Tektronix®

Model 4200A-SCS

Prober and External Instrument Control

4200A-913-01C April 2025

KEITHLEY
A Tektronix Company

Model 4200A-SCS

Parameter Analyzer

Prober and External Instrument Control

Document number: 4200A-913-01 Rev. C April 2025

KEITHLEY
A Tektronix Company

Safety precautions

The following safety precautions should be observed before using this product and any associated instrumentation. Although some instruments and accessories would normally be used with nonhazardous voltages, there are situations where hazardous conditions may be present.

This product is intended for use by personnel who recognize shock hazards and are familiar with the safety precautions required to avoid possible injury. Read and follow all installation, operation, and maintenance information carefully before using the product. Refer to the user documentation for complete product specifications.

If the product is used in a manner not specified, the protection provided by the product warranty may be impaired.

The types of product users are:

Responsible body is the individual or group responsible for the use and maintenance of equipment, for ensuring that the equipment is operated within its specifications and operating limits, and for ensuring that operators are adequately trained.

Operators use the product for its intended function. They must be trained in electrical safety procedures and proper use of the instrument. They must be protected from electric shock and contact with hazardous live circuits.

Maintenance personnel perform routine procedures on the product to keep it operating properly, for example, setting the line voltage or replacing consumable materials. Maintenance procedures are described in the user documentation. The procedures explicitly state if the operator may perform them. Otherwise, they should be performed only by service personnel.

Service personnel are trained to work on live circuits, perform safe installations, and repair products. Only properly trained service personnel may perform installation and service procedures.

Keithley products are designed for use with electrical signals that are measurement, control, and data I/O connections, with low transient overvoltages, and must not be directly connected to mains voltage or to voltage sources with high transient overvoltages. Measurement Category II (as referenced in IEC 60664) connections require protection for high transient overvoltages often associated with local AC mains connections. Certain Keithley measuring instruments may be connected to mains. These instruments will be marked as category II or higher.

Unless explicitly allowed in the specifications, operating manual, and instrument labels, do not connect any instrument to mains.

Exercise extreme caution when a shock hazard is present. Lethal voltage may be present on cable connector jacks or test fixtures. The American National Standards Institute (ANSI) states that a shock hazard exists when voltage levels greater than 30 V RMS, 42.4 V peak, or 60 VDC are present. A good safety practice is to expect that hazardous voltage is present in any unknown circuit before measuring.

Operators of this product must be protected from electric shock at all times. The responsible body must ensure that operators are prevented access and/or insulated from every connection point. In some cases, connections must be exposed to potential human contact. Product operators in these circumstances must be trained to protect themselves from the risk of electric shock. If the circuit is capable of operating at or above 1000 V, no conductive part of the circuit may be exposed.

Do not connect switching cards directly to unlimited power circuits. They are intended to be used with impedance-limited sources. NEVER connect switching cards directly to AC mains. When connecting sources to switching cards, install protective devices to limit fault current and voltage to the card.

Before operating an instrument, ensure that the line cord is connected to a properly-grounded power receptacle. Inspect the connecting cables, test leads, and jumpers for possible wear, cracks, or breaks before each use.

When installing equipment where access to the main power cord is restricted, such as rack mounting, a separate main input power disconnect device must be provided in close proximity to the equipment and within easy reach of the operator.

For maximum safety, do not touch the product, test cables, or any other instruments while power is applied to the circuit under test. ALWAYS remove power from the entire test system and discharge any capacitors before connecting or disconnecting cables or jumpers, installing or removing switching cards, or making internal changes, such as installing or removing jumpers.

Do not touch any object that could provide a current path to the common side of the circuit under test or power line (earth) ground. Always make measurements with dry hands while standing on a dry, insulated surface capable of withstanding the voltage being measured.

For safety, instruments and accessories must be used in accordance with the operating instructions. If the instruments or accessories are used in a manner not specified in the operating instructions, the protection provided by the equipment may be impaired.

Do not exceed the maximum signal levels of the instruments and accessories. Maximum signal levels are defined in the specifications and operating information and shown on the instrument panels, test fixture panels, and switching cards.

When fuses are used in a product, replace with the same type and rating for continued protection against fire hazard.

Chassis connections must only be used as shield connections for measuring circuits, NOT as protective earth (safety ground) connections.

If you are using a test fixture, keep the lid closed while power is applied to the device under test. Safe operation requires the use of a lid interlock.

If a ⏚ screw is present, connect it to protective earth (safety ground) using the wire recommended in the user documentation.

The ⚠ symbol on an instrument means caution, risk of hazard. The user must refer to the operating instructions located in the user documentation in all cases where the symbol is marked on the instrument.

The ⚡ symbol on an instrument means warning, risk of electric shock. Use standard safety precautions to avoid personal contact with these voltages.

The hot-surface symbol on an instrument shows that the surface may be hot. Avoid personal contact to prevent burns.

The chassis-terminal symbol indicates a connection terminal to the equipment frame.

If this (Hg) symbol is on a product, it indicates that mercury is present in the display lamp. Please note that the lamp must be properly disposed of according to federal, state, and local laws.

The **WARNING** heading in the user documentation explains hazards that might result in personal injury or death. Always read the associated information very carefully before performing the indicated procedure.

The **CAUTION** heading in the user documentation explains hazards that could damage the instrument. Such damage may invalidate the warranty.

The **CAUTION** heading with the ⚠ symbol in the user documentation explains hazards that could result in moderate or minor injury or damage the instrument. Always read the associated information very carefully before performing the indicated procedure. Damage to the instrument may invalidate the warranty.

Instrumentation and accessories shall not be connected to humans.

Before performing any maintenance, disconnect the line cord and all test cables.

To maintain protection from electric shock and fire, replacement components in mains circuits — including the power transformer, test leads, and input jacks — must be purchased from Keithley. Standard fuses with applicable national safety approvals may be used if the rating and type are the same. The detachable mains power cord provided with the instrument may only be replaced with a similarly rated power cord. Other components that are not safety-related may be purchased from other suppliers as long as they are equivalent to the original component (note that selected parts should be purchased only through Keithley to maintain accuracy and functionality of the product). If you are unsure about the applicability of a replacement component, call a Keithley office for information.

Unless otherwise noted in product-specific literature, Keithley instruments are designed to operate indoors only, in the following environment: Altitude at or below 2,000 m (6,562 ft); temperature 0 °C to 50 °C (32 °F to 122 °F); and pollution degree 1 or 2.

To clean an instrument, use a cloth dampened with deionized water or mild, water-based cleaner. Clean the exterior of the instrument only. Do not apply cleaner directly to the instrument or allow liquids to enter or spill on the instrument. Products that consist of a circuit board with no case or chassis (e.g., a data acquisition board for installation into a computer) should never require cleaning if handled according to instructions. If the board becomes contaminated and operation is affected, the board should be returned to the factory for proper cleaning/servicing.

Safety precaution revision as of June 2018.

Table of contents

Section 1

Introduction

In this section:

Introduction

This document contains information about using switch matrices, probers, and other external equipment with the 4200A-SCS.

Section 2

Using switching matrices

In this section:

Typical test systems using a switching matrix

A switching matrix enhances the connectivity of the 4200A-SCS by allowing any SMU or preamplifier signal to be connected to any DUT pin. The following paragraphs summarize recommended switching mainframes and matrix cards, and also show typical connecting schemes with SMUs and preamplifiers.

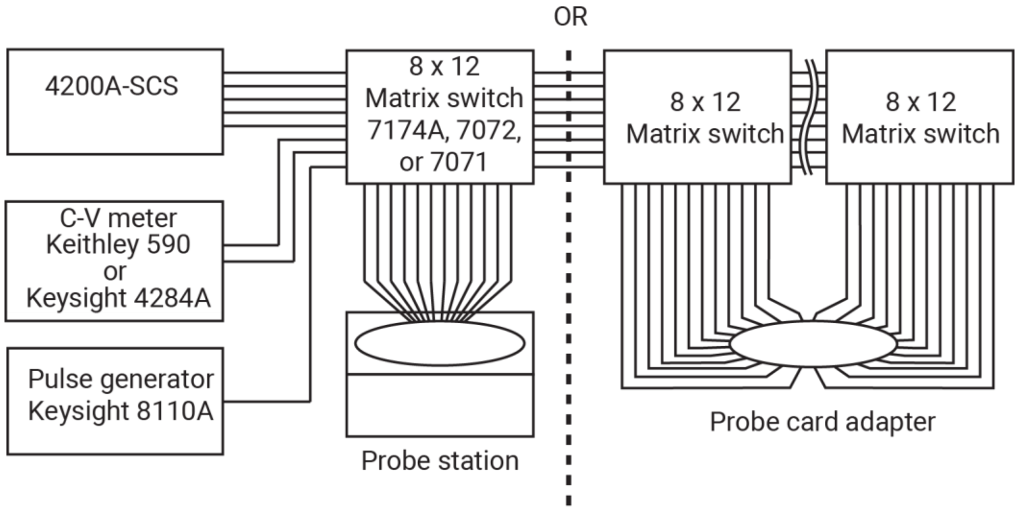
A switching matrix provides automatic switching for test instrumentation and devices under test (DUTs). Typical switching matrix systems are shown in the following figure.

The 4200A-SCS supports the Keithley Series 700 Switching System as external instruments. This series includes the 707, 707A, and 707B, which have six slots for matrix cards. This provides up to 72 pins of switching. This series also includes the 708, 708A, and 708B, which support a single matrix card for 12 pins of matrix switching.

When using a switching matrix, one probe station or one test fixture must be present in the system configuration because the probe station or test fixture establishes the number of test-system pins. The matrix is cabled to the test system pins, and instrument terminals are routed through the matrix to the pins using the user modules in the `Matrixulib` user library.

The following figure shows switching matrix cards connected to a probe station in order to test a wafer. You can also replace a probe station with a test fixture to test discrete devices.

Figure 1: Typical systems using a switching matrix



Matrix card types

The recommended Keithley matrix cards are:

- Model 7072 8 x 12 Semiconductor Matrix Card, <1 pA offset current
- Model 7174A 8 x 12 Low Current Matrix Card, <100 fA offset current

Note that a key characteristic of these cards is low offset current to minimize the negative effects of offset currents on low-current measurements.

7072 Semiconductor Matrix Card

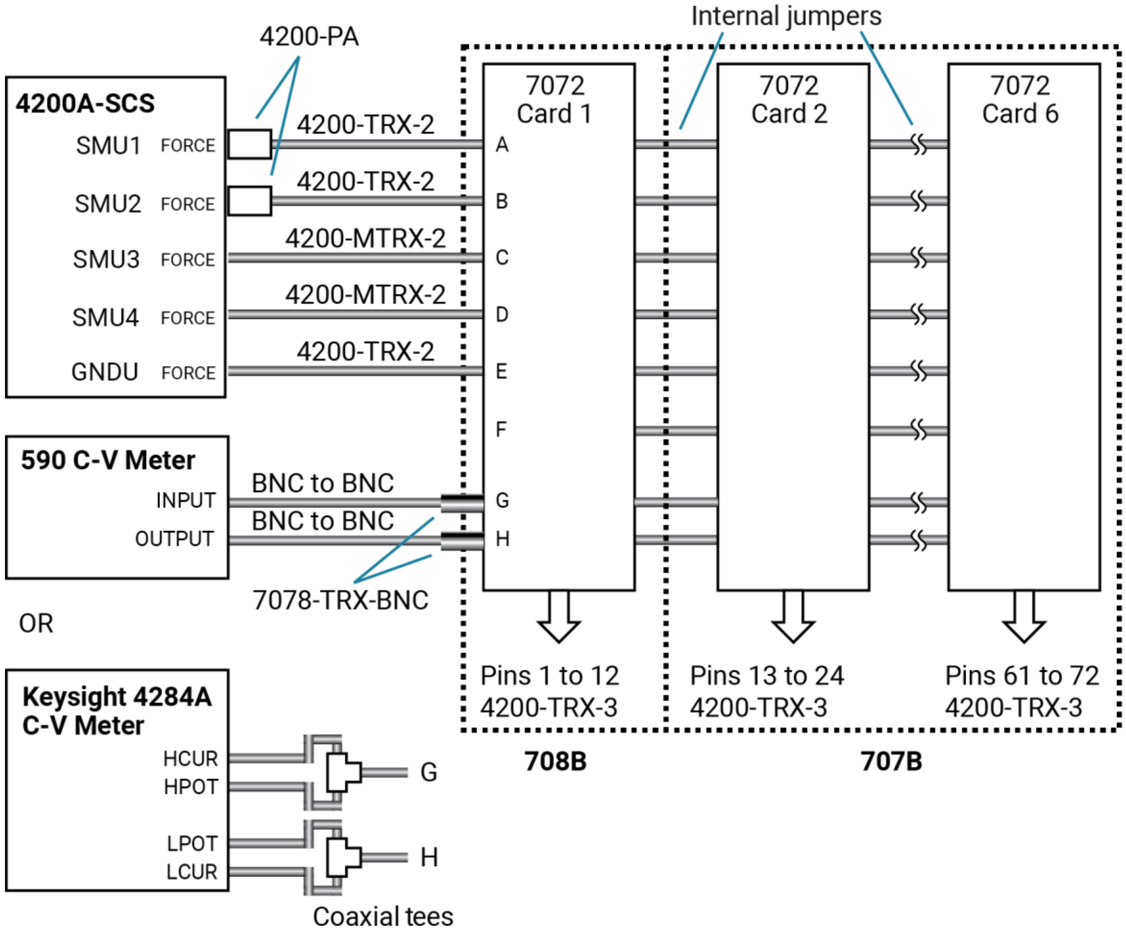
The 7072 provides two two-pole low-current paths that have <1 pA offset current (rows A and B), two one-pole CV paths for characterization from DC to 1 MHz (rows G and H), and four two-pole paths for general purpose switching (rows C, D, E, and F). The card is equipped with 3-lug triaxial connectors for signal connections. The maximum signal level is 200 V, 1 A. The maximum leakage is 0.01 pA/V and the 3 dB bandwidth is 5 MHz (CV channels).

The following figure shows a test system using 7072 matrix cards. The connection requirements for this card are the same as the connection requirements for the 7174A. Notice that the C-V meter is connected to rows G and H. These two rows are optimized for C-V measurements.

If using preamplifiers with the 4200A-SCS, they should be connected to the first two rows of the 7072 matrix card.

The following figure and the C-V Analyzer signal paths (on page 2-22) figures show how signals are routed through 7072 matrix switches to a DUT.

Figure 2: Test system using 7072 matrix cards



7174A Low Current Matrix Card

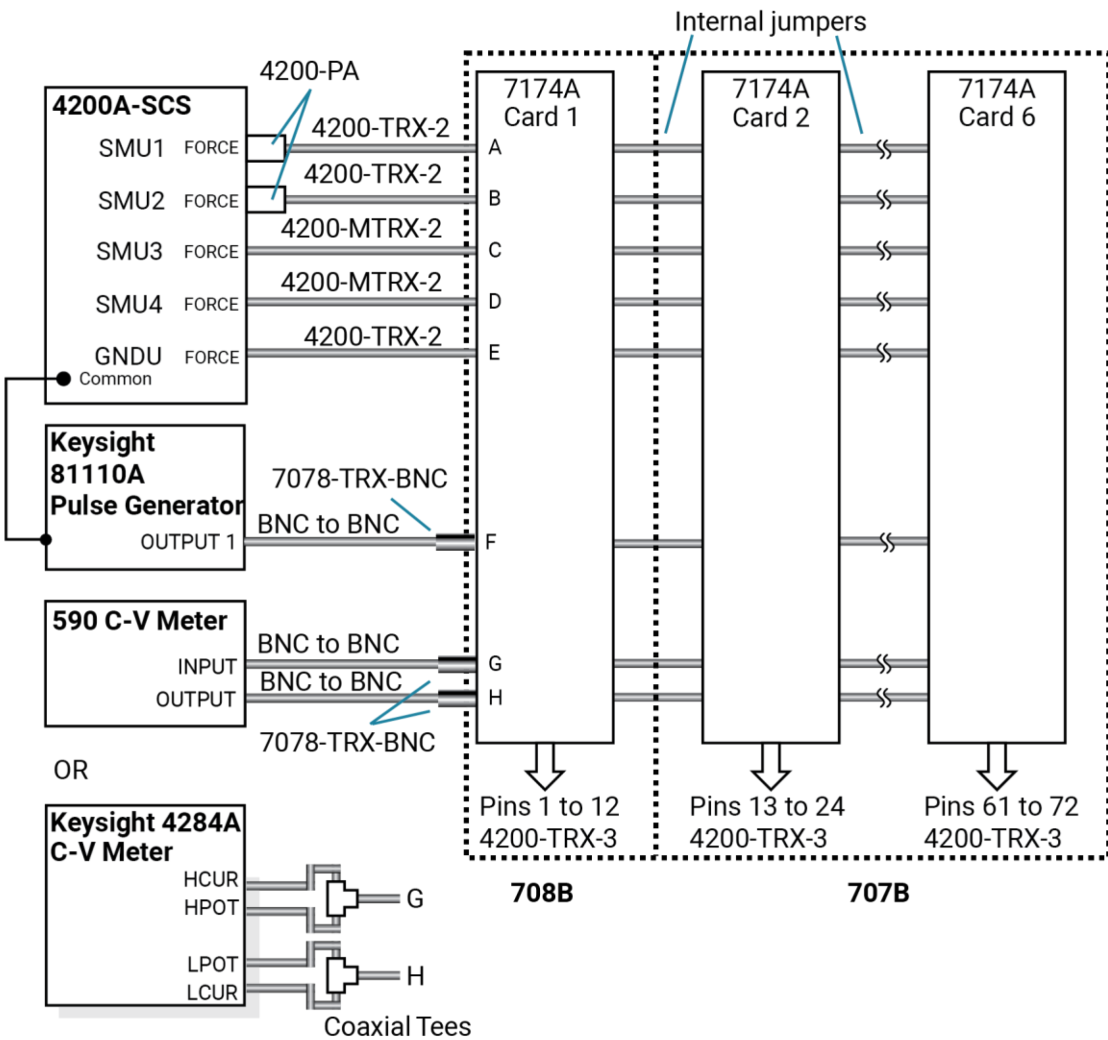
The 7174A provides high quality, high performance switching of I-V and C-V signals. This matrix card uses 3-pole switching (HI, LO, Guard) with 10 fA typical offset current. The card is equipped with 3-lug triaxial connectors for signal connections.

The following figures show test systems using 7174A matrix cards. The supplied triaxial cables connect the 4200A-SCS directly to matrix rows. The other instruments in the system are fitted with BNC connectors that require the use of BNC-to-triaxial adapters.

7174A connections for local sensing

The following figure shows a system that uses local sensing. Coaxial tees adapt the Keysight 4284A C-V meter for two-terminal operation.

Figure 3: Test system using 7174A matrix cards



7174A connections for remote sensing

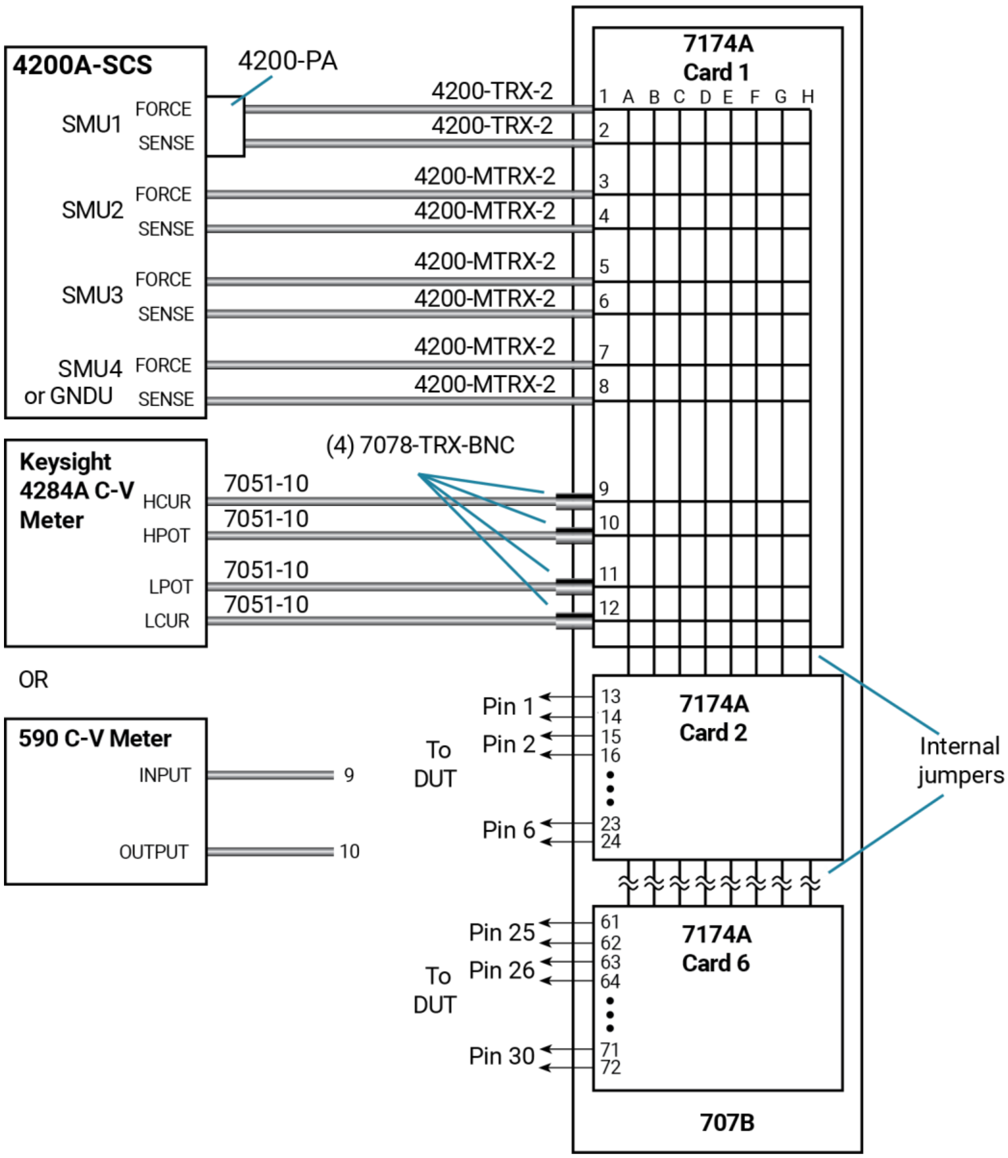
The following figure shows how to connect instrumentation for remote sense operation. Since there are not enough matrix rows, the instruments are connected to the matrix columns. In this configuration, two switching relays are closed to complete a path from an instrument to a device under test (DUT). With five DUT matrix cards installed in a Series 700 Switching System mainframe, up to 30 DUT pin-pairs can be used.

NOTE

In the following figure, the C-V Analyzer signal paths (on page 2-22) for the Keysight Model 4980A and Keysight Model 8110A pulse generator signal path (on page 2-25) show how signals are routed through 7174A matrix switches to a DUT.

In this example, the instrumentation is connected to matrix columns, so the switching matrix is rotated 90° for illustration purposes.

Figure 4: Remote sense test system using 7174A matrix cards



Switching matrix mainframes

The 4200A-SCS provides a user library that contains preconfigured data acquisition and control user modules for the Series 700 Switching System.

You can use the 4200A-SCS with switching matrices from other vendors. However, you will need to develop software to control these matrices from Clarius+. See *Model 4200A-SCS KULT and KULT Extension Programming* (4200A-KULT-907-01) for information about developing user modules and libraries.

Card installation

Refer to the instructions for your matrix card for card installation instructions.

GPIB connections

The 4200A-SCS controls the switching matrix using the GPIB interface. Connect the GPIB port of the switching matrix to the 4200A-SCS using a shielded GPIB cable.

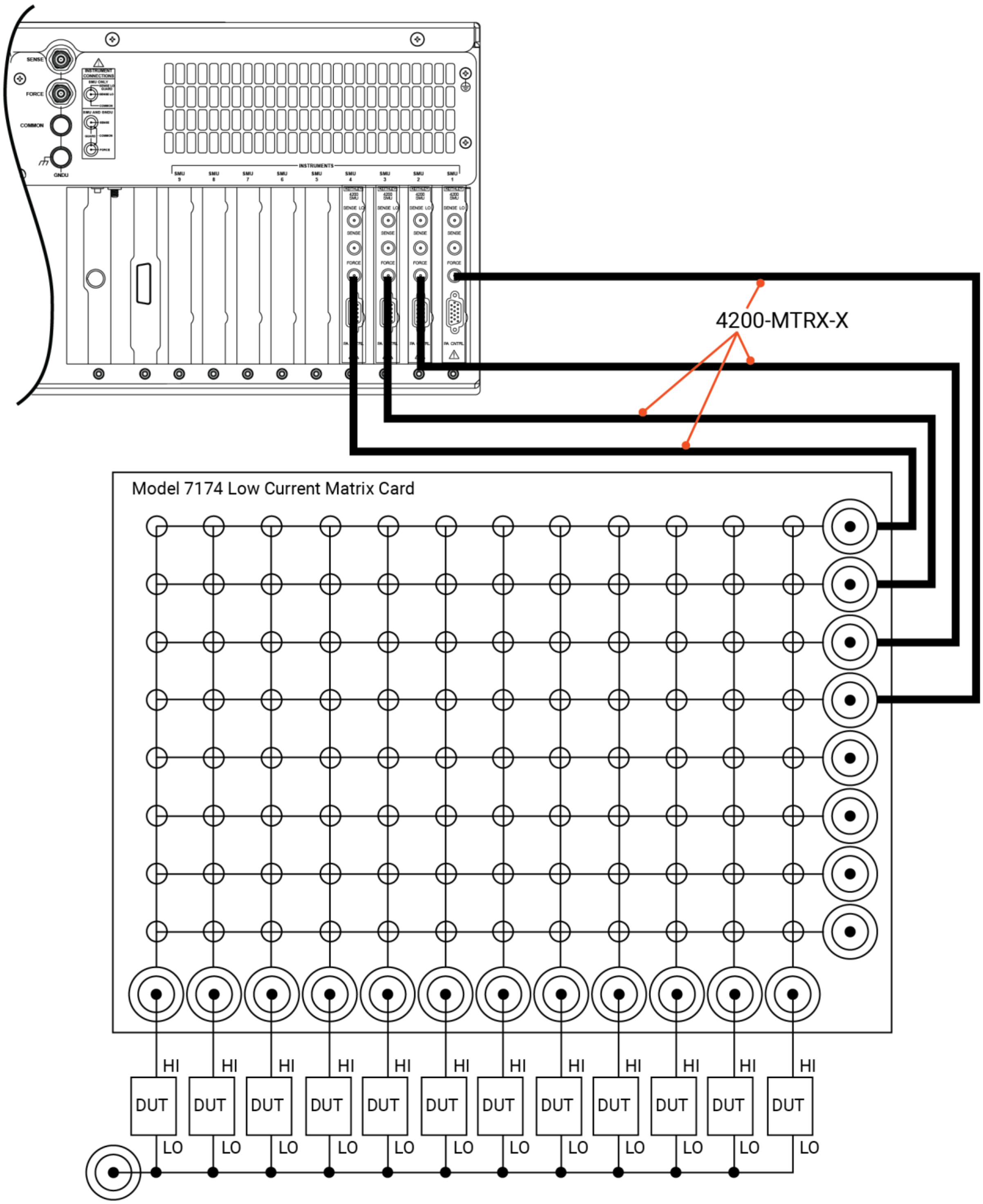
Switching matrix connections

A switching matrix enhances the connectivity of the 4200A-SCS by allowing any SMU or preamplifier signal to be connected to any DUT pin. Typically, devices are connected to columns and instruments are connected to rows. The following topics summarize recommended switching mainframes and matrix cards. They also show typical connection schemes with SMUs and preamplifiers.

Typical SMU matrix card connections

The following figure shows typical SMU matrix card connections using local sensing. The four SMU FORCE terminals are connected to the matrix card rows, while the DUT HI terminals are connected to the matrix card columns. All 12 DUT LO terminals are connected together, and the DUT LO signal is connected to the ground unit FORCE terminal. Any SMU FORCE terminal can be connected to any DUT HI terminal simply by closing the appropriate matrix crosspoint.

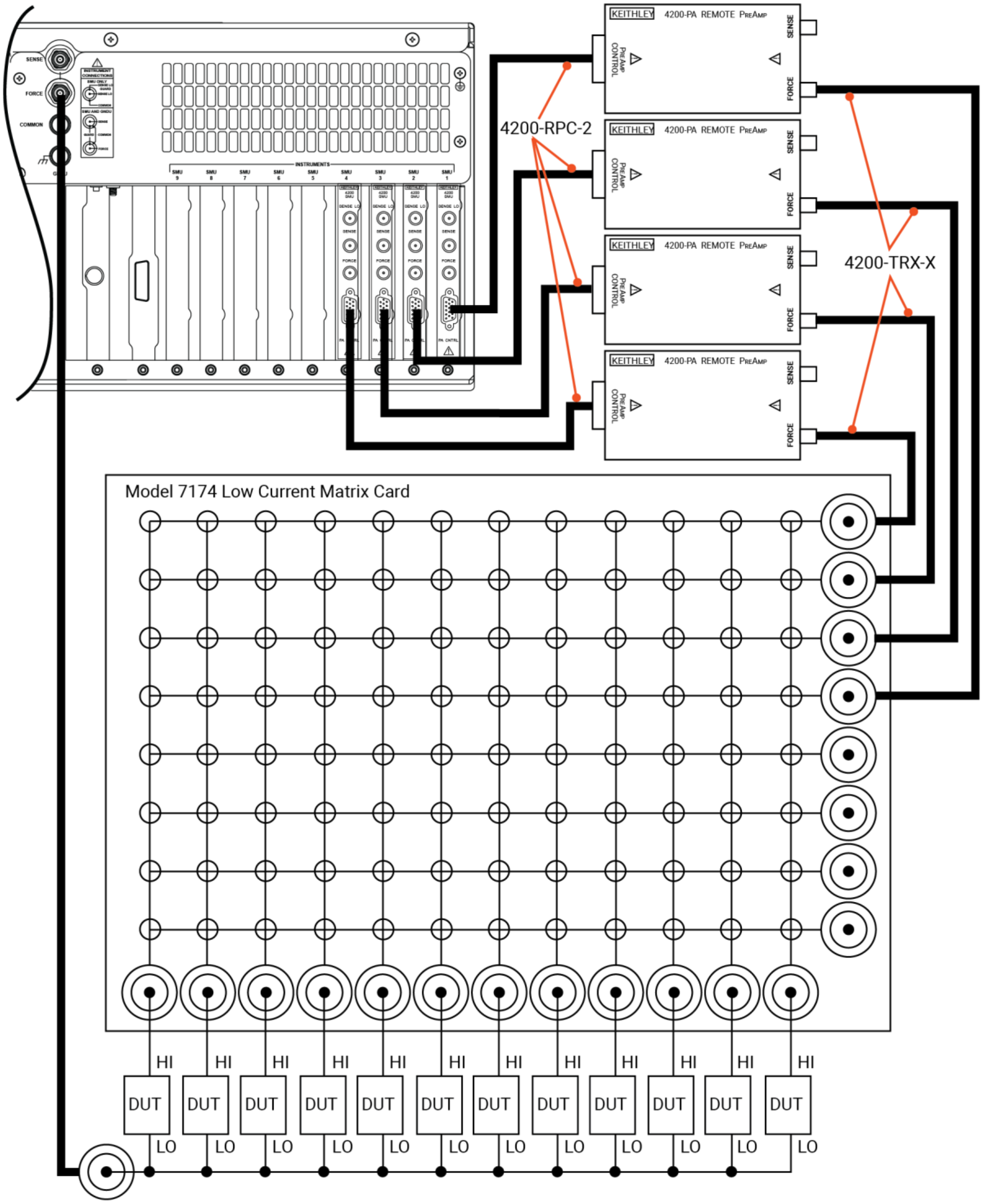
Figure 5: Typical SMU matrix card connections



Typical preamplifier matrix card connections

The following figure shows typical preamplifier matrix card connections using local sensing. This configuration is similar to the SMU configuration shown in the previous figure, except that preamplifiers are added for low-current source-measure capabilities. The preamplifier FORCE terminals are connected to the matrix card rows, while the DUT HI terminals are connected to the matrix card columns. All 12 DUT LO terminals are connected together, and the common DUT LO signal is connected to the ground unit FORCE terminal. Any preamplifier FORCE terminal can be connected to any DUT HI terminal by closing the appropriate matrix crosspoint.

Figure 6: Preamplifier matrix card connections



Typical CVU matrix card connections

In your project, you can automate the use of a CVU and other instrumentation using a switching matrix and actions to control the switching. When the project is run, the switching matrix automatically makes the required instrument connections for each test in the project.

The next figures show typical connections for a switching system using a Series 700 Switching System with the 7174A Matrix Card installed.

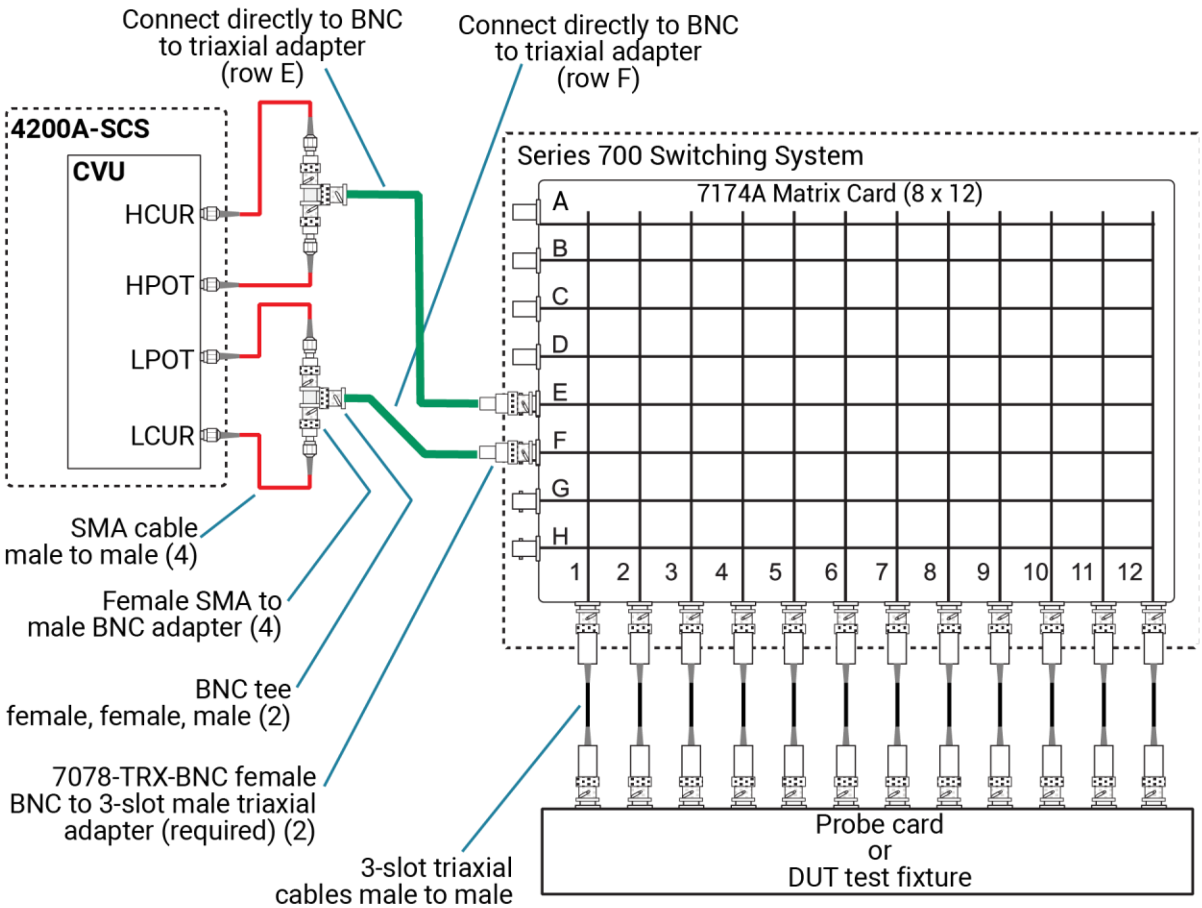
NOTE

You can also use the 7072 Matrix Card for C-V testing. If you are using the 7072, you must use rows G and H and local (2-wire) sensing.

The SMA cables and adapters shown in the following figures are supplied with the CVU or the 4200-CVU-PROBER-KIT. The triaxial and BNC cables are not supplied. The prober kit includes two types of BNC-to-triaxial adapters that connect directly to the rows of the matrix. The 7078-TRX-BNC has the guard connected to the inner shield of the adapter. The 7078-TRX-GND has the guard disconnected.

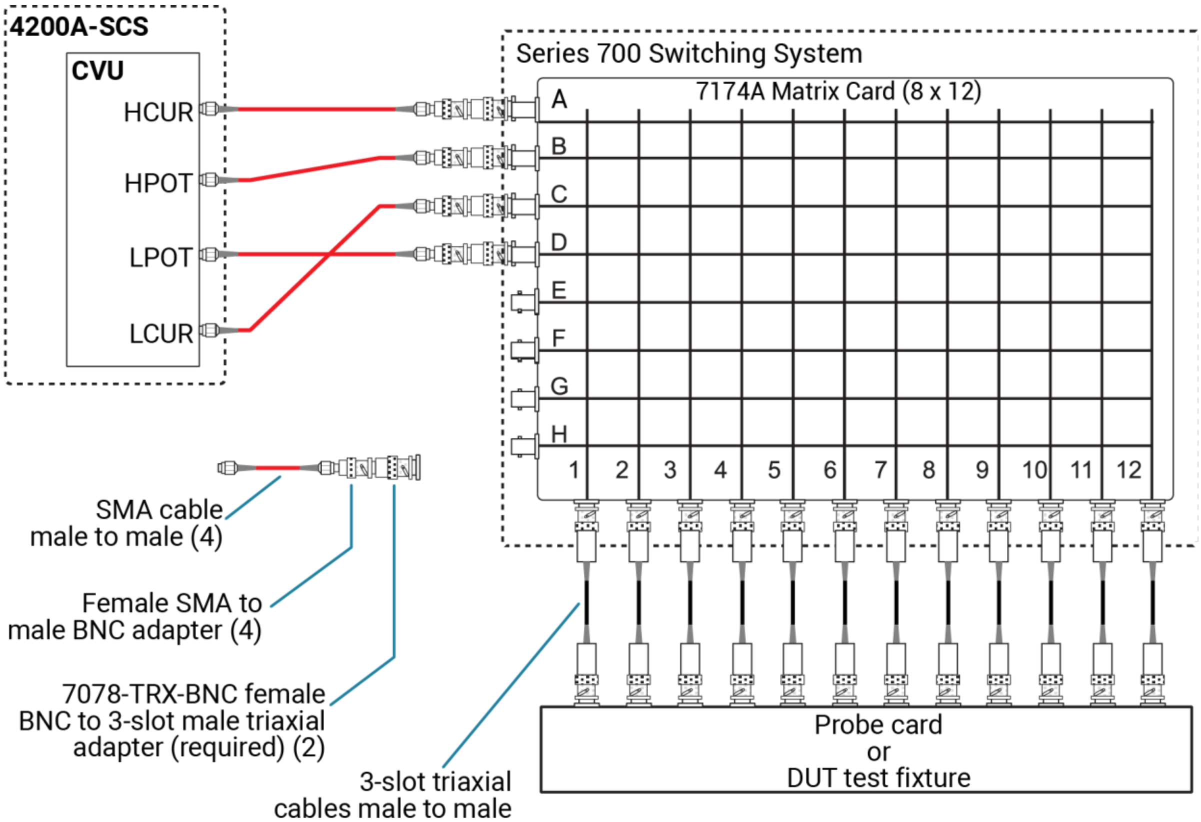
This figure shows connections for local (2-wire) sensing with the CVU connected to rows E and F of the matrix. This is the connection scheme for the `cap-iv-cv-matrix` project. For details, see “cap-iv-cv-matrix” in the *Model 4200A-SCS Capacitance-Voltage Unit (CVU) User's Manual*.

Figure 7: Test connections for a switching matrix - local (2-wire) sensing



The following figure shows connections for remote (4-wire) sensing.

Figure 8: Test connections for a switching matrix - remote (4-wire) sensing



NOTE

The 7078-TRX-BNC adapters must be used in order to extend SMA shielding through the matrix card.

NOTE

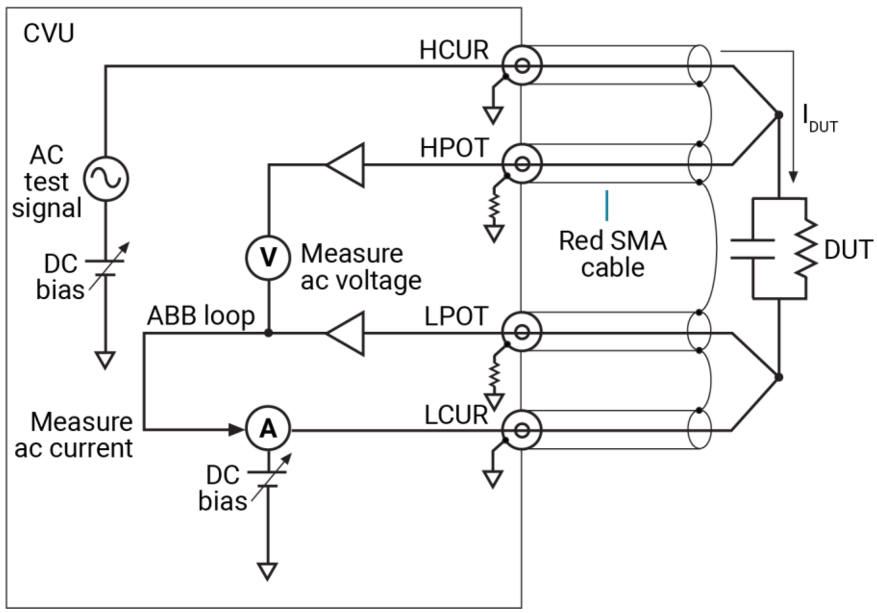
The shields of the SMA cables must be connected together and extended as far as possible to the DUT, as shown in Typical CVU test connections to a DUT (on page 2-13).

Typical CVU test connections to a DUT

The shields of the SMA cables must be connected together and extended as far as possible to the device under test (DUT), as shown in the following figure.

Use the torque wrench supplied with the 4200A-SCS to tighten the SMA connections to 8 in. lb.

Figure 9: Measurement circuit (simplified)



NOTE

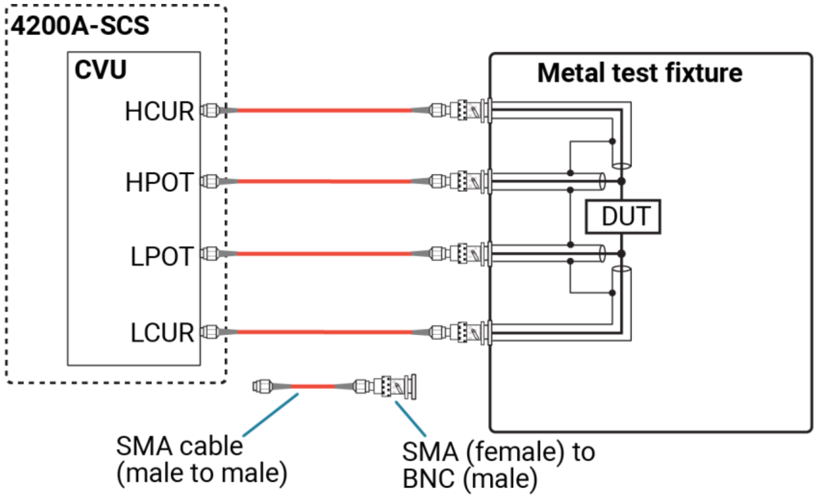
You can swap the HCUR and HPOT and LCUR and LPOT terminal functionality in Clarius.

NOTE

The shields of the red SMA cables must be connected together near the DUT.

The following figure shows typical connections to a DUT installed in a test fixture that has BNC bulkhead connectors. Use a conductive test fixture with the bulkhead connectors mounted directly to the test fixture. Do not use insulators between the connectors and test fixture. The cables and adapters are supplied with the 4210-CVU or 4215-CVU.

Figure 10: Typical CVU connections to a DUT in a test fixture



Connection scheme settings

The following connection scheme settings are set from the Keithley Configuration Utility (KCon) when the switching matrix is added to the system configuration. See Using KCon to add a switching matrix to the system (on page 2-26).

Row-column or instrument card settings

You select the scheme for interconnections between the instruments, the switching-matrix rows and columns, and the test system (prober or test fixture). You can select:

- **Row-Column:** Connect instruments to rows and prober or test fixture to columns.
- **Instrument Card:** Both instruments and prober or test fixture are connected to columns. Matrix rows are not used.

The row-column setting is the simplest connection scheme. In this scheme, instruments are connected to the switching-matrix rows. The prober/test fixture pins or the device under test (DUT) are connected to the switching-matrix columns (see Switching matrix control (on page 2-18) and 4200A-SCS signal paths (on page 2-19)).

When you set up a matrix, you also select the sense. You can select:

- **Local sense:** 2-wire connections. Connections are only to instrument FORCE terminals.
- **Remote sense:** 4-wire connections. Connections are to both instrument FORCE and SENSE terminals.

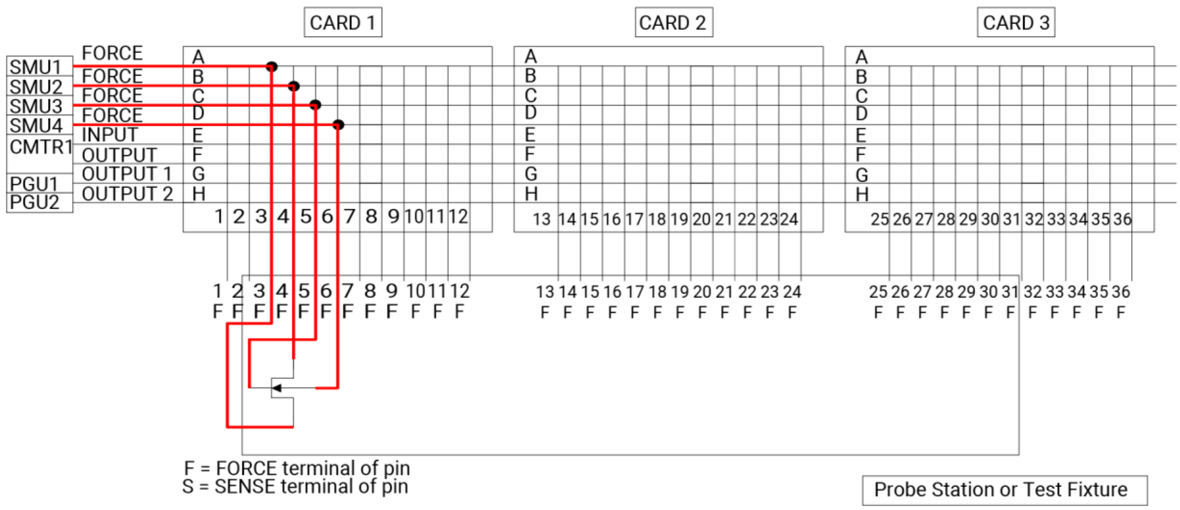
For more information regarding local and remote sense, refer to “Remote sensing” in the *Model 4200A-SCS Source-Measure Unit (SMU) User's Manual*.

Row-column scheme

The row-column setting is the simplest connection scheme. In this scheme, instruments are connected to the switching-matrix rows. The prober/test fixture pins or the device under test (DUT) are connected to the switching-matrix columns (see Switching matrix control (on page 2-18) and 4200A-SCS signal paths (on page 2-19)).

Instrument signals can route to prober/test-fixture pins through only one matrix card, as shown in the following figure. However, the row-column scheme limits the number of external instruments. If the instrumentation requirements exceed eight paths (rows), you must use the instrument card configuration.

Figure 11: Row-Column, Local Sense Connection Scheme example



Instrument card scheme for local sense

Use local sense when the measurement-pathway resistance is small and the associated voltage errors are negligible. The measurement pathway is comprised of the following conductors, connected in series:

- The cables used to connect the instruments to the matrix
- The internal matrix-card signal path
- The cables used to connect the matrix to the prober or test fixture

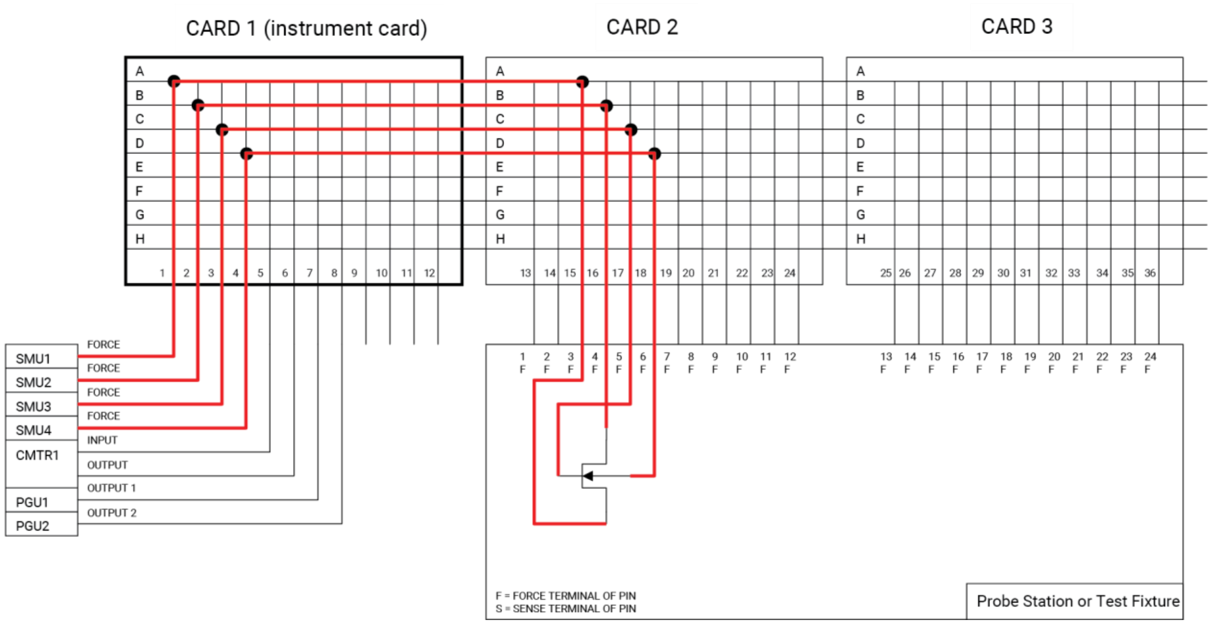
Current flowing through the measurement pathway creates a voltage drop (an error voltage) that is directly proportional to the pathway resistance. This error voltage is present in all local sense voltage measurements.

When local sense is selected, only the connection paths specified by the connected action are completed. For example, in the figure in Switching matrix control (on page 2-18), the specified connection paths would be:

- SMU2, 6 (connect SMU2 to Pin 6)
- GNDU, 3 (connect GNDU to Pin 3)

For the instrument card scheme, both the instrumentation and the prober/text-fixture pins or DUT are connected to switching-matrix columns. No external connections are made to matrix rows. In this configuration, two switching relays are closed to complete a path from an instrument to a DUT. Instrument signals route to the prober/test-fixture pins through two or more matrix cards, as shown in the following figure. This connection scheme can support large systems with numerous instruments by removing the eight-row instrument connection limitation.

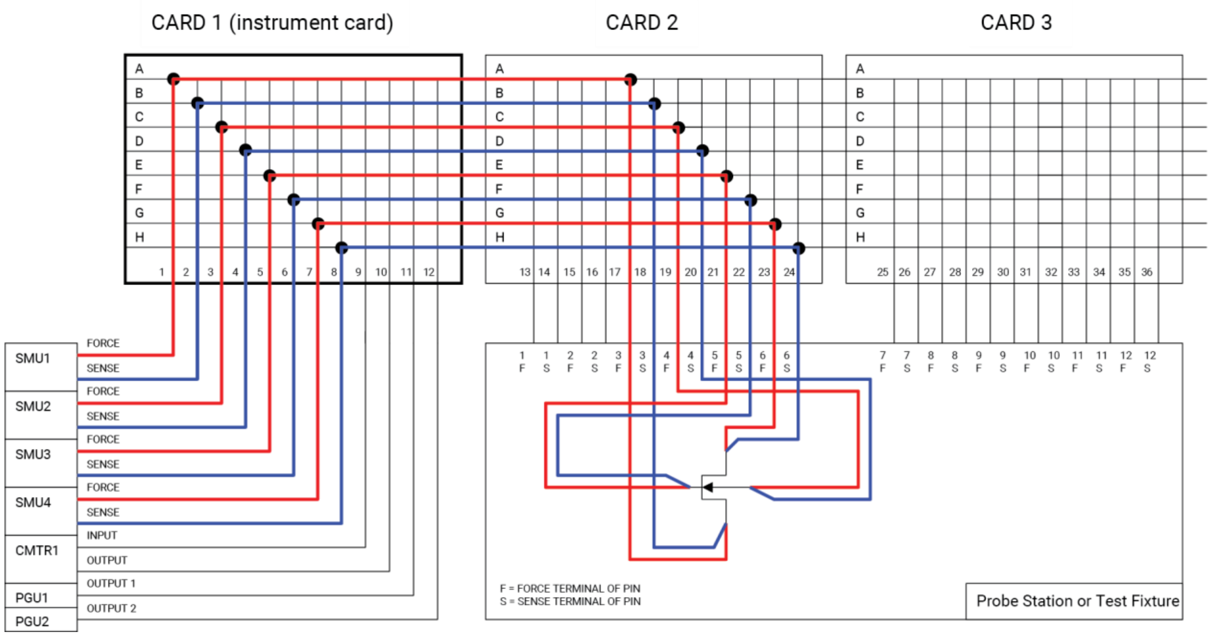
Figure 12: Instrument Card, Local Sense Connection Scheme example



Instrument card scheme for remote sense

Use remote sense to eliminate the effects of measurement pathway resistance. The following figure illustrates the use of remote sense in an instrument card configuration. Note that remote sense requires twice as many measurement pathways. The FORCE pathways (in red) are the current-carrying pathways, and the SENSE pathways (in blue) are the measurement pathways.

Figure 13: Instrument Card, Remote Sense Connection Scheme example



When remote sense is selected, rows and columns are paired together as shown in the following table.

Row A paired with row B	Column 1 paired with Column 2
Row C paired with row D	Column 3 paired with Column 4
Row E paired with row F	Column 5 paired with Column 6
Row G paired with row H	Column 7 paired with Column 8
	Column 9 paired with Column 10
	Column 11 paired with Column 12

When you specify a connection path in the `connect` action, the paired connection path is also completed. For example, in the figure in 4200A-SCS signal paths (on page 2-19), the specified connection paths would be:

- SMU1, 4 (connect SMU1 to Pin 4)
- GNDU, 3 (connect GNDU to Pin 3)

Switching matrix control

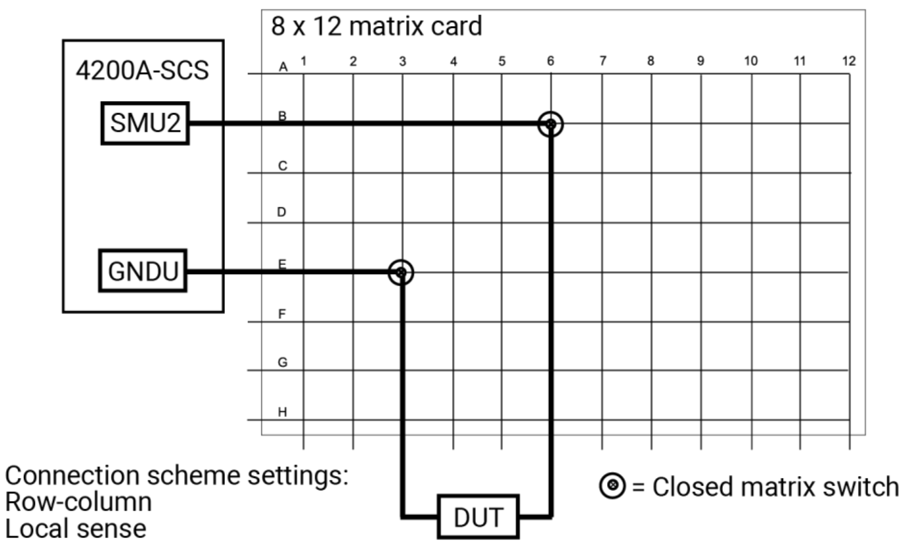
To control switching, you can use the connect action in the ivcvswitch project. You can also use the ConnectPins user module in the Matrixulib user library.

The connect action uses the ConnectPins user module to control a switching matrix. You specify the instrument terminal and pin pairs. For example, for the row-column connection scheme shown in the following figure, you set the parameters:

- TermIDStr2 to SMU2 and Pin2 to 6, which connects SMU2 to pin 6.
- TermIDStr8 to GNDU and Pin8 to 3, which connects GNDU (ground unit) to pin 3.

A matrix control example using the ConnectPins user module is provided in Switching matrix control example (on page 2-34). Detailed information for ConnectPins is provided in Matrixulib user library (on page 2-35).

Figure 14: Row-column connection scheme



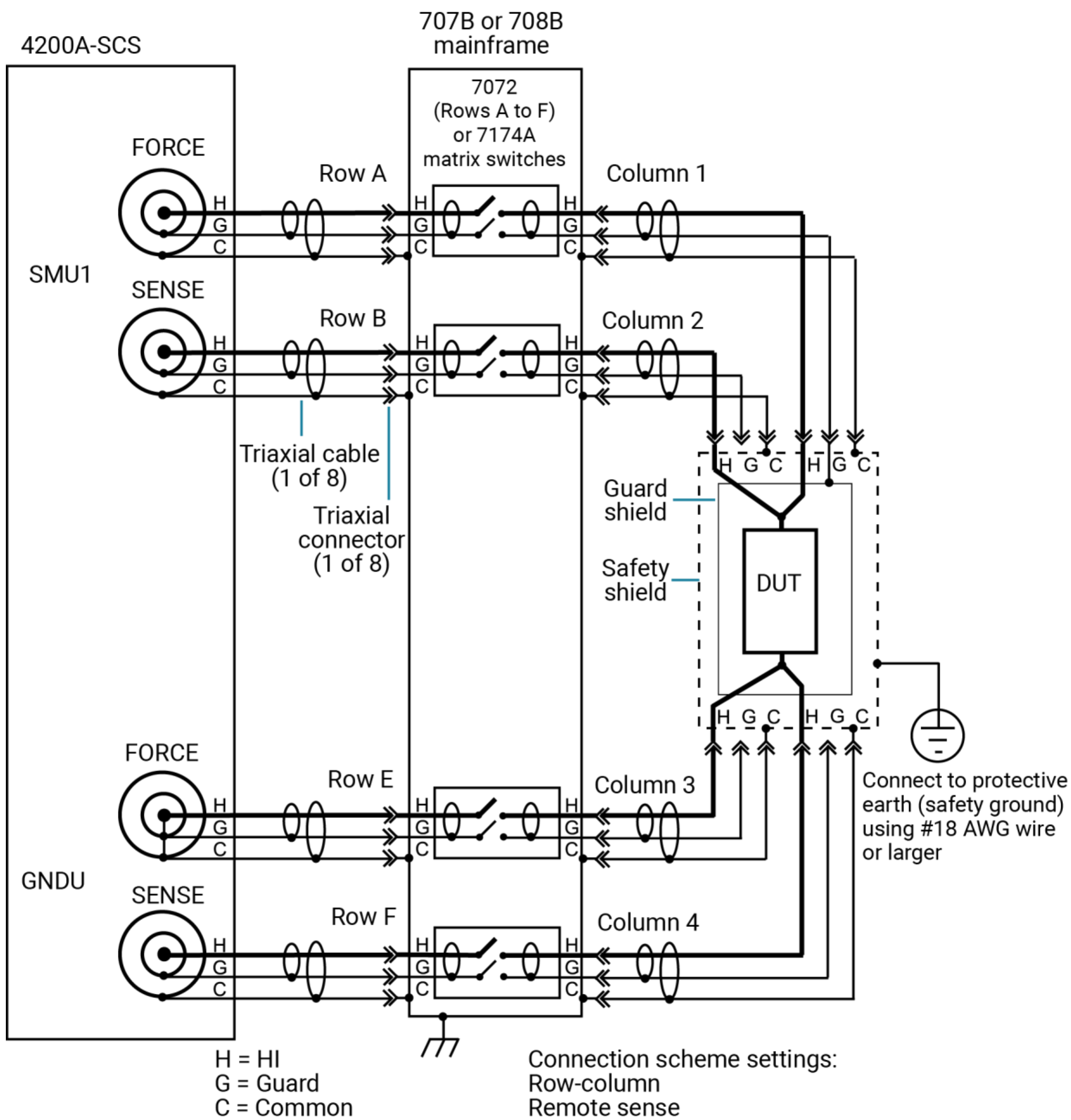
Signal paths to a DUT

The following figures show signal path examples from the various test instruments through the matrix switches to a DUT.

4200A-SCS signal paths

The following figure shows remote sensing (4-wire) signal paths through a matrix card using two-pole switching. Two-pole switching is provided by the 7174A and 7072 (rows A through F).

Figure 15: 4200A-SCS signal paths through a two-pole matrix card using remote sensing



Sense setting

To make the connections shown in 4200A-SCS signal paths (on page 2-19), you must select remote sensing.

When remote sensing is selected, the rows and columns are paired together as follows:

Row A (force) paired with row B (sense)	Column 1 (force) paired with column 2 (sense)
Row E (force) paired with row F (sense)	Column 3 (force) paired with column 4 (sense)

When the FORCE matrix switches are closed by the `ConnectPins` user module, the SENSE matrix switches are also closed.

For local sensing (2-wire), the connections from the SENSE terminals of the 4200A-SCS are not used.

NOTE

For more information regarding local and remote sense, refer to "Remote sensing" in the *Model 4200A-SCS Source-Measure Unit (SMU) User's Manual*.

Connection setting

The row-column setting must be used when connecting instrumentation to matrix rows, as shown in 4200A-SCS signal paths (on page 2-19).

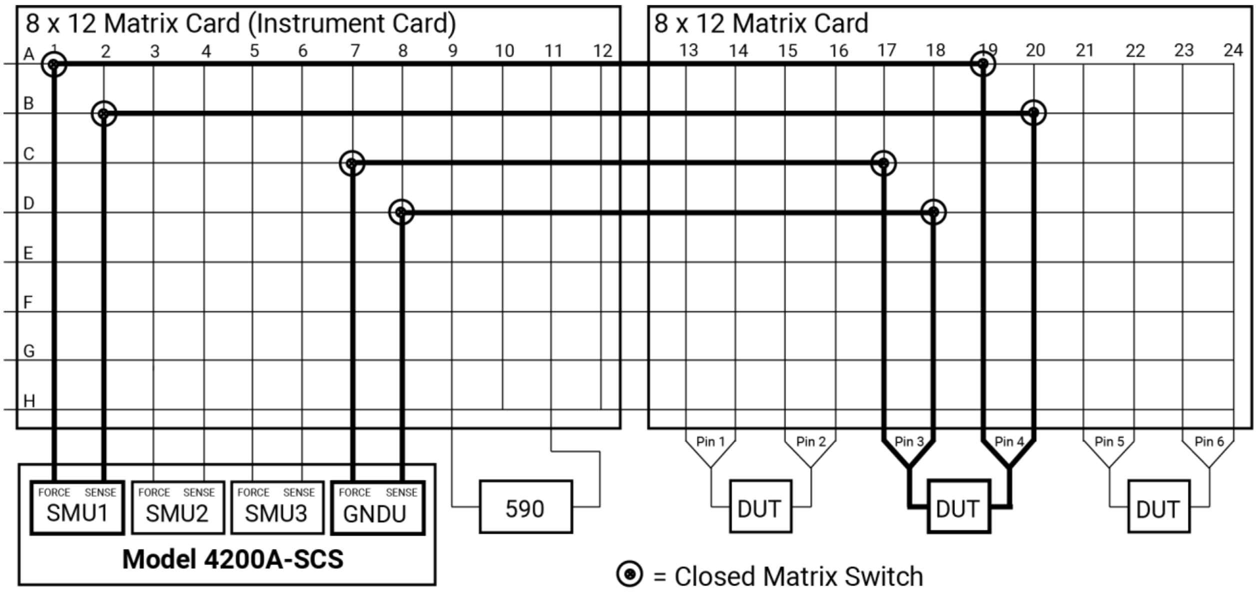
The maximum number of rows available to the test system is eight. If instrumentation needs more than eight pathways, they must be connected to matrix columns, and the instrument card setting must be used.

The following figure shows a test system with both the instruments and the DUT connected to matrix columns.

NOTE

See Connection scheme settings (on page 2-14) for details on the row-column and instrument card settings. The connection scheme settings shown in this figure are Instrument Card and Remote Sense.

Figure 16: Instrument card connection scheme



NOTE

The 4200A-SCS automatically selects the first available rows to make connections to the DUT. In this example, rows A through D are the first available rows.

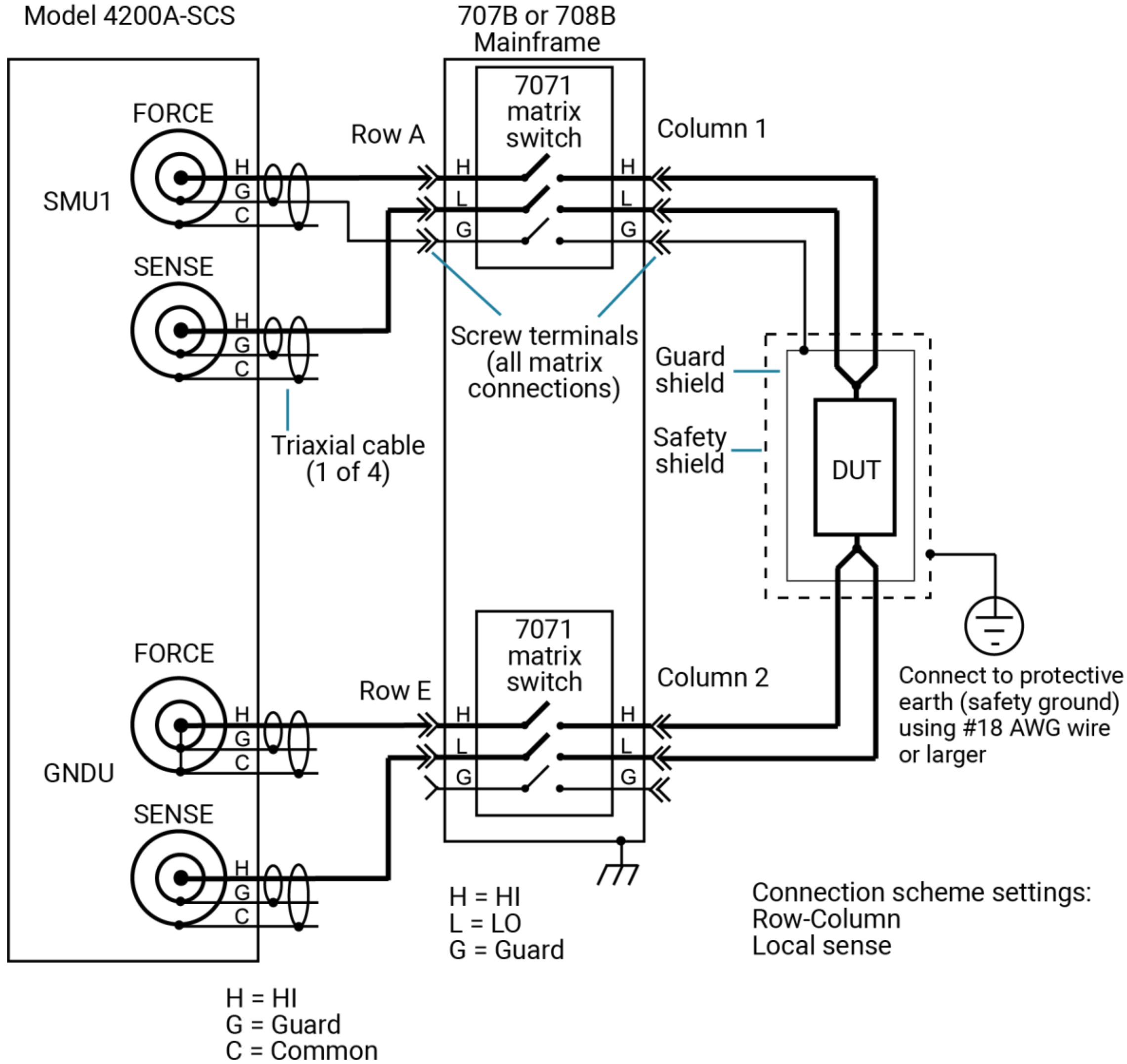
The following shows 4200A-SCS signal paths through a 3-pole 7071 matrix card using remote sensing. Note that for this configuration, each FORCE and SENSE connector does not use a separate path (row). Unlike the configuration shown in 4200A-SCS signal paths (on page 2-19), each FORCE/SENSE connector pair is routed through a single 3-pole matrix switch. Since row pairing is not required, the local sense setting must be used.

For two-wire local sense connections, do not use the SENSE connectors of the 4200A-SCS.

WARNING

To avoid high voltage exposure that could result in personal injury or death, whenever the interlock of the 4200A-SCS is asserted, the FORCE and GUARD terminals of the SMUs and preamplifier should be considered to be at high voltage, even if they are programmed to a nonhazardous voltage current.

Figure 17: 4200A-SCS signal paths through a 3-pole matrix card using remote sensing



C-V Analyzer signal paths

The following figures show local sense C-V Analyzer signal paths through rows B and H of a 7072 matrix card. A C-V analyzer can be used with any of the three matrix card types; however, rows G and H of the 7072 are optimized for C-V measurements.

Figure 18: 590 signal paths through 7072 matrix card using local sensing

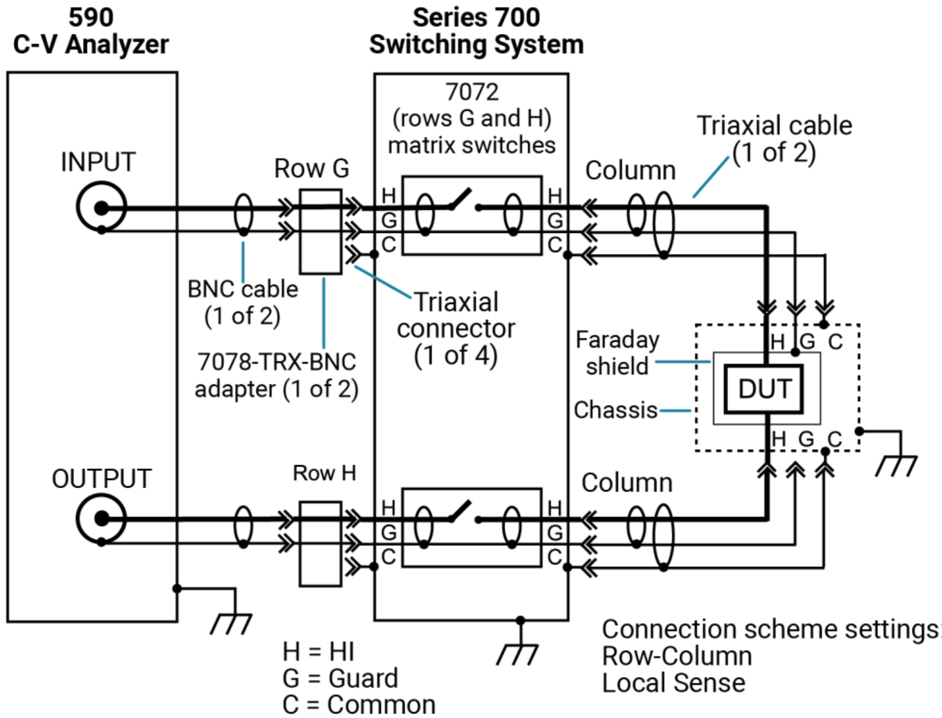
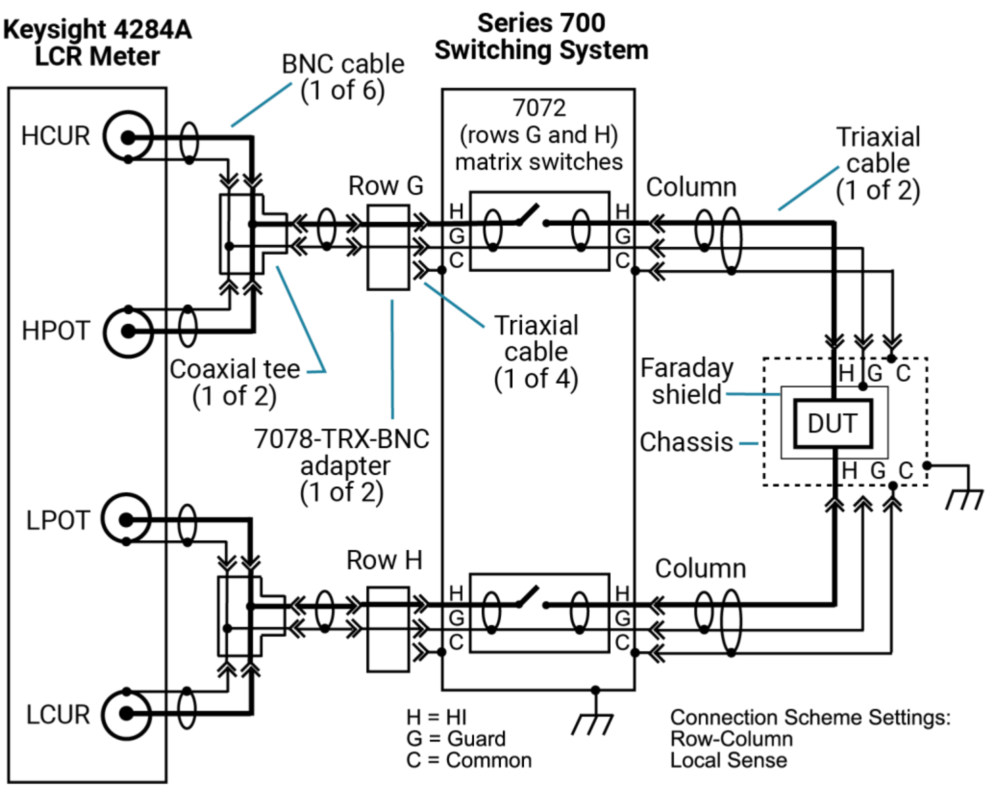
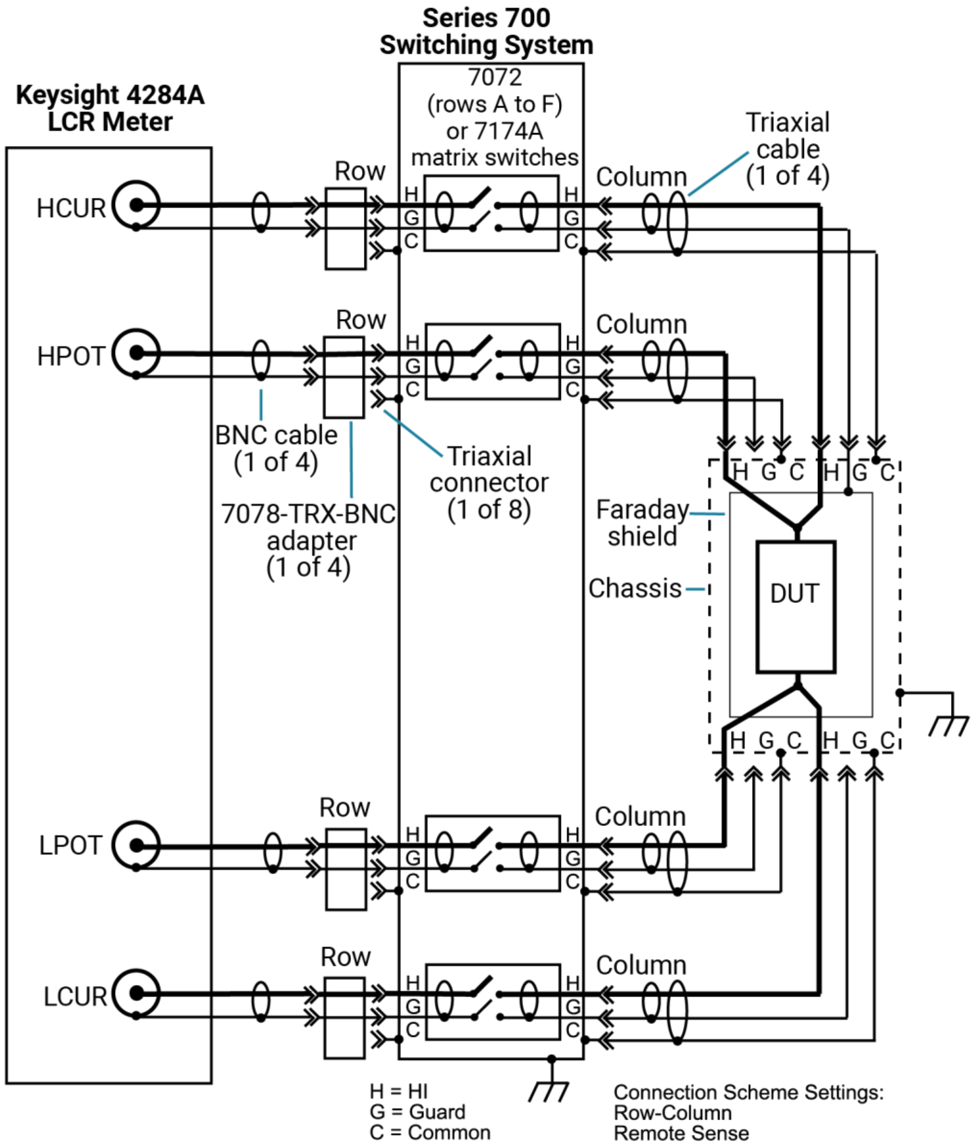


Figure 19: Keysight Model 4980A signal paths through 7072 matrix card using local sensing



The following figure shows the remote sense signal paths for the Keysight Model 4980A LCR meter through a 2-pole matrix card. Since row pairing is required, the remote sense setting must be used.

Figure 20: Keysight Model 4980A signal paths through a two-pole matrix card using remote sensing

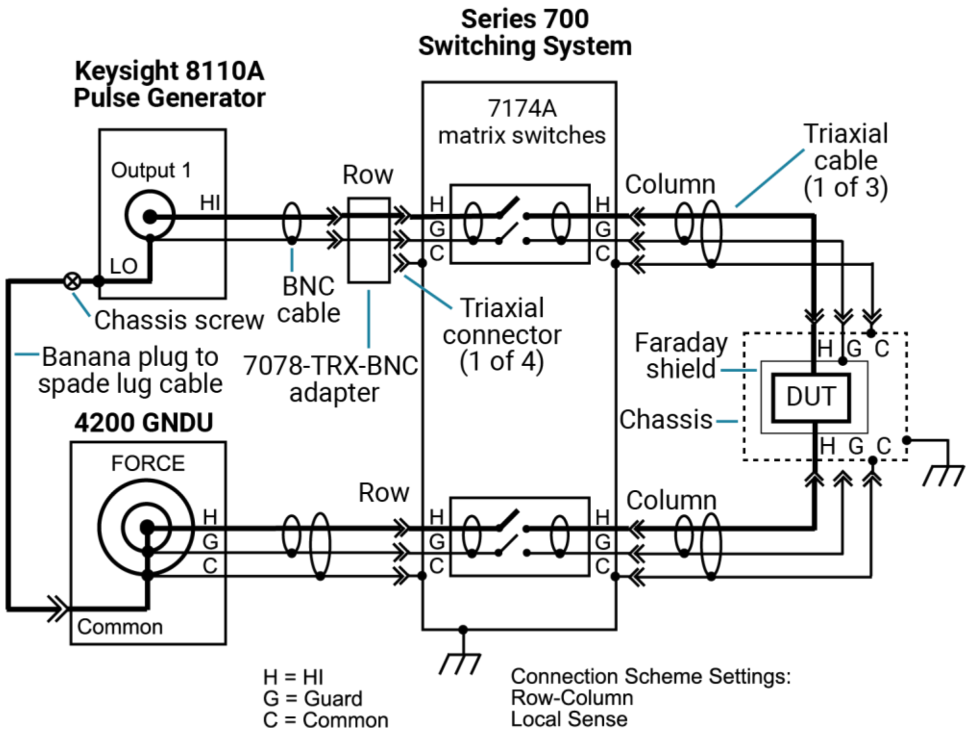


Keysight Model 8110A pulse generator signal path

The following figure shows the HI signal path through the 7174A matrix card. However, the pulse generator can also be used with other matrix card types.

The pulse generator LO is not routed through the matrix card. A separate external return path is required. The chassis of the pulse generator is output LO. As shown in the following figure, use a banana plug cable that is terminated with a spade lug on one end. Connect the banana plug end of the cable to the Common banana jack of the GNDU, and attach the spade lug end to a chassis screw on the pulse generator.

Figure 21: Keysight Model 8110A signal path through a 7174A matrix card



Use KCon to add a switching matrix to the system

You use Keithley Configuration Utility (KCon) to manage the configuration of all instrumentation controlled by the 4200A-SCS software. To use the 4200A-SCS to control a switching matrix, you must add the switching matrix to the system configuration using KCon.

If you are testing discrete device under test (DUTs), you must use the switching matrix with a test fixture. If you are testing a wafer, you must use the switching matrix with a probe station. The test fixture or probe station is also added to the system configuration using KCon.

You specify physical instrument-to-card and card-to-prober or fixture connections in KCon.

These and other KCon switching matrix settings result in simplified matrix connections. Initially, you need to:

- Add the test fixture or probe station.
- Configure the Instrument Connection Scheme and Switch Cards areas.
- Specify the physical instrument-to-card and card-to-prober/fixture connections.
- Physically make the specified instrument-to-card and card-to-prober/fixture connections.

After the initial setup, you can specify instrument-to-prober/fixture connections by specifying the corresponding terminal and prober/fixture pins in a Clarius user test module (UTM). You do not need to specify matrix cross points. The 4200A-SCS automatically routes the signals through the matrix.

For additional detail on KCon, refer to the *Model 4200A-SCS Setup and Maintenance User's Manual*.

NOTE

If you are using a 707B or 708B Switching System, you must use the control panel on the front of your switching system to enable DDC and change the command set to 70XB by following these steps:

1. Select **Menu**.
2. Select **DDC**.
3. Select **Enable**.
4. Select **70XB-VERSION**.

This allows the switching system to be controlled by the 4200A-SCS.

Step 1. Exit Clarius and open KCon

To exit Clarius and open KCon:

1. Exit Clarius.
2. On the Windows desktop, select the **KCon** icon.

Step 2. Add a test fixture or probe station

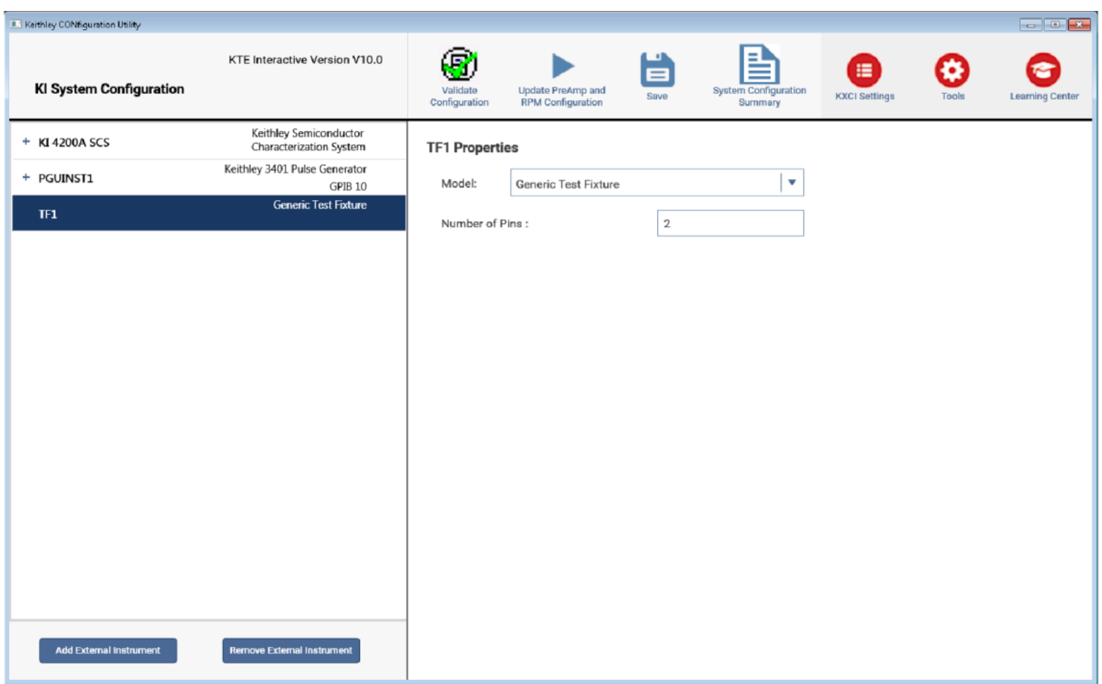
You must use a test fixture or a probe station with the switching matrix. However, both cannot be in the system configuration together. If you need to remove a component, refer to "Remove an external instrument" in *Model 4200A-SCS Parameter Analyzer Setup and Maintenance*.

Add a test fixture

To add a test fixture to the system configuration:

1. Select **Add External Instrument**.
2. Select **Test Fixture**.
3. Select **OK**.
4. In the System Configuration list, select the test fixture (prefix is TF).

Figure 22: Add test fixture



5. From the **Model** list, select the appropriate test fixture.
6. Enter the number of pins. You can enter 2 to 72 pins.

NOTE

The number of pins defined in the test fixture properties determines the pins that are available to assign to a switching matrix card column. Make sure the number of pins assigned is appropriate for your system.

Add a probe station

Supported probe stations include:

- Fake Prober
- Manual Prober
- Micromanipulator 8860 Prober
- Micromanipulator P200L Prober
- Cascade Microtech (Karl Suss) PA200 Prober
- Formfactor Cascade Summit-12000 Prober
- Signatone CM500 (WL250) Prober
- MPI TS2000, TS2000-DP, TS2000-HP, TS2000-SE, TS3000, and TS3000-SE Probers

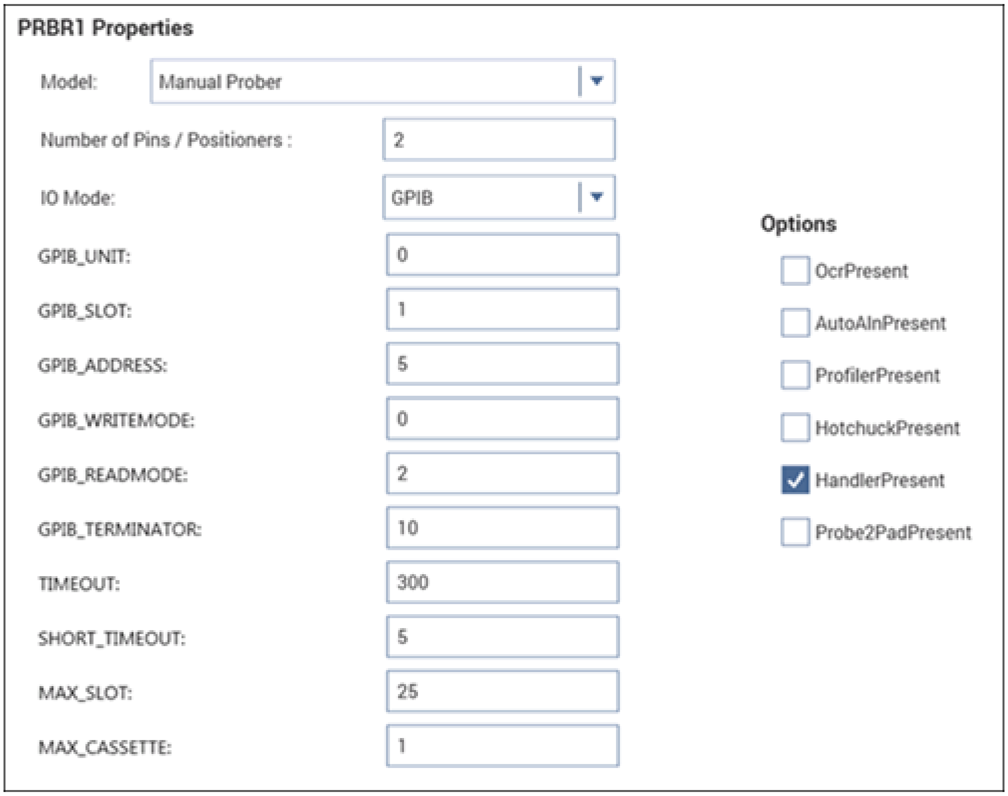
NOTE

Contact Keithley for the most up-to-date list of supported probers. If you are using an unsupported prober, you must create a user library and module to control it.

To add a probe station to the system configuration:

1. Select **Add External Instrument**.
2. Select **Probe Station**.
3. Select **OK**.
4. In the System Configuration list, select the probe station. The Properties are displayed.

Figure 23: Probe station properties



5. From the **Model** list, select the prober.
6. Enter the **Number of Pins / Positioners**.
7. Select the options that are appropriate for your prober.

NOTE

The number of pins defined in the probe station properties determines the pins that are available to assign to a switching matrix card column. Make sure the number of pins assigned is appropriate for your system.

Step 3. Add switching system mainframe

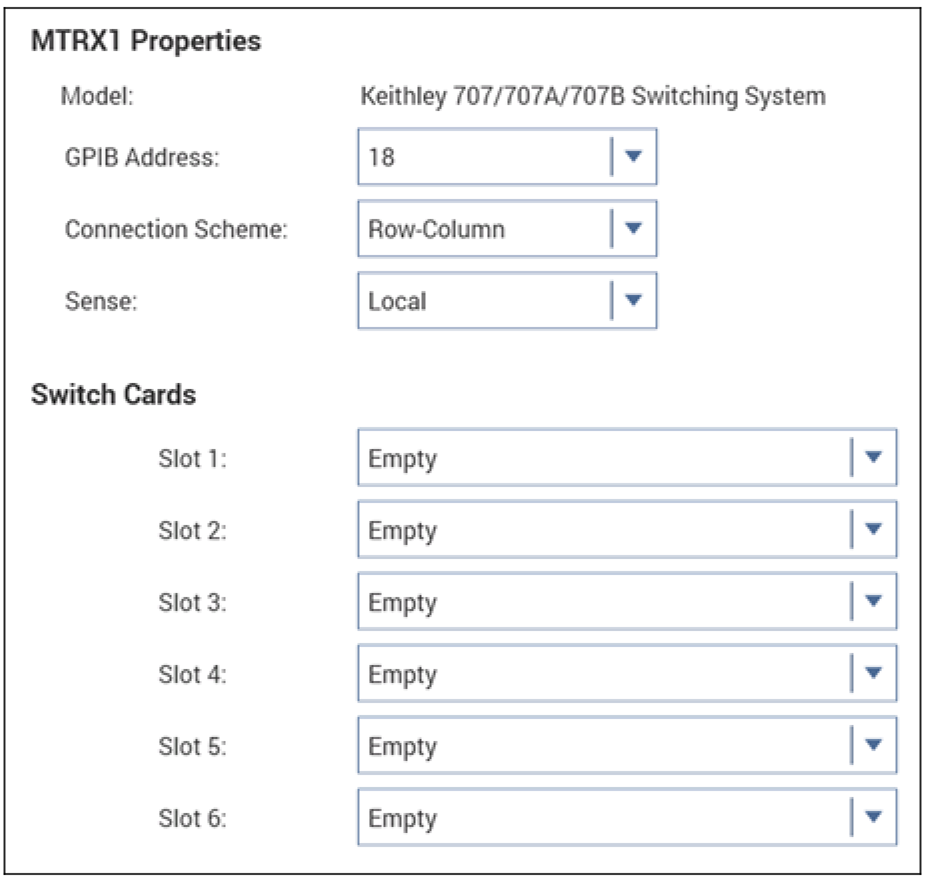
NOTE

The default GPIB address for the Series 700B Switching systems is 16.

To add a switching system mainframe:

1. Select **Add External Instrument**.
2. Select the **Keithley 707/707A/707B Switching Matrix** or **Keithley 708/708A/708B Switching Matrix**.
3. Select **OK**.
4. In the System Configuration list, select the switching matrix. The properties are displayed. The following figure shows the properties for the 707/707A/707B. If the 708/708A/708B mainframe is selected, there is only one switch card slot.

Figure 24: KCon MTRX1 Properties



Step 4. Set GPIB address

The GPIB address setting in the properties must match the actual GPIB address of the mainframe. The address for the switching system mainframe is briefly displayed during its power-on sequence.

To set the GPIB address:

1. Select the GPIB Address from the list. Addresses that are in use are displayed with asterisks (*) next to them. The range of addresses is 0 to 30 (GPIB address 31 is reserved as the 4200A-SCS controller address). If the selected GPIB address conflicts with the GPIB address of another system component, a red exclamation-point symbol (!) is displayed next to the selected address.
2. Select **Save** to save the change.

NOTE

You can programmatically read the GPIB address and other instrument properties from the system configuration using the LPT library `getinstattr` function. Proper use of `getinstattr` allows you to develop user libraries that are independent of the configuration. For more information, refer to *Model 4200A-SCS KULT and KULT Extension Programming* (4200A-KULT-907-01).

Step 5. Configure the instrument connection scheme

To configure the instrument connection scheme:

1. Select the **Connection Scheme** from the list:
   - If you are connecting the instrumentation to matrix rows and the device under test (DUT) to matrix columns, select **Row-Column**.
   - If all connections (instrumentation and DUT) are made to matrix columns only, select **Instrument Card**.
2. Select **Local Sense** or **Remote Sense**:
   - For 2-wire connections to the DUT, select **Local Sense**.
   - For 4-wire connections to the DUT, select **Remote Sense**.

Step 6. Assign switch cards to mainframe slots

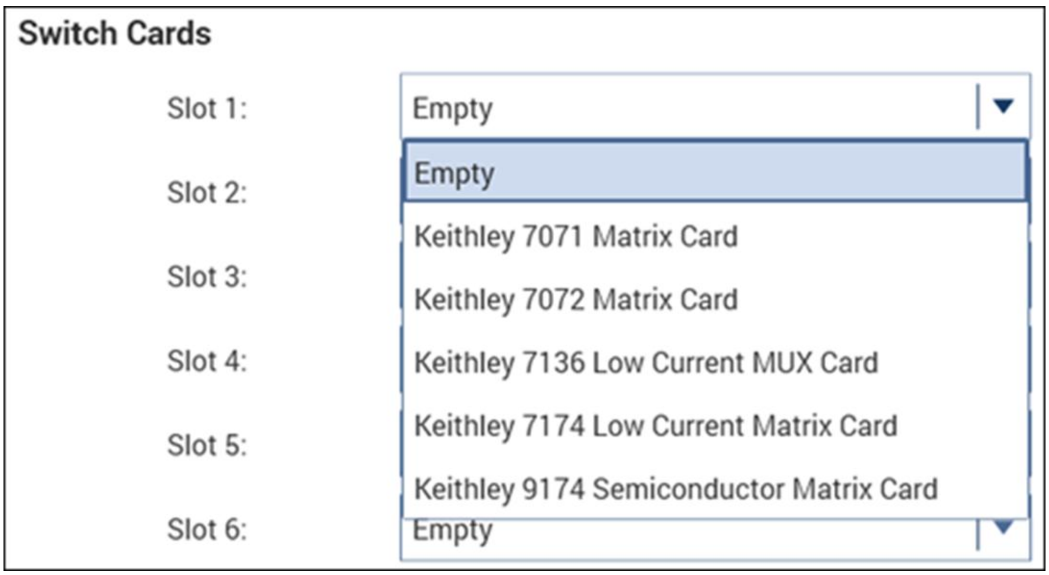
To assign switch cards to mainframe slots:

1. For each slot that contains a matrix card, select the model number of the matrix card.
2. For each slot that is empty, select **Empty**.

NOTE

You cannot mix matrix card models. For example, if you set slot 1 to Keithley 7174 Low Current Matrix Card, all other slots can only be set to the 7174 or Empty. To select a different model, you must set all slots to Empty and then make the new selection.

Figure 25: Assign switch cards to slots



Step 7. Set matrix card properties

The matrix card properties set the connections:

- Between the measurement instrumentation and the matrix card
- Between the matrix card and the test system (prober or test fixture)

NOTE

The number of pins defined in the properties for a probe station or test fixture determines the pins that are available to assign to a switching matrix card column. Make sure the number of pins assigned is appropriate for your system. Refer to Add a test fixture (on page 2-27) or Add a probe station (on page 2-28) for additional information.

To set matrix card properties:

1. In the System Configuration list, expand the switching matrix.
2. Select the card. The properties are displayed. Each row and column has a list to set the card properties. If the row-column connection scheme is selected, instruments are assigned to the rows and the test fixture pins or probe pins are assigned to the columns. If the instrument card connection scheme is selected, both instrumentation and test fixture/probe pins are assigned to columns.

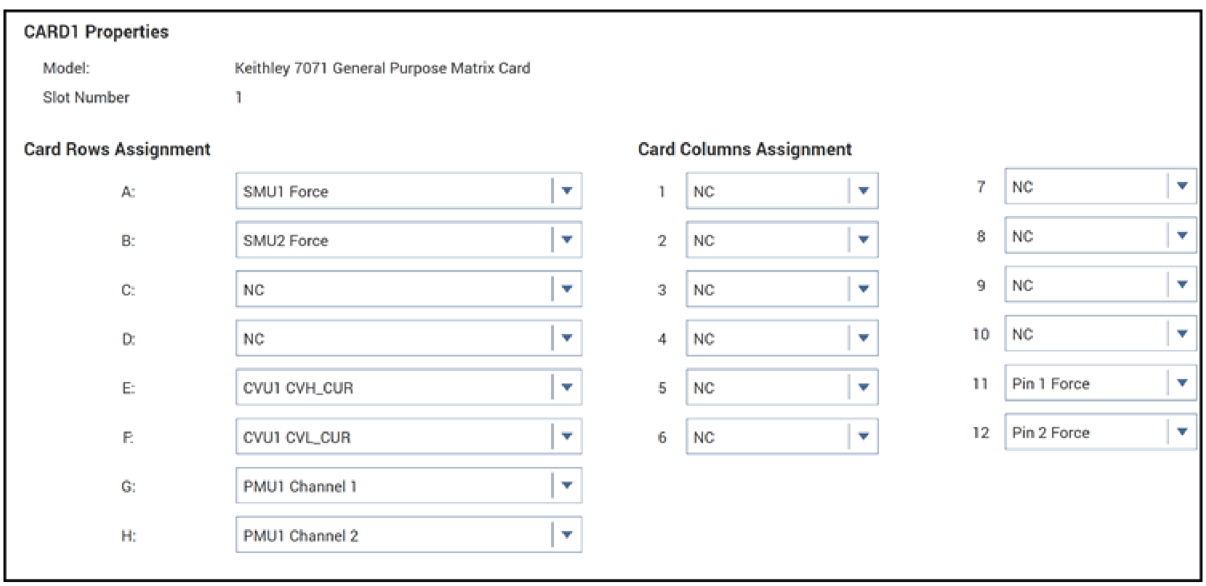
   The following figure shows the 7071 Matrix Card Properties settings that are required to support the physical connections that are shown in Row-column or instrument card settings (on page 2-14).
3. Select from the lists to connect the rows and columns to instrument terminals and prober or test fixture pins. Note that card properties must match the actual physical connections to the matrix card.

   In the following figure, the lists labeled **A** to **H** correspond to the eight rows of the Keithley matrix cards that are compatible with the Series 700 Switching System. The lists labeled **1** to **12** correspond to the 12 columns of these matrix cards.
4. Select **Validate**.

NOTE

Prober or test-fixture pins are always connected to matrix-card columns.

Figure 26: Keithley 7071 Matrix Card Properties



Step 8. Save configuration

To save the KCon configuration:

1. Select **Save**.

Step 9. Close KCon and open Clarius

To close KCon and open Clarius:

1. To close KCon, select the close button in the upper right.
2. On the Windows desktop, select the Clarius icon.

Switching matrix control example

This example demonstrates how the `connectpins` action controls a switching matrix. You modify the `connectpins` action to connect SMU2 to a DUT, as shown in the Switching matrix control (on page 2-18) figure. It assumes that the switching matrix is set for row-column connections with local sense selected. It also assumes that the matrix card properties are set as shown in Switching matrix control (on page 2-18).

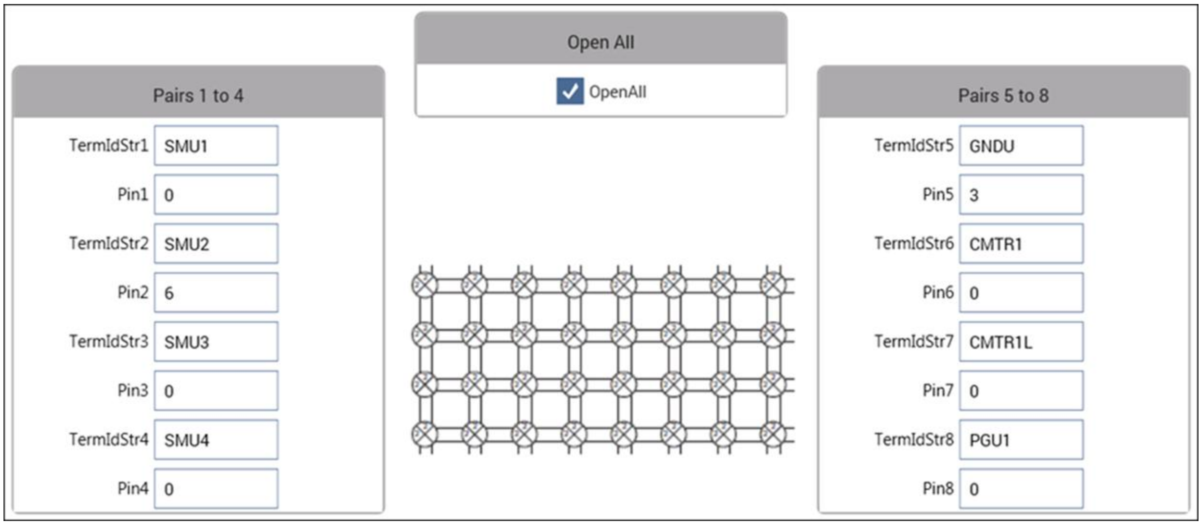
The `connectpins` action is based on the `ConnectPins` user module. Details on `ConnectPins` is provided in Matrixulib user library (on page 2-35).

Set up and run a switching matrix in Clarius

To set up and run the `connectpins` action:

1. Choose **Select**.
2. Select **Actions**.
3. Search for **connectpins**.
4. Select the `connectpins` action.
5. Select **Add**.
6. In the project tree, select the **connectpins** action.
7. Select **Configure**. The parameter settings are displayed, as shown in the following figure.

Figure 27: connectpins settings



8. Set Pin2 to **6**. This connects SMU2 to point 6.
9. Select **OpenAll** to open all matrix card switches.
10. Set Pin5 to **3**. This connects GNDU to pin 5.
11. Leave all other pin settings at 0 to indicate that no connection will be made.
12. Select **Run**. The 4200A-SCS connects to the DUT as shown in Switching matrix control (on page 2-18).

Matrixulib user library

The `Matrixulib` connects instrument terminals to output pins using a Keithley Series 700 Switching System. It is for use with switching systems that are configured as a general purpose, low current, or ultra-low current matrix.

Matrixulib user module

User module	Description
`ConnectPins`	Allows you to control your switching matrix.

ConnectPins user module

The `ConnectPins` module allows you to control your switching matrix.

Usage

```
status = ConnectPins(int OpenAll, char *TermIdStr1, int Pin1, char *TermIdStr2, int
    Pin2, char *TermIdStr3, int Pin3, char *TermIdStr4, int Pin4, char *TermIdStr5,
    int Pin5, char *TermIdStr6, int Pin6, char *TermIdStr7, int Pin7, char
    *TermIdStr8, int Pin8);
```

status	Returned values; see **Details**
OpenAll	Controls if the switching matrix is cleared before making any new connections: ■ Clear all previous connections: `1` ■ Leave previous connections intact: `0`
TermIdStr1 *TermIdStr2* *TermIdStr3* *TermIdStr4* *TermIdStr5* *TermIdStr6* *TermIdStr7* *TermIdStr8*	Terminal identification string; refers to an instrument as defined by `TermIdStr8` in KCon; valid inputs (configuration dependent) are: `SMUn`, `CMTRn`, `CMTRnL`, `PGUn`, `GPIn`, `GPInL`, `GNDU` (where *n* is a number from 1 through 8)
Pin1 *Pin2* *Pin3* *Pin4* *Pin5* *Pin6* *Pin7* *Pin8*	The DUT pin number (configuration dependent) to which the instrument will be attached; if a number less than 1 is specified, no connection is made; valid inputs: –1 to 72

Details

This user module allows you to control a switching matrix. The default input parameters are shown in the following figure. Typically, OpenAll (line 1) is set to 1 to initially open all connections. If set to 0, the present connections are not affected.

The rest of the input parameters are structured as terminal/pin pairs. Each terminal/pin pair specifies the signal path through the matrix. For example, if the specified pin parameter for SMU1 is 4, then SMU4 will connect to pin 4 of the test fixture or prober when the UTM is run. The pin parameter value 0 (or –1) indicates that no connection will be made.

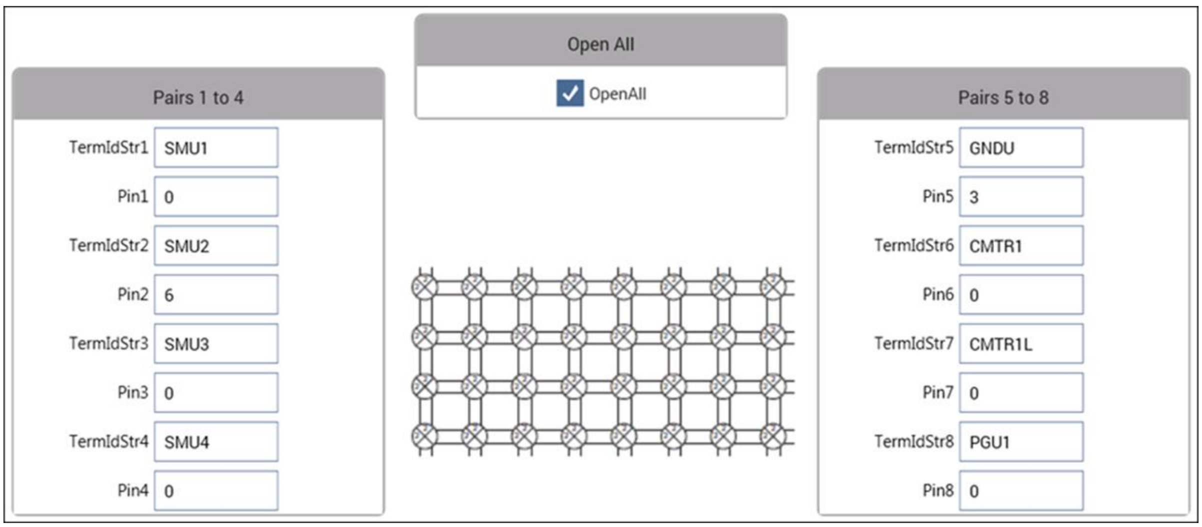
Terminal ID: Terminal identification for the most common components used in the system configuration are as follows:

- SMU1 to SMU4: These are the signal HI terminals for the four SMUs.
- GNDU: This is common terminal for the Ground Unit of the 4200A-SCS.
- CMTR1: This is used for a C-V Analyzer. For the 590, it is the OUTPUT terminal. For the Keysight Model 4980A, it is the HCUR terminal.
- CMTR1L: This is also used for a C-V Analyzer. For the 590, it is the INPUT terminal. For the Keysight Model 4980A, it is the LCUR terminal.
- PGU1: This is output HI for the Keysight Model 8110A Pulse Generator.

NOTE

A test example demonstrates how this user module controls the switching matrix (see Switching matrix control example (on page 2-34)).

Figure 28: connectpins settings



You can connect the instrument terminals to one or more DUT pins. If the DUT pin number is less than 1, then that connection is ignored (not performed), otherwise the specified instrument is connected to the desired DUT pin. If you wish to connect an instrument to more than one DUT pin, you may specify that instrument terminal again in the parameter list.

If the `OpenAll` parameter is less than one, then the matrix is not cleared before making connections; if `OpenAll` is 1, then all previous matrix connections are cleared before making the new connections.

Returned values are placed in the Analyze sheet and can be:

- `0 OK`.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist. This generally means that there is no instrument with the specified ID in your configuration.
- `-10001 (INVAL_PIN_SPEC)`: An invalid DUT pin number was specified.
- `-10003 (NO_SWITCH_MATRIX)`: No switching matrix was found.
- `-10004 (NO_MATRIX_CARDS)`: No matrix cards were found.

Example

To connect SMU1 to pin 7, SMU2 to pin 8, SMU3 to pin 12, SMU4 to pin 1, ground pin 15, connect the pulse generator to pin 13, connect the CMTR to pins 9 and 10, and clear the previous connections:

```
ConnectPins(1, SMU1, 7, SMU2, 8, SMU3, 12, SMU4, 1, GNDU, 15 PGU1, 13, CMTR1,
    9, CMTR1L, 10)
```

Also see

None

Section 3

Configure and use a Series 700 Switching System

In this section:

Introduction

This section describes how to configure a Keithley Series 700 Switching System (707, 707A, 707B, 708, 708A, or 708B) in the Keithley Configuration Utility (KCon). You can then use the system to connect any instrument terminal to any test system pin without changing connections. You can also create a new project for an n-channel MOSFET transistor and use the project to make both I-V and C-V measurements using the switching system.

Switching systems are controlled by the 4200A-SCS using the GPIB bus. Use a shielded GPIB cable to connect your switching system to the 4200A-SCS. Once the switching system and test fixture have been defined in KCon, you use Clarius to set up the connections and automatically connect the instruments to the test system pins using the switching system.

In Clarius, the `connectpins` action from the Action Library is used to control switching systems. This action controls the opening and closing of crosspoints in a switching system so that you can connect any row of the matrix card to any (or multiple) columns of the matrix card. The `connectpins` action is added to the project and runs twice in this example. Each run establishes new connection settings.

Equipment required

- One 4200A-SCS with the following instruments:
  - Three 4200-SMUs, 4201-SMUs, 4210-SMUs, or 4211-SMUs
  - One 4210-CVU or 4215-CVU
- Eight 4200-MTRX-X triaxial cables or 4200-TRX-X cables if using preamplifiers
- Four CA-447B SMA cables (supplied with the CVU)
- Four CS-1247 SMA socket to BNC plug adapters (supplied with the CVU)
- Two CS-701A BNC Tee adapters (socket, plug, socket)
- Two 7078-TRX-BNC BNC socket to triaxial plug adapters
- One Series 700 Switching System with a 7072 8x12 Matrix Card
- One shielded four-terminal test fixture with triaxial inputs
- One n-channel MOSFET transistor

Device connections

The next topics detail the connections from the 7072 to the n-channel MOSFET and the connections from the SMUs or CVU, and GNDU to the 7072 Matrix Card in the Series 700 Switching System.

WARNING

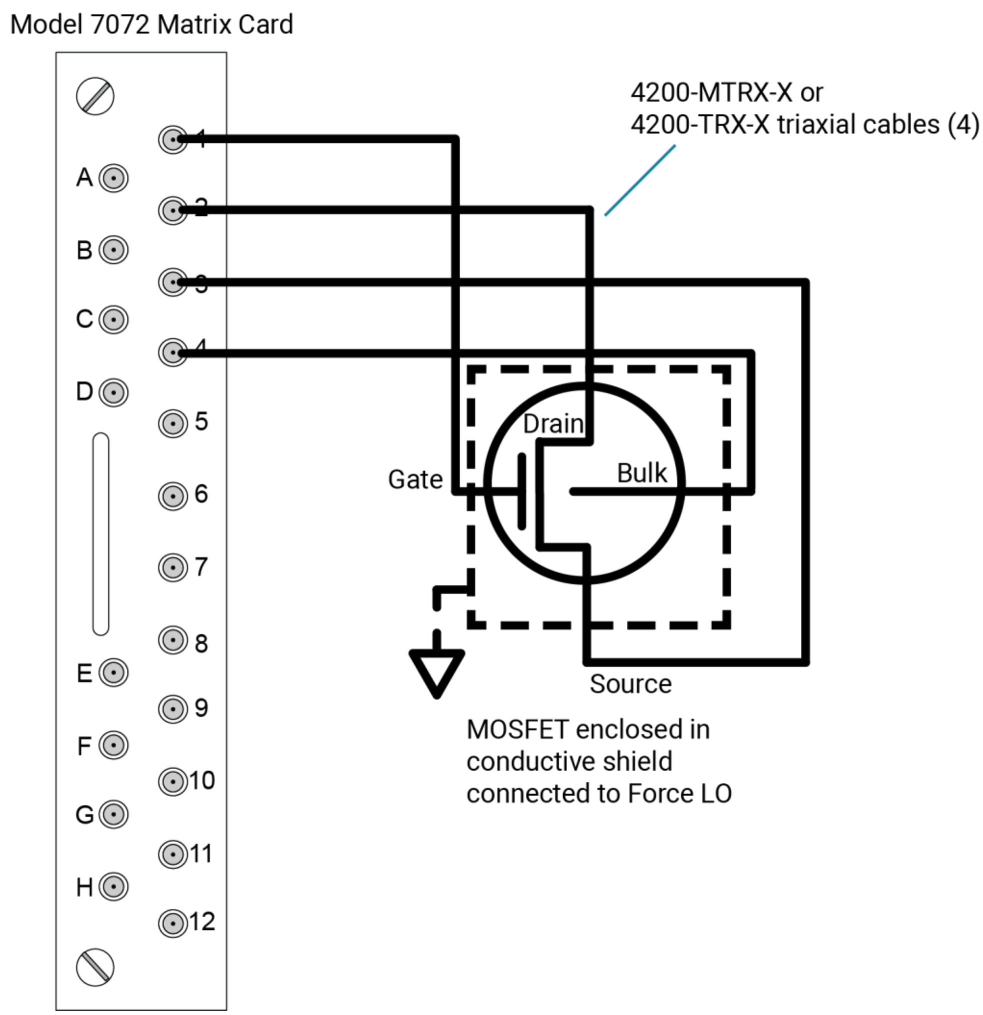
Hazardous voltages may be present on all output and guard terminals. To prevent electrical shock that could cause injury or death, never make or break connections from the instrument while the instrument is powered on. Turn off the equipment from the front panel or disconnect the main power cord from the rear of the 4200A-SCS before handling cables. Putting the equipment into an output-off state does not guarantee that the outputs are powered off if a hardware or software fault occurs.

To prevent electric shock, test connections must be configured such that the user cannot come in contact with test leads, conductors, or any device under test (DUT) that is in contact with the conductors. It is good practice to disconnect DUTs from the instrument before powering up the instrument. Safe installation requires proper shields, barriers, and grounding to prevent contact with test lead and conductors.

Connect the 7072 to the DUT

The hardware connections from the 7072 Matrix Card to the 4-terminal MOSFET DUT are shown in the following figure. Use four triaxial cables to connect to the input terminals of your test fixture. For systems without a preamplifier, use 4200-MTRX-X triaxial cables. For system with preamplifiers, use 4200-TRX-X triaxial cables.

Figure 29: Connections from the 7072 Matrix Card to the MOSFET DUT



Connect the 4200A-SCS to the 7072

This section describes connections to the 7072.

To connect the 4200A-SCS and SMUs to the 7072:

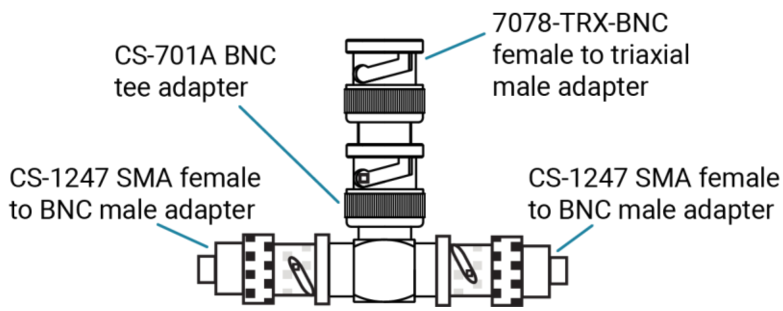
Using four 4200-MTRX-X or 4200-TRX-X triaxial cables, make the following connections:

- 4200A-SCS GNDU FORCE to 7072 input terminal E
- 42x0 SMU channel 1 Force to 7072 input terminal A
- 42x0 SMU channel 2 Force to 7072 input terminal B
- 42x0 SMU channel 3 Force to 7072 input terminal C

To connect the 4210-CVU or 4215-CVU to the 7072:

1. Using the parts in the following figure, assemble a tee adapter to connect the CVU to the 7072.

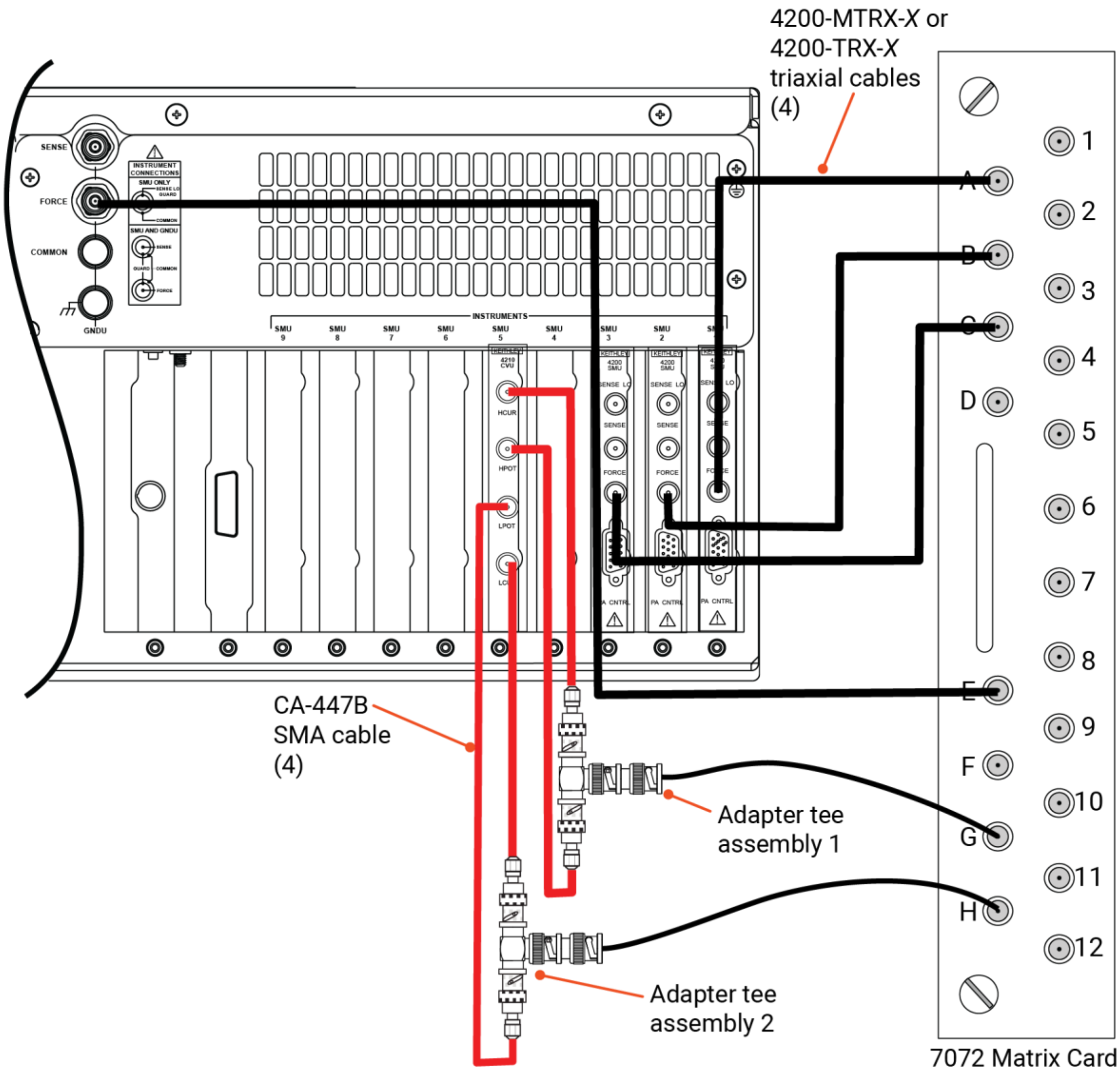
Figure 30: 4210-CVU to 7072 adapter tee assembly



2. Using four CA-447B SMA cables, make the following connections:
   - CVU HCUR to adapter tee assembly 1
   - CVU HPOT to adapter tee assembly 1
   - CVU LPOT to adapter tee assembly 2
   - CVU LCUR to adapter tee assembly 2
3. Connect adapter tee assembly 1 to input terminal G of the 7072.
4. Connect adapter tee assembly 2 to input terminal H of the 7072.

The connections are shown in the following figure.

Figure 31: 4200A-SCS to 7072 Matrix Card connections



Update the switch configuration in KCon

After completing the switch and device connections, use KCon to manage the configuration of all instrumentation controlled by the 4200A-SCS software. You use KCon to:

- Add the switching system to the 4200A-SCS configuration
- Add the test fixture to the system configuration
- Configure the test fixture
- Add a matrix card to the switching system
- Configure the matrix card connections

NOTE

If you are using a 707B or 708B Switching System, you must use the control panel on the front of your switching system to enable DDC and change the command set to 70XB by following these steps:

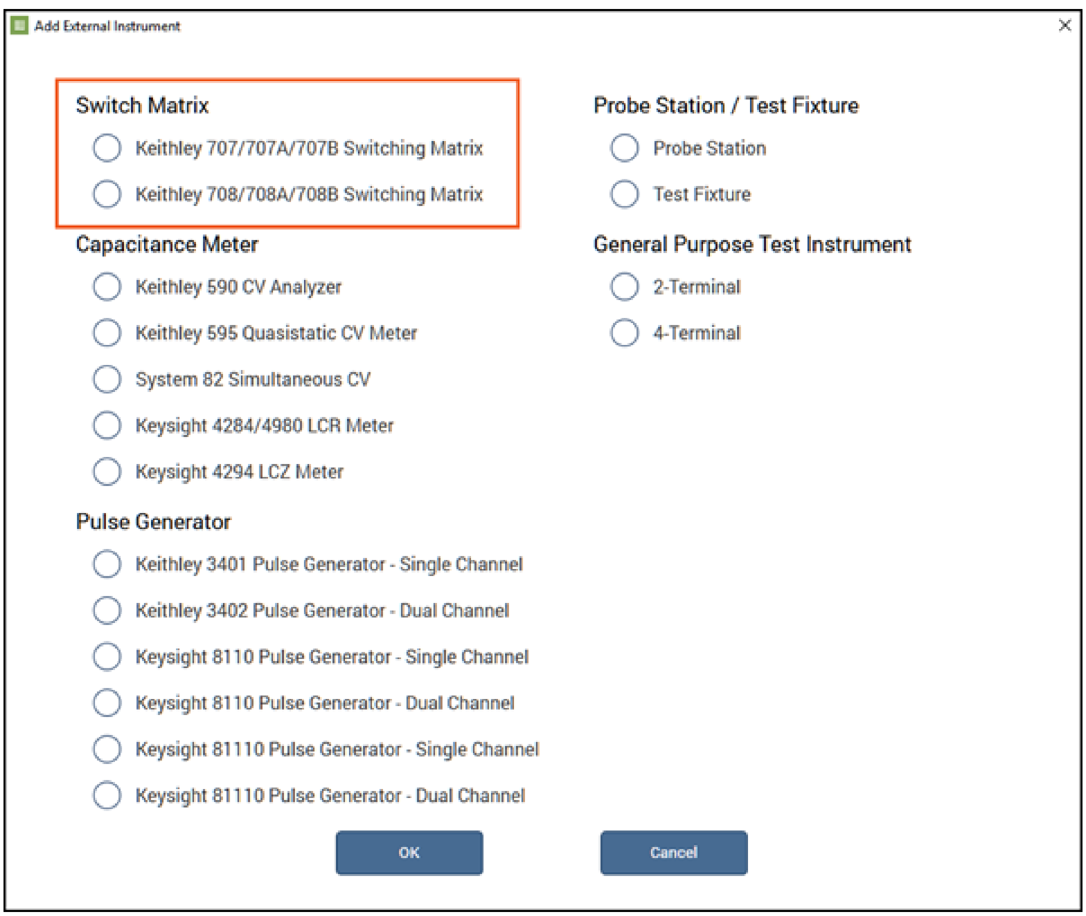
1. Select **Menu**.
2. Select **DDC**.
3. Select **Enable**.
4. Select **70XB-VERSION**.

This allows the switching system to be controlled by the 4200A-SCS.

To add a switching system to the 4200A-SCS configuration:

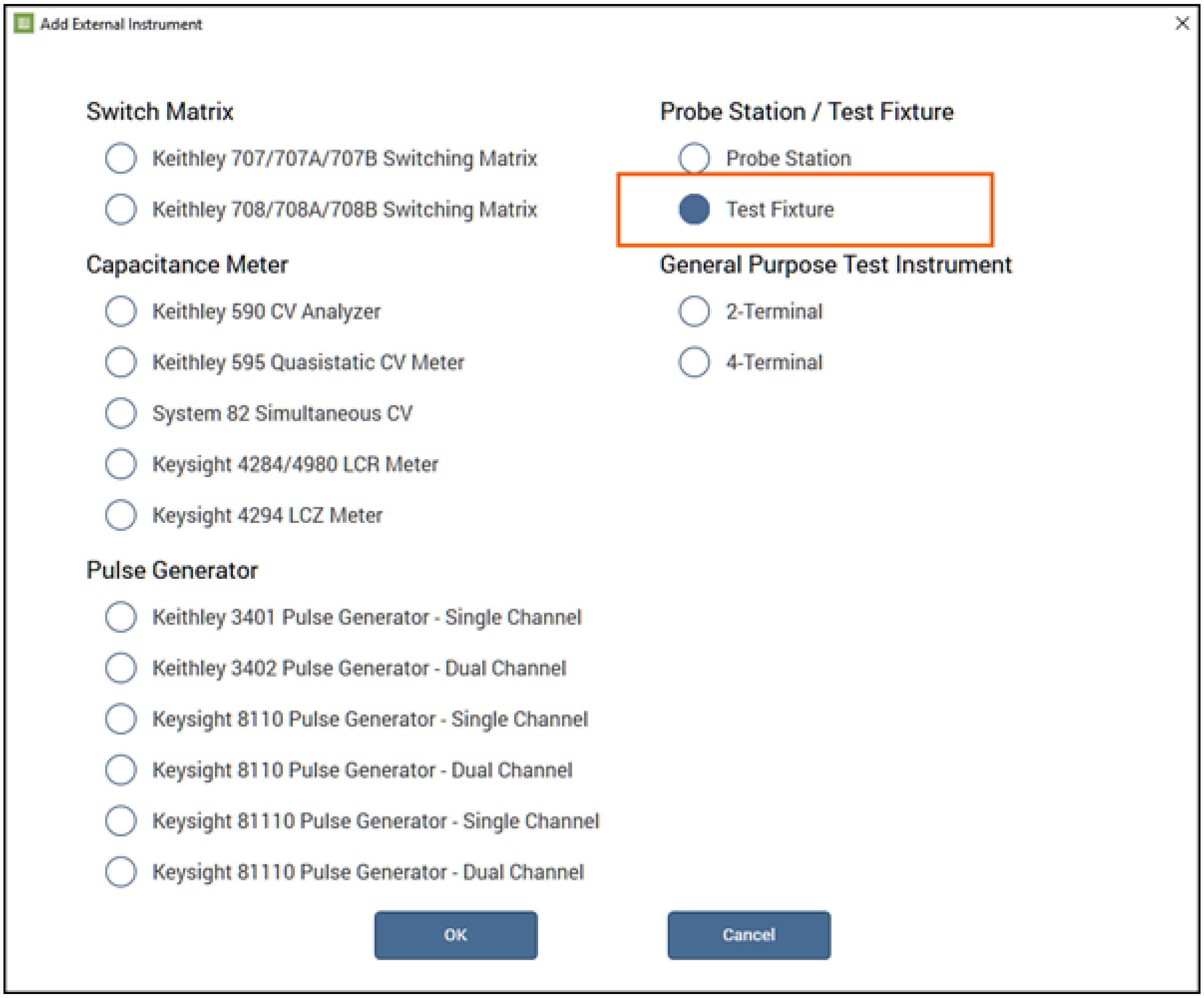
1. From the desktop, open the KCon application.
2. In the bottom left of the KCon window, select **Add External Instrument**.
3. Select your switching system. The Series 700 Switching Systems are highlighted in the following figure.

Figure 32: Add External Instrument box, Series 700 Switching Systems highlighted



4. Select **OK**.
5. Select **Add External Instrument** again.
6. Select **Test Fixture**.

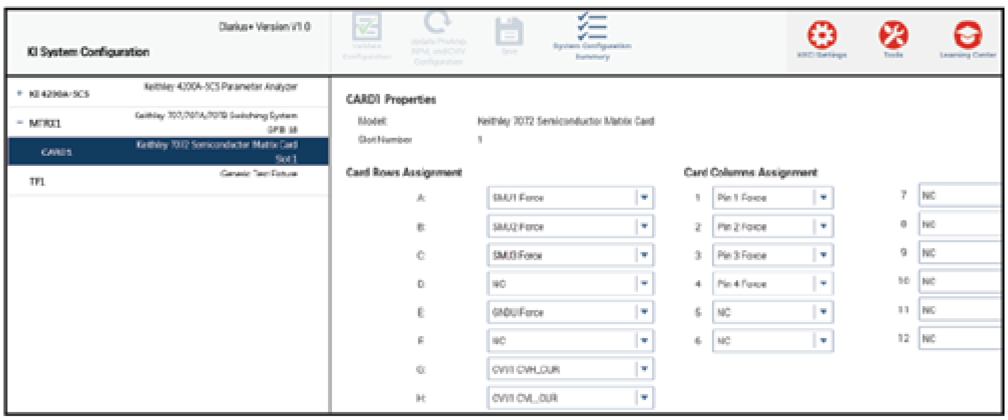
Figure 33: Add External Instrument dialog, Test Fixture highlighted



7. Select **OK**.
8. From the System Configuration list, select the test fixture you added (**TF1**).
9. Set the number of pins equal to the number of output pins in your switching system (**12** for this example, using one 7072 matrix card).
10. From the System Configuration list, select the switching system you added (**MTRX1**).
11. In the Properties pane, add the 7072 Matrix Card to the correct slot of the switching system.
12. Confirm that the GPIB Channel of your device (0 to 30) matches the channel shown in the Properties.
13. Open **MTRX1** in the System Configuration list.

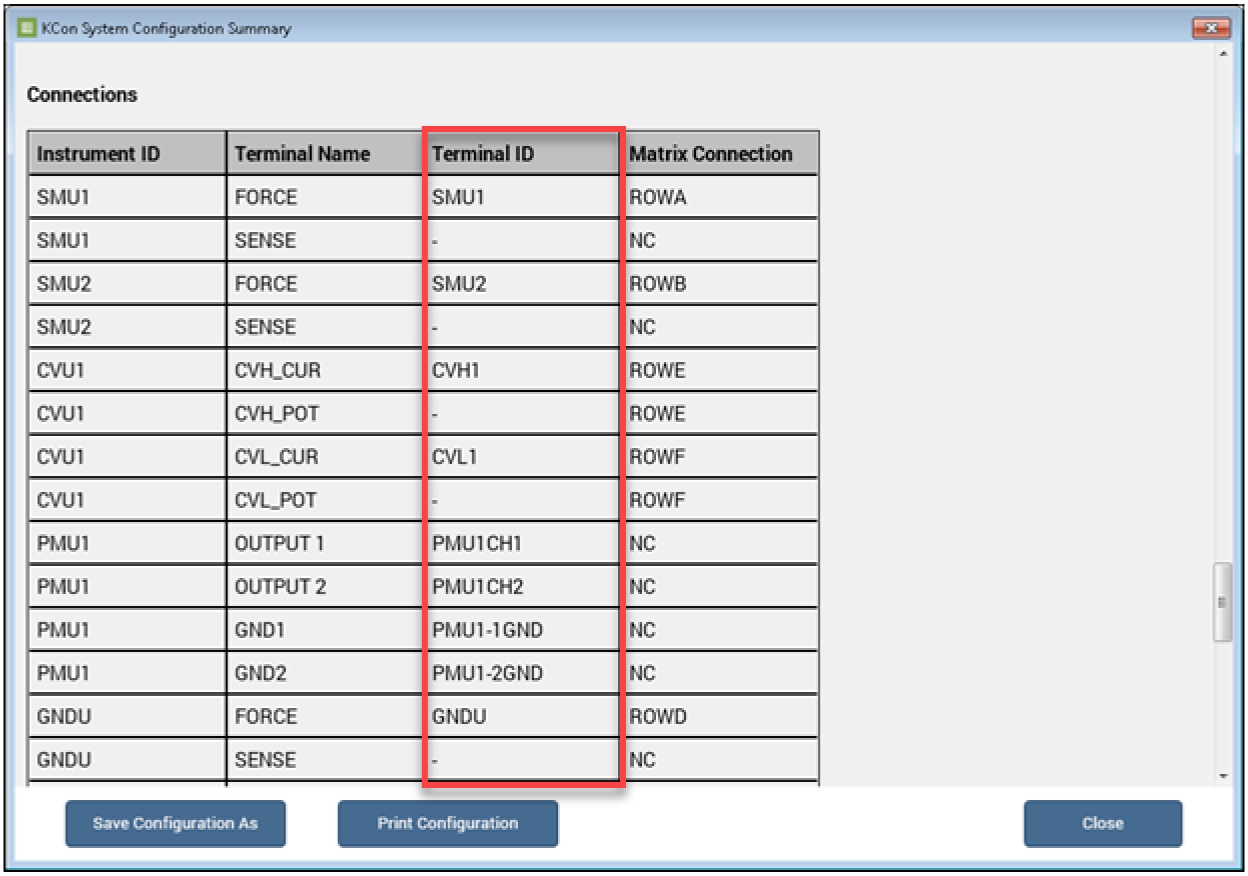
14. Select **CARD1**. The Properties for the 7072 Matrix Card are displayed, as shown in the following figure.
15. Complete the Card Rows Assignments according to how you connected the instruments to the 7072. For this example, the assignments are:
    - **Row A - SMU1 Force**
    - **Row B - SMU2 Force**
    - **Row C - SMU3 Force**
    - **Row E - GNDU Force**
    - **Row G - CVU1 CVH_CUR**
    - **Row H - CVU1 CVL_CUR**
16. Under Card Columns Assignment, designate at least the first four columns with pin assignments that match their column number. For example, Pin 1 Force to column 1.

Figure 34: Completed Properties pane for the 7072 Matrix Card



17. From the KCon toolbar, select **Validate** to ensure that the switching system is connected properly.
18. Select **Save** to save the system configuration.
19. Select **Summary**, then scroll down to the **Connections** section. You need the names from the Terminal ID column when setting the switching system connections in Clarius. You can select **Save Configuration As** or **Print Configuration** to record the terminal IDs. The default values for the most common instruments are shown in the following figure.

Figure 35: Summary: Default Terminal ID connections



Instrument ID	Terminal Name	Terminal ID	Matrix Connection
SMU1	FORCE	SMU1	ROWA
SMU1	SENSE	-	NC
SMU2	FORCE	SMU2	ROWB
SMU2	SENSE	-	NC
CVU1	CVH_CUR	CVH1	ROWE
CVU1	CVH_POT	-	ROWE
CVU1	CVL_CUR	CVL1	ROWF
CVU1	CVL_POT	-	ROWF
PMU1	OUTPUT 1	PMU1CH1	NC
PMU1	OUTPUT 2	PMU1CH2	NC
PMU1	GND1	PMU1-1GND	NC
PMU1	GND2	PMU1-2GND	NC
GNDU	FORCE	GNDU	ROWD
GNDU	SENSE	-	NC

20. Close the window when you are finished.
21. Close the KCon application.

Set up the measurements in Clarius

After closing KCon, open the Clarius application from the desktop. In this section, you use the Clarius application to configure and run two tests on an n-channel MOSFET transistor: A plot of drain current versus drain voltage using the SMUs and a C-V sweep. By using the Series 700 Switching System, you do not need to rearrange cables between the tests.

For this example, you use the Clarius application to:

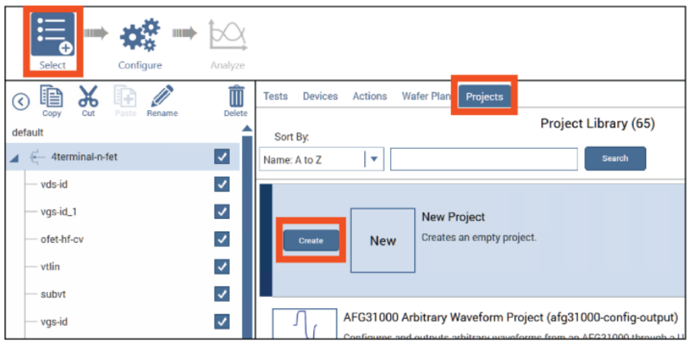
- Create a new project
- Add a device
- Add an action
- Configure the action
- Search for and add two tests
- Run the project and view the tests

Create a new project

To create a new project:

1. Choose **Select**.
2. In the Library, select **Projects**.
3. Select **New Project**.
4. Select **Create**.

Figure 36: Create new project



5. Select **Yes** when prompted to replace the existing project.

Add a device

To add a device:

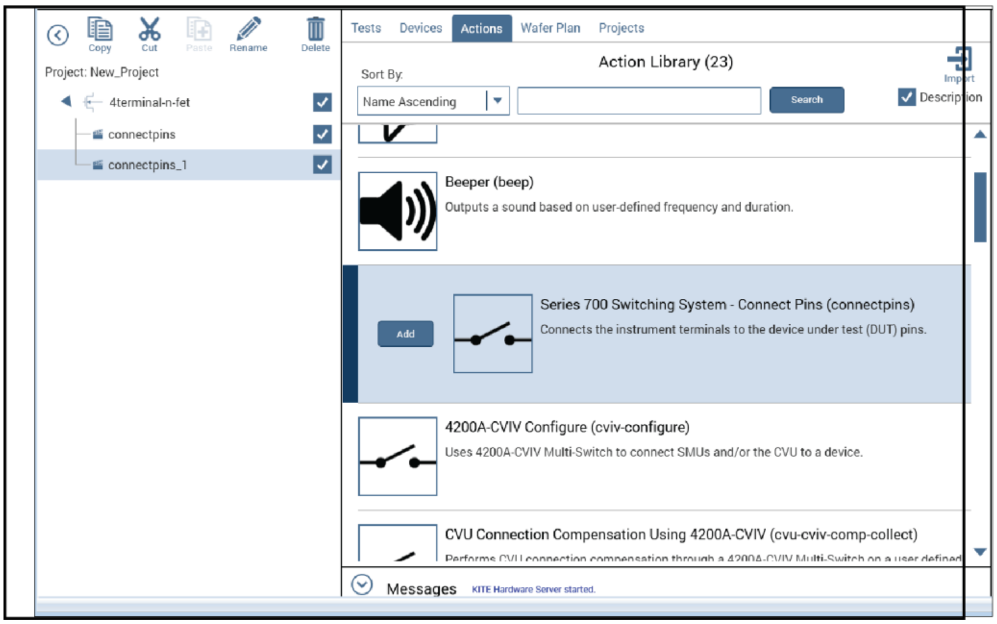
1. Select **Devices**.
2. Enter **MOSFET** in the search box.
3. Select **Search**.
4. Scroll to the **MOSFET, n-type, 4 terminal** (`4terminal-n-fet`) device.
5. Select **Add** to add it to the project tree.

Add the connectpins action

To add the connectpins action:

1. Select **Actions**.
2. Type **connect** in the search box.
3. Select **Search**.
4. Scroll to the `connectpins` action, then **Add** it to the project tree twice.

Figure 37: connectpins added twice

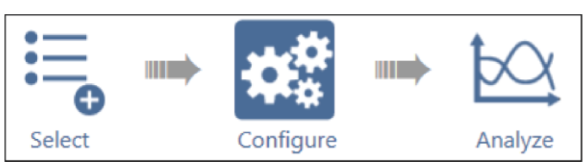


Configure the connectpins action

To configure the connectpins action:

1. Select the first `connectpins` action you added to the project tree.
2. Select **Configure**.

Figure 38: Configure highlighted

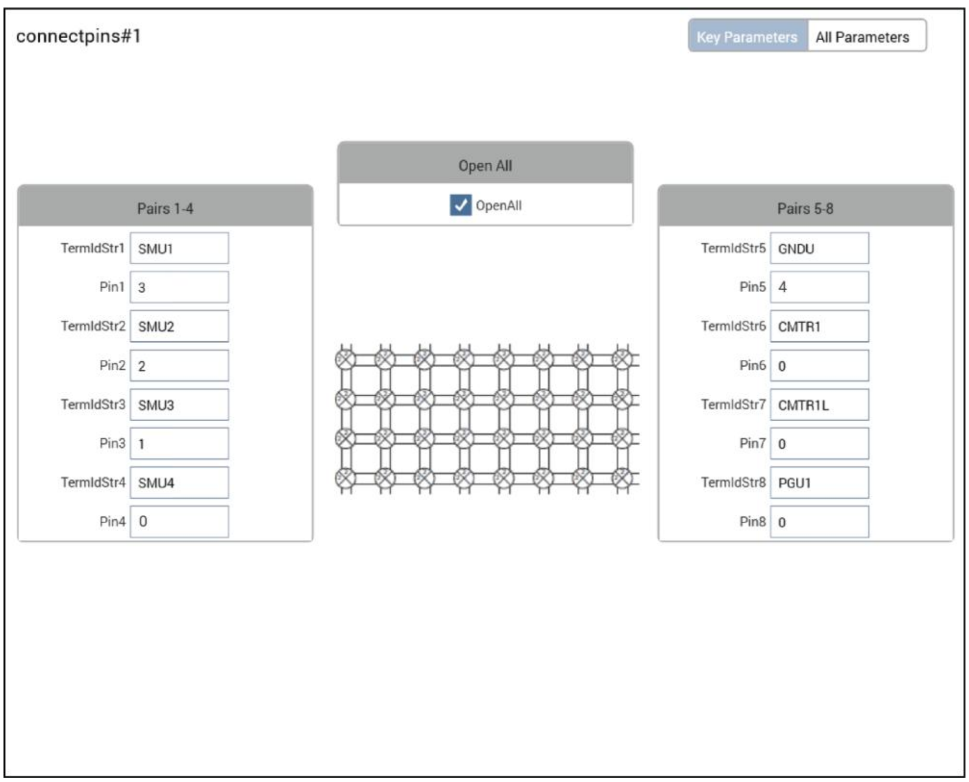


3. Make the following connections using the pairs of TermIdStr# and Pin# fields in the action:
   - **SMU1 – Pin 3**
   - **SMU2 – Pin 2**
   - **SMU3 – Pin 1**
   - **GNDU – Pin 4**

   When you are finished, the Key Parameters view of the action should look like the next graphic. The order of the instruments does not matter if each instrument is paired with the correct pin number.

   In this example, assigning TermIdStr1 to **SMU1** and Pin 1 to **3** connects SMU1 to Pin 3 on the matrix.

Figure 39: connectpins device connections



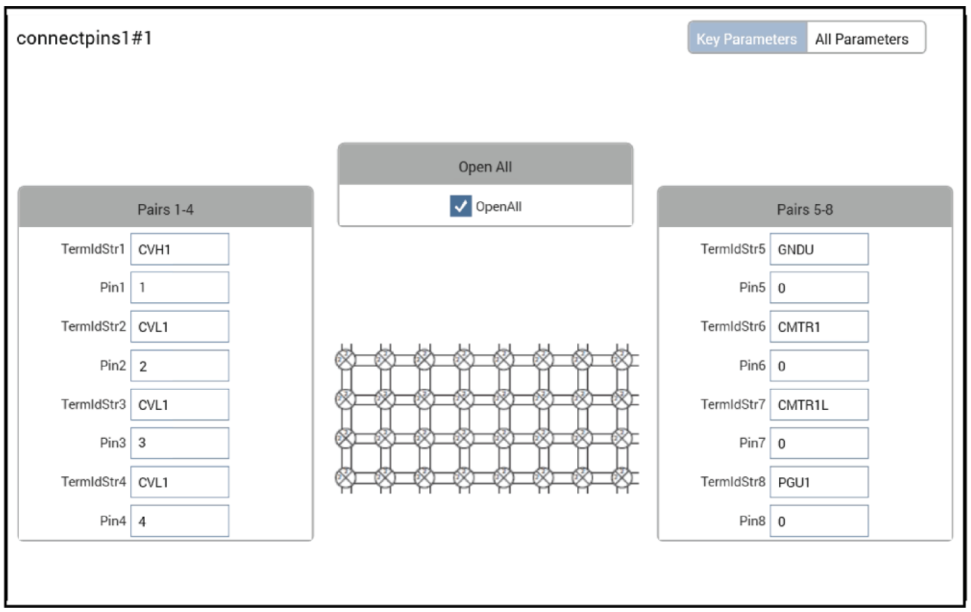
NOTE

If the OpenAll checkbox is selected, the `connectpins` action opens all crosspoints before closing the specified pairs. This is the default and is usually the preferred behavior. However, since `connectpins` only has eight field pairs, the action can only close eight crosspoints during each run. To close more crosspoints, use multiple `connectpins` actions.

4. Select **Save**.
5. Select the second `connectpins` action you added to the project tree.
6. Make the following connections using the pairs of TermIdStr# and Pin# text fields in the action:
   - **CVH1 – Pin 1**
   - **CVL1 – Pin 2**
   - **CVL1 – Pin 3**
   - **CVL1 – Pin 4**

   When you are finished, the Key Parameters for the action look like the following figure.

Figure 40: Second connectpins connections



7. Select **Save**.

Search for and add existing tests from the Test Library

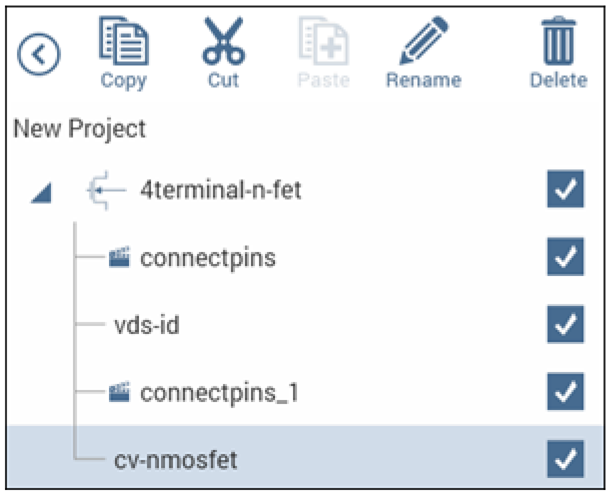
To search for and add existing tests from the Test Library:

1. Choose **Select**.
2. Select **Tests**.
3. Type **vds** into the search box.
4. Select **Search**.
5. Scroll to the `vds-id` test.
6. Select **Add** to add it to the project tree.
7. Drag the `vds-id` test to between the two `connectpins` actions.

8. Clear the search box.
9. Type **cv** into the search box.
10. Select **Search**.
11. Scroll to the **cv-nmosfet** test. Select **Add** to add it to the project tree.
12. Drag the `cv-nmosfet` test after the second `connectpins` action.

Your project tree looks like the following figure.

Figure 41: Project tree showing tests and actions in the proper order



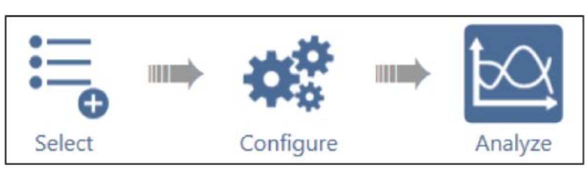
Run the project and view the tests

To run the project and view the tests:

1. In the project tree, select `New Project`.
2. Make sure the items in the project tree are checked.
3. Select **Run** to start the test. The actions and tests run sequentially. The `connectpins` actions set the crosspoints before the tests are executed.

You can select **Analyze** when you run the project to view test results in real time.

Figure 42: Analyze highlighted



To view the results of a test either as it runs or after it has completed, select the test in the project tree.

Section 4

Using a Model 590 C-V Analyzer

In this section:

Introduction

This section describes how to set up and use a Model 590 C-V Analyzer with the 4200A-SCS.

For details on 590 operation, refer to the *Model 590 C-V Analyzer Instruction Manual* (document number 590-901-01), available at tek.com/keithley.

C-V measurement basics

The Keithley Model 590 C-V Analyzer measures capacitance versus voltage (C-V) and capacitance versus time (C-t) of semiconductor devices. Typically, C-V measurements are made on capacitor-like devices, such as a metal-oxide-silicon capacitor (MOS capacitor).

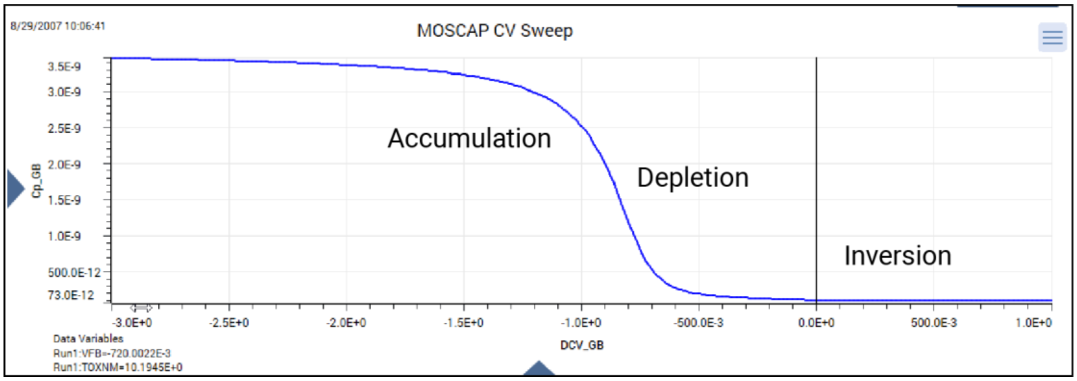
The measurements of MOS capacitors study:

- The integrity of the gate oxide and semiconductor doping profile
- The lifetime of semiconductor material
- The interface quality between the gate oxide and silicon
- Other dielectric materials used in an integrated circuit

The voltage sweeping capability of the 590 makes it easy to make a series of capacitance measurements that span the three regions of a C-V curve: the accumulation region, depletion region, and inversion region.

The following figure shows the three regions of a typical C-V curve for a MOS capacitor.

Figure 43: Typical C-V curve for a MOS capacitor



Capacitance measurement tests

The 4200A-SCS provides the following tests to perform C-V tests using the 590:

- **590 C-V Sweep (590-cvsweep):** Makes a capacitance measurement at each step of a user-configured linear voltage sweep.
- **590 C-V Pulse Sweep (590-cvpulsesweep):** Makes a capacitance measurement at each step of a user-configured pulsed voltage sweep.
- **590 C-t Sweep (590-ctsweep):** Makes a specified number of capacitance measurements at a specified time interval. Voltage is held constant for these capacitance measurements.
- **590 Capacitance Measurements (590-cmeas):** Makes capacitance and conductance measurements at a fixed bias voltage.

NOTE

There are also user modules available for the 590. Refer to KI590ulib user library (on page 4-12).

Connections

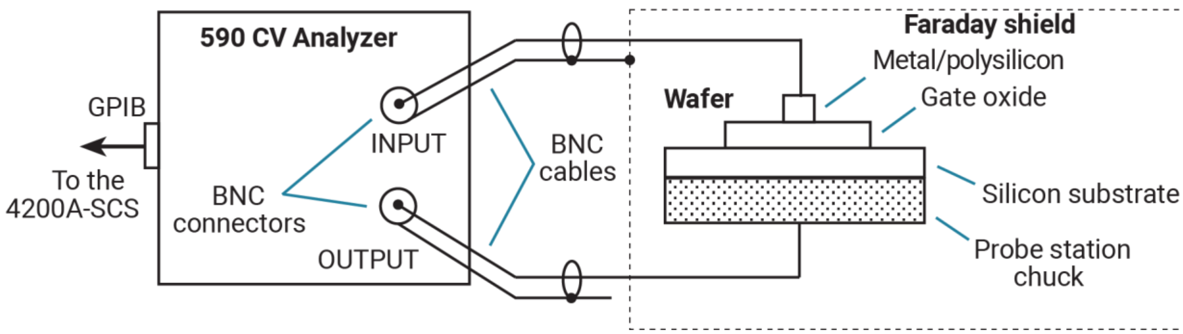
This section describes basic BNC, triaxial, and GPIB connections. For additional information about 590 connections, see the *Model 590 C-V Analyzer Instruction Manual* (document number 590-901-01), available at tek.com/keithley.

Signal connections

Basic signal connections for the 590 are shown in the following figure.

The center conductors of the BNC connectors are connected to the device under test (DUT). The outer shield of one of the coaxial cables is typically connected to a Faraday shield. The Model 590 output is typically connected to the wafer backside (or well). The input is typically connected to the gate of a MOS capacitor.

Figure 44: Basic 590 connections to the DUT



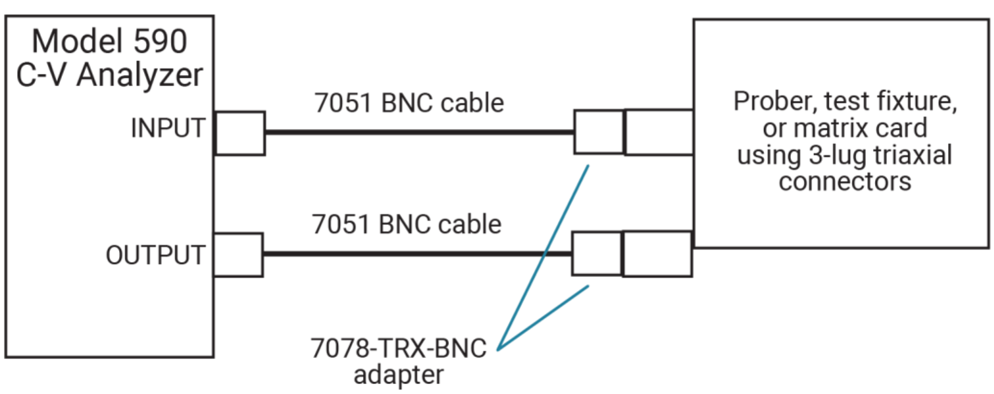
Triaxial connectors

Adapters are required to connect the 590 to equipment (for example, a probe station, test fixture, or matrix card) that uses triaxial connectors. The 7078-TRX-BNC is a 3-lug triaxial to BNC adapter. As shown in the following figure, connect the adapters to the 3-slot triaxial connectors and then use a 7051 BNC cable to make the connections to the 590.

NOTE

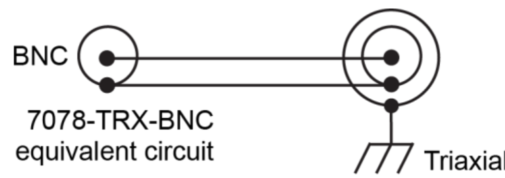
See Using Switch Matrices (on page 2-1) for details on using a switching matrix with the 590 C-V Analyzer.

Figure 45: Connecting the 590 to equipment with triaxial connectors



The following figure shows the equivalent circuit for the adapter.

Figure 46: 7078-TRX-BNC equivalent circuit



GPIB connections

The 4200A-SCS controls the 590 through the general purpose interface bus (GPIB). Use a shielded GPIB cable to connect the GPIB of the 590 to the GPIB of the 4200A-SCS.

Cable compensation

Signal pathways through the test cables, switching matrix, test fixture, and prober contribute unwanted capacitances that may adversely affect the measurement.

To correct for these unwanted capacitances, you should do cable compensation before measuring the capacitance of DUT. In general, you do cable compensation by connecting precisely known capacitance sources in place of the DUT and then measuring them. The 590 then uses these measured values to make corrections when measuring the DUT.

During cable compensation:

1. The 590 calculates the compensation parameters based on the comparison between the given and measured values.
2. The 590 makes a probe-up offset measurement and suppresses any remaining offset capacitance. This step is done every time a new measurement is made.

Typically, cable compensation is done for all measurement ranges (2 pF, 20 pF, 200 pF, and 2 nF) of the 590. Once cable compensation is done, it does not have to be done again unless the connections to the DUT are changed or power is cycled.

For each measurement range of the 590, you must use a low-capacitance source and a high-capacitance source. The following table lists the Keithley Model 5909 capacitance sources that can be used for each 590 range.

5909 capacitance sources

590 range	Low capacitance source	High capacitance source
2 pF	0.5 pF	1.5 pF
20 pF	4.7 pF	18 pF
200 pF	47 pF	180 pF
2 nF	470 pF	1.8 nF

Cable compensation user modules

The 4200A-SCS `KI590ulib` user library includes the following user modules for cable compensation:

- **SaveCableCompCaps590: Enter and save capacitance source values:** The user enters the actual capacitance values of the capacitance sources. When the test is executed, the capacitance values are stored in a file at a user-specified directory path.
- **DisplayCableCompCaps590: Places capacitance values into the Analyze spreadsheet:** When this test is executed, the capacitance values saved by `SaveCableCompCaps82` are placed into the Analyze spreadsheet.
- **CableCompensate590: Performs cable compensation:** The user specifies the ranges and test frequencies for cable compensation. When this test is executed, on-screen prompts guide you through the cable compensation process.
- **LoadCableCorrectionConstants:** This function reads the cable compensation parameters for the range and frequency specified from the cable compensation file and sends these parameters to the 590.

NOTE

Details on all user modules for the 590 are provided in KI590ulib user library (on page 4-12).

Using KCon to add a 590 C-V Analyzer to system

To use the 4200A-SCS to control an external instrument, that instrument must be added to the system configuration. The 590 C-V Analyzer is added to the test system using the Keithley Configuration Utility (KCon).

Refer to “Use KCon to add equipment to the 4200A-SCS” in *Model 4200A-SCS Parameter Analyzer Setup and Maintenance* for instruction.

Model 590 test examples

The test examples for the Model 590 C-V Analyzer are controlled by user test modules (UTMs) in the `ivcvswitch` project.

A switching matrix is not used for these examples.

Cable compensation example

This example assumes that the 590 is connected directly to the DUT. The DUT could be a device installed in a test fixture or a MOS capacitor on a wafer.

NOTE

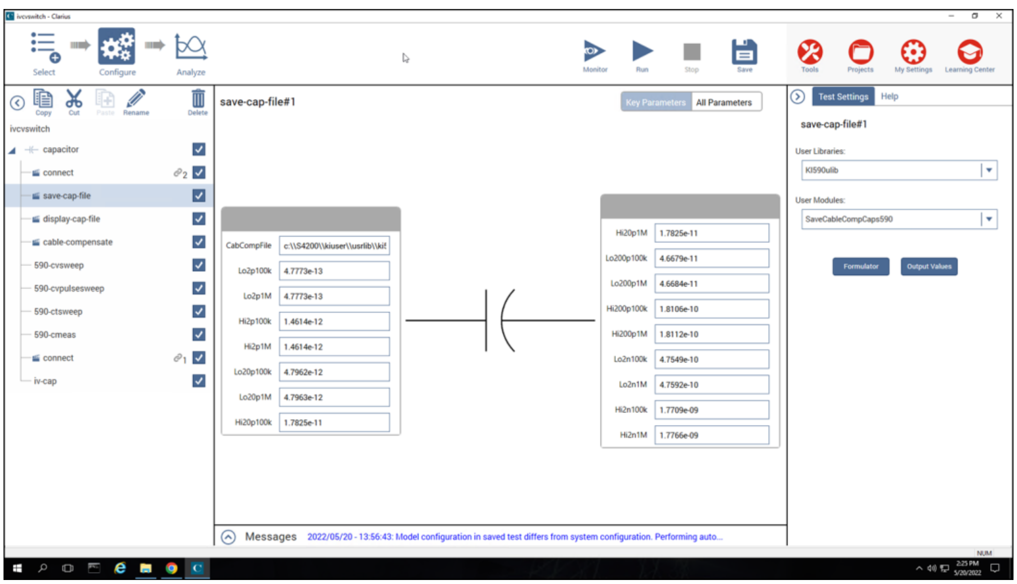
The user modules for cable compensation must share the same file for capacitance source values. Therefore, the same file directory path must be used in all three user modules. For this example, use the default file directory path (see line 1 of the parameter list in the following figures).

Enter and save capacitance source values (save-cap-file)

To enter and save the capacitance source values:

1. Select **Configure**.
2. In the project tree, select `save-cap-file`. The default parameters for the user module are displayed.

Figure 47: save-cap-file action and SaveCableCompCaps590 user module



3. Enter the capacitance source calibration value for each range and frequency. If you are using the 5909, each capacitor has a label indicating the calibration value at 100 kHz and at 1 MHz.

For example, assume the low capacitance source for the 2 pF range is 0.47773 pF (100 kHz) and 0.47786 pF (1 MHz). Enter these values using scientific notation:

- Lo2p100k: Enter `0.47773e-12`
- Lo2p1M: Enter `0.47786e-12`

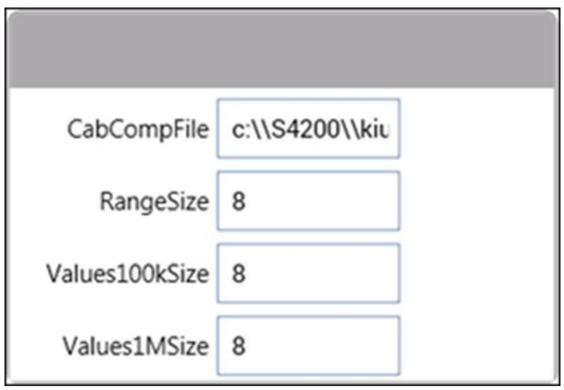
4. In the project tree, select the action.
5. Select **Run** to execute the action.

Place capacitance source values in a spreadsheet (display-cap-file)

To place the source values in the Analyze sheet:

1. Select **display-cap-file**. The default parameters are displayed.

Figure 48: DisplayCableCompCaps590 default parameters



2. Ensure that the `CabCompFile` field has the same file directory path that is used in `save-cap-file` (Enter and save capacitance source values (save-cap-file) (on page 4-6)).
3. Set the array size in the `RangeSize`, `Values100kSize`, and `Values1MSize` fields.
4. Select **Run**. The calibration source values are placed into the Analyze sheet for this action.
5. Select **Analyze** to view the spreadsheet for `display-cap-file`. An example spreadsheet is shown in the following figure.

Figure 49: Display-cap-file spreadsheet showing capacitor source values

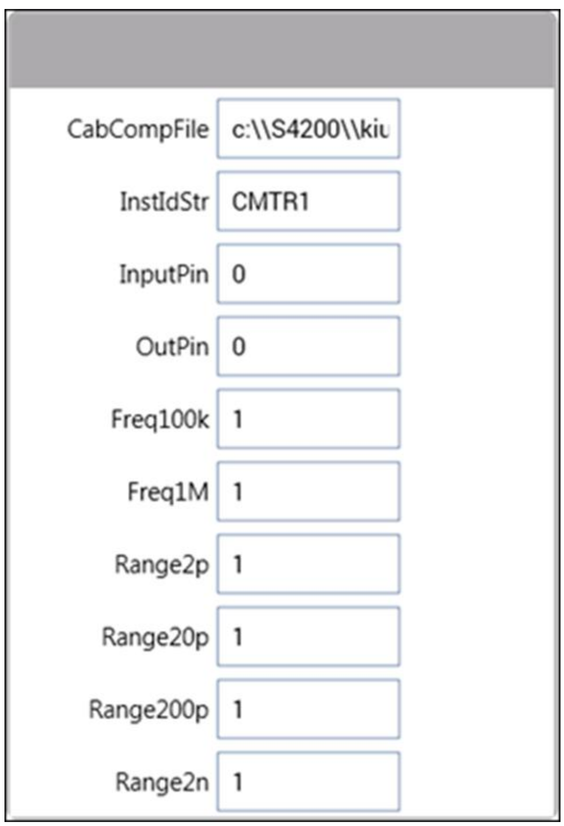
	DisplayCableC	Range	Values100k	Values1M
1	0	2.0000E-12	477.7300E-15	477.7300E-15
2		2.0000E-12	1.4614E-12	1.4614E-12
3		20.0000E-12	4.7962E-12	4.7963E-12
4		20.0000E-12	17.8250E-12	17.8250E-12
5		200.0000E-12	46.6790E-12	46.6840E-12
6		200.0000E-12	181.0600E-12	181.1200E-12
7		2.0000E-9	475.4900E-12	475.9200E-12
8		2.0000E-9	1.7709E-9	1.7766E-9
9				

Perform cable compensation (CableCompensate)

To do cable compensation:

1. Select **Configure**.
2. In the project tree, select **cable-compensate** to open the action. The following figure shows the default parameters for the action.

Figure 50: CableCompensate590 default parameters

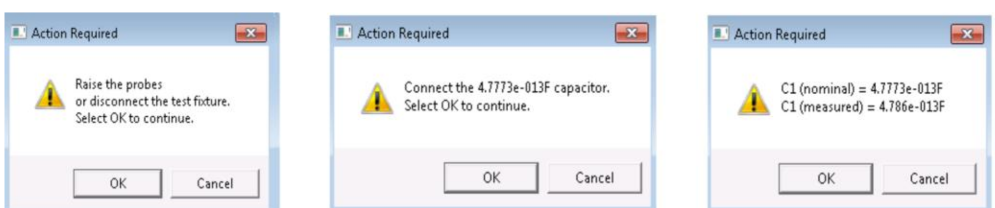


3. Ensure that `CabCompFile` has the same file directory path that is used in `save-cap-file` (Enter and save capacitance source values (SaveCableCompCaps590) (on page 4-6)).
4. Enable or disable cable compensation. The `Freq`*N* and `Range`*N* parameters either disable (`0`) or enable (`1`) cable compensation for the frequencies and ranges. The figure above shows cable compensation enabled for all ranges and test frequencies.
5. Select **Run** to execute the action. Dialogs guide you through the cable compensation process. The basic dialogs are shown in the following figure:
   - Raise the probes: This dialog indicates that an offset (open circuit) measurement is required. Open the circuit as close to the DUT as possible.
   - Connect the capacitor: The value in the dialog corresponds to a calibration value entered by the user in Enter and save capacitance source values (on page 4-6). Connect the capacitance source as close to the DUT as possible.
   - Compare readings: This dialog compares the measured value to the calibration (nominal) value entered by the user. The two readings should be fairly close. If they are not, you probably connected the wrong capacitance source or had an open circuit condition. In that case, select **Cancel** to abort the cable compensation process.

NOTE

Selecting **Cancel** in a cable compensation dialog aborts the cable compensation process. You can start over by selecting **Run**.

Figure 51: Cable compensation dialogs



Measure offset | Measure capacitance source | Compare readings

C-V sweep example

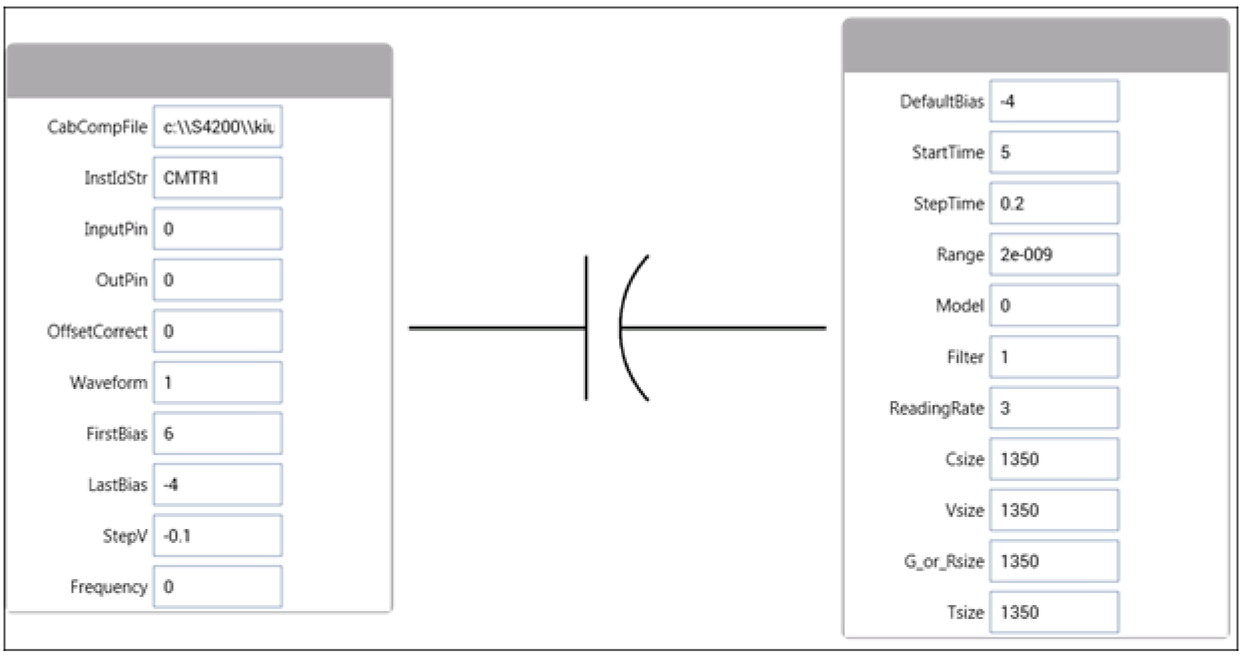
This example demonstrates how to control a Keithley 590 C-V Analyzer to acquire capacitance versus voltage (CV) data from a MOS capacitor. In this example, the C-V Analyzer applies a linear staircase voltage sweep to a capacitor. A capacitance measurement is made on every voltage step of the sweep.

This example assumes that the 590 is connected directly to the DUT. The DUT could be a device installed in a test fixture or a MOS capacitor on a wafer.

To do a C-V sweep:

1. Choose **Select**.
2. Select **Tests**.
3. Search for **590**.
4. Select **Add** for the 590 C-V Sweep (`590-cvsweep`) test.
5. In the project tree, select **590-cvsweep**.
6. Select **Configure**. The parameters are displayed.

Figure 52: CvSweep590 user module (590-cvsweep UTM)

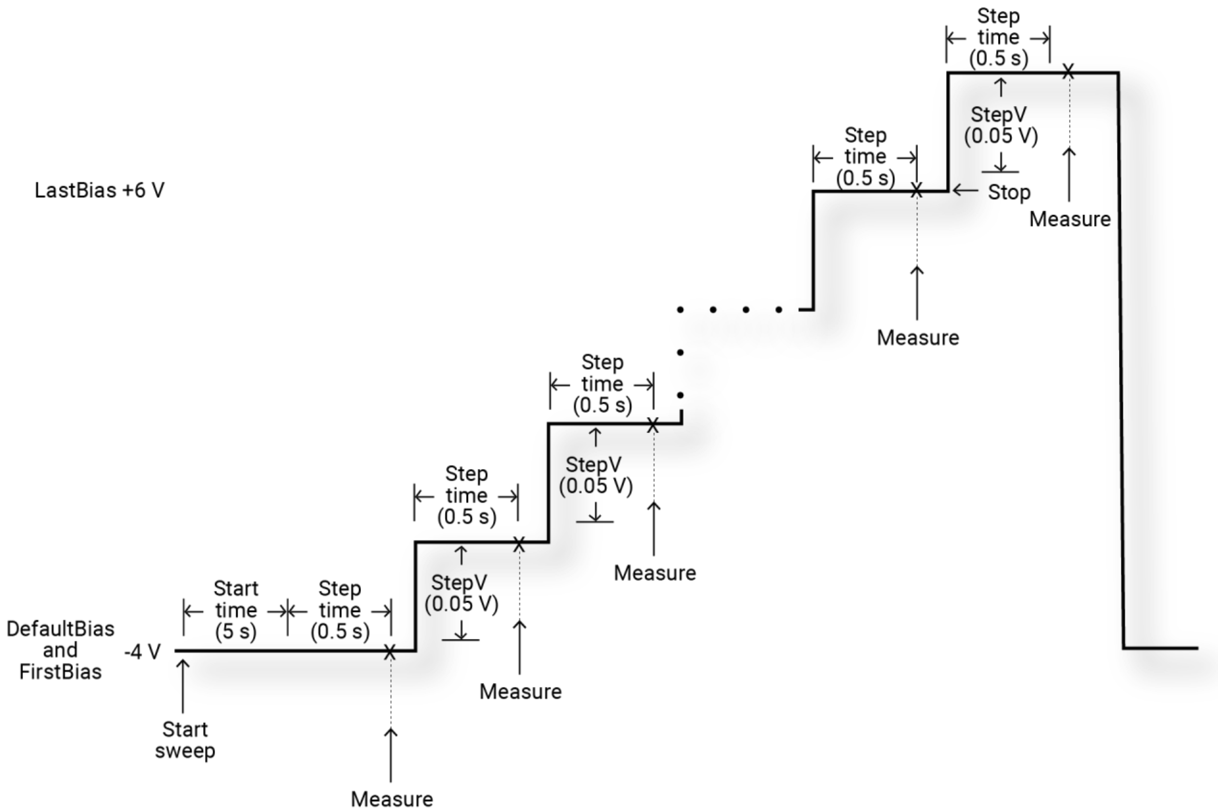


7. Change the test parameters as needed.
8. Execute the test by selecting **Run**.

This test uses the `CvSweep590` user module. For details on this test description, see CvSweep590 user module (on page 4-28).

If you use the default parameters, this test causes the 590 to do a –4 V to +6 V staircase sweep using 50 mV steps. The following figure shows the default sweep.

Figure 53: C-V linear staircase sweep



KI590ulib user library

The user modules in the KI590ulib user library are used to control the 590 C-V Analyzer. These user modules are summarized in the following table. Also listed in the table are names of the user test modules (UTMs) and actions in Clarius that use the user modules.

KI590ulib user modules

User module	UTM or action name	Description
CableCompensate590	cable-compensate in the ivcvswitch project	Performs cable compensation using known capacitance source values.
Cmeas590	590-cmeas	Makes a single capacitance measurement.
CtSweep590	590-ctsweep	Makes a capacitance versus time measurement.
CvPulseSweep590	590-cvpulsesweep	Makes capacitance versus voltage measurements using a pulse sweep.
CvSweep590	590-cvsweep	Makes capacitance versus voltage measurements using a staircase sweep.
DisplayCableCompCaps590	display-cap-file in the ivcvswitch project	Places capacitance source values in a spreadsheet.
LoadCableCorrectionConstants	n/a	Reads the cable compensation parameters for the range and frequency specified from the cable compensation file and sends these parameters to the 590.
SaveCableCompCaps590	save-cap-file in the ivcvswitch project	Saves entered capacitance source values to a file.

CableCompensate590 user module

The CableCompensate590 routine performs the 590 cable compensation procedure using the capacitor values that are stored in the specified cable compensation file. The resultant compensation values generated by the compensation process are stored in the same file.

Usage

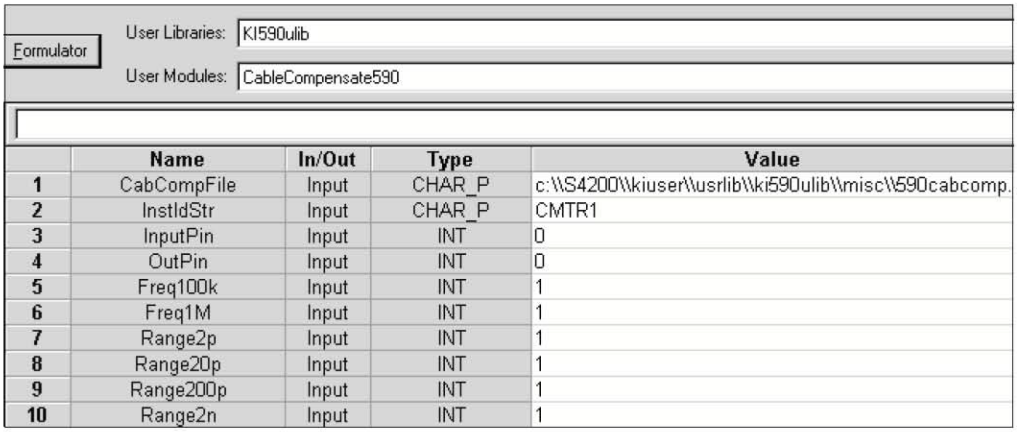
```
status = CableCompensate590(char *CabCompFile, char *InstIdStr, int InputPin, int
    OutPin, int Freq100k, int Freq1M, int Range2p, int Range20p, int Range200p, int
    range2n);
```

status	Returned values; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
InstIdStr	The CMTR instrument ID; `CMTR1`, `CMTR2`, `CMTR3`, or `CMTR4`, depending on your system configuration
InputPin	The DUT pin to which the 590 input terminal is attached (−1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
OutPin	The DUT pin to which the 590 output terminal is attached (−1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
Freq100k	Determines if compensation is done for the 100 kHz frequency: ■ Do not compensate: `0` ■ Compensate: `1`
Freq1M	Determines if compensation is done for the 1 MHz frequency: ■ Do not compensate: `0` ■ Compensate: `1`
Range2p	Determines if compensation is done for the 2 pF range: ■ Do not compensate: `0` ■ Compensate: `1`
Range20p	Determines if compensation is done for the 20 pF range: ■ Do not compensate: `0` ■ Compensate: `1`
Range200p	Determines if compensation is done for the 200 pF range: ■ Do not compensate: `0` ■ Compensate: `1`
range2n	Determines if compensation is done for the 2 nF range: ■ Do not compensate: `0` ■ Compensate: `1`

Details

This user module performs cable compensation for the selected ranges and test frequencies of the 590. The following figure shows the default input parameters for a UTM that uses the `CableCompensate590` user module.

Figure 54: CableCompensate590 default parameters


Formulator

User Libraries: KI590ulib

User Modules: CableCompensate590

	Name	In/Out	Type	Value
1	CabCompFile	Input	CHAR_P	c:\\S4200\\kiuser\\usrlib\\ki590ulib\\misc\\590cabcomp.
2	InstIdStr	Input	CHAR_P	CMTR1
3	InputPin	Input	INT	0
4	OutPin	Input	INT	0
5	Freq100k	Input	INT	1
6	Freq1M	Input	INT	1
7	Range2p	Input	INT	1
8	Range20p	Input	INT	1
9	Range200p	Input	INT	1
10	Range2n	Input	INT	1

If the default parameters are used, cable compensation is done for the 2 pF, 20 pF, 200 pF, and 2 nF ranges and for the 100 kHz and 1 MHz test frequencies. The line 1 input parameter indicates the directory path where the user-input capacitor source values are saved. These values are entered and saved using the `SaveCableCompCaps590` user module.

Test example 1 demonstrates how to do cable compensation (see Model 590 test examples (on page 4-5)).

If the file defined for *`CabCompFile`* does not exist, or there is no path specified (null string), the default compensation parameters are used. When entering the path, use two backslash (`\\`) characters to separate each directory. For example, if your cable file is in `C:\calfiles\590cal.dat`, you enter the following:

```
C:\\calfiles\\590cal.dat
```

NOTE

If a switching matrix to route signals is being controlled by a connection action (for example, `connect`), there is no need to connect *`InputPin`* and *`OutPin`*. Set these parameters to `0`.

Returned values are placed in the Analyze sheet.

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist
- `-10021 (COMP_FILE_NOT_EXIST)`: The specified compensation file does not exist
- `-10022 (KI590_NOT_IN_KCON)`: There is no CMTR defined in your system configuration.

Procedure

For each range and test frequency specified by the input parameters:

1. You are prompted to open the circuit so that an offset capacitance measurement can be made.
2. When the offset capacitance measurement is completed, you are prompted to connect the low value capacitor for the selected range. The system does the low value capacitor compensation.
3. You are prompted to connect the high value capacitor for the selected range. The system does the high value capacitor compensation.
4. You are prompted to reconnect the low capacitor.
5. The nominal and measured values are displayed in a dialog. You can:
   - Select **Cancel** to abort the procedure if the results are not correct. No changes to the cable compensation file occur.
   - Select **Save** to save the cable compensation values.

Also see

None

Cmeas590 user module

The `Cmeas590` routine measures capacitance and conductance using the Keithley 590 C-V Analyzer. You can make an offset correction measurement and use cable compensation.

Usage

```
status = Cmeas590(char *CabCompFile, char *InstIdStr, int InputPin, int OutPin, int
    OffsetCorrect, int Frequency, double DefaultBias, double StartTime, double
    Range, int Model, int Filter, int ReadingRate, double *C, double *V, double
    *G_or_R);
```

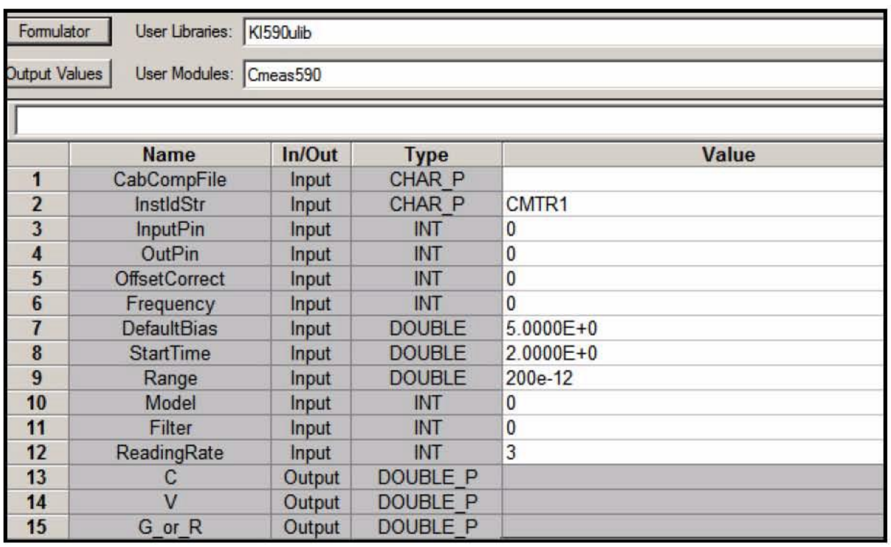
status	Returned values; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
InstIdStr	The CMTR instrument ID; `CMTR1`, `CMTR2`, `CMTR3`, or `CMTR4`, depending on your system configuration
InputPin	The DUT pin to which the 590 input terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
OutPin	The DUT pin to which the 590 output terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
OffsetCorrect	Determines if an offset correction measurement should be made: ▪ Do not make offset measurement: `0` ▪ Make offset measurement: `1`
Frequency	The measurement frequency to use: ▪ 100 kHz: `0` ▪ 1 MHz: `1`
DefaultBias	The DC bias to use for the measurement: –20 to +20 V

StartTime	The amount of time to delay after applying the *DefaultBias* voltage step (0.001 s to 65 s)
Range	The measurement range to use: 1 to 4; see **Details**
Model	Measurement model: ■ Series model: 0 ■ Parallel model: 1
Filter	Enables or disables the analog filter: ■ Disable: 0 ■ Enable (can minimize the amount noise in the readings, but increases measurement time): 1
ReadingRate	The reading rate used to acquire the measurements (0 to 4); see **Details**
C	Output: The measured capacitance
V	Output: The bias voltage used
G_or_R	Output: ■ Parallel measurement model: *G_or_R* is the measured conductance ■ Series measurement model: *G_or_R* is the measured resistance

Details

This user module makes a single, fixed-bias capacitance and conductance measurement. The default parameters for the `590-cmeas` UTM, which uses the `Cmeas590` user module, are shown in the following figure.

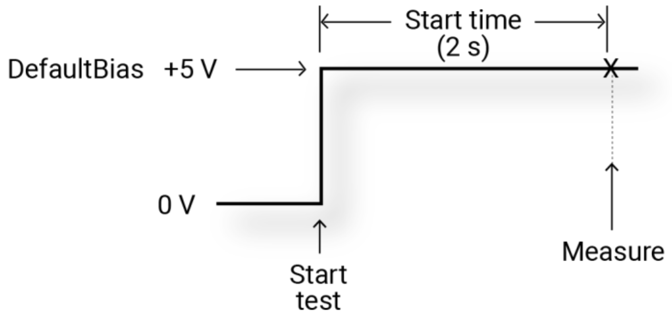
Figure 55: Cmeas590 (cmeas UTM parameters)



	Name	In/Out	Type	Value
1	CabCompFile	Input	CHAR_P	
2	InstIdStr	Input	CHAR_P	CMTR1
3	InputPin	Input	INT	0
4	OutPin	Input	INT	0
5	OffsetCorrect	Input	INT	0
6	Frequency	Input	INT	0
7	DefaultBias	Input	DOUBLE	5.0000E+0
8	StartTime	Input	DOUBLE	2.0000E+0
9	Range	Input	DOUBLE	200e-12
10	Model	Input	INT	0
11	Filter	Input	INT	0
12	ReadingRate	Input	INT	3
13	C	Output	DOUBLE_P	
14	V	Output	DOUBLE_P	
15	G_or_R	Output	DOUBLE_P	

In general, the 590 is set to source 5 V for 2 seconds, then make a measurement, as shown in the following figure.

Figure 56: Cmeas590 measurement



If the file defined for *CabCompFile* does not exist, or there is no path specified (null string), the default compensation parameters are used. When entering the path, use two backslash (\\) characters to separate each directory. For example, if your cable file is in `C:\calfiles\590cal.dat`, you enter the following:

```
C:\\calfiles\\590cal.dat
```

NOTE

If a switching matrix to route signals is being controlled by a connection action (for example, `connect`), there is no need to connect *InputPin* and *OutPin*. Set these parameters to `0`.

For *Range*, the measurement range values are shown in the following table.

Range	100 kHz	1 MHz
1	2 pF / 2 µs	20 pF / 200 µs
2	20 pF / 20 µs	20 pF / 200 µs
3	200 pF / 200 µs	200 pF / 2 ms
4	2 nF / 2 ms	2 nF / 20 ms

The reading rates and resolutions for the *ReadingRate* parameter are described in the following table.

Reading rate	Nominal reading rate (per second)	Display readings	Resolution (digits)
0	1000	C	3.5
1	75	C,G,V	3.5
2	18	C,G,V	4.5
3	10	C,G,V	4.5
4	1	C,G,V	4.5

Returned values are placed in the Analyze sheet:

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist.
- `-10020 (COMP_FILE_ACCESS_ERR)`: There was an error accessing the specified cable compensation file.
- `-10021 (COMP_FILE_NOT_EXIST)`: The specified compensation file does not exist.
- `-10022 (KI590_NOT_IN_KCON)`: There is no CMTR defined in your system configuration.
- `-10023 (KI590_MEAS_ERROR)`: A measurement error occurred.
- `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred.
- `-10091 (GPIB_TIMEOUT)`: A timeout occurred during communications.
- `-10102 (ERROR_PARSING)`: There was an error parsing the response from the 590.
- `-10104 (USER_CANCEL)`: The user canceled the correction procedure.

Procedure

1. You are prompted to open the circuit so that an offset capacitance measurement can be made, if needed.
2. If a cable compensation file is specified, the compensation information in that file for the selected range and frequency is loaded. If not, instrument default compensation is used.
3. The capacitance and conductance are measured.

Also see

None

CtSweep590 user module

The `CtSweep590` routine performs a capacitance versus time (C_t) sweep using the Keithley 590 C-V Analyzer. You can make an offset correction measurement and use cable compensation.

Usage

```
status = CtSweep590(char *CabCompFile, char *InstIdStr, int InputPin, int OutPin,
    int OffsetCorrect, int Frequency, double DefaultBias, double Bias, double
    StartTime, double StepTime, double Range, int Model, int Filter, int Count, int
    ReadingRate, double *C, int Csize, double *G_or_R, int G_or_Rsize, double *T,
    int Tsize);
```

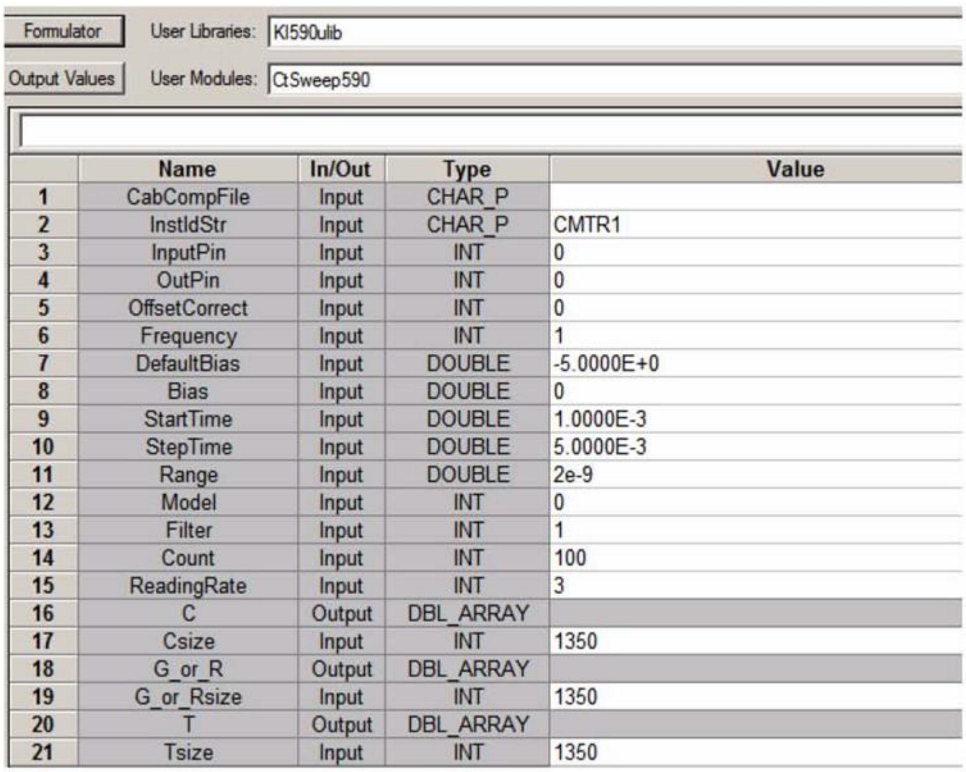
status	Returned values; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
InstIdStr	The CMTR instrument ID; `CMTR1`, `CMTR2`, `CMTR3`, or `CMTR4`, depending on your system configuration
InputPin	The DUT pin to which the 590 input terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
OutPin	The DUT pin to which the 590 output terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
OffsetCorrect	Determines if an offset correction measurement should be made: ■ Do not make offset measurement: `0` ■ Make offset measurement: `1`
Frequency	The measurement frequency to use: ■ 100 kHz: `0` ■ 1 MHz: `1`
DefaultBias	The DC bias to use for the measurement in volts: –20 to +20
Bias	The DC bias that is applied during a sweep in volts: –20 to +20
StartTime	The time that occurs on the first bias step, from the point the instrument is first triggered until the first step time in seconds: 0.001 to 65
StepTime	The period after a transition to a new bias step and before the instrument begins a measurement in seconds: 0.001 to 65
Range	The measurement range to use in F: 2E-12, 20E-12, 200E-12, or 2E-9; see **Details**
Model	Measurement model: ■ Series model: `0` ■ Parallel model: `1`
Filter	Enable or disable the analog filter, which can minimize the amount of noise that appears in the readings; however, it increases the measurement time: ■ Disable the filter: `0` ■ Enable the filter: `1`
Count	The number of readings per sweep: 1 to 1350
ReadingRate	Selects the reading rate used to acquire the measurements: 0 to 4; see **Details**
C	Output: The measured array of capacitance values
Csize	Set to a value that at minimum is equal to the *Count*; if in doubt, set to 1350

G_or_R	Output: ▪ When the parallel measurement model (`1`) is selected, *G_or_R* is the measured conductance ▪ When the series measurement model (`0`) is selected, this is the measured resistance
G_or_Rsize	Set to a value that at minimum is equal to the *Count*; if in doubt, set to 1350
T	The array of timestamps for each measurement step
Tsize	Set to a value that at minimum is equal to the *Count*; if in doubt, set to 1350

Details

This user module performs a capacitance versus time (Ct) sweep. The following figure shows the default parameters for the `ctsweep-590` UTM, which uses the `CtSweep590` user module.

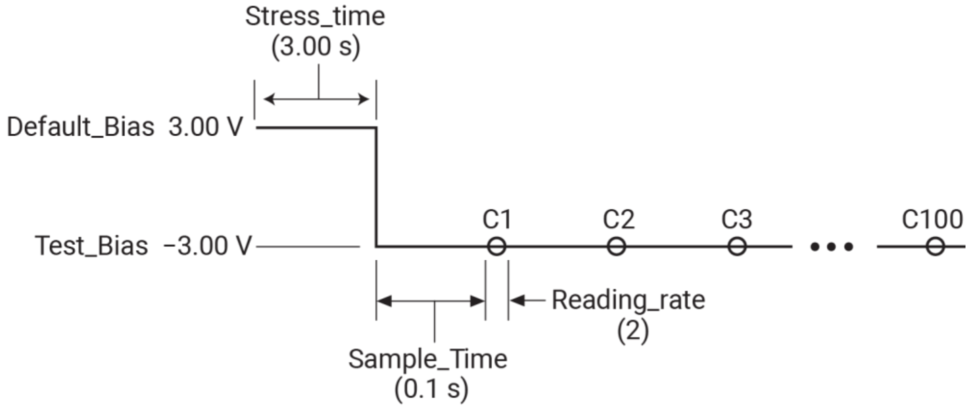
Figure 57: Starting KULT



	Name	In/Out	Type	Value
1	CabCompFile	Input	CHAR_P	
2	InstIdStr	Input	CHAR_P	CMTR1
3	InputPin	Input	INT	0
4	OutPin	Input	INT	0
5	OffsetCorrect	Input	INT	0
6	Frequency	Input	INT	1
7	DefaultBias	Input	DOUBLE	-5.0000E+0
8	Bias	Input	DOUBLE	0
9	StartTime	Input	DOUBLE	1.0000E-3
10	StepTime	Input	DOUBLE	5.0000E-3
11	Range	Input	DOUBLE	2e-9
12	Model	Input	INT	0
13	Filter	Input	INT	1
14	Count	Input	INT	100
15	ReadingRate	Input	INT	3
16	C	Output	DBL_ARRAY	
17	Csize	Input	INT	1350
18	G_or_R	Output	DBL_ARRAY	
19	G_or_Rsize	Input	INT	1350
20	T	Output	DBL_ARRAY	
21	Tsize	Input	INT	1350

In this example, the 590 is set to source –5 V and performs 100 capacitance measurements using a 5 ms time interval, as shown in the following figure.

Figure 58: C-t measurements



The time interval between reading samples is the sum of the set Sample_Time and the Reading_rate time. The Reading_rate time for Reading_rate is 1/18 s (0.0555 s). Therefore:

Time interval = Sample_Time + Reading_rate
= 0.100 + 0.056
= 0.156 s

If the file defined for *`CabCompFile`* does not exist, or there is no path specified (null string), the default compensation parameters are used. When entering the path, use two backslash (`\\`) characters to separate each directory. For example, if your cable file is in `C:\calfiles\590cal.dat`, you enter the following:

```
C:\\calfiles\\590cal.dat
```

NOTE

If a switching matrix to route signals is being controlled by a connection action (for example, `connect`), there is no need to connect *`InputPin`* and *`OutPin`*. Set these parameters to `0`.

For *`Range`*, the measurement range values are shown in the following table.

Range	100 kHz	1 MHz
1	2 pF / 2 µs	20 pF / 200 µs
2	20 pF / 20 µs	20 pF / 200 µs
3	200 pF / 200 µs	200 pF / 2 ms
4	2 nF / 2 ms	2 nF / 20 ms

The reading rates and resolutions for the *ReadingRate* parameter are described in the following table.

Reading rate	Nominal reading rate (per second)	Display readings	Resolution (digits)
0	1000	C	3.5
1	75	C,G,V	3.5
2	18	C,G,V	4.5
3	10	C,G,V	4.5
4	1	C,G,V	4.5

Returned values are placed in the Analyze sheet.

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist.
- `-10020 (COMP_FILE_ACCESS_ERR)`: There was an error accessing the specified cable compensation file.
- `-10021 (COMP_FILE_NOT_EXIST)`: The specified compensation file does not exist.
- `-10022 (KI590_NOT_IN_KCON)`: There is no CMTR defined in your system configuration.
- `-10023 (KI590_MEAS_ERROR)`: A measurement error occurred.
- `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred.
- `-10091 (GPIB_TIMEOUT)`: A timeout occurred during communications.
- `-10101 (ARRAY_SIZE_TOO_SMALL)`: The specified value for *Csize*, *G_or_Rsize*, *Vsize*, or *Tsize* was too small for the number of steps in the sweep.
- `-10102 (ERROR_PARSING)`: There was an error parsing the response from the 590.
- `-10104 (USER_CANCEL)`: The user canceled the correction procedure.

Procedure

1. You are prompted to open the circuit so that an offset capacitance measurement can be made.
2. If a cable compensation file is specified, the compensation information in that file for the selected range and frequency is loaded. If not, instrument default compensation is used.
3. A C-t sweep is performed.

Also see

None

CvPulseSweep590 user module

The `CvPulseSweep590` routine performs a capacitance versus voltage (C-V) sweep using the pulse waveform capability of the Keithley 590 C-V Analyzer. You can make an offset correction measurement and use cable compensation.

Usage

```
status = CvPulseSweep590(char *CabCompFile, char *InstIdStr, int InputPin, int
    OutPin, int OffsetCorrect, double FirstBias, double LastBias, double StepV, int
    Frequency, double DefaultBias, double StartTime, double StopTime, double
    StepTime, double Range, int Model, int Filter, int ReadingRate, double *C, int
    Csize, double *V, int Vsize, double *G_or_R, int G_or_Rsize, double *T, int
    Tsize);
```

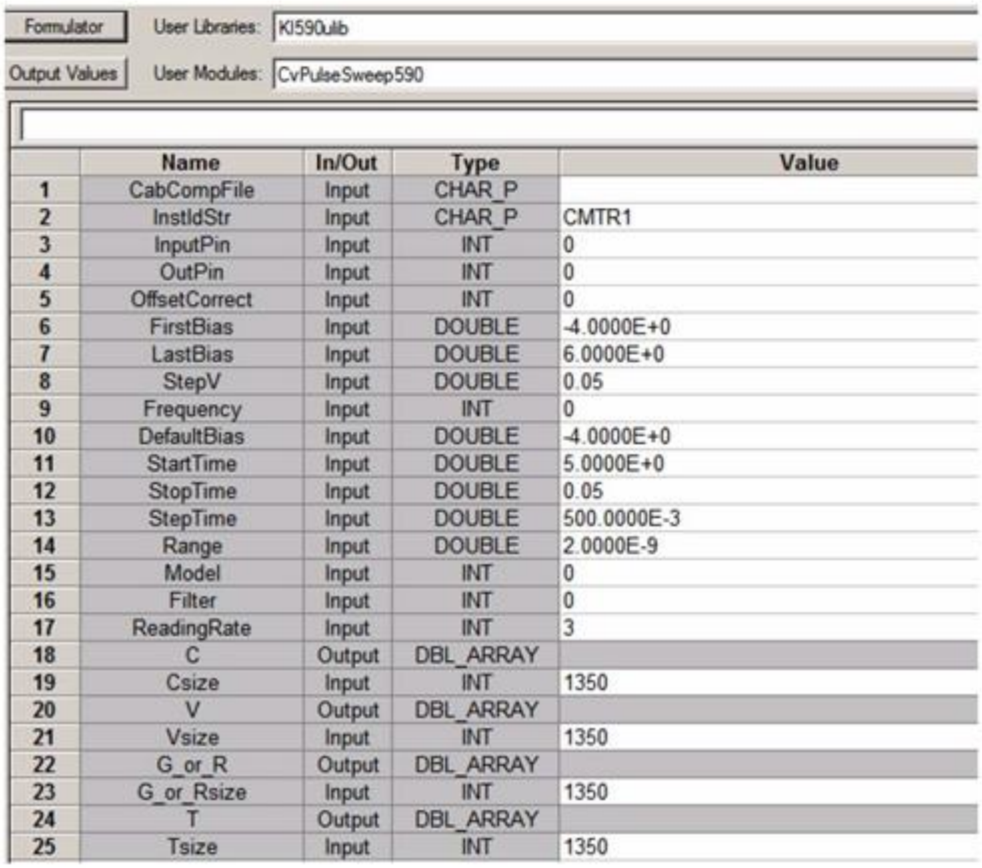
status	Returned values; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
InstIdStr	The CMTR instrument ID; `CMTR1`, `CMTR2`, `CMTR3`, or `CMTR4`, depending on your system configuration
InputPin	The DUT pin to which the 590 input terminal is attached (−1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
OutPin	The DUT pin to which the 590 output terminal is attached (−1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
OffsetCorrect	Determines if an offset correction measurement should be made: ▪ Do not make offset measurement: `0` ▪ Make offset measurement: `1`
FirstBias	The starting bias for the sweep in volts: −20 to +20; see **Details**
LastBias	The last voltage used in the sweep: −20 to +20; see **Details**
StepV	The voltage step size: −20 to +20; see **Details**
Frequency	The measurement frequency to use: ▪ 100 kHz: `0` ▪ 1 MHz: `1`
DefaultBias	The DC bias applied before and after a sweep in volts: −20 to +20
StartTime	The time that occurs on the first bias step, from the point the instrument is first triggered until the first step time in seconds: 0.001 to 65
StopTime	The time between pulses with the 590 at the default bias in seconds: 0.001 to 65
StepTime	The time after a transition to a new bias step and before the instrument begins a measurement: 0.001 s to 65 s
Range	The measurement range to use in F: 2E-12, 20E-12, 200E-12, or 2E-9; see **Details**
Model	Measurement model: ▪ Series model: `0` ▪ Parallel model: `1`
Filter	Enable or disable the analog filter, which can minimize the amount of noise that appears in the readings; however, it increases the measurement time: ▪ Disable the filter: `0` ▪ Enable the filter: `1`
ReadingRate	Selects the reading rate used to acquire the measurements: 0 to 4; see **Details**
C	Output: The measured array of capacitance values

Csize	Set to a value that is equal to or greater than the *G_or_Rsize* or number of voltage steps in the sweep, or is equal to ((*LastBias* – *FirstBias*) /*StepV*) + 1; when this function is called from a Clarius, this parameter is fixed at 1350
V	The array of bias voltages used
Vsize	Set to a value that is equal to or greater than the *G_or_Rsize* or number of voltage steps in the sweep, or is equal to ((*LastBias* – *FirstBias*) /*StepV*) + 1; when this function is called from a Clarius, this parameter is fixed at 1350
G_or_R	Output: ▪ When the parallel measurement model (1) is selected, *G_or_R* is the measured conductance ▪ When the series measurement model (0) is selected, this is the measured resistance
G_or_Rsize	Set to a value that is equal to or greater than the *G_or_Rsize* or number of voltage steps in the sweep, or is equal to ((*LastBias* – *FirstBias*) /*StepV*) + 1; when this function is called from a Clarius UTM, this parameter is fixed at 1350
T	The array of timestamps for each measurement step
Tsize	Set to a value that is equal to or greater than the *G_or_Rsize* or number of voltage steps in the sweep, or is equal to ((*LastBias* – *FirstBias*) /*StepV*) + 1; when this function is called from a Clarius UTM, this parameter is fixed at 1350

Details

This user module performs a capacitance versus voltage pulse sweep. The following figure shows the default parameters for the `590-cvpulsesweep` UTM, which uses the `CvPulseSweep590` user module.

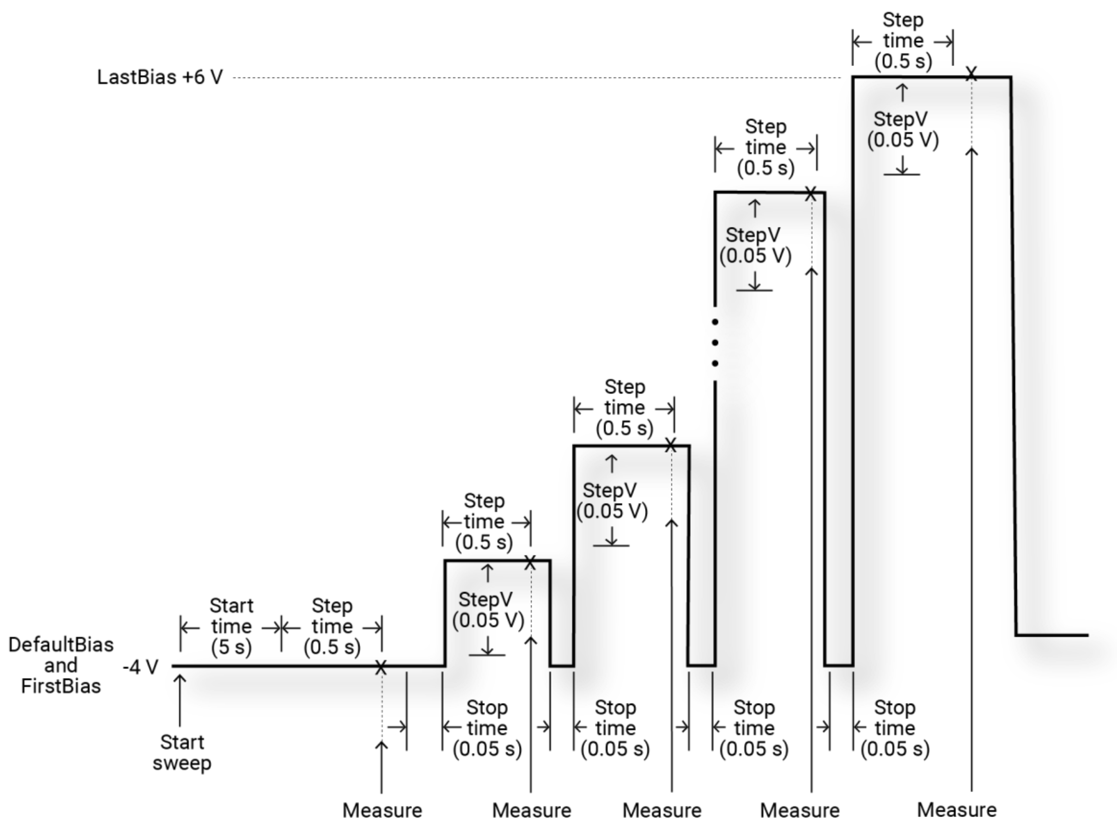
Figure 59: CvPulseSweep590 (cvpulsesweep UTM parameters)



	Name	In/Out	Type	Value
1	CabCompFile	Input	CHAR_P	
2	InstIdStr	Input	CHAR_P	CMTR1
3	InputPin	Input	INT	0
4	OutPin	Input	INT	0
5	OffsetCorrect	Input	INT	0
6	FirstBias	Input	DOUBLE	-4.0000E+0
7	LastBias	Input	DOUBLE	6.0000E+0
8	StepV	Input	DOUBLE	0.05
9	Frequency	Input	INT	0
10	DefaultBias	Input	DOUBLE	-4.0000E+0
11	StartTime	Input	DOUBLE	5.0000E+0
12	StopTime	Input	DOUBLE	0.05
13	StepTime	Input	DOUBLE	500.0000E-3
14	Range	Input	DOUBLE	2.0000E-9
15	Model	Input	INT	0
16	Filter	Input	INT	0
17	ReadingRate	Input	INT	3
18	C	Output	DBL_ARRAY	
19	Csize	Input	INT	1350
20	V	Output	DBL_ARRAY	
21	Vsize	Input	INT	1350
22	G_or_R	Output	DBL_ARRAY	
23	G_or_Rsize	Input	INT	1350
24	T	Output	DBL_ARRAY	
25	Tsize	Input	INT	1350

In this example, the 590 outputs a series of pulses in 50 mV steps from –4 V to +6 V. As shown in the following figure, a measurement is made on each pulse step.

Figure 60: C-V pulse sweep measurements



If the file defined for `CabCompFile` does not exist, or there is no path specified (null string), the default compensation parameters are used. When entering the path, use two backslash (\\) characters to separate each directory. For example, if your cable file is in `C:\calfiles\590cal.dat`, you enter the following:

```
C:\\calfiles\\590cal.dat
```

NOTE

If a switching matrix to route signals is being controlled by a connection action (for example, `connect`), there is no need to connect `InputPin` and `OutPin`. Set these parameters to `0`.

The value of ((`LastBias` – `FirstBias`) / `StepV`) +1 must be less than or equal to the `Csize`, `Vsize`, `G_or_Rsize`, and `Tsize` parameters.

For `Range`, the measurement range values are shown in the following table.

Range	100 kHz	1 MHz
1	2 pF / 2 µs	20 pF / 200 µs
2	20 pF / 20 µs	20 pF / 200 µs
3	200 pF / 200 µs	200 pF / 2 ms
4	2 nF / 2 ms	2 nF / 20 ms

The reading rates and resolutions for the *ReadingRate* parameter are described in the following table.

Reading rate	Nominal reading rate (per second)	Display readings	Resolution (digits)
0	1000	C	3.5
1	75	C,G,V	3.5
2	18	C,G,V	4.5
3	10	C,G,V	4.5
4	1	C,G,V	4.5

Returned values are placed in the Analyze sheet.

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist.
- `-10020 (COMP_FILE_ACCESS_ERR)`: There was an error accessing the specified cable compensation file.
- `-10021 (COMP_FILE_NOT_EXIST)`: The specified compensation file does not exist.
- `-10022 (KI590_NOT_IN_KCON)`: There is no CMTR defined in your system configuration.
- `-10023 (KI590_MEAS_ERROR)`: A measurement error occurred.
- `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred.
- `-10091 (GPIB_TIMEOUT)`: A timeout occurred during communications.
- `-10101 (ARRAY_SIZE_TOO_SMALL)`: The specified value for *Csize*, *G_or_Rsize*, *Vsize*, or *Tsize* was too small for the number of steps in the sweep.
- `-10102 (ERROR_PARSING)`: There was an error parsing the response from the 590.
- `-10104 (USER_CANCEL)`: The user canceled the correction procedure.

Procedure

1. You are prompted to open the circuit so that an offset capacitance measurement can be made if needed.
2. If a cable compensation file is specified, the compensation information in that file for the selected range and frequency is loaded. If not, instrument default compensation is used.
3. A C-V pulse sweep is done.

Also see

None

CvSweep590 user module

The `CvSweep590` routine does a capacitance versus voltage (C-V) sweep using the Keithley 590 C-V Analyzer. You can make an offset correction measurement and use cable compensation.

Usage

```
status = CvSweep590(char *CabCompFile, char *InstIdStr, int InputPin, int OutPin,
    int OffsetCorrect, int Waveform, double FirstBias, double LastBias, double
    StepV, int Frequency, double DefaultBias, double StartTime, double StepTime,
    double Range, int Model, int Filter, int ReadingRate, double *C, int Csize,
    double *V, int Vsize, double *G_or_R, int G_or_Rsize, double *T, int Tsize);
```

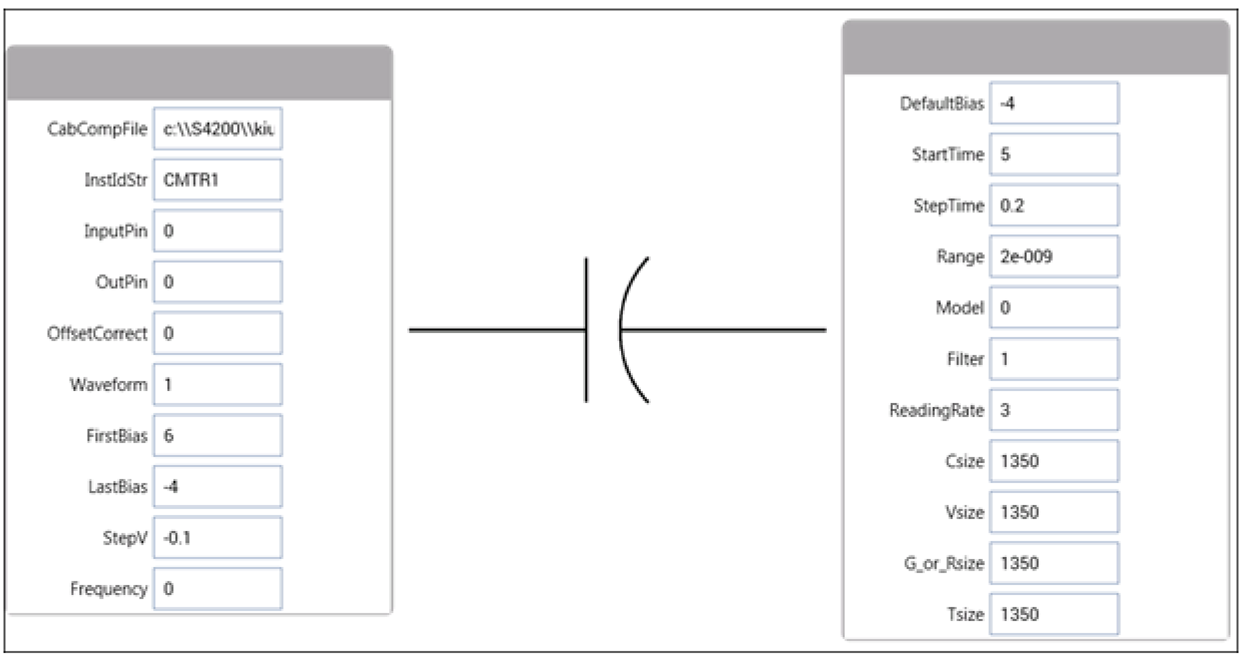
status	Returned values; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
InstIdStr	The CMTR instrument ID; `CMTR1`, `CMTR2`, `CMTR3`, or `CMTR4`, depending on your system configuration
InputPin	The DUT pin to which the 590 input terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
OutPin	The DUT pin to which the 590 output terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
OffsetCorrect	Determines if an offset correction measurement should be made: ■ Do not make offset measurement: `0` ■ Make offset measurement: `1`
Waveform	Selects either the single staircase or dual staircase waveform: ■ Single: `1` ■ Dual: `2`
FirstBias	The starting voltage for the sweep in volts: –20 to +20
LastBias	The last voltage used in the sweep in volts: –20 to +20
StepV	The voltage step size in volts: –20 to +20; the value of ((*LastBias* – *FirstBias*) / *StepV*) + 1 must be less than or equal to the *Csize*, *Vsize*, *G_or_Rsize*, and *Tsize* parameters
Frequency	The measurement frequency to use: ■ 100 kHz: `0` ■ 1 MHz: `1`
DefaultBias	The DC bias that is applied before and after a sweep in volts: –20 to +20
StartTime	The time that occurs on the first bias step, from the point the instrument is first triggered until the first step time in seconds: 0.001 to 65
StepTime	The period after a transition to a new bias step and before the instrument begins a measurement (seconds): 0.001 to 65
Range	The measurement range to use in F: 2E-12, 20E-12, 200E-12, or 2E-9; see **Details**
Model	Measurement model: ■ Series model: `0` ■ Parallel model: `1`

Filter	Enable or disable the analog filter, which can minimize the amount of noise that appears in the readings; however, it increases the measurement time: ■ Disable the filter: `0` ■ Enable the filter: `1`
ReadingRate	Selects the reading rate used to acquire the measurements: 0 to 4; see **Details**
C	Output: The measured array of capacitance values
Csize	Set to a value equal to or greater than the number of voltage steps in the sweep or equal to ((*LastBias* – *FirstBias*) / *StepV*) + 1
V	Output: The array of bias voltages used
Vsize	Set to a value equal to or greater than the number of voltage steps in the sweep or equal to ((*LastBias* – *FirstBias*) / *StepV*) + 1
G_or_R	Output: ■ When the parallel measurement model (0) is selected, *G_or_R* is the measured conductance ■ When the series measurement model (1) is selected, this is the measured resistance
G_or_Rsize	When this function is called from a Clarius UTM, this parameter is fixed at 1350
T	The array of timestamps for each measurement step
Tsize	When this function is called from a Clarius UTM, this parameter is fixed at 1350

Details

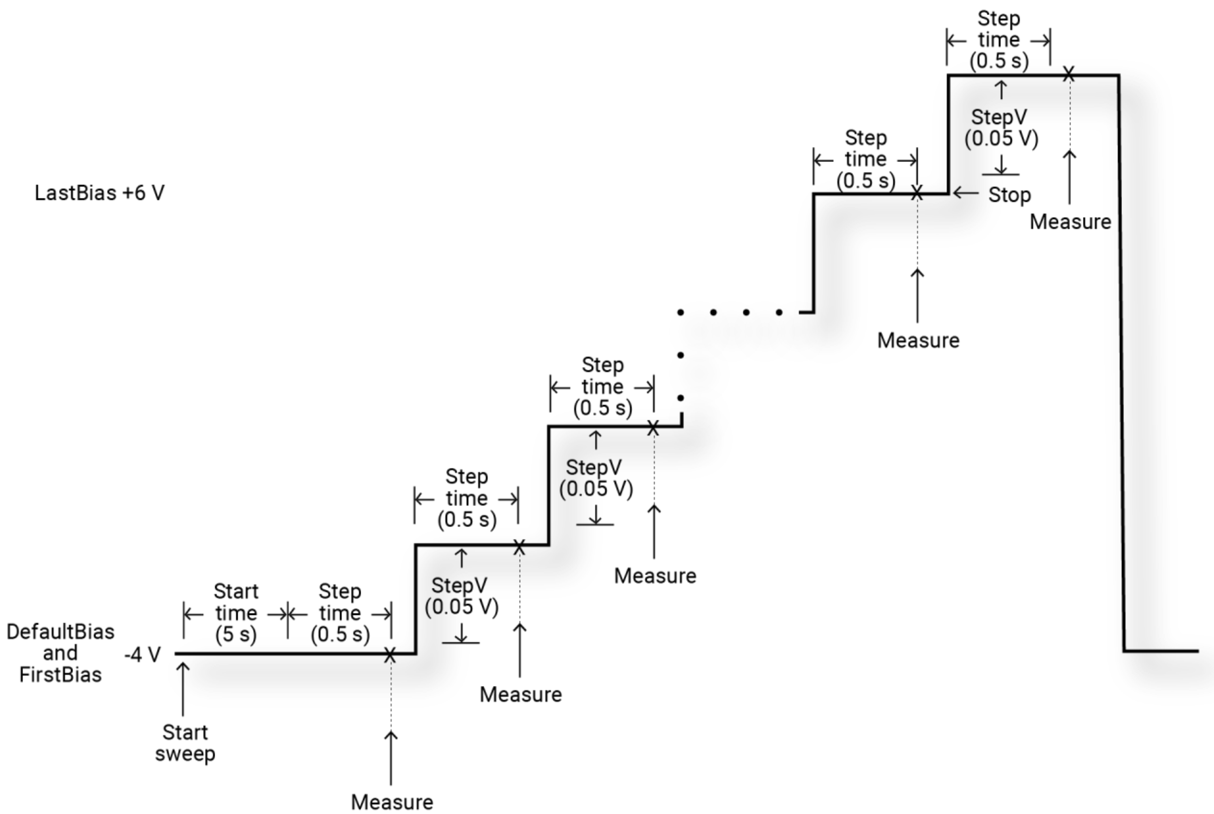
This user module performs a capacitance versus voltage staircase sweep. The following figure shows the default parameters for the `590-cvsweep` UTM, which uses the `CvSweep590` user module.

Figure 61: CvSweep590 user module (590-cvsweep UTM)



In general, the 590 outputs a linear staircase voltage sweep from –4 V to +6 V in 50 mV steps. As shown in the following figure, a capacitance measurement is made on each step of the sweep. A test example demonstrates how to perform a C-V sweep (see Model 590 test examples (on page 4-5)).

Figure 62: C-V linear staircase sweep



If the file defined for `CabCompFile` does not exist, or there is no path specified (null string), the default compensation parameters are used. When entering the path, use two backslash (\\) characters to separate each directory. For example, if your cable file is in `C:\calfiles\590cal.dat`, you enter the following:

```
C:\\calfiles\\590cal.dat
```

NOTE

If a switching matrix to route signals is being controlled by a connection action (for example, `connect`), there is no need to connect *InputPin* and *OutPin*. Set these parameters to `0`.

For *Range*, the measurement range values are shown in the following table.

Range	100 kHz	1 MHz
1	2 pF / 2 µs	20 pF / 200 µs
2	20 pF / 20 µs	20 pF / 200 µs
3	200 pF / 200 µs	200 pF / 2 ms
4	2 nF / 2 ms	2 nF / 20 ms

The reading rates and resolutions for the *ReadingRate* parameter are described in the following table.

Reading rate	Nominal reading rate (per second)	Display readings	Resolution (digits)
0	1000	C	3.5
1	75	C,G,V	3.5
2	18	C,G,V	4.5
3	10	C,G,V	4.5
4	1	C,G,V	4.5

Returned values are placed in the Analyze sheet:

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist.
- `-10020 (COMP_FILE_ACCESS_ERR)`: There was an error accessing the specified cable compensation file.
- `-10021 (COMP_FILE_NOT_EXIST)`: The specified compensation file does not exist.
- `-10022 (KI590_NOT_IN_KCON)`: There is no CMTR defined in your system configuration.
- `-10023 (KI590_MEAS_ERROR)`: A measurement error occurred.
- `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred.
- `-10091 (GPIB_TIMEOUT)`: A timeout occurred during communications.
- `-10101 (ARRAY_SIZE_TOO_SMALL)`: The specified value for *Csize*, *G_or_Rsize*, *Vsize*, or *Tsize* was too small for the number of steps in the sweep.
- `-10102 (ERROR_PARSING)`: There was an error parsing the response from the 590.
- `-10104 (USER_CANCEL)`: The user canceled the correction procedure.

Procedure

1. You are prompted to open the circuit so that an offset capacitance measurement can be made if needed.
2. If a cable compensation file is specified, the compensation information in that file for the selected range and frequency is loaded. If not, instrument default compensation is used.
3. A C-V sweep is performed.

Also see

None

DisplayCableCompCaps590 user module

`DisplayCableCompCaps590` reads the nominal cable compensation values that are stored in the compensation file, and returns them to the calling function or, in the case of Clarius, to the Analyze sheet.

Usage

```
status = DisplayCableCompCaps590(char *CabCompFile, double *Range, int RangeSize,
   double *Values100k, int Values100kSize, double *Values1M, int Values1MSize);
```

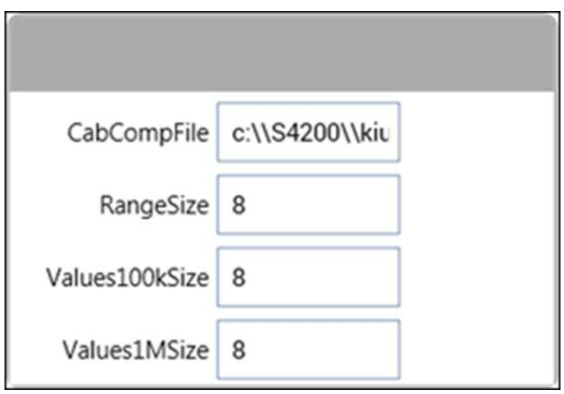
status	Returned values are placed in the Analyze sheet; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
Range	Output: An 8-element array that receives the nominal range values
RangeSize	The size of the *Range* array; set to 8
Values100k	Output: An 8-element fixed array that receives the nominal capacitor values used for the cable compensation at the 100 kHz frequency
Values100kSize	The size of the *Values100k* array; set to 8
Values1M	Output: An 8-element fixed array that receives the nominal capacitor values used for the cable compensation at the 1 MHz frequency
Values1MSize	The size of the *Values1M* array; set to 8

Details

This user module is used for 590 cable compensation. When this test is run, the nominal capacitance source values saved by the `SaveCableCompCaps590` user module are placed into a spreadsheet.

The default parameters for this user module are shown in the following figure. Line 1 specifies the file directory path where the capacitance values are saved. This file directory path must be the same as the one used by the `SameCableCompCaps590` user module.

Figure 63: DisplayCableCompCaps590 default parameters



To prevent unpredictable results, the array size values for the *RangeSize*, *Values100kSize*, and *Values1MSize* must be set to `8`, as shown in the figure above.

See Example 1: Cable compensation (on page 4-6) for a demonstration of how cable compensation is done.

The returned arrays are arranged in the order shown in the following table.

Reading_rate valid inputs

Range	100 kHz values	1 MHz values
2E-12	2 pF low comp value	2 pF low comp value
2E-12	2 pF high comp value	2 pF high comp value
20E-12	20 pF low comp value	20 pF low comp value
20E-12	20 pF high comp value	20 pF high comp value
200E-12	200 pF low comp value	200 pF low comp value
200E-12	200 pF high comp value	200 pF high comp value
2E-9	2 nF low comp value	2 nF low comp value
2E-9	2 nF high comp value	2 nF high comp value

If the file defined for *CabCompFile* does not exist, or there is no path specified (null string), the default compensation parameters are used. When entering the path, use two backslash (\\) characters to separate each directory. For example, if your cable file is in C:\calfiles\590cal.dat, you enter the following:

```
C:\\calfiles\\590cal.dat
```

NOTE

If a switching matrix to route signals is being controlled by a connection action (for example, connect), there is no need to connect *InputPin* and *OutPin*. Set these parameters to 0.

The return values from *status* can be:

- 0: OK.
- -10021 (COMP_FILE_NOT_EXIST): The specified compensation file does not exist.
- -10022 (KI590_NOT_IN_KCON): There is no CMTR defined in your system configuration.

Also see

SaveCableCompCaps590 user module (on page 4-35)

LoadCableCorrectionConstants

`LoadCableCorrectionConstants` reads the cable compensation parameters for the range and frequency specified from the cable compensation file and sends these parameters to the 590.

Usage

```
status = LoadCableCorrectionConstants(char *CabCompFile, char *instr_id,
   *frequency, int range);
```

status	Returned values are placed in the Analyze sheet; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
instr_id	The CMTR instrument ID; this can be `CMTR1` through `CMTR4`, depending on the configuration of your system
frequency	The frequency of the correction constant: `0` = 1 MHz; `1` = 100 kHz
range	The range of the correct constant; see **Details**

Details

If the file specified by *CapCompFile* does not exist, it is created. The path that you specify must exist. When entering the path information, be sure to use two \ characters to separate each directory level. For example, if your cable compensation file is in file `C:\calfiles\590cal.dat`, you would enter `C:\\calfiles\\590cal.dat`.

range values

Range	100 kHz values	1 MHz values
1	2 pF/2 µs	20 pF/200 µs
2	20 pF/20 µs	20 pF/200 µs
3	200 pF/200 µs	200 pF/2 ms
4	2 nF/2 ms	2 nF/20 ms

The return values from *status* can be:

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: An invalid instrument ID was specified. This generally means that there is no instrument with the specified ID in your configuration.
- `-10020 (COMP_FILE_ACCESS_ERR)`: There was an error accessing the cable compensation file.
- `-10021 (COMP_FILE_NOT_EXIST)`: The specified compensation file does not exist.
- `-10022 (KI590_NOT_IN_KCON)`: There is no CMTR defined in your system configuration.

Also see

SaveCableCompCaps590 user module (on page 4-35)

SaveCableCompCaps590 user module

This function saves the nominal values of the capacitors used to do the 590 cable compensation procedure to the indicated file. If no cable compensation file exists, this module creates one if the user has the proper system permissions.

Usage

```
status = SaveCableCompCaps590(char *CabCompFile, double Lo2p100k, double Lo2p1M,
   double Hi2p100k, double Hi2p1M, double Lo20p100k, double Lo20p1M, double
   Hi20p100k, double Hi20p1M, double Lo200p100k, double Lo200p1M, double
   Hi200p100k, double 200p1M, double Lo2n100k, double Lo2n1M, double Hi2n100k,
   double Hi2n1M);
```

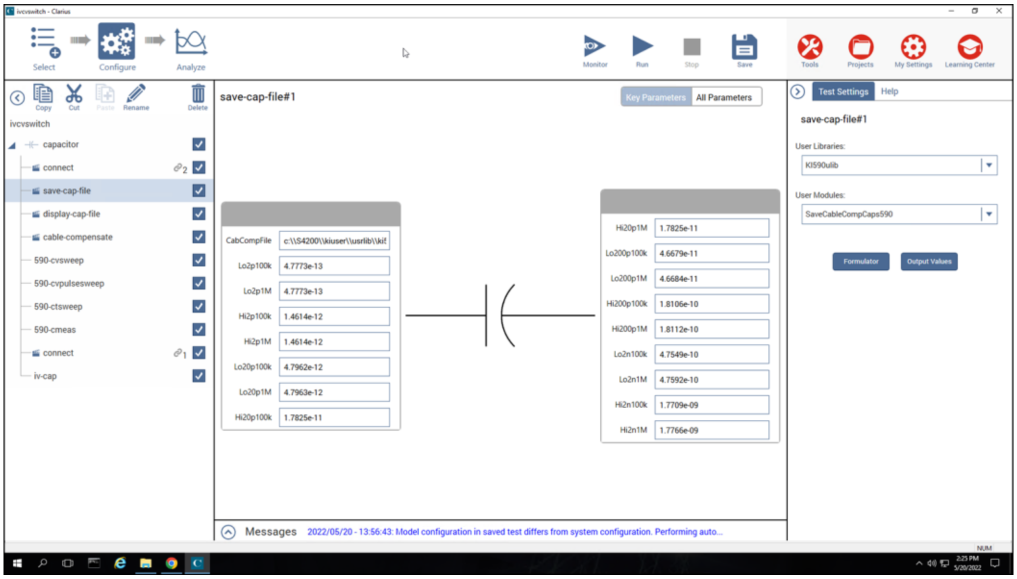
status	Returned values are placed in the Analyze sheet; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
Lo2p100k	The nominal value of the low-range capacitor used for cable compensation for the 2 pF range and 100 kHz frequency: 0 F to 0.95E−12 F
Lo2p1M	The nominal value of the low-range capacitor used for cable compensation for the 2 pF range and 1 MHz frequency: 0 F to 0.95E−12 F
Hi2p100k	The nominal value of the high-range capacitor used for cable compensation for the 2 pF range and 100 kHz frequency: 1E−12 F to 2E−12 F
Hi2p1M	The nominal value of the high-range capacitor used for cable compensation for the 2 pF range and 1 MHz frequency: 1E−12 F to 2E−12 F
Lo20p100k	The nominal value of the low-range capacitor used for cable compensation for the 20 pF range and 100 kHz frequency: 0 F to 9.5E−12 F
Lo20p1M	The nominal value of the low-range capacitor used for cable compensation for the 20 pF range and 1 MHz frequency: 0 F to 9.5E−12 F
Hi20p100k	The nominal value of the high-range capacitor used for cable compensation for the 20 pF range and 100 kHz frequency: 10E−12 F to 20E−12 F
Hi20p1M	The nominal value of the high-range capacitor used for cable compensation for the 20 pF range and 1 MHz frequency: 10E−12 F to 20E−12 F
Lo200p100k	The nominal value of the low-range capacitor used for cable compensation for the 200 pF range and 100 kHz frequency: 0 F to 95E−12 F
Lo200p1M	The nominal value of the low-range capacitor used for cable compensation for the 200 pF range and 1 MHz frequency: 0 F to 95E−12 F
Hi200p100k	The nominal value of the high-range capacitor used for cable compensation for the 200 pF range and 100 kHz frequency: 100E−12 F to 200E−12 F
Hi200p1M	The nominal value of the high-range capacitor used for cable compensation for the 200 pF range and 1 MHz frequency: 100E−12 F to 200E−12 F
Lo2n100k	The nominal value of the low-range capacitor used for cable compensation for the 2 nF range and 100 kHz frequency: 0 F to 995E−12 F
Lo2n1M	The nominal value of the low-range capacitor used for cable compensation for the 2 nF range and 1 MHz frequency: 0 F to 995E−12 F
Hi2n100k	The nominal value of the high-range capacitor used for cable compensation for the 2 nF range and 100 kHz frequency: 1000E−12 F to 2000E−12 F
Hi2n1M	The nominal value of the high-range capacitor used for cable compensation for the 2 nF range and 1 MHz frequency: 1000E−12 F to 2000E−12 F

Details

This user module is used for 590 cable compensation. You enter precise capacitance source values. When this test is run, the capacitance source values are saved to a user-specified file. The user module to perform cable compensation (`CableCompensate590`) can then access the capacitance source values from this file.

The default parameter values for this user module are shown in the following figure. These are suggested low and high values that can be used for cable compensation. You must replace these values with the calibration values of the capacitance sources.

Figure 64: save-cap-file action and SaveCableCompCaps590 user module



Example 1: Cable compensation (on page 4-6) demonstrates how cable compensation is done.

If the file defined for `CabCompFile` does not exist, or there is no path specified (null string), the default compensation parameters are used. When entering the path, use two backslash (`\\`) characters to separate each directory. For example, if your cable file is in `C:\calfiles\590cal.dat`, you enter the following:

```
C:\\calfiles\\590cal.dat
```

NOTE

If a switching matrix to route signals is being controlled by a connection action (for example, `connect`), there is no need to connect *`InputPin`* and *`OutPin`*. Set these parameters to `0`.

The return values from *status* can be:

- 0: OK.
- -10000 (INVAL_INST_ID): The specified instrument ID does not exist. This generally means that there is no instrument with the specified ID in your configuration.
- -10001 (INVAL_PIN_SPEC): An invalid DUT pin number was specified.
- -10003 (NO_SWITCH_MATRIX): No switching matrix was found.
- -10004 (NO_MATRIX_CARDS): No matrix cards were found.
- -10020 (COMP_FILE_ACCESS_ERR): There was an error accessing the specified cable compensation file.
- -10021 (COMP_FILE_NOT_EXIST): The specified compensation file does not exist.
- -10022 (KI590_NOT_IN_KCON): There is no CMTR defined in the configuration of your system.

Also see

CableCompensate590 user module (on page 4-12)

Section 5

Using a Keysight 4284/4980A LCR Meter

In this section:

Introduction

This section contains information on using the 4200A-SCS with the Keysight Models 4284A and 4980A.

For details on Keysight Model 4284A operation, refer to the *Keysight Model 4284A Operation Manual*. For details on Keysight Model 4980A operation, refer to the *Keysight Model 4980A Operation Manual*.

C-V measurement basics

The Keithley 4200A-SCS can control a Keysight 4284A or 4980A LCR Meter to measure capacitance versus voltage (C-V) of semiconductor devices. Typically, C-V measurements are performed on capacitor-like devices, such as a metal-oxide-silicon capacitor (MOS capacitor).

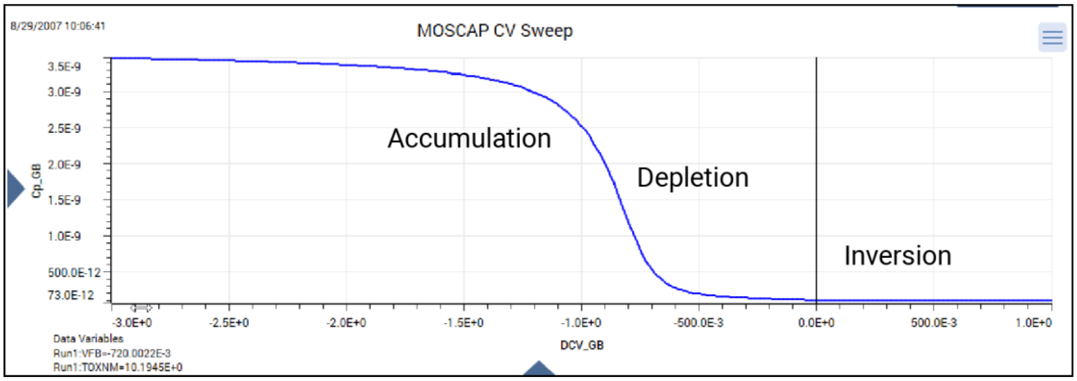
The measurements of MOS capacitors study:

- The integrity of the gate oxide and semiconductor doping profile
- The lifetime of semiconductor material
- The interface quality between the gate oxide and silicon
- Other dielectric materials used in an integrated circuit

A user-configured voltage sweep allows capacitance measurements that can span the three regions of a C-V curve: The accumulation region, depletion region, and inversion region.

The following figure shows the three regions of a typical C-V curve for a MOS capacitor.

Figure 65: Typical C-V curve for a MOS capacitor



Capacitance measurement tests

The 4200A-SCS provides the following user modules to perform C-V tests using the Keysight Models 4284A and 4980A:

- **CvSweep4284.** C-V sweep test: Performs a capacitance and conductance measurement at each step of a user-configured linear voltage sweep.
- **Cmeas4284.** C measurement: Performs a capacitance and conductance measurement at a fixed bias voltage.

Details on the user modules for the Keysight Models 4284A and 4980A are provided in the HP4284ulib User Library Reference (on page 5-8).

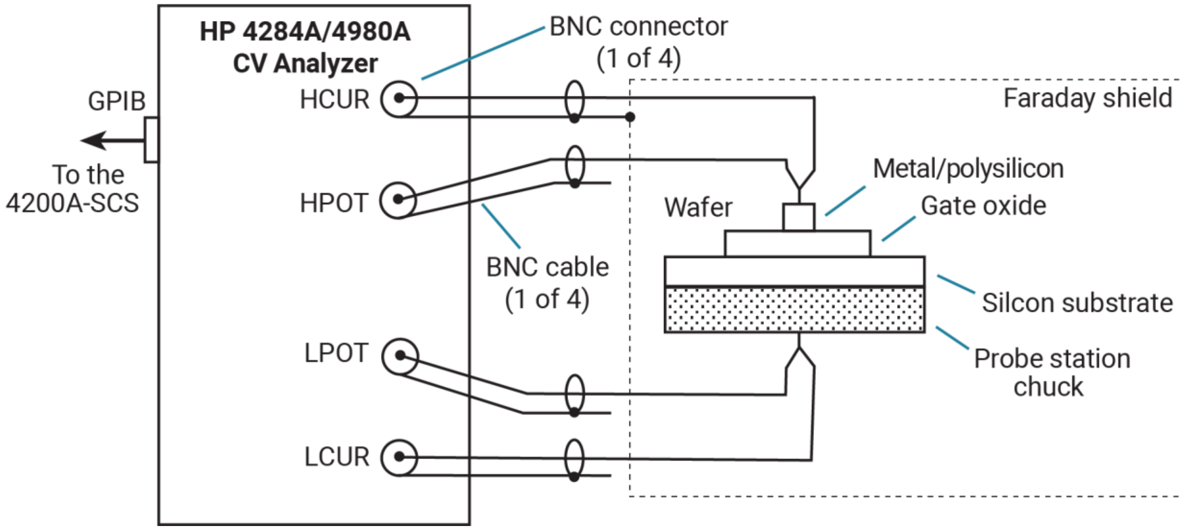
NOTE

If needed, you can initially do an OPEN and SHORT correction on the 4284A or 4980A to achieve the most accurate C-V measurements. See the Keysight 4284A or 4980A *Operation Manual* for details.

Signal connections

Basic 4-wire signal connections for the Model 4284A or 4980A are shown in the following figure. The center conductors of the BNC connectors are connected to the device under test (DUT). The outer shield of one of the coaxial cables is typically connected to a Faraday shield. The Model 4284A or 4980A output is typically connected to the wafer backside (or well). The input is typically connected to the gate of a MOS capacitor.

Figure 66: Basic Model 4284A and 4980A connections to DUT



Triaxial connections

Adapters are required to connect the Model 4284A or 4980A to equipment (for example, a probe station, test fixture, or matrix card) that uses triaxial connectors.

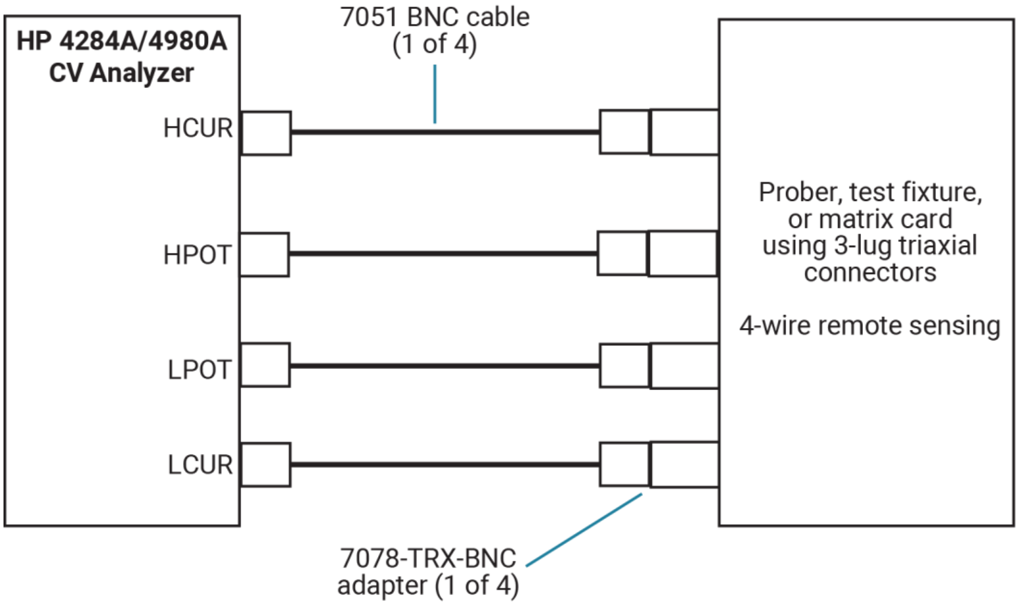
NOTE

See Using Switching Matrices (on page 2-1) for details on using a switching matrix with the Model 4284A or 4980A LCR Meter.

4-wire remote sensing

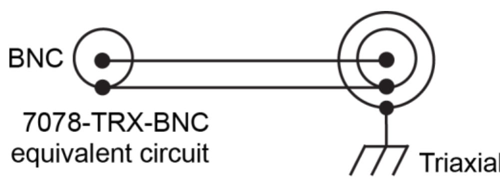
The following figure shows 4-wire remote sense connections. The 7078-TRX-BNC is a 3-lug triaxial-to-BNC adapter. As shown in the figure, connect the adapters to the 3-slot triaxial connectors, and then use a 7051 BNC cable to make the connections to the Model 4284A or 4980A.

Figure 67: 4-wire remote sense connections to equipment using triaxial connectors



The following figure shows the equivalent circuit for the adapter.

Figure 68: 7078-TRX-BNC equivalent circuit



2-wire local sensing

For 2-wire local sense connections, coaxial tees are required to adapt dual BNC cables to single BNC cables, as shown in the following figure.

Figure 69: 2-wire local sense connections to equipment using triaxial connectors

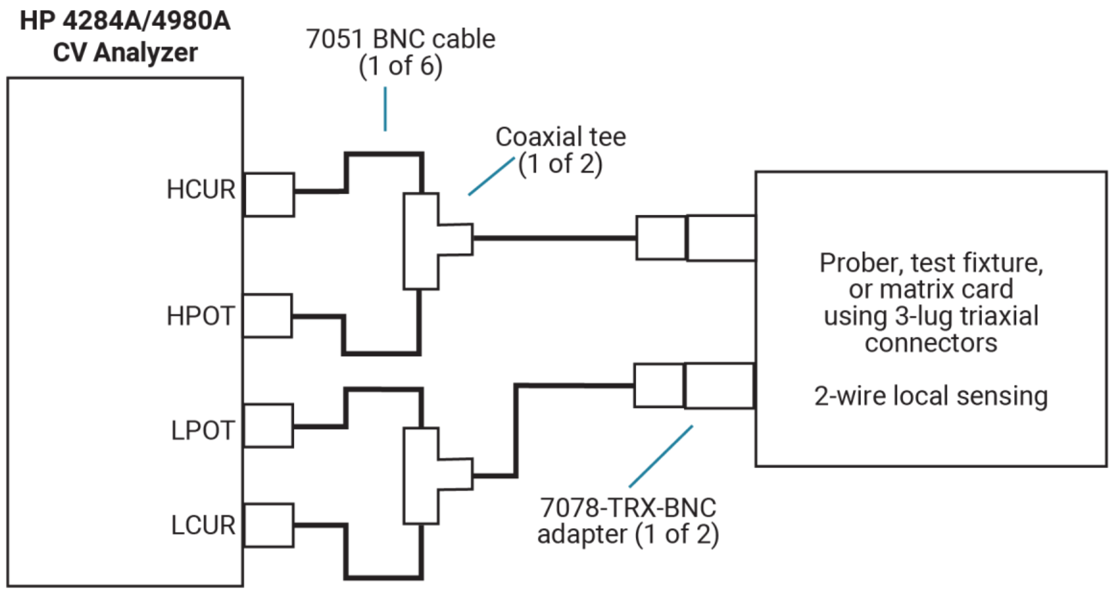
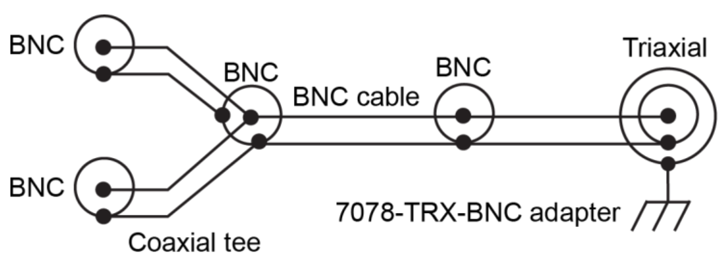


Figure 70: Equivalent circuit



GPIB connections

The 4200A-SCS controls the Model 4284A or 4980A through the general purpose interface bus (GPIB). Use a shielded GPIB cable to connect the GPIB port of the Model 4284A or 4980A to the GPIB port of the 4200A-SCS.

Using KCon to add a Keysight LCR Meter to the system

To use the 4200A-SCS to control an external instrument, you must add the instrument to the 4200A-SCS system configuration. You use Keithley Configuration Utility (KCon) to add the Keysight Model 4284A or 4980A to the test system.

Refer to "Use KCon to add equipment to the 4200A-SCS" in *Model 4200A-SCS Parameter Analyzer Setup and Maintenance* for instruction.

Model 4284A or 4980A C-V sweep test example

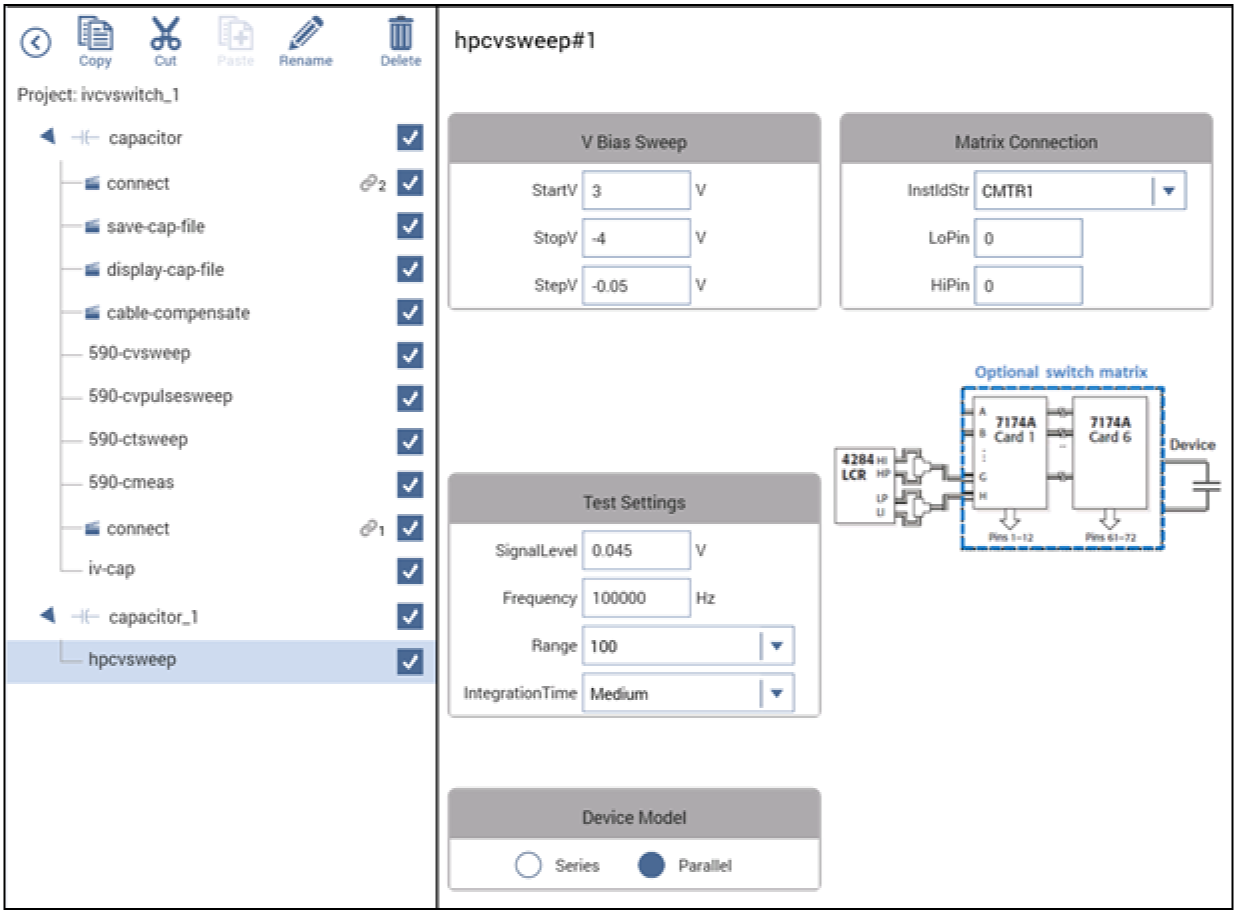
The following test example for the Model 4284A or 4980A LCR Meter is controlled by the `ivcvswitch` project with an added user test module (UTM) created from a user module from the `HP4284ulib` user library. A switching matrix is not used for this example.

This example assumes that the Model 4284A or 4980A is already connected directly to the device under test (DUT). The DUT can be a device installed in a test fixture or a MOS capacitor on a wafer. Complete the following steps to do a C-V sweep.

Set up the project:

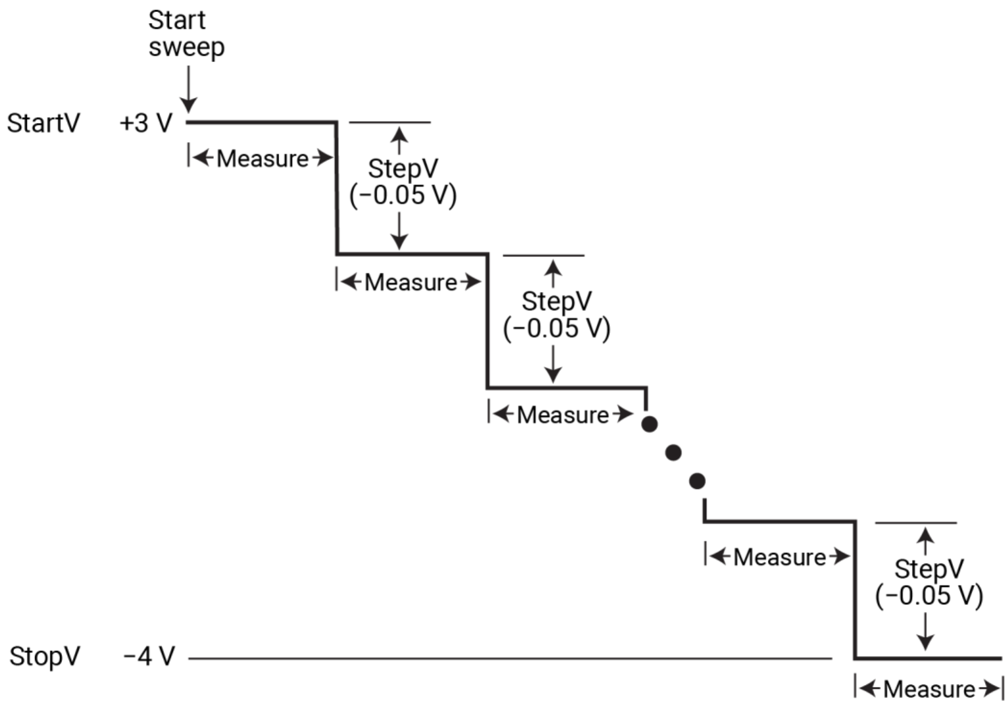
1. From the Project Library, select the **ivcvswitch** project.
2. Select **Create**.
3. At the bottom of the project tree, add another **capacitor** from the Device Library.
4. From the Test Library, select **Custom Test**, **Choose a test from the preprogrammed library (UTM)**.
5. Drag **Custom Test** to the project tree under the capacitor. The test has a red triangle next to it to indicate that it is not configured.
6. Select **Rename**.
7. Enter **hpcvsweep** and press **Enter**.
8. Select **Configure**.
9. In the Test Settings pane, select the user library **HP4284ulib**.
10. Select the user module **CvSweep4284**.
11. On the Configure pane, you can modify the test parameters as needed. With the settings shown in the following figure, the Model 4284A or 4980A does a +3 V to –4 V staircase sweep using 50 mV steps. A measurement is made on each step of the sweep. For details on the test description, see CvSweep4284 User Module (on page 5-9).
12. Select **Run** to execute the test.

Figure 71: CvSweep4284 user module example



A configured sweep is shown in the following figure.

Figure 72: C-V linear staircase sweep



HP4284ulib user library

You use the user modules in the `HP4284ulib` user library to control the Keysight 4284A or 4980A LCR Meter. These user modules are summarized in the following table.

HP4284ulib user library

User module	Description
`Cmeas4284`	Makes a single capacitance measurement.
`CvSweep4284`	Makes capacitance versus voltage measurements using a staircase sweep.

CvSweep4284 User Module

The `CvSweep4284` routine performs a capacitance versus voltage (C-V) sweep using the Keysight Model 4284A or Model 4980 LCR Meter.

Usage

```
status = CvSweep4284(char *InstIdStr, int LoPin, int HiPin, double StartV, double
    StopV, double StepV, double SignalLevel, double Frequency, double Range, int
    Model, int IntegrationTime, double *C, int Csize, double *V, int Vsize, double
    *G_or_R, int G_or_Rsize);
```

status	Returned values; see **Details**
InstIdStr	The CMTR instrument ID; `CMTR1` or `CMTR2`, depending on your system configuration
LoPin	The DUT pin to which the 4284A or 4980 low terminal is attached (−1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
HiPin	The DUT pin to which the 4284A or 4980 high terminal is attached (−1 to 72); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
StartV	Starting voltage of the sweep: −40 V to 40 V
StopV	Ending voltage of the sweep: −40 V to 40 V
StepV	The sweep voltage step size: −40 V to +40 V; the value of ((StopV − StartV)/StepV) + 1 must be less than or equal to the values for *Csize*, *Vsize*, and *G_or_Rsize*
SignalLevel	The oscillator output voltage level (5e−3 V to 20 V)
Frequency	Measurement frequency of the sweep: 20 Hz to 1e6 Hz
Range	The measurement range to use (in ohms): `0` (Auto), `100`, `300`, `1000`, `3000`, `10000`, `30000`, or `100000`
Model	Measurement model: `Series` or `Parallel`
IntegrationTime	The integration time to use: ▪ Short: `0` ▪ Medium: `1` ▪ Long: `2`
C	Output: The measured array of capacitance values
Csize	A value equal to or greater than the *G_or_Rsize* number of steps in the sweep or = ((*StopV* − *StartV*)/*StepV*) + 1; when this function is called from a Clarius UTM, the value is fixed at 1350
V	Output: The array of voltage biases used in the sweep
Vsize	A value equal to or greater than the *G_or_Rsize* number of steps in the sweep or = ((*StopV* − *StartV*)/*StepV*) + 1; when this function is called from a Clarius UTM, the value is fixed at 1350
G_or_R	Output: ▪ When the parallel measurement model (`1`) is selected, *G_or_R* is the measured conductance ▪ When the series measurement model (`0`) is selected, this is the measured resistance
G_or_Rsize	A value equal to or greater than the *G_or_Rsize* number of steps in the sweep or = ((*StopV* − *StartV*)/*StepV*) + 1; when this function is called from a Clarius UTM, the value is fixed at 1350

Details

This user module performs a capacitance versus voltage staircase sweep. For an example of how to run a C-V sweep, see Model 4284A or 4980A C-V Sweep Test Example (on page 5-6). In this example, the Model 4284A or Model 4980A outputs a linear staircase voltage sweep from +3 V to −4 V in 50 mV steps. A capacitance measurement is made on each step of the sweep.

NOTE

If a switching matrix to route signals is being controlled by a connection action UTM (for example, `connect`), there is no need to connect *LoPin* and *HiPin*. Set these parameters to 0.

Returned values are placed in the Analyze spreadsheet.

- `0`: OK.
- `-10030 (HP4284_NOT_IN_KCON)`: No Keysight 4284A or Keysight 4980 LCR is defined in your system configuration.
- `-10031 (HP4284_MEAS_ERROR)`: A measurement error occurred.
- `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred.
- `-10091 (GPIB_TIMEOUT)`: A timeout occurred during communications.
- `-10100 (INVAL_PARAM)`: An invalid input parameter is specified.
- `-10102 (ERROR_PARSING)`: There was an error parsing the response from the Model 4284A or 4980.
- `-10101 (ARRAY_SIZE_TOO_SMALL)`: The specified value for *Csize*, *G_or_Rsize*, *Vsize*, or *Tsize* was too small for the number of steps in the sweep.

Also see

None

Cmeas4284 User Module

The `Cmeas4284` routine measures capacitance and conductance using the Keysight Model 4284A or 4980 LCR Meter.

Usage

```
status = Cmeas4284( char *InstIdStr, int LoPin, int HiPin, double SignalLevel,
    double Frequency, double BiasV, double Range, int Model, int IntegrationTime,
    double *C, double *V, double *G_or_R);
```

status	Returned values; see **Details**
InstIdStr	The CMTR instrument ID; `CMTR1` or `CMTR2` (default), depending on your system configuration
LoPin	The DUT pin to which the Model 4284A or 4980 low terminal is attached (–1 to 72; default 0); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
HiPin	The DUT pin to which the Model 4284A or 4980 high terminal is attached (–1 to 72; default 0); if a value of less than 1 is specified, no switching matrix connection is made; see **Details**
SignalLevel	The oscillator output voltage level: 5 mV to 20 V; default 0.045 V
Frequency	Measurement frequency of the sweep: 20 Hz to 1e6 Hz; default 100e3 Hz
BiasV	The DC bias to use for the measurement: –40 V to +40 V; default 1.0 V
Range	The measurement range to use (in ohms): 0 (Auto, the default), 100, 300, 1000, 3000, 10000, 30000, or 100000
Model	Measurement model: `Series` or `Parallel`
IntegrationTime	The integration time to use: ■ Short: `0` ■ Medium: `1` (default) ■ Long: `2`
C	Output: The measured capacitance
V	Output: The bias voltage used
G_or_R	Output: ■ Parallel measurement model (*G_or_R* is the measured conductance): `1` ■ Series measurement model (*G_or_R* is the measured resistance): `0`

Details

This user module makes a single, fixed-bias capacitance and conductance measurement.

NOTE

If a switching matrix to route signals is being controlled by a connection action UTM (for example, `connect`), there is no need to connect *LoPin* and *HiPin*. Set these parameters to `0`.

Returned values are placed in the Analyze spreadsheet.

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist.
- `-10030 (HP4284_NOT_IN_KCON)`: No Keysight 4284A or Keysight 4980 LCR is defined in your system configuration.
- `-10031 (HP4284_MEAS_ERROR)`: A measurement error occurred.
- `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred.
- `-10091 (GPIB_TIMEOUT)`: A timeout occurred during communications.
- `-10102 (ERROR_PARSING)`: There was an error parsing the response from the Model 4284A or 4980.

Also see

None

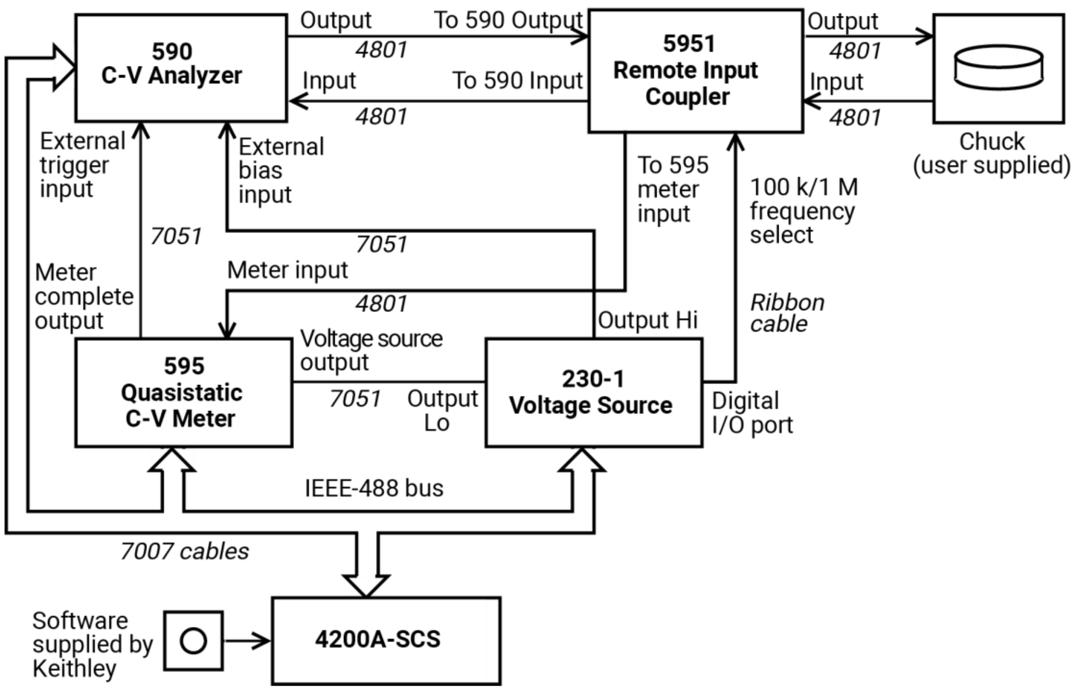
Using a Model 82 C-V System

In this section:

Introduction

The 82 C-V System uses a Keithley 590 C-V Analyzer and a Keithley 595 Quasistatic C-V Meter to make simultaneous C-V measurements. The complete system is shown in the following figure. Projects for the 4200A-SCS are provided to make simultaneous C-V measurements, STVS measurements for mobile ion extraction, and minority carrier generation lifetime measurements.

Figure 73: 82 C-V System block diagram



Capacitance measurement tests

The 4200A-SCS provides the following user modules for capacitance testing using the Model 82:

- **CtSweep82: C-t measurements**: Performs a specified number of capacitance measurements at a specified time interval. Voltage is held constant for these capacitance measurements.
- **SIMCVsweep82: Simultaneous C-V sweep test**: Performs a simultaneous capacitance vs. voltage (C-V) sweep.
- **QTsweep82: Quasistatic capacitance and leakage current test**: Measures quasistatic capacitance and leakage current as a function of delay time to determine the equilibrium condition.

Details on all user modules for the Model 82 are provided in ki82ulib user library (on page 6-31).

C-t measurements

A C-t sweep performs a specified number of capacitance measurements at a specified time interval with voltage held constant. An example of a C-t waveform is shown in the following figure.

When the sweep is started, the device is stressed at a default voltage for a specified period of time. The test bias is then applied and a specified number of capacitance measurements are performed at a specific time interval.

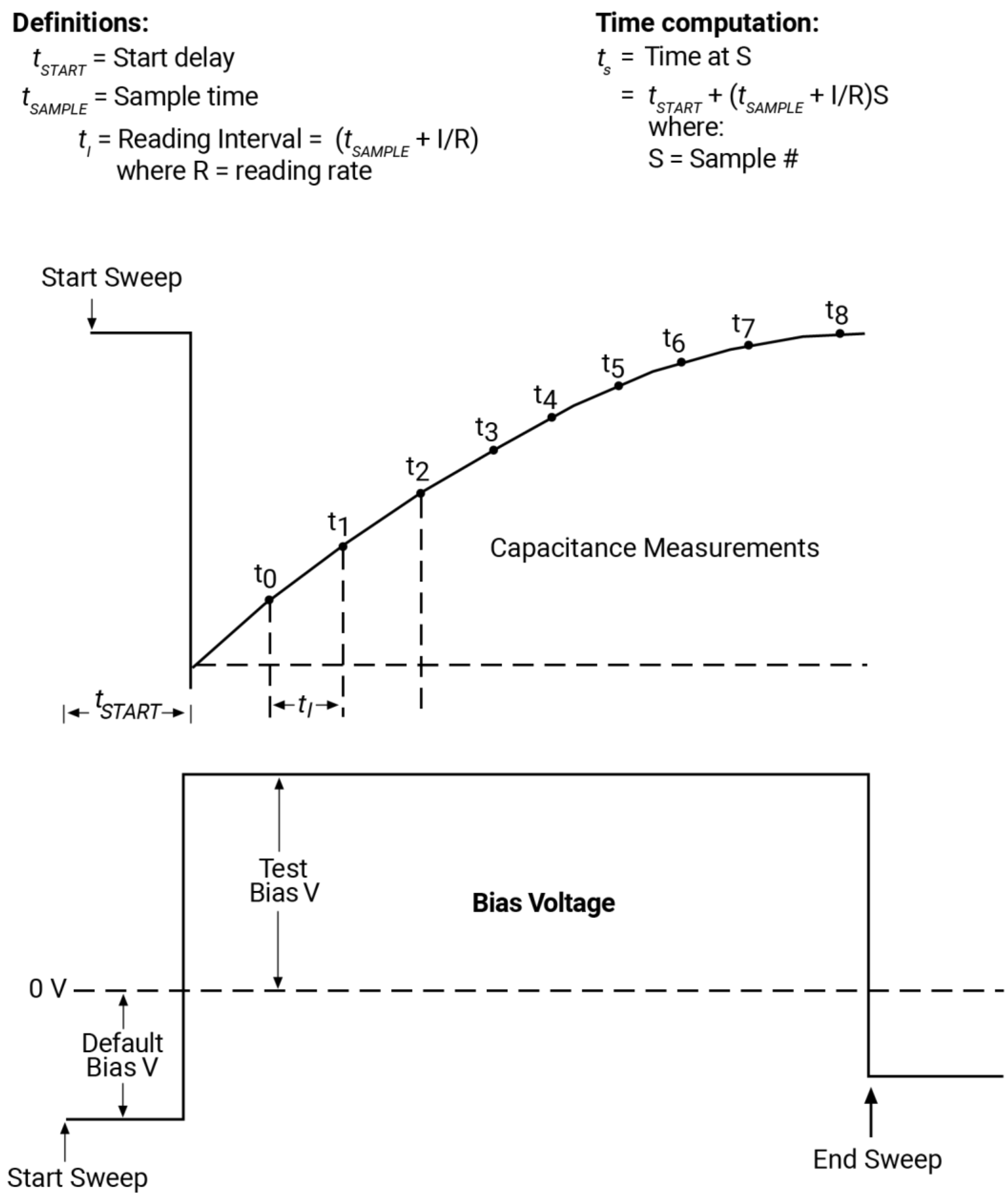
The time interval between each reading is the sum of the specified time between samples (`Sample_Time`) and the reading rate time (as determined by `Reading_Rate`) for each measurement.

NOTE

See Model 82 projects (on page 6-10) for details on the test to perform C-t measurements.

Details on all parameters for the test using the CtSweep82 user module are in the ki82ulib user library (on page 6-31).

Figure 74: C-t waveform



Simultaneous C-V measurements

For simultaneous C-V measurements, the 590 and 595 both measure capacitance during the same voltage sweep. The readings from the two instruments are synchronized using external triggering and are taken alternately during the sweep.

The following figure shows a simplified representation of the stepped bias voltage supplied by the 595 during a measurement sweep. Each vertical voltage step size is determined by the programmed 595 bias step, while each horizontal time step is determined by the programmed delay time.

A quasistatic measurement is a two-step process that requires at least two charge measurements. Initially, at the end of step S_1, the first charge measurement, Q_1, is made, after which the voltage goes to the next step. Following the programmed delay period, the Q_3 charge measurement is made, and the capacitance is then calculated from these values and the step size. Here we see that two voltage steps are necessary for every low-frequency capacitance measurement.

The 590 is triggered one delay time after the completion of each 595 reading. As a result, high-frequency measurements are made on only every other step (as represented by the small rectangles in the waveform figure). Also, the high-frequency measurements are not made at exactly the same voltage as the quasistatic measurements. High-frequency capacitance measurements CH_m and CH_{m+1} are made at voltages VH_m and VH_{m+1}, respectively. Quasistatic measurements result from the charge transfer as the voltage transitions from one step to the next, so that quasistatic capacitance measurement CQ_m is reported at a voltage half-way between the voltages at which its charge measurements Q_1 and Q_3 are made, which is $VQ_m = (V_n - 0.5*V_{step})$.

To compensate for this voltage skew, an adjusted quasistatic capacitance value is calculated by interpolation to correspond to the voltages at which the high frequency measurements were made. The result is a new array of capacitance values CQ'_n corresponding to each high frequency result, CH_n and VH_n.

$$CQ'_n = CQ(VH_n) = CQ_m + \left[\frac{CQ_{m+1} - CQ_m}{VQ_{m+1} - VQ_m}\right] * \frac{Vstep}{2} = CQ_m + \frac{CQ_{m+1} - CQ_m}{4}$$

NOTE

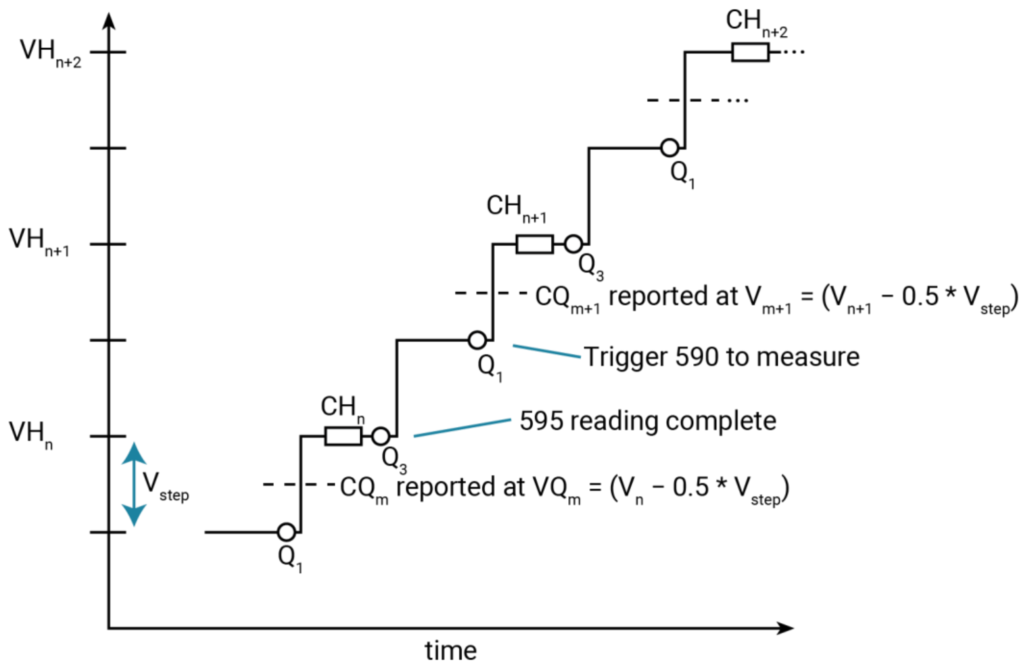
See Model 82 projects (on page 6-10) for details on the test to perform simultaneous C-V measurements.

Details on all parameters for the test are provided in the ki82ulib user library (on page 6-31) for the CVsweep82 user module (on page 6-35).

NOTE

As shown in the following figure, the first high frequency measurement (CH1) is made during the second phase of the voltage sweep. Only quasistatic capacitance (C1) is measured during the first phase and is disregarded.

Figure 75: Simultaneous C-V waveform



In the figure:

- CQ represents 595 quasistatic capacitance readings
- VQ represents voltage reported by the 595 corresponding to CQ
- CH represents Model 590 high frequency capacitance readings
- VH represents the voltage reported by the 590 corresponding to CH

Cable compensation

Ideally, the Model 82 would only measure the capacitance of the DUT. However, signal pathways through the test cables, switching matrix, test fixture, and prober contribute unwanted capacitances that may adversely affect the measurement.

To correct for these unwanted capacitances, cable compensation should be done before measuring the capacitance of the DUT. In general, compensate for cables by connecting precisely known capacitance sources in place of the DUT and then measuring them. The Model 590 then uses these measured values to correct for unwanted capacitance when measuring the DUT.

Cable compensation involves these steps:

1. The Model 82 calculates the compensation parameters based on the comparison between the given and measured values.
2. The Model 82 performs a probe-up offset measurement and suppresses any remaining offset capacitance. This step is done every time a new measurement is made.

Typically, cable compensation is done for all four measurement ranges (2 pF, 20 pF, 200 pF, and 2 nF) of the Model 590. Once cable compensation is done, it does not have to be done again unless the connection scheme to the DUT is changed or power is cycled.

Cable compensation user modules

The Model 82 user modules for cable compensation are:

- **SaveCableCompCaps82 (on page 6-43): Enter and save capacitance source values:** The user enters the actual capacitance values of the capacitance sources. When the test is executed, the capacitance values are stored in a file at a user-specified directory path.
- **DisplayCableCompCaps82 (on page 6-38): Places capacitance values into the Analyze spreadsheet:** When this test is executed, the capacitance values saved by `SaveCableCompCaps82` are placed into the Analyze spreadsheet.
- **CableCompensate82 (on page 6-32): Performs cable compensation:** The user specifies the ranges and test frequencies for cable compensation. When this test is executed, on-screen prompts guide you through the cable compensation process.
- **CabCompFile**: Each of the user modules for cable compensation uses a cable compensation file to save and load capacitor source values. Therefore, these user modules must use the same file directory path.

Connections

The system block diagram in the Introduction (on page 6-1) shows the overall system configuration for the Model 82. Connect all cables as shown in the diagram.

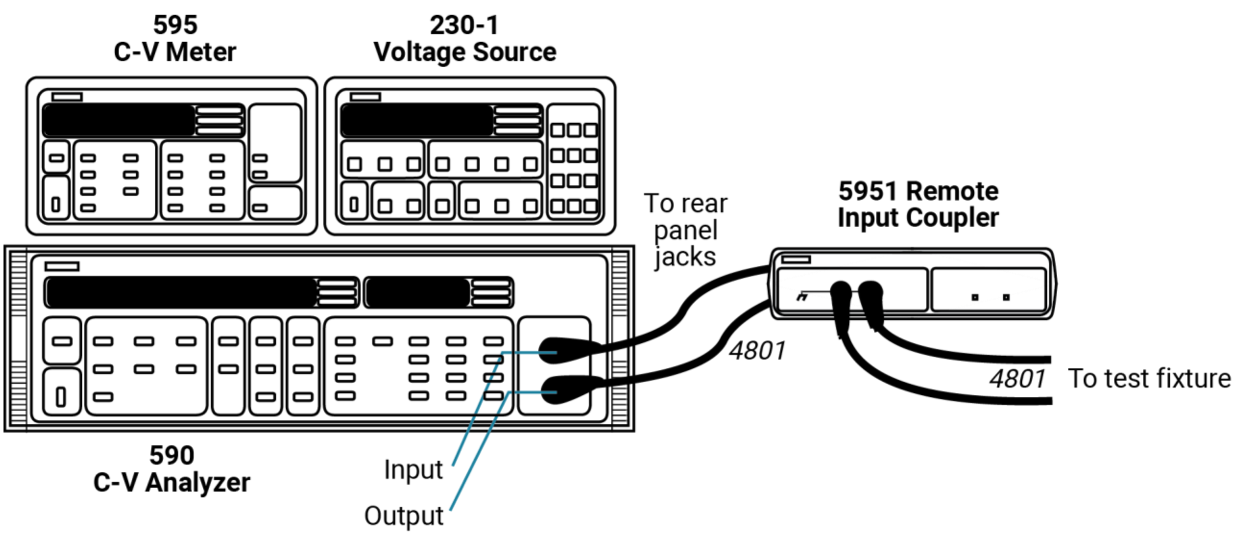
Front-panel connections

The following front-panel connections figure shows the connections from the Model 5951 Remote Input Coupler to the Model 590.

To make front-panel connections:

1. Take one low-noise Model 4801 BNC cable and connect the 590 INPUT on the front of the Model 590 to the TO 590 INPUT on the back of the Model 5951.
2. Use another Model 4801 cable and connect the 590 OUTPUT, also on the front of the Model 590, to the TO 590 OUTPUT on the back of the Model 5951.
3. Connect a low-noise cable from the front-panel 5951 output to the substrate.
4. Connect a low-noise cable from the front-panel 5951 input to the substrate.
5. Connect the dark box to the cable grounds only. If this is not possible, connect a #18 AWG wire between the dark box and the white banana jack on the back of the Model 595.

Figure 76: System 82 C-V system front-panel connections



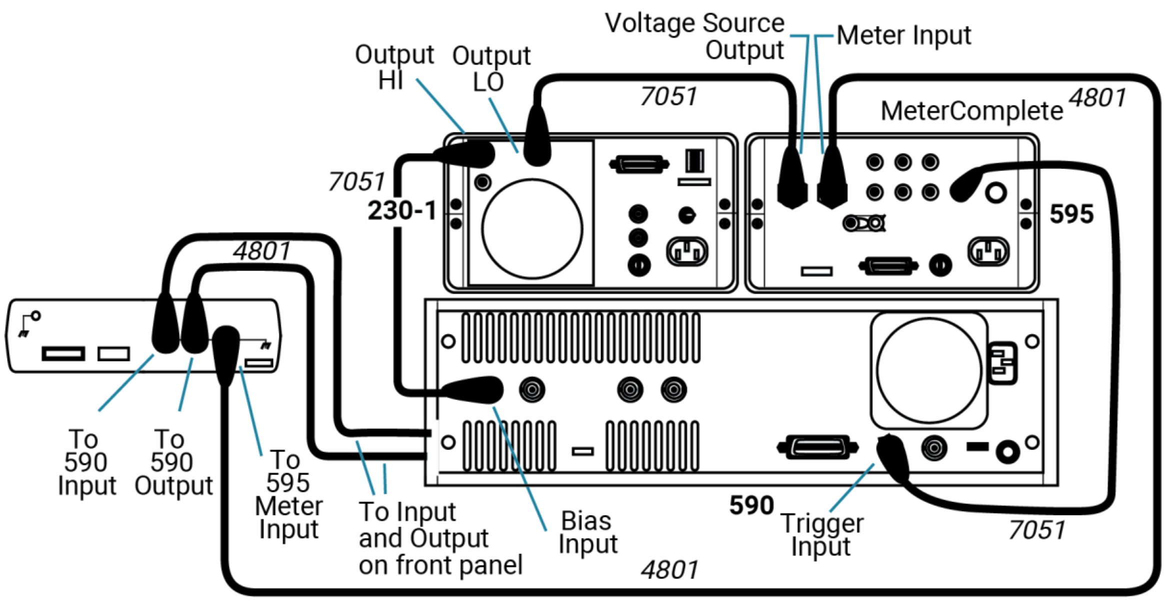
Rear-panel connections

The following rear-panel connections figure shows the rest of the main cabling configuration.

To make rear-panel connections:

1. Use a Model 4801 cable to connect the METER INPUT on the back of the Model 595 to the TO 595 INPUT on the Model 5951.
2. Use a Model 7051 BNC cable to connect the METER COMPLETE port on the back of the Model 595 to the TRIGGER INPUT on the back of the Model 590.
3. Use a Model 7051 cable and connect the OUTPUT HI on the back of the Model 230-1 to the BIAS INPUT on the back of the Model 590.
4. Use a Model 7051 BNC cable to connect the OUTPUT LO on the back of the Model 230-1 to the VOLTAGE SOURCE OUTPUT on the back of the Model 595.

Figure 77: System 82 C-V system rear-panel connections

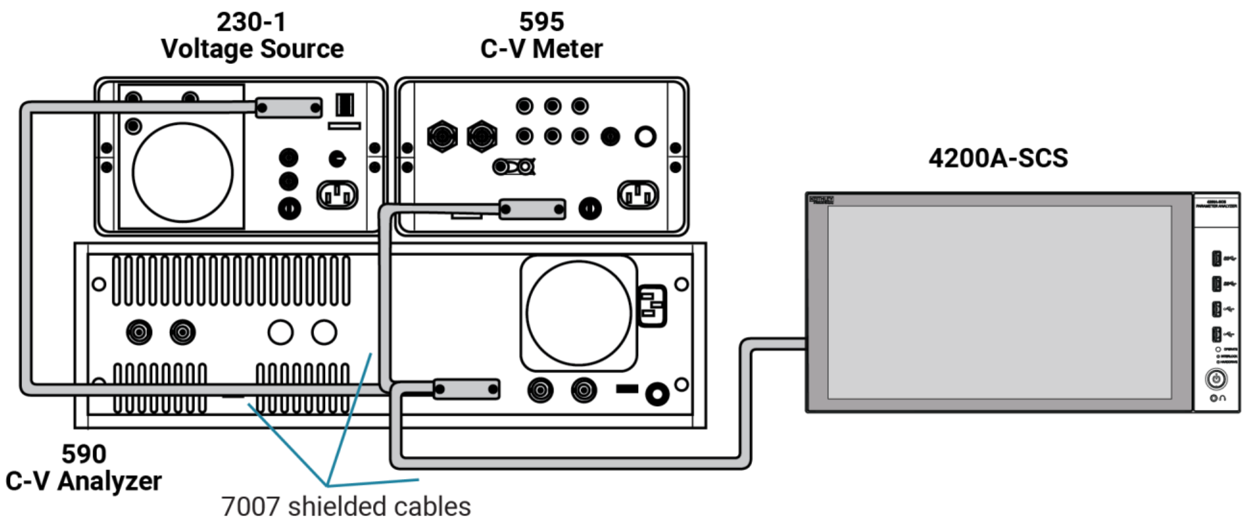


Make power and GPIB connections

To attach the power and GPIB connections:

1. Use the ribbon cable to connect the DIGITAL I / O PORT on the back of the Model 230-1 to the TO 230-1 DIGITAL I / O on the back of the Model 5951.
2. Use the power cables to plug in the units.
3. The following figure shows the connections for the GPIB bus cables. Use the GPIB bus cables and connect the Model 590, the Model 595, and the Model 230-1 to the 4200A-SCS through the GPIB card.

Figure 78: System 82 IEEE-488 connections



Using KCon to add Model 82 C-V System

To use the 4200A-SCS to control instruments in the C-V system, you must add the system to the 4200A-SCS system configuration. You use the Keithley Configuration Utility (KCon) to add the 82 C-V system to the test system.

Refer to “Use KCon to add equipment to the 4200A-SCS” in *Model 4200A-SCS Parameter Analyzer Setup and Maintenance*.

Model 82 projects

The Model 82 projects are:

- `simcv`: This project uses the `qtsweep` test (`QTsweep82` user module) to make quasistatic capacitance measurements. This test optimizes delay time for quasistatic measurements so that the entire simultaneous C-V test is done at DUT equilibrium. Then the `system82-cvsweep` test (`SIMCVsweep82` user module) makes simultaneous C-V measurements.
- `stvs`: This project uses the `ThermalChuck` test to prompt the user to increase the temperature of the thermal chuck, and then uses the `cvsweep` test (`SIMCVsweep82` user module) to make simultaneous C-V measurements.
- `lifetime`: This project uses the `cvsweep` test (`SIMCVsweep82` user module) to make simultaneous C-V measurements, and then uses the `ctsweep` test (`CTsweep82` user module) to make C-t measurements at the condition determined by the `cvsweep` test.

Each project begins by performing cable compensation.

NOTE

Details on all parameters for the compensation and capacitance tests are provided in the ki82ulib user library (on page 6-31).

Cable compensation tests

Complete the following steps to do cable compensation.

These tests assume that the calibration capacitors are installed as close to the wafer-chuck end of the cable as possible.

NOTE

The user modules for cable compensation must share the same file for capacitance source values. Therefore, the same file directory path must be used in all three user modules. For this example, use the default file directory path (see line 1 of the parameter lists for the user modules).

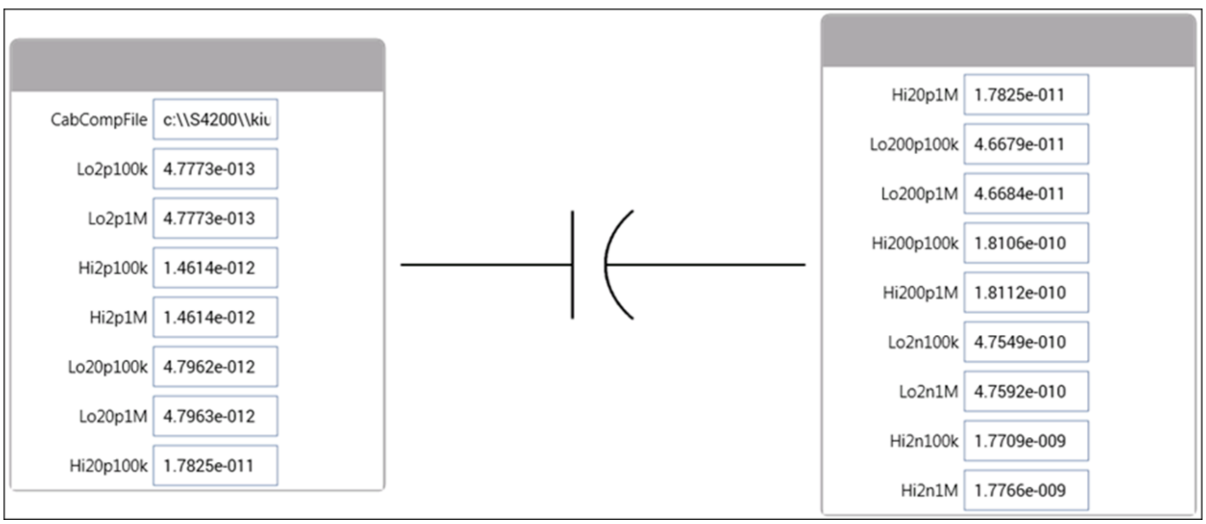
Enter and save capacitance source values (SaveCableCompCaps82)

To enter and save capacitance source values:

1. Open the project.
2. Select **save-cap-file** or **savecablecompfile**. These actions use the `SaveCableCompCaps82` user module
3. Select **Configure**. The Configure pane for these actions is shown in the following figure.
4. In the parameter list, enter the capacitance source calibration value for each range and frequency. For example, assume the low capacitance source for the 2 pA range is 0.47773 pF (100 kHz) and 0.47786 pA (1 MHz). Enter these values, using scientific notation:

   Lo2p100k: Enter **0.47773e-12**
   Lo2p1M: Enter **0.47786e-12**
5. Select **Run** to execute the test. The capacitor source values entered into the action are saved in the file using the directory path specified in the `CapCompFile` field.

Figure 79: SaveCableCompCaps82 user module

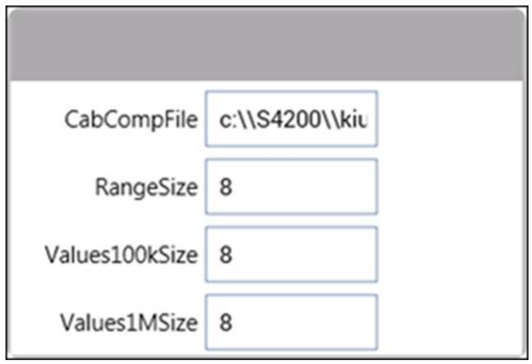


Place capacitance source values in a spreadsheet (DisplayCableCompCaps82)

To place capacitance source values in a spreadsheet:

1. In the project tree, select **display-cap-file** or **displaycablecomp**. The parameter list for the `DisplayCableCompCaps82` user module is shown in the following figure.

Figure 80: DisplayCableCompCaps82 user module



2. Ensure that the CabCompFile field has the same file directory path that is used in Enter and save capacitance source values (SaveCableCompCaps82) (on page 6-11).
3. Set the other fields to **8**.
4. Select **Run**. The calibration source values entered into the action are placed into its spreadsheet.
5. Select **Analyze**. The sheet displays the values.

Figure 81: display-cap-file spreadsheet with capacitor source values

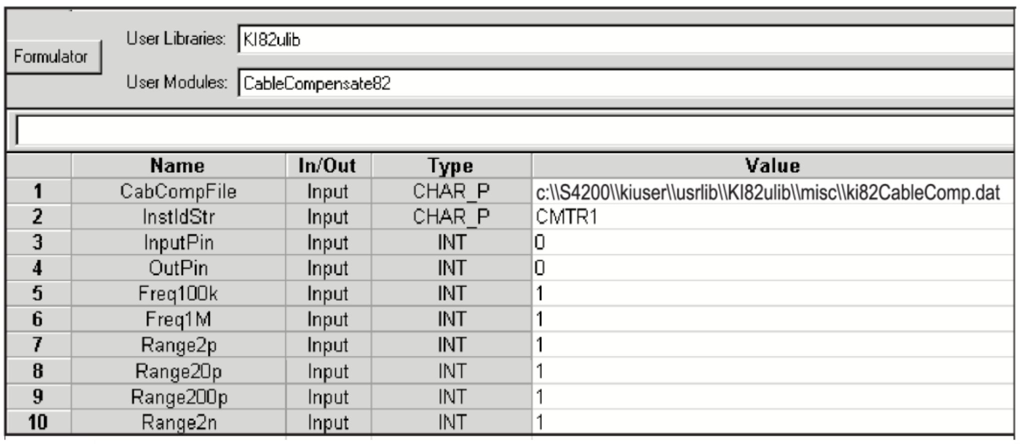
	A	B	C	D
1	DisplayCable	Range	Values100k	Values1M
2	0	2.00000E-12	4.77700E-13	4.77700E-13
3		2.00000E-12	1.46100E-12	1.46100E-12
4		2.00000E-11	4.79600E-12	4.79600E-12
5		2.00000E-11	1.78300E-11	1.78300E-11
6		2.00000E-10	4.66800E-11	4.66800E-11
7		2.00000E-10	1.81100E-10	1.81100E-10
8		2.00000E-09	4.75500E-10	4.75900E-10
9		2.00000E-09	1.77100E-09	1.77600E-09
10				

Compensate for cable capacitance (CableCompensate82)

To compensate for cable capacitance:

1. In the project tree, select **cable-compensate** or **cablecomp**.
2. Select **Configure**.
3. Ensure that the `CabCompFile` field of the parameter list has the same file directory path that is used in Enter and save capacitance source values (on page 6-11).
4. Enable or disable cable compensation: Use the frequency and range fields to either disable (`0`) or enable (`1`) cable compensation for the test frequencies and ranges. The following figure shows cable compensation enabled for all ranges and test frequencies.

Figure 82: CableCompensate82 user module



	Name	In/Out	Type	Value
1	CabCompFile	Input	CHAR_P	c:\\S4200\\kiuser\\usrlib\\KI82ulib\\misc\\ki82CableComp.dat
2	InstIdStr	Input	CHAR_P	CMTR1
3	InputPin	Input	INT	0
4	OutPin	Input	INT	0
5	Freq100k	Input	INT	1
6	Freq1M	Input	INT	1
7	Range2p	Input	INT	1
8	Range20p	Input	INT	1
9	Range200p	Input	INT	1
10	Range2n	Input	INT	1

5. Select **Run** to execute the test.
6. Follow the instructions on the dialogs, which guide you through the cable compensation process. The basic dialogs are shown in the following figure.
   - **Measure offset:** An open circuit measurement is required. Open the circuit as close to the DUT as possible.
   - **Measure capacitance source:** Connect a capacitance source in place of the DUT. The value in the dialog corresponds to a calibration value you entered in Enter and save capacitance source values (on page 6-11). Connect the capacitance source as close to the DUT as possible.
   - **Compare readings:** Compares the measured value to the calibration (nominal) value you entered. The two readings should be fairly close. If they are not, the wrong capacitance source may have been connected or an open circuit condition occurred. In that case, select **Cancel** to abort the cable compensation process.

Figure 83: Cable compensation dialogs

NOTE

Selecting **Cancel** in a cable compensation dialog aborts the cable compensation process. To start over, select **Run**.

Capacitance tests

The following topics describe the user modules that can be made into tests Clarius.

QTsweep (equilibrium test)

To achieve accurate simultaneous C-V test results, measurements must be made with the device in equilibrium. A device is considered to be in equilibrium when all internal capacitances are fully charged before measuring the capacitance. For a fully charged capacitor, any measured current is leakage.

After voltage step is applied to the device, a delay time is used to allow capacitances to fully charge before measuring quasistatic capacitance. If the delay time is too short (capacitors still charging), the quasistatic capacitance measurement will not be accurate. This test allows you to determine the delay time required to achieve equilibrium.

This example assumes that the Model 82 is connected directly to the DUT. The DUT could be a device installed in a test fixture, or a substrate on a wafer.

To open and execute the QTsweep UTM:

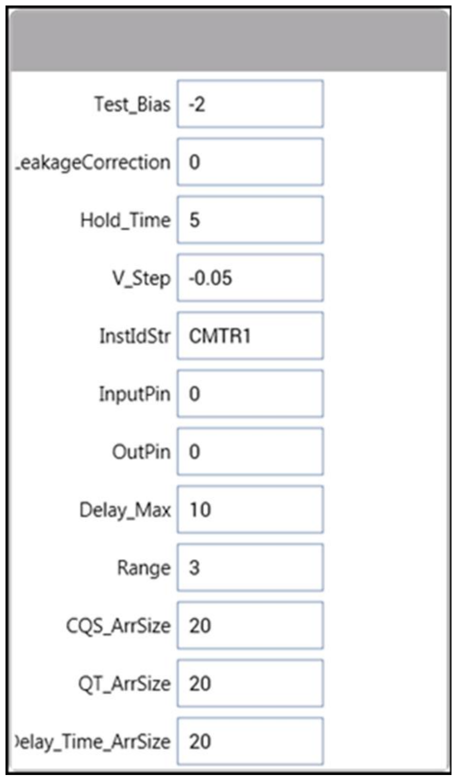
1. Choose **Select**.
2. In the Test Library, select Custom Test, **Choose a test from the preprogrammed library (UTM)**.
3. Drag the test to the project tree.
4. Select **Configure**.
5. In the Test Settings pane, under User Libraries, select **ki82ulib**.
6. Under User Modules, select **QTsweep82**.

7. Modify the test parameters as needed. Refer to QTsweep82 user module (on page 6-40) for parameter definitions.

   If you use the parameters shown in the following figure, the Model 82 makes 20 quasistatic capacitance measurements using 20 mV pulses (`V_Step`) ranging from 0.07 seconds to 1 second (`Delay_Max`).

8. Select **Run** to execute the test.

Figure 84: QTsweep82 user module

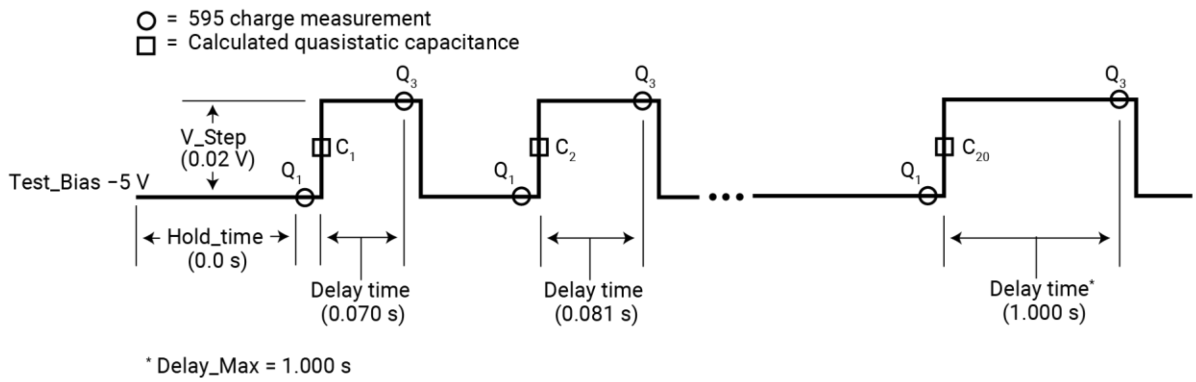


Equilibrium test (QTsweep) description

This test performs a quasistatic capacitance measurement (CQS) using 20 different delay times. The voltage bias and pulse amplitude are held constant during the test. The current (Q / t) at the end of each reading sample is also calculated ($i = \Delta Q/\Delta t$).

The following figure shows the pulse stream for the equilibrium test using the parameters shown in QTsweep (equilibrium test) (on page 6-14). As shown, the last reading sample uses a set delay (`Delay_Max`) of one second, while the first reading sample uses a delay of 70 ms (which is the minimum). The delay times for the other 18 reading samples are then automatically set. After the first pulse, each subsequent pulse delay time increases logarithmically in progression up to the maximum delay (`Delay_Max`).

Figure 85: Equilibrium test

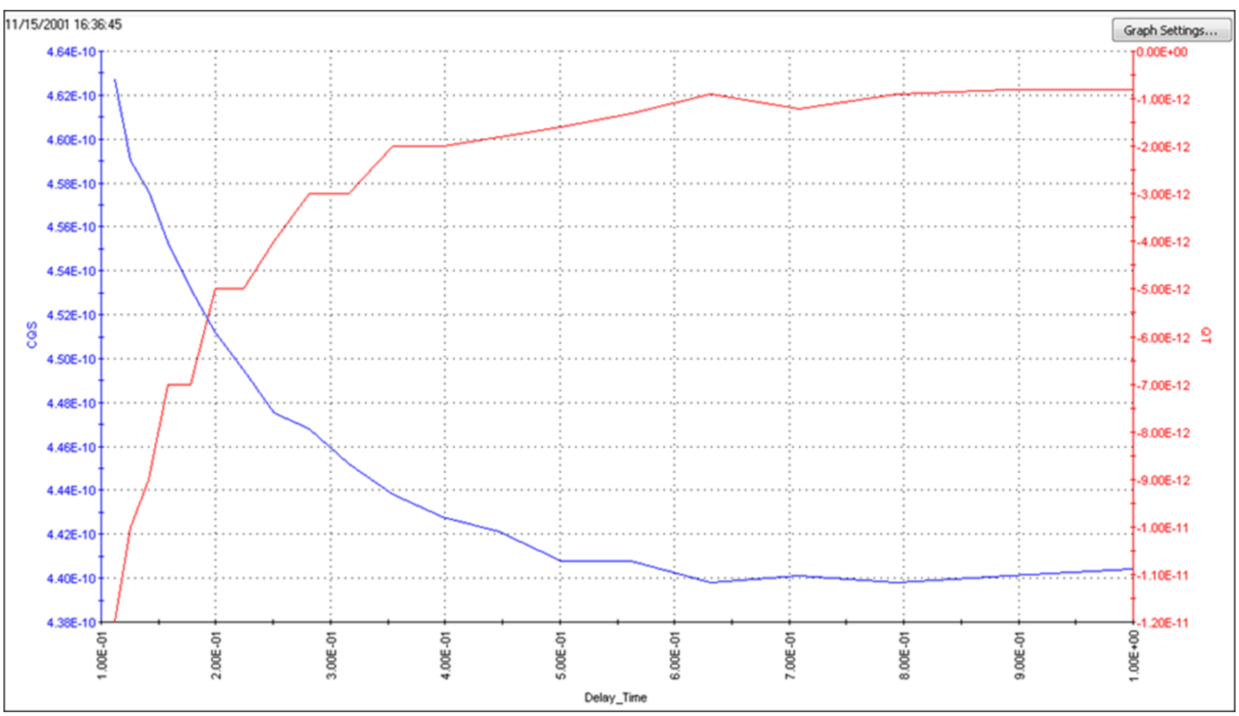


The generated graph for this test plots:

- Quasistatic capacitance (CQS) vs. delay time
- Leakage current (Q / t) vs. delay time

A typical graph for the equilibrium test is shown in the following figure. The optimal delay time occurs when both curves flatten out to a slope of zero. For maximum accuracy, choose the second point on the curves after they have flattened out.

Figure 86: Equilibrium test graph



Simultaneous C-V sweep

The Model 82 uses the Models 595 and 590 to perform simultaneous C-V measurements. Refer to Simultaneous C-V measurements (on page 6-4) for details on simultaneous C-V measurements.

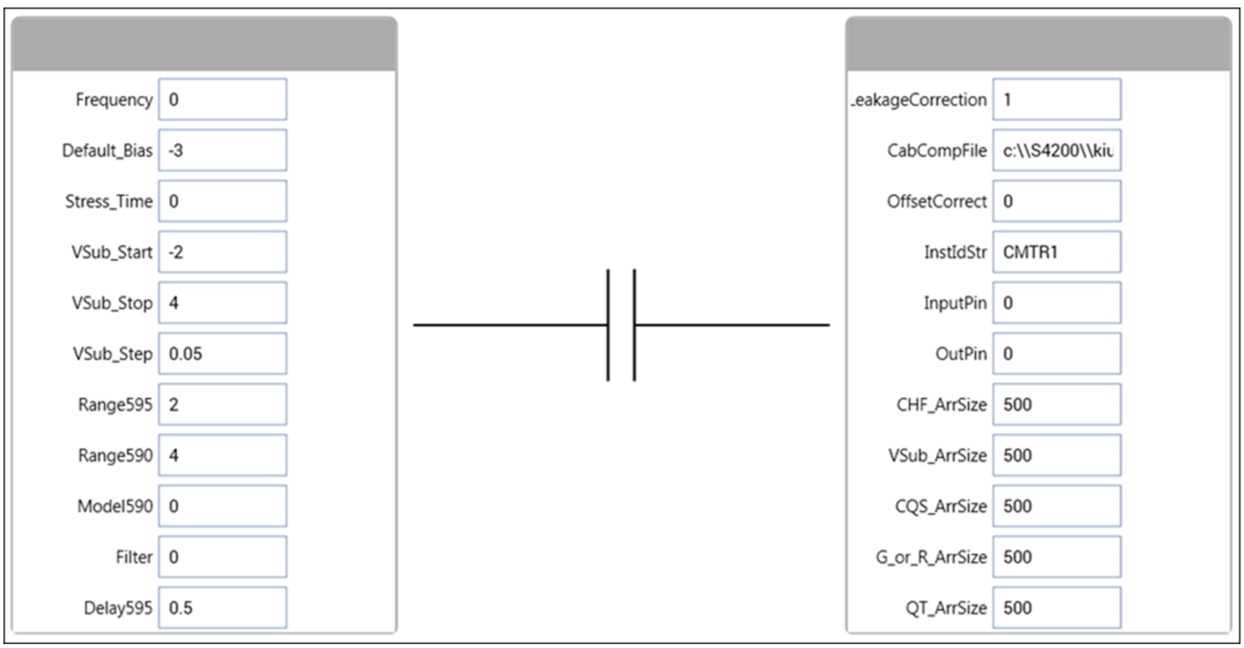
This example assumes that the Model 82 is connected directly to the DUT. The DUT could be a device installed in a test fixture, or a substrate on a wafer.

To do a simultaneous C-V sweep:

1. In the project tree, select **cvsweep**.
2. Select **Configure**.
3. Modify the test parameters as needed. Refer to SIMCVsweep82 user module (on page 6-46) for definitions.
4. Select **Run**.

If you use the parameters shown in the following figure, the Model 82 performs a –3 C to +3 V staircase sweep using 20 mV steps, delaying 70 ms on each step.

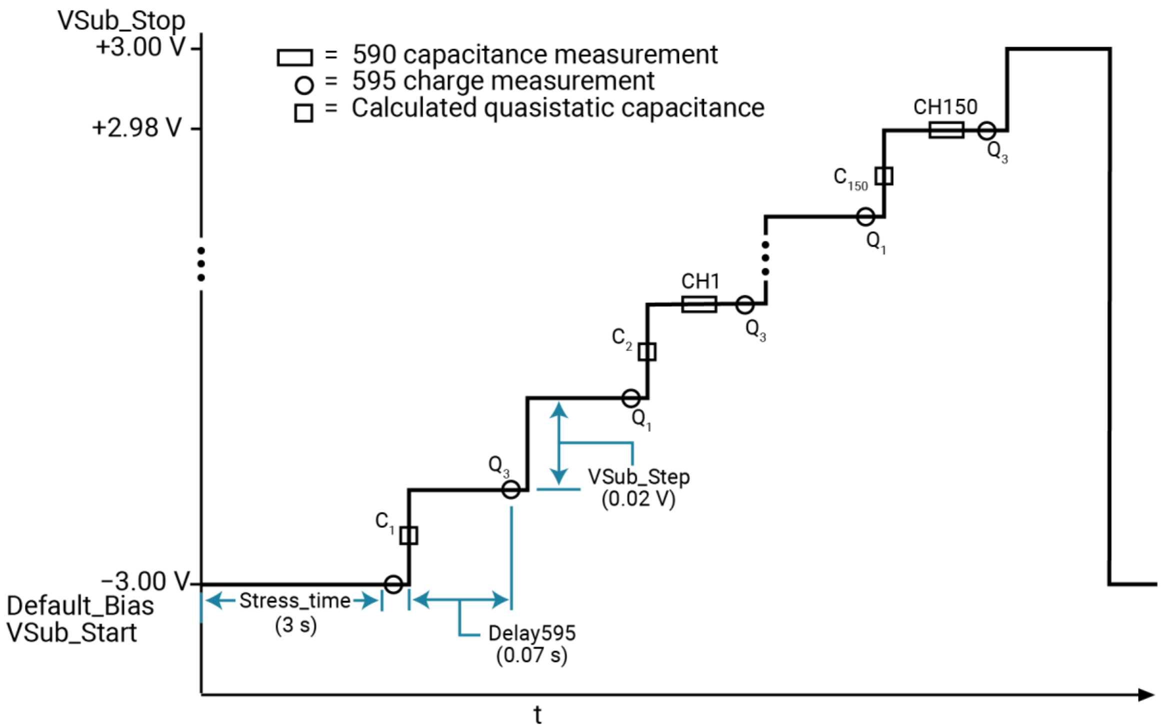
Figure 87: SIMCVsweep82 user module



cvsweep test description

As described in Simultaneous C-V sweep (on page 6-17), the cvsweep UTM uses the `SIMCVsweep82` user module to make simultaneous C-V measurements. A 595 quasistatic measurement is a two-step process that requires at least two charge measurements. As shown in the following figure, charge measurements on two steps are made to yield a single quasistatic reading. The 590 makes a capacitance measurement on every second step of the staircase sweep.

Figure 88: Simultaneous C-V linear staircase sweep

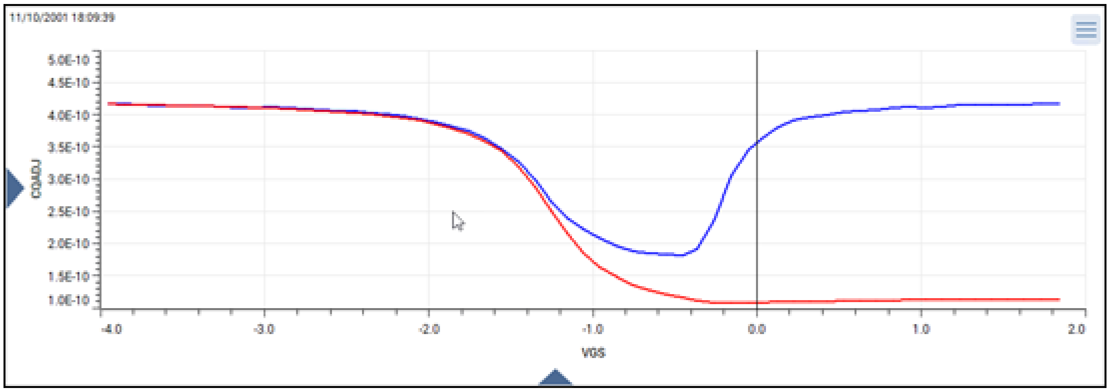


The graph for this test plots 590 capacitance (red trace) and 595 quasistatic capacitance (blue trace) versus bias voltage. The following figure shows a typical graph that is generated by this test.

NOTE

The shape of the curves in the following figure indicate that measurements were made with the device in equilibrium. If the curves for your test deviate significantly, the device was probably not in equilibrium. Do the equilibrium test (`QTsweep82`) to determine the optimum delay time (`Delay595` parameter) to use for the `simcv` test (`SIMCVsweep82` user module).

Figure 89: cvsweep graph



C-t sweep

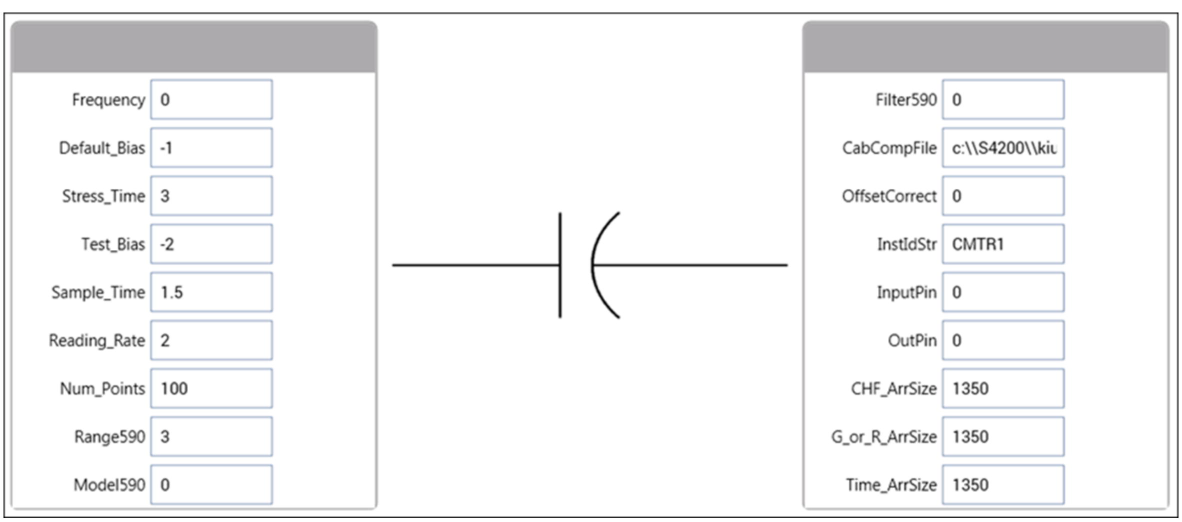
The Model 82 uses the Model 590 to make a specified number of capacitance measurements using a specified time interval between reading samples. The specified voltage bias is held constant for this test. Details on simultaneous C-t measurements are provided in C-t measurements (on page 6-2).

This example assumes that the Model 82 is connected directly to the DUT. The DUT can be a device installed in a test fixture or a substrate on a wafer.

To perform a CtSweep:

1. In the project tree, select **ctsweep**.
2. Select **Configure**.
3. Modify the test parameters as needed. If you use the parameters shown in the following figure, the Model 82 makes 100 capacitance measurements.
4. Select **Run**.

Figure 90: CtSweep82 user module

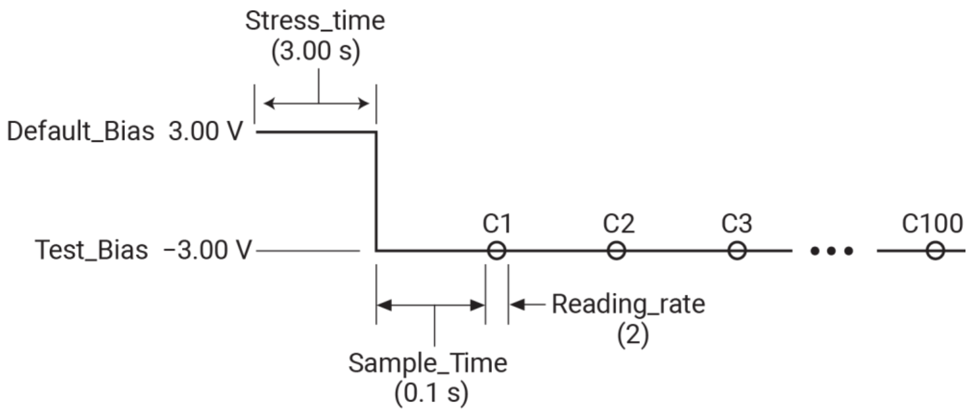


CtSweep test description

As shown in C-t sweep (on page 6-19), the `CtSweep` UTM uses the **CTsweep82** user module to make C-t measurements. Refer to CtSweep82 user module (on page 6-35) for definitions of the input parameters.

If using the parameters shown in C-t sweep (on page 6-19), the 590 performs 100 capacitance measurements using a 100 ms sample time between reading samples. The time interval between reading samples is determined by the set sample time and the selected reading rate.

Figure 91: C-t measurements

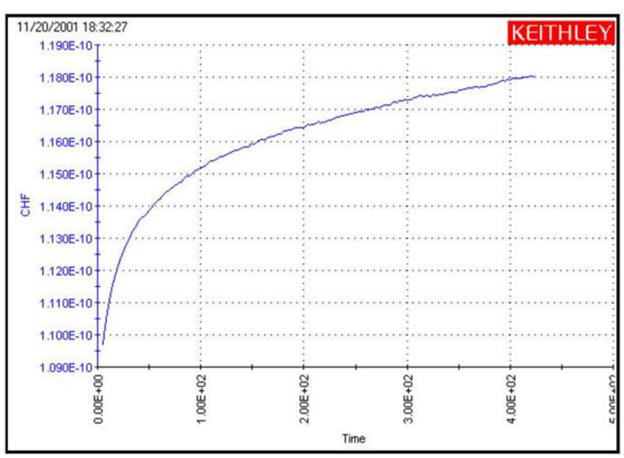


The time interval between reading samples is the sum of the set Sample_Time and the Reading_rate time. The Reading_rate time for Reading_rate is 1/18 s (0.0555 s). Therefore:

Time interval = Sample_Time + Reading_rate
= 0.100 + 0.056
= 0.156 s

The graph below plots capacitance versus time for this test.

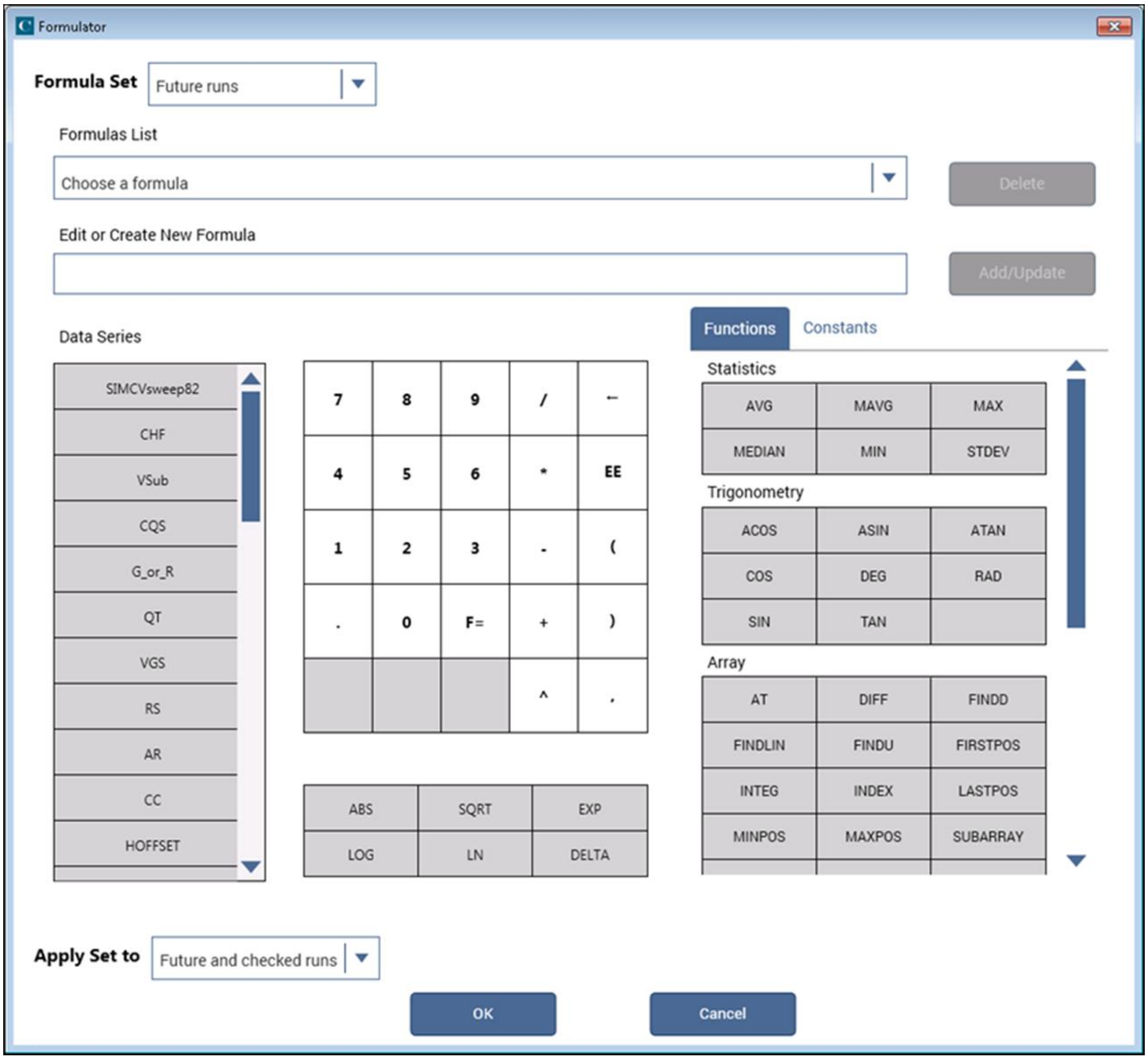
Figure 92: CtSweep graph



Formulas for capacitance tests

Formulas to calculate data for graphs are in the Formulator for each test. To open the Formulator dialog, select **Formulator** in the Test Settings pane for the selected test. The following figure shows the Formulator for the `system82-cvsweep` test used in the `simcv` project.

Figure 93: Formulator for system82-cvsweep test (simcv project)



Formulas for the `system82-cvsweep` test (`simcv` project) are shown in Formulas for system82-cvsweep test (simcv project) (on page 6-23).

Formulas for `ctsweep` test (`lifetime` project) are shown in Formulas for ctsweep test (lifetime project) (on page 6-24)

Formulas for `cvsweep` test (`stvs` project) are shown in Formulas for cvsweep test (stvs project) (on page 6-25).

The values for constants used in the formulas are in the Constants area in the Formulator. The constants include:

- Area, with a value of 0.012 mm^2
- eOX, with a value of 3.4e-013 F/cm
- eS, with a value of 1.04e-012 F/cm

NOTE

Refer to Simultaneous C-V analysis (on page 6-49) for details on simultaneous C-V theory and the formulas.

Formulas for system82-cvsweep test (simcv project)

Formula name	Description and formula
VGS	Gate voltage: VGS = −VSub
RS	Serial resistance calculated by high frequency CV: RS = (AT(MAVG(G_OR_R,5)/(WF*MAVG(CHF,5)),MAXPOS(MAVG(CHF,5))))^2/ ((1+(AT(MAVG(G_OR_R,5)/(WF*MAVG(CHF,5)),MAXPOS(MAVG(CHF,5))))^2) *(AT(MAVG(G_OR_R,5),MAXPOS(MAVG(CHF,5)))))
AR	Intermediate parameter for calculation of CC: AR = G_OR_R-(G_OR_R^2+(WF*CHF)^2)*RS
CC	Corrected high frequency capacitance by compensating serial resistance: CC = ((G_OR_R^2+(WF*CHF)^2))*CHF/(AR^2+(WF*CHF)^2)
HOFFSET	Offset for high frequency capacitance (entered by user): HOFFSET = 0
QGAIN	Gain for quasistatic capacitance (entered by user): QGAIN = 1
QOFFSET	Offset for quasistatic capacitance (entered by user): QOFFSET = 0
CQADJ	Adjusted quasistatic capacitance by using QGAIN and QOFFSET: CQADJ = QGAIN*CQS+QOFFSET
HGAIN	Gain for calculated high frequency capacitance that is calculated: HGAIN = AT(MAVG(CQS,5)/MAVG(CC,5),MAXPOS(MAVG(CC,5)))
CHADJ	Adjusted high frequency capacitance by using HGAIN and HOFFSET: CHADJ = HGAIN*CC+HOFFSET
COX	Oxide capacitance: COX = MAX(MAVG(CHADJ,5))+1E-15
CMIN	Minimum capacitance from high frequency: CMIN = MIN(MAVG(CHADJ,5))+1E-15
TOXNM	Calculated thickness of oxide (in nanometers): TOXNM = 1E7*AREA*EOX/COX
INVCSQR	Inversed square of high frequency capacitance: INVCSQR = 1/(MAVG(CHADJ,5))^2
STRETCHOUT	Stretch out factor due to interfacial states: STRETCHOUT = MAVG((1-CQADJ/COX)/(1-CHADJ/COX),5)
NDOPING	Doping density: NDOPING = ABS(-2*STRETCHOUT/(AREA^2*Q*ES)/(DELTA(INVCSQR)/DELTA(VGS)))
DEPTHM	Depletion depth (in meters): DEPTHM = 1E-2*AREA*ES*(1/CHADJ-1/COX)
N90W	Doping density at 90% of maximum depletion depth: N90W = AT(NDOPING,FINDLIN(DEPTHM,0.9*MAX(DEPTHM),2))
DEBYEM	Debye length (in meters): DEBYEM = SQRT(ES*K*TEMP/(ABS(N90W)*Q^2))*1E-2
CFB	Flatband capacitance: CFB = (COX*ES*AREA/(DEBYEM*1E2))/(COX+(ES*AREA/(DEBYEM*1E2)))
VFB	Flatband voltage: VFB = AT(VGS,FINDLIN(CHADJ,CFB,2))
PHIB	Bulk potential: PHIB = (-1)*K*TEMP/Q*LN(ABS(N90W)/NI)*DOPETYPE
VTH	Threshold voltage: VTH = VFB+DOPETYPE*(AREA/COX*SQRT(4*ES*Q*ABS(N90W*PHIB))+ 2*ABS(PHIB))

Formula name	Description and formula
WMS	Work function difference between metal and semiconductor: WMS = WM-(WS+(EBG/2)-PHIB)
QEFF	Effective charge in oxide: QEFF = COX*(WMS-VFB)/AREA
BEST_LO	Index from DEPTHM array that is three Debye lengths from the surface: BEST_LO = FINDD(DEPTHM,3*DEBYEM,2)
BEST_HI	Index from DEPTHM array that is 95% of maximum depletion length, or twice the screening length in the semiconductor, whichever is larger: BEST_HI = FINDD(DEPTHM,COND(2*DEBYEM*SQRT(LN(ABS(N90W/NI))), MAX(DEPTHM),2*DEBYEM*SQRT(LN(ABS(N90W/NI))),0.95*MAX(DEPTHM)), 2)
NAVG	Average doping calculated between index BEST_HI and BEST_LO: NAVG = AVG(SUBARRAY(NDOPING,COND(BEST_HI, BEST_LO, BEST_HI, BEST_LO),COND(BEST_HI, BEST_LO,BEST_LO,BEST_HI)))
DIT	Interfacial states density: DIT = 1/(AREA*Q)*(1/(1/CQADJ-1/COX)-1/(1/CHADJ-1/COX))
PSISPSIO	PSIS - PSIO, which is surface potential: PSISPSIO = SUMMV((1-CQADJ/COX)*DELTA(VGS))*DOPETYPE
PSIO	Offset in surface potential due to calculation method and flatband voltage: PSIO = AT(PSISPSIO,FINDLIN(VGS,VFB,2))
PSIS	Silicon surface potential. More precisely, this value represents band bending and is related to surface potential via the bulk potential: PSIS = PSISPSIO-PSIO
EIT	Interface trap energy with respect to mid band gap: EIT = PSIS+PHIB

Formulas for ctsweep test (lifetime project)

Formula name	Description and formula
NAVG	Average doping: NAVG = 1E15
COX	Oxide capacitance (in picofarads): COX = 450
WF	Equilibrium inversion depth (in centimeters): WF = ES*AREA*(1/MAX(CHF)-1E12/COX)
WWF	W - WF, where W is the depletion depth (in centimeters): WWF = ES*AREA*(1/CHF-1E12/COX)-WF
GNI	Generation rate in S^{-1} divided by intrinsic carrier concentration: GNI = -(ES*AREA*NAVG*COX/1E12)*DIFF(1/CHF^2,TIME)/NI

Formulas for cvsweep test (stvs project)

Formula name	Description and formula
VGS	Gate voltage: VGS = -VSub
RS	Serial resistance calculated by high frequency CV: RS = AT(MAVG(G_OR_R,5)/(WF*MAVG(CHF,5)),MAXPOS(MAVG(CHF,5))))^2/((1+(AT(MAVG(G_OR_R,5)/(WF*MAVG(CHF,5)),MAXPOS(MAVG(CHF,5))))^2)*(AT(MAVG(G_OR_R,5),MAXPOS(MAVG(CHF,5)))))
AR	Intermediate parameter for calculation of CC: AR = G_OR_R-(G_OR_R^2+(WF*CHF)^2)*RS
CC	Corrected high frequency capacitance by compensating serial resistance: CC = ((G_OR_R^2+(WF*CHF)^2))*CHF/(AR^2+(WF*CHF)^2)
HOFFSET	Offset for high frequency capacitance (entered by user): HOFFSET = 0
DELAY	595 delay time: DELAY = 0.15
HGAIN	Gain for calculated high frequency capacitance that is calculated: HGAIN = AT(MAVG(CQS,5)/MAVG(CC,5),MAXPOS(MAVG(CC,5)))
CHADJ	Adjusted high frequency capacitance by using HGAIN and HOFFSET: CHADJ = HGAIN*CC+ HOFFSET
VSTEP	595 step voltage: VSTEP = 0.02
LEAKSLP	Average slop of leakage current neglecting the contribution of mobile ion: LEAKSLP = LINEFITSLP(VGS, QT, 49, 200) 49 and 200 are indexes on QT array to fit the slope.
QGAIN	Gain for quasistatic capacitance (entered by user): QGAIN = 1
CQADJ	Adjusted quasistatic capacitance by using QGAIN and QOFFSET: CQADJ = QGAIN*CQS+QOFFSET
NM	Mobile ion density: NM = AVG((CQADJ- CHADJ)*ABS(DELTA(VGS))*(LASTPOS(DELTA(VGS))-FIRSTPOS(DELTA(VGS)))/Q/AREA

Choosing the right parameters

This section describes how to choose the correct parameters for:

- Simultaneous C-V measurement
- The delay time to ensure that the device remains in equilibrium in the inversion region during a sweep
- Controlling errors at the source

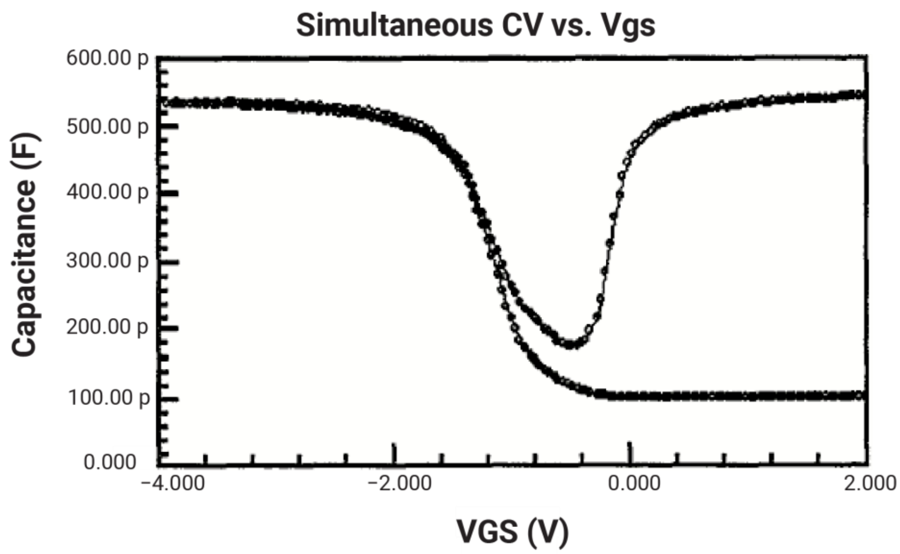
Optimal C-V measurement parameters

Simultaneous C-V measurement is a complicated matter. Besides system considerations, you should carefully choose the measurement parameters. Refer to the following discussion for considerations when selecting these parameters.

Start, stop, and step voltages

Most C-V data is derived from the sweep transition, or depletion region of the C-V curve. For that reason, start and stop voltages should be chosen so that the depletion region makes up about 1/3 to 2/3 of the voltage range.

Figure 94: Typical simultaneous C-V curve



The upper flat, or accumulation region of the high frequency C-V curve defines the oxide capacitance, COX. Since most analysis relies on the ratio C/COX, it is important that you choose a start or stop voltage (depending on the sweep direction) to bias the device into strong accumulation at the start or the end of the sweep.

You should carefully consider the size of the step voltage. Start, stop, and step size determine the total number of points in the sweep. Some compromise is necessary between having too few points in one situation, or too many points in the other.

For example, the complete doping profile is derived from data taken in the depletion region of the curve by using a derivative calculation. As the point spacing decreases, the vertical point spacing is increasingly caused by noise rather than changes in the signal. Consequently, choosing too many points in the sweep results in increased noise rather than an increased resolution in C-V measurement. It also takes more time to perform a C-V sweep.

Many calculations depend on good measurements in the depletion region, and too few points in this region give poor results. A good compromise results from choosing parameters that yield a capacitance change per step of approximately ten times the error in the signal.

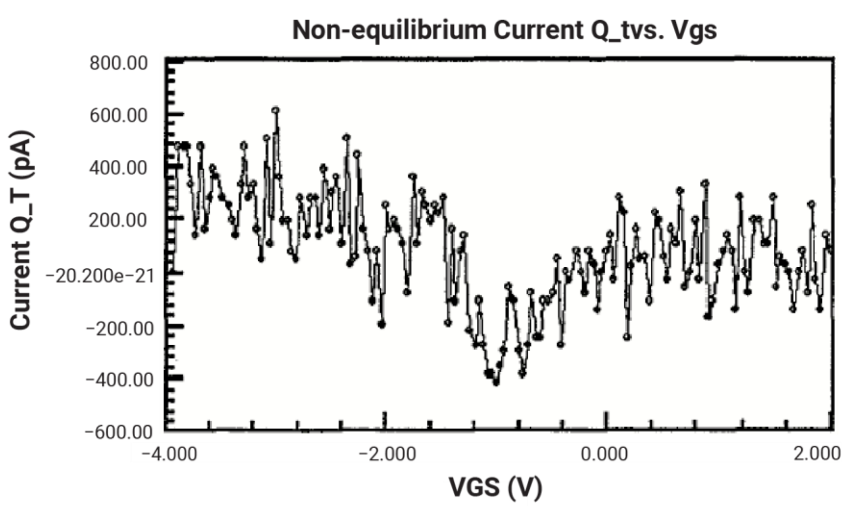
Sweep direction

For C-V sweeps, you can sweep either from accumulation to inversion, or from inversion to accumulation. Sweeping from accumulation to inversion will allow you to achieve deep depletion, profiling deeper into the semiconductor than you otherwise would obtain by maintaining equilibrium. When sweeping from inversion to accumulation, you should use a light pulse to achieve equilibrium more rapidly before the sweep begins.

Delay time

For accurate measurement, delay time must be carefully chosen to ensure that the device remains in equilibrium in the inversion region during a sweep. With a sweep that is too fast, the device remains in nonequilibrium, affecting Q/t, as shown in the following figure, and also results in skewed C-V curves.

Figure 95: Leakage current Q/t through device



Determining the optimal delay time

For accurate interface trap density measurement, delay time must be carefully chosen to ensure that the device remains in equilibrium in the inversion region during a sweep. An equilibrium test is provided to determine the optimum delay time.

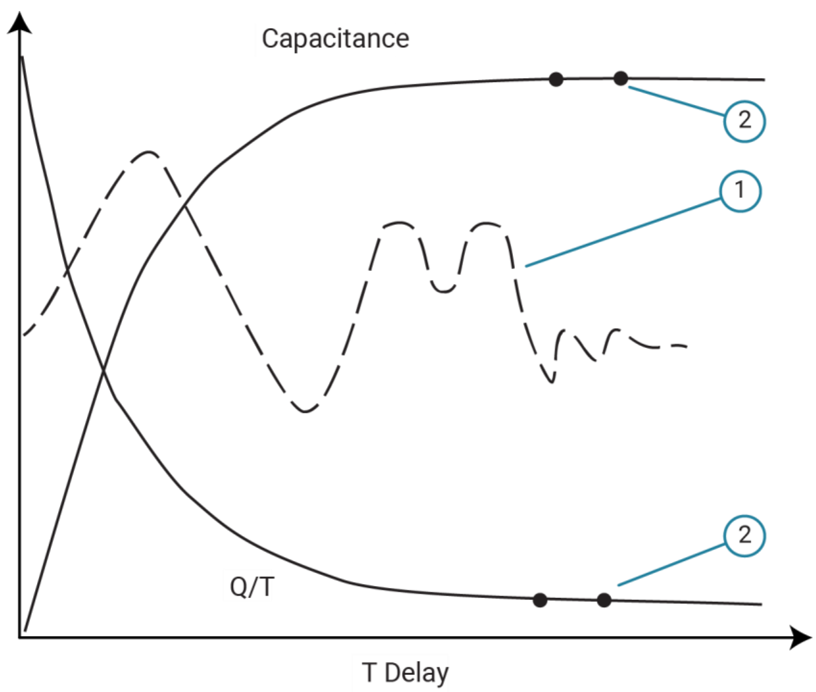
The equilibrium test uses the Model 595 to perform a series of quasistatic capacitance and Q/t current measurements using different delay times. The following figure shows the typical capacitance and Q/t curves generated for this test. As shown, the optimal delay is the second TDelay point after both curves have flattened out.

For long delay times, the measurement process can become very long with some devices. You may be tempted to speed up the test by using a shorter delay time. However, doing so is not recommended because it is difficult to quantify the amount of accuracy degradation in any given situation.

NOTE

See QTsweep (equilibrium test) (on page 6-14) for details on the equilibrium test.

Figure 96: Choosing optimal delay time



1	Erroneous curve because maximum delay time is too short
2	Choose second point for optimum delay time

Determining delay time with leaky devices

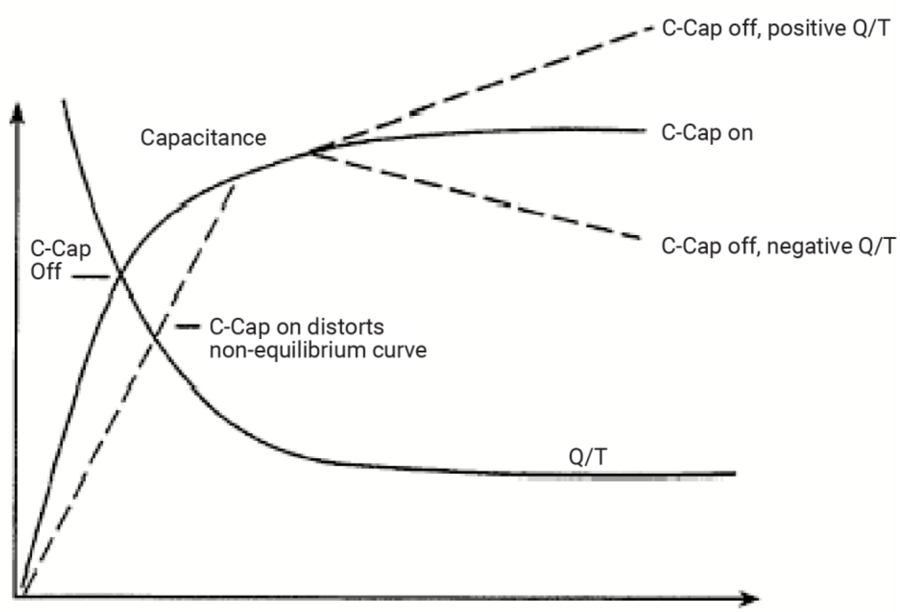
When testing for delay time on devices with relatively large leakage currents, it is recommended that you use the corrected capacitance feature, which is designed to compensate for leakage current. The reason for doing so is illustrated in the following figure. When large leakage currents are present, the capacitance curve does not flatten out in equilibrium, but instead either continues to rise (positive Q/t) or begins to decay (negative Q/t).

Using corrected capacitance results in the normal flat capacitance curve in equilibrium due to leakage compensation. Note, however, that the curve taken with corrected capacitance will be distorted in the nonequilibrium region, so data in that region should be considered to be invalid when using corrected capacitance. If it is necessary to use corrected capacitance when determining delay time, it is recommended that you make all measurements on that particular device using corrected capacitance.

NOTE

Corrected capacitance can be enabled for simultaneous C-V measurements by setting the `LeakageCorrection` parameter to `1` (see line 12 of the SIMCVsweep82 user module (on page 6-46)).

Figure 97: Capacitance and leakage current curves of leaky device



Testing slow devices

A decaying noise curve, such as the dotted line shown in the figure in Determining the optimal delay time (on page 6-28), will result if the maximum delay time is too short for the device being tested. This phenomenon, which is most prevalent with slow devices, occurs because the signal range is too small. To eliminate such erroneous curves, choose a longer maximum delay time. A good starting point for unknown devices is a 30-second maximum delay time.

Correcting residual errors

Controlling errors at the source is the best way to optimize C-V measurements, but doing so is not always possible. Remaining residual errors include offset, gain, noise, and voltage-dependent errors. Methods of correcting these error sources are discussed in the following paragraphs.

Offsets

Offset capacitance and conductance caused by the test apparatus can be eliminated by performing a suppression with the probes in the up position. These offsets will then be nulled out when the measurement is made. Whenever the system configuration is changed, the suppression procedure should be repeated. For maximum accuracy, it is recommended that you perform a probes-up suppression or at least verify before every measurement.

NOTE

Suppression can be enabled for simultaneous C-V measurements by setting the `OffsetCorrect` parameter to `1` (see line 14 of the SIMCVsweep82 user module (on page 6-46)).

Gain and nonlinearity errors

Gain errors are difficult to quantify. For that reason, gain correction is applied to every measurement. Gain constants are determined by measuring accurate calibration sources during the cable correction process.

Nonlinearity is normally more difficult to correct for than are gain or offset errors. The cable correction provides nonlinearity compensation for high-frequency measurements, even for non-ideal configurations such as switching matrices.

Voltage-dependent offset

Voltage-dependent offset (curve tilt) is the most difficult to correct error associated with quasistatic C-V measurements. It can be eliminated by enabling corrected capacitance (LeakageCorrection parameter set to 1). In this technique, the current flowing in the device is measured as the capacitance value is measured. The current is known as Q/t because its value is derived from the slope of the charge integrator waveform. Q/t is used to correct capacitance readings for offsets caused by shunt resistance and leakage currents.

Care must be taken when using the corrected capacitance feature, however. When the device is in nonequilibrium, device current adds to any leakage current, with the result that the curve is distorted in the nonequilibrium region. The solution is to keep the device in equilibrium throughout the sweep by carefully choosing the delay time.

Noise

You can minimize residual noise on the C-V curve by using filtering when taking your data. The Filter parameter sets the filter (see line 10 of SIMCVsweep82 user module (on page 6-46)). However, the filter reduces the sharpness of the curvature in the transition region of the quasistatic curve depending on the number of points in the region. This change in the curve can cause CQ to dip below CH, resulting in erroneous DIT calculations. If this situation occurs, turn off the filter or add more points.

ki82ulib user library

The user modules in the ki82ulib user library control the Model 82 C-V System. They perform simultaneous C-V, C-t, and Q/t measurements and cable compensation. The following table lists the user modules. It also provides the name of tests and actions in Clarius that are based on these user modules.

ki82ulib user modules

User module	Test and action names	Description
`Abortmodule82`	n/a	Puts the three System 82 instruments into a known state when a test is aborted. This function is used by other library modules in the `atexit()` function.
`CableCompensate82`	`cable-compensate` `cablecomp`	Performs 590 cable compensation using the capacitor values stored in the specified cable compensation file. The resultant compensation values generated by the compensation process are stored in the same file.
`CTsweep82`	`ctsweep`	Measures capacitance as a function of time at a certain bias.
`DisplayCableCompCaps82`	`display-cap-file`	Places capacitance source values in a spreadsheet.

ki82ulib user modules

User module	Test and action names	Description
LoadCableCorrectionConstants82	n/a	Read the cable compensation parameters and sends them to the 590. This module is for internal use by the SIMCVsweep82 and CTsweep82 modules. It is not normally used as a stand-alone module.
QTsweep82	qtsweep	Performs a quasistatic measurement sweep.
SaveCableCompCaps82	save-cap-file savecablecompfile	Saves entered capacitance source values in a file.
SIMCVsweep82	system82-cvsweep cvsweep	Performs simultaneous C-V sweep.

Abortmodule82

The `Abortmodule82()` function puts the three System 82 instruments into a known state when a test is aborted. This function is used by other library modules in the `atexit()` function.

Usage

```
atexit(Abortcleanup);
```

CableCompensate82 user module

The `CableCompensate82` routine performs 590 cable compensation using the capacitor values stored in the specified cable compensation file. The resultant compensation values generated by the compensation process are stored in the same file.

Usage

```
status = CableCompensate82(char *CabCompFile, char *InstIdStr, int InputPin, int
   OutPin, int Freq100 k, int Freq1M, int Range2p, int Range20 p, int Range200 p,
   int range2n);
```

status	Returned values; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
InstIdStr	KCon instrument ID; default is `CMTR1`; can be `CMTR1` to `CMTR4`, depending on your system configuration
InputPin	The DUT pin to which the 5951 input terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connections are made; see **Details**
OutPin	The DUT pin to which the 5951 output terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connections are made; see **Details**
Freq100 k	Use compensation for the 100 kHz frequency: ▪ Skip compensation for this frequency: `0` ▪ Do compensation for this frequency: `1`
Freq1M	Use compensation for the 1 MHz frequency: ▪ Skip compensation for this frequency: `0` ▪ Do compensation for this frequency: `1`

Range2p	Use compensation for the 2 pF range: ■ Skip compensation for this range: `0` ■ Do compensation for this range: `1`
Range20 p	Use compensation for the 20 pF range: ■ Skip compensation for this range: `0` ■ Do compensation for this range: `1`
Range200 p	Use compensation for the 200 pF range: ■ Skip compensation for this range: `0` ■ Do compensation for this range: `1`
range2n	Use compensation for the 2 nF range: ■ Skip compensation for this range: `0` ■ Do compensation for this range: `1`

Details

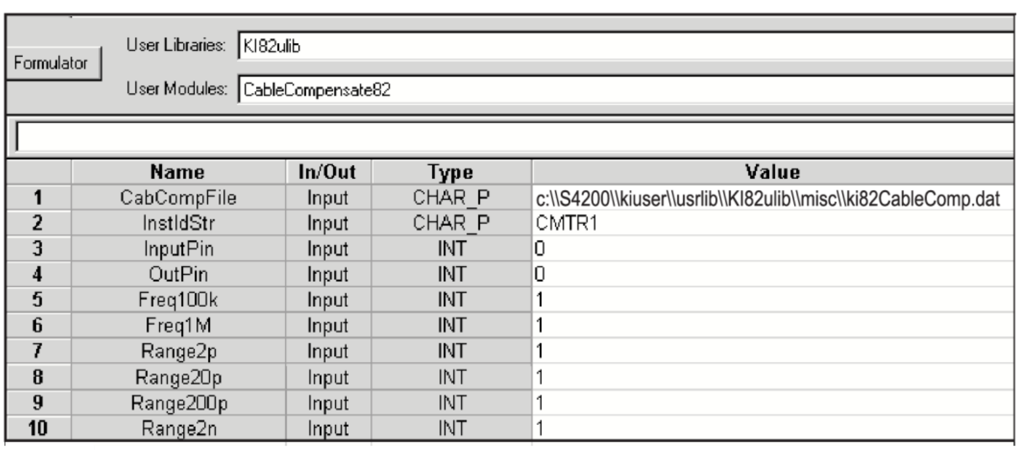
This user module, shown in the following figure, does cable compensation for the selected ranges and test frequencies of the 590. For the input parameters shown in the figure, cable compensation for the 590 is done for the 2 pF, 20 pF, 200 pF, and 2 nF ranges and for both the 100 kHz and 1 MHz test frequencies. The line 1 input parameter indicates the directory path where the user-input capacitor source values are saved. These values are entered and saved using the SaveCableCompCaps82 user module (on page 6-43).

User-entered parameters and returned outputs for this user module are explained in the user module description.

NOTE

For details on the procedure to perform cable compensation, see Cable compensation tests (on page 6-10).

Figure 98: CableCompensate82 user module



Formulator

User Libraries: KI82ulib

User Modules: CableCompensate82

	Name	In/Out	Type	Value
1	CabCompFile	Input	CHAR_P	c:\\S4200\\kiuser\\usrlib\\KI82ulib\\misc\\ki82CableComp.dat
2	InstIdStr	Input	CHAR_P	CMTR1
3	InputPin	Input	INT	0
4	OutPin	Input	INT	0
5	Freq100k	Input	INT	1
6	Freq1M	Input	INT	1
7	Range2p	Input	INT	1
8	Range20p	Input	INT	1
9	Range200p	Input	INT	1
10	Range2n	Input	INT	1

The return values from *status* can be:

- 0: OK.
- -10000 (INVAL_INST_ID): The specified instrument ID does not exist
- -10021 (COMP_FILE_NOT_EXIST): The specified compensation file does not exist
- -10022 (KI590_NOT_IN_KCON): There is no CMTR defined in your system configuration
- -10090 (GPIB_ERROR_OCCURRED): A GPIB communications error occurred
- -10100 (INVAL_PARAM): An invalid input parameter is specified

If *CabCompFile* does not exist, or if there is no path specified (null string), the default compensation parameters are used. When entering the path, be sure to use two \ characters to separate each directory. For example, if your cable file is in:

```
C:\calfiles\82cal.dat
```

You would enter:

```
C:\\calfiles\\82cal.dat
```

If you are controlling a switching matrix to route signals using a connection UTM (for example, “connect”), you do not need connect *InputPin* and *OutputPin*. Set these parameters to 0.

Procedure

For each range and test frequency specified by the input parameters:

1. You are prompted to open the circuit so that an offset capacitance measurement can be made.
2. Once the offset capacitance measurement is completed, you are prompted to connect the low value capacitor for the selected range. The system performs the low capacitor compensation.
3. You are prompted to connect the high value capacitor for the selected range. The system does the high value capacitor compensation.
4. You are prompted to reconnect the low capacitor.
5. The nominal and measured values are displayed in a dialog.
6. Verify the values. If you are unsatisfied with the measurement, select Cancel to abort the procedure. If you select Cancel, the cable compensation file is not affected.
7. When all selected ranges and frequencies have been compensated successfully, the cable compensation values are saved.

Also see

None

CtSweep82 user module

The CtSweep82 user module measures capacitance as a function of time at a certain bias.

Usage

```
status = CtSweep82(int Frequency, double Default_Bias, double Stress_Time, double
    Test_Bias, double Sample_Time, int Reading_rate, int Num_Points, int Range590,
    int Model590, int Filter590, char *CabComFile, int OffsetCorrect, char
    *instr_id, int InputPin, int OutPin, double *CHF, int CHF_ArrSize, double
    *G_or_R, int G_or_R_ArrSize, double *Time, int Time_ArrSize);
```

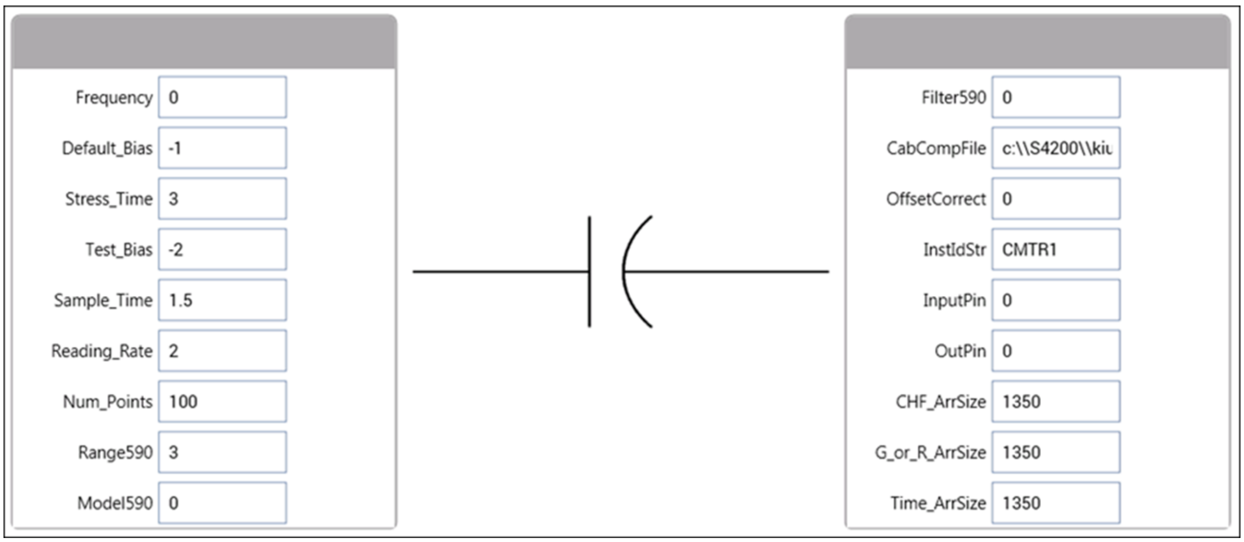
status	Returned values are placed in the Analyze sheet; see **Details**
Frequency	The measurement frequency: ■ 100 kHz: `0` ■ 1 MHz: `1`
Default_Bias	DC bias applied before and after a C-t sweep (–20 V to +20 V)
Stress_Time	Duration of the default bias before test bias is applied (0.001 s to 65 s)
Test_Bias	Voltage bias for capacitance measurements (–20 V to +20 V)
Sample_Time	Time delay between each sampling measurement (0.001 s to 65 s)
Reading_rate	The reading rate used to acquire the measurements (1 to 4; see **Details**)
Num_Points	Number of sampling points (1 to 1350)
Range590	The measurement range for the 590 (1 to 4; see **Details** for valid range values)
Model590	The measurement model to use for high frequency measurement: ■ Parallel mode: `0` ■ Series model: `1`
Filter590	Enable or disable the analog filter; see **Details**: ■ Disable the filter: `0` ■ Enable the filter: `1`
CabCompFile	The complete name and path for the cable compensation file; see **Details**
OffsetCorrect	Enable or disable an offset correction measurement: ■ Disable offset correction: `0` ■ Enable offset correction: `1`
instr_id	KCon instrument ID; default is `CMTR1`; can be `CMTR1` to `CMTR4`, depending on your system configuration
InputPin	The DUT pin to which the 5951 input terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connections are made; see **Details**
OutPin	The DUT pin to which the 5951 output terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connections are made; see **Details**
CHF	Output; the measured array of high frequency capacitance values
CHF_ArrSize	Set to 1350
G_or_R	Output; the array of measured conductance (G) or resistance (R) values
G_or_R_ArrSize	Set to 1350
Time	Output; the array of time from the 595 output for each measurement step
Time_ArrSize	Set to 1350

Details

This method can be used for minority carrier lifetime measurements using Zerbst plot.

The following figure shows the default parameters for the `ctsweep` UTM, which uses the `CtSweep82` user module. In this example, the Model 82 is set to first stress the DUT at +3 V for three seconds, and then perform 100 capacitance measurements at –3 V using a 0.1 s time interval (see CtSweep test description (on page 6-20)). For details on C-t measurements, refer to C-t sweep (on page 6-19).

Figure 99: CtSweep82 user module



The analog filter, enabled with *`Filter590`*, can minimize the amount of noise that appears in the readings. It does, however, increase the measurement time.

***`Reading_rate`* valid inputs**

Reading rate	Nominal reading rate (per second)	Readings	Display resolution (digits)
1	75	C,G,V	3.5
2	18	C,G,V	4.5
3	10	C,G,V	4.5
4	1	C,G,V	4.5

`Range590` valid range values

Range	100 kHz	1 MHz
1	2 pF / 2 µs	20 pF / 200 µs
2	20 pF / 20 µs	20 pF / 200 µs
3	200 pF / 200 µs	200 pF / 2 ms
4	2 nF / 2 ms	2 nF / 20 ms

If `CabCompFile` does not exist, or if there is no path specified (null string), the default compensation parameters are used. When entering the path, be sure to use two \ characters to separate each directory. For example, if your cable file is in:

```
C:\calfiles\82cal.dat
```

You would enter:

```
C:\\calfiles\\82cal.dat
```

If you are controlling a switching matrix to route signals using a connection UTM (for example, “connect”), you do not need connect *InputPin* and *OutputPin*. Set these parameters to 0.

The return values from *status* can be:

- 0: OK.
- -10000 (INVAL_INST_ID): The specified instrument ID does not exist(INVAL_INST_ID): The specified instrument ID does not exist
- -10020 (COMP_FILE_ACCESS_ERR): There was an error accessing the specified cable compensation file
- -10021 (COMP_FILE_NOT_EXIST): The specified compensation file does not exist
- -10045 (KI82_NOT_IN_KCON): There is no CMTR defined in your system configuration
- -10023 (KI590_MEAS_ERROR): A measurement error occurred
- -10101 (ARRAY_SIZE_TOO_SMALL): The specified value for *CHF_ArrSize*, *G_or_R_ArrSize*, or *Time_ArrSize* was too small for the number of steps in the sweep
- -10102 (ERROR_PARSING): There was an error parsing the response of the 590.
- -10104 (USER_CANCEL): The user canceled the correction procedure.
- -10090 (GPIB_ERROR_OCCURRED): A GPIB communications error occurred
- -10100 (INVAL_PARAM): An invalid input parameter is specified

Procedure

1. If set, you are prompted to open the circuit so that an offset capacitance measurement can be made.
2. If a cable compensation file is specified, the compensation information in that file for the selected range and frequency will be loaded. If not, instrument default compensation is used.
3. A C-t sweep is performed.

Also see

None

DisplayCableCompCaps82 user module

This user module is used for Model 82 cable compensation. When this test is run, the nominal capacitance source values saved by the `SaveCableCompCaps82` user module are placed into a spreadsheet for viewing.

Usage

```
status = DisplayCableCompCaps82(char *CabCompFile, double *Range, int RangeSize,
   double *Values100 k, int Values100 kSize, double *Values1M, int Values1MSize);
```

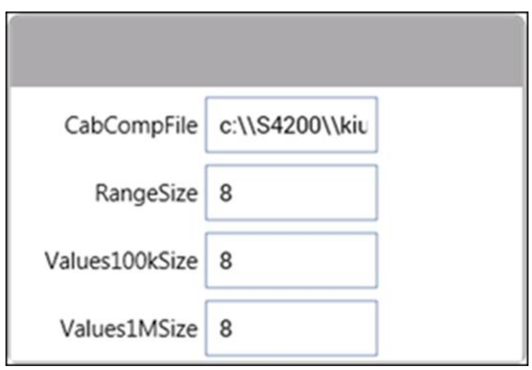
status	Returned values are placed in the Analyze sheet; see **Details**
CabCompFile	The complete name and path for the cable compensation file; see **Details**
Range	Output; an 8-element array that receives the nominal range values
RangeSize	The size of the *Range* array; set to 8
Values100k	Output; an 8-element (fixed) array that receives the nominal capacitor values used for the cable compensation at the 100 kHz frequency
Values100kSize	The size of the *Values100k* array; set to 8
Values1M	Output; an 8-element (fixed) array that receives the nominal capacitor values used for the cable compensation at the 1 MHz frequency
Values1MSize	The size of the *Values1M* array; set to 8

Details

The `DisplayCableCompCaps82` user module reads the nominal cable compensation values that are stored in the compensation file and returns them to the calling function. In the case of Clarius, it returns the values to the UTM data sheet.

The default parameters for this user module are shown in the following figure. Line 1 specifies the file directory path where the capacitance values are saved. This file directory path must be the same as the one used by the `SaveCableCompCaps82` user module.

Figure 100: DisplayCableCompCaps82 user module



To prevent unpredictable results, the array size values for the `RangeSize`, `Values100kSize`, and `Values1MSize` arrays must be set to 8.

NOTE

For details on the procedure to perform cable compensation, refer to Cable compensation tests (on page 6-10).

The returned arrays are arranged in the order shown in the following table.

Reading_rate **valid inputs**		
Range	**100 kHz values**	**1 MHz values**
2E-12	2 pF low comp value	2 pF low comp value
2E-12	2 pF high comp value	2 pF high comp value
20E-12	20 pF low comp value	20 pF low comp value
20E-12	20 pF high comp value	20 pF high comp value
200E-12	200 pF low comp value	200 pF low comp value
200E-12	200 pF high comp value	200 pF high comp value
2E-9	2 nF low comp value	2 nF low comp value
2E-9	2 nF high comp value	2 nF high comp value

If *CabCompFile* does not exist, or if there is no path specified (null string), the default compensation parameters are used. When entering the path, be sure to use two \ characters to separate each directory. For example, if your cable file is in:

`C:\calfiles\82cal.dat`

You would enter:

```
C:\\calfiles\\82cal.dat
```

The return values from *status* can be:

- 0: OK.
- -10021 (COMP_FILE_NOT_EXIST): The specified compensation file does not exist
- -10022 (KI590_NOT_IN_KCON): There is no CMTR defined in your system configuration
- -10090 (GPIB_ERROR_OCCURRED): A GPIB communications error occurred
- -10100 (INVAL_PARAM): An invalid input parameter is specified

Also see

SaveCableCompCaps82 user module (on page 6-43)

QTsweep82 user module

This user module uses the 595 to determine the equilibrium point for a device by measuring quasistatic capacitance using different delay times.

Usage

```
status = QTsweep82(double Test_Bias, int LeakageCorrection, double Hold_time,
   double V-Step, char *InstldStr, int InputPin, int OutPin, double Delay_Max, int
   Range, double *CQS, int, CQS_ArrSize, double *QT, int QT_ArrSize, double
   *Delay_time, int Delay_time_ArrSize);
```

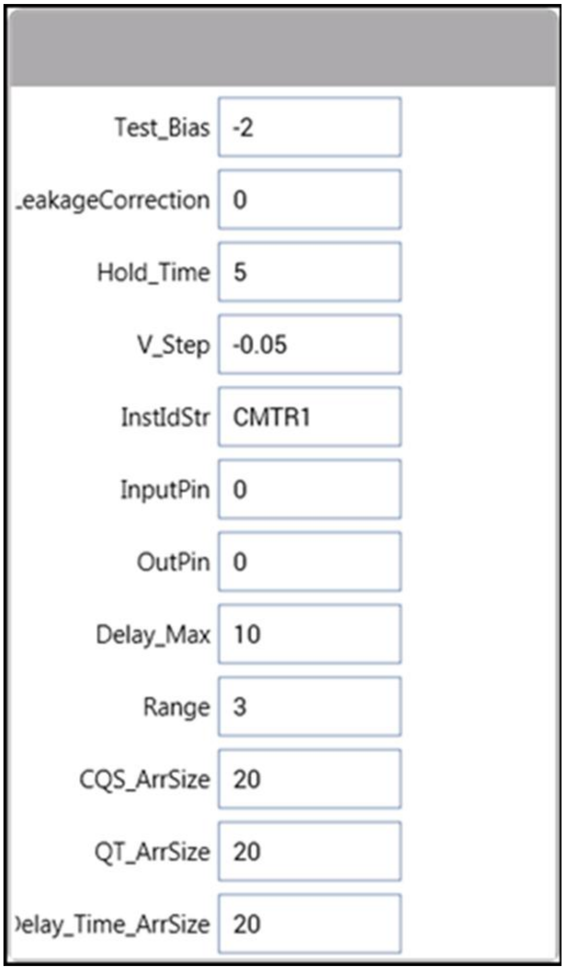
status	Returned values are placed in the Analyze sheet; see **Details**
Test_Bias	Voltage bias for capacitance measurements (–120 V to +120 V)
LeakageCorrection	Disable: `0` Enable: `1`
Hold_Time	Hold time at the beginning of the sweep (0 s to 200 s; default 5)
V-Step	Step voltage size: ±0 V, ±0.01 V, ±0.02 V, ±0.05 V, ±0.1 V
InstldStr	KCon instrument ID; default is `CMTR1`; can be `CMTR1` to `CMTR4`, depending on your system configuration
InputPin	The DUT pin to which the 5951 input terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connections are made
OutPin	The DUT pin to which the 5951 output terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connections are made
Delay_Max	Maximum delay time: 1 s to 199.99 s (default 10 s)
Range	The measurement range for the 595 to use: 1 to 3; see **Details**
CQS	Output; the measured array of quasistatic capacitance values
CQS_ArrSize	Set to 20
QT	Output; the measured array of leakage current Q/T
QT_ArrSize	Set to 20
Delay_time	Output; the array of *Delay_Time* used up to *Delay_Max* in logarithm scale
Delay_time_ArrSize	Set to 20

Details

The module measures quasistatic capacitance and leakage current as a function of delay time using the 595. It is used to determine the equilibrium condition. Each quasistatic capacitance reading is calculated from charge measurements performed on every two steps of a voltage sweep. Leakage current at the end of each reading sample is also calculated ($i = \Delta Q/\Delta t$).

The following figure shows the default parameters for the `QTsweep82` user module.

Figure 101: QTsweep82 user module



The Q/T sweep in Equilibrium test (QTsweep) description (on page 6-15) acquires 20 quasistatic capacitance readings. After the graph for quasistatic capacitance and leakage current versus time is plotted, the optimum delay time for equilibrium can be determined.

NOTE

For details on quasistatic measurements, see QTsweep (on page 6-14). For details on C-t measurements, see C-t sweep (on page 6-19).

The Range values are shown in the following table.

Range values	
Value	**595 range**
1	200 pF
2	2 nF
3	20 nF

The return values from `status` can be:

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist
- `-10045` (`KI82_NOT_IN_KCON`): There is no CMTR defined in your system configuration
- `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred
- `-10091 (GPIB_TIMEOUT)`: A timeout occurred during communications
- `-10020 (COMP_FILE_ACCESS_ERR)`: There was an error accessing the specified cable compensation file
- `-10021 (COMP_FILE_NOT_EXIST)`: The specified compensation file does not exist
- `-10100 (INVAL_PARAM)`: An invalid input parameter is specified
- `-10101 (ARRAY_SIZE_TOO_SMALL)`: The specified value for *CQS_ArrSize*, *QT_ArrSize*, or *Delay_Time_ArrSize* was too small for the number of steps in the sweep
- `-10102 (ERROR_PARSING)`: There was an error parsing the response.
- `-10104 (USER_CANCEL)`: The user canceled the correction procedure.

Procedure

1. If set, you are prompted to open the circuit so that an offset capacitance measurement can be made.
2. If a cable compensation file is specified, the compensation information in that file for the selected range and frequency will be loaded. If not, instrument default compensation is used.
3. A Q/T sweep is performed.

Also see

None

SaveCableCompCaps82 user module

The user module saves the nominal values of the capacitors used with the 590 cable compensation procedure to a file.

Usage

```
status = SaveCableCompCaps82(char *CabCompFile, double Lo2p100k, double Lo2p1M,
   double Hi2p100k, double Hi2p1M, double Lo20p100k, double Lo20p1M, double
   Hi20p100k, double Hi20p1M, double Lo200p100k, double Lo200p1M, double
   Hi200p100k, double 200p1M, double Lo2n100k, double Lo2n1M, double Hi2n100k,
   double Lo2n1M);
```

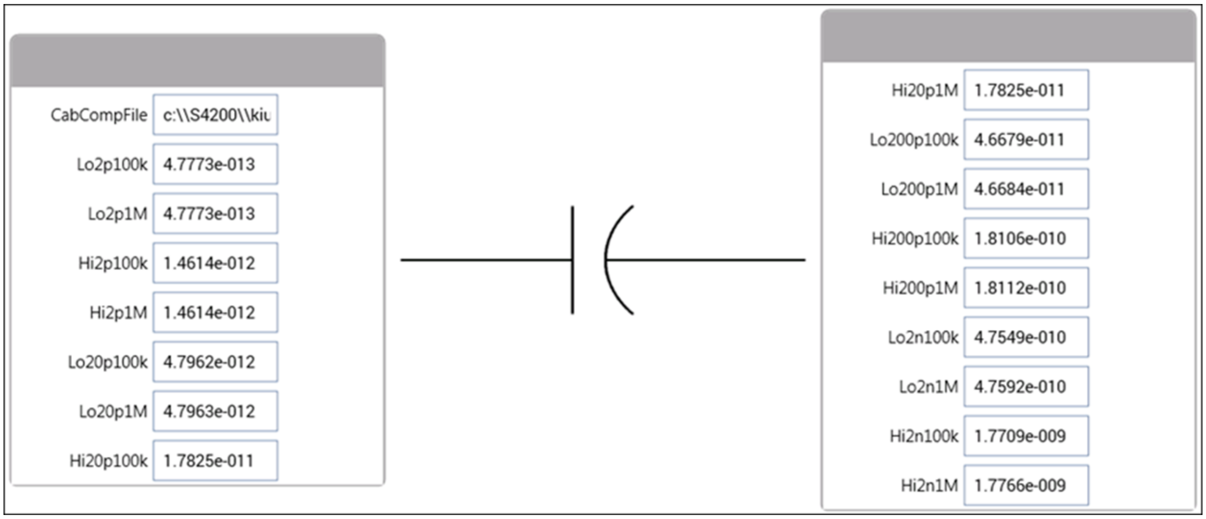
CabCompFile	The complete name and path for the cable compensation file; see **Details**
Lo2p100k	The nominal value of the low range capacitor used to perform cable compensation for the 2 pF range and 100 kHz frequency: 0 F to 0.95E−12 F
Lo2p1M	The nominal value of the low range capacitor used to perform cable compensation for the 2 pF range and 1 MHz frequency: 0 F to 0.95E−12 F
Hi2p100k	The nominal value of the high range capacitor used to perform cable compensation for the 2 pF range and 100 kHz frequency: 1E−12 F to 2E−12 F
Hi2p1M	The nominal value of the high range capacitor used to perform cable compensation for the 2 pF range and 1 MHz frequency: 1E−12 F to 2E−12 F
Lo20p100k	The nominal value of the low range capacitor used to perform cable compensation for the 20 pF range and 100 kHz frequency: 0 F to 9.5E−12 F
Lo20p1M	The nominal value of the low range capacitor used to perform cable compensation for the 20 pF range and 1 MHz frequency: 0 F to 9.5E−12 F
Hi20p100k	The nominal value of the high range capacitor used to perform cable compensation for the 20 pF range and 100 kHz frequency: 10E−12 F to 20E−12 F
Hi20p1M	The nominal value of the high range capacitor used to perform cable compensation for the 20 pF range and 1 MHz frequency: 10E−12 F to 20E−12 F
Lo200p100k	The nominal value of the low range capacitor used to perform cable compensation for the 200 pF range and 100 kHz frequency: 0 F to 95E−12 F
Lo200p1M	The nominal value of the low range capacitor used to perform cable compensation for the 200 pF range and 1 MHz frequency: 0 F to 95E−12 F
Hi200p100k	The nominal value of the high range capacitor used to perform cable compensation for the 200 pF range and 100 kHz frequency: 100E−12 F to 200E−12 F
Hi200p1M	The nominal value of the high range capacitor used to perform cable compensation for the 200 pF range and 1 MHz frequency: 100E−12 F to 200E−12 F
Lo2n100k	The nominal value of the low range capacitor used to perform cable compensation for the 2 nF range and 100 kHz frequency: 0 F to 995E−12 F
Lo2n1M	The nominal value of the low range capacitor used to perform cable compensation for the 2 nF range and 1 MHz frequency: 0 F to 995E−12 F
Hi2n100k	The nominal value of the high range capacitor used to perform cable compensation for the 2 nF range and 100 kHz frequency: 1000E−12 F to 2000E−12 F
Hi2n1M	The nominal value of the high range capacitor used to perform cable compensation for the 2 nF range and 1 MHz frequency: 1000E−12 F to 2000E−12 F

Details

This user module is used for 590 cable compensation. The user enters precise capacitance source values. When this test is run, the capacitance source values are saved to a user-specified file. If no cable compensation file exists, this module creates one. The user module to perform cable compensation (CableCompensate82) can then access the capacitance source values from this file. The user must have the proper system permissions in order for this user module to create a file.

The default parameter values for this user module are shown in the following figure. These are example low and high values that can be used for cable compensation. You must replace these values with the calibration values of the actual capacitance sources.

Figure 102: SaveCableCompCaps82 user module



NOTE

For details on the procedure to perform cable compensation, see Cable compensation tests (on page 6-10).

The return values from `status` can be:

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist
- `-10001 (INVAL_PIN_SPEC)`: An invalid DUT pin number was specified
- `-10003 (NO_SWITCH_MATRIX)`: No switching matrix was found
- `-10004 (NO_MATRIX_CARDS)`: No matrix cards were found
- `-10020 (COMP_FILE_ACCESS_ERR)`: There was an error accessing the specified cable compensation file
- `-10021 (COMP_FILE_NOT_EXIST)`: The specified compensation file does not exist
- `-10022 (KI590_NOT_IN_KCON)`: There is no CMTR defined in your system configuration
- `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred
- `-10100 (INVAL_PARAM)`: An invalid input parameter is specified

If *CabCompFile* does not exist, or if there is no path specified (null string), the default compensation parameters are used. When entering the path, be sure to use two `\` characters to separate each directory. For example, if your cable file is in:

```
C:\calfiles\590cal.dat
```

You would enter:

```
C:\\calfiles\\590cal.dat
```

If you are controlling a switching matrix to route signals using a connection UTM (for example, “connect”), you do not need connect *InputPin* and *OutputPin*. Set these parameters to `0`.

Also see

None

SIMCVsweep82 user module

The `SIMCVsweep82` routine performs a simultaneous capacitance versus voltage (C-V) sweep using the Keithley 82 C-V System.

Usage

```
status = SIMCVsweep82(double Frequency, double Default_Bias, double Stress_Time,
   double VSub_Start, double VSub_Stop, double VSub_Step, int Range595, int
   Range590, int Model590, int Filter, double Delay595, int LeakageCorrection, char
   *CabCompFile, int OffsetCorrect, char *InstldStr, int InputPin, int OutPin,
   double *CHF, int CHF_ArrSize, double *VSub, int VSub_ArrSize, double *CQS, int
   CQS_ArrSize, double G_or_R, int G_or_R_ArrSize, double *QT, int QT_ArrSize);
```

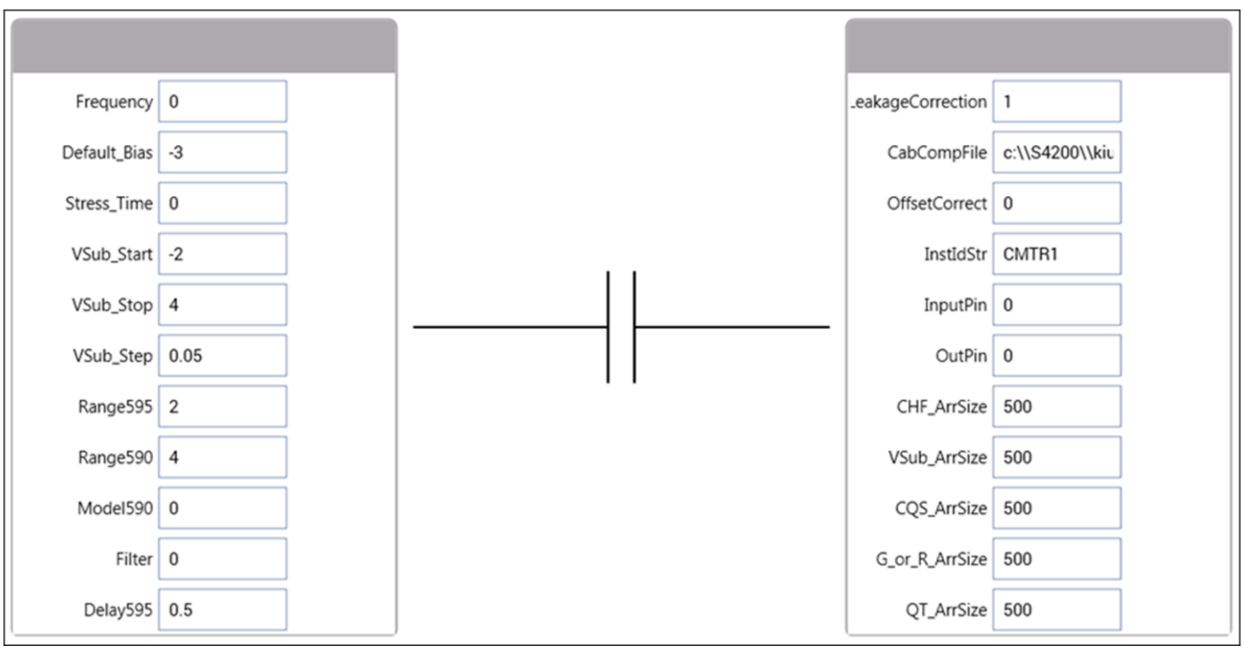
status	Returned values are placed in the Analyze sheet; see **Details**
Frequency	The measurement frequency to use for the 590: ■ 100 kHz: `0` ■ 1 MHz: `1`
Default_Bias	DC bias applied before and after a voltage sweep (–100 V to +100 V)
Stress_Time	Time for which default bias is stressed on the device before voltage sweep: 0 s to 999 s
VSub_Start	Start voltage on substrate: –120 V to +120 V
VSub_Stop	Stop voltage on substrate: –120 V to +120 V
VSub_Step	Voltage step size: ±0 V , ±0.01 V, ±0.02 V, ±0.05 V, or ±0.1 V
Range595	The measurement range for the 595 to use: ■ 200 pF: `1` ■ 2 nF: `2` ■ 20 nF: `3`
Range590	The measurement range for the 590 to use: `1` to `4`; refer to **Details**
Model590	The measurement model to use for high frequency measurement: ■ Parallel mode: `0` ■ Series model: `1`
Filter	Enable or disable the digital filter: ■ 1 reading: `0` ■ 3 readings: `1` ■ 9 readings: `2` ■ 24 readings: `3`
Delay595	Delay time for 595; maximum 199 s (default 0.07 s)
LeakageCorrection	Enable or disable the leakage current correction of the 595: ■ Disable: `0` ■ Enable: `1`
CabCompFile	The complete name and path for the cable compensation file; see **Details**
OffsetCorrect	Enable or disable an offset correction measurement: ■ Disable offset correction: `0` ■ Enable offset correction: `1`

InstldStr	CMTR 82 instrument ID; default is `CMTR1`; can be `CMTR1` to `CMTR4`, depending on your system configuration
InputPin	The DUT pin to which the 5951 input terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connections are made
OutPin	The DUT pin to which the 5951 output terminal is attached (–1 to 72); if a value of less than 1 is specified, no switching matrix connections are made
CHF	Output; the measured array of high frequency capacitance values
CHF_ArrSize	This must be set to a value equal to the number of voltage steps in the sweep or value = ((VSub_Stop – VSub_Start) / VSub_Step – 1)
VSub	Output; the array of bias voltages used
VSub_ArrSize	This must be set to a value equal to the number of voltage steps in the sweep or value = ((VSub_Stop – VSub_Start) / VSub_Step – 1)
CQS	Output; the measured array of quasistatic capacitance values
CQS_ArrSize	This must be set to a value equal to the number of voltage steps in the sweep or value = ((VSub_Stop – VSub_Start) / VSub_Step – 1)
G_or_R	Output; the array of measured conductance (G) or resistance (R) values
G_or_R_ArrSize	This must be set to a value equal to the number of voltage steps in the sweep or value = ((VSub_Stop – VSub_Start) / VSub_Step – 1)
QT	Output; the array of Q/T from 595 output for each measurement step
QT_ArrSize	This must be set to a value equal to the number of voltage steps in the sweep or value = ((VSub_Stop – VSub_Start) / VSub_Step – 1)

Details

This user module uses the 590 and 595 to perform simultaneous C-V measurements. The following figure shows the default parameters for the `SIMCVsweep82` user module.

Figure 103: SIMCVsweep82 user module



It performs a staircase sweep from –3 V to +3 V in 20 mV steps, as shown in cvsweep test description (on page 6-18).

You can make an offset correction measurement and use the cable compensation.

NOTE

For details on quasistatic measurements, see Simultaneous C-V sweep (on page 6-17).

The following table lists the valid range values for *Range590*.

Range590 valid range values

Range	100 kHz	1 MHz
1	2 pF / 2 µs	20 pF / 200 µs
2	20 pF / 20 µs	20 pF / 200 µs
3	200 pF / 200 µs	200 pF / 2 ms
4	2 nF / 2 ms	2 nF / 20 ms

If *CabCompFile* does not exist, or if there is no path specified (null string), the default compensation parameters are used. When entering the path, be sure to use two \ characters to separate each directory. For example, if your cable file is in:

```
C:\calfiles\590cal.dat
```

You would enter:

```
C:\\calfiles\\590cal.dat
```

The return values from *status* can be:

- `0`: OK.
- `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist
- `-10020 (COMP_FILE_ACCESS_ERR)`: There was an error accessing the specified cable compensation file
- `-10021 (COMP_FILE_NOT_EXIST)`: The specified compensation file does not exist
- `-10023 (KI590_MEAS_ERROR)`: A measurement error occurred
- `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred
- `-10091 (GPIB_TIMEOUT)`: A timeout occurred during communications
- `-10100 (INVAL_PARAM)`: An invalid input parameter is specified
- `-10101 (ARRAY_SIZE_TOO_SMALL)`: The specified value for *CHF_ArrSize*, *G_or_R_ArrSize*, *V_ArrSize*, *CQS_ArrSize*, or *QT_ArrSize* was too small for the number of steps in the sweep
- `-10102 (ERROR_PARSING)`: There was an error parsing the 590 response
- `-10104 (USER_CANCEL)`: The user canceled the correction procedure
- `-10045 (KI82_NOT_IN_KCON)`: KI82 is not in KCon

Procedure

1. If set, you are prompted to open the circuit so that an offset capacitance measurement can be made.
2. If a cable compensation file is specified, the compensation information in that file for the selected range and frequency will be loaded. If not, instrument default compensation is used.
3. A simultaneous C-V sweep is made.

Also see

None

Simultaneous C-V analysis

This section discusses the theory and techniques used in the various Keithley Simultaneous C-V libraries. For more detailed discussions, refer to the References and bibliography of C-V measurements (on page 6-70).

Analysis methods

The following figures show fundamental C-V curves for p-type and n-type materials. Both high-frequency and quasistatic curves are shown in these figures. Note that the high-frequency curves are highly asymmetrical, while the quasistatic curves are almost symmetrical. Accumulation, depletion, and inversion regions are also shown on the curves. The gate-biasing polarity and high-frequency curve shape can be used to determine device type, as shown in the following figures.

Figure 104: C-V characteristics of p-type material

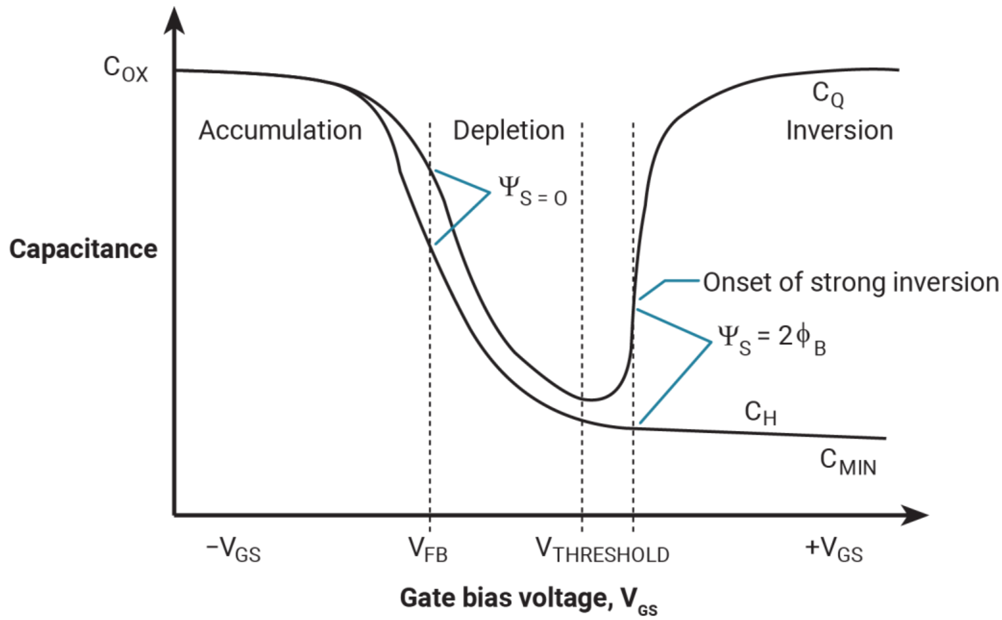
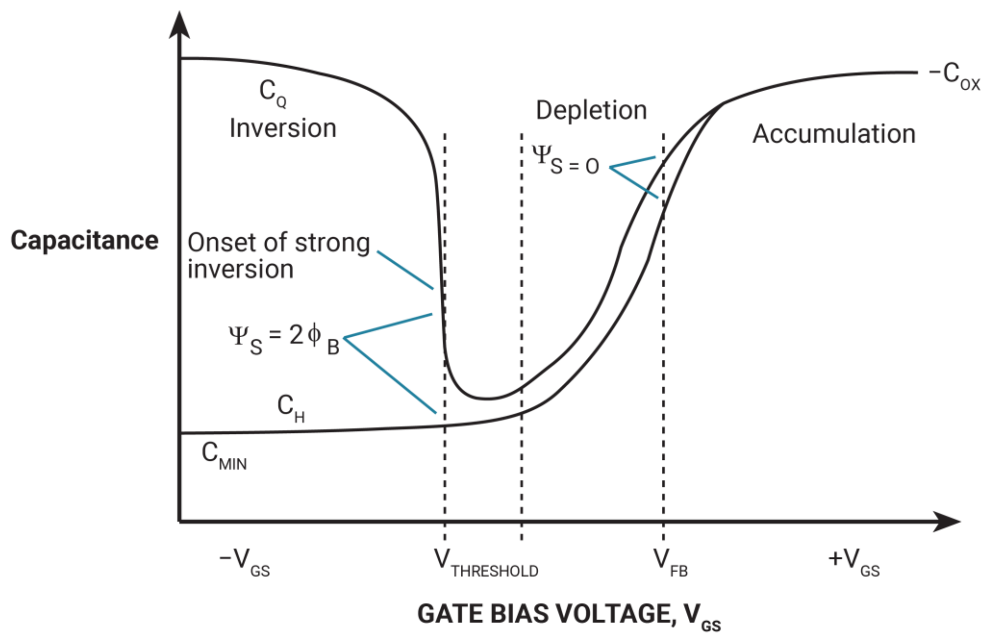


Figure 105: C-V characteristics of n-type material



Basic device parameters

The following topics provide additional detail on device parameters and how they are calculated.

Determining device type

The semiconductor conductivity type (p or n dopant ions) can be determined from the relative shape of the C-V curves (see Analysis methods (on page 6-49)). The high-frequency curve gives a better indication than the quasistatic curve because of its highly asymmetrical nature. Note that the C-V curve moves from the accumulation to the inversion region as gate voltage, V_{GS}, becomes more positive for p-type materials, but the curve moves from accumulation to inversion as V_{GS} becomes more negative with n-type materials (Nicollian and Brews 372-374).

- If C_H is greater when V_{GS} is negative than V_{GS} when positive, the substrate material is p-type.
- If C_H is greater with positive V_{GS} than negative V_{GS}, the substrate is n-type.
- The end of the curve where C_H is greater is the accumulation region, while the opposite end of the curve is the inversion.

Oxide capacitance, thickness, and gate area

The oxide capacitance, C_{OX}, is the high-frequency capacitance with the device biased in strong accumulation. Oxide thickness is calculated from C_{OX} and gate area as follows:

$$t_{ox} = \frac{A\,\varepsilon_{ox}}{(1 \times 10^{-19})C_{ox}}$$

Where:

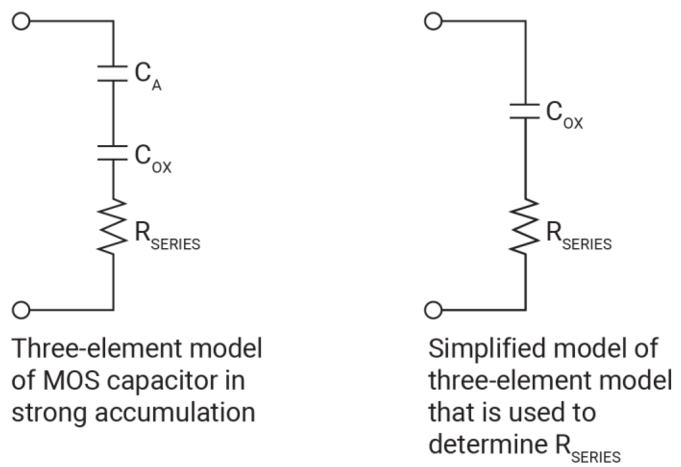
- t_{ox} = oxide thickness (nm)
- A = gate area (cm^2)
- ε_{ox} = permittivity of oxide material (F/cm)
- C_{ox} = oxide capacitance (pF)

You can rearrange the above equation to calculate gate area if the oxide thickness is known. Note that ε_{OX} and other constants are initialized for use with silicon substrate, silicon dioxide insulator, and aluminum gate material, but may be changed for other materials.

Series resistance

The series resistance, R_{SERIES}, is an error term that can cause measurement and analysis errors unless this series resistance error factor is taken into account. Without series compensation, capacitance can be lower than normal, and C-V curves can be distorted. The software compensates for series resistance using the simplified three-element model shown in the simplified model in the following figure. In this model, C_{OX} is the oxide capacitance. CA is the capacitance of the accumulation layer. The series resistance is represented by R_{SERIES}.

Figure 106: Simplified model to determine series resistance



From Nicollian and Brews 224, the correction capacitance, C_C, and corrected conductance, G_C, are calculated as follows:

$$C_C = \frac{(G_M^2 + \omega^2 C_M^2) C_M}{a^2 + \omega^2 C_M^2}$$

and:

$$G_C = \frac{(G_M^2 + \omega^2 C_M^2) a}{a^2 + \omega^2 C_M^2}$$

Where:

- $a = G_M - (G^2_M + \omega^2 C^2_M) R_{SERIES}$
- C_C = series resistance compensated parallel model capacitance
- C_M = measured parallel model capacitance
- G_C = series resistance compensated conductance
- G_M = measured conductance
- R_{SERIES} = series resistance

Gain and offset

Gain and offset can be applied to C_Q and C_H data to allow for curve alignment or to compensate for measurement errors. A gain factor is a multiplier that is applied to all elements of C_Q or C_H array data before plotting or graphics array calculation. Offset is a constant value added to or subtracted from all C_Q and C_H data before plotting or array calculation.

For example, assume that you compare the C_Q and Cn values at reading #3, and you find that C_Q is 2.3 pF less than Cn. If you then add an offset of +2.3 pF to C_Q, the C_Q and C_H values at reading #3 will then be the same, and the C_Q and C_H curves will be aligned at that point.

Gain and offset values do not affect raw C_Q and C_H values stored in the data file, but the gain and offset values are stored in the data file so compensated curves can be easily regenerated at a later date.

Flatband capacitance and flatband voltage

The Model 82 uses the flatband capacitance method of finding flatband voltage, V_{FB}. The Debye length is used to calculate the ideal value of flatband capacitance, C_{FB}. Once the value of C_{FB} is known, the value of V_{FB} is interpolated from the closest V_G values (Nicollian and Brews 487-488).

The method used is invalid when interface trap density becomes very large (1012-1013 and greater). However, this algorithm should give satisfactory results for most users. Those who are dealing with high values of D_{IT} should consult the appropriate literature for a more appropriate method.

Based on doping, the calculation of C_{FB} uses N at 90% W_{MAX}, or user-supplied N_A (bulk doping for p-type, acceptors) or N_D (bulk doping for n-type, donors).

C_{FB} is calculated as follows:

$$C_{FB} = \frac{C_{OX}\ \varepsilon_S A/(1\times10^{-4})(\lambda)}{(1\times10^{-12})(C_{OX}) + \varepsilon_S A/(1\times10^{-4})(\lambda)}$$

Where:

- C_{FB} = flatband capacitance (pF)
- C_{OX} = oxide capacitance (pF)
- ε_S = permittivity of substrate material (F/cm)
- A = gate area (cm^2)
- 1×10^{-4} = units conversion for λ
- 1×10^{-12} = units conversion for C_{OX}

And λ = extrinsic Debye length =

$$(1 \times 10^4)\left(\frac{\varepsilon_S kT}{q^2 N_x}\right)^{1/2}$$

Where:

- kT = thermal energy at room temperature (4,046 × 10^{-21} J)
- q = electron charge (1.60219 × 10^{-19} coulombs)
- N_x = N at 90% W_{MAX}, or N_A, or N_D when input by the user
- N at 90% W_{MAX} is chosen to represent bulk doping

Threshold voltage

The threshold voltage, V_{TH}, is the point on the C-V curve where the surface potential ψ_S, equals twice the bulk potential, ϕ_B. This point on the curve corresponds to the onset of strong inversion. For an enhancement mode MOSFET, V_{TH} corresponds to the point where the device begins to conduct.

V_{TH} is calculated as follows:

$$V_{TH} = \left[\pm \frac{A}{10^{12} C_{OX}} \sqrt{4 \, \varepsilon_S \, q \left| N_{BULK} \right| \left| \phi_B \right|} + 2 \left| \phi_B \right| \right] + V_{FB}$$

Where:

- V_{TH} = threshold voltage (V)
- A = gate area (cm^2)
- C_{OX} = oxide capacitance (pF)
- 10^{12} = units multiplier
- ε_S = permittivity of substrate material
- q = electron charge (1.60219×10^{-19} coulombs)
- N_{BULK} = bulk doping (cm^{-3})
- ϕ_B = bulk potential (V)
- V_{FB} = flatband voltage (V)

Metal semiconductor work function difference

The metal semiconductor work function difference, W_{MS}, is commonly referred to as the work function. It contributes to the shift in V_{FB} from the ideal zero value, along with the effective oxide charge (Nicollian and Brews 462 to 477; Sze 395402). The work function represents the difference in work necessary to remove an electron from the gate and from the substrate, and it is derived as follows:

$$W_{MS} = W_M - \left[W_S + \frac{E_G}{2} - \phi_B \right]$$

Where:

- W_M = metal work function (V)
- W_S = substrate material work function (electron affinity) (V)
- E_G = substrate material bandgap (V)
- ϕ_B = bulk potential (V)

In tests, the values for W_M, W_S, and E_G are listed in the Formulator as constants. You can change the values depending on the type of materials.

For silicon, silicon dioxide, and aluminum:

$$W_{MS} = 4.1 - \left[4.15 + \frac{1.12}{2} - \phi_B\right]$$

$$W_{MS} = -0.61 + \varphi_B$$

$$W_{MS} = -0.61 - \left(\frac{kT}{q}\right) 1n\left(\frac{N_{BULK}}{n_i}\right) \; (DopeType)$$

Where:

- k = Boltzmann constant (1.3807 x 10^{-23} J/K)
- T = Test temperature (K)
- q = Electron charge (1.60219 x 10^{-19} C)
- N_{BULK} = Bulk doping (cm^{-3})
- DopeType = is +1 for p-type materials and −1 for n-type materials; the value for DopeType is changed in the Constants area of the Formulator

For example, for a MOS capacitor with an aluminum gate and p-type silicon (N_{BULK} = $10^{16}cm^{-3}$), W_{MS} = −0.95 V.

For the same gate and n-type silicon (N_{BULK} = $10^{16}cm^{-3}$), W_{MS} = −0.27 V.

Because the supply voltage of modern CMOS devices is decreasing and since aluminum reacts with silicon dioxide, heavily doped polysilicon is often used as the gate material. The goal is to achieve a minimal work-function difference between the gate and the semiconductor, while maintaining the conductive properties of the gate.

Effective oxide charge

The effective oxide charge, Q_{EFF}, represents the sum of oxide fixed charge, Q_F, mobile ionic charge, Q_M and oxide trapped charge, Q_{OT}. Q_{EFF} is distinguished from interface trapped charge, Q_{IT}, in that Q_{IT} varies with gate bias and $Q_{EFF} = Q_F + Q_M + Q_{OT}$ does not (Nicollian and Brews 424-429, Sze 390-395). Simple measurements of oxide charge using C-V measurements do not distinguish the three components of Q_{EFF}.

These three components can be distinguished from one another by temperature cycling, as discussed in Nicollian and Brews, 429, Fig. 10.2. Also, since the charge profile in the oxide is not known, the quantity Q_{EFF} should be used as a relative, not absolute, measure of charge. It assumes that the charge is in a sheet at the silicon-silicon dioxide interface. From Nicollian and Brews, Eq. 10. 10, we have:

$$V_{FB} - W_{MS} = -\frac{Q_{EFF}}{C_{OX}}$$

Note that C_{OX} here is per unit of area. So that,

$$Q_{EFF} = \frac{C_{OX}\left(W_{MS} - V_{FB}\right)}{A}$$

However, since C_{OX} is in F, we must convert to pF by multiplying by 10^{-12} as follows:

$$Q_{EFF} = 10^{-12}\frac{C_{OX}\left(W_{MS} - V_{FB}\right)}{A}$$

Where:

- Q_{EFF} = effective charge (coul/cm^2)
- C_{OX} = oxide capacitance (pF)
- W_{MS} = metal semiconductor work function (V)
- A = gate area (cm^2)

For example, assume a 0.01cm^2 50 pF capacitor with a flatband voltage of −5.95 V, and a p-type $N_{BULK} = 10^{16}$cm^{-3} (resulting in W_{MS} = -0.95 V). In this case, $Q_{EFF} = 2.5 \times 10^{-4}$ coul/cm^2.

The effective oxide charge concentration, N_{EFF}, is computed from effective oxide charge and electron charge as follows:

$$N_{EFF} = \frac{Q_{EFF}}{q}$$

Where:

- N_{EFF} = effective concentration of oxide charge (Units of charge/cm^2)
- Q_{EFF} = effective oxide charge (coulombs/cm^2)
- q = electron charge (1.60219×10^{-19} coulombs)

For example, with an effective oxide charge of 2.5×10^{-8} coul/cm^2, the effective oxide charge concentration is:

$$N_{EFF} = \frac{2.5 \times 10^{-8}}{1.60219 \times 10^{-19}}$$

$$N_{EFF} = 1.56 \times 10^{11} units/cm^2$$

Doping profile

The doping profile of the device is derived from the C-V curve based on the definition of the differential capacitance (measured by the 590 and 595) as the differential change in depletion region charge produced by a differential change in gate voltage (Nicollian and Brews 380-389).

Depletion depth versus gate voltage (VGS)

The Model 82 computes the depletion depth, w, from the high-frequency capacitance and oxide capacitance at each measured value of V_{GS} (Nicollian and Brews 386). In order to graph this function, the program computes each w element of the calculated data array as shown below:

$$w = A\,\varepsilon_S\left(\frac{1}{C_H} - \frac{1}{C_{OX}}\right)$$

Where:

- w = depth (µm)
- ε_S = permittivity of substrate material
- C_H = high-frequency capacitance (pF)
- C_{OX} = oxide capacitance (pF)
- A = gate area (cm^2)

1/C^2 versus gate voltage

A $1/C^2$ graph can yield important information about doping profile. N is related to the reciprocal of the slope of the $1/C^2$ versus V_{GS} curve, and the V intercept point is equal to the flatband voltage caused by surface charge and metal-semiconductor work function (Nicollian and Brews 385).

Doping concentration versus depth

The standard N versus w analysis discussed here does not compensate for the onset of accumulation, and it is accurate only in depletion. This method becomes inaccurate when the depth is less than two Debye lengths.

In order to correct for errors caused by interface traps, the error term $(I-C_Q/C_{OX})/1-C_H/C_{OX})$ is included in the calculations as follows:

$$N = \frac{(-2\times10^{-24})[(1-C_Q/C_{OX})/(1-C_H/C_{OX})]}{A^2 q\,\varepsilon_S}\left[\frac{d}{dV_{GS}}\left(\frac{1}{C_H^2}\right)\right]^{-1}$$

Where:

- N = doping concentration (cm^{-3})
- C_Q = quasistatic capacitance (pF)
- C_{OX} = oxide capacitance (pF)
- $(1-C_Q/C_{OX})/1-C_H/C_{OX})$ = voltage stretchout term
- C_H = high-frequency capacitance (pF)
- A = gate area (cm^2)
- q = electron charge (1.60219×10^{-19} coulombs)
- ε_S = permittivity of substrate material
- 1×10^{-24} = units conversion factor

Interface trap density

Interface trapped charges (Q_{it}) are electrons or holes trapped in localized surface states near the $Si\text{-}SiO_2$ interface. These charges are one of four general types associated with the $Si\text{-}SiO_2$ interface. Interface charges interact electrically with the silicon substrate, which affects MOSFET channel carrier mobility.

Band bending versus gate voltage

As a preliminary step, surface potential ($\psi_S - \psi_0$) vs. V_{GS} is calculated with the results placed in the ψ_S column of the array. Surface potential is calculated as follows:

$$(\Psi_S - \Psi_0) = \sum_{V_{GS}\#1}^{V_{GS}Last} (1 - C_Q / C_{OX})(2V_{STEP})$$

Where:

- ($\psi_S - \psi_0$) = surface potential (V)
- C_Q = quasistatic capacitance (pF)
- C_{OX} = oxide capacitance (pF)
- V_{STEP} = step voltage (V)
- V_{GS} = gate-substrate voltage (V)

Note that the (ψ_S -ψ_0) value is accumulated as the column is built, from the first row of the array (V_{GS} #1) to the last array row (V_{GS} last). The number of rows will, of course, depend on the number of readings in the sweep, which is determined by the Start, Stop, and Step voltages.

Once (ψ_S -ψ_0) values are stored in the array, the value of (ψ_S -ψ_0) at the flatband voltage is used as a reference point and is set to `0` by subtracting that value from each entry in the (ψ_S -ψ_0) column, changing each element in the column to ψ_S.

Interface trap capacitance CIT and density DIT

The density of interface traps (D_{it}) is a function of the silicon orientation and the fabrication process. It is determined by performing simultaneous high frequency and quasistatic C-V sweeps. The measurements are extracted mostly from the depletion and inversion regions near mid-band.

Interface trap density is calculated from C_{IT} as shown below (from Nicollian and Brews 332, see References (on page 6-70)).

$$C_{IT} = \left(\frac{1}{C_Q} - \frac{1}{C_{OX}}\right)^{-1} - \left(\frac{1}{C_H} - \frac{1}{C_{OX}}\right)^{-1}$$

$$D_{IT} = \frac{C_{IT}}{Aq}$$

Where:

- C_{IT} = interface trap capacitance (F)
- D_{IT} = interface trap density (cm^{-2} eV^{-1})
- C_Q = quasistatic capacitance (F)
- C_H = high-frequency capacitance (F)
- C_{OX} = oxide capacitance (F)
- A = gate area (cm^2)
- q = electron charge (1.60219×10^{-19} coulombs)

Mobile ion charge concentration

Mobile ion contaminants in an oxide layer can cause problems in the manufacture and performance of integrated circuits. To measure the concentration of mobile ions in the oxide layer, you can use the triangular voltage sweep (STVS) method, developed by Keithley to monitor mobile ion charge in MOS structures.

You can also use the flatband voltage shift or temperature-bias stress method to measure oxide charge density.

Mobile ion monitoring with triangular voltage sweep (STVS) method

STVS is a technique developed by Keithley to monitor mobile ion charge in MOS structures. Compared with other mobile ion monitoring techniques, such as the BTS and flatband shift methods, it offers faster and more accurate measurement. STVS measures ionic current instead of voltage shift. It has the ability to identify species, and it eliminates the need for temperature cycling of the device under test (DUT). The STVS method has proven to be effective in monitoring mobile ion charge in dielectrics to levels down to $10^9 cm^{-3}$.

The STVS library can perform the corresponding mobile ion charge analysis. It has a built-in correction algorithm to eliminate the problems associated with leakage current. Many parameters, including mobile ion charge concentration, can be extracted from this measurement.

The STVS method improves on the conventional TVS method (discussed below) by measuring both C_Q and C_H and then computing mobile ion charge concentration as follows:

$$N_M = \frac{\sum_{-V_{GS}}^{+V_{GS}} (C_Q - C_H)\Delta V_{GS}}{q}$$

Where:

- N_M = mobile ion density ($1/cm^3$)
- V_{GS} = gate-substrate voltage (V)
- ΔV_{GS} = change in gate-substrate voltage (step voltage) (V)
- C_Q = quasistatic capacitance measured by Model 595 (F)
- C_H = high-frequency capacitance measured by Model 590 (F)
- q = electron charge (coulombs)

Flatband voltage shift method

The primary method for measuring oxide charge density is the flatband voltage shift or temperature-bias stress method (Snow, et al). In this case, two high-frequency C-V curves are measured, both at room temperature. Between the two curves, the device is biased with a voltage at 200-300° to drift mobile ions across the oxide. The flatband voltage differential between the two curves is then calculated, from which charge density can be determined.

From Nicollian and Brews (426, Eq. 10.9 and IO. Io), we have:

$$V_{FB} - W_{MS} = \frac{\bar{x} Q_O}{\varepsilon_{OX}} = \frac{\bar{x} Q_O}{X_O C_{OX}}$$

Where:

- $\bar{x} Q_O$ = the first moment of the charge distribution
- $\bar{x}$ = charge centroid
- W_{MS} = metal semiconductor work function (constant)
- ε_{OX} = oxide dielectric constant
- X_O = oxide thickness
- C_{OX} = oxide capacitance

So that:

$$\Delta V_{FB} = \Delta(V_{FB} - W_{MS})$$

$$\Delta V_{FB} = \Delta \frac{\bar{x} Q_O}{\varepsilon_{OX}}$$

$$\Delta V_{FB} = \frac{Q_O}{C_{OX}} \Delta \frac{\bar{x}}{X_O}$$

For the common case of thermally grown oxide, x (before) = X_O and x (after) = 0, so that

$$\Delta V_{FB} = \frac{-Q_O}{C_{OX}}$$

Where Q_O is the effective charge. Divide Q_O by the gate area to obtain mobile ion charge density per unit area.

Simultaneous triangular-voltage sweep method for determining mobile oxide charges

The simultaneous triangular-voltage sweep (STVS) method is very useful in determining the amount and type of mobile carriers that are in the oxide. This method uses a triangular voltage ramp applied to the gate of the device. The Model 595 applies a similar voltage ramp during its measurement. The Model 595 measures the ionic displacement current, while the device is at an elevated temperature. Elevating the temperature to approximately 300°C causes the high frequency curve to rise in inversion until it is similar to the quasistatic curve. If there are no mobile charges, the quasistatic curve remains approximately the same shape, except the depletion capacitance starts to approach the oxide capacitance. If mobile charges exist, a capacitance spike will appear on the quasistatic C-V curve when the mobile charges move from one side of the oxide to the other.

The quasistatic curve will peak during the movement of the mobile charge. Calculation of the mobile charge involves taking the difference in the high frequency and quasistatic capacitance and multiplying by the change in VGS as shown in the following:

$$N_m = \frac{+V_{GS}}{-V_{GS}} (C_q - C_b) V_{GS} / (qA)$$

Where:

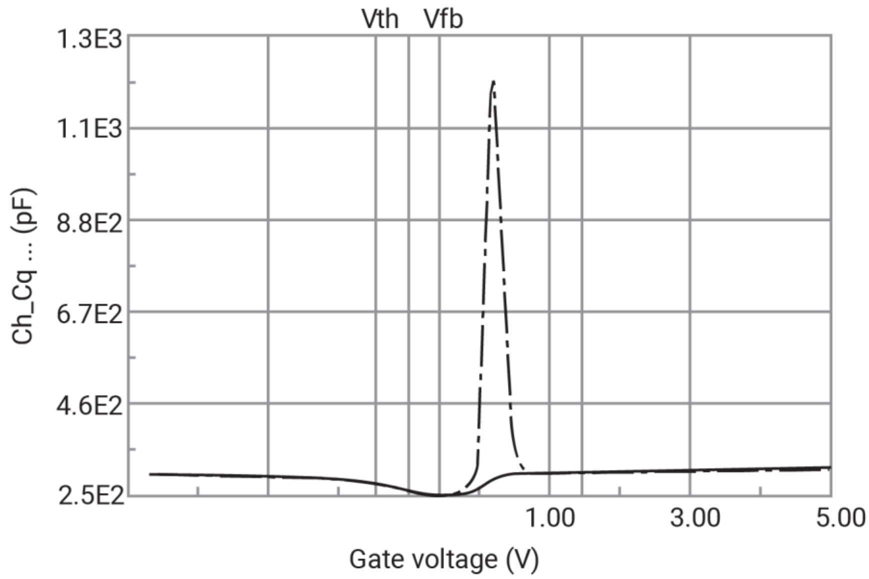
- N_m = mobile ion concentration (cm^{-2})
- $+V_{GS}$ = gate-substance voltage (V)
- $-V_{GS}$ = change in gate-substrate voltage (V)
- C_q = quasistatic capacitance at given VGS (pF)
- C_b = high frequency capacitance at given V_{GS} (capacitance without mobile charges)(pF)
- q = electron charge = $1.60219X10^{-19}$C
- A = area of gate capacitor (cm^2)

The following figure demonstrates what a contaminated oxide should produce for a STVS curve.

This method has four advantages over the BTS method:

1. It determines the mobile charges without interference from the interface trap charges.
2. It can determine the type of ion (sodium or potassium) that is contaminating the oxide, because the peak in gate current for different ions occurs at different gate biases.
3. It provides measurements an order of magnitude more sensitive than bias temperature stress BTS.
4. It is faster than the BTS method, since the device only needs heating once and the calculation needs only one curve.

Figure 107: Simultaneous TVS plot on a highly contaminated wafer



Calculation of the mobile charge concentration could come from the measured V_{GS}, C_q, and C_h data. Alternatively, one can calculate the concentration graphically from the displayed simultaneous C-V curves.

Generation velocity and generation lifetime (Zerbst plot)

Zerbst analysis requires two types of data: C-V and C-t. Important data taken from the C-V measurement includes C_{OX}, C_{MIN}, and doping concentration (N_{AVG} and N_{BULK}). The results of the C-V analysis are integrated with data taken during a C-t measurement to compute generation velocity and generation lifetime of electron-hole pairs. These two parameters are computed from the slope and y-axis intercept of the graph of G/nI vs. w-w_F as outlined in the following computation information.

G/nl computation

$$G / n_l = - \varepsilon_S A N_{AVG} C_{OX} \bullet \left[\frac{\frac{1}{C^2_{t(i+1)}} - \frac{1}{C^2_{t(i-1)}}}{n_l t_{int}} \right] \bullet \left(\frac{1 \text{ x } 10^{12}}{2} \right)$$

Where:

- G = generation rate (s^{-1})
- ε_S = permittivity of semiconductor (F/cm)
- A = gate area (cm^2)
- N_{AVG} = average doping concentration (cm^{-3})
- C_{OX} = oxide (maximum) capacitance (pF)
- $C_{t(i+1)}$ = (i+1) value of measured C-t capacitance (pF)
- $C_{t(i-1)}$ = (i−1) value of measured C-t capacitance (pF)
- n_I = intrinsic carrier concentration (cm^{-3})
- t_{int} = time interval between C-t measurements (s)
- i = [2, #Rdgs−1]

w - wF computation

$$w - w_F = 1{\times}10^{12} \varepsilon_S A \left(\frac{1}{C_{ti}} - \frac{1}{C_{OX}} \right) - w_F$$

$$w_F = 1{\times}10^{12} \varepsilon_S A \left(\frac{1}{C_{ti}} - \frac{1}{C_{OX}} \right)$$

$$w_F = \varepsilon_S A \left(\frac{1}{C_{MIN}} - \frac{1}{C_{OX}} \right)$$

$$w - w_F = \varepsilon_S A \left(\frac{1}{C_{ti}} - \frac{1}{C_{OX}} \right) - w_F$$

Where:

- w = depletion depth (cm)
- w_F = equilibrium inversion depth (cm)
- ε_S = permittivity of semiconductor (F/cm)
- A = gate area (cm^2)
- C_{ti} = i(th) value of measured C-t capacitance (pF)
- C_{MIN} = equilibrium minimum capacitance (pF)

Determining generation velocity and generation lifetime

The generation lifetime, τ_G, is equal to the reciprocal of the slope of the linear portion of the Zerbst plot, while the generation velocity, s, is the y-axis (G/n_I) intercept of the same linear section of the Zerbst plot.

Constants, symbols, and equations used for analysis

In order to perform correct analysis, it may be necessary for you to verify or modify the analysis constants to suit your particular device. Before making measurements, it is strongly recommended that you verify that constants are correct to ensure that your analysis is performed correctly.

Default material constants

The following table lists default material constants, values, descriptions, and symbols.

Default material constants

Symbol	Description	Default value
q	Electron charge (coulombs)	1.60218e-019 coulombs
k	Boltzmann constant (J/°K)	1.38065e-023 J/°K
T	Test temperature (°K)	297.13 °K
εOX	Permittivity of oxide (F/cm)	3.4e-013 F/cm
εS	Semiconductor permittivity (F/cm)	1.04e-012 F/cm
EG	Semiconductor energy gap (eV)	1.12 eV
n_I	Intrinsic carrier concentration (I/cm^3)	1.45e010 cm^{-3}
W_{MS}	Metal work function (V)	4.1 V
W_M	Electron affinity (V)	4.15 V

Data symbols

The following table summarizes data symbols in the library, including a description of each symbol.

Data symbols

Symbol	Description	Units
A	Device gate area.	cm^2
C_{FB}	Flatband capacitance, corresponding to no band bending.	pF
C_H	High-frequency capacitance, as measured by the Model 590 at either 100 kHz or 1 MHz.	pF
C_{HADJ}	The high-frequency capacitance that is adjusted according to gain and offset values. C_{HADJ} is the value that is actually plotted and printed.	pF
C_Q	Quasistatic capacitance as measured by Model 590.	pF
C_{QADJ}	The quasistatic capacitance that is adjusted according to gain and offset values. C_{QADJ} is the value that is actually plotted and printed.	pF

Data symbols

Symbol	Description	Units
C_Q'	Interpolated value of C_Q set to correspond to the quasistatic capacitance at V.	pF
C_{MIN}	Minimum high-frequency capacitance in inversion.	pF
C_{OX}	Oxide capacitance, usually set to the maximum C_H in accumulation.	pF
D_{rr}	Density or concentration of interface states.	$1/cm^2/eV$
E_C	Energy of conduction band edge (valence band is E_V).	eV
E_T	Interface trap energy.	eV
G	High-frequency conductance, as measured by the Model 590 at either 100 kHz or 1 MHz.	S
N_A	Bulk doping for p-type (acceptors).	$1 / cm^3$
N_D	Bulk doping for n-type (donors).	$1 / cm^3$
N_{AVG}	Average doping concentration.	$1 / cm^3$
N_{BULK}	Bulk doping concentration.	$1 / cm^3$
N_{EFF}	Effective oxide charge concentration.	$1 / cm^2$
N(90% W_{MAX})	Doping corresponding to 90% maximum w profile (approximates doping in the bulk).	$1 / cm^3$
N_M	Mobile ion concentration in the oxide.	$1 / cm^3$
Q_{EFF}	Effective oxide charge.	$coul / cm^2$
Q / t	Current measured by the Model 595 at the end of each capacitance measurement with the unit in the capacitance function.	A
R_{SERIES}	Series resistance.	Ω
t_{OX}	Oxide thickness.	nm
V_{GS}	Gate voltage. More specifically, the voltage at the gate with respect to the substrate.	V
V_{FB}	Flatband voltage, or the value of V_{GS} that results in C_{FB}.	V
V_H	Voltage reading sent by Model 590 with matching C_H and G.	V
V_{TH}	The point where the surface potential, ψS, is equal to twice the bulk potential, φB.	V
w	Depletion depth or thickness. Silicon under the gate is depleted of minority carriers in inversion and depletion.	µm
ψ_S	Silicon surface potential as a function of V_{GS}. More precisely, this value represents band bending and is related to surface potential via the bulk potential.	V
ψ_0	Offset in ψS due to calculation method and V_0.	V
ϕ_B	Silicon bulk potential.	V
l	Extrinsic Debye length.	m

Summary of analysis equations

The analysis equations used by the Model 82 software are summarized in the following.

Band bending surface potential

$$(\Psi_S - \Psi_0) = \sum_{V_{GS}\#1}^{V_{GS}Last} (1 - C_Q / C_{OX})(2V_{STEP})$$

Depletion depth

$$w = A\,\varepsilon_S\left(\frac{1}{C_H} - \frac{1}{C_{OX}}\right)$$

Doping concentration

$$N = \frac{(-2\times10^{-24})[(1-C_Q/C_{OX})/(1-C_H/C_{OX})]}{A^2 q\,\varepsilon_S}\left[\frac{d}{dV_{GS}}\left(\frac{1}{C_H^2}\right)\right]^{-1}$$

Effective oxide charge

$$Q_{EFF} = \frac{C_{OX}\left(W_{MS} - V_{FB}\right)}{A}$$

Effective charge concentration

$$N_{EFF} = \frac{Q_{EFF}}{q}$$

Flatband capacitance

$$C_{FB} = \frac{C_{OX}\ \varepsilon_S A/(1\times10^{-4})(\lambda)}{(1\times10^{-12})(C_{OX}) + \varepsilon_S A/(1\times10^{-4})(\lambda)}$$

Where λ = extrinsic DeBye length =

$$(1\times10^4)\left(\frac{\varepsilon_S kT}{q^2 N_x}\right)^{1/2}$$

N_x = N at 90% W_{MAX}, or N_A, or N_D when input by the user

Flatband voltage shift

$$V_{FB} - W_{MS} = \frac{\bar{\chi} Q_o}{\varepsilon_{OX}} = \frac{\bar{\chi} Q_o}{X_o C_{OX}}$$

$$\Delta V_{FB} = \frac{-Q_o}{C_{OX}}$$

Interface trap capacitance and Interface trap density

$$C_{IT} = \left(\frac{1}{C_Q} - \frac{1}{C_{OX}}\right)^{-1} - \left(\frac{1}{C_H} - \frac{1}{C_{OX}}\right)^{-1}$$

$$D_{IT} = \frac{C_{IT}}{Aq}$$

Mobile ion charge concentration – TVS method

$$\sum_{-V_{GS}}^{+V_{GS}} (C_{MEAS} - C_{OX}) \Delta V_{GS} = Q_O$$

Mobile ion charge concentration – STVS method

$$N_M = \frac{\sum_{-V_{GS}}^{+V_{GS}} (C_Q - C_H) \Delta V_{GS}}{q}$$

Oxide thickness / gate area

$$t_{ox} = \frac{A\, \varepsilon_{ox}}{(1 \times 10^{-19}) C_{OX}}$$

Series resistance compensation

$$C_C = \frac{(G_M^2 + \omega^2 C_M^2) C_M}{a^2 + \omega^2 C_M^2}$$

$$G_C = \frac{(G_M^2 + \omega^2 C_M^2) a}{a^2 + \omega^2 C_M^2}$$

$a = G_M - (G^2{}_M + \omega^2 C^2{}_M)\, R_{SERIES}$

Threshold voltage

$$V_{TH} = \left[\pm \frac{A}{10^{12} C_{ox}} \sqrt{4\, \varepsilon_S\, q |N_{BULK}| |\phi_B|} + 2|\phi_B| \right] + V_{FB}$$

Work function

$$W_{MS} = W_M - \left[W_S + \frac{E_G}{2} - \phi_B \right]$$

Zerbst plot (generation lifetime and velocity)

$$G / n_i = -\varepsilon_S A N_{AVG} C_{OX} \bullet \left[\frac{\frac{1}{C^2_{t(i+1)}} - \frac{1}{C^2_{t(i-1)}}}{n_i t_{\text{int}}} \right] \bullet \left(\frac{1 \times 10^{12}}{2} \right)$$

$$w - w_F = 1{\times}10^{12} \varepsilon_S A \left(\frac{1}{C_{ti}} - \frac{1}{C_{OX}} \right) - w_F$$

$$w_F = 1{\times}10^{12} \varepsilon_S A \left(\frac{1}{C_{ti}} - \frac{1}{C_{OX}} \right)$$

$$w_F = \varepsilon_S A \left(\frac{1}{C_{MIN}} - \frac{1}{C_{OX}} \right)$$

$$w - w_F = \varepsilon_S A \left(\frac{1}{C_{ti}} - \frac{1}{C_{OX}} \right) - w_F$$

References

The references below are cited in this chapter:

Nicollian, E.H. and Brews, J.R., *MOS Physics and Technolog*y. Wiley, New York (2003).

Sze, S.M., *Physics of Semiconductor Devices 2nd edition*. Wiley, New York (1985).

Snow, E.H. Grove, A.S., Deal, B.E., and Sah, C.T.J., *Ionic Transport Phenomena in Insulating Films*, Appl. Phys., 36, 1664 (1965).

Bibliography of C-V Measurements

Texts

Grove, A.S., *Physics and Technology of Semiconductor Devices*, Wiley, New York (1967).

Sze, S.M., *Semiconductor Devices, Physics and Technology*, Wiley, New York (1985).

Articles and Papers

Feedback Charge Method

Mego, T.J., *Improved Feedback Charge Method for Quasistatic CV Measurements in Semiconductors*, Rev. Sci. Instr. 57, 11 (1986).

Mego, T.J., "Improved Quasistatic CV Measurement Method for MOS," *Solid State Technology*, 29, 11, 519-21 (1986).

Markgraf, W., Baumann, M., Beyer, A., Arst, P., Rennau, M., *Nutzung der statischen CU-Methode im Ranen eines mikrorechnergesteuerten MOS-Messplatzes*, Phys. d. Halbleiteroberflaeche, 15, 73 (1984).

Q-V Static Method

Ziegler, K. and Klausmann, E., "Static Technique for Precise Measurements of Surface Potential and Interface State Density in MOS Structures," *Appl. Phys. Lett.* 26, 400 (1975).

Kirov, K., Alexsandrova, S., and Minchev, C., "Error in Surface State Determination Caused by Numerical Differentiation of Q-V Data," *Solid State Electronics*, 18, 341 (1978).

Q-C Method and Simultaneous High-low Frequency C-V

Nicollian, E.H. and Brews, J.R., "Instrumentation and Analog Implementation of the Q-C Method for MOS Measurements," *Solid State Electronics*, 27, 953 (1984).

Boulin, D.M., Brews, J.R., and Nicollian, E.H., "Digital implementation of the Q-C Method for MOS Measurements," *Solid State Electronics*, 27, 977 (1984).

Derbenwick, G.F., Automated C-V and |Y|-w Curves for MOS Device Analysis, Sandia Report SAND80-1308 (1982).

Lubzens, D., Kolodny, A., and Shacham-Diamond, Y.J., "Automated Measurement and Analysis of MIS Interfaces in Narrow-Bandgap Semiconductors," *IEEE transactions on Electron Devices*, ED-28, 5 (1981).

Ramp Method

Kuhn, M., "A Quasistatic Technique for MOS C-V and Surface State Measurements," *Solid State Electronics*, 13, 873 (1970).

Castagne, R., "Détermination de la densité d'états lents d'une capacité métak-isolant semiconducteur par l'étude de la charge sour une tension croissant line áirement," *C.R. Acad. Sci* 267, 866 (1968).

Kerr, D.R., "MIS Measurement Technique Utilizating Slow Voltage Ramps," *Int. Conf. Properties and Use of MIS Structures*, Grenoble, France, 303 (1969).

Castagne, R., and Vapaille, A., "Description of the SiO2-Si Interface Properties by Means of Very Low Frequency MOS Capacitance Measurements," *Surface Science*, 28, 157 (1971).

Kuhn, M. and Nicollian, E.H., "Nonequilibrium Effects in Quasi-static MOS Measurements," *J. Electrochem. Soc.*, 118, 373 (1971).

Lopez, A.D., "Using the Quasistatic Method for MOS Measurements", *Rev. Sci. Instr.* 44, 200 (1973).

Interface States / Doping Profiles

Berglund, C.N., "Surface States at Steam Grown Silicon-Silicon Dioxide Interfaces," *IEEE Trans. Electron. Dev.*, 13, 701 (1966).

DeClerck, G., *Characterization of Surface States at the Si-SO2 Interface, Nondestructive Evaluation of Semiconductor Materials and Devices (J.N. Zemel, ed.)*, Plenum Press, New York, p. 105 (1979).

Brews, J.R., "Correcting Interface-State Errors in MOS Doping Profile Determinations," *J. Appl. Phys.* 44, 3228 (1973).

Gordon, B.J., "On-Line Capacitance-Voltage Doping Profile Measurement of Low-Dose Ion Implants," *IEEE Trans. Dev.*, ED-27, 12 (1980).

VanGelder, W., and Nicollian, E.H., "Silicon Impurity Distribution as Revealed by Pulsed MOS C-V Measurements," *J. Electrochem, Soc. Solid State Science*, 118, 1 (1971).

MOS Process Characterization

Zaihinger, K.H. and Heiman, F.P., "The C-V Technique as an Analytical Tool", *Solid State Technology*, 13:5-6 (1970).

McMillian, L., "MOS C-V Techniques for IC Process Control," *Solid State Technology*, 15, 47 (1972).

Zerbst, M., Relaxationseffekte an Halbeiter Isolator-Grenzflaechen, Z.Angnew, Phys. 22, 30 (1966).

Mobile Ion Charge Monitoring

Stauffer, L., et al., "Mobile Ion Monitoring by Simultaneous Triangular Voltage Sweep," *Solid State Technology*, 38, S3 (1995).

Using a Keysight 8110A/8111A Pulse Generator

In this section:

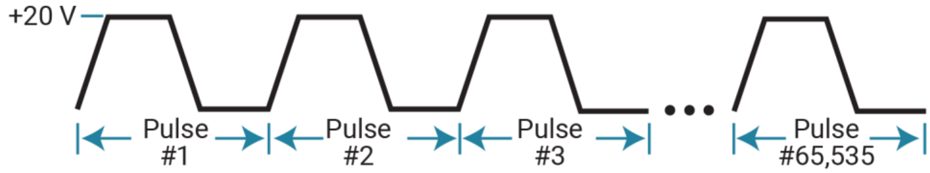
Introduction

NOTE

For details on all aspects of the HP pulse generator operation, refer to the *Keysight Model 8110A User's Manual*.

The 4200A-SCS can control a Keysight Model 8110A Pulse Generator to output from 1 to 65,535 pulses. The following figure shows an example pulse output.

Figure 108: Pulse generator output example



Timing parameters that can be set for the output pulse include pulse delay time, pulse width, pulse period, pulse rise time, and pulse fall time. Details on all parameters for the output pulse are provided in HP8110ulib user library (on page 7-5).

One of the applications for a pulse generator in a semiconductor characterization test system is stress testing. The stress is a burst of pulses applied by the pulse generator to a semiconductor device, such as a flash memory cell. The 4200A-SCS performs before-stress and after-stress characterization tests on the device.

Pulse generator tests

The 4200A-SCS includes the following user modules to run tests using a Keysight pulse generator:

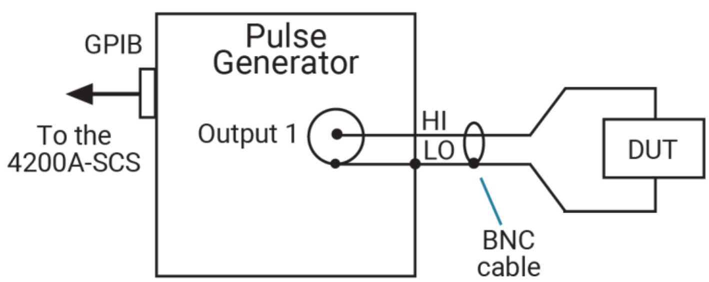
- **PguInit8110: Initialization:** Disables the pulse generator output and returns it to a default setup configuration.
- **PguSetup8110: Set up pulse:** Used to define the output pulse.
- **PguTrigger8110: Trigger output:** Used to specify the number of pulses and trigger the pulse output process.

Details on the user modules for the Keysight pulse generator library are in HP8110ulib user library (on page 7-5).

Signal connections

Basic signal connections for an output of the pulse generator are shown in the following figure. The output LO is connected to the chassis of the pulse generator.

Figure 109: Basic pulse generator connections to DUT



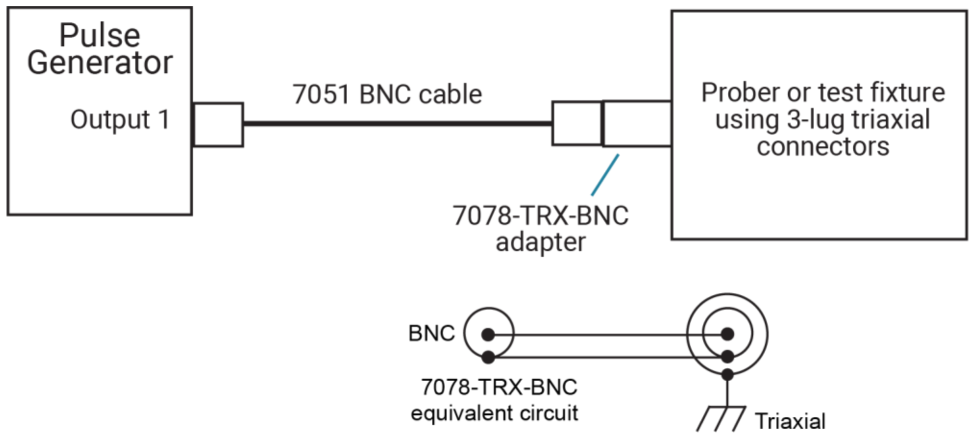
Triaxial connections

Adapters are required to connect the pulse generator to equipment that uses triaxial connectors, such as the probe station, test fixture, and matrix card.

Probe station and test fixture connections

The following figure shows connections to a probe station or a test fixture that is equipped with 3-slot triaxial connectors. The 7078-TRX-BNC is a 3-lug triaxial to BNC adapter. As shown, connect the adapter to the 3-slot triaxial connector and then use a 7051 BNC cable to make the connection to the pulse generator. This figure also shows the equivalent circuit for the adapter.

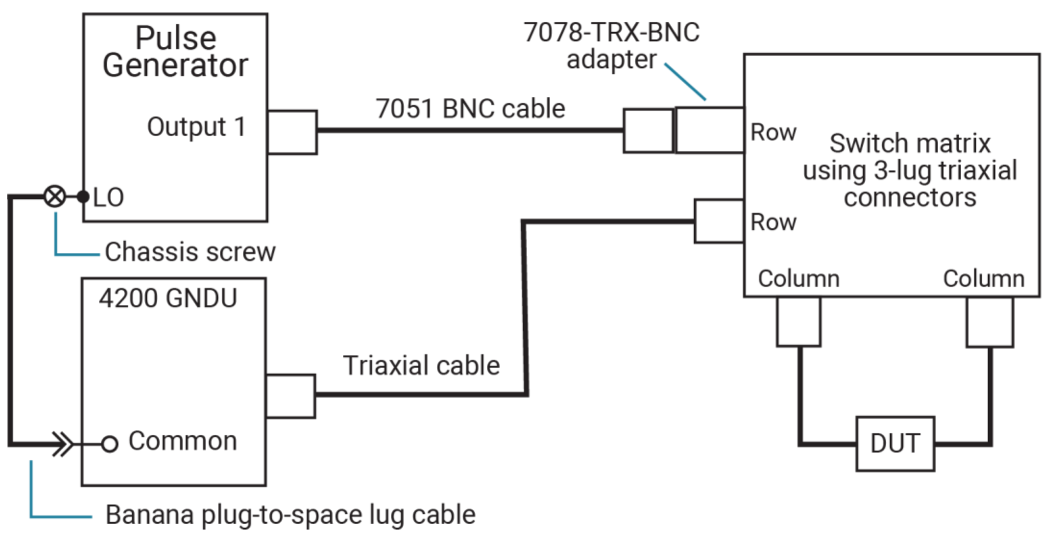
Figure 110: Connections to prober or test fixture equipped with triaxial connectors



Switching matrix connections

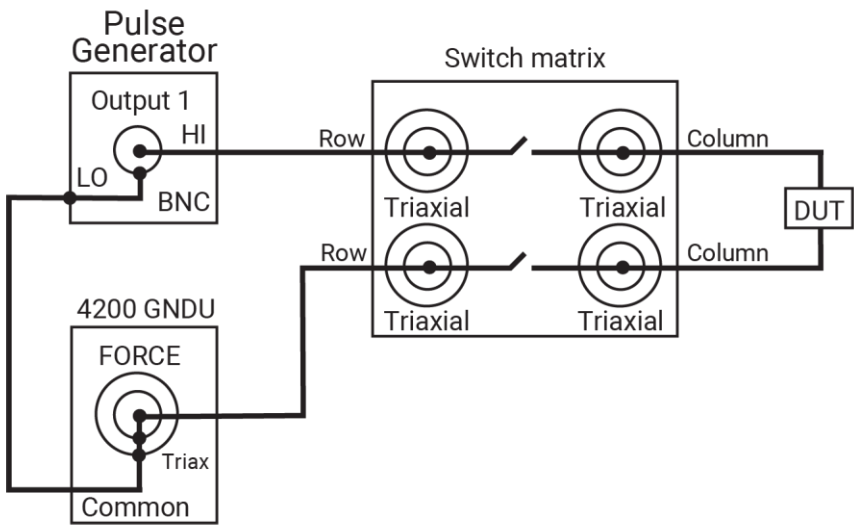
When using a switching matrix that is equipped with triaxial connectors, separate HI-to-LO matrix paths are required for the pulse generator. A typical connection scheme for this type of switching matrix is shown in the following figure. As shown, OUTPUT 1(HI) is connected to a matrix row and the return path (LO) from the switching matrix is connected to the ground unit (GNDU). To complete the return path, a separate cable connection from GNDU to the chassis of the pulse generator is required. The chassis of the pulse generator is output LO.

Figure 111: Connections to switching matrix equipped with triaxial connectors



The following figure shows the actual pulse output signal path through the switching matrix to the device under test (DUT) and back to the pulse generator. A more detailed look at signal paths is provided in Using Switching Matrices (on page 2-1).

Figure 112: Pulse output signal path



GPIB connections

The 4200A-SCS controls the pulse generator through the general purpose interface bus (GPIB). Use a shielded GPIB cable to connect the GPIB port of the pulse generator to the GPIB port of the 4200A-SCS.

Using KCon to add a Keysight pulse generator to the system

In order for the 4200A-SCS to control an external instrument, that instrument must be added to the system configuration. The pulse generator is added to the test system using the Keithley Configuration Utility (KCon).

Refer to “Use KCon to add equipment to the 4200A-SCS” in *Model 4200A-SCS Parameter Analyzer Setup and Maintenance* for instruction.

HP8110ulib user library

Use the user modules in the `HP8110ulib` user library to control a Keysight Model 8110A Pulse Generator. These user modules are summarized in the following table. The table also lists the user test modules (UTM) created by Keithley that use the user modules.

HP8110ulib user modules

User Module	UTM Name	Description
`PguInit8110`	`pgu1-init`	Initializes the pulse generator to the default setup.
`PguSetup8110`	`pgu1-setup`	Sets the output pulse parameters.
`PguTrigger8110`	`pgu-trigger`	Specifies pulse count and trigger start of output.

PguInit8110 user module

This user module initializes the pulse generator to a default setup.

Usage

`status = PguInit8110(char *instr_id);`

status	Returned values are placed in the Analyze sheet: ■ `0`: OK ■ `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist ■ `-10040 (HP8110_NOT_IN_KCON)`: No PGU was found in the system configuration ■ `-10041 (HP8110_NOT_INITED)`: The PGU was never initialized ■ `-10042 (HP8110_PULSE_ERROR)`: There was an error during pulsing ■ `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred ■ `-10091 (GPIB_TIMEOUT)`: A time-out occurred during communications ■ `-10100 (INVAL_PARAM)`: An invalid input parameter is specified
instr_id	The PGU (pulse generator) instrument ID: PGU*X*, where *X* is a number from 1 through 8 (configuration dependent); the PGU instrument ID effectively corresponds to a single pulse generator channel

Details

The user module used by the `pgu1-init` UTM.

The `PguInit8110` user module initializes the Keysight 8110A pulse generator as follows:

- Disables the output of the specified channel.
- Resets (*`RST`) to ensure that all errors are cleared.
- Sets the output polarity to NORMAL.
- Sets the trigger count to 1.
- Sets the trigger source to MANUAL.
- Enables SINGLE PULSE mode.
- Allows the rise/fall to be independently programmable.
- Sets the pulse height to 0.2 V and base to 0 V.
- Sets the rise/fall to 100e-9 s.
- Sets the width to 300e-9 s.
- Disables error checking.

Also see

None

PguSetup8110 user module

This user module defines the output pulse of the pulse generator (PGU).

Usage

```
status = PguSetup8110(char *instr_id, double DelayTime, double RiseTime, double
   FallTime, double Width, double Period, double BaseValue, double Amplitude,
   double OutImpedance, double LoadImpedance, double OutpEnable);
```

status	Returned values are placed in the Analyze sheet: ▪ `0`: OK ▪ `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist ▪ `-10040 (HP8110_NOT_IN_KCON)`: No PGU was found in the system configuration ▪ `-10041 (HP8110_NOT_INITED)`: The PGU was never initialized ▪ `-10042 (HP8110_PULSE_ERROR)`: There was an error during pulsing ▪ `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred ▪ `-10091 (GPIB_TIMEOUT)`: A time-out occurred during communications ▪ `-10100 (INVAL_PARAM)`: An invalid input parameter is specified
instr_id	The PGU instrument ID: `PGU`*X*, where *X* is a number from 1 through 8 (configuration dependent); the PGU instrument ID effectively corresponds to a single pulse generator channel
DelayTime	The amount of time to delay after receiving the trigger (0 s to 0.999 s)
RiseTime	Sets the pulse rise time (2e-09 s to 0.2 s)

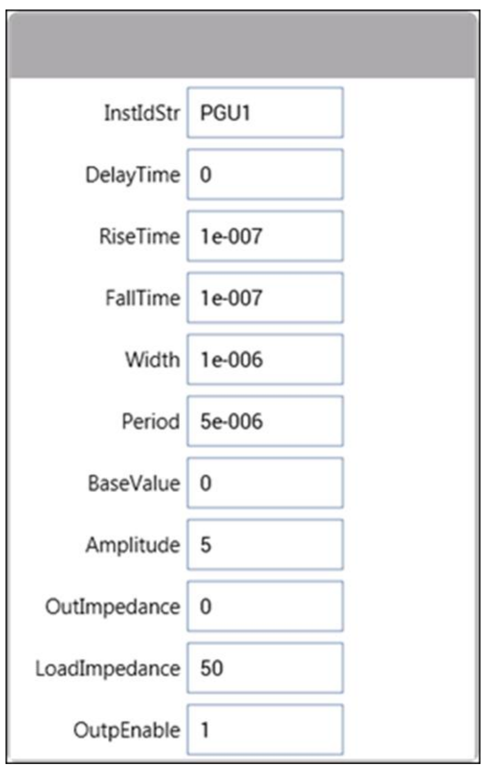
FallTime	Sets the pulse fall time (2e-09 s to 0.2 s)
Width	Sets the pulse width (3.3e-09 s to 0.999 s)
Period	Sets the period to use if more than one pulse will be triggered; if a single pulse is output (as opposed to a burst of pulses), this parameter is ignored; (6.65e-09 to 999; 6.65e-09 to 0.999 if there is no PLL option installed in the pulse generator)
BaseValue	The base value of the pulse (–20 V to +20 V); for a pulse with no DC offset, set this parameter to 0
Amplitude	The amplitude of the pulse as measured from the base value (–20 V to +20 V)
OutImpedance	Sets the output impedance of the PGU: ▪ `0`: 50 Ω ▪ `1`: 1000 Ω
LoadImpedance	The expected impedance of the load (DUT) (0 to 999 kΩ); if unsure, enter the maximum value
OutpEnable	A flag that determines whether to enable or disable the output relay of the PGU: ▪ `0`: Disable the output ▪ `1`: Enable the output

Details

The `PguSetup8110` user module defines the pulse timing and voltage settings. Once defined, the pulse can be triggered using the `PguTrigger8110` user module.

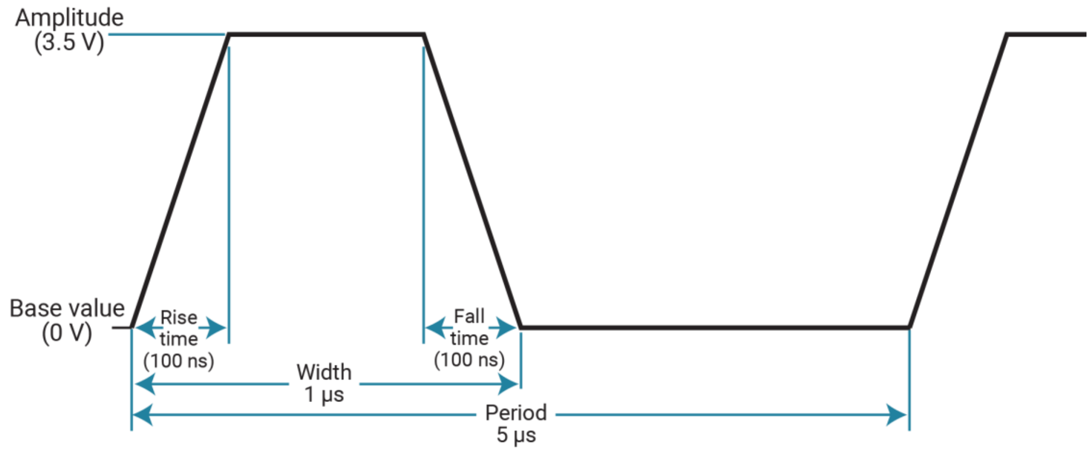
The following figure shows the default parameters for `pgu1-setup` UTM.

Figure 113: PguSetup8110 (pgu1-setup UTM)



The following figure shows the output pulse for the default UTM setup.

Figure 114: pgu1-setup UTM pulse specifications



Also see

PguTrigger8110 (on page 7-8)

PguTrigger8110 user module

This user module specifies number of pulses to output and triggers the start of the pulse output process.

Usage

`status = PguTrigger8110(char *InstIdStr, double Count);`

status	Returned values are placed in the Analyze sheet: ■ `0`: OK ■ `-10000 (INVAL_INST_ID)`: The specified instrument ID does not exist ■ `-10040 (HP8110_NOT_IN_KCON)`: No PGU was found in the system configuration ■ `-10041 (HP8110_NOT_INITED)`: The PGU was never initialized ■ `-10042 (HP8110_PULSE_ERROR)`: There was an error during pulsing ■ `-10090 (GPIB_ERROR_OCCURRED)`: A GPIB communications error occurred ■ `-10091 (GPIB_TIMEOUT)`: A timeout occurred during communications ■ `-10100 (INVAL_PARAM)`: An invalid input parameter is specified
InstIDStr	The PGU (pulse generator) instrument ID: `PGU`*X*, where *X* is a number from 1 through 8 (configuration dependent); the PGU instrument ID effectively corresponds to a single pulse generator channel
Count	The number of pulses to output; if *Count* is > 1, a burst of pulses with a period as defined in the `PguSetup8110` function is output; if *Count* is 1, a single pulse is output

Details

The `PguTrigger8110` function triggers the pulse (or pulses) defined using the `PguSetup8110` function.

Also see

PguSetup8110 user module (on page 7-6)

Set up a probe station

In this section:

Prober control overview

Semi-automatic and fully-automatic probe stations are typically controlled programmatically through a GPIB or RS-232 communications interface. In this situation, the 4200A-SCS acts as the system controller and is connected to the probe station using the appropriate communications interface.

The 4200A-SCS facilitates automated wafer-level testing through various prober control mechanisms. Standard prober drivers are included with the 4200A-SCS, and a number of commercially available automated probe stations are supported. The 4200A-SCS can control supported probers without requiring the user to develop any additional software.

For probers that are not supported by the standard drivers, the open architecture of the 4200A-SCS software allows you to integrate prober control into the test flow by creating a user library.

A probe station is controlled by the 4200A-SCS with user modules. User modules are created in the Keithley User Library Tool. Refer to *Model 4200A-SCS Parameter Analyzer KULT and KULT Extension Programming* for more information regarding user libraries.

The `PRBGEN` library of prober user modules is provided with the 4200A-SCS to simplify prober control. This generic prober user library, developed and maintained by Keithley, allows Clarius to control all supported probers in the same manner. Therefore, Clarius projects that use `PRBGEN` work with any prober supported by Keithley. Refer to Supported probers (on page 8-4) for a list of supported probers and links to additional information.

Many of the PRBGEN user modules have already been built into Clarius as actions. You can add these actions to the project tree in any location that makes sense for your system. The position of the action in the project tree determines when the action is run during a test. For example, in a device with multiple tests, the device level can be run directly, which executes each test under the device sequentially. If an action also exists in the device level, the action runs in sequence with the tests. Similarly, actions under the subsite, site, or project levels execute automatically when the subsite, site, or project is run.

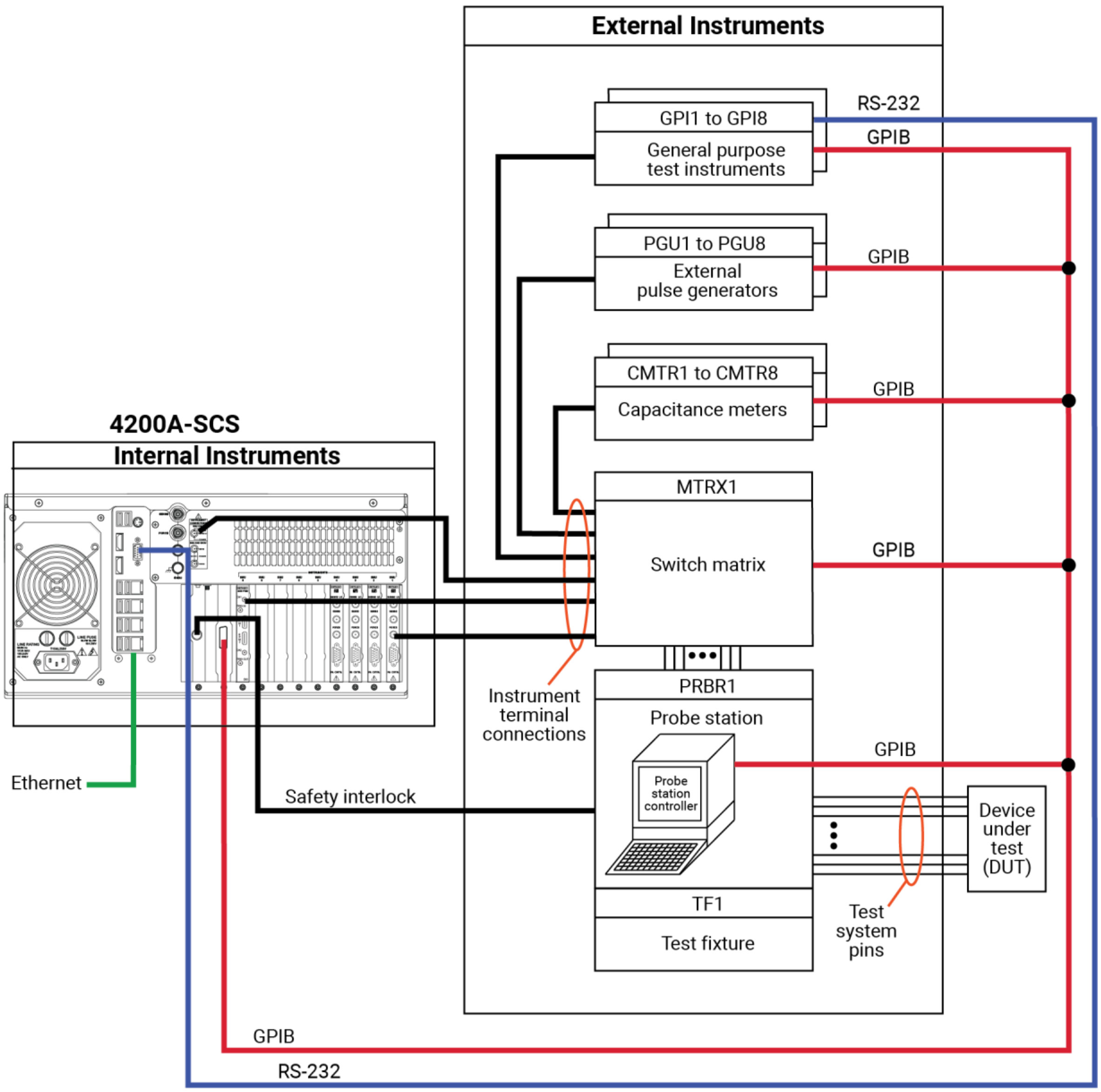
You can connect 4200A-SCS measurement signals to most commercially available wafer probers. Probers that provide triaxial connections to their probes and chuck are the easiest because of the triaxial connections on the SMU, 4200-PA, and GNDU. For other connections, you can get adapters and cable kits from Keithley that allow the 4200A-SCS to be adapted to any connection environment.

WARNING

Turning the 4200A-SCS output off does not place the instrument in a safe state (an interlock is provided for this function). Hazardous voltages may be present on all output and guard terminals. To prevent electrical shock that could cause injury or death, never make or break connections to the 4200A-SCS while the instrument is powered on. Turn off the equipment from the front panel or disconnect the main power cord from the rear of the 4200A-SCS before handling cables. Putting the equipment into an output-off state does not guarantee that the outputs are powered off if a hardware or software fault occurs. Precautions must be taken to prevent a shock hazard by surrounding the test device and any unprotected leads (wiring) with double insulation rated for 250 V, Category O.

Basic system connections are illustrated in the following figure.

Figure 115: Example system connections



Supported probers

For the latest list of supported probers, refer to the *4200A-SCS Parameter Analyzer Datasheet*.

Supported probe station	Additional information
Cascade Microtech (Karl Suss) Model PA200	Cascade Microtech PA200 Prober (on page 9-1)
Micromanipulator Model 8860	Micromanipulator 8860 Prober (on page 10-1)
Micromanipulator Model P200L	Using a Micromanipulator P200L Prober (on page 11-1)
Manual or Fake	Using a Manual or Fake Prober (on page 12-1)
Formfactor Cascade Summit-12000	Cascade Summit-12000 Prober (on page 13-1)
Signatone CM500 (WL250)	Signatone CM500 Prober (on page 14-1)
MPI TS2000, TS2000-DP, TS2000-HP, TS2000-SE, TS3000, and TS3000-SE	Using an MPI Probe Station (on page 15-1)

NOTE

Contact Keithley for the most up-to-date list of supported probe stations.

Use KCon to add the prober to the instrument list. See the information for the specific prober for details. For information on KCon, refer to “Keithley Configuration Utility (KCon)” in *Model 4200A-SCS Setup and Maintenance*.

PRBGEN user modules

Prober-control software provided by supported prober vendors gives access to the full feature set of each prober. You use the prober-control software to define a list of wafer locations to be probed. The 4200A-SCS relies on the prober controller and associated software to maintain this probe list. The `PRBGEN` user modules communicate with the prober controller, normally through the GPIB bus or COM1 (serial bus) port, to instruct it to step through the probe list. This technique of prober control is referred to as learn mode because the prober-control software is taught the physical location of each probe location. The following table summarizes the user modules included in the `PRBGEN` prober control user library.

User module	Description
`PrInit`	Initializes the prober driver and establishes the reference site or die.
`PrChuck`	Instructs the prober to move the probe station into contact or to break contact between the wafer and the test system pins (probe needles).
`PrSSMovNxt`	Instructs the prober to move to the next subsite or test element group in the probe list.
`PrMovNxt`	Instructs the prober to move to the next site or die in the probe list.

Before you can execute a Clarius project that uses the `PRBGEN` user library, you must create the probe list using the appropriate vendor-specific prober-control software. Instructions for creating the probe list for each supported prober are included in this manual (refer to Supported probers (on page 8-4)).

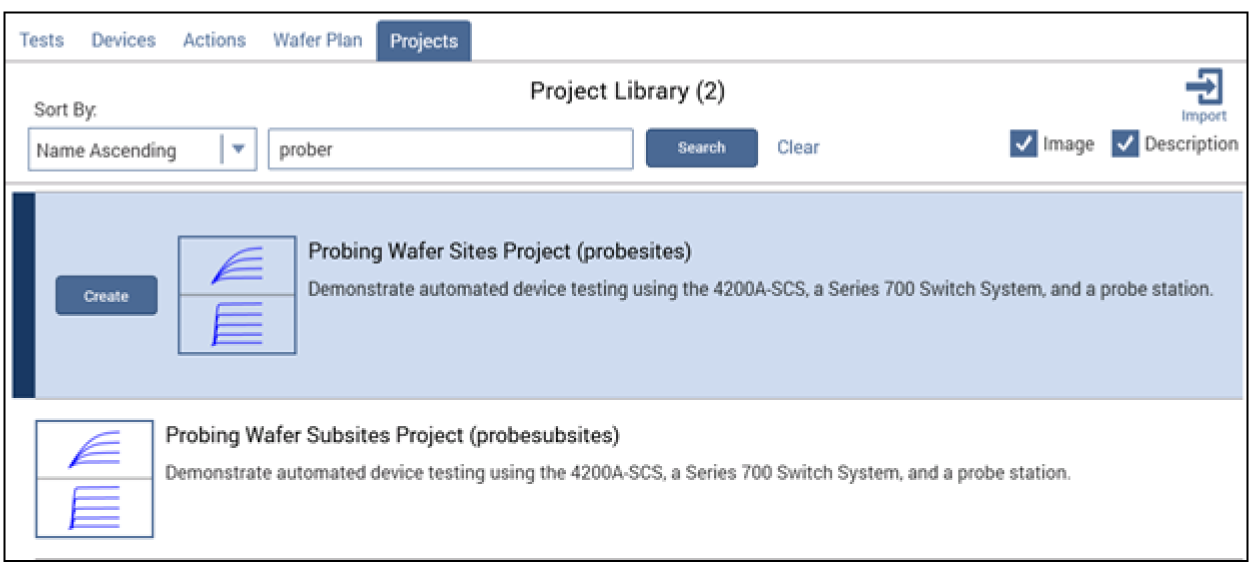
The example projects in the following topics describe a typical project setup. However, you can add and arrange sites, subsites, and prober actions in the project in any order that is appropriate for your system. The 4200A-SCS runs the items in the project in the order in which they are presented, from top to bottom. You can also select the starting point for each run. For example, if you highlight a device, only the tests and actions that are selected and under that device will be used when you select Run.

If you use a semi-automatic prober, a Clarius probe action only triggers movements that are already programmed in the prober controller. Each execution of the action advances the prober to the next site in this programmed sequence. Site numbers are not communicated between the prober and Clarius. Therefore, if you evaluate multiple sites, the range of site numbers that you specify in the Clarius Project window must agree with the sequence of site numbers in the prober controller program.

Set up a Clarius project that controls a prober:

1. Choose **Select**.
2. Select **Projects**.
3. Search for **probesites** or **probesubsites**.
4. Select **Create** to add the prober project to the project tree.

Figure 116: probesites project



Example test execution sequence: probesites project

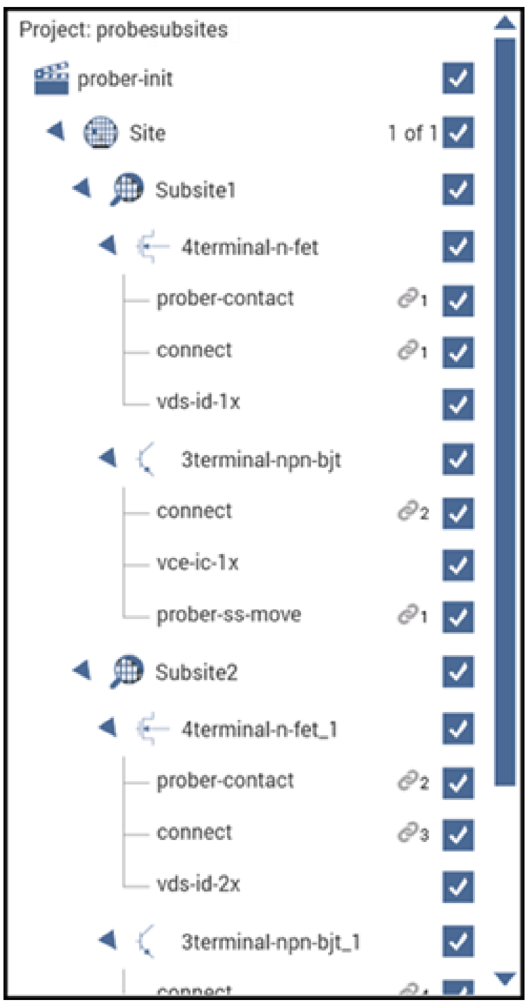
Configure the probesites project:

1. In Clarius, select **Configure**.
2. In the project tree, select **probesites**.
3. Set the Project Execution Loop Settings as needed for your project.
4. Select **Run**.
5. Select **Analyze** to review the data.

Example test execution sequence: probesubsites project

In this example, the `probesubsites` project is selected. When you run the test for the site, tests are run for each of the subsites.

Figure 117: probesubsites project tree



Understanding site coordinate information

The next topics describe the reference site, probe sites, and chuck movement.

Reference site (die)

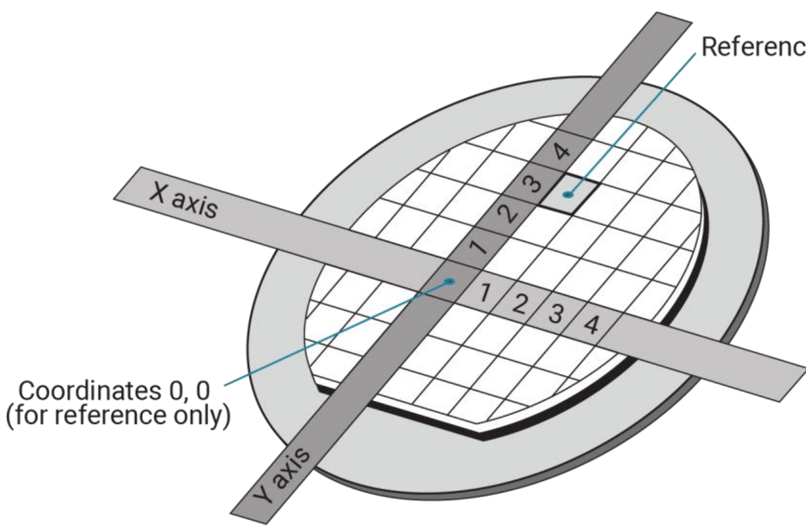
The designated reference site is defined in the `prober-init` action by selecting **Configure** and entering the parameters. This is the first stopping point of the prober once aligned. The physical location of the reference site may be any coordinate that is selected on the wafer and is selected for probing or marked for probing through the prober software. The coordinate system of the wafer is also defined through the prober software. For example, the coordinates of the reference site shown in the following figure are (3, 1).

For parameter descriptions, refer to the Help pane.

NOTE

The defined reference site must match the physical location of the wafer. This is the location on the wafer directly below the probe pins after the wafer has been loaded.

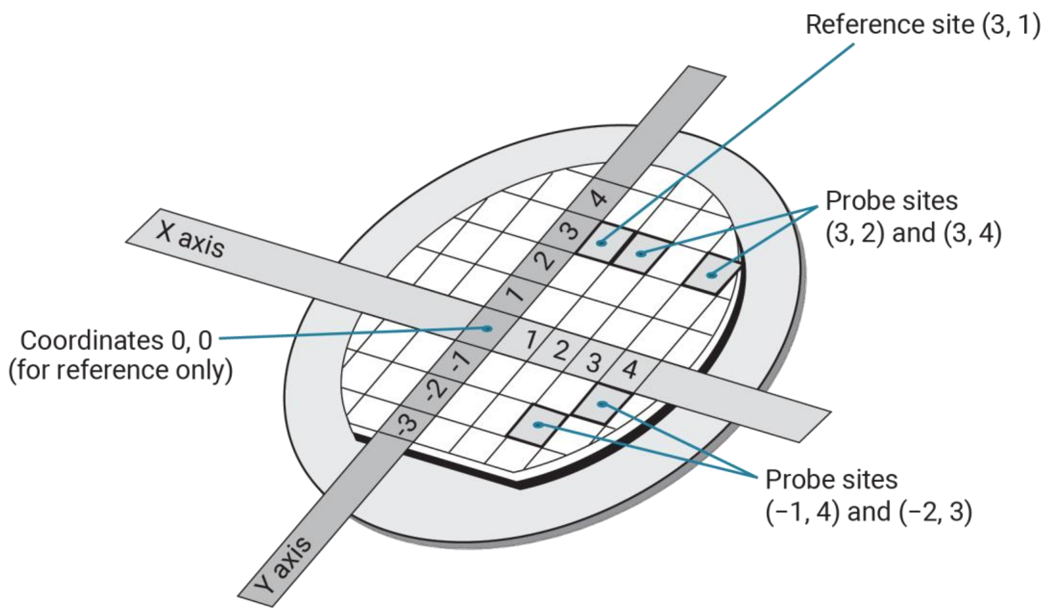
Figure 118: Sample reference site location



Probe sites (die)

Dies marked as probe sites in the prober software define the areas to be tested. The physical location of the probe site can be any coordinates selected on the wafer. Marking a die as a probe site also selects the site for probing. The coordinates of each probe site are referenced with respect to the coordinates of the reference site. For example, with the reference site of (3, 1), the coordinates of the five probe sites shown in the following figure are (3,1), (3, 2), (3, 4), (-1, 4), and (-2, 3).

Figure 119: Sample probe site location



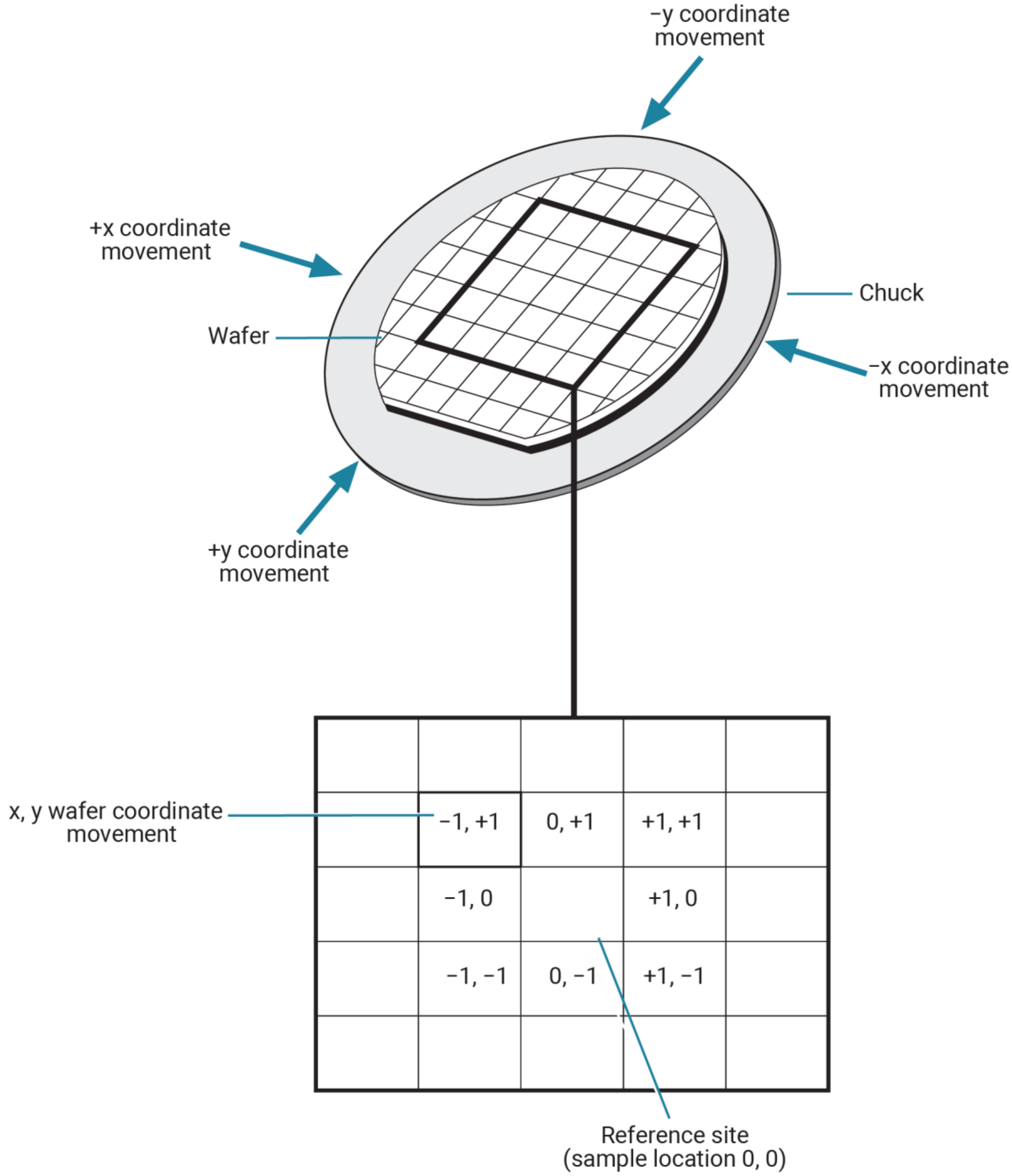
Chuck movement

Coordinate movements are described using a first quadrant coordinate system and x, y coordinates (+x values move east and +y values move north). To accommodate this system, you must configure the correct quadrant (prober dependent). Applicable quadrant setup instructions are in the chapter for your prober. When you specify chuck movements, use the coordinates of the site. The chuck will automatically move in the proper direction to position the probe pins over the correct die. For example, to move from the reference site to the die up one and over one, command the chuck to move (1, 1). Refer to the following figure for a representation of the relationship between chuck movement and (x, y) coordinates.

NOTE

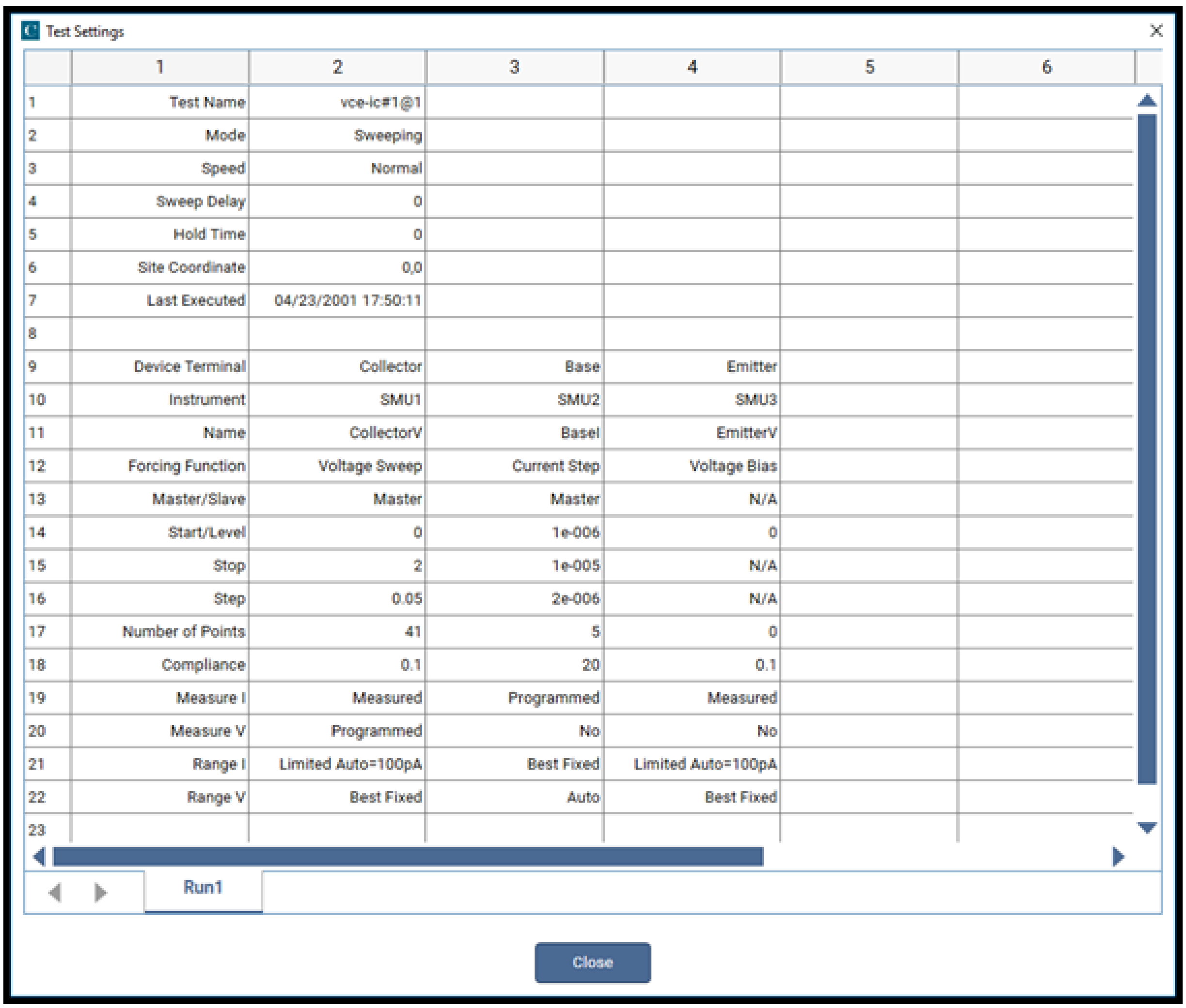
The chuck moves and the probe pins remain stationary. Notice that the chuck movement is opposite of the coordinate system of the probe pins.

Figure 120: Chuck movement



At the conclusion of a test, the site coordinates are recorded in the sheet settings. These coordinates are only valid if a project uses the remote prober control (real prober). The coordinate system is based on the `xstart_position` and `ystart_position` parameters of the `prober-init` action. The site coordinates change only after a site movement is performed; the coordinates change when the bottom of the project loop is reached. At the top of each iteration, the site coordinates remain the same until the site movement is done. The following figure shows an example of a table that contains site coordinates (see column 2, row 6).

Figure 121: Clarius: Example of site coordinates: Analyze sheet



	1	2	3	4	5	6
1	Test Name	vce-ic#1@1				
2	Mode	Sweeping				
3	Speed	Normal				
4	Sweep Delay	0				
5	Hold Time	0				
6	Site Coordinate	0,0				
7	Last Executed	04/23/2001 17:50:11				
8						
9	Device Terminal	Collector	Base	Emitter		
10	Instrument	SMU1	SMU2	SMU3		
11	Name	CollectorV	BaseI	EmitterV		
12	Forcing Function	Voltage Sweep	Current Step	Voltage Bias		
13	Master/Slave	Master	Master	N/A		
14	Start/Level	0	1e-006	0		
15	Stop	2	1e-005	N/A		
16	Step	0.05	2e-006	N/A		
17	Number of Points	41	5	0		
18	Compliance	0.1	20	0.1		
19	Measure I	Measured	Programmed	Measured		
20	Measure V	Programmed	No	No		
21	Range I	Limited Auto=100pA	Best Fixed	Limited Auto=100pA		
22	Range V	Best Fixed	Auto	Best Fixed		
23						

PRBGEN user library

The PRBGEN user library provides test modules to initialize the prober, move to the next site or subsite in the wafer map of the prober, make or break contact between the probes and the wafer, and get the X position and Y position of the prober. It allows Clarius to control all supported probers in the same manner. Clarius projects that use PRBGEN work with any prober supported by Keithley.

The user modules in the PRBGEN user library are provided as actions in Clarius.

PRBGEN user modules

User module	Clarius action	Description
PrChuck	prober-contact	Directs the prober to have the probe pins contact the wafer or separate the pins from the wafer.
PrInit	prober-init	Initializes the prober with die size, first coordinate (X and Y), units (mm or mils), and mode information.
PrMovNxt	prober-move	In learn mode, the PrMovNxt command causes the prober to move to the next site after inking.
PrSSMovNxt	prober-ss-move	In learn mode, the PrSSMovNxt command causes the prober to move to the next subsite after inking.

PrInit

This command initializes the prober with die size, first coordinate (X and Y), units (mm or mils), and mode information.

Usage

```
status = PrInit(int mode, double x_die_size, double y_die_size, int
   x_start_position, int y_start_position, int units, int subprobtype);
```

status	Returned values; see **Details**
mode	The mode to be used with the prober (see **Details**): ■ 1: Manual prober ■ 2: External automatic prober ■ 6: Learn (typically used with semi-automatic probers)
x_die_size	The x die size (units are set by the *units* parameter)
y_die_size	The y die size (units are set by the *units* parameter)
x_start_position	The x location of the prober position at alignment
y_start_position	The y location of the prober position at alignment
units	The units: ■ 0: Mils ■ 1: Millimeters
subprobtype	Not supported for 4200A-SCS

Library

Dependency: PRBCOM

Details

The mode defines the capabilities of the prober. Select External automatic mode when the tester explicitly directs all the prober actions. Use Learn mode when the prober is configured with all the wafer stepping information. When learn is selected, the tester commands the prober to do the next operation. Please confirm the correct mode of operation for each specific application. Supported modes vary from prober to prober.

The `PrInit` function returns the values:

- `1`: Success (OK)
- `-1005`: Failure setting units
- `-1008`: Failure setting mode
- `-1009`: Failure setting die size
- `-1011`: Operation invalid in mode
- `-1013`: Unintelligible response
- `-1015`: Unexpected error
- `-1017`: Bad chuck position
- `-1027`: Invalid parameter

Example

```
status = PrInit(6,2,2,1,1,1,0);
```

Also see

None

PrChuck

This command directs the prober to have the probe pins contact the wafer or separate the pins from the wafer.

Usage

```
status = PrChuck(int chuck_position);
```

status	Returned values; see **Details**
chuck_position	The chuck position: ■ `0`: Separate from the chuck ■ `1`: Contact the chuck

Library

Dependency: PRBCOM

Details

The `PrChuck` function returns the values:

- `1`: Success (`PR_OK`)
- `-1006`: Invalid mode
- `-1013`: Unintelligible response
- `-1015`: Unexpected error
- `-1017`: Bad chuck position

Example

```
status = PrChuck(1);
```

Also see

None

PrSSMovNxt

In learn mode, the `PrSSMovNxt` command causes the prober to move to the next subsite. If needed, you can specify the inker to fire before the move.

Usage

```
status = PrSSMovNxt(int ink_number);
```

status	Returned values; refer to **Details**
ink_number	The inkers to fire: ▪ `0`: No inker; move only ▪ `1`: 1 ▪ `2`: 2 ▪ `3`: 1, 2 ▪ `4`: 3 ▪ `5`: 1, 3 ▪ `6`: 2, 3 ▪ `7`: 1, 2, 3 ▪ `8`: 4 ▪ `9`: 1, 4 ▪ `10`: 2, 4 ▪ `11`: 1, 2, 4 ▪ `12`: 3, 4 ▪ `13`: 1, 3, 4 ▪ `14`: 2, 3, 4 ▪ `15`: All 4

Library

Dependencies: PRBCOM

Details

The `PrMovNxt` function returns the values:

- `1`: Success (`PR_OK`)
- 2: Prober moved to next die (confirmed)
- `4`: Next wafer loaded (confirmed)
- `-1008`: Invalid mode
- `-1011`: Operation invalid in mode
- `-1013`: Unintelligible response
- `-1014`: Movement failure
- `-1015`: Unexpected error
- `-1027`: Invalid parameter

Example

```
status = PrSSMovNxt(0);
```

Also see

None

PrMovNxt

In learn mode, the `PrMovNxt` command causes the prober to move to the next site. If needed, you can specify the inker to fire before the move.

Usage

`status = PrMovNxt(int ink_number);`

status	Returned values; refer to **Details**
ink_number	The inkers to fire: ■ `0`: No inker; move only ■ `1`: 1 ■ `2`: 2 ■ `3`: 1, 2 ■ `4`: 3 ■ `5`: 1, 3 ■ `6`: 2, 3 ■ `7`: 1, 2, 3 ■ `8`: 4 ■ `9`: 1, 4 ■ `10`: 2, 4 ■ `11`: 1, 2, 4 ■ `12`: 3, 4 ■ `13`: 1, 3, 4 ■ `14`: 2, 3, 4 ■ `15`: All 4

Library

Dependencies: PRBCOM

Details

The `PrMovNxt` function returns the values:

- `1`: Success (`PR_OK`)
- `4`: Next wafer loaded (confirmed)
- `-1008`: Invalid mode
- `-1011`: Operation invalid in mode
- `-1013`: Unintelligible response
- `-1014`: Movement failure
- `-1015`: Unexpected error
- `-1027`: Invalid parameter

Example

```
status = PrMovNxt(0);
```

Also see

None

Tutorial: Control a probe station

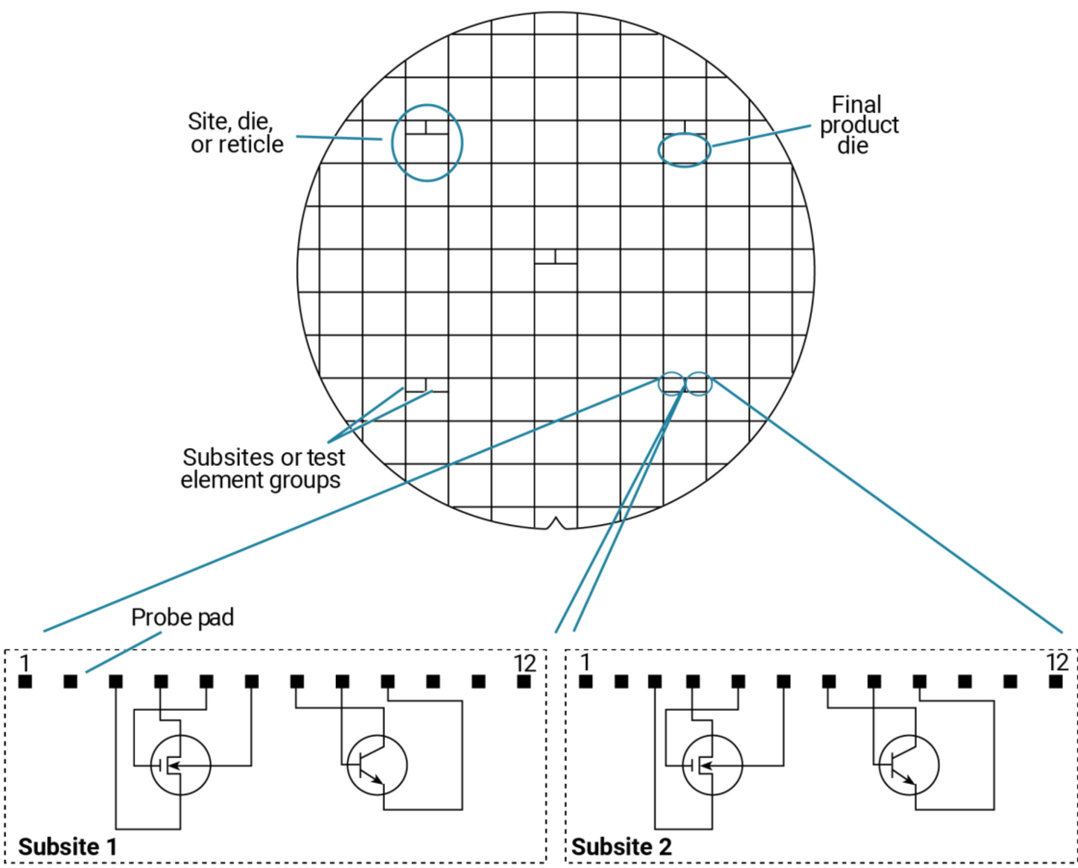
This tutorial demonstrates how to control a probe station to test five identical sites (or die or reticles) on a sample wafer.

Each wafer site has two subsites (or test element groups). At each subsite there are two devices (or test elements) to be tested:

- 4-terminal N-channel MOSFET
- 3-terminal NPN transistor

The subsites do not need to be identical, but for simplicity they are assumed to be the same. This is illustrated in the following figure.

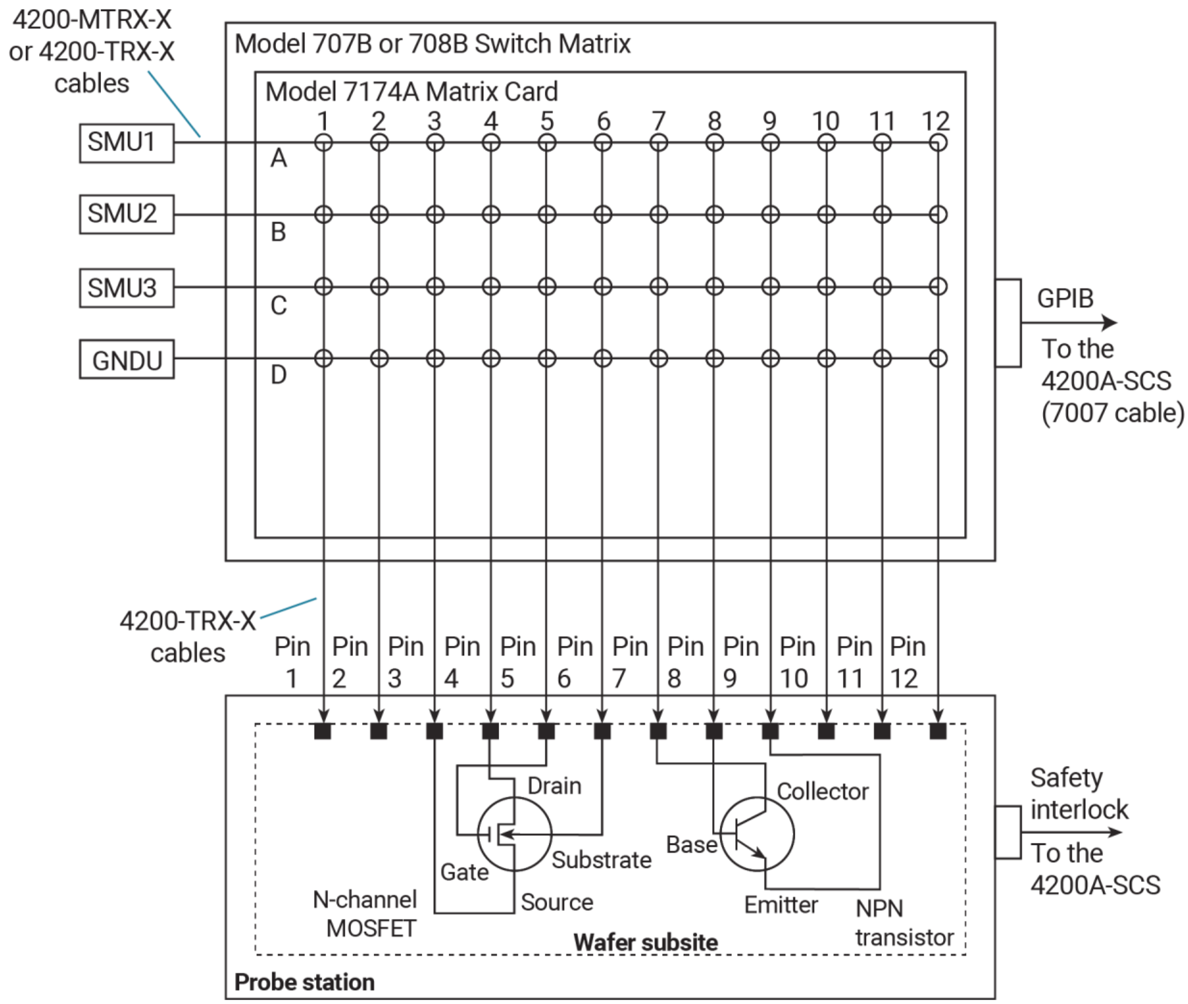
Figure 122: Sample wafer organization



Test system connections

A typical test system for this tutorial is shown in the following figure. The 4200A-SCS and probe station are connected to a 7174A matrix card. The matrix card is installed in the switching matrix, and the switching matrix and probe station are controlled through the GPIB bus.

Figure 123: System configuration for the probesubsites project



KCon setup

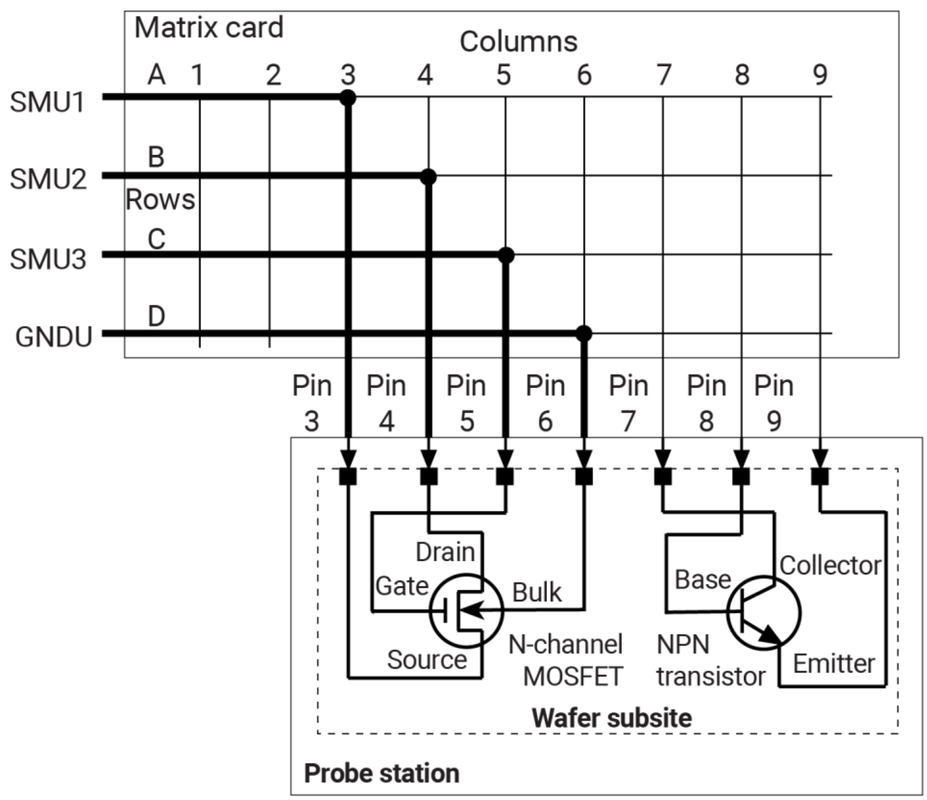
Refer to Use KCon to add a switching matrix to the system (on page 2-26).

Test flow

When you run the `probesubsites` project, the following occurs:

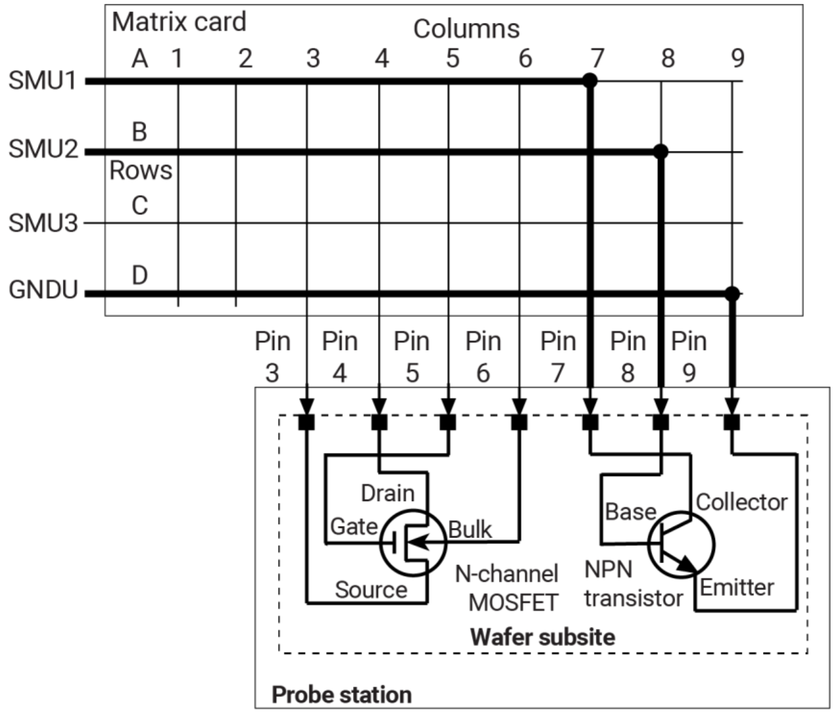
1. The action `prober-init` initializes the prober driver.
2. The test moves to `subsite1`, `4terminal-n-fet`.
3. The action `prober-contact` moves the chuck to the wafer.
4. The action `connect` connects the SMUs to the probes for the n-channel MOSFET as shown in the following figure.

Figure 124: Connect SMUs to N-channel MOSFET



5. The test runs vds-id-1x, which generates a family of curves (I_D vs. V_D) for the MOSFET.
6. The test moves to `3terminal-npn-bjt`.
7. The action `connect` connects the SMUs to the probes for the npn transistor as shown in the following figure.

Figure 125: Connect SMUs to NPN transistor



8. The test runs `vce-ic-1x`, which generates a collector family of curves (I_C vs. V_C) for the transistor.
9. The action `prober-ss-move` moves the prober to the next subsite.
10. The tests continue with `subsite2` and `subsite3`.
11. After all the subsites have run, the action `prober-separate` separates the prober pins from the wafer.
12. The action `prober-prompt` displays the message “Wafer Test Complete” at the end of the test.

Section 9

Using a Cascade Microtech PA200 Prober

In this section:

Cascade Microtech PA200 prober software

To configure and operate the Cascade Microtech (Karl Suss) PA200 prober with the Keithley 4200A-SCS, you need the following applications:

- **ProberBench NT:** Provides easy access to configuration and help programs.
- **Wafer Map:** Use to configure wafer geometry, set origin, set home, select dies to probe, and align the wafer.
- **Chuck Navigator:** Use to move the chuck and select subsites.
- **PB-GPIB:** Use to configure the GPIB interface.
- **PB-RS-232:** Use to configure the serial interface.
- **Prober Setup (in the service programs folder):** Use to initialize the serial communications port.

Software versions

The following list contains the software versions used to verify the configuration of the PA200 prober with the 4200A-SCS.

Product Name:	WaferMap for ProberBench NT
Product Version:	3.1 (Feb 12, 1999)
Copyright:	© Karl Suss 1998 - All Rights Reserved
Kernel:	3.000000 ProberBench Kernel Version 3.10 12-7-98
Control Box:	2.400000 ProberBench Control Box 2.4

Product Name:	NI-GPIB for ProberBench NT
Product Version:	3.10 (Feb 12, 1999)
Copyright:	© Karl Suss 1998 - All Rights Reserved
Kernel:	3.000000 ProberBench Kernel Version 3.10 12-7-98
Control Box:	2.400000 ProberBench Control Box 2.4

Product Name:	PBRS232 Interface for ProberBench NT
Product Version:	3.00
Copyright:	© Karl Suss 1998 - All Rights Reserved
Kernel:	3.000000 ProberBench Kernel Version 3.10 12-7-98
Control Box:	2.400000 ProberBench Control Box 2.4

Product Name:	Navigator for ProberBench NT
Product Version:	3.1 (Feb 12, 1999)
Copyright:	© Karl Suss 1998 - All Rights Reserved
Kernel:	3.000000 ProberBench Kernel Version 3.10 12-7-98
Control Box:	2.400000 ProberBench Control Box 2.4

Product Name:	TableView for ProberBench NT
Product Version:	3.1 (Feb 12, 1999)
Copyright:	© Karl Suss 1998 - All Rights Reserved
Kernel:	3.000000 ProberBench Kernel Version 3.10 12-7-98
Control Box:	2.400000 ProberBench Control Box 2.4

Product Name:	Remote Communicator for ProberBench NT
Product Version:	3.00
Copyright:	© Karl Suss 1998 - All Rights Reserved
Kernel:	3.000000 ProberBench Kernel Version 3.10 12-7-98
Control Box:	2.400000 ProberBench Control Box 2.4

Probe station configuration

CAUTION

Make sure that you are familiar with the Cascade MicroTech® PA200 Prober and its supporting documentation before you attempt setup, configuration, or operation.

To set up and configure the PA200 prober for use with the 4200A-SCS, you will:

- Set up communications (on page 9-8)
- Set up wafer geometry (on page 9-11)
- Create a site definition and define a probe list (on page 9-13)
- Load, align, and contact the wafer (on page 9-14)

Set up communications

You need to set communications between the 4200A-SCS and the prober.

Make connections between the 4200A-SCS and the prober

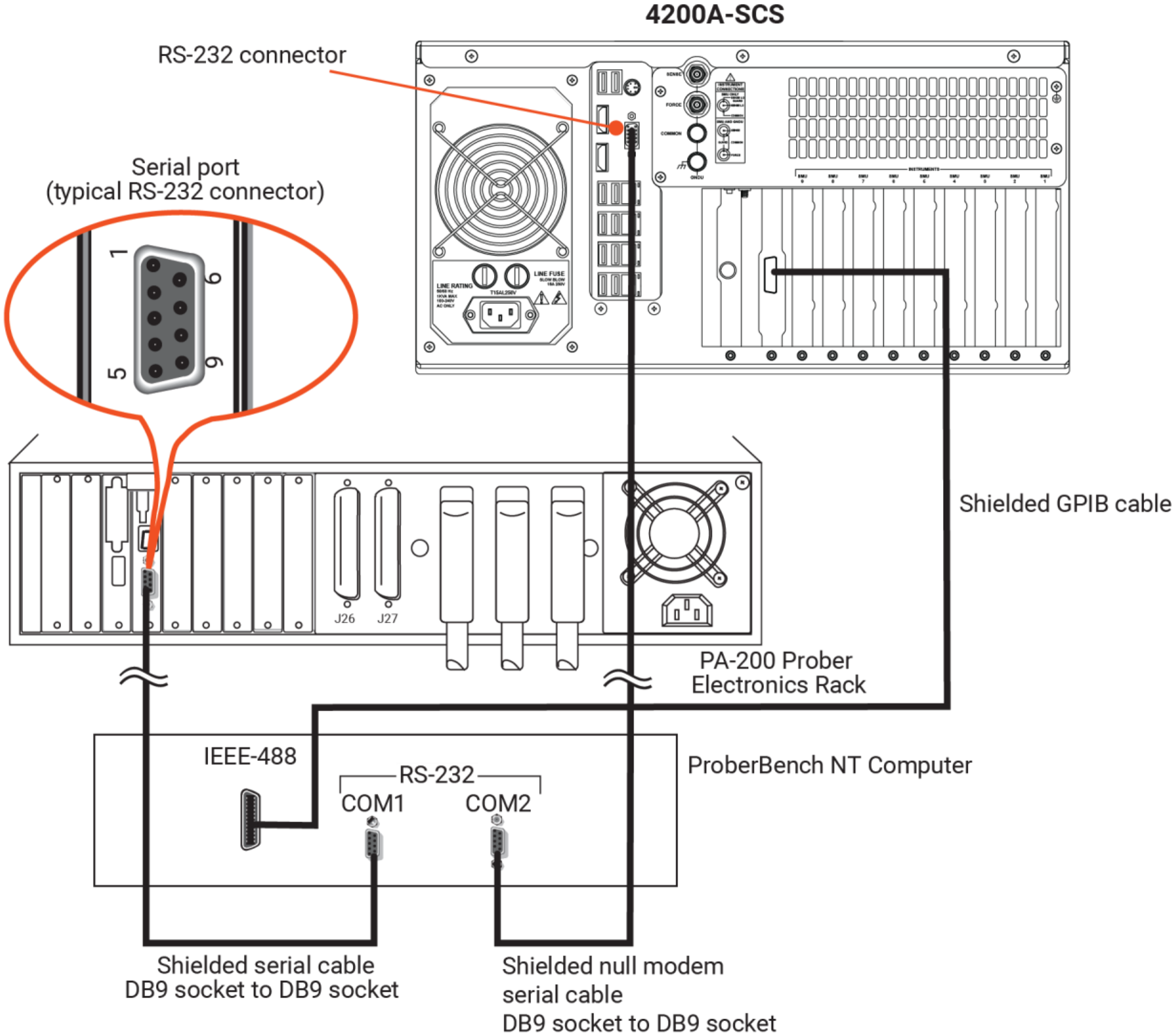
To make the connections:

1. Connect the COM2 port of the ProberBench NT computer to the 4200A-SCS COM1 port using a DB9 socket to DB9 socket cable (shielded null modem cable). Refer to the following figure.
2. Connect the ProberBench NT computer serial port (COM1) to the PA200 Prober Electronics Rack serial port.
3. Connect the 4200A-SCS GPIB port and the GPIB port of the ProberBench NT computer using a shielded GPIB cable. Refer to the following two graphics and table for a connection diagram, connector diagram, and connector pinout definitions.

NOTE

Do not use the GPIB port on the Prober Electronics Rack. Make sure to connect the cable between the GPIB ports on the 4200A-SCS and the ProberBench NT computer as shown.

Figure 126: 4200A-SCS and PA-200 serial port connection



Shielded serial cable DB9 socket to DB9 socket

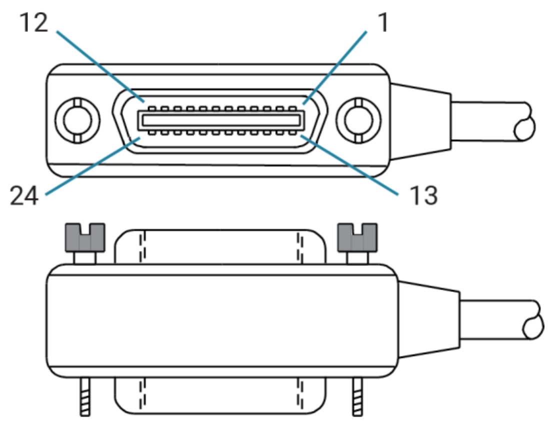
Prober electronics rack	Prober Bench NT computer
1	4
2	3
4	1, 6, 8
5	5
6, 8	4

Prober Bench NT computer	4200A-SCS
1	4
2	3
3	2
4	1, 6
5	5
6	4
7	8
8	7
9	N.C.

GPIB control connector terminals

The contact numbers are shown in the following figure. The GPIB designation and type are shown in the following table.

Figure 127: IEEE-488 connector contact numbers



GPIB control connector terminals

Contact number	GPIB designation	Type
1	DI01	Data
2	DI02	Data
3	DI03	Data
4	DI04	Data
5	EOI (24)*	Management
6	DAV	Handshake
7	NRFD	Handshake
8	NDAC	Handshake
9	IFC	Management
10	SRQ	Management
11	ATN	Management
12	SHIELD	Ground
13	DI05	Data
14	DI06	Data
15	DI07	Data
16	DI08	Data
17	REN (24)*	Management
18	Gnd (6) *	Ground
19	Gnd (7) *	Ground
20	Gnd (8) *	Ground
21	Gnd (9) *	Ground
22	Gnd (10) *	Ground
23	Gnd (11) *	Ground
24	Gnd, LOGIC	Ground

Set up communications on the 4200A-SCS

On the 4200A-SCS, you need to set the communications through the prober configuration file.

The configuration file for use with serial communications is shown in the following code example. To configure the prober for use with a GPIB communications setup, use a text editor to comment out (`#`) the lines after "Configuration for PA200 probers" and activate the lines (remove the `#`) after "Configuration for direct GPIB probers." Be sure to only activate the lines that start with `PROBER_1_....`

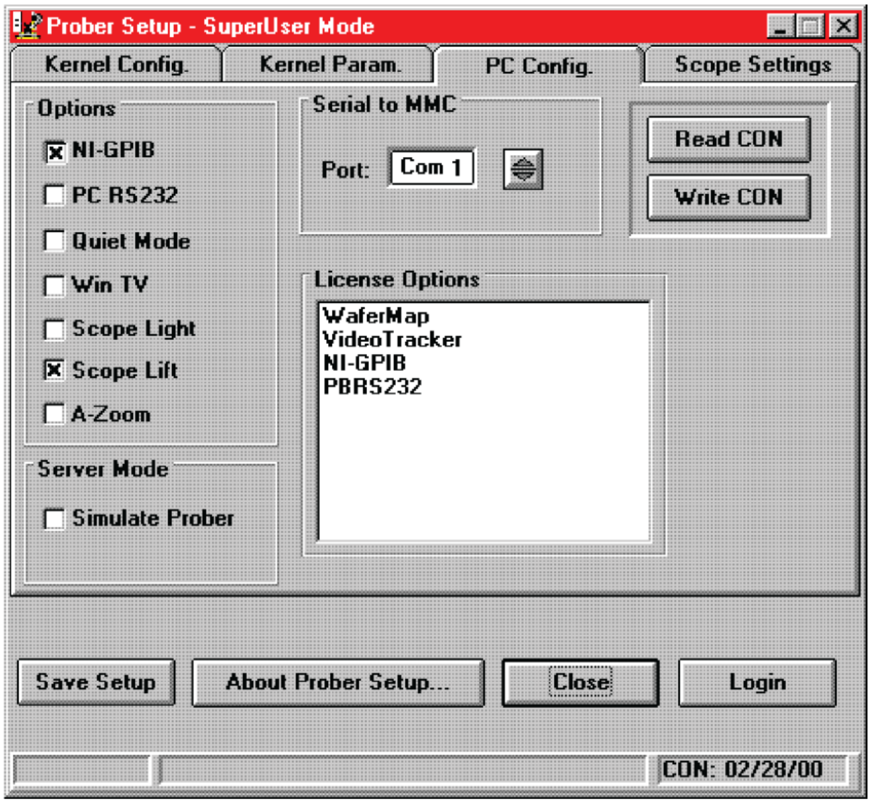
Configuration file location: `C:\s4200\sys\dat\prbcnfg_PA200.dat`

```
# prbcnfg_PA200.dat  - DEFAULT Prober Configuration File
#
# The following tag, "PRBCNFG", is used by the engine in order to determine
# the MAX number of SLOTS and CASSETTES for a given prober at runtime.
#
<PRBCNFG>
#
# for OPTIONS "" == NULL, max 32 chars in string
#
# Example
#           01234567890
#PROBER_1_OPTIONS=1,1,1,1,1,1
#
#
#   OcrPresent
#   AutoAlnPresent
#   ProfilerPresent
#   HotchuckPresent
#   HandlerPresent
#   Probe2PadPresent
#
#
# Configuration for PA200 probers:
#  PA200
#
PROBER_1_PROBTYPE=PA200
PROBER_1_OPTIONS=0,0,0,0,1,0
PROBER_1_IO_MODE=SERIAL
PROBER_1_DEVICE_NAME=COM1
PROBER_1_BAUDRATE=9600
PROBER_1_TIMEOUT=300
PROBER_1_SHORT_TIMEOUT=5
PROBER_1_MAX_SLOT=25
PROBER_1_MAX_CASSETTE=1
#
#
#
# Configuration for direct GPIB probers:
#  PA200
#
#PROBER_1_PROBTYPE=PA200
#PROBER_1_OPTIONS=0,0,0,0,1,0
#PROBER_1_IO_MODE=GPIB
#PROBER_1_GPIB_UNIT=0
#PROBER_1_GPIB_SLOT=1
#PROBER_1_GPIB_ADDRESS=8
#PROBER_1_GPIB_WRITEMODE=0
#PROBER_1_GPIB_READMODE=2
#PROBER_1_GPIB_TERMINATOR=13
#PROBER_1_TIMEOUT=300
#PROBER_1_SHORT_TIMEOUT=5
#PROBER_1_MAX_SLOT=25
#PROBER_1_MAX_CASSETTE=1
#
#
```

Set up communications on the prober

You can configure the PA-200 prober for serial or GPIB communications. Ensure that the prober is set up for the type of communications interface that is defined on the 4200A-SCS (see Set up communications on the 4200A-SCS (on page 9-6)).

Figure 128: Prober setup: PC Config tab



Set up serial communications

To set up communications for RS232:

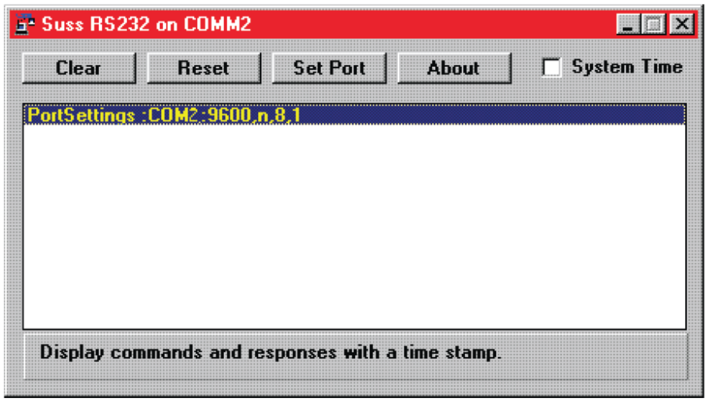
1. On the prober computer, double-click the **ProberBench NT** icon.
2. Double-click the **Service Programs** file.
3. Double-click the **Prober Setup** file in the Service Programs directory.
4. Select the PC **RS232** option.
5. Clear the **Simulate Prober** box in the Server Mode section of the dialog.
6. Select **Save Setup**.
7. Select **Close**.
8. From the ProberBench NT window, double-click the **PBRS232** file.

NOTE

COM2 is used for communications between the 4200A-SCS and the ProberBench NT. COM1 is used for communications between the ProberBench NT and the electronics rack.

9. From the Suss RS232 on COMM2 dialog, select **Set Port**.

Figure 129: Suss RS232 on COMM2 dialog



10. Set communications protocol to 9600, n, 8, 1 for serial port COM2.
11. Make sure Disable COM Port is not selected.

NOTE

Leave the **Suss RS232 on COMM2** dialog open. This ensures its services are available for the WaferMap program.

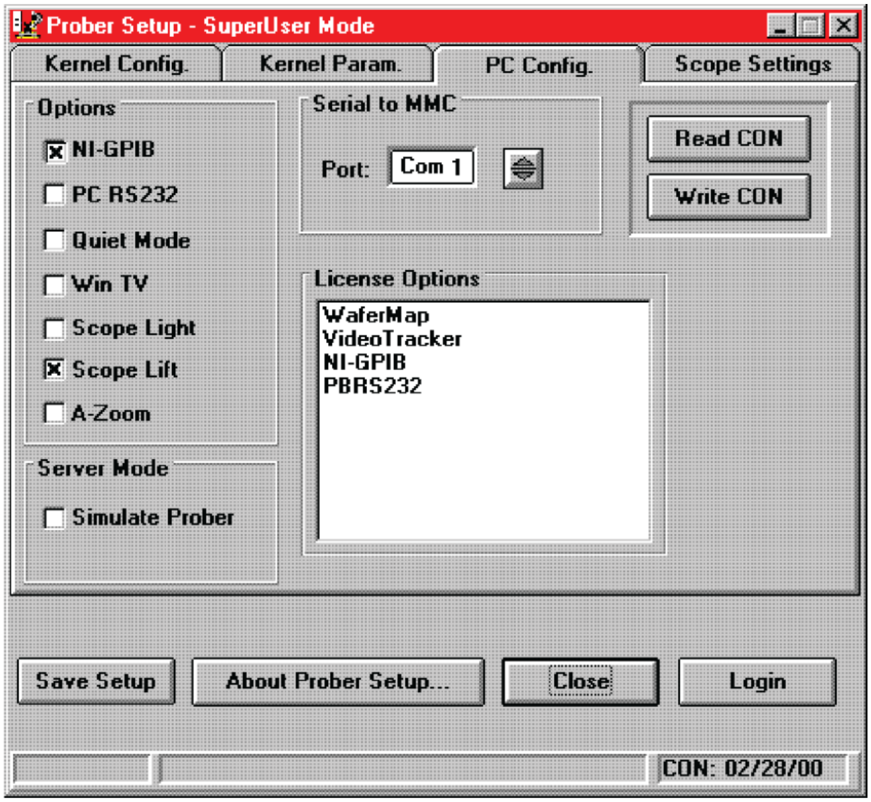
12. Select **Save** and **Exit**.
13. From the **Suss RS232 on COMM2** dialog, select **Reset**.

Set up GPIB communications

To set up GPIB communications:

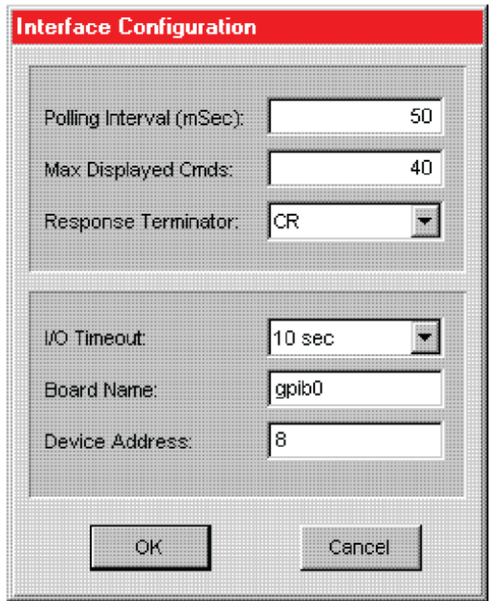
1. Double-click the ProberBench NT icon (shortcut) on desktop.
2. Double-click the Service Programs file.
3. Double-click the Prober Setup file in the Service Programs directory. The Prober Setup dialog is displayed.

Figure 130: Prober setup: PC Config tab



4. Select the **NI-GPIB** option.
5. Clear the **Simulate Prober** box in the Server Mode section of the dialog.
6. Select **Save Setup**.
7. From the ProberBench NT window, double-click the **PB-GPIB** file.
8. From the ProberBench GPIB Interface, from the Configure menu, select **Interface Driver.**
9. From the Interface Configuration dialog, change Response Terminator to **CR**.

Figure 131: Interface Configuration dialog



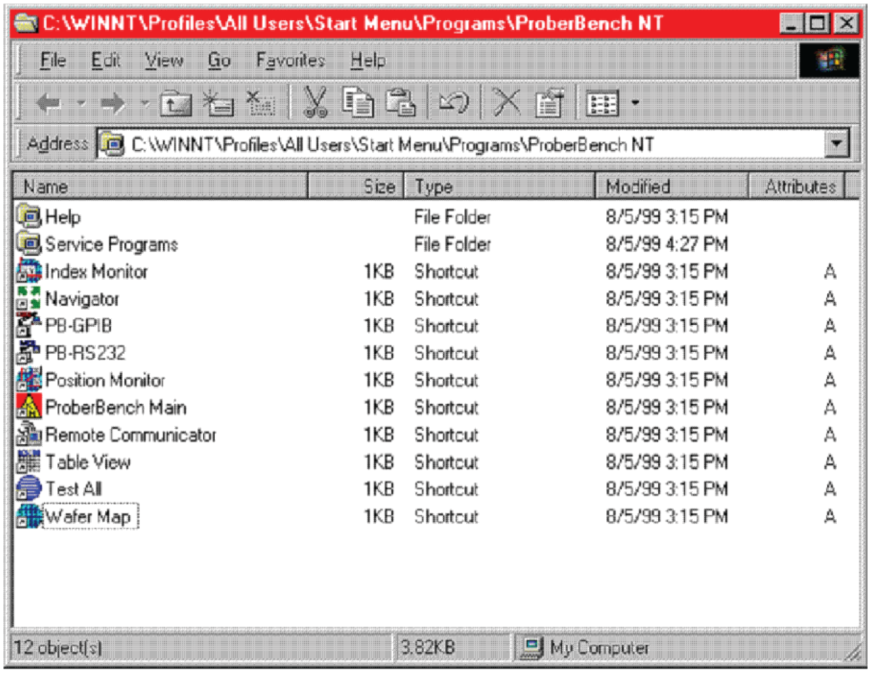
10. GPIB only: Ensure that the GPIB address matches the address in the configuration file.
11. Select **OK**.

Set up wafer geometry

On the ProberBench NT computer:

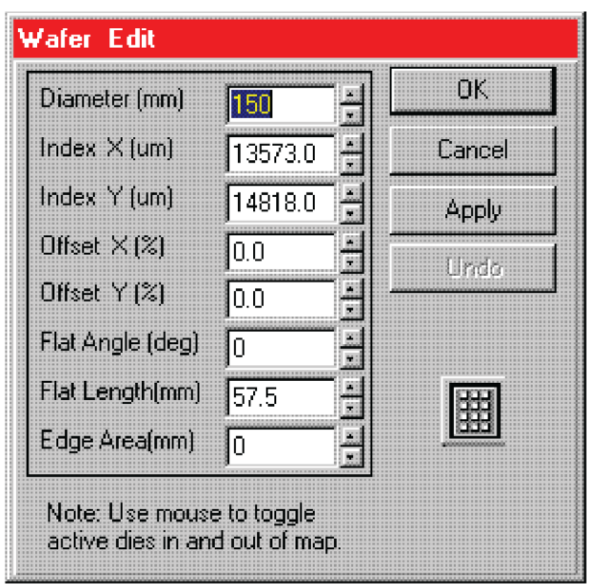
1. Select the ProberBench **NT** icon (shortcut) on the desktop.
2. From the ProberBench NT window, select the **Wafer Map** file.

Figure 132: ProberBench NT window



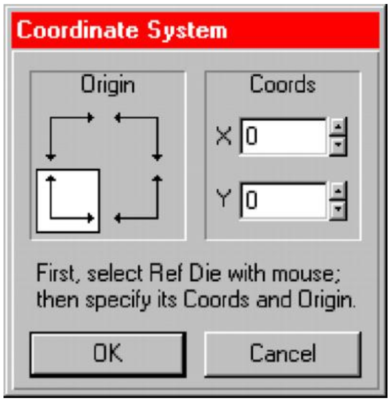
3. From the WaferMap dialog, create or open a WaferMap.
4. From the Configure menu, select **Edit Map**.
5. Enter the wafer geometry values and select **Apply**.
6. Select **OK**.

Figure 133: Wafer Edit dialog



7. From the Configure menu, select **Coordinates**.
8. From the Coordinate System dialog, set **Origin**, as shown. You can set any initial X and Y coordinates.

Figure 134: Coordinate System dialog



9. Select **OK**.

NOTE

Refer to Clarius `probesites` and `probesubsites` examples for specifics on selecting sites to probe.

10. Select **File > Project > Save** to save the WaferMap settings.

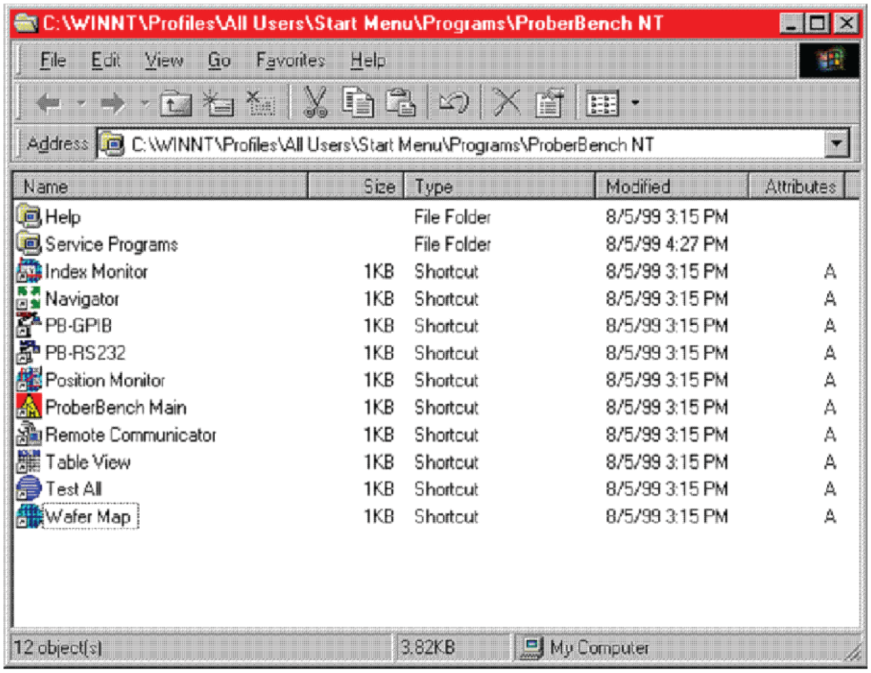
Create a site definition and define a probe list

Creating a site definition for single subsites for each die involves using the software to create a selection of dies to probe. If a single subsite per die is to be probed, refer to Probesites Clarius project example (on page 10-18). Creating a site definition for multiple subsites for each die involves using the software to create a selection of dies to probe, but also includes creating a selection of the subsites on each die that will be probed. If multiple subsites for each die will be probed, refer to Probesubsites Clarius project example (on page 13-20).

To load a previously defined and saved site definition and a probe list:

1. Select the ProberBench **NT** icon on the desktop.
2. From the ProberBench NT window, select the **WaferMap** file.

Figure 135: ProberBench NT window



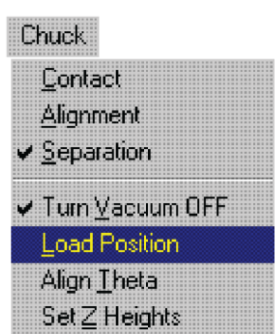
3. From the WaferMap window, select and open the appropriate file.

Load, align, and contact the wafer

Using the ProberBench NT computer:

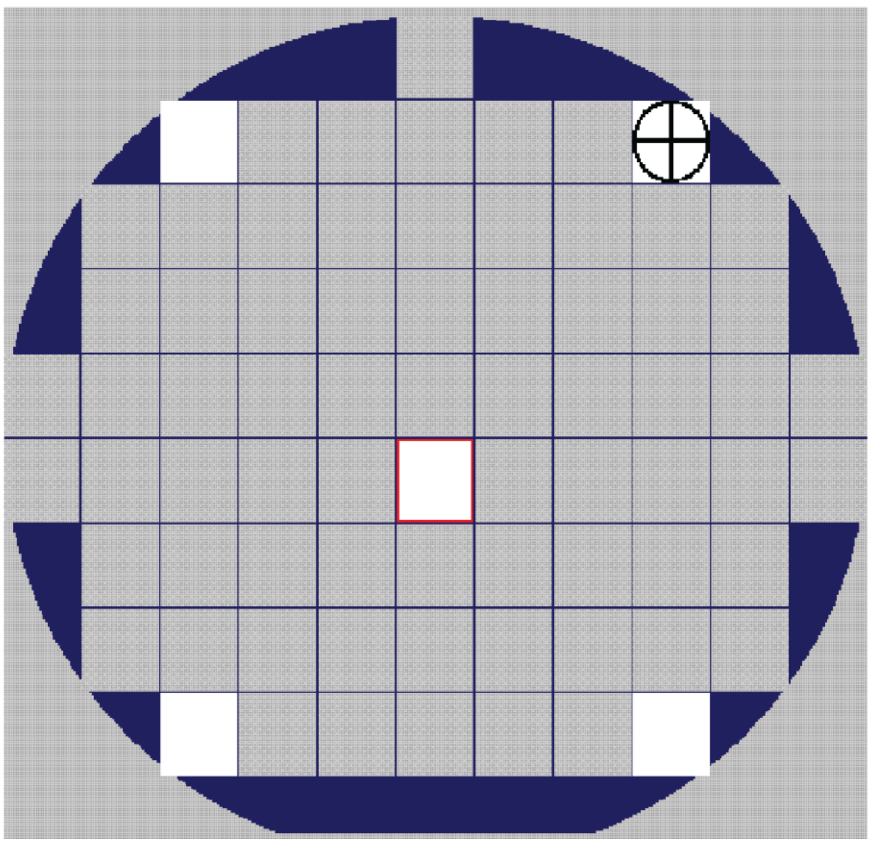
1. From the WaferMap Chuck menu, select **Load Position**. This brings the chuck to the front of the prober.

Figure 136: Chuck menu



2. Place the wafer on the chuck.
3. From the Chuck menu, select **Turn Vacuum ON**.
4. Manually move the wafer to the Home Die.
5. From the Setup menu, select **Home Die**.
6. Choose the home die on the WaferMap. When choosing the home die:
   - The wafer should be on the chuck and physically in the correct HOME position.
   - Select the die on the wafer map UI that will be the home die.
   - A crosshair appears when a die has been selected as the home die.

Figure 137: WaferMap home die selection

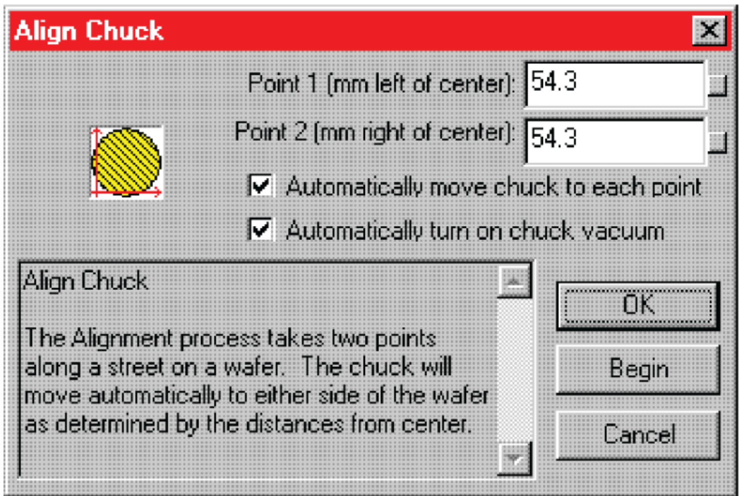


7. From the Chuck menu, select **Align Theta**.
8. Align wafer using the following steps.

Aligning the wafer

1. Enter Point 1 and Point 2 distances from the center using specific X die size multiples. In other words, if the die size is: X = 13.573 mm, and Y = 14.818 mm, set up to move four die to the left and also the right at 54.292 mm (4 • 13.573 mm = 54.292 mm).

Figure 138: Align the chuck

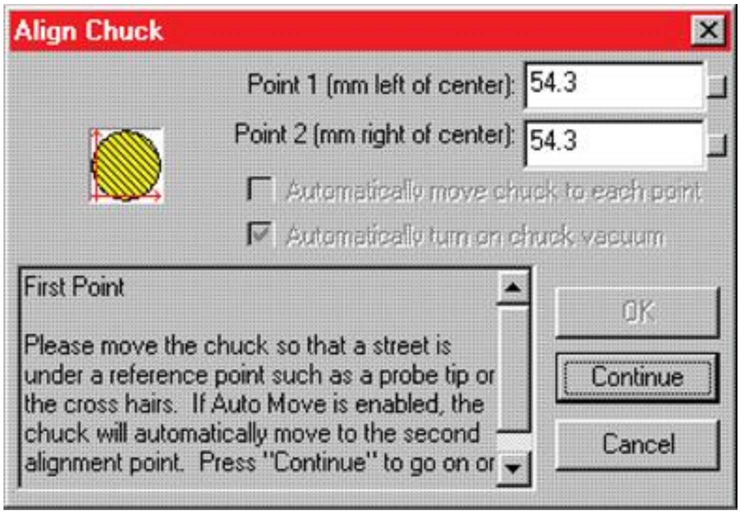


2. Select **Automatically move chuck to each point**.
3. Select **Automatically turn on chuck vacuum**.
4. Select **Begin**.

Start the Alignment Wizard

1. Move to Point 1 (left of center die align pad and pins).
2. Select **Continue** to start the Alignment Wizard.
3. Manually align pins and pads (POINT 1).
4. Select **Continue** (from POINT 1) and move 8 die (for this example) to the right.

Figure 139: Aligning the wafer: First point



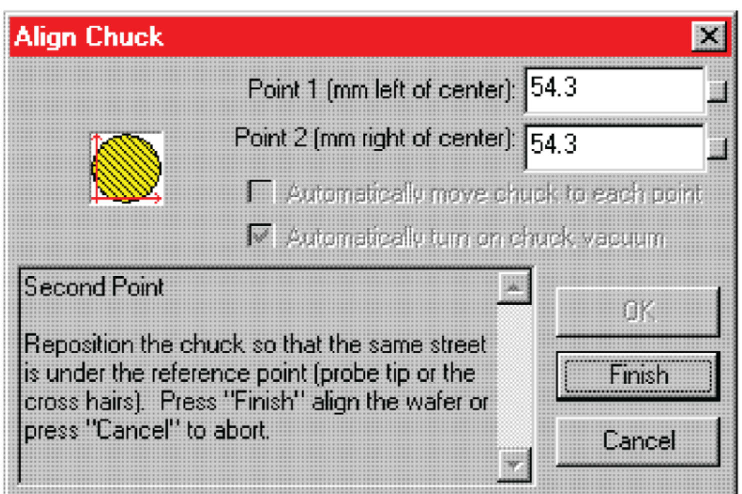
5. Manually align pins and pads (POINT 2) and select **Finish**.

Verify wafer alignment

Confirm that the alignment is correct (the alignment procedure is repeated). To check, manually use the joystick to move the chuck in index moves and confirm that the pins and pads are aligned.

If the alignment is not correct, repeat the alignment. If the alignment is correct, select **Finish**.

Figure 140: Aligning the wafer: Point 2

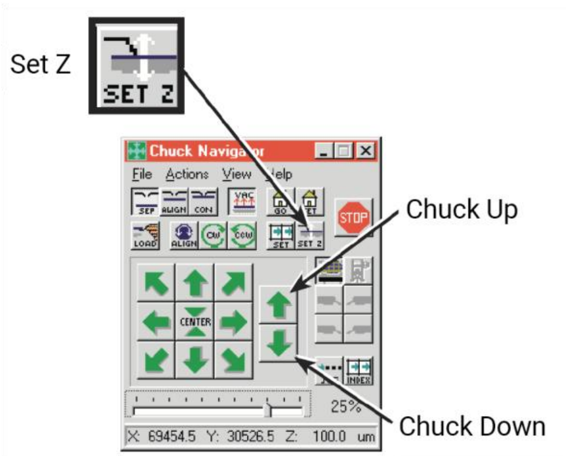


Set the chuck heights

To set the chuck heights:

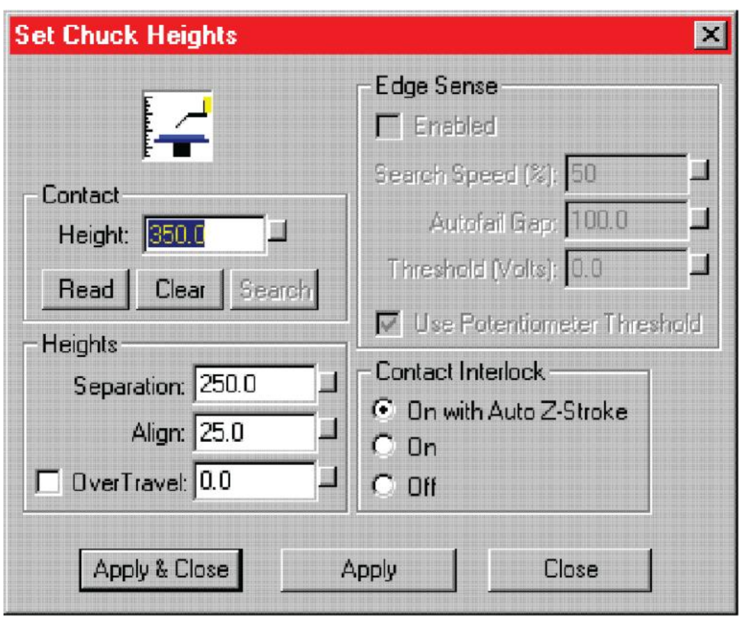
1. Launch the navigator from the ProberBench NT window icon.
2. In the Chuck Navigator dialog, use the chuck up and down arrows to contact the wafer on the home die and home subsite.

Figure 141: Chuck navigator dialog wafer height



3. Select **Set-Z**. The Set Chuck Heights dialog is displayed.

Figure 142: Set chuck heights



4. Select **Read**. The contact height value changes to the present height.
5. Select **Apply**.
6. Select **Close**.
7. Select **File > Project > Save** to save the Chuck Navigator settings.
8. Select **File > Project > Save** to save the WaferMap configuration.

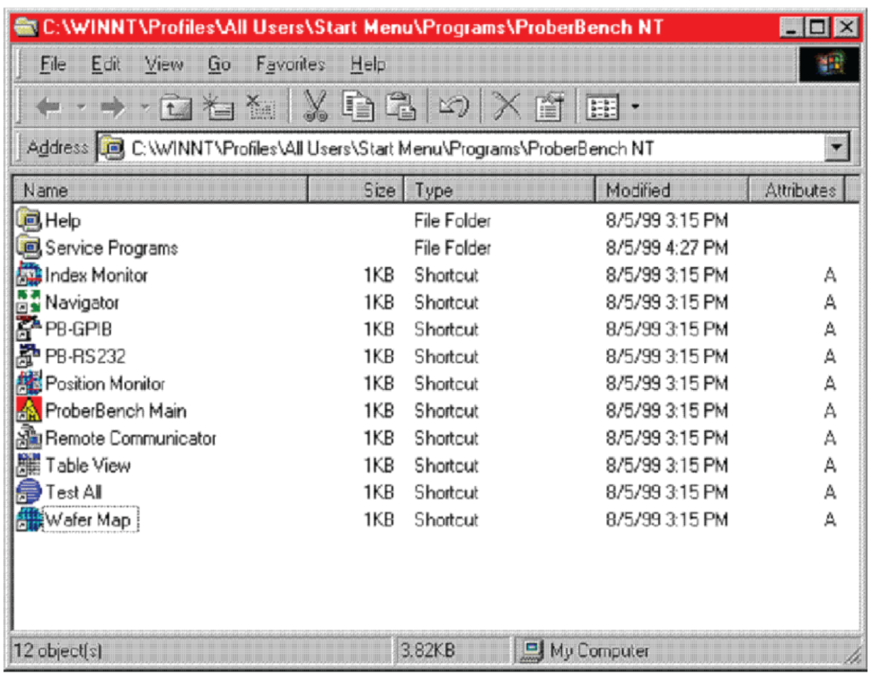
Clarius probesubsites project example

The following is a step-by-step procedure to configure the PA-200 so the `probesubsites` Clarius project executes successfully.

On the ProberBench NT computer:

1. Select the ProberBench NT icon (shortcut) on the desktop.
2. From the ProberBench NT window, select **WaferMap** file.

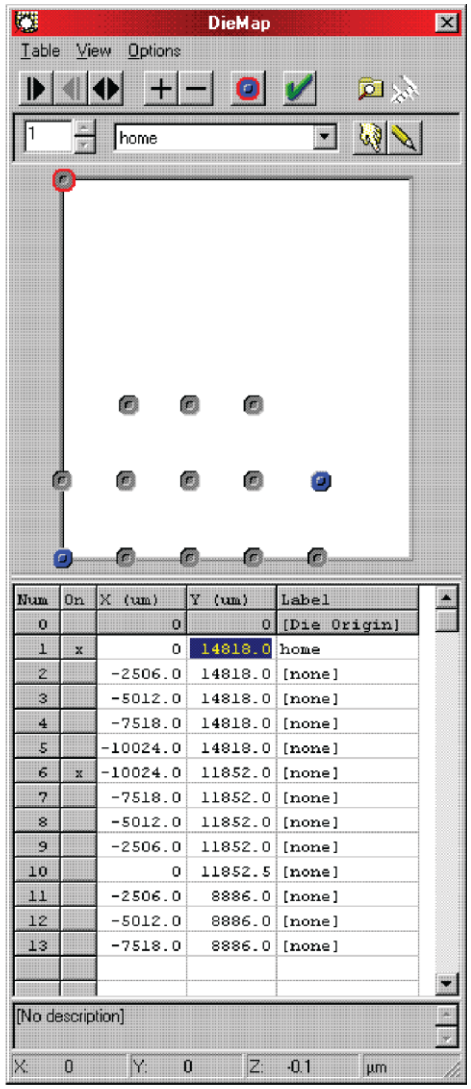
Figure 143: ProberBench NT window



3. From the WaferMap window, from the Mark Dies menu, select **Mark to Skip**.
4. Use Mark to Skip and Mark to Probe to set dies. Select a die in the WaferMap window to either set or clear the die. The color of the die indicates status (either probe or skip). With Mark to Probe selected, drag to select multiple dies. With Mark to Skip selected, drag to clear multiple dies. When done, clear **Mark to Skip** or **Mark to Probe**. Otherwise, the Chuck menu remains grayed.

5. From the View menu, select **Die Map**.
6. Set up the die map.
7. From the View menu, select the **Table editor**. The spreadsheet portion of the Die Map is displayed.

Figure 144: DieMap dialog



8. From the Options menu, select units (Microns or Mils).
9. Edit the table with the coordinates of the subsites.
10. From the Table menu, select **Save** or **Save As**.

NOTE

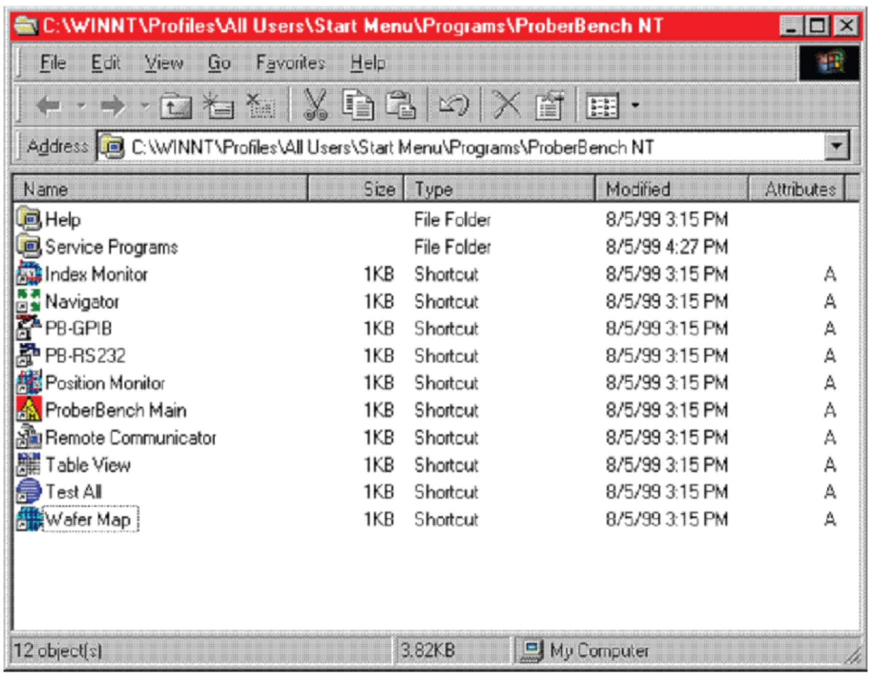
An x in the On column defines the subsites that will be probed when using the subsite probing project (in other words, when using **PrSSMovNxt**). Other subsites may be defined in the list.

Set the wafer map

On the ProberBench NT computer:

1. Select the **ProberBench NT** icon on the desktop.
2. From the ProberBench NT window, select the **Wafer Map** file.

Figure 145: ProberBench NT window



3. From the Mark Dies menu, use **Mark to Skip** and **Mark to Probe** to set dies. Select a die in the WaferMap window to either set or clear the die. The color of the die indicates status (probes white dies, skips blue dies).

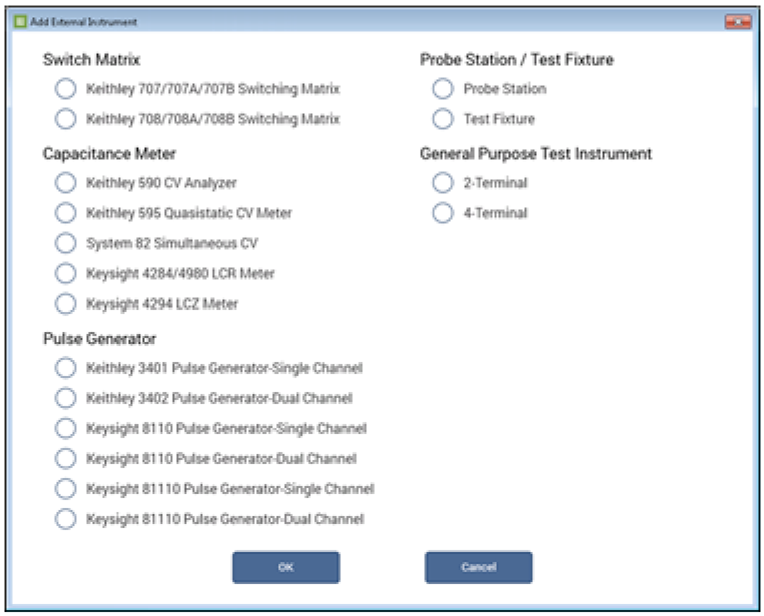
   With **Mark to Probe** selected, drag to select multiple dies. With **Mark to Skip** selected, drag to clear multiple dies. When done, clear **Mark to Skip** or **Mark to Probe**. Otherwise, the Chuck menu remains grayed.
4. Select **File > Project> Save** to save the WaferMap configuration.

Use KCon to add a prober

On the 4200A-SCS, use KCon to add the prober to the configuration:

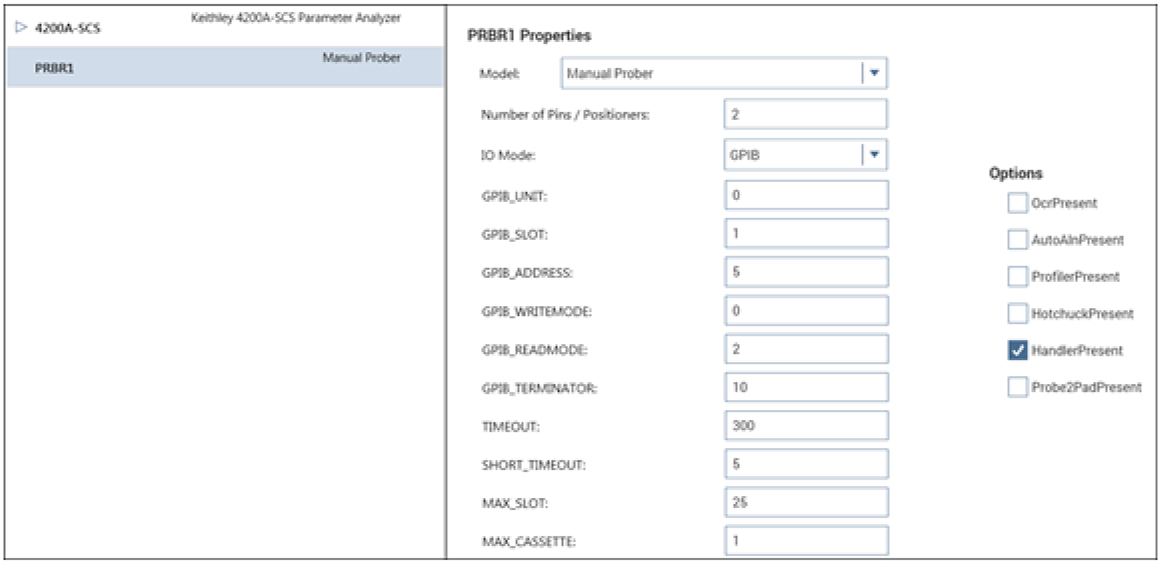
1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

Figure 146: Add a prober in KCon



3. Select **Probe Station**.
4. Select **OK**. KCon displays the properties for the prober.

Figure 147: Use KCon to select a prober



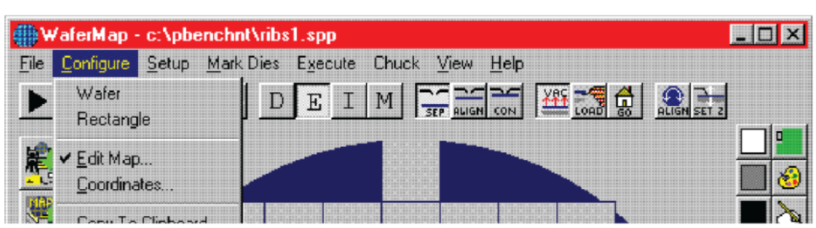
5. Select the **Karl Suss PA200 Prober** as the model.
6. Ensure that the **Number of Pins / Positioners** is correct. The number of pins defined here determines the pins that are available to assign to a switching matrix card column.
7. Select **Save**.
8. Exit KCon.

Running projects

On the ProberBench NT computer:

1. After the wafer is set up and alignment is complete, select the **File > Project > Save**.
2. Select **File > Map > Save**.
3. Select **File > Table > Save**. The wafer is ready to probe.
4. Place the prober in Run mode.
5. Ensure that the "E" in the **WaferMap** toolbar is selected.

Figure 148: WaferMap toolbar



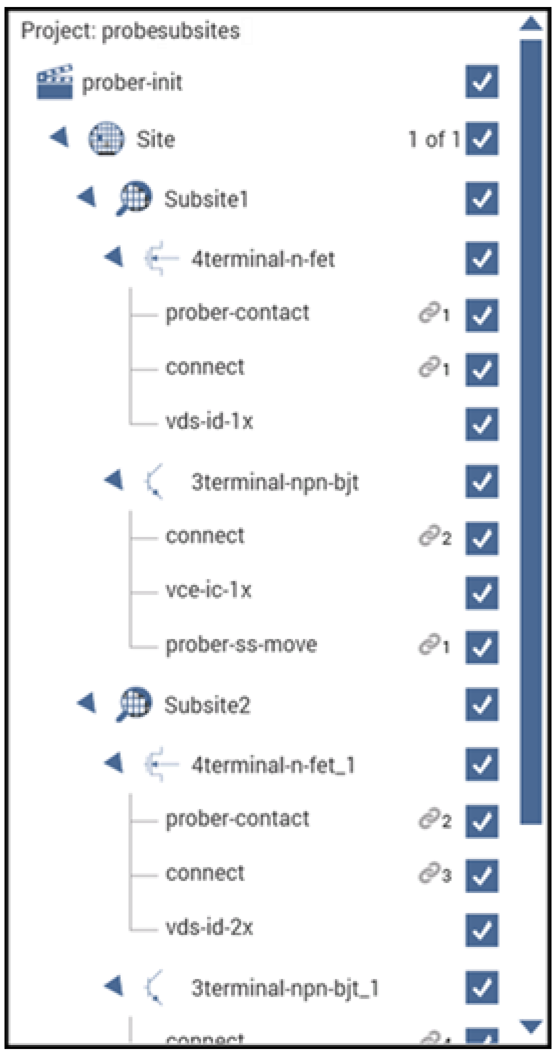
Clarius

Use Clarius to load and run the `probesites` or `probesubsites` project using the new KCon configuration file, which allows you to execute the project for this prober.

On the 4200A-SCS:

1. Open Clarius.
2. Choose **Select**.
3. Select **Projects**.
4. Search for **probesubsites**.
5. Drag the **probesubsites** project to the project tree.

Figure 149: probesubsites project tree



6. Select **Run**.

Commands and error symbols

The following table contains error and status symbols listed by command.

Available commands and responses

	PrChuck	PrInit	PrMovNxt	PrSSMovNxt
PR_OK	X	X	X	X
BAD_CHUCK	X			
INVAL_MODE	X			
UNINTEL_RESP	X	X	X	X
INVAL_PARAM		X		
BAD_MODE		X	X	X
UNEXPE_ERROR		X	X	X
PR_WAFERCOMPLETE			X	X

Information and error code return values and descriptions

Value	Constant	Explanation
1	PR_OK	Success (OK)
4	PR_WAFERCOMPLETE	Next wafer loaded (confirmed)
-1008	INVAL_MODE	Invalid mode
-1011	BAD_MODE	Operation invalid in mode
-1013	UNINTEL_RESP	Unintelligible response
-1015	UNEXPE_ERROR	Unexpected error
-1017	BAD_CHUCK	Bad chuck position
-1027	INVAL_PARAM	Invalid parameter

Using a Micromanipulator 8860 Prober

In this section:

Micromanipulator 8860 prober software

You need to have the following software programs on the Micromanipulator 8860 to configure and operate the 8860 prober with the Keithley 4200A-SCS:

- **pcBridge**: Used to configure the communications setup (icon on the desktop)
- **pcLaunch**: Used to launch various wafer controls and utilities (icon on the desktop)
- **pcIndie**: Used to probe multi-subsites per die (button in pcLaunch window)
- **pcWafer**: Used to probe single subsites per die (button in pcLaunch window)
- **pcNav**
- **pcRouter**

NOTE

pcIndie and pcWafer, which are not included with standard prober software, are required. Refer to the prober manufacturer, Micromanipulator, for availability.

Software versions

The following list contains the software versions used to verify the configuration of the 8860 prober with the 4200A-SCS:

Product Name:	pcBridge
Product Version:	2.0.2
Product Name:	pcIndie
Product Version:	2.0.7

Product Name:	pcLaunch
Product Version:	2.0.9
Product Name:	pcNav
Product Version:	2.0.9
Product Name:	pcWafer
Product Version:	2.0.8
Product Name:	pcRouter
Product Version:	2.0.9

Probe station configuration

CAUTION

Ensure that you are familiar with the Micromanipulator 8860 prober and its supporting documentation before attempting setup, configuration, or operation.

To set up and configure the 8860 prober for use with the 4200A-SCS, you will:

- Set up communications (on page 10-2)
- Set up wafer geometry (on page 10-6)
- Create a site definition and define a probe list (on page 10-8)
- Load, align, and contact the wafer (on page 10-9)

Each step is detailed in the following topics.

Set up communications

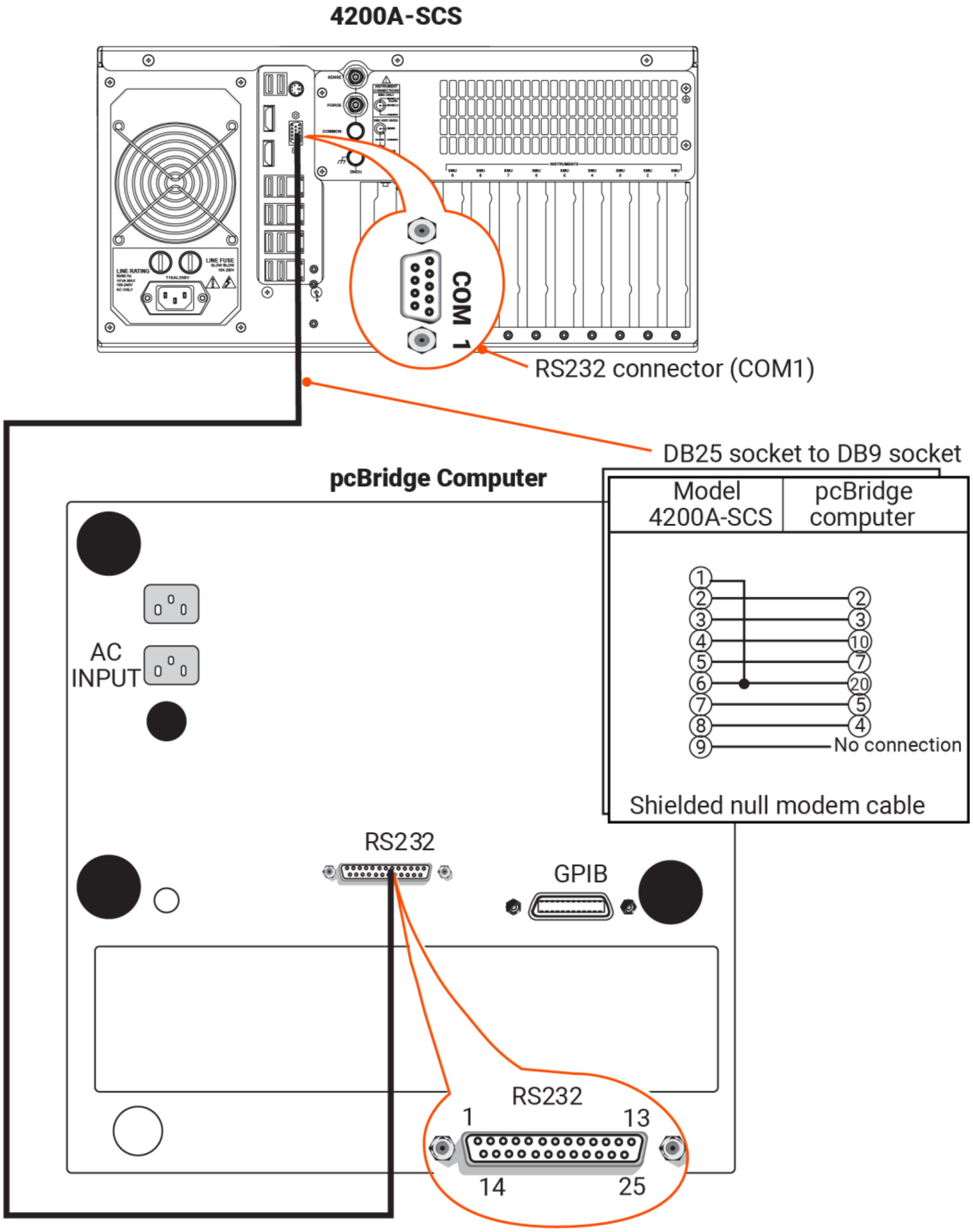
To set up communications:

1. Turn on power to the 4200A-SCS.
2. Turn on power to the prober.
3. Ensure that the vacuum has been properly connected.
4. On the rear panel of the pcBridge computer, connect the RS232 port to the 4200A-SCS COM1 port. Use a DB25 socket to DB9 socket cable (shielded null modem cable). See the following figure for details.

NOTE

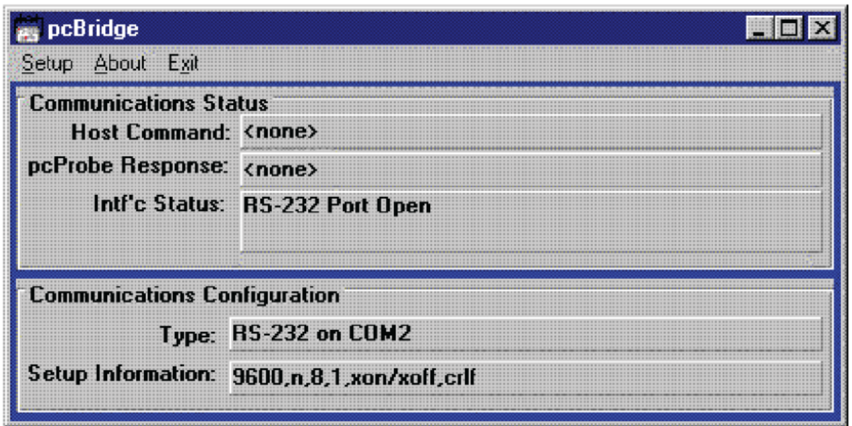
Mouse and keyboard connections are on the front of the pcBridge computer.

Figure 150: Prober setup: Serial connections



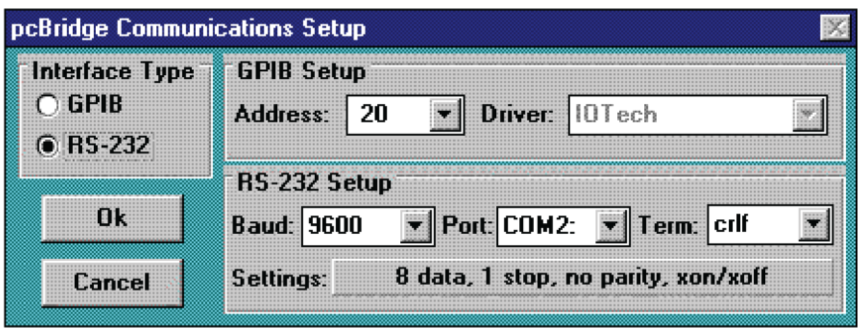
5. Double-click the **pcBridge** icon on the desktop to open the main pcBridge window.

Figure 151: Main pcBridge window



6. Select the **Setup** menu to open the pcBridge Communications Setup window.

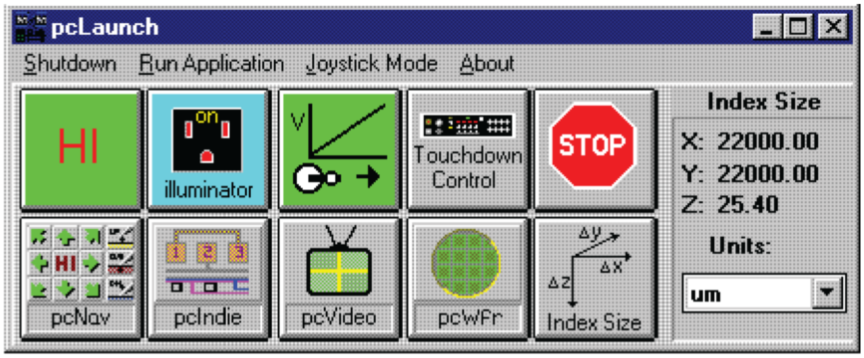
Figure 152: pcBridge Communications Setup window



7. Use the pcBridge Communications Setup to configure the communications settings. These settings should be 8 data, 1 stop, no parity, xon/xoff.
   - Interface Type: RS232
   - Baud: 9600
   - Port: COM2
   - Term: crlf (termination character of carriage-return and line-feed)
8. Select **OK**.

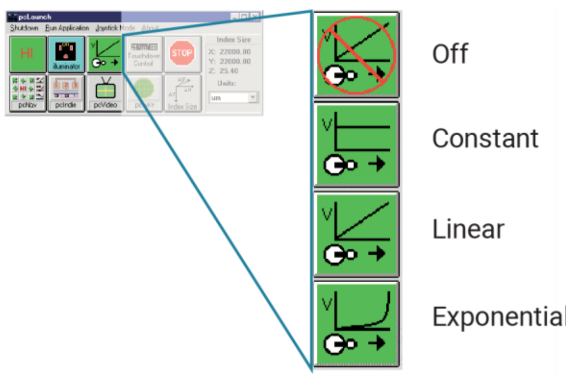
9. Select the **pcLaunch** icon to open the main pcLaunch window.

Figure 153: pcLaunch window



10. From the pcLaunch window, set the Joystick Mode for Linear.

Figure 154: Joystick modes



Modify the prober configuration file

The default prober configuration file is shown in the following code example. As shown, the file is configured for use with serial communications.

Configuration file location: `C:\s4200\sys\dat\prbcnfg_MM40.dat`

Use the 4200A-SCS to modify the file if needed.

```
# prbcnfg.dat - EXAMPLE Prober Configuration File for MM40 Prober
#
# The following tag, "PRBCNFG", is used by the engine in order to determine
# the MAX number of SLOTS and CASSETTES for a given prober at runtime.
#
<PRBCNFG>
#
# for OPTIONS ""== NULL, max 32 chars in string
#
# Example
#           01234567890
#PROBER_1_OPTIONS=1,1,1,1,1,1
#
#
#   OcrPresent
#   AutoAlnPresent
#   ProfilerPresent
#   HotchuckPresent
#   HandlerPresent
#   Probe2PadPresent
#
#
# The PROBER_x_PROBTYPE fields needs to be set to one of the following names.
# Configuration for serial probers:
#
#  Example configuration for MM40 prober
#
#
PROBER_1_PROBTYPE=MM40
PROBER_1_OPTIONS=0,0,0,0,0,0
PROBER_1_IO_MODE=SERIAL
PROBER_1_DEVICE_NAME=COM1
PROBER_1_BAUDRATE=9600
PROBER_1_TIMEOUT=300
PROBER_1_SHORT_TIMEOUT=5
PROBER_1_MAX_SLOT=25
PROBER_1_MAX_CASSETTE=1
#
#
```

Set up wafer geometry

On the pcBridge computer:

1. From the pcLaunch window, select the **pcNav** button to open the pcNav window.

Figure 155: pcNav button

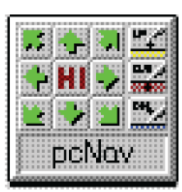
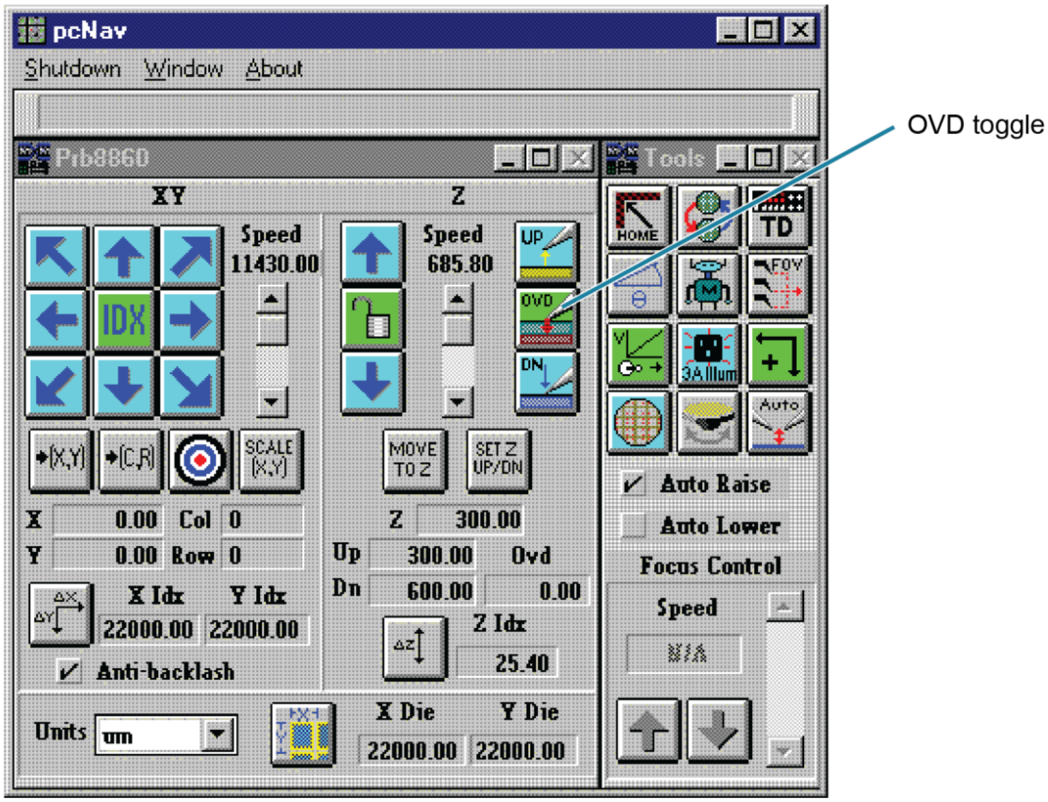


Figure 156: pcNav window



NOTE

When starting pcNav for the first time, the warning “Device Prb8860 must be initialized if positioning accuracy within published specifications is required” is displayed. Select **OK** and continue the configuration (the device is initialized when the chuck is homed).

NOTE

Since the platen moves to make or break contact between the pins and pad, selecting **Auto Raise** automatically separates the pins from the pads. **Auto Lower** allows automatic contact.

2. Select Auto Raise on the pcNav Tools window.
3. Select Anti-backlash on the pcNav Prb8860 window.

Create a site definition and define a probe list

On the pcBridge computer, create a site definition for a single subsite for each die. To do this, use the software to create a selection of dies to probe. If a single subsite for each die is to be probed, refer to Probesites Clarius project example (on page 10-18). Creating a site definition for multiple subsites for each die also uses the software to create a selection of dies to probe and create a selection of the subsites on each die that will be probed. If multiple subsites for each die will be probed, refer to the Probesubsites Clarius project example (on page 10-24).

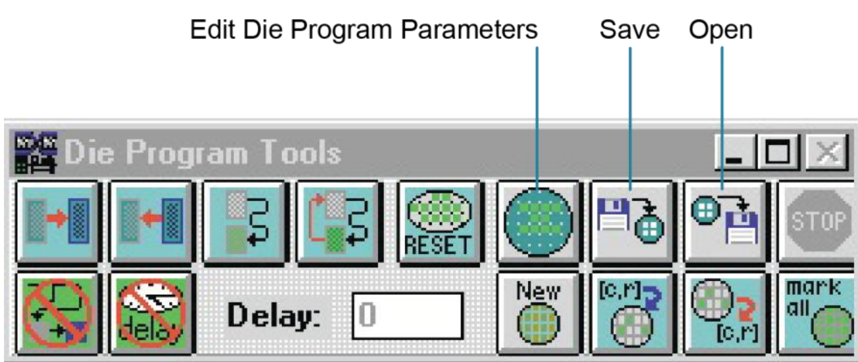
Use the following information to load a previously-defined and saved site definition.

Single subsite per die

To open the file:

1. Start pcWafer by selecting the **pcWfr** button in the pcLaunch window. The pcWfr window is displayed. See the following two figures.

Figure 157: Die Program Tools window



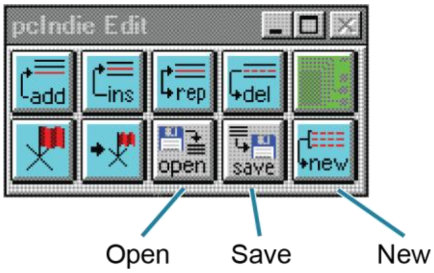
2. Select **Open** on the Die Program Tools window to open an existing file or **New** to create a new file.
3. Select the file and select **OK**.

Multiple subsites per die

To open the file:

1. Select the **pcIndie** button in the pcLaunch window. The pcIndie dialog is displayed.
2. Select the **Open** button on the pcIndie Edit dialog to open an existing file or **New** to create a new file.

Figure 158: pcIndie Edit window



3. Select the file and select **OK**.

Load, align, and contact the wafer

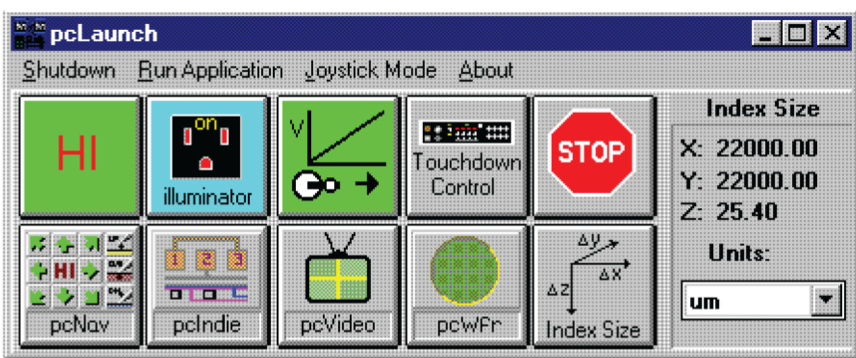
The following topics describe how to contact the wafer.

Home the chuck

To home the chuck:

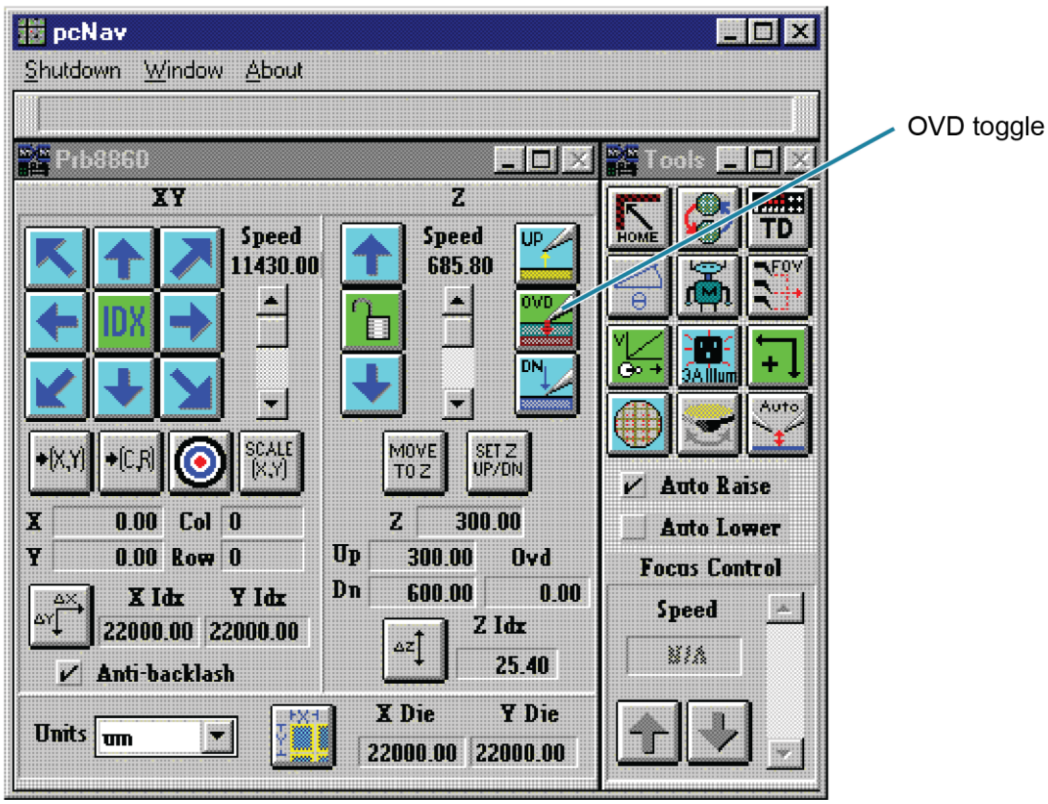
1. On the pcBridge computer, select the pcLaunch icon. The pcLaunch window is displayed.

Figure 159: pcLaunch window



2. To home the chuck, from the pcLaunch window, select the **pcNav** button. The pcNav window opens.

Figure 160: pcNav window

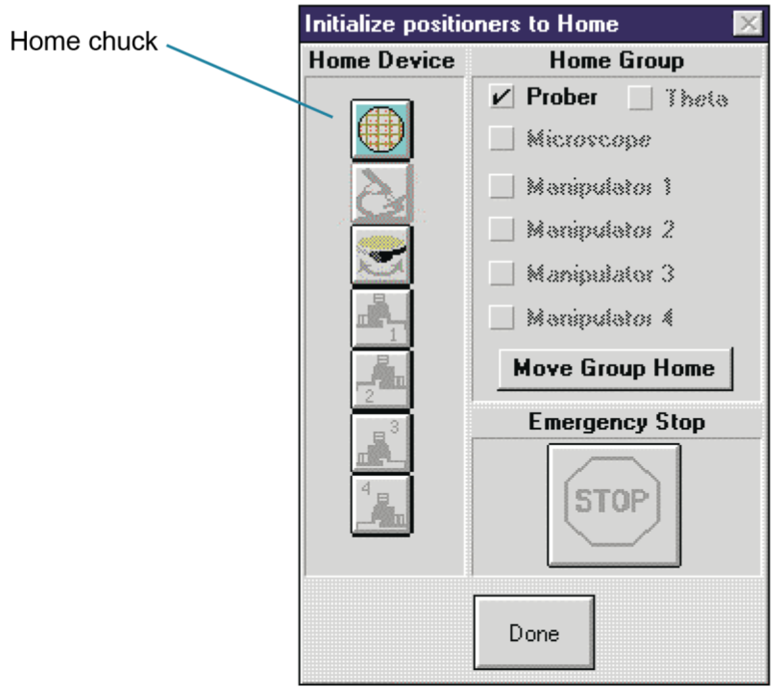


NOTE

The OVD button toggles the state of the overdrive (on or off).

3. Select the **Home** button on the Tools panel of the pcNav window. The Initialize positioners to Home window opens.

Figure 161: Initialize positioners to Home window



4. Select **Home chuck**. The chuck moves to the back left corner and then to the middle.
5. Select **Done** when the chuck is home. The **Done** button turns from grayed to active when the chuck is home.

Load the wafer

Load the wafer:

1. Make sure that the vacuum is off.
2. Select the **Load wafer** button on the Tools panel of the pcNav dialog. The Load Wafer dialog is displayed. See the following two figures.

Figure 162: Load Wafer button

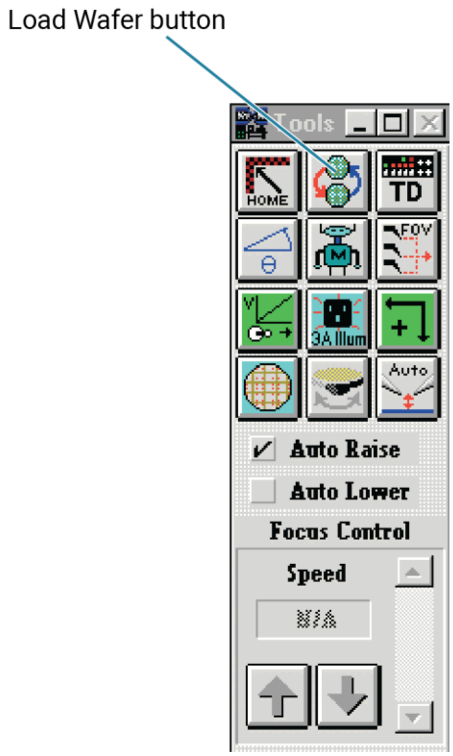
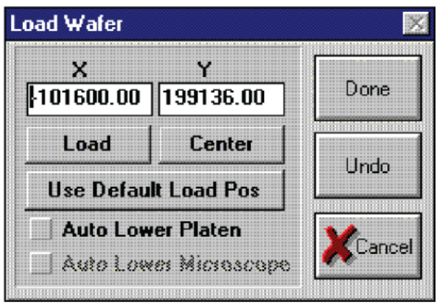


Figure 163: Load Wafer dialog



3. In the Load Wafer dialog, select **Load**.
4. After the chuck moves to the front, place wafer on the chuck aligning the flat or notch in the proper orientation.
5. Apply vacuum.
6. Select **Center**.
7. Select **Done**.

Set the Z-height

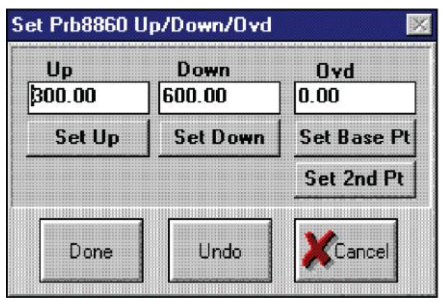
To set the Z-height:

NOTE

This part of the procedure sets Z-height (contact height). The platen moves up and down (Z) while the chuck moves X and Y but not Z. When changing Z-height (moving the platen up or down), a higher number moves closer to contact while a lower number moves away from contact (for example, if 300 is contact, 200 would be noncontact).

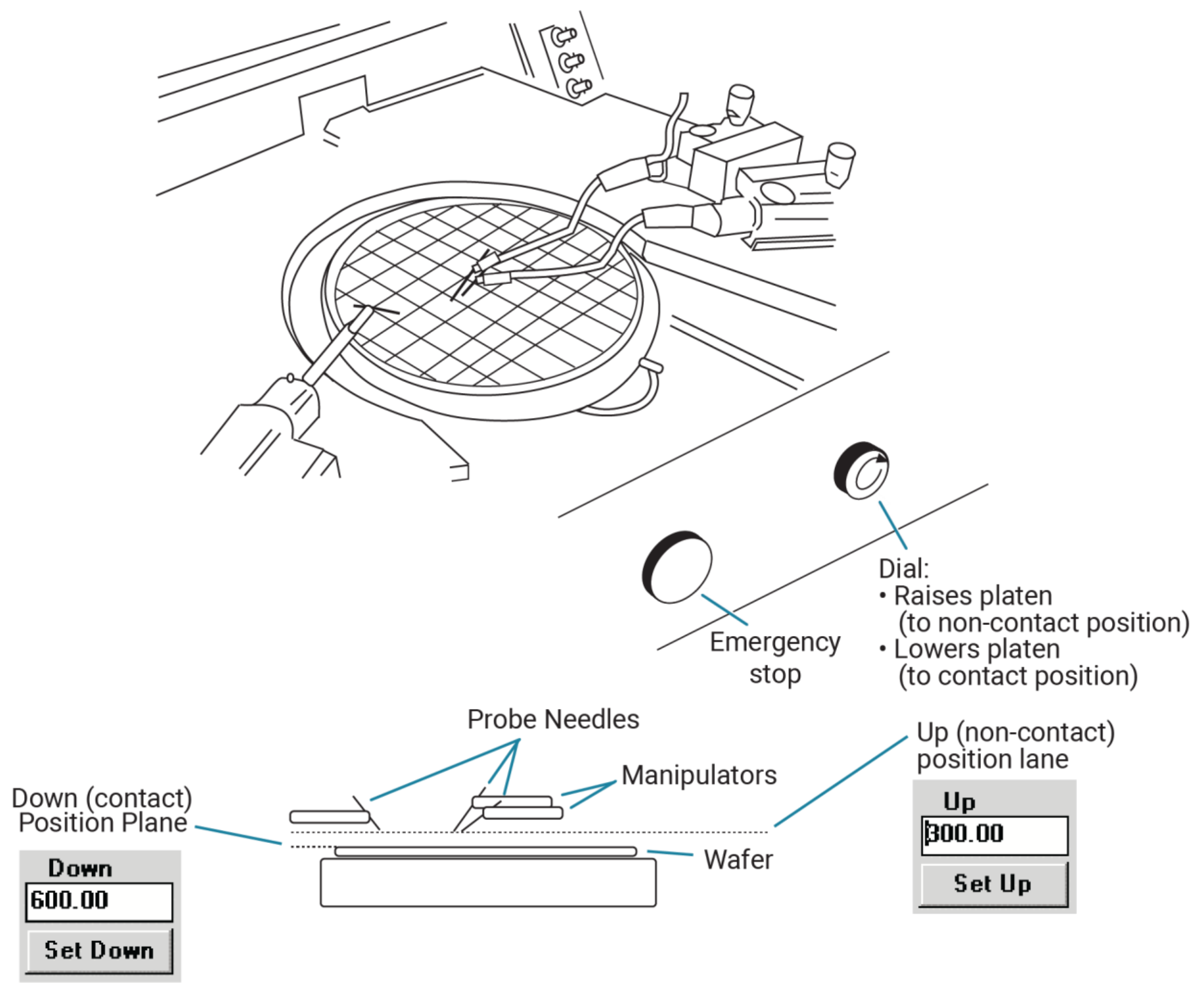
1. Use the joystick to manually move the wafer (pads) underneath the pins.
2. Select the **SET Z UP/DN** button on the pcNav dialog to open the SET Prb8860 Up/Down/Ovd dialog.

Figure 164: SET Prb8860 Up-Down-Ovd window



3. Using the Dial, bring the platen to a positive Z-height (the height in the example is 600). This is a noncontact position with the pads in focus but without the pins touching the pads.

Figure 165: Set Z-height



NOTE

The sample values (600.00 and 300.00) are for illustrative purposes only. The down value (contact will be greater than the up value (non-contact).

4. Use the manual Z-dial to lower the platen to make initial contact with pads (this assumes that the pins are planar).
5. Select the **Set Down** button when all pins are in contact with their respective pad.
6. Use the Dial to move the pins to a noncontact position (this height in the example is 300).
7. Select the **Set Up** button.

NOTE

If the pins are not aligned to the same plane, excessive overdrive/scrub may result (overdrive is the Z-height change necessary to exert adequate contact pressure on the pad). Keep this as equal as possible when manually setting the pins on the pads. Using uneven contact pressure to overcome planarization problems can cause faulty test results or damage to the pad.

8. Set overdrive (user preference).
9. In the pcNav window, select the **DN** button, then select the **Set Z UP/DN** button.
10. When the SET Prb8860 Up/Down/Ovd window opens, press the **Set Base Pt** button.
11. Lower the platen to the point where you see good clean probe marks.
12. Select the **Set 2nd Pt** button.
13. Select the **OVD** button in pcNav to ensure that the overdrive will be used. Test the settings by pressing the UP and DN buttons in pcNav.
14. Select **Done**.

Align the wafer

To align the wafer:

1. Select the **Align Wafer** button on the Tools panel of the pcNav window. The Prb8860 Alignment dialog opens.

Figure 166: Align wafer button

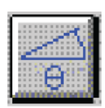
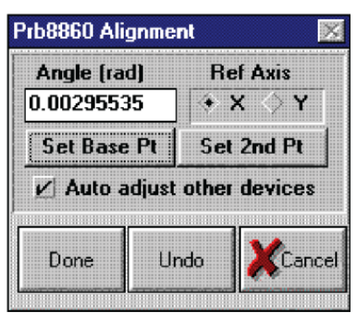


Figure 167: Prb8860 Alignment window



2. Select **Ref Axis X** in the Prb8860 Alignment dialog.

3. Select **Auto adjust other devices**.
4. Move prober chuck to extreme left of the wafer. Look through the microscope and ensure the pins are over the pads.
5. Select **Set Base Pt**.
6. Move to a die on the extreme right of the wafer.
7. Use the joystick (low mode) and theta adjustment to align the pins to the same pads as the first die (both along the same row of die).
8. Select **Set 2nd Pt**.
9. Repeat this process until the Angle (rad) is as close to zero as possible.
10. When the alignment is complete, select **Done**.

Set the units and die size

To set the units and die size:

1. Set **Units** to either microns or mils from the Prb8860 window (lower left corner) of pcNav.
2. Select the **Set X, Y die size** button in the lower middle of the Prb8860 window. The Set X, Y Die Size dialog opens. If die size is known, enter it. If not known, calculate it. See Calculating die sizes (on page 10-17) for more information.

Figure 168: Set X, Y die size button

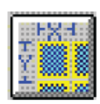
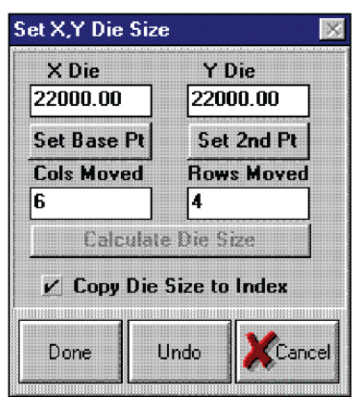


Figure 169: Set X, Y Die Size dialog



Calculate die sizes

To calculate die sizes:

1. Place pins over pads in upper left corner of wafer (although the upper left corner die is used in this example, any die may be selected as a base point).
2. Select **Set Base Pt**.
3. Move over and down a known number of dies. Enter these values into Column moved (columns) and Row moved (rows).
4. Select the **Set 2nd Pt** button.
5. Select the **Copy Die Size to Index** button.
6. Select the **Calculate Die Size** button.

   This determines the accurate die size.
7. To complete, Select **Done** (the grayed-out button is available after the calculation completes).
8. Select the **Set Reference Die** button in the Setup Options window to open the Set Reference dialog.

Figure 170: Set Reference Die button

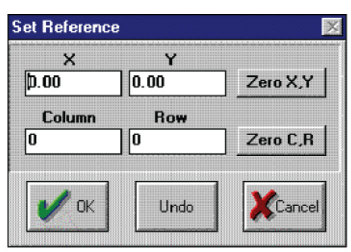
9. Move the pins over the pads of the die to be set as the reference die.
10. Zero out the X/Y, Column, and Row (click Zero X,Y button and Zero C,R).

NOTE

If you want the columns and rows to be something other than 0,0 (1,1 for instance), edit values in Set Reference dialog as needed before selecting **Done**.

11. Select **Done**.

Figure 171: Set Reference dialog



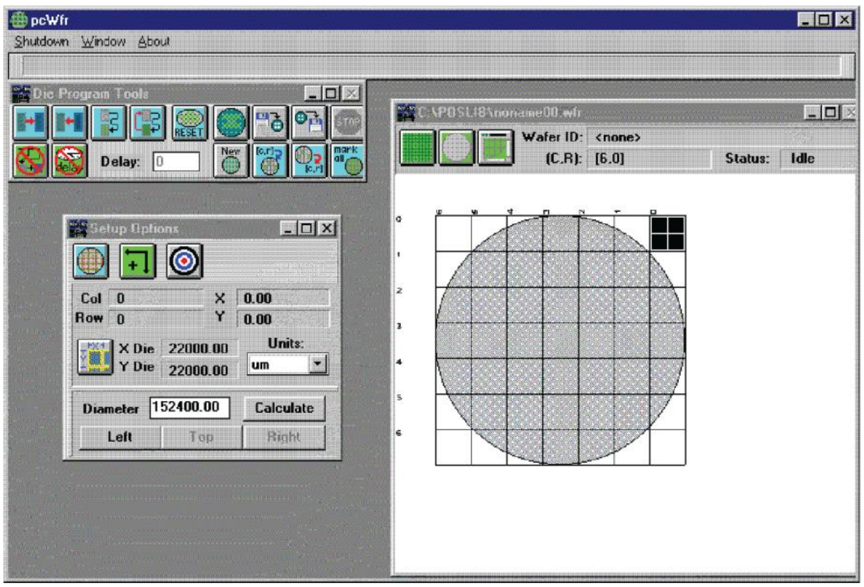
Probesites Clarius project example

The following is a step-by-step procedure to configure the 8860 so the `probesites` Clarius project executes successfully.

On the pcBridge computer:

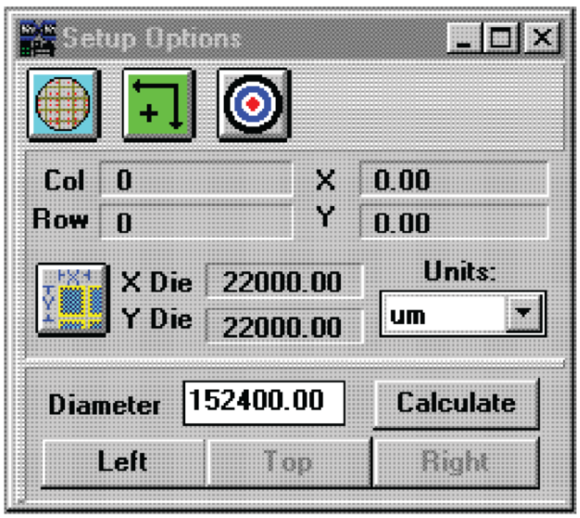
1. Use the pcWafer program to probe a single subsite on multiple dies.
2. In the pcLaunch dialog, select the **pcWfr** button to start pcWafer. The pcWfr dialog is displayed.

Figure 172: PcWfr window



3. Set units of measure (microns or mils).
4. Calculate the wafer diameter:
   a. Move the pins to the left edge of the wafer.
   b. Select **Left** on the Setup Options dialog.
   c. Repeat for the top and right edges of the wafer, clicking the respective buttons after each movement.
   d. Select the **Calculate** button.
5. Set units to either microns or mils from the units of measure Units list in Setup Options dialog.

Figure 173: Setup Options window



6. Select **Set X, Y die size** button in the Setup Options dialog. The Set X, Y Die size dialog opens.
7. Select the **Set X, Y Die size** button.
8. If die size is known, enter it. If not known, calculate (see Calculating die sizes (on page 10-17)).

Set spline pattern (optional)

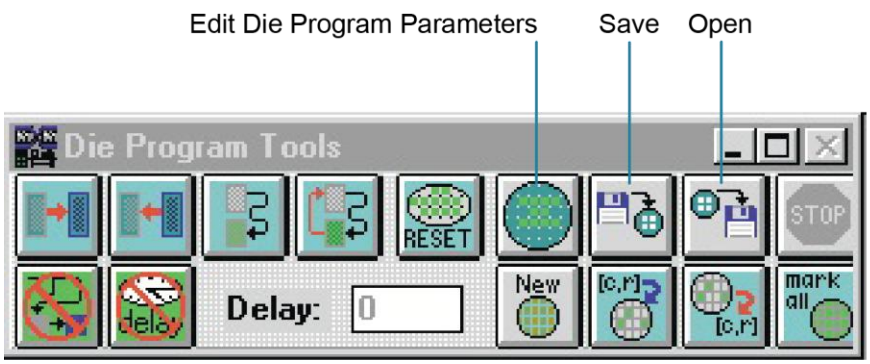
NOTE

The order of selection of the die, the spline pattern (change using edit die program), and the reference die location determine test order sequence.

To set spline pattern:

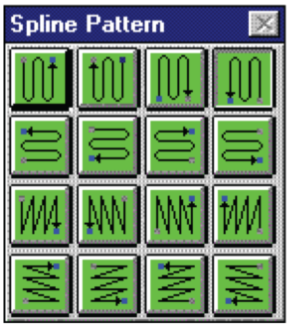
1. Select the **Edit Die Program Parameters** button on the Die Program Tools window of pcWfr. The Edit Die Program Parameters window opens.

Figure 174: Die Program Tools window



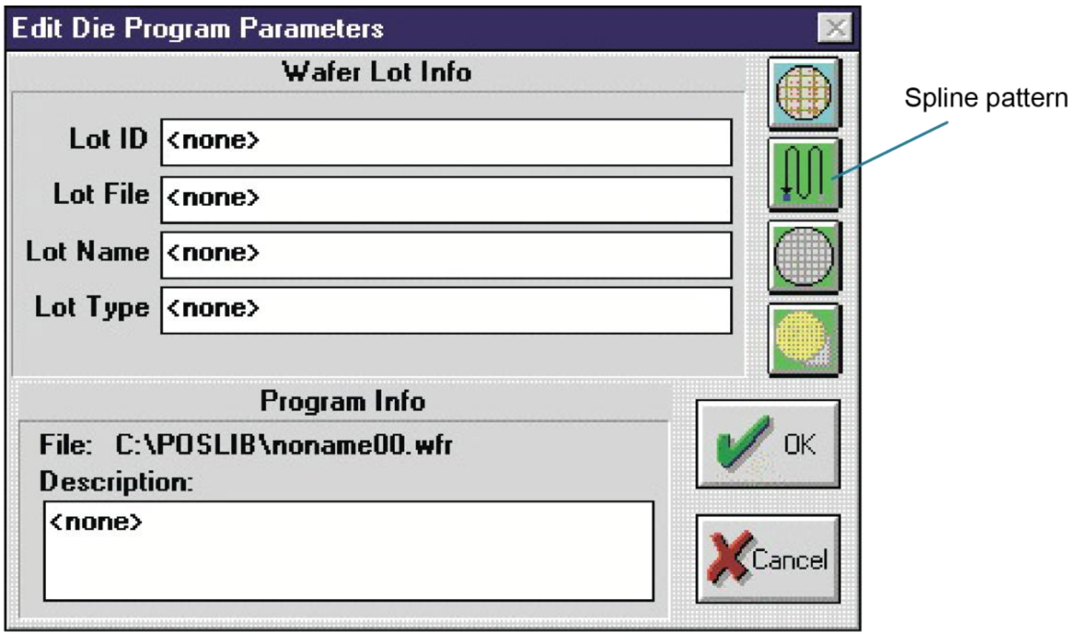
2. Select the **Spline Pattern** button on the Edit Die Program Parameters window. The Spline Pattern window opens.

Figure 175: Spline Pattern window



3. Select the spline pattern. The icon of the active spline pattern is transferred to the Edit Die Program Parameters window.

Figure 176: Edit Die Program Parameters window



4. Select **Save** on the Die Program Tools window.
5. To open an existing program listing file, select the **pcWfr Open** button on the Die Program Tools window. Select the file and select **OK**.

NOTE

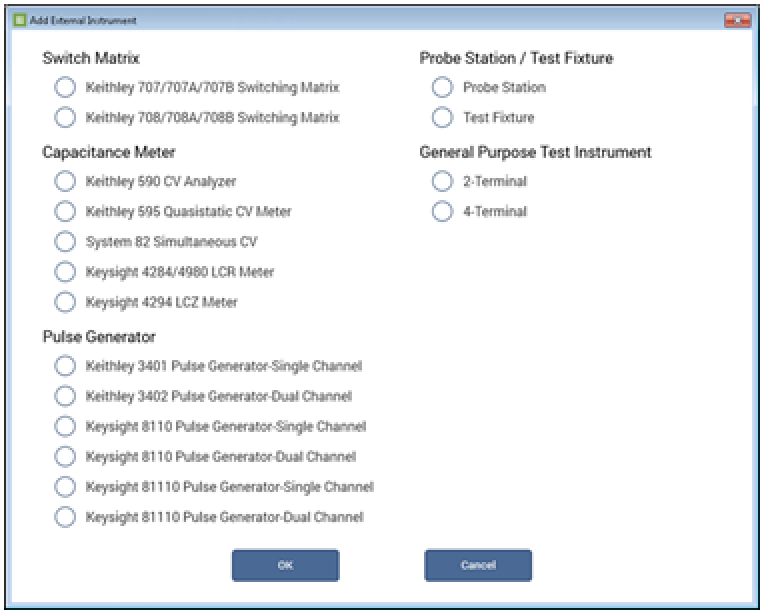
Before starting testing, physically align the pins over the reference die.

Use KCon to add a prober

On the 4200A-SCS, use KCon to add the prober to the configuration:

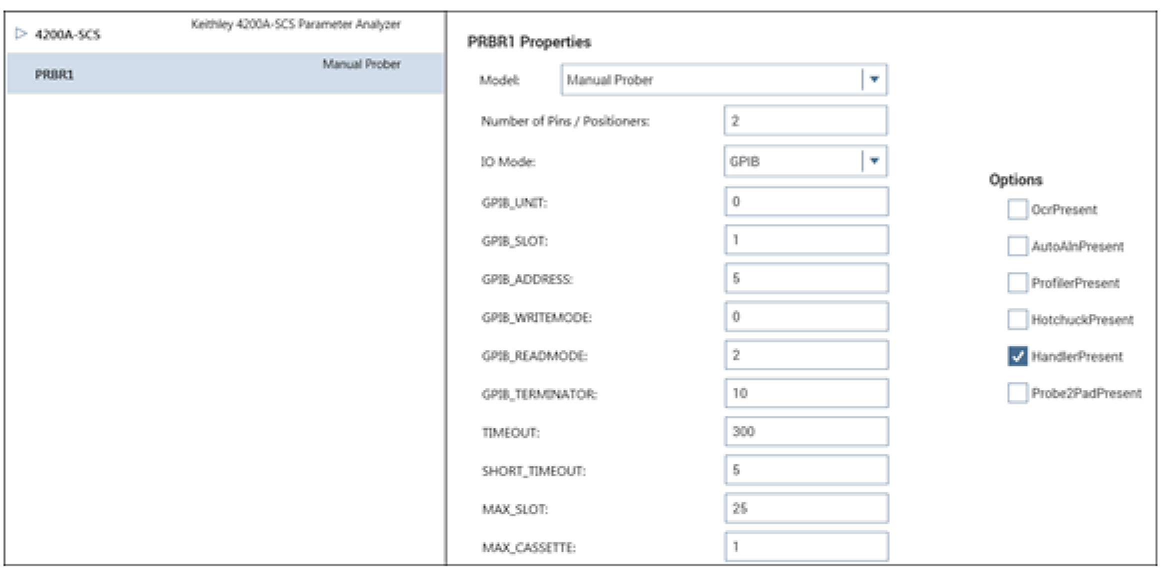
1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

Figure 177: Add a prober in KCon



3. Select **Probe Station**.
4. Select **OK**. KCon displays the properties for the prober.

Figure 178: Use KCon to select a prober



5. Select the **Micromanipulator 8860 Prober** as the model.
6. Ensure that the **Number of Pins / Positioners** is correct. The number of pins defined here determines the pins that are available to assign to a switching matrix card column.
7. Select **Save**.

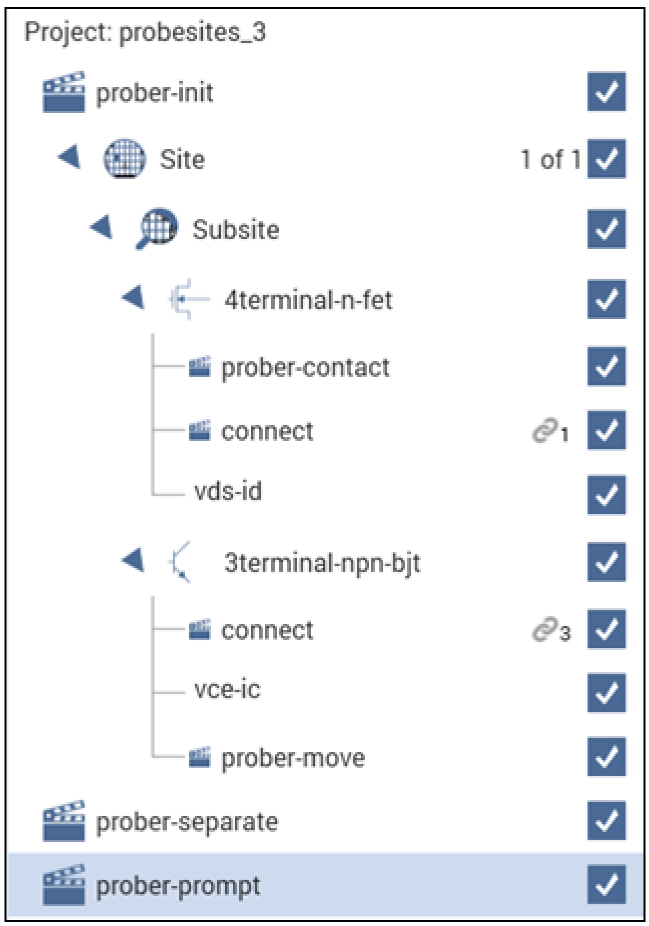
Clarius

Use Clarius to load and run the `probesites` project using the new KCon configuration file. The configuration file allows you to execute a project for this prober.

On the 4200A-SCS:

1. Open Clarius.
2. Choose **Select**.
3. Select **Projects**.
4. Search for **probe**.
5. Drag the **probesites** project to the project tree.

Figure 179: probesites project tree



6. Select **Run**.

Probesubsites Clarius project example

The following is a step-by-step procedure to configure the 8860 so the `probesubsites` Clarius project executes successfully.

When using **pcIndie**, ensure that the project and the program listing on the Micromanipulator match (the program listing is a list of absolute chuck moves in the order of execution). When creating the program listing, use a repeatable pattern.

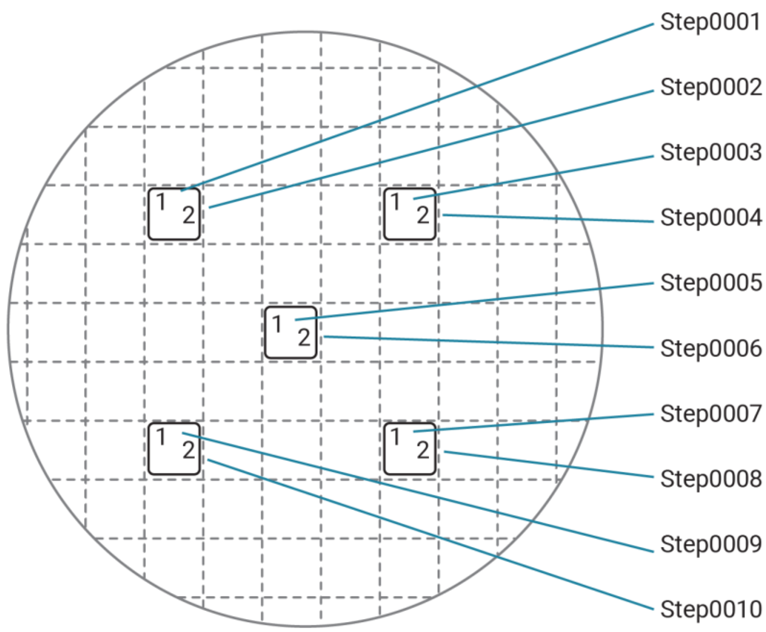
In this example, five dies have been selected for probing. On each die, two subsites have been selected.

Use the pcBridge to configure the 8860:

1. Move to the first subsite of the first die.
2. Add it to the program listing.
3. Move to the second subsite on the first die.
4. Add it to the program listing.
5. Move to the first subsite on the second die.
6. Add it to the program.
7. Continue moving and adding until all subsites have been entered into the list.

Using this type of pattern allows the project structure to issue two **PrSSMovNxt** commands in the loop for each die to be probed. Refer to the following figure for an illustration of a repeatable pattern.

Figure 180: Multiple subsites per die



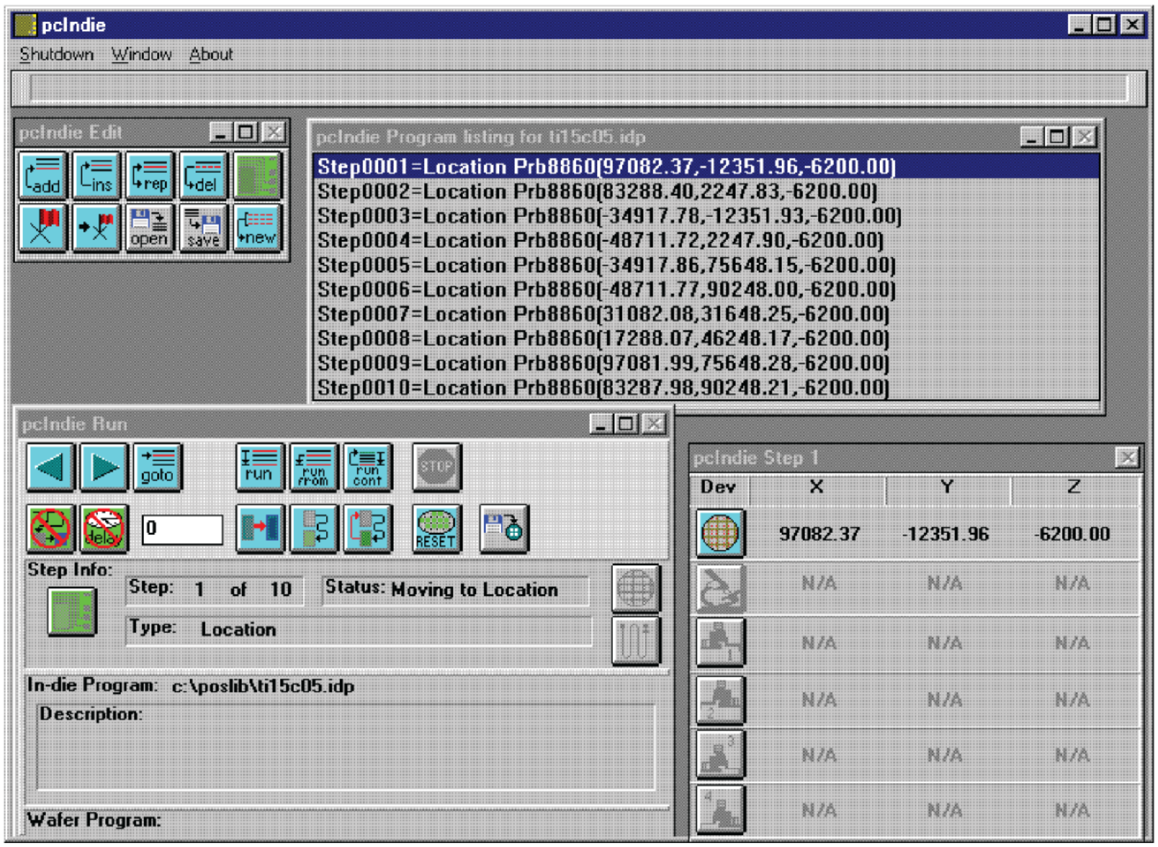
NOTE

Ensure that all steps of Setup have been completed before starting pcIndie.

To start pcIndie:

1. Select the **pcIndie** button in the pcLaunch window. The pcIndie window is displayed, as shown in the following figure.

Figure 181: pcIndie window



2. Use the joystick and microscope to move to the first subsite to be tested.
3. Select **add** or **ins** (insert) in the list.

NOTE

The **add** button adds the description of the present position to the end of the program listing and the **ins** button inserts the present position above the highlighted entry in the program listing. The **rep** button replaces the highlighted entry with the present position and the **del** button deletes the highlighted entry.

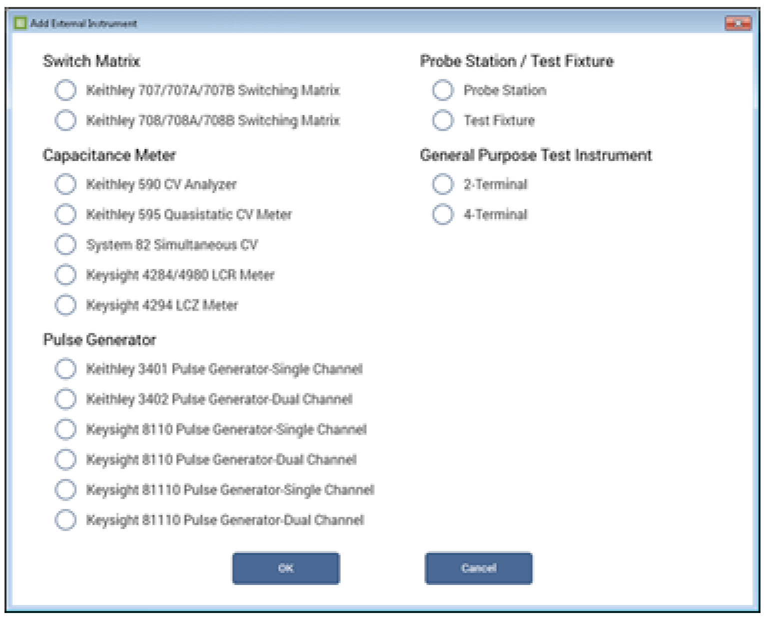
4. Repeat steps 2 and 3 for each subsite to be entered into the program listing.
5. Save by selecting the pcIndie **save** button and assigning the listing a unique file name (`*.idp`).
6. To open an existing program listing file, select the pcIndie **open** button in the pcIndie window. Select the file and select **OK**.

Use KCon to add a prober

On the 4200A-SCS, use KCon to add the prober to the configuration:

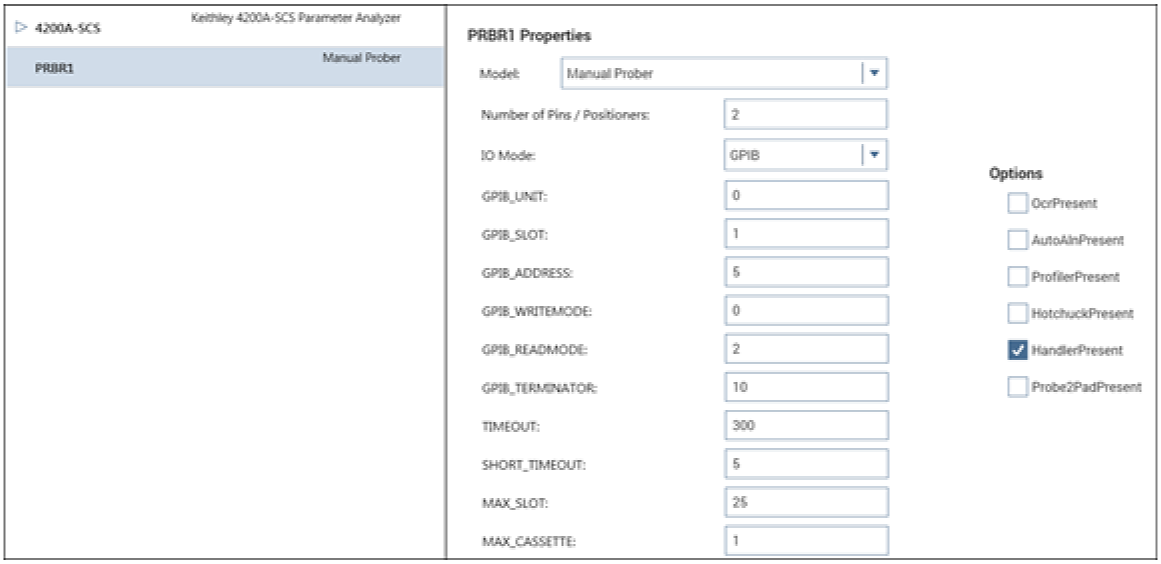
1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

Figure 182: Add a prober in KCon



3. Select **Probe Station**.
4. Select **OK**. KCon displays the properties for the prober.

Figure 183: Use KCon to select a prober



5. Select the **Micromanipulator 8860 Prober** as the model.
6. Ensure that the **Number of Pins / Positioners** is correct. The number of pins defined here determines the pins that are available to assign to a switching matrix card column.
7. Select **Save**.

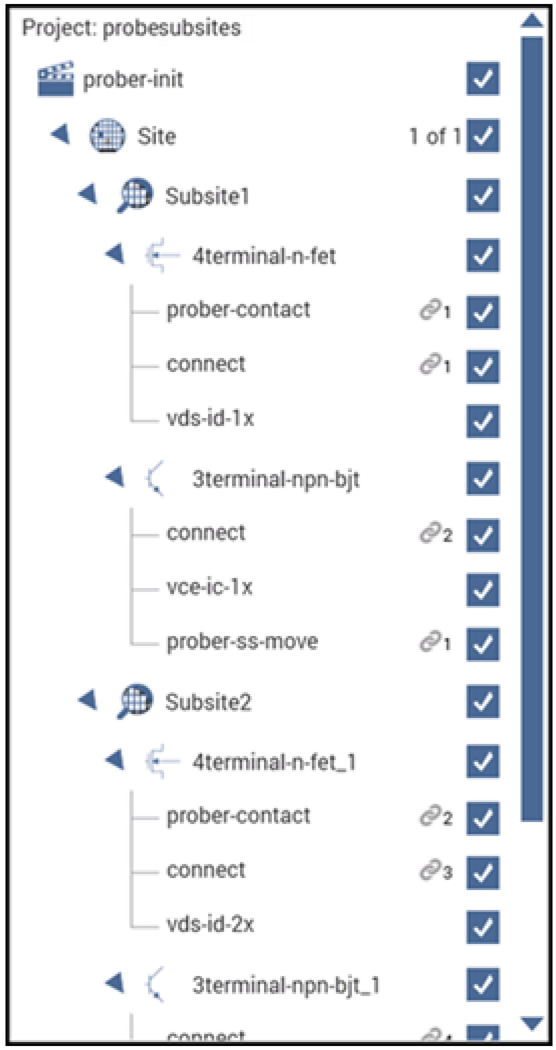
Clarius

Use Clarius to load and run the `probesites` or `probesubsites` project using the new KCon configuration file, which allows you to execute the project for this prober.

On the 4200A-SCS:

1. Open Clarius.
2. Choose **Select**.
3. Select **Projects**.
4. Search for **probesubsites**.
5. Drag the **probesubsites** project to the project tree.

Figure 184: probesubsites project tree



6. Select **Run**.

Commands and error symbols

The following list contains error and status symbols listed by command.

Available commands and responses

	PrChuck	PrInit	PrMovNxt	PrSSMovNxt
PR_OK	X	X	X	X
BAD_CHUCK	X	X		
UNINTEL_RESP	X	X	X	X
UNEXPE_ERROR		X		
SET_MODE_FAIL		X		
INVAL_PARAM		X		
SET_UNITS_FAIL		X		
SET_DIE_FAIL		X		
BAD_MODE			X	X
PR_WAFERCOMPLETE			X	
MOVE_FAIL			X	X
PR_MOVECOMPLETE				X

Information and error code return values and descriptions

Value	Constant	Explanation
1	PR_OK	Success (OK)
2	PR_MOVECOMPLETE	Prober moved to next die (confirmed)
4	PR_WAFERCOMPLETE	Next wafer loaded (confirmed)
-1005	SET_UNITS_FAIL	Failure setting units
-1006	SET_MODE_FAIL	Failure setting mode
-1009	SET_DIE_FAIL	Failure setting die size
-1011	BAD_MODE	Operation invalid in mode
-1013	UNINTEL_RESP	Unintelligible response
-1014	MOVE_FAIL	Movement failure
-1015	UNEXPE_ERROR	Unexpected error
-1017	BAD_CHUCK	Bad chuck position
-1027	INVAL_PARAM	Invalid parameter

Section 11

Using a Micromanipulator P200L Prober

In this section:

Micromanipulator P200L Probe Station requirements

The computer that contains the netProbe 7 (NP7) software must have an NI GPIB-USB-HS interface installed. The 4200A-SCS must have a GPIB interface and Keithley-approved drivers installed.

The following NP7 applications are required to configure and operate the Micromanipulator P200L Semi-Automatic Probe Station with the Keithley 4200A-SCS:

- **P200L_Server/P300L_Server:** Starts the embedded device drivers and provides access to the navigation applications.
- **Navigator:** Used to probe multi-subsites per die. It is accessed in the netProbe 7 Server dialog.
- **Wafer:** Used to probe single subsites per die. It is accessed in the netProbe 7 Server dialog.
- **Router:** Communicates from the host to NP7 applications. It is accessed in the netProbe 7 Server dialog.

You must also have the following NI GPIB software:

- NI VISA Version 17.5.
- NI 488.2 Version 17.6.

Probe station configuration

CAUTION

Ensure that you are familiar with the Micromanipulator P200L Probe Station and its supporting documentation before attempting setup, configuration, or operation.

To set up and configure the P200L Probe Station for use with the 4200A-SCS, you:

- Set up communications (on page 10-2)
- Set up wafer geometry (on page 10-6)
- Create a site definition and define a probe list (on page 10-8)
- Load, align, and contact the wafer (on page 10-9)

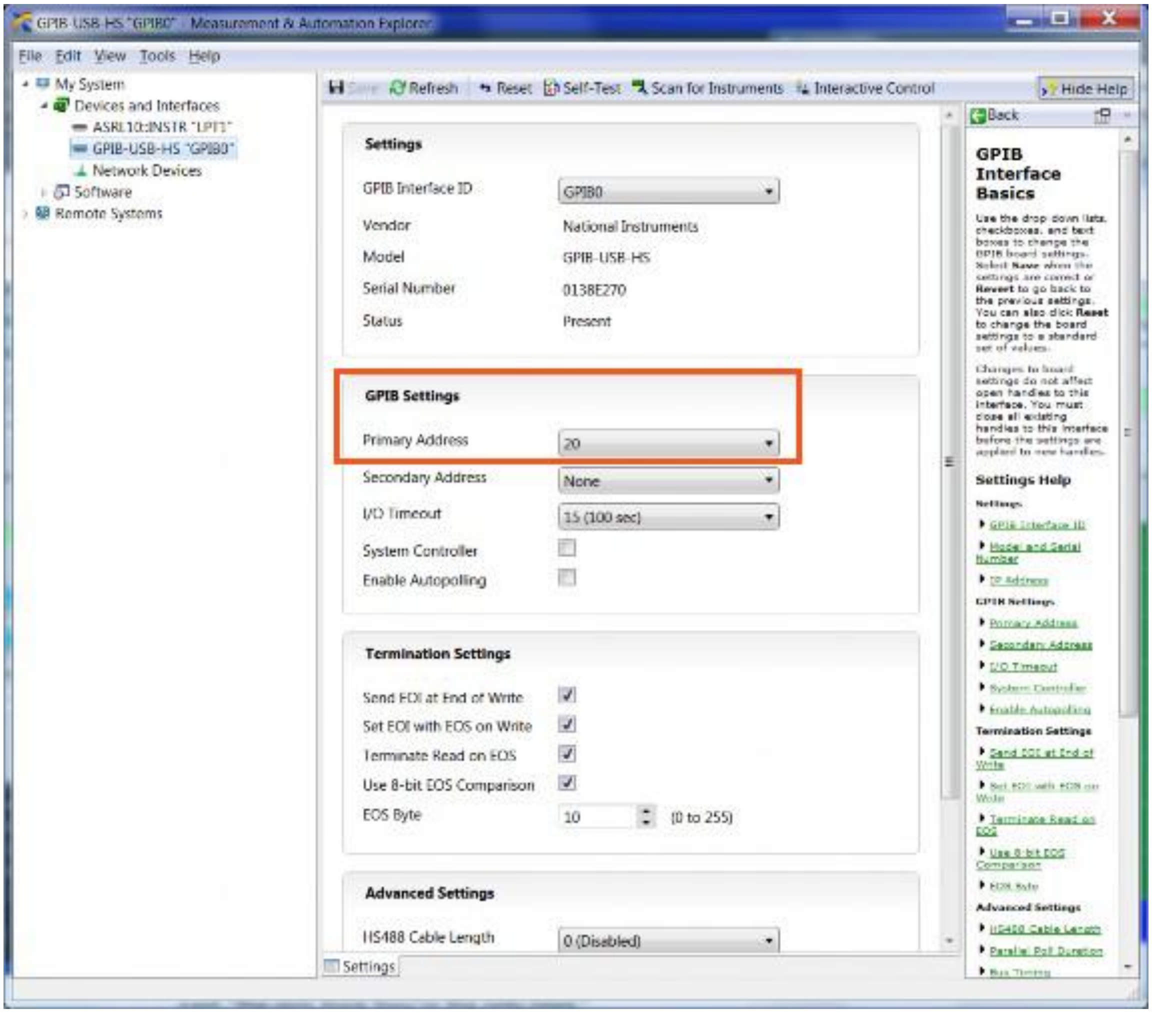
Each step is detailed in the following topics.

Set up communications

To set up communications:

1. Turn on power to the 4200A-SCS.
2. Turn on power to the probe station.
3. Ensure that the vacuum is properly connected.
4. Connect the GPIB-USB-HS cable of the NP7 computer to the GPIB connector on the rear panel of the 4200A-SCS.
5. Open NI-MAX.
6. In the Application Settings dialog, select **GPIB-USB-HS**.

Figure 185: GPIB Primary Address setting

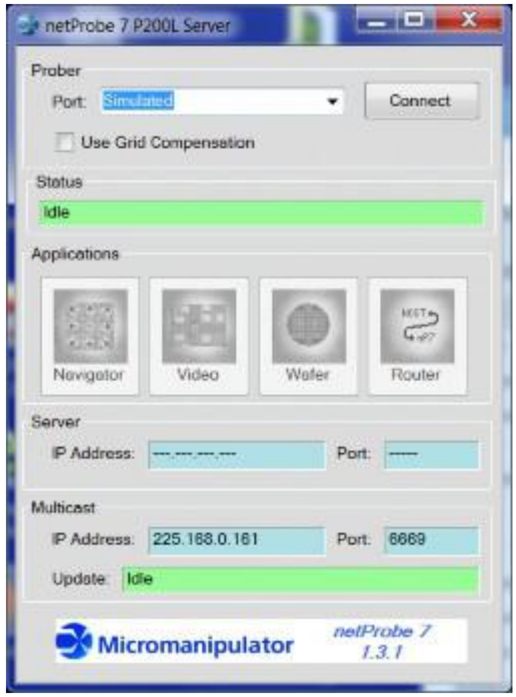


7. Set the **Primary Address** to the probe station GPIB address.
8. Select **Save**.

Set up wafer geometry

Start netProbe 7:

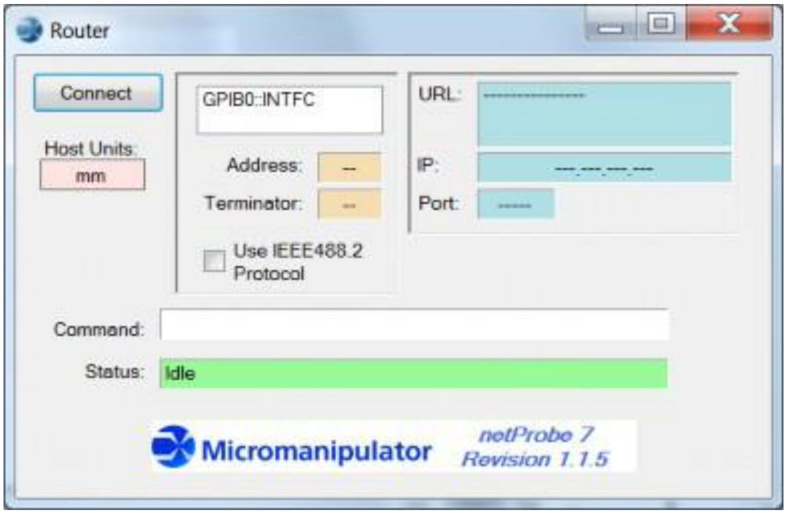
1. On the desktop, double-click **P200L_Server or P300L_Server**.



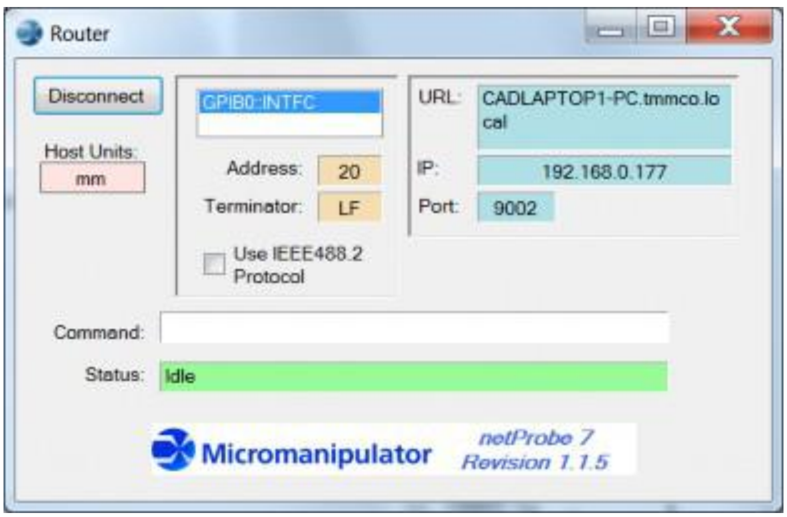
2. From the **Port** list, select the USB Port that connects the controller and the netProbe 7 computer. The port name is in the format `ASRL<comport#>::INSTR`, for example, `ASRL3:INSTR`. If a physical probe station is not available, select **Simulated**.
3. Select **Connect**.

Figure 186: NP7 connected

4. Select **Router** to open the Router dialog:



5. Select **GPIB0::INTFC**.
6. Select **Connect**.



7. Verify that the GPIB address and string terminator character of the netProbe 7 router matches the NI MAX settings.

Create a site definition and define a probe list

To create a site definition for a single subsite for each die, use the Navigator to select the die to probe. For an example of a project that has a single subsite per die to be probed, refer to Clarius project example for probesites (on page 11-11).

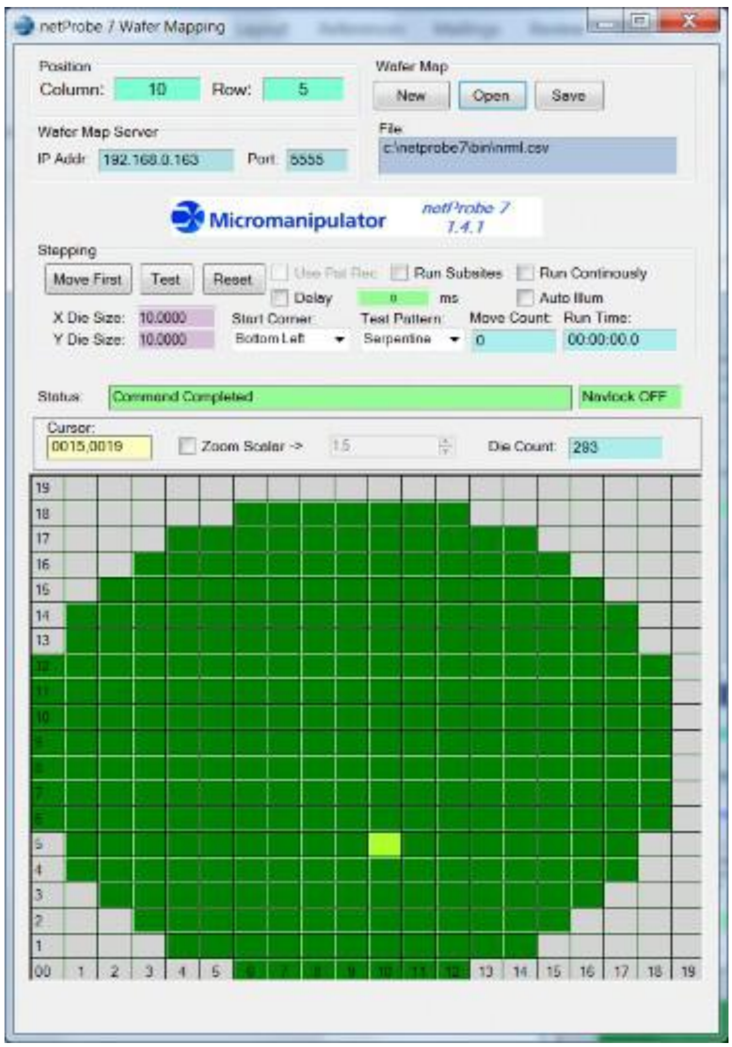
To create a site definition for multiple subsites for each die, you can use the `probesubsites` Clarius project.

Use the following information to load a previously created site definition.

To create site definition for a single subsite for each die:

1. Select the wafer button to display the Wafer Mapping dialog.

Figure 187: NP7 Wafer Mapping dialog

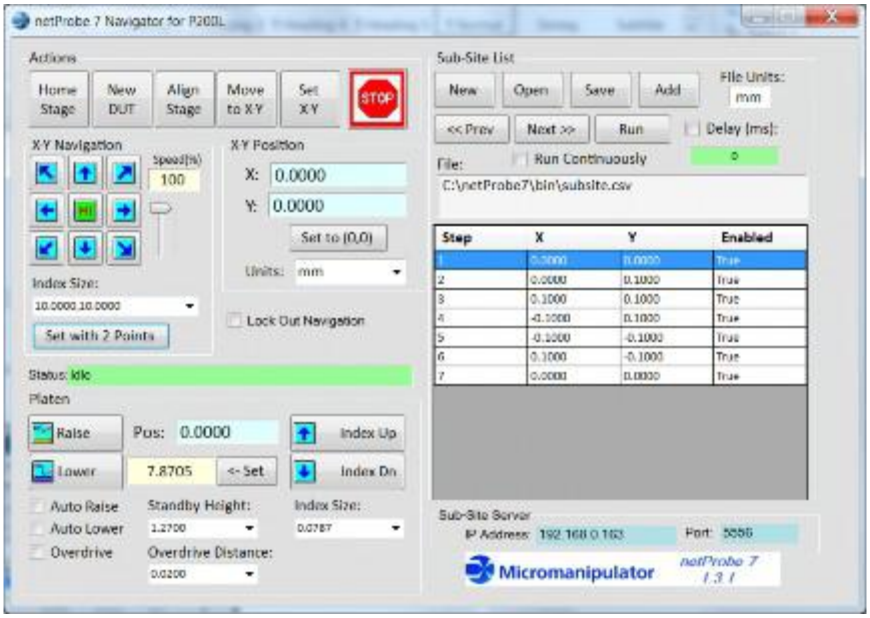


2. Select **Open**.
3. Select the file, then select **OK**. For example, from the `netProbe7\bin_200` folder, select `hp_wafer.csv`.

To create site definition for multiple subsites for each die:

1. From the Server dialog, select **Navigator**. The Navigator dialog is displayed.

Figure 188: NP7 Navigator dialog



2. Select the Sub-Site List **Open** button.
3. Select the file, then select **OK**. For example, from the `netProbe7\bin_200` folder, select `HP_Subsite.csv`.

Load, align, and contact the wafer

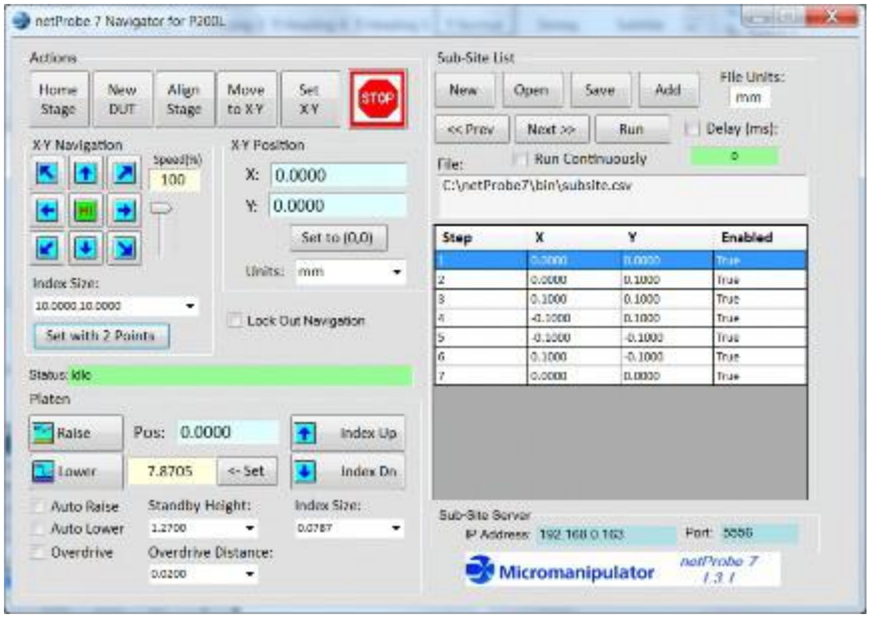
The following topics describe how to contact the wafer.

Move the stage to the home position

To move the stage to the home position:

1. From the Server dialog, select **Navigator**. The Navigator dialog is displayed, as shown in the following figure.

Figure 189: NP7 Navigator dialog



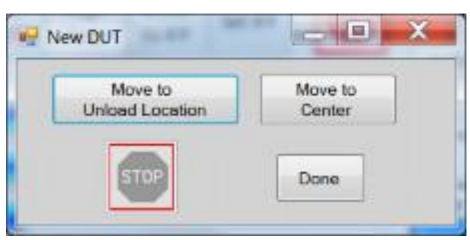
2. Select the **Home Stage** button in the Navigator.
3. Ensure that the microscope home path (left rear) is clear and that probes will safely clear any obstacles or objectives.
4. Select **Home XYZ** to move the stage to the home position. The status line on the netProbe Navigator dialog shows the home status.
5. Select **Done** when all moves are complete.

Load a wafer or DUT

To load a wafer or device under test (DUT):

1. In the Navigator, select **New DUT**. The New DUT dialog is displayed.

Figure 190: New DUT dialog

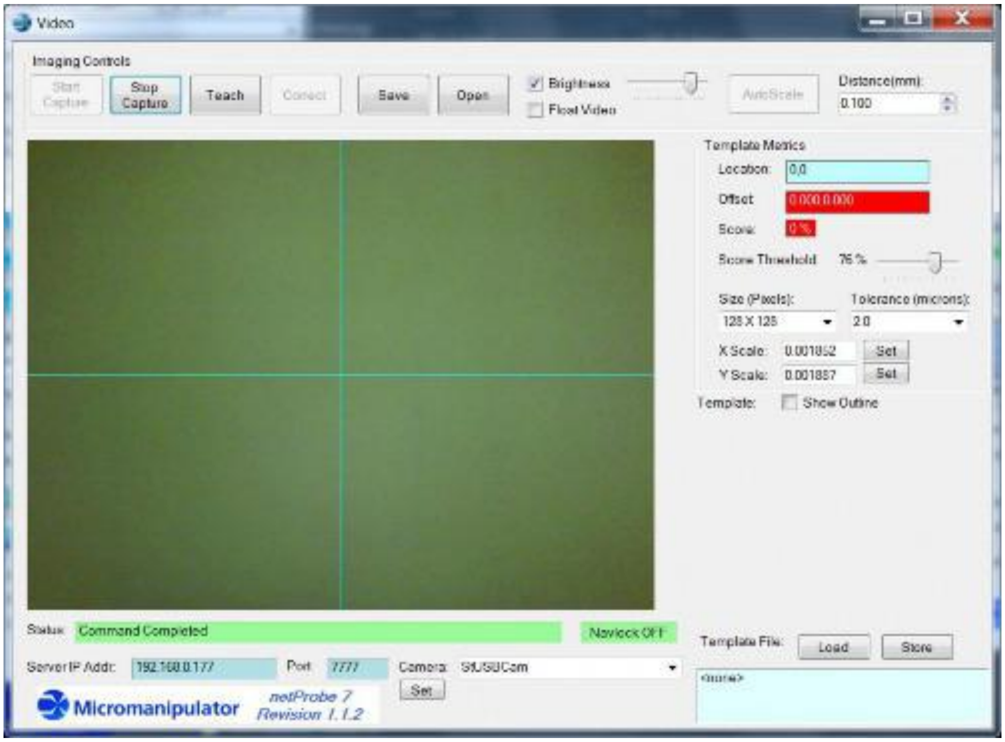


2. To move the stage to the preset unload location, select **Move to Unload Location**.
3. To move the probe station stage to its center of travel, select **Move to Center**.
4. Select **Done**.

Align the stage

To align the stage:

1. From the Navigator, select **Video**. The Video dialog is displayed.
2. Select **Start Capture** to begin streaming live video from the probe station. The dialog looks similar to the following figure.



3. Position the probe station so the video crosshair is perfectly aligned with the bottom edge of a die on the middle left side of the wafer. Approach the point from the upper right to ensure accuracy.
4. Move the probe station along the row of dies to the right until the bottom edge of the target die is in the field of view.
5. Rotate the manual theta knob until the bottom edge of the die is aligned with the crosshair.
6. Move to the opposite side of the wafer and manually rotate the theta knob until the die is aligned with the crosshair.
7. Verify the alignment by moving along the wafer and observing deviation in Y as motion occurs in X.
8. Perform these steps until the required alignment accuracy is attained.

Set the Z height

To set the Z height:

1. From the Navigator, enter the **Standby Height** and **Overdrive Distance** values.
2. Turn off the Auto Raise and Lower options. An example of the settings is shown in the following figure.

Figure 191: Platen settings to prepare for Z setup

3. Rotate the X, Y, and Z knobs on each manual manipulator until the X,Y, and Z stages are near the center of available manipulator travel. This gives the broadest range of adjustment motion.
4. Observe the Z probe tip positions on all the manipulators. If the Z height of all the probe tips vary by more than 100 mils (2540 microns), it may be necessary to adjust the probe tips in the probe holders until all the tips are close to the same height.
5. Identify the manipulator with the lowest probe tip. If there is some doubt, lower one until it is clearly the lowest.
6. Place the manipulators approximately in the final configuration.
7. Observe the Z distance between the probe tips and the surface of the wafer.
8. Lower the platen using the joystick until the lowest probe tip is about 20 mils (500 microns) above the wafer. As the probe tips approach the wafer, slow the rate of descent to prevent unintentional probe contact with the wafer.
9. Select the **Set** button.
10. Place the lowest manipulator so the tip is over its intended target.
11. Position the microscope over the lowest probe tip.
12. Slowly lower the platen until the probe tip contacts the target. Larger tipped probes usually start to scrub across the wafer when contact is made. Very small tipped probes tend to penetrate the surface. It may be necessary to adjust the X,Y position of the manipulator tip before contact is made to ensure that the probe tip contacts the target.
13. Select the **Set** button to set the probe contact position.
14. Select the **Raise** button to raise the platen by the standby distance.
15. Select the **Lower** button and verify probe contact.

Clarius project example for probesites

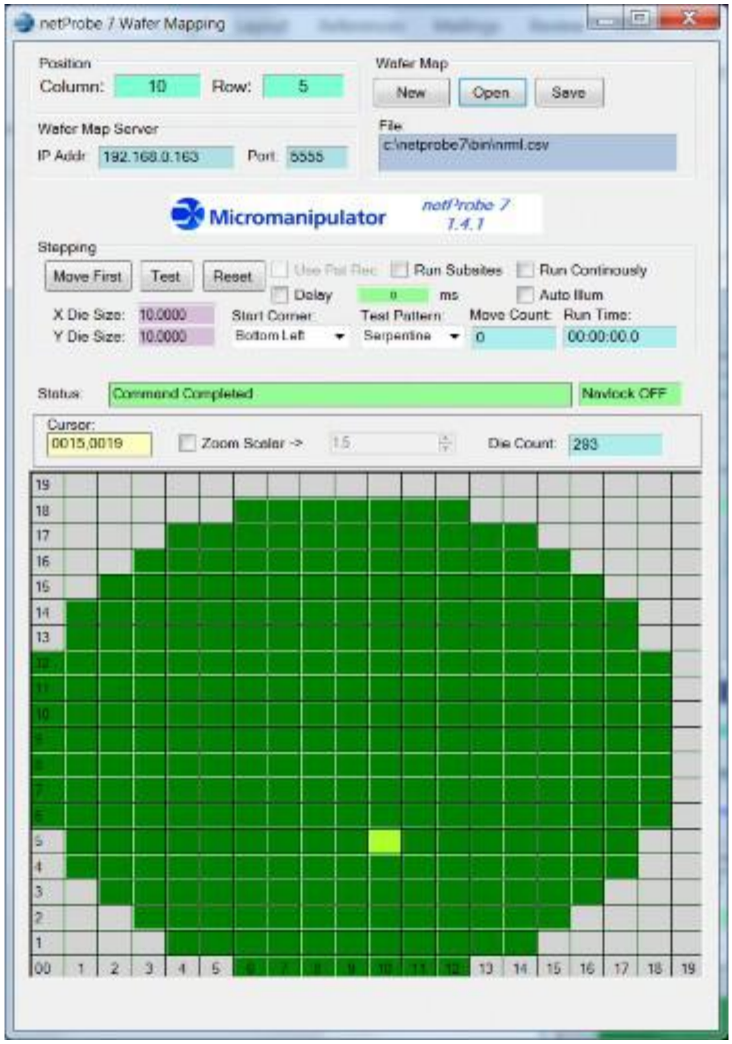
The following procedure configures the P200L and netProbe 7 server so the `probesites` Clarius project executes successfully.

To run this example:

1. Open the netProbe 7 P200L Server.

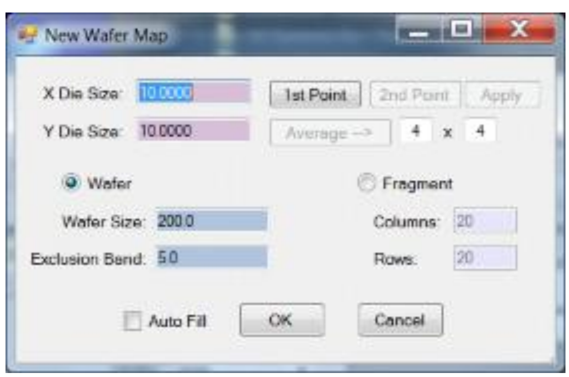
2. Select the wafer button to display the Wafer Mapping dialog.

Figure 192: NP7 Wafer Mapping dialog



3. Under Wafer Map, select **New**. The New Wafer Map dialog is displayed.

Figure 193: New Wafer Map dialog



4. Enter the die sizes or calculate them using two set points.
5. Enter the **Wafer Size** and **Exclusion Band**.

NOTE

You can also select **Auto Fill** to calculate the true range of rows and columns.

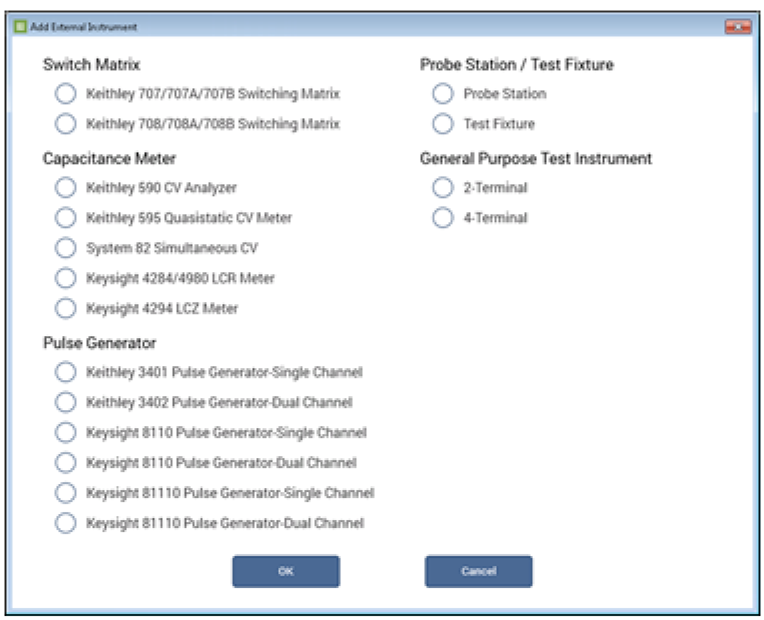
6. Select **OK** to create a new map.
7. In the Wafer dialog, select the **Start Corner** and **Test Pattern**.
8. Select **Move First**.
9. Position the first die under the probes and align the probes to their corresponding pads.
10. Right click the first die and select **Reference** from the list.

Use KCon to add a probe station

On the 4200A-SCS, use KCon to add the prober to the configuration:

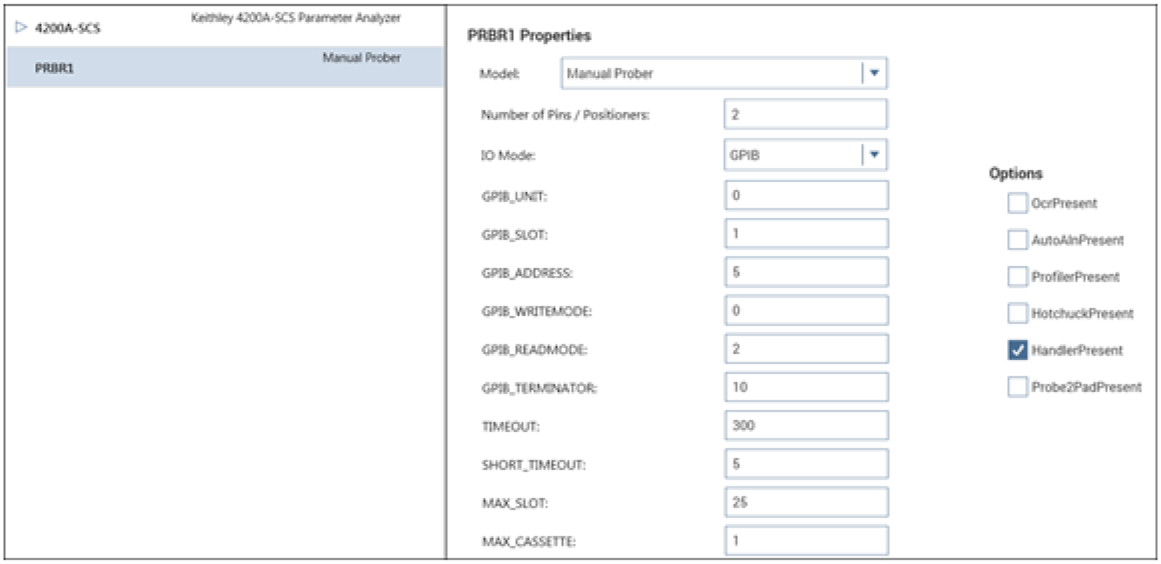
1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

Figure 194: Add a prober in KCon



3. Select **Probe Station**.
4. Select **OK**. KCon displays the properties for the prober.

Figure 195: Use KCon to select a prober



5. Select the **Micromanipulator P200L Prober** as the model.
6. Set the properties as follows:
   - **Number of Pins / Positioners:** 2
   - **IO Mode:** GPIB
   - **GPIB_UNIT:** 0
   - **GPIB_SLOT:** 1
   - **GPIB_ADDRESS:** 20
   - **GPIB_READMODE:** 2
   - **GPIB_WRITEMODE:** 0
   - **TERMINATOR:** 10
   - **TIMEOUT:** 300
   - **SHORT_TIMEOUT:** 5
   - **MAX_SLOT:** 25
   - **MAX_CASSETTE:** 2
7. Select **Save**.
8. Exit KCon.

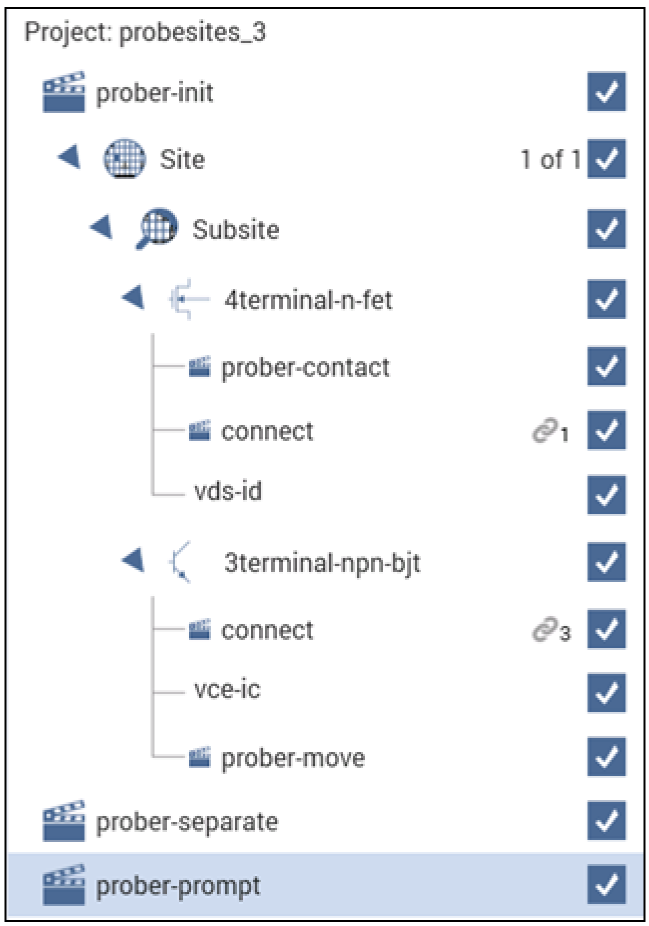
Clarius

Use Clarius to load and run the `probesites` project using the new KCon configuration file. The configuration file allows you to execute a project for this prober.

On the 4200A-SCS:

1. Open Clarius.
2. Choose **Select**.
3. Select **Projects**.
4. Search for **probe**.
5. Drag the **probesites** project to the project tree.

Figure 196: probesites project tree



6. Select **Run**.

Commands and errors

The following table lists the errors listed by command.

Available commands and responses

	PrChuck	PrInit	PrMovNxt	PrSSMovNxt
BAD_CHUCK	X	X	—	—
BAD_MODE	—	—	X	X
MOVE_FAIL	—	—	X	X
PR_OK	X	X	X	X
PR_WAFERCOMPLETE	—	—	X	X
SET_MODE_FAIL	—	X	—	—
UNINTEL_RESP	X	X	X	X
UNEXPE_ERROR	—	X	—	—

Using a manual or fake prober

In this section:

Using a manual or fake prober software

The Keithley 4200A-SCS provides all software required for both manual and fake prober operation; no additional software is needed.

Remote control of the prober is disabled when using manual or fake probers. The probe station must be manually controlled. The user is also responsible for the prober station setup.

Manual prober overview

Use the MANL prober to test without using automatic prober functionality. Configuring the environment for a MANL prober replaces all computer control of the prober with that of the operator, while allowing the user to step through each command in the sequence. At each prober command, a dialog is displayed that instructs the operator.

The probing sequence using the MANL prober:

1. Start a project.
2. Issue a `PrInit` command to tell the user to initialize the prober. The `PrInit` dialog opens.
3. Select **OK** to continue. The project sets up the measurement system to test the first site.
4. Issue a `PrChuck` command to tell the user to ensure that the first test site is ready for testing. The `PrChuck` dialog opens and the user continues by selecting **OK**. Tests on the site are executed.
5. Issue a `PrMovNxt` command to tell the user to move to the next site to be tested on the wafer. The `PrMovNxt` dialog opens and the user continues by selected **OK**.

NOTE

Subsite probing uses the `PrssMovNxt` command to move to the next subsite.

6. Issue `PRChuck` and `PRMovNxt` commands until all sites are tested.

Fake prober overview

Use the FAKE prober to test without probing. You can use this to take the prober offline when you want to run the test without modifying your project. Configuring the environment for a FAKE prober stops all prober actions.

When using the FAKE prober, you can execute tests individually or in a loop. This allows you to debug projects without removing prober calls. Situations when the FAKE prober mode may be useful:

1. Looping on the same wafer location using a project that supports wafer prober operations (for instance, testing one site 100 times instead of testing 100 different sites once).
2. Disabling prober function calls until the testing portions of the project are functioning correctly.

Modifying the prober configuration file

NOTE

You can modify these files using the 4200A-SCS.

The default prober configuration file for a manual prober (MANL) is shown in the following code example. The only relevant line for this prober type is `PROBER_1_PROBTYPE=MANL`.

```
# prbcnfg.dat  - EXAMPLE Prober Configuration File MANL Prober
#
# The following tag, "PRBCNFG", is used by the engine in order to determine
# the MAX number of SLOTS and CASSETTES for a given prober at runtime.
#
<PRBCNFG>
#
# for OPTIONS "" == NULL, max 32 chars in string
#
# Example
#        01234567890
#PROBER_1_OPTIONS=1,1,1,1,1,1
#
#
#   OcrPresent
#   AutoAlnPresent
#   ProfilerPresent
#   HotchuckPresent
#   HandlerPresent
#   Probe2PadPresent
#
# Configuration for MANuaL probers (S900NT):
#  MANL
#
PROBER_1_PROBTYPE=MANL
PROBER_1_OPTIONS=0,0,0,0,1,0
PROBER_1_IO_MODE=GPIB
PROBER_1_GPIB_UNIT=0
PROBER_1_GPIB_SLOT=1
PROBER_1_GPIB_ADDRESS=5
PROBER_1_GPIB_WRITEMODE=0
PROBER_1_GPIB_READMODE=2
PROBER_1_GPIB_TERMINATOR=10
PROBER_1_TIMEOUT=300
```

The default prober configuration file for a fake prober is shown in the following code example. The only relevant line for this prober type is `PROBER_1_PROBTYPE=FAKE`.

```
# prbcnfg.dat  - EXAMPLE Prober Configuration File, FAKE prober
#
# The following tag, "PRBCNFG", is used by the engine in order to determine
# the MAX number of SLOTS and CASSETTES for a given prober at runtime.
#
<PRBCNFG>
#
# for OPTIONS "" == NULL, max 32 chars in string
#
# Example
#        01234567890
#PROBER_1_OPTIONS=1,1,1,1,1,1
#
#
#   OcrPresent
#   AutoAlnPresent
#   ProfilerPresent
#   HotchuckPresent
#   HandlerPresent
#   Probe2PadPresent
#
#
# The PROBER_x_PROBTYPE fields needs to be set to one of the following names.
# Configuration for serial probers:
#  FAKE
#
#
PROBER_1_PROBTYPE=FAKE
PROBER_1_OPTIONS=0,0,0,0,1,0
PROBER_1_IO_MODE=SERIAL
PROBER_1_DEVICE_NAME=COM1
PROBER_1_BAUDRATE=9600
PROBER_1_TIMEOUT=300
PROBER_1_SHORT_TIMEOUT=5
PROBER_1_MAX_SLOT=25
PROBER_1_MAX_CASSETTE=1
```

As shown, the manual configuration file is configured for use with GPIB prober communications, while the fake configuration file is configured for serial prober communications. To make changes, use a text editor such as Microsoft™ Notepad.

Configuration file locations:

- Fake: `C:\s4200\sys\dat\prbcnfg_FAKE.dat`
- Manual: `C:\s4200\sys\dat\prbcnfg_MANL.dat`

Probesites Clarius project example

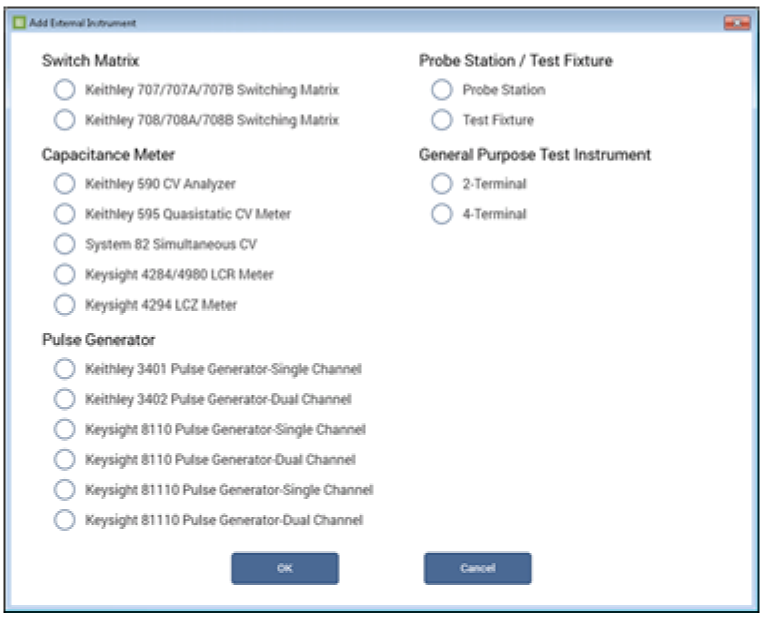
The following is a step-by-step procedure to configure the manual prober so the `probesites` Clarius project executes successfully. The user is responsible for the probe station setup.

Use KCon to add a prober

On the 4200A-SCS, use KCon to add the prober to the configuration:

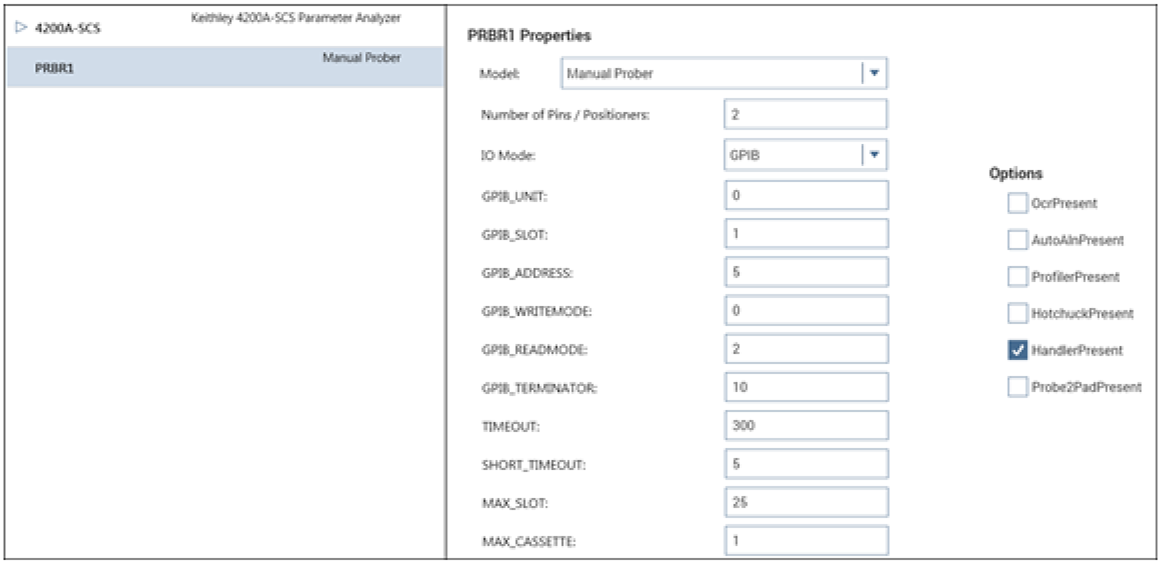
1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

Figure 197: Add a prober in KCon



3. Select **Probe Station**.
4. Select **OK**. KCon displays the properties for the prober.

Figure 198: Use KCon to select a prober



5. Select the **Manual Prober** or the **Fake Prober** as the model.
6. Ensure that the **Number of Pins / Positioners** is correct. The number of pins defined here determines the pins that are available to assign to a switching matrix card column.
7. Select **Save**.

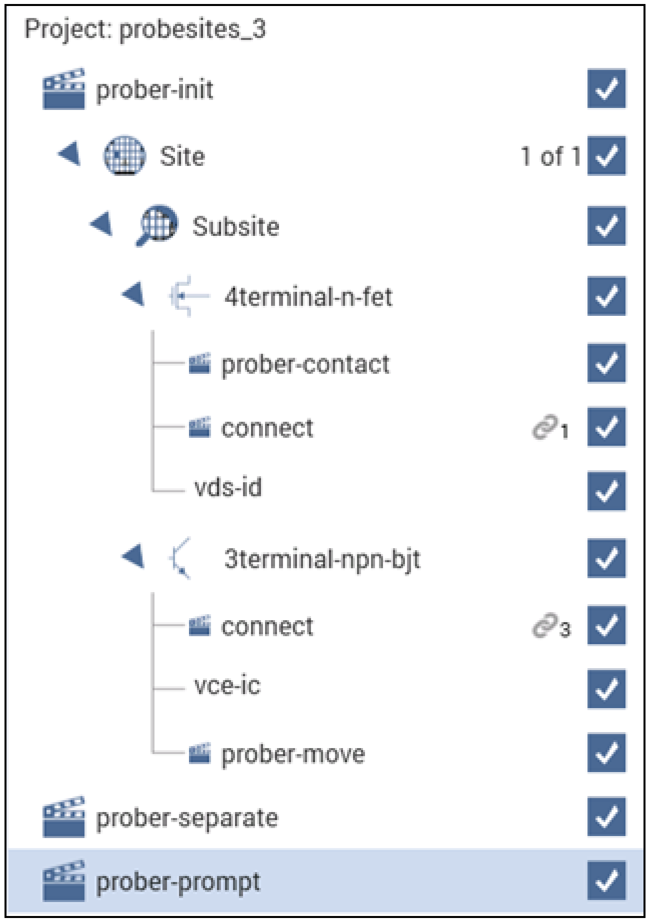
Clarius

Use Clarius to load and run the `probesites` project using the new KCon configuration file. The configuration file allows you to execute a project for this prober.

On the 4200A-SCS:

1. Open Clarius.
2. Choose **Select**.
3. Select **Projects**.
4. Search for **probe**.
5. Drag the **probesites** project to the project tree.

Figure 199: probesites project tree



6. Select **Run**.

Probesubsites Clarius project example

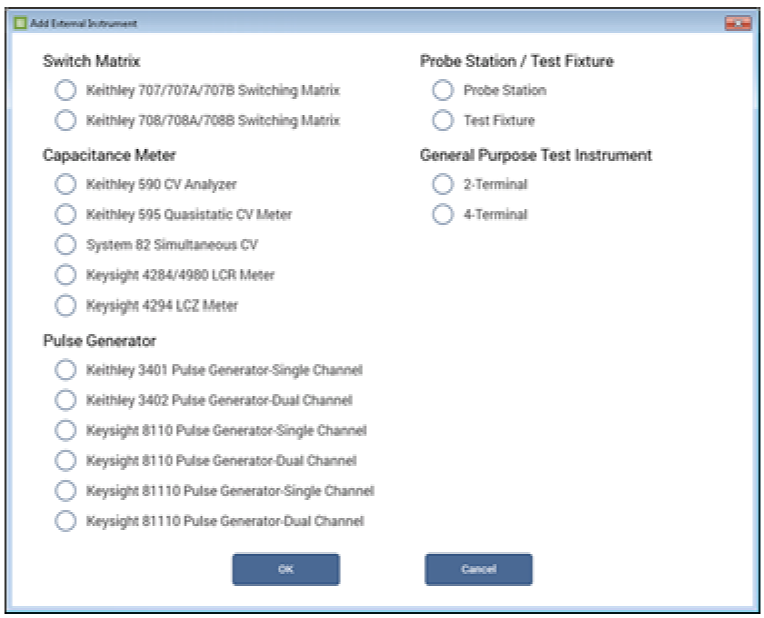
The following is a step-by-step procedure to configure the manual prober so the `probesubsites` Clarius project executes successfully. The user is responsible for the probe station setup.

Use KCon to add a prober

On the 4200A-SCS, use KCon to add the prober to the configuration:

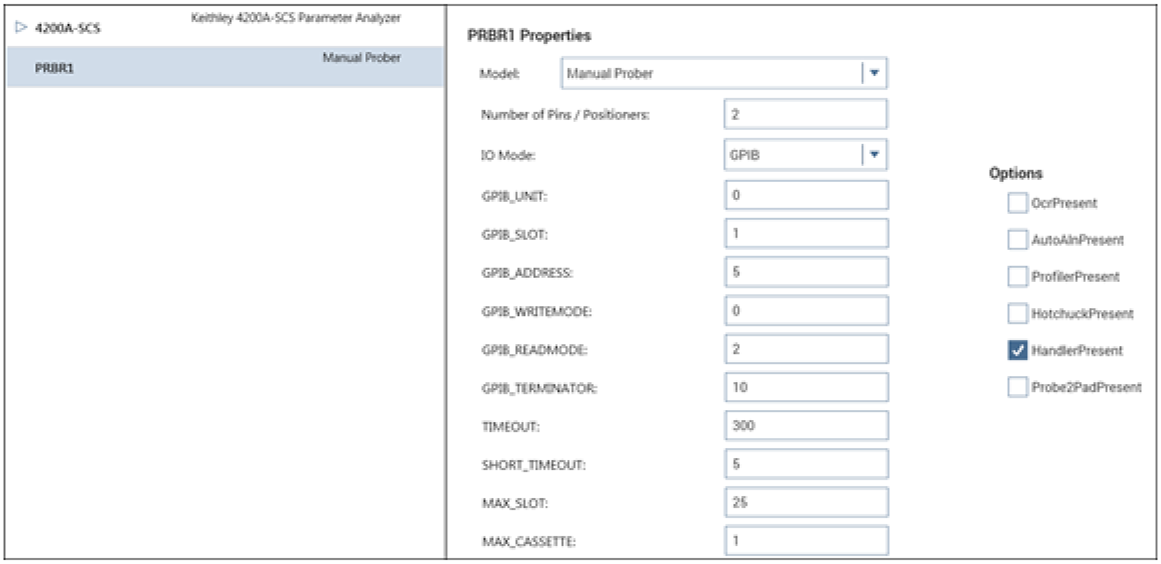
1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

Figure 200: Add a prober in KCon



3. Select **Probe Station**.
4. Select **OK**. KCon displays the properties for the prober.

Figure 201: Use KCon to select a prober



5. Select the **Manual Prober** or the **Fake Prober** as the model.
6. Ensure that the **Number of Pins / Positioners** is correct. The number of pins defined here determines the pins that are available to assign to a switching matrix card column.
7. Select **Save**.

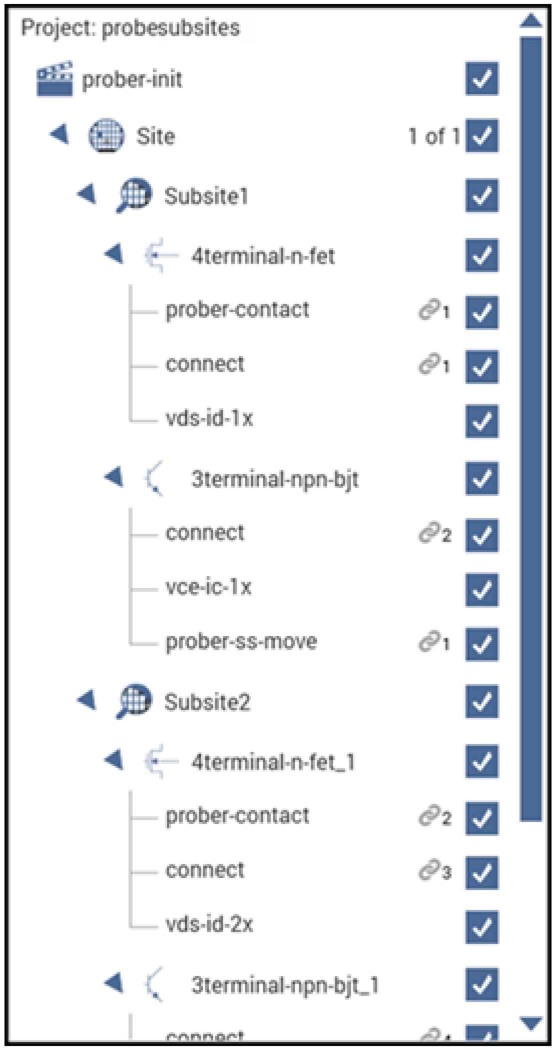
Clarius

Use Clarius to load and run the `probesites` or `probesubsites` project using the new KCon configuration file, which allows you to execute the project for this prober.

On the 4200A-SCS:

1. Open Clarius.
2. Choose **Select**.
3. Select **Projects**.
4. Search for **probesubsites**.
5. Drag the **probesubsites** project to the project tree.

Figure 202: probesubsites project tree



6. Select **Run**.

Using a Cascade Summit-12000 Prober

In this section:

Cascade Summit 12000 prober software

The Formfactor Cascade Summit 12000 prober will have one of the following software products installed. Use either of the following software products to configure and operate the Summit 12000 prober with the Keithley 4200A-SCS:

- **Velox prober control software**: Provides access to configuration and help programs

  or

- **Nucleus UI prober control software**: Provides access to configuration and help programs

Software version

The following software versions were used to verify the configuration and remote command set of the Summit-12000 prober with the 4200A-SCS:

- Velox ver. 2.3

  or

- Nucleus UI ver. 2.0

Probe station configuration

CAUTION

Make sure that you are familiar with the Summit-12000 Prober and its supporting documentation before attempting setup, configuration, or operation.

The general steps required to set up and configure the Summit-120000 prober for use with the 4200A-SCS:

- Set up communications (on page 13-2)
- Set up wafer geometry (on page 13-8)
- Create a site definition and define a probe list (on page 13-11)
- Load, align, and contact the wafer (on page 13-13)

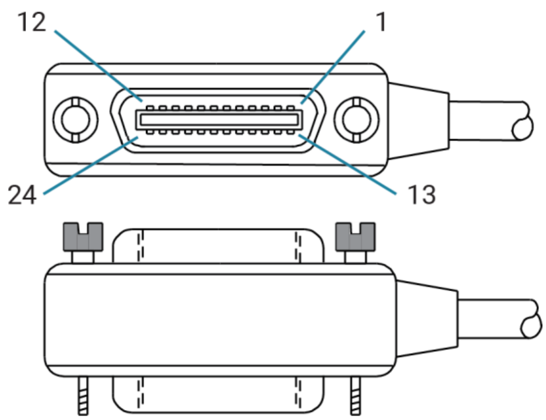
Set up communications

The Summit-12000 prober is configured for GPIB communications only.

Connect the 4200A-SCS and the probe station

To connect the equipment, connect the 4200A-SCS GPIB port and the probe station PC GPIB port using a shielded GPIB cable. See the following figure and table for pinouts.

Figure 203: IEEE-488 connector contact numbers



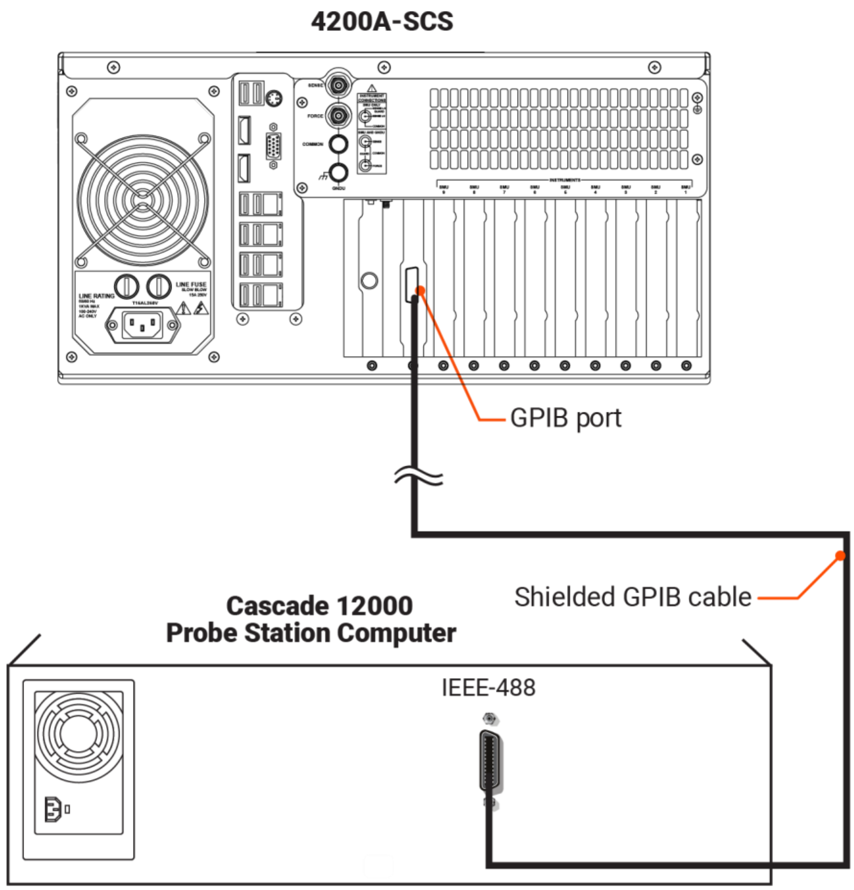
GPIB control connector terminals

Contact number	GPIB designation	Type
1	DI01	Data
2	DI02	Data
3	DI03	Data
4	DI04	Data
5	EOI (24)*	Management
6	DAV	Handshake
7	NRFD	Handshake
8	NDAC	Handshake
9	IFC	Management
10	SRQ	Management
11	ATN	Management
12	SHIELD	Ground
13	DI05	Data
14	DI06	Data
15	DI07	Data
16	DI08	Data
17	REN (24)*	Management
18	Gnd (6) *	Ground
19	Gnd (7) *	Ground
20	Gnd (8) *	Ground
21	Gnd (9) *	Ground
22	Gnd (10) *	Ground
23	Gnd (11) *	Ground
24	Gnd, LOGIC	Ground

*Numbers in parentheses refer to signal ground return of referenced contact number. EOI and REN signal lines return on contact 24.

The following figure shows connections between the Cascade Summit-12000 prober to the Keithley 4200A-SCS.

Figure 204: Connection diagram



Set up probers using the Velox prober control software

For probers using the Velox prober control software:

On the probe station computer, refer to the Velox user manual and help content for information on how to setup and configure the GPIB interface. The following settings should be made in Velox to the GPIB configuration for operation with the 4200A:

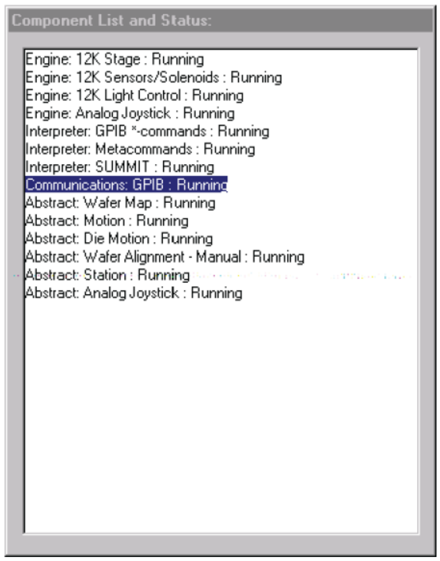
- Device Address: 28 - Recommend using 28, as this is the default address used by the 4200A
- Response Terminator: CR - Carriage Return
- Service Request: OFF
- TSK emulation: OFF
- SCPI string responses: ON
- Legacy SCPI status byte: ON

Set up probers using the Nucleus UI prober control software

For probers using the Nucleus UI prober control software:

1. On the probe station computer, double-click the **Nucleus** icon.
2. Log in using the **Nucleus System Login**.
3. After login is complete, the prober initializes the stage. Select **Proceed** when the prober has completed initialization.
4. Maximize the system manager **Component List and Status** program (right-click the system manager label on the taskbar and choose **Maximize**).
5. Select **Communications: GPIB** on the component list.

Figure 205: Component list and status dialog

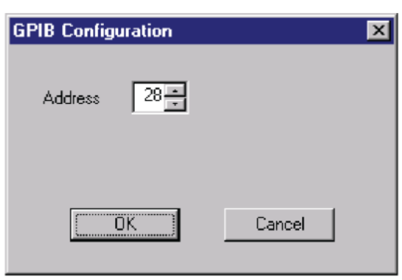


NOTE

If the **Communications: GPIB** component is not on the list, you must add it. To add it, select **Add** from the Add component dialog, then select **Communications: GPIB**.

6. If the **Communications: GPIB** component is running, select **Stop**, or proceed to the next step (setup).
7. Select **Setup** to open the GPIB configuration dialog.

Figure 206: GPIB Configuration dialog



8. Change the address as needed. The default value is 28.
9. Save the configuration file by selecting **Save**.

Figure 207: Save button

10. Start the component by selecting **GO**.
11. Minimize, but do not close, the system manager window.
12. Select **Remote** on the Nucleus UI toolbar to display the Remote Window. See the following figures.

Figure 208: Remote button

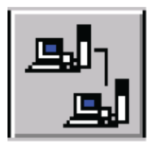
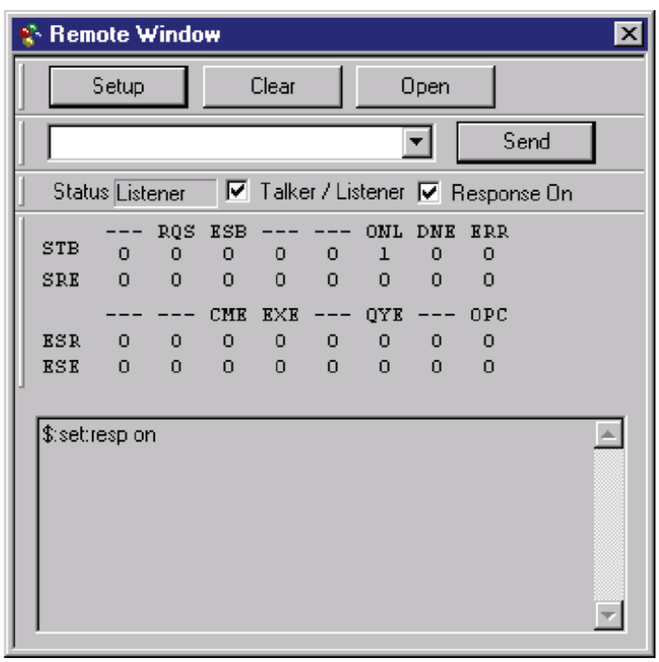
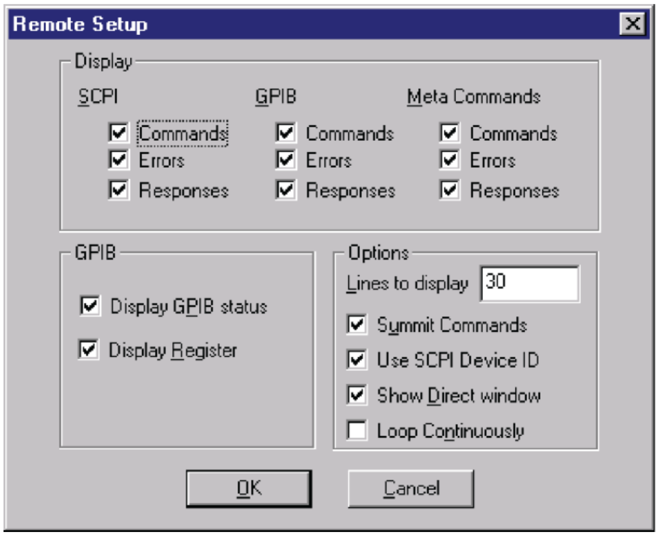


Figure 209: Remote Window



13. Select the **Talker / Listener** and **Response On** boxes in the Remote Window.
14. Select **Setup** on the Remote Window dialog to display the Remote Setup dialog.

Figure 210: Remote setup window



15. Select the items to be displayed.
16. Select **OK**.

NOTE

Selecting boxes on the Remote Setup dialog only affects the DISPLAY properties. It will not change the GPIB physical setting. Use the dialog in the GPIB configuration window to make changes to the GPIB address.

Set up wafer geometry

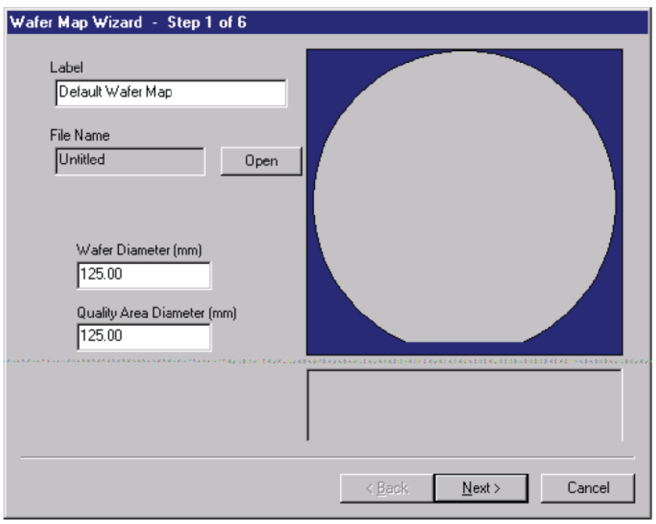
For probers using the Velox prober control software:

On the probe station computer, refer to the Velox user manual and help content for information on how to create wafer maps. Note a wafer map wizard is available in the Velox prober control software to assist with this. The die size, reference position, and testing sequence should be specified.

For probers using the Nucleus UI prober control software:

1. On the probe-station computer, if the Nucleus toolbar is not already open, double-click the **Nucleus** icon on the Windows desktop.
2. Log in.
3. From the Window menu of the Nucleus toolbar, select **WaferMap** to display the wafer map window.
4. From the **File** menu of the Wafer Map window, select **Wizard** to start the Wafer Map wizard.

Figure 211: Step 1: Wafer Map Wizard

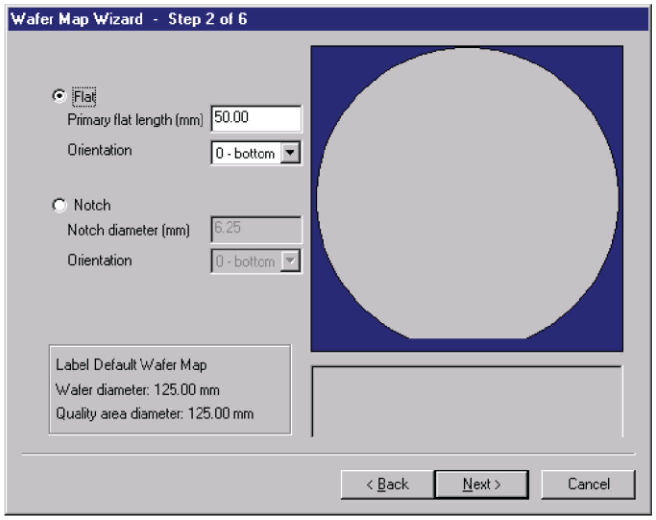


5. Enter the label and wafer diameter in the Wafer Map Wizard window.
6. Select **Next**.
7. Select **Flat** or **Notch** based on the actual wafer.
8. Enter either the primary flat length or the notch diameter in millimeters.
9. Select the orientation of the flat or notch as applicable.

NOTE

Bottom is toward the front of the prober.

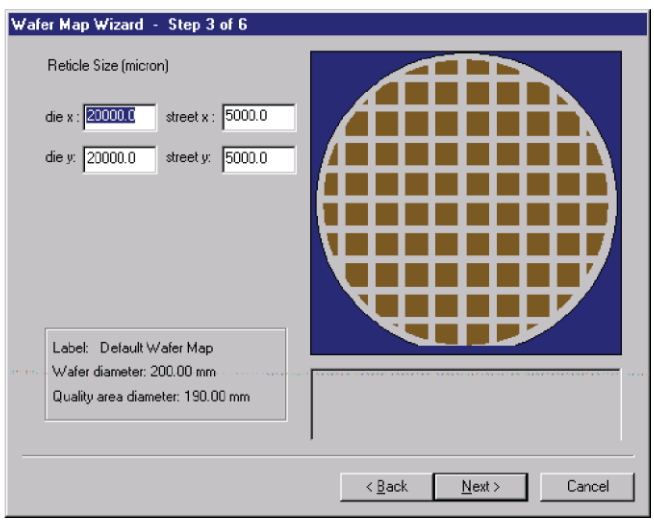
Figure 212: Step 2: Wafer Map Wizard



10. Select **Next**.

11. Enter the correct die and street sizes.

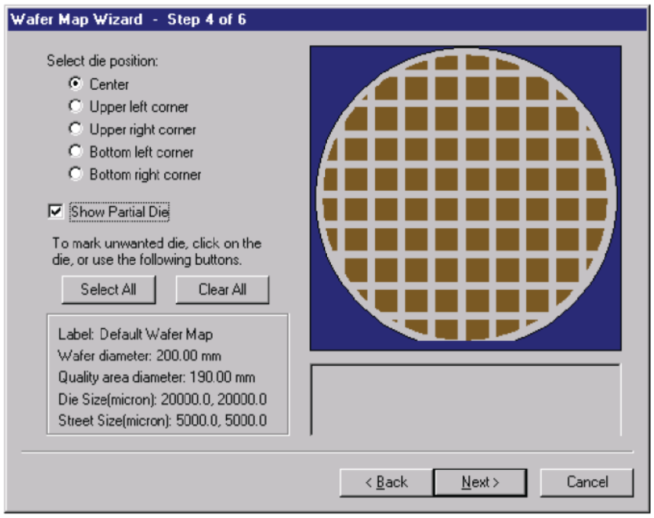
Figure 213: Step 3: Wafer Map Wizard



12. Select **Next**.

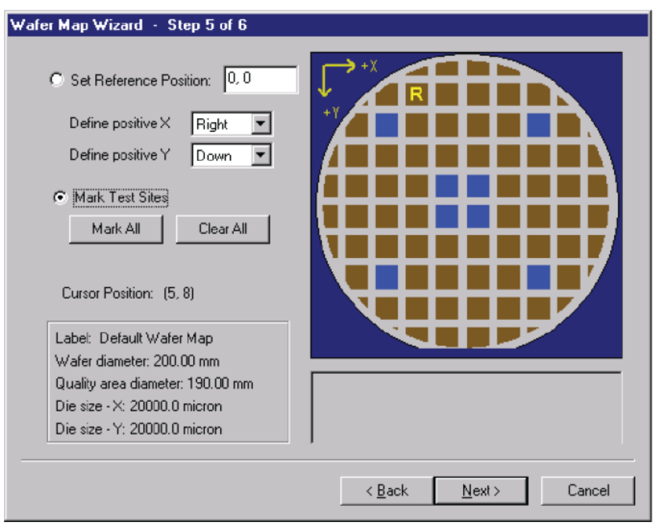
13. Select the die position. Optionally, select **Show Partial Die**.

Figure 214: Step 4: Wafer Map Wizard



14. Select **Next**.
15. Set the reference position.
16. Enter positive X and Y value directions (this defines the coordinate). For example, setting **Define Positive X: Right**, and **Define Positive Y: Up** would define the coordinate as Quadrant I, while setting **Define Positive X: Right**, and **Define Positive Y: Down** would define the coordinate as Quadrant IV.
17. Select **Mark Test Sites**. You can drag to select multiple sites.

Figure 215: Step 5: Wafer Map Wizard

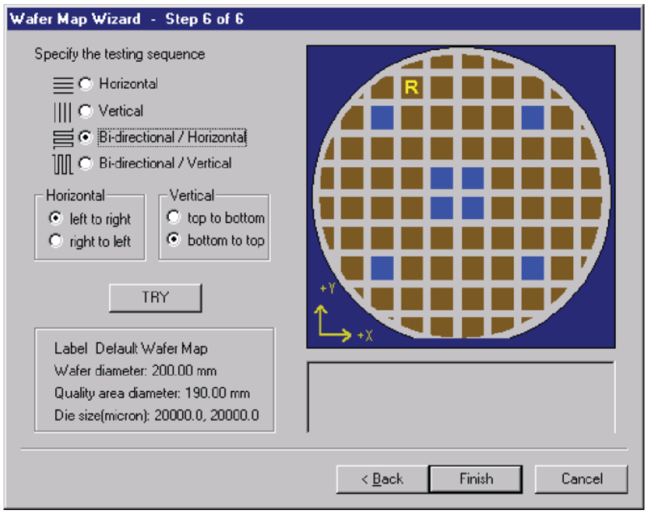


NOTE

Refer to the `probesites` and `probesubsites` Clarius project examples for specifics on selecting sites to probe.

18. Select **Next**.
19. Specify the test sequence.

Figure 216: Step 6: Wafer Map Wizard



20. Select **Finish**.
21. Save the Wafer Map settings.

Create a site definition and define a probe list

Creating a site definition for a single subsite per die involves using the software to create a selection of dies to probe. If a single subsite per site (die) is to be probed, refer to Probesites Clarius Project example (on page 13-17).

Creating a site definition for multiple subsites per die involves using the software to create a selection of dies to probe, but also includes creating a selection of the subsites for each site (die) that will be probed. If multiple subsites per site will be probed, refer to Probesubsites Clarius Project example (on page 13-20).

For probers using the Velox prober control software

Velox prober control software:

On the probe station computer, refer to the Velox user manual and help content for information on how to add and edit sites (dies) and subsites (subdies).

Additionally, a test sequence must be specified. The 4200A-SCS operates by sending commands that cause the prober to move to the next site or subsite as determined by the test sequence.

NOTE

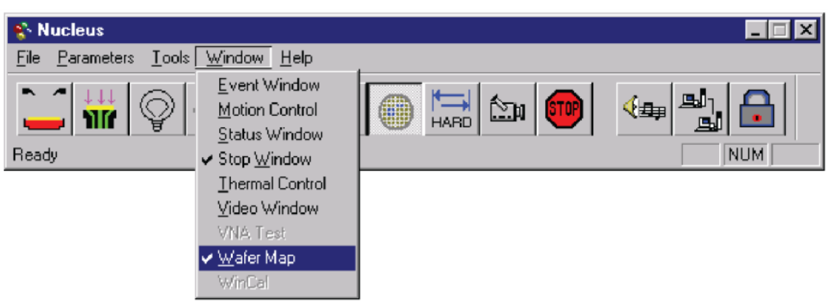
The 4200A-SCS is not aware of the specific sites or subsites present in the wafer map.

For probers using the Nucleus UI prober-control software

Nucleus UI prober-control software: Perform the following steps on the probe-station computer to open a previously defined site definition and probe list:

1. If the Nucleus toolbar is not already open, double-click the Nucleus UI icon on the Windows desktop.
2. Log in.
3. From the **Nucleus** toolbar, select **Tools > WaferMap**.
4. Select **Window > Wafer Map**. The Wafer Map window is displayed.

Figure 217: Nucleus toolbar



5. From the Wafer Map window, select **File > Open**.
6. Open the wafer map file.

Load, align, and contact the wafer

Velox prober-control software

For probers using the Velox prober-control software:

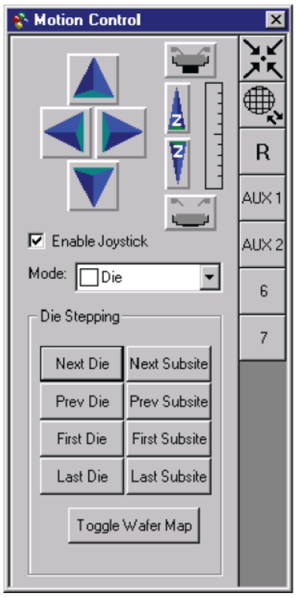
On the probe station computer, refer to the Velox user manual and help content for information on how to load, unload, set chuck heights, align, contact, and set the home (also called reference) position of the wafer.

Nucleus UI prober-control software

For probers using the Nucleus UI prober-control software:

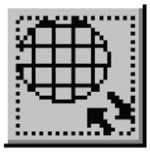
1. On the probe-station computer, from the Nucleus toolbar, select **Window > Motion Control**. The Motion Control dialog opens.

Figure 218: Motion Control dialog



2. From the Motion Control dialog, select the **Chuck to front** button.

Figure 219: Chuck to front button

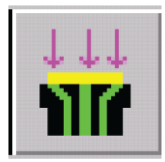


3. From the Nucleus toolbar, select the **Enable Joystick** button.

Figure 220: Enable Joystick button

4. Place a wafer on the chuck.
5. From the Nucleus UI toolbar, toggle the vacuum from **OFF** to **ON**.

Figure 221: Vacuum control



6. From the Nucleus UI toolbar, turn on the camera screen by selecting the **Video** button.

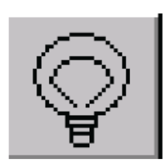
Figure 222: Video button

NOTE

If the LIGHT is off, the video will be blank.

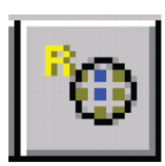
7. From the Nucleus UI toolbar, turn on the light by selecting the **Light** button.

Figure 223: Light button



8. From the Wafer Map dialog, select the **Reference Die** button. The Align dialog opens.

Figure 224: Reference Die button



9. From the Align dialog, select **Move to calculated position**.
10. Select **OK**.
11. Manually move the wafer to the reference die. A confirmation dialog is displayed.
12. Select **Yes** to set the reference die to the present position. When choosing the reference die:
    - The wafer should be on the chuck and physically in the correct reference position.
    - Select the die on the wafer map UI that will be the reference die.
    - An R appears when a die has been selected as the home die.
13. From the Nucleus UI toolbar, select the **Hard Align** button to display the Hard Align dialog.

Figure 225: Hard Align button

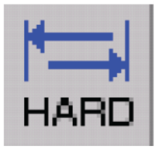
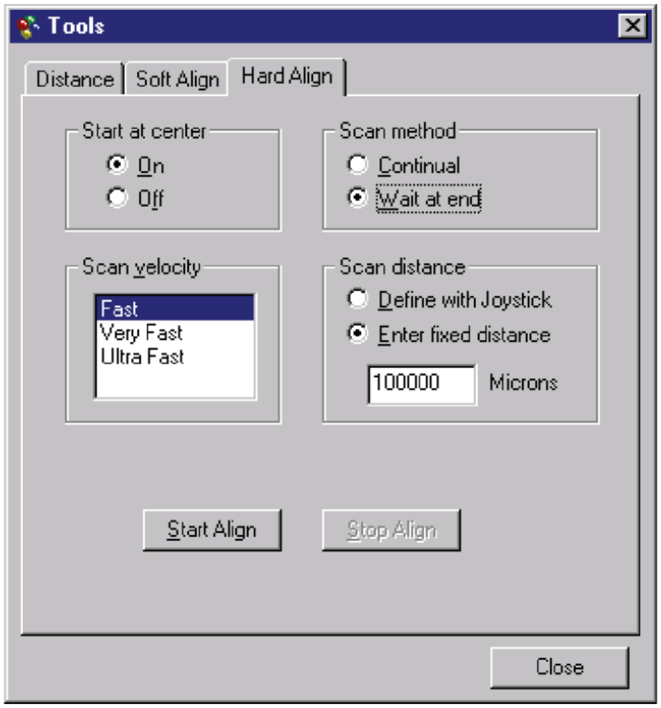


Figure 226: Hard Align tab



14. For **Start at center**, select **On**.
15. For **Scan method**, select **Wait at end**.
16. Set the **Scan velocity**.
17. Set the **Scan distance**. You can enter a fixed distance or define it with the joystick.

Align the wafer

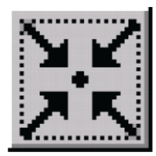
To align the wafer:

1. Move to wafer center by selecting the **Center** button on the Motion Control window.
2. Select **Start Align** on the Hard Align dialog.

NOTE

Raise the platen arm if prompted (a prompt only appears if the platen arm is down when you start the alignment).

Figure 227: Center button



3. Watch on the monitor while the stage moves down the street to position the needles near the left edge of the wafer.
4. Adjust the theta knob on the stage while moving across the wafer.
5. Select **Yes** at the prompt that appears on the screen.
6. Watch on the monitor and continue to adjust theta while moving down the street to position the needles near the right edge of the wafer.
7. Make a small adjustment in theta when motion stops.
8. Select **No** when the alignment is correct.
9. Set the contact position (set the current Z as contact position).

NOTE

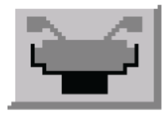
The Z contact position is the specified point where probe needles contact the wafer when using the Raise/Lower button. The Raise/Lower button is on the left side of the Nucleus toolbar. Select the button to toggle to the make-contact or break-contact position.

NOTE

Good contact occurs when the probe tips contact the probe pad, accounting for the tolerances of the probe needles and wafer plus any additional overdrive. Overdrive is the additional Z motion of the probe needles relative to the wafer after the initial contact. Overdrive ensures tolerable contact resistance by causing the probe tips to scrub through test pad surface oxide.

10. Either using the **Z Up/Z Down** buttons on the Motion Control window, or the joystick if set for Scan Z Axis (see **CAUTION**), make contact with the wafer.
11. When probe tips are making good contact with the wafer, right-click the **Contact** button.
12. Select the **Set to Current Position** button.

Figure 228: Set to Current Position button



CAUTION

When the Joystick mode is set to "Scan Z Axis," the joystick will control Z movement. While in this mode, the prober beeps providing an audible alert. When this alert is heard, care should be exercised when using the joystick for Z travel adjustments. Avoid damage to the probe needle or the wafer while changing the Z height.

The **Up/Down arrows** may be used to set Z contact. When using the arrows, travel is fast (coarse adjustment) when away from the Z contact position, and slow (fine adjustment) when close to the Z contact position.

When setting the Z contact, the camera stays focused on the probe needles (not on the wafer).

Probesites Clarius Project example

The following is a step-by-step procedure to configure the Summit 120000 so the `probesites` Clarius project executes successfully.

Nucleus UI or Velox software

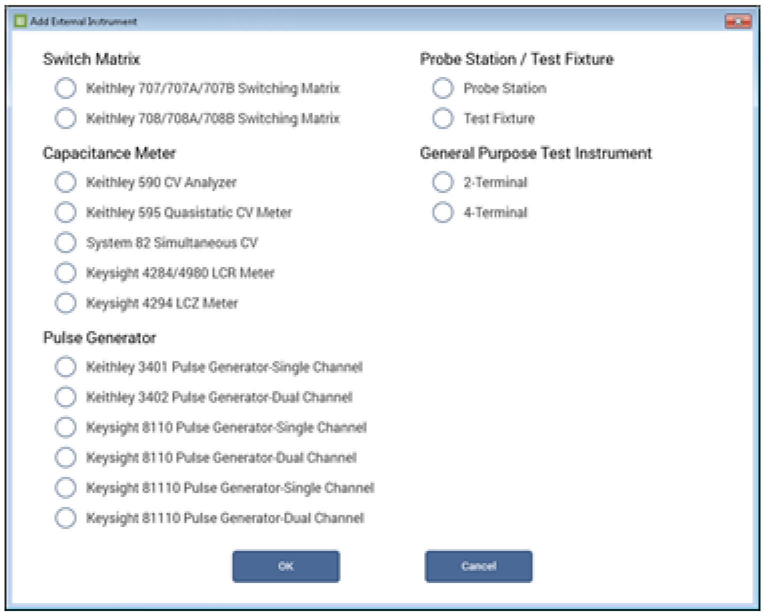
Using either the Velox software or the Nucleus UI software on the probe station computer, edit and open a wafer map file as described in Set up wafer geometry (on page 13-8) and Create a site definition and define a probe list (on page 13-11).

Use KCon to add a prober

On the 4200A-SCS, use KCon to add the prober to the configuration:

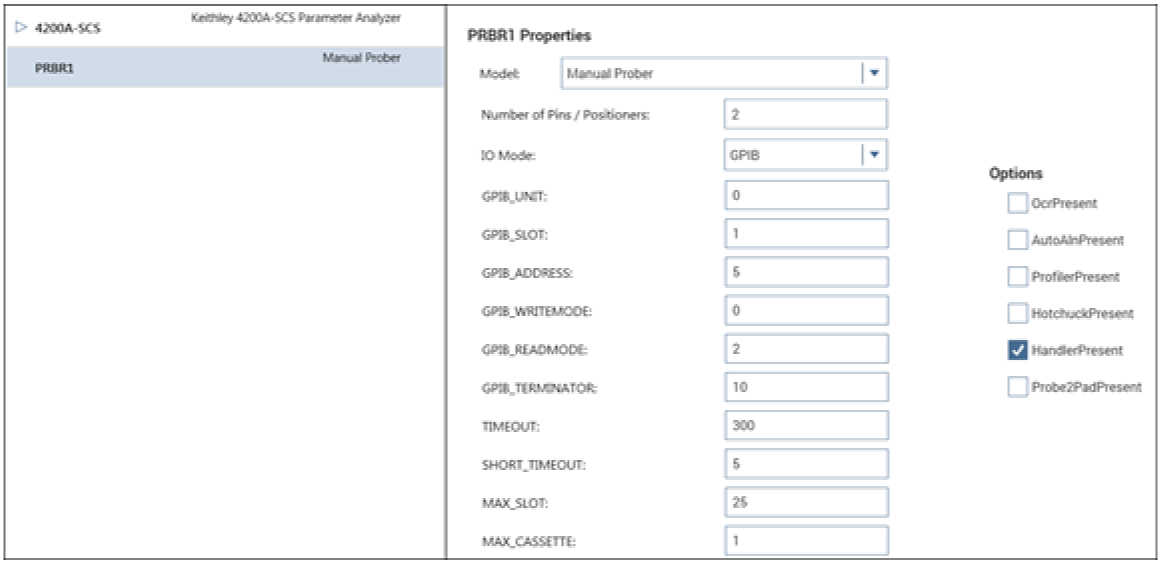
1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

Figure 229: Add a prober in KCon



3. Select **Probe Station**.
4. Select **OK**. KCon displays the properties for the prober.

Figure 230: Use KCon to select a prober



5. Select the Cascade 1200 prober as the model.
6. Make sure the **Number of Pins / Positioners** is correct. The number of pins defined here determines the pins that are available to assign to a switching matrix card column.
7. Verify the IO Mode is set to **GPIB.**
8. Verify the **GPIB_ADDRESS** is set to the address of the prober. This address was set in the section Set up communications (on page 14-2). The default address is 28.
9. Select **Save**.
10. Exit KCon.

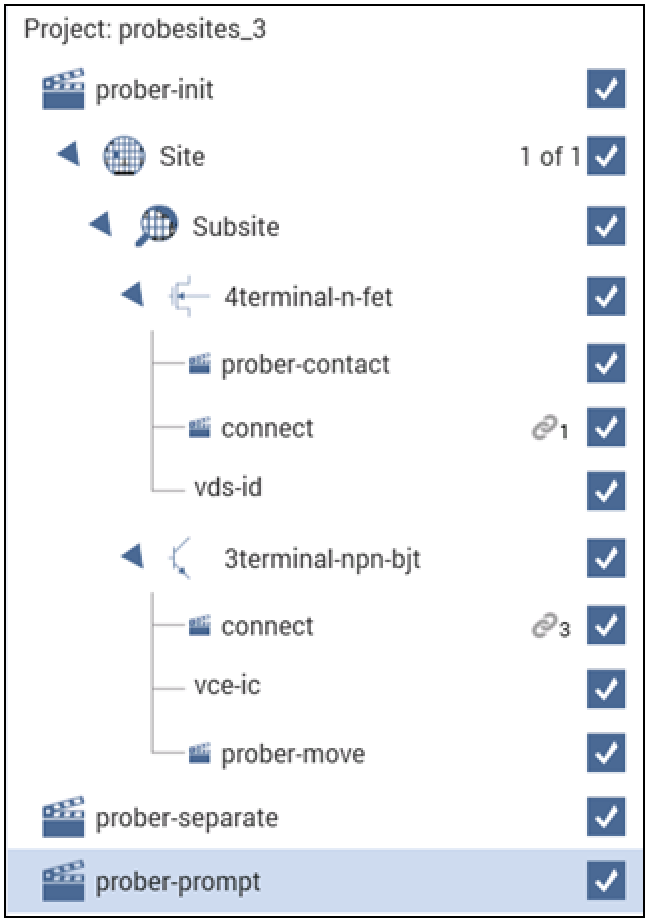
Clarius

Use Clarius to load and run the `probesites` project using the new KCon configuration file. The configuration file allows you to execute a project for this prober.

On the 4200A-SCS:

1. Open Clarius.
2. Choose **Select**.
3. Select **Projects**.
4. Search for **probe**.
5. Drag the **probesites** project to the project tree.

Figure 231: probesites project tree



6. Select **Run**.

Probesubsites Clarius Project example

The following is a step-by-step procedure to configure the Summit-12000 so the `probesubsites` project executes successfully.

Velox prober control software

For probers using the Velox prober control software:

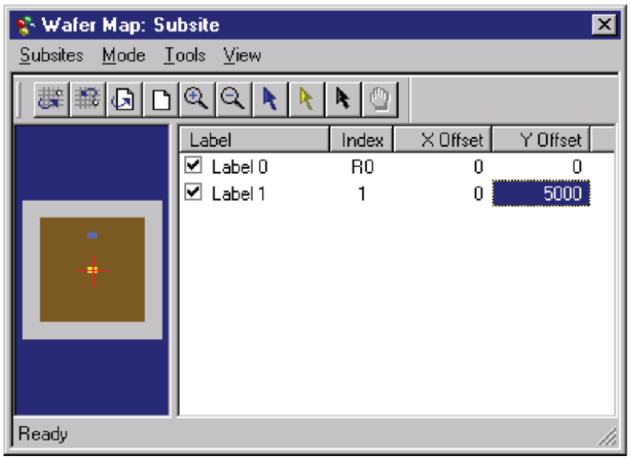
On the probe station computer, refer to the Velox user manual and help content for information on setting up and loading a wafer map that contains sites (dies) and subsites (subdies). A test sequence also needs to be specified.

Nucleus UI prober control software

For probers using the Nucleus UI prober control software:

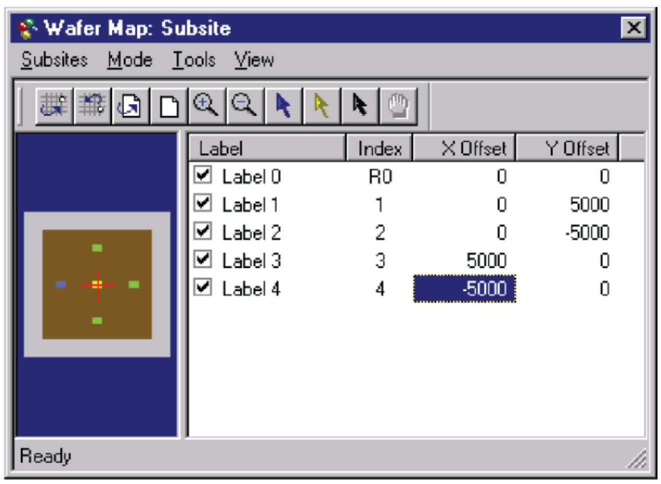
1. On the probe-station computer, if the Nucleus toolbar is not already open, double-click the **Nucleus** icon on the Windows desktop.
2. Log in.
3. From the Window menu of the Nucleus toolbar, select **WaferMap** to display the wafer map window.
4. From the Wafer Map window, select **File > Open** to open a wafer map file.
5. Select **Wafer > Sub Die** from the Wafer Map menu. A subsite dialog opens, as shown in the following figure.
6. Select **Subsites > New Subsite** to create a new subsite Label 1.
7. Enter the corresponding X and Y offset of the new subsite.

Figure 232: Enter x and y offset



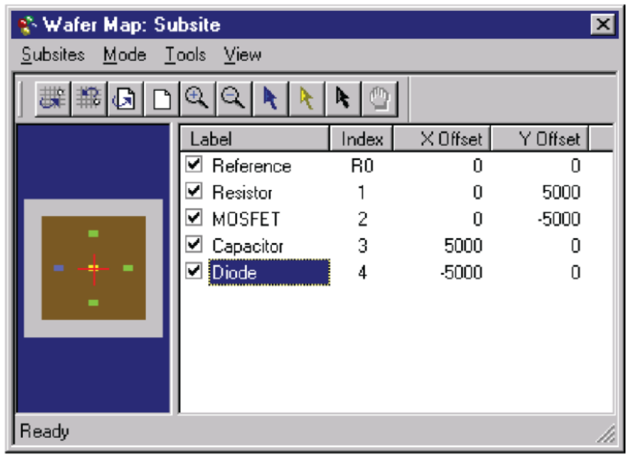
8. Continue to add new subsites until finished.

Figure 233: Make four new subsites



9. Select the label name and type in a new description to relabel each subsite.

Figure 234: Relabel the subsites



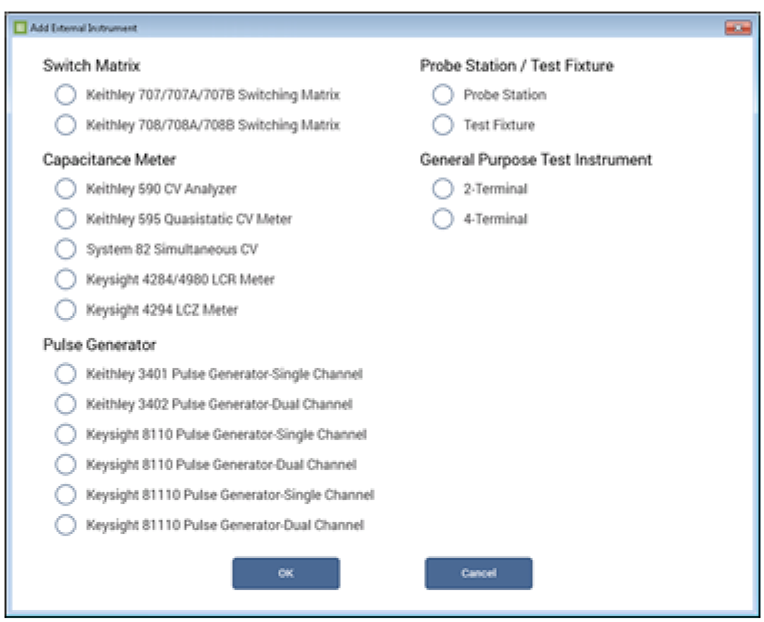
10. To choose a subsite for testing, select the box at the front of each label. To skip testing the subsite, clear the box at the front of each label.
11. Select **File > Save** on the Wafer Map dialog to save the wafer map.

Use KCon to add a prober

On the 4200A-SCS, use KCon to add the prober to the configuration:

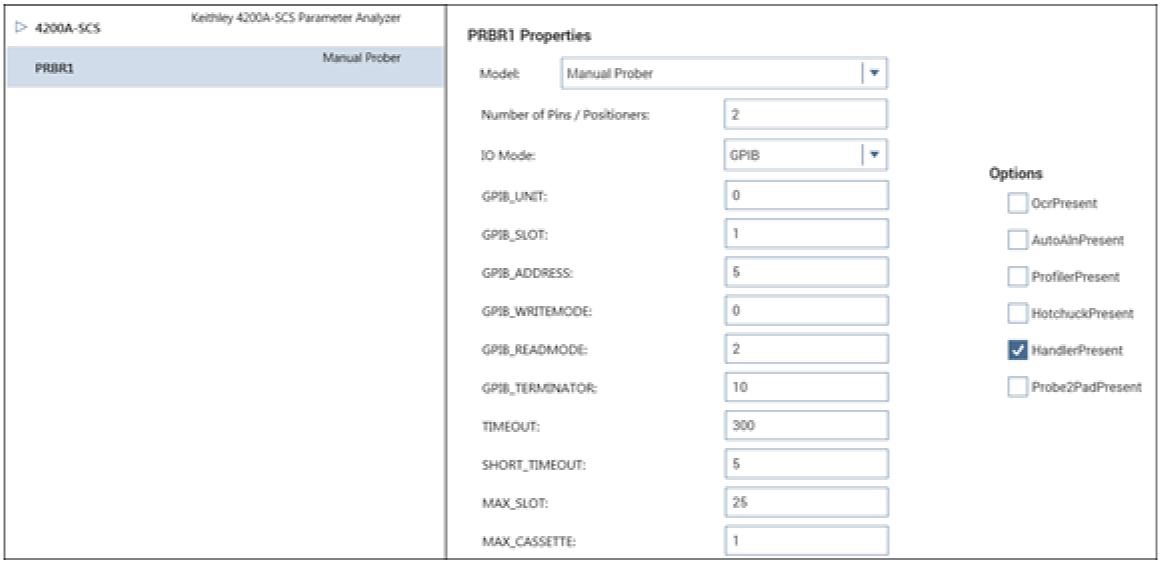
1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

Figure 235: Add a prober in KCon



3. Select **Probe Station**.
4. Select **OK**. KCon displays the properties for the prober.

Figure 236: Use KCon to select a prober



5. Select the Cascade 1200 prober as the model.
6. Make sure the **Number of Pins / Positioners** is correct. The number of pins defined here determines the pins that are available to assign to a switching matrix card column.
7. Verify the IO Mode is set to **GPIB.**
8. Verify the **GPIB_ADDRESS** is set to the address of the prober. This address was set in the section Set up communications (on page 14-2). The default address is 28.
9. Select **Save**.
10. Exit KCon.

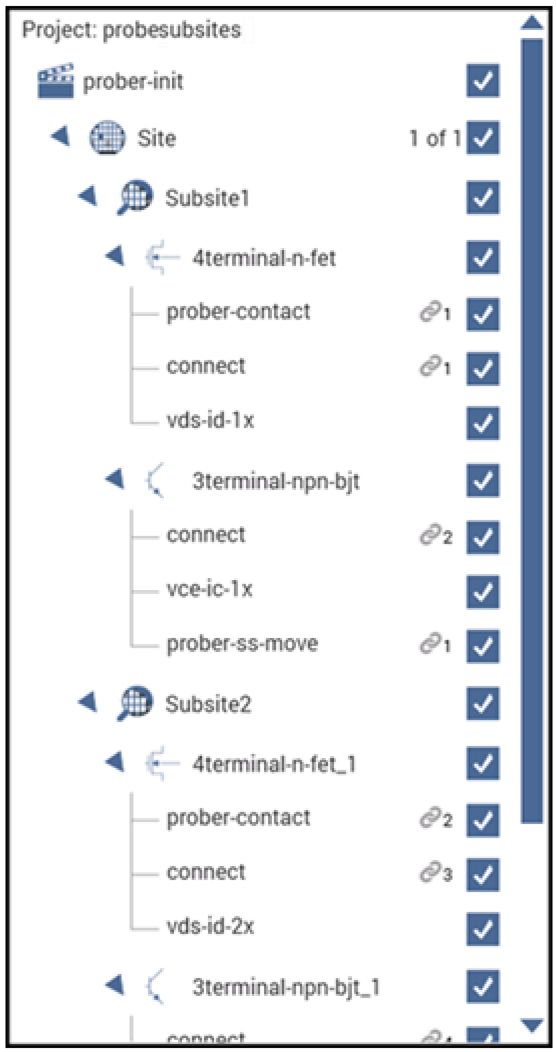
Clarius

Use Clarius to load and run the `probesites` or `probesubsites` project using the new KCon configuration file, which allows you to execute the project for this prober.

On the 4200A-SCS:

1. Open Clarius.
2. Choose **Select**.
3. Select **Projects**.
4. Search for **probesubsites**.
5. Drag the **probesubsites** project to the project tree.

Figure 237: probesubsites project tree



6. Select **Run**.

Commands and error symbols

The following table contains error and status symbols listed by command.

Available commands and responses

	PrChuck	PrInit	PrMovNxt	PrSSMovNxt
PR_OK	X	X	X	X
BAD_CHUCK	X			
INVAL_MODE	X			
UNINTEL_RESP	X	X	X	X
INVAL_PARAM		X		
BAD_MODE		X	X	X
PR_WAFERCOMPLETE			X	X

Information and error code return values and descriptions

Value	Constant	Explanation
1	PR_OK	Success (OK)
4	PR_WAFERCOMPLETE	Next wafer loaded (confirmed)
-1008	INVAL_MODE	Invalid mode
-1011	BAD_MODE	Operation invalid in mode
-1013	UNINTEL_RESP	Unintelligible response
-1017	BAD_CHUCK	Bad chuck position
-1027	INVAL_PARAM	Invalid parameter

Using a Signatone CM500 Prober

In this section:

Signatone CM500 prober software

This section describes set up for the Signatone CM500 (WL250) prober. The CM500 driver provided with the Keithley 4200A-SCS also works with other Signatone probers with Interlink controllers, such as the WL250 and S460SE. The name CM500 used in the configuration and setup in this documentation applies to all Signatone semi-auto prober systems with Interlink controllers.

Software versions

The following software versions on the CM500 prober was used to verify the configuration of the prober with the 4200A-SCS:

- CM500.exe version 2.5
- For the S460-SE prober: S460SE.exe version 2.5

Probe station configuration

CAUTION

Refer to the Signatone CM500 or S460 Prober supporting documentation before attempting setup, configuration, or operation.

The general steps required to set up and configure the CM500 or S460 prober for use with the 4200A-SCS are:

- Set up communications (on page 14-2)
- Set up wafer geometry (on page 14-4)
- Load, align, and contact the wafer (on page 14-6)
- Set up programmed sites without a subsite (on page 14-9)
- Set up programmed sites with a subsite (on page 14-11)

Set up communications

The Signatone CM500 prober is configured for GPIB communications only. Make sure the prober configuration is set up properly for the GPIB communications interface.

To set up communications:

1. Double-click the **CM500** icon on the Windows desktop. The prober initializes the wafer XY stage, theta, and Z chuck.
2. Select the **Utility** menu and select **Remote Host Interface**. The Set Host Interface dialog is displayed.

Figure 238: CM500 Utility menu

3. Select **IEEE488** (GPIB). The Signatone GPIB driver window opens.

Figure 239: Signatone GPIB driver window

4. Select **Addr** and verify that the GPIB address matches the `GPIB_Address` setting in the 4200A-SCS prober configuration file `prbcnfg_CM500.dat` at `C:\s4200\sys\dat`. The default GPIB address is 28.
5. If the address does not match, enter the new GPIB address, then select **OK**.

Modify the prober configuration file

The default prober configuration file is shown in the following code example. As shown, the file is configured for use with a GPIB communications setup. Use a text editor such as Microsoft™ Notepad to work with this file if needed.

On the 4200A-SCS, the configuration file is at `C:\s4200\sys\dat\prbcnfg_CM500.dat`.

```
# prbcnfg_CM500.dat - DEFAULT Prober Configuration File
#
# The following tag, "PRBCNFG", is used by the engine in order to determine
# the MAX number of SLOTS and CASSETTES for a given prober at runtime.
#
<PRBCNFG>
#
# for OPTIONS "" == NULL, max 32 chars in string
#
# Example
#          01234567890
#PROBER_1_OPTIONS=1,1,1,1,1,1
#
#
#   OcrPresent
#   AutoAlnPresent
#   ProfilerPresent
#   HotchuckPresent
#   HandlerPresent
#   Probe2PadPresent
#
#
# Configuration for direct GPIB probers:
# CM500
#
```

```
PROBER_1_PROBTYPE=CM500
PROBER_1_OPTIONS=0,0,0,0,1,0
PROBER_1_IO_MODE=GPIB
PROBER_1_GPIB_UNIT=0
PROBER_1_GPIB_SLOT=1
PROBER_1_GPIB_ADDRESS=28
PROBER_1_GPIB_WRITEMODE=0
PROBER_1_GPIB_READMODE=2
PROBER_1_GPIB_TERMINATOR=10
PROBER_1_TIMEOUT=300
PROBER_1_SHORT_TIMEOUT=5
PROBER_1_MAX_SLOT=25
PROBER_1_MAX_CASSETTE=1
#
#
```

Set up wafer geometry

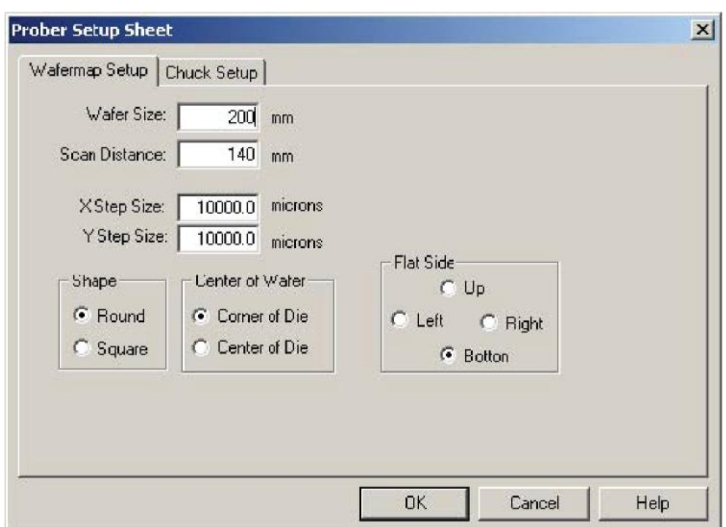
To set up wafer geometry:

1. Select the **Prober Setup** icon on the toolbar.

Figure 240: CM500 Prober Setup icon

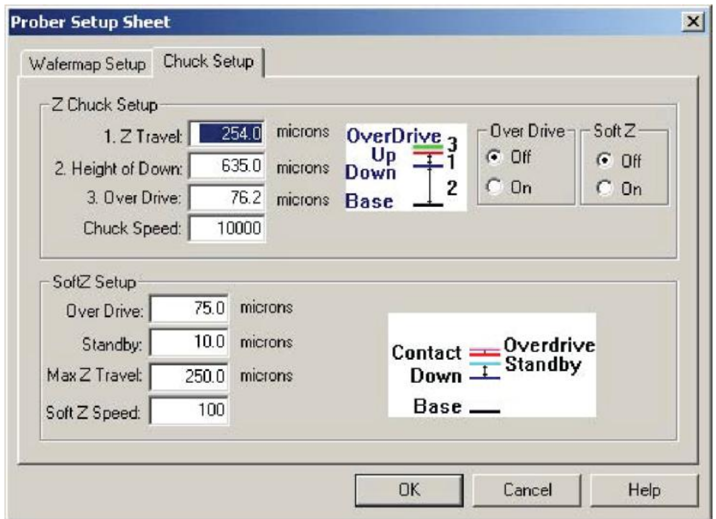
2. Select **Wafermap Setup** tab to set up wafer information, such as wafer size, scan distance, X step size, and Y step size.

Figure 241: CM500 Prober Setup Sheet



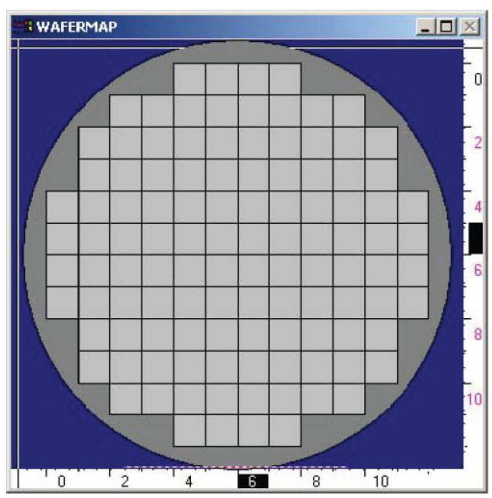
3. Select the **Chuck Setup** tab to enter Z chuck information, such as Z travel and overdrive distance.

Figure 242: CM500 Chuck Setup Sheet



4. After selecting **OK**, a new wafermap is displayed.

Figure 243: CM500 Prober wafermap



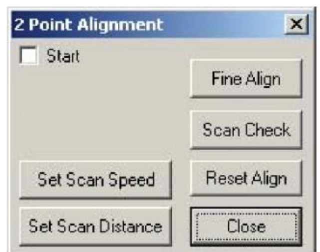
Load, align, and contact the wafer

1. Select the **Load wafer** icon on toolbar.

Figure 244: CM500 Prober load wafer icon

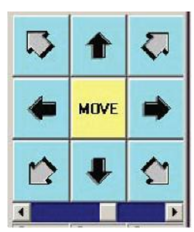
2. Select **Start** to move the wafer to Home and begin the sequences of 2-point alignment.

Figure 245: CM500 Prober 2 Point Alignment 1



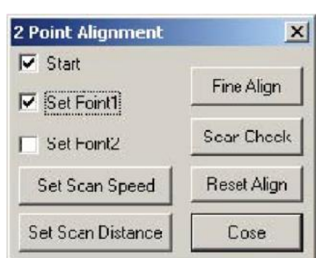
3. Select the **Arrow** buttons on the window to move the wafer stage to reference point 1.

Figure 246: CM500 Prober manual MOVE buttons



4. Select **Set Point1**.

Figure 247: CM500 Prober 2 Point Alignment 3



5. The Wafer stage moves to the other side as set by the scan distance.
6. Select the **Arrow** buttons on the window to move wafer stage to reference point 2.
7. Select **Set Point2**.
8. The Prober software rotates the theta motor for the proper alignment.
9. Select **Scan Check** to verify that the wafer is aligned correctly.
10. Select **Fine Align** to make a minor alignment.
11. After the wafer is aligned, set the HOME die of the wafer and wafermap.

Set the Home die of the wafer

Set the Home die of the wafer:

1. Move the wafer stage to the actual location that needs to be set as HOME.
2. When completed moving the wafer stage, select the **Set Home** icon on the toolbar.

Figure 248: CM500 Prober Set Home icon

Set the Home die of the wafermap

Set the Home die of the wafermap:

1. To set **HOME** on the wafermap, select the **Edit wafermap** icon on the toolbar.

Figure 249: CM500 Prober Editmap function icon

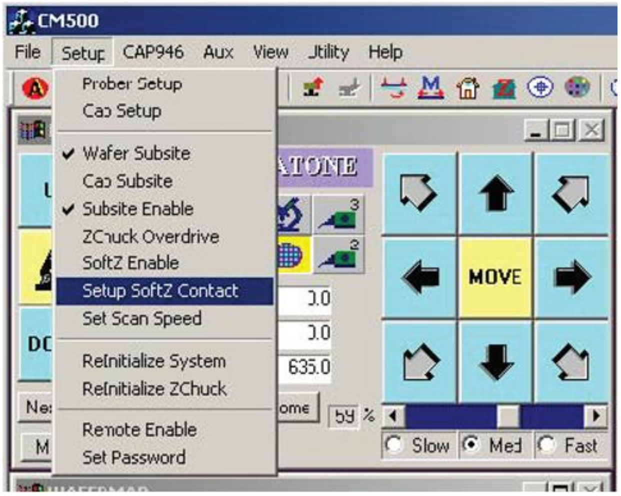
2. Select the **Set Home of Map** function.
3. Select the Home die on the wafermap that needs to be set as Home.
4. **Close** the Edit Map Function window.

Adjust the Z chuck

Adjust the Z chuck:

1. If an edge sense card is being used as the contact input for Z Chuck, you must select the **Setup SoftZ Contact** option from the **Setup** menu.

Figure 250: CM500 Prober Setup softz contact command

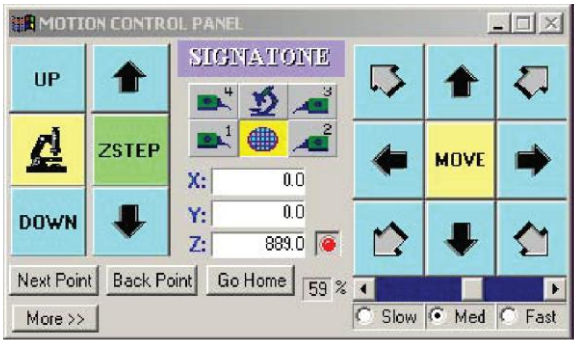


2. Follow the instructions on the window to adjust the height of platen and to determine the contact position of the Z Chuck.
3. Move the Z Chuck up to confirm contact condition using the **Contact** icon on the toolbar.

Figure 251: CM500 Prober Z Chuck Up (CONTACT) icon

4. If the edge sense is plugged in for contact input, turn **ON** SoftZ. A red LED will appear in the motion control panel.

Figure 252: CM500 Prober Motion Control Panel



5. Move the Z Chuck down using the **Separate** icon on the toolbar.

Figure 253: CM500 Prober Z Chuck Down (SEPARATE) icon

Set up programmed sites without a subsite

1. Select the **Program Site** icon on the toolbar.

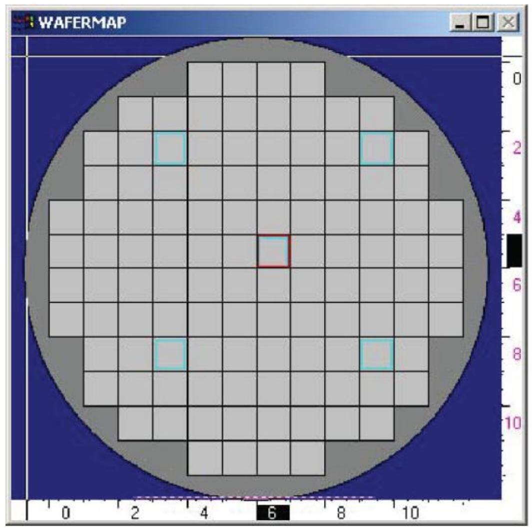
Figure 254: CM500 Prober Edit Program Sites icon

2. Select the **Enter Site Map** function.

Figure 255: CM500 Prober Edit Program Site window

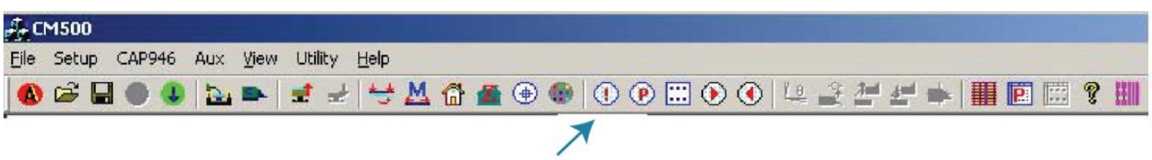
3. Move the mouse onto the WAFERMAP window and then do one of the following actions:
   - Select the dies to be tested on the wafermap and select **Enter**.
   - Select **Enter All** to test all dies.

Figure 256: CM500 Prober wafermap includes program sites



4. To step through all the programmed sites, select the **Run Program Site** icon on the toolbar.

Figure 257: CM500 Prober Run Program Sites icon



5. Select the **To First Site** button to move the prober to the first programmed site for testing. Make sure the Subsite (template) is disabled here.
6. In the **File** menu bar, select the **Save setup As** command to save the file to a hard disk. You can load this setup later to restore this setup if needed.

The Prober is now ready to accept a remote command from the 4200A-SCS.

Set up programmed sites with a subsite

1. Select the **Edit Subsite** icon on the toolbar.

Figure 258: CM500 Prober Edit Subsite icon

2. Select **Wafer** as the subsite device.
3. Move the wafer stage to the HOME position.

NOTE

All data recorded for the subsite is relative to the corner of the home die. You can record the position of the subsite either by keying in the coordinates of the subsite using the keyboard, or by moving the wafer to the actual position and selecting **Enter**.

4. To step through all the programmed sites and subsites, select the **Run Program Site** icon on the toolbar.
5. Make sure the Subsite (template) is **Enabled** if subsites are to be used.
6. Select the **To First Site** button to move wafer stage to first site of probing lists.

Figure 259: CM500 Prober Enable Subsite

7. In the **File** menu bar, select the **Save setup As** command to save the file to a hard disk. You can load this setup next time without going through all of the procedures again.
8. The Prober is now ready to accept a remote command from the 4200A-SCS.

Clarius project example for probe sites

The following is a step-by-step procedure to configure the Clarius project to execute testing and automatic wafer stepping to all programmed sites successfully. When the CM500 prober is connected to the 4200A-SCS by GPIB interface, the 4200A-SCS is the GPIB master controller and the CM500 is always in listening mode. The 4200A-SCS will send control commands to the CM500 to move the prober to next site during the automatic testing. The interface commands are `PrInit`, `PrChuck`, `PrMovNxt`, and `PrSSMovNxt`. You will need to add these commands into the Clarius project.

CM500

On the probe station computer, complete the procedures in the Probe station configuration (on page 14-2) section.

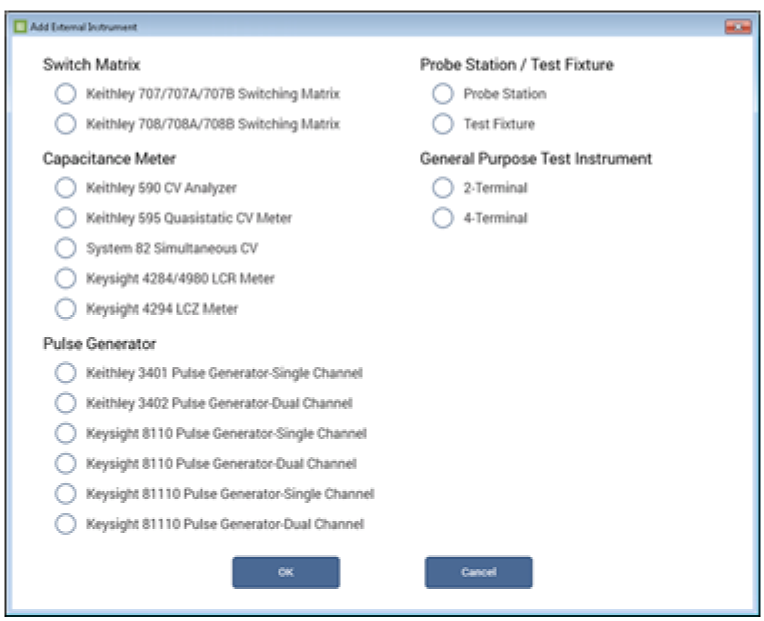
Use KCon to add a prober

You use KCon on the 4200A-SCS to add the prober to the configuration.

On the 4200A-SCS, use KCon to add the prober to the configuration:

1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

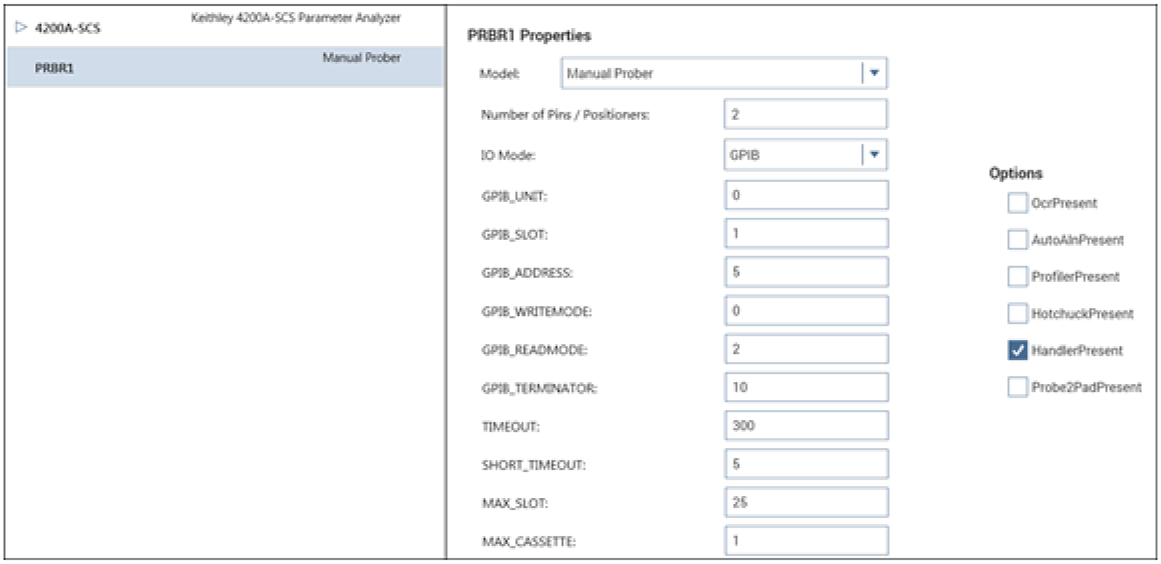
Figure 260: Add a prober in KCon



3. Select **Probe Station**.

4. Select **OK**. KCon displays the properties for the prober.

Figure 261: Use KCon to select a prober



5. For the Model, select the **Signatone CM500 (WL250) Prober**.
6. Make sure the **Number of Pins / Positioners** is correct. The number of pins defined here determines the pins that are available to assign to a switching matrix card column.
7. Save the configuration.
8. Exit KCon.

Clarius project example

To set up a new prober project:

1. Start **Clarius**.
2. Choose **Select**.
3. Select the **Projects** tab.
4. Drag **New Project** into the project tree.
5. Choose **Yes** to create a new project.
6. Rename the project **PRB_CM500**.
7. Select the **Wafer Plan** tab.
8. Drag **Site** to the project tree.
9. Select the **Actions** tab.
10. In the Search box, enter **prober**.
11. Drag the **Prober Initialization (prober-init)** action to the project tree. Make sure it is under the subsite.

Configure the prober project:

1. Select **Configure**. Make sure `prober-init` is selected in the project tree.

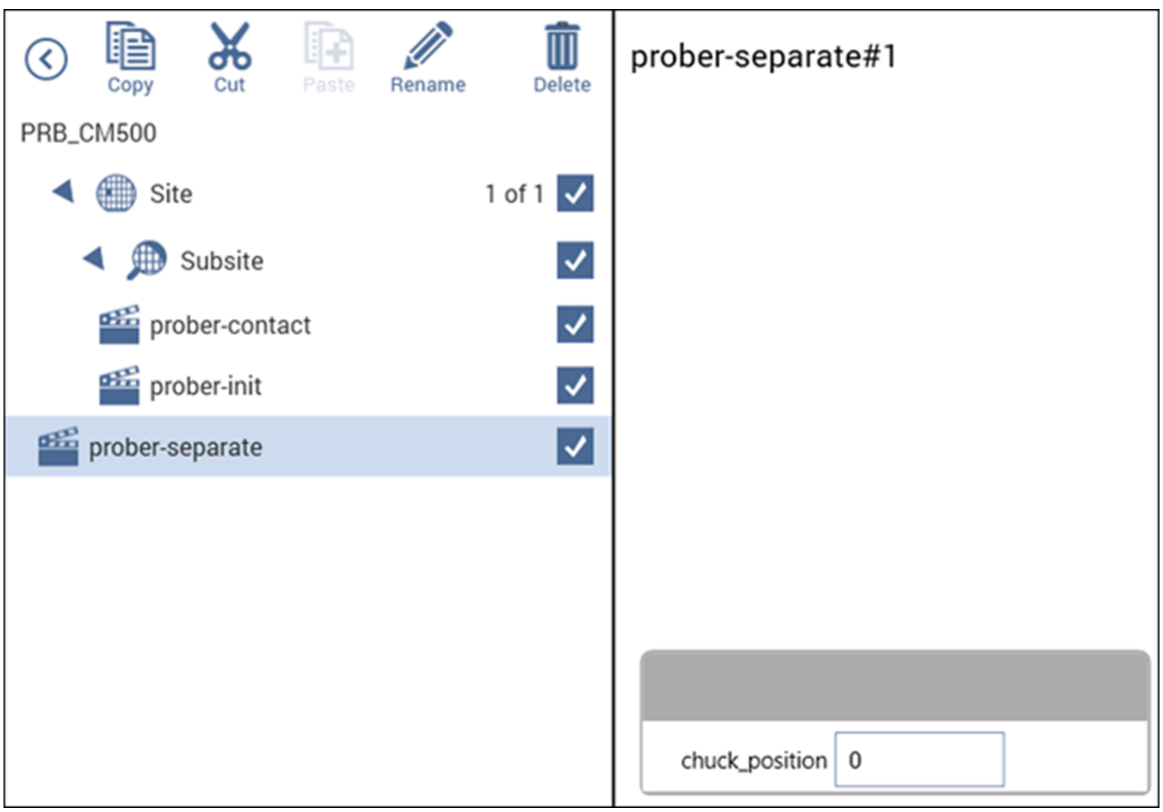
Figure 262: Set prober-init parameters

2. Set the **mode** to 6.
3. Set the **xdie_size** and **ydie_size** for your wafer.
4. Set units to either **0** for English or **1** for metric.
5. Check the **subprobtype**. If the CM500 prober is presently not at its first site, set **subprobtype** to **1**; otherwise, set it to **0**.

Set up actions:

1. In the project tree, select **Subsite**.
2. Choose **Select**.
3. Select the **Actions** library.
4. Add the **Prober Chuck Position (prober-contact)** action twice.
5. Select the **prober-contact_1** action.
6. Rename the action **prober-separate**.
7. Select **Configure**.
8. Set **chuckposition** to **0**. This moves the Z chuck to the down (separate) position.
9. Select the **prober-contact** action.
10. Set **chuckposition** to **1**. This moves the Z chuck to the contact position.
11. Place the **prober-separate** action at the bottom of the project tree.
12. Right-click **prober-separate** and select Promote Action twice so that `prober-separate` is at the site level.

Figure 263: New prober-separate UTM



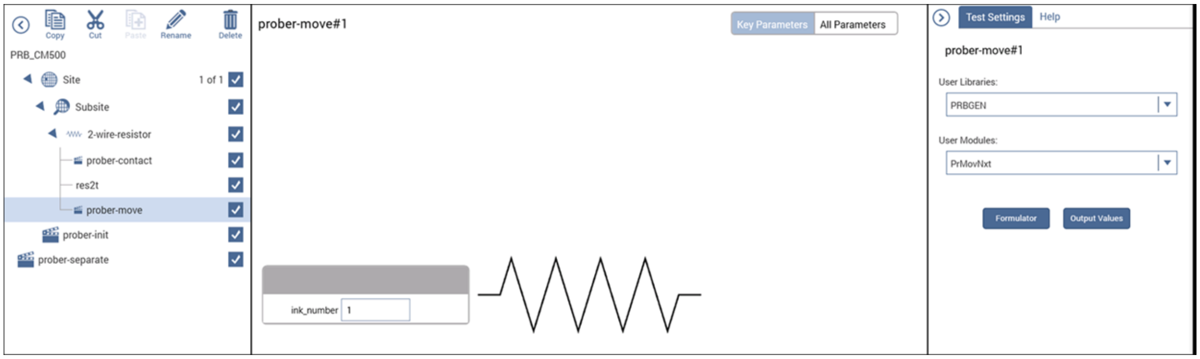
NOTE

The position of the action in the project tree determines when the action is run during a test. For example, in a device with multiple tests, you can run the device level directly. The tests under that device are executed sequentially. If an action is under the device level, the action runs in sequence with the tests. Similarly, actions under the subsite, site, or project levels execute automatically when the subsite, site, or project is run.

Create a test in the subsite level:

1. Choose **Select**.
2. Choose the **Tests** library.
3. Select a test for the device on your wafer.
4. Add the test to the subsite. When you add a test, an appropriate device is automatically added. You can also add a device from the Devices library and then add a test to the device.
5. Choose the **Actions** library.
6. Add the **Prober Move to Next Site (prober-move)** action.
7. Drag **prober-contact** so that it is immediately before your test.
8. Drag **prober-move** so that it is immediately after your test.
9. Select **Configure**.
10. For `prober-move`, set the **inknumber** to 1 if you need to trigger inker 1; otherwise, set it to 0.

Figure 264: prober-next in the project tree



Probesites Clarius project example

On the 4200A-SCS, use Clarius to open and run the `probesites` project using the new configuration file, which allows you to execute the project for this prober. This project uses a Series 700 Switching System and the `connect` action to change the instruments connected to each pin without changing the physical configuration.

To set up `probesites`:

1. In Clarius, choose **Select**.
2. Select the **Projects** library.
3. In the Search box, enter **probesites**.
4. Create the `probesites` project.
5. Select **Configure**.
6. Set the `prober-init` mode to **6**.
7. Set the **subprobtype**. If the CM500 prober is presently not at its first site, set the **subprobtype** to **1**; otherwise, set it to **0**.

Figure 265: Set prober-init mode parameters

8. In the project tree, select **probesites**.
9. Choose **Run** to execute the entire project.

Probesubsites Clarius project example

The following procedure configures a Clarius project to execute testing and automatic wafer stepping to all programmed subsites.

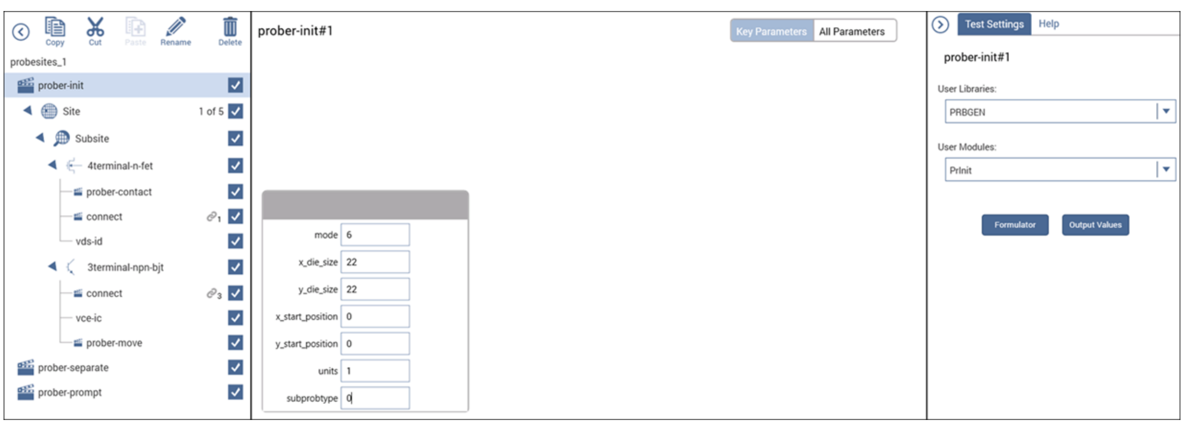
Use the 4200A-SCS to do this example. Use Clarius to open and run the `probesubsites` project using the new configuration file, which will allow you to execute the project for this prober.

This project uses a Series 700 Switching System and the `connect` actions to change the instruments connected to each pin without changing the physical configuration.

From Clarius:

1. Choose **Select**.
2. Select the **Projects** library.
3. Create the `probesubsites` project.
4. Select **Configure**.
5. In the project tree, select the **prober-init** action.
6. Set the **mode** to **6**.

Figure 266: Set prober-init mode parameters



7. Set the **subprobtype**. If the CM500 prober is not at its first site, set the `subprobtype` to **1**; otherwise, set it to **0**.
8. In the project tree, select **probesubsites_1**.
9. Select **Run** to execute the project.

Commands and error symbols

The following list contains error and status symbols listed by command.

Available commands and responses

	PrChuck	PrInit	PrMovNxt	PrSSMovNxt
PR_OK	X	X	X	X
BAD_CHUCK	X			
INVAL_MODE	X		X	X
UNINTEL_RESP	X	X	X	X
INVAL_PARAM		X		
BAD_MODE			X	X
PR_WAFERCOMPLETE			X	X
UNXPE_ERROR			X	X

Information and error code return values and descriptions

Value	Constant	Explanation
1	PR_OK	Success (OK)
4	PR_WAFERCOMPLETE	Next wafer loaded (confirmed)
-1008	INVAL_MODE	Invalid mode
-1011	BAD_MODE	Operation invalid in mode
-1013	UNINTEL_RESP	Unintelligible response
-1017	BAD_CHUCK	Bad chuck position
-1027	INVAL_PARAM	Invalid parameter

Using an MPI Probe Station

In this section:

MPI prober software

MPI supported probers include the TS2000, TS2000-DP, TS2000-HP, TS2000-SE, TS3000, and TS3000-SE.

To configure and operate one of the supported MPI probers with the Keithley 4200A-SCS, you need the MPI Sentio Software Suite application. This application provides access to configuration and help programs.

The MPI prober computer has MPI Sentio Software Suite installed. This is the main control software for the MPI prober. It provides the configuration and setup needed so that the prober can be controlled remotely using the 4200A-SCS.

Software version

The following software version was used to verify the configuration of the MPI probe station with the 4200A-SCS:

- MPI Sentio Software Suite version 2.9

Probe station configuration

CAUTION

Make sure that you are familiar with the MPI prober and its supporting documentation before attempting setup, configuration, or operation.

The general steps required to set up and configure the MPI prober for use with the 4200A-SCS include:

- Set up communications (on page 15-2)
- Load, align, and contact the wafer (on page 15-4)
- Set up wafer geometry (on page 15-4)
- Create a site definition and define a probe list (on page 15-4)

Set up communications

The MPI prober supports either GPIB or RS-232 communications to the 4200A-SCS. The following sections describe the steps to configure the prober and 4200A-SCS communications for either GPIB or RS-232.

Set up communications on the prober

The following steps describe how to set up the MPI prober for GPIB or RS-232 communications with a 4200A-SCS.

Set up the GPIB connection

To set up the GPIB connection:

1. Connect the MPI probers GPIB port to the 4200A-SCS GPIB port using a shielded GPIB cable.
2. Open the MPI Sentio configuration file, which is located on the MPI prober computer at: `C:\ProgramData\MPI Corporation\Sentio\config\config.xml`
3. Locate the communications configuration, which is in the node `Configuration / Main / RemoteServer`.
4. In the `RemoteServer` node, set the Type attribute to `GPIB`.

5. Set the Config attribute to *BoardName*:*BoardAddress*:*VenderCode*, where:
   - *BoardName* is the name of GPIB interface of the prober, such as GPIB0. Refer to the GPIB documentation to determine the name of the GPIB interface.
   - *BoardAddress* is the GPIB address of the MPI prober. This is an integer from 1 to 31. This address must be unique. You cannot use duplicate addresses on the same GPIB communications channel.
   - *VenderCode* is a string that identifies the vendor GPIB driver that will be used on the MPI prober. This must be either NI or ADLINK.

An example configuration for a GPIB card identified as GPIB0, with GPIB address 11, and that is a NI GPIB card is:

```
<RemoteServer Type="GPIB" Config="GPIB0:11:NI" />
```

Set up RS-232 communications

To set up the RS-232 connection:

1. Connect the COM port of the MPI probe station computer to the 4200A-SCS COM1 port using a DB9 socket to DB9 socket cable (shielded null modem cable).
2. Open the MPI Sentio configuration file, which is located on the MPI prober computer at: C:\ProgramData\MPI Corporation\Sentio\config\config.xml
3. Locate the communications configuration, which is in the node Configuration / Main / RemoteServer.
4. In the RemoteServer node, set the Type attribute to RS232.
5. Set the Config attribute to *ComPort*:*BaudRate*:*Parity*:*Handshake*, where:
   - *ComPort* is the name of the RS-232 COM port, such as COM1, that is being used on the MPI prober.
   - *BaudRate* is the baud rate of the selected COM port. Set to 9600.
   - *Parity* is the parity checking to be used. Set to NONE.
   - *Handshake* is the handshaking to be used. Set to OFF.

An example configuration of an R-S232 connection on COM1 with a baud rate of 9600, parity checking of none, and handshaking turned off is:

```
<RemoteServer Type="RS232" Config="COM1:9600:NONE:OFF" />
```

Set up communications on the 4200A-SCS

On the 4200A-SCS, KCon is used to add the MPI prober to the present system configuration and to edit the prober communications settings. Refer to Use KCon to add a prober (on page 15-5) for detailed information.

For more information on adding equipment to the 4200A-SCS, refer to "Keithley Configuration Utility (KCon)" in *Model 4200A-SCS Setup and Maintenance*.

Load, align, and contact the wafer

Refer to the *MPI Sentio User Manual* for information on how to load, unload, set chuck heights, align, contact, and set the home position of the wafer.

Set up wafer geometry

Refer to the *MPI Sentio User Manual* for information on how to set up the wafer map.

Create a site definition and define a probe list

Refer to the *MPI Sentio User Manual* for information on how to add and edit subsites and sites. To create a site definition for a single subsite for each die, you need to use the MPI software to create a selection of dies to probe.

To create a site definition for multiple subsites for each die, you need to use the MPI software to create a selection of dies to probe and create a selection of subsites on each die to be probed.

Clarius probesites and probesubsites project example

The following is a step-by-step procedure to configure an MPI prober so the `probesites` or `probesubsites` Clarius projects execute successfully.

MPI Sentio setup

Using MPI Sentio on the prober, edit and open a wafer map file. Refer to the MPI Sentio documentation to:

- Load, align, and contact the wafer
- Set up wafer geometry
- Create a site definition
- Define a probe list

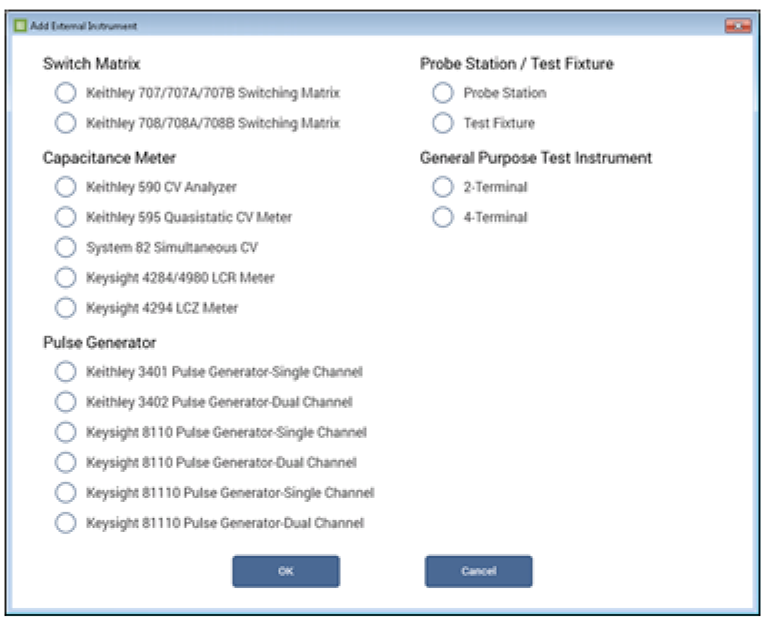
The wafer map file allows Clarius to send commands to instruct MPI Sentio to move the prober to the next site or the next subsite.

Use KCon to add a prober

On the 4200A-SCS, use KCon to add the prober to the configuration:

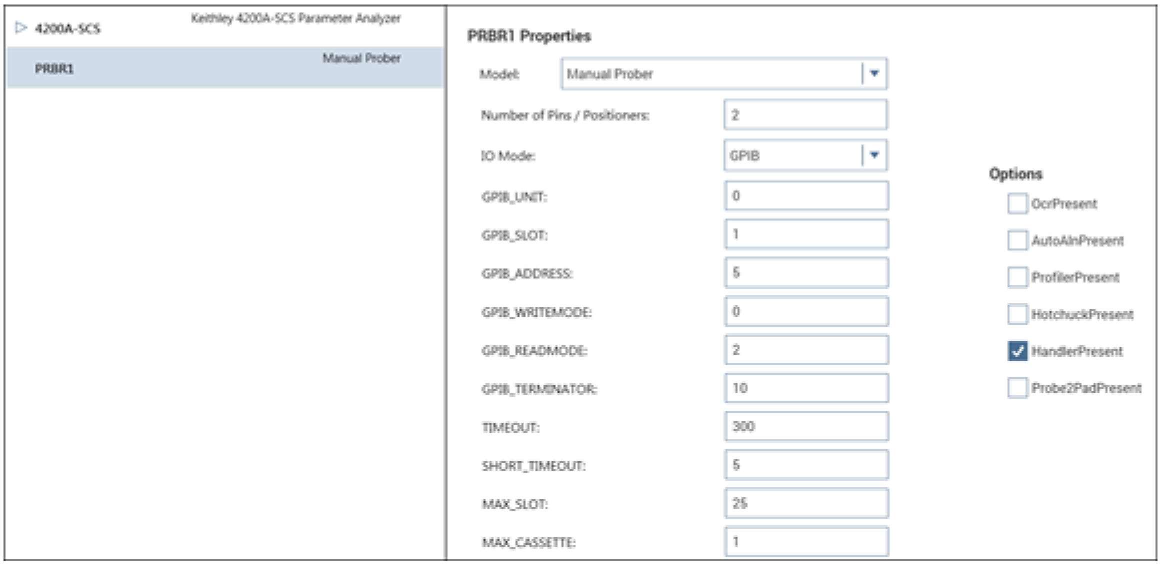
1. Open KCon.
2. At the bottom of the System Configuration list, select **Add External Instrument**. The Add External Instrument dialog is displayed, as shown in the following figure.

Figure 267: Add a prober in KCon



3. Select **Probe Station**.
4. Select **OK**. KCon displays the properties for the prober.

Figure 268: Use KCon to select a prober



5. Select **MPI Prober** as the model.
6. If you are using a switching matrix, make sure the **Number of Pins / Positioners** is correct. The number of pins defined here determines the pins that are available to assign to a switching matrix card column.
7. Set the **IO Mode** parameter to the type of communications being used with the prober, either GPIB or RS-232 (Serial).
8. If using GPIB, be sure to set the **GPIB_ADDRESS** parameter to the address of the MPI prober.
9. If using RS-232, make sure the **BAUDRATE** parameter is set to the same speed as the MPI prober COM port, typically 9600.
10. Select **Save**.
11. Exit KCon.

Clarius

Use Clarius to load and run either the `probesites` or `probesubsites` project using the new MPI prober configuration.

On the 4200A-SCS:

1. Open Clarius.
2. Choose **Select**.
3. Select **Projects**.
4. Search for **probe**.
5. Drag the **probesites** project (if only one subsite is used) or **probesubsites** project (if more than one subsite is used) to the project tree.

Figure 269: probesites project tree

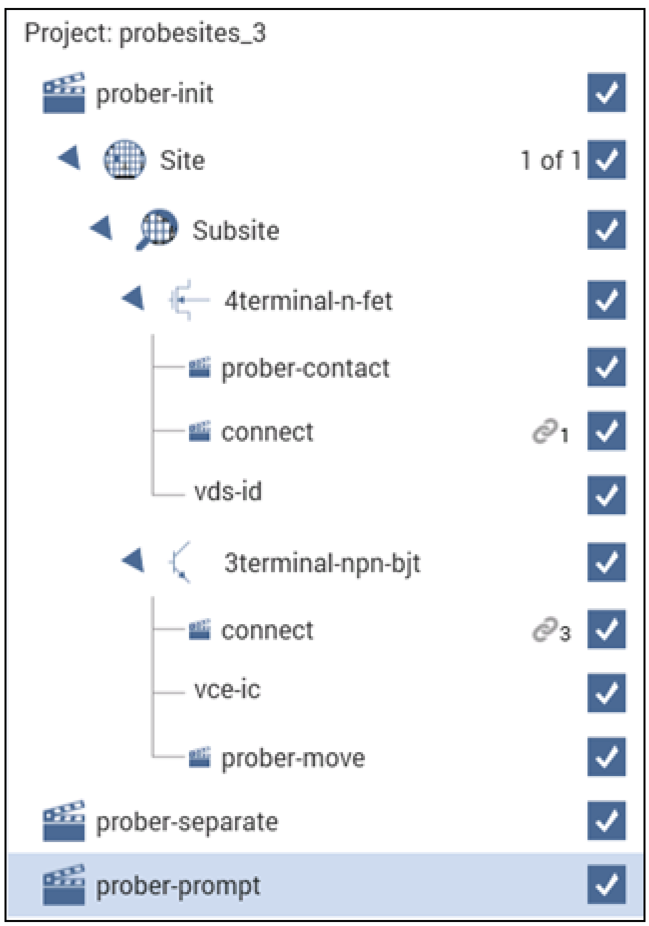
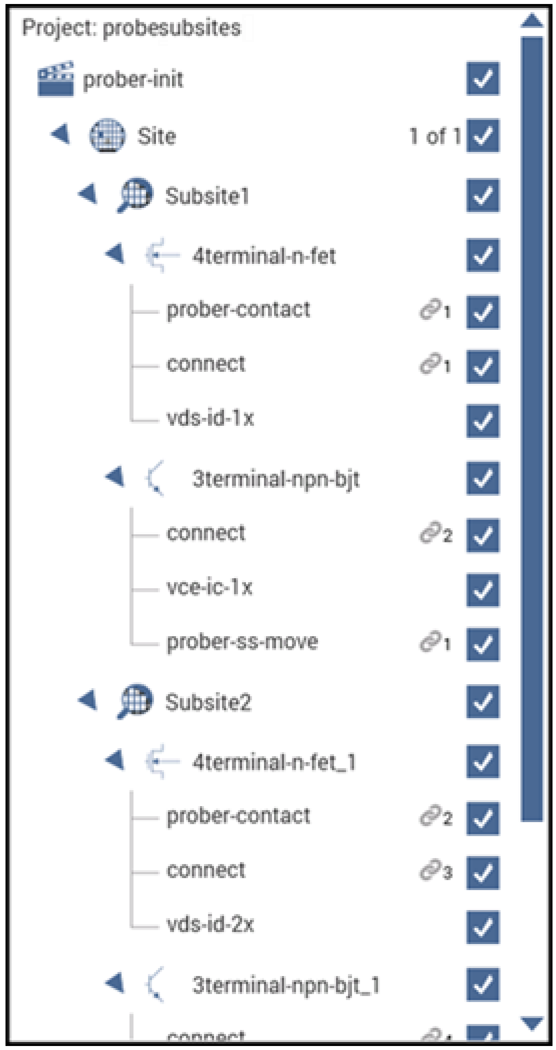


Figure 270: probesubsites project tree



6. Edit the project so that the number of sites or subsites matches the wafer map.
7. Delete or replace tests with relevant device tests for the wafer being tested. Refer to "Run a complex test" in the *Model 4200A-SCS Clarius User's Manual* for additional information on running a test from the project level.
8. Select project name in the project tree in Clarius.
9. Select **Run**.

Clarius initializes the MPI prober and runs through all sites and subsites in the project, sending the MPI prober commands as needed to either contact the chuck, step to the next site, or step to the next subsite.

Commands and error symbols

The following table contains error and status symbols listed by command when using the MPI prober through the PRBGEN user library.

Available commands and responses

	PrChuck	PrInit	PrMovNxt	PrSSMovNxt
PR_OK	X	X	X	X
BAD_CHUCK	X			
UNINTEL_RESP	X	X	X	X
INVAL_PARAM		X	X	X
BAD_MODE			X	X
UNEXPE_ERROR	X	X	X	X
MOVE_FAIL			X	X

Information and error code return values and descriptions

Value	Constant	Explanation
1	PR_OK	Command executed properly
-1011	BAD_MODE	Operation invalid in mode
-1013	UNINTEL_RESP	Unintelligible response
-1014	MOVE_FAIL	Movement failure
-1015	UNEXPE_ERROR	Unexpected error number
-1017	BAD_CHUCK	Bad chuck position
-1027	INVAL_PARAM	Invalid parameter

Keithley Instruments • 28775 Aurora Road • Cleveland, Ohio 44139 • 1-800-833-9200 • tek.com/keithley

KEITHLEY
A Tektronix Company

04/2022