



Datafox GmbH • Dermbacher Straße 12-14 • D-36419 Geisa • www.datafox.de

Manual DatafoxStudioIV

Flexible data collection with method



© 2013 Datafox GmbH

This document has been created by Datafox GmbH and is copyrighted against third parties. Datafox GmbH considers all contained information, knowledge and depictions as its sole property. All rights, including also translation, reprint or copy of the whole document or parts of it, require written consent of Datafox GmbH. The assertion of all rights in this respect is reserved to Datafox GmbH, especially in case of the grant of a patent. The handover of this documentation does not establish a claim to the license or the use of the soft- and hardware. Copies of the floppy disks and CDs may only be made for the purpose of data backup. Every unauthorized copy of this documentation or the Datafox software will be prosecuted.

This figure shows which Datafox devices of each section below is valid.
 If a function valid for all devices then is the chapter without this figure.
 The specific device or devices for which this section applies are marked with ☒.

									
☒	☒	☒	☒	☒	☒	☒	☒	☒	☒

Alternations

Alternation in this Dokument

Date	Chapter	Discription
15.03.2013	all	Revision the manual to new version 04.02.04.xx Please note that not all chapters are in English. We are working on it.
05.08.2013	6.3.4 / 5.7	Translate

Alternations of the version

With the device generation IV a new versioning scheme has been introduced. According to this scheme the file name of the device firmware and the setup program (DatafoxStudioIV) is composed as follows:

Product name	XX. Device genera- tion	YY. Compatibility (which versions can be used to- gether)	ZZ. Version number (functional exten- sion)	Build Troubleshooting (with a new version the Build number is reset)
z. B. AE-MasterIV	04.	02.	01.	04

The use of the manual depends on the version of the firmware and the DatafoxStudioIV or the DFComDLL. Gather from the following table which manual matches which version. For different combinations no support can be offered.

Firmware StudioIV and DLL validity

Firmware: 4.02.04.xx.
Studio: 4.02.04.xx
Dll: 4.02.04.xx

The DatafoxStudioIV is backward compatible. This means that you can configure a device with a newer DatafoxStudioIV also older firmware, the device only supports the natural functions that are implemented in the older firmware version. I.e., relevant to the functions that are possible, is always the manual state that the firmware associated with the Setup equivalent. It is not possible to provide a centering firmware configured with a stand of DatafoxStudioIV to who is older than the firmware. recommendation:

If possible, use always the current version of DatafoxStudioIV.

What features are supported in which software versions, is from the file:

Datafox MasterIV, SW version xxx.pdf list as shown.

The file is located on the Datafox DVD and for download on the homepage. Please also note the instructions in each chapter in the manual. The updates are available on our website under www.datafox.de download.

Inhalt

1.	Software	1
1.1.	System Requirements:	1
1.2.	Installation	1
1.2.1.	Installation of DatafoxStudioIV	1
1.2.2.	Using DatafoxStudioIV.exe and DFComDLL.dll	2
2.	Kompatibilität Compatibility	3
2.1.	Firmware File Archive (*.dfz)	3
2.2.	Datafox Devices and Device Firmware	3
2.3.	Device Firmware and Device Setup	3
2.4.	Device Firmware and Communications DLL	4
2.5.	Communications DLL and DatafoxStudioIV	4
2.6.	DatafoxStudioIV and Device Setup	4
2.7.	Update / Downgrade	5
3.	User Interface	6
3.1.	Arrangement of Windows	6
3.2.	Operation	7
4.	Functions in DatafoxStudioIV	8
4.1.	Setup	8
4.1.1.	Editing Setup	8
4.1.2.	Converting	8
4.1.3.	Importing Lists	9
4.1.4.	Importing Access Control Lists	9
4.2.	Communication	10
4.2.1.	Setup	10
4.2.1.1.	Transferring Setup to the Device	10
4.2.1.2.	Reading Setup from the Device	10
4.2.2.	Loading Lists	11
4.2.3.	Reading Lists	11
4.2.4.	Loading Access Control Lists	11
4.2.5.	Reading Access Control Lists	11
4.2.6.	Importing and Loading Timeboy Lists	12
4.2.7.	Reading, Deleting Data	12
4.2.8.	Reading, Deleting, Viewing Data	13
4.2.9.	Deleting Data	14
4.2.10.	Reading Serial Number	14
4.2.11.	Setting Time	14
4.2.12.	Sending a Message	14
4.2.13.	Reading Global Variables	15
4.2.14.	Processing Batches	15
4.2.15.	Settings (Communication)	16
4.2.16.	Verschlüsselung der Kommunikation mit MasterIV Geräten	17
4.2.16.1.	Erstellung und Hinterlegen des Schlüssels im Gerät	17
4.2.16.2.	Hinterlegen des Passwortes im StudioIV	18
4.2.16.3.	Übergabe des Schlüssels in die DFComDLL	18
4.2.16.4.	Löschen des Kommunikationsschlüssels	19
4.3.	Configuration	19

4.3.1.	Transferring Firmware	20
4.3.1.1.	Changing Additional Options	21
4.3.2.	Language Table for Device, Device Texts	22
4.3.2.1.	Editing File for Language Table	22
4.3.2.2.	Transferring File for Language Table	23
4.3.3.	Color Selection for Devices with Color Display	24
4.3.3.1.	Editing Color Data of Firmware	24
4.3.3.2.	Transferring Color Data of Firmware	24
4.3.4.	Configuring Data Storage	25
4.3.5.	Configuring USB Stick	25
4.3.5.1.	USB Host at Master IV	26
4.3.5.2.	Creating Directory Structure and Password at USB Stick	27
4.3.5.3.	Changing Password for Communication	29
4.3.5.4.	Transfer from Master IV to USB Stick	30
4.3.5.5.	Updating the USB Host	30
4.3.6.	System Variables of Signal Processing	31
4.3.7.	System Variables HTTP / GPRS	34
4.3.7.1.	Sending Data Records with HTTP via Mobile Communications	34
4.3.7.2.	HTTP response, and optional parameters	35
4.3.7.3.	Sending Data Records with HTTP via LAN / WLAN	38
4.3.7.4.	Encryption of Data Fields for HTTP (GPRS) Sending	40
4.3.8.	System Variables Active Connection & Configuration	43
4.3.8.1.	Description	43
4.3.8.2.	Configuration of an Active Connection	44
4.3.8.3.	Maintenance via Active Connection	46
4.3.9.	Device Configuration BIOS	47
4.3.9.1.	Accessing Device BIOS	48
4.4.	Extras	50
4.4.1.	Device Maintenance via Modem Connection	50
4.4.2.	Retrieving Statuses of Access Modules	51
4.4.3.	Recovering Data Records	51
4.4.4.	Update Biokey3000/4000 Fingerprint Module	52
4.4.5.	Backup/Restore of Finger Templates	53
4.4.6.	Reading System Logs	54
4.4.7.	Reading Memory	54
4.4.8.	Configuring Data Storage	55
4.5.	Office Connect	56
4.5.1.	Allgemeine Informationen	56
4.5.2.	Bedienung	56
4.5.2.1.	Exportaufgaben erstellen	57
4.5.2.2.	Export starten	57
4.5.2.3.	GPS-Daten extrahieren und Anzeigen	58
4.5.2.4.	Erweiterung für Office-Connect	59
4.6.	Help	60
4.6.1.	Information about DatafoxStudioIV	60
5.	Setup Structure	61
5.1.	Global Settings	63
5.1.1.	Basic Settings	63
5.1.2.	Global Variables	65
5.1.3.	Transponder	65
5.1.4.	Fingerprint	66
5.1.5.	Timeboy	67
5.1.6.	Summer-/Wintertime	67

5.1.7.	Operation Mode	68
5.1.8.	Operating Type	69
5.2.	Definition of Data Structures	70
5.3.	Definition of Master Data Structure.....	71
5.4.	Recording Procedures of Operation	72
5.4.1.	Structure of Operation	72
5.4.2.	Starting an Input Sequence	72
5.4.3.	Configuring an Input Sequence	74
5.5.	Recording Procedures of Signal Processing.....	75
5.5.1.	Structure of Signal Processing	75
5.5.1.1.	Digital Inputs	76
5.5.1.2.	Analog Inputs	78
5.5.1.3.	Timeboy Events	80
5.5.1.4.	GPRS-Alive.....	81
5.5.1.5.	Timer Events.....	82
5.5.1.6.	GPS Events	83
5.6.	Recording Procedure of Access Control.....	84
5.7.	Input field-functions in the device setup.....	85
5.7.1.	Normal (value transfer from RFID card, barcode etc...)	87
5.7.1.1.	Default value are displayed before an input value	88
5.7.1.2.	The transponder signalling readiness	88
5.7.2.	Current Date / Time	89
5.7.3.	Constant	89
5.7.4.	Select from a list	90
5.7.4.1.	Choose from a list.....	90
5.7.4.2.	List selection	91
5.7.5.	Write in a list	92
5.7.6.	Confirmation	93
5.7.7.	Global variables in field copy	93
5.7.8.	Field in a Global variable copy.....	94
5.7.9.	To perform math, logic, or format operations.....	94
5.7.10.	Write value on RFID-Tag	96
5.7.11.	Serial number assume	96
5.7.12.	Digital state assume	97
5.7.13.	Counter assume	97
5.7.14.	Analog measurement assume	97
5.7.14.1.	Integralfunction for analog measurement	98
5.7.15.	Analog limits assume	99
5.7.16.	GPRS Alivecounter assume	99
5.7.17.	Firmware version assume (xx.xx.xx.xx)	99
5.7.18.	Status der Summer-/Winter time assume	100
5.7.19.	GPS data assume.....	100
5.7.20.	GPS - data assume (variable selection)	101
5.7.21.	GPS - assume distance covered	102
5.7.22.	Mobile phone parameters assume	102
5.7.23.	Relay switch.....	103
5.7.24.	Buzzer switch.....	103
5.7.25.	Data on Card (Fieldfunction).....	103
5.7.26.	Send SMS.....	104
5.7.27.	Server status assume (online/offline)	104
5.7.28.	Switch communication	105
5.7.29.	Access check with GV	106
5.7.30.	Access: ZM (ZutrittsMaster = access master) assume	106

5.7.31.	Access: TM(TürModu = door module IDI) assume	106
5.7.32.	Access: ID number assume	107
5.7.33.	Access: State assume	107
5.7.34.	Fingerprint: Scannen	107
5.7.35.	Fingerprint: Perform identification	108
5.7.36.	Fingerprint: fingerprint to train	108
5.7.37.	Fingerprint: Finger template delete	109
5.7.38.	Fingerprint: Perform verification	110
5.7.39.	Fingerprint: Read Finger template from ID card	110
5.8.	Data on Card	111
5.8.1.	Allgemeine Informationen	111
5.8.2.	Einrichtung Data on Card	112
5.9.	Extended Jump Function in Input Sequences	116
5.9.1.	Behavior after Entering Field Content	117
5.9.2.	Behavior after Canceling an Input by ESC	117
5.9.3.	Behavior at Leaving the Menu	118
6.	The RFID Technology (Transponder)	119
6.1.	General Information on RFID	119
6.2.	RFID Methods Supported by Datafox	120
6.3.	Most Important RFID Reading Methods.....	123
6.3.1.	125 kHz RFID Reader	124
6.3.1.1.	Unique	124
6.3.1.2.	Hitag1	124
6.3.1.3.	Hitag2	125
6.3.1.4.	Titan.....	126
6.3.2.	Legic Transponder Technology	127
6.3.2.1.	Overview Prime and Advant	127
6.3.2.2.	Important Settings in DatafoxStudioIV	127
6.3.3.	HID-ProxKey und HID-iClass.....	131
6.3.4.	13,56MHz RFID Reader (ISO14443 u. ISO15693).....	133
6.3.4.1.	ISO 14443A - Mifare Familie	134
6.3.4.1.1.	Mifare Mini	134
6.3.4.1.2.	Mifare Classic	134
6.3.4.1.3.	Mifare Plus	134
6.3.4.1.4.	Mifare Ultralight.....	135
6.3.4.1.5.	Mifare DESFire	136
6.3.4.1.6.	ISO 14443B – Calypso, CEPAS und Moneo	139
6.3.4.1.7.	ISO 14443-2 iClass, Picopass	139
6.3.4.1.8.	ISO 14443-3 SRX von ST Microelectronics.....	139
6.3.4.1.9.	ISO 15693 – iCode, Tag-it, MyD.....	139
6.3.4.1.10.	ISO 14443/15693 read serial number.....	140
7.	Tips and Tricks	141
7.1.	Adding leading zeros at ID.....	141
8.	Index	142

1. Software

DatafoxStudioIV is required for creating and modifying the device setups. Setups are saved device-specifically. Thus, switching the device type before opening a setup is not necessary. But it is possible to convert a setup from one device type to another. Mandatory, type-specific modifications are made automatically by DatafoxStudioIV.

1.1. System Requirements:

- PC with Microsoft Windows XP or higher
- at least 200 MB free hard disk space

1.2. Installation

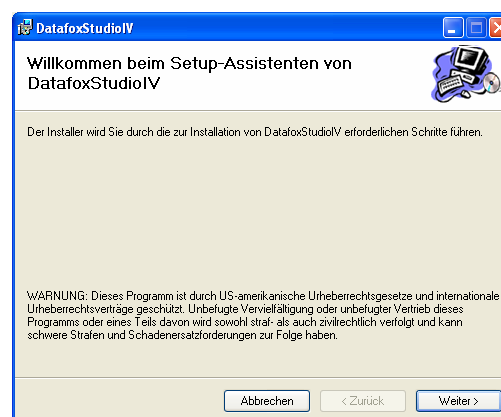
1.2.1. Installation of DatafoxStudioIV



If it is necessary to install DatafoxStudio due to user rights, an installation version is available. In order to perform a complete installation, you must possess the necessary user rights for the server/PC. If you do not possess these rights, contact your administrator. To start the installation, execute the application [DatafoxStudioIVSetup.msi](#). The installation comprises the following 5 steps.

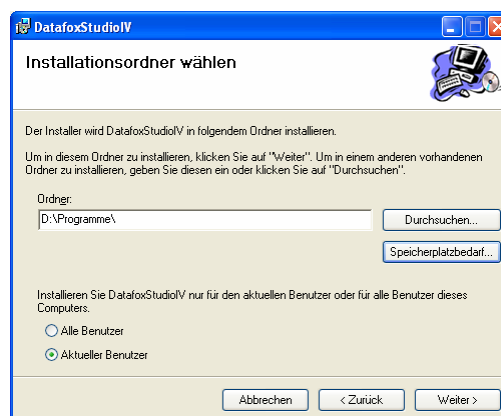
Step 1

Start installation, click on Next.



Step 2

Specify the directory where to install DatafoxStudioIV.



Step 3

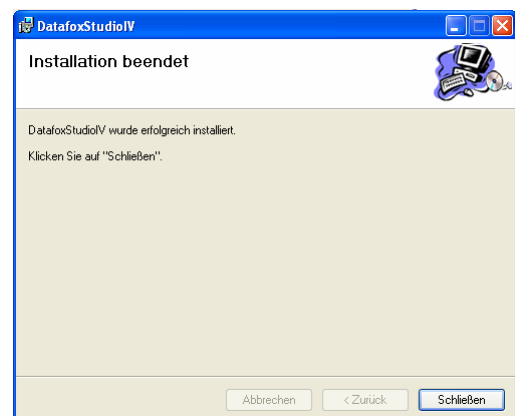
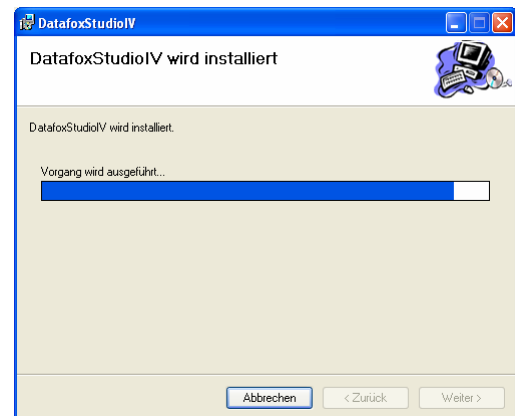
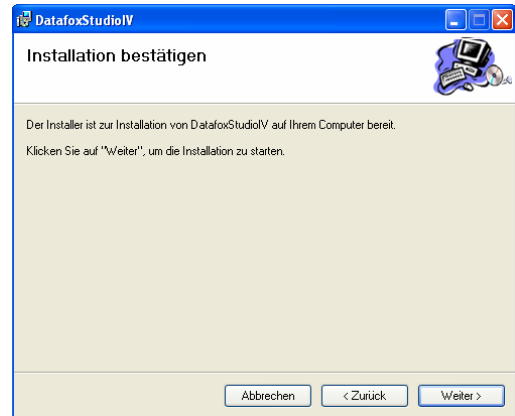
Continue installation by clicking on "Next".

Step 4

DatafoxStudioIV is installed.
The bargraph displays the installation progress.

Step 5

The installation is complete.
Close the dialog window.



1.2.2. Using DatafoxStudioIV.exe and DFComDLL.dll

An installation may be necessary if the resources required are not available at the PC. The setup and communication program comprises only the files DatafoxStudioIV.exe and DFComDLL.dll. An installation is not necessary. You can directly execute the .exe file and work with it. Copy the files "DatafoxStudioIV.exe" and "DFComDLL.dll" to the desired directory and, if desired, create a shortcut to DatafoxStudioIV.exe in the program menu or on the desktop.



Note:

It is recommended to always use the latest version of DatafoxStudioIV and the DFComDLL.dll.

2. Kompatibilität Compatibility

The compatibility must be observed urgently between:

- Datafox devices and the device firmware
- Device firmware and device setup
- Device firmware and communication DLL
- Communication DLL and DatafoxStudioIV
- DatafoxStudioIV and device setup

2.1. Firmware File Archive (*.dfz)

Description

Device files (*.hex) of the MasterIV devices are delivered in a common firmware file archive. It has the file extension DFZ (stands for Datafox Zip). Now simply the firmware file archives (*.dfz) are indicated instead of the device files (*.hex). This applies to the DatafoxStudioIV and the DLL. The indication of device files (*.hex) is still possible.

Function of the Archive

The transfer routine of the device file selects the right file from the firmware file archive on the basis of the hardware options available in the device. Thus, it is guaranteed that all hardware components available in the device are supported by the corresponding firmware.

Manual Selection of a File

If you do not want to integrate the archive in your installation, you have the possibility to add single device files from the archive to the installation.

The file format of the firmware file archive is ZIP. Hence, you can open the archive with every standard ZIP-program. Via the entry "Open With" in the context menu you can select an appropriate program for opening the file. If necessary, you can call up a program combined with this file format to open the file by renaming the file from DFZ to ZIP.

In the archive you find a file named "Inhalt.pdf"; it contains information which file (*.hex) of the archive matches your device. Extract the desired device file (*.hex) and rename it if necessary. A renaming of a file is possible at any time, because all information are in the file itself.

You can state the device file extracted before as device file in DatafoxStudioIV and at calling the DLL function. It is still tested if the file can be loaded into the chosen device before the transfer takes place.

2.2. Datafox Devices and Device Firmware

Each Datafox device has an electronic flat module. The module has specific hardware equipment concerning the options (e.g. mobile radio, WLAN, fingerprint,...). Due to technical conditions, different options are mutually exclusive. Currently, not all hardware options can be supported in one firmware file due to limited program memory. This means that each device with specific hardware options needs a proper firmware to support the hardware options by the software.



Caution:

Hardware generation V 3 is supported from version 04.02.00.x onwards. The DatafoxStudioIV is compatible up to and including firmware version 04.01.x.y. Older versions 04.00.x.y are not supported any more.

2.3. Device Firmware and Device Setup

The firmware (operating system) of the device and the device setup (*.aes data file = application program) form a unit. By the device setup, the runtime behavior of the device (the firmware) is determined. This means the response of the device to input events by the user or the environment (e.g. digital inputs). In principle, only those functions of the device are executed that are supported by the firmware and defined via the setup. Prior to the productive commencement, you should there-

fore test each setup with the corresponding device or on a device with the same hardware options and firmware.

2.4. Device Firmware and Communications DLL

A firmware supports certain functions, dependent on the hardware options. The communication DLL is the interface between the firmware and the DatafoxStudioIV or your processing software. Therefore, the firmware must always have the same or a lower version number as the communication DLL.



Note:

If your application uses a newer version of the DLL than the firmware does, you can only use functions that are supported by the firmware. Otherwise, you will receive an error message (e.g. function not supported) which has to be analyzed.

2.5. Communications DLL and DatafoxStudioIV



Note:

The DatafoxStudioIV and the communication DLL are developed and released as a bundle. Therefore, they have to be used as a bundle. A newer version of DatafoxStudioIV does not work with an older DLL.

2.6. DatafoxStudioIV and Device Setup

With the DatafoxStudioIV, you create a device setup (application program) for the Datafox device. That means that in the setup only those functions were defined which were available in the DatafoxStudioIV version at the time of the setup creation. The DatafoxStudioIV you use for opening a device setup may thus only be newer but never older than the DatafoxStudioIV version you used to create the device setup.



Note:

The updates are always available for download on our homepage www.datafox.de.



Caution:

When new devices are delivered, the latest firmware is loaded on the devices. If you wish to work with an older firmware version, please perform a downgrade. Please observe the compatibility notes in the release notes of the respective firmware version.

The data file <Device name>, Software Versionen Stand <version number>.pdf shows which functions are supported by which software release.

You will find the file on the product CD. Please also follow the instructions given in the chapters of the manual.

2.7. Update / Downgrade

A firmware update or downgrade is a very sensitive process. Possibly, a reset of the main communication to RS232 may occur. In any case, consider the information regarding the compatibility in the software version list.

Firmware Update



Caution:

Before starting a firmware update, please check on the basis of the software version list whether there are any version dependencies that must be observed.

For example: when changing from Version 04.00.xx to version 04.01.xx, at least version 04.00.23.769 or higher must be present in order to run the update to version 04.01.xx successfully.

Firmware Downgrade

A firmware downgrade is not recommended.

We are constantly working towards improving the software/firmware; all functionalities are still included in new versions. New software always offers better functionalities and possible bugs are fixed.

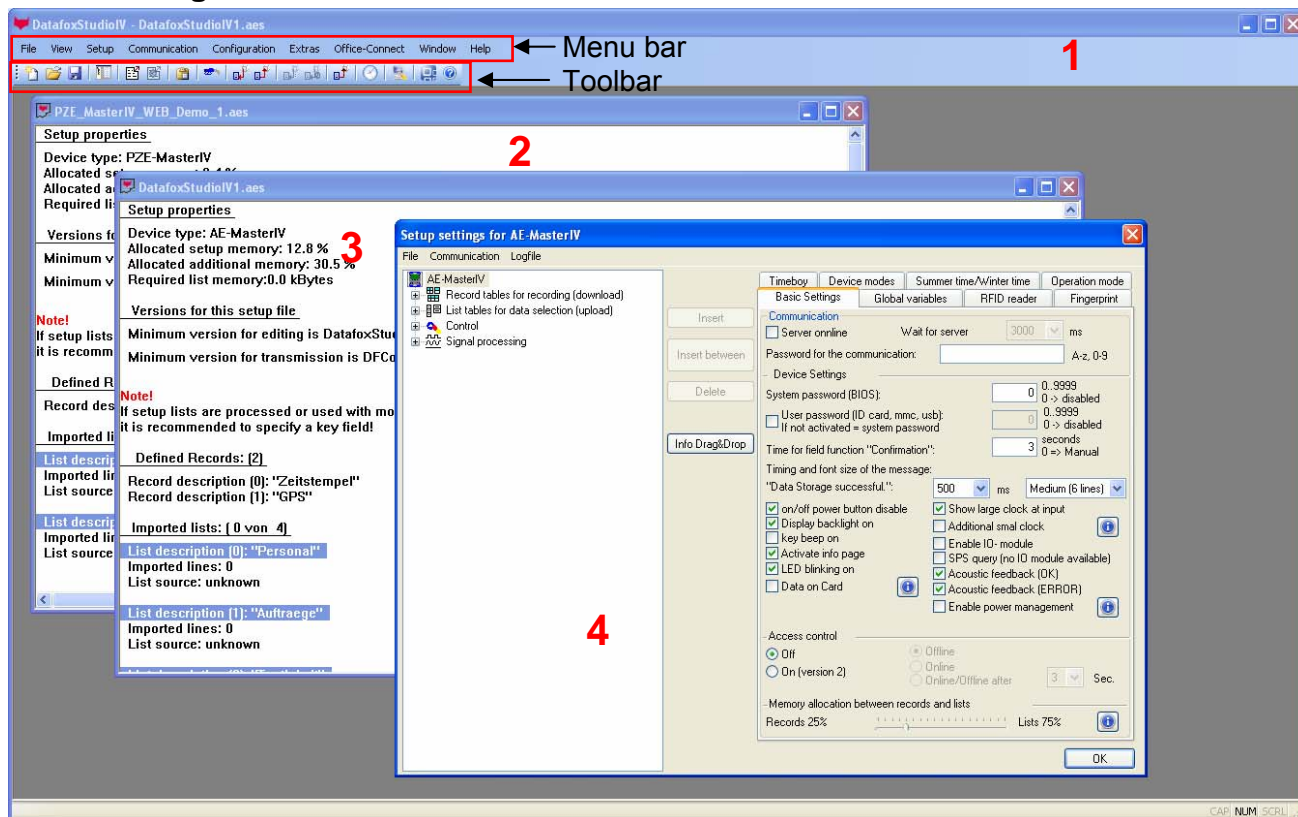



Caution:

When performing a firmware downgrade the firmware has to be transmitted to the device twice. This has technical reasons. Errors shown on the display of the device after the first transfer can be ignored.

3. User Interface

3.1. Arrangement of Windows



At DatafoxStudioIV, several windows (2 and 3) of a device setup can be opened simultaneously. But only one window can be edited. You can open a setup for editing (4) via menu entry "Setup ->Edit" or the symbol .

The mask (4) can also be opened by double-clicking on the window. For more information see chapter "[Edit Setup](#)".

3.2. Operation

After starting DatafoxStudioIV, an empty window with a menu and a toolbar is displayed. This menu provides all functions which can be executed on a device without a device setup. If you create or open a device setup, the menu is extended by functions for a device setup.

We recommend the following procedure when working with DatafoxStudioIV:

- Opening a setup (each setup is tied to a device type).
- Setting the communication interface to a device (how the device is accessible).
- Selecting the desired function via the menu or the toolbar (not all menu entries are also available in the toolbar).
- Configuring connection variables for transfer with http via LAN or GPRS (specifying additional parameters at the configuration file *.ini).
- Executing the function selected (editing data or transfer data via DFComDLL).



Caution:

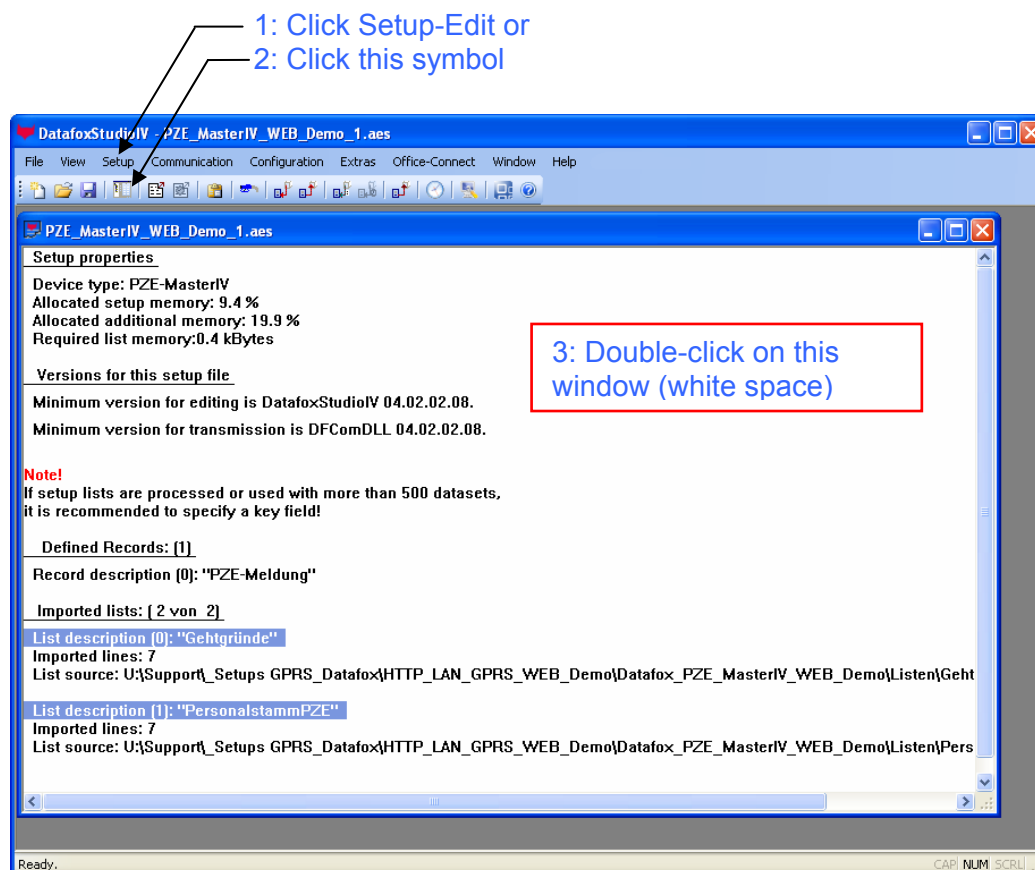
Before each communication with a device, make sure that you address the right device.

4. Functions in DatafoxStudioIV

4.1. Setup

4.1.1. Editing Setup

In order to edit a setup, a device setup must be open.
There are **three** possibilities to open a setup for editing.

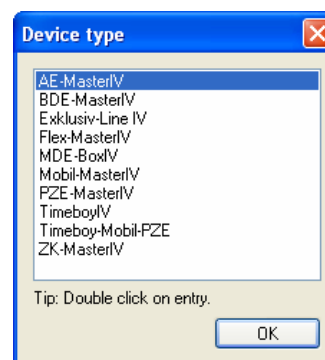


4.1.2. Converting

A setup which you have created for another device can be converted for a different device type via the function "Converting".


This function saves you time when developing device setups. Prerequisite is an opened device setup.

Click the tab Setup-Convert. This window opens.
Here you can select for which Datafox device the setup is to be converted.

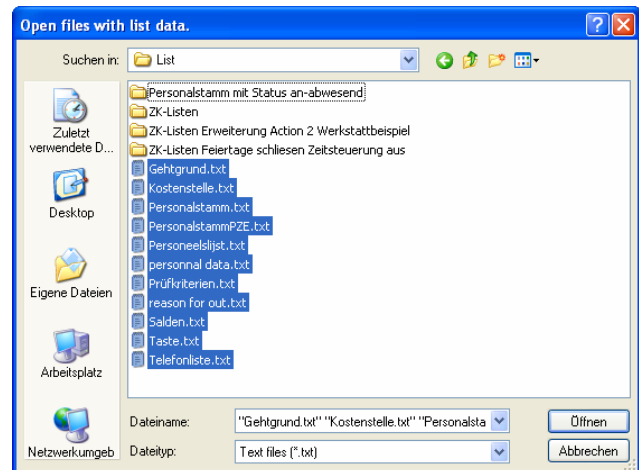


4.1.3. Importing Lists

If lists are defined and used in a device setup, they can also be transferred to the Datafox device. For this purpose, it is necessary to import the lists into the setup.


Click the symbol  to import an existing list.
You can also open the import dialog via Setup – „Import Lists“.

Specify where the lists are saved and select the lists to be imported. Several lists can be imported simultaneously.

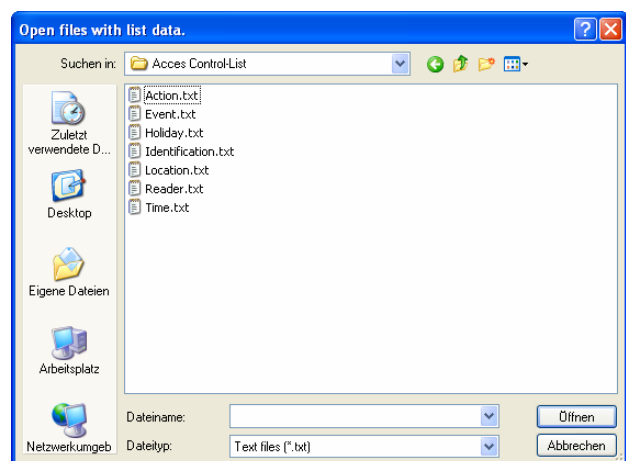


4.1.4. Importing Access Control Lists

If access control lists are defined and used in a device setup, they can also be transferred to the Datafox device. For this purpose, it is necessary to import the lists into the setup.

Click the symbol  to import an existing access control list.
You can also open the import dialog via Setup – „Import Access Control Lists“.

Specify where the lists are saved and select the lists to be imported. Several lists can be imported simultaneously.



4.2. Communication

This chapter describes all settings for transfers to a Datafox device.

4.2.1. Setup

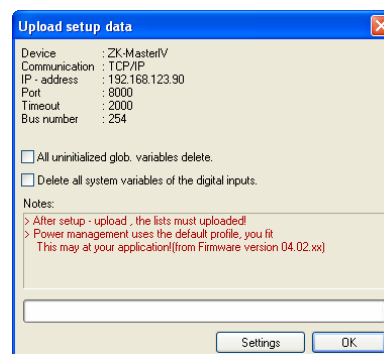
4.2.1.1. Transferring Setup to the Device

When you have created a setup, you can transfer it to the Datafox device as follows.

Click on this sym-  bol or use the tab "Communication" – „Upload setup" in order to transfer the setup.

The following dialog window opens:

Click "OK" to start the transfer.



4.2.1.2. Reading Setup from the Device


It is possible to read a setup from a device.

If a saved setup is lost, this is a good possibility to recover the setup.



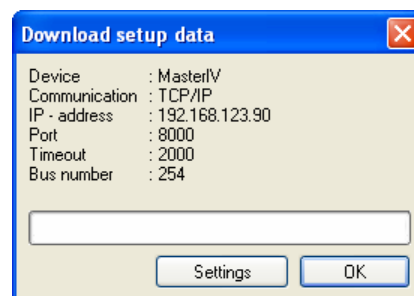
Note:

First, open a new setup and save it under a name of your choice. The read setup is saved in the currently edited setup.

Click on this sym-  bol or use the tab "Communication" – „Read setup" in order to transfer the setup.

The following dialog window opens:

Click "OK" to start the transfer.



4.2.2. Loading Lists

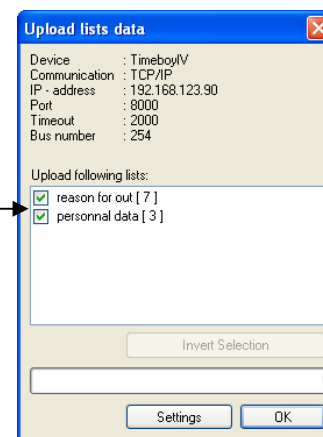
If lists are defined and imported into a device setup, they must be transferred to the device in order to use them there.

Click on this symbol  or use the tab "Communication" – „Load lists" in order to transfer the lists to the device.

The following dialog window opens:

By activating the check-boxes you can specify which lists are to be transferred.

Click "OK" to start the transfer.



4.2.3. Reading Lists

Prerequisite is an accessible device with a device setup and available list data.

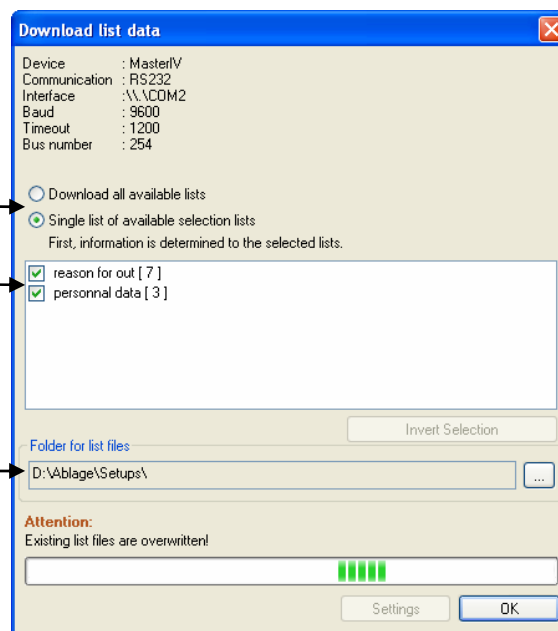
The dialog window opens via the tab "Communication" – „Download lists".

Specify whether all lists are to be read or you want to select certain lists.

By activating the checkboxes you can specify which lists are to be read from the device.

Specify where the lists are to be stored.

! If lists with the same name are in this folder, they are overwritten.



4.2.4. Loading Access Control Lists

Prerequisite is an opened device setup where the function Access Control is activated. Proceed as described in the section "Loading Lists".

4.2.5. Reading Access Control Lists

Prerequisite is an accessible device with a device setup where the function Access Control is activated. Proceed as described in the section "Reading Lists".

4.2.6. Importing and Loading Timeboy Lists

Prerequisite is an accessible device with a device setup which is configured for a Timeboy connection. After calling the function, a dialog opens for configuring the import and transfer process:

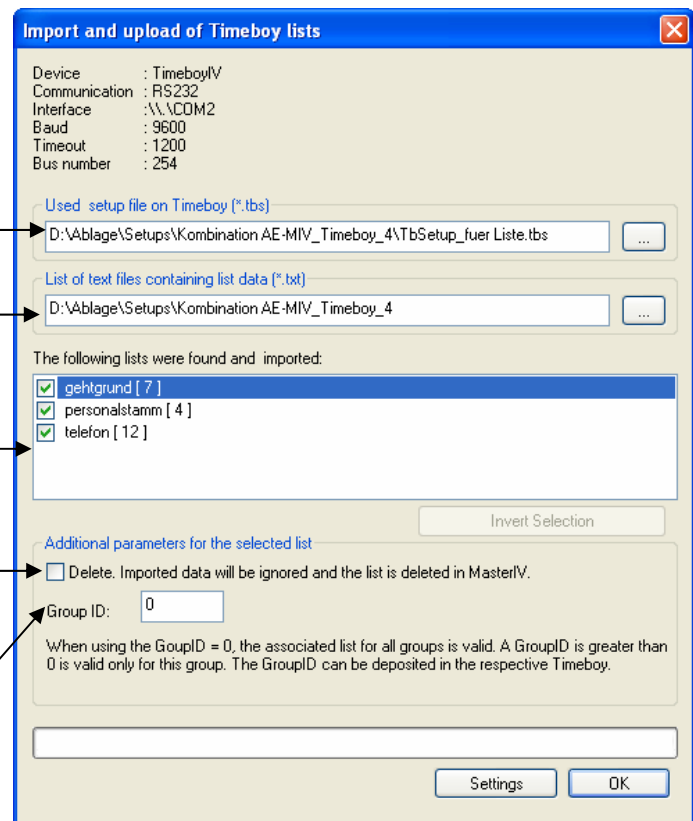
Select the Timeboy device setup containing the lists to be transferred.

Specify the directory where the lists (*.txt files) are to be saved.

Mark the lists you want to transfer.

Specify whether the lists in the device are to be deleted. If this option is set, no lists are transferred. The marked lists only determine which lists are to be deleted in the device.

Enter the GroupID (0 = list valid for all Timeboys, Value > 0 defines a group of Timeboys with this GroupID).



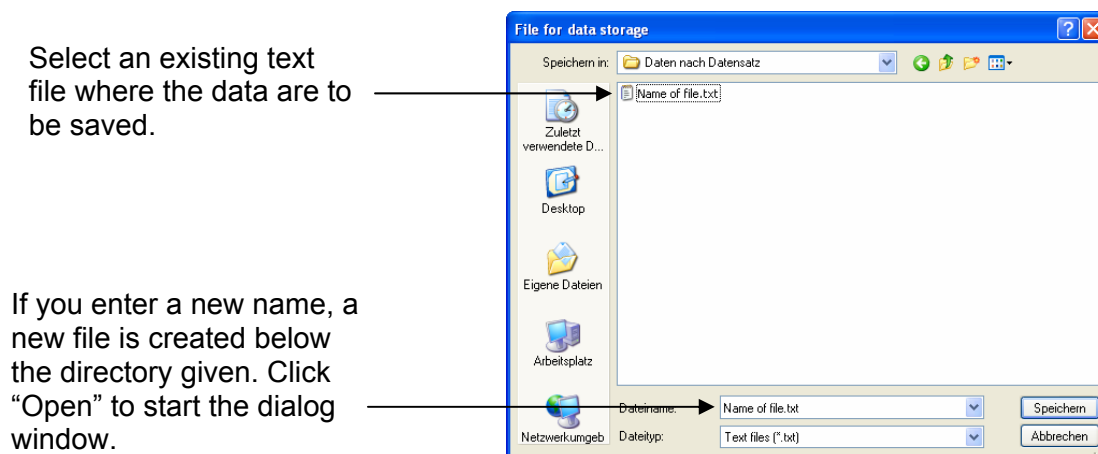
Confirm all settings with "OK" and start the process. The dialog closes automatically after execution.

4.2.7. Reading, Deleting Data

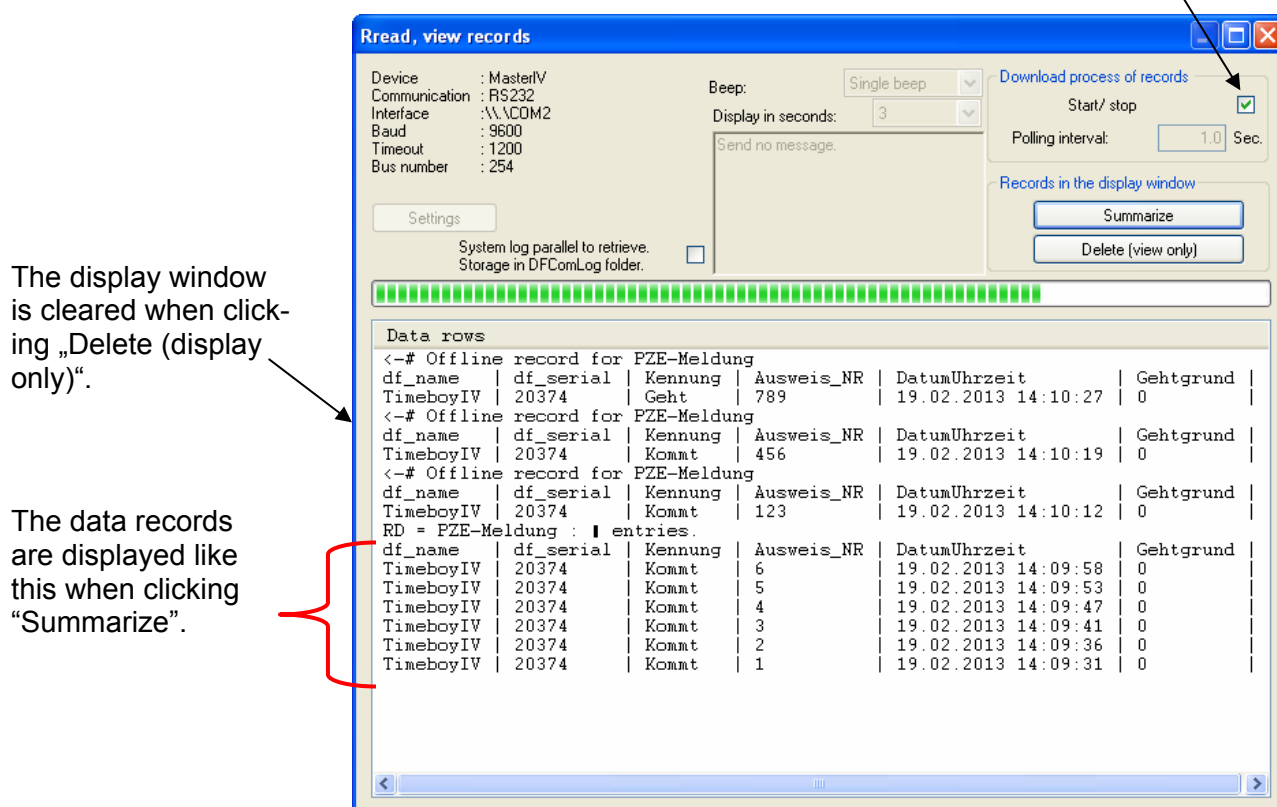
Prerequisite is an accessible device with a device setup. In the dialog for opening a file, select an existing text file (*.txt) or enter a new file name. Confirm your entries with "Open". In the following dialog, specify whether the data are to be read once or the device is to be polled. If required, specify the polling frequency. Start operation with "OK".

4.2.8. Reading, Deleting, Viewing Data

In order to read data records, they must be stored in the Datafox device.
Start this function via the tab "Communication" – "Read, delete, view data".
The following dialog window opens:



By activating this checkbox, transfer is started.



4.2.9. Deleting Data

Prerequisite is an accessible device with a device setup which contains data records.



Caution:

Before executing this function, make sure that you address the right device. Once the data are deleted, they cannot be restored.

4.2.10. Reading Serial Number

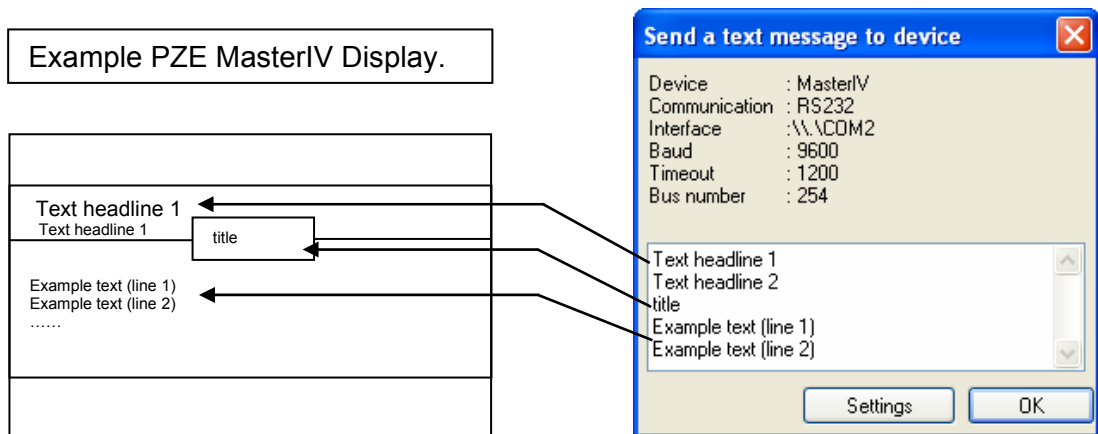
Prerequisite is an accessible device. After successful execution of the function, the serial number of the device is displayed in a dialog window.

4.2.11. Setting Time

Prerequisite is an accessible device. After successful execution of the function, date and time of the device has been synchronized with the system time of the PC from which the function has been executed.

4.2.12. Sending a Message

Prerequisite is an accessible device with display. Enter the message in the text field and confirm your entry with "OK". The display can show 23 message lines at most with max. 250 characters altogether.



Note:



Because the font used is no proportional font, it is not possible to say how many characters can be shown per line.

Each character is displayed with an individual width.

For example, "iiiiii" needs less space than "mmmmm".

If you use this function via a server application, please test previously whether the text length is not exceeded on a line.

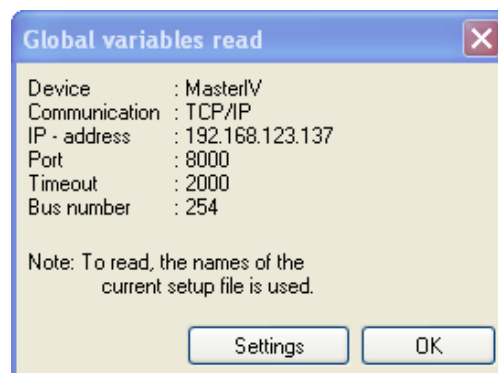
If the text is too long, the rest is truncated.

4.2.13. Reading Global Variables

If you use GVs in a device setup, it is possible to display them. In order to ensure that all global variables can be read, first read the setup of the device. Only this way you ensure that all available global variables are read.

Call this function via the tab "Communication" – "Read global variables".

Click OK to read the GVs.



In this example 2 GVs are displayed.



4.2.14. Processing Batches

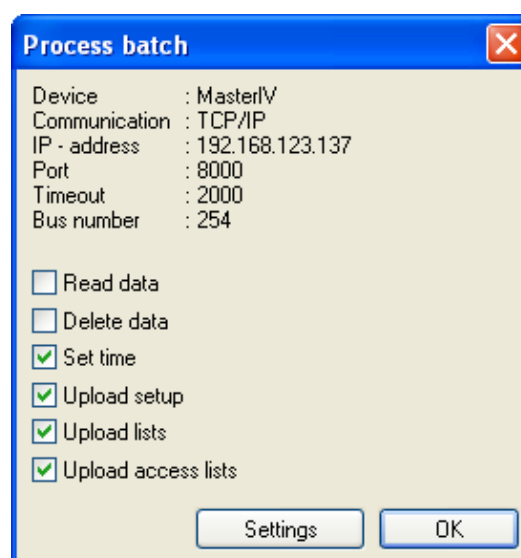
Prerequisite is an accessible device. Activate the function you want to execute on the device.



Note:

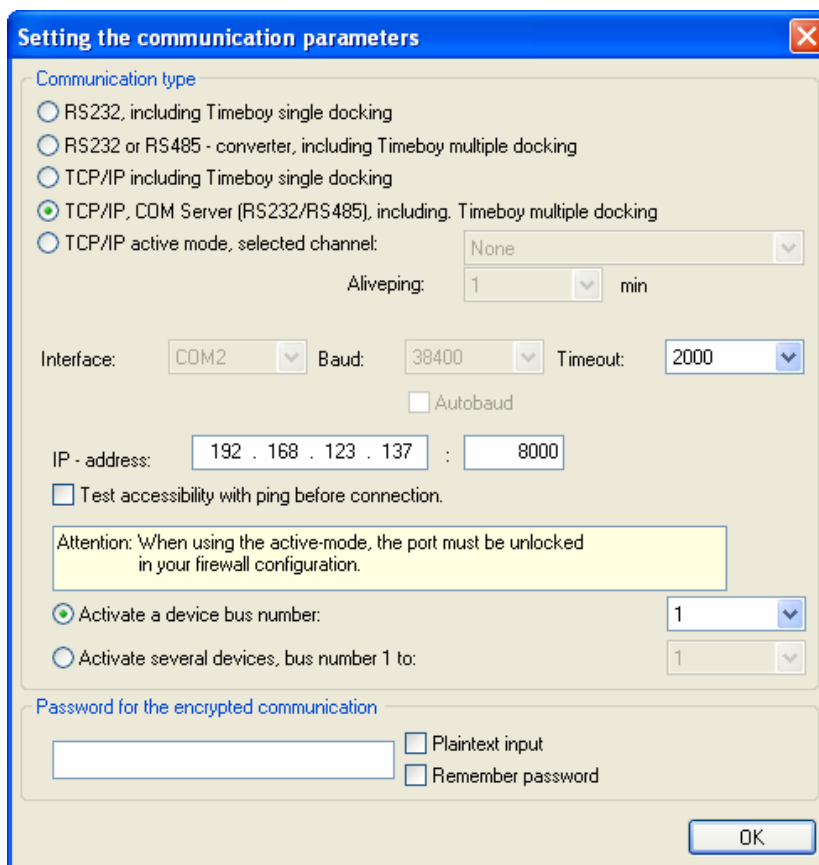
Note that the function "Load setup" is executed on the currently opened setup. If you also want to execute the function "Upload lists", you must ensure that all lists have been imported. Same applies for the option "Upload access lists".

By activating the checkboxes, you determine which functions are to be executed.



4.2.15. Settings (Communication)

Specify via which interface you want to communicate with a device. For this purpose, you have to know how the device is set. For information concerning the device configuration see the device BIOS. Depending on the interface selected (RS232, TCP/IP,...), further parameters are activated. Make the all settings required and confirm your entries with OK.



Note:

If you select the option "TCP/IP active connection", specify timeout and port. Confirm your entries with OK. With the confirmation (the dialog window is closed), the active connection on the PC is started. Now, a device can connect with the PC, provided that the device is configured for an active connection with the PC. Wait a few minutes before opening the dialog "Settings" again. Then, you can select the active channel and communicate with the device via this active channel.



The time the device needs to connect with the PC depends on the configuration of the active connection of the device. An influencing factor is the number of connection attempts and the timeout between these attempts.

Example:

A device tries connecting with a PC 3 times and then pauses for a minute. In the worst case, you must wait 1 minute before the active channel is shown in the Settings dialog window.

4.2.16. Verschlüsselung der Kommunikation mit MasterIV Geräten

Datafox Geräte sind nun in der Lage, die Kommunikation per AES zu verschlüsseln.

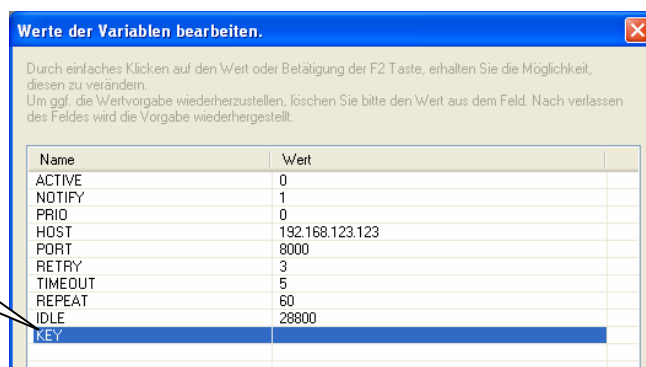
Es gibt damit nur 3 Arten der Kommunikation:

1. Unverschlüsselte Kommunikation
2. Verschlüsseln mit Datafox-Key
3. Verschlüsseln mit User-Key

4.2.16.1. Erstellung und Hinterlegen des Schlüssels im Gerät

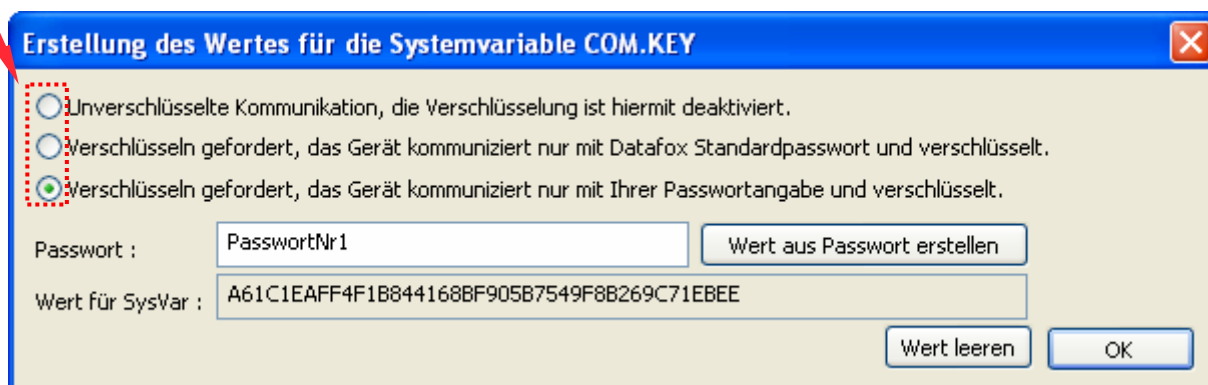
Unter dem Menüpunkt Konfiguration „Systemvariablen Aktive Verbindung“ öffnen Sie bitte die Konfigurationsdatei (z.B.: active.ini) zum Bearbeiten.

Durch einen Mausklick auf der Zeile KEY, öffnet sich das Fenster für die Erstellung des Schlüssels.



Name	Wert
ACTIVE	0
NOTIFY	1
PRIQ	0
HOST	192.168.123.123
PORT	8000
RETRY	3
TIMEOUT	5
REPEAT	60
IDLE	28800
KEY	

Hier können Sie zwischen den Kommunikationsvarianten wählen



Erstellung des Wertes für die Systemvariable COM.KEY

☐ Unverschlüsselte Kommunikation, die Verschlüsselung ist hiermit deaktiviert.
☐ Verschlüsseln gefordert, das Gerät kommuniziert nur mit Datafox Standardpasswort und verschlüsselt.
☒ Verschlüsseln gefordert, das Gerät kommuniziert nur mit Ihrer Passwortangabe und verschlüsselt.

Passwort :

Wert für SysVar :

Möchten Sie, dass die Kommunikation mit einem eigenen hinterlegten Passwort verschlüsselt wird, geben Sie ein Passwort ein und klicken auf den Button „Wert aus Passwort erstellen“.

Es wird nun ein Kommunikationsschlüssel erstellt. Schließen Sie die Eingabe mit „OK“ ab. Nach der Erstellung eines Schlüssels und Übertragung der Datei active.ini, wird nur noch unter Angabe des Passwortes eine Kommunikation zum Gerät zugelassen.

4.2.16.2. Hinterlegen des Passwortes im StudioIV

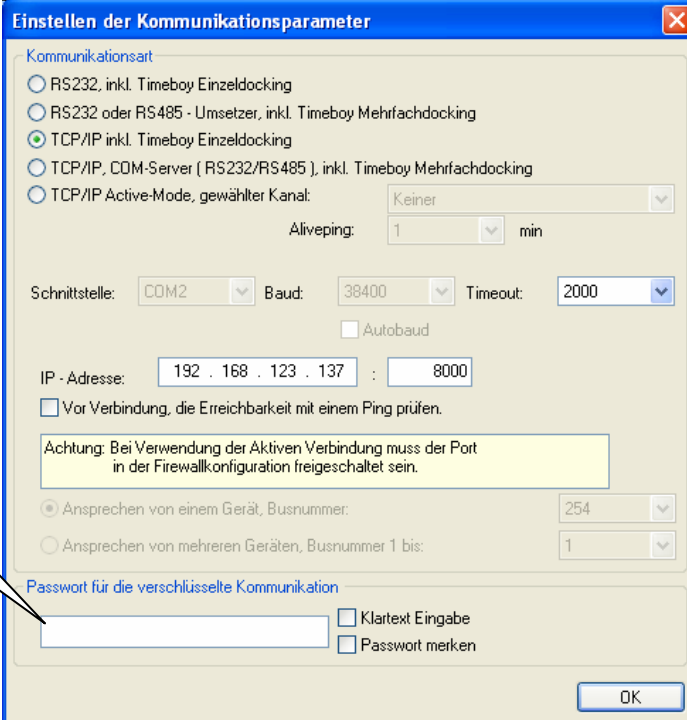
Ist bei einem Gerät ein Kommunikationsschlüssel hinterlegt, so muss das Passwort im StudioIV auch angegeben werden, da sonst keine Kommunikation mit dem Datafox Gerät möglich ist.

Unter dem Menüpunkt „Einstellungen->Kommunikation“ kann das Passwort hinterlegt werden.

Das Passwort gilt für alle Kommunikationsarten.

Geben Sie hier Ihr Passwort ein.

Klartext Eingabe funktioniert nur bei der ersten Eingabe. Wird dieses Fenster erneut geöffnet, ist der Klartext nicht mehr darstellbar.



4.2.16.3. Übergabe des Schlüssels in die DFComDLL

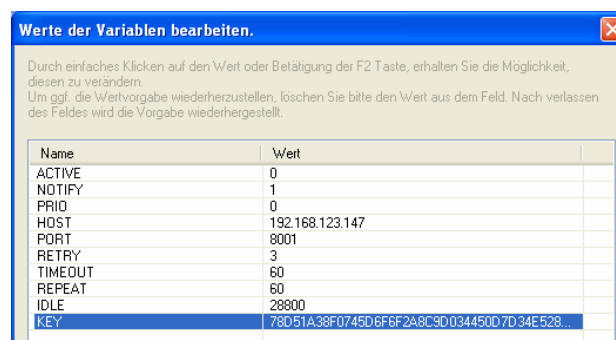
Der Schlüssel wird über die DLL-Routine DFSetCommunicationPassword gesetzt. Hierbei handelt es sich um den „echten“ Schlüssel (private Key), nicht der, der über das DatafoxStudioIV generiert wurde. Der Schlüssel wird im Klartext übergeben.

Mehr dazu finden Sie in der Dokumentation für die DFComDLL.

4.2.16.4. Löschen des Kommunikationsschlüssels

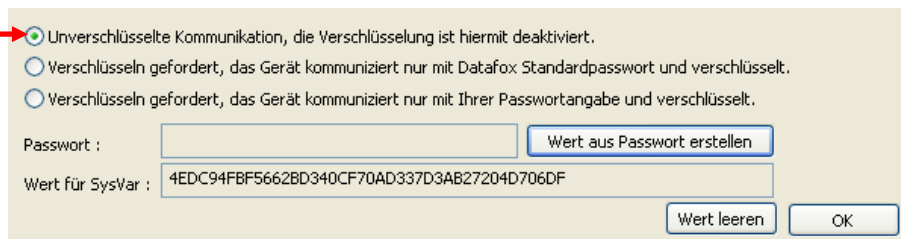
Ist ein Kommunikationsschlüssel hinterlegt, wird diese wie folgt gelöscht.

Klicken Sie auf KEY, um diesen zu bearbeiten.



Name	Wert
ACTIVE	0
NOTIFY	1
PRIQ	0
HOST	192.168.123.147
PORT	8001
RETRY	3
TIMEOUT	60
REPEAT	60
IDLE	28800
KEY	78D51A38F0745D6F6F2A8C9D034450D7D34E528...

Schalten Sie auf



☒ Unverschlüsselte Kommunikation, die Verschlüsselung ist hiermit deaktiviert.
☐ Verschlüsseln gefordert, das Gerät kommuniziert nur mit Datafox Standardpasswort und verschlüsselt.
☐ Verschlüsseln gefordert, das Gerät kommuniziert nur mit Ihrer Passwortangabe und verschlüsselt.

Passwort : Wert aus Passwort erstellen
 Wert für SysVar : Wert leeren OK

Klicken Sie anschließend auf „Wert leeren“.

Danach klicken Sie auf „Wert aus Passwort erstellen“. Dieser „leere Wert“ ist notwendig, um das Passwort im Gerät zu löschen.

Speichern Sie die Datei und übertragen diese an das Gerät.

Name	Wert
ACTIVE	0
NOTIFY	1
PRIQ	0
HOST	192.168.123.147
PORT	8001
RETRY	3
TIMEOUT	60
REPEAT	60
IDLE	28800
KEY	99F85A9C8FF989A26B96B217C33757B597EAFD7B...

Anschließend können Sie auch den „KEY“ ganz aus der .ini-Datei löschen.


Name	Wert
ACTIVE	0
NOTIFY	1
PRIQ	0
HOST	192.168.123.147
PORT	8001
RETRY	3
TIMEOUT	60
REPEAT	60
IDLE	28800
KEY	

4.3. Configuration

In this section, functions not involving a setup are described.

4.3.1. Transferring Firmware

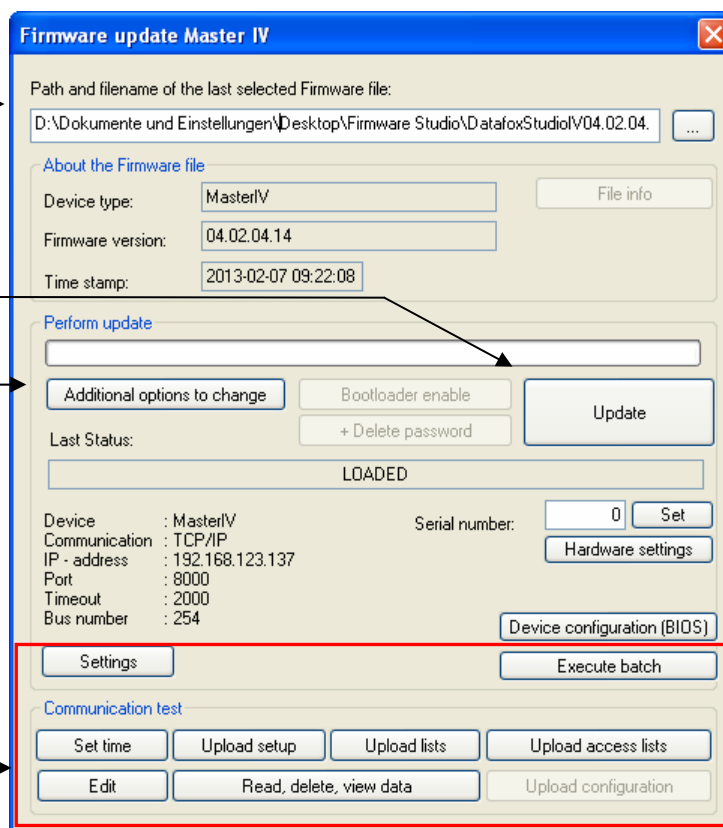
The firmware is the operating system of the device. Sometimes it may be necessary to load a new firmware on the Datafox device. Reasons could be debugging, new functions of the firmware or compatibility. The latest firmware is available on our website at any time.

This is window opened via Menu – Configuration – Load Firmware or with click on the symbol .

Specify which firmware is to be loaded.
After selecting the proper file, the update is started by clicking the button “Update”.

Before transferring the firmware, set the corresponding additional options. For more information see the next section.

All functions set here are described in the chapter “Communication”.



We recommend working only with *.dfz archives because in this case a compatibility check is carried out.



Note:

Please observe the compatibility notes of the devices and the firmware in the respective manuals.

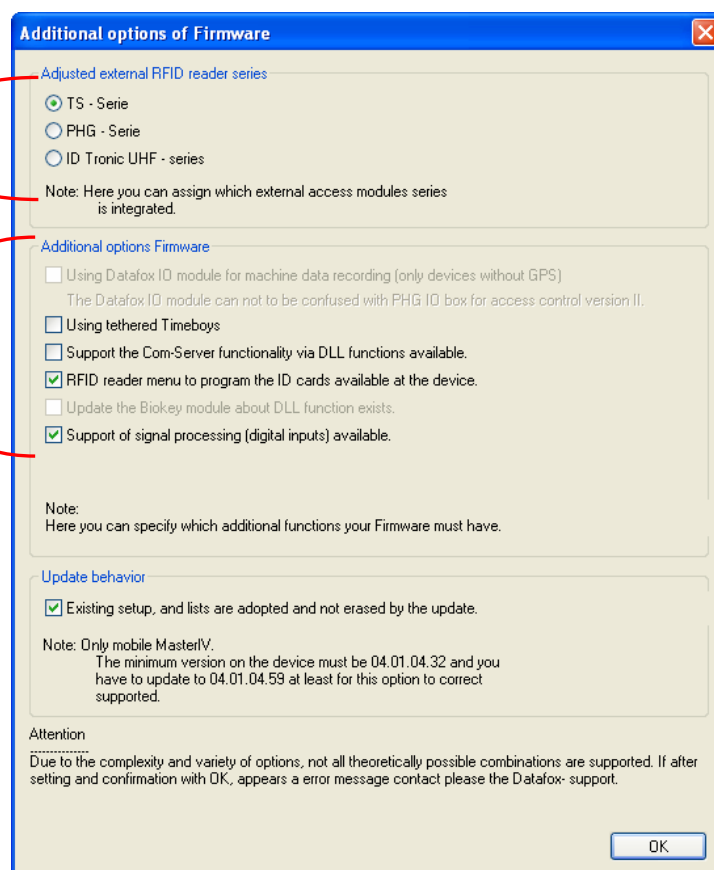
4.3.1.1. Changing Additional Options

Due to the numerous options of the Datafox devices, it is not possible to include all additional options in one firmware. By additionally selecting the settings it is determined which firmware is transferred to the device. The device independently selects the suitable firmware from the device file archive and an automatic compatibility check is carried out.

If, for example, the external reader is replaced, this must be set in the additional options and a firmware update must be run. Only then the reader is supported by the firmware.

If external readers are used, specify the reader type here. Depending on the reader type, a corresponding firmware is selected from the archive.

Further additional options to be set for transferring a firmware dependent on this.



Additional options of Firmware

Adjusted external RFID reader series

☒ TS - Serie
☐ PHG - Serie
☐ ID Tronic UHF - series

Note: Here you can assign which external access modules series is integrated.

Additional options Firmware

☐ Using Datafox IO module for machine data recording (only devices without GPS)
The Datafox IO module can not to be confused with PHG IO box for access control version II.

☐ Using tethered Timeboys

☐ Support the Com-Server functionality via DLL functions available.

☒ RFID reader menu to program the ID cards available at the device.

☐ Update the Biokey module about DLL function exists.

☒ Support of signal processing (digital inputs) available.

Note:
 Here you can specify which additional functions your Firmware must have.

Update behavior

☒ Existing setup, and lists are adopted and not erased by the update.

Note: Only mobile MasterIV.
 The minimum version on the device must be 04.01.04.32 and you have to update to 04.01.04.59 at least for this option to correct supported.

Attention
 Due to the complexity and variety of options, not all theoretically possible combinations are supported. If after setting and confirmation with OK, appears a error message contact please the Datafox- support.

OK



Note:

Before performing the firmware update, ensure that the additional options are properly set. In any case, all options which are not required should be deactivated.

If additional options are activated which are not mandatory, it may occur that no suitable firmware, which supports all activated options, is found in a DFZ archive.

4.3.2. Language Table for Device, Device Texts

4.3.2.1. Editing File for Language Table

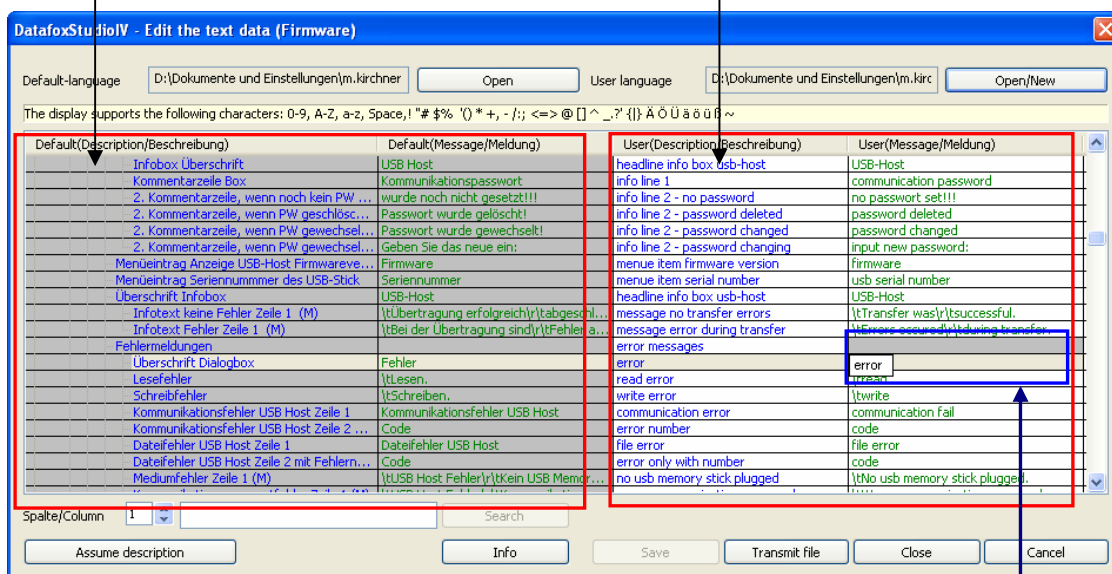
In order to ensure language compatibility, it is possible to edit the texts and messages displayed by the firmware.

Open the editing dialog via the menu

"Configuration – Language file for device (*.dfl) – Edit file for language table".

Open a device file archive (firmware)*.dfz. The default texts of the firmware with a description and the corresponding message are displayed.

Open or create a new language file for the firmware with the extension *.dfl. If you have created a new file, the right column of the list is empty.



Work within the lists with single mouse clicks only. **NO** double-clicks! Select a line from the list with a single click.

With another single click on the column User (Description/...) or User (Message/...) the cursor is displayed in this field.

Now you can enter or edit the text. When you finish the entry, the description from the column Default (Description/...) is taken over and you can edit it as well. You can find prepared .dfl-files on the Datafox DVD.

In order to transfer user text data, you first must save all changes. A full-text search for the list is available. Enter the text and choose the column you want to search for the text. By this way, you can edit texts quickly and efficiently.

DFL files can be found here:

< _Datafox DVD\MasterIV-Serie\Datafox Geräte\Datafox Software MasterIV-04.02.00_Release\Gerätedateiarchiv (Firmware)>

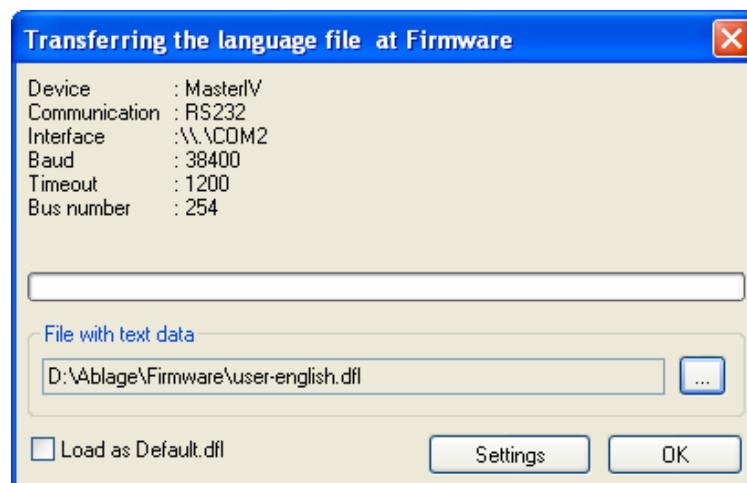
4.3.2.2. Transferring File for Language Table

Save the text data after editing.

Via the button "Transfer file" the text data is transferred to the connected device.

If an already edited DFL file is available, it can be transferred directly. Call the function via "Configuration – Language file for device (*.dfl) – Transfer file of language table". The following window opens.

Specify the location where the .dfl-file to be transferred is saved and click OK.



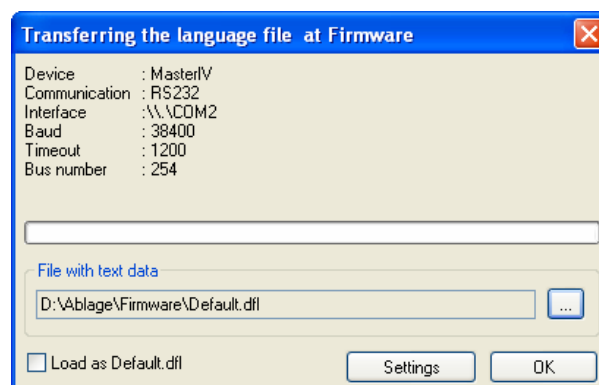
Note:

Cyrillic and Chinese characters cannot be displayed.

Restoring Default Settings

If you want to restore the default settings for the language (German), there are 2 possibilities:

1. You transfer a DFL file with the respective German texts.
2. You transfer a default DFL (empty). In this case, the basic settings are restored.
A Default.dfl can be found in each firmware DFZ file. Change the file extension from DFZ to ZIP and extract the file.



4.3.3. Color Selection for Devices with Color Display

4.3.3.1. Editing Color Data of Firmware



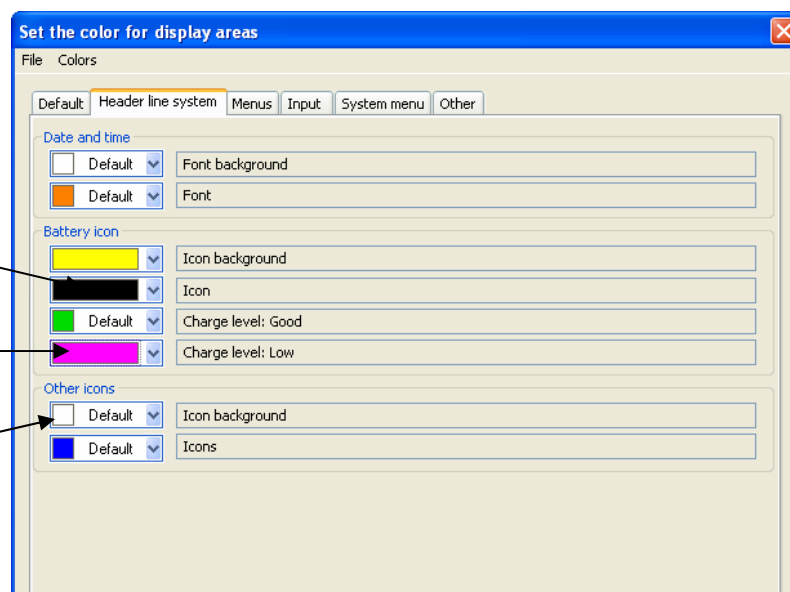
For color displays, the colors for background, fonts and symbols can be customized. This function is available via "Configuration – Color selection for devices with color display (*.dfc) – Edit file of color selection".

Via File you can open a DFC file or create a new one.

Selected color for the selection.

Set default color.

Valid area for displaying.



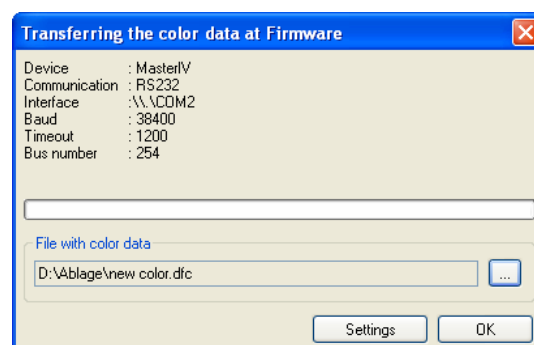
Create a new color scheme as DFC file for single areas of the display or load a color scheme from a DFC file for editing. Save all changes of the color scheme and close the dialog window. If you have transferred a new color scheme to the device, it is retained after transferring a new setup.

Via "File" you can save the color scheme created and transfer it to the device.

4.3.3.2. Transferring Color Data of Firmware


This function is available via "Configuration – Color selection for devices with color display (*.dfc) – Transfer file of color selection".

Specify the location where the DFC file to be transferred is saved and click OK to transfer the new color selection.



4.3.4. Configuring Data Storage

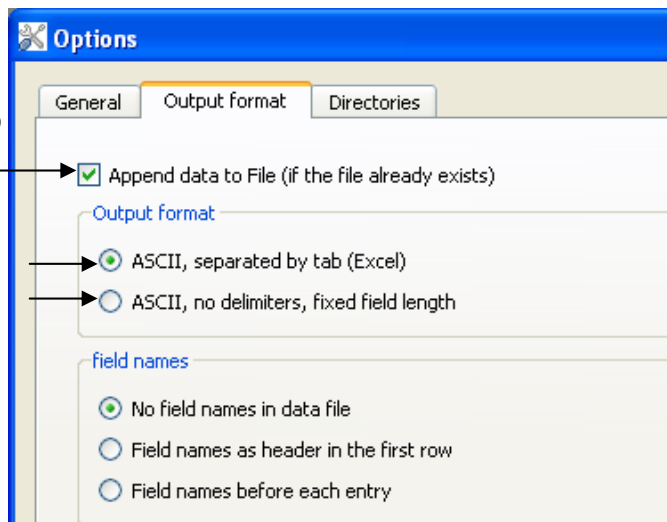
Data which are read from a Datafox device can be stored according to certain specifications.

Call this function via the symbol  or the tab "Configuration – Configure data storage".

Specify whether an existing file is to be overwritten or the data are to be appended.

Determine the output format.

See example 1
See example 2




Caution:

If you want to save the data of several devices in one file, you must activate the option "Append data to file". Otherwise, already existing data are overwritten.

Example 1 (separated with TAB(→))

Data record	Date Time	ID	Name	Label
	2011-09-16 15:38:03 →	00044591 →	JohnPublic →	103 ↵

Example 2 (the value is preceded by space characters depending on the data field length)

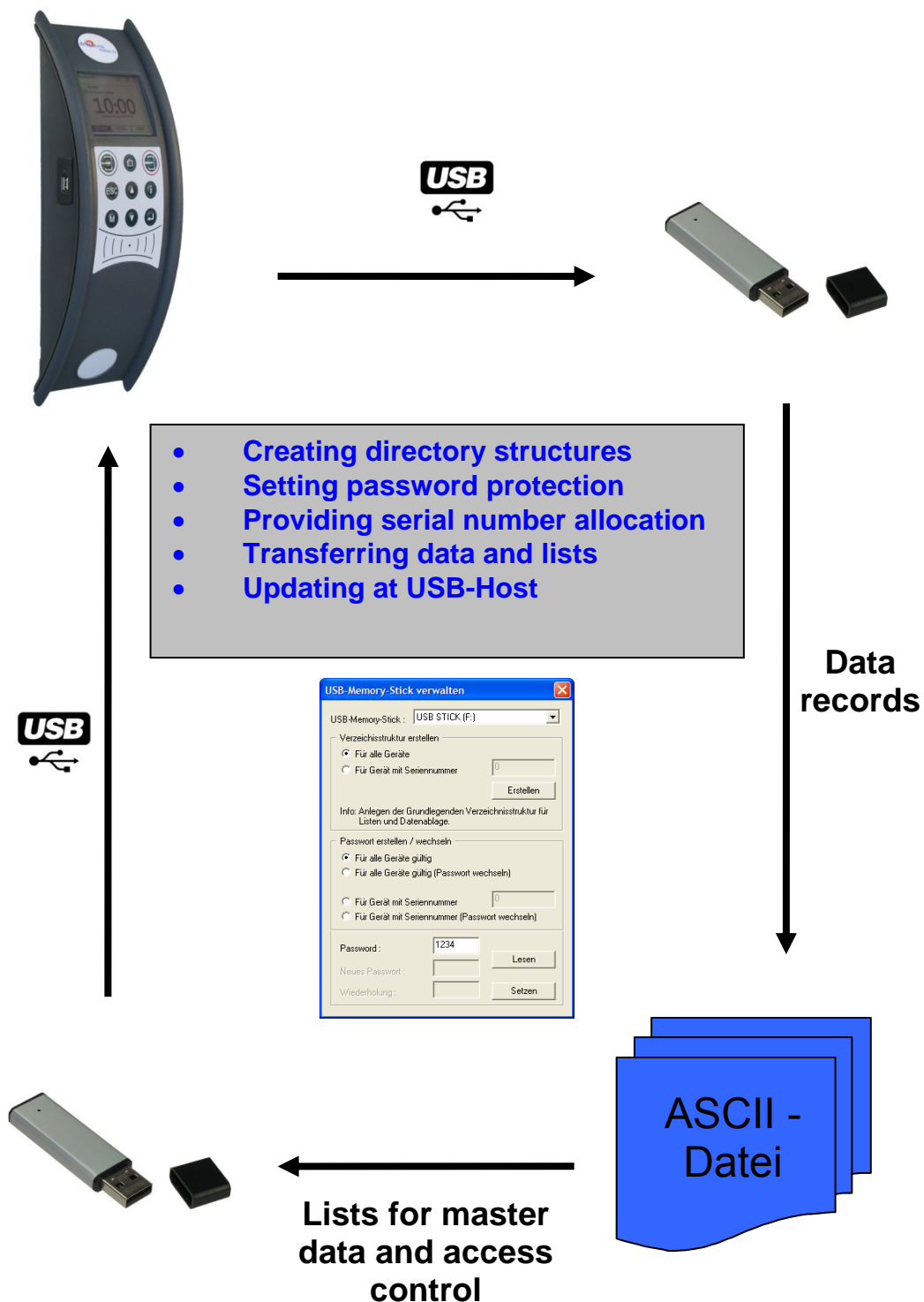
Data record	Date Time	ID	Name	Label
	2011-09-16 15:38:03	00044591	JohnPublic	103↵
	2011-09-16 15:39:07	00044598	JanePublic	109↵

4.3.5. Configuring USB Stick

With this function you prepare a USB stick for use as external data medium. For a separate description of the configuration see the documentation "Manual USB-Host Master IV_Costumer_.pdf" on the Product DVD.

< _Datafox DVD\MasterIV-Serie\Datafox Geräte-Optionen (eingebaute Module)\USB-Host für Datenübertragung per USB-Stick\Dokumentation>

4.3.5.1. USB Host at Master IV



4.3.5.2. Creating Directory Structure and Password at USB Stick

Call this function via the tab "Configuration – Configure USB flash drive". In order to guarantee the data transfer between the terminal and the USB stick, you first have to create a directory structure on the USB stick.

In the steps 1 to 5 the data structure and the password are stored on the USB stick. With it all USB terminals are controlled, irrespective of their serial number.

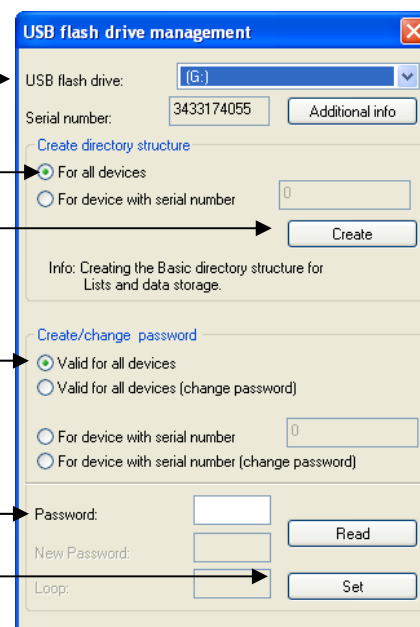
1. Select the drive which was assigned to the USB stick.

2. Create directory structure for all devices independent from serial number.

3. Create password which is valid for all devices. The correct password is the basis for data transfer between terminal and USB stick. Thus, it is prevented that every USB stick with the set data structure can read data from the device.

4. Provide password, e.g. 1234

5. Set password on USB



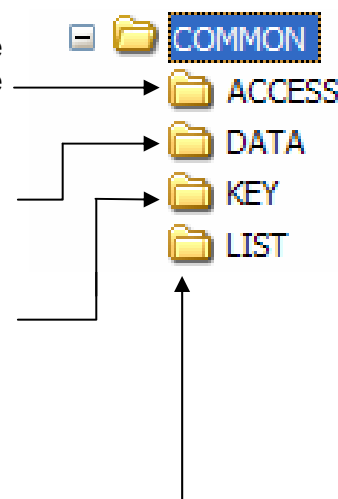
A new directory structure COMMON is created on the USB stick and used as storage for the transfer data.

The Access-folder contains access control lists which are to be transferred to the terminal. Lists have to be save das TXT files.

The Data-folder contains the data records as TXT files which are written on the USB stick by the terminal. Lists have to be save das TXT files.

The folder Key contains the key as DAT file enabling communication between terminal and USB stick. If no password has been created, the folder stays empty. Lists have to be saved as TXT files.

In the List-folder all lists which are to be transferred to the terminal are saved as TXT files.



The storage structure COMMON on the USB stick is used by all terminals, that support main communication via USB. All TXT files which are filed on the USB stick must correspond to the list descriptions in the setup concerning designation*, field size and format. Use a tabulator as field separator and CR + LF at the end of line.

*The designations of the text files (lists or data) can only be selected in the format 8 dot 3. That means, that each text file has to be unique on the basis of its first 8 digits. If the list descriptions are

not unique within the first 8 digits, a termination of the communication may occur and no lists are transferred to the terminal.

If you want to transfer data and lists terminal oriented, you have to create an additional data structure in the following steps. The selection is based on the serial number of the terminal.

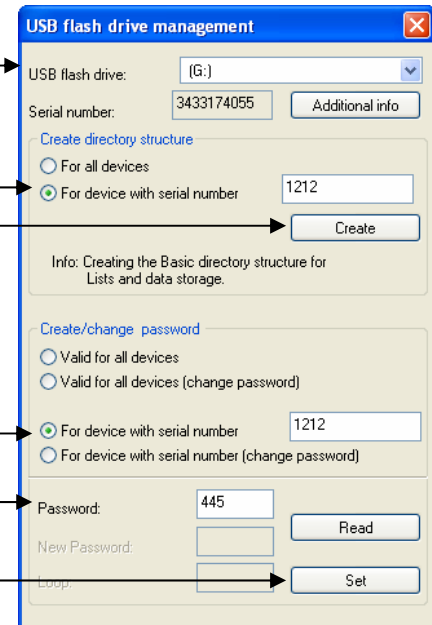
6. Select the drive which was assigned to the USB stick.

7. Create directory structure for a terminal with the corresponding serial number.

8. Create password which is only valid for the terminal with the serial number provided (e.g. 1212). The correct password is the basis for data transfer between terminal and USB stick. Thus, it is prevented that every USB stick with the set data structure can read data from the device.

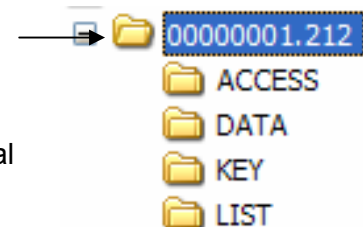
9. Provide password, which is to be set only for this terminal, e.g. 445.

10. Set password on USB



The screenshot shows the 'USB flash drive management' window. It includes fields for 'USB flash drive:' (G:), 'Serial number:' (3433174055), and 'Additional info'. There are two main sections: 'Create directory structure' and 'Create/change password'. In the first section, 'For device with serial number' is selected with the value '1212'. In the second section, 'For device with serial number' is also selected with the value '1212'. A 'Password:' field contains '445'. There are 'Read' and 'Set' buttons at the bottom.

On the USB stick an additional directory structure only for the terminal with the serial number 1212 has been created.



All lists and data can only be read or written by the terminal with the serial number 1212.

The key provided is valid for this terminal only.

When communicating with the terminal (1212), the terminal only accesses the directory structure created for that purpose. Thus, no transfer with the general directory COMMON takes place. An own directory structure can be created for each terminal.

When plugging the USB stick in the terminal for the first time, the setting and the password set are written on the terminal. From this moment on, communication is only possible, if the correct password was entered.



Note:

The USB stick should only be used for communication and data transfer of terminal and PC. Data and folder structures which are not related to the data transfer might cause negative effects concerning the writing of data on the USB stick. A termination of communication with the USB stick may occur and data records may be damaged.

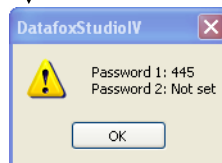
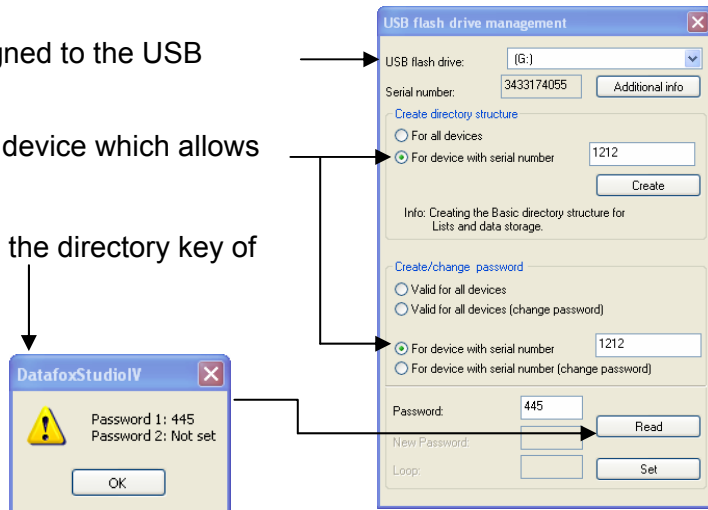
4.3.5.3. Changing Password for Communication

In order to change an already existing password on the USB stick and on the terminal, you have to use the same application, you already used for creating the directory structure.

Select the drive which was assigned to the USB stick.

Specify the serial number of the device which allows for changing the password.

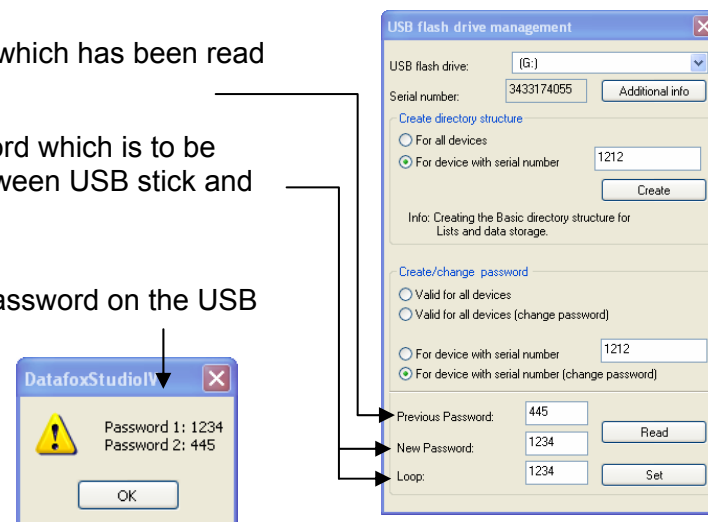
Read the current password from the directory key of the USB stick.



Provide the previous password which has been read from the USB stick.

Provide and set the new password which is to be used for the communication between USB stick and terminal in future.

In order to check the changed password on the USB stick click „Read“.



The new password is changed on the terminal during the next communication. Until then, both passwords are stored on the USB stick.

If you cannot access to the password set due to loss of the USB stick or your records, the password can be deleted on the terminal.

Press the keys "ESC" and "F2" at the terminal. You are now in the USB host bios menu. Changing the password is possible via the menu entry "Change communication password". The previous, unknown communication password is required. Because in this case it is unknown to you, enter "*****". The entry field for the new password remains empty. This way, the communication password is not overwritten but deleted.

During the next communication with the USB stick the communication password stored there is transferred to the terminal. Thus, the new password for the terminal has been taken over.

You receive the ***** password from your sales partner specifying your serial number.

4.3.5.4. Transfer from Master IV to USB Stick

The BIOS of the Master IV terminal has to be set as main communication interface USB host. In order to do this, press both arrow keys ▲ ▼ at the same time. You are now in the menu BIOS. Select "System menu BIOS". In the following submenu select the entry "communication" and confirm your selection with "Enter". You are then asked whether you want to cancel communication to make settings. Confirm with "Enter". The current main communication is displayed as "Interface". In order to change it, make a selection with "Enter". Select "USB host" in order to permit communication with a USB stick. After changing the settings in the BIOS, disconnect the terminal from voltage for a short moment, so that all settings can be taken over after a restart.

Auto-start of communication:

In order to transfer data from Master IV terminal to a USB stick, the stick with the created data structure must only be plugged into the USB port. The transfer of data and list starts automatically. The state of communication is displayed in form of a bargraph. After successful communication the operation of the terminal is restarted. The USB stick can be unplugged and you can start data recording at the terminal again.

Manual start of communication:

In order to enable a manual start of communication, the USB stick must be plugged into the USB port of the terminal. Press the keys "ESC" and "F2" to start the USB host bios menu. In this menu, only single actions can be executed like only writing data on the USB stick or only transferring lists from the USB stick to the terminal. After confirming, the transfer is started and the status displayed.

Start manual data backup:

All data records which have been created on the terminal since the last writing of the setup, can be transferred to the USB stick by a data backup. Via the backup function, also data records which have been read from the terminal with the USB stick are written as not yet read data records. Thus, data can be reproduced even in cases of data loss during processing or loss of the USB stick. In order to enable a manual start of communication, the USB stick must be plugged into the USB port of the terminal. Press the keys "ESC" and "F2" to start the USB host bios menu. Start the data backup. After confirming, the transfer is started and the status displayed.

4.3.5.5. Updating the USB Host

Firmware of the USB host:

Start the USB host bios menu. The entry "Firmware" provides information about the current firmware in the USB host. Updating the firmware via a USB stick is possible. For this purpose, proceed as follows:

Transfer the firmware file "ftrfb.ftd" to an empty USB stick. No directory structure must be available. Set the main communication of the terminal to RS232. Disconnect the terminal from voltage. Plug the USB stick into the USB host at the terminal. Connect the terminal to voltage. In the following 5 seconds the firmware on the USB host is updated. Unplug the USB stick and reset "USB as main communication". Check whether the update was successful by checking the version of the firmware at the USB host bios menu.

4.3.6. System Variables of Signal Processing

Basics:

System variables are similar to global variables. The values of the variables can be changed by different ways. They act like global variables.

Special attention must be paid to the use of these variables in order that no errors may occur.

The variables are set in the setup via "Signal processing".

The **Digital Input I1 und I2** are used here.

These variables can be edited with a function. (next page)

Figure 2 shows the function for modifying or overwriting these variables.

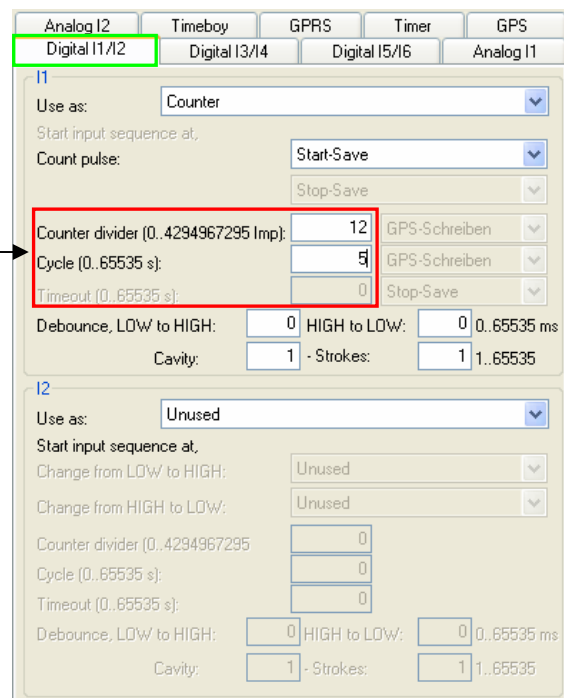


Figure 1

Editing the System Variables of Signal Processing

You can access the variables of signal processing via the tab "Configuration – System variables signal processing".

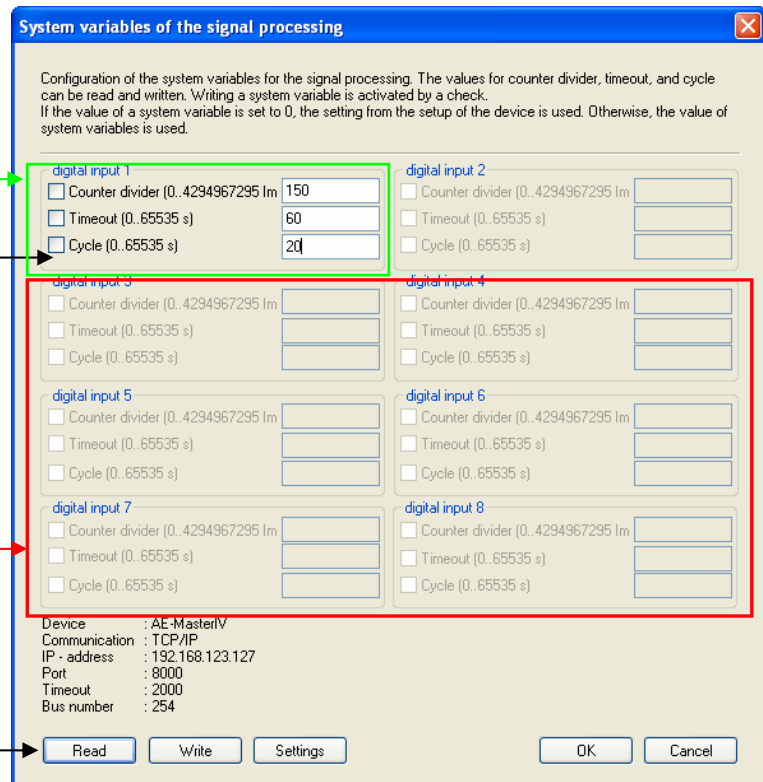
Via this function the behavior of a terminal is controlled with the help of system variables of signal processing.

System variables de-allocated by the setup.

By activating the checkboxes you determine which variables are to be written.

System variables locked by the setup.

After clicking on Read, the system variables used are displayed.



Configuration of the system variables for the signal processing. The values for counter divider, timeout, and cycle can be read and written. Writing a system variable is activated by a check. If the value of a system variable is set to 0, the setting from the setup of the device is used. Otherwise, the value of system variables is used.

Digital Input	Counter divider (0..4294967295 Im)	Timeout (0..65535 s)	Cycle (0..65535 s)
digital input 1	<input checked="" type="checkbox"/> 150	<input checked="" type="checkbox"/> 60	<input checked="" type="checkbox"/> 20
digital input 2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
digital input 3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
digital input 4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
digital input 5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
digital input 6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
digital input 7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
digital input 8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Device : AE-MasterIV
 Communication : TCP/IP
 IP - address : 192.168.123.127
 Port : 8000
 Timeout : 2000
 Bus number : 254

Read Write Settings OK Cancel

Figure 2

If the setup is in the device, you can read the system variables from it. Only those system variables are activated in the dialog which are defined by the setup and used.

If no system variable is activated after reading, check the setup settings.

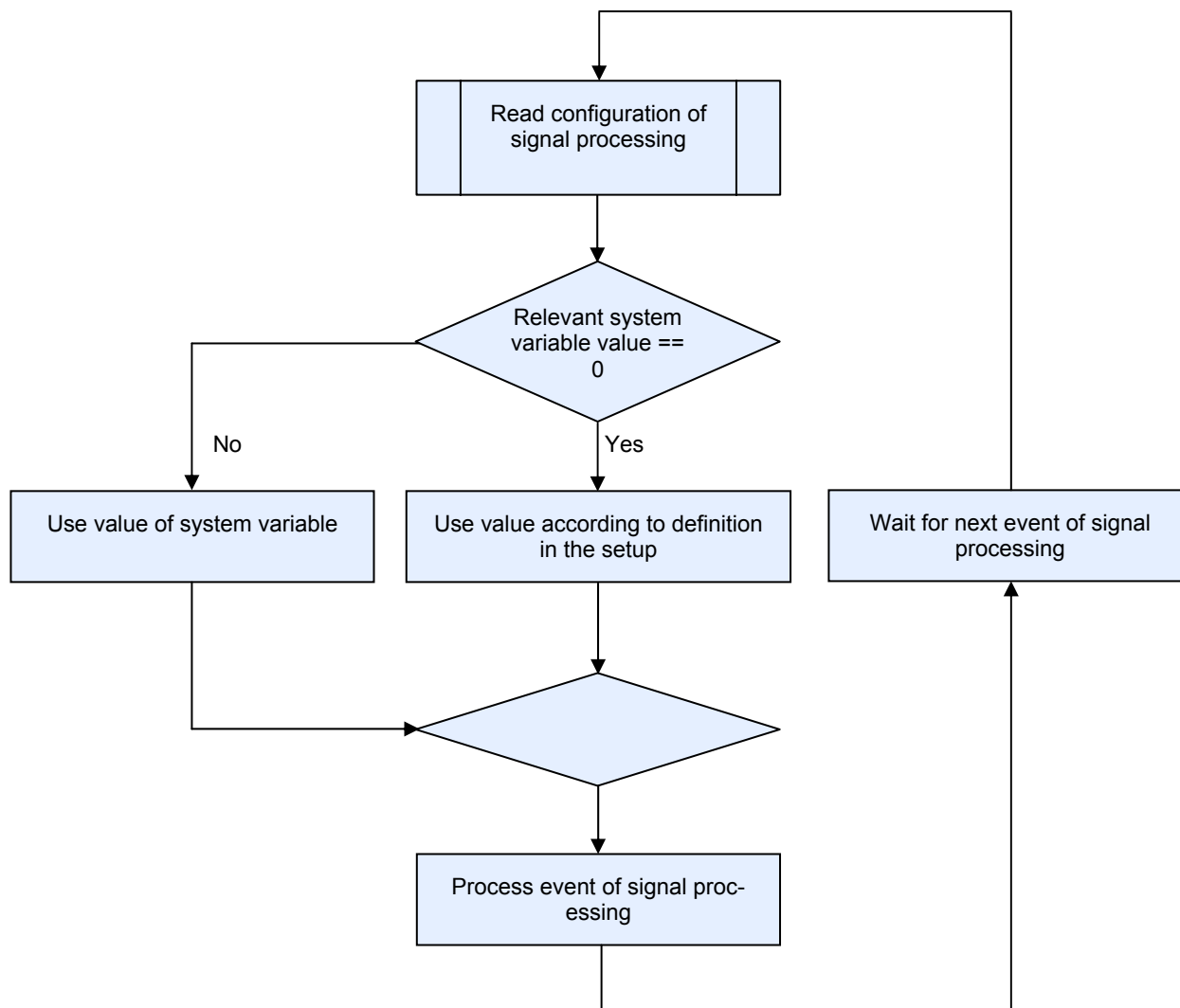


Note:

If the system variables are changed by this function, they are also used for the run time of the device. The values of the system variables set in the setup (see *Picture 1*) are not considered.

If you want to ensure that the values defined in the setup are used for all active system variables of signal processing, set all values to zero in the dialog and write these zero-values into the device.

The firmware works according to the following flow chart:



4.3.7. System Variables HTTP / GPRS

4.3.7.1. Sending Data Records with HTTP via Mobile Communications



The Datafox device is able to send booking data promptly to a web server via GPRS. For this purpose, it is necessary to configure the device for this communication type. When data is created in the device, firstly a TCP/IP connection is established and then the following character string is sent:

Plaintext request

```
getdatagv.php?table=BB&bTYP=Manu&bLOG=Log&bDAT=2011-05-24_08:30:12&bPER=Per&checksum=2120
```

Plaintext response

```
status=ok&checksum=2120 status=ok&checksum=2120
```

- GET example/getdata.php? is the prefix of the HTTP data and specifies the path on the web-server where the corresponding php-script for processing the TTP data is to be found.
- **table** is a data record description from the setup (the table from which data are to be transferred).
- checksum serves for error detection during data transfer.

You should enter only a few characters for the tables and field names in order to keep the transfer volume low.

The 2120 (checksum) is the sum of all ASCII values of the transferred parameter values (only of the values, not of the filed name; that means everything that is written between = and &). The web-server has to send back the following answer within HTTPTIMEOUT:

- 1.) In case of success (checksum correct): status=ok&checksum=pruefsumme.
Then the data record is deleted in the device.
- 2.) In case of an error (checksum incorrect): status=error&checksum=pruefsumme.
Then the last data record is sent again.

Caution:



If the server does not accept the data record (incorrect checksum, etc.), the device tries to send the data record again. As long as this is not successful, the device can send no more data records.

After several unsuccessful attempts you should assume a problematic data record and save it separately on the server for review.

It also causes higher data volume and possibly higher costs.

4.3.7.2. HTTP response, and optional parameters

As fixed parameters to be specified:

“**status=ok&checksum=xxxx\r\n**” or “**status=error&checksum=xxxx\r\n**”

allowed optional parameters:

“**&time=**” give the actually time from server back

example: **status=ok&checksum=3142&time=2010-10-28_12%3A18%3A24\r\n**

The time can be synchronized at each response as the device only takes over the time sent if it deviates by 10 seconds. This only applies for the HTTP response, for the DLL command DFCS setTime.

“**&beep=?**” for signal beep

- | | | |
|------------------|-------------------------|--------------------------------|
| • 0 = no beep | • 4 = 1 x short 1x long | • 8 = 3 x long |
| • 1 = „OK“ Beep | • 5 = 2 x short | • 9 = 1 x short - long - short |
| • 2 = ERROR Beep | • 6 = 2 x long | • 10 = 1 x long – short - long |
| • 3 = 1 x long | • 7 = 3 x short | • 11= SMS |

“**&service=1**”

Call for service connection

More information gives the „Active Mode“ and *.ini file.

Some options are possible: give back, host and port.

Example:

- service=1\r\n**
- service=1&host=www.datafox.de\r\n**
- service=1&host=123.123.123.123\r\n**
- service=1&host=www.datafox.de&port=4711\r\n**

Example a) connection „Active-Mode“ to server via in the device (in “active.ini”) registered server.

at b) and c) conection to server with „IP 123.123.123.123 and the port from in the device (in “active.ini”) registered server-port.

In the last example, make a connection to the Datafox-Server on the Port 4711.

Set of global variabel via Response

It is possible to change the response of the setup and global variables to be set. For example, the setup is the first global variable with the name, id 'created, you can address this via the following statement:

- to call with ID: **&setup.1=1234**
- to call with name: **&setup.”id”=1234**

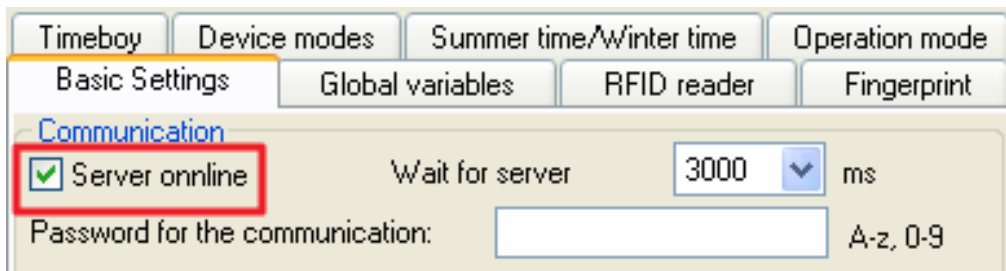
Example Answer (plain text): **status=ok&checksum=2027&setup.id=1234\r\n**

Start a input sequence in the signal processing

&ek=Name (Name of sequence) \r\n.

In this case, the name must match the name input sequence completely, otherwise it will not run. Now a device receives this text, the input string is executed.

The following parameters are only possible with the option „Server online“



“&message=” show the message on the display (only online)

Example: status=ok&checksum=3142&message=Hallo\rMessage from\rDatafox&delay=5\r\n

“&delay=” the time, How long the message is to be displayed (Value in Seconds).

Specify as fixed parameters:

“status=ok&checksum=” or “status=error&checksum=”

Parameterization of Configuration File "GPRS/HTTP".ini

Open the configuration file (e.g. GPRS.ini) for editing.

Make all settings for transferring data with HTTP via GPRS.

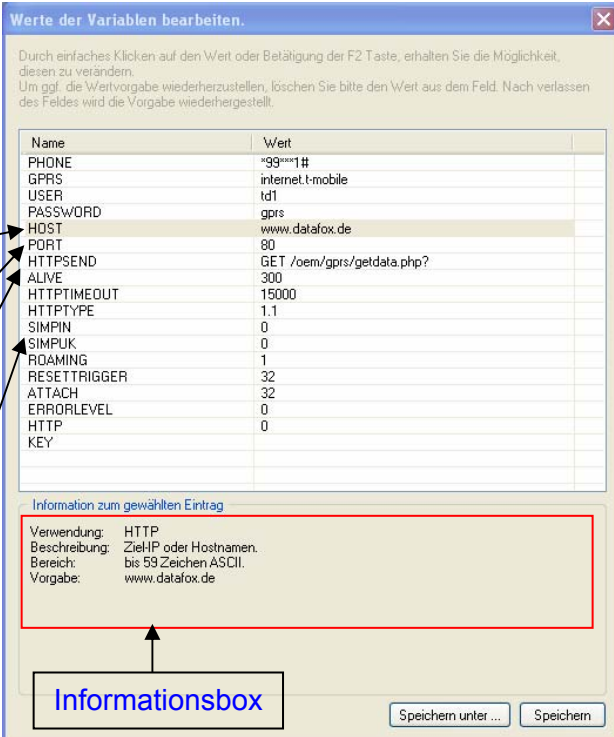
You can find .ini-files for certain providers on the Datafox DVD under:
<DVD\MasterIV-Serie\Datafox Geräte\Datafox Software MasterIV-04.02.02_Release\Kommunikationsmodul http>

HOST: IP address or host name at which the server is accessible.

PORT: on the server.

HTTPSEND: GET-request with corresponding php-script

SIMPIN and SIMPUK are only temporarily saved in the .ini and deleted after closing the transfer dialog. However, the PIN is stored in the device. If you replace the SIM card you must transfer the PIN again only if it has changed.



Name	Wert
PHONE	"99""1#
GPRS	internet.t-mobile
USER	td1
PASSWORD	gprs
HOST	www.datafox.de
PORT	80
HTTPSEND	GET /oem/gprs/getdata.php?
ALIVE	300
HTTPTIMEOUT	15000
HTTPTYPE	1.1
SIMPIN	0
SIMPUK	0
ROAMING	1
RESETTRIGGER	32
ATTACH	32
ERRORLEVEL	0
HTTP	0
KEY	

Information zum gewählten Eintrag

Verwendung: HTTP

Beschreibung: Ziel-IP oder Hostnamen.

Bereich: bis 59 Zeichen ASCII.

Vorgabe: www.datafox.de

Informationsbox

The **information box** shows hints for the currently selected line.
By clicking on a line, it can be edited.



Note:

We recommend using T-Mobile and Vodafone as providers for mobile communication in Germany. Experience shows that for other providers you must expect more frequent dial-ups, delayed data transfer and possibly higher costs.


Note:

12 communication errors per 15 minutes are tolerated. Then the device makes a transmission pause of up to 15 minutes. After the pause, the device attempts to send the data again. Thus, unnecessary transmission costs are prevented, in case the server is unavailable or the radio connection is too bad.


Caution:

Depending on the provider the prevention of roaming (mobile communication via third-party networks) can pose difficulties. Please check this behavior for your application. If possible, roaming should be deactivated for the SIM card.

In case of connection problems, the error analysis can be facilitated by an "Alive data record". The Alive data record helps you determine whether the device was online or offline, e.g. in case of power blackouts. On the basis of the Alive counter in the Alive data record, you can also determine whether the web server was permanently reachable. With each failed attempt to send data the Alive counter is increased. If no data reach the server and the Alive counter in the Alive data record has the value 1, the device has been removed from the power supply.

For more information on the Alive data record see the manual "DatafoxStudioIV".


Caution:

Alive data are temporary data. If the Alive data record cannot be sent (e.g. server is not reachable), it will be deleted and the Alive counter will be increased by one. The function "Alive" is activated via the Alive parameter in the GPRS.ini. Additionally to the activation, the GPRS chain has to be available in the signal processing. Take care that this function does not create unintentional data (traffic).

URL-Codierung

All data are sent via HTTP 1.1 protocol and received must be URL-Encoded.

Characters without URL-Code are:

letters **A-Z**, **a-z**, numbers **0-9** und **- _ . ~**.

All other characters displayed are follows: %ASCII code.

Example: colon: **%3A**

4.3.7.3. Sending Data Records with HTTP via LAN / WLAN



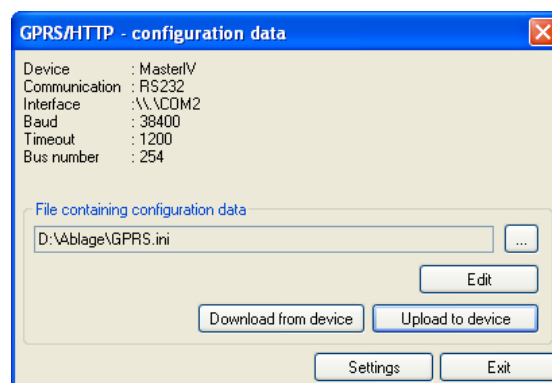
Subsequently, the term LAN is also used for WLAN.

Until now, it has been possible to send the data records created in the device to a webserver with HTTP via the cellular network GPRS. This functionality has been expanded to LAN.

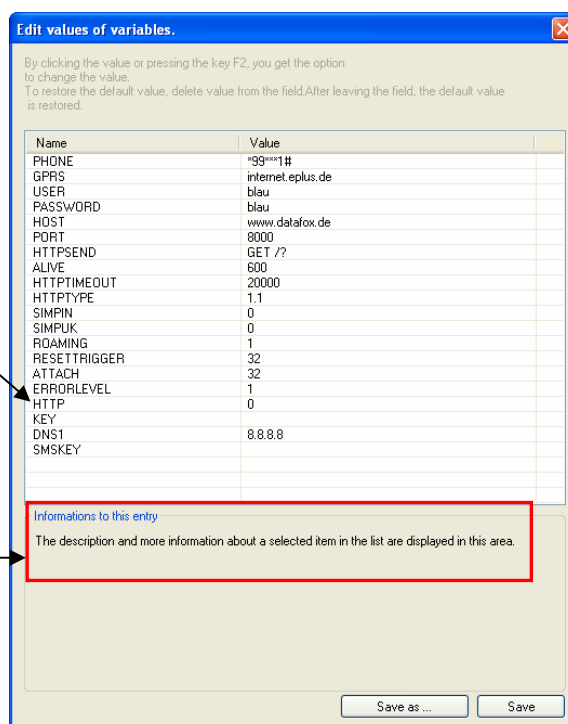
Activation via DatafoxStudioIV

The connection parameters can be set via the menu entry Configuration "System variables GPRS / HTTP".

Here, the configuration file (GPRS.ini) can be opened, edited, read from and written into the Datafox device. You can make all settings for sending the data with HTTP.



This file contains the settings for connection via GPRS (cellular network) and via LAN. The [information box](#) shows which settings have to be made for HTTP and which settings are required for GPRS only. Under HTTP, you can activate the data transfer via http/LAN. For more information see the [information box](#).



Note:

The HOST must be specified using dot notation. A name resolution for LAN is not provided, yet.

Example: instead of (www.datafox.de) use the IP address (192.168.0.1).

**Caution:**

Not all firewalls allow data transfer via HTTP. Problems could arise with Cisco-Firewall V5.0.

Activation via BIOS Menu at the Device

In each device with a TCP/IP interface, you can activate HTTP in the BIOS menu of the device under Communication. For this purpose, the entry "http" must be set to "YES".

Prerequisite for sending data with HTTP via LAN are the proper settings of the parameters in the .ini file and the communication must be set on TCP/IP.

**Note:**

If you have activated the transfer of data records via HTTP/LAN, a connection from the network cannot be accepted any more (e.g. for transferring a setup). For resetting, please use the BIOS mode via the Datafox Studio or the BIOS menu at the device.

Activation of BIOS Mode at Datafox Studio

Via the setting Gerätekonfiguration(Bios) at the Datafox Studio, you can access the BIOS of the device. Turn off the device and press "Switch to BIOS mode".

Then, turn the device on. While booting, the device queries the serial interface and switches communication to it. Here, you can call and edit the "GPRS.ini" via the button GPRS/HTTP. Reset the HTTP settings to "0" and transfer them to the device.

4.3.7.4. Encryption of Data Fields for HTTP (GPRS) Sending

If data records are sent via HTTP, field content can be transferred encrypted. The data fields of the data record are encrypted with a RC4 encryption. The encrypted characters are transferred as field content in hexadecimal format.

This document describes how to use extensions, encryption of data and the setting of global variables.

Activating Encryption via DatafoxStudioIV

Open the configuration file (e.g. GPRS.ini) for editing via the menu entry Configuration "GPRS / HTTP - Configuration".

Name	Wert
PHONE	*99***1#
GPRS	internet.t-mobile
USER	
PASSWORD	
HOST	www.datafox.de
PORT	80
HTTPSEND	
ALIVE	0
HTTPTIMEOUT	20000
HTTPTYPE	1.1
SIMPIN	0
SIMPUK	0
ROAMING	1
RESETTRIGGER	32
ATTACH	32
ERRORLEVEL	0
HTTP	0
KEY	

By clicking on the line KEY, the window for creating the key opens.

Enter your password here.

Creation of the value for the system variable HTTP.KEY

Password:

Value for SysVar:

Notes:

From your entered password is created by pressing the button, an encrypted value for the system variable KEY. Your password is thus always present in an unreadable format for storage in the ini file and transfer.

- The password must be minimum of 6 characters long and can consist of maximum 16 characters.
- Do you want to delete the password in the device, please create a value with an empty Password and transmit it.
- To delete the value in the ini file, please clear the value and accept with OK.

Creation of the value for the system variable HTTP.KEY

Password:

Value for SysVar:

Notes:

From your entered password is created by pressing the button, an encrypted value for the system variable KEY. Your password is thus always present in an unreadable format for storage in the ini file and transfer.

- The password must be minimum of 6 characters long and can consist of maximum 16 characters.
- Do you want to delete the password in the device, please create a value with an empty Password and transmit it.
- To delete the value in the ini file, please clear the value and accept with OK.

By clicking the button "Create value from password", a key for transfer is generated.

Click "OK" to take the key over. Subsequently, you can save the settings and transfer them to the Datafox device.

Deactivating Encryption

In order to deactivate the key transferred to the device, you must create an empty password field by clicking the button "Empty value" and transfer the empty key to the device. The data records are sent unencrypted, then.

Illustration of the GET Request

in plaintext (unencrypted) and encrypted:

Plaintext request
getdatagv.php?table=BB&bTYP=Manu&bLOG=Log&bDAT=2011-05-24_08:30:12&bPER=Per&checksum=2120
Plaintext response
status=ok&checksum=2120 status=ok&checksum=2120
Encrypted request
getdatagv.php?dfcb=1000&table=e977&bTYP=14dce883&bLOG=4d7876&...&checksum=c01de865&dfce=019c1bd2
Encrypted response
dfcb=1000&status=2b97&checksum=1726950d&...&setup_2=a449fd9c&setup_blue=a9375c8d0672&dfce=b99239f3

Detection of an Encryption

In order to detect whether the data fields are sent encrypted, the beginning of the encryption is marked with 'dfcb' (Datafox Crypt Begin) and the end is marked with 'dfce' (Datafox Crypt End). 'dfcb' is the first field of the GET request and 'dfce' the last field.

The value of the field 'dfcb' is transferred in plaintext and is the 'public key'. It is a random number between 1000 and 9999. Combined with the user password, the value must be used for encryption and decryption.

Encryption of data thus is achieved by "private key + public key" as password key.

In the response, the field 'dfcb' must be returned exactly. This ensures that decryption has been successful and the response matches the request.

The value of the field 'dfce' is the same as 'dfcb' but it is transferred encrypted. During encryption it can be ensured that the key used is correct. The value of 'dfce' must equal the value of 'dfcb' after encryption.

If problems occur during encryption, the response 'dfc=error' must be sent. Additionally, information must be entered in the fields 'dfcb' and 'dfce'.

The following errors must be considered by the evaluating script:	
'dfcb' is not a number or is outside the value range of 1000 to 9999	<ul style="list-style-type: none"> Response: dfc=error&dfcb=range&dfce=unknown/missing <ul style="list-style-type: none"> Range means that the value is outside the value range. Unknown means not determined but available. Missing means not specified in the request.
'dfcb' without closing 'dfce'	<ul style="list-style-type: none"> Response: dfc=error&dfcb=1000&dfce=missing
'dfce' is not a number or is outside the value range of 1000 to 9999	<ul style="list-style-type: none"> Response: dfc=error&dfcb=1000&dfce=range
'dfce' without incipient 'dfcb'	<ul style="list-style-type: none"> Response: dfc=error&dfcb=missing&dfce=unknown
'dfce' does not equal 'dfcb'	<ul style="list-style-type: none"> Response: dfc=error&dfcb=1000&dfce=different <ul style="list-style-type: none"> Different means that 'dfce' is different from 'dfcb' after decryption.

Response of the Web Server

The field content of the request is deciphered successively using the RC4 stream cipher. The field content of the response is regarded as part of the overall data stream and is ciphered again with the current status of the stream cipher after decryption. Only exception is the first field value of 'dfcb'. It is sent back exactly like in the request.

To the response, 'dfce' must be added as last encrypted field. The value of 'dfce' must equal the value of 'dfcb'.

Activation via Script

The script must use the known "plaintext" password, not the encrypted one generated at the Studio. See the example-php at the product DVD: "dfanalyser.php".

For further information see the DLL description on the product DVD under: DVD\MasterIV-Serie\Datafox Geräte\Datafox Software MasterIV-04.02.02_Release\Kommunikationsmodul DFComDLL 04.02.02 (Windows, Linux)

4.3.8. System Variables Active Connection & Configuration

Active connection means that the Datafox device establishes a connection to a PC/server via TCP/IP or GPRS independently, logs on and that the data are received by the **application involving the DFCom.dll** or maintenance is performed by the **application**.

If you have **not** programed an **own application**, the active connection must always be deactivated so that the device cannot accept a connection.

In the DFCom.dll and the device, certain parameters must be activated in order to establish an active connection. For more information see chapter "**Configuration of an Active Connection**".

This function is available if main communication is TCP/IP, WLAN or GPRS (from GSM module [mobile communications modem] MC55 onwards). The connection is always bidirectional full duplex. The communication is based on the Datafox protocol of the MasterIV series.

4.3.8.1. Description

The concept for an active connection encompasses the realization of an initialization of the TCP/IP connection between the device software (firmware) and the DLL software. The connection is always initialized by the firmware. The connection negotiation is performed via appropriate commands with the DLL.



Note:

For most providers, establishing a (TCP/IP) connection "from outside" is not possible. Therefore, the connection must be established by the firmware. Either the connection attempts are blocked directly by the provider or the IP address determined for the PC is not the "real" one of the device.

A connection can be established in TCP/IP networks (also GPRS). Because the devices do not permit several connections, no connection must be established at first, in order to initiate a connection establishment.

The establishment of a connection (request to the DFComDLL) is negotiated as follows: The DLL receives the connection request at a listen-socket. The connection management checks whether a channel object can be created. When a channel object has been created, the connection is established and remains active for further usage.



Caution:

The "active connection" must be set to "**Yes**" in the BIOS menu of the device. If you do not have an own application allowing an active connection, the "active connection" must be set to "**No**" in the BIOS menu.

4.3.8.2. Configuration of an Active Connection

An active connection requires the following parameters to be set in the device or the application (DFComDLL.dll).

- com.active (0 = deactivated, 1 = activated) Switching the active connection on/off.
- com.notify (0 = deactivated, 1 = activated) Switching the active data record message on/off.
- com.prio (0 = highest, 65535 = lowest) Priority of the event messages in the queue.
- com.host (0.0.0.0 means all) Host to which a connection is to be established.
- com.port Port to which a connection is to be established.
- com.retry Number of the attempts for establishing a connection.
- com.timeout Timeout after the set number of connection attempts has failed.
- com.repeat Timeout until a new notification is sent if existing data records have not been retrieved after a success notification.
- com.alive Timeout when the terminal closes an existing communication channel (connection terminated) if no communication takes place. If a communication channel is not to be closed, the DLL must send a "ping" to the device cyclically. The DLL passes the value to the device at connection acceptance.
- The bus address of the DLL (for call from DFComOpenIV) is predefined with 31. The number of simultaneously existing connections is limited to 50 connections per DLL instance.

The following value ranges and default values have to be observed:

Description	Name of system variable	Value range	Default value
Activation	com.active	[0, 1]	0
Active data record message	com.notify	[0, 1]	1
Priority	com.prio	[0, ..., 65535]	0
Host	com.host	[IP address]	0.0.0.0
Port	com.port	[0, ..., 65535]	8000
Connection establishment	com.retry	[0, ..., 65535]	3
Communication timeout	com.timeout	[0, ..., 4294967295]	900
Message repetition	com.repeat	[0, ..., 4294967295]	60
Connection check	com.alive	[0, ..., 4294967295]	0

Parameters and default values for configuring an active connection

The parameters required for configuring a device for an active connection are provided in the .ini file. The .ini file contains both the parameters for active connection and parameters for a GPRS/GSM connection. The file is structured as follows:

; This area is for GPRS/GSM connection and is not discussed in detail here.

[MODEM_MC35i]

...

1) This area is for the active connection

Activates (1) or deactivates (0) the active connection (GPRS via TCP/IP)

ACTIVE=1

2) Activates (1) or deactivates (0) active notifications to a server about created data.

NOTIFY=0

3) Determines the priority according to which notifications from a terminal are processed at the server.

PRI0=0

4) fixed IP address of the server

HOST=192.168.123.169

5) port where the server receives requests.

PORT=9001

6) number of attempts for establishing a connection before a break is taken for the duration ; of the timeout

RETRY=3

7) Duration of timeout after several failures (RETRY) to establish a connection.

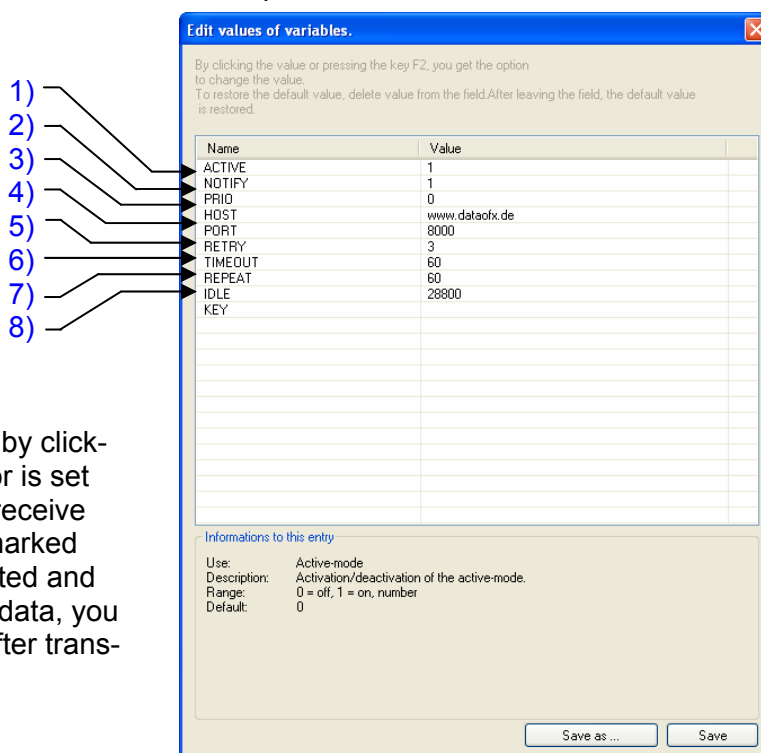
TIMEOUT=60

8) Time span after which a sent data record which has not been retrieved from the server ; is sent again.

REPEAT=60

When reading data from a device, you must specify whether to overwrite the current .ini file or to write the data into a new or different .ini file.

When the configuration of the active connection has been read from the device successfully, you can select whether to edit the data or save them in the provided .ini file.



Name	Value
ACTIVE	1
NOTIFY	1
PRI0	0
HOST	www.datafox.de
PORT	8000
RETRY	3
TIMEOUT	60
REPEAT	60
IDLE	28800
KEY	

Informations to this entry

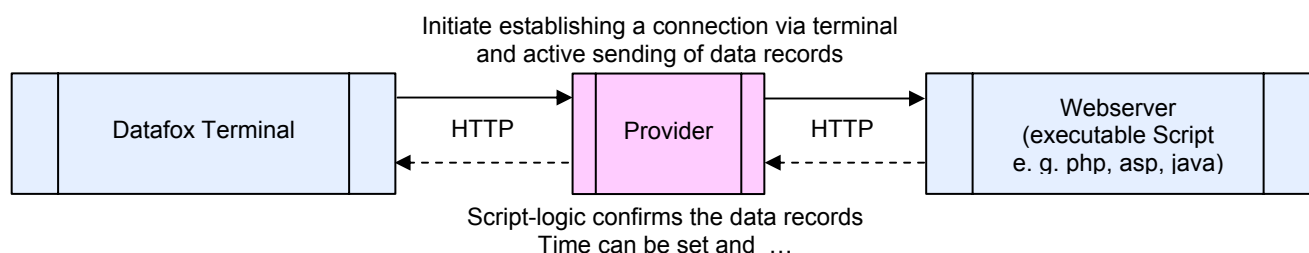
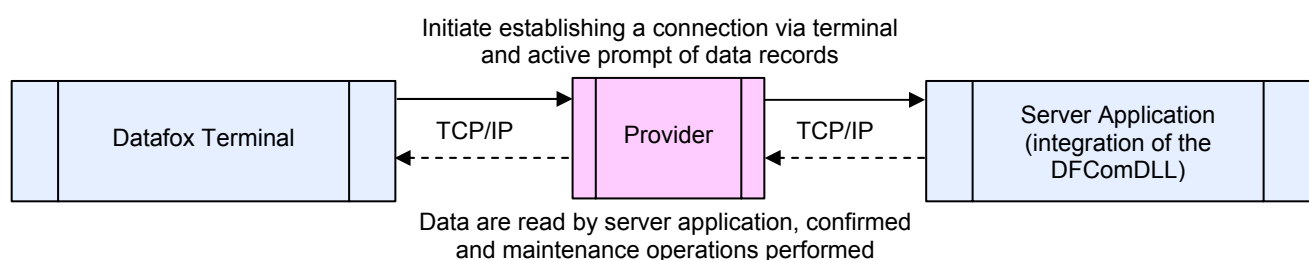
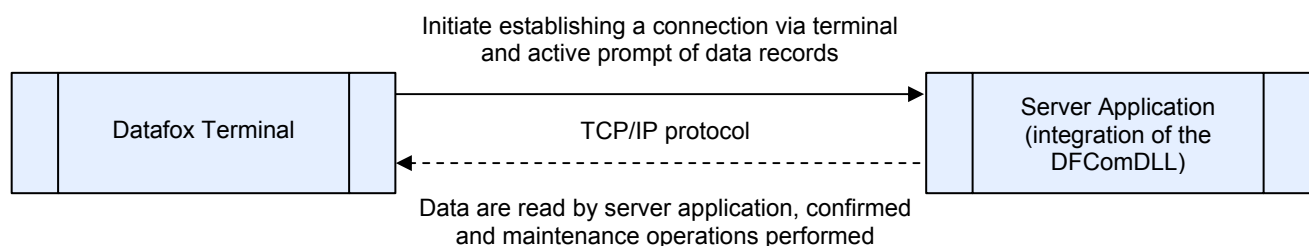
User: Active-mode
Description: Activation/deactivation of the active-mode.
Range: 0 = off, 1 = on, number
Default: 0

Save as ... Save

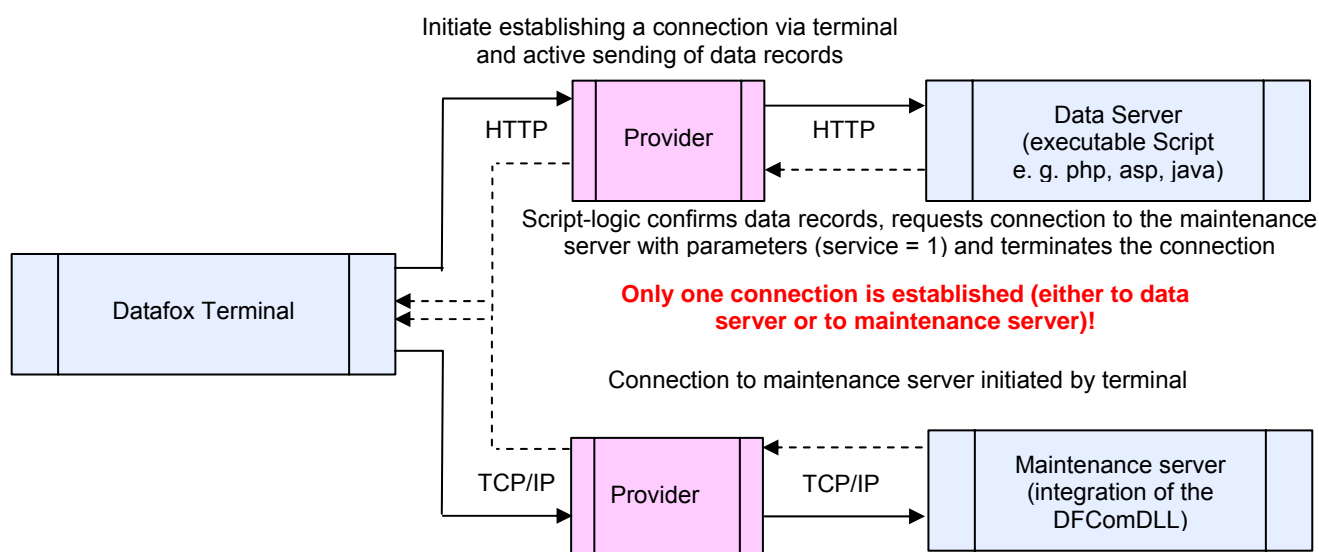
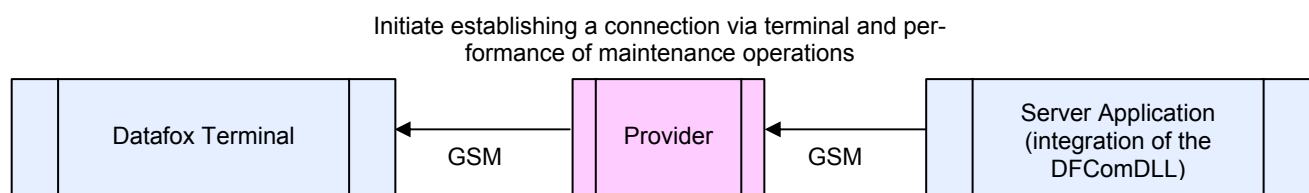
Mark the line you want to edit and by clicking in the column "value" the cursor is set for entering data. In addition, you receive information about the parameter marked concerning the value range permitted and the default value. After saving the data, you must write them into the device. After transfer you receive a status message.

4.3.8.3. Maintenance via Active Connection

The following overview shows the procedures for the active connection and the possibilities for maintaining of the terminals.



Maintenance possible only via GSM connection!



4.3.9. Device Configuration BIOS

This function is important for devices without display. All settings concerning the communication with the device can be made via this function.

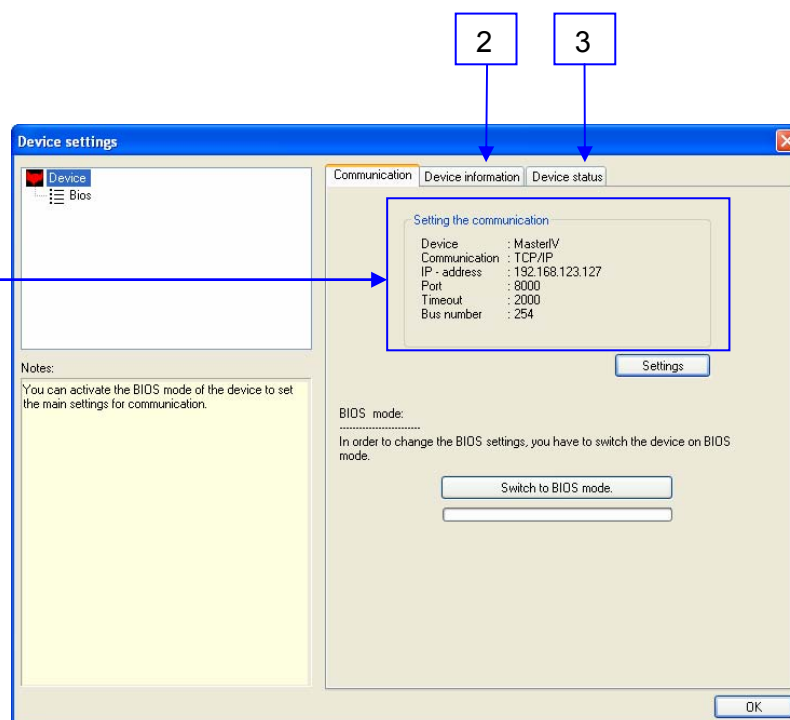
Prerequisite is that a connection to the device is established.

Further information provided here are:

1. [Set type of communication](#)
2. [Device information](#)
3. [Device status](#)

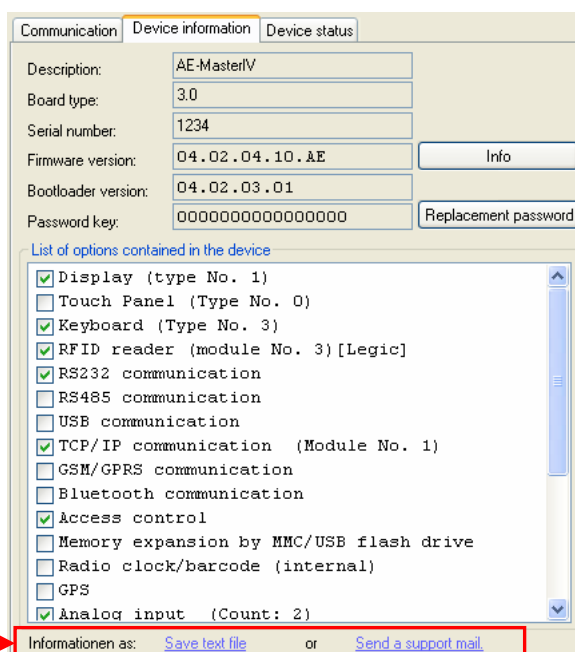
1: Set type of communication.

The communication can be set by clicking the button „Settings“.



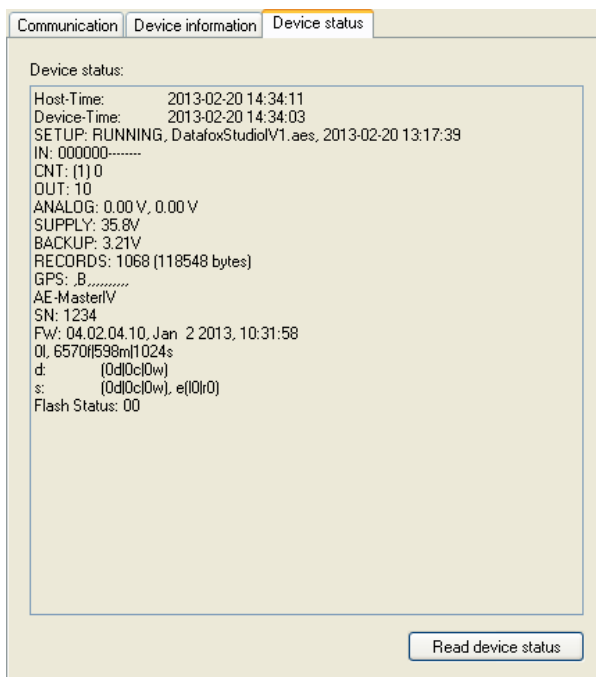
2:

The tab page „Device information“ shows with which modules the device is equipped and which device functions are supported. The communication password can be reset here as well.



This information can be exported as *.txt file or sent via e-mail for support purposes.

3:
The tab page „Device status“ shows information concerning the current device status. If some time has passed, the current status can be read again.

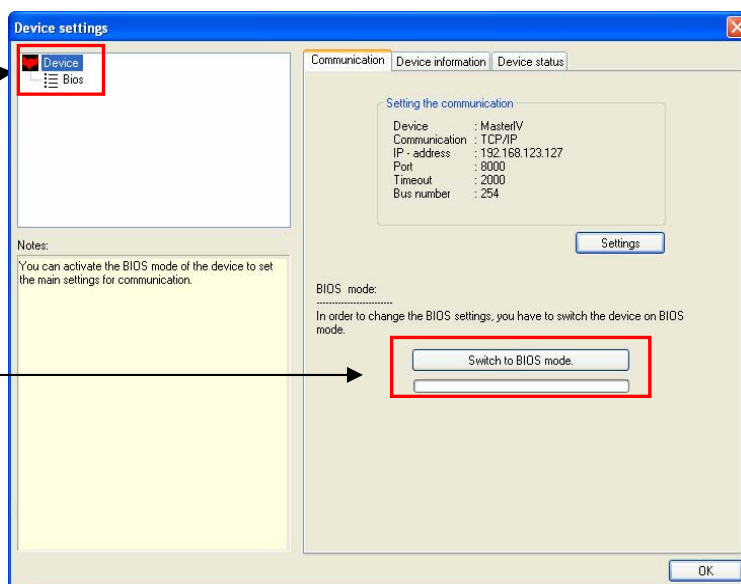


4.3.9.1. Accessing Device BIOS

If the communication type is set properly, you can switch to the BIOS mode of the device.

Step 2:
If the BIOS has been activated successfully, click on BIOS to call the settings dialog for BIOS parameters.

Step 1:
Here, you start switching to BIOS mode.



Note:

If the device does not switch to BIOS mode, another main communication (e.g. GPRS or WLAN) is set for the device.
Turn off the device and press "Switch to BIOS mode".
Then, turn the device on. While booting, the device queries the serial interface and switches communication to it.
This is especially important for switching the main communication for devices **without display**.



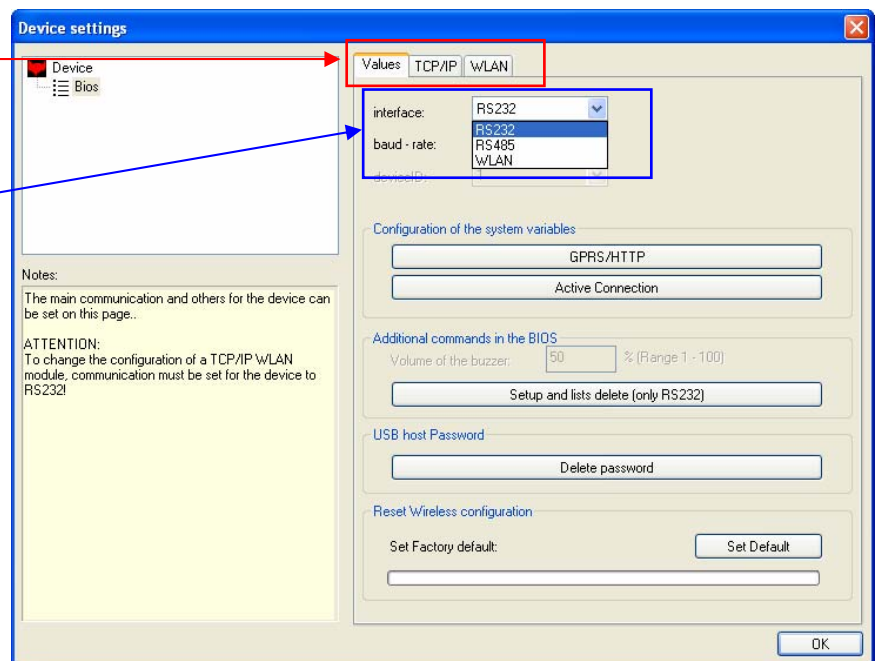
If the BIOS is activated successfully and you have switched to BIOS mode, the following settings are available:

1. Setting main communication (interface, baud rate, deviceID).
2. Configuring system variables (GPRS / HTTP and active connection). For more information see chapter HTTP about [LAN](#) and [GPRS](#) and chapter [Active Connection](#).
3. Setting volume of buzzer.
4. Deleting setup and lists from a device (only possible via RS232).
5. Resetting communication password of USB host.
6. Resetting WLAN configuration.

The tabs available depend on the equipment of the device

Under **interface**, the main communication of the device can be set.

If the main communication is changed, the device must be restarted in order to take over the settings.



If you want to change WLAN communication for a device with WLAN module, you must set the main communication RS232 at first. Confirm the changes and switch to BIOS mode again. Only now, the tab page "WLAN" is shown. We recommend proceeding step by step.

- If problems occurred during WLAN configuration, you can execute the function "Set factory default" on the tab page "Values". The progress of the function is displayed.
- Switch to the tab page "TCP/IP". The TCP/IP settings are read (see figure TCP/IP settings). Change the values according to your requirements and write the changed values back to the device.
- Switch to the tab page "WLAN". The WLAN settings are read (see figure WLAN settings). Change the values according to your requirements and write the changed values back to the device.

If you have switched between the tab pages just once, the current values for TCP/IP and WLAN are not read again automatically. In this case, you can run the process via the corresponding tab page manually.

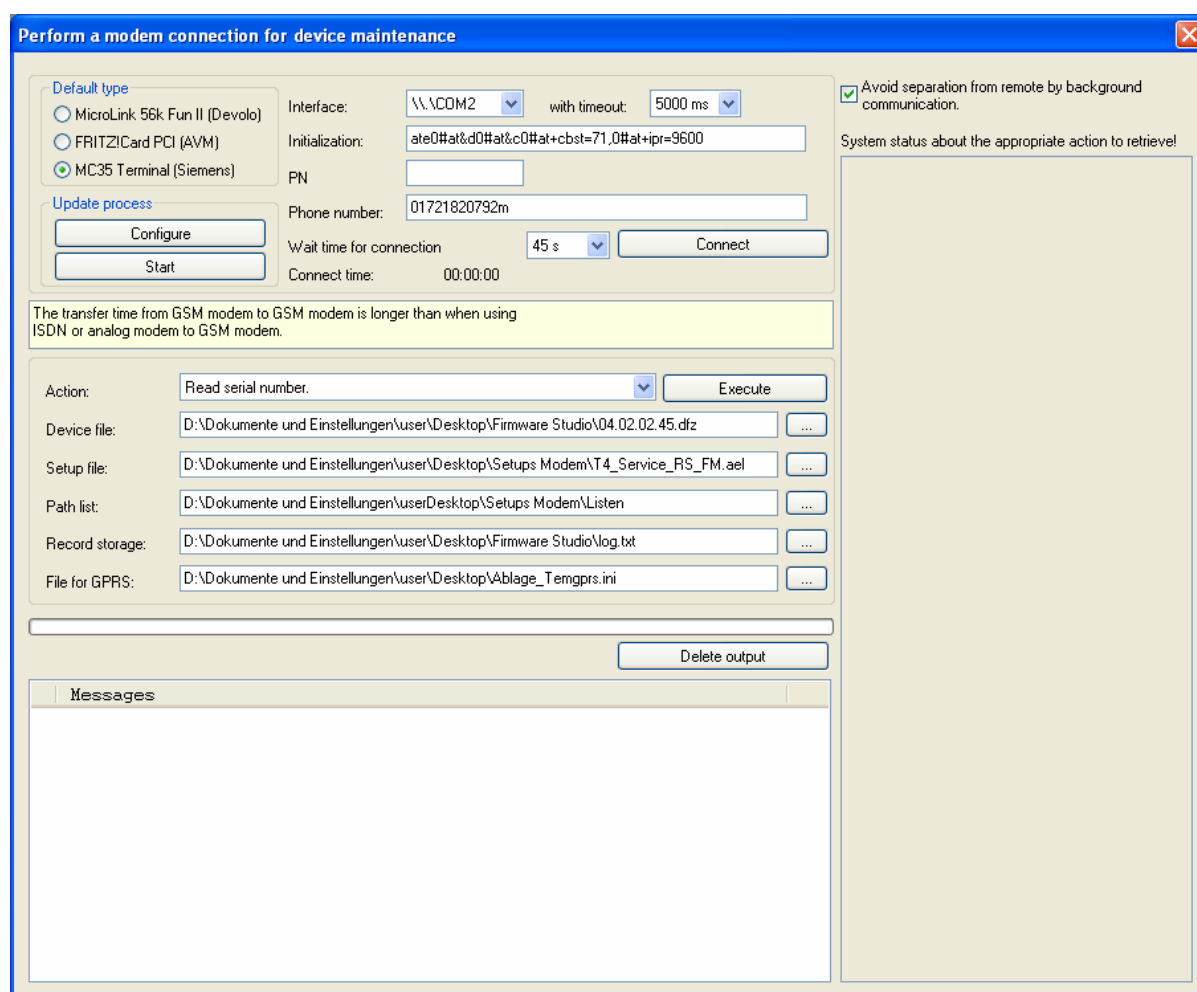
4.4. Extras

This section contains advanced functions which are rarely used and require special knowledge.

4.4.1. Device Maintenance via Modem Connection

A device using mobile communication can be maintained in different ways. One way is the device maintenance via modem connection. Another way is the implementation of the DFComDLL into a processing software which executes the functions from the DFComDLL for maintenance.

When maintaining via modem connection, note that an existing connection is terminated automatically after ca. 1 minute if no communication with the device takes places. First, a communication channel to the device must be established. For this purpose, select the modem type (analog, ISDN or cellular). If you use a different modem than the default type given, enter the AT commands for the initialization of the modem. Enter the calling number and, if necessary, the PIN for a SIM card. After establishing the connection, you can select and execute an action from the combo box. If a firmware, setup or other data are to be transferred to the device, first select the corresponding files and directories before executing the desired action. The status of the executed action is displayed at the dialog window. Additionally, the option "Determine system status cyclically" can be activated. By this option, an existing communication channel is kept open.



Perform a modem connection for device maintenance

Default type

☐ MicroLink 56k Fun II (Devolo)

☐ FRITZ!Card PCI (AVM)

☒ MC35 Terminal (Siemens)

Update process

Interface: with timeout:

Initialization:

PN:

Phone number:

Wait time for connection:

Connect time:

☒ Avoid separation from remote by background communication.

System status about the appropriate action to retrieve!

The transfer time from GSM modem to GSM modem is longer than when using ISDN or analog modem to GSM modem.

Action:

Device file:

Setup file:

Path list:

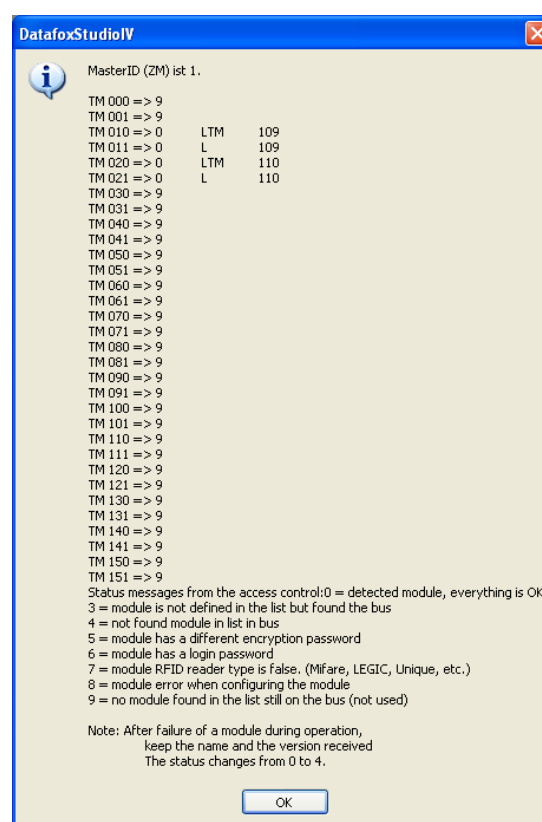
Record storage:

File for GPRS:

Messages

4.4.2. Retrieving Statuses of Access Modules

For commissioning an access control or for error analysis, status information of the access modules can be retrieved in a system. After executing the function, all modules found in the bus according to configuration are displayed. The status of a module corresponds to the value from the list. The first column shows all possible modules. TM 000 corresponds to a module at the RS485 bus with the bus-no. 0, TM 001 corresponds to a module which is connected to the module at the RS485 bus with the bus-no. 0 via RS232. This means, that 000, 010, 020, ..., 070 are all modules at the RS485 bus. All modules with the label 001, 011, 021, ..., 071 are modules connected to the corresponding module at the RS485 bus via RS232. 000 and 001, 010 and 011, 020 and 021, ..., 070 and 071 each form a module pair. The second column shows the module type. R means a reader, TM a door module (with relay for door control) and LTM a reader with door module function. The third column shows the firmware version of the corresponding module from the RS485 bus.



4.4.3. Recovering Data Records

If it is necessary to read the data records again, this function helps you to reset the record pointer to the last valid data record. Valid data records are available at the device only if no firmware update has been run, no setup has been transferred or the function to delete all data has not yet been executed.

After executing this function, all valid data records can be read from the device again.

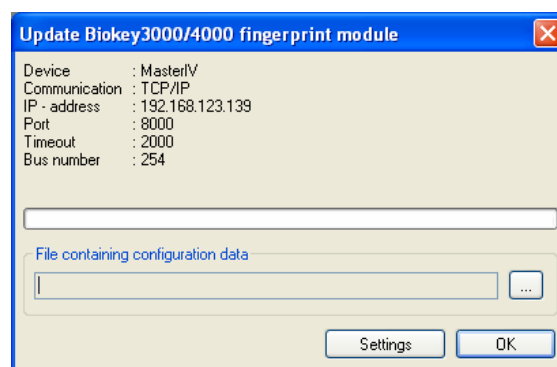
4.4.4. Update Biokey3000/4000 Fingerprint Module

The Biokey firmware update is executed via the main communication or via DLL (see DLL documentation). For this purpose, the firmware file with the format *.up3 for Biokey 3000 and *.up4 for Biokey 4000 is stored temporarily in the flash of the PZE-Master and checked. Then, the update is run within the device. There are some device configurations where this function is not contained by default. In such a case, proceed as follows: Ensure that the device firmware (*.dfz), device setup (*.aes) and the lists required are available. Then, you must transfer the firmware (PZEBioKeyUpdate.hex or AEBio-KeyUpdate.hex) to the device which provides the function for the Biokey firmware update only for hardware 2.0 and 2.1. Subsequently, you execute the Biokey firmware update as described below. After successful execution, you must restore the device with the original firmware (not Biokey), setup and lists.

Prerequisites:

- DatafoxStudioIV version 04.02.00.xx or higher
- Firmware version 04.01.05.11 or higher

Select the firmware file to be transferred
(Bi-okey3000_vXXX_datafox.up3 or Bi-okey4000_vXXX_datafox.up4).
The XXX stands for the firmware version.



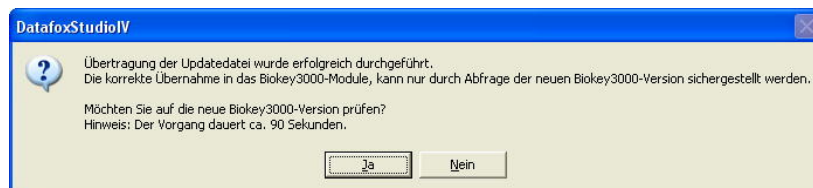
Caution:

Observe that only approved firmware files are loaded, otherwise operation failure of the Biokey3000 and Biokey4000 module may occur.

Before executing the update, the current version of the module is checked. Now you can decide whether to run the update or not.

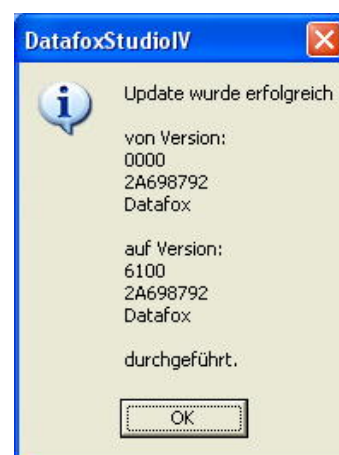


After transfer to the terminal, the new firmware is directly transferred into the Biokey3000 / Biokey4000.



By clicking “Yes” you can display an additional message after checking the update and by clicking “No” you can finish the update without getting a final message.

After finishing the firmware update, this message is shown if you have clicked “Yes” in the former dialog window. Keep to this procedure also when updating via modem connection. The only difference is that you do not start the process via the menu but the modem dialog.



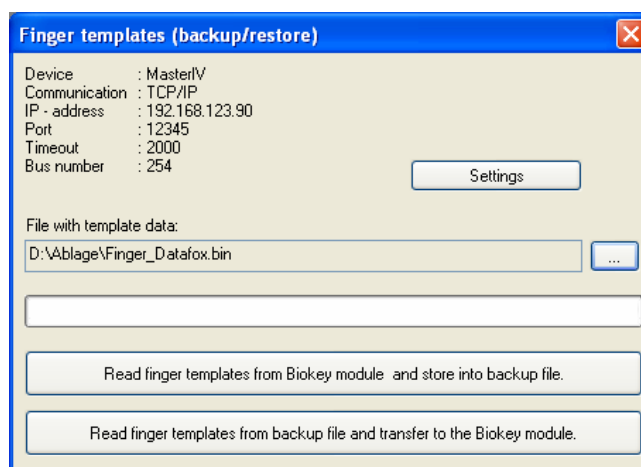
Caution:

From Biokey3000 version 6100 onwards, the parameters for image quality must be reduced from ca. 70 to ca. 40 in the basic settings of the setup.

4.4.5. Backup/Restore of Finger Templates

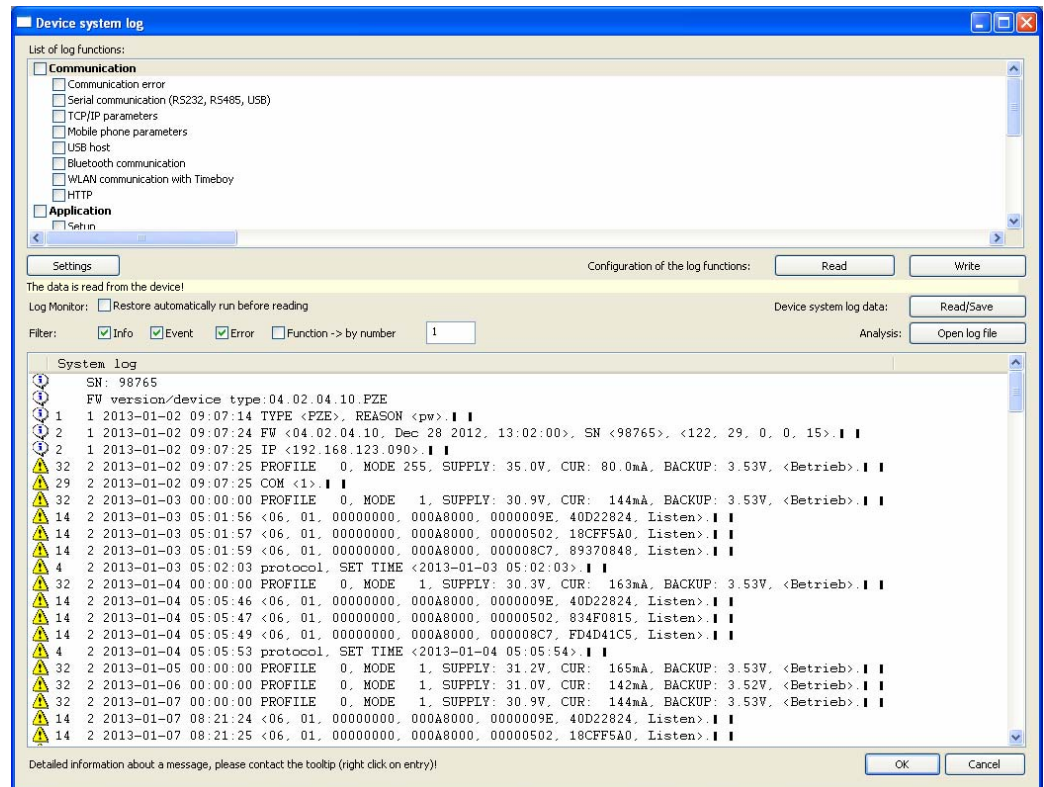
DatafoxStudioIV provides a backup and restore function for the BioKey3000 module. With this function, it is possible to create a complete backup of all finger templates from a BioKey terminal and to transfer it to a new terminal.

In order to execute the function, you must provide a backup-file. It can be a not yet existing file in order to create a new backup or you provide an existing file in order to transfer an existing backup to a new terminal.



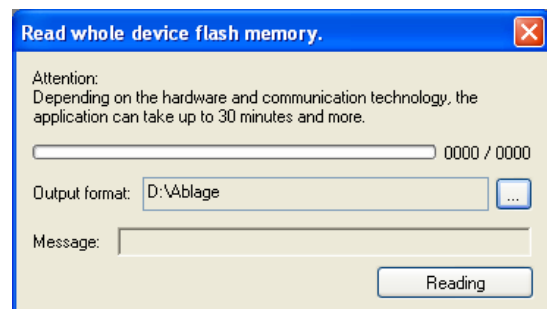
4.4.6. Reading System Logs

In case of undefined behavior of the device, you can read the system logs and analyze them. When executing this function, the following dialog window opens. Via the button "Read/Save" the current system log of a device is read. The file name proposed is a unique file name due to its timestamp and should not be altered. Only this way, you ensure that the file can be read and analyzed by a different application of this dialog. In order to ensure that all system logs are available for an analysis, you must activate the option "Run restore automatically before reading". Via the checkboxes Info, Event, Error and Function you can apply a filter to the data and analyze the data systematically.




4.4.7. Reading Memory

In order to offer qualitative support in case of unexpected device behavior, it is possible to read the entire memory via the menu item "Extras". Please use this functionality if detailed information about memory data is required for support.



4.4.8. Configuring Data Storage

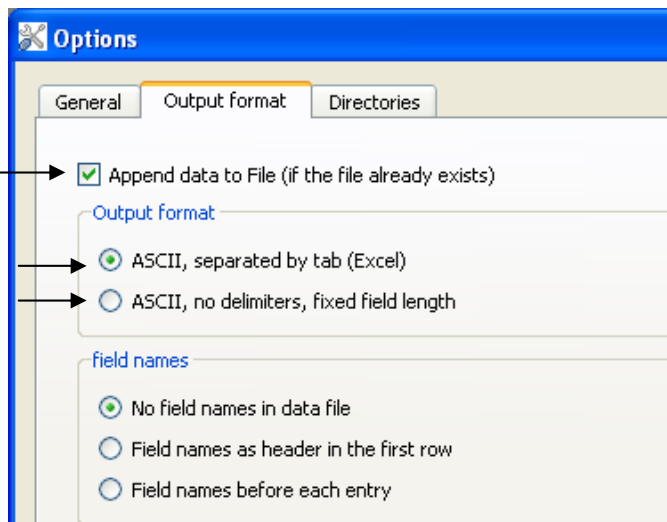
Data which are read from a Datafox device can be stored according to certain specifications.

Call this function via the symbol  or the tab "Configuration – Configure data storage".

Specify whether an existing file is to be overwritten or the data are to be appended.

Determine the output format.

See example 1
See example 2




Caution:

If you want to save the data of several devices in one file, you must activate the option "Append data to file". Otherwise, already existing data are overwritten.

Example 1 (separated with TAB(→))

Data record	Date Time	ID	Name	Label
	2011-09-16 15:38:03 →	00044591 →	JohnPublic →	103 ↵

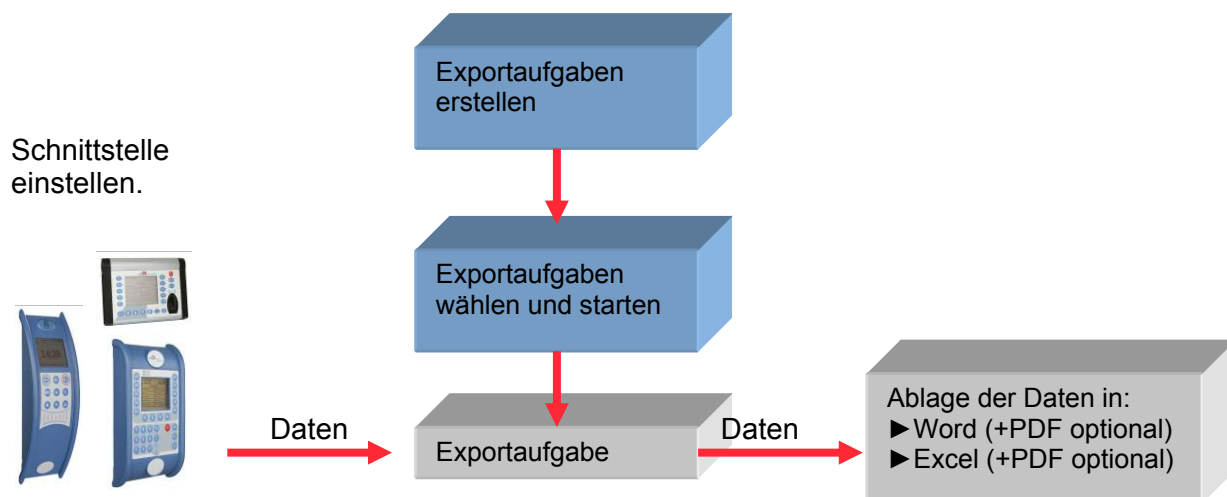
Example 2 (the value is preceded by space characters depending on the data field length)

Data record	Date Time	ID	Name	Label
	2011-09-16 15:38:03	00044591	JohnPublic	103↵
	2011-09-16 15:39:07	00044598	JanePublic	109↵

4.5. Office Connect

4.5.1. Allgemeine Informationen


Office Connect stellt eine einfache Möglichkeit dar, die Daten aus Ihren Datafox MasterIV Geräten für die Einsichtnahme oder die Weiterverarbeitung bereitzustellen. Unterstützt werden die Ausgaben in die üblichen Office-Formate XLS und DOC. Zusätzlich kann eine Protokolldatei im PDF-Format zu jedem Export angelegt werden. Um die Einstellungen der Ausgaben zu speichern, legen Sie Exportaufgaben an. Sie stellen dann nur die Verbindungsparameter zu einem Gerät ein, wählen die Exportaufgabe und starten den Export. Diese legt die Daten aus dem eingestellten MasterIV Gerät in der zuvor gewählten Datei (Word, Excel, PDF) ab.

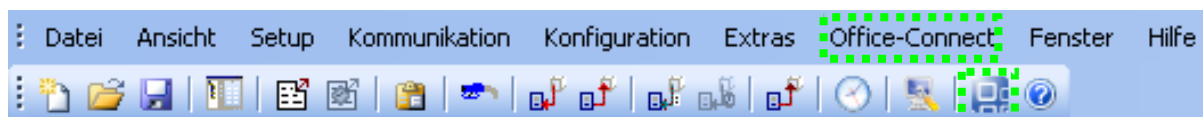


Achtung:

Die Einstellung der Schnittstelle zu dem jeweiligen Gerät muss vor dem Starten einer Exportaufgabe vorgenommen werden.
Die Exportaufgabe bezieht sich nur auf die Art der Ablage der Daten und nicht auf ein bestimmtes Gerät.

4.5.2. Bedienung

Um die Anwendung "Office Connect" zu starten, klicken Sie auf "Office-Connect" oder auf dieses Symbol .



Hinweis:

Für Office-Connect wird eine Mindestversion von Word / Excel 2003 vorausgesetzt.

4.5.2.1. Exportaufgaben erstellen

Eine Exportaufgabe enthält alle Einstellungen für einen Export der Daten von den Datafox MasterIV Geräten.

Das Erstellen und Ändern der Exportaufgaben erfolgt mit Hilfe eines Assistenten.

Die Einstellungen im Assistenten sind weitestgehend selbsterklärend.



Exportaufgabe Bearbeiten



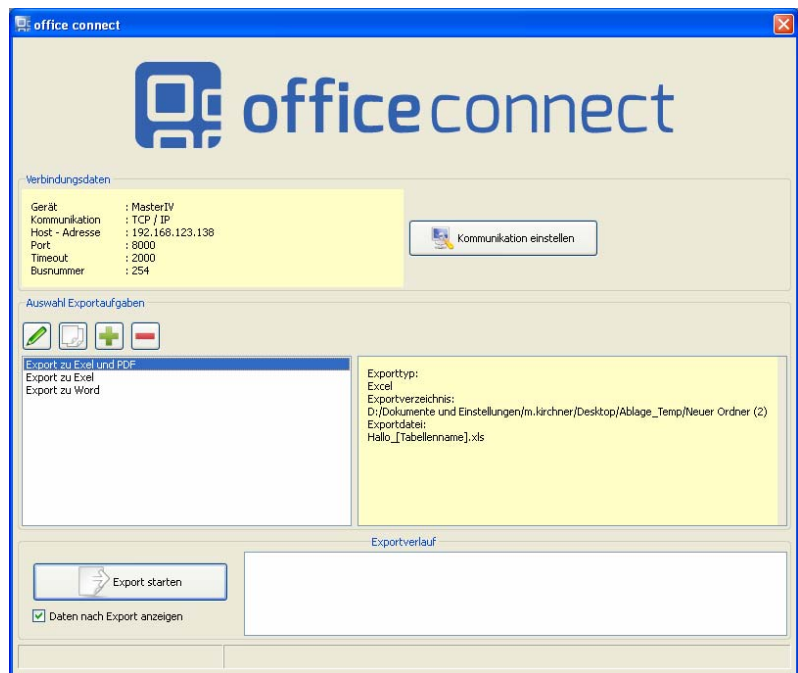
Exportaufgabe kopieren



Neue Exportaufgabe erstellen



Exportaufgabe löschen



Mit einem Klick auf den Button starten Sie den jeweiligen Assistenten.

Im Assistenten können Sie eine Exportvorlage heranziehen. Auf der Produkt-DVD finden Sie entsprechende Vorlagen.

Sie können nach dem ersten Export die Datei bearbeiten und Formatierungen ändern. Wenn dann ein erneuter Export (mit Option Daten anhängen) gestartet wird, werden diese Formatierungen beibehalten. (Spaltenbreite; Textformatierungen; Randeinstellungen usw.)



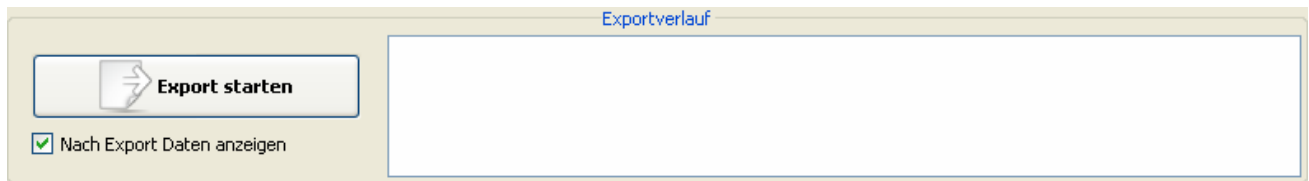
Hinweis:

Sie können in der Dokumentenvorlage bei Exporten vom Typ Word einen Textmarker mit der Bezeichnung „**insert**“ setzen, um zu bestimmen, an welcher Stelle die Daten in dem Dokument gespeichert werden sollen.

4.5.2.2. Export starten

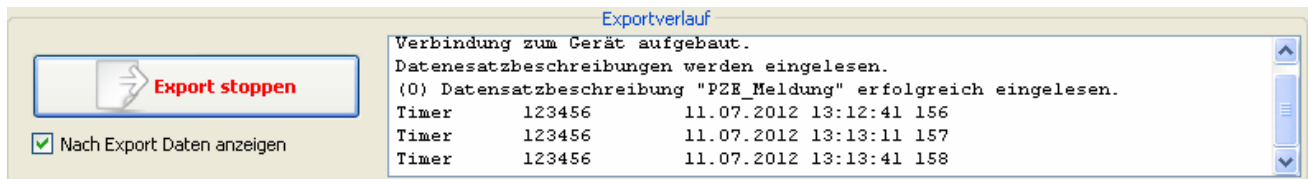
Um einen Export durchzuführen, müssen Sie zuerst die Verbindung zu einem Datafox Gerät herstellen. Sie kennen diesen Dialog aus dem Menü ->Kommunikation ->Einstellungen.

Mit der Schaltfläche „Export starten“ starten Sie eine Exportaufgabe, die Sie im folgendem Dialog aus der Liste auswählen.



Wählen Sie zusätzlich die Option  **Nach Export Daten anzeigen** wird die Exportdatei nach dem Export der Daten geöffnet.

Den Verlauf des Exports können Sie im Feld „Exportverlauf“ anschauen.

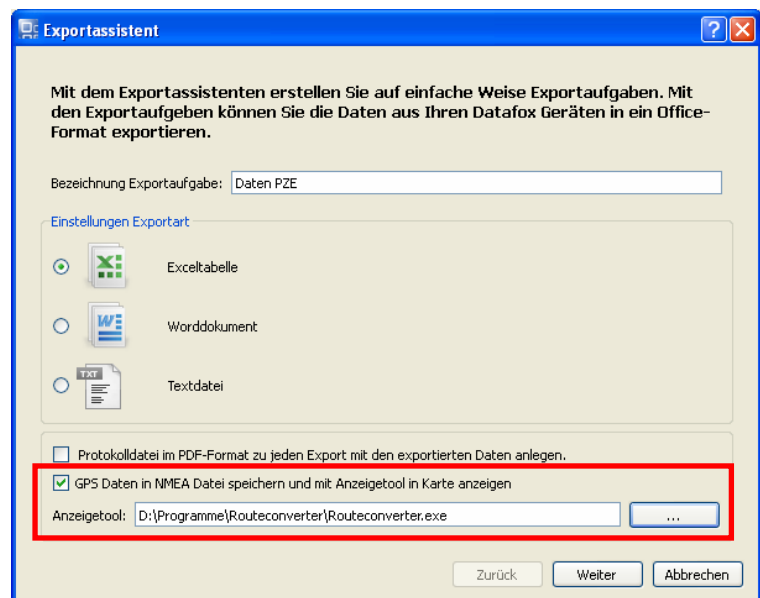


Sie können einen laufenden Export zu jeder Zeit stoppen mit der Schaltfläche „Export stoppen“. Alle Daten, die zu diesem Zeitpunkt bereits exportiert wurden, werden in der Exportdatei gespeichert und der Export wird beendet.

4.5.2.3. GPS-Daten extrahieren und Anzeigen

Sind in Datensätzen GPS-Daten enthalten, kann daraus automatisch eine NMEA-Datei erzeugt werden.

Aktiviert wird dies, mit dem Setzen dieses Häkchens.

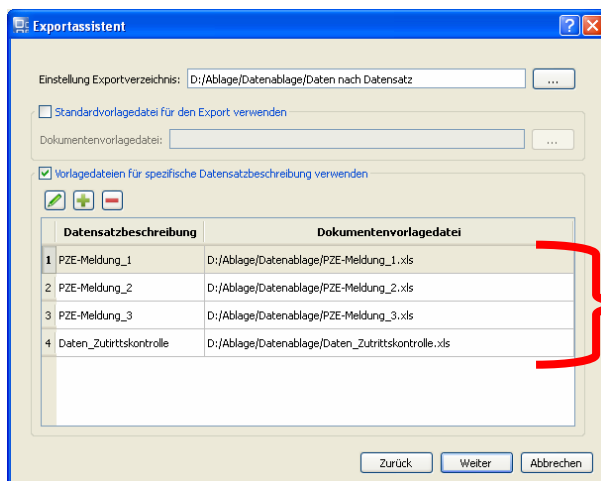
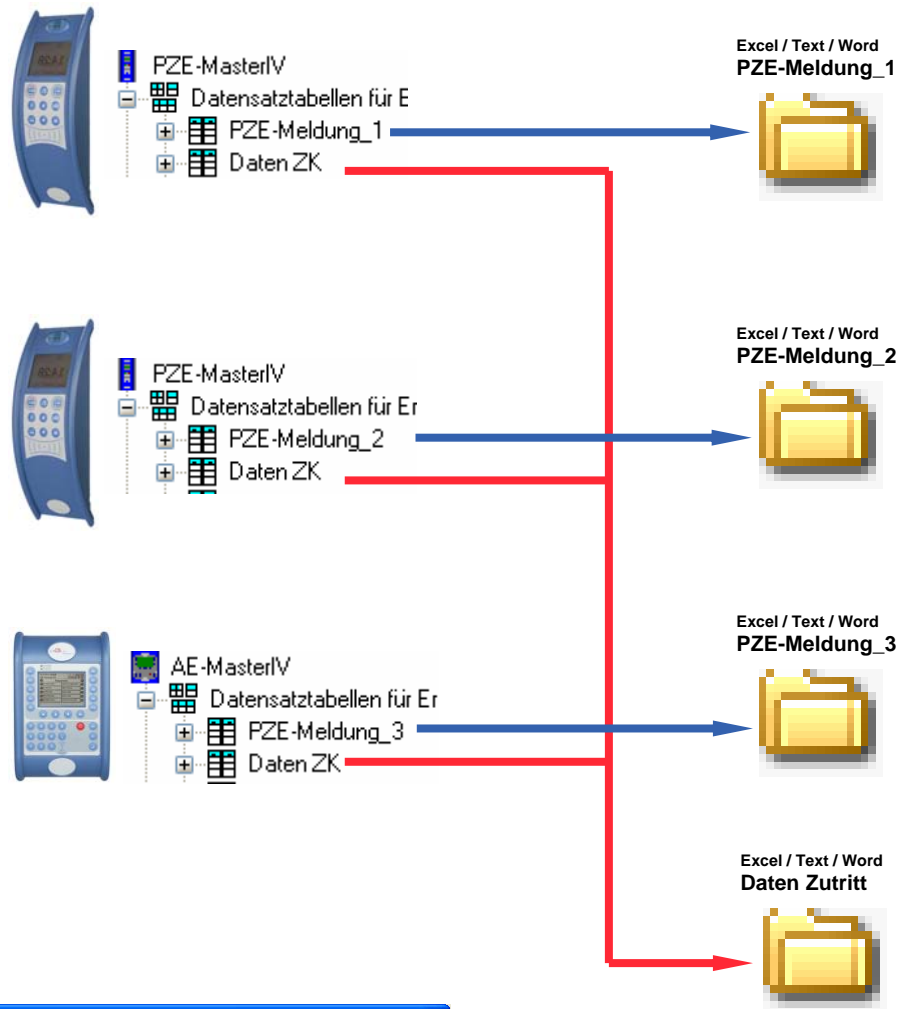


4.5.2.4. Erweiterung für Office-Connect

Bei der Ablage der Daten haben Sie nun die Möglichkeit für jede Datensatzbeschreibung in einem Gerät eine eigene Vorlage zu erstellen bzw. zu wählen.

So können z.B. gleiche Datensätze der selben Datensatzbeschreibung aus verschiedenen Geräten in einer Datei gespeichert werden.

Beispiel:



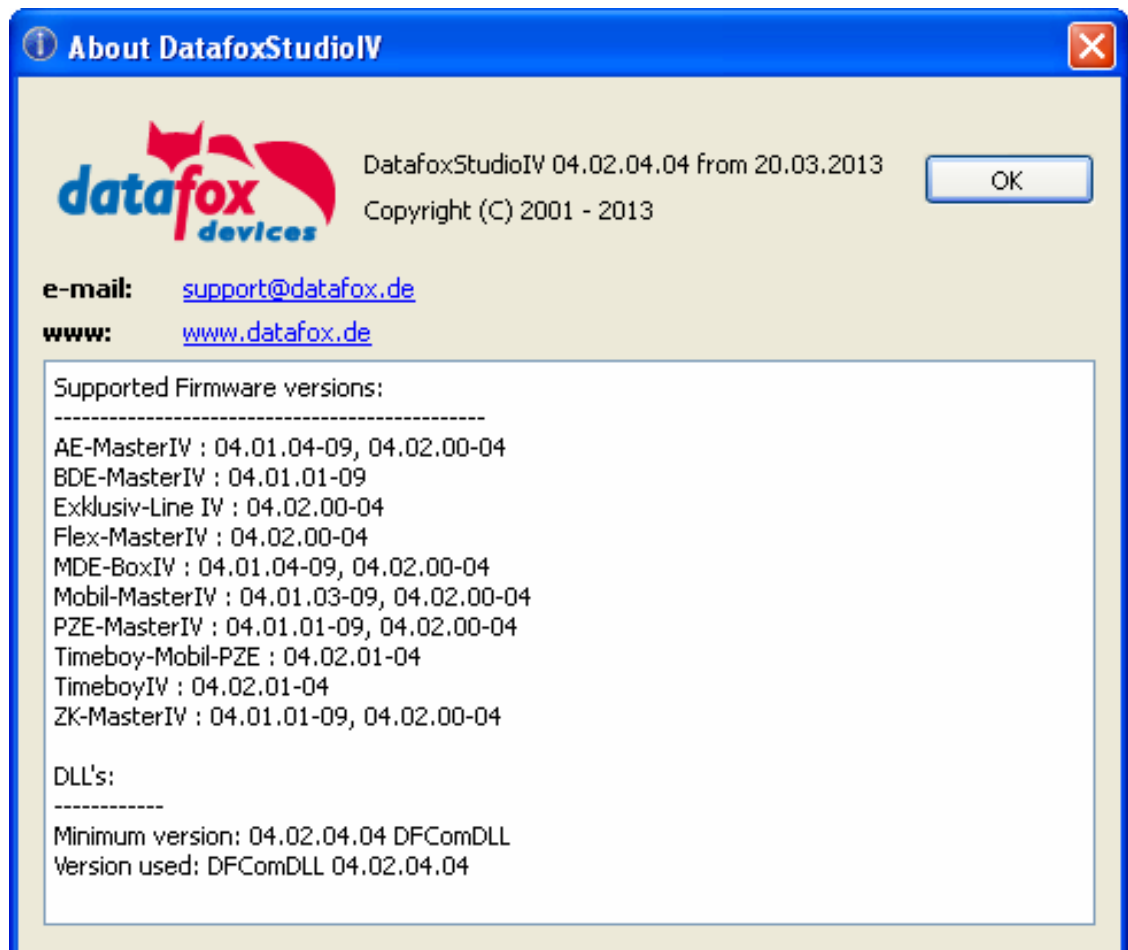
Auf diese Art können in einer Exportaufgabe verschiedene Zuordnungen erreicht werden.

Wird für einen Datensatz keine Vorlage gefunden, wird die Standardvorlage (wenn angegeben) genutzt. Ist keine Standardvorlage angegeben, werden die Daten einfach in ein leeres Dokument geschrieben.

4.6. Help

4.6.1. Information about DatafoxStudioIV

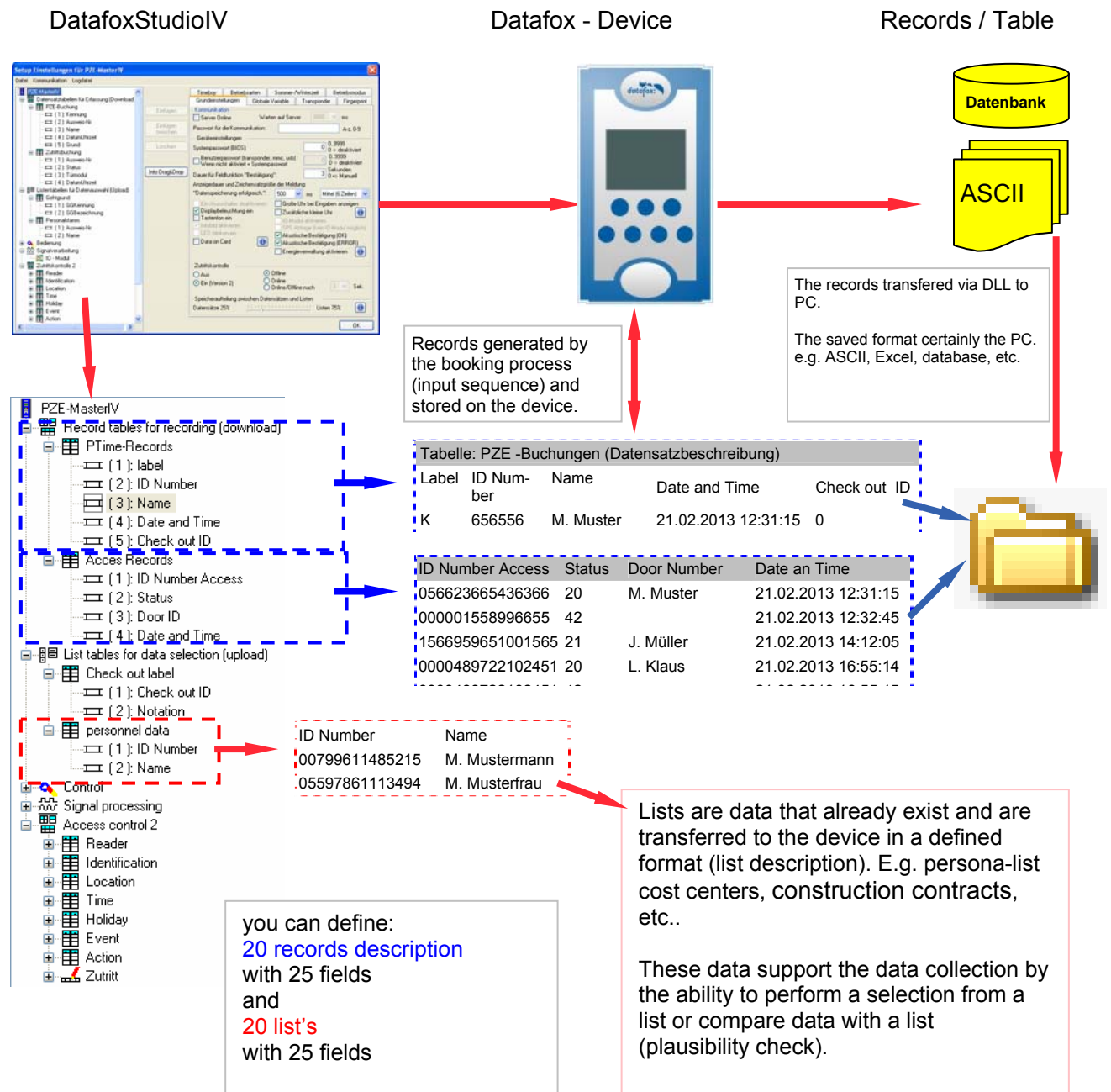
The information dialog window of DatafoxStudioIV shows the software version of DatafoxStudioIV. The build no. is not shown, but the build date. Also the supported firmware and DLL versions are listed.



5. Setup Structure

Before turning the PC on and creating the device setup, you should plan the data recording procedure and thus the setup structure. For this purpose, only a few steps are necessary. If you are well prepared, creating the setup can be done very quickly.

The figure shows the dependencies between parameterization and results data. Further support in form of working models can be found on the Datafox CD.



Planning Steps for Creating a Setup

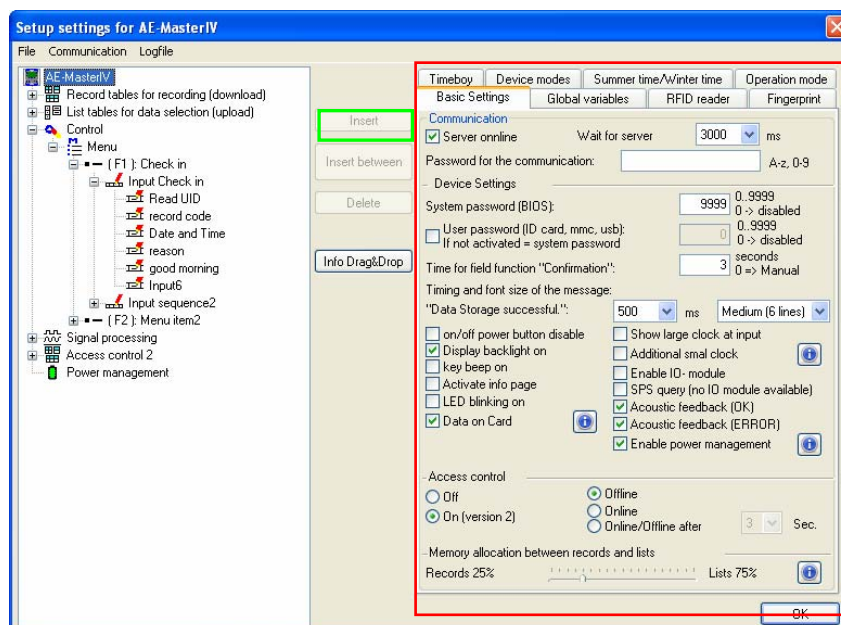
- ▶ Define all tables for the data records to be recorded:
 - Field sequence, field name, field format
- ▶ Set the method of data recording for each field of a data record via the operation (input sequence fields):
 - Bar code, transponder, list, input via keys, constants, global variables, etc. Combinations are possible.
- ▶ If lists are to be used, they must be defined analogous to the data record descriptions:
 - Field sequence, field name, field format
- ▶ The most important step is planning the recording procedure (input sequences of operation). The following questions should be considered, among others:
 - Which is the easiest sequence to enter the fields?
 - Are loops and jump labels necessary?
 - Are global variables necessary?
 - Are dependent lists necessary? For example projects with special activities. After selecting the project, only the corresponding activities are available for selection.
 - Turns the devices off automatically after entering a data record?

The setup must be created in the mask displayed here.

All settings which can be made in the areas marked **red** are explained in the following chapters.

By clicking on a line in the tree structure, the corresponding editing area is displayed.

Via the button "Insert" the tree structure can be expanded.



5.1. Global Settings

All settings made here, apply for the whole setup. Exceptions are some transponder settings and "Server online" which can partially be changed in the input sequences.

5.1.1. Basic Settings

(1) Server online:

Activate the option "Server online" in order to operate the device in dialog mode with a server. Specify the time in milliseconds the device is to wait for a server response. This time is also the indicator when the status of a created data record switches from online to offline.

(2) Password for communication:

You can use a communication password for the device. If it is set, it is requested for each communication. The Studio stores the password temporarily so that you do not have to enter it for each communication. A password is especially important if you want to protect your intellectual property (e.g. unauthorized reading of the setup).

(3) System password (BIOS):

Access to the BIOS menu at the device can be protected by a password so that unauthorized persons are not able to change the settings of the device.

Passwords can be set for the settings and system menu. Thus, a differentiation between user and administrator is possible.

(4) Time for field function "Confirmation":

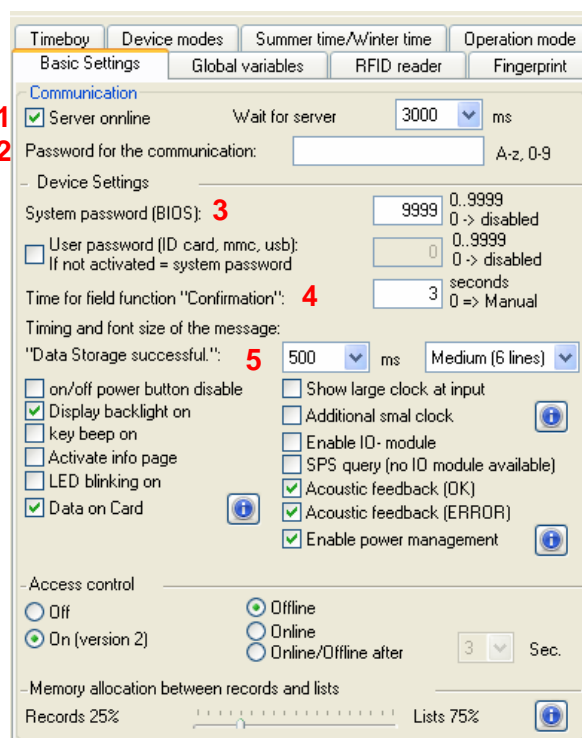
Determine how long a query text is displayed. That means, during the time given, the user can confirm the query text with ENTER or cancel it with ESC.

If the user does not react during the time given, this is interpreted as ENTER.

If a 0 is set in this field, the terminal waits until an input is made.

(5) Data storage successful:

Determine at which position and in which size the message „Data storage successful“ is displayed.



The screenshot shows the 'Basic Settings' window in Datafox Studio. The 'Communication' section is expanded, showing the following settings:

- 1** ☒ **Server online**: Wait for server 3000 ms
- 2** **Password for the communication:** [empty field] A-z, 0-9
- 3** **System password (BIOS):** 9999 (0..9999, 0 -> disabled)
- ☐ **User password (ID card, mmc, usb):** 0 (0..9999, 0 -> disabled)
- 4** **Time for field function "Confirmation":** 3 seconds (0 -> Manual)
- 5** **"Data Storage successful":** 500 ms, Medium (6 lines)

Other settings visible include:

- ☐ on/off power button disable
- ☒ Display backlight on
- ☐ key beep on
- ☐ Activate info page
- ☐ LED blinking on
- ☒ Data on Card
- ☐ Show large clock at input
- ☐ Additional smal clock
- ☐ Enable IO- module
- ☐ SPS query (no IO module available)
- ☒ Acoustic feedback (OK)
- ☒ Acoustic feedback (ERROR)
- ☒ Enable power management

Access control:

- ☐ Off
- ☒ On (version 2)
- ☐ Offline
- ☐ Online
- ☐ Online/Offline after 3 Sec.

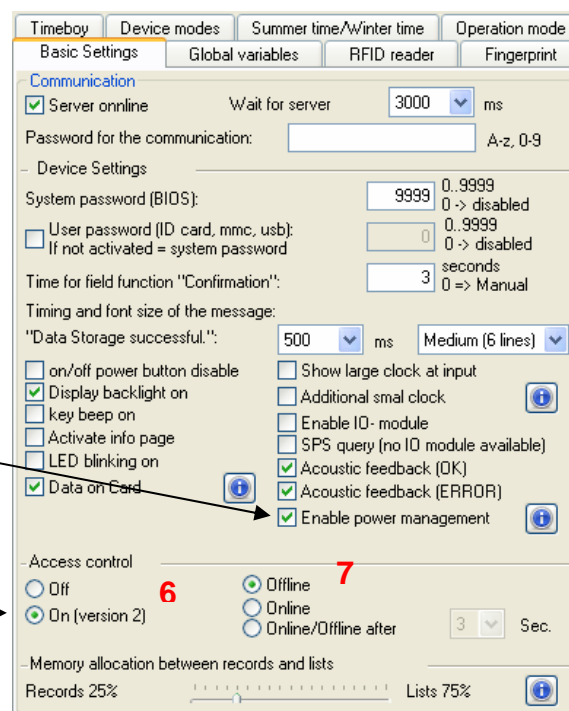
Memory allocation between records and lists:

- Records 25%
- Lists 75%

This area should be self-explanatory.

The device manuals contain a separate chapter concerning power management ("Power Management").

If you want to manage an access control via the device, you must activate it here.



(6) Access control:

If you have activated access control here, you can choose between three operation modes. Offline means that only the access configurations (lists) on the device are used to verify a record (access authorization).

Online means that a record is read by a server application for verifying access authorization. Then, the server application performs the necessary steps (e.g. opening the door). This means that in this operation mode of access control, no configuration data for access control are needed to be available on the device.

Online/ offline after n seconds means that initially a record remains at the device. If the record is not read by the server application, the device performs the verification of access authorization. This means that in this operation mode of access control, the configuration data for access control must be available on the device as well.

(7) Storage arrangement:

Determine how much storage the firmware uses for booking data (data records) and for master data (lists). If you want to ensure that the device is able to create and save data in case of a longer server breakdown without the data being read, it is recommended to increase the memory share for data records.

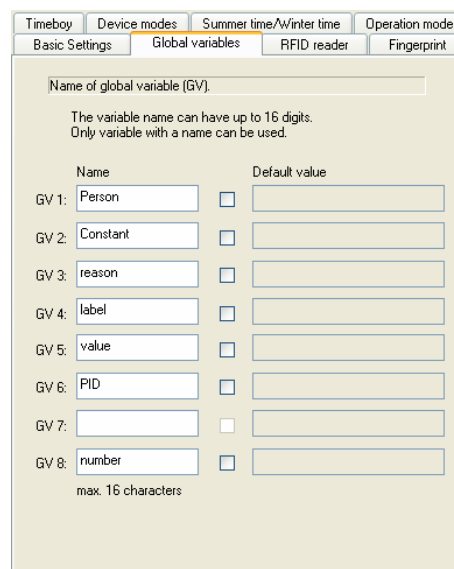
If you work with large lists, e.g. for access control, it is necessary increase the memory share for lists and read data from the device more often.

5.1.2. Global Variables

Global variables can be changed everywhere in the setup, via DatafoxStudioIV or by an application via the DLL.

Ensure that you do not accidentally use GV twice at the wrong place.

You can define up to eight global variables.
It is recommended to provide default values in order to restore a defined state after a restart of the device.
Default values can be modified during the term.



5.1.3. Transponder

Depending on the transponder type selected, an input formula is displayed where the settings for the transponder type are made.

For the most important transponders and their settings see the chapter [„Transponder Methods“](#).

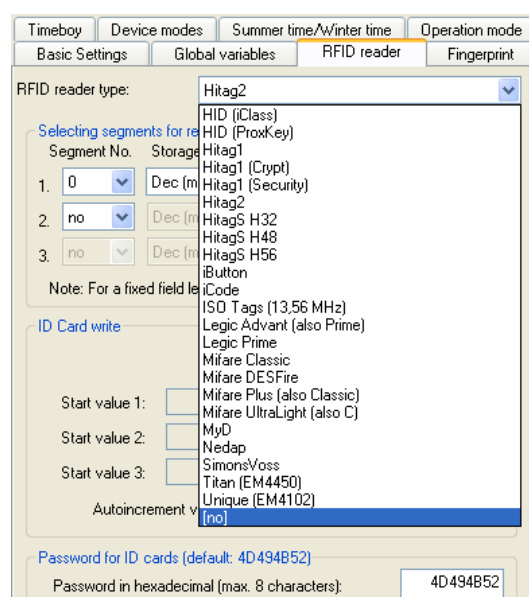


Figure 1

For certain transponders the settings can be modified via input sequences (Figure 2).
Global settings are overridden (Figure 1).

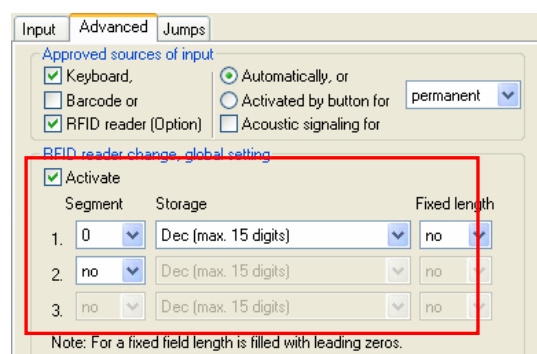
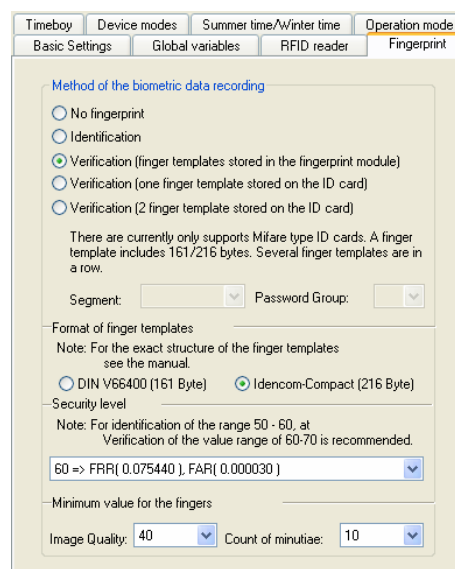


Figure 2

5.1.4. Fingerprint



Activate the method you want to use for validating a record (fingerscan).



► Identification

- The finger characteristics are recorded via the BIO-Key module. Then the data pool is checked for matches. If a match is found, the PID of the person is returned, otherwise an error.

► Verification

- An employee identifies himself via a transponder. The PID (employee number) is read from the ID card. Then the employee has to run his finger over the scanner of the BIO-Key module. In the data pool of the BIO-Key module all primary keys (combination of PID and template) with this PID are determined (up to ten assignments are possible) and checked for matches with the scanned finger characteristics.
- The transponder type additionally supports saving the finger templates on the ID card so that both options are available for this transponder type.

► Segment and password group

- Determine in which segment and with which password group (depending on transponder type) the finger templates are saved on the ID card.

► Format of the finger templates

- We recommend using the Idencom-Compact format because it contains more information and thus enables higher accuracy.

► Security level

- For the recommended settings for this parameter see the dialog. It is a combination of the "False Rejection Rate" = FRR and the "False Acceptance Rate" = FAR.

► Minimum values for finger acceptance

- The minimum image quality specifies in percent the size of the scanned picture, which can be used for determining the finger information.
- The minimum number of minutia defines the number of finger information which must be determined from a scanned picture.



Caution:

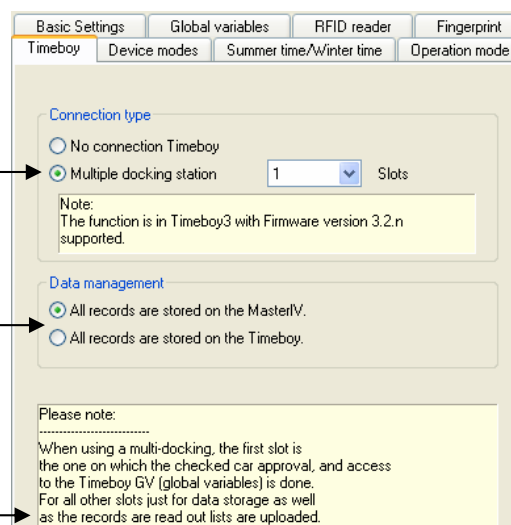
The PID must not exceed the decimal value of 4294967295 ($2^{32}-1$). We recommend working with a 9-place PID.

5.1.5. Timeboy

When connecting a Timeboy docking station, you must specify how many slots the docking station has.

You determine which device is the data sink and responsible for transfer of data (Timeboy or MasterIV).

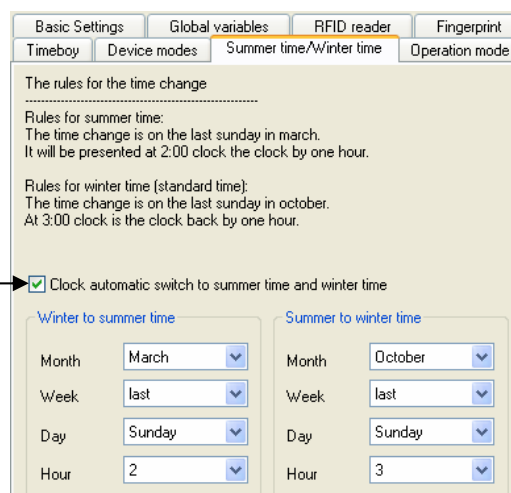
For a multiple docking station observe the information given in this dialog field.



5.1.6. Summer-/Wintertime

If the device is used in a different time zone where a different time model is used for changeover from summer and winter times, the settings can be changed here.

Here, you can activate the automatic changeover.



- Month : The month is set when the changeover takes place.
- Week : Concerns the week of the previously set month.
- Day : Day of the previously set week.
- Hour : Hour at which the clock is set forward or back.

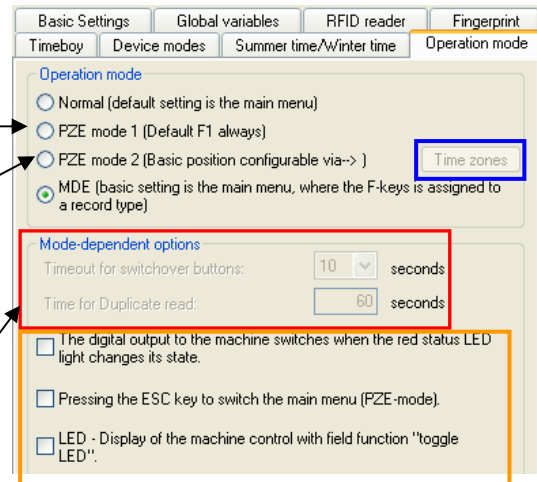
5.1.7. Operation Mode

The operation mode controls the behavior for keyboard switching. With a coming input sequence below the F1 key and a going input sequence below the F2 key, you directly switch to main menu after recording at a device with operation mode "normal".

In PZE mode 1 the F1-key (coming) is always activated, i.e. it is not necessary to press the F1-key. Only the transponder must be held up.

In PZE mode 2, the key preselection can be defined via a [time model](#).

The time for duplicate read only concerns two successive records. If card A is read and the time for duplicate read is set to 60 seconds, card A can be read again within this 60 seconds if a card B has been read in between.



Determine how the optical signaling of the LED should work and how the terminal behaves when pressing the ESC key at PZE mode.

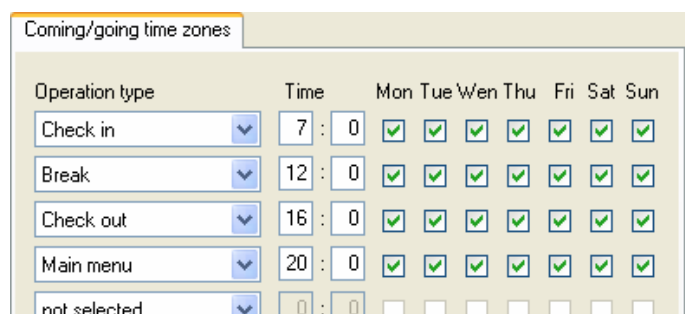
PZE – Mode 2 – Starting the input sequences via a [time model](#).

F1-key, active starting 7:00 am.

F3-key, active starting 12:00 am.

F2-key, active starting 4:00 pm.

Starting 8:00 pm no preselection of function keys.



Operation type	Time	Mon	Tue	Wen	Thu	Fri	Sat	Sun
Check in	7 : 0	✓	✓	✓	✓	✓	✓	✓
Break	12 : 0	✓	✓	✓	✓	✓	✓	✓
Check out	16 : 0	✓	✓	✓	✓	✓	✓	✓
Main menu	20 : 0	✓	✓	✓	✓	✓	✓	✓
not selected	0 : 0							

5.1.8. Operating Type

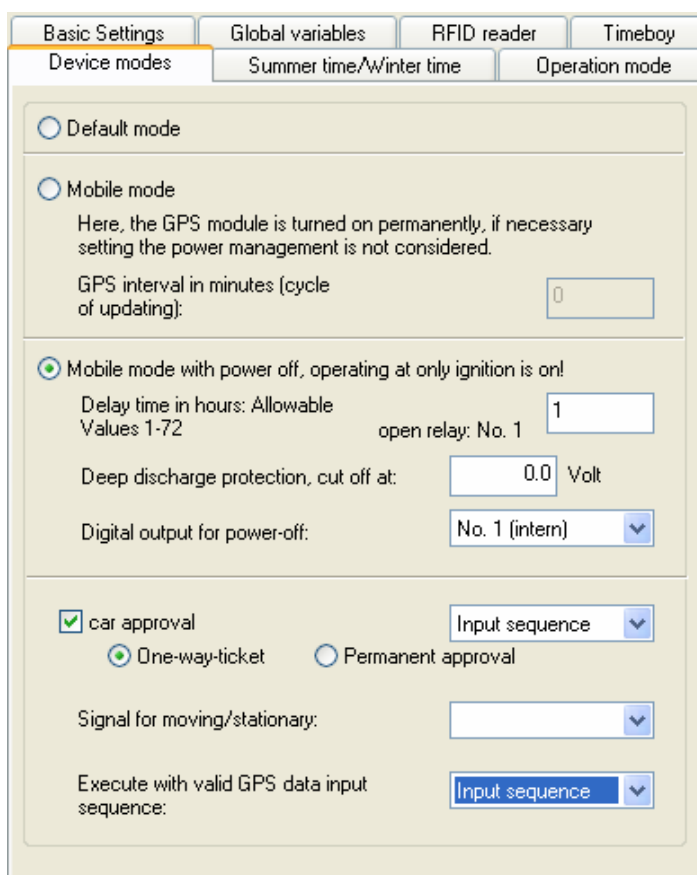
The operating type can only be set for mobile devices. The main purpose is to minimize the energy consumption of the device in order to avoid unnecessarily draining the vehicle battery, for example. In this case you should select the operation type "Mobile operation with power off, operation only if ignition is on!" This way you ensure that the device is not unnecessarily draining the vehicle battery.

The options shown here serve for the operation for commissioning a system (e.g. a vehicle via an explicit car approval).

Determine which digital input is used for detecting start/stop.
Specify in hours the turn-off delay after which the device is turned off.

The deep discharge protection ensures that the vehicle battery cannot be deeply discharged by the device. Specify the lowest voltage value of the vehicle battery at which the device should turn off. Also specify the digital output via which the voltage supply is switched.

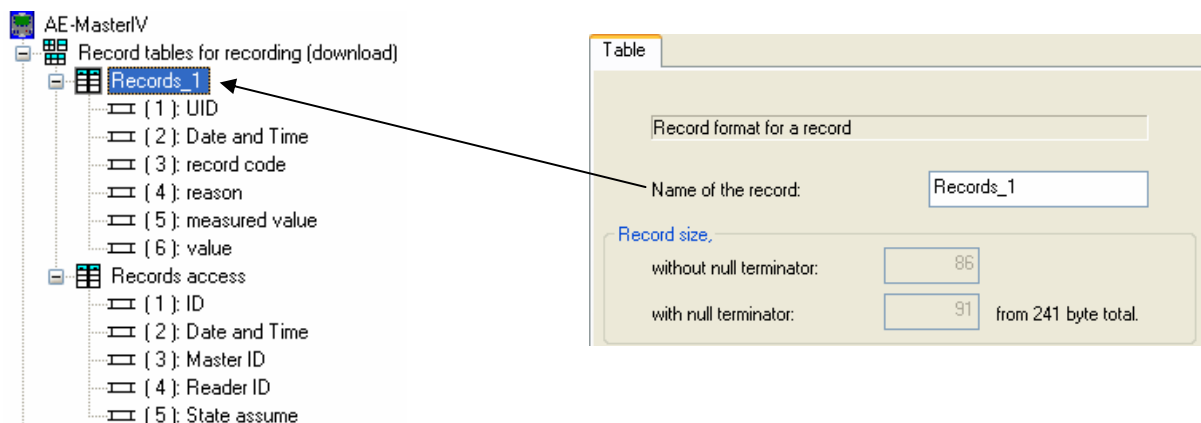
If you want to create a GPS data record cyclically, e.g. for theft protection, specify an input sequence for creating a GPS data record. This input sequence is executed cyclically every 10 minutes provided that valid GPS coordinates could be determined.



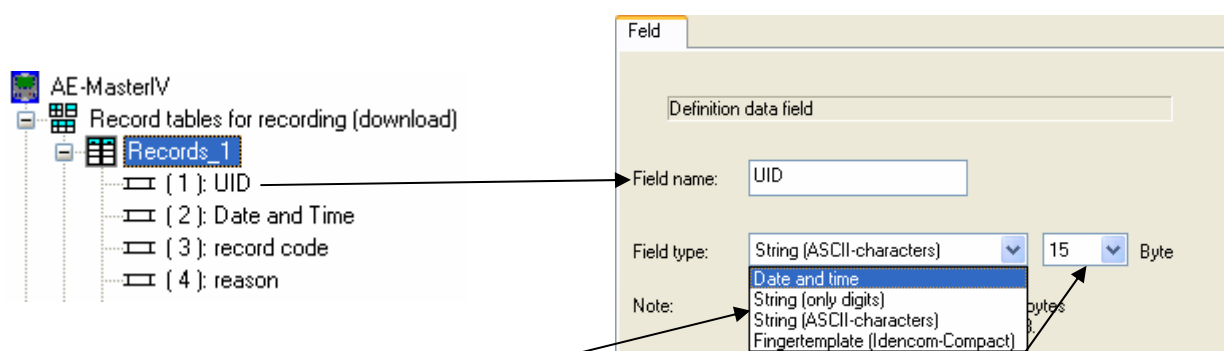
Basic Settings	Global variables	RFID reader	Timeboy
Device modes			
<input type="radio"/> Default mode			
<input type="radio"/> Mobile mode Here, the GPS module is turned on permanently, if necessary setting the power management is not considered. GPS interval in minutes (cycle of updating): <input type="text" value="0"/>			
<input checked="" type="radio"/> Mobile mode with power off, operating at only ignition is on! Delay time in hours: Allowable Values 1-72 <input type="text" value="1"/> open relay: No. 1 Deep discharge protection, cut off at: <input type="text" value="0.0"/> Volt Digital output for power-off: <input type="text" value="No. 1 (intern)"/>			
<input checked="" type="checkbox"/> car approval <input type="text" value="Input sequence"/>			
<input checked="" type="radio"/> One-way-ticket <input type="radio"/> Permanent approval			
Signal for moving/stationary: <input type="text"/>			
Execute with valid GPS data input sequence: <input type="text" value="Input sequence"/>			

5.2. Definition of Data Structures

With defining the data record structures you determine which data are stored in a data record. You determine the sequence of information in the data record, the data type of the single fields and the length of information in a field.



Specify a unique name for each data record description (table). It facilitates the association of a data record description to a booking procedure.



Specify the field type and field length in byte for each data record description.

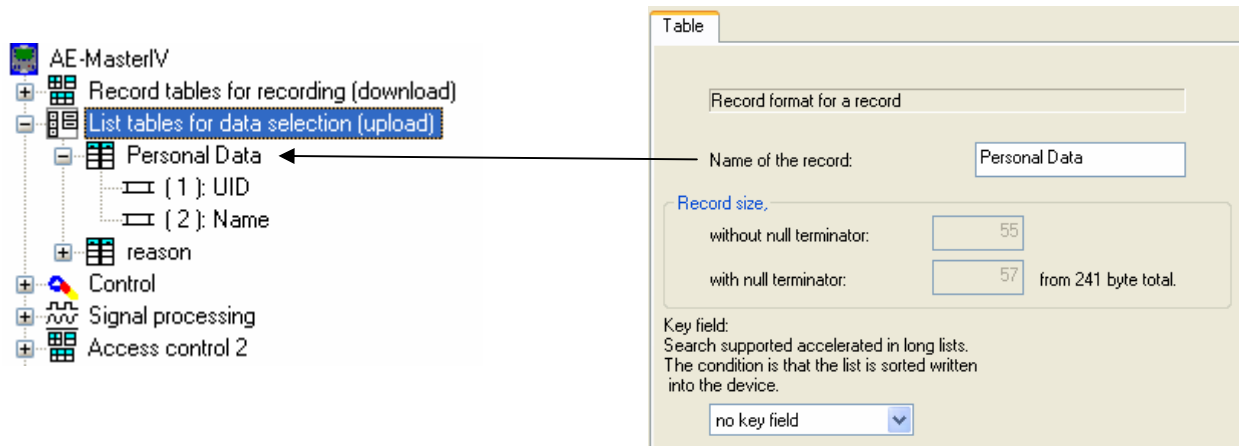
5.3. Definition of Master Data Structure

The master data provide information and data required by the device.

For example: Association of transponder number to a name.

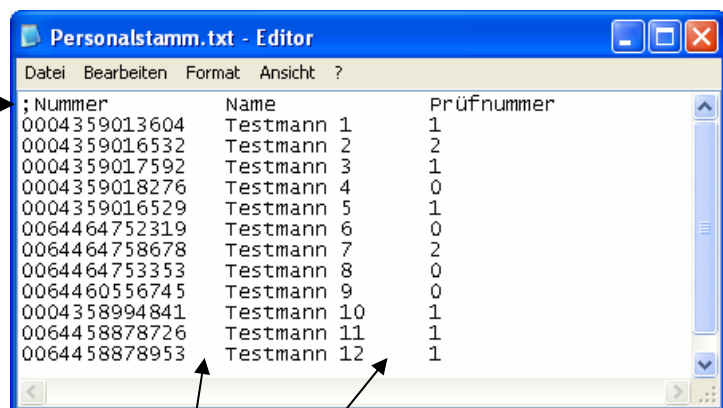
These data are provided in form of lists in a text file. The lists can be created with a simple text editor.

If you create, for example, a staff list with an editor as *.txt (ASCII file), you separate the columns defined, ID card no., name and verification by a tab.



Specify a unique name for each list description (table) and each list field. This facilitates selecting the list during a booking procedure at list selection or writing a list field.

If a line starts with a semicolon, comments can be entered at this place. This helps to structure the text file clearly.



TAB - separated

If you use a list with more than 500 data records, we recommend specifying a key field for sorting the list. This leads to a faster list selection (search in a long list).

5.4. Recording Procedures of Operation

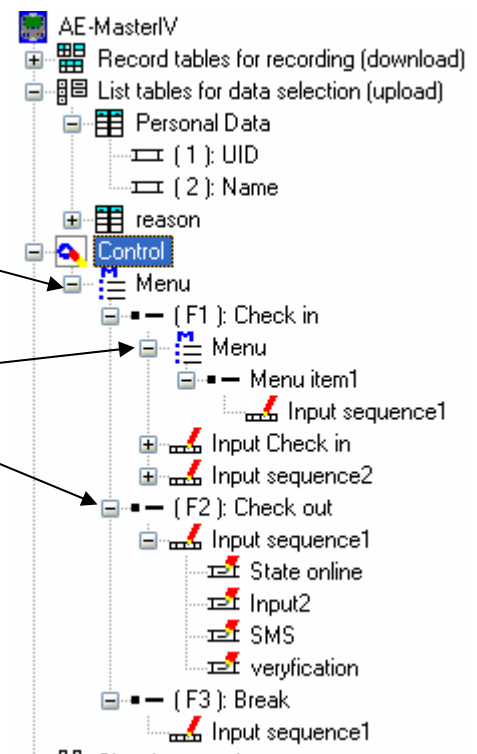
Recording procedures are the engine of a setup. If no recording procedures are defined, nothing is going to happen. Recording procedures are generally referred to as input sequences in the context of a device setup.

5.4.1. Structure of Operation

The central entry point is always the main menu. Below the main menu are the menu entries.

Below a menu entry a submenu can be created or an input sequence be defined.

Below the main menu are the menu entries. In this first structure level, the menu entries can be assigned to the function keys of the keyboard.



In the entire structure of operation, three menu levels with respective menu entries and input sequences are possible.

5.4.2. Starting an Input Sequence

Input sequences of operation are generally started by a keyboard event. For example, the user presses the F1 key (coming) and the first input sequence below the F1 key is started. Also several input sequences can be defined below an F key. At this variant, the number of entry points is restricted by the number of freely configurable F keys on the keyboard of the device type.

Another possibility for starting an input sequence is the operation mode. You can determine via the operation mode "PZE Mode 1" that the device in basic status always waits for an input in the first input sequence below the F1 key.

The third possibility for starting an input sequence is via time zones. The operation mode "PZE Mode 2" must be activated. You can determine with up to 15 time models when the device is to await an input in which input sequence. This offers the possibility to start input sequences which are not assigned to a physical F-key of the keyboard. An example for a PZE-MasterIV with its five physical F-keys would be an F6 chain for an Alive data record.

F1-key, active starting 7:00 am.
 F3-key, active starting 12:00 am.
 F2-key, active starting 4:00 pm.
 Starting 8:00 pm no preselection
 of function keys.

Coming/going time zones			Mon	Tue	Wen	Thu	Fri	Sat	Sun
Operation type	Time								
Check in	7 : 0		✓	✓	✓	✓	✓	✓	✓
Break	12 : 0		✓	✓	✓	✓	✓	✓	✓
Check out	16 : 0		✓	✓	✓	✓	✓	✓	✓
Main menu	20 : 0		✓	✓	✓	✓	✓	✓	✓
not selected	0 : 0								

These mechanisms only work for input sequences in the first structure level as entry point directly below a menu entry.

The forth possibility to start an input sequence directly from the main menu is a read barcode or transponder. In this case the value read must start with EKxx. The xx stands for the number of the function key selected from the operation structure.

Menu	
Line 1 of the display (fix):	Menu
Text of line 1 on GV (dyn.):	Line 1 of the display (fix) ▼
Line 2 of the display (fix):	
Text of line 2 on GV (dyn.):	Line 2 of the display (fix) ▼
Attention! Selection of lines of ad (text or GV) is supported from Firmware version 04.01.07.x.	
<input type="checkbox"/> Entries MasterIV two columns show. The key F6 - F10 are shown in the right half of the display.	
Select menu items by barcode/ID card.	
<input checked="" type="radio"/> Don't use.	
<input type="radio"/> Select menu items by barcode.	
<input type="radio"/> Select menu items by ID card.	
Residual value at GV store:	not selected ▼
In GV the value without the intent EKxx is saved!	
In GV the value without the intent EKxx is saved! The barcode/ID card values must begin with EKxx, then xx is the number for the selected function key. Not with "access check in the main menu" combined!	

5.4.3. Configuring an Input Sequence

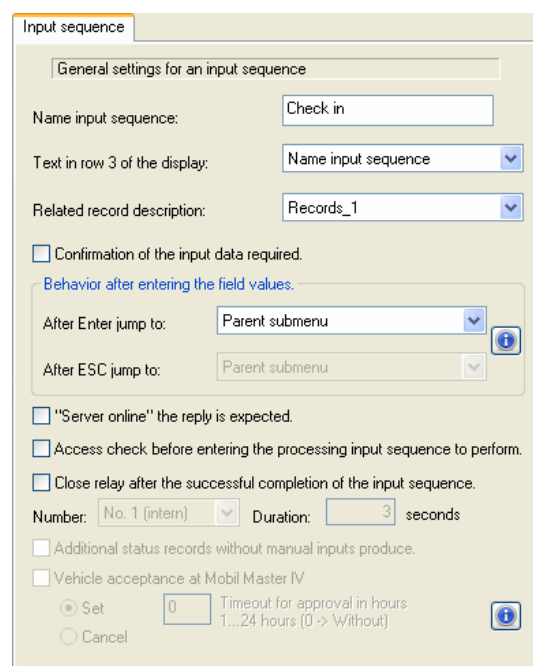
An input sequence contains parameters influencing the execution of the input sequence and the behavior of the device after the input sequence is completed.

By allocating a data record description you determine in which structure and format the entered data are to be saved.

Determine whether the user has to confirm the entered data before the input sequence is completed. If you activate this option, you must also specify where to branch if the input is cancelled with ESC. You also must determine where to branch if the input sequence is (successful) completed with ENTER.

Further attributes of the input sequence are the execution of two functions. The function "Access check before processing the input sequence" only refers to the internal transponder reader of the device. That means that the first expected event is a transponder. After successful access check, the input sequence is started and a transponder expected again, provided that a transponder is to be processed within this input sequence. In order to avoid reading the transponder twice, you can use the field function "Perform access check with GV".

The second function as attribute of the input sequence is "Close relay after successful completion of the input sequence". This function can only be executed for one of the internal relays for a duration of n seconds.

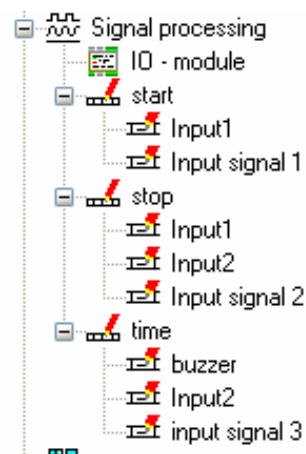


After defining the attributes of the input sequence, you must determine the single processing steps, i.e. in which sequence are which data to be created (entered), edited or verified. Each processing step is described in one or several input sequence fields. For a detailed description of the single field functions see section "[Functions in the Device Setup](#)".

5.5. Recording Procedures of Signal Processing

In the section Recording Procedures of Operation, it has been described how input sequences (recording procedures) can be started by the user by pressing a key.

In signal processing, this is performed automatically. As the name implies, signal processing plays a crucial role. Signals can act upon the digital and analog inputs of a device. Such external signals automatically trigger input sequences. Another kind of signals are internal signals in the form of timer events. A further kind of signals are GPS coordinates via the internal module.



5.5.1. Structure of Signal Processing

As stated before, for signal processing exist several possibilities to detect signals and use the information (data) of these signals for starting an input sequence. At this process the data of the signals are saved within the started input sequences in the data records.

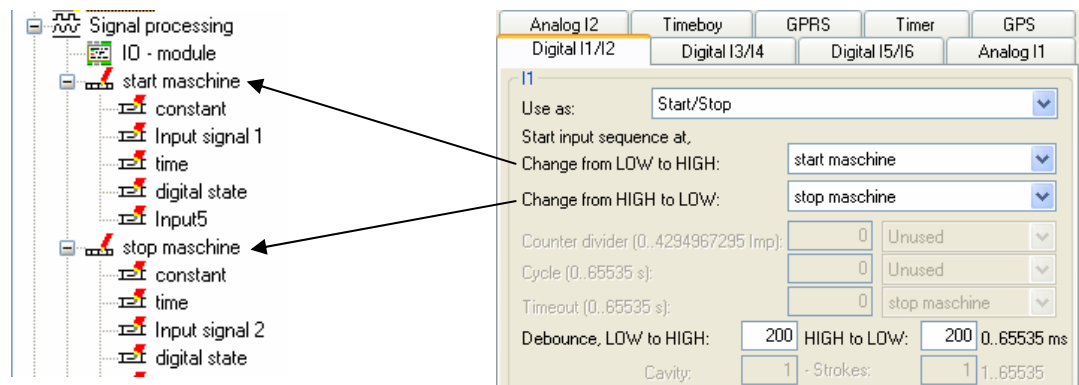
5.5.1.1. Digital Inputs

Digital inputs are used to process two-valued information (logic 0 and logic 1). A voltage of 0 – 1 V at the first digital input corresponds to a value of logic 0. A voltage of 3.5 – 30 V corresponds to a value of logic 1. For the other digital inputs a voltage of 0 – 3 V corresponds to logic 0 and a voltage of 12 – 30 V corresponds to logic 1. The voltages given define what a valid signal at a digital input is.

Application of the Digital Inputs

► Start/Stop

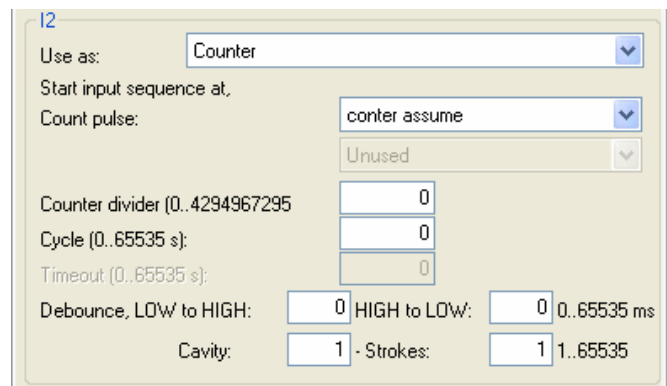
- The input sequence "Start machine 1" is executed at a starting event (in this case switch from LOW to HIGH).
- When switching from HIGH to LOW the input sequence "Stop machine 1" is started.



► Counter (a valid signal at the digital input is logged)

An input sequence "impulse counter" is started after reaching the value set in the counter divider.

If a cycle time is set, the input sequence is started after this time. 0 = no cycle.



Debouncing times define a duration during which a signal level must act steadily upon a digital input in order that this signal level is interpreted as valid impulse.

If the debouncing time is set to zero, the input reacts on every impulse. The sampling rate for this is at 5 kHz. (See technical appendix of the devices.)

Cavity determines how many tasks are necessary for producing a part.

The **number of strokes** determines how many tasks can be executed simultaneously.

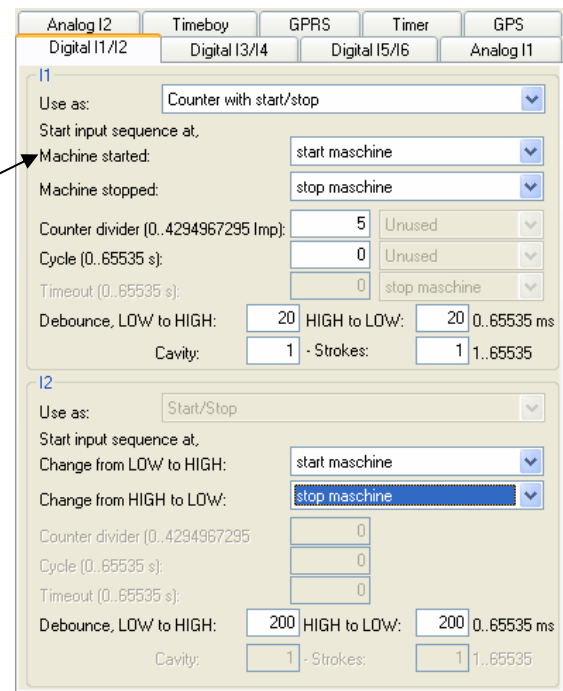
Counter with Start/Stop

This function combines the two preceding ones. E1 and E2 are kind of inter-linked.

Only after the machine is started, the impulses at E1 are analyzed.

An input sequence is started when:

- The machine is turned ON.
- The machine is turned OFF.
- The value of the counter divider is reached.
- The cycle time is expired.



► Counter with Start/Stop per Timeout

The counter for produced parts is at the same time the signal for a running machine. In order to detect an interruption or shutdown of the machine, the timeout is monitored. If the timeout has expired but no counting impulse has been registered, an interruption data record is created. The timeout is reset at each impulse.

► Counter with Start/Stop and 1. Pulse per Timeout

The input sequence allocated to the 1. impulse and counting impulse is executed at the first impulse (switching from Low to High) and afterwards executed according to the counter/divider. The counter/divider determines how many impulses are necessary for triggering the allocated input sequence. At 0 the input sequence is not executed. If no new impulse is detected after the last, impulse within the duration of the timeout the input sequence for the timeout is executed. A following impulse then executes the input sequence for the 1. impulse.

Caution: Setting the debouncing times is a task for the installer and therefore lies within their responsibility.

General Notices for Debouncing:



Normally, bouncing only occurs at mechanical contacts. Digital signals do not bounce and thus do not lead to miscounts even without debouncing. Mechanical contacts are not designed for several switchings per second so that you should expect significantly longer debouncing times. In order to debounce short and fast signals, a solution via hardware is necessary, e.g. by parallel connection of a capacitor.

5.5.1.2. Analog Inputs

The analog inputs are used for time and value-continuous signals. The measurement range of an analog input is 0 – 10 V.

When using **measuring value** you must specify the identifiers, the physical quantity to be measured or the measuring process as well as the respective units, e.g. liter or fuel level in l.

You also must specify the voltage range of the encoder connected as minimum and maximum input voltage. Furthermore, specify the measurement range of the encoder according to the manufacturer as min./max. measurement value.

Note: Measuring range from 0 to 10V.

Use as: Measuring value

☐ Show value on info page. Sampling: 200 msec.

Identifier: a+1 => Unit V

Max. Measurement value: 10 => Input Voltage 10

Min. measurement value: 0 => Input Voltage 0

Derogation limits +/-: 0.1 % => +/- 0.01 V

When using **measuring value with threshold value test**, you must additionally specify the threshold values (limits). This encompasses 5 zones: Starting from below, the lower alarm limit, the lower warning limit, the NORMAL zone, the upper warning limit and the upper alarm limit.

If a warning or alarm limit is reached, the corresponding input sequence is started.

Analog I2
Timeboy
GPRS
Timer
GPS
Digital I1/I2
Digital I3/I4
Digital I5/I6
Analog I1

Note: Measuring range from 0 to 10V.

Use as: Measuring value with threshold value test

☐ Show value on info page. Sampling: 200 msec.

Identifier: a+1 => Unit V

Max. Measurement value: 10 => Input Voltage 10

Min. measurement value: 0 => Input Voltage 0

Derogation limits +/-: 0.1 % => +/- 0.01 V

To be monitored limits

Information on the status value of the limit value is exceeded

Limit => Voltage => assigned input sequence

Upper alarm limit: 9 => alarm high

Upper warning limit: 8 => warning high

Lower warning limit: 2 => Unused

Lower alarm limit: 1 => warning low

alarm low

Example:

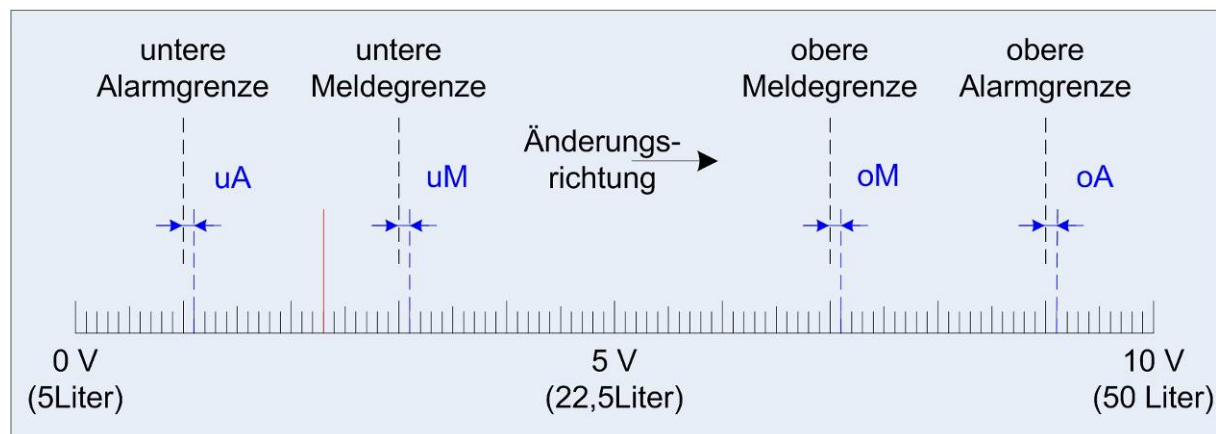
In order to illustrate the effect of the limit deviation, we assume the following scenario. The fuel level is to be monitored; the encoder works within a voltage range of 0 to 10 volts and has a measurement range of 5 to 50 liters.

10 V = 50 Liter = 100 %

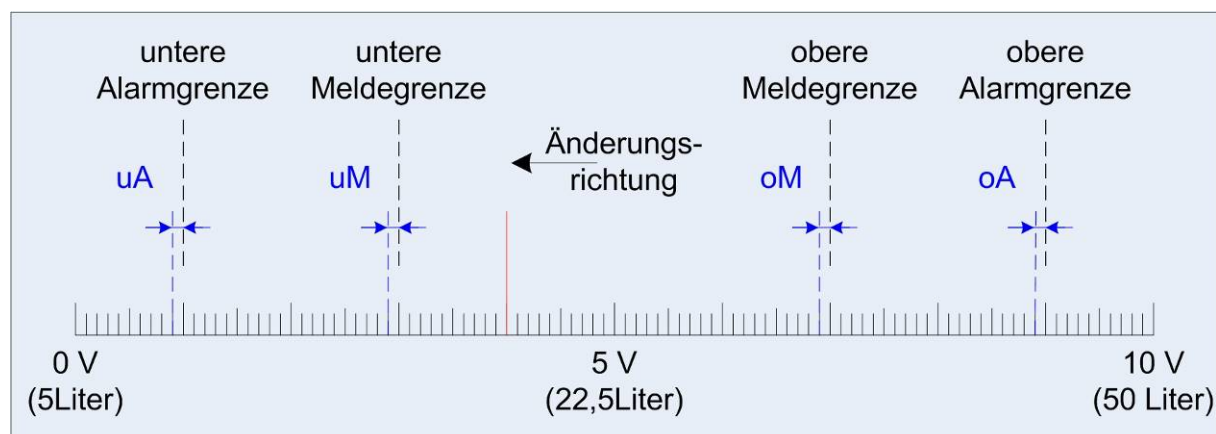
\updownarrow 45 Liter = 100% Grenzwertabweichung von 1% = 0,45 Liter (0,1 V)

0 V = 5 Liter = 0 %

The direction of the change in the following figures shows whether the threshold value is exceeded or falls below it. In the following figure for a change of the fuel level towards the normal value, the voltage level of the encoder must increase to 3.1 V (13.95 liters) - i.e. exceed the set threshold value for the lower warning limit by 0.1 V (0.45 liters) - in order to create a data record for the normal zone.

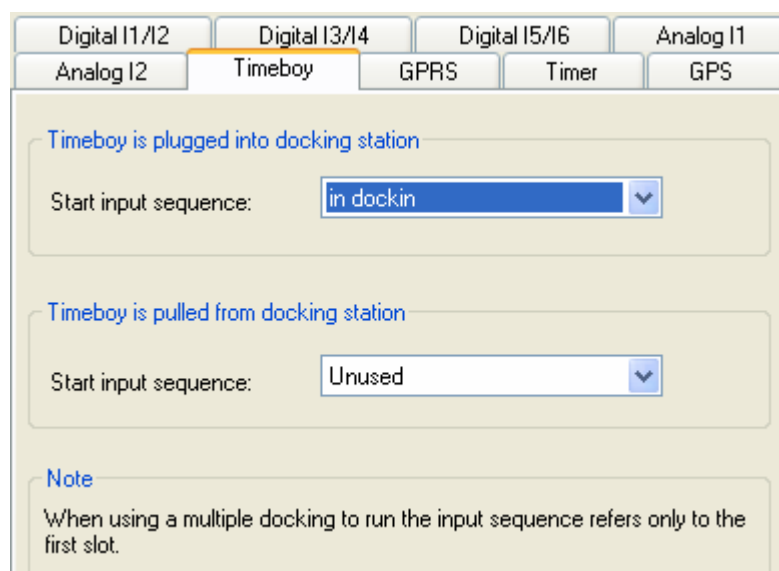


For a change of the fuel level below the lower warning limit, the voltage level of the encoder must decrease to 2.9 V (13.05 liters) - i.e. falls below the set threshold value for the lower warning limit by 0.1 V (0.45 liters) - in order to create a data record for the lower warning limit.



5.5.1.3. Timeboy Events

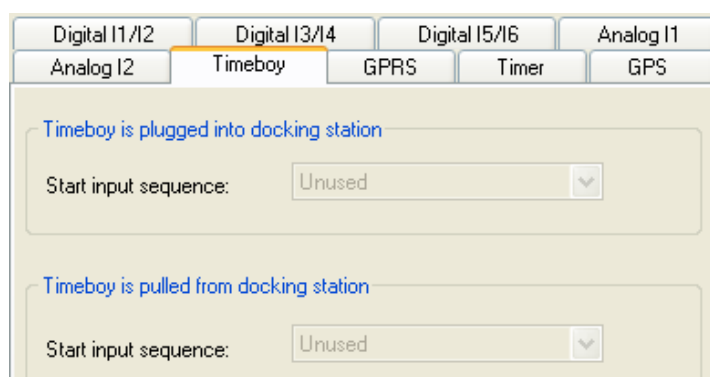
Two events are possible if a Timeboy is to be connected to a device, e.g. in order to read the data from the Timeboy and sent them to a central processing server via the device. Firstly, the plug-event when the Timeboy is plugged into the docking station connected to the device. Secondly, the pull-event when the Timeboy is pulled out from the docking station connected to the device. To each of these events, an input sequence can be allocated which is executed automatically when the event occurs.



The screenshot shows the 'Timeboy' tab selected in the configuration window. The 'Timeboy is plugged into docking station' event has the 'Start input sequence' set to 'in dockin'. The 'Timeboy is pulled from docking station' event has the 'Start input sequence' set to 'Unused'. A note at the bottom states: 'When using a multiple docking to run the input sequence refers only to the first slot.'

If the area is grayed out, the Timeboy connection must be activated in the global settings first.

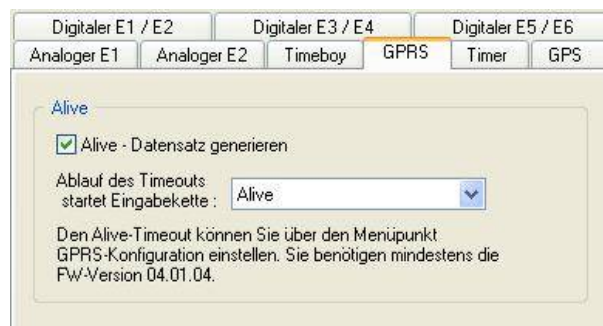
See chapter [“Global Settings – Timeboy”](#).



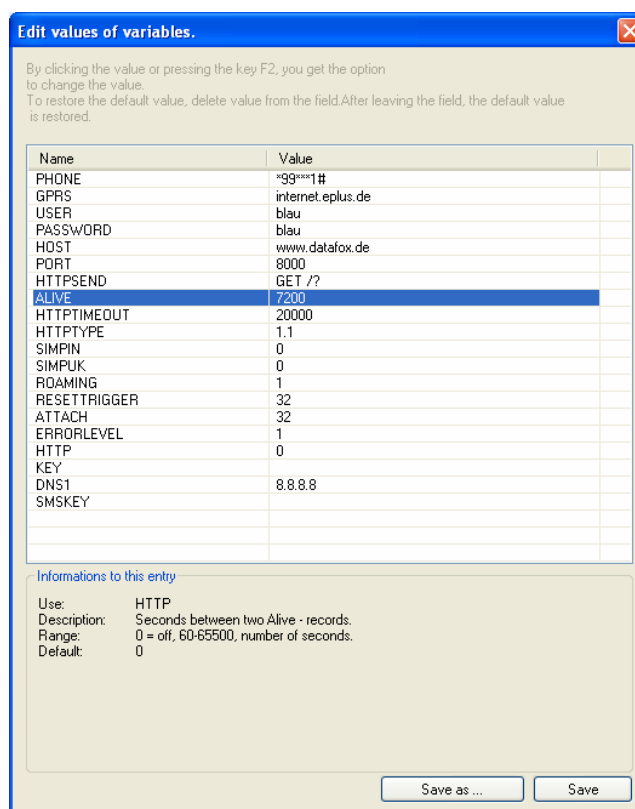
This screenshot shows the same configuration window, but the 'Timeboy' tab and its associated settings are grayed out, indicating that the Timeboy connection is not yet activated in the global settings.

5.5.1.4. GPRS-Alive

If, for example, a stationary terminal is installed autarchically far from any infrastructure (except mobile communications) and the terminal is difficult to access for a service engineer, you can make the device create an Alive data record cyclically. After creation, the Alive data record is immediately sent to a server. The Alive counter gives you information on how many Alive data records could not be sent. The Alive counter is reset to 0 only if the data record has successfully been sent. Otherwise, it is increased by 1 for each unsuccessful attempt and the unsent data record is deleted.



In both cases (Alive data record via operation or signal processing) the Alive parameter must be configured in the GPRS.ini. You must specify a value higher than 60 seconds. When configuring the parameter, keep in mind that sending Alive data causes costs. Therefore, we recommend setting the value not to low.



Name	Value
PHONE	*99***1#
GPRS	internet.eplus.de
USER	blau
PASSWORD	blau
HOST	www.datafox.de
PORT	8000
HTTPSEND	GET /?
ALIVE	7200
HTTPTIMEOUT	20000
HTTPTYPE	1.1
SIMPIN	0
SIMPUK	0
ROAMING	1
RESETTRIGGER	32
ATTACH	32
ERRORLEVEL	1
HTTP	0
KEY	
DNS1	8.8.8.8
SMSKEY	

Use: HTTP
 Description: Seconds between two Alive - records.
 Range: 0 = off, 60-65500, number of seconds.
 Default: 0



Caution:

Alive data are temporary data. If the Alive data record cannot be sent (e.g. server is not reachable), it will be deleted and the Alive counter will be increased by one. The function "Alive" is activated via the Alive parameter in the GPRS.ini. Additionally to the activation, the GPRS chain has to be available in the signal processing. Take care that this function does not create unintentional data (traffic).

5.5.1.5. Timer Events

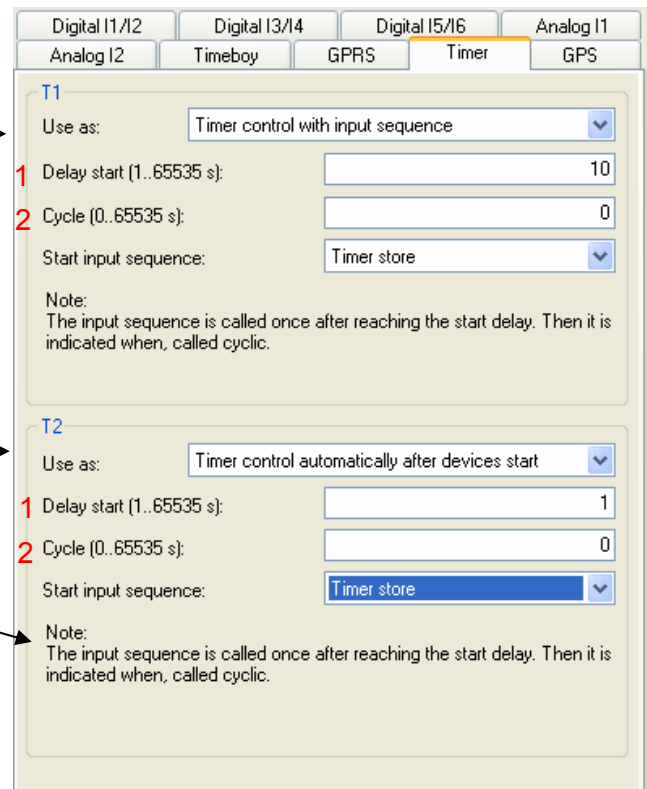
Two timers are available.

The timer event of signal processing can be used for a single or cyclical execution of an input sequence. The timer can be started wither via an input sequence or automatically after starting the device.

Start and stop timer by the field function "Start/stop timer" in an input sequence.

If the timer is restarted in an input sequence, it is reset to zero, i.e. the set time starts again.

Start timer automatically after starting the device.
See Note:



T1

Use as: Timer control with input sequence

1 Delay start (1..65535 s): 10

2 Cycle (0..65535 s): 0

Start input sequence: Timer store

Note:
The input sequence is called once after reaching the start delay. Then it is indicated when, called cyclic.

T2

Use as: Timer control automatically after devices start

1 Delay start (1..65535 s): 1

2 Cycle (0..65535 s): 0

Start input sequence: Timer store

Note:
The input sequence is called once after reaching the start delay. Then it is indicated when, called cyclic.

1) The starting delay specifies after which delay time the timer event is triggered after the timer has been started.

2) A cycle value of 0 determines that the timer event is executed once after starting the timer. If the timer event is to be executed again, the timer must be restarted.

By triggering the timer event, the corresponding input sequence is triggered.

5.5.1.6. GPS Events

The processing of GPS data has been extended for the devices AE-, Mobil- and PZE-MasterIV. Now input sequences for the following events can be started on the GPS page at signal processing in the setup: Start, stop, cycle, change of direction and moving.

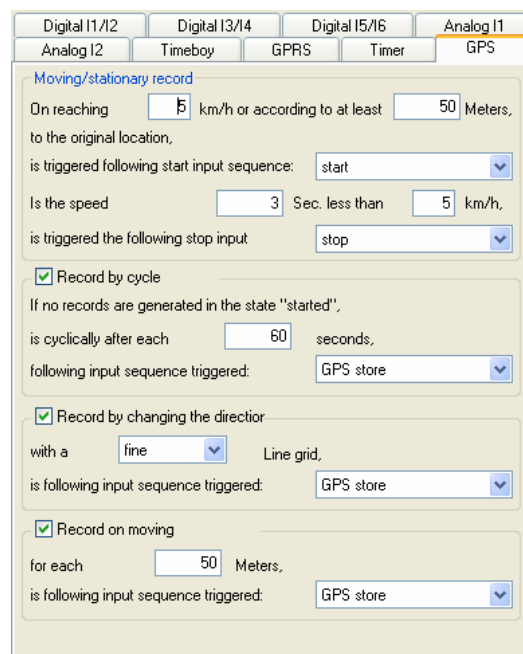
The input sequence "Start" is started when reaching a velocity of 5 km/h or after driving 50m, depending on which event occurs first.

The input sequence "Stop" is started if the velocity falls below 5 km/h for 3 seconds.

A data record is created every 60 seconds.

Creating data record depending on distance.

Depending on the settings (fine, medium, coarse) a data record can be created. The following table provides an overview of the settings.



Settings Overview for Change of Direction

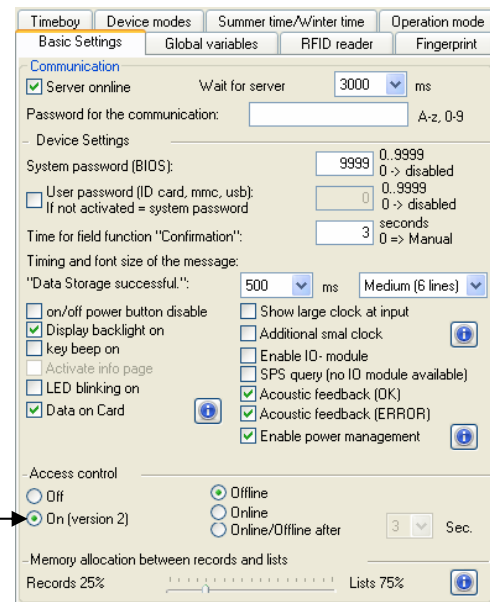
Settings	Creation of data record	Change of direction by				
		90°	45°	22.5°	11.25°	≈6°
coarse	after sec.	6	12	24	48	96
medium	after sec.	3	6	12	12	24
fine	after sec.	1	2	4	8	16

The input sequence assigned under change of direction is executed in case of a change of direction of the GPS coordinates. Depending on the resolution selected, more or less data records are created. The data density also depends on the distance travelled. For a winding road more data are created than for a long, straight motorway.

5.6. Recording Procedure of Access Control

Access control has a special status. It belongs - like signal processing - to the automatized processes. Access logistics, i.e. which person is granted or denied access when and where, is defined via seven lists. The following lists are required: Reader, Identification, Location, Time, Holiday, Event and Action. They are simple ASCII (*.txt) lists, which are loaded into the device and processed when a booking is performed at an access module.

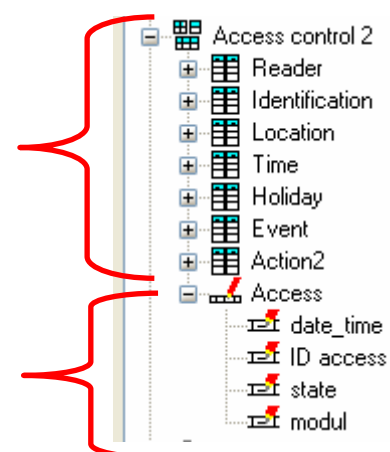
In order to use this functionality of the access control, it must be activated in the setup. An exception is the ZK-Master because access control is always activated for it.



Because the ZK-MasterIV has neither keyboard nor display, an operation in form of direct user input is not possible. The input sequences created for access control verifies whether a person is authorized for access or not. Interaction with the user is restricted on holding the ID card to a reader of the system. Depending on the verification result, a relay or open-collector is switched or not. A data record with a status message is created. Thus, it is possible to track which person tried to access an area.

The lists of the access control are predefined. This concerns structure and name.

An input sequence of access recording is always executed in the background if a record is done at an access reader.



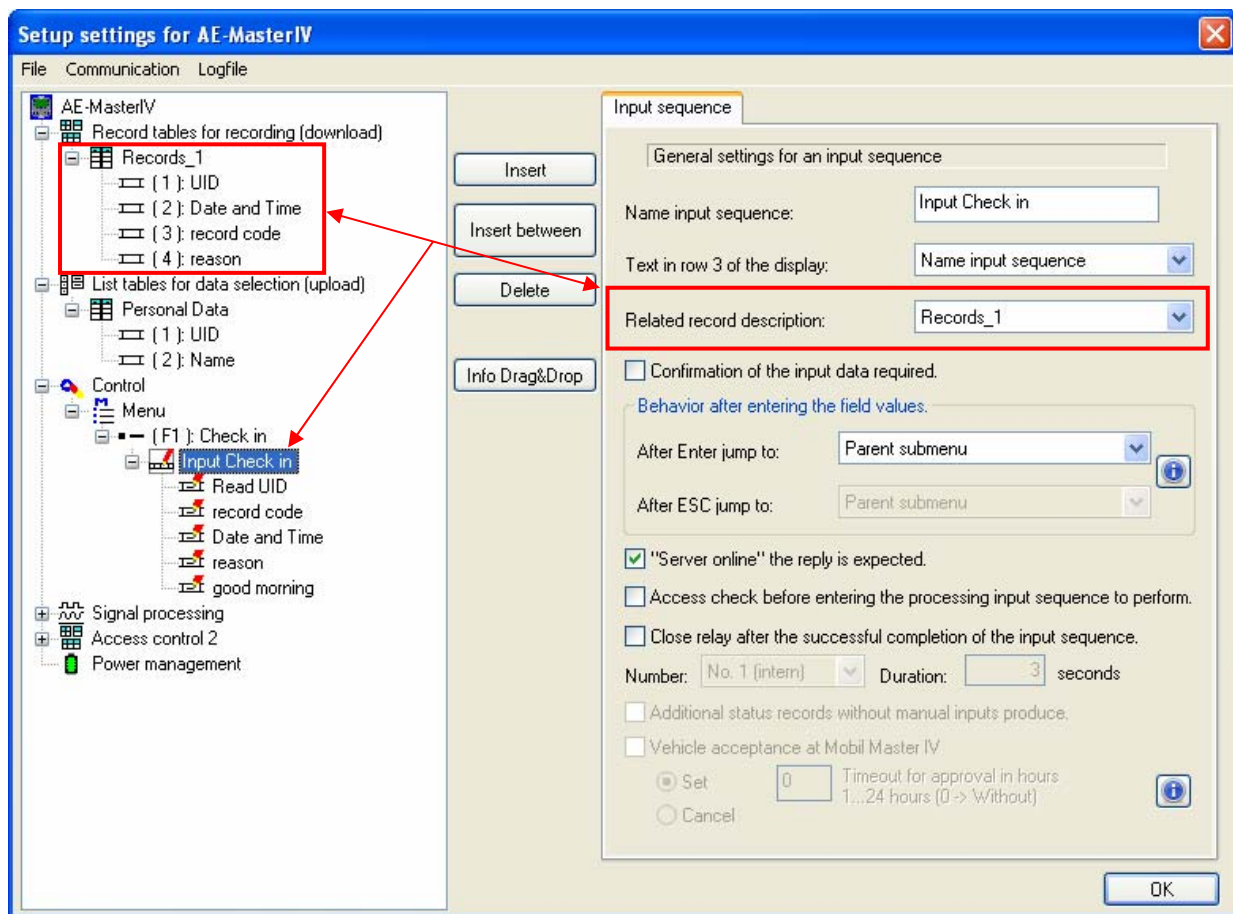
All functions available for the input in an input sequence are described in the following chapter.

5.7. Input field-functions in the device setup

As we have noted, a posting procedure in the form of a command chain for generating data be defined as required. In such an input string (booking process) the data are generated (information in the fields of a record) in the input string fields through the execution of field functions or edited. Each input string field you can assign it a field function and configure.

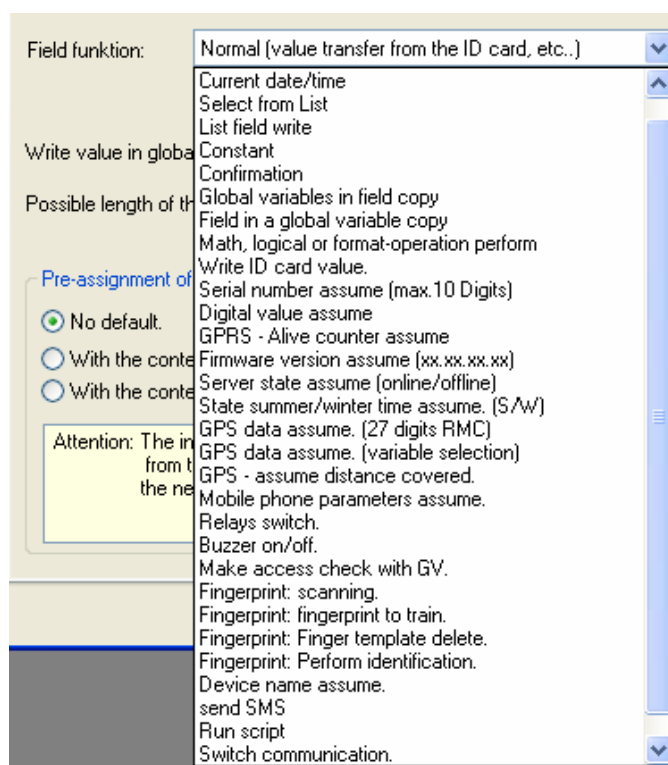
Basics:

The main purpose of the device is to collect data. So the collected data is also stored, the input string a record associated with it.



Then processed the input sequence, saved the device the records in the related record description. If the Data to be saved, you see this in the Display with the displayed text „Data Storage successful“.

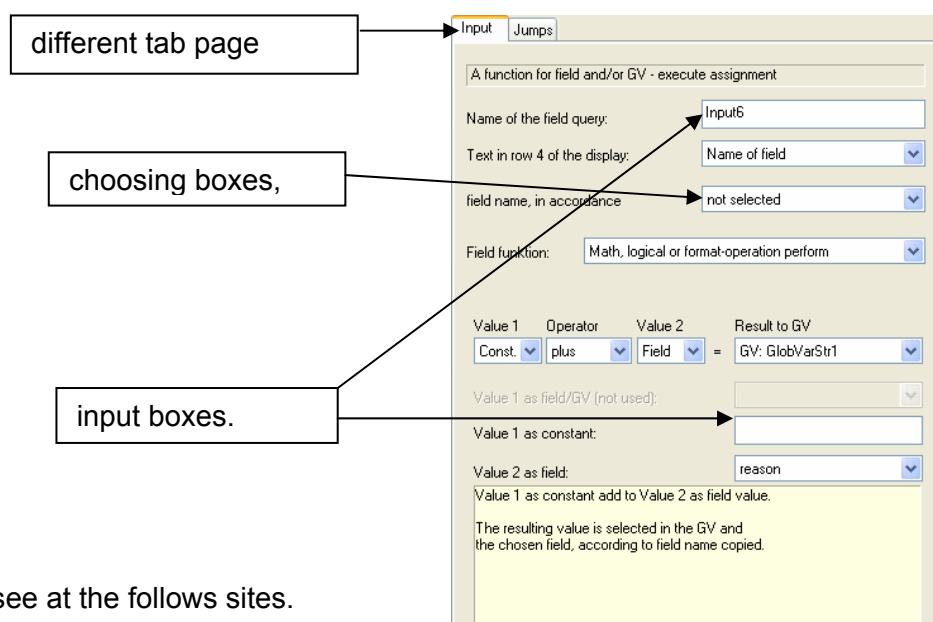
This Input field-functions you can use:



Input field-functions in the access control:

Access: ZM (ZutrittsMaster = access master) assume
Access: TM (TürModul = door module). assume
Access: ID number assume
Access: State assume

Depending on the chosen Input field-functions switch on/off,



More information you see at the follows sites.

5.7.1. Normal (value transfer from RFID card, barcode etc...)

This function can use to, read an RFID-chip, read a barcode or gives an input via Keys.

You choose a field from the records, to save the value.

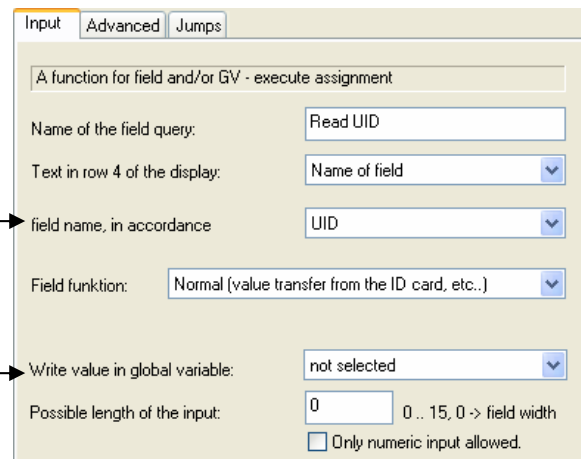
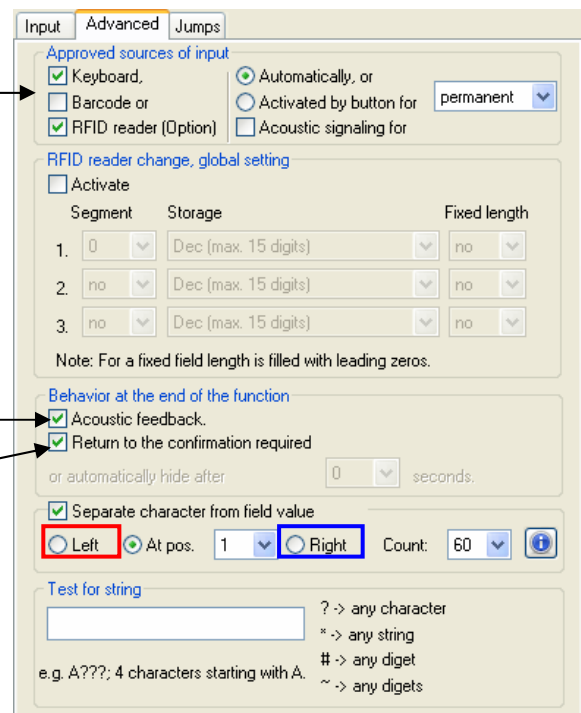
The value can also save in a GV. So you can use the value later in the setup.

Choose, how you want the input.

In each input field, can you choose other settings. You can select different Segments or UID. If you have set the checkmark, are not active the global Transponder settings.

Here you can activate an acoustic feedback after the successful reading. Activate a necessary confirmation with pressed return for an Input.

- at the position (**Left**) and max. 60 characters
- from the position X and quantity of characters, from begin the value.
- from the end off value (**Right**) and quantity of characters

Example to check an input value to confirm value:

- *123 - check of the number "123" on a flexible place (* = any length of characters).
- ??? - read exact 3 any characters.
- ## - read exact 2 any digit.
- F*99 - the first character must be an "F" and folder on a flexible place a 99.
- ?* - read at the minimum 1 character and then flexible, but 1 more is minimum

! For the tab page „jumps“, you find a discription in the chapter „ Jumps“.





5.7.1.1. Default value are displayed before an input value

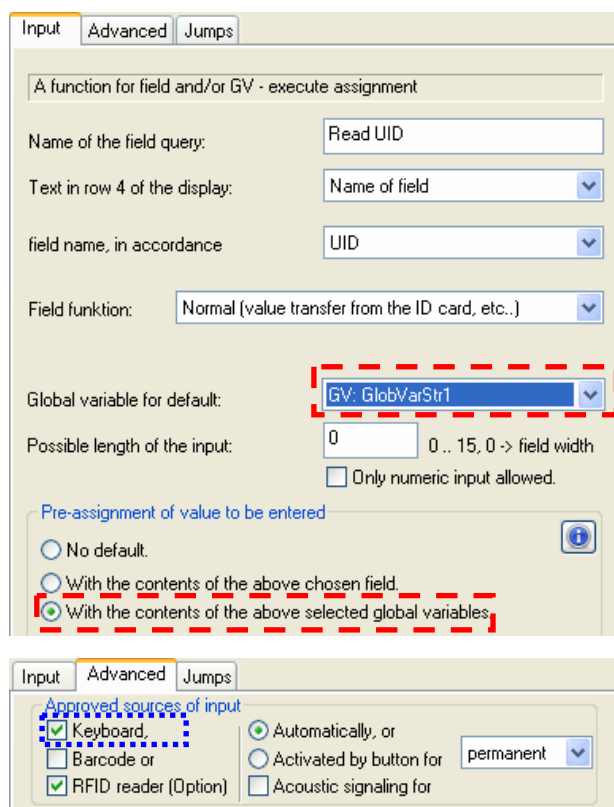
Most helpful is this function, if you control or change the value from GV.

With this function is possible to display a default input value. The default value can read from record-field or from GV.

Here displayed the value from the **GV**. This value can be changed now.

Display example of TimeboyIV, to change the value from GV:

15.08.11 11:28:08				
Timeboy IV				
Demo Setup 3				
123456 Test_				



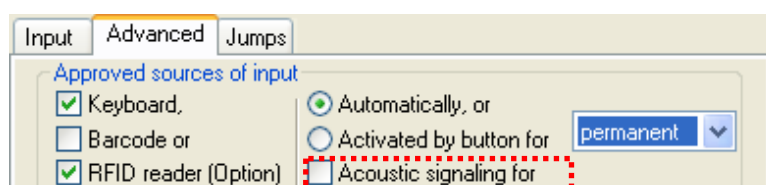
If you reading a transponder, to overwrite the value.

When using the keyboard example a Datafox TimeboyIV or AE-MasterIV can change this default value.

The same applies if the default is derived from the data field. So the data record may subsequently be tested gradually correct inputs

5.7.1.2. The transponder signalling readiness

Under the "Advanced" tab, you have the opportunity to provide an acoustic signal when a value is to be read. You can choose how long the signalling is active.

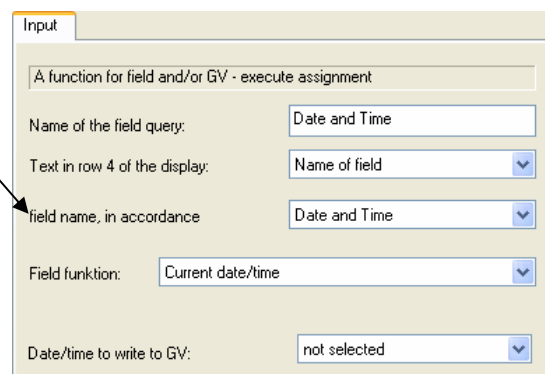


5.7.2. Current Date / Time

This field function provides the current date / time available. The values to be taken from the clock of the device

The data field in which the date will be stored, as a field type date and time to be defined.

If you save the date and time in a global Variable, get the Value in seconds since 01.01.2000 00:00 to date. The second's value can then be used for calculations and comparisons.



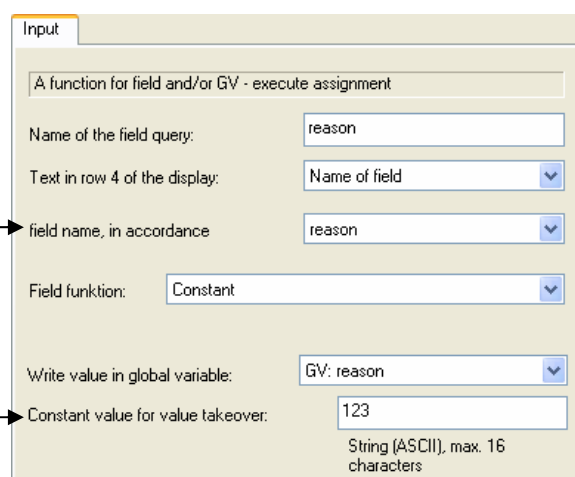
The screenshot shows the 'Input' tab with the following configuration:

- Name of the field query: Date and Time
- Text in row 4 of the display: Name of field
- field name, in accordance: Date and Time
- Field funktion: Current date/time
- Date/time to write to GV: not selected

5.7.3. Constant

Assigned to the corresponding data field from the data set description

Input value for the constant.



The screenshot shows the 'Input' tab with the following configuration:

- Name of the field query: reason
- Text in row 4 of the display: Name of field
- field name, in accordance: reason
- Field funktion: Constant
- Write value in global variable: GV: reason
- Constant value for value takeover: 123

String (ASCII), max. 16 characters

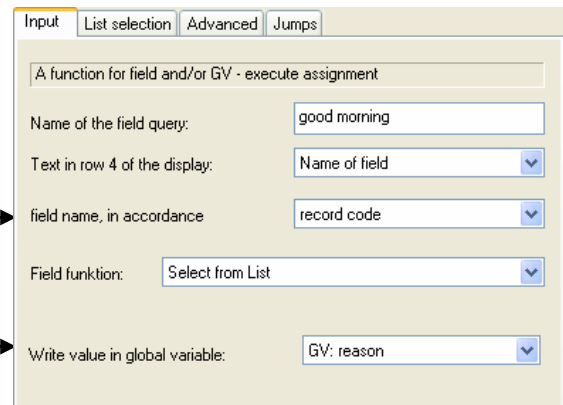
This function can also be used to in order to change the value of a global variable. It is not absolutely necessary, to assign the value of a record field and store.

5.7.4. Select from a list

Through lists the device or the user with additional information is made available to. This can e.g. be a list of reasons goes, from which the user can select the reason for the work stoppage. Similarly, a list, e.g. HR master, a transponder number are assigned to the appropriate name.

Enter the appropriate field of the record in which the value read from the list to be saved.

In addition, this can also be stored in a GV.

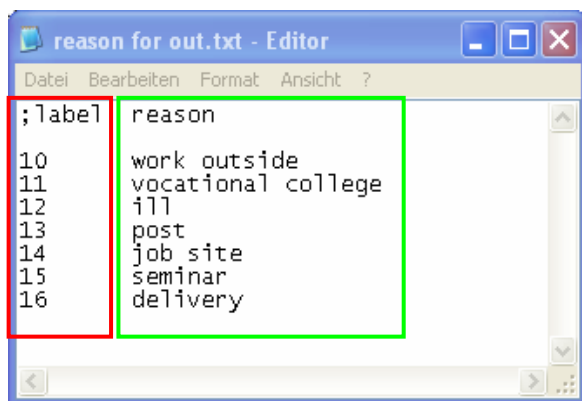
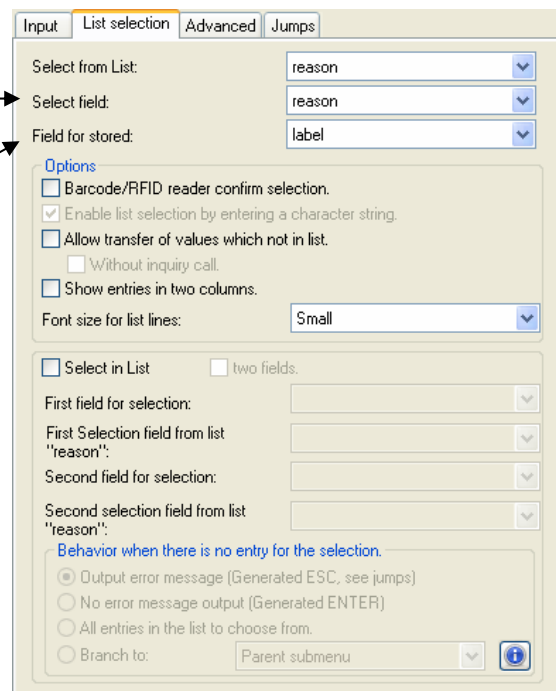


5.7.4.1. Choose from a list

An example of the selection list will be shown. We take here a list of reason to check out.

Specify here which **column (field)** the list on the unit's display to display.

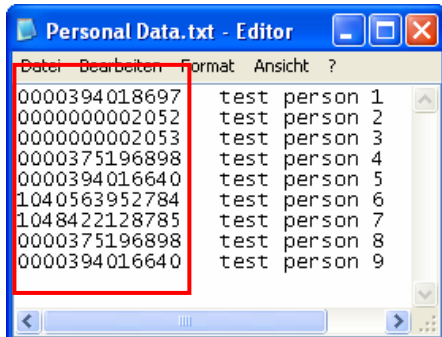
Enter here, from which column of the list, the data to be included in the record.

Choose the user now "post" to save the value 13 in the records and/or in the global variables.

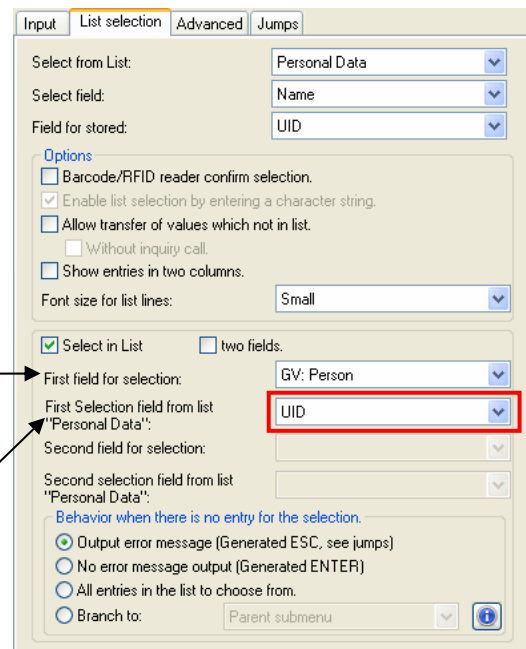
5.7.4.2. List selection

Here is to show the example of a personnel list will be selected and displayed as a read transponder number, the corresponding name from a list.



Enter here to search for what number.
The number may be from a record field of the current record or from a GV.

Specify here which column of the list "Personal Data" the **above number** to search for.

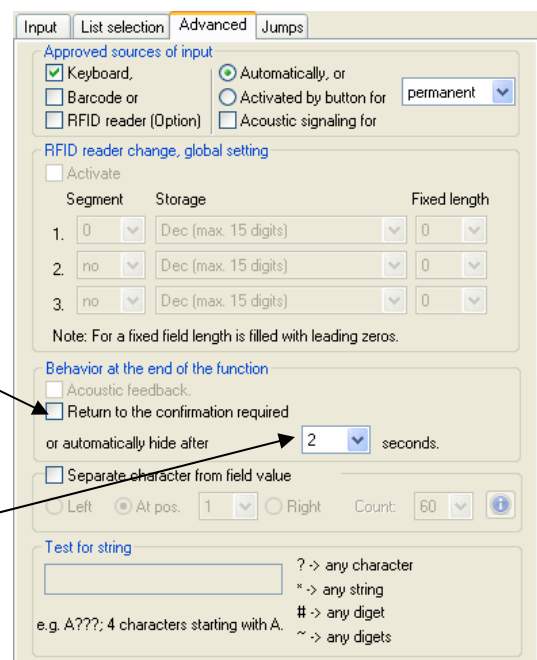


Setting on the register „Advanced“.

If you use this checkmark then must press the user "enter" to confirm the input.

The second option is to display the selected names for a certain time.

If the time is set to 0, the selection is automatic (no indication on the display).



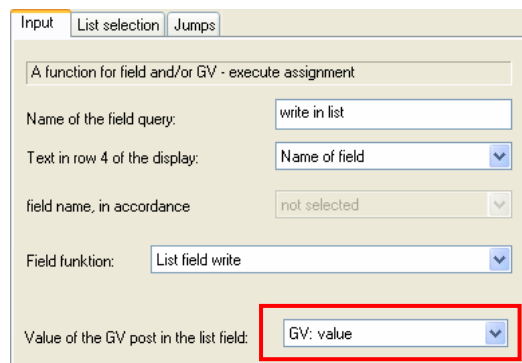
5.7.5. Write in a list

The entries in a list in the player can be changed with this function.

Prerequisite for this is created in the setup list table with the appropriate fields.

The list, in which the data is to be written must have been transferred to the device.

The value to written in the list, can you take from a GV.

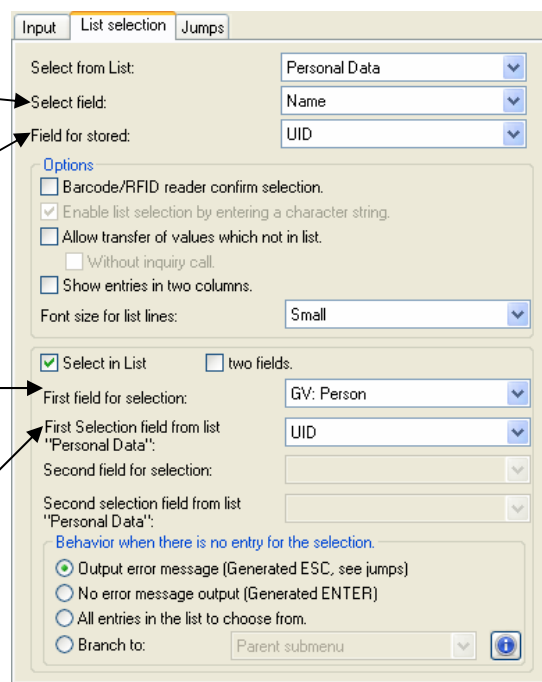


Since no data displayed from the list, this information is not needed.

Specify here which column in the list, the value of the GV is to be saved.

Enter here to search for what number. The number may be from a record or a GV.

You choose here the column from the list of "Personal Data" to search the selected number.



Note:

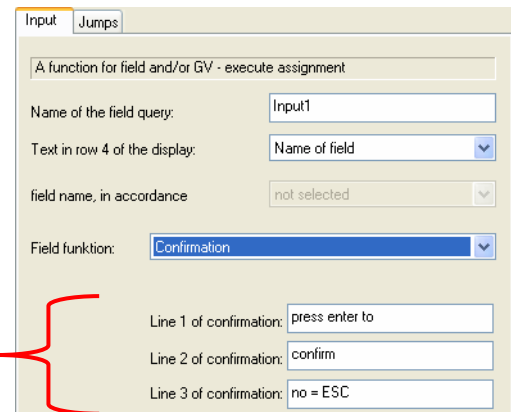


If the selection list is not selected, the value of the specified GM is stored in the first row of the list.
The "write list" function only makes sense if the value is stored at the corresponding position in the list.

5.7.6. Confirmation

A confirmation field is used for check of input for the user. Here can the user control his inputs. Also is possible to confirm the input and conditional from inputs to jump. More information for the jumps you find in the chapter „[Jumps](#)“.

This field you can use to give the user some information. This text displayed on device.

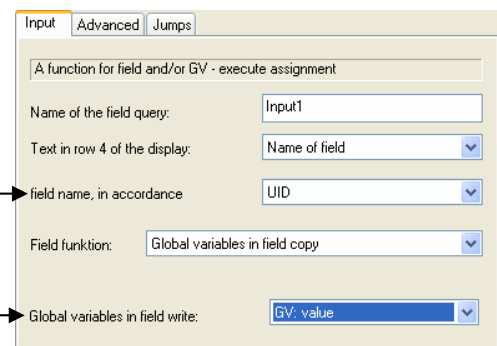


5.7.7. Global variables in field copy

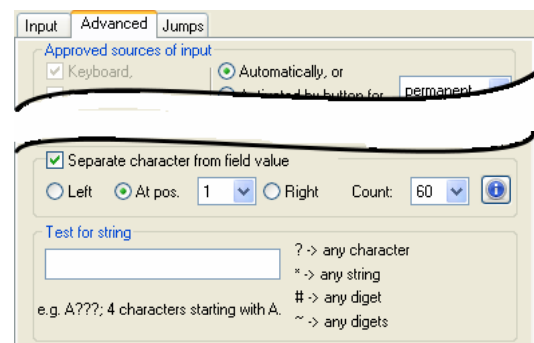
The content of global variables can you here copy in the records and cut the length.

You choose a field from the records, to save the content of the GV.

Select here the GV do you want save.



On the tab “Advanced” you can the content of the GV cut the length.

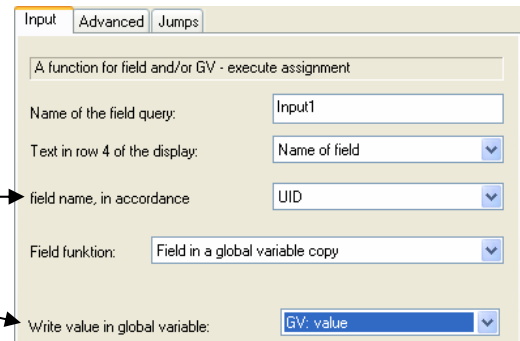


5.7.8. Field in a Global variable copy

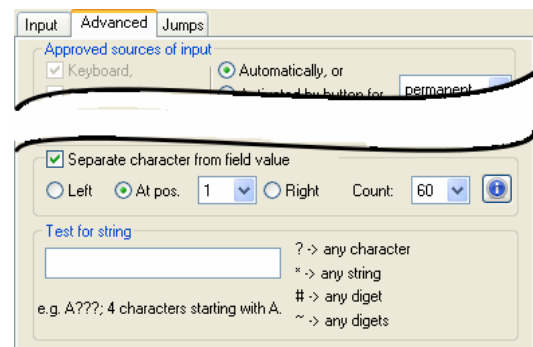
A current value from the records you can save back in a global variable.
This to make sense if you use the content from the field later or you need a part of length.

You choose a field from the records, to save the content from the field in a GV.

Select here the GV to you want save.



On the tab "Advanced" you can the content of the GV cut the length.



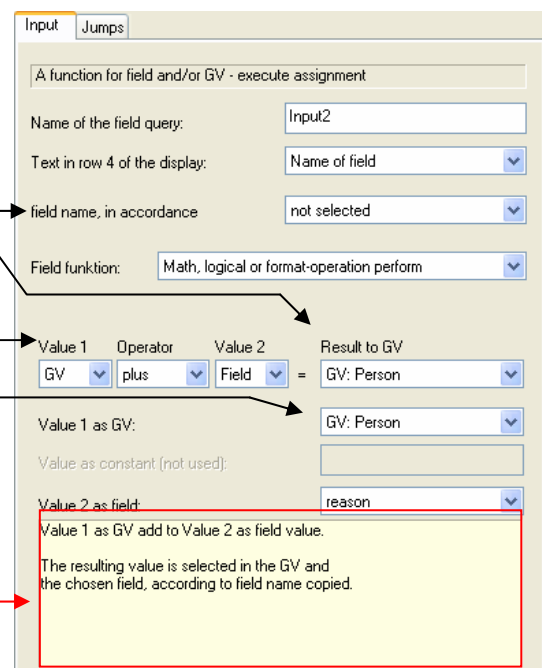
5.7.9. To perform math, logic, or format operations

You can use value from GV or data fields.

The result can be saved in the records or in GV.

If you use content from a record field then can you only select numeric field.

! You get here some information about the function.



Examples for the operations

To add /plus:

Value 1	plus add	Value 2	result save in GV
123456789	+	1	123456790
GV Person	+	Value 2 as a constant	GV Person

Value 1: GV, Operator: plus, Value 2: Const., Result to GV: GV: Person

Value 1 as GV: GV: Person

Value 2 as constant: 1

To compare:

The result at the compare to be saved at false=0 or true=1 in the GV.

Value 1	greater	Value 2	result save in GV
123456789	>	1	1
GV value	>	Value 2 as a constant	GV label

Value 1: GV, Operator: greater, Value 2: Const., Result to GV: GV: label

Value 1 as GV: GV: value

Value 2 as constant: 1

Value 2 as field/GV (not used):

Conversion to hexadecimal value:

The operation "to Hex" is available for this action.

The value mask serves as indicator for the number of characters. More characters than in the initial value can be used in the value mask. By this, leading zeros can be added, for example.

Examples decimal to hexadecimal with 13 character value mask:

Same applies for conversion of decimal values to hex.

Value 1	to Hex	Value 2	result in GV
000005202	to Hex	0000000000000	0000000000804
164166271	to Hex	0000000000000	0000009C8FA7F
000002052	to Hex	FFFFFFFFF00000	FFFFFFFFF00804
164166271	to Hex	FFFFFFFFF00000	FFFFF09C8FA7F
GV GlobCardNo	to Hex	Value 2 as value mask	GV GlobPersonnelNo

Value 1: GV, Operator: plus, Value 2: Const., Result to GV: GV: GlobVarID2

Value 1 as GV: GV: GlobVarID1

Value 2 as constant: 0000000000000

Value 2 as field/GV (not used):

Value 1 as GV add to Value 2 as constant.

The resulting value is selected in the GV and the chosen field, according to field name copied.



Caution:

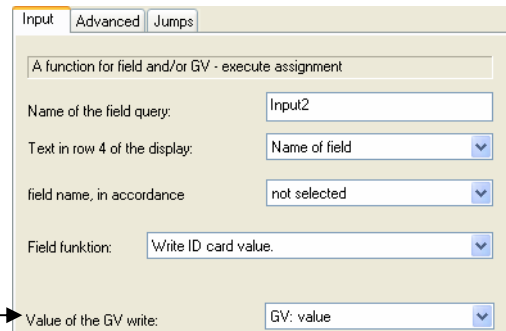
Due to system design, for the data in a GV it is not indexed whether it contains hexadecimal or decimal values. Therefore, you must make sure that a hexadecimal value is not converted again to a hexadecimal value and a decimal value to a decimal value.

5.7.10. Write value on RFID-Tag

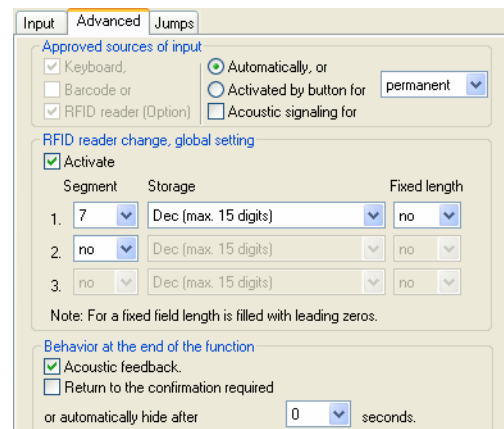
This function makes possible to write data on a transponder.

The content of a global variable (GV) you can save on ID card.

Choose here the global variable.



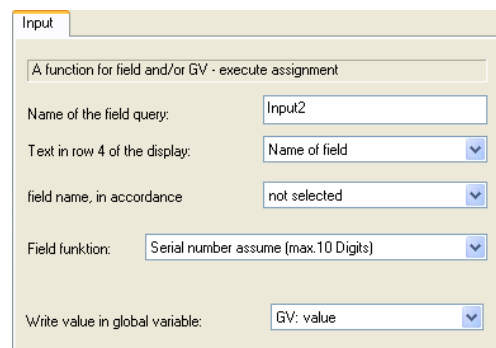
Select here the sector where to write the value on the ID card.



5.7.11. Serial number assume

With assume the serial number it is possible to indicate the terminal.

The serial number is always unique for a device.



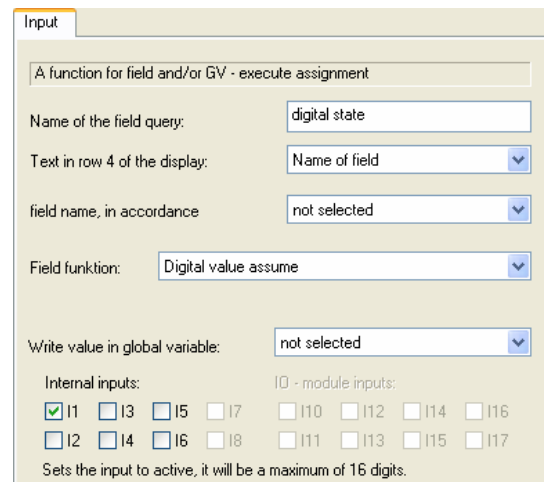
5.7.12. Digital state assume

The condition of the digital input is queried and stored in the data field and / or in the specified GV.

The stored format as follow:

1----- E1 = 1(high)
0----- E1 = 0(low)

- = no active input channel
0 or 1 active

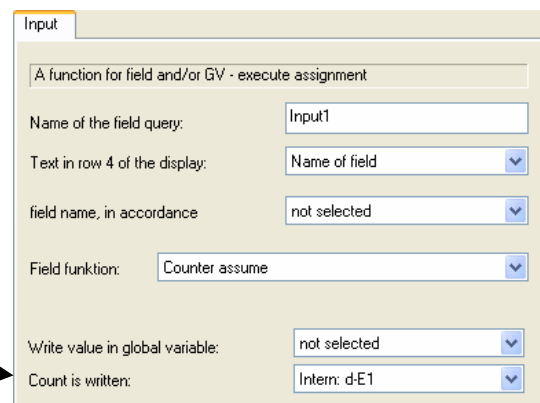


The screenshot shows the 'Input' configuration window. The title is 'Input'. Below it is a subtitle 'A function for field and/or GV - execute assignment'. The 'Name of the field query:' is set to 'digital state'. The 'Text in row 4 of the display:' is set to 'Name of field'. The 'field name, in accordance' is set to 'not selected'. The 'Field funktion:' is set to 'Digital value assume'. The 'Write value in global variable:' is set to 'not selected'. Below this, there are two sections: 'Internal inputs:' and 'IO - module inputs:'. Under 'Internal inputs:', I1 is checked, and I2 through I17 are unchecked. Under 'IO - module inputs:', I10 through I17 are unchecked. At the bottom, it says 'Sets the input to active, it will be a maximum of 16 digits.'

5.7.13. Counter assume

The value of the digital counter is queried and stored in the data field and / or in the specified GV. The counter is reset to 0 after the reading!

Select here the counter of the digital input canal.



The screenshot shows the 'Input' configuration window. The title is 'Input'. Below it is a subtitle 'A function for field and/or GV - execute assignment'. The 'Name of the field query:' is set to 'Input1'. The 'Text in row 4 of the display:' is set to 'Name of field'. The 'field name, in accordance' is set to 'not selected'. The 'Field funktion:' is set to 'Counter assume'. The 'Write value in global variable:' is set to 'not selected'. The 'Count is written:' is set to 'Intern: d-E1'. An arrow points from the text 'Select here the counter of the digital input canal.' to the 'Count is written:' dropdown menu.

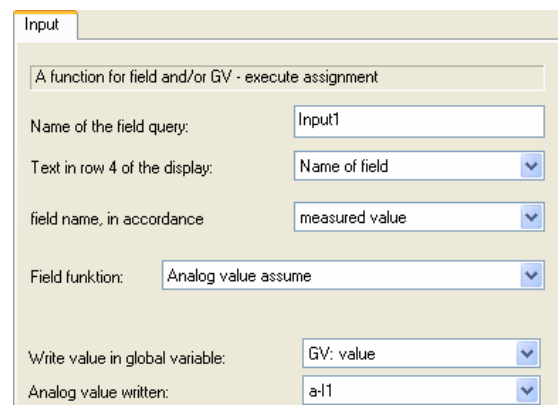
5.7.14. Analog measurement assume

The state of the analog input is polled and stored in the data field and / or in the specified GV.

For more details see Chapter „[Analog inputs](#)“.

Here an example for the saved format:

- 34.5 °F
- 9.2 V



The screenshot shows the 'Input' configuration window. The title is 'Input'. Below it is a subtitle 'A function for field and/or GV - execute assignment'. The 'Name of the field query:' is set to 'Input1'. The 'Text in row 4 of the display:' is set to 'Name of field'. The 'field name, in accordance' is set to 'measured value'. The 'Field funktion:' is set to 'Analog value assume'. The 'Write value in global variable:' is set to 'GV: value'. The 'Analog value written:' is set to 'a-I1'.

5.7.14.1. Integralfunction for analog measurement

The existing field function "*analog value assume*" has been extended to an integral functionality. If the checkbox is checked for the integrated data of the selected input is accepted.

This makes it possible to measure a "volume", for example the energy consumption of devices by current measurements.

In addition, you have the possibility to save the period of measurement in seconds, either in another field in the record or store it in a GV for farther processing.

Input

A function for field and/or GV - execute assignment

Name of the field query:

Text in row 4 of the display:

field name, in accordance

Field function:

Write value in global variable:

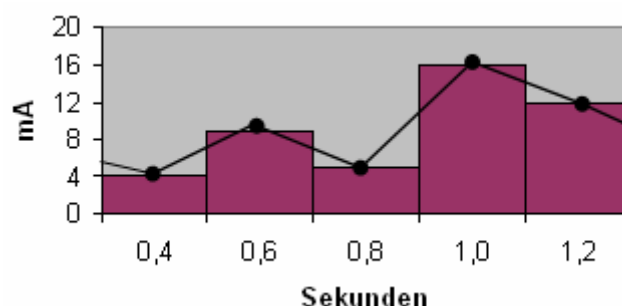
Analog value written:

☒ Accept integral value

☐ Measurement time in field or

☒ write in global variable

The sampling rate is 200 ms for all analog inputs. I.e. each analog input is measured five times in the second and the value added.



The picture shows that accurate values are possible only if the view values change slowly.

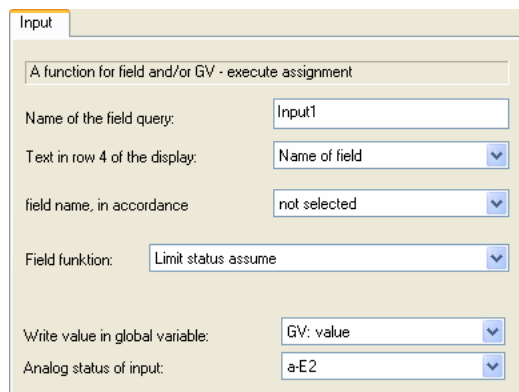
5.7.15. Analog limits assume

The limits of the monitored analog inputs can be saved as status in a data field and / or in the specified GV.

The storage format is, depending on the direction of change:

1->2 2->3 3->2
3->4 4->5 5->4

More information you find in the chapter „[analog input](#)“.



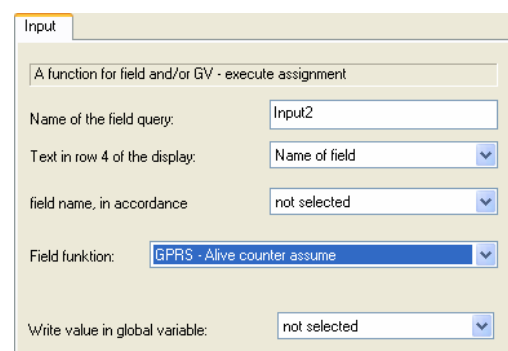
The screenshot shows the 'Input' configuration window. The title bar says 'Input'. Below it is a tabbed interface with the first tab selected. The main area is titled 'A function for field and/or GV - execute assignment'. It contains several fields and dropdown menus: 'Name of the field query:' with the value 'Input1'; 'Text in row 4 of the display:' with the value 'Name of field'; 'field name, in accordance' with the value 'not selected'; 'Field funktion:' with the value 'Limit status assume'; 'Write value in global variable:' with the value 'GV: value'; and 'Analog status of input:' with the value 'a-E2'.

5.7.16. GPRS Alivecounter assume

The value of the alivecounter can be saved as status in a data field and / or in the specified GV.

More information you find in the chapter :

„[Recording Procedures of Signal Processing -> GPRS-Alive](#)“

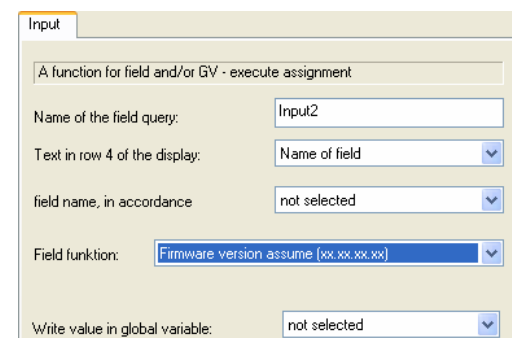


The screenshot shows the 'Input' configuration window. The title bar says 'Input'. Below it is a tabbed interface with the first tab selected. The main area is titled 'A function for field and/or GV - execute assignment'. It contains several fields and dropdown menus: 'Name of the field query:' with the value 'Input2'; 'Text in row 4 of the display:' with the value 'Name of field'; 'field name, in accordance' with the value 'not selected'; 'Field funktion:' with the value 'GPRS - Alive counter assume'; 'Write value in global variable:' with the value 'not selected'.

5.7.17. Firmware version assume (xx.xx.xx.xx)

The version number of the firmware can be saved as status in a data field and / or in the specified GV.

The firmware contained on the device is dependent on the current firmware at the time of surrender or of any updates made. See chapter "Version Changes" at the beginning of this manual.



The screenshot shows the 'Input' configuration window. The title bar says 'Input'. Below it is a tabbed interface with the first tab selected. The main area is titled 'A function for field and/or GV - execute assignment'. It contains several fields and dropdown menus: 'Name of the field query:' with the value 'Input2'; 'Text in row 4 of the display:' with the value 'Name of field'; 'field name, in accordance' with the value 'not selected'; 'Field funktion:' with the value 'Firmware version assume (xx.xx.xx.xx)'; 'Write value in global variable:' with the value 'not selected'.

5.7.18. Status der Summer-/Winter time assume

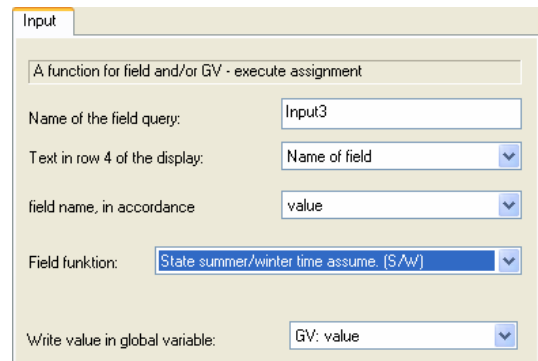
The status of Sommer-/Winter time can be saved as status in a data field and / or in the specified GV.

The saved format is as follows:

S summertime

W wintertime

More information you find in the chapter „[Setup structure / summer-/wintertime](#)“

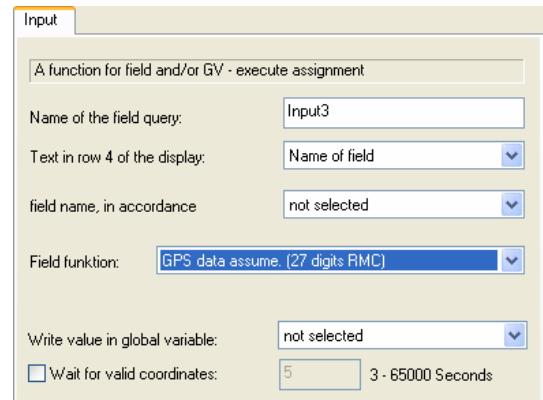


The screenshot shows the 'Input' tab of a configuration window. It contains the following fields and options:

- Name of the field query:** Input3
- Text in row 4 of the display:** Name of field (dropdown)
- field name, in accordance:** value (dropdown)
- Field funktion:** State summer/winter time assume. (S/W) (dropdown)
- Write value in global variable:** GV: value (dropdown)

5.7.19. GPS data assume

The value can be saved as status in a data field and / or in the specified GV.



The screenshot shows the 'Input' tab of a configuration window. It contains the following fields and options:

- Name of the field query:** Input3
- Text in row 4 of the display:** Name of field (dropdown)
- field name, in accordance:** not selected (dropdown)
- Field funktion:** GPS data assume. (27 digits RMC) (dropdown)
- Write value in global variable:** not selected (dropdown)
- Wait for valid coordinates:** ☐ 5 3 - 65000 Seconds

The saved format is as follows:

A,5043.1526,N,00957.6707,E, bzw. V,5043.1526,N,00957.6707,E,

meaning:

- **validity:**
 - A= available
 - V= void
- **Latitude** N/S
- **longitude** E/W

5.7.20. GPS - data assume (variable selection)

It will be provided to you with this function GPS data available.

The data you can save in a record field and/or save in a GV.

Set your checkmarks for the data what you want. To watch out of the size in the record field.

Please read the notes.

Input

A function for field and/or GV - execute assignment

Name of the field query:

Input3

Text in row 4 of the display:

Name of field

field name, in accordance

value

Field funktion:

GPS data assume. (variable selection)

Write value in global variable:

GV: value

Wait for valid coordinates:

5

3 - 65000 Seconds

UTC - Time

☐

Validity

☒

N/S latitude

☒

E/W longitude

☒

Speed (in knots over the ground)

☐

True course (no movement)

☐

UTC date

☐

Magnetic declination

☐

GPS mode

☒

Loop

☐

Memory required for RMC data field:

29 bytes of a possible 32.

Construction of an RMC data record:

By enabling the individual RMC data values, an RMC be compiled. This results in the construction of the RMC records from the activated values in the order from left top (UTC) right down (Loop,...)

Note link:

http://de.wikipedia.org/wiki/NMEA_0183#Recommended_Minimum_Sentence_C_.28RMC.29

http://en.wikipedia.org/wiki/NMEA_0183



Note:

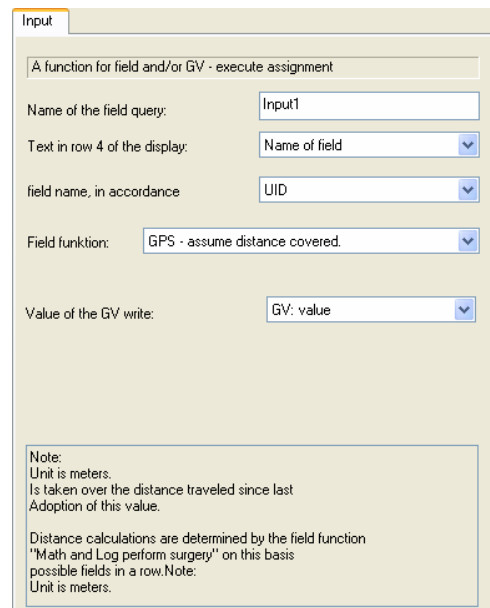
The maximum at the data field is 40 byte. Use more data fields to save all Necessary RMC data.

5.7.21. GPS - assume distance covered

distance assume

The data/distance from the last call of this function you can save in a record field and/or save in a GV.

The value range is between 0 und $2^{32}-1$.



Input

A function for field and/or GV - execute assignment

Name of the field query: Input1

Text in row 4 of the display: Name of field

field name, in accordance: UID

Field funktion: GPS - assume distance covered.

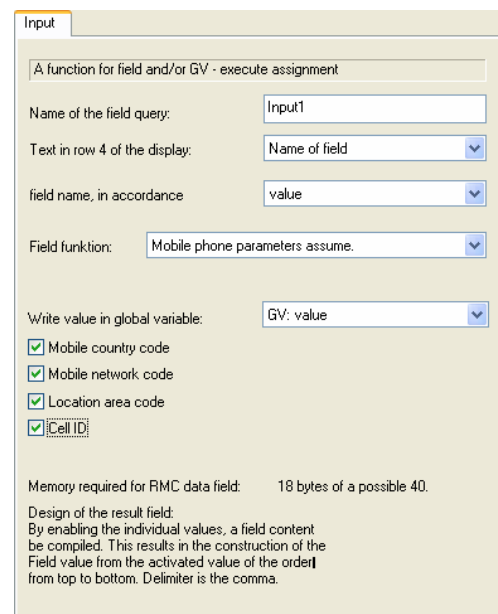
Value of the GV write: GV: value

Note:
Unit is meters.
Is taken over the distance traveled since last Adoption of this value.
Distance calculations are determined by the field function "Math and Log perform surgery" on this basis possible fields in a row. Note:
Unit is meters.

5.7.22. Mobile phone parameters assume

It will put you with this function GPS data available.

The data you can save in a record field and/or save in a GV.



Input

A function for field and/or GV - execute assignment

Name of the field query: Input1

Text in row 4 of the display: Name of field

field name, in accordance: value

Field funktion: Mobile phone parameters assume.

Write value in global variable: GV: value

☒ Mobile country code
☒ Mobile network code
☒ Location area code
☒ Cell ID

Memory required for RMC data field: 18 bytes of a possible 40.

Design of the result field:
By enabling the individual values, a field content be compiled. This results in the construction of the Field value from the activated value of the order from top to bottom. Delimiter is the comma.

Here some links:

- Mobile Country Code: http://en.wikipedia.org/wiki/Mobile_Country_Code
- Mobile Network Code: http://en.wikipedia.org/wiki/Mobile_Network_Code
- Location Area Code: http://en.wikipedia.org/wiki/Location_Area
- Cell ID: http://en.wikipedia.org/wiki/Cell_of_Origin

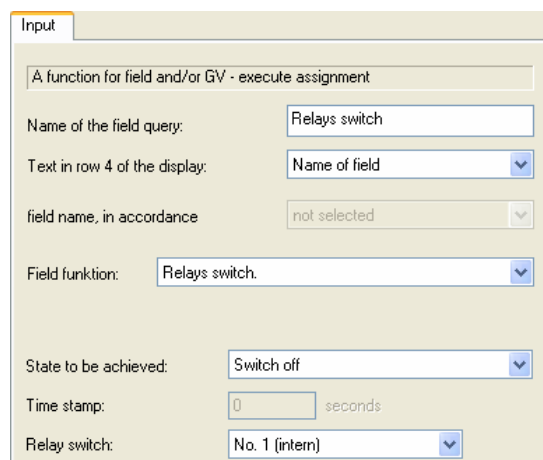
5.7.23. Relay switch

With this function it is possible to switch the internal relay of the device.

Switching options are available:

- switch off
- switch on
- switch
- switch off for time
- switch on for time
- switch for time
- switch off after time
- switch on after time
- switch after time

Choose the relays what you switch.



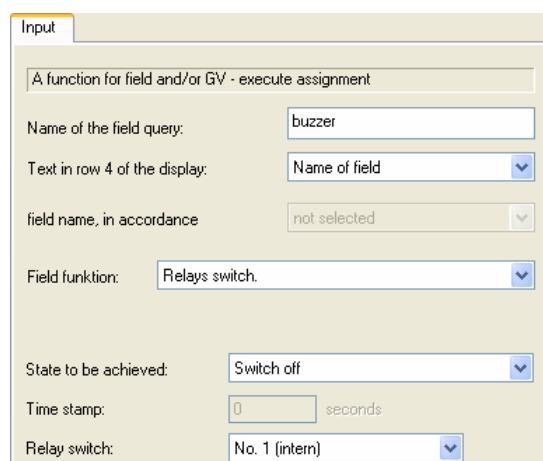
The screenshot shows the 'Input' tab of a configuration window titled 'A function for field and/or GV - execute assignment'. The 'Name of the field query' is set to 'Relays switch'. The 'Text in row 4 of the display' is set to 'Name of field'. The 'field name, in accordance' is set to 'not selected'. The 'Field funktion:' is set to 'Relays switch.'. The 'State to be achieved:' is set to 'Switch off'. The 'Time stamp:' is set to '0 seconds'. The 'Relay switch:' is set to 'No. 1 (intern)'.

5.7.24. Buzzer switch

With this function it is possible to switch the internal buzzer of the device.

Switching options are available:

- switch off
- switch on
- switch
- switch off for time
- switch on for time
- switch for time
- switch off after time
- switch on after time
- switch after time

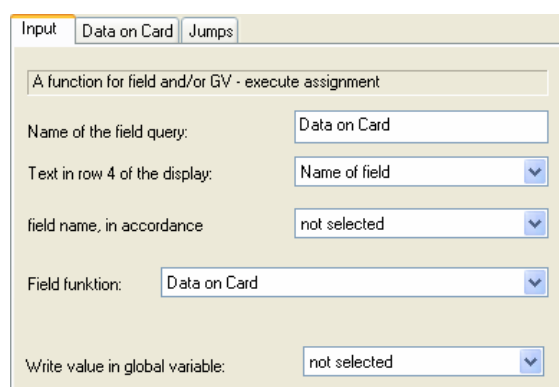


The screenshot shows the 'Input' tab of a configuration window titled 'A function for field and/or GV - execute assignment'. The 'Name of the field query' is set to 'buzzer'. The 'Text in row 4 of the display' is set to 'Name of field'. The 'field name, in accordance' is set to 'not selected'. The 'Field funktion:' is set to 'Relays switch.'. The 'State to be achieved:' is set to 'Switch off'. The 'Time stamp:' is set to '0 seconds'. The 'Relay switch:' is set to 'No. 1 (intern)'.

5.7.25. Data on Card (Fieldfunction)

For this field function needs a detailed description.

The information you find in the chapter "Data on Card".



The screenshot shows the 'Input' tab of a configuration window titled 'A function for field and/or GV - execute assignment'. The 'Name of the field query' is set to 'Data on Card'. The 'Text in row 4 of the display' is set to 'Name of field'. The 'field name, in accordance' is set to 'not selected'. The 'Field funktion:' is set to 'Data on Card'. The 'Write value in global variable:' is set to 'not selected'.

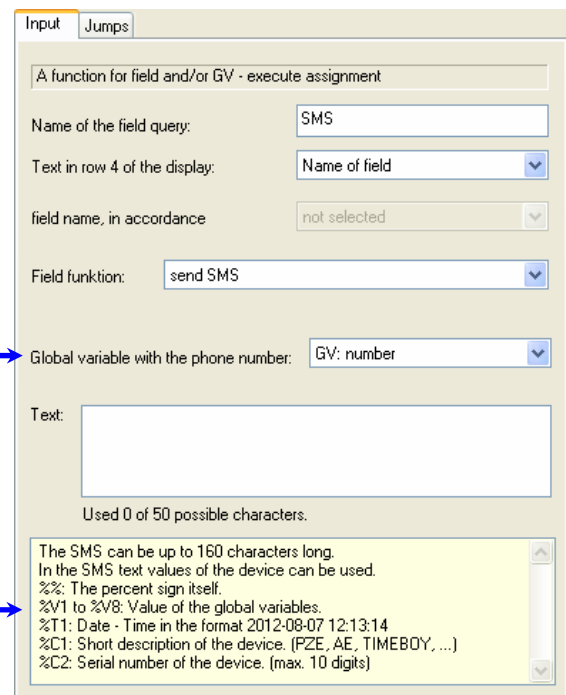
5.7.26. Send SMS

This function provides the ability to send an SMS.

Condition to send a SMS with a device is equipped with a mobile radio modem.

The phone number to which the SMS is to be sent is stored in a GV.

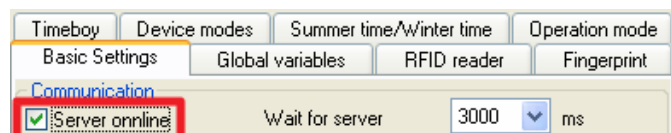
Intern information about the device you can integrate in the SMS here.



More information to this topic you find in the device manual „[Communication via SMS](#)“.

5.7.27. Server status assume (online/offline)

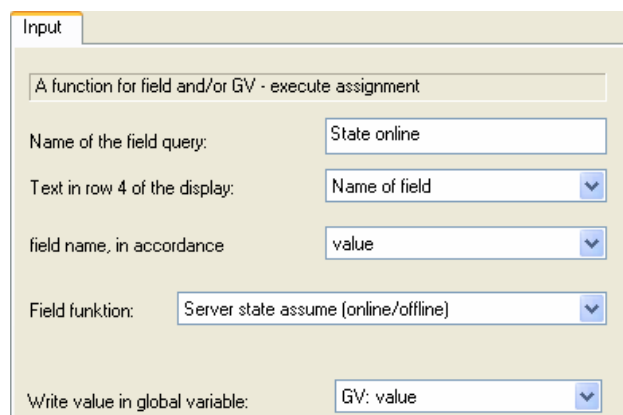
Condition to use this function is activating the online-mode.



If the polling time shorter than the settings



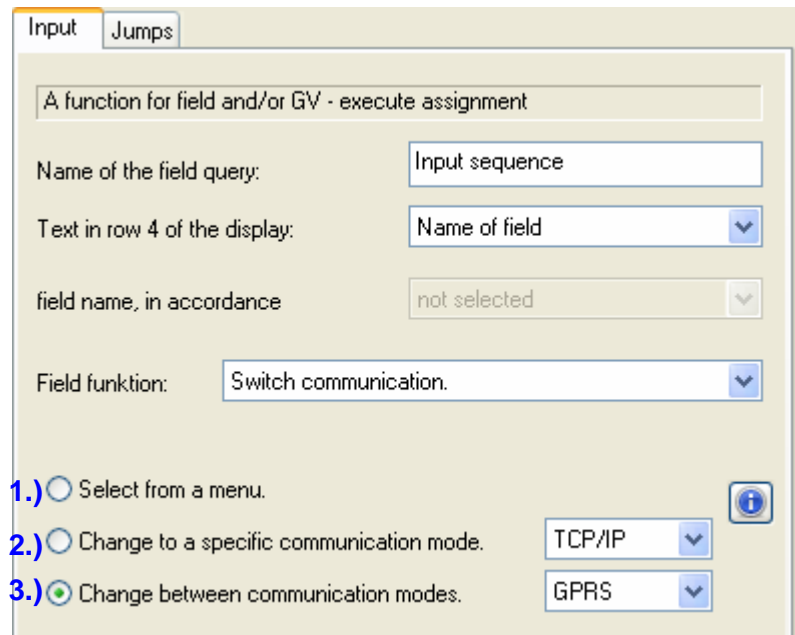
get the value state „online“, otherwise you get the value state „offline“.



5.7.28. Switch communication

With this function it is possible to switch the main-communication in an input sequence. Before the integration of this function, was the switching of communication only been possible in the bios-menu.

There are three possibilities:



1.) Selection via menu

This function is only for devices with display

2.) Switch the main-communication to an specific mode

Here change the communication to the selected mode. If the selected mode already active then no changes.

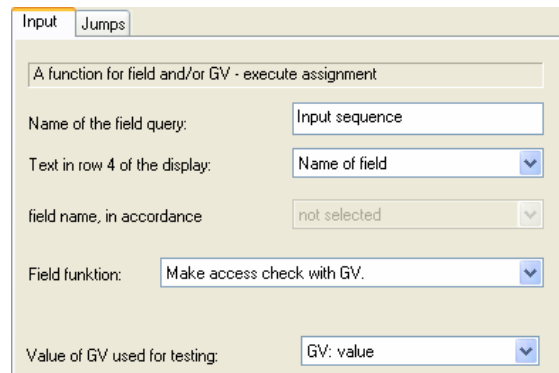
3.) Change between two communications

With every call of this function change the device the other mode.

5.7.29. Access check with GV

If activate the access control then is possible to check an access in the control.
The value for this check must be saving in a GV.

Select here the GV, with the value for the access control.

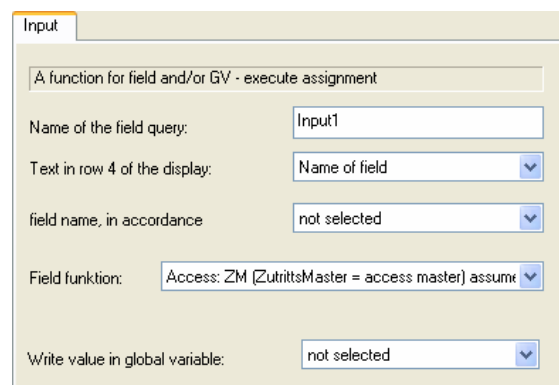


5.7.30. Access: ZM (ZutrittsMaster = access master) assume

This function is only available in an input sequence of the access control.

Here assume the master ID.

The data you can save in a record field and/or save in a GV.

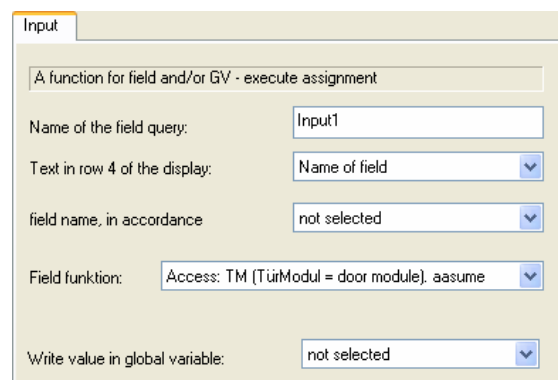


5.7.31. Access: TM (TürModul = door module ID) assume

This function is only available in an input sequence of the access control.

Here assume the bus number of the door module.

The data you can save in a record field and/or save in a GV.

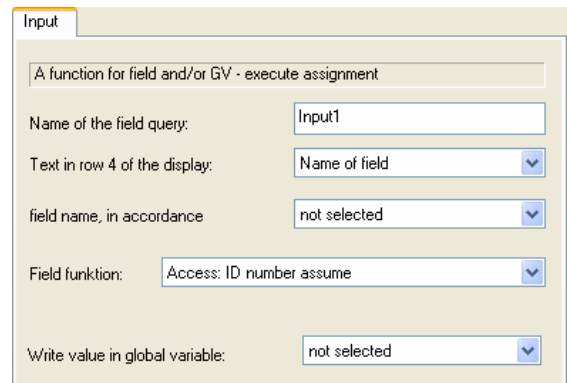


5.7.32. Access: ID number assume

This function is only available in an input sequence of the access control.

Here assume the read RFID- number from a door module.

The data you can save in a record field and/or save in a GV.



Input

A function for field and/or GV - execute assignment

Name of the field query: Input1

Text in row 4 of the display: Name of field

field name, in accordance: not selected

Field funktion: Access: ID number assume

Write value in global variable: not selected

5.7.33. Access: State assume

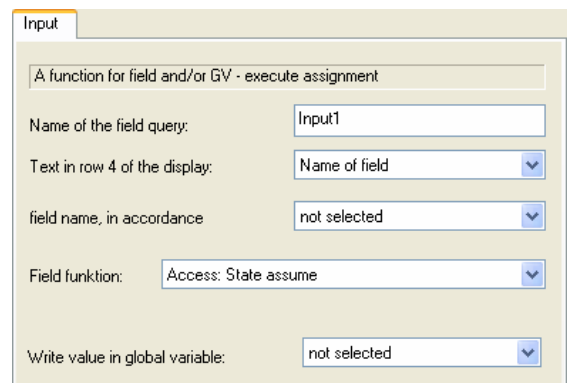
This function is only available in an input sequence of the access control.

The value is a digit value.

This state is important for the troubleshooting.

The meaning of the state numbers you find in the chapter "State of access control".

The data you can save in a record field and/or save in a GV.



Input

A function for field and/or GV - execute assignment

Name of the field query: Input1

Text in row 4 of the display: Name of field

field name, in accordance: not selected

Field funktion: Access: State assume

Write value in global variable: not selected

5.7.34. Fingerprint: Scannen

The "Fingerprint Scan" function activates the fingerprint scanner. Is now a finger scanned, the scanner remembers this scan (template). Following this, the scanner needs to be told what is to be done with the template:

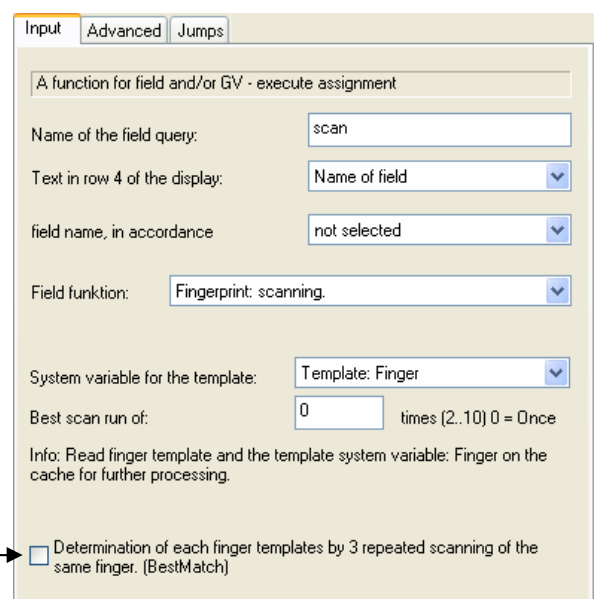
e.g. Identification, verification, teaching or deletion.

The scanning of the finger may be stored in a data field.

! The record field as the field type

should be **Fingertemplate (DIN V66400)**

Use to teach in a finger the function "BestMatch". After the 3 scans save the fingerprint module the best scan.



Input Advanced Jumps

A function for field and/or GV - execute assignment

Name of the field query: scan

Text in row 4 of the display: Name of field

field name, in accordance: not selected

Field funktion: Fingerprint: scanning.

System variable for the template: Template: Finger

Best scan run of: 0 times (2..10) 0 = Once

Info: Read finger template and the template system variable: Finger on the cache for further processing.

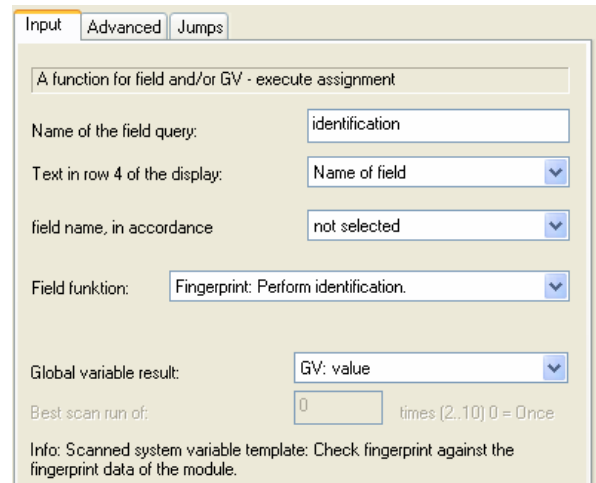
☐ Determination of each finger templates by 3 repeated scanning of the same finger. (BestMatch)

5.7.35. Fingerprint: Perform identification

After the "Fingerprint Scan", the fingerprint module performs the finger scan the last scan by an ID. Upon successful testing (compared with the stored templates in the module), the PID of the associated Finger scans is delivered. This can then be used for further processing.

The returned after successful identification ID you can save in a record field and/or save in a GV.

The returned after successful identification ID you can save in a record field and/or save in a GV.



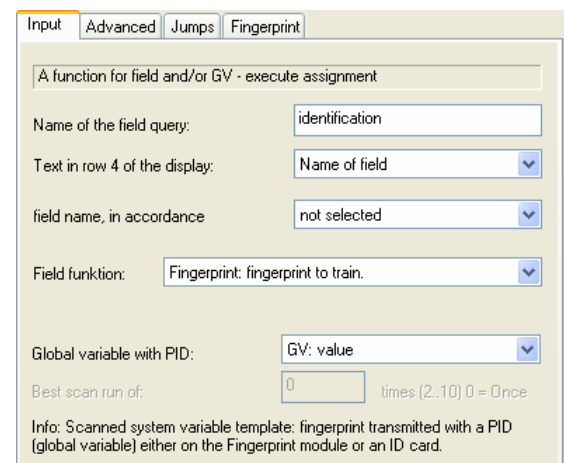
The screenshot shows the 'Input' tab of a configuration window. It contains the following fields and options:

- Name of the field query:** identification
- Text in row 4 of the display:** Name of field (dropdown)
- field name, in accordance:** not selected (dropdown)
- Field funktion:** Fingerprint: Perform identification. (dropdown)
- Global variable result:** GV: value (dropdown)
- Best scan run of:** 0 times (2..10) 0 = Once
- Info:** Scanned system variable template: Check fingerprint against the fingerprint data of the module.

5.7.36. Fingerprint: fingerprint to train

After the "Fingerprint Scan", stores the fingerprint module the finger scan with this function. The finger scan is stored in association with a number (PID) as contiguous data (template). Depending on the global settings, the template is saved on the Biokey module or on a Mifare card.

If you want save the teaching process, select a field of the associated record description in order to save the returned PID.



The screenshot shows the 'Input' tab of a configuration window. It contains the following fields and options:

- Name of the field query:** identification
- Text in row 4 of the display:** Name of field (dropdown)
- field name, in accordance:** not selected (dropdown)
- Field funktion:** Fingerprint: fingerprint to train. (dropdown)
- Global variable with PID:** GV: value (dropdown)
- Best scan run of:** 0 times (2..10) 0 = Once
- Info:** Scanned system variable template: fingerprint transmitted with a PID (global variable) either on the Fingerprint module or an ID card.

Select a GV with the PID.
The Biokey module saved the template with this PID.



The screenshot shows the 'Fingerprint' tab of a configuration window. It contains the following fields and options:

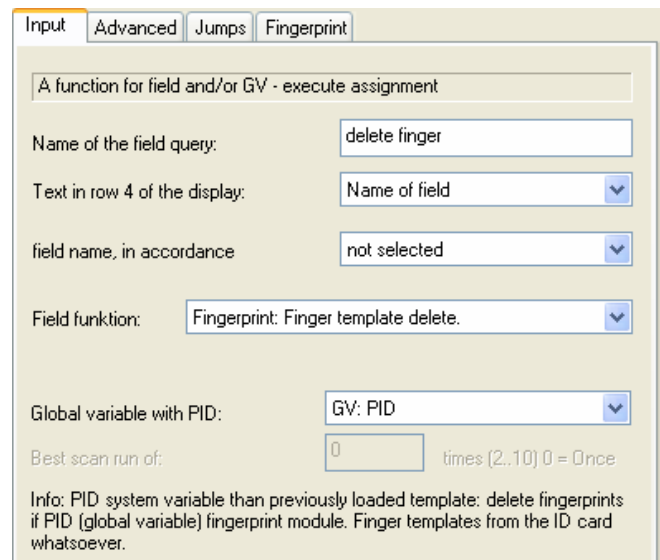
- Additional processing data:**
 - Mapped PID:** GV: Person (dropdown)
 - On Fingerprint module:**
 - ☒ Delete with Template: fingerprint
 - ☐ Delete with PID: GV: Person (dropdown)

5.7.37. Fingerprint: Finger template delete

There are two possibilities to delete a saved template:

1. Delete with using the PID.
All saved templates to be deleted with this selected PID.
2. Delete with the scanning of a saved finger.
In this option, only the matching template is deleted to the scan.

This function gives the PID of the deleted template back.
The PID you can save in a record field and/or save in a GV.



Input Advanced Jumps Fingerprint

A function for field and/or GV - execute assignment

Name of the field query: delete finger

Text in row 4 of the display: Name of field

field name, in accordance: not selected

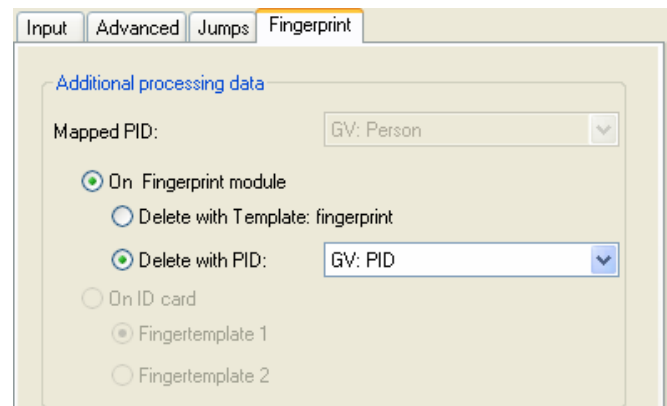
Field funktion: Fingerprint: Finger template delete.

Global variable with PID: GV: PID

Best scan run of: 0 times (2..10) 0 = Once

Info: PID system variable than previously loaded template: delete fingerprints if PID (global variable) fingerprint module. Finger templates from the ID card whatsoever.

Select the function as they want to delete a finger template.



Input Advanced Jumps Fingerprint

Additional processing data

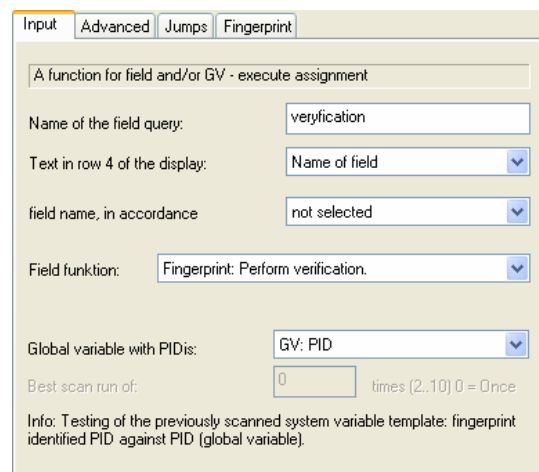
Mapped PID: GV: Person

☒ On Fingerprint module
☐ Delete with Template: fingerprint
☒ Delete with PID: GV: PID
☐ On ID card
☐ Fingertemplate 1
☐ Fingertemplate 2

5.7.38. Fingerprint: Perform verification

The identification of the user is done with RFID chip or choosing from a list.
After the identification scanning the user the finger and the matching is only with the templates of this PID. The matching to perform with saved templates on Mifare RFID chips or saved templates on the Biokey module.

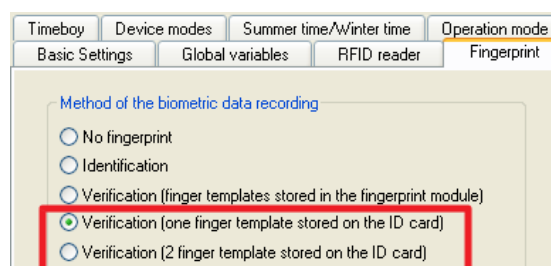
After successful matching (Verification) give the function the PID back.
The PID you can save in a record field and/or save in a GV.



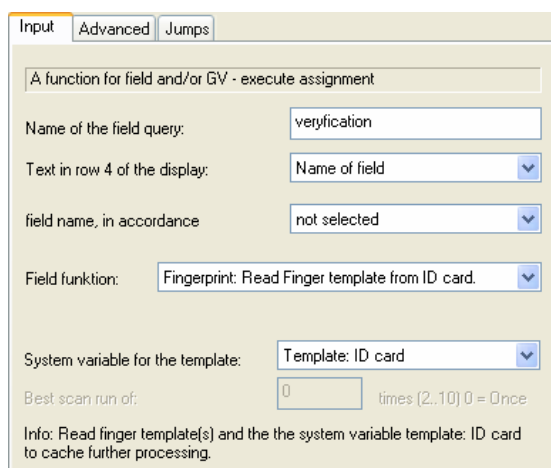
5.7.39. Fingerprint: Read Finger template from ID card

This function is using if you want read a template from a Mifare RFID chip.

Condition is the selected function "Verification (one or two finger templates stored on ID card)"



The template to be saved in the system variable, "Template: ID card".



5.8. Data on Card

5.8.1. Allgemeine Informationen

Mit der Funktion Data on Card ist es möglich, Daten mit einer individuellen Struktur auf einen Transponder zu schreiben.

Diese Daten werden in Form einer Liste von Ihrer Anwendung zur Verfügung gestellt.

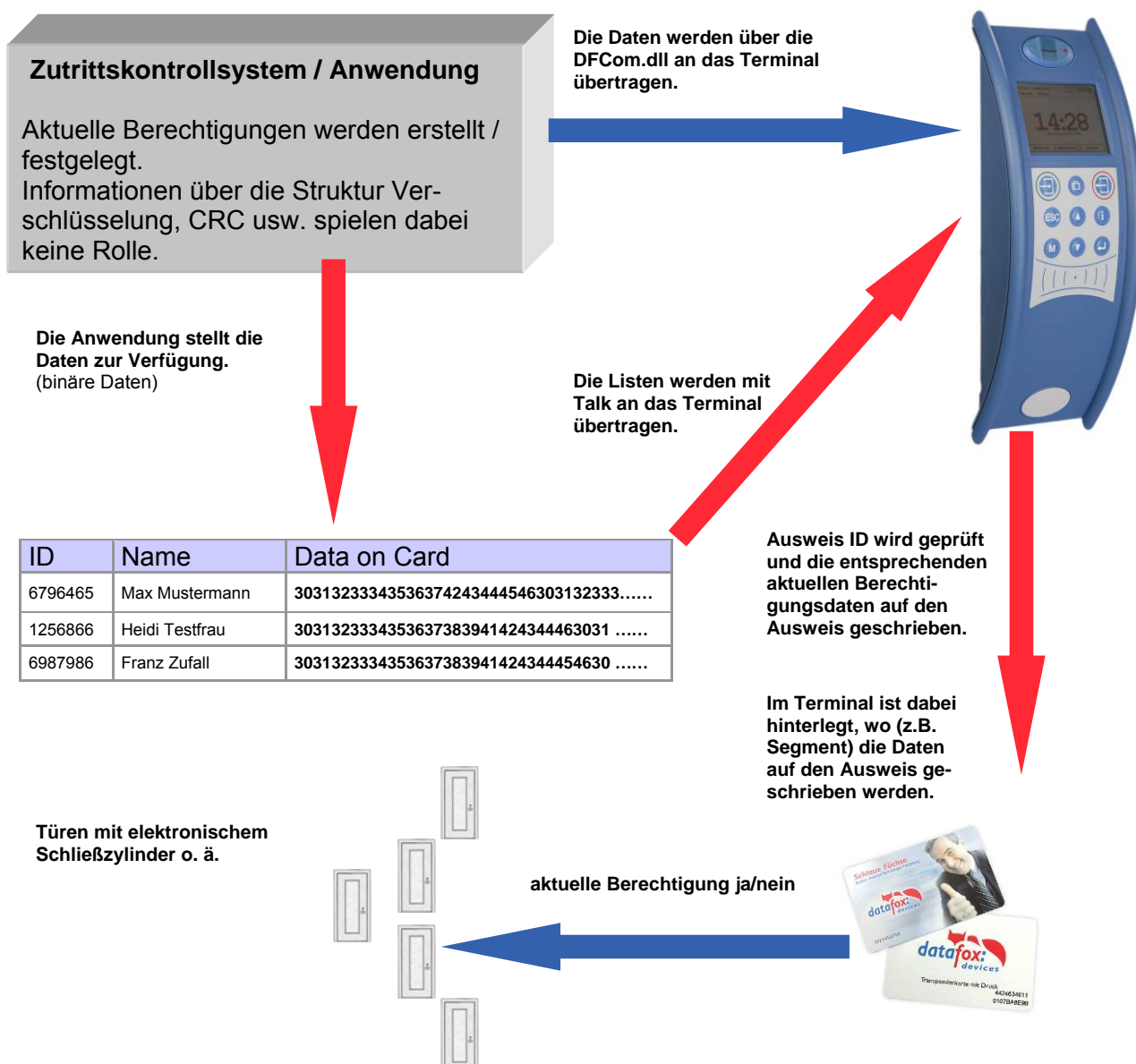
Diese Liste wird dann in das Terminal geladen und beim Vorhalten eines Transponders werden die Daten dann auf diesen geschrieben.

Folgende Transponderverfahren unterstützen die Funktion Data on Card:

- Mifare
- Legic
- iCode
- MyD

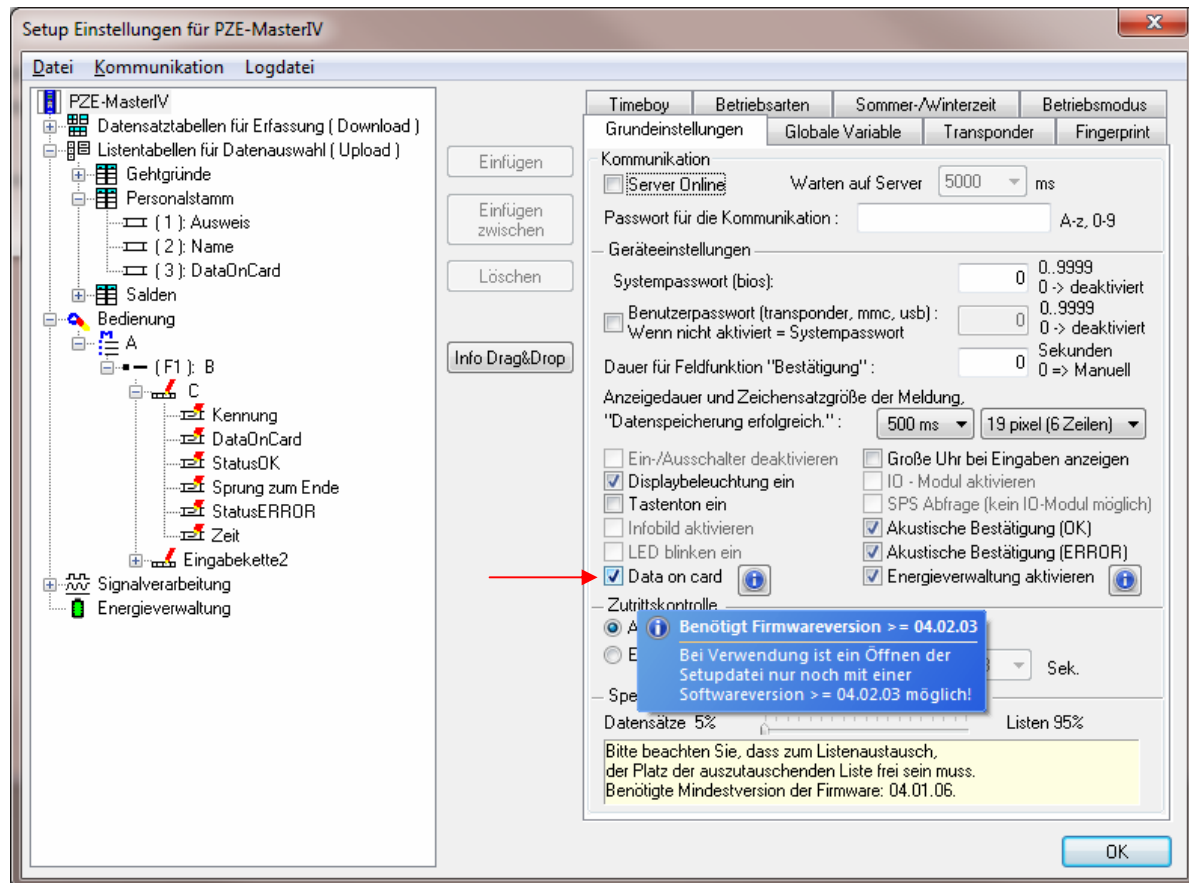
Folgendes Beispiel:

In Gebäuden mit „elektronischen“ Schließzylinder soll die aktuelle Tagesberechtigung für den Zutritt auf eine Transponderkarte geschrieben werden.



5.8.2. Einrichtung Data on Card

Data on Card ist eine Option des Gerätes bei der aus Listen Daten auf einen Transponder geschrieben werden können. Die Option muss bei der Bestellung mit angegeben werden. Bei Geräten die diese Option nicht haben, wird beim Ausführen eine Fehlermeldung angezeigt.



Data on Card arbeitet in **3 Schritten**:

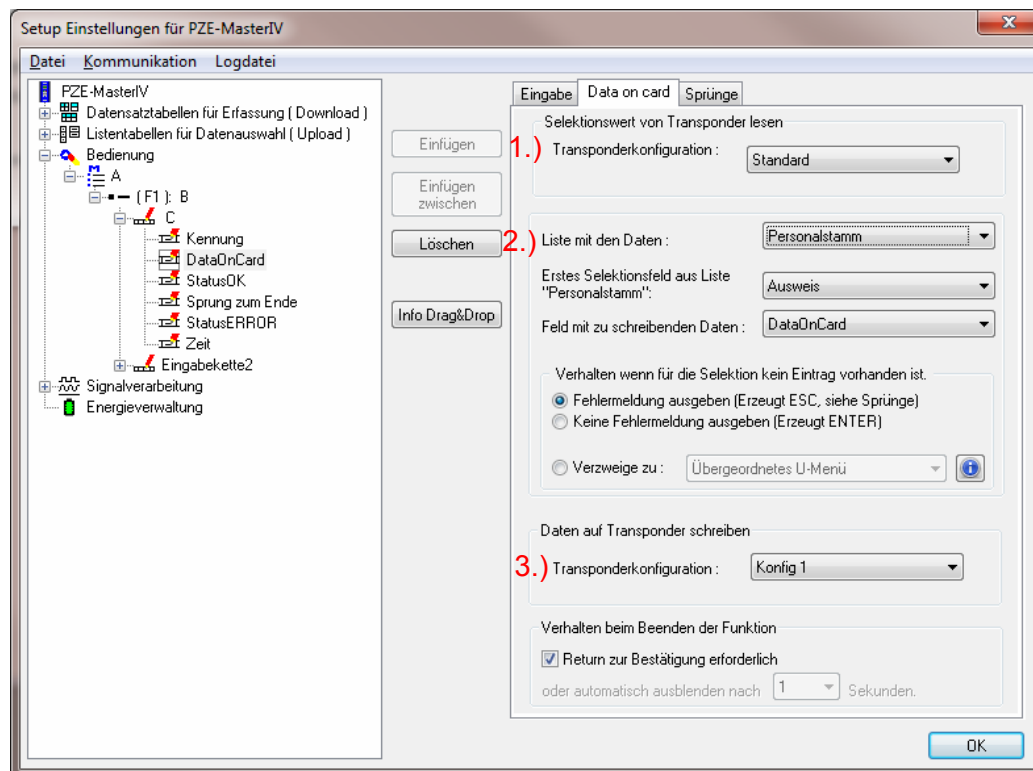
- Lesen eines Wertes vom Transponder, z.B. Seriennummer.
- Der Wert wird zum Selektieren einer Liste mit Binärfeld benutzt um die binären Daten zu lesen.
- Die binären Daten werden auf den Transponder geschrieben.

Der Rückgabewert der Funktion Data on Card für GV oder Datensatzfeld ist der Wert aus dem ersten Schritt „Lesen eines Wertes vom Transponder“. Bei Fehlern wie „der Wert wird nicht in der Liste gefunden“ oder das „Schreiben auf den Ausweis schlug fehl“ erzeugt die Funktion ein ESC. Über die Seite Sprünge kann dann entschieden werden, wie in der Eingabekette weiter gearbeitet wird.

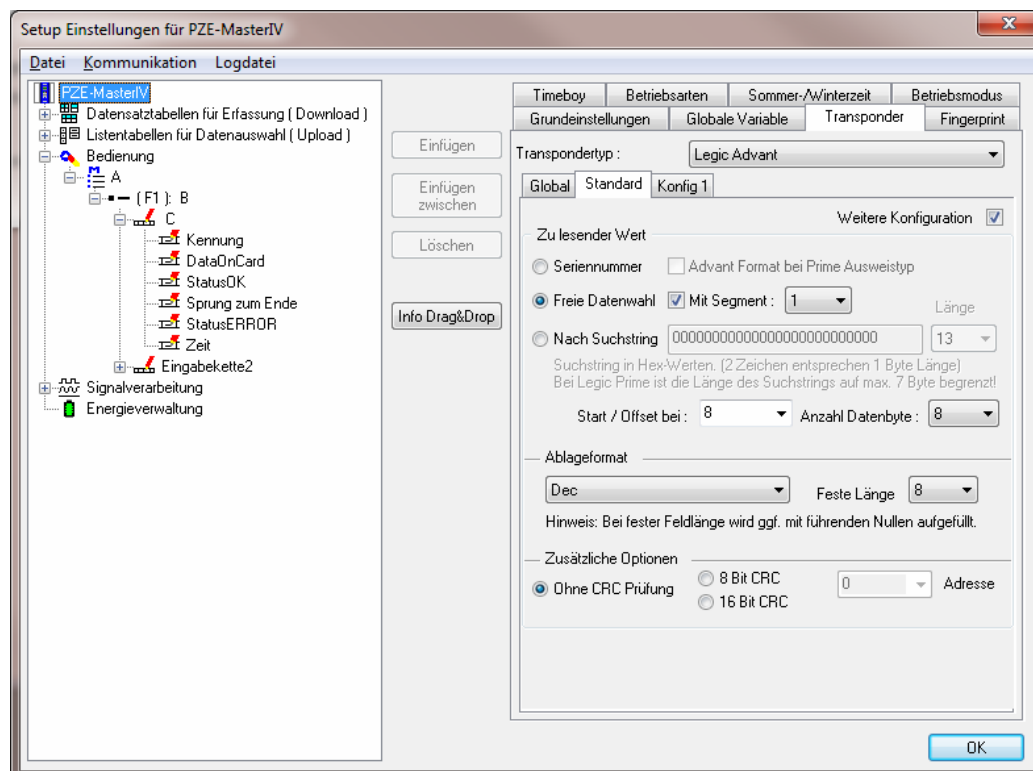
Die binären Felddaten innerhalb der Datei, die das DatafoxStudioIV importiert und überträgt, sind als Hexzeichenfolge anzugeben. Beim importieren über die DLL sind die Daten als Binärdaten zu übergeben.

Über die Bearbeitungsfunktionen von Listendaten DFCGetField, DFCSetField arbeiten Sie mit Zeichenfolgen, wobei die Firmware eine Konvertierung der Hexstrings in und zu den Binärdaten vornimmt.

Einstellungen bei Data on Card



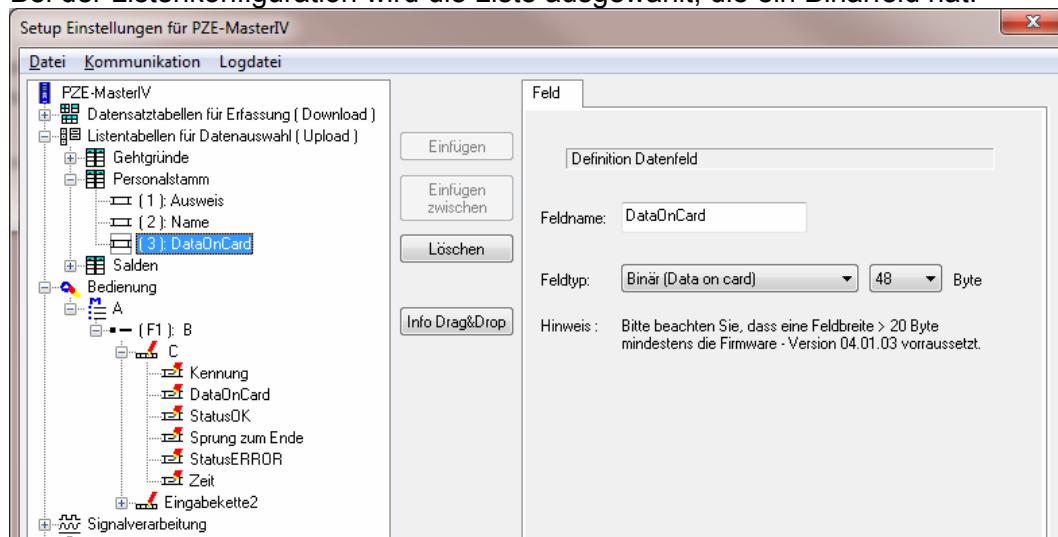
1.) Transponder Konfiguration Lesen



Die Transponderkonfiguration für das Lesen kann frei gewählt werden. Sie muss aber zuvor in den Tranpondergrundeinstellungen definiert worden sein.

2.) Listenkonfiguration

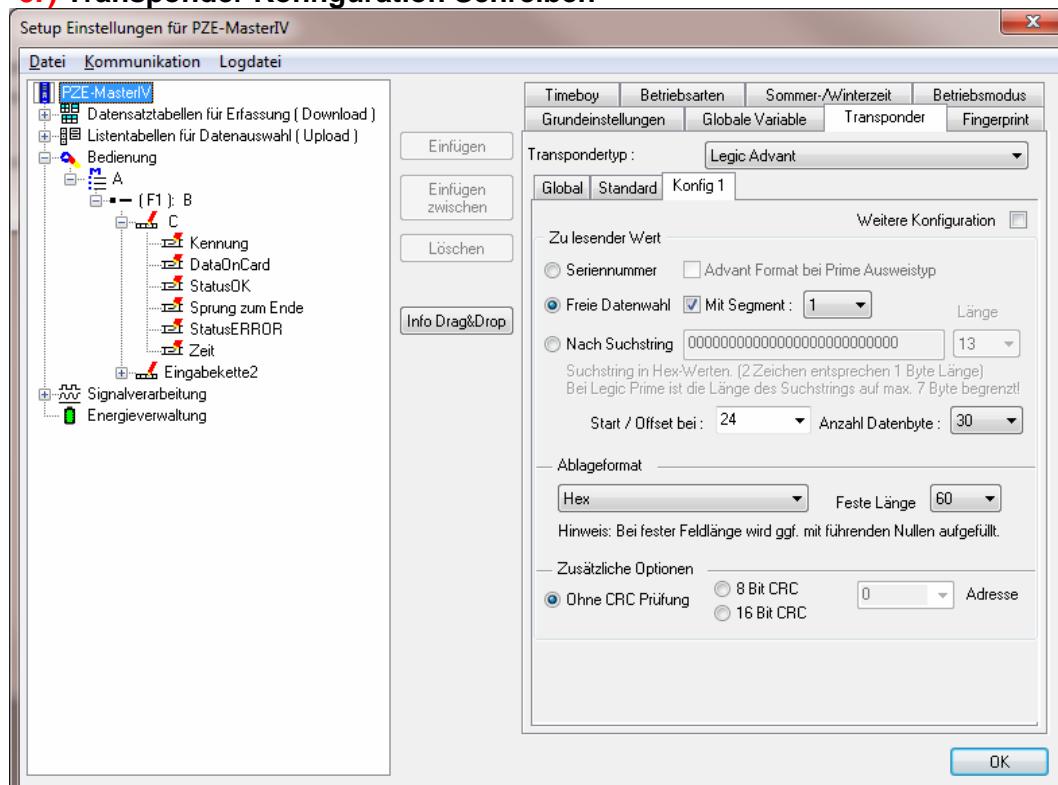
Bei der Listenkonfiguration wird die Liste ausgewählt, die ein Binärfeld hat.



In diesem Beispiel ist im Feld Ausweis der Wert vom Transponder Lesen, der in der Liste gesucht wird. Die zu schreibenden Daten sind im Feld Data on Card vom Typ Binär. Die max. Feldgröße darf 220 Byte nicht überschreiten.

Anschließend kann noch bei Listenfehlern die weitere Verfahrensweise festgelegt werden.

3.) Transponder Konfiguration Schreiben



Die Transponderkonfiguration für das Lesen kann frei gewählt werden, sie muss aber zuvor in den Tranpondergrundeinstellungen definiert worden sein.



Hinweis:

Zuerst die Transponderkonfiguration fertig stellen, dann die Liste mit dem Binärfeld anlegen und zum Schluss die Feldfunktion Data on Card parametrieren.

Beispiel für Data on Card:

Ausweis mit Seriennummer **1848989745**

Listeneintrag für **1848989745** in der Datei vor dem Übertragen in das Gerät

Feld ID Feld Data(Binärfeld) hier als Hex-Bytes

1848989745 30313233343536373839414243444546303132333435363738394142434445463031323334353637383941

Daten nach Konvertierung bzw. innerhalb des Gerätes

Feld ID Feld Data(Binärfeld) hier binär

1848989745 0123456789ABCDEF0123456789ABCDEF0123456789A

Auf den Ausweis werden folgende Daten geschrieben

0123456789ABCDEF0123456789ABCDEF0123456789A

Binär sehen die Daten so aus:

0x30, 0x31, 0x32, 0x33, 0x34

5.9. Extended Jump Function in Input Sequences

The extension of the jump functions allows creating a setup suitable for many application possibilities. Until now, the jumps have been limited to the corresponding input sequence and jumping to the parent submenu or the next input sequence.

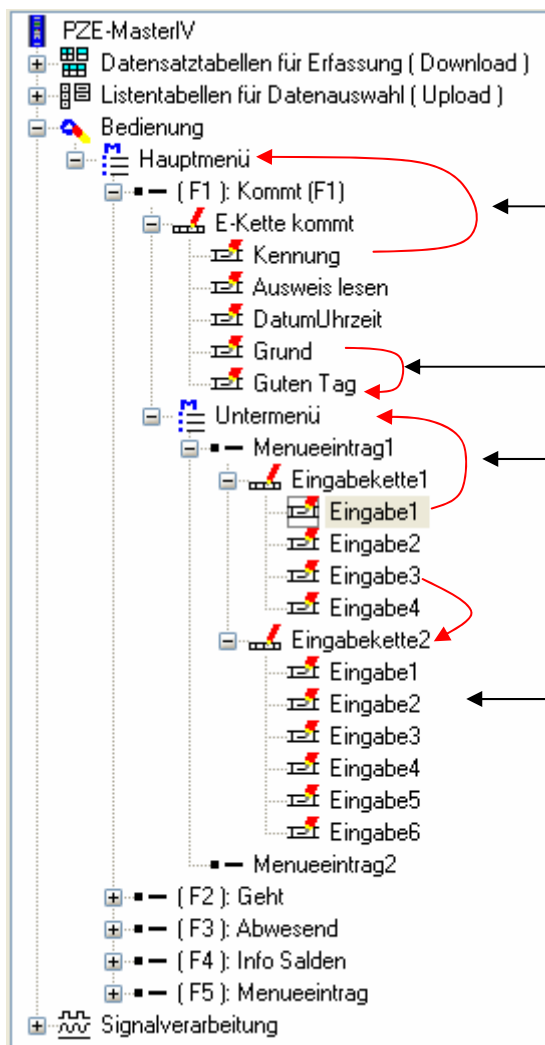
The improvements concern the following functions:

Jump to:

- Main menu
- Function key X
- Parent submenu
- Next input sequence or submenu
- Next input field
- Input sequence X
- Input field X

Abbreviation:

F-key:	Function key
S-menu:	Submenu
I-sequence:	Input sequence
I-field:	Input field



Sample Figure for Jumps

Jump to main menu
via ESC or conditional jump

Jump to the next input field via ESC

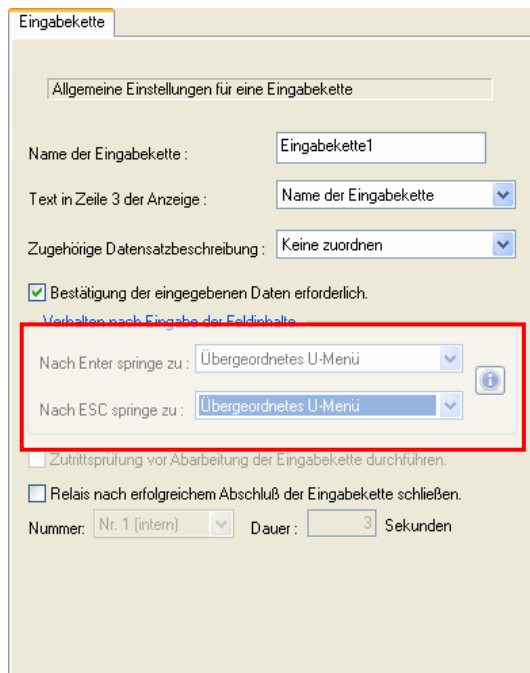
Jump to parent submenu

Next input sequence / submenu

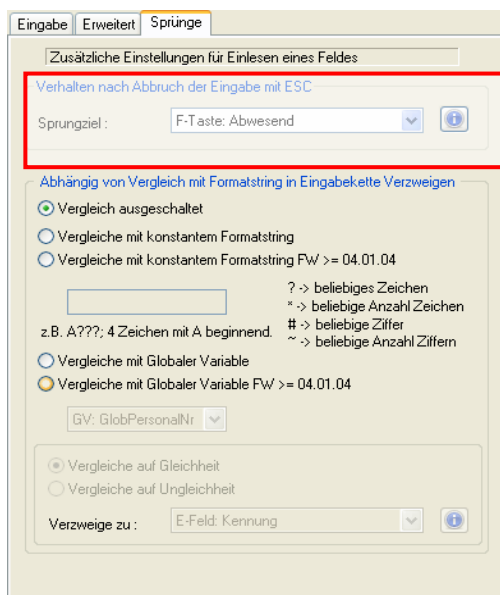
Jumps can be used for the following applications:

1. Behavior after entering field contents.
2. Behavior after canceling an input by ESC.
3. Branching in input sequence, dependent on comparison with format string.
4. Behavior at leaving the menu.

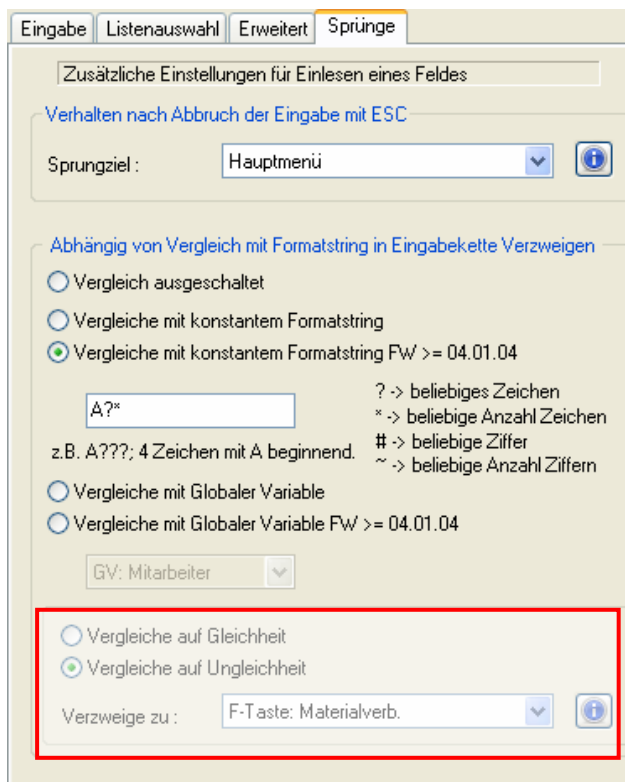
5.9.1. Behavior after Entering Field Content



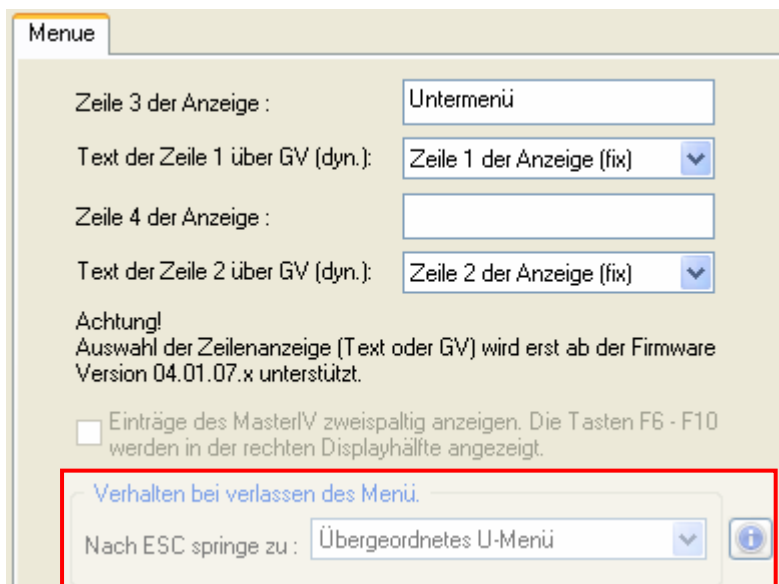
5.9.2. Behavior after Canceling an Input by ESC



Branching in input sequence, dependent on comparison with format string.



5.9.3. Behavior at Leaving the Menu



6. The RFID Technology (Transponder)

6.1. General Information on RFID

For contactless identification (RFID = Radio Frequency IDentification), transponders are distinguished into two types:

Passive Transponders

Passive transponders are systems which obtain the energy required for communication and processing internal processes solely from the field of the read-write unit.

Passive responders do not need an own power supply but can only work on short distances.

The best known type is the Radio Frequency Identification RFID. Typical applications: Identification of objects, pet registration chips or chip cards for access control systems. An active sensor (in connection with a PC) reads and decodes data the passive transponder sends.

As no own power supply is required, the consequences are very small dimensions enabling the installation of passive transponders in small casings. Thus, objects and persons can easily be equipped with an electronically readable data medium.

Active Transponders

Active systems have an own power supply. Either they have an built-in battery or are connected to an external electrical grid. This allows longer communication ranges, the management of larger data storage devices and the operation of built-in sensor technology. Simple active transponders are used for example for the identification of objects or persons:

This kind of transponders is not described here. An active transponder currently supported by Datafox is Simons & Foss.

6.2. RFID Methods Supported by Datafox

The amount of information which can be stored on a transponder depends on the RFID type used and the corresponding reading method.

Simple methods only support a unique serial number, the so-called ID. This ID can only be read. Complex methods offer different segments and sectors which are protected by a password partly. These segments can be read and written. Thus, apart from the ID-card number also additional information as status, departmental affiliation, personal data like blood group, access authorizations, bank accounts etc. can be stored at the RFID medium.

The following table gives an overview of the RFID methods supported by Datafox.

Reader	Reading method	Frequency	Tech. Data	Description
TSR32	Unique / EM4102	125 kHz	Serial number only	Unique / EM4102 is mere reading method. The number of the card is a globally unique ID and is used in all imaginable fields. A 64 bit information is stored on the card with the unique ID using only 40bit. The remaining bits serve as checksum.
	Hitag1	125 kHz	64 segments, 4 byte each 0 = serial number 1 - 31 = passwords, 32 to 63 = freely available	Hitag1 is organized in 16 blocks with 4 segments each. Each segment is 32 bits long. The block numbers 4 to 7 can be protected by password (secret) or used freely (public). The free segments can be used, for example, for saving a company code, a card number, account for cafeterias, etc.
	Hitag2	125 kHz	8 segments, 4 byte each 0 = serial number 1 to 3 = passwords, 4 to 8 = freely available	Hitag2 is organized in 8 segments. Each segment is 32 bits long. The free segments can be used, for example, for saving a company code, a card number, account for cafeterias, etc.
	HitagS	125 kHz	Serial number (segment 0) and, depending on type, segments 1 - 63 freely available.	This method is distinguished into "HitagS H32", "HitagS H56" and "HitagS H48". HitagS H32 means that this transponder only has a 32 bit value, the serial number of the card (see Unique). H56 means that the transponder has 8 registers with a 32 bit value each, in total 256 bit (see Hitag2). H48 means that the transponder has 64 registers with 32 bit each, in total 2048 bit (see Hitag1). The free segments can be used, for example, for saving a company code, a card number, account for cafeterias, etc.
	Titan / EM4450 (Hewi)	125 kHz	34 segments: 0 to 2 = passwords 3 to 31 = freely available 32 to 33 serial/device ID	Titan (EM4450) is organized in 34 segments. Each segment is 32 bits long. The serial number is contained in segment 32. The free segments can be used, for example, for saving a company code, a card number, account for cafeterias, etc.
	DOM Hitag1	125 kHz	Reading serial number	Hitag1 with crypto processor, therefore only serial number can be read

	DOM Hitag2	125 kHz	Reading serial number	Hitag1 with crypto processor, therefore only serial number can be read
Prox-Point® Plus OEM Module 4065	HID Prox-Point	125 kHz	ID card number only	Facility code and card number or card number only, depending on the card format from 26 bit format to 84 bit format Correct number determination only for public formats H10301, H10302 and H10304. For all non-public formats the binary value incl. parity bit is provided in hexadecimal format.
IClass OEM 50	HID IClass	13.56 MHz	ID card number only	Facility code and card number or card number only, depending on the card format from 26 bit format to 84 bit format Correct number determination only for public formats H10301, H10302 and H10304. For all non-public formats the binary value incl. parity bit is provided in hexadecimal format.
Mifare Easy	Mifare Classic	13.56 MHz	Serial number and 16 sectors, each with a reading and writing password Available as 1Kbyte and 4Kbyte variant	Mifare Classic 1k is organized in 16 sectors, each with 4 blocks with 16 byte per block. Mifare Classic 4k is organized in 32 sectors, each with 4 blocks with 16 byte per block and in 8 sectors, each with 16 blocks and 16 byte per block. Every 4th block serves for administrating the data on the transponder. It contains a password for reading and writing permissions divided into a key A and a key B, each 6 byte long, and the "Access Conditions" where the sector formats are defined. Depending on the application, all blocks of a sector can be available in default format (i.e. key A is the reading and writing protection key) or in Data or Value format (i.e. key A is the reading key and key B the master key for reading and writing). Advantages are the high speed and the large storage volume making the transponder suitable for biometry.
	Mifare Desfire	13.56 MHz	Serial number only	Data are available in encrypted form in a file system. Access via applications and file.
	Mifare Ultralight	13.56 MHz	Serial number only	Mifare Ultralight consists of 16 pages with 4 byte each and has a 7 byte serial number.
TWN3 Multi-ISO (available from the beginning of 2012)	Mifare Classic	13.56 MHz	Serial number and 16 sectors, each with a reading and writing password Available as 1Kbyte and 4Kbyte variant	Mifare Classic 1k is organized in 16 sectors, each with 4 blocks with 16 byte per block. Mifare Classic 4k is organized in 32 sectors, each with 4 blocks with 16 byte per block and in 8 sectors, each with 16 blocks and 16 byte per block. Every 4th block serves for administrating the data on the transponder. It contains a password for reading and writing permissions divided into a key A and a key B, each 6 byte long, and the "Access Conditions" where the sector formats are defined. Depending on the application, all blocks of a sector can be available in default format (i.e. key A is the reading and writing protection key) or in Data or Value format (i.e. key A is the reading key and key B the master key for reading and writing). Advantages are the high speed and the large storage volume making the transponder suitable for biometry.
	Mifare Desfire	13.56 MHz	Serial number and file system with reading and writing password Available as 2, 4, 8Kbyte and 72Kbyte variant	Data are available in encrypted form in a file system. Access via applications and file. Depending on the card type, from 2 kByte up to 72 kByte, 28 applications, each with 13 keys/passwords and 32 files per application are possible. Mifare Desfire is one of the safest transponder methods on the world.
	Mifare Ultralight	13.56 MHz	Serial number only	Mifare Ultralight has 64 byte capacity, consists of 16 pages with 4 byte each and has a 7 byte serial

				number.
	Mifare Ultralight C	13.56 MHz	Serial number only	Mifare Ultralight C has 192 byte capacity, consists of 48 pages with 4 byte each and has a 7 byte serial number. User data can be read and written in a range of 35 pages (148 byte). Mifare Ultralight C has a crypto processor using 3DES encryption.
	Mifare Plus SL1 and SL2	13.56 MHz	Serial number and 16 sectors, each with a reading and writing password (Security level 1 and 2 only)	MifarePlus is structurally equivalent to a Mifare Classic but it is available with different security levels. Security level 1 4 byte UID (may exist more than once) and 6byte keys Security level 2 7 byte UID (globally unique) and 16byte keys
	Mifare Plus SL3	13.56 MHz	Serial number only (security level 3)	7 byte UID (globally unique) / Access only with SAM (crypto processor unit) via APDUs (direct transponder commands)
	I Code SLI, SLI-S, SLI-L	13.56 MHz	Serial number and 8 – 64 blocks, 4 byte each	UID Mode 40 bit UID Block mode (4 byte per block) 8, 28, 32, 40 and 64 blocks depending on card chipset
	ICODE UID	13.56 MHz	Serial number and 12 byte data	Storage capacity of 96 bit / 12 byte. UID (40 bit) USER DATA (192 bits) CRC16 of user data (16 bit) Destroy Code(24 bit) The UID (serial number) cannot be altered.
	ICODE EPC	13.56 MHz	12 byte data only	Storage capacity of 96 bit / 12 byte. USER DATA (136 bits) CRC16 of user data (16 bit) Destroy Code(24 bit) The card has no UID (serial number).
	MyD	13.56 MHz	Serial number and 96 – 1024 blocks, 8 byte each	MyD is a transponder of the company Infineon and can have up to 10 kByte (1024 blocks). The cards have a serial number and a data area. Similar to Mifare, block 0 is the serial number.
Primo110	Legic Prime	13.56 MHz	Serial number and 256 or 1024Byte	Legic is only used in the German-speaking regions. Segmented and not segmented ID cards are available. For a not segmented card, the data are read by means of a position and length information. For segmented cards, besides the length information the segment must be provided from which the data are to be read.
Primo130	Legic Prime	13.56 MHz	Serial number and 256 or 1024Byte	Legic is only used in the German-speaking regions. Segmented and not segmented ID cards are available. For a not segmented card, the data are read by means of a position and length information. For segmented cards, besides the length information the segment must be provided from which the data are to be read.

	Legic Advant	13.56 MHz	Both Advant types are supported. ISO14443 and ISO15693 Serial number and 128 to 4096Byte	Segmented and not segmented ID cards are available. For a not segmented card, the data are read by means of a position and length information. For segmented cards, besides the length information the segment must be provided from which the data are to be read. A segment can also be selected via a search string.
i-Button	i-Button	Contact-based method	Fixed 15-digit serial number	This is a contact-reading method. The i-button only has a serial number which is read at contact with the transponder.
Smart Relay	SimonsVoss	25 kHz	10-digit number 1 to 5 = facility number 6 to 10 = card	SimonsVoss is an active contactless reading method. Each card has a unique 10-digit decimal code. Digit 1-5 is the company code, digit 6-10 the card number. Up to 8000 cards with corresponding profiles can be stored on a SmartRelay.
XS070	Nedap	125kHz	Serial number only	Nedap is a mere reading method; the cards only provide a number

For more information on the options supported for a reading method see the manual of the respective device. The manuals are available for download as PDF on our homepage.

If the reading method you need is not listed in the table, please contact us. We are expanding the range of supported reading methods permanently and also for customer projects.

6.3. Most Important RFID Reading Methods

For some transponders, a detailed description of the settings and the technology of the transponder type is necessary.

This concerns:

- Legic Prime
- Legic Advant
- HID-ProxKey
- HID-iClass



Note:

Simple transponder methods are not explained further. For additional information on RFID technology see the product DVD.

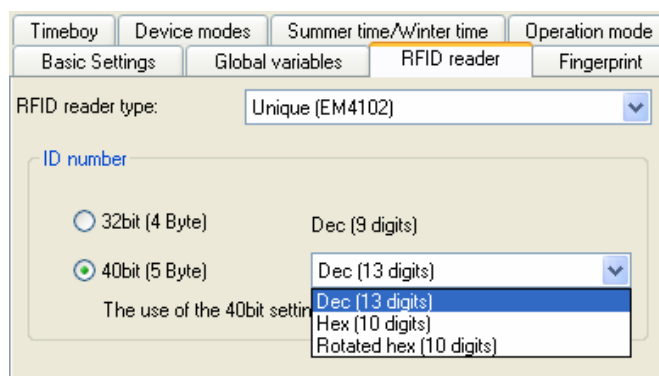
<_Datafox DVD\Werbe- und Infomaterial\MasterIV Werbe-Info-Material aktuell\1_Prospekte und Produktbeschreibungen/ Datafox _MasterIV-Serie, Option Kommunikationsarten-Übersicht V1.0.pdf>

6.3.1. 125 kHz RFID Reader

6.3.1.1. Unique

Unique is mere reading method. The number of the card is a globally unique ID and is used in all imaginable fields. A 64bit information is stored on the card with the unique ID using only 40bit. The remaining bits serve as checksum.

When using Unique, the value of the ID card can be selected in the AESetup as 40 or 32bit value for further processing.



6.3.1.2. Hitag1

Hitag1 is organized in 16 blocks with 4 segments each. Each segment is 32 bits long. The block numbers 4 to 7 can be protected by password (secret) or used freely (public).

Caution!

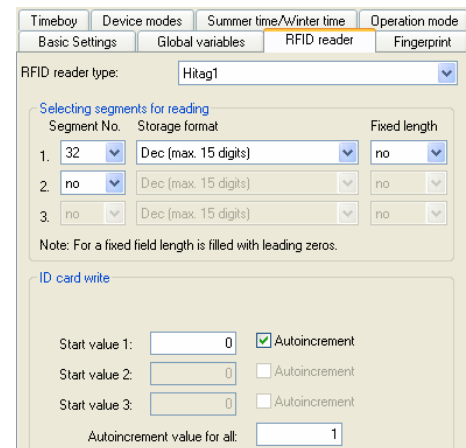
The AESetup only supports the segments **0** and **8..63**. The segments from **32..63** can always be read and written, the segment **0** can always be read. Depending on the content of the segments **1..7** it could happen that reading or writing of the segments **8..31** is not possible.

	Block Number	Segment Number	Data
Public	0	0	Serial number
Secret		1	Configuration
		2	Key A
		3	Key B
Secret	1	4	Logdata 1B
		5	Logdata 0A
		6	Logdata 1A
		7	Logdata 0B
Secret	2 and 3	4 * BlockNumber + 0	User data
		4 * BlockNumber + 1	User data
		4 * BlockNumber + 2	User data
		4 * BlockNumber + 3	User data
Secret or Public depending on con- figuration	4 to 7	4 * BlockNumber + 0	User data
		4 * BlockNumber + 1	User data
		4 * BlockNumber + 2	User data
		4 * BlockNumber + 3	User data
Public	8 to 15	4 * BlockNumber + 0	User data
		4 * BlockNumber + 1	User data
		4 * BlockNumber + 2	User data
		4 * BlockNumber + 3	User data

3 segments at most are available for simultaneous processing. They can be selected in the AESetup under transponder.

The “*Storage format*” determines for what the 32 bit value is used.

With the option “*Fixed Length*” the card value read is reduced to the given number of digits and leading zeros (0) are added if necessary.



For writing the cards an initial value can be determined per segment. If the option “*Autoincrement*” is activated next to the respective initial value, the current segment value is increased by the “*autoincrement value*” after each writing process. The initial values of the segments can be edited in the device BIOS. The “*autoincrement value*” is only displayed in the device BIOS but cannot be changed at the device.

6.3.1.3. Hitag2

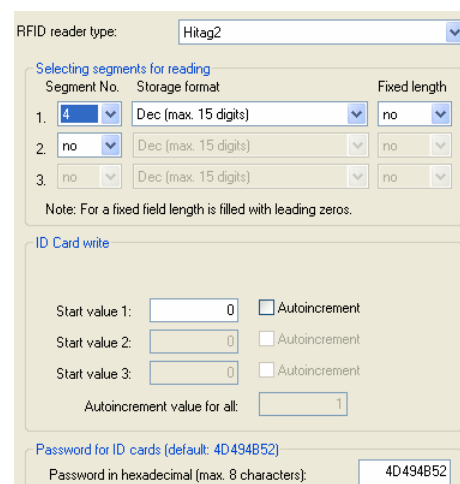
Hitag2 is organized in 8 segments. Each segment is 32 bits long.

Page	Content
0	Serial number
1	Password RWD
2	Reserved
3	8 bit configuration, 24 bit password TAG
4	Read/write page
5	Read/write page
6	Read/write page
7	Read/write page

3 segments at most are available for simultaneous processing. They can be selected in the AESetup under transponder.

The “*Storage format*” determines for what the 32 bit value is used.

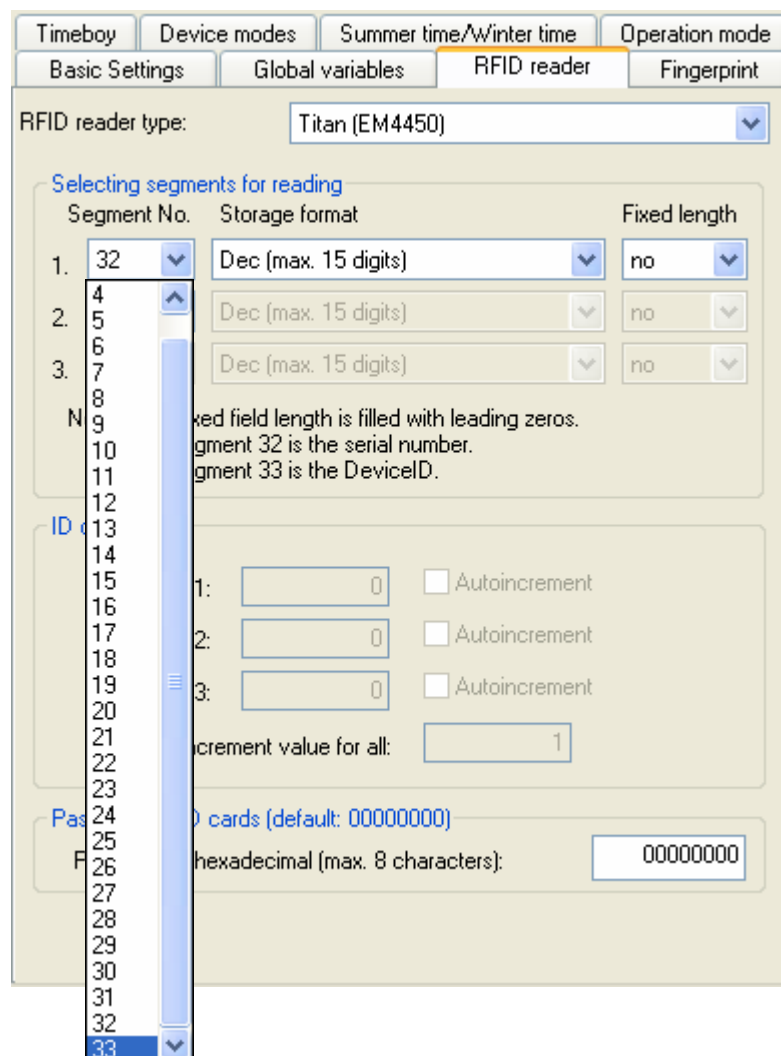
With the option “*Fixed Length*” the card value read is reduced to the given number of digits and leading zeros (0) are added if necessary.



For writing the cards an initial value can be determined per segment. If the option “*Autoincrement*” is activated next to the respective initial value, the current segment value is increased by the “*autoincrement value*” after each writing process. The initial values of the segments can be edited in the device BIOS. The “*autoincrement value*” is only displayed in the device BIOS but cannot be changed at the device. The writing of cards can be protected by a “*menu password*”.

6.3.1.4. Titan

Titan (EM4450) is organized in 34 segments. Each segment is 64 bits long. The serial number is contained in segment 32.



Segment No.	Storage format	Fixed length
1. 32	Dec (max. 15 digits)	no
2. 5	Dec (max. 15 digits)	no
3. 6	Dec (max. 15 digits)	no

Fixed field length is filled with leading zeros.
Segment 32 is the serial number.
Segment 33 is the DeviceID.

1: 0 ☐ Autoincrement
2: 0 ☐ Autoincrement
3: 0 ☐ Autoincrement

Increment value for all: 1

Cards (default: 00000000)
hexadecimal (max. 8 characters): 00000000

Three segments at most are available for simultaneous processing. They can be selected at DatafoxStudioIV under transponder via the segment number. With the "Storage format" it is set for what the 64 bit value is to be used. With "Window length" the card value read is cut to the given number of characters and leading zeros (0) are added if necessary.

For writing the cards an initial value can be determined per segment. If the option "Autoincrement" is activated next to the respective initial value, the current segment value is increased by the "autoincrement value" after each writing process.

The initial values of the segments and the "autoincrement value" can be edited in the device BIOS.

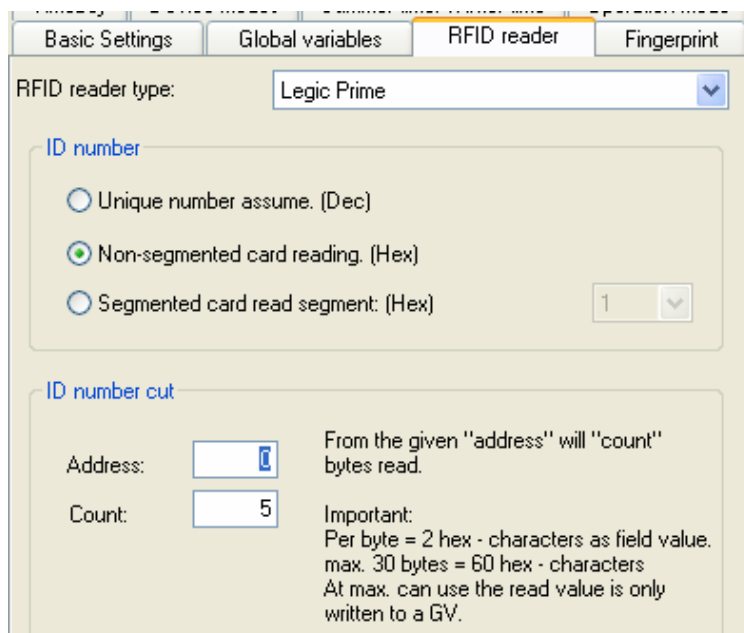
6.3.2. Legic Transponder Technology

Legic is a 13.56 MHz technology with two fundamentally different types Legic Prime and Legic Advant. The Advant has two different transfer versions ISO 14443A and ISO 15693. Datafox terminals only support the reading of ID cards. Writing values on the card is not possible.

6.3.2.1. Overview Prime and Advant

	Prime	Advant ISO15693	Advant ISO14443A
Hardware	Security chip with proprietary protocol	Microcontroller with ISO standard	Microcontroller with ISO standard
Transfer, advantages/disadvantages	ISO15693, better range, lower data transfer	ISO15693, better range, lower data transfer	ISO14443A, faster data transfer, worse range
Reader hardware in Datafox terminals	Primo100 (PHG) Primo130 (PHG)	Primo130 (PHG)	Primo130 (PHG)
Data structures	Freely definable data areas in segments	Freely definable data areas in segments	Freely definable data areas in segments
Special security per segment	Kaba Group Header	Access Segment Definition	Access Segment Definition

6.3.2.2. Important Settings in DatafoxStudioIV



Basic Settings | Global variables | **RFID reader** | Fingerprint

RFID reader type: Legic Prime

ID number

☐ Unique number assume. (Dec)

☒ Non-segmented card reading. (Hex)

☐ Segmented card read segment: (Hex) 1

ID number cut

Address:

Count:

From the given "address" will "count" bytes read.

Important:
Per byte = 2 hex - characters as field value.
max. 30 bytes = 60 hex - characters
At max. can use the read value is only written to a GV.

Devices with Primo 100 or Primo 130 are able to read the Legic Prime and configure the reader via this dialog.

Via this dialog, both readers can be configured, but only for Legic Prime cards.

Former setups still use this dialog. Due to compatibility reasons, it has not been removed.

Timeboy	Device modes	Summer time/Winter time	Operation mode
Basic Settings	Global variables	RFID reader	Fingerprint

RFID reader type: Legic Advant (also Prime) ▼

Global Default

Supported ID card type

☒ Legic Advant
 ☐ Legic Prime
 ☐ Legic Advant and Prime

☒ Advant ISO 14443 ID cards allowed
☒ Advant ISO 15693 ID cards allowed

Additional options

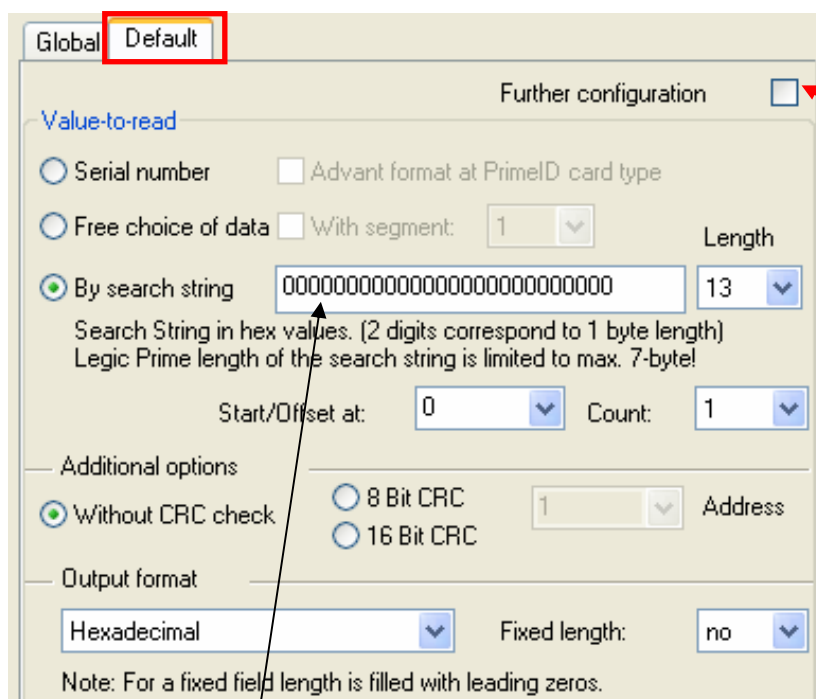
☐ Access to LEGIC Access segment definition (Advant only)
☐ Kaba Group Header (Only Prime)

Devices with Primo 130 are able to read Legic Prime and Legic Advant. You can select whether both types or just one type is to be read.



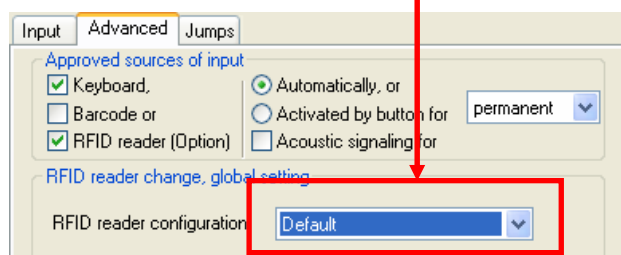
Note:

If you select the state field function in the power management of a Primo 130, you must expect a booting time of the reader of at least 2 seconds after tuning it on. During this time, it is not possible to read cards.



Here you can activate a further configuration and can choose in the field function "normal",

Read a segment and can search a value by a string in the segment.



Explanation of Parameters

Supported card type

- Legic Advant
- Legic Prime
- Legic Advant and Prime

Advant ISO 14443 format cards permitted

- If you select this option, Legic Advant cards with ISO 14443 format can be read, otherwise these cards are ignored.

Advant ISO 15693 format cards permitted

- If you select this option, Legic Advant cards with ISO 15693 format can be read, otherwise these cards are ignored.

Serial number – unique number of the card

Advent format for Prime card type

- Option Advant format for Prime card type refers to the value output according to Prime or Advant rules (byte 2 and 4 switched).

Free Data Selection

There are segmented and not segmented ID cards. The defined data area can be read from those cards.

Via Search String

A transponder has one or more segments, for example.

Each segment starts with an IAM number.

IAM number e.g.: 85 00 6B 00 1C

Example segment structure:

	IAM 85 0...	SSC		ID card number		X Data		IAM 73....	SSC					
Byte	1 - 5	6	7	8 - 13		x...		1 - 5	6					

Segment 1

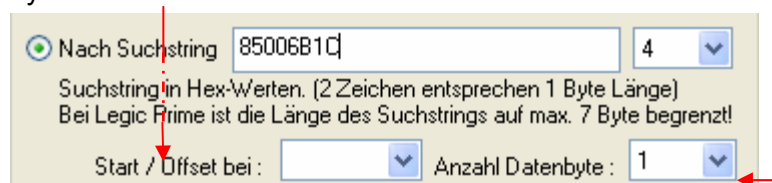
Segment 2

If the segments 1, 2 or 3 are not always identical for a customer, the right segment can be determined via a search string which contains the IAM number, for example.

During start / offset, you specify at which point of the data area you want to start reading. At number of data bytes, you specify the length of the data to be read. If it is read beyond the data area, the reader provides no data. The values set must correspond to the Legic card.

Example of a search string based on the IAM no.: 85 00 6B 00 1C.

If you want to read the **ID card number**, you have to set "Start/Offset" to 8 and "Number of data bytes" to 6.



Storage format:

Decimal: the read binary value of the card is converted to a decimal number. For this format, 8 byte/64 bit are possible at most.

Hex: the read binary value of the card is converted to a hexadecimal number. For this format, 20 byte are possible at most. Example: 74001B00004A

Reverse hex: the read binary value of the card is converted to a hexadecimal number. However, it is not read from front to back but from back to front. For this format, 20 byte are possible at most. Example: 4B0064000082

CRC (cyclic redundancy check),

If data areas are read where they are secured by a CRC, the reader can perform the check independently. For this purpose, the size and the address must be provided on the card. If the CRC is false, no data are provided.

Access to Legic Access Segment Definition,

is access to a segment with a strictly defined header structure oriented towards the Legic standard. This segment type provides a high security level of data.

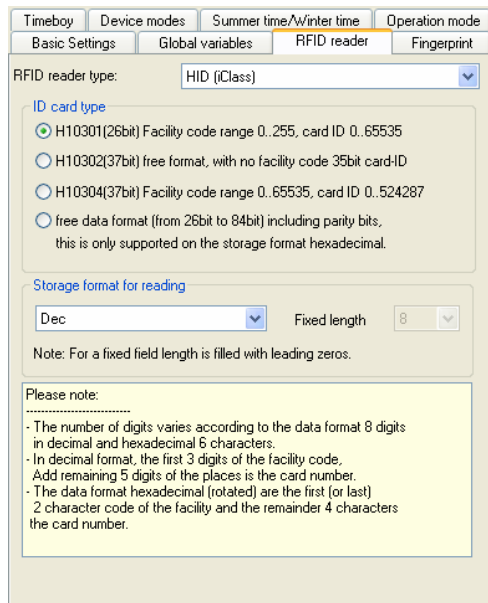
Kaba Group Header for Prime cards,

increases data security with additional CRCs of a segment. If the CRCs are false, no data are provided.

6.3.3. HID-ProxKey und HID-iClass



HID-ProKey and HID-iClass are two RFID methods which are compatible with each other in formats but use different transfer frequencies. Therefore, Datafox terminals have two different reader modules. The settings in DatafoxStudioIV only differ in the selection of the transponder types.



Timeboy Device modes Summer time/Winter time Operation mode

Basic Settings Global variables RFID reader Fingerprint

RFID reader type: HID (iClass)

ID card type

- ☒ H10301(26bit) Facility code range 0..255, card ID 0..65535
- ☐ H10302(37bit) free format, with no facility code 35bit card-ID
- ☐ H10304(37bit) Facility code range 0..65535, card ID 0..524287
- ☐ free data format (from 26bit to 84bit) including parity bits, this is only supported on the storage format hexadecimal.

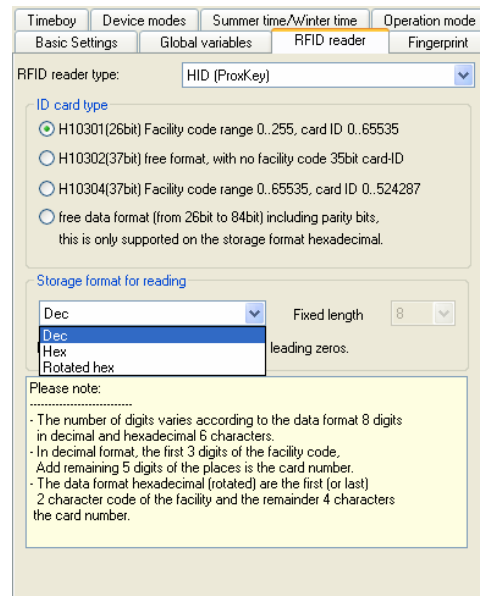
Storage format for reading

Dec Fixed length 8

Note: For a fixed field length is filled with leading zeros.

Please note:

- The number of digits varies according to the data format 8 digits in decimal and hexadecimal 6 characters.
- In decimal format, the first 3 digits of the facility code, Add remaining 5 digits of the places is the card number.
- The data format hexadecimal (rotated) are the first (or last) 2 character code of the facility and the remainder 4 characters the card number.



Timeboy Device modes Summer time/Winter time Operation mode

Basic Settings Global variables RFID reader Fingerprint

RFID reader type: HID (ProxKey)

ID card type

- ☒ H10301(26bit) Facility code range 0..255, card ID 0..65535
- ☐ H10302(37bit) free format, with no facility code 35bit card-ID
- ☐ H10304(37bit) Facility code range 0..65535, card ID 0..524287
- ☐ free data format (from 26bit to 84bit) including parity bits, this is only supported on the storage format hexadecimal.

Storage format for reading

Dec Fixed length 8

Dec Hex Rotated hex leading zeros.

Please note:

- The number of digits varies according to the data format 8 digits in decimal and hexadecimal 6 characters.
- In decimal format, the first 3 digits of the facility code, Add remaining 5 digits of the places is the card number.
- The data format hexadecimal (rotated) are the first (or last) 2 character code of the facility and the remainder 4 characters the card number.

Three public formats (H1030x) can be set in the Studio. All customer-specific formats must be read via a free data format. The data are output including parity bits because the parity bits and information are differently stored in the card for the different customer formats. Therefore, all data are output because they are unique. A validation of the values read could be executed within the input sequence by a positive list and thus the ID card number can be determined.

Data Format H10301(26bit) Facility Code Range 0..255, Card ID 0..65535

- Formatting decimal (3 digits facility code + 5 digits Card-ID)
- Formatting hexadecimal (2 characters facility code + 4 characters Card-ID)
- Formatting reverse hexadecimal (2 characters facility code + 4 characters Card-ID)

Example:

Facility code is 15 and Card ID is 5.

Decimal output value 015 00005

Hexadecimal output value 0F 00 05

Hexadecimal reverse output value 05 00 0F

Data Format H10302(37bit) Free Format, without Facility Code 35bit Card ID

Formatting decimal (max. 11 digits Card-ID)

Formatting hexadecimal (8 characters Card-ID)

Formatting reverse hexadecimal (8 characters Card-ID)

35 bit	←most significant bit	least significant bit→
Binary	111 1111 1111 1111 1111 1111 1111 1111	
Decimal	34 359 738 367	
Hexadecimal	07 FF FF FF FF	

Example:

Card ID is 5.

Decimal output value 00 000 000 005

Hexadecimal output value 00 00 00 05

Hexadecimal reverse output value 05 00 00 00

Data Format H10304(37bit) Facility Code Range 0..65535, Card ID 0..524287

- Formatting decimal (5 digits are the facility code, 6 digits the Card-ID)
- Formatting hexadecimal (4 characters are the facility code, 6 characters the Card-ID)
- Formatting reverse hexadecimal (4 characters are the facility code, 6 characters the Card-ID)

Example:

Facility code is 15 and Card ID is 5.

Decimal output value 000 015 000 005

Hexadecimal output value 00 0F 00 00 05

Hexadecimal reverse output value 05 00 00 0F 00

Free data format (from 26bit to 84bit)

Formatting hexadecimal (20 characters Card-ID)

84 bit	MSB
LSB	
Binary	1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111
	1111 1111 1111 1111 1111 1111
Decimal	019 342 813 113 809 551 615
Hexadecimal	0003 FFFF FFFF FFFF FFFF FFFF

Example:

Data from reader are 65535.

Decimal output value is not supported.

Hexadecimal output value 00 0000 0000 0000 0000 FFFF

Hexadecimal reverse output value is not supported.

6.3.4. 13,56MHz RFID Reader (ISO14443 u. ISO15693)

There are 2 RFID reader for 13,56 MHz available. See price list:

1. device with 13,56 MHz

ISO 14443 for Mifare-Desfire
ISO 14443 Mifare-Plus, -Classic, -Ultralight

This reader (TWN3 Mifare NFC) is due to the antenna design especially suitable for RFID - cards

2. device with 13,56 MHz

ISO 14443 for Mifare-Classic, -Plus, ...
ISO 15693 for ICode, My-D, Tag-it, ...

This reader (TWN5 Multi-ISO) is due to the antenna design especially suitable for RFID - Keychains and offers these the best reading range.

overview table with RFID technology

IEC/ISO	description	read	write	read	write
14443A	Mifare Mini	X	X	X	X
	Mifare Classic 1k und 4k	X	X	X	X
	Mifare Plus S und X	X	X	X	X
	Mifare Ultralight/ Mifare Ultralight C	X	X	X	X
	Mifare DESFire	X	X	X	X
	LegicAdvant (ISO14443 Typ) serial number only *	X*	-	X*	-
	Andere ISO14443 Typen (SmartMX) serial number only*	X*	-	X*	-
14443B	Calypso, CEPAS, Moneo serial number only *	X*	-	X*	-
14443-2	iClass, Piccopass serial number only *	X*	-	X*	-
14443-3	SRX Transponder von ST Microelectronics, serial number only *	X*	#	X*	#
15693	iCode	X	X	-	-
	My-D Vicinity	X	#	-	-
	Tag-it serial number only *	X*	#	-	-
	LegicAdvant (ISO15693 Typ) serial number only *	X*	-	-	-
	24LR16 / 24LR64 von ST Microelectronics, serial number only *	X*	#	-	-
	MB89R118/MB89R119 von Fujitsu, serial number only *	X*	#	-	-
	LRI 2k / 64k von ST Microelectronics, serial number only *	X*	#	-	-
	Andere ISO15693 Typen serial number only *	X*	-	-	-

* reading, serial number only.

writing, can be implemented on request.

6.3.4.1. ISO 14443A - Mifare Familie

6.3.4.1.1. Mifare Mini

The Mifare Mini is a compatible transponder for Mifare Classic 1k, only with a memory limit of 320 bytes. So he has not 16, but only 5 sectors.



Hint:

The Mifare Mini can be configured through the Mifare Plus basic settings in the DatafoxStudioIV.

6.3.4.1.2. Mifare Classic

Mifare is organized in 16 sectors a 4 blocks of 16 bytes. Every 4th Block is used to encrypt the data on the transponder and contains, divided into a Key-A and Key B, each 6 bytes long a password for write and read access, as well as the "Access Condition," in which the sector formats are defined. Depending on the application, all the blocks of a sector can be in the default format (ie, key A is the read and write protection key) or in the data or the value format, with key A, the read password and key B is the master key for reading and writing. The Datafox devices up to 4.1.4.xx version currently only support the default format.



Attention:

The value format is not supported.
The data format we can read only with KeyA.
The default format we can read and write with Key A.



Hint:

It is also possible to configure the Mifare Classic in DatafoxStudioIV with the MifarePlus basic settings. This is necessary when multiple types of Mifare need to be used for reading or even at DataOnCard.

6.3.4.1.3. Mifare Plus

Mifare Plus is similar to the data structure such as Mifare Classic. The first 16 sectors have the 4 blocks and the following ones have 16 blocks a 16-byte. The difference with the Mifare Classic is that one has different security levels (security level) which impact primarily on the cryptography. Similar to the Mifare Classic there are Key A and Key B with different lengths depending on the security level.

Security level	description
SL0	Delivery condition, the serial number can be read. Accessing the data areas is not possible. With the Card Master Key and Level switch key the security level can be configured.
SL1	Kompatibilitätsmodus zum MifareClassic, der Mifare Plus ist 100% kompatibel zum Mifare Classic und benutzt auch den Crypto1 Algorhytmus mit 6Byte(48Bit) Keys.
SL2 Only at Plus X	Data access as in MifareClassic with Cryptokey, but authentication is done via 16-byte AES-keys. In SL2 there are 2 Crypto1 Keys and 2 AES keys.
SL3	ISO1443-4 communication protocol and AES authentication and pre-shared keys, and MACing. There are two AES keys. Weiterhin ist zum Schutz der Seriennummer optional Random UID möglich. Furthermore optional random UID is possible to protect the serial number. Bei MifarePlus X erfolgt die Kommunikation über AES Verschlüsselung und beim MifarePlus S ist die Kommunikation unverschlüsselt. In MifarePlus X communication via AES encryption and the MifarePlus S the communication is unencrypted.


Note:

In Security Level 1 MifarePlus, Mifare Classic and Mifare Mini can be read and written, where the set sectors exist on the card.


Caution:

Mifare Plus cards can switch up their security levels. Reverse is not available. Changing the security level on MasterIV devices is not possible.

6.3.4.1.4. Mifare Ultralight

Mifare Ultralight uses in contrast to Mifare Classic and Plus no sectors and blocks, but only 16 pages of 4 bytes, which may be regarded as a segment in Hitag transponders. The simple Ultralight has no access protection or write protection only, the Ultralight C, however, can be protected by a 3DES authentication. In the delivery state this key is not set and the Ultra Light behaves like a simple ultralight.

The simple Ultralight has 16 pages with 4 bytes, only 12 pages can be written.

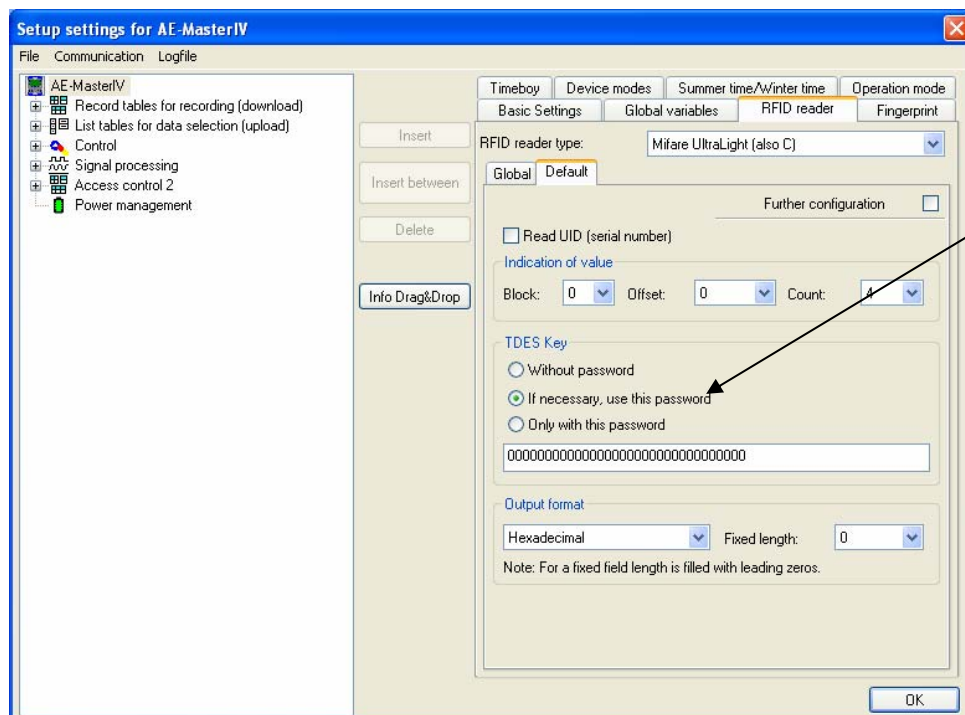
The Ultralight C has 48 pages, 40 of which can be written.


Caution:

Mixed operation of Mifare Classic and Mifare Ultralight is not possible due to the different data structures.


Note:

Mixed operation of Ultralight and Ultralight C is possible. Depending on the security requirements, the key usage is selected



key usage

6.3.4.1.5. Mifare DESFire

The Mifare DESFire is next to the Mifare Plus X one of the safest and most complex transponder. In contrast to the Mifare Plus or Classic the DESFire is not built on the solid data structures with sectors and blocks, but a file system as on a PC is used for DESFire.

There are two basic types of DESFire: the old DF40, referred to as native type and the successor series DF80, also known as the EV1.

The security relies on encryption, authentication negotiation of session keys and different communication modes. It is also possible, to protect the serial number (UID) by a random UID against unauthorized reading.

	MF3IC40	MF3IC21-EV1	MF3IC41-EV1	MF3IC81-EV1
Memory size	4k	2k	4k	8k
Free memory	4096 Bytes	2272 Bytes	4832 Bytes	7936Bytes
Max. number of Applications	28	28	28	28
max. Number of files per application	16	32	32	32
encryption	DES, TDES (DF4)	DES, TDES, (DF4 und ISO) 3KTDES, AES	DES, TDES, (DF4 und ISO) 3KTDES, AES	DES, TDES, (DF4 und ISO) 3KTDES, AES
Number of keys per application	14	14	14	14

The DF4 is based on the DES encryption algorithm, but with special NXP treatises.

ISO encryption is based on the DES algorithm according to the ISO standard.

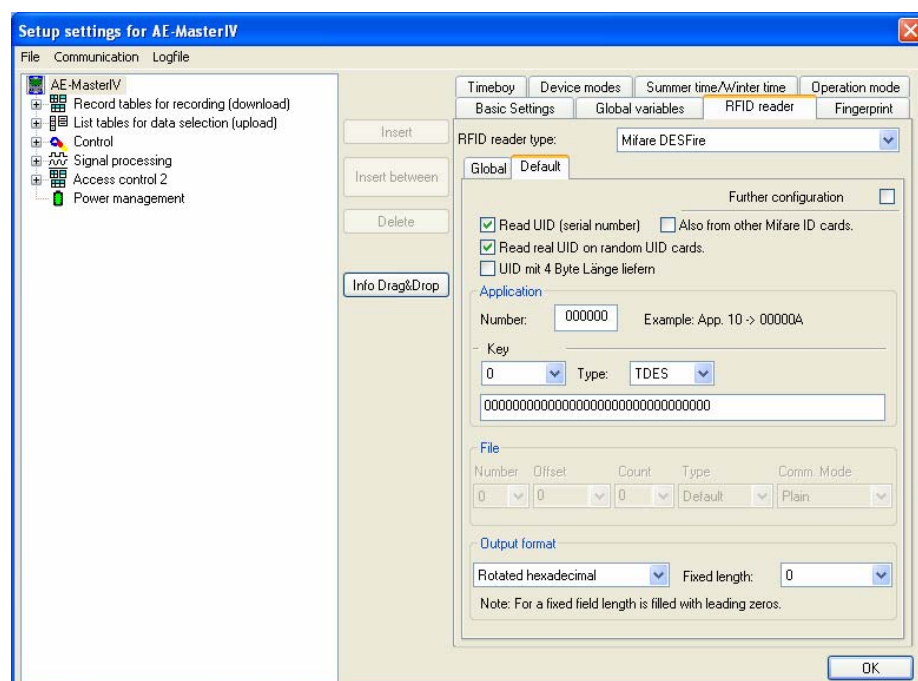
Both may DES with 8 byte keys and TDES with two 8-byte keys, ie 16 bytes.

For systems that use only one 8-byte key, the key must be entered twice.

The DESFire recognize about whether DES or TDES encryption should be used.

communication modes	Description
Plain	Unencrypted, the data between the reader and the RFID chip will be transmitted unencrypted (useful for development)
MACed	Unencrypted with encrypted checksum (MAC), the data between the reader and the RFID chip will be transmitted unencrypted, but it will calculate a checksum of the data and its transmission is encrypted (old transponder method for data validation)
Enchiphered (Crypt)	Encrypted, the data between the reader and the RFID chip is encrypted (data only)

Reading of the serial number



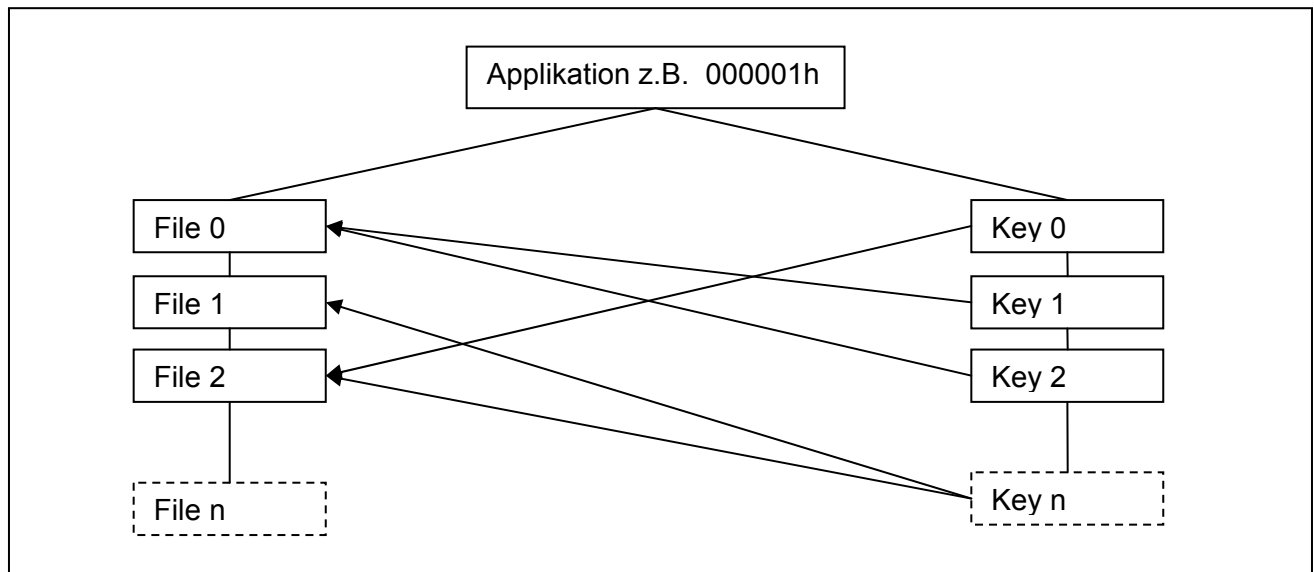
To read the serial number the check mark must be set for it. If in addition to DESFire cards other Mifare Cards should be read, an additional check mark has to be set. If DESFire cards with random UID are used, an authentication into an application shall be used for the determination of the real serial number. Therefore, in this case, also the AID (Application Identifier) in the range of 000000 to FFFFFFFF has to be specified, the number from 0 to 13 and the type of the key and the string of hexadecimal key.

Reading and Writing of files

Typ	description
Standard Data File	Datafile
Backup Data File	Backup Data file
Value File	Number file (is currently not supported)
Linear Record File	Database table (is currently not supported)
Cyclic Record File	Database table, limited, old data will be overwritten (is currently not supported)

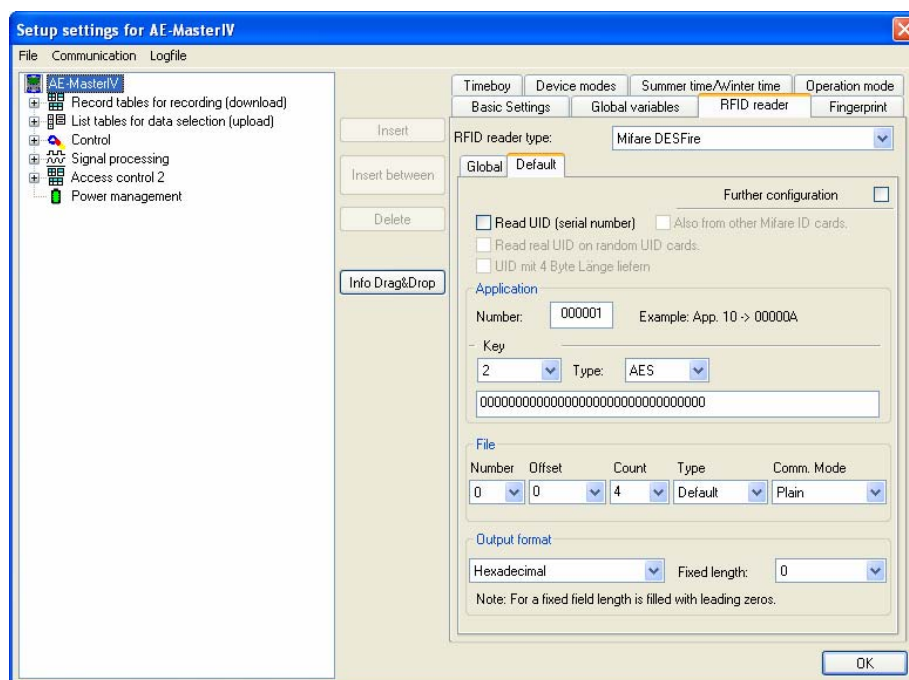
Table File Types

Access to the data is carried out via the selection of an application and subsequent Authentication with a key. The key must be assigned to the file you want to access.



Caution:

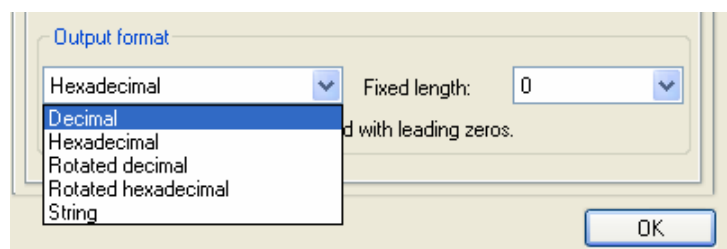
Files of an application can use different keys. It is important to select the correct key for read or write access.



The following settings must be made for access to a file:

- Number of application
- number of the key in the Application
- type of key (TDES, AES ...), DF4 cards can only DF4-encryption
- string of the key
- number of the file
- Offset, Starting position of the data within the file
- Number of bytes to read
- Type of File (Standard, Backup)
- Communication mode (Plain, MACed, Crypt)

Output Format



There are several output formats, how the data is processed in the data set and in the display. Important here is the approach which byte is the most significant. There are also rotated variants in decimal and hexadecimal so that the final outcome shows the correct values. The fixed length is used to bring any to small numbers to a defined length by leading zeros.



Caution:

Decimal numbers are based on the basis of 32-bit (4-byte) operations. It is possible to process up to 64 bits (8 bytes), larger data is leading to conversion errors.

6.3.4.1.6. ISO 14443B – Calypso, CEPAS und Moneo

With these transponders only the serial number can be determined. An access to data areas is not possible. These transponders can be selected under ISO tags (13.56 MHz).

6.3.4.1.7. ISO 14443-2 iClass, Picopass

With these transponders only the serial number can be determined. An access to data areas is not possible. These transponders can be selected under ISO tags (13.56 MHz).

6.3.4.1.8. ISO 14443-3 SRX von ST Microelectronics

With these transponders only the serial number can be determined. An access to data areas is possible on request. These transponders can be selected under ISO tags (13.56 MHz).

6.3.4.1.9. ISO 15693 – iCode, Tag-it, MyD

iCode

I code is available in different memory sizes up to 64 blocks a 4 bytes. The data areas can be read and written, except for the 8-byte serial number. Data areas can be provided with a write protection.

Tag-it

With these transponders you can only read the serial number. An access to data areas is not possible. These transponders can be selected under ISO tags (13.56 MHz).

My-D Vicinity by Infineon

My-D Vicinity is available in different memory sizes up to 1024 bytes. These transponders have depending on the size up to 128 pages of 8 byte or 256 pages of 4 bytes. Currently, we only communicate unencrypted. When the cards are assigned with a key, only the serial number can be determined.

24LR16 / 24LR64 von ST

With these transponders only the serial number can be determined. An access to data areas is not possible. These transponders can be selected under ISO tags (13.56 MHz).

MB89R118 / MB89R119 by Fujitsu

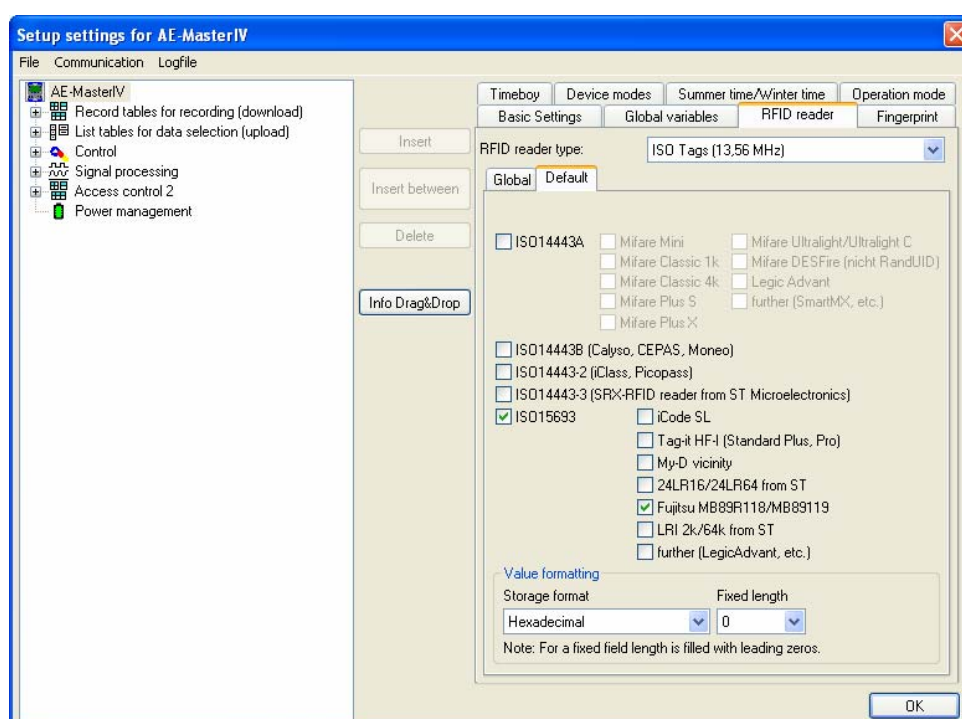
With these transponders only the serial number can be determined. An access to data areas is not possible. These transponders can be selected under ISO tags (13.56 MHz).

LRI 2k

With these transponders only the serial number can be determined. An access to data areas is not possible. These transponders can be selected under ISO tags (13.56 MHz).

6.3.4.1.10. ISO 14443/15693 read serial number

With the TWN3 Multi ISO reader, it is possible to read all serial number of the 13.56MHz ISO14443 and ISO15693 transponders. So you can choose from various methods which transponder types you actually want and what not. You can select the type of transponder ISO tags (13.56 MHz), the respective desired.



Note:

It is often better to work with the serial numbers with the hexadecimal format, because the serial number larger then 7 bytes and contains the manufacturer information's.

7. Tips and Tricks

7.1. Adding leading zeros at ID

With the help of conversion possibilities from hexadecimal to decimal values and vice versa, the total length of a string can be corrected by providing a value mask.

Conversion to hexadecimal value:

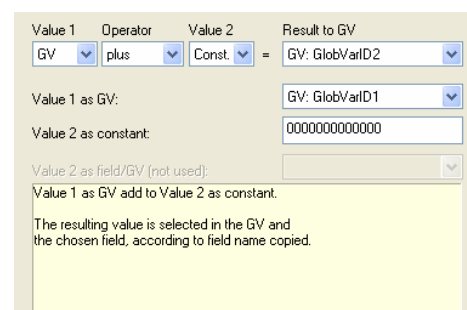
The operation "to Hex" is available for this action.

The value mask serves as indicator for the number of characters. More characters than in the initial value can be used in the value mask. By this, leading zeros can be added, for example.

Examples decimal to hexadecimal with 13 character value mask:

Same applies for conversion of decimal values to hex.

Value 1		Value 2	result in GV
000005202	to Hex	0000000000000	0000000000804
164166271	to Hex	0000000000000	0000009C8FA7F
000002052	to Hex	FFFFFFFFF00000	FFFFFFFFF00804
164166271	to Hex	FFFFFFFFF00000	FFFFF09C8FA7F
GV GlobCardNo	to Hex	Value 2 as value mask	GV GlobPersonnelNo



Caution:



Due to system design, for the data in a GV it is not indexed whether it contains hexadecimal or decimal values. Therefore, you must make sure that a hexadecimal value is not converted again to a hexadecimal value and a decimal value to a decimal value.

8. Index

A

access with PHG modules 21
access with TS modules 21
Advant 128
AES-Key 17
Analogmesswert 97

B

Biokey 66
Bios 47, 48

C

clock 14, 89, 100
Color of Display 24
Communication 10
Communication Key 17

D

Data on Card 103, 111
Datum 89, 100
digital output 76

F

Fingerprint 66
Firmware 5
Functions in DatafoxStudioIV 8

G

global variable 63, 65
GPS 83, 100

H

HID 132
HTTP answer and options 35

I

ISO 14443 134, 135
ISO 15693 134, 140

L

language 22
Legic 128
load list 11
Logische Operationen 94

M

Math-Operationen 94
Mifare 134, 135

O

Online State 104

P

Passwort 17, 29

R

Relais 103
Repair 46
RFID reader 65

S

selection in lists 90
Setup Structure 61
SMS 104
Software 1, 8
Sprünge im Setup 116
State 107
 access 107
Systemlog 54

T

The RFID Technology (Transponder) 119
Tips and Tricks 142

U

Update 20
URL-Codierung 35
USB 25, 26
User interface 6
User Interface 6