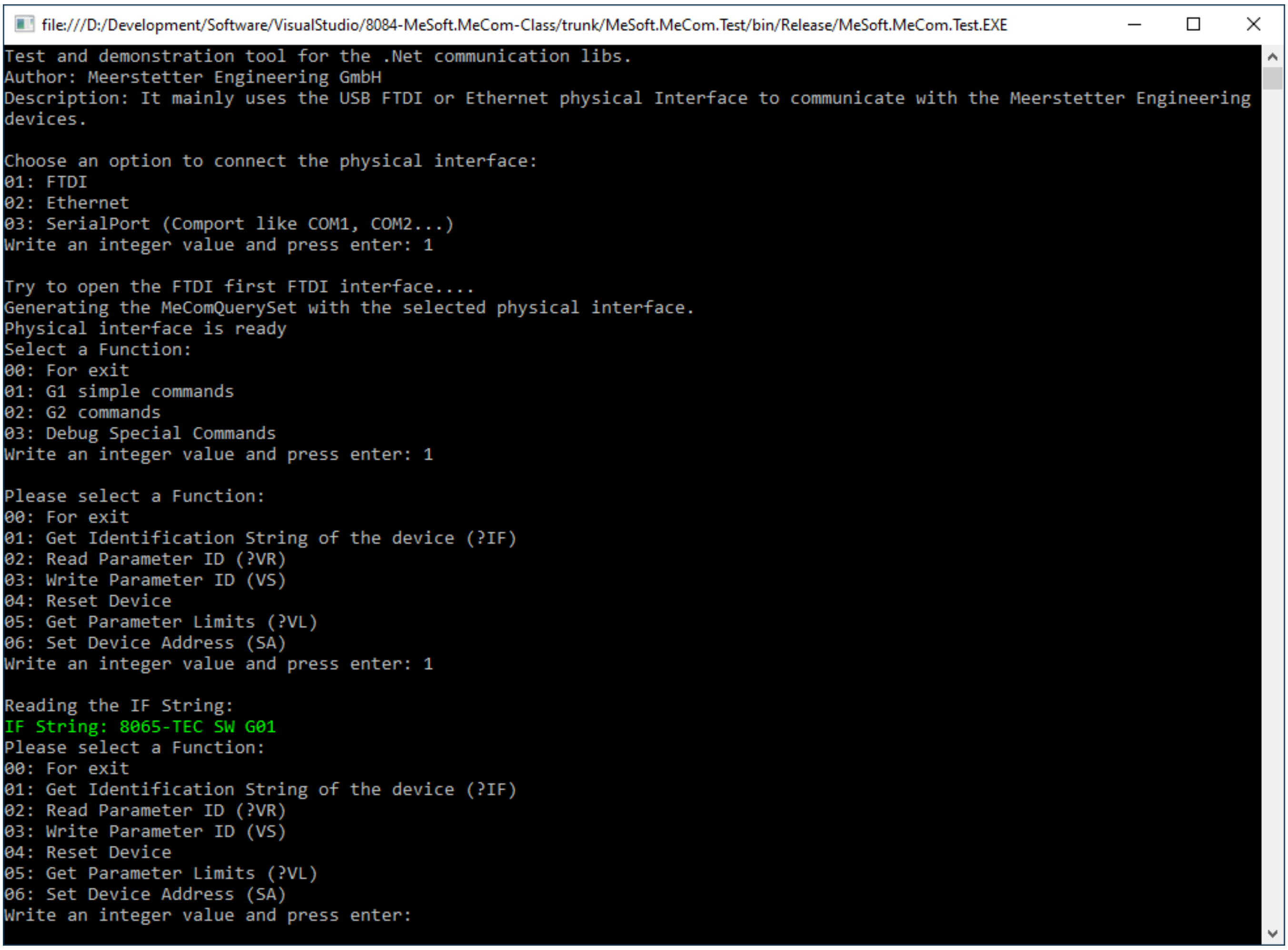
MeComAPI for .NET

Communication API for
LDD, TEC, LTC Families
and LTR Rack Enclosures

The package consists of a C# Code Library and a sample application.
The pre-compiled DLLs can also be used for a Visual Basic project.

The application exemplifies the control of LDD-, TEC- and LTC-Family devices over a USB FTDI, Ethernet Interface or a simple SerialPort (COMxy)


file:///D:/Development/Software/VisualStudio/8084-MeSoft.MeCom-Class/trunk/MeSoft.MeCom.Test/bin/Release/MeSoft.MeCom.Test.EXE

```
Test and demonstration tool for the .Net communication libs.
Author: Meerstetter Engineering GmbH
Description: It mainly uses the USB FTDI or Ethernet physical Interface to communicate with the Meerstetter Engineering
devices.

Choose an option to connect the physical interface:
01: FTDI
02: Ethernet
03: SerialPort (Comport like COM1, COM2...)
Write an integer value and press enter: 1

Try to open the FTDI first FTDI interface....
Generating the MeComQuerySet with the selected physical interface.
Physical interface is ready
Select a Function:
00: For exit
01: G1 simple commands
02: G2 commands
03: Debug Special Commands
Write an integer value and press enter: 1

Please select a Function:
00: For exit
01: Get Identification String of the device (?IF)
02: Read Parameter ID (?VR)
03: Write Parameter ID (VS)
04: Reset Device
05: Get Parameter Limits (?VL)
06: Set Device Address (SA)
Write an integer value and press enter: 1

Reading the IF String:
IF String: 8065-TEC SW G01
Please select a Function:
00: For exit
01: Get Identification String of the device (?IF)
02: Read Parameter ID (?VR)
03: Write Parameter ID (VS)
04: Reset Device
05: Get Parameter Limits (?VL)
06: Set Device Address (SA)
Write an integer value and press enter:
```

Demo Application

1 General Description

1.1 General

- The MeComAPI for .Net provides C# code to fully control LDD-, TEC and LTC - Family devices.
- Please have a look at the inline XML comments for more details.
- The user will only need to call some simple functions to set or read parameters.
- The MeComAPI for .Net does everything that is necessary to have a reliable communication interface:
  - Implements nearly all functions used to communicate with the devices.
  - CRC calculations and checks
  - Sequence Number monitoring
  - Data resend on timeout and error management
- To create a trace MeSoft.Core.dll is also linked to the projects. In the standard configuration the whole serial communication is being traced to the trace file, which is located next to the .exe file. By changing the settings in the .confg file, it is also possible to disable the full trace out. Please contact us for further information.

1.2 Documents and Versions

This project shows the C-code implementation of the following specification documents:

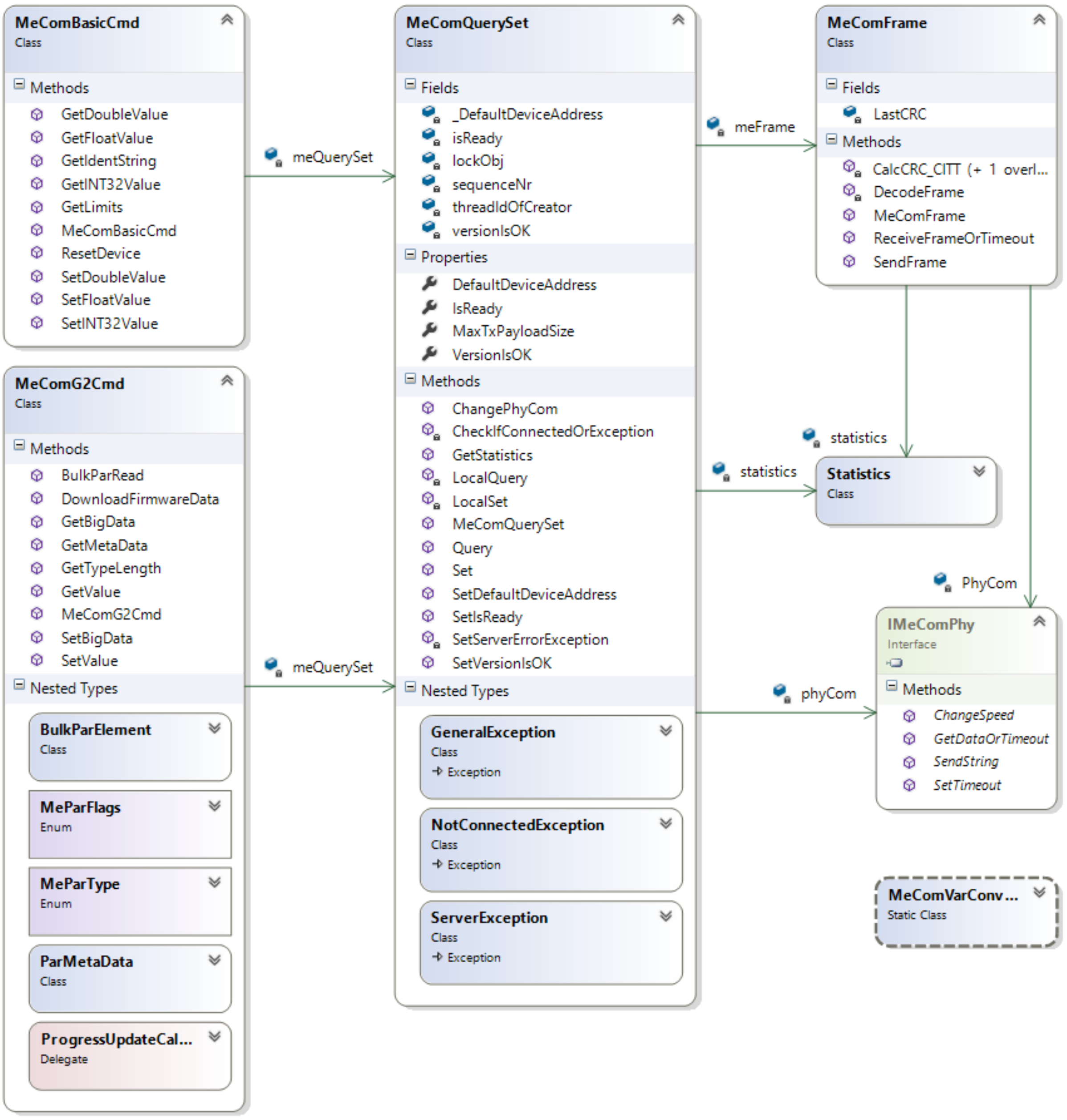
- MeCom Protocol Specification 5117C
- Implements most functions of the LDD and TEC Communication Specifications:
  - Laser Diode Driver Communication Protocol 5130
  - TEC Controller Communication Protocol 5136
  - LTC Communication Protocol 5199

2 Class Diagrams of the MeComAPI for .Net

2.1 MeSoft.MeCom.Core

The core part of the Library provides the main Functions to communicate with the devices.

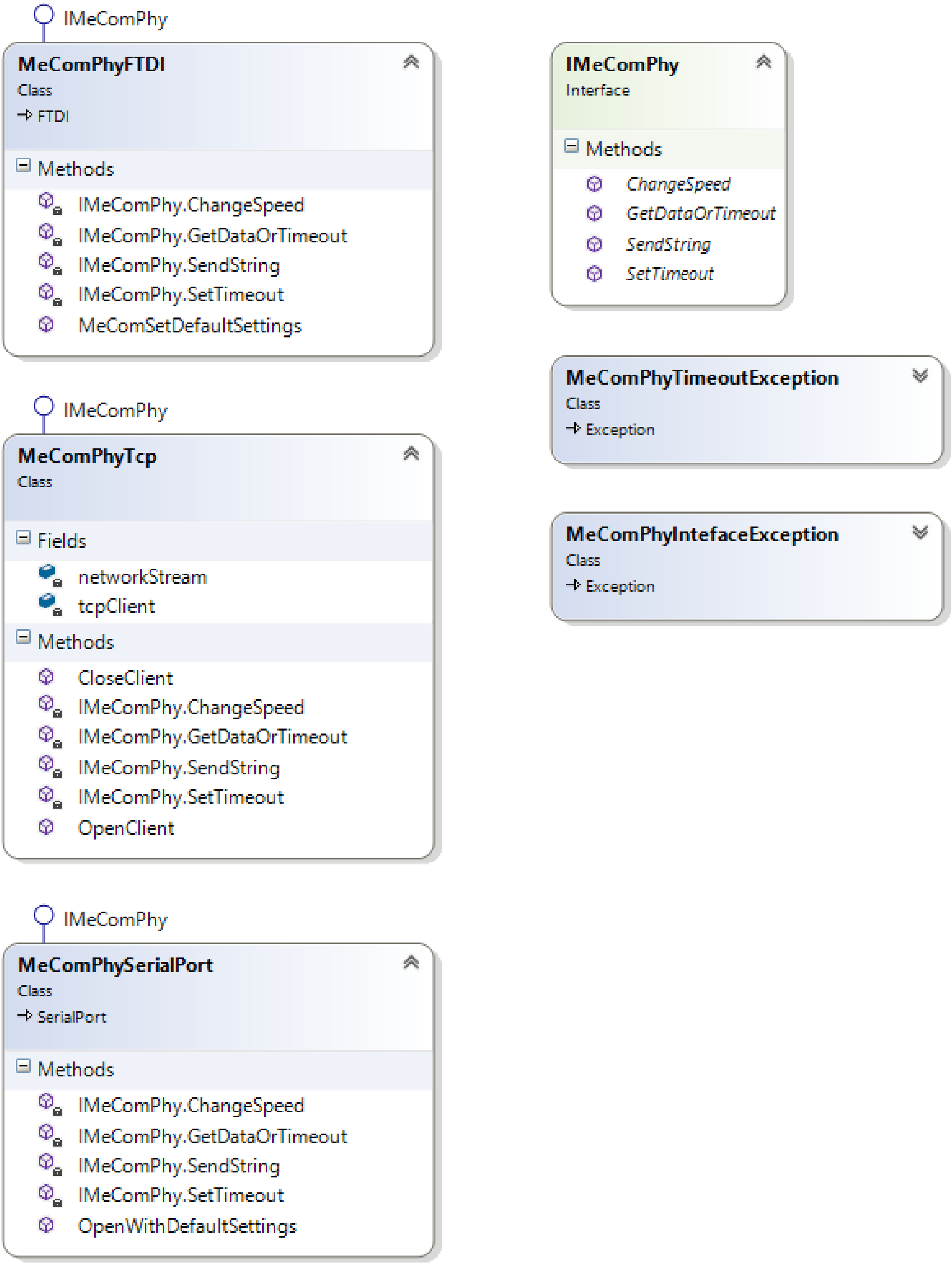
- The classes MeComBasicCmd and MeComG2Cmd provide the top functions to be used.
- The MeComQuerySet represents the layer that is responsible that the data is transferred correctly. It does automatically re-query if no data returns or the returned data is corrupted. It does also handle multi-threading queries to the same interface.
- The MeComFrame creates and decodes the effective package and transfers it to the physical layer.
- The Statistics part provides some statistics about the communication.



2.2 MeSoft.MeCom.PhyWrapper

The PhyWrapper part is responsible for connecting the physical layer.

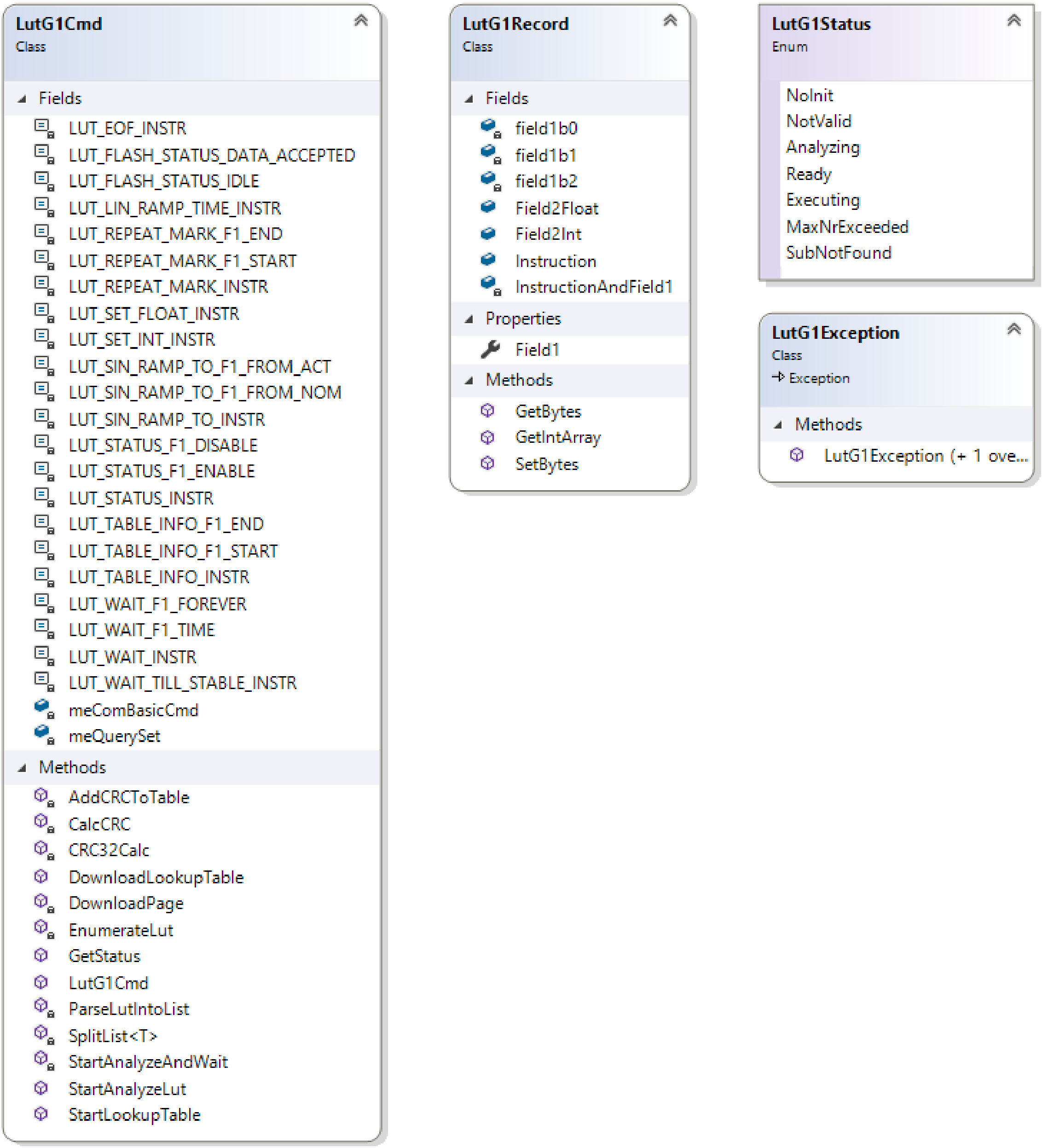
Currently, only USB FTDI Interfaces and TCP connections are supported, but the project is simple extendable for any other physical interfaces. For example, RS232 / RS485 interfaces. Please contact us…



2.3 MeSoft.MeCom.TEC

The TEC part contains all functionality that is specific to TEC devices only.

Currently, this project only contains the functions required to download lookup tables onto a TEC controller.



LutG1Cmd
Class

Fields
- LUT_EOF_INSTR
- LUT_FLASH_STATUS_DATA_ACCEPTED
- LUT_FLASH_STATUS_IDLE
- LUT_LIN_RAMP_TIME_INSTR
- LUT_REPEAT_MARK_F1_END
- LUT_REPEAT_MARK_F1_START
- LUT_REPEAT_MARK_INSTR
- LUT_SET_FLOAT_INSTR
- LUT_SET_INT_INSTR
- LUT_SIN_RAMP_TO_F1_FROM_ACT
- LUT_SIN_RAMP_TO_F1_FROM_NOM
- LUT_SIN_RAMP_TO_INSTR
- LUT_STATUS_F1_DISABLE
- LUT_STATUS_F1_ENABLE
- LUT_STATUS_INSTR
- LUT_TABLE_INFO_F1_END
- LUT_TABLE_INFO_F1_START
- LUT_TABLE_INFO_INSTR
- LUT_WAIT_F1_FOREVER
- LUT_WAIT_F1_TIME
- LUT_WAIT_INSTR
- LUT_WAIT_TILL_STABLE_INSTR
- meComBasicCmd
- meQuerySet

Methods
- AddCRCToTable
- CalcCRC
- CRC32Calc
- DownloadLookupTable
- DownloadPage
- EnumerateLut
- GetStatus
- LutG1Cmd
- ParseLutIntoList
- SplitList<T>
- StartAnalyzeAndWait
- StartAnalyzeLut
- StartLookupTable

LutG1Record
Class

Fields
- field1b0
- field1b1
- field1b2
- Field2Float
- Field2Int
- Instruction
- InstructionAndField1

Properties
- Field1

Methods
- GetBytes
- GetIntArray
- SetBytes

LutG1Status
Enum
- NoInit
- NotValid
- Analyzing
- Ready
- Executing
- MaxNrExceeded
- SubNotFound

LutG1Exception
Class
→ Exception

Methods
- LutG1Exception (+ 1 ove...

3 Change Log

Changed	Dok	API Version	Compatible with / Change Log
12 July 2016	A	1.10	Initial Release. Tested with TEC Firmware Version 1.70 LTC-114x Firmware Version 0.30
8 Nov 2016	B	1.20	• Add: Option to connect to a SerialPort like COM1, COM2, … • Bug: SetIsReady(true) added in the demo application. Without this it is not possible to access the com functions from a different thread.
26 Jan 2017	C	1.30	• The MeSoft.Core lib has been added. This is used to output a trace. In the standard configuration, the demo application writes the whole serial communication to a trace.txt file next to the .exe file. By changing the trace level, it is also possible to disable the full printout. • A lot of the Get functions used out parameters to return the queried values. These out parameters have been replaced by the normal return parameters. This may be a breaking change for your application if you just update. • An example to get the data from the LTC-1141 has been added. • The functions do not expect an answer if you send to the address 255. This is a new definition for broadcasting. Till now, only the TEC-Family Firmware does support it in the version 3.00.
29 June 2018	D	1.40	• Add: ChangeComSpeed command (CS) • Big Data Commands (VB / ?VB): ○ Add: BYTE Type ○ Bug: SetBigData failed, when no feedback function was specified ○ Add: Send and receive ASCII strings. Type is called LATIN1 • Add: MeSoft.MeCom.SCPI Project. This is a project which might be used as interface to our TEC Controllers, but it is not yet sure if we want to continue this project. It was used for a customer who wanted to control our TEC Controller from a Excel VBA script. • Add: Set device address command (SA)
14 April 2019	E	1.45	• Add: DownloadSettingsDumpFile • Add: Message if a package must be re-sent • Bug: SCPI did not start the internal thread • Bug: GetMetaData did not support Latin1 and Byte types • Add: LTC-1141 special test command (reset and check status while) • Add: GetBigData ProgressUpdateCallback and expectedNrOfElements • Bug: MeComG2Cmd SetBigData nr of package calculation wrong (caused package loss in conjunction with LTC-1141 Lookup Table download)
06 October 2020	F	1.47	• Add: Lookup table download functionality for TEC controllers