AMSTRAD

PCW 8256/8512

RS 232C SERIAL & PARALLEL CENTRONICS INTERFACE

USER GUIDE

Pace Micro Technology
Juniper View, Allerton Road,
Bradford BD15 7AG
Tel: 0274 488211

Int. Tel: +44/274/488211 Telex: 51338 Telecom Gold: 79:PCE001 Prestel Mailbox: 274729306

(A Division of Pace Software Supplies Ltd.)

Pace Micro Technology

RS232 Serial and Centronics parallel Interface

for the

Amstrad PCW Series Microcomputers

Pace Micro Technology
Allerton Road
Bradford
West Yorkshire
BD15 7AG

Tel : (0274) 488211
Tlx : 51338

Contents

1. Introduction

The Pace RS232/Centronics interface, for the Amstrad PCW series, micro-computers, provides two additional ports offering the means to connect alternative parallel or serial printers and a variety of other devices such as modems.

The interface fits neatly onto the expansion port on the rear of the computer and once fitted need never be adjusted. The connector labelled A in figure 1 is a standard parallel Centronics output of the type used to connect to most parallel printers including daisywheels.

The second connector, labelled B, provides a single RS232 input/output port for use with modems or other serially driven devices. The connector is a standard male 25-way D-type, the pin designations for which are given in Appendix A.

The CP/M operating system incorporates a number of utilities to assign and configure these ports but in the case of the serial port you will normally want to operate using a commercially available communications package such as Commstar PCW.

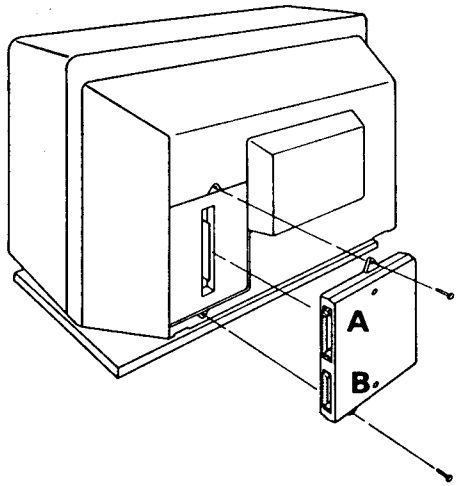


Figure 1 : Position of interface for fitting.

2. Installation

Figure 1 below illustrates the position of the expansion interface on the rear of the PCW monitor/ housing. The PCW232 connects directly onto this port and is held in place by two screws positioned as indicated on the top and bottom edges of the interface housing. Once fitted, the connectors for the two ports will exit to the right of the computer when viewed from the front. To avoid damaging either the computer or interface during installation follow the instructions given below :

1) Switch OFF the computer and disconnect it from the power supply.

2) If the expansion port is already occupied the item will have to be removed before the PCW232 can be fitted.

3) Push the PCW232 into place on the edge connector and secure it with the two fixing screws provided.

Power may now be re-applied and the machine should 'boot' as usual. You will however notice that the message at the top of the screen now includes the wording :

SIO/Centronics add-on.

indicating that the operating system has recognised the presence of the interface.

3. CP/M Device Assignments

The two ports provided on the interface are assigned as the physical devices SIO (RS232) and CEN (Centronics) under the CP/M Plus BIOS. The relationship between the physical devices and the logical devices used by programs can be altered at any time using the CP/M DEVICE utility supplied on side 3 of the system discs. Thus, input and output for programs such as PIP can be re-directed to any desired port.

In particular, the three physical devices SIO, CRT and LPT may be mapped as necessary onto the logical devices CON, AUX and LST. Full details of the DEVICE command are contained in the User manual and CP/M guides for the PCW computer. This section details some of the commands relevent to the Pace RS232/Centronics interface.

Normally, the logical device CON:, often referred to as the console under CP/M, is assigned to the physical device CRT: (keyboard and screen).

The logical printer device, LST: is assigned to the internal parallel port LPT and the auxiliary device AUX:, is assigned to the physical device SIO. A list of the current assignments can be obtained using the command :

DEVICE

This will display a list such as that shown below when the serial interface is fitted.

```
Physical Devices:
I=Input,O=Output,S=Serial,X=Xon-Xoff
CRT   NONE IO    LPT   NONE O  SIO 9600 IOS CEN  NONE O

Current assignments:
CONIN:  CRT
CONOUT: CRT
AUXIN:  SIO
AUXOUT: SIO
LST:    LPT
```

The device assigments may be changed using commands of the form :

DEVICE <logical device>=<physical device>

e.g. DEVICE CON:=SIO

This command would allow the computer to be controlled remotely from a terminal attached to the RS232 port.

4. Using the Centronics port

A Centronics parallel printer (or plotter) should be connected to the 36-way port using a length of 26-way ribbon cable with an identical 36-way connector at each end (not all of the pins are used). Most reputable dealers will be able to provide such a lead which should be no longer than about 1.5 metres. In case of doubt a specification for the connector is provided in Appendix B.

The Centronics port is fully supported by the CP/M operating system and output may be re-directed to it by using the DEVICE command :

DEVICE LST:=CEN

This will cause all printer output to be directed to the Centronics port instead of the internal PCW printer. You can test this by typing [ALT]P at the keyboard (hold the ALT key down and tap the 'P' key before releasing ALT again). Everything you now type and anything CP/M displays, will be printed. To disable printing type [ALT]P again.

Printer output will be restored to the internal parallel port each time you switch OFF or reset the computer, or move from Locoscript to CP/M. Therefore, if you intend using the Centronics port for most of your printing, you should program the re-direction as part of the CP/M boot process. This is achieved by incorporating the DEVICE command DEVICE LST:=CEN into the PROFILE.SUB file. Following this, each time you switch on the computer, the Centronics parallel port will be selected automatically.

You may restore printer output to the internal parallel port manually by entering :

DEVICE LST:=LPT

5. Using the RS232 Serial Port

The RS232 serial port provided on the PCW232 interface can be used to connect the PCW series micro-computers to a wide variety of serially driven devices. Probably the most common application is for connection to remote database and information services via a modem. In addition some printers and most modern electronic typewriters are fitted with serialinterfaces and with an appropriate cable can be readily attached to the computer.

To connect a device to the serial port you will first need a cable. For those who are familiar with RS232, the PCW232 serial port is fitted with a 25-way D-Type Male plug and is wired as a standard DTE. A one-to-one cable will therefore suffice to connect to a modem or other equipment which is wired as DCE. Hardware handshaking is provided using DTR, RTS, DCD and CTS.

Serial Port configuration

When CP/M Plus is first loaded, the RS232 interface is initialised with a number of default values. In particular the baud rate is set to transmit and receive at 9600 and the data format is set to 8 data bits, no parity and 1 stop bit. When operating directly from CP/M rather than from within a communications program, there are a number of commands which may be used to configure the RS232 port. The most versatile of these is the SETSIO command and it is usual to incorporate the appropriate form of this command into the PROFILE.SUB file so that your own requirements for the values are automatically set when CP/M is first loaded.

Baud Rate Selection

The default baud rate for the RS232 port is 9600/9600. If you require a different rate it must be selected explicitly each time you reset the computer. This may be achieved using the DEVICE command as follows :

DEVICE SIO [300]

which will configure the port to send and receive at 300 baud. Alternatively you may use the SETSIO utility provided on side 2 of the systems discs. In it's simplest form, for example :

5. Using the RS232 Serial Port

The RS232 serial port provided on the PCW232 interface can be used to connect the PCW series micro-computers to a wide variety of serially driven devices. Probably the most common application is for connection to remote database and information services via a modem. In addition some printers and most modern electronic typewriters are fitted with serialinterfaces and with an appropriate cable can be readily attached to the computer.

To connect a device to the serial port you will first need a cable. For those who are familiar with RS232, the PCW232 serial port is fitted with a 25-way D-Type Male plug and is wired as a standard DTE. A one-to-one cable will therefore suffice to connect to a modem or other equipment which is wired as DCE. Hardware handshaking is provided using DTR, RTS, DCD and CTS.

Serial Port configuration

When CP/M Plus is first loaded, the RS232 interface is initialised with a number of default values. In particular the baud rate is set to transmit and receive at 9600 and the data format is set to 8 data bits, no parity and 1 stop bit. When operating directly from CP/M rather than from within a communications program, there are a number of commands which may be used to configure the RS232 port. The most versatile of these is the SETSIO command and it is usual to incorporate the appropriate form of this command into the PROFILE.SUB file so that your own requirements for the values are automatically set when CP/M is first loaded.

Baud Rate Selection

The default baud rate for the RS232 port is 9600/9600. If you require a different rate it must be selected explicitly each time you reset the computer. This may be achieved using the DEVICE command as follows :

DEVICE SIO [300]

which will configure the port to send and receive at 300 baud. Alternatively you may use the SETSIO utility provided on side 2 of the systems discs. In it's simplest form, for example :

SETSIO 300

it is equivalent to the DEVICE command shown above. However, using SETSIO you may also set the send and receive rates independently, (such as for use with split baud rate systems).

For example :

SETSIO RX 1200, TX 75

Note that when using the DEVICE command to interrogate I/O port settings, changes made to the baud rate using the SETSIO command will not be apparent.

Data Format

Apart from the speed of transmission, it may also be necessary to configure the framing of data. Asynchronous transmission of data via an RS232 port requires the addition of extra information in order to ensure correct timing and to provide limited error checking. This information consists of Parity bits and Stop bits. The type of parity and the number of stop bits can be changed as can the number of actual data bits used. The following values are allowed :

Data bits : The RS232 is capable of sending either 7 or 8 data bits.

Parity : This is a simple form of error check which can be set to ODD, EVEN or NONE.

Stop bits : The end of each character in the data stream is signified by either 1, 1.5 or 2 'stop' bits.

For further explanations of these terms refer to the glossary (Appendix C) at the rear of this manual.

The default values for these settings are 8 data bits, No parity and 1 Stop bit but these can easily be altered using the SETSIO command.

For instance :

SETSIO BITS 7 PARITY EVEN STOP 2

would set the data format to 7 data bits, Even parity and 2 stop bits (the format used by Telecom Gold). Most bulletin boards use 8 data bits, no parity and 1 stop bit.

In fact the SETSIO command allows any combination of the RS232 parameters to be changed using the following clauses :

SETSIO 300

it is equivalent to the DEVICE command shown above. However, using SETSIO you may also set the send and receive rates independently, (such as for use with split baud rate systems).

For example :

SETSIO RX 1200, TX 75

Note that when using the DEVICE command to interrogate I/O port settings, changes made to the baud rate using the SETSIO command will not be apparent.

Data Format

Apart from the speed of transmission, it may also be necessary to configure the framing of data. Asynchronous transmission of data via an RS232 port requires the addition of extra information in order to ensure correct timing and to provide limited error checking. This information consists of Parity bits and Stop bits. The type of parity and the number of stop bits can be changed as can the number of actual data bits used. The following values are allowed :

Data bits : The RS232 is capable of sending either 7 or 8 data bits.

Parity : This is a simple form of error check which can be set to ODD, EVEN or NONE.

Stop bits : The end of each character in the data stream is signified by either 1, 1.5 or 2 'stop' bits.

For further explanations of these terms refer to the glossary (Appendix C) at the rear of this manual.

The default values for these settings are 8 data bits, No parity and 1 Stop bit but these can easily be altered using the SETSIO command.

For instance :

SETSIO BITS 7 PARITY EVEN STOP 2

would set the data format to 7 data bits, Even parity and 2 stop bits (the format used by Telecom Gold). Most bulletin boards use 8 data bits, no parity and 1 stop bit.

In fact the SETSIO command allows any combination of the RS232 parameters to be changed using the following clauses :

TX <b>

Sets the transmit baud rate. The parameter <baud rate> must be one of the following :

50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600, 19200

Note that if TX clause is omitted then both the transmit and receive rates will be set to the specified speed so that SETSIO 300 will set the RS232 to operate at 300 in both directions. If you alter the baud rate without changing the number of stop bits, 1 stop bit will automatically be selected for speeds higher than 110 baud. Speeds lower than this will set the number of stop bits to 2.

RX <b>

This is used in the same way as the TX clause and once again if RX is omitted the transmit rate will also be set to the specified value.

BITS <n>

The parameter <n> is the number of data bits to be used and can be set to 5, 6, 7 or 8. The default is 8.

PARITY <p>

The parameter <p> is the parity to be used, EVEN, ODD or NONE. The default is NONE.

STOP <s>

The parameter <s> is the number of stop bits, 1, 1.5 or 2. The default is 1 unless the baud rate is 110 or less in which case it is 2.

XON <x>

This option is used to invoke or cancel the use of XON/XOFF (software) flow control. The parameter x must be ON or OFF, the default is OFF.

HANDSHAKE <h>

Hardware flow control using RTS and CTS defaults to ON but may be cancelled using this command with the parameter <h> as OFF.

Note that it is only necessary to use the initial letter of each clause. For example, to configure the RS232 for 1200/1200 full duplex operation with 7 data bits, even parity, 2 stop bits and XON/XOFF flow control active, the following would suffice:

SETSIO 1200 B 7 P EVEN S 2 X ON

The configuration will be confirmed with :

1200 Bits 7 Stop 2 Parity even Xon on Handshake on

Any errors in the command line i.e. missing or incorrect parameters will generate an appropriate error message and leave the SIO unchanged.

APPENDIX A RS232 Fundamentals

In most cases your dealer will be able to supply a lead suitable for connecting the RS232 interface to your modem. The notes below are intended to clarify some aspects of RS232 for those intending to use the interface with unusual or non-standard equipment. It is not intended to provide a full technical description of RS232 and certain information has been deliberately omitted in the interests of brevity.

The RS232 standard was originally laid down by the American Electronics Industries Association (AEIA) as a means of interconnecting serially driven devices. Although having been in common use for many years there has always been much confusion about it's actual application and as a result it can be infuriatingly difficult to get two RS232 devices talking to each other successfully.

Minimum Requirements

Two-way serial communication between two devices, usually a a terminal and a modem, requires a minimum of 3 connections, one for received data (into the terminal) one for transmitted data (from the terminal) and a common line (Ground), to complete the circuit.

Note that for the purposes of this discussion the terminal is assumed to be an Amstrad PCW computer running terminal software such as Commstar PCW.

In the RS232 standard there are 23 defined signals and a 25-way D-type socket and plug, are used to provide the physical connection. Consequently, it is rarely possible to successfully connect two RS232 devices using three wires only.

According to the RS232 standard, Receive Data is pin 3 of the 25-way connector. You may also see this referred to as Data In, RD or RxD.

RS232 Pin 2 is assigned to Transmit Data and may be referred to as Data Out, TD or TxD.

Signal Ground (GND) is pin 7. Notice that these pin descriptions are given with respect to the terminal, not the modem i.e. TD is transmitted data from the terminal.

Assuming that a three wire connection would work, characters typed at the terminal would be sent via pin 2 to the modem which then transmits them to the remote system via the telephone line.

Characters received from the remote system would be passed from the modem, via pin 3, to the terminal. This is shown in Figure A-1 below.

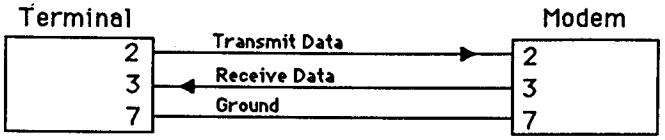


Figure A-1 Three wire serial connection.

A major disadvantage of relying on this minimal configuration is that neither party has any means of preventing the other from transmitting data, nor can they determine whether or not transmitted data has been correctly received i.e. there is no means of providing hardware handshaking between the devices.

Control Signals

In order to overcome problems such as data overrun (where the receiver cannot handle a continuous stream of incoming data), the RS232 standard introduces a number of additional signals which are used for control purposes only. These include :

DTR - Data Terminal Ready
DSR - Data Set Ready
RTS - Request to Send
CTS - Clear to Send
DCD - Data Carrier Detect

Generally, at least CTS and RTS must be used in addition to TD, RD and GND in order to establish a connection. In understanding the operation of these signals and the RS232 interface in general, it is important to realise that the standard was originally designed to allow two different types of device to be connected i.e. 'modems' and 'terminals'.

Devices classified as modems are generally used to extend a connection (usually via the telephone network), to a remote system.

Terminal equipment however varies enormously, ranging from dumb printing teletypes, through sophisticated intelligent terminals to micro-computers running 'terminal' software.

When referring to RS232, the terminology for 'modems' is DCE (Data Communications Equipment), or Data Set. Terminals (or micros running terminal software), are called DTE (Data Terminal Equipment). The reason for distinguishing between these two types of device is obvious when you consider their function. Terminals actually send and receive data, whereas modems simply pass data on in one direction or the other.

Thus, when connecting a Terminal to a Modem (DTE to DCE), you simply connect the pins straight through i.e. Pin 2 to Pin 2, Pin 3 to Pin 3 etc. In other words, Pin 2 of the RS232 connector in the modem is still called Transmit Data even though it serves to receive data from the terminal for transmission down the telephone line. Figure A-2 below shows a straight-through connection with handshaking lines.

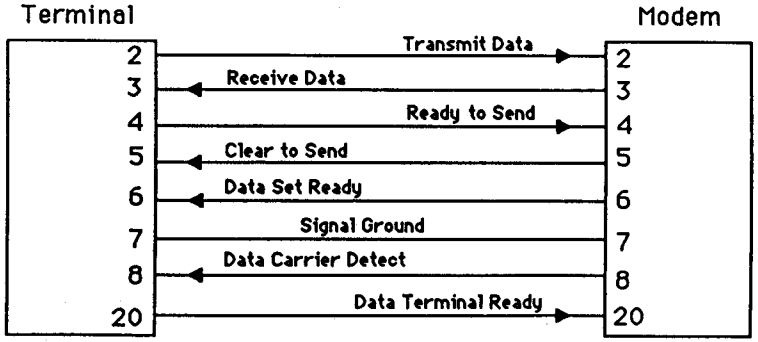


Figure A-2 Serial link with 'handshaking'

On the other hand, when connecting two terminal (DTE to DTE) devices together directly i.e. RS232 to RS232 without a modem, you would have to crossover the data connections i.e. Pin 2 to Pin 3 and Pin 3 to Pin 2. This is so that the transmitted data from one terminal becomes the received data for the other and vice versa. With respect to the control lines, you would also have to crossover the appropriate pairs i.e. RTS on one terminal to CTS on the other.

A cable which carries out this function is called a 'Null Modem' because it replaces the two back-to-back modems which would normally complete the connection. Figure A.3 shows the connections required for NULL modem connection without handshaking.

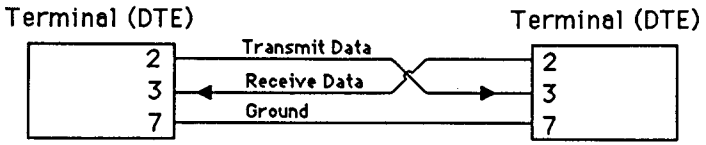


Figure A-3 Three wire Null-modem

Now let's consider the function of the control signals :

DTR : As it's name suggests, Data Terminal Ready is a signal FROM THE TERMINAL which is used to tell the modem that it is switched ON and ready to communicate.

DSR : This is the complement of DTR, it is used by the modem to inform the terminal that it also is switched ON and ready to communicate. Note : as with the Amstrad RS232, there is no provision for this signal on Pace RS232 interface. If your modem provides DSR just ignore it.

In terms of voltage, both DTR and DSR are taken high when they are asserted. Once both signals are asserted, the remaining handshaking lines take over :

RTS : This is the signal FROM THE TERMINAL which is used to tell the modem that there is data to be transmitted.

CTS : In response to RTS, the modem will use CTS to tell the terminal that transmission may proceed.

As with DTR and DSR, RTS and CTS are both taken to a high voltage when asserted.

DCD : Another signal that may be used is Data Carrier Detect which is RS232 Pin 8. This signal is asserted when the modem detects and locks on to a remote carrier and may be required so that software running in the terminal can detect carrier for auto-answer purposes.

Remember that all of these control lines operate between the terminal and the modem only, they do not control the link between the two modems. Furthermore, not all signals are always used, some systems may rely entirely on CTS and RTS only, in which case you may have fool one or other of the two connected devices by 'tying' some signals high. The following examples show some instances where 'jumping' of signals might be required.

Connecting to the Amstrad V21/23 modem

Some modems such as the Amstrad V21/23 modem, provide RTS and CTS but dispense with DTR, DCD and DSR i.e. there is no DTR input pin on the modem to which DTR from the terminal can be connected nor is there a suitable output from the modem to connect to DCD at the terminal. DTR can simply be left unconnected because it is an output from the computer which the modem does not require. Similarly, the lack of DSR from the modem does not matter because the RS232 interface does not require it. However, if the terminal software is relying on DCD to confirm that a connection has been established it may not operate correctly if DCD is not connected. In this instance the terminal (or rather the communications software running on the computer), must be fooled into thinking that the modem is providing DCD, by linking the DCD and CTS pins at the computer end of the lead. Now when the modem is ready to accept data i.e. CTS is asserted, DCD is also asserted as if the modem had actually detected carrier. The actual lead configuration for connecting the Amstrad V21/23 modem is illustrated below. A lead wired in this way will operate correctly with the Commstar PCW software.

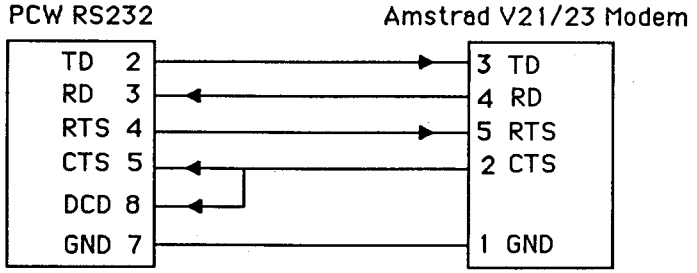
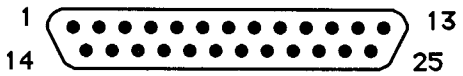


Figure A-4 Pace RS232 to Amstrad V21/23 modem.

The RS232 plug itself is wired according to the following :

RS232 Plug pin designations

Pin 2	Transmit data
Pin 3	Receive data
Pin 4	RTS (output)
Pin 5	CTS (input)
Pin 7	Ground
Pin 8	DCD (input)
Pin 20	DTR (output)
Pin 22	Ring Ind. (input)



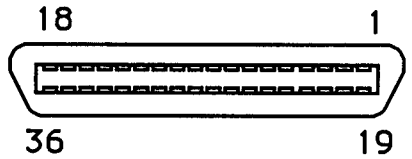
RS232 pin numbering

Appendix B Centronics port connections

Centronics socket pin designations

Pin 1	STROBE	Pin 6	D4
Pin 2	D0	Pin 7	D5
Pin 3	D1	Pin 8	D6
Pin 4	D2	Pin 9	D7
Pin 5	D3	Pin 11	Busy

Pins 10, 12-15, 17, 18, 29-32 not connected
Pins 16, 19-28, 33 Gnd



Appendix B Centronics port connections

Centronics socket pin designations

Pin 1	STROBE	Pin 6	D4
Pin 2	D0	Pin 7	D5
Pin 3	D1	Pin 8	D6
Pin 4	D2	Pin 9	D7
Pin 5	D3	Pin 11	Busy

Pins 10, 12-15, 17, 18, 29-32 not connected
Pins 16, 19-28, 33 Gnd

Appendix C Glossary

Baud rate

The baud rate is an indication of the speed at which data is passed between two communicating devices, a higher baud equating to a faster transfer rate. In most cases, the bit rate is the same as the baud rate so that when the word length is 10 bits, each word representing one character, then 300 characters/second can be transferred via a 300 baud connection.

Bulletin board

A bulletin board is a computer 'service', usually running on a micro-computer, that accepts calls via an auto-answer modem and allows callers to leave messages, transfer files etc. The software that manages the service can be left unattended to control access to the various facilities that may be available.

Communications software

This term is generally used to describe software that allows a micro or other computer to emulate a terminal device. Packages that are available range from simple, 'dumb' terminal emulators to more sophisticated packages which provide facilities such as automatic and selective storage of incoming data, automatic dialling of sequences of telephone numbers at particular times, error protection protocols, multi-level security etc.

Data Communications Equipment (DCE)

The international body which governs telecommunications, the CCITT (International Consultative Committee for Telegraph and Telephones) uses the term DCE to describe communications equipment such as modems, multiplexors etc.

Data Terminal Equipment (DTE)

This is the term used by the CCITT to describe terminal equipment in a communications environment. A DTE might actually be a terminal or it could be a micro-computer running some communications software.

Downloading

The term used to describe the process of transferring data or programs from a remote information/data service to a user.

Appendix C Glossary

Baud rate

The baud rate is an indication of the speed at which data is passed between two communicating devices, a higher baud equating to a faster transfer rate. In most cases, the bit rate is the same as the baud rate so that when the word length is 10 bits, each word representing one character, then 300 characters/second can be transferred via a 300 baud connection.

Bulletin board

A bulletin board is a computer 'service', usually running on a micro-computer, that accepts calls via an auto-answer modem and allows callers to leave messages, transfer files etc. The software that manages the service can be left unattended to control access to the various facilities that may be available.

Communications software

This term is generally used to describe software that allows a micro or other computer to emulate a terminal device. Packages that are available range from simple, 'dumb' terminal emulators to more sophisticated packages which provide facilities such as automatic and selective storage of incoming data, automatic dialling of sequences of telephone numbers at particular times, error protection protocols, multi-level security etc.

Data Communications Equipment (DCE)

The international body which governs telecommunications, the CCITT (International Consultative Committee for Telegraph and Telephones) uses the term DCE to describe communications equipment such as modems, multiplexors etc.

Data Terminal Equipment (DTE)

This is the term used by the CCITT to describe terminal equipment in a communications environment. A DTE might actually be a terminal or it could be a micro-computer running some communications software.

Downloading

The term used to describe the process of transferring data or programs from a remote information/data service to a user.

Host system

This is a general term which is used to describe a computer which accepts data calls from other remote computers or terminals via a suitable transmission medium, usually the public telephone network. In their simplest form they may offer electronic messaging and 'noticeboard' facilities e.g. bulletin boards. At the other end of the spectrum there are some very powerful information services such as the Source and Knowledge index which offer complex search and retrieval facilities for a whole range of specialist information areas. Other facilities which may be found include Telex, Electronic mail, on-line statistical analysis etc.

Modem

A device which both MOdulates and DEModulates data. Modulation is the process of converting digital computer data into a form suitable for transmission over the analogue speech telephone system. Demodulation is the reverse of this, the conversion of incoming analogue data into digital form suitable for use by a computer.

Parity bit

A parity bit is an extra bit that is sometimes added to a character as a simple form of error check. It is best illustrated by example.

The binary form of the ASCII character 'A' is 01000001. You can see that there are two '1' bits in this character. With EVEN parity added this character would become 010000010 i.e. after the addition of EVEN parity the number of 1 bits is still EVEN. If ODD parity was used instead, the character would become 010000011.

The idea is that the parity bit maintains the 'eveness' of 'oddness' of the number of 1 bits in a character. Thus when a with character EVEN parity is transmitted over a communication line it should still have an EVEN number of 1 bits at the receiver. If an error occurs during transmission which changes a 1 bit to a 0 (or vice versa) the total number of 1 bits will be no longer be EVEN and the receiver can detect the error.

You can see from this that it is very important for the sender and receiver to be configured for the same type of parity.

Parity checking in this way is not very effective in terms of error protection, for example, if two bits in a character are corrupted then the parity will be correct at the receiver.

Protocol

A protocol is a set of rules which both the sending and receiving equipment obey during communication.

RS232

RS232 is the name given to a standard which defines the electrical and physical characteristics of an asynchronous serial interface. The standard originated in America but is now widely accepted throughout the world.

Serial Interface

A serial interface is a device which converts parallel data into a serial form which may be transmitted over a two or three wire circuit. It may also carry out the reverse operation of converting serial data into parallel form. Most modems, some printers and a few other devices require serial input and a serial interface allows the connection of such devices to terminals and computers.

Start bits

This is an extra bit added to a character before it is transmitted so that the receiving equipment recognises the start of the character. The start bit is automatically generated by the transmitting RS232 interface and then removed by the receiving interface.

Stop bits

During data transfer, 1 or 2 extra bits may be added to the end of each character by the RS232 interface so that the receiving equipment can detect the end of the character. These are the 'stop' bits and depending on the quiescent state of the line may be 0's or 1's. The generation of the stop bits at the sender and then their removal at the receiving equipment, is totally transparent to the user and is carried out by the RS232 nterfaces.

Terminal

The term originally used to describe a teletypewriter and keyboard which were used to communicate with the Central Processing Unit (CPU) of a computer. The definition has been expanded in recent years to include glass teletypes (VDU's) and micro-computers running terminal software.

Word length

The total number of bits which are recognised as comprising a unit of data transfer, a word. This need not be the same as the word length within the computer, extra formatting bits are added during data transfer (e.g. stop bits, start bits, parity).

XON/XOFF

A simple form of data transfer protocol which is used to control the flow of data. If the receiving system wishes to stop the transmission, for example, to save the data it has already received into a disc file, then it sends an 'XOFF' signal to the host. When the sender receives an XOFF signal it ceases transmission and will not resume until it receives an 'XON' signal from the receiver. XOFF is usually ASCII 19 (CTRL-S) and XON is ASCII 17 (CTRL-Q).

XMODEM

A relatively sophisticated form of file transfer protocol, developed by Ward Christiensen, that allows the transfer of any type of file, either ASCII or binary, without conversion. The protocol involves breaking the data into blocks and adding calculated checksums which can be verified by the receiver. It also caters for flow control in that the receiver can request re-transmission of blocks that have been received with errors.