AMSTRAD

CPS8256

RS232C SERIAL INTERFACE
&
PARALLEL CENTRONICS INTERFACE

USER INSTRUCTION BOOK

AMSTRAD CPS8256
RS232C SERIAL INTERFACE
& PARALLEL
(CENTRONICS) INTERFACE

AMSOFT

A division of

AMSTRAD

CONSUMER ELECTRONICS PLC.

AMSOFT welcome comments and suggestions relating to the product or to this manual.

All correspondence should be addressed to:

AMSOFT
Brentwood House
169 Kings Road
Brentwood
Essex CM14 4EF

CPS8256, PCW8256
First Published 1985

Written by Roland Perry
Illustrated by Alexander Martin and Julie Morement

Programming by AMSOFT:

MAIL232 by Vik Olliver and Roland Perry

Published by AMSTRAD
Typeset by KAMSET typesetting graphics (Brentwood)

AMSTRAD CPS8256 Instruction Manual

When you have fitted the AMSTRAD CPS8256 to your PCW8256 you have the means to connect to alternative printers, modems and other computers. Both an RS232C and a Parallel 'Centronics' output are available. In order to use the CPS8256 you need both suitable programs or instructions (software) and an appropriate wire connection. Implementing such a connection is often regarded as a very mysterious and complicated business, if only because of the serial/Centronics interface's inherent flexibility and versatility. To help you use your AMSTRAD CPS8256 interface, we have provided simple introductory instructions.

Full technical explanation is available in a series of appendices. The simple examples are not intended to be an exhaustive survey of all the possible uses of the CPS8256. They do represent, however, the majority of applications encountered.

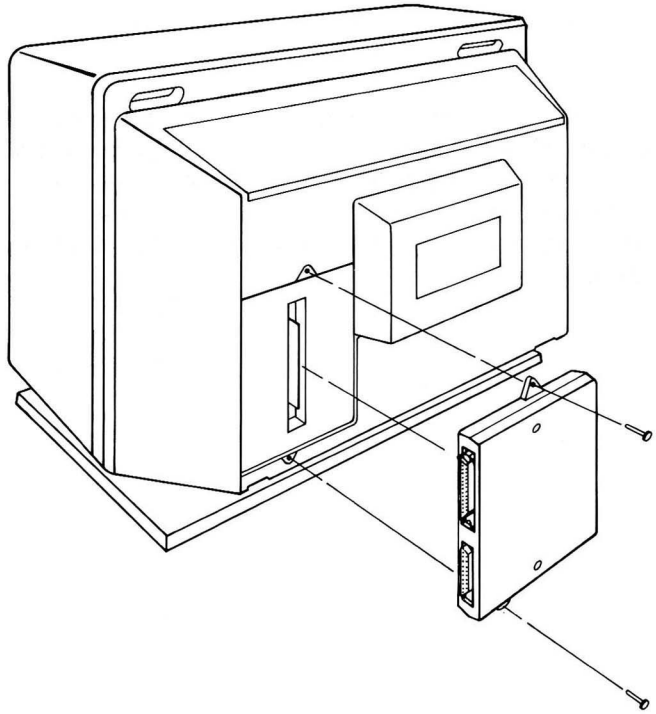
The AMSTRAD CPS8256 Serial/Centronics interface, with these instructions, is only suitable for use with the AMSTRAD PCW8256 computer. The CP/M software on side 2 of your system discs already contains instructions to access the CPS8256 when fitted. The sign-on message is automatically adjusted to include a reference to the CPS8256 (SIO/Centronics add-on).

Side 1 of your system discs contains a program called MAIL232 which allows you to communicate with electronics mail services and emulate a terminal connected to other computers. [NB: Although this program is supplied on Side 1, for reasons of diskette space, it is not part of LocoScript and can only be used after CP/M has been loaded from side 2 of your system discs.]

Use of the serial interface section of the CPS8256 allows us to connect equipment together using very simple wires up to 50 feet long. It is possible to convert the signals in these wires, using a modem, into a form in which they can be sent almost any distance to another modem over standard telephone lines. Most equipment manufacturers have agreed the connection details to a standard called 'RS232C'. It is also possible to connect devices which conform to the alternative 'RS423' standard.

Normally, the parallel (Centronics) section of the CPS8256 will allow connection to printers via a multi-way cable of up to 2 metres, depending on the exact electrical characteristics of the printer.

Stage 1:
Connecting the CPS8256 to your computer



Switch off the computer and remove anything connected to the port marked 'EXPANSION'.

Fit the CPS8256 unit by pushing it onto the connector plug of the port marked 'EXPANSION' making sure that the serial and parallel output connectors point towards the outside of the computer. The CPS8256 can then be secured by two fixing screws.

Power for the CPS8256 is derived from inside the PCW8256 and is applied when the PCW8256 is switched on.

The serial output lead (25-way chassis plug) and parallel output lead (36-way chassis socket) may be connected or disconnected at any time, regardless of power supply considerations.

WARNING - ELECTRIC SHOCK

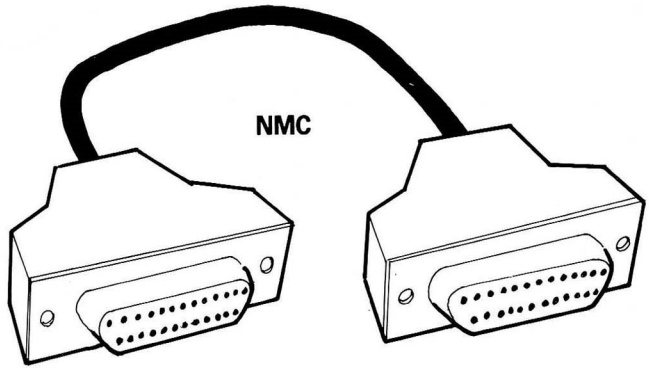
NEVER ATTEMPT TO OPERATE THE CPS8256 UNIT WITH ITS CASE REMOVED.

Stage 2:
Cables for connecting a Parallel-Interface Printer (or plotter...)

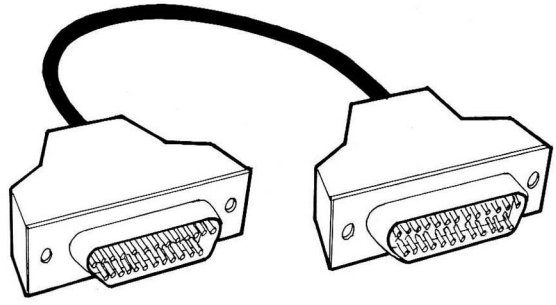
The cable required to connect to a parallel-interface printer (or any other output device) consists of two identical connectors, linked by a 36-way cable. The cable can be connected either way around. In fact, only about half of the wires in the cable are actually used, but it is common practice to use a cable with all 36 wires connected. A full specification of the parallel connector is given in Appendix 4.

Stage 3:
Cables for connecting a Serial-Interface Printer (or plotter...)

The cable required to connect to a printer (or any other output device) is called a Null Modem Cable (NMC). A technical explanation and drawing of the connections required is given in Appendix 1.



A Null Modem Cable has Cable-Sockets at each end, to plug into the Chassis-Plugs on the CPS8256 and the printer. If your printer has a Chassis-Socket then you will also require a converter cable comprising two back-to-back Cable-Plugs.



Stage 4:
Setting the speed of your printer.
(Serial-Interface only)

It is possible to alter the speed at which characters are transmitted from the CPS8256 serial interface to the printer. (When using the parallel interface the transfer rate is automatically controlled by the printer.) The speed is measured by a figure called the 'Baud Rate', which has a value of approximately ten times the number of characters per second.

You need to make sure that the CPS8256 is transmitting at the same speed as the printer is receiving. Do not confuse this serial interface speed with the speed at which the printer actually prints characters on the paper. If the printer cannot keep up with the rate at which characters are arriving from the serial interface then it will send special signals back to the CPS8256 instructing it to stop sending until the printer catches up. This process is known as 'flow control' or 'Hardware Handshaking'.

Your printer will probably have some switches (possibly inside the case) to set its baud rate. The best speed to choose is 9600 baud (approx 960 characters per second) because that is the default speed of the CPS8256.

Stage 5:
Setting the Speed of the CPS8256 (Serial-Interface connection only).

The CPS8256 operates at 9600 Baud unless instructed otherwise.

MAIL232: The speed of the serial interface is set by operating a 'pull-down menu', found by pressing the function key **[f1]**. The large inverse bar cursor can be moved from line to line using the `UP` and `DOWN` cursor keys. Select the required send and receive baud rates from those available by pressing the **[+]** key and finish by pressing the **[EXIT]** key. Note that MAIL232 resets all the options available in the **[f1]** menu whenever it is loaded.

Other CP/M Plus programs: If there is no method provided within the program, the Baud rate must be changed while in CP/M Direct Mode (i.e. at the `A>` prompt);

The command is `SETSIO,`‹baud rate›

..or the alternative `DEVICE SIO` [‹baud rate›]

`SETSIO` is more versatile, in particular it will allow the setting of split transmit/receive baud rates.

examples:

```
SETSIO 300
DEVICE SIO [300]          .. sets both send and receive to
                             300 baud;
```

or

```
SETSIO RX 1200, TX 75     .. sets transmit to 75 baud,
                             receive to 1200 baud.
```

If you require a baud rate other than 9600 it must be set up every time you reset the computer or move from LocoScript to CP/M or load `MAIL232`. It is possible to program the alternative baud rate as part of the loading process by incorporating the appropriate command into the '`PROFILE.SUB`' file.

(NB: Under CP/M Plus, if the baud rate is changed by `SETSIO`, interrogation by use of the command `DEVICE` will not acknowledge the change.)

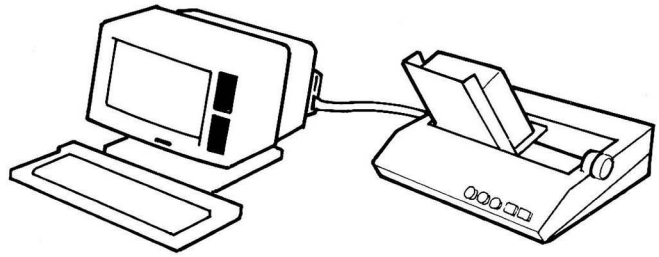
Stage 6:
Setting printer framing bits (Serial-Interface only).

Characters sent by the CPS8256 have properties additional to their speed. These properties are: the number of 'Data' bits, the number of 'Stop' bits and the type of 'Parity'. It is not important to understand the exact nature of these properties, but, as with the baud rate, the printer and CPS8256 should be arranged to match.

Most printers will have switches for these properties. It is not always essential that the RS232 and the printer match exactly - the final test is to try a particular configuration and see if everything works as expected.

The CPS8256 is set by default to 8 data bits, 1 stop bit and no parity. See appendix 2 (CP/M) or Stage 11 (MAIL232), for a complete description of the commands required to alter the CPS8256 framing bits.

Stage 7:
Redirecting the printer output via the CPS8256



Normally printer output is sent to the internal PCW8256 Printer. When operating in CP/M Plus, it can be re-directed to either the Parallel (Centronics) or Serial Interface of the CPS8256. It is possible, also, to re-direct printer output to the parallel port whilst simultaneously using the serial interface for communications.

The following command will cause all printer output to be sent via the serial interface.

```
DEVICE LST:=SIO
```

The following command will cause all printer output to be sent via the parallel (Centronics) port.

```
DEVICE LST:=CEN
```

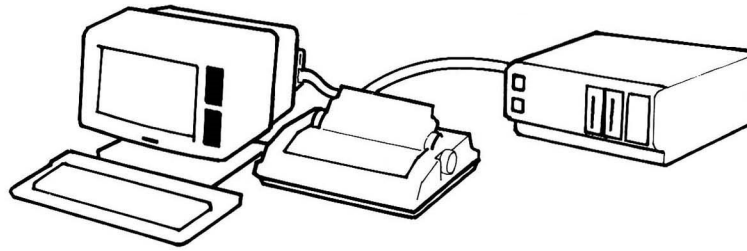
When operating with CP/M it is possible to program the redirection as part of the loading process by incorporating the command given above into the 'PROFILE.SUB' file.

Stage 8:
Restoring the printer output to internal printer.

Every time you reset the computer or move from LocoScript to CP/M the printer output will be restored to the parallel port. This can also be done using the command:

```
DEVICE LST:=LPT
```

Stage 9:
Cables to connect Terminal Emulator to a minicomputer.



In this manual, a commercial microcomputer that requires a terminal or Visual Display Unit (VDU) in order to operate is also referred to as a minicomputer.

It is possible to connect your computer fitted with CPS8256 as a substitute for a terminal attached directly to a minicomputer. The cable to use for this will depend on the minicomputer. Either use a Null Modem Cable (Cable-Socket to Cable-Socket) if the minicomputer has a chassis plug, or a Modem Cable (Cable-Socket to Cable-Plug) if the minicomputer has a Chassis-Socket. If in doubt, consult appendix 1 and the hardware manual of the minicomputer.

Stage 10:
Attaching the Terminal Emulator to a minicomputer.

The baud rate and framing parameters of the CPS8256 and minicomputer should agree, as discussed previously when attaching to printers.

The Terminal Emulator is part of the MAIL232 program. Load CP/M Plus from Side 2 of your system discs and then insert Side 1 and type:

```
MAIL232
```

The main screen shows a number of options related to use as an electronic mail terminal. In this Mail Terminal configuration, the PCW8256 will operate as a very simple 'glass teletype' printing characters and obeying only Carriage Return and Linefeed codes. It can be toggled from Online to Local by entering the **[f5]** menu and pressing the **[+]** then **[EXIT]** keys.

Having set the baud rate and framing bits, (**[f1]** menu) the Terminal Emulator is entered by selecting the first option of the **[f7]** menu and pressing **[ENTER]**. The codes obeyed by the Terminal Emulator are broadly in line with the codes for a Heath/Zenith H19/Z19 or DEC VT52.

If the minicomputer can support it, it is best to operate the terminal with hardware handshake enabled (see **[f1]** menu) otherwise some characters may occasionally be lost.

The Terminal Emulator is 'exit'ed by pressing **[ALT] [STOP]**, and the Mail Terminal is 'exit'ed by selecting the second option (using the Down Cursor key) of the **[f7]** menu and pressing **[ENTER]**.

Note that the Terminal Emulator 'inherits' whatever keyboard values have been previously set up by CP/M except that **[◄DEL]** is re-configured as `H` (Code 8). The **[EXIT]** key will normally be set to the value known as 'Escape'.

Stage 11:
Connecting to another computer via a modem.

A modem is simply a way of extending the length of the serial connection between two computers - normally via the public telephone network. The connection to the modem is made by a Modem Cable (Cable-Socket to Cable-Plug) if the modem is fitted with a suitable Chassis-Socket. Otherwise a special cable is required, and appendix 1 gives explanations and examples of this.

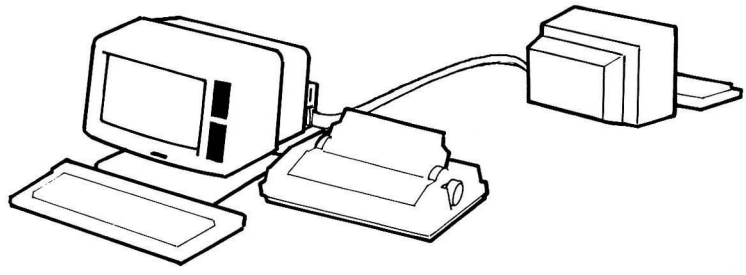
The baud rate and framing parameters of the CPS8256 should be set to match the modem and distant computer.

For example, two common speeds for modems are 300 baud and 1200 baud (receive)/ 75 baud (transmit). It is also quite common to require the framing bits to be : 7 Data Bits, Odd Parity, 1 Stop Bit.

The framing bits can be set within MAIL232 by operating the **[f1]** menu. The large inverse bar cursor can be moved from line to line using the **UP** and **DOWN** cursor keys. Select the required parameters from those available by pressing the **[+]** key and finish by pressing the **[EXIT]** key. Note that MAIL232 resets all the options available in the **[f1]** menu whenever it is loaded.

See stage 10 for instructions on entering the Terminal Emulator software. Note that a Modem link will not normally support the use of hardware handshaking, and it is possible that some characters may be lost when operating at high transfer rates.

Stage 12:
Connecting two computers for file transfers.



It is often convenient to transfer files between different computers via a serial connection, particularly when there is no common disc format between the two machines.

It is assumed that you will normally wish to receive files on your AMSTRAD computer, transmitted from some other computer. The protocol is published in appendix 3 should you wish to program an alternative computer to receive files. A transmitting program will also be required if your transmitting computer is not a CP/M 80 computer.

Follow the guidelines for connecting your computer to the transmitting computer or modem (as appropriate). Make sure that the baud rates and framing parameters match. If you are to use the '`Transfer as HEX`' option you must use the default 8 'data' bits.

Stage 13:
Receiving the file on your AMSTRAD computer.

With the PCW8256/CPS8256 set to the appropriate baud rate and framing bits, select the [f3] option of the MAIL232 program.

If the file you wish to receive is plain text (ASCII) or is being sent by a computer which has no specific knowledge of your AMSTRAD computer then use this default mode. Otherwise, if the file is a CP/M program, LocoScript file, etc. or you are receiving from another MAIL232 or CPC464/664/6128 computer with Serial Interface, move the large bar cursor to the bottom line of the menu and press **[+]** to select '`Transfer as HEX`'.

The name of the file into which the information will be received should be typed into the 'Receive' field of the **[f3]** menu, and when **[ENTER]** is pressed, the reception will begin. It is possible to 'pre-load' the filename, then press **[EXIT]** and return to the **[f3]** menu later, then merely pressing **[ENTER]** to initiate the transfer.

If the file being received is ASCII:

The screen will clear and characters received will be displayed. The file will initially be absorbed by various RAM buffers within the PCW8256, but there will come a time when a physical write to disc is required. If no hardware handshaking is in operation, it is quite likely that a few characters will be missed while the PCW8256 is fully occupied writing to the disc. (NB: Hardware handshaking is not normally possible via a Modem link.)

The transfer is ended by pressing **[ALT] [STOP]** at the discretion of the receiving user, at which point the computer returns to the Mail Terminal Mode.

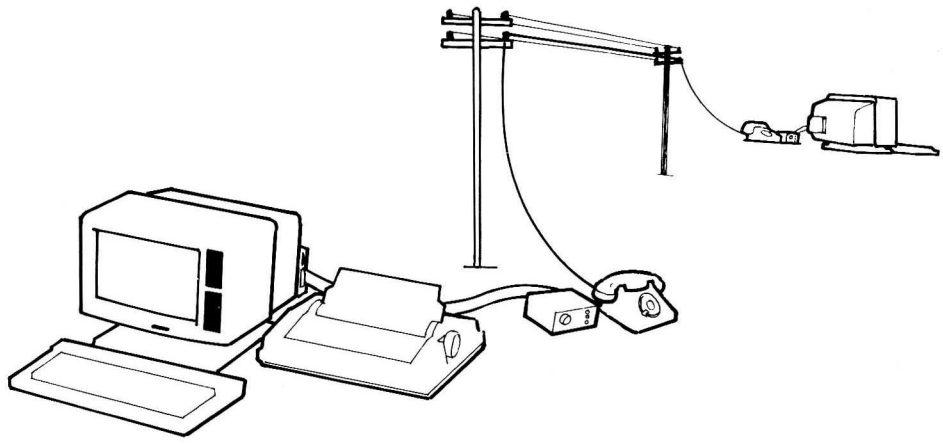
The PCW8256 does NOT echo the characters received back to the sender. (See stage 14).

If the file being received is HEX:

The transfer should proceed automatically, with protocol and error checking agreed by the sending software. While the transfer is in progress the small character cursor will disappear. When the transfer has ended the 'Receive' field of the **[f3]** menu will clear. If an unrecoverable error is discovered then your computer will beep, and the character cursor will re-appear. The transfer may be re-started simply by pressing **[ENTER]**. (NB: The sending end must also manually re-start its transfer.)

After a successful or aborted transfer, press **[EXIT]** to return to the Mail Terminal Mode.

Stage 14:
Transmitting the file from an AMSTRAD computer



This is particularly appropriate when two distant AMSTRAD computers are connected by modem, making a simple physical transfer of disc impossible. The receiving program checks the incoming data to ensure that there are no errors, and requests a re-transmission if necessary.

With the PCW8256/CPS8256 set to the appropriate baud rate and framing bits, select the **[f3]** option of the MAIL232 program.

It is normally best to use the Transfer as HEX option. (cursor down to the bottom line of the **[f3]** menu and press **[+]**).

The name of the file from which the information will be sent should be typed into the 'Send' field of the **[f3]** menu, and when **[ENTER]** is pressed, the transmission will begin. It is possible to 'pre-load' the filename, then press **[EXIT]** and return to the **[f3]** menu later, then merely pressing **[ENTER]** to initiate the transfer.

If the file sending mode is ASCII:

The screen will clear and characters received will be displayed. Note that it is the responsibility of the *RECEIVING* computer to echo the characters sent to it in order that the sending computer has anything to display. It is quite possible that a transfer could successfully take place without any display at all at the sending end. Also, the echo may take place after a short delay, causing recently-sent characters to be lost, if there is no hardware handshaking in operation, while the sending computer is occupied full-time reading the next section of the file from the disc. (NB: Hardware handshaking is not normally possible via a Modem link.)

The transfer is ended by end of file. The computer returns to the Mail Terminal Mode when a Newline echo character is received, or when any key on the keyboard is pressed.

If the file being sent is HEX:

The transfer should proceed automatically, with protocol and error checking agreed by the receiving software. While the transfer is in progress the small character cursor will disappear. When the transfer has ended the 'Send' field of the **[f3]** menu will clear. If an unrecoverable error is discovered then your computer will beep, and the character cursor will re-appear. The transfer may be re-started simply by pressing **[ENTER]**. N.B. The receiving end must also manually re-start its transfer.

After a successful or aborted transfer, press **[EXIT]** to return to the Mail Terminal Mode.

Stage 15:
Transmitting the file from a CP/M computer

If the file can be transmitted as a simple text (ASCII) file, then simply arrange to send the file to the serial output of the CP/M computer.

```
PIP PUN:=FILE.TYP  ..is a typical command for CP/M 2.2
PIP AUX:=FILE.TYP  ..is a typical command for CP/M Plus.
```

There are, however, advantages in sending as HEX; mainly due to the error detection this provides. Program files must ALWAYS be sent as HEX.

Enter the following HEX dump into your CP/M computer using a text editor or PIP. e.g:

```
PIP SEND.HEX=CON:
```

‹‹ Enter the dump a line at a time, each terminated by **[ENTER] [CTRL]J** ››

```
:180100003A5D00FE20CA0502115C000E0FCD0500FEFFCA0E02CD26013A
:180118000E10CD050011A1020E09CD0500C721FFFF22B8021E020E044E
:18013000CD0500CD8101D22C012AB8022322B802CD9C01CDB001CDD02F
:180148001B7C26A01CDB801CDC001CDC801CD810138DECD9C01CDB0C4
:18016000001C34D01CD9C01CDB0011E000E04CD05002100022BA02CDBF
:18017800C801CD8101D26401C90E03CD0500FE03C29601F1F1117D02A8
:18019000E09CD0500C9FE0637C83FC9215C007EF6405F0E04E5CD0541
:1801A80000E123060BC3170221B8020602C3F5011E800E04CD0500C967
:1801C0002180000680C3F50121BA020602C3F501210000022BA020E1488
:1801D800115C00CD05002180000680E516005E2ABA021922BA02E1236F
:1801F00005C2E301C9E5C55E0E04CD0500C1E12305C2F501C9112A020F
:18020800E09CD0500C7115020E09CD0500C7CDF50106041E000E0419
:18022000C5CD0500C105C21C02C90A074E6F2066696C652073706563 67
:1802380069666965642E0D0A0A5472616E736665722061626F7274657C
:18025000640D0A0A240A0746696C65206E6F7420666F756E642E0D0A6A
:18026800A5472616E736665722061626F72746564 0D0A0A240A075484
:18028000726 16E736665722041626F72746564206279206F7468657257
:1802980020656E642E0D0A0A240A5472616E736665722 0636F6D706CFA
:0C02B000657 4652E0D0A0A240000000091
:0000000000
[CTRL]Z
```

Convert the .HEX file to a .COM file with the command LOAD, e.g:

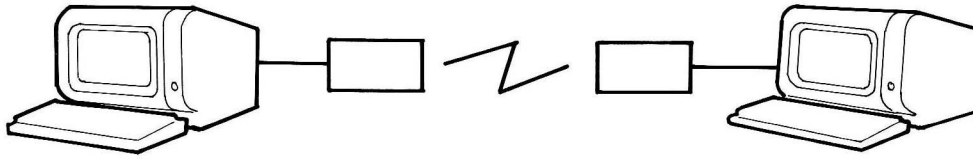
```
LOAD SEND        ..CP/M 2.2
HEXCOM SEND      ..CP/M Plus
```

The program assumes that the serial input/output from the CP/M computer is configured to the Reader/Punch logical devices and that the baud rate and framing bits have been set appropriately. Remember, you must use 8 data bits for transfers in HEX.

Run the program by typing: `SEND FILE.TYP`

Appendix 1: RS232C Connections

For a complete understanding of the connections required between the RS232C and the outside world, it is important to realise that all devices with a serial interface can be classified either as a modem or as a terminal. Modems are merely a way of extending the length of the connection (often via a telephone wire), and Fig 1 (below) shows a simplified, idealised, connection between two terminals.



IDEALISED TERMINAL TO TERMINAL CONNECTION

Fig 1

The standard connector used for serial interfaces has 25 pins although only up to 7 are required in most cases. When connecting a terminal to a modem, a 'one to one' cable is used, i.e. pin 1 to pin 1, pin 2 to pin 2....pin 25 to pin 25. Assuming such cables are in use, data is transferred as follows:

Following the signal path from left to right, characters from the keyboard are sent out of pin 2 of the left-hand terminal, to pin 2 of the modem (the connection marked 'transmit data'). The left-hand modem then sends the characters, via the telephone line, to the right-hand modem. The characters are received at pin 3 of the right-hand modem (the connection marked 'receive data') which sends them to pin 3 of the right-hand terminal. On receipt of the characters, the right-hand terminal displays them on the screen.

Notice how the names of the connections 'transmit data' and 'receive data' are expressed from the point of view of the terminal, not the modem.

The data path from left to right just described, is exactly matched by a data path from right to left which uses the same numbered connections, i.e. pin 2 from terminal to modem (transmitting), and then pin 3 from modem to terminal (receiving). This arrangement is perfectly symmetrical, and there is no confusion over who is using which pin number, and for what direction of data transfer.

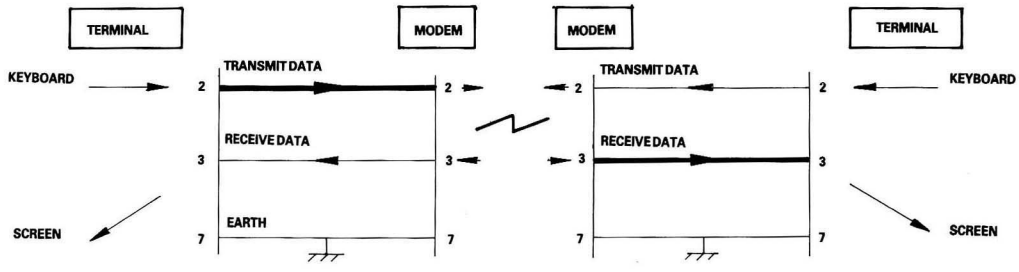


Fig 2

Problems of definition arise, however, when we wish to connect two terminals together locally, without the intervening pair of modems. We cannot connect pin 2 to pin 2 because both keyboards will be transmitting head-on, and neither screen is connected to anyone who is sending. The obvious solution is to cross over pins 2 and 3 so that the transmit pin of each terminal is connected to the receive pin of the other. A cable containing such a cross-over connection is known as a 'Null-modem' cable because of the way in which it replaces the pair of back to back modems.

The earth pin (pin 7) is still common to both terminals using this arrangement.

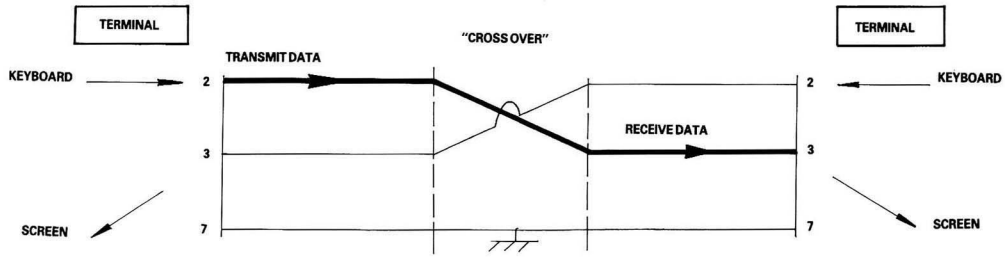


Fig 3

Naturally, the combination of an Amstrad computer+RS232C is considered a terminal, and therefore to connect to a modem, (for example, to access a dial-up database) requires a simple one-to-one cable.

The Null-modem cable is required for connecting to other terminals. The sort of equipment we mean by other terminals is: a second Amstrad computer+RS232C, a conventional Visual Display Unit (VDU), a printer with a serial interface, or perhaps a desk-top computer which requires a VDU.

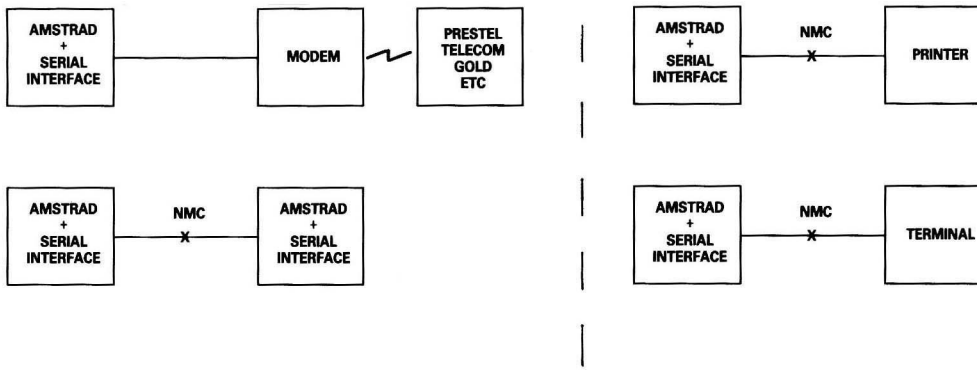


Fig 4

There is a point to be noted here: many manufacturers of desk-top computers wire up the serial interface (for a VDU or a printer) as if it were a modem, not a terminal. This is in the belief that life will therefore be simpler because VDU's and printers can be connected to that computer with one-to-one cables.

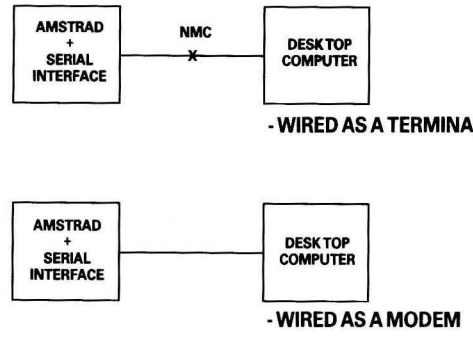


Fig 5

In a perfect world, it would be possible to identify which serial devices behave like modems and which behave like terminals, by examining the 'sex' of the 25-way connector - terminals should have a 'male' connector, and modems a 'female' connector. This is not, unfortunately, as reliable a guide as it should be, as many manufacturers of terminals and printers equip them with 'female' connectors, mostly for reasons of electrical safety.

If in doubt, the ultimate test is to examine the user manual and determine the function of PIN 2 - if the description includes the word 'TRANSMIT' then the equipment is wired as a Terminal, and if it includes the word 'RECEIVE' then the equipment is wired as a modem.

Hardware flow control

The simplified connection described so far does not allow any control of the data flow. In practice, we often wish the receiving device to have some control over the transmitting device, thus preventing the receiving device from being overwhelmed (where it is slower in digesting the input than the rate at which the input is arriving). In addition, if the transmitting device has reason to mistrust the data which it is sending, there should be provision for it to disable the receiving device.

In the case of modem to terminal connection; when the terminal is happy to transmit it activates pin 4 - the RTS pin (Request To Send). When the modem is ready to receive input, it activates pin 5 - the CTS pin (Clear To Send). The terminal will only send when CTS is activated. Thus the modem can control the flow rate using CTS.

When the modem considers that the data which it is about to send is suitable, it activates pin 8 - the DCD pin (Data Carrier Detect). When the terminal is ready to receive input it activates pin 20 - the DTR pin (Data Terminal Ready). The modem will only transmit when DTR is activated. Thus the terminal can control the flow rate using DTR.

There are two further signals which must be introduced here. One is on pin 22 - the Ring Indicator, which simply allows the modem to tell the terminal that the 'phone is ringing! (at which point software in the terminal might be expected to wake up). The other signal is on pin 6 - DSR (Data Set Ready). This signal is ignored by the receiving side of the RS232C; the modem will activate this signal at much the same time as it activates DCD, and therefore no functionality is lost by ignoring DSR.

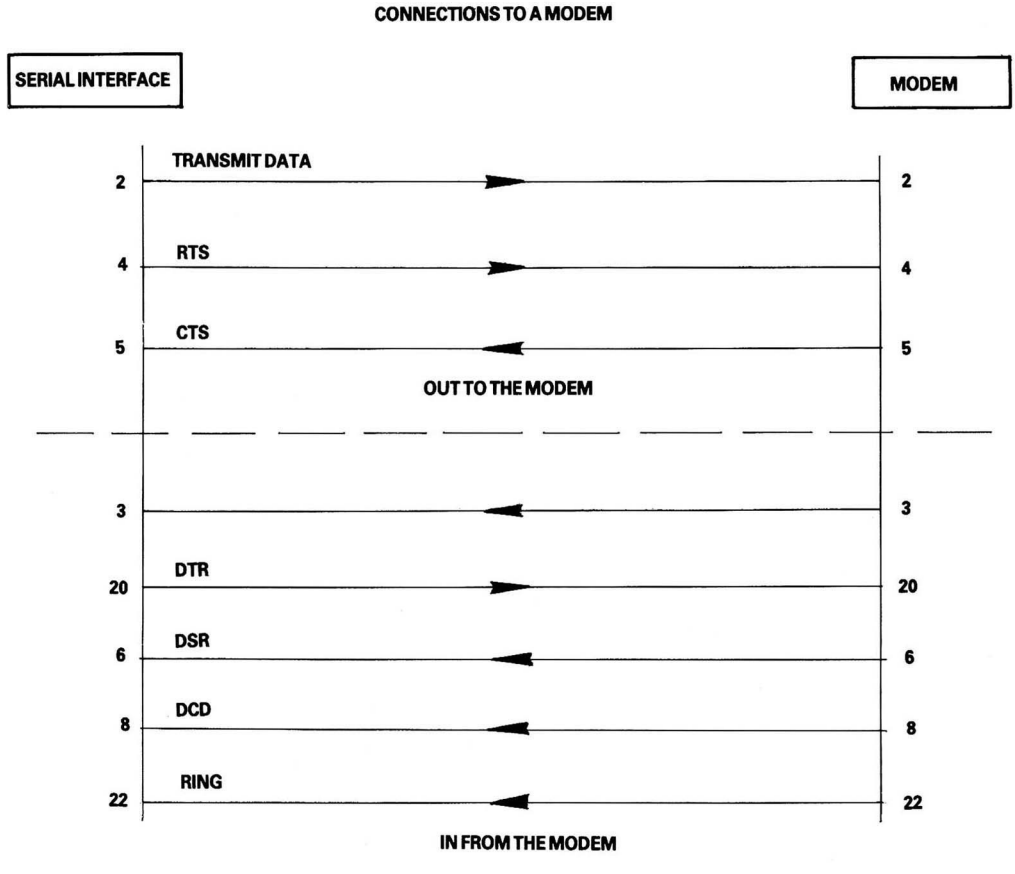

CONNECTIONS TO A MODEM
SERIAL INTERFACE
MODEM
TRANSMIT DATA
2
2
RTS
4
4
CTS
5
5
OUT TO THE MODEM
3
3
DTR
20
20
DSR
6
6
DCD
8
8
RING
22
22
IN FROM THE MODEM

Fig 6

In the case of terminal-to-terminal connections, the Null-modem cable must be used with the additional connections to pins 2, 3, and 7 as already discussed. The full Null-modem cable swaps pins 4 and 8 - the RTS/DCD 'I am happy to send' signals, and pins 20 and 5 - the DTR/CTS 'Busy' signals. To be on the safe side, pin 6 (DSR) is connected to pin 8 (DCD) in case *that* end of the cable is ever connected to a terminal which is fussy and requires DSR as well as DCD.

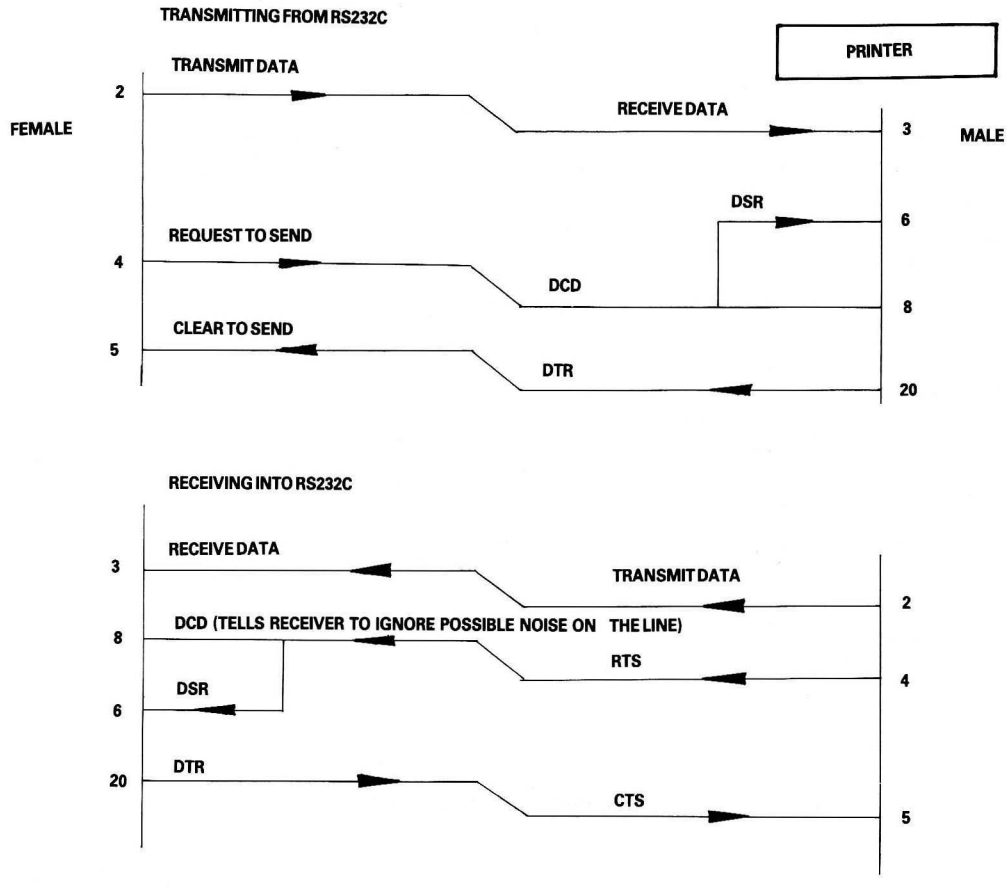
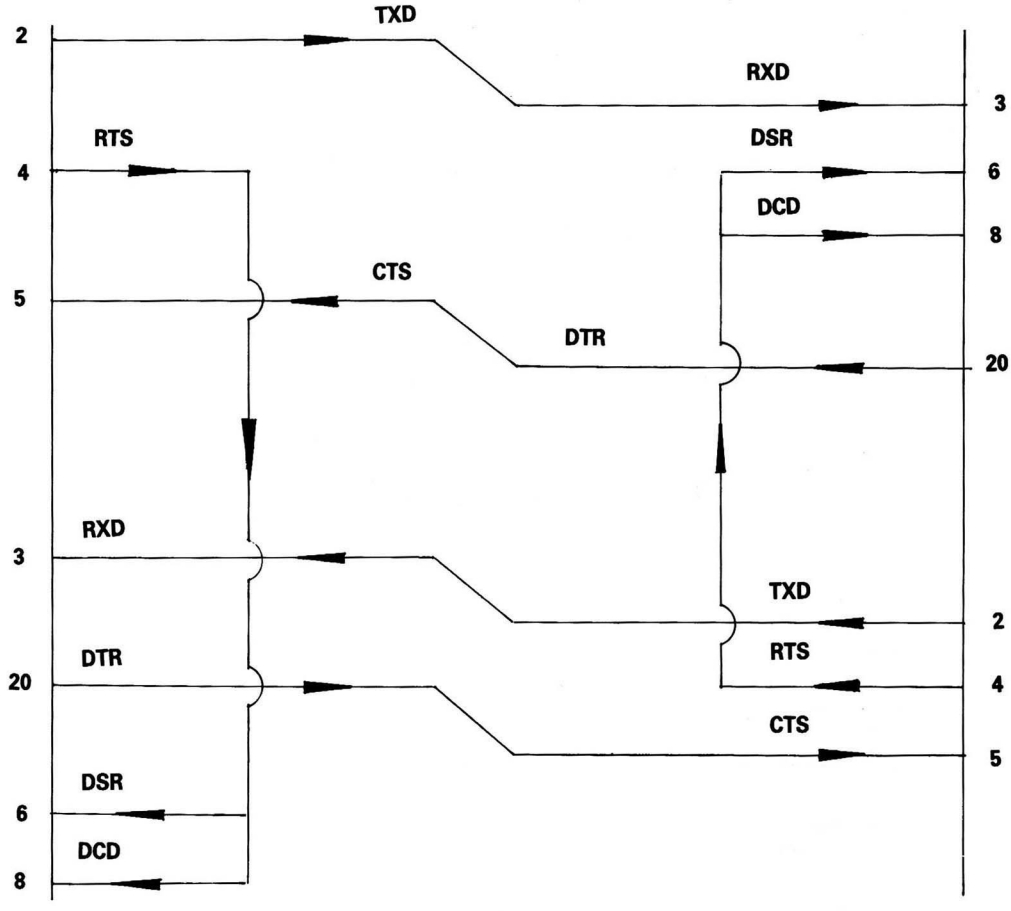


Fig 7

There is a school of thought which says that a Null-modem cable, unlike the pair of modems it replaces, is ALWAYS 'happy to send'. Therefore, it is quite in order to generate DCD (and DSR) permanently. This is achieved by connecting them to the RTS at the same end of the cable, rather than to the RTS at the other end of the cable.



THE RECOMMENDED NMC

Fig 8

Finally, if the transmission rate from one of the two terminals is known to be unstoppable (e.g. a person typing at a keyboard), or is so slow and infrequent (e.g. the software handshake characters 'XON, XOFF' sent by a printer) that there is no danger of over-running the receiving end, then it is permissible to permanently enable the transmission by linking pin 5 (CTS) to pin 4 (RTS), i.e. to always send if ready (at the transmitting end of the cable). It may well be facilitated in any case, for the transmitting terminals to ignore the state of CTS under these circumstances.

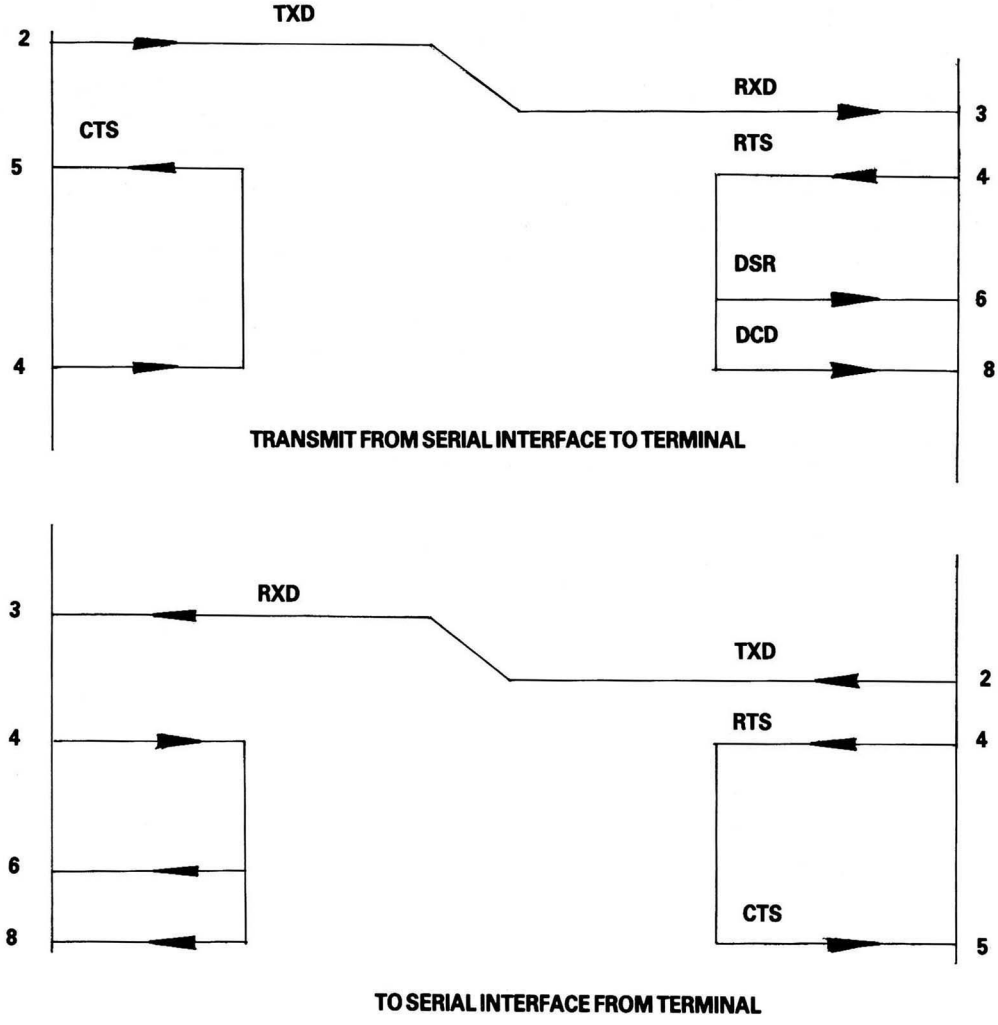

TXD
2
RXD
3
CTS
5
RTS
4
DSR
6
DCD
4
8
TRANSMIT FROM SERIAL INTERFACE TO TERMINAL
RXD
3
TXD
2
4
RTS
4
6
8
CTS
5
TO SERIAL INTERFACE FROM TERMINAL

Fig 9

Appendix 2: Using the CPS8256 in CP/M

Full details of the facilities and programs described below may be obtained by reference to the User Manual (Book 1) and CP/M Guides for the PCW8256.

The CPS8256 parameters are set to default values when CP/M Plus is loaded. The various parameters can be altered at any time by using the program `SETSIO`. It is intended that `SETSIO` will normally be invoked as part of the operation of the startup file `PROFILE.SUB`.

The AMSTRAD BIOS for CP/M Plus implements the CPS8256 as the physical devices `SIO` (for the Serial Interface) and `CEN` (for the parallel port). The utility program `DEVICE` allows us to re-assign these physical devices, and thus permits us to refer to the CPS8256 when operating the utility program PIP or when writing assembler programs which access the BDOS (CALL 5) interface.

`DEVICE` provides the facility of re-assigning the physical devices (SIO, CRT and LPT) to the logical devices `CON:` (which can be sub-divided into `CONIN:` and `CONOUT:`), `AUX:` (which can be sub-divided into `AUXIN:` and `AUXOUT:`) and `LST:`

The console (`CON:`) is normally assigned to the keyboard/screen (`CRT`).

The printer (`LST:`) is normally assigned to the built-in printer (`LPT:`).

The auxiliary device (`AUX:`) is normally assigned to the CPS8256 serial interface (`SIO`).

The current assignments can be interrogated with the command:

```
DEVICE
```

... and may be changed by commands of the form:

`DEVICE` ‹logical device›`=`‹physical device›

eg: `DEVICE LST:=SIO`

... which assigns the CPS8256 serial interface to send printer output, or:

```
DEVICE LST:=CEN
```

... which assigns the CPS8256 parallel interface to send printer output, or:

```
DEVICE CON:=SIO
```

... which allows operation of the computer from a terminal attached to the CPS8256 serial interface. (`DEVICE CON:=CRT` typed on the remote terminal will restore operation to the local keyboard and screen).

If an output assignment is made to a non-existant, or permanently busy, physical device then the computer will issue a warning message to prevent a lock-up condition.

PIP is a file-copying program which regards input and output logical devices as if they were files. Thus...

```
PIP AUX:=FILE.TYP    will send a file to the CPS8256; and
PIP FILE.TYP=AUX:    will receive a file from the CPS8256
```

... if the default assignments are in force. Transfers are generally terminated by a **[CTRL]Z** character. (Do not confuse this simple method of file input and output with the facilities of MAIL232).

SETSIO displays and changes the parameters of the CPS8256 serial interface. Any number of the parameters may be changed - simply include the required clauses in any order.

e.g. `SETSIO TX 75 RX 1200 BITS 7 PARITY ODD`

Only the initial letter of the first word in each clause is required; the rest of that word is optional. An illegal clause will produce an error message, and if a clause is specified twice then the latter clause is used.

If the Baud rate (TX or RX) is omitted, then both TX and RX are set. If the baud rate is changed, but the number of stop bits is not, then the number of stop bits will be set to 1 (if the baud rate is greater than 110), or will otherwise be set to 2.

Clauses available are:

TX ‹baud rate› - default 9600 - sets transmitter baud rate.

RX ‹baud rate› - default 9600 - sets receiver baud rate.

... where ‹baud rate› must be one of the following numbers 50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600 or 19200.

BITS ‹n› - default 8 - sets the number of data bits.

... where the number of bits must be one of the numbers 5, 6, 7 or 8.

PARITY ‹p› - default NONE - sets the parity

... where ‹p› must be one of the words EVEN, ODD or NONE

STOP ‹n› - default 1 - sets the number of stop bits

... where ‹n› must be one of the numbers 1, 1.5, or 2.

XON ‹x› - default OFF - invokes or cancels the software flow control

... where ‹x› must be one of the words ON or OFF.

HANDSHAKE ‹x› - default ON - invokes or cancels the hardware flow control

... where ‹x› must be one of the words ON or OFF.

Appendix 3: Mail Terminal Program MAIL232.COM

This program allows the PCW8256 and CPS8256 to be used as a simple terminal and as a medium to transfer files between computers.

The program is supplied on Side 1 of the system discs, although it is not part of LocoScript. It can only be used after CP/M has been loaded from Side 2.

The Mail Terminal program provides facilities for changing the baud rate and framing bits, for transferring files, and for attaching as a 'VT52/Z19' terminal.

A number of menus are operated either by the codes generated, by default, by the function keys, up and down cursor keys and the **[+]** key. If any of these keys has been re-defined then the program operation may be modified. The program itself redefines the **[←DEL]** key as ASCII 8 (**[CTRL]H**) known as backspace.

When the program is first entered, a very simple Mail Terminal is emulated, with the following characteristics: 9600 baud TX/RX, 8 data bits, 1 stop bit, NO parity, ONLINE, Hardware handshake OFF, obeying control codes Carriage Return, Line Feed and Backspace only.

Each menu is entered by pressing the relevant function key. Options are toggled by the **[+]** key and menus are left by the **[EXIT]** key (back to Mail Terminal operation) or by the **[ENTER]** key (to initiate file transfer, enter 'VT52/Z19' emulator or Return to CP/M).

f1 menu:

This menu allows the user to set the baud rate and framing bits. Position the large bar cursor using the cursor keys and then select the required parameters by stepping through the available options using the **[+]** key.

Hardware handshaking is not normally appropriate when connected via a modem, but where there is a direct connection it prevents any loss of received characters due to screen operations such as multiple new-lines taking place slower than the subsequent received data.

This shows the menu set up for transmission to a typical electronic mail service.

f3 menu:

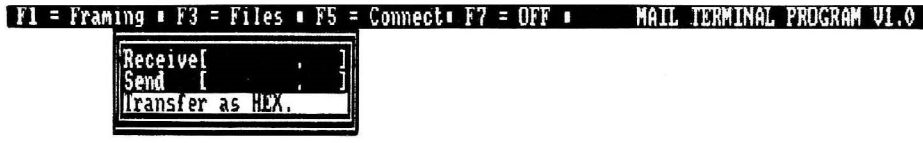
As well as character-based operations the MAIL232 program will permit file transfers. These transfers can either be in plain ASCII (normally 7-bit) or assembled into blocks allowing transfer of HEX. In order to transfer as HEX both the sending and receiving computers must operate the intelligent file transfer protocol shown below. This protocol ensures a level of error detection and correction.

CPC464/664/6128 computers using the AMSTRAD RS232C serial interface support the intelligent file protocol. A listing for sending from other CP/M computers is shown in the main section of this manual.

Transferring as HEX will allow any file to be sent; transferring as ASCII restricts the user to text files. When receiving as ASCII the file is displayed on the screen and the reception is concluded at the discretion of the receiving operator when he presses **[ALT] [STOP]**. When sending as ASCII, any echo from the receiving computer is displayed and the transmission is concluded when the end-of-file is found. The sending computer continues to receive and display until either a carriage return is received, or any key on the keyboard is pressed.

MAIL232, when acting as a receiver, does not echo back to the sending computer. This is because it will often be operating in 1200 baud with a 75 baud return channel, so that any echo could not keep up with the received data. Although the received file is initially stored in RAM buffers, there will come a time when a disc access will possibly cause the loss of a few received characters. Similarly, when sending, a disc access may slightly corrupt any reception of echoed characters.

The filenames may be 'preloaded' into the slots in the **f3** menu as the transfer is executed only when **[ENTER]** is pressed. Pressing **[EXIT]** removes the menu, preserving the filename. The 'Transfer as ASCII/HEX' toggle is operated by pressing **[+]** when the cursor bar is positioned over the relevant entry.



If a HEX transfer is successful then the filename is cleared. If an error is detected then the computer will beep, and the small character cursor will re-appear on the filename. To retry the transfer type **[ENTER]** (after making any necessary corrections to the filename).

The protocol for the intelligent file transfer is as follows:

Sending computer:

(1) Send STX, Listen for ACK.

(2) Send 16 byte filename, 2 byte block number (first block is block 0), 1 byte block length (0-128), data (0-128 bytes), 2 byte checksum (sum of all data bytes). Zero block length means end-of-file.

(3) Listen for ETX, ACK or NAK

(4) If ETX then abort, else if NAK, goto (2) retrying same block; else if ACK goto (2) sending next block, or finish if last block.

Receiving Computer:

(1) Listen for STX, respond with ACK. NOTE - This means that the receiving program should be started before the transmitting program, otherwise the initial STX might be missed.

(2) Receive filename, block number, block length, data, checksum.

(3) Check for same filename as block 1, and consecutive block numbering, if error then send ETX and abort.

(4) Check for hardware error or checksum error; if OK send ACK; else send NAK.

(5) Check block length; if zero finish, else goto (2).

(For your reference)

STX = [CTRL]B = ASCII 2
ETX = [CTRL]C = ASCII 3
ACK = [CTRL]F = ASCII 6
NAK = [CTRL]U = ASCII 21

f5 menu:

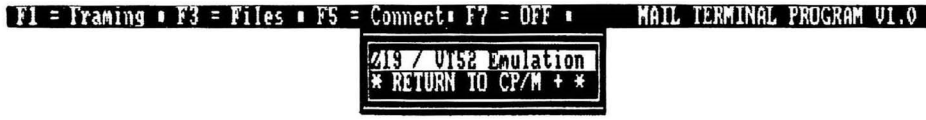
This simply toggles the ONLINE/OFFLINE status of the Mail Terminal simple terminal emulator. Select using **[+]**, execute using **[EXIT]**.

f7 menu:

Two options are available here: Z19/VT52 emulation or 'Exit'ing to CP/M, select using the cursor keys, execute using **[ENTER]**. The emulation mode is 'exit'ed by pressing **[ALT] [STOP]**.

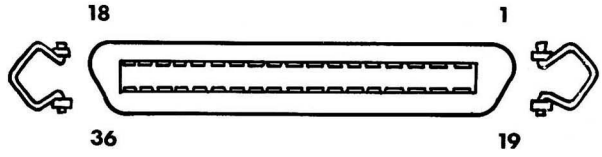
NOTE. In the emulation mode the program does not attempt to simulate the named terminals PRECISELY, and it is not intended that the emulator be regarded as a 100% intelligent terminal replacement. However, most of the common screen control codes are obeyed, allowing operation of typical utility and data processing programs.

The emulation mode is particularly useful when connected without hardware handshaking (e.g: when operating with a modem) and the simple Mail Terminal is unable to keep up with the data.



Appendix 4: Hardware

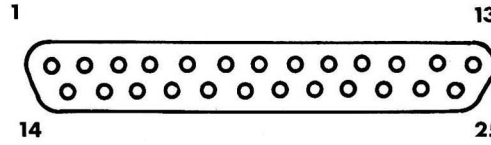
Parallel/Centronics Socket



PIN 1	$\overline{\text{STROBE}}$	PIN 19	GND
PIN 2	D0	PIN 20	GND
PIN 3	D1	PIN 21	GND
PIN 4	D2	PIN 22	GND
PIN 5	D3	PIN 23	GND
PIN 6	D4	PIN 24	GND
PIN 7	D5	PIN 25	GND
PIN 8	D6	PIN 26	GND
PIN 9	D7	PIN 27	GND
PIN 11	BUSY	PIN 28	GND
PIN 16	GND	PIN 33	GND

ALL OTHER PINS NC

Serial/RS232C Plug



PIN 2	DATA OUT (TXD)
PIN 3	DATA IN (RXD)
PIN 4	RTS OUT
PIN 5	CTS IN
PIN 7	GND
PIN 8	DCD IN
PIN 20	DTR
PIN 22	RING IND. IN

ALL OTHER PINS NC