**Lab work**

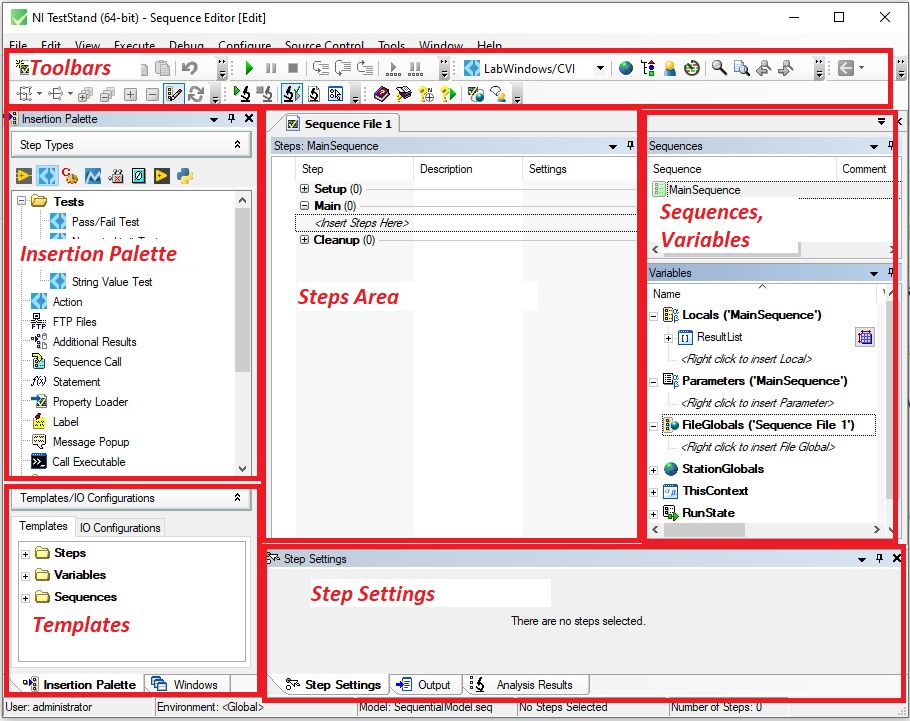
**TestStand introduction**

**Work objective**

Get acquainted with National Instruments automated test management software TestStand.

**Theoretical part**

TestStand is application software that helps engineers quickly develop robust automated test and validation systems. You can extend the functionality of your system by developing test sequences in TestStand that integrate code modules written in any programming language. You can use built-in functionality to profile and optimize speed and parallelism before deploying test systems to production. TestStand provides extensible plug-ins for reporting, database logging, and connectivity to other systems, meeting the needs of any environment. With TestStand, you can confidently deploy test systems with increased throughput that run at the speed of production.



**Fig. 1.** TestStand environment

Fig 1 shows TestSand environment after startup. In the central part („Steps Area“) is a sequence of tests performed. The parameters of each individual step test are changed in the lower part of the Step Settings environment.

More about TestStand: <http://www.ni.com/en-us/shop/electronic-test-instrumentation/application-software-for-electronic-test-and-instrumentation-category/what-is-teststand.html>

**Practical Part**

Launch the TestStand: *Windows Programs - > NI TestStand 2021*

Use default username, *administrator*, leave password field empty and click OK

Complete the following steps to load and view a workspace file:

* Select *File -> Open File* and navigate to the *<TestStand Public>\Tutorial* directory (*C:\Documents and Settings\All Users\Documents\National Instruments\TestStand)*
* Select *Tutorial.tsw* and click *Open*

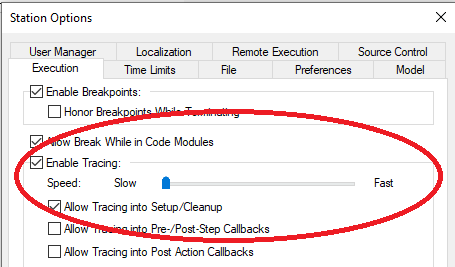
Graphical user interface, application

Description automatically generated

**Fig 2**. TestStand tutorials

Change test execution settings:

* Select *Configure -> Station Options* to launch the Station Options dialog box
* In the tab *Execution* mark „*Enable Tracing*“, move slider to the left to slow the test execution speed.
* Click *OK*.



**Fig 3**. Test Execution settings

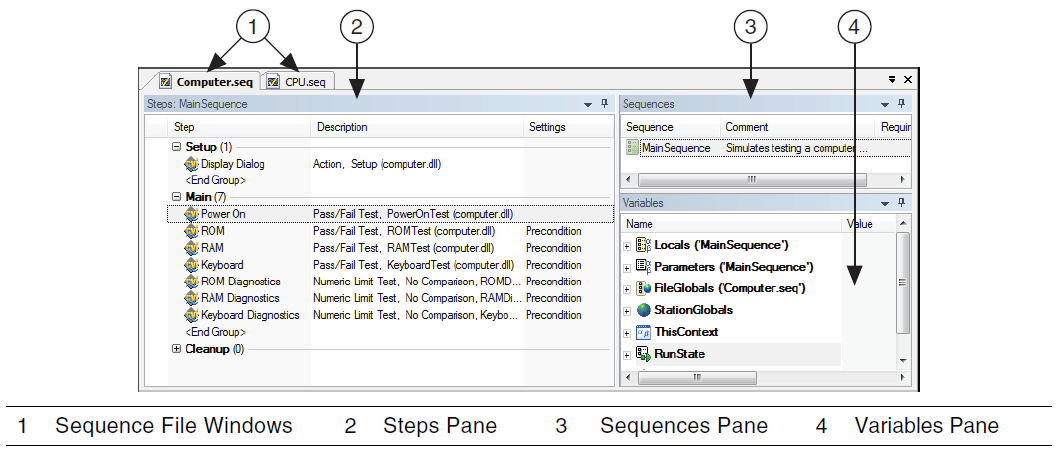
**Example 1. Computer test**

Open computer test sequence – double click on „*Computer.seq*“ in the *Workspace* window.

The test sequence will be shown in the main window. A sequence consists of a series of steps. A step can perform many actions, such as initializing an instrument, performing a complex test, or controlling the flow of execution in a sequence.

The Setup, Main, and Cleanup groups of the Steps pane display a list of the steps in the step group. TestStand executes the steps in the Setup step group first, the Main step group second, and the Cleanup step group last.

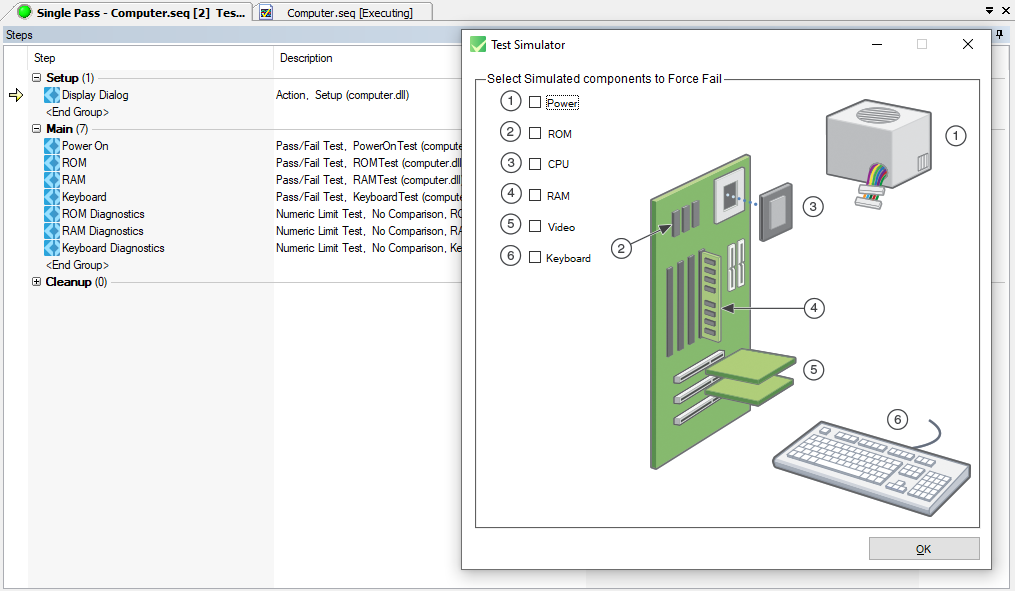
* The Setup step group typically contains steps that initialize instruments, fixtures, or a UUT.
* The Main step group typically contains the bulk of the steps in a sequence, including the steps that test the UUT.
* The Cleanup step group typically contains steps that power down or restore the initial state of instruments, fixtures, and the UUT.



**4 pav**. Computer test sequence

To execute tests, select *Execute - > Single Pass (Ctrl + F5)*

The first step in the Setup step group launches dialog box. You can select component to simulate failure during the test execution.



**Fig 5.** Dialog box to simulate failure

After test sequence completed, report will be generated. You can close report selecting *Windows -> Close Completed Executions*.

**Task 1.**

Repeat test sequence, simulating different component failure. Describe which tests are executed in each case and why.

**Example 2. Executing a Sequence Using the Sequential Process Model**

The Sequential process model includes the following Execution entry points:

* Test UUTs Execution Entry Point—The Test UUTs Execution entry point initiates a loop that repeatedly identifies and tests UUTs. Common operations of the process model include identifying the UUT, notifying the operator of pass/fail status, logging results, and generating reports.
* Single Pass Execution Entry Point—The Single Pass Execution entry point tests a single UUT without identifying it. Use the Single Pass Execution entry point when you want to debug tests or determine whether the sequence execution proceeds as you intended.

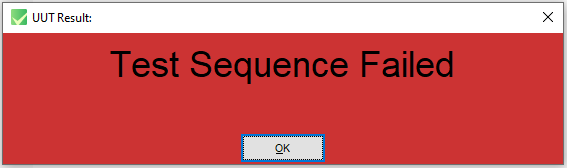
Complete following steps to run Test UUTs Execution Entry Point

* Select *Configure - > Station Option*, tab *Model.* Verify that *SequentialModel.seq* is selected from the Station Model ring control to select the Sequential model as the default process model and click OK.
* Select *Execute -> Test UUTs*. Before executing the steps in MainSequence, the process model sequence launches a UUT Information dialog box that prompts you for a serial number.

Graphical user interface, application

Description automatically generated

* Enter any serial number and click *OK*.
* After completing the steps in MainSequence, the process model displays a banner that indicates the result of the UUT.



* Click *OK* to close the UUT Result banner. TestStand generates a report, but does not display the eport until you finish testing all the UUTs. TestStand launches the UUT Information dialog box again.
* Click Stop in the UUT Information dialog box to stop the loop and complete the execution.

**Task 2.**

Perform test Using the Sequential Process Model. Save the report.

**Example 3. Executing a Sequence Using the Batch Process Model**

Use the Batch model to control a set of test sockets that test multiple UUTs as a group. For example, if you have a set of circuit boards attached to a common carrier, use the Batch model to ensure that you start and finish testing all boards at the same time. With the synchronization features of the Batch model, you can direct a step that applies to the batch as a whole to run only once per batch instead of once for each UUT. You can also specify whether certain steps or groups of steps cannot run on more than one UUT at a time or whether certain steps must run on all UUTs at the same time. The Batch model generates batch reports that summarize the test results for the UUTs in the batch.

Complete the following steps to run the *BatchUUT.seq* sequence file using the Test UUTs Execution entry point of the Batch model:

* Open <TestStand Public>\Tutorial\BatchUUT.seq and examine the steps and comments in the sequence file to familiarize yourself with the functionality of the sequence.
* Select *Configure -> Model Options* to launch the Model Options dialog box

Graphical user interface, application

Description automatically generated

* Change *Number of Test Sockets: 2, Tile Execution Windows, Don‘t Synchronize*.
* Select *Execute -> Test UUTs* Before executing the steps in MainSequence, the process model sequence launches the UUT Information dialog box for the Batch model, which prompts you for a batch serial number and UUT serial numbers for each test socket. You can also disable test sockets in the UUT Information dialog box.
* Enter any batch serial number and UUT serial numbers. Click *Go*.

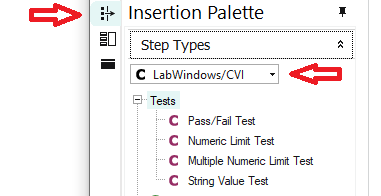
**Task 3.**

Perform test Using the Batch Process Model. Change Number of Test Sockets to 4. Examine the reports that include results for each batch and UUT.

**Example 4. Adding a New Steps in to Sequence**

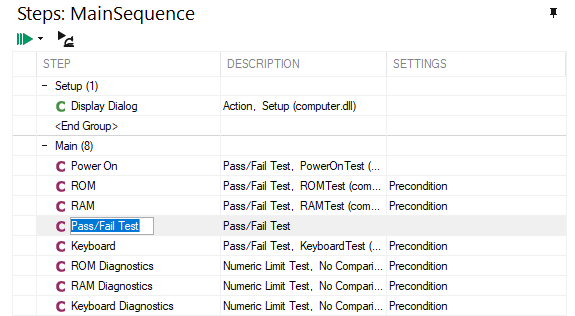
Complete the following steps to add a Pass/Fail Test step to a sequence and configure the step to call a function in a LabWindows/CVI DLL code module by specifying the module adapter to use. TestStand does not need the source code to invoke a DLL code module. Instead, TestStand uses module adapters to determine the type of code module, how to call the code module, and how to pass parameters to the code module.

* Open *<TestStand Public>\Tutorial\Computer.seq*
* Important! Select *File -> Save <filename> As* and save the sequence file as *Computer2.seq* in the *<TestStand Public>\Tutorial* directory to preserve original sequence file.
* Open *Insertion Palette* and select *LabWindows/CVI* to specify the module adapter the step uses.



**Fig. 6**. Insertion Palette

* On the Insertion Palette, select *Tests -> Pass/Fail Test* and drag the step below the RAM step on the Steps pane to add a *Pass/Fail Test* step. By default, the name of the new step is Pass/Fail Test.

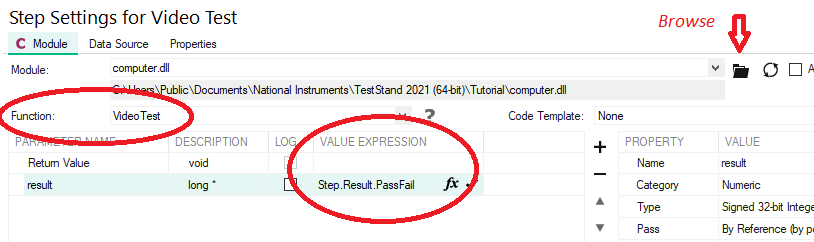


**Fig. 7**. New step in the MainSequence

* Rename the new step *Video Test* by selecting the step on the Steps pane and pressing the <F2> key.

Complete the following steps to specify the code module the step executes and to specify the parameters for a function in the code module for the Computer2.seq sequence file.

* Select the Video Test step and click the Module tab of the Step Settings pane.
* Click the Browse button located to the right of the Module control, select <TestStand Public>\Tutorial\computer.dll, and click Open. When you select a DLL, TestStand reads the type library or exported information of the DLL and lists the functions TestStand can call in the Function ring control.



**Fig. 8**. Step settings configuration

* Select *VideoTest* from the Function ring control. TestStand uses the prototype information stored in the type library or the exported information of the DLL to populate the Parameters Table.
* In the *Value Expression* column of the result parameter, enter *Step.Result.PassFail*.

Complete the following steps to examine and modify step properties.

Each step in a sequence contains properties. All steps have a common set of properties that determine the following attributes: When to load the code module for the step; when to execute the step; what information TestStand examines to determine the status of the step, etc.

* Select the *Video Test* step on the Steps pane and click the Properties tab of the Step Settings pane.
* Click *Preconditions* on the *Properties tab* to show the Preconditions panel. A precondition is a set of conditions for a step that must evaluate to True for TestStand to execute the step during the normal flow of execution in a sequence.
* Click the *Precondition Builder* button, as shown in the following figure and located to the right of the Precondition Expression control on the Preconditions panel, to launch the Precondition Builder dialog box.



* In the Insert Step Status section, select the Power On step from the list of step names for the Main step group and click the Insert Step Pass button. The Conditions control now contains the string PASS Power On, which indicates that the step executes only if the Power On step passes.
* Click OK to close the Precondition Builder dialog box and confirm that the Preconditions panel matches the settings shown in Figure below

Graphical user interface, text, application

Description automatically generated

**Fig. 9**. Preconditions pane

* Click *Post Actions* on the *Properties* tab of the Step Settings pane to show the Post Actions panel, on which you can specify what type of action occurs after the step executes. You can make the action conditional on the pass/fail status of the step or on any custom condition expression.
* Select *Terminate execution* from the *On Fail* ring control.
* Click *Looping* on the *Properties* tab of the Step Settings pane to show the Looping panel. Set the following values into the corresponding controls:
  + Loop Type – Fixed number of loops
  + Number of Loops – 10
  + Loop result is Fail if - <80 %
* Confirm that the Settings column on the Steps pane of the Sequence File window shows that the Video Test step contains Loop, Precondition, and Post Action settings.

Table

Description automatically generated

**Fig. 10**. Main Sequence

* Select *Execute -> Single Pass* to execute test sequency.

**Task 4.**

Perform computer test with Video failure and without. What tests are performed in one case and another?