

FieldTalk™ Modbus Master Library for .NET

FT-MBMP-NET-EVAL Read Me Notes

Revision 2.15.2, 2025-04-03

Welcome to the evaluation of FieldTalk™ Modbus Master Library for .NET!

This evaluation package allows you to:

- Read the product manual
- Browse the Modbus Master library API documentation
- Study the source code of the Modbus examples provided
- Modify and compile the Modbus examples
- Create your own Modbus programs and test them using the library

Documentation

To browse the *API Documentation* please click here [[doc/html/index.html](#)].

To read or print the *Product Manual* please click here [[doc/FieldTalk.Modbus.Master.pdf](#)].

Please refer to the additional *Application Notes* published on our web site: <http://www.ModbusDriver.com/doc>

Limitations and Restrictions

The evaluation version is functional limited to 100 modbus operations. Once 100 operations are reached, all library functions will return error code 3 `FTALK_EVALUATION_EXPIRED`.

In addition it does not contain any redistributable components. You must not redistribute any source, object or library files.

This package is provided for the purpose of evaluating whether to purchase an ongoing license to use the Package.

THE EVALUATION PERIOD IS LIMITED TO 30 DAYS and does not include the right to reproduce and distribute Applications using the Package. At the end of the evaluation period, if You do not purchase a license, You must uninstall the Package from the computers or devices You installed it on.

THE EVALUATION VERSION IS FUNCTIONAL LIMITED TO 100 MODBUS OPERATIONS!

Files part of the package

README, README.pdf

These Read Me notes.

`LICENSE-LIB, LICENSE-LIB.pdf`

Library License containing the Terms & Conditions of use of this software.

`doc\FieldTalk.Modbus.Master.pdf`

Documentation in PDF format.

`doc\html\index.html`

Documentation in HTML format.

`samples`

Directory with example programs as a starting point for your application.

`lib\FieldTalk.Modbus.Master.2.15.2.0.nupkg`

NuGet package with

- .NET interop libraries for .NET/.NET Core (with .NET Standard ≥ 2.0) and classic .NET Frameworks 4.8, 4.7, 4.6, 4.5, 4.0 and Mono 5.x, 4.x.
- Native code DLLs for Windows x86, x64, Arm32 and Arm64 processors
- Native code shared libraries for Linux x64, Arm32 and Arm64 processors

`lib\net`

.NET interop libraries for .NET 9.0, 8.0, 7.0, 6.0, 5.0 and .NET Core (with .NET Standard ≥ 2.0)

`lib\net40`

.NET interop libraries for classic .NET Frameworks 4.8, 4.7, 4.6, 4.5, 4.0 and Mono 5.x, 4.x.

`lib\net20`

.NET interop libraries for classic .NET Frameworks 3.5, 3.0, 2.0

`lib\netcf20`

.NET interop libraries for .NET Compact Frameworks 3.9, 3.7, 3.5, 2.0

`runtimes\win-...\native\libmbusmaster.dll`

Native code DLLs for Windows x86, x64, Arm32 and Arm64 processors

`runtimes\linux-...\native\libmbusmaster.dll`

Native code shared libraries for Linux x64, Arm32 and Arm64 processors

`runtimes\wce500-...\native\libmbusmaster.dll`

Native code DLLs for Windows CE 5.0, 6.0 for x86 and ARMv4

`runtimes\wce800-...\native\libmbusmaster.dll`

Native code DLLs for Embedded Compact 2013 (aka CE 8.0) for x86 and ARMv7 processors

Distributable components

This is an evaluation edition and does not contain any redistributable components.



You must not redistribute any executables, source, object or library files which are based on this evaluation edition.

Release history

2.15.2 (2025-04-03)

Bug fixes

- Version 2.15.0 and 2.15.1 have wrong (older) native code DLL which is missing the read/write record and 64-bit export functions

2.15.1 (2025-04-02)

Bug fixes

- 64-bit data types (int64_t, double) support was missing for FC4 readInputRegisters

2.15.0 (2025-02-26)

Enhancements

- First class support for .NET 9, fixed warning NETSDK1206 when compiling against .NET Core 8 and 9
- Added assembly and runtime files separate to nuget file

Bug fixes

- readDeviceIdentification() for a slave conformity level of 1 triggered wrongly an FTALK_INVALID_REPLY_ERROR

2.14.0 (2024-07-27)

Enhancements

- Added support for 64-bit data types (int64_t, double) for FC3, FC4 and FC16

Bug fixes

- readFileRecord() with a record count of 2 or less triggered FTALK_ILLEGAL_ARGUMENT_ERROR

2.13.0 (2024-05-14)

Enhancements

- Added Modbus function 43 (hex 2B) subfunction 14 (hex 0E), Read Device Identification
- Added Modbus function 20 (hex 14) Read File Record
- Added Modbus function 21 (hex 15) Write File Record

2.12.1 (2024-01-05)

Enhancements

- Updated .NET core examples to .NET Core 8

2.12.0 (2023-11-11)

Enhancements

- Added Lufkin ELAM protocol variant encapsulated in TCP (ELAM over TCP)

2.11.3 (2023-10-06)

Enhancements

- Added setClosingTimeout method for synchronous socket closure
- TCP/IP connection closing can be made synchronous by configuring a closing time-out. This then prevents the master from opening a new socket before the slave has acknowledged that the old socket has been closed.
- Lufkin ELAM protocol now supports variable length replies
- Added validation for implied reply length against actual reply length
- Modbus/TCP: An FTALK_INVALID_MBAP_ID error will force close Modbus/TCP connection
- Modbus/TCP: Added check for extraneous reply bytes in TCP data stream

Bug fixes

- Fixed misspelled FTALK_MBUS_ILLEGAL_ADDRESS_RESPONSE constant
- Fix bug in serial ELAM protocol which reports wrong exception code
- Fix bug in serial ELAM protocol which configures only the first 255 slaves when using configure...() functions (like configureCountFromZero())
- Modbus/TCP: Fix wrong invalid MBAP ID/invalid frame indication for the following frame if extraneous characters are sent in the TCP stream

2.11.2 (2022-05-12)

Bug fixes

- Method `configureCountFromZero(Int32 slaveAddr)` was misspelled as `onfigureCountFromZero`
- Fixed compilation for .NET 2.0

Enhancements

- Variable length replies are now supported for RTU and TCP

API changes

- customFunction's responseData array is now passed as ref

2.11.0 (2021-04-02)

Enhancements

- Added .NET Core support
- Added Windows 10 UWP ARM64 platform
- Added Linux ARM32 platform
- Moved buffer clearing after poll delay period expires

2.10.5 (2020-07-15)

API changes

- Added IDisposable interface so the using statement can be used.

2.10.4 (2019-08-02)

Bug fixes

- Fixed "mbusSlave_get_ErrorText" regression introduced in 2.10.3

2.10.3 (2019-03-15)

Enhancements

- Added .NET Core support for UWP for x86, x64 and ARM

Bug fixes

- Modbus/TCP only: readExceptionStatus() always returned FTALK_REPLY_TIMEOUT_ERROR

2.10.2 (2019-02-25)

Enhancements

- Added .NET Core support for Linux ARM32 (Raspberry Pi et al)

- TCP/IP connection closing can be made synchronous by configuring a closing time-out. This then prevents the master from opening a new socket before the slave has acknowledged that the old socket has been closed.

2.10.1 (2018-11-07)

Bug fixes

- Fixed bug converting COM-style port names on WinCE

2.10.0 (2018-04-06)

Enhancements

- Added Modbus/UDP protocol variant
- .NET Core support added
- Windows Embedded Compact support added
- Library is now a NuGet package
- Updated all examples to VS2017 project files
- Added bit functions to RegisterConverter class

API changes

- Renamed `MbusSerialMasterProtocol` class to `MbusSerialClientBase`. To compile old code using this base class, add using `MbusSerialMasterProtocol = FieldTalk.Modbus.Master.MbusSerialClientBase;`
- Renamed `MbusTcpBaseProtocol` class to `MbusIpClientBase`

Other changes

- Assembly name changed to `FieldTalk.Modbus.Master.dll`
- With the exception of Windows CE, support for Visual Studio 2005/2008 has been discontinued and the relevant project/solution files removed.

2.9.1 (2016-06-16)

Bug fixes

- Fixed reply array size mismatch in RTU and ASCII implementation of customFunction

2.9.0 (2015-12-17)

Enhancements

- Lufkin ELAM protocol flavour added

2.8.0 (2015-05-12)

Enhancements

- Modbus ASCII over TCP protocol flavour added

API changes

- New base class MBusTcpBaseProtocol introduced, MBusRtuOverTcpProtocol now inherits from MBusTcpBaseProtocol rather MBusTcpProtocol.

2.7.1 (2015-03-24)

Enhancements

- DLL is now compiled against static runtime library. The previous 2.7.0 version required the Visual C++ 8.0 (CLR 2.0) or 10.0 (CLR 4.0) run-time to be installed.

2.7.0 (2014-09-16)

Enhancements

- writeSingleRegister for UInt16 data type added
- Added VS2010 and VS2012 solution files for examples
- Added support for .NET framework versions 4.x (CLR version 4 based frameworks)
- Added RegisterConverter class

API changes

- Namespace changed from *FieldTalk* to *FieldTalk.Modbus.Master*

2.6.2 (2013-01-11)

Enhancements

- Slave ID of 0 is supported for Modbus/TCP
- COMn will now automatically be substituted with \\.\COMn for more convenient access to serial ports ≥ 10
- Added support for Daniel/Enron style 32-bit mode.
- Added support for PDU based register addressing model.
- Endianess can now be configured on a per slave basis.
- customFunction method added for Modbus/TCP. This allows implementation of custom Modbus function codes.

- Modbus function 8, sub-function 0 `returnQueryData` method added.
- Modbus function 8, sub-function 01 `restartCommunicationsOption` method added.
- Added new function `adamSendReceiveAsciiCmd` for MODBUS/TCP to support ADAM 5000/6000 ASCII commands.

Bug fixes

- `MbusTcpMasterProtocol`'s `hostName` property was not working

2.6.1 (2009-12-15)

Enhancements

- Assembly is now signed and strong-named

2.6.0 (2009-11-26)

API Changes

- `readMultipleRegisters`, `writeMultipleRegisters` and `readInputRegisters` dynamically sense the array type and automatically adjust for 16 and 32-bit lengths. Accepted array types are `Int16[]`, `UInt16[]`, `Int32[]`, `UInt32[]` and `float[]`. This also makes it easier to use for unsigned data as no cast is required.
- Defaults for serial protocols changed to 19200 and even parity as per Modbus specification.
- Updated maximum registers and coils to match latest Modbus specifications: Function code 16 supports now 123 registers, was 100 before. Function code 15 supports now 1968 coils, was 800 before. Also updated max values for function code 23.
- `readMultipleLongInts`, `writeMultipleLongInts`, `readInputLongInts`, `readMultipleFloats`, `writeMultipleFloats` and `readInputFloats` are deprecated. Use `readMultipleRegisters`, `writeMultipleRegisters` and `readInputRegisters` instead.
- Removed some redundant and rarely used overloaded methods:
`MbusTcpMasterProtocol(string hostName, Int16 portNo)`,
`MbusRtuMasterProtocol(string portName, Int32 baudRate, Int32 dataBits, Int32 stopBits, Int32 parity)`,
`MbusAsciiMasterProtocol(string portName, Int32 baudRate, Int32 dataBits, Int32 stopBits, Int32 parity)`,
`openProtocol(string portName, Int32 baudRate)`, `openProtocol(string hostName, Int16 portNo)`

Enhancements

- Added VS2008 and 64-bit solution files for examples
- Added support for .NET compact framework 3.5
- Updated and added examples

- Windows platform does not require the second DLL `libmbusmaster.dll` anymore. The MSIL code and the native code has been combined into one hybrid DLL `mbusmaster.net.dll` (Windows CE still requires a MSIL and native DLL to be deployed).
- Using relative timers on Windows instead of absolute system time for time-out control.
- Improved RS-485 support, delay not required anymore, `enableRs485Mode` can now be used reliably with a delay of 0.

Documentation

- Fixed error in documentation of Modbus function 22 / `maskWriteRegister`.

Bug fixes

- Fixed double time-out time for Modbus/TCP in case of a broken network connection. The time-out is now applied only once.
- Destructor call did not close the socket.
- TCP connections are now closed gracefully using a FIN/ACK sequence rather a RST.

Other changes

- Support for Visual Studio 2003 and .NET 1.1 has been discontinued and the relevant project file removed.
- Support for .NET 1.0 compact framework has been discontinued.

2.4.0 (2006-10-2)

- Based on native `libmbusmaster` library 2.4.0 functionality.
- New directory structure.

Enhancements

- Added run-time check for `libmbusmaster.dll`.
- Dot NET 2.0 CLS compliant.
- Compatible with WinCE.net 4.1.
- New solution files for better support of Visual Studio 2003 and 2005.
- C# examples added.

2.3.3 (2005-11-11)

Enhancements

- RTS is set when opening port and not in RS-485 mode for better support of handshake line powered devices.

Bug fixes

- Fixed import issue of libmbusmaster.dll for Windows CE.
- Fixed getTimeout bug.
- Fixed issue that readInputLongInts called readMultipleLongInts and vice versa.

2.3.2 (2005-08-05)

Bug fixes

- Fixed bug in VB.net method forceMultipleCoils() which did not write first coil

2.3.1.1 (2005-07-27)

- First release which included .NET interface.