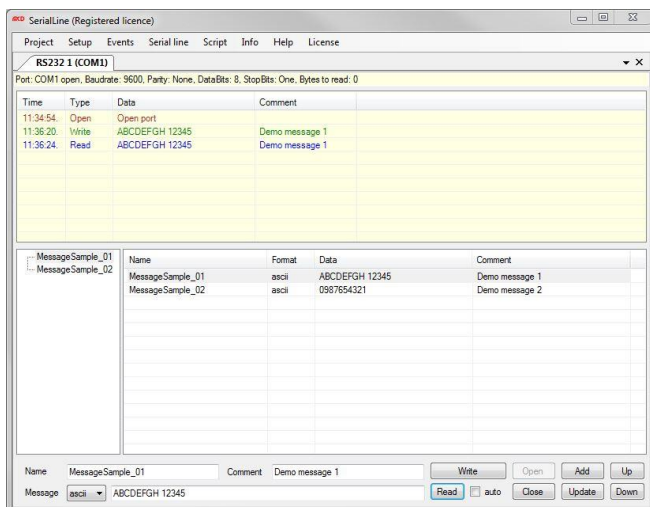


Manual

Toolmonitor Serialline



GET IN **touch**
WITH SENSITIVE TESTING

Softline

Modline

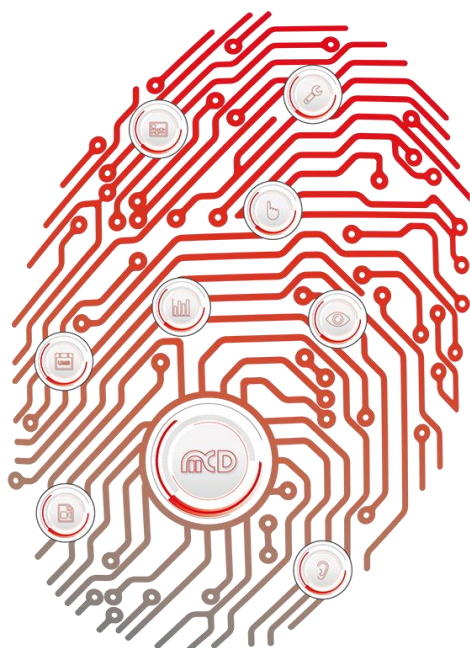
Conline

Boardline

Avidline

Pixline

Application



MCD Elektronik GmbH

Hoheneichstr. 52

75217 Birkenfeld

Tel. +49 (0) 72 31/78 405-0

Fax +49 (0) 72 31/78 405-10

info@mcd-elektronik.de

www.mcd-elektronik.com

HQ: Birkenfeld

Managing CEO: Bruno Hörter

Register Court Mannheim

HRB 505692

Table of Contents

1. GENERAL	3
2. INSTALLATION OF SOFTWARE, DRIVERS	4
2.1. PREREQUISITES	4
2.2. LICENSE	4
2.3. REGISTER COM SERVER	6
3. INTRODUCTION TO THE TOOLMONITOR SERIALLINE	7
3.1. INTERFACES	7
3.2. COMMUNICATION CHANNELS	7
3.3. MANUAL OPERATION	8
3.4. REMOTE CONTROL BY MCD TESTMANAGER CE	8
4. PROJECT MANAGEMENT	10
4.1. COMMAND EXPLANATION	10
5. EVENTS	12
5.1. LOGGING	12
5.2. TRACE	12
6. SETUP	13
6.1. SELECTION OF THE COMMUNICATION CHANNELS	13
6.2. CONFIGURATION OF A COMMUNICATION CHANNEL	13
6.3. GENERAL SETTINGS	14
6.4. LOGGER	15
7. PROGRAMMING	17
7.1. LINE	17
7.1.1. <i>Class Description</i>	17
7.1.2. <i>Virtual Interface Commands</i>	17

1. General

The Toolmonitor Serialline provides a universal, simple communications interface for a number of serial - controlled devices.

Freely definable telegrams or those stored in a library may be sent and received in a simple manner.

A simulator makes testing communications possible even without connected hardware.

All messages, depending on the priority setting, can be logged and displayed.

The program interface can largely be designed with an open hand and customized to the users' requirements. Once created, configuration can be saved in project files and loaded as needed. All telegrams can be sent and received automatically with the help of an integrated script engine. Asynchronous processes can be stored in the Toolmonitor.

Using third - party software, the Toolmonitor can be entirely remote controlled. COM & DCOM or a .Net assembly may be used as the interface for this. Thereby, the Toolmonitor can be integrated into a number of applications, including: Microsoft Visual Studio® (C#, C++ and Visual Basic), Microsoft Office® (such as Excel®), Open Office®, LabView® and MCD TestManager CE.

Order number: # 118951

2. Installation of Software, Drivers

2.1. Prerequisites

- Windows (Windows XP to Windows 8.1, 32 - or 64 - bit)
- .Net Framework version 3.0

Copying SerialLineMonitor.exe to any directory on the target system will suffice to install Toolmonitor Serialline. Alternatively, the installer provided (SerialLineInstall.msi) may be executed.

2.2. License

To protect the software against unauthorized use, the Toolmonitor must be licensed after installation.

For demonstration and test purposes, the Toolmonitor can be run for 30 minutes even without a license. Several program features will be disabled in that case.

To work around the wait period for activation (such as on weekends), a **license for a 24-hour period** may be activated.

To activate the Toolmonitor, open "License Registration" dialog using the License → Register menu options.



Figure 1: Open Registration Dialog

1. The current status of your license will be displayed on the „current license dialog“.

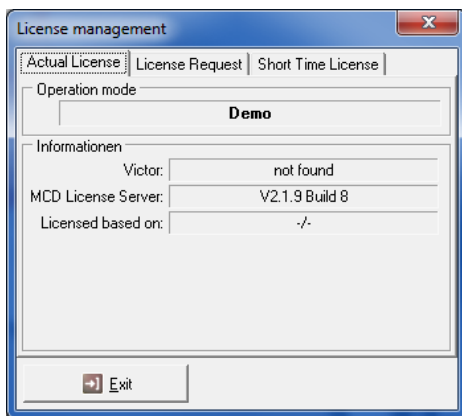


Figure 2: Display License Status

2. To request a permanent license for your software, proceed as follows:

- Select the „Request License“.
- Enter the number of required licenses (for your PC) in the „Number of Licenses“ entry.
- Confirm this entry using the „Created Request File“ button.
- The next dialog will open, which will asking you to save the *MCD Licenser Request* (*.mlr) file.
- Save this file and then send it via email to the following address: **info@mcd-elektronik.de**

- Please enter a contract or project number to assist us with the licensing process.
- Afterwards, you will receive an email from MCD Elektronik with your License file (MCD License Key *.mlk) as an attachment.
- Finally, save this file either in the C:\Windows directory or in the directory, where the .exe - file is stored.
- **After** your software has been **restarted**, its **full scope of features** will be enabled.

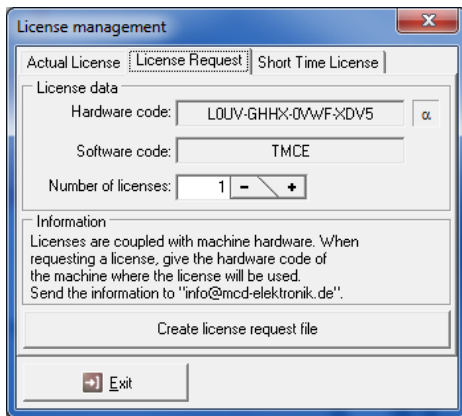


Figure 3: Request for a Permanent License

3. To activate a license for a 24 - hour period, select the „*short - term license*“ tab. Enter the sequence of numbers from the left pane into the entry on the right.

If you cannot read the numbers, click the „*new number*“ button, to get a new number. Once you entered the number correctly, you will be able to activate this short - term license by clicking the „*Activate*“ button. Please note that the short - term license will expire as soon as you shutdown the software. You can re - activate the short - term license as often as desired.

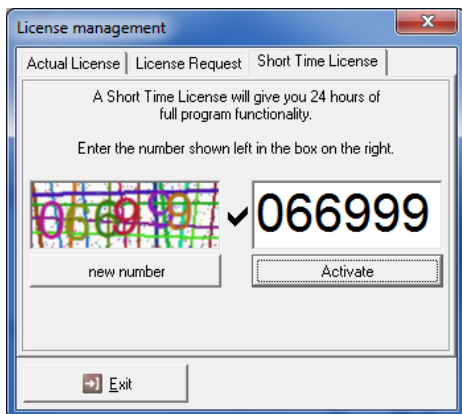


Figure 4: Request Short Term License

2.3. Register COM Server

The Toolmonitor can be registered as a COM server using this menu option. This will be necessary if the Toolmonitor should be controlled remotely by other programs such as MCD TestManager.

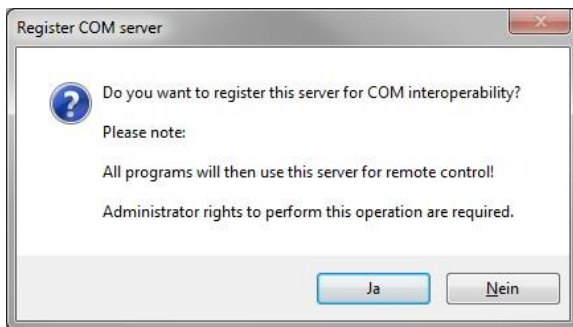


Figure 5: Register COM Server

3. Introduction to the Toolmonitor Serialline

3.1. Interfaces

Supported Interfaces:

- RS232
- USB
- IEC
- TCP/IP
- HTTP
- LIN
- CAN
- Visa
- MOST 25
- MOST 150
- AgilentIO Suite

3.2. Communication Channels

All configured communications channels can be accessed from the menu option “Serialline” on the main menu.

Open communications can be monitored, or the user's messages can be sent, with the help of the respective communications channel. Various types of messages can be sent and received through the manual or automated operation of the Toolmonitor.

Aside from this, several modules can be used at the same time.

The received data can be logged. Furthermore, messages, or sequences of messages, created by the user can be saved and loaded.

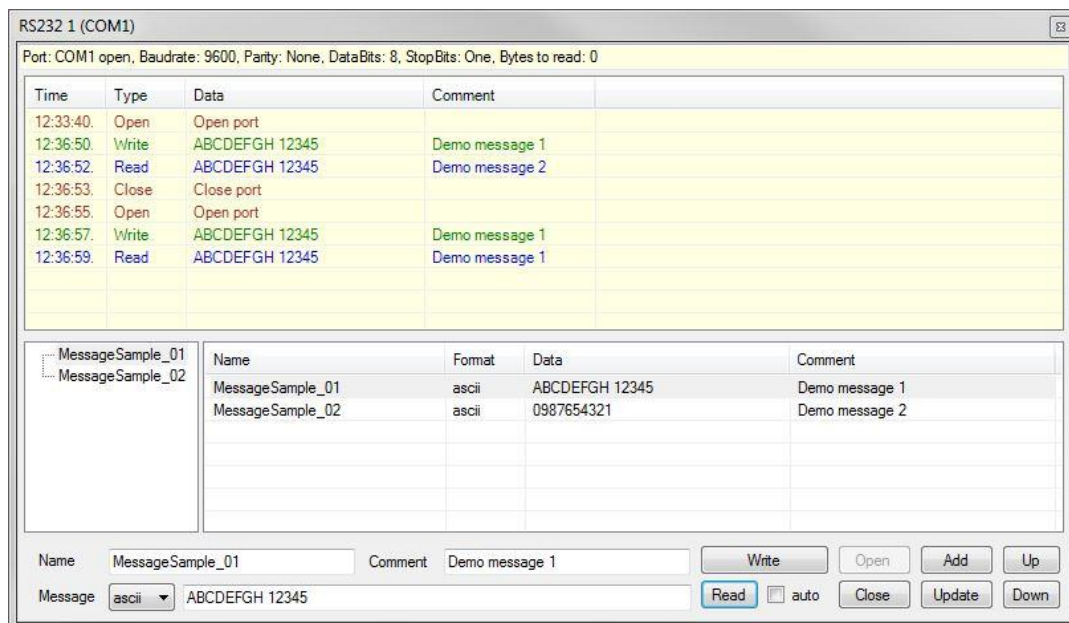


Figure 6: RS232 Communication Channel

3.3. Manual Operation

Communications can be started by clicking the "Open" button. Therefore settings will automatically be made to the channel based on the options.

User messages can be created using an entry dialog. The message type and data content can be entered to this purpose.



Figure 7: Input Mask for Messages by Manual Operation

The message will be sent by clicking the "Write" button.

Messages, if any exist, can be received with the help of the "Read" button.

If the option "auto" has been enabled, incoming messages will automatically be received and displayed.

Additional information about the messages can be stored in the entries "Name" and "Comment", which can be stored by clicking the "Add" button.

Communications can be terminated by clicking the "Close" button.

3.4. Remote Control by MCD TestManager CE

A MCD Toolmonitor can be controlled remotely by the MCD TestManager CE software.

Therefore, open the Toolmonitor and select *Setup* → *Register COM server*. Please confirm the following two dialogs with "YES" and "OK". The Toolmonitor is now registered and can be used in the TestManager CE.

Next, please open the configuration menu of the MCD TestManager under *Setup* or *Options*. Figure 8 shows the required settings for the implementation of the Toolmonitor.

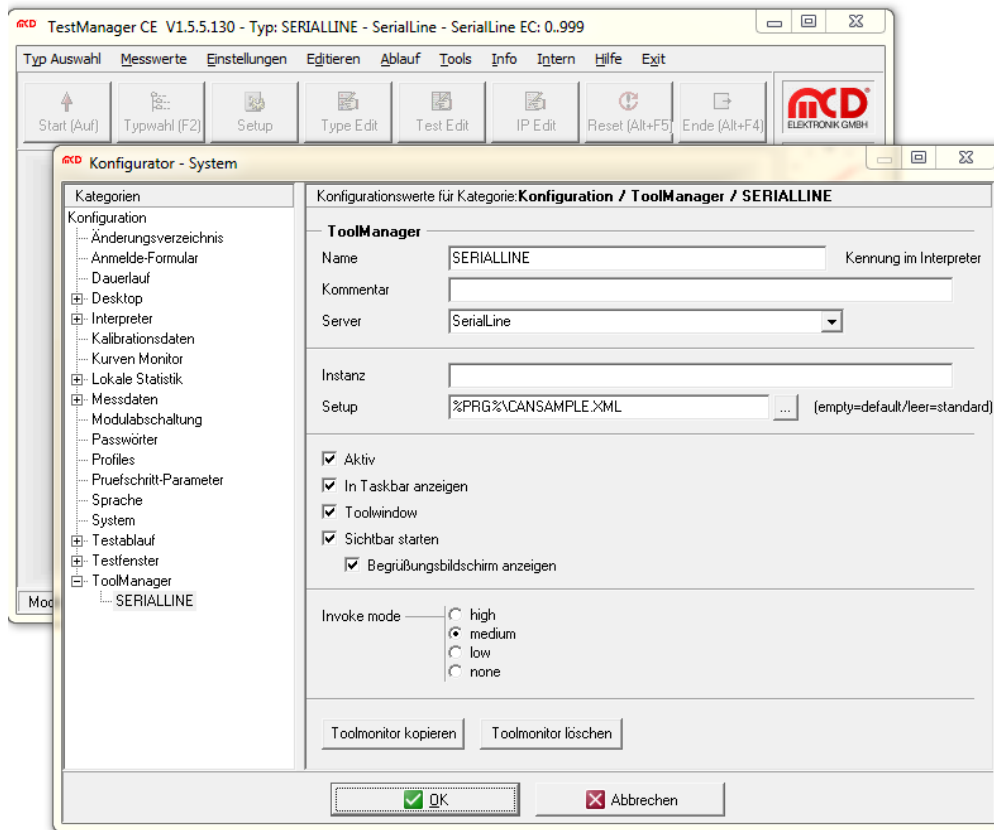


Figure 8: Toolmanager Configuration in the MCD TestManager CE

By the next start of the TestManager, the now registered Toolmonitor will be opened automatically.

Using the commands of the *Virtual Interface* (Available in the Toolmonitor under *Script* → *Virtual Interface*) the Toolmonitor can be remotely controlled from a TestManager IP-step.

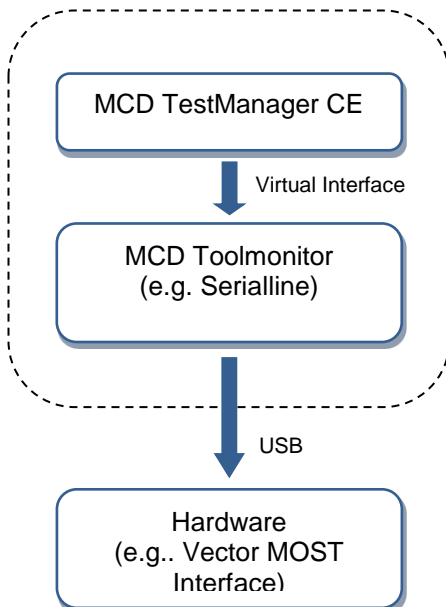


Figure 9: Schematic Diagram of the Remote Control via MCD TestManager

4. Project Management

The current settings and the Toolmonitor layout can be saved and loaded from the menu options under the menu „*Project*“. (All windows can be positioned freely and adjusted according to the user's own needs).

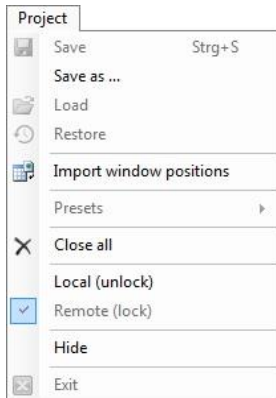


Figure 10: Project Menu

4.1. Command Explanation

Save / Save as...

All current settings can be saved in a project file by clicking the “*Save*” item. Even the current window positions will be stored in the file.

Load

Previously saved settings can be re - loaded by clicking the “*Load*” item. Even the original window positions will be recreated.

Import Window Positions

The “*Import Window Positions*” item imports window positions from a saved setup file. All original settings will not be affected by this.

Presets

Predefined settings can be set using the menu item “*Presets*”.

Close All

The menu item “*Close all*” will close all open forms. The Toolmonitor stays active.

Local

If the Toolmonitor is remote controlled or the setup has been protected with a password, most user actions are protected against direct entry. Activating the “*Local*” mode removes this protection and all operating controls will be accessible again. If the setup process included the entry of a password, it must be entered to release Toolmonitor.

Remote

Activating the menu item "*Remote*" will return the Toolmonitor into the protected mode.

Hide

Activating „*Hide*“ the Toolmonitor will hide, but keep running. If it is not controlled remotely, the Toolmonitor can be re - activated using an icon on the taskbar.

Exit

Activating "*Exit*" the Toolmonitor will terminate. (Not available, if the Toolmonitor is controlled remotely).

5. Events

The logging and trace message displays can be accessed from the menu “Events”.



Figure 11: Event Menu

5.1. Logging

The logging messages for general events, warnings, errors etc. will be displayed here.

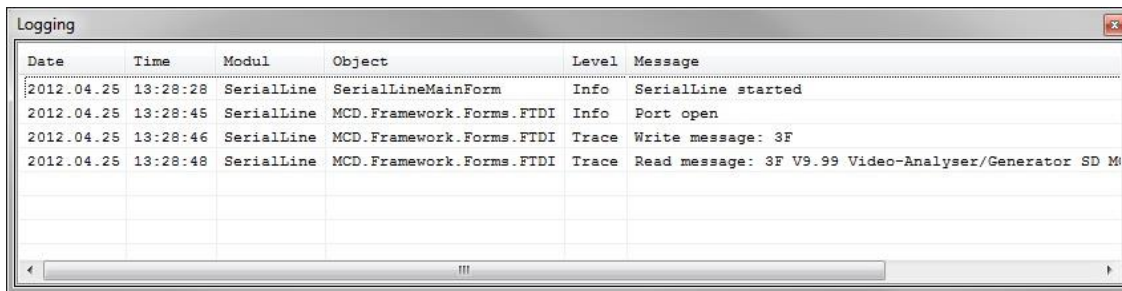


Figure 12: Log Monitor

5.2. Trace

The trace messages (sent or received messages) will be displayed using this menu option.

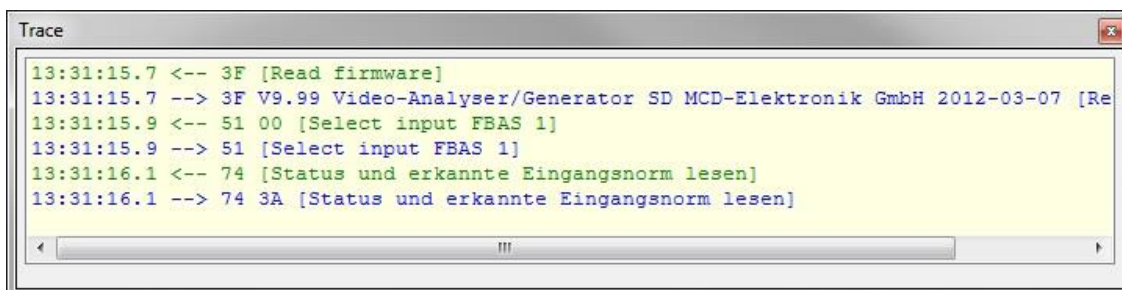


Figure 13: Display of Trace Messages

6. Setup

The Setup menu provides access to the project - specific options and registration of the Toolmonitor as a COM server.

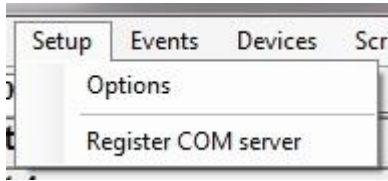


Figure 14: Setup Menu

6.1. Selection of the Communication Channels

From the Options dialog, users can determine which types of communications will be used and how many communications will be needed for each type of communications available to the Toolmonitor Serialline. In addition, a user - specific name can be assigned to each respective type of communication.

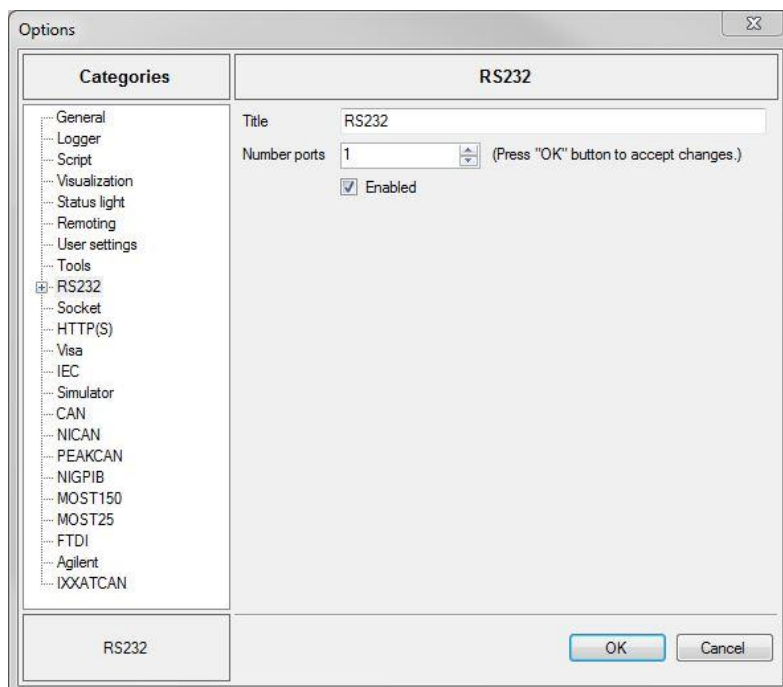


Figure 15: Options of Toolmonitor Serialline

6.2. Configuration of a Communication Channel

Settings that are general or communications - specific can be made for each communication channel at the respective setup.

The general settings includes:

- The name of the communications channel and a comment
- A flag indicating if the channel is active, meaning currently in use
- A flag indicating if the channel should be opened automatically
- A flag indicating if the corresponding channel may automatically check if incoming data is available and if it should be received, as needed

All other settings will depend on the respective type of communication.

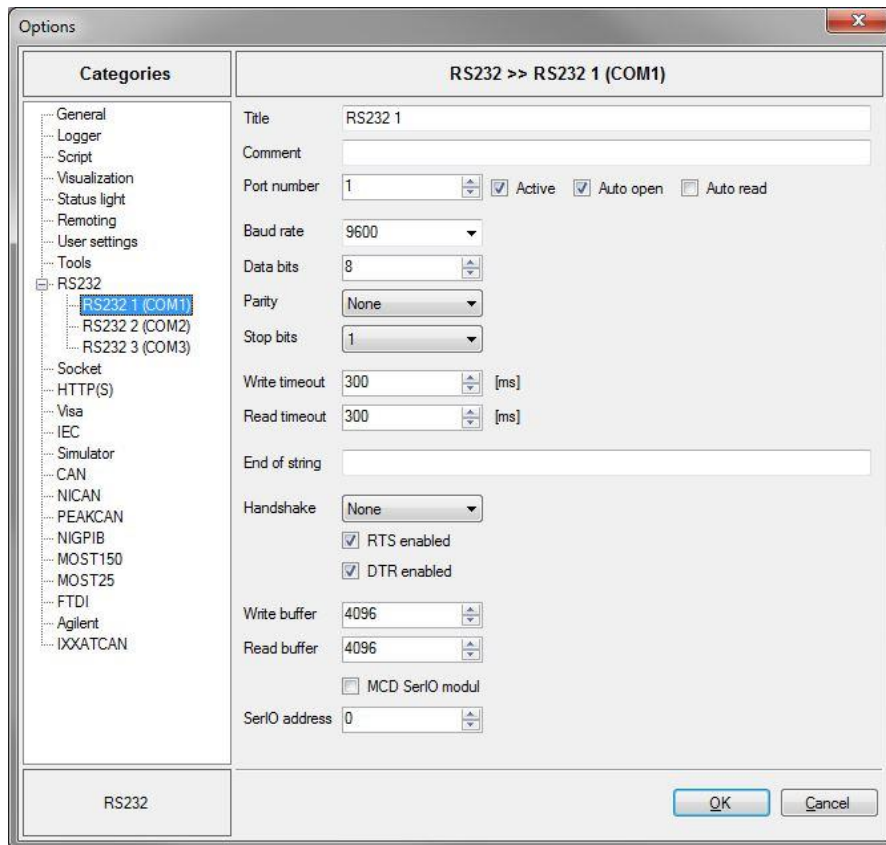


Figure 16: Options for RS232 Communication via Toolmonitor Serialline

6.3. General Settings

General settings can be implemented here.

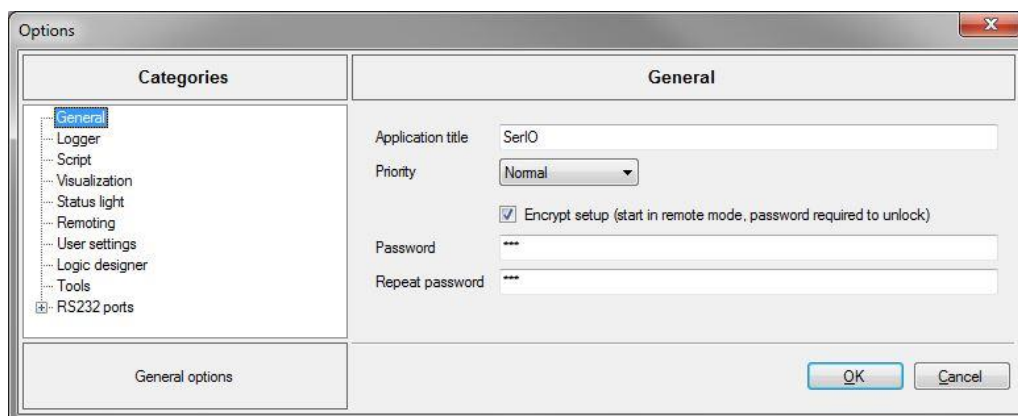


Figure 17: Options

Application Title

The title of the Toolmonitor window can be conformed here.

Priority

The base priority of the Toolmonitor can be set using this entry. This value should only be changed when there is a need for it.

Encrypt Setup

Whether the setup should be stored in an encrypted format or not can be determined using this setting. In that case, the Toolmonitor will be started in the remote mode. In addition, a password must be set, which can be used to access the Toolmonitor. As a control measure, this password must be confirmed. If the password is lost, it will no longer be possible to change the associated setup.

NOTE:

If the Toolmonitor is password - protected, the local and remote modes can no longer be controlled remotely.

6.4. Logger

Settings for the logging of the Toolmonitor can be made here.

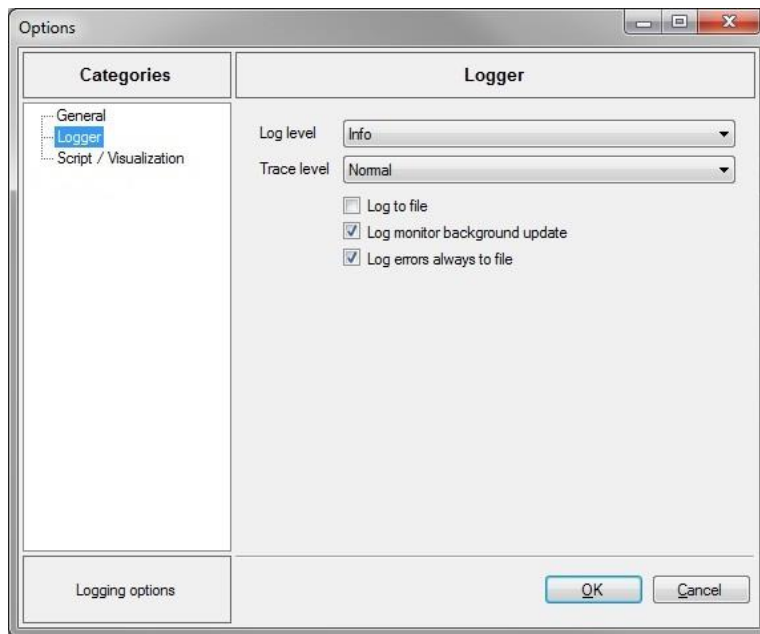


Figure 18: Logger Menu

Log Level

The log level for general event messages, warnings, errors etc. can be set from this page.

- ExtendedDebug
- Debug
- Trace
- Info
- Warning
- ErrorTrace
- CriticalWarning
- Error

Trace Level

The trace level for logging of sent or received messages can be set from this page. The options includes:

- Streaming
- Cyclic
- Normal

Log to File

If this option is activated, all event messages will be saved in a file. The name of the file will consist of the name of the Toolmonitor and a timestamp. The file will be stored in the same folder as the Toolmonitor executable file.

Log Monitor Background Update

Normally, event messages are always generated, even when displaying event messages has not been enabled in the monitor. If such is not desired, this feature can be deactivated using this option.

Log Errors Always to File

Normally, error messages will be written to an additional log file. If such is not desired, this feature can be deactivated using this option. The name of the file will consist of the Toolmonitor name and the literal "Exceptions". The file will be stored in the same folder as the Toolmonitor executable file.

7. Programming

7.1. Line

7.1.1. Class Description

This is the base class for serial communication using a variety of communication channels

7.1.2. Virtual Interface Commands

The following is a list of the commands for the *Virtual Interface* of the communications channel.

Issuing these commands (SetEvent, SetValue, GetValue or GetString, for example) will trigger events in the communications channel's Virtual Interface.

Command	Description
<i>Open</i>	Opens the communication channel.
<i>Close</i>	Closes the communication channel.
<i>Clear</i>	Deletes the data in the communication channel's send and receive buffers.
<i>IsActive</i>	Indicates if the communication channel in the basic settings is active.
<i>IsOpen</i>	Indicates if the communication channel has been opened.
<i>DataAvailable</i>	Indicates if data is ready for receipt. This command is not supported by all communication channels.
<i>BytesAvailable</i>	Returns the number of bytes ready for receipt. This command is not supported by all communication channels.
<i>Write</i>	Sends the indicated data through the communication channel.
<i>WriteMessage</i>	Sends the message stored under the indicated name through the communication channel.
<i>WriteSequence</i>	Sends the string stored under the indicated name through the communication channel
<i>Read</i>	Reads the data from the communication channel.
<i>Execute</i>	<p>Sends the indicated command through the communication channel for execution and saves the associated Response.</p> <p>This command is only intended for communication channels, which do not use separate read / write functions.</p>
<i>ExecuteMessage</i>	<p>Sends the command stored under the indicated name through the communication channel for execution and saves the associated Response.</p> <p>This command is only intended for communication channels, which do not use separate read / write functions.</p>
<i>ExecuteSequence</i>	<p>Sends the string stored under the indicated name through the communication channel for execution and saves the associated Response.</p> <p>This command is only intended for communications channels, which do not use separate read / write functions.</p>

<i>Response</i>	Returns the last message received about an execute command.
<i>TraceLevel</i>	Changes the trace level for sending and receiving through the Virtual Interface. Allowed values: Streaming = 0 Cyclic = 1 Normal = 2 None = 3
<i>Timeout</i>	Changes the timeout for sending and receiving through the Virtual Interface.
<i>Count</i>	Changes the byte count for sending and receiving through the Virtual Interface.
<i>Comment</i>	Changes the comment for sending and receiving through the Virtual Interface.
<i>EndOfLine</i>	Changes the "EndOfLine string" for sending and receiving through the Virtual Interface.
<i>ThrowTimeoutException</i>	Activates, or deactivates, the timeout exception when sending or receiving messages through the Virtual Interface.
<i>GetMessage</i>	Returns the message stored under the indicated name.