ELEKTRONIK GMBH

GET IN touch
WITH SENSITIVE TESTING

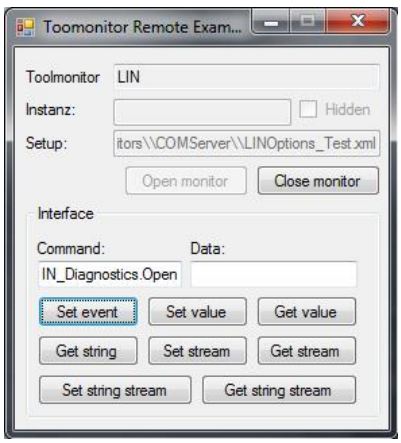
Softline
Modline
Conline
Boardline
Avidline
Pixline

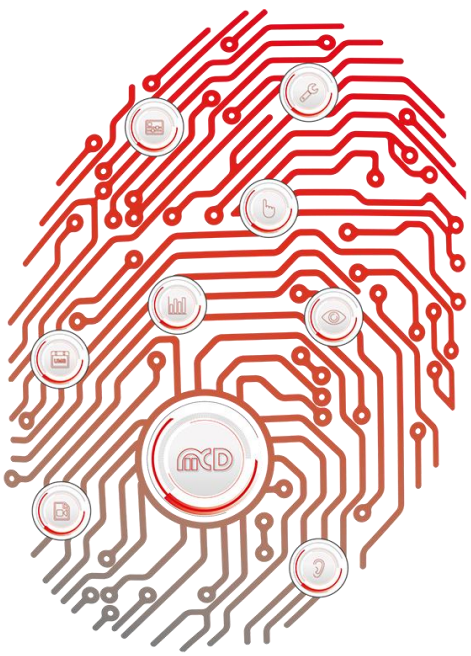
Application

Manual

Integration of a MCD Toolmonitor as .NET Assembly

Example in C# using Visual Studio®





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, Germany
Managing CEO: Bruno Hörter
Register Court Mannheim
HRB 505692

Template version: 5.4 / 2018-04-09
V1.1 2019-06-25 / BM (TS)

Content

1. General

All MCD Toolmonitors can be controlled remotely as .NET Assemblies. This example describes how a Toolmonitor can be controlled remotely using a simple C# class that calls virtual interface commands of the Toolmonitor. To do so, it is not necessary to reference the Toolmonitor in the Visual Studio® project. Registering them as COM server is sufficiant.

This document references the Visual Studio® project *ToolmonitorRemoteControlExample.sln* containing an exemplary user interface and a C# class called *RemoteController*.

2. RemoteController

The *RemoteController* is an exemplary C# class which can be used to control a Toolmonitor remotely. The source code is included in the example project and can be adjusted as needed. The functions of the class are being described in the following section.

2.1. OpenToolmonitor

The function “OpenToolmonitor” starts an instance of the specified Toolmonitor.

```
void OpenToolmonitor(string tool, string instance, string setup, bool hideOnStartup)
```

tool:	Name of the Toolmonitor. For example LIN.
instance:	In general, a Toolmonitor can be started several times. The instance name allows an instance of the Toolmonitor to be uniquely identifiable.
setup:	Optionally, the path to a setup file can be defined which will be loaded when the Toolmonitor is started. If an empty string is passed into the function, no setup file will be loaded.
hideOnStartup:	Setting this flag will start the Toolmonitor invisible.

2.2. CloseToolmonitor

Using the function “CloseToolmonitor” will close the previously opened Toolmonitor. The RemoteController class can only open one Toolmonitor and one instance of it at once. In general, a Toolmonitor can be started multiple times. To achieve this, the source code can be adjusted as needed.

```
void CloseToolmonitor()
```

2.3. Set

In the following chapter, the various “Set” functions of the interface are being described.

2.3.1. SetEvent

The “SetEvent” function calls the event with the specified name via the virtual interface of the Toolmonitor.

```
void SetEvent(string name)
```

```
name:          Name of the command, the user wants to call as event.
```

2.3.2. SetValue

The function “SetValue” can be used to set the specified value via the virtual interface of the Toolmonitor. Only strings can be passed on to the Toolmonitor. To set a numerical value, the value has to be converted to a string first.

```
void SetValue(string name, string value)
```

```
name:          Name of the value.
value:         The value the user wants to assign.
```

2.3.3. SetStream

The function “SetStream” can be used to pass a numeric stream into the Toolmonitor.

```
void SetStream(string name, double[] value)
```

```
name:          Name of the value.
value:         The numeric stream the user wants to assign.
```

2.3.4. SetStringStream

The function “SetStream” can be used to pass a string stream into the Toolmonitor.

```
void SetStringStream(string name, string[] value)
```

```
name:          Name of the value.
value:         The stream the user wants to assign.
```

2.4. Get

In the following chapter, the various “Get” functions of the interface are being described.

2.4.1. GetValue

Gets the specified numeric value via the virtual interface of the Toolmonitor.

```
double GetValue(string name)
name:           Name of the value to be requested.
```

2.4.2. GetString

Gets the specified string value via the virtual interface of the Toolmonitor.

```
string GetString(string name)
name:           Name of the value to be requested.
```

2.4.3. GetStream

Gets the specified numeric stream via the virtual interface of the Toolmonitor.

```
double[] GetStream(string name)
name:           Name of the stream to be requested.
```

2.4.4. GetStringStream

Gets the specified string stream via the virtual interface of the Toolmonitor.

```
string[] GetStringStream(string name)
name:           Name of the stream to be requested.
```

3. Example application

The provided example program illustrates on the basis of a simple user interface how a Toolmonitor can be controlled remotely. The corresponding source code is available and can be found attached to this documentation.

Referencing the Toolmonitor you want to control remotely in the project is not necessary. It is sufficient to register the Toolmonitor as COM server.

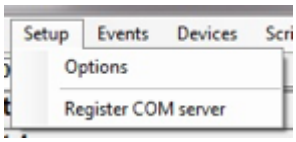


Figure 1: Registration as COM Server

To start a Toolmonitor using the example application, type in the name of the Toolmonitor into the first text box. For example LIN, SerialLine, etc. Additionally, an instance name can be added if several instances of the Toolmonitor are required. This example application can only start a single instance of a Toolmonitor at once, but the code shows the concept of how several instances of the same Toolmonitor can be started. Furthermore, it is possible to start the monitor hidden and add the path to a setup file which will be loaded when the Toolmonitor is started.

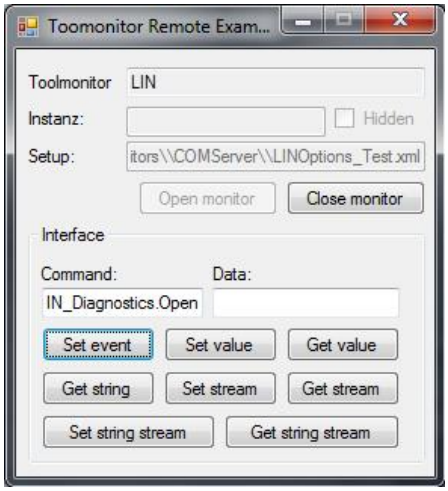


Figure 2: Entry of the Toolmonitor Name and Setup

Once the Toolmonitor is started, virtual interface commands can be executed through the user interface. To do so, type the name of the virtual interface command into the “Command” text box. For “Set” commands add the values into the “Data” field. To send streams split the values with semicolons. When executing a “Get” command, the result is going to be shown in the “Data” field. When executing a “SetEvent” command, the “Data” field is ignored.

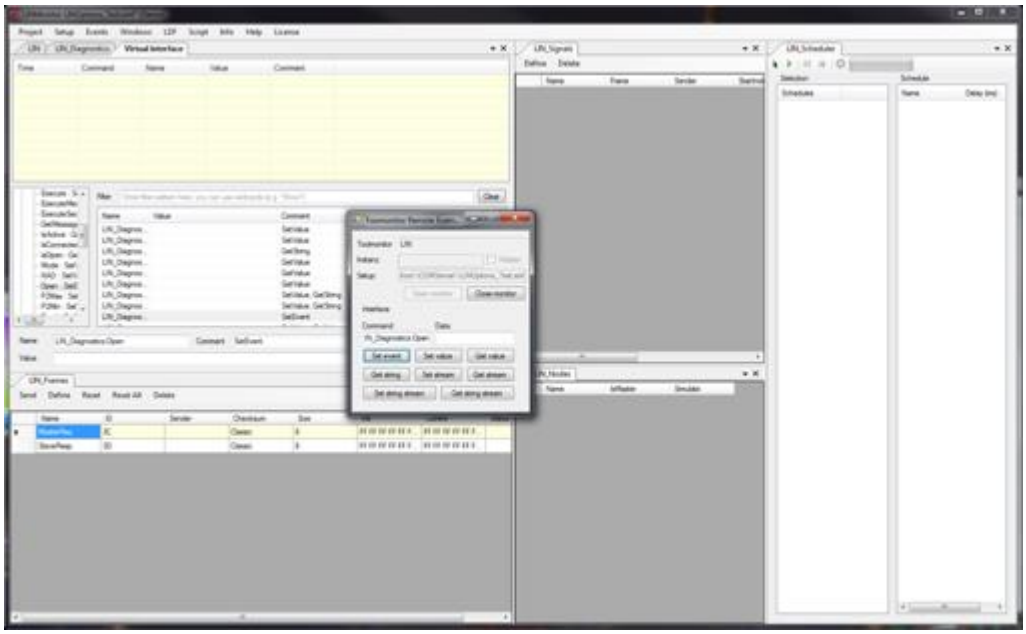


Figure 3: User Interface