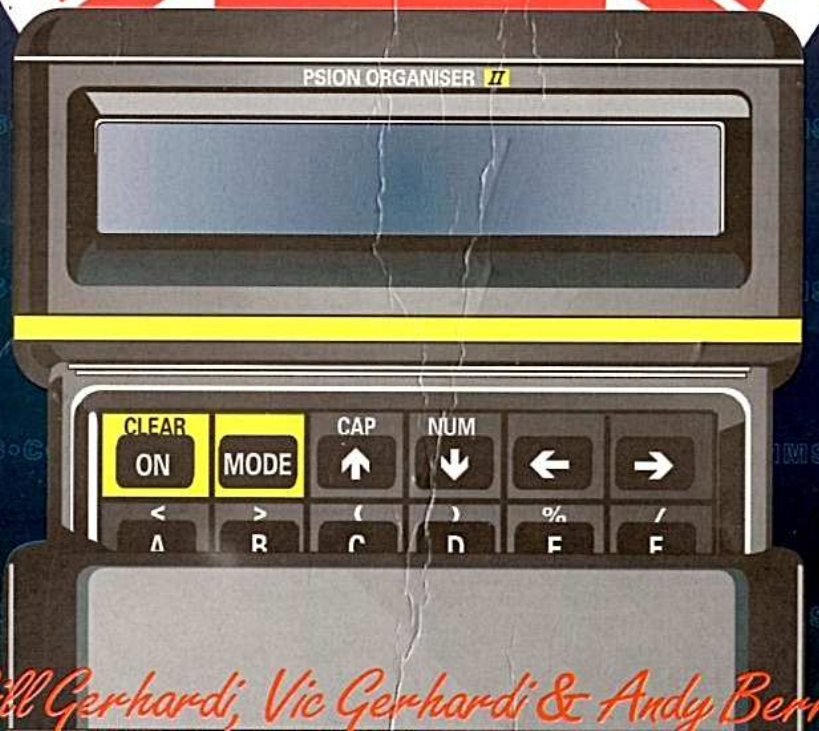
PSION ORGANISER
COMMS HANDBOOK



Gill Gerhardi, Vic Gerhardi & Andy Berry

The Psion Organiser

Communications Handbook

Gill Gerhardi, Vic Gerhardi and Andy Berry

Published by
Kuma Computers Ltd.

First Published 1990
Kuma Computers Ltd
12 Horseshoe Park
Pangbourne
Berkshire. RG8 7JW
Tel 0734 844335. Fax 0734 844339

Printed in Great Britain

ISBN 0-7457-0154-X

Acknowledgements

We would like to say a big "Thank you" to Anne from next door who again managed to transform our abysmal typing into readable text!

By the same authors:

Z88 Magic

The Psion Organiser Deciphered

In the same series from Kuma:

Using & Programming the Psion Organiser by Mike Shaw

File Handling & Other Programs for the Psion Organiser by Mike Shaw

Machine Code Programming on the Psion Organiser by Bill Aitken

Other Kuma Publications:

A selection of recent titles from the rapidly expanding range of computing books published by Kuma. Full up-to-date details are available on request.

Desktop Publishing Sourcebook - Fonts & Clip Art for the Mac by Jami Lynne Borman

Desktop Publishing Sourcebook - Fonts & Clip Art for the PC by Jami Lynne Borman

ZBasic Quick Reference Guide by John Sumner

Pagemaker 4.0 for the IBM Enviroment by William Sanders

Intuition: A Practical Guide for Amiga Programmers by Mike Nelson

The Atari ST Explored by John Braga

The Authors

Vic Gerhardi and Andy Berry have been friends since school days and have been business associates for over ten years. Andy is the software specialist. He graduated from Middlesex Polytechnic and Manchester University with BSc.s and MSc.s coming out of his ears! He then worked in systems design for a number of companies before becoming a Director of Rakewell. Vic has an electrical engineering background. He has designed, tested and maintained hardware systems for as long as he can remember! Running Rakewell now takes most of his time.

Rakewell is a thriving computer consultancy, It has a large customer base which reflects their willingness to deal with individuals and the corporate sector alike. The company can design and supply hardware or software to meet most of the hugely diverse needs of computer users. They pride themselves on their customer support services. After supplying a number of Cambridge Computer Z88s they began to realise that many of their customers were not getting as much as they could out of the Z88. After much thought they designed and wrote two training courses which were very favourably received and are still running (in October 1990). In the summer of 1989 they shut themselves away to convert the course notes into a book. They soon found that it was a far bigger task than they had time for and Gill was drafted in to help. The result was "Z88 Magic" which was published in November 1989.

Hot from that highly successful launch into the world of publishing the team were commissioned to write "Organisers Deciphered" and the "Organiser Communication

Handbook". Having learnt the hard way how much time writing a book takes it was decided that Gill should write 'Deciphered' (which was published in July 1990) leaving the Comms book to the experts. But as with all good plans... Gill ended up writing the 'Comms Handbook' as well! She had to lean very heavily on her technical advisors for this one however, which makes it even more a team effort than 'Deciphered' had been, although Gill got most of the work, again!

Gill was married to Vic while she was still at the University of Sussex reading Social Administration. Since graduating she has spent the last twelve years bringing up their two sons, writing, painting and working voluntarily in the Arts.

Contents

Preface

This book has turned out differently to the way we envisaged when we took it on. We planned a large section with the 'setup' parameters and 'pin-outs' for lots of different individual machines, but that became more impractical the more research we did. Why? Well partly because the information just isn't there! No one has ever collated the solutions that have been found and we didn't have the time or resources to do the research and testing ourselves.

The other reason for not giving precise information is the fast rate of progress in the computer world. If we were to give you specific tables to connect your Organiser to every computer, printer and modem on the market now, ten per of them would be obsolete within six months, or less, of the book being published. It would also mean that there would be as many new machines on the market not covered in the book at all. That seemed a very unsatisfactory way to tackle the problem.

So instead we have written this book. It gives you all the information you need to create a link between your Organiser and any other machine. This book actually will allow you to become your own communications expert.

Another reason why the book changed from it's original form was because it had a change of author. You will see from 'The Authors' that it is Vic and Andy who are the experts and this was supposed to be their book not mine. But I had time, they didn't! I knew nothing about communication, RS-232 or anything else before I started this. I had to

go back to first principles and understand those before I had a hope of understanding the finer points!

The result is a book that you can use in several different ways:-

> For the beginner, it can be read from cover to cover as the complete guide to Organiser communication.
>
> If you have a specific communication problem, you can find the solution by following the step by step instruction sections at the beginning of most chapters.
>
> If you already have a basic knowledge of RS-232 and data communications. you will be able to skip the first two or three chapters and dive in at the deep end to follow the testing strategy using BOBs, or use our chapter on file conversion which will help you solve the more complicated problems.

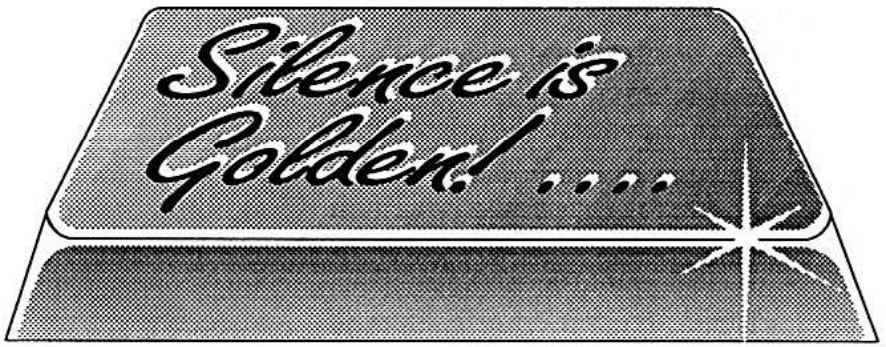
To say that this has not been an easy book for me to write would be an understatement. But Kuma have been very patient and although 'the experts' did run out of patience with my endless questions (especially when I had already asked the same one twice before) we didn't quite fall out altogether! We are all very pleased with the result. But please Kuma could you make the next book slightly easier!

Hope you enjoy reading this, oh, and learn a lot of course!

Gill

Introduction

Why Communicate?



Communication is the way ideas, information, and feelings are exchanged. The saying "no man is an island" can only be true if each individual communicates his thoughts and knowledge to others.

Communication is one of the most difficult things that we humans ever have to learn to do. What we understand by another's verbal or non verbal communication is highly coloured by our cultural background and past experiences,

the sum total of our knowledge, what we feel about the person concerned and the way we feel at that particular moment. Not to mention the fact that we need to speak the same language to start with. It is not surprising that we find ourselves walking through minefields occasionally.

Getting computers to communicate with each other and their accessories should be easier, 'should' being the operative word. After all there are many fewer factors to go wrong. The computer industry has made many attempts to standardise to make such communication easier, but so far these have not been taken up by all manufacturers which means that inter-computer communication is still no easy task.

There are factors that make communication both between computers and between computers and their accessories, difficult. The speed at which the evolutionary process is moving is one. It means that somebody somewhere is always thinking up a better, easier, faster or smarter way of solving the problems that are endemic to computer design, manufacture and usage. This has led to a myriad of manufacturers with a multitude of models. Many manufacturers produce a whole range of computer equipment and as an inducement to the customer to only buy his range they produce their own means of transferring information between their own bits and pieces. That's all right if you want to buy that complete range, but many of us already have printers etc. and when we buy either a new computer or different accessories we want them to be able to work with what we already have.

Why you Need your Organiser to Communicate

If you have a printer, another computer or a modem (to connect your Organiser to a computer that is not in the same room), you will not be fulfilling your equipment's full potential if you don't link them together in some way. Opening the lines of communication will enable you to -

a print files

and

b exchange files between computers.

There are many practical reasons why you may want your Organiser to do this.

Printing

You will need to connect your Organiser to a printer to get a 'hard' copy on paper of anything that your Organiser is holding in memory. Whether it is your database, diary, a letter etc. or a spreadsheet or a Pager message, they will often be easier to study in depth if you have the whole thing in front of you instead of seeing it all in fragmented bits and trying to piece it all together in your head. You may also find it cheaper to keep files that you no longer want to do any more work on - past diary entries for example - on paper rather than on much more expensive Ram/Datapaks.

Linking Computers Together

There are huge advantages if you can allow your computers to talk together. If you have a desktop computer you can work on a file at home or in the office and then carry on

working with it elsewhere on your Organiser. You can share your database so you can have a copy of all your telephone numbers wherever you are in the world. Likewise with your diary files - and notes if you have an LZ/LZ64 - you can carry them around in your pocket as well as having a copy back at base.

Being able to share information allows you to have multiple copies of your files should anything go wrong with one or both of the computers. Although computers generally have their own 'safe' way to store files outside their own memory (Ram/Datapaks on the Organiser or floppy disks for desk top models), computers do still go wrong. Rarely we grant you, but the possibilities for disaster are there. As is the possibility that you do something wrong and you lose information that way. But if you have information stored on two machines you have at least got the chance to carry on using the data while the other machine is down. You will still need to have multiple copies, but it covers all major eventualities if you have dual copies on different media. It is also a good idea to keep copies on media that have separate power supplies when continual power is needed to keep the memory 'alive' so that at least one copy is safe.

Wherever the information is stored on your Organiser, either in the internal ram or on a Ram/Datapak, you will be able to print it, or send it to another computer once you have established the correct connectors, cables and Set-Up settings on the Comms Link. To anyone not used to working in this field getting all that right is a nightmare but we will try to unravel the mysteries for you.

About the Book

We have written this so that if you are establishing a link for the first time with a computer or accessory you can work through the practical sections and be up and running with the minimum of fuss. For the tougher problems you may need the Delving Deeper sections. For the real tough ones you will also need the later chapter, which will encourage you to experiment until you find the answer yourself. To experiment successfully you will need to be very patient and understand the theory so we will try and get this over as painlessly as possible!

Communication down cables seems complicated because there are so many elements to get right before you will see anything happening. There are, some basic principles which simplify the whole process which will guide you on your way.

The Model

To help you visualise what communication is all about we have thought up a model which describes the process in another way. In other books we have written we have used the models to illustrate points throughout but we will not be doing that with this one because of the complex multi-faceted nature of communication. But we have included the model in here to whet your appetite and to illustrate how many variables are involved.

An adequate analogy was not easy to find. We needed a system where information was being copied and then transferred between two points with some degree of traffic control in operation. There also needed to be a variety of routes with a number of departure/destination points on one

side, with only one of each on the other. We know that on one end - and that is the only fixed point of reference we do have - we are working with the Organiser. After much consideration we decided that a military command post might fit the bill quite well.

Imagine that you are Commander In Chief of the latest alliance's land forces. The alliance, although not at war, is fighting a sporadic border operation against its enemies. To be effective you need up to the minute information on each squirmish as it develops, with a cast iron command line to transmit your orders. You are old fashioned enough not to trust radio. You know that the enemy can hear anything sent that way as clearly as your own men can. You prefer a dispatch rider system backed up by engineers to ensure that the roads and bridges are passable. Each message is split into many different parts with each segment being carried by a different rider. This system ensures that the information doesn't get compromised if one rider is captured. It needs a fair degree of marshalling at either end. Your Command is made up of units from many different countries which has created massive communication problems in the past. These have been increased by cultural differences which have lead to differing interpretations of the same command. You have solved some of these by insisting on re-training the Liaison officers when they join your force but there are still differences which have a habit of causing you problems at the least opportune moment.

When you wish to make contact with a unit - or when they want to contact you - there are a number of things that need to be considered. The list below are as far as possible in the order that the factors need to be considered, although in reality they would overlap a great deal or need to be dealt with simultaneously.

1 Does the unit at the other end have the capability to receive a number of riders riding in formation, all arriving at once? Or does it need riders one at a time?

2 Is the road intact?

3 Is the route safe, with no bridges knocked out or enemy snipers lurking?

4 Is the message ready to be sent?

5 Does it need translating?

6 Does the message obey the unit's cultural rules?

7 Does the unit know how to deal with the messages?

8 Is it ready to receive the current message? Is the Liaison Officer at the other end ready to receive it, sort it into the correct order and check that it is all there - that all the riders got through - before giving it to his commanding officer?

You can only send the message when the answers to all these questions are satisfactory. You were wondering why communication wasn't simple weren't you!!

How is all this relevant to you sending files from your Organiser to another piece of electronic wizardry? Well, if your command post is your Organiser and the unit you need to get the message through to is either a printer, a second computer or a modem, you need to know whether the unit you want to communicate with -

1 Works on the serial or parallel principle of communication

2 Needs a different cable to the one you already have on your Comms Link

3 Needs a different connector to the one you have got on the end of your Comms Link

You also need to check whether -

4 The file is saved and ready to be sent.

5 The file needs to be converted into an acceptable form for the recipient to cope with.

6 Both Machines are using the same protocol.

7 The receiving machine is turned on, with any relevant software loaded and waiting to talk to the Organiser.

8 There is a protocol available to check that the information is arriving intact and it is able to send confirmation back to the Organiser.

What you Need for your Organiser to Communicate

Basic Requirements

An Organiser
A Comms Link
Cable and/or connectors to connect the Comms Link to the receiving unit
The receiving unit's Manual

If Communicating with Another Computer

Relevant software and documentation for the receiving unit, although if you are using a IBM P.C. compatible or Apple Mac. the software (CL) is supplied with the Comms Link. The Organiser has its software built into the Comms Link.

You may also need

Gender Adapters
Break Out Box
Lots of time
Bags of patience
An understanding partner!

Useful

Access to as many electronic components mail order catalogues as you can get hold of!

What this book aims to give you

A basic understanding of:

The way computers communicate
The Comms LINK
RS-232

And, a shoulder to cry on or something to throw at the cat - depending on what type of person you are - when things go wrong, or rather when they don't do anything! You will learn very quickly that there are no half measures with communication. It either works or it doesn't! If you get the connections wrong nothing happens. But even after you have made contact you can still get gobbledegook being sent instead of your nice friendly file if the translation or handshaking is not correct.

Testing... Testing....

When writing other books we have tested every little statement before writing it down. With this book, because of the multiplicity of things that your Organiser could communicate with, it has been impossible for us to test everything on every type of machine to our normal standard. So if we get it wrong for your 'other' machine we are saying sorry in advance. We are confident however that there is enough in the

book to help you sort out whatever we have got wrong for yourself and still end up a happy communicator!

Well that is enough generality. It is time to get down to details. You have a lot to get to grips with before you will even know when a unit is under attack let alone be able to send orders to your men on how they should repel it or indeed retreat! So lets get started....

CHAPTER 1

Things You Need To Know Before You Start

This is where we deal with the meaning of life, communication and everything! Well nearly! This chapter will cover the concepts that you will need to know if you are starting from scratch and you want to get into the theory straight away. If theory is not your cup of tea you can skip this and move on to the practical sections but you may have to come back to this one later and use it as a glossary of concepts. Individual terms will be generally covered in the chapters as they arise. When you come across a concept that you are not sure of you can look back here for an explanation.

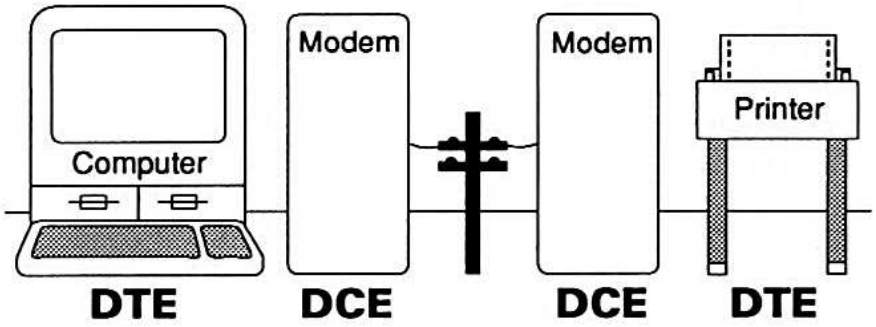
DTE - Data Terminal Equipment
DCE - Data Communication Equipment

These are the two categories of electronic equipment that were defined as the RS-232 standard. The best explanation we have found to date likens the 'terminal' in Data Terminal Equipment to a rail or bus terminal which is usually only found at the two extreme ends of a train or bus route. DCEs on the other hand, with the emphasis on communication, can fit in between DTEs and help the two terminals talk to each other. Unlike the stops on a bus route however, there will normally only be two DCEs between a pair of DTEs.

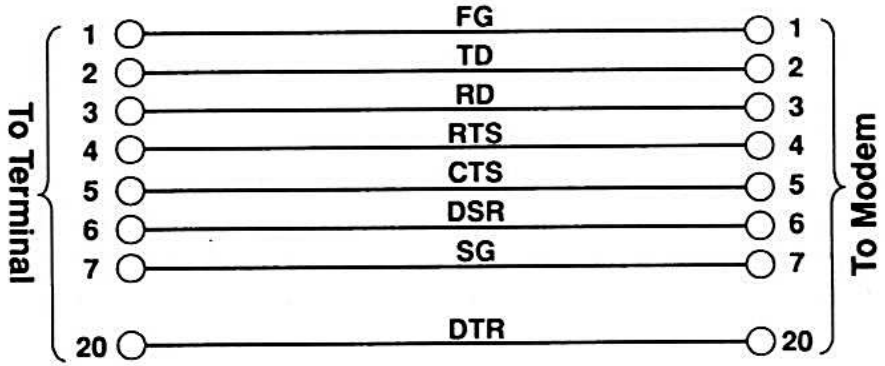
An alternative definition separates the two by what type of signals they can work with. DCEs convert RS-232 signals

into a form that another communication medium can use, e.g. the telephone line. DTEs on the other hand can only send, receive and work with RS-232 signals.

It is important to know whether a piece of equipment is a DTE or DCE because the RS-232 ports are wired differently on each. When the standard was written it was envisaged that a line of communication would look something like this



The RS-232 standard makes the wiring between a DTE (a computer or terminal) and a DCE (a modem) the easiest to configure because the wires go straight across. Pin 2 on the computer connects up with pin 2 on the modem's RS-232 port.



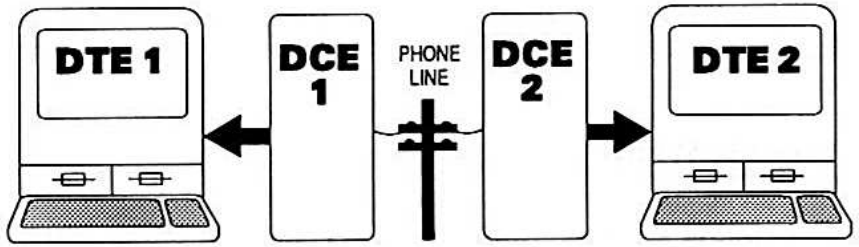
All the wires apart from Frame and Signal Ground are 'paired up' by function. Each wire in a 'pair' handles the same type of signal but in different directions. For example RD and TD (these are sometimes known as TxD and RxD), TD carries the outgoing data while RD carries data going the other way. Other 'pairs' are RTS/CTS and DTR/DSR.

Although the RS-232 standard allows for a 25-way connection all 25 wires are rarely used. The wires are connected to pins in the plugs/sockets at either end of the wire and are commonly referred to by the pin number that they need to be connected to. It is possible to communicate with just three wires connected but because the Organiser can use all the handshaking (flow control) facilities that RS-232 offers it normally has eight wires. If you do not need the hardware handshaking wires connected you could just connect the four wires that go to pins 1,2,3 and 7.

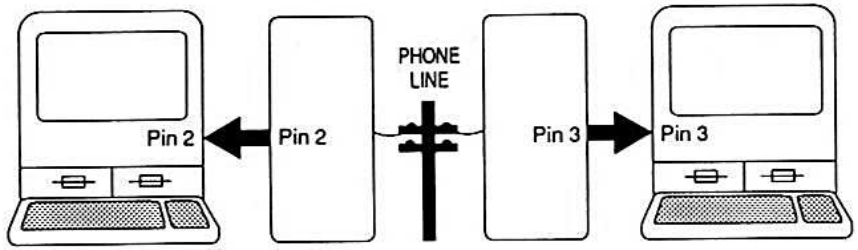
The Organiser Uses:-

pin no.	name	function
1	FG (frame ground)	Earth
2	RD (receive data)	Carries incoming data
3	TD (transmit data)	Carries outgoing data
4	CTS (clear to send)	RTS/CTS handshaking incoming
5	RTS (request to send)	RTS/CTS handshaking outgoing
6	DTR (data terminal ready)	DTR/DSR Handshaking outgoing
7	SG (signal ground)	All signal's common return path
20	DSR (data set ready)	DTR/DSR handshaking incoming

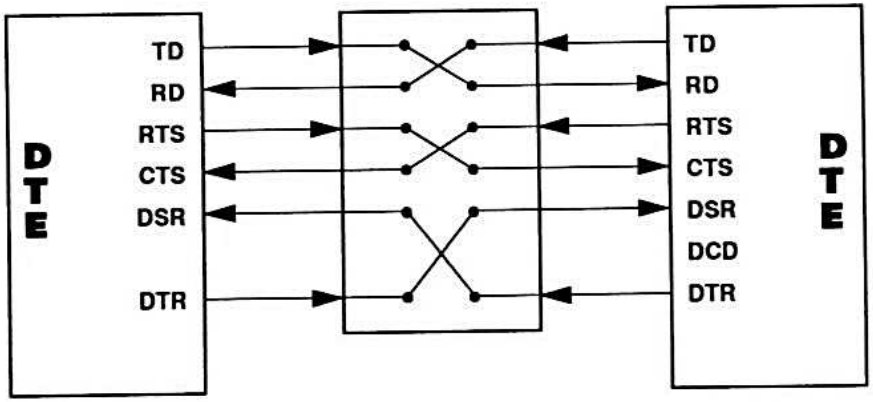
If you are trying to connect two DCEs or two DTEs together the pairs of wires have to be crossed to make them think they are talking to their opposite number even when there isn't one in sight. This is because in a total line of communication which has two DCEs connected together by a telephone wire.



one modem (DCE1) will be transmitting while the other (DCE2) will be receiving. If the data is being transmitted on pin 3 it will be picked up by DCE1 on its pin 3. But on the other side of the telephone line DCE2 will pick up the signal on its pin 2 and it will then go to DTE2's pin 2 as well.



When two of the same are talking to each other RD on one should be wired to TD on the other, RTS should be wired to CTS, and DTR should be wired to DSR.



What is What

If we follow the general rule that DTEs occupy either end of a communication a DTE should be either a computer or printer. On both the line of communication stops with them and does not go through and on to something else. DCEs on the other hand, can 'normally' sit on the line and send and/or receive data both ways.

So according to the rules a DCE is a modem.

Rules are there to be broken however and computer designers and manufacturers seem to like breaking them! The Organiser is a good example - or a bad one depending on which way you look at it - because although it is a computer and therefore should be a DTE, it isn't. It has been wired up as a DCE. It was thought, probably quite rightly, that us users would want to use the Comms Link either to send data to be printed or to a bigger computer, much more than we would be sending data down the telephone line. It would therefore be easier to connect the wires straight through for the majority of the time and only have to cross them over if you are talking to another DCE 'normally' a modem.

Null Modem

The crossed wires are sometimes referred to as a null modem or a null modem cable. This is because if you were dealing with the situation anticipated by the standard, you need to cross the wires between the pair of pins (see section on DTEs/DCEs above) where two computers are talking direct rather than through a pair of modems. The crossover is actually taking the place of the pair of modems and as null means nothing a 'null modem' cable means a 'no modem' cable.

Simplex, Half / Full Duplex

These terms describe the direction that your system of communication will allow data to travel in at any one time.

Simplex will only allow data to flow one way (from A to B) and not in the reverse direction.

Half duplex will allow data to flow in two directions (from A to B and B to A) but only in one direction at any one time. This is one situation where Handshaking or flow control is needed because both ends of the line need to know when to send data and when to receive.

Full duplex will allow data to flow in both directions, all the time. So data can be flowing from A to B at the same time as another batch of data is travelling from B to A.

The Terminal Emulator on your Organiser's Comms Link uses full duplex.

The Organiser can use any of the three modes of operation. Which one is actually being used is normally determined by the 'other' participant in the communication.

Asynchronous and Synchronous Transmission - 'Timing'

'Timing'

Timing is about how fast the information can travel between the two pieces of equipment. The speed is at least partly determined by how the flow is regulated.

Synchronous

Synchronous transmission is not used by the Organiser but it allows data to be transferred smoothly and fast because the speed and the flow are controlled globally by a timing signal which is sent along the clock wire at an optimum speed that the whole system can cope with. It has the advantage of speed over asynchronous transfer partly because it needs less control characters and therefore can pack more data across the wire in the same time.

Asynchronous

In an asynchronous transfer the speed is not constant but jerky (it is sometimes called On/Off or Stop/Go communication). Each binary character is framed by control 'bits' at the beginning and end so that the receiver knows when to start. All the frames are sent across the link singularly.

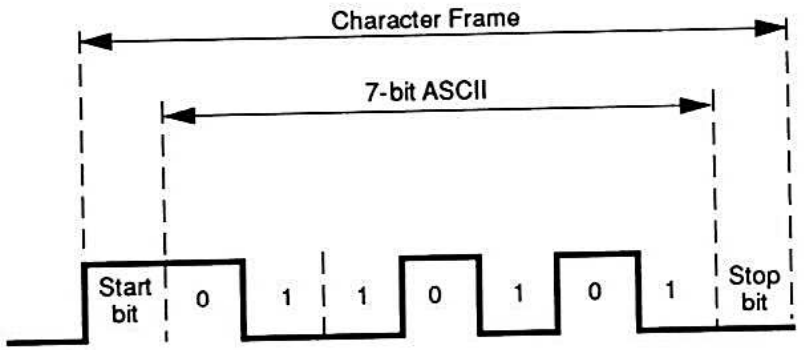
The speed of flow is controlled by the Baud Rate and the framing control parameters. These are selected from Setup, which you can get to from the COMMS top level menu.

Framing

When data is being sent serially there needs to be agreement between the two communicating machines about the precise format of what they are sending and receiving. The frame contains all the information they both need to decipher the incoming data. A frame consists of a Start bit, seven or eight data bits, an optional Parity bit and one or two Stop bits.

The data bits contains the character which is being sent. All the other parts of the frame are added on by the communica-

tion software to aid recognition of the data, and hopefully, the weeding out of bad frames that get corrupted on the way.



The Start bit alerts the receiving unit that there is a frame on the way. It says:

> 'Wake up you lazy lot. What is coming immediately behind me is a piece of information.

The next group of bits in the convoy are the data bits. They hold the binary code for the character that is being sent. These are followed by one or two stop bits which tell the receiving unit where the end of that frame is. There might be a Parity bit in between the data byte and stop bits. The parity bit is a crude form of error checking which can guard against about fifty percent of the possible errors that could change the character being sent, for example, from an 'a' to a 'c'.

CHAPTER 2

Forging the Route

The route between the Organiser and whatever you want it to talk to will have two or three component parts. These will be:-

A A Comms Link
B Software for your other machine
C Possibly an additional cable or adapter

If you have an IBM P.C. or a Mac and that is all you want your Organiser to communicate with, everything you need

is supplied with the Comms Link. If you want to communicate with anything else you may have to buy B and C in addition to the Comms Link.

A. The Comms Link

The first thing you will have to do to get your Organiser to communicate is to plug in the Comms Link. There is a set procedure that you must follow when doing this otherwise you could damage the Comms Link, or your Organiser, OR BOTH, so be careful!

Enough Memory?

Before you start, it might be as well to check that you have enough space in your Organiser's internal memory for the communication program to work with. It needs about 4 'K' which is over half of the total memory available on a CM.

If there isn't enough spare memory the Organiser will put the 'OUT OF MEMORY' message on the display. You will have to clear some memory before you can start communicating. To check how much free memory you have go into the CALCulator and type 'FREE'. If it says that you have less than '4000' bytes you will have to delete or move some records or files off the internal Ram. When you have done that check the figure again to make sure you have cleared enough space. Once you have room for it to operate you can plug in the Comms Link.

Plugging the Comms Link into your Organiser

1. Turn your Organiser Off.

2. Take any Rampaks out of the device compartments - otherwise they could be formatted accidentally.

3. Plug in the Comms Link by :-

   a. Opening the Expansion Port door at the top of your Organiser.

   b. Holding the bigger end of the Comms Link with the silver Comms Link label uppermost.

   c. Align the plug with the socket and then plug it gently in before switching the power on.

   d. Then and only then, plug it into the mains - if you want to conserve batteries - by:-

      Plugging in the three pin mains plug and switching the power on

      and then -

      pushing the small plug on the mains adaptor into the socket at the back of the Comms Link/Organiser plug.

4. Turn your Organiser back on and then press the ON key a second time to load the Comms Link software which is incorporated into the plug. COMMS should now appear on the bottom of the main menu just before OFF.

5. If COMMS doesn't appear on the main menu

   a Turn the Organiser off
   b Check that all the plugs are inserted properly
   c Turn it on again and try loading the software.

If COMMS still does not add itself to your main menu

   d reverse the above procedure and start again.

5. If Comms is on the menu turn your Organiser OFF again and put any Rampaks back into the device compartments.

6. Then turn your Organiser back ON. It is now ready to work with the Comms Link.

B. Software

There will be a need for software to control both the communication itself, and the two pieces of equipment (even printers need to know what you want them to do although they don't need an actual program inserted into them), to make the passage of information possible. The Comms Link has it's own software on board but if you are communicating with another computer that will need a communication software package loaded and ready to run.

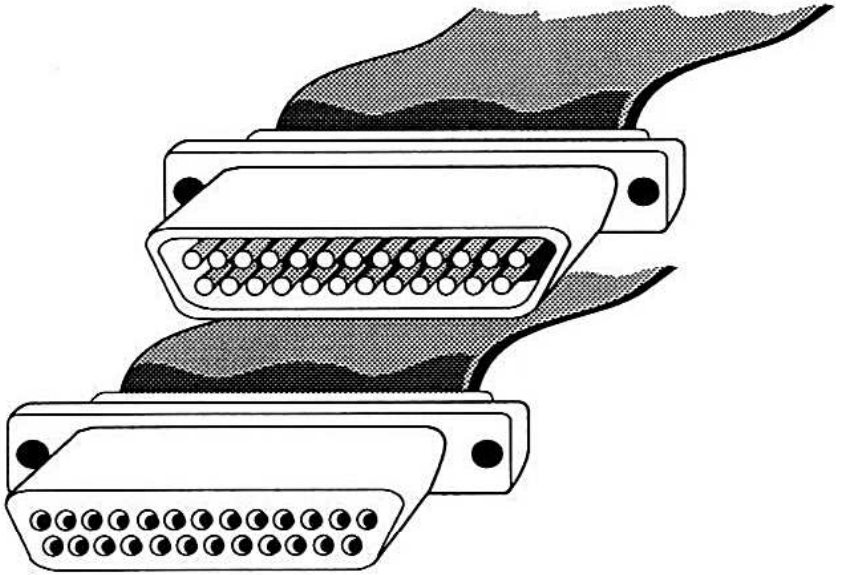
C. Additional Cable and/or Adapter

The cable and the second plug on the end of the cable attached to the Comms Link may not be the type that the 'other' unit needs. There are two methods of communicat-

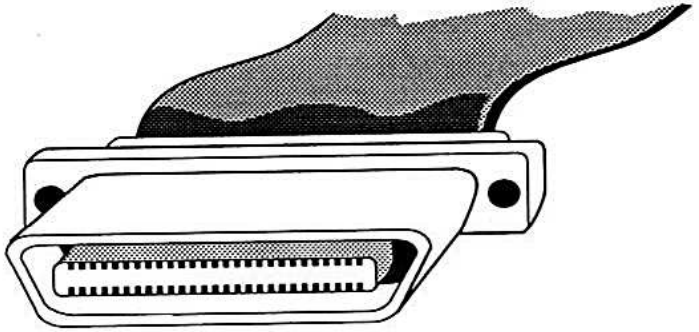
ing information between electronic equipment down cables. They both need different wires and different types of plugs and sockets. When you have sorted out which standard your 'other' unit works on you may need to get the correct adapters to go on the end of the Comms Link.

So first things first. How do you find out which standard the 'other' unit works on? Well the first thing to do is have a look and see what its communication port looks like. You will probably find one of the following:-

A A rectangular shaped (female) socket with a number of holes arranged in a 'D' shaped formation. or you could have a plug with pins sticking out of it in the same 'D' shaped formation.



B A rectangular socket with one long thin groove in the centre which takes the connectors and a second groove which goes all the way round the socket which is where the rest of the plug locates into it.



If you have (A), your receiving unit works on the serial standard of communication. However you are not home and dry yet because it could either be working on the RS-232 or RS-422 standards. Both of which the Comms Link can use. You may be able to tell which it is using by looking at the socket but there again you may not!

Although by rights the RS-232 standard has 25 wires each with its own hole in the socket (or pin on the plug) you may have any number of wires and holes on the socket because not all 25 are normally used. Likewise with the standard RS-422, this should have 37 wires, and therefore 37 pins on the plug and 37 holes in the socket but very often doesn't! Unless the socket has more than 25 holes - in which case you can be certain that it is using RS-422 - you may have to resort to digging out the manual to check. If you can't find the RS standard listed in the Index or the Contents of the manual look for words like 'Interface' or 'Connections'.

If you have (B), your 'other' unit supports parallel or Centronic communication. The Comms Link uses the serial RS-232 standard and has an RS-232 plug on both ends. If the 'other' machine has a different type of plug/socket it will probably use either the RS-422 or parallel standard. You will need to get a converter to enable your RS-232 plug to fit in their socket. Parallel to serial converters are obtainable 'off the shelf' but if you are unlucky enough to be dealing with RS-422 (which is still fairly rare) you will have to get an adapter made up because they are not readily available.

IBM Plugs and Sockets

If you have an IBM compatible computer and you cannot find either a 9 way or 25 way male socket you will have to get a plug in Serial RS-232 adapter fitted which will include a serial port.

If you do find a socket it may not be like either of the ones we have described above. IBM ATs do not use the standard plugs and sockets at all. They use a nine way 'D' type socket for the RS-232 connections and a twenty-five way 'D' type socket for Parallel printers - not a centronics plug at all.

Sex?

It is quite possible that even if the plugs/sockets that you want to connect up are both the same type of communicators

they could be sexually incompatible! There could be a plug or a socket on both units with no way you can fit them together. This is easily remedied because all you need is the appropriate 'gender changer'. These are small boxes with either two plugs or two sockets on either side. They are available for all the configurations. You might also find a ready made cable with the plug/socket configuration that you need.

Moving on

The above is all you need to know to get the hardware link sorted out. We will explain the differences between the serial and parallel systems of communication in greater detail in the Delving Deeper section further on in this chapter.

Matching Up Your Software

Having the physical link (route) sorted out is a good step in the right direction to getting your Organiser to transmit and/or receive information to or from another piece of equipment. But it is only the first of many steps that you will have to take.

The next major thing you have got to do is enable the two lots of software to match up so that they can work together. Before you can begin to do that however, you need to know what the Comms Link has got and what it needs to see at the other end. So the next chapter will be about the Comms Link

Delving Deeper

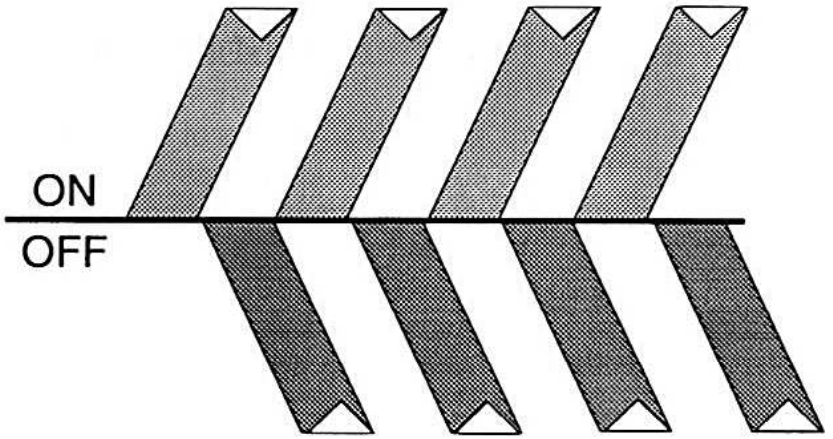
Why are there Different Standards?

The different standards are really snapshots of evolutionary development. They are attempts to create common ground between all manufacturers using the best technology available at that time. But within a year the technology has marched on making much better solutions available to the standard setters. Each standard has advantages and disadvantages. One may be faster while another may be more accurate. None of these need worry you because you probably won't have a say in which one you are going to use.

Your computer and any accessories already have the standard built in.

How do Computers and Their Accessories Communicate?

Computers communicate the same way as they do everything else, with electrical impulses. These impulses can be different levels but there are three states that a computer is interested in. A signal can be positive, negative or zero volts which is neither positive nor negative. We only need to know about the positive and negative signals for now. A computer's memory consists of millions of minute electrical switches which are grouped together in blocks of eight. Each of these switches can either be turned on or off. To accommodate this a new two digit numeric system was developed called 'binary' which just works with '1's and '0's.



If a positive signal is sent to a switch it will switch itself on - which will signify a '0' - whereas if a switch receives a negative signal it will switch itself off which will be interpreted as a '1'. One switch on its own is known as a 'bit', four bits together are known as a 'nibble' and eight bits joined together are known as a 'byte'. The total quantity of internal memory on your Organiser is either eight, thirty two or sixty four thousand bytes - 'K', or if you want the full unabridged version, kilobytes - which is a staggering sixty four, two hundred and fifty six, or five hundred and twelve thousand bits! The exact number depends on which model you have. You can store approximately the same number of characters as you have bytes because the codes that make up a character take up one byte.

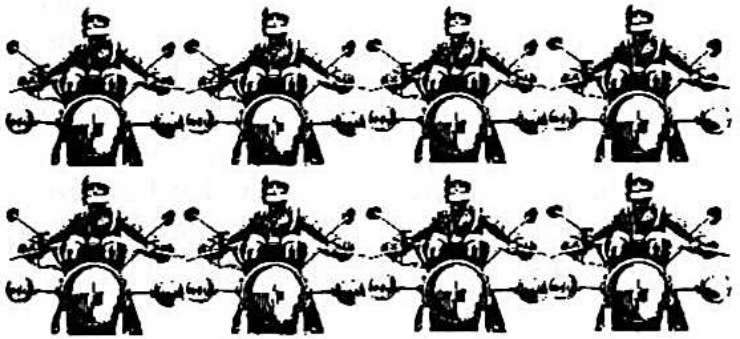
Each time the computer receives an input, when you press a key on the keyboard for example, it effects a byte in memory, turning each bit within it either on or off to the appropriate configuration for the character in binary. To get from a character on the keyboard to the correct binary number the computer has to go through two levels of translation. It first has to translate the character that we have pressed into an ASCII code. Once it knows the code it can then translate that into the binary number and store it in the on/off switches in the byte of memory that has been allocated. All of the translating happens incredibly fast so that us mere mortals are not aware of it happening.

When you are sending files from your Organiser to another computer or a printer the binary codes (which is how it is stored anyway) for each character in the file is sent down the wire. The binary codes are only translated back into characters that we understand when we need to see them.

'Bitsize' or 'Bytesize'

The RS-232 (RS-422) and Centronics standards have chosen to send these binary codes in different ways. RS232 sends 'bitsize' pieces down one wire (serially) so that the receiving unit does not know the whole number that it is being sent until it has received the last bit of that byte. In our military communication analogy this would be where the message is split between lots of dispatch riders who are then sent down the route on their motor-bikes one after the other.

Units using the Centronics or parallel standard send all eight bits together (in parallel) down eight separate wires. You need a wider route for this type of communication with enough room for eight motor-bikes to travel side by side in formation!

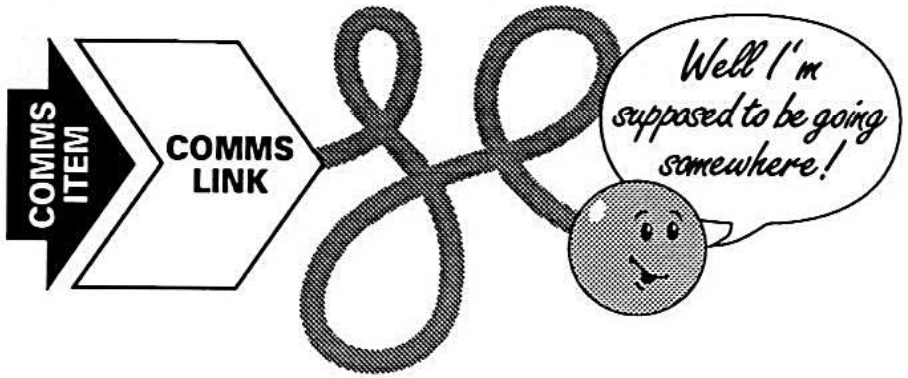


Parallel communication is much faster but it is normally used for one way communication. Parallel interfaces are used to enable computers to send data to printers and it is very rare for printers to need to send data back the other way (although they may need to send back handshaking signals). Parallel communication was designed for 8 bit data transfer and has no error checking.

Serial interfaces, on the other hand, are able both to send and receive data which means they need a more sophisticated traffic control system. Since serial communication was designed to carry data over longer distances and over more difficult links (such as telephone lines) they have a limited error checking facility available.

CHAPTER 3

The Comms Link



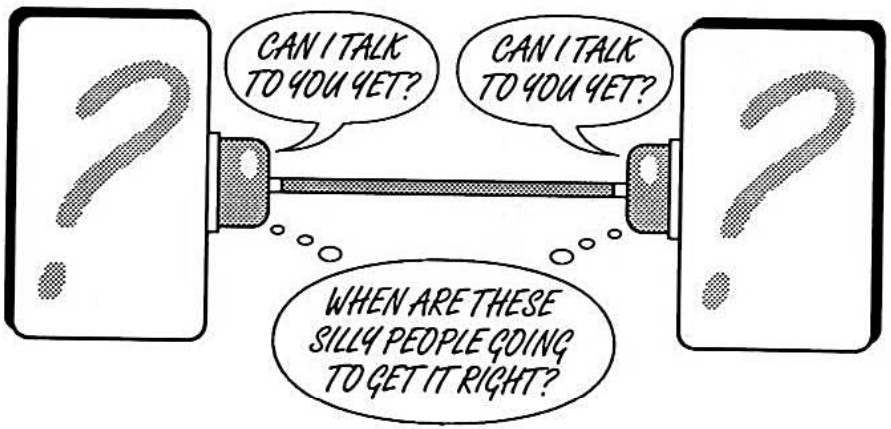
Selecting COMMS

If you select COMMS from the main menu you will see a secondary menu with options ranging from Transmit through to Boot. There are two that are useful while you are trying to establish the lines of communication - Setup and Auto. You use Setup to alter the parameters so that the Organiser's communication software can be on the same wavelength as the other unit's software. Auto is a way of getting the Organiser to find out what parameters the receiving

unit is looking for. Auto is a definite asset but it doesn't cover all the parameters so you may still have to do some work yourself!

To Communicate Successfully

The biggest single factor for achieving successful communication is to make sure that both pieces of equipment agree how the data will be sent. The way you do this is to make sure that the communication parameters on both units match. To get the two units to talk to one another - or talk and listen - you need to set your Organiser to the same parameters as the receiving unit is using. You will need to read the relevant pages in the 'other' unit's manual to find out what parameters it uses. It may be possible to get the 'other' machine to give you a printout or display of the parameters it is using.



To find out what you need to look for, select Setup from the Comms menu. You will then see the fifteen different parameters that you need to line up with the other's settings.

To get the two sides set up to the same communication parameters - once you know what they are - you have two choices. You can either change each one by hand or you can select Auto and let the Organiser sort out the first ones and then alter the rest yourself.

Auto

When you select Auto your Organiser will automatically start trying to send a message to the other machine. It goes through every combination of the four parameters that it can handle Baud, Parity, Bits and Stop. It tries to send a short message, which includes the parameter values that it is currently set to, down the line. When it does get through you will have, either on paper or on the other computer's screen, the values that match. When that appears you can either stop Auto by pressing the ON key or you can wait until Auto has completed its cycle and then the Comms top level menu will reappear on the screen. In both cases you will then need to select Setup and put the values in.

That at least is the theory. When we tried it out for the first time it didn't find any of the Parameters for us.

It is possible that Auto won't work properly with some computers or their accessories because it uses codes which may be interpreted as control codes by the other machine. Even after it has found the matching codes for you, you may need to change some of the parameters. If a control character is sent accidentally it could change a setting even after a match has been made.

Using Setup

When you select Setup you will see the parameter menu, which we will call the 'Setup Sheet'.

The Setup Sheet

The Display

On the SETUP sheet display you will see the top two or four parameters, depending on which model of Organiser you have, and their present values. You can move between the rest of the fifteen in the normal way with the Up/Down arrows. A small right arrow symbol to the right of the parameter's name takes on the cursor's role of showing you where you are.

The Keyboard

Inside SETUP some keys change from their normal functions.

ON
The ON key puts the parameter back to it's default value, if you have altered it. If you press it before you have altered any of the parameters nothing will change.

If you are changing a parameter which has a wide range of choices and you have made a mistake, pressing ON once will put the last value you entered back in. If you press ON a second time the factory defined default value will be put back for you.

LEFT/RIGHT Arrows

The LEFT/RIGHT Arrows will change the value if there are only a limited number of alternatives. You need to press the LEFT or RIGHT Arrows until the one you want is put into the display. If you press the arrows and nothing happens it means that there is a wider range of alternatives and you will have to type in the one you want.

EXE

If you press EXE once it will clear the existing value and put you into your Organiser's normal line editor. You can then type in the value you want. Pressing EXE a second time will ask the Organiser to check that the value you have entered is acceptable and then put it into the Setup sheet. If the value is not within the correct boundaries you will find that you are still in the line editor and you can have another bite of the cherry!

The Parameter's Default Values

The default values are what your Organiser was originally programmed with -

Before you change anything on the Setup Sheet it should look like this:-

BAUD	9600
PARITY	NONE
BITS	8
STOP	1
HAND	XON
PROTOCOL	NONE
ECHO	HOST

WIDTH	NONE
TIMEOUT	NONE
REOL	no default
REOF	no default
RTRN	NONE
TEOL	no default
TEOF	no default
TTRN	NONE

The first four settings are the most important. If they are set to the wrong values we can guarantee nothing will get through. If you get those right but some of the others wrong something will get through although it may not be the same as you sent!

Spoilt for Choice?

In some situations both units will be able to use more than one of the options available. If that happens you can either use one unit's default settings, experiment to see which settings work the best, or make an educated guess about which option to choose after reading the 'Delving Deeper' section of this chapter below.

Leaving the Setup Sheet

Once you have changed the Setup parameters to what you think they should be, leave the Setup Sheet by pressing Mode. The Setup menu will then appear on the display. At this point you only need to select EXIT which will return you to the Comms top level menu.

Delving Deeper

The Parameters

The parameters can be broadly divided into three groups.

The first group control the physical (hardware) side of the communication line. These define things like the speed that the data can be sent and how both units can recognise the data when it arrives and check that it is intact.

The second group help the software at both ends of the link work together. These determine things like the file transfer protocol, the markers that show the software where the end of lines and end of file are. They also cope with character translation.

The third group is made up of miscellaneous parameters. These may only affect one type of communication like Echo (which you normally use when your Organiser is a terminal), or their functions may be purely cosmetic like Width.

The Hardware Control Parameters

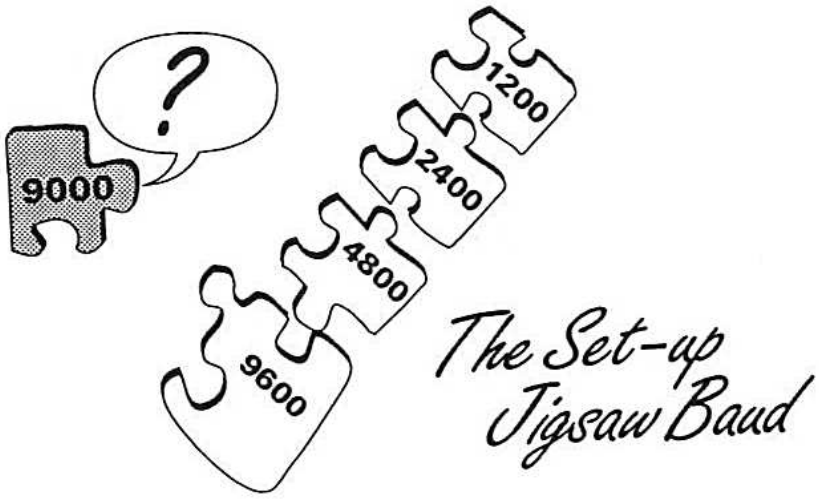
The Baud Rate

This sets the speed at which the data travels. Both the transmitting unit and the receiving unit must be set to the same speed. There are a wide range of speeds that your Organiser can work with and you may well find it easier to set it to what the 'other' machine is set to rather than trying to alter it on the 'other' unit.

The Baud rate tells you how many bits can be sent per second. The Organiser sends and receives approximately ten bits for each character. There are seven or eight 'data bits' per character but each byte of data is enclosed in a 'frame' which can contain up to eleven bits. The actual number of bits per frame depends on what you have got the Bits, Parity and Stop parameters set to. If you divide the Baud Rate by ten the result approximately corresponds to the number of characters being sent per second.

You can select the Baud rate on your Organiser from the following alternatives:- 50, 70, 110, 150, 300, 600, 1200, 2400, 4800 and 9600.

Normal speeds for printers are 1200 and 9600.



Normal speeds for modems are 300, 1200 or 2400. Modems can only accept a limited range of speeds, whilst computers

can function over almost the whole range. If you are communicating with another computer via a modem it makes sense to set both computers to the modem's baud rate rather than trying to do it the other way round.

Most other computers can operate at 9600 when there isn't a modem in between. You should experiment however to find the fastest speed that the two can exchange data without errors. If you start getting a number of errors drop the speed systematically on either side of the line until the errors stop.

Some modems and electronic mail services have two baud rates, one for transmitting data and a different one for receiving data. Your Organiser has only one setting and has to do everything at the one speed. If you want your Organiser to communicate with equipment that needs to use two speeds you will need to put a modem in between that can hold the information that has been sent to it in a buffer - a block of memory that stores incoming and outgoing data temporarily until it can be moved - while it changes the speed. It can then send it out at the speed that your Organiser needs.

The Framing Parameters

The next three parameter are related and determine how the character frame is made up.

Parity

Parity is next on the Setup Sheet. It can be added into the frame in between the data bits and stop bit(s). There are five settings available for this parameter:- None, Odd, Even, Mark and Space. The last two are less often used; the

Comms Link manual says hardly ever, but one of our 'other' computers uses Mark so it cannot be as rare as all that!

Parity set to None

If the receiver dictates that you select None then a parity bit is not added into the frame. This means that no error checking will be done by the hardware. If the receiving unit can use either Xmodem or Psion protocol having a parity bit in place is not necessary as these methods of communication employ sophisticated error checking and correction techniques in any case. If the receiver cannot cope with either of those protocols but can work with Odd or Even Parity you might like to experiment to find out which works better, either Parity set to None, or Odd/Even on both units.

Parity Set to Odd or Even

In its error checking role Parity gives the receiving unit the chance to check that the frame it has just received has not been corrupted by bad connections or power surges. Parity enables the sending computer to tell the receiver that the frame being sent has a data byte which contains either an odd or even number of binary number '1's, depending on which setting you use.

If you are using Even parity and the data byte being sent has an even number of '1's, the sender will set the parity bit to a '0' because the receiver will be looking for an even number of '1's which it has already got. If however the data byte has an odd number of '1's the sender will set the parity bit to a '1' to make it up to an even number. The receiver then knows immediately that something is wrong if it receives a data byte that has an odd number of '1's.

Limitations of Odd Or Even Parity

We said earlier that using Parity as a way of checking that the correct data has been sent is a crude way of doing it. Odd or Even parity will only detect that something has gone wrong half of the time. This is because it can only detect a faulty byte if there are one, three or five errors in the byte. If there are two, four or six errors they will not be noticed by the receiving unit at all! This is because if you are using Even Parity an error will only be detected if there are an odd number of '1's. But if there are two errors the number of '1's will be even so the receiving unit will think that everything is rosy. In that situation you will end up with garbled characters that have slipped through the net in broad daylight! So using parity, providing that is what the receiving unit is looking for, reduces the number of errors down to a half of what they would otherwise be. Cutting the risk of problems down further - and picking up the frames that parity has found to be wrong but doesn't have the ability to do anything constructive about - has to be done with software, which is what the Protocol parameter is all about.

It should be added here that errors are fairly rare unless you are trying to communicate down a noisy telephone line. In normal conditions it is perfectly possible to transfer a file successfully without either parity or protocol to catch the errors because there aren't any there to catch!

Parity Set to Mark or Space

Mark and Space do not work in the same way as Odd and Even. Mark will always set the parity bit to '1' while Space will always set it to a '0' regardless of how the rest of the byte is made up. They take on the role of an extra Stop bit and will probably only be used if you are using seven data bits in

the byte and need another bit to fill the eight data bit, see Bits below for more details.

Parity Error Detected?

When the receiver detects a Parity error it will throw the bad byte away. If you were not using any protocol you would end up either with a missing character in the transferred file or the Organiser could break the communication and put 'DEVICE READ ERROR' on the display. If your Organiser is emulating a terminal the bad byte will be converted to a filled in square. So here at least you know exactly where the problem is.

Bits

The Bits Parameter controls the number of data bits you want to use in the frame. The data bits are the ones that carry the binary code for the character. There are now eight bits in a byte but before eight-bit computers hit the scene the previous generation could only handle seven bits. The original ASCII code worked with seven bits rather than eight so all the characters that we normally use were covered in seven bits.

When eight-bit computers came into operation, because of the nature of binary, the extra bit effectively doubled the number of characters that could be generated. Instead of rewriting the ASCII table, however, the normal characters were left where they always had been in the first half of the table, leaving the second half free for extra characters, although it has never been standardised. It is up to individual manufacturers to define their own characters in that half of the table. Consequently you will only be using the first half of the table for normal communication.

The eighth bit is used by protocols to help either end of the communication line to verify the byte they have just sent or received. So if you are using a protocol the Bits parameter must be set to eight.

If you set the Bits parameter to seven on your Organiser it will automatically 'lose' the most significant bit (which holds the code for the higher number in the byte) - by not sending it. Effectively this means that the Organiser will automatically add a Parity Space on top of whatever parity setting you have asked for when you set Bits to seven. But here comes the clever bit. If you want to force your 'communicators' to only use the characters on the second half of the ASCII table - which perhaps has a special font for your printer - you would set the Bits parameter to seven and Parity to Mark.

Stop

The Stop bit(s) signal the end of the frame to the receiving unit. You can either have one or two Stop bits depending on what the receiver is looking for. You normally need two Stop bits if you are sending data at slow speeds (baud rates 300 or below). If you are using a faster baud rate setting the Stop bit to '1' should be appropriate.

If you are in any doubt about which setting to use, set it to two because that will still allow the frames to go through, although at a slightly slower speed since each additional bit takes time to transfer.

If however, it is set to one on the Organiser when the receiving unit is expecting to see two Stop bits the receiver will register a 'framing error' and will discard the entire frame, Since it will discard every frame in the same way nothing will get through at all!

Limitations In Frame Size

Your Organiser can only work with frames that are between ten and eleven bits long. If the combination of parameter values that you have selected on the Setup Sheet will give you a frame that is less than ten bits or more than eleven the Organiser will adjust it automatically by adding or taking off one of the Stop bits.

For example, if you have selected 7 data bits, Parity set to none and 1 Stop bit the frame will only consist of 9 bits, including the start bit at the beginning, The Organiser will make it up to ten by selecting two stop bits regardless of what you asked for on the Setup Sheet.

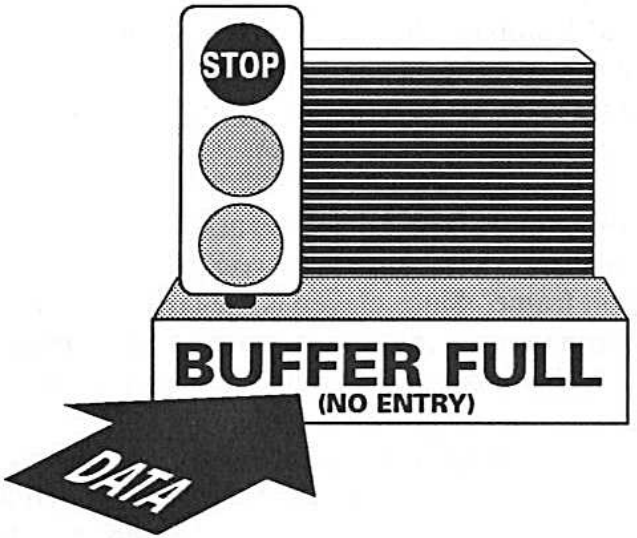
At the other end of the scale if you ask for 8 data bits, a Parity bit of whatever persuasion and two Stop bits, you have got a twelve bit frame. Since eleven is the largest frame your Organiser can work with that is too big. Your Organiser will automatically make the second stop bit invisible!

To set the receiving unit up to the same values as the Organiser you must take what the Organiser is going to do automatically into account and set the receiver to the values that it will actually receive rather than what you have set it to!

You may have a slight problem if the receiver needs two stop bits on the end of a frame together with eight data bits and a parity bit. One way of dealing with that is to take out the Parity bit because most of the things you might want to communicate with can work with Parity or without. If it is included as the receiver's default setting you may have to do some detective work to find out how to turn it off.

The opposite is also true. If the receiver is set to look for 7 data bits, no parity and 1 Stop bit (and will not work with 2 Stop bits) you will have to find a parity setting that they can both work with to take up the additional bit that the Organiser needs.

Hand



The Hand parameter allows the receiving unit to control the flow of information that it is being sent and therefore keep it to manageable levels. Printers and modems generally can't pick up the information as fast as computers can send it. Any computer equipment capable of sending and receiving information this way has a small buffer which is used to stockpile information that is on the way in or out. From there the incoming data has to be either printed if it's a printer, translated into audio signals by a modem, or on

another computer it would have to be moved to more permanent memory where it is no longer in the way of other data trying to get into the buffer behind it.

If the information is coming in faster than it can be moved out the buffer fills up and can't take any more. If the "sender" carries on sending after that point, the "receiver" will not know that anything has been sent so consequently that section of information will be lost. The various handshaking systems give the "receiver" the ability to tell the "sender" to stop sending while that batch of information is being forwarded and then to start again when there is room in the buffer for more.

Your Organiser can use three different handshaking systems. These are XON/XOFF, RTS/CTS and DTR. They can be either used individually or they can be combined together.

Your options range from:- None, through the three systems on their own, to combinations of two together, and finally to All three together.

XON/XOFF

XON/XOFF is the only one of the three that does not have a direct 'wire through' connection between the two communicators. It works by allowing the receiving unit to send control characters that work like traffic lights. XON is green for 'go' and XOFF is the equivalent of red for 'stop'. When the receiver wants some data it will send an XON to the Organiser to tell it to start sending. When its buffer is full or nearly full it will then send an XOFF to halt the flow while it copes with the traffic jam. When the bottleneck has

cleared it will send another XON signal and the sender will start the flow again.

XON/XOFF are ASCII control characters and so are unprintable. They are included in the first 32 ASCII codes. To make things interesting they are not known as XON/XOFF but are called 'DC1' and 'DC3' respectively - DC is the abbreviation for 'device control' and this type of handshaking is sometimes known by the ASCII characters name DC1 and DC3.

Because XON/XOFF are ASCII characters you may not be able to successfully transfer a file that contains instructions in machine code such as an OPL file. The ASCII codes for DC1 and DC3 are '17' and '19' respectively and may be used for something totally different in machine code. If you are using either Xmodem or Psion protocol everything should be sorted out but if you cannot use one - i.e. the Protocol parameter is set to 'none' - you may have to use one of the other handshaking systems because if you use XON/XOFF the communication may stop half way through.

If your other computer can use XON/XOFF handshaking, you don't need to worry about the other handshaking systems. If you have too many ways of stopping the flow you may find a 'lock-up' situation occurring, where XON/XOFF turns it off and then because the hardware handshaking systems are also stopping it you will not be able to send an XON character to start the data flowing again.

When you are using XON/XOFF handshaking your Organiser will always make sure that there is enough room in the buffer for a little more data after you have sent an XOFF just in case the other machine doesn't stop sending data immediately. You may need this facility if you are

communicating with a multi user and/or multi-tasking machine because that may be away doing something else when the Organiser tells it to stop sending it data. If you are using one of the hardware handshaking systems in this situation, because they stop receiving immediately without any facilities for dealing with incoming data after that, you may lose up to eighty characters every time your Organiser stops the transfer. If you are communicating with a large multi-user/tasking system, XON/XOFF handshaking will probably give you the best results.

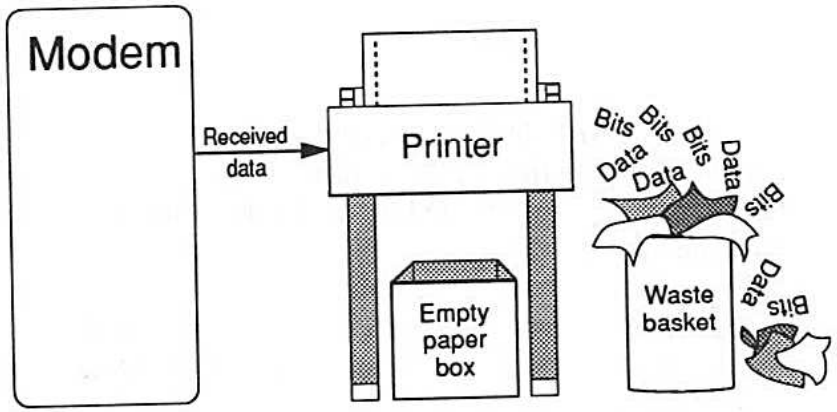
DTR

Unlike XON/XOFF, DTR is a hardware based handshaking system in which the flow of information is controlled by the signal received by the DTR or DSR pins on both communicator's RS-232 ports.

Although DTR comes after RTS/CTS in the Handshaking options available to you in the Hand parameter it is actually more important than RTS/CTS if it is being used. DTR, or rather it's opposite number DSR (on the Organiser), is the thing that starts the whole communication ball rolling. Once you have plugged in the Comms Link and have selected DTR handshaking here in Setup, the DSR pin at the Organiser's end will be telling whatever it is connected to, that the Organiser wants to communicate with it - whether in fact it does or not. Communication will not be able to start until the printer or whatever it is at the other end of the wire, says it's ready by putting the relevant signal on its DTR pin.

DTR and Printers

The Comms Link manual states over and over again that DTR handshaking is only normally used for printers. This, as you will see below, may be an over generalisation because it can be used as widely as any other hardware based handshaking system.



Printers do not normally need to talk back to your Organiser so they are designed just to receive information. The DTR/DSR line may be the way the printer tells your Organiser when it is ready to go to work or when it needs to stop, either because the buffer is full or because it has run out of paper. However the opposite may also be true and you may have a sophisticated printer which can use all the handshaking facilities that the Organiser can work with and a few more besides. Which is why as you will see from pages 151 or B.7 (depending on which edition you have) of the

Comms link manual the Psion Printer Adapter has the full compliment of connections.

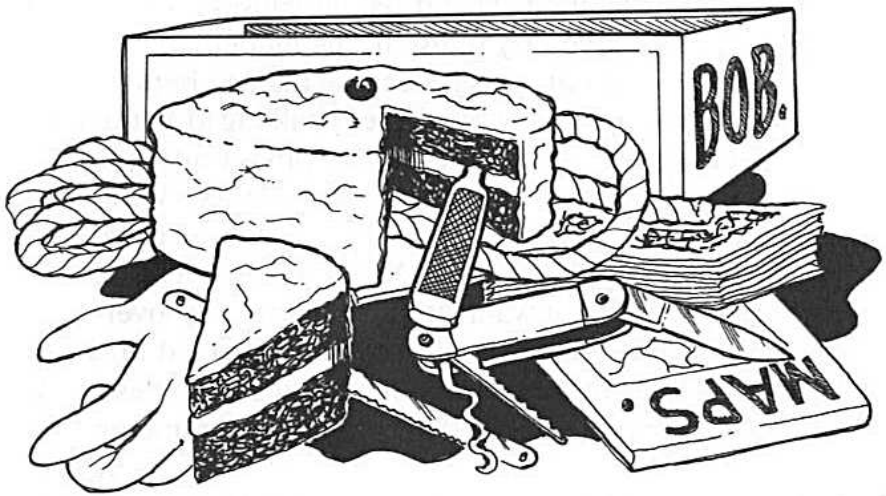
Just in case you are wondering what the TD line from the printer is used for, it enables XON/XOFF signals to be sent back from the printer when it cannot handle any more information. This of course is only used if that form of flow control is available and has been selected on the Setup Sheet.

DTR and Modems

The DTR/DSR method of handshaking causes problems for some modems. If you try to use it, the modem may not let the Organiser dial out. This is because the line between the DTR pin (on whatever the Organiser is talking to, in this case the modem) and the Organiser's DSR pin is waiting for the modem to say it has a message for the Organiser. With DSR connected there is no facility for the Organiser to tell the modem that it wants the modem NOW, because it keeps telling the modem that it wants it all the time. To overcome this the DTR/DSR line should not be connected in either direction and DTR, or any combination that includes DTR, or All, should not be selected for the Hand parameter.

A different problem arises if you have a 'smart' modem. They will stay switched on the whole time waiting for a message to come through and are able to then turn the attached computer on automatically to take the message. This is great because it allows the Organiser to conserve power by staying asleep until there is a call for it. To take advantage of this you will need DSR (pin 6) on the modem to be connected through to DTR (pin 20) on the Organiser. The modem can then give the Organiser a prod when a call is coming through.

If you need your modem to do both and you don't want to rewire the adapter every time you want to dial out, the best option is to have two adapters. One adapter would have pins 6 and 20 not connected, whilst the other would have it connected through one way to facilitate auto answering. If you have a Break Out Box - we will explain about BOBs in great detail later so don't worry if they sound like a prisoner's escape kit at the moment! - you will save the expense of a second adapter because all you need do is change the patch wires.



RTS/CTS

The RTS/CTS pins form the basis of another hardware handshaking system. If you are using it in conjunction with DTR handshaking it will only work after the DTR/DSR lines have established contact between the communicators. If you are

not using the DTR lines RTS/CTS can initiate the proceedings on its own.

In theory RTS/CTS Handshaking works like this:- your Organiser will send a signal from the RTS pin to whatever is on the other end of the line to tell it that the Organiser is ready to send some information. On receipt of that signal the other machine will put power onto its RTS or CTS pin - depending on whether it is a modem or another computer/printer - which will appear on the Organiser's CTS pin. Then transmission of data will start.

In practise it works slightly differently depending on whether full duplex communication as opposed to half duplex communication (see Chapter 1 if you are not sure what the differences are between these) is being used.

For half duplex the theory is correct except that the other device which is communicating with your Organiser may have a DCD signal on pin 8. DCD is the data carrier detect line which will be on while there is data being sent. The receiving unit will check its pin 8 and only send a CTS signal if the way is clear. The Organiser doesn't have pin 8 connected up, but if there is a modem between the two, that will have a DCD pin wired up. If whatever the Organiser is communicating with needs to see a DCD signal connecting it's pin 8 to the Organisers' pin 5 (DSR) will have the same effect.

For full duplex communication less checks are needed. Both RTS and CTS may be turned on all the time to allow a free flow of information. Even though full-duplex allows data to be sent and received at the same time, when transfering a file, you can only select one option (Send or Receive) from the top level menu on the Organiser. But

when you are using it as a terminal and talking to an electronic mail service it may well be working in full duplex. In that case the modem will hold RTS and CTS on all the time.

To summarise the various options available to you in Hand, we think that it is true to say that XON/XOFF and DTR are means by which both communicators can control each other whilst RTS/CTS is used much more as a way for the 'sender' to check whether it has access to the other end of the line.

The Software Parameters

The following parameters are more relevant to DTEs at the beginning and end of the communication line than they are for the DCEs in between them. DCEs (normally modems) just let the data through however many pieces of equipment there are on the line of communication as long as the hardware parameters match. DCE's do not need to know about protocols, or end of line/file characters because those just look like part of the data as far as the modem is concerned.

The Protocol Parameter

Protocols are the means by which files are checked to make sure they get across the link in their entirety. There are many different protocols on the market but the rule here is the same as ever, both sender and receiver must be using the same one. There is nothing to stop you sending a file across the link without a protocol. All that means is that the data will cross the link without any error checking being done and the receiver may not know when the transfer has finished. Most of the time the file will arrive intact. Printers seldom

use protocols and yet we take printing for granted and don't generally inspect the printed copy of a file to check for printer errors.

Protocols allow both sides to check the version they are being sent but they can also re-transmit bytes of data that were wrong the first time. This makes them much better at error checking than Parity, which will acknowledge that the byte is wrong but can't do anything about it.

Some protocols can also give a level of control to the Organiser over the 'other' communicator. The Psion Protocol is the most sophisticated of those available on your Organiser. That will send the file name and co-ordinate the whole transfer from the Organiser, but the number of machines that can use it are very limited. If you are working with no protocol you are at the other end of the spectrum. You will have to control both ends of the transfer manually and tell the receiving machine the name of the file because just the characters that are in the file are sent and nothing else. There will be a different way of using the Transmit and Receive options from the Comms Link's top level menu depending on what you have selected for the Protocol parameter.

Your Organiser has three options available in the Protocol parameter. We will discuss them in order of the probable amount of use.

None

Any computer, with the correct software and RS-232 connection installed, will be able to send or receive a file to or from your Organiser without a protocol. Although known as protocol 'None' on the Organiser it may be known as

'ASCII file transfer' - because you are sending bare ASCII characters - on the 'other' machine.

If you are printing you must normally have protocol set to 'none' because most printers will not work with a protocol.

File Transfer without A Protocol

When you are sending information without a protocol:- you will need to manually set up the computer at the other end. Normally you will also have to type in the 'filename' at that end as well.

You can either get the 'other end' ready to communicate yourself manually via its keyboard or, if the other computer is a long way away, you could get someone else to do it for you.

Down The Telephone line

If you are communicating with something that is at the other end of a telephone line, like an electronic mail service, you will have to send it the commands that you want it to carry out - to transfer the file - through the RS-232 port via the Terminal emulation facility. Once the 'other end' is ready, you can then send and receive files by either using the Organiser's Transmit and Receive options or the Capture facility.

Sending data with Transmit is no problem because the other machine will wait until you are ready.

Receiving data with the Capture option is also fairly straightforward because once you have selected Capture everything typed or loaded from either machine will be

stored in the Organiser's capture buffer. The only problem with that is that it records everything, commands (passwords prompts and menus if you are communicating with electronic mail systems) as well as data, which means that your Organiser's memory will fill up fairly quickly.

Receiving data with the Receive command from a 'remote' source at the other end of a telephone line is slightly more difficult. If you set the 'other' end up to send the data it would normally start transmitting straight away, and you would lose the beginning of the file that it is sending to you while you are trying to get back into the Comms top level menu to tell your Organiser to start Receiving.

There is a way round it. After you have set up the 'other' end so that it is ready to start sending the data, you can put it into pause by sending it our old friend XOFF (ASCII 19). You do not have to select XON/XOFF handshaking in the Hand Parameter if you don't want to use it generally because here you are using it just to manually control the other machine. You then have all the time you need to set up your Organiser.

When you tell the Organiser to start receiving it will automatically send an XON signal anyway which will start the other machine rolling. You must check that the REOF parameter is the same as the one that the 'other' end is using otherwise the Organiser will not know where the file ends.

Finishing the File Transfer Manually

You can stop the Organiser's Receive command by pressing the ON key. But as you will have no indication of where the end of the file is you could stop it before the end and lose some information.

Xmodem

Xmodem is a widely available communication protocol. Having said that your machine will probably be one of the very few that doesn't have it on board! If whatever you want your Organiser to communicate with does support Xmodem you may not be home and dry as there are two different versions which do the error checking differently. The one on your Organiser uses Checksums but the other sort is known as CRC.

The software we use on our IBM PC has both. Once we send or receive data to that - after selecting Xmodem - it tries using CRC first. If it can't get a response it will then try the Checksum version. If your 'other' computer only has the CRC version of Xmodem you will not be able to use Xmodem to communicate between that and your Organiser.

No Luck Getting Through With Xmodem?

If you have selected Xmodem and been unable to communicate, check what you have got the Bits parameter set to. You must have the Bits set to '8' for any protocol to work because the final bit is used by the error checking system. There are many different forms of checking that can be done but generally the last bit in a byte is used to store a control which allows the receiver to know that it has got the correct information. Without the eighth bit the error checking system will not be able to work.

If it still doesn't work, change the protocol to None (or ASCII) at both ends. If you can make them communicate with that you can be pretty sure that the 'other' end has got the wrong version of Xmodem.

Using Xmodem for file transfer does not allow you any remote control over whatever is at the 'other' end so you will need someone there to type in the commands for you. If you want to use Xmodem to exchange files with a remote source you will have to use the Organiser's Terminal facility to get the other machine ready the same as you would if you were not using a protocol at all. The only difference is that while you are using Xmodem you will not be able to use the Capture facility because Xmodem will send the file in a special format which needs to be decoded. Capture does not do anything with the data - it just catches it and stores it in its buffer.

Psion

To run the Psion protocol on your Organiser you must (as ever) have the software that can handle it on the computer at the other end. You will have a copy of the 'CL' program, which contains the Psion protocol, but you can only run this on IBM PCs/ATs, their clones, and also Apple Macs. Just install the program on the computer and run it.

The Psion protocol is the big brother of protocols available to your Organiser. It will sort out the parameter settings it needs automatically and will override the settings for Bits, Parity, Stop and the Hand XON/XOFF options that you have set up.

Psion protocol will also allow you to control the IBM PC or Apple Mac from your Organiser's keyboard. In fact you can't control 'CL' from the PC or Mac at all. Once 'CL' is running you have to operate it from the Organiser anyway. This allows you to send data between your Organiser and your office computer for example, with hundreds of miles between them via modems, single handed.

Conversion Parameters

Although the following six parameters fall broadly into the software group they could form a sub group all of their own. They exist because the software field is as unstandardised - if not more so - as the hardware field and different packages use different codes to do the same thing.

The first four convert characters to or from the ones that the Organiser uses to note where lines or files finish. The last two allow a similar translation process to take place on one other 'free' character.

The REOL And TEOL Parameters

These stand for Receive End Of Line and Transmit End Of Line. Use them to tell your Organiser what characters the 'other' machine uses to show where one line ends and another starts.

TEOLs and REOLs are sent or received at the end of Main and Xfile records. Records are stored in memory as one line with a tab character for each carriage return that you have divided it up with. TEOL and REOL are also used at the end of procedure and notepad lines.

These two parameters can either be set to None, one character, or two characters. If you are using two characters they must be separated by a comma. If both are set to None all carriage return characters will be disregarded and data will be sent in both directions in a continuous stream with no line breaks at all.

If you want the data broken up in a meaningful way you must put the ASCII codes here for the one or two characters you need. The default codes are '13' and '10' which are the codes for Carriage Return and Line Feed and are displayed as <CR> and <LF>. These two are normally the commands a printer needs to start a new line but they are used by most computers too. Unlike a typewriter which will do both operations if you press the carriage return key - or lever on a manual model - the carriage return command on a printer will only take the carriage back to the beginning of the line that it is already on. You need the Line Feed command to get to the next line down.

Once these parameters have been set, your Organiser will be continually trying to match the characters you have set it to with whatever is in the file. If you have set it to two characters, it will only put in a new line if it finds both of them together.

REOL and TEOL in the Terminal

If you are using your Organiser as a terminal for an electronic mail service you may only need to use the carriage return code for TEOL because the machine at the other end will send back a '13' and a '10' anyway.

When you press EXE in terminal mode your Organiser will send a TEOL automatically. When a REOL is received the incoming data will be displayed on a new line.

TEOL and REOL with OPL

TEOL will be transmitted by the TRIG$ and LIST/PRINT (on the PROG menu) OPL commands. It will also be transmitted by LPRINT providing the statement doesn't end with

a semicolon outside of any inverted commas. A semicolon at the end of the statement signals that there is more to go on the printed line and printing will continue when the next LPRINT statement is executed.

REOL is used by both TRIG$ and LINPUT$ OPL procedures. When characters are received, a continual attempt is made to match the REOL character or pair of characters. If a match is found, all characters received before the match are returned as a string. The characters matching REOL are discarded.

Something Not Quite Right?

Things can easily go wrong with the REOL and TEOL parameters if your Organiser and the other device are set up differently. With other parameters, if they are set up wrongly nothing may work at all so you will probably have no indication of what could be wrong. With these two you see things happening in front of your eyes, which makes diagnosis a great deal easier. It is also easier as you are only dealing with two variables on each side of the Comms Link.

You will know when you have not got the TEOL or REOL codes right because you will either get lines of text overwriting each other or you will have two, or more, clear lines between each line of text.

If lines of text are overwriting each other you have only got the Carriage Return codes in the Setup Sheet when you need the Line Feed codes as well.

If you have got Line Feeds but no Carriage Returns you may not notice anything wrong especially if the lines are all the same length and/or you have got the Width parameter set

because when the cursor gets to the end of the line it may automatically move to the beginning of the next one. You will only notice it when a line of text ends in the middle of the line because the next line will start immediately underneath where the previous one finished. You could end up with a 'staircase' of lines going down from top left to bottom right.

If you don't put either a Carriage Return or Line Feed you will get all the lines in a row across the sheet or screen.

If you get double line spacing it could be due to too many line feeds or it could be that the Echo parameter is set to the wrong option. It is common for printers to provide their own line feed codes and if that is the case your Organiser will not need to put in a Line Feed as well. Once again we must stress that you must check what the 'other' side of the equation is doing with all the parameters, including any switch settings as well as its Setup table. If you cannot find two sets of line feed commands anywhere it might be Echo causing the problem.

If you are getting four spare lines between each line of text Echo is probably set wrongly as well as a Line feed in TEOL and/or REOL that is not needed.

If your data records subdivide into two or more smaller records, it means that the database on the 'other' computer is being sent with Carriage Return characters where the Organiser doesn't want them. If carriage returns are being used as the TEOL and REOL as well as for another purpose which causes the record to split, you must try and get rid of the extra carriage returns in the 'other' computer's file before sending it to the Organiser.

The REOF And TEOF Parameters

REOF (Receive End Of File) and TEOF (Transmit End Of File) tell your Organiser what character the 'other' machine needs to see or use when it has sent or received the whole file. Both machines need to know when the whole file has been sent otherwise they could both be sitting around waiting for more for ages.

These two are set to the default ASCII code '26' which is displayed as '<sub>' in the Setup Sheet. If your 'other' machine needs something else you can set it to None, one or two ASCII characters. But please check carefully because different manufacturers appear to call ASCII '26' different names. We had never heard of a character called '<sub>' before. Our printer uses 'Control Z' and our PC uses '1A' as their end of file codes. When we checked their ASCII tables however, both were in fact ASCII '26'. Thank heaven for what little standardisation we do have!

Please note that if you set these parameters to 'None' any ASCII code '26's in the file will be deleted.

The RTRN and TTRN Parameters

These two are character translators. RTRN allows you to translate a character being received while TTRN works on data being transmitted. You can also use them to delete an unwanted character, i.e. one that one side uses but the other side cannot cope with.

As with all the other dual receive/transmit parameters you can set these to 'None' or, one or two characters. If you are putting in characters you should type in the ASCII codes for

the characters you want, even though your Organiser will put the actual character (or a representation of the character for those it cannot display) in the appropriate spaces.

'None' will not do any translation. If you put in one character your Organiser will remove it from all communicated data. If you put in two characters, with a comma to separate them, the first character will be translated into the second. The comma is important but it will not be displayed so you will only know you have forgotten it when the translation you have asked for doesn't happen!

An example of where you might use these would be to make your data file on your Organiser compatible with the database on the 'other' computer to enable you to transfer

files between them. Database files must have one designated character which defines where each field ends and the next one starts. When you are using Main or an Xfile the Organiser puts in a tab character every time you press the DOWN Arrow key to go to a new line in the record and stores the whole record as one line. Other database programs use other characters, 'comma delimited' files (where each field is separated by a comma and each record is on a separate line) are quite common. So if the database on your 'other' computer does use commas you will need to change commas to tabs one way and tabs to commas the other. To do this you need TTRN to be set to translate tabs to commas by typing in '44' (the ASCII code for a comma), a comma to separate them and then a '9' (the ASCII code for a tab character). The ASCII codes should be reversed for RTRN.

Tab characters that you send to your Organiser in procedures or LZ Notepad files will get translated automatically into spaces.

You can only translate one character in the body of the file per file transfer since you can't alter RTRN and TTRN in the middle of a transfer.

Need to Translate More than One Character?

If you need to translate more than one character and are unable to change them easily on the other computer you will have to send the file backwards and forwards as many times as you have characters that need changing. The first time you send it across the link you will translate one character. You then send it back to your Organiser with no translations being done. You are then able to translate another character on the next transfer. You could re-send the file like this,

after changing TTRN on the Setup Sheet, as many times as you need until the file is in an acceptable form for the 'other' side. It is a bit of a laborious process but unless you are extremely unlucky with the software on your 'other' computer you will only have to go through the process once or twice.

The communication program on the other device may be able to handle more complicated translations than the Organiser can, so it might be a good idea to check what you can do from there before trying to do multiple translations on the Organiser.

TTRN Is Processed Before TEOL/TEOF

You must be very careful when using the TTRN parameter because if you get the translated character confused with the TEOL or TEOF characters you could end up in a mess. To try and stop you doing that TTRN and RTRN will not translate the carriage return character (ASCII 13) or the 'sub' character (ASCII 26). Your Organiser reckons that those should be used as all or part of the TEOL and TEOF respectively. But because the TTRN/RTRN parameters are processed by the Organiser before TEOL and TEOF it is possible to translate a character into the TEOL or TEOF character. For example if you had set the TEOL parameter to a 'tab' character and then set TTRN to translate spaces to 'tabs' you could end up sending a file with one word per line. That would not be too drastic, but if you made the same mistake with TEOF you would end up sending a file across the link with only the first word of your original file in it!

REOL/REOF Before RTRN

Crossing the Link the other way the processing order is reversed. The Organiser will look for REOL and REOF

characters first before translating anything, so you should not have so many problems.

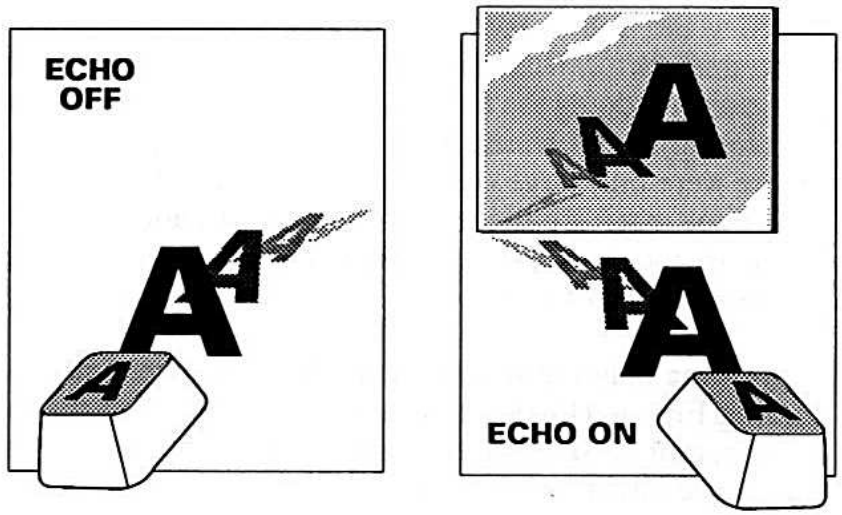
You might have a problem when you are using the OPL procedures Trig$ and Linput$ because the codes for REOL and REOF are treated as part of the data being received and are passed straight through to RTRN for any translation that needs doing. If you are not very careful with the translations and make sure that they don't end up as either REOL or REOF characters you could land up with similar problems as with TTRN.

No Translation Required

If you are using PSION Protocol it has been decided by 'the powers that be' (whoever they are!) that you will not need any character translation done if you are transferring procedure or binary files. Anything that you have put into the Setup Sheet for RTRN and TTRN will be ignored if those types of files are being transferred. It is a different matter if you are transferring a data file, then both RTRN and TTRN will come into action again!

Miscellaneous Parameters

Echo



The Echo Parameter is only used when you are using your Organiser as a terminal. You can either get into the terminal emulator through the TERM option on the main menu or the Term facility from the CAPTURE secondary menu (see the Trial Run chapter).

While working as a Terminal whatever you type on your keyboard will appear on the other computer's display and vice versa. Some 'other' computers will send the characters that you have just sent immediately back to you. If that doesn't happen you can see what you are typing by turning your Echo parameter on. Echo will then send what you are

typing to your display as well as out to the other computer. If you got into the terminal emulator via Capture every character sent or received will also be stored in the capture buffer as well as echoed.

The Echo Parameter has two available options, Host or Local. Host is the default setting which is where you are expecting the 'other' computer to echo what you have sent straight back to you. Local on the other hand means that the Organiser is doing the Echoing back itself. These somewhat confusing names result from the terminology used when there were only large computers which were called 'hosts'. We think that it is easier to remember which does what if we visualise the Host option as the Organiser's Echo off switch, while Local turns Echo on.

If you can only use simplex or half duplex communication setting Echo to Host will not have any effect because the data either, cannot come back the other way anyway or, can only go in one direction at a time.

Things That Can Go Wrong

If you get the Echo setting wrong you could end up with either of two extremes. You could either not be able to see what you are typing on your display or you could end up 'sseeeeiinngg ddoouubbllee'!

If neither machine is echoing you will only see what the other person is sending to you and nothing of what you are sending out. In that case all you need to do is turn Echo on (set it to Local).

If however, you end up seeing double it probably means that both machines are echoing to give you two copies of every

character you type. The remedy for this is equally as simple as above. All you need to do is turn your Organiser's Echo off (set the parameter to Host) and everything should return to normal.

Using Echo as a Testing Facility

If you are trying to communicate, but something is going wrong, with, for example, a printer that will not normally echo what you are sending you can get your Organiser to show you what you are sending by setting Echo to Local.

Many printers now have the facility to print out the ASCII codes that they are receiving from you rather than just echoing the character you are sending. It may be better to use that as well as Echo because then you can see what the printer is actually receiving as well as what the Organiser thinks it is sending out.

The Timeout Parameter

This parameter is used to stop the lines of communication being blocked forever by one of the machines being 'off line' - and hence not in a position to accept the information. If a printer has run out of paper for example, it cannot print any more and can't store any more in its buffer once that is full.

If you set the Timeout to 10 seconds and no data has got through the link in that time, the lines of communication will be broken and you will know that you have got to check for whatever has stopped the communication before trying again. If you are not using handshaking the 'other' machine will not be able to tell the Organiser to stop sending it data even when it cannot deal with any more. Your Organiser

will not know anything is wrong and will carry on sending data which will be lost.

The default setting is None but you can enter any number of seconds between 1 and 255. The '10' seconds in the example above would probably be too short especially if you are sending a file to another computer via a modem. In this case, 60 seconds is probably more appropriate, particularly if you are sending data over the international telephone network.

If you set Timeout to something like '75' seconds, it will stop the Organiser 'locking up' for good if something has gone wrong with the communication link. It will also give you time to give the printer more paper so that it can carry on printing from where it left off. If you have Timeout set to too short a length of time you will not have time to put things right and will have to start sending or receiving the whole file again.

The Width Parameter

Width is used for purely cosmetic purposes - it allows you to specify how long the lines should be. Width will have no effect on data files because it is not used by the Transmit or Receive file transfer commands. If it had been allowed to work on data files it would send the way you organise your database records haywire because they are stored as one record per line.

You will be able to use Width from the terminal and LIST/PRINT from the PROG menu or when you are using OPL Procedures, LPRINT and TRIG$.

Effectively Width will force a REOL or TEOL (depending on which way the data is flowing) into the stream of information crossing the Link when the length of the line exceeds this parameter.

The default setting is None. There is an overall limit of 250 characters on a line. You can set it to any width between 1 and 250.

Width's primary use is to cut lines from whatever width is used in the file to sixteen or twenty characters wide to fit on the Organiser's display. It will change it the other way as well.

You can use Width if your printer does not 'wrap' text from the end of one line to the beginning of the next. You would just set it to the number of characters the printer puts on a line.

Unlike the 'wraparound' feature that many wordprocessors have, Width has no concept of the integrity of words. It will just count the number of characters in the line and split everything beyond the specified limit down onto the next line whether it is at the beginning, end, or half way through a word.

When leaving Setup

After pressing MODE a whole new menu will appear on the screen. This is the Setup menu and it lists a number of things that you can do with the Setup Sheet that you have just finished editing. Exit is the top option. Use this to make the edited version of the Setup Sheet the 'current' one, i.e. the one you want to use now. It is also the way back to the top level Comms menu, but there are lots of other options open to you from here.

All The Things You Can Do With Setup Sheets

There are a set of options that allow you to do things with the sheet that you have just made. Another group allows you to do things with previously Saved Setup Sheets.

Abandon, Edit and Save, as well as Exit which we have talked about already, make up the first group.

Abandon

Abandon will put you back into the top level Comms menu, throwing away any changes to the current sheet that you have

just made. It will leave the Setup Sheet as it was before you edited it this last time.

Edit

Edit will put you back into the Setup Sheet that you have just left. So if you press the MODE key by mistake you can return to the same Sheet and carry on editing.

Save

Save allows you to save the Setup Sheet that you have just created whether it is the current Sheet or not. In practise it is unlikely to be the current sheet because you have to Exit the Setup menu to make it the current Sheet and then go back into it again. You can always decide to Save a sheet later after it has been the current one for some time.

You only need to Save Setup Sheets if you want your Organiser to communicate with several different things like a printer and a computer which require different parameters. The ability to Save Sheets means that you only need to edit the Setup parameters once for each piece of equipment you want to communicate with.

To Save a Setup Sheet you follow exactly the same rules as you do for saving any other type of file. You select the memory you want to use and then type in a filename of no more than eight characters starting with a letter.

It will be much easier to find saved Sheets and to recognise them in the general directory if you establish a system for giving them meaningful names. For example Setup1, Setup2, or you could use Setuppr, Setuppc and Setupmo.

You will then know instantly what sort of files they are and what each one contains.

Working with more than One Setup Sheet

Once you have got more than one sheet saved then another group of options come into their own.

Load

Load allows you to load a previously Saved sheet. The parameter values will not be turned into the current Setup Sheet until you Exit the Setup menu. You could just load a sheet in to check what is there, then select Abandon and you will not have changed your current sheet at all.

Once you have selected Load all you need do is choose the right memory (using MODE) and type in the name of the file you want.

Erase

Erase enables you to erase a Setup Sheet that you had previously saved but no longer need. If you have got files you don't want any more you should erase them anyway to conserve memory. You Erase a Setup Sheet like any other file. If you are not sure of the name, especially if you have a CM model, you may need to check it with the Dir option before selecting Erase.

Dir

Dir will give you a directory listing of all your saved Setup Sheets.

Reset

Reset is the final option on the Setup menu. It will change all the parameter settings back to their default values.

Note that Save, Load, Erase and Reset do not change the current parameters being used by the Comms Link. Any changes you have made to the Setup Sheet will only take effect if you Exit the Setup menu. Once you have returned to the top level Comms menu via Exit, the changes will be accepted as the 'current' Setup Sheet and will be used by any file transfer and by the Terminal facility.

CHAPTER 4

Trial Run

From the top level menu you have a choice in the way you are going to try and send or receive the information. You can either transfer data across the link with the Terminal emulator which transfers each character as you type it, or you can use the file transfer commands. For the initial test we will use the terminal emulator just to test the water by putting our big toe into it! If that works we will then take the plunge by trying to transfer a whole file. If however you are an 'all or nothing' person you may want to miss the Terminal test out - especially as it may mean changing some of

the parameter settings that you have only just sorted out - and go straight to the file transfer sections. If you do that and nothing gets through the link you will then have to come back to the Terminal test as the first port of call in the search for a diagnosis of what could be wrong. Of course, if you only want to use the terminal you are in the right place anyway.

Terminal

To start with it would be easier to connect the Organiser and Comms Link to a computer close at hand, regardless of whether the final configuration is two modems and a computer two hundred miles away. You will be able to see and be in control of both ends of the line.

While using your Organiser as a Terminal some of the parameter settings in the Setup Sheet may need to change. Check the manual of the communication software that the 'other' computer uses to see what parameter settings are needed.

Parameters You May Need To Change

The Baud rate must match that of the other machine as must Hand TEOL and REOL. TEOL may need to be changed from the default <CR> <LF>, to just <CR>.

The rest of the parameters are not essential, but changing Echo to Local will allow you to see what you are typing on the Organiser's display as well as on the other computer's screen. Setting Width to the number of characters you can see at any one time on the display will stop each line scrolling horizontally if you send a longer line.

Once you have altered the Setup Sheet press MODE and then select Exit. You should then be back in the top level Comms menu and from there all you need to do is select TERM.

The Organiser is now in Terminal mode. You need to run some software to put the other computer into Terminal mode as well before any communication can take place. Try to select a simple terminal emulator, either TTY, ANSI, or BBS. One of these should be available on your terminal software. The more complicated terminal emulators can send streams of control characters which would confuse the situation.

Once both computers are set up and ready to talk to one another, any character that you press on the Organiser should appear on the other's screen. Whatever you type on the other computer will appear on the Organiser's display. Key presses will only be shown on both computers if you have set the Echo parameter to Local.

If you can see what you are typing on the other computer and vice versa your Link has passed the first hurdle. You can now try the next one, file transfer!

If Nothing got Through on the Terminal Test

Start again! It is often the little things that we all overlook, like whether both machines are turned on! But once you have gone through everything that we have talked about (including the information in this chapter and all the preceding chapter's 'Delving Deeper' sections) and you are absolutely sure that you have got everything right so far, it may be something more fundamental, like the Comms Link cable not being wired to the pins that the other machine needs.

The next chapter will help you sort out what is wrong and put it right.

File Transfer

Preparation

First you need to prepare a short file to send. When you are using the file transfer commands you can only send Saved files. Check the file name if you are unsure of the exact name. To do that:-

> ON a CM, have a look in the DIRectory option on your Organiser's main menu. Then return to the Comms top level menu.
>
> On an LZ, enter the appropriate option (Prog for OPL files, Notes for a notepad etc.), select Dir and press the DOWN ARROW key. Alternatively you could use the Dir option in Utils.

Once you are sure of the filename you are ready to start.

Starting to Communicate

If protocol is set to 'None' or Xmodem, the receiver must be set up first. If you start the sender first it will have already started sending the file before the other end is ready to start receiving. You will then lose the beginning of the file.

If you are using Psion it doesn't matter which end you start first. Psion protocol allows the Organiser to control the PC or Mac at the other end anyway, so as long as it is turned on

and running CL (the Comms Link software) there is no problem.
Get both machines ready, i.e. type in the relevant commands followed by EXEs or ENTERs to action them, apart from the final EXE which would start them sending or receiving. Whichever one you start first will be trying to get through straight away and you will only have a limited amount of time before it will give up and you will have to start it all over again.

Once they are both ready, barring the last EXE or ENTER, you should start the receiver first.

Specifying the Filename

You will be prompted for the name of the file you want to send, whichever direction it is going, but depending on what protocol you are using you may also have to specify the filename at the other end as well.

Transferring The Filename

When Protocol is set to 'None' or Xmodem the Organiser does not transfer the filename with the file. When you set the receiver up you need to type in the name of the file you want the file called on the other computer, this doesn't have to be the same name that you used on the Organiser.

If you are using Psion Protocol the name of the file will be automatically transferred with the file although you will be given the opportunity to change the name if you want to.

If you are sending a file to the Organiser you must add an extension to the filename to enable the Organiser to distinguish what sort of file it is. See section on Filename

Extensions in the 'Delving Deeper' section of this chapter for more details.
If your Organiser can't find the file you have asked to be sent, it will put 'FILE NOT FOUND' on the display.

If the Organiser is receiving a file it cannot accept two files of the same name in the same memory. So if you ask it to receive a file that already exists the transmission will not go ahead. You will be given details of the options available in the 'Delving Deeper' section below but for the trial run just choose a file that you have not already got, or change the name of the file that you are sending, to avoid any problems.

Final Preparations

When you select Transmit you will be offered a choice of 'File' or 'Procedure' (and 'Notes' on the LZ). If you want to transfer a database file - either Main or a file you have created - or a Diary file on the LZ, 'File' is the one you want. Procedure and Notes as the names suggest will transfer OPL procedures and LZ Notepads respectively.

You will then need to select the memory on your Organiser where the file is, or where you want it to go, with the MODE key. The Organiser may give you a filename that it thinks you might want. The Default filename will either be the last file that you transferred or Main if you have not transferred a file before. If it has not got the filename you want press ON to clear the default name and type in the name of the file you want to send or receive. The default filename for an incoming file is also Main so be careful because if the file already exists and you select the 'Erase' option you will delete the main data file on your Organiser.

It may be necessary to type the appropriate command either on the 'other' computer or on the Organiser to tell the other machine that you are ready to start but that will depend on what you are communicating with and what protocol you are using.

How Do You Know The File is being Sent?

If Protocol has been set to 'None' you will see what you are sending/receiving on the Organiser's display, although you may not be able to read it because it will probably be going too fast. When the transfer is complete the display will prompt you for another filename to send or receive, but if you don't want to transfer any more files you can get back either to the top level COMMS menu or the Organiser's main menu with the ON key.

If you are using Psion or Xmodem protocol the data is divided up into blocks, each individual block is checked and the number of each block is displayed on the Organiser's screen. If the information is flowing across the link successfully you will gradually see the number of blocks sent or received increasing. When it has finished your Organiser will display 'SENT OK' or 'RECEIVED OK' and you will know that it has worked!

Not working?

You didn't really expect it work first time did you? If it did work first time you are an extremely lucky person. If it didn't, welcome to the wonderful world of computers!

If absolutely nothing got through and your link has not already passed the Terminal test go back and try that.

If The Transfer is Unsuccessful?

If the transfer has not been successful different protocols will give you different error messages.

If you are not using a protocol and nothing has got across you will get little clue as to what has gone wrong. You may see the file flashing across the sender's display but nothing on the receiver's. You may get an error message or you may just be informed that the receiver has given up trying.

No Protocol

While you are not using a protocol, if the file that arrives doesn't look quite the same as what was sent, it is more than likely something not quite right on the Setup Sheet.

For example:-

Sympton The first few characters are correct but then they degenerate into rubbish

Cause There is probably a framing error

Action Check what you have set the Bits and Stop parameters to change those until all the characters are correct.

or,

Symptom You lose either, a lump off the end or lumps in the middle and the end.

Cause The handshaking is wrongly set.

Action You need to alter the Hand parameter. Check what handshaking the other machine(s) can use and try setting the Organiser to a different one

but please note that

If you get the above symptom when you are trying to communicate with a multi-user multi tasking system it may be that handshaking is working, but that the other system is doing another task when the Organiser sends a stop signal and doesn't see it until two or more seconds later. When it does come back to you and stops sending your Organiser data you will have lost a chunk of the file it was sending.

Action You need to just use XON/XOFF handshaking. See XON/XOFF section in previous chapter

or

Symptom You are printing and half the characters are rubbish.

Cause Incorrect frame size.

Action You may have got the Bits or Parity settings wrong. Double check.

Psion and Xmodem

Psion and Xmodem protocol will tell you what it is doing as it does it. It will display the block numbers as it transmits them.

SENDING BLOCK 1.

When there is a problem i.e. a block either didn't get through or didn't get through correctly it will display

ERROR BLOCK 2

and then try and send it again. You don't have any idea what problem caused the error. One or two isolated errors are not necessarily a problem that you have to sort out, they could be caused by a surge on the mains, but if you get persistent errors it might be worth checking the Setup settings to make sure they match.

If the Transfer is Successful

Once you have achieved transfer with one 'other' machine you will think that you know what to do with all the other 'others' that you want your Organiser to communicate with. Don't be surprised if they pose different problems because no two things are the same.

Delving Deeper

The Terminal

Keyboard

The keyboard on the Organiser has been extended in the Terminal. You send the full range of normal characters using the SHIFT, CAP and NUMber keys with other keys. Since the Terminal can only transfer characters across the link and has no editing capability the cursor keys are redundant. But they haven't been left with nothing to do -

The UP/DOWN Arrow Keys

These will stop and start the display. If the other machine is sending a large amount of information to your display and it is coming in too fast for you to read, you can stop it with the DOWN Arrow, read that display full and then press UP Arrow to start the data pouring in again. One thing to remember about selecting TERM from the top level COMMS menu is that once the characters have passed through the display they are gone. If you want to keep them you should go into the CAPTURE option from the COMMS menu and then select TERM from there.

The LEFT/RIGHT Arrow Keys

These become control keys which give you access to yet another layer of characters. You use them like SHIFT in conjunction with a letter to produce different codes. The LEFT Arrow becomes like the control key on an IBM PC.

so that Left Arrow plus 'A' through the alphabet to 'Z' produce the same effect as the PC control codes, Control A to Control Z. These together with RIGHT Arrow 'A' to 'E' gives you access to the thirty two unprintable ASCII control characters. We have come across a few of these already like DC1/DC3 (XON/XOFF), <LF> and <CR> for example.

RIGHT Arrow 'F' to RIGHT Arrow 'W' gives you sixteen additional printable characters that the Organiser cannot normally handle, such as ! and the commercial @ sign.

ASCII Unprintable Characters

LEFT ARROW plus	ASCII CHAR	FUNCTION
A	<SOH>	
B	<STX>	
C	<ETX>	
D	<EOT>	
E	<ENQ>	
F	<ACK>	
G	<BEL>	
H	<BS>	
I	<HT>	Horizontal TAB
J	<LF>	Line Feed
K	<VT>	
L	<FF>	Form Feed
M	<CR>	Carriage Return
N	<SO>	
O	<SI>	
P	<DLE>	
Q	<DC1>	XON
R	<DC2>	

S	<DC3>	XOFF
T	<DC4>	
U	<NAK>	
V	<SYN>	
W	<ETB>	
X	<CAN>	
Y	<EM>	
Z	<SUB>	End of File

RIGHT ARROW plus

A	<ESC>
B	<FS>
C	<GS>
D	<RS>
E	<US>

Additional Characters

RIGHT ARROW Plus

F	!
G	#
H	&
I	'
J	?
K	@
L	[
M	\
N	]
O	^
P	_
Q	`
R	{
S	|

T	}
U	£
V	DEL
W	?
X	?
Y	?
Z	?

The only editing facility you have here is the DEL key. So the presentation of what you are sending depends on the accuracy of your typing! If your typing is not very accurate and fast, you can use the Capture facility as a way of preparing information before you send it.

Scroll Delay

To allow you time to read what you are receiving the normal scroll (both vertical and horizontal) will be delayed while you are in the Terminal. There is no easy way to control the vertical scroll, but you can stop the horizontal scroll by changing the Width parameter to the width of your display. However, if you enter Terminal in Capture the scroll delays are turned off anyway.

Power Conservation

Normally the Organiser only puts power on the RS-232 port, and incidentally the Ram/Datapak connectors too, when data is actually being transferred. Sending data down the Comms Link is every bit as costly in power as Saving to a Ram/Datapak, so if you are running your Organiser off batteries you need to be using the Comms Link for as short a time as possible. This is particularly true when using the Terminal because the RS-232 port needs power the whole time you are communicating. Here again Capture will help

as you can prepare what you want to send 'off line' and only take the extra power when you are actually sending the data.

Leaving The Terminal

You leave the Terminal emulator and return to the top level COMMS menu by pressing the ON key.

If you need to leave the Terminal in the middle of transmitting or receiving data there is no problem providing you have your Hand parameter set correctly. The 'other' end will just be put on hold until you go back to it. For example, to send a message that you have already prepared to your electronic mailbox, you will first need to dial the number and 'log on' (give your User identity and the correct password etc. to the service's controlling computer). You would then need to leave TERM and get into the Transmit Option in the CAPTURE menu to send the message. You could do the whole operation from the CAPTURE menu, but then you would be more likely to run out of memory because everything that you type in and receive back in the 'logging on' and signing off processes will be stored in memory as well as the actual data.

Capture

Capture has a Line Editor (called Edit on the CM) which allows you to prepare messages before you open the communication lines. It also allows you to store incoming messages so that you can read them at your leisure. Using Capture therefore allows you to remove the time pressures that are inherent when you are 'on line' and communicating by modem over the telephone network. It also allows you to

cut down on power consumption because while you are not transmitting the RS-232 port can remain closed. If you are running your Organiser off batteries, you need to be 'on line' for as short a time as possible.

To capture data enter the Terminal (outside CAPTURE to conserve memory) and then get the 'other end' to the point where it is ready to send you the information you want. Leave TERM with the ON key and after selecting CAPTURE, the handshaking will take the other machine off hold and it will then send the data. If you have plenty of memory space you could do the entire operation from CAPTURE.

Once you have got some data in the capture buffer it is on the internal Ram of your Organiser and it doesn't matter what you do with the Comms Link. You could even unplug it if you want to. The data is now subject to the same constraints as everything else in the internal memory, i.e. if you run out of battery power you lose the lot! Other than that, data held in the Capture buffer will stay there indefinitely until you clear it. The Clear option will erase the whole buffer so it probably isn't a good idea to store more than one piece of information in the buffer at any one time.

Options From The CAPTURE Menu

Term

This puts you into the terminal emulator with capture turned on. Everything that you type and receive from the other end is stored in memory. You will therefore need a great deal of free memory, particularly if you are using a high Baud rate. For example, 9600 Baud will consume approximately 1K per second if the 'other' computer is sending at full speed.

The capture buffer will take up all your free memory. So if you wish to save the capture buffer to a file you need double the memory space make sure that you leave enough space to save it into.

If You Run Out Of Memory In Capture

If you get an 'OUT OF MEMORY' message while receiving data pressing SPACE will take you back to the CAPTURE menu. You then need to make more room in memory. To create space you can use the Save and/or Clear options from the CAPTURE menu. Use Save to save what you have captured into a Ram/Datapak. Then Clear the Capture buffer. Or you could return to the main menu and erase some other files.

Editing

On the CM there's an Edit option on the Capture Menu. The LZ puts you into the line editor as soon as you enter the CAPTURE menu.

The Line Editor Keyboard

The line editor here uses similar keys to the one on the PROG or NOTES main menu options.

The UP/DOWN Arrows act like normal cursor keys and move you up or down the Capture buffer.

The LEFT/RIGHT Arrows act like normal cursor keys and move the cursor within the line.

DEL deletes character(s) to the left of the cursor.

SHIFT DEL deletes character(s) to the right of the cursor.

EXE splits the current line at the cursor's position.

MODE takes you to the Capture menu. It will also put you back into the Capture editor from the capture menu on the LZ.

You can insert character(s) at the cursor position.

FIND

Will find a search clue from data held in the buffer in the normal way.

To use it:-

Type in the clue - you can edit it by moving around with LEFT/RIGHT Arrows

Press EXE until it finds the right place.

Once you have found what you are looking for, pressing ON will allow you to edit the data that is stored in the capture buffer.

Save - all models

Save makes a copy of whatever is stored in the capture buffer. The data will remain in the buffer even after you have saved it. So when you are saving it to the internal Ram you must have the same amount of memory free as the buffer itself occupies. To find out how much data you have captured, check how much free memory you have before and after capturing by using the FREE command in the calculator.

You can Save data that you have captured to a data file, a procedure file or a notes file on the LZ. You select which type of file you want it to be, then which memory you want it to be stored in, making sure there is enough space to take the whole file otherwise you will get an OUT OF MEMORY message and the Save will be aborted.

Transmit - all models

This works exactly the same as the main file transfer command except that you don't have to select a file type for the file you are sending.

Just select Transmit and type in the filename you want it to be called on the other end either on the Organiser if you are using Psion Protocol, or on the other computer if you are using Xmodem or No Protocol. See section on Files at the end of the chapter for more details about these options. Pressing EXE will then start sending the file.

Your Organiser will use the settings for Protocol, TEOL, TEOF and TTRN in the current Setup Sheet.

Home

LZ only - will move the cursor to the beginning of the data in the buffer.

End

LZ only - will move the cursor to the end of the data in the buffer.

Clear

On all models - erases everything in the capture buffer.

All you Need to know about Files

Transferring Different Types Of Files

If you are using no protocol, i.e., the Protocol Parameter set to 'None' you can only transfer text (ASCII) files with the file transfer commands. Text files are:-

Database files - on all models
OPL Procedure (text) - on all models
LZ Diary files
LZ Notepad files (text)

Like OPL Procedures there are two types of Notepad file. The text file will be just what you type into the Notepad. But if you have done any calculations or used any OPL functions these will be stored in a separate binary (non text) file which you will not be able to transfer without a protocol. There is a problem in establishing which type of Notepad file you have because the only difference is the extension of a dot and three letters that is added to the filename internally by your Organiser. Notepad text filenames have '.NTS' added to them while binary Notepads files have '.OB7' added to the end of their names.

Non text (binary) files will need an OPL procedure or Xmodem protocol - Psion Protocol if you have an IBM

PC/AT or Apple Mac - to aid their transfer. Binary files are:-

LZ Notepad files (binary)
CM/XP Diary files
OPL Object procedure files
Comms Link Setup files
Spreadsheet files
Pager files

Specification for Filenames

Filenames can be up to eight characters long (letters or numbers) but the first character must be a letter. Do not include spaces or punctuation marks in filenames.

Procedure Names

Procedure filenames are normally included as the first line of the procedure with a colon after it (which isn't part of the filename). If there is a discrepancy between the filename and the top line, say because you wrote the procedure on another computer, the filename will determine the top line and not the other way round.

Filename Extensions

Your Organiser adds extensions or suffixes automatically to the filename. The extension added will vary according to the type of file. The only time you will have to worry about them is when you are sending files to your Organiser.

If the file hasn't been on your Organiser before you just need to type in the appropriate extension after the filename with a full stop between the two. For example, the filename for

an OPL procedure called VAT would be 'VAT.OPL' (the Organiser is equally at home using either case for filenames). Or, for a diary file called 'd900415' you would type in 'd900415.odb'.

List of Extensions Used

Type of file	Extension
Data files/LZ Diary files	ODB
Procedure files (text)	OPL
CM Diary files	OB2
Procedure files	OB3
Comms Link Setup Sheets	OB4
Spreadsheet files	OB5
Pager Setup files	OB6
LZ Notepad files	OB7
unused file types	OB8 OB9 OBA OBB OBC OBD OBE OBF
LZ Notepad files (text)	NTS

If the file you are sending is one that originated on the Organiser but you have sent it to another computer, the suffix on the filename may have got changed. If you get the wrong extension on the filename your Organiser will not recognise the file that it has just received. To avoid problems the best idea is to put the extension on the filename anyway whenever you send a file to your Organiser, then you are covered for all eventualities.

If The File Already Exists

If there is a file with the same name as the one that you are trying to send on your Organiser already you will not be allowed to transfer it. The options available to you vary slightly depending on what type of file it is. If it is a Procedure or LZ Notepad you are only given the choice of Deleting or not

Deleting. For Data files on all models and LZ Diary files you will be given the opportunity to Append as well as Erase the existing file.

To Delete Or Not To Delete

With most file types you will be given the option to 'Delete' or 'Erase' the existing file so that you can replace it with the one you are sending. We are always nervous about deleting files that we can't see because, unless you have a backup copy, it is so utterly final. With the Delete option you will be asked to confirm that that is really what you want to do.

If you do not elect to delete the existing file you can press 'N' which will return you to the 'Receive (your filename)' prompt where you can change the name to avoid the clash. The change doesn't have to be great, one character will be enough.

If you make a point of keeping your filenames short you will then have room to add version identifiers onto the end. Either '1,2,3.' or 'a.b.c' etc. will be adequate to create separate files. You will need enough spare memory to store the file twice while you check the earlier file before Erasing it.

If you are given and then select the Erase option, you will not be given the opportunity to change your mind or to change the filename so be very careful.

Appending

The Append option allows you to add the file you want to send back to your Organiser onto the end of an existing file.

Append is only available with Receive although there are OPL procedures available (XFOPEN in the newer Comms Link Manual) which allow you to Transmit a file and then append it to a file that already exists on the other computer as long as its software can handle it.

If The File Exists on the Other Computer

Without a relevant OPL procedure you are not given any warning that the file of the same name already exists on the other computer. The existing file will be automatically deleted and the file you are sending put in its place, so be careful.

CHAPTER 5

Nothing Getting Through All or Some Of The Time?

In this chapter we assume that, by the time you have got to this point in the book, you have read and carried out everything we have already described. Unless you already know about what has gone before, jumping in here will not solve all your problems.... You need to have got everything aligned on the two machines first.

If you are one of the many who have plodded through and done everything to the letter and still nothing works - bad luck! Don't despair, we aren't going to leave you high and dry, well not yet anyway!

If you have got this far with no success, or you are having intermittent problems getting data across the link, it is possible that you have got a wiring problem. Either some or all the component parts of the physical link maybe incorrect, or it is possible that you have a faulty cable or connector (plug). You may even have a problem with the machines themselves so the testing strategy below will give you the ability to test everything to establish the cause of the problem.

There are several ways of testing a communication link. Many people get by using a 'poke and see' philosophy but we don't recommend that way. There are just so many configurations that you have to be either very lucky or very dedicated to solve wiring problems by trusting in the laws of probability. We are going to suggest that you enlist the help of a BOB.

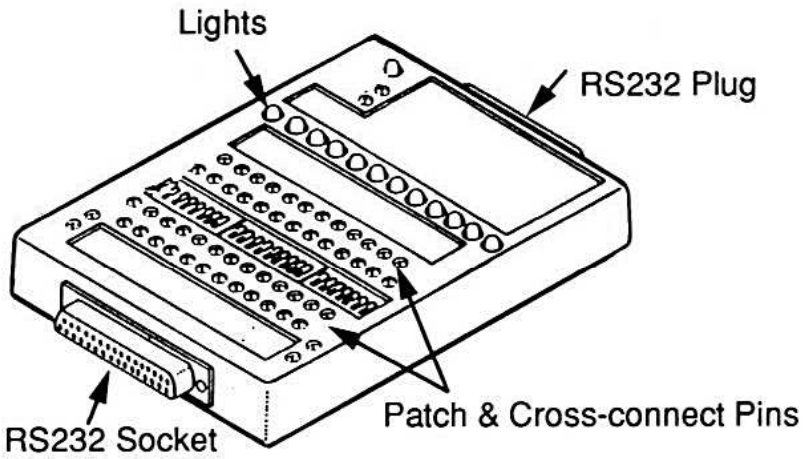
BOBs themselves are expensive, but there are much cheaper equivalents on the market. BOBs and their equivalents are readily available and are ideal for giving you that professional image that really will impress your friends! They will also give you access to what is happening inside your link and

the means to correct anything that is wrong without you having to fork out for an electronic engineer's tool box.

What Is A Break Out Box (BOB)

A Break Out Box - which we always call a BOB - is a box with a 25-way RS-232 serial plug and socket at either end. On top, it will have a row of switches down the middle with one or two lines of holes on either side of them. It will also have a row of dual colour LEDs (Light Emitting Diodes or little bulbs) which correspond to some or all of the holes.

Although it is possible to get BOBs with single colour LEDs these will not allow you to see which of the three possible states the wire is in. For example if the light is off does it mean that there is no signal on the wire at all or does it mean it has a negative signal going through it? For the data transmission lines you can solve the problem by sending data down them which will change the state anyway but the Handshaking wires are more difficult to test. We therefore recommend that you use a dual colour BOB or one with two complete sets of LEDs.



There will be some patch wires supplied with the BOB and you may also get an RS232 extension cable with 25 way serial plugs/sockets on either end.

Alternatives to BOBs

A fully fledged BOB is expensive, between £80 and £100 or more, but if you combine two units to give you the same facilities as a BOB you can cut the price to somewhere between £20 and £30 for a Mini Tester (RS-232) and something called a Wiring Box.

The Mini Tester will have the lights, whilst the Wiring Box will enable you to swap wires over without using a soldering iron. If you get the two units all you need to do is plug them together and then plug the Comms link in one end and whatever you want your Organiser to communicate with straight into the other end of the two boxes.

What Does a BOB Do?

Either a BOB or the cheaper equivalent will enable you not only to test your link and get the wiring right, but you can also use your link with the BOB in the middle until you can get a proper cable made up.

A BOB will allow you to see the data going across and whether the handshaking lines are on or off. When the BOB detects a signal on any wire it will change the colour of the LED from whatever colour it uses for a negative signal to the other colour it has available.

If data is not getting through the BOB will allow you to change the connection between your Organiser and the other device by putting some temporary wires to join them

together. These temporary wires are normally called Patches or Jumpers.

Your Testing Strategy

Step 1

1 A

Check that the Organiser and Comms Link are working.

Put your Organiser, with Comms Link plugged in, into Term - turn to previous chapter for parameter settings but deselect XON/XOFF handshaking - then plug your BOB or equivalent into the other end of the Comms Link. Make sure that all the little switches are ON so that it connects both ends of the BOB together and check that the LED lights are on.

No lights? - move directly below.
Flickering lights? - move to 1 C.
Lights on? - move to 1 B.

If you get no lights at all check:-

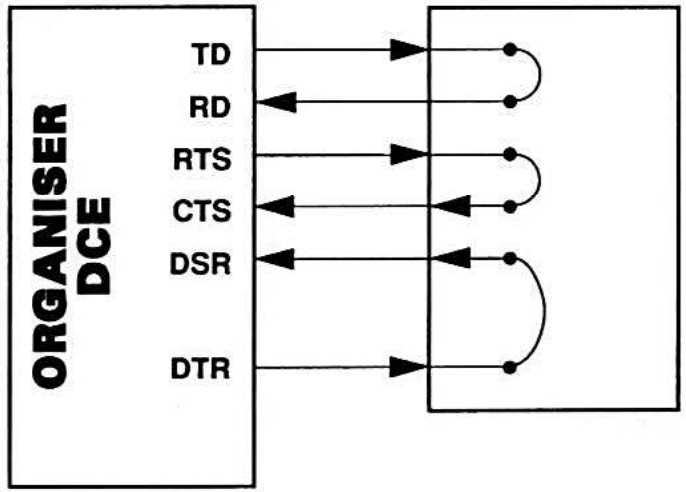
The Organiser is turned on with COMMS selected.

If you still have no lights on move to 1 D.

1 B

Switch all the switches off on the BOB and plug in the patch wires so that the 'pairs' of wires are swapped with each other.

That is, Pin 2 goes to pin 3, 4 goes to 5 and 6 goes to 20 on the same sides. This configuration is called 'loop-back' and allows you to test just the Comms Link without worrying about the other end of the line.



Whatever you type on the Organiser's keyboard now will be sent back to its display if the Organiser and Comms Link are working correctly. If the Echo parameter in the Setup Sheet is set to Local you should be 'sseeeeiinngg ddoouubbllee.'

Each time you press a key, you may see the LED connected to pin 2/3 flicker - you are seeing the data being sent. If you feel like seeing this in slow motion, try reducing the Baud rate to 50 in the Setup Sheet. Don't forget to reset it to the correct value afterwards.

If nothing gets through move to 1 D

1 C

If some lights are on but they flicker on and off when you move the wire, especially if you started with an intermittent problem, you probably have got a faulty connection in the Comms Link's cable or plugs.

1 D

Return the Comms Link to where you brought it and get it checked or exchanged.

Step 2

2 A Testing compatibility

If three or four of the lights come on you know the Comms Link is probably working. Before unplugging the BOB from the Comms Link draw a quick diagram so that you can remember which lights came on. Note the colour of the lights because that may be useful later on.

2 B

Carry out the same thing on the 'other end' by plugging the opposite side of the BOB into that - if it works on the parallel standard you will need to plug the BOB into the parallel to Serial converter - and again note down which lights are on.

If no lights come on check that you have:-

Turned the 'other' machine on and that it is ready to communicate with your Organiser. For example, if it is a printer it must be 'on line'.
and

Run the correct software, if relevant, and check that it is properly set up.

and

If your 'Other' machine has more than one serial port check that you have plugged the BOB or equivalent into the port that the software is working with.

If these all seem correct and you still have no lights on:-

Your machine may not be functioning properly so send it back to the retailer or manufacturer (or get them to come to you) to get it checked over.

Step 3

Compare the two diagrams you have drawn of the lights at each end of the link.

If there are no overlaps i.e. a totally different set of lights come on for each side move to step 5.

If there is an overlap, i.e. one or more of the same lights are coming on on both sides, move to step 4.

Step 4

4A If one or more lights are overlapping it is because both pieces of equipment have been wired as DCEs and you will have to swap some wires round. To do that:-

Plug in the BOB or Mini Tester and wiring box into the Comms link.

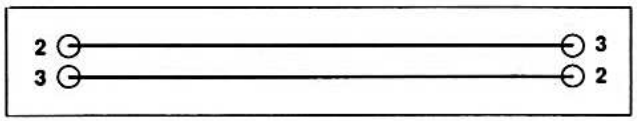
4B

If you are using a BOB turn all the switches OFF except pin 7.

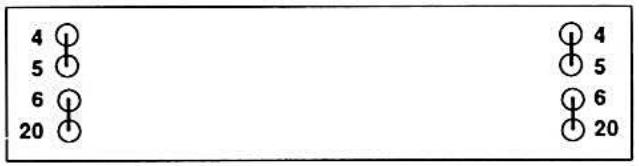
4C

If you have a clash on Pins 2 or 3, put in a couple of patch wires to swap the data lines. You connect pin 2 on the Organiser side to pin 3 in the other. You can 'loop-back' the handshaking lines (so they do not work) to start with.

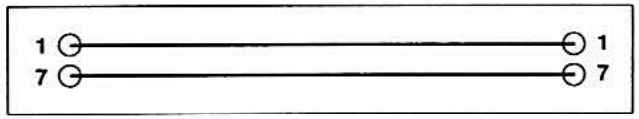
To do that, use the wire links to link the following Pin numbers on the BOB or Wiring box.



The rest of the pairs are connected so that one goes back to the other one in the pair on the same side. Wire pins 4 and 5, and 6 and 20 together on both sides.



Pins 1 and 7 carry on going straight through -



Pin 1 and Pin 7 on each end of the link are always connected to their matching pins at the other end, no matter what other inter-linkings are made.

Connect the 'other side' to the BOB or equivalent and type some characters on the Organiser and see if you can send data across the link.

4D

If you have a clash on Pins 4 or 5, swap over these two links so that Pin 4 goes to 5 and Pin 5 goes to Pin 4.

.4 - 5
.5 - 4

4E

If you have a clash on Pins 6 or 20, swap over these two links so that Pin 6 goes to 20 and Pin 20 goes to Pin 6.

.6 - 20
20 - 6

Pin 8 is not connected on the Organiser side of the link but if it is used on the other side it can be linked into pin 6 on the same side.

4F

Again try sending characters down the link and, if successful, try transferring a file. If not, move to Step 5.

Step 5

If data is still not going through:-

Disconnect the handshaking wires (if they aren't already) by pulling out the wires that go to pins 4/5 and 6/20.

Go into the Setup Sheet, and set handshaking to None and Baud to the slowest speed that both machines can tolerate.

Don't forget to change Handshaking and Baud rate on the other machine too, after you've made a note of the current settings.

Try sending some more characters. They should get through this time. If not, it's worth checking the Setups at both ends of the links to make quite sure that you haven't made a silly and easily rectifiable mistake. We are all humans, after all!

If the link now works, you need to find out whether it was a problem with the Baud or the Hand setting (or both) that stopped data getting through.

First you need to test the Baud settings by increasing the speed on both sides in steps. Go up one level at a time until you get to a rate where data gets corrupted or nothing gets transferred. You now know the fastest speed that the link between these two particular pieces of equipment will work at and you can set it back to that.

Now you need to look at the Hand setting. It depends what handshaking systems the other machine can use and whether you can use them singly. If you have to use all or nothing and the link doesn't work when they are selected you will have to settle for nothing! But if you can select them individually, select one (RTS/CTS for example), and plug the patch wire into the relevant holes on the BOB or Wiring Box (pin 4 on the Organiser across to either pin 4 or 5 depending on whether the other machine is a DTE or a DCE).

You can then try the link again, but you will not know whether the handshaking system works until you get the 'other' machine to say it doesn't want any more data. You can get it to do that by sending enough data to fill up the buffer (which might take a long time!) or, if it is a computer you could use the PAUSE command which most terminal programs have. You can make a Printer go 'off line' by flicking a switch, or as a last resort by taking the paper out. If your Organiser stops sending more characters (possibly after

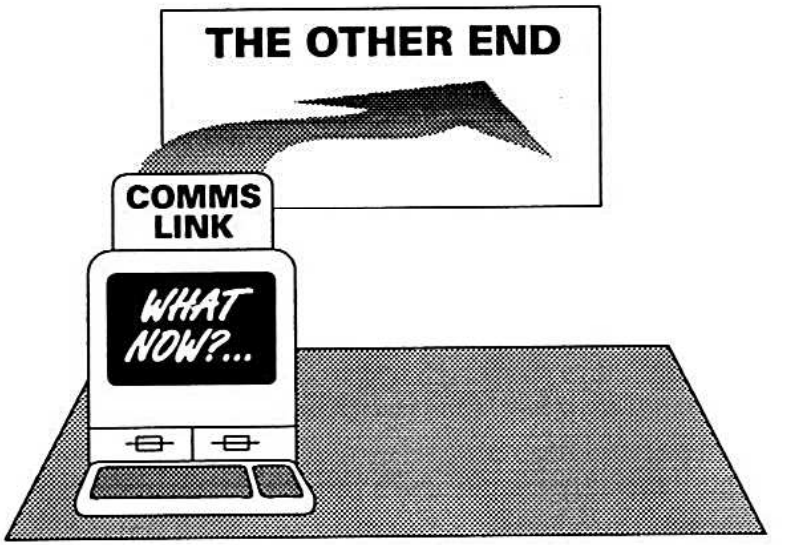
a short delay) you know that the handshaking is working. If the Organiser carries on sending data you know that it doesn't work.

If that handshaking works you needn't look any further because you only need one. If it doesn't work, you could try another one.

At the end of all these tests you will know both what speed and what handshaking this particular link likes to use. Don't forget to note down what links and settings you used, and Save the current Setup Sheet, so that you'll know what to do to restore this link even if you need to communicate with other devices.

CHAPTER 6

The Other End



This chapter will concentrate on modems and printers. You should use it in conjunction with the relevant manuals because we will only supplement those, helping to illustrate some key points that you will need to know when using particular machines with your Organiser.

Setting up Printers and Modems

You will need to set up the communication parameters, or at least check what parameter settings the printer/modem is using, before sending characters and/or files across the communication link to your Organiser.

Printers and modems have two ways of storing their controlling parameters. They either have a bank of DIP switches with the position of each switch (or group of switches) determining the setting for one parameter, or they have battery backed-up memory where the settings are stored.

To change the setting on the DIP switches is easy because you just change the position of the switch. To change parameter settings that are stored in memory you will probably have two options. For setting up the communication parameters you may have to use a combination of the front-panel switches. One switch may determine the type of parameter and then another switch may be used to select the setting. Printers usually use a printout or a small display to guide you through the Setup process.

You may also be able to send codes over the communication link that will change the settings from the Organiser in Terminal mode. Our printer uses a combination of switches to set the communication parameters up to start with, but codes sent from the Organiser can change the parameters as and when you need to. Our modem on the other hand uses codes all the time, both at the initial set up stage and later.

An initial Setup with Switches

For example, on our printer if you turn it on pressing the Select button at the same time, it will print the current memory settings. To accept them you just press the Feed button, but to change them you press the Print Style button. It then prints the current setting of each parameter, one at a time. You press the Feed button to leave the parameter as it is and move onto the next one. If you press the Print Style button you select a new setting. Pressing the button again moves on to the next value, so that if you want to select 2400 Baud and it is the fifth one on the list, you press the Print Style button five times. Finally the Quiet button will exit the parameter setup sequence at any time and put the printer back into 'on line' status ready to start printing.

One thing to remember is that a printer or modem may not retain the factory settings. Once you have changed the settings they become the default values for those parameters and whenever you turn the printer on, your settings will reappear, not the ones the factory set. If that happens, the only time you will see the factory settings again is if you clear the values that you have set by doing a 'hard reset' by physically pressing a reset button which is normally located deep in the innards of the printer. Having said that however, our modem does retain the factory settings as its default values.

Printer Parameters

You will find that your printer has a great many more parameters than your Organiser does. The majority of the Organiser's parameters come under the communication or interface heading while your printer will have printing and page layout parameter sections too.

The printing style parameters determine options like language set, character font and character size. The layout parameters include page length (pages are nearly always referred to as 'forms' in printer jargon), number of lines per inch and whether automatic line feed and/or carriage return is enabled.

As long as you match up the parameters that appear on your Organiser's Setup Sheet you can set the others to suit whatever effect you want to achieve. Check that you have covered all the parameters on the Setup Sheet because they do not all come under the communication parameter heading. It's quite likely that TEOL and TEOF may come under the page layout section.

Setting up Hayes Modems

If you are using a Hayes compatible modem, most of the communication parameters will be set up for you. The modem will look at the first 'AT' it receives (the wake up command) to see what Baud, Bits and Parity settings the sender is using. It will then automatically set its own settings to the same ones. There will be a number of other commands on your modem; some of these will need to be set before you can start transmitting data to it.

Hayes Setup Parameters

There are three groups of parameters or commands that are common to all the pieces of equipment on a communication link which includes a modem.

The first group will be common to everything and will need to be matched accordingly. Some parameters on your Organiser's Setup Sheet, like handshaking, will need to be

matched both with the parameters on the computer at the other end and the modem. On the modem you will find a relevant command to turn individual handshaking systems on or off. You may need to refer back to the 'Delving Deeper' sections on handshaking and check the way you have got the Link wired before you know which commands to select. You also have to consider Echo as it comes into this group.

The second group will just be relevant to the modem. They will involve commands to help the modem establish the connection over the telephone line. They range from whether you need the number dialled by pulses or tones, through to how long the modem is to wait before automatically answering the phone.

The third group is made up of parameters on your Organiser's Setup Sheet which the modem will not need to know about at all. This group will include Protocol, REOF/TEOF and any character translations. These are only relevant to the two computers at either end of the link and are invisible to the modems in between.

Changing Parameters on a Modem

To change the settings on your modem you will need to put your Organiser into Terminal mode and have the modem plugged into the Comms Link. After pressing 'AT' the modem will send back 'OK'.

To change anything in the modem's setup sheet or indeed change any other command you will need to find the Hayes code for whatever command you want to use. You will find a whole list in your modem's manual. To enter a command you type 'AT' followed by one or two letters which tell the

modem which parameter you want to change, then you normally need a number to tell it which option you want to set it to.

For example, 'E' is for Echo, 'E0' will turn echo off, whilst 'E1' will turn it on.

'H' is for Hook, 'H0' puts the metaphoric receiver back on the hook (hangs up), whilst 'H1' pick up the phone line (takes it off the hook).

Modems have a number of memory locations where additional parameter settings are stored. These are separate to the User profiles (or Setup Sheets) and are called Registers. Register No 7 for example controls the number of seconds the modem waits after it has dialled a telephone number, before it receives a tone or carrier from that number. It is normally set to 30 seconds, but this may not be long enough if you are dialling to the USA on pulse dialling. To change the value in this register you simply type 'ATS7 = 90' which changes the time to 90 seconds.

CCITT/BELL tones

Another command that is often overlooked is the CCITT/BELL tones. There are currently two standards of tones that are used on the telephone. The CCITT standard has been adopted by the UK and most European countries, whilst in the USA the BELL standard is used. Most BT (British Telecom) approved modems are set to CCITT. If the modem is set to the wrong standard it will not be able to connect. You can change this to BELL when you are calling the USA direct by using the following command.

ATB1

Of course this facility can also be used the other way round allowing cheaper imported modems to be used on the CCITT standard.

ATB0

Trying the Changes Out

Printers

The only way to know definitely that you have got everything set to the correct parameters is to try it out. Note that if you are printing from the terminal many printers will only start printing after they have a certain amount of data in their buffer. Low cost daisywheel printers usually print individual characters. Dot matrix printers need a line of text normally, but you can test them by typing a few characters followed by a carriage return. Laser printers normally need a whole page before they start work, but here again if you type in a few characters followed by a form feed character that is Left Arrow 'L' it will print a page straight away. Some laser printers need the form feed at the start of a line - immediately after a carriage return.

If your printer is printing but not printing the right characters you can see the ASCII codes that it is receiving by putting it into Hexadecimal dump mode and sending some more characters. By comparing the codes it is actually receiving to the characters you are sending you should be able to work out what is going wrong.

One of the commonest problems of this sort is caused by the printer working with the other part of the ASCII table which

you can put right by using parity as an extra Stop bit. See section on Parity to refresh your memory.

Changing Parameters after the Initial Setup

Hayes Modems

To change any of the parameters later is easy, because you just use the 'AT' command as before. If, however, you need to send the modem a command whilst communication is in progress you must get your modem's ATtention by sending three escape characters in very quick succession.

The escape character for a Hayes modem is a '+' sign. You need to type the three '+'s quickly because of something called the 'guard time'. The guard time is initially set to a fiftieth of a second (although you can change it) and if there is a greater time gap than that between key presses the previous '+' will be disregarded and you will not be able to enter the command state. This has been done to prevent you accidentally entering the modem's 'Setup Sheet' in the middle of sending a file because that would stop the transfer in its tracks. It is fairly easy to get '+'s that quickly on the Organiser's keyboard, just hold the SHIFT key down and press the 'x' down repeatedly. If you prefer, press SHIFT and DOWN Arrow together - then you only have to worry about the 'x' key. You must wait for 1 or 2 seconds after you have sent the three '+'s before sending any other character, otherwise it will be ignored.

Printers

Changing parameters on a printer after you have set the communication parameters is normally remarkably similar to setting the parameter values on a modem. On most printers you again use ESC (only one this time!) followed by the relevant codes. If you look back at the terminal section you will find the key strokes for characters you haven't got on the Organiser's keyboard. For example ESC is RIGHT Arrow plus A. If you look in your printer's manual and find out the code you need to type in to change the parameter settings and then find the keys you need to press in terminal mode then the parameter should be changed.

For example, suppose you wanted to change the print style to underlined condensed print, you would first look in the printer manual for the command. On our printer the general change print style command is 'ESC!' followed by a number. The number for condensed is four (if you are using more than one available print style you have to add the numbers together) added to 128 (the code for underlining) gives you 132. So the full code is 'ESC! 132'. To type that into the Organiser's Terminal mode you need to press 'RIGHT Arrow' 'A' 'RIGHT Arrow' 'F' '132'.

Saving the Changes

Printers

If the changes you are making are going to be useful to do later on as well as now, you can save them into a file by typing the codes into your Organiser via the Capture Editor after clearing anything that was already in the capture buffer.

Once the codes are in the buffer you can test them by transmitting them to your printer and checking that they do what you thought they should by leaving Capture, going back into the terminal and printing a line of text. If that works you can go back into Capture and save the codes to a file that you will name appropriately. You will then be able to re-transmit the file whenever you want your printer or modem to obey that string of commands.

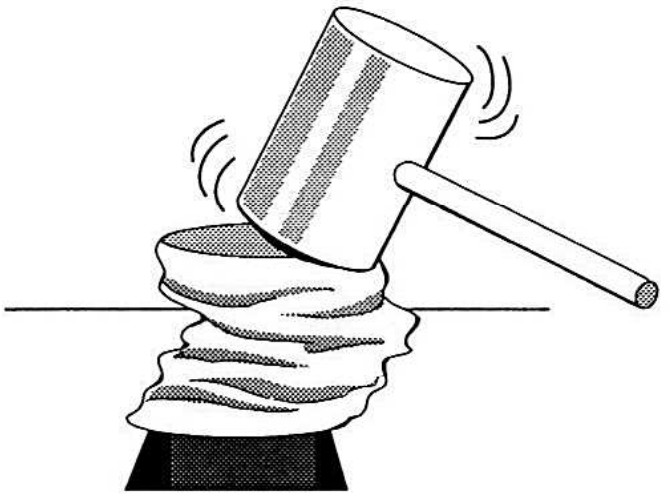
Modems

You may have more than one computer that you want to communicate with and your modem may have the ability to store more than one set of parameters. Our modem has enough space in memory to store two complete sets of parameters. They are called User Profiles 1 and 2 and you can use them like Organiser setup files.

If your modem doesn't have the facility of storing setups internally, or you need access to more than two, you can always save the whole set by Capturing the commands as you type them in and saving them as we described above. Once you have a file stored in your Organiser all you need to do is transmit that file back to the modem whenever you need them.

CHAPTER 7

File Conversion



When transferring files between different computer systems it is not always possible to use the same software package on both sides of the link. It is therefore important that the file you are sending is in a form that the receiving software can deal with. Otherwise a file that starts across the link as a nicely ordered database or Diary file could change into an unrecognisable mess on arrival.

It would be impossible for us to tell you how to convert your Organiser's database files into every possible database pack-

age on the market. There are just too many of them for that. We would have to consider all the other applications as well.

So instead of trying to do the impossible, we will adopt another approach. We will show you how you can convert your Organiser file into standard formats (for several different applications) that have the facilities to convert into a number of other file formats.

There are several software packages that have become standard data formats, partly because they are what other packages can be interfaced with (or perhaps because you can convert so many other file formats into theirs - we don't know which came first). If you know how to convert a file into one of these (unofficial) standard formats, there are normally a number of opportunities to convert it from that to what you really need. In effect these standard formats can act as a 'halfway house' between two incompatible formats.

Databases

Database files must have a character which is only used to signify the end of one field and the beginning of another. Most database packages can import files in Data Interchange Format (DIF) which is comma delimited. That means that the program places a comma between fields which you may not see.

On the Organiser, you see records on the display as one field per line, but the whole record is stored in memory as one line. The difference between the Organiser and other databases is that the fields are separated by TAB characters rather than commas. In this case it is fairly easy to convert the Organiser file to a comma delimited file; you just put the

relevant codes into the RTRN and TTRN parameters on the Setup Sheet. Put in the ASCII codes for the character you want to change, followed by the code for the character you want to replace it with (the codes should be separated by a comma!). See RTRN and TTRN section in the Comms Link chapter.

Once you have sent the file across the link with commas instead of tabs in between fields, you can then consult the manual for your database package. It will tell you what you need to do to the file to get it into the format that that particular package needs.

Sending/Receiving Only Part of a Database

It is very likely that you will not want to load the whole of a database into your Organiser, on grounds of space alone.

There are two ways that you can make the database smaller for your Organiser.

The first way is to use the 'Filter' or 'Select' commands in your database on the other machine. Filter will allow you to select certain records whilst Select will allow you to choose only some fields in the records. This will reduce the number of fields or records to send and store on the Organiser.

If you need less fields on your Organiser database than you have got on the other one. Another way that you can cut down the number of fields is to use your other computer's Mailmerge program.

What you need to do is to make up a dummy letter with all the page layout options set to zero, and which just contain the relevant field identifiers (which can be in any order you

like) in a line separated by a comma. If you then 'print' the 'letter' to a file it will automatically extract the fields you have asked for from your database and load them into the file. The file is now ready for you to send across the link to your Organiser.

To add records that you have collected on your Organiser to your 'master' database, you send the file across the link and make up a dummy letter, again setting all the page layout options to zero. This time you have to put in as many commas as you have fields in your master database. You then put in the field identifiers for the fields you have got in the Organiser, in between the two relevant commas wherever you want those fields to be in the master database. Once your dummy letter is ready you then print it to a file. You will then in effect have two database files; your master and the one you have just created and you should be able to update you master by merging two files together.

Wordprocessor Files

You may be asking why you need to know about wordprocessors when Psion don't supply a wordprocessor package. But there are wordprocessors available from third party suppliers. Also you can use the line editor in PROG or NOTES to produce text files. Finally you may have something that you have started on your wordprocessor that you want to take out with you on your Organiser.

If you want to transfer a file into a wordprocessor package, or back to the Organiser from one, you have two options. The simplest one is to send the file across the link as it is, an ASCII file. This may be known as a 'plain text' or 'non-document' file on your other machine.

If your wordprocessor will accept an ASCII file the conversion is easy because there will be nothing to convert! All you do is send it across the link, save it on the receiving computer, load it into the wordprocessor and see what you have got! If you are lucky the file will look the same as it did before it left your Organiser. If it doesn't you may have to send it again after setting the TTRN parameter to translate the characters that are different. See the section on translation parameters for more details.

If your word processor cannot accept straight ASCII but can convert WordStar (tm) files into whatever format it needs to work with, you will need to send it across the link in WordStar format and then load or import it into your wordprocessor.

Sending a file from the Organiser in WordStar format is simple because the only thing you must change is TEOL. There is a problem here because WordStar actually uses two different end of line characters, one for 'soft' carriage returns where the line has been wrapped down onto the next line and the end of line could change, and one for 'hard' carriage returns for places like the end of paragraphs where it is not going to change. The ASCII code for the hard one is the same as the normal carriage return (13) whilst the code for a soft one is '141'. Whichever one you put in as TEOL you will have to do some adjustment manually on the other side, but we recommend that you try preserving the soft one first because there will be less of the hard returns to change. If that doesn't come out right change TEOL to the other one. To convert a file from WordStar format so that you can use it on the Organiser you need to strip off all the control characters that WordStar uses (which will also convert the soft carriage returns so it will solve that problem as well!). To do that is simple. You knock off the last Bit in every byte

by changing the BIT parameter to '7'. You will not however be able to use a protocol to receive the file because they need the eighth bit.

Notepads and OPL Files

You can use the same process as we discussed above for converting wordprocessor files to convert Notepad files that do not contain any calculations. You can also use them to convert OPL text files. You will not be able to convert their sister binary files (Notepads with calculations in them and OPL object files) without the help of OPL procedures.

Spreadsheets

Spreadsheets are binary files which are not easy to convert without added software. If you have the Psion Pocket Spreadsheet package you do have the ability to convert the files into '.WKS format' which can be used by some spreadsheet packages, notably Lotus 123 (tm). But you can only do that if you have an IBM PC or AT running Psion's CL software. Without that you would need an OPL program which the User group (IPSO) might be able to help you with.

Diary Files

When you want to convert Diary files you are in a similar situation to that with spreadsheets. If you have a model CM or XP you need the Diary Link software package, which is only available for the IBM PC or AT only if you use SideKick. If you have a model LZ you can work with your Diary on another computer because they use the normal text format. If you have a CM Diary file you will only be able to

send it to another computer by first loading it into an LZ (via a Ram/Datapak) and getting it translated into LZ Diary format by using Xrestore.

Sending Files across the Link for Storage

All these file conversion considerations disappear if you want to just store files on the machine at the other end of the link. You only have to worry about converting them if you want to work with them while they are there. So using your other computer for backing up important Organiser files is simply a matter of sending them across the Link and storing them there until you need them.

CHAPTER 8

Epilogue

Now that we are at the end of this book we are going to mix metaphors by putting in the boot and throwing in the towel!

Boot

Boot enables you to Run OPL procedures that you have stored in your IBM PC or Apple Mac, as long as it is running CL. This is useful if you are short of space in your Organiser's memory because you can send both the text file and object file (the translated copy) to the PC or Mac and delete the entire procedure from the Organiser

Throwing in the Towel

In the extremely unlikely event that you have read this book and still can't solve your problem..... Well all right we surrender! But if we can't help you we know someone who can! Here are two helpful addresses:-

Technical Support
Psion PLC
Psion House
Harcout Street
London W1H 1DT

Tel: 071 262 5580

They have a customer helpline on 071 262 8188

In addition to Psion you could also contact the user group whose members have a wealth of experience between them. Their address is:-

IPSO (Independent Psion Organiser User Group)
130 Stapleford Lane
Beeston
Nottingham
NG9 1GB

They also publish a newsletter which is edited by Mike O'Regan who can be contacted at the above address.

Bibliography and Further Reading

RS232 Simplified by Byron W Putman published by Prentice-Hall, New Jersey

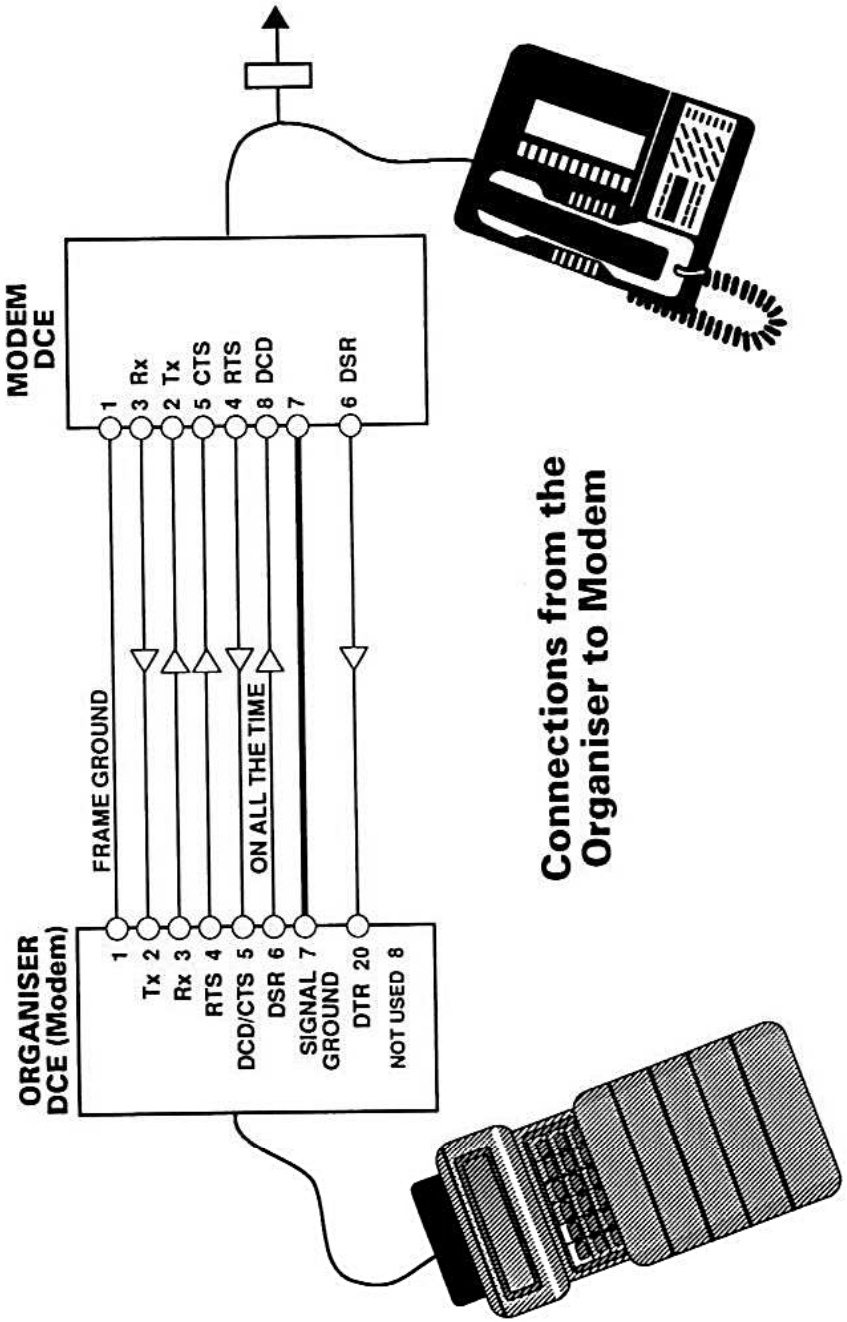
RS232 Made Easy by Martin D Seyer published by Prentice-Hall, New Jersey

Interfacing Standards for Computers by A C Maine published by The Institute of Electrical and Electronic Incorporated Engineers, London

Note

The above books contain extracts of the various standards in operation. You can obtain the full unabridged version of the RS232C standard from:

EIA Engineering Dept. Standards Sales 2001 Eye St. N.W. Washington D.C. 20006 (202) 457-4966



Solving Printer Problems

There are four types of problems that you could have with your Printer.

We have listed them below, together with corresponding solutions.

	Symptom	Suggested Actions
POWER	Printer does not respond when turned on (no lights or carriage movement)	Check the Power Cable is connected both in the mains socket and the printer
		Try the same mains socket with another appliance and see if that works
		Check/Replace any fuses with the correct value
	Add paper indicator flashes - printer stops	Press the On Line button
		Turn the printer off (for 5 seconds) and then on again
		Try another Mains socket
INTERFACE	Printer has power but does not respond to host	Attach Printer cable
		Place printer 'online'

	Symptom	Suggested Action
		Check that you have set the printer up in text mode and not graphics
		Check the SETUP parameters are correct.
PRINT	No print - printer operating but carriage moves	Check Ink Ribbon is properly located
		Move printer head closer to the paper
	Poor print quality	Change printer ribbon
		Adjust Printer head position
	Line of text characters carries over to the next line.	Change the printer setting to use 12 cpi instead of 10 cpi
		Change the WIDTH setting on the Setup Sheet to a number of characters your printer can print on a line.
PAPER	Paper does not feed properly	Remove paper from printer and discard any damaged pages
		If using tractor feed paper adjust the width of the tractor roller to match the holes in the paper

Symptom	Suggested Action
	Check clean the paper path in the printer. Paper dust and other debris accumulate in printers, use a small vacuum cleaner to remove
	Paper can curl on the platen when the printer is not used for a period. Check printing of the first sheet to make sure it exits properly

9 to 25 way adapter Pin-outs

IBM AT	Comms Link
9 way socket	25 way Plug
1	8
2	3
3	2
4	20
5	7
6	6
7	4
8	5
9	22

RS-232 Pinouts and Circuit Numbers.

No	Name	No	Name	DIRECTION	Pin
101	Protective Ground	AA	Protective Ground		1
102	Signal Ground	AB	Signal Ground		7
103	Transmitted Data	BA	Transmitted Data	To modem	2
104	Received Data	BB	Received Data	From modem	3
105	Request to Send	CA	Request to Send	To modem	4
106	Ready for Sending	CB	Clear to Send	From modem	5
107	Data Set Ready	CC	Data Set Ready	From modem	6
108.1	Connect Data Set to line	CD	Data Terminal Ready	To modem	20
108.2	Data Terminal Ready	CD	Data Terminal Ready	To modem	20
109	Received Line Signal Detector	CF	Data Carrier Detect	From modem	8
109	Data channel Received/	CF	Data Carrier Detect	From modem	8
110	Data Signal Quality Selector	CG	Signal Quality Detector	From modem	21
111	Data Signal rate selector (DTE)	CH	Speed Selector (only ONE used)	To modem	23
112	Data Signal Rate Selector (DCE)	CI	Speed Selector (only ONE used)	From modem	23
113	Tx Signal Element Timing (DTE)	DA	Tx Signal element timing	To modem	24
114	Tx Signal element timing	DB	Tx Signal element timing	From modem	15
115	Rx Signal element timing (DCE)	DD	Rx Signal element timing	From modem	17
116	Tx Block Timing			To modem	
117	Rx Block Timing			From modem	
118	Tx S Channel Data (Tx Backwards)	SBA	S Tx Data	To modem	14
119	Rx S Channel Data (Rx Backwards)	SBB	S Rx Data	From modem	16
120	Tx S Channel Carrier(Tx Baokwards)	SCA	S Request-to-Send	To modem	19
121	S (Backward) Channel Ready	SCB	S Clear-to-Send	From modem	13

No	Name	No	Name	DIRECTION	Pin
122	S (Backward)Channel Carrier Detect	SCF	S Received line signal detector	From modem	12
123	S Channel Signal Quality Detector			From modem	
124	Data receiver cut OFF				
125	Calling Indicator	CE	Ring Indicator	From modem	22
126	Select Tx Frequency			To modem	
127	Select Rx Frequency			To modem	
128	Rx Signal Element Timing (DTE)			To modem	
129	S Receiver Cut-Off			To modem	
			(Reserved for Data Set Testing)		10
			(Reserved for Data Set Testing)		9
			Unassigned		11
			Unassigned		18
			Unassigned		25

Key

S = Secondary

Summary of CCITT Recommendations

Name	Speed	Code - See Key		Comments
V1				Equivalence between binary notation symbols and the significant conditions of a two conditioned code.
V2				Power levels for data transmission over telephone lines
V3				International Alphabet No 5
V4				General structure of signals of International Alphabet No.5 code for data transmissions over public telephone networks.
V5				Standardisation of modulation rates and data signalling rates for synchronous data transmission in general switched network.
V6				Same as V5, on leased telephone-type circuits.
V10				Electrical characteristics for unbalanced double-current interchange circuits for general use with integrated circuit equipment in the field of data communications.
V11				Same as V10, but for balanced double-current interchange circuits.

Name	Speed	Code - See Key		Comments
V13				Answerback unit simulator
V15				Use of acoustic coupling for data transmission
V16				Recommendation for modems for transmissions of medical dialogue data.
V19				Modems for parallel data transmissions using telephone signalling frequencies
V20				Parallel data transmission modems standardised for universal use in the general switched network.
V21	300	F	A	300 baud modem standardised for use in the switched telephone network
V22	1200	F	A	1200 bps full-duplex 2-wire modem standardised for use in the general switched telephone network
V22bis	2400	F	A	As V22 but 2400 bps.
V23	1200/75	F	A	Split Speed modem standardised for use in the general switched telephone network
	600/1200	H	A	
	1200/1200	H	A	
V24				List of definitions for interchange circuits between DTE Data Terminal Equipment and DCE Data Circuit-terminating equipment (i.e. Modem)

Name	Speed	Code - See Key		Comments
V25				Automatic calling and/or answering equipment on the general switched telephone network
V25bis	9600	F	A	CCITT auto answer / auto dial standard
V26	2400	L	S	2400 bps modem for use on 4 wire leased point-to-point circuits
V26bis	2400	H	S	2400/1200 bps modem standardised for use in the general switched telephone network
V27	4800	L	S	4800 bps modem for leased circuit
V27bis	4800/2400	L	S	4800/2400 bps modem with automatic adaptive equaliser standardised for use on leased circuits
V27ter	4800	H	S	4800/2400 bps modem standardised for use in the general switched telephone network
V28				Electrical characteristics for unbalanced double-current interchange circuits
V29	9600	L	S	9600 bps modem for use on leased circuits
V31				Electrical characteristics for single-current interchange circuits controlled by contact closure
V32	9600	D	A	9600 bps modem standardisation for use in general switched network

Name	Speed	Code - See Key		Comments
V33	14400	L	S	
V35	48000			Data transmission at 48 kilobits per second using 60-108 kHz group band circuits
V36				Modems for synchronous data transmission using 60-108 kHz group band circuits
V40				Error indication with electromechanical equipment
V41				Code-independent error control system
V42				Error Correction standard incorporating MNP4 and LAP-M
V42bis				Data compression standard used in conjunction with V42 giving up to a 4:1 data compression ratio.
V50				Standard limits for transmission quality of data transmission
V51				Organisation of the maintenance of international telephone-type circuits used for data transmission
V52				Characteristics of distortion and error rate measuring apparatus for data transmission
V53				Limits for the maintenance of telephone type circuits used for data transmission

Name	Speed	Code - See Key	Comments
V54			Loop test devices for modems
V55			Specification for an impulsive noise measuring instrument for telephone type circuits
V56			Comparative tests for modems for use over telephone-type circuits

Other Technical Terms

EPAD	Error corrected entry into PSS (Packet Switch) or IPSS
MNP	Microcom Network Protocol - a popular error correction scheme
ARQ	Automatic Re-send reQuest - an error correction scheme
PULSE	Loop disconnect method of dialling, used on PSTN lines
DTMF	Dual Tone Multi Frequency dialling used on some switchboards and certain new System-X B.T. exchanges
bps	Bits Per Second - asynchronous normally at 10 bits for each character
AA	Auto-Answer. Modem able to "answer" the phone.
AD	Auto-Dial. Modem able to dial a telephone number.
HAYES	Modem command codes

LL	Leased Line - a permanent point-to-point, 2 or 4 wire circuit
B.T	British Telecom - the main British telephone network supplier.
PSTN	Public Switched Telephone Network - the B.T. phone system
PABX	Private Automatic Branch eXchange - in house switchboard
PSS	Packet Switch Stream - B.T.'s public X25 network
X25	
IPSS	International Packet Switch Stream
Packet	A method of sending data from a number of different locations to other different locations down the same channel by splitting the message into blocks, adding and addressing each one.
CCITT	Governing body that sets communications standards. Part of the United Nations.
Error	Ensures data transmission is free Correction from errors caused by line noise
MNP4	Error Correction standard
LAM-M	Error Correction standard

Key

F	Full Duplex
H	Half Duplex
L	Leased Line
A	Asynchronous
S	Synchronous

Useful Items

Maplin

Gender Changers
25 Way D Connectors

M/M (low profile)	JM48C
F/F	JM49D

Adapters

9F/25M Changer	JM08J
9F/25F Changer	JM50E

Testing Bits and BOB's

Wiring Box	YP79L
RS232 Mini Tester or	YP80B
RS232 Quick Tester	YP81C

Permanent Changes use the:-

RS232 Jumper Box	YP78K

Maplin Electronics PO Box 3 Rayleigh Essex SS6 8LR
Tel Sales : 0702 554161

Framing Error Detection Table

The tables on the following pages are an ASCII table plus some more. As well as showing the ASCII Hexadecimal and Decimal codes for your Organiser's complete character set, it also shows you the binary code that is actually sent and how that is effected by the framing parameters.

You use these tables when your Organiser or the 'other' machine are receiving one or more characters that are different to the ones that have been sent. If you compare the binary codes for both the characters on the table you should be able to get some clues as to what is going wrong.

Key to the heading of the binary column

The code reads from left to right

The upper case S is the Start bit The numbers from 1 to 7 (or 8) are the Data bits The upper case 'P' is the Parity bit (when used) the lower case 'S' is the Stop bit

Note:

For the Psion Organiser character set above 128 in the 8 Bit listing, see the character set chart on page 211.

If there is a conflict between the tables and the Organiser chart on page 211, the Organiser chart takes precedence.

7 Bit

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
NULL	07	00	7	N	1	000000001
SOH	1	01	7	N	1	010000001
STX	2	02	7	N	1	001000001
ETX	3	03	7	N	1	011000001
EOT	4	04	7	N	1	000100001
ENQ	5	05	7	N	1	010100001
ACK	6	06	7	N	1	001100001
BEL	7	07	7	N	1	011100001
BS	8	08	7	N	1	000010001
HT	9	09	7	N	1	010010001
LF	10	0A	7	N	1	001010001
VT	11	0B	7	N	1	011010001
FF	12	0C	7	N	1	000110001
CR	13	0D	7	N	1	010110001
SO	14	0E	7	N	1	001110001
SI	15	0F	7	N	1	011110001
DLE	16	10	7	N	1	000001001
DC1	17	11	7	N	1	010001001
DC2	18	12	7	N	1	001001001
DC3	19	13	7	N	1	011001001
DC4	20	14	7	N	1	000101001
NAK	21	15	7	N	1	010101001
SYN	22	16	7	N	1	001101001
ETB	23	17	7	N	1	011101001
CAN	24	18	7	N	1	000011001
EM	25	19	7	N	1	010011001
SUB	26	1A	7	N	1	001011001
ESC	27	1B	7	N	1	011011001
FS	28	1C	7	N	1	000111001
GS	29	1D	7	N	1	010111001
RS	30	1E	7	N	1	001111001
US	31	1F	7	N	1	011111001
SP	32	20	7	N	1	000000101
!	33	21	7	N	1	010000101
"	34	22	7	N	1	001000101
£	35	23	7	N	1	011000101
$	36	24	7	N	1	000100101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
%	37	25	7	N	1	010100101
&	38	26	7	N	1	001100101
'	39	27	7	N	1	011100101
(	40	28	7	N	1	000010101
)	41	29	7	N	1	010010101
*	42	2A	7	N	1	001010101
+	43	2B	7	N	1	011010101
,	44	2C	7	N	1	000110101
-	45	2D	7	N	1	010110101
.	46	2E	7	N	1	001110101
/	47	2F	7	N	1	011110101
0	48	30	7	N	1	000001101
1	49	31	7	N	1	010001101
2	50	32	7	N	1	001001101
3	51	33	7	N	1	011001101
4	52	34	7	N	1	000101101
5	53	35	7	N	1	010101101
6	54	36	7	N	1	001101101
7	55	37	7	N	1	011101101
8	56	38	7	N	1	000011101
9	57	39	7	N	1	010011101
:	58	3A	7	N	1	001011101
;	59	3B	7	N	1	011011101
<	60	3C	7	N	1	000111101
=	61	3D	7	N	1	010111101
>	62	3E	7	N	1	001111101
?	63	3F	7	N	1	011111101
@	64	40	7	N	1	000000011
A	65	41	7	N	1	010000011
B	66	42	7	N	1	001000011
C	67	43	7	N	1	011000011
D	68	44	7	N	1	000100011
E	69	45	7	N	1	010100011
F	70	46	7	N	1	001100011
G	71	47	7	N	1	011100011
H	72	48	7	N	1	000010011
I	73	49	7	N	1	010010011
J	74	4A	7	N	1	001010011
K	75	4B	7	N	1	011010011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
L	76	4C	7	N	1	000110011
M	77	4D	7	N	1	010110011
N	78	4E	7	N	1	001110011
O	79	4F	7	N	1	011110011
P	80	50	7	N	1	000001011
Q	81	51	7	N	1	010001011
R	82	52	7	N	1	001001011
S	83	53	7	N	1	011001011
T	84	54	7	N	1	000101011
U	85	55	7	N	1	010101011
V	86	56	7	N	1	001101011
W	87	57	7	N	1	011101011
X	88	58	7	N	1	000011011
Y	89	59	7	N	1	010011011
Z	90	5A	7	N	1	001011011
[	91	5B	7	N	1	011011011
\	92	5C	7	N	1	000111011
]	93	5D	7	N	1	010111011
^	94	5E	7	N	1	001111011
_	95	5F	7	N	1	011111011
`	96	60	7	N	1	000000111
a	97	61	7	N	1	010000111
b	98	62	7	N	1	001000111
c	99	63	7	N	1	011000111
d	100	64	7	N	1	000100111
e	101	65	7	N	1	010100111
f	102	66	7	N	1	001100111
g	103	67	7	N	1	011100111
h	104	68	7	N	1	000010111
i	105	69	7	N	1	010010111
j	106	6A	7	N	1	001010111
k	107	6B	7	N	1	011010111
l	108	6C	7	N	1	000110111
m	109	6D	7	N	1	010110111
n	110	6E	7	N	1	001110111
o	111	6F	7	N	1	011110111
p	112	70	7	N	1	000001111
q	113	71	7	N	1	010001111
r	114	72	7	N	1	001001111
s	115	73	7	N	1	011001111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
t	116	74		N	1	000101111
u	117	75	7	N	1	010101111
v	118	76	7	N	1	001101111
w	119	77	7	N	1	011101111
x	120	78	7	N	1	000011111
y	121	79	7	N	1	010011111
z	122	7A	7	N	1	001011111
{	123	7B	7	N	1	011011111
\| | 124 | 7C | 7 | N | 1 | 000111111 |
} | 125 | 7D | 7 | N | 1 | 010111111 |
~ | 126 | 7E | 7 | N | 1 | 001111111 |
DEL | 127 | 7F | 7 | N | 1 | 011111111 |

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
NULL	0	00	7	E	1	0000000001
SOH	1	01	7	E	1	0100000011
STX	2	02	7	E	1	0010000011
ETX	3	03	7	E	1	0110000001
EOT	4	04	7	E	1	0001000011
ENQ	5	05	7	E	1	0101000001
ACK	6	06	7	E	1	0011000001
BEL	7	07	7	E	1	0111000011
BS	8	08	7	E	1	0000100011
HT	9	09	7	E	1	0100100001
LF	10	0A	7	E	1	0010100001
VT	11	0B	7	E	1	0110100011
FF	12	0C	7	E	1	0001100001
CR	13	0D	7	E	1	0101100011
SO	14	0E	7	E	1	0011100011
SI	15	0F	7	E	1	0111100001
DLE	16	10	7	E	1	0000010011
DC1	17	11	7	E	1	0100010001
DC2	18	12	7	E	1	0010010001
DC3	19	13	7	E	1	0110010011
DC4	20	14	7	E	1	0001010001
NAK	21	15	7	E	1	0101010011
SYN	22	16	7	E	1	0011010011
ETB	23	17	7	E	1	0111010001
CAN	24	18	7	E	1	0000110001
EM	25	19	7	E	1	0100110011
SUB	26	1A	7	E	1	0010110011
ESC	27	1B	7	E	1	0110110001
FS	28	1C	7	E	1	0001110011
GS	29	1D	7	E	1	0101110001
RS	30	1E	7	E	1	0011110001
US	31	1F	7	E	1	0111110011
SP	32	20	7	E	1	0000001011
!	33	21	7	E	1	0100001001
" | 34 | 22 | 7 | E | 1 | 0010001001 |
£ | 35 | 23 | 7 | E | 1 | 0110001011 |
$ | 36 | 24 | 7 | E | 1 | 0001001001 |
% | 37 | 25 | 7 | E | 1 | 0101001011 |
& | 38 | 26 | 7 | E | 1 | 0011001011 |
' | 39 | 27 | 7 | E | 1 | 0111001001 |

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
(	40	28	7	E	1	0000101001
)	41	29	7	E	1	0100101011
*	42	2A	7	E	1	0010101011
+	43	2B	7	E	1	0110101001
,	44	2C	7	E	1	0001101011
-	45	2D	7	E	1	0101101001
.	46	2E	7	E	1	0011101001
/	47	2F	7	E	1	0111101011
0	48	30	7	E	1	0000011001
1	49	31	7	E	1	0100011011
2	50	32	7	E	1	0010011011
3	51	33	7	E	1	0110011001
4	52	34	7	E	1	0001011011
5	53	35	7	E	1	0101011001
6	54	36	7	E	1	0011011001
7	55	37	7	E	1	0111011011
8	56	38	7	E	1	0000111011
9	57	39	7	E	1	0100111001
:	58	3A	7	E	1	0010111001
;	59	3B	7	E	1	0110111011
<	60	3C	7	E	1	0001111001
=	61	3D	7	E	1	0101111011
	62	3E	7	E	1	0011111011
?	63	3F	7	E	1	0111111001
@	64	40	7	E	1	0111111001
A	65	41	7	E	1	0100000101
B	66	42	7	E	1	0010000101
C	67	43	7	E	1	0110000111
D	68	44	7	E	1	0001000101
E	69	45	7	E	1	0101000111
F	70	46	7	E	1	0011000111
G	71	47	7	E	1	0111000101
H	72	48	7	E	1	0000100101
I	73	49	7	E	1	0100100111
J	74	4A	7	E	1	0010100111
K	75	4B	7	E	1	0110100101
L	76	4C	7	E	1	0001100111
M	77	4D	7	E	1	0101100101
N	78	4E	7	E	1	0011100101
O	79	4F	7	E	1	0111100111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
P	80	50	7	E	1	0000010101
Q	81	51	7	E	1	0100010111
R	82	52	7	E	1	0010010111
S	83	53	7	E	1	0110010101
T	84	54	7	E	1	0001010111
U	85	55	7	E	1	0101010101
V	86	56	7	E	1	0011010101
W	87	57	7	E	1	0111010111
X	88	58	7	E	1	0000110111
Y	89	59	7	E	1	0100110101
Z	90	5A	7	E	1	0010110101
[	91	5B	7	E	1	0110110111
\	92	5C	7	E	1	0001110101
]	93	5D	7	E	1	0101110111
^	94	5E	7	E	1	0011110111
_	95	5F	7	E	1	0111110101
`	96	60	7	E	1	0000001101
a	97	61	7	E	1	0100001111
b	98	62	7	E	1	0010001111
c	99	63	7	E	1	0110001101
d	100	64	7	E	1	0001001111
e	101	65	7	E	1	0101001101
f	102	66	7	E	1	0011001101
g	103	67	7	E	1	0111001111
h	104	68	7	E	1	0000101111
i	105	69	7	E	1	0100101101
j	106	6A	7	E	1	0010101101
k	107	6B	7	E	1	0110101111
l	108	6C	7	E	1	0001101101
m	109	6D	7	E	1	0101101111
n	110	6E	7	E	1	0011101111
o	111	6F	7	E	1	0111101101
p	112	70	7	E	1	0000011111
q	113	71	7	E	1	0100011101
r	114	72	7	E	1	0010011101
s	115	73	7	E	1	0110011111
t	116	74	7	E	1	0001011101
u	117	75	7	E	1	0101011111
v	118	76	7	E	1	0011011111
w	119	77	7	E	1	0111011101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
x	120	78	7	E	1	0000111101
y	121	79	7	E	1	0100111111
z	122	7A	7	E	1	0010111111
{	123	7B	7	E	1	0110111101
|	124	7C	7	E	1	0001111111
}	125	7D	7	E	1	0101111101
~	126	7E	7	E	1	0011111101
DEL	127	7F	7	E	1	0111111111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
NULL	0	00	7	O	1	0000000011
SOH	1	01	7	O	1	0100000001
STX	2	02	7	O	1	0010000001
ETX	3	03	7	O	1	0110000011
EOT	4	04	7	O	1	0001000001
ENQ	5	05	7	O	1	0101000011
ACK	6	06	7	O	1	0011000011
BEL	7	07	7	O	1	0111000001
BS	8	08	7	O	1	0000100001
HT	9	09	7	O	1	0100100011
LF	10	0A	7	O	1	0010100011
VT	11	0B	7	O	1	0110100001
FF	12	0C	7	O	1	0001100011
CR	13	0D	7	O	1	0101100001
SO	14	0E	7	O	1	0011100001
SI	15	0F	7	O	1	0111100011
DLE	16	10	7	O	1	0000010001
DC1	17	11	7	O	1	0100010011
DC2	18	12	7	O	1	0010010011
DC3	19	13	7	O	1	0110010001
DC4	20	14	7	O	1	0001010011
NAK	21	15	7	O	1	0101010001
SYN	22	16	7	O	1	0011010001
ETB	23	17	7	O	1	0111010011
CAN	24	18	7	O	1	0100110011
EM	25	19	7	O	1	0100110001
SUB	26	1A	7	O	1	0010110001
ESC	27	1B	7	O	1	0110110011
FS	28	1C	7	O	1	0001110001
GS	29	1D	7	O	1	0101110011
RS	30	1E	7	O	1	0011110011
US	31	1F	7	O	1	0111110001
SP	32	20	7	O	1	0000001001
!	33	21	7	O	1	0100001011
"	34	22	7	O	1	0010001011
£	35	23	7	O	1	0110001001
$	36	24	7	O	1	0001001011
%	37	25	7	O	1	0101001001
&	38	26	7	O	1	0011001001
'	39	27	7	O	1	0111001011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
(	40	28	7	O	1	0000101011
)	41	29	7	O	1	0100101001
*	42	2A	7	O	1	0010101001
+	43	2B	7	O	1	0110101011
,	44	2C	7	O	1	0001101001
-	45	2D	7	O	1	0101101011
.	46	2E	7	O	1	0011101011
/	47	2F	7	O	1	0111101001
0	48	30	7	O	1	0000011011
1	49	31	7	O	1	0100011001
2	50	32	7	O	1	0010011001
3	51	33	7	O	1	0110011011
4	52	34	7	O	1	0001011001
5	53	35	7	O	1	0101011011
6	54	36	7	O	1	0011011011
7	55	37	7	O	1	0111011001
8	56	38	7	O	1	0000111001
9	57	39	7	O	1	0100111011
:	58	3A	7	O	1	0010111011
;	59	3B	7	O	1	0110111001
<	60	3C	7	O	1	0001111011
=	61	3D	7	O	1	0101111001
	62	3E	7	O	1	0011111001
?	63	3F	7	O	1	0111111011
@	64	40	7	O	1	0000000101
A	65	41	7	O	1	0100000111
B	66	42	7	O	1	0010000111
C	67	43	7	O	1	0110000101
D	68	44	7	O	1	0001000111
E	69	45	7	O	1	0101000101
F	70	46	7	O	1	0011000101
G	71	47	7	O	1	0111000111
H	72	48	7	O	1	0000100111
I	73	49	7	O	1	0100100101
J	74	4A	7	O	1	0010100101
K	75	4B	7	O	1	0110100111
L	76	4C	7	O	1	0001100101
M	77	4D	7	O	1	0101100111
N	78	4E	7	O	1	0011100111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
O	79	4F	7	O	1	0111100101
P	80	50	7	O	1	0000010111
Q	81	51	7	O	1	0100010101
R	82	52	7	O	1	0010010101
S	83	53	7	O	1	0110010111
T	84	54	7	O	1	0001010101
U	85	55	7	O	1	0101010111
V	86	56	7	O	1	0011010111
W	87	57	7	O	1	0111010101
X	88	58	7	O	1	0000110101
Y	89	59	7	O	1	0100110111
Z	90	5A	7	O	1	0010110111
[	91	5B	7	O	1	0110110101
\	92	5C	7	O	1	0001110111
]	93	5D	7	O	1	0101110101
^	94	5E	7	O	1	0011110101
_	95	5F	7	O	1	0111110111
`	96	60	7	O	1	0000001111
a	97	61	7	O	1	0100001101
b	98	62	7	O	1	0010001101
c	99	63	7	O	1	0110001111
d	100	64	7	O	1	0001001101
e	101	65	7	O	1	0101001111
f	102	66	7	O	1	0011001111
g	103	67	7	O	1	0111001101
h	104	68	7	O	1	0000101101
i	105	69	7	O	1	0100101111
j	106	6A	7	O	1	0010101111
k	107	6B	7	O	1	0110101101
l	108	6C	7	O	1	0001101111
m	109	6D	7	O	1	0101101101
n	110	6E	7	O	1	0011101101
o	111	6F	7	O	1	0111101111
p	112	70	7	O	1	0000011101
q	113	71	7	O	1	0100011111
r	114	72	7	O	1	0010011111
s	115	73	7	O	1	0110011101
t	116	74	7	O	1	0001011111
u	117	75	7	O	1	0101011101
v	118	76	7	O	1	0011011101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
w	119	77	7	O	1	0111011111
x	120	78	7	O	1	0000111111
y	121	79	7	O	1	0100111101
z	122	7A	7	O	1	0010111101
{	123	7B	7	O	1	0110111111
|	124	7C	7	O	1	0001111101
}	125	7D	7	O	1	0101111111
~	126	7E	7	O	1	0011111111
DEL	127	7F	7	O	1	0111111101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
NULL	0	00	7	M	1	0000000011
SOH	1	01	7	M	1	0100000011
STX	2	02	7	M	1	0010000011
ETX	3	03	7	M	1	0110000011
EOT	4	04	7	M	1	0001000011
ENQ	5	05	7	M	1	0101000011
ACK	6	06	7	M	1	0011000011
BEL	7	07	7	M	1	0111000011
BS	8	08	7	M	1	0000100011
HT	9	09	7	M	1	0100100011
LF	10	0A	7	M	1	0010100011
VT	11	0B	7	M	1	0110100011
FF	12	0C	7	M	1	0001100011
CR	13	0D	7	M	1	0101100011
SO	14	0E	7	M	1	0011100011
SI	15	0F	7	M	1	0111100011
DLE	16	10	7	M	1	0000010011
DC1	17	11	7	M	1	0100010011
DC2	18	12	7	M	1	0010010011
DC3	19	13	7	M	1	0110010011
DC4	20	14	7	M	1	0001010011
NAK	21	15	7	M	1	0101010011
SYN	22	16	7	M	1	0011010011
ETB	23	17	7	M	1	0111010011
CAN	24	18	7	M	1	0000110011
EM	25	19	7	M	1	0100110011
SUB	26	1A	7	M	1	0010110011
ESC	27	1B	7	M	1	0110110011
FS	28	1C	7	M	1	0001110011
GS	29	1D	7	M	1	0101110011
RS	30	1E	7	M	1	0011110011
US	31	1F	7	M	1	0111110011
SP	32	20	7	M	1	0000001011
!	33	21	7	M	1	0100001011
"	34	22	7	M	1	0010001011
£	35	23	7	M	1	0110001011
$	36	24	7	M	1	0001001011
%	37	25	7	M	1	0101001011
&	38	26	7	M	1	0011001011
'	39	27	7	M	1	0111001011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
(	40	28	7	M	1	0000101011
)	41	29	7	M	1	0100101011
*	42	2A	7	M	1	0010101011
+	43	2B	7	M	1	0110101011
,	44	2C	7	M	1	0001101011
-	45	2D	7	M	1	0101101011
.	46	2E	7	M	1	0011101011
/	47	2F	7	M	1	0111101011
0	48	30	7	M	1	0000011011
1	49	31	7	M	1	0100011011
2	50	32	7	M	1	0010011011
3	51	33	7	M	1	0110011011
4	52	34	7	M	1	0001011011
5	53	35	7	M	1	0101011011
6	54	36	7	M	1	0011011011
7	55	37	7	M	1	0111011011
8	56	38	7	M	1	0000111011
9	57	39	7	M	1	0100111011
:	58	3A	7	M	1	0010111011
;	59	3B	7	M	1	0110111011
<	60	3C	7	M	1	0001111011
=	61	3D	7	M	1	0101111011
	62	3E	7	M	1	0011111011
?	63	3F	7	M	1	0111111011
@	64	40	7	M	1	0100000111
A	65	41	7	M	1	0010000111
B	66	42	7	M	1	0010000111
C	67	43	7	M	1	0110000111
D	68	44	7	M	1	0001000111
E	69	45	7	M	1	0101000111
F	70	46	7	M	1	0011000111
G	71	47	7	M	1	0111000111
H	72	48	7	M	1	0000100111
I	73	49	7	M	1	0100100111
J	74	4A	7	M	1	0010100111
K	75	4B	7	M	1	0110100111
L	76	4C	7	M	1	0001100111
M	77	4D	7	M	1	0101100111
N	78	4E	7	M	1	0011100111
O	79	4F	7	M	1	0111100111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
P	80	50	7	M	1	0000010111
Q	81	51	7	M	1	0100010111
R	82	52	7	M	1	0010010111
S	83	53	7	M	1	0110010111
T	84	54	7	M	1	0001010111
U	85	55	7	M	1	0101010111
V	86	56	7	M	1	0011010111
W	87	57	7	M	1	0111010111
X	88	58	7	M	1	0000110111
Y	89	59	7	M	1	0100110111
Z	90	5A	7	M	1	0010110111
[	91	5B	7	M	1	0110110111
\	92	5C	7	M	1	0001110111
]	93	5D	7	M	1	0101110111
^	94	5E	7	M	1	0011110111
_	95	5F	7	M	1	0111110111
`	96	60	7	M	1	0000001111
a	97	61	7	M	1	0100001111
b	98	62	7	M	1	0010001111
c	99	63	7	M	1	0110001111
d	100	64	7	M	1	0001001111
e	101	65	7	M	1	0101001111
f	102	66	7	M	1	0011001111
g	103	67	7	M	1	0111001111
h	104	68	7	M	1	0000101111
i	105	69	7	M	1	0100101111
j	106	6A	7	M	1	0010101111
k	107	6B	7	M	1	0110101111
l	108	6C	7	M	1	0001101111
m	109	6D	7	M	1	0101101111
n	110	6E	7	M	1	0011101111
o	111	6F	7	M	1	0111101111
p	112	70	7	M	1	0000011111
q	113	71	7	M	1	0100011111
r	114	72	7	M	1	0010011111
s	115	73	7	M	1	0110011111
t	116	74	7	M	1	0001011111
u	117	75	7	M	1	0101011111
v	118	76	7	M	1	0011011111
w	119	77	7	M	1	0111011111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
x	120	78	7	M	1	0000111111
y	121	79	7	M	1	0100111111
z	122	7A	7	M	1	0010111111
{	123	7B	7	M	1	0110111111
|	124	7C	7	M	1	0001111111
}	125	7D	7	M	1	0101111111
~	126	7E	7	M	1	0011111111
DEL	127	7F	7	M	1	0111111111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567Ps
NULL	0	00	7	S	1	0000000001
SOH	1	01	7	S	1	0100000001
STX	2	02	7	S	1	0010000001
ETX	3	03	7	S	1	0110000001
EOT	4	04	7	S	1	0001000001
ENQ	5	05	7	S	1	0101000001
ACK	6	06	7	S	1	0011000001
BEL	7	07	7	S	1	0111000001
BS	8	08	7	S	1	0000100001
HT	9	09	7	S	1	0100100001
LF	10	0A	7	S	1	0010100001
VT	11	0B	7	S	1	0110100001
FF	12	0C	7	S	1	0001100001
CR	13	0D	7	S	1	0101100001
SO	14	0E	7	S	1	0011100001
SI	15	0F	7	S	1	0111100001
DLE	16	10	7	S	1	0000010001
DC1	17	11	7	S	1	0100010001
DC2	18	12	7	S	1	0010010001
DC3	19	13	7	S	1	0110010001
DC4	20	14	7	S	1	0001010001
NAK	21	15	7	S	1	0101010001
SYN	22	16	7	S	1	0011010001
ETB	23	17	7	S	1	0111010001
CAN	24	18	7	S	1	0000110001
EM	25	19	7	S	1	0100110001
SUB	26	1A	7	S	1	0010110001
ESC	27	1B	7	S	1	0110110001
FS	28	1C	7	S	1	0001110001
GS	29	1D	7	S	1	0101110001
RS	30	1E	7	S	1	0011110001
US	31	1F	7	S	1	0111110001
SP	32	20	7	S	1	0000001001
!	33	21	7	S	1	0100001001
"	34	22	7	S	1	0010001001
£	35	23	7	S	1	0110001001
$	36	24	7	S	1	0001001001
%	37	25	7	S	1	0101001001
&	38	26	7	S	1	0011001001
'	39	27	7	S	1	0111001001

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
(	40	28	7	S	1	0000101001
)	41	29	7	S	1	0100101001
*	42	2A	7	S	1	0010101001
+	43	2B	7	S	1	0110101001
,	44	2C	7	S	1	0001101001
-	45	2D	7	S	1	0101101001
.	46	2E	7	S	1	0011101001
/	47	2F	7	S	1	0111101001
0	48	30	7	S	1	0000011001
1	49	31	7	S	1	0100011001
2	50	32	7	S	1	0010011001
3	51	33	7	S	1	0110011001
4	52	34	7	S	1	0001011001
5	53	35	7	S	1	0101011001
6	54	36	7	S	1	0011011001
7	55	37	7	S	1	0111011001
8	56	38	7	S	1	0000111001
9	57	39	7	S	1	0100111001
:	58	3A	7	S	1	0010111001
;	59	3B	7	S	1	0110111001
<	60	3C	7	S	1	0001111001
=	61	3D	7	S	1	0101111001
	62	3E	7	S	1	0011111001
?	63	3F	7	S	1	0111111001
@	64	40	7	S	1	0000000101
A	65	41	7	S	1	0100000101
B	66	42	7	S	1	0010000101
C	67	43	7	S	1	0110000101
D	68	44	7	S	1	0001000101
E	69	45	7	S	1	0101000101
F	70	46	7	S	1	0011000101
G	71	47	7	S	1	0111000101
H	72	48	7	S	1	0000100101
I	73	49	7	S	1	0100100101
J	74	4A	7	S	1	0010100101
K	75	4B	7	S	1	0110100101
L	76	4C	7	S	1	0001100101
M	77	4D	7	S	1	0101100101
N	78	4E	7	S	1	0011100101
O	79	4F	7	S	1	0111100101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
P	80	50	7	S	1	0000010101
Q	81	51	7	S	1	0100010101
R	82	52	7	S	1	0010010101
S	83	53	7	S	1	0110010101
T	84	54	7	S	1	0001010101
U	85	55	7	S	1	0101010101
V	86	56	7	S	1	0011010101
W	87	57	7	S	1	0111010101
X	88	58	7	S	1	0000110101
Y	89	59	7	S	1	0100110101
Z	90	5A	7	S	1	0010110101
[	91	5B	7	S	1	0110110101
\	92	5C	7	S	1	0001110101
]	93	5D	7	S	1	0101110101
^	94	5E	7	S	1	0011110101
_	95	5F	7	S	1	0111110101
`	96	60	7	S	1	0000001101
a	97	61	7	S	1	0100001101
b	98	62	7	S	1	0010001101
c	99	63	7	S	1	0110001101
d	100	64	7	S	1	0001001101
e	101	65	7	S	1	0101001101
f	102	66	7	S	1	0011001101
g	103	67	7	S	1	0111001101
h	104	68	7	S	1	0000101101
i	105	69	7	S	1	0100101101
j	106	6A	7	S	1	0010101101
k	107	6B	7	S	1	0110101101
l	108	6C	7	S	1	0001101101
m	109	6D	7	S	1	0101101101
n	110	6E	7	S	1	0011101101
o	111	6F	7	S	1	0111101101
p	112	70	7	S	1	0000011101
q	113	71	7	S	1	0100011101
r	114	72	7	S	1	0010011101
s	115	73	7	S	1	0110011101
t	116	74	7	S	1	0001011101
u	117	75	7	S	1	0101011101
v	118	76	7	S	1	0011011101
w	119	77	7	S	1	0111011101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S1234567s
x	120	78	7	S	1	0000111101
y	121	79	7	S	1	0100111101
z	122	7A	7	S	1	0010111101
{	123	7B	7	S	1	0110111101
|	124	7C	7	S	1	0001111101
}	125	7D	7	S	1	0101111101
~	126	7E	7	S	1	0011111101
DEL	127	7F	7	S	1	0111111101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678s
NULL	0	00	8	N	1	0000000001
SOH	1	01	8	N	1	0100000001
STX	2	02	8	N	1	0010000001
ETX	3	03	8	N	1	0110000001
EOT	4	04	8	N	1	0001000001
ENQ	5	05	8	N	1	0101000001
ACK	6	06	8	N	1	0011000001
BEL	7	07	8	N	1	0111000001
BS	8	08	8	N	1	0000100001
HT	9	09	8	N	1	0100100001
LF	10	0A	8	N	1	0010100001
VT	11	0B	8	N	1	0110100001
FF	12	0C	8	N	1	0001100001
CR	13	0D	8	N	1	0101100001
SO	14	0E	8	N	1	0011100001
SI	15	0F	8	N	1	0111100001
DLE	16	10	8	N	1	0000010001
DC1	17	11	8	N	1	0100010001
DC2	18	12	8	N	1	0010010001
DC3	19	13	8	N	1	0110010001
DC4	20	14	8	N	1	0001010001
NAK	21	15	8	N	1	0101010001
SYN	22	16	8	N	1	0011010001
ETB	23	17	8	N	1	0111010001
CAN	24	18	8	N	1	0000110001
EM	25	19	8	N	1	0100110001
SUB	26	1A	8	N	1	0010110001
ESC	27	1B	8	N	1	0110110001
FS	28	1C	8	N	1	0001110001
GS	29	1D	8	N	1	0101110001
RS	30	1E	8	N	1	0011110001
US	31	1F	8	N	1	0111110001
SP	32	20	8	N	1	0000001001
!	33	21	8	N	1	0100001001
"	34	22	8	N	1	0010001001
£	35	23	8	N	1	0110001001
$	36	24	8	N	1	0001001001
%	37	25	8	N	1	0101001001
&	38	26	8	N	1	0011001001

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678s
'	39	27	8	N	1	0111001001
(	40	28	8	N	1	0000101001
)	41	29	8	N	1	0100101001
*	42	2A	8	N	1	0010101001
+	43	2B	8	N	1	0110101001
,	44	2C	8	N	1	0001101001
-	45	2D	8	N	1	0101101001
.	46	2E	8	N	1	0011101001
/	47	2F	8	N	1	0111101001
0	48	30	8	N	1	0000011001
1	49	31	8	N	1	0100011001
2	50	32	8	N	1	0010011001
3	51	33	8	N	1	0110011001
4	52	34	8	N	1	0001011001
5	53	35	8	N	1	0101011001
6	54	36	8	N	1	0011011001
7	55	37	8	N	1	0111011001
8	56	38	8	N	1	0000111001
9	57	39	8	N	1	0100111001
:	58	3A	8	N	1	0010111001
;	59	3B	8	N	1	0110111001
<	60	3C	8	N	1	0001111001
=	61	3D	8	N	1	0101111001
	62	3E	8	N	1	0011111001
?	63	3F	8	N	1	0111111001
@	64	40	8	N	1	0000000101
A	65	41	8	N	1	0100000101
B	66	42	8	N	1	0010000101
C	67	43	8	N	1	0110000101
D	68	44	8	N	1	0001000101
E	69	45	8	N	1	0101000101
F	70	46	8	N	1	0011000101
G	71	47	8	N	1	0111000101
H	72	48	8	N	1	0000100101
I	73	49	8	N	1	0100100101
J	74	4A	8	N	1	0010100101
K	75	4B	8	N	1	0110100101
L	76	4C	8	N	1	0001100101
M	77	4D	8	N	1	0101100101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678s
N	78	4E	8	N	1	0011100101
O	79	4F	8	N	1	0111100101
P	80	50	8	N	1	0000010101
Q	81	51	8	N	1	0100010101
R	82	52	8	N	1	0010010101
S	83	53	8	N	1	0110010101
T	84	54	8	N	1	0001010101
U	85	55	8	N	1	0101010101
V	86	56	8	N	1	0011010101
W	87	57	8	N	1	0111010101
X	88	58	8	N	1	0000110101
Y	89	59	8	N	1	0100110101
Z	90	5A	8	N	1	0010110101
[	91	5B	8	N	1	0110110101
\	92	5C	8	N	1	0001110101
]	93	5D	8	N	1	0101110101
^	94	5E	8	N	1	0011110101
_	95	5F	8	N	1	0111110101
`	96	60	8	N	1	0000001101
a	97	61	8	N	1	0100001101
b	98	62	8	N	1	0010001101
c	99	63	8	N	1	0110001101
d	100	64	8	N	1	0001001101
e	101	65	8	N	1	0101001101
f	102	66	8	N	1	0011001101
g	103	67	8	N	1	0111001101
h	104	68	8	N	1	0000101101
i	105	69	8	N	1	0100101101
j	106	6A	8	N	1	0010101101
k	107	6B	8	N	1	0110101101
l	108	6C	8	N	1	0001101101
m	109	6D	8	N	1	0101101101
n	110	6E	8	N	1	0011101101
o	111	6F	8	N	1	0111101101
p	112	70	8	N	1	0000011101
q	113	71	8	N	1	0100011101
r	114	72	8	N	1	0010011101
s	115	73	8	N	1	0110011101
t	116	74	8	N	1	0001011101
u	117	75	8	N	1	0101011101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678s
v	118	76	8	N	1	0011011101
w	119	77	8	N	1	0111011101
x	120	78	8	N	1	0000111101
y	121	79	8	N	1	0100111101
z	122	7A	8	N	1	0010111101
{	123	7B	8	N	1	0110111101
|	124	7C	8	N	1	0001111101
}	125	7D	8	N	1	0101111101
~	126	7E	8	N	1	0011111101
DEL	127	7F	8	N	1	0111111101
*	128	00	8	N	1	0000000011
*	129	01	8	N	1	0100000011
*	130	02	8	N	1	0010000011
*	131	03	8	N	1	0110000011
*	132	04	8	N	1	0001000011
*	133	05	8	N	1	0101000011
*	134	06	8	N	1	0011000011
*	135	07	8	N	1	0111000011
*	136	08	8	N	1	0000100011
*	137	09	8	N	1	0100100011
*	138	0A	8	N	1	0010100011
*	139	0B	8	N	1	0110100011
*	140	0C	8	N	1	0001100011
*	141	0D	8	N	1	0101100011
*	142	0E	8	N	1	0011100011
*	143	0F	8	N	1	0111100011
*	144	10	8	N	1	0000010011
*	145	11	8	N	1	0100010011
*	146	12	8	N	1	0010010011
*	147	13	8	N	1	0110010011
*	148	14	8	N	1	0001010011
*	149	15	8	N	1	0101010011
*	150	16	8	N	1	0011010011
*	151	17	8	N	1	0111010011
*	152	18	8	N	1	0000110011
*	153	19	8	N	1	0100110011
*	154	1A	8	N	1	0010110011
*	155	1B	8	N	1	0110110011
*	156	1C	8	N	1	0001110011
*	157	1D	8	N	1	0101110011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678s
*	158	1E	8	N	1	0011110011
*	159	1F	8	N	1	0111110011
*	160	20	8	N	1	0000001011
*	161	21	8	N	1	0100001011
*	162	22	8	N	1	0010001011
*	163	23	8	N	1	0110001011
*	164	24	8	N	1	0001001011
*	165	25	8	N	1	0101001011
*	166	26	8	N	1	0011001011
*	167	27	8	N	1	0111001011
*	168	28	8	N	1	0000101011
*	169	29	8	N	1	0100101011
*	170	2A	8	N	1	0010101011
*	171	2B	8	N	1	0110101011
*	172	2C	8	N	1	0001101011
*	173	2D	8	N	1	0101101011
*	174	2E	8	N	1	0011101011
*	175	2F	8	N	1	0111101011
*	176	30	8	N	1	0000011011
*	177	31	8	N	1	0100011011
*	178	32	8	N	1	0010011011
*	179	33	8	N	1	0110011011
*	180	34	8	N	1	0001011011
*	181	35	8	N	1	0101011011
*	182	36	8	N	1	0011011011
*	183	37	8	N	1	0111011011
*	184	38	8	N	1	0000111011
*	185	39	8	N	1	0100111011
*	186	3A	8	N	1	0010111011
*	187	3B	8	N	1	0110111011
*	188	3C	8	N	1	0001111011
*	189	3D	8	N	1	0101111011
*	190	3E	8	N	1	0011111011
*	191	3F	8	N	1	0111111011
*	192	40	8	N	1	0000000111
*	193	41	8	N	1	0100000111
*	194	42	8	N	1	0010000111
*	195	43	8	N	1	0110000111
*	196	44	8	N	1	0001000111
*	197	45	8	N	1	0101000111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678s
*	198	46	8	N	1	0011000111
*	199	47	8	N	1	0111000111
*	200	48	8	N	1	0000100111
*	201	49	8	N	1	0100100111
*	202	4A	8	N	1	0010100111
*	203	4B	8	N	1	0110100111
*	204	4C	8	N	1	0001100111
*	205	4D	8	N	1	0101100111
*	206	4E	8	N	1	0011100111
*	207	4F	8	N	1	0111100111
*	208	50	8	N	1	0000010111
*	209	51	8	N	1	0100010111
*	210	52	8	N	1	0010010111
*	211	53	8	N	1	0110010111
*	212	54	8	N	1	0001010111
*	213	55	8	N	1	0101010111
*	214	56	8	N	1	0011010111
*	215	57	8	N	1	0111010111
*	216	58	8	N	1	0000110111
*	217	59	8	N	1	0100110111
*	218	5A	8	N	1	0010110111
*	219	5B	8	N	1	0110110111
*	220	5C	8	N	1	0001110111
*	221	5D	8	N	1	0101110111
*	222	5E	8	N	1	0011110111
*	223	5F	8	N	1	0111110111
*	224	60	8	N	1	0000001111
*	225	61	8	N	1	0100001111
*	226	62	8	N	1	0010001111
*	227	63	8	N	1	0110001111
*	228	64	8	N	1	0001001111
*	229	65	8	N	1	0101001111
*	230	66	8	N	1	0011001111
*	231	67	8	N	1	0111001111
*	232	68	8	N	1	0000101111
*	233	69	8	N	1	0100101111
*	234	6A	8	N	1	0010101111
*	235	6B	8	N	1	0110101111
*	236	6C	8	N	1	0001101111
*	237	6D	8	N	1	0101101111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678s
*	238	6E	8	N	1	0011101111
*	239	6F	8	N	1	0111101111
*	240	70	8	N	1	0000011111
*	241	71	8	N	1	0100011111
*	242	72	8	N	1	0010011111
*	243	73	8	N	1	0110011111
*	244	74	8	N	1	0001011111
*	245	75	8	N	1	0101011111
*	246	76	8	N	1	0011011111
*	247	77	8	N	1	0111011111
*	248	78	8	N	1	0000111111
*	249	79	8	N	1	0100111111
*	250	7A	8	N	1	0010111111
*	251	7B	8	N	1	0110111111
*	252	7C	8	N	1	0001111111
*	253	7D	8	N	1	0101111111
*	254	7E	8	N	1	0011111111
*	255	7F	8	N	1	0111111111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
NULL	0	00	8	E	1	00000000001
SOH	1	01	8	E	1	01000000011
STX	2	02	8	E	1	00100000011
ETX	3	03	8	E	1	01100000001
EOT	4	04	8	E	1	00010000011
ENQ	5	05	8	E	1	01010000001
ACK	6	06	8	E	1	00110000001
BEL	7	07	8	E	1	01110000011
BS	8	08	8	E	1	00001000011
HT	9	09	8	E	1	01001000001
LF	10	0A	8	E	1	00101000001
VT	11	0B	8	E	1	01101000011
FF	12	0C	8	E	1	00011000001
CR	13	0D	8	E	1	01011000011
SO	14	0E	8	E	1	00111000011
SI	15	0F	8	E	1	01111000001
DLE	16	10	8	E	1	00000100011
DC1	17	11	8	E	1	01000100001
DC2	18	12	8	E	1	00100100001
DC3	19	13	8	E	1	01100100011
DC4	20	14	8	E	1	00010100001
NAK	21	15	8	E	1	01010100011
SYN	22	16	8	E	1	00110100011
ETB	23	17	8	E	1	01110100001
CAN	24	18	8	E	1	00001100001
EM	25	19	8	E	1	01001100011
SUB	26	1A	8	E	1	00101100011
ESC	27	1B	8	E	1	01101100001
FS	28	1C	8	E	1	00011100011
GS	29	1D	8	E	1	01011100001
RS	30	1E	8	E	1	00111100001
US	31	1F	8	E	1	01111100011
SP	32	20	8	E	1	00000010011
!	33	21	8	E	1	01000010001
"	34	22	8	E	1	00100010001
£	35	23	8	E	1	01100010011
$	36	24	8	E	1	00010010001
%	37	25	8	E	1	01010010011
&	38	26	8	E	1	00110010011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
'	39	27	8	E	1	01110010001
(	40	28	8	E	1	00001010001
)	41	29	8	E	1	01001010011
*	42	2A	8	E	1	00101010011
+	43	2B	8	E	1	01101010001
,	44	2C	8	E	1	00011010011
-	45	2D	8	E	1	01011010001
.	46	2E	8	E	1	00111010001
/	47	2F	8	E	1	01111010011
0	48	30	8	E	1	00000110001
1	49	31	8	E	1	01000110011
2	50	32	8	E	1	00100110011
3	51	33	8	E	1	01100110001
4	52	34	8	E	1	00010110011
5	53	35	8	E	1	01010110001
6	54	36	8	E	1	00110110001
7	55	37	8	E	1	01110110011
8	56	38	8	E	1	00001110011
9	57	39	8	E	1	01001110001
:	58	3A	8	E	1	00101110001
;	59	3B	8	E	1	01101110011
<	60	3C	8	E	1	00011110001
=	61	3D	8	E	1	01011110011
	62	3E	8	E	1	00111110011
?	63	3F	8	E	1	01111110001
@	64	40	8	E	1	00000001011
A	65	41	8	E	1	01000001001
B	66	42	8	E	1	00100001001
C	67	43	8	E	1	01100001011
D	68	44	8	E	1	00010001001
E	69	45	8	E	1	01010001011
F	70	46	8	E	1	00110001011
G	71	47	8	E	1	01110001001
H	72	48	8	E	1	00001001001
I	73	49	8	E	1	01001001011
J	74	4A	8	E	1	00101001011
K	75	4B	8	E	1	01101001001
L	76	4C	8	E	1	00011001011
M	77	4D	8	E	1	01011001001
N	78	4E	8	E	1	00111001001

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
O	79	4F	8	E	1	01111001011
P	80	50	8	E	1	00000101001
Q	81	51	8	E	1	01000101011
R	82	52	8	E	1	00100101011
S	83	53	8	E	1	01100101001
T	84	54	8	E	1	00010101011
U	85	55	8	E	1	01010101001
V	86	56	8	E	1	00110101001
W	87	57	8	E	1	01110101011
X	88	58	8	E	1	00001101011
Y	89	59	8	E	1	01001101001
Z	90	5A	8	E	1	00101101001
[	91	5B	8	E	1	01101101011
\	92	5C	8	E	1	00011101001
]	93	5D	8	E	1	01011101011
^	94	5E	8	E	1	00111101011
_	95	5F	8	E	1	01111101001
`	96	60	8	E	1	00000011001
a	97	61	8	E	1	01000011011
b	98	62	8	E	1	00100011011
c	99	63	8	E	1	01100011001
d	100	64	8	E	1	00010011011
e	101	65	8	E	1	01010011001
f	102	66	8	E	1	00110011001
g	103	67	8	E	1	01110011011
h	104	68	8	E	1	00001011011
i	105	69	8	E	1	01001011001
j	106	6A	8	E	1	00101011001
k	107	6B	8	E	1	01101011011
l	108	6C	8	E	1	00011011001
m	109	6D	8	E	1	01011011011
n	110	6E	8	E	1	00111011011
o	111	6F	8	E	1	01111011001
p	112	70	8	E	1	00000111011
q	113	71	8	E	1	01000111001
r	114	72	8	E	1	00100111001
s	115	73	8	E	1	01100111011
t	116	74	8	E	1	00010111001
u	117	75	8	E	1	01010111011
v	118	76	8	E	1	00110111011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
w	119	77	8	E	1	01110111001
x	120	78	8	E	1	00001111001
y	121	79	8	E	1	01001111011
z	122	7A	8	E	1	00101111011
{	123	7B	8	E	1	01101111001
|	124	7C	8	E	1	00011111011
}	125	7D	8	E	1	01011111001
~	126	7E	8	E	1	00111111001
DEL	127	7F	8	E	1	01111111011
*	128	00	8	E	1	00000000111
*	129	01	8	E	1	01000000101
*	130	02	8	E	1	00100000101
*	131	03	8	E	1	01100000111
*	132	04	8	E	1	00010000101
*	133	05	8	E	1	01010000111
*	134	06	8	E	1	00110000111
*	135	07	8	E	1	01110000101
*	136	08	8	E	1	00001000101
*	137	09	8	E	1	01001000111
*	138	0A	8	E	1	00101000111
*	139	0B	8	E	1	01101000101
*	140	0C	8	E	1	00011000111
*	141	0D	8	E	1	01011000101
*	142	0E	8	E	1	00111000101
*	143	0F	8	E	1	01111000111
*	144	10	8	E	1	00000100101
*	145	11	8	E	1	01000100111
*	146	12	8	E	1	00100100111
*	147	13	8	E	1	01100100101
*	148	14	8	E	1	00010100111
*	149	15	8	E	1	01010100101
*	150	16	8	E	1	00110100101
*	151	17	8	E	1	01110100111
*	152	18	8	E	1	00001100111
*	153	19	8	E	1	01001100101
*	154	1A	8	E	1	00101100101
*	155	1B	8	E	1	01101100111
*	156	1C	8	E	1	00011100101
*	157	1D	8	E	1	01011100111
*	158	1E	8	E	1	00111100111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	159	1F	8	E	1	01111100101
*	160	20	8	E	1	00000010101
*	161	21	8	E	1	01000010111
*	162	22	8	E	1	00100010111
*	163	23	8	E	1	01100010101
*	164	24	8	E	1	00010010111
*	165	25	8	E	1	01010010101
*	166	26	8	E	1	00110010101
*	167	27	8	E	1	01110010111
*	168	28	8	E	1	00001010111
*	169	29	8	E	1	01001010101
*	170	2A	8	E	1	00101010101
*	171	2B	8	E	1	01101010111
*	172	2C	8	E	1	00011010101
*	173	2D	8	E	1	01011010111
*	174	2E	8	E	1	00111010111
*	175	2F	8	E	1	01111010101
*	176	30	8	E	1	00000110111
*	177	31	8	E	1	01000110101
*	178	32	8	E	1	00100110101
*	179	33	8	E	1	01100110111
*	180	34	8	E	1	00010110101
*	181	35	8	E	1	01010110111
*	182	36	8	E	1	00110110111
*	183	37	8	E	1	01110110101
*	184	38	8	E	1	00001110101
*	185	39	8	E	1	01001110111
*	186	3A	8	E	1	00101110111
*	187	3B	8	E	1	01101110101
*	188	3C	8	E	1	00011110111
*	189	3D	8	E	1	01011110101
*	190	3E	8	E	1	00111110101
*	191	3F	8	E	1	01111110111
*	192	40	8	E	1	00000001101
*	193	41	8	E	1	01000001111
*	194	42	8	E	1	00100001111
*	195	43	8	E	1	01100001101
*	196	44	8	E	1	00010001111
*	197	45	8	E	1	01010001101
*	198	46	8	E	1	00110001101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	199	47	8	E	1	01110001111
*	200	48	8	E	1	00001001111
*	201	49	8	E	1	01001001101
*	202	4A	8	E	1	00101001101
*	203	4B	8	E	1	01101001111
*	204	4C	8	E	1	00011001101
*	205	4D	8	E	1	01011001111
*	206	4E	8	E	1	00111001111
*	207	4F	8	E	1	01111001101
*	208	50	8	E	1	00000101111
*	209	51	8	E	1	01000101101
*	210	52	8	E	1	00100101101
*	211	53	8	E	1	01100101111
*	212	54	8	E	1	00010101101
*	213	55	8	E	1	01010101111
*	214	56	8	E	1	00110101111
*	215	57	8	E	1	01110101101
*	216	58	8	E	1	00001101101
*	217	59	8	E	1	01001101111
*	218	5A	8	E	1	00101101111
*	219	5B	8	E	1	01101101101
*	220	5C	8	E	1	00011101111
*	221	5D	8	E	1	01011101101
*	222	5E	8	E	1	00111101101
*	223	5F	8	E	1	01111101111
*	224	60	8	E	1	00000011111
*	225	61	8	E	1	01000011101
*	226	62	8	E	1	00100011101
*	227	63	8	E	1	01100011111
*	228	64	8	E	1	00010011101
*	229	65	8	E	1	01010011111
*	230	66	8	E	1	00110011111
*	231	67	8	E	1	01110011101
*	232	68	8	E	1	00001011101
*	233	69	8	E	1	01001011111
*	234	6A	8	E	1	00101011111
*	235	6B	8	E	1	01101011101
*	236	6C	8	E	1	00011011111
*	237	6D	8	E	1	01011011101
*	238	6E	8	E	1	00111011101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	239	6F	8	E	1	01111011111
*	240	70	8	E	1	00000111101
*	241	71	8	E	1	01000111111
*	242	72	8	E	1	00100111111
*	243	73	8	E	1	01100111101
*	244	74	8	E	1	00010111111
*	245	75	8	E	1	01010111101
*	246	76	8	E	1	00110111101
*	247	77	8	E	1	01110111111
*	248	78	8	E	1	00001111111
*	249	79	8	E	1	01001111101
*	250	7A	8	E	1	00101111101
*	251	7B	8	E	1	01101111111
*	252	7C	8	E	1	00011111101
*	253	7D	8	E	1	01011111111
*	254	7E	8	E	1	00111111111
*	255	7F	8	E	1	01111111101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
NULL	0	00	8	O	1	00000000011
SOH	1	01	8	O	1	01000000001
STX	2	02	8	O	1	00100000001
ETX	3	03	8	O	1	01100000011
EOT	4	04	8	O	1	00010000001
ENQ	5	05	8	O	1	01010000011
ACK	6	06	8	O	1	00110000011
BEL	7	07	8	O	1	01110000001
BS	8	08	8	O	1	00001000001
HT	9	09	8	O	1	01001000011
LF	10	0A	8	O	1	00101000011
VT	11	0B	8	O	1	01101000001
FF	12	0C	8	O	1	00011000011
CR	13	0D	8	O	1	01011000001
SO	14	0E	8	O	1	00111000001
SI	15	0F	8	O	1	01111000011
DLE	16	10	8	O	1	00000100001
DC1	17	11	8	O	1	01000100011
DC2	18	12	8	O	1	00100100011
DC3	19	13	8	O	1	01100100001
DC4	20	14	8	O	1	00010100011
NAK	21	15	8	O	1	01010100001
SYN	22	16	8	O	1	00110100001
ETB	23	17	8	O	1	01110100011
CAN	24	18	8	O	1	00001100011
EM	25	19	8	O	1	01001100001
SUB	26	1A	8	O	1	00101100001
ESC	27	1B	8	O	1	01101100011
FS	28	1C	8	O	1	00011100001
GS	29	1D	8	O	1	01011100011
RS	30	1E	8	O	1	00111100011
US	31	1F	8	O	1	01111100001
SP	32	20	8	O	1	00000010001
!	33	21	8	O	1	01000010011
"	34	22	8	O	1	00100010011
£	35	23	8	O	1	01100010001
$	36	24	8	O	1	00010010011
%	37	25	8	O	1	01010010001
&	38	26	8	O	1	00110010001
'	39	27	8	O	1	01110010011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
(	40	28	8	O	1	00001010011
)	41	29	8	O	1	01001010001
*	42	2A	8	O	1	00101010001
+	43	2B	8	O	1	01101010011
,	44	2C	8	O	1	00011010001
-	45	2D	8	O	1	01011010011
.	46	2E	8	O	1	00111010011
/	47	2F	8	O	1	01111010001
0	48	30	8	O	1	00000110011
1	49	31	8	O	1	01000110001
2	50	32	8	O	1	00100110001
3	51	33	8	O	1	01100110011
4	52	34	8	O	1	00010110001
5	53	35	8	O	1	01010110011
6	54	36	8	O	1	00110110011
7	55	37	8	O	1	01110110001
8	56	38	8	O	1	00001110001
9	57	39	8	O	1	01001110011
:	58	3A	8	O	1	00101110011
;	59	3B	8	O	1	01101110001
<	60	3C	8	O	1	00011110011
=	61	3D	8	O	1	01011110001
	62	3E	8	O	1	00111110001
?	63	3F	8	O	1	01111110011
@	64	40	8	0	1	00000001001
A	65	41	8	O	1	01000001011
B	66	42	8	O	1	00100001011
C	67	43	8	O	1	01100001001
D	68	44	8	O	1	00010001011
E	69	45	8	O	1	01010001001
F	70	46	8	O	1	00110001001
G	71	47	8	O	1	01110001011
H	72	48	8	O	1	00001001011
I	73	49	8	O	1	01001001001
J	74	4A	8	O	1	00101001001
K	75	4B	8	O	1	01101001011
L	76	4C	8	O	1	00011001001
M	77	4D	8	O	1	01011001011
N	78	4E	8	O	1	00111001011
O	79	4F	8	O	1	01111001001

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
P	80	50	8	O	1	00000101011
Q	81	51	8	O	1	01000101001
R	82	52	8	O	1	00100101001
S	83	53	8	O	1	01100101011
T	84	54	8	O	1	00010101001
U	85	55	8	O	1	01010101011
V	86	56	8	O	1	00110101011
W	87	57	8	O	1	01110101001
X	88	58	8	O	1	00001101001
Y	89	59	8	O	1	01001101011
Z	90	5A	8	O	1	00101101011
[	91	5B	8	O	1	01101101001
\	92	5C	8	O	1	00011101011
]	93	5D	8	O	1	01011101001
^	94	5E	8	O	1	00111101001
_	95	5F	8	O	1	01111101011
`	96	60	8	O	1	00000011011
a	97	61	8	O	1	01000011001
b	98	62	8	O	1	00100011001
c	99	63	8	O	1	01100011011
d	100	64	8	O	1	00010011001
e	101	65	8	O	1	01010011011
f	102	66	8	O	1	00110011011
g	103	67	8	O	1	01110011001
h	104	68	8	O	1	00001011001
i	105	69	8	O	1	01001011011
j	106	6A	8	O	1	00101011011
k	107	6B	8	O	1	01101011001
l	108	6C	8	O	1	00011011011
m	109	6D	8	O	1	01011011001
n	110	6E	8	O	1	00111011001
o	111	6F	8	O	1	01111011011
p	112	70	8	O	1	00000111001
q	113	71	8	O	1	01000111011
r	114	72	8	O	1	00100111011
s	115	73	8	O	1	01100111001
t	116	74	8	O	1	00010111011
u	117	75	8	O	1	01010111001
v	118	76	8	O	1	00110111001
w	119	77	8	O	1	01110111011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
x	120	78	8	O	1	00001111011
y	121	79	8	O	1	01001111001
z	122	7A	8	O	1	00101111001
{	123	7B	8	O	1	01101111011
|	124	7C	8	O	1	00011111001
}	125	7D	8	O	1	01011111011
~	126	7E	8	O	1	00111111011
DEL	127	7F	8	O	1	01111111001
*	128	00	8	O	1	00000000101
*	129	01	8	O	1	01000000111
*	130	02	8	O	1	00100000111
*	131	03	8	O	1	01100000101
*	132	04	8	O	1	00010000111
*	133	05	8	O	1	01010000101
*	134	06	8	O	1	00110000101
*	135	07	8	O	1	01110000111
*	136	08	8	O	1	00001000111
*	137	09	8	O	1	01001000101
*	138	0A	8	O	1	00101000101
*	139	0B	8	O	1	01101000111
*	140	0C	8	O	1	00011000101
*	141	0D	8	O	1	01011000111
*	142	0E	8	O	1	00111000111
*	143	0F	8	O	1	01111000101
*	144	10	8	O	1	00000100111
*	145	11	8	O	1	01000100101
*	146	12	8	O	1	00100100101
*	147	13	8	O	1	01100100111
*	148	14	8	O	1	00010100101
*	149	15	8	O	1	01010100111
*	150	16	8	O	1	00110100111
*	151	17	8	O	1	01110100101
*	152	18	8	O	1	00001100101
*	153	19	8	O	1	01001100111
*	154	1A	8	O	1	00101100111
*	155	1B	8	O	1	01101100101
*	156	1C	8	O	1	00011100111
*	157	1D	8	O	1	01011100101
*	158	1E	8	O	1	00111100101
*	159	1F	8	O	1	01111100111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	160	20	8	O	1	00000010111
*	161	21	8	O	1	01000010101
*	162	22	8	O	1	00100010101
*	163	23	8	O	1	01100010111
*	164	24	8	O	1	00010010101
*	165	25	8	O	1	01010010111
*	166	26	8	O	1	00110010111
*	167	27	8	O	1	01110010101
*	168	28	8	O	1	00001010101
*	169	29	8	O	1	01001010111
*	170	2A	8	O	1	00101010111
*	171	2B	8	O	1	01101010101
*	172	2C	8	O	1	00011010111
*	173	2D	8	O	1	01011010101
*	174	2E	8	O	1	00111010101
*	175	2F	8	O	1	01111010111
*	176	30	8	O	1	00000110101
*	177	31	8	O	1	01000110111
*	178	32	8	O	1	00100110111
*	179	33	8	O	1	01100110101
*	180	34	8	O	1	00010110111
*	181	35	8	O	1	01010110101
*	182	36	8	O	1	00110110101
*	183	37	8	O	1	01110110111
*	184	38	8	O	1	00001110111
*	185	39	8	O	1	01001110101
*	186	3A	8	O	1	00101110101
*	187	3B	8	O	1	01101110111
*	188	3C	8	O	1	00011110101
*	189	3D	8	O	1	01011110111
*	190	3E	8	O	1	00111110111
*	191	3F	8	O	1	01111110101
*	192	40	8	O	1	00000001111
*	193	41	8	O	1	01000001101
*	194	42	8	O	1	00100001101
*	195	43	8	O	1	01100001111
*	196	44	8	O	1	00010001101
*	197	45	8	O	1	01010001111
*	198	46	8	O	1	00110001111
*	199	47	8	O	1	01110001101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	200	48	8	O	1	00001001101
*	201	49	8	O	1	01001001111
*	202	4A	8	O	1	00101001111
*	203	4B	8	O	1	01101001101
*	204	4C	8	O	1	00011001111
*	205	4D	8	O	1	01011001101
*	206	4E	8	O	1	00111001101
*	207	4F	8	O	1	01111001111
*	208	50	8	O	1	00000101101
*	209	51	8	O	1	01000101111
*	210	52	8	O	1	00100101111
*	211	53	8	O	1	01100101101
*	212	54	8	O	1	00010101111
*	213	55	8	O	1	01010101101
*	214	56	8	O	1	00110101101
*	215	57	8	O	1	01110101111
*	216	58	8	O	1	00001101111
*	217	59	8	O	1	01001101101
*	218	5A	8	O	1	00101101101
*	219	5B	8	O	1	01101101111
*	220	5C	8	O	1	00011101101
*	221	5D	8	O	1	01011101111
*	222	5E	8	O	1	00111101111
*	223	5F	8	O	1	01111101101
*	224	60	8	O	1	00000011101
*	225	61	8	O	1	01000011111
*	226	62	8	O	1	00100011111
*	227	63	8	O	1	01100011101
*	228	64	8	O	1	00010011111
*	299	65	8	O	1	01010011101
*	230	66	8	O	1	00110011101
*	231	67	8	O	1	01110011111
*	232	68	8	O	1	00001011111
*	233	69	8	O	1	01001011101
*	234	6A	8	O	1	00101011101
*	235	6B	8	O	1	01101011111
*	236	6C	8	0	1	00011011101
*	237	6D	8	O	1	01011011111
*	238	6E	8	O	1	00111011111
*	239	6F	8	O	1	01111011101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	240	70	8	O	1	00000111111
*	241	71	8	O	1	01000111101
*	242	72	8	O	1	00100111101
*	243	73	8	O	1	01100111111
*	244	74	8	O	1	00010111101
*	245	75	8	O	1	01010111111
*	246	76	8	O	1	00110111111
*	247	77	8	O	1	01110111101
*	248	78	8	O	1	00001111101
*	249	79	8	O	1	01001111111
*	250	7A	8	O	1	00101111111
*	251	7B	8	O	1	01101111101
*	252	7C	8	O	1	00011111111
*	253	7D	8	O	1	01011111101
*	254	7E	8	O	1	00111111101
*	255	7F	8	O	1	01111111111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
NULL	0	00	8	M	1	00000000011
SOH	1	01	8	M	1	01000000011
STX	2	02	8	M	1	00100000011
ETX	3	03	8	M	1	01100000011
EOT	4	04	8	M	1	00010000011
ENQ	5	05	8	M	1	01010000011
ACK	6	06	8	M	1	00110000011
BEL	7	07	8	M	1	01110000011
BS	8	08	8	M	1	00001000011
HT	9	09	8	M	1	01001000011
LF	10	0A	8	M	1	00101000011
VT	11	0B	8	M	1	01101000011
FF	12	0C	8	M	1	00011000011
CR	13	0D	8	M	1	01011000011
SO	14	0E	8	M	1	00111000011
SI	15	0F	8	M	1	01111000011
DLE	16	10	8	M	1	00000100011
DC1	17	11	8	M	1	01000100011
DC2	18	12	8	M	1	00100100011
DC3	19	13	8	M	1	01100100011
DC4	20	14	8	M	1	00010100011
NAK	21	15	8	M	1	01010100011
SYN	22	16	8	M	1	00110100011
ETB	23	17	8	M	1	01110100011
CAN	24	18	8	M	1	00001100011
EM	25	19	8	M	1	01001100011
SUB	26	1A	8	M	1	00101100011
ESC	27	1B	8	M	1	01101100011
FS	28	1C	8	M	1	00011100011
GS	29	1D	8	M	1	01011100011
RS	30	1E	8	M	1	00111100011
US	31	1F	8	M	1	01111100011
SP	32	20	8	M	1	00000010011
!	33	21	8	M	1	01000010011
"	34	22	8	M	1	00100010011
£	35	23	8	M	1	01100010011
$	36	24	8	M	1	00010010011
%	37	25	8	M	1	01010010011
&	38	26	8	M	1	00110010011
'	39	27	8	M	1	01110010011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
(	40	28	8	M	1	00001010011
)	41	29	8	M	1	01001010011
*	42	2A	8	M	1	00101010011
+	43	2B	8	M	1	01101010011
,	44	2C	8	M	1	00011010011
-	45	2D	8	M	1	01011010011
.	46	2E	8	M	1	00111010011
/	47	2F	8	M	1	01111010011
0	48	30	8	M	1	00000110011
1	49	31	8	M	1	01000110011
2	50	32	8	M	1	00100110011
3	51	33	8	M	1	01100110011
4	52	34	8	M	1	00010110011
5	53	35	8	M	1	01010110011
6	54	36	8	M	1	00110110011
7	55	37	8	M	1	01110110011
8	56	38	8	M	1	00001110011
9	57	39	8	M	1	01001110011
:	58	3A	8	M	1	00101110011
;	59	3B	8	M	1	01101110011
<	60	3C	8	M	1	00011110011
=	61	3D	8	M	1	01011110011
	62	3E	8	M	1	00111110011
?	63	3F	8	M	1	01111110011
@	64	40	8	M	1	00000001011
A	65	41	8	M	1	01000001011
B	66	42	8	M	1	00100001011
C	67	43	8	M	1	01100001011
D	68	44	8	M	1	00010001011
E	69	45	8	M	1	01010001011
F	70	46	8	M	1	00110001011
G	71	47	8	M	1	01110001011
H	72	48	8	M	1	00001001011
I	73	49	8	M	1	01001001011
J	74	4A	8	M	1	00101001011
K	75	4B	8	M	1	01101001011
L	76	4C	8	M	1	00011001011
M	77	4D	8	M	1	01011001011
N	78	4E	8	M	1	00111001011
O	79	4F	8	M	1	01111001011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
P	80	50	8	M	1	00000101011
Q	81	51	8	M	1	01000101011
R	82	52	8	M	1	00100101011
S	83	53	8	M	1	01100101011
T	84	54	8	M	1	00010101011
U	85	55	8	M	1	01010101011
V	86	56	8	M	1	00110101011
W	87	57	8	M	1	01110101011
X	88	58	8	M	1	00001101011
Y	89	59	8	M	1	01001101011
Z	90	5A	8	M	1	00101101011
[	91	5B	8	M	1	01101101011
\	92	5C	8	M	1	00011101011
]	93	5D	8	M	1	01011101011
^	94	5E	8	M	1	00111101011
_	95	5F	8	M	1	01111101011
`	96	60	8	M	1	00000011011
a	97	61	8	M	1	01000011011
b	98	62	8	M	1	00100011011
c	99	63	8	M	1	01100011011
d	100	64	8	M	1	00010011011
e	101	65	8	M	1	01010011011
f	102	66	8	M	1	00110011011
g	103	67	8	M	1	01110011011
h	104	68	8	M	1	00001011011
i	105	69	8	M	1	01001011011
j	106	6A	8	M	1	00101011011
k	107	6B	8	M	1	01101011011
l	108	6C	8	M	1	00011011011
m	109	6D	8	M	1	01011011011
n	110	6E	8	M	1	00111011011
o	111	6F	8	M	1	01111011011
p	112	70	8	M	1	00000111011
q	113	71	8	M	1	01000111011
r	114	72	8	M	1	00100111011
s	115	73	8	M	1	01100111011
t	116	74	8	M	1	00010111011
u	117	75	8	M	1	01010111011
v	118	76	8	M	1	00110111011
w	119	77	8	M	1	01110111011

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
x	120	78	8	M	1	00001111011
y	121	79	8	M	1	01001111011
z	122	7A	8	M	1	00101111011
{	123	7B	8	M	1	01101111011
|	124	7C	8	M	1	00011111011
}	125	7D	8	M	1	01011111011
~	126	7E	8	M	1	00111111011
DEL	127	7F	8	M	1	01111111011
*	128	00	8	M	1	00000000111
*	129	01	8	M	1	01000000111
*	130	02	8	M	1	00100000111
*	131	03	8	M	1	01100000111
*	132	04	8	M	1	00010000111
*	133	05	8	M	1	01010000111
*	134	06	8	M	1	00110000111
*	135	07	8	M	1	01110000111
*	136	08	8	M	1	00001000111
*	137	09	8	M	1	01001000111
*	138	0A	8	M	1	00101000111
*	139	0B	8	M	1	01101000111
*	140	0C	8	M	1	00011000111
*	141	0D	8	M	1	01011000111
*	142	0E	8	M	1	00111000111
*	143	0F	8	M	1	01111000111
*	144	10	8	M	1	00000100111
*	145	11	8	M	1	01000100111
*	146	12	8	M	1	00100100111
*	147	13	8	M	1	01100100111
*	148	14	8	M	1	00010100111
*	149	15	8	M	1	01010100111
*	150	16	8	M	1	00110100111
*	151	17	8	M	1	01110100111
*	152	18	8	M	1	00001100111
*	153	19	8	M	1	01001100111
*	154	1A	8	M	1	00101100111
*	155	1B	8	M	1	01101100111
*	156	1C	8	M	1	00011100111
*	157	1D	8	M	1	01011100111
*	158	1E	8	M	1	00111100111
*	159	1F	8	M	1	01111100111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	160	20	8	M	1	00000010111
*	161	21	8	M	1	01000010111
*	162	22	8	M	1	00100010111
*	163	23	8	M	1	01100010111
*	164	24	8	M	1	00010010111
*	165	25	8	M	1	01010010111
*	166	26	8	M	1	00110010111
*	167	27	8	M	1	01110010111
*	168	28	8	M	1	00001010111
*	169	29	8	M	1	01001010111
*	170	2A	8	M	1	00101010111
*	171	2B	8	M	1	01101010111
*	172	2C	8	M	1	00011010111
*	173	2D	8	M	1	01011010111
*	174	2E	8	M	1	00111010111
*	175	2F	8	M	1	01111010111
*	176	30	8	M	1	00000110111
*	177	31	8	M	1	01000110111
*	178	32	8	M	1	00100110111
*	179	33	8	M	1	01100110111
*	180	34	8	M	1	00010110111
*	181	35	8	M	1	01010110111
*	182	36	8	M	1	00110110111
*	183	37	8	M	1	01110110111
*	184	38	8	M	1	00001110111
*	185	39	8	M	1	01001110111
*	186	3A	8	M	1	00101110111
*	187	3B	8	M	1	01101110111
*	188	3C	8	M	1	00011110111
*	189	3D	8	M	1	01011110111
*	190	3E	8	M	1	00111110111
*	191	3F	8	M	1	01111110111
*	192	40	8	M	1	00000001111
*	193	41	8	M	1	01000001111
*	194	42	8	M	1	00100001111
*	195	43	8	M	1	01100001111
*	196	44	8	M	1	00010001111
*	197	45	8	M	1	01010001111
*	198	46	8	M	1	00110001111
*	199	47	8	M	1	01110001111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	200	48	8	M	1	00001001111
*	201	49	8	M	1	01001001111
*	202	4A	8	M	1	00101001111
*	203	4B	8	M	1	01101001111
*	204	4C	8	M	1	00011001111
*	205	4D	8	M	1	01011001111
*	206	4E	8	M	1	00111001111
*	207	4F	8	M	1	01111001111
*	208	50	8	M	1	00000101111
*	209	51	8	M	1	01000101111
*	210	52	8	M	1	00100101111
*	211	53	8	M	1	01100101111
*	212	54	8	M	1	00010101111
*	213	55	8	M	1	01010101111
*	214	56	8	M	1	00110101111
*	215	57	8	M	1	01110101111
*	216	58	8	M	1	00001101111
*	217	59	8	M	1	01001101111
*	218	5A	8	M	1	00101101111
*	219	5B	8	M	1	01101101111
*	220	5C	8	M	1	00011101111
*	221	5D	8	M	1	01011101111
*	222	5E	8	M	1	00111101111
*	223	5F	8	M	1	01111101111
*	224	60	8	M	1	00000011111
*	225	61	8	M	1	01000011111
*	226	62	8	M	1	00100011111
*	227	63	8	M	1	01100011111
*	228	64	8	M	1	00010011111
*	229	65	8	M	1	01010011111
*	230	66	8	M	1	00110011111
*	231	67	8	M	1	01110011111
*	232	68	8	M	1	00001011111
*	233	69	8	M	1	01001011111
*	234	6A	8	M	1	00101011111
*	235	6B	8	M	1	01101011111
*	236	6C	8	M	1	00011011111
*	237	6D	8	M	1	01011011111
*	238	6E	8	M	1	00111011111
*	239	6F	8	M	1	01111011111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	240	70	8	M	1	00000111111
*	241	71	8	M	1	01000111111
*	242	72	8	M	1	00100111111
*	243	73	8	M	1	01100111111
*	244	74	8	M	1	00010111111
*	245	75	8	M	1	01010111111
*	246	76	8	M	1	00110111111
*	247	77	8	M	1	01110111111
*	248	78	8	M	1	00001111111
*	249	79	8	M	1	01001111111
*	250	7A	8	M	1	00101111111
*	251	7B	8	M	1	01101111111
*	252	7C	8	M	1	00011111111
*	253	7D	8	M	1	01011111111
*	254	7E	8	M	1	00111111111
*	255	7F	8	M	1	01111111111

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
NULL	0	00	8	S	1	00000000001
SOH	1	01	8	S	1	01000000001
STX	2	02	8	S	1	00100000001
ETX	3	03	8	S	1	01100000001
EOT	4	04	8	S	1	00010000001
ENQ	5	05	8	S	1	01010000001
ACK	6	06	8	S	1	00110000001
BEL	7	07	8	S	1	01110000001
BS	8	08	8	S	1	00001000001
HT	9	09	8	S	1	01001000001
LF	10	0A	8	S	1	00101000001
VT	11	0B	8	S	1	01101000001
FF	12	0C	8	S	1	00011000001
CR	13	0D	8	S	1	01011000001
SO	14	0E	8	S	1	00111000001
SI	15	0F	8	S	1	01111000001
DLE	16	10	8	S	1	00000100001
DC1	17	11	8	S	1	01000100001
DC2	18	12	8	S	1	00100100001
DC3	19	13	8	S	1	01100100001
DC4	20	14	8	S	1	00010100001
NAK	21	15	8	S	1	01010100001
SYN	22	16	8	S	1	00110100001
ETB	23	17	8	S	1	01110100001
CAN	24	18	8	S	1	00001100001
EM	25	19	8	S	1	01001100001
SUB	26	1A	8	S	1	00101100001
ESC	27	1B	8	S	1	01101100001
FS	28	1C	8	S	1	00011100001
GS	29	1D	8	S	1	01011100001
RS	30	1E	8	S	1	00111100001
US	31	1F	8	S	1	01111100001
SP	32	20	8	S	1	00000010001
!	33	21	8	S	1	01000010001
"	34	22	8	S	1	00100010001
£	35	23	8	S	1	01100010001
$	36	24	8	S	1	00010010001
%	37	25	8	S	1	01010010001
&	38	26	8	S	1	00110010001
'	39	27	8	S	1	01110010001

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
(	40	28	8	S	1	00001010001
)	41	29	8	S	1	01001010001
*	42	2A	8	S	1	00101010001
+	43	2B	8	S	1	01101010001
,	44	2C	8	S	1	00011010001
-	45	2D	8	S	1	01011010001
.	46	2E	8	S	1	00111010001
/	47	2F	8	S	1	01111010001
0	48	30	8	S	1	00000110001
1	49	31	8	S	1	01000110001
2	50	32	8	S	1	00100110001
3	51	33	8	S	1	01100110001
4	52	34	8	S	1	00010110001
5	53	35	8	S	1	01010110001
6	54	36	8	S	1	00110110001
7	55	37	8	S	1	01110110001
8	56	38	8	S	1	00001110001
9	57	39	8	S	1	01001110001
:	58	3A	8	S	1	00101110001
;	59	3B	8	S	1	01101110001
<	60	3C	8	S	1	00011110001
=	61	3D	8	S	1	01011110001
	62	3E	8	S	1	00111110001
?	63	3F	8	S	1	01111110001
@	64	40	8	S	1	00000001001
A	65	41	8	S	1	01000001001
B	66	42	8	S	1	00100001001
C	67	43	8	S	1	01100001001
D	68	44	8	S	1	00010001001
E	69	45	8	S	1	01010001001
F	70	46	8	S	1	00110001001
G	71	47	8	S	1	01110001001
H	72	48	8	S	1	00001001001
I	73	49	8	S	1	01001001001
J	74	4A	8	S	1	00101001001
K	75	4B	8	S	1	01101001001
L	76	4C	8	S	1	00011001001
M	77	4D	8	S	1	01011001001
N	78	4E	8	S	1	00111001001

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
O	79	4F	8	S	1	01111001001
P	80	50	8	S	1	00000101001
Q	81	51	8	S	1	01000101001
R	82	52	8	S	1	00100101001
S	83	53	8	S	1	01100101001
T	84	54	8	S	1	00010101001
U	85	55	8	S	1	01010101001
V	86	56	8	S	1	00110101001
W	87	57	8	S	1	01110101001
X	88	58	8	S	1	00001101001
Y	89	59	8	S	1	01001101001
Z	90	5A	8	S	1	00101101001
[	91	5B	8	S	1	01101101001
\	92	5C	8	S	1	00011101001
]	93	5D	8	S	1	01011101001
^	94	5E	8	S	1	00111101001
	95	5F	8	S	1	01111101001
`	96	60	8	S	1	00000011001
a	97	61	8	S	1	01000011001
b	98	62	8	S	1	00100011001
c	99	63	8	S	1	01100011001
d	100	64	8	S	1	00010011001
e	101	65	8	S	1	01010011001
f	102	66	8	S	1	00110011001
g	103	67	8	S	1	01110011001
h	104	68	8	S	1	00001011001
i	105	69	8	S	1	01001011001
j	106	6A	8	S	1	00101011001
k	107	6B	8	S	1	01101011001
l	108	6C	8	S	1	00011011001
m	109	6D	8	S	1	01011011001
n	110	6E	8	S	1	00111011001
o	111	6F	8	S	1	01111011001
p	112	70	8	S	1	00000111001
q	113	71	8	S	1	01000111001
r	114	72	8	S	1	00100111001
s	115	73	8	S	1	01100111001
t	116	74	8	S	1	00010111001
u	117	75	8	S	1	01010111001
v	118	76	8	S	1	00110111001

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
w	119	77	8	S	1	01110111001
x	120	78	8	S	1	00001111001
y	121	79	8	S	1	01001111001
z	122	7A	8	S	1	00101111001
{	123	7B	8	S	1	01101111001
\| | 124 | 7C | 8 | S | 1 | 00011111001 |
} | 125 | 7D | 8 | S | 1 | 01011111001 |
~ | 126 | 7E | 8 | S | 1 | 00111111001 |
DEL | 127 | 7F | 8 | S | 1 | 01111111001 |
* | 128 | 00 | 8 | S | 1 | 00000000101 |
* | 129 | 01 | 8 | S | 1 | 01000000101 |
* | 130 | 02 | 8 | S | 1 | 00100000101 |
* | 131 | 03 | 8 | S | 1 | 01100000101 |
* | 132 | 04 | 8 | S | 1 | 00010000101 |
* | 133 | 05 | 8 | S | 1 | 01010000101 |
* | 134 | 06 | 8 | S | 1 | 00110000101 |
* | 135 | 07 | 8 | S | 1 | 01110000101 |
* | 136 | 08 | 8 | S | 1 | 00001000101 |
* | 137 | 09 | 8 | S | 1 | 01001000101 |
* | 138 | 0A | 8 | S | 1 | 00101000101 |
* | 139 | 0B | 8 | S | 1 | 01101000101 |
* | 140 | 0C | 8 | S | 1 | 00011000101 |
* | 141 | 0D | 8 | S | 1 | 01011000101 |
* | 142 | 0E | 8 | S | 1 | 00111000101 |
* | 143 | 0F | 8 | S | 1 | 01111000101 |
* | 144 | 10 | 8 | S | 1 | 00000100101 |
* | 145 | 11 | 8 | S | 1 | 01000100101 |
* | 146 | 12 | 8 | S | 1 | 00100100101 |
* | 147 | 13 | 8 | S | 1 | 01100100101 |
* | 148 | 14 | 8 | S | 1 | 00010100101 |
* | 149 | 15 | 8 | S | 1 | 01010100101 |
* | 150 | 16 | 8 | S | 1 | 00110100101 |
* | 151 | 17 | 8 | S | 1 | 01110100101 |
* | 152 | 18 | 8 | S | 1 | 00001100101 |
* | 153 | 19 | 8 | S | 1 | 01001100101 |
* | 154 | 1A | 8 | S | 1 | 00101100101 |
* | 155 | 1B | 8 | S | 1 | 01101100101 |
* | 156 | 1C | 8 | S | 1 | 00011100101 |
* | 157 | 1D | 8 | S | 1 | 01011100101 |
* | 158 | 1E | 8 | S | 1 | 00111100101 |

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	159	1F	8	S	1	01111100101
*	160	20	8	S	1	00000010101
*	161	21	8	S	1	01000010101
*	162	22	8	S	1	00100010101
*	163	23	8	S	1	01100010101
*	164	24	8	S	1	00010010101
*	165	25	8	S	1	01010010101
*	166	26	8	S	1	00110010101
*	167	27	8	S	1	01110010101
*	168	28	8	S	1	00001010101
*	169	29	8	S	1	01001010101
*	170	2A	8	S	1	00101010101
*	171	2B	8	S	1	01101010101
*	172	2C	8	S	1	00011010101
*	173	2D	8	S	1	01011010101
*	174	2E	8	S	1	00111010101
*	175	2F	8	S	1	01111010101
*	176	30	8	S	1	00000110101
*	177	31	8	S	1	01000110101
*	178	32	8	S	1	00100110101
*	179	33	8	S	1	01100110101
*	180	34	8	S	1	00010110101
*	181	35	8	S	1	01010110101
*	182	36	8	S	1	00110110101
*	183	37	8	S	1	01110110101
*	184	38	8	S	1	00001110101
*	185	39	8	S	1	01001110101
*	186	3A	8	S	1	00101110101
*	187	3B	8	S	1	01101110101
*	188	3C	8	S	1	00011110101
*	189	3D	8	S	1	01011110101
*	190	3E	8	S	1	00111110101
*	191	3F	8	S	1	01111110101
*	192	40	8	S	1	00000001101
*	193	41	8	S	1	01000001101
*	194	42	8	S	1	00100001101
*	195	43	8	S	1	01100001101
*	196	44	8	S	1	00010001101
*	197	45	8	S	1	01010001101
*	198	46	8	S	1	00110001101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	199	47	8	S	1	01110001101
*	200	48	8	S	1	00001001101
*	201	49	8	S	1	01001001101
*	202	4A	8	S	1	00101001101
*	203	4B	8	S	1	01101001101
*	204	4C	8	S	1	00011001101
*	205	4D	8	S	1	01011001101
*	206	4E	8	S	1	00111001101
*	207	4F	8	S	1	01111001101
*	208	50	8	S	1	00000101101
*	209	51	8	S	1	01000101101
*	210	52	8	S	1	00100101101
*	211	53	8	S	1	01100101101
*	212	54	8	S	1	00010101101
*	213	55	8	S	1	01010101101
*	214	56	8	S	1	00110101101
*	215	57	8	S	1	01110101101
*	216	58	8	S	1	00001101101
*	217	59	8	S	1	01001101101
*	218	5A	8	S	1	00101101101
*	219	5B	8	S	1	01101101101
*	220	5C	8	S	1	00011101101
*	221	5D	8	S	1	01011101101
*	222	5E	8	S	1	00111101101
*	223	5F	8	S	1	01111101101
*	224	60	8	S	1	00000011101
*	225	61	8	S	1	01000011101
*	226	62	8	S	1	00100011101
*	227	63	8	S	1	01100011101
*	228	64	8	S	1	00010011101
*	229	65	8	S	1	01010011101
*	230	66	8	S	1	00110011101
*	231	67	8	S	1	01110011101
*	232	68	8	S	1	00001011101
*	233	69	8	S	1	01001011101
*	234	6A	8	S	1	00101011101
*	235	6B	8	S	1	01101011101
*	236	6C	8	S	1	00011011101
*	237	6D	8	S	1	01011011101
*	238	6E	8	S	1	00111011101

ASCII	DEC	HEX	DATA	PAR	STOP	COMBINATION S12345678Ps
*	239	6F	8	S	1	01111011101
*	240	70	8	S	1	00000111101
*	241	71	8	S	1	01000111101
*	242	72	8	S	1	00100111101
*	243	73	8	S	1	01100111101
*	244	74	8	S	1	00010111101
*	245	75	8	S	1	01010111101
*	246	76	8	S	1	00110111101
*	247	77	8	S	1	01110111101
*	248	78	8	S	1	00001111101
*	249	79	8	S	1	01001111101
*	250	7A	8	S	1	00101111101
*	251	7B	8	S	1	01101111101
*	252	7C	8	S	1	00011111101
*	253	7D	8	S	1	01011111101
*	254	7E	8	S	1	00111111101
*	255	7F	8	S	1	01111111101

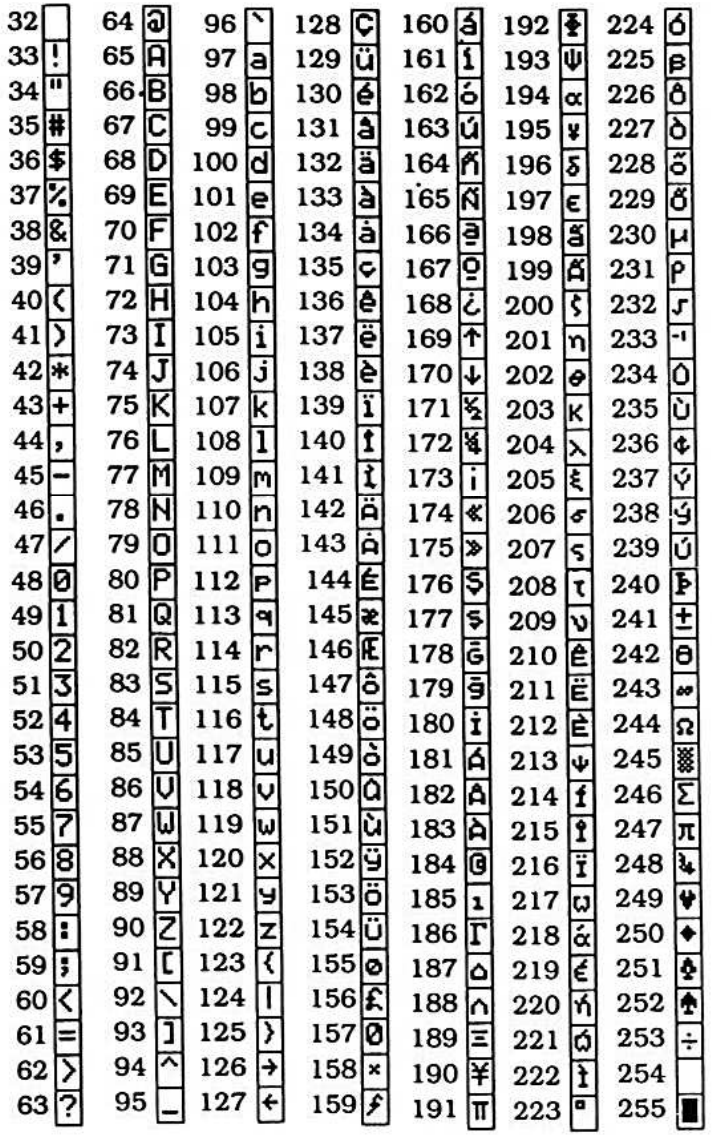
RS-232 Voltage and Logic Levels

Voltage	Logic	Handshaking Control	Teletype	L E D Colour
+3V to +25V	0	On Active Asserted	Space	Red
-3V to -25V	1	Off Inactive Idle	Mark	Green

The above table shows the two logic states of a line on RS-232 together with names that are given to those states which mean the same thing.

We have always found it confusing knowing which logic state the RS-232 lines are in. This table will help.

Remember that the logic number (0 or 1) should be used with the ASCII tables and that when a line is waiting in an Idle state it waits at logic 1.



The table on this page shows the characters which have the ASCII codes 32 to 255. 32 is the space character.

Index

R

S

T

U

W

Software for the Psion Organiser

Special Offer

RVH - Assembler/Debugger £99.95 incl VAT

This package allows the programmer to write, run and debug machine code programs. It has many attractive features such as main assembler or mini-assembler, machine code monitor for patching RAM directly, disassembler, RAM dumping, single stepping or executing to a break-point and printer output (via Comms Link). **Quote from "Ipso Facto" Psion user group newsletter Jan '90:- "From a purely personal point of view, this is the best piece of software I have seen for the Psion Organiser.....Les Ball"**

Psion Comms Link RS232C (IBM PC) £59.95 incl VAT

This unit includes both hardware and software to link the Organiser to an IBM PC or clone. The ROM software included in the interface also allows communication with many other devices.

Autoscribe Plus (from Widget Software) **£49.95 incl VAT**

An exceptional word processor which allows on-the-spot recording of reports, letters etc. Featuring:- Up to 240 characters to a line and document size limited only by size of RAM Pack. Full cursor control and insert. Text automatically saved. Automatic word wrap. Special non-keyboard character feature. Search and replace. List documents. Cut and paste. Send and receive via the Comms Link. Uses bold, underline, italics etc on Epson compatible printers.

Paralink £59.95 incl VAT

Allows the Organiser to print directly to a printer with a standard Centronics compatible interface (that's most of them!). Works with your own OPL programs. Includes main socket.

Transfile ST £49.95 incl VAT

Allows error free transfer of informaton between the Atari ST and an Organiser via the Comms Link. It can easily transmit and receive all data files, diary files, notepads and OPL programs. This program uses the Psion protocol for data transfer and allows you to print from your Organiser via the Atari ST.

Back-it-up £49.95 incl VAT

This program is run from an IBM PC, XT, AT or clone and allows all the information to be pulled from your Organiser and saved. This includes data, diaries, alarms and programs from all drives. A Comms Link must connect your PC to the Organiser. Restoration of data is of course very easy.

Filemaster (from Widget Software) **£49.95 incl VAT**

The user can take control of the Organiser's filing structure and use a long list of additional functions including sort, multiple criteria, search, select. printing of records, diary print, can edit up to 110 database files. Allows checking of input data, auto-load and printing of data in columns.

Additional Hardware Items

128K and 256K RAM.PAGE packs provide a massive amount of battery backed RAM for the Psion Organiser II within a standard size casing. These can be used in exactly the same way as normal RAMpaks or Datapaks.

These programs are obtainable from Kuma Computers Ltd, 12 Horseshoe Park, Pangbourne, Berks, RG8 7JW, phone 0734 844335, or your local Psion dealer.

NOTES

NOTES

NOTES

NOTES

PSION ORGANISER COMMS HANDBOOK

Computer communications can be a daunting subject, especially to the uninitiated.

Written in a clear, sensitive and highly readable form, this book provides the reader with all the information needed to link the Psion Organiser to another machine.

It begins with an in-depth introduction to the world of computer communications, going on to explain testing and fault finding techniques. The text is supported by detailed explanation, along with numerous illustrations and diagrams.

All this information is made easily accessible by the extensive index and contents pages.

Published by

Kuma

ISBN 0-7457-0154-X

01495

Kuma Computers Ltd,
Pangbourne, Berkshire, England
Telephone: 0734 - 844335

£14.95 net