Advanced
Serial Port
Monitor

Watch It!

1

1.1

(RS232, COM port, UART)

IBM -   Windows

:

- (   )
  (   );
- ;
- ,
  ;
- ();
-                          ;
- (,   , . .);
- ;
- ,   .

1.2   (COPY)

.

1.   Advanced Serial Port Monitor   -
.

2.
.

3.   Advanced Serial Port Monitor   ,
.

4.   ,
,
.

5.   Advanced Serial Port Monitor
.   sales@aggsoft.ru

6. Advanced Serial Port Monitor   ,   ,
.

Advanced Serial Port Monitor   .

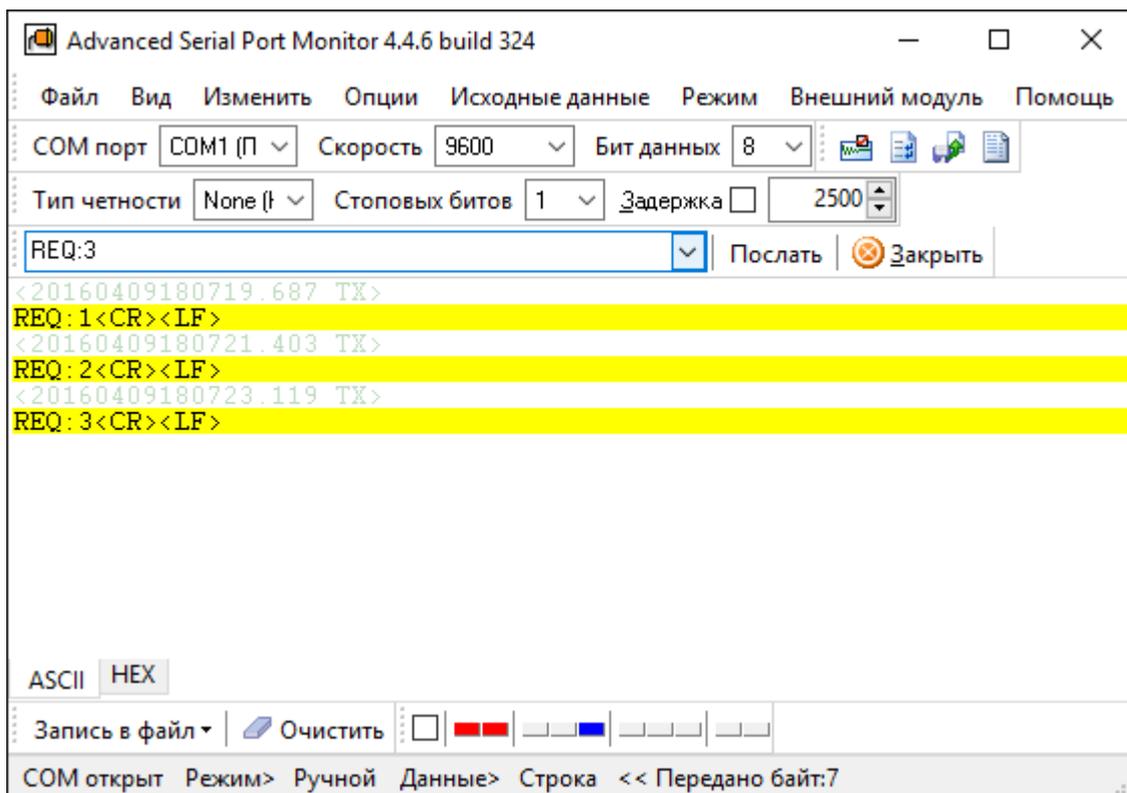7. **Advanced Serial Port Monitor**   **"AS IS".**
                                                            .
                                    .
                                ,

                    .

8.   ,   ,   ,   ,
                    ,   ,   ,   ,   ,
                        ,
            ,   ,   .

                                                .

            ,   ,   .

9.   Advanced Serial Port Monitor   ,
                                                .

10.   -   ,
        ,   ,
        Advanced Serial Port Monitor
                        Advanced Serial Port Monitor.

1.3

                                Shareware.
                            ,
                .

                                        ,
            ,   ,   ,   .
                        ,   1 .

    _____   .
                        ,   .
        ,   ,   .
                            ,   .

                                        ,   ,
            SoftKey.ru   _____
        .

2

2.1

:

- Windows 2000　　　　(　　　　　　　　　　　　　　　).
- Windows Server 2003　　　　(　　　　　　　　　　　　　　　).

　　　　　　　　　　　　　　　　　　　　　　　　　　'
(　　　,　　　),　　　　　　　　　　　　　　　　　.'

2.2

aspmon4.exe.

　　　　　　　　　　　　　　　　　-　　　,　　　　　　　.

　　　　　　　　　　　　　　　　Advanced Serial Port Monitor

　　　　, Advanced Serial Port Monitor　　　　　　　　　　　　　"/Programs Files/
Advanced Serial Port Monitor"　　　　　　　　　,　　　　　　　　　　　.

　　　　　　　　Advanced Serial Port Monitor

　, 　　　　　　　　　　　　　　　____.

3

3.1

　　　'
　.　　　　　　　　　　　'
　　.　_____ 5
　.

　　　　　　　　'　　　　　　　'
(　.1).

.1. .

COM

,
:
1. (, ,
. .);
2. COM , ;
3. " ". , " COM
, . .
COM ,
;
4. "Enter" " "
.

COM .

.
, 19 ,
.
.

5

" ".

.

3.2

.   :

,   ,   .   .

,   COM-

.

.   RS-485,

,   RTS

.

"   ".

(   ) COM-   .

,   ,   -

.

:   -

,   -

Windows.   *(UART -*

)   .   *BIOS/*

Setup.

:   ,   BIOS   UART,

Setup' .   .

(.   .1   ):

- •   ;
- •   COM   ;
- •   ;
- •   ,
  [15];
- •   ,
                                            ;
- •   ;

:

- • _____[16] -   "   ";
- • _____[16] -   ,   ,
              ,   "   "   ;
- • _____[17] -   -
  COM   .   COM   ;
- • _____[18] -
  (   )   ,   .
  .

.   .

.

"_____ 18 ".

- _____ 18 -

.

< #20h,

;

- _____ 19 -

,

.

.

,

.

.

.

.

.

_____ 13

.

,

" |

".   (.1),

.

.1

3.3

"COM |
"

.1.

ASCII

( )

),   "   ".

3.4

5   " | "   (-):

.1.

1.   -

       ,   ;

2.   -   ,
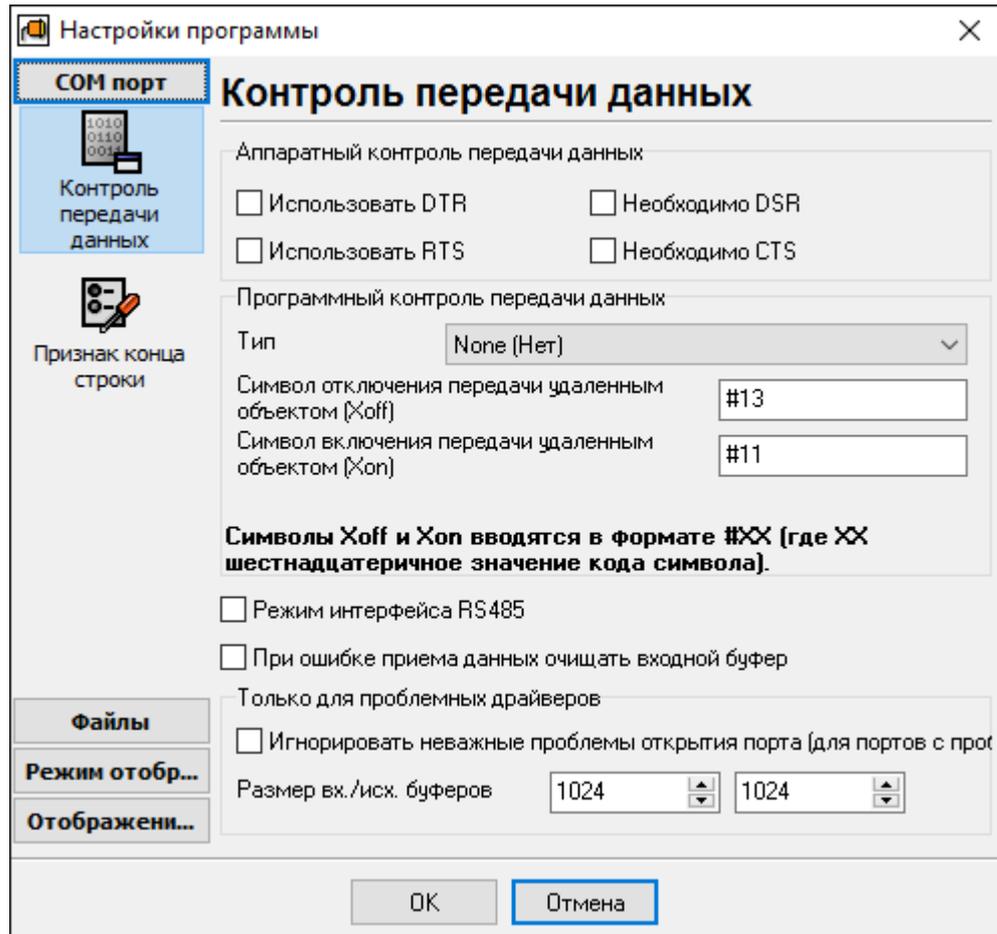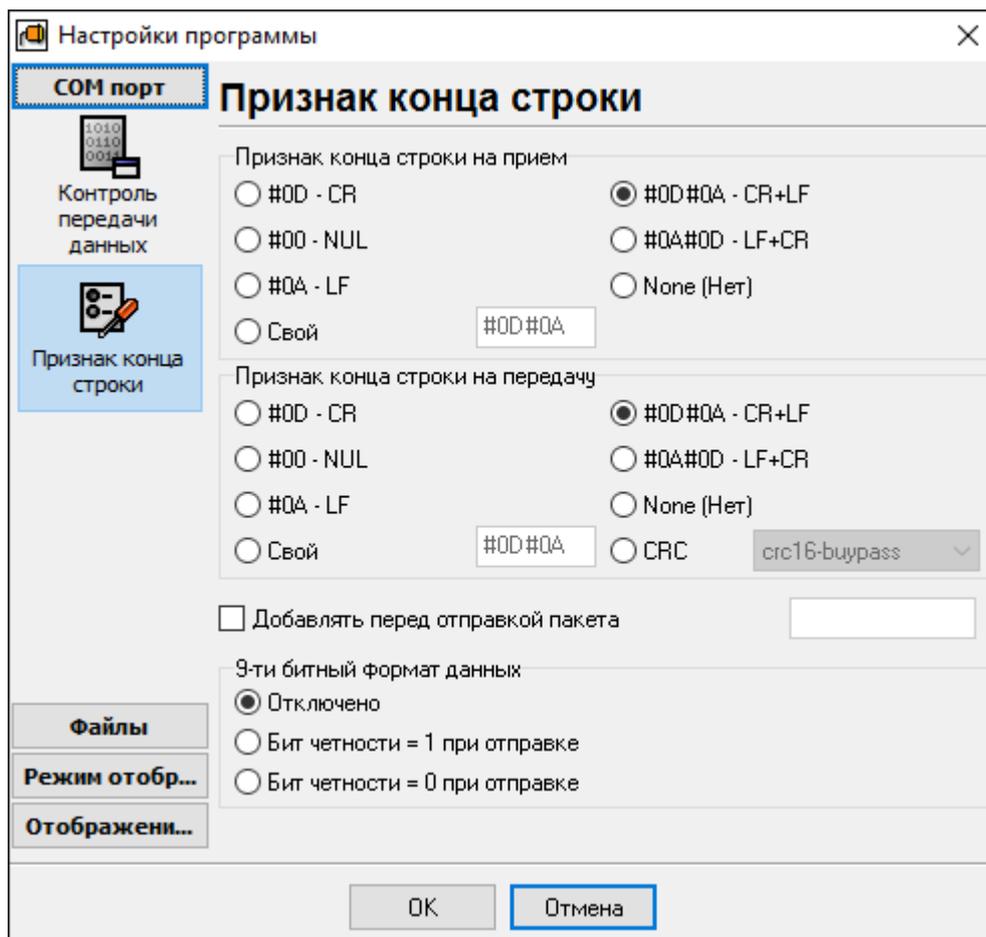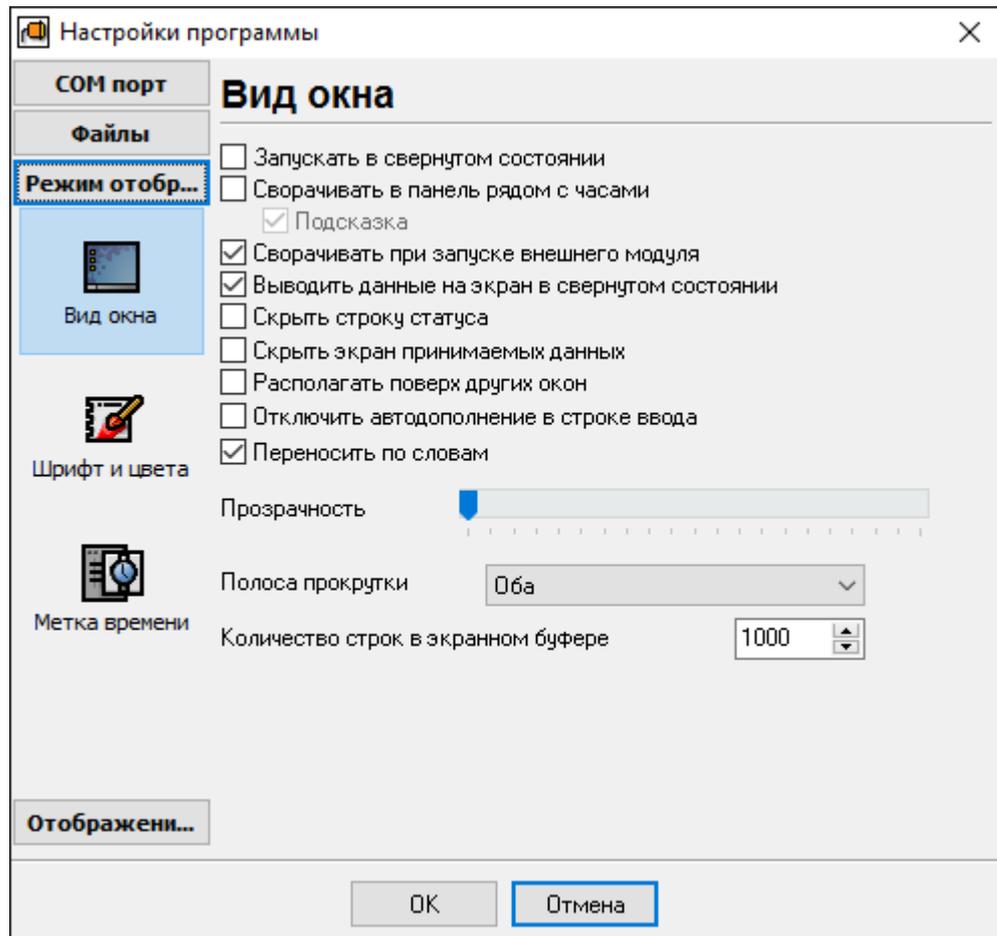
                        (SysTray).   ,

        ,   .

         .   Systray

             Advanced Serial Port Monitor.

3.   -

       .

4.   -

       .

       .

5.   ,   -

      ,   .

       .

6.   -

7.   -

8.   -   ,
.

" | "
:
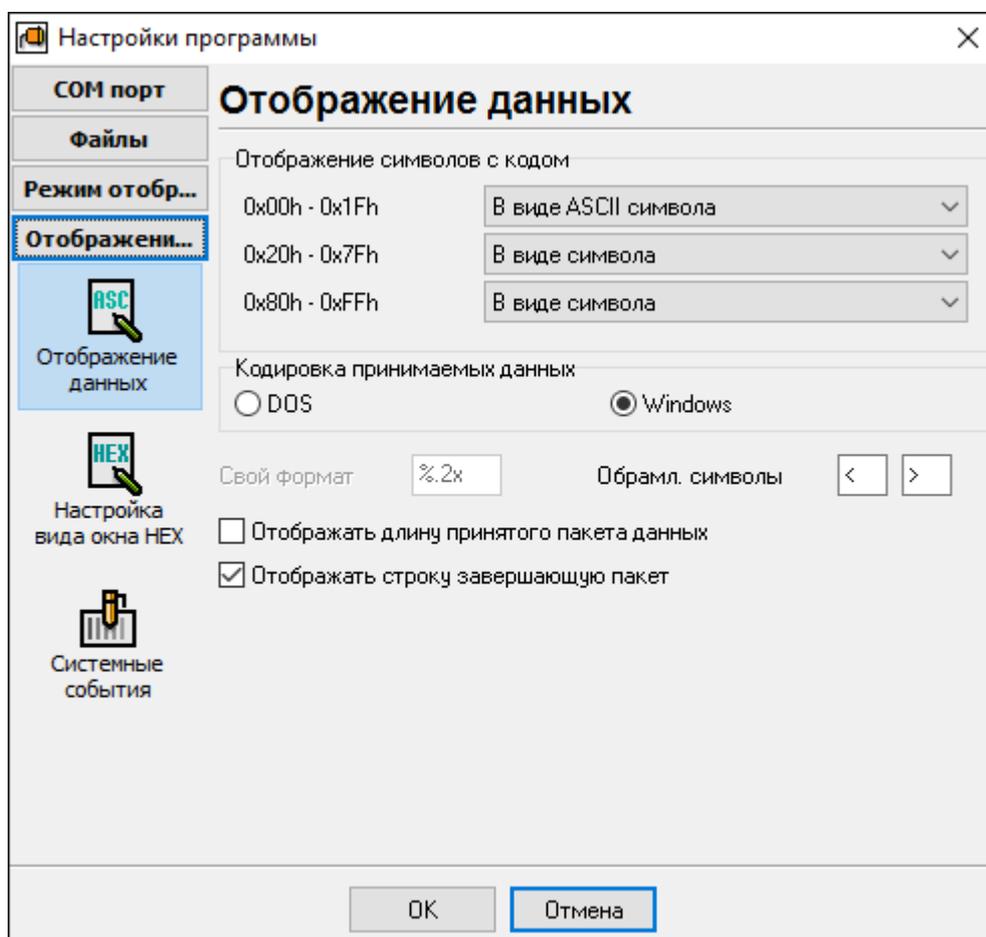
.2.

1.   < 20 Hex -
< 20 Hex.
,
.

2.   > 20 Hex -
> 20 Hex.

,   .

3.   .   %d
,   %x -
.

.

4.   ASCII   ,

5.

6.

3.5

- ;
- "   "   "Enter"   ;
- ,   ,   COM   ;
- "   "
  :
  - Ctrl+S -   "   ";
  - Ctrl+O -   "   /   ";
  - Ctrl+W -   /   ;
  - Ctrl+C -   "   ";
  - F4 -   ;
  - Ctrl+T -   ,   ;
  - Ctrl+F -   ,   ;
  - Ctrl+Alt+E -   ;
  - Ctrl+Alt+F -   ;
  - Ctrl+Alt+   -   COM   .

- "   ".
  ;
- RTS   DTR

3.6

"   "   "   "
"
.
"
"   "   ".   "
"   ,   .
"   "   .
"   ",   ,
"   ".
(TX -   , RX -   , SYS -
).
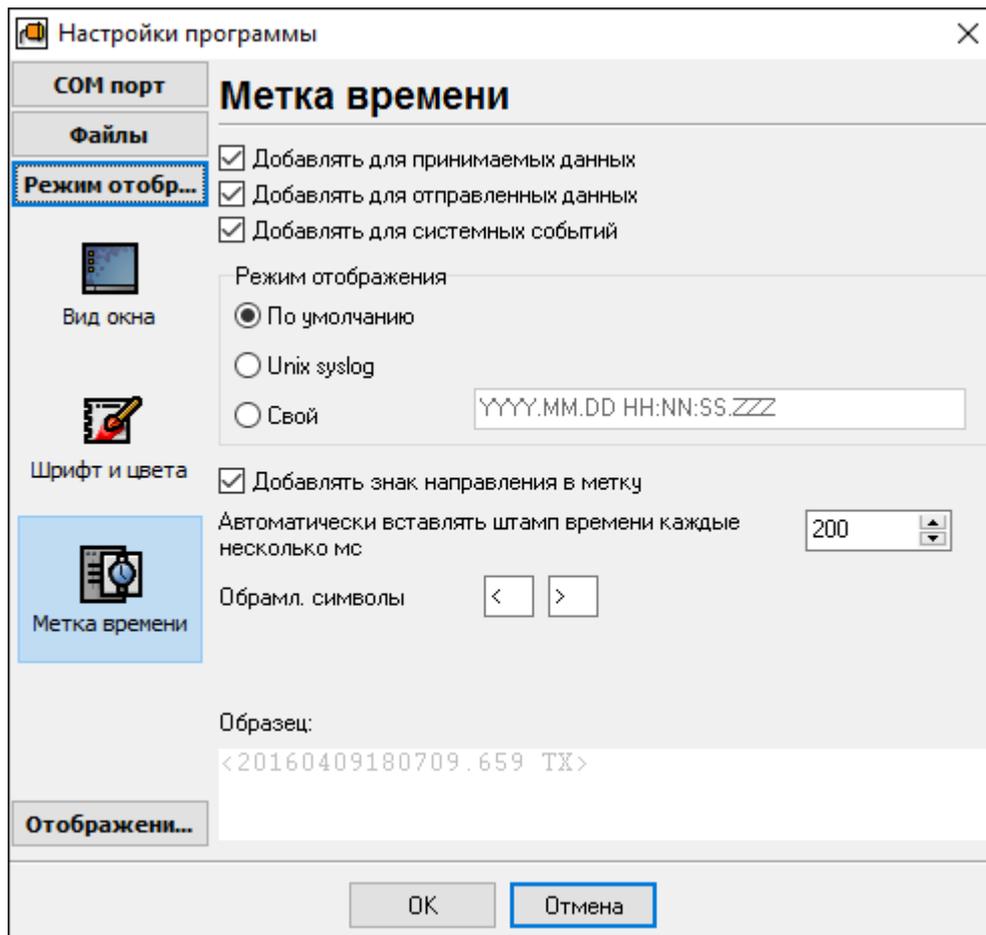
.1.

3.7

ASPM

. ASPM

.

aspmon.exe -f script.txt

script.txt.

.

3.8

,
. "custombaud.txt"
, . ,
ASPM.

:

```
; Please, add your custom baud rates here
; One baudrate per row. Please, use digits only
; After you finished, please, restart ASPM
28800
```

3.9

3.9.1

Advanced Serial Port Monitor .

- _____(_____)_____ [13] -
COM .
- _____ [15] - ,
COM ;

.

, .

3.9.2

.
:

Настройки программы	✕
COM порт	
Файлы	**Исходные данные**
Исходные данные	Имя файла
	"C:\Program Files (x86)\Debugging Tools for Windows\cdb.exe" 📂
Файлы протокола	Режим вывода файла
	○ Однократный ● Многократный
	Тип файла
	○ Текстовый файл ● Двоичный файл
	Размер блока данных при чтении
	○ Фиксированный (байты): 255 ⬍
	● Случайный (байтов мин/макс): 20 ⬍ 40 ⬍
	○ Читать из файла до появления символа конца блока данных
Режим отобр...	
Отображени...	
	OK Отмена

.1.

- ·   -   .
                  ,
        (   )   ,   ;
- ·   -   .
         ,   .
              COM   .
    ,   ,
        .

      ,   ,
.   ,
         .

    ,
      .

        .
      "   "
    .   .

3.9.3

Advanced Serial
Port Monitor   .

,   "   "   .

.

5

.

COM порт	**Файлы протокола**
Файлы	Режим создания нового файла
Исходные данные	⦿ Один файл для всех данных
	◯ Отсылаемые и получаемые данные в разных файлах
	◯ Каждый пакет данных в отдельном файле
Файлы протокола	Разделитель пакетов данных
	☐ Записывать метку времени перед записью данных и событий
	Настроить вид метки времени
	Формат вывода файла Как видно на экране
	Имя файла
	"E:\test data.bin"
Режим отобр...	
Отображени...	
	ОК Отмена

.1

:

- -
,   ;
- -
.

;

- -
(

).   ,   .
(   )
.

300　.

,

,

.

.

,

.

"

"

:

,　　　　　　　　　　　，

,

!

-　　　　　.

.

.

.　　　　　　　　　　Far Manager,

.

3.10

3.10.1

-

"　　".　　　　　.　　　:

- 　　　　　　　　　　　　-

.

;

- 　　　　　-

,

_____ 13 .

| #02REQ:1#03 | ✔ | Послать | ⊗ Закрыть |

.1.

3.10.2

,　　　　　"

"　　".

_____ 16 ".

,　　　　　(ASCII)

,

.　　　　　　　　　　，　　　　　　.

-

.   .

3.10.3

Advanced Serial Port Monitor   ,

,   .

,   COM

.   ,

COM   ,

.   " |

"   ,

(   ).   15   .

,   Advanced Serial Port Monitor

COM   .

.1
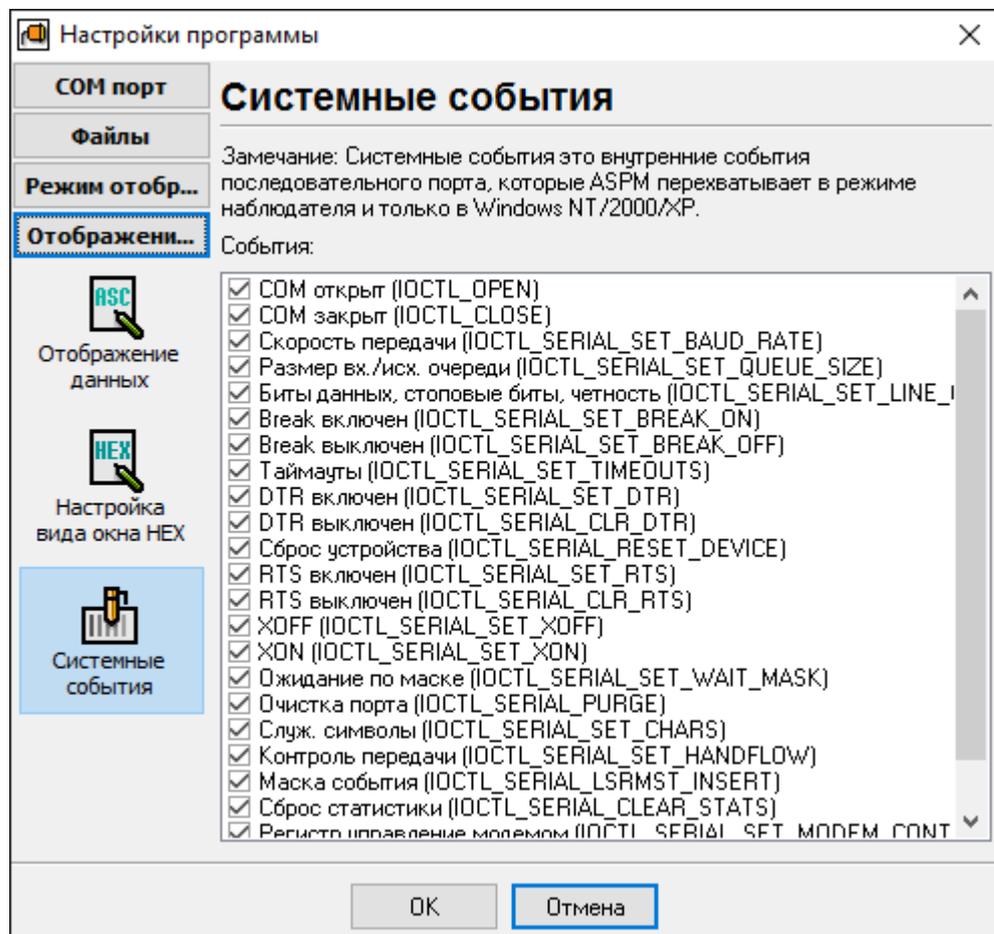
3.10.4

Advanced Serial Port Monitor
.

RS232   RS485.

Advanced Serial Port Monitor

,   .

.

.

3.11

3.11.1

COM
.

,   ,   .
.

#02REQ:1#03   ▼ | Послать | ⊗ Закрыть

.1.

Hex   **#XX** -
0   FF Hex.   (
)   #XX,   # -   -   ,   XX
.   #,
.

✓   Находить Hex коды в виде #XX

Находить ASCII коды с обрамляющими символами

Находить Hex коды в виде XX

.2

1:   128 (Dec)   80(Hex).
: #80.
2:   2 (Dec)   2(Hex).
: #02.
3:   #02MB.   :
##02MB.

- **ASCII**   -
ASCII   : NUL, SOH, STX, ETX, EOT, ENQ, ACK, BEL, BS, HT,
LF, VT, FF, CR, SO, SI, DLE, DC1, DC2, DC3, DC4, NAK, SYN, ETB, CAN, EM, SUB, ESC, FS,

GS, RS, US.   "<",   ASCII, ">" (   ).

5: <STX>Test<ETX>

                       ,   Enter   .
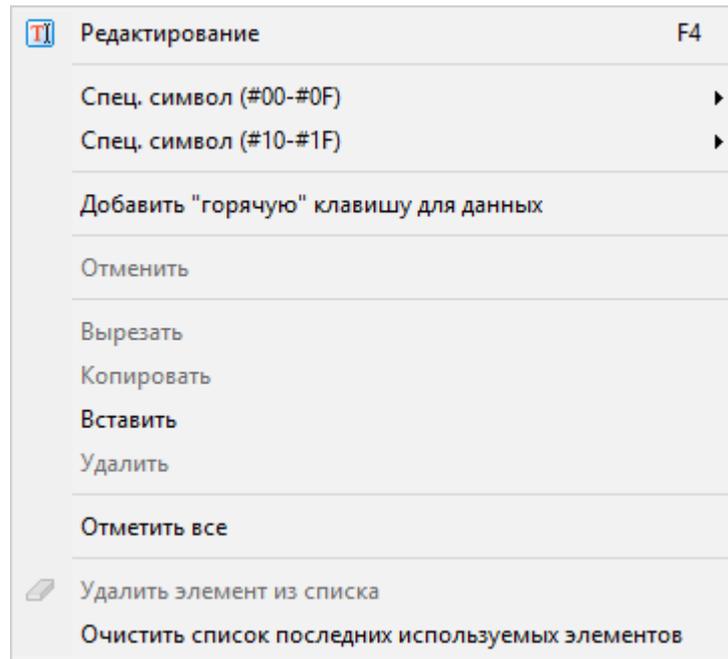
.3

3.11.2

         .   "_____ 13 "

3.12   (   )

3.12.1 Script Language

The basic syntax of the script language of Advanced Serial Port Monitor (ASPM) is:

```
<command> <data1> <data2>          ;<comment>
```

where <command> describes the action to perform, <data1> and <data2> are optional arguments, and <comment> is an optional comment. The format of the arguments vary among commands. The various components of each line must be separated by at least one space or a comma. Additional spaces are permitted but ignored. Commands are not case-sensitive.

The following is a list of supported commands followed by brief descriptions and discussions of the relationships between various commands:

```
:<label>
;<comment>
INITPORT <COM1..COM99>
IF CONNECTED <label>
DONEPORT
SEND 'XXXXXX'
WAIT 'XXXX' <timeout in ms>
IF SUCCESS <label>
IF TIMEOUT <label>
IF FAIL <label>
WAITMULT 'XXX|YYY|ZZZ' <timeout in ms>
IF 1,2,3...127
GOTO <label>
DISPLAY 'XX XX'
SENDBREAK <duration in ms>
DELAY <duration in ms>
SET <option> <data>

RUN <command> <wait>
```

`:<label>`

A point in the script file that can be jumped to via a GOTO or IF instruction. A label name can be any type of string without embedded spaces. For example ':TopOfLoop', ':TOP_OF_LOOP' are both acceptable; ':top of loop' is not.

`;<comment>`

Any line that starts with a semicolon is considered a comment. Blank lines are also considered comments and may be freely added for readability.

`INITPORT <COM1..COM99>`

Opens the specified port. Only one port at a time may be opened.

`DONEPORT`

Closes a port previously opened with INITPORT.

`SEND 'XXXXXX'`

Transmits the string 'XXXXXX'. Control characters may transmitted by preceding a character with '^'. For example, a control C character is represented by ^C. You'll use this feature most often when

sending carriage returns. For example, SEND 'myname^M' might be an appropriate response to a logon prompt where you would normally type your name and press <Enter>. NOTE: The control characters must be *inside* the quote marks, if quote marks are necessary. If the string does not contain any embedded blanks the beginning and ending quotes can be omitted. The quotes are required if the string has embedded blanks. Here are some examples to illustrate this point:

```
SEND ABC        sends ABC
SEND 'ABC'      sends ABC
SEND A B C      sends only the A ('B C' is considered a comment)
SEND 'A B C'  sends A B C
```

WAIT `'XXXXX' <timeout in ms>`

Waits up to <timeout in ms> milliseconds for a particular received string. The string comparison is always case insensitive. However, the string comparison need not be complete. If, for example, a host returns the string 'Host XXXX ready' where XXXX might vary from session to session, the WAIT command should wait for 'ready' only. As with the SEND command, beginning and ending quotes are only required if the string contains embedded blanks. This command sets one of three conditions: SUCCESS, FAIL or TIMEOUT, which can be tested with the IF command. SUCCESS is set if the string is received before the timeout. TIMEOUT is set if the timeout expires before the string is received. FAIL is set if the timeout expires and all retries are exhausted.

IF SUCCESS/TIMEOUT/FAIL/RING/RTS/CTS/DTR/DSR `<label>`

Tests the condition set by the last command and, if the tested condition is true, script execution jumps to <label>. If the condition is not true then execution continues with the next statement.

RING/RTS/CTS/DTR/DSR condition tests corresponding pin state.

WAITMULTI `'XXX|ZZZ|YYY', <timeout in ms>`

Waits up to <timeout in ms> milliseconds for one of several substrings. The bar character (|) separates the substrings. The comparisons are always case insensitive. The maximum length of the entire string is 255 characters. As with the SEND command, beginning and ending quotes are only required if the string contains embedded blanks.
This command sets a numeric condition result based on the substring received: '1' is set if the first substring is received, '2' is set if the second substring is received, and so on. If none of the strings are received then TIMEOUT is set; if all retries have been exhausted then FAIL is set.

Note: YYY is being tested first then ZZZ and XXX. ZZZ is being tested first then XXX. If YYY is a part of ZZZ or XXX, but a serial port received ZZZ or XXX, then YYY condition will be executed.

IF `1,2,3...127 <label>`

Tests the condition set by the last WAITMULTI command and, if the tested condition is true, script execution jumps to <label>. If the condition is not true then execution continues with the next statement.
The following example sends a modem dial command, then waits for one of CONNECT, NO CARRIER, or BUSY responses. If none of the responses are received then control falls through to the GOTO statement:

```
send 'atdt260-9726^m'
waitmulti 'connect|no carrier|busy' 60000
if 1 HandleConnect
if 2 HandleNoConnect
```

```
   if 3 HandleBusy
   goto HandleTimeout
 :HandleConnect
    ...proceed with session
 :HandleNoConnect
    ...handle noconnect error
 :HandleBusy
    ...handle busy error
 ...
```

GOTO `<label>`

Unconditionally jumps to <label>.

DISPLAY `'Just did something'`

This can be used to monitor the progress of the script and to aid in debugging.

EXIT

Exit the program.

SENDBREAK `<duration in ms>`

Transmits a break of <duration in ms> milliseconds.

DELAY `<duration in ms>`

Delays for <duration in ms> milliseconds. The script doesn't yield during delays so keep the delays as short as possible.

PLUGIN START `<file name>`

Execute plugin module with `<file name>` file name. This plugin must exists in the Plugins folder of ASPM.

PLUGIN STOP `<file name>`

Stop `<file name>` plugin's execution. This plugin must exists in the Plugins folder of ASPM.

PLUGIN SHOW `<file name>`

Sets the plugin's main window to normal state.

PLUGIN HIDE `<file name>`

Sets the plugin's main window to minimized state.

MODE `<mode>`

Value:
- **manual** - manual mode;
- **spy** - spy mode.

SOURCE `<data source>`

Value:
- **string** - use the input string as a data source;
- **file** - use file as a data source.

CLEAR - clear data in the screen.

SET BAUD <number>
Sets the Baud property of ASPM.

SET DATABITS <5,6,7,8>
Sets the DataBits property of the ASPM. Allowable values are 5, 6, 7 or 8.

SET FLOW <RTS/CTS,XON/XOFF,NONE>
Sets flow control options for the ASPM. Allowable values are RTS/CTS for hardware flow control, XON/XOFF for software flow control, and NONE to turn off all flow control.

SET PARITY <NONE,ODD,EVEN,MARK,SPACE>
Sets the Parity property of the ASPM. Allowable values are NONE, ODD, EVEN, MARK or SPACE.

SET STOPBITS <1,2>
Sets the StopBits property of the ASPM. Allowable values are 1 and 2.

SET RETRY <data>
Sets an internal retry count that is incremented whenever WAIT or WAITMULTI result in a TIMEOUT condition. When <retry count> TIMEOUTs have occurred the FAIL condition is set. The default is 1, meaning no retries are attempted.

SET LEFT <data>
Sets the position of main window on screen.

SET TOP <data>
Sets the position of main window on screen.

SET WIDTH <data>
Sets the main window width in pixels.

SET HEIGHT <data>
Sets the main window height in pixels.

SET WINDOWSTATE <data>
Sets the main window state. 0 - maximized, 1 - minimized, 2 - normal state.

RUN <command> <wait>
Executes the specified command, batch file or program. <wait> can be *true* or *false*; and determines whether the script waits for the command to complete its execution. <wait> is *true* by default.

Note: the command should contain full path to batch file or program. If the path contains spaces then the command should be enclosed by "".

Command example: RUN "c:\Program Files\Advanced Serial Port Monitor\command.bat" false

Here's an example logon script showing how these commands might be used to log into a host or terminal server:

```
SET RETRY 10                 ;Try 10 times
:Again
SEND ^C                      ;Send an attention character
WAIT 'READY' 182             ;Wait 10 seconds for response
IF SUCCESS Logon             ;Got prompt, continue with logon
IF TIMEOUT Again             ;Try again if we timed out
IF FAIL, Done                ;Give up after 10 tries
:Logon
SEND 'Name, password^M'      ;Send name and password

...

:Done
SEND 'Bye^M'
```

3.12.2 Additional set commands

Main window options

SET MAIN.OUTINPUTDATATOSCR <value> - Output data sent on &screen
Value: True, False

SET MAIN.AUTOSCROLL <value> - Auto-scroll monitor window
Value: True, False

SET MAIN.COM1VISIBLE <value> - Visibility of the COM port #1 toolbar
Value: True, False

SET MAIN.COM2VISIBLE <value> - Visibility of the COM port #2 toolbar
Value: True, False

SET MAIN.COMBOVISIBLE <value> - Visibility of the COM port management toolbar
Value: True, False

SET MAIN.STATUSVISIBLE <value> - Visibility of the line status toolbar
Value: True, False

SET MAIN.BUTTONSVISIBLE <value> - Visibility of the quick access toolbar

Value: True, False

SET MAIN.STATESVISIBLE <value> - Visibility of the additional toolbar
Value: True, False

SET MAIN.SHORTCUTSVISIBLE <value> - Visibility of the shortcuts toolbar
Value: True, False

SET MAIN.PARSEXX <value> - Parse #XX Hex codes
Value: True, False

SET MAIN.PARSEASCII <value> - Parse ASCII codes with framing characters
Value: True, False

SET MAIN.DIFFERENTSETT <value> - Use different settings for each program instance
Value: True, False

SET MAIN.PORT <value> - Avalaible COM port number
Value:
- **0** - COM1
- **1** - COM2
- etc..

SET MAIN.SPEED <value> - COM port baud rate
Value:
- **0** - 110
- **1** - 300
- **2** - 600
- **3** - 1200
- **4** - 2400
- **5** - 4800
- **6** - 9600
- **7** - 14400
- **8** - 19200
- **9** - 38400
- **10** - 56000
- **11** - 57600
- **12** - 115200
- **13** - 230400
- **14** - 460800
- **15** - 921600
- **16** - 15600 (custom baud)
- **17** - 28800 (custom baud)

`SET MAIN.BITS` <value> - Number of data bits
Value:
- **0** - 5
- **1** - 6
- **2** - 7
- **3** - 8

`SET MAIN.PARITY` <value> - Parity
Value:
- **0** - None
- **1** - Odd
- **2** - Even
- **3** - Mark
- **4** - Space

`SET MAIN.STOPBITS` <value> - Stop bits (1.5 will be exposed automatically for 5 data bits and 2 stop bits)
Value:
- **0** - 1
- **1** - 2

`SET MAIN.DELAY` <value> - Data output delay (in miliseconds)
Value: Data output delay (in miliseconds)

`SET MAIN.AUTO` <value> - &Auto delay
Value: True, False

`SET MAIN.COMMAND` <value> - Input string. Enter data and press 'Enter' (F4)
Value: Input string. Enter data and press 'Enter' (F4)

`SET MAIN.OUTRECEIVEDTOFILE` <value> - Output data received to a file
Value: True, False

`SET MAIN.OUTINPUTDATATOFILE` <value> - Output data sent to a file
Value: True, False

`SET MAIN.OUTSYSDATATOFILE` <value> - Output system events (in the spy mode) to a file

Value: True, False

Data flow control

Hardware flow control

`SET USEDTR` <value> - Use DTR
Value: True, False

`SET USERTS` <value> - Use RTS
Value: True, False

`SET REQUIREDSR` <value> - Require DSR
Value: True, False

`SET REQUIRECTS` <value> - Require CTS
Value: True, False

Software flow control

`SET FLOWCONTROL` <value> - Software flow control
Value:
- **0** - None
- **1** - On receiving
- **2** - On transmitting
- **3** - Both

`SET XOFFCHAR` <value> - Character that is sent to disable remote sending (Xoff). You should type string, where each #XX block will be replaced with character with XX hex code.
Value: One or more characters

`SET XONCHAR` <value> - Character that is sent to enable remote sending (Xon). You should type string, where each #XX block will be replaced with character with XX hex code.
Value: One or more characters

SET RS485MODE <value> - RS485 interface mode
Value: True, False

SET FLUSHBUFFER <value> - Flush incoming buffer on line error
Value: True, False

For problem drivers only

SET IGNOREERR <value> - Ignore non-important port opening errors (for problematic ports only)
Value: True, False

Signs of end of string

SET INEND <value> - Termination string or character while receiving
Value:
- **0** - #0D - CR
- **1** - #00 - NUL
- **2** - #0A - LF
- **3** - Custom
- **4** - #0D#0A - CR+LF
- **5** - #0A#0D - LF+CR
- **6** - None

SET OUTEND <value> - Termination string or character while transmitting
Value:
- **0** - #0D - CR
- **1** - #00 - NUL
- **2** - #0A - LF
- **3** - Custom
- **4** - #0D#0A - CR+LF
- **5** - #0A#0D - LF+CR
- **6** - None

SET CUSTOMSEND <value> - Custom termination string
Value: One or more characters

SET CUSTOMREC <value> - Custom termination string
Value: One or more characters

`SET GB9BIT` <value> - 9-bit data transfer format
Value:
- **0** - Off
- **1** - Mark parity on send
- **2** - Space parity on send

`SET ADDBEFORE` <value> - Add before a data packet while transmitting
Value: True, False

`SET ADDBEFORE` <value> - Send data before each data block. You should type string, where each #XX block will be replaced with character with XX hex code.
Value: One or more characters

File with source data

`SET SOURCEFILE` <value> - The name of a file with data, which will be sent over serial port
Value: 'drive:directory\file name'. If your path and/or file name contains spaces, then you should add double quotes before and after value: '"drive:directory\file name"'

`SET FILEMODE` <value> - File output mode
Value:
- **0** - Unitary
- **1** - Repeated

`SET FILETYPE` <value> - File type
Value:
- **0** - ASCII or plain text
- **1** - Binary

ASCII mode

`SET FILEINTERPRET` <value> - Interpret characters in the file like #XX or (depend on the corresponding option) as a character with same Hex code
Value: True, False

Binary mode

Reading data block size

SET BLOCKFIXEDSIZE <value> - Fixed size (bytes):
Value: True, False

SET BLOCKRANDOMSIZE <value> - Random size (bytes min/max):
Value: True, False

SET BLOCKFIXEDSIZE <value> - Reading data blocks from the source file with this size
Value: 1..1000000

SET BLOCKRANDOMSIZEMIN <value> - Reading data blocks from the source file with this min size
Value: 1..1000000

SET BLOCKRANDOMSIZEMAX <value> - Reading data blocks from the source file with this max size
Value: 1..1000000

SET BLOCKUNTIL <value> - Reading from a file until occurrence of a termination character
Value: True, False

Protocol files

SET CREATEFILEMODE <value> - Create new file mode
Value:
- **0** - One file for all data
- **1** - Data sent and received in different files
- **2** - Each sent and received packet in a different file

One file for all data

SET OUTPUTFILE <value> - File name
Value: 'drive:directory\file name'. If your path and/or file name contains spaces, then you should

add double quotes before and after value: '"drive:directory\file name"'

Data sent and received in different files

SET FILEFORSENT <value> - File name for data sent
Value: 'drive:directory\file name'. If your path and/or file name contains spaces, then you should add double quotes before and after value: '"drive:directory\file name"'

SET FILEFORRECEIVED <value> - File name for data received
Value: 'drive:directory\file name'. If your path and/or file name contains spaces, then you should add double quotes before and after value: '"drive:directory\file name"'

Each sent and received packet in a different file

SET DIRECTORYEDIT <value> - Folder for files with data received and sent
Value: '"drive:directory\"'. If your path contains spaces, then you should add double quotes before and after value: '"drive:directory"'

SET FILENAMEPREFIXSENT <value> - File name prefix for data sent
Value: One or more characters

SET FILENAMEPREFIXRECEIVED <value> - File name prefix for data received
Value: One or more characters

SET OVERWRITEFILES <value> - Overwrite existing files
Value: True, False

SET ADDNAMESTAMP <value> - Add date/time stamp to a file name
Value: True, False

SET FILENAMEEXT <value> - File name extension
Value: One or more characters

SET COUNTERFMT <value> - File name counter format
Value: One or more characters

SET MAXCOUNTER <value> - Maximum counter value (-1 - no limitations)
Value: Any decimal number

SET FILENAMESUFFIXSENT <value> - File name suffix for data sent
Value: One or more characters

SET FILENAMESUFFIXRECEIVED <value> - File name suffix for data received
Value: One or more characters

SET COMMONCNT <value> - Common counter for received and sent files
Value: True, False

SET SEPARATOR <value> - Separator between data packets. You can type any characters here, where each #XX block will be replaced with character with XX hex code.
Value: One or more characters

SET ADDBODYSTAMP <value> - Write date/time stamp to a file before writing data and events
Value: True, False

SET FILEFORMAT <value> - Format of the file output
Value:
- **0** - Without any changes
- **1** - As you see it on screen

Window view

SET STARTMINIMIZED <value> - Start up in minimized state
Value: True, False

SET HIDETOSYSTRAY <value> - Minimize to Systray (a panel near clock)
Value: True, False

SET HINTBALOON <value> - Hint balloon
Value: True, False

SET HIDEMAINFORM <value> - Hide on plugin start up
Value: True, False

SET SHOWINHIDE <value> - Display data sent and received in minimized state
Value: True, False

SET HIDESTATUS <value> - Hide status bar
Value: True, False

SET HIDESCR <value> - Hide window with data
Value: True, False

SET STAYONTOP <value> - Stay on top (over other windows)
Value: True, False

SET AUTOCOMPLETE <value> - Disable autocompletion in the input string
Value: True, False

SET WRAPWORDS <value> - Wrap words
Value: True, False

SET LINESCOUNT <value> - Number of strings in a screen buffer
Value: 10..30000

Font and colors

SET HIGHLIGHT <value> - Highlight data sent on screen
Value: True, False

SET COLORHIGHLIGHT <value> - Data highlighting color
Value: Data highlighting color

SET FONTLIST <value> - Courier
Value: Courier

SET FONTCOLOR <value> - Main window a font color
Value: Main window a font color

SET BACKCOLOR <value> - Main window a background color
Value: Main window a background color

SET COMBOCOLORPROFILE <value> - Default
Value:
- **0** - Dark
- **1** - Default

SET FONTBOLD <value> - Bold
Value: True, False

SET FONTITALIC <value> - Italic
Value: True, False

HEX view

SET COLGRIDLINES <value> - Grid lines color
Value: Grid lines color

SET COLEVENCOL <value> - Grid lines color
Value: Grid lines color

SET COLOFFSET <value> - Grid lines color
Value: Grid lines color

Data view

SET CHARSET <value> - Charset of data received
Value:
- **0** - DOS
- **1** - Windows

View mode of characters with code

SET SYM20HEX <value> - 0x00h - 0x1Fh
Value:
- **0** - Don't display
- **1** - As Hex #XX code
- **2** - User's format
- **3** - As ASCII character code

SET SYM80HEX <value> - 0x20h - 0x7Fh
Value:
- **0** - Don't display
- **1** - As Hex #XX code
- **2** - As same character
- **3** - User's format

SET SYMFFHEX <value> - 0x80h - 0xFFh
Value:
- **0** - Don't display
- **1** - As Hex #XX code
- **2** - As same character
- **3** - User's format

SET SHOWSTRLEN <value> - View received packet length
Value: True, False

SET SHOWEND <value> - View string that completes a data packet
Value: True, False

SET USERFORMAT <value> - User view mode of characters with code < 20 Hex %d - decimal value of character code %x - hexdecimal value %.2x - hexdecimal value with leading zero
Value: One or more characters

SET FRAMEBEGIN <value> - Beginning characters of a frame for an ASCII code
Value: One or more characters

SET FRAMEEND <value> - Ending characters of a frame for an ASCII code
Value: One or more characters

HEX window view

SET BYTESROW <value> - Bytes per row
Value: 1..128

SET BYTESCOL <value> - Bytes per column
Value: 1..8

SET HEXTRANS <value> - Translation
Value:
- **0** - Windows
- **1** - ASCII
- **2** - BCD
- **3** - DOS
- **4** - Mac

SET SHOWOFFSET <value> - Show offset
Value: True, False

SET HEXLOWER <value> - Hex lowercase
Value: True, False

SET SWAPNIBBLES <value> - Swap nibbles
Value: True, False

SET GRIDLINES <value> - Grid lines
Value: True, False

SET BYTESROWAUTO <value> - Automatically calculate number of bytes per row
Value: True, False

Date/time stamp

SET STAMPVIEW <value> - View mode
Value:

- **0** - Default
- **1** - Unix syslog
- **2** - Custom

SET CUSTOMSTAMPFMT <value> - Custom view mode of a datetime stamp
Value: One or more characters

SET ADDSCRSTAMPSYS <value> - Add to display output for system events
Value: True, False

SET ADDSCRSTAMPSEND <value> - Add to display output for data sent
Value: True, False

SET ADDSCRSTAMPREC <value> - Add to display output for data received
Value: True, False

SET STAMPFRAMEBEGIN <value> - Beginnig characters of a frame for a datetime stamp
Value: One or more characters

SET STAMPFRAMEEND <value> - Ending characters of a frame for a datetime stamp
Value: One or more characters

SET ADDDATADIR <value> - Add data direction sign to a stamp
Value: True, False

SET TIMEOUTINT <value> - Auto-insert a stamp every (ms) in a continuous data flow
Value: 1..65535

System events

SET EVENTS <value> - 0|1|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16|17|18|19|20|21|22|
Value: 0|1|2|3|4|5|6|7|8|9|10|11|12|13|14|15|16|17|18|19|20|21|22|

3.12.3 Control characters

Control characters may transmitted by preceding a character with '^'. For example, a control C character is represented by ^C. You'll use this feature most often when sending carriage returns. For example, SEND 'myname^M' might be an appropriate response to a logon prompt where you would normally type your name and press <Enter>. NOTE: The control characters must be *inside* the quote marks, if quote marks are necessary.

Control character	Dec	Hex
^@	0	0x00
^A	1	0x01
^B	2	0x02
^C	3	0x03
^D	4	0x04
^E	5	0x05
^F	6	0x06
^G	7	0x07
^H	8	0x08
^I	9	0x09
^J	10	0x0A
^K	11	0x0B
^L	12	0x0C
^M	13	0x0D
^N	14	0x0E
^O	15	0x0F
^P	16	0x10
^Q	17	0x11
^R	18	0x12
^S	19	0x13
^T	20	0x14
^U	21	0x15
^V	22	0x16
^W	23	0x17
^X	24	0x18
^Y	25	0x19
^Z	26	0x1A
^[	27	0x1B
^\	28	0x1C
^]	29	0x1D
^_	31	0x1F

4 ?

4.1

, . .

,

.

, SoftIce

- .

, ,

support@aggsoft.ru !

4.2 (FAQ)

: , " " - ?
: , - (-).
DOS- , .

: ?
: , (DOS -
DOS -).
.

: ?
: , ,
.

: , ,
?
: Advanced Serial Port Monitor COM
.