PHILIPS

SERIAL INTERFACE

NMS1210
NMS1211
NMS1212

USER GUIDE

New Media Systems

TABLE OF CONTENTS

INTRODUCTION

The Philips serial communication interface cartridge is specially designed for use with MSX-computers. This interface allows your MSX computer to communicate with other computers, devices or peripherals that have an RS232-C interface.

Terminal programs on the enclosed 3.5" diskette enable you to transfer files under the KERMIT protocol and the XMODEM protocol

The diskette also contains a BASIC interpreter and BIOS extension for the interface, which enables you to write and run RS232-C control programs in BASIC.

An RS232-C device driver is also provided for advanced programming in Assembler.

In this user guide you will find a product description, an introduction to the application programs, a hardware description and other relevant information.
If you carefully read this user guide, you will soon be able to use the RS232-C interface to its full extent. Your RS232-C interface cartridge will greatly extend the capabilities of your MSX computer.

1. PRODUCT DESCRIPTION

Philips produces 3 versions of the MSX serial comunication interface:

- NMS 1210
- NMS 1211
- NMS 1212 (for professional users)

The NMS 1210, NMS 1211 and NMS 1212 serial interface cartridges are MSX-compatible and allow 2-way communication between MSX computers and any other computer, device or peripheral equipment with a serial interface.
The three versions are configured as follows:

- NMS 1210: a single RS232-C interface including a cable with a 25-pin D-type connector
- NMS 1211: a dual RS232-C interface including two cables and two 25-pin D-type connectors
- NMS 1212: a RS232-C and RS422 interface including cables with a 25-pin and 9-pin D-type connector
  In addition to the RS232-C communication facilities, the RS422 option allows an MSX computer to be coupled to a Local Area Network (LAN).

The interface cartridge plugs directly into one of the MSX computer's external expansion slots and has interface cables about 0.75 metres in length. These cables are equipped with the connectors as described above.

The Philips communication interfaces are supplied with a diskette, containing the following programs:

- A terminal program for MSX-1 and MSX-2 computers with a file transfer protocol based on KERMIT
- A terminal program for MSX-2 computers with a file transfer protocol based on XMODEM
- A BASIC interpreter and BIOS extension, to allow MSX users to write and run communication programs under BASIC
- A device driver, to enable advanced programmers to write communication programs in Assembler

1.1 The product contents

Your Philips communication interface cartridge package contains the following items:

- The communication interface cartridge, with interface cable(s)
- A 3.5" diskette with various software programs and user instruction document files
- A general user guide

2. INSTALLING THE COMMUNICATION INTERFACE

Follow these steps to install the communication interface:

1. Switch off your computer to avoid accidental damage.

2. Insert the interface cartridge into one of the cartridge slots of your computer.

3. Connect the interface cable to the other computer or device.
   In general, the male interface cartridge plug will directly connect to modems and other devices known as DCE (Data Computer Equipment), which are fitted with female plugs.
   For further details, see Appendix C.

4. Now switch your MSX computer on.

   The MSX logo appears on the sceen.
   Wait a few moments and the following message appears on the screen:

   MSX DOS version etc.

It is advisable to make a backup copy of the program diskette before you start working with your communication inmterface. Store the original diskette in a safe place and continue working with the copy you have made.

3. PROGRAM REVIEW

In addition to the actual programs the diskette contains document files, with important instructions for each of these programs. You can recognise these files with the extension .DOC.

The diskette contains several groups of files as follows:

- MSXDOS.SYS
- COMMAND.COM

These are the MSX-DOS files and need no further explanation.

- KERMITR.COM
- KERMITR.INI
- KERMITR.DOC (document file)

These files refer to the KERMITR terminal program

- XMODEM.COM
- XMODEM.INI
- XMODEM.DOC (document file)

These files refer to the XMODEM terminal program.

- RS
- AUTOEXEC.BAS
- XBASIC.BAT
- RSUSER.DOC (document file)
- RSBASIC.DOC (reference file)
- RSBIOS.DOC (reference file)

These files refer to the BASIC interpreter and to the Extended BIOS Call routines.

- RMAIN.REL
- RBLOCK.REL
- RTIMER.REL
- RRAM.MAC
- RSDRIVER.DOC (document file)

These files refer to the RS232-C driver software

In general, you can start a program by typing the program name and press <RETURN>.

To display a document file with instructions concerning a specific program you should type:

TYPE <filename of the document file> <RETURN>

To stop the text on your screen from scrolling , press <CTRL + s>.
You start the scrolling again by pressing <CTRL + q>
When you want a printout, press <CTRL + p> before you give a TYPE command.
To stop the printing, press <CTRL + n>.

3.1 KERMITR terminal program

You can run the KERMITR program on an MSX-1 and an MSX-2 computer.
This terminal program is based on the KERMIT file transfer protocol. This protocol sets certain minimal standards for controlled transmission of files via fixed connections or via a modem and telephone lines.
The features of this program are fully described in the KERMITR document file.

To start the program type:

KERMITR <RETURN>

You can display the KERMITR document file by typing:

TYPE KERMITR.DOC <RETURN>

If you want to change the default initialisation, you must do the following entries:

A. Type:
COPY CON KERMITR.INI

B. Type all the commands that you want to include in the file. After entering a command you should press <RETURN>.

C. When all the commands are entered, press <CTRL + z>, next press <RETURN>.

D. The file will be written on the diskette as KERMITR.INI. When ready, the MSX-DOS prompt will appear.

3.2 XMODEM terminal program

You can run the XMODEM program on any MSX-2 computer.
This terminal program is based on the XMODEM protocol. This protocol sets minimal standards for controlled transmission of files via fixed connections or via a modem and telephone lines.
The features of this program are fully described in the XMODEM document file.

To start the XMODEM program type:

XMODEM <RETURN>

You can display the XMODEM document file by typing:

TYPE XMODEM.DOC <RETURN>

3.3 RS232-C BASIC interpreter extension

You can run the RS232-C BASIC interpreter extension on any 64K user RAM MSX-1 or any MSX-2 computer.
This BASIC interpreter allows you to write and run communication programs under BASIC.You can use the statements described in the "Philips MSX2-BASIC and MSX-DOS" reference manual as well as some extra statements described in the RSBASIC document file.

You can load the BASIC interpreter from MSX-DOS by typing:

XBASIC <RETURN>

You can display the RSBASIC document file by typing:

TYPE RSBASIC.DOC <RETURN>

There is an additional document file, containing general information on the use of MSX BASIC.

You can display this document file by typing:

TYPE RSUSER.DOC <RETURN>

The RS232-C BIOS can be called by application programs using the "Extended BIOS Call" routines. This is described in the RSBIOS document file.

You can display the RSBIOS document file by typing:

TYPE RSBIOS.DOC <RETURN>

3.4 RS232-C/RS422 device driver

The device driver of the RS232-C interface is contained in four files:

-RMAIN.REL
-RTIMER.REL
-RBLOCK.REL
-RRAM.MAC

RRAM.MAC needs assembly to RRAM.REL before use. Only RMAIN.REL and RRAM.REL have to be linked with an application program. The other two files are optional and cannot be linked without RMAIN.REL and RRAM.REL.

-RMAIN.REL contains the main part of the RS232-C device driver
-RTIMER.REL contains the watchdog timer routines
-RBLOCK.REL contains the block transfer routines
-RRAM.MAC has been supplied to give the programmer the possibility to assign the RAM addresses the driver should use for its work areas, and control of buffer parameters.

You can examine this source file by typing:

TYPE RRAM.MAC <RETURN>

Another document file has been included with a description of the device driver. This is a technical description of the usage of the device driver. The main part of this file describes the routines, With the information stored in this file, an advanced programmer can write communication programs in Assembler.

You can display this file by typing:

TYPE RSDRIVER.DOC <RETURN>

APPENDIX A.
TECHNICAL HARDWARE DESCRIPTION

This section describes the hardware of the serial interface for MSX-1 and MSX-2 computers.

Technical data

Interface cartridge bus specifications

Pin	Name	Description
7	WAIT*	Wait signal
8	INT*	Interrupt request signal
9	M1	Opcode fetch signal cycle of CPU
10	BUSDIR	Direction of the external data
11	IORQ	I/O request signal
13	WR	Write signal
14	RD	Read signal
15	RESET	System reset signal
21	A7	Address bit 7
22	A6	Address bit 6
27	A1	Address bit 1
28	A0	Address bit 0
29	A3	Address bit 3
30	A2	Address bit 2
31	A5	Address bit 5
32	A4	Address bit 4
33	D1	Data bit 1
34	D0	Data bit 0
35	D3	Data bit 3
36	D2	Data bit 2
37	D5	Data bit 5
38	D4	Data bit 4
39	D7	Data bit 7
40	D6	Data bit 6
41	GND	Ground
42	CLOCK	CPU clock, 3.579 MHz
43	GND	Ground
44	SW1	Detect insert
45	+5 V	+5 V power supply
46	SW2	Remove for protection
47	+5 V	+5 V power supply
48	+12 V	+12 V power supply
50	-12 V	-12 V power supply

* = open collector output

Note: WAIT signals generated by both the SCC channels are connected to the MSX bus to permit high-speed block data transmission/reception synchronization. This feature is software-selectable, and is restricted to certain baud rates (and possibly certain computers).

Tx/Rx clock rates

The Tx clock rates for both channels are set by the SCC, while the Rx clock rates are set by the PIT. Timer 0 controls Rx on channel-A, and Timer 1 controls Rx on channel-B.
Note that Timer 1 may also affect the interval timer.
External Tx/Rx clocks are software-selectable, together with the use of hardware jumpers on the interface board. PCLK is driven by the crystal, and is within specification for interfacing to the MSX bus (max. 2.9% error).

Asynchronous and synchronous communications are possible for a wide range of speeds between 50 bits/s and 38.4 Kbits/s for any 2 channels, e.g.: 1 x EIA RS232-C + 1 x EIA RS422.

The following bits/s rates are set by a 3.6864 MHz chrystal:

Baud rates	Error
38.4 Kbits/s	-
19.2 Kbits/s	-
9.6 Kbits/s	-
7.2 Kbits/s	-
4.8 Kbits/s	-
3.6 Kbits/s	-
2.4 Kbits/s	-
2.0 Kbits/s	0.175%
1.8 Kbits/s	-
1.2 Kbits/s	-
1050 bits/s	-0.260%
600 bits/s	-
300 bits/s	-
150 bits/s	-
134.5 bits/s	0.059%
110 bits/s	-
75 bits/s	-0.069%
50 bits/s	-

Fully independent speeds are possible on all transmit and receive lines, e.g. 1 x RS232-C at 75/1200 and 1 x RS422 at 9600/9600.

Interrupts

Interrupts are generated by the SCC and PIT as programmed.

I/O port addresses

The I/O port addresses are assigned as follows:

Misc. status level	27	37 Hex.
Channel B Command	28	38
Channel B Data	29	39
Channel A Command	2A	3A
Channel A Data	2B	3B
Internal Timer 0	2C	3C
Internal Timer 1	2D	3D
Internal Timer 2	2E	3E
Mode register	2F	3F

(Range is selectable 2x or 3x; 3x is default)

Note: port addresses are assigned to be compatible with the I/O port addressing of the NMS 1255 MSX interface/modem cartridge.

Hardware interface settings

Depending on the interface version, up to 5 hardware jumpers are provided on the interface board for the flexible setting of I/O port addresses and Tx/Rx clock assignments.
The locations of these jumper switches are shown in the PCB layout.

RS422 operating modes

Two modes of RS422 operation are possible: multidrop (Mode A) and half duplex (Mode B).

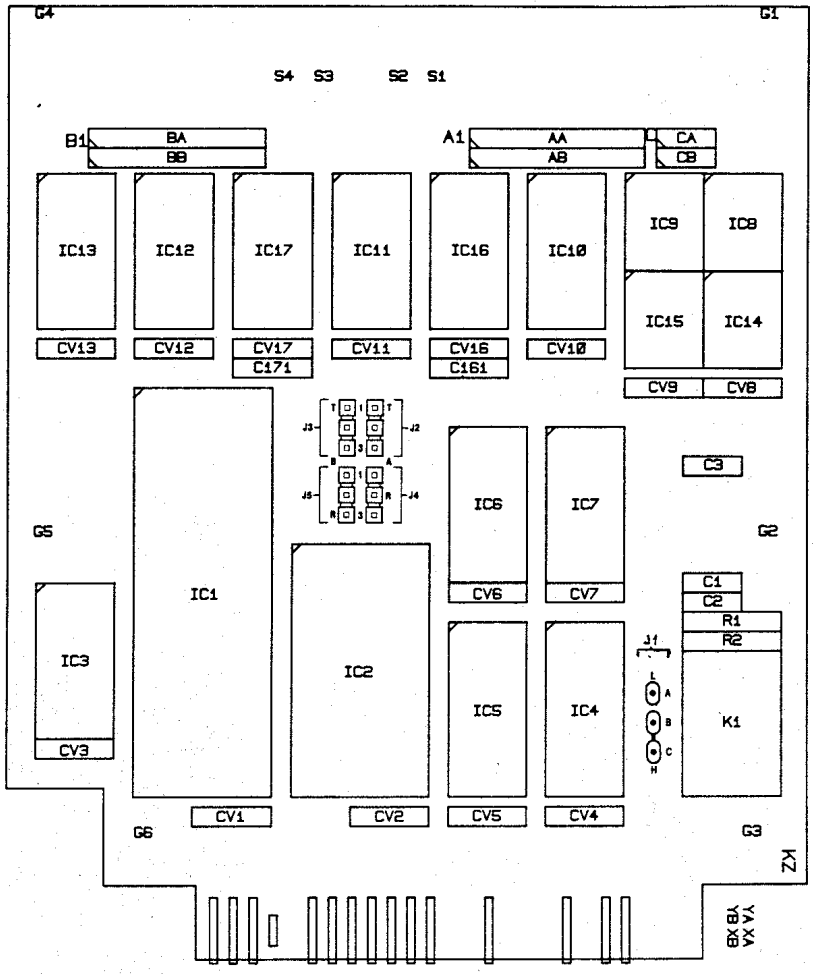
1. Multidrop (Mode A)
   In the multidrop mode, full duplex operation is possible over 2 pairs of balanced wires

2. Half duplex (Mode B)
   This mode is compatible with the Philips Webwork network system. Half duplex-operation with clock is possible over 2 pairs of balanced wires. The clock signal is generated by the transmitting station. The RS422 transceivers are enabled by the $\overline{\text{RTS}}$ signal, which determines the direction of data flow.

   For operation in this mode the RS422 plug must be re-pinned as follows:

Pin 1	NW DAT	Brown
Pin 2	NC	
Pin 3	Signal gnd.	Black
Pin 4	NC	
Pin 5	NW CLK	Red
Pin 6	NW DAT	Orange
Pin 7	NC	
Pin 8	NC	
Pin 9	NW CLK	Yellow

PCB layout



IC installation configuration

	RS232-C channel A	RS232-C channel B	RS422 mode A	RS422 mode B
Single RS232-C	10,11,16			
Dual	10,11,16	12,13,17		
RS232-C/RS422		12,13,17	8,9	(14,15)

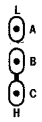
To operate in mode B the transceivers on position nr. 1 and 9 have to be moved to IC position nr. 14 and 15.

JUMPER OPTIONS

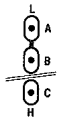
1. I/O port addresses

The 9 I/O ports of the interface cartridge can be mapped to alternative addresses, according to the setting of the tracks connection J1 on the interface board.

Factory setting uses addresses 37 (Hex.) to 3F (Hex.).

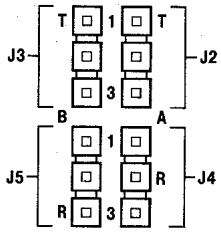


Alternative setting uses addresses 27 (Hex.) to 2F (Hex.).
To select these port addresses, the track between points B and C must be cut, and a jumper soldered to connect A and B.

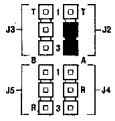


2. Transmission clock

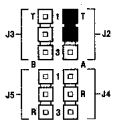
The transmisson clock signal (TxC) for a channel can either be provided by the serial interface itself, or by an external clock source. The selection is made by jumper J2 (for channel-A RS232-C of RS422) or J3 (for channel-B RS232-C, if provided).



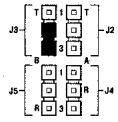
Factory setting: the serial interface provides TxC (signal DA-pin 24) on channel-A RS-232-C or RS422 TxC (mode B).



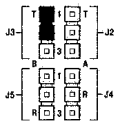
An external clock source is provided for TxC (signal DB-pin 15) on channel-A RS232-C).



Factory setting channel-B is installed: the serial interface provided TxC (signal DA-pin 24) on channel-B RS232-C, if provided.



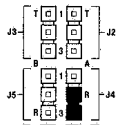
An external clock source is provided for TxC (signal DB-pin 15) on channel-B RS232-C, if provided.



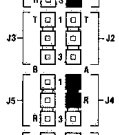
3. Reception clock

The reception clock signal (RxC) for a channel can either be provided by the serial interface itself, or by an external clock source. The selection is made by jumper J4 for channel-A RS232-C or RS422, or J5 for channel-B RS232-C, if provided.

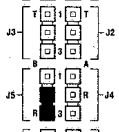
Factory setting: the serial interface provides RxC on channel-A RS232-C or RS422 RxC.



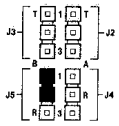
An external clock source is provided for RxC (signal DD-pin 17) on channel-A RS232-C or RS422 RxC (mode B).



Factory setting if channel-B is installed: the serial interface provides RxC on channel-B RS232-C.



An external clock source is provided for RxC (signal DD-pin 17) on channel-B RS232-C.



Electrical specification RS232-C

The electrical specification is in accordance with RS232-C/V28.

The main characteristics are:

- unbalanced signaling (common return).
- only one drive/one receiver per circuit.
- length 50 feet maximum.
- 20KB data rate maximum.
- slew rate 30V/uS maximum.

The slew rate (and noise rejection) is achieved using an external capacitor with bi-polar drivers/receivers (typically 470pF). C-MOS drivers/receivers have built-in slew-rate and noise rejection (14C88/14C89). The NMS interface is not fitted with these capacitors.

Higher data rates can be achieved in good conditions,(e.g.short cable length) but are outside the specification RS-232 (section 2)/V28.

Pin-out of 25-pin D-type RS-232-C male connectors (DTE).

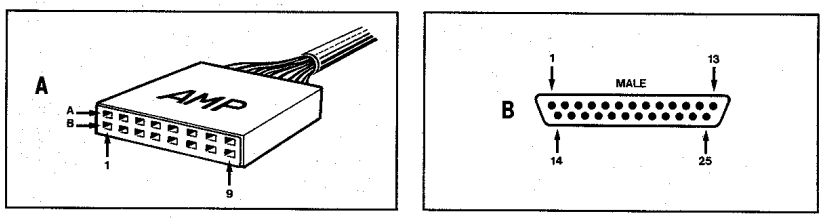
A.Pin no. signal on 20-pin modu IV		B. Pin no. signal on 25-in D shell		Full signal name	
A1	to	1 - Prot. Gnd.	AA	Protective ground shield	
B2	to	2 - TD	BA	Transmitted data	to DCE
A2	to	3 - RD	BB	Received data	to DTE
B8	to	4 - RTS	CA	Request to send	to DCE
A3	to	5 - CTS	CB	Clear to send	to DTE
A4	to	6 - DSR	CC	Data set ready	to DTE
A5	to	7 - S Gnd.	AB	Signal ground	
A6	to	8 - CD	CF	Received carrier detect	to DTE
B5	to	15 - TxC	DB	Transmission signal element timing	to DTE
B1	to	17 - RxC	DD	Receiver signal element timing	to DTE
B6	to	20 - DTR	CD	Data terminal ready	to DCE
B9	to	22 - RI	CE	Ring indicator	to DTE
B4	to	24 - TxC	DA	Transmit signal element timing	to DCE

The interface provides the possibility of using the alternate channel as secondary to the main (and,and). A different extended interface cable is then required.

In this case, the above pin-out will be extended as follows:

A7	to	12 - SCD	SCF	(Secondary received carrier detect)	to DTE
A8	to	13 - SCTS	SCB	(Secondary clear to send)	to DTE
A9	to	14 - STD	SBA	(Secondary transmitted data)	to DCE
B3	to	16 - SRD	SBB	(Secondary received data)	to DTE
B7	to	19 - SRTS	SCA	(Secondary ready to send)	to DCE

See Appendix C for further connection details.



Electrical specification RS422

The electrical specification is in accordance with RS422/V11.

The main characteristics are:

- differential signalling (two wire).
- single driver/multiple receivers per circuit.
- length : 40 feet 400 feet 4000 feet
- data rate: 10 Mb 1 Mb 100 Kb

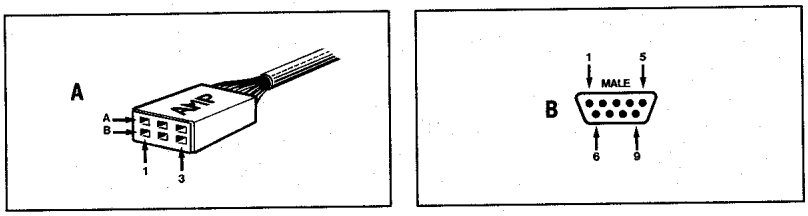
Slew rate and noise rejection are built into the transceivers. However, impedance matching at the receiver is usually required, and depends on the cable length. A typical value would be 100R. The NMS serial interface is not fitted with these resistors.

Multiple drivers with tri-state feature can be used (the NMS interface Webwork mode of operation), but care must be taken to ensure the bus stays within a limited common-mode voltage range (- 0,25V to 6V). If a contention occurs, damage may be caused to both receivers and drivers.

Pin-out of 9-pin D-type RS422 connector

A. Pin no. signal on 6-pin modu IV		B. Pin no. signal on 9-pin D shell		Full signal name
A3	to	1 - Prot. Gnd.		Protective ground (shield)
A1	to	3 - TD	SD	Send data
A2	to	4 - RD	RD	Receive data
B1	to	7 - TD return	SD*	Send data return
B2	to	8 - RD return	RD*	Receive data return
B3	to	9 - Signal ground	SG	Signal ground

*Note: - SD and RD are category 1 circuits
- RS422 is subset of RS449, which is also compatible with RS232-C.



APPENDIX B

Technical data hardware

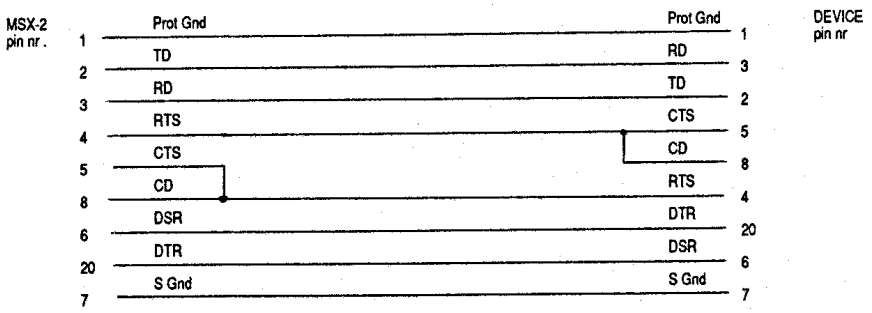
External interface	NMS 1210	1x RS232-C/V28
	NMS 1211	2x RS232-C/V28
	NMS 1212	1x RS232-C/V28 1x RS422/V11
Main IC's		Z 8530 SCC I 8253 CTC
Asynchronous mode		start/stop asynchronous high speed block transfer
Communication method		full/half duplex
Character transmission		programmable format
Standard options		- Xon/Xoff control - CTS/RTS - stop bit length (1 /1.5 /2) - parity (odd/even/ignore) - character length (5/ 6/ 7/ 8 bits) - independent baud rates for channel A and B, Rx and TX (50 - 38.400 baud) - start/stop block - baud rate <= 250KB - dual I/O port mapping - break detection/generation
Power requirements		Supplied by MSX computer
Cable length		Approx. 0.75 m.
Interface plug		25-pin D-type connector

APPENDIX C

RS232-C connection to other equipment

The interface cartridge is fitted with a male plug on a lead configured as DTE (Data Terminal Equipment). This permits direct connection to modems and other equipment configured in the complementary way as DCE (Data Computer Equipment) fitted with female sockets

If connecting to another DTE, the signals need to be complemented by "crossing- over", e.g.: TD is connected to RD, RD is connected to TD, etc.
As both the MSX interface and the other equipment have male plugs, a cross-over adapter fitted at both ends with female sockets should be used. To make one yourself, the wiring of the pins is as follows:



Many other wiring schemes are possible; study the other equipment's handbook carefully.

The principal circuits used for RS232-C/V28 are as follows (always connect pin 1 to 1 and pin 7 to 7 !)

			V24 circuit#	MSX pin#	other pin#
Data transmission:					
	Transmitted Data		103	2	3
	Received Data	IN	104	3	2
XON/XOFF protocol:					
	-no further connection required				
CTS/RTS Handshake:					
	RTS		105	4	5+8
	CTS	IN	106	5	4
	CD		109	8	4
DTR/DSR Handshake:					
	DSR	IN	107	6	20
	DTR		108.2	20	6

Signals marked IN are inputs.

The CTS signal can be NUL wired if required by connecting pins 4, 5and 8 together.

The DSR signal can be NUL wired if required by connecting pins 6 and 20 together

All signals are under software control in the case of the MSX. Their actual use therefore depends on the programmer or application program. Usually, different wiring is only required where the other equipment uses the circuits in a non-standard way.