668-7000 ◆ FAX (510) 668-7017
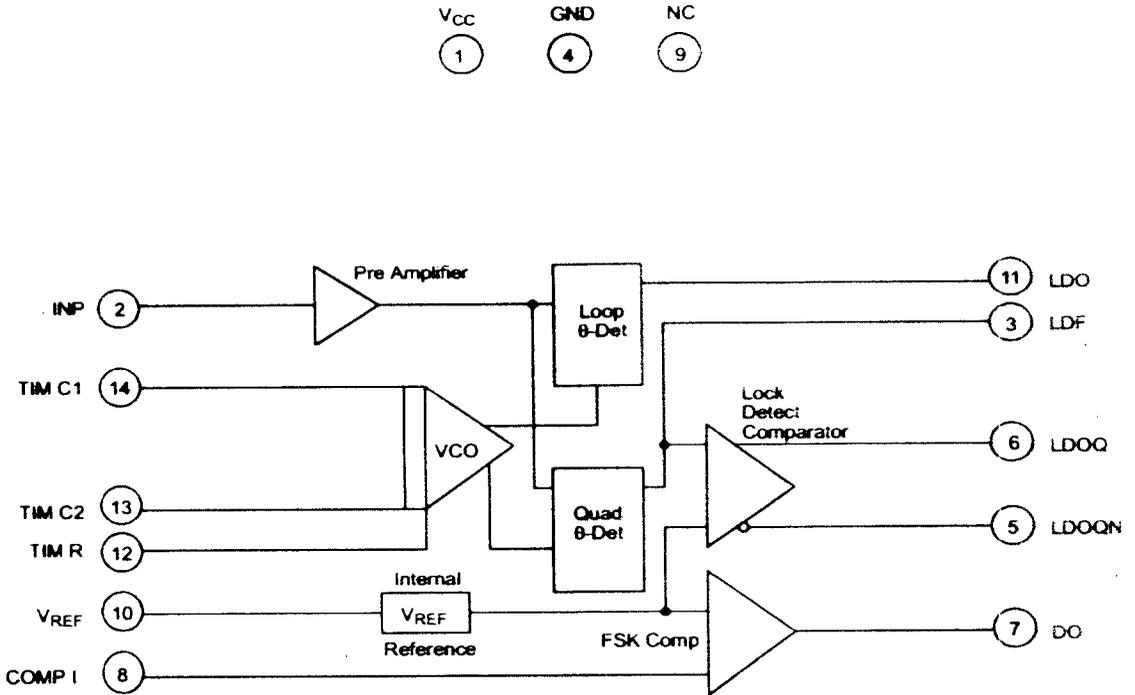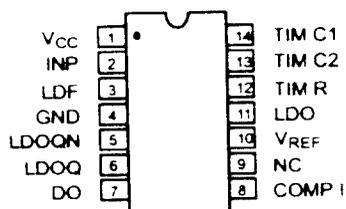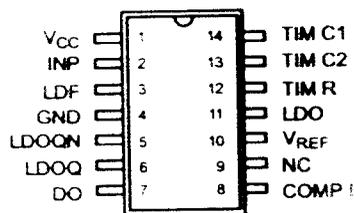
XR-2211

EXAR

BLOCK DIAGRAM

Figure 1. XR-2211 Block Diagram

EXAR

PIN CONFIGURATION

14 Lead CDIP, PDIP (0.300")

14 Lead SOIC (Jedec, 0.150")

PIN DESCRIPTION

Pin #	Symbol	Type	Description
1	V_{CC}		Positive Power Supply.
2	INP	I	Receive Analog Input.
3	LDF	O	Lock Detect Filter.
4	GND		Ground Pin.
5	LDOQN	O	Lock Detect Output Not. This output will be low if the VCO is in the capture range.
6	LDOQ	O	Lock Detect Output. This output will be high if the VCO is in the capture range.
7	DO	O	Data Output. Decoded FSK output.
8	COMP I	I	FSK Comparator Input.
9	NC		Not Connected.
10	V_{REF}	O	Internal Voltage Reference. The value of V_{REF} is $V_{CC}/2$ - 650mV.
11	LDO	O	Loop Detect Output. This output provides the result of the quadrature phase detection.
12	TIM R	I	Timing Resistor Input. This pin connects to the timing resistor of the VCO.
13	TIM C2	I	Timing Capacitor Input. The timing capacitor connects between this pin and pin 14.
14	TIM C1	I	Timing Capacitor Input. The timing capacitor connects between this pin and pin 13.

TQM

ELECTRICAL CHARACTERISTICS

Test Conditions: V_{CC} = 12V, T_A = +25°C, R_O = 30KΩ, C_O = 0.033μF, unless otherwise specified.

Parameter	Min.	Typ.	Max.	Unit	Conditions
General					
Supply Voltage	4.5		20	V	
Supply Current		4	7	mA	$R_0 \geq$ 10KΩ. See *Figure 4*.
Oscillator Section					
Frequency Accuracy		±1	±3	%	Deviation from f_O = 1/R_0 C_C
Frequency Stability					
Temperature		±20	±50	ppm/°C	See *Figure 8*.
Power Supply		0.05	0.5	%/V	V_{CC} = 12 ±1V. See *Figure 7*.
		0.2		%/V	V_{CC} = ± 5V. See *Figure 7*.
Upper Frequency Limit	100	300		kHz	R_0 = 8.2KΩ, C_0 = 400pF
Lowest Practical Operating Frequency		0.01		Hz	R_0 = 2MΩ, C_0 = 50μF
Timing Resistor, R_0 - See *Figure 5*					
Operating Range	5		2000	KΩ	
Recommended Range	5			KΩ	See *Figure 7* and *Figure 8*.
Loop Phase Dectector Section					
Peak Output Current	±150	±200	±300	μA	Measured at Pin 11
Output Offset Current		1		μA	
Output Impedance		1		MΩ	
Maximum Swing	±4	± 5		V	Referenced to Pin 10
Quadrature Phase Detector				Measured at Pin 3	
Peak Output Current	100	300		μA	
Output Impedance		1		MΩ	
Maximum Swing		11		V_{PP}	
Input Preampt Section				Measured at Pin 2	
Input Impedance		20		KΩ	
Input Signal					
Voltage Required to Cause Limiting		2	10	mV rms	

Notes
Parameters are guaranteed over the recommended operating conditions, but are not 100% tested in production.
Bold face parameters *are covered by production test and guaranteed over operating temperature range.*

XR-2211

EXAR

Linear FM Detection

XR-2211 can be used as a linear FM detector for a wide range of analog communications and telemetry applications. The recommended circuit connection for this application is shown in *Figure 13*. The demodulated output is taken from the loop phase detector output (pin 11), through a post-detection filter made up of R_F and C_F, and an external buffer amplifier. This buffer amplifier is necessary because of the high impedance output at pin 11. Normally, a non-inverting unity gain op amp can be used as a buffer amplifier, as shown in *Figure 13*.

The FM detector gain, i.e., the output voltage change per unit of FM deviation can be given as:

$$V_{OUT} = \frac{R_1 \cdot V_{REF}}{100 \cdot R_0}$$

where VR is the internal reference voltage ($V_{REF} = V_{CC}/2$ - 650mV). For the choice of external components R_1, R_0, C_D, C_1 and C_F see the section on design equations.
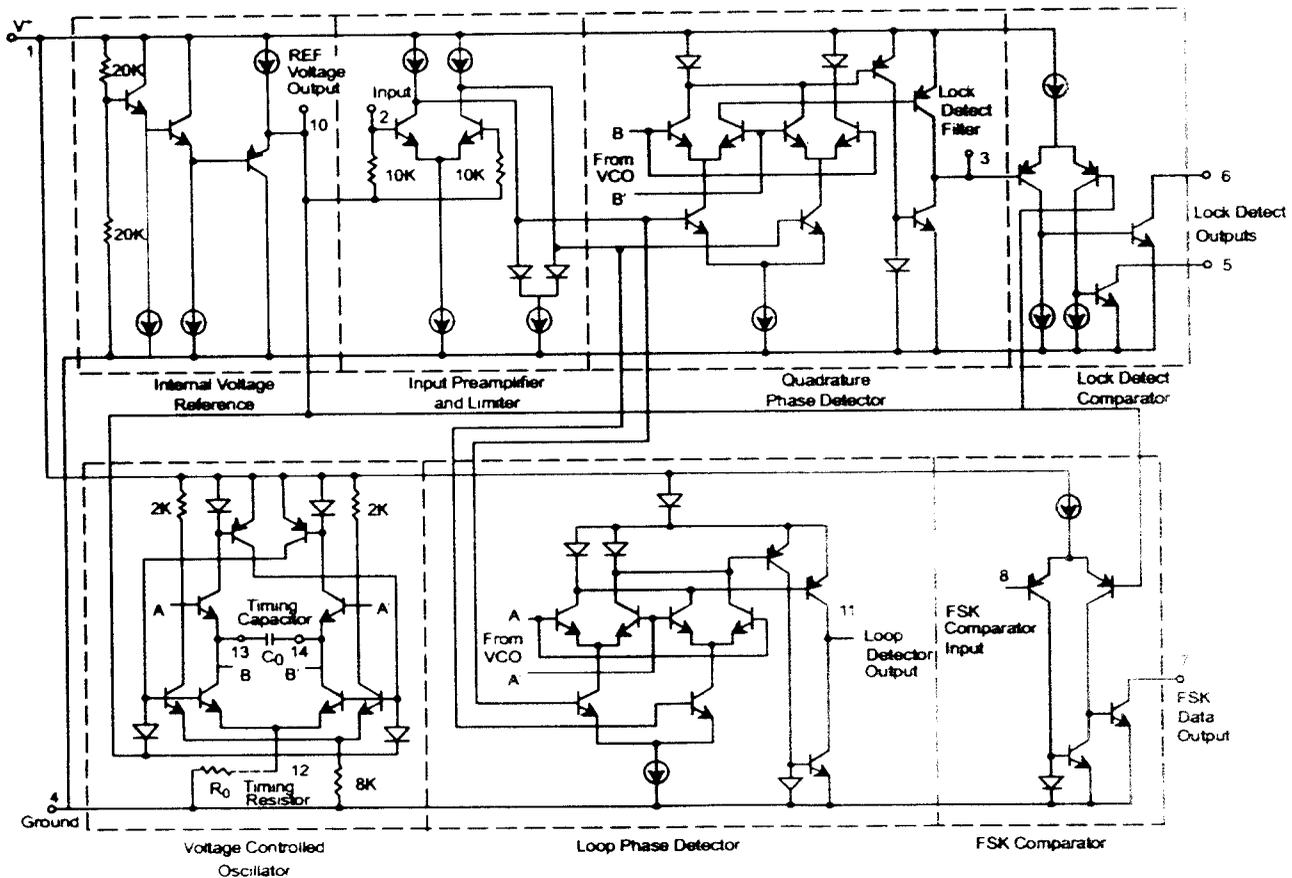

Figure 14. Equivalent Schematic Diagram

BIBLIOGRAPHY

- Simon Haykin, Communication System.

- John Grover, RS-232 Complete Solution.

- Mark Nelson, Serial communication.

- Douglas V.Hall, Microprocessors and Interfacing programming and hardware.

- Joe Campbell, C-Programmer's Guide to serial Communication.

- George Kennedy, Electronic communication system.

- Robert J.Schoenbeck, Communication Engineering.

- William A. Shay, Understanding Data Communications & Networks.