Kern & Sohn GmbH
Ziegelei 1
D-72336 Balingen
Email: info@kern-sohn.com
Tel: + 49- [0] 7433- 9933-0
Fax: + 49- [0] 7433-9933-149
Internet: www.kern-sohn.com

Instruction manual
KERN PC software

Balance Connection

Software Version 4.2.1

(KERN SCD-4.0)

01/2016
GB

i	You can find the latest version of this guide online at: http://balanceconnection.kern-sohn.com/Manual.pdf

SCD-BA-e-1546

KERN PC software Balance Connection

GB

Software version 4.2.1

Instruction manual

Index

1 Hardware and software requirements ... 5
2 Demo version ... 5
3 Installation ... 6
4 Program start ... 9
5 Simple mode / Quick Start ... 10
5.1 Data transfer from a KERN balance to Microsoft Excel ... 10
5.2 Take readings in a text file ... 12
5.3 Window "Select application" ... 15
5.3.1 Save and automatic detection of the target window ... 15
5.4 “Balance“ window ... 17
5.4.1 Select predefined type of balance ... 17
5.4.2 Edit the available types of weighing equipment ... 18
5.5 “Interface settings“ window ... 18
5.6 “Data transfer“ window ... 20
6 Expert Mode - Overview ... 21
6.1 Device types and device instances ... 21
6.2 (Hardware) interfaces (ports) ... 21
6.3 Inputs ... 22
6.4 Protocols / pattern recognition (Parser) ... 22
6.5 Outputs ... 22
6.5.1 Output patterns (templates) ... 23
6.5.2 Output Filters ... 23
6.5.3 Output scenarios ... 23
6.6 Triggers ... 23
6.7 Command sequences ... 23
7 General user interface in Expert Mode ... 24
7.1 Window / docking ... 24
7.2 Main toolbar and menubar ... 25
7.2.1 Menubar ... 25
7.2.2 Main Toolbar ... 27
7.3 Autostart with Windows ... 27
7.4 Program Log ... 28
8 Device / Device Types ... 29
8.1 Manage device instances ... 29
8.1.1 Add / Delete device instance ... 29
8.2 Manage device types ... 30
8.3 Text protocols ... 30
8.3.1 Overview ... 30
8.3.2 Management of interface protocols ... 31

8.4 Device operations ... 37
9 Interfaces (ports) ... 38
9.1 Overview ... 38
9.2 General configuration ... 38
9.2.1 Add a port ... 39
9.2.2 Properties of an input and port ... 39
9.2.3 Tools ... 41
9.3 RS-232 ports ... 42
9.4 RS-485 ports ... 42
9.5 IP network (TCP / IP, UDP / IP, Ethernet) ... 43
9.5.1 TCP or UDP / IP client (connecting) ... 43
9.5.2 Server (listening / waiting) ... 43
9.6 Bluetooth ... 44
10 Outputs / output methods - General ... 45
10.1 Overview ... 45
10.2 Configuration of outputs ... 46
10.2.1 General properties ... 46
10.2.2 Data Sources ... 46
10.2.3 Output Scenarios ... 46
10.3 Text Formatting (output patterns / templates) ... 48
10.3.1 Rules on output patterns ... 48
10.3.2 First example of an output pattern ... 49
10.3.3 Configure a simple pattern ... 49
10.3.4 Configure complex output patterns ... 50
11 Output methods in detail ... 51
11.1 Key simulation (keyboard wedge) ... 51
11.1.1 Global keys simulation (active application) ... 51
11.1.2 Targeted key simulation (to a particular application) ... 51
11.2 On-Screen Displays / In program displays ... 53
11.2.1 Big display ... 54
11.2.2 Recording in a text log ... 54
11.2.3 Chart ... 55
11.2.4 Table ... 56
11.3 File record ... 56
11.4 Forwarding to another port (e.g. RS-232 printer) ... 57
11.5 Forwarding to a Windows printer ... 57
11.6 Databases / ODBC / SQL queries ... 57
11.7 Start a program with arguments ... 58
11.8 HTTP / Website / Webservice ... 58
11.9 Transfer data to Microsoft Excel ... 59
11.9.1 Excel Output Mode "tabular output" ... 60
11.9.2 Excel output mode "fill cell ranges" ... 60
11.9.3 Excel Output Mode "separate text into cells" ... 61
12 Output Filters ... 62
12.1 General information about filters ... 62
12.2 Types of filters ... 62
12.2.1 Synchronisation (-> Table) ... 62
12.2.2 Each n-th value / All x milliseconds ... 62
13 Triggers / Queries ... 63

13.1 Overview 63
13.2 General information about triggers 63
13.3 Hotkeys / shortcuts 64
13.4 Timers / time controlled requests 64
14 Command sequences 65
15 Application and configuration examples 66
15.1 Print a user-defined label on a YKC-01 printer 66
15.2 Drying protocol with the drying process for a moisture analyzer 67
16 Software Update 70
17 Common Errors / Troubleshooting 71
Appendix I. Software License Agreement 72

1 Hardware and software requirements

Software requirements:

- Operating system: Microsoft Windows XP from (Windows Vista, 7, 8, 8.1, 10)
- Microsoft .NET Framework 4.0 (included)

Hardware requirements:

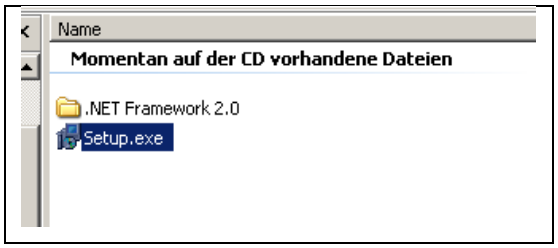
- free serial port (RS232), or
- Ethernet interface, or
- Bluetooth interface

2 Demo version

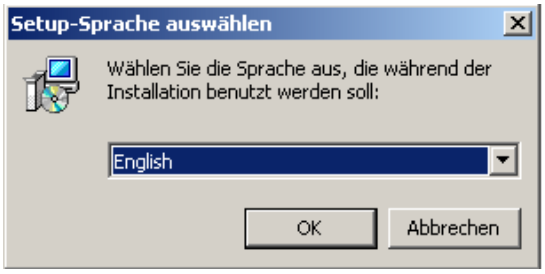
You can find the demo version of the Balance Connection software within KERN Webshop in the download area (KERN SCD-4.0). It can be used for 10 days (days on which the software has not been started are not included).

3 Installation

⇨ Insert the CD into the CD-ROM drive.



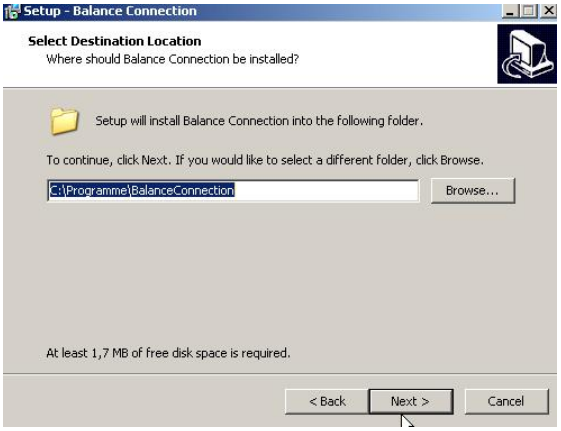
⇨ Start the file "setup.exe".



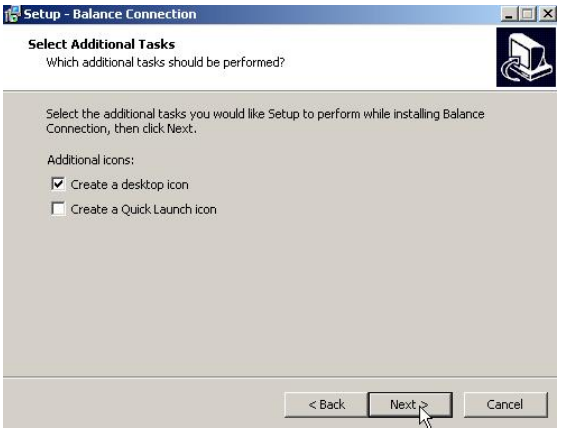
⇨ Select language (German⇆ English)

⇨ Continue by clicking on **CONTINUE**,

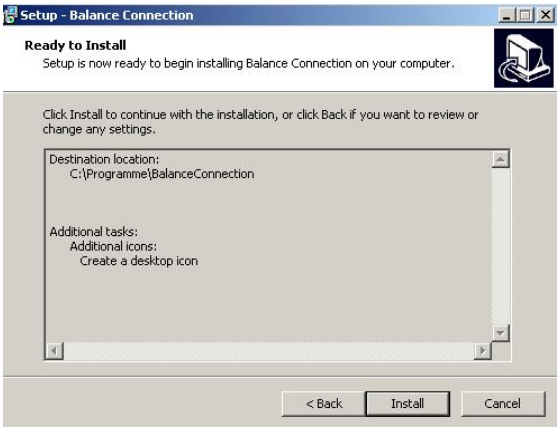
⇨ Accept the software license agreement by clicking "I accept the agreement". Continue by clicking on **CONTINUE**,



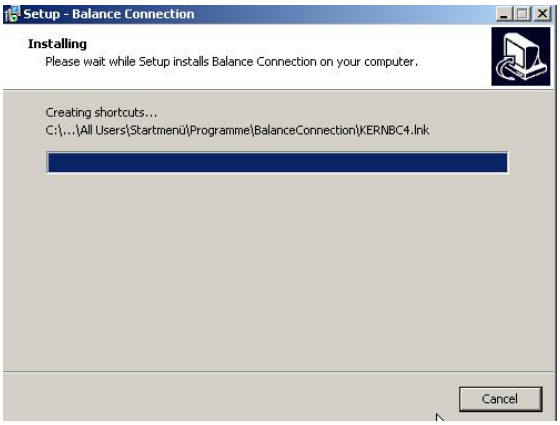
⇨ Select installation folder. Continue by clicking on **CONTINUE**,



⇨ Select additional tasks as desired.



⇨ By clicking on**TO INSTALL** start installation

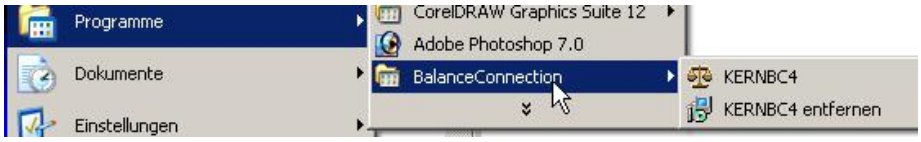


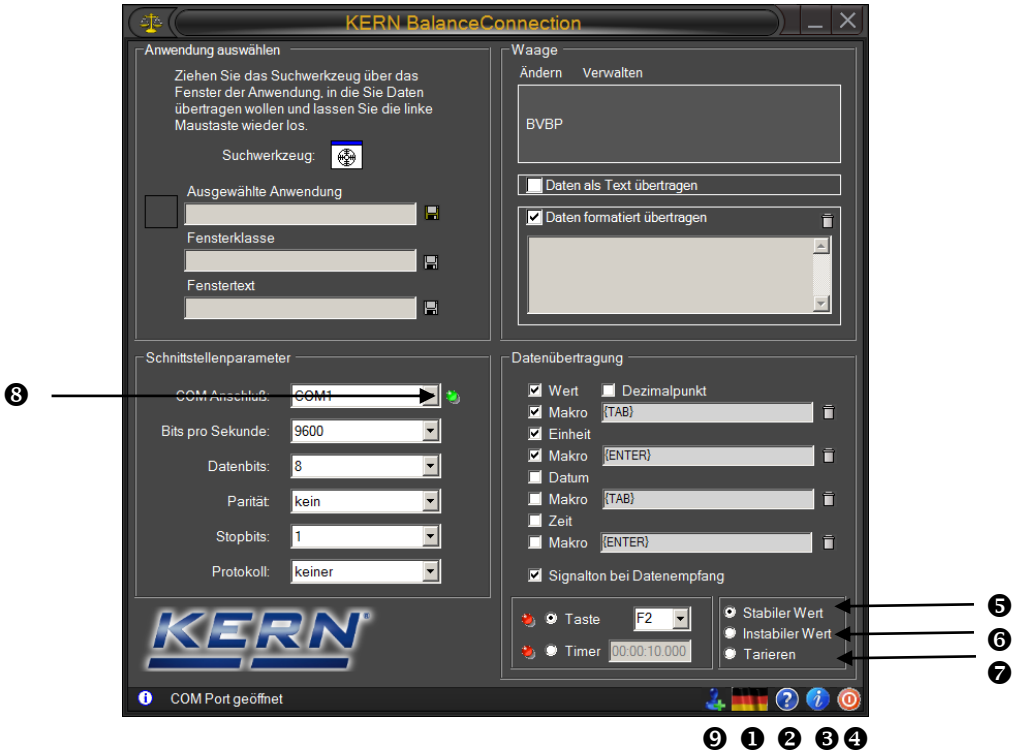
After successful installation the following window appears:

⇨ Installation by clicking on **COMPLETE** end. Balance Connection is started.

4 Program start

⇨ Start this program by clicking the desktop icon or through the Start menu.





❶ Click to change this icon to set the language (German⇆ English).

❷ Click this icon to toggle tooltip help information about the user interface on or off.

❸ Click this icon to display on information about the program.

❹ Click on this icon to quit the program.

❺ Predefined remote control command: It will receive only stable weight values of the scale.

❻ Predefined remote control command: It received unstable and stable weight values of the scale.

❼ Predefined remote control command: Command to tare scale is sent.

❽ A green light when the interface is open, red when the port is closed.

❾ **Enable the expert mode.**

5 Simple mode / Quick Start

The program starts in the "Easy Mode". This mode makes it very easy to pass data from a scale over RS-232 to the computer to another Windows application.

Advanced features are offered by the "Expert Mode". This mode is activated as described on page 9 by button ❾.

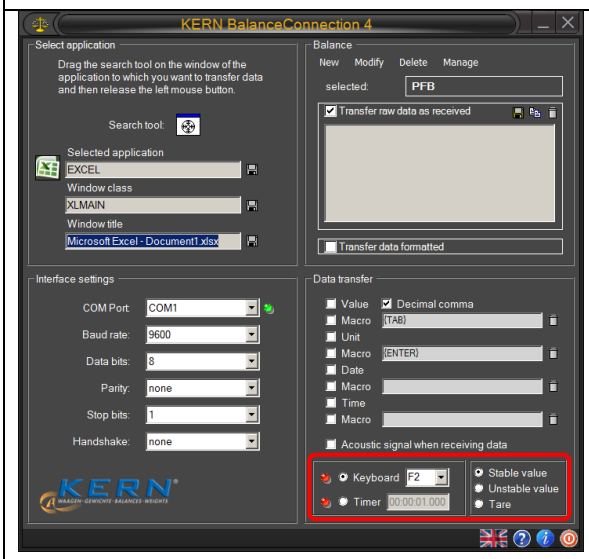
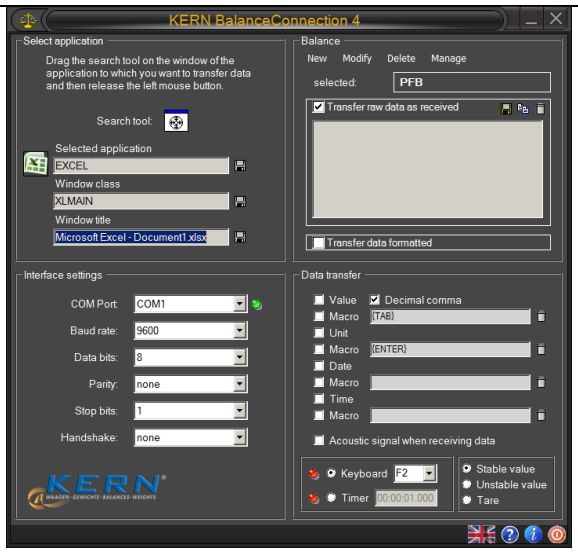
5.1 Data transfer from a KERN balance to Microsoft Excel

The required display is as follows: first column displays the weight value, second column displays the related unit, third column displays the current time, forth column displays the current date.

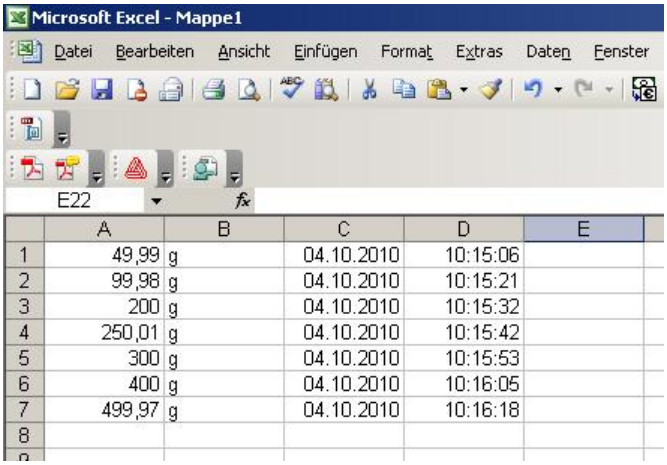
Required settings:

- ⇨ Select using search tool using Microsoft Excel, s. Chap. 5.3
- ⇨ Select Balance type of list, s. Chap. 5.4
- ⇨ "Formatted data transfer" button
- ⇨ Enter COM port.
- ⇨ Click on the desired transmission data, s. Chap, 5.6

For data transmission, there are two options available:

Data transfer via remote control commands	**Data transfer from balance**
	
⇨ Select remote control command (s. Chap. 5 ❺ ❻ ❼). ⇨ Select data transfer button or timer.	⇨ Data is transferred by pressing a button (z. B. PRINT key) at the scales or continuously, depending on the setting in the scale menu.
+ The measurement data are displayed in the right window.	

Corresponding output in EXCEL:



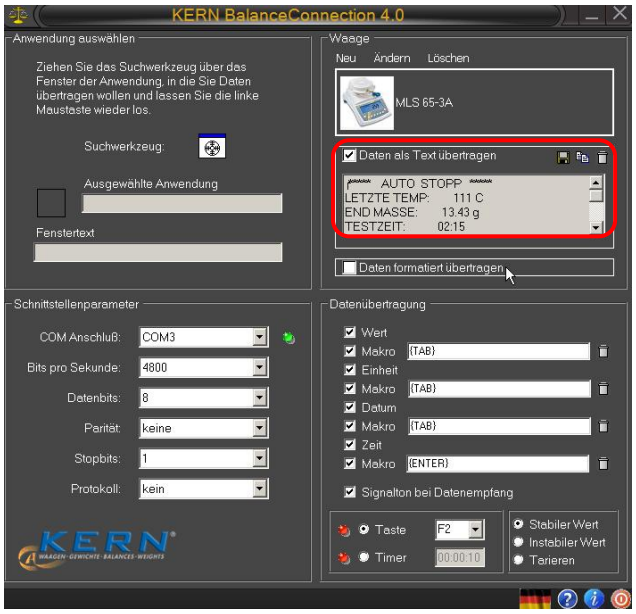
	A	B	C	D	E
1	49,99	g	04.10.2010	10:15:06	
2	99,98	g	04.10.2010	10:15:21	
3	200	g	04.10.2010	10:15:32	
4	250,01	g	04.10.2010	10:15:42	
5	300	g	04.10.2010	10:15:53	
6	400	g	04.10.2010	10:16:05	
7	499,97	g	04.10.2010	10:16:18	
8					

5.2 Take readings in a text file

Example moisture analyzer KERN MLS 65-3A:

Required settings:

- ⇨ Select MLS 65-3A from the list, s. Chap. 5.4
- ⇨ "Transfer data as text" click
- ⇨ Enter COM port.
- ⇨ Start the drying process
- ⇨ After drying press PRINT button on the device, the data is displayed in the right window.



with button save the result as a text file...

or

with button Copy result into clipboard and then in other another application eg Microsoft Word.

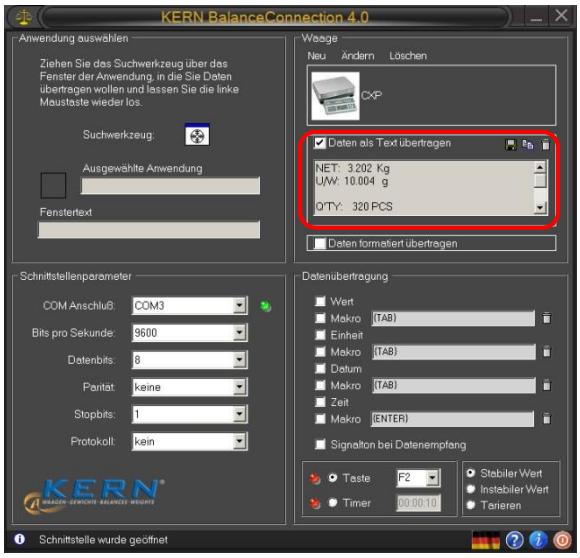
```
***** ***** auto stop
LAST TEMP: 110 C
END MASS: 13.85 g
TEST TIME: 02:15
MASS LOST: 0:19 g
RESULTS: 1.35% M
```

with button clear the data window.

For counting scale:

Required settings:

- ⇨ Select Balance type, for example KERN CXP from list, s. Chap. 5.4
- ⇨ "Transfer data as text" click
- ⇨ Enter COM port.
- ⇨ Determine the reference weight.
- ⇨ Place the parts to be counted and press M + key on the balance. The weight on, reference weight and the number of parts placed on display in the right window.



with button save as a text file..

or

with button copy the results into clipboard and then in other application, like Microsoft Word.

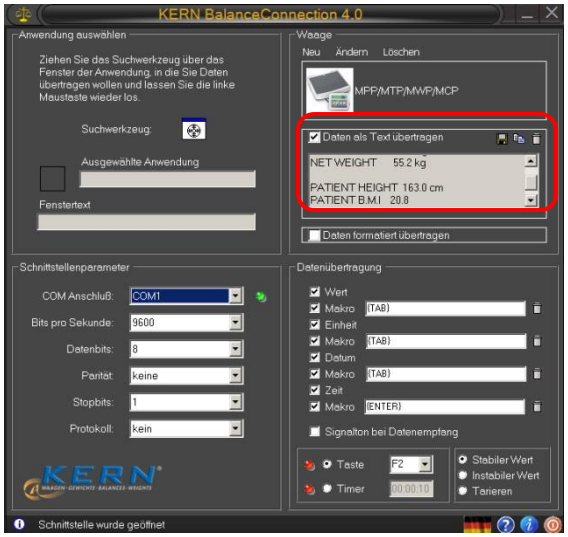
```
NET: 3.202 Kg
U / W: 10.004 g
Q'TY: 320 PCS
```

with button Delete data.

Example bathroom scale:

Required settings:

- ⇨ Select Balance type, for example KERN MPP from list, s. Chap. 5.4
- ⇨ "Transfer data as text" click
- ⇨ Enter COM port.
- ⇨ Determine Body Mass Index, return to the weighing mode and press the PRINT button on the balance. The weight, body length and body mass index calculated from it are displayed in the right window.



with button save result as a text file...

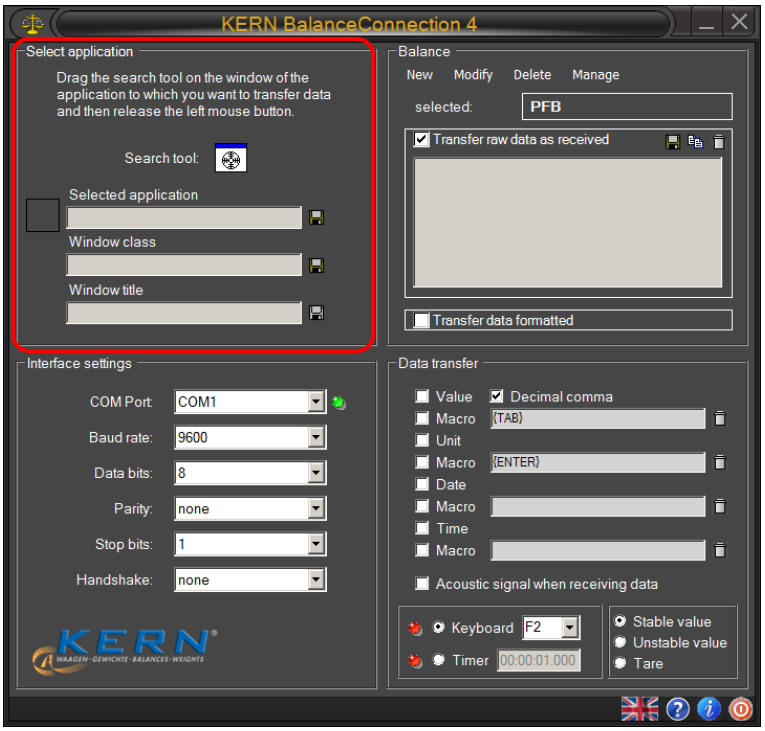
or

with button copy result into clipboard and in other application, like Microsoft Word.

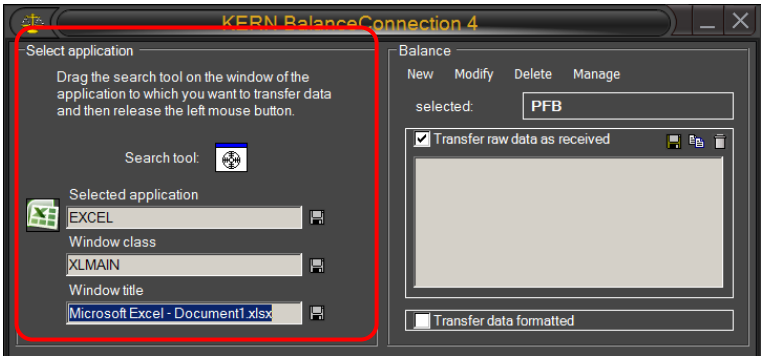
```
55.2 kg gross weight
TARE WEIGHT 0.0 kg
NET WEIGHT 55.2 kg
PATIENT HEIGHT 163.0 cm
PATIENT B.M.I 20.8
```

with button clear the data window.

5.3 Window "Select application"



The software asks you to select the application to which you wish to transmit data. Selecting an application is quite simple. Start your application program, keep it opened in the background and left-click your mouse, keeping the mouse button pressed down, so that you can drag the searching tool with your selected application to the window of your application and then release the mouse button. Afterwards the application selected by you will appear in the box under **SELECTED APPLICATION** (e.g. Microsoft Excel).



5.3.1 Save and automatic detection of the target window

When you close the process or the corresponding window selected in the software, the program loses the connection to the window, until you manually select a new window again.

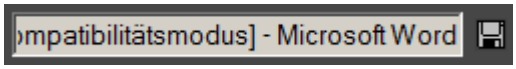
If you want the program to remember the target window by some specific properties and restore the target window upon restart, you have to define those properties. There are three ways how a target window can be identified:

- **Process name (selected application):**
  The name of the process / name of the executable file of the target application. A filter based on this property cannot distinguish between different instances of the same application

- **Window class:**
  The programmer of an application can assign a window “class” to a window. This window class is often more specific than the name of the application.

- **Text window:**
  The text in the title of a window. The title often changes depending on the current state of the target application. Excel, for example, includes the name of the currently opened document in the window title.

The text required to be matched by one of those criteria is defined in the corresponding textbox:



To enable filtering by a property, click on the disk symbol next to the textbox.

An active filter is displayed with a green disk symbol.

After activating a filter, you can adjust the filter text in the textbox.

As an example, the following entry matches all windows, whose title contains the word “Measurement”, regardless of which program owns the window (Microsoft Excel or Word).

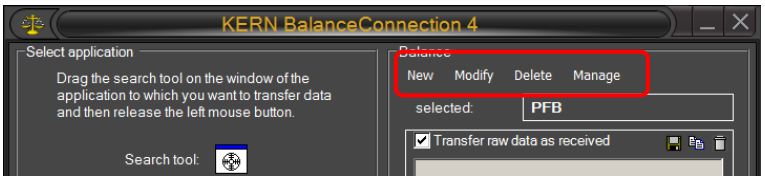
If at least one criterion is active, starts Balance Connection hidden in the taskbar.

5.4 “Balance“ window

To enter the type of balance used by you go to **BALANCE**.

5.4.1 Select predefined type of balance

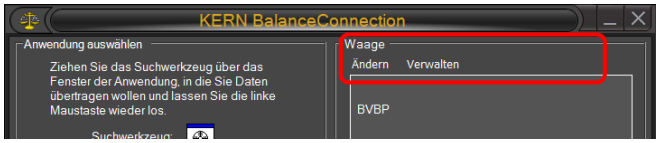
On delivery the software contains predefined type records for KERN balances. Please not that all settings belonging to the respective type of balance will automatically be entered under settings/interface when selecting a predefined type. All that may be left to correct could be the matching COM connection.



- Click on the BUTTON and a selection menu will appear. Scroll up or down, using ▼ or ▲ and select the desired balance.

i	• Balance Connection is only fully operational, when the balance is set to the weighing mode. Applications (e.g. piece counting) are not supported by the Simple Mode in Balance Connection. • Models **KERN ABT, ACS/ACJ, ABS/ABJ, PBS/PBJ** Activate the setting „Handshake off“ in the menu. For further information on how to operate your balance please refer to the operating instructions supplied with each balance.

5.4.2 Edit the available types of weighing equipment



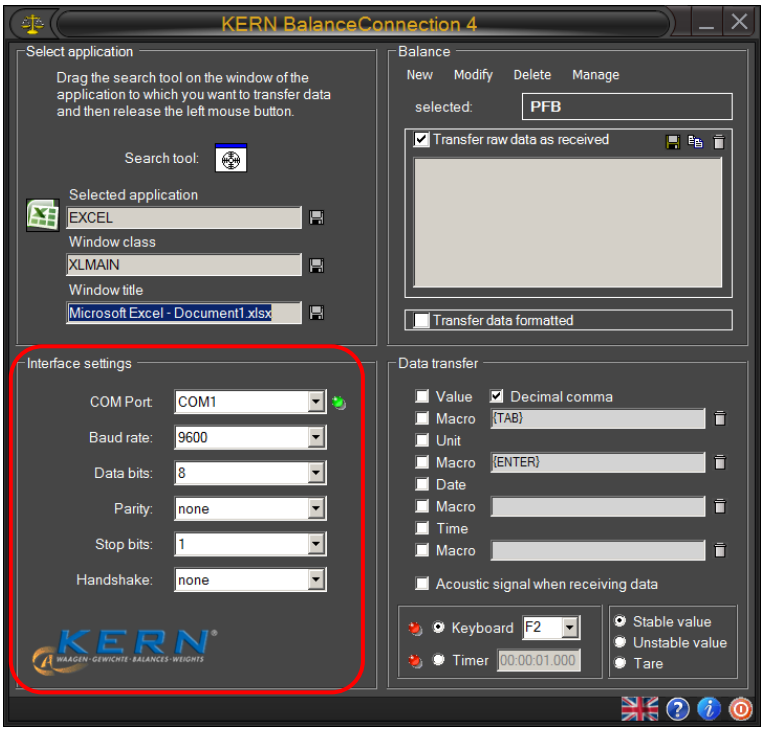
To edit the available device types have the following options:

- **Change:** Opens the properties of the currently selected device.
  See section 8 for details on editing its properties.
- **Manage:** Opens the list of available device types.
  See section 8 for details.

i	Many parameters of a device type are not relevant in Simple Mode. Important features are: • Standard interface parameters • Protocol - interactions for remote control commands (PC as an initiator).

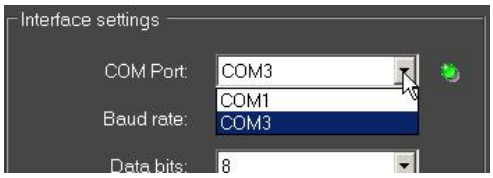
5.5 “Interface settings“ window

This window is used to customise your interface parameters if none of the types of balances on the selection menu for predefined balances matches your requirements.

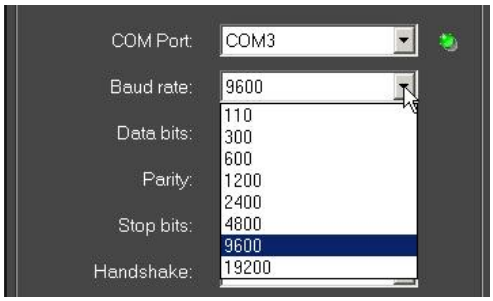


The balance must be connected to the PC via a serial cable. Prior to transferring data you must ensure that the same interface parameters are set on the balance and the PC. Select this program item for this purpose.

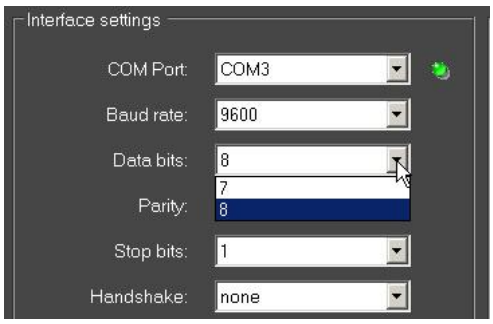
- **COM PORT**: Select the interface at which the connection to the balance is in place.



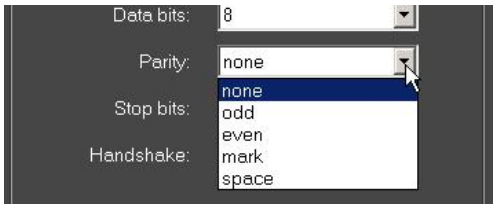
- **BAUD RATE:** Select the speed you require for data transmission (110 to 19200 Baud).



- **DATA BITS:** Select the number of bits used for depicting a character.



- **PARITY:** This selection is the precondition for the PC attributing a parity bit to each character. Possible settings include *space, mark, even, odd* and *none* for parity generation.

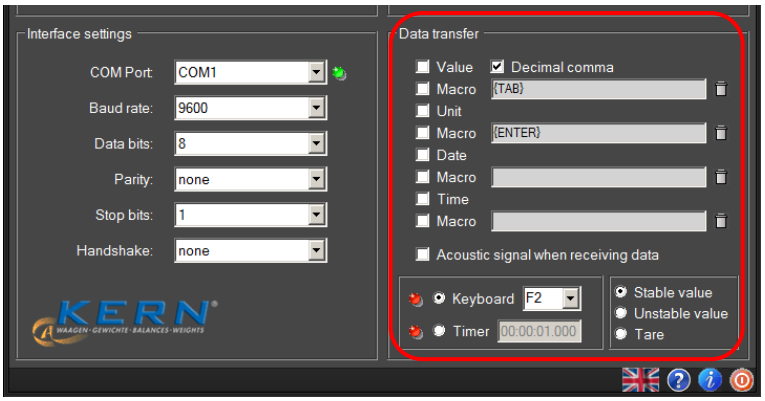


- **STOPBITS:** Select the number of bits to be sent after each character.

- **HANDSHAKE:** Select the control of data flow via software- (Xon/Xoff) or hardware handshake (RTS/CTS).

5.6 “Data transfer“ window

Possible setting options include:



- **VALUE:** Select whether a transmitted value is to be displayed in the application program. It is possible to define a **macro** (This example showing {TAB}, that is, the cursor jumps to the next table segment after each data transmission).

- **UNIT:** Transmitted values are transmitted to the application in the **unit selected for the balance**. It is possible to define a **macro** that is executed immediately after transmission.

- **DATE:** Dependent on necessity, **transmission of date** can be selected either with or without **macro**.

- **TIMR:** Transmission of values additionally with **statement of time**, optionally in 12 or 24 hour mode. It is also possible to define a **macro**.

- **SOUND SIGNAL ON RECEPTION OF DATA:** Determine whether each successful data transmission from the balance to the PC should be followed by an acoustic sound signal. This enables you to monitor acoustically whether data received from the balance has been transmitted to your application.

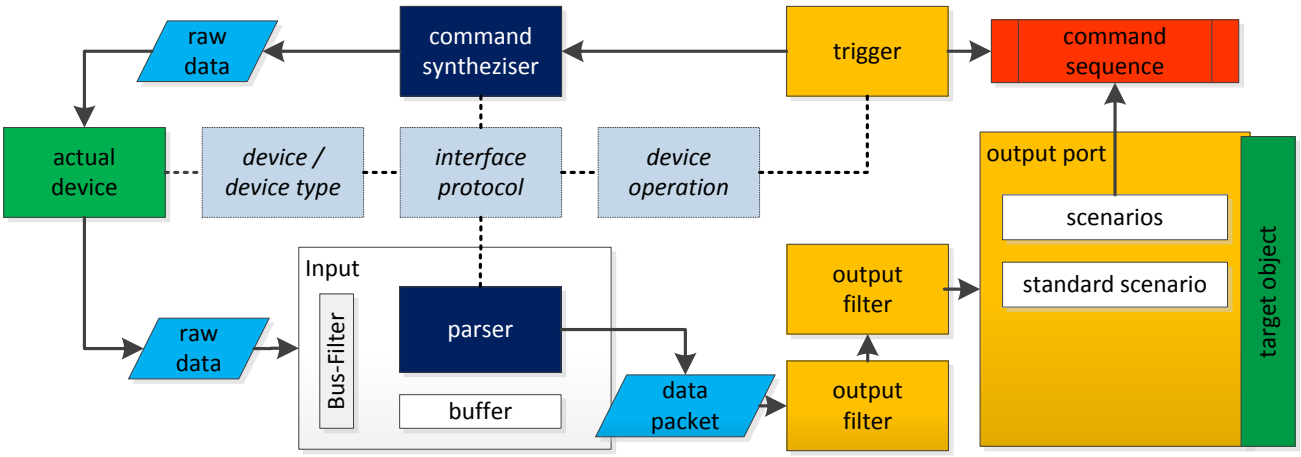
Click KEYBOARD or TIMER and define whether a data transmission of selected remote control commands (see ❺ ❻ ❼) takes place each time a predefined key is pressed or whether this should take place timer-controlled (time interval includes seconds, minutes and hours for exact definition). We have chosen the F2-key for our example.

6 Expert Mode - Overview

Compared to the simple mode, the expert mode provides a rich feature set and allows to very flexible pass on the data from the connected measuring devices to various target applications or other target devices like printers.

Using the flexibility in expert mode requires a good knowledge of the software and its concepts. Below we therefore briefly give an overview of the terms used in the rest of these instructions. These concepts will be discussed in other chapters in detail.

The diagram below gives an overview of the structure and interaction of the elements in the software:



6.1 Device types and device instances

For correct functioning, the software requires information on the connected devices. This information can be divided into specific design characteristics (e.g. interface protocol, available operations, interface parameters, ...) as well as device-specific characteristics (serial number, individual settings).

The design-specific properties are stored in the software with the "device types". A type of device can be used as a "template" for a "device instance". These save additional information such as serial number and also allows to then alter the inherited device type properties. It should be noted that some properties, such as the interface protocol, are shared between all device instances and types.

6.2 (Hardware) interfaces (ports)

A "port" in the software is an abstraction over a hardware port or software communication interface of the computer, to which a measuring device can be connected. So they basically combine RS-232/485 or IP / Ethernet ports in a single concept. The devices raw data are received or sent over these ports.

6.3 Inputs

On certain types of hardware interfaces, multiple devices can be connected. So, for example, RS-458 is a bus system where several scales on the same line are connected to the same PC interface.

In order to address these devices separately in the software, any hardware interface can be assigned one or more "inputs". These inputs accomplish the filtering of incoming data according to the bus ID of the transmitting device.

The more important function of the inputs is the “interpretation” of the transmitted digital information from the devices. This information reaches the interface of the PC in an unstructured format (bytes), it is further referred to as "raw data". To be able to process the information on the PC further, the data thus has to be analyzed with a so called “parser” (it has to be "parsed").

6.4 Protocols / pattern recognition (Parser)

The software supports several types of the aforementioned "parsers":

- **Weight value parser:**

  This simple parser examines the incoming data looking for weight values (that is, a number followed by a unit of weight).

  Various types of weight values (e.g. gross, tare and net) are not distinguished by this parser but treated equally.

- **Text log parser:**

  The text protocol parsers allow the "dismantling" of transmitted text data into its components / parts. These components (e.g. gross / net / tare / enumerator) are then passed separately through the program and can then be output separately.

  To accomplish this, the parser needs information about the interface protocol of the device. An interface protocol consists of "interactions", and further general information on the protocol. An interaction can be initiated by the PC (for example, remote control commands) or it is initiated by the device itself (eg "Print Key").

- **Binary protocols:**

  The current version of the software does not support binary protocols.

6.5 Outputs

A transmission from a device that has been disassembled into its component values is then passed through a "pipeline" to the configured outputs. An output can, for example, be a file, an Excel spreadsheet, another application, a presentation directly on the computer or even another device (e.g. printer).

6.5.1 Output patterns (templates)

To output the data received from the scale flexibly, most output methods offer a possibility to specify an output pattern (template). An output pattern is a string (text) which may also contain variables, shortcuts and functions. On each output action, these parts are replaced with the appropriate value.

6.5.2 Output Filters

In the "pipeline" of data packets, output filters can be interposed between an input and an output. These filters allow you to filter out data packets, alter them (computed fields) or to synchronize packets (record multiple scales at the same time).

6.5.3 Output scenarios

In its basic configuration, upon receipt of information, an output passes data in the defined format on to the destination, regardless of the meaning of the data.
Most output methods also allow for different types of data (adjustment record, stable measured values, ...) to be treated separately by defining different "output scenarios". Thus, for example, the adjustment record is transferred to a different location or in a different way, as a regular reading. Moreover individual command sequences in this context can be triggered (playing sounds, display messages).

6.6 Triggers

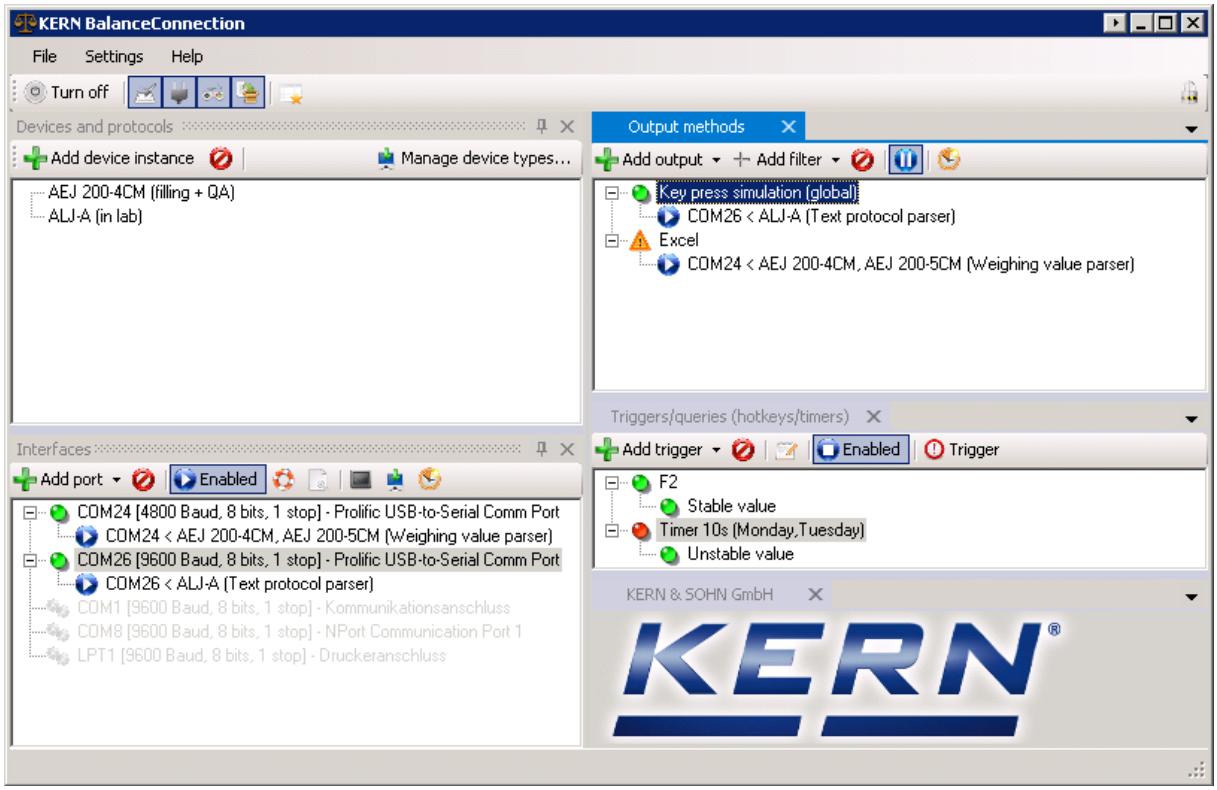
Most measurement devices with communication interface allow for remote control commands to trigger certain functions / operations / responses of the device. A "trigger" specifies when and what device which operation has to be triggered, with what parameters.

A "device operation" can be a query of the measured value, setting an internal parameter or trigger the adjustment, zero or tare of the balance. Depending on operation, several parameters may be necessary. Also whether the device acknowledges or answers depends on the type of operation and device.

6.7 Command sequences

In the context of triggers or output methods, complex command sequences can be triggered instead of the standard function. A command sequence consists of a series of commands with individual parameters. Examples are clicks, keystrokes, activation of windows or, as a special example, highlighting a cell range in Excel.

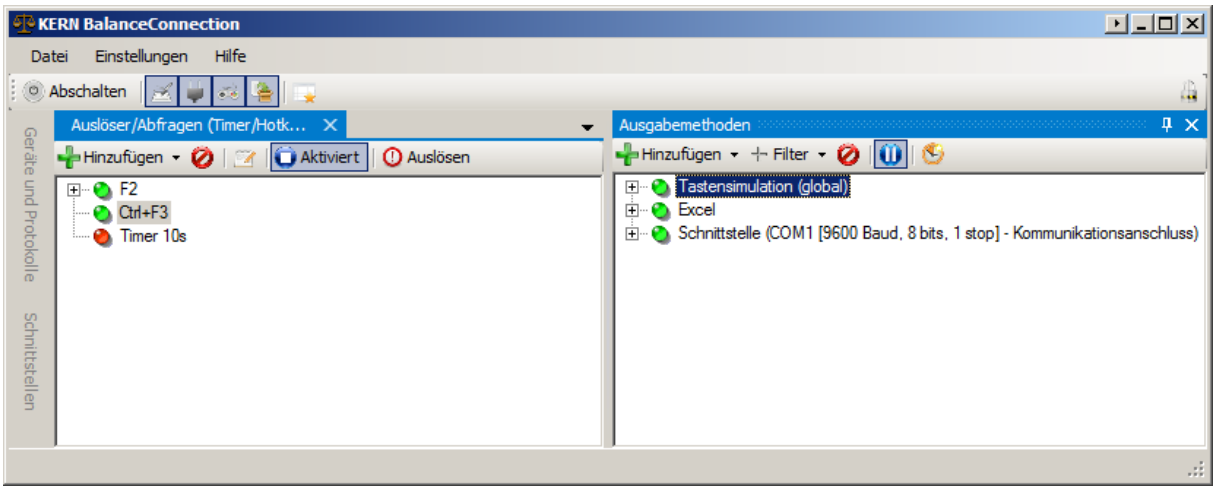
7 General user interface in Expert Mode



7.1 Window / docking

The user interface of the software is modular. Each window or dialog can be moved and "docked" to any location. Many dialogs you can open and continuously adjust settings.

For this, the window's title bar is simply pulled via drag and drop and release at the desired location. The window is docked accordingly.



If the pin needle of a window operated, the window automatically hides after a certain time and is represented as a tab.

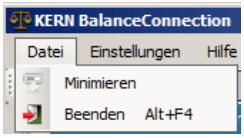
7.2 Main toolbar and menubar

7.2.1 Menubar

The menu bar can be found at the top of the window. It provides the basic settings and functions. The individual submenus are described in the following sections.

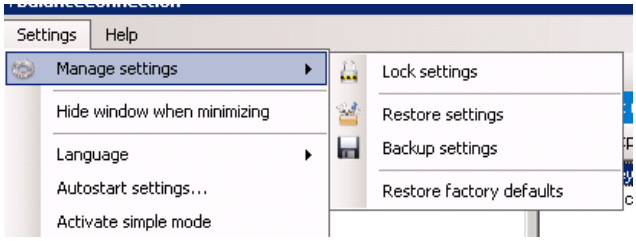
7.2.1.1 File Menu

The File menu includes the following items:



- **Minimize:** Minimizes the window. Whether the window is hidden in the taskbar or there remains visible depends on the minimization setting (see below).
- **Quit:** Stops Balance Connection software.

7.2.1.2 Settings menu



The settings menu contains the basic settings of the program.
It includes the following points:

- **Manage settings:**

  Functions for managing the settings file, in particular backup and restore of the configuration of the program. We recommend you backup your settings after completing configuration. It is however not guaranteed that settings of older software versions can be restored without errors.

- **Hide window when minimizing:**

  If this option is activated, the main window will be completely removed from the system tray when minimized. It can then be restored by clicking the icon in the icon area of the taskbar.

- **Language:**

  The language of the user interface can be changed by clicking on the desired language here. The program is restarted hereby.

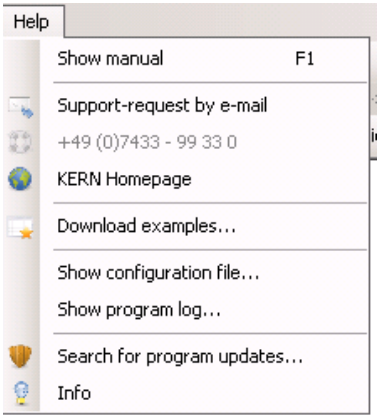
- **Autostart settings:**

  See Chapter 7.3,

- **Enable Simple Mode:**

  Enables Simple-Mode. Achtung: your settings in Expert Mode will be lost.

7.2.1.3 Help menu



If you have any problems or questions when using the software, surely the functions of the Help menu can help:

- **Open operating instructions:**

  Opens the software enclosed version or the online version of the operating instructions.

- **Request support via EMail:**

  Opens an e-mail draft a request to the KERN Balance Connection support with important information about your installation.

- **KERN homepage:**

  Opens the KERN homepage in your default browser.

- **Download sample templates:**

  Opens a website with examples around the use of Balance Connection.

- **See configuration file:**

  Displays the last saved configuration file Balance Connection.

- **Program log:**

  Opens a view of the program logs of Balance Connection. For more information, see the chapter 7.4,

- **Search program update:**

  Checks online if a new version of Balance Connection available. See chapter 16 for more information.

- **Info:**

  Displays basic information about your installation of Balance Connection.

7.2.2 Main Toolbar

The main toolbar can be found under the menu bar:

The toolbar contains the following functions:

- **Pausing transfer:**

  With the "Turn off" button, all triggers, inputs and outputs can be stopped with one click.

- **Show / hide sub-windows:**

  If not needed, the main configuration window can be hidden in the productive operation of these buttons.

- **Open an additional window:**

  The additional window can be used to arrange more docking windows in a separate floating window.

- **Locking of settings:**

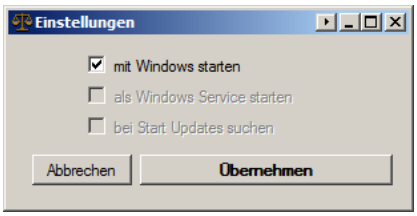
  This button allows you to lock down the settings of the program so as to prevent inadvertent changes to the settings of the program. If blocked, the program asks for a password, which is needed to unlock the settings. Entry of a password is optional.

7.3 Autostart with Windows

In the menu settings you can find the startup options:

- **Start with Windows:**

  If this option is enabled, Balance Connection starts with the machine, under the current user profile.



- **Startup as a Windows service:**

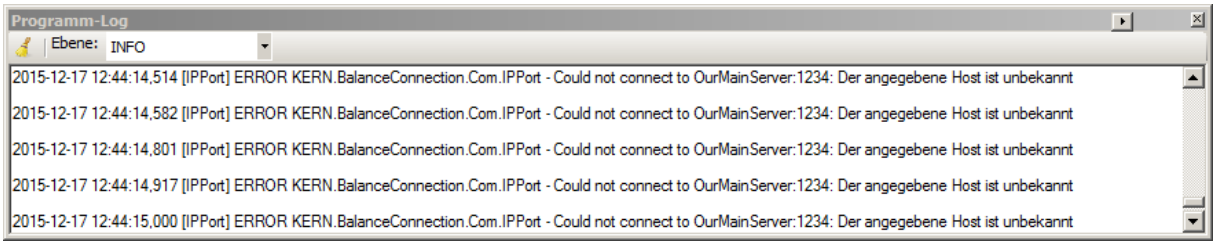
  This option is available in a future release of Balance Connection.

- **Search for updates at start:**

  This option is available in a future release of Balance Connection.

7.4 Program Log

All components of the Balance Connection Software report errors, information and detailed status messages to a central point: the program log. When troubleshooting unexpected behavior of the program, it is usually worthwhile to inspect the program log. You can open the program log by the Help menu.



Use the dropdown "Level" to set the threshold at which messages are displayed. For example, at the default level "INFO" only informational messages, warnings and errors appear, but no debug messages.

8 Device / Device Types

For correct functioning, the software requires information on the connected devices. This information can be divided into two parts: device type dependent (e.g. interface protocol, available operations, interface parameters, ...) as well as individual device-specific characteristics (serial number, individual settings).

The device-type specific properties are stored in the software as "device types".

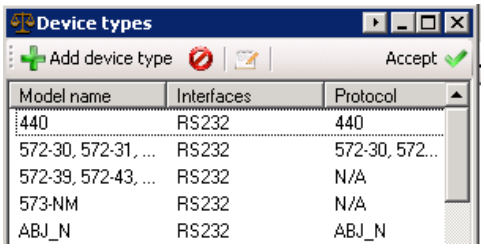
A device type can be used as a "template" for a device instance. This saves additional information such as serial number and also allows to change the inherited device type properties. It should be noted that some properties, such as the interface protocol is shared between all device instances and types.

8.1 Manage device instances

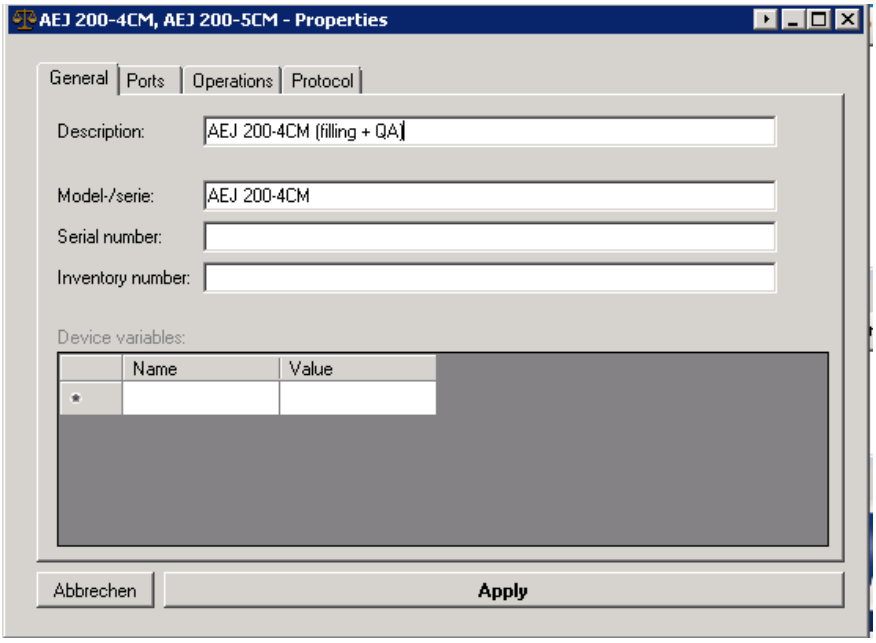
The list "devices and protocols" in the main window of the program lists all device instances the user already added.

8.1.1 Add / Delete device instance

The button "add device" opens the list of available device types. Already predefined KERN device types are listed here grouped by their model series.



After selection of the device type, a copy of the device type is created as a new device instance. It will open the properties dialog for the new device instance:

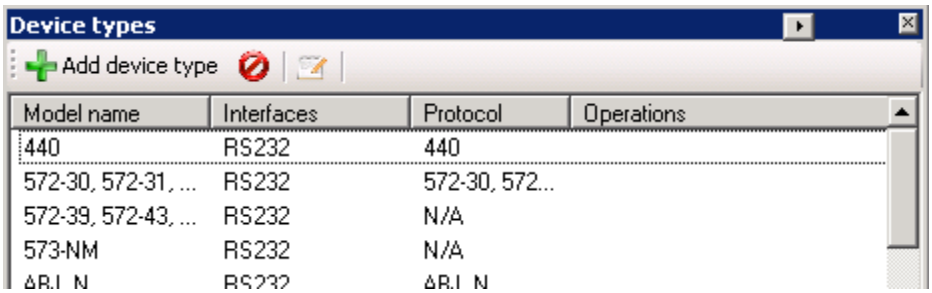


Device instances inherit most of the properties of the device type and these can then be edited independently. The device type properties are explained below. The following features are available only for devices instances:

- **Description:** A user-defined description of the device for ease of recognizing it within the software.
- **Serial number / inventory number (optional):** The serial number or inventory number of the device instance.
- **Other variables:** Other arbitrary variables can be defined here. These variables can be used at certain points in the program. Furthermore, the latest value sent from the device is shown here and is always passed on to all parts of the program respectively.
  Thus, e.g. to query date of last calibration of a balance by a trigger or after the adjustment process is shown here, when the balance has sent a calibration protocol.

8.2 Manage device types

The screen for managing device types can be opened via the button "Manage Device Types" in the main window.



Via the Toolbar you can add other device types, or remove or edit existing ones.

The properties of each device type are:

- **Model / Serial:** Unique identification of the device type.
- **Interfaces:** The available interfaces of the device type as well as their default communication parameters.
- **Operations:** The operations supported by the device. This field is purely informational. See section 0 for more information about device operations.
- **Protocol:** The interface protocol used by the device. More information on interface protocols can be found right below.

8.3 Text protocols

8.3.1 Overview

An important function of an input is the interpretation of the transmitted digital information from the devices. This information reaches the interface of the PC unstructured. To be able to further process the information with the PC, the data has to be “parsed", that is, separated into its components.

The software supports several types of the aforementioned "parsers":

- **Weight value parser:**

  This simple parser examines the incoming data looking for weight values (that is, a number followed by a unit of weight).

  Various types of weight values (e.g. gross, tare and net) are not distinguished by this parser but treated equally.

- **Text log parser:**

  The text protocol parsers allow the "dismantling" of transmitted text data into its components / parts. These components (e.g. gross / net / tare / enumerator) are then passed separately through the program and can then be output separately.

  To accomplish this, the parser needs information about the interface protocol of the device. An interface protocol consists of "interactions", and further general information on the protocol. An interaction can be initiated by the PC (for example, remote control commands) or it is initiated by the device itself (eg "Print Key").

- **Binary protocols:**

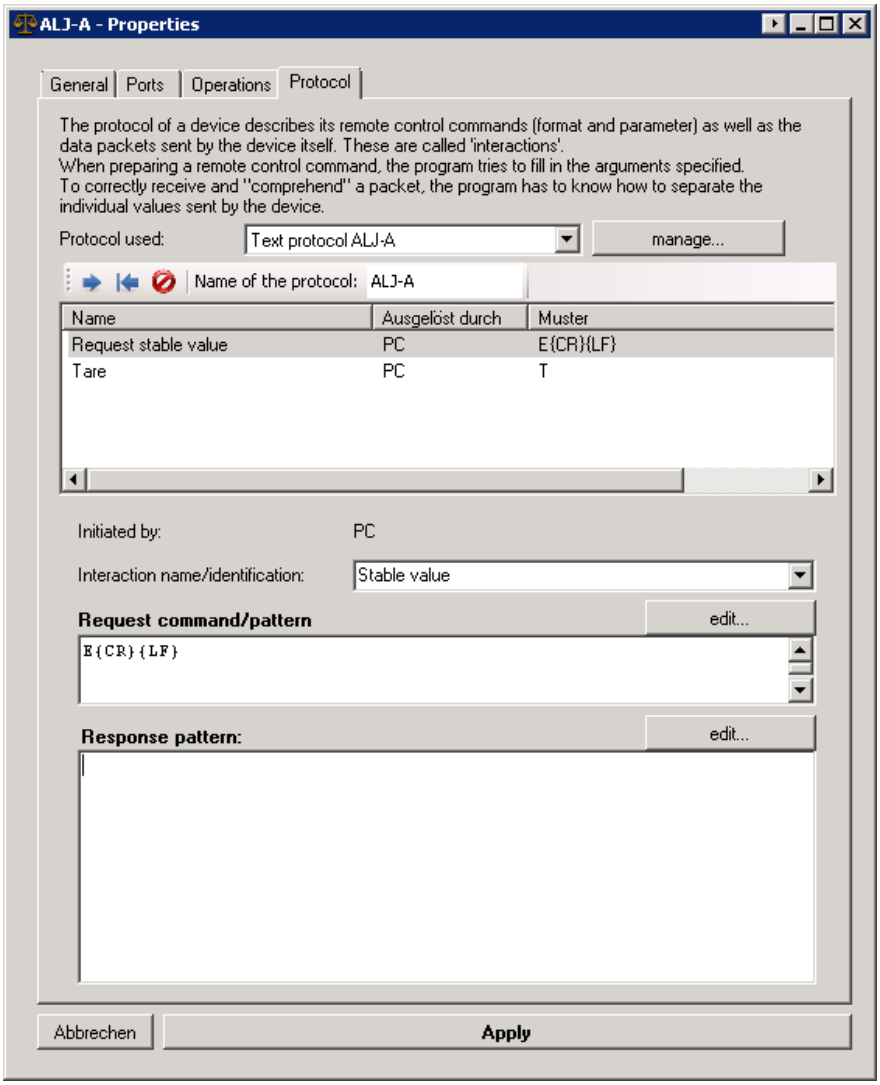
  The current version of the software does not support binary protocols.

8.3.2 Management of interface protocols

i	The management of interface protocols and defining patterns can be a challenge. When missing entries or other problems with interface protocols, please contact KERN.

Interface protocols, as mentioned previously, are shared over all types of devices. A change to an interface protocol has an effect on all devices with this protocol, in particular all the same devices.

The management of the interface protocols can therefore be made in the properties of each device instance or of each device type in the tab "Protocol".



In the "Protocol used" dropdown, the protocol of the current device type is set. The selected protocol itself can be edited directly in this view. Additional protocols can by managed by clicking the "Manage ..." button.

A protocol is uniquely identified by its "name". Assign a unique, easy to understand name.

The "interactions" of the Protocol describe the protocol in detail. They are divided into two general types:

- **Initiated by the device:**

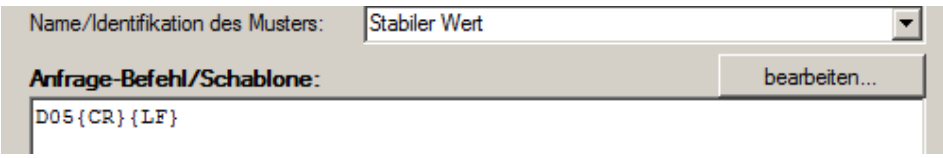
  Examples of interactions which are triggered by the device itself without request by the PC are the Print button of the scale, Auto-Print or continuous transmission. Here the PC "unexpectedly" receives raw data from the device, which interprets, i.e. must be separated into its components. For this, it always uses the "response template" for parsing. As soon as a defined interaction "fit" to the received raw data, this raw data is further processed and then deleted from the input buffer.

- **Initiated by the computer:**

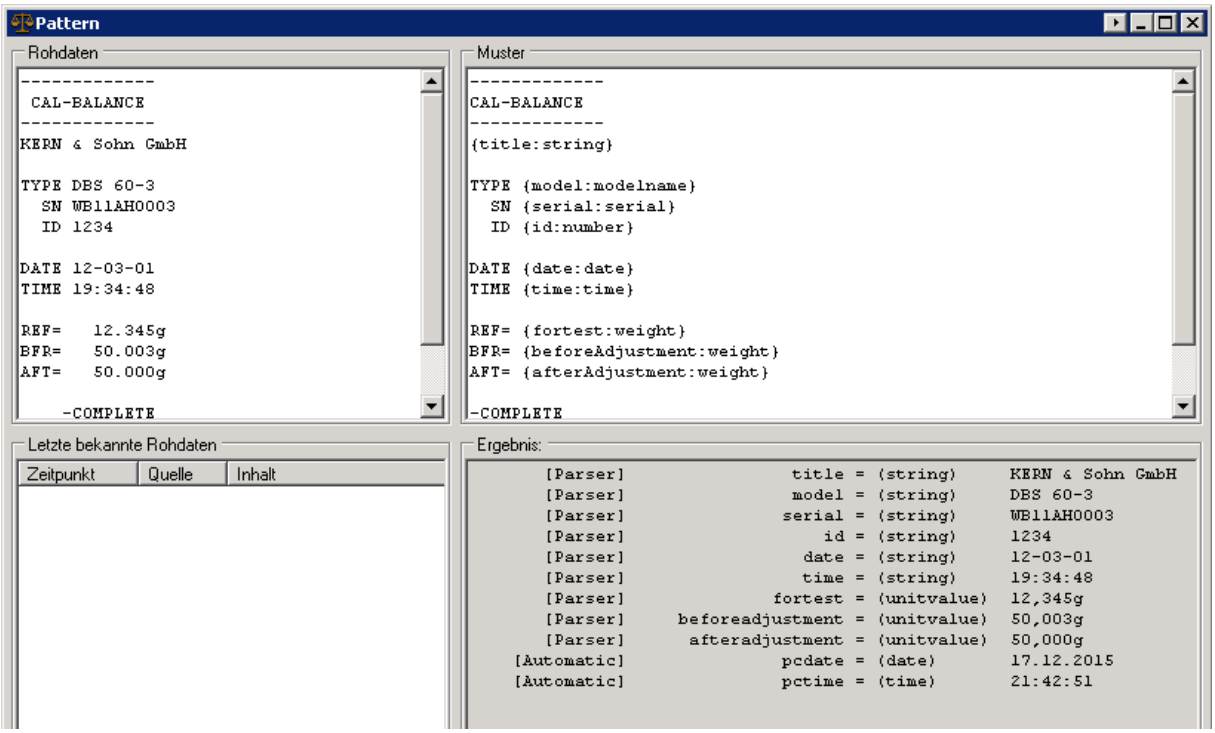
  When the PC wants to trigger an operation of the device, it needs to sent a correctly formulated remote control command. As a template for such a remote command the software will use the "request command / mask" pattern of the interaction that has the same key (internal name) as the desired operation. Within this pattern parameters defined by the trigger are filled in according to the general specification of the text protocol (e.g. escape characters).

8.3.2.1 Definition of parser patterns

For the specification of interactions of an interface protocol, the correct formulation of recognition patterns is essential. In order to simplify building a pattern, the program provides a preview mode for both request- and response pattern. Click the "Edit ..." button to open the Preview dialog.



This opens the following dialog:



The Preview window consists of four parts:

- **Raw data:**

  Copy here the raw data sent by the device, at least the parts that relevant for this interaction, or select from the known raw data. Also you can, for example, extract raw data from the terminal window of an input port (see 9.2.3.1) or from the devic-

es instruction manual. The pattern is applied to this raw data.

- **Last known raw data:**

  Most recently received raw data from the instrument is stored by the software. Instead of entering the raw data, you can choose from the listed data here. Multiple items can be selected at once, those are combined into a single string.

- **Pattern:**

  Code for the recognition pattern.
  Guidelines on recognition patterns are described below.

- **Result:**

  Could the pattern on the raw data be successfully applied, the "Result" window shows all extracted variables from the raw data in the following format:

  ```
  [Source] VariableName = (DataType) Value
  ```

8.3.2.1.1 Structure of parser patterns

The code in parser patterns is made to be as simple as possible. They are compared directly to the raw data of the code. Additional, they have the following rules:

- Multiple spaces in raw data or patterns are not considered.
- Single line breaks are considered.
- Several successive line breaks are ignored.
- For extracting variables are written in {}, according to the following pattern (without spaces):
  `{VariableName : DataType}`

8.3.2.1.2 Data types of variables in patterns

The software so far supports the following data types for variables inside patterns. Values of a data type can provide further sub-properties, which can be used in output patterns (see chapter 10.3). For example, a value of type "weight" has sub-properties like sign (".sign"), unit (".UNIT") or numeric value (".value").

Name	Description
`string`	String within a line
.len .12	Length of string Character at 12th position in the string
`string-nospaces`	String without space
`modelname`	String with usual characters of item numbers / model names
`serial`	String with usual characters of serial numbers
`weight`	Weight value (consisting of numerical value and unit) .Units: kg, g, gm, ct, mg
.value .unit .valueabs .sign	only numerical value only unit absolute numerical value sign
`number`	Integer or floating point number (comma or period)
.abs .sign	Absolute numerical value sign
`date`	Date (only numbers, separated by a hyphen or period)
.day / .d .month / .m .year / .y .dow	day month year Weekday (Day of Week)
`time`	Time (hh: mm: ss or hh: mm)
.hour / .h .minute /.m .second / .s .ms	hours minutes seconds milliseconds

8.3.2.1.3 Example of a complex pattern

The following example using the adjustment log of a moisture analyzer DBS 60-3 shows how the data transmitted from the device can be split. The variables shown under result can be processed further on the program and then be handed over to a target application.

Raw data	Parser pattern
`----------`	`----------`
`CAL-BALANCE`	`CAL-BALANCE`
`----------`	`----------`
`KERN & Sohn GmbH`	`{title:string}`
`TYPE DBS 60-3`	`TYPE {model:modelname}`
`  SN WB11AH0003`	`  SN {serial:serial}`
`  ID 1234`	`  ID {id:number}`
`DATE 12-03-01`	`DATE {date:date}`
`TIME 19:34:48`	`TIME {time:time}`
`REF=  12.345g`	`REF= {referenceWeight:weight}`
`BFR=  50.003g`	`BFR= {beforeAdjustment:weight}`
`AFT=  50.000g`	`AFT= {afterAdjustment:weight}`
`  -COMPLETE`	`-COMPLETE`
`  -SIGNATURE-`	`-SIGNATURE-`

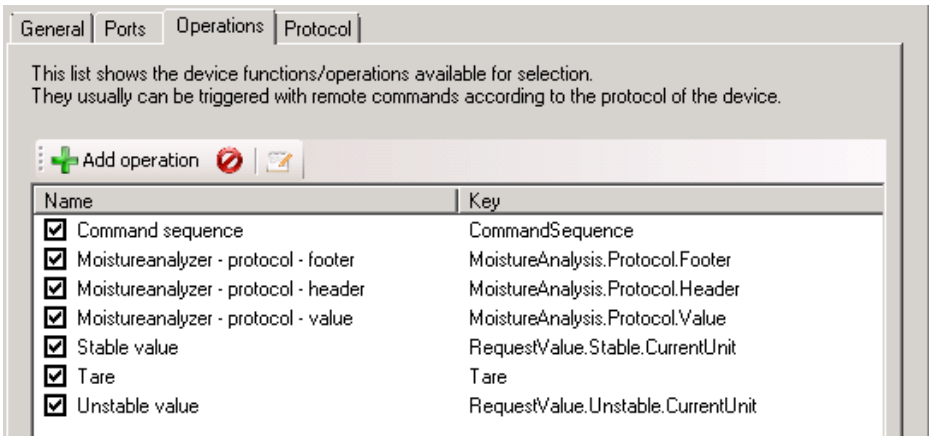
Ergebnis:

```
       [Parser]                title = (string)     KERN & Sohn GmbH
       [Parser]                model = (string)     DBS 60-3
       [Parser]               serial = (string)     WB11AH0003
       [Parser]                   id = (string)     1234
       [Parser]                 date = (string)     12-03-01
       [Parser]                 time = (string)     19:34:48
       [Parser]      referenceweight = (unitvalue)  12,345g
       [Parser]     beforeadjustment = (unitvalue)  50,003g
       [Parser]      afteradjustment = (unitvalue)  50,000g
    [Automatic]               pcdate = (date)       17.12.2015
    [Automatic]               pctime = (time)       22:40:56
```

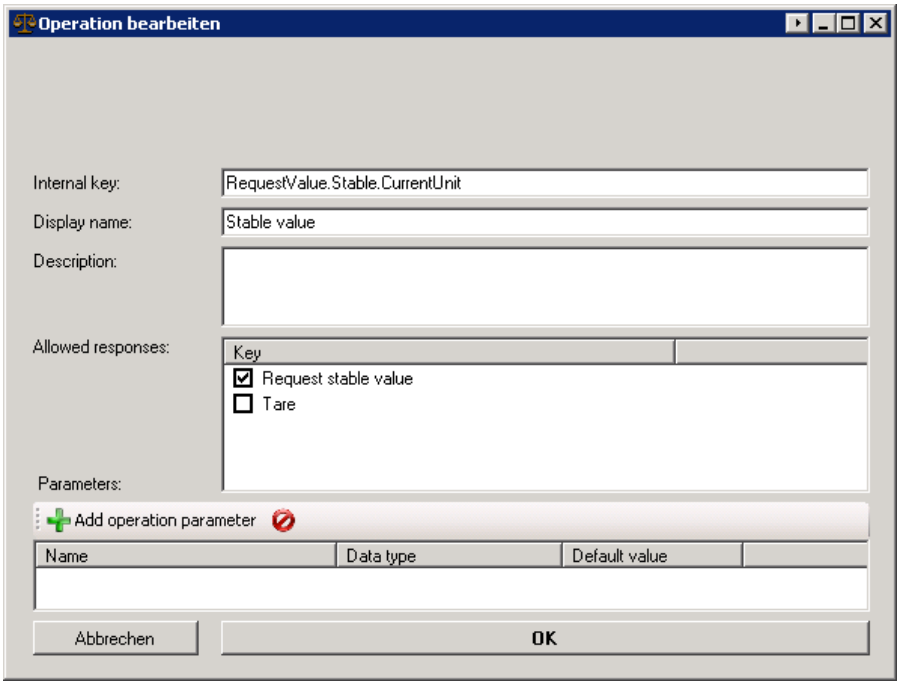
8.4 Device operations

A "Device Operation" can be a query of the measured value, setting a parameter or triggering adjustment of the balance, zeroing or taring. Depending on operation and device type, parameters may be necessary and the device may send a confirmation.

You can edit the available operations for each device type. Newly created operations are available for all device types.



By double-clicking you can edit the parameters of an operation:



9 Interfaces (ports)

9.1 Overview

A "port" in the software is an abstraction over a hardware port or software communication interface of the computer, to which a measuring device can be connected. So they basically combine RS-232/485 or IP / Ethernet ports in a single concept. The devices raw data are received or sent over these ports.

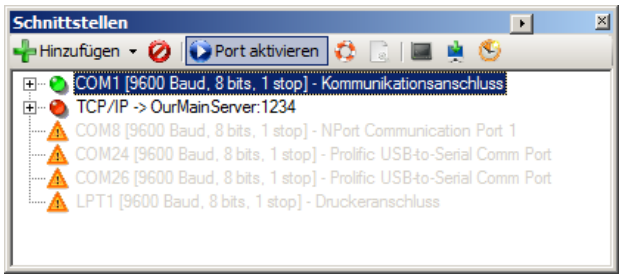
On certain types of hardware interfaces, multiple devices can be connected. So, for example, RS-458 is a bus system where several scales on the same line are connected to the same PC interface.

In order to address these devices separately in the software, any hardware interface can be assigned one or more "inputs". These inputs accomplish the filtering of incoming data according to the bus ID of the transmitting device.

The more important function of the inputs is the “interpretation” of the transmitted digital information from the devices. This information reaches the interface of the PC in an unstructured format (bytes), it is further referred to as "raw data". To be able to process the information on the PC further, the data thus has to be analyzed with a so called “parser” (it has to be "parsed").

9.2 General configuration

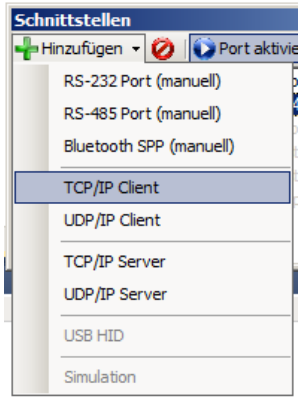
The "Interfaces / Ports" window lists all the hard- or software interfaces in the system that are recognized and/or configured by the user. In the sub-tree of every configured ports the "inputs" connected to it are listed.



With the functions of the associated toolbar or the context menu, the parameters of the ports can be altered.

9.2.1 Add a port

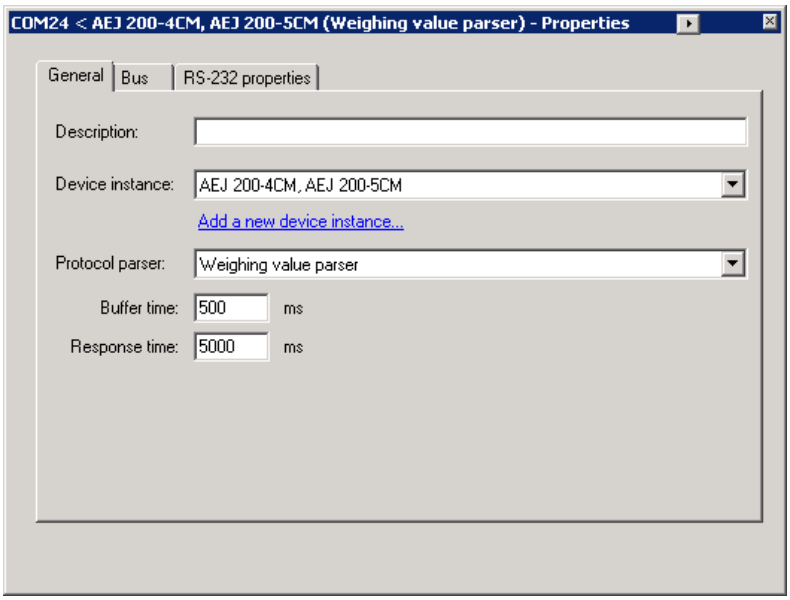
To add ports, there are two possibilities:



- Ports that were detected in the system are listed in the overview list and can be easily added by "activating" them.
- Non-configured or detected ports can be created via the "Add" button. The desired type can be selected from the drop-down list.

In both cases, it opens the properties dialog as described below.

9.2.2 Properties of an input and port



In the properties dialog of an input, both its properties and the properties of the connected hardware port can be configured.

For all inputs following tabs are available:

9.2.2.1 General

This tab displays the general properties of an input. These include:

- **Description:** A description entered here is displayed instead of the automatically generated description.

- **Device instance:** The device instance connected to this input. When first assigning a device to an input, the default settings can be applied to the input and the underlying hardware port.

- **Protocol parser:** The parser used for this input. Depending on the type of the parser, the protocol assigned to the selected device is used for the analysis of incoming data and for the generation of remote control commands.

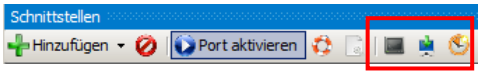
- **Buffer time:** Data sent from the device will be stored in the input buffer for some buffer time prior to processing them with the parser. This is to prevent the dissemination of incomplete information.

- **Response time:** The maximum time until a response of the device should arrive for on a remote control command. After this time, the restriction of the data processed by the input data is lifted.

9.2.2.2 Bus (bus filter / bus systematics)

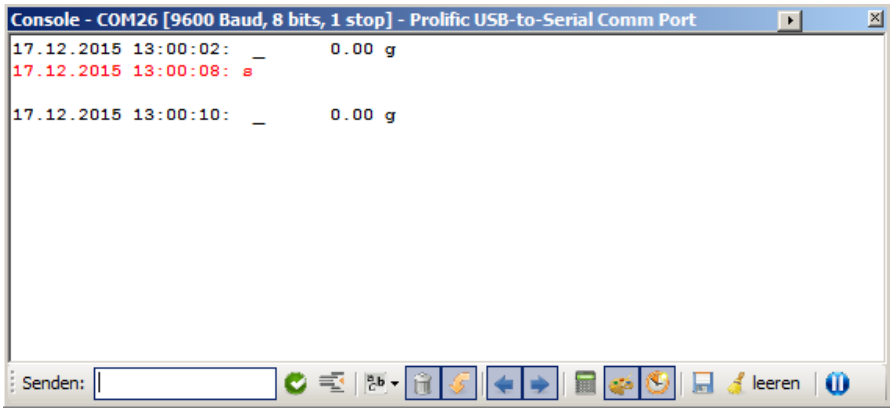
In the current version, scales connected over a bus system can be distinguished only by different protocols and output scenarios.

9.2.3 Tools



You can open the following tool windows from the toolbar.

9.2.3.1 Terminal

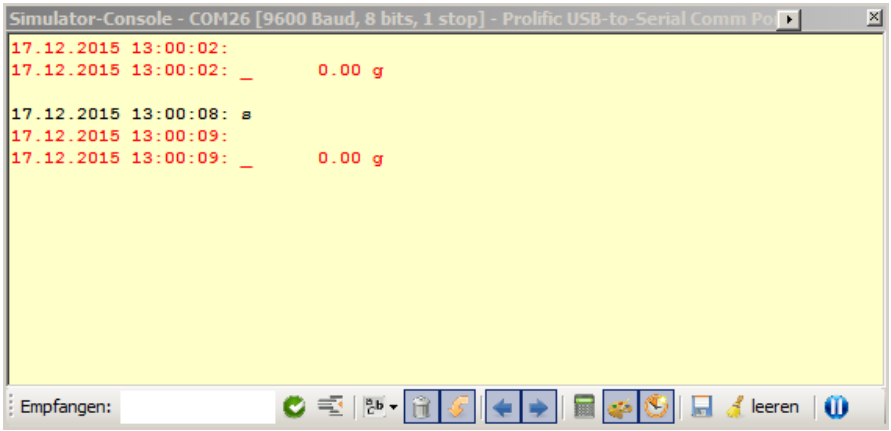


The terminal displays the data received from the connected device, and the data transmitted to the device by the software as raw data. In the toolbar you can choose between a variety of display options.

Furthermore, any data can be sent to the connected device. This is especially useful for testing the response to remote control commands.

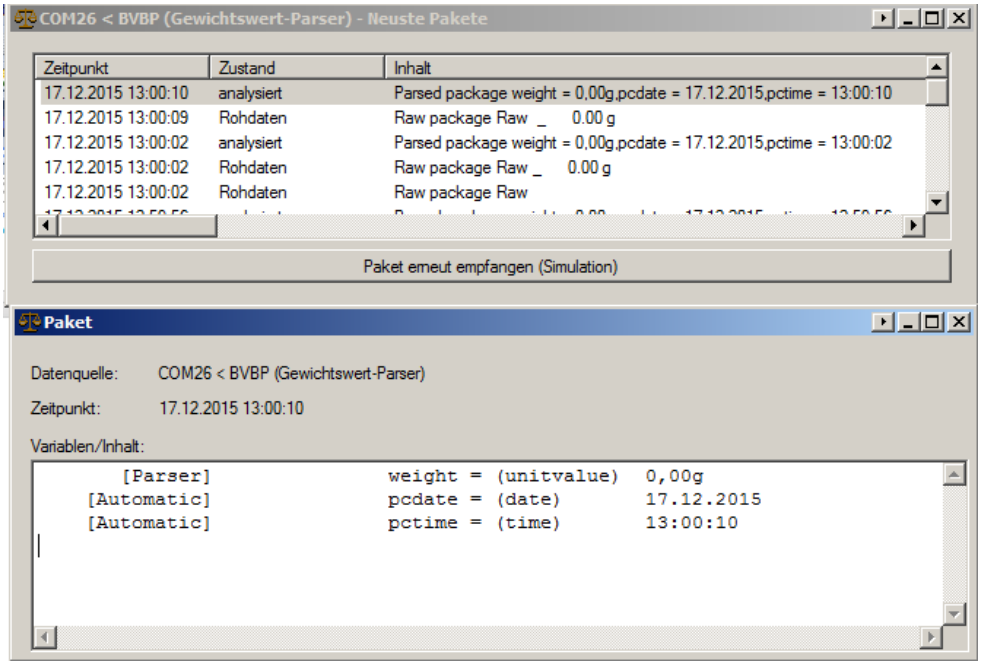
9.2.3.2 Simulation

The "simulation terminal" is analogous to the previously described terminal, but represents the point of view of the connected device. The "send" function is therefore replaced by a "receive" function, which allows simulate raw data from the device to the PC without the device having to actually send any data.



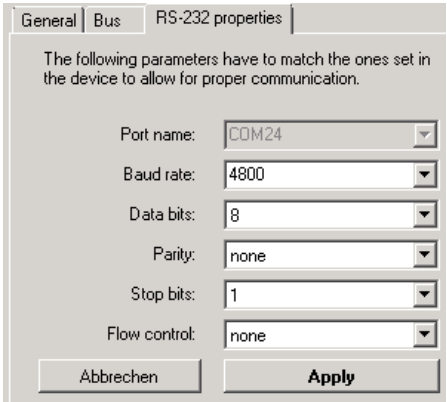
9.2.3.3 Latest received packets and raw data

All raw data received from an input and passed- analyzed by the parser assigned data packets are listed here. Double-click on a package at its detailed presentation (if available) opens.



A data packet can be “received again”. It will then be passed through the pipeline down to the outputs again.

9.3 RS-232 ports



In addition to the characteristics described above some additional settings have to be configured for an RS-232 port.

The interface parameters must match exactly with the parameters set on the device. If this is not the case, the incoming data is not readable ("hiero-glyphs"). Also remote control commands will not be transmitted correctly, and the device does not re-spond accordingly.

The interface parameters will be changed automatically when assigning the device instance to the input and should therefore already be consistent to the factory set-tings.

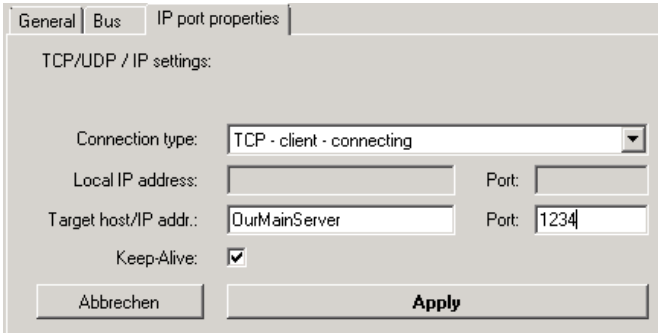
9.4 RS-485 ports

A RS-485 interface is configured similarly to a RS-232 interface. In addition, a bus-filter (see above) can be configured.

9.5 IP network (TCP / IP, UDP / IP, Ethernet)

For communication with a device that is connected via an Ethernet interface to the (company) network, the software supports TCP / IP and UDP / IP ports (sockets). For both types, it is possible to either connect to the balance (client mode) or to wait for the device connecting to the computer (server mode).

Mode and destination of the connection can be set at the "IP Port Properties" of the input.



An IP port by default is configured so that the connection to the remote site is always maintained (keep-alive). The status icon in the list of ports shows the current connection status. The program log lists information about the connection attempts by the program when required.

In the following sections, the configuration of each mode will be described in more detail.

9.5.1 TCP or UDP / IP client (connecting)

In client mode, the computer running the Balance Connection software connects to the balance. The balance has to be configured so that it provides a TCP or UDP server on a particular port.

To add a client port in the Balance Connection software, select the required option from the list of available interfaces.

The port is added and its settings are opened. For a client port, choose the option "TCP / UDP - Client - connecting". Also, specify the connection target (target host name or IP address) and the destination port.

9.5.2 Server (listening / waiting)

Creating a server port is similar to creating a client port. Select TCP or UDP server mode in the mode drop-down and enter the port number on which the program should listen for incoming connections.

The port entered must not be already used and it should be greater than 1024 in order to avoid permission problems. In case the status of the port does not change to OK, please check the program log.

9.6 Bluetooth

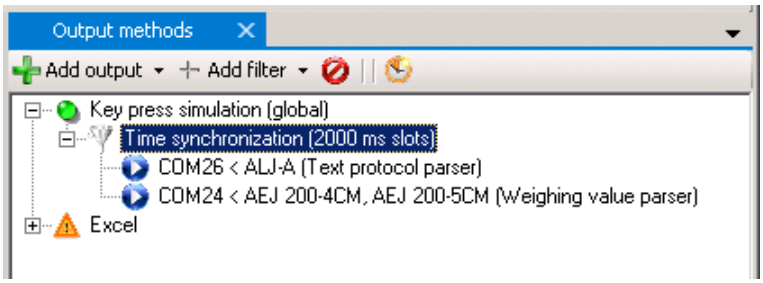
To connect to a Bluetooth device, it has to be already coupled to the computer and have a virtual COM port assigned. For this, the device must support the Bluetooth Serial Port Profile (SPP). The virtual COM port will automatically appear in the list of available interfaces and is configured similar to an RS-232 port.

10 Outputs / output methods - General

A transmission from a device that has been disassembled into its component values is then passed through a "pipeline" to the configured outputs. An output can, for example, be a file, an Excel spreadsheet, another application, a presentation directly on the computer or even another device (e.g. printer).

10.1 Overview

The "Output Methods" window in the main window shows all currently configured outputs and output filters.

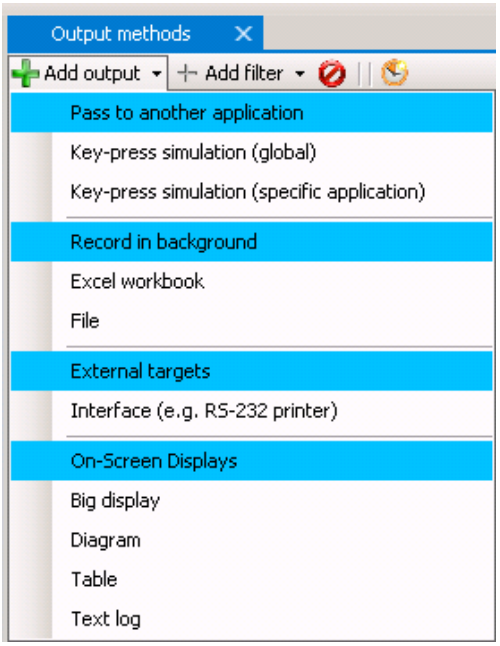


At the top level, the tree displays all outputs and filters that are yet assigned to any output. Associated data sources (i.e. inputs or upstream filters) are displayed as sub-nodes.

To manage the entries use the toolbar or the context menu (right mouse button):

- **Add new outputs:**

  To do so press the "Add" button. This will open a dropdown menu with a selection of available output methods, grouped by category of the method. Simply choose the desired type. A new instance of the type will be added. For details on how to edit its properties see below.



- **Adding new output filters:**

  For details on output filters see chapter 12.

- **Editing spending and filters:**

  Double-click entries to edit the properties of the filter or output.

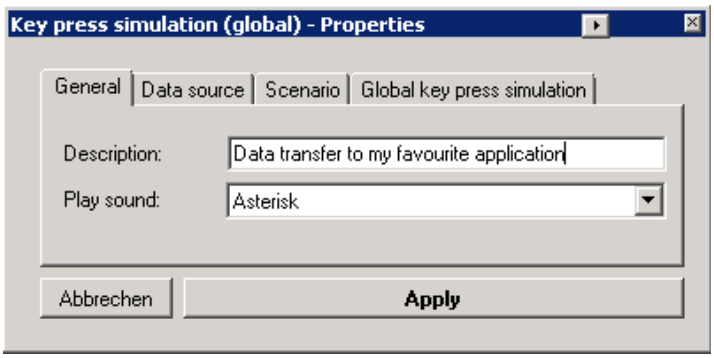
- **Deleting outputs and filters:**

  To delete an output / filter, select it and press the "Delete" button.

10.2 Configuration of outputs

The configuration dialog for output opens when you create or double-click on an output. Each output has the following properties:

10.2.1 General properties



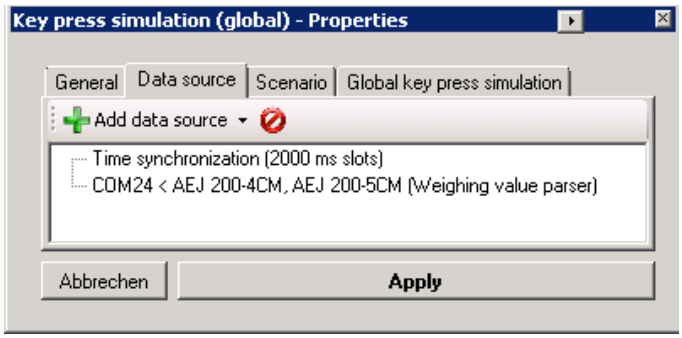
- **Description (optional):**

  A description you enter here will be displayed instead of the automatically generated description for the output.

- **Play sound (optional):**

  Plays a Windows-sound or a custom sound file whenever an output receives a data packet.

10.2.2 Data Sources



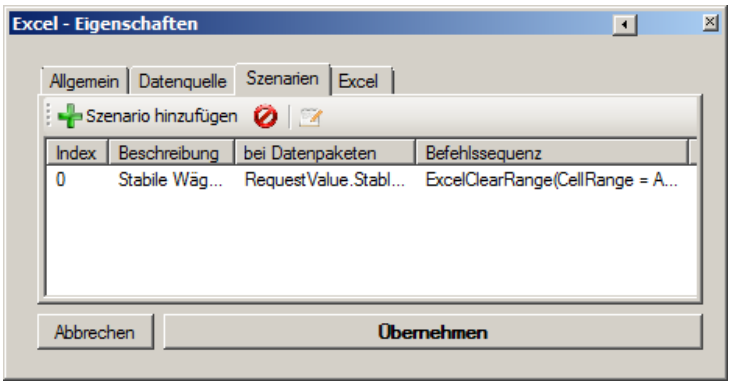
In the "Data Source" tab, you can manage the inputs or upstream filters will be used as sources of data packets for the output. Data packets of these inputs are thus forwarded to the output and processed there.

10.2.3 Output Scenarios

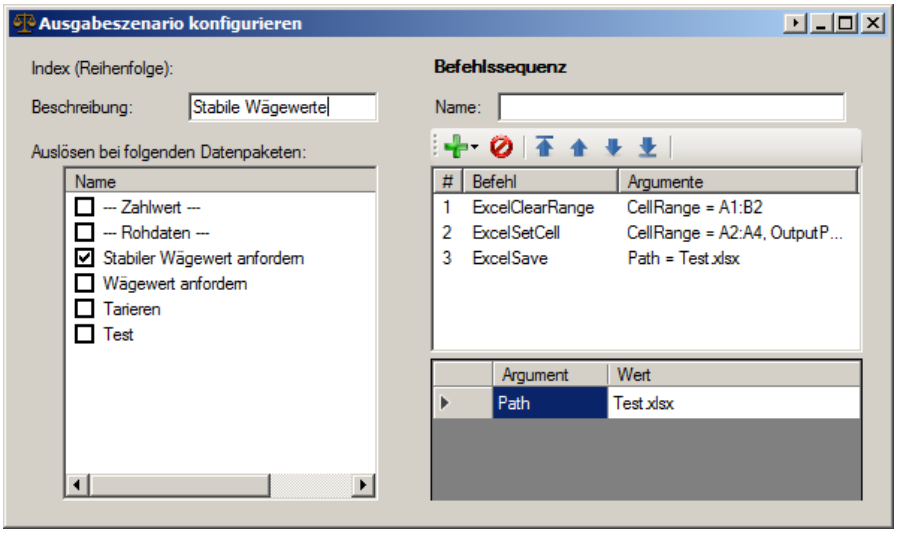
In its basic configuration, an output passes data on whenever it receives a data packet, regardless of the type or meaning of the data. Most output methods allow to specify different “scenarios” for different types of data (e.g. calibration protocol, stable measured value) and react accordingly.For example, a calibration protocol can be transferred to a different location or in a different way than a regular reading. This is

achieved by individual command sequences that are run in the context of this output scenario.

For the configuration of output scenarios, see the tab "scenarios".



Using the toolbar you can add, delete or edit scenarios.
The configuration dialog of an output scenario:



An output scenario consists of the following components:

- **Description:** Short description to distinguish between scenarios.

- **Accepted data packets:**

  When there is one or more types of packages selected here, the set command sequence will be executed only upon receipt of such a data packet.

- **Command sequence:**

  Whenever the output receives a data packet of one of the selected types the command sequence runs in the context of this output and data packet. For more information about command sequences see chapter 14.

i	If you want to trigger the default behavior of an output in an output scenario, please use the command "StandardOutput()".

10.3 Text Formatting (output patterns / templates)

To output the data received from the scale flexibly, most output methods offer a possibility to specify an output pattern (template). An output pattern is a string (text) which may also contain variables, shortcuts and functions. On each output action, these parts are replaced with the appropriate value.

10.3.1 Rules on output patterns

An output pattern is a simple text with the following extensions:

- **Key press: {F3}**

  In an output pattern key presses can be integrated in braces (e.g. {F3}). For outputs not compatible with keys, the program tries to convert the key press into a text element. For special keyboard shortcuts, the configuration dialogs described below offer a designated input box. See the Microsoft .NET Framework SendKeys() function for more details on the key specification.

- **Direct variables: <<<VariableName>>>**

  The program checks for a variable with the given name. Is there no variable with that name, the program is looking for a variable with a sub-property of that name.

- **Sub-properties of variables: <<<VariableName.SubProperty>>>**

  When a variable name is followed by a period, the specified sub-property of the value of the variable is output instead. The available sub-features are in section described 0.

- **Functions: <<<FunctionName(Argument1,Argument2, ...)>>>**

  Functions allow for complex transformations of output values. This ranges from simple functions such as capitalization, extracting parts of strings and even to rendering image code for label printers. The available functions are listed in the configuration dialog for complex patterns. When selecting a function in the configuration dialog, the expected parameters will be prefilled with example values.

An output pattern can be specified in many places of the program. For simple configuration, there are two input dialogs for output patterns, explained in detail below.

10.3.2 First example of an output pattern

In the following example, the output pattern is applied to a simple data packet containing a variable "weight" of type weight (sub-properties “sign”, “value” and “unit”):

```
<<<pcdate.y>>>  <<<weight.sign>>>   KERN  {F3}
<<<weight.value>>> {LEFT} <<<uppercase(unit)>>>{ENTER}
```

The result is (depending on the output method):

```
2015  +   KERN
 0.00  G
```

Explanation:

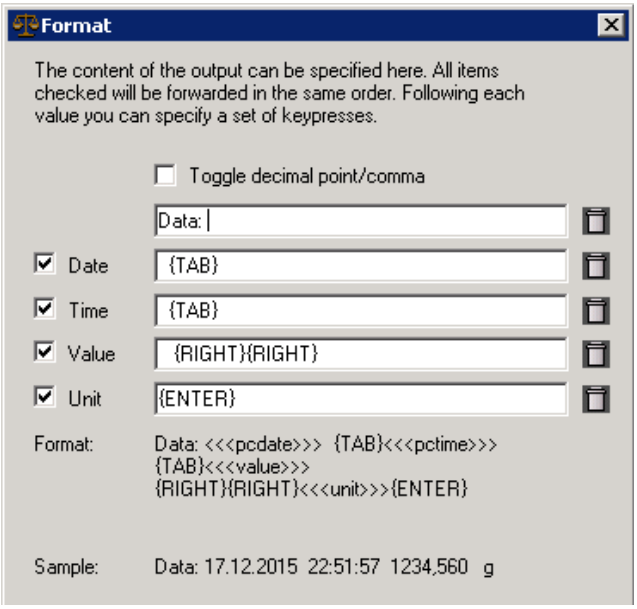
- The variable "pcdate" includes the current date of the computer. The sub-property "y" instead displays only the year.
- The variable "weight" includes the transferred weight value with sign, numerical value and unit. The sub-property "sign" extracts only the sign and this represents after year.
- The text "KERN" is displayed directly.
- The code "{F3}" triggers a keystroke on F3 for some types of outputs. For non-compatible types of expenditure is trying to convert the key press in a text element.
- "weight.value" will be resolved similar to "weight.sign" but instead displays the numerical value of "weight".
- {Left} actuates the left cursor key.

"Uppercase (unit)" calls the output function "uppercase" on the argument "unit". The variable "unit" is not included in the data packet. Therefore, the program checks for a variable with a sub-property "unit". Therefore, "weight.unit" will be evaluated and converted to uppercase.

10.3.3 Configure a simple pattern

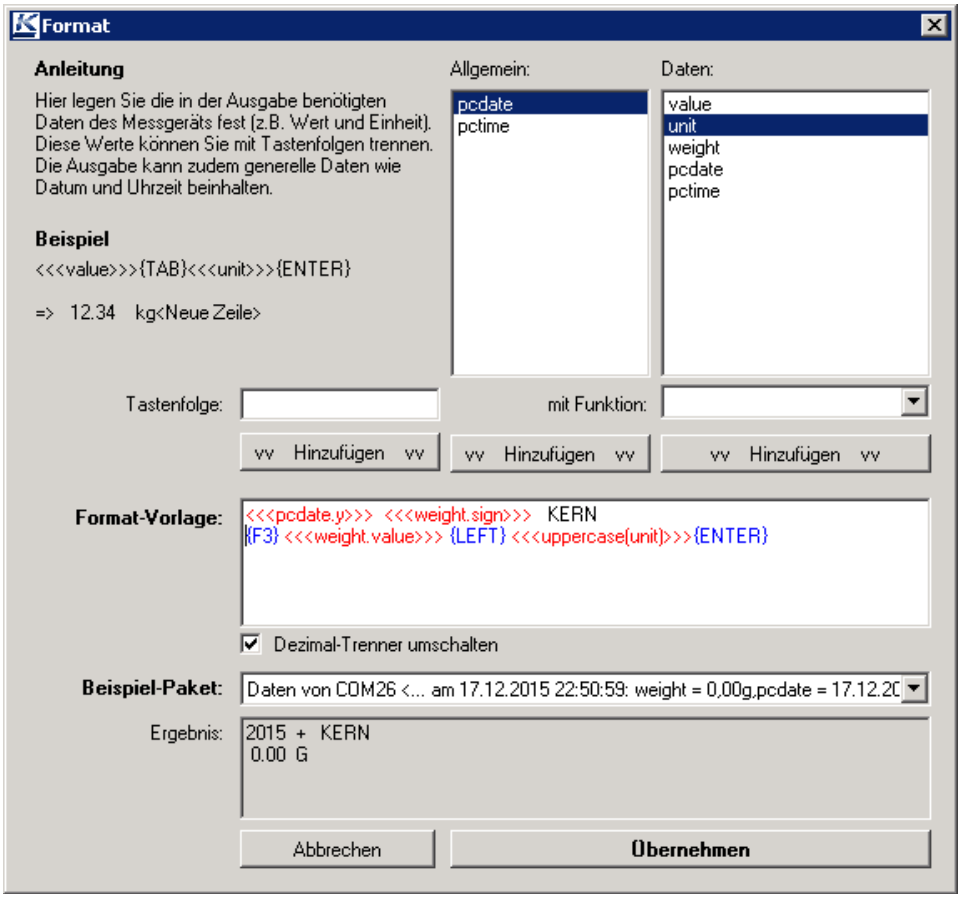
By default, when you edit an output pattern, the dialogue for a simple pattern opens first. See section 5.6 on how to operate it.

Click on "advanced" to open the configuration dialog for complex output patterns described below.



10.3.4 Configure complex output patterns

The configuration dialog for complex output patterns includes not only lists for known variables and functions but also have the option simply enter complex key sequences.



The compiled output patterns can be applied to a preview of recently received packets. To do this, select the desired package from the dropdown "sample package". The result of the pattern applied to the sample package will be shown in the field "result".

11 Output methods in detail

This section describes the output methods available in the current version of BalanceConnection. For general information on output methods see chapter 10.

11.1 Key simulation (keyboard wedge)

When the target application only provides a simple interface for entering values and does not allow import of measurement values, simple keyboard simulation may be the best solution to transfer measurement data to it. This type of transfer of measured values works with all Windows applications, however, requires cooperation with the users, since the keyboard / application cannot be used for other purposes while the program is “typing”.

The Balance Connection software supports two concepts of transfer by keyboard simulation:

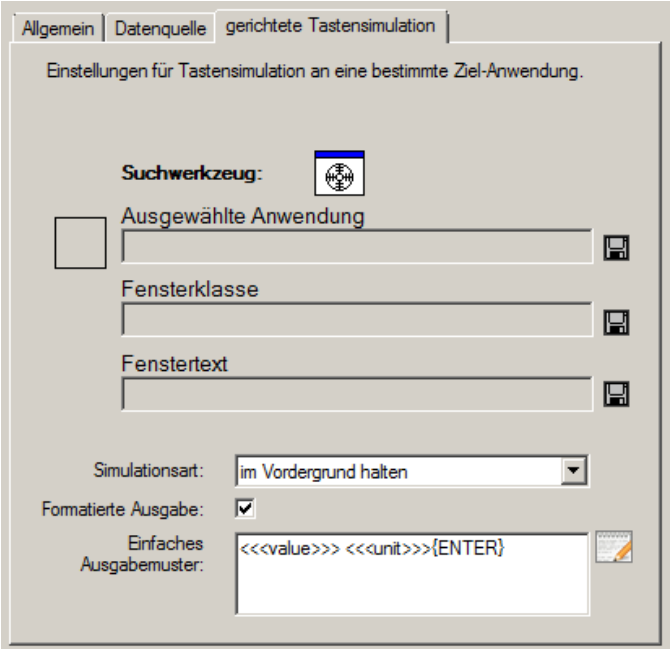
11.1.1 Global keys simulation (active application)

With global key simulation, the output is always directed to the active application. Here it is important that the input focus is in the desired (start) input field.

The configuration of the output pattern is described in Section 10.3.

11.1.2 Targeted key simulation (to a particular application)

In contrast to the global key simulation targeted key simulation allows specifying a target application and when to transfer the data:



The configuration dialog offers the following additional configuration options:

- **Target program / window:**

  The target program or the target window can be selected and saved as described in Chapter 5.3.

- **Simulation mode:**

  The simulation mode determines how and when keys will be transferred to the target application. Supported modes are:

  - **keep in the foreground:**
    On each keystroke, it is made sure that target window is still in focus. If not, it is brought into focus again.

  - **bring in the foreground (once):**
    The target window is brought into focus once and keys are simulated then.

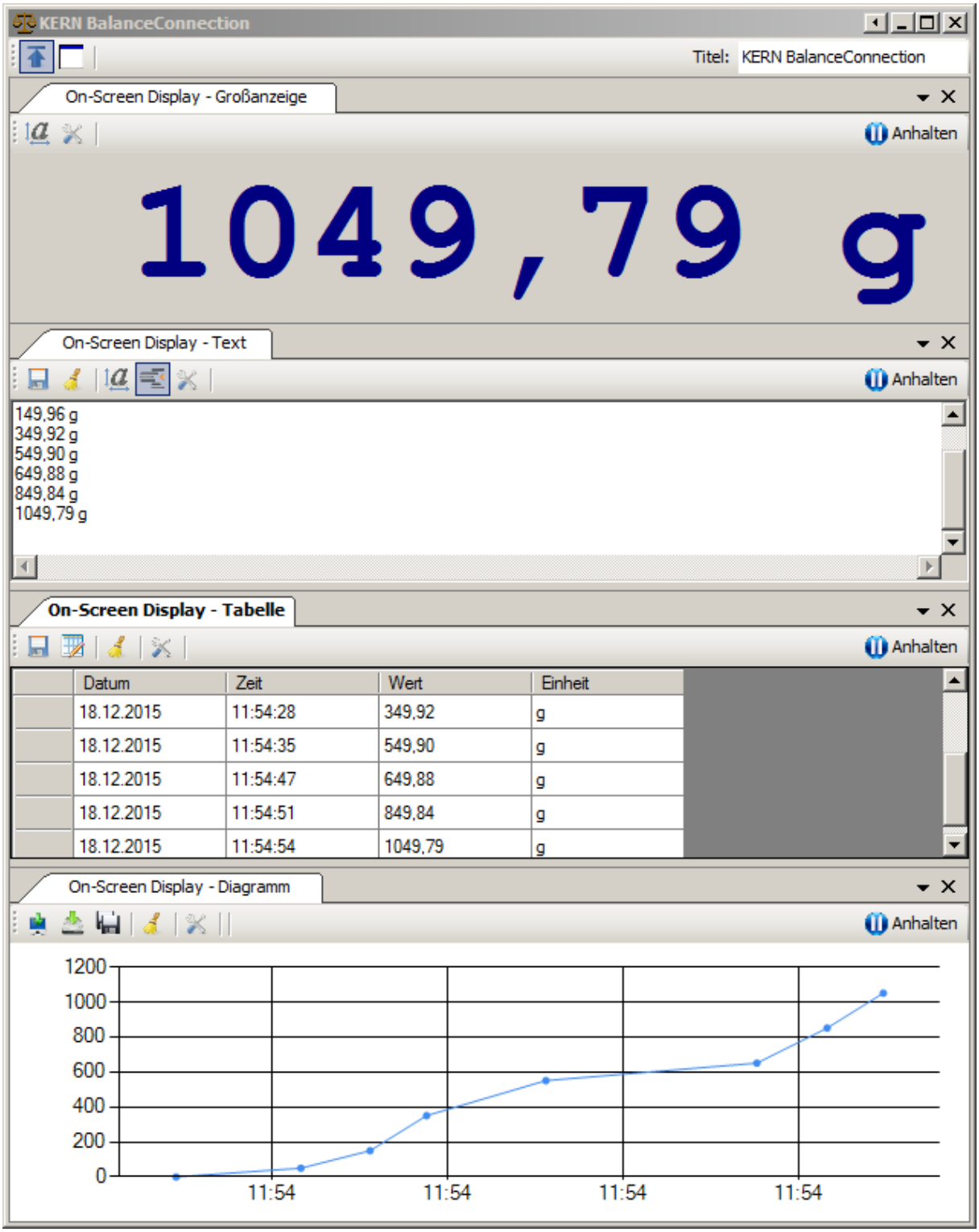
  - **wait until in foreground:**
    Keystrokes are cached until the application is activated by the user. After activating the window, all outstanding keystrokes are sent.

  - **regardless of application activation:**
    Identical to the global key simulation.

  - **send in the background:**
    The key simulation is performed via Windows messages in the background. This method is not officially supported by Microsoft and is therefore unreliable and does not work with all applications.

11.2 On-Screen Displays / In program displays

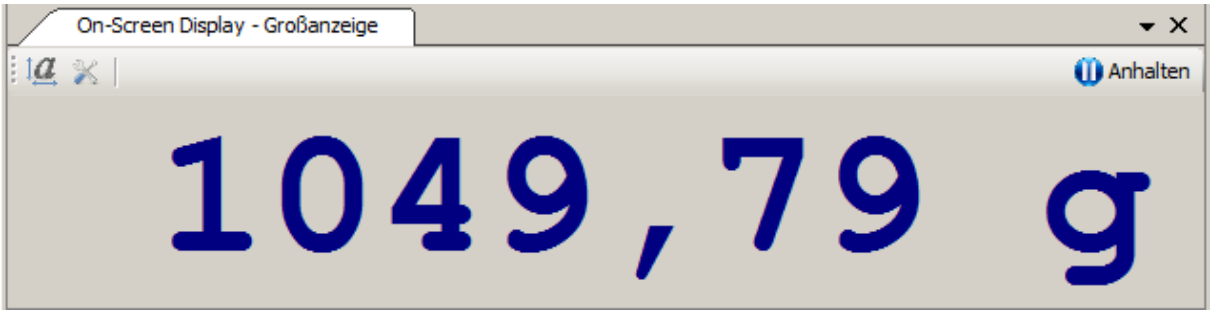
An on-screen display (OSD) is a display for measured values within the Balance Connection software. Such representations often provide a good overview of current readings of the connected devices and can be used simultaneously with a permanent record of the readings. Most on-screen displays also allow you to save the results displayed there permanently. The following screenshot shows several on-screen displays:



On-screen displays are, like other windows of the program, dockable and can be moved and rearranged within a window. Such a window can be configured so that it is always displayed in the foreground. The title of the window can be modified in the toolbar as needed. The update of each on-screen display can be interrupted by clicking on "Pause".

On-screen displays are created and configured same as the other outputs. The following sections describe the particularities of the OSDs.

11.2.1 Big display

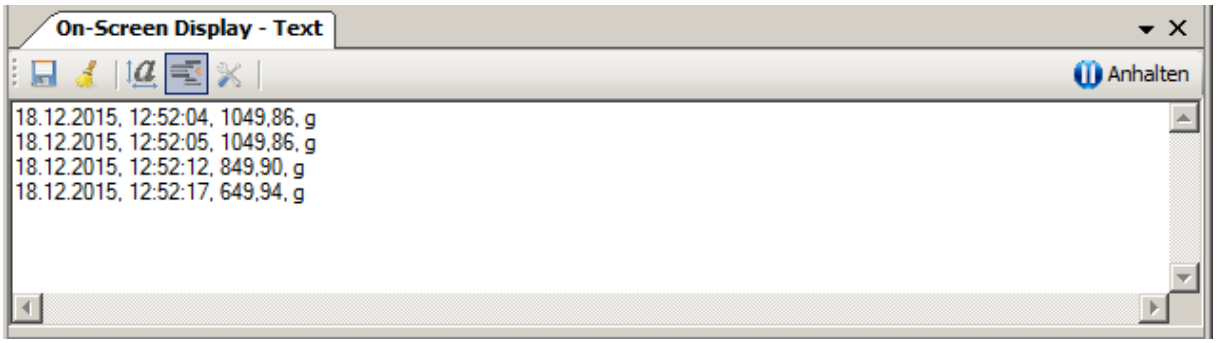


The big display shows the result of the application of an output pattern to an incoming data packet in big font in a custom color.

The configuration of font and output patterns is done via the toolbar.

Please note that the big display requires data from the scale to update. Please configure, for example, a timed trigger (timer) to request the current weighing result or set the balance to continuous transmission.

11.2.2 Recording in a text log

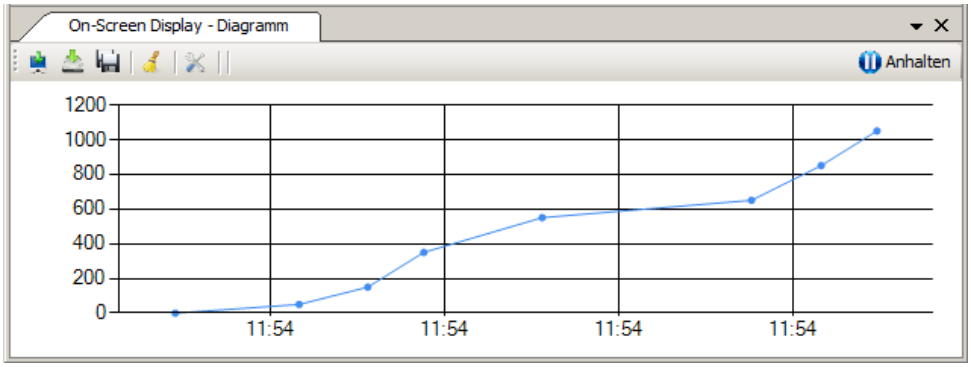


This on-screen-display works very similar to the big display, to the extent that incoming data packets are converted to formatted text using an output pattern. The resulting text here is, however, recorded in a continuous fashion, similar to a file recording or a printer.

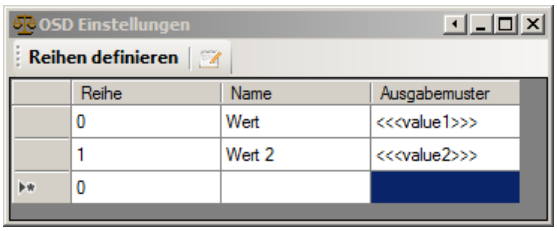
As with the big display, font and output pattern can be changed via the toolbar. Click the "Formatted output" to display raw data (useful for complicated scale output).

Using the "Save" button, the current contents of the window can be saved as a text file.

11.2.3 Chart



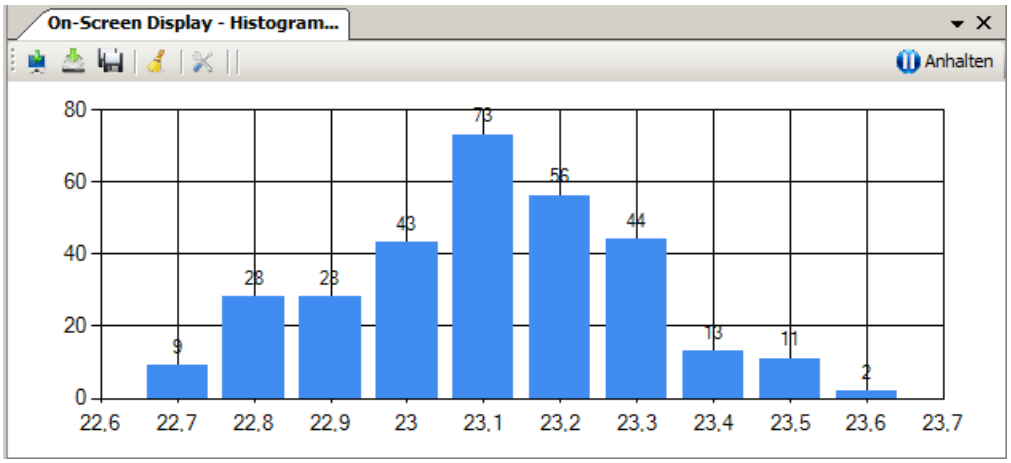
This on-screen display allows measurements in a chart/graph display (plotted over time). The data series shown in the chart is defined as combinations of name and output patterns:



The result of the output pattern has to be interpretable as a number (without unit) (eg, "123,456").

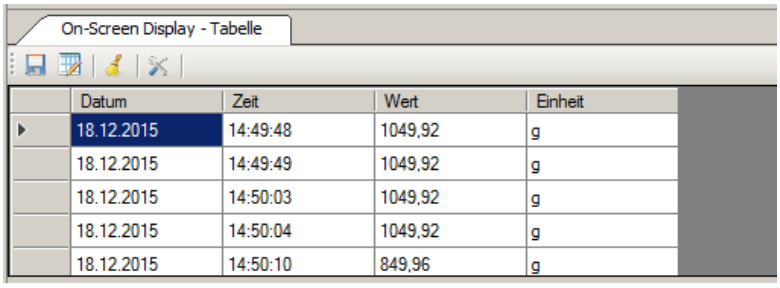
With buttons on the toolbar, the image of the chart can be copied to the clipboard, stored as a file (with manual selection of the destination) and automatically (with the current date and time).

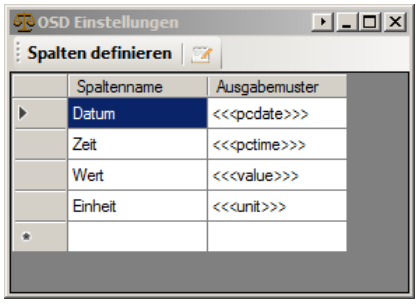
11.2.3.1 Histogram



A special type of chart is the „histogram“. It categorizes the weighing values into buckets of a given size. The on-screen display then shows a bar graph with the number of packets received for each class. This allows visualizing the underlying probability distribution easily.

11.2.4 Table





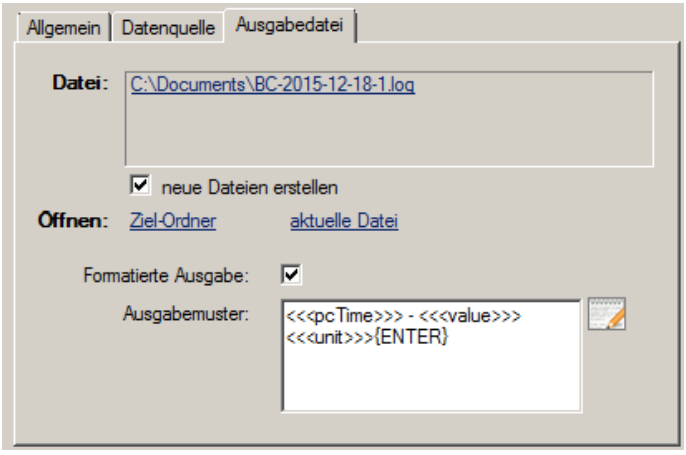
This on-screen display allows to show readings in table form. The definition of the columns shown in the table consists of combinations of name and output patterns:

If several devices are sending at the same time, use a synchronization filter as data source for this output method with multiple inputs (see chapter 12.2.1).

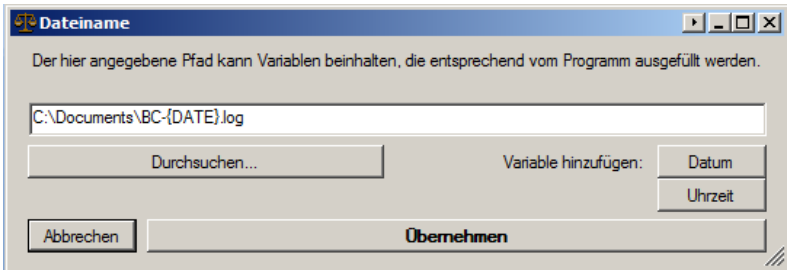
The table shown can be exported via the "Save" button as a CSV file, and can as such, for example, be processed in Microsoft Excel. However, for this use, the Excel output method is better suited (for flexible output to Excel, see chapter 0).

11.3 File record

This output method stores the result of applying an output pattern to a data packet in a text file. The following screen shows the configuration dialog



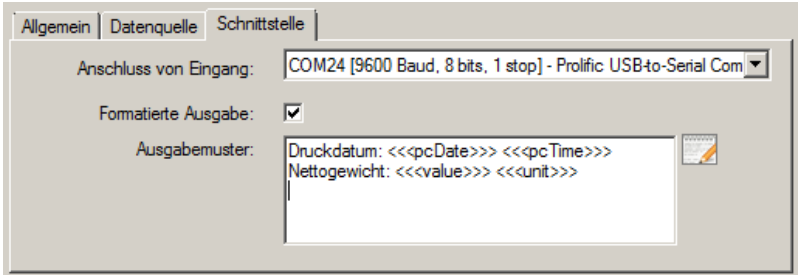
Note that the output path can include variable names.
For easy configuration of the file name, click on the file name shown above.



This shows the configuration dialog for the file name. Here the directory path and file name pattern can be set accordingly.

11.4 Forwarding to another port (e.g. RS-232 printer)

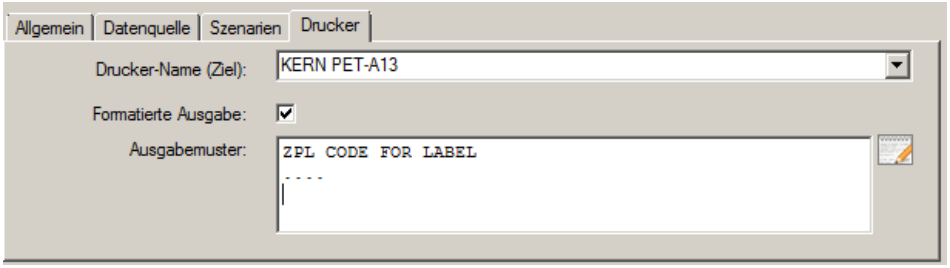
This output method sends the result of applying an output pattern to a data packet to another connected interface.



This allows, for example, to format a scale output in a particular way and forward it to a printer. Special functions for ZPL2 printers allow to use images or rendered PC fonts with compatible printers.

11.5 Forwarding to a Windows printer

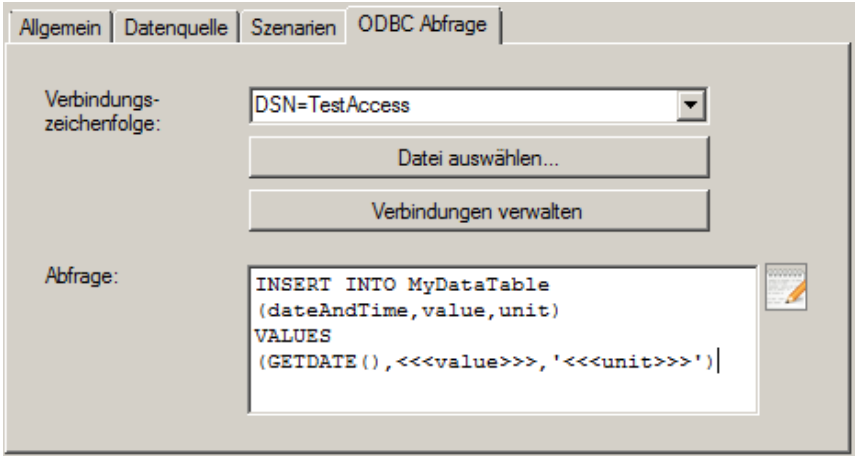
This output method sends the result of applying an output pattern to a data packet to a printer port installed in Windows. Please note that this uses only the printer port to send raw data and does not utilize the printer driver to generate data. Therefore, it will not work with PCL printers yet. Compared to the previously mentioned output to another port, this method allows to use printers connected to the PC over USB.



This allows, for example, to format a scale output in a particular way and forward it to a printer. Special functions for ZPL2 printers allow to use images or rendered PC fonts with compatible printers.

11.6 Databases / ODBC / SQL queries

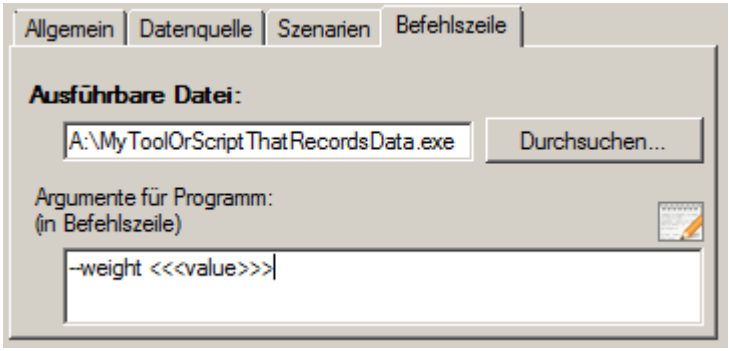
An ODBC connection allows accessing a variety of database systems like Microsoft SQL Server, Access, Excel oder third-party databases like MySQL, SQLListe or similar. For this, the correct ODBC driver has to be installed within Windows.



A universal connection string specifies the target database. It either can be a system or user DSN or a file DSN. The software creates an ODBC connection to the specified database and executes the query given by the user with variables filled in accordingly (please refer to the above image for an example). The query usually is an INSERT statement, but also UPDATE statements could be used.

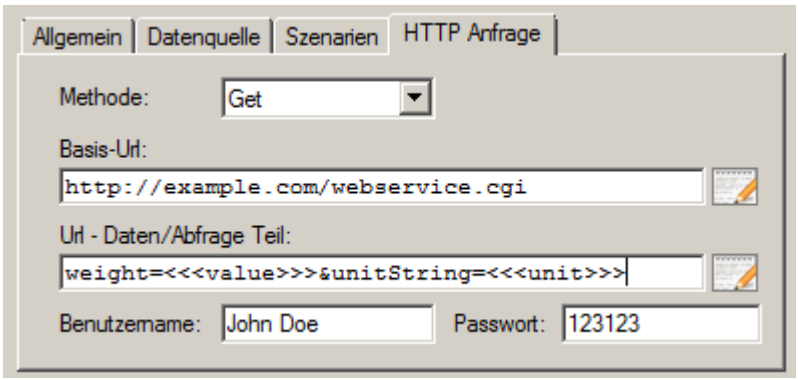
11.7 Start a program with arguments

Is there already a program or script that is able to record or process the weighing values as desired, then this output method allows to execute it with a certain set of command line arguments:



11.8 HTTP / Website / Webservice

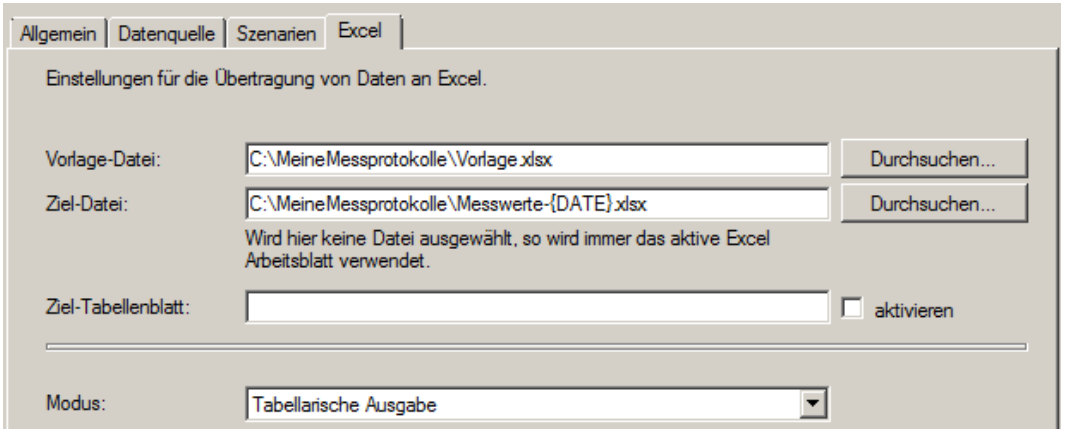
Is there already a web service or web form that is able to record or process the weighing values as desired, then this output methods allows to pass the measurement values over HTTP to the webserver within the URL query string:



11.9 Transfer data to Microsoft Excel

One of the most commonly used output methods is the transfer to Microsoft Excel. Compared to logging data first in a file or a on-screen tabular output, data here can be further processed live. Examples are formulas or charts that are constantly updated. See section 15.2 for an example application.

Another advantage of the Excel output method is that it, unlike the key simulation, can work in the background and that multiple worksheets can be filled simultaneously.



The configuration dialog for the Excel output method includes the following general properties:

- **Template file:**

  If the target Excel file does not exist or the current file name is no longer valid (for example, on the next day), the template file specified here is opened and saved as a new target file.

- **Target file:**

  The active Excel application is searched for the file specified here. If it is found, the data will be transferred to the open file. If is not found, the file will be either be created based on the template (if defined) or opened as an empty file.

  If no target file is specified, the output will be directed to the active worksheet.

- **Target worksheet:**

  When the target file contains multiple worksheets, the desired worksheet can be defined here. If this is empty, the active worksheet is used.

  This is particularly useful when multiple outputs try access the same Excel file.

  If the "activate” control box is set, the target worksheet will be activated (brought to the front) before inserting data.

- **Mode:**

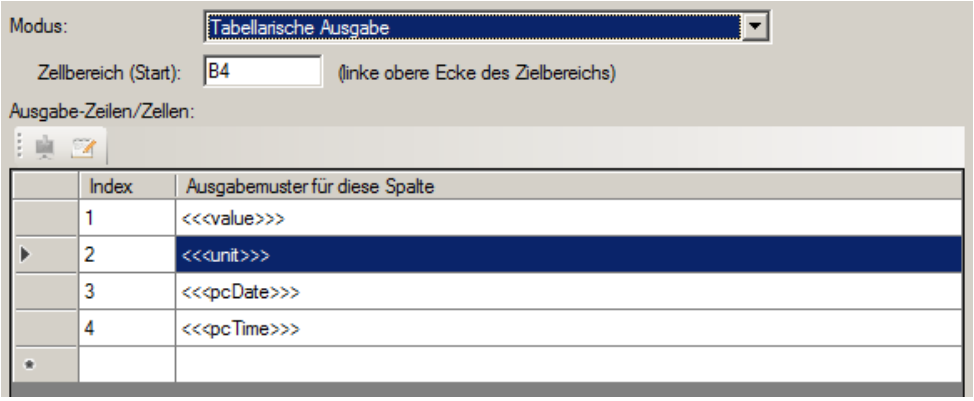
  The specified mode here describes which cells are to be filled with what content. The modes and their configuration are described in more detail in the following sections.

  If no mode fits the task, it may be possible to use an output scenario (see chapter 10.2.3).

11.9.1 Excel Output Mode "tabular output"

The mode "tabular output" fills, starting from the given top-left corner cell, a table column by column with defined values. The values are always entered into the next completely blank line.

For example, if the start area and column definitions filled as follows ...



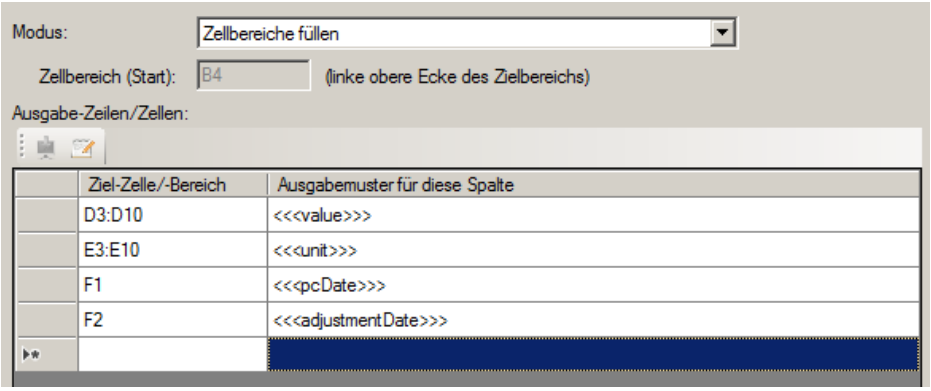
... the result is a table of this structure.

	A	B	C	D	E
1					
4		1049,96	g	18.12.2015	17:43:02
5		1049,96	g	18.12.2015	17:43:04
6		1049,96	g	18.12.2015	17:43:05
7		1049,96	g	18.12.2015	17:43:05

Any cells not affected by the table definition can be pre-filled with other values or formulas.

11.9.2 Excel output mode "fill cell ranges"

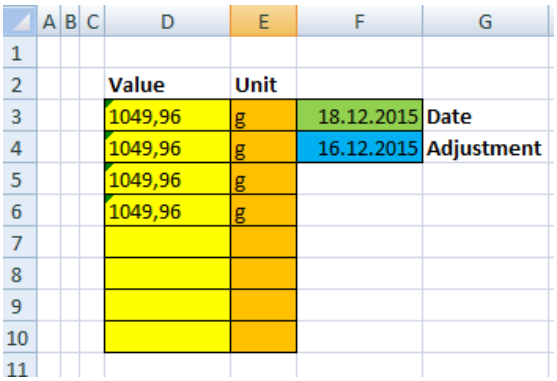
When a simple tabular output is not possible or too inflexible, the mode “fill cell ranges" offers more configuration options. In this mode, for individual cells or entire cell ranges output patterns can be define, as demonstrated in the following window:



A range of cells can be specified in usual Excel style as a single cell address ("F1"), as a range of cells ("D3:F10") or as the name of the cell (if defined).

The outgoing data is always filled into the next free cell in a range. If only one cell is specified, the value of this cell will always be overwritten.

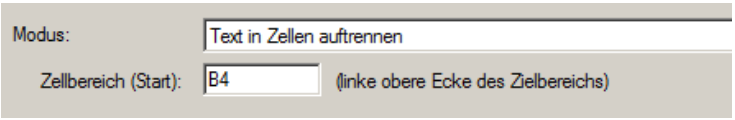
The specification shown above fills the following Excel spreadsheet correctly:



	A	B	C	D	E	F	G
1							
2				Value	Unit		
3				1049,96	g	18.12.2015	Date
4				1049,96	g	16.12.2015	Adjustment
5				1049,96	g		
6				1049,96	g		
7							
8							
9							
10							
11							

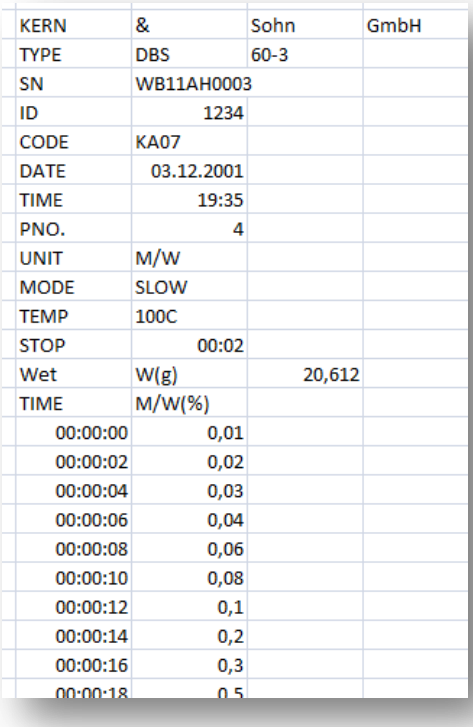
11.9.3 Excel Output Mode "separate text into cells"

Is the interface protocol of a device very complex and you want to process all data sent in Excel anyway, consider using the mode "separate text into cell".



Here, the transmitted raw data from the device (as text) will be split at set separator characters (eg spaces, tabs) into parts and these parts are then stored into individual cells of the Excel file. The split up data can then be further processed easily and individually.

As in tabular mode, the data is output in the first free cell of the target range.

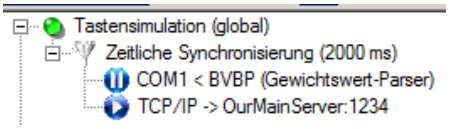


KERN	&	Sohn	GmbH
TYPE	DBS	60-3	
SN	WB11AH0003		
ID	1234		
CODE	KA07		
DATE	03.12.2001		
TIME	19:35		
PNO.	4		
UNIT	M/W		
MODE	SLOW		
TEMP	100C		
STOP	00:02		
Wet	W(g)	20,612	
TIME	M/W(%)		
00:00:00	0,01		
00:00:02	0,02		
00:00:04	0,03		
00:00:06	0,04		
00:00:08	0,06		
00:00:10	0,08		
00:00:12	0,1		
00:00:14	0,2		
00:00:16	0,3		
00:00:18	0,5		

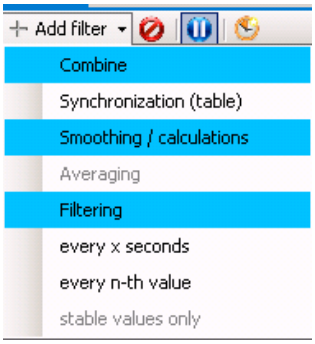
12 Output Filters

In the "pipeline" of data packets, output filters can be interposed between an input and an output. These filters allow you to filter out data packets, alter them (computed fields) or to synchronize packets (record multiple scales at the same time).

12.1 General information about filters



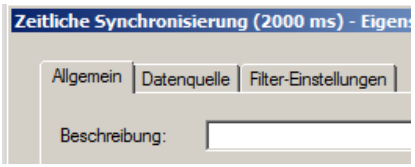
An output filter can be added through the filter drop-down menu Select the type of filter required.



Analogous to an output, a filter has a description field and a data sources tab. Additionally if has other parameters that depend on the type of filter. These are described in more detail below.

12.2 Types of filters

12.2.1 Synchronisation (-> Table)



If multiple devices are connected to the same computer, each device will send its measurement value to a different point in time to another input. In some use cases, it is necessary to synchronize the inputs – to “bring them to one line”.

The filter "time synchronization" serves to synchronize the processed data packets from several devices; that is simultaneously forward them to an output. Depending on the nature of the output, this allows to fill or to form a table of values or calculate values by combining different measurement devices.

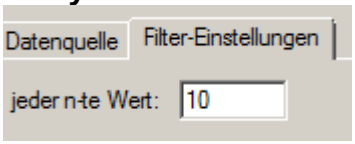
Up to what point measurements of the devices arrive at the “same time” is defined by the property "time slot".

If a device does not responds within this period its variables will remain undefined in the final packet. The variable names in the data packets of the responding devices are passed on with an index (suffix, starting with 1).

12.2.2 Each n-th value / All x milliseconds

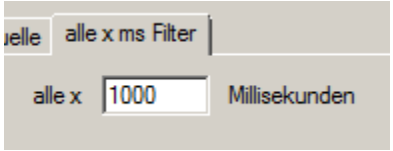
When a measuring instrument is set to continuous transmission, it often provides values faster than the target application can process. The following filters allow discarding superfluous values:

Every n-th value



Passes only every n-th value. Other values are discarded.

All x-millisecond



Passes a value only every x milliseconds. Other values are discarded.

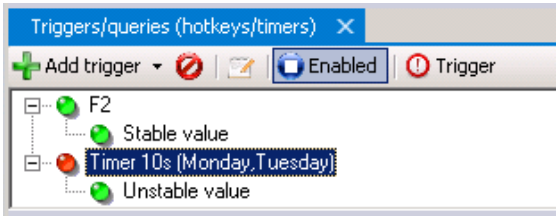
13 Triggers / Queries

Most measurement devices with communication interface allow to trigger certain functions / operations of the device via the interface by means of remote control commands. A "trigger" specifies when what device will trigger a particular operation with certain parameters.

A "device operation" can be a query of the measured value, setting an internal parameter or trigger the adjustment, zero or tare of the balance. Depending on operation, several parameters may be necessary. Also whether the device acknowledges or answers depends on the type of operation and device.

13.1 Overview

The list of triggers in the main program window shows all currently configured triggers.

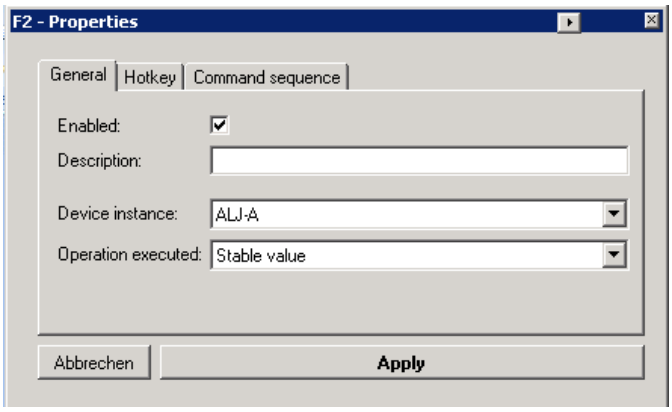


As with the other main windows you can alter the state, as well as the parameters of the trigger on the toolbar and the context menu.

The toolbar also offers the possibility with the “trigger" button to execute the underlying action manually. If a trigger is fully configured, it also can be triggered by double-clicking on it.

13.2 General information about triggers

Trigger on the dropdown button can be added with the "Add trigger" button. As with the other dialogs of the program, choose the type of the trigger from the context menu. This will open the Properties dialog of the trigger:



The general part of the properties of a trigger includes the following items:

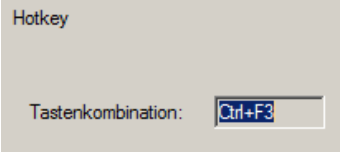
- **Enabled:** Determines whether the trigger is activated / is operational.

- **Description:** The name you enter here will be displayed instead of the automatically generated name of the trigger.
- **Device instance:** The device instance on which the operation has to be triggered.
- **Device operation:** The device operation to be triggered. Alternatively, "command sequence" can be selected. In this case, the tab "command sequence" specifies the commands to be run within the context of the trigger.

Below, all available types of triggers are described in more detail.

13.3 Hotkeys / shortcuts

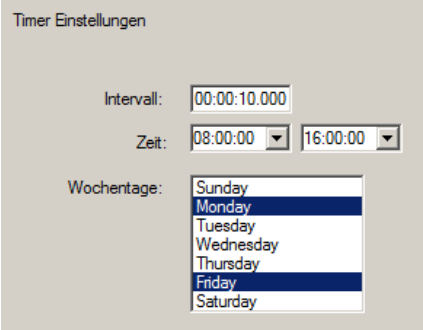
A "hotkey" trigger waits for the press of a key combination within Windows. The desired key combination is defined in the tab "Hotkey":



Press the desired key combination in the input box for it to be recognized.

13.4 Timers / time controlled requests

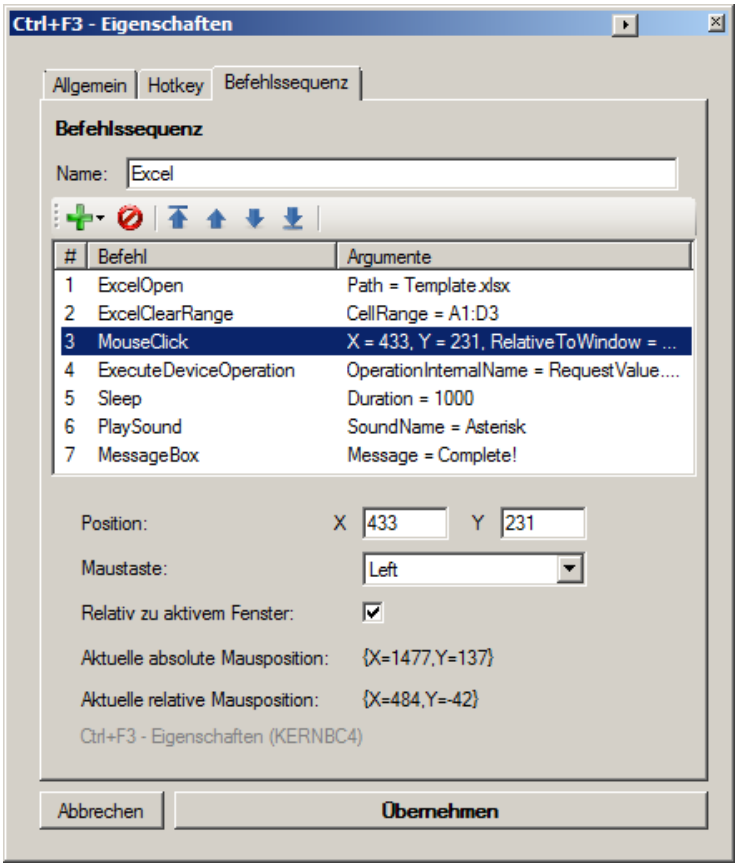
A "timer" trigger triggers the desired action in a defined regular time interval.
The active time can be further constrained by specifying start and end time as weell as active weekdays.



14 Command sequences

In the context of triggers or output methods complex command sequences can be triggered instead of only the hard-wired standard function. A command sequence consists of a series of commands with individual parameters. Therefore they can be compared to scripts; however the flexibility is not as high as in written code.

Examples of command sequences are clicks, keystrokes, activation of windows or as a special example highlighting a cell range in Excel.



The processing of an instruction sequence is analogous to the other main windows of the program. Command steps can be individually added, deleted or moved within the command sequence.

Each command has a number of parameters. Depending on the type of the command is the processing of the command parameter from a simple list. For complex commands a special input screen helps you configure.

Depending on the context of the command sequence (within a trigger or different output methods) are not all of the commands available.

If you have problems with command sequences, please note the output in the program log (see section 7.4).

15 Application and configuration examples

15.1 Print a user-defined label on a YKC-01 printer

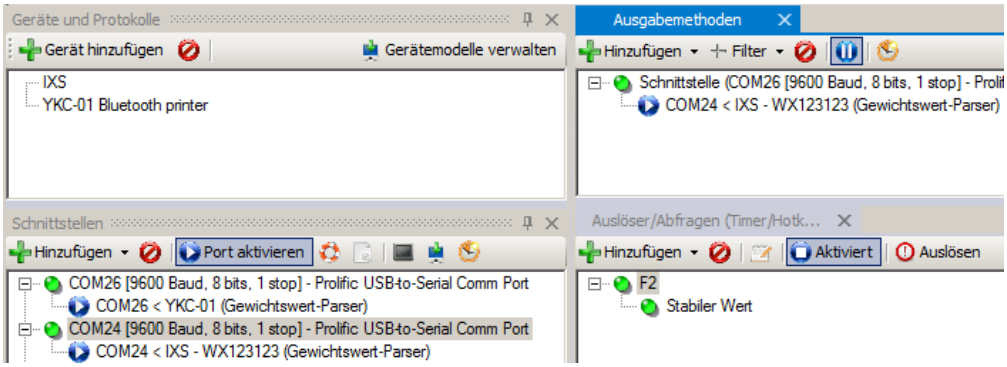
Hardware Setup:

- **COM26**: KERN YKC-01 Bluetooth printer over YKI-02 Bluetooth adapter (or built-in PC Bluetooth interface)
- **COM24**: IXS KERN platform scale

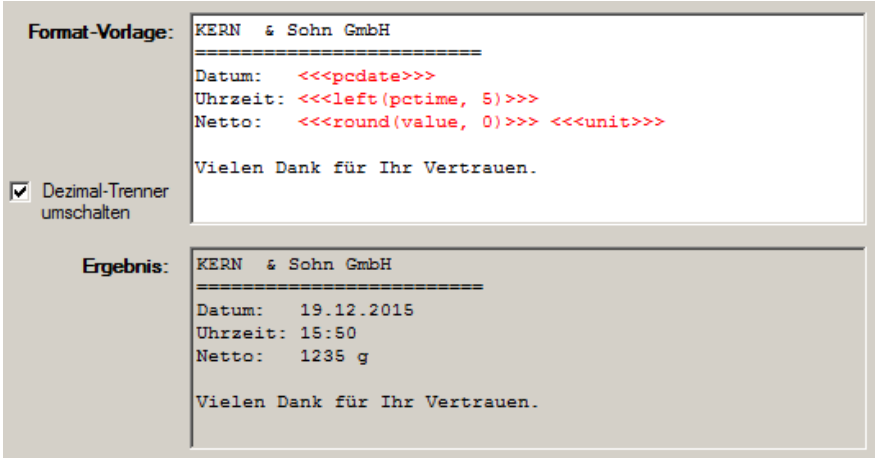
Goal:

- By pressing a key, the current value of the balance is to be requested and it should be printed on a printout with date / time (but the weighing value should have no decimal places).

Configuration of inputs and outputs:



Printer and balance are connected to different ports. For request, a hotkey trigger (F2 key) is used, which requests the stable weighing value. As output method redirect to an interface is setup (printer to COM26). The output pattern is defined as follows:



Functions are used to round the weighing value and remove the digits part of the PC time. The full label contents will be printed after pressing the F2 button.

15.2 Drying protocol with the drying process for a moisture analyzer

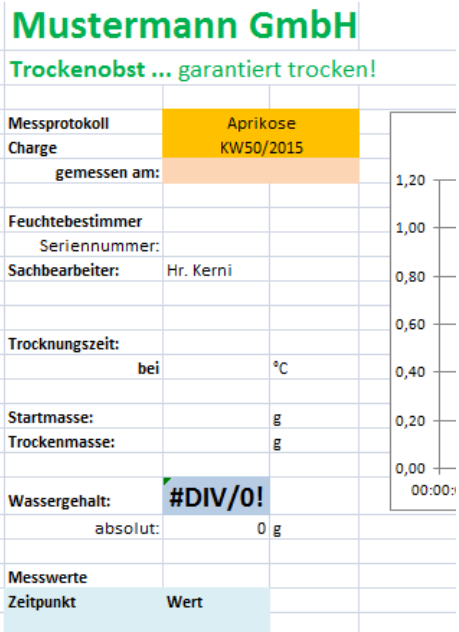
Hardware Setup:

- DBS 60-3 via USB on COM25

Goal:

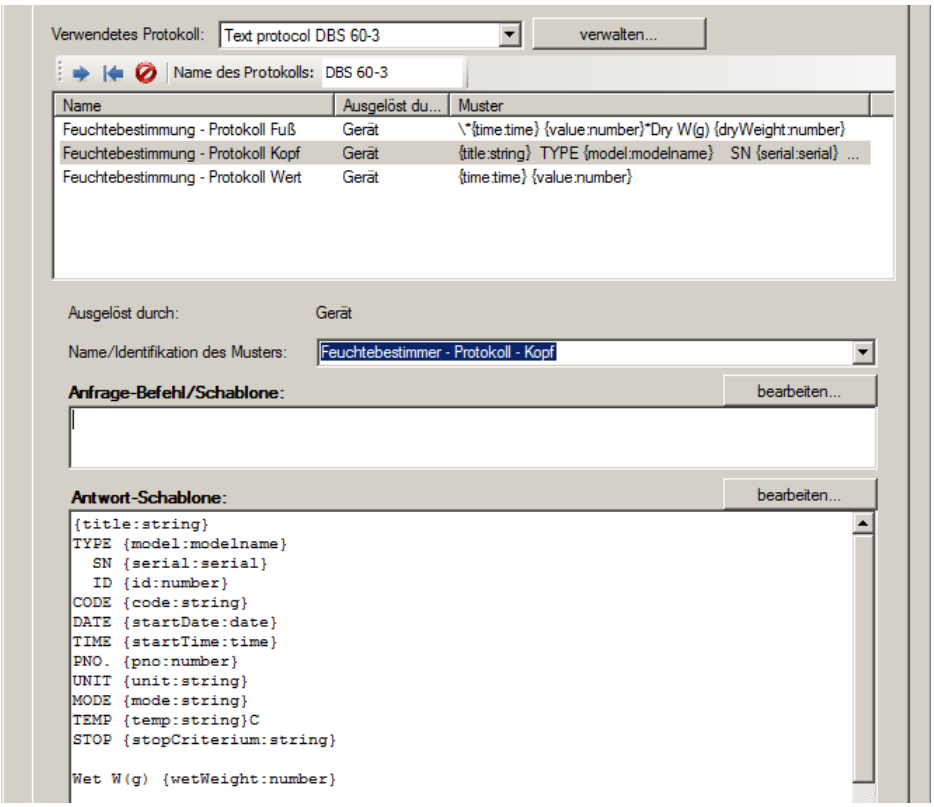
- Recording a drying process in an Excel form during the drying process. The current drying value is to be displayed in a large format display.

- Excel form (empty state) to fill (right)



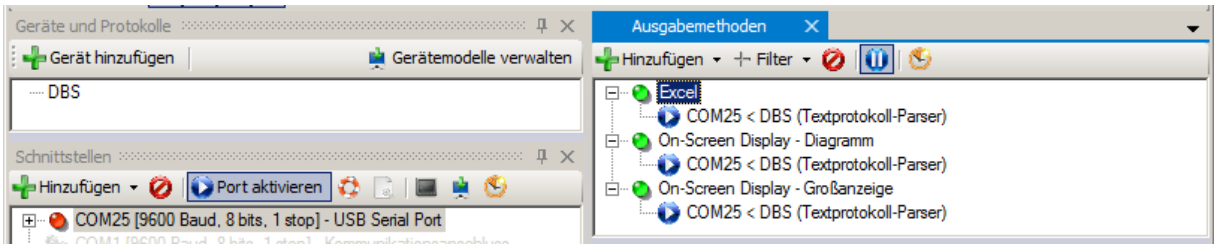
Configuration - interface protocol:

The interface protocol of DBS 60-3 for a measurement is divided into three parts: head, single value or footer. The variables defined here may be used for filling the form.

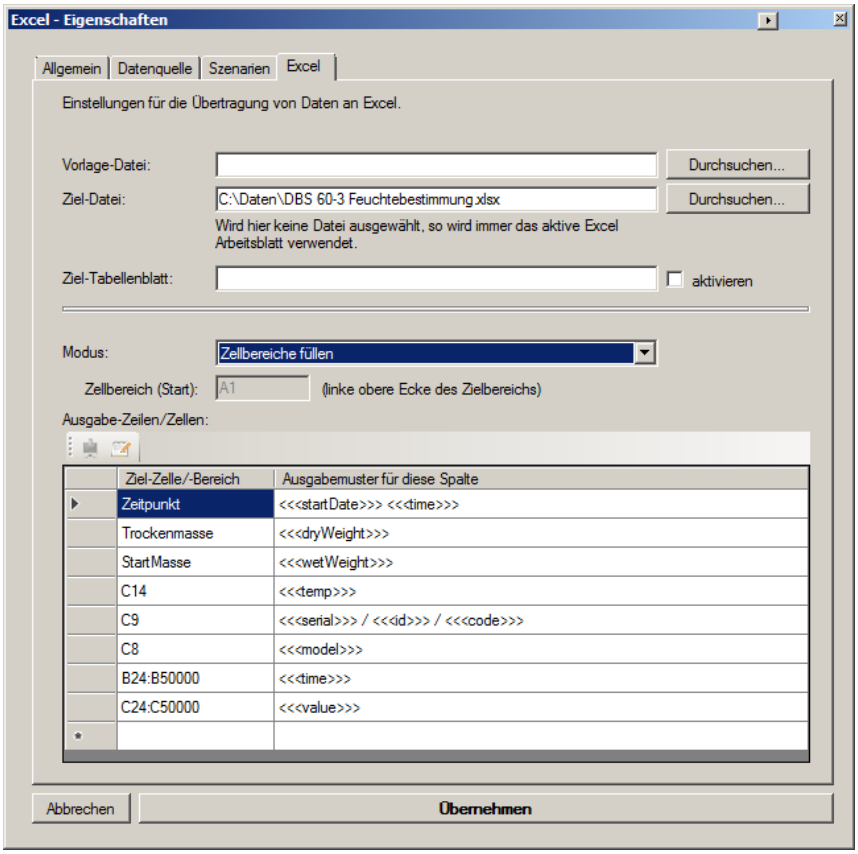


Configuration - Inputs and outputs:

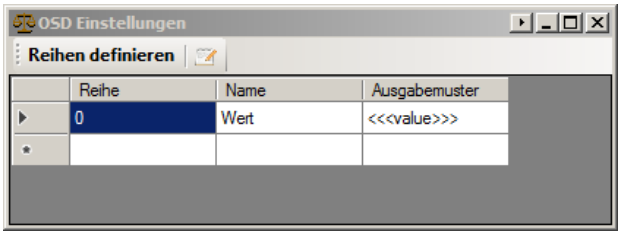
The following view shows the inputs and outputs configuration.
All outputs have a source to DBS moisture analyzer. It itself connected to COM25.



As mode for Excel output "cell ranges fill" is used. There are all target cells indicated:



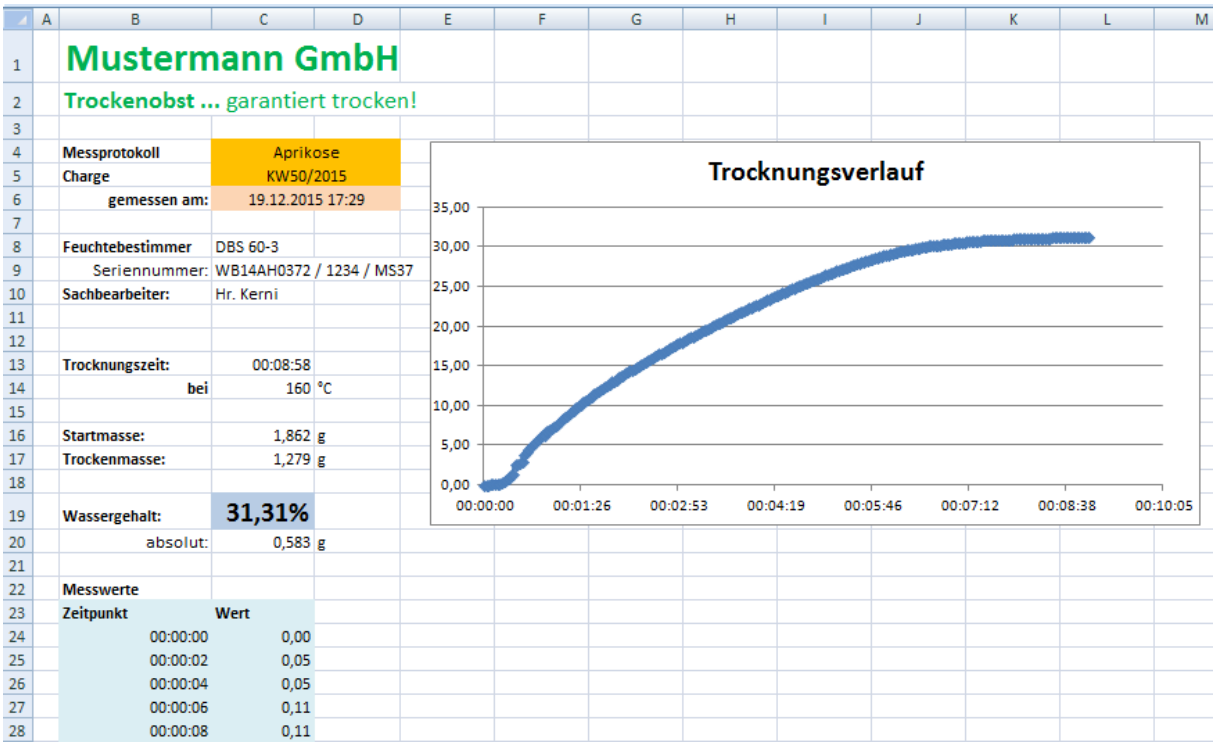
The series definition for the on-screen diagram is very simple:



The big display is defined analogously (output pattern: "<<<value>>> %")

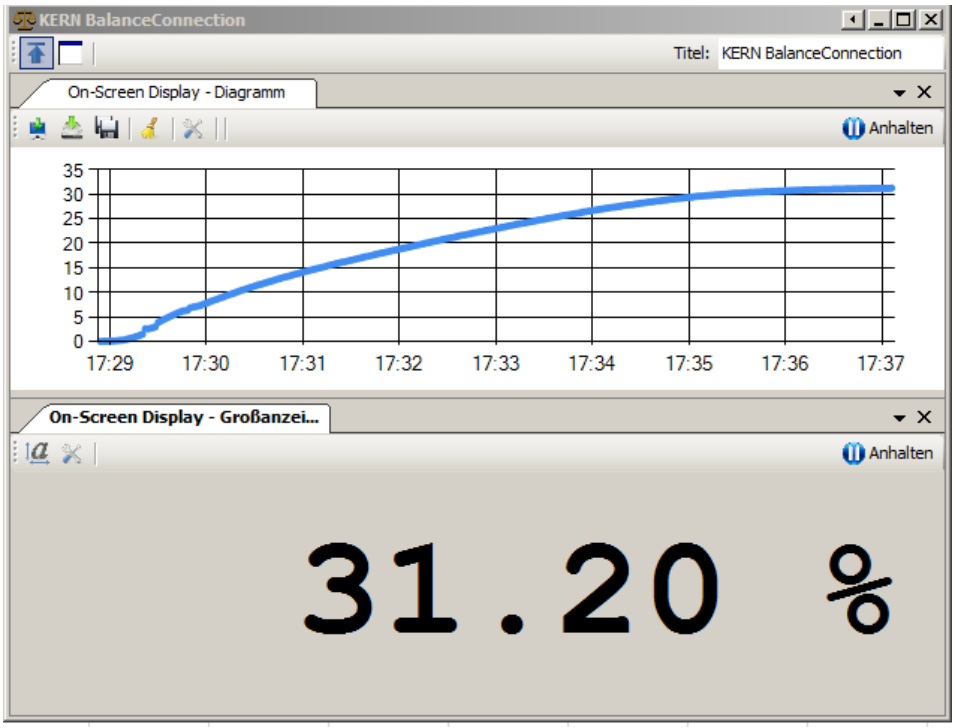
Result:

After completing a moisture measurement with live transmission, the result can be seen on the PC. The drying process was recorded both in Balance Connection itself as well as in Excel.



1	**Mustermann GmbH**		
2	**Trockenobst ...** garantiert trocken!		
3			
4	**Messprotokoll**	Aprikose	
5	**Charge**	KW50/2015	
6	**gemessen am:**	19.12.2015 17:29	
7			
8	**Feuchtebestimmer**	DBS 60-3	
9	Seriennummer:	WB14AH0372 / 1234 / MS37	
10	**Sachbearbeiter:**	Hr. Kerni	
11			
12			
13	**Trocknungszeit:**	00:08:58	
14	**bei**	160	°C
15			
16	**Startmasse:**	1,862	g
17	**Trockenmasse:**	1,279	g
18			
19	**Wassergehalt:**	**31,31%**	
20	absolut:	0,583	g
21			
22	**Messwerte**		
23	**Zeitpunkt**	**Wert**	
24	00:00:00	0,00	
25	00:00:02	0,05	
26	00:00:04	0,05	
27	00:00:06	0,11	
28	00:00:08	0,11	

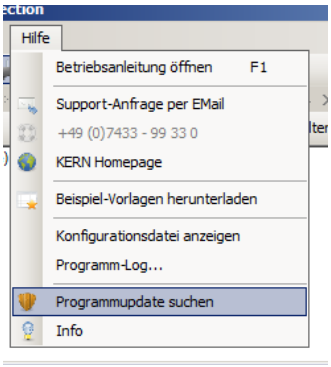
Result in Balance Connection (simultaneously):



The diagram in Balance Connection can be saved as an image.

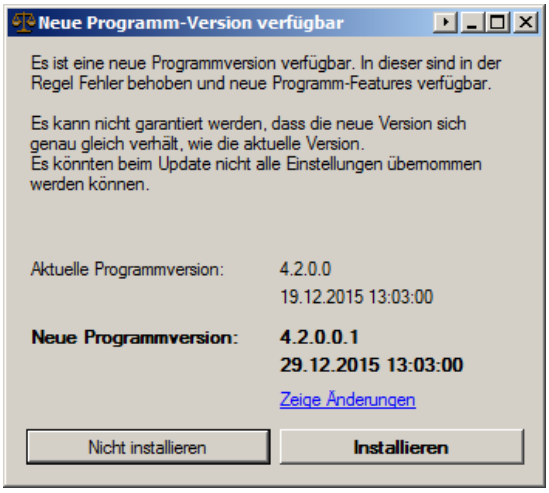
16 Software Update

The Balance Connection software is regularly updated to correct errors and to implement new features and improvements. You will receive software updates via the Internet and by contacting KERN.



In Info dialog box and the Help menu of expert mode, you will find "Search Program Update" option. Please make sure before clicking that the computer has a working Internet connection (TCP / IP port 80).

If a new program version available, so the dialog box to the right appears.



If you want to install the update, press the "Install" button.

Please note that a program update can also bring unexpected / unwanted changes. Furthermore, it is possible that your configuration cannot be adopted without errors in the new version.

17 Common Errors / Troubleshooting

Values are not passed on correctly.

Please proceed as follows:

- Check the status of inputs, outputs and filters. Do those already shown an error?
- Open the terminal for the interface used.
  Can you see data here?
- Open the most recently received packets of the input.
  Is data visible here and has it been parsed correctly?
- Open the most recently received packets of the output or intermediate filters.
  Is data visible here and has it been parsed correctly?
- Check the program log for errors.

The interface protocol in the software does not fit on my KERN scale.

Due to product changes, improvements on the interface protocol may have been carried out so that the software version is no longer compatible with the revision of the scale. Please contact KERN in this case.

The program is crashing. / The program does not start.

Please contact KERN. You will receive an immediate solution or a software update.

Appendix I. Software License Agreement

Single User Licence

By using this software or declaration of consent - which constitutes the precondition for the installation of this software – you agree to the terms and conditions of this **Software Licence Agreement** between you as the customer (**licensee**) and Kern & Sohn GmbH (**licensor**).

§ 1 Copyright

1.

This software is protected by copyright under the provisions for the protection of computer programs. The copyright encompasses in particular the program code, documentation, appearance, structure and organisation of the program files, program name, logos and other forms of expression within the software. The licensor in his capacity as manufacturer reserves all rights arising from copyright.

2.

The licensee shall undertake not to disclose this software or documentation, either in whole or in part, to third parties, or to forward the aforesaid, except where the licensee is permitted to do so under the provisions of this contract by virtue of another written agreement with the licensor.

3.

Copyright mention, serial numbers as well as other features serving program identification must not under any circumstances be removed or changed.

§ 2 Usage Rights

The assignment of usage rights for the software is subject to the following agreements:

a) Scope of Licence

1.

The licensee shall obtain elementary, non-exclusive usage rights, categorically without time limit, for the software as well as related documentation for own use after unconditional payment in full of the licence fee (acquisition cost of licence).

2.

Once the usage right has expired, the licensee shall surrender the software, possibly existing copies thereof as well as any documentation to the licensor. If physical surrender of the software and copies thereof turns out to be impossible due to technical reasons the licensee shall delete them and confirm this action in writing to the licensor.

The software must not be used for, or in connection with, the operation of nuclear power plants, aeroplanes, communication systems, air traffic control, in connection with life-sustaining devices or other life production systems.

b) Reproductions

1.

The licensee may copy the software insofar as this is necessary for the use of the software.

Necessary reproductions include installing the software from the original data carrier to the hard disk of the hardware as well as loading the software to the main memory.

2.

The licensee may copy the software once for the purposes of data backup to a permanent data carrier. Saved copies of the software must be specifically identified as such.

3.

Any other reproductions (including the issue of the program code to a printer and the production of printouts intended for photocopying the program description) are prohibited.

c) Multiple Use

The software is intended for one workstation only and must not be used within a network or any other kind of multiple workstation computer system if this would entail a simultaneous use of the software.

d) Redistribution

1.

The licensee may pass on permanently the software and related documentation to third parties provided the acquiring party acknowledges the binding nature of the terms and conditions of this agreement with respect to copyright and usage rights.

After redistributing the software to a third party the licensee will no longer be permitted to use the software. Redistribution requires that existing software copies are either handed over to the acquiring party or, that copies not handed over are destroyed.

2.

The software must not be hired out for commercial purposes.

Apart from that, the licensee may relinquish the software for a fixed period of time to a third party provided the third party acknowledges the binding nature of the provisions of this agreement and provided the licensee hands over all existing software copies including possibly existing backup copies or destroys all copies not handed over.

3.

The licensee may not relinquish the software to a third part for use or pass it on to a third party if there is any suspicion that the third party is likely to violate the provisions of this agreement.

e) Decompilation and Changes to Programs

1.

Reverse translations of supplied program codes into different code types (decompilation) and other kinds of redevelopment of the various creation steps of the software (reverse engineering) are prohibited. If interface information is required for the purposes of establishing interoperability for an independently created computer program, this can be acquired – insofar as technically feasible on the part of the licensor – against reimbursement of outputs from the licensor or a third party to be designated by him.

2.

Program translation, editing, among other things re-engineering of software or splitting and multiplication of results thus achieved shall only be allowed insofar as this is required for the use of the software by the licensee.

§ 3 Software Warranty

1.

If there are faults present in the software, the licensor shall either repair or deliver in addition (supplementary performance) according to licensee's choice.

2.

If the licensor is unwilling or unable to provide supplementary performance or if the deadline for such supplementary performance is exceeded for reasons that are the responsibility of the licensor or if supplementary performance fails in any other way, the licensee shall be entitled within the framework of existing legislation to further warranty claims.

§ 4 Liability of the Licensor

1.

The licensor shall accept liability for deliberate action and gross negligence. The licensor shall accept liability for ordinary negligence only insofar as major obligations have been violated. The licensor's liability for ordinary negligence shall be limited in value to the foreseeable damage typical of the contract.

2.

In case of initial impossibility, the licensor shall only accept liability if he was aware of the performance ratio or if he was unaware of it due to gross negligence.

3.

Claims for defect and claims for compensation shall be time-barred after one year from the commencement of the statute of limitations period.

4.

The limitations of liability above shall not apply to claims made for injury to life, limb or health resulting from negligent breach of duty by the licensor or deliberate or negligent breach of duty by a legal representative or persons employed in performing an obligation of the licensor.

5.

The licensor shall not accept liability in case of any use by the licensee contrary to the terms of this agreement.

§ 5 Miscellaneous

1.

This licence agreement shall be governed by German Law.

2.

Exclusive place of jurisdiction for all disputes arising from the commercial dealings in connection with this contractual relationship shall be the licensor's place of business. The licensor shall also be entitled to open proceedings at the licensee's place of business.

3.

The licensee shall not be entitled – subject to different provisions in this software licence agreement – to transfer unfulfilled rights or the full agreement to a third party unless the licensor gives his explicit consent in writing. The licensor shall give his consent if legitimate concerns by the licensee with respect to the transfer of rights outweigh the interests of the licensor.

4.

Alterations of this software licence agreements must be drawn up in writing; this shall also apply to an alteration of this written-form clause.

No collateral agreements have been made.

5.

If individual provisions of this licence agreement should be or become ineffective, this shall not affect the validity of the remaining provisions of this licence agreement.

§ 6 Support

We are happy to help your with technical issues arising during the use of this software. Please contact us under.