

GRL Automation Framework

Installation and Quick Start Guide

Rev. v1.1

The screenshot displays the GRL Automation Framework software interface, which is organized into three main horizontal panels. Each panel has a title bar with a menu (Application, Options, View, Preference, License, Help) and a toolbar with various icons for navigation and actions.

Equipment Setup

This panel contains a table for configuring equipment. The table has columns for Name, ID, Address, Type, Vendor, and Lib. The equipment listed includes a Scope, BERT, and ISI Generator.

Name	ID	Address	Type	Vendor	Lib
Scope	Scope	GPIO8::1::INSTR	Oscilloscope	Tektronix	TekDPOJETSc
BERT	BERT	127.0.0.5:23	BERT	Tektronix	TekBertScope
ISI Generator	ISIGen	COM13	ISI Generator	Artek	StandardSerial

Session Info

This panel displays session information. It includes tabs for DUT Info, Test Info, and Software Info. The Software Info tab is currently selected, showing the Software Revision: 1.0.0.0.

Software Revision: 1.0.0.0

Report

This panel displays a report of test results. It includes a table with columns for No, Test Name, Result, Limits, Value, and Lane SSCData Rat. The report shows results for various calibration tests, all of which passed.

No	Test Name	Result	Limits	Value	Lane SSCData Rat
16	Rj Calibration	PASS	True/False	True	Any A.. HBR2
17	Sj Fixed Calibration	PASS	True/False	True	Any A.. HBR2
18	Sj Sweep Calibration	PASS	True/False	True	Any A.. HBR2
19	Sj Sweep Calibration	PASS	True/False	True	Any A.. HBR2
20	Sj Sweep Calibration	PASS	True/False	True	Any A.. HBR2
21	Sj Sweep Calibration	PASS	True/False	True	Any A.. HBR2
22	Crosstalk Calibration	PASS	True/False	True	Any A.. HBR2
23	Eye Height Calibration	PASS	True/False	True	L... A.. HBR2

Below the table, there are buttons for "Generate report", "Delete", and "Delete All". A checkbox labeled "Plot Calibration Data" is also present.

Installation

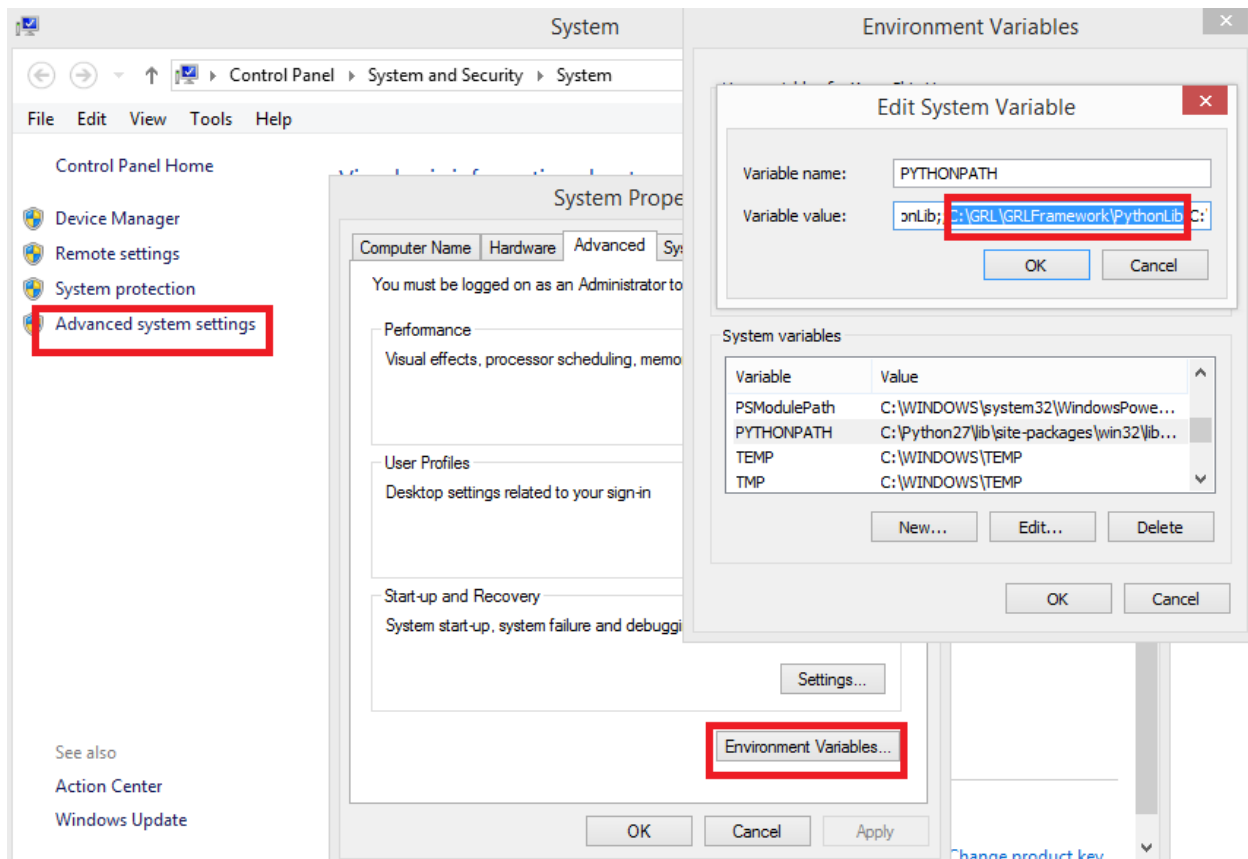
1. Install the installer package GRLFrameworkInstaller0000x.exe
2. Launch the app, Go to license details, copy the license string to me and send to support@graniteriverlabs.com
3. Install python27 from <http://www.python.org/download/releases/2.7.6/>. Make sure the install path is C:\Python27. Note: Do not use version Python33
4. Install Python for .net <http://sourceforge.net/projects/pythonnet/files/>... This is required for Remote IO control.(optional).
5. Install Visual C# if script is to be written in C#.
6. Install VISA component(TekVISA, Agilent IO for example)
7. Install other third party component(eg, ScopeApp/RX app RPI module).
8. GRL Framework is an automation platform to ease automation efforts only and does not license Python, Visual C# or any third-party components. Users of GRL Framework need to ensure eligibility to use these third party tools.

Setting Up Python Environment

1. To Setup environment for python development, go to Control Panel\System and Security\System
2. Click Advanced System Settings.
3. Click on Environment Variables.
4. Add a System Variables:

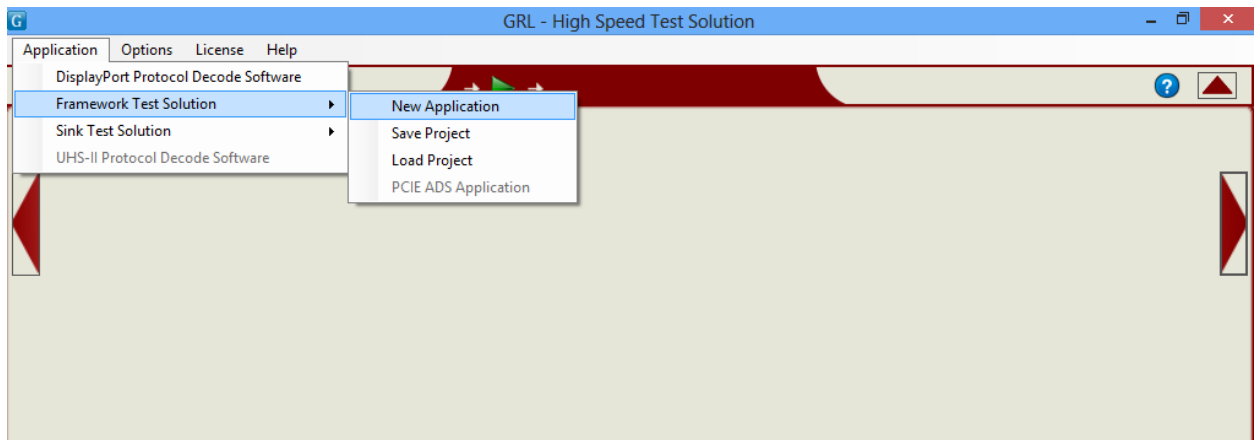
Variable Name: PYTHONPATH

Variable Value: C:\GRL\GRLFramework\PythonLib

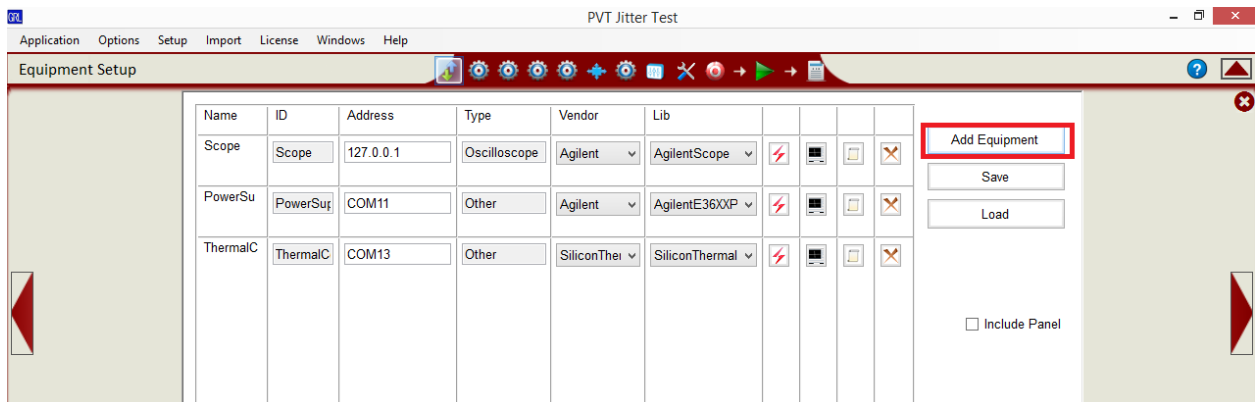


Chapter 1: Using the Automation Framework

1. Select “GRL Framework” from Start Menu ->GRL.
2. Go to Framework Test Solution to start a new Application, Save Project or Load Project.

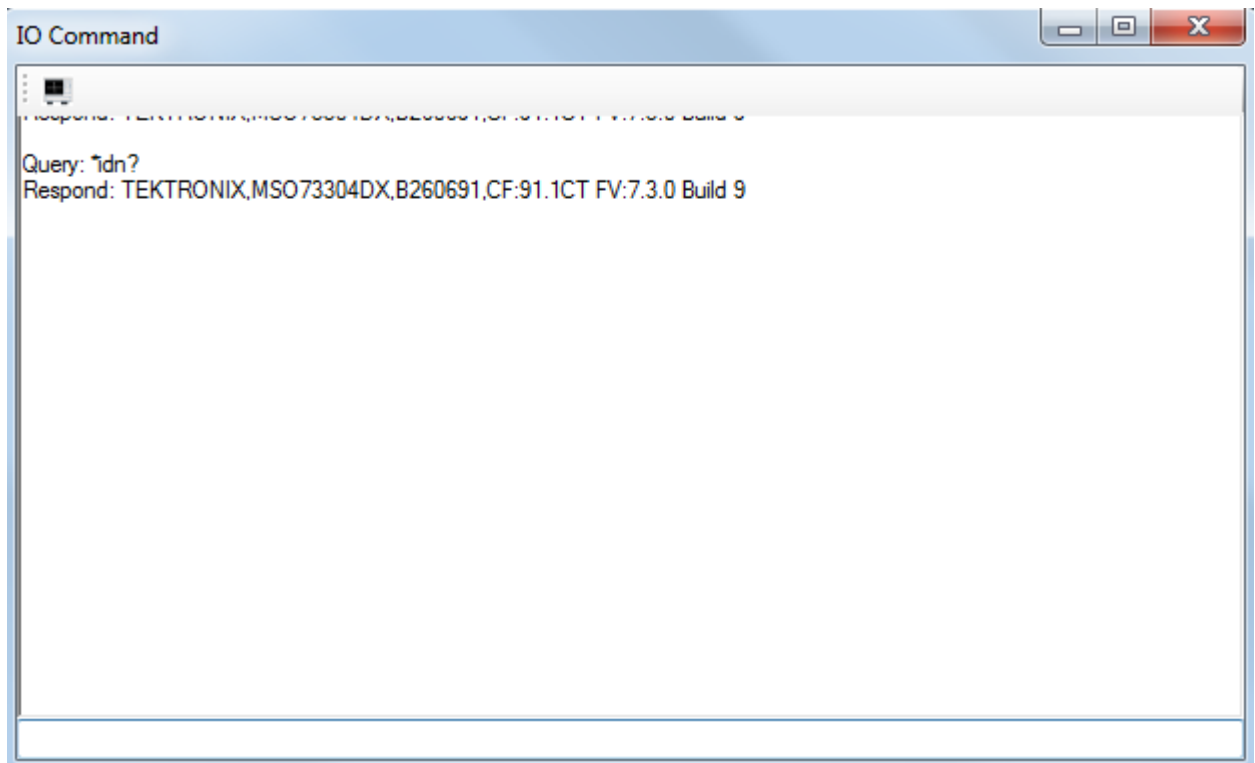


3. Equipment Setup:
 - Click “Add Equipment” To add an equipment
 - Skip the page if equipment is not required.
 - “ID” is used by code to access the equipment.



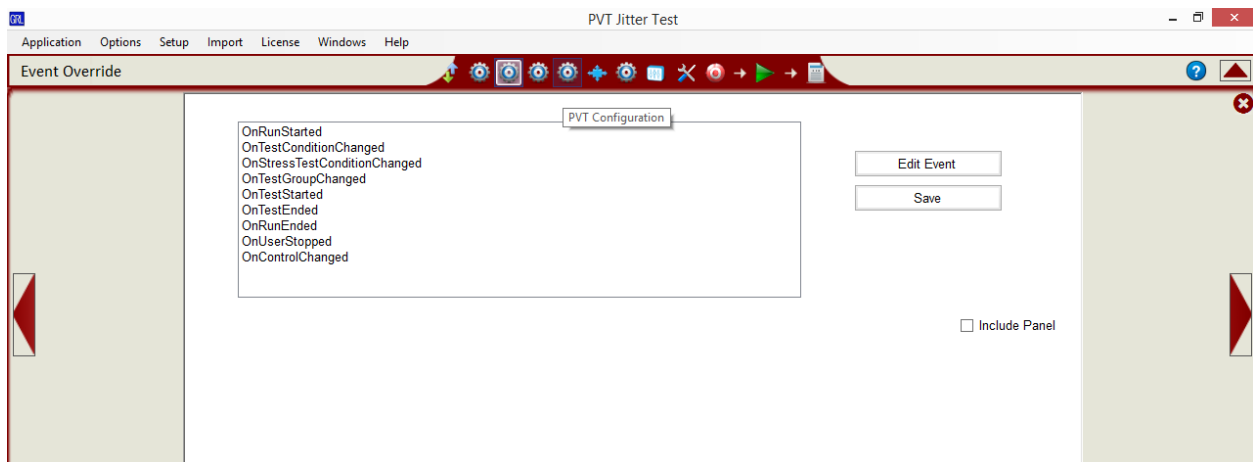
- Developer can check connectivity to equipment by clicking on  icon beside each equipment added. If connection is successfully established, the icon will turn green .
- Developer can also click  icon to issue command manually to equipment. For example, for GPIB supported equipment, “*idn?” is always a good way to check if equipment respond to framework correctly.

Example:



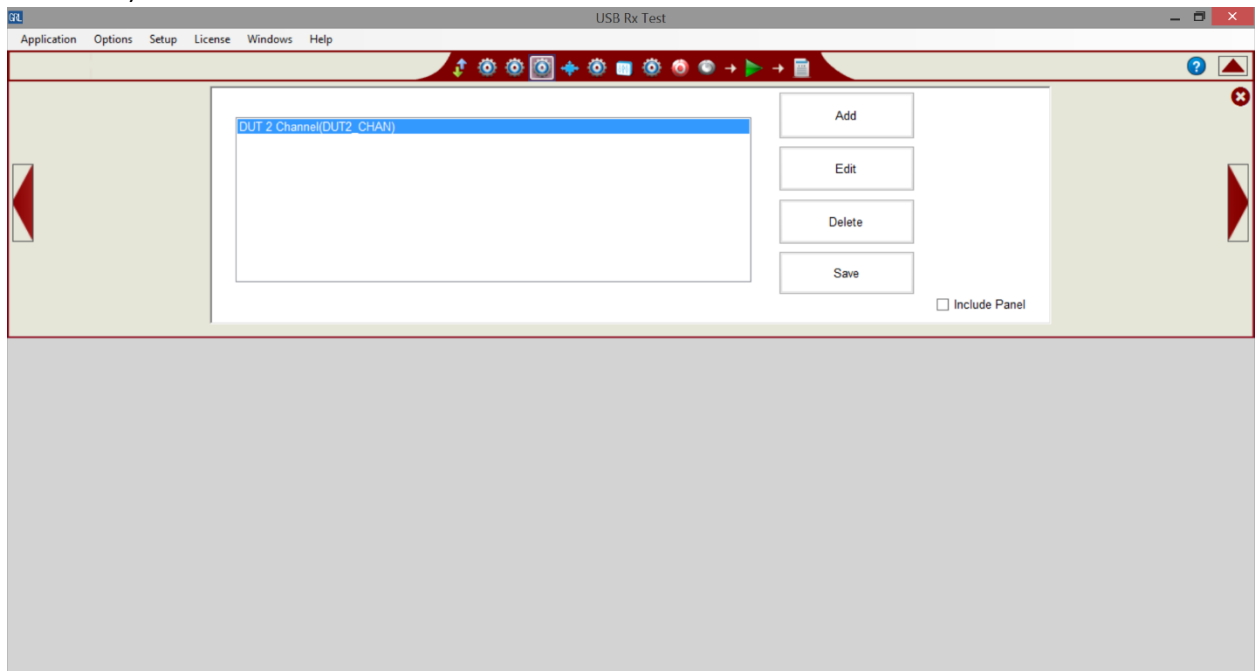
4. Event Override

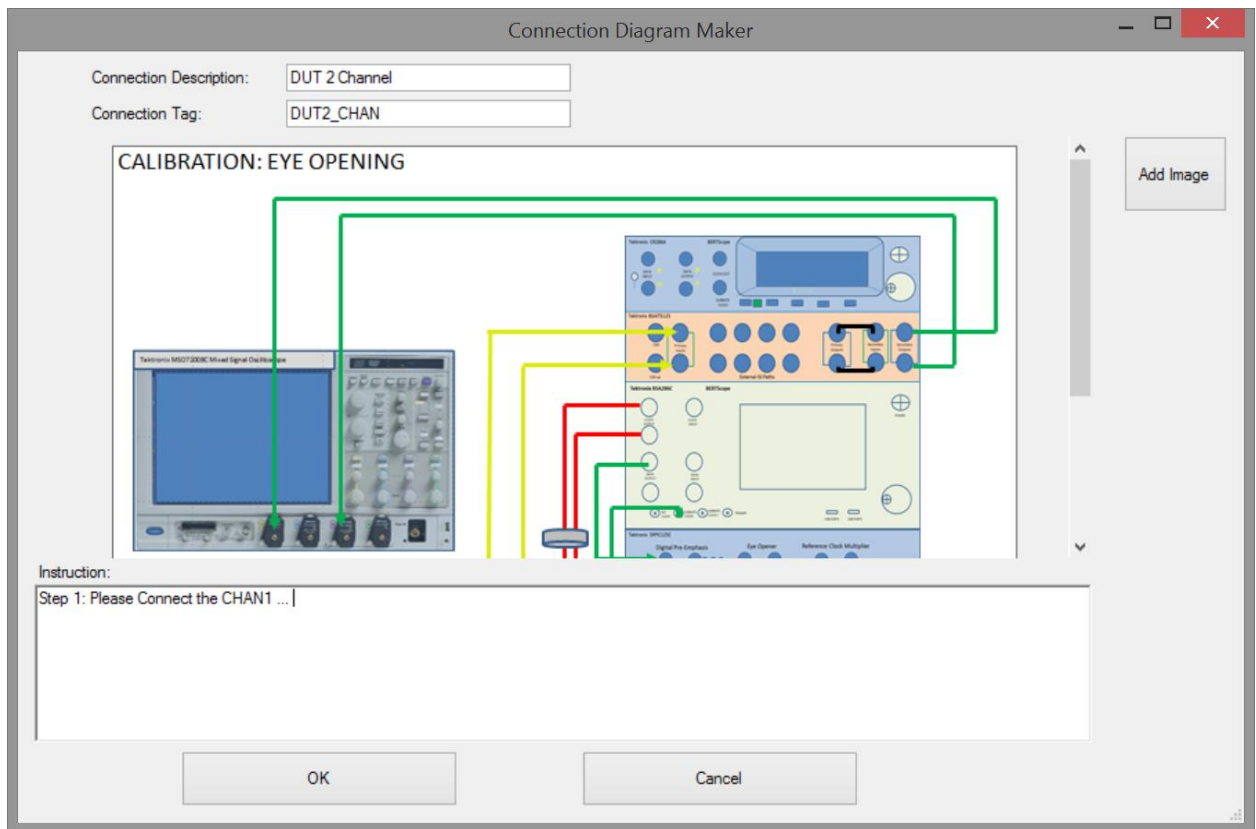
- Event Override page allows you to override an event with your own code.
- The following events are supported.
 - OnRunStarted : An event that is triggered before a new run session is started. For instance, one might initiate DUT initial state here.
 - OnTestConditionedChanged: An event that is triggered when any condition that is defined in “Condition” page(described as item 7 below) is called. For instance, this might be useful for one to change a DUT state(eg, changing DUT bit rate or control switch matrix to a different lane)
 - OnStressTestConditionChanged: An event that is triggered when condition defined in “PVT Condition tab” is changed.(eg, changing Voltage or temperature)
 - OnTestGroupChanged: An event that is triggered when “Test group” of test is changed.
 - OnTestStarted: An event that is triggered before starting a test item
 - OnRunEnded: An event that is triggered after Run session is completed. For instance, one might aggregate data in this event.
 - OnUserStopped: An event that is triggered when test stop after user click on the “Stop” button. For instance, user might clean up/reset DUT state user cancel a test.
 - OnControlChanged: An event that is triggered when a GUI control component(eg: checkbox, combobox, textbox) defined in the framework receives an user input. For instance, once might hide certain tests to be seen by end users when a different specs(defined by a text box) is changed.
- Select on any of the event and select Edit event.
- Provide a “Name” and click edit script. A script will be popped up for you to override the event.



5. Connection Diagram

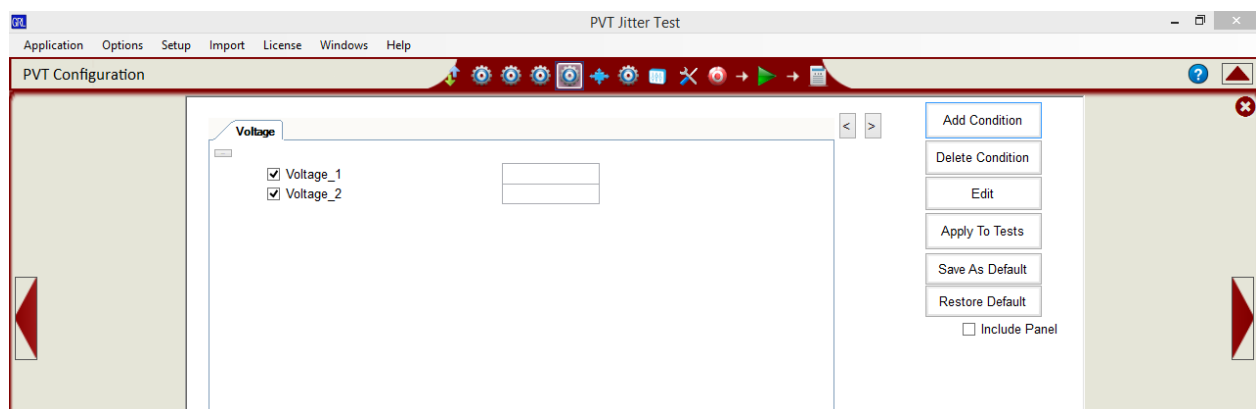
- Developer can add meaningful connection diagram in this page and give a unique name to it.
- Connection diagram can then be tied to the test created later in "Select Tests" Page(item 9 below).





6. PVT Configuration

- This tab integrate PVT capability in such a way that user only need to define the PVT conditions and a PVT test plan will be automatically populated for all the written tests.



- To add a PVT condition, Click on “Add Condition” and a dialog will pop up as below.
- In the example below, you are defining temperature with 3 corners, T1,T2 and T3. And the default for T1,T2,T3 are 10C,20C,30C respectively.

New Condition Group:

Description:

Condition To Add:

Description:

Variable	Value
T1	10
T2	20
T3	30

-
- Click OK after defined the PVT condition. Now a new column of condition is added to the “PVT Configuration” page

PVT Jitter Test

Application Options Setup Import License Windows Help

PVT Configuration

Voltage Temperature

☒ T1 10

☒ T2 20

☒ T3 30

☐ Include Panel

-
- To access the condition parameter in the code, use `FWAppObj.GetProperty("Condition")`. For example, `FWAppObj.GetProperty("Temperature")` should return either T1,T2,T3.
- To access the value(or temperature tie to the condition) use `FWAppObj.GetProperty("EitherCondition_PVTValue")`. For example, `FWAppObj.GetProperty("T1_PVTValue")` return "10" in this case
-

7. Conditions

- In some circumstances, a DUT specific test plan might be required. For instance, some DUT/standards might have a different conditions to tests and different conditions might have different specifications.
- In such a case, then you might want to define/organize a more DUT/standard specific conditions to handle and this can be done in Conditions page.
- For example, Displayport DUT required Tests to be carried out on
 - o 5.4Gbps, 2.7Gbps, 1.62Gbps Data rate.
 - o 400 mV, 600 mV, 800 mV Output level.
- Condition page allow you to create a test plan/permutation table so that you only need to write code for a test once and then the software can permutate through all the conditions defined. You can also defined "condition-specific limit" for the test.
- To add condition, click "Add Condition"
- Add a Condition Group and the condition you'd like to permutate
- Example:

Test Conditions

New Condition Group: DataRate

Description: Data Rate

Condition To Add: RBR

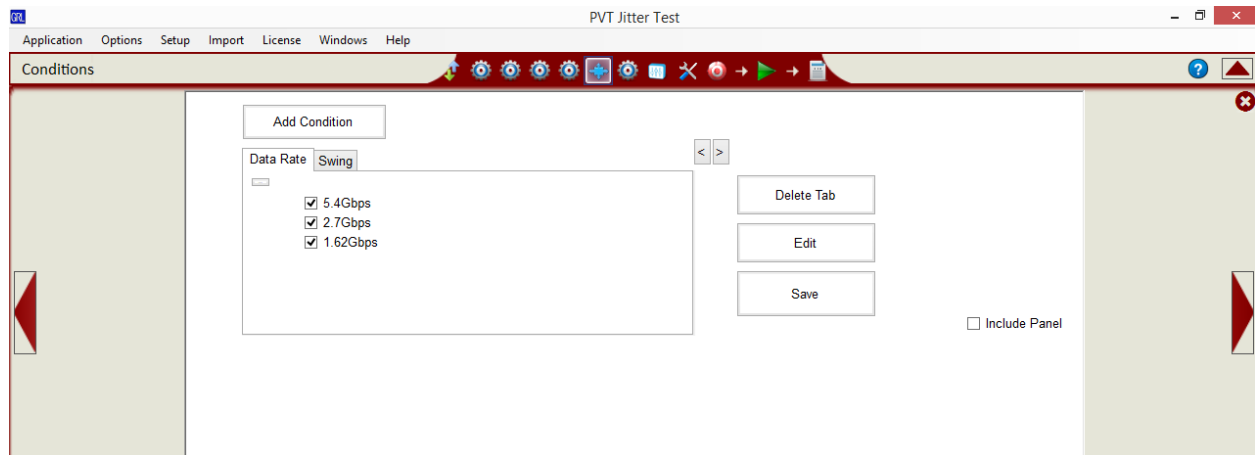
Description: 1.62Gbps

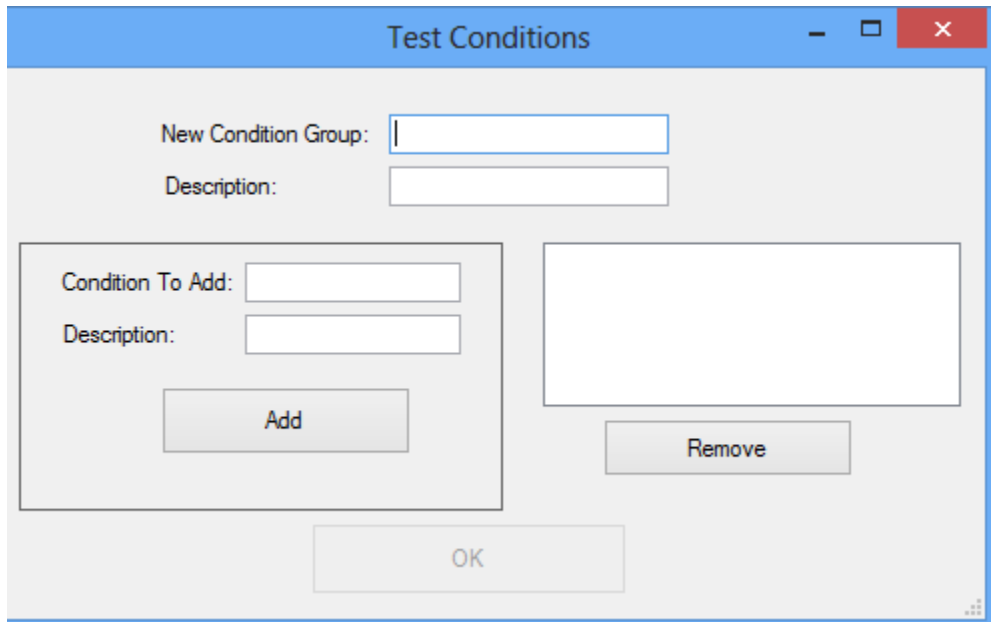
Add

HBR2
HBR
RBR

Remove Edit

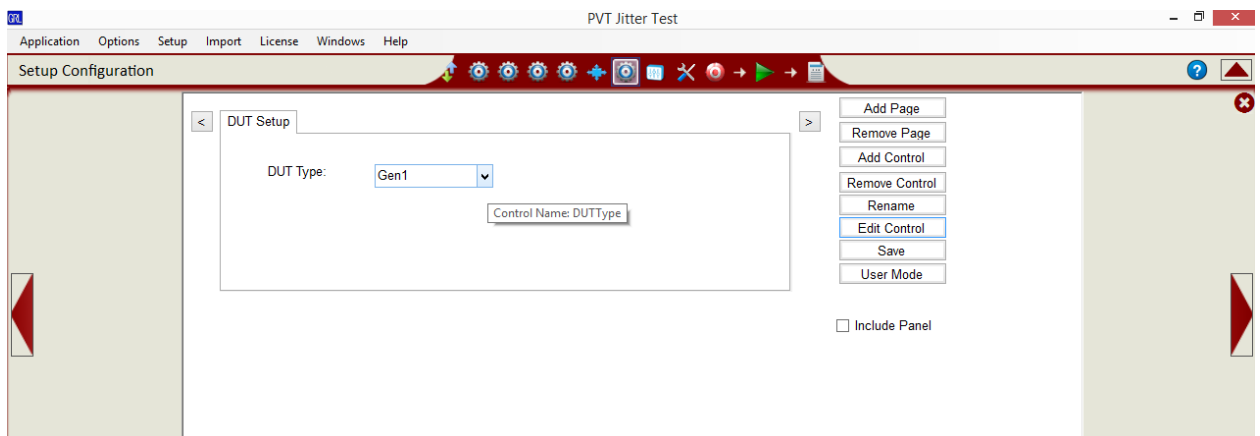
OK





8. Setup Configuration Page.

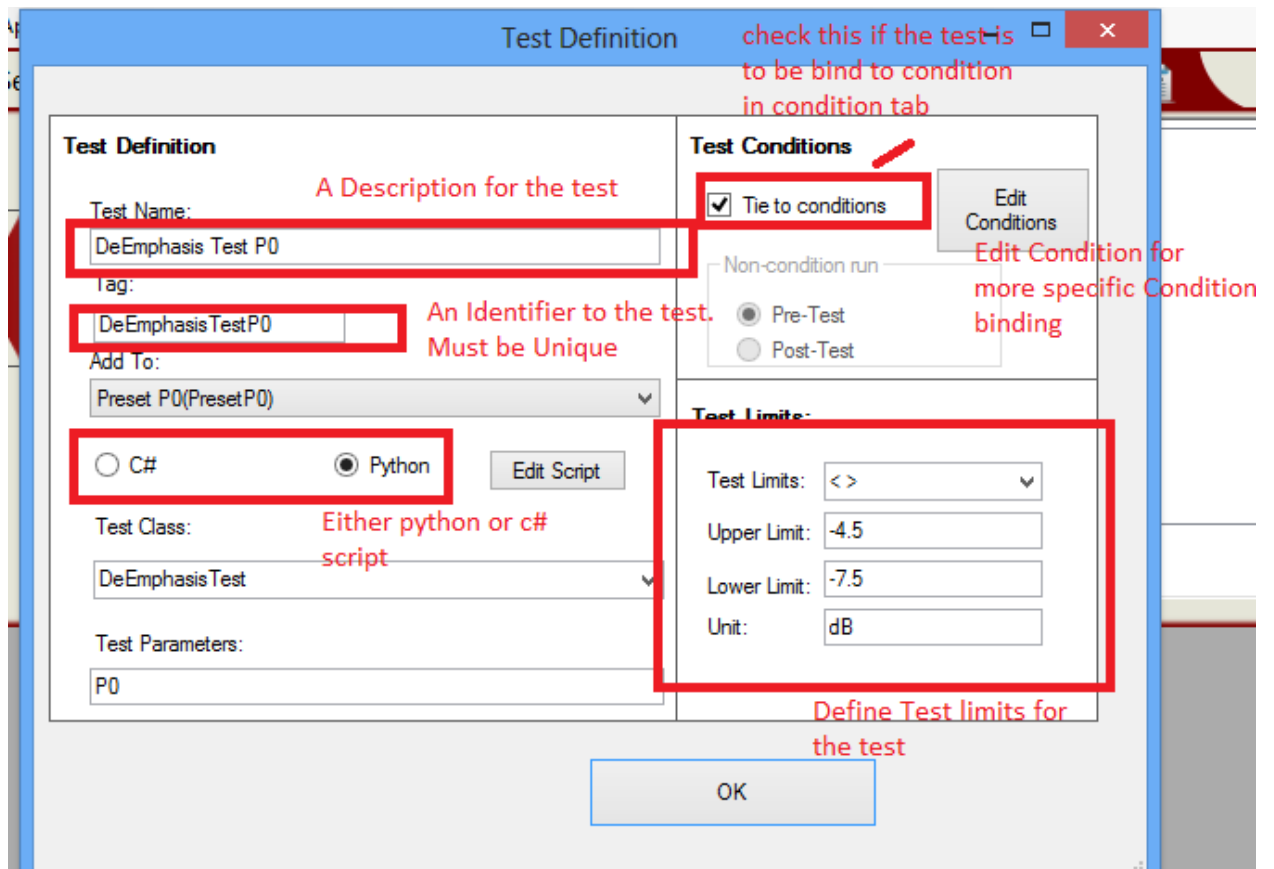
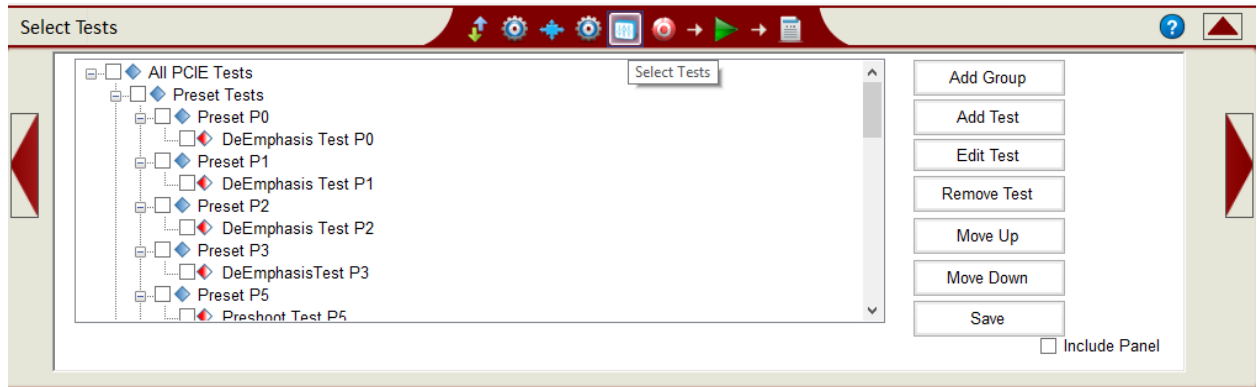
- This is the page developer can add custom GUI control components like label, textbox, combobox, TextBox.
- The control created can be accessed by `FWAppObj.GetProperty("ControlTag")`
- Switch to "Developer mode" when you like to add/remove GUI control
- Switch to "User Mode" to test the control added.



- For instance, in the above example, user has defined a control called "DUTType" with Option of "Gen1" and "Gen2".
- To access this control from script, user can use `FWAppObj.GetProperty("DUTType")`

9. Select Tests Tab:

- A Tab for user to select Test, meaning you as developer need to create tests.
- "Add Group" to add Group. Always organize your group well for optimized test flow.
- "Add Test" to Add Test. Additional info about Creating Test as below:



- To tie test with conditions defined earlier(described in 6), check “Tie to conditions” and click Edit Conditions and choose the conditions the test need to be run

The image shows two overlapping dialog boxes from the Granite River Labs software. The 'Test Definition' dialog is in the foreground, and the 'Test Condition Permutation' dialog is in the background.

Test Definition Dialog:

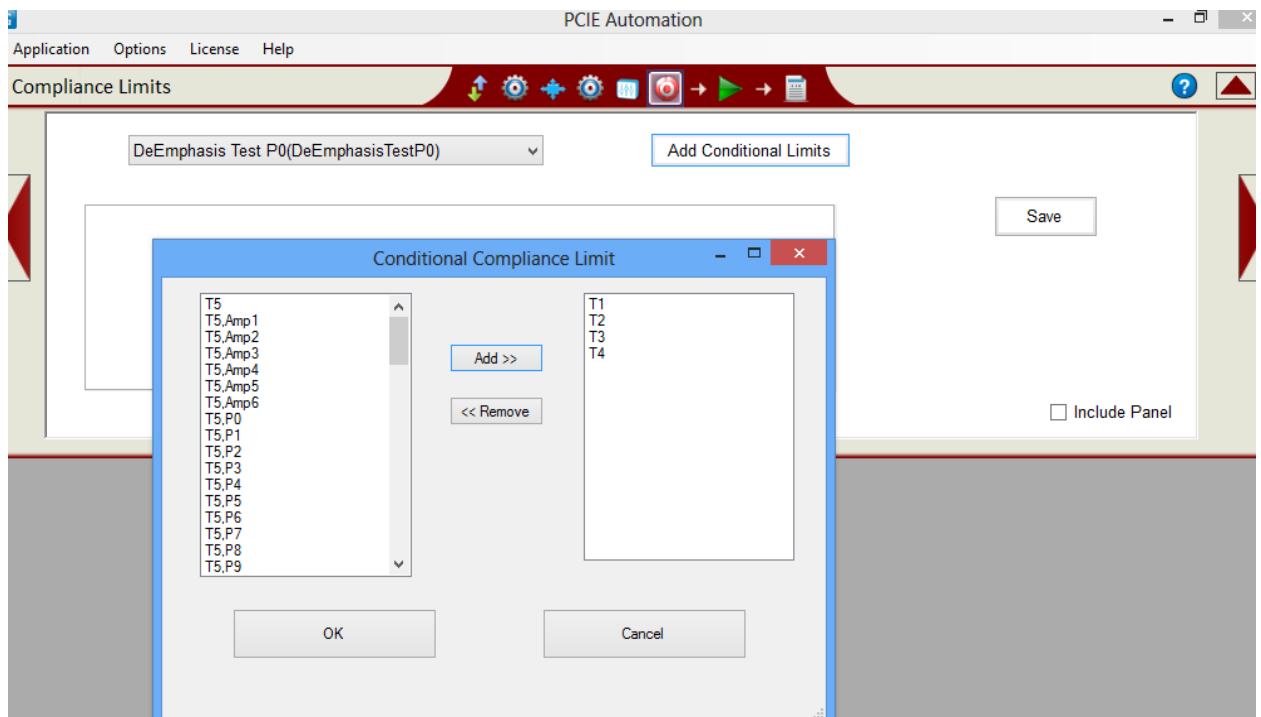
- Test Name:** Total Jitter Test
- Tag:** TotalJitterTest
- Add To:** All Test(AllTest)
- Language:** C# (selected), Python, Visual
- Test Class:** TotalJitterTest
- Test Parameters:** (empty field)
- Connection Tag:** None
- Test Conditions:** ☒ Tie to conditions
- Non-condition run:** ☒ Pre-Test, ☐ Post-Test
- Test Limits:** ☐ Informative Test
- Upper Limit:** 0.5
- Lower Limit:** (empty)
- Unit:** UI
- Lock State:** SuperDev

Test Condition Permutation Dialog:

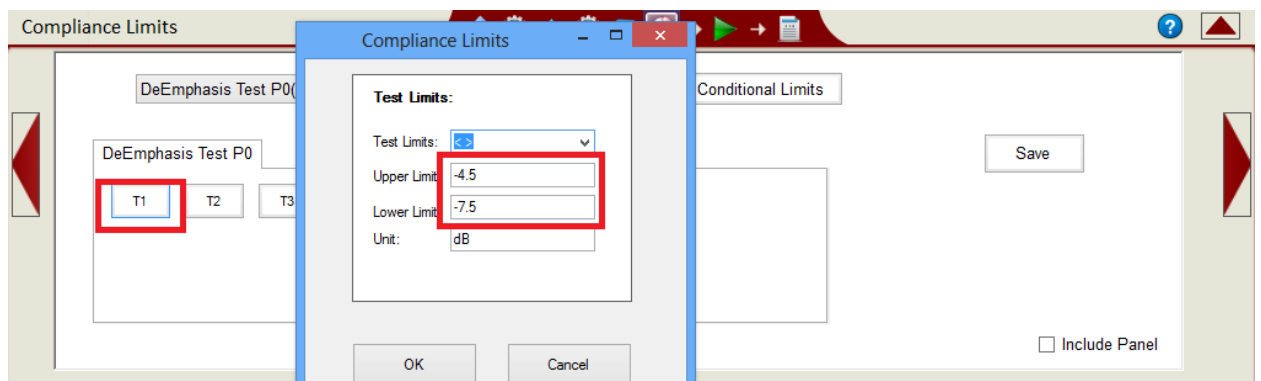
- All Combination:** ☒
- Data Rate:** Swing
- Conditions:**
 - ☒ All
 - ☐ Any
 - ☐ Highest
 - ☐ Lowest
 - ☐ HBR2
 - ☐ HBR
 - ☐ RBR
- Generate Conditions:** (button)
- Conditions List:**
 - HBR2_Level1
 - HBR2_Level2
 - HBR2_Level3
 - HBR_Level1
 - HBR_Level2
 - HBR_Level3
 - RBR_Level1
 - RBR_Level2
 - RBR_Level3
- Add/Remove Buttons:** << Add, Remove >>

10. Compliance Limit Tab

- Allow you to customize compliance limit for each test condition.
- The option is only available for test that is defined as “Tied to condition”
- To override a limit, click “Add Conditional Limits and then select the condition to be overridden.

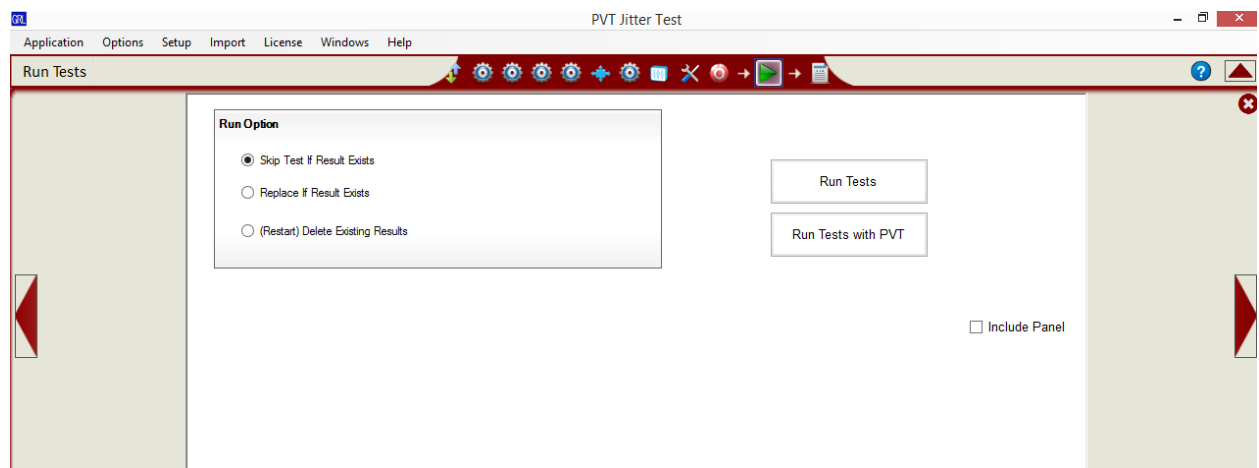


- To Change the limit, select the condition and then change the limits.



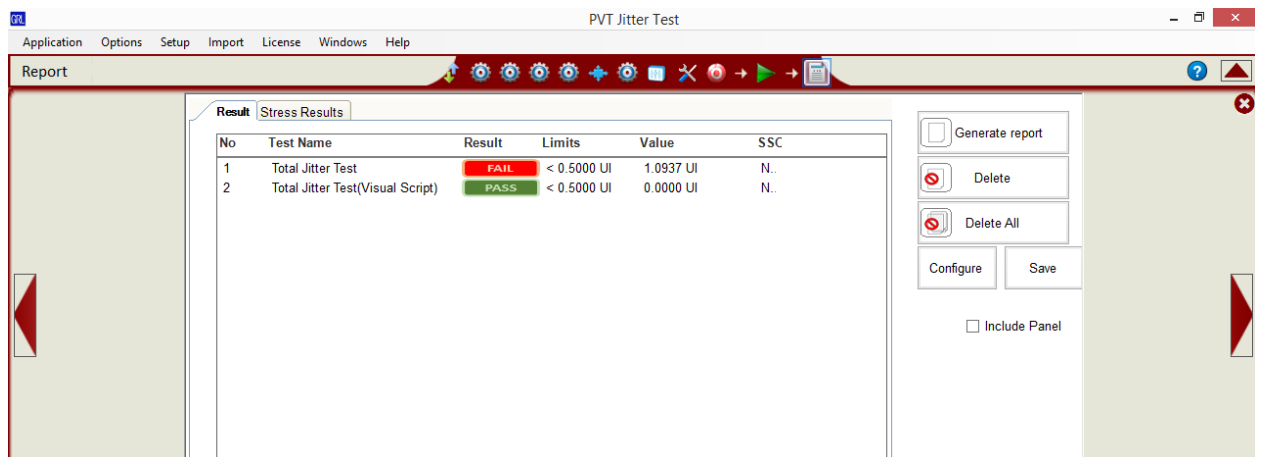
11. Run Test Tab

- The Tab that initiate the test run.
- “Skip Test If Result Exists” – Skip the test if test result already exist. Allow you to resume test from previous test.
- “Replace if Result Exists”- if Test Result exist, the test is being rerun and new result will replace old one.
- “Deleting Existing Results”- Delete all results before test started so that you get a fresh report.
- “Run Tests” will run all the tests in a single trial.
- “Run Tests” will run all the tests with all the conditions defined in PVT Configuration page.



12. Report Tab

- Show test result from Previous Run with test condition shown.
- To Generate Full report, click “Generate Report”. Example report is attached.
- To Delete Item, select the item and click “Delete”



Sample report:

PVT Jitter Test Stress Test Report(Temperature=Temperature_1,Voltage=Voltage_2)

No	TestName	Limits	Value	Results	SSC
1	Total Jitter Test	< 0.5000 UI	1.1035 UI	Fail	
2	Total Jitter Test(Visual Script)	< 0.5000 UI	0.0000 UI	Pass	

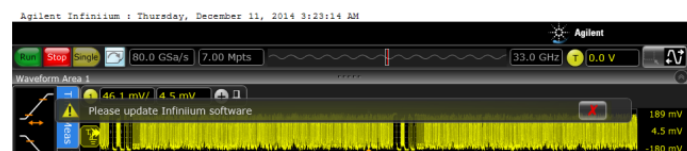
PVT Jitter Test Stress Test
Report(Temperature=Temperature_1,Voltage=Voltage_2)

1. Total Jitter Test

Test Limits : < 0.5000 UI

Result : 1.1035

Screen shots



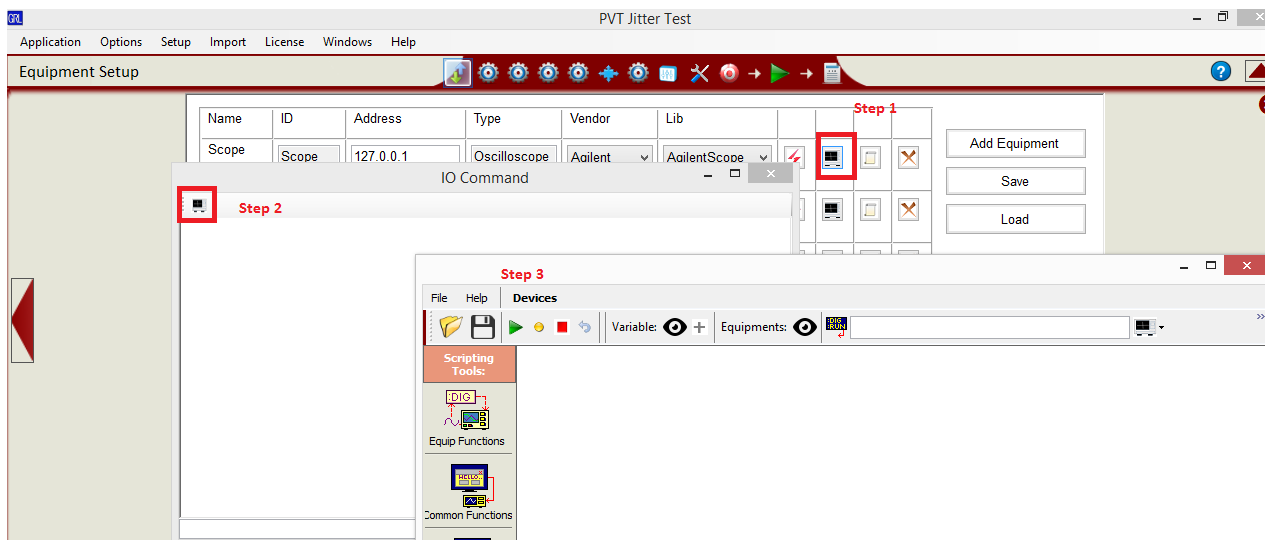
Chapter 2: Using the GRL Visual Scripting Tool

While GRL KayaQ framework utilize third party tools for instance python and Visual studio for C# scripts, the framework is also integrated with GRL proprietary Visual scripting tool. Visual scripting tool is a user interactive based tool that allow developer to create scripts in a flow chart like manner.

1. Launch Visual script tool:

a) Visual script tool can be launched by:

- i) Click on the  on Equipment page and then  again. Or
- ii) Click Edit script when the option to Edit script is available



Test Definition

Test Name:

Total Jitter Test

Tag:

TotalJitterTest

Add To:

All Test(AllTest)

C#

Python

Visual

Edit Script

Edit Script

Test Class:

TotalJitterTest

Test Parameters:

Connection Tag:

None

Test Conditions

Tie to conditions

Edit Conditions

Non-condition run

Pre-Test

Post-Test

Test Limits:

Informative Test

Test Limits: <

Upper Limit: 0.5

Lower Limit:

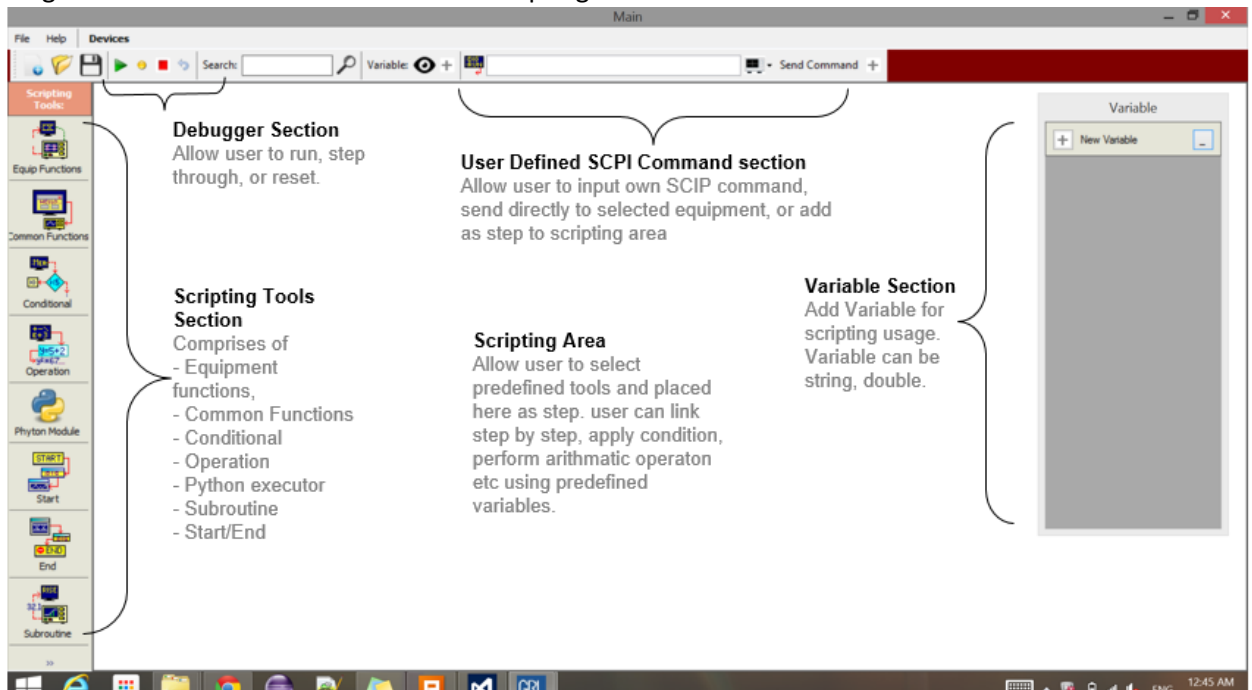
Unit: UI

Lock State:

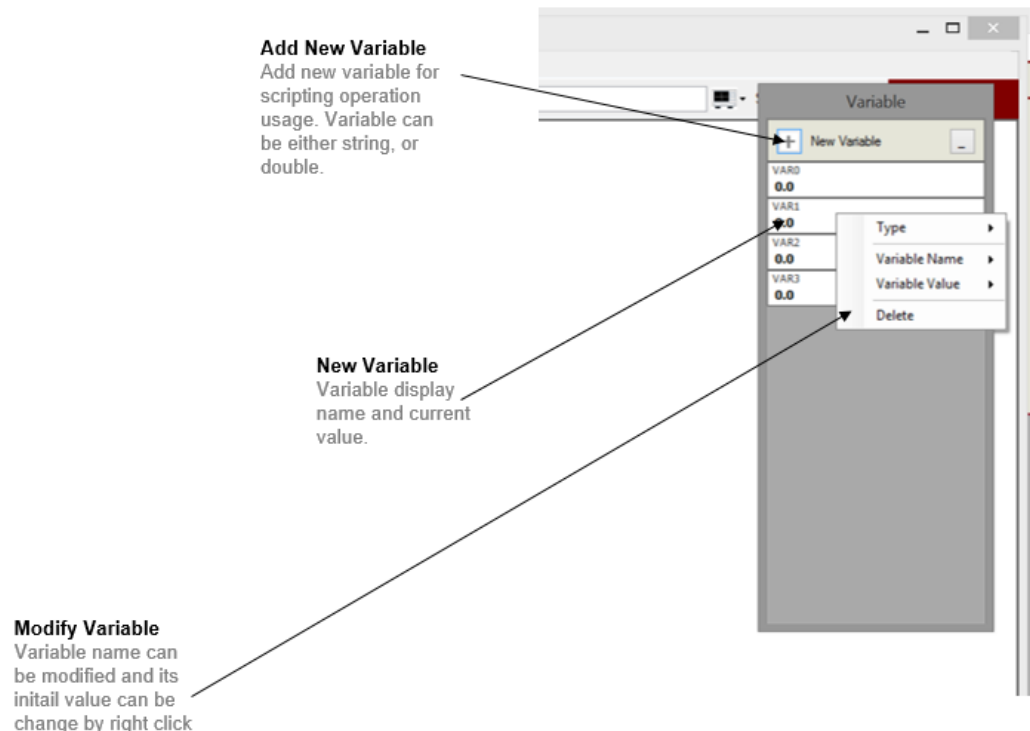
SuperDeveloper

OK

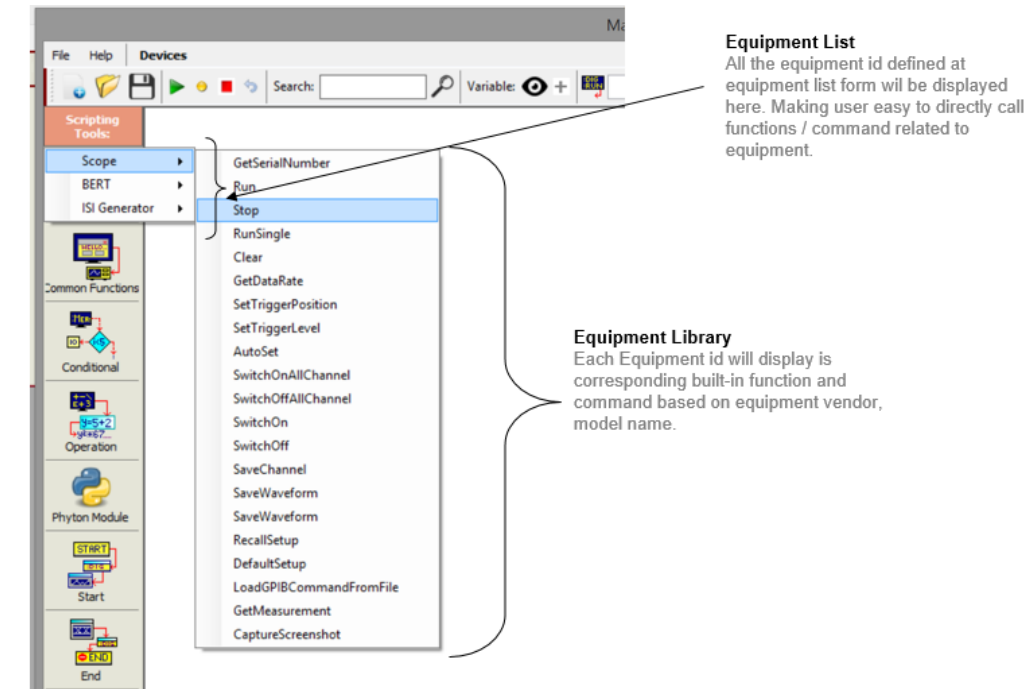
2. Diagram below describe features of GRL scripting tool



3. User can stored local variables in visual tool as shown below:

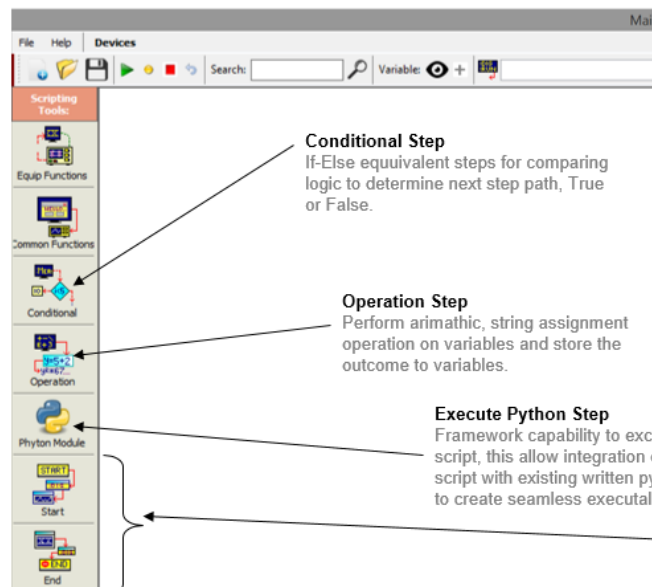


4. To add a step node, user can click on the buttons at the left panels to view API available and select a node to add.



5. Other operational steps include

- Start Node
- End Node
- If-Else Node
- Node to call Python Script
- Common arithmetic node



6. User can also debug and step when using visual script.

