

4 Measuring Resistance

PRELIMINARY NOTE: This section applies only to CableEye Models with resistance measurement capabilities. Models M2U-Basic and M2U do not apply for this section.

About this Section

In this section you will learn how to use the CableEye system to correctly measure cable resistance and embedded resistors in a cable. Although, the tester will automatically measure resistance out of the box for you, there are several preferences and options to understand for unusual situations, and the purpose of the section is to show you why, how and when to change these settings.

There are two major parts to this section; the first one covers normal *2-Wire resistance measurement* while the second part is intended for *4-Wire testing*; note that a *CableEye HVX High Voltage* tester with *4-Wire* option is required for the latter.

You will also learn about compensating for trace resistance values when testing with your custom made fixtures.

Follow this index to jump quickly to a topic of interest:

4.1 Resistance Measurement and Thresholds.....	4-2
4.2 Resistance Measurement Capabilities	4-3
4.3 Measuring Resistors Automatically	4-5
4.4 Comparing Two Cables.....	4-6
4.5 Measuring Resistors by Direct Control.....	4-7
4.6 Editing Cable's Resistance.....	4-8
4.7 Resistance Tolerance.....	4-8
4.8 Accuracy	4-9
4.9 Measurement Time.....	4-9
4.10 Multinode Groups - Testing Complex Circuits.....	4-10
4.11 Trace Resistance - Fixture Resistance Compensation.....	4-16
4.12 Causes of Inconsistent Resistance Readings.....	4-19
4.13 4-Wire Kelvin Measurement Option	4-20

4.1 Resistance Measurement and Thresholds

CableEye Models M3U and newer offer a high-speed resistance scanning circuit not found on previous CableEye models. This capability permits you to check the quality of a connection in addition to basic wiring correctness. For most cables, you do not need to measure the actual resistance of each wire. Rather, you would like to confirm that the resistance of a good connection does not exceed a maximum limit, and that the isolation between unconnected wires does not fall below a minimum limit. We refer to these two limits as thresholds.

Whenever you test a cable, the thresholds you have chosen in the Resistance Limits preferences are applied automatically. Wires that fail the limits test appear in the wire list with a resistance value and will cause any comparison made with Match Data to fail unless a corresponding resistance appears in the Match Data wire list.

You may also measure the actual resistance value of one or more resistors that may be embedded in a cable. This will happen automatically whenever a cable is tested if the resistor value falls between the two resistance thresholds.

Checking for Good Connections and NON-Connections

Wire connections in a cable that exceed your maximum resistance threshold will be flagged as defective. High-resistance connections like this may be caused by incomplete crimps, by insufficient mating between the knife-pin and wire conductor in IDC flat cables, or by cold solder joints in soldered connections. These marginal connections may then result in circuit misoperation, intermittent connections, or physical heating in the cable if enough current flows through that connection. In a properly managed cable assembly shop, high resistance connections occur rarely. However, resistance testing is needed to ensure that absolutely no bad cables make their way into a customer's product. You may set CableEye Model M3U to check for an acceptable connection resistance of as little as $0.3\ \Omega$.



In addition to certifying that a wire connection passes a low resistance test, we must also ensure that internal short circuits do not exist. While this seems obvious, we need to define exactly what constitutes a short circuit. To certain sensitive electronic devices, especially in medical applications, a "short circuit" may exist in a cable when as much as a $1\ \text{M}\Omega$ (or higher) resistance path appears between two pins. In contrast, the high output drive capability of present-day CMOS digital circuits can easily maintain the logic '1' minimum voltage across a short circuit of as small as several hundred ohms. Ideally, we would like an infinite resistance between isolated conductors. As a practical matter, though, a lesser amount will be quite adequate for most applications.

We recommend a minimum isolation threshold of $1\ \text{M}\Omega$ for general purposes, although you may set the isolation threshold to be as high as $10\ \text{M}\Omega$ with CableEye model M3U and $5\ \text{M}\Omega$ for HVX models.

4.2 Resistance Measurement Capabilities

Unlike a benchtop multimeter which measures resistance across a single set of probes, CableEye measures the resistance between every possible combination of at least 128 different test points, and must do so very quickly (in less than a second). Consequently, CableEye's measurement characteristics are quite different from a typical multimeter. The following subsections itemize important information about how to set thresholds, how to measure resistors, and what you can and cannot measure.

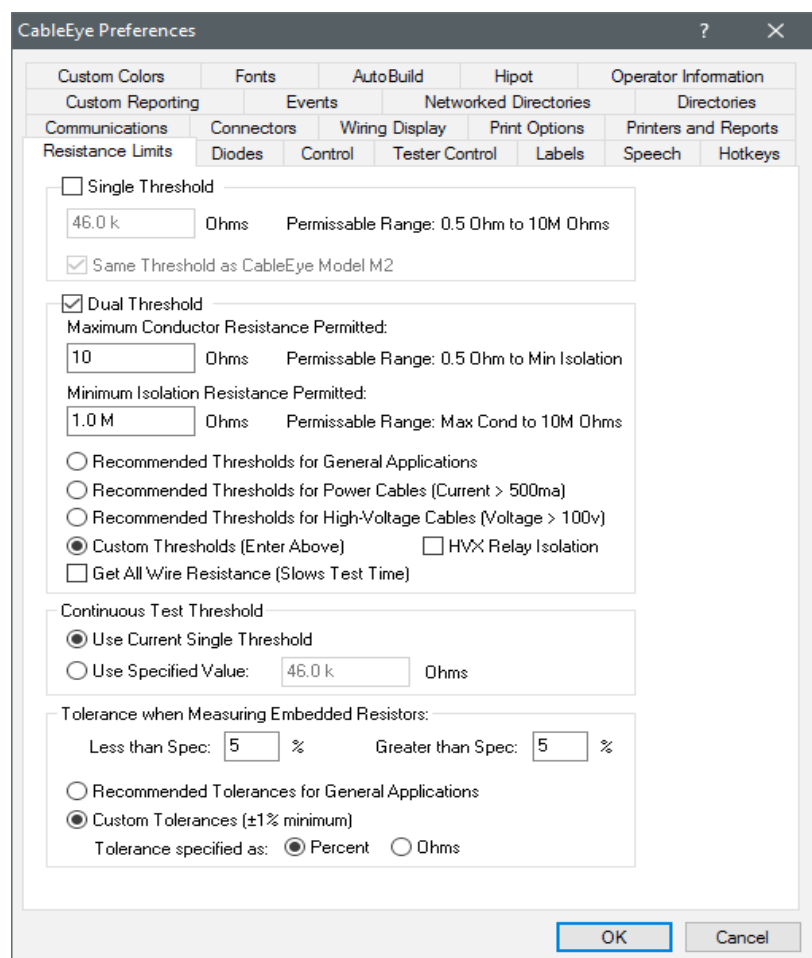
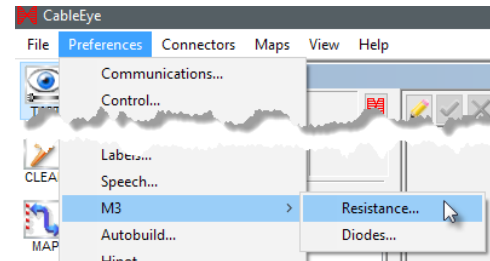
Setting Thresholds

Choose the **M3|Resistance** in the **Preferences** menu, shown on the right, to access CableEye's *resistance control panel* shown below. You can set all the resistance parameters from the resistance control panel.

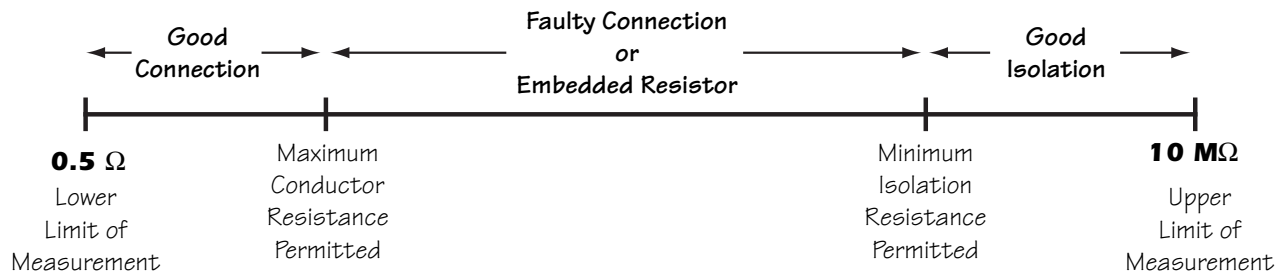
For our purposes, a "threshold" is simply a resistance value that separates decision regions. If you choose a *Single Threshold*, you are dividing the entire range of conductor resistance, from zero ohms (a perfect conductor) to an infinitely high resistance (a perfect insulator) into two regions. Values below the threshold signify an acceptable conductor, and values above it signify acceptable isolation between conductors.

When you use a single threshold, the measurement time will be faster, but you will not be able to detect embedded resistors nor will you be able to test the quality of a connection. The single threshold option was provided for applications where you need to configure model M3U to behave exactly like our simpler model M2U tester, or for special applications.

Example: if you choose a single threshold of 1000 Ω , all connections below that value qualify as *acceptable* wire connections, and all connections above that value are considered "open circuits". Typically, good wire connections have a resistance of less than 10 Ω (often only a fraction of an ohm), and isolated conductors offer a resistance of 1M Ω or greater (typically tens of megohms, depending on humidity).



For most purposes, you would select the *Dual Threshold* option in the Resistance Control Panel. In this case, we divide the entire range of conductor resistance into three regions. As in the single threshold option, the lower region contains all acceptable resistance values for a “good” connection (a wire), and the upper region contains all acceptable resistance values for isolated wires where no connection exists. We also have a *middle* region which represents either questionable connections, or discrete resistors that may be embedded in a cable.



The *Maximum Conductor Resistance Permitted* sets the highest value of resistance acceptable for a “good” connection. For most applications, we recommend a value of 10 Ω, but it could be set to as little as 0.5 Ω in the M3U model or 0.1 Ω for the M3UH and HVX models.

If you set this value for less than 5 Ω, the measurement time may increase by a factor of two or more due to the increased sensitivity you are asking for, and you may pick up marginally higher resistance values that will have no effect on the function of the cable. Use the smallest threshold of 0.5 Ω or less only when you test power cables or any cables that are expected to carry significant current (20 gauge or thicker wire).

The *Minimum Isolation Resistance Permitted* sets the lowest value of isolation resistance acceptable for unconnected wires. For most applications, we recommend a value of 1 MΩ, but it could be set to as high as 10 MΩ.

Note that the contact resistance of dry human skin is about 2 or 3 MΩ. To minimize the probability of false negatives during production testing, we recommend setting the isolation threshold above 1 MΩ *only* when you are testing cables that will be used in a high voltage or extremely low current application. It is interesting to observe that you would need to apply 1000 V across a 1 MΩ resistor to create a current flow of 1 mA.

4.3 Measuring Resistors Automatically

CableEye measures embedded resistors automatically when the resistor's value falls *between* the high and low thresholds. When you save a cable in the database, the resistance values associated with any embedded resistors are saved along with the wire list and other cable information. Thus, embedded resistors become one of the characteristics of the stored cable and must be present in a test cable to successfully match against the saved data.

When you compare two cables that contain embedded resistors, the resistance values measured in the unit under test must fall within the *tolerances* you set at the bottom of the *Resistance Limits* preferences. Missing resistors, misconnected resistors, or resistors that are present but fall outside of the tolerance range, will flag an error when the cables are compared.

Threshold Violations

If any threshold violations are found, they will appear in the *Value* column on the netlist, as shown in the right. The wiring diagram corresponding to this Net List appears in the image below. Note that a threshold violation may be caused either by a faulty connection or by an embedded resistor. CableEye cannot know how to interpret the Test Data unless you have loaded Match Data. A resistor present in the Match Data netlist requires that a corresponding resistor appear in the Test Data netlist and fall within the resistance tolerance specified in the Resistance Control Panel in order that a Pass condition be achieved.

Line	DB9 Male J3	DB9 Female B2-2	Value
1	SH	SH	
2	1	1	
3	2	2	2.70 kΩ
4	3	3	
5	+4	-4	⚡
6	-5	+5	⚡
7	6	6	
8	7	7	
9	8	8	
10	9	9	

Set resistance thresholds and tolerances using this Resistance Limits tab found in the CableEye's Preferences Menu.



This wiring schematic corresponds to the netlist seen above and shows either a threshold violation or an embedded resistor in the cable, as well as two diodes.

4.4 Comparing Two Cables

Match data need not be loaded to detect simple threshold violations. As seen in the previous figures, you may refer to the wire list to see any violations detected. However, in order to generate a *Pass* or *Fail* condition and cause the appropriate indicator lamp to light, Match Data must be present. The Match Data you load specifies what constitutes a “good” cable. To achieve a Pass condition, the following requirements must be met after you measure a cable:

- 1 - The wiring topology must match perfectly including presence or absence of shield.
- 2 - The connector types and genders must be identical
- 3 - There must be no resistance threshold violations.
- 4 - Any embedded resistors or other components present in the Match Data must also be present in the Test Data and found to be within the tolerance range specified in *Resistance Preferences* panel.



Reporting Results

Note that you may specify two or more *different* low thresholds in a cable if desired. This might be necessary if you have a combination of power and signal lines in which the current-carrying conductors have a more stringent low resistance requirement, or if you are measuring a multi-headed cable or wiring harness in which one of the legs involves much longer run of wire than the other.

As an example, assume you need to set a threshold for signal lines to 20 Ω , and for two power conductors (heavier gauge wire) in the same cable for 0.5 Ω . First, **set the low threshold in Resistance Preferences** to the **highest value, 20 Ω** in this case. Then, manually **enter resistance values of 0.5 Ω** using the Netlist Editor (as detailed in Section 3) for those conductors which are to have lower thresholds.

The Test Data shown in the right has some threshold violations. In this case, we had set the low threshold at 10 Ω and the resistance tolerance at 20%. Because the connections 4 to 4 and 5 to 5 do not show resistors defined in the Match Data, they must be faulty connections and thus appear in the *Differences List* as shown in the next page. The connection 1 to 1 does have a corresponding resistor in the Match Data. Although the resistances are not exactly equal, they are within the 20% tolerance we specified, so no error is flagged for the connection 1 to 1.

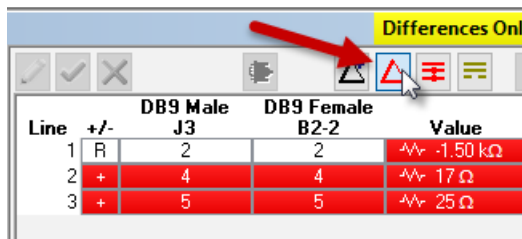
Finally, the connection 2 to 2 has a corresponding resistor in the Match Data, but the value measured in Test Data exceeds the tolerance and thus is reported as an error. Note that in this case we report the difference between the measured resistance and the Match Data value in the *Differences List*.

Line	DB9 Male J3	DB9 Female B2-2	Value
1	SH	SH	
2	1	1	2.14 k Ω
3	2	2	2.70 k Ω
4	3	3	
5	4	4	17 Ω
6	5	5	25 Ω
7	6	6	
8	7	7	
9	8	8	
10	9	9	

Line	DB9 Male J3	DB9 Female B2-2	Value
1	SH	SH	
2	1	1	2.00 k Ω
3	2	2	4.20 k Ω
4	3	3	
5	4	4	
6	5	5	
7	6	6	
8	7	7	
9	8	8	
10	9	9	

Here you see Test Data with threshold violations reported, while the Match Data only has two defined resistors, the Test Data reports four, and the test is flagged as Fail.

A Pass condition exists only when no errors appear in the Differences List. At that time, the PASS lamp will light on the tester and a large green box with a yellow checkmark will appear on the computer's screen. If the Differences List show any entries at all, a Fail condition will be asserted, causing the FAIL lamp to light and a large red box with a black X to appear on the computer's screen.



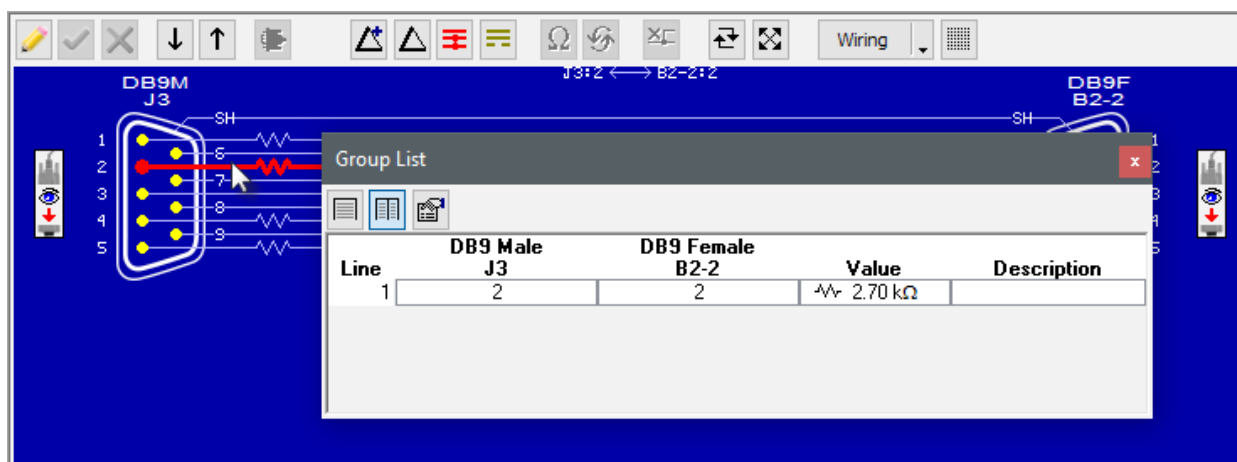
Line	+/-	DB9 Male J3	DB9 Female B2-2	Value
1	R	2	2	~1.50 kΩ
2	+	4	4	~17 Ω
3	+	5	5	~25 Ω

Clicking on the View Differences button will highlight all resistance violations as well as normal cable faults, like opens and shorts.

4.5 Measuring Resistors by Direct Control

Occasionally, you may need to measure a resistor whose value falls within either the good connection or good isolation resistance range (for example, measuring a 5 Ω resistor with the low threshold set at 10 Ω). The CableEye software will normally *not* report this resistors when you use Test Cable. However, you may accomplish this in one of four other ways:

1. If you define embedded resistors in the Match Data, the tester will measure and report the resistance value between the points where the resistance is expected. If these targeted measurements fail to detect a resistor, or detect one that is outside of the tolerance you specified, an error will be flagged.
2. To manually test the resistance of any specific connection, **display the schematic wiring diagram** and use the ↑ or ↓ keys on the keyboard or click in a connection to highlight it. Once the desired connection is highlighted, a **Group Netlist** window pops up showing the resistor's value. **Click the Measure Resistance** button ("Ω" symbol) in the menu bar to measure the resistance of that connection once, or the **Measure Resistance Continuously** button to measure constantly; pressing the Measure Resistance Continuously button again stops the measuring cycle.

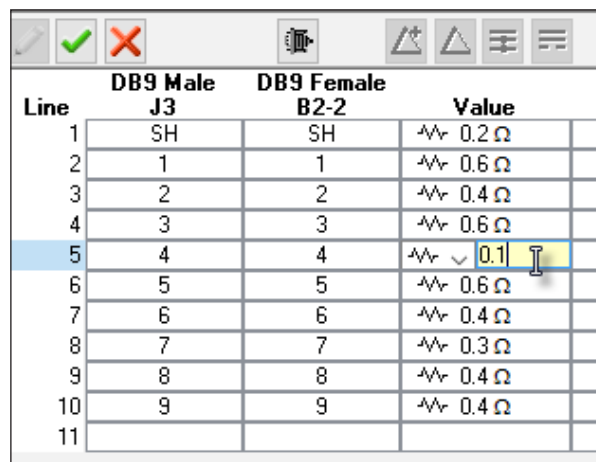


3. You can use the *test probe* to test the resistance on *flying unconnected leads*. Go back to Section 3, page 3-29 for a detailed explanation on how to do this.
4. Use the "TEST RESISTANCE" Macro instruction during an automatic test sequence. Learn more about this method in Section 6, Automatic Testing.

4.6 Editing Cable's Resistance

Wiring Data is normally acquired by measuring a cable or by loading a previously measured cable from the database. You may also enter wiring data from the keyboard, or edit existing wiring data, by manually typing pin-to-pin connections and resistance values. You may add connections, delete connections, define resistance values, insert diodes, and change the connector styles. Refer to section 3, page 3-22 for general information on Netlist Editing.

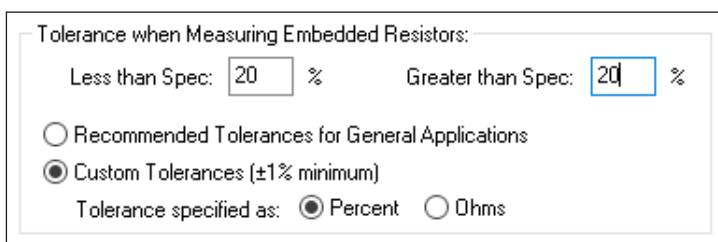
CableEye allows you to enter or edit *resistance values* that appear in a netlist. To do so, simply double-click the cell in the *Value* column, and type in the desired values (see right). You may use "k" for kilohms and "M" for megohms if desired. You do not need to type the " Ω " symbol. Examples of allowable entries: 0.4, 2300, 2.3k, 2.3 k, 5000000, 5 m, 5M. **Click the green Checkmark** to accept the changes and **Save** the cable when ready.



Line	DB9 Male J3	DB9 Female B2-2	Value
1	SH	SH	$\sim 0.2 \Omega$
2	1	1	$\sim 0.6 \Omega$
3	2	2	$\sim 0.4 \Omega$
4	3	3	$\sim 0.6 \Omega$
5	4	4	~ 0.1
6	5	5	$\sim 0.6 \Omega$
7	6	6	$\sim 0.4 \Omega$
8	7	7	$\sim 0.3 \Omega$
9	8	8	$\sim 0.4 \Omega$
10	9	9	$\sim 0.4 \Omega$
11			

4.7 Resistance Tolerance

CableEye allows you to set either *symmetric* or *Asymmetric Individual Tolerances* for each conductor if required. As explained before, a global tolerance value can be set on the **Resistance Preferences Panel** as shown below, but there are cases where you want an specific connection or connections to be compared with a different tolerance.



Tolerance when Measuring Embedded Resistors:

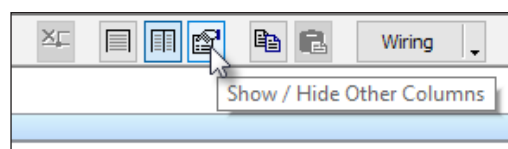
Less than Spec: % Greater than Spec: %

☐ Recommended Tolerances for General Applications

☒ Custom Tolerances ($\pm 1\%$ minimum)

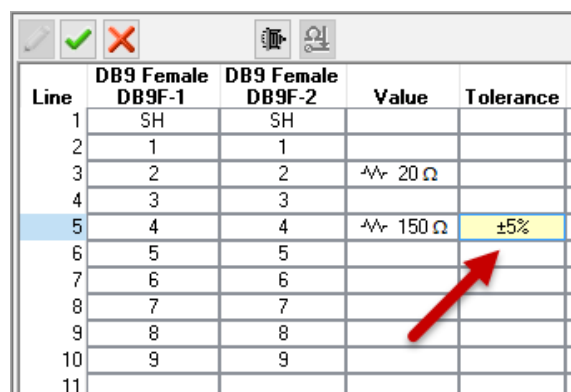
Tolerance specified as: ☒ Percent ☐ Ohms

To accomplish this, you need to **edit the Match Data netlist** by adding the new tolerance values for the desired connections. The new values are typed in the **Tolerance** column; note that this column might not be visible by default and that you will need to bring it up by clicking in the **Show/Hide Other Columns** button and select **Tolerance**.



Double click on the cell under the **Tolerance** column that you want to edit, and type in the tolerance value. The value can be inserted as absolute ohms or percentage. In the example on the right, the tolerance between pins 4 and 4 was reduced to $\pm 5\%$. Simply **type in 5** and then **enter or tab**. You'll notice that it automatically added " \pm " indicating that the 5% tolerance is applied symmetrically.

If you type in the letter "o" after the value, you force the tolerance to absolute ohms. For example **type in 5o** and you will get $\pm 5.0 \Omega$. You type in asymmetrical tolerances like this: +10 -5.



Line	DB9 Female DB9F-1	DB9 Female DB9F-2	Value	Tolerance
1	SH	SH		
2	1	1		
3	2	2	$\sim 20 \Omega$	
4	3	3		
5	4	4	$\sim 150 \Omega$	$\pm 5\%$
6	5	5		
7	6	6		
8	7	7		
9	8	8		
10	9	9		
11				

4.8 Accuracy

Unlike a digital multimeter, CableEye's resistance measurements vary in accuracy over the measurement range. This means that the most accurate measurements occur in the middle of the range where you might expect to find embedded resistors. Fewer digits of precision are offered at the extremes where we are simply checking against performance thresholds. This characteristic results from designing a resistance thresholding circuit that operates at the highest possible speed. The table below summarizes the ranges and accuracies offered.

Resistance Range	M3U	M3UH & HVX	Numeric Examples
0.3 Ω to 0.9 Ω	$\pm 0.2 \Omega$	$\pm 0.1 \Omega$	0.7 Ω ,
1.0 Ω to 9.9 Ω	$\pm 1.0 \Omega$	$\pm 1.0 \Omega$	3 Ω
10 Ω to 99 Ω	$\pm 10\%$	$\pm 10\%$	83 Ω
100 Ω to 999 k Ω	$\pm 1\%$	$\pm 1\%$	586 Ω , 1.39 k Ω
1 M Ω to 2.9 M Ω	$\pm 10\%$	$\pm 10\%$	1.5 M Ω , 2.1 M Ω
3 M Ω to 4.9 M Ω	$\pm 20\%$	$\pm 20\%$	4 M Ω
5.0 M Ω to 10 M Ω	$\pm 20\%$	N/A	9 M Ω

These accuracy values apply when you measure embedded resistors. When you set thresholds, we have added an offset to ensure that most of the inaccuracies occur on the inside of the threshold. That is, if you set the maximum conductor resistance threshold to 5 Ω , the software will internally set the threshold slightly lower than that to guarantee that nothing greater than 5 Ω will be allowed.

Remember that when you test for resistance values less than 100 Ω or greater than 1 M Ω , the measurement accuracy decreases to $\pm 20\%$, and at ranges below 1 Ω may have an even higher variation. In the interest of scanning speed, the method of testing at these extreme ranges was intended primarily for thresholding and not for precision measurements.

4.9 Measurement Time

Generally, CableEye needs about 0.5 s to complete a measurement with 152 active test points. If you need only 128 test points (no expander cable attached to the CB boards) or 64 test points (only the Bank 1 in use), the scanning time will be faster. Under the following circumstances, however, the scanning time may increase:

1 – If you connect one or more expansion modules, the test time will increase. For 256 test points, the measurement time will typically be one second. For 1024 test points, the measurement time will be about 6.2 s. For 2048 test points will be about 20 s.

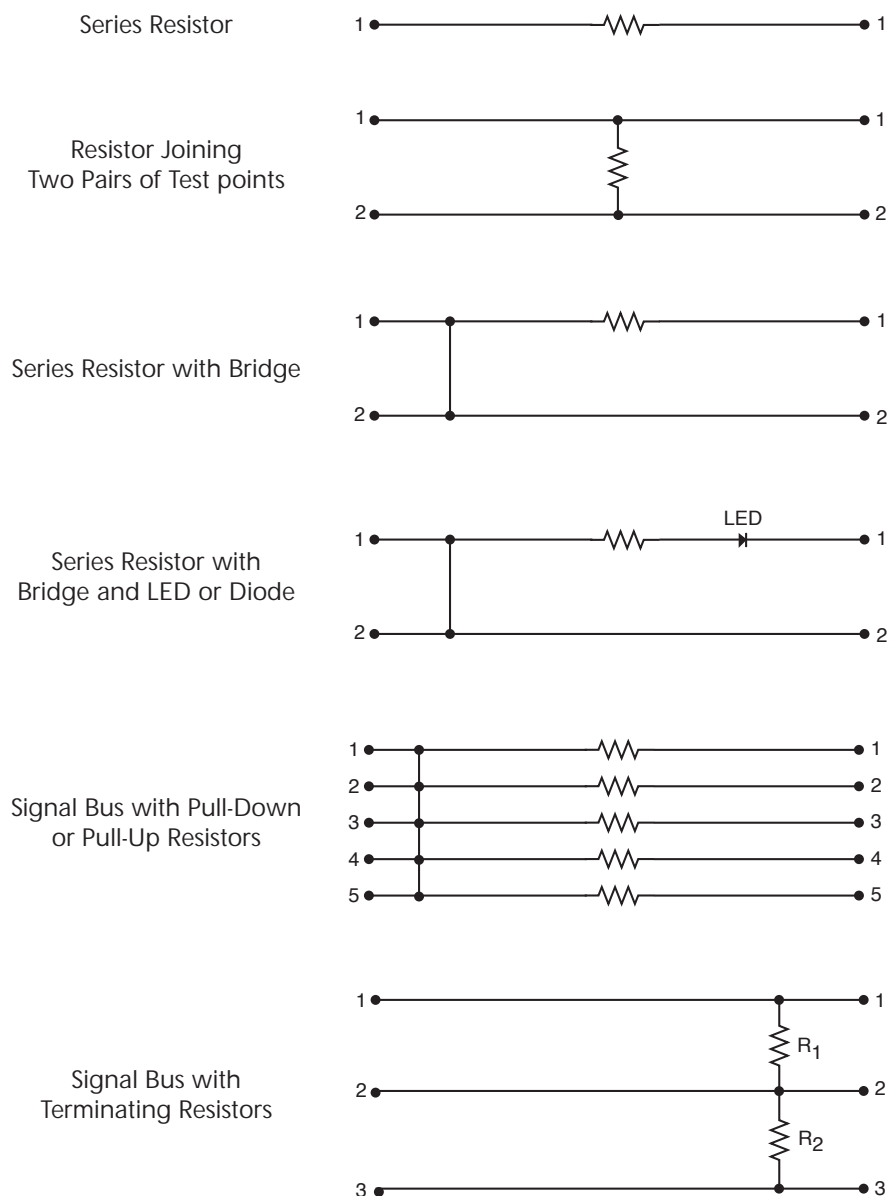
2 – If your cable contains embedded resistors, the measurement time will increase slightly in proportion to the number of resistors present.

3 – If the cable has many threshold violations, the software will test each individually for a resistance value. Doing so will take time and may result in a long list of resistors in the net list. For example, if all conductors in a 64-conductor flat cable had threshold violations, the resistance of each conductor would be separately measured, requiring about 1.1 s to complete rather than the usual 0.5 s when no threshold violations occur.

4.10 Multinode Groups - Testing Complex Circuits

In addition to the simple case of a series resistor embedded in a cable, you may find other situations in which one or more resistors and wires are connected in a network. We refer to such networks as multinode groups. CableEye can make resistance measurements on most configurations, while in some complex situations, the interconnections between components may be determined, but without resistance measurements. The figure below shows examples of networks for which the embedded resistors can be accurately measured. The first circuit shows a simple series resistor. The others are multinode groups that include at least one resistor, and involve three or more test points.

4.10.1 Multinode Resistor Connections



Simple resistor networks in which the resistance can be accurately measured

When CableEye detects a multinode group, the wire list shown on the screen may appear different from the physical layout of your circuit. This occurs because several different circuit topologies could give rise to electrically equivalent networks. In the absence of Match Data, measurements of multinode groups containing resistors may show resistors in incorrect locations, and their values may be inaccurate. The next section describes how to define the Match Data in advance of testing a cable to ensure that CableEye understands exactly how you have connected multinode groups and where the resistors are located. Once the Match Data reflects the actual resistor positions in the network, consistent and accurate measurements of their values will be produced.

4.10.2 CableEye's automatic representation of multinode networks

In cases where it is not possible to show an exact representation of the network, CableEye uses the following methods for reporting it:

1 – The wire list generally shows a network in what we refer to as “Normal” form, even for simple displayable networks. In Normal form, the connections involved in the network are sorted numerically according to the left-sided pins, and then reduced to the simplest possible arrangement of connections. While the Normal form may not necessarily reflect the physical connections of the network, it is electrically identical and ensures that a complex wirelist can be easily compared to other wirelists without generating false comparison errors.

Note that when resistors are involved in a multi-node network and Match Data is not present to specify the network topology, the Normal form may show resistors in incorrect locations and with incorrect values when you execute Test Cable. This results simply because the Normal form of the network topology may be one of many equivalent forms that correspond to the measurements Test Cable produced.

Consider the two sets of multinode networks shown below. Each three circuits have different physical connections but they are electrically identical.

	<p style="text-align: center;">CableEye's Normal Form</p> <table border="1"> <thead> <tr> <th>Line</th> <th>Trans Board B1</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1, 2, 3, 4</td> </tr> </tbody> </table>	Line	Trans Board B1	1	1, 2, 3, 4	<p style="text-align: center;">Possible Equivalent Forms</p> <table border="1"> <thead> <tr> <th>Line</th> <th>Trans Board B1</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1, 2</td> </tr> <tr> <td>2</td> <td>1, 3</td> </tr> <tr> <td>3</td> <td>1, 4</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Line</th> <th>Trans Board B1</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1, 2</td> </tr> <tr> <td>2</td> <td>2, 3</td> </tr> <tr> <td>3</td> <td>3, 4</td> </tr> </tbody> </table>	Line	Trans Board B1	1	1, 2	2	1, 3	3	1, 4	Line	Trans Board B1	1	1, 2	2	2, 3	3	3, 4																
Line	Trans Board B1																																					
1	1, 2, 3, 4																																					
Line	Trans Board B1																																					
1	1, 2																																					
2	1, 3																																					
3	1, 4																																					
Line	Trans Board B1																																					
1	1, 2																																					
2	2, 3																																					
3	3, 4																																					
<p>These three electrically identical networks all contain a 100 Ω resistor in slightly different positions.</p>	<p style="text-align: center;">CableEye's Normal Form</p> <table border="1"> <thead> <tr> <th>Line</th> <th>Trans Board B1</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1, 2</td> <td></td> </tr> <tr> <td>2</td> <td>1, 3</td> <td>$\sim 200 \Omega$</td> </tr> <tr> <td>3</td> <td>1, 4</td> <td>$\sim 200 \Omega$</td> </tr> </tbody> </table>	Line	Trans Board B1	Value	1	1, 2		2	1, 3	$\sim 200 \Omega$	3	1, 4	$\sim 200 \Omega$	<p style="text-align: center;">Possible Equivalent Forms</p> <table border="1"> <thead> <tr> <th>Line</th> <th>Trans Board B1</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1, 2</td> <td></td> </tr> <tr> <td>2</td> <td>2, 3</td> <td>$\sim 100 \Omega$</td> </tr> <tr> <td>3</td> <td>3, 4</td> <td></td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Line</th> <th>Trans Board B1</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1, 2</td> <td></td> </tr> <tr> <td>2</td> <td>2, 4</td> <td>$\sim 100 \Omega$</td> </tr> <tr> <td>3</td> <td>3, 4</td> <td></td> </tr> </tbody> </table>	Line	Trans Board B1	Value	1	1, 2		2	2, 3	$\sim 100 \Omega$	3	3, 4		Line	Trans Board B1	Value	1	1, 2		2	2, 4	$\sim 100 \Omega$	3	3, 4	
Line	Trans Board B1	Value																																				
1	1, 2																																					
2	1, 3	$\sim 200 \Omega$																																				
3	1, 4	$\sim 200 \Omega$																																				
Line	Trans Board B1	Value																																				
1	1, 2																																					
2	2, 3	$\sim 100 \Omega$																																				
3	3, 4																																					
Line	Trans Board B1	Value																																				
1	1, 2																																					
2	2, 4	$\sim 100 \Omega$																																				
3	3, 4																																					

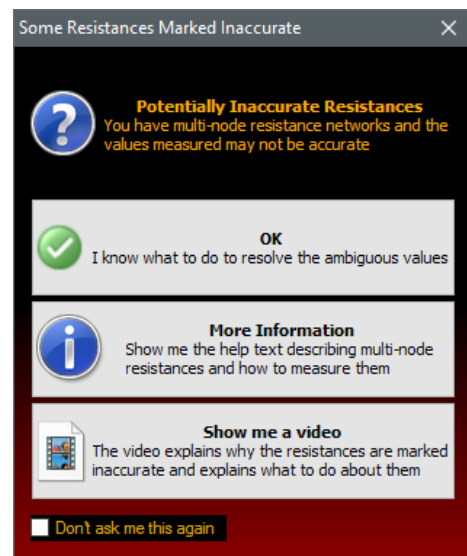
2 – When CableEye finds a multinode group in a cable, the wiring schematic displayed for that group shows dashed lines connecting the involved test points without actually drawing the circuit. To provide details of any connection denoted by a dashed line, simply highlight the dashed line using ↑ or ↓ Arrow keys and a pop-up window will appear showing the details of the network.



3 – If the network includes diodes or a combination of resistors and diodes, the wiring schematic also shows dashed lines connecting the involved test points without showing the component details.

Note that the first time that you test a cable that contains multi-node resistor networks, you will get the message shown on the right. You can click **OK** and the tester will display the results. You can also click **More Information** to open the CableEye help menu which displays the same information shown in this section.

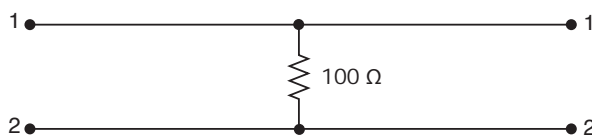
Note: This message will not appear when Match Data defines the position and values of the resistors in the network.



4.10.3 Multinode Network Examples

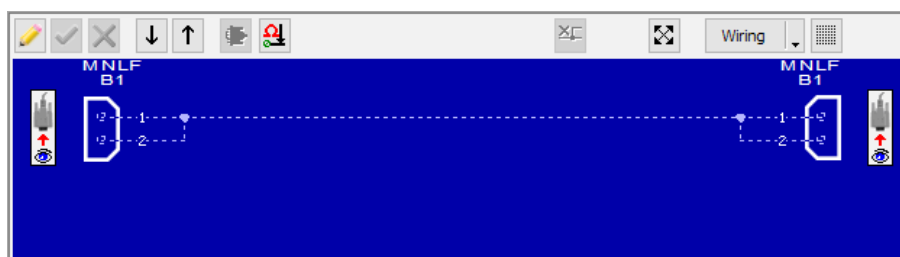
The following examples illustrate how CableEye displays various multinode groups containing resistors:

Example 1: This simple network shows a resistor connecting two parallel conductors. It is a multinode group because more than two test points are interconnected. In the netlist, we distinguish the left and right side test points with the prefix “L” for Left and “R” for right:

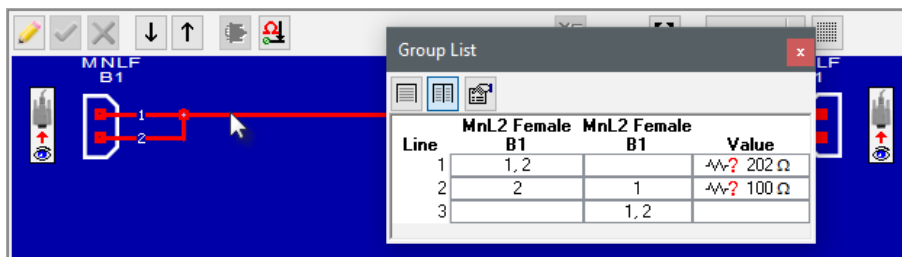


L-1 to R-1
L-1 to L-2, 100 Ω
L-2 to R-2

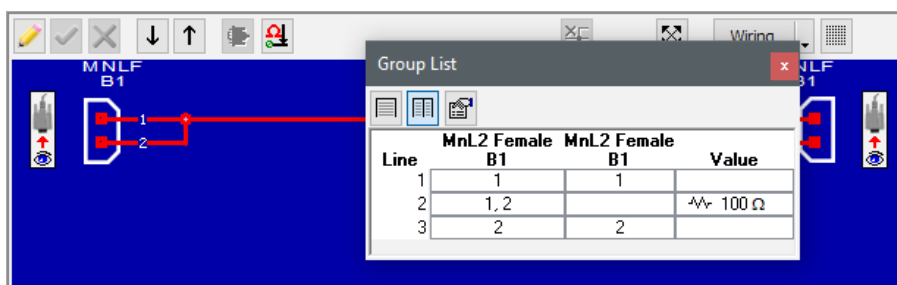
CableEye produces this wiring schematic for the multinode group (the connectors are arbitrary). Dashed lines show which test points are interconnected without drawing the components.



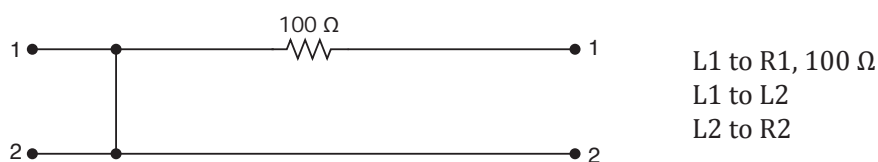
When you highlight the dashed line, a “Group Netlist” window appears showing the connections in Normal Form. Because Match Data was not present to define the topology when Test Cable was executed, the Group Netlist shows the Normal Form netlist.



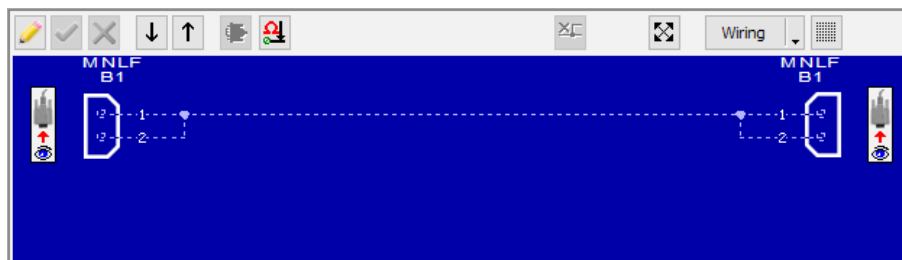
As you can see, this does not reflect the actual circuit topology. Two parallel resistors appear of about twice the value of the actual single resistor; when these two resistors are electrically combined into one, a single resistor of the correct value would be shown. The red question mark next to the resistor symbol indicates that Match Data for this group was not found. With proper Match Data entered, the correct Group Netlist would be shown, as below.



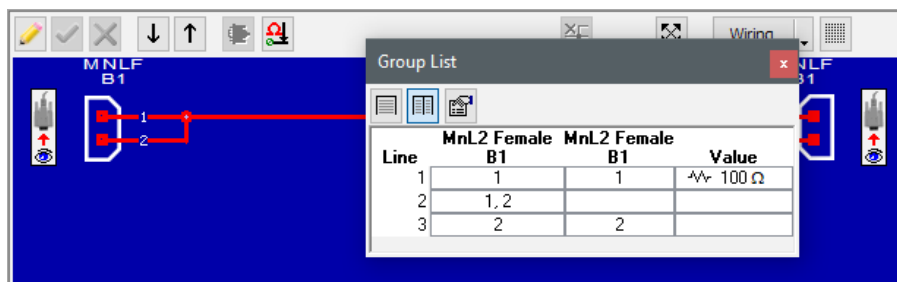
Example 2: This network shows a connection between a conductor with an embedded resistor and a conductor without any resistors.



Because the same test points are connected in this example as were connected in the last example, the dashed line drawing will be the same.

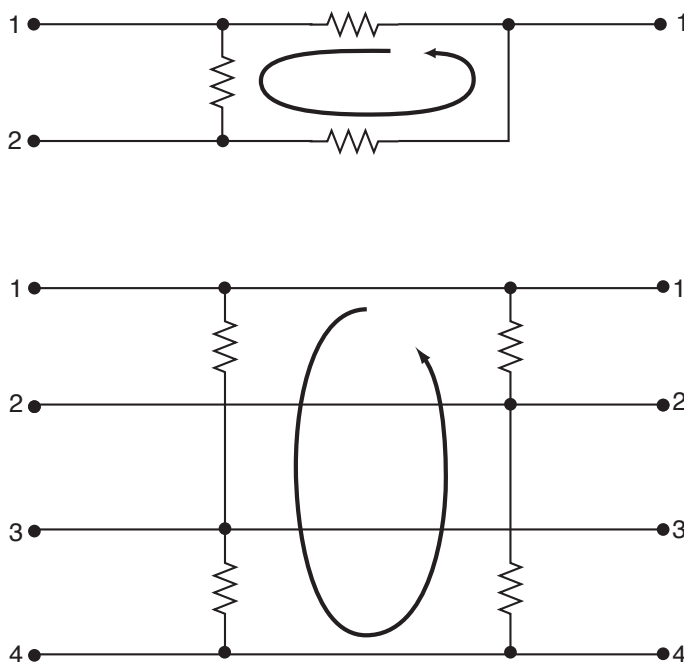


In this case, when the dashed line network is highlighted, a different Group Netlist provides details of the connections. A red question mark in the netlist shows that Match Data is not present. However, because the Normal Form happens to be the correct topology, the netlist is correct as it stands. Were proper Match Data loaded, you would see the same netlist but the red question mark would disappear.



4.10.4 Networks that can NOT be accurately measured

The figures below show examples of networks that cannot be accurately measured. These circuits contain loops, as shown. The software would require knowledge of at least one of the resistor values in advance in order that the other values be computable based on the measurements. If you measure an arrangement like this, resistor values will be shown and successive measurements will be consistent, but the network topology and values will be incorrect. If you attempt to enter a Match Data netlist that contains loops, the software will detect this and produce a warning message "The netlist you have specified contains circular connections ..."

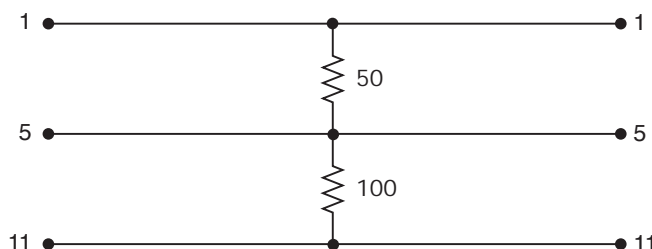


These networks of multiple resistors contain circular connections (loops). As a result, accurate resistance measurements are not possible.

4.10.5 Entering Multinode Connections

To ensure that resistors in multinode groups are placed correctly and measured accurately when you execute Test Cable, you must predefine the connections in Match Data. This simple process will take only a few moments, and once done, may be stored in the database for future reference. The example below outlines the process.

Example 3: This drawing shows a multinode network involving six test points and five branches.



Using Test Cable to directly measure this network without Match Data present produces the Normal Form netlist shown on the right.

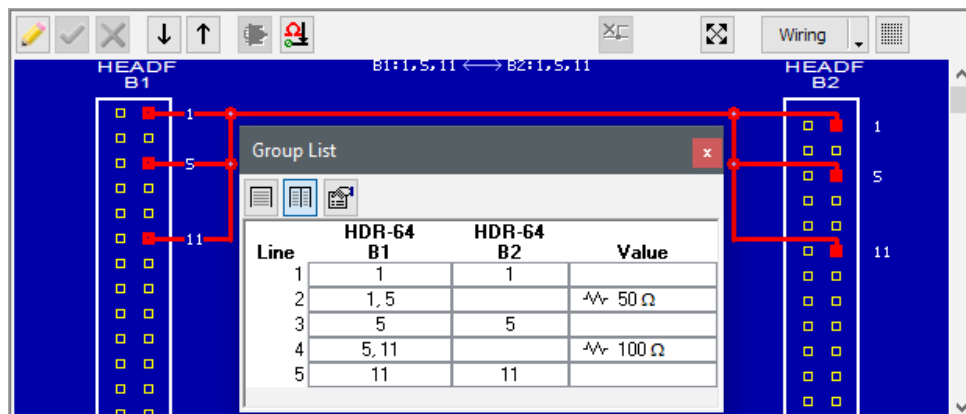
	HDR-64	HDR-64	Value
1	1, 5	2	$\sim 99 \Omega$
2	5, 11		$\sim 49 \Omega$
3	11	1	
4		1, 5	
5		5, 11	$\sim 197 \Omega$

Clearly, the position of the resistors and their values in the Normal form netlist do not accurately reflect the actual layout of the network. To correct this, first use **Learn Cable** to read the Normal Form measurements directly into Match Data. Then, **click the yellow pencil** button to begin editing.

Enter the actual network connections and resistance values, eliminating any spurious resistors shown in the original netlist. Press the green checkmark button when finished. Then, save the edited netlist in the database.

Line	HDR-64 B1	HDR-64 B2	Value
1	1	1	
2	1, 5		$\sim 50 \Omega$
3	5	5	
4	5, 11		$\sim 100 \Omega$
5	11	11	
6			

Finally, with the correct Match Data loaded, execute Test Cable again to produce the proper netlist with accurate resistance values.



4.11 Trace Resistance - Fixture Resistance Compensation

When you use CableEye to test small resistances - less than a few ohms - the resistance of the test fixture or jig itself can become a significant part of the measured value. For the purposes of this discussion, we refer to this resistance as **Trace Resistance**. Contributors to this resistance can be several and include:

- Connector interface between the tester and the CB Board.
- Conductive traces on the CB Board itself.
- Resistance within the mating connector.
- Resistance between the wiring of the fixture of jig.
- Soldering quality.

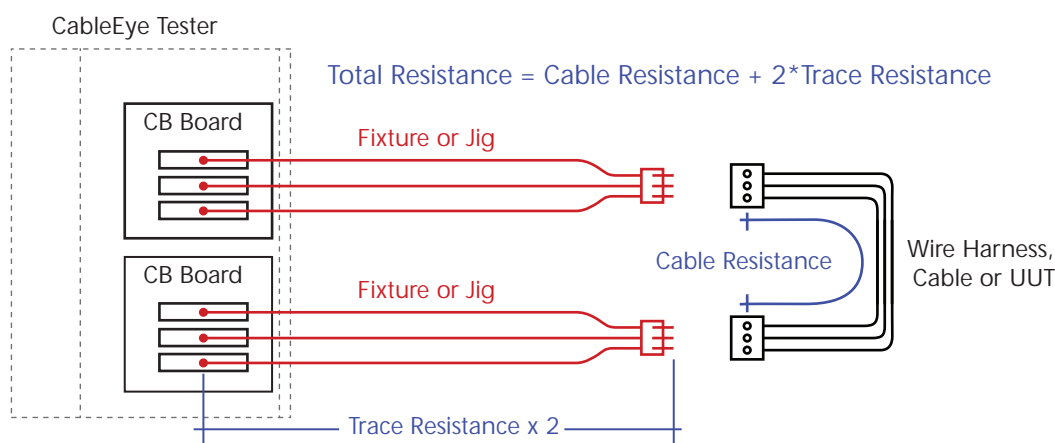


Image shows a typical custom fixture used to test a wire harness or cable. You can easily tell that the resistance in the fixture can be significant compared to the resistance in the cable or harness. CableEye's Trace Resistance allows us to compensate for this measurement.

Since normally these resistances are fixed (they do not change from test to test) ideally you would like to *subtract them* off from the measured value to get the *actual resistance* of the wiring under test. CableEye provides you three ways to do this:

1. Specify manually a value for the Trace Resistance property for each wire and resistance in the Match Data. When the corresponding wire or resistor is acquired and measured as the Test Data, CableEye subtracts of the Trace Resistance value before reporting the resistance.
2. If the Fixture ends have mating connectors that you can plug together (removing the cable under test), you can **Learn for Trace Resistance** and the measured values will be automatically entered in the netlist property.
3. Using the Fixture Editor (PinMap) automatically measure and add the Trace Resistance value using the test probe. In the same way, this Trace Resistance is subtracted before reporting the actual resistance value. For details in using this method, read Section 9.7, page 9-20 of this manual.
4. Employ 4-Wire kelvin measurements. For more information read section 4.13.

The simple rule of thumb is: If you can accurately measure trace resistances in your fixture map using the probe, do it that way. Otherwise, add trace resistances manually to your Match Data.

4.11.1 Adding Trace Resistance to your Match Data

You can assign Trace Resistance property values to each wire and resistor in your Match Data. CableEye subtracts off the trace resistance from the measured resistance before posting the actual value to the Test Data. *Note that if the trace resistance is greater than the measured resistance, a negative resistance will be shown in the Test Data.*

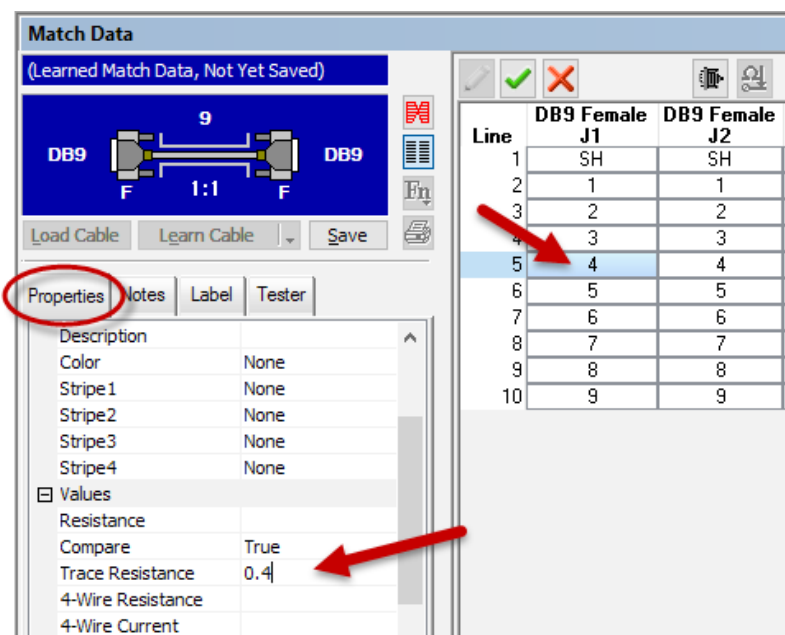
If you have the Trace Resistances values that you want to assign to each component, you can add them directly to your match data using either the Properties window or using the netlist editor.

Using the Properties Window to Add Trace Resistances

In the **Netlist** editor, **click** on the connection to which you wish to add the Trace Resistance value. **Select** the **Properties** tab and enter the Trace Resistance in the Values section, as shown in the right:

In this example 0.4 Ω was added as trace resistance. When this cable is tested, the CableEye will measure the resistance in this connection and subtract 0.4 Ω from it to report the final resistance value.

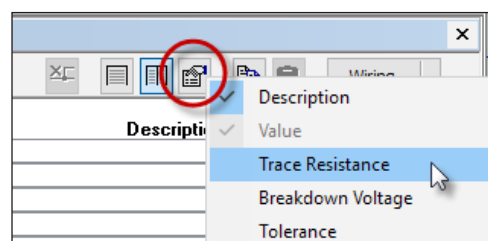
Don't forget to **Click Save** when you are done entering Trace Resistances.



Using the Netlist Editor to Add Trace Resistances

Ensure that the Trace Resistance column is displayed in the netlist. If it's not, **click** on the **Show Columns** button and select **Trace Resistance** as shown in the right.

You can then enter the Trace Resistance to each wire and resistor for which the value applies as shown below. **Click Save** when you are done updating.

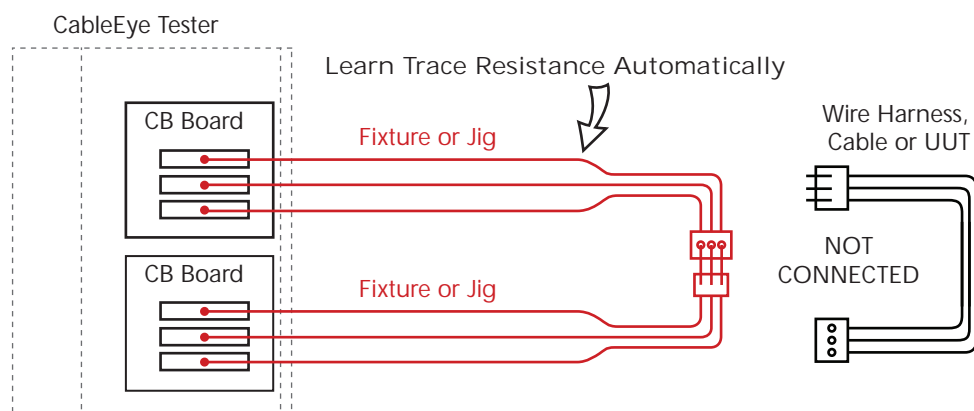


Line	DB9 Female J1	DB9 Female J2	Value	Trace
1	1	1	2.7 Ω	0.6 Ω
2	2	2	2.7 Ω	0.4 Ω
3	3	3	2.7 Ω	0.5 Ω
4	4	4	2.7 Ω	0.4 Ω
5	5	5	2.7 Ω	0.6 Ω
6	6	6	2.7 Ω	
7	7	7	2.7 Ω	
8	8	8	2.7 Ω	
9	9	9	2.7 Ω	

4.11.2 Automatically Acquiring Trace Resistance Values

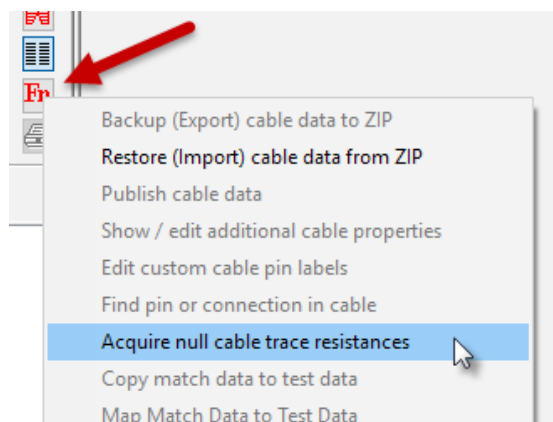
If your fixture setup has mating connectors that you can plug together (removing the cable under test), leaving only connected the section that contains the trace resistance, you can then easily *Learn* the Trace Resistance to enter the measured values automatically into the netlist property.

The following diagram explains when this method can be used:



Once your custom fixture is properly connected at its ends, you can automatically learn the trace resistance:

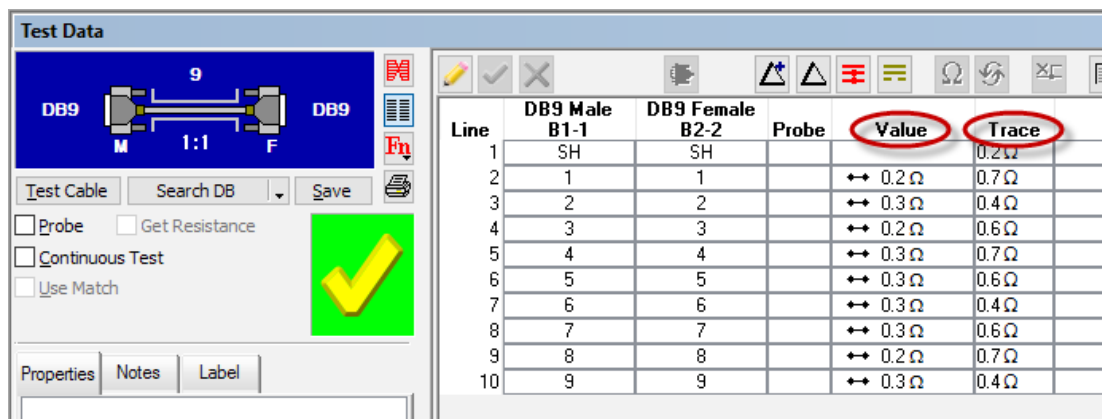
1. In the **Match Data** panel click on the **Special Functions** button and select the **Acquire null cable trace resistances** item:



The **Match Data Netlist** then contains cable data with the measured null cable resistances read into the wire Trace Resistance values:

Line	DB9 Male B1-1	DB9 Female B2-2	Value	Trace
1	SH	SH		0.2 Ω
2	1	1		0.7 Ω
3	2	2		0.4 Ω
4	3	3		0.6 Ω
5	4	4		0.7 Ω
6	5	5		0.6 Ω
7	6	6		0.4 Ω
8	7	7		0.6 Ω
9	8	8		0.7 Ω
10	9	9		0.4 Ω

2. Attach the Cable, Harness or UUT to the tester.
3. Press the **Test Cable** button in the Test Data panel. The Test Data cable now contains the nominal resistance values for your wiring under test and the corresponding Trace Resistance values that CableEye will use to convert measured values to actual values:



4. Press the **Save** button in the Test Data panel to save this as a new model cable. When you use this setup to test wiring, load the saved cable (which now includes trace resistance data) as Match Data to get accurate actual resistance values for your wiring under test.

4.12 Causes of Inconsistent Resistance Readings

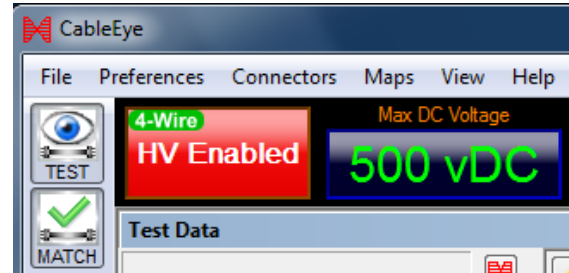
You may encounter situations where the resistance reading of a connection varies slightly from one measurement to the next, particularly when the resistance is below 100 Ω or above 100 KΩ. This results from increased amplification needed at the extreme ends of the test range which adds noise to the measurement. By averaging several readings, you will obtain a more accurate result.

Note that faulty connections caused by cold solder joints, bad crimps, or broken wires may exhibit nonlinear resistance characteristics. Because such connections do not behave like carbon resistors, extremely minor changes in position, vibration, voltage, or temperature changes may cause highly variable readings. Use CableEye's *Continuous Test* function, described in section 3.4, to flex wires while you continuously send test signals through the cable. Variations in resistance that exceed the low threshold will flag an error.

4.13 4-Wire Kelvin Measurement Option

4.13.1 Overview

This section describes how to set up and use the 4-Wire Measurement function on CableEye HVX testers, an *optional* function installed on your tester only if ordered. Refer to the “HV Enabled” indicator in the top left corner of the screen to see if 4-wire capability is present. Testers originally ordered without this function can be upgraded at any time. Contact CAMI Research for details.



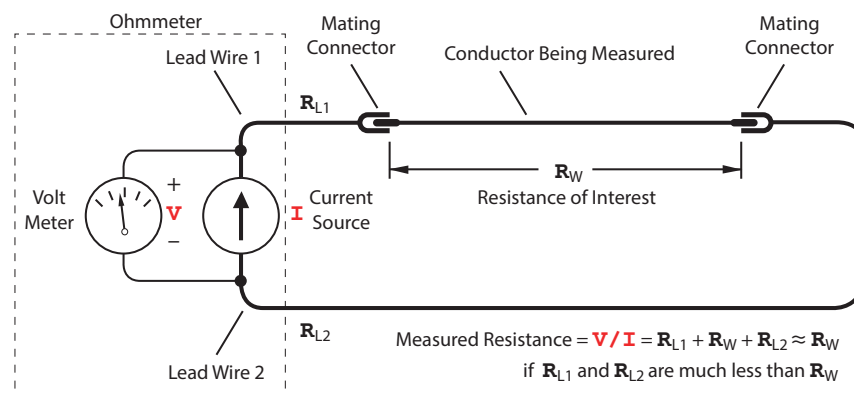
Four-Wire Kelvin measurement makes it possible to measure low resistance values up to 100 times more accurately than HVX testers not equipped with this function. While a standard HVX has a low range sensitivity of $0.1 \Omega \pm 0.1 \Omega$, the 4-Wire option increases the sensitivity to $0.001 \Omega \pm 0.001 \Omega$. Such precision resistance measurements become necessary when testing cables intended to carry significant current, or when extremely high reliability must be ensured in medical or military applications. Milliohm-sensitive measurement helps to locate bad solder joints, faulty crimps, recessed pins, pin contact contamination, improper wire gauge, and stress-extruded wire.

Comment: Many unfortunate accidents have developed in the past because of high-resistance connections in cables and wire harnesses which, in some cases, resulted in loss of life. This underscores the importance of accurately measuring resistance at the time these electrical components are manufactured.

For cables intended to transmit power or convey information between two points, we seek to minimize the natural resistance that the wire and connectors present to the source. During manufacture, cables must be tested to ensure that assembly errors or component defects do not result in resistance above some maximum threshold. The allowable maximum threshold depends on the application, wire gauge, and other factors. *Untested cables with abnormally high internal resistance could cause fire or thermal damage in power cables, and result in intermittent connections or circuit misoperation in signal cables.*

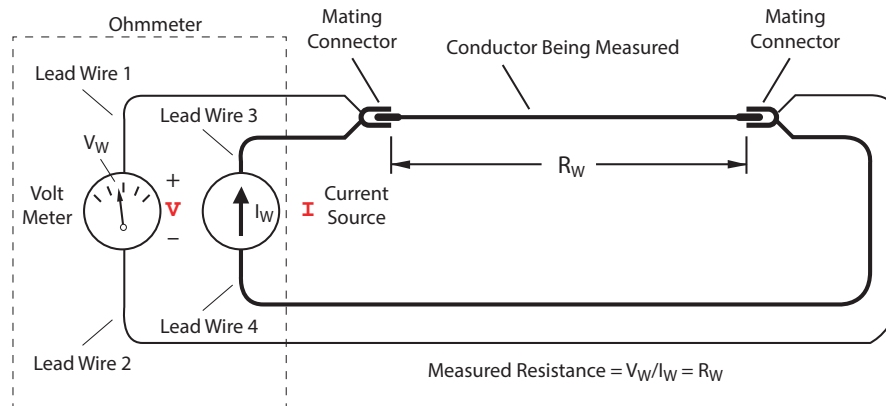
4.13.2 Principles of 4-Wire Measurement

Ohm's law defines resistance, “R”, as the ratio of voltage across a component, “V”, divided by the current passing through it, “I”: $R = V/I$. To measure resistance, we apply a test current to a wire and detect the voltage drop developed. From this, we easily calculate the resistance as shown in the following figure.



We measure the resistance of interest, R_W , between the conductor's two mating pins. The entire circuit, however, includes the resistance of the lead wires, R_{L1} and R_{L2} , so the voltage drop used in the calculation includes all three of these resistances. In many situations the lead wire resistance is much lower than the resistance of the conductor or component we aim to measure and therefore can be disregarded.

In some situations, however, the resistance of interest, R_W , approaches the resistance value of the lead wires used to measure it resulting in an inaccurate reading. We correct this problem by moving the voltage measurement points out to the endpoints of the mating pins, thus, bypassing any voltage drop that may occur in the lead wires. Refer to the figure below:



The Ohmmeter then appears to have four wires coming from it. Because we now use *four* lead wires instead of two, we refer to this approach as “4-wire measurement”, or alternatively “4-Wire Kelvin” measurement in honor of the 19th century British physicist, Lord Kelvin, who originally developed it.

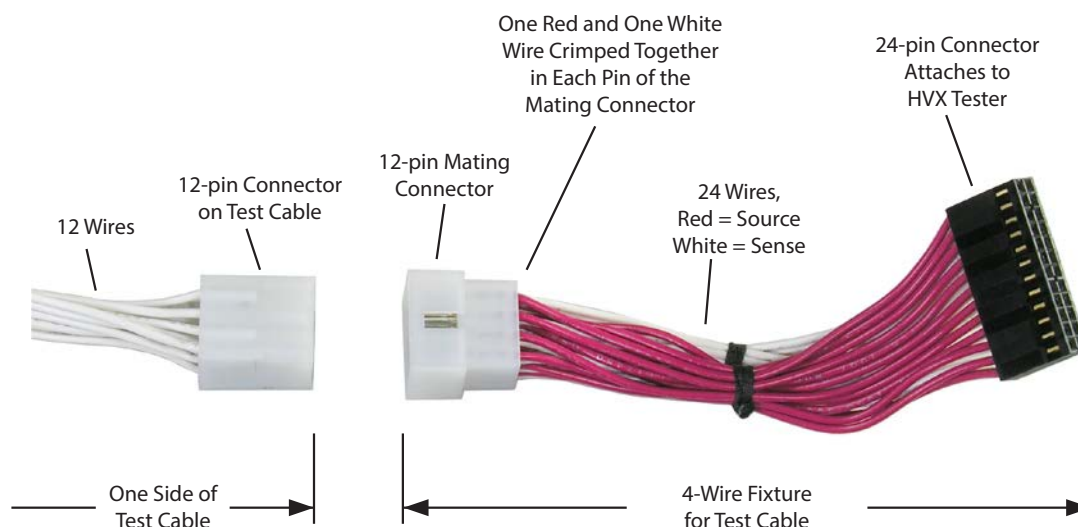
Note that the current flowing through the voltage-measuring circuit of a 4-wire system is extremely small, typically on the order of fractions of a microamp (six or more orders of magnitude less than the test current), so virtually no voltage drop occurs across these lead wires, and its effect on the resistance measurement is negligible. In summary, if there is no current flowing through a wire, there is no voltage drop across it *regardless of its length*.

The principal advantage of 4-wire measurement is that it eliminates any effect of fixture resistance (the lead wires) to obtain a precise resistance value of the unit under test (UUT). Because 4-wire measurements typically employ test currents well above those needed for two-wire testing, a secondary advantage comes through the use of a high-current stress test for wiring by driving a current of up to 1 A through each conductor, and setting a dwell time from 100 ms to 3 minutes; observing a slowly-increasing resistance during the dwell period resulting from thermal heating may reveal problems not detected with a shorter measurement interval.

The CableEye software permits individual conductors within a UUT to be independently disabled from 4-wire or high voltage test by User selection to avoid potential damage to fuses or other sensitive components. Users may also independently set different test currents for each conductor.

4.13.3 Test Fixtures for 4-Wire Measurement

The advantages of 4-wire measurement come at a cost. First, the test system requires *twice* the number of test points that would normally be required significantly increasing the equipment cost. Second, test fixtures must utilize *two wires* for every pin on the mating connector, one wire for the current source, and the other for voltage sense. This increases the cost and complexity of the test fixture. The illustration below shows a typical test fixture (adapter cable) for one end of a test cable.



The complete fixture attached to the tester appears in the image below. Note that 48 test points are required to test this 12-conductor cable.



Note that you *cannot* use standard CB boards or any custom fixture not specifically designed for this 4-wire testing! A 4-wire fixture requires TWO test points on the tester for every pin on your connectors. One of these two points is the *Source* which provides a programmable drive current into a pin, and the other the *Sense* which picks up the voltage response at that pin. Typically, we assign odd numbered header pins for the source and even number pins for the sense. This may also be reversed, if that is your standard. However, once you agree on a standard, all of your fixtures should be wired in this manner.

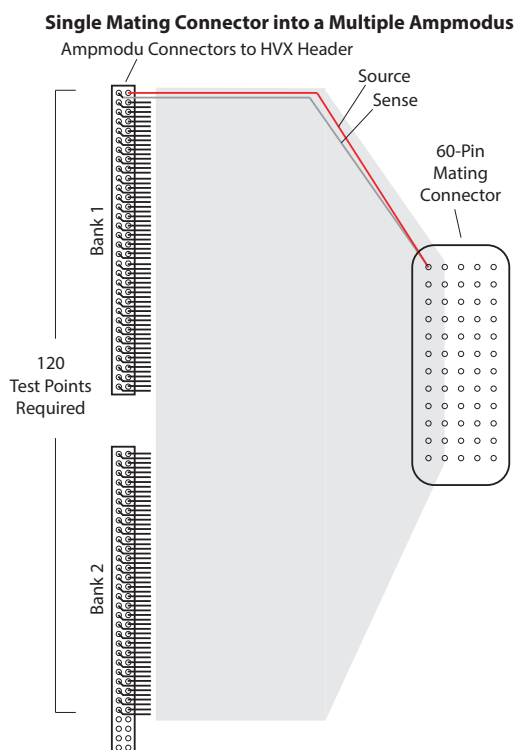
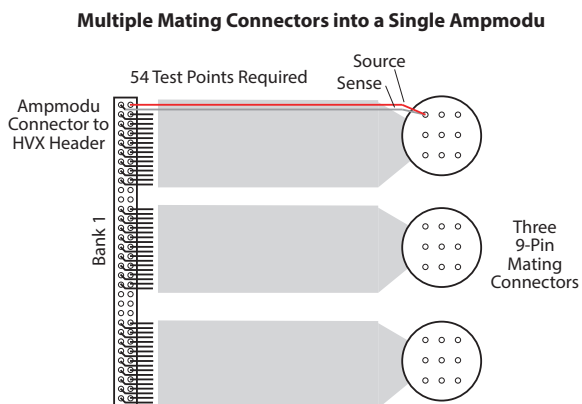
You may build rugged 4-wire fixtures using Ampmodu connectors like the ones shown in these photos. When the fixture is based on dual-row sockets like this, you may either directly connect them to the tester, or use a set of CB48 Header Isolator boards to lower the chance of damaging a pin on the tester.

When using Generic Headers for your connector choice, the graphics will default to dual-row headers to match the connectors on the tester, so if the cable you are testing has a different connector like the white Molex connectors shown in the previous page and you need a representative graphic, you will need to make a custom pin map for this fixture using the PinMap software, catalog Item 708. Details for creating a custom 4-Wire fixture map are described in Section 9.11, page 9-32.

When building 4-wire fixtures, keep in mind the following points:

- Because the Source pin can drive a current of up to 1000 mA into a pin, we recommend a 22-gauge or larger wire for this purpose. The Sense pin, however, will carry almost no current at all, so the wire used for Sense can be a much thinner which might be an advantage when trying to crimp two wires into a single pin of the mating connector.
- The length of the wire in the test fixture is unimportant in 4-wire measurements since the lead wire is not part of the resistance measurement.
- Before building the fixture, be sure that your tester has sufficient test points for the fixture you need! Remember that you need TWO test points for every pin in your mating connectors. To determine the minimum test point requirement, add up all the pins on all the connectors of your cable or harness, including any ground or shell conductors, and double this number to determine the total test points required.
- When building fixtures for multi-headed cables or wire harnesses, you may wire multiple connectors to a single Ampmodu connector to avoid wasting test points. When you do this, always ensure that an odd-numbered test point 'n' and the even-numbered test point immediately next in the sequence 'n+1' go to the same pin on the mating part. For example, TP1 and TP2 go to Pin 1 on the mating connector, TP3 and TP4 go to Pin 2 on the mating connector, etc.

If you wire multiple mating connectors into a single Ampmodu connector as shown below, or use two Ampmodu connectors for a single mating connector with 32 or more pins as shown on the right, you can use CableEye's PinMap software to obtain proper graphics and pin labels on the screen.



4.13.4 High Voltage Testing with 4-Wire Measurement

You may use any custom 4-Wire fixture for both 4-wire and hipot testing on the same cable during the same test cycle. If you choose to do both tests, the HVX system sequences tests in the following order: (1) low voltage test to determine connectivity and confirm that wire resistance is within global thresholds (if this test fails, the test process halts with a failure indicated), (2) hipot testing to check for dielectric breakdown and insulation resistance (optional), and (3) 4-wire testing to determine wire and contact resistance to the nearest m Ω . If a high voltage test is not required, only steps 1 and 3 will be performed.

Ampmodu connectors can withstand the maximum voltage capable of being produced by the HVX testers, so 4-wire fixtures wired with care using Ampmodu connectors will also suffice for high voltage testing at any voltage limit the tester can produce.

4.13.5 Initial Software and Hardware Setup

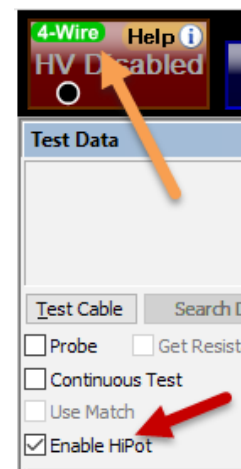
Your HVX-4W tester must run with software version 5.3, Build 1052, or higher, to use the 4-Wire test function. Once the software is installed and the tester turned on, set the **Active Test Points Switch** on the base unit to **128 test points** if it is not already there. If you have expansion modules and require these for the test, be sure each needed module is set to 128 test points.

Connect your 4-wire adapter cables or test fixture to the HVX system before continuing. Reminder: most standard CableEye CB boards are NOT 4-wire fixtures and cannot be used unless specifically designed for this purpose!

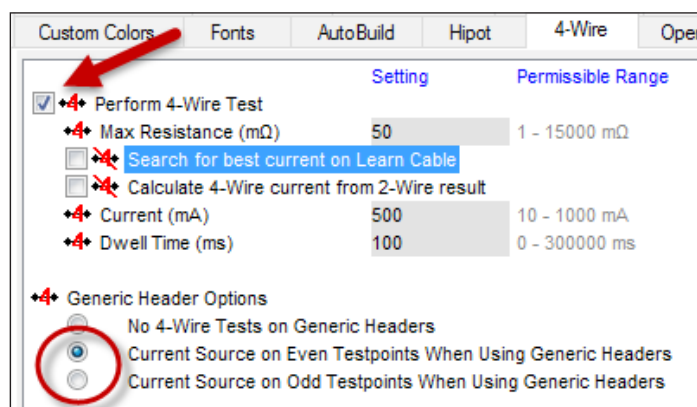
1. If the software detects the optional 4-wire module installed on your tester, you will see “4W” in the tool bar at the bottom of the screen. If you enable the checkbox “Enable HiPot”, you will also see a small green label in the upper left corner that says “4-Wire”. Note: Click on the green label to go directly to 4-Wire Help!



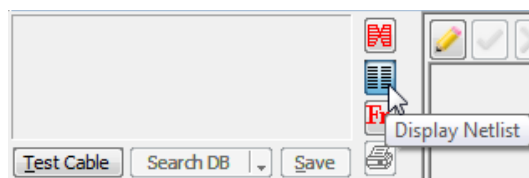
2. Double click on 4W on the bottom of the screen, or choose 4-Wire... from the Preferences Menu on the top to open the 4-Wire preferences panel.
3. Once the 4-Wire preferences panel is open, check the Perform 4-Wire Test checkbox to enable 4-Wire measurement.
4. Also choose Current Source on Even Pins, or Odd Pins, depending on how your fixture is wired.



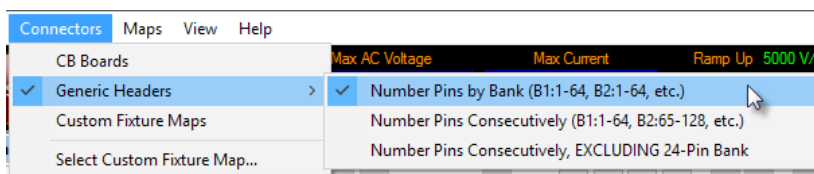
Note: if you use the same gauge wire for both source and sense pins, the the even and odd pins may be reversed, and the result will still be accurate with 4-Wire testing.



- You need to **Set the Test Data and Match Data Panels for Display Netlist**. This type of display will show the measured 4-wire resistance values which cannot be shown in the graphic display.

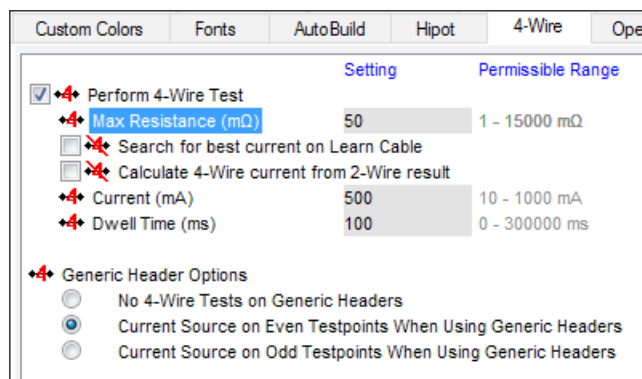


- In the **Connectors** menu, set the system for **Generic Headers** and in the second menu that appears, choose **Number Pins by Bank**.



4.13.6 Setting the 4-Wire Test Current and Resistance Limits

- Access 4-wire preferences again, by **double clicking** on **4W** on the bottom of the screen or from the preferences menu. If you are not planning to test the cable at high voltage, it is not necessary to press the black button on the front panel to enable it. The 4-wire function will operate without enabling high voltage.
- Max Resistance (mΩ)** sets the threshold above which a fault will be declared. The maximum resistance value permitted is 15000 mΩ (15 Ω), however, most 4-wire measurements will normally have a threshold of under 1 Ω.
- Search for Best Current on Learn Cable** scans each wire when a cable is learned to find that current which will produce the most accurate resistance reading. If the test current is not specified and you plan to learn the first article, we recommend this setting.
- The **Calculate 4-Wire Current from 2-Wire Result** option estimates the best test current based on the resistassnce reading obtained in the initial continuity scan. Because the relays are not involved in this test, the Learn function proceeds much more quickly. This would be important if you have a large harness.
- Current (mA)** allows you to fix the test current at a value of your choice from 100 mA to 1000 mA. Use this if your test instructions require a specific test current. In generally, a high test current, such as 1000 mA, provides a useful current stress test for conductors under 0.5 Ω
- Dwell Time (ms)** specifies the time that the test signal will be applied above the 50 ms minimum time necessary to make a measurement. The range is 0 ms to 300 s.
- Once the desired settings have been selected, click the OK button at the bottom of the control panel to save this as your global settings. These values will be retained when you next start the tester.

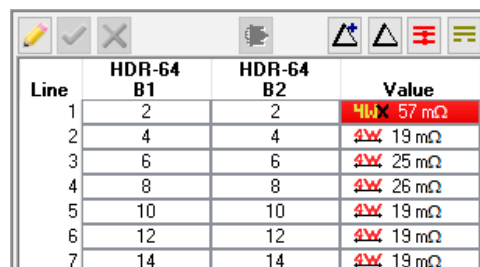


4.13.7 Testing Cables and Viewing Results

1. Connect a good cable to the tester and **click Learn Cable**. Confirm that the cable is represented accurately in the Match Data Netlist.

Note: with the system set to Generic Headers, you will see consecutive pins on your cable labeled as all even numbers (2, 4, 6, 8, ...) because two test points are required for each connector pin. You may use PinMap to create a custom map for your 4-Wire fixture to show proper graphics and pin numbering.

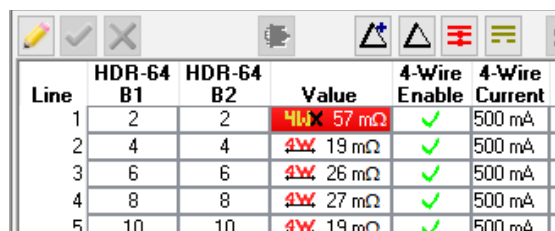
2. Leave the good cable in place and **click Test Cable**. The relays will click briefly while the test current is applied to each wire in the cable. When the test finishes, you will see the **measured resistance** appear in the **Value** column. Resistances found to be above the threshold you set earlier will appear with a red background and set the Fail flag. Make any adjustment to the 4-wire Pass/Fail threshold that may be necessary to obtain a correct Pass/Fail result.



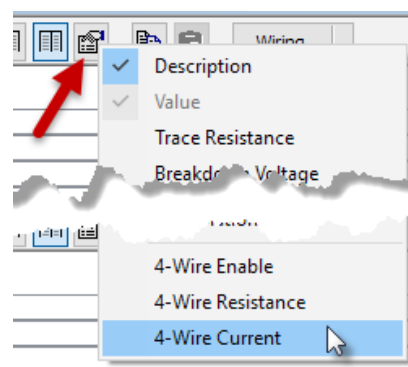
Line	HDR-64 B1	HDR-64 B2	Value
1	2	2	4W 57 mΩ
2	4	4	4W 19 mΩ
3	6	6	4W 25 mΩ
4	8	8	4W 26 mΩ
5	10	10	4W 19 mΩ
6	12	12	4W 19 mΩ
7	14	14	4W 19 mΩ

In this case the first line reports a fail with 57 mΩ, which is higher than the 50 mΩ Max Resistance set in the 4-Wire preference panel.

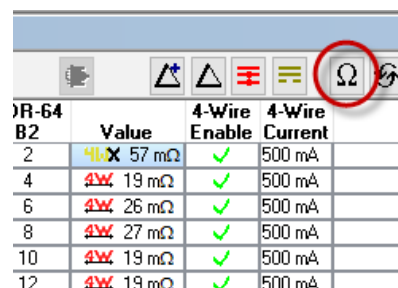
3. In the **Match Data** window, **click Save** to save the cable wiring and 4-Wire setup in the database. You are now ready to begin production testing with 4-Wire resistance.
4. (optional) If you wish to also test the cable at high voltage, return to the high voltage preferences panel and choose the high voltage tests and voltages that you need. Click OK and repeat the test. Note: Be sure that your 4-Wire fixture will sustain the test voltage you choose.
5. If you want to see the 4-Wire current value used, **click** on the **Show/Hide Other Columns** button in the tool bar and choose **4-Wire Current** at the bottom of the list.



Line	HDR-64 B1	HDR-64 B2	Value	4-Wire Enable	4-Wire Current
1	2	2	4W 57 mΩ	✓	500 mA
2	4	4	4W 19 mΩ	✓	500 mA
3	6	6	4W 26 mΩ	✓	500 mA
4	8	8	4W 27 mΩ	✓	500 mA
5	10	10	4W 19 mΩ	✓	500 mA

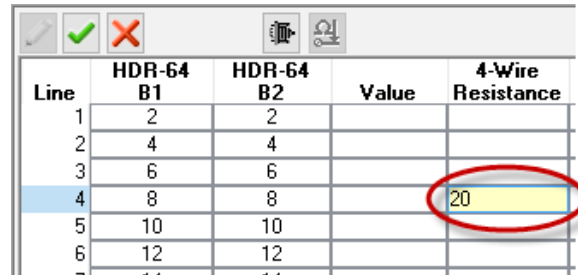


6. You may manually re-measure one specific conductor by first clicking on the value that interests you, and then clicking on the **Ω button** in the tool bar. You may, if you wish, test the connection by flexing the cable and remeasuring to see if the resistance value fluctuates; if it does, you may have an intermittent connection.



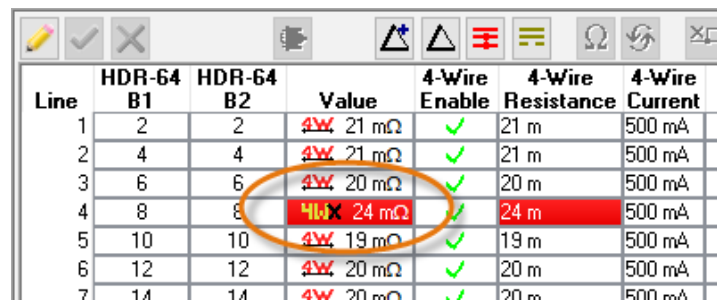
HDR-64 B2	Value	4-Wire Enable	4-Wire Current
2	4W 57 mΩ	✓	500 mA
4	4W 19 mΩ	✓	500 mA
6	4W 26 mΩ	✓	500 mA
8	4W 27 mΩ	✓	500 mA
10	4W 19 mΩ	✓	500 mA
12	4W 19 mΩ	✓	500 mA

7. Also notice the **4-Wire Enable** column in the previous image. You may block any conductor from a 4-Wire test by **double-clicking** on the **Green Checkmark** to change this to a **Red X**. This would be important if your cable contained low-current fuse or other sensitive component.
8. You can set individual 4-Wire Resistance Thresholds or limits for a Pass/Fail condition in each connection. To do this, add the **4-Wire Resistance** column if you haven't done it yet and type in the resistance value in $m\Omega$ for the desired connection. In this example we type in 20 for 20 $m\Omega$ like shown in the image. When done, click on the green checkmark to accept the changes.



Line	HDR-64 B1	HDR-64 B2	Value	4-Wire Resistance
1	2	2		
2	4	4		
3	6	6		
4	8	8		20
5	10	10		
6	12	12		

Finally you should test the cable again with the new resistance threshold. Click on Test Cable and wait for the result. You will notice that this time, that specific connection failed at 24 $m\Omega$ comparing it to the new value of 20 $m\Omega$ set on the match data, and ignoring the 50 $m\Omega$ generic threshold.



Line	HDR-64 B1	HDR-64 B2	Value	4-Wire Enable	4-Wire Resistance	4-Wire Current
1	2	2	21 $m\Omega$	✓	21 m	500 mA
2	4	4	21 $m\Omega$	✓	21 m	500 mA
3	6	6	20 $m\Omega$	✓	20 m	500 mA
4	8	8	24 $m\Omega$	✓	24 m	500 mA
5	10	10	19 $m\Omega$	✓	19 m	500 mA
6	12	12	20 $m\Omega$	✓	20 m	500 mA
7	14	14	20 $m\Omega$	✓	20 m	500 mA