Manual

Smart Safety Test®

SILworX

All technical specifications and notes in this manual have been written with great care and effective quality assurance measures have been implemented to ensure their validity. For questions, please contact HIMA directly. HIMA appreciates any suggestion on which information should be included in the manual.

Equipment subject to change without notice. HIMA also reserves the right to modify the written material without prior notice.

All the current manuals can be obtained upon request by sending an e-mail to: documentation@hima.com

Contact

HIMA Paul Hildebrandt GmbH

P.O. Box 1261

68777 Brühl, Germany

Phone: +49 6202 709-0

Fax: +49 6202 709-107

E-mail: info@hima.com

Document Designation	Description
HI 801 494 D, Rev. 11.00 (1922)	German original document
HI 800 495 E, Rev. 11.00.00 (1924)	English translation of the German original document

Inhaltsverzeichnis

1 Introduction

This manual describes the configuration of Smart Safety Test in SILworX for controllers of the system families HIMax, HIMatrix and HIQuad X.

1.1 Structure and Use of This Manual

The manual contains the following chapters:

- Introduction and writing conventions
- Safety
- Product description
- Example of a Smart Safety Test

Additionally, the following documents must be taken into account:

Name	Content	Document no.
HIMax system manual	Hardware description of the HIMax system	HI 801 001 E
HIMax safety manual	Safety functions of the HIMax system	HI 801 003 E
HIMatrix safety manual	Safety functions of the HIMatrix system	HI 800 023 E
HIMatrix compact system manual	Hardware description of the HIMatrix system	HI 800 141 E
HIMatrix modular system manual	Hardware description of the HIMatrix system	HI 800 191 E
HIQuad X system manual	Hardware description of the HIQuad X system	HI 803 211 E
HIQuad X safety manual	Safety functions of the HIQuad X system	HI 803 209 E
Communication manual	Description of the communication protocols.	HI 801 101 E
Getting started with SILworX	Introduction to SILworX.	HI 801 103 E

The current manuals can be obtained upon request by sending an e-mail to: documentation@hima.com. For registered HIMA customers, the product documentation is available at https://www.hima.com/en/downloads/.

1.2 Target Audience and Required Competence

This document is aimed at the planners, design engineers and programmers of automation systems as well as the persons authorized to start up, operate and maintain the devices and systems concerned. Specialized knowledge of safety-related automation systems is required.

All specialist staff (planning, installation, start-up) must be instructed concerning the risks and the associated possible consequences which can arise as a result of modifications to a safety-related automation system.

The operator is responsible for qualifying the operating and maintenance personnel and providing them with appropriate safety instructions.

1.3 Writing Conventions

To ensure improved readability and comprehensibility, the following writing conventions are used in this document:

Format	Description
Bold	To highlight important parts. These are names of buttons, menu functions and tabs that can be clicked and used in SILworX.
Italics	Parameters, system variables and references to other text passages.
`Courier`	Literal user input or displays which are identical to the printed value.
RUN	Operating states are designated by capitals.
Chapter 1.2.3	Cross-references to other chapters. They are implemented as hyperlinks. Click the hyperlink to jump to the referenced position in the document.

Safety notices and operating tips are specially marked.

1.3.1 Safety Notices

The safety notices must be strictly observed to ensure that the risk to which users are exposed is as low as possible.

The safety notices are represented as follows:

- Signal word: warning, caution
- Type and source of risk.
- Consequences arising from non-observance.
- Risk prevention.

SIGNAL WORD
Type and source of risk!
Consequences arising from non-observance
Risk prevention

Meaning of the signal words:

- **Warning:** It indicates hazardous situations which, if not avoided, could result in death or serious injury.
- **Caution:** It indicates hazardous situation which, if not avoided, could result in minor or moderate injury.

1.3.2 Operating Tips

Additional information is structured as follows:

The text giving additional information is located here.

2 Safety

All safety information, notes and instructions specified in this document must be strictly observed. The HIMA controllers may only be used if all guidelines and safety instructions are adhered to.

The HIMA controllers are operated with SELV or PELV. No imminent risk results from the controllers themselves. Use in the Ex zone is only permitted if additional measures are taken.

2.1 Intended use

To use the HIMA controllers, all pertinent requirements must be met, see Chapter 1.1 in the corresponding manuals.

2.2 Residual Risk

No imminent risk results from a HIMA system itself.

Residual risk may result from:

- Faults related to engineering.
- Faults in the user program.
- Faults related to the wiring.

2.3 Safety Precautions

Observe all local safety requirements and use the protective equipment required on site.

2.4 Emergency Information

A HIMA system is a part of the safety equipment of an overall system. If the controller fails, the system enters the safe state.

In case of emergency, no action that may prevent the HIMA systems from operating safely is permitted.

2.5 Automation security in HIMA systems

Industrial controllers must be protected against IT-specific problem sources. Those problem sources are:

- Attackers inside and outside of the customer's plant
- Operating failures
- Software failures

All requirements for protection against manipulation specified in the safety and application standards must be met. The operator is responsible for authorizing personnel and implementing the required protective actions.

WARNING
Protect the controller against unauthorized access!

Careful planning should identify the measures to implement. The required measures are to be implemented after the risk analysis is completed. Such measures can include:

- Meaningful allocation of user groups.
- Maintained network maps help to ensure that secure networks are permanently separated from public networks and, if required, only a well-defined connection exists (e.g., via a firewall or a DMZ).
- Use of appropriate passwords.

A periodical review of the security measures is recommended, e.g., every year.

The user is responsible for implementing the necessary measures in a way suitable for the plant!

For further details, refer to the HIMA automation security manual (HI 801 373 E).

3 Smart Safety Test

The Smart Safety Test can be used to verify and evaluate the user program or parts of it to see whether the defined requirements are met. Afterwards, a PDF report can be created for documenting the test results.

The Smart Safety Test is created in the PADT and can be executed offline or online. The Smart Safety Test is interference-free, meaning that the user program code and the CRC remain unchanged.

Use of the Smart Safety Test is especially recommended for tests of the user program which must be carried out repeatedly with a great deal of effort.

For inspection authorities, the Smart Safety Test can be used to provide proof that changes in the user program have not caused any new errors in the previously tested parts of the user program.

The underlying technique of the Smart Safety Test is the forcing of global variables. Default values can thus be set and expected result values read and checked by the PADT.

License Option

The Smart Safety Test requires a software activation code.

For further details, refer to the HIMA website www.hima.com -> Products & Services -> Product Registration -> Options SILworX

Follow the registration steps on the HIMA website to get the activation code.

DEMO License

Without the Smart Safety Test license option, exactly 1 DEMO test plan can be created and processed.

Restrictions of the DEMO test plan are:

- A DEMO test plan can contain a maximum of 50 test plan elements.
- A DEMO test plan cannot be expanded, regardless of the existence of the Smart Safety Test license option.
- The Paste Element action is disabled in the context menu when the items in the clipboard come from a DEMO test plan.
- The item *Save Table Contents as CSV* is disabled in the context menu.
- Each page of the PDF report is covered by the word *Demo* (similar to a watermark).

3.1 Creating Smart Safety Test

Exactly one Smart Safety Test folder can be created per resource. This folder can contain any number of test plans.

To create a Smart Safety Test folder

- In the structure tree, right-click a resource and select **New** from the context menu. The *New Object* dialog box appears.
- Alternatively: Select a resource in the structure tree and click **New** on the Action Bar.
- Select Smart Safety Test and click **OK**. The Smart Safety Test folder is created in the structure tree as a child element of the current resource.

3.2 Creating a Test Plan

A test plan can only be created once the Smart Safety Test folder has been created.

The test plan can be created directly in SILworX or be imported from a CSV file.

To create a test plan directly in SILworX

- In the structure tree, right-click the Smart Safety Test folder and select **New** from the context menu. The *New Object* dialog box appears.
- Alternatively: Select the Smart Safety Test folder in the structure tree and click **New** on the Action Bar.
- Select a test plan in the *New Object* dialog box. Enter a new name for the test plan right away or accept the default. The test plan can also be renamed at a later point in time using the test plan properties.

To import a test plan from a CSV file

- In the structure tree, right-click the Smart Safety Test folder and select **Import Test Plan from CSV File** from the context menu. The *Import: Import Test Plan from CSV File* dialog box appears.
- In the *Import: Import Test Plan from CSV File* dialog box, select *Import File*, *Delimiter* and *Coding* and click **OK** to confirm.

3.3 Saving the Test Plan as a CSV File

To export a test plan to a CSV file

- In the structure tree, right-click the Smart Safety Test and select **Save Test Plan as a CSV File** from the context menu. The *Save Test Plan as a CSV File* dialog box appears.
- In the *Export: Export Test Plan to a CSV File* dialog box, select *Export File*, *Delimiter* and *Coding* and click **OK** to confirm.

4 Test Plan Editor

This section describes the structure of the Test Plan Editor and the elements available for a test plan.

To open the Test Plan Editor

In the structure tree, right-click a test plan and select **Edit** from the context menu. Or simply double-click the test plan.
Alternatively: Select the test plan and click **Edit** in the Action Bar. The Test Plan Editor appears.

The Test Plan Editor contains the following tabs:

- Test plan
- Properties
- Global variables
- Test element types

4.1 The Test Plan Tab

The **Test Plan** tab shows the elements of the test plan in a table for editing. It has a hierarchical structure.

Tip! To copy the test plan elements

Click the line number directly to select the entire line (line will be highlighted in yellow).

Right-click the selected line and select **Copy** from the context menu.

The **Paste** context menu function is used to insert the test element at a suitable position in the test plan.

4.2 The Properties Tab

The following test plan parameters can be configured on the **Properties** tab:

Parameter	Description
Type	Test plan
Name	A maximum of 120 characters. Default value: Test Plan
Description	User text to describe the test plan. The field on the cover sheet of the report has room for 14 lines of text, and they can be slightly longer than in the editor. If the text is longer than 14 lines, the text may overprint other cover sheet elements beyond the intended field. Default value: Empty
Default Tolerance for REAL	Specifies the tolerance which is to be applied when checking the test variables of type REAL if no individual tolerance specifications were made for it. Range of values: REAL ≥ 0.0 Default value: 0.0

Parameter	Description
Default Tolerance for LREAL	Specifies the tolerance which is to be applied when checking the test variables of type LREAL if no individual tolerance specifications were made for it. Range of values: LREAL ≥ 0.0 Default value: 0.0
Test Data Consistency Check	This parameter specifies whether the PADT should issue a message if a single data packet is not sufficient to transfer all the process or force values of the test step variables between PADT and PES. Range of values: BOOL Default value: TRUE

Important note concerning the test data consistency check.
The PES processes at most one data packet for process/force values per CPU cycle. If a single data packet is not sufficient to transfer a certain combination of values, the values concerned are read or set distributed over several cycles, and are therefore not consistent. This can cause problems if there are dependencies among the values be checked by the reading operation or that are important for the setting operation.

4.3 Object Panel

The Object Panel is located in the lower half of the Test Plan Editor and includes the tabs **Global Variables** and **Test Element Types** which are available via drag&drop for creating the test plan.

4.3.1 The Global Variables Tab

The **Global Variables** tab of the Object Panel contains all the global variables that can be used via drag&drop as test variables in the current test plan.

A test variable has the Setpoint property which may contain an optional tolerance value specification if the test variable has ANY_REAL, ANY_INT or TIME as the data type and belongs to a test step with type = CHECK_VALUE.

Range of values for the setpoint of a test variable:

- Valid literals of the data type.
- The tolerance value specification, if present, must be in square brackets 1…4 spaces behind the actual setpoint literal and must be formulated in accordance with the data type, e.g. **, 1.3e + 2 [0.00001]**, **T # 1200s [T # 500ms]**, **2500 [10]**. Within the brackets, the tolerance value specification may be surrounded by any number of spaces.
- In the case of ANY_REAL, **+ inf** (inf), **-inf** and **nan** are also allowed as setpoints, but only without tolerance value specification or tolerance value.
  Example of invalid entry: **-inf [0.01]**, **1.0 [inf]**.
- Floating point numbers displayed in the execution mode of the Test Plan Editor.
  Values of the data type (L)REAL are represented with an accuracy corresponding to this data type.

  Note: Under the test element CHECK_VALUE, a change in the nominal value can be observed when the data type is (L)REAL! Before execution, the setpoint column displays the user's original input: **0.3**
  After execution, the setpoint column displays the actually processed value as a floating point number: **0.300000012**

4.3.2 The Test Element Types Tab

The **Test Element Types** tab of the Object Panel contains all the test element types that can be used in the current test plan.

To add test element types

Drag the **test element types** from the Object Panel and drop them onto the test plan.

CASE

The test element CASE is inserted below the test element SEQUENCE. Usually, the test element PREPARATION is inserted before the test element CASE.

A CASE describes the actual test case. Here, the values of the input variables are set to the required state for the test using SET_VALUE and the values of the output variables are checked using CHECK_VALUE.

CHECK_VALUE

The test element CHECK_VALUE is inserted below the test element PREPARATION or CASE.

For the test variables listed below, CHECK_VALUE compares the setpoints with the process values of the connected force variables.

To synchronize the test plan with the user program to be tested, CHECK_VALUE can be assigned a waiting time. If the desired setpoints are reached within the waiting time, the test execution is resumed immediately.

Optionally, the *Setpoint* parameter of the test variable may contain one of the following operator statements (< , >,<>, <=, >=). To do so, the test variable must be of the data type ANY_REAL, ANY_INT or TIME and be inserted below a test element of the type CHECK_VALUE.

Operator statement	Description	Behavior for +inf/ -inf/ NaN	
		Actual value	Result
<	Less than Actual value < Set-point	+inf	Always false
		-inf	True if setpoint := {+inf, any finite value}, otherwise false
		NaN	Always false
<=	Less than or equal Actual value <= Set-point	+inf	True if setpoint := +inf, otherwise false
		-inf	True if setpoint := {-inf, +inf, any finite value}, otherwise false.
		NaN	nan: True if setpoint := nan, otherwise false
>	Less than Actual value < Set-point	+inf	True if setpoint := {-inf, any finite value}, otherwise false
		-inf	Always false
		NaN	Always false
>=	Greater than or equal Actual value >= Set-point	+inf	True if nominal value := {-inf, +inf, any finite value}, otherwise false
		-inf	True if setpoint := -inf, otherwise false
		NaN	True if setpoint := nan, otherwise false

Operator statement	Description	Behavior for +inf/ -inf/ NaN	
		Actual value	Result
<>	Not equal Actual value <> Set-point	+inf	False if setpoint := +inf, otherwise true
		-inf	False if setpoint := -inf, otherwise true
		NaN	Always true

This is independent of a specified tolerance value, see above, *Range of values for the setpoint of a test variable*.

Examples of entries in the *Setpoint* field with operator statement, nsetpoint, tolerance value:

- "< 1.3e+2 [0.00001]"
- "> T#1200s [T#500ms]"
- "<> 2500"

CHECKPOINT

The test element CHECKPOINT is inserted below the test element PREPARATION or CASE.

CHECKPOINT is inserted at a location in the test plan at which a manual action by the user is required for execution of the test plan.

The setpoint is a multi-line text field for the CHECKPOINT test element. The user can enter here the description of an action that is to be carried out manually.

When the test plan is being carried out, the test will be interrupted at the CHECKPOINT by a message which prompts the user to perform a manual action. Clicking the **OK** button confirms the message.
Use the **Continue** button to open the CHECKPOINT dialog box with a multi-line text field. The user is prompted to enter a comment on this manual action. Clicking the **Yes** button resumes the execution of the test plan.

PREPARATION

The test element PREPARATION is inserted below the test element SEQUENCE.

A PREPARATION serves to prepare the controller for the actual test case (CASE). Here, the values of the input variables are set to the necessary state before the test case using SET_VALUE.

RESET

The test element RESET can be inserted at all levels of the test plan.

A RESET repeats the execution of all SET_VALUE test steps present in the test plan up to this point in the same and in subordinate structure levels. When doing so, however, the setpoints are not transferred for the associated test variables. Instead, a force data reset is performed for the forced test variables.

RESET_VALUE

The test element RESET_VALUE can be inserted below the test element PREPARATION or CASE.

A RESET_VALUE performs a force data RESET for the test variables listed below.

SEQUENCE

A SEQUENCE describes a group of sequential test element types that are associated with one another and are processed sequentially. A sequence is self-contained and is separate from the preceding and subsequent sequences.

A sequence represents the highest level in the hierarchy of the test plan elements.

SET_VALUE

The test element SET_VALUE is inserted below the test element PREPARATION or CASE.

The test variables which are required for the preparation or the test case and are to be set, are inserted below SET_VALUE. The setpoints, which are transferred to the controller as the force values in the connected force variables including setting the associated force switches, are entered here for test variables.

WAIT

The test element WAIT is inserted below the test element PREPARATION or CASE.

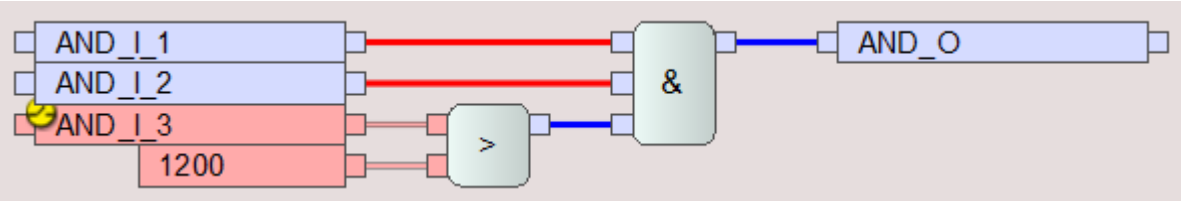
WAIT is used to specify a defined waiting time at any desired location below the test element PREPARATION or CASE. When executing a WAIT, SILworX allows the time entered in the setpoint to elapse before moving on to the next test element.

The use of WAIT is always to be preferred when there is no specific setpoint to which CHECK_VALUE and optional waiting time can be applied to wait for a specific period of time.

5 Example of a Smart Safety Test

Suppose a Smart Safety Test is to be carried out for the function contained in a user program.

Function in the User Program



Test Plan

For a Smart Safety Test of this function, a test plan could be structured as follows:

The Boolean inputs (AND_I_1, AND_I_2) can be simulated by setting force values or switches on the digital inputs.

The analog input AND_I_3 can be simulated by setting a force value or a current source on the analog inputs.

Fortsetzen | Einzelschritt | Zurücksetzen | PDF-Report | CSV-Report

Testplan | Eigenschaften | Ausführungsinformationen

	Name	Typ	Sollwert	Istwert	Status
1	SEQUENCE	SEQUENCE			In Ausführung
2	PREPARATION	PREPARATION			OK
3	SET_VALUE	SET_VALUE			OK
4	AND_I_1	BOOL	FALSE	FALSE	OK
5	AND_I_2	BOOL	FALSE	FALSE	OK
6	CASE	CASE			In Ausführung
7	CHECKPOINT	CHECKPOINT	Analoger Eingang AND_I_3 > 1200 ?		Unterbrochen wegen Prüfpunkt
8	CHECK_VALUE_2	CHECK_VALUE			
9	AND_I_3	INT	> 1200		
10	AND_O	BOOL	TRUE		

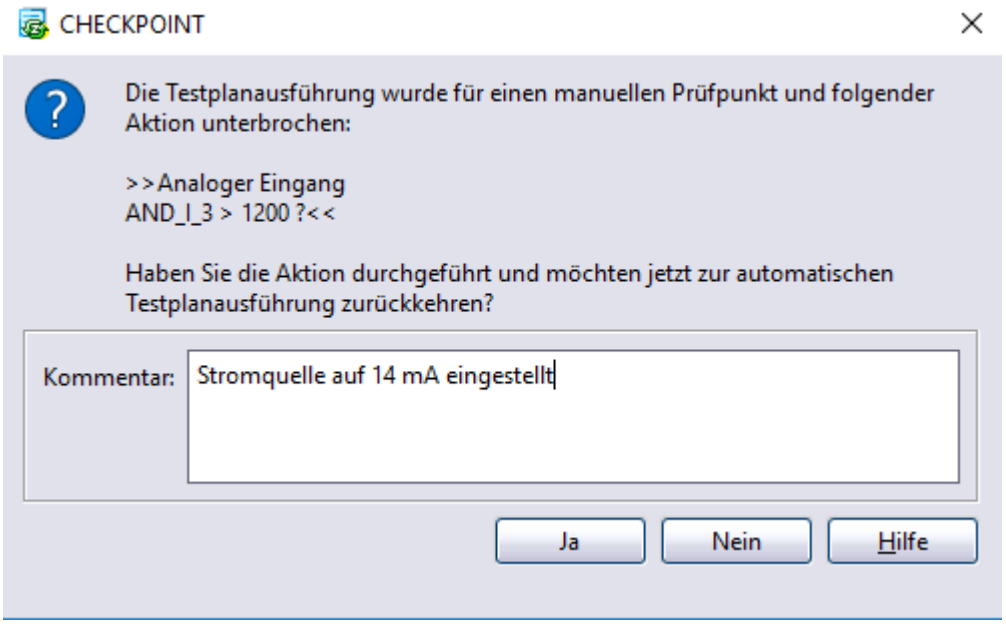
CHECKPOINT

When the test plan is being carried out, the test will be interrupted at the CHECKPOINT by a message which prompts the user to perform a manual action. Clicking the **OK** button confirms the message.

Use the **Continue** button to open the CHECKPOINT dialog box with a multi-line text field. The user is prompted to enter a comment on this manual action. Clicking the **Yes** button resumes the execution of the test plan.

To successfully carry out this test plan example, the external current source on the analog input AND_I_3 must be set to >12.0 mA.

Clicking the **OK** button resumes the execution of the test plan.



Test Result

Testplan | Eigenschaften | Ausführungsinformationen

	Name	Typ	Sollwert	Istwert	Status
1	SEQUENCE	SEQUENCE			OK
2	PREPARATION	PREPARATION			OK
3	SET_VALUE	SET_VALUE			OK
4	AND_I_1	BOOL	FALSE	FALSE	OK
5	AND_I_2	BOOL	FALSE	FALSE	OK
6	CASE	CASE			OK
7	CHECKPOINT	CHECKPOINT	Analoger Eingang AND_I_3 > 1200 ?	Stromq...	OK
8	CHECK_VALUE_2	CHECK_VALUE			OK
9	AND_I_3	INT	> 1200	1400	OK
10	AND_O	BOOL	TRUE	TRUE	OK

After carrying out the test plan, the status can have one of the two following statuses:

- Execution completed successfully; all test plan elements were successful!
  Status table cells are given a green background if the expected result is obtained and the execution was successful.
- Execution failed; at least one of the test plan elements was not successful!
  Status table cells are given a red background and the font color is changed to white if the expected result is not obtained and the execution was not successful.

6 Test Plan Editor in Run Mode

To execute the Test Plan Editor, it must be opened via the actions **Offline** or **Online**. The requirements for running the Test Plan Editor are listed in the following sections.

Until the automatic execution of the test plan, the online values displayed in the online status view are refreshed cyclically. Once the test plan has begun, the cyclical refreshing of the online values stops so that the online values frozen before the starting time are used.

6.1 Testing the Test Plan in Offline Simulation

The user must ensure that the following requirements are met for testing the test plan offline:

- A successful syntax and runtime validation of the test plan.
- The current user program which contains the logic to be tested must be in the *RUN* state.

To open a test plan in the offline view

In the structure tree, right-click a test plan and select **Offline** from the context menu.

If several user programs exist for a resource, a dialog box appears in which the user program for which the logic is to be simulated must be selected.

Alternatively: Select the test plan and click **Offline** in the Action Bar. The Test Plan Editor appears.

6.2 MultiForcing and Smart Safety Test

If MultiForcing is active, the current number of users with forcing permissions is displayed in the online view of the Smart Safety Test.

During execution of the Smart Safety Test, there must be no forced values which concern the Smart Safety Test because otherwise the test results might not be usable.

Falsified test results due to forced values are possible!
Forced values can lead to incorrect output values and thus to falsified test results. Forcing is only possible after consultation with the testing agency responsible for system acceptance testing; see the safety manual of the respective controller.

6.3 Testing the Test Plan Online

The user must ensure that the following requirements are met for testing the test plan online:

- A successful syntax and runtime validation of the test plan.
- An online connection to the controller (at least with write access).
- *Global Forcing* must be activated on the controller.
- The controller and the user programs required for the test must be in the *RUN* state.
- The execution status *Forcing not available* remains if the loaded configuration in the controller does

not match the project status in the PADT.

- If the Force Editor is open, it can display force variables and support active forcing.

The timing of the test steps when executing the test plan can be influenced, e.g., by the network traffic of other online tools or by computationally intensive PADT functions.

To open a test plan in the online view

If you are not already logged in to the controller, you are asked to perform a *System-Login*. After the login, the Test Plan Editor appears and the test plan can be edited in online mode.

In the structure tree, right-click the test plan and select **Online** from the context menu.

6.4 Buttons of the Online (Offline) Test Plan Editor

Button	**Description**
Start, Resume Stop	This button can be used by the user to start or stop the test plan. The button switches between the functions *Start, Resume* and *Stop* depending on the current state.
Single Step	This button can be used by the user to initiate the automatic execution of a single test step (single step).
Reset	After the executed test plan has been reset, it is ready to be run again and is in the state *Ready for Execution*.
PDF Report	This button can be used by the user to create a report. When the PDF Report button is actuated, a dialog appears in which the following options can be specified for the report: • Directory path and filename of the PDF file to which the report is to be written. • Selection of whether the report should be created in black and white or in color.
View box	**Description**
Execution Status	Range of values: • Test plan contains syntax errors • Forcing not available • Ready for execution, execution in progress… • Single step in progress… • The execution is being halted • The execution was halted • Export successfully completed • Execution failed • Resetting in progress…
Execution Time	Time required (until now) for executing the test plan. Interruptions of the execution (e.g. due to break points, single steps, stops) are not included in the calculation.

6.5 Executing the Test Plan

Before the test plan is started in the *Online (Offline) Test Plan -Editor* the **Start/Resume** and **Single Step** buttons are active.

6.5.1 Buttons

Start

Actuating the **Start/Resume** button starts the test run. The **Start/Resume** button changes to the **Stop** button.

A validation of the test plan is performed. If a validation error is detected, one of the following error messages is issued:

- *Test plan start was aborted.*
- *Execution cannot be started*

If the validation is successful, the test run begins and runs until the end or to possible break points.

Stop

When a test run is completed or stopped, the **Stop** button becomes the **Start/Resume** button once again.

The test plan table shows the results obtained (actual value, status).

Creating a PDF Report

The test protocol can be created in PDF format by using the **PDF Report** button.

Resetting the Test Plan

Use the **Reset** button to reset the test plan. On reset, the PADT performs a forced data reset for all test variables set by means of SET_VALUE up to this point in this test plan.

6.5.2 Context Menu Functions

Set Break Points

The user can set *Break Points* for the interruption of the automatic execution of the test plan at any test elements desired.

To set a break point

In the test plan, right-click a test element and select **Set/Reset Break Point** from the context menu.

When the execution arrives at the break point, the test plan has the status *The execution was halted*.

The execution can be resumed by using the **Resume** or **Single Step** button.

Activate/Deactivate Test Elements

The user is able to *activate/deactivate* any test elements desired in the test plan so that they are not executed during the automatic execution of the test plan.

This function can be carried out before the execution in the *Ready for execution* status or in *Test plan contains syntax errors*.

This feature allows you to disable test elements with syntax errors and thereby to bring the entire test plan into a ready-to-run state.

To activate/deactivate test elements

In the test plan, right-click a test element and select **Activate/Deactivate** from the context menu.

After the test plan has been reset, the deactivated test plan elements remain deactivated. The deactivated test elements can be activated again using the **Activate/Deactivate** context menu function.

6.5.3 Interactions Between Smart Safety Test and Other SILworX Functions

The force data query of global variables is restricted during the execution of a test plan for the respective resource.

The display of the user program on the PADT has a lower priority during the execution of the Smart Safety Test. The updating of visible variables occurs only for variables and force state data required by CHECK_VALUE test steps. The non-updated variables are identified by gray text in the force tables, empty OLT fields, and black lines in the FBD logic.

These restrictions do not apply during the execution of a WAIT test step.

The timing of the test steps when executing the test plan can be influenced, e.g., by the network traffic of other online tools or by computationally intensive PADT functions.

7 Support

Refer to the following table for any question, concern or suggestion related to the programming tool.

Area	Website
News, manuals	All the current manuals can be obtained upon request by sending an e-mail to: documentation@hima.com.
Contact details:	https://www.hima.com/en/about-hima/contacts-worldwide/
Technical support	https://www.hima.com/en/products-services/support/
Seminar program	https://www.hima.com/en/products-services/seminars/

Manual
Smart Safety Test Rev. 11.00.00

HI 801 495 E (1924)

Für weitere Informationen kontaktieren Sie:

HIMA Paul Hildebrandt GmbH
Albert-Bassermann-Str. 28
68782 Brühl, Deutschland

Telefon: +49 6202 709-0
E-Mail: info@hima.com

www.hima.com/de/

HIMA SMART SAFETY.

www.hima.com