

imc Software

Import-/Export-Filter

Doc. Rev.: 2.21 - 2023-01-23



Disclaimer of liability

The contents of this documentation have been carefully checked for consistency with the hardware and software systems described. Nevertheless, it is impossible to completely rule out inconsistencies, so that we decline to offer any guarantee of total conformity.

We reserve the right to make technical modifications of the systems.

Copyright

© 2023 imc Test & Measurement GmbH, Germany

This documentation is the intellectual property of imc Test & Measurement GmbH. imc Test & Measurement GmbH reserves all rights to this documentation. The applicable provisions are stipulated in the "imc Software License Agreement".

The software described in this document may only be used in accordance with the provisions of the "imc Software License Agreement".

Open Source Software Licenses

Some components of imc products use software which is licensed under the GNU General Public License (GPL). Details are available in the About dialog.

If you wish to receive a copy of the GPL sources used, please contact our Hotline.

Notes regarding this document

The manual for imc Software is available both as an e-book, chm and/or in PDF format.

What is the best way to read the imc Software documents?

Setting up

Before installing the software, start by reading the setting up description. This contains important tips for achieving problem-free installation.

Manual - complete documentation

The manual serves as reference material and describes how to operate the software.

Special notes



Warning

Warnings contain information that must be observed to protect the user from harm or to prevent damage to property.



Note

Notes denote useful additional information on a particular topic.



Reference

A reference in this document is a reference in the text to another text passage.

Table of contents

1 Import / Export Filter	6
1.1 ADF	8
1.2 Agilent	9
1.3 Alborn	10
1.4 Anovis	10
1.5 ASCII Import (single-column)	11
1.6 ATFX	14
1.7 B&S Multidata	15
1.8 Binary format	16
1.9 BLF	22
1.10 Byte Blob	27
1.11 Caesar Datasystems	28
1.12 DASyLab	29
1.13 Dewetron	30
1.14 Dewetron DMD	33
1.15 DIAdem	35
1.16 Digitek DataView	40
1.17 DSP Development Corporation DADiSP	40
1.18 DTS	41
1.19 ETAS/Bosch	41
1.20 Excel (XLS)-Format	42
1.21 FlexPro	46
1.22 G.i.N.	47
1.23 Google Earth	48
1.24 Gould	49
1.25 Grabau Sound Power	49
1.26 Graphtec	50
1.27 Haag	50
1.28 HBM	51
1.29 HBM PNRF	52
1.30 HDF5	53
1.31 HEAD acoustics	60
1.32 HERF4	64
1.33 HIOKI measurement devices	65
1.34 Hi-Techniques	66
1.35 Horiba	66
1.36 Hydrotechnik	67
1.37 iba	67
1.38 IEEE Standard Common Format	68
1.39 imc Bus Filter	68
1.40 imc STUDIO Metadata	72

1.41 Instron	72
1.42 IRIG106	74
1.43 K&S Elektronik	75
1.44 Kipp & Zonen	75
1.45 Knorr SD	76
1.46 LabVIEW	80
1.47 LeCroy Oscilloscopes	80
1.48 LEM	81
1.49 LMS	81
1.50 Matlab	82
1.51 MDF	85
1.52 MDF4	91
1.53 MKT CAN Logger	97
1.54 MTS System Corporation	97
1.55 Nicolet	98
1.56 nSoft-DAC	101
1.57 Picture Format	101
1.58 Ono Sokki	102
1.59 OROS	102
1.60 Remes DIGISKOP	103
1.61 RPC-3	104
1.62 SigView	105
1.63 Sirec	106
1.64 SoMat	106
1.65 Sony	108
1.66 Spider Measurement System	109
1.67 Stabicontrol	109
1.68 Staehle	110
1.69 Stiegele Datasystems	110
1.70 TEAC	111
1.71 Tektronix	112
1.72 Telsonic	113
1.73 TurboLab	113
1.74 Universal File Format UFF58	113
1.75 Vestas INT	117
1.76 Vispiron	117
1.77 WAV	117
1.78 Yokogawa	118
Index	120

1 Import / Export Filter

Import / Export filter serve the purpose of data exchange between FAMOS and other programs having their own data formats.

FAMOS supports two types of import/export filters.

- Filters with Dynamic Link Libraries (*.dll)
- Filters with the FAMOS File Assistant (*.FAS)

Filters based on Dynamic Link Libraries (*.dll) do not have a fixed range of functions. The range of functions generally comprises import and export of various data types and data structures for an external data format. Filters based on the File Assistant (*.FAS) are only for import and generally designed for a particular data type within a certain data format. The range of functions is more narrow than that of the library filters, but is not fixed and can be modified by means of the File Assistant.

Filter options

External data formats feature a wide range of complexity. For this reason, it is possible to customize the data import process for filters based on Dynamic Link Libraries (*.dll).

This customization/adaption process is accomplished by means of selecting options which apply either globally or according to individual specifications.

Global options:

A filter's options are configured by means of the filter's Options dialog. These options take effect upon manual data import (Load File) and when importing to a sequence. The settings are saved on the local PC in an init-file for FAMOS data filters.

Individual options in a FAMOS sequence:

When a FAMOS sequence is run, the options are passed to the function "FileOpenFAS(...)" as parameters. The options applied are only in effect for this single data import operation. For particular options which are not passed as parameters, the global options settings apply. Be aware of this behavior when using FAMOS sequences on different PCs.

Using the filters

With a few exceptions, the import filters can be used immediately once FAMOS is installed. To use the following filters, additional libraries from the respective manufacturers are needed:


- National Instruments DIAdem
- Somat
- FlexPro
- Dewetron
- HBM PNRF
- Hydrotechnik

For licensure reasons, these libraries do not come with imc FAMOS and must be acquired separately.

Detailed information is provided in the documentation of the respective filter.

The **common data storage location** to which new imc filter-DLLs, as well as any support libraries from 3rd-party manufacturers, must be copied depends on whether the 32- or 64-bit version of the imc software is used:

C:\Program Files\imc\shared\Extensions

An example is presented [here](#) .

1.1 ADF

Filter

ADF

Type

Import Filter

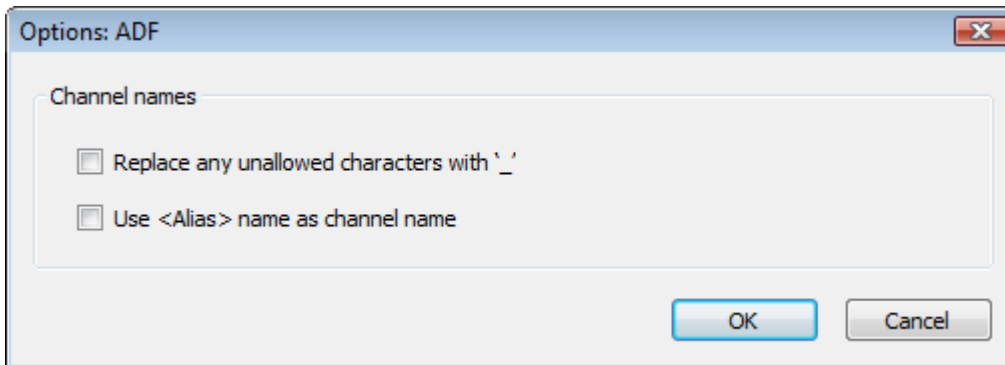
Description

Filter name	Filename of filter	Version	Import File Extension	Device Types
ADF	ADF.dll, ADF.409	10.0.0.0	*.adf	Kistler DAS-3

Import:

- Normal Waveforms if value type "ANALOG,BINARY,NATURAL"
- XY-Digital Waveforms if value type "BINARYTIMED"
- Support of import in multithread mode

Options



Replace any unallowed characters with '_\'

If the channel name contains any characters which are not permitted, such as "+, -, ...", these characters are replaced by '_\'.

Use <Alias> name as channel name

The format includes tags for the channel name and an Alias channel name. If the <Alias> name has been specified, it is used as the channel name.

Optionen and Sequences

There are two ways to specify options:

- Options for the format's filter or options for the format's derivative data filters

- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

Example:

```
idFile = FileOpenFAS(FileName , "#ADFdll|ADF Format|UseAliasName=1",0)
```

Option	Definition
SubstInvalidChar	0 - Name is adopted without any checking 1 - Replace any unallowed characters with ' _ '
UseAliasName	0 - Use <Name> tag as channel name 1 - Use <Alias> tag as channel name, if it has been specified

1.2 Agilent

Filter

Agilent

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Agilent BIN-Format	AgilentBinaryImport.dll	10.0.0.0	*.bin

Import:

Agilent Channle Types	FAMOS data type
LOGIC	digital Channel
NORMAL- und AVERAGE	FAMOS Normal Waveforms

- Support of import in multithread mode

1.3 Alborn

Filter

Alborn

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension	Device Types
Alborn AlMemo	Alborn_Alemo.fas	*.* (*.txt,...)	Alborn-Meßsystem
Alborn AlMemo2	Alborn_Alemo2.fas	*.* (*.txt,...)	Alborn-Meßsystem
Alborn AMR	Alborn_AMR.fas	*.* (*.txt,...)	Alborn AMR-Meßsystem

Import:

- Data into complex FAMOS data sets (R,I)

1.4 Anovis

Filter

Anovis

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Anovis Measurement Data	AMEImport.dll	10.0.0.0	*.ame
Anovis Signal File	ASDImport.dll	10.0.0.0	*.asd

Import:

- equidistant data into FAMOS Normal Waveforms
- Parameter "all" reads channel data without dialog:

```
ID = FileOpenFas("<Filename>", "#AMEImport.dll|Anovis Measurement Data|all", 0)
```

AME V1.1.0.0:

- Import of channel properties

- Import of file informations

AME V10.0.0.0:

- Support of import in multithread mode
-

1.5 ASCII Import (single-column)

Filter

ASCII

Type

Import Filter

This import filter is still retained for compatibility reasons. However, the more powerful option of using the ASCII Import Assistant is preferable

This option can be selected for loading files in simple ASCII-formats. Select "ASCII" in the dialog "Load file..".

Note: *If the ASCII-format to be loaded is used quite frequently, it may be worthwhile to make a special **import filter** using the imc File Assistant or ASCII import assistant. The effort needed to create the filter can pay off due to the significant extra speed and comfort of using it, especially where files contain multiple channels and also additional characteristics (such as channel name, sampling rate, etc.) also are to be read out. Also, the imc Homepage offers a number of already created import filters for common standard formats.*

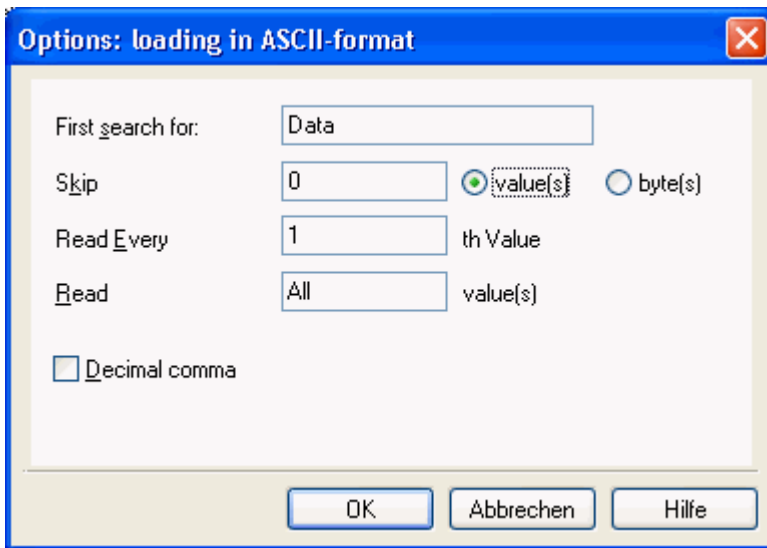
Many programs save data in the form of ASCII files, including spreadsheets, text processing programs, as well as mathematical, data acquisition and data processing programs.

To read the data from these files in FAMOS, a flexible ASCII format has been implemented. There is no standard for ASCII, each program uses its own conventions. What all ASCII formats have in common is that data, i.e. the numbers, are listed consecutively and are separated by a string. Texts may be included in the ASCII files at any position. The user may specify in a dialog box how the special ASCII files are to be read in order to read the correct numbers.

ASCII files are read according to the following procedure:

- First, the program searches for a specified character string.
- A certain number of values or bytes are skipped. The user may specify either values or bytes.
- Then the numerical values are read. The user can specify which n^{th} value is to be read (e.g. each third); all others are ignored (skipped over).
- Either a certain number of values (e.g. 100 values) or all values are read. The number of values read determines the length of the waveform generated in FAMOS.
- Numerical values may be written with a decimal point (e.g. 1.35 V) or decimal comma (e.g. DM 3,75).
- All characters between numerical values are ignored.

The button "Options" in dialog "Load file.." calls the dialog for defining your special ASCII format. This dialog can also be accessed via "Extra"/ "Options"/ "File-Load/Import"/ "Import Filter"/ "ASCII (single-column)" -> "Options".



Dialog: Options loading in ASCII-format

First search for

First enter the string in the text box which is to be searched for in the ASCII file. If no such string is present, leave this box empty.

Skip

Enter the number of values or bytes which should be skipped. This must be an integer greater than or equal to zero.

Byte / Values

Select "Byte" to skip a specified number of bytes; select "Values" to skip a specified number of complete values.

Read every n^{th} value

Enter in the box which n^{th} numerical value should be read. This must be an integer greater than zero.

Read...values

Enter in the text box how many values should be read in. This may be a fixed integer greater than zero if the number is to be defined. You may enter "all" or a zero if all numbers are to be read to the end of the file.

Decimal comma

Select this option when the values have been written with decimal comma instead of a decimal point.

Target format

Here, you enter the data format for the data set to be generated. "4-byte Real" is accurate to about 7 decimal positions and is adequate for most practical purposes.

If higher precision is needed, select "8-byte Real".

Example 1

All values (without date!) are to be read from the following file:

```
imc Test & Measurement GmbH
                                2023
Dear Madame, Dear Sir,
Please read the following values:
1, 2, 3.4, 5e-6.
Do not forget these values:
7
88
99
Best regards
imc
...
```

Select the following setups:

```
Read from string: values:
Skip              0 values
Read each         1st value
Read              all values
Decimal comma     no
```

The following values are then written in the waveform generated during loading:

```
1.0 2.0 3.4 0.005 7.0 88.0 99.0
```

Example 2

The first 5 values in the second column are to be read from the following file:

```
.....
...All costs in DM
...1.00 10.00 100.00
...2.00 20.00 200.00
...3.00 30.00 300.00
...4.00 40.00 400.00
...5.00 50.00 500.00
...6.00 60.00 600.00
...7.00 70.00 700.00
...8.00 80.00 800.00
...9.00 90.00 900.00
.....
```

Select the following setups:

```
Read from string: costs in DM
Skip              1. value
Read each         3rd value
Read              5 values
Decimal comma     yes
```

The following values are written in the waveform:

```
10.0 20.0 30.0 40.0 50.0
```

 **Note**

Data read are always created in FAMOS in real 4 byte format (floating point).

During reading of ASCII files, units, x-axis scaling, etc. are not read from the files, they are always set to standard values. This can be changed after data are read. It is often convenient to create a small sequence to automate the procedure.

Please note that Windows uses the ANSI character set, which does not correspond to the ASCII set for special characters. Files containing special characters may be read incorrectly.

For complex or frequently used ASCII-formats, we recommend creating an import filter with the help of the imc File Assistant.

Reference

FAMOS Function Reference, Functions: [FileOpenASCII](#), [FileLineRead](#), [ASCII](#), [LOAD](#)

1.6 ATFX

Filter

ATFX (ASAM ODS Transport File Format Xml)

Type

Export/Import Filter

Important

The Bus Filter needs the FAMOS 6.1 installation.

Description

Filter name	Filename of filter	Version	Import File Extension
ASAM ATFX Format	atfx.dll	10.0.0.0	*.atfx

Import:

- A FAMOS group is created for a measurement (instance of the application element "AoMeasurement").
- The FAMOS channel names are the names of the instances of the application elements "AoLocalColumn" of the measurement's submatrices (instances of the application element "AoSubmatrix").
 - units if available
 - 3D data and referenced data sets are imported as individual FAMOS data sets
 - additional information provided as text variables in the FAMOS group "Import information"
- Description of the AoMeasurementQuantity loaded as a comment
- The measurement's version (if applicable) is added to the measurement's name (AoMeasurement).
- Support of import in multithread mode

Export:

- For export, an application model is generated. This model defines an "AoTest" and an "AoSubtest".
- Export of FAMOS data sets proceeds as follows:
 - FAMOS groups are exported as measurements (instances of "AoMeasurement") along with the group's name
 - If single signals not belonging to any group exist, a measurement named "FAMOS-Measurement" is generated
 - FAMOS channels are exported as instances of the application element "AoLocalColumn"
 - If a unit, offset or scaling for X-data are specified, then an independent X-channel is generated. An "_X" is appended to the channel name.
 - For the FAMOS data sets "XY" and "Complex", 2 instances of "AoLocalColumn" are generated:
 - 1st Instance: X-component data with "_X"
 - 2nd Instance: Y-component data with "_Y"
- Submatrices ("AoSubmatrix") are generated automatically and receive the name "SM-...".
- MeasurementQuantities ("AoMeasurementQuantity") are formed with the FAMOS channel names.
- If specified, units are exported as instances of "AoUnit".

- Quantities ("AoQuantity") are formed from known units. If the Quantity can not be determined from the unit text, the text determines the Quantity.
- Physical dimensions are derived from the unit. If the unit is not known, no physical dimension is exported.
- The channel's data are located in the file "<exp-Filename>.bin". Only one file is supported.
- Saving of the comment as Description of the AoMeasurementQuantity

- HVH-application-model

The NVH application model is supported as follows:

- A FAMOS data set is assigned the MimeType...

"application/x-asam.aomeasurementquantity.tachovector",

...if the data set is a Normal Waveform having the scaling "1", an offset of "0" and no X-unit specified.

"application/x-asam.aomeasurementquantity.streamed",

...if the condition for "tachovector" is not met or if the data set has multiple components.

"application/x-asam.aomeasurementquantity.spectrum",

...if the data set is a complex "(RI)" data set and the X-unit was specified to take "Hz".

The NVH - parameter set with the associated "required" parameters is completed with "default" values.

- X-axes of 2D spectra are added to the FAMOS channel

- X- and Z-axes of 3D spectra are added to the FAMOS channel. The results are segmented data sets.

- Support of import in multithread mode

1.7 B&S Multidata

Filter

B&S Multidata

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
BS Multidata ME3 V2.5	BSMultidata_ME3V25.FAS	*.dat
BS Multidata ME3 V3.0	BSMultidata_ME3V30.FAS	*.dat
B+S Multidata Format	BS_Multidata.dll, V10.0.0.0	*.dat

B+S Multidata Format

Import:

- single-event data sets; always having exactly 1 event
 - channels having identical names, but originating from different header blocks, are assigned indices
 - a sampling rate is calculated based on the parameters "#Timebase" and "#SamplingRateMultiplier"
 - the Y-unit and Y-scaling factor are determined by the parameters #CH_USER "Unit" &UserMul &UserOffset in the channel description
 - Support of import in multithread mode
-

1.8 Binary format

Filter

Binär

Typ

Import Filter

Choose this option to load files in any binary format. Select "Binary" from the format list in the dialog "Load file..".

Note

*If the BINARY-format to be loaded is used quite frequently, it may be worthwhile to make a special **import filter** using the imc File Assistant. The effort needed to create the filter can pay off due to the significant extra speed and comfort of using it, especially where files contain multiple channels and also additional characteristics (such as channel name, sampling rate, etc.) also are to be read out. Also, the imc Homepage offers a number of already created import filters for common standard formats.*

Many programs save data in binary form, for example data acquisition and data processing programs. A flexible binary format is implemented to read data from such files in FAMOS. Binary format does not have a standard, each program uses its own conventions. Conventional binary formats group data in a compact block, usually preceded by a header.

You may now specify in a dialog box how the special binary format should be read in order to correctly read out the values.

- Binary files are read according to the following procedure:
- The program searches for a specified string.
- A specified number of bytes are skipped.
- Either a specified number of values are read or all values are read. This determines the length of the waveform generated in FAMOS.
- A space of defined width in bytes is skipped between the values to be read. This allows information between the values to be skipped. It is also possible to read only ever n^{th} value. The space is always specified in bytes.

The data type is highly significant in reading data. Floating pointing-point numbers in 4, 6 or 8 byte representation as well as integers in 1, 2 or 4 byte representation can be read.

Floating pointing-point numbers in 4 or 8 byte representation follow the IEEE standard format.

In addition, floating pointing-point numbers can be read in a special 6-byte format, as used by Turbo-Pascal (? Borland International, Inc) for the data type REAL.

In floating pointing-point numbers, it is not necessary to enter any information about value range, since the entire range of values is already included.

Integers in 1-, 2- or 4-byte representation as complement on two, that means with sign, are interpreted, for example for two bytes

```
0000H  0d
ffffH  -1d
0001H  1d
```

(H = hex, d = decimal).

Integers in 1, 2 or 4-byte representation without sign are interpreted, for example for two bytes

```
0000H      0d
0001H      1d
ffffH     65536d
```

To transform the range of values to the actual range of (physical) quantity, the minimum and maximum of the physical value range can be specified. The number of bits which determine the measurement range can also be defined, i.e. the resolution of the device generating the data.

Integers are transformed according to the following rules:

Signed (two's complement):

$$\text{Transform} = (\text{Range_max} - \text{Range_min}) * (\text{NumberWS} + 2\text{Bits} - 1) / (2\text{Bits} - 1) + \text{Range_min}$$

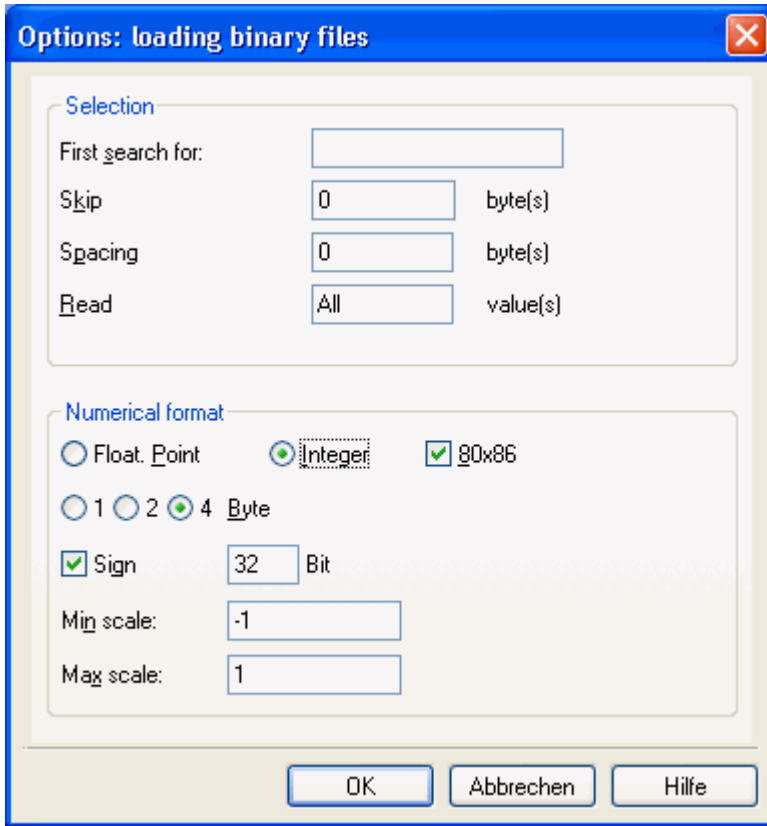
Unsigned:

$$\text{Transform} = (\text{Range_max} - \text{Range_min}) * \text{NumberUS} / (2\text{Bits} - 1) + \text{Range_min}$$

Abbreviations:

NumberWS	Integer with sign (1, 2 or 4 bytes)
NumberUS	Integer without sign (1, 2 or 4 bytes)
Bits	Number of bits which compose the measurement range (specify in dialog)
Range_min	Minimum of the physical value range (specify in dialog)
Range_max	Maximum of the physical value range (specify in dialog)
Transform	Transformed number written in the FAMOS waveform.

To open the binary format options dialog, click on <Options> in the "Load File dialog. You can also find this dialog under "Extra/ Options", on the index card "Load file | BINARY". This dialog is accessed via "Extra"/ "Options"/ "File-Load/Import"/ "Import Filter"/ "BINARY" -> "Options".



Dialog: Options loading binary files

First search for:

Enter which string the program should search for first. Leave the box empty if the program should not search for a string.

Spacing...

Enter the number of bytes which should be skipped between the values.

Read...values

Enter the number of values to be read. If all values to the end of the file should be read, enter a zero or "all".

Floating point

Select this option to read floating point numbers (real numbers) from the file.

Integer

Select this option to read integers.

Number of bytes per value

Select the corresponding option for the number of bytes which should be represented with the values. In floating pointing-point numbers, choose between 4, 6, and 8 bytes; in integers, select between 1, 2, and 4 bytes.

Sign

Select this option to read integers present in two's complement representation.

Bit

Enter in the text box how many bits make up the measurement range when integers are read. The number must be greater than zero and smaller than 8 times the number of bytes.

Min. Scale / Max Scale

Enter the minimum and maximum of the value range to specify the range of integer values in FAMOS.

80x86

Select this option to read files in Intel format. In this format, the lowest-value byte is located at the first position, as opposed to other formats (e.g. Motorola), in which the highest-value byte is at the first position. Do not select the option "80x86" for these formats.

Select the "OK" button to implement the format settings or "Cancel" to discard any changes.

The dialog box is not closed if incorrect values, for example, an invalid number of bytes, are entered in the text boxes when "OK" is selected.

Example 1

A file to be read in binary format according to the Intel convention has the following content: first come 18 bytes with text "AD-converter channel 1", next, 2 bytes with unknown content, and then 1000 bytes with 500 integers per 2 bytes, originating directly from an AD-converter, i.e. without sign (not in two's complement, measurement range -10V...+10V.).

The following setups should then be selected in the dialog box to read the file:

Read from string:		
Skip	20	Byte
Spacing	0	Byte
Read	Alle	Werte
Value type	Ganz	
Bytes	2	nein
Sign	12	
Bit	min	
Range min	max	-10
Range max	ja	10
80 x 86		

Example 2

A file to be read contains 10000 bytes with integer samples from four channels in Intel format. The values have already been converted into an integer 16-bit format in two's complement by data storage software. The third channel is of interest: an 8-bit channel with a range from 0...5V. The data are arranged as follows: AD-Converter, no sign (no complement on two), measurement range -10V...+10V. To read the file use the following settings:

2 Byte	1	Value	Channel 1
2 Byte	1	Value	Channel 2
2 Byte	1	Value	Channel 3
2 Byte	1	Value	Channel 4
2 Byte	2	Value	Channel 1
2 Byte	2	Value	Channel 2

Select the following setups to read the third channel from the file:

Read from string:		
Skip	4	Byte
Space	6	Byte
Read	all	Value
Value type	integer	
Bytes	2	
Sign	yes	
Bits	8	
Range min	min	0
Range max	max	5
80 x 86	yes	

Example 3

At the beginning of a file, the date and other information is provided, followed by a unique string to initialize data and then the data as real numbers with simple accuracy.

- 7 bytes with date, e.g. "15.6.89", may vary
- 1 byte with ID for floating pointing-point values
- 12 bytes with sampling time in ASCII ("dt = 3.5 ms", may vary)
- 15 bytes with remark "Data on motor "
- 6 bytes with unique string: "Data##:"
- 4000 bytes with 4-byte floating pointing-point values in Intel format

Select the following setups to read the file:

Read from string:	Data##:	
Skip	0	Byte
Space	0	Byte
Read	all	Value
Value type	floating-point	
Bytes	4	
80 x 86	yes	

**Note**

Reference

FAMOS Function Reference: [BINARY, LOAD](#)

Manual for imc [File Assistant](#).

1.9 BLF

Filter

BLF-Format (Binary Log Format)

Type

Import

Description

Filter name	Filename of filter	Version	Import File Extension
BLF-Format	BLF.dll, BLF.409, binlog.dll (Vector Informatik GmbH)	10.0.0.0	*.blf

Import:

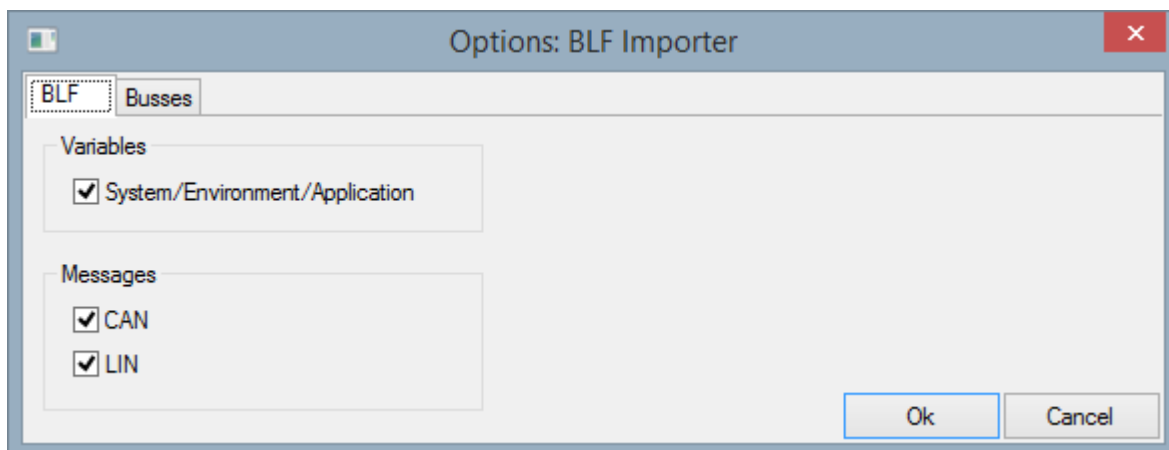
- XY data sets
- Normal Waveforms
- Support of import in multithread mode

General functioning

The Filter serves the purpose of extracting values from *.blf logging files containing binary data sets (frames) along with associated time stamps. On the basis of message- and channel descriptions, data are extracted from the logged data sets and assigned to FAMOS channels.

Options

BLF



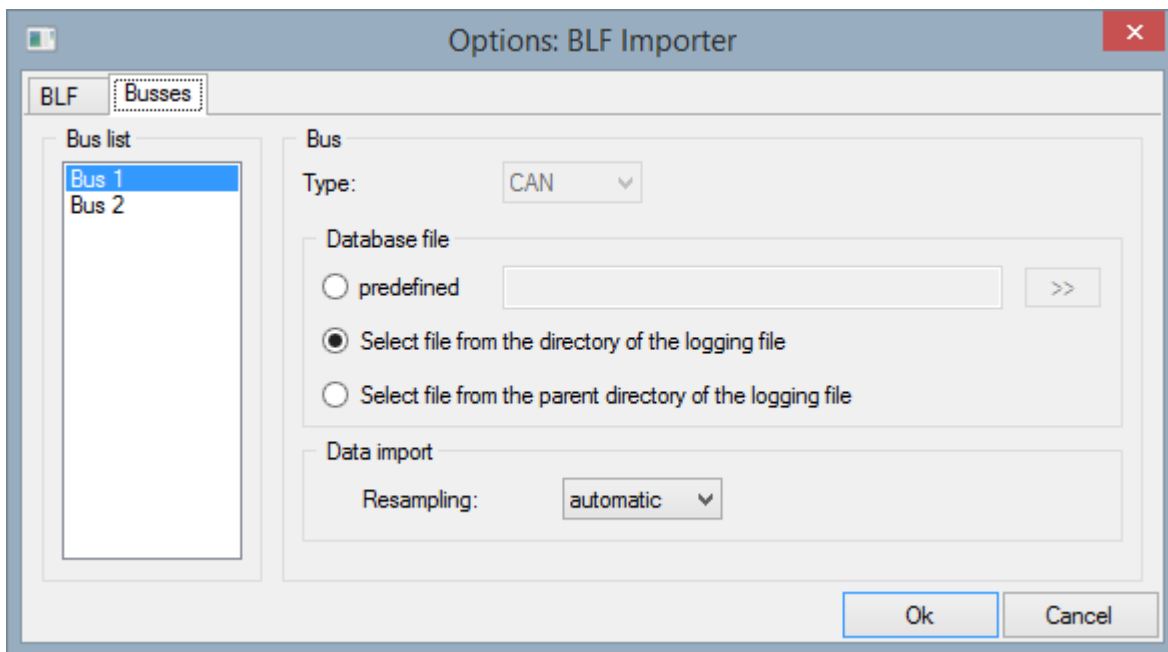
Variables:

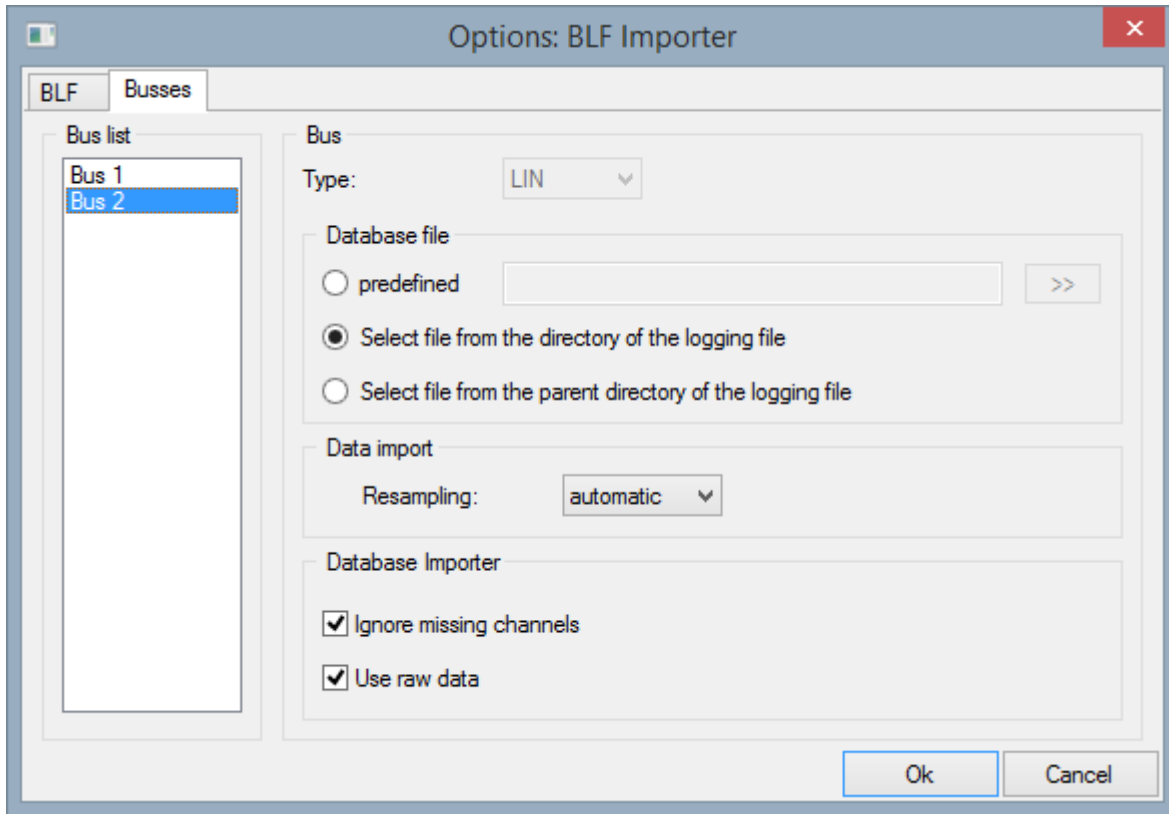
VAR: The system, environment and application variables are imported.

Messages:

CAN: The CAN messages are imported.

LIN: The LIN Messages are imported.

CAN-Bus**LIN-Bus**



Bus

Type: The type of the bus, only "CAN" or "LIN".

Database file

predefined

The basis file to use is specified directly. Specifying the relative pathname ("..\testfile.dbc") is permitted. The relative pathname refers to the data file currently open. The specified basis file is used for all logfiles.

A "*" instead of the filename ("..*.dbc") enters the name of the logfile at the position of the "*".

Select file from the directory of the logging file

An "Open File" dialog is displayed. The default folder is the folder of the logfile which is open.

Select file from parent directory of the logging file

An "Open File" dialog is displayed. The default folder is the folder one level above the folder of the logfile which is open.

Data Import

Resampling: "automatic"

If the control data contain specifications of the channel's or the frame's interval, then the channel is resampled according to this specified interval.

In FAMOS, a simple data set is generated with specifications of the interval and offset.

If there is no specified interval, an X/Y data set is created with an X-component as the time specification.

Resampling: "Time-stamp"

The data in the channels are supplied with time stamps.

In FAMOS an X/Y data set is created with an X-component as the time specification.

Database importer**Ignore missing channels**

Use raw data: If offset and factor are missing the the raw data are imported.

Options and Sequences

There are two ways to specify options:

- Options for the format's filter or options for the format's derivative data filters
- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

List of options:

Option	Definition
VAR=<0 1>	Enables/disables the import of system, environment and application variables: 0 : no import of variables 1 : import of variables
CAN=<0 1>	Enables/disables the import of CAN messages: 0 : no CAN messages 1 : decoding CAN messages
CAN_DBLOAD=<0 1 2>	Defines the load procedure of the CAN data base file: 0: predefined, only to use in combination with option "CAN<id>_DB" 1: Select file from the directory of the logging file 2: Select file from parent directory of the logging file
CAN_DB=<filename>	Defines the CAN data base file name
CAN_TS=<0 1>	Option: Resampling 0: automatic 1: Time-stamp
LIN=<0 1>	Enables/disables import of LIN messages: 0 : no LIN messages 1 : decoding LIN messages
LIN_DBLOAD=<0 1 2>	Defines the load procedure of the LIN data base file: 0 : Predefined by configuration or parameter „LIN_DB<id>_DB“ 1 : Select file from the directory of the logging file 2 : Select file from parent directory of the logging file
LIN_DB=<filename>	Defines the LIN data base file name

LIN_TS	Defines the resampling of the message channels: 0 : „automatic“ 1 : „time-stamp“
Channels=<name1,name2>	Channel filter list with channel names separated by commas.
IGNOREMISSCHANNELS=<0 1>	0 : Error message, if any channel is missing 1 : ignore the channel
USERAWDATA=<0 1>	0 : Error message, if a channel description (offset, factor) is missing 1 : import the raw data of the channel

The table below shows the data types of the logging format that are supported.

BLF Data Types	Name of FAMOS Variable	FAMOS Waveform
BL_OBJ_TYPE_CAN_MESSAGE	defined by database file (*.dbc, *.cba, *.ldf)	XY-Waveform
BL_OBJ_TYPE_CAN_MESSAGE2	defined by database file (*.dbc, *.cba, *.ldf)	XY-Waveform
BL_OBJ_TYPE_LIN_MESSAGE	defined by database file (*.dbc, *.cba, *.ldf)	XY-Waveform
BL_OBJ_TYPE_LIN_MESSAGE2	defined by database file (*.dbc, *.cba, *.ldf)	XY-Waveform
BL_OBJ_TYPE_CAN_ERROR	CanErrorFrame_Length	Normal Waveform
BL_OBJ_TYPE_CAN_OVERLOAD	CanOverloadFrame	Normal Waveform
BL_OBJ_TYPE_CAN_DRIVER_ERROR	CanDriver_TX_Error CanDriver_ErrorCode CanDriver_ErrorCode	Normal Waveforms
BL_OBJ_TYPE_CAN_STATISTIC	CanStatistics_BusLoad CanStatistics_StandardDataFrames CanStatistics_ExtendedDataFrames CanStatistics_StandardRemoteFrames CanStatistics_ExtendedRemoteFrames CanStatistics_ErrorFrames CanStatistics_OverloadFrames	Normal Waveforms
BL_OBJ_TYPE_ENV_INTEGER	name of the environment variable	Normal Waveform
BL_OBJ_TYPE_ENV_DOUBLE	name of the environment variable	Normal Waveform
BL_OBJ_TYPE_ENV_STRING	name of the environment variable	Array of text
BL_OBJ_TYPE_APP_TEXT	name of the application text variable	Array of text
BL_SYSVAR_TYPE_DOUBLE	name of the system variable	Normal Waveform
BL_SYSVAR_TYPE_LONG	name of the system variable	Normal Waveform
BL_SYSVAR_TYPE_STRING	name of the system variable	Array of text
BL_OBJ_TYPE_LIN_SLEEP	LinSleepModeEvent_Flags	Normal Waveforms

BLF Data Types	Name of FAMOS Variable	FAMOS Waveform
	LinSleepModeEvent_Reason	
BL_OBJ_TYPE_LIN_WAKEUP	LinSleepWakeUpEvent_Signal LinSleepWakeUpEvent_External	Normal Waveforms
BL_OBJ_TYPE_LIN_WAKEUP2	LinSleepWakeUpEvent2_Signal LinSleepWakeUpEvent2_External LinSleepWakeUpEvent2_Length	Normal Waveforms

The channel information for message decoding (CAN, LIN) can be specified from the following database files:

- Vector CAN Database (*.dbc)
- LIN Database (*.ldf)
- imc CAN Assistant (*.cba)
- imc Blob (*.fdb)

Examples:

Example for calling without import of CAN and LIN messages:

```
FileOpenFAS("test.blf", "#BLF|BLF Format|LIN=0|CAN=0|VAR=1",0)
```

Example for calling with import of CAN messages and data base:

```
FileOpenFAS("test.blf", "#BLF|BLF Format|CAN=1|VAR=0|LIN=0|CAN_DBLOAD=0|CAN_DB=D:\test.dbc",0)
```

Example for 4 buses (CAN):

```
success=FileLoad("C:\tmp\Messdatei.blf", "#BLF.dll|BLF Format|CAN=1|VAR=0|LIN=0|CAN1_DBLOAD=0|CAN1_DB=C:\tmp\CAN_53.dbc|CAN2_DBLOAD=0|CAN2_DB=C:\tmp\CAN_53.dbc|CAN3_DBLOAD=0|CAN3_DB=C:\tmp\CAN_53.dbc|CAN4_DBLOAD=0|CAN4_DB=C:\tmp\CAN_53.dbc",0);
```

1.10 Byte Blob

Filter

Byte Blob

Type

Import/Export filter

Description

Filter name	Format ID	Import file extension
-------------	-----------	-----------------------

imc Byte Blob	"imc/ByteBlob"	any
---------------	----------------	-----

Import:

- Any files. Typical applications include images, videos, PDF.

General function notes

A file's complete contents are imported to a data set in **binary and uninterpreted** condition. The format ID is "imc/ByteBlob".

The resulting data format is '1 Byte, unsigned'. This is generally only required for special applications, e.g. in conjunction with the FAMOS-Database Kit, for the purpose of saving any arbitrary kind of file (e.g. PDF, images, videos) along with the actual associated measurement data in databases.

- Existing variables having duplicate names are overwritten without prior warning.

Parameters of the FileLoad and FileOpenFAS functions**Example: Import**

```
ret= FileLoad("C:\test\MyPicture1.png", "imc/ByteBlob", 0)
idFile= FileOpenFAS("D:\temp\Channels_Event4.png", "imc/ByteBlob", 0)
```

Export:

- Any variables. Typical applications include variables which had previously been loaded using the "ByteBlob" filter -so for instance an image file imported to a variable with the ByteBlob-filter and saved to the database. Then later the variable is read out of the database and exported with "ByteBlob", in order to restore the original image file.

Example: Export

```
ret= FileSave("C:\test\MyPicture1_copy.png", "imc/ByteBlob", 0, MyPicture1)
```

1.11 Caesar Datasystems

Filter

Caesar Datasystems

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Caesar Datasystems Remus	Caesar_Remus.FAS	*.rms + *.dat
Caesar Datasystems Remus groups	Caesar_Remus_Groups.FAS	*.rms + *.dat

Import:

- Amplitudes and phases into Normal FAMOS Waveforms
 - Groups
-

1.12 DASyLab

Filter

DASyLab

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
DasyLab file format	DasyLabImport.dll	10.0.0.0	*.ddf
DASyLab ASCII	DasyLab_ASCII.FAS	-	*.asc
DASyLab ASCII V9.0	DasyLab_ASCII_V90.FAS	-	*.asc

Import:

- equidistant data into FAMOS Normal Waveforms

DDF V1.2.0.0:

- Import of Y-Units

DDF V2.0.0.0:

- Import performance improved

DDF V2.1.0.0:

- Universal-Format 3
- multi-event data

DDF V10.0.0.0:

- Support of import in multithread mode
-

1.13 Dewetron

Filter

Dewetron

Typ

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Dewetron	Dewetron.dll	10.0.0.0	*.d7d, *.dxd

Import:

- equidistant and non equidistant data into FAMOS X/Y Waveforms
- Normal / Event-based Waveforms
- Meta data listed in the file header are imported as a text variable in the group "FileInformation".
- support of arrays (spectrum)
- first value of a text array as a text variable
- Support of import in multithread mode

Options

Options: Dewetron

Optimization

- Always XY-waveform
- Generate simple waveform from equidistant data
- Always generate events
- Only generate events if there is more than 1 Event

Channel names

- Replace any unallowed characters with '_'

Reduced data

- Read reduced data

Arrays

- The Z-axis data are taken as equidistant (offset: value 1, delta: value2 - value 1)
- Generate simple waveform from Z-axis data (if not equidistant)

OK Cancel

Always XY-waveform

Always generates an X/Y-waveform

Generate simple waveform from equidistant data

When this option is used, the system investigates the X/Y-waveform to see if the X-values are equidistant. If they are adequately equidistant, a "simple waveform" is generated. Simple waveforms are easier to process in sequences. This option is highly suited to situations where the measurement procedure is equidistant but the MDF format causes the data to take the X/Y-waveform form.

Always generate events

Event-based waveforms are always generated

Only generate events if there is more than 1 Event

Event-based waveforms are generated if more than 1 event exists. If the data set contains only 1 event, then a non-event-based waveform is generated.

Replace any unallowed characters with ' _ '

If the channel name contains any characters which are not permitted, such as "+,-, ...", these characters are replaced by ' _ '.

Import reduced data

A channel can contain reduced data. These reduced data consist of the data set's minimum, maximum, mean and RMS values. These values are imported into separate FAMOS channels.

The channel names are formed as shown below:

```
<ChannelName>_MIN,  
<ChannelName>_MAX,  
<ChannelName>_MEAN,  
<ChannelName>_RMS
```

Z-Axis data are taken as equidistant (offset: value1, delta: value2 - value1)

The linear z-axis of the 3-dimensional waveform gets for the offset value the value1 from the imported z-axis data and the difference of value2 and value1 for the delta value.

If the option is not selected, the linear z-axis gets the offset "0.0" and the delta "1.0".

Generate simple waveform from Z-axis data (if not equidistant)

A simple waveform can be generated for the z-axis data. The 3-dimensional waveform gets the offset of "0.0" and the delta of "1.0" for the z-axis.

Options and Sequences

There are two ways to specify options:

- Options for the format's filter or options for the format's derivative data filters
- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

Example:

```
idFile = FileOpenFAS(FileName , "#Dewetron.dll|Dewetron Format|AlwaysXY=0|AlwaysEvents=0",0)
```

Option	Definition
AlwaysX/Y	0 - Generate simple waveform from equidistant data 1 - Always XY-waveform
AlwaysEvents	0 - Only generate events if there is more than 1 Event 1 - Always generate events
SubstInvalidChar	0 - Name is adopted without any checking 1 - Replace any unallowed characters with '_'
Channels	List with channel names separated by commas: <name1>,<name2> Important note: The channel names are stated as they are named in the data set. The option "SubstInvalidChar=1" is only run after the channel is located in the file. Both options remain independent in this way.
ReadReducedData	0 – don't import reduced data 1 – reduced data are imported to <ChannelName>_MIN, _MAX, _MEAN, _RMS
Z_TO_WAVEFORM	0 - The Z-axis data are taken as offset "0" and interval "1" 1 - The Z-axis data are taken as equidistant (offset: value1 , Interval: value 2 - value 1)
Z_TO_SEPARATE_WAVEFORM	0 - No simple waveform is generated from Z-axis data 1 - A simple waveform is generated from Z-axis data if not equidistant

Example of multiple channels

1st step: Creating the basis options

```
var_Options = "#Dewetron.dll|Dewetron Format|AlwaysXY=1|AlwaysEvents=0"
```

2nd step: Appending the individual channels

```
var_Options = var_Options + "|Channels="
```

```
var_Options = var_Options + "<name1>"
```

```
var_Options = var_Options + ", <name2>"
```

...

```
var_Options = var_Options + ", <nameN>"
```

3. Calling the import function


```
idFile = FileOpenFAS(FileName ,var_Options,0)
```

Important Note

To be able to work, the filter requires the Support-library "DWDataReaderLib" (V3.0.0.7) from the company DEWETRON in the folder

```
DWDataReaderLib64.dll -> "c:\Program Files\imc\shared\Extensions".
```

For copyright reasons, this library is not included in the product package. Please contact the company DEWETRON if necessary.

1.14 Dewetron DMD

Filter

Dewetron DMD

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Dewetron DMD	Dewetron.dll, Dewetron.409	10.0.0.0	*.dmd

Import:

- Normal / Event-based Waveforms
- X/Y Waveforms
- Reduced Waveforms
- Complex Waveforms
- Support of import in multithread mode

General notes:

The filter is based on the import interface (API) of Version 1.1 of the DMD Reader. In this minor version, the filter accordingly accepts all versions ≥ 1 . In consequence, further development of the Reader is possible without the filter's version checker rejecting the use of the Reader. When the major version is increased, incompatibility of the interface is assumed and an error message is posted.

The filter imports the data according to the following assignment of DMD sample types:

DMDReader_SAMPLETYPE_DOUBLE

- "simple data set" if all "sweeps" contain a sampling frequency
- "XY-data set" if any "sweep" contains no sampling frequency

DMDReader_SAMPLETYPE_REDUCED

The FAMOS data sets are generated as with “DMDReader_SAMPLETYPE_DOUBLE”. The channel names are accordingly supplemented with “<name>_min”, “<name>_max”, “<name>_avg” and “<name>_rms”.

DMDReader_SAMPLETYPE_SINT32

- “digital data set” if all “sweeps” contain a sampling frequency
- “digital time-stamped data set” if any “sweep” contains no sampling frequency

DMDReader_SAMPLETYPE_DOUBLE_VECTOR

Two data sets are generated to model this data type in FAMOS.

1. Data set:

A “segmented data set” with a segment length of the maximum possible dimension. This data set contains the individual vectors in one segment each. The Z-axis represents the time position.

2. Data set:

This data set records the length of the associated vector at the respective moment in time. The data set is formed as a:

- “simple data set” if all “sweeps” contain a sampling frequency
- “XY-data set” if any “sweep” contains no sampling frequency

The name is formed according the instruction “<ChannelName>_TimeAndDim”.

If the 2nd data set is a “simple data set”, then the time increment and offset are additionally plotted along the Z-axis of the 1st data set

Important note:

If a vector does not contain all elements of the specified dimension, then the vector is filled with “0”s up to the specified dimension. When the vector is subjected to operations, its current length is reflected by the 2nd data set.

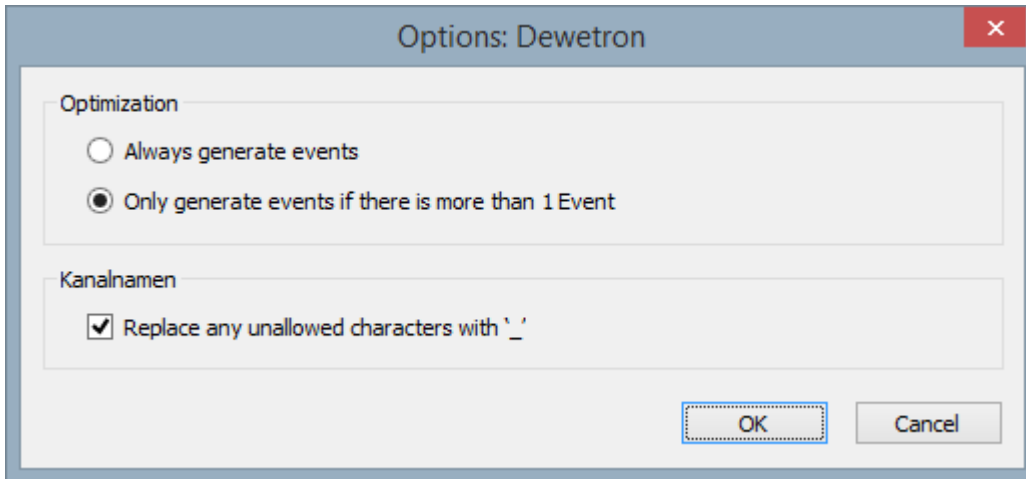
DMDReader_SAMPLETYPE_COMPLEX_VECTOR

The FAMOS data sets are generated as with “DMDReader_SAMPLETYPE_DOUBLE_VECTOR”.

The 1st data set contains the complex value pairs.

Sweeps

Sweeps are generated in the data set as events. Since there is always at least one sweep, an option is available by means of which to set whether the channel in FAMOS is created with just one event, or without any event.

Options**Always generate events**

Event-based waveforms are always generated

Only generate events if there is more than 1 Event

Event-based waveforms are generated if more than 1 event exists. If the data set contains only 1 event, then a non-event-based waveform is generated.

Replace any unallowed characters with '_'

If the channel name contains any characters which are not permitted, such as "+, -, ...", these characters are replaced by '_'.

Important note

In order to be able to run, the filter requires the support libraries "dmd_reader_api*" from the company DEWETRON in the folder...

dmd_reader_api_x64.dll -> "c:\Program Files\imc\shared\Extensions".

For reasons concerning updates, these libraries are not included in the scope of delivery. If needed, please contact the company DEWETRON.

1.15 DIAdem

Filter

National Instruments DIAdem

Type

Import/Export Filter

Description

Filter name	Filename of filter	Version	Import File Extension
DIAdem TDM / TDMS	TdmImportExport.dll	10.0.0.0	*.tdm, *.tdms

Import:

- equidistant unsegmented data in the TDM /TDMS format into FAMOS Normal Waveforms
- The time axis scaling properties are evaluated.
- Evaluation of the properties "Y_Gain" and "Y_Offset" for scaling the data
- Channels are imported either in or without associated data groups (optional)
- File header data are imported in the group FileInformation (optional)
- Support of import in multithread mode

Export:

- FAMOS Normal Waveforms to equidistant unsegmented data in the TDM /TDMS format
- Support of export in multithread mode

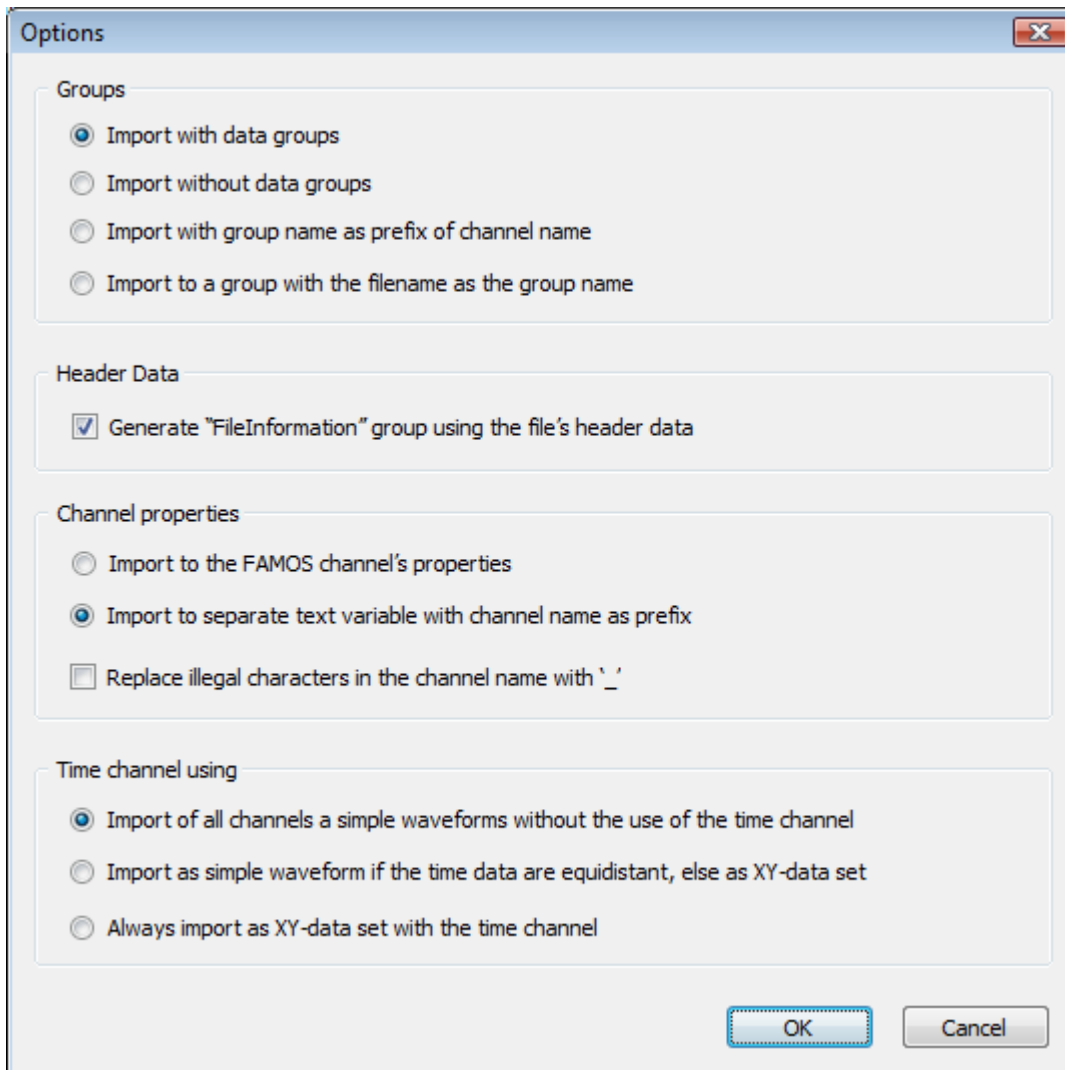
Description

Filter name	Filename of filter	Import File Extension
DIAdem ASCII	DIAdem_ASCII.FAS	*.dat + *.asc
DIAdem I16	DIAdem_PCI16.FAS	*.dat + *.I16
DIAdem I32	DIAdem_PCI32.FAS	*.dat + *.I32
DIAdem R32	DIAdem_PCR32.FAS	*.dat + *.R32
DIAdem R48	DIAdem_PCR48.FAS	*.dat + *.R48
DIAdem R64	DIAdem_PCR64.FAS	*.dat + *.R64
DIAdem R64_2	DIAdem R64_2.FAS	*.dat + *.R64

Import:

- X,Y data into FAMOS XY-waveforms
- R64_2 like R64, without adapting the name of imc naming conventions

Options



Groups

Import with data groups

The channels are imported with the data file's data groups.

Import without data groups

The channels are imported without data groups.

Import with group name as prefix of channel name

The data group name is prefixed to the channel names.

Import to a group with the filename as the group name

The channels are imported in a FAMOS group with the name of the data file. The group ordering within the file is not adopted.

Header Data

Generate "FileInformation" group using the file's header data

The file's header data are imported as text variables to a FAMOS group named "Fileinformation".

Channel Properties

Import to the FAMOS channel's properties

Channel information (metadata) is imported to the properties of the channel variable.

Import to separate text variable with channel name as prefix

Channel information (metadata) is imported to a text variable with the channel name as the prefix. This option serves exclusively to retain compatibility with sequences created with earlier versions of FAMOS. This option should no longer be used.

Replace illegal characters in the channel name with '_'

Illegal characters in channel names are replaced by '_'.

Use of the time channel

Import of all channels a simple waveforms without the use of the time channel

A data group's time channel is ignored.

Import as simple waveform if the time data are equidistant, else as XY-data set

The system checks whether the time channel's data are equidistant. If they are, a Normal Waveform is generated. If the time channel is not equidistant, an X/Y-data set is imported.

Always import as XY-data set with the time channel

There is no testing for equidistance. An X/Y-data set is generated.

Use of the options

There are two ways to specify options:

- Options for the format's filter or options for data filters derivative of the formats by means of the Options dialog.
These options are saved globally.
- Options in call parameters within the FileOpenFAS(...) function.
These options are saved locally in the sequence.

The options passed as call parameters overwrite the options which were set in the Options dialogs. If the value for an option is not specified, the option's default setting applies. Options which had been passed as call parameters are not saved.

Call parameters within the FileOpenFAS(...) function (import only):

```
FileOpenFAS(<FilePathName>,"#TDMImportExport.dll|DIAdem TDM / TDMS|ExpandGroups=0|  
FileInfoGroup=0|TimeChanUse=0|PropTextVarUse=0|SubstInvalChar=0",0)
```

"Groups"

Option	Parameter + Value
Import with data groups	ExpandGroups=0
Import without data groups	ExpandGroups=1
Import with group name as prefix of channel name	ExpandGroups=2
Import to a group with the filename as the group name	ExpandGroups=3

"Header Data"

Option	Parameter + Value
Generate "FileInformation" group using the file's header data	FileInfoGroup=1

"Channel Properties"

Option	Parameter + Value
Import to the FAMOS channel's properties	PropTextVarUse=0
Import to separate text variable with channel name as prefix	PropTextVarUse=1
Replace illegal characters in the channel name with '_'	SubstInvalChar=1

"Use of the time channel"

Option	Parameter + Value
Import of all channels a simple waveforms without the use of the time channel	TimeChanUse=0
Import as simple waveform if the time data are equidistant, else as XY-data set	TimeChanUse=1
Always import as XY-data set with the time channel	TimeChanUse=2

Important notes:

- For licensing reasons, certain National Instruments components necessary for using the filters must be obtained directly from National Instruments (nilibddc.dll a.o.). These can be downloaded from National Instruments homepage: <http://www.ni.com/white-paper/3727/en/>. On this page all aspects about the data format are described. At the bottom of the page is the link to download with the necessary DLLs : "c_dll_tdm.zip".
- Unpack the downloaded files and subfolders
 - from the folder "**dev\bin\64-bit**" and copy them to the folder: "**c:\Program Files\imc\shared\extensions**".

- Furthermore the **Visual C++ Redistributable for Visual Studio 2015 or 2017** is required. It will be installed with imc FAMOS version 7.4 or higher. For all previous versions of imc FAMOS these have to be installed additionally.

1.16 Digitek DataView

Filter

Digitek DataView

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Data View Binary File Format	DataViewImport.dll	10.0.0.0	*.bin

- Support of import in multithread mode

1.17 DSP Development Corporation DADiSP

Filter

DSP Development Corporation

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
DSP Development Corporation DADiSP	DSP_Dadisp_hdr.FAS	*.hdr, *.dat
DSP Development Corporation DADiSP	DSP_Dadisp_dad.FAS	*.dad

1.18 DTS

Filter

DTS (SLICEWave XML File Format)

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
DTS-Format	DTS.dll	10.0.0.0	*.dts

Import:

- Normal waveforms
- Channel properties
- Support of import in multithread mode

1.19 ETAS/Bosch

Filter

ETAS/Bosch

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
ETAS/Bosch MDF Format	ETAS_Bosch_Mdf.FAS	*.dat



Note

This import filter corresponds to the [MDF 3](#)  format. Current devices use the [MDF 4](#)  format.

Import:

- equidistant data into FAMOS Normal Waveforms

1.20 Excel (XLS)-Format

Filter

EXCEL

Typ

Import Filter

This functionality can only be used if a version of the spreadsheet program Microsoft® EXCEL which is supported is installed.

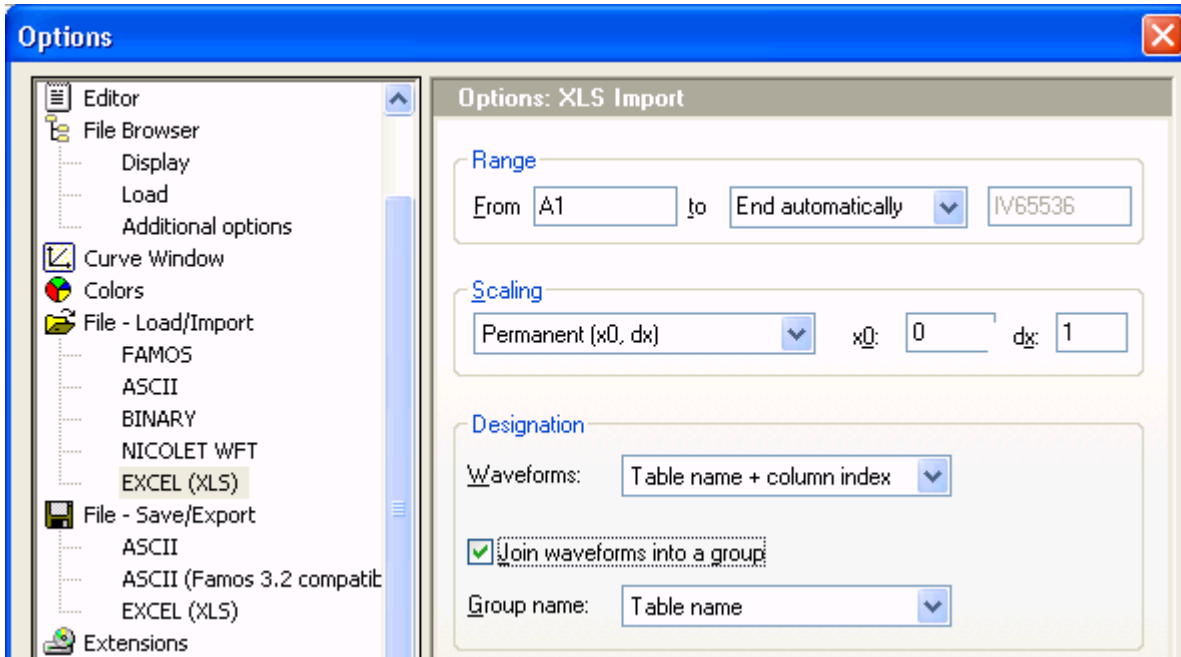
XLS-format is the default file format for the widely circulated spreadsheet program Microsoft® Excel. If Excel is installed parallel to FAMOS, XLS-files can be imported to FAMOS directly. For this purpose, Excel is remotely controlled by a mechanism called "OLE-Automation". Excel is started in the background, opens the desired file, transfers the data to FAMOS and is closed again.

Constraints

- Only the first table in an XLS-workbook is imported.
- Only numbers and values for the date can be imported.
- Import is performed column-by-column, i.e., the contents of a column make up a FAMOS-variable. The values to be imported must follow "seamlessly" upon each other; an empty cell in a table is interpreted as a column's end.
- Also make note of the various XLS-format versions. Excel 95, for instance, cannot read XLS-files saved in the Excel 2000-format.

To **load an XLS-file**, select the menu item "*Load file*", and in the file selection dialog which then appears, the format list entry "*EXCEL XLS*".

Excel-import comes with a number of options; the dialog with the controls for these options can be selected via the menu item "Extra"/"Options"/"File-Load/Import" -> "Excel" ->"Options", or via the button "Options" in the "Load file"-dialog, if Excel-format is selected in the format selection list.



Dialog: Options XLS-import

Range

Here you must specify the portion of the table to be read in. All references to cells are to be entered in the "A1"-notation normally used by Excel, where "A" denotes the first column and "IV" the last column (Number 256). This is followed by the row number

Examples

A1	Upper left cell
IV65536	Column 256, Row 65536 (max values from Excel 97 on)
Z1	Column 26, Row 1
AA10	Column 27, Row 10

From

Defines the upper left cell of the table portion to be read in.

to

Defines the bottom right cell of the table portion to be read in.

Read-in begins at the specified starting point (cell) and is carried out column-by-column, i.e. a FAMOS-variable is generated for each column. An empty cell in a column is interpreted as a column's end. Read in stops at the first empty column (in other words, if the cell in the selected starting row is empty or doesn't contain a numerical value).

"to> maximum ...": Reads column-by-column, until the selected range is imported or until the first empty column.

"to> End automatically:" Reads column-by-column until the first empty column.

In table #

Index of the imported table.

Scaling:

One data set is generated for each EXCEL-column. Here, you can enter how it is scaled.

Permanent (x0,dx): A normal waveform is generated and initialized with the x0/dx (or sampling interval/ pretrigger) specified here.

1st column contains scaling: The scaling information for all waveforms is derived from the first column read in.

If the option box "**Equidistant**" is activated the scaling column is assumed to be equidistant (constant distance between 2 adjacent readings) and a normal waveform is generated with the calculated x0/dx.

Otherwise, XY-data sets are generated, where the first column is used as the x-component for all further data sets.

Designation

Affects how the data sets loaded are named.

1st row contains name: The first row to read is assumed to contain the channel name and only afterwards do the actual data begin. Invalid characters for a variable name, such as <SPACE>, parentheses, etc., are replaced by an underline '_'.

Permanent name + serial index: The specified name is extended with the index of the data set read in (the first data set has the index 1)

Permanent name + column index: The specified name is extended with the Excel column index. This is different from the serial index, if read-in doesn't begin in the 1st column

Table name + ...: The name is formed from the Excel spreadsheet name and a number.

File name + ...: The name is formed from the Excel filename and a number.

Join waveforms into a group

If this option is activated, all the channels which are loaded are joined to form a group. The group's name is determined by the selection in the list "**Group name**".

Example

The following Excel table is to be imported:

	A	B	C	D	E	F
1	Prepared by: Joe Blow					
2						
3		Channel 1	Channel 2		Temperature	
4	11.04.02 10:17	12.12	123.11	A	22.1	
5	11.04.02 10:18	12.45	131.42	C	22.3	
6	11.04.02 10:19	12.59	139.73	D	22.4	
7	11.04.02 10:20	12.86	148.04	A	22.6	
8	11.04.02 10:21	13.09	156.35	A	22.7	
9	11.04.02 10:22	13.33	164.66	A	22.9	
10	11.04.02 10:23	13.56	172.97	H	23.0	
11						
12						
13						

EXCEL-Beispieltabelle

For this purpose, the following options are to be set:

The screenshot shows the configuration options for an Excel import. It is organized into three main sections:

- Range:** The 'From' field is set to 'A3', the 'to' dropdown is set to 'End automatically', and the 'IV65536' field is present.
- Scaling:** The dropdown menu is set to '1st column contains scaling', and the 'Equidistant' checkbox is unchecked.
- Designation:** The 'Waveforms' dropdown is set to '1st row contains name', the 'Join waveforms into a group' checkbox is checked, and the 'Group name' dropdown is set to 'Table name'.

Configuring a sample EXCEL import

Thus, a data group "Workbook2" is created in FAMOS, having 2 channels "Channel_1" and "Channel_2" (both of data type XY). Read-in is stopped at Column 4, since it contains no numerical values; and the column "Temperature" is also not imported.

Reference

In this section, manual loading of XLS-files is described. The options offered only permit loading of tables having simple structure. For sequences, corresponding functions exist, see FAMOS Function Reference, functions '[FileOpenXLS](#)', '[FileXLSColumnRead](#)' and '[FileXLSCellRead](#)'. The latter function can also be used to read out individual cells, including cells containing text. Thus, sequences can be programmed which import more complex table structures.

1.21 FlexPro

Filter

FlexPro

Type

Import/Export Filter

Description

Filter name	Filename of filter	Version	Import File Extension
FlexPro Database Format	FlexPro.dll	1.0.0.0	*.fpd

Import:

- Conversion table from FlexPro data structures into FAMOS data types

FlexPro data structure	FAMOS data type
Scalar value - Simple	Normal Waveform
Scalar value - Complex	Complex Waveform (Real-/Imaginary-part)
Data Series - Simple Y (without X-info)	Normal Waveform
Data Series - Complex Y (ohne X-Info)	Complex Waveform (Real-/Imaginary-part)
Data Matrix - Simple Y (ohne X-Info)	Normal Waveform - segmented
Data Matrix - Complex	Y (without X-info) "Complex Waveform (Real-/Imaginary-part) - segmented
Signal - Simple - X,Y	XY-waveform
Signal - Complex - X,Y(r,i)	1. XY-waveform X,Y(r) 2. XY-waveform X,Y(i)
Signal Series - Simple - X,Y	XY-waveform - segmented - 1st X-segment is duplicated
Signal Series - Complex - X,Y(r,i)	1. XY-waveform X,Y(r) - segmented - 1st X-segment is duplicated 2. XY-waveform X,Y(i) - segmented - 1st X-segment is duplicated
Signal Series 2D-X - Simple - X,Y	XY-waveform - segmented
Signal Series 2D-X - Complex - X,Y(r,i)	1. XY-waveform X,Y(r) - segmented 2. XY-waveform X,Y(i) - segmented
Space Curve - Simple - X,Y,Z	1. Normal Waveform for X 2. Normal Waveform for Y 3. Normal Waveform for Z

FlexPro data structure	FAMOS data type
Space Curve - Complex - X,Y(r,i),Z	1."Normal Waveform for X 2."Complex Waveform (Real-/Imaginary-part) for Y 3."Normal Waveform for Z
String	Text variable This data type can also exist as the data type in components.

Import notes

- Complete installation of FlexPro required, use of programming interface (flexpro.tlb, BCGCBPro.tlb) which supports FlexPro for Visual Basic. FlexPro is automatically started and stopped when data are imported.
- The FlexPro database can be organized on folders. The paths are prefixed to the names in order to obtain unique names.

Export:

- FPAccess interface (COM-Interface); complete installation not necessary. The FPAccess interface only provides interface for creation of a FlexPro database and for writing data sets.
- Normal Waveforms are exported as signals (X,Y), since FlexPro does not allow additional parameters with simple data series.
- Any data sets are divided into individual signals and saved to a folder with the signal name.

1.22 G.i.N.

Filter

G.i.N.

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
G.i.N. CAN Logger	Gin_Fcd.FAS	*.fcd

Import:

- (X,Y) - pairs into FAMOS XY-waveforms

1.23 Google Earth

Filter

Google Earth

Type

Export Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Google Earth Export	GoogleEarthExport.dll	10.0.0.0	*.KML

Export:

- Export of longitude and latitude specifications as a KML file. The data must be specified in $1/10^7$ degrees or degrees.
- Support of export in multithread mode

Note:

- Large KML files are no longer displayed in GoogleEarth. No GPS data should be used which contain more than 65520 points per data set. In order to reduce the amount of points, it would be possible to use the function "MvMean", for example.
- The longitude- and latitude-data are saved as an XY-data set.
- Alternatively, the two components are transferred individually. In this variant method, an input window appears, in which the assignment is requested.

Example 1:

Data reduction with moving mean value:

```
long = MvMean(long, 1,1) ; = x
lat = MvMean(lat, 1,1) ; = y
```

Transferred as a XY-data set

```
dataXY= xyof( long, lat)
Filesave( "C:\temp\MyTrack.kml", "#GoogleEarthExport.dll|Google Earth
Export",
0, dataXY)
```

Example 2:

Transfer with two equidistant data sets. Upon execution, a prompt for the assignment appears.

```
Filesave( "C:\temp\MyTrack.kml", "#GoogleEarthExport.dll|Google Earth
Export",
0, long, lat)
```


1.24 Gould

Filter

Gould

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension	Device Types
Gould DSO720	Gould_DSO720.FAS	*.dat	Datasys 720
Gould HighRes DataSys 840	Gould_HIRES840.FAS	*.dat	Datasys 840, High Res Modus

Import:

- equidistant data into FAMOS Normal Waveforms

1.25 Grabau Sound Power

Filter

Grabau Sound Power

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Grabau Sound Power GR880/SLP	Grabau_SoundPower.FAS	*.slp
Grabau Sound Power INGA	Grabau_SoundPowerINGA.FAS	*.*32

1.26 Graphtec

Filter

Graphtec

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Graphtec GBD 820/220	GraphtecGBD.FAS	1.3	*.gbd
Graphtec GBD 7000	GraphtecGBD7000.FAS	1.5	*.gbd
Graphtec GL Series 820/840/220/240/900/7000	GraphtecGL.dll, GraphtecGL.409	10.0.0.0	*.gbd

Import:

- Normal Waveform
- Support of import in multithread mode

1.27 Haag

Filter

Haag

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension	Device Types
Haag EWS92/94 Harmonic ASCII	Haag_EWS_Os_IEC.fas	*.asc	Haag EWS 92/94
Haag EWS94 Harmonic ASCII	Haag_EWS_Os.fas	*.asc	Haag EWS 94
Haag EWS94 ASCII	Haag_EWS.fas	*.asc	Haag EWS 94

Import:

- equidistant data into FAMOS Normal Waveforms
- data into FAMOS complex waveforms (R,I), (EWS_ASC, EWS_OS)

1.28 HBM

Filter

HBM

Type

Import/Export

Description

Filter name	Filename of filter	Version	Import File Extension
Catman 5.0 binary file format	CatmanImportExport.dll	10.0.0.0	*.bin

Import:

- integer and floating point data types into imc FAMOS Normal Waveforms
- The names of the data sets are imported in the same way that they are registered in the file to be imported. If the name contains special characters, then in imc FAMOS these variables must be written in {} brackets.

V1.1.0.0:

- Loading time improved

V10.0.0.0:

- Support of import in multithread mode

Export:

- Normal Waveform with integer and floating point data types
- Support of export in multithread mode

Notes:

- imc FAMOS' multi-component and event-based waveforms can be partitioned using imc FAMOS into individual components or events for exporting purposes.
- An associated Catman Test file is saved.

1.29 HBM PNRF

Filter

HBM PNRF

Type

Import

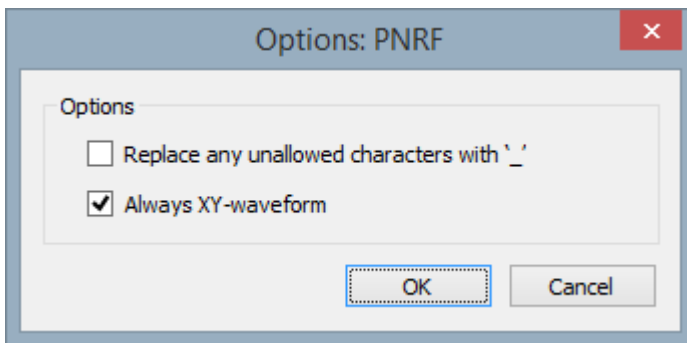
Description

Filter name	Filename of filter	Version	Import File Extension
HBM PNRF	PNRF.dll, PNRF.409	10.0.0.0	*.pnrf

Import:

- integer and floating point data types into FAMOS Normal data sets
- simple FAMOS data sets
- FAMOS data sets with events
- if data in "sweeps" are recorded in a 2nd data track, an additional channel with the channel name and the suffix "_sweeps" is generated in FAMOS.
- optional import of time-stamped data (XY datasets)
- Support of import in multithread mode

Options



Replace any unallowed characters with '_'

If the channel name contains any characters which are not permitted, such as "+, -, ...", these characters are replaced by '_'.

Options and Sequences

There are two ways to specify options:

- Options for the format's filter or options for the format's derivative data filters
- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

Option	Definition
SubstInvalidChar	0 - Name is adopted without any checking 1 - Replace any unallowed characters with ' _ '
Channels	List with channel names separated by commas: <name1>,<name2> Important note: The channel names are stated as they are named in the data set. The option "SubstInvalidChar=1" is only run after the channel is located in the file. Both options remain independent in this way.
AlwaysXY	0 - Generate simple waveform from equidistant data 1 - Always XY-waveform if time data available

Examples

```
FileOpenFAS("Messung_1.pnrf", "#PNRF|HBM PNRF",0)
FileOpenFAS("Messung_1.pnrf", "#PNRF|HBM PNRF|SUBSTINVALIDCHAR=1",0)
FileOpenFAS("Messung_1.pnrf", "#PNRF|HBM PNRF|Channels=Spannung",0)
```

Komponente „SetupPNRFReader.exe“

For the purpose of elementary access to the “*.pnrf” file, the FAMOS import filter needs an API provided by the company HBM via a COM (Component Object Model) interface. Installation of the necessary libraries, as well as their registration in Windows, is performed by installation of the PNRF Reader Toolkit.

The „PNRF Reader Toolkit 32/64-bit” can be downloaded for free on the HBM company’s Internet page below:

["https://www.hbm.com/en/2475/support-downloads-perception/#c95059"](https://www.hbm.com/en/2475/support-downloads-perception/#c95059)

The company imc is not authorized to supply the setup directly.

1.30 HDF5

Filter

HDF5

Type

Import/Export filter

Description

Filter name	Filename of filter	Version	import file extension
HDF5 Format	HDF5.dll	10.0.0.0	*.h5, *.hdf5

Import:

- integer and floating point data types in normal FAMOS-data sets
- 2-dimensional arrays in segmented data sets
- Matrices having more than 2 dimensions are resolved to single variables by means of the channel's filename.
- Text and text arrays
- Time-stamped data as (compound) data structure with measurement time and measured value
- Data with data widths of 1 bit as digital channels
- Data sets of variable length as Time Stamp Ascii (TSA)
- Complex numbers as "compound" data structure
- Metadata as attributes (trigger time, sapling rate, units)
- Support of import in multithread mode

Export:

- Simple channels as data set
- XY-channels as data set with "compound" data type
- Segmented channels as a 2-dimensional table
- Event-based channels as a group consisting of individual data sets
- Text variables, text arrays
- Digital channels
- Channel parameters as attributes of the data set or of the data group (trigger time, sampling rate, units)
- Support of export in multithread mode

Import options

The set of options specifies attribute names used by the import filter to import additional variable parameters.

The HDF5 format can contain a list of key/value-pairs for each data set. Keys (attribute names) for pre-defined metadata are not specified. In order to import a parameter of a variable, attribute names can be assigned for each parameter, according to which the value can later be extracted from the list of attributes and imported to the variable's parameter.

Note: All attributes of a data set are always imported to the variable's or the channel's properties. After an initial import, the attributes are known and the options can be modified accordingly.

Forming the variable's name:

- Replace any invalid characters with '_'.
The characters " ,+- ()*/^={}|@<>!;" are replaced with the character '_'.
- Assign an index to duplicate channel names
If multiple variables in the data file have the same name, assign a consecutive index to these channels. If this option is not selected, only the 1st channel of that name is imported.
- Group name
Data sets in the HDF5 format can be nested in groups of any nesting depth. Tree diagram structures having more than 2 levels can not be represented in FAMOS directly. The individual successive group names are separated by a selectable separator character. The filter can optionally prefix the channel name to the group path.
- Attribute name
The name of the variable/channel can optionally be extracted from an attribute. If the attribute is not available, the name of the data set is used.

With multi-dimensional matrices, any dimensions > 2 are imported as independent variables. The index of the dimension is appended to the variable's name.

Members of structures (compounds) are imported as independent variables. The name of the member is appended to the variable's name. Structures can, in turn, contain other structures. In such cases, the names of all structure elements in the hierarchy are incorporated into the name.

Options for special case: HDF5 compounds having 2 structure elements

Structures having 2 components (members) can be joined to XY-data sets (e.g. as time-stamped data).

- **Name of X-reference**
The name determined which structure element contains the data for the 2 components.
- **Resolve structure with 1, 2 elements**
This option lets the user set whether the structure's elements are to be imported as simple data sets or only the XY-variable is created.
- **Create simple data set when equidistant**
When this option is selected, and the x-component consists of equidistant measured data, a simple data set with an offset and increment is generated.

The default set of attributes can be assigned by clicking on the button "Default Import Attribute Set".

Attribute set:

Attribute for	Default name	Description
Unit:	"Unit"	Unit for the variable/channel
Comment:	"Comment"	Text as the comment
Time created:	"Tcreate"	The creation time in the specified time format
Delta:	"Xdelta" "Zdelta" "Pdelta"	Resolution of X-axis Resolution of Z-axis, in case of segmentation Resolution of parameter in case of 2-component channels (e.g. frequency, in case of spectra)
Offset:	"Xdelta" "Zdelta" "Pdelta"	Offset for X-axis Offset for Z-axis, in case of segmentation Offset of parameter in case of 2-component channels (e.g. fundamental frequency, in case of spectra)
Unit:	"Xdelta" "Zdelta" "Pdelta"	Unit of X-axis (e.g. "s") Unit of Z-axis (e.g. "count") Unit of parameter (e.g. "Hz")
Name of the X-reference member	"ts"; "Xref"	Name of the structure component to be used as the 2nd component (e.g. X-axis)

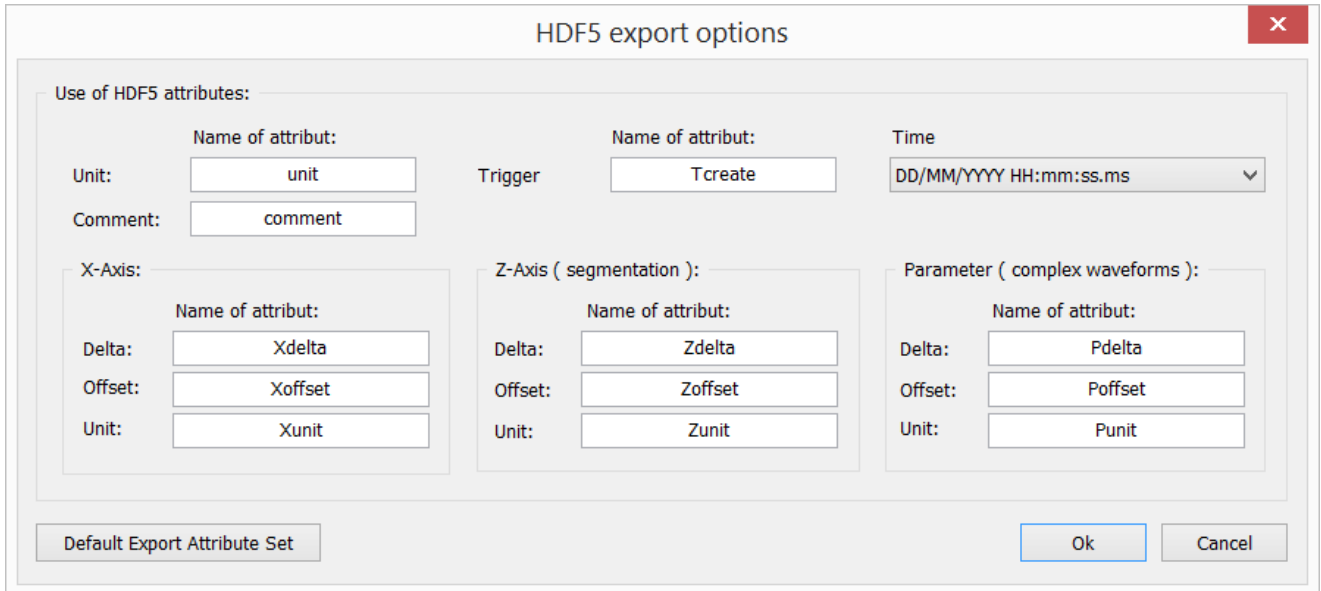
Error messages

- **Import to sequences with message about non-supported data types**
When new data types can not be imported, a message is posted. However, if the data affected are not necessary, it is possible to suppress this message in in automated sequences.

Export options

FAMOS variables are exported as a data set, or in the case of event-based channels, as data sets in a group. In order to add the extra channel parameters to the HDF5 data sets as attributes, a set of attribute names is **necessary**. This set of attributes is freely definable and can be modified to accommodate an existing external importer.

Note: For purposes of FAMOS exporting and importing procedures, the attribute sets must be identical for export and import, in order to get the metadata back.



Attribute set:

The set of attributes is identical to the default set for importing.

Assignment of the data types:

The following table contains the FAMOS variable type and the HDF5 data structures generated.

FAMOS data set	HDF5 data set
Simple data set	Data set
Simple data set, segmented	Data set + attribute: "IMCSEGMENTS", value: "true"
Simple data set, event-based	Group: <ChannelName> + attribute: "IMCSPECIALGROUP", value: "Events" Data set: "Event 1" Data set: "Event 2" ...
Simple data set, event-based, segmented	Group: <ChannelName> + Attribute: "IMCSPECIALGROUP", Value: "Events" Data set: "Event 1" + Attribute: "IMCSEGMENTS", Value: "true" Data set: "Event 2" + Attribute: "IMCSEGMENTS", Value: "true" ...

XY-data set	Data set with compound-subset + Attribute: "imcCompound", Value: "XY"
XY-data set, segmented	Data set with compound-subset + Attribute: "IMCCOMPOUND", Value: "XY" + Attribute: "IMCSEGMENTS", Value: "true"
XY-data set, event-based	Group: <ChannelName> + Attribute: "IMCSPECIALGROUP", Value: "Events" Data set: "Event 1" with compound-subset + Attribute: "IMCCOMPOUND", Value: "XY" Data set: "Event 2" with compound-subset + Attribute: "IMCCOMPOUND", Value: "XY" ...
XY-data set, event-based, segmented	Group: <ChannelName> + Attribute: "IMCSPECIALGROUP", Value: "Events" Data set: "Event 1" with compound-subset + Attribute: "IMCCOMPOUND", Value: "XY" + Attribute: "IMCSEGMENTS", Value: "true" Data set: "Event 2" with compound-subset + Attribute: "IMCCOMPOUND", Value: "XY" + Attribute: "IMCSEGMENTS", Value: "true" ...
Complex data set	Data set + Attribute: "IMCCOMPOUND", Value: "RI" Data set + Attribute: "IMCCOMPOUND", Value: "MagRad" Data set + Attribute: "IMCCOMPOUND", Value: "MagDeg" Data set + Attribute: "IMCCOMPOUND", Value: "DbRad" Data set + Attribute: "IMCCOMPOUND", Value: "DbDeg"
TSA channel	Data set + Attribute: "IMCCOMPOUND", Value: "TSA"
Digital channel	Data set with 1 Byte per sample (internal compressed and automatically decompressed upon exporting)
Text	Data set
Text array	Data set with Array-subset

Special attributes

The special attributes "imc..." describe the content of the data set.

These attributes are set internally and serve to support the special properties of the FAMOS data format.

Special attribute	Content	Description
IMCSPECIALGROUP	"Folder"	This group is a FAMOS.

IMCSPECIALGROUP	"Events"	This group has been generated by the exporter for the purpose of storing the individual events' data sets in it. The importer uses this attribute in order to generate an event-based channel from the group.
IMCSPECIALGROUP	"EventsZOX0"	Like "Events", "ZOX0" means that for the import, the Z-axis is used as the time axis.
IMCCOMPOUND	"XY"	X: significance is determined by the unit Y: significance is determined by the unit
IMCCOMPOUND	"MagRad"	X: Phase in Radians Y: Amplitude
IMCCOMPOUND	"MagDeg"	X: Phase in degrees Y: Amplitude
IMCCOMPOUND	"DbRad"	X: Phase in Radians Y: Amplitude in Decibels
IMCCOMPOUND	"DbDeg"	X: Phase in degrees Y: Amplitude in Decibels
IMCCOMPOUND	"TSA"	X: Structure consisting of Year, Month, Day, Hour, Minute, Second Y: Data of variable length
IMCSEGMENTS	"true"	The importer uses this information to generate segments if the size of the 2nd dimension is "1". The purpose of this attribute is to ensure the data set is always segmented. By default, segmented channels are only generated when the 2nd dimension is > 1.

The filter uses software components which are employed in accordance with the license below:

HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright (c) 2006-2018, The HDF Group.

NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright (c) 1998-2006, The Board of Trustees of the University of Illinois.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted for any purpose (including commercial purposes) provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or materials provided with the distribution.
3. Neither the name of The HDF Group, the name of the University, nor the name of any Contributor may be used to endorse or promote products derived from this software without specific prior written permission from. The HDF Group, the University, or the Contributor, respectively.

DISCLAIMER:

THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS "AS IS" WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED. IN NO EVENT SHALL THE HDF GROUP OR THE CONTRIBUTORS BE LIABLE FOR ANY DAMAGES SUFFERED BY THE USERS ARISING OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.31 HEAD acoustics

Filter

Head acoustics

Type

Import/[Export](#)  Filter

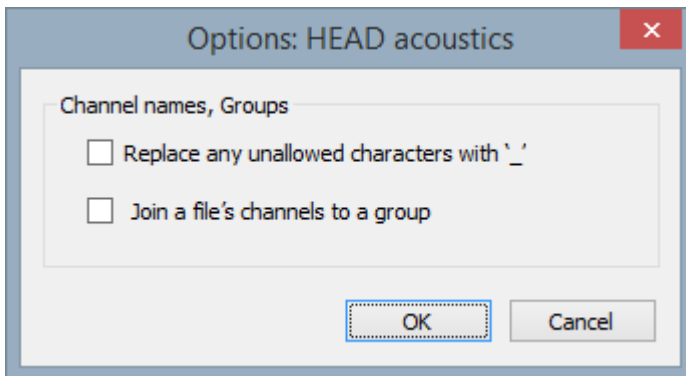
Description

Filter name	Filename of filter	Version	Import File Extension
Head acoustics	hdfimpdll.dll, hdfimpdll.409	10.0.0.0	*.hdf

Import:

- equidistant data into FAMOS Normal Waveforms
- Support of import in multithread mode

Options



Channel names, Groups

Replace any unallowed characters with '_'

If the channel name contains any characters which are not permitted, such as "+, -, ...", these characters are replaced by '_'.

Join a file's channels to a group

This option generates a group having the name of the file imported as its group name. All channels are imported to this group.

Use channel names from the commen field

When this option selected, the channel names are used from the field ";#ext name str:". If this field is empty, than the names are used from the field "name str:".

Optionen and Sequences

There are two ways to specify options:

- Options for the format’s filter or options for the format’s derivative data filters
- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

Example:

```
idFile = FileOpenFAS(FileName ,"#hdfimpdll.dll|HEAD acoustics|SubstInvalidChar=1|
Channels=Channel1,Channel20",0)
```

Option	Definition
SubstInvalidChar	0 - Name is adopted without any checking 1 - Replace any unallowed characters with ' _ '
Channels	List with channel names separated by commas: <name1>,<name2>,<name*>,<name1?> Important note: The channel names are stated as they are named in the data set. The option "SubstInvalidChar=1" is only run after the channel is located in the file. Both options remain independent in this way.
USECOMMENTNAMES	0 - Names from the field "name str:" 1 - Names from the comment field ";#ext name str:", when empty than from the field "name str:"

Example of multiple channels

```
1st step: Creating the basis options
var_Options = "#hdfimpdll.dll|HEAD acoustics"
```

```
2nd step: Appending the individual channels
var_Options = var_Options + "|Channels="
var_Options = var_Options + "<name1>"
var_Options = var_Options + ", <name2>"
...
var_Options = var_Options + ", <nameN>"
```

3rd Calling the import function

```
idFile = FileOpenFAS(FileName ,var_Options,0)
```

Export:

General notes:

The formats imc FAMOS (*.dat) and HEAD acoustics (*.hdf) differ so substantially that export of FAMOS variables to HEAD acoustics format is subject to certain restrictions. The export process must conform to the HEAD acoustics format specification V2.1.

There are two fundamental issues regarding export:

- Is the FAMOS variable's format supported by the hdf-format?
- Can the variables be exported jointly to one data file?

Exportable FAMOS variables

- simple data sets,
- X/Y-data sets,
- Complex data sets (spectra)

FAMOS variables which either are not or are only conditionally exportable

- digital channels 1),
- TSA ,
- segmented, event-structured channels,
- polar coordinates in [rad] angle units 2)
- texts, text arrays
- groups 3)

- 1) A digital channel can be encoded into the most significant bit of an analog channel. Such a channel is designated as "pulse data". The "digital" channel name is not adopted in the format. For export purposes, a reference in the analog channel variable to a digital channel would be required, or a generally applicable convention on unique nomenclature of digital channels, for the purpose of subsequent assignment.
- 2) The channel is to be converted to the unit [°].
- 3) The group is dissolved.

Export file, capture of all channels

Since the FAMOS Variables list, even when all non-exportable variables are excluded, can contain a collection of various variable types (channels, groups), data formats (int, real) and data structures (simple, XY, complex), it is only possible under certain conditions to export all variables in a shared file.

Export to a shared export file can only succeed on one of the following conditions:

- Exclusively simple channels all of the same length and having an identical basis (offset, increment, physical unit).
- Exclusively X/Y-channels all of the same length, whose X-components have identical values all in the same physical unit.
- Exclusively spectra of identical spectral resolution.

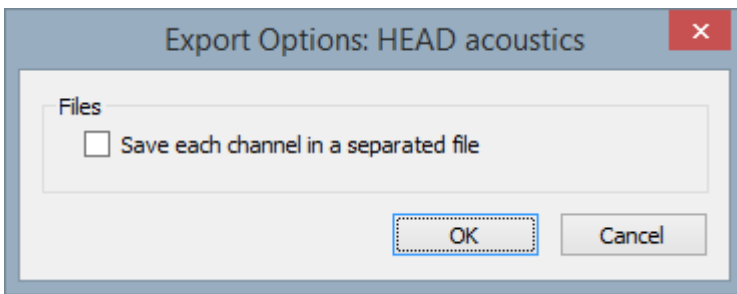
- If exclusively simple channels are to be exported, which only don't have an identical x-increment, they can still be exported to a shared file if:
 - the increment of one individual channel is an integer factor of the largest increment among all the channels, and
 - if the product of the increment and length results in the same value. In the time domain, this would pertain to the measurement duration (if a sampling rate is twice as high, then there would have to be twice as many values).

If these shared channel properties are not present, then the channels must be exported to separate data files. Export then must be configured using the associated option; it will not be performed automatically. The filename for export is formed from the user-specified filename as the prefix plus the channel name itself.

If this option is not selected, the channels are inspected to see if they meet the required properties. If they do not, an error message is posted.

Filter options

Export filtering options can be set globally, via the Options dialog:



or specified as a call parameter of the function "FileSave" within a sequence.

When call parameters are missing from the "FileSave" function, the global settings apply.

Sequenz parameter	Definition
SINGLEFILES= <0 1>	Enables/disables saving of single files on the basis of the channel name 0: channels exported to one file 1: channels exported to separate files

Examples

```
FileSave("Test.hdf", "#hdfimpdll.dll|HEAD acoustics",0,CH1,CH2)
```

```
FileSave("Test.hdf", "#hdfimpdll.dll|HEAD acoustics|SINGLEFILES=1",0,CH1,CH2)
```

1.32 HERF4

Filter

HERF4 (Fa. HYDAC)

Type

Import/Export Filter

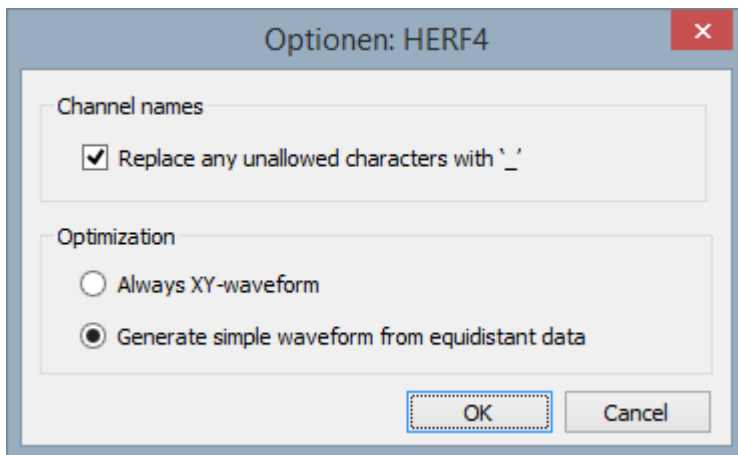
Description

Filter name	Filename of filter	Version	Import File Extension
HERF4-Format	herf4.dll, herf4.409	10.0.0.0	*.herf4

Import:

- Normal Waveforms
- XY-data sets
- Event-based Waveforms
- Support of import in multithread mode

Options



Replace any unallowed characters with '_'

If the channel name contains any characters which are not permitted, such as "+, -, ...", these characters are replaced by '_'.

Always XY-waveform

Always generates an X/Y-waveform

Generate simple waveform from equidistant data

When this option is used, the system investigates the X/Y-waveform to see if the X-values are equidistant. If they are adequately equidistant, a "simple waveform" is generated. Simple waveforms are easier to process in sequences.

Options and Sequences

There are two ways to specify options:

- Options for the format's filter or options for the format's derivative data filters
- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

Option	Definition
SubstInvalidChar=<0 1>	0 - Name is adopted without any checking 1 - Replace any unallowed characters with ' _ '
AlwaysXY=<0 1>	0 - Generate simple waveform from equidistant data 1 - Always XY-waveform

Examples

```
FileOpenFAS("Measurement_1.herf4", "#Herf4.dll | HERF4 Format", 0)
```

```
FileOpenFAS("Measurement_1.herf4", "#Herf4.dll | HERF4 Format | SUBSTINVALIDCHAR=1", 0)
```

1.33 HIOKI measurement devices

Filter

HIOKI measurement devices

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension	Device Types
Hioki Recordscopes 88xx	Hioki_Rec88xx.FAS	*.mem, *.rec	8826,8835,8841, 8842,8807,8808 V3.0
Hioki Recordscopes 8825,8840,Dig	Hioki_Rec882540Dig.FAS	*.* (*.dat,...)	8825,8840,Digital
Hioki MR-8847 Hioki MR-8875	Hioki.dll ,Hioki.409 V10.0.0.0	*.mem, *.rec	MR-8847, MR-8875

Import:

- equidistant data into FAMOS Normal Waveforms
- text
- Support of import in multithread mode

1.34 Hi-Techniques

Filter

Hi-Techniques

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Hi-Techniques HT600	HiTechniques_HT600.FAS	*.dat

Import:

- equidistant data into FAMOS Normal Waveforms
-

1.35 Horiba

Filter

Horiba

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
STARS	Stars.FAS	*.*

1.36 Hydrotechnik

Filter

Hydrotechnik

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Hydrotechnik Hydrocom	Hydrotechnik_Hydrocom.fas	*.mwf
Hydrotechnik Hydrocom TXT	Hydrotechnik_HydrocomTXT.fas	*.txt
Hydrotechnik MWF-Format	HydrotechMWF2.dll , V10.0.0.0	*.mwf

Import:

- equidistant data into FAMOS Normal Waveforms
- Support of import in multithread mode

Notes:

- The "Hydrotechnik MWF-Format" Filter supports *.wmf file versions > 7.

Important Note

To be able to work, the filter requires the Support-library HTMWFLibrary.dll, HTMWFLibraryx64.dll" (as of V3.4.8.51)" from the company Hydrotechnik.

The libraries are expected in the directory:

HTMWFLibraryx64.dll -> "x:\Program Files\imc\shared\Extensions".

For copyright reasons, this library is not included in the product package. Please contact the company Hydrotechnik if necessary.

Additionally the libraries require [Microsofts Visual C++ 2008 Redistributable Package \(x64\)](#). These are **no longer** installed with the installation of the Hydrocom libraries!

1.37 iba

Filter

iba

Type

Import Format

Description

Filter name	Filename of filter	Import File Extension
iba IBAAnalyzer	IBAAnalyzer.fas	*.dat

Import:

- equidistant data into FAMOS Normal Waveforms
-

1.38 IEEE Standard Common Format

Filter

IEEE Standard Common Format for Transient Data Exchange for Power Systems

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
IEEE Comtrade	IEEE_Comtrade.FAS	*.cfg + *.dat

Import:

- (X,Y)-data pairs into FAMOS XY waveforms
-

1.39 imc Bus Filter

Filter

imc Bus Filter

Type

Import Filter

Important

The Bus Filter needs the FAMOS 6.2 installation.

Description

Filter name	Filename of filter	Version	Import File Extension
imc Bus Format	BusFilter.dll	20.2.0.0	*.raw
Vector CAN Bus Logging	BusFilter.dll	20.2.0.0	*.asc, *.log
PEAK Trace Format	BusFilter.dll	20.2.0.0	*.trc

Import:

- CAN Bus Logging data into the formats
 - imc Time Stamp ASCII (*.log)
 - Vector ASCII (*.asc)
 - Vector MDF (*.mdf)
 - Support of multiple busses in Vector *.asc, *.log files
 - SAEJ1939 Support
 - PEAK Trace format
- The channel information can be specified from the following database files:
 - imc CAN Assistant (*.cba)
 - imc Blob (*.fdb)
 - Vector CAN Database (*.dbc)
 - LIN Description File (*.ldf)
 - FlexRay Description File (*.FRY)
- Support of import in multithread mode

General functioning

The Bus Filter serves the purpose of extracting values from Logging files containing binary or ASCII data sets (frames) along with associated time stamps. On the basis of binary control data (message- and channel-descriptions), data are extracted from the logged data sets and assigned to FAMOS channels.

The binary control data

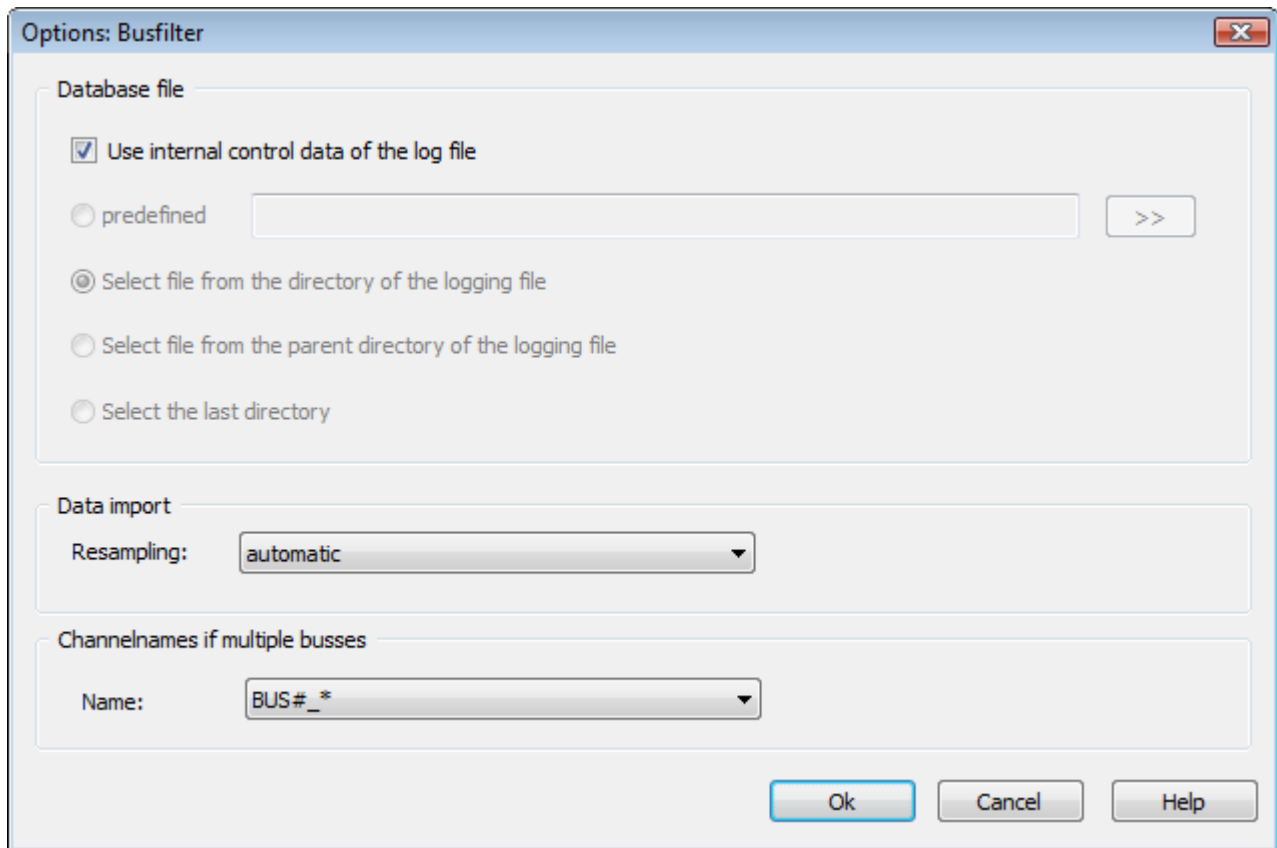
The control data take an internal data format for imc systems. They contain the information on the structure of the logging data as well as on the data channels whose contents are transported. The binary control data may be located in the data set (*.raw) or be generated in the filter itself from external database files.

The following database files are supported:

- Vector Database (*.dbc)
- imc CAN Assistant (*.cba)
- imc binary control file (*.fdb)

When files in the formats "Vector Database" and "imc CAN Assistant" are used as the basis, the filter contains a built-in converter for generated the necessary binary control data. If the control data are expressed in the internal imc format as a control file, then they can be used directly.

Options



The options under "Database File" determine the available selection for a basis format.

The options in detail are as follows:

Use internal control data of the log file

If the control data are already integrated into the Logging data set, no additional specification of the database is required.

predefined

The basis file to use is specified directly. Specifying the relative pathname ("..\testfile.cba") is permitted. The relative pathname refers to the data file currently open. The specified basis file is used for all logfiles.

A "*" instead of the filename ("..*.cba") enters the name of the logfile at the position of the "*".

Select file from the directory of the logging file

An "Open File" dialog is displayed. The default folder is the folder of the logfile which is open.

Select file from parent directory of the logging file

An "Open File" dialog is displayed. The default folder is the folder one level above the folder of the logfile which is open.

Select last directory

An "Open File" dialog is displayed. The default folder is the folder from which the last basis file was opened.

The options under "Data import" determine specifications for generating the data channels.

The options in detail are as follows:

Resampling: "automatic"

If the control data contain specifications of the channel's or the frame's interval, then the channel is resampled according to this specified interval.

In FAMOS, a simple data set is generated with specifications of the interval and offset.

If there is no specified interval, an X/Y data set is created with an X-component as the time specification.

Resampling: "Time-stamp"

The data in the channels are supplied with time stamps.

In FAMOS an X/Y data set is created with an X-component as the time specification.

Channel names at multiple busses

Option	Channel name
*	Channel name only
BUS#_*	"BUS" + identifier + "_ " + channel name
*_BUS#	Channel name + "_BUS" + identifier

Parameter of function FileOpenFAS

The "Parameter" part of `<TxFormat>` ("#DLLName|FormatName|Parameter") of the `FileOpenFAS` function contains following values:

Parameter	Definition
1	<p><name>: name of database file (*.cba, *.dbc, *.fdb)</p> <p><empty>: The internal control data of the logfile will be used.</p>
2	<p>Resampling of channels:</p> <p>0: automatic (if sample rate is available)</p> <p>1: Time-stamp</p> <p><empty>: the value of the options is used</p>

Example for calling with an external database file:

```
FileOpenFAS("test.raw", "#BusFilter.dll|imc Bus Format|C:\\test.dbc|1",0)
```

Example for calling with internal control data:

```
FileOpenFAS("test.raw", "#BusFilter.dll|imc Bus Format||1",0)
```

1.40 imc STUDIO Metadata

Filter

imc STUDIO Metadata import

Typ

Import Filter

Important Note

This filter requires imc FAMOS version 6.3 or higher.

Description

Filter name	File name of filter	Version	Import File Extension
imc STUDIO Metadata files	imcSTUDIO_MetadataImport.dll	10.0.0.0	*.csv, *.csv.lnk

Import:

- Metadata are imported with one or more metadata tables.
- All metadata are imported in a common group "Metadata".
- The file is expected to adhere to Unicode format, just as imc STUDIO exports it.
- All metadata are imported as text.
- Support of import in multithread mode

1.41 Instron

Filter

Instron Format

Typ

Import Filter

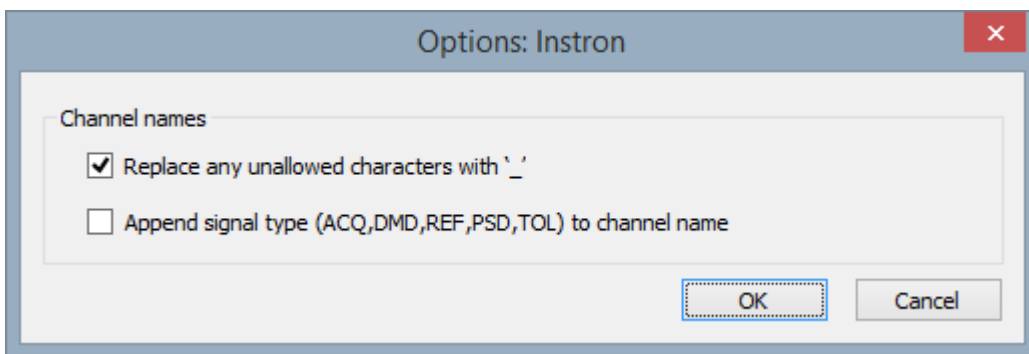
Description

Filter name	Filename of filter	Version	Import File Extension
Instron Format	Instron.dll, Instron.409	10.0.0.0	*.acq

Import:

- equidistant data into FAMOS Normal Waveforms
- complex data into FAMOS Complex Waveforms
- Support of the "Cyclic RigSys" Format
- Support of import in multithread mode

Options



Replace any unallowed characters with '_'

If the channel name contains any characters which are not permitted, such as "+,-, ...", these characters are replaced by '_'.

Append signal type (ACQ , DMD , REF , PSD , TOL) to channel name

The file format specification allows to use a same channel name several times for different types of signals. With this option, channel names will be unique after importing to imc FAMOS.

Options and Sequences

There are two ways to specify options:

- Options for the format's filter or options for the format's derivative data filters
- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

Example:

```
idFile = FileOpenFAS(FileName , "#Instron.dll|Instron Format|AddSigTypeToName=0|SubstInvalidChar=1|Channels=Channel1,Channel20",0)
```

Option	Definition
SubstInvalidChar	0 - Name is adopted without any checking 1 - Replace any unallowed characters with '_'
AddSigTypeToName	0 - only channel name 1 - channel name with signal type
Channels	List with channel names separated by commas: <name1>,<name2> Important note: The channel names are stated as they are named in the data set. The option "SubstInvalidChar=1" is only run after the channel is located in the file. Both options remain independent in this way.

Example of multiple channels

1st step: Creating the basis options

```
var_Options = "#Instron.dll|Instron Format|AddSigTypeToName=0|SubstInvalidChar=1"
```

2nd step: Appending the individual channels

```
var_Options = var_Options + "|Channels="
```

```
var_Options = var_Options + "<name1>"
```

```
var_Options = var_Options + ", <name2>"
```

```
...
```

```
var_Options = var_Options + ", <nameN>"
```

3. Calling the import function

```
idFile = FileOpenFAS(FileName ,var_Options,0)
```

1.42 IRIG106

Filter

IRIG 106 Chapter 10

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Irig Chapter 10 Format	Irig.dll, Irig.409	10.0.0.0	*.ch10, *.c10

Import:

- Analog channels into simple waveforms
- Message data into TSA channel
- Support of import in multithread mode

1.43 K&S Elektronik

Filter

K&S Elektronik

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
K&S Electronics Wavework	KS_Electronics_Wavework.fas	*.txt

Import:

- (X,Y)-data into FAMOS XY-data sets

1.44 Kipp & Zonen

Filter

Kipp & Zonen

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Kipp & Zonen BD300	KippZonen_BD3000.FAS	*.rec

1.45 Knorr SD

Filter

SD-Knorr

Type

Import

Description

Filter name	Filename of filter	Version	Import File Extension
SD-Knorr	SDKnorr.dll, SDKnorr.409	10.0.0.0	*.sd

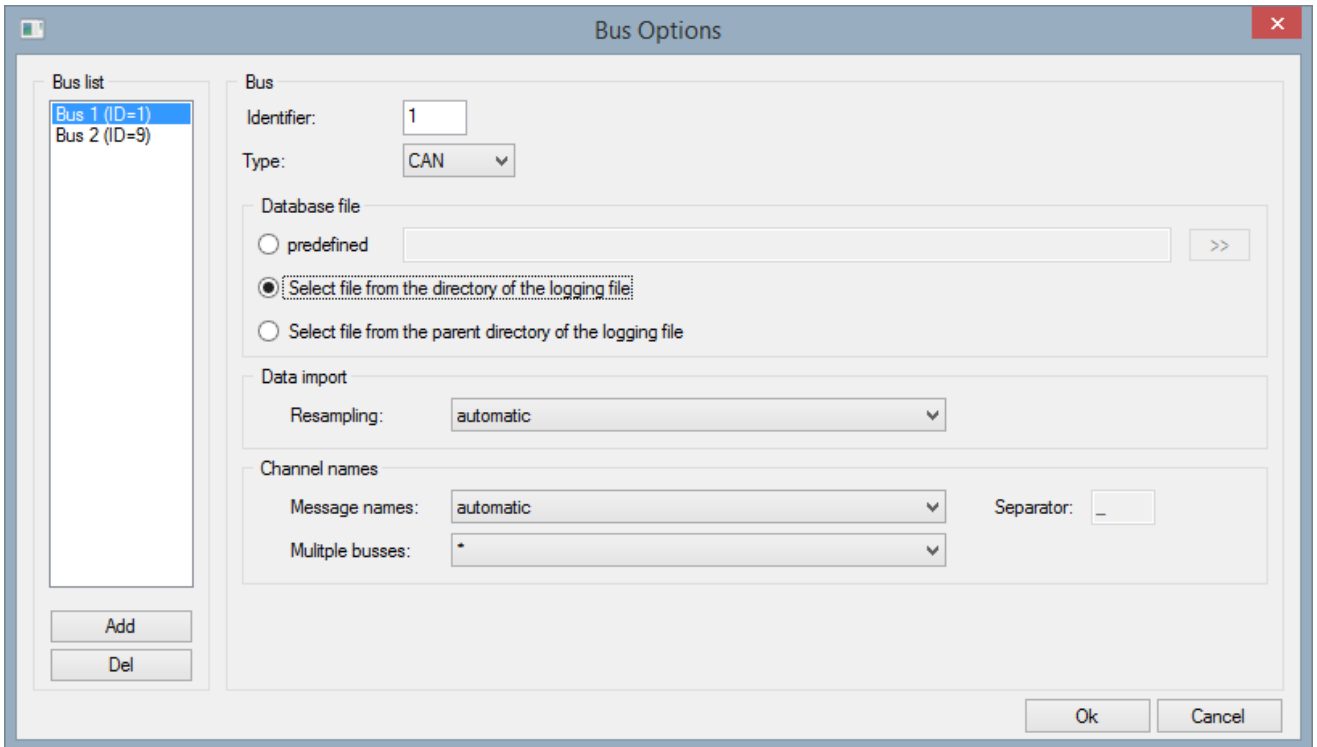
Import:

- XY data sets
- Normal Waveforms
- Support of import in multithread mode

General functioning

The Filter serves the purpose of extracting values from *.sd logging files containing binary data sets (frames) along with associated time stamps. On the basis of message- and channel descriptions, data are extracted from the logged data sets and assigned to FAMOS channels.

Options



Bus

Identifier: The bus identifier of the option set. The identifier begins at "1". The identifier of the bus has to be unique. The editor automatically checks and corrects the user input.

Type: The type of the bus, only "CAN".

Add: An option set for a bus is appended. The assigned identifier of the bus is the next number, that follows the greatest identifier.

Del: The selected option set will be deleted.

Database file

predefined

The basis file to use is specified directly. Specifying the relative pathname ("..\testfile.dbc") is permitted. The relative pathname refers to the data file currently open. The specified basis file is used for all logfiles.

A "*" instead of the filename ("..*.dbc") enters the name of the logfile at the position of the "*".

Select file from the directory of the logging file

An "Open File" dialog is displayed. The default folder is the folder of the logfile which is open.

Select file from parent directory of the logging file

An "Open File" dialog is displayed. The default folder is the folder one level above the folder of the logfile which is open.

Data Import

Resampling: "automatic"

If the control data contain specifications of the channel's or the frame's interval, then the channel is resampled according to this specified interval.

In FAMOS, a simple data set is generated with specifications of the interval and offset.

If there is no specified interval, an X/Y data set is created with an X-component as the time specification.

Resampling: "Time-stamp"

The data in the channels are supplied with time stamps.

In FAMOS an X/Y data set is created with an X-component as the time specification.

Channel names

Message names

Option	Kanalname
automatic	channel name as from the database
as channel prefix if identical channel names	If equal names, the message name is preceded the channel name.
as channel suffix if identical channel names	If equal names, the message name is appended to the channel name.
always as channel prefix	The message name is always preceded the channel name.
always as channel suffix	The message name is always appended to the channel name.

Multiple busses

Option	Channel name
*	Channel name only (without bud information)
BUS#_*	The channel name is composed with: "BUS" + identifier + "_" + channel name
*_BUS#	The channel name is composed with: Channel name + "_BUS" + identifier

Options and Sequences

There are two ways to specify options:

- Options for the format's filter or options for the format's derivative data filters
- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

List of options:

Option	Definition
CAN<id>_DBLOAD=<0 1 2>	Option: Database 0: predefined, only to use in combination with option "CAN<id>_DB" 1: Select file from the directory of the logging file 2: Select file from parent directory of the logging file
CAN<id>_DB=<filename>	Option: Name of database
CAN<id>_TS=<0 1>	Option: Resampling 0: automatic 1: Time-stamp
CAN<id>_MSGNAME=<0 1 2 3 4>	Option: Message names 0: automatic 1: as channel prefix if identical channel names 2: as channel suffix if identical channel names 3: always as channel prefix 4: always as channel suffix
CAN<id>_SEP	Option: Separator
CAN<id>_BUSNAME=<0 1 2>	Option: Multiple busses 0: "*" 1: BUS#_* 2: *_BUS#

Example:

; the name of the logging file

```
var_CurrFilePath = "D:\Examples\<<filename>.SD"
```

; standard configuration for selecting the import filter

```
var_FormatOptions = "#SDKnorr.dll|SD-Knorr"
```

; option for loading the database: "automatic"

; option for the bus name: "BUS#_*"

```
var_FormatOptionsBus1 = "|CAN1_DBLOAD=0|CAN1_DB=.\<database>.DBC|CAN1_BUSNAME=1"
```

```
var_FormatOptionsBus2 = "|CAN9_DBLOAD=0|CAN9_DB=.\<datavase2>.DBC|CAN9_BUSNAME=1"
```

; the options are composed

```
var_FormatOptions = var_FormatOptions + var_FormatOptionsBus1
```

```
var_FormatOptions = var_FormatOptions + var_FormatOptionsBus2
```

; the data will be imported

```
var_fileId = FileOpenFAS(var_CurrFilePath ,var_FormatOptions, 0)
```

1.46 LabVIEW

Filter

LabVIEW

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
LabVIEW ASCII	LabVIEW_ASCII.FAS	*.asc

Import:

- equidistant data into FAMOS Normal Waveforms

1.47 LeCroy Oscilloscopes

Filter

LeCroy Oscilloscopes

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
LeCroy File Formats	LeCroy_FileFormats.FAS	*.* (z.B. *.wav, *.w11)
LeCroy CSV	LeCroy_Csv.FAS	*.csv

Import:

- File Format: 1.0, 1.1, 2.1, 2.2, 2.3, XMLdig
- equidistant data into FAMOS Normal Waveforms

1.48 LEM

Filter

LEM

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension	Device Types
Lem Topas 1000*	Lem_Topas_1000.FAS	*.*	Topas 1000
Lem Topas 1000tri	Lem_Topas_1000tri.FAS	*.tri + *.trg	Topas 1000

Import:

- equidistant data into FAMOS Normal Waveforms

1.49 LMS

Filter

LMS

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
LMS Skalar20	LMS_Skalar20.FAS	*.tdf

Import:

- equidistant data into FAMOS Normal Waveforms

1.50 Matlab

Filter

Matlab 4.0 / 5.x / 6.x / 7 / 7.3

Type

Import/Export Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Matlab 5 File Format 4-7 + 7.3 without options	MatlabImportExport.dll	10.0.0.0	*.mat
MATLAB 4-7 + 7.3 with options	MatlabImportExport.409		

As of Version 7.3, the MATLAB internal file format has been converted to the HDF5 format. In consequence there are new possibilities for exchanging data between MATLAB and FAMOS, which is also reflected in an extensive range of options. For this reason, a new filter called "MATLAB" has been created in order to make this expansion evident. Whenever the import of a MATLAB file requires import options from 7.3, use the "MATLAB" filter. The formats 4-7 are imported in the identical way as the "MATLAB 5 File Format" filter.

For the purpose of creating new sequences, the exclusive use of the "MATLAB" filter is recommended.

Note:

To ensure compatibility with existing sequences, the import filter "MATLAB 5 File Format" also supports Format 7.3. Options are not available.

Import "MATLAB"

- MATLAB 4-7 supported like filter "MATLAB 5 File Format"

- MATLAB 7.3 identical to HDF5 import filter, see "[HDF5](#)"

Import "MATLAB 5 File Format":

- Matlab 4-7 supported
- File formats V6.x / 7.0 are identical to Matlab 5, compression in Matlab 7 is supported
- Special data types
 - MATLAB Array
 - 6-Bit Unicode (Unicode characters mapped to ASCII), 8-bit Unicode, 32-bit Unicode
- Supported Array types:
 - Numeric Array (also complex (R/I), 2-dimensional (MATLAB 4.0 only))
 - Structure Array (only one-dimensional)
 - Cell Array (only one-dimensional)
 - Character Arrays (1+2 dimensional (MATLAB 4.0 only))
- Conversion table from MatLab data structures into FAMOS data types
- MATLAB 7.3 supported (internally HDF5)
- Support of import in multithread mode

MatLab Data Structure	FAMOS Data Type
Numeric Array - one-dimensional	Normal Waveform (equidistant)
Numeric Array - multi-dimensional	Normal Waveform (equidistant, segmented)
Numeric Array - one value (1x1)	Single Value
Numeric Array - R/I data	Complex data set (Real-/Imaginary-parts)
Character Array	Text
Character Array - two-dimensional	Group of texts
Structure Array - one-dimensional	Channel group with normal waveforms (equidistant) Elements of the array are numbered from 0001 - xxxx
Cell Array - one-dimensional	Elements of the array are numbered from 0001 - xxxx As cells, numeric arrays, character arrays as well as single-element structure and cell arrays are permitted. Numeric Arrays as channel get the elements name. If the cell contains a structure, then a group is created for the cell and the structure's fields are entered into the group as channels/texts.

- Data types not supported
 - 64-bit, signed, unsigned

- Array types not supported
 - Sparse Array
 - Cell Array (multi-dimensional)
 - Structure Array (multi-dimensional)
 - Object Array
 - Character Array (multi-dimensional)
- Channel names with suffixes

It is possible to define matlab variables contains values for properties for channels imported. The content of the special variable is assigned to a specific property of the channel.

It is able to assigned via the suffix:

- <channel name>_xdelta for the sample time (X-Delta),
- <channel name>_xoffset for the sample time offset (X0),
- <channel name>_xunit for the unit of the X-Axis (X-Unit),
- <channel name>_yunit for the unit of the Y-Axis (Y-Unit),
- <channel name>_date for the creation time (Created on).

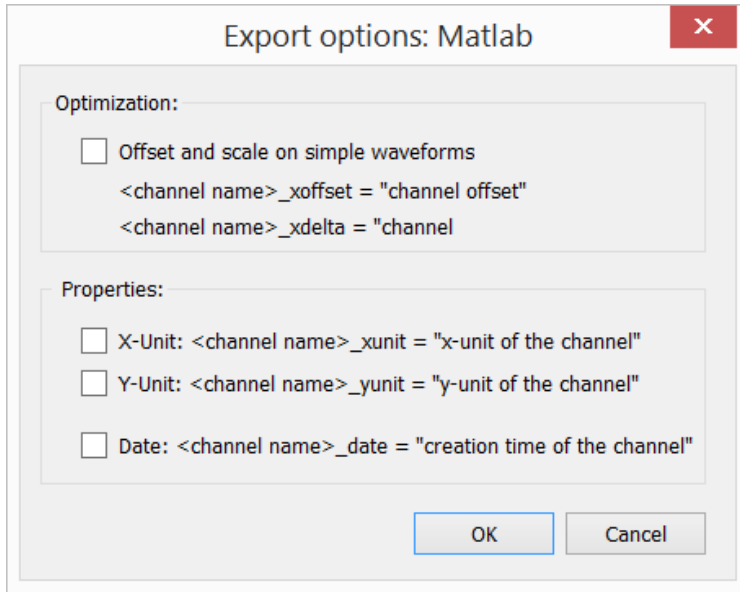
These special variable are **not** imported.

Note:

- Support of file compression (zlib)

Export "MATLAB 5 Format:

- no export of event-based FAMOS waveforms
- no export of FAMOS text variables
- Time variable exported as an additional channel
- Channel properties can optionally be exported to the MATLAB-file by means of special variables. A suffix is appended to each property.
- Support of export in multithread mode



Note:

If following installation of the import filter there is no import filter available in FAMOS for selection, the system is probably trying to access an incompatible version of the file zlib1.dll (installed with newer versions of MatLab.)

Remedy: Delete all additional versions of zlib1.dll from the computer. The version included should be adequate for MatLab.

Export "MATLAB 7, 7.3 format: see "[HDF5](#)"

1.51 MDF

Filter

MDF 3.x

Type

Import/Export Filter

Important

The Bus Filter needs the FAMOS 6.1 installation.

Description

Filter name	Filename of filter	Version	Import File Extension
MDF Format	mdf.dll	10.0.0.0	*.mdf

Import:

- FAMOS data types:

- Normal Waveforms
- XY-data sets
- Text variables
- File comments as text variable "FileComment"

- The following Version 3.2, 3.3 contents are not supported:
 - Trigger blocks "TR", assigned to data blocks "DR"
 - "CE" Channel-Extension blocks which are assigned to the channels
 - "CD" Channel-Dependency blocks which are assigned to channels

V1.2.1.0

- additional options
- import of Textchannels to FAMOS time stamp ascii data type

V10.0.0.0

- Support of import in multithread mode

Export:

- export to the formats 2.0 and 3.0.
- normal, XY-, complex data sets
- groups are dissolved
- specifications of time for the MDF File-Header are derived from the 1st channel.

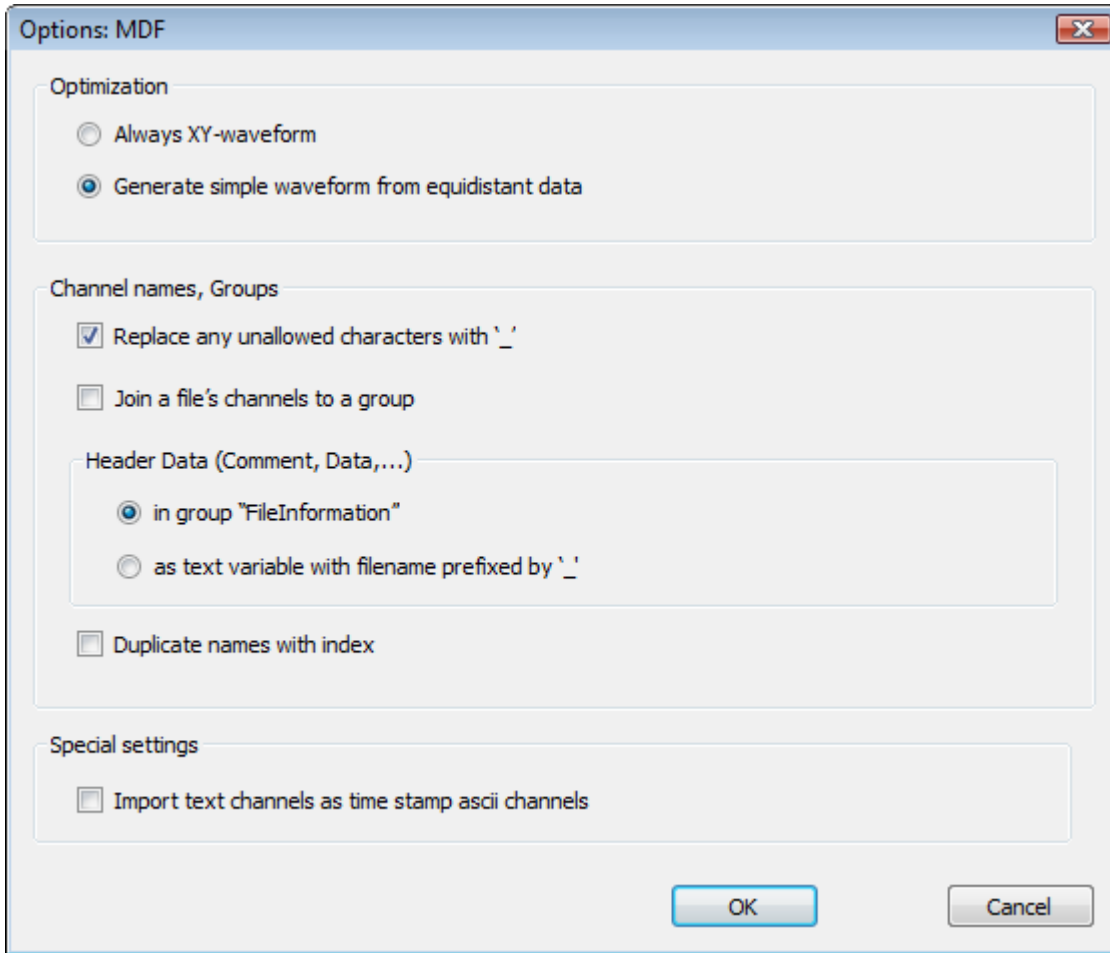
V1.0.3.0

- long channel names (Extension text box) supported

V10.0.0.0

- Support of export in multithread mode

Options



Optimization

Always XY-waveform:

Always generates an X/Y-waveform

Generate simple waveform from equidistant data

When this option is used, the system investigates the X/Y-waveform to see if the X-values are equidistant. If they are adequately equidistant, a "simple waveform" is generated. Simple waveforms are easier to process in sequences. This option is highly suited to situations where the measurement procedure is equidistant but the MDF format causes the data to take the X/Y-waveform form.

Channel names, Groups

Replace any unallowed characters with '_'

If the channel name contains any characters which are not permitted, such as "+, -, ...", these characters are replaced by '_'.

Join a file's channels to a group

This option generates a group having the name of the file imported as its group name. All channels are imported to this group.

Header Data (Comment, Data, ...)

in group "FileInformation"

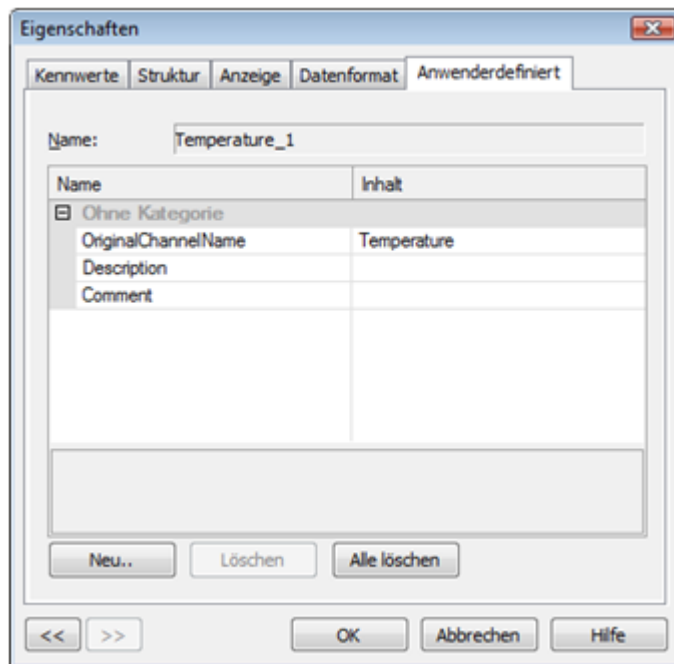
The header information listed at the top is all saved as a text variable in the group "FileInformation".

as text variable with filename prefixed by '_'

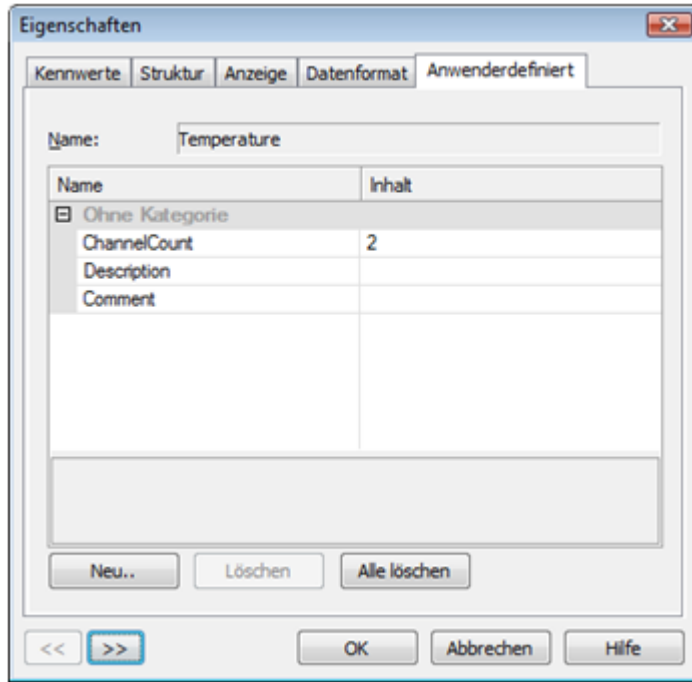
The header information listed at the top is all saved as a text variable, consisting of the pre-defined name (Date, Tester, Division) and the filename, in the same level as the channels.

Duplicate names with index

In MDF data sets, there can be channels having the same name but located in different data groups. In order to make the names of these channels more unique, it is possible to assign an index to them.



For purposes of unique identification of these channels, the user-defined property "OriginalChannelName" is set up among the channel's properties, and its value is the channel's original name.



The channel to first be imported from the data set is assigned the user-defined property "ChannelCount", whose value represents the amount of channels which have this name.

By querying the existence and content of these user-defined properties, it is possible to identify and edit channels originally having the same name.

Parameter of FileOpenFAS Function (import only):

All options are also available as call parameters in sequences.

Option	Definition
AlwaysXY=	0: Generate simple waveform from equidistant data 1: Always XY-waveform
SubstInvalidChar=	0: Name is adopted without any checking 1: Replace any unallowed characters with '_'
FileAsGroup=	0: Do not join a file's channels to a group 1: Join a file's channels to a group
HeaderInfosWithFilePrefix=	0: in group "FileInformation" 1: as text variable with filename prefixed by '_'
UseChannelNameIndex=	0: Duplicate channel names within a group are allowed, otherwise overwrite 1: Duplicate channel names with index
TextChannelAsTsa=	0: import as text variable 1: import as TSA channel

Example:

```
file = FileOpenFAS(FileName , "#MDF.dll|MDF Format|AlwaysXY=1|UseChannelNameIndex=1|
FileAsGroup=0",0)
file = FileLoad(FileName , "#MDF.dll|MDF Format|AlwaysXY=1|UseChannelNameIndex=1|FileAsGroup=0",0)
```

There are two ways to specify options:

- Options for the format's filter or options for the format's derivative data filters
- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

Import von zusätzlichen Information:

A mdf-file can contain additional information not assigned to any channel directly. These data are adopted into the group "FileComments" as text variables and are available for further processing.

The text variables of the comment group are generated from pre-defined boxes in the header block (hd_xxxx) and from freely definable boxes ("FIELD") belonging to the file comment.

Fields of the header block:

Blockvariable	FAMOS Textvariablen in der Gruppe FileComments	Inhalt (Beispiel)
hd_date	"Date"	"01.01.2011"
hd_time	"Time"	"08:00:00"
hd_author	"Tester"	"Max Mustermann"
hd_division	"Division"	"Tests"
hd_project	"Project"	"Measurement"
hd_subject	"Subject"	"Device"

Fields of the file comment:

The file comment's boxes are not defined in the MDF specifications; their syntax is as follows::

\$FIELD:"<name1>","<Inhalt1>"\$FIELD:"<name2>","<Inhalt2>"\$

By means of these boxes, it is possible to import any arbitrary data into the FileComments" group.

Example 1:

```
$FIELD:"<TestModule1>","<GS500>"$FIELD:"<TestModule2>","<GS502>"$
```

Variables of text in the group "FileComments"	Content
"TestModule1"	"GS500"
"TestModule2"	"GS502"

The file comment itself may consist of a comment and the additional fields ("FIELD").

```
<My comment> $FIELD:"<name1>","<Content1>"$FIELD:"<name2>","<Content2>"$
```

Example 2:

```
Test of Moduls GS500 and GS502$FIELD:"<TestModule1>","<GS500>"$FIELD:"<TestModule2>","<GS502>"$
```

Variables of text in the group "FileComments"	Content
"Comment"	"Test of Moduls GS500 and GS502"
"TestModule1"	"GS500"
"TestModule2"	"GS502"

1.52 MDF4

Filter

MDF4

Type

Import/Export Filter

Important

The Bus Filter needs the FAMOS 6.3 installation.

Description

Filter name	Filename of filter	Version	Import File Extension
-------------	--------------------	---------	-----------------------

MDF4 Format	AsamMdf.dll, AsamMdf.409	10.0.0.0	*.mf4, *.mdf
-------------	--------------------------	----------	--------------

Import:

- FAMOS data types:
 - Normal Waveforms
 - XY-data sets
 - Text variables
 - Event-based Waveforms
 - Text channels as time stamp ascii (TSA) channels

V 5.5.0.0

- Import of group- and channel properties as properties for support of the tree diagram structure of the FAMOS Variables list
- Optional use of group- and channel properties for forming unique channel names

V 10.0.0.0

- Support of import in multithread mode

Options

Options: MDF4 X

Optimization

Always XY-waveform

Generate simple waveform from equidistant data

Always generate events

Only generate events if there is more than 1 Event

Channel names

Replace any unallowed characters with ` _ `

Fixed:

Free:

↑

→ ←

↓

%ChannelSourceBusType%
 %ChannelSourceName%
 %ChannelSourcePath%
 %ChannelSourceType%
 %GroupAcquisitionName%
 %GroupAcquisitionSourceBusType%
 %GroupAcquisitionSourceName%
 %GroupAcquisitionSourcePath%
 %GroupAcquisitionSourceType%

Separator:

Special settings

Import text channels as time stamp ascii channels

Optimization

Always XY-waveform:

Always generates an X/Y-waveform

Generate simple waveform from equidistant data

When this option is used, the system investigates the X/Y-waveform to see if the X-values are equidistant. If they are adequately equidistant, a "simple waveform" is generated. Simple waveforms are easier to process in sequences. This option is highly suited to situations where the measurement procedure is equidistant but the MDF format causes the data to take the X/Y-waveform form.

Channel names

Replace any unallowed characters with '_'

If the channel name contains any characters which are not permitted, such as "+, -, ...", these characters are replaced by '_'.

Name formation

A channel's name formed in either of the ways

- "fixed", from a pre-defined list of various constructions, or
- "free", from the stock of group- and channel properties in any arbitrary order

The following assignment of "placeholders" is made:

"Fixed"	"Free"	MDF4
Channel name only	%ChannelName%	Name of channel from channel block (CNBLOCK)
Source name	%ChannelSourceName%	Name of source from source information block field " si_tx_name " from the Source Information block (SIBLOCK) of the channel block (CNBLOCK)
Source path	%ChannelSourcePath%	Name of source path from source information block field " si_tx_path " from the Source Information block (SIBLOCK) of the channel block (CNBLOCK)
-	%ChannelSourceType%	Source type from the field " si_type " from the Source Information block (SIBLOCK) of the channel block (CNBLOCK)
-	%ChannelSourceBusType%	Bus type from the field " si_bus_type " from the Source Information block (SIBLOCK) of the channel block (CNBLOCK)
Group name	%GroupAcquisitionName%	Acquisition name of the channel group from the field " cg_tx_acq_name " of the channel group block (CGBLOCK)
-	%GroupAcquisitionSourceName%	Acquisition name from the field " si_tx_name " from the Source Information block (SIBLOCK) of the channel group block (CGBLOCK)
Source path	%GroupAcquisitionSourcePath%	Acquisition path from the field " si_tx_path " from the Source Information block (SIBLOCK) of the channel group block (CGBLOCK)
-	%GroupAcquisitionSourceType%	Source type from the field " si_type " from the Source Information block (SIBLOCK) of the channel group block
-	%GroupAcquisitionSourceBusType%	Bus type from the field " si_bus_type " from the Source Information block (SIBLOCK) of the channel group block

- **"fixed"** name formation

For the name formation style **"fixed"**, the following constructions are available:

- Channel name only
- Channel name + source name
- Channel name + source name + source path
- Group name + channel name
- Group name + channel name + source name
- Group name + channel name + source name + source path

The "source path" is derived from the channel first, then the group.

- **"free"** name formation

The name formation style **"free"** enables freestyle extension/supplementation of the channel name. Extending the channel name may, on one hand, serve the purpose of generating a unique name, or on the other hand it could be used to form groups in the FAMOS Variables window (as of FAMOS V7.3). The separator character enables the individual components to be distinguished.

Since the specification made applies uniformly to all of the channels, it may happen that not all channels receive the necessary data. In such cases, the placeholder is replaced with "---".

Remarks:

The terms CNBLOCK, SIBLOCK, CGBLOCK are data structures belonging to the MDF4 format and are described in the documentation "ASAM MDF VERSION 4.0.0". It is recommended to use the tool "Vector MDF Validator", in order to inspect the individual blocks.

Special settings

Text channels are imported as Time Stamp Ascii channels.

Parameter of FileOpenFAS Function (import only):

All options are also available as call parameters in sequences.

Option	Definition
AlwaysXY=	0: Generate simple waveform from equidistant data 1: Always XY-waveform
NotAlwaysEvents=	0: Generate event-based waveform always 1: Generate event-based waveform if more than one events occurs
SubstInvalidChar=	0: Name is adopted without any checking

	1: Replace any unallowed characters with ' _ '
ChannelNames=	0: Channel name only 1: Channel name + source name 2: Channel name + source name + path name 3: Group name + channel name 4: Group name + channel name + source name 5: Group name + channel name + source name + path name
TextChannelAsTsa=	0: Text channel as text variable (max. 255 Char.) 1: Import text channel as TSA channel
NameCreation=	0: "Fixed" 1: "Free"
NameSeparator=	1 Character
NameComposition=	name placeholders with separator character

Example:

```
file = FileOpenFAS(FileName , "#AsamMdf.dll |MDF4 Format | ChannelNames=3 | AlwaysXY=1", 0)
file = FileOpenFAS(FileName , "#AsamMdf.dll |MDF4 Format | NameCreation=1 | NameSeparator=\\ |
NameComposition=%GroupAcquisitionSourceType%\%ChannelName%", 0)
```

There are two ways to specify options:

- Options for the format's filter or options for the format's derivative data filters
- Options in call parameters

The options specified as call parameters overwrite the options belonging to the Options dialogs. If no option is specified, then the pre-defined option is valid. Options specified as call parameters are not saved.

Export:

- FAMOS data types:
 - Normal Waveforms
 - XY-data sets
 - Complex data sets
 - Event-based Waveforms
 - Waveforms with segments
 - TSA channels as byte array
 - Text variable as byte array with cycle counter of 1
- Support of export in multithread mode

1.53 MKT CAN Logger

Filter

MKT CAN Logger

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
MKT CAN Logger Data File	MKTImport.dll	10.0.0.0	*.cld

Import:

- Decoding of the signals which are described in the CLD file.
- The data are imported as FAMOS XY-data sets, since the file contains no specified sampling interval.
- GPS data are imported.
- References to DBC files are ignored.
- Support of import in multithread mode

1.54 MTS System Corporation

Filter

MTS System Corporation

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
MTS System Corporation Series 793 Software	MTS_Series793_Software.FAS	*.dat

1.55 Nicolet

Filter

Nicolet

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Nicolet Team	Nicolet_TEAM.FAS	*.*
Nicolet Frequency Domain	Nicolet_FreqDomain.FAS	*.wff
NICOLET	integrated	*.wft

Import:

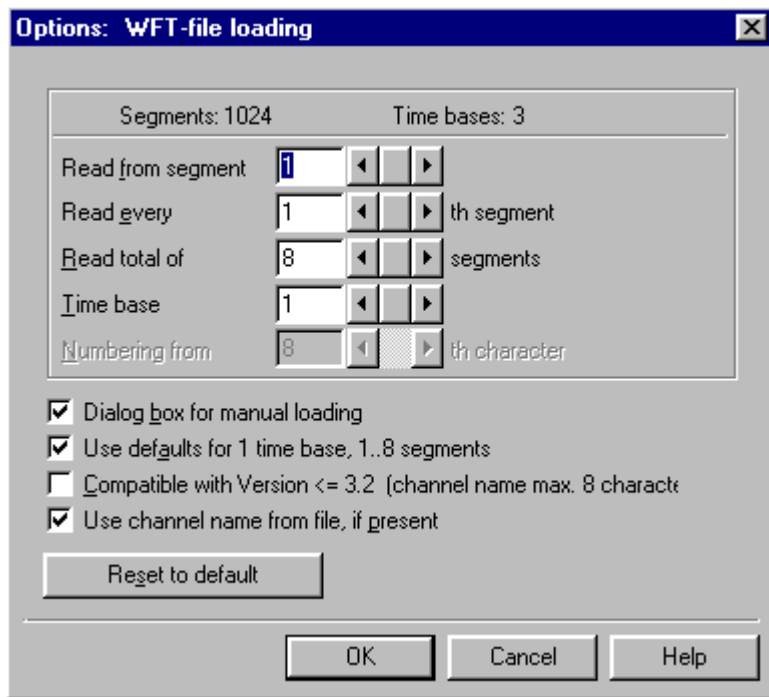
- equidistant data into FAMOS Normal Waveforms, Nicolet Team
- data in complex FAMOS data sets (R,I), Nicolet Frequency Domain
- equidistant data into FAMOS Normal Waveforms

Options

NICOLET (*.wft) Format

Select "Nicolet" in the dialog "Load file" to read files in WFT format (defined by the firm **Nicolet Instrument Corporation**).

When the button <Options> is selected, the following dialog appears for specification of the file section to be read



Dialog for manual loading

When this option is selected, the dialog appears when a file is loaded above the FAMOS menu, so that options for the present file may be selected individually. Please note that this option is canceled when the option "Default for 1 time base..." is selected.

Please note further that the dialog box automatically appears when a file cannot be loaded with the specified options. The option "Dialog for manual loading" may be inactive.

Default for one time base, 1..8 segments

When this option is selected, this dialog never appears when WFT files with exactly one time base and maximum 8 segments are loaded and all segments contained are loaded every time. This option should be selected for convenient loading of simple WFT files. Do **not** select this option if the "Dialog for manual loading" is active and the dialog box should always appear!

Reset to default

Select this button to shift the dialog to a standard state, from which changes can be made.

Compatible with FAMOS 3.2

This option ensures that variables opened are named in a manner compatible with earlier FAMOS versions. In such versions, the resulting variable name was limited to 8 characters and was always derived from the filename.

Use channel name from file

If a channel name is specified in the file, this option causes it to be used. The option can only be selected if the option "**Compatible with FAMOS 3.2**" is deactivated. Otherwise, the variable name is derived from the filename.

Time base

Specify the time base to be read in the text box "**Time base**". Values of 1 to 3 are permitted, depending on the file.

Numbering from...

In this box you can specify the position within a FAMOS variable name, at which an index number is to be inserted. The numbering pertains to different segments. This option is only enabled if the option **"Compatible with FAMOS 3.2"** is also enabled. Otherwise, numbering may be appended to the filename in the form of : "_Seg#".

Example

The following examples, which illustrate "FAMOS 3.2-compatible numbering", use the file names "WAVE" and "WAVEFORM".

Rules for numbering

Variables are numbered with as many places as necessary to register all segments in the file. For example, if "WAVE" contains 5 segments, they are numbered with "WAVE1"..."WAVE5"; if it contains 150 segments, they are numbered "WAVE001".."WAVE150".

If the file name is short enough, numbering is simply appended. A file name may have a maximum of 8 characters. For example, if the file name has 6 characters and contains less than 100 segments, the file is numbered by appending numbers to the name.

If the file name is too long to append numbering, the user can enter the position from which numbering should start. A file called "WAVEFORM" set for numbering from the second position, a waveform "W01EFORM" or "W100FORM" could result.

If too large a position is specified for numbering, that position is automatically placed at the front. For example, in numbering from the 8th position and a file "WAVEFORM" with 1000 segments, the name "WAVE0001"..."WAVE1000" is selected.

No numbering is assigned to a file with only one segment.

Selecting segments

Additional text boxes allow selection of segments. Selection is often necessary due to memory, clarity and loading duration. The user determines from which segment loading should start, which segment should be loaded and how many segments should be loaded in total. For example, if a file contains 80 segments and you read in

from segment 10

each 5th segment

total of 4th segments,

the following segments are read:

10, 15, 20, 25

The following variables are generated from a file "WAVE":

"WAVE10", "WAVE15", "WAVE20", "WAVE25"

The number of available segments is displayed in the dialog box.

A tone is generated when the dialog is ended and if input errors occurred. The input cursor is set to the corresponding text box.

If, during loading, invalid variable names are generated by numbering or file names themselves, or if some of the variables already exist, the program allows these names to be changed.

Reference

Chapter 'Sequence Commands (Reference)', command: [WFTLOAD](#)

1.56 nSoft-DAC

Filter

nSoft-DAC

Type

Import/Export Filter

Description

Filter name	Filename of filter	Version	Import File Extension
nSoft DAC Format	nSoft-DAC.dll	10.0.0.0	*.dac

Import:

- as FAMOS Normal Waveforms
- (X,Y)-pairs as XY-data sets
- Matrix data into a FAMOS segmented data set
- event-based data not supported
- Support of import in multithread mode

Export:

- Normal, XY- and Complex data sets as (X,Y)-pairs
- segmented data sets as Matrix
- event-based data sets not supported
- Support of export in multithread mode

Note:

- Depending on the format, export and re-import of complex data sets can cause XY-data sets to be converted. When this principle is carried out, it can result in graphics diagrams which do not make sense, since in an FFT an amplitude (X-data) generally is not plotted over its phase (Y-data).

1.57 Picture Format

Filter

Picture

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Picture Format	integrated	*.bmp, *.jpg, *.jpeg, *.gif, *.png, *.tif, *.tiff

Import:

- Import of image files that are saved as segmented data and a color flag. By means of this flag, the curve window recognizes that the variable is to be displayed as an image. Furthermore, any functions which work with RGB-values also interpret the data as color values.

1.58 Ono Sokki

Filter

Ono Sokki

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
OnoSokki Narrow-Band - ASCII	OnoSokki_Narrow_Band_ASCII.FAS	*.prn
OnoSokki 1/3-Octave - ASCII	OnoSokki_Thirds_ASCII.FAS	*.prn

Import:

- Spectra into FAMOS XY-data sets

1.59 OROS

Filter

OROS

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
OROS Wave	Oros_Wav.FAS	*.wav
OROS EA2	Oros_EA2.FAS	*.ae2

Import:

- equidistant data into FAMOS Normal Waveforms
- text variables (OROS EA2)

1.60 Remes DIGISKOP

Filter

Remes DIGISKOP(c)

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
DIGISCOP	integrated	*.drd

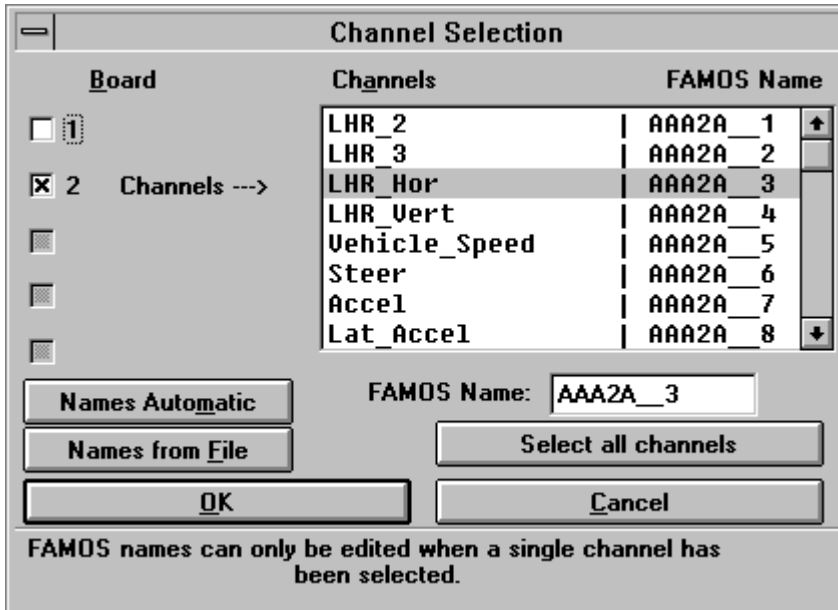
Import

- equidistant data into FAMOS Normal Waveforms

Options

Select the entry "DIGISKOP" in the dialog "Load File" to read files in DIGISKOP(c) format (defined by the company **remes**).

Once the desired files have been selected and confirmed with <OK>, the following dialog appears for each file. Selection of channels is determined with this dialog.



At the left of the dialog, individual boards or all boards contained in the file group may be selected. Channels contained in the corresponding boards may be selected in the box on the right side of the dialog (the corresponding board is marked by an arrow). Channel names from the file are listed on the left; on the right are the names under which the channel will be loaded in FAMOS.

The FAMOS names can be determined by converting automatically into numbered variable names by selecting the button "Names Automatic", by selecting the button "Names from File" with the labels contained in the file, or by selecting the channels individually in the input box "FAMOS Name".

Remarks

Please note that each variable name in FAMOS must be unique. When several channels are to be read in, it is recommended to use automatic selection of names. Otherwise, FAMOS may overwrite already existing variables with the same name.

Reference

Chapter 'Sequence Commands (Reference)', command: [DSPLOAD](#)

1.61 RPC-3

Filter

RPC-3

Type

Import/Export Filter

Description

Filter name	Filename of filter	Version	Import File Extension
RPC-3 Format	RPC3Filter.dll	10.0.0.0	*.rsp, *.txt

Import:

- RPC3-time signals in binary format (Little- and Big-Endian) into FAMOS Normal Waveforms
- Data type: 16-bit Integer
- Formats and Types supported:
- Support of import in multithread mode
 - Data format
 - ASCII Format
 - Data type (FILE_TYPE according to Spec. MTS dev_002713.pdf)
 - HISTOGRAM
 - CONFIGURATION
 - MATRIX
 - FATIGUE
 - ROAD_SURFACE
 - SPECTRAL
 - START
- Support of import in multithread mode

Export:

- FAMOS Normal-, XY-data sets, dissolved FAMOS groups into RPC3-time signals
- The time resolution of the data sets must be identical
- Support of export in multithread mode

1.62 SigView

Filter

SigView

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
SigView Format	SigView.FAS	*.sig

1.63 Sirec

Filter

Sirec

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Sirec Thermocouples	Sirec_Thermocouples.FAS	*.rec

Import:

- equidistant data into FAMOS Normal Waveforms

1.64 SoMat

Filter

SoMat

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
SoMat Information File Format V1.1	SomatImport.dll	10.0.0.0	*.sif

Import:

- equidistantly sampled data (Time History/PV Slice) into FAMOS Normal Waveforms
- Modes
 - in RAM mode
 - in PCMCIA mode
 - in multiplex mode of eDAQ devices
- Support of import in multithread mode

Export:

- Normal Waveforms
- Support of export in multithread mode

Description

Filter name	Filename of filter	Version	Import File Extension
SoMat SIE V1.0	SomatImport.dll	10.0.0.0	*.sie

Import:

- into FAMOS Normal Waveforms
- Support of import in multithread mode

Important note:

- For import of data in the SIE-format, two additional libraries from the company HBM, "libapr-1.dll" and "libsie.dll", are required.
These are provided free of cost by HBM in the download area of their webpage under Somat as "SoMat libsie".
The downloaded archive must be unpacked and the files "libapr-1.dll" and "libsie.dll" copied to the folder "**c:\Program Files\imc\shared\Extensions**".
- Furthermore the **Visual C++ Redistributable** for **Visual Studio 2015** or **2017** is required. It will be installed with imc FAMOS version 7.4 or higher. For all previous versions of imc FAMOS these have to be installed additionally.

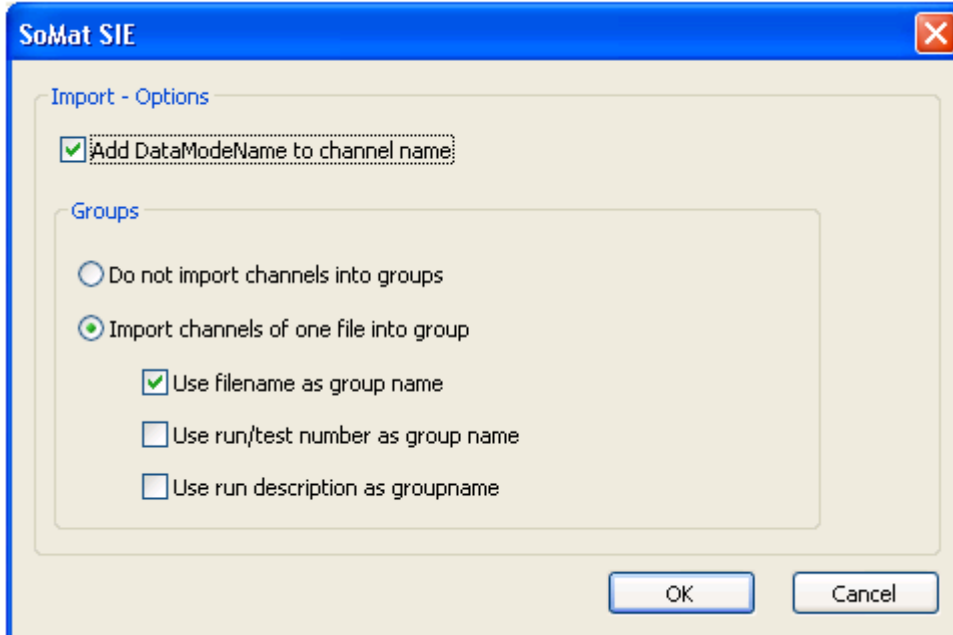
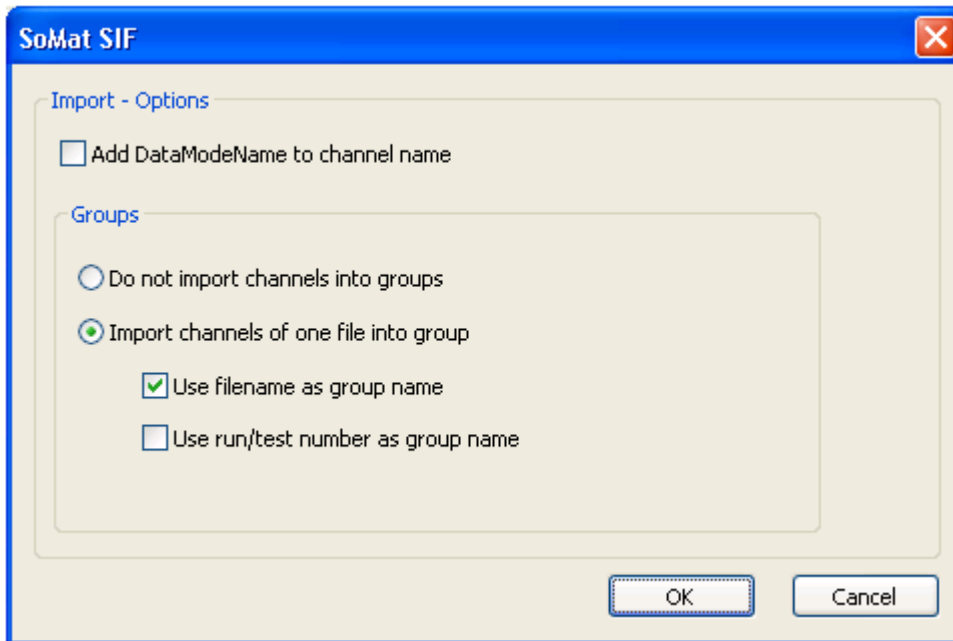
Options

The channel designators in SoMat have the following structure:

DataModeName@Channelname.Run/Testnummer

Example: Test@Channel1.RN_1

The options are used to set whether the DataModeName is to appear among the channel names in FAMOS, and whether the data sets are to be imported by group. For the latter case, the groups can be named by the filename, the Run/Test-number, or the test description (only SIE format).



1.65 Sony

Filter

Sony

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension	Device Types
Sony ASCII (DOS) Int	Sony_Int.fas	*.Log	SONY DAT-Recorder PC216AX

Import:

- equidistant data into FAMOS Normal Waveforms

1.66 Spider Measurement System

Filter

Spider Measurement System

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Spider Measurement System	Spider_Measurement.FAS	*.hdr + *.asc

1.67 Stabicontrol

Filter

Stabicontrol

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Stabicontrol SMF Format	Stabicontrol_SMF.FAS	*.smf

Import:

- equidistant data into FAMOS Normal Waveforms
 - (X,Y)- data pairs into FAMOS XY-data sets
-

1.68 Staehle

Filter

Staehle

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Stähle Robot	Staehle_Robot.FAS (V1.1)	*.dat

Import:

- equidistant data into FAMOS Normal Waveforms with "LOG_RATE" for time interval

1.69 Stiegele Datasystems

Filter

Stiegele Datasystems

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Stiegele Datasystems MicroEdition	Stiegele_MircoEdition.FAS	*.mdf

Import:

- equidistant data into FAMOS Normal Waveforms
-

1.70 TEAC

Filter

TEAC

Type

Import/Export Filter

Description

Filter name	Filename of filter	Version	Import File Extension
TEAC TAffmat Format	TAffmatImportExport.dll	10.0.0.0	*.hdr + *.dat

Import:

- data in 2-byte Signed Integer format or in 4-byte Floating Point format into FAMOS Normal Waveforms
- other data types not supported at this time
- the header (*.hdr) and data (*.dat) are expected in two separate files (having the same name)
- import of data files with embedded header information
- Data must be saved in Interlaced mode. (From each channel, first the first sample then the second, etc.)
- Support of import in multithread mode

Export:

- FAMOS Normal Waveforms into 4-byte Floating Point data or 2-byte Signed Integer format
- DAT files with embedded header information are exported.
- All units are exported in quotation marks in order to avoid problems with interpretation of units in DaDISP.
- Export for the TEDAM software is subject to the following constraints:
 - Time statements are rounded to whole seconds.
 - Export as FLOAT
 - No quotation marks with the units
 - Time channel written to the exported file
- The data to be exported must have the same sampling rate/ x-unit and start time.
- If the channels to be exported have different data types, export is always in 4-byte Floating Point format.
- Other data types are not supported at this time.
- Support of import in multithread mode

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension	Device Types
-------------	--------------------	-----------------------	--------------

TEAC DAT Recorder 135T	Teac_135t.FAS	*.t2p	TEAC DAT Recorder RD-135T
TEAC DAT Recorder 145T	Teac_145t.FAS	*.*	TEAC DAT Recorder RD-134T

Import:

- equidistant data into FAMOS Normal Waveforms

1.71 Tektronix

Filter

Tektronix

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension	Device Types
Tektronix TDSx000	Tektronix_TDSwfm000.FAS	*.wfm	TDS 5000, TDS 6000, TDS 7000
Tektronix TDSx000/B	Tektronix_TDSwfm5000B.FAS	*.wfm	TDS 5000B, TDS 6000B, TDS 7000B
Tektronix DPO,DSA70000	Tektronix_DPO_DSA_70000.FAS	*.wfm	DPO7000, DPO70000, DSA70000
Tektronix TDS 700A,++	Tektronix_TDS700A.FAS	*.wfm	TDS-700A TDS-500B:4.1e; TDS-600B:4.3e; TDS-640A:3.8.8e; TDS-400A:1.0.2e
Tektronix TDS 744	Tektronix_TDS744.FAS	*.wfm	TDS 744
Tektronix TDS 744A Mathcad	Tektronix_TDS744A_Mathcad.fas	*.dat	TDS744A
Tektronix DPO 3034 BSP	Tektronix_DPO_3034_BSP.fas	*.isf	DPO 3034 BSP

Import:

- equidistant data into FAMOS Normal Waveforms

1.72 Telsonic

Filter

Telsonic

Type

Import Filter

Description

Filter name	Filename of filter	Import File Extension
Telsonic USS-Data	Telsonic_USS_Data.FAS	*.asc

1.73 TurboLab

Filter

TurboLab

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
TurboLab DAFF	TurbolabImp.dll	2.0.0.0	*.dat

Import:

- Normal Waveforms

1.74 Universal File Format UFF58

Filter

Universal File Format UFF58

Type

Import/Export Filter

Description

Filter name	Filename of filter	Version	Import File Extension
Universal File Format	UffImportExport.dll	10.0.0.0	*.uff, *.unv, *.bunv

Import:

- import of binary or ASCII files in the UFF58 format
- import of both real and complex data (equidistant or XY-data)
- Support of import in multithread mode
 - binary or ASCII files into the UFF58 format.
 - Normal Waveforms and Complex data sets
 - segmented waveforms are supported
 - Function type,
 - Reference Direction,
 - Response Direction.
- Support of import in multithread mode

For the transfer or reference and response parameters to the file the channel properties can be used:

Name der Property	Example
FunctionType	Spectrum
ResponseEntityName	<RespName>
ResponseNode	1000
ResponseDirection	Resp+Xtran
ReferenceEntityName	<RefName>
ReferenceNode	2000
ReferenceDirection	Ref+Xtran

Examples:

```
UserPropSet(channel, "FunctionType", "Spectrum", 1, 2)
UserPropSet(channel, "ResponseEntityName", "Response1", 1, 2)
UserPropSet(channel, "ResponseNode", "1000", 1, 2)
UserPropSet(channel, "ResponseDirection", "Resp+Xtran", 1, 2)
UserPropSet(channel, "ReferenceEntityName", "Reference1", 1, 2)
UserPropSet(channel, "ReferenceNode", "2000", 1, 2)
UserPropSet(channel, "ReferenceDirection", "Ref+Xtran", 1, 2)
```

The function types:

Funktionstyp	Text value
0 - General or Unknown	General or Unknown
1 - Time Response	Time Response
2 - Auto Spectrum	Auto Spectrum
3 - Cross Spectrum	Cross Spectrum
4 - Frequency Response Function	Frequency Response Function
5 - Transmissibility	Transmissibility
6 - Coherence	Coherence
7 - Auto Correlation	Auto Correlation
8 - Cross Correlation	Cross Correlation
9 - Power Spectral Density (PSD)	Power Spectral Density oder PSD
10 - Energy Spectral Density (ESD)	Energy Spectral Density oder ESD
11 - Probability Density Function	Probability Density Function
12 - Spectrum	Spectrum
13 - Cumulative Frequency Distribution	Cumulative Frequency Distribution
14 - Peaks Valley	Peaks Valley
15 - Stress/Cycles	Stress
16 - Strain/Cycles	Strain
17 - Orbit	Orbit
18 - Mode Indicator Function	Mode Indicator Function
19 - Force Pattern	Force Pattern
20 - Partial Power	Partial Power
21 - Partial Coherence	Partial Coherence
22 - Eigenvalue	Eigenvalue
23 - Eigenvector	Eigenvector
24 - Shock Response Spectrum	Shock Response Spectrum
25 - Finite Impulse Response Filter	Finite Impulse Response Filter
26 - Multiple Coherence	Multiple Coherence
27 - Order Function	Order Function

Reference direction and response directions:

UFF code	Direction	Text value	
		Reference	Response
0	Scalar	(default)	(default)
1	+X Translation	Ref+Xtran	Resp+Xtran

-1	-X Translation	Ref-Xtran	Resp-Xtran
2	+Y Translation	Ref+Ytran	Resp+Ytran
-2	-Y Translation	Ref-Ytran	Resp-Ytran
3	+Z Translation	Ref+Ztran	Resp+Ztran
-3	-Z Translation	Ref-Ztran	Resp-Ztran
4	+X Rotation	Ref+Xrot	Resp+Xrot
-4	-X Rotation	Ref-Xrot	Resp-Xrot
5	+Y Rotation	Ref+Yrot	Resp+Yrot
-5	-Y Rotation	Ref-Yrot	Resp-Yrot
6	+Z Rotation	Ref+Zrot	Resp+Zrot
-6	-Z Rotation	Ref-Zrot	Resp-Zrot

1.75 Vestas INT

Filter

VestasINT

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension
VestasINT	VestasINT.dll, VestasINT.409	10.0.0.0	*.int

Import:

- Normal Waveforms
- Channel names are formrd from "Sensor_n". n : Number of sensor in the data record
- Support of import in multithread mode

1.76 Vispiron

Filter

Vispiron

Type

Filter Import

Description

Filter name	Filename of filter	Import File Extension
Vispiron ROTEC Transform	Vispiron_ROT_EC_Transform.FAS	* Zeit1.*,Mess1.*

1.77 WAV

Filter

WAV

Type

Import Filter

Description

Filter name	Filename of filter	Version	Import File Extension	Device Types
WAV	wav.dll	10.0.0.0	*.wav	AUDIO Devices RION-DA-20

Import:

- equidistant data into FAMOS Normal Waveforms
- Support of import in multithread mode

Supported contents of "RIFF" format (Resource Interchange File Format)

- WAVE (Wave Audio File Format)
 - + Formate (Tags)
 - PCM
 - IEEE FLOAT
 - + Device specific sections (Chunks)
 - 1) RION DA-20 ("rion")
 - Channel On/Off Flag
 - Comment of channel
 - Y-scalation
 - Y-units

1.78 Yokogawa

Filter

Yokogawa Measurement Devices

Type

Import Filter

Description

Filter name	Filename of filter	Version	Filename Extension	Device Types
Yokogawa DLO	YokogawaImport.dll	10.0.0.0	*.dlo	XL100

Yokogawa WDF	YokogawaImport.dll + DL850.dll, DL950.dll, DLM2000.dll, DLM4000.dll SL1000.dll FAMOS 32-Bit only DL750.dll DL1600.dll, DL1700.dll, DL9000.dll DLM6000.dll SL1400.dll	10.0.0.0	*.wdf	DL350, DL750, DL850, DL950, SL1000, SL1400 DL1600, DL1700, DL9000 DLM2000, DLM4000, DLM6000
Yokogawa GDS Yokogawa GEV	FAMOS 32-Bit only YokogawaImport.dll + DIIGXFileAccess.dll	10.0.0.0	*.gds, *.gev	GP/GX
Yokogawa MV Yokogawa DX	FAMOS 32-Bit only YokogawaImport.dll + DIIDXADVFileAccess.dll	10.0.0.0	*.dae, *.dad	MV/DX/1000/2000
Yokogawa UniFormat	Yokogawa_Universal.fas		*.hdr + *.wvf	DL708, DL1540, DL2700, DL4080

Import:

- Import of all channel types (Analog/ Logic / Pulse / Communication / Calculation) is supported.
- Mapping to Normal Waveforms; Logic-channels are mapped to digital bits
- Support of import in multithread mode

Index

A

ADF (import filter) 8
Agilent (import filter) 9
Alborn (import filter) 10
ASCII format (import filter) 11
ATFX (import/export filter) 14
Avonis (import filter) 10

B

B&S Multidata (import filter) 15
BINARY file (import filter) 16
binary format 16
 signed integers 16
 unsigned integers 16

C

Caesar Datasystems (import filter) 28
CAN Analyzer 85, 91
Catman (import/export filter) 51

D

DAFF (import filter) 113
DAS-3 (import filter) 8
DASyLab (import filter) 29
Dewetron 33
Dewetron (import filter) 30
DIAdem (import/export filter) 35
dialog:options
 XLS Import 42
DIGISKOP (import filter) 103
Digitek DataView (import filter) 40
DMD 33
DSP Development Corporation DADiSP (import filter) 40
DTS 41

E

ETAS/Bosch (import filter) 41
Excel file format (import filter) 42
export filter
 ATFX 14
 CAN 85
 Catman 51
 FlexPro 46
 Google Earth 48
 HBM 52
 Knorr SD 76
 Matlab 82
 MDF4 91
 NI 35

RPC-3 104
SD Knorr 76
TAFMat 111
TEAC 111
UFF58 113

F

FlexPro (import/export filter) 46
format
 ASCII 11
 BINARY 16
 XLS 42

G

G.i.N. (import filter) 47
gbd (import filter) 50
Google Earth (export filter) 48
Gould (import filter) 49
Grabau sound power(import filter) 49
Graphtec (import filter) 50

H

Haag (import filter) 50
HBM (import filter) 52
HBM (import/export filter) 51
HDF5 53
 Exportoptionen 56
 Importoptionen 54
HEAD acoustics (import filter) 60
HERF4 64
herf4, 64
HIOKI measurement devices (import filter) 65
Hi-Techniques (import filter) 66
Horiba 66
HYDAC 64
Hydrotechnik (import filter) 67

I

iba (import filter) 67
IEEE Standard Common Format (import filter) 68
imc BusFilter (import filter) 68
imc STUDIO Metadata (import filter) 72
import / export filter introduction 6
import filter
 ADF 8
 Agilent 9
 Alborn 10
 ATFX 14
 Avonis 10
 B&S Multidata 15
 Caesar Datasystems 28

- import filter
 - CAN 85
 - Catman 51
 - DAFF 113
 - DASYLab, 29
 - Dewetron 30
 - DIGISKOP 103
 - Digitek DataView 40
 - DSP 40
 - ETAS/Bosch 41
 - FlexPro 46
 - G.i.N. 47
 - gbd 50
 - Gould 49
 - Grabau sound power 49
 - Haag 50
 - HBM 52
 - HEAD acoustics 60
 - HIOKI 65
 - Hi-Techniques 66
 - Horiba 66
 - Hydrotechnik 67
 - iba 67
 - IEEE 68
 - imc BusFilter 68
 - imc STUDIO (Metadata) 72
 - Irig, Irig106 74
 - K&S Elektronik 75
 - Kipp & Zonen 75
 - Knorr SD 76
 - LabVIEW 80
 - LeCroy 80
 - LEM 81
 - LMS 81
 - Matlab 82
 - MDF4 91
 - MKT CAN Logger 97
 - MTS 97
 - NI 29, 35
 - Nicolet 98
 - nSoft-DAC 101
 - Ono Sokki 102
 - OROS 102
 - RPC-3 104
 - SD Knorr 76
 - SigView 105
 - Sirec 106
 - SoMat 106
 - Sony 108
 - Spider 109
 - Stabicontrol 109
 - Staeble 110
 - Stiegele Datasystems 110
 - TAFFmat 111
 - TEAC 111
 - Tektronix 112
 - TurboLab 113
 - UFF58 113
 - Vector 68
 - Vestas INT 117
 - Vispiron 117
 - Yokogawa 118
- INT Vestas (import filter) 117
- Irig, Irig106 (import filter) 74
- K**
 - K&S Elektronik (import filter) 75
 - Kipp & Zonen (import filter) 75
 - Kistler (import filter) 8
 - Knorr (import filter) 76
 - Knorr-Bremse (import filter) 76
- L**
 - LabVIEW (import filter) 80
 - LeCroy Oszilloskope (import filter) 80
 - LEM (import filter) 81
 - LMS (import filter) 81
- load file
 - XLS 42
- M**
 - Matlab (import/export filter) 82
 - MDF (import/export filter) 85
 - MDF4 (import/export filter) 91
 - Metadata imc STUDIO (import filter) 72
 - MKT CAN Logger (import filter) 97
 - MTS System Corporation (import filter) 97
- N**
 - National Instruments (import filter) 29, 80
 - National Instruments (import/export filter) 35
 - NI (import/export filter) 35
 - Nicolet (import filter) 98
 - nilibddc, nilibddc.dll 35
 - nSoft-DAC (import filter) 101
- O**
 - Ono Sokki (import filter) 102
- options File-Load
 - Binary 16
- options:file-load
 - ASCII 11
 - XLS 42

OROS (import filter) 102

P

picture format 101

PNRF (import/export filter) 52

R

RGB image 101

RPC-3 (import/export filter) 104

S

SD Knorr (import filter) 76

signed integers

 binary format 16

SigView (import filter) 105

Sirec (import filter) 106

SLICEWave 41

SoMat (import filter) 106

Sony (import filter) 108

Spider Measurement System (import filter) 109

Stabicontrol (import filter) 109

Staeble Robot (import filter) 110

STARS (import filter) 66

Stiegele Datasystems (import filter) 110

T

TAFFmat (import/export filter) 111

TDM, TDMS 35

TEAC (import/export filter) 111

Tektronix (import filter) 112

TurboLab (import filter) 113

two's complement 16

U

Universal File Format UFF58 (import/export filter) 113

unsigned integers

 binary format 16

V

Vector (import filter) 68, 91

Vector (import/export filter) 85

Vestas INT (import filter) 117

Vispiron (import filter) 117

X

XLS-format 42

Y

Yokogawa (import filter) 118