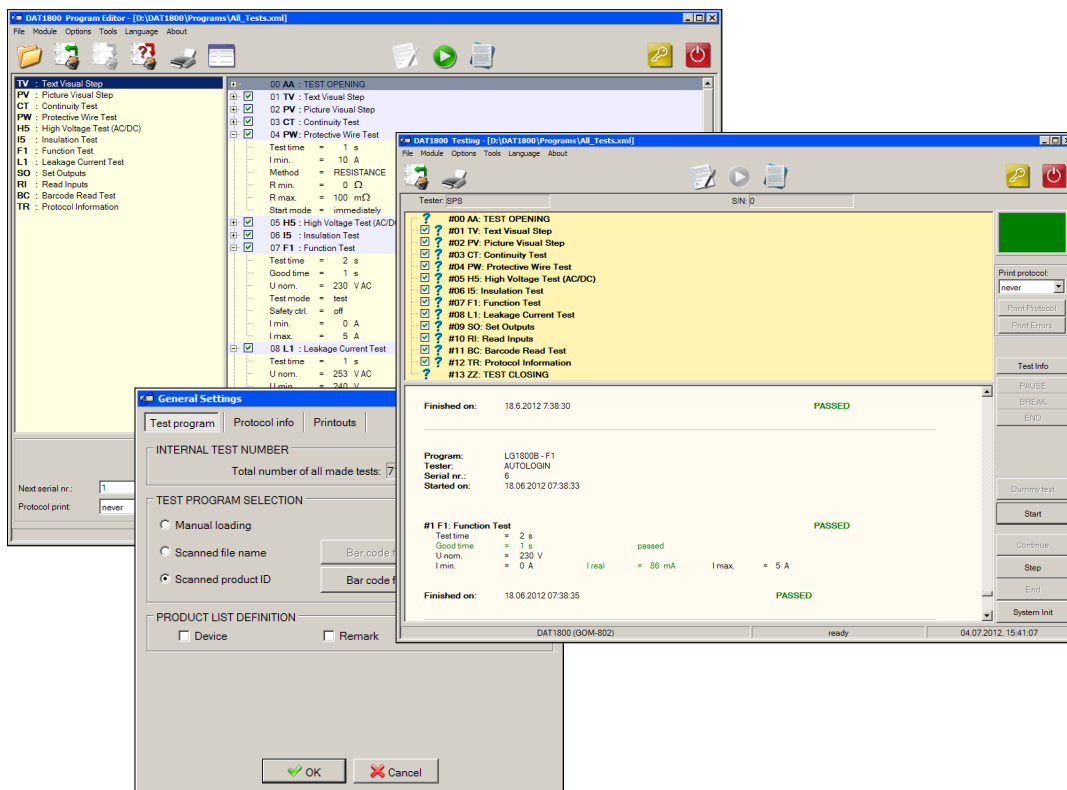


Operating Manual for Remote Control Software DAT3805 Surge Tester ST3800

Software version 2.x

Date of issue: 29 Mar 2018



SPS electronic GmbH
Blätteracker 18, 74523 Schwäbisch Hall

Phone: +49 7907 878 - 0
Fax: +49 7907 878 - 999

e-mail: info@spselectronic.com
Internet: www.spselectronic.com

Contents

1	Hardware Description	5
1.1	Range of delivery	5
1.2	Device functions	5
1.3	Design and functions	6
1.3.1	Front panel	6
1.3.2	Rear panel	7
2	About the Surge Test	9
2.1	Methods of evaluation	10
2.1.1	Error area	10
2.1.2	Differential error area	11
2.1.3	Tolerance band method	12
2.1.4	Glow discharge ("corona" method)	12
2.1.5	Double impulse	13
2.2	Mastercurve Editor	14
2.2.1	Recording a new master curve	17
3	Description of the Software	18
3.1	Overview	18
3.2	The menu bar	19
3.2.1	General Settings	20
3.2.2	Hardware Settings	23
3.2.3	Environment settings	28
3.2.4	User Administration	33
3.3	The Product List	34
3.4	Module „Editor“	35
3.4.1	General	35
3.4.2	Editor: Testinfo	36
3.4.2.1	Protocolling to a Printer	37
3.4.2.2	Test Statistics	37
3.4.2.3	Protocol settings	38
3.4.3	Editor: Test steps	39
3.5	Description of Test Parameters	40
3.5.1	General Information	40
3.5.2	Start of Test (AA)	41
3.5.3	Text Visual Step (TV)	42
3.5.4	Picture Visual Step (PV)	43
3.5.5	Surge Test (SG)	44
3.5.6	Insulation Test (I2)	47
3.5.7	High Voltage Test (H2)	48
3.5.8	FD: Field Direction test	49

3.5.9 R3: 3-phase Resistance Test	50
3.5.10 R5: Resistance Test	51
3.5.11 BC: Barcode Read Test	52
3.5.12 Protocol Information (TR).....	53
3.5.13 ZZ: Test Closing	54
3.6 Program Module "Testing"	55
3.6.1 Start of Test	56
3.6.2 Testing mode "Step"	56
3.6.3 Faulty DUT	56
3.6.4 Error-free DUT	56
3.6.5 Interrupting a running test.....	57
3.7 Program Module "Results"	58
Annex	59
A Interface configuration	59
A-1 External I/O-Interface X1	59
A-2 Second external I/O Interface X2	61
A-3 Analog Interface X3.....	63
A-4 Connector for sensor of direction of rotation X4.....	64
A-5 Serial RS-232 Interface X5.....	64
B Technical Data	65
C Terms of warranty	66

1 Hardware Description

1.1 Range of delivery

While unpacking the device, make sure that the following parts are complete:

Number	Description
1	Test device ST 3800 in housing
2	Power cable
2 resp. 4	Test connection cable, red, with terminals
1	CD or DVD media with application software 3800DAT
1	Operating manual

1.2 Device functions

With the Surge Tester ST 3800 winding goods like stators, rotors, transformers etc. can be tested for insulation damages and predamages.

Not only already existing interturn short-circuits are being detected, but also the test procedure detects predamages, not yet recognized as electric interturn short-circuit.

Highlights:

- test voltage 500 VDC – 6000 VDC
- surge capacity 47 nF
- integrated PC for control, evaluation and storage
- WINDOWS surface for user-oriented operation
- Switching matrix for 3-phase DUTs (ST 3800D and ST 3800E)
- integrated LC color display (ST 3800B and ST 3800E only)

1.3 Design and functions

1.3.1 Front panel



Fig. 1: Front view of ST 3800

1.3.2 Rear panel

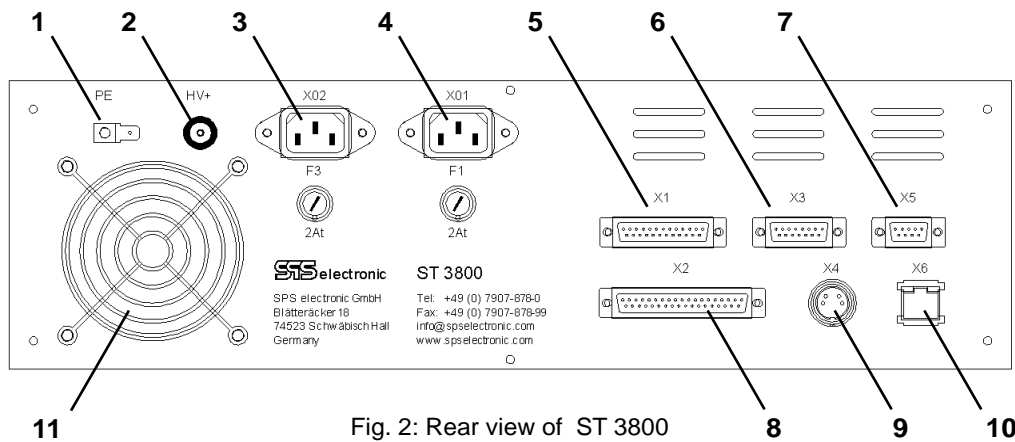


Fig. 2: Rear view of ST 3800

- 1 PE: DUT connector „Minus“
- 2 HV+: DUT connector „Plus“
- 3 X02: Mains supply for voltage generator, with fuse F3 (2At)
- 4 X01: Mains supply for internal PC, with fuse F1 (2At)
- 5 X1: digital IO-Interface („internal IO“)
- 6 X3: Interface for analog IO
- 7 X5: serial Interface for additional external device
- 8 X2: digital IO-interface („external IO“)
- 9 X4: Connector for sensor of direction of rotation
- 10 X6: Network connector for Ethernet
- 11 Ventilation grid – keep absolutely free of obstruction!

2 About the Surge Test

The most significant difference between the surge test and all other EST tests (high voltage tests, insulation tests, etc.) is that there are no fixed thresholds to judge the test results as **GOOD** or **FAIL**. Instead, an electric oscillation gets excited within the DUT by a line surge. Then, the task is to judge the *characteristics* of the resulting oscillation!

Therefore, prior to performing any real testing, it must be evaluated how the oscillation of the DUT should look at all. For that purpose, several test runs with DUTs confirmed to be error free are performed. By averaging their oscillation curves, the so-called **master curve** is obtained. Later, when doing real world test runs, the measurement of the DUTs is compared to that master curve to decide if the result is **GOOD** or **FAIL**. The software evaluates the percental deviation of the measured curve to the master curve. The percentage of the maximally allowed deviation can be specified by the user.

2.1 Methods of evaluation

The Surge Tester ST3800 offers several methods of curve evaluation. In the following, the currently implemented evaluation methods are described.

(More evaluation methods are planned.)

2.1.1 Error area

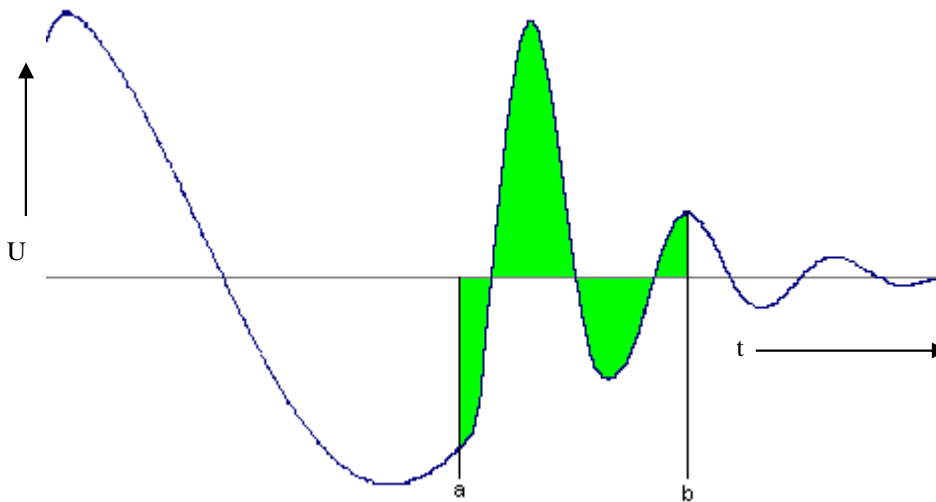


Fig. 3: Curve with evaluation of error area

Relevant for this method is the area included between a curve and the time axis. The area derived from the actually measured curve is compared to the area of the reference curve, and the percentual deviation is calculated. Fig. 3 shows an example with fixed integral boundaries, between which the evaluation is done.

Mathematically, the used formula is this:

$$\frac{\int_a^b |U(t)\{\text{Prüfling}\}| dt}{\int_a^b |U(t)\{\text{Master}\}| dt} = A_{\text{Fehler}} \text{ in } \%$$

The areas of the reference curve and of the test specimen are computed. Subsequently, the deviation is calculated by division of the two areas, and indicated in per cent.

The crucial point for error detection is the size of the curve area. Phasing is not considered. Thus, the testing is sensitive to short-circuited coil, since the change of area size is proportional to the energy loss after the surge, and energy loss increases vastly due to short-circuit current.

The optimal result of this test is 100% (area of measured curve == area of reference curve).

The more the result becomes smaller or bigger than 100%, the more different the DUT is to the master.

2.1.2 Differential error area

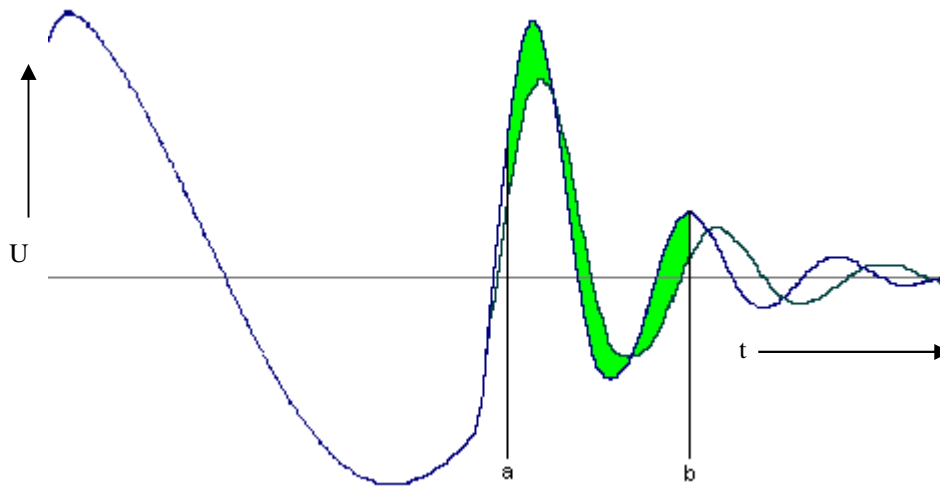


Fig. 4: Curves with differential error area

This method determines the difference area of master curve and specimen curve, then calculates the ratio of that difference and the area of the master curve. This method is more stringent than the error area method, in that it also evaluates phase shifts caused by winding tolerances. Therefore this method is used when uniformity of coil windings and inductive reactance is of major concern (e.g. exploring coils).

The evaluation of master curve's area is the same as used in the error area method, however instead of using the area of the measured curve, the difference between measured curve and master curve is calculated, and the area of this difference is used:

$$\frac{\int_a^b (|U(t)\{Master\} - U(t)\{Prüfling\}|) dt}{\int_a^b |U(t)\{Master\}| dt} = A_{Fehler} \text{ in \%}$$

The optimal result of this test method is 0% (measured curve shows no difference to the master curve).

The bigger the result's percentual value gets, the more different the DUT is to the master.

The relative size of the result is highly dependent on the amplitude of the master curve: if the amplitude of the master curve is rather small, then even relatively small deviations of the DUT may lead to "big numbers": results in range of 1000% are absolutely common..

Therefore, for this test method it is necessary ...

- to choose the voltage range as small as possible, so that the master curve has a sufficiently big extension in "y"-direction
- to place the evaluation period so that only the very first oscillations (after the swing-in transient) are measured, and not the swing-out transient.

2.1.3 Tolerance band method

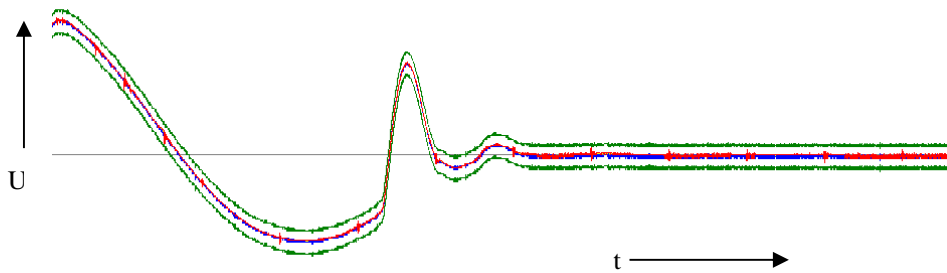


Fig. 5: Tolerance band method

With this evaluation method – also called "envelope method" – the surge curve has to be located inside of a programmable tolerance band. The tolerance band is given as a percentual value. By adding and subtracting this percentage from each sample of the master curve, two curves are derived: these are the envelope curves.

In reference to Fig. 5:

the green curves are the envelope, the area between the green curves is the tolerance band.

The blue curve is the master curve, from which the envelope has been derived.

The red curve is made of the samples measured from the DUT.

This test method evaluates the number of test samples that are located outside of the tolerance band, then builds the ratio of this number to the number of all measured samples.

The example in Fig. 5 has a result of 0%, i.e. there are no samples outside of the tolerance band.

2.1.4 Glow discharge ("corona" method)

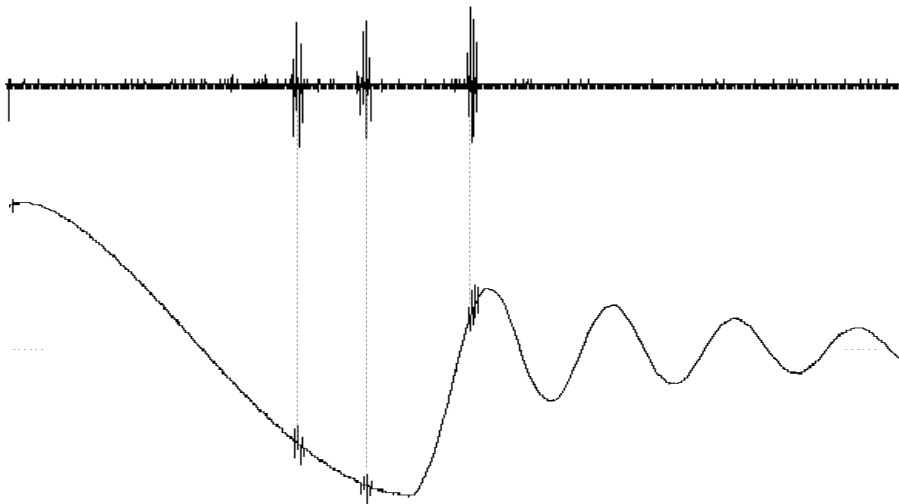


Fig. 6: Glow discharge (corona)

This method deals only for recognizing of incipient weaknesses of wire insulation on the inside of the coil, which manifest by partial discharges, glow discharges (corona), and in extreme cases by arc-overs.

Using a dedicated mathematical procedure, the amount of "high-frequency energy" on the test curves is evaluated (visible on the oscillation curve as "needles" or "points"). In equivalence to an analogue circuit, this means to apply a highpass to the curve (differentiation), then evaluate the energy content (area of the differentiated curve). Other errors (i.e. galvanic) are not covered by this method.

2.1.5 Double impulse

With this method no reference curve is compared with the DUT but two winding impulses are applied one shortly after the other at the same DUT. If it is a PASS DUT the two measurement curves are congruent. If one DUT is insulation weak the two curves will differ because arc-overs and glow discharges occur in irregular intervals. It has to be taken into account that in case of coils with iron cores the course of the curve changes due to the re-magnetization process. In such a situation a certain number of impulses have to be supplied until the iron core is saturated. With this method only the insulation of a DUT can be tested however not the interturn short circuit or winding deviations. The evaluation is performed similar to the tolerance band method only the limits are set considerably lower.

This method recognizes the same faults as the glow discharge method only that less computing performance is required instead the DUT is exposed twice to an impulse voltage.

Working with master curves

The application stores master curves as *.MCU files. Within a MCU file, several different master curves can be stored, or only a single one.

If a test program opened in the editor is saved, the actual chosen MCU file and active master curve gets linked to the program. The next time that program is opened - be it within the editor or during testing - that particular master curve of the linked MCU file is used again.

This behaviour enables different strategies to work with master curves:

- When working with only a small amount of different DUT types, it might be easier to have one MCU file for every kind of DUT, where every MCU file is named according to the DUT type, and holds only one master curve.
- When working with a big range of products, where perhaps certain series of DUT are splitted into several sub-types, one will probably prefer to have an MCU file for every series, and to hold different master curves in them, one for every DUT variant of a given series.

2.2 Mastercurve Editor

For the test step "surge test", the configuration dialog consists of two different windows.

In the 1st window, the general evaluation parameters for the test are set. These parameters correspond to the actually chosen master curve, the name of which is shown in the Field MASTER CURVE:

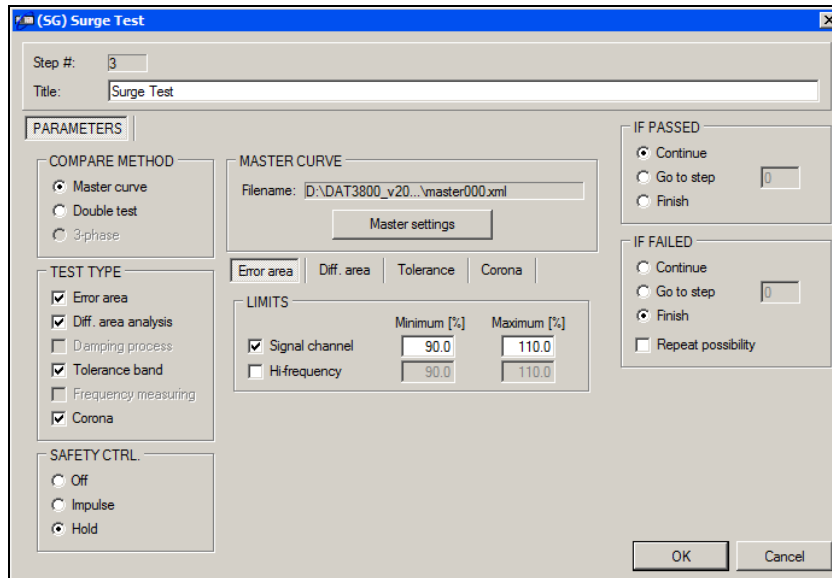


Fig. 7: Test parameters of surge test

In order to choose another master curve to use for the test (or to edit an existing master curve, or to record an all-new master curve), the button *Master Curve Editor* has to be used. This will open the main window of the master curve editor, with which all these actions can be carried out.

This is described on the following pages.

The Main Window of the Master Curve Editor:

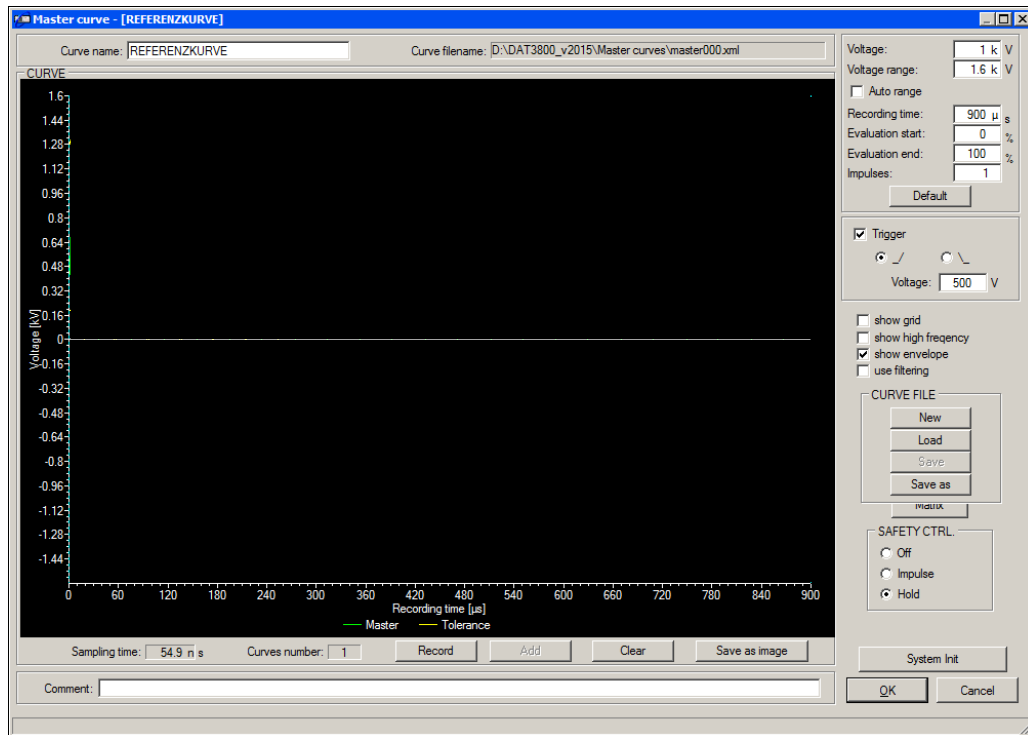


Fig. 8: Master curve editor

At the top of the screen, the name of the actual mastercurve is shown, as well as the file in which the master curve is saved.

At the bottom of the screen, a comment to the actual mastercurve can be entered.

On the right-top, the values to use for recording a curve are set:

- Voltage** Defines the test voltage for the surge test. Possible values are 500 V up to 6000 V.
- Voltage range** Defines the voltage range (y axis) manually.
- Auto range** Sets the voltage range automatically, according to the test voltage.
- Recording time** Defines the range for the x axis.
- Evaluation start** In case that the evaluation range should not cover the entire recording time, it can be restricted. The start of evaluation can be entered in %, and is shown in the diagram by a dashed line.
- Evaluation end** In accordance to the previous field, the end of evaluation can be defined (in %). This is also shown in the diagram by a broken line.
- Impulses** This defines the number of surge impulses. Only the last impulse will force a visible recording. For certain DUTs it is advantageous to perform several impulses prior to the recording, in order to get more stable results.
- Trigger** When activated, the data recording is started when the chosen trigger settings are met, i.e. when the chosen voltage is reached for the first time on the rising (resp. trailing) edge.

Further on, the buttons at the right side offer the following actions:

- New** – clears the curve memory, in order to record an all-new curve.
- Load** – loads a saved curve from the hard disk
- Save** – saves the actual curve (keeping the actual filename).
- Save as** – saves the actual mastercurve to a new file.

2.2.1 Recording a new master curve

Once that curve name and recording parameters are set, the recording of a new master curve may begin. This is done by the buttons right below the black display field:

- Record** Starts the recording of a new curve with the shown parameters.
Attention: High voltage is applied to the DUT !
 The recorded curve will be displayed in red.
- Add** This adds the actually recorded curve (red) to the storage. By doing so, the curve's color will change to green.
- Clear** Clears curve memory for further recordings.
- Save as image** Saves the displayed curve as bitmap graphic.
- Curves number** The number of recorded curves that have been put into the curve memory.

After recording a curve, one will get a display similar like this:

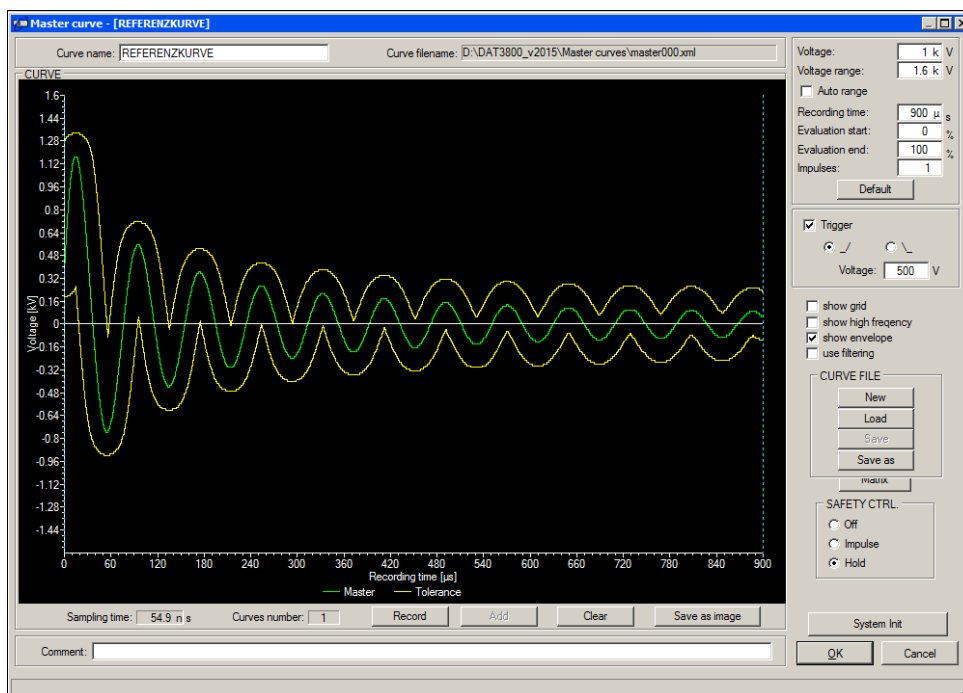


Fig. 9: Recorded curve

Typically, one will build up a master curve by doing several recordings of different DUTs. Since the software will calculate the average of all recordings, the resulting mastercurve will be the representative mean of all DUTs that were used to build the master curve.

Immediately after a recording, the recorded curve is shown in red. If the curve seems to be reasonable, it can be added to the internal curve memory by the button "Add". This will compute the new recording into the existing average of all recordings. The counter "curves number" is increased by 1, and the panel shows the resulting master curve in green.

3 Description of the Software

3.1 Overview

The system software of Surge Tester ST 3800 is based on the operating system MS WINDOWS and therefore offers optimal user friendliness. With just one click of your mouse the program will execute the respective command. All the functions which WINDOWS offers as a standard, are also included in this software.

The compilation of a test run is really quite easy: the individual tests can be "clicked together" via mouse in any number and order and can be deleted in the same way. Even later relocations in the test run are easily possible.

The test programs can be created on any IBM-compatible PC under Windows and can be transmitted to the ST 3800 transportable media or network.

Each program maintains statistics on the tests executed, which can be used for quality assurance. All the measuring results are stored in a type related file or a file of the day and can be printed at will via the standard printer interface. At the same time a printout of the current measuring data can be had after each test. An unmistakable assignment between DUT and measuring protocol is, even for years, easily possible.

Please find explanations and information pertaining to the graphic operating surface WINDOWS in your Microsoft manuals!



3.2 The menu bar

<i>Menu</i>	<i>Menu item</i>	<i>Function</i>
<i>File</i>	<i>New</i>	To create an all-new test program with file name NONAME.PRG.
	<i>Load</i>	Load an existing test program from disk
	<i>Save</i>	Store the current test program to disk
	<i>Save as</i>	Store current test program with a new name
	<i>Print</i>	Prints the current test program, with all parameters
	<i>Product list</i>	Starts the Product List editor. See chp. 3.3.
<i>Options</i>	<i>General settings</i>	General settings about test program selection, DUT serial numbers, etc. See chp. 3.2.1.
	<i>Hardware settings</i>	Hardware-related settings. See chp. 3.2.2.
	<i>Environment</i>	Other options about pathnames, results management, default settings, etc. See chp. 3.2.3.
	<i>Printer setup</i>	Configuration of the printer to use for printing & protocolling.
	<i>Users & rights</i>	Management of registered users and their rights. See chp. 3.2.4.
	<i>Change password</i>	Here the current user can change his password.
<i>Tools</i>	<i>Programs managment</i>	---
	<i>Service module</i>	---
	<i>Connection matrix</i>	---
	<i>System DB managment</i>	---
<i>Language</i>	<i>Deutsch</i>	Sets the software to appear in German language.
	<i>Englisch</i>	Sets the software to appear in English language.
<i>About</i>	—	Shows information about the software ST3800.

3.2.1 General Settings

Tab "Test program":

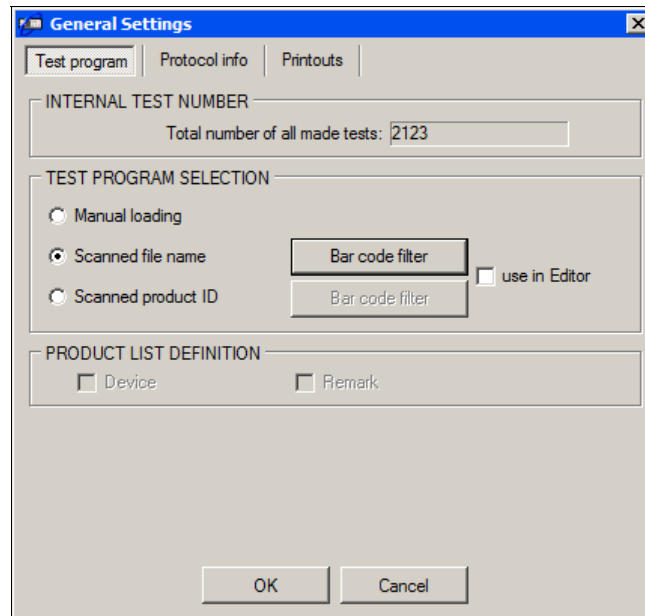


Fig. 10: Tab "Test program"

The field *Internal test number* shows the total number of all tests performed so far.

With *Product list definition* one can choose whether the positions "Device" and/or "Remark" shall be available in the product list.

In the field *Test program selection*, it is specified how the required test program is determined:

- *Manual loading*: when changing the used test program, the user will have to load the new one manually
- *Scanned file name* : The filename is taken directly from the specified positions of the scanned barcode.
- *Scanned product ID* : Here the product list deals as a "lookup table". Based on the product ID, the according test program is determined and loaded from the product list.

The button „Bar code filter“ opens a dialog, where one can specify how the barcode data is evaluated:
(The item "Product ID" changes to "File name" if *program selection* is set to "scanned file name".)

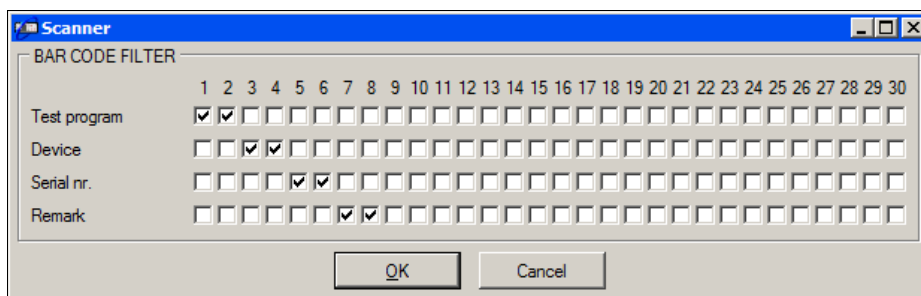


Fig. 11: Definition of barcode evaluation

Tab "Protocol Info":

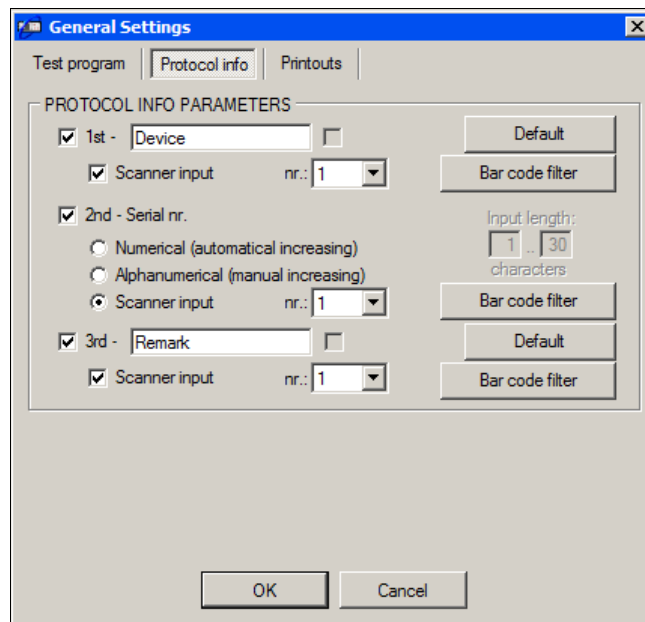


Fig. 12: Tab "Protocol info"

In this register it is specified which informations about the DUT will be included in the test protocolling, and where they are taken from.

Generally, only positions that have the leftmost checkbox activated are included in the protocol. When unchecked, the respective position is **not** included in the protocol.

If "Scanner input" is checked for an item, then the according data is taken from the scanned barcode data. Else, the according data is read out of the product list (except when in the register „test program“, field *definition product list*, the according position has been deactivated).

The serial number can either be read from the barcode, or be handled numerically/alphanumerically:

- Numerical serial numbers are automatically increased by »1« with each test.
- if alphanumerical serial numbers are used, then one can assign an individual serial number for each new DUT.

Additionally, for all positions set to »Scanner input«, the number of the scanning operation to retrieve the respective data with can be chosen.

Example: DUTs may carry multiple barcode lables, and one might want to read the test program from the first barcode label, but the serial number and DUT description from a second barcode label. In this case, one would set the »nr.« field for »device« to »1«, and the fields for serial number and remark to »2«.

For protocolling purposes, the items "device" and "remark" can be renamed individually. The according items in the register "Test program", as well as those in the product list, will automatically reflect the change. (The »default« buttons will revert these fields to the original namings.)

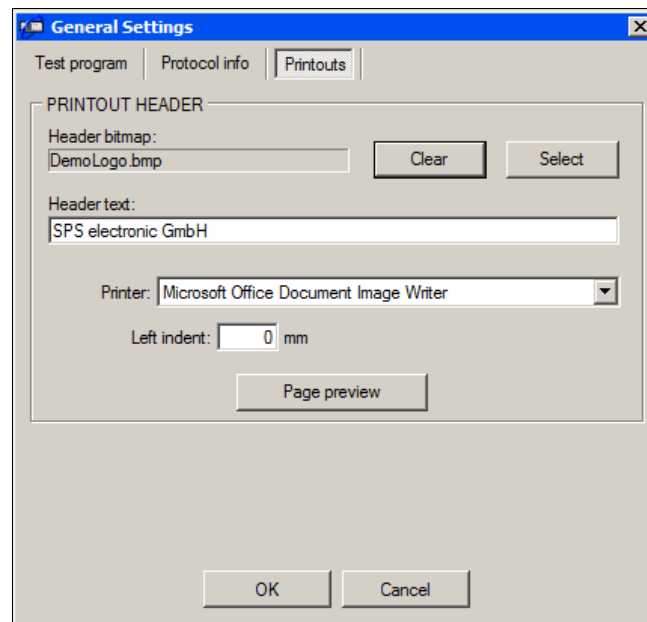
Tab "Printouts":

Fig. 13: Tab "Printouts"

With the dialog "Header-Bitmap" you can use a previously created bitmap as protocol header.

(Note: To prevent your bitmap from quality loss through scaling, the image should be 500 pixels wide, or an integer multiple of that.)

With "Header text" you can enter any text which will be put at the top of each new test.

"Page preview" will show a visualization of how the printed protocol will look like later on.

3.2.2 Hardware Settings

Tab "Device I":

Common:

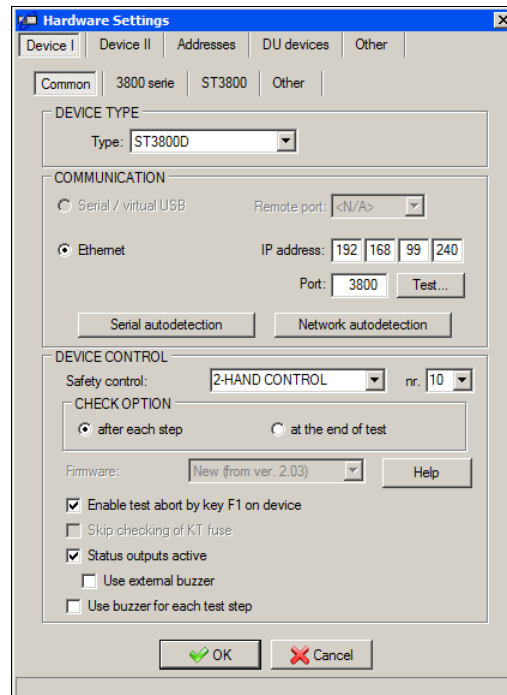


Fig. 14: Tab "Device I / Common"

- Under *Device Type* the type of testing device is chosen. It's important that the exact type is chosen – otherwise it could happen that the device is controlled with wrong or unsuited commands.
- In *Communication* it is specified over which connection the testing device and the remote PC with DAT3800 software are communicating:

If the device is "locally" connected to the PC, the option "*Serial / virtual USB*" has to be chosen, and the actually used interface must be specified under "Remote port".

By means of the button "autodetection" the software can automatically recognize the type of testing device and the used interface, given that the tester is powered on, set to operating mode "automatic", and is connected to the PC.

Only possible for device series 3800: For network operation the option "Ethernet" has to be chosen. In this case, the IP address of the tester has to be specified as well as the network communication port. The button "Test" opens another window in which the communication can be tested.

- Device Control:

In the list field *Safety control* it is specified how a test run is started, resp. how the DUT gets connected.

For device series 3800: if an external start signaller is used, the digital input by which the start signal is fed in must be specified in the dropdown-list "Nr.". (See also operating manual of the 3800 device.)

For some devices the *Firmware* version of the device must be specified.

The option *Enable test abort by key F1 on device* enables the possibility to abort a running test with the operating elements on the device's front panel.

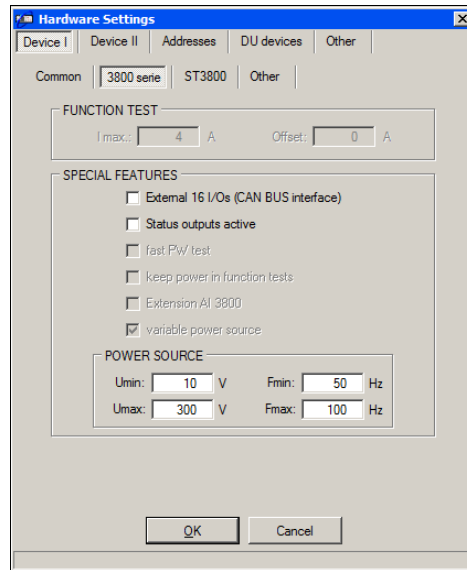
3800 serie:

Fig. 15: Tab "Device I / Common"

For series 3800 devices, the following device specific options can be changed:

- External 16 I/Os*** For accordingly equipped devices, this enables the 2nd I/O-Interface for usage
- Status outputs active*** Enables or disables outputting of device status signals over the standard I/O-interface (see also operating manual of device 3800)

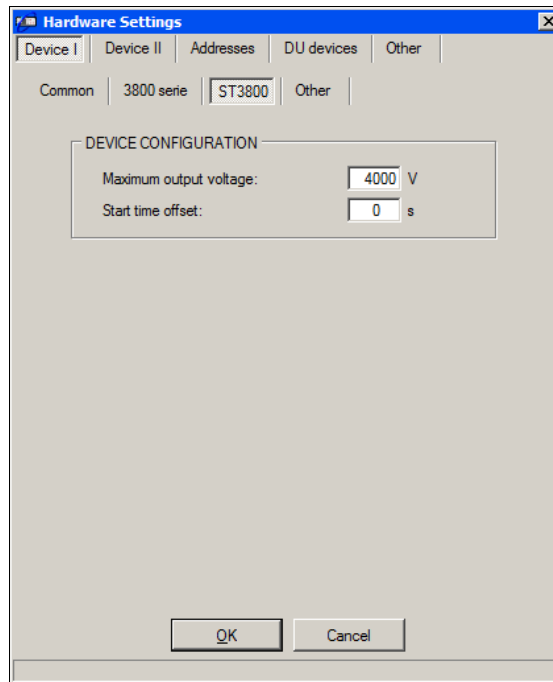
ST 3800:


Fig. 16: Tab "Device I / ST3800"

- Device Configuration

- *Maximal output voltage*

Here the maximum output voltage for the surge test can be limited.
The possible range for max. voltage is 4000V to 6000V.

- *Start time offset*

Defines the delay time after initiating the surge impulse, before the measuring is started.
The possible range is 0s – 10 μ s.

Safety note:

According to EN 50191, devices without safety circuit may be operated only if the following conditions are met:

DC : Current < 10 mA and electric charge < 350 mJ

With devices ST3800, the limit of 350 mJ is reached with an output voltage of 4000V.

Therefore, when operating without additional safety measures, the output voltage must not be set higher than 4000V!

To use voltages higher than 4000V, it is necessary to install additional safety measures acc. to EN 50191 !



Tab "Device II":

This tab is intended for the case that two different testing devices are controlled by the software 3800DAT.

In that case, the settings for the 2nd device can be configured in this tab. The available settings are the same as described in the preceding pages for "Device I".

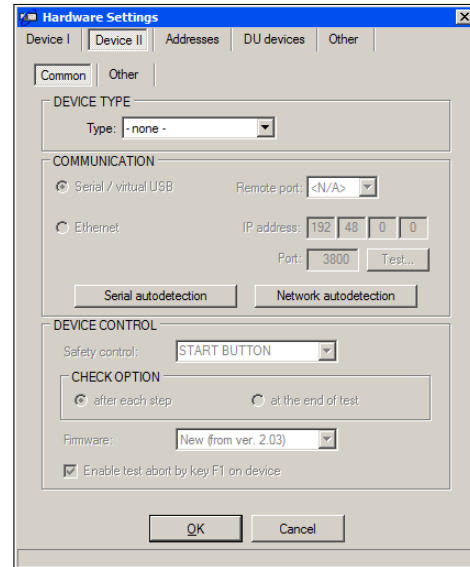


Fig. 17: Tab "Device II"

Tab "Addresses":

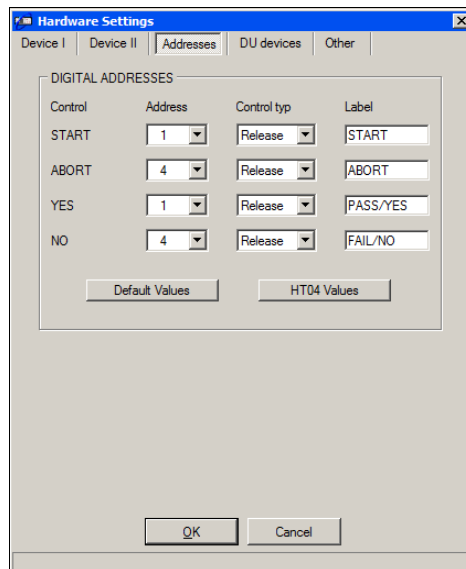


Fig. 18: Tab "Addresses"

By means of *Digital Addresses* you can occupy certain inputs by default. This is very handy, since you can e. g. acknowledge a dialog with the start button, not having to pop around between PC/Mouse and DUT.

With the *Control type* fields, you can choose whether the button shall react immediately when it is *depressed*, or not until the button is *released* again.

- Tab *Other*:

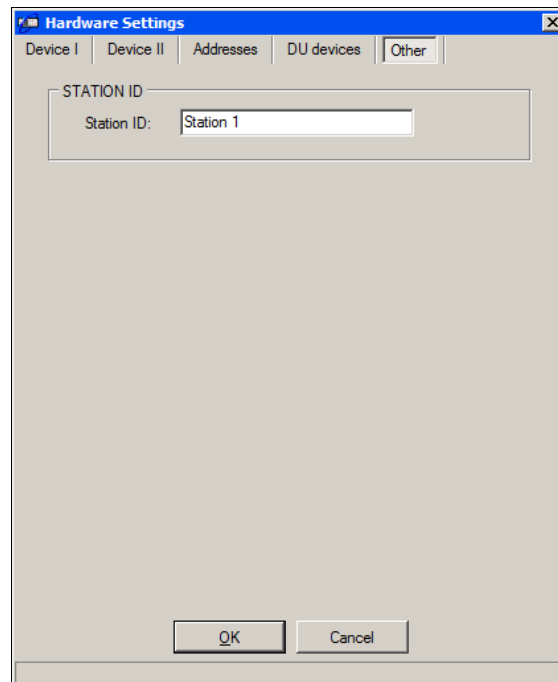


Fig. 19: Tab "Other"

The "Station ID" deals for identification of the test station, and is included in all test protocols. If several test stations are connected in a network, and all results are stored in one central database, the test results can easily be back-tracked to the test station on which the test has been performed.

3.2.3 Environment settings

Choosing *Options* → *Environment*, the following tabs are available:

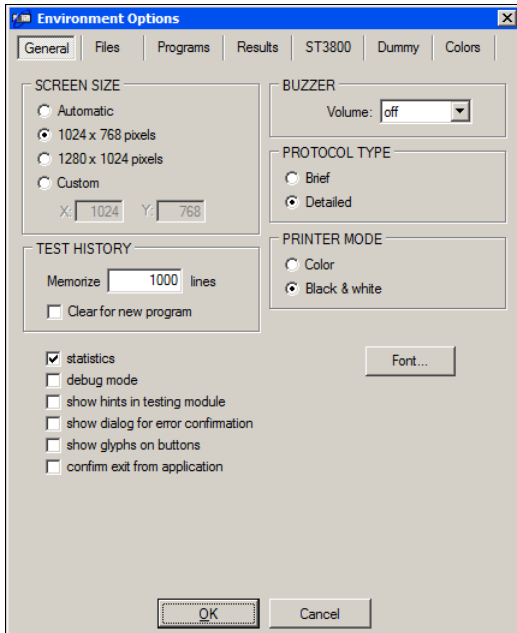


Fig. 20: Tab "General"

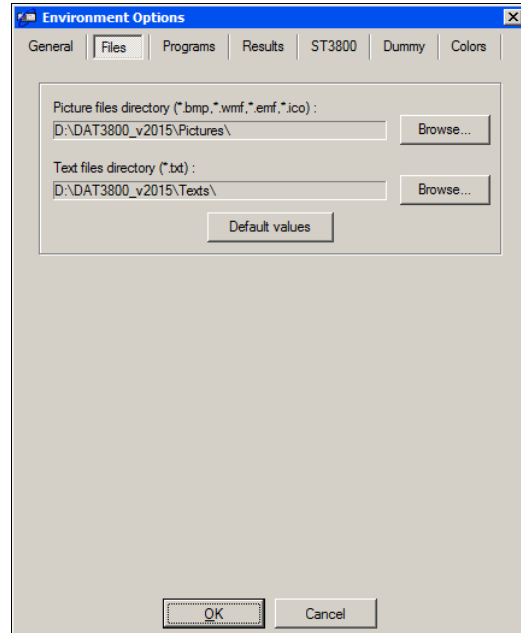


Fig. 21: Tab "Files"

Tab "General":

- In *Screen Size* the dimensions of the application window can be set.
- The *Buzzer* of the testing device can be adjusted.
- *Protocol Type* switches between "Brief" (only results) and "Detailed" (results with detailed test information).
- The *Printer Type* should be set to "Color" or "Black&White" according to the printer type you're actually using.
- The option *statistics* activates or deactivates the internal tests statistics module.
- *Debug Mode* is only needed for service matters, and normally shouldn't be used.
- *Show hints in testing module* will activate tooltip-balloons in the testing module: when the mouse cursor is located on top of a test step, the test parameters of the test step are shown.
- In the lower part, one can change the folder paths to store picture files and text files.

Tab "Files":

Here one can choose the destination folders for saving of Picture-files and Text-files, which are used by several test steps.

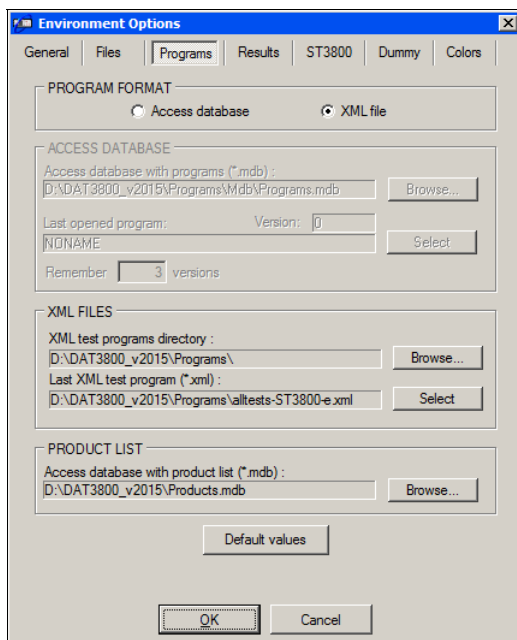


Fig. 22: Tab "Programs"

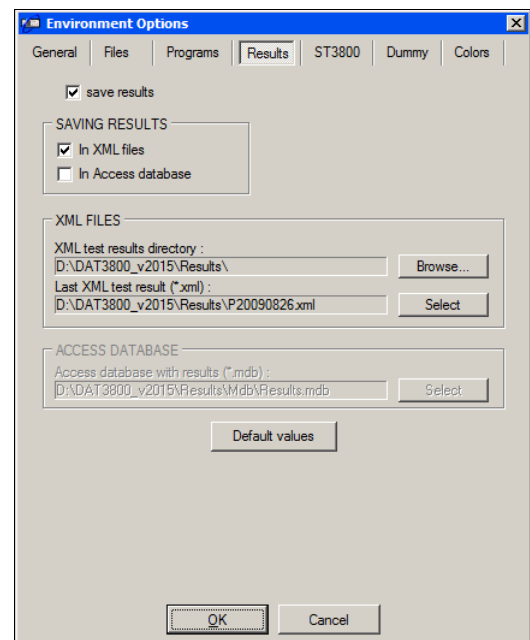


Fig. 23: Tab " Results"

Tab "Programs":

Here one can choose whether the test programs shall be saved as XML data files (each test program is saved in an individual *.xml file), or if all test programs shall be stored in a data base.

Depending on the choice, the according options are enabled, where the storage path for the XML files, resp. the target data base can be chosen.

If program storage in a data base is chosen, there is the possibility to keep a certain number of "previous versions" when a test program is edited.

Tab "Results":

Here one can choose whether the test results shall be saved as XML data files (each result protocol is saved in an individual *.xml file), be stored in a data base, or both together.

Depending on the choice, the according options are enabled, where the storage path for the XML files, resp. the target data base can be chosen.

By disabling the checkbox "save results" it is possible to switch off the results protocolling completely. This may come handy e.g. for set-up operation with new DUT types.

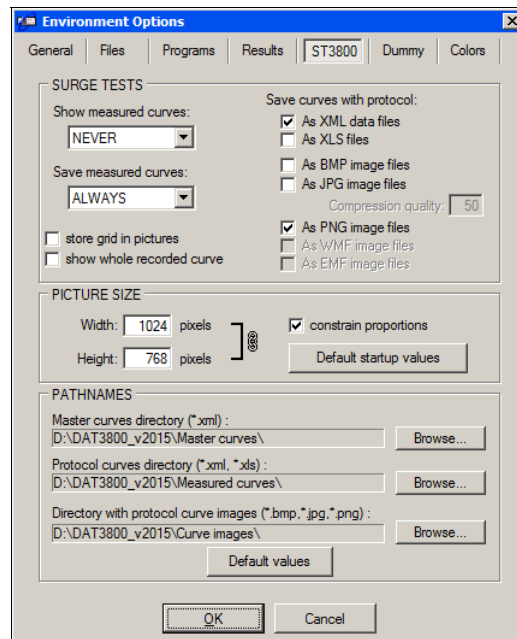


Fig. 24: Tab "ST3800"

Tab "ST3800":

- Under "*Surge Test*" below adjustments are performed:
 - Under "Show measured curves" it can be selected under which conditions the recorded curve is to be displayed on the screen after an impulse winding test. Possible are *ALWAYS*, *PASS*, *FAIL* and *NEVER*.
 - Under "Save measured curves" it can be selected under which conditions the recorded curve is to be stored after an impulse winding test. Here, too, *ALWAYS*, *PASS*, *FAIL* and *NEVER* are possible. Additionally it can be selected in which format the curves are to be stored. Available are the pure data format *.pcu, and the picture / image formats *.bmp, *.jpg, and *.png.
- In the field "*Picture Size*" the standard picture size can be preset in pixel for the diagrams. If "keep proportions" has been activated the respective other value will be automatically adapted when entering a new value for width or height.
- In the field "*Pathnames*" the folder locations for storing of the various data files can be specified.

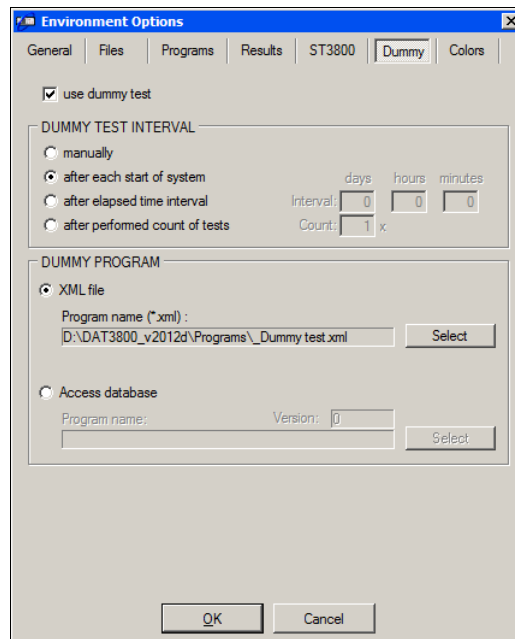


Fig. 25: Tab "Dummy"

Tab "Dummy":

With the options of this tab, the execution of a regular dummy test can be forced, e.g. to ensure that the system is functioning correctly.

In the field "*Dummy Test Interval*" the execution interval of the dummy test can be chosen.

In the field "*Dummy Test Program*" the test program to use for this test can be specified.

If a dummy test is pending since the time interval criterium is met, then the software will automatically force the dummy test to be executed. Regular test operation is possible again only after this dummy test has been passed successfully.

(Note: Users with the right "*Skip Dummy Test*" are authorized to skip a pending dummy test.)

Tab "Colors":

In the multi- tab "Colors", the colors of various elements of the application can be changed to liking:

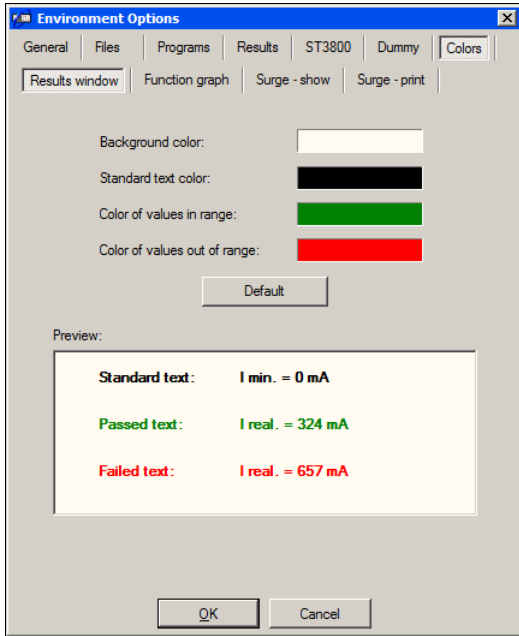


Fig. 26: Tab "Results window"

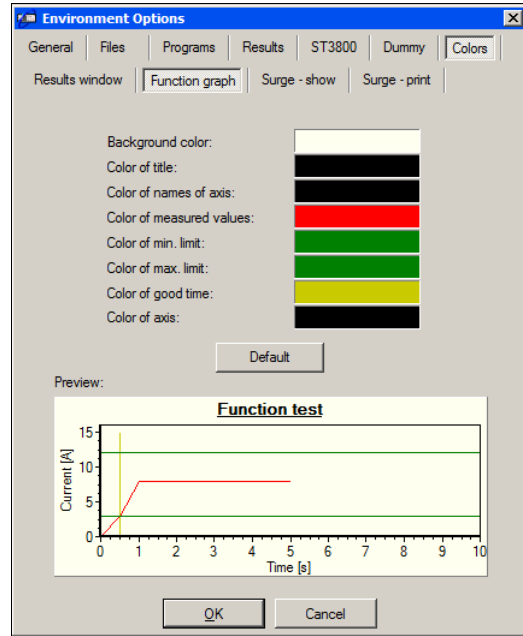


Fig. 27: Tab "Function Graph"

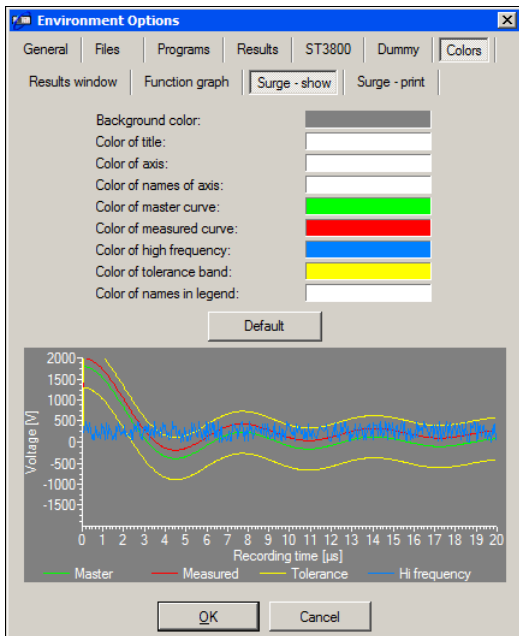


Fig. 28: Tab "Surge – Show"

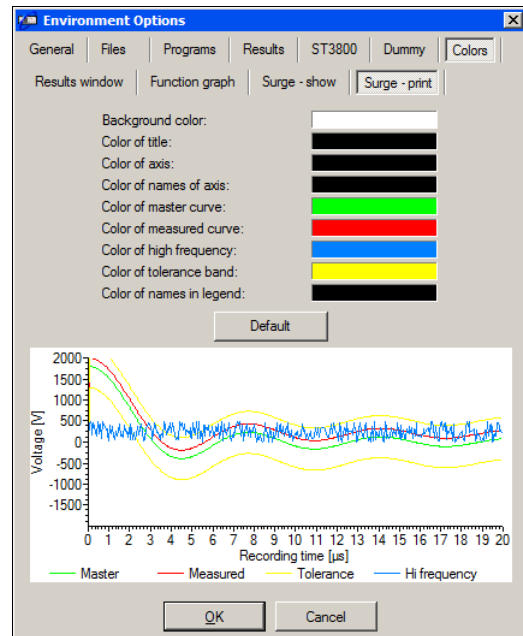


Fig. 29: Tab "Surge – Print"

3.2.4 User Administration

The software has a built-in *User administration*. Here all users of the software must be registered. To get into the program for the first time, you have to enter "SPS" as user and password both. After this, you can enter new users with their depending passwords and rights (see Fig. 30).

After having given the users and rights confirming your needs, we recommend to delete the user "SPS" in order to avoid abuse of the software.

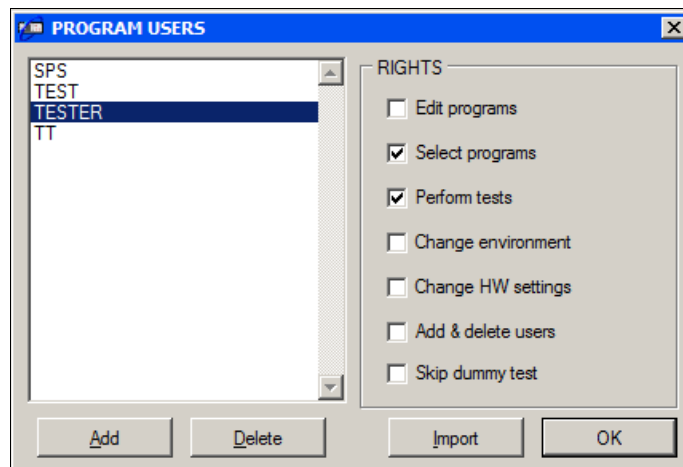


Fig. 30: User administration

Explanation of the rights:

<i>Edit programs</i>	enables editing of test programs
<i>Select programs</i>	enables loading of test programs
<i>Perform tests</i>	enables to switch to the module »testing«
<i>Change environment</i>	enables changes of program environment
<i>Change HW settings</i>	enables changes of hardware settings
<i>Add & delete users</i>	Enables to make changes in <i>Options / User administration</i>
<i>Skip dummy tests</i>	Allows to skip the daily dummy test (if dummy test is activated)

3.3 The Product List

The product list contains the assignments between devices/article numbers and the test program to use for each kind of device. Hereby it is possible to automatically load the correct test program for each DUT by scanning its barcode.

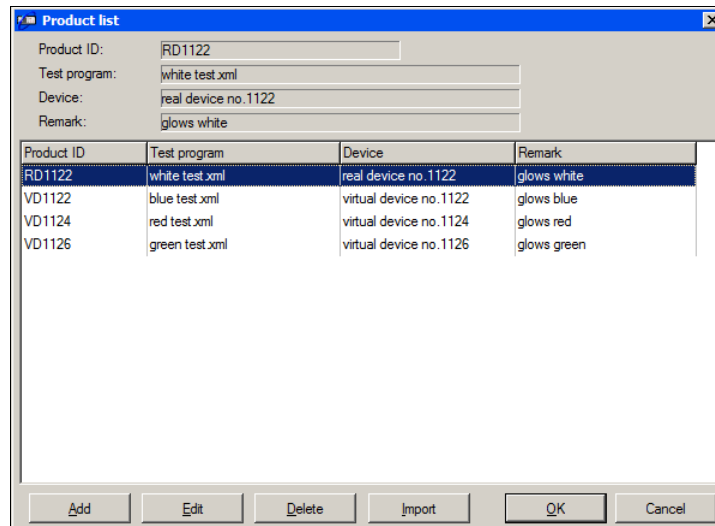


Fig. 31: the Product List

The values for *Product ID* and *Test program* are of particular importance:

- For *Product ID* one must enter exactly that character sequence that will be read from the DUT during testing.
- For *Test program* the exact name of the test program to use for the given type of DUT must be entered.
- The entry for *Device* may be of free form – it deals only for information of the tester, and for logging purposes in the test protocol.
- The same goes for the entry *Remark*.

Note: The entries „Device“ and „Remark“ are present only if in the menu *Options/General* it was defined to **not** read these values from the barcode.

To enter a new kind of device into the product list, or to edit the properties of an existing one, the buttons „Add“ resp. „Edit“ will open a new dialog window, in which the needed data can be entered:

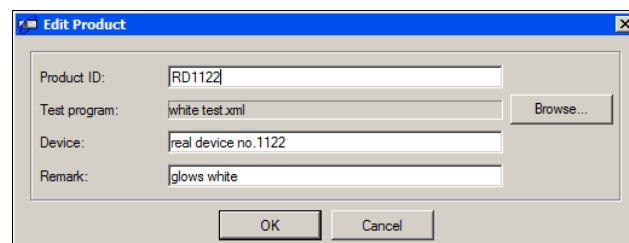


Fig. 32: dialog window „new product“

3.4 Module „Editor“

3.4.1 General

With the editor module you arrange the test, parametrize the single test steps and archive the test programs.

All test programs created with the editor are stored in the built-in hard disk and are available for later testing. Each test program has a definite name (plus extension *.prg). The name should be product-related for programs to be easy to identify.

Each test program has the following structure:

- general information: name of DUT, author, etc.
- protocol information: Whether a protocol will be created, and how should it look like.
- sequence of test run
- an internal statistic module (numeric), embedded in the program file.

The sequence of the test run is displayed in the middle of the window and can be changed with the editor's tools.

The single test steps can be:

- inserted: a new test step is inserted by double-clicking a step from the list on the left side.
- deleted: the test to be deleted has to be highlighted, then it can be deleted by keyboard's "del" key.
- edited: double-clicking a test step in the program listing will open the parameter window.
- moved to a different location in the program: this is done by "dragging" a test step with the mouse.
- copied: duplicating a test step can be done by additionally holding the "strg" key while moving the test step.

By means of the menu item "*File / Print*" the actual program inclusive all test parameters can be printed to a connected printer.

Each test step can be changed by either marking it and selecting "edit" from the right-click context menu, or by simply double-clicking it.

By doing so, a new window will open in which all parameters of the test step can be adjusted. (See chapter 3.5: "Description of test parameters".)

Each test step by default receives a name characterizing the test step. In the parameter window this name can be edited to conform with the DUT the program is written for (like e.g. "*Surge Test L2-N*").

3.4.2 Editor: Testinfo

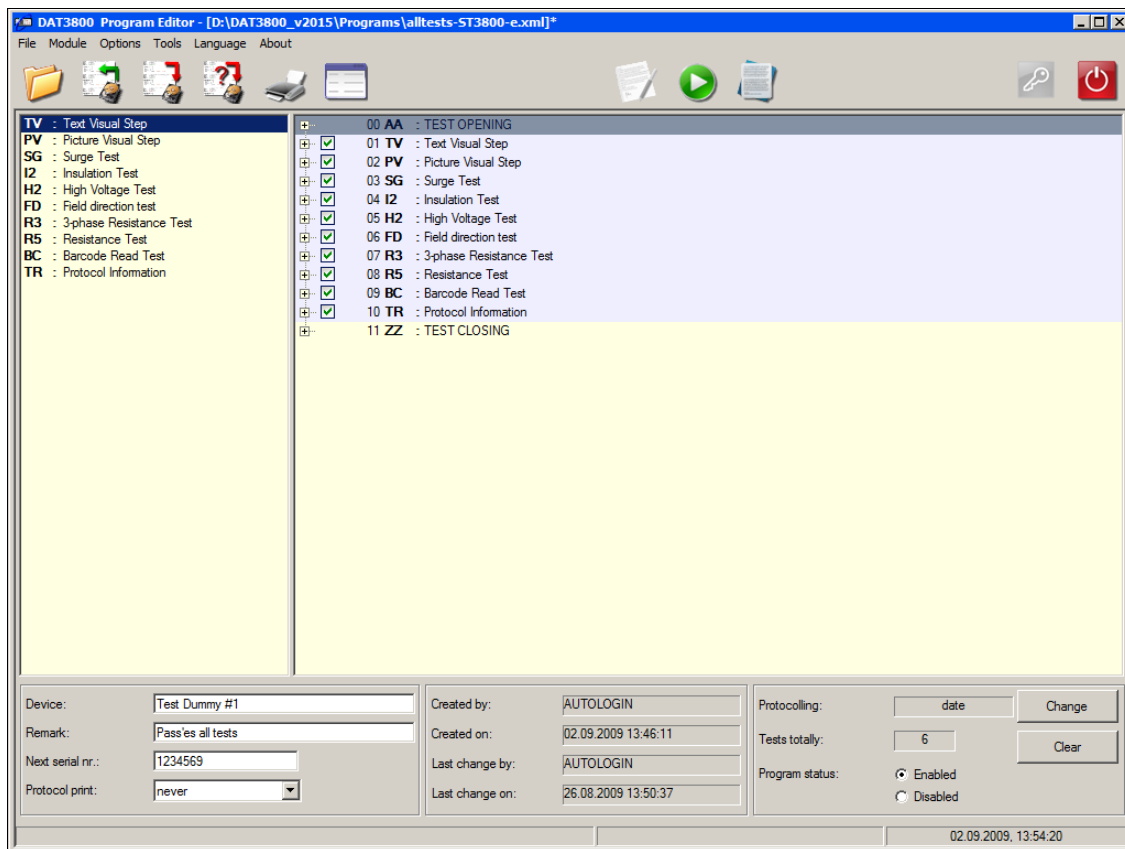


Fig. 33: Program editor

- In the title bar you see the path and name of the actual test program.
- Below there is the menu bar (see paragraph 3.2, p.19).
- The list window on the left shows all test steps that are available to be used.
- The list window on the right shows the actual test program.

In the lower part of the window there are the input fields for the general data on the DUT. You can enter here the person who is doing the testing, description of device, comments and serial number. This information does not effect the test run will, however, be shown for information in different windows and written out in the documentation.

3.4.2.1 Protocolling to a Printer

The program allows the output of the test results on the printer. Same happens after each test run. Via the drop-down list "Protocol print" this procedure can be controlled.

Below print possibilities are available to the user:

<i>List element</i>	<i>Function</i>
<i>never</i>	do not print protocol
<i>always</i>	print protocol after each test
<i>if failed</i>	print protocol only if test result was "Fail"
<i>if passed</i>	print protocol only if test result was "Passed"
<i>Failed steps</i>	Protocol only the test step that caused the error

The chosen settings are specific to the actually loaded test program, and are saved together with it.

3.4.2.2 Test Statistics

In this test program there is a tabular statistics for each program. The passed, failed and invalid tests are counted and the results of each single test step are recorded. The info window shows the number of all the tests.

The tabular statistics can be deleted via the button "Clear". Deletion of the statistics has to be acknowledged via a safety inquiry.

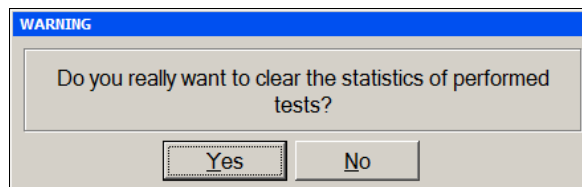


Fig. 34: Safety inquiry

If the statistics should be completely disabled (e.g. temporarily for set-up tests with new DUT types), this can be done by going to "Options / Environment / General", and disabling the entry "statistics".

3.4.2.3 Protocol settings

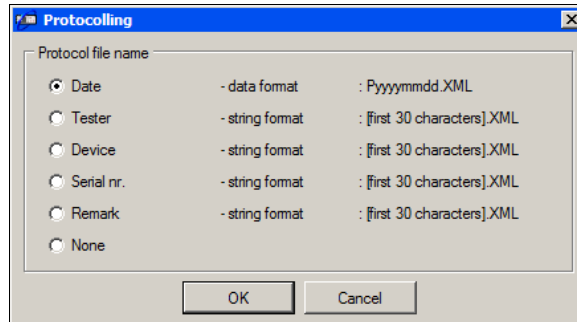


Fig. 35: Dialog window "Protocolling"

After each test the test and measuring results are stored in a protocol file. By pressing button "Change" in the panel "Protocolling" the user can determine the name of the protocol. Fig. 35 shows the dialog window for defining the name.

<i>Item</i>	<i>Function</i>
<i>Date</i>	File name is made up from date of day. The date format is Pyymmdd. E.g. a protocol file created on May 07, 2009 would be filed under P20090507.XML. This pre-setting will be advantageous in as much as a new file is created every day.
<i>Tester</i>	File name is created from the first 30 letters of the operator name
<i>Device</i>	File name is created from the first 30 letters of the name of the device.
<i>Serial nr.</i>	File name is created from the first 30 letters of the serial number.
<i>Remark</i>	File name is created from the first 30 letters of the remark.

3.4.3 Editor: Test steps

The organisation of the test steps and the definition of the test run is managed directly in the main window of the program editor:

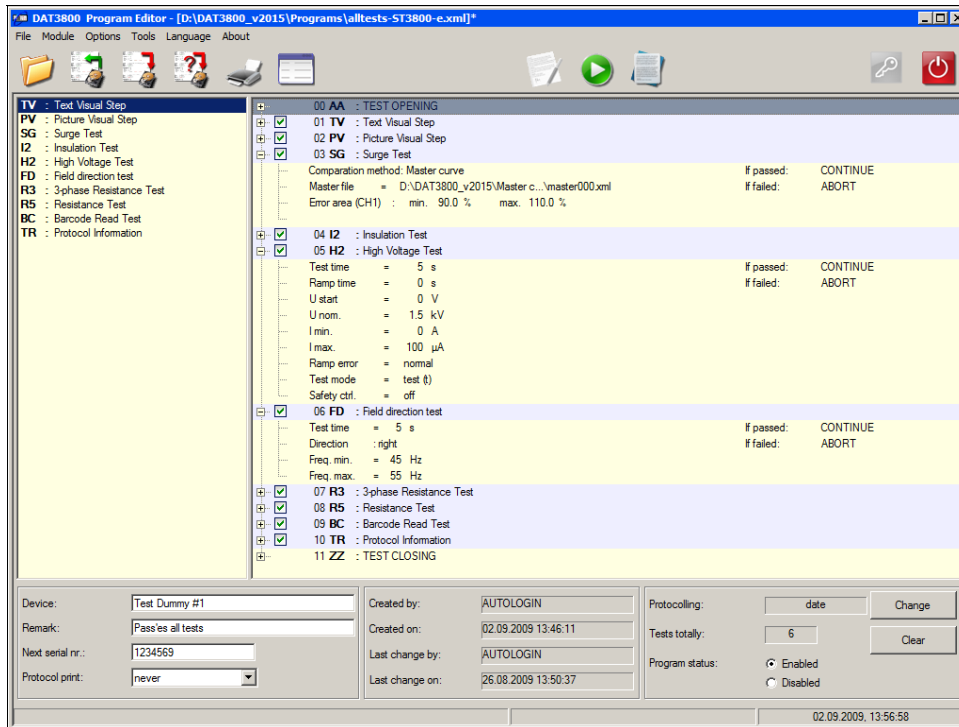


Fig. 36: Editing of a test program

The list window on the left shows all the test steps available in the program. By double-clicking one of these test steps same is included in the test run list. Before inserting a new test step the step after which this new step is to be inserted should be marked in the test run list. With this double click the respective parameter window of the new test step opens automatically, and after editing the parameter and closing the parameter window the new test step will appear at the desired position in the test run list.

The list window on the right shows the current test program. The test steps are shown one by one with step number, grammalogue and definition of each step

Before and after each test run the test steps "TEST OPENING" and "TEST CLOSING" are arranged automatically. This way certain operations can be defined at the beginning and at the end of a test run.

By means of the [+] symbol, each test step can be switched between "brief" and "detailed" display mode. In "brief" mode, only the step number, step type and step name are displayed. When switching a step to "detailed", it is shown with all of its parameters and settings.

To change the order of test steps, each test step can be "dragged" with the mouse to another position in the test program. (Except for the steps "AA" and "ZZ", which cannot be moved.)

Moreover, test steps can be handled in the Windows-typical manner of "cut", "copy" and "paste". To do so, a test step must be marked by the mouse, then one can perform the appropriate action by right-clicking the step, and using the context menu that will pop up.

To save the current test program to the harddisk, choose the menu item "File / Save" or "File / Save as...".

"Save" will just save the file with the current name.

"Save as..." allows to choose another file name.

3.5 Description of Test Parameters

3.5.1 General Information

The test steps have all common dialog elements or test parameters. With the "Text Visual Step" as an example, the make up of the dialog windows for the common test parameters is explained:

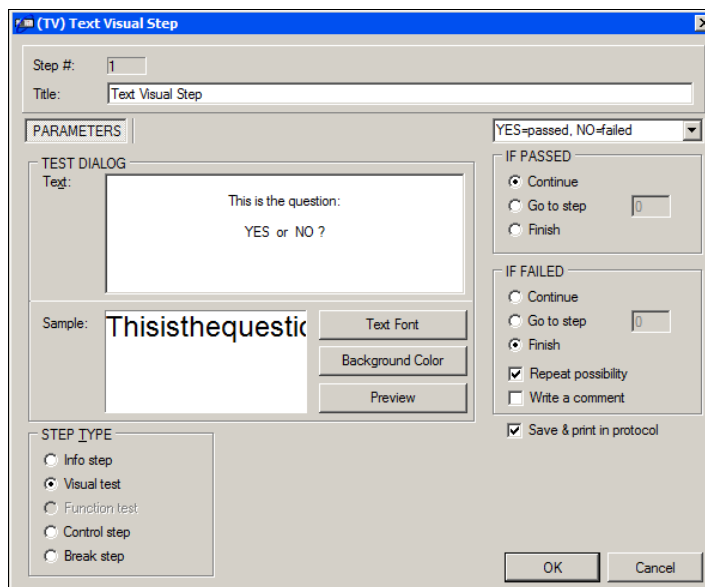


Fig. 37: Dialog window with test parameters (e.g.: 'Text Step')

Common paramters:

<i>Dialog element</i>	<i>Function</i>
Step #	The number of the actual test step in the test program.
Title	Labelling of test step. Display on test run window and during test run. You can also give instructions in the title for the testing person, e.g. »protective wire test at fan/ventilated motor«
Test time	Duration of the test step. (Not available for TV & PV.)
IF PASSED / IF FAILED	It is possible to make "jumps" in the order of the test program, depending on whether the result of a test step was "Passed" or "Failed". The possibilities of branching are: <ul style="list-style-type: none"> - Continue The test run is continued with the next step in the list - Go to A jump is done to the test step entered here. - Finish A jump to the test step "ZZ" is performed. - Repeat possibility If the test step result was "Failed", the tester is asked whether the step shall be repeated. If there's no error during the repeated step, the test result is "Passed." - Write a comment Only for Text & Picture Visual Step - Save & print in protocol Only for Text & Picture Visual Step
OK	Accept all changes and close the parameter window..
Abbruch	Reject all made changes and close the parameter window.

3.5.2 Start of Test (AA)

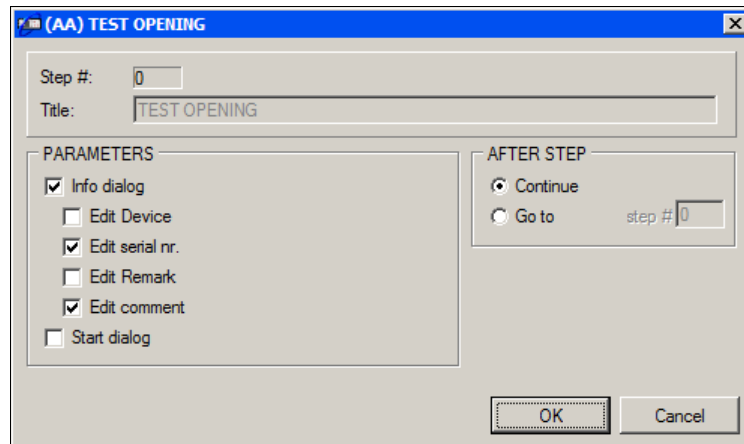


Fig. 39: Test parameters "Start of Test"

When "Info Dialog" is checked, a window with information about the DUT will be shown at the start of each test (see Fig. 57).

If the user shall be able to edit certain DUT data when a test starts, the wished positions have to be checked here.

When "Start dialog" is checked, then at every start of a test run the testing person will be requested to contact the DUT.

3.5.3 Text Visual Step (TV)

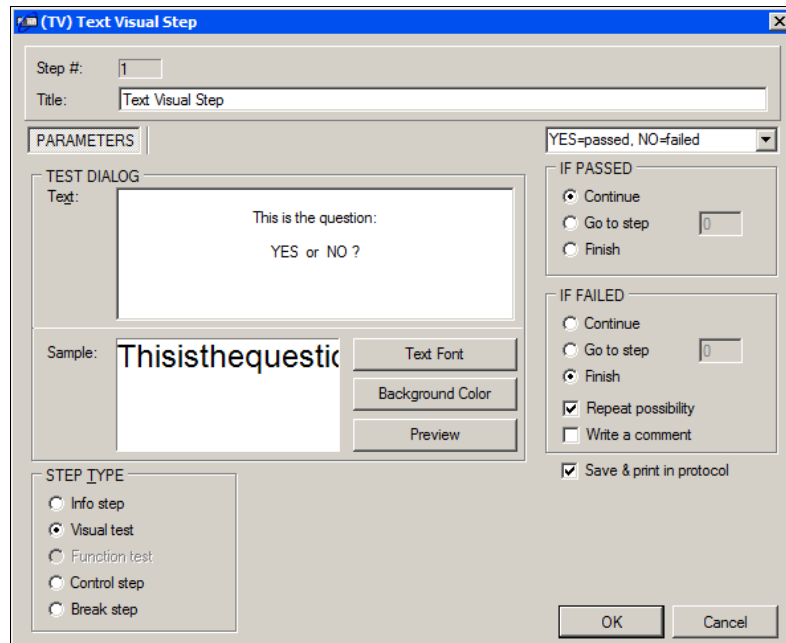


Fig. 40: Test parameters "Text visual step"

In the test step *Text visual step* the dialog elements have below functions:

<i>Dialogelement</i>	<i>Funktion</i>
<i>Text</i>	Contents of this field is displayed when step is executed. The message has to consist of a minimum of five characters.
<i>Text font</i>	Choose the desired font for the message text
<i>Background color</i>	Choose the color of background on which the message will be displayed
<i>Preview</i>	Have a look on how your message will appear
<i>Write a comment</i>	If the test is confirmed with NO, one can enter a comment to the test
<i>Save&print in protocol</i>	If not checked, the test will neither appear in the PKL nor will it be printed
<i>YES=passed, NO=failed</i>	By switching this parameter, it is possible to revert answer logic for certain questions ("Is there smoke coming out of the DUT?" → "NO" → result PASS)

Step type

<i>Info step</i>	Choose this to give any information to the tester. There will only be an OK button to acknowledge the message.
<i>Visual test</i>	Choosing this option forces a dialog to appear which can be quitted with YES or NO. In case of NO, DUT has failed the test
<i>Function test</i>	<i>Not available for device ST3800</i>
<i>Control step</i>	This test type is for information only. No test result PASSED or FAILED.

3.5.4 Picture Visual Step (PV)

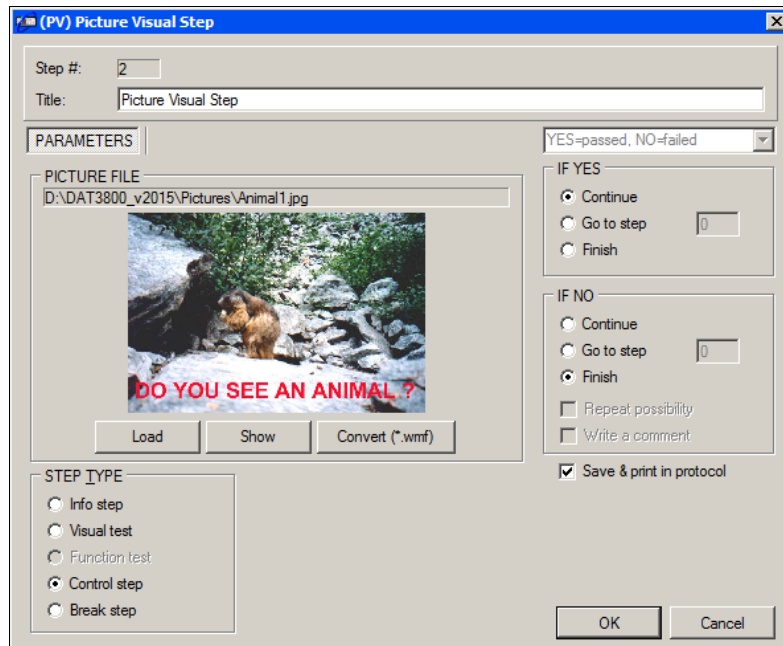


Fig. 41: Test parameters "Picture visual step"

In the test step *Text visual step* the dialog elements have below functions:

<i>Dialog element</i>	<i>Function</i>
Load	Opens standard »open file« dialogue to choose an image
Preview	Have a look on how things will look on paper
Create	If you want to create a new image, you can start the image editor with this button. You can set the path to your preferred image editor in the menu options/environments/editor. The default setting points to MS Paint
Write a comment	If the test is confirmed with NO, one can enter a comment to the test.
Save&orint in protocol	If not checked, the test will neither appear in the PKL nor will it be printed
YES=passed, NO=failed	reverts answer logic (see <i>Text visual step</i>)
Step type	
Info step	Choose this to give any information to the tester. There will only be an OK button to acknowledge the message.
Visual test	Choosing this option forces a dialog to appear which can be quitted with YES or NO. In case of NO, DUT has failed the test
Function test	<i>Not available for device ST3800</i>
Control step	This test type is for information only. No test result PASSED or FAILED.

3.5.5 Surge Test (SG)

Choosing the test step "Surge Test", the following window appears:

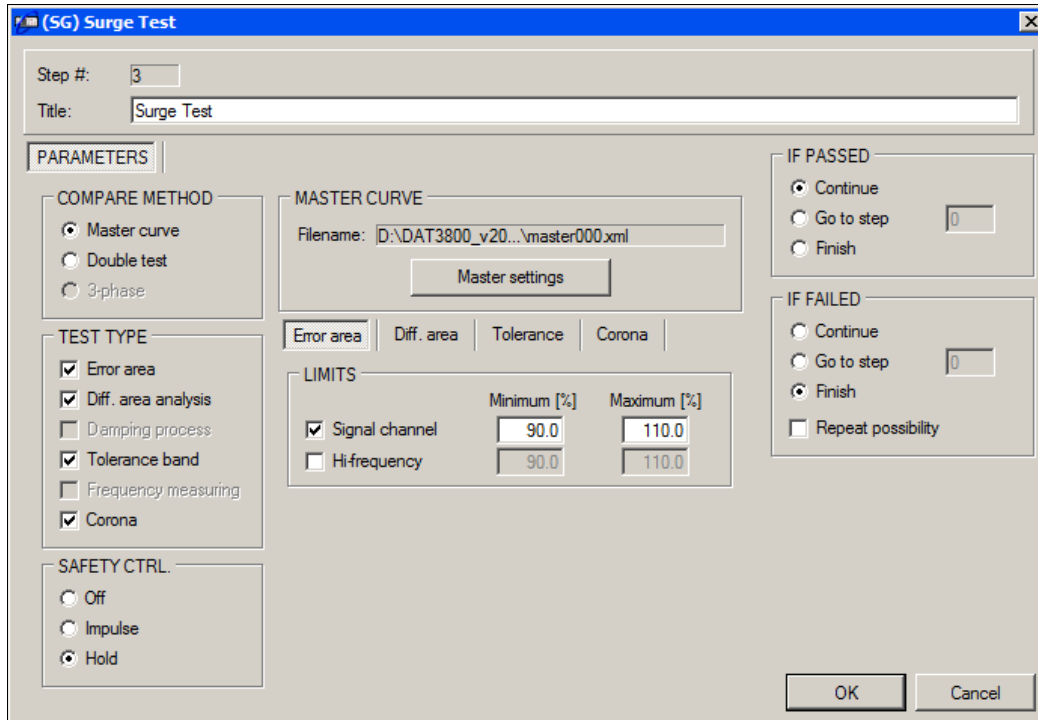


Fig. 42: Test parameters "Surge test"

Dialog element

Function

COMPARE METHOD

- *Master curve*
- *Double test*

Chooses which comparison method to use:

The curve obtained from the DUT is compared with a reference curve

The curves of two different surge impulses are taken from the DUT, and are compared with each other

MASTER CURVE

- *Master settings*

Shows the name of the actually chosen master curve.

Opens the masterkurve editor, with which one can edit existing curves, or record new ones. Please see chapter 2.2, page 14ff.

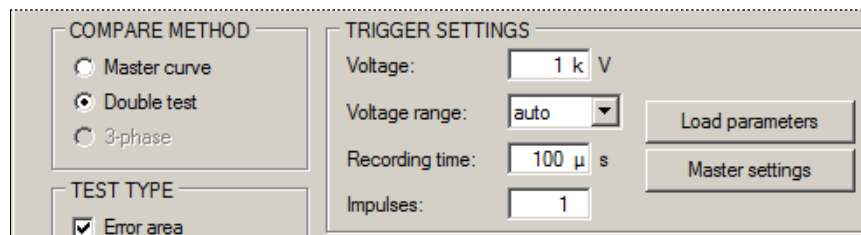


Fig. 43: Method „Double test“

<i>Dialog element</i>	<i>Function</i>
TRIGGER SETTINGS	Settings for the comparison method "Double test":
- Voltage	Sets the test voltage to perform the surge test with. Possible settings are 500 – 6000 V.
- Voltage range	Sets the voltage range for the scale of the diagram (y-axis). This can be either done automaticall (acc. to the chosen test voltage), or can be set manually. For some DUTs it may be necessary to chose the range manually, in order to get an optimal display.
- Recording time	Sets the range of the x-axis.
- Impulses	With "Envelope", the distance between master curve and tolerance band is defined. "Tolerance" then defines how much of all measured samples are allowed to be located outside of the tolerance band.
- Load parameters	A previously saved data set for "Trigger settings" can be loaded here
- Master settings	Opens the masterkurve editor, with which one can edit existing curves, or record new ones. Please see chapter 2.2, page 14ff.

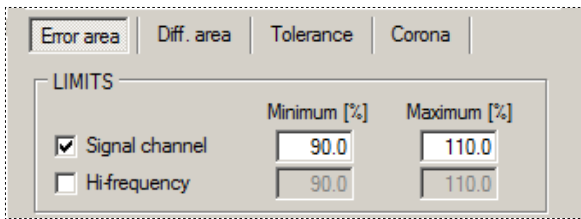


Fig. 44: Tab "Error area"

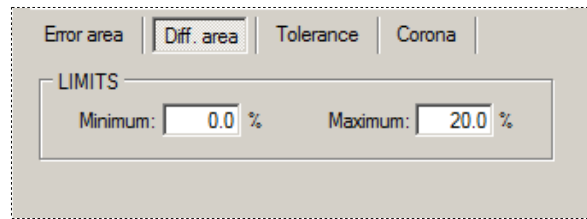


Fig. 45: Tab "Diff. area"

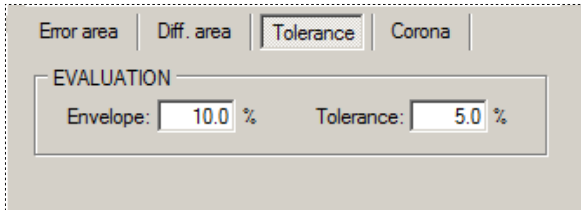


Fig. 46: Tab "Tolerance"

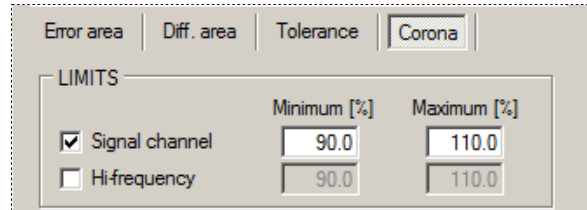


Fig. 47: Tab "Corona"

Dialog element**Function****TEST TYPE**

Chooses the method of curve evaluation:

- Error area

The values "minimum" and "maximum" define how big the area (i.e. the integral) of the measured curve is allowed to be, in percentual relation to the master curve.

- Diff. area analysis

The values "minimum" and "maximum" define, how big the difference area between master curve and measured curve may be, in percentual relation to the master curve.

- Tolerance band

With "Envelope", the distance between master curve and tolerance band is defined. "Tolerance" then defines how much of all measured samples are allowed to be located outside of the tolerance band.

- Corona

This method evaluates the high-frequency component (noise) of the input signal. Thresholds are given in percent, relative to the mean absolute signal amplitude.

3.5.6 Insulation Test (I2)

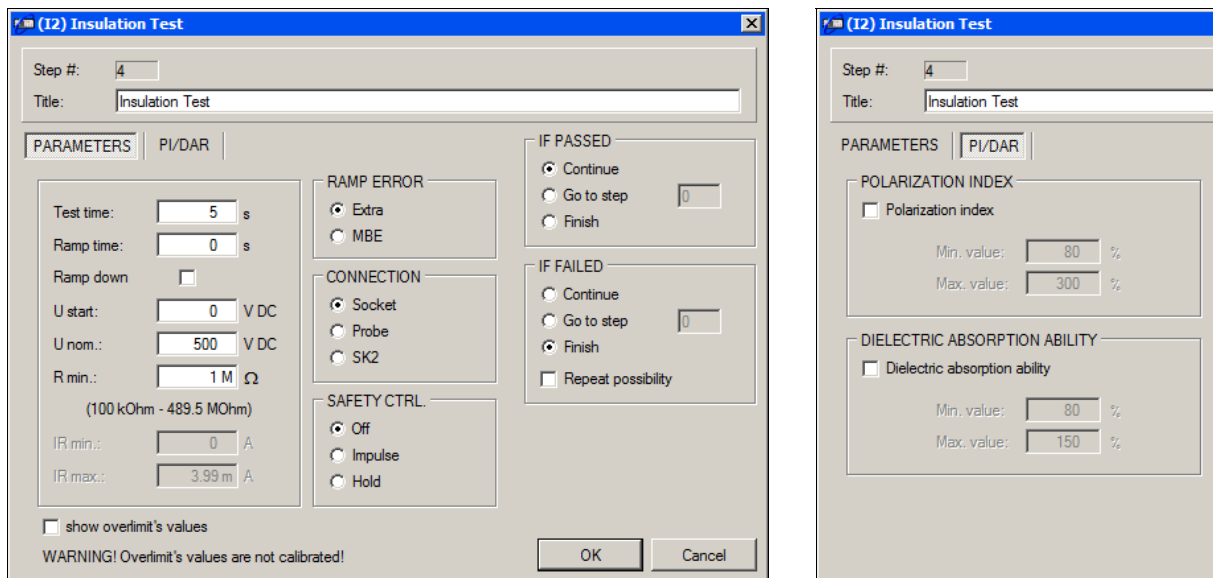


Fig. 48: Test parameters "Insulation test"

This is the test setup for the Insulation test of ST3800 devices:

<i>Dialog element</i>	<i>Function</i>
-----------------------	-----------------

Ramp time	Time for voltage to be increased to maximum (when "0", no ramp is used)
Ramp down	If checked, at the test's end the voltage will be decreased instead of just switched off (same time as for "ramp up")
U start	Starting value of test voltage when voltage ramp is used
U nom	Nominal value of test voltage
R min	Minimum of required resistance for test result "PASS"
Ramp error	(see also device 3800 operating manual)
Extra / MBE	Type of current control during voltage ramp: software control (Extra) or hardware control (MBE)
Safety control	Method of safety control resp. test starting:
Off	Off = test is started immediately
Impulse	Impulse = test starts with simple activating of start signaller
Hold	Hold = start signaller must be hold until the test step has finished

3.5.7 High Voltage Test (H2)

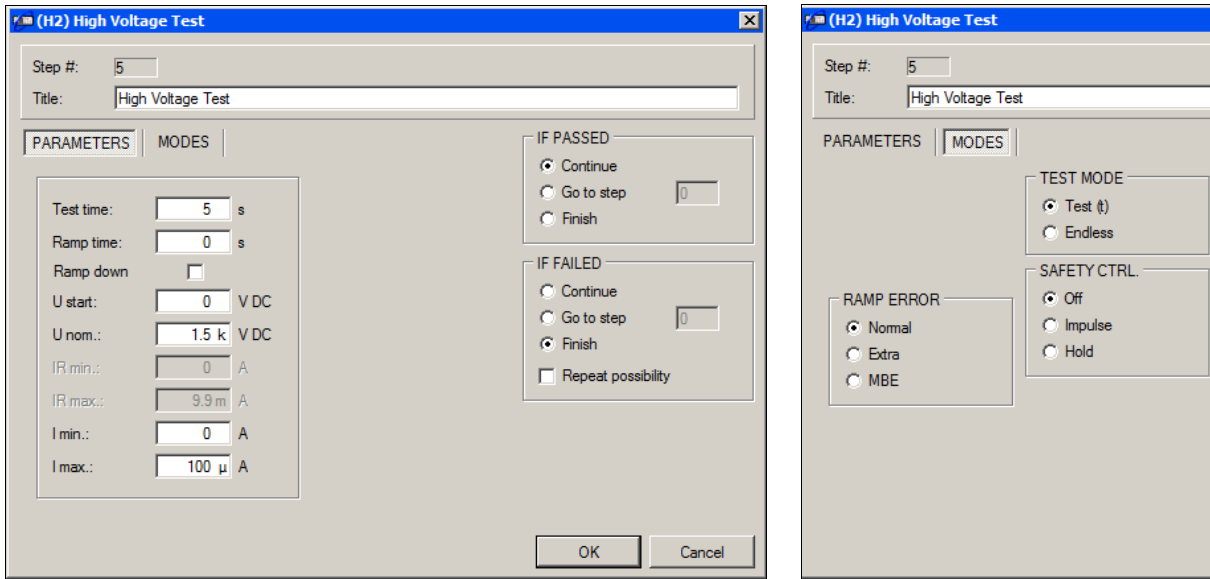


Fig. 49: Test parameters "High voltage test"

This is the setup for the High Voltage test of ST3800 devices:

<i>Dialog element</i>	<i>Function</i>
-----------------------	-----------------

<i>Ramp time</i>	Time for voltage to be increased to maximum (when »0«, no ramp is used)
<i>Ramp down</i>	If checked, at the test's end the voltage will be decreased instead of just switched off (same time as for "ramp up")
<i>U start</i>	Starting value of test voltage when voltage ramp is used
<i>U nom</i>	Nominal value of test voltage
<i>IR min</i>	Minimum of allowed current during volt.ramp (only active when <i>Ramp error</i> = <i>Extra</i>)
<i>IR max</i>	Maximum of allowed current during volt.ramp (only active when <i>Ramp error</i> = <i>Extra</i>)
<i>I min</i>	Minimum of allowed current during actual test
<i>I max</i>	Maximum of allowed current during actual test
<u>Ramp error</u>	(see also device 3800 operating manual)
<i>Normal / Extra / MBE</i>	Type of current control during voltage ramp: software control (Extra) or hardware control (Normal or MBE)
<u>Test mode</u>	
<i>Test (t)</i>	test ends when [test time] has expired
<i>Endless</i>	test is running endlessly (must be aborted manually)
<u>Safety Control</u>	<i>Method of safety control resp. test starting:</i>
<i>Off</i>	Off = test is started immediately
<i>Impulse</i>	Impulse = test starts with simple activating of start signaller
<i>Hold</i>	Hold = start signaller must be hold until the test step has finished

3.5.8 FD: Field Direction test

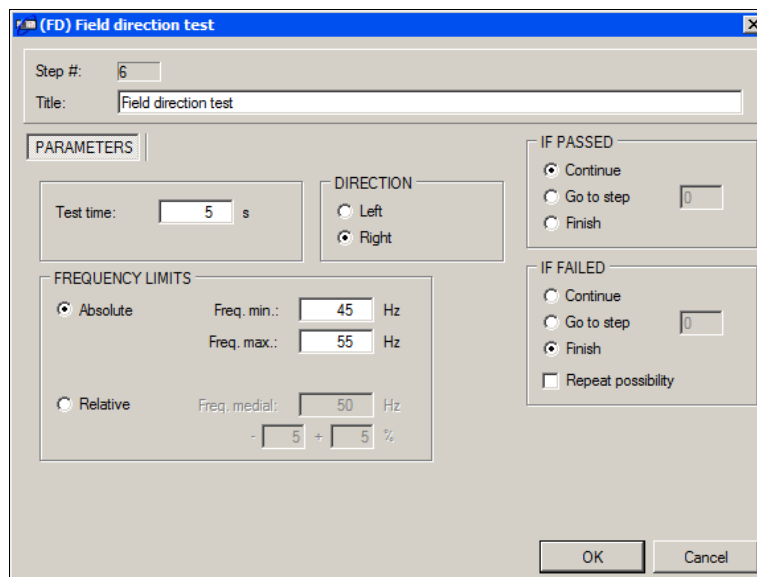


Fig. 50: Test parameters "Field direction test"

This is the dialog for the field direction test of ST3800 devices:

<i>Dialog element</i>	<i>Function</i>
<i>Test time</i>	Time for the FD test to be performed
<i>Direction</i>	Chooses the expected field direction: either "Left" or "Right"
<i>Frequency Limits</i>	<i>Setting of limits for the rotary frequency</i>
<i>Absolute</i>	<i>Uses absolute limits:</i>
<i>Freq. min</i>	Minimum of required frequency
<i>Freq. max</i>	Maximum of accepted frequency
<i>Relative</i>	<i>Uses relative limits:</i>
<i>Freq. medial</i>	Sets the average wanted frequency
-	Maximum allowed negative deviation of <i>f</i>
+	Maximum allowed positive deviation of <i>f</i>

3.5.9 R3: 3-phase Resistance Test

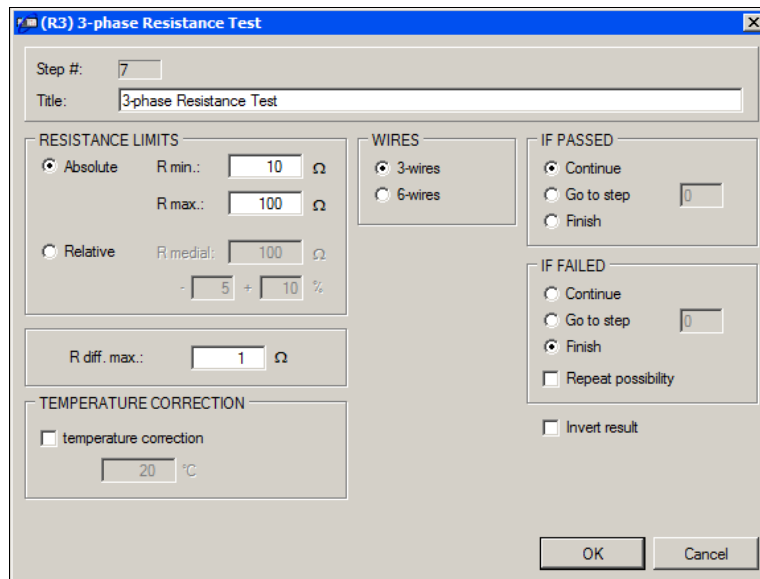


Fig. 51: Test parameter "3-phase resistance test" (R3)

This is the dialog for the 3-phase resistance test:

<i>Dialog element</i>	<i>Function</i>
<u>Wires</u>	Specifies whether the measurement is done by separate sense-lines ("6-wires"), or conventionally over the source lines ("3-wire"). (The contacting of the DUT has to be done in accordance to this setting!)
<u>Absolute</u>	Uses absolute thresholds for resistance:
<i>R min</i>	Minimum allowed resistance
<i>R max</i>	Maximum allowed resistance
<u>Relative</u>	Uses relative thresholds for resistance:
<i>R medial</i>	Sets the average expected resistance
-	max.allowed percentual deviation, negative
+	max. allowed percentual deviation, positive
<i>R diff. max.</i>	Maximum allowed difference between the resistances obtained from each wire
<i>Temperature Correctn.</i>	When chosen, the obtained resistance is normalized to a standard temperature (usually 20°C).

3.5.10 R5: Resistance Test

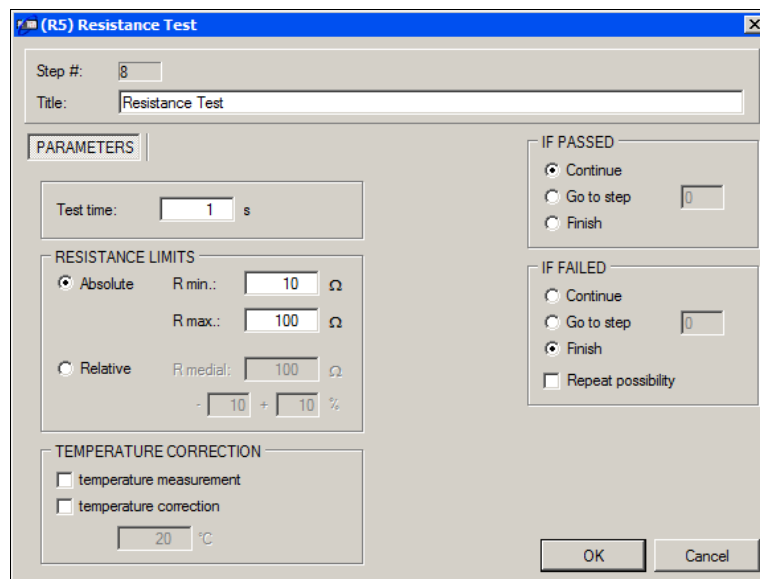
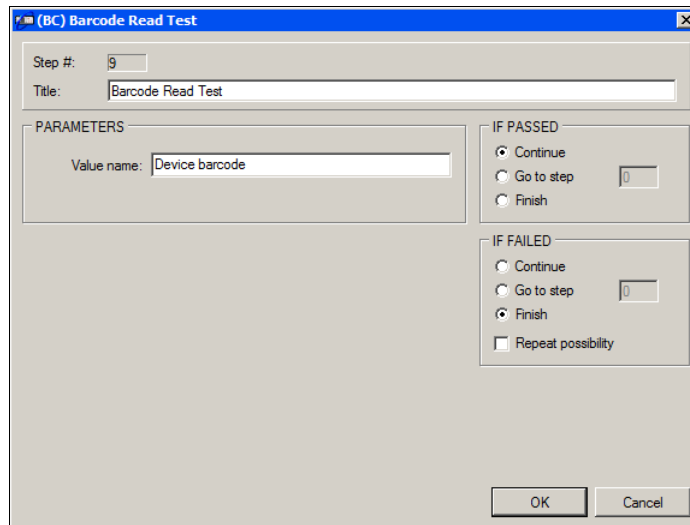


Fig. 52: Test parameters "Resistance Test" (R5)

This is the dialog for the test step "Resistance Test":

<i>Dialog element</i>	<i>Function</i>
<u>Absolute</u>	<i>Uses absolute thresholds for resistance:</i>
<i>R min</i>	Minimum allowed resistance
<i>R max</i>	Maximum allowed resistance
<u>Relative</u>	<i>Uses relative thresholds for resistance:</i>
<i>R medial</i>	Sets the average expected resistance
-	max.allowed percentual deviation, negative
+	max. allowed percentual deviation, positive
<i>temperature measurement.</i>	This option has to be chosen for the very first resistance test within a test program, in order to record the actual environmental temperature.
<i>temperatur correction</i>	When chosen, the obtained resistance is normalized to a standard temperature (usually 20°C).

3.5.11 BC: Barcode Read Test



The screenshot shows a dialog box titled "(BC) Barcode Read Test". It contains the following fields and options:

- Step #: 9
- Title: Barcode Read Test
- PARAMETERS: Value name: Device barcode
- IF PASSED:
 - Continue
 - Go to step 0
 - Finish
- IF FAILED:
 - Continue
 - Go to step 0
 - Finish
 - Repeat possibility
- Buttons: OK, Cancel

Fig. 53: Test parameters "Protocol information"

By means of this test step, it is possible to read the barcode from a DUT during a running test program.

The read barcode will be included in the test protocol.

3.5.12 Protocol Information (TR)

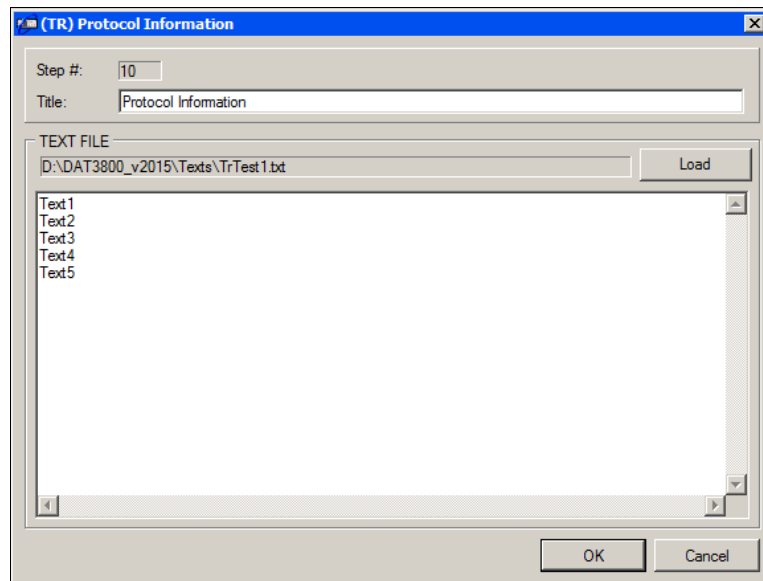


Fig. 54: Test parameters "Protocol information"

The step "Protocol Information" is not really a "test", its purpose is to insert any text into the test protocol.

To do so, one has to create the desired text with an editor (e.g. Windows-Notepad), and save it to a suited location, e.g. C:\ST3800\TXT.

At that place of the test sequence, where the text shall appear in the protocol, one adds the step "protocol information". By means of the button *load*, one selects the desired *.txt file.

Its content is then displayed in the preview window.

3.5.13 ZZ: Test Closing

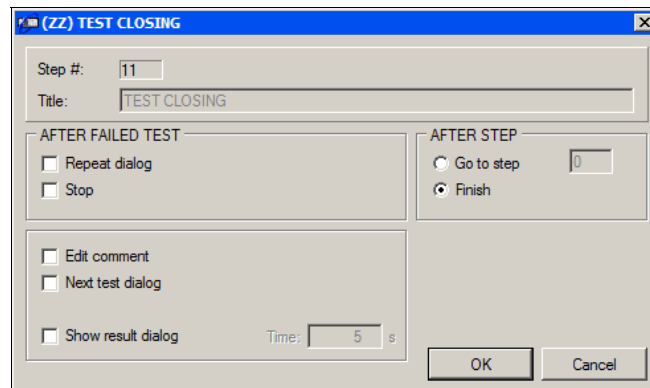


Fig. 55: Parameters of "Test closing" (ZZ)

Here the following adjustments can be made:

<i>Dialog element</i>	<i>Function</i>
<i><u>After Failed Test</u></i>	Decision what to do in case of test result "Fail":
<i>Repeat dialog</i>	When checked, the possibility to repeat the whole test is given. In case of test repetition, the serial number is not auto-incremented.
<i>Stop</i>	If the test run failed, the testing cycle is stopped.
<i>Edit comment</i>	Offers the possibility to edit the "comment" field after a test run.
<i>Next test dialog</i>	After the test run a message box is displayed asking if there is another test to perform.
<i>Wait time</i>	Waiting time in seconds in which the result of the test is displayed.

3.6 Program Module "Testing"

When programming of the test programs has been done, testing can begin. With the button "Test" or with the key "F1" one gets to the menu *Test*. Now it is possible to work with either the actual program (as shown in the window's title bar), or to load a different program by means of the button "Load".

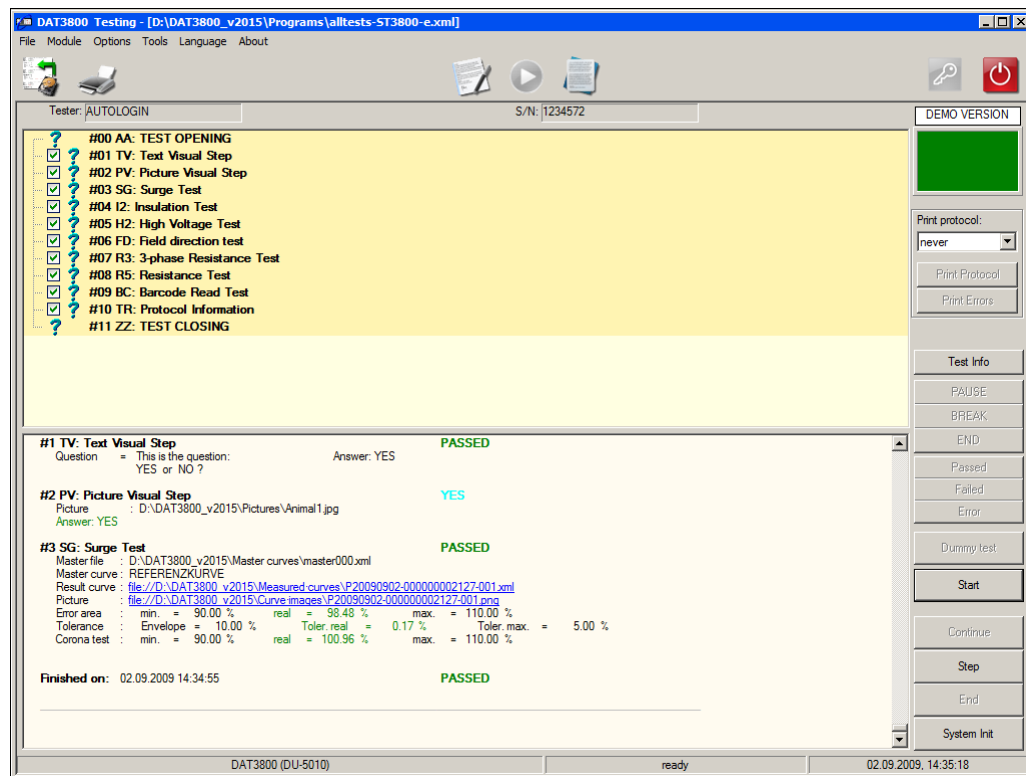


Fig. 56: Program Module "Testing"

In the upper part, the test program with its single test steps is shown.

To the upper right, the symbolized „Signal lamps“ show the actual testing status:

yellow – test is running

green – test result "Passed"

red – test result "Failed"

On the right side, there are the functional buttons to control the test runs:

Start – This will start the automatic test run. All test steps of the program are executed sequentially.

Step – With this button, a "single step" test can be performed. When this button is used, only the test step that's next in order will be executed. After that step has finished, the test run is paused, until the next step is called by using the "Step" button once more.

Continue – If a test had been begun in "Step" mode, it is possible to switch to automatic mode again by using the "Continue" button.

System init – this will initialize all connected hardware devices. If the Initialization returns an error, the test program can not be started.

3.6.1 Start of Test

When pushing the "Test info" button, or when "Info dialog" has been set in test step AA, the below window will open. According to the settings in step AA, the contents of the fields can be edited.

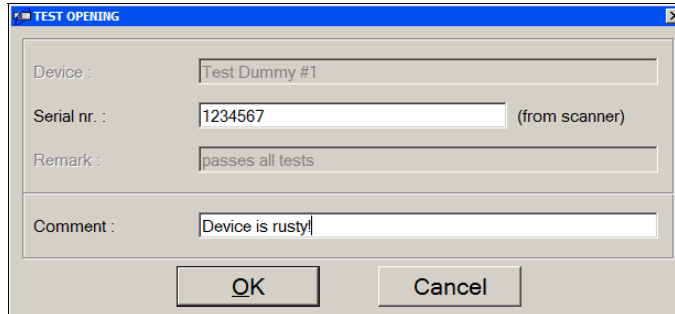


Fig. 57: Dialog window at the start of a test

3.6.2 Testing mode "Step"

The step by step test run guarantees that the DUT can e.g. be connected somewhere else or that changes can be made between the test steps. When starting the test, the Start Window will appear first in which - if it was activated in the editor - you can make changes. Then you are asked to connect the DUT. Each test step has to be started with the key "Step". (A similar effect can be achieved by means of "info steps").

3.6.3 Faulty DUT

A faulty DUT is shown on the monitor by the message "FAILED". At the tester the red lamp "fail" lights up (via ext. I/O of device, too) as long as the test is re-started or until the device is initialized again.

If "repetition possibility" was chosen for failed test steps (see chapter 3.5.1), this dialog appears:

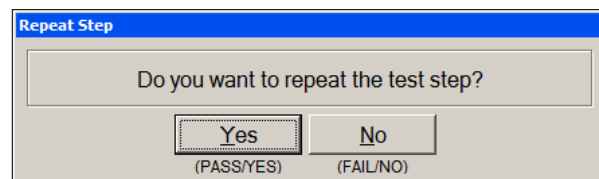


Fig. 58: Dialog "Repeat Step"

3.6.4 Error-free DUT

If there is no error during the test then the message "PASSED" appears on the monitor and the green lamp "pass" lights up (via ext. I/O of device, too) as long as the test is re-started or the device is initialized again.

3.6.5 Interrupting a running test

A running test can be interrupted by means of the buttons "Pause" and "Break". Test steps that open a new window (i.e. Text or Picture steps) do have their own Pause/Break buttons. Tests that are displayed directly in the main window of the Testing module, these buttons are located at the right side of the main window.)

"PAUSE" interrupts the test run **after** the current test step has been ended.

"BREAK" will abort a running test **immediately**.

A stopped test can either be cancelled with "End" or picked up with "Continue".

The status of the test is displayed in the upper text field. Here is displayed whether the test result was pass or fail or if the test was stopped.

3.7 Program Module "Results"

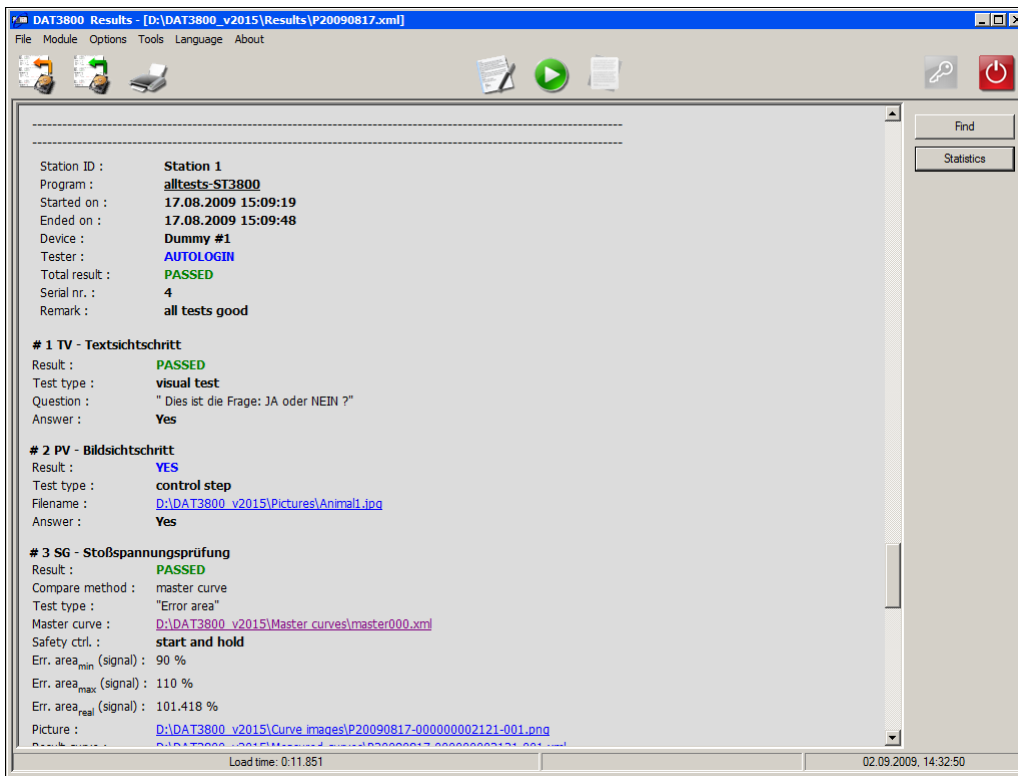


Fig. 59: Main window of the "Results" module

- Via the menu *File / Load results*, already existing protocol files can be loaded. One can also load test programs via *File / Load program*, to view program related test statistics.
- The protocol can be printed via *File / Print*.
- The button "Statistics" opens a new window, showing the overall statistics of the loaded test program or results protocol:

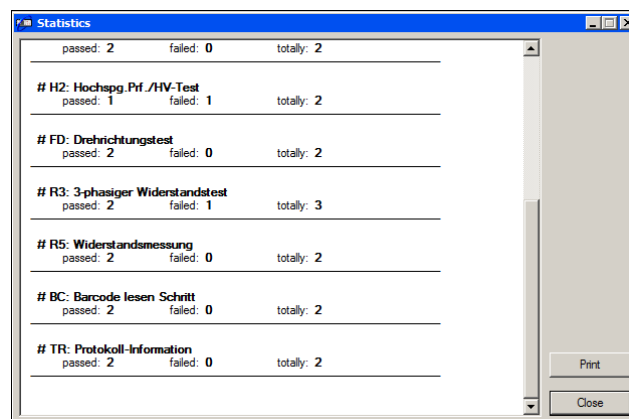


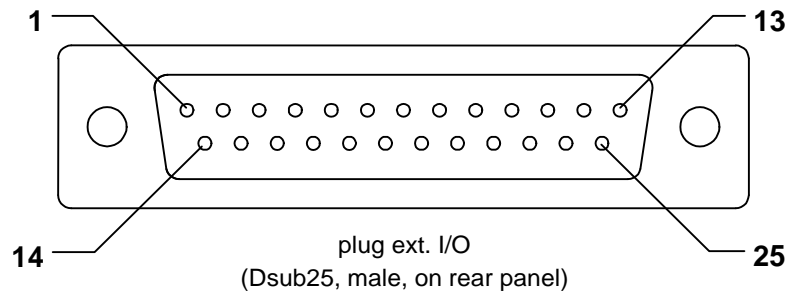
Fig. 60: Dialog window "Statistics"

By means of the "Windows Clipboard", marked parts of these display windows can be transferred to another Windows application – e.g. Microsoft Word®. By doing so, you can record the protocols in a documentation.

Annex

A Interface configuration

A-1 External I/O-Interface X1

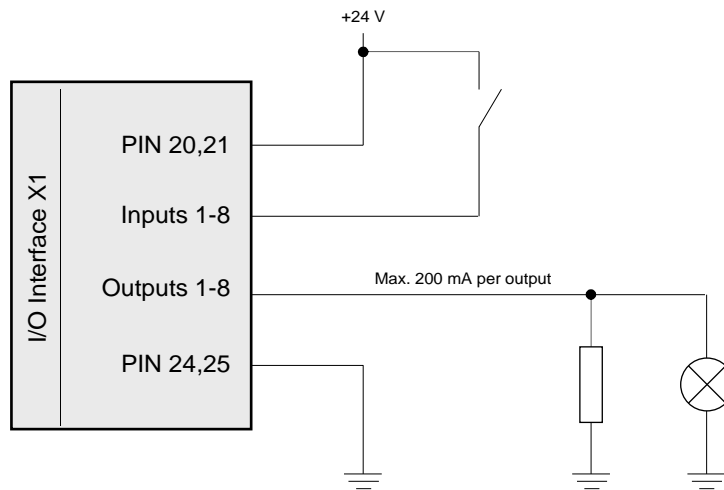


PIN	Signal	Assignment
1	Output 1	EXT_PASS
2	Output 2	EXT_FAIL
3	Output 3	EXT_BUZZER
4	Output 4	EXT_TEST (active during complete test)
5	Output 5	<i>not used</i>
6	Output 6	<i>not used</i>
7	Output 7	<i>not used</i>
8	Output 8	<i>not used</i>
9	Analog Input ¹⁾	
10	—	<i>not used</i>
11	Input 1	<i>not used</i>
12	Input 2	<i>not used</i>
13	Input 3	<i>not used</i>
14	Input 4	<i>not used</i>
15	Input 5	<i>not used</i>
16	Input 6	<i>not used</i>
17	Input 7	<i>not used</i>
18	Input 8	<i>not used</i>
19	—	<i>not used</i>
20	+24 V DC	ext. voltage against ground ²⁾
21	+24 V DC	ext. voltage against ground ²⁾
22	—	<i>not used</i>
23	—	<i>not used</i>
24	GNS	ground
25	GNS	ground

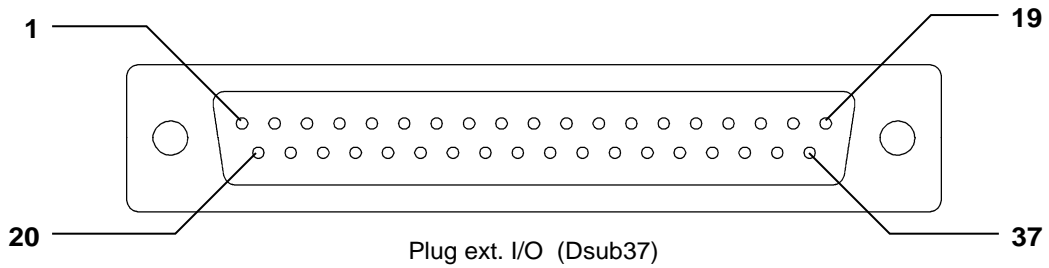
¹⁾ electrically insulated from internal supply

²⁾ has to be fed in from external

Basic circuit for I/O interface X1:



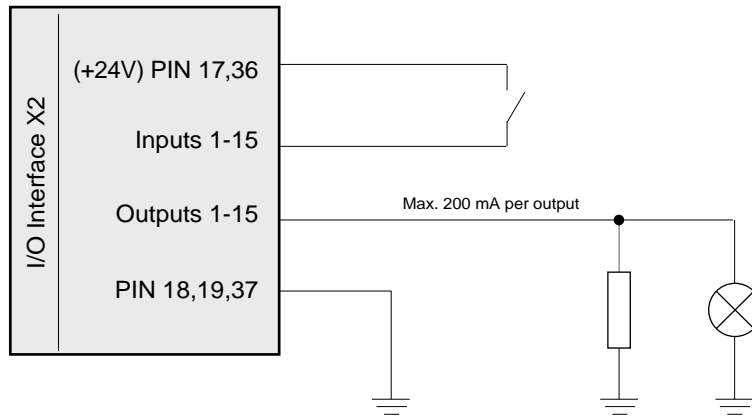
A-2 Second external I/O Interface X2



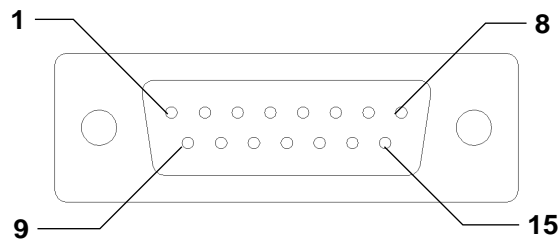
PIN	Signal	Assignment
1	Output 1	<i>not used</i>
2	Output 3	<i>not used</i>
3	Output 5	<i>not used</i>
4	Output 7	<i>not used</i>
5	Output 9	<i>not used</i>
6	Output 11	<i>not used</i>
7	Output 13	<i>not used</i>
8	Output 15	<i>not used</i>
9	Output 17	<i>not used</i>
10	Output 19	<i>not used</i>
11	Output 21	<i>not used</i>
12	Output 23	<i>not used</i>
13	N.C.	<i>not used</i>
14	N.C.	<i>not used</i>
15	N.C.	<i>not used</i>
16	N.C.	<i>not used</i>
17	+24 V DC	Voltage against ground ^{*)}
18	Gnex	
19	Gnex	
20	Output 2	<i>not used</i>
21	Output 4	<i>not used</i>
22	Output 6	<i>not used</i>
23	Output 8	<i>not used</i>
24	Output 10	<i>not used</i>
25	Output 12	<i>not used</i>
26	Output 14	<i>not used</i>
27	Output 16	<i>not used</i>
28	Output 18	<i>not used</i>
29	Output 20	<i>not used</i>
30	Output 22	<i>not used</i>
31	Output 24	<i>not used</i>
32	N.C.	<i>not used</i>
33	N.C.	<i>not used</i>
34	N.C.	<i>not used</i>
35	N.C.	<i>not used</i>
36	+24 V DC	Voltage against ground ^{*)}
37	GNex	

^{*)} internally generated voltage (**not** needed to be fed in externally)

Basic circuit for interface X2:



A-3 Analog Interface X3



PIN	Signal	Assignment
1		Analog in
2		Analog out
3		24 V intern
4		24 V intern
5		
6		
7		24 V extern
8		24 V extern
9		0 V intern
10		0 V intern
11		0 V intern
12		
13		0 V extern
14		0 V extern
15		0 V extern

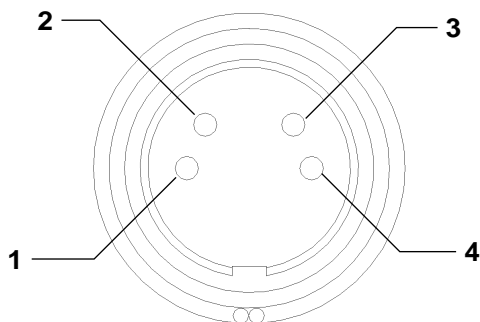
Note:

To use this interface without external supply, the PINs have to be bridged: 3+4 against 7+8, and 9+10+11 against 13+14+15.

When this interface is used with internal supply only, then no devices with a current consumption bigger than 0.5 A should be attached.

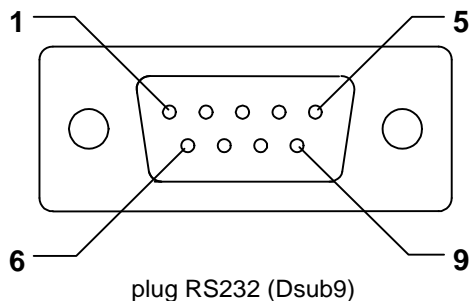
When the interface is used with external supply, attached devices may have a current consumption up to 3 A.

A-4 Connector for sensor of direction of rotation X4



PIN	Signal	Assignment
1		+ 12 V
2		GND
3		Sensor left
4		Sensor right

A-5 Serial RS-232 Interface X5



PIN	Signal	Assignment
1		
2		
3		
4		
5		
6		
7		
8		
9		

Interface configuration: 9600 baud, 8 data bits, 1 stopbit, no parity.

B Technical Data

Technical data of ST3800:

Dimensions:	H*W*D : 5 HU (222mm) * 19" * 440 mm	
Power supply:	230 VAC +/- 10 % / 50 Hz	
Power fuse:	2 x 2 As (slow)	
Power consumption:	max. 380 VA	
Fan :	built-in (blowing out through rear panel)	
Environmental conditions:	temperature:	+15°C – +40°C
	humidity:	max. 70% (not condensating)
	storage temp.:	+5°C – 50°C

Testing part:

High voltage source:	programmable 100 V – 6000 V DC precision, reproduceability: ± 1 % of max. dial
Surge capacity:	40 nF
AD converter for surge curve recording:	8 bit / 50 MHz

C Terms of warranty

1. Warranty period

The warranty period is 12 months after delivery.

2. Conditions for a guarantee claim

- The Surge tester ST 3800 must have been put into operation by qualified personnel at the customer's.
- Inspections must be carried out regularly (once a year) and thoroughly (by SPS electronic GmbH).
- Defective or worn parts have to be replaced immediately. The operation of such parts is forbidden for safety reasons.
- Defective parts, subject to guarantee claims, have to be sent to SPS electronic GmbH for inspection.
- Defects occurred must be reported to SPS electronic GmbH immediately.

3. Beginning of guarantee period

Guarantee starts with the date of delivery note.

4. Guarantee

SPS electronic GmbH guarantees a good function of the high voltage test device, a conscientious and professional design and manufacture as well as the use of high-quality material.

All parts are being replaced free of charge if parts became defective or useless during the guarantee period due to the use of inadequate material, manufacturing faults or an imperfect engineering.

5. Excluded from guarantee

- Damages due to outside influences, above all because of handling malpractices or of local conditions.
- Damages at devices from which the serial number has been removed, destroyed or falsified.
- Wear parts, such as fuses, signal bulbs, etc.