Software Programmers Manual

MK AT Editor

Version 1.700

Page Intentionally Left Blank

MK Test Systems Limited believes the information described in this manual to be accurate and reliable, however, no responsibility, financial or otherwise, is accepted for any consequence arising out of the use of the information contained within this manual.

The information contained in the manual is subject to change at any time without prior notice.

In no event shall MK Test Systems Limited be liable to anyone or body for special, collateral, incidental or consequential damages or losses with or arising out of purchase or use of the MK Test Systems Limited produced software.

MK Test Systems Limited reserves the right to revise and improve its products as it sees fit.

Page Intentionally Left Blank

MODIFICATIONS

Issue	Date	Modified by	Modified Pages	Observations
1.000	05/May/2016	APK	All	First revision
1.200	14/Oct/2016	JP		Separate documents for runner and editor
1.300	06/July/2017	APK		Updated to include latest changes
1.400	07/August/2018	SE	All	Updated to incorporate latest changes to date.
1.500	09/Sep/2018	CDVM	All	Updates for version 10.9.0
1.600	29/01/2019	DW		Updates for version 10.10.1
1.700	04/03/2019	CDVM		Updates for version 10.11.0/1

Table of Contents

Table of Contents

1 Software System Requirements

Note: the following is the recommended specifications that should be suitable in most cases, but the ideal specification will depend on individual system characteristics and configuration.

Processor: i7

RAM: 16Gb min.

HHD: 160 GB min.

Operating system: Windows 10 (64 bit)

Note: MK recommend using the latest service packs and updates provided by Microsoft. Due to the pace at which new software technologies become available, support for such technologies is provided to the end user by means of a service pack. MK endeavour to utilise technologies available that will improve performance and stability in our software. When upgrading or installing the latest version of software, ensure your operating system has been recently updated to mitigate any compatibility issues that may arise as a result of an underlying change.

2 General Information

2.1 Before You Begin

2.1.1 Familiarisation

Before using the MK AT test system, it is strongly advised that you read all of the manuals provided and ensure that you are familiar with the concepts in operating this form of automatic test equipment

2.2 About This Manual

This manual is designed in such a way that a first-time user of the MK AT system will, by following this manual, be taken through the MK AT software in a logical manner. Those users more familiar with the MK AT may wish only to use this manual as a reference.

2.3 MK AT Operating Concepts

The MK AT software operates on the following concepts:

a) The user manufactures a piece of equipment (UUT), which is tested by conventional means and known to be of the required quality standard.

b) The MK AT 'learns' about the UUT characteristics by being programmed by the user. The MK AT can automatically learn the UUT connections. They can be programmed manually or they can be imported from an external source.

c) New build items of the same design can now be tested by the MK AT by being compared against the information acquired in B above. The operator simply selects the item with the same part number (or some equivalent parameter) from the index.

The test referred to in c) above consists of one or more of the following stages.

1) Continuity comparison on the connections of the UUT on the index against the UUT being tested. *Referred to as the positive test or continuity test*

2) A check through the UUT to ensure that no connections exist, which were not present in the UUT on the index. *Referred to as the negative test or shorts test.*

3) Assuming the successful completion of 1) and 2) above, a high voltage insulation test can be performed. This test takes each point in turn to a specified high voltage whilst all other points are connected together. The resistance is then measured and compared against a threshold.

As an alternative to or in addition to option 3) above, a high voltage hi-pot test can be performed if required, which subjects the UUT to a specified high voltage and compares the leakage current against a threshold.

For twisted pair configurations, a capacitance test can be performed which will compare the capacitance against threshold parameters.

External equipment can also be connected and switched through the MK AT switching matrix.

If an insulation test and hi-pot test are selected, the hi-pot test will only commence if the insulation test is successful without any point failures.

In the event of the hi-pot test being performed before the insulation test, then the insulation test will only commence if the hi-pot test is successful.

All of the test sections described are supported by screen, print to file and printer reports, if required.

2.4 MK AT Electrical Concepts

Even if the operator is familiar with automatic test equipment, the following basic concepts will help to understand the MK AT operation.

The MK AT measures resistances or volt drops by applying a fixed current and using a voltmeter to measure the dropped voltage. If a resistance value is required, then a simple ohm's law calculation is performed. For higher resistances then the internal impedance of the MK AT is also taken into consideration.

For a four-wire measurement, the resultant value will be a close accurate value of the UUT. For a two-wire measurement, the resultant value is likely to contain an element due to the connecting interface.

2.4.1 What Test Current Should I Use?

It is always best to use the maximum test current available, this will give the greatest volt drop which can be measured with the highest accuracy. It will also stress the UUT more which may show a potential problem.

Never select a test current which will damage the UUT or a current which result in too much power dissipation in a resistive load.

2.4.2 What is Voltage Limit?

The test current is supplied from a power supply which produces approximately 42 volts on most MK AT systems. If the current is supplied into an open circuit i.e. no current flows, then 42 volts will appear across the UUT

The voltage limit is a programmable limit to this open circuit voltage and is typically set to 30 volts.

If the UUT has a maximum voltage requirement, then the voltage limit can be lowered accordingly.

2.4.3 Auto Learn and Shorts Test Methodology

MK AT will apply the current from point one to all other points. If the voltage measured is equal to voltage limit i.e. open circuit, then MK AT will move on to point two etc. If the voltage measured is less than clamp i.e. a connection, then a binary split will be performed until the end points are established.

The current will be applied, and voltage read to establish the connection value.

Auto learn and shorts test will only learn and test values within the electrical constraints of the programmed current and voltage limit. Values outside these parameters will be missed by the auto learn or show up as open circuits by the shorts test. To cover the maximum range for auto learn and shorts test, select the minimum current and maximum voltage limit.

2.4.4 Low Voltage Testing Methodology (Continuity Test)

MK AT will scan the list of expected connections, apply the current set up in the continuity test parameters and measure the resultant voltage. The voltage or resistance will then be compared to

the limits, and a result produced. If the voltage limit is reached when the current is applied, then a > maximum resistance will be reported.

2.4.5 Low voltage short circuit

MK AT will take each net list (a set of points that are common) and test them using low voltage to all other net lists defined within the shorts sub test. If a short circuit is found (i.e. low resistance so voltage limit is not reached), then the MK AT will use a binary chop to determine which other net list it is shorted to. It will then remove that net list from the list to be tested against, and repeat. This will allow the system to find any number of shorted net lists. If selected by the programmer, the system can also determine the resistance path between each point on both the shorted nets and show the lowest 10 resistance paths on the report. This is so that a user can quickly identify where the short is between the nets. Note this is limited to 1000 measurements to prevent too much data being generated. If two very large nets are shorted together, e.g. netlist1 contains test points 1,2,3,4 and netlist2 contain test points 10,11 we measure the resistance between 1->10, 1->11,2->10, 2->11, 3->10, 3->11, 4->10, 4->11.

2.4.6 High Voltage Testing Methodology (Insulation and Hi Pot)

MK AT will then take each net list, as defined in the insulation sub test, and test against each additional net list defined within that sub test. The sequence of how this is done is detailed below.

An insulation resistance or leakage current calculation is then performed, and a comparison made with the values previously defined to produce a result.

It is possible that the high voltage test will report values not detected by the low voltage test because of the constraints of the low voltage test detailed above.

2.4.6.1 High Voltage DC and AC Ramp Test Algorithm

This is the normal algorithm applied during high voltage testing of AC and the majority of DC systems and consists of the following stages:

2.4.6.1.1 Relays Switched On

The appropriate test points (from and to) are switched onto the bus.

2.4.6.1.2 Ramp Up

This is a programmable ramp in which the high voltage is increased in a controlled manner until the high voltage potential is achieved. If not achieved, then the software acts accordingly and reports as such.

2.4.6.1.3 Pre-Dwell

This is a programmable dwell where the high voltage remains present to allow for any capacitive effects.

2.4.6.1.4 Measure Dwell

This is a programmable period over which numerous readings of leakage current are made to establish the insulation resistance or HiPot value.

2.4.6.1.5 Ramp Down

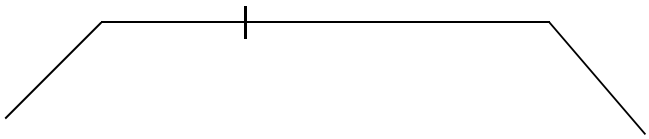
This is a programmable ramp in which the high voltage is decreased in a controlled manner until the high voltage potential is removed.

2.4.6.1.6 Safety Dump

This is a programmable time for which the safety dump is activated to discharge any residual potential. The high voltage is then checked to ensure that it has been discharged.

2.4.6.1.7 Relays Switched Off

The appropriate test point sources and monitors (signal and sense) are switched off the bus. The complete cycle then repeats.



Ramp Up Pre dwell Measure Dwell Ramp Down

If required, it is possible to select the option to “find which nets failure is shorted to”. This will then use the same binary splitting algorithm as short circuit detection to determine which of the “to” nets the HV has failed to. Note that in some circumstances it may not be possible to determine the shorted net, as there could be more than one resistance path.

Note on some large systems, if a short / HV fail has been detected, then the system may need a longer pre-dwell on the next measurement to ensure correct operation. The amount of time can be adjusted by the “After HV Failure, adjust pre-dwell time by a factor of”. This parameter will multiple the pre-dwell of the next parameter by this value.

3 Software User Interface

MK AT software has been written with the primary intention of being simple to use, the aim being that familiarisation is quick and consistent in its use. As such the software is split into two major parts: the MK AT Editor and the MK AT Runner. The ***Editor*** is a desktop application that allows an operator to create test programs not only for MK AT, but any MK test products. The ***Runner*** is designed to facilitate the shop floor operation of connecting to and testing of the UUT. Refer to the MKAT Runner Software Manual for information about running a test and interacting with the MK AT hardware.

3.1 MK AT Editor

Please note that this document only covers program creation and editing within the MK AT Editor software. For details of configuration creation and editing, please refer to the "MKAT Runner Software Manual.docx" document. The only exception to this is the *System Mask* feature which can be configured in the Editor but is not currently available in the Runner. This feature is detailed at the end of this document.

On start-up you can select from the front screen which items you wish to edit / create. On the RHS will be a list of items that you have recently edited. The MK Editor is not only used to create / edit test programs for MK's range of products, but also MK AT hardware configurations. In order to be able to create a MK AT program, you will also need the configuration of the system that you intend to run the program on. This is so that the editor can ensure test points are available, and also ensure that you have access to the correct externally programmable devices. If you create a program with the incorrect configuration loaded, then the program may not load or run correctly in the MK AT runner.

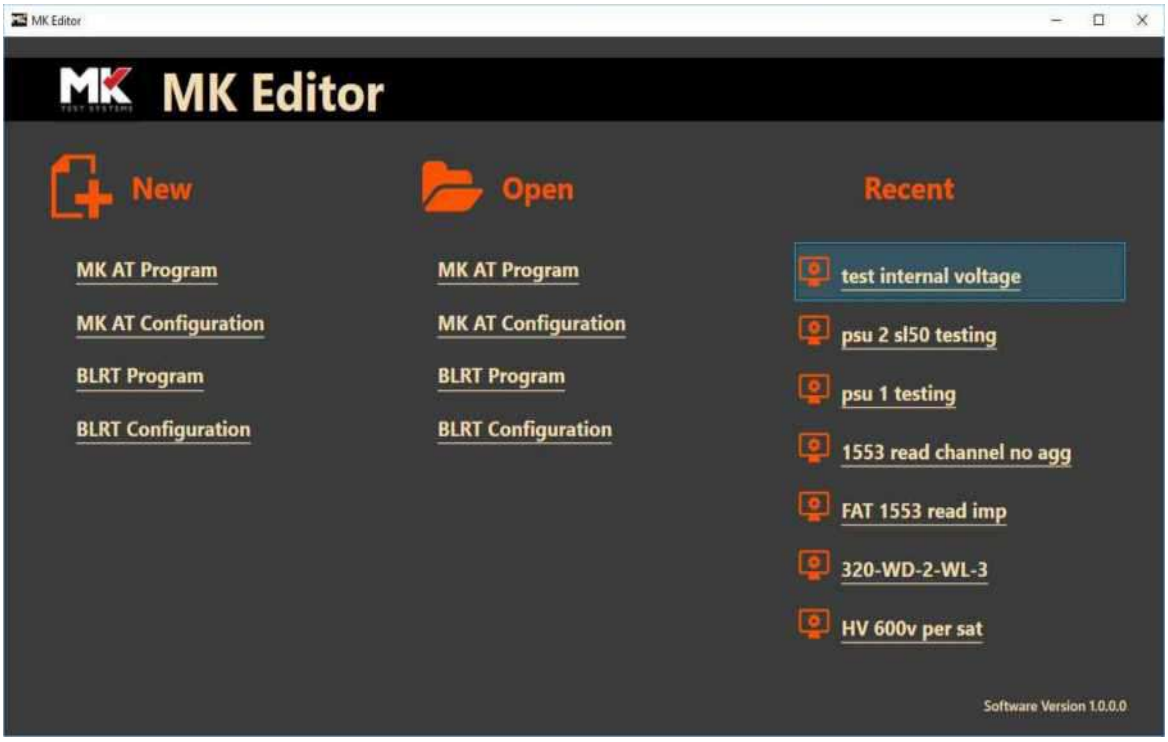


Figure 1: MK AT main start up screen.

3.1.1 Creating a program.

When you select the ***MK AT Program*** from the front screen, the Editor opens on the ***Sub-Tests*** tab by default. Selecting the ***Options*** tab opens the following screen:

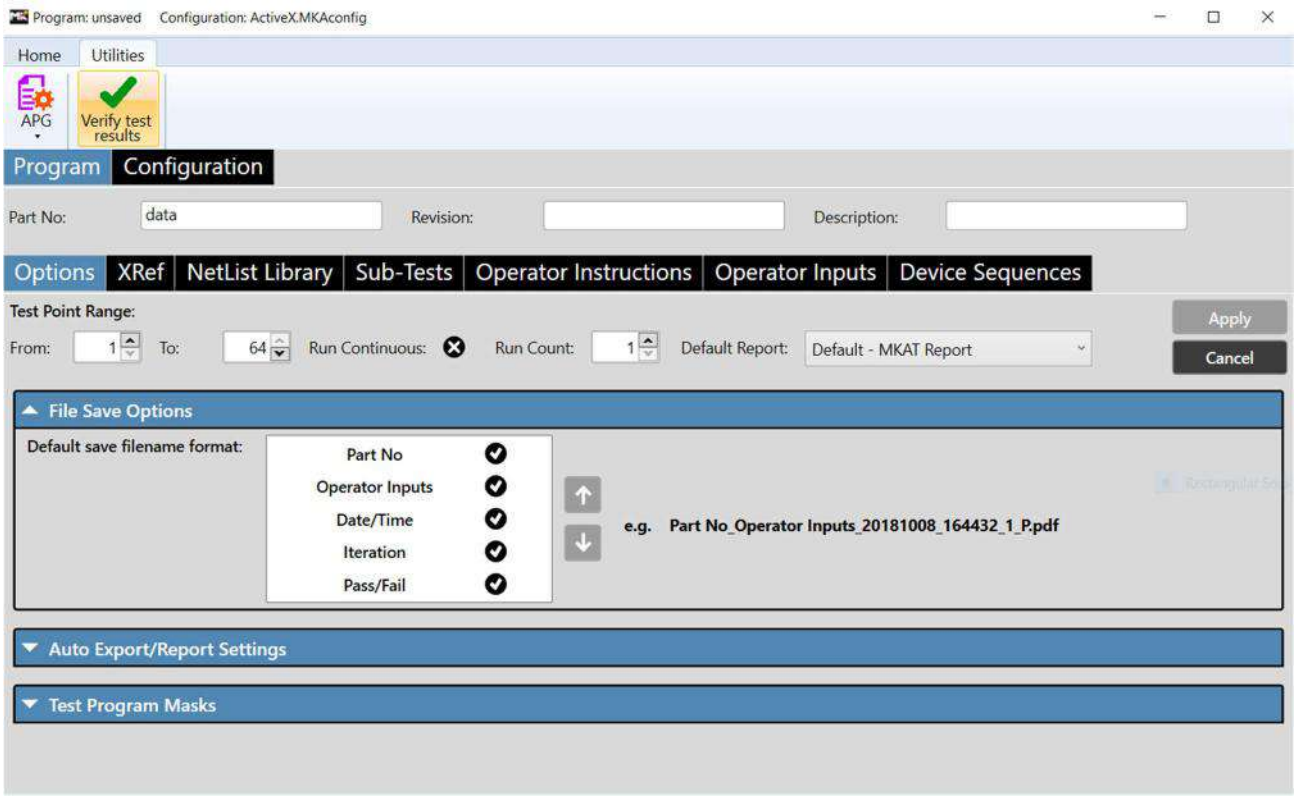


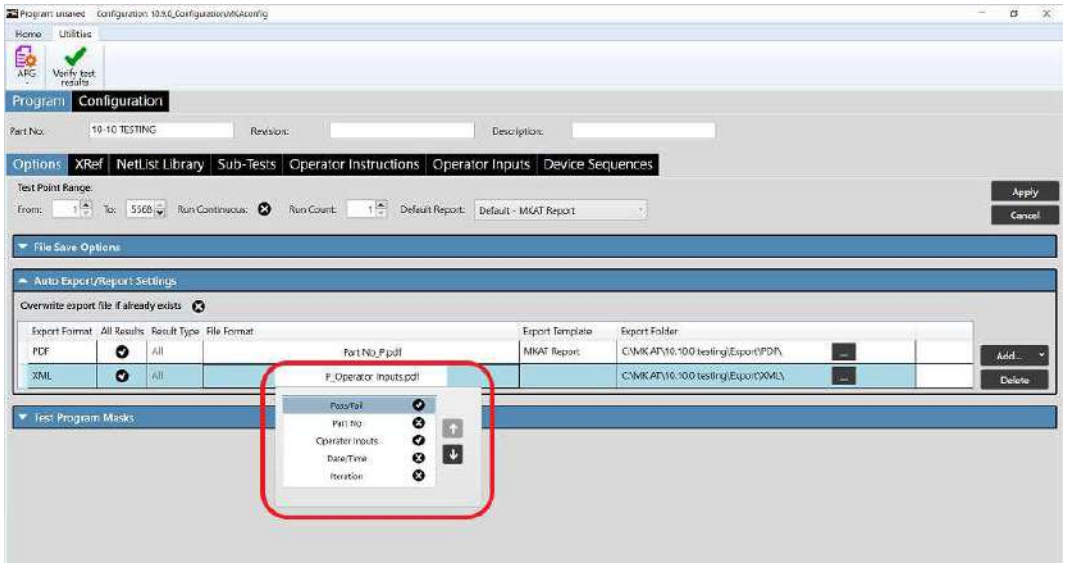
Figure 2: Main MK AT program creation screen.

On this screen, the default “Options” tab is selected. Here you can enter details about the UUT and also define the test point range which needs to be used for this program. Test points outside of this range will not be considered in use so will be excluded for shorts and HV testing. If there are points that do not need to be tested, then these a can be added to the test program masks section i.e. these are points that are within the test point range but still do not need to be included in any tests. Import and Export of masks to “XML” or “TSV” formats can also be carried out.

File Save Options allows for configuration of the filename used when exporting files following a test. The filename can be constructed from the components shown and reordered using the up and down arrows.

The Auto Export/Report Settings section allows for the selection of defined reports which are automatically exported at the end of the test run. Selecting the “Add” button allows an export in any combination of either “PDF”, “Print” and “XML”.

As of version 10.10.1 the filenames can be configured individually for each auto export file.



3.1.1.1 Menu options

This allows you to abandon the current program and or configuration and start with a new one. Note: please ensure you save the current one if you want to save your changes.

This will allow you to open an already existing program or configuration. Note: you can have only one of each type open at the time, so please ensure you saved your work before opening a new one.

This allows you to view the program as a report. The reports can be saved as PDF and can be printed.

This allows you to view and analyse exported XML result data from the runner. If you wish to “re-generate” a report and / or just view them on screen, then select this option. This will show the screen below.

Figure 3: Viewing / analysing report data.

From this screen you can use the “Import Test Data” option to select one or more XML results file. Selecting any individual report on the LHS will display that result data. Selecting the ‘Analyse Test’ will produce a “trend” analysis report as shown below:

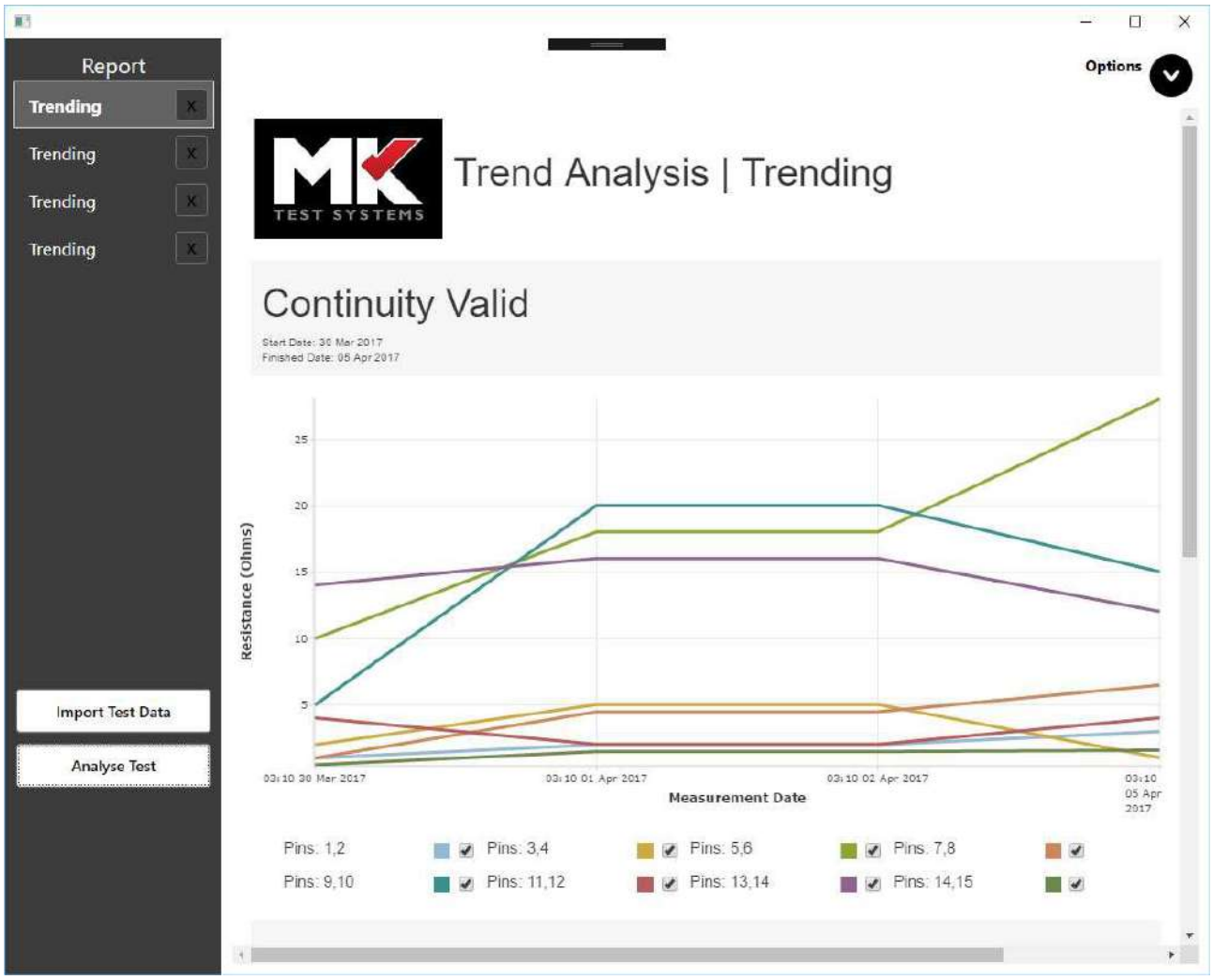


Figure 4: Trend analysis report.

Note your subtest parameters have to be the same in each subtest (unedited) for the matching / trend analysis. Each parameter in a subtest is given a unique ID when it is generated, and this is used to match the parameters together.

3.1.1.1.1 Editing Reports

The Edit Report button allows the user to customise the layout of the report for viewing. Pressing the button enables four main options: “Import Layout”, “Export Layout”, “Save Changes” and “Cancel”. Two options also appear: “Show each net on a new line” and “Max. rows to show for nets:”. The “Edit Report” button caption also changes to “Reset Report”. See the following screenshot:

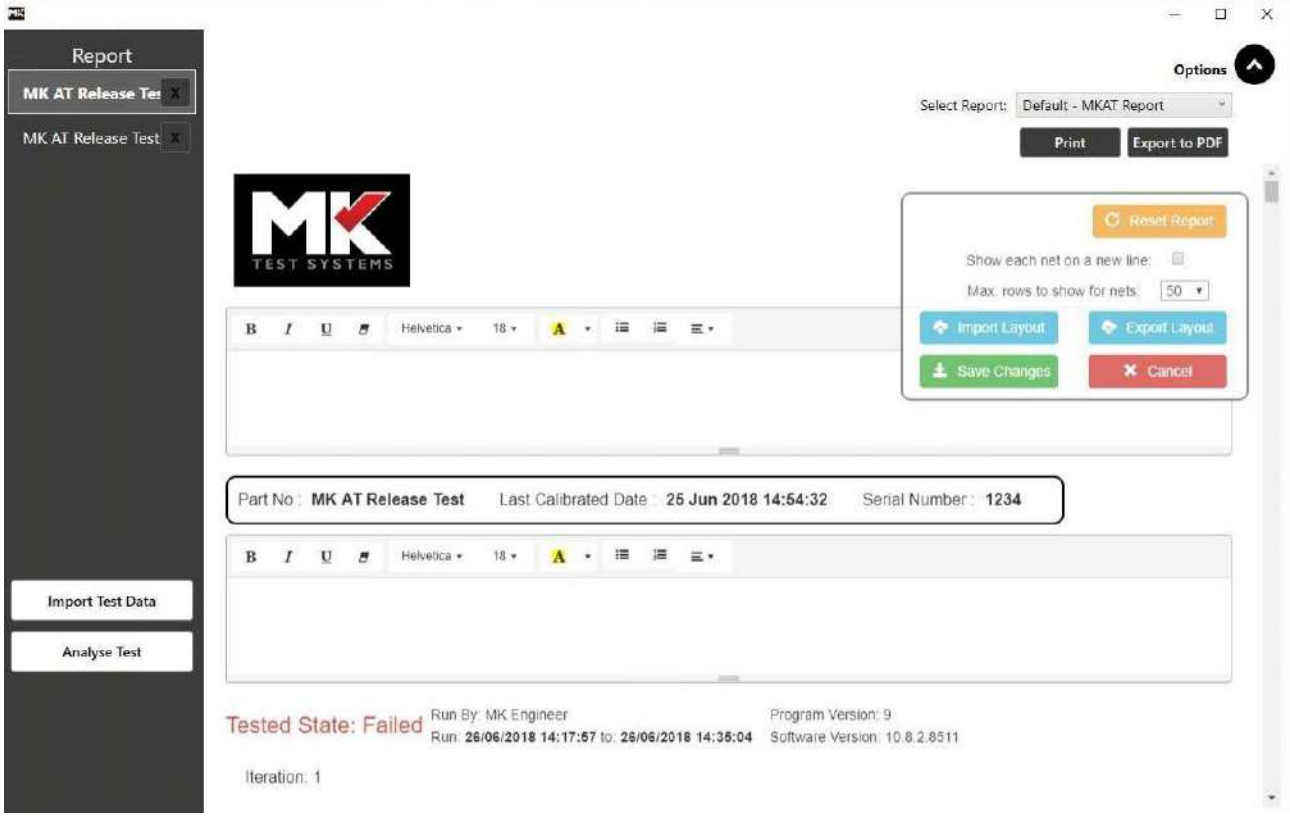


Figure 5: Edit report.

The import and export layout functions allow saving to and importing from a “.json” file. Importing will see the current view updated to the view as defined in the ”.json” file.

Selecting the “Show each….” option will force any multiple test points in a net to be shown individually on the next line, rather than appended on one line.

If the “Max. rows..” option is selected, the report will only show the number of lines as chosen in the drop down.

Table columns can also be edited. The green arrows allow for the column positions to be reordered and the orange buttons allow the columns to be hidden.

1. Continuity - Continuity - 2 Wire. 14 total, 2 passed, 12 failed.

Continuity - 2 Wire

Failure Action: Continue

Tested State: Failed

Pre Sub-Test Operator Inputs

Question	Response
Please Enter Your Name	Chris

Pin From	Pin To	Name	Minimum Value	Maximum Value	Measured Value	Units	Tested State	Measurement Type	Voltage Limit (V)	Current (mA)
-1-(1)	--(15)		0.50	2.00	0.94	Ω	Passed	In range	30	200
-2-(2)	--(16)		1.00	3.00	1.00	Ω	Passed	In range	30	200
-3-(3)	--(17)		0.50	2.00	N/A (check for open circuit)	Ω	Failed	In range	30	1
--(5)	--(29)		0.50	2.00	N/A (check for open circuit)	Ω	Failed	In range	30	1

The layout changes can be saved to the report template using the Save Changes button. The changes will then be applied automatically whenever this template is used. Formatting can also be reset to the original template layout using the Reset Report button. There is also provision for the addition of a custom header, sub-header and a footer.

Save the currently loaded program or configuration.

Save the currently loaded program or configuration to a new file.

Close the currently loaded program or configuration.

3.1.1.2 Creating XRef for the program.

Test programs in MK AT are all related to MK test point (TP) numbers. These however will not necessarily be meaningful to end users. You can therefore provide a cross reference (XRef) between the TP's and your

connectors. Because all TP numbers are consecutive in both two wire and four wire mode you will have to enter both 2 wire and 4 wire XRefs.

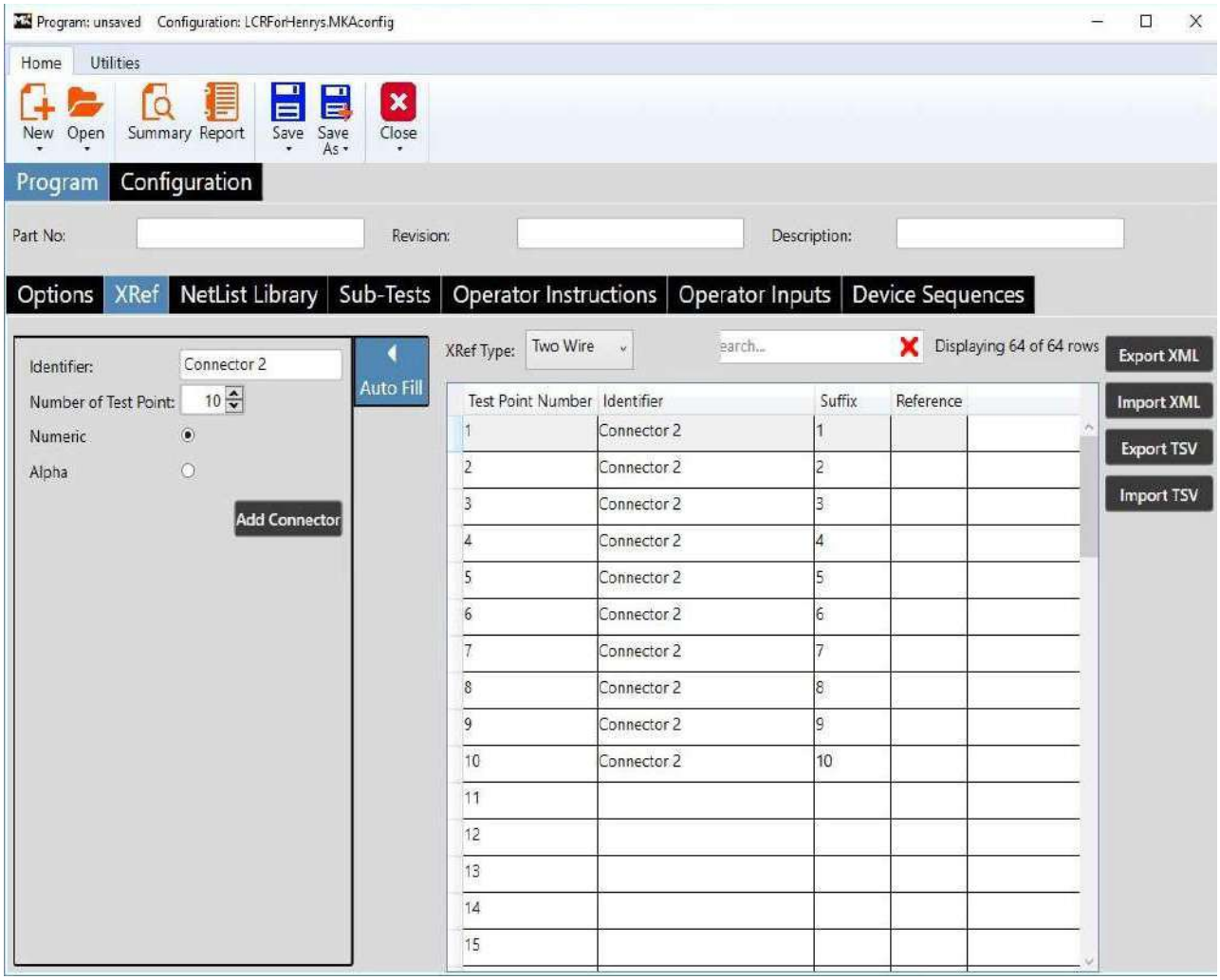
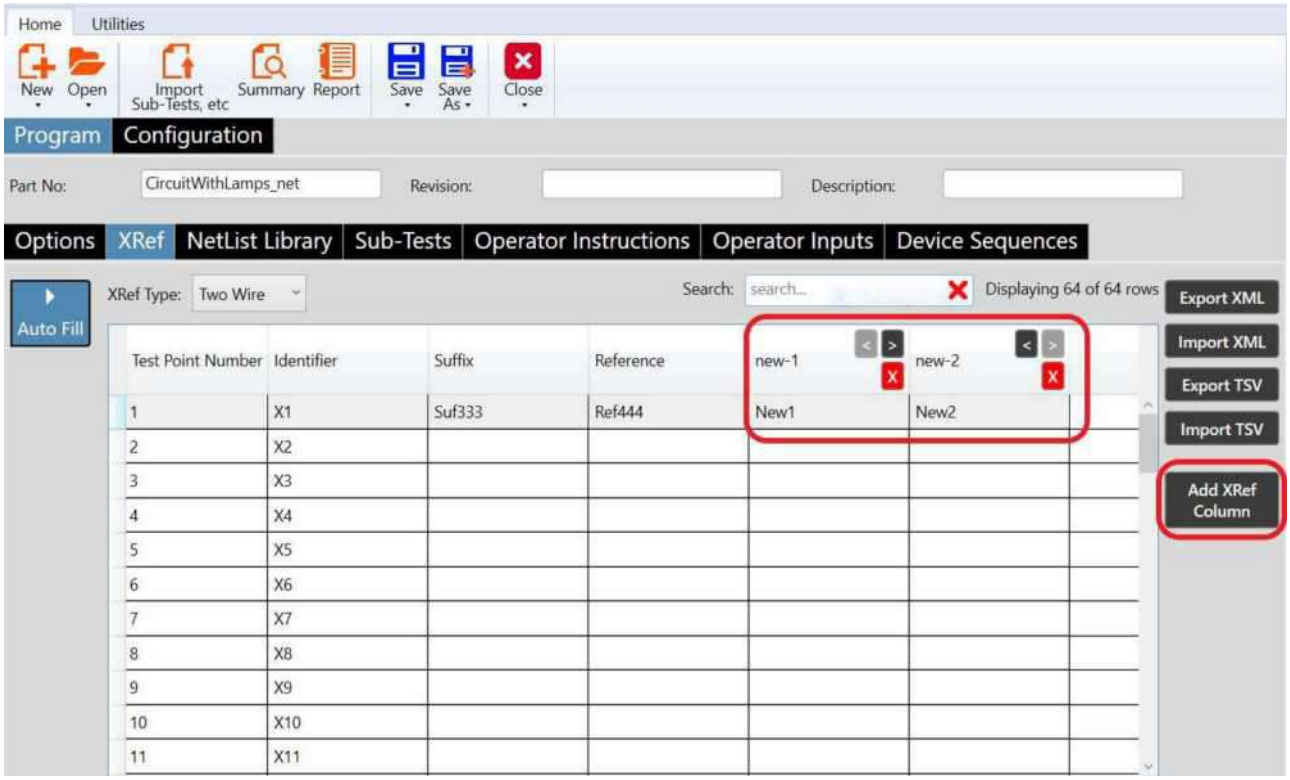


Figure 6: Entering XRefs

The operator may manually enter xref data line by line, however this is likely to be time consuming should there be a large number of test points. To assist in part automating the process, the operator can select the "Auto Fill" button, which will reveal the "Add Connector" options as shown. There are also the options to import and export xref's in XML or TSV format to further assist in speeding up the process.

Additional XRef Fields can be added if required using the Add XRef Column button, shown below. These fields can be reordered using the *arrow* buttons at the top of each column or deleted using the *x* button. This

additional XRef data is displayed everywhere that the XRef information is normally displayed and can be formatted in the *Parameters* section of *Configuration*.



3.1.1.3 Adding netlist to the library

If you require more control on how HV tests are carried out, you can create your own netlist libraries. Using this feature will allow you to define not only the netlists but also the ability to "group" netlists together. This will then allow you to perform "group" to "group" HV testing rather than just net to net testing.

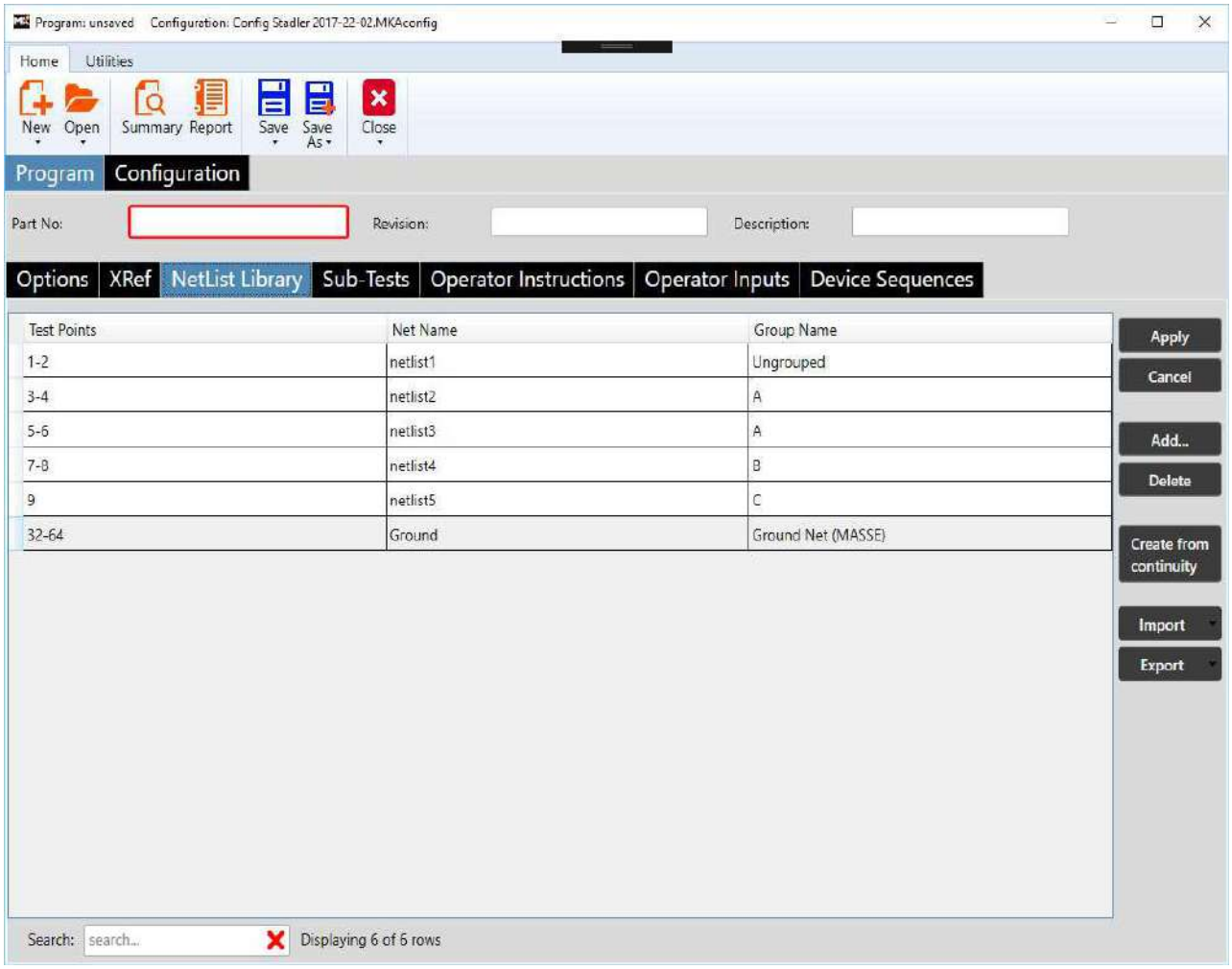


Figure 7: Simple example of netlists with groups.

3.1.1.4 Adding /Editing Sub-Tests.

The next tab is the sub-test tab. This is where you can create and define the running order of subtests to be run against the UUT.

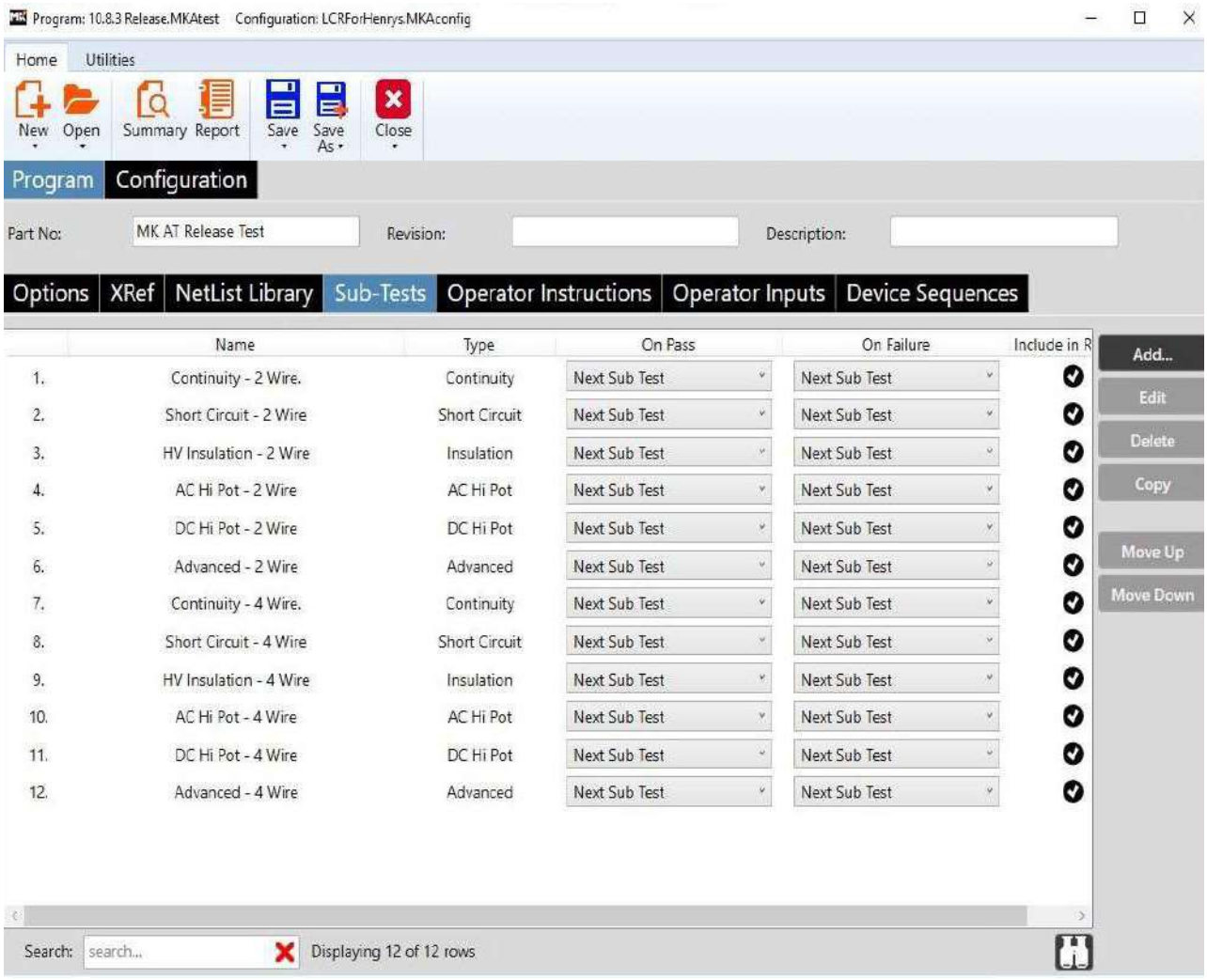


Figure 8: List of sub-tests created for this program, in the order they will be run.

In the sub test selection screen, you change add, edit and delete a sub-test. You can also change the order in which they run (the order is from the top of the screen down in sequence). If you wish to perform a “branch” then you select which sub test you wish to execute next in the “On Pass” / “On Fail” columns. The software will jump to that position when the condition is met and continue from there. You can also copy a sub test, which will be added as the last sub test.

For programs that contain a large amount of sub tests, there is a search / find facility provided by clicking on the binocular icon at the foot of the screen. This will load up a small window in which you can enter text. If you then select Find Previous or Find Next, the software will search for this text within the “Name” column and jump to that sub test if it finds a match.

3.1.1.4.1 Continuity

To add a continuity test you add a new sub test, enter the sub test name (or leave the default generated name). On the next screen you can select the defaults for each continuity measurement. These are the values that will be used when a new measurement is added to the next screen. These can be overridden for any item, so it is best to set these to the more commonly used values. You can then quickly add connections, just changing the ones that the defaults are not applicable for.

3.1.1.4.2 Shorts

Adding a shorts test is very similar to adding a continuity test. The steps that you follow are identical, but the data required for the defaults is different. Also, when creating a test, you can generate the net list (the list of common connections) direct from a continuity sub-test e.g. If you have a continuity test containing the following connections:

From	To
1	2
1	3
1	4
5	6
5	7
8	9

If you select this continuity subtest as the source of data and ensure that the test point range used covers a larger range than the test points used in the continuity subtest for example, we will use the test point range 1 to 10. The shorts test generated will contain the net lists *1-4 5-7 8,9* and *10*. So you will end up with 4 nets to be tested to each other, note that for any test points that are in the test point range that are not part of a continuity measurement will be added as single point nets, this is so you can prove that there are no short circuits to pins that are not connected.

3.1.1.4.3 HV (Insulation and AC)

These tests are created in the same way as the short circuit. The only difference is in that HV testing is done in both directions i.e. all nets are tested to all other nets in the list, in both directions. If there is a net that you only wish to test to and not from, then you set the voltage for that net to 0, as shown below.

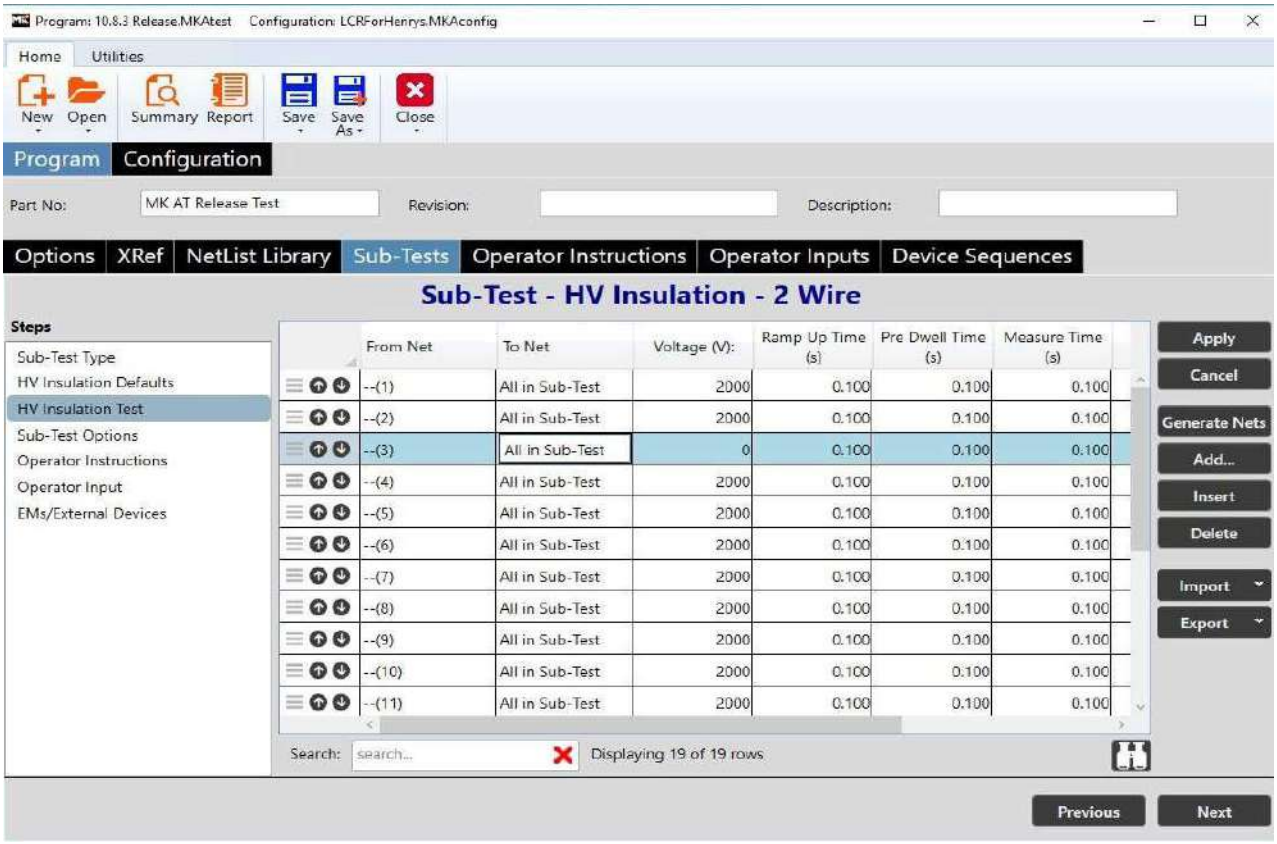
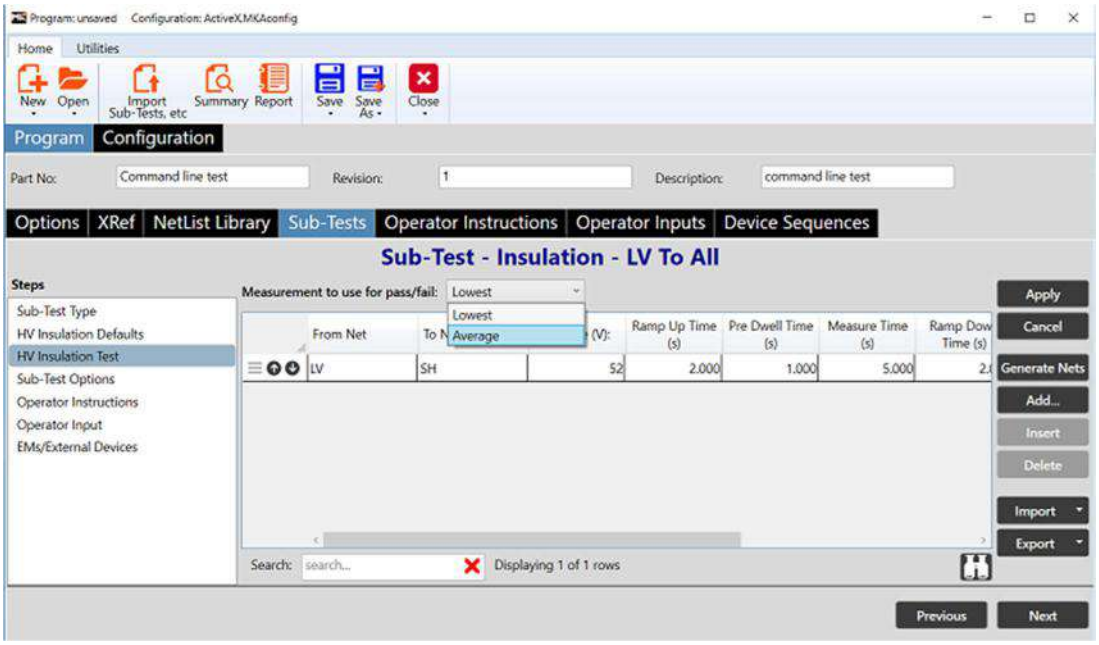


Figure 9: HV test with masked point.

HV Insulation tests default to using the lowest measurement to determine a pass or fail. This can be changed so that the average measurement is used instead, see the example below.



If you have used the netlist library feature, you can select the netlists in the from and to nets from the library. Below are some examples of how you could use them.

1: Simple nets to all in sub-test:

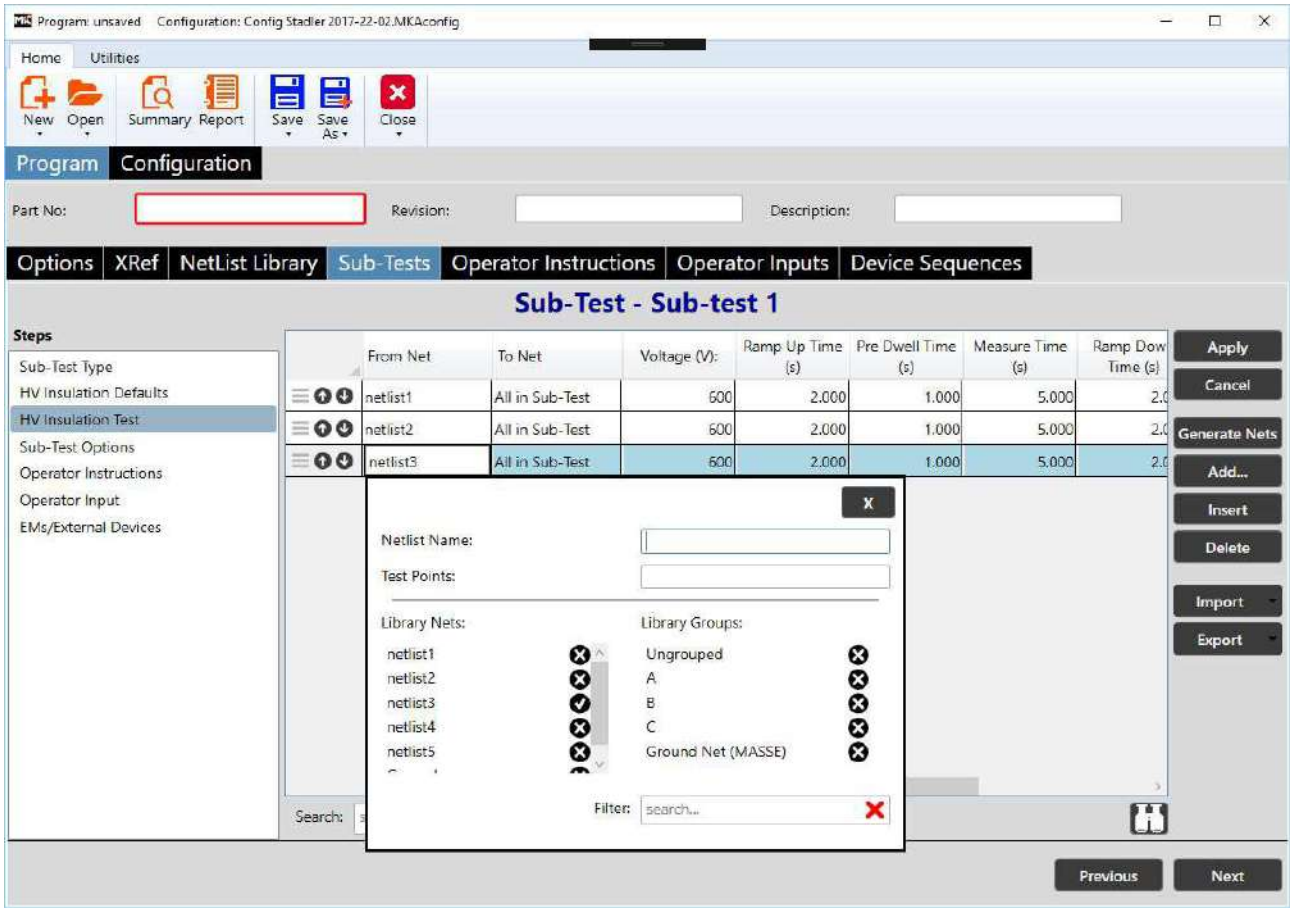


Figure 10: Select the from net you wish to test from.

2: Test net to net for all but the ground (Masse) net.

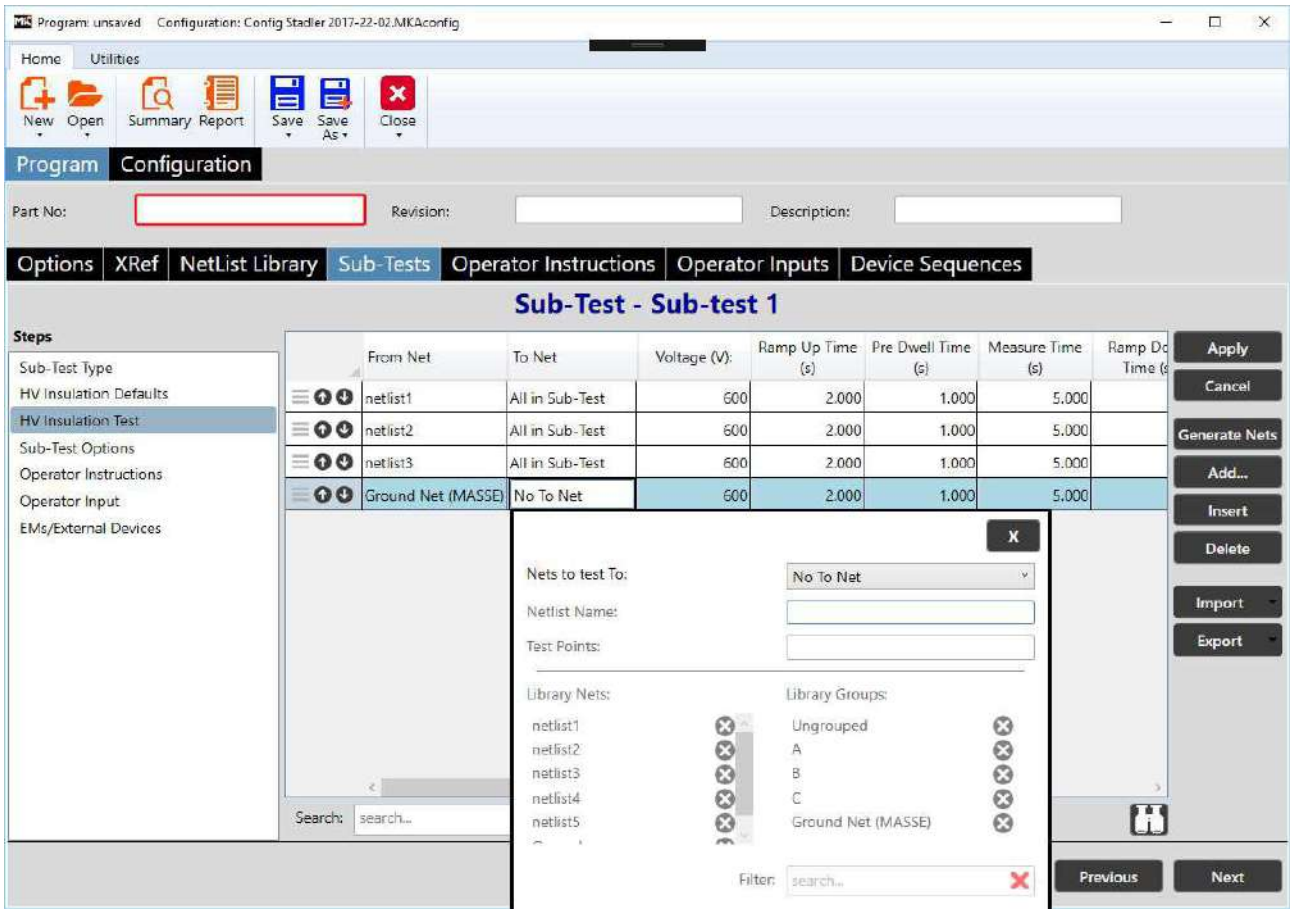


Figure 11: testing to a ground, but not from a ground net.

3: Group to group testing. The example below shows how you would use the netlists and the grouping from the library section to perform a simple group to group test.

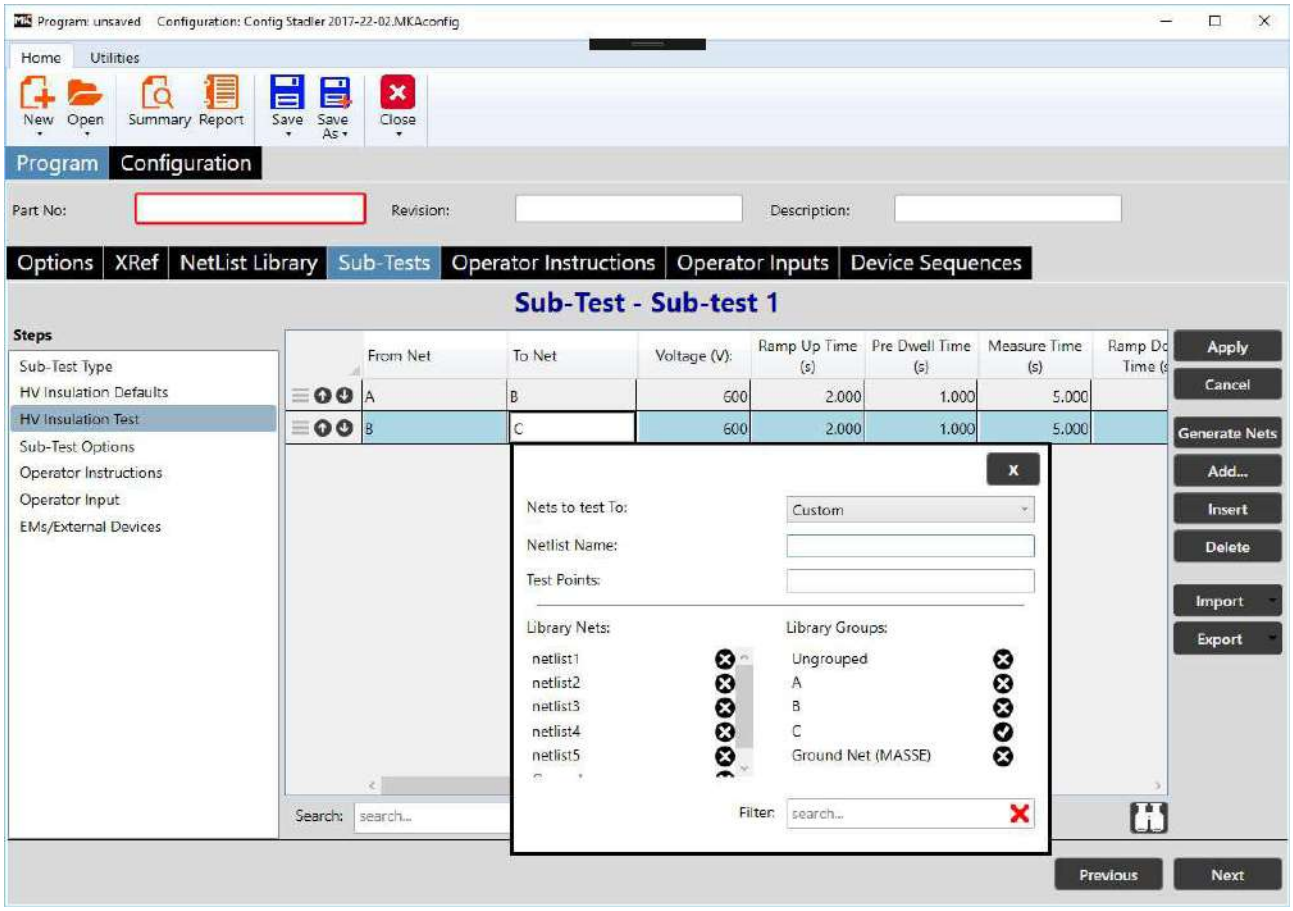


Figure 12: Group to group testing.

3.1.1.4.4 External device.

If your system has an external device fitted such as an LCR, external meter, or oscilloscope option, then you will need to use this sub test type to read the value and set the limits. Different external device that you are reading from may require input of different parameters and values.

3.1.1.4.5 Advanced

The advanced sub test is designed to allow you to carry out more complex measurements, and/or use more than one device to measure or compare data. This advanced test is used to create the Calibration Verification programs, as these need to display the internal meter value, and compare with an operator entered value. A pass / fail will then be calculated on a percentage difference between the values. This gives an idea of how this advanced sub test type can be used. Below is a set of screen shots showing the steps required to create an advanced sub test. This test type can also be used for volt drop testing, and time to reach value test (such as time to open / close). For the timed reading function, a pass is true if the value to reach is achieved after the minimum time, and is maintained until the maximum time, so if the value is reached before the minimum time it is a fail. If the value to reach is triggered after the minimum time but then drops out of the limit before the maximum time, this is also a fail.

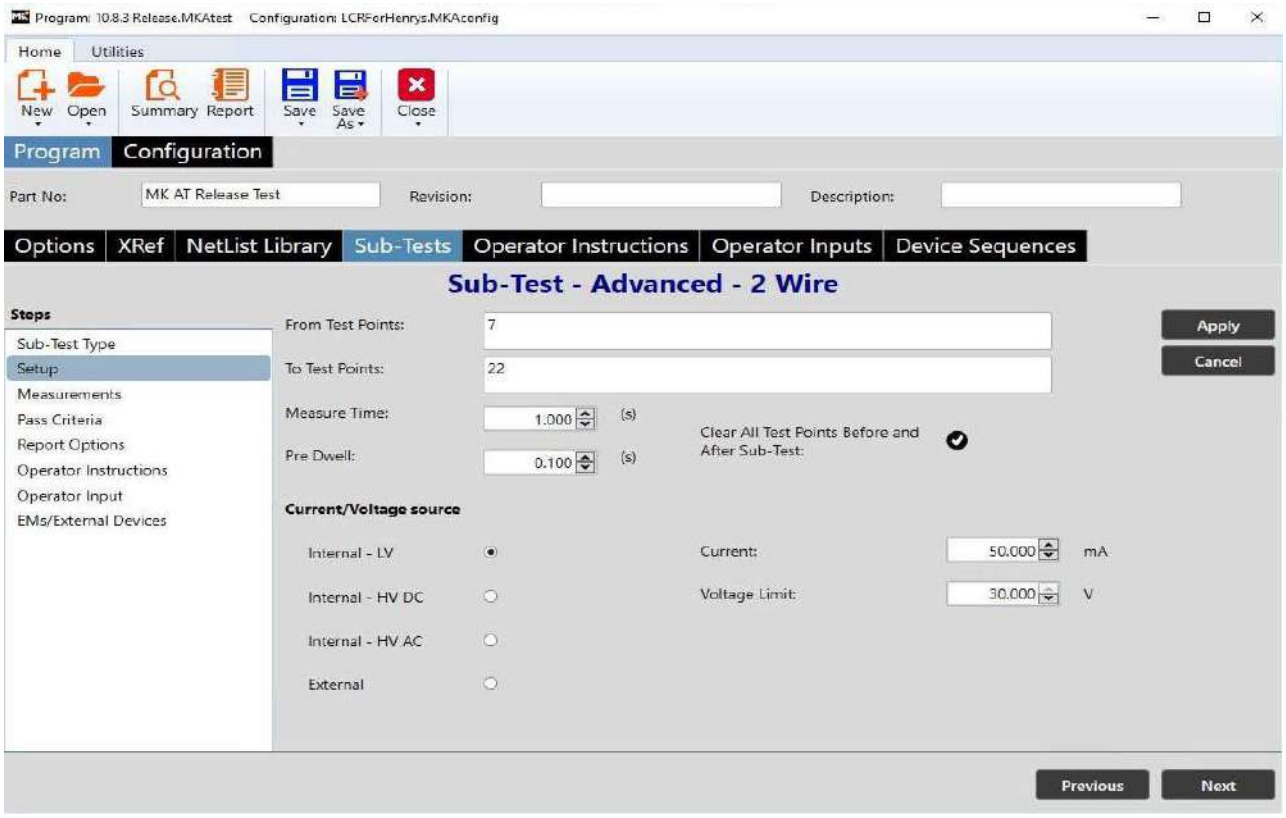


Figure 53: Advanced sub test setup screen.

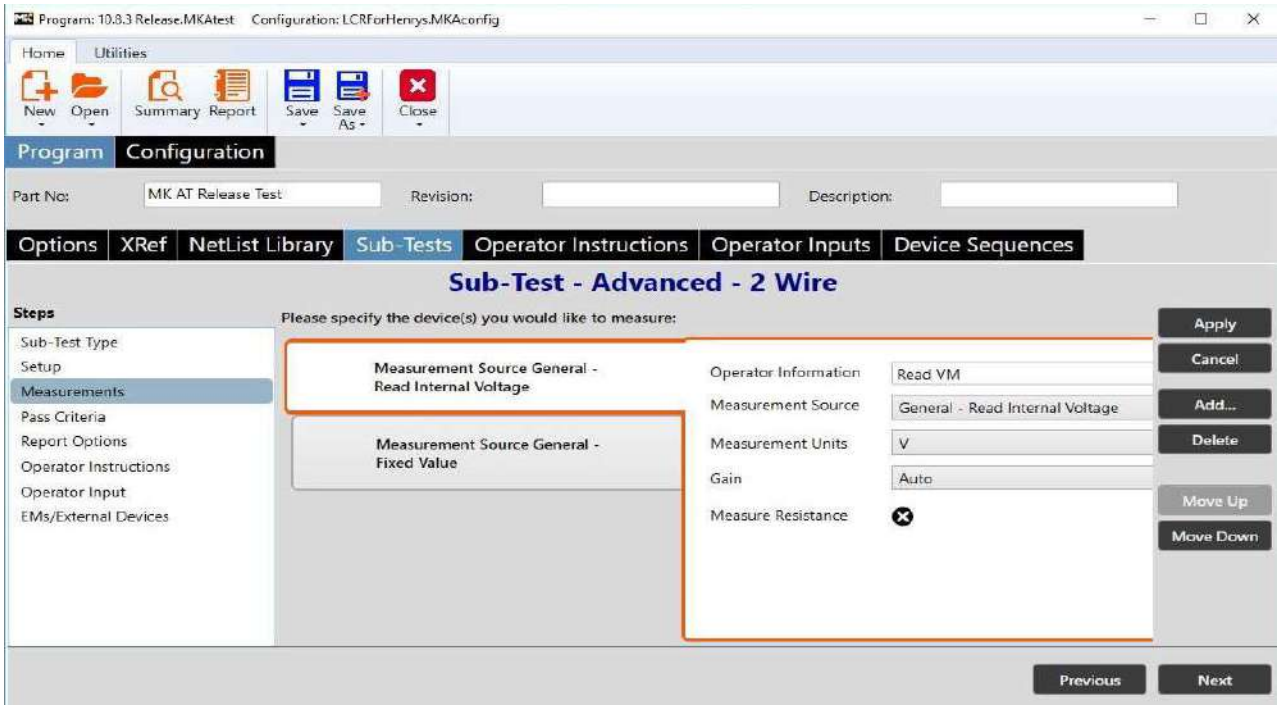


Figure 14: Setup of one or measurements to be taken, and or values to be used.

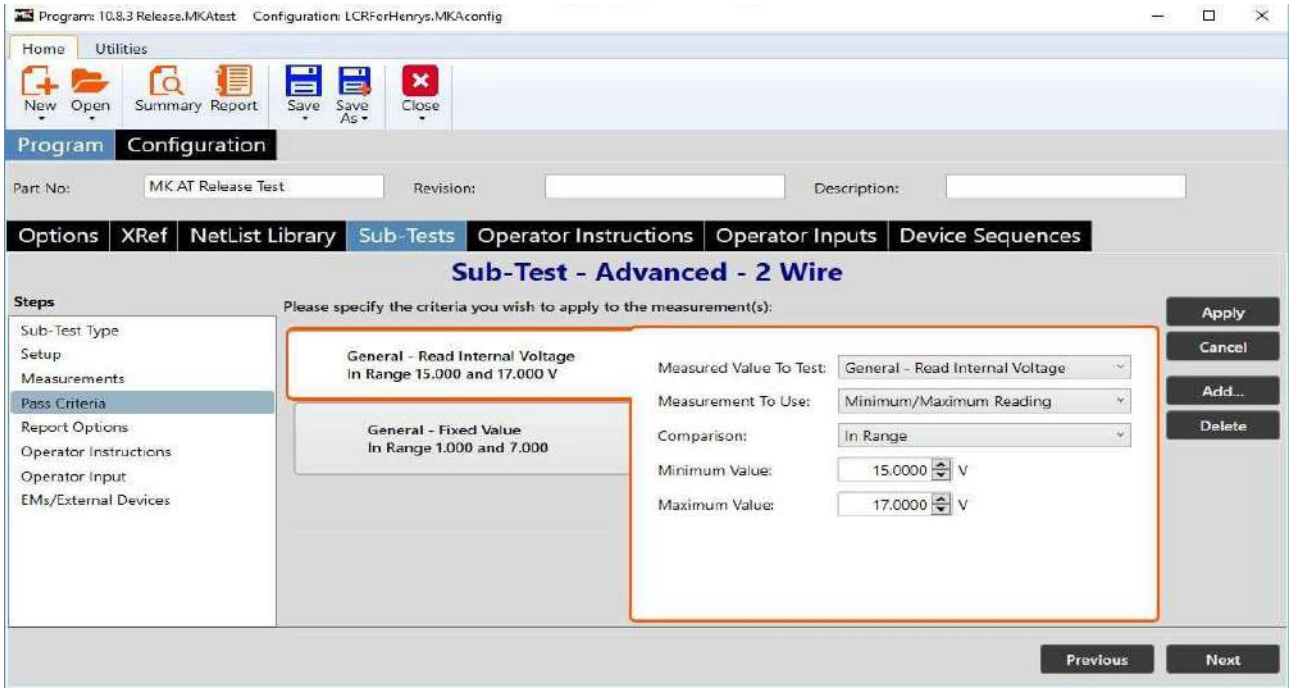


Figure 6: Setup of advanced sub test pass criteria, note that you can have many pass criteria defined.

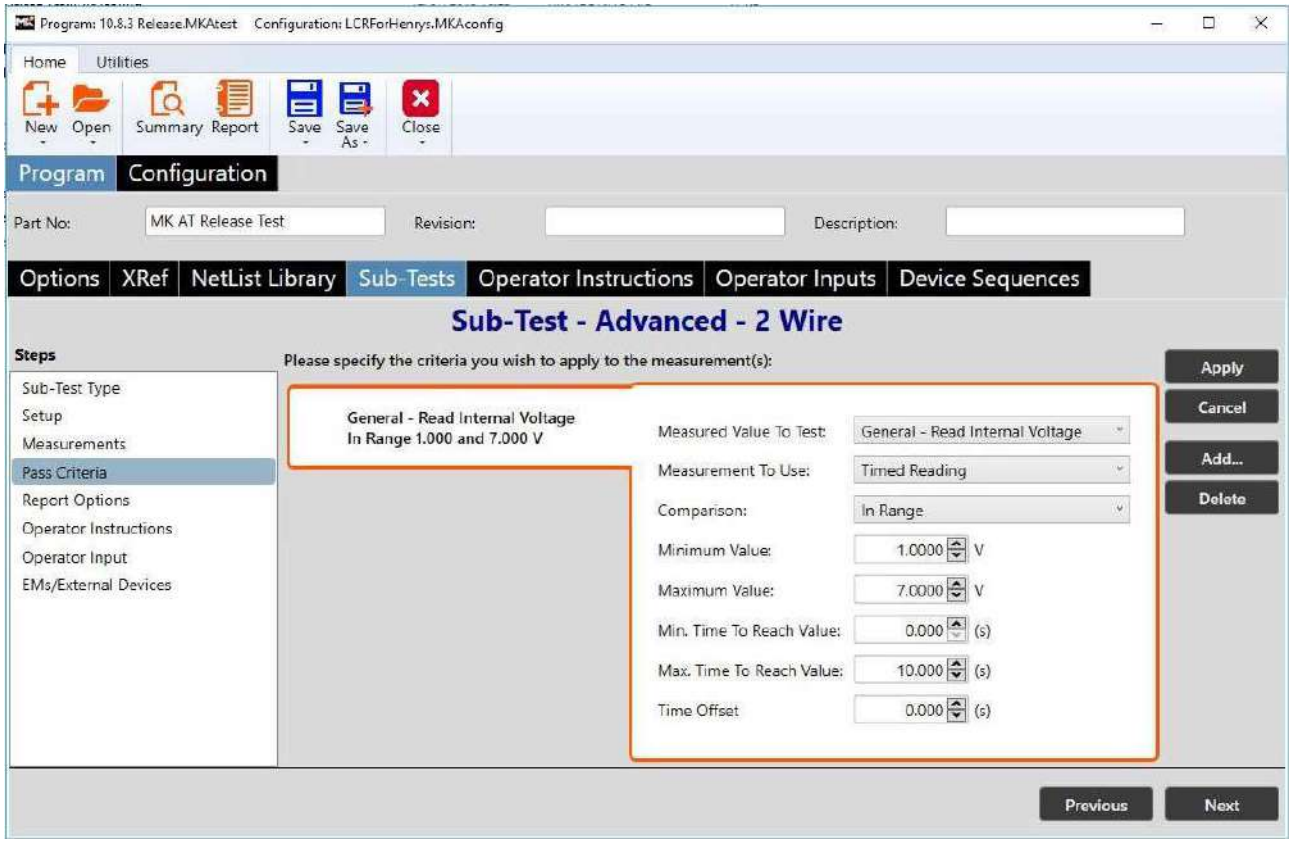
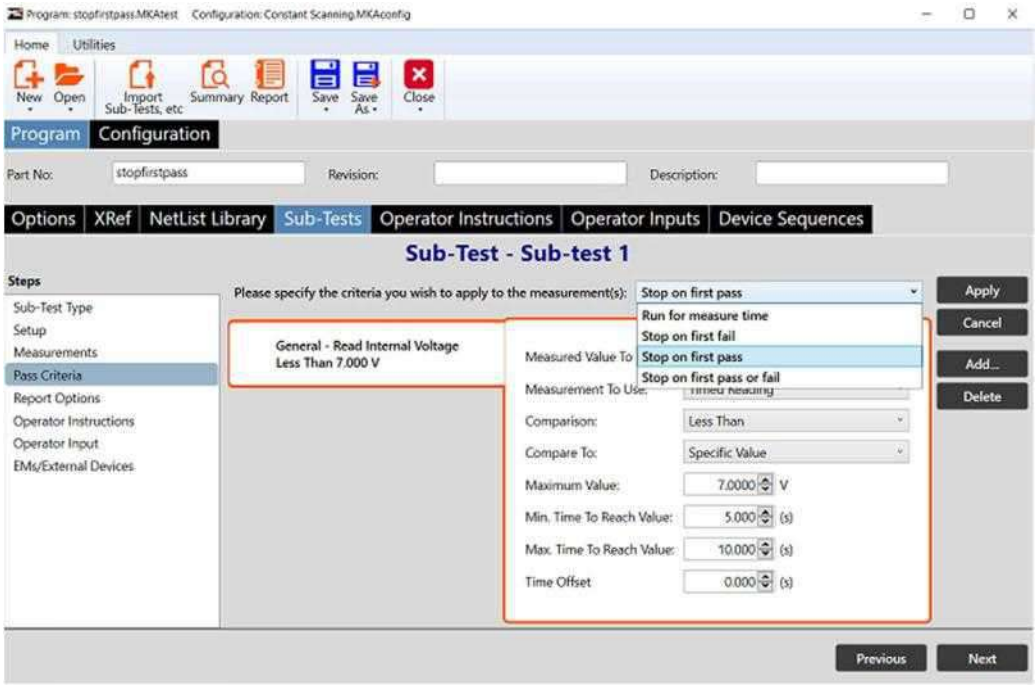


Figure 16: Advanced subtest setup for timed measurement.

The other options are similar for all test programs, i.e. the addition of operator instructions, inputs and device sequences is the same for the advanced sub test as they are for other sub tests.

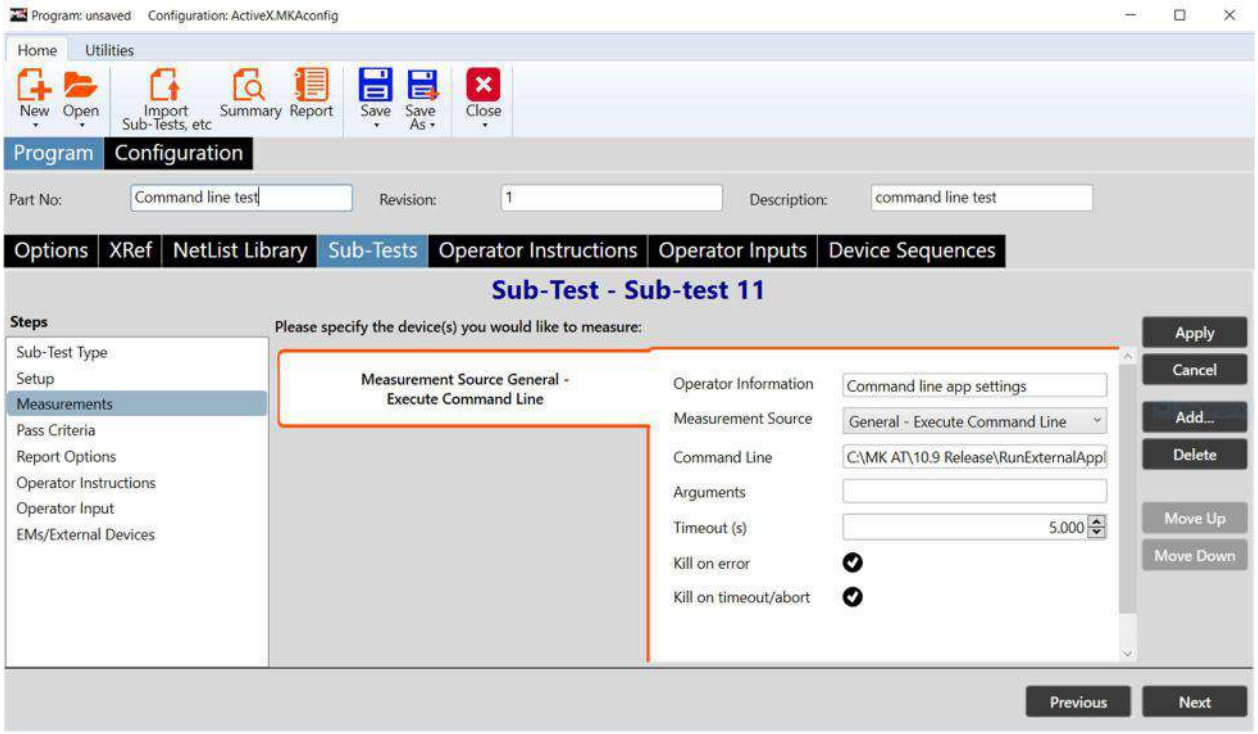
As of version 10.10.1 there is additional pass criteria

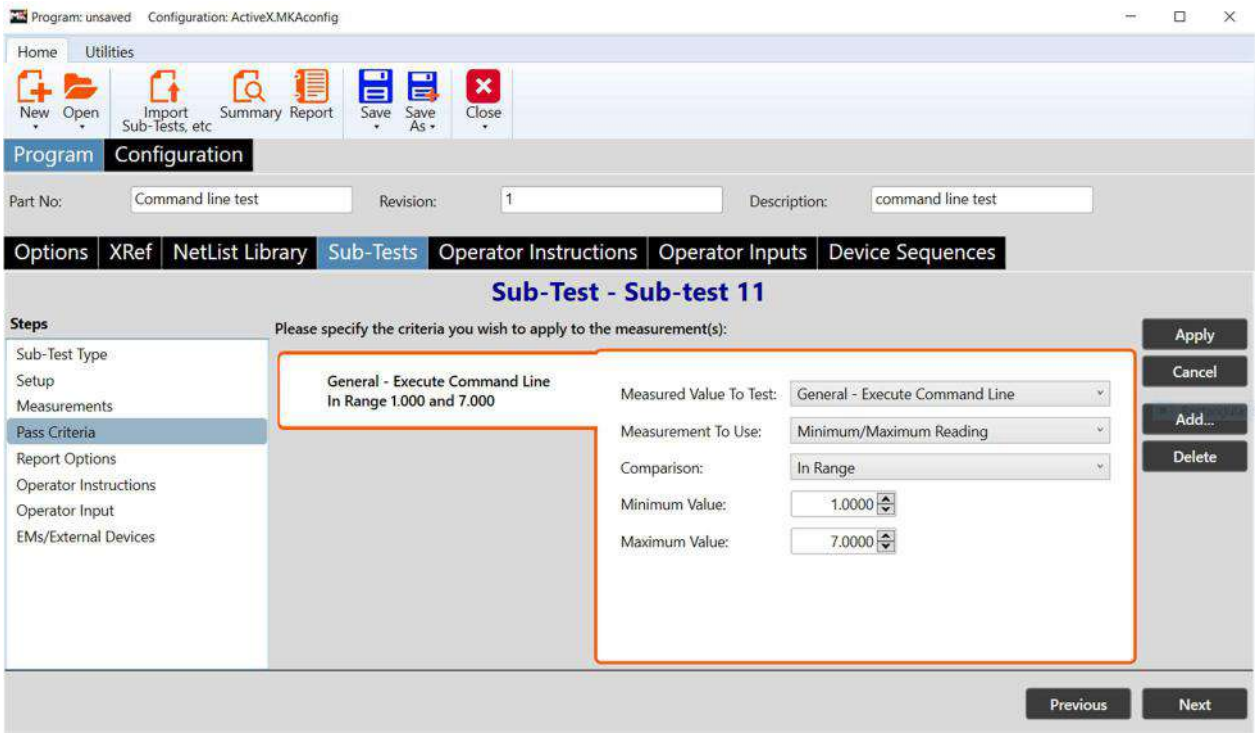


- Stop on first pass:
    - Test stops immediately with a pass status as soon as the target is achieved inside the specified time window. For all other cases the test will continue for the specified Measurement Time.
- Stop on first fail:
    - Test stops with a fail status as soon as a failure is measured. In the example above a reading of less than 7v at any time outside of the 5-10 second time window will end the test immediately with a fail status.
    - Where the target is achieved the test will continue for the specified Measurement Time then the test will stop with a pass status.
- Run for measure time:
    - In all conditions, the test will run to the end of the specified Measurement Time. The only condition that will result in a pass status is when the target is achieved in the specified time window.
- Stop on first fail or pass:
    - The test will stop immediately under pass or fail conditions.

3.1.1.5 Command line runner

A command line application can be run from an advanced test type, with the return value being used to determine a pass or fail. The *Measurement Source* is *General- Execute Command Line* and the field *Command Line* holds the path to the application to run.





3.1.1.6 Adding operator instructions.

Operator instructions are embedded rich text documents. The initial screen shows you a list of operator instructions that you have defined within the test program. Each operator instruction can be used any number of times across the subtests, and within a subtest. When adding instructions to a subtest you can define when they are shown - pre-subtest, post-subtest, post-subtest when failed, post-subtest when passed.

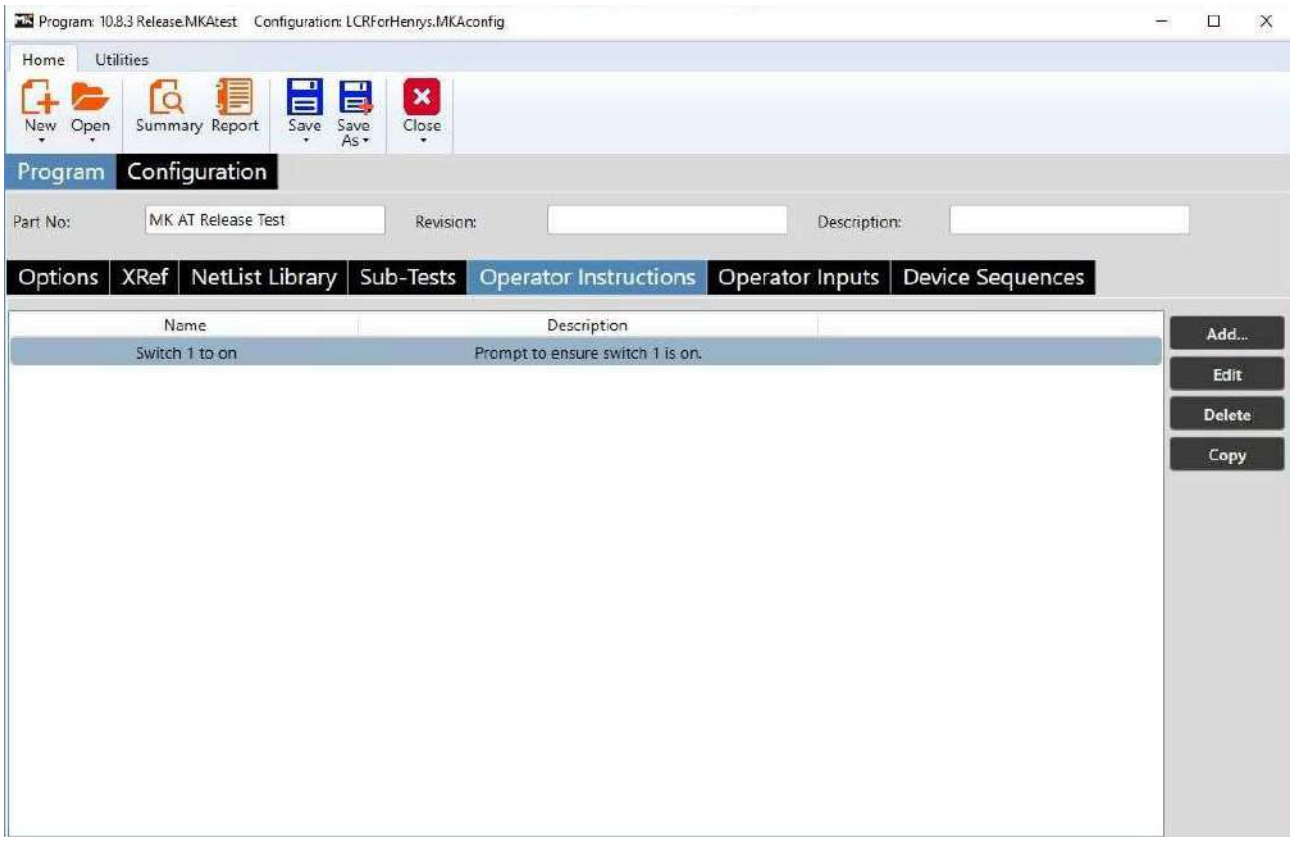


Figure 17: List of operator instructions available to program.

3.1.1.7 Adding operator inputs.

There are 3 types of operator input, and for each type you enter the question / message that the operator will be presented with, along with the data entry field appropriate to the type of data required. Details of the inputs types are as follows:

- Text input. This is a simple non-validated data entry.

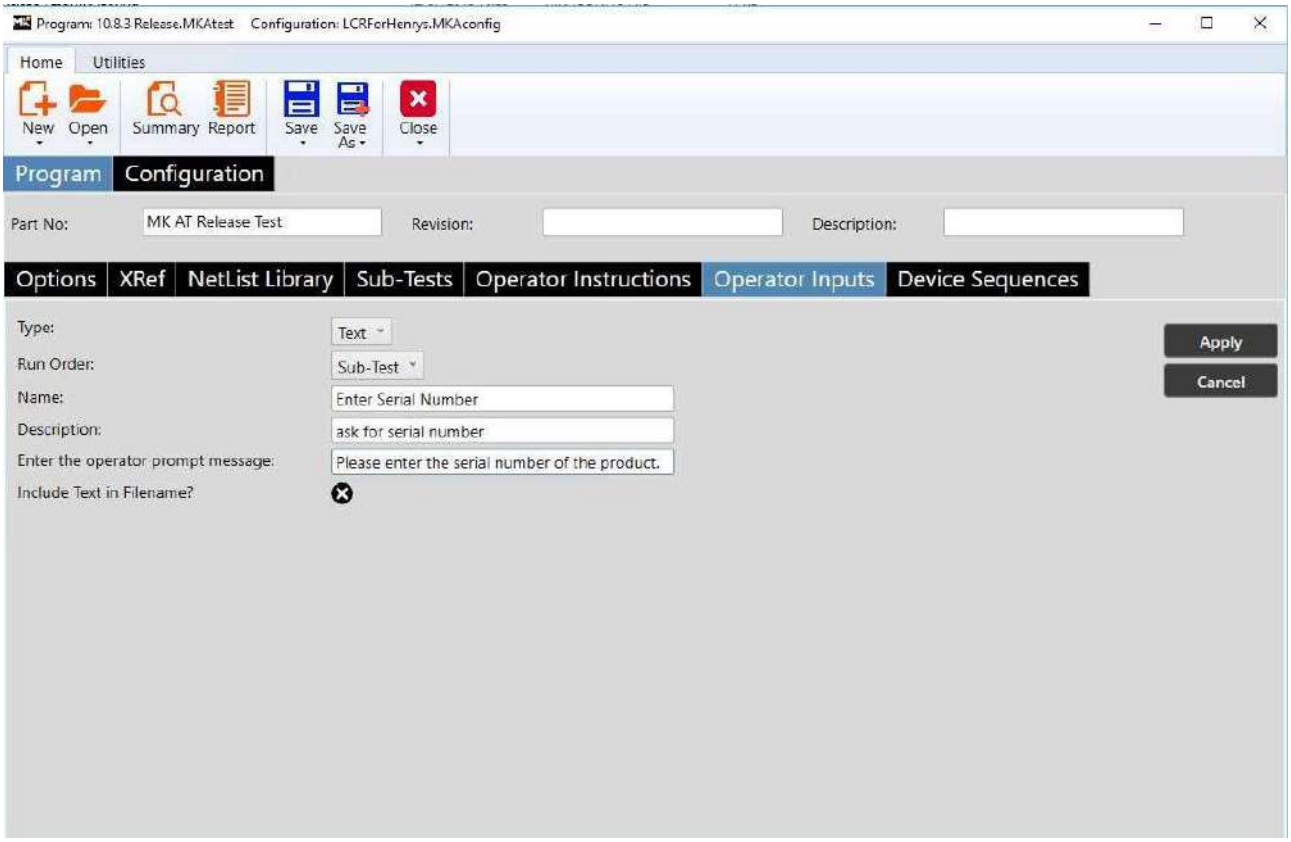


Figure 18: Setup for asking a simple text-based operator input.

- Value input. This allows a subtest to pass / fail if the value is outside the specified limits. You can specify the type of limit to be applied, as shown in the following screen shot.

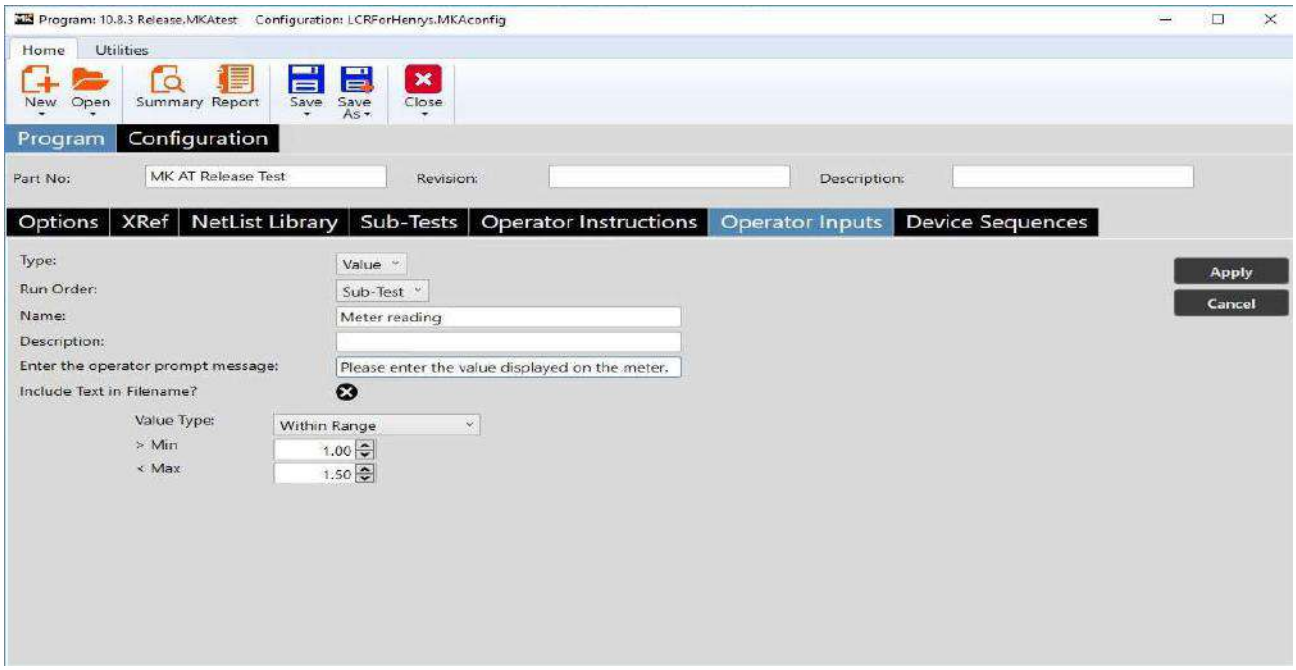


Figure 19: Value input setup. Values can be tested within limits, > Value, or < Value

- Question and Answer: The operator can select from the answers provided, each answer can be a test pass or a test fail. You can have as many pass or fail answers as required.

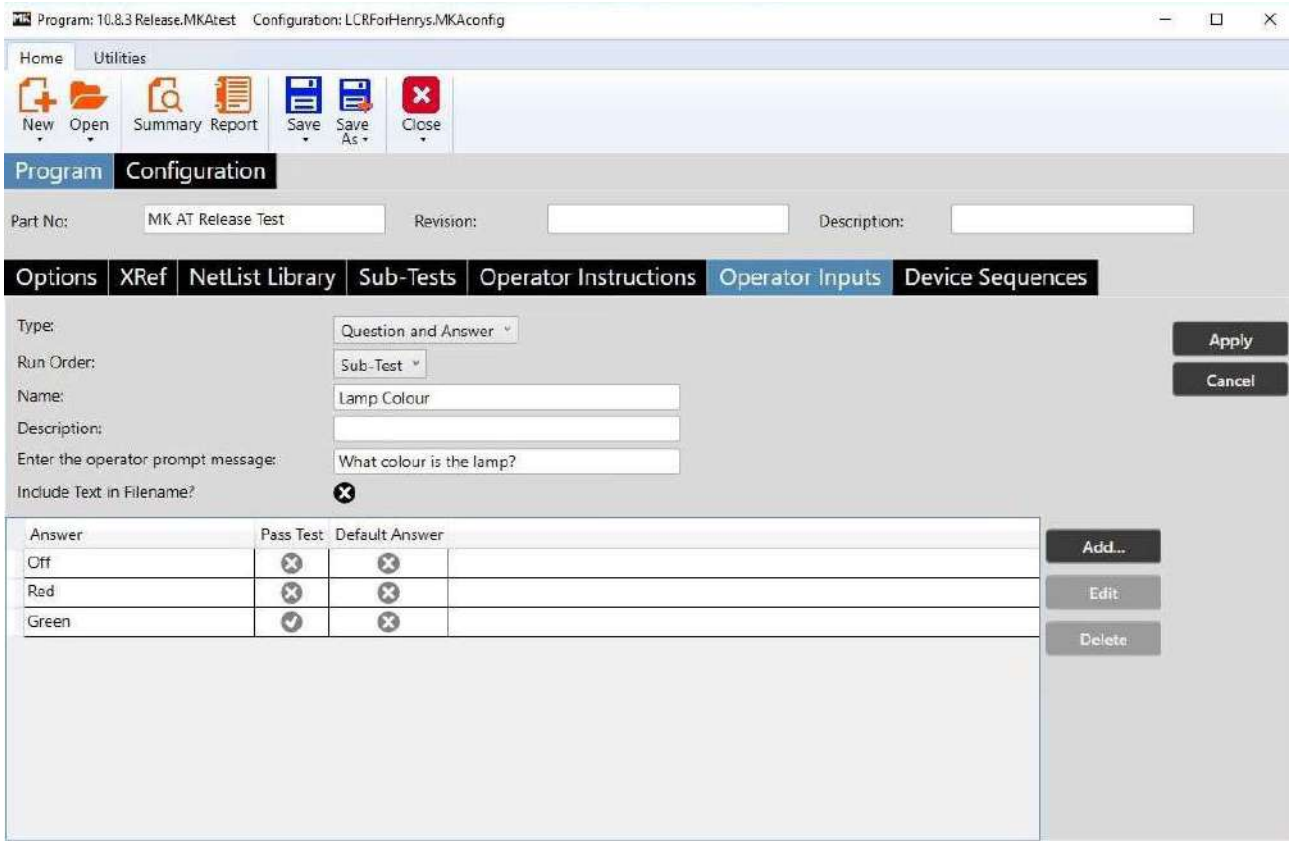


Figure 20: Question with many answers. Each can be used to determine sub-test pass / fail state.

3.1.1.8 Device Sequences

A device sequence is a series of commands to be executed on external, or some internal devices (such as EM's). Depending on your system configuration, you may see different devices available in the drop down that you can select to execute commands on. A device sequence can be made up of any number of commands, but we recommend that you restrict the sequence to one device. This will allow better re-use of commands. For example, if you have a single sequence to set a voltage and output, it will not be easy to reuse. However, you could have a separate set voltage and set current command, then a new sequence to just turn the power supply on. This way you could re-use the power supply 'on' sequence again. Also, if you find that you need to add a delay to allow a power supply to activate, then it would be easy to add to one power supply 'on' sequence, rather than having to find many sequences where you may have to use the 'on' command.

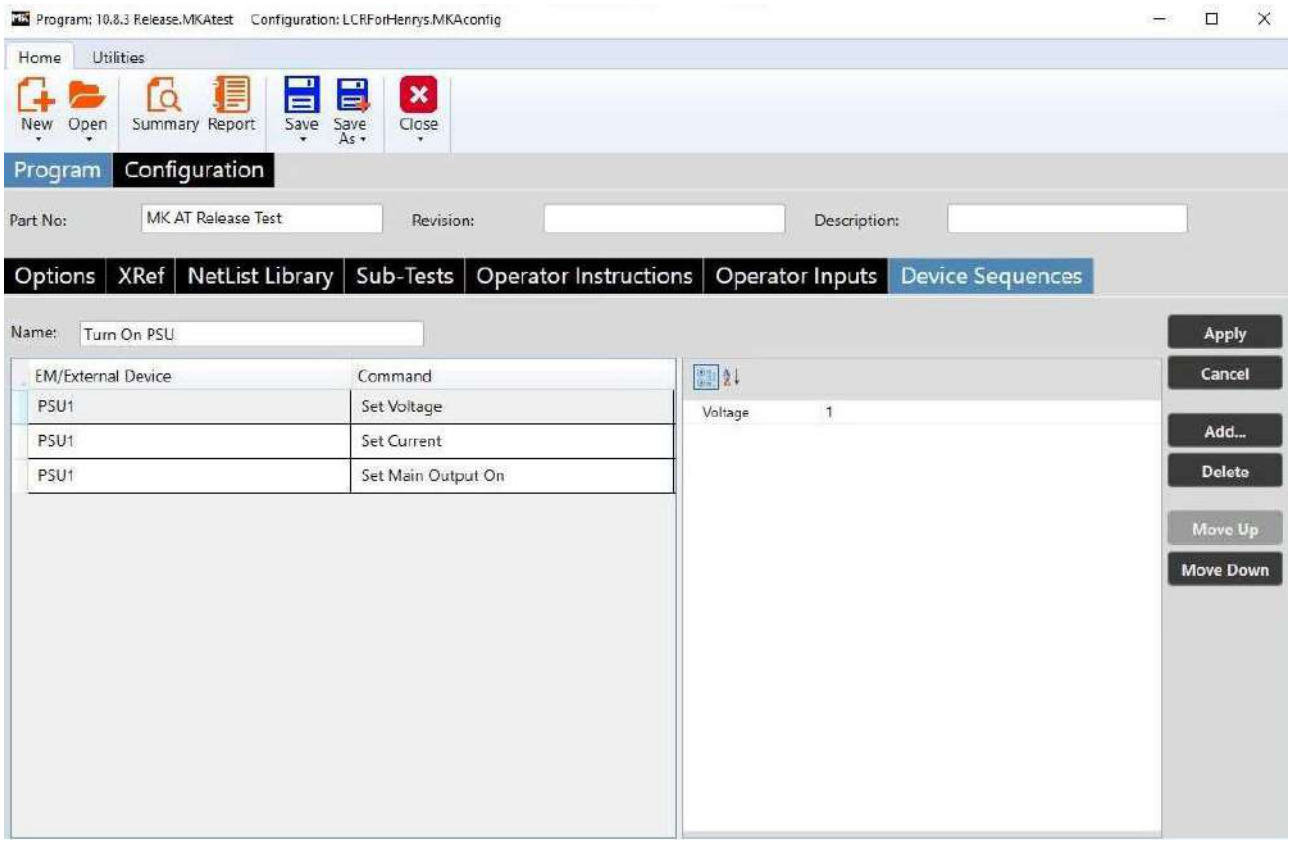


Figure 21: Example of setting voltage and current on PSU1. Note – consider the commands you allocate in one sequence carefully to avoid unnecessary sequencing and longer test times.

To turn on any EM's that are present in your system, you need to select the relevant action (either changeover or setting EM to channel A or B). EM's are always numbered consecutively within a system.

3.1.1.9 Sub-test creation.

When creating / editing a sub test then the sub-test tab will switch to a wizard mode to help guide you through the steps required to create the sub test.

- The first step allows you to name and select the type of test. You cannot change the type of sub-test when editing an existing sub-test. If this is necessary, you would have to delete it and create a new test of the correct type.
- The next step allows you to define the "defaults" applicable for the selected sub test type.
- Next you define the connection / net lists and parameters that are applicable for that test.
- The next screen allows you to define the options for this sub test. These are the "actions" that execute when a 'fail' condition occurs.
- Operator instructions are next. These can be defined as pre sub test, post sub test, post sub test on pass or post sub test on fail. If you have not yet created the operator instruction, then you can simply switch to that tab, create the instruction then switch back to the sub test tab and add it.
- Operator inputs and EM / External devices (Device Sequences) work in the same way as operator instructions.

You need to ensure that you have selected the correct "test mode", i.e. ensure that you select two wire (or deselect it to use four wire measurement).

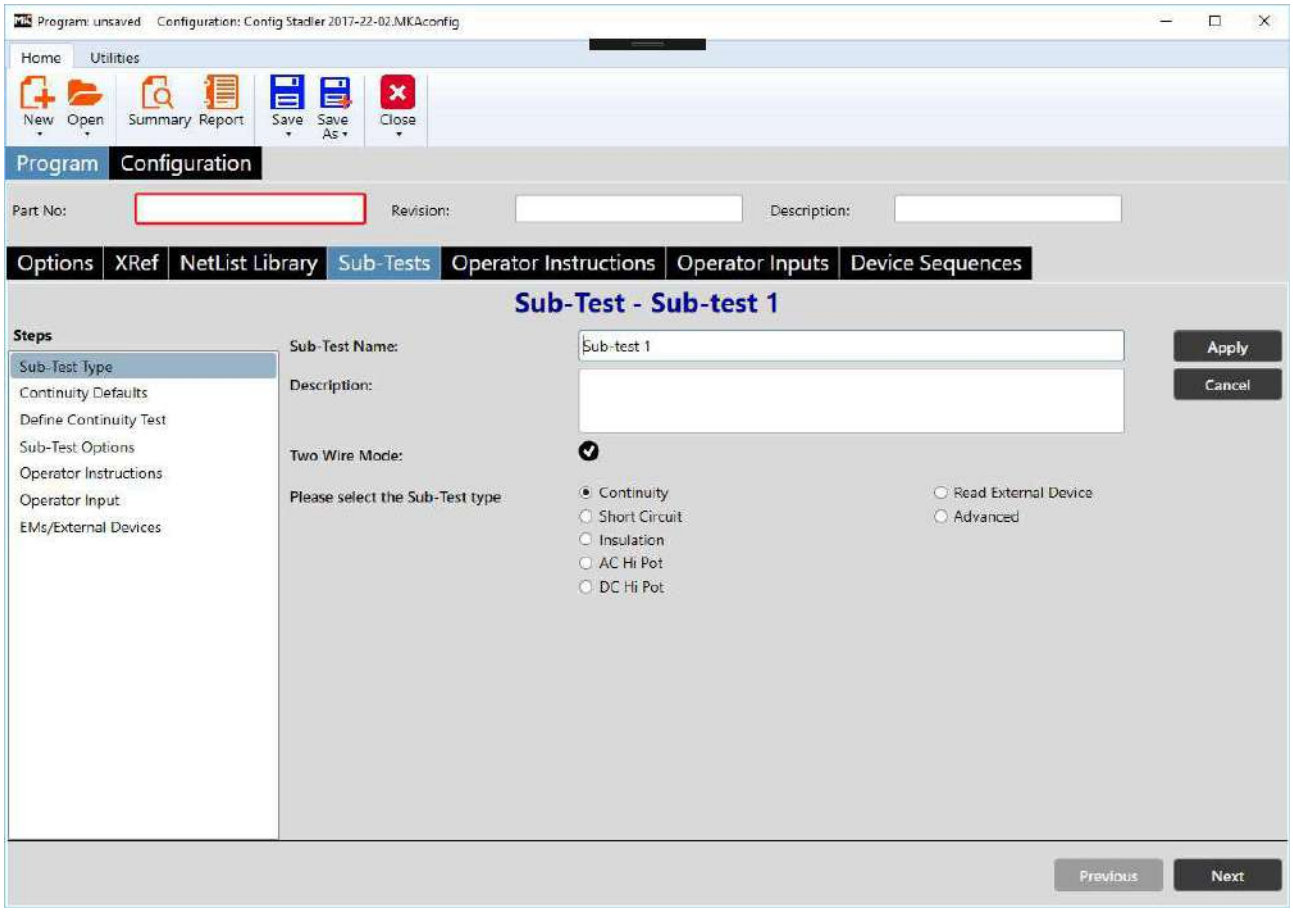


Figure 22: Edit / create sub test wizard.

If you have edited a sub-test but do not wish to retain the changes, selecting 'Cancel' will revert any changes made. To ensure that your changes are saved, you must press the 'Apply' button.

Once you have created and selected the sub test type that you wish to create, you will see a set of sub test specific screens, as detailed in the following sections:

3.1.1.9.1 Continuity Sub-Test creation.

The first continuity specific defaults screen appears as shown below:

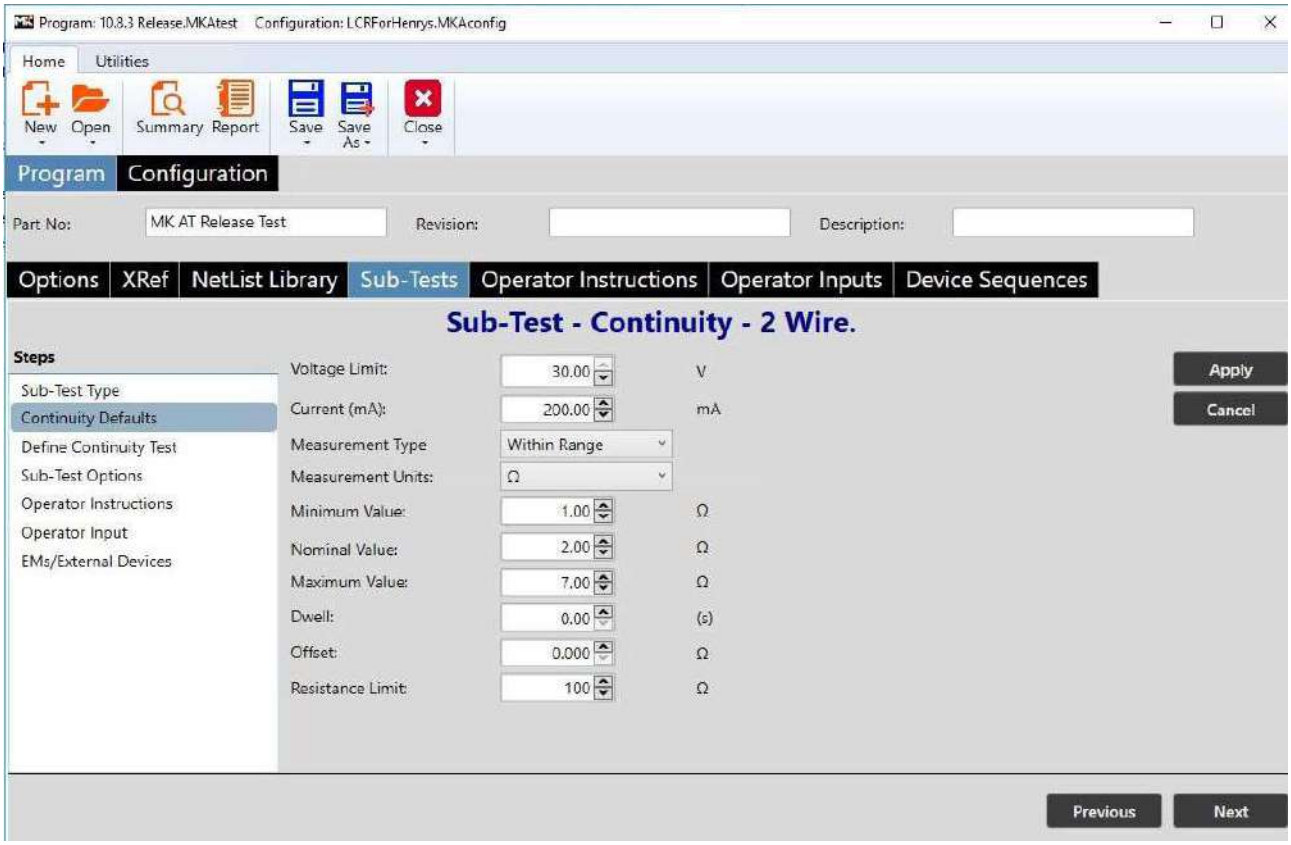


Figure 23: Continuity defaults screen.

This screen allows you to set the default parameters that are used when you add a new continuity measurement.

- Voltage Limit: This is the maximum voltage that the system will deliver (note this may not be the actual test voltage, as we are working in constant current mode. This is the voltage that will not be exceeded if we discover open circuits etc).
- Current: This is the current that you wish to test with. Please note that this must be set to an appropriate value for the resistance you are attempting to measure.
- Measurement type: This allows you to specify how you wish to apply the limits, and hence the pass / fail of the test. There are 3 type of measurement type:
    - Within Range - the measured value must be > minimum and < maximum.
    - > Min.
    - < Max.
- Nominal value: This is the nominal value that you expect the resistance to be.
- Minimum value. Dependent on the measurement type, this may be used as part of the pass / fail criteria
- Maximum value: Used in the same way as Minimum value.
- Dwell: time in seconds that the system will wait after the current has been applied, and before taking a measurement. Required if the connection needs time to charge.

- Offset: allows an offset resistance to be input and calculated during the measurement.
- Resistance limit: This is the maximum resistance that the system will attempt to measure (the overall system limit for LV is 100K Ohm). The greater this value is over the nominal value, then MK AT will make more measurements at different current settings from those specified in the test. This can make the system run slower if there are many connections that are failing the initial measurement. If finding the actual resistance measured is not required, then this can be set close to the maximum value (10% above is recommended).

The next screen allows you to define the connections you wish to test.

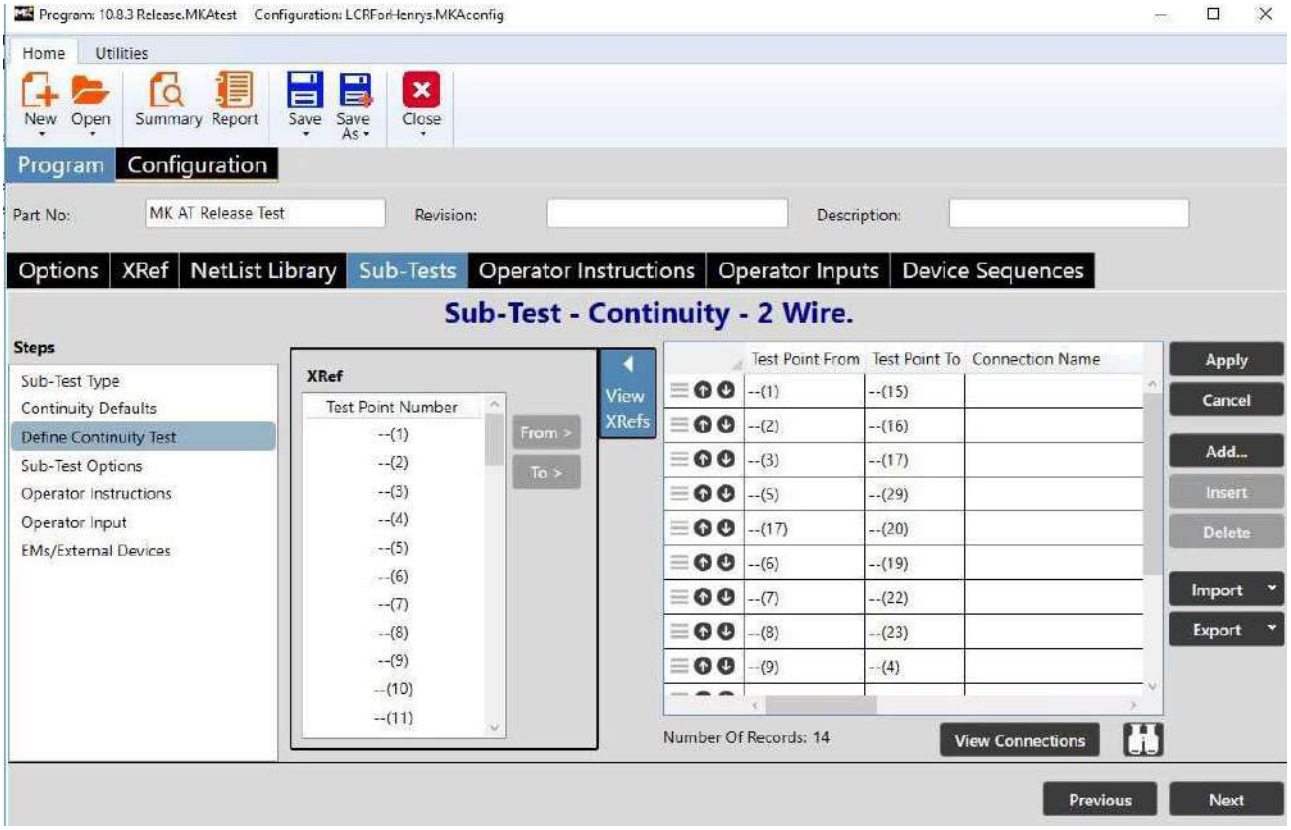
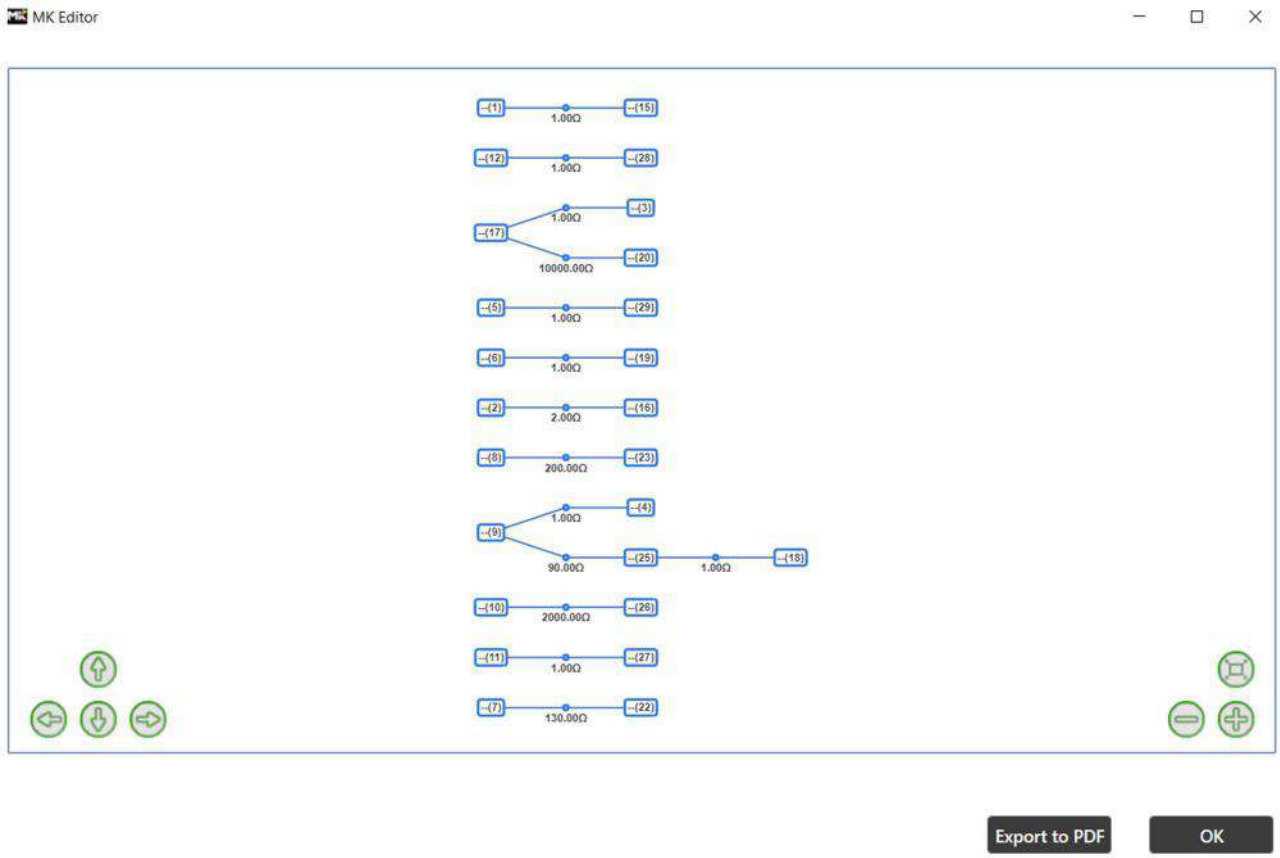


Figure 24: Define continuity screen showing the *View XRef* section expanded

Here you can add or edit the connections and the parameters which are used to measure the connection. Each item added will have the default values applied from the previous screen. If you click in the cell you can edit the entered value. You can either enter the test point number directly in the "from" and "to" test point columns, or they can be selected from an existing xref. The xref data can be viewed by selecting the *View XRefs* button, as shown in the above example, then the "*From >*" and "*To >*" buttons can be used to populate the fields directly. If you edit a test point, you can only enter the test point number. When the edit is complete, the cell will then show the test point, complete with XRef formatting, if this has been defined.

You can import or export data directly to or from this list in either XML or tab separated value TSV formats. This will allow you edit or create the connections in other editors and re-import them, if required.

The *View Connections* button provides a graphical representation of the connections that have been defined in the continuity sub test. This opens in a new window as shown below:



The connections are labelled with their xref names, if available, along with their test point number in brackets and the resistance values shown are the nominal values for the connection. The layout diagram can be navigated using the arrow buttons and the plus and minus buttons zoom in and out and the button fits the layout to the window. The diagram can be exported to PDF, if required using the *Export to PDF* button.

The *Sub-Test Options* screen allows you to define what will happen if the system encounters a failure whilst testing.

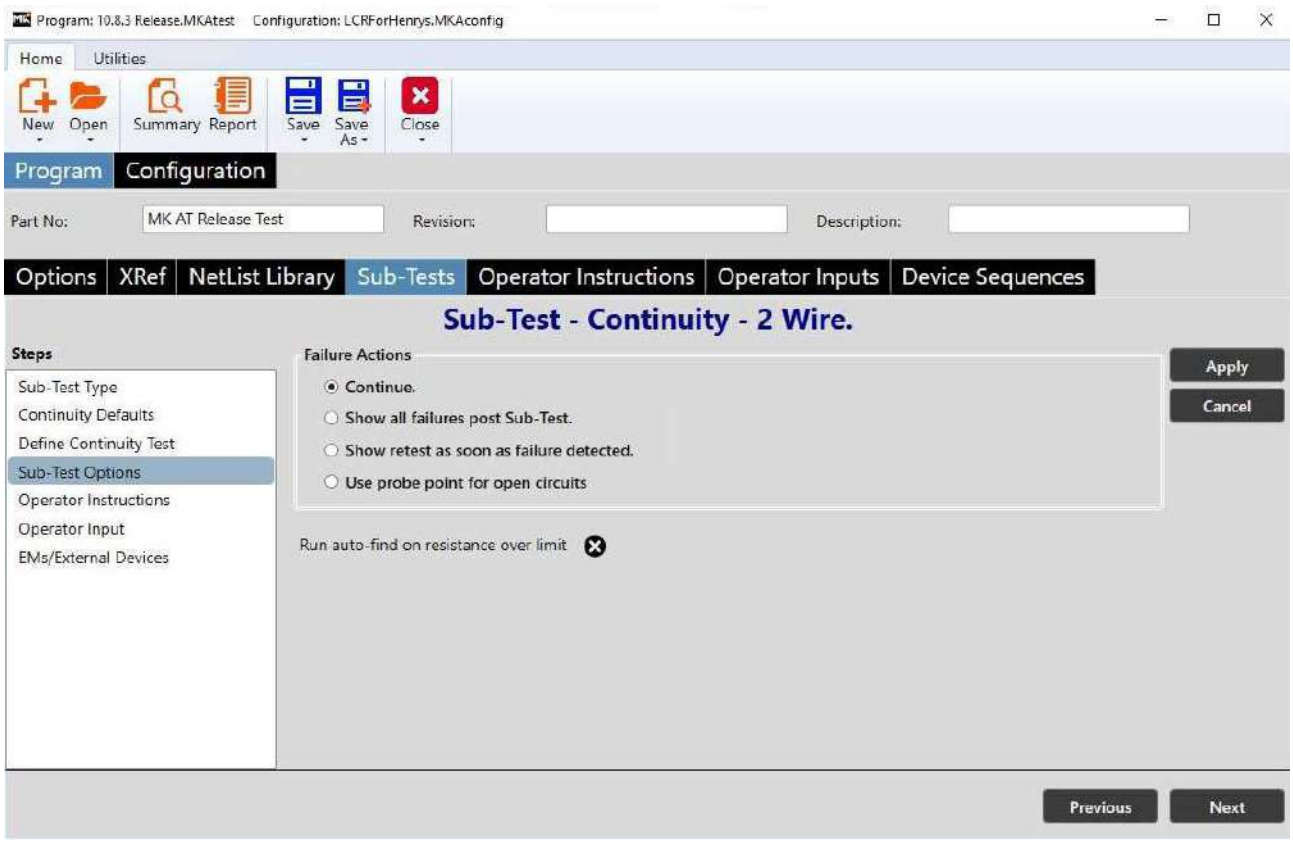


Figure 25: Options to be applied when measurement fails.

- Continue: the system will just continue to the next measurement and record the failed item.
- Show all failures post Sub-Test: This will "pause" the testing and show the operator all the failures in that sub-test. Each failed item is shown with the option to re-test the item, or skip. If retest is selected, the system will test that item again. If it still fails it will remain in the list. If the item passes, it is removed from the list and the final overall report will not reflect the new passed state. If the Skip option is selected, then the original failed state will be reported on the final report. This option allows you to fix any continuity failures before moving on to the next sub-test.
- Show retest as soon as failure detected: This is similar to the above option, but it will show the retest screen as soon as the first failure is detected, so only one failure will ever be shown. The other actions work the same as the Show all post sub-test option.
- Use probe point for open circuits: if a failure is recorded, the re-test screen loads with the option to use the probe point for diagnosis.
- Run auto-find on resistance over limit. This assists in finding mis-wires, etc. If a continuity test fails with resistance over limit (possibly an open circuit) we then test both ends of the continuity test to "find" if there are any short circuits to the test points. This is reported so the user knows where the mis-wire has occurred.

The next screen allows the addition of operator instructions to a sub test. More than one operator instruction can be applied. Each instruction can be shown at the start of the sub test (Pre_SubTest), when the sub test fails, when the sub-test passes, or always when a sub-test ends.

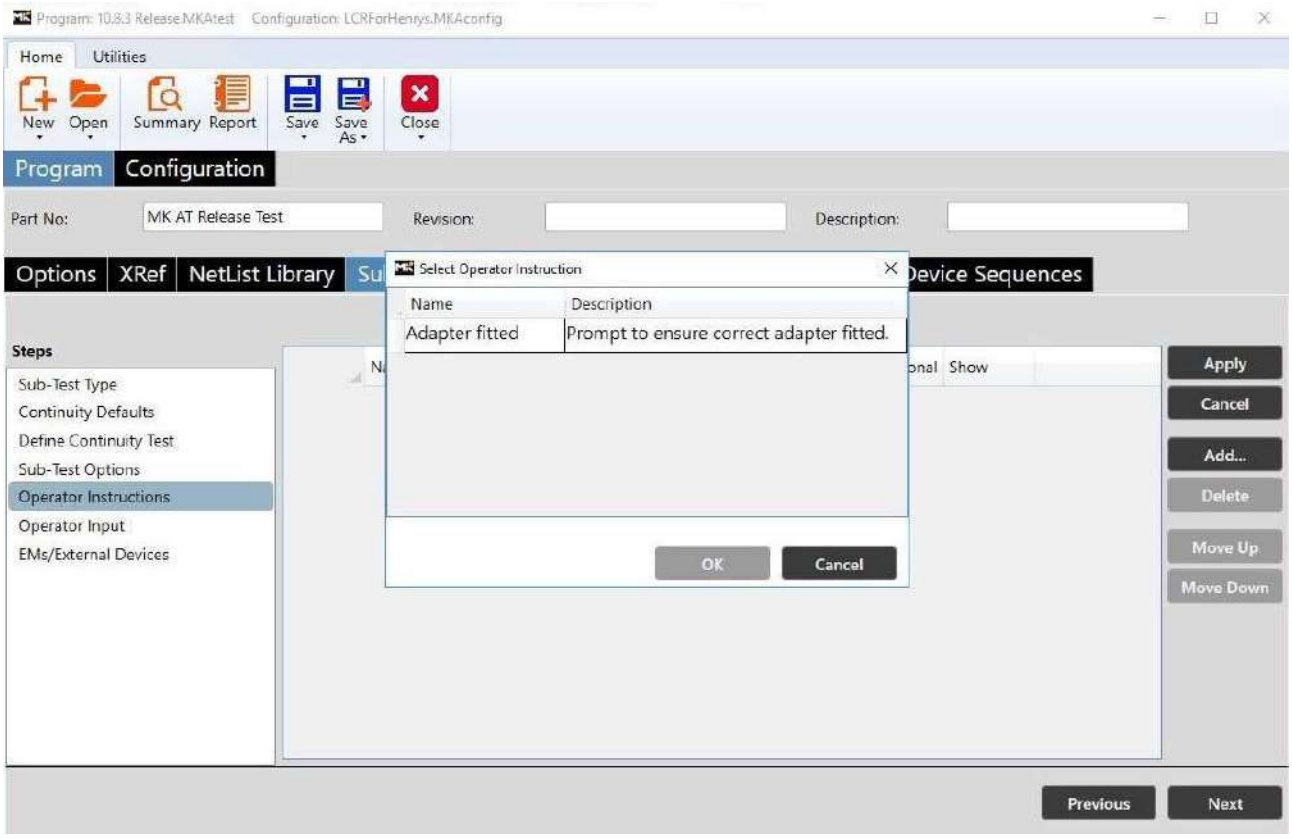


Figure 26: Selection of the operator instruction to be added to the sub test.

The next screen that defines operator inputs works in a similar manner to the previous operator instructions.

The last screen concerning device sequences is similar to the previous two, but allows device sequence definition and action.

3.1.1.9.2 Short circuit detection.

Short circuit sub-tests are designed to quickly identify any low resistance shorts circuits that may exist in the system. The sequence of screens is similar to continuity. You first select the defaults values that you wish to use to find short circuits.

The next screen allows you to define the “netlists”. A net list is a collection of test points that are common (or need to be treated as common) and will be tested together to find any low resistance connection to all other net lists defined WITHIN the sub-test. It is important to remember that only netlists defined are used to find shorts, so if you wish to exclude something from testing then it is simple task to delete that net from the list. When you are first presented with the screen, you will be shown the following dialog:

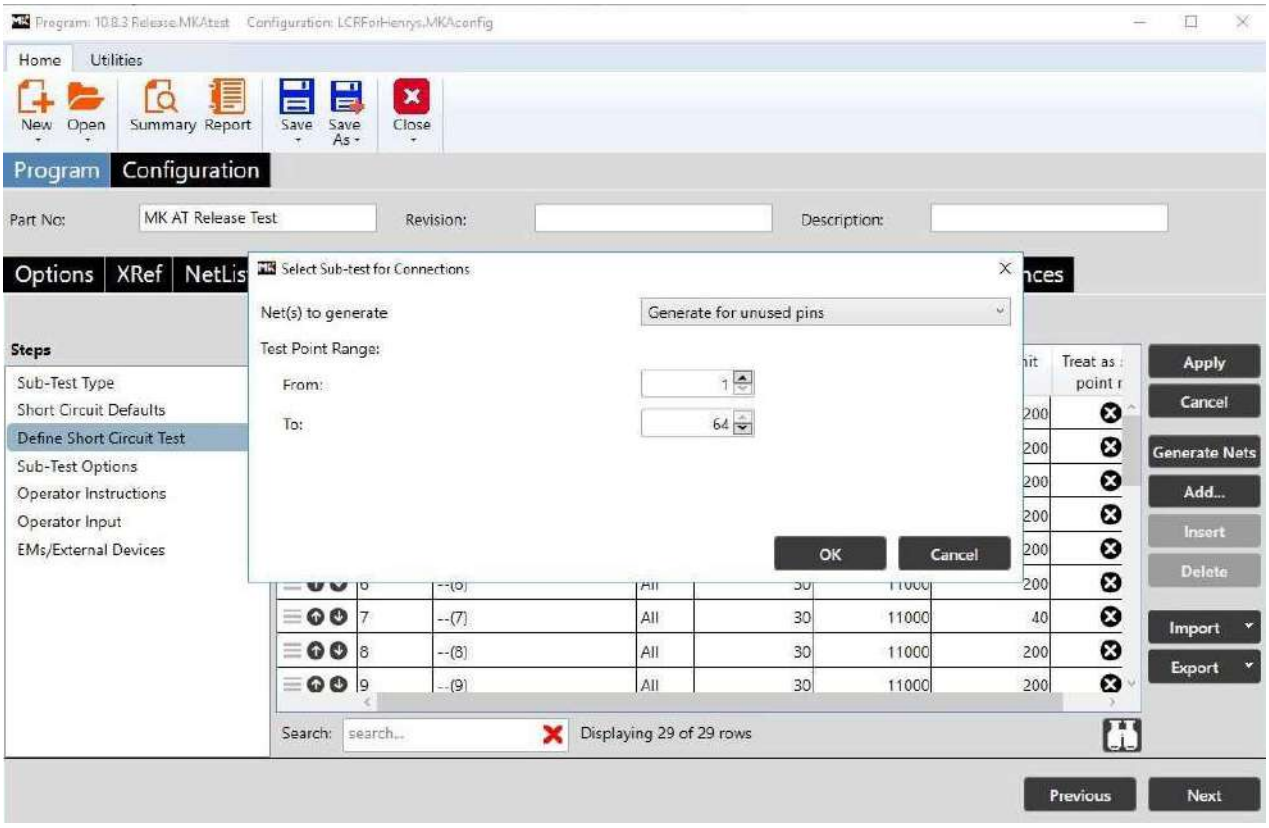


Figure 27: Generating net lists for unused pins dialog.

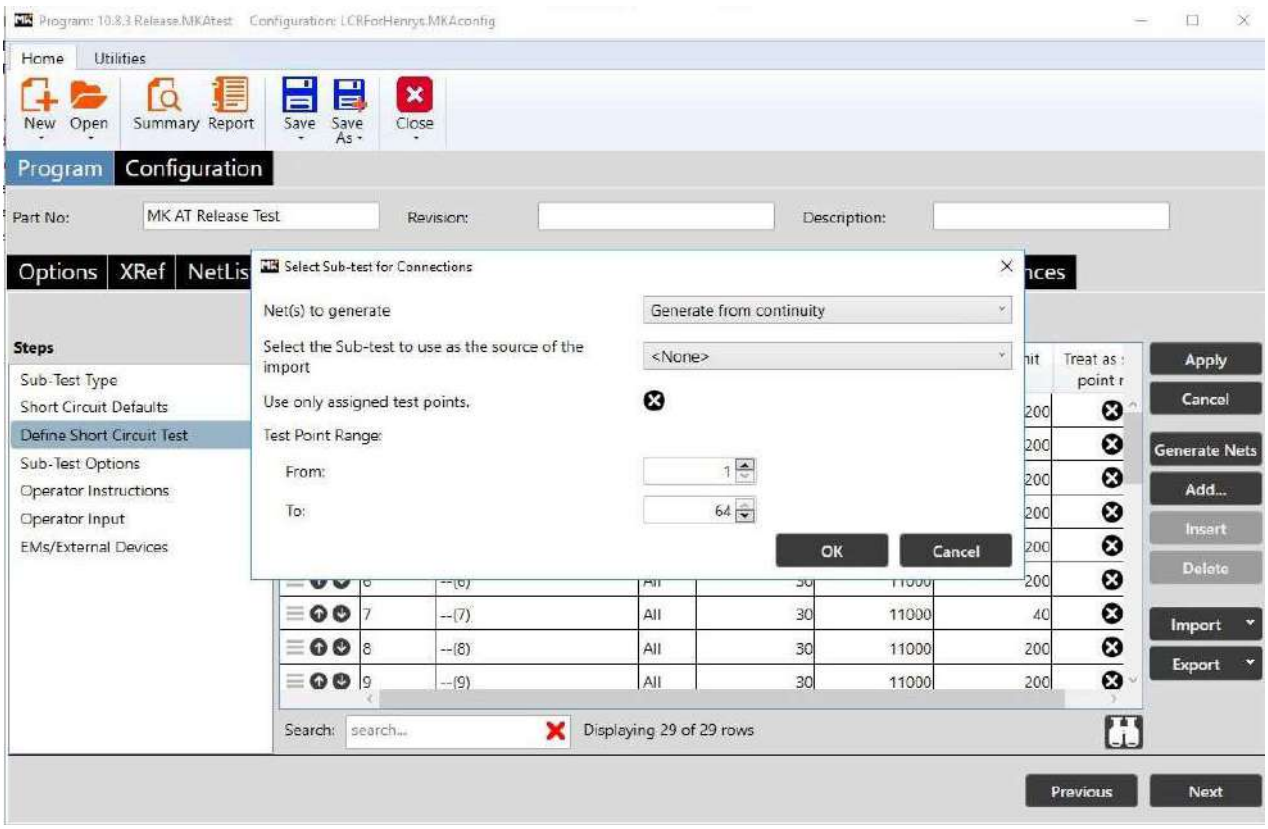


Figure 28: Connection selection dialog to generate netlists.

This allows you to automatically generate the netlists from a continuity sub-test, and control the test point range that is used to generate the nets. Using this option, the software will automatically search for any common connections within the continuity program and generate the netlist.

You can also select the 'Generate from unused pins' option. This will use the list of nets currently defined and generate a single net with all the other pins in it. It will also select the "treat as single point nets" option. This means that although there could be many points in the net, when testing we will treat them as individual nets. If a short is found to any point in that net, then the software will report which point it is. This is a quick and easy way to add a large amount of "unused" test points that you want to ensure are not connected, but need to know which point has failed if there is a problem.

If you wish you can then edit the auto generated list afterwards, note that the auto generate clears the current list of nets before generation.

You can also create netlists manually. Enter a set of single points, and use the "," separator e.g. 1,5,8. This would be test points, 1, 5 and 8 included in a single net. You can also enter a range of test points using the "–" separator, so 1-10 would be test points 1 to 10 inclusive. You can mix and match the syntax so 1, 4-8, 24, 55-58 is a valid entry.

Editing the remainder of the short circuit screens is similar to editing the continuity screen.

3.1.1.9.3 Insulation sub-tests.

The creation of this sub-test is very similar to the short circuit sub test, with only the number and type of default parameters changing. There is one minor change in the way the test is carried out. Short circuit is a forward only search, so once we have tested netlist A to netlist B we do not need to test netlist B to netlist A, but for insulation testing we do. If you wish to test in one direction only, set the voltage of the netlist to 0. This will mean that you will test to that TP from the other TP, but never from it. This can be useful when testing screened cables etc. By setting the voltage on the next to TP to 0, it will be shown in the report as being masked.

3.1.1.9.4 AC and DC Hi-Pot sub-tests.

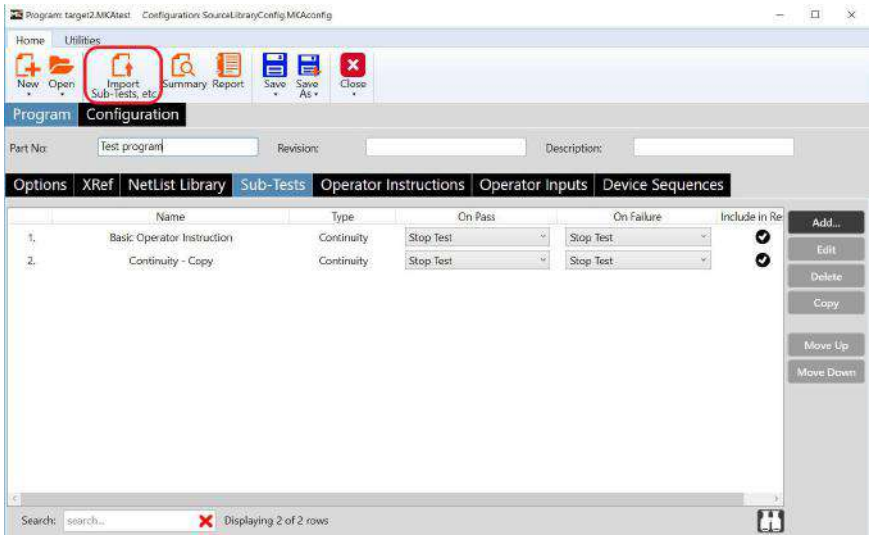
The setup and control of these tests are almost identical to the insulation sub test, the only thing that will change are the default parameters. For AC and DC HiPot tests, the default measurement used for determining a pass or fail is *highest* measurement. This can be changed to *average* measurement if required.

3.1.2 Saving the Program and Configuration.

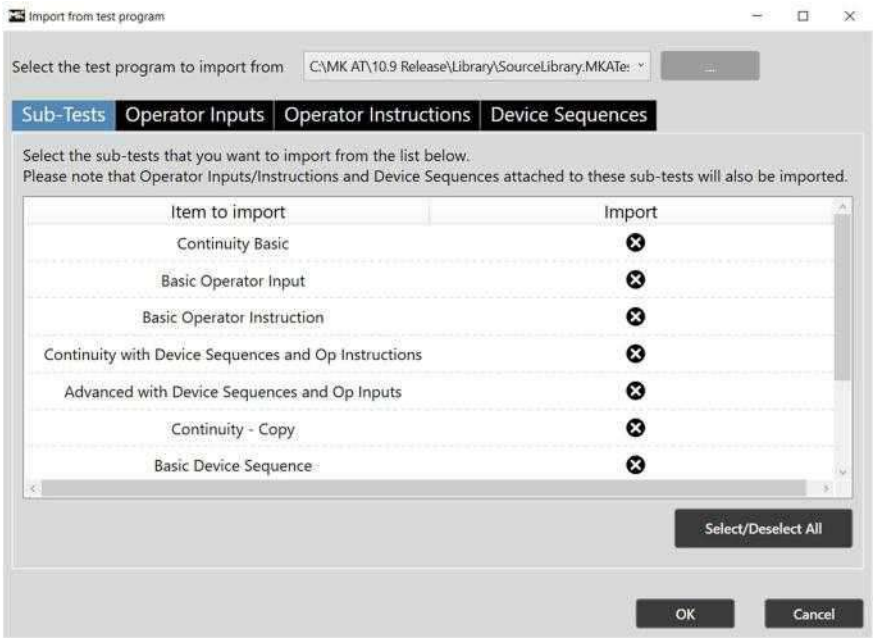
Changes to programs and configurations are not automatically saved. If you wish to save changes, you can save the program via the save menu option. If you wish to save your program or configuration to another file, then use the save as option, and then you can specify the new name and or location.

3.2 Importing Sub-Tests

Sub-tests, Operator Inputs, Operator Instructions, and Device Sequences can be imported using the *Import Sub-Tests* button, shown below.

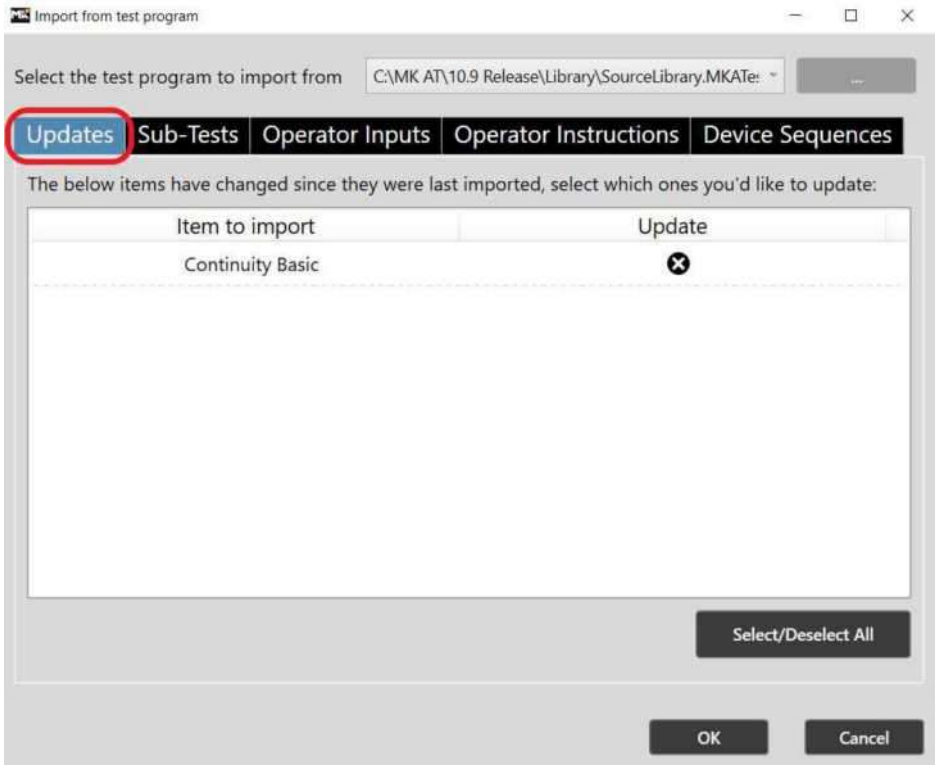


This feature allows sub-tests from existing programs to be imported. Any associated operator inputs, operator instructions or device sequences that are used in the selected sub-tests will be imported as well.



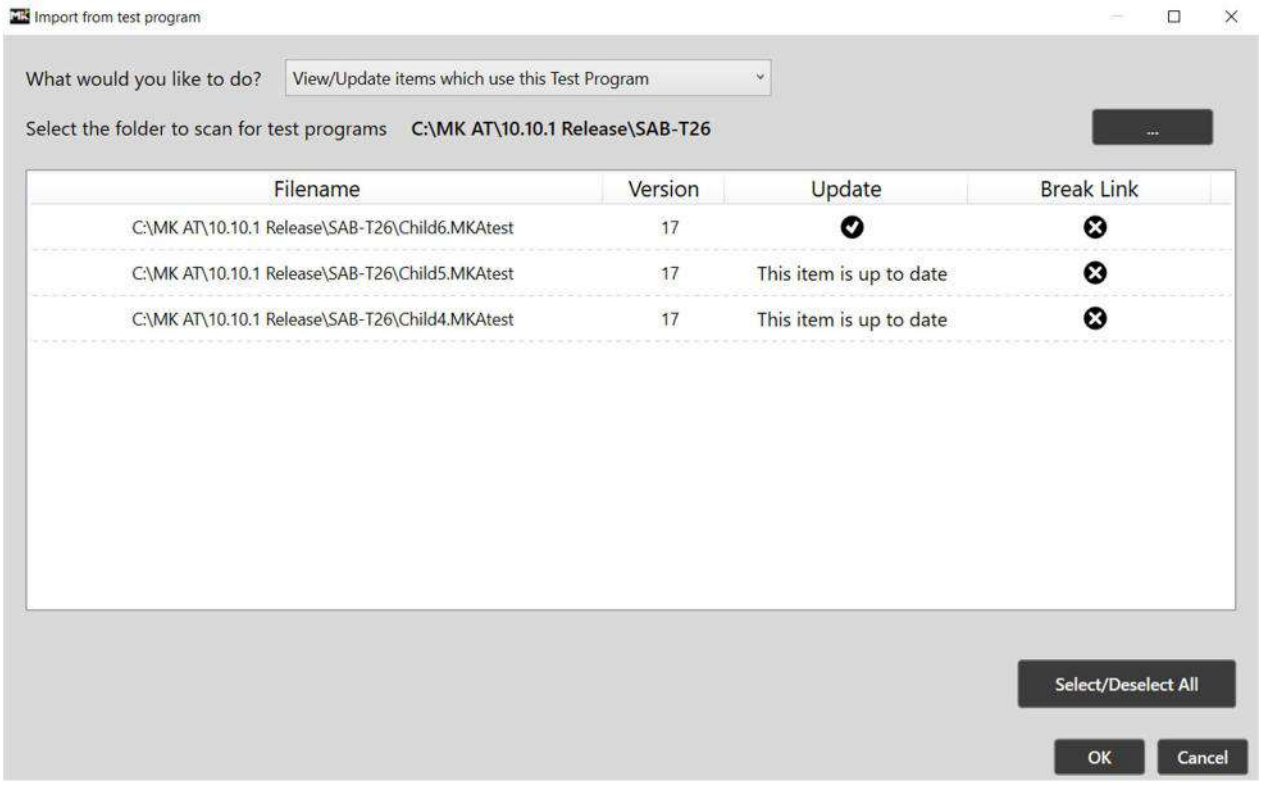
Operator inputs, operator instructions and device sequences can also be imported independently of their associated sub-tests if required.

If the source program is changed, there is the option to update any changes to the imported sub-tests. This is done by selecting the *Import Sub-Tests* button again and any available changes are listed under the *Updates* tab:



3.2.1 Automatically Update Test Programs from a Master Program:

- Changes made to the master template can be pushed to all child programs that imported from the template.
- The import window shown below lists all test programs that have imported from this template.
- If the child program has nothing to update from the template, they will be marked as *This item is up to date*.
- Where updates are available, there will be the option to update the child program.
- The *Break Link* option removes the link between the template and child program and prevents it from appearing in the update listing again.



Utilities menu.

Clicking on the Utilities menu allows access to two different menu options: “APG” and “Verify test results". See the following screenshot:

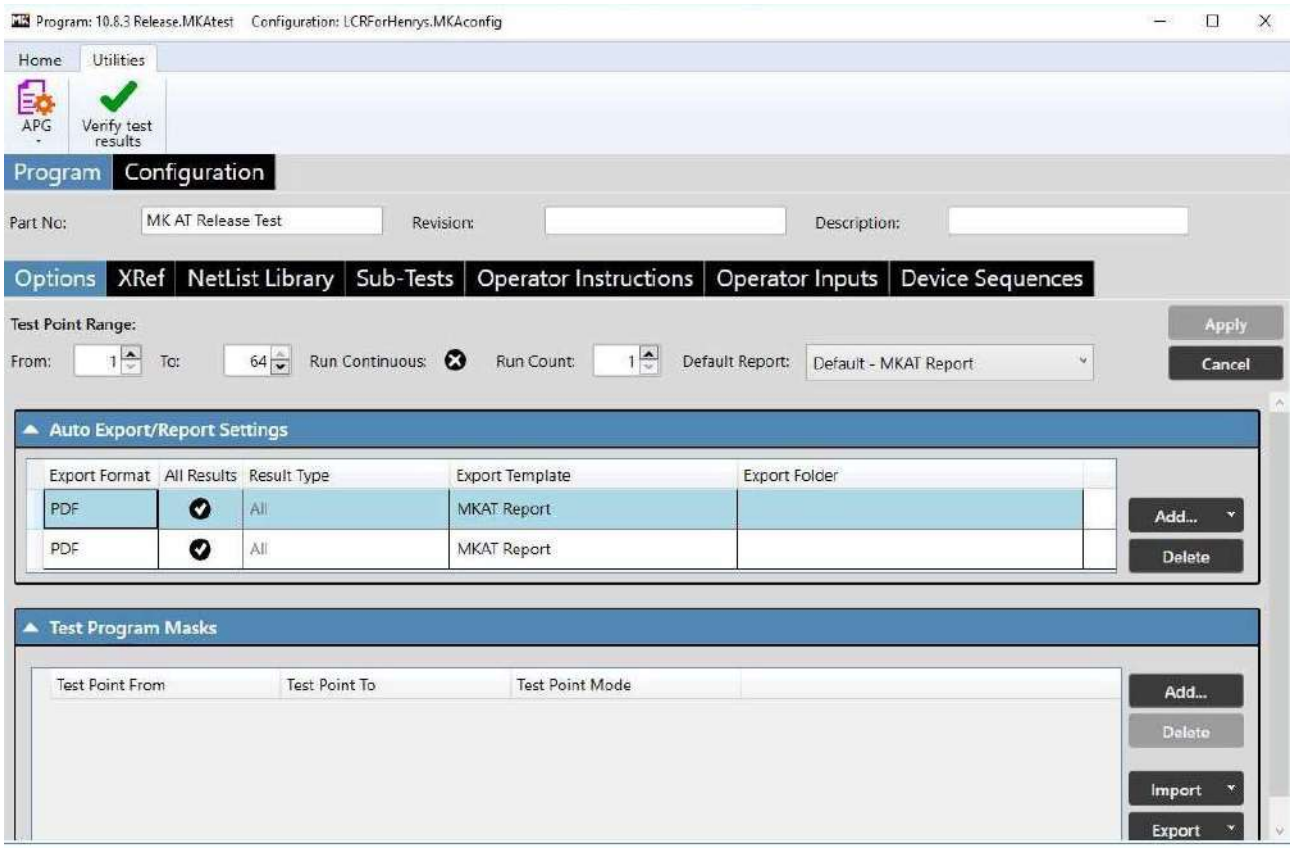


Figure 29: Utilities menu.

3.3 APG menu option.

Selecting the “APG” menu option will reveal a drop-down menu of five different APG options. These options are not covered in this document. Please instead refer to the following documentation for a description of these options:

1. MK AT Standard APG Manual.docx.
2. MK AT Component Library Manual.docx.
3. MK AT active component APG and File Structure Reference Manual 1.0.docx.
4. Gen 2 – Active component APG and Fil Structure Reference Manual 1.2.docx.

3.4 Verify test results menu option.

Selecting the “Verify” option opens a dialog to allow navigation to, and selection of a test result file. Once selected, the software will run a check to ensure the results have not been altered externally and notifies the user of the result.

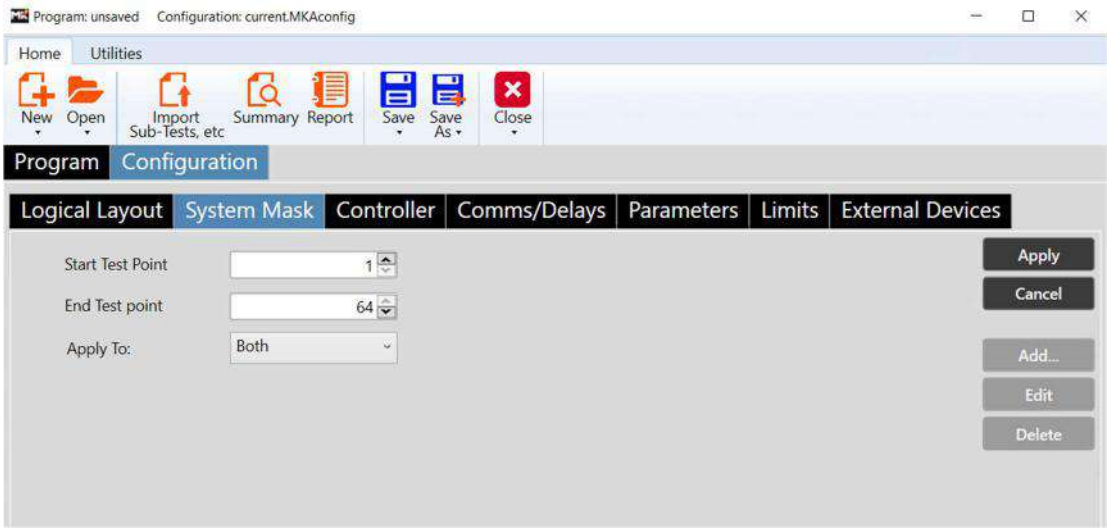
4 Configuration

Configuration features are covered by the Runner manual so please refer to *MKAT Runner Software Manual.docx* for details. The only exception to this is System Mask which is not currently available in the Runner.

4.1 System Mask

A test point or range of test points can be added to the system mask table which means that they become globally masked and will be excluded from all test programs run on the system. Note that masking can also be applied to individual programs as outlined in the *Creating a program* section.

To add an entry, select the *Add* button and enter a Start Test Point and End Test Point. From the *Apply To* drop-down menu select whether the mask should apply to two wire, four wire or both and then select the Apply button to confirm. The mask can be changed using the *Edit* button and removed using the *Delete* button.



In the example below, test point 1 has been globally masked for both 2 and 4 wire tests, test points 2 to 6 have been masked for 2 wire tests and test points 9 and 10 have been masked for 4 wire tests: