Keysight Technologies
Using BenchVue Test Flow to Create Test Sequences without Programming

Application Note

KEYSIGHT
TECHNOLOGIES

Unlocking Measurement Insights

Introduction

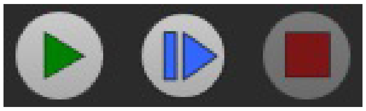
Keysight BenchVue software lets you connect, control and capture data from your choice of an ever-growing list of supported Keysight instruments—digital multimeters, oscilloscopes, power supplies, function generators, and RF signal sources and analyzers, just to name a few. The BenchVue Test Flow application allows you to create sequences of setups and measurements, all without having to write a computer program. This application note explains some of the many things you can do with the Test Flow application, including:

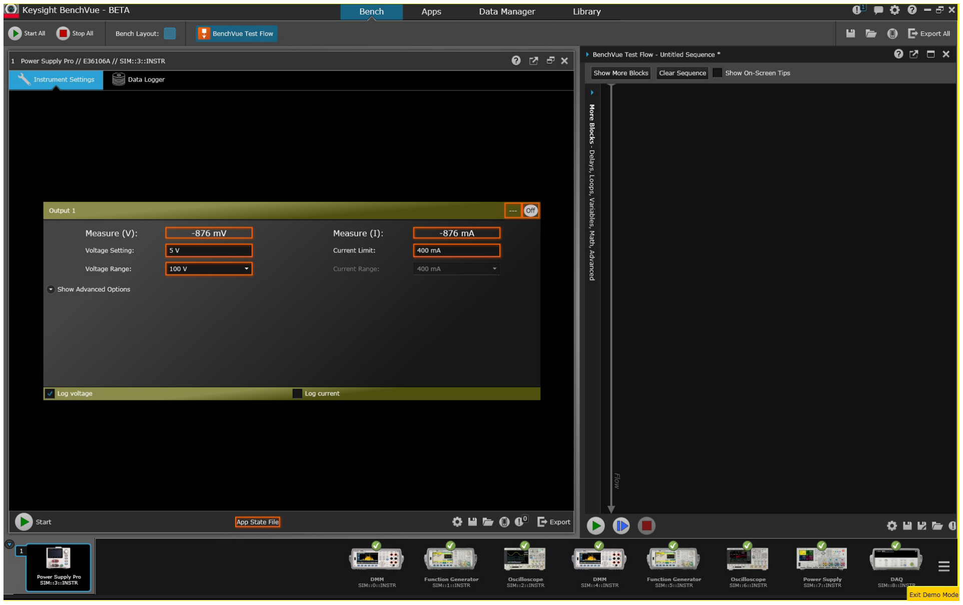
- Dragging and dropping instrument settings and retrieving measurements into a sequence
- Creating loops, delays, prompts, and simple “if-then” conditions
- Using mathematical operations
- Automatically exporting data and images to Microsoft Excel, MATLAB, CSV or Microsoft Word files
- Controlling equipment not on the BenchVue supported instrument list
- Using instrument commands not already built in to BenchVue

Example: Powering up a board

Suppose you want to power-up a circuit board and perform some basic measurements. You might want to apply power, wait a short time for circuit elements to stabilize, measure the current drain, and immediately stop if the current drain is too high, also outputting a warning message. With Test Flow, this is an easy task.

Figure 1 shows a Power Supply Application running on the left side of the BenchVue screen and an empty Test Flow sequence workspace on the right. A long downward arrow labeled “Flow” tells you that execution will be from top to bottom. Anything that can be “automated” (i.e., used in Test Flow) is highlighted by an orange rectangle. Any object placed into the test sequence is called a “block.” The run controls are at the bottom of the Test Flow pane and in the following order from left to right: Start, Single Step and Stop:





Keysight BenchVue - BETA

1 Power Supply Pro // E36106A // SIM::3::INSTR

Output 1

Measure (V): -876 mV

Voltage Setting: 5 V

Voltage Range: 100 V

Measure (I): -876 mA

Current Limit: 400 mA

Current Range: 400 mA

☑ Log voltage

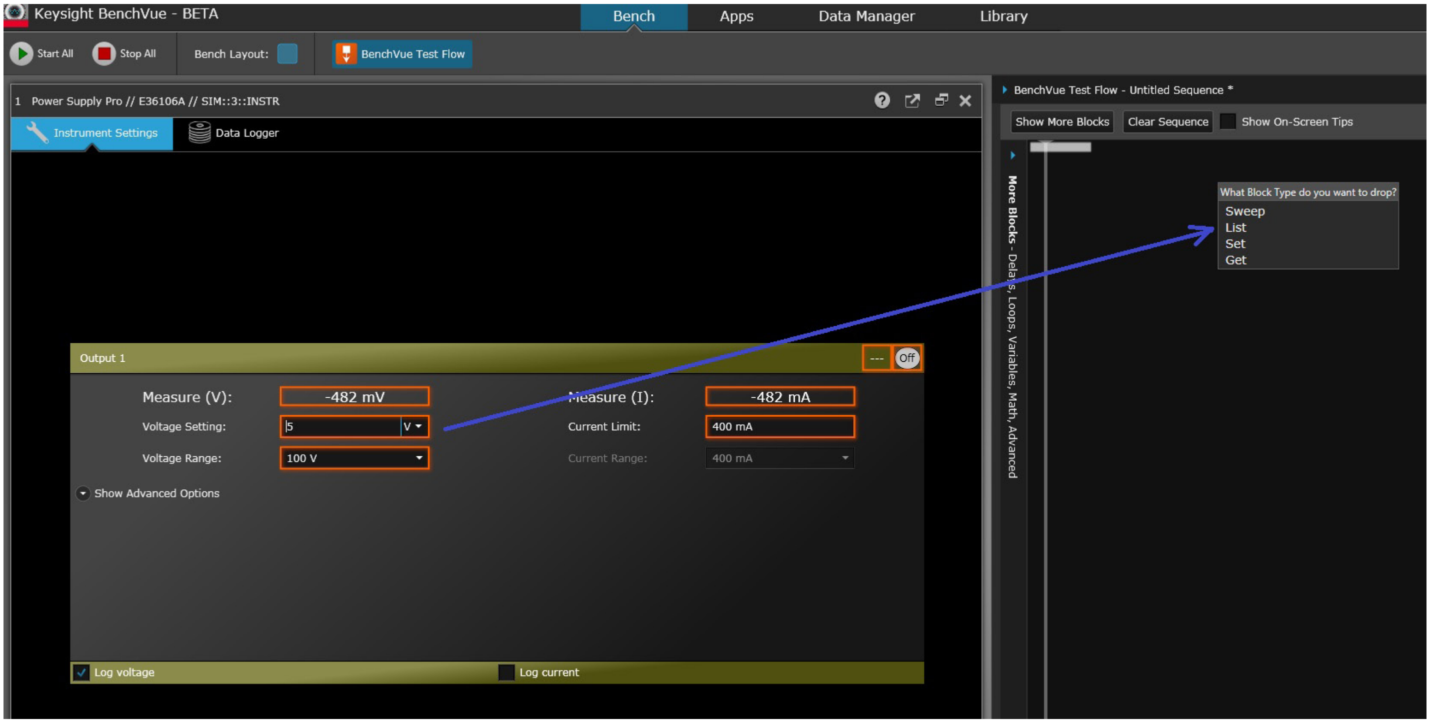
□ Log current

BenchVue Test Flow - Untitled Sequence *

□ Show On-Screen Tips

Figure 1. Power Supply Application (left panel) and a blank Test Flow sequence (right panel).

To start creating a sequence, left click on one of the orange boxes and drag it into the sequence workspace as shown in Figure 2. When you drop it, a dialog box pops up that lets you choose what kind of operation you want to do: Set a value, Get a value, create a List of values, or create a Sweep of values. Some boxes offer fewer choices, such as Set and Get only, and all measurements are only of the "Get" type. Some blocks do not offer a choice.


1 Power Supply Pro // E36106A // SIM::3::INSTR

BenchVue Test Flow - Untitled Sequence *

Measure (V): -482 mV

Voltage Setting: 5 V

Voltage Range: 100 V

Measure (I): -482 mA

Current Limit: 400 mA

Current Range: 400 mA

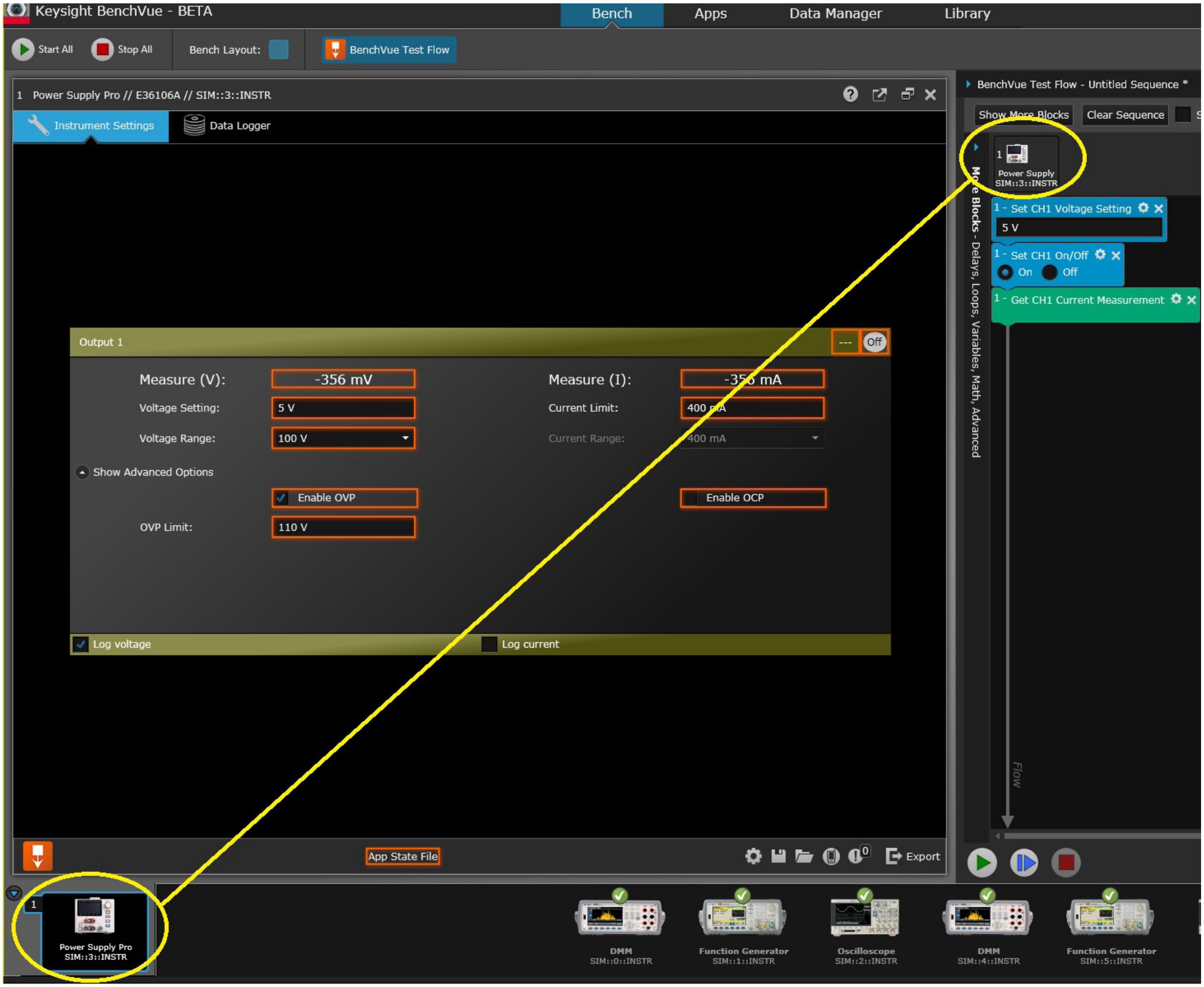
☑ Log voltage

□ Log current

What Block Type do you want to drop?
- Sweep
- List
- Set
- Get

Figure 2. A "Voltage Setting" block can be dragged into the Test Flow workspace and its type chosen (e.g., "Set," in this case).

To have the supply output a voltage, its output must be turned on. To do this, drag the "On/Off" button (Off) into the Test Flow sequence and change its state from Off to On. Note that no actual state changes occur on the instrument as you do this. Finally, measure the current drain by dragging the "Measure(I)" box into the sequence.



1 Power Supply Pro // E36106A // SIM::3::INSTR

Output 1

Measure (V): -356 mV

Voltage Setting: 5 V

Voltage Range: 100 V

Measure (I): -356 mA

Current Limit: 400 mA

Current Range: 400 mA

Show Advanced Options

☑ Enable OVP

OVP Limit: 110 V

□ Enable OCP

☑ Log voltage

□ Log current

BenchVue Test Flow - Untitled Sequence *

1 Power Supply SIM::3::INSTR

1 - Set CH1 Voltage Setting: 5 V

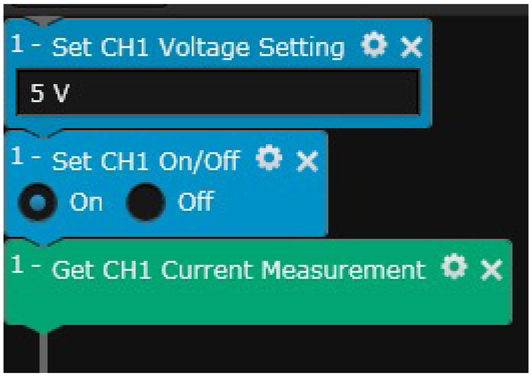
1 - Set CH1 On/Off: ● On ● Off

1 - Get CH1 Current Measurement

1 Power Supply Pro SIM::3::INSTR

Figure 3. When an instrument application is run, it is given a number starting at 1. That number is shown above the sequence and in each step that uses that instrument.

Once an operation from an application is placed in the Test Flow sequence, a picture of the instrument is shown above the sequence. Its position (e.g., 1, 2, 3, 4) is also shown. Each step that accesses that instrument refers to that number as shown in Figure 3.



1 - Set CH1 Voltage Setting

5 V

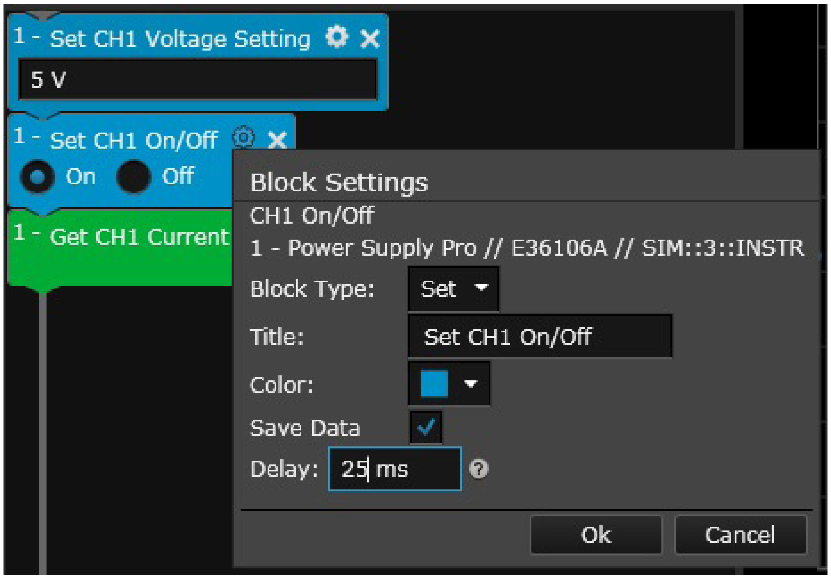
1 - Set CH1 On/Off

● On ● Off

1 - Get CH1 Current Measurement

Figure 4. This is a ready-to-run sequence that sets a power supply to 5 V, turns on its output and measures the resulting current.

The sequence to this point is shown in Figure 4. Although it could be run, no delays have yet been added. One way to add a delay is to use the gear icon that is present in many blocks. Some blocks provide a way to specify a delay that will be applied after each step, as shown in Figure 5. Another way to do this is to add a Delay Block. This block is kept in a section called "More Blocks," located to the left of the main sequence. Figure 6 shows all other groups of blocks that are available—basic, loops, variables, math, and advanced.



1 - Set CH1 Voltage Setting

5 V

1 - Set CH1 On/Off

● On ● Off

1 - Get CH1 Current

Block Settings

CH1 On/Off

1 - Power Supply Pro // E36106A // SIM::3::INSTR

Block Type: Set

Title: Set CH1 On/Off

Color: 🟦

Save Data ✓

Delay: 25 ms

Figure 5. Clicking the gear icon in a block brings up a dialog box that allows you to rename it, change its color, save any data that might be gathered, and apply a delay after the step is executed. In this case, a 25-ms delay is used to make sure circuit elements have stabilized before the sequence continues.

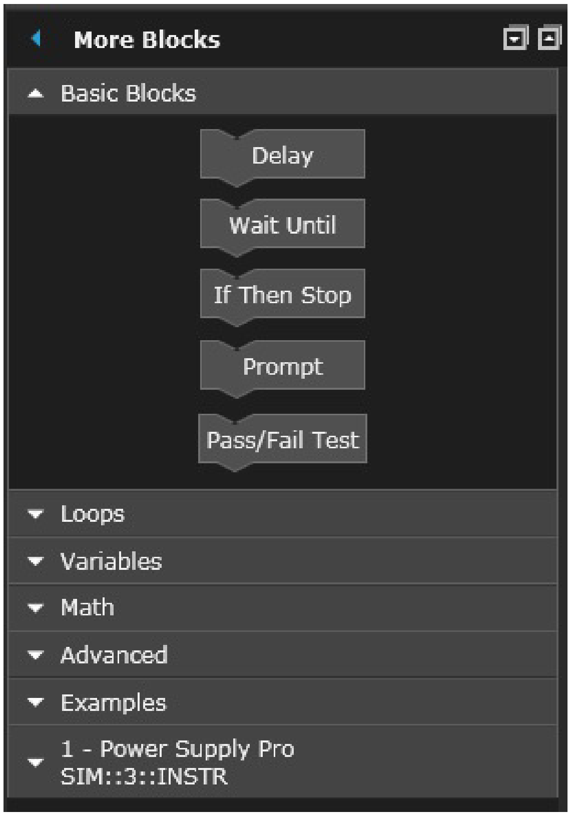
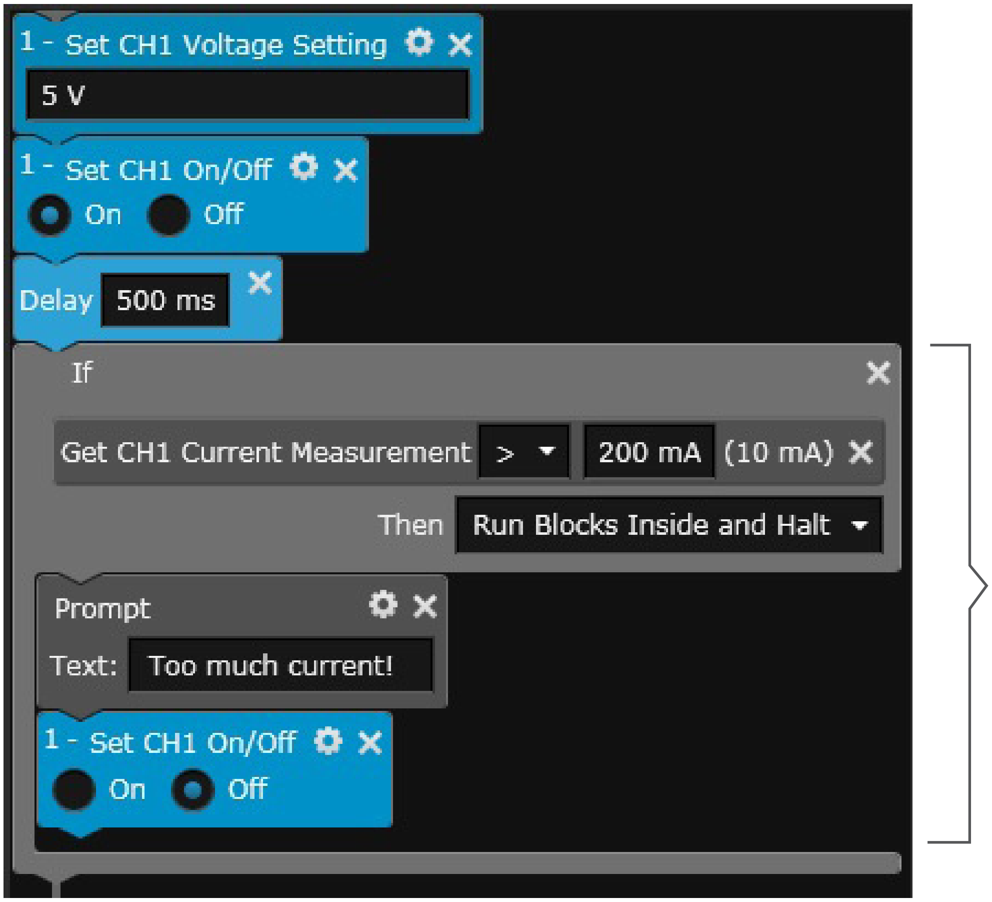


Figure 6. More Blocks tool bar.

The Basic Blocks section contains a Delay block which can be used if a more "visible" delay is desired than the one inside the gear icon. Finally, if the current drain was higher than desired, it might be desirable to output a prompt telling the user that the current was too high and to shut off the supply output. In this case, you can use an "If-Then-Stop" block, which allows you to place one or more conditions in the "If" section, and then choose what to do if the condition is met, whether run additional blocks and halt, or halt immediately (Figure 7).



1 - Set CH1 Voltage Setting
5 V

1 - Set CH1 On/Off
● On ○ Off

Delay 500 ms

If
Get CH1 Current Measurement > 200 mA (10 mA)
Then Run Blocks Inside and Halt

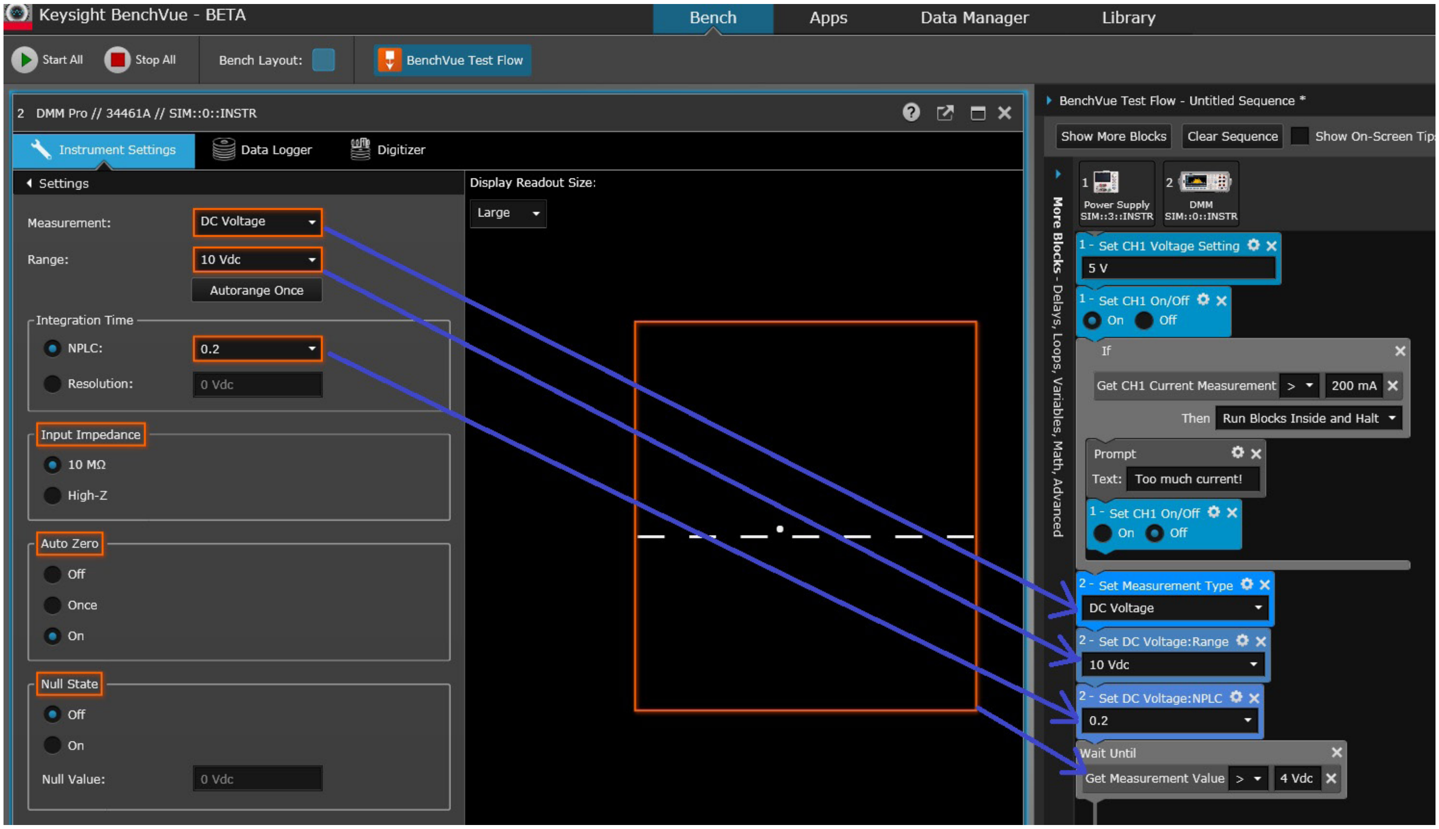
Prompt
Text: Too much current!

1 - Set CH1 On/Off
○ On ● Off

If the condition(s) are met, the Prompt and the Set Ch1 On/Off blocks will execute. In this case, when the sequence was run, the measured current was 10 mA, which is OK, so the sequence continued.

Figure 7. A 500-ms delay is added to allow circuits to stabilize. An "If-Then-Stop" block tests the current and execution stops if too much current is drawn.

Now suppose that the circuit-under-test had a DC output that required execution to pause until the measured voltage was greater than 4 V. One way to do this would be to use the "Wait Until" block, using a digital multimeter (DMM) measurement for the exit condition. First start the DMM application and drag some setup properties over to the Test Flow workspace. Then make a measurement repeatedly until the result is greater than 4 V, as shown in Figure 8. Note that this could run forever if the circuit is defective and the voltage never goes above 4 V!



Measurement: DC Voltage

Range: 10 Vdc

Integration Time

- ● NPLC: 0.2
- ○ Resolution: 0 Vdc

Input Impedance

- ● 10 MΩ
- ○ High-Z

Auto Zero

- ○ Off
- ○ Once
- ● On

Null State

- ● Off
- ○ On

Null Value: 0 Vdc

Display Readout Size: Large

1 - Set CH1 Voltage Setting: 5 V

1 - Set CH1 On/Off: ● On ○ Off

If Get CH1 Current Measurement > 200 mA Then Run Blocks Inside and Halt

Prompt Text: Too much current!

1 - Set CH1 On/Off: ○ On ● Off

2 - Set Measurement Type: DC Voltage

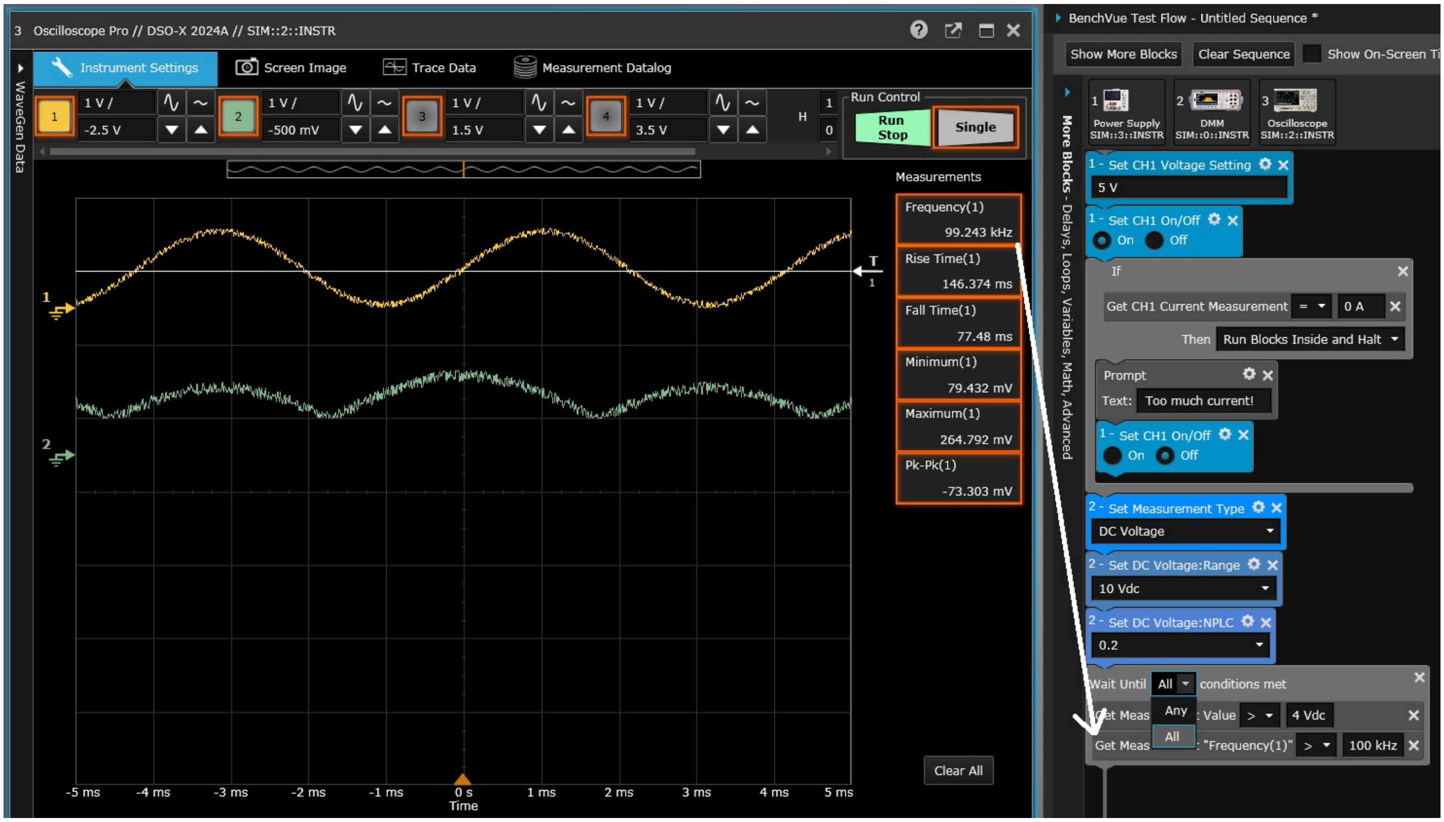
2 - Set DC Voltage:Range: 10 Vdc

2 - Set DC Voltage:NPLC: 0.2

Wait Until Get Measurement Value > 4 Vdc

Figure 8. Example of the use of the Wait-Until block.

When blocks contain conditions that are to be tested, such as in Figure 8 where Wait-Until and If-Then-Stop blocks are used, it is possible to use more than one condition. Suppose, for example, that in addition to a voltage measurement, it was desirable to wait until a certain AC signal frequency was greater than 100 kHz. Starting the oscilloscope application, a frequency measurement could also be dragged into the list of conditions. The choice can then be made as to whether ALL of the conditions must be true or ANY of the conditions must be true, as shown in Figure 9 and the close-up view of this block in Figure 10.



1: 1 V / -2.5 V
2: 1 V / -500 mV
3: 1 V / 1.5 V
4: 1 V / 3.5 V

Measurements

Frequency(1) 99.243 kHz
Rise Time(1) 146.374 ms
Fall Time(1) 77.48 ms
Minimum(1) 79.432 mV
Maximum(1) 264.792 mV
Pk-Pk(1) -73.303 mV

1 - Set CH1 Voltage Setting: 5 V

1 - Set CH1 On/Off: ● On ○ Off

If Get CH1 Current Measurement = 0 A

Then Run Blocks Inside and Halt

Prompt Text: Too much current!

1 - Set CH1 On/Off: ○ On ● Off

2 - Set Measurement Type: DC Voltage

2 - Set DC Voltage:Range: 10 Vdc

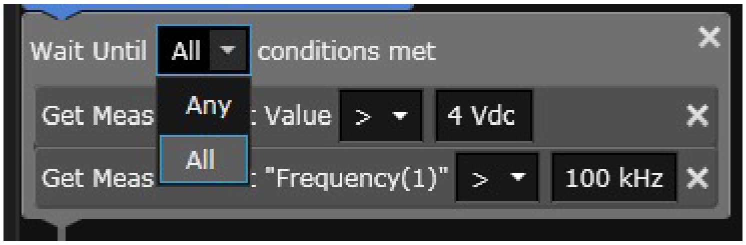
2 - Set DC Voltage:NPLC: 0.2

Wait Until All conditions met

Get Meas Any : Value > 4 Vdc

Get Meas All : "Frequency(1)" > 100 kHz

Figure 9. An oscilloscope measurement AND a DMM measurement are both used as conditions for testing to continue.


Wait Until All conditions met

Get Meas Any : Value > 4 Vdc

Get Meas All : "Frequency(1)" > 100 kHz

Figure 10. A close-up view of a block with multiple exit conditions. Note that measurements on the hardware are taken repeatedly until the exit conditions are true.

Loops

Figure 11 shows the various loops that are available in Test Flow. These cause the blocks inside to be run for a given Count, time (Duration), While conditions are true, or Until conditions become true. There is also a Forever loop that will run until the sequence is either aborted by an If-Then-Stop block or manually stopped.

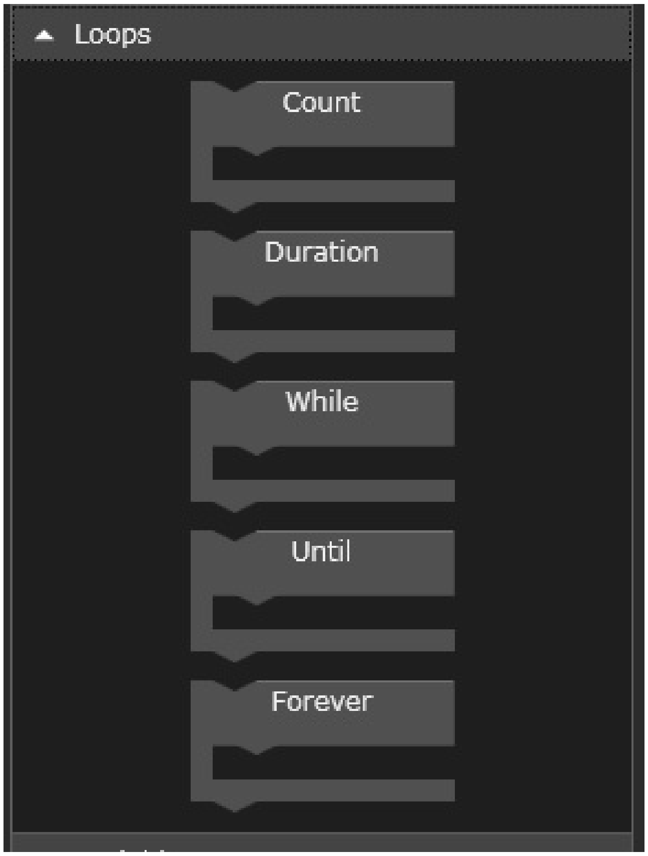


Figure 11. Loop blocks.

Variables

In the examples above, measurements such as “Get CH1 Current Measurement” have been dragged into test conditions where they can be used only once (i.e., any data that the measurement returns can’t be used for any other tests). Sometimes it’s desirable to take a reading and use it for more than one condition. For example, you might want to make sure that a single reading is greater than 4.75 V and less than 5.25 V. This can be done by creating variables, as shown in Figure 13. In this case, two variables were created by clicking the Create Variable button and naming them “volts” and “frequency.” The variables were dragged into the sequence and then the relevant measurement was dragged into the right side of the equation. The voltage measurements were made using a power supply and the frequency measurements were made using an oscilloscope.

Note that there are two choices for variables: Set or Output (Shown in Figure 12). Set provides a way to assign a variable to a quantity (e.g., a measurement, an instrument setting, another variable, or a constant), but it does not maintain the units (e.g., “V” or “A”) that may be associated with the right side of the equation. The “Output” block outputs the variable to the results table and allows units to be attached to the value.

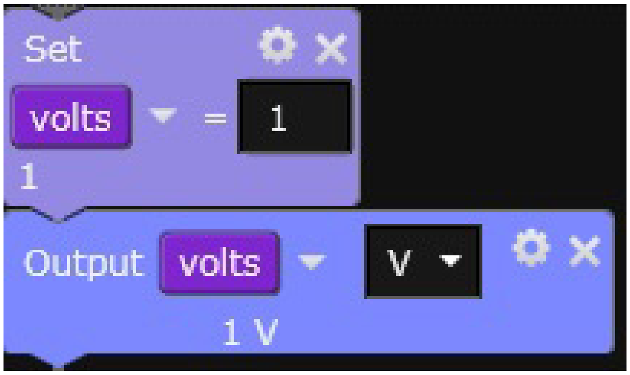
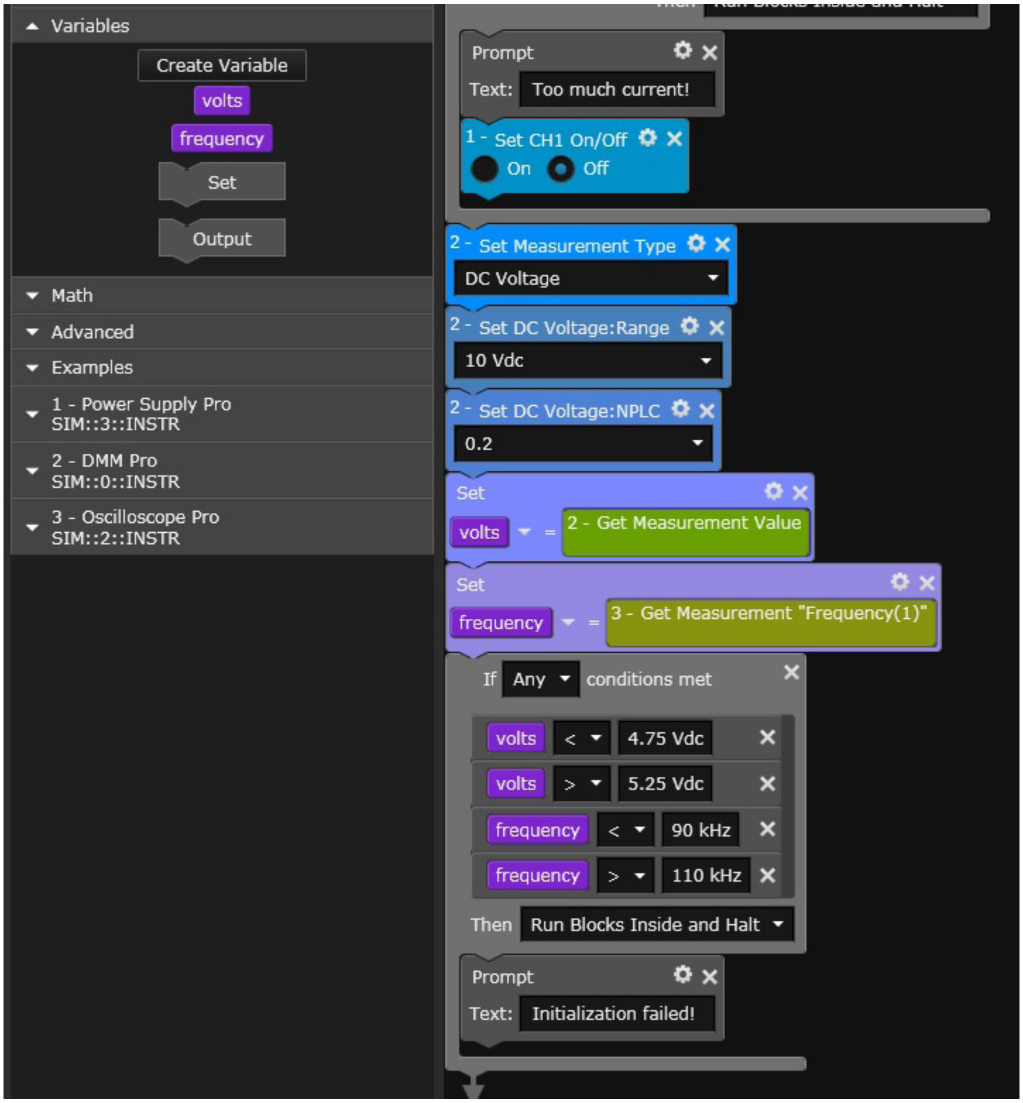


Figure 12. There are two choices for variables: Set or Output.

An If-Then-Stop block was chosen to use the two variables. By selecting Any at the top of the block, the Halt condition will be met if the voltage is outside the range 4.75 to 5.25 V, or the frequency is outside the range 90 to 110 kHz.



Prompt
Text: Too much current!

1 - Set CH1 On/Off
○ On ● Off

2 - Set Measurement Type
DC Voltage

2 - Set DC Voltage:Range
10 Vdc

2 - Set DC Voltage:NPLC
0.2

Set
volts = 2 - Get Measurement Value

Set
frequency = 3 - Get Measurement "Frequency(1)"

If Any conditions met

volts < 4.75 Vdc
volts > 5.25 Vdc
frequency < 90 kHz
frequency > 110 kHz

Then Run Blocks Inside and Halt

Prompt
Text: Initialization failed!

Figure 13. Variables “volts” and “frequency” can be created and used instead of real-time measurements.

Math Blocks

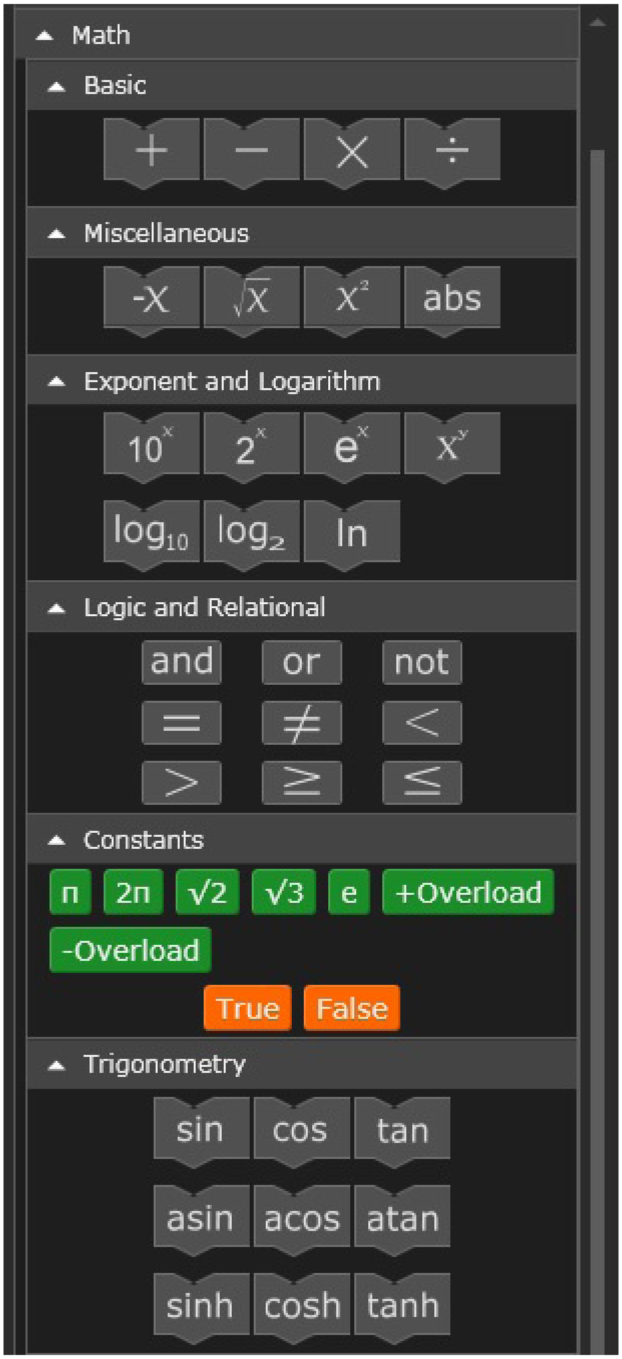
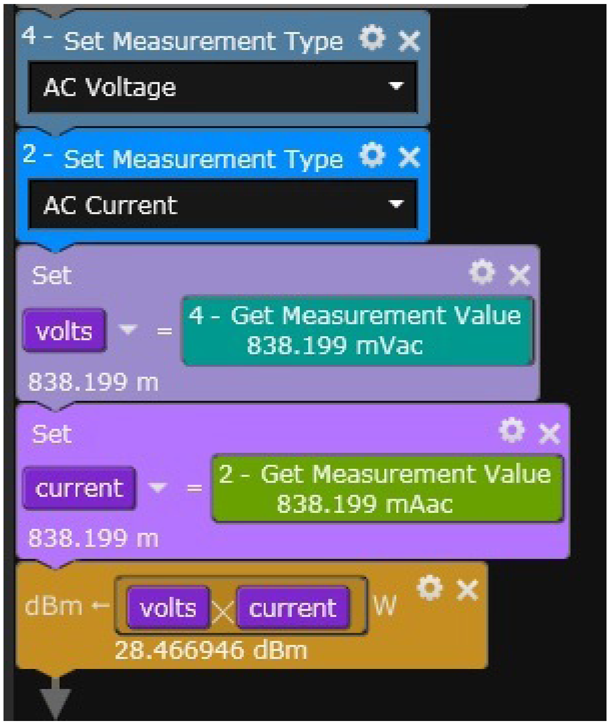


Figure 14. Math functions and constants.

There are many ways to create mathematical expressions in Test Flow. Figure 14 shows the available functions. These can be nested and used with variables or directly with measurements.

For example, suppose you are measuring voltage and current from two DMMs simultaneously and want to compute power in dBm. Figure 15 shows exactly what this would look like.



4 - Set Measurement Type
AC Voltage

2 - Set Measurement Type
AC Current

Set
volts = 4 - Get Measurement Value
838.199 mVac
838.199 m

Set
current = 2 - Get Measurement Value
838.199 mAac
838.199 m

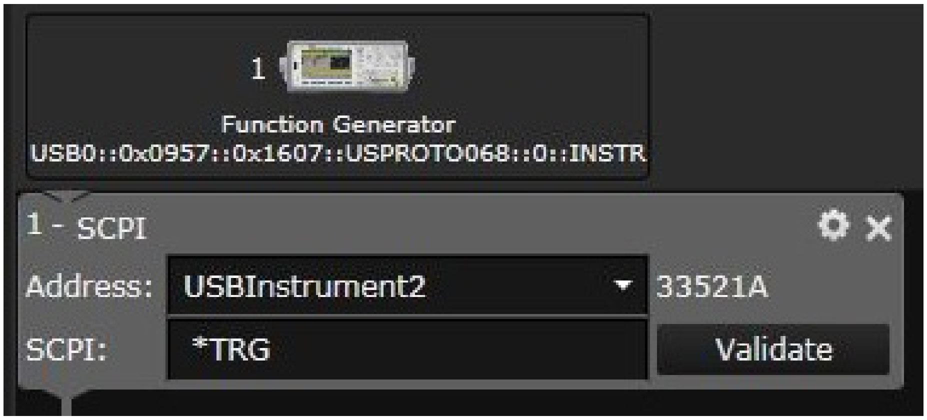
dBm ← volts × current W
28.466946 dBm

Figure 15. Here, one DMM (instrument #4) takes an AC voltage reading and puts the result into the variable "volts." The other DMM (instrument #2) takes an AC current reading and puts the result into the variable "current." The two variables, volts and current, are then dragged into the multiplication block, assigned the value "W" (for Watts). Next, that math function is dragged into the unit conversion block "dBm <- W." When the sequence is run, actual values are shown in each block.

SCPI Blocks

The applications that are provided with BenchVue are meant to provide coverage of the most commonly used functions of BenchVue supported instruments. In some applications; however, there are settings that are not implemented. The SCPI (Standard Commands for Programmable Instruments) block allows you to get around that limitation by sending and receiving commands in the native language of the instrument—SCPI (pronounced "skippy"). You can still drag and drop other commands from an application directly into the Test Flow sequence, and then add additional SCPI commands to do more advanced operations.

For example, the Function Generator application has no way to cause a manual trigger on a 33500A or 33600A True*form* waveform generator. But the SCPI command for a trigger is "*TRG." Figure 16 shows what it would look like to send this command in a Test Flow sequence.



Address: USBInstrument2

SCPI: *TRG

Figure 16. SCPI block example.

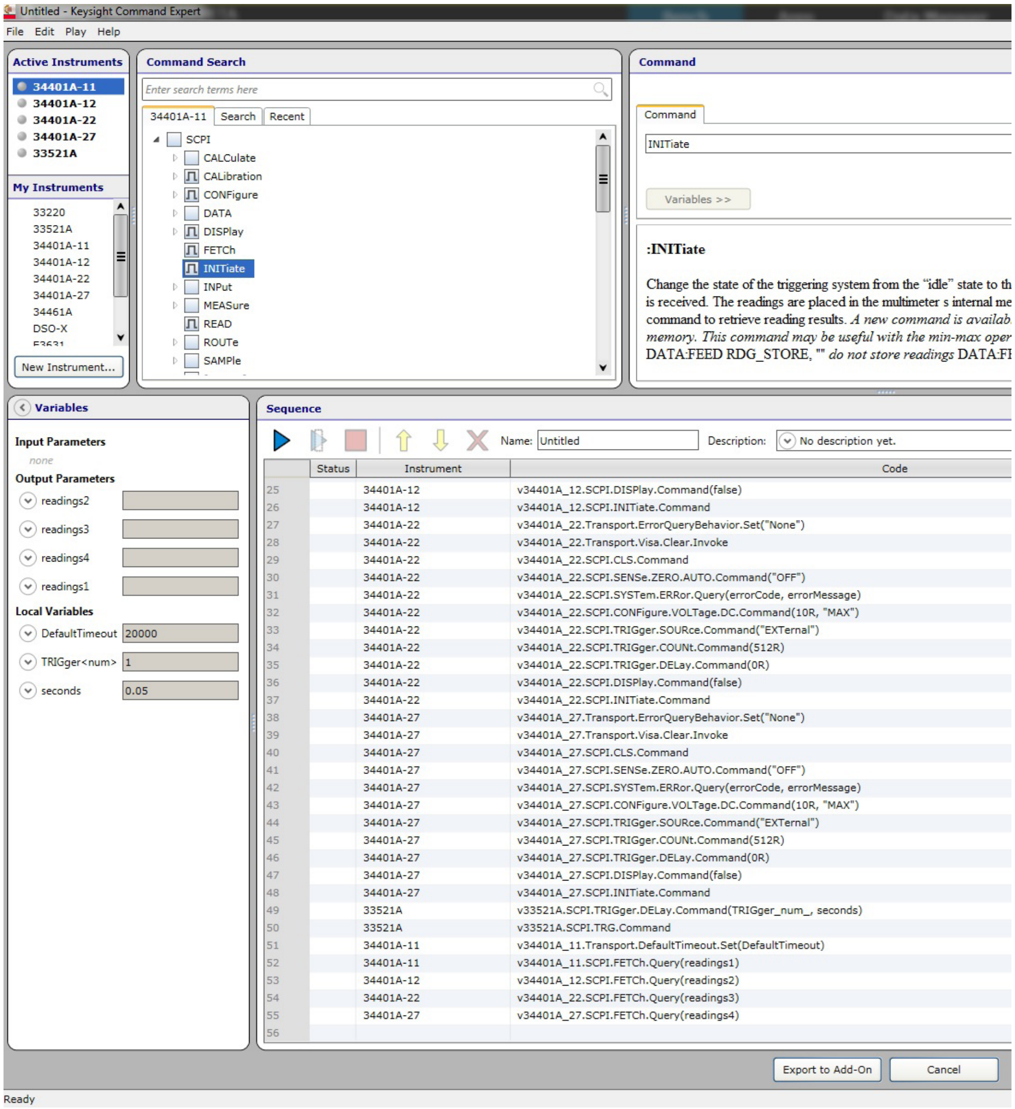
The Address field allows you to select any instrument that has been discovered. This means that it can be used on unsupported or non-Keysight instruments, as long as they use SCPI, as well as supported instruments. In Figure 16, notice that there is a Validate button on the side. This can be used to test the SCPI command immediately without running the entire sequence. This button sends the command, followed by a "SYST:ERR?" query. Consequently, if you are using this command with unsupported instruments not made by Keysight, you must make sure that the "SYST:ERR?" command is supported. This error check is also run when the SCPI block is executed during a normal run of a Test Flow sequence. If that is not desirable, it can be turned off by unchecking the box in the gear icon labeled "Clear Errors."

SCPI commands that return data from an instrument always have "?" as the last character. When the SCPI block sees "?," it allows the selection of several data types: Boolean, Double, String (the default), Image, Double Array and Byte Array. Consult the manual for the instrument you are using to see what data type to select.

When getting images from your instrument, note that the image data type requires that the data be in "IEEEBinaryType. BinaryType_UI1" format. A .bmp file is created as a result of executing this block.

Command Expert

Sometimes you might need to send a sequence of SCPI commands or may not know what those commands are. If you are unsure of which commands to use with various instruments, you can use Keysight's Command Expert software to help you create commands or command sequences that can then be exported to many programming languages. In BenchVue, the sequence can be created and then brought into Test Flow.



Command: INITiate

:INITiate

Change the state of the triggering system from the "idle" state to th
is received. The readings are placed in the multimeter s internal me
command to retrieve reading results. *A new command is availab*
memory. This command may be useful with the min-max oper
DATA:FEED RDG_STORE, "" *do not store readings* DATA:FI

Input Parameters
none

Output Parameters
readings2
readings3
readings4
readings1

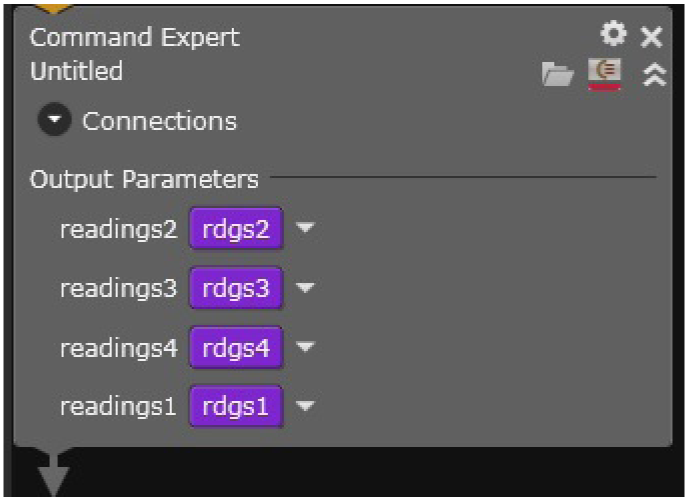
Local Variables
DefaultTimeout 20000
TRIGger<num> 1
seconds 0.05

Name: Untitled   Description: No description yet.

	Status	Instrument	Code
25		34401A-12	v34401A_12.SCPI.DISPlay.Command(false)
26		34401A-12	v34401A_12.SCPI.INITiate.Command
27		34401A-22	v34401A_22.Transport.ErrorQueryBehavior.Set("None")
28		34401A-22	v34401A_22.Transport.Visa.Clear.Invoke
29		34401A-22	v34401A_22.SCPI.CLS.Command
30		34401A-22	v34401A_22.SCPI.SENSe.ZERO.AUTO.Command("OFF")
31		34401A-22	v34401A_22.SCPI.SYSTem.ERRor.Query(errorCode, errorMessage)
32		34401A-22	v34401A_22.SCPI.CONFigure.VOLTage.DC.Command(10R, "MAX")
33		34401A-22	v34401A_22.SCPI.TRIGger.SOURce.Command("EXTernal")
34		34401A-22	v34401A_22.SCPI.TRIGger.COUNt.Command(512R)
35		34401A-22	v34401A_22.SCPI.TRIGger.DELay.Command(0R)
36		34401A-22	v34401A_22.SCPI.DISPlay.Command(false)
37		34401A-22	v34401A_22.SCPI.INITiate.Command
38		34401A-27	v34401A_27.Transport.ErrorQueryBehavior.Set("None")
39		34401A-27	v34401A_27.Transport.Visa.Clear.Invoke
40		34401A-27	v34401A_27.SCPI.CLS.Command
41		34401A-27	v34401A_27.SCPI.SENSe.ZERO.AUTO.Command("OFF")
42		34401A-27	v34401A_27.SCPI.SYSTem.ERRor.Query(errorCode, errorMessage)
43		34401A-27	v34401A_27.SCPI.CONFigure.VOLTage.DC.Command(10R, "MAX")
44		34401A-27	v34401A_27.SCPI.TRIGger.SOURce.Command("EXTernal")
45		34401A-27	v34401A_27.SCPI.TRIGger.COUNt.Command(512R)
46		34401A-27	v34401A_27.SCPI.TRIGger.DELay.Command(0R)
47		34401A-27	v34401A_27.SCPI.DISPlay.Command(false)
48		34401A-27	v34401A_27.SCPI.INITiate.Command
49		33521A	v33521A.SCPI.TRIGger.DELay.Command(TRIGger_num_, seconds)
50		33521A	v33521A.SCPI.TRG.Command
51		34401A-11	v34401A_11.Transport.DefaultTimeout.Set(DefaultTimeout)
52		34401A-11	v34401A_11.SCPI.FETCh.Query(readings1)
53		34401A-12	v34401A_12.SCPI.FETCh.Query(readings2)
54		34401A-22	v34401A_22.SCPI.FETCh.Query(readings3)
55		34401A-27	v34401A_27.SCPI.FETCh.Query(readings4)
56			

Figure 17. Last portion of a Command Expert sequence that sets up four 34401A DMMs for simultaneous readings. By starting this from Test Flow and then clicking "Export to Add-On," the sequence becomes part of Test Flow.

Consider the following example of how to work around a common Keysight 34401A DMM challenge—the inability to use the external trigger in BenchVue. Assume it was desired to externally trigger four 34401A DMMs from a function generator and then bring the data back into Test Flow where it could be exported to Excel and plotted. Since these particular DMMs prevent further communication until all readings have been fetched, all four were set up to expect 512 readings, and additional SCPI commands were sent to configure the DMMs correctly for this test, followed by an "INIT" command to place each DMM into a wait-for-trigger state. Finally, a "FETC?" command was issued to all four DMMs sequentially to extract their readings. Figure 17 shows how the last portion of the sequence looks in Command Expert. Figure 18 shows how the Command Expert block is built in BenchVue to return the readings.



Output Parameters

readings2 rdgs2

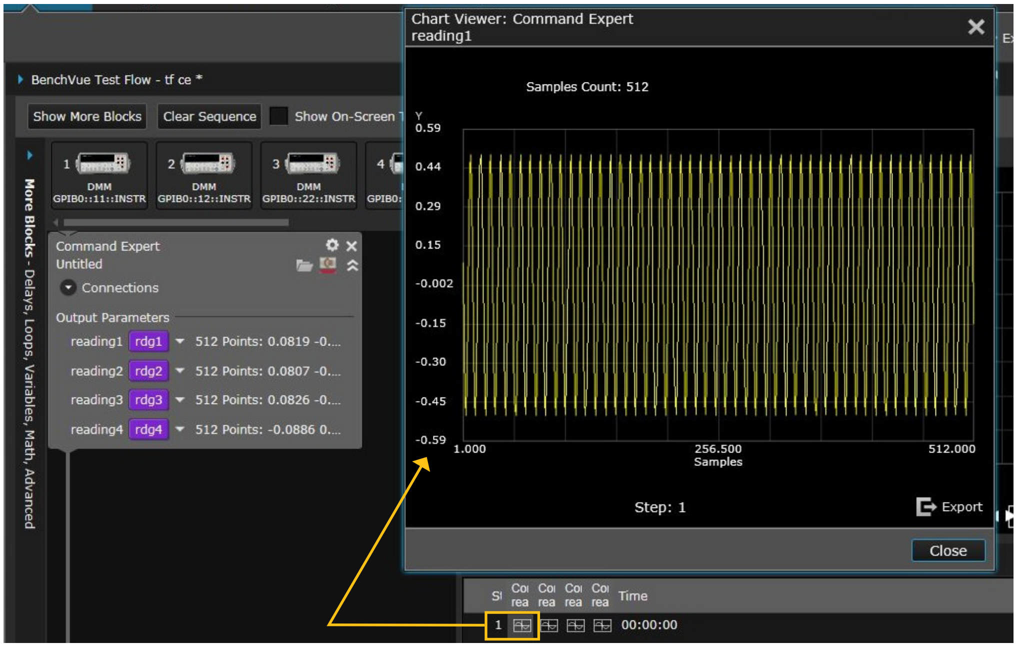
readings3 rdgs3

readings4 rdgs4

readings1 rdgs1

Figure 18. A completed sequence that has returned variables looks like this screen capture. Variables rdgs1-4 were created and then dragged into the block.

In Test Flow, the readings were put into four variables and all result data was exported to Excel using the export block. The data from each of the DMMs can be seen in the Results Table View in Figure 19 to the right of the Test Flow workspace. All steps that return data are shown here. Double clicking on the Trace Data icon brings up a chart of the data that can then be exported to Excel, Word, CSV, or MATLAB. In this example, a sine wave was fed into the four DMMs, so all charts are identical. Exports can also be included in the sequence so that this action is peromed automatically using the “Export” block.



Output Parameters

reading1 rdg1 512 Points: 0.0819 -0....

reading2 rdg2 512 Points: 0.0807 -0....

reading3 rdg3 512 Points: 0.0826 -0....

reading4 rdg4 512 Points: -0.0886 0....

St	Coı rea	Coı rea	Coı rea	Coı rea	Time
1	⊞	⊞	⊞	⊞	00:00:00

Figure 19. Result data is stored in a table. Click the Trace Data icon to pop up the trace data for export. An “Export” block in Test Flow is also available if the data is to be exported automatically from the test sequence.

A video of this experiment, as well as other BenchVue videos is available at **www.keysight.com/find/BenchVue_videos**.

More information about Command Expert can be found at **www.keysight.com/find/commandexpert**.

Chart Data

Another feature of Test Flow is the Chart windows. The Results table in Figure 19 can also be seen graphically in the Line Chart view, as shown in Figure 20.

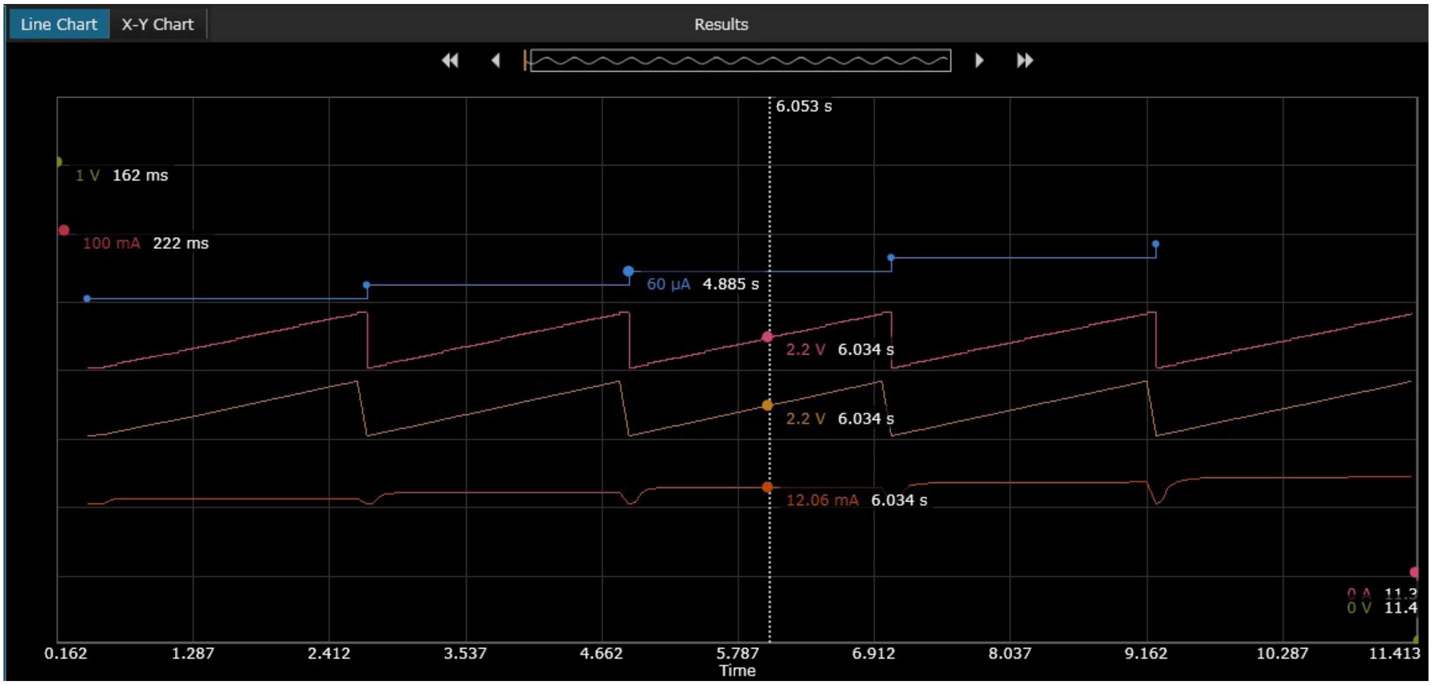


Figure 20. Line chart showing values of data and setup information at each execution step.

Select X-Y Axes for Custom Views

An X-Y chart, in which X and Y axes are independently chosen, can be generated by Test Flow. Multiple overlays with different sets of data can also be plotted if desired. Figure 21 shows an example of a transistor curve tracer chart that was made by selecting X and Y axis values (in this case collector current and collector voltage) and a value (base current) that selects which group of X-Y data is to be plotted.

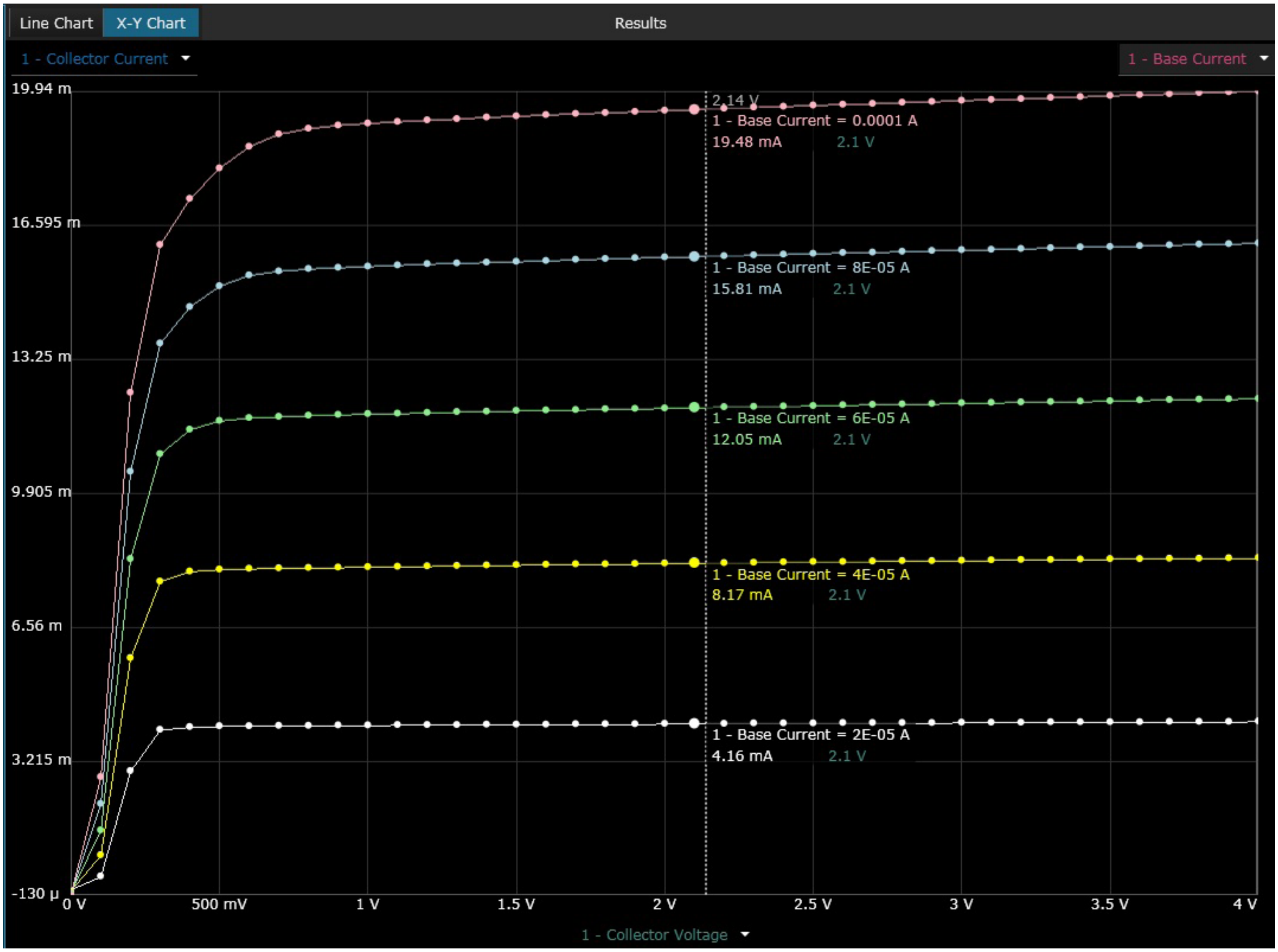
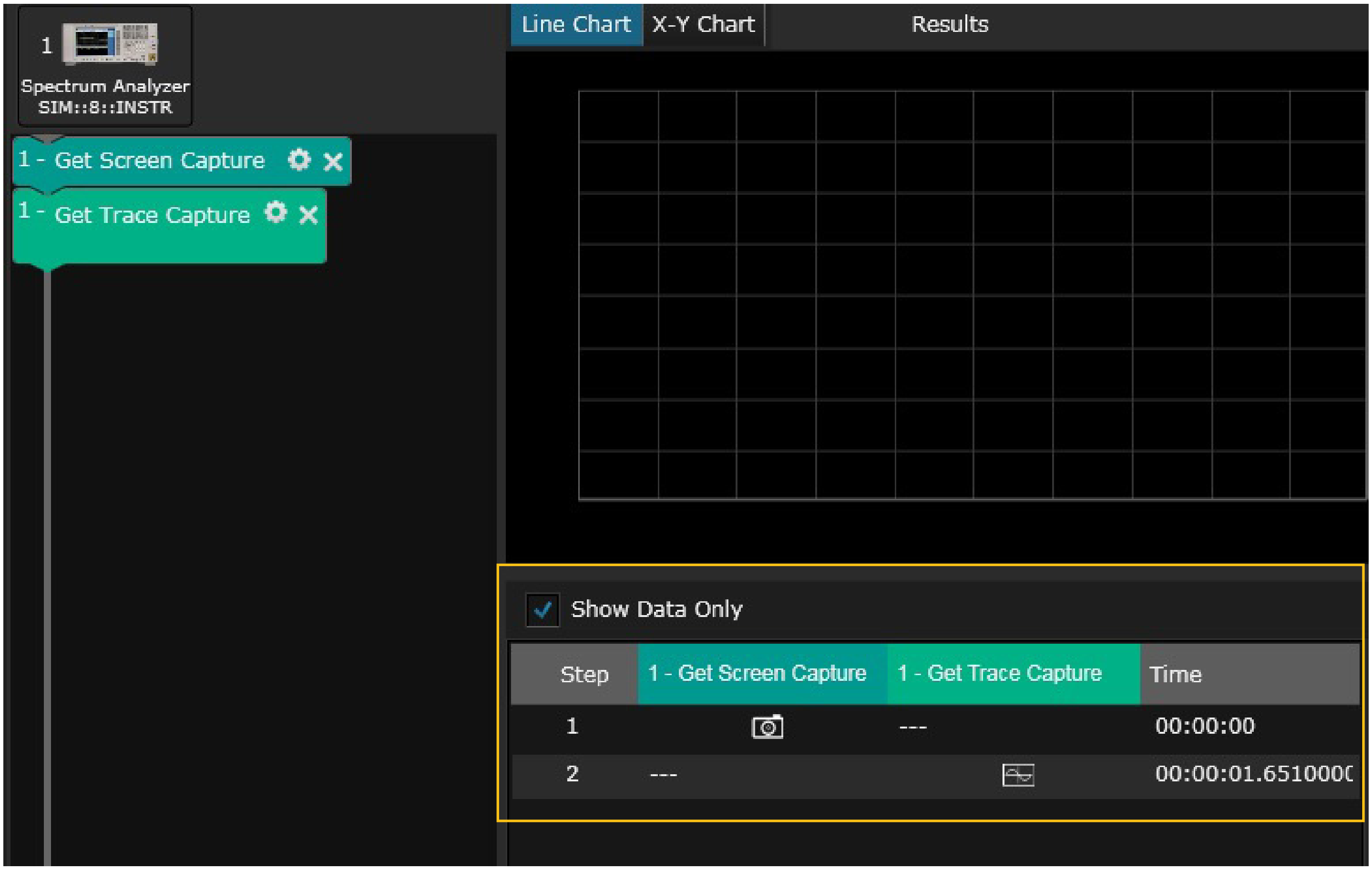


Figure 21. X-Y chart constructed from multiple measurements.

Screen and Trace Data Capture

Some BenchVue apps can capture images and trace data. If the feature is enabled for use in Test Flow (i.e., it has an orange rectangle around it), then it can be dragged into the Test Flow workspace. When the sequence runs, a Trace Data or Screen Capture icon is placed into the Results table. Clicking on the icons cause the images or trace data to pop up, where they can be exported (Figure 22). Or, you can use the Export block to export this data as part of the automated sequence.



☑ Show Data Only

Step	1 - Get Screen Capture	1 - Get Trace Capture	Time
1	📷	---	00:00:00
2	---	∿	00:00:01.651000

Figure 22. Screen and trace captures can be automated into Test Flow sequences.

Conclusion

As this application note has shown, there are many things you can do with BenchVue's Test Flow application to save time and achieve faster results—all without ever having to write a computer program. The intuitive drag-and-drop interface in Test Flow enhances your work flow with the ability to create custom, automated test sequences; a task that used to require days using traditional programming languages. Moreover, you can obtain a deeper understanding of your data using Test Flow's built-in analysis tools, which log and graph with selectable x-axis and y-axis formats and much more.

For more Test Flow application details and examples, go to **www.keysight.com/find/BenchVueTestFlow**.

For more information about Keysight's BenchVue software, refer to **www.keysight.com/find/BenchVue**.

Download your next insight

Keysight software is downloadable expertise. From first simulation through first customer shipment, we deliver the tools your team needs to accelerate from data to information to actionable insight.

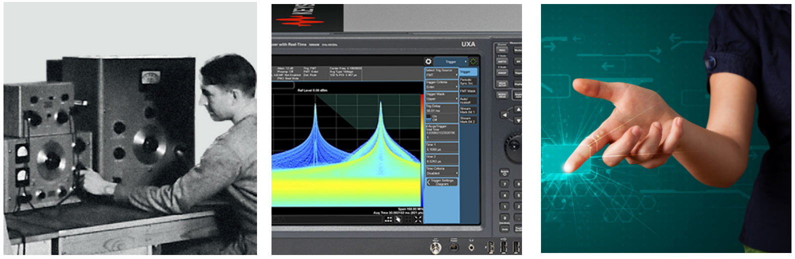
- Electronic design automation (EDA) software
- Application software
- Programming environments
- Productivity software

Learn more at
www.keysight.com/find/software

Start with a 30-day free trial.
www.keysight.com/find/free_trials

Evolving

Our unique combination of hardware, software, support, and people can help you reach your next breakthrough. We are unlocking the future of technology.



From Hewlett-Packard to Agilent to Keysight

myKeysight
www.keysight.com/find/mykeysight
A personalized view into the information most relevant to you.

www.keysight.com/find/BenchVueTest Flow

For more information on Keysight Technologies' products, applications or services, please contact your local Keysight office. The complete list is available at:
www.keysight.com/find/contactus

Americas	
Canada	(877) 894 4414
Brazil	55 11 3351 7010
Mexico	001 800 254 2440
United States	(800) 829 4444

Asia Pacific	
Australia	1 800 629 485
China	800 810 0189
Hong Kong	800 938 693
India	1 800 11 2626
Japan	0120 (421) 345
Korea	080 769 0800
Malaysia	1 800 888 848
Singapore	1 800 375 8100
Taiwan	0800 047 866
Other AP Countries	(65) 6375 8100

Europe & Middle East	
Austria	0800 001122
Belgium	0800 58580
Finland	0800 523252
France	0805 980333
Germany	0800 6270999
Ireland	1800 832700
Israel	1 809 343051
Italy	800 599100
Luxembourg	+32 800 58580
Netherlands	0800 0233200
Russia	8800 5009286
Spain	800 000154
Sweden	0200 882255
Switzerland	0800 805353
	Opt. 1 (DE)
	Opt. 2 (FR)
	Opt. 3 (IT)
United Kingdom	0800 0260637

For other unlisted countries:
www.keysight.com/find/contactus
(BP-06-08-16)

KEYSIGHT TECHNOLOGIES
Unlocking Measurement Insights

This information is subject to change without notice.

Published in USA, August 11, 2016
5992-1690EN
www.keysight.com