

**MSX RS232C software specification**

ASCII Microsoft  
July 4th, 1984

I N D E X

- 1.0 STATEMENTS
  - 1.1 Setup Communication Parameters
  - 1.2 Open and Close Communication Channels
  - 1.3 Sequential Input and Output
  - 1.4 Program Load/Save statements
  - 1.5 Event trap control statements
  - 1.6 Miscellaneous Control statements
- 2.0 FUNCTIONS
- 3.0 TERMINAL MODE
- 4.0 HELP FUNCTION

APPENDIX MISCELLANEOUS INFORMATIONS

- A.1 THE BEHAVIOR OF CONTROL SIGNALS
- A.2 HANDLING OF EOF

This document describes the specifications of the BASIC language support for the RS-232C communication interface on MSX home personal computer.

## 1.0 STATEMENTS

### 1.1 Setup Communication Parameters

#### CALL COMINI

```
[[[<string exp>][,<Rx baud rate>][,<Tx baud rate>][,<time out>]]]]]
```

; To initialize RS232C port with specified parameter. <string exp> is a string that specifies the channel control parameters. See detailed description below.

#### BAUD RATE

It is possible to set different baud rate for transmitter and receiver. The possible value for the baud rate are as follows:

50, 75, 110, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200,  
9600, 19200

When only the receiver baud rate is specified, the baud rate for transmitter takes same speed to the receiver. When only the transmitter baud rate is specified, the baud rate for receiver takes default value.

When negative value is specified, its absolute value is written to i8253 timer/counter directly.

#### TIME OUT

The RS232C driver waits for CTS (Clear To Send) turn on or/and XON is received when the character is send. The driver will generates time out error when wait for it, if specified time passed. Its time is specified this value at second. If 0 specified then the driver doesn't generate time out error and waits forever.

## STRING FIELDS

```
"[0:][8[N[1[X[H[N[N[N]]]]]]]"
```

	Channel No.
+-----	
	When the system has more than one channel, this parameter specifies the channel number. May be omitted if the system has only one channel. The default value is 0.
+-----	
	Data length
+-----	
	"5" - 5 bits
	"6" - 6 bits
	"7" - 7 bits
	"8" - 8 bits
+-----	
	Parity flag
+-----	
	"E" - Even parity
	"O" - Odd parity
	"I" - Ignore (Illegal when data length is 8 bits)
	"N" - No parity
+-----	
	Stop bits length
+-----	
	"1" - 1 bit
	"2" - 1.5 bits
	"3" - 2 bits
+-----	
	XON/XOFF control
+-----	
	"X" - do control
	"N" - do not control
+-----	
	CTS-RTS hand-shake
+-----	
	"H" - do hand-shake
	"N" - do not hand-shake
+-----	
	Insert Line Feed to buffer when Carriage Return is received.
+-----	
	"A" - insert Line Feed to buffer
	"N" - do not insert
+-----	
	Send Line Feed after Carriage Return sent.
+-----	
	"A" - do not send Line Feed
	"N" - always send Line Feed
+-----	
	Shift-in/Shift-out control. Illegal when data length is other than 7 bits.
+-----	
	"S" - do control
	"N" - do not control

## Example:

```
CALL COMINI ("0:7E1XHNNN",600,1200,30)
CALL COMINI ("8N1",9600)
```

The default value for those switches are as follows:

```
"0:8E3XHNNN",1200,1200,0
```

Note that no previous value are taken as default. If omitted, above value is assumed always.

## 1.2 Open And Close Communication Channels

OPEN "COM[n]:" [FOR <mode>] AS [#] <file number>

; This statement opens the RS-232C channel for further processing. That is, allocate a buffer for I/O and set the mode that will be used with the buffer. Also, RTS signal will be activated.

<mode> is one of the following:

OUTPUT : Specifies sequential output mode  
INPUT : Specifies sequential input mode

If <mode> clause is not specified, the channel is able to be accessed both input and output and no EOF character handling is performed.

<file number> is an integer expression whose value is between one and the maximum number of files specified in a MAXFILES= statement.

<file number> is the number that is associated with the file for as long as it is OPEN and is used by other I/O statements to refer to the file.

An OPEN must be executed before any I/O may take place to the file using any of the following statements, or any statement or function requiring a file number:

PRINT #, PRINT # USING  
INPUT #, LINE INPUT #  
INPUT\$

Example:

OPEN "COM0:" AS #1

### NOTE

Random access to RS-232C channel is illegal. Only sequential accesses are logically possible.

CLOSE [[#]<file number>[,<file number>]]

; To close the channel and releases the buffer associated with it. If no <file number>'s are specified, all open channels are closed.

If the channel was opened in output mode, the EOF character is sent.

### 1.3 Sequential Input And Output

After the channel is opened in input mode or file mode (open without <mode> clause), characters from communication channel can be input by following statement in sequential manner.

```
INPUT #n
LINE INPUT #n
INPUT$(#n,m)
```

After the channel is opened in output mode or file mode (open without <mode> clause), characters can be output to the communication channel by following statement in sequential manner.

```
PRINT #n
PRINT #n USING
```

Refer to language reference manual for details of those statements.

### 1.4 Program Load/save Statements

```
SAVE "COM[<n>:]"[,A]
; To send a BASIC program to the communication channel. Control-Z
is treated as end-of-file. The program is sent in ASCII form
regardless the optional parameter "A". No file name is allowed.
```

```
LOAD "COM[<n>:]"
; To load a BASIC program from the channel.
```

LOAD closes all open files and deletes the current program from memory. However, with the "R" option, all data files remain OPEN and execute the loaded program. Control-Z is treated as end-of-file.

```
MERGE "COM[<n>:]"
; To merge the lines from an ASCII program which is received through
the communication channel into the program currently in memory.
```

If any lines in the program being merged have the same line number as lines in the program in memory, the lines from the channel will replace the corresponding lines in memory. Control-Z is treated as end-of-transmission.

After the MERGE command, the MERGED program resides in memory, and BASIC returns to command level.

```
RUN "COM[<n>:]"[,R]
; To load a program from channel into memory and run it.
RUN closes all open files and deletes the current contents of
memory before loading the designated program. However, with
the "R" option, all data files remain OPEN.
```

### 1.5 Event Trap Control Statements

CALL COMON("[<n>:]")  
; To enable event trapping caused by incoming character from the communication channel.

CALL COMOFF("[<n>:]")  
; To disable event trapping caused by incoming character from the communication channel. The communication buffer is flushed.

CALL COMSTOP("[<n>:]")  
; To suspend event trapping caused by incoming character from the communication channel.

CALL COM ([<n>:],GOSUB <line number>)  
; To set up a line numbers for BASIC to trap to when character is received at the communication channel.

When the trap occurs an automatic CALL COMSTOP is executed so receive traps can never take place. The RETURN from the trap routine will automatically do a CALL COMON unless an explicit CALL COMOFF has been performed inside the trap routine.

Event trapping does not take place when BASIC is not executing a program. When an error trap (resulting from an ON ERROR statement) takes place this automatically disables all trapping (including ERROR, STRIG, STOP, SPRITE, INTERVAL and KEY).

### 1.6 Miscellaneous Control Statements

An OPEN must be executed before any one of following statements may executed. The default channel number is 0 for all following statements.

CALL COMBREAK("[<n>:"],<expression>)  
; To send break characters specified by <expression> to the channel specified by <n>. The range of the <expression> should be between 3 and 32767.

CALL COMDTR("[<n>:"],<expression>)  
; To turn off DTR signal when the <expression> is zero, otherwise turn on the DTR signal.

CALL COMSTAT("[<n>:"],<name of variable>)  
; To read the status of the communication channel. The status returned by the hardware is assigned to the variable. The assignment of each bits are as follows:

BIT NO.	Description
15	Buffer over flow error 0 - no buffer over flow 1 - buffer over flow
14	Time out error ( TMENBT ) 0 - no time out error occurred 1 - time out error occurred
13	Framing error 0 - no framing error occurred 1 - framing error occurred
12	Over run error 0 - no over run error occurred 1 - over run error occurred
11	Parity error 0 - the character hasn't parity error 1 - the character has parity error
10	Control break key was pressed ( BRONBT ) 0 - control break key wasn't pressed 1 - control break key was pressed
9	Not used, reserved
8	Not used, reserved
7	Clear To Send 0 - false 1 - true
6	Timer/counter output-2 0 - timer/counter output-2 is negated 1 - timer/counter output-2 is asserted
5	Not used, reserved
4	Not used, reserved
3	Data Set Ready 0 - false 1 - true
2	break detect 0 - not yet detect 1 - detect
1	Ring Indicator 0 - false 1 - true
0	Carrier Detect 0 - false 1 - true

## 2.0 FUNCTIONS

EOF(<file number>)  
; Return -1 (true) if the EOF character is received. Otherwise, returns 0. Use EOF to test for end-of-transmission while INPUTing, to avoid 'Input past end' errors.

LOC(<file number>)  
; Return the number of characters received in the communication buffer. The size of the communication buffer is 255 characters.

LOF(<file number>)  
; Return the number of free spaces left in the communication buffer.

## 3.0 TERMINAL MODE

CALL COMTERM("<n>:")  
; To enter a terminal emulator mode. The channel should be closed when this statement is invoked. Following function keys have special function in the terminal mode as described below.

F-6 : Turn on/off literal mode. In the literal mode, control characters are displayed with upper allow and the character offset by 40H. For example, character whose code is 01H is displayed as "A".  
Initial mode: Literal mode off

F-7 : Toggle Half/Full duplex mode. In the Half duplex mode, characters keyed in are echoed to the screen as well as sent to the communication channel.  
Initial mode: Full duplex

F-8 : Turn on/off printer echo. When the printer echo is on, all the characters sent to the screen are also sent to the printer.  
Initial mode: Printer echo off

## 4.0 HELP FUNCTION (OPTIONAL FEATURE)

```
CALL COMHELP[(<n>:)]
```

```
; To print out a brief description for the parameters set by  
COMINI
```

```
statement on the screen as follows.
```

```
Initialize statement options
```

```
CALL COMINI ("  
<device# {0,1,2...9}>:  
<character length {5,6,7,8}>  
<parity {E,O,I,N}>  
<stop bits {1,2,3}>  
<XON/XOFF {X,N}>  
<CTS hand-shake {H,N}>  
<auto LF on receive {A,N}>  
<auto LF on transmit {A,N}>  
<SI/SO {S,N}>"  
,<receiver baud rate>  
,<transmitter baud rate>  
)
```

```
Default:
```

```
CALL COMINI("0:8N1XHNNN"  
,1200,1200,0)
```

APPENDIX A  
MISCELLANEOUS INFORMATIONS

A.1 THE BEHAVIOR OF CONTROL SIGNALS

	RESET	COMINI	OPEN	CLOSE
RTS - inactive		no effect	active	inactive
DTR - active		active	no effect	no effect

RTS signal is affected in following cases:

1. OPEN statement is executed - activated.
2. CLOSE statement is executed - inactivated.
3. The rest of the communication buffer is less than 16 byte and CTS-RTS handshake is enabled - inactivated.
4. When it is inactive and the rest of the communication buffer becomes more than byte and CTS-RTS handshake is enabled - activated.

DTR signal is affected by CALL COMDTR and CALL COMINI statements.

A.2 HANDLING OF EOF

EOF is transmitted when CLOSE is executed when the open mode was output.