Model 4200-SCS Semiconductor Characterization System

Applications Manual

4200-904-01 Rev. E / June 2008

ECA 44845

Model 4200-SCS

License Agreement

NOTICE TO USERS: CAREFULLY READ THE FOLLOWING LICENSE AGREEMENT (THE “AGREEMENT”). USE OF THE SOFTWARE (THE "SOFTWARE") PROVIDED WITH THE 4200-SEMICONDUCTOR CHARACTERIZATION SYSTEM (THE “4200-SCS”) CONSTITUTES YOUR ACCEPTANCE OF THESE TERMS. IF YOU DO NOT AGREE WITH THE TERMS OF THIS AGREEMENT, PROMPTLY RETURN THE SOFTWARE AND THE ACCOMPANYING ITEMS, INCLUDING ANY WRITTEN MATERIALS AND PACKAGING, TO THE LOCATION WHERE YOU OBTAINED THEM FOR A FULL REFUND.

Grant of License

Keithley Instruments ("Keithley") grants to you, subject to the terms and conditions of this Agreement, a non-exclusive, non-transferable license to use the portion of the Software developed and owned by Keithley (the “Keithley Software”) on the 4200-SCS and to use the manuals and other related materials pertaining to the Software which are necessary or desirable for the implementation, training or use of the Software (the “Documentation”) for your own internal business use and not for the benefit of any other person or entity. You may copy the Keithley Software into any machine-readable or printed form only for backup purposes or as necessary to use the Keithley Software or the 4200-SCS in accordance with this Agreement. The Keithley Software and Documentation and any copies or modifications thereof are referred to herein as the “Licensed Product.”

Ownership

Keithley and certain third party suppliers (the “Owners”) own all right, title and interest in and to the Licensed Product. You acknowledge that all right, title and interest in and to the Licensed Product will remain the exclusive property of the Owners, and you will not acquire any rights in or to the Licensed Product except as expressly set forth in this Agreement. The Licensed Product contains material that is protected by U.S. copyright laws, trade secret laws and international treaty provisions.

Limitations on Use

You may not make the Software available over the Internet or any similar networking technology. You may not remove any copyright, trademark or other proprietary notices from the Licensed Product or any media relating thereto. You agree that you will not attempt to reverse compile, reverse engineer, modify, translate, adapt or disassemble the Software, nor attempt to create the source code from the object code for the Software, in whole or in part.

Sublicense

You may sublicense the Keithley Software, subject to the sublicensee’s acceptance of the terms and conditions of this Agreement. You may not rent, lease or otherwise transfer the Licensed Product.

Termination

This Agreement is effective until terminated. Either party shall have the right to terminate this Agreement if the other fails to perform or observe any provision, term, covenant, warranty or condition of this Agreement (a “Default”) provided fifteen (15) days notice of termination (the “Notice”) is provided to the defaulting party and the defaulting party fails to cure the claimed Default within ten (10) days from the date of receipt of the Notice. Within three (3) days from the date of any termination of this Agreement, each and every embodiment of the Software in any form whatsoever, and all documentation, files and other materials in any form relating thereto, shall be destroyed, and all traces of the Software shall be permanently purged from the 4200-SCS.

Export Restrictions

You may not export or re-export the Software or any copy or adaptation in violation of any applicable laws or regulations.

U.S. Government Restricted Rights

Use, duplication and disclosure by the U.S. Government is subject to the restrictions as set forth in FAR §52.227-14 Alternates I, II and III (JUN 1987), FAR §52.227-19 (JUN 1987), and/or FAR §12.211/12.212 (Commercial Technical Data/Computer Software), and DFARS §252.227-7015 (NOV 1995) (Technical Data) and/or DFARS §227.7202 (Computer Software), as applicable.

Limited Warranty

Keithley does not warrant that operation of the Software will be uninterrupted or error-free or that the Software will be adequate for the customer's intended application or use. Keithley warrants to you that the Keithley Software will substantially perform in accordance with the specifications set forth in this manual for a period of ninety (90) days after your receipt of the Keithley Software (the “Warranty Period”); provided the Keithley Software is used on the products for which it is intended and in accordance with the Documentation. If the Keithley Software is not performing as warranted during the Warranty Period, as determined by Keithley in its sole discretion (a “Nonconformity”), your exclusive remedy under this limited warranty is either a correction of the Keithley Software or an explanation by Keithley of how to use the Keithley Software despite the Nonconformity, at Keithley’s option. The foregoing limited warranty shall be null and void upon any modification of the Software, unless approved in writing by Keithley. The portions of the Software not developed and owned by Keithley shall not be covered by this limited warranty, and Keithley shall have no duty or obligation to enforce any third party supplier’s warranties on your behalf. The failure to notify Keithley of a Nonconformity during the Warranty Period shall relieve Keithley of its obligations and liabilities under this limited warranty.

EXCEPT FOR THE FOREGOING, THE SOFTWARE IS PROVIDED “AS IS” WITHOUT ANY WARRANTY OF ANY KIND, INCLUDING BUT NOT LIMITED TO, THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

THE SOFTWARE IS NOT FAULT TOLERANT AND IS NOT DESIGNED OR INTENDED FOR USE IN HAZARDOUS ENVIRONMENTS REQUIRING FAIL-SAFE PERFORMANCE INCLUDING WITHOUT LIMITATION, IN THE OPERATION OF NUCLEAR

FACILITIES, AIRCRAFT NAVIGATION OR COMMUNICATION SYSTEMS, AIR TRAFFIC CONTROL, WEAPONS SYSTEMS, DIRECT LIFE-SUPPORT MACHINES, OR ANY OTHER APPLICATION IN WHICH THE FAILURE OF THE SOFTWARE COULD LEAD TO DEATH, PERSONAL INJURY OR SEVERE PHYSICAL OR PROPERTY DAMAGE (COLLECTIVELY "HAZARDOUS ACTIVITIES"). KEITHLEY EXPRESSLY DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY OF FITNESS FOR HAZARDOUS ACTIVITIES.

Limitation of Liability

KEITHLEY'S SOLE LIABILITY OR OBLIGATION UNDER THIS AGREEMENT IS SET FORTH ABOVE IN THE LIMITED WARRANTY SECTION OF THIS AGREEMENT. IN NO EVENT SHALL KEITHLEY BE LIABLE FOR ANY DAMAGES. WITHOUT LIMITING THE FOREGOING, KEITHLEY SHALL NOT BE LIABLE OR ASSUME LIABILITY FOR: (1) ECONOMICAL, INCIDENTAL, CONSEQUENTIAL, INDIRECT, SPECIAL, PUNITIVE OR EXEMPLARY DAMAGES, WHETHER CLAIMED UNDER CONTRACT, TORT OR ANY OTHER LEGAL THEORY, (2) LOSS OF OR DAMAGE TO YOUR DATA OR PROGRAMMING, (3) PENALTIES OR PENALTY CLAUSES OF ANY DESCRIPTION, OR (4) INDEMNIFICATION OF YOU OR OTHERS FOR COSTS, DAMAGES, OR EXPENSES RELATED TO THE GOODS OR SERVICES PROVIDED UNDER THIS LIMITED WARRANTY.

Miscellaneous

In the event of invalidity of any provision of this Agreement, the parties agree that such invalidity shall not affect the validity of the remaining portions of this Agreement. This Agreement shall be governed by and construed in accordance with the laws of the state of Ohio, without regard to conflicts of laws provisions thereof. This is the entire agreement between you and Keithley and supersedes any prior agreement or understanding, whether written or oral, relating to the subject matter of this license. Any waiver by either party of any provision of this Agreement shall not constitute or be deemed a subsequent waiver of that or any other provision.

Should you have any questions concerning this Agreement, or if you desire to contact Keithley Instruments for any reason, please call Keithley at 1-800-552-1115, or write at Keithley Instruments, 28775 Aurora Rd., Solon, Ohio, USA 44139.

Limited Hardware Warranty

Keithley warrants to you that the Keithley manufactured portion of the hardware (the "Keithley Hardware") purchased by you will substantially perform in accordance with the specifications set forth in this manual for a period of one (1) year after your receipt of the Keithley Hardware (the "Warranty Period"); provided the Keithley Hardware is used on the products for which it is intended and in accordance with the documentation. This limited warranty shall be null and void upon (1) any modifications of the Keithley Hardware, unless approved in writing by Keithley, (2) any operation of the 4200-Semiconductor Characterization System (the "4200-SCS") with third party software, unless the software is explicitly approved and supported by Keithley, and (3) any operation of the 4200-SCS on an operating system not explicitly approved and supported by Keithley.

If the Keithley Hardware is not performing as warranted during the Warranty Period, as determined in Keithley's sole discretion (a "Nonconformity"), your exclusive remedy under this limited warranty is the repair or replacement of the Keithley Hardware, at Keithley's option. The portions of the hardware not developed and owned by Keithley shall not be covered by this limited hardware warranty, and Keithley shall have no duty or obligation to enforce a third party supplier's warranties on your behalf. The failure to notify Keithley of a Nonconformity during the Warranty Period shall relieve Keithley of its obligations and liabilities under this limited hardware warranty.

EXCEPT FOR THE FOREGOING, THE HARDWARE IS PROVIDED "AS IS" WITHOUT ANY WARRANTY OF ANY KIND, INCLUDING BUT NOT LIMITED TO, THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

KEITHLEY'S SOLE LIABILITY OR OBLIGATION UNDER THIS LIMITED HARDWARE WARRANTY IS THE REPAIR OR REPLACEMENT OF THE KEITHLEY HARDWARE. IN NO EVENT SHALL KEITHLEY BE LIABLE FOR ANY DAMAGES. WITHOUT LIMITING THE FOREGOING, KEITHLEY SHALL NOT BE LIABLE OR ASSUME LIABILITY FOR: (1) ECONOMICAL, INCIDENTAL, CONSEQUENTIAL, INDIRECT, SPECIAL, PUNITIVE OR EXEMPLARY DAMAGES, WHETHER CLAIMED UNDER CONTRACT, TORT OR ANY OTHER LEGAL THEORY, (2) LOSS OF OR DAMAGE TO YOUR DATA OR PROGRAMMING, (3) PENALTIES OR PENALTY CLAUSES OF ANY DESCRIPTION, OR (4) INDEMNIFICATION OF YOU OR OTHERS FOR COSTS, DAMAGES OR EXPENSES RELATED TO THE GOODS OR SERVICES PROVIDED UNDER THIS LIMITED HARDWARE WARRANTY.

Should you have any questions concerning this Agreement, or if you desire to contact Keithley Instruments for any reason, please call 1-800-552-1115, or write at Keithley Instruments, 28775 Aurora Rd., Solon, Ohio, USA 44139.

KEITHLEY

A GREATER MEASURE OF CONFIDENCE

Keithley Instruments, Inc.

Corporate Headquarters • 28775 Aurora Road • Cleveland, Ohio 44139
440-248-0400 • Fax: 440-248-6168 • 1-888-KEITHLEY (534-8453) • www.keithley.com

03/07

4200-SCS
Semiconductor Characterization System
Applications Manual

Document Number: 4200-904-01 Rev. E / June 2008

KEITHLEY

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 non-hazardous voltages, there are situations where hazardous conditions may be present.

This product is intended for use by qualified 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 Instruments products are designed for use with electrical signals that are rated Measurement Category I and Measurement Category II, as described in the International Electrotechnical Commission (IEC) Standard IEC 60664. Most measurement, control, and data I/O signals are Measurement Category I and must not be directly connected to mains voltage or to voltage sources with high transient over-voltages. Measurement Category II connections require protection for high transient over-voltages often associated with local AC mains connections. Assume all measurement, control, and data I/O connections are for connection to Category I sources unless otherwise marked or described in the user documentation.

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 30V RMS, 42.4V peak, or 60VDC 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 1000V, 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.

The instrument and accessories must be used in accordance with its specifications and operating instructions, or the safety of the equipment may be impaired.

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

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 safety earth 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 safety earth ground using the wire recommended in the user documentation.

The  symbol on an instrument indicates that the user should refer to the operating instructions located in the user documentaion.

The  symbol on an instrument shows that it can source or measure 1000V or more, including the combined effect of normal and common mode voltages. Use standard safety precautions to avoid personal contact with these voltages.

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

The  symbol indicates a connection terminal to the equipment frame.

The **WARNING** heading in the user documentation explains dangers 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.

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 Instruments. Standard fuses with applicable national safety approvals may be used if the rating and type are the same. 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 Instruments to maintain accuracy and functionality of the product). If you are unsure about the applicability of a replacement component, call a Keithley Instruments office for information.

To clean an instrument, use a damp cloth 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.

Table of Contents

1 Graphical Data Analysis and Basic Test Sequencing

SubVt slope 1-2
- Open “default” project 1-2
- Open “subvt” test and display graph 1-2
- Line-fit analysis 1-3
- Modify the line-fit 1-4

Graphical analysis 1-5
- Open “default” project and “vds-id” test 1-5
- Display and analyze the “vds-id” graph 1-6

Sequencing tests on a single device 1-9
- Open “default” project 1-9
- Open “4terminal-n-fet” tests 1-9
- Modify tests 1-10
- Change the execution sequence 1-10
- Run “4terminal-n-fet” test sequence 1-11
- Save and export test data 1-12

2 Advanced Applications

Controlling a switch matrix 2-2
- KCON setup 2-4
- Open KITE and the “ivswitch” project 2-6
- Running test sequences 2-7
- “connect” test description 2-9

Sequencing tests on multiple devices 2-10
- Open “ivswitch” project 2-10
- Execute the test sequence (Subsite Plan) 2-12

Customizing a user test module (UTM) 2-12
- Open KULT 2-13
- Open the “ki42xxulib” user library 2-14
- Open the “Rdson42XX” user module 2-15
- Copy “Rdson42XX” to “RdsonAvg” 2-16
- Open and modify the “RdsonAvg” user module 2-17
- Save, compile, and build the modified library 2-19
- Add a new UTM to the “ivswitch” project 2-20
- Test description 2-21

3 Controlling External Equipment

Controlling external equipment overview 3-2
Controlling a CV Analyzer 3-5
- Connections 3-5

KCON setup 3-6
Create a new project 3-7
Add a Subsite Plan 3-8
Add a Device Plan 3-8
Add a UTM 3-9
Modifying the “cvsweep” UTM 3-11
Executing the test 3-11
Controlling a pulse generator 3-12
Test system connections 3-12
KCON setup 3-13
Open the “ivpgswitch” project 3-15
Description of tests 3-16
Running the test sequence 3-19
Compare the test results 3-19
Controlling a probe station 3-21
Prober control overview 3-22
Test system connections 3-23
KCON setup 3-23
Probe station configuration 3-26
Open the “probesubsites” project 3-26
Open the project plan window 3-27
Test descriptions 3-27
Running the test sequence 3-31
Test data 3-32
Running individual plans or tests 3-32

4 Pulse Applications

Charge Pumping 4-2
CP Procedure 4-2
Charge pumping UTM descriptions 4-6
amplsweep 4-6
basesweep 4-8
FallTimeLinearSweep 4-9
FreqFactorSweep 4-10
FreqLinearSweep 4-12
RiseTimeLinearSweep 4-13
Pulse IV 4-15
Introduction (PIV-A and PIV-Q) 4-15
Pulse IV for CMOS: Model 4200-PIV-A 4-16
PIV-A test connections 4-17
Using the PulseIV-Complete project for the first time 4-24
Pulse IV UTM descriptions 4-36
cal_pulseiv 4-36
vdsid_pulseiv 4-37
Vdid_Pulse_DC_Family_pulseiv 4-39
..... 4-42
Vgid_DC_Pulse_pulseiv 4-43
..... 4-46
vgsid_pulseiv 4-47
scopeshot_cal_pulseiv 4-49
..... 4-51
scopeshot_pulseiv 4-51
vdsid_pulseiv_demo 4-53
vgsid_pulseiv_demo 4-53
scopeshot_pulseiv_demo 4-53

Slow single pulse charge trapping high K gate stack 4-54
Charge trapping procedure 4-55
Charge Trapping UTM descriptions 4-58
chargetrapping_single_pulse_slow 4-58
AC stress for WLR 4-61
Q-Point Pulse IV – Model 4200-PIV-Q 4-64
What is the PIV-Q Package? 4-64
Target applications and test projects 4-64
PIV-Q Test Procedure 4-65
Interconnect Assembly Procedure 4-66
..... 4-69
Using the Model 4200 Project QPulseIV-Complete for the first time 4-70
Running AutocalScope 4-71
Running CableCompensation 4-71
PIV-Q user libraries 4-77
Pulse adapters, cables, hardware and PCU 4-118
Flash Memory Testing 4-119
Introduction 4-119
Theory of operation 4-119
FLASH Connections 4-132
Flash Projects 4-140

List of Figures

1 Graphical Data Analysis and Basic Test Sequencing

Figure 1-1 Graph for the “subvt” test 1-2
Figure 1-2 Open Formulator window 1-3
Figure 1-3 Formulator for “subvt” test 1-3
Figure 1-4 Start and stop points for the line-fit 1-4
Figure 1-5 Changing the STOPI value 1-4
Figure 1-6 Modified line fit 1-5
Figure 1-7 “vds-id” graph 1-6
Figure 1-8 Tools menu access to Graph Settings 1-6
Figure 1-9 Graph Settings 1-7
Figure 1-10 Cursors window 1-7
Figure 1-11 Initial Cursor position 1-8
Figure 1-12 Graph with Cursors 1-8
Figure 1-13 Project Navigator — “4terminal-n-fet” tests 1-9
Figure 1-14 “4terminal-n-fet” tests opened in Workspace 1-9
Figure 1-15 Device Plan window for “4terminal-n-fet” 1-10
Figure 1-16 New order for test sequence table 1-11
Figure 1-17 Project Navigator — new execution sequence 1-11
Figure 1-18 “4terminal-n-fet” selected to run 1-11
Figure 1-19 Saving test data 1-12

2 Advanced Applications

Figure 2-1 Devices connected to 707A switching matrix 2-3
Figure 2-2 Add a switch matrix to the system configuration 2-4
Figure 2-3 Add a test fixture to the system configuration 2-4
Figure 2-4 Add a switch card to the system configuration 2-5
Figure 2-5 Define the system connections 2-5
Figure 2-6 Save the system configuration 2-6
Figure 2-7 Project Navigator - “ivswitch” project 2-6
Figure 2-8 Signal paths for “4terminal-n-fet” tests 2-7
Figure 2-9 Signal paths for “3terminal-npn-bjt” tests 2-7
Figure 2-10 Signal paths for “2-wireresistor” tests 2-8
Figure 2-11 Signal paths for “diode” tests 2-8
Figure 2-12 Signal paths for “capacitor” test 2-8
Figure 2-13 “connect” test 2-9
Figure 2-14 “connect” parameters for “4terminal-n-fet” device 2-10
Figure 2-15 Project Navigator - “ivswitch” project 2-10
Figure 2-16 Subsite Plan window 2-11
Figure 2-17 “diode” moved to top of sequence table 2-11
Figure 2-18 “diode” moved to top of Project Navigator 2-12
Figure 2-19 Execution indicator box 2-12

Figure 2-20 KULT main window ... 2-13
Figure 2-21 Open “ki42xxulib” library ... 2-14
Figure 2-22 Open “Rdson42XX” module ... 2-15
Figure 2-23 Copy “Rdson42xx” module as “RdsonAvg” ... 2-16
Figure 2-24 KULT module window ... 2-17
Figure 2-25 Program modifications ... 2-18
Figure 2-26 Module name for Description ... 2-19
Figure 2-27 Save, compile, and build library ... 2-19
Figure 2-28 Project Navigator for “ivswitch” project ... 2-20
Figure 2-29 Add new UTM ... 2-20
Figure 2-30 “rdson10” added to Project Navigator ... 2-21
Figure 2-31 “rdson10” UTM ... 2-21

3 Controlling External Equipment

Figure 3-1 System configuration with external instruments ... 3-3
Figure 3-2 Relationships between KULT and KITE and between user libraries, user modules, and UTMs 3-4
Figure 3-3 Typical CV curve ... 3-5
Figure 3-4 Keithley Model 590 CV Analyzer DUT connections ... 3-5
Figure 3-5 Adding a Keithley 590 CV Analyzer to the system configuration ... 3-6
Figure 3-6 Setting the Model 590 GPIB address ... 3-6
Figure 3-7 Saving the system configuration ... 3-6
Figure 3-8 New Project menu selection ... 3-7
Figure 3-9 Define new project ... 3-7
Figure 3-10 Add a new Subsite Plan to a KITE project ... 3-8
Figure 3-11 Add a new Subsite Plan ... 3-8
Figure 3-12 Add a new Device Plan to a KITE project ... 3-8
Figure 3-13 Add a Device Plan ... 3-9
Figure 3-14 Default test library folders ... 3-10
Figure 3-15 Add the “cvsweep” UTM ... 3-10
Figure 3-16 “cvsweep” UTM ... 3-11
Figure 3-17 Test system for “ivpgswitch” project ... 3-12
Figure 3-18 Adding a pulse generator ... 3-13
Figure 3-19 Pulse generator configuration ... 3-13
Figure 3-20 Adding a switch matrix ... 3-13
Figure 3-21 Configuring the switch matrix ... 3-14
Figure 3-22 Adding a probe station ... 3-14
Figure 3-23 Connecting the switch matrix ... 3-15
Figure 3-24 Saving the system configuration ... 3-15
Figure 3-25 Project Navigator - “ivpgswitch” project ... 3-15
Figure 3-26 First “connect” test - connects the device to the SMUs ... 3-16
Figure 3-27 Signal paths for the pre and post stress tests ... 3-16
Figure 3-28 Second “connect” test - connects the device to the PGU ... 3-17
Figure 3-29 Signal paths to apply the pulse stress ... 3-17
Figure 3-30 PGU initialization ... 3-17
Figure 3-31 “pgu1-setup” - configure the PGU channel ... 3-18
Figure 3-32 PGU stress pulse specifications ... 3-18
Figure 3-33 “pgu-trigger” test - trigger the burst of stress pulses ... 3-18
Figure 3-34 Buttons to close or reduce size of test documents ... 3-19
Figure 3-35 “id-vg” graphs ... 3-20
Figure 3-36 Graph scale settings ... 3-20
Figure 3-37 Sample wafer organization ... 3-21
Figure 3-38 System configuration for the “probesubsites” project ... 3-23
Figure 3-39 Adding a switch matrix ... 3-24
Figure 3-40 Configuring the switch matrix ... 3-24

Figure 3-41 Adding a probe station 3-24
Figure 3-42 Connecting the switch matrix 3-25
Figure 3-43 Saving the system configuration 3-25
Figure 3-44 Project Navigator - probesubsites project 3-26
Figure 3-45 Modified project plan settings 3-27
Figure 3-46 prober-init 3-28
Figure 3-47 Connect SMUs to N-channel MOSFET 3-29
Figure 3-48 Connect SMUs to NPN transistor 3-29
Figure 3-49 prober-separate 3-30
Figure 3-50 prober-prompt test and dialog window 3-30
Figure 3-51 Test sequence 3-31
Figure 3-52 Site Navigator 3-32
Figure 3-53 KITE title bar 3-32

4 Pulse Applications

Figure 4-1 Charge Pumping—hardware setup block diagram 4-3
Figure 4-2 Charge pumping — hardware connection 4-4
Figure 4-3 Two types of sweeps for charge pumping 4-4
Figure 4-4 Example data plots for N_{it} 4-5
Figure 4-5 Pulse IV—hardware setup block diagram 4-17
Figure 4-6 Pulse IV—hardware connections 4-18
Figure 4-7 Side view of scope card connections 4-18
Figure 4-8 Model 8101-PIV test fixture 4-19
Figure 4-9 Model 8101-PIV schematic 4-19
Figure 4-10 PRB-C adapter cable – pulse SMA to SSMC Y 4-20
Figure 4-11 Schematic diagram of the PRB-C adapter cable 4-21
Figure 4-12 Pulse IV connections using PRB-C adapter cables 4-22
Figure 4-13 Pulse IV connections using RF G-S-G probes 4-23
Figure 4-14 Pulse IV connections using the 8101-PIV test fixture 4-23
Figure 4-15 DUT inserted in pulse socket of 8101-PIV test fixture 4-24
Figure 4-16 Project plan for Pulse-IV Complete 4-25
Figure 4-17 PulseIVCal dialog boxes 4-26
Figure 4-18 8101-PIV shorted/through socket 4-26
Figure 4-19 Default definition and typical graph for Vds-id 4-27
Figure 4-20 Default Definition tab and GUI For vds-id-pulse 4-28
Figure 4-21 Default definition and typical graph for vds-id-pulse-vs-dc 4-28
Figure 4-22 Default definition and typical graph for vgs-id 4-29
Figure 4-23 Default definition and typical graph for vgs-id-pulse 4-30
Figure 4-24 Default GUI definition and typical graph for vgs-id-pulse-vs-dc 4-30
Figure 4-25 Typical graphical result for scope-shot 4-31
Figure 4-26 Highlighting all entries in vds-id data sheet 4-34
Figure 4-27 Data from vds-id pasted into vds-id-pulse calc sheet 4-34
Figure 4-28 Graph Definition dialog box and resulting graph that shows the three added curves 4-35
Figure 4-29 Trapping and de-trapping in a single gate voltage pulse 4-54
Figure 4-30 Slow single pulse—hardware setup block diagram 4-55
Figure 4-31 Slow single pulse—hardware connection 4-56
Figure 4-32 Example slow single pulse waveform graph 4-56
Figure 4-33 Single slow pulse example data plot 4-57
Figure 4-34 AC Pulse stress-measure—hardware setup block diagram 4-61
Figure 4-35 AC Pulse stress-measure—hardware matrix card simplified schematic 4-62
Figure 4-36 AC Pulse stress-measure—hardware connections 4-63
Figure 4-37 PIV-Q Block Diagram 4-65
Figure 4-38 PIV-Q connections diagram 4-66
Figure 4-39 Photo showing PIV-Q connections 4-69
Figure 4-40 PIV-Q connections using RF G-S-G probes 4-69

Figure 4-41 PIV-Q connections using the 8101-PIV test fixture 4-70
Figure 4-42 Proper orientation for the plastic package TO-92 n-channel enhancement mode FET .. 4-70
Figure 4-43 QPulse-IV-Complete project 4-72
Figure 4-44 Definition and Graph tabs for Vd-Id-Pulse (Vgs = 5V) 4-73
Figure 4-45 Default definition and typical graph for Vd-Id-Family 4-73
Figure 4-46 Vg-Id-Pulse Graph tab 4-74
Figure 4-47 Default GUI definition for Vg-Id-Pulse-vs-DC 4-75
Figure 4-48 Typical graph for Vg-Id-Pulse-vs-DC 4-75
Figure 4-49 Typical graphical result for ScopeShot -FET 4-76
Figure 4-50 Pulse adapters, cables and hardware 4-118
Figure 4-51 Cross section of a floating gate transistor in both the erased and programmed states 4-119
Figure 4-52 Graph of shifted voltage threshold, VT, due to stored charge on floating gate on a 1 bit (2 level) cell. 4-120
Figure 4-53 Fowler-Nordheim tunneling Program and Erase. 4-121
Figure 4-54 Hot Electron Injection (HEI) Program and Erase. 4-121
Figure 4-55 Block diagram of an example flash test setup using a switch matrix 4-122
Figure 4-56 Block diagram of a flash test setup without using a switch matrix (direct connect) 4-122
Figure 4-57 Program pulse waveforms for a floating gate DUT, with separate pulse waveforms for the DUT gate, drain, source, and bulk 4-123
Figure 4-58 Example erase pulse waveforms for a floating gate DUT, with separate pulse waveforms for the DUT gate, drain, source, and bulk 4-123
Figure 4-59 Program + Erase pulse waveforms for a floating gate DUT, with separate pulse waveforms for the DUT gate, drain, source, and bulk. 4-124
Figure 4-60 Basic schematic of flash testing without a switch matrix 4-124
Figure 4-61 Example results of VT shift in an Endurance test on a NOR flash cell. 4-125
Figure 4-62 Disturb testing – configuration to test a single device 4-126
Figure 4-63 Kpulse showing the Segment Arb settings for the 4205-PG2 card in the lowest numbered-slot (PG2-1) 4-128
Figure 4-64 Kpulse showing the Segment Arb settings for the 4205-PG2 card in the second lowest-numbered slot (PG2-2) 4-128
Figure 4-65 Flash-NAND Project Definition Tab, including arrows for the 6 input arrays 4-131
Figure 4-66 Flash-NAND PulseVoltages Array Entry and PrePulseDelays Entry 4-132
Figure 4-67 Flash connections – program/erase and endurance testing using direct connection to a single, stand-alone 4 terminal device 4-133
Figure 4-68 Flash direct DUT connections – Disturb testing 4-134
Figure 4-69 Flash Switch connections – Characterization, endurance or disturb testing 4-135
Figure 4-70 Supplied items for 4200-Flash package 4-135
Figure 4-71 KCON Row-Column Card Properties for Flash testing with 4 SMUs and 4 VPU pulse channels 4-139
Figure 4-72 Project listing _Memory folder 4-140
Figure 4-73 Flash-NAND project 4-141
Figure 4-74 Parameters for Program or Erase UTMs (using single_pulse_flash module) 4-142
Figure 4-75 Parameters for Fast Program-Erase pulse waveform (using double_pulse_flash module) 4-142
Figure 4-76 Flash-NAND project – Program definition tab 4-142
Figure 4-77 Flash-NAND project – Erase definition tab 4-143
Figure 4-78 Flash-NAND project – Fast-Program-Erase definition tab 4-143
Figure 4-79 Flash-NAND project – SetupDC definition tab 4-144
Figure 4-80 Flash-NAND project – Vt-MaxGm definition tab 4-145
Figure 4-81 Flash-Switch project 4-146
Figure 4-82 ConPin-Pulse test Definition tab 4-146
Figure 4-83 ConPin-Pulse test GUI definition dialog 4-147
Figure 4-84 FlashEndurance-NAND project plan 4-153
Figure 4-85 FlashEndurance-NAND project – Subsite Plan tab 4-154
Figure 4-86 FlashEndurance-NAND project – Device Stress Properties 4-154
Figure 4-87 FlashEndurance-NAND project – Subsite Graph tab 4-155

Figure 4-88 FlashEndurance-NAND project – Program Definition tab 4-156
Figure 4-89 FlashEndurance-NAND project – SetupDC Definition tab 4-157
Figure 4-90 FlashEndurance-NAND project – Vt-MaxGm-Program Definition tab 4-158
Figure 4-91 FlashEndurance-NAND project – Vt-MaxGm-Program Graph tab 4-158
Figure 4-92 FlashEndurance-NAND project – Erase Definition tab 4-159
Figure 4-93 FlashEndurance-Switch project 4-160
Figure 4-94 FlashDisturb-NAND project – Subsite Setup tab 4-164
Figure 4-95 FlashDisturb-NAND project – Device Stress Properties 4-164

List of Tables

1 Graphical Data Analysis and Basic Test Sequencing

Figure 1-1 Graph for the “subvt” test 1-2
Figure 1-2 Open Formulator window 1-3
Figure 1-3 Formulator for “subvt” test 1-3
Figure 1-4 Start and stop points for the line-fit 1-4
Figure 1-5 Changing the STOPI value 1-4
Figure 1-6 Modified line fit 1-5
Figure 1-7 “vds-id” graph 1-6
Figure 1-8 Tools menu access to Graph Settings 1-6
Figure 1-9 Graph Settings 1-7
Figure 1-10 Cursors window 1-7
Figure 1-11 Initial Cursor position 1-8
Figure 1-12 Graph with Cursors 1-8
Figure 1-13 Project Navigator — “4terminal-n-fet” tests 1-9
Figure 1-14 “4terminal-n-fet” tests opened in Workspace 1-9
Figure 1-15 Device Plan window for “4terminal-n-fet” 1-10
Figure 1-16 New order for test sequence table 1-11
Figure 1-17 Project Navigator — new execution sequence 1-11
Figure 1-18 “4terminal-n-fet” selected to run 1-11
Figure 1-19 Saving test data 1-12

2 Advanced Applications

Figure 2-1 Devices connected to 707A switching matrix 2-3
Figure 2-2 Add a switch matrix to the system configuration 2-4
Figure 2-3 Add a test fixture to the system configuration 2-4
Figure 2-4 Add a switch card to the system configuration 2-5
Figure 2-5 Define the system connections 2-5
Figure 2-6 Save the system configuration 2-6
Figure 2-7 Project Navigator - “ivswitch” project 2-6
Figure 2-8 Signal paths for “4terminal-n-fet” tests 2-7
Figure 2-9 Signal paths for “3terminal-npn-bjt” tests 2-7
Figure 2-10 Signal paths for “2-wireresistor” tests 2-8
Figure 2-11 Signal paths for “diode” tests 2-8
Figure 2-12 Signal paths for “capacitor” test 2-8
Figure 2-13 “connect” test 2-9
Figure 2-14 “connect” parameters for “4terminal-n-fet” device 2-10
Figure 2-15 Project Navigator - “ivswitch” project 2-10
Figure 2-16 Subsite Plan window 2-11
Figure 2-17 “diode” moved to top of sequence table 2-11
Figure 2-18 “diode” moved to top of Project Navigator 2-12
Figure 2-19 Execution indicator box 2-12

Figure 2-20 KULT main window ... 2-13
Figure 2-21 Open “ki42xxulib” library ... 2-14
Figure 2-22 Open “Rdson42XX” module ... 2-15
Figure 2-23 Copy “Rdson42xx” module as “RdsonAvg” ... 2-16
Figure 2-24 KULT module window ... 2-17
Figure 2-25 Program modifications ... 2-18
Figure 2-26 Module name for Description ... 2-19
Figure 2-27 Save, compile, and build library ... 2-19
Figure 2-28 Project Navigator for “ivswitch” project ... 2-20
Figure 2-29 Add new UTM ... 2-20
Figure 2-30 “rdson10” added to Project Navigator ... 2-21
Figure 2-31 “rdson10” UTM ... 2-21

3 Controlling External Equipment

Figure 3-1 System configuration with external instruments ... 3-3
Figure 3-2 Relationships between KULT and KITE and between user libraries, user modules, UTMs 3-4
Figure 3-3 Typical CV curve ... 3-5
Figure 3-4 Keithley Model 590 CV Analyzer DUT connections ... 3-5
Figure 3-5 Adding a Keithley 590 CV Analyzer to the system configuration ... 3-6
Figure 3-6 Setting the Model 590 GPIB address ... 3-6
Figure 3-7 Saving the system configuration ... 3-6
Figure 3-8 New Project menu selection ... 3-7
Figure 3-9 Define new project ... 3-7
Figure 3-10 Add a new Subsite Plan to a KITE project ... 3-8
Figure 3-11 Add a new Subsite Plan ... 3-8
Figure 3-12 Add a new Device Plan to a KITE project ... 3-8
Figure 3-13 Add a Device Plan ... 3-9
Figure 3-14 Default test library folders ... 3-10
Figure 3-15 Add the “cvsweep” UTM ... 3-10
Figure 3-16 “cvsweep” UTM ... 3-11
Figure 3-17 Test system for “ivpgswitch” project ... 3-12
Figure 3-18 Adding a pulse generator ... 3-13
Figure 3-19 Pulse generator configuration ... 3-13
Figure 3-20 Adding a switch matrix ... 3-13
Figure 3-21 Configuring the switch matrix ... 3-14
Figure 3-22 Adding a probe station ... 3-14
Figure 3-23 Connecting the switch matrix ... 3-15
Figure 3-24 Saving the system configuration ... 3-15
Figure 3-25 Project Navigator - “ivpgswitch” project ... 3-15
Figure 3-26 First “connect” test - connects the device to the SMUs ... 3-16
Figure 3-27 Signal paths for the pre and post stress tests ... 3-16
Figure 3-28 Second “connect” test - connects the device to the PGU ... 3-17
Figure 3-29 Signal paths to apply the pulse stress ... 3-17
Figure 3-30 PGU initialization ... 3-17
Figure 3-31 “pgu1-setup” - configure the PGU channel ... 3-18
Figure 3-32 PGU stress pulse specifications ... 3-18
Figure 3-33 “pgu-trigger” test - trigger the burst of stress pulses ... 3-18
Figure 3-34 Buttons to close or reduce size of test documents ... 3-19
Figure 3-35 “id-vg” graphs ... 3-20
Figure 3-36 Graph scale settings ... 3-20
Figure 3-37 Sample wafer organization ... 3-21
Figure 3-38 System configuration for the “probesubsites” project ... 3-23
Figure 3-39 Adding a switch matrix ... 3-24
Figure 3-40 Configuring the switch matrix ... 3-24

Figure 3-41 Adding a probe station 3-24
Figure 3-42 Connecting the switch matrix 3-25
Figure 3-43 Saving the system configuration 3-25
Figure 3-44 Project Navigator - probesubsites project 3-26
Figure 3-45 Modified project plan settings 3-27
Figure 3-46 prober-init 3-28
Figure 3-47 Connect SMUs to N-channel MOSFET 3-29
Figure 3-48 Connect SMUs to NPN transistor 3-29
Figure 3-49 prober-separate 3-30
Figure 3-50 prober-prompt test and dialog window 3-30
Figure 3-51 Test sequence 3-31
Figure 3-52 Site Navigator 3-32
Figure 3-53 KITE title bar 3-32

4 Pulse Applications

Figure 4-1 Charge Pumping—hardware setup block diagram 4-3
Figure 4-2 Charge pumping — hardware connection 4-4
Figure 4-3 Two types of sweeps for charge pumping 4-4
Figure 4-4 Example data plots for N_{it} 4-5
Figure 4-5 Pulse IV—hardware setup block diagram 4-17
Figure 4-6 Pulse IV—hardware connections 4-18
Figure 4-7 Side view of scope card connections 4-18
Figure 4-8 Model 8101-PIV test fixture 4-19
Figure 4-9 Model 8101-PIV schematic 4-19
Figure 4-10 PRB-C adapter cable – pulse SMA to SSMC Y 4-20
Figure 4-11 Schematic diagram of the PRB-C adapter cable 4-21
Figure 4-12 Pulse IV connections using PRB-C adapter cables 4-22
Figure 4-13 Pulse IV connections using RF G-S-G probes 4-23
Figure 4-14 Pulse IV connections using the 8101-PIV test fixture 4-23
Figure 4-15 DUT inserted in pulse socket of 8101-PIV test fixture 4-24
Figure 4-16 Project plan for Pulse-IV Complete 4-25
Figure 4-17 PulseIVCal dialog boxes 4-26
Figure 4-18 8101-PIV shorted/through socket 4-26
Figure 4-19 Default definition and typical graph for Vds-id 4-27
Figure 4-20 Default Definition tab and GUI For vds-id-pulse 4-28
Figure 4-21 Default definition and typical graph for vds-id-pulse-vs-dc 4-28
Figure 4-22 Default definition and typical graph for vgs-id 4-29
Figure 4-23 Default definition and typical graph for vgs-id-pulse 4-30
Figure 4-24 Default GUI definition and typical graph for vgs-id-pulse-vs-dc 4-30
Figure 4-25 Typical graphical result for scope-shot 4-31
Figure 4-26 Highlighting all entries in vds-id data sheet 4-34
Figure 4-27 Data from vds-id pasted into vds-id-pulse calc sheet 4-34
Figure 4-28 Graph Definition dialog box and resulting graph that shows the three added curves 4-35
Figure 4-29 Trapping and de-trapping in a single gate voltage pulse 4-54
Figure 4-30 Slow single pulse—hardware setup block diagram 4-55
Figure 4-31 Slow single pulse—hardware connection 4-56
Figure 4-32 Example slow single pulse waveform graph 4-56
Figure 4-33 Single slow pulse example data plot 4-57
Figure 4-34 AC Pulse stress-measure—hardware setup block diagram 4-61
Figure 4-35 AC Pulse stress-measure—hardware matrix card simplified schematic 4-62
Figure 4-36 AC Pulse stress-measure—hardware connections 4-63
Figure 4-37 PIV-Q Block Diagram 4-65
Figure 4-38 PIV-Q connections diagram 4-66
Figure 4-39 Photo showing PIV-Q connections 4-69
Figure 4-40 PIV-Q connections using RF G-S-G probes 4-69

Figure 4-41 PIV-Q connections using the 8101-PIV test fixture 4-70
Figure 4-42 Proper orientation for the plastic package TO-92 n-channel enhancement mode FET .. 4-70
Figure 4-43 QPulse-IV-Complete project 4-72
Figure 4-44 Definition and Graph tabs for Vd-Id-Pulse (Vgs = 5V) 4-73
Figure 4-45 Default definition and typical graph for Vd-Id-Family 4-73
Figure 4-46 Vg-Id-Pulse Graph tab 4-74
Figure 4-47 Default GUI definition for Vg-Id-Pulse-vs-DC 4-75
Figure 4-48 Typical graph for Vg-Id-Pulse-vs-DC 4-75
Figure 4-49 Typical graphical result for ScopeShot -FET 4-76
Figure 4-50 Pulse adapters, cables and hardware 4-118
Figure 4-51 Cross section of a floating gate transistor erased with stored charge, and programmed in the floating gate 4-119
Figure 4-52 Graph of shifted voltage threshold, VT, due to stored charge on floating gate on a 1 bit (2 level) cell. 4-120
Figure 4-53 Fowler-Nordheim tunneling Program and Erase. 4-121
Figure 4-54 Hot Electron Injection (HEI) Program and Erase. 4-121
Figure 4-55 Block diagram of an example flash test setup using a switch matrix 4-122
Figure 4-56 Block diagram of a flash test setup without using a switch matrix (direct connect) 4-122
Figure 4-57 Program pulse waveforms for a floating gate DUT, with separate pulse waveforms for the DUT gate, drain, source, and bulk 4-123
Figure 4-58 Example erase pulse waveforms for a floating gate DUT, with separate pulse waveforms for the DUT gate, drain, source, and bulk 4-123
Figure 4-59 Program + Erase pulse waveforms for a floating gate DUT, with separate pulse waveforms for the DUT gate, drain, source, and bulk. 4-124
Figure 4-60 Basic schematic of flash testing without a switch matrix 4-124
Figure 4-61 Example results of VT shift in an Endurance test on a NOR flash cell. 4-125
Figure 4-62 Disturb testing – configuration to test a single device 4-126
Figure 4-63 Kpulse showing the Segment Arb settings for the 4205-PG2 card in the lowest numbered-slot (PG2-1) 4-128
Figure 4-64 Kpulse showing the Segment Arb settings for the 4205-PG2 card in the second lowest-numbered slot (PG2-2) 4-128
Figure 4-65 Flash-NAND Project Definition Tab, including arrows for the 6 input arrays 4-131
Figure 4-66 Flash-NAND PulseVoltages Array Entry and PrePulseDelays Entry 4-132
Figure 4-67 Flash connections – program/erase and endurance testing using direct connection to a single, stand-alone 4 terminal device. 4-133
Figure 4-68 Flash direct DUT connections – Disturb testing 4-134
Figure 4-69 Flash Switch connections – Characterization, endurance or disturb testing 4-135
Figure 4-70 Supplied items for 4200-Flash package 4-135
Figure 4-72 Project listing _Memory folder 4-140
Figure 4-73 Flash-NAND project 4-141
Figure 4-74 Parameters for Program or Erase UTMs (using single_pulse_flash module) 4-142
Figure 4-75 Parameters for Fast Program-Erase pulse waveform (using double_pulse_flash module) 4-142
Figure 4-76 Flash-NAND project – Program definition tab 4-142
Figure 4-77 Flash-NAND project – Erase definition tab. 4-143
Figure 4-78 Flash-NAND project – Fast-Program-Erase definition tab 4-143
Figure 4-79 Flash-NAND project – SetupDC definition tab 4-144
Figure 4-80 Flash-NAND project – Vt-MaxGm definition tab 4-145
Figure 4-81 Flash-Switch project 4-146
Figure 4-82 ConPin-Pulse test Definition tab. 4-146
Figure 4-83 ConPin-Pulse test GUI definition dialog 4-147
Figure 4-84 FlashEndurance-NAND project plan 4-153
Figure 4-85 FlashEndurance-NAND project – Subsite Plan tab 4-154
Figure 4-86 FlashEndurance-NAND project – Device Stress Properties 4-154
Figure 4-87 FlashEndurance-NAND project – Subsite Graph tab 4-155
Figure 4-88 FlashEndurance-NAND project – Program Definition tab 4-156

Figure 4-89 FlashEndurance-NAND project – SetupDC Definition tab 4-157
Figure 4-90 FlashEndurance-NAND project – Vt-MaxGm-Program Definition tab 4-158
Figure 4-91 FlashEndurance-NAND project – Vt-MaxGm-Program Graph tab 4-158
Figure 4-92 FlashEndurance-NAND project – Erase Definition tab 4-159
Figure 4-93 FlashEndurance-Switch project 4-160
Figure 4-94 FlashDisturb-NAND project – Subsite Setup tab 4-164
Figure 4-95 FlashDisturb-NAND project – Device Stress Properties 4-164

1 Graphical Data Analysis and Basic Test Sequencing

Section Topics List

SubVt slope, page 1-2
- Open “default” project, page 1-2
- Open “subvt” test and display graph, page 1-2
- Line-fit analysis, page 1-3
- Modify the line-fit, page 1-4

Graphical analysis, page 1-5
- Open “default” project and “vds-id” test, page 1-5
- Display and analyze the “vds-id” graph, page 1-6

Sequencing tests on a single device, page 1-9
- Open “default” project, page 1-9
- Open “4terminal-n-fet” tests, page 1-9
- Modify tests, page 1-10
- Change the execution sequence, page 1-10
- Run “4terminal-n-fet” test sequence, page 1-11
- Save and export test data, page 1-12

SubVt slope

This application demonstrates how to use the Formulator to determine the slope of a specified portion of an IV curve. For additional information regarding the Formulator and parameter extraction, refer to the 4200-SCS Reference manual.

Open "default" project

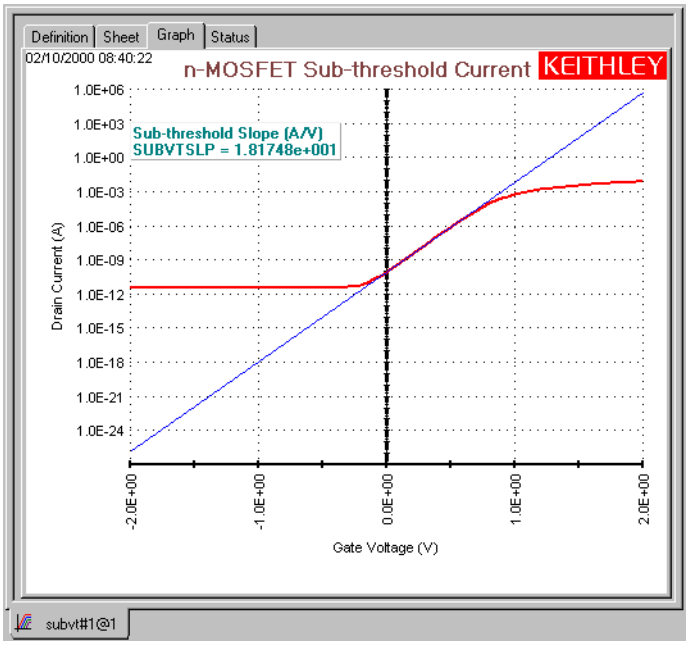
If the "default" project is not currently open, open it using the **Open Project** item of the **File** menu on the toolbar.

Open "subvt" test and display graph

The test is opened by double-clicking "**subvt**" in the Project Navigator. With the test in the Workspace, click the **Graph** tab to display the graph (Figure 1-1).

The Formulator is used to determine the sub-threshold slope for the IV curve. The slope is calculated by performing an exponential line-fit over a specified portion of the IV curve. The straight blue line (IDFIT) is the result of the line-fit. The displayed slope value (SUBVTSLP) is the slope of IDFIT and, in this case, the slope of the fitted portion of the IV curve.

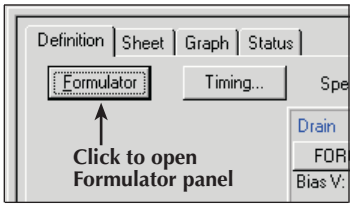
Figure 1-1
Graph for the "subvt" test



Line-fit analysis

The formulas to calculate sub-threshold slope were created using the Formulator. To open the **Formulator** window, click the **Definition** tab in the Workspace, and then click the **Formulator** button as shown in Figure 1-2.

Figure 1-2
Open Formulator window



The formulas for the “subvt” test are shown in Figure 1-3. The formulas created for the test are listed below the formula definition box at the top of the window.

The **STARTI** and **STOPI** formulas specify the portion of the IV curve for the line-fit. These two current data points are shown in Figure 1-4. Notice that these start/stop points section off a linear portion of the IV curve.

The **IDFIT** formula uses the STARTI and STOPI values to calculate the data points for the IDFIT line, which is the straight blue line in the graph. Finally, the **SUBVTSLP** formula calculates the slope of the IDFIT line.

NOTE *Some engineers prefer to view the inverse of the subthreshold slope. This is easily accomplished by adding the formula:*
INVSUBVTSLP = 1/SUBVTSLP.

Figure 1-3
Formulator for “subvt” test

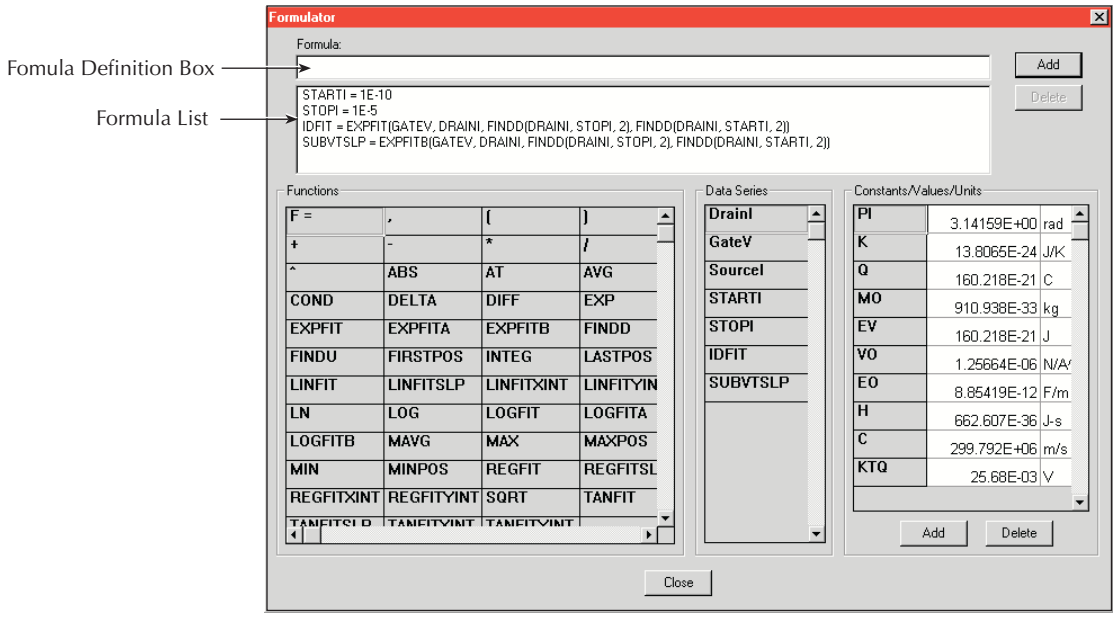
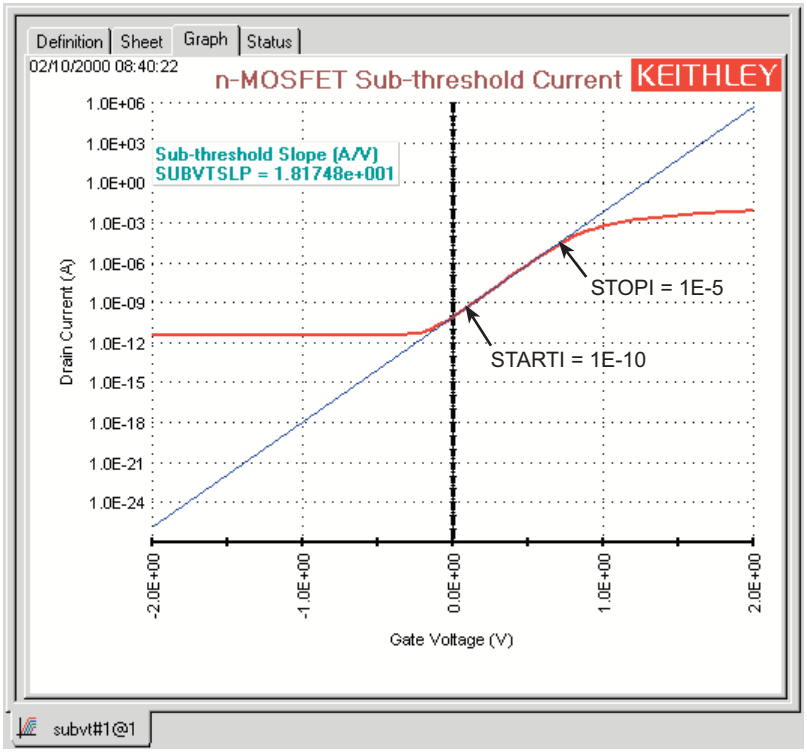


Figure 1-4
Start and stop points for the line-fit



Modify the line-fit

The following exercise shows how to modify the line-fit.

1. Open the **Formulator** window for the “subvt” test.
2. In the formula list, double-click **STOPI = 1E-5** to place it in the formula definition box.
3. Using the keyboard, change the stop value to **1.0E-2** as shown in Figure 1-5.
4. Click the **Add** button (Figure 1-5) to place the modified formula in the list.

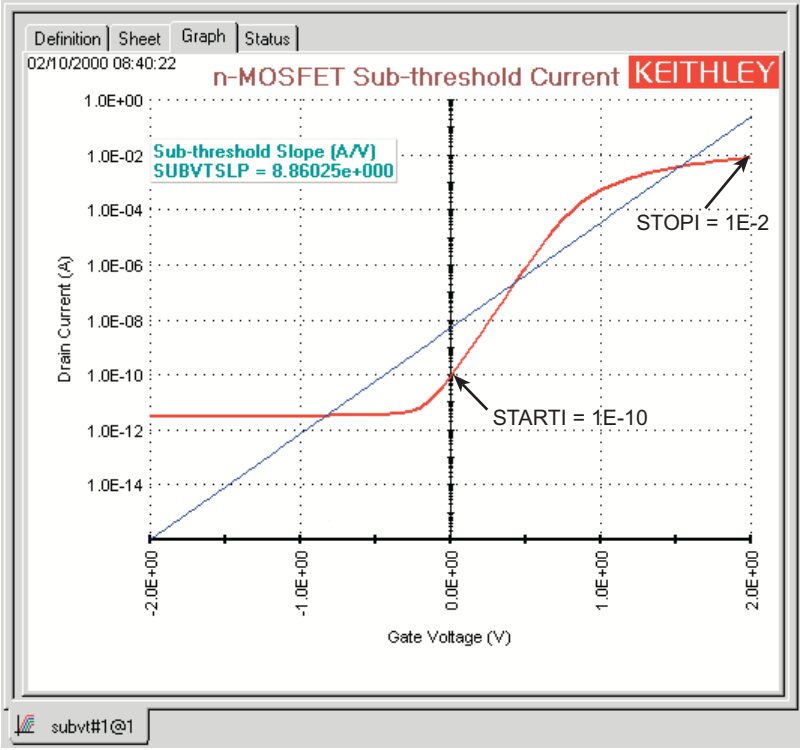
NOTE *A pop-up menu will indicate that the formula already exists. Click* **Yes** *to update the formula.*

5. Close the Formulator by clicking the **Close** button at the bottom of the window.
6. In the Workspace, click the **Graph** tab to display the graph.

Figure 1-5
Changing the STOPI value

As shown in Figure 1-6, the IV curve does not fit the slope of the IDFIT line. This is because the exponential line-fit was performed on a non-linear portion of the IV curve. The start/stop points in Figure 1-6 show the non-linear portion of the IV curve used for the line-fit. This of course, invalidates the SUBVTSLP results.

Figure 1-6
Modified line fit



The above exercise demonstrates the value of displaying the IDFIT line in the graph. If there is a good line-fit as shown in Figure 1-1, then the “SUBVTSLP” value is the slope for the fitted portion of the IV curve.

Graphical analysis

This application demonstrates how to analyze graphical data using Cursors. With a Cursor positioned on a curve, the X and Y coordinate readings for the graph point are displayed in the graph. For details, refer to the 4200-SCS Reference manual.

Open “default” project and “vds-id” test

If the “default” project is not currently open, open it using the **Open Project** item of the **File** menu on the toolbar.

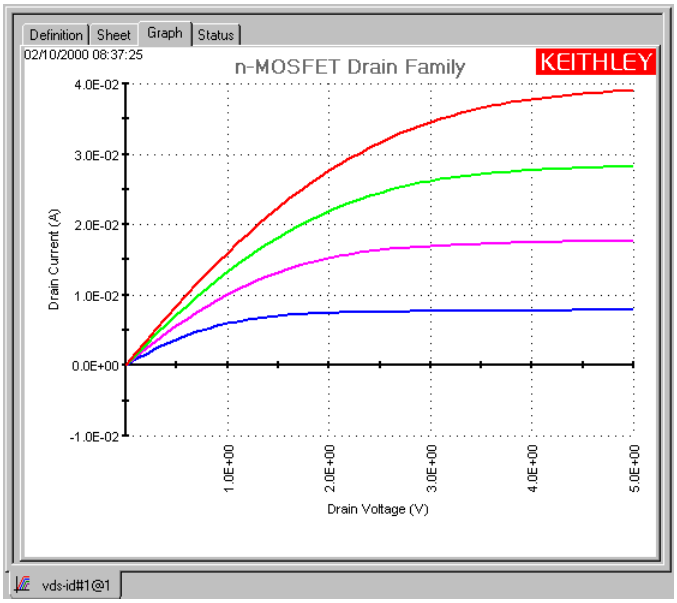
The test is opened by double-clicking “**vds-id**” in the Project Navigator.

Display and analyze the “vds-id” graph

Step 1. Display the graph

In the Workspace, click the **Graph** tab for the “vds-id” test to display the graph. A typical graph for this test is shown in Figure 1-7.

Figure 1-7
“vds-id” graph



Step 2. Open Graph menu

While a graph is displayed, the **Graph** menu can be opened from the **Tools** menu as shown in Figure 1-8. It can also be opened by placing the mouse pointer in an open area of the graph, and clicking the right mouse button. The **Graph** menu is shown in Figure 1-9.

Figure 1-8
Tools menu access to Graph Settings

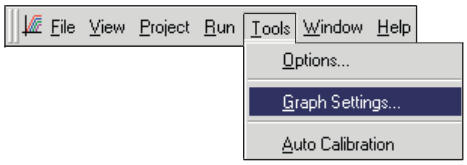
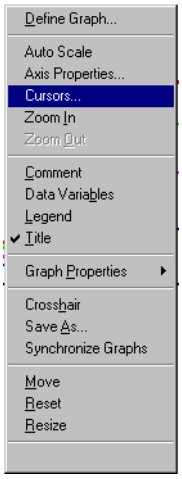


Figure 1-9
Graph Settings



Step 3. Enable Cursors

In the **Graph** menu, select the **Cursors** item to display the Cursors window. The Cursors window in Figure 1-10 shows that **Cursors 1**, **2**, **3** and **4** are enabled. A Cursor is enabled (√) or disabled by clicking the **Visible** box in the **Cursor** area of the window. With the desired Cursor(s) enabled, click **OK** to close the window. For details, refer to the 4200-SCS Reference manual..

The Cursor(s) will appear at the first data point of the first data series as shown in Figure 1-11.

Figure 1-10
Cursors window

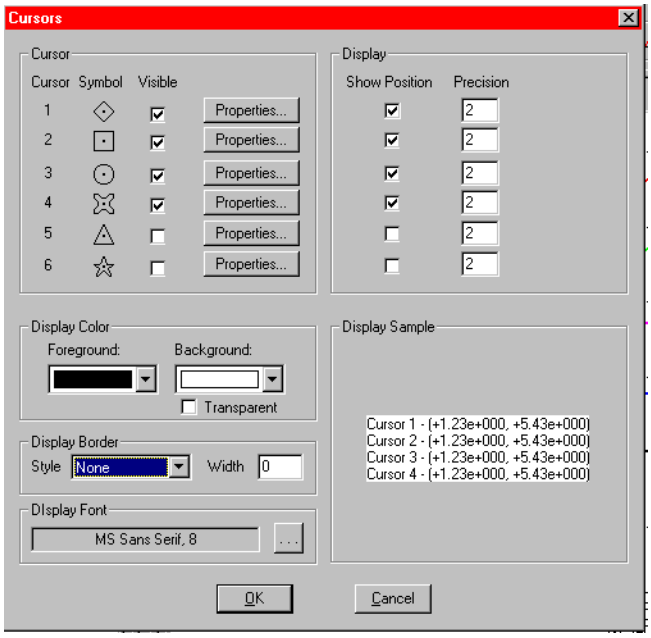
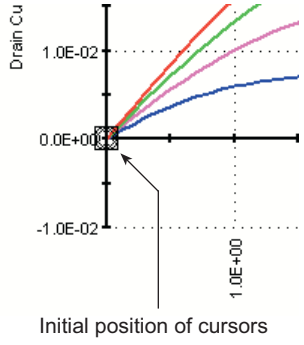


Figure 1-11
Initial Cursor position



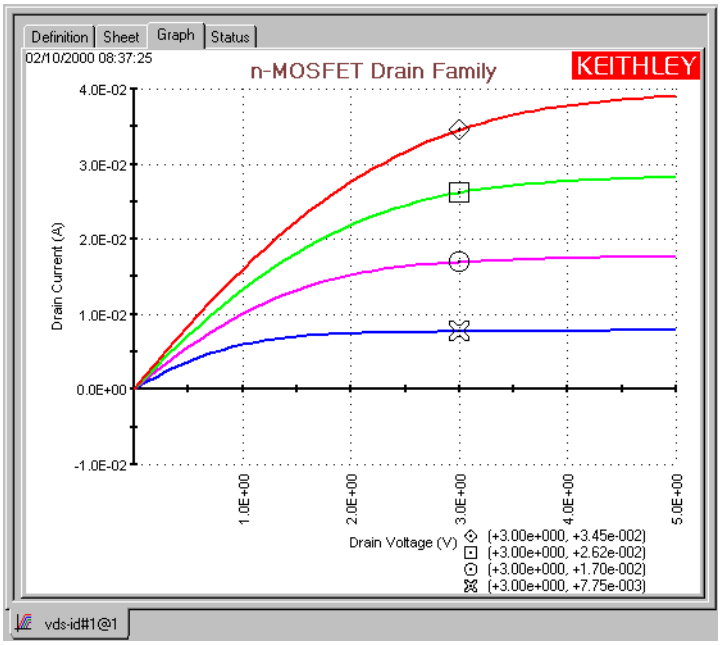
Step 4. Position Cursor on I-V curve

To position a Cursor, place the mouse pointer on the Cursor, hold down the left mouse button, and drag it to the desired point on an IV curve. The drain voltage (x-axis) and drain current (y-axis) readings for the graph point are displayed in the Cursor Display at the bottom of the graph.

The properties of each Cursor can be set by right-clicking the Cursor. The **Cursors window** can be opened by right-clicking the Cursor Display.

In Figure 1-12, Cursor data provides the drain current readings for each IV curve at a drain voltage of 3V.

Figure 1-12
Graph with Cursors



Sequencing tests on a single device

This application demonstrates how to run a test sequence on a single device. When the test sequence is started, the tests for the device will execute in the order that they are presented in the Project Navigator. That is, they will be executed in top-down order. For details refer to the 4200-SCS Reference manual.

This application will also show you how to change the order of execution for the test sequence. For details, refer to the 4200-SCS Reference manual..

Open "default" project

If the "default" project is not currently open, open it using **File -> Open Project**.

Open "4terminal-n-fet" tests

The partial Project Navigator in Figure 1-13 shows the five tests for the "**4terminal-n-fet**" device. Double-click each test to open it and place it in the Workspace.

Figure 1-13
Project Navigator — "4terminal-n-fet" tests

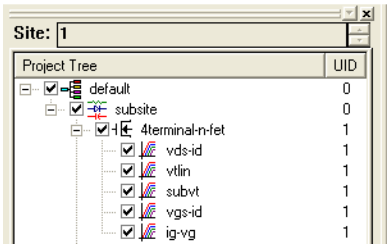
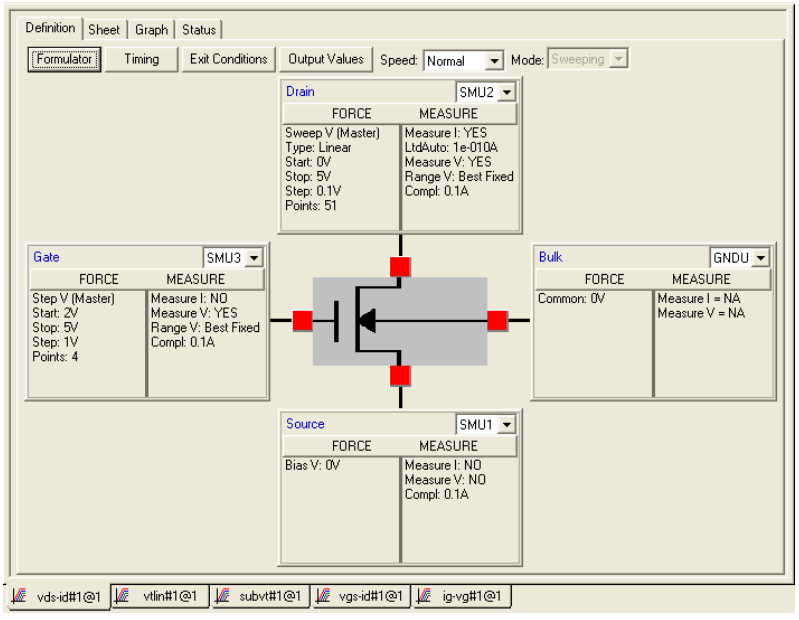


Figure 1-14 shows all five tests opened in the Workspace. A test is displayed by clicking the test name tab at the bottom of the Workspace, or double-clicking on the test in the Project Navigator.

Figure 1-14
"4terminal-n-fet" tests opened in Workspace



Modify tests

All instrument selections shown on the **Definition** tab must match the actual physical connections to the device. If you change the instrument selections for one test, you must also make the same change to the other four tests on that device so that the tests can be executed as a sequence.

Change the execution sequence

The order of presentation in the Project Navigator determines the execution sequence. For the Project Navigator in Figure 1-13, the “4terminal-n-fet” tests will execute in the following, top-down, order:

“vds-id” ⇒ “vtlin” ⇒ “subvt” ⇒ “vgs-id” ⇒ “ig-vg”

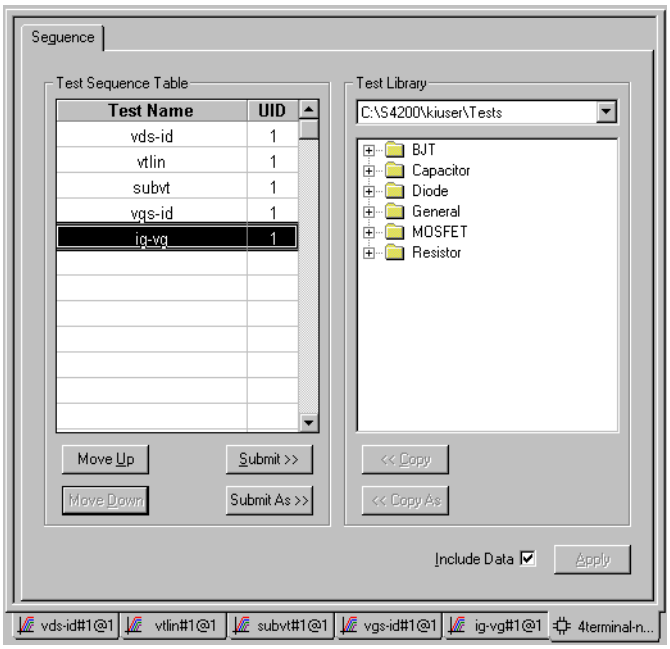
Perform the following steps to change the execution sequence so that “ig-vg” is the first test to be executed:

Step 1. Open Device Plan window

The Device Plan for the “4terminal-n-fet” device is opened by double-clicking “**4terminal-n-fet**” in the Project Navigator. The Device Plan **Sequence** tab is shown in Figure 1-15.

Notice that the execution sequence for the tests in the Project Navigator appears in the **Test Sequence Table** of the Device Plan.

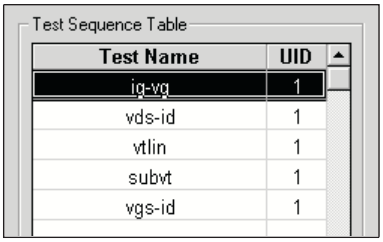
Figure 1-15
Device Plan window for “4terminal-n-fet”



Step 2. Select and move “ig-vg”

The “ig-vg” test is selected by clicking on it in the **Test Sequence Table**. Figure 1-15 shows “**ig-vg**” selected. Use the **Move Up** button to move the test to the top of the sequence table. Figure 1-16 shows the new order for the **Test Sequence Table**.

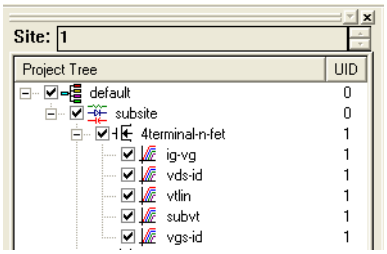
Figure 1-16
New order for test sequence table



Step 3. Apply new execution sequence to the Device Plan

In the Device Plan window (Figure 1-15), click the **Apply** button to apply the new execution sequence to the project. The partial Project Navigator in Figure 1-17 shows the new execution sequence.

Figure 1-17
Project Navigator — new execution sequence

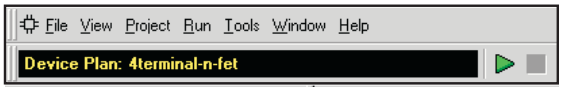


Run “4terminal-n-fet” test sequence

Step 1. Select “4terminal-n-fet” Device Plan

In the Project Navigator, click “4terminal-n-fet” to select the test sequence. That Device Plan name appears in the Test/Plan Indicator box as shown in Figure 1-18.

Figure 1-18
“4terminal-n-fet” selected to run



By selecting “**4terminal-n-fet**” and pressing the **Run** button, all of the tests associated with this device will be executed.

Step 2. Click Run button to execute the test sequence

The green **Run** button starts the test sequence. The name of each individual test will be displayed in the Test/Plan Indicator box while it is being executed.

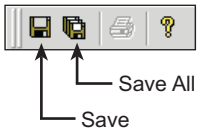
You can observe data being graphed while each test is in progress. A test is displayed by clicking its tab at the bottom of the Workspace. For example, click the **ig-vg#1@1** tab to display the test. The graph for the test is then displayed by clicking the **Graph** tab at the top of the Workspace.

Save and export test data

The toolbar buttons to save test data are shown in Figure 1-19. Click **Save** to save test data and the setup for the selected (displayed) test. Click **Save All** to save test data and setups for all tests in project.

As previously explained, data for each test can be exported as an Excel workbook.

Figure 1-19
Saving test data



2 Advanced Applications

Section Topics List

Controlling a switch matrix, page 2-2
- KCON setup, page 2-4
- Open KITE and the “ivswitch” project, page 2-6
- Running test sequences, page 2-7
- “connect” test description, page 2-9

Sequencing tests on multiple devices, page 2-10
- Open “ivswitch” project, page 2-10
- Execute the test sequence (Subsite Plan), page 2-12

Customizing a user test module (UTM), page 2-12
- Open KULT, page 2-13
- Open the “ki42xxulib” user library, page 2-14
- Open the “Rdson42XX” user module, page 2-15
- Copy “Rdson42XX” to “RdsonAvg”, page 2-16
- Open and modify the “RdsonAvg” user module, page 2-17
- Save, compile, and build the modified library, page 2-19
- Add a new UTM to the “ivswitch” project, page 2-20
- Test description, page 2-21

In this section, you will learn the following:

- **Controlling a switch matrix** — Demonstrates how to use a switch matrix to automatically connect any instrument terminal to any test system pin.
- **Sequencing tests on multiple devices** — Demonstrates how to run a test sequence that will utilize a switch matrix to automatically test all of the devices in the “ivswitch” project.
- **Customizing a user test module (UTM)** — Demonstrates how to modify a user module using the Keithley User Library Tool (*KULT*).

The following equipment is required to complete this tutorial and obtain data that functionally correlates with data included with the “ivswitch” sample project.

- 1 - Keithley Model 4200-SCS with a total of three SMUs (PreAmps not required)
- 1 - Keithley Model 8006 Component Test Fixture
- 1 - Keithley Model 707A or 708 Switch Matrix
- 1 - Keithley Model 7072 or 7174A 8×12 matrix card
- 4 - Keithley Model 4200-MTRX-X cables (0 if using PreAmps)
- 12 - Keithley Model 4200-TRX-X cables (16 if using PreAmps)
- 1 - Keithley Model 7007 GPIB cable
- 1 - Keithley Model 236-ILC-3 safety interlock cable
- 1 - NPN transistor (2N3904 or similar)
- 1 - N-channel MOSFET (Temic SD210DE or similar)
- 1 - Capacitor (10pF)
- 1 - Resistor (1GΩ)
- 1 - Diode (1N970B or similar)

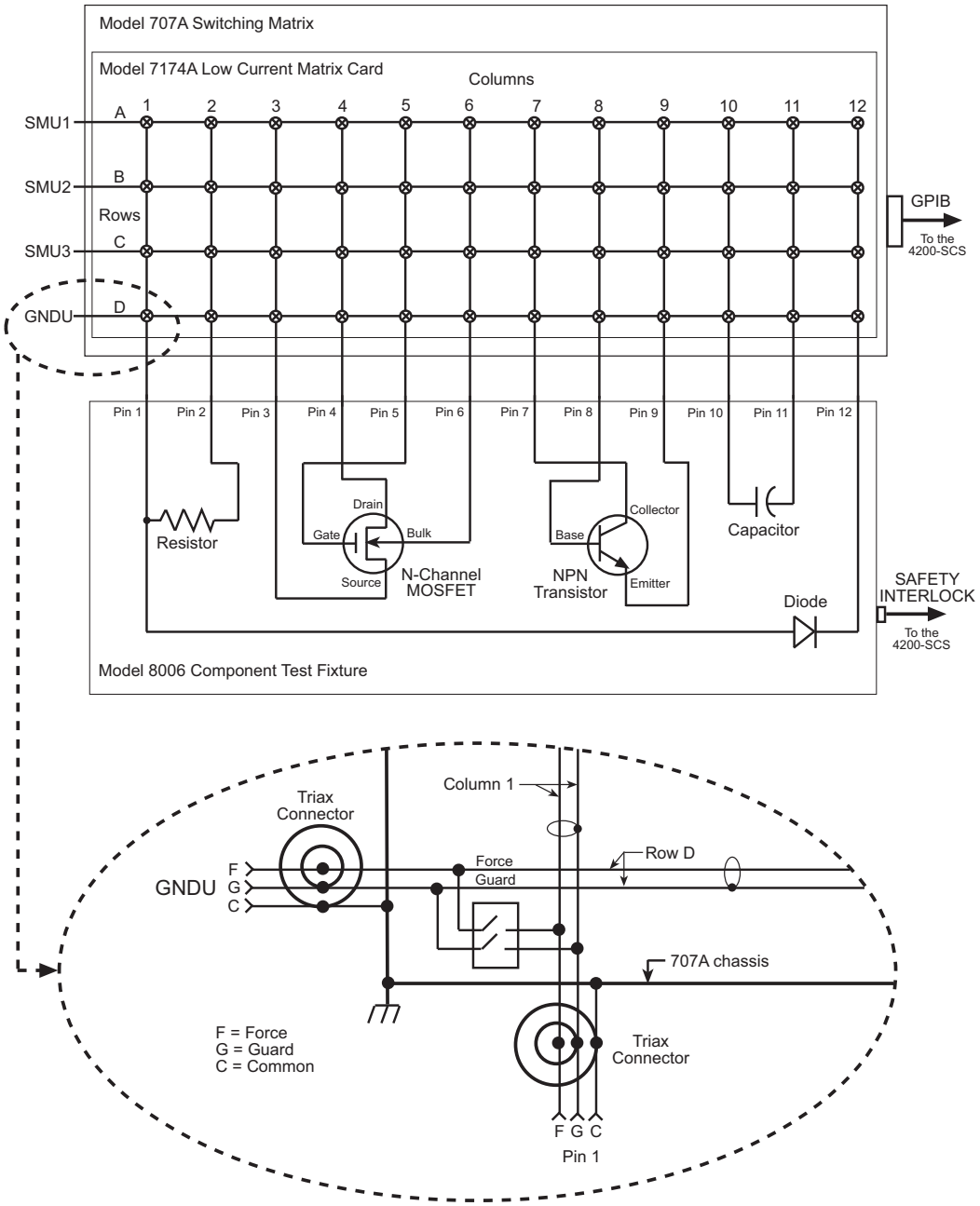
Controlling a switch matrix

This tutorial demonstrates how to use a switch matrix to automatically connect any instrument terminal to any test system pin. The “ivswitch” sample project will be used to illustrate this functionality. Before loading and running the “ivswitch” project, the 4200-SCS, switch matrix, and component test fixture must be connected as illustrated in Figure 2-1.

The switch matrix is controlled by the 4200-SCS via the GPIB bus. Use a Model 7007 GPIB cable to connect the Model 707 Switching Matrix to the 4200-SCS. For connection details, refer to the 4200-SCS Reference manual. This example shows a Model 7174A matrix card installed in slot 1 of a Model 707A Switching Matrix. The row-column connection scheme is used for this tutorial.

A User Test Module (UTM) is used to control the switch matrix. When a test sequence for a device is started, the UTM will close the appropriate matrix crosspoints to connect the specified instrument terminals to the appropriate test system pins. For details on UTMs, refer to the 4200-SCS Reference manual.

Figure 2-1
Devices connected to 707A switching matrix



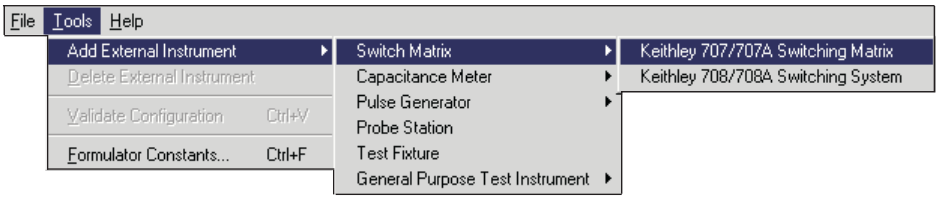
KCON setup

After connecting the system as indicated in Figure 2-1, run the Keithley CONfiguration utility (*KCON*) to add the switch matrix and test fixture to the system configuration. In general, *KCON* is used to manage the configuration of all instrumentation controlled by the 4200-SCS software. Once the matrix and test fixture have been added, and the instrument-to-matrix-to-pins connections have been defined, simply specify an instrument terminal and test system pin and *KITE* will automatically connect the two using the matrix. In general, changes to the system configuration will only be necessary when changes to instrument-to-matrix-to-pins wiring are made.

Follow the steps below to start KCON and modify the system configuration as described above. For additional information regarding KCON, refer to the 4200-SCS Reference manual. Similarly, for additional information regarding switch matrix configuration and usage, refer to the 4200-SCS Reference manual.

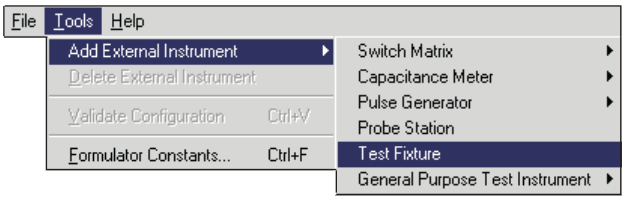
1. On the desktop, double-click the **KCON** icon to open *KCON*.
2. Using the **Tools** menu, add a switch matrix to the system configuration as indicated in Figure 2-2.

Figure 2-2
Add a switch matrix to the system configuration



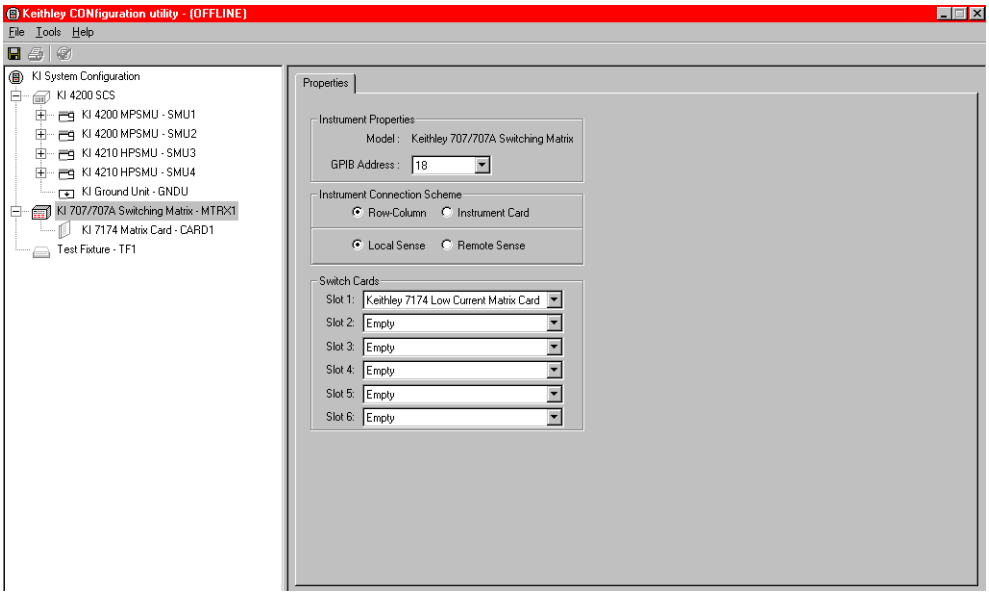
3. Using the **Tools** menu, add a test fixture to the system configuration as indicated in Figure 2-3.

Figure 2-3
Add a test fixture to the system configuration



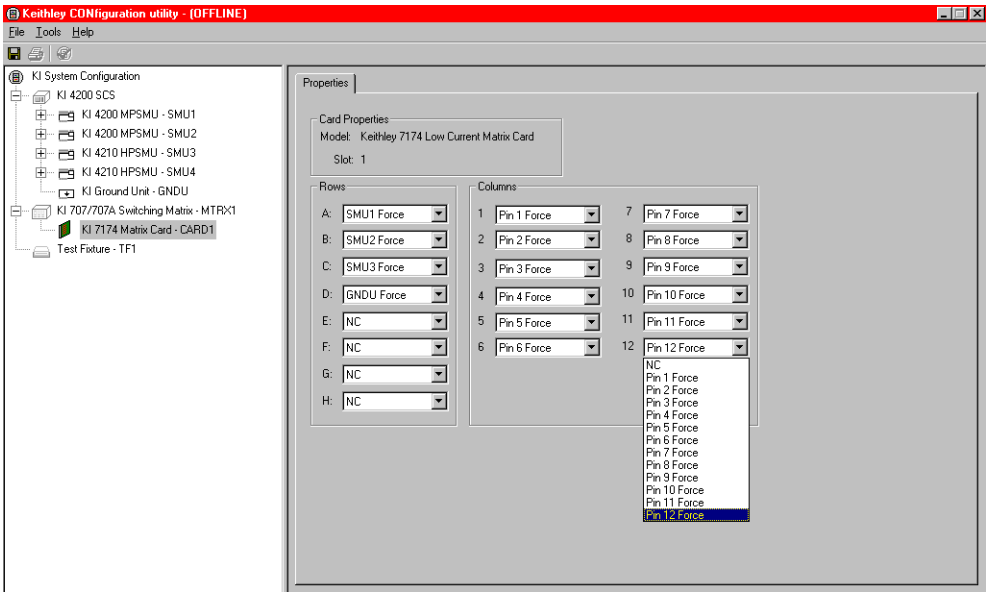
4. Select the **KI 707/707A Switching Matrix - MTRX1** item in the Configuration Navigator (tree control on left side of screen) and add a **Keithley 7174 Low Current Matrix Card** to **Slot 1** of the switch matrix. Add the switch card using the pull down menu on the **Properties** tab. See Figure 2-4.

Figure 2-4
Add a switch card to the system configuration



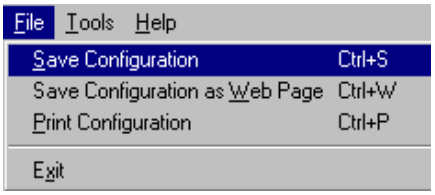
5. Select the **KI 7174 Matrix Card - CARD1** item in the Configuration Navigator. Connect the SMUs, GNDU, and test fixture pins as indicated in Figure 2-1 using the pull down menus on the **Properties** tab. See Figure 2-5.

Figure 2-5
Define the system connections



6. **Save** the system configuration and **Exit** *KCON*. See Figure 2-6.

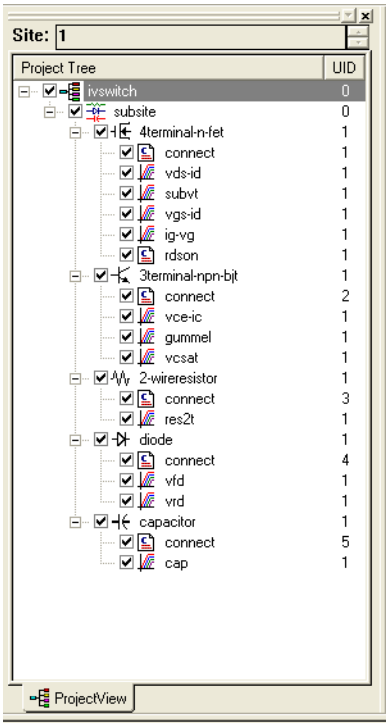
Figure 2-6
Save the system configuration



Open *KITE* and the "ivswitch" project

1. On the desktop, double-click the **KITE** icon to open *KITE*.
2. Open the "**ivswitch**" project from the **File** menu on the *KITE* toolbar (click **Open Project**). The Project Navigator for the "ivswitch" project is shown in Figure 2-7.

Figure 2-7
Project Navigator - "ivswitch" project



Running test sequences

NOTE *For detailed information regarding test and sequence execution, refer to the 4200-SCS Reference manual.*

The “ivswitch” project uses the same ITMs that are used in the “default” project. The primary difference between the two projects is that the “ivswitch” project uses “connect” UTMs to control the switch matrix. As shown in Figure 2-7, there is a “connect” UTM at the beginning of each device test sequence.

A test sequence for a device is executed by selecting the **Device Plan**, and then clicking the green **Run** button. When a Device Plan is started, the connect test closes the appropriate matrix crosspoints to connect the instruments to the appropriate device.

All devices may be tested by selecting the **Subsite Plan** and clicking the green **Run** button.

Figure 2-8 through Figure 2-12 show the signal paths that are automatically selected for the five devices.

Figure 2-8
Signal paths for “4terminal-n-fet” tests

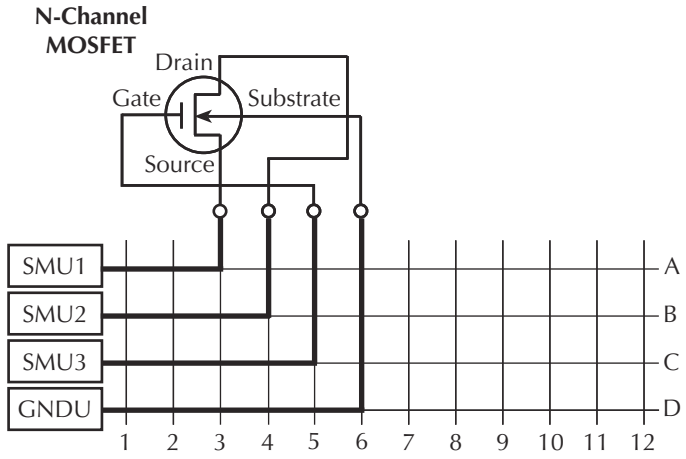


Figure 2-9
Signal paths for “3terminal-npn-bjt” tests

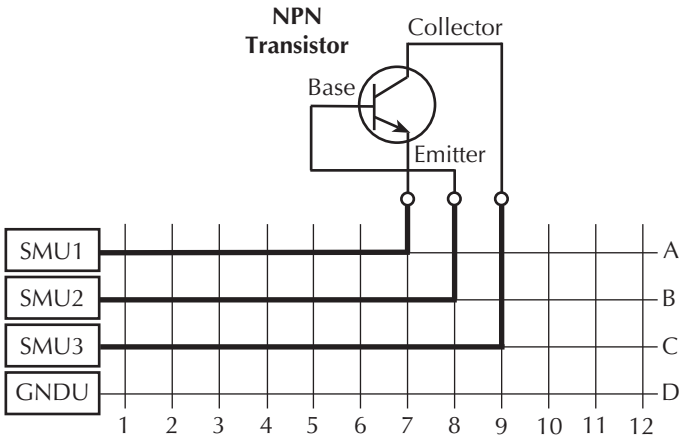


Figure 2-10
Signal paths for "2-wireresistor" tests

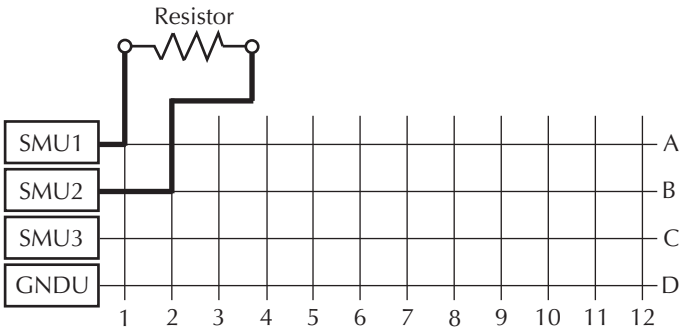


Figure 2-11
Signal paths for "diode" tests

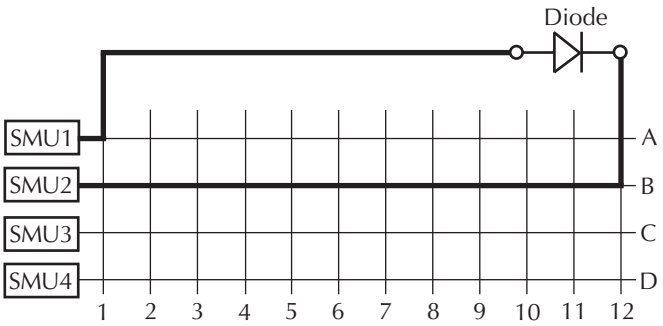
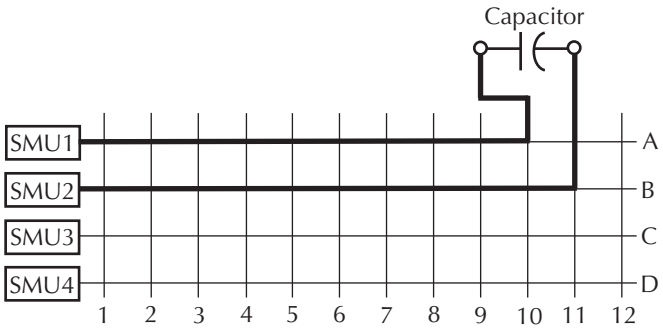


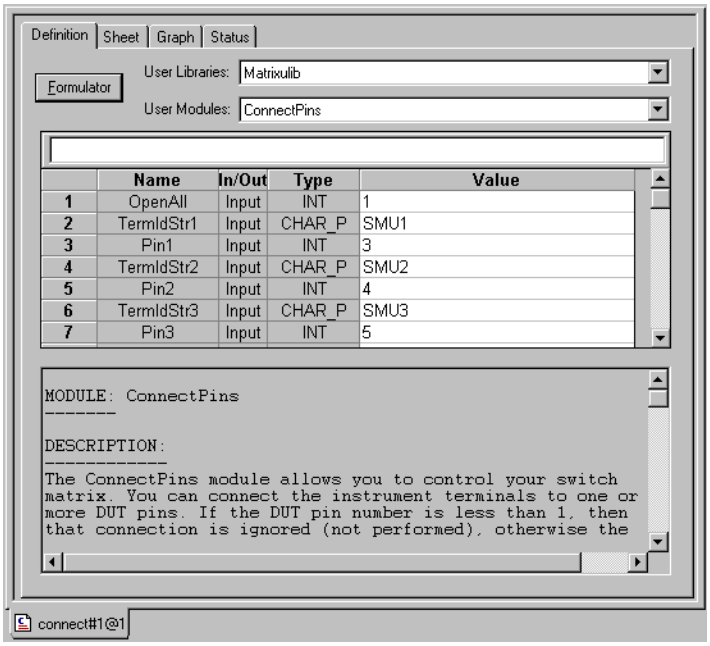
Figure 2-12
Signal paths for "capacitor" test



"connect" test description

In the Project Navigator, double-click "**connect**" under the "4terminal-n-fet" device to open the test. The test is shown in Figure 2-13.

Figure 2-13
"connect" test



The "connect" test is a User Test Module (UTM). *KITE* supports two types of test modules; Interactive Test Modules (ITMs) and UTMs. A UTM, like an ITM, has **Definition**, **Sheet**, **Graph**, and **Status** tabs. The operation of each tab, regardless of test module type, is identical except for the **Definition** tab.

On the UTM **Definition** tab, the user connects the UTM to a User Module located within a User Library, and sets the module parameter values. This information is stored with the UTM when it is saved. When a UTM is executed, the parameters will be passed from the UTM to the user module and the user module will be executed. User libraries and user modules are created and managed using the Keithley User Library Tool (*KULT*). Refer to the 4200-SCS Reference manual for more information regarding user libraries.

In this example, the "connect" UTM is connected to the **ConnectPins** user module in the **Matrixulib** user library. **ConnectPins** has a total of 17 parameters. The first parameter, **OpenAll**, will cause **ConnectPins** to open all matrix crosspoints before closing any additional crosspoints. It is a good practice to open all the switch connections before making any new closures. Inadvertent switch closures may damage DUT.

The 16 additional parameters are comprised of eight terminal-pin-pairs. As shown in Figure 2-14, each specified terminal-pin-pair causes **ConnectPins** to make the desired matrix connection. Because the instrument-to-matrix-to-pin connectivity was defined using *KCON*, *KITE* is able to automatically connect the specified instrument terminals to the appropriate tester pins.

NOTE *If a Pin parameter is < 1, the terminal-pin-pair is ignored and no matrix connections are made.*

Figure 2-14
"connect" parameters for "4terminal-n-fet" device

	Name	In/Out	Type	Value	
1	OpenAll	Input	INT	1	← *Opens all relays*
2	TermIdStr1	Input	CHAR_P	SMU1	← *Connects SMU1 to pin 3 of test fixture*
3	Pin1	Input	INT	3	
4	TermIdStr2	Input	CHAR_P	SMU2	← *Connects SMU2 to pin 4 of test fixture*
5	Pin2	Input	INT	4	
6	TermIdStr3	Input	CHAR_P	SMU3	← *Connects SMU3 to pin 5 of test fixture*
7	Pin3	Input	INT	5	
•					
•					
•					
16	TermIdStr8	Input	CHAR_P	GNDU	← *Connects GNDU to pin 6 of test fixture*
17	Pin8	Input	INT	6	

Sequencing tests on multiple devices

For the previous tutorial, a switch matrix was added to the test system to automate connection changes for different devices. When a test sequence for a device (Device Plan) is executed, the "connect" test closes the appropriate matrix crosspoints to connect that device to the appropriate instrumentation. The test sequence stops after the Device Plan has been executed.

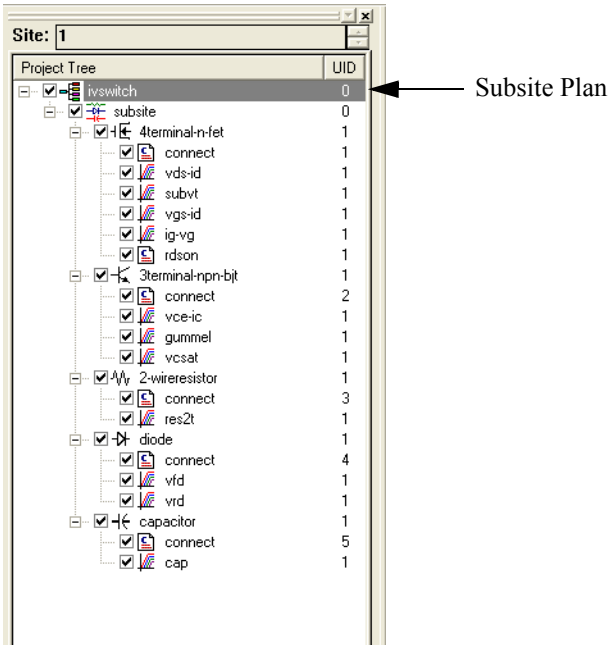
This tutorial demonstrates how to run a test sequence that will automatically test all the devices in the "ivswitch" project. After all the devices have been tested, the test sequence will stop.

Open "ivswitch" project

If the "ivswitch" project is not currently open, open it using the **Open Project** item of the **File** menu on the toolbar. The Project Navigator for the "ivswitch" project is shown in Figure 2-15.

With a switch matrix added to the system, all the devices can be tested by starting the test sequence from the subsite level of the Project Navigator. Figure 2-15 shows the **Subsite Plan** selected (highlighted) to execute.

Figure 2-15
Project Navigator - "ivswitch" project



Modify test sequence

The Project Navigator shows the execution sequence for the **Subsite Plan**. As shown in Figure 2-15, the “4terminal-n-fet” will be tested first, followed by tests for the other four devices.

The device test sequence may be changed using the **Subsite Plan Sequence** tab. The following exercise shows how to change the test sequence by making “diode” the first device in the sequence:

1. In the Project Navigator, double-click **“subsite**” to open the **Subsite Plan** window (see Figure 2-16).
2. In the **Device Sequence Table**, click “**diode”** to select it. Figure 2-16 shows “diode” selected.
3. Use the **Move Up** button to move “diode” to the top of the sequence table (Figure 2-17).
4. At the bottom right-hand corner of the **Subsite Plan** window, click the **Apply** button to change the sequence (see Figure 2-18).

Figure 2-16
Subsite Plan window

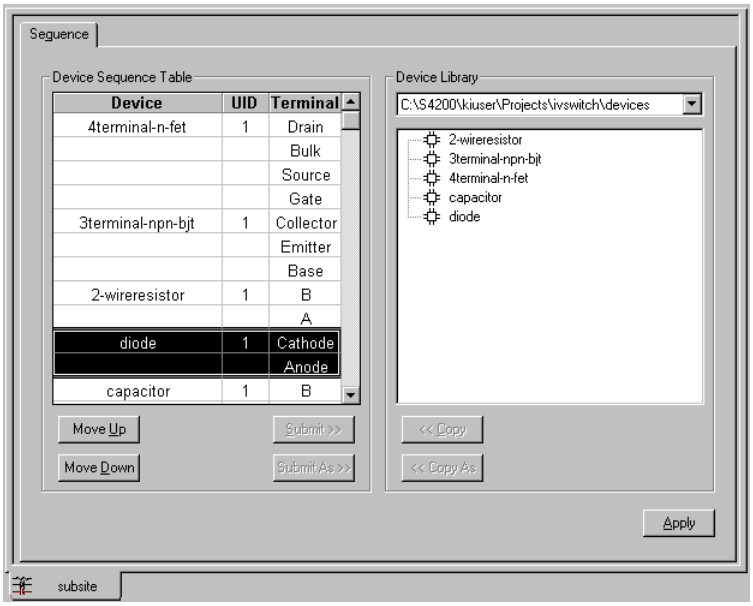


Figure 2-17
“diode” moved to top of sequence table

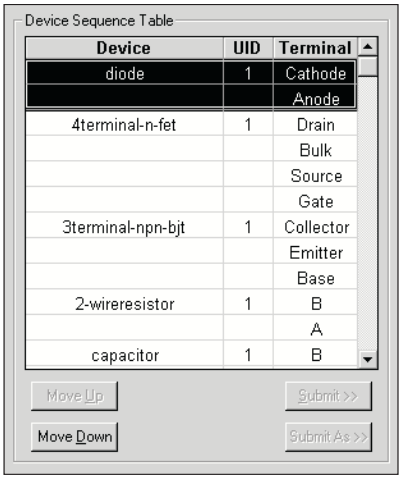
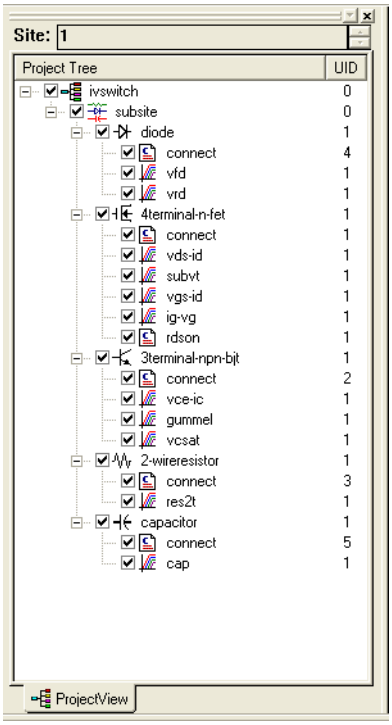


Figure 2-18
"diode" moved to top of Project Navigator



Execute the test sequence (Subsite Plan)

To select the **Subsite Plan**, click **"subsite"** in the Project Navigator. The **Subsite Plan** name will appear in the execution indicator box as shown in Figure 2-19.

To execute the **Subsite Plan**, click the green **Run** button. The first test for each device will control the switch matrix, which connects the device to the instrumentation. The switch matrix was added in the previous application "Controlling a switch matrix" on page 2-2.

While each test is running, the test name will appear in the execution indicator box. After the last test "vt" is executed, the testing process will stop.

Figure 2-19
Execution indicator box

Customizing a user test module (UTM)

This tutorial demonstrates how to modify a user module using the Keithley User Library Tool (*KULT*). In the "ivswitch" project, there is a test named "rdson." The "rdson" test measures the drain-to-source resistance of a saturated N-channel MOSFET as follows:

1. Applies 2V to the gate (Vg) to saturate the MOSFET.
2. Applies 3V to the drain (Vd1) and performs a current measurement (Id1).
3. Applies 5V to the drain (Vd2) and performs another current measurement (Id2).
4. Calculates the drain-to-source resistance "rdson" as follows:
   "rdson" = (Vd2-Vd1) / (Id2-Id1)

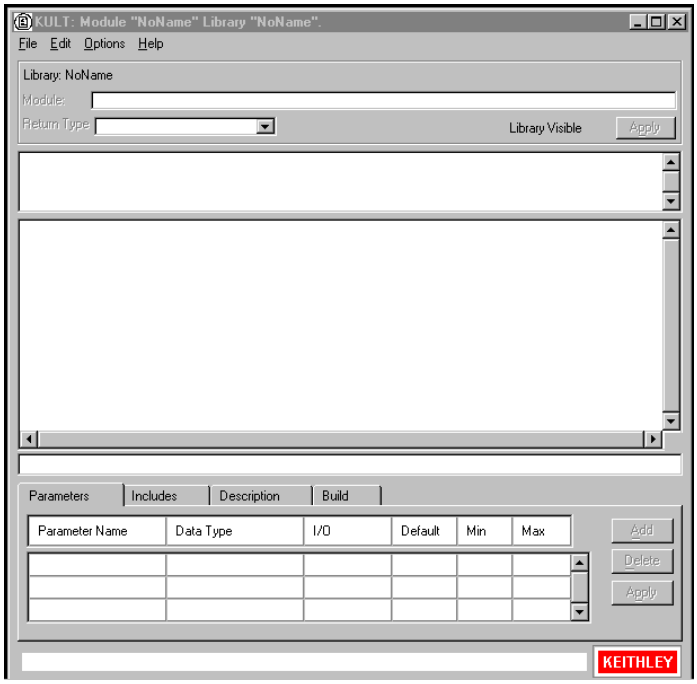
The “rdson” test has a potential shortcoming. If the drain current is noisy, the two current measurements may not be representative of the actual drain current. Therefore, the calculated resistance may be incorrect.

In this example, the user module will be modified in *KULT* such that 10 current measurements will be performed at Vd1 and 10 more at Vd2. The current readings at Vd1 will be averaged to yield Id1, and the current readings at Vd2 will be averaged to yield Id2. Using averaged current readings *smooths* out the noise. For details on using *KULT*, refer to the 4200-SCS Reference manual.

Open *KULT*

From the desktop, open the *KULT* tool by double-clicking the **KULT** icon. The **KULT** main window is shown in Figure 2-20.

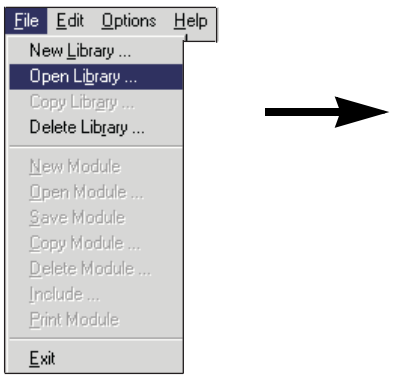
Figure 2-20
KULT main window



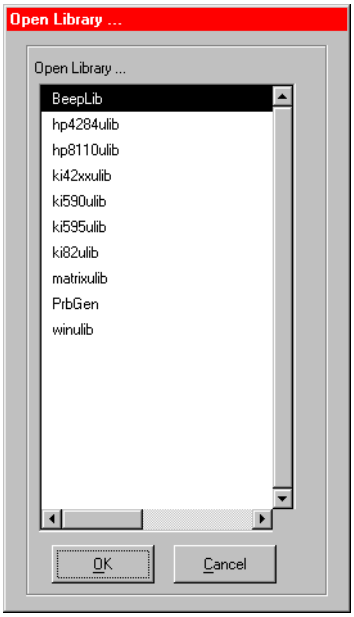
Open the “ki42xxulib” user library

1. From the **File** menu, select the **Open Library** item (see Figure 2-21A).
2. From the **Open Library** window, select “**ki42xxulib**” as shown in Figure 2-21B and click **OK**.

Figure 2-21
Open “ki42xxulib” library



*A. Select **Open Library***

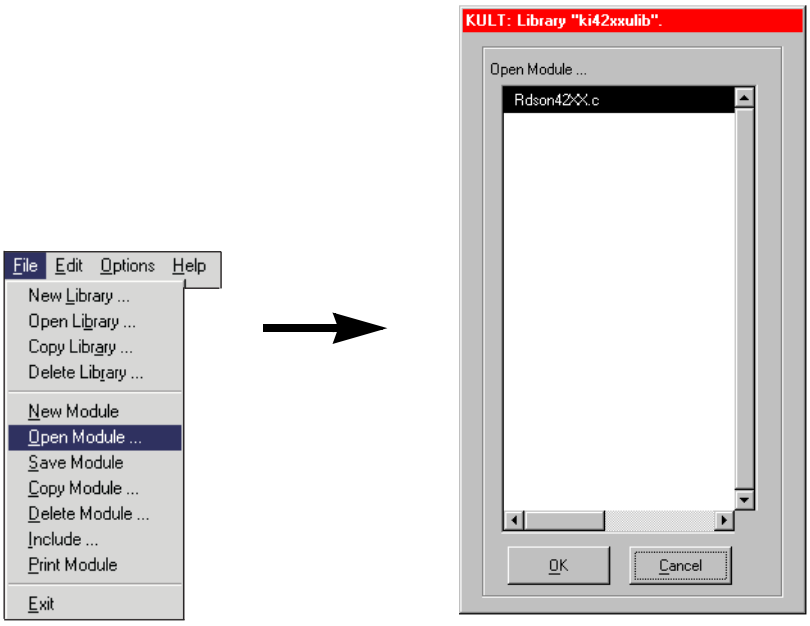


*B. Select “**ki42xxulib**” and click **OK***

Open the “Rdson42XX” user module

1. From the **File** menu, select the **Open Module** item (see Figure 2-22A).
2. From the **Open Module** window, select “**Rdson42XX.c**” as shown in Figure 2-22B, and click **OK**. The “**Rdson42XX**” module will open.

Figure 2-22
Open “Rdson42XX” module



*A. Select **Open Module***

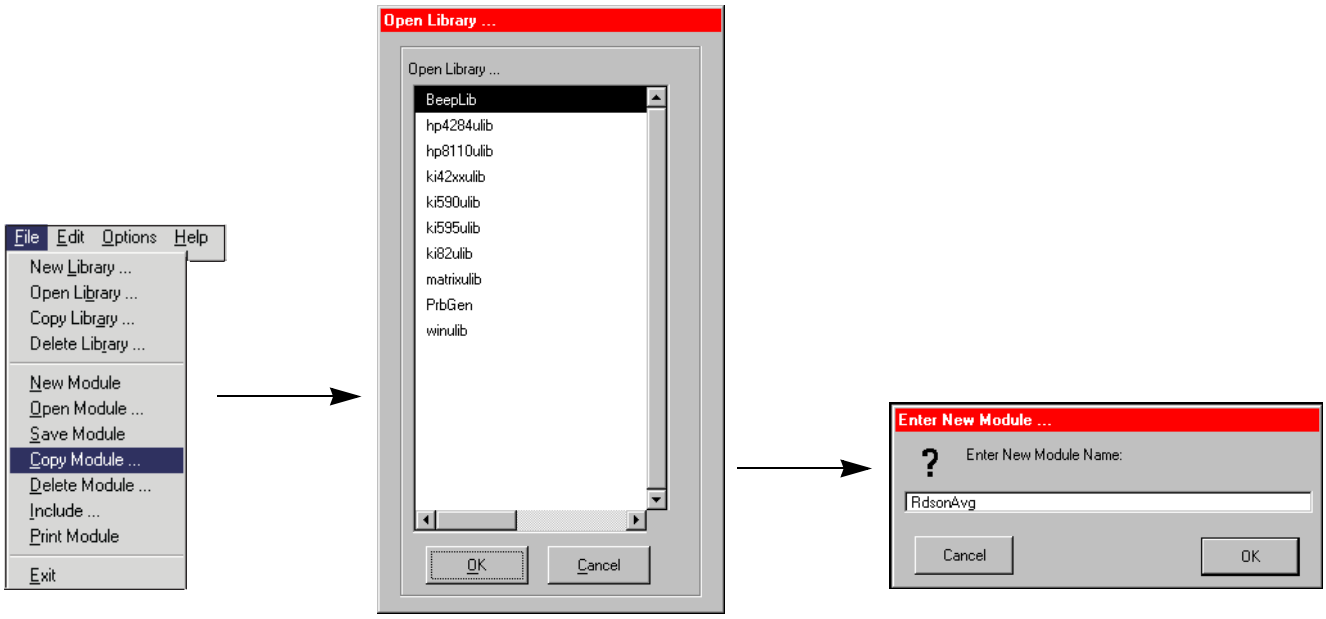
*B. Select “**Rdson42XX.c**” and click **OK***

Copy “Rdson42XX” to “RdsonAvg”

The new module will be created by copying the “Rdson42XX” module as “RdsonAvg” and then making the appropriate changes to the test module.

1. From the **File** menu, select the **Copy Module** item (see Figure 2-23A).
2. From the **Copy Module** window, select “**ki42xxulib**” as shown in Figure 2-23B and click **OK**. This selects the library for the module.
3. From the **Enter New Module Name** window, type in the name as shown in Figure 2-23C and click **OK**. A dialog box will remind you that the library using the new module will have to be built. Click **OK**.

Figure 2-23
Copy “Rdson42xx” module as “RdsonAvg”



*A. Select **Copy Module***

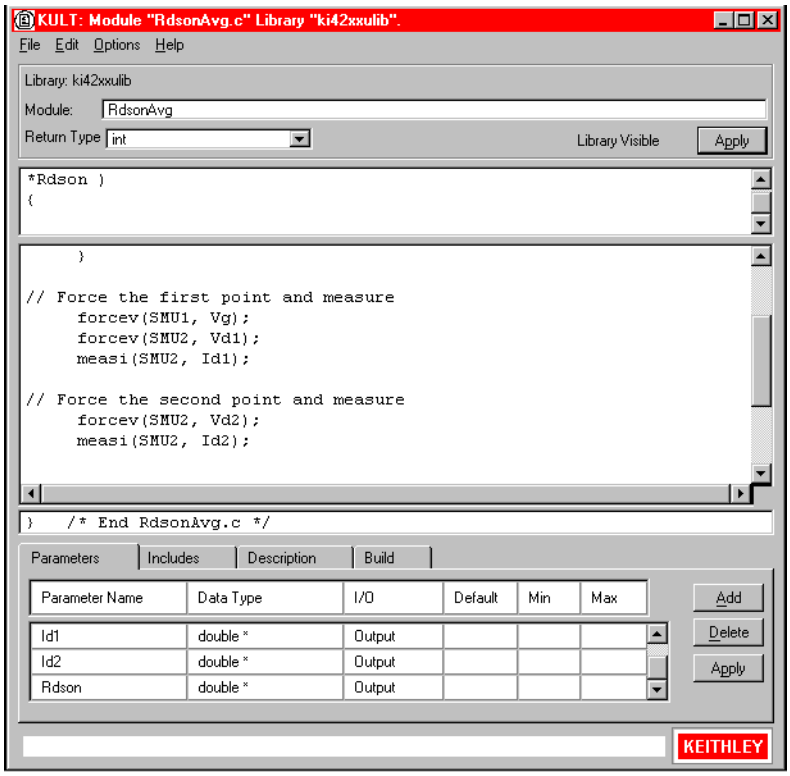
*B. Select “**ki42xxulib**” and click **OK***

*C. Type in “**RdsonAvg**” and click **OK***

Open and modify the "RdsonAvg" user module

From the **File** menu, select **Open Module**, and then select "**RdsonAvg.c**" from the **Open Module** window. The "RdsonAvg" module is shown in Figure 2-24.

Figure 2-24
KULT module window



Modify the user module code

The **measi** commands are to be replaced with **avgi** commands. While a **measi** command performs a single measurement, an **avgi** command performs a specified number of measurements, and then calculates the average reading. For example:

avgi (SMU2, Id1, 10, 0.01);

For the above command, **SMU2** performs 10 current measurements and then calculates the average reading (**Id1**). The **0.01** parameter is the delay between measurements (**10**ms).

The source code for the module is located in the module code area of the window. In this area, make the changes indicated in the following NOTE.

NOTE *For details on modifying a KULT program, refer to the 4200-SCS Reference manual.*

Figure 2-25
Program modifications

```
// Make the connections
   if ((GatePin > 0) && (DrainPin > 0) && (SourcePin > 0))
   {
      // Switch matrix used.
         conpin(SMU1, GatePin, 0);
         conpin(SMU2, DrainPin, 0);
         conpin(GND, SMU1L, SMU2L, SourcePin, BulkPin, 0);
   }

// Force the first point and measure
   forcev(SMU1, Vg);
   forcev(SMU2, Vd1);
   measi(SMU2, Id1);

// Force the second point and measure
   forcev(SMU2, Vd2);
   measi(SMU2, Id2);

// Clean up and clear the connections and the instrument
   devint();

   *Rdson = (Vd2-Vd1)/(Id2-Id1);    // Calculate Rdson

   return( OK );

// Program changes:

   *Rdson10 = (Vd2-Vd1)/(Id2-Id1);   // Calculate Rdson10
   avgi (SMU2, Id2, 10, 0.01);  // Perform averaged I measurement
   avgi (SMU2, Id1, 10, 0.01);  // Perform averaged I measurement
```

Change a parameter name

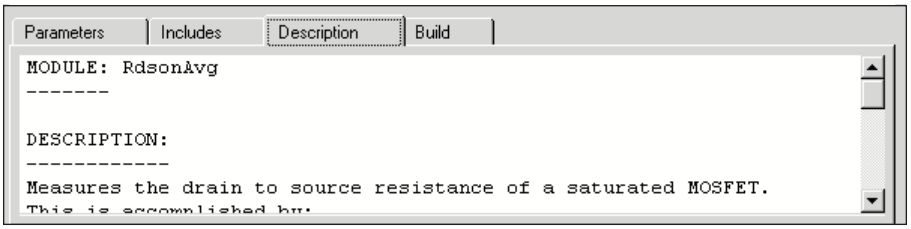
With the **Parameters** tab selected, the parameter names for the module are listed in a table located at the bottom of the window. Change the parameter name **"Rdson"** (shown in Figure 2-24) to **"Rdson10"**. After typing in the new parameter name, click **Apply** to enter the change.

NOTE *For details on the Parameters tab, refer to the 4200-SCS Reference manual.*

Change the module description

Click the **Description** tab to display the description for the module. Above **DESCRIPTION**, change **MODULE: Rdson42xx** to **MODULE: RdsonAvg** as shown in Figure 2-26. In addition, replace all occurrences of **Rdson** with **Rdson10**. In *KITE*, any UTMs that are connected to this module will show the text that is entered on the **Description** tab in *KULT*.

Figure 2-26
Module name for Description

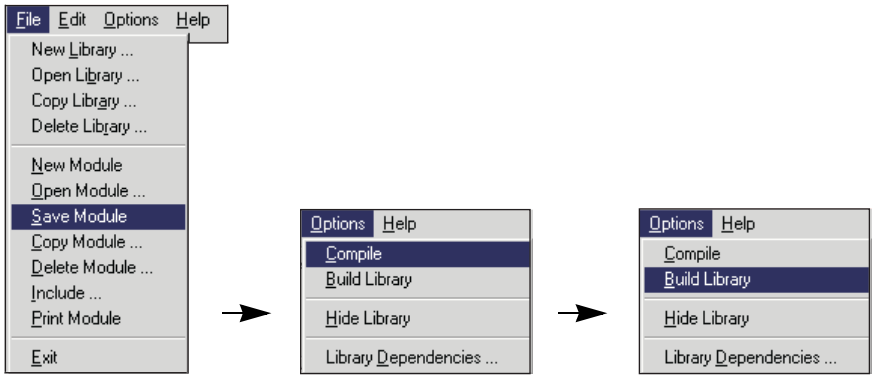


Save, compile, and build the modified library

The user module has to be saved and compiled. Finally, the library must be rebuilt to ensure that the new module is available for use by *KITE* UTMs. These operations are performed from the **File** and **Options** menus.

In the order shown in Figure 2-27, **save**, **compile**, and **build** the library. Note that *pop-up* windows will be displayed to indicate that the compile and library building operations are in process. For details, refer to the 4200-SCS Reference manual.

Figure 2-27
Save, compile, and build library



A. Select ***Save Module*** *B. Select* ***Compile*** *C. Select* ***Build Library***

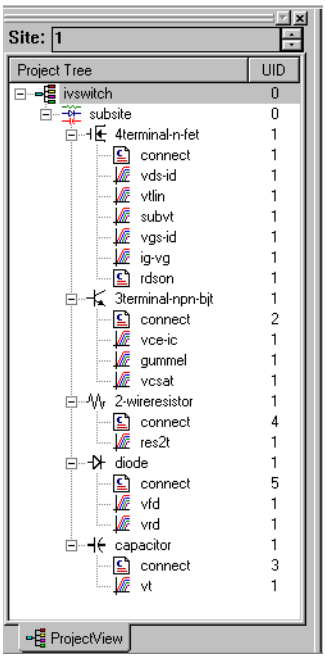
Add a new UTM to the “ivswitch” project

Open *KITE* and load the “ivswitch” project

1. From the desktop, open *KITE* by double-clicking the **KITE** icon.
2. Open the “**ivswitch**” project from the **File** menu.

The Project Navigator for the “ivswitch” project is shown in Figure 2-28. Notice that **rdson** is the last test for the “4terminal-n-fet” device.

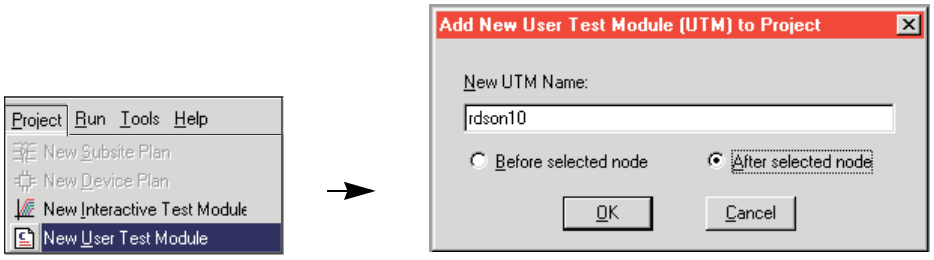
Figure 2-28
Project Navigator for “ivswitch” project



Add a new UTM

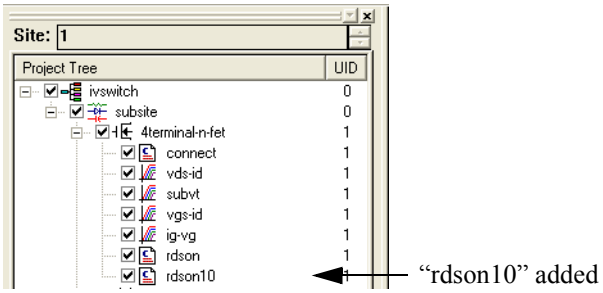
1. In the Project Navigator, single-click **rdson** to select it. This establishes the position for the new UTM.
2. From the **Project** menu, select **New User Test Module** (see Figure 2-29A).
3. In the **Add New User Test Module (UTM) to Project** window, type in the new name as shown in Figure 2-29B and click **OK**. Figure 2-30 shows the new UTM added to the Project Navigator.

Figure 2-29
Add new UTM



*A. Select **New User Test Module*** *B. Type in **“rdson10”** and click **OK***

Figure 2-30
"rdson10" added to Project Navigator

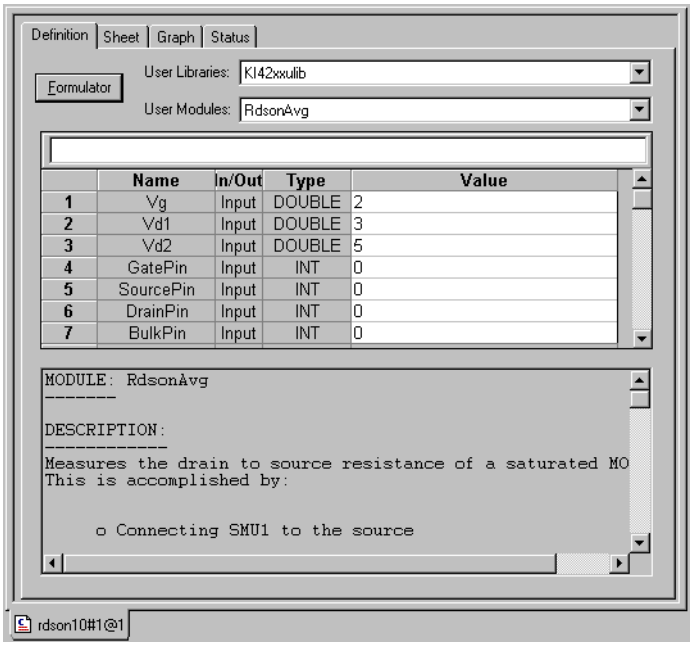


Connect the "rdson10" UTM to the "RdsonAvg" user module

In the Project Navigator, double-click **"rdson10"** test to open it. The test will open in the Workspace with the **Definition** tab blank.

Referring to Figure 2-31, select **"ki42xxulib"** from the drop-down menu for the **User Libraries** box in the UTM. Next, select **"RdsonAvg"** from the menu for the **User Modules** box.

Figure 2-31
"rdson10" UTM



Test description

The "rdson10" test measures the drain-to-source resistance of a saturated MOSFET. Using the user-input parameter values shown in Figure 2-31, the MOSFET is tested as follows when "rdson10" is executed:

1. Applies **2**V to the gate (**Vg**) to saturate the MOSFET.
2. Applies **3**V to the drain (**Vd1**) and performs 10 current measurements.
3. Averages the 10 current readings to yield a single reading (Id1).
4. Applies **5**V to the drain (**Vd2**) and performs 10 more current measurements.
5. Averages the 10 current readings to yield a single reading (Id2).
6. Calculates the drain-to-source resistance (rdson10) as follows:
   "Rdson10" = (Vd2-Vd1) / (Id2-Id1)

3 Controlling External Equipment

Section Topics List

Controlling external equipment overview, page 3-2
Controlling a CV Analyzer, page 3-5
- Connections, page 3-5
- KCON setup, page 3-6
- Create a new project, page 3-7
- Add a Subsite Plan, page 3-8
- Add a Device Plan, page 3-8
- Add a UTM, page 3-9
- Modifying the “cvsweep” UTM, page 3-11
- Executing the test, page 3-11

Controlling a pulse generator, page 3-12
- Test system connections, page 3-12
- KCON setup, page 3-13
- Open the “ivpgswitch” project, page 3-15
- Description of tests, page 3-16
- Running the test sequence, page 3-19
- Compare the test results, page 3-19

Controlling a probe station, page 3-21
- Prober control overview, page 3-22
- Test system connections, page 3-23
- KCON setup, page 3-23
- Probe station configuration, page 3-26
- Open the “probesubsites” project, page 3-26
- Open the project plan window, page 3-27
- Test descriptions, page 3-27
- Running the test sequence, page 3-31
- Test data, page 3-32
- Running individual plans or tests, page 3-32

In this section, you will learn the following:

- **Controlling External Equipment Overview** — Generically describes how external instruments are controlled by the Model 4200-SCS.
- **Controlling a CV Analyzer** — Demonstrates how to create a *KITE* project that uses a Keithley Model 590 CV Analyzer to acquire CV data from a MOS capacitor.
- **Controlling a Pulse Generator** — Demonstrates how to use the "ivpgswitch" *KITE* project to control an HP Model 8110A/81110A Pulse Generator. The pulse generator is used to stress a semiconductor device and the effects of the stress are then analyzed.
- **Controlling a Probe Station** — Demonstrates how to use the "probesubsites" *KITE* project to five identical sites (or die or reticles) on a semi-conductor wafer. Each test site is comprised of two subsites (or test element groups). Therefore, the wafer will be probed a total of 10 times.

The following equipment is required to complete this tutorial and obtain data that functionally correlates with the sample data provided with the sample projects.

- 1 - Keithley Model 4200-SCS with a total of three SMUs (PreAmps not required)
- 1 - Keithley Model 590 CV Analyzer
- 1 - Hewlett Packard 8110A/81110A Pulse Generator
- 1 - Keithley Model 707 or 708 Switch Matrix
- 1 - Keithley Model 7072 or 7174 8×12 matrix card
- 1 - Keithley Model 8006 Component Test Fixture
- 1 - Probe station (manual or supported semi-automatic) and a wafer containing test devices (MOS capacitor, N-channel MOSFET, and NPN bi-polar transistor)
- 2 - Keithley Model 4801 BNC cables
- 1 - Keithley Model 7078-TRX-BNC adapter
- 1 - Keithley Model 8007-GND-3 cable
- 4 - Keithley Model 4200-MTRX-X cables (0 if using PreAmps)
- 8 - Keithley Model 4200-TRX-X cables (11 if using PreAmps)
- 2 - Keithley Model 7007 GPIB cables
- 1 - Keithley Model 236-ILC-3 safety interlock cable

Controlling external equipment overview

In general, the Model 4200-SCS can control any external instrument or component connected to either of the following communication interfaces:

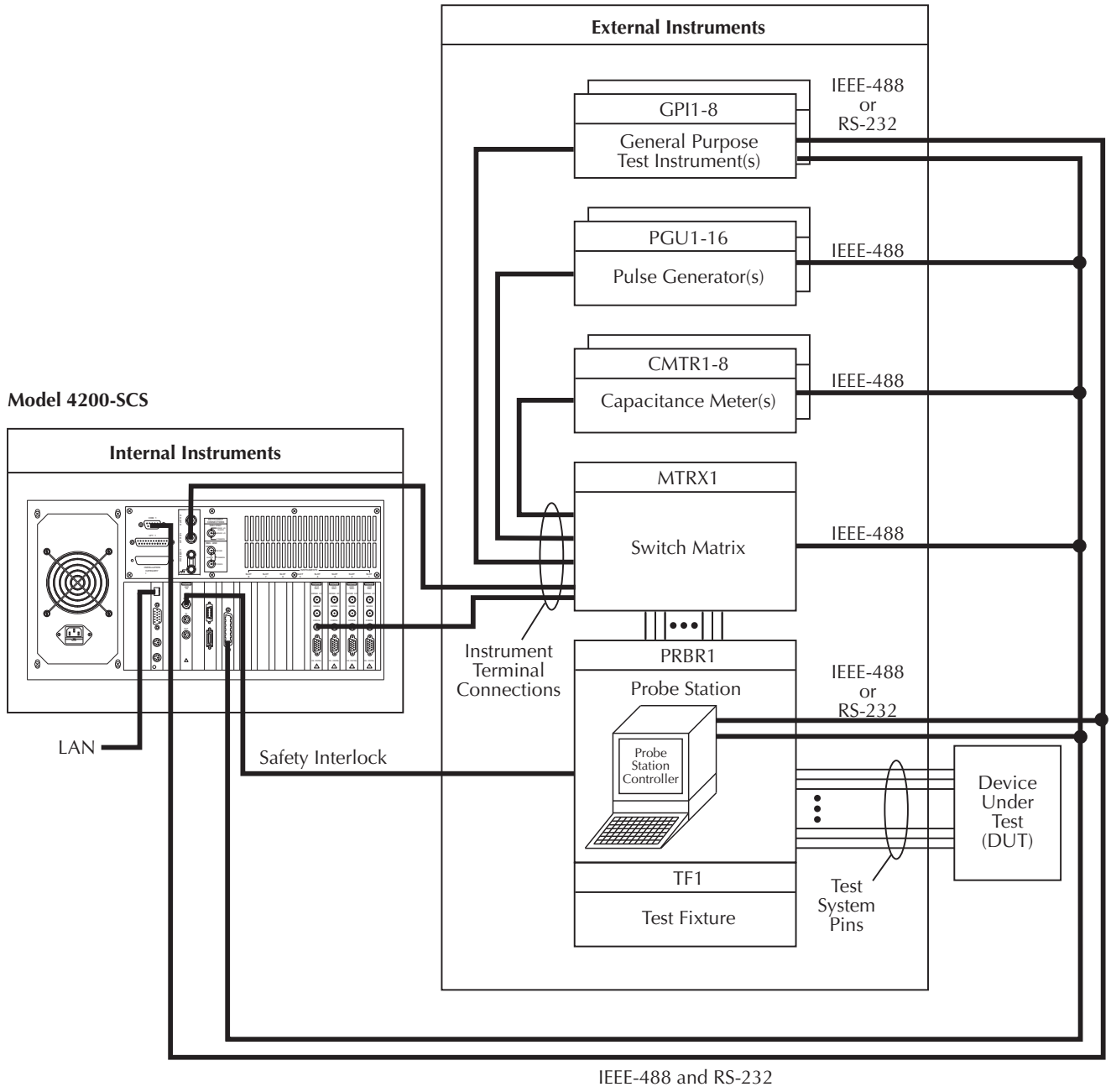
- IEEE-488 (GPIB) bus
- RS-232 (COM1) port

When an external instrument is added to the system configuration, it is grouped into one of the following categories:

- Switch Matrix
- Capacitance Meter
- Pulse Generator
- Probe Station or Test Fixture
- General Purpose Test Instrument

This is illustrated in Figure 3-1. The properties associated with each instrument category are further discussed in the Reference sections.

Figure 3-1
System configuration with external instruments



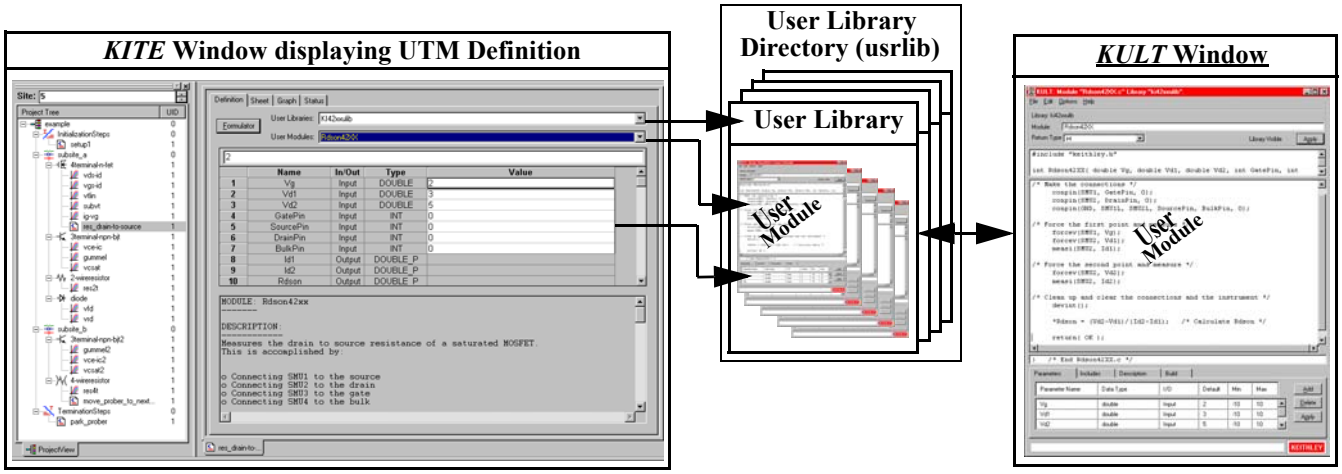
User modules are utilized to access these communication interfaces, and hence control external equipment. User modules are stored in user libraries which are created and maintained with the Keithley User Library Tool (*KULT*). See the Reference sections for additional information regarding creating and maintaining user libraries.

To execute a *KULT* user module in *KITE*, you create a *KITE* User Test Module (UTM) and connect it to the user module. Once this user module is connected to the UTM, the following occurs each time *KITE* executes the UTM:

- *KITE* dynamically loads the user module and the appropriate user library.
- *KITE* passes the user-module parameters—stored in the UTM—to the user module.
- Data generated by the user module is returned to the UTM for interactive analysis.

Figure 3-2 below illustrates the relationships between user libraries, user modules, UTMs, *KITE*, and *KULT*.

Figure 3-2
Relationships between KULT and KITE and between user libraries, user modules, and UTMs



Keithley provides a number of standard user libraries to control external equipment typically used in semiconductor characterization applications. Standard libraries of user modules for the following equipment are provided:

Table 3-1
Supported external equipment table

Category	Instrument	Keithley User Library / Additional Information
Switch Matrix	Keithley Model 707/707A Switching Matrix	matrixulib / 4200-SCS Reference manual
Capacitance Meter	Keithley Model 590 CV Analyzer	ki590ulib / 4200-SCS Reference manual.
	ki595ulib	Model 595 Quasistatic CV Meter Instruction Manual (document number 595-901-01)
	ki82ulib	4200-SCS Reference manual.
	Hewlett Packard Model 4284 LCR Meter	hp4284ulib / 4200-SCS Reference manual.
Pulse Generator	Hewlett Packard Model 8110A Pulse Generator	hp8110ulib / 4200-SCS Reference manual.
Probe Station	Karl Suss Model PA-200 Semiautomatic probe station	prbgen / 4200-SCS Reference manual.
	Micromanipulator Model 8860 Semiautomatic probe station	prbgen / 4200-SCS Reference manual.
	Manual and/or Fake probe station	prbgen / 4200-SCS Reference manual.
Test Fixture	Keithley Model 8006 Component Test Fixture	(not applicable)
	Keithley Model 8007 Semiconductor Test Fixture	(not applicable)
	Generic test fixture	(not applicable)
General Purpose Test Instrument	(any IEEE-488 or RS-232 controlled instrument or equipment)	(created by user)

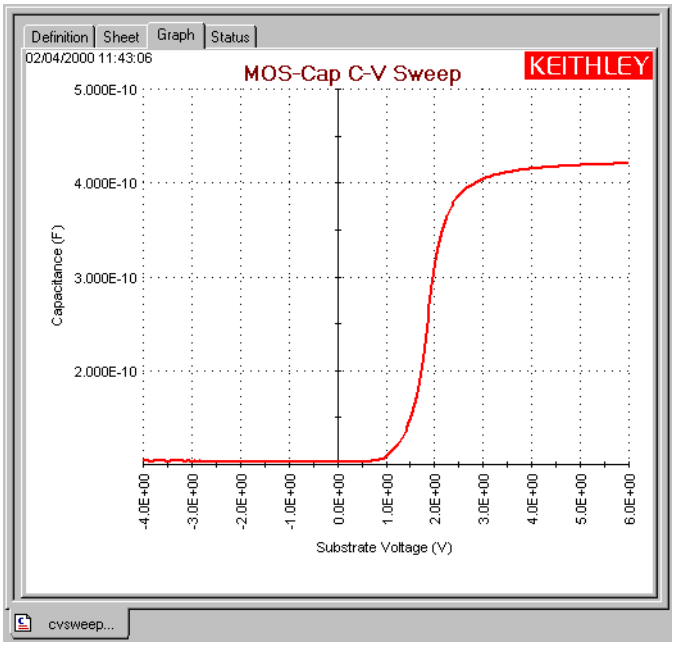
NOTE *Contact Keithley for the most up to date list of supported external equipment.*

Controlling a CV Analyzer

This tutorial demonstrates how to control a Keithley Model 590 CV Analyzer to acquire capacitance vs. voltage (CV) data from a MOS capacitor. This tutorial also demonstrates how to create a new *KITE* project. The new project will contain one User Test Module (UTM) that is connected to a standard CV user module supplied with each 4200-SCS.

The CV Analyzer will apply a linear staircase voltage sweep to a capacitor. A capacitance measurement will be performed on every voltage step of the sweep. Figure 3-3 shows a typical CV curve generated by this test.

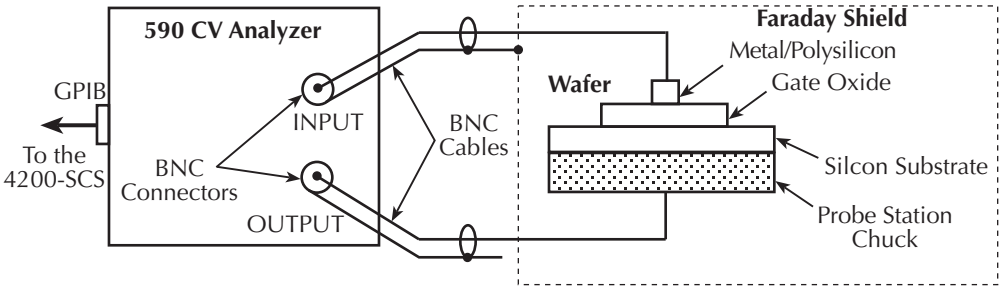
Figure 3-3
Typical CV curve



Connections

Connection details for the Model 590 CV Analyzer are provided in the 4200-SCS Reference manual. In general, the INPUT and OUTPUT connectors of the Model 590 are connected to the capacitor using Model 4801 (RG-58) BNC cables. The Model 590 is controlled by the 4200-SCS through the GPIB bus. Use a Model 7007 GPIB cable to connect the Model 590 to the Model 4200-SCS. Figure 3-4 provides an illustration of these connections.

Figure 3-4
Keithley Model 590 CV Analyzer DUT connections



KCON setup

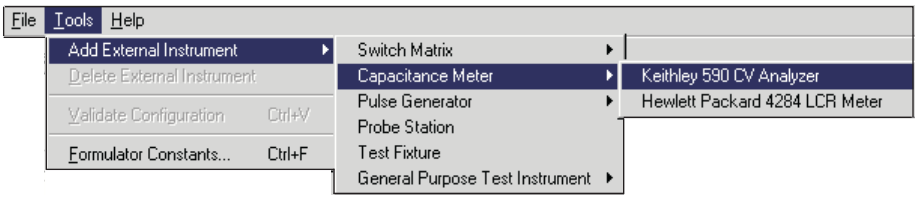
For this tutorial, the Model 590 CV Analyzer must be included in the 4200-SCS system configuration. The Keithley CONfiguration utility (*KCON*) is used to add external equipment and instrumentation to the test system. Follow the steps below to add the Model 590 to the system configuration using *KCON*:

NOTE *For details on KCON, refer to the 4200-SCS Reference manual.*

Step 1. Start *KCON*. Double click on the **KCON** icon or use the **Start** menu, **Start -> Programs -> Keithley -> KCON**.

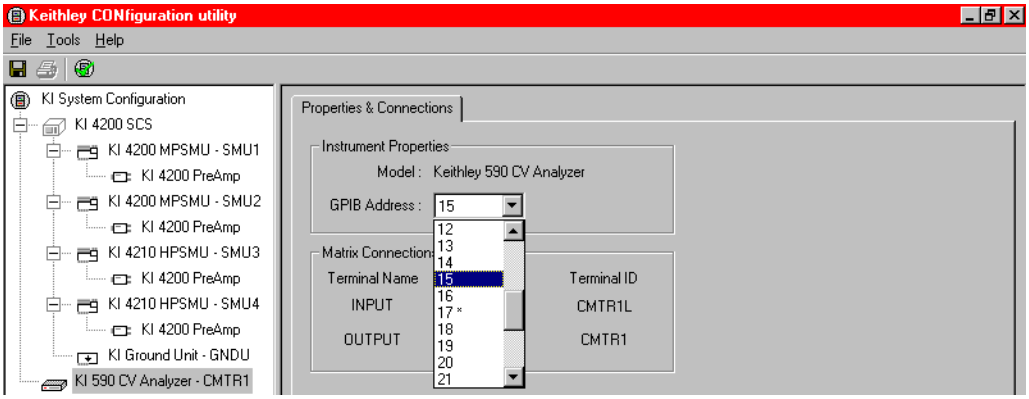
Step 2. Add the Keithley Model 590 CV Analyzer to the system configuration using the **KCON Tools** menu as illustrated in Figure 3-5.

Figure 3-5
Adding a Keithley 590 CV Analyzer to the system configuration



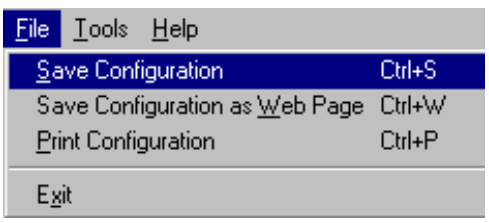
Step 3. Set the GPIB address for the Model 590 by selecting the **KI 590 CV Analyzer - CMTR1** in the Configuration Navigator and entering the appropriate GPIB address on the **Properties & Connections** tab. This is illustrated in Figure 3-6.

Figure 3-6
Setting the Model 590 GPIB address



Step 4. **Save** the configuration using the *KCON* **File** menu as illustrated in Figure 3-7.

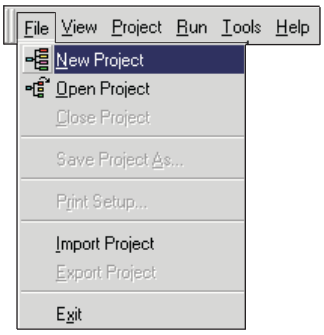
Figure 3-7
Saving the system configuration



Create a new project

On the *KITE* toolbar, select **New Project** from the **File** menu (see Figure 3-8) to open the **Define New Project** window. The new project definition window is shown in Figure 3-9A.

Figure 3-8
New Project menu selection

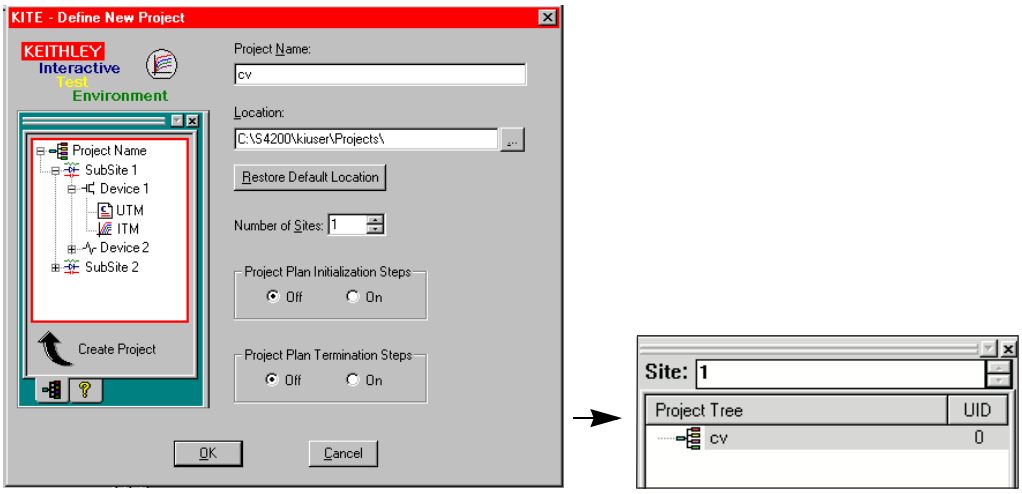


Type in the name of the project (**cv**) and define it as shown in Figure 3-9A. The directory path shown in the **Location** box is the default location where the factory defined projects are located. Make sure the specified **Number of Sites** is **1**, and initialization and termination steps are **Off**.

With the project defined as shown in Figure 3-9A, click the **OK** button at the bottom of the window. The project name will appear in the Project Navigator as shown in Figure 3-9B.

NOTE *For details on creating a project, refer to the 4200-SCS Reference manual.*

Figure 3-9
Define new project



A. Define New Project window *B. Project Navigator*

Add a Subsite Plan

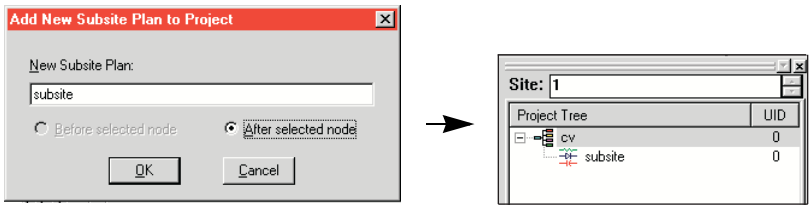
A subsite, or test element group, is a collection of devices to be tested. Open the **Add New Subsite Plan to Project** window by clicking the **Add new Subsite Plan** button on the toolbar (see Figure 3-10A). It can also be opened by clicking the **New Subsite Plan** item on the **Project** menu (see Figure 3-10B).

Figure 3-10
Add a new Subsite Plan to a KITE project

*A. **Add new Subsite Plan** button* *B. **Project** menu*

With the **Add New Subsite Plan to Project** window open (see Figure 3-11A), type in the name **subsite** and click **OK**. The Subsite Plan appears in the Project Navigator as shown in Figure 3-11B.

Figure 3-11
Add a new Subsite Plan

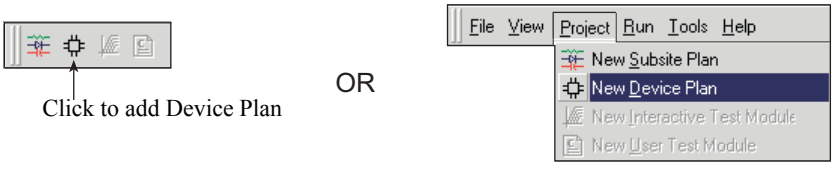


A. Window to specify the Subsite Plan name *B. Project Navigator*

Add a Device Plan

1. A Device Plan is a collection of tests to be performed on a particular device. **Open the Add New Device Plan to Project** window by clicking the **Add new Device Plan** button on the toolbar (see Figure 3-12A). It can also be opened by clicking the **New Device Plan** item on the **Project** menu (see Figure 3-12B).

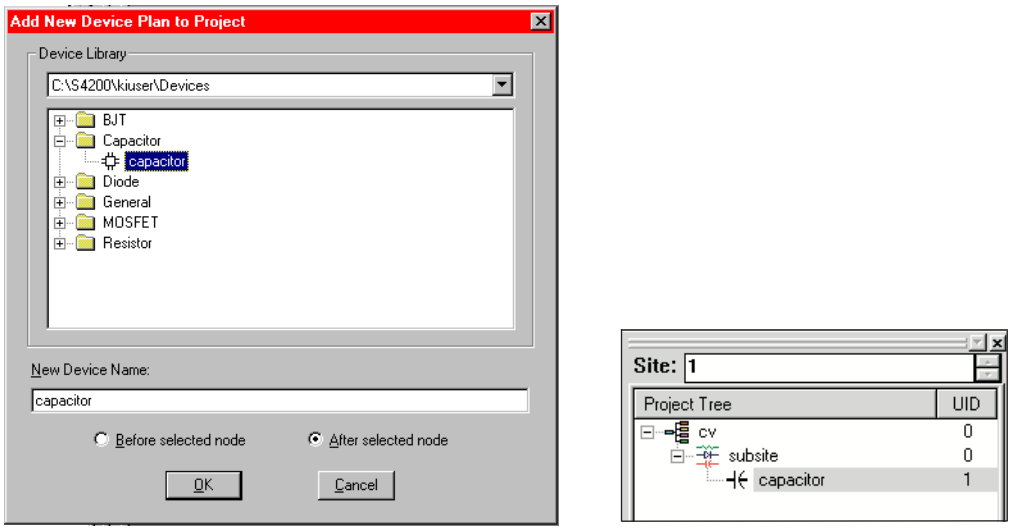
Figure 3-12
Add a new Device Plan to a KITE project



*A. **Add new Device Plan** button* *B. **Project** menu*

2. In the window to add a Device Plan (Figure 3-13A), double-click the **Capacitor** folder to open it, and then click **capacitor** to select that Device Plan name.
3. With the **capacitor** Device Plan selected, as shown in Figure 3-13A, click **OK** at the bottom of the window. The Device Plan will appear in the Project Navigator as shown in Figure 3-13B.

Figure 3-13
Add a Device Plan



A. Window to specify Device Plan name *B. Project Navigator*

Add a UTM

The “cvsweep” UTM is added to the new project by copying it from the default test library (`C:\S4200\kiuser\tests`) as follows:

1. In the Project Navigator, double-click on the **capacitor** device to open the Device Plan window.
2. On the **Sequence** tab of the Device Plan window, use the **Test Library** pull-down menu to select the default test library as shown in Figure 3-14.
3. Double-click the **Capacitor** folder to open it and display the available tests for that device. Figure 3-15A shows the **Capacitor** folder opened.
4. For the **Capacitor** folder, click “**cvsweep**” to select it. Figure 3-15A shows “cvsweep” selected.
5. Click **Copy** to place the test in the **Test Sequence Table**. Figure 3-15A shows “cvsweep” copied into the **Test Sequence Table**.
6. At the bottom of the Device Plan window, click **Apply** to copy the test into the Project Navigator. Figure 3-15B shows the “cvsweep” UTM added to the project.
7. If desired, the Device Plan window may be closed by pressing the close (**X**) button. The close (**X**) button is located on the right, above the Device Plan window.

Figure 3-14
Default test library folders

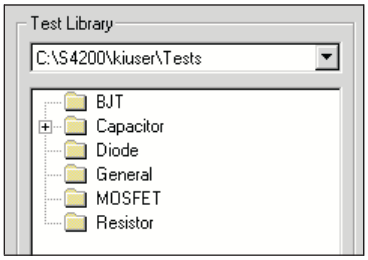
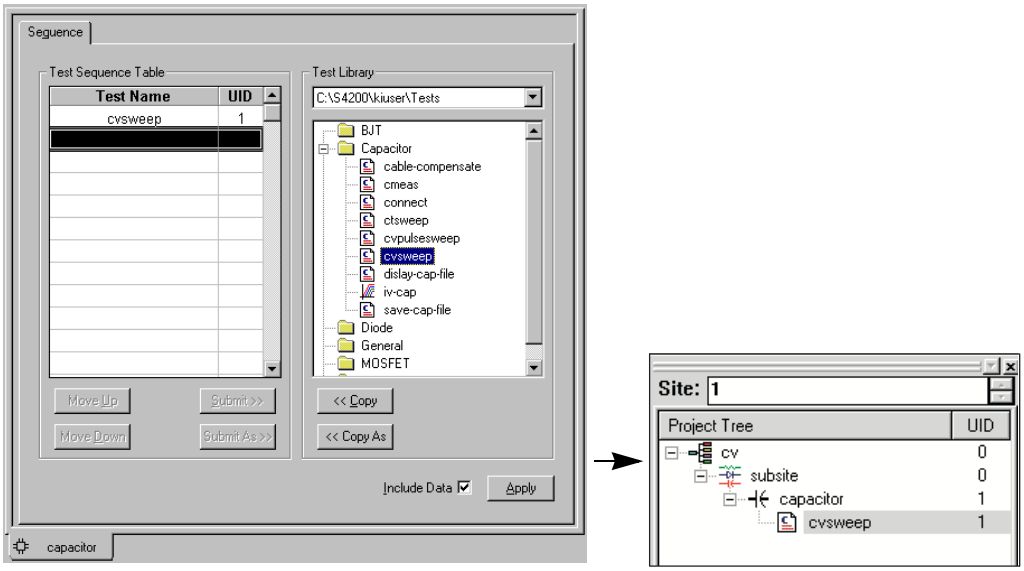


Figure 3-15
Add the "cvsweep" UTM



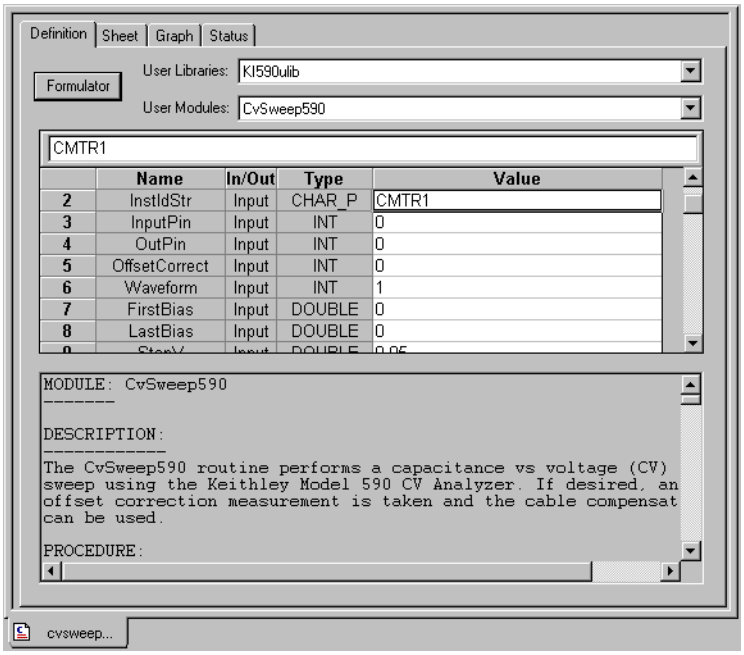
A. Window to specify the test module name *B. Project Navigator*

Modifying the “cvsweep” UTM

The default “cvsweep” parameters will sweep the voltage from -4V to +6V. If these parameters are acceptable, proceed to “Executing the test” on page 3-11. To modify the parameters, perform the following steps:

1. In the Project Navigator, double-click on the “**cvsweep**” UTM to open it. The window in Figure 3-16 will be displayed:

Figure 3-16
“cvsweep” UTM



2. Click on the **Definition** tab and make the desired parameter changes to the test.

NOTE *For details on the “cvsweep” UTM, refer to the 4200-SCS Reference manual.*

Executing the test

Since this new project has only one Subsite Plan and only one Device Plan, the test can be run from any level in the Project Navigator. To run the “**cvsweep**” test, simply click the green **Run** button. After the test is finished, use the **Sheet** and **Graph** tabs to view and analyze the results.

NOTE *The 4200-SCS also supports the Keithley Model 595 Quasistatic C-V Meter and the Keithley Model 82-WIN Simultaneous C-V System. For more information, refer to the 4200-SCS Reference Manual.*

Controlling a pulse generator

This tutorial demonstrates how to control a pulse generator to stress a semiconductor device and analyze the effects of the stress. The applied stress is a burst of 3.5V pulses across the gate-substrate (bulk) terminals of an N-Channel MOSFET. The basic test sequence is as follows:

1. Measure the transfer characteristics of the device before the stress.
2. Apply a stress burst of 3.5V pulses.
3. Measure the transfer characteristics of the device after the stress.

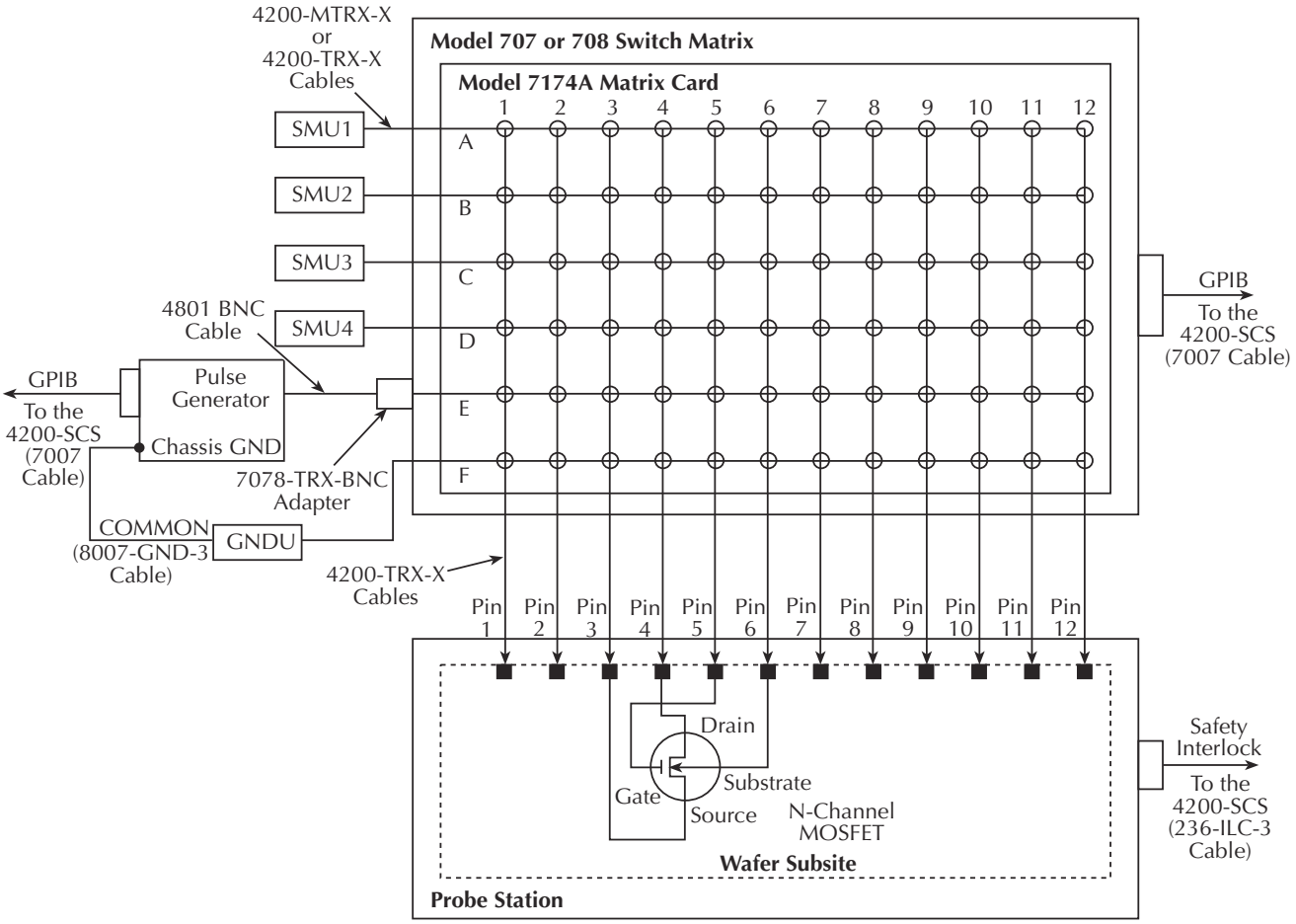
The *after-stress* characteristics can then be compared to the *before-stress* characteristics.

Test system connections

A typical test system for this application is shown in Figure 3-17. As shown, the Model 4200-SCS, HP Model 8110A/81110A Pulse Generator (PGU), and the DUT are connected to the Model 7174A Low Current Matrix Card. User Test Modules (UTMs) are used to control the switch matrix and the PGU. For details on SMU, GNDU (Ground Unit) and matrix card connections, refer to the 4200-SCS Reference manual.

The Model 7174A matrix card is installed in the Model 707/707A or Model 708/708A Switching Matrix. The switch matrix and PGU are controlled through the GPIB. Use the Model 7007 GPIB cables to connect the switch matrix and PGU to the Model 4200-SCS. For details on GPIB connections, refer to the 4200-SCS Reference manual.

Figure 3-17
Test system for "ivpgswitch" project



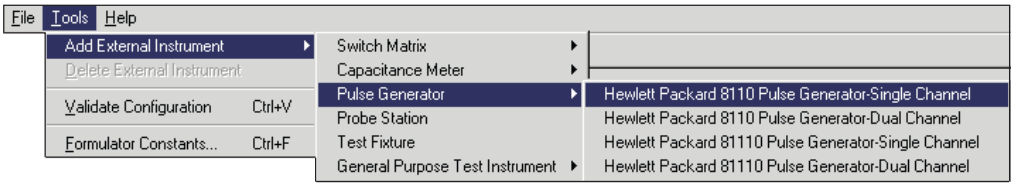
KCON setup

For this tutorial, a Hewlett Packard Model 8110A/81110A Pulse Generator, Keithley Model 707A Switching Matrix, Keithley Model 7174A Low Current Matrix Card, and a test fixture must be added to the system configuration. The Keithley CONfiguration utility (*KCON*) is used to add external equipment and instrumentation to the test system. Follow the steps below to add these components to the system configuration. Detailed information regarding *KCON* can be found in the 4200-SCS Reference manual.

Step 1. Start *KCON*. Double click on the **KCON** icon or use the **Start** menu, **Start -> Programs -> Keithley -> KCON**.

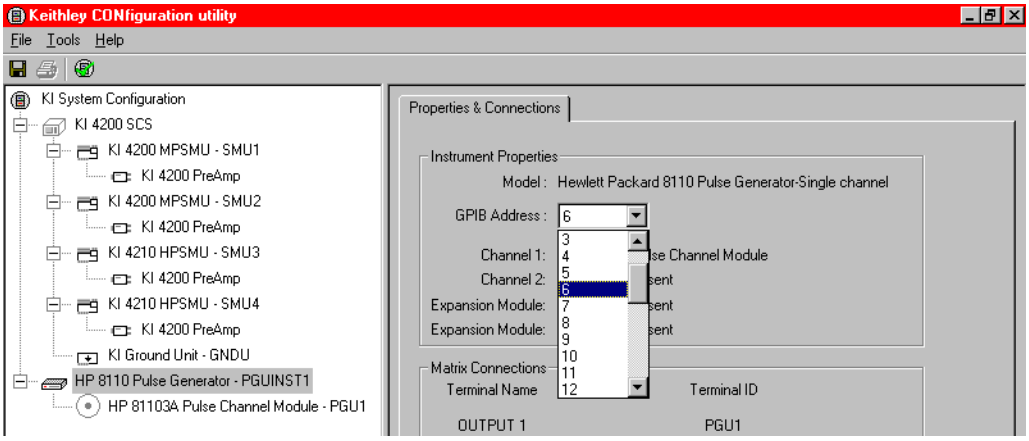
Step 2. Add the Hewlett Packard Model 8110A/81110A Pulse Generator to the system configuration using the **KCON Tools** menu as illustrated in Figure 3-18.

Figure 3-18
Adding a pulse generator



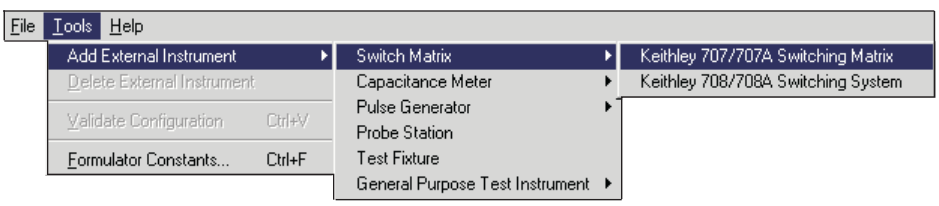
Step 3. Set the GPIB Address for the pulse generator by selecting it in the Configuration Navigator and entering the appropriate **GPIB Address** on the **Properties & Connections** tab. This is illustrated in Figure 3-19.

Figure 3-19
Pulse generator configuration



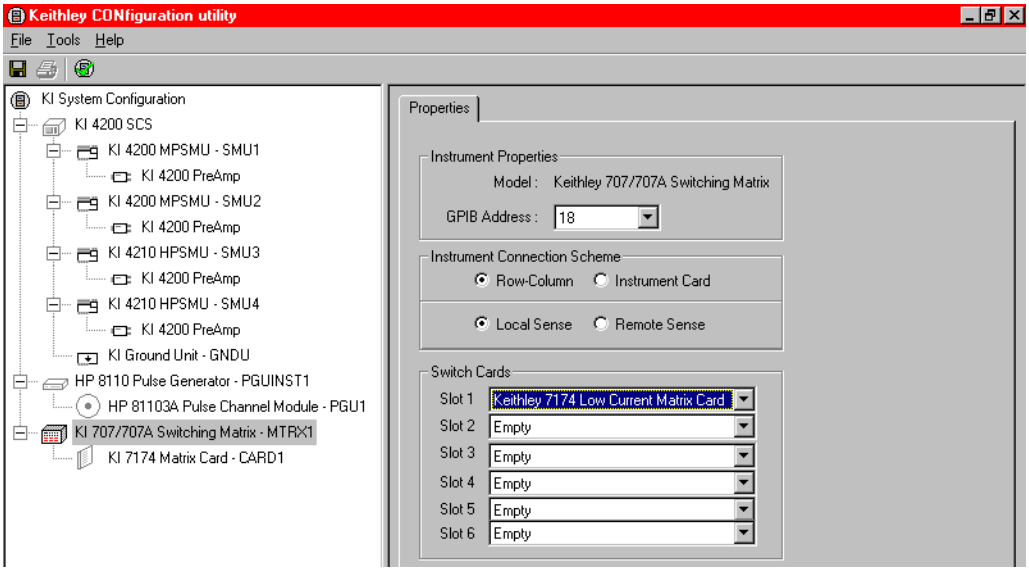
Step 4. Add the **Keithley Model 707/707A Switching Matrix** to the system configuration using the **KCON Tools** menu as illustrated in Figure 3-20.

Figure 3-20
Adding a switch matrix



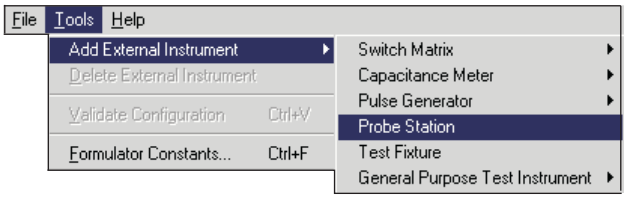
Step 5. Set the GPIB Address for the switch matrix and add the 7174A matrix card in Slot 1 as illustrated in Figure 3-21.

Figure 3-21
Configuring the switch matrix



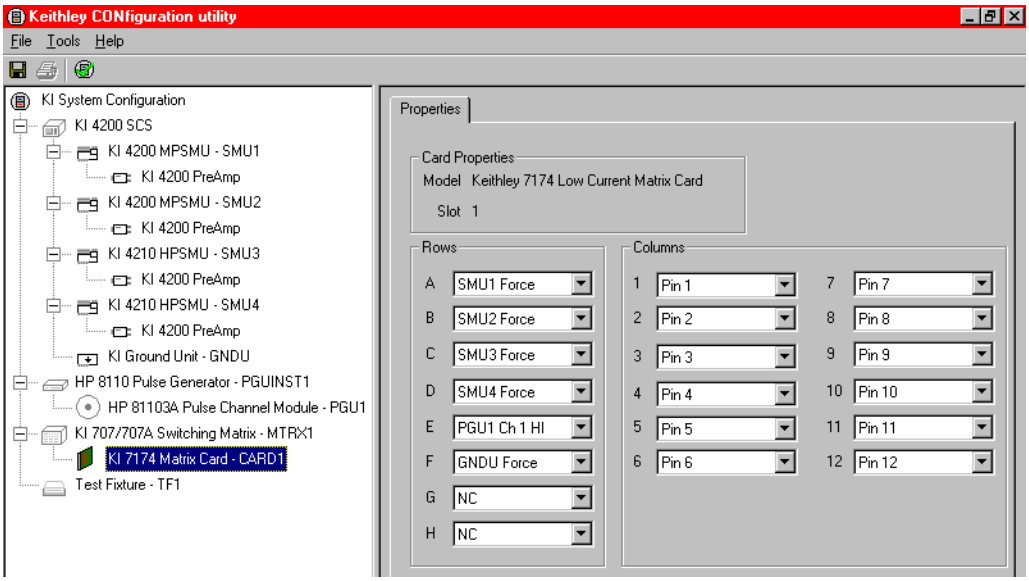
Step 6. Add a manual probe station to the system configuration using the **KCON Tools** menu as illustrated in Figure 3-22. If a test fixture is already part of the configuration, it must be removed before the probe station can be added. To remove any external component from the system configuration, select it in the Configuration Navigator and press the **DELETE** key.

Figure 3-22
Adding a probe station



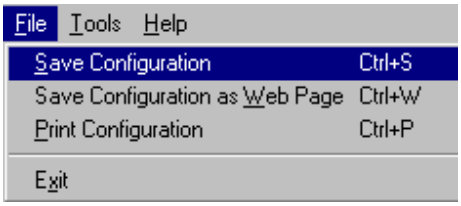
Step 7. Connect the instrument terminals and probe station pins to the switch matrix by selecting the **KI 7174 Matrix Card - CARD1** in the Configuration Navigator and configuring it as illustrated in Figure 3-23. Detailed information regarding switch matrix configuration can be found in the 4200 Reference Manual - Appendix B, “Using Switch Matrices”.

Figure 3-23
Connecting the switch matrix



Step 8. Save the configuration using the **KCON File** menu as illustrated in Figure 3-24.

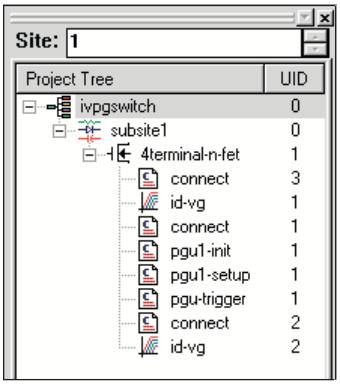
Figure 3-24
Saving the system configuration



Open the “ivpgswitch” project

Open the “**ivpgswitch**” project from the **File** menu (select **Open Project**). The Project Navigator for the “ivpgswitch” project is shown in Figure 3-25.

Figure 3-25
Project Navigator - “ivpgswitch” project



Description of tests

The project tests are described in the same order that they are presented in the Project Navigator.

First "connect" test

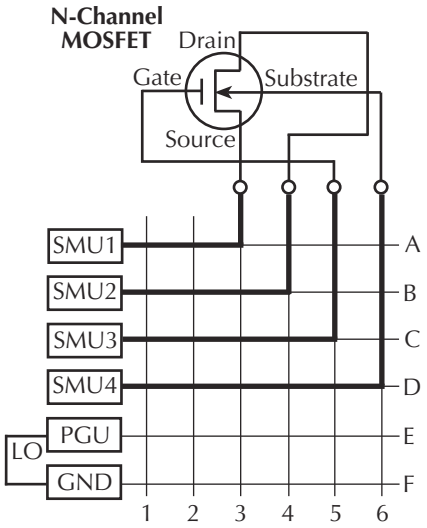
The first test, "connect", is a UTM that connects the device to the four SMUs. In the Project Navigator, double-click the first **"connect"** UTM to open it. Figure 3-26 shows the parameters that connect the device to the SMUs. Note that the first parameter (line **1**) opens any relays that may have been closed by a previous test. For the other parameter shown in Figure 3-26, the device connects to the SMUs as shown in Figure 3-27.

NOTE *For details on the "connect" UTM, refer to the 4200-SCS Reference Manual.*

Figure 3-26
First "connect" test - connects the device to the SMUs

	Name	In/Out	Type	Value	
1	OpenAll	Input	INT	1	← *Opens all relays*
2	TermIdStr1	Input	CHAR_P	SMU1	← *Connects SMU1 to pin 3 of test fixture*
3	Pin1	Input	INT	3	
4	TermIdStr2	Input	CHAR_P	SMU2	← *Connects SMU2 to pin 4 of test fixture*
5	Pin2	Input	INT	4	
6	TermIdStr3	Input	CHAR_P	SMU3	← *Connects SMU3 to pin 5 of test fixture*
7	Pin3	Input	INT	5	
8	TermIdStr4	Input	CHAR_P	SMU4	← *Connects SMU4 to pin 6 of test fixture*
9	Pin4	Input	INT	6	

Figure 3-27
Signal paths for the pre and post stress tests



First "id-vg" test

The "id-vg" ITM measures the transfer characteristics of the N-channel MOSFET. The I_D vs. V_G data points are graphed. The test also calculates and graphs transconductance. This is the *before-stress* characterization test.

Second “connect” test

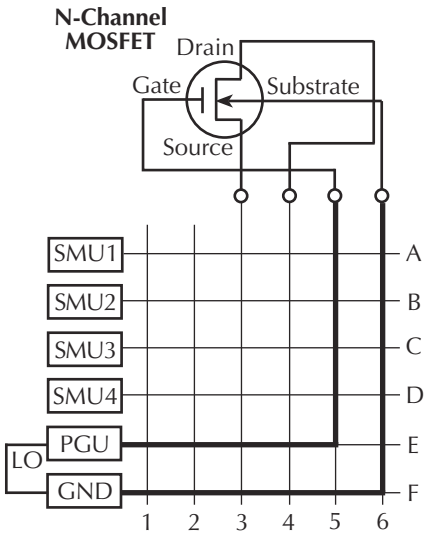
This “connect” UTM connects the device to the PGU and the Ground Unit (GNDU). In the Project Navigator, double-click the second **“connect”** test to open it. Figure 3-28 shows the parameters that connect the device to the PGU. Not shown is line **1** (**OpenAll**) that opens the relays closed by the previous “connect” test. Line **1** is shown in Figure 3-26.

For the parameters shown in Figure 3-28, the device connection pathways to the PGU and GNDU are shown in Figure 3-29. Keep in mind that if your physical matrix connections are different, you will have to change the connection parameters in the UTM accordingly.

Figure 3-28
Second “connect” test - connects the device to the PGU

	Name	In/Out	Type	Value
10	TermIdStr5	Input	CHAR_P	PGU ← *Connects PGU to pin 5 of test fixture*
11	Pin5	Input	INT	5
12	TermIdStr6	Input	CHAR_P	CMTR1
13	Pin6	Input	INT	0
14	TermIdStr7	Input	CHAR_P	CMTR1L
15	Pin7	Input	INT	0
16	TermIdStr8	Input	CHAR_P	GNDU ← *Connects GNDU to pin 6 of test fixture*
17	Pin8	Input	INT	6

Figure 3-29
Signal paths to apply the pulse stress



“pgu1-init” test

In the Project Navigator, double-click **“pgu1-init”** to open the test. This one parameter test (see Figure 3-30) initializes the PGU. For example, it disables the output, resets errors and sets triggering. More information on the initialized state is provided in the **DESCRIPTION** area of the **Definition** tab. For details on the UTMs for the pulse generator, refer to the 4200-SCS Reference manual.

Figure 3-30
PGU initialization

	Name	In/Out	Type	Value
1	InstIdStr	Input	CHAR_P	PGU1 ← *Initializes HP 8110*

"pgu1-setup" test

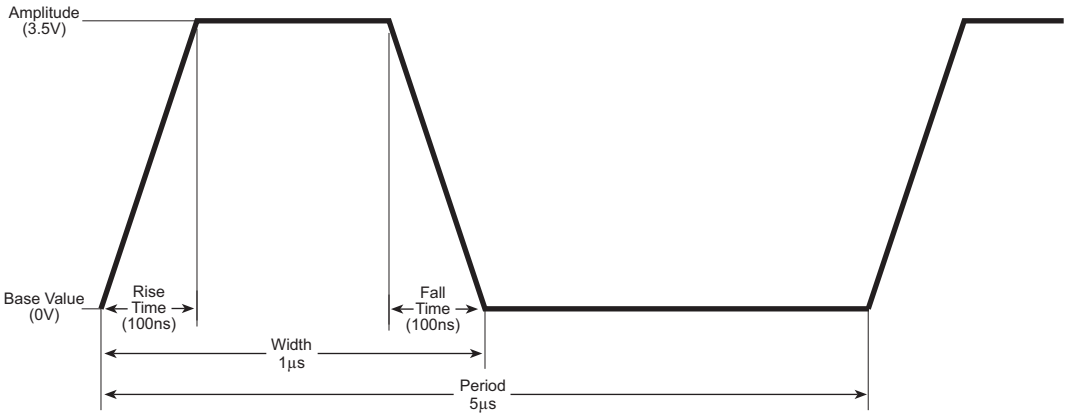
In the Project Navigator, double-click **"pgu1-setup"** to open the test. The complete parameter listing for the test is shown in Figure 3-31. These parameters to configure the PGU are explained in the **DESCRIPTION** area of the **Definition** tab.

Figure 3-32 shows the pulse that is configured by this test. Note that the pulse is not drawn to scale.

Figure 3-31
"pgu1-setup" - configure the PGU channel

	Name	In/Out	Type	Value
1	InstIdStr	Input	CHAR_P	PGU1
2	DelayTime	Input	DOUBLE	0
3	RiseTime	Input	DOUBLE	100e-09
4	FallTime	Input	DOUBLE	100e-09
5	Width	Input	DOUBLE	1.0e-06
6	Period	Input	DOUBLE	5e-06
7	BaseValue	Input	DOUBLE	0.0
8	Amplitude	Input	DOUBLE	3.500000e+000
9	OutImpedance	Input	INT	0
10	LoadImpedance	Input	DOUBLE	50
11	OutpEnable	Input	INT	1

Figure 3-32
PGU stress pulse specifications



"pgu-trigger" test

In the Project Navigator, double-click "**pgu-trigger"** to open the test. The 2-line parameter list for this test is shown in Figure 3-33. This test triggers the PGU to output 60,000 pulses to the N-channel MOSFET.

Figure 3-33
"pgu-trigger" test - trigger the burst of stress pulses

	Name	In/Out	Type	Value
1	InstIdStr	Input	CHAR_P	PGU1
2	Count	Input	INT	60000 ← *Triggers burst of pulses*

Third "connect" test

This "connect" test is the same as the first "connect" test. That is, it connects the device to the SMUs so that the transfer characteristics can be determined after applying the pulse stress (see Figure 3-26 and Figure 3-27).

Second “id-vg” test

This “id-vg” test is the same as the first “id-vg” test. That is, it measures the transfer characteristics of the N-channel MOSFET. This is the *after-stress* characterization test.

Running the test sequence

To run the test sequence, select (click) the “**4terminal-n-fet”** device in the Project Navigator, and then click the green **Run** button. The test sequence is summarized in Table 3-2.

Table 3-2
Test sequence for “ivpgswitch” project

	Test	Description
1	“connect”	Connects the MOSFET to the four SMUs.
2	“id-vg”	Measures the initial transfer characteristics of the MOSFET.
3	“connect”	Connects the MOSFET to the PGU.
4	“pgu1-init”	Initializes the PGU.
5	“pgu1-setup”	Configures the PGU output pulse.
6	“pgu-trigger”	Triggers the PGU to output a burst of pulses.
7	“connect”	Connects the MOSFET to the four SMUs.
8	“id-vg”	Measures the final transfer characteristics of the MOSFET.

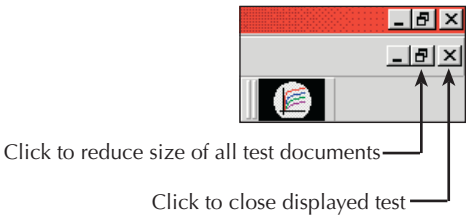
Compare the test results

A way to compare “id-vg” test results is to do a side-by-side visual inspection of the two graphs. In the Project Navigator, double-click the two “**id-vg”** tests to open them in the Workspace.

Close some UTMs - To reduce clutter, you may want to remove any other tests (UTMs) from the Workspace. Figure 3-34 shows the button to close a displayed test.

Make room for the graphs - To make room for the two graphs, (1) hide the Project Navigator to expand the size of the Workspace; and (2) reduce the size of the test documents. The close button (**X**) is located at the top right-hand corner of the Project Navigator. Figure 3-34 shows the button to reduce the size of the test documents in the Workspace.

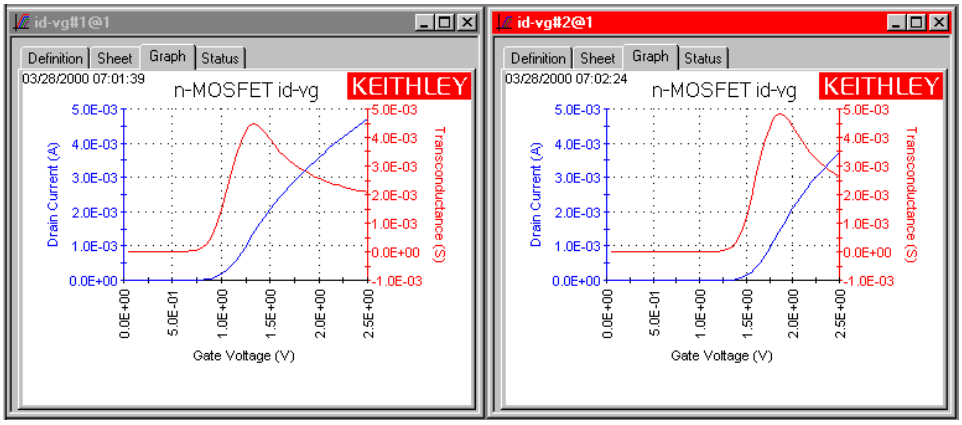
Figure 3-34
Buttons to close or reduce size of test documents



Position tests side-by-side - A test document is moved by clicking the title bar at the top of the document and dragging it to the desired location in the Workspace.

Display the graphs - The graph for each test is displayed by clicking the **Graph** tab. Figure 3-35 shows typical graphs for the two "id-vg" tests.

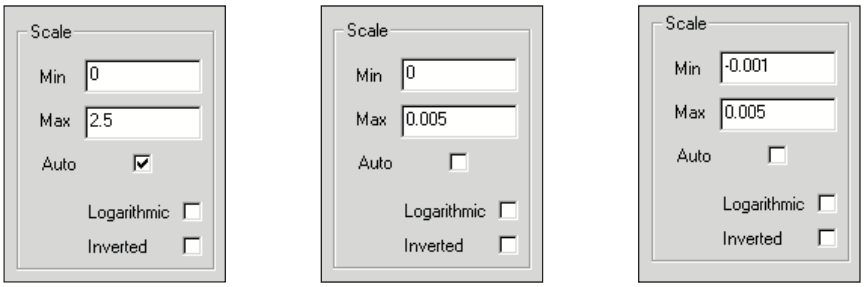
Figure 3-35
"id-vg" graphs



A. Before-stress graph *B. After-stress graph*

Scale settings - To effectively compare the two graphs, they must both have the same scale settings. Figure 3-36 shows the scale settings for the graphs in Figure 3-35. Scale settings for a graph are set by clicking the **Axis Properties** item in the **Graph** menu. A **Graph** menu is displayed by placing the mouse pointer in an open area of the graph, and then right-clicking the mouse. Keep in mind that there is a separate **Graph** menu (and **Axis Properties** window) for each graph.

Figure 3-36
Graph scale settings



A. X-axis *B. Y-axis* *C. Y2-axis*

Compare graphs - Visually inspect the two graphs for differences caused by the stress. You can also click the **Sheet** tabs and compare the data collected for the two tests.

Overlaying graphs

Another way to compare the two graphs is to lay the *after-stress* graph over the *before-stress* graph as follows:

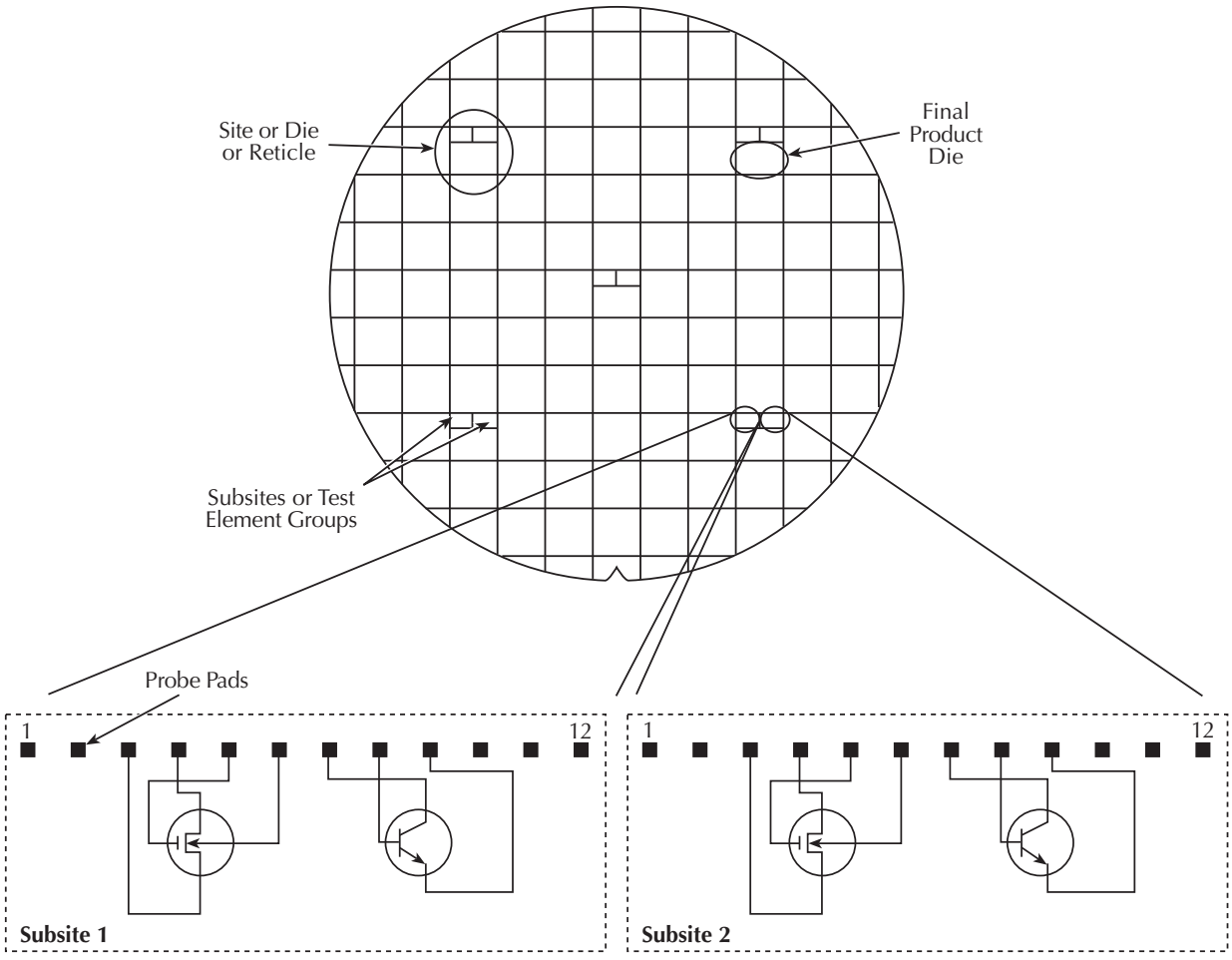
1. For the *after-stress* test, click the **Sheet** tab to display the data spreadsheet.
2. Select all five columns by clicking and dragging the mouse pointer from column **A** through column **E**. Press **CTRL + C** to copy those columns.
3. For the *before-stress* test, click the **Sheet** tab, and then the **Calc** tab (located at the bottom).

4. In the Calc spreadsheet, click cell **A1** to select it, and then press **CTRL + V**. This pastes the copied columns into the Calc spreadsheet. This *after-stress* data is now available to be graphed.
5. In the Calc spreadsheet, give the **DrainI** and **GM** columns new names to distinguish them as after-stress (AS) data. For example, change **DrainI** to **DrainI(AS)**, and change **GM** to **GM(AS)**.
6. Click the **Graph** tab for the present test (*before-stress*). In an open area of the graph, right-click the mouse to open the graph menu. In the graph menu, click **Define Graph** to open the **Graph Definition** window.
7. In the **Graph Definition** window, click the **Y1/DrainI**(**AS**) cell and the **Y2/GM**(**AS**) cell to select them, and click **OK**. The graph will now show the overlaid data.
8. From the graph menu, use the **Legend** and **Graph Properties** - **Series** items to add a legend and to change the line properties of the graph, if desired.

Controlling 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; a 4-terminal N-channel MOSFET and a 3-terminal NPN transistor. The subsites need not be identical, but for simplicity they are assumed to be the same. This is illustrated below in Figure 3-37.

Figure 3-37
Sample wafer organization



Prober control overview

NOTE *The information provided in this overview is a summary of the information provided in the 4200-SCS Reference manual.*

A probe station, like any other external instrument, is controlled by the 4200-SCS through user modules. Basic system connections are illustrated in Figure 3-1. A library of user modules, called `prbgen`, is provided with the 4200-SCS to facilitate prober control. This *generic* prober user library, developed and maintained by Keithley, allows *KITE* to control all supported probers in the same manner. Therefore, *KITE* projects utilizing `prbgen` will work with any prober supported by Keithley. Refer to Table 3-3 for the list of supported prober.

Table 3-3
Supported probers

Supported Probe Station	Additional Information
Karl Suss Model PA-200	Refer to the 4200-SCS Reference manual.
Micromanipulator Model 8860	
Manual (or Fake)	

NOTE *Contact Keithley for the most up to date list of supported probe stations.*

Sophisticated prober control software, available from each supported prober vendor, provides access to the full feature set of each prober. In all cases, this prober control software provides the ability to define a list of wafer locations to be probed. The 4200-SCS relies on the prober controller, and associated software, to maintain this probe list. The `prbgen` user modules communicate with the prober controller, through the GPIB bus or COM1 port in most cases, 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 where each probe location is physically located. Table 3-4 summarizes the user modules included in the `prbgen` prober control user library.

Table 3-4
prbgen user modules

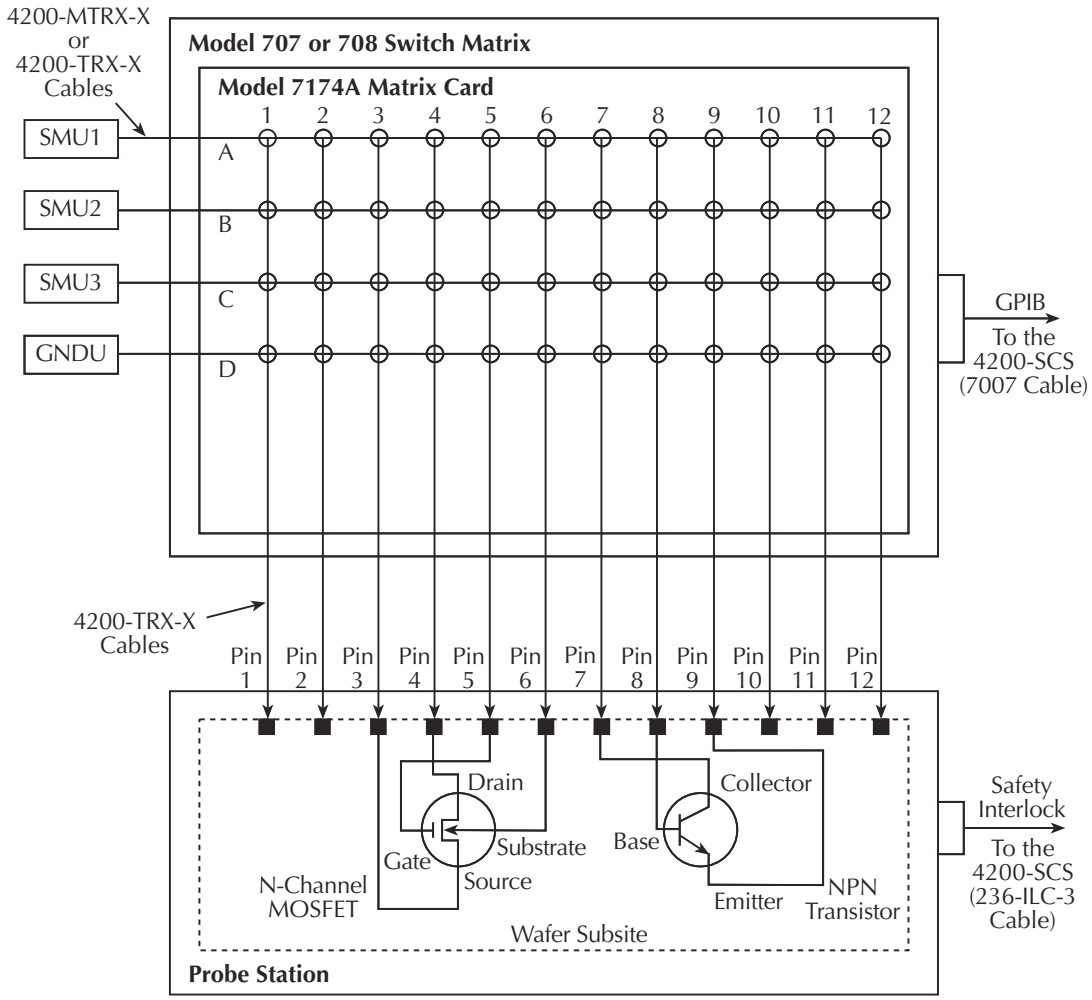
User Module	Description
PrInit	Initializes the prober driver and establishes the reference site (or die). All ITM or UTM data acquired by *KITE* will be tagged with [row, column] site coordinate information that is relative to the reference site.
PrChuck	Instructs the prober to move the probe station chuck up or down, making or breaking contact between the wafer and 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 a *KITE* project that utilizes the `prbgen` user library can be executed, the probe list must be created using the appropriate prober control software. Helpful instructions for creating the probe list for each supported prober are included in the 4200-SCS Reference sections. Refer to Table 3-3 for additional information.

Test system connections

A typical test system for this tutorial is shown in Figure 3-38. As shown, the Model 4200-SCS and probe station are connected to a 7174A matrix card. The matrix card is installed in the switch matrix and the switch matrix and probe station are controlled through the GPIB bus. For connection details as well as information on the Keithley CONfiguration Utility, refer to the 4200-SCS Reference manual.

Figure 3-38
System configuration for the "probesubsites" project



KCON setup

For this tutorial, the following external equipment must be added to the system configuration:

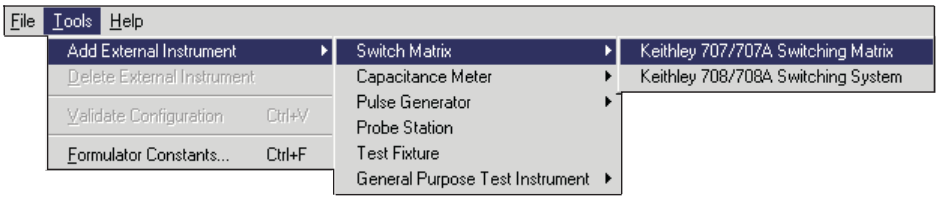
- Switch matrix
- Matrix card
- Probe station

The Keithley CONfiguration utility (*KCON*) is used to add external equipment and instrumentation to the test system. Below is a step by step procedure for adding the necessary equipment to the system configuration:

Step 1. Start *KCON*. Double click on the **KCON** icon or use the **Start** menu, **Start -> Programs -> Keithley -> KCON**.

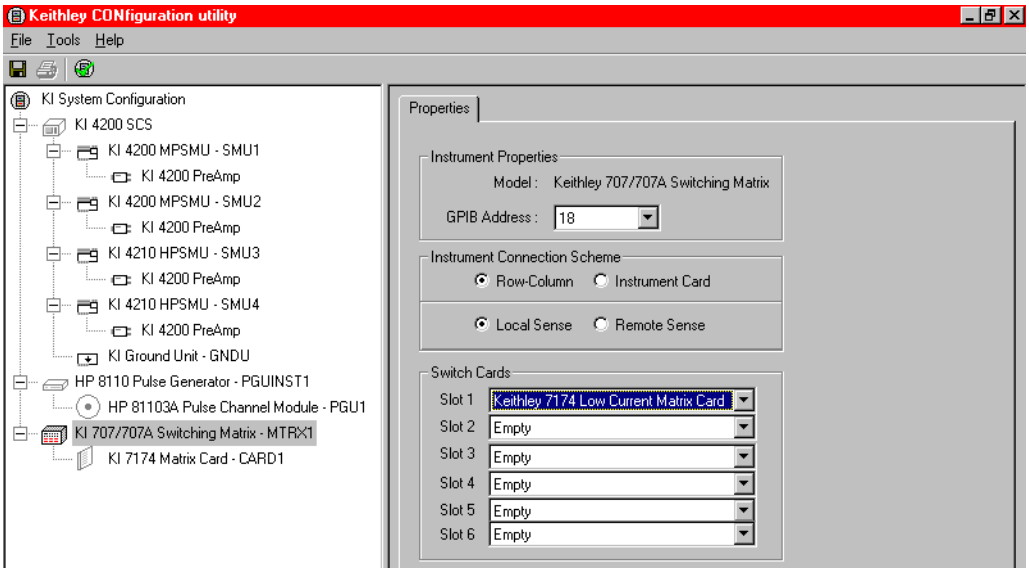
Step 2. Add the **Keithley Model 707/707A Switching Matrix** to the system configuration using the **KCON Tools** menu as illustrated in Figure 3-39.

Figure 3-39
Adding a switch matrix



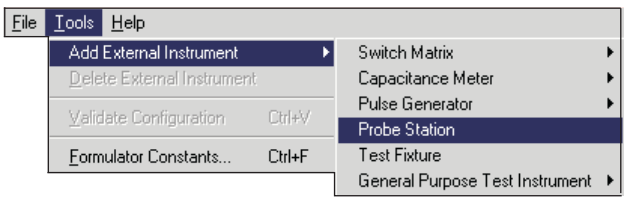
Step 3. Set the GPIB Address for the switch matrix and add the 7174A matrix card in Slot 1 as illustrated in Figure 3-40.

Figure 3-40
Configuring the switch matrix



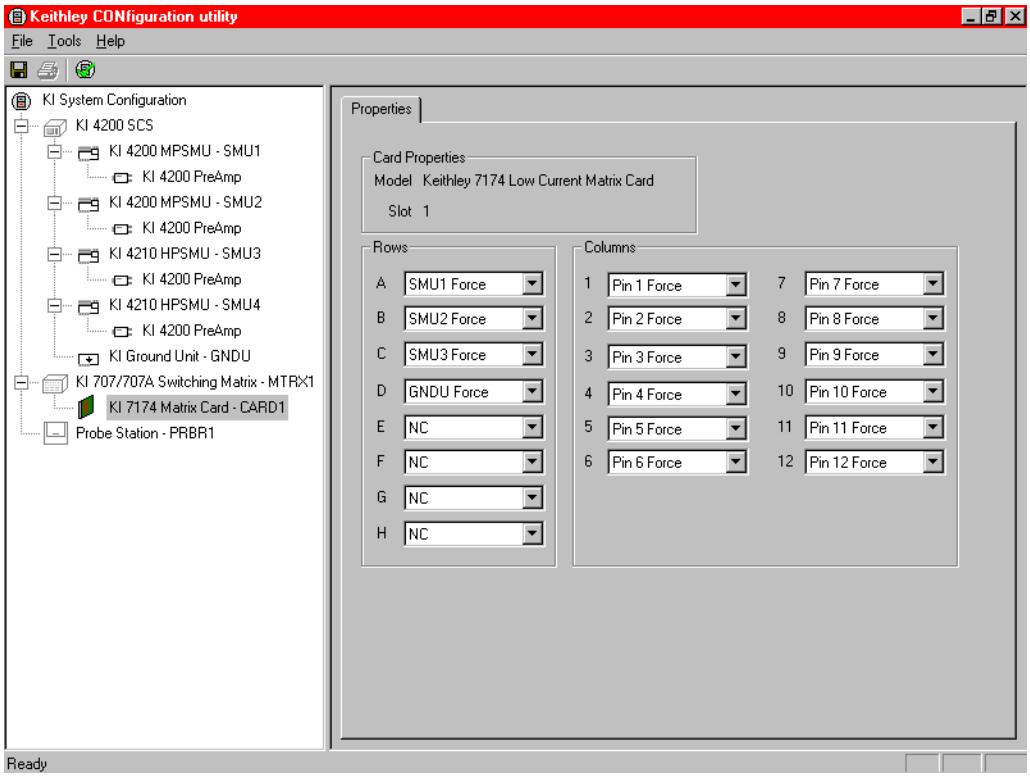
Step 4. Add a manual probe station to the system configuration using the **KCON Tools** menu as illustrated in Figure 3-41. If a test fixture is already part of the configuration, it must removed before the probe station can be added. To remove any external component from the system configuration, select it in the Configuration Navigator and press the **DELETE** key.

Figure 3-41
Adding a probe station



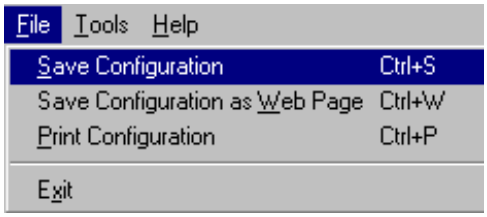
Step 5. Connect the instrument terminals and probe station pins to the switch matrix by selecting the **KI 7174 Matrix Card - CARD1** in the Configuration Navigator and configuring it as illustrated in Figure 3-42. Detailed information regarding switch matrix configuration can be found in the 4200-SCS Reference manual.

Figure 3-42
Connecting the switch matrix



Step 6. **Save** the configuration using the **KCON File** menu as illustrated in Figure 3-43.

Figure 3-43
Saving the system configuration



Probe station configuration

Before *KITE* can begin controlling a probe station, the probe station must be properly configured. Probe station configuration includes:

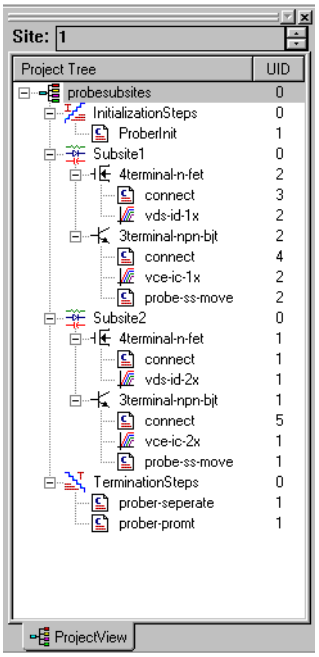
1. Making test system measurement and communication connections.
2. Creating a probe list using the appropriate prober control software.
3. Loading and aligning the wafer.

Helpful configuration instructions for each supported prober are include in the 4200-SCS Reference sections. Refer to Table 3-3 for additional information. Because this tutorial uses a Manual probe station, probe station configuration is simple because step 2 above can be omitted. To configure a manual probe station, simply connect the test system measurement signals to the probe station as indicated in Figure 3-38 and align the prober to the first subsite (test element group) in the test sequence.

Open the “probesubsites” project

Open the “**probesubsites**” project from the **File** menu on the *KITE* toolbar (click **Open Project**). The Project Navigator for the “probesubsites” project is shown in Figure 3-44.

Figure 3-44
Project Navigator - probesubsites project

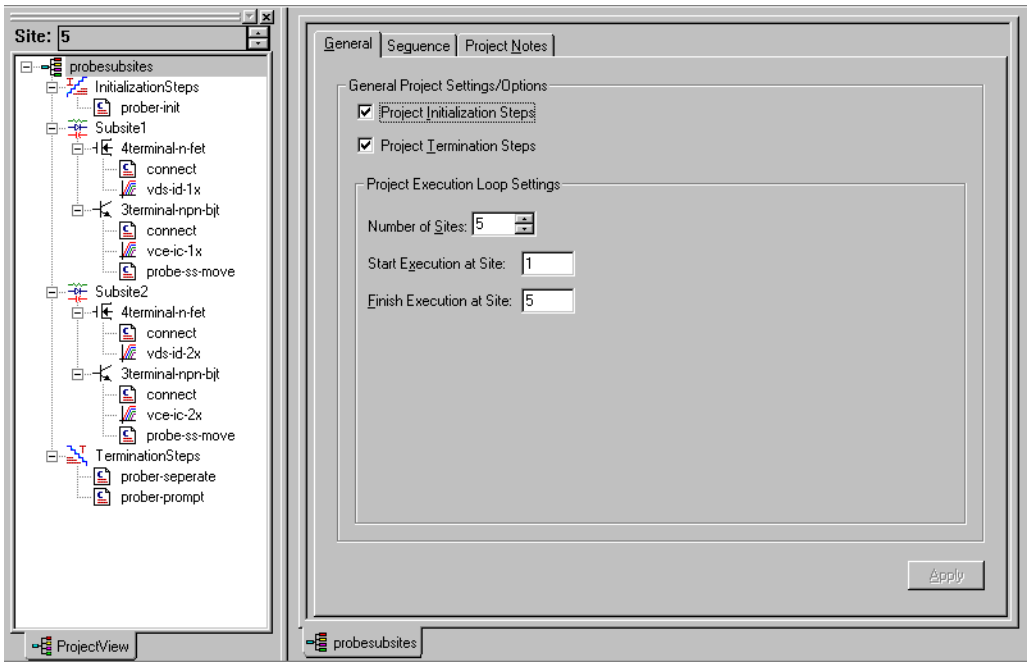


Open the project plan window

In the Project Navigator, double-click **“probesubsites”** to open the project plan window. For this tutorial, five sites on a wafer are to be tested. As shown in Figure 3-45, set up the project plan as follows and click the **Apply** button at the bottom right-hand corner of the window:

- Enable (√) Project Inititialization Steps
- Enable (√) Project Termination Steps
- Start Execution at Site: 1
- Finish Execution at Site: 5

Figure 3-45
Modified project plan settings



Test descriptions

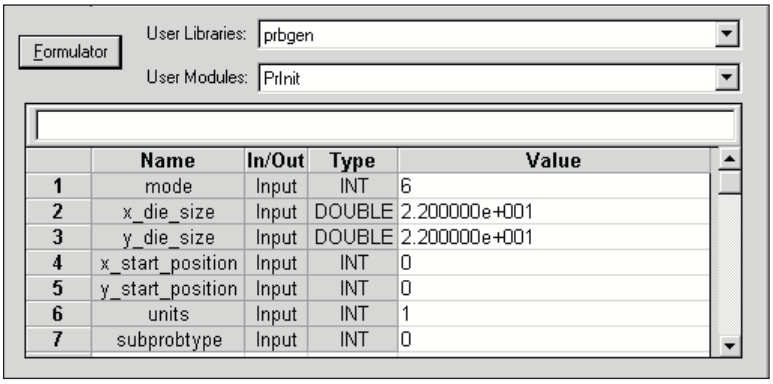
Test descriptions for the “probesubsites” project are provided in Table 3-5. Tests can be opened in the Workspace by double-clicking them in the Project Navigator.

NOTE *The “connect” UTMs are used to control the switch matrix.*

Table 3-5
"probesubsites" test descriptions

"probesubsites" Project	**Test Description**
InitializationSteps prober-init	Initializes the prober driver (see Figure 3-46).
Subsite1 4terminal-n-fet connect vds-id-1x 3terminal-npn-bjt connect vce-ic-1x probe-ss-move Subsite2 4terminal-n-fet connect vds-id-2x 3terminal-npn-bjt connect vce-ic-2x probe-ss-move	 Connects the SMUs to the probes for the N-channel MOSFET (see Figure 3-47). Generates a family of curves (I_D vs. V_D) for the MOSFET. Connects the SMUs to the probes for the NPN transistor (see Figure 3-48). Generates a collector family of curves (I_C vs. V_C) for the transistor. Moves prober to next subsite. Connects the SMUs to the probes for the N-channel MOSFET (see Figure 3-47). Generates a family of curves (I_D vs. V_D) for the MOSFET. Connects the SMUs to the probes for the NPN transistor (see Figure 3-48). Generates a collector family of curves (I_C vs. V_C) for the transistor. Moves prober to the first subsite of the next site.
TerminationSteps prober-separate prober-prompt	The following steps occur after all three sites are tested: Separates the prober pins from the wafer (see Figure 3-49). Displays a *pop-up* window indicating that testing is finished (see Figure 3-50).

Figure 3-46
prober-init



	Name	In/Out	Type	Value
1	mode	Input	INT	6
2	x_die_size	Input	DOUBLE	2.200000e+001
3	y_die_size	Input	DOUBLE	2.200000e+001
4	x_start_position	Input	INT	0
5	y_start_position	Input	INT	0
6	units	Input	INT	1
7	subprobtype	Input	INT	0

Line 1 — Parameter value **6** selects the Learn control mode. Assumes that the probe list is maintained by the prober controller software.

Lines 2 and 3 — These parameters (along with the units setting in Line 6) input a die size of 22mm x 22mm.

Lines 4 and 5 — These parameters input the initial prober position as the **0, 0** coordinates.

Line 6 — Parameter value **1** sets units for die size (lines 2 and 3) to metric.

Figure 3-47
Connect SMUs to N-channel MOSFET

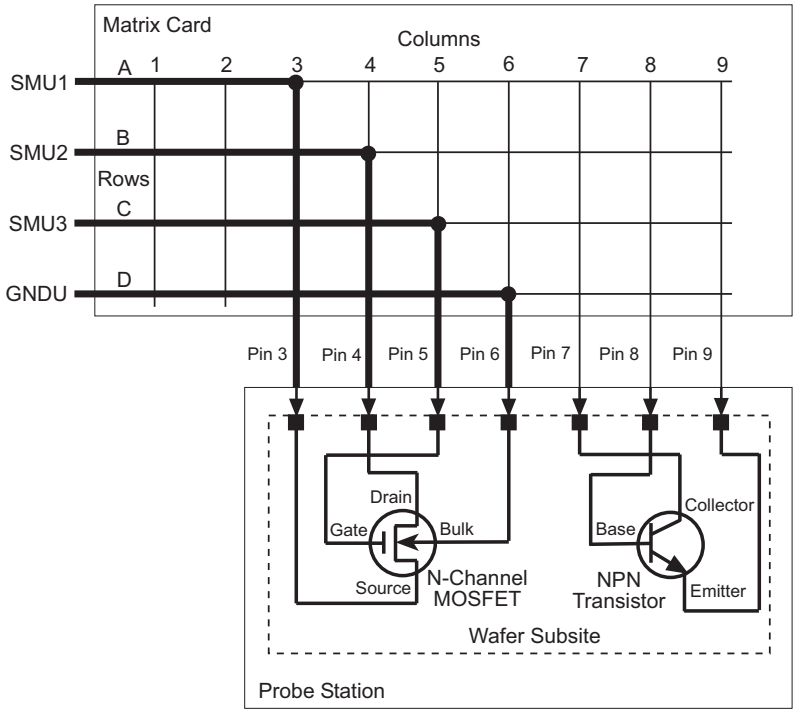


Figure 3-48
Connect SMUs to NPN transistor

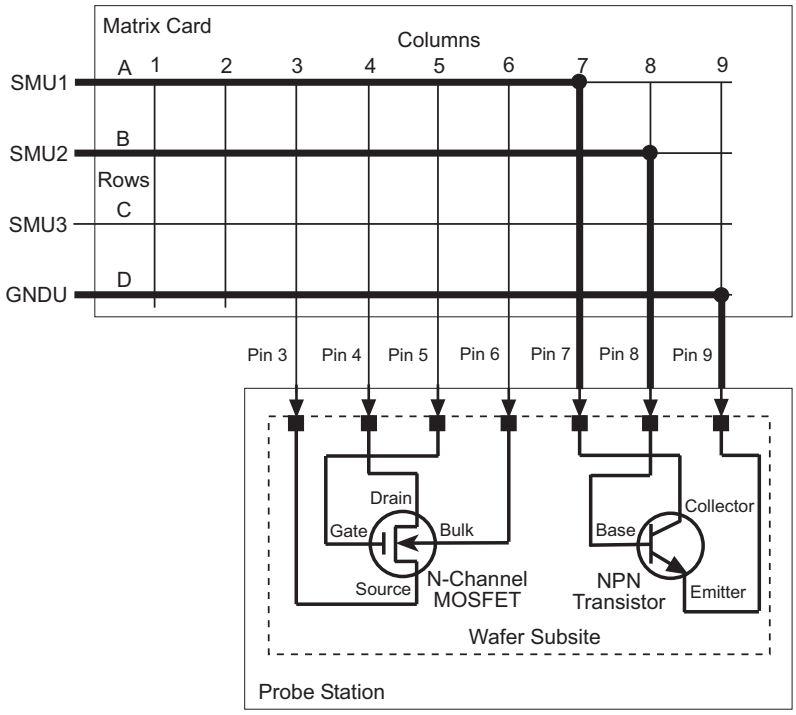
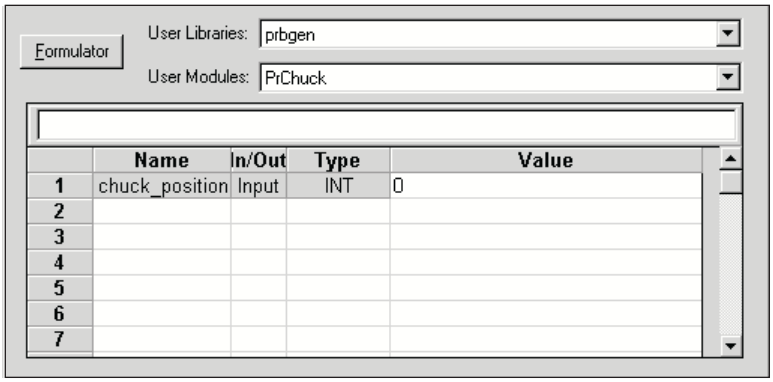
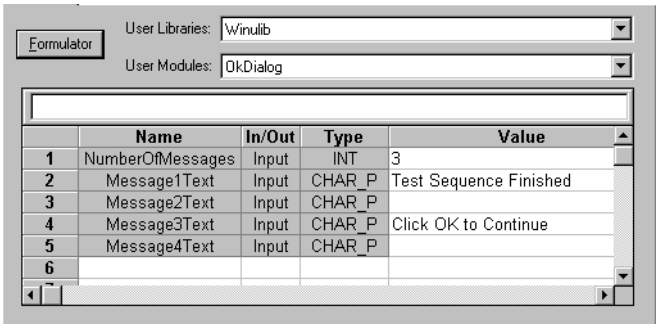


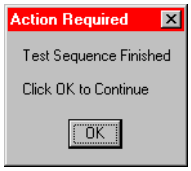
Figure 3-49
prober-separate



Line 1 — Parameter value **0** separates the prober pins from the wafer.

Figure 3-50
prober-prompt test and dialog window





A. Prober-prompt test window

B. Dialog window

Line 1 — Parameter value **3** specifies three lines of text to be displayed.

Lines 2 thru 4 — Text messages to be displayed in dialog window

Running the test sequence

The five wafer sites are tested as follows:

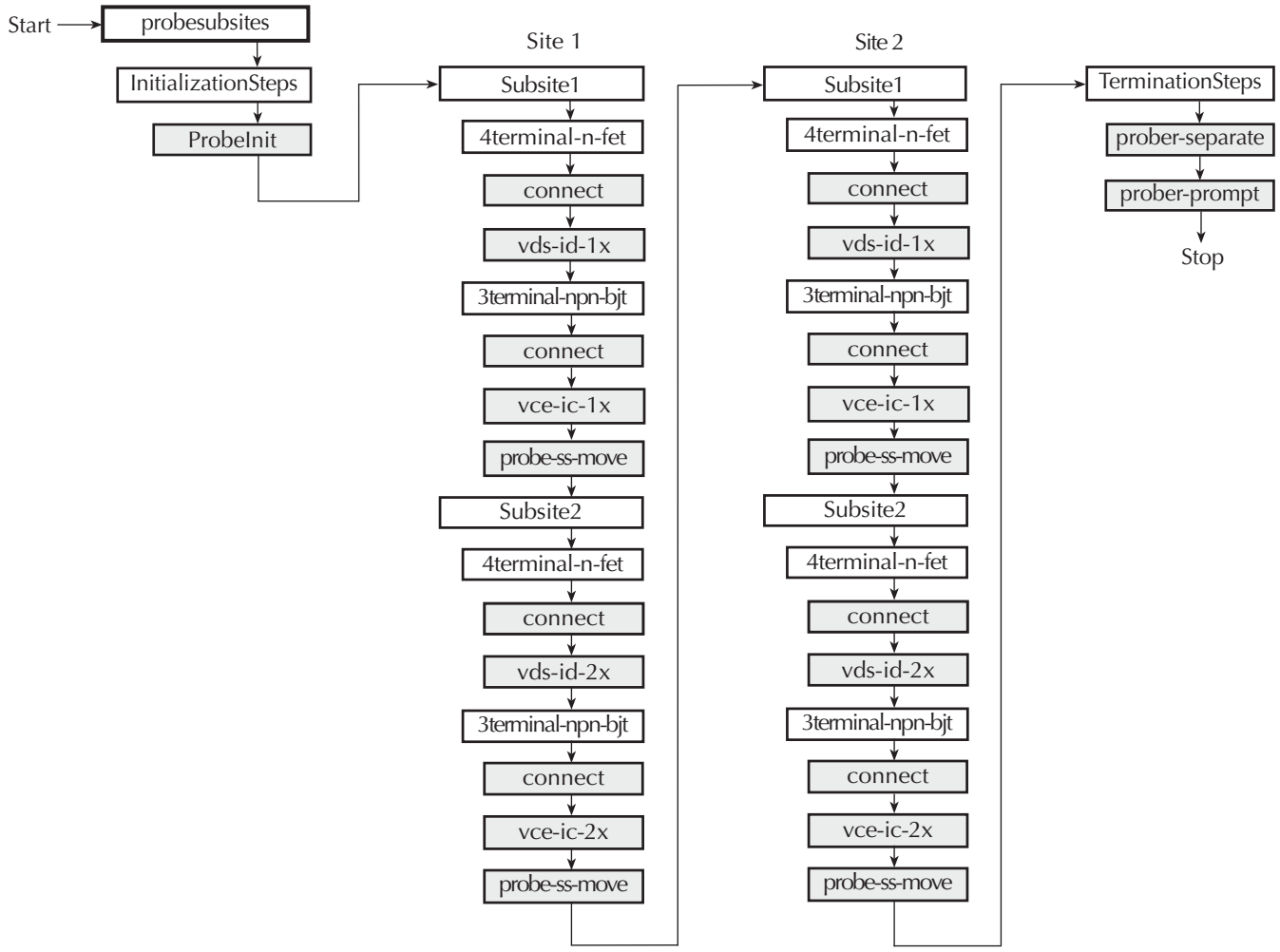
1. Manually align the prober to test Subsite 1 of Site 1. Make sure the prober pins are making contact with the wafer probe pads.
2. In the Project Navigator, click “**probesubsites**” in the *KITE* Project Navigator to select the project.
3. Click the green **Run** button to execute the test sequence.

NOTE *Because a manual probe station is being used, the prober will not actually move when the prober control UTMs are executed. However, a pop-up dialog window will appear instructing you to move the probes to the next subsite in the test sequence.*

The test sequence is shown in Figure 3-51. After the prober is initialized by “prober-init”, the tests for Subsite 1 and Subsite 2 are performed at Site 1. The last test for Site 1 (“probe-ss-move”) moves the prober to Site 2 where the subsite tests are repeated.

After all five sites are tested, the prober pins separate from the wafer (prober-separate), and a dialog window (prober-prompt) will alert that the test sequence is finished (see Figure 3-50B). Click **OK** to continue.

Figure 3-51
Test sequence

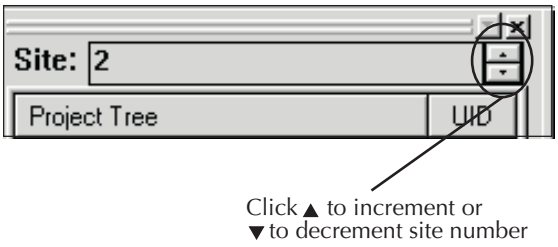


Test data

Since five sites were tested, there will be five sets of test data; one for each site. Remember, a test is opened by double-clicking it in the Project Navigator. Test data is viewed by clicking the **Graph** or **Sheet** tab for the test.

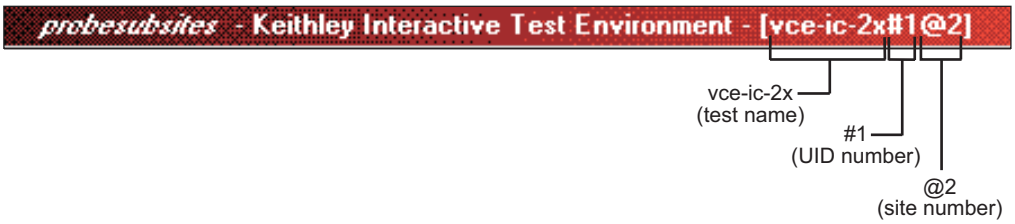
When you double-click a test to open it, its test data corresponds to the site number displayed by the Site Navigator at the top of the Project Navigator. As shown in Figure 3-52, click the up or down arrow to change the site number. For example, to view test data for Site 2, set the Site Navigator to Site 2 and double-click the desired test.

Figure 3-52
Site Navigator



The title bar at the top of the *KITE* panel indicates which test is presently being displayed. In Figure 3-53, test “vce-ic-2x” for Site 2 is being displayed. The unique identifier (UID) number distinguishes this test from any other test having the same name.

Figure 3-53
KITE title bar



Running individual plans or tests

You can run any Subsite Plan, Device Plan, or test in the project. The test sequence will stop after the plan or test is finished. The following steps show how to run the “3terminal-npn-bjt” Device Plan for Subsite 2 of Site 2:

1. Manually position the prober to test Subsite 2 of Site 2. Make sure the prober pins are making contact with the subsite pads.
2. Set the Site Navigator to Site **2**.
3. In the Project Navigator, click **“3terminal-npn-bjt”** for *Subsite2* to select the Device Plan.
4. Click the green **Run** button to start the test sequence.

4 Pulse Applications

Section Topics List

Charge Pumping, page 4-2
- CP Procedure, page 4-2
- Charge pumping UTM descriptions, page 4-6
- amplsweep, page 4-6
- basesweep, page 4-8
- FallTimeLinearSweep, page 4-9
- FreqFactorSweep, page 4-10
- FreqLinearSweep, page 4-12
- RiseTimeLinearSweep, page 4-13

Pulse IV, page 4-15
- Introduction (PIV-A and PIV-Q), page 4-15
- PIV-A test connections, page 4-17
- Using the PulseIV-Complete project for the first time, page 4-24
- Pulse IV UTM descriptions, page 4-36
- cal_pulseiv, page 4-36
- vdsid_pulseiv, page 4-37
- vgsid_pulseiv, page 4-47
- scopeshot_pulseiv, page 4-51
- vdsid_pulseiv_demo, page 4-53
- vgsid_pulseiv_demo, page 4-53
- scopeshot_pulseiv_demo, page 4-53

Slow single pulse charge trapping high K gate stack, page 4-54
- Charge trapping procedure, page 4-55
- Charge Trapping UTM descriptions, page 4-58
- chargetrapping_single_pulse_slow, page 4-58

AC stress for WLR, page 4-61

Q-Point Pulse IV – Model 4200-PIV-Q, page 4-64
- PIV-Q Test Procedure, page 4-65
- Interconnect Assembly Procedure, page 4-66
- Using the Model 4200 Project QPulseIV-Complete for the first time, page 4-70
- Running AutocalScope, page 4-71
- Running CableCompensation, page 4-71
- PIV-Q user libraries, page 4-77

Pulse adapters, cables, hardware and PCU, page 4-118

Flash Memory Testing, page 4-119
Introduction, page 4-119
Theory of operation, page 4-119
FLASH Connections, page 4-132
Flash Projects, page 4-140

There are many possible applications for using pulse source and measure with DC source and measure. This section contains the following:

- Charge pumping for interface characterization for CMOS (requires one channel of the pulse generator card and one Model 4200 SMU)
- Pulse IV to eliminate charge trapping or self-heating effects in new CMOS material and structure technologies (requires full 4200-PIV package)
- Slow Single Pulse Charge Trapping for high K gate stack structures (requires PIV-A package)
- AC stress for WLR (requires pulse generator card and Model 4200 SMUs, and optionally a switch matrix and RBTs)
- The Model 4200-PIV-Q package provides q-point pulse IV testing for higher power compound semiconductor or LDMOS RF transistors, or any device may benefit from low duty cycle pulse IV testing.

NOTE *The various adapters, cables and hardware used for pulsing are shown in Figure 4-50 (located at the end of this section).*

Charge Pumping

Charge Pumping (CP) is a useful technique for understanding gate stack behavior, that is increasingly important as high κ films become more commonly used for transistor gates. CP characterizes interface and charge-trapping phenomena. The change in the CP results can be used to determine the amount of degradation caused by typical reliability test methods, employing either DC or pulsed stress: hot carrier injection (HCI), negative bias temperature instability (NBTI), and time dependent dielectric breakdown (TDDB).

Pulsed voltage provides a key capability for investigating inherent material, interface, and reliability properties of high κ films, and devices based on these new films. Pulsing a voltage while simultaneously measuring the DC current is the basis for charge pumping, that is valuable for measuring inherent charge trapping. Used in conjunction with DC or pulsed stress, CP can also study charge trapping, as well as new charge creation on the high κ-Si interface and within the high κ film. Pulsed stress also provides a stress method that better mimics actual stresses seen by the in-circuit devices, that is useful for various device reliability tests, including NBTI, TDDB, and HCI. In addition, pulsed stress provides insight into device reliability behaviors not available using DC stress. Pulsed stress complements traditional DC techniques to provide a better understanding of device reliability behavior.

This application demonstrates CMOS charge pumping for interface characterization. The schematic in Figure 4-1 shows source and drain of the transistor connected to ground, while the gate is pulsed with fixed frequency and amplitude. The body is connected to ground using a Source-Measure Unit (SMU), that is used to measure the current through the gate (I_{CP}).

NOTE *Although the pulse train must be applied to the DUT before the SMU current measurement begins, there is no strict timing requirement between the pulse applied to DUT and the corresponding SMU current measure. An oscilloscope may be used to monitor the pulse characteristics for initial setup and troubleshooting, but is not used for pulse measurement during the test.*

CP Procedure

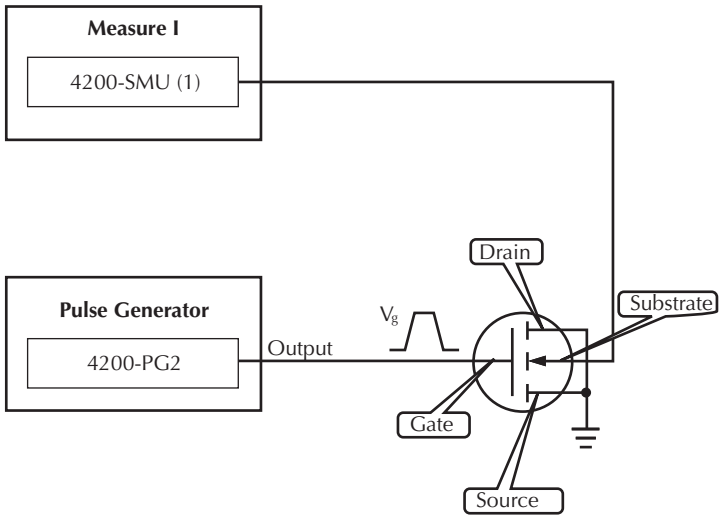
1. Connect DUT (transistor) as shown in Figure 4-2. Make sure source and drain are connected to ground.
2. Choose desired test method (Figure 4-3). For a brief overview of the test methods contained in this manual, refer to . To go directly to a specific charge pumping UTM, see the list of test methods and page numbers below:

NOTE *Each CP test method is in a separate UTM (User Test Module).*

- Amplitude sweep — see
- Base voltage sweep — see
- Fall time sweep — see
- Frequency factor sweep — see
- Frequency linear sweep — see
- Rise time linear sweep — see

3. Choose voltage (V) steps and current (I) measurement parameters.
4. The UTM then pulses gate with a train of pulses at a fixed base/amplitude.
5. While pulsing, the UTM will measure DC substrate current with SMU.
6. Once one measurement is finished, change pulse characteristics and re-measure again. Pulse parameters are changed based on type of voltage sweep. See upper half of Figure 4-3 for type of sweeps. Both plots in Figure 4-3 (base voltage sweep and amplitude sweep) show ICP (measured charge pumping current) vs. pulse voltage.

Figure 4-1
Charge Pumping—hardware setup block diagram



NOTE *The pulse width is not explicitly set, but is derived from the frequency and the duty cycle.*

Table 4-1
Key pulse generator parameters

Parameters	Range/Specification
Variable rise time and fall time	100ns to 500us
Duty cycle	0.01% to 99%, default 50%
Frequency	100 Hz to 12.5 MHz
Pulse amplitude	-5 to +5V
[1]Base + Amplitude must not exceed -5V or +5V absolute.	

Table 4-1 (cont.)
Key pulse generator parameters

Parameters	Range/Specification
Base voltage	+/- 5V[1]
Load impedance	50Ω or 1MΩ.

[1]Base + Amplitude must not exceed -5V or +5V absolute.

Figure 4-2
Charge pumping — hardware connection

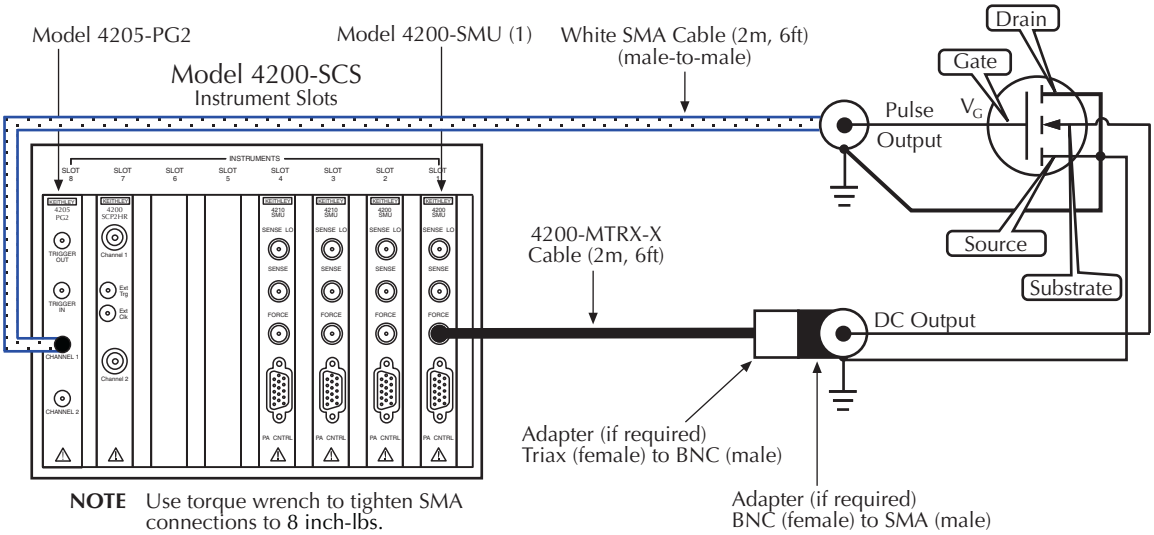
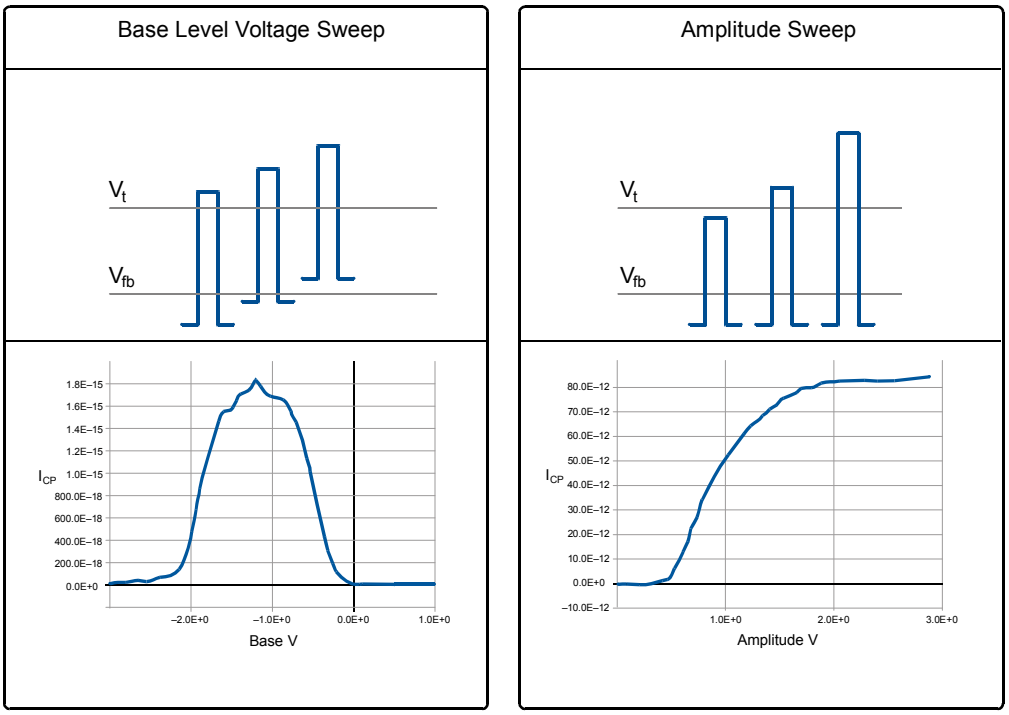


Figure 4-3
Two types of sweeps for charge pumping



7. Outputs curves:

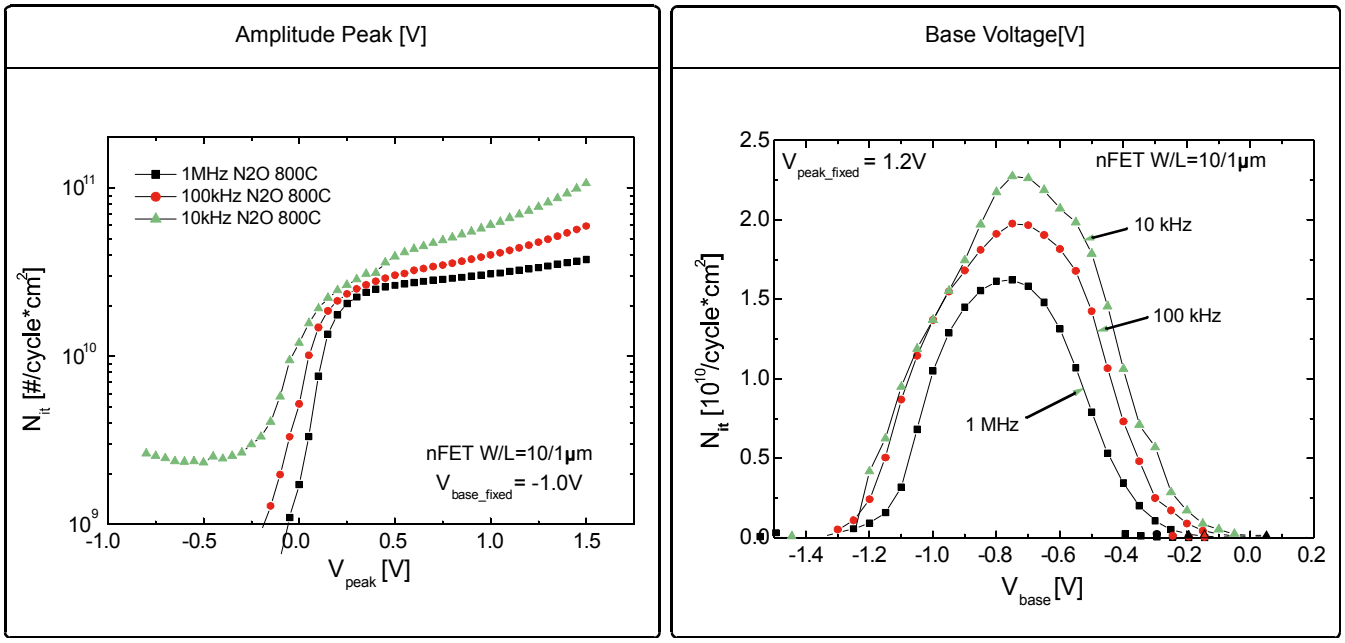
a. Plot I_{CP} vs swept parameter (e.g., amplitude, base, frequency, rise time).
b. An example of the calculated parameters as shown in Figure 4-4:

   Interface Trap Charge $N_{it} = \frac{I_{CP}}{qfA}$

   where:
   Icp = charge pumping current measured by the SMU
   q = elemental charge of an electron
   f = frequency of the pulses
   A = area of the capacitor
c. Another useful calculated parameter:

   Interface Trap Density $D_{it} = \frac{I_{CP}}{qAf\Delta E}$
   where:
   Icp = charge pumping current measured by the SMU
   q = elemental charge of an electron
   A = area of the capacitor
   f = frequency of the pulses
   ΔE = difference between the inversion Fermi level and the accumulation Fermi level

Figure 4-4
Example data plots for N_{it}



Charge pumping UTM descriptions

The chargepumping user library contains modules required to characterize interface and charge-trapping phenomena. The modules contained in the charge pumping user library are listed in Table 4-2 with detailed information following the table.

Table 4-2
Charge pumping UTMs

User Module	Description
	Performs and graphs a linear sweep of the pulse amplitude.
	Performs and graphs a linear sweep of the pulse base or offset.
	Performs and graphs a linear sweep of the falling transition time of the pulse.
	Performs and graphs a log or multiply frequency sweep of the pulse.
	Performs and graphs a linear sweep of the frequency of the pulse.
	Performs and graphs a linear sweep of the rising transition time of the pulse,

amplsweep

Description The amplsweep is a charge pumping routine that performs a linear sweep of the pulse amplitude, graphing the resulting charge pumping current measured by a 4200 SMU. This routine controls a single channel of the pulse generator card as well as a 4200 SMU.
Make sure to set the appropriate values for the charge pumping parameters (Table 4-3). Table 4-4 contains the routines output parameters. The rise time and fall time parameters are the full transition times (0–100%), not the 10%–90% times. For the 5V range of the pulse generator card, the 10–90% rise times are about 20% less than the full 0–100% transition times that are used to program the pulse.

Connection This procedure requires connection of the appropriate pulse channel to the gate of the DUT (device under test) and the substrate/well to the 4200 SMU Force. The other DUT pins should be connected to ground. For detailed connection information, refer to the .

Table 4-3
Inputs for amplsweep

Input	Type	Description
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with a single pulse generator card.
PulseChan	int	The pulse generator card output channel, 1 or 2.
SubSMU	char *	The SMU for the substrate/well. This can be SMU1 up to the maximum number of SMUs in the system.
StartVampl	double	Starting pulse amplitude (V). This can be set from -80V to +80V.
StopVampl	double	Stopping amplitude voltage for the sweep (V). This can be set from -80V to +80V.
StepVampl	double	Step size for the amplitude sweep (V). This can be set from -80V to +80V.
PulseOffset	double	Offset, or base, of the pulse (V). This can be set from -40V to +40V (inclusive of amplitude).
PulseRiseTime	double	Transition rise time for the pulse. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseFallTime	double	Transition fall time for the pulse. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseFrequency	double	Pulse frequency. This can be set from 1Hz to 20Mhz.
DutyCyclePercent	double	Duty Cycle percent. This can be set from 0.001% to 99.9%.
PulseLoad	double	DUT load or impedance (ohm). This can be set from 50 to 1E6. This value is used to adjust the pulse amplitude sourced by the pulse generator card to compensate for non-50 ohm termination. For example, setting the load = 1E6 means the pulse generator card will output half the voltage compared to load = 50.
PulseRange	double	Selects pulse range. Set this value to 5 for high speed or to 20 for high voltage
NPLC	int	Integration time in power line cycles. This can be set from 0.01 to 10.
SMUCompliance	double	Current limit for the SMU. Set from 10e-12 to 100e-3.
PulseAmpl_size Icp_size Qcp_size	double	Set to a value that is at least equal to the number of steps in the sweep. All _size values must be equal to each other.
LowestIRange	double	Lowest current measure range used during limited auto range. This can be set from 10e-12 to 100e-3

Table 4-4
Outputs for amplsweep

Output	Type	Description
PulseAmpl	double *	The array of pulse amplitudes used.
Icp	double *	The array of current values measured by the SMU.
Qcp	double *	The array charge values calculated from the Icp values, where: Qcp = Icp/(Frequency)

basesweep

Description The basesweep is a charge pumping routine to perform a linear sweep of the pulse base or offset, graphing the resulting charge pumping current measured by a 4200 SMU. This routine controls a single channel of the pulse generator card as well as a 4200 SMU.
Make sure to set the appropriate values for the charge pumping parameters (Table 4-5). Table 4-6 contains the routines output parameters. The rise time and fall time parameters are the full transition times (0–100%), not the 10%–90% times. For the 5V range of the pulse generator card the 10–90% rise times are about 20% less than the full 0–100% transition times that are used to program the pulse.

Connection Using this routine requires connection of the appropriate pulse channel to the gate of the DUT (device under test) and the substrate or well to the 4200 SMU Force. The other DUT pins should be connected to ground. For detailed connection information, refer to

Table 4-5
Inputs for basesweep

Input	Type	Description
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with a single pulse generator card.
PulseChan	int	The pulse generator card output channel, 1 or 2.
SubSMU	char *	The SMU for the substrate/well. This can be SMU1 up to the maximum number of SMUs in the system.
StartVBase	double	Starting pulse base (V). This can be set from -40 to +40V.
StepVBase	double	Step size for the base sweep (V). This can be set from -40 to +40V.
StopVBase	double	Stopping amplitude voltage for the sweep (V). This can be set from -40 to +40V.
PulseAmplitude	double	This sets pulse amplitude (V) from -80 to +80V (inclusive of base).
PulseRiseTime	double	Transition rise time for the pulse. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseFallTime	double	Transition fall time for the pulse. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseFrequency	double	Pulse frequency. This can be set from 1Hz to 20Mhz.
DutyCyclePercent	double	Duty Cycle percent. This can be set from 0.001% to 99.9%.
PulseLoad	double	DUT load or impedance (ohm). This can be set from 50 to 1E-6. This value is used to adjust the pulse amplitude sourced by the pulse generator card to compensate for non-50 ohm termination. For example, setting the load = 1E6 means the pulse generator card will output half the voltage compared to load = 50.
PulseRange	double	Selects pulse range. Set this value to 5 for high speed or to 20 for high voltage.
NPLC	double	Integration time in power line cycles. This can be set from 0.01 to 10.
SMUCompliance	double	Current limit for the SMU. Set from 10e-12 to 100e-3.
BaseV_size Icp_size Qcp_size	int	Set to a value that is at least equal to the number of steps in the sweep. All _size values must be equal to each other.
LowestIRange	double	Lowest current measure range used during limited auto range. This can be set from 10e-12 to 100e-3.

Table 4-6
Outputs for basesweep

Output	Type	Description
BaseV	double *	The array of pulse base voltages
Icp	double *	The array of current values measured by the SMU.
Qcp	double *	The array charge values calculated from the Icp values, where: Qcp = Icp/(Frequency)

FallTimeLinearSweep

Description The FallTimeLinearSweep is a charge pumping routine that performs a linear sweep of the falling transition time of the pulse, graphing the resulting charge pumping current measured by a 4200 SMU. This routine controls a single channel of the pulse generator card as well as a 4200 SMU.
Make sure to set the appropriate values for the charge pumping parameters (Table 4-7). Table 4-8 contains the routines output parameters. The rise time and fall time parameters are the full transition times (0–100%), not the 10%–90% times. For the 5V range of the pulse generator card the 10–90% rise times are about 20% less than the full 0–100% transition times that are used to program the pulse.

Connection Using this routine requires connection of the appropriate pulse channel to the gate of the device under test (DUT) and the substrate or well to the 4200 SMU Force. The other DUT pins should be connected to ground. For detailed connection information, refer to the .

Table 4-7
Inputs for FallTimeLinearSweep

Input	Type	Description
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with a single pulse generator card.
PulseChan	int	The pulse generator card channel, 1 or 2.
SubSMU	char	The SMU number. This can be SMU1 to the maximum number of SMUs in the system.
StartFallTime	double	Starting transition fall time for sweep (s). This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
StopFallTime	double	Stopping transition fall time for sweep (s). This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
StepFallTime	double	Stepsize for transition fall time sweep (s). This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseRiseTime	double	Transition rise time for sweep. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseAmplitude	double	Amplitude of pulse (V). This can be set from -40V to +40V (inclusive of offset).
PulseFrequency	double	Pulse frequency. This can be set from 1 Hz to 20 Mhz.
PulseOffset	double	Offset, or base, of the pulse (V). This can be set from -40V to +40V (inclusive of amplitude).
DutyCyclePercent	double	Duty Cycle percent. This can be set from 0.001% to 99.9%.

Table 4-7 (cont.)
Inputs for FallTimeLinearSweep

Input	Type	Description
PulseLoad	double	DUT load or impedance (ohm). This can be set from 50 to 1E6. This value is used to adjust the pulse amplitude sourced by the pulse generator card to compensate for non-50 ohm termination. For example, setting the load = 1E6 means the pulse generator card will output half the voltage compared to load = 50.
PulseRange	double	Selects pulse range. Set this value to 5 for high speed or to 20 for high voltage
NPLC	int	Integration time in power line cycles. This can be set from 0.01 to 10.
SMUCompliance	double	Current limit for the SMU. Set from 10e-12 to 100e-3.
FallTimeSize Icp_size Qcp_size	int	Set to a value that is at least equal to the number of steps in the sweep. All _size values must be equal to each other.
LowestIRange	double	Lowest current measure range used during limited auto range. This can be set from 10e-12 to 100e-3

Table 4-8
Outputs for FallTimeLinearSweep

Output	Type	Description
FallTransTime	double *	The array of fall transition times used.
Icp	double *	The array of current values measured by the SMU.
Qcp	double *	The array charge values calculated from the Icp values, where: Qcp = Icp/(Frequency)

FreqFactorSweep

Description The FreqFactorSweep is a charge pumping routine that performs a log or multiply frequency sweep of the pulse, graphing the resulting charge pumping current measured by a 4200 SMU. This routine controls a single channel of the pulse generator card as well as a 4200 SMU.
Make sure to set the appropriate values for the charge pumping parameters (Table 4-9). Table 4-10 contains the routines output parameters. The rise time and fall time parameters are the full transition times (0–100%), not the 10%-90% times. For the 5 V range of the pulse generator card the 10–90% rise times are about 20% less than the full 0–100% transition times that are used to program the pulse.

Connection Using this routine requires connection of the appropriate pulse channel to the gate of the device under test (DUT) and the substrate or well to the 4200 SMU Force. The other DUT pins should be connected to ground. For detailed connection information, refer to .

Table 4-9
Inputs for FreqFactorSweep

Input	Type	Description
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with a single pulse generator card.
PulseChan	int	The pulse generator card channel, 1 or 2.
SubSMU	char	The SMU number. This can be SMU1 to the maximum number of SMUs in the system.
FreqMultFactor	double	Multiplier factor to control step size. Next Freq = Previous frequency * FreqMultFactor. Use Factors > 1 for sweeping to higher frequencies. Use Factors < 1 for sweeping to lower frequencies.
NumPoints	int	Number of points in the frequency sweep.
PulseRiseTime	double	Transition rise time for sweep. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseFallTime	double	Transition fall time for the pulse. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseAmplitude	double	Amplitude of pulse (V). This can be set from -80 to +80V (inclusive of offset).
PulseFrequency	double	Pulse frequency. This can be set from 1Hz to 20Mhz.
PulseOffset	double	Offset, or base, of the pulse (V). This can be set from -40V to +40V (inclusive of amplitude).
DutyCyclePercent	double	Duty Cycle percent. This can be set from 0.001% to 99.9%.
PulseLoad	double	DUT load or impedance (ohm). This can be set from 50 to 1E6. This value is used to adjust the pulse amplitude sourced by the pulse generator card to compensate for non-50 ohm termination. For example, setting the load = 1E6 means the pulse generator card will output half the voltage compared to load = 50.
PulseRange	double	Selects pulse range. Set this value to 5 for high speed or to 20 for high voltage
NPLC	int	Integration time in power line cycles. This can be set from 1 to 10.
SMUCompliance	double	Current limit for the SMU. Set from 10e-12 to 100e-3.
Frequency_size Qcp_size Icp_size	(int)	Set to a value that is at least equal to the number of steps in the sweep. All _size values must be equal to each other.
LowestIRange	double	Lowest current measure range used during limited auto range. This can be set from 10e-12 to 100e-3

Table 4-10
Outputs for FreqFactorSweep

Output	Type	Description
Frequency	double *	The array of frequencies used.
Icp	double *	The array of current values measured by the SMU.
Qcp	double *	The array charge values calculated from the Icp values, where: Qcp = Icp/(Frequency)

FreqLinearSweep

Description The FreqLinearSweep is a Charge Pumping routine to perform a linear sweep of the pulse frequency, graphing the resulting charge pumping current measured by a 4200 SMU. This routine controls a single channel of the pulse generator card as well as a 4200 SMU.
Make sure to set the appropriate values for the charge pumping parameters (Table 4-11). Table 4-12 contains the routines output parameters. Note that the rise time and fall time parameters are the full transition times (0–100%), not the 10%–90% times. For the 5 V range of the pulse generator card the 10–90% rise times are about 20% less than the full 0–100% transition times that are used to program the pulse.

Connection Using this routine requires connection of the appropriate pulse channel to the gate of the device under test (DUT) and the substrate or well to the 4200 SMU Force. The other DUT pins should be connected to ground. For detailed connection information, refer to the .

Table 4-11
Inputs for FreqLinearSweep

Input	Type	Description
InstIdStr	char *	The instrument ID. This should be set to VPU1 for 4200 systems with a single pulse generator card.
PulseChan	int	The pulse generator card output channel, 1 or 2.
SubSMU	char	The SMU number. This can be SMU1 to the maximum number of SMUs in the system.
StartFreq	double	Starting pulse frequency for sweep (Hz). This can be set from 1 to 20 MHz (20E6).
StopFreq	double	Stopping pulse frequency for sweep (Hz). This can be set from 1 to 20 MHz (20E6).
StepFreq	double	Pulse frequency step size for sweep (Hz). This can be set from 1 to 20 MHz (20E6).
PulseRiseTime	double	Transition rise time for sweep. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseFallTime	double	Transition fall time for sweep. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseAmplitude	double	Amplitude of pulse (V). This can be set from -80V to +80V (inclusive of offset).
Frequency	double	Pulse frequency. This can be set from 1Hz to 10Mhz.
PulseOffset	double	Offset, or base, of the pulse (V). This can be set from -40V to +40V (inclusive of amplitude).
DutyCyclePercent	double	Duty Cycle percent. This can be set from 0.001% to 99.9%.
PulseLoad	double	DUT load or impedance (ohm). This can be set from 50 to 1E6. This value is used to adjust the pulse amplitude sourced by the pulse generator card to compensate for non-50 ohm termination. For example, setting the load = 1E6 means the pulse generator card will output half the voltage compared to load = 50.
PulseRange	double	Selects pulse range. Set this value to 5 for high speed or to 20 for high voltage
NPLC	int	Integration time in power line cycles. This can be set from 0.01 to 10.
SMUCompliance	double	Current limit for the SMU. Set from 10e-12 to 100e-3.

Table 4-11 (cont.)
Inputs for FreqLinearSweep

Input	Type	Description
Frequency_size Qcp_size Icp_size	int	Set to a value that is at least equal to the number of steps in the sweep. All _size values must be equal to each other.
LowestIRange	double	Lowest current measure range used during limited auto range. This can be set from 10e-12 to 100e-3

Table 4-12
Outputs for FreqFactorSweep

Output	Type	Description
Frequency	double *	The array of frequencies used.
Icp	double *	The array of current values measured by the SMU.
Qcp	double *	The array charge values calculated from the Icp values, where: Qcp = Icp/(Frequency)

RiseTimeLinearSweep

Description The RiseTimeLinearSweep is a charge pumping routine to perform a linear sweep of the rising transition time of the pulse, graphing the resulting charge pumping current measured by a 4200 SMU. This routine controls a single channel of the pulse generator card as well as a 4200 SMU.
Make sure to set the appropriate value for the charge pumping parameters (Table 4-13). Table 4-14 contains the routines output parameters. The rise time and fall time parameters are the full transition times (0–100%), not the 10%–90% times. For the 5V range of the pulse generator card the 10–90% rise times are about 20% less than the full 0–100% transition times that are used to program the pulse.

Connection Using this routine requires connection of the appropriate pulse channel to the gate of the device under test (DUT) and the substrate or well to the 4200 SMU Force. The other DUT pins should be connected to ground. For detailed connection information, refer to the .

Table 4-13
Inputs for RiseTimeLinearSweep

Input	Type	Description
InstIdStr	char *	The instrument ID. This should be set to VPU1 for 4200 systems with a single pulse generator card.
PulseChan	int	The pulse generator card channel, 1 or 2.
SubSMU	char	The SMU number. This can be SMU1 to the maximum number of SMUs in the system.
StartRiseTime	double	Starting transition rise time for sweep (s). This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
StopRiseTime	double	Stopping transition rise time for sweep (s). This can be set from 10E-9 (10 ns) to 1 second with 10 ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
StepRiseTime	double	Stepsize for transition rise time sweep (s). This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseFallTime	double	Transition fall time for sweep. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this value programs the full transition time (0–100%) not 10–90%.
PulseAmplitude	double	Amplitude of pulse (V). This can be set from -80V to +80V (inclusive of offset).
PulseFrequency	double	Pulse frequency. This can be set from 1 Hz to 20Mhz.
PulseOffset	double	Offset, or base, of the pulse (V). This can be set from -40V to +40V (inclusive of amplitude).
DutyCyclePercent	double	Duty Cycle percent. This can be set from 0.001% to 99.9%.
PulseLoad	double	DUT load or impedance (ohm). This can be set from 50 to 1E6. This value is used to adjust the pulse amplitude sourced by the pulse generator card to compensate for non-50 ohm termination. For example, setting the load = 1E6 means the pulse generator card will output half the voltage compared to load = 50.
PulseRange	double	Selects pulse range. Set this value to 5 for high speed or to 20 for high voltage
NPLC	int	Integration time in power line cycles. This can be set from 0.01 to 10.
SMUCompliance	double	Current limit for the SMU. Set from 10e-12 to 100e-3.
RiseTime_size Qcp_size Icp_size	int	Set to a value that is at least equal to the number of steps in the sweep. All _size values must be equal to each other.
LowestIRange	double	Lowest current measure range used during limited auto range. This can be set from 10e-12 to 100e-3

Table 4-14
Outputs for FreqFactorSweep

Output	Type	Description
Frequency	double *	The array of fall transition times used.
Icp	double *	The array of current values measured by the SMU.
Qcp	double *	The array charge values calculated from the Icp values, where: Qcp = Icp/(Frequency)

Pulse IV

Introduction (PIV-A and PIV-Q)

Pulse IV is used in addition to DC IV test results to address two DUT behaviors: self heating (also called joule heating) and transient charging. For RF Transistors, especially those implemented with compound semiconductor materials, these two effects are typically called dispersion. The self heating and/or charging effects cause the DC and Pulse IV responses to differ.

Pulse IV addresses self heating by permitting the use of a low duty cycle, typically <0.1%, pulses to virtually eliminate heating within the DUT. Pulse IV addresses the charging effects by using pulse widths short enough so that charges cannot be sufficiently mobile within the pulse.

NOTE *The UTMs used for Pulse IV tests are described in the following paragraphs. These UTMs control all instrumentation for these applications. The pulse generator and scope cards can also be used as stand-alone instruments.*

Section 11 of the Reference Manual explains front panel operation and provides remote programming information for the pulse generator and scope. For remote programming, the pulse generator card uses LPTLib functions, while the scope card uses kiscopeulib UTMs.

Section 12 of the Reference Manual provides additional information on projects for the PIV-A and PIV-Q packages.

What is Pulse IV?

Pulse IV provides a user with the capability of running parametric curves on devices using pulsed rather than DC signals. A pulse source with a corresponding pulse measurement can be used in two general ways.

The first method is to provide DC-like parametric tests, where the measurement happens during the flat, settled part of the pulse. Typical tests are IV sweeps, such as a Vds-Id family of curves or a Vgs-Id curve used for Vt extraction.

The second method is transient testing, where a single pulse waveform is used to investigate time varying parameter(s). An example of this second case would be using a single pulse waveform to investigate the Id degradation versus time due to charge trapping or self-heating.

Why use Pulse IV?

Both methods of Pulse IV (PIV) testing listed above are typically used to overcome or study the effects of self heating (joule heating) and for time-domain studies, such as transient charge trapping in the device under test (DUT). As a general trend, pulse and pulse IV testing is increasingly important in semiconductor research, device and process development.

This section will focus on the DC-like IV sweep capability of the PIV-A package, although other types of pulse testing are possible, such as charge pumping, single pulse charge trapping, AC stress, and non-volatile memory testing. Because charge pumping and floating gate memory testing typically use a pulse source with DC measure, these methods are not using pulse IV (pulse source with pulse measure) capabilities.

What PulseIV Packages are available for the Model 4200-SCS?

PIV-A Package – The Model 4200-PIV-A package provides pulse IV self heating for CMOS SOI for $\leq$ 45nm technology node or any device that may benefit from low duty cycle pulsed IV testing to reduce the amount of power provided to the DUT during the test. The PIV-A package utilizes bias tees to permit both DC and pulse IV tests without re-cabling and pulses the DUT gate while DC biasing the DUT drain. See “Pulse IV for CMOS: Model 4200-PIV-A” for details on using the PIV-A package.

PIV-Q Package – The Model 4200-PIV-Q package provides higher power pulsing than the PIV-A package, while also permitting voltage pulsing from a non-zero bias, or quiescent point. The PIV-Q package provides voltage pulses to both the DUT gate and drain simultaneously. The PIV-Q package is appropriate for pulse IV testing of LDMOS and compound semiconductor FETs (HEMT, pHEMT) and other devices that require two channels of voltage pulsing, such as some HBTs. The PIV-Q package also provides DC tests without re-cabling. See "Q-Point Pulse IV – Model 4200-PIV-Q" for details on using the PIV-Q package.

Pulse IV for CMOS: Model 4200-PIV-A

What is the PIV-A PulseIV Package?

The PIV-A package is an optional factory-installed kit to the Model 4200-SCS. The focus for the PIV-A package is testing lower power CMOS transistors that exhibit self-heating or charge trapping effects.Self-heating has been an issue for some higher power devices, but is emerging as a problem for lower power devices based on smaller dimensions and silicon-on-insulator (SOI) technology, where it is more difficult for the heat generated by the transistor to leave its immediate surroundings.

In addition to smaller dimensions, high k materials are being considered to greatly lower gate leakage current for future transistor technology. Unfortunately, these high k materials and related integration processes are not yet perfected and have both interface and bulk lattice imperfections that can cause charges to be trapped.

Both the charge trapping and self-heating effects can be largely avoided by using pulse IV instead of DC parametric testing.

To accomplish pulse IV testing of CMOS transistors, the PIV-A package consists of the following:

- Model 4205-PG2 Dual channel voltage pulse generator
- Model 4200-SCP2 Dual channel oscilloscope
- Pulse IV Interconnect Model 4205-RBT Remote Bias Tees to combine both DC and pulse signals
- Pulse IV software – Projects and test routines for testing of CMOS transistors, including cable compensation and load-line algorithms to provide DC-like sweep results

The configuration and test concepts of the PIV-A package are based on concepts and work by K.A. Jenkins[1] and A. Kerber[2], with results by C.D. Young[3] and others (see).

Target applications and test projects for PIV-Q

The PIV-A package includes test projects that address the most common parametric transistor tests: Vds-id and Vgs-id. These tests are provided in both DC and Pulse modes, allowing correlation between the two test methods, and have been configured for testing leading edge, lower-power CMOS devices. These tests, as well as initialization steps for scope auto-calibration and cable compensation, are included in a single Model 4200 test project, Pulse-IV-Complete.

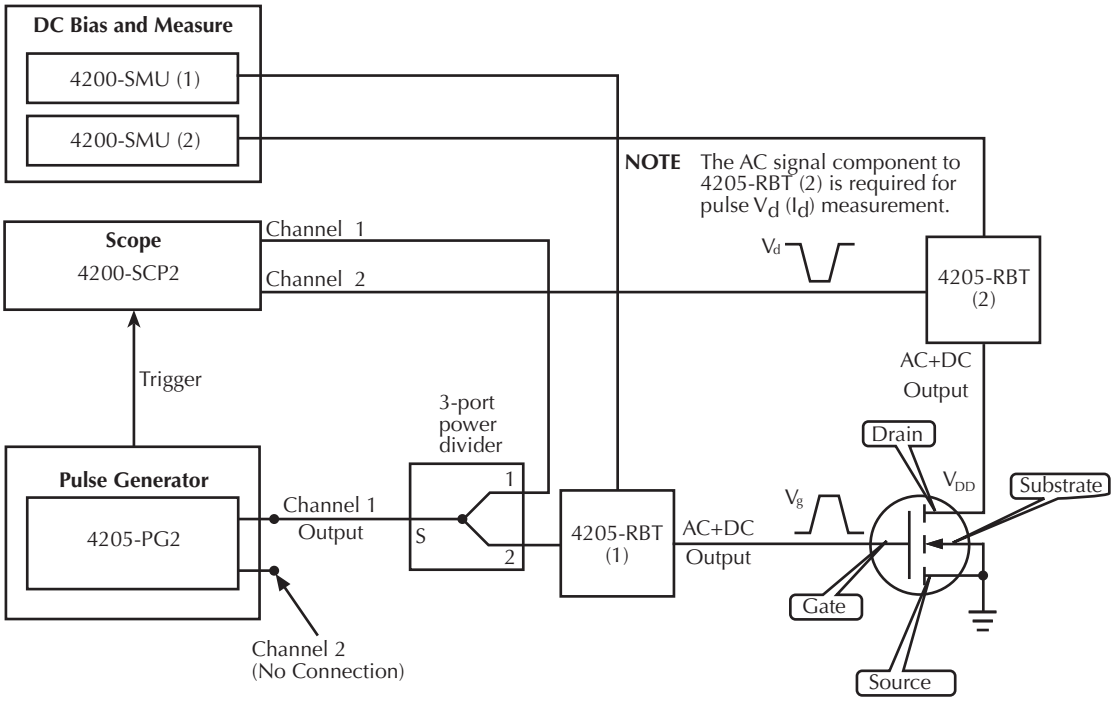
There is another Pulse IV test project, Demo-PulseIV. This Demo project is a subset of PulseIVComplete and is intended for demonstrating the Pulse IV capabilities using a packaged demonstration DUT.

NOTE *The UTMs used for Pulse IV tests are described in the following paragraphs. These UTMs control all instrumentation for these applications. The pulse generator and scope cards can also be used as stand-alone instruments. Section 11 of the Reference Manual explains front panel operation and provides remote programming information for the pulse generator and scope. For remote programming, the pulse generator card uses LPTLib functions, while the scope card uses kiscopeulib UTMs.*

PIV-A test connections

The block diagram for PIV-A testing is shown in Figure 4-5, and the hardware connections are shown in Figure 4-6. A side view of the scope card is provided in Figure 4-7 to show the adapters.

Figure 4-5
Pulse IV—hardware setup block diagram



Supplied interconnect parts

The interconnect parts listed in Table 4-15 are supplied with the PIV-A package.

Table 4-15
Supplied interconnect parts for PIVA

Qty	Description	Comment
1	4.25in/10.8cm white SMA cable	Interconnect for trigger
2	6in/15cm white SMA cables	Interconnect between RBT and prober manipulator
2	13 in/33 cm white SMA cables	Interconnect between RBT and prober manipulator (optional)
3	6.6ft/2m white SMA cables	Interconnect between SCP2, pulser and RBTs
4	6.6 ft/2m black Triax cables	Interconnect between SMUs and RBTs
2	SMA female to BNC Male	Adapt SCP2 BNC channels to SMA
1	SMA female to SMB plug	Adapt SCP2 SMB trigger to SMA
1	Power Divider, Male/Female/Male	Connects to Gate side RBT AC IN connector

Supplied tools

The following tools are supplied with the Model 4200-SCS or PIV-A package:

- #1 Phillips screwdriver
- Torque wrench, 8in/lb, with 5/16in head installed

NOTE *The various adapters, cables and hardware used for the pulse projects are shown in Figure 4-50.*

Figure 4-6
Pulse IV—hardware connections

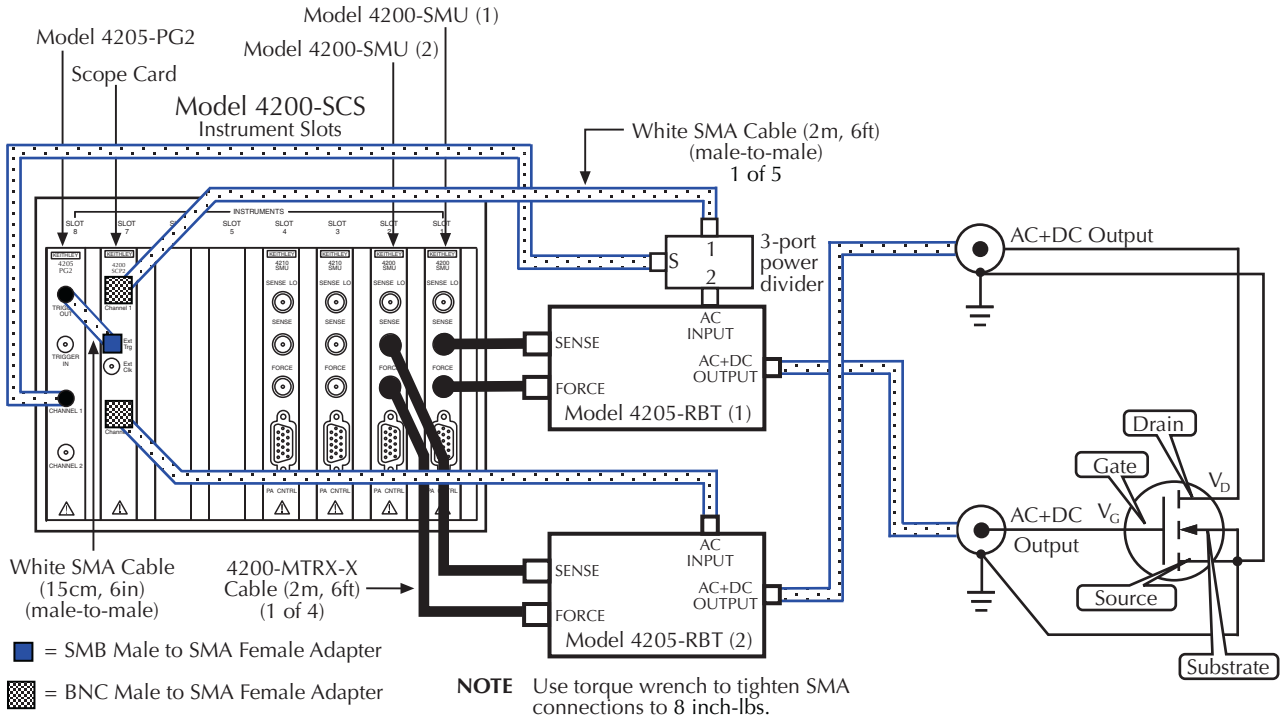
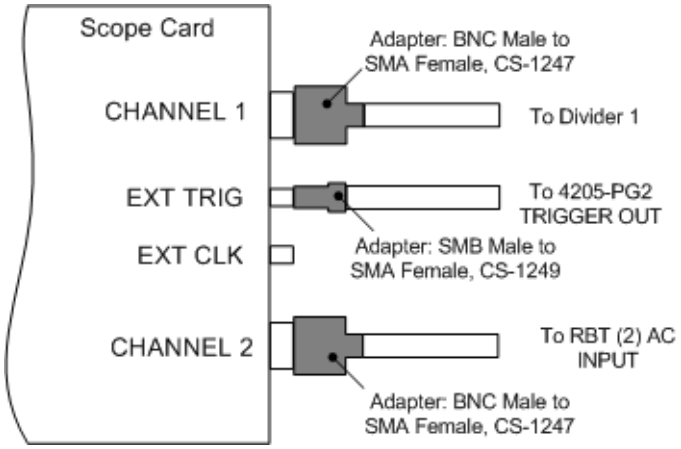


Figure 4-7
Side view of scope card connections



Model 8101- PIV test fixture

The 4200-PIV-A includes a test fixture and DUTs to verify proper PIV-A setup and operation, and is also useful for troubleshooting.

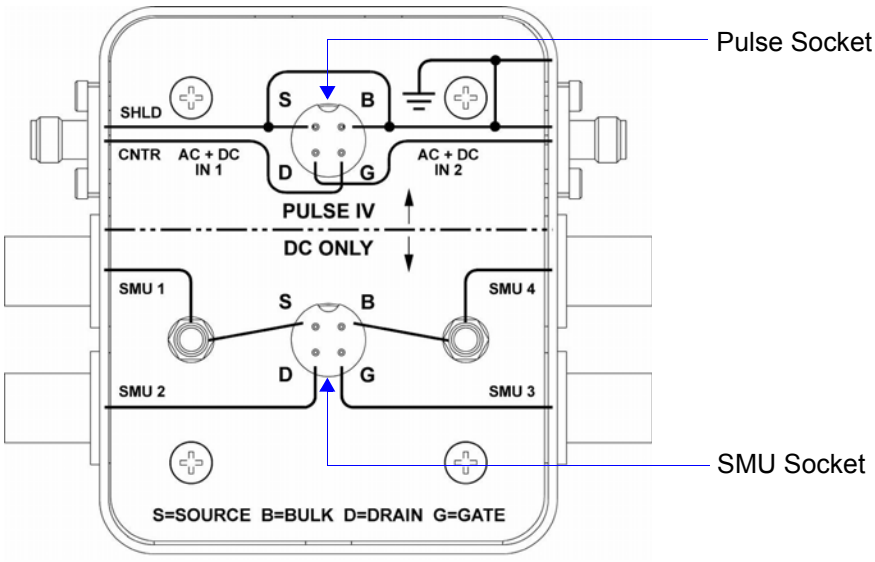
The test fixture, Model 8101-PIV (shown in Figure 4-8), has two electrically separate sockets to support testing three and four leaded devices. The lower socket, located near the latch, is for DC testing with SMUs and uses four Triax connectors. The upper, or back, socket, located near the hinge, is for Pulse IV or Pulse IV + DC testing and uses two SMA connectors. The SMU socket has a triax connector for each of the 4 DUT pins. The pulse socket uses only 2 SMA connectors, where the DUT source and bulk connections are connected to ground (SMA coax shield) and are

optimized for use with the 4200-PIV-A package. The fixture may also be used with the 4200-PIV-Q package, but higher power testing, either DC or pulse IV, requires additional care to prevent damage to the included DUTs. The schematic of the 8101 test fixture is shown in Figure 4-9.

The tests included in the PulseIV-Complete project have parameter defaults that provide reasonable results with the metal TO can DUT (SD-210 nMOS FET).

Figure 4-8
Model 8101-PIV test fixture

Figure 4-9
Model 8101-PIV schematic



Prober Interconnect

The PIV-A package provides both DC and Pulse capability to the DUT pins without re-cabling or switching. The key to this capability is the RBT (remote bias tee), that uses passive electrical components to combine the low frequency DC signals with the high frequency pulse signals. For further information on the RBT, see Section 12 in the Model 4200 Reference Manual.

The cabling from the RBT is SMA coax, that will connect to many RF and DC probe manipulators.

DC Prober Interconnect

For DC structures, an adapter cable (Model 4200-PRB-C) is included to convert from the SMA to dual SSMC connections on DC manipulators. The adapter cable is shown in Figure 4-10.

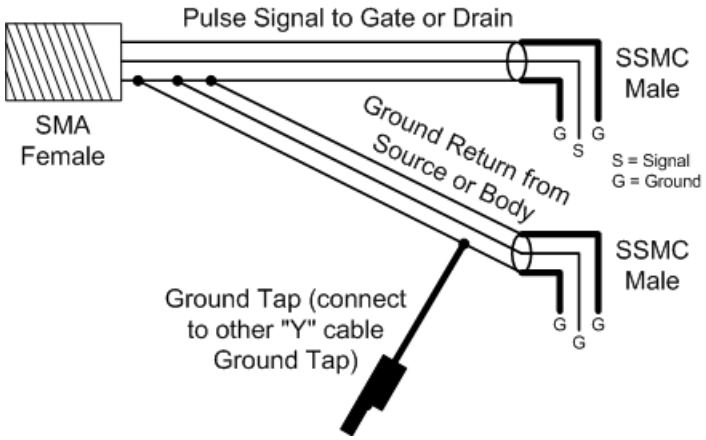
Many DC probe manipulators are available with SSMC connections at the probe needle holder:

- Cascade DCM-2xx Kelvin DC probe manipulators
- Suss Microtec probe tips
- Signatone SCA-50 coaxial probes
- American Probe and Technologies:
  - 74CJ series coaxial probe holder
- Any probe interconnect with SSMC connectors near the probe tip

Figure 4-11 shows the schematic diagram of the PRB-C adapter cable. The Ground Tap is to be connected to the Ground Tap of the second PRB-C adapter cable, as shown in Figure 4-12.

Figure 4-10
PRB-C adapter cable – pulse SMA to SSMC Y

Figure 4-11
Schematic diagram of the PRB-C adapter cable



These SMA to SSMC Y adapter cables are appropriate for on-wafer pulse IV testing of nominally DC structures. Figure 4-12 shows Pulse IV connections from RBTs to DC probes for a DC layout DUT structure, using the PRB-C Y adapter cable.

These “Y” cables are not appropriate for higher frequency devices. The upper frequency limit is not specified, because the effect of actual device layout and probe configuration can have a significant impact. In general, any device that has an F_T much above 1GHz might oscillate when using a DC probe connection scheme and the PRB-C cables.

RF Prober Interconnect

If the device has an RF layout (G-S-G), the Y adapter cables and DC probe manipulators will most likely be insufficient. In the case of RF G-S-G pad layout, do not use the 4200-PRB-C Y cables, simply use the shorter SMA cables (6in/15cm) supplied with the PIV-A package to connect directly from the RBTs to the RF manipulators. The RBT with the power divider is connected to the Gate.

For additional information see the documentation included with the Model 4200-PRB-C (PA-928).

PIV-A interconnect assembly procedure

1. Using the “Supplied interconnect parts” and “Supplied tools”, refer to Figure 4-5, Figure 4-6 and Figure 4-7 to configure the test setup for PIV testing. Use the supplied torque wrench for the SMA connections on the RBTs, power divider and manipulators. Use care when installing the cable to the scope card trigger SMB connector.
2. Perform one of the following procedures to connect the test system to the DUT:
   - For DC structures, prepare the probe connection by disconnecting all DC cables from the SSMC connectors on the needle holders. Continue setup of PIV-A by connecting a PRB-C cable to the 15cm (6in) SMA cable attached to each RBT. Refer to Figure 4-12. Don’t forget to connect the black shield jumpers to each other as shown in the middle of Figure 4-12. Connecting shields together is necessary and very important, as it greatly reduces the inductance that is caused by the loop area of the interconnect.
   - For RF probes, connect the SMA cables from the RBTs to the RF probe manipulators, as shown in Figure 4-13.
   - To use the supplied 8101-PIV test fixture (Figure 4-8), connect the SMA cables from the RBTs to the 8101-PIV Test fixture as shown in Figure 4-14. Install DUT as shown in Figure 4-15.
3. Finish the setup by verifying connections and running a scope-shot test from the Pulse IV-Complete project.

Figure 4-12
Pulse IV connections using PRB-C adapter cables

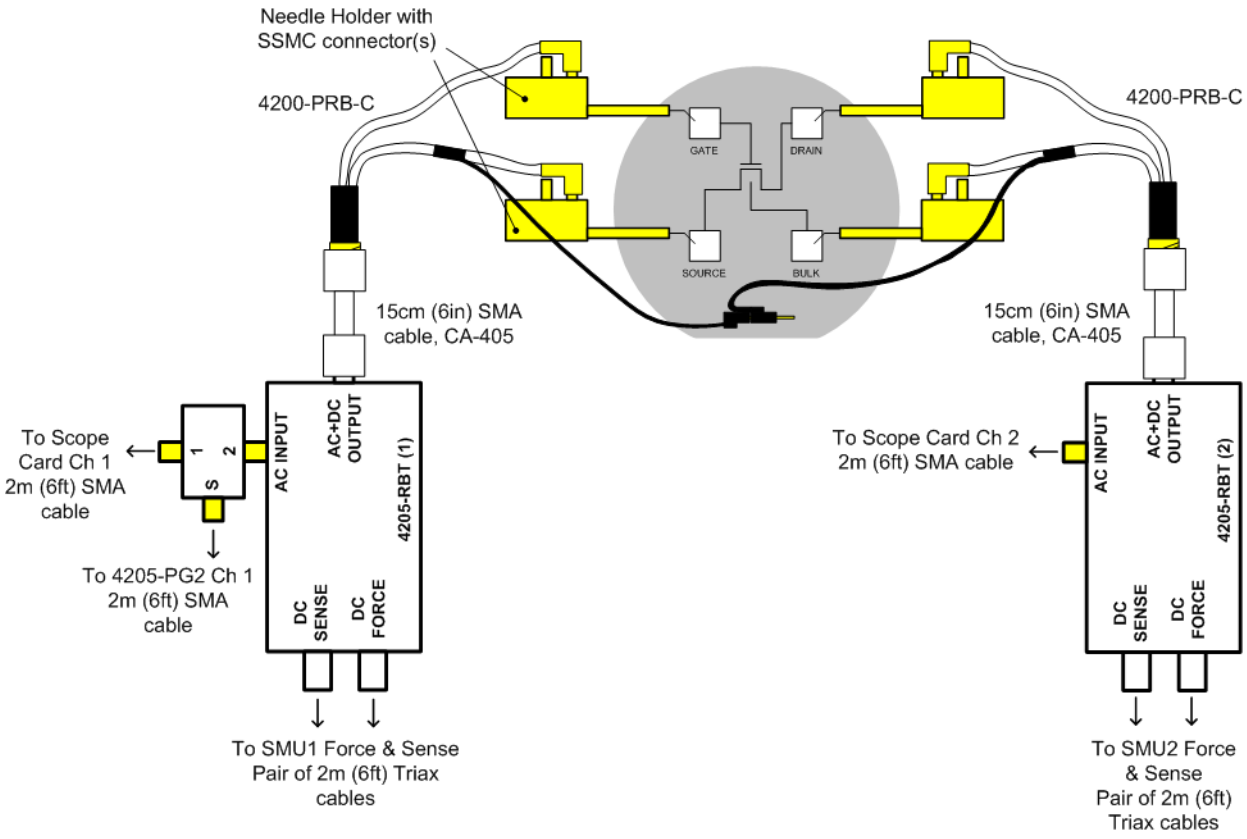


Figure 4-13
Pulse IV connections using RF G-S-G probes

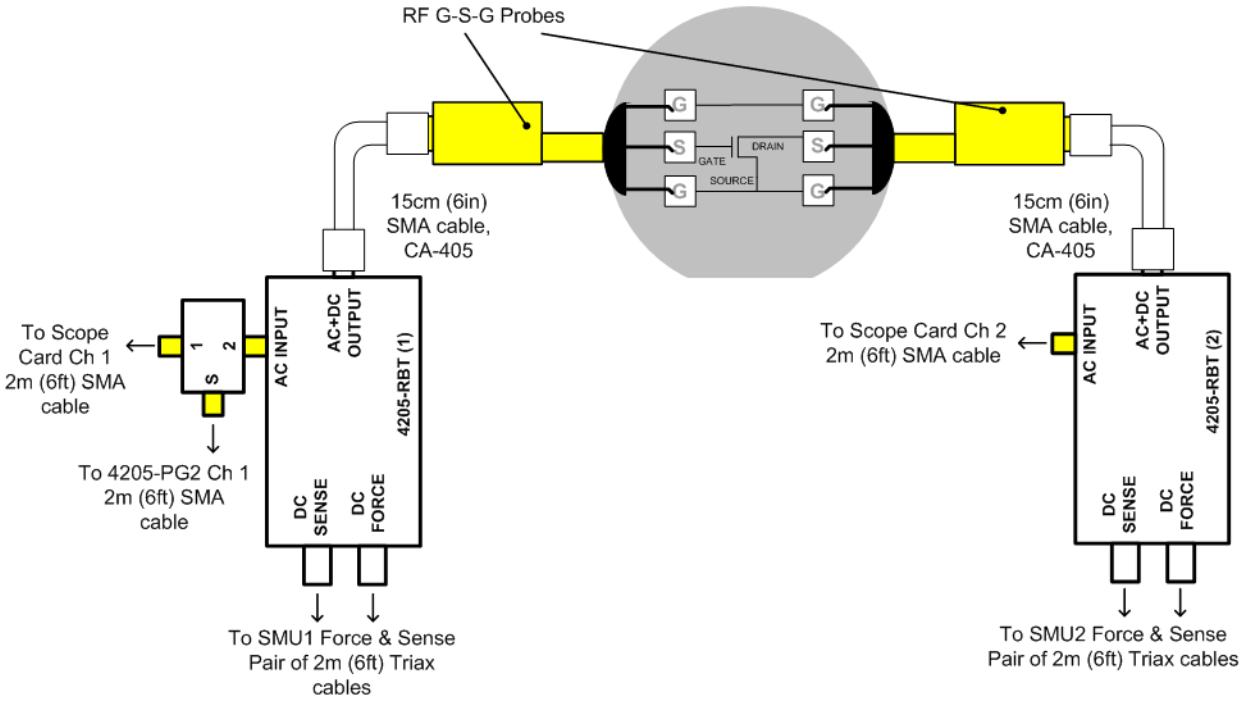


Figure 4-14
Pulse IV connections using the 8101-PIV test fixture

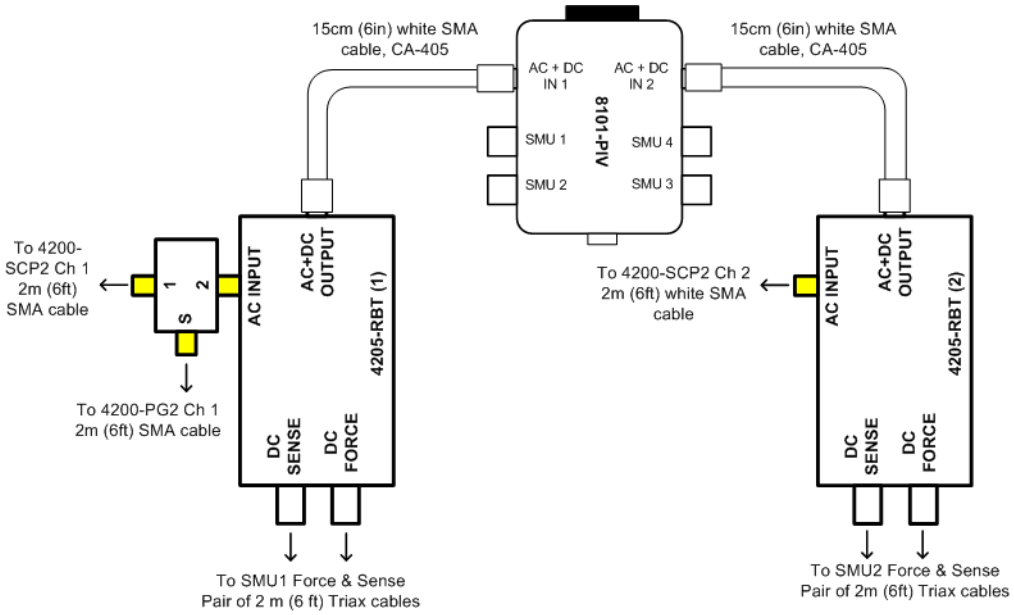
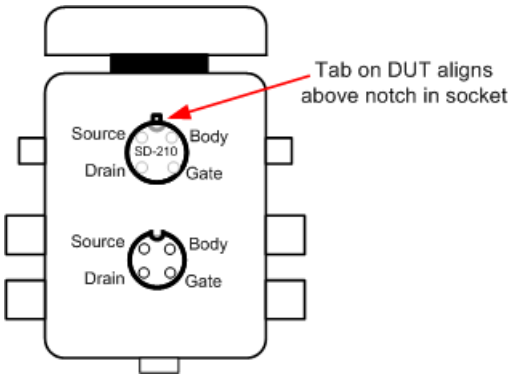


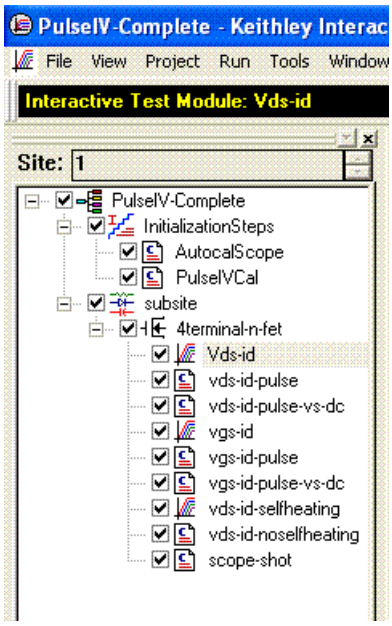
Figure 4-15
DUT inserted in pulse socket of 8101-PIV test fixture



Using the PulseIV-Complete project for the first time

1. Connect PIV-A as explained above in "PIV-A interconnect assembly procedure".
2. If KITE is not running, start KITE by double-clicking the **KITE** icon on the Model 4200 desktop.
3. Open the PulseIV-Complete project as follows:
   a. Click **File** > Select **Open Project**.
   b. If necessary, move up one level to display all the project folders, and double-click the _Pulse folder.
   c. Double-click **Pulse-IV-Complete.kpr** to open the project. Figure 4-16 shows the project plan that is displayed on the left side of the KITE window.
4. Connect or touch-down on the chosen device under test (DUT).
5. Verify the setup as follows:
   a. Step 1: Follow the instructions for "Running scope-shot" to validate proper setup and operation of the PIV-A package. Ensure that both the gate and drain waveforms are visible and do not have any significant ringing or overshoot (see Figure 4-25).
   b. Step 2: Try running Vds-id-pulse ("Running vds-id-pulse UTM") and/or Vgs-id-pulse ("Running vgs-id-pulse UTM")and look for a characteristic response. If desired, DC IV tests may also be run ("Running Vds-id DC ITM", "Running vgs-id DC ITM"). Once both the scope-shot and a pulse IV test have been verified, pulse system calibration can be performed.
6. Calibration: Perform the necessary pulse calibrations explained in "Running AutocalScope" and "Running PulseIVCal".
7. After successful pulse calibrations, the system is now ready to be used for pulse and DC characterization of transistor devices.

Figure 4-16
Project plan for Pulse-IV Complete



Running AutocalScope

AutocalScope should be run before any pulse calibration is performed. For best Pulse IV results, the AutocalScope should also be run before the first experiments of the day.

1. The Model 4200-SCS should be turned on at least 30 minutes before performing any calibration or measurements.
2. Double-click AutocalScope in the Project Navigator (Figure 4-16).
3. Click the green Run button.
4. Follow the instructions given in the pop-up dialog box and disconnect all connections to the scope card.
5. The scope performs an autocal, that takes about one minute.
6. The test is complete when the Run button turns green. In the Sheet tab, autoCalStatus=0 means that there were no errors.
7. Reconnect the cables to the scope card. Use care when installing the cable to the scope card trigger SMB connector.

Running PulseIVCal

Verify proper setup by running a scope-shot. For on-wafer testing, have a through, or short, structure available, or ensure that sharing a pad for both the gate and drain probes provides a good connection. There are two steps to the calibration, open, and through/short.

1. If not already performed, run AutocalScope as explained above.
2. Double-click PulseIVCal in the Project Navigator (Figure 4-16).
3. Click the green Run button to start the PulseIVCal.
4. Click OK on the first dialog box to continue the PulseIVCal (Figure 4-17, left dialog box).
5. The second dialog box requests that the probe pins be raised from the wafer, breaking contact. Raise the probe pins or lower the wafer to create the Open condition. If using the 8101-PIV Test fixture, ensure that the pulse socket (near the fixture hinge), is empty.
6. Click OK on the Open dialog box (Figure 4-17, middle dialog box). The Open portion should take about one minute.

7. The third dialog box requests that the probe pins be connected to each other through another device. Lower the probes onto another device.
8. Click OK on the Through dialog box (Figure 4-17, right dialog box). The Through portion should take about one minute.
9. The test is complete when the Run button turns green. In the Sheet tab, cal_pulseiv=0 implies that there were no errors.
10. The system is now ready to test regular devices.

Figure 4-17
PulseIVCal dialog boxes

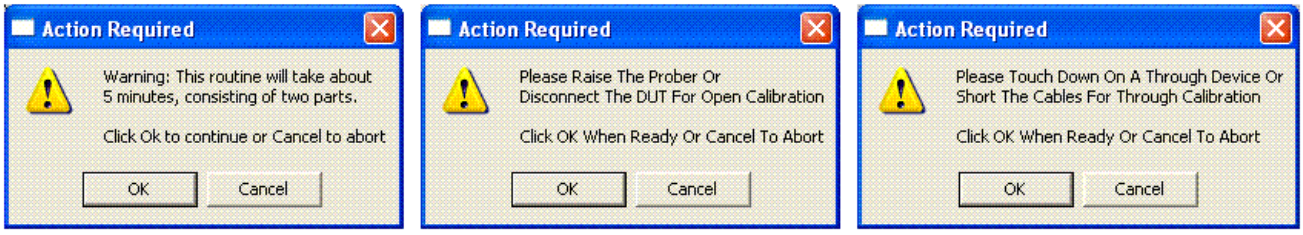
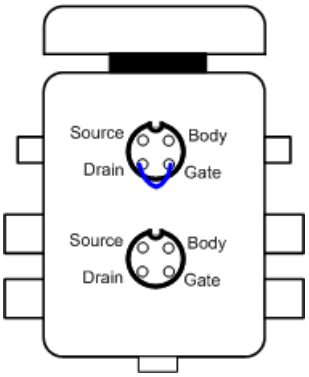


Figure 4-18
8101-PIV shorted/through socket

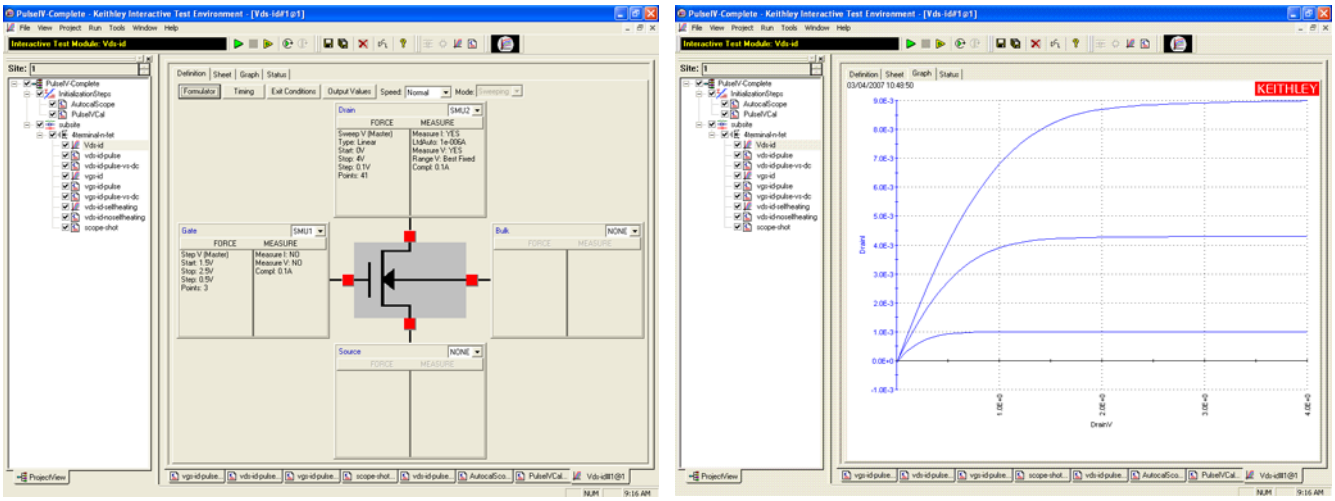


Running Vds-id DC ITM

The default settings sweep the drain from 0-4V in 100mV steps while stepping through three gate voltages: 1.5V, 2.0V and 2.5V (see Figure 4-19). When changing these settings, make note of the voltages and step size so that the same settings can be used in vds-id-pulse.

1. Double-click Vds-id ITM in the Project Navigator (Figure 4-16).
2. Click the green Run button. Three Vds-id curves will be generated and displayed on the graph.

Figure 4-19
Default definition and typical graph for Vds-id

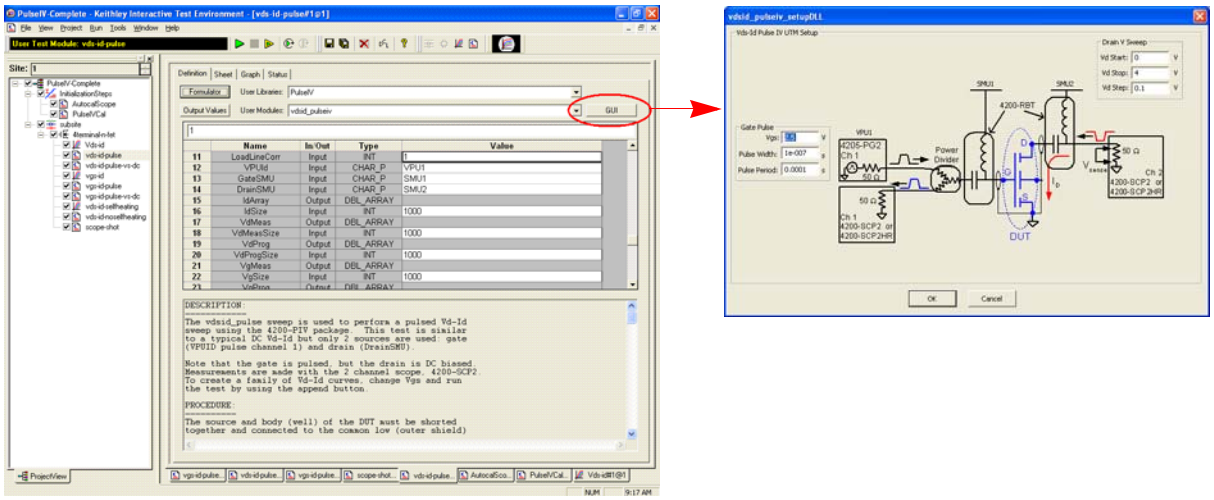


Running vds-id-pulse UTM

The default vds-id-pulse test uses (see Figure 4-20) the same drain voltage settings as the DC Vds-id. The vds-id-pulse does not have the automatic step capability of the DC Vds-id. There are two ways to generate a family of pulse IV curves. The easier way is to use the vds-id-pulse-vs-DC (see "Running vds-id-pulse-vs-dc UTM" below). If using the 8101-PIV test fixture, insert the metal can (SD-210) DUT as shown in Figure 4-15. To run the three gate voltages using single curve vds-id-pulse, perform the following steps:

1. Ensure that the VdStart, VdStop, VdStep values match the values in the DC Vds-id. To sweep from a high to a low voltage, enter voltages so that VdStop < VdStart and use a negative value for VdStep. If any values need to be modified, remember to press the Enter key after typing in the value.
2. Set Vgs to the first voltage. The default is 1.5V. Make sure to press the Enter key after typing in the value.
3. Click the green Run button.
4. After the test is finished, set Vgs to the second voltage. The default is 2.0V.
5. Click the yellow and green Append button.
6. After the test is finished, set Vgs to the third voltage. The default is 2.5V.
7. Click the yellow and green Append button.
8. To add or update the DC results on the pulse Graph, perform the procedure for "Comparing DC and pulse results".

Figure 4-20
Default Definition tab and GUI For vds-id-pulse

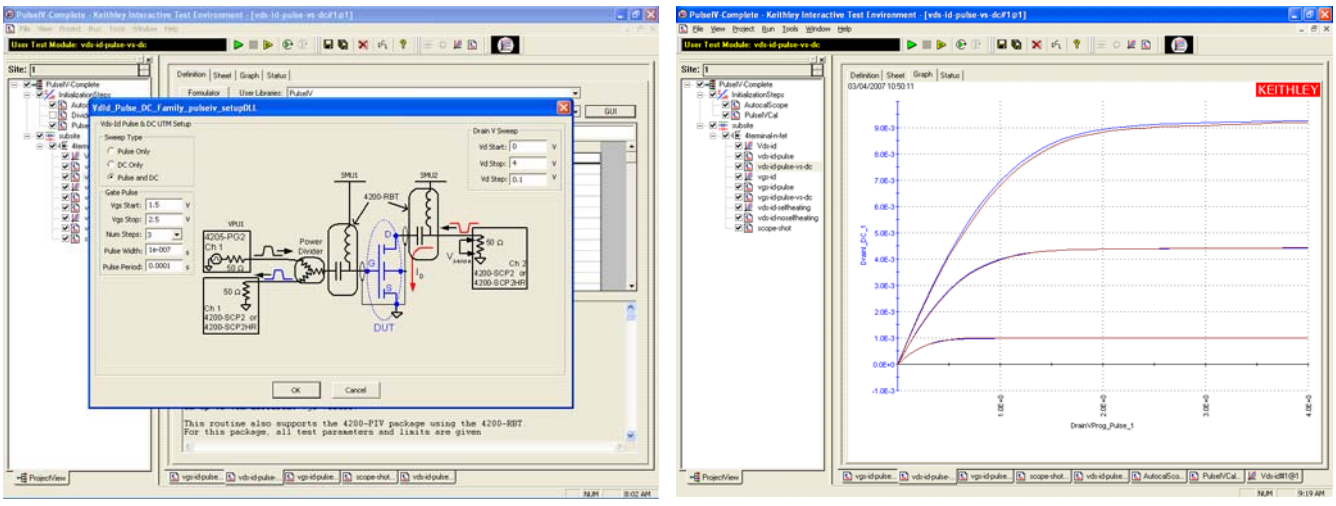


Running vds-id-pulse-vs-dc UTM

The default settings are the same as the vds-id-pulse UTM, with the addition of the DC measurement parameters (see Figure 4-21):

1. If measurement parameters (pulse average, NPLC, measure range) need to be set, use the definition table.
2. (Optional) If only source parameters need to be changed, use the UTM GUI by clicking the GUI button on the vds-id-pulse-vs-DC test. Modify the source parameters in the GUI, and click OK when finished.
3. Click the green Run button. For a test with three curves and 40 points per curve, the test should take about 1.5-2 minutes. During the test, neither the Graph tab or Sheet tab is updated.

Figure 4-21
Default definition and typical graph for vds-id-pulse-vs-dc

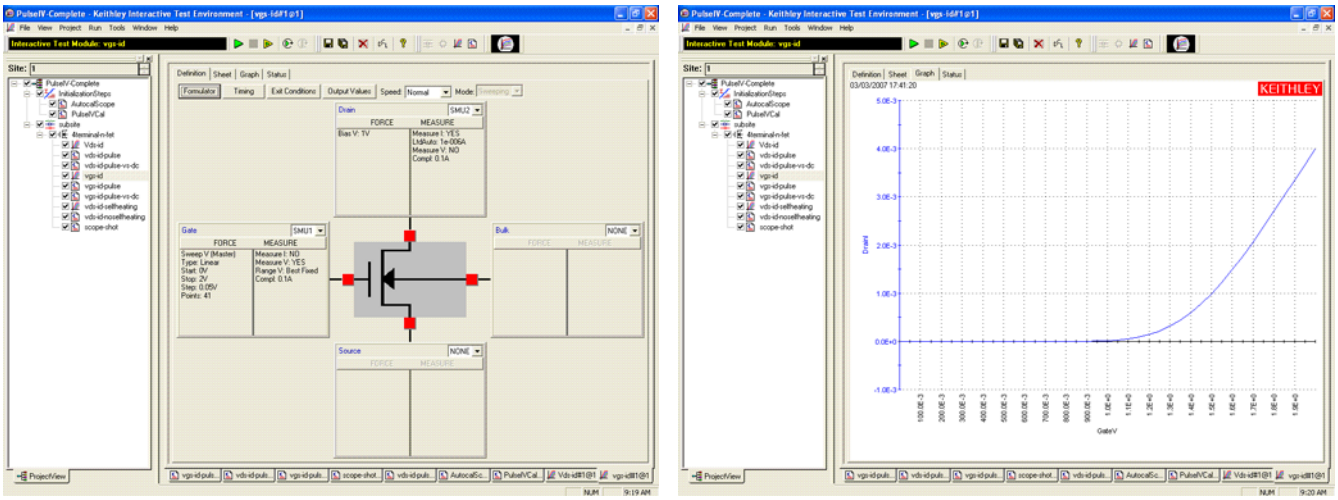


Running vgs-id DC ITM

The default settings sweep the pulses on the gate from 0-2V in 50mV steps and set the drain voltage to 1V (see Figure 4-22). When changing these settings, note the voltages and step size used so the same settings can be used in vgs-id-pulse.

1. Double-click vgs-id ITM in the Project Navigator.
2. Click the green Run button. The Vgs-id curve will be generated and displayed on the graph.

Figure 4-22
Default definition and typical graph for vgs-id

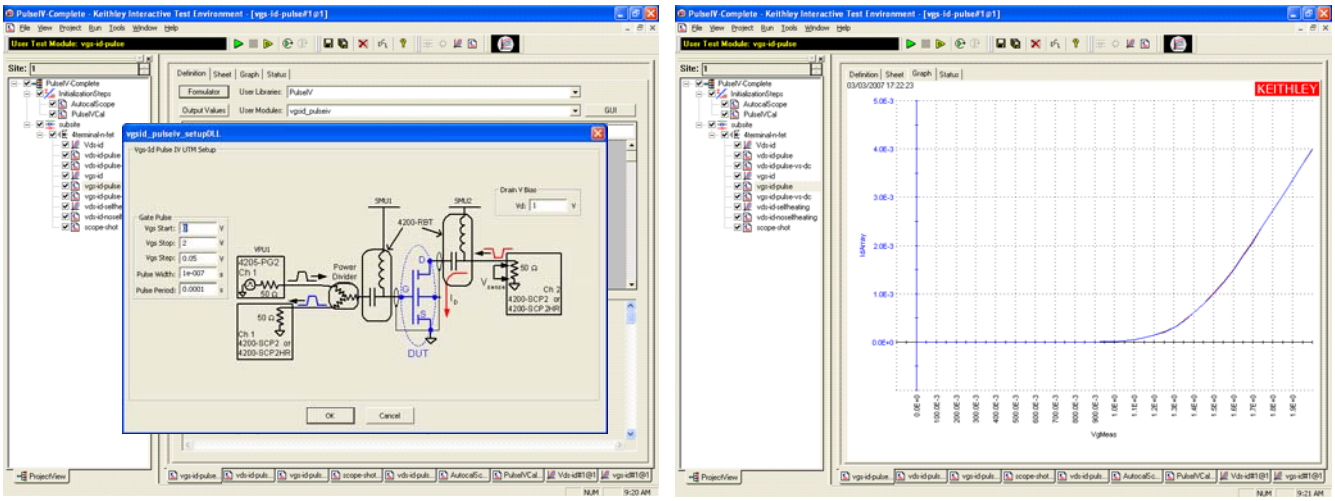


Running vgs-id-pulse UTM

The default Vgs-id-pulse uses the same default settings as the DC Vgs-id (see Figure 4-23). If comparing Vgs-Id results for DC and Pulse IV, use this pulse-only routine and the "Comparing DC and pulse results" or use the single DC and Pulse UTM as described in "Running vgs-id-pulse-vs-dc UTM". Alternately, the source values may be entered using the UTM GUI:

1. Ensure that the Vds, VgStart, VgStop, VgStep values match the values in the DC Vgs-id. To sweep from a high to a low voltage, enter voltages so that VdStop < VdStart and use a negative value for VdStep. If any values need to be modified, remember to press the Enter key after typing in the value.
2. Click the green Run button.
3. To add or update the DC results on the pulse Graph, perform the procedure for “Comparing DC and pulse results”.
4. To reduce noise, the smaller subthreshold currents of this test require a larger number of measurements to be averaged. For best results on smaller signals (Id < 500uA), use AverageNum = 0 to enable the adaptive filtering mode, where lower scope ranges will use a large AverageNum and higher ranges a lower AverageNum. If desired, a fixed number may be entered for AverageNum.

Figure 4-23
Default definition and typical graph for vgs-id-pulse

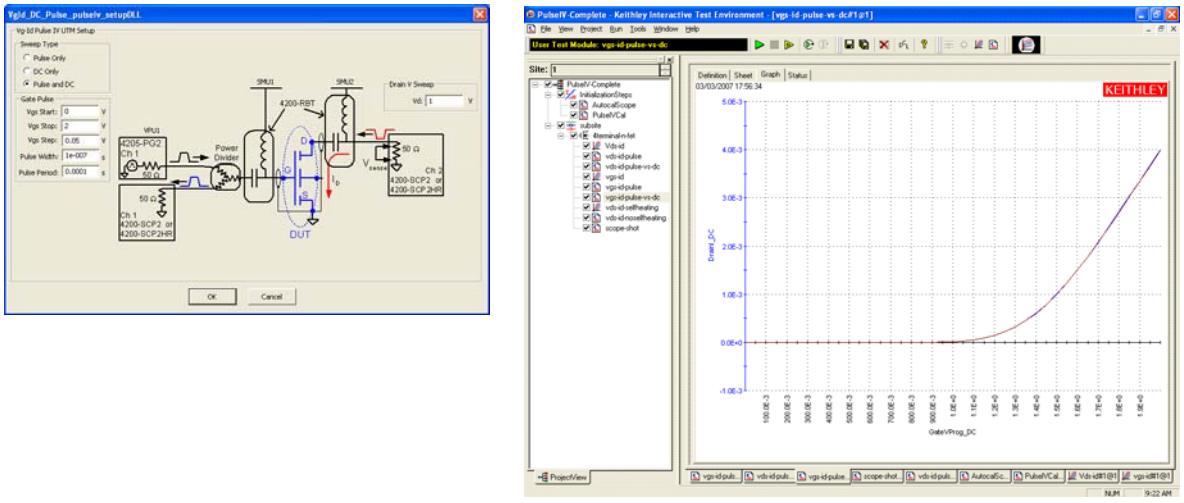


Running vgs-id-pulse-vs-dc UTM

Instead of using the separate vgs-id ITM and vgs-id-pulse UTM to compare DC and pulse Vg-Id results, the vgs-id-pulse-vs-DC UTM combines both DC and pulse tests (see Figure 4-24):

1. If measurement parameters (pulse average, NPLC, measure range) need to be set, use the definition table.
2. (Optional) If only source parameters need to be changed, use the UTM GUI by clicking on the GUI button on the Vds-id-pulse-vs-DC test. Modify the source parameters and click OK when finished.
3. Click the green Run button. For a test with 40 points, the test should take about one minute. During the test, neither the graph tab or sheet tab is updated.

Figure 4-24
Default GUI definition and typical graph for vgs-id-pulse-vs-dc



Running scope-shot

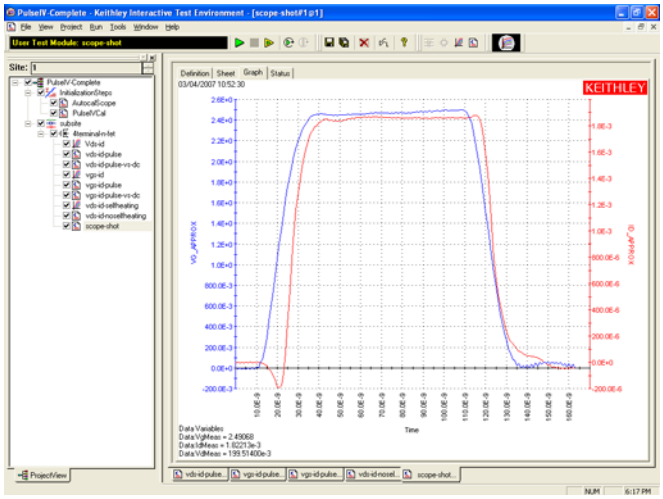
The scope-shot test is used to verify proper connection and system setup. The waveform shown in Figure 4-25 is a typical result; actual results should be similar. If waveform has significant ringing or overshoot, the pulse IV tests will not provide good results. Check the pulse interconnects to ensure proper cabling and ensure all connections are tight. If using 4200- PRB-C cables (Y

adapter cable for pulsing with DC interconnect and structures), ensure that the two ground lugs are connected together.

The left pulse curve (blue) is the pulse applied to the gate. The displayed waveform data has approximate calibration factors applied, but the calibrated measurement is given in the lower left portion of the Graph tab. The right pulse curve (red) is the drain current, shown with approximate calibration factors applied, with the calibrated Vd and Id measurements listed in the lower left corner of the graph.

Note that the AverageNum value specifies the number of pulses that are averaged together to provide the data. The DUT will have more pulses applied than AverageNum due to other test factors, such as load line correction and measurement autoranging.

Figure 4-25
Typical graphical result for scope-shot



Adjustable parameters in scopeshot_cal_pulseiv

Vds	DC voltage for the drain
Vgs	Pulse voltage level for gate
PulseWidth	Vgs pulse width, full width half maximum (FWHM)
PulseAmplitude	Vgs, gate voltage pulse
PulsePeriod	Vgs pulse period. When using the Pulse IV setup with RBTs, use a PulsePeriod 1000 x PulseWidth, to keep the pulse duty cycle less than or equal to 0.1%. For most cases, it is best to use 200E-6, that will allow an appropriate duty cycle across the range of supported 40-150ns pulse widths.
GateRange	Scope card gate voltage range. Use 0 for autoranging, or a specific value for a fixed range. The scope range is centered around zero, so the 5V range on the scope covers -2.5V to +2.5V. As an example, for a 3V signal, use GateRange = 10 (-5V to +5V), not 5 (-2.5 to +2.5V). Available ranges for the scope card: 0.05, 0.10, 0.25, 0.5, 1, 2, 5, 10 V.
DrainRange	Scope card drain voltage range. Use 0 for autoranging, or a specific value for a fixed range. To calculate an appropriate fixed range, use DrainRange = (Estimated Id) x 50 x 2. See above for valid scope card ranges.
AverageNum	Number of pulses to average. For larger currents, Id 500µA, AverageNum = 10-25 is usually sufficient. For smaller Id, use 50-100. Larger values provide minimal additional improvement. All pulse IV tests have this setting, that controls how many pulses are used to return a result, not how many pulses are sent to the DUT.
LoadLineCorr	Turns drain side load line correction on or off. This is similar to the vds-id-pulse and vgs-id-pulse tests and is a routine to ensure that the desired Vd is provided to the drain DUT terminal, regardless of the amount of Id flowing through the DUT.
VPUID	4205-PG2 identification string: VPUID = VPU1.
GateSMU	SMU for DC Vg. Default is SMU1, but any other available SMU may be used.
DrainSMU	SMU for DC Vd. Default is SMU2, but any other available SMU may be used.

Tips for using Pulse IV

- Confirm connection: Use scope-shot as the first test after touching down on a device to confirm that there is proper connection to the DUT, *before* running PulseIVCal or any pulse tests.
- Always calibrate after any setup changes (new probe tips or manipulators, cable replacement).
- Proper pulse IV performance can be verified by testing a device that does not exhibit any selfheating or charge trapping effects. The 8101-PIV Test Fixture and SD-210 DUT provide good DC and pulse correlation for Id < 10mA and Vd < 5V. For Id < 1mA, set AverageNum = 0 to use the Adaptive Filtering. If a fixed number for AverageNum is desired, use AverageNum = 2000 for Id < 500µA.
- Pulse IV measurements have less resolution and sensitivity than typical DC results, so test parameters, such as averaging or smaller steps sizes, and post-test processing, such as curve fitting, may be required to obtain roughly equivalent results.

Comparing DC and pulse results

There are two methods for comparing DC and Pulse IV results. The first method uses the UTMs that combine pulse and DC tests: Vds-id-pulse-vs-dc, vgs-id-pulse-vs-dc. The second method, described below, uses the data in the Sheet tab of KITE to compare any results across tests. This procedure explains how to copy the DC results into a pulse UTM to allow comparison between pulse and DC IV results in a single graph.

1. In the PulseIV-Complete project navigator, double-click the **Vds-id** ITM in the project navigator.
2. Click the **Sheet** tab.
3. Choose the desired results worksheet. If there is only one set of curves, then the results are in the **Data** tab. If additional tests have been appended, choose the desired Append tab.
4. Highlight all of the data in the desired worksheet by clicking the upper-left **Entries Selection Cell** shown in Figure 4-26.
5. To copy the data, right-click on the **Entries Selection Cell** and select **Copy** from the drop-down menu.
6. In the PulseIV-Complete project navigator, double-click on **Vds-id-pulse** UTM.
7. Click the **Sheet** tab.
8. Click the **Calc** worksheet tab.
9. Click on cell A1.
10. To paste DC data into the pulse Calc worksheet, right-click on cell A1 and select **Paste** from the drop-down menu. If there is previous data in the Calc worksheet, the paste operation will overwrite it. Figure 4-27 shows the data that was pasted from the Vds-id ITM.
11. All comparison DC and pulse data is now located in the same test. The graph needs to be defined to display the DC data located in the Calc worksheet.
12. For Vds-id-pulse, click on the **Graph** tab.
13. Right-click on the graph and select the first option (**Define Graph**) from the drop-down menu. The Define Graph dialog box is shown in Figure 4-28.
14. In Define Graph, click the cells in column Y1, to add the appropriate DC curves. In this case, three Vdsid curves have been added: DrainI(1), DrainI(2), DrainI(3). These Y1 cells are circled in red. All three of these Data Series are located in the Calc worksheet, as noted in the Sheet column. Also shown in Figure 4-28 is graph with the three added curves.
15. To change graph colors or add data point shapes/patterns, move the cursor along the desired curve until the pointer appears. With the pointer displayed, right-click to get the Data Series Properties dialog box. Select a Shape property to demarcate each data.
16. To verify pulse operation, use a DUT that does not exhibit any self-heating or transient charging effect. In a properly configured and calibrated system, the pulse IV results should correlate to the DC results within ±4%, with many results less than or equal to 2%, when testing a device that does not exhibit heating or charging effects.

Figure 4-26
Highlighting all entries in vds-id data sheet

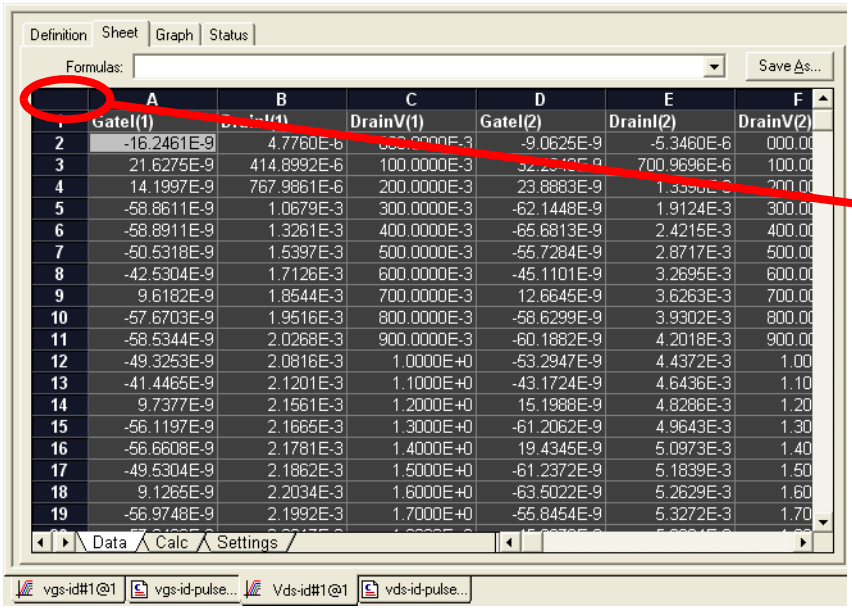
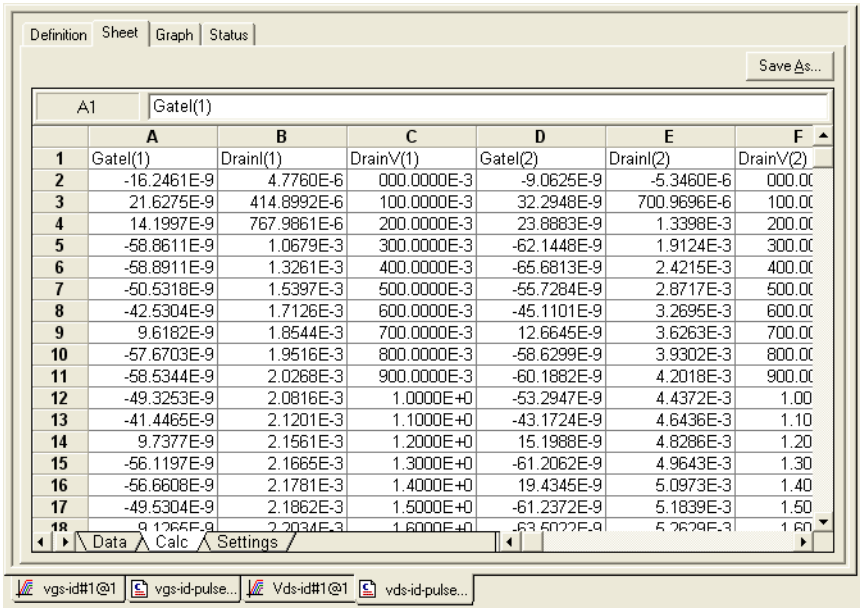
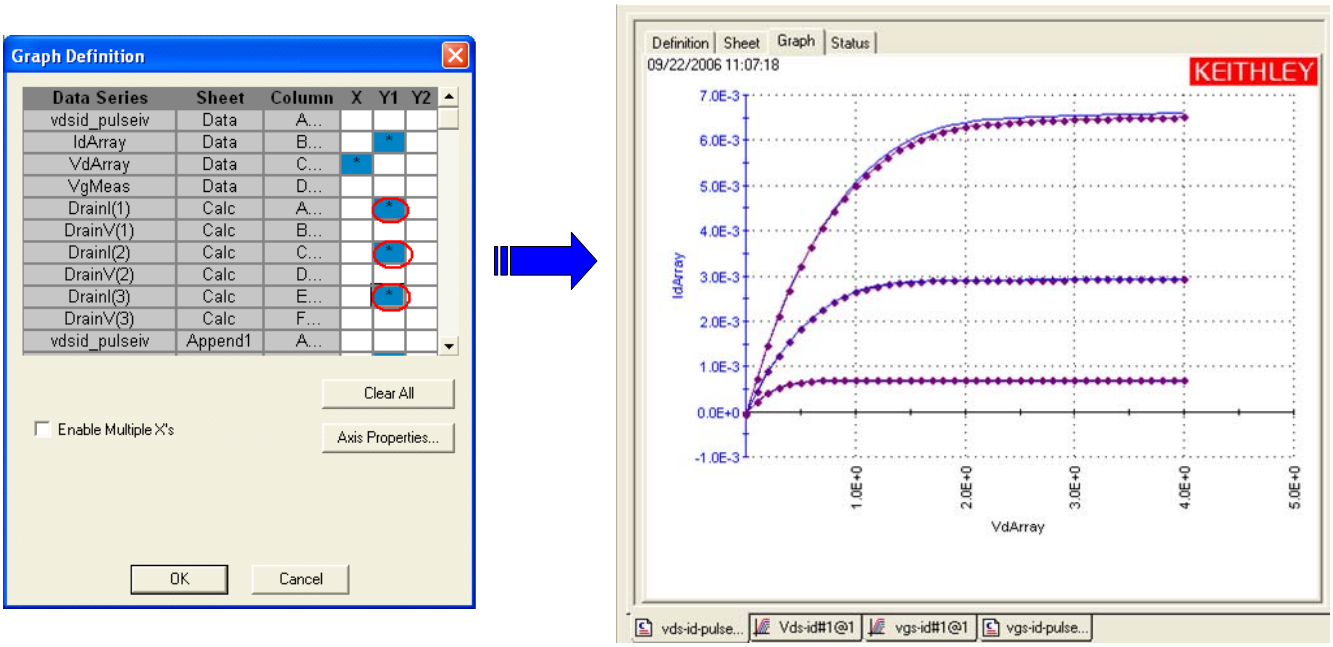


Figure 4-27
Data from vds-id pasted into vds-id-pulse calc sheet



	A	B	C	D	E	F
1	Gatel(1)	Drainl(1)	DrainV(1)	Gatel(2)	Drainl(2)	DrainV(2)
2	-16.2461E-9	4.7760E-6	000.0000E-3	-9.0625E-9	-5.3460E-6	000.0(
3	21.6275E-9	414.8992E-6	100.0000E-3	32.2948E-9	700.9696E-6	100.0(
4	14.1997E-9	767.9861E-6	200.0000E-3	23.8883E-9	1.3398E-3	200.0(
5	-58.8611E-9	1.0679E-3	300.0000E-3	-62.1448E-9	1.9124E-3	300.0(
6	-58.8911E-9	1.3261E-3	400.0000E-3	-65.6813E-9	2.4215E-3	400.0(
7	-50.5318E-9	1.5397E-3	500.0000E-3	-55.7284E-9	2.8717E-3	500.0(
8	-42.5304E-9	1.7126E-3	600.0000E-3	-45.1101E-9	3.2695E-3	600.0(
9	9.6182E-9	1.8544E-3	700.0000E-3	12.6645E-9	3.6263E-3	700.0(
10	-57.6703E-9	1.9516E-3	800.0000E-3	-58.6299E-9	3.9302E-3	800.0(
11	-58.5344E-9	2.0268E-3	900.0000E-3	-60.1882E-9	4.2018E-3	900.0(
12	-49.3253E-9	2.0816E-3	1.0000E+0	-53.2947E-9	4.4372E-3	1.00
13	-41.4465E-9	2.1201E-3	1.1000E+0	-43.1724E-9	4.6436E-3	1.10
14	9.7377E-9	2.1561E-3	1.2000E+0	15.1988E-9	4.8286E-3	1.20
15	-56.1197E-9	2.1665E-3	1.3000E+0	-61.2062E-9	4.9643E-3	1.30
16	-56.6608E-9	2.1781E-3	1.4000E+0	19.4345E-9	5.0973E-3	1.40
17	-49.5304E-9	2.1862E-3	1.5000E+0	-61.2372E-9	5.1839E-3	1.50
18	9.1265E-9	2.2034E-3	1.6000E+0	-63.5022E-9	5.2629E-3	1.60
19	-56.9748E-9	2.1992E-3	1.7000E+0	-55.8454E-9	5.3272E-3	1.70

Figure 4-28
Graph Definition dialog box and resulting graph that shows the three added curves



References

1. Jenkins, K.A. Sun, J.Y.-C. Gautier, J., “Characteristics of SOI FET's under pulsed conditions,” *IEEE Transactions on Electron Devices*, vol 44, issue 11, pp. 1923-1930, Nov 1997.
2. A. Kerber, E. Cartier, L. Pantisano, M. Rosmeulen, R. Degraeve, T. Kauerauf, G. Groeseneken, H. E. Maes, and U. Schwalke, "Characterization of the VT instability in SiO2/ HfO2 gate dielectrics," Proc. IEEE., IRPS pp. 41, 2003.
3. C. D. Young, R. Choi, J. H. Sim, B. H. Lee, P. Zeitzoff, Y. Zhao, K. Matthews, G. A. Brown, and G. Bersuker, “Interfacial Layer Dependence of HfSixOy Gate Stacks on Vt Instability and Charge Trapping Using Ultra-short Pulse I-V Characterization,” Proc. IEEE IRPS, p. 75-79, 2005

Pulse IV UTM descriptions

The pulse IV user library contains modules required to provide low duty cycle pulsed IV testing. The modules contained in the pulse IV user library are listed in Table 4-16 with detailed information following the table.

Table 4-16
Pulse IV UTMs

User Module	Description
cal_pulseiv	Performs a cable compensation routine.
vdsid_pulseiv	Performs a pulsed Vd-Id sweep.
vdId_Pulse_DC_Family_pulseiv	Performs a Pulsed vs. DC Vd-Id sweep.
vgId_DC_DC_pulseiv	Performs a Pulsed vs. DC Vg-Id sweep.
vgsid_pulseiv	Performs a pulsed Vg-Ig sweep.
scopeshot_cal_pulseiv	Used to display a single Pulse IV scopeshot_pulseiv.
scopeshot_pulseiv	Displays a single Pulse IV scopeshot.
vdsid_pulseiv_demo	Performs a pulsed Vd-Id sweep, with simplified parameter list.
vgsid_pulseiv_demo	Performs a pulsed Vg-Id sweep, with simplified parameter list.

cal_pulseiv

Description The cal_pulseiv module is used to perform a cable compensation routine for the 4200-PIV package. This routine permits the system to compensate for losses in the cabling from the 4200 to the connection to the device under test (DUT). Use this routine during initial system setup and whenever changes are made in any part of the interconnect (cables, 4200 Remote Bias Tees, Probe manipulators or pins).

There are two main steps to this procedure:

- Open cal—the Gate signal is measured while there is no connection to the DUT.
- Through cal—the Drain signal is measured while making contact on a Through structure (or by shorting the two 4200 Remote Bias Tee AC+DC Outputs with an appropriate cable or adapter).

The factors generated by this routine are used during any testing where the 4200 Remote Bias Tees are used (vdsid_pulse, vgsid_pulse). Make sure to set the appropriate values for the cal_pulseiv parameters Table 4-17. Table 4-18 and Table 4-19 contain outputs and return values, respectively.

Connection The source and body (well) of the DUT must be shorted together and connected to the common low (outer shield) of the SMA cables on the AC+DC output of the Remote Bias Tees. The RBT connected to GateSMU (with the Power Divider) should be connected to the gate. The RBT connected to DrainSMU should be connected to the drain. For detailed connection information, refer to the .

Table 4-17
Inputs for cal_pulseiv

Input	Type	Description	Default
vRange	int	The pulse generator card voltage source range to be calibrated (V). Valid values are: 5, 20.	5

Table 4-17
Inputs for cal_pulseiv

Input	Type	Description	Default
PulsePeriod	double	The pulse period for the Vgs pulse. The period can be set from 40 us to 1s (10ns resolution). The period must be set so that the Duty Cycle (DC) is no more than 0.1%.	100e-6
Vs_Size Vm1_size Vm2_size	int	Set to a value that is at least equal to the number of steps in the sweep and all three must be the same value.	100 100 100

Table 4-18
Outputs for cal_pulseiv

Output	Type	Description
Vs	double *	The pulse source value (V).
Vm1	double *	The measured voltage from channel 1 of the scope card.
Vm2	double *	The measured voltage from channel 2 of the scope card.

Note: These outputs are included for compatibility with older setups. They no longer return any information.

Table 4-19
Return values for cal_pulseiv

Value	Description
0	Ok
-13001	Array Sizes Do Not Match
-13002	Arrays Not Large Enough For Data
-13003	Invalid Instruments
-13004	Unable To Malloc Memory
-13005	Unable To Find Delay Between Channels
-13006	Scope Measurement Error
-13007	Unable To Write To Calibration Files
-13008	Invalid Range
-13009	Invalid Calibration Type
-13010	Calibration Data Does Not Meet Correlation Specification
-13998	Calibration Constant Error
-13999	Divider Cal Error

vdsid_pulseiv

Description The vdsid_pulse sweep is used to perform a pulsed Vd-Id sweep using the 4200-PIV package. This test is similar to a typical DC Vd-Id but only 2 sources are used: gate (VPUID pulse channel 1) and drain (DrainSMU). The gate is pulsed, but the drain is DC biased.

Measurements are made with the 2 channel scope card. To create a family of Vd-Id curves, change Vgs and run the test by using the append button. Make sure to set the appropriate values for the Vds-Id parameters (Table 4-20). Table 4-21 and Table 4-21 contain outputs and return values, respectively.

Connection The source and body (well) of the DUT must be shorted together and connected to the common low (outer shield) of the SMA cables on the AC+DC output of the 4200 Remote Bias Tees (RBT). The RBT connected to GateSMU (with the Power Divider) should be connected to the gate. The RBT connected to DrainSMU should be connected to the drain. For detailed connection information, refer to the .

Table 4-20
Inputs for vdsid_pulseiv

Input	Type	Description
Vgs	double	The pulsed gate-source voltage bias, output by channel 1 of the pulse generator card (VPUID).
Vg_off	double	The DC bias applied by the GateSMU to put device in the OFF state. Normally set to 0V for enhancement FETs (may be non-zero for depletion FETs).
VdStart	double	The starting sweep value for Vd, output by the DrainSMU (defined below).
VdStop	double	The final sweep value for Vd, output by the DrainSMU (defined below).
VdStep	double	The sweep step size for the Vd sweep, output by the DrainSMU (defined below).
PulseWidth	double	The Vgs pulse width (PW). The PW can be 40ns to 150ns (10ns resolution). Pulses wider than 150ns will begin to be attenuated by the capacitor in the 4200 Remote Bias Tee.
PulsePeriod	double	The pulse period for the Vgs pulse. The period can be set from 100µs to 1s (10ns resolution). The period must be set so that the Duty Cycle (DC) is no more than 0.1%. The period is most easily calculated by multiplying the largest desired pulse width (PW) by1000. Example: PW = 150 ns, so Period = 150 us.
AverageNum	int	The number of pulses to average at each step of the sweep. For best low signal performance, set AverageNum = 0 for Adaptive Filtering.
GateRange	double	The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.050, 0.1, 0.2, 0.5, 1, 2, 5, 10.
DrainRange	double	The voltage measure range for the scope channel measuring the Drain. Use 0 for scope autoranging, or specify a voltage value for a fixed range, where V = I * 50 ohm. Valid voltages are 0.050, 0.1, 0.2, 0.5, 1, 2, 5, 10.
LoadLineCorr	int	Determines whether to use load line correction to compensate for the voltage drop caused by the 50 ohm sense resistor used to measure the drain current (Id). 1 = load line correction active. 0 = no load line correction.
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with the 4200-PIV package.
GateSMU	char *	The SMU used for the Gate. This can be SMU1 up to the maximum number of SmUs in the system.
DrainSMU	char *	The SMU used for the Drain. This can be SMU1 up to the maximum number of SmUs in the system. This is the SMU that applies the DC bias to the DUT drain during the sweep.
IdSize VdSize vgsize	int	Set to a value that is at least equal to the number of steps in the sweep and all three must be the same value.

Table 4-21
Outputs for vdsid_pulseiv

Output	Type	Description
IdArray	double *	The measured drain current from channel 2 of the scope card. This current is determined by measuring the voltage drop across the scope card 50 ohm termination, giving Id = Vd/50 ohm.
VdArray	double *	The measured drain voltage from channel 2 of the scope card.
VgMeas	double *	The measured gate voltage from channel 1 of the scope card.

Table 4-22
Return values for vdsid_pulseiv

Value	Description
0	Ok
-1	Invalid value for Vgs
-2	Invalid value for VdStart
-3	Invalid value for VdStop
-4	Invalid value for VdStep
-5	Invalid value for PulseWidth
-6	Invalid value for PulsePeriod
-7	Invalid value for AverageNum
-8	Invalid value for LoadLineCorr
-9	Array sizes do not match
-10	Array sizes not large enough for sweep
-11	Invalid VPUId
-12	Invalid GateSMU
-13	Invalid DrainSMU
-14	Unable to initialize PIV solution

Vdid_Pulse_DC_Family_pulseiv

Description

The VdId_Pulse_DC_Family_pulseiv sweep is used to perform a Pulsed vs DC Vd-Id sweep using the 4200-PIV-A package. This test is similar to a typical Vd-Id but only two sources are used: one for the DUT Gate and one for the DUT Drain. Pulsed Measurements are made with the 2-channel scope, 4200-SCP2.
To create a family of curves, choose an appropriate start and stop value for Vgs, and a number of steps.

This routine can run the sweeps in three different ways: 1) DC only; 2) Pulse only; 3) Pulse and DC curves. This routine supports from one to 10 Vd-Id curves based on up to 10 different Vgs values.

This routine also supports the 4200-PIV-A package using the 4200-RBT. For this package, all test parameters and limits are given below, except the 4200-PIV-A with the 4200-RBT has a max pulse width of 150ns, not the 250ns of the 4205-RBT.

All voltage levels specified below assume a 50Ω DUT load.

Connection

The source and body (well) of the DUT must be shorted together and connected to the common low (outer shield) of the SMA cables on the AC+DC output of the Remote Bias Tees (4205-RBT). The RBT connected to GateSMU (the RBT with the Power Divider) should be connected to the gate. The RBT connected to DrainSMU

should be connected to the drain. Use either G-S-G probes for RF structures, or use DC probes with the 4200-PRB-C adapter cables for DC structures.

Set the appropriate values for the Vds-Id parameters. Inputs, outputs and returned values are provided in Table 4-23, Table 4-24 and Table 4-25.

Table 4-23
Inputs for VdId_Pulse_DC_Family_pulseiv

Input	Type	Description
VgStart	double	The starting step value for Vg. For DC only sweeps, VgStart must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VgStart must be between -5V to +5V.
VgStop	double	The final step value for Vg. For DC only sweeps, VgStop must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VgStop must be between -5V to +5V.
VgNumSteps	double	The number of steps for Vg (Max = 10).
Vg_off	double	The DC bias applied by the GateSMU to put device in the OFF state. Normally set to 0V for enhancement FETs (may be non-zero for depletion FETs).
VdStart	double	The starting sweep value for Vd. For DC only sweeps, VdStart must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT.
VdStop	double	The final sweep value for Vd. For DC only sweeps, VdStop must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT.
VdStep	double	The number of steps for the Vd sweep. (Max = 10000).
PulseWidth	double	The Vgs pulse width (PW). The PW can be 40ns to 250ns (10ns resolution). Pulses wider than 250ns will begin to be attenuated by the coupling capacitor in the Remote Bias Tee (4205-RBT).
PulsePeriod	double	The pulse period for the Vgs pulse. The period can be set from 100µs to 1s (10ns resolution). The period must be set so that the Duty Cycle (DC) is no more than 0.1%. The period is most easily calculated by multiplying the largest desired pulse width (PW) by1000. Example: PW = 150ns, so Period = 150µs.
AverageNum	int	The number of pulses to average at each step of the sweep. For best low signal performance, set AverageNum = 0 for Adaptive Filtering.
GateScpRange	double	The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.050, 0.1, 0.2, 0.5, 1, 2, 5, 10. Note that these ranges are Vpp. For example, the 0.5 range covers -250 to +250mV.
DrainScpRange	double	The voltage measure range for the scope channel measuring the Drain. Use 0 for scope autoranging, or specify a voltage value for a fixed range, where V = I * 50 ohm. Valid voltages are 0.050, 0.1, 0.2, 0.5, 1, 2, 5, 10. Note that these ranges are Vpp. For example, the 0.5 range covers -250 to +250mV.

Table 4-23 (cont.)
Inputs for Vdld_Pulse_DC_Family_pulseiv

Input	Type	Description
GateSMURange	int	The current measurement range to be used for the SMU on the DUT Gate terminal. Values correspond to the table below. Note that Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents. 1 Full Auto 2 Limited Auto 10pA 3 Limited Auto 100pA 4 Limited Auto 1nA 5 Limited Auto 10nA 6 Limited Auto 100nA 7 Limited Auto 1uA 8 Limited Auto 10uA 9 Limited Auto 100uA 10 Limited Auto 1mA 11 Limited Auto 10mA 12 Limited Auto 100mA
DrainSMURange	int	The current measurement range to be used for the SMU on the DUT Drain terminal. Values correspond to the table below. Note that Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents. 1 Full Auto 2 Limited Auto 10pA 3 Limited Auto 100pA 4 Limited Auto 1nA 5 Limited Auto 10nA 6 Limited Auto 100nA 7 Limited Auto 1uA 8 Limited Auto 10uA 9 Limited Auto 100uA 10 Limited Auto 1mA 11 Limited Auto 10mA 12 Limited Auto 100mA
LoadLineCorr	int	Determines whether to use load line correction to compensate for the voltage drop caused by the 50ohm sense resistor used to measure the drain current (Id). 1 = load line correction active. 0 = no load line correction.
GateCompliance	double	The SMU current compliance for the DUT Gate.
DrainCompliance	double	The SMU current compliance for the DUT Drain.
NPLC	double	The DC measurement integration time in NPLC (Number of Power Line cycles).
DCSweepDelay	double	Time, in seconds, between the DC source and measure for each sweep point.
DC_vs_Pulse	int	Determines whether to run a DC and Pulse test or a DC only test or a Pulse only test. 0 - Pulse Only, 1 - DC Only, 2 - DC and Pulse.
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with a 4200-PIV package.
GateSMU	char *	The SMU used for the Gate. This can be SMU1 up to the maximum number of SMUs in the system.

Table 4-23 (cont.)
Inputs for Vdld_Pulse_DC_Family_pulseiv

Input	Type	Description
DrainSMU	char *	The SMU used for the Drain. This can be SMU1 up to the maximum number of SMUs in the system. This is the SMU that applies the DC bias to the DUT drain during the sweep.
DrainVMeas_DC_X_Size (int) DrainVProg_DC_X_Size DrainI_DC_X_Size GateVMeas_DC_X_Size GateVProg_DC_X_Size DrainVMeas_Pulse_X_Size DrainVProg_Pulse_X_Size DrainI_Pulse_X_Size GateVMeas_Pulse_X_Size GateIProg_Pulse_X_Size	int	Sizes of the output arrays. Note that all arrays should be the same size and need to be large enough to hold all sweep points.

Table 4-24
Outputs for Vdid_Pulse_DC_Family_pulseiv

Output	Type	Description
DrainVProg_DC_X DrainVProg_Pulse_X	double	Array of programmed drain voltage values.
DrainVMeas_DC_X DrainVMeas_Pulse_X	double	Array of measured drain voltage values.
DrainI_DC_X DrainI_Pulse_X	double	Array of measured drain currents.
GateVMeas_DC_X GateVMeas_Pulse_X	double	Array of measured gate voltages.
GateVProg_DC_X GateVProg_Pulse_X	double	Array of programmed gate voltages.

Table 4-25
Return values for Vdid_Pulse_DC_Family_pulseiv

Value	Description
0	Ok
-1	Invalid value for Vgs
-2	Invalid value for VdStart
-3	Invalid value for VdStop
-4	Invalid value for VdStep
-5	Invalid value for PulseWidth
-6	Invalid value for PulsePeriod
-7	Invalid value for AverageNum
-8	Invalid value for LoadLineCorr

Table 4-25
Return values for Vdid_Pulse_DC_Family_pulseiv

Value	Description
-9	Array sizes do not match
-10	Array sizes not large enough for sweep
-11	Invalid VPUId
-12	Invalid GateSMU
-13	Invalid DrainSMU
-14	Unable to initialize PIV solution
-15	Invalid GateSMU Range
-16	Invalid DrainSMU Range

Vgid_DC_Pulse_pulseiv

Description The VgId_DC _Pulse pulseiv sweep is used to perform a Pulsed vs DC Vg-Id sweep using the 4200-PIV-A package. This test is similar to a typical Vg-Id but only two sources are used: one for the DUT Gate and one for the DUT Drain. Pulsed Measurements are made with the 2-channel scope, 4200-SCP2.

This routine can run the sweeps in three different ways: 1) DC only; 2) Pulse only; 3) Pulse and DC curves. This routine supports from one to 10 Vd-Id curves based on up to 10 different Vgs values.

This routine also supports the 4200-PIV-A package using the 4200-RBT. For this package, all test parameters and limits are given below, except the 4200-PIV-A with the 4200-RBT has a max pulse width of 150ns, not the 250ns of the 4205-RBT.

All voltage levels specified below assume a 50Ω DUT load.

Connection The source and body (well) of the DUT must be shorted together and connected to the common low (outer shield) of the SMA cables on the AC+DC output of the Remote Bias Tees (4205-RBT). The RBT connected to GateSMU (the RBT with the Power Divider) should be connected to the gate. The RBT connected to DrainSMU should be connected to the drain. Use either G-S-G probes for RF structures, or use DC probes with the 4200-PRB-C adapter cables for DC structures.

Set the appropriate values for the Vds-Id parameters. Inputs, outputs and returned values are provided in Table 4-26, Table 4-27 and Table 4-28.

Table 4-26
Inputs for Vgid_DC_Pulse_pulseiv

Input	Type	Description
Vds	double	The voltage value for Vd. For DC only sweeps, Vds must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, Vds must be between -5V to +5V.
VgStart	double	The starting step value for Vg. For DC only sweeps, VgStart must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VgStart must be between -5V to +5V.

Table 4-26 (cont.)
Inputs for Vgid_DC_Pulse_pulseiv

Input	Type	Description
VgStop	double	The final step value for Vg. For DC only sweeps, VgStop must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VgStop must be between -5V to +5V.
VgStep	double	The sweep step size for the Vg sweep, output by channel 1 of the 4200-PG2 (VPUID).
Vg_off	double	The DC bias applied by the GateSMU to put device in the OFF state. Normally set to 0V for enhancement FETs (may be non-zero for depletion FETs). Note that this package does not support a similar capability for the drain. For full quiescent, or bias, point testing, review the 4200-PIV-Q specs.
PulseWidth	double	The Vgs pulse width (PW). The PW can be 40ns to 250ns (10ns resolution). Pulses wider than 250ns will begin to be attenuated by the coupling capacitor in the Remote Bias Tee (4205-RBT).
PulsePeriod	double	The pulse period for the Vgs pulse. The period can be set from 100µs to 1s (10ns resolution). The period must be set so that the Duty Cycle (DC) is no more than 0.1%. The period is most easily calculated by multiplying the largest desired pulse width (PW) by1000. Example: PW = 150ns, so Period = 150µs.
AverageNum	int	The number of pulses to average at each step of the sweep. For best low signal performance, set AverageNum = 0 for Adaptive Filtering.
GateScpRange	double	The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.050, 0.1, 0.2, 0.5, 1, 2, 5, 10. Note that these ranges are Vpp. For example, the 0.5 range covers -250 to +250mV.
DrainScpRange	double	The voltage measure range for the scope channel measuring the Drain. Use 0 for scope autoranging, or specify a voltage value for a fixed range, where V = I * 50 ohm. Valid voltages are 0.050, 0.1, 0.2, 0.5, 1, 2, 5, 10. Note that these ranges are Vpp. For example, the 0.5 range covers -250 to +250mV.
GateSMURange	int	The current measurement range to be used for the SMU on the DUT Gate terminal. Values correspond to the table below. Note that Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents. 1 Full Auto 2 Limited Auto 10pA 3 Limited Auto 100pA 4 Limited Auto 1nA 5 Limited Auto 10nA 6 Limited Auto 100nA 7 Limited Auto 1uA 8 Limited Auto 10uA 9 Limited Auto 100uA 10 Limited Auto 1mA 11 Limited Auto 10mA 12 Limited Auto 100mA

Table 4-26 (cont.)
Inputs for Vgid_DC_Pulse_pulseiv

Input	Type	Description
DrainSMURange	int	The current measurement range to be used for the SMU on the DUT Drain terminal. Values correspond to the table below. Note that Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents. 1 Full Auto 2 Limited Auto 10pA 3 Limited Auto 100pA 4 Limited Auto 1nA 5 Limited Auto 10nA 6 Limited Auto 100nA 7 Limited Auto 1uA 8 Limited Auto 10uA 9 Limited Auto 100uA 10 Limited Auto 1mA 11 Limited Auto 10mA 12 Limited Auto 100mA
LoadLineCorr	int	Determines whether to use load line correction to compensate for the voltage drop caused by the 50ohm sense resistor used to measure the drain current (Id). 1 = load line correction active. 0 = no load line correction.
GateCompliance	double	The SMU current compliance for the DUT Gate.
DrainCompliance	double	The SMU current compliance for the DUT Drain.
NPLC	double	The DC measurement integration time in NPLC (Number of Power Line cycles).
DCSweepDelay	double	Time, in seconds, between the DC source and measure for each sweep point.
DC_vs_Pulse	int	Determines whether to run a DC and Pulse test or a DC only test or a Pulse only test. 0 - Pulse Only, 1 - DC Only, 2 - DC and Pulse.
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with a 4200-PIV package.
GateSMU	char *	The SMU used for the Gate. This can be SMU1 up to the maximum number of SMUs in the system.
DrainSMU	char *	The SMU used for the Drain. This can be SMU1 up to the maximum number of SMUs in the system. This is the SMU that applies the DC bias to the DUT drain during the pulse or DC sweep.
DrainV_DC_Size DrainVMeas_DC_X DrainVProg_Pulse_X GateVMeas_DC_X GateVProg_Pulse_X DrainISize_Pulse DrainVMeasSize_Pulse DrainVProgSize_Pulse GateVMeasSize_Pulse GateVProgSize_Pulse	int	These values *must* be set to a value that is at least equal to the number of steps in the sweep and all values must be the same.

Table 4-27
Outputs for Vgid_DC_Pulse_pulseiv

Output	Type	Description
DrainI_DC/Pulse	double	The measured drain current from channel 2 of the 4200-SCP2 or the DrainSMU. In the case of Pulse, this current is determined by measuring the voltage drop across the 4200-SCP2 50 ohm termination, giving Id = Vd/50 ohm.
DrainVMeas_DC/Pulse	double	The measured drain voltage from channel 2 of the 4200-SCP2 in the case of pulse and the measured voltage on the DrainSMU in the case of DC.
GateVMeas_DC/Pulse	double	The measure gate voltage from channel 1 of the 4200-SCP2 in the case of pulse and the measured voltage on the GateSMU in the case of DC.
GateVProg_DC/Pulse	double	The programmed gate voltage, either supplied by the PG2 or Gate SMU.

Table 4-28
Return values for Vgid_DC_Pulse_pulseiv

Value	Description
0	Ok
-1	Invalid value for Vds
-2	Invalid value for VgStart
-3	Invalid value for VgStop
-4	Invalid value for VgStep
-5	Invalid value for PulseWidth
-6	Invalid value for PulsePeriod
-7	Invalid value for AverageNum
-8	Invalid value for LoadLineCorr
-9	Array sizes do not match
-10	Array sizes not large enough for sweep
-11	Invalid VPUId
-12	Invalid GateSMU
-13	Invalid DrainSMU
-14	Unable to initialize PIV solution
-15	Invalid GateSMU Range
-16	Invalid DrainSMU Range

vgsid_pulseiv

Description

The vgsid_pulse sweep is used to perform a pulsed Vg-Ig sweep using the 4200-PIV package. This test is similar to a typical DC Vg-Id but only 2 sources are used: gate (VPUID pulse channel 1) and drain (DrainSMU). The gate is pulsed, but the drain is DC biased.

Measurements are made with the 2 channel scope card. Set the appropriate values for the Vgs-Id parameters (Table 4-29). Table 4-30 and Table 4-31 contain outputs and return values, respectively.

Connection

The source and body (well) of the DUT must be shorted together and connected to the common low (outer shield) of the SMA cables on the AC+DC output of the 4200 Remote Bias Tees (RBT). The RBT connected to GateSMU (with the Power Divider) should be connected to the gate. The RBT connected to DrainSMU should be connected to the drain. For detailed connection information, refer to the .

Table 4-29
Inputs for vgsid_pulseiv

Input	Type	Description
Vds	double	The drain-source voltage, output by the DrainSMU (defined below).
Vg_off	double	The DC bias applied by the GateSMU to put device in the OFF state. Normally set to 0V for enhancement FETs (may be non-zero for depletion FETs).
VgStart	double	The starting sweep value for Vg, output by channel 1 of the pulse generator card (VPUID).
VgStop	double	The final sweep value for Vg, output by channel 1 of the pulse generator card (VPUID).
VgStep	double	The sweep step size for the Vg sweep, output by channel 1 of the pulse generator card (VPUID).
PulseWidth	double	The Vgs pulse width (PW). The PW can be 40ns to 150ns (10ns resolution). Pulses wider than 150ns will begin to be attenuated by the capacitor in the 4200 Remote Bias Tee.
PulsePeriod	double	The pulse period for the Vgs pulse. The period can be set from 100µs to 1s (10ns resolution). The period must be set so that the Duty Cycle (DC) is no more than 0.1%. The period is most easily calculated by multiplying the largest desired pulse width (PW) by1000. Example: PW = 150 ns, so Period = 150 us.
AverageNum	int	The number of pulses to average at each step of the sweep. For best low signal performance, set AverageNum = 0 for Adaptive Filtering.
GateRange	double	The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.050, 0.1, 0.2, 0.5, 1, 2, 5, 10.
DrainRange	double	The voltage measure range for the scope channel measuring the Drain. Use 0 for scope autoranging, or specify a voltage value for a fixed range, where V = I * 50 ohm. Valid voltages are 0.050, 0.1, 0.2, 0.5, 1, 2, 5, 10.
LoadLineCorr	int	Determines whether to use load line correction to compensate for the voltage drop caused by the 50 ohm sense resistor used to measure the drain current (Id). 1 = load line correction active. 0 = no load line correction.
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with the 4200-PIV package.
GateSMU	char *	The SMU used for the Gate. This can be SMU1 up to the maximum number of SmUs in the system.
DrainSMU	char *	The SMU used for the Drain. This can be SMU1 up to the maximum number of SmUs in the system. This is the SMU that applies the DC bias to the DUT drain during the sweep.
IdSize VdSize vgsize	int	Set to a value that is at least equal to the number of steps in the sweep and all three must be the same value.

Table 4-30
Outputs for vgsid_pulseiv

Output	Type	Description
IdArray	double *	The measured drain current from channel 2 of the scope card. This current is determined by measuring the voltage drop across the scope card 50 ohm termination, giving Id = Vd/50 ohm.

Table 4-30
Outputs for vgsid_pulseiv

Output	Type	Description
VgArray	double *	The measured gate voltage from channel 2 of the scope card.
VdArray	double *	The measured drain voltage from channel 1 of the scope card.

Table 4-31
Return values for vgsid_pulseiv

Value	Description
0	Ok
-1	Invalid value for Vds
-2	Invalid value for VgStart
-3	Invalid value for VgStop
-4	Invalid value for VgStep
-5	Invalid value for PulseWidth
-6	Invalid value for PulsePeriod
-7	Invalid value for AverageNum
-8	Invalid value for LoadLineCorr
-9	Array sizes do not match
-10	Array sizes not large enough for sweep
-11	Invalid VPUId
-12	Invalid GateSMU
-13	Invalid DrainSMU
-14	Unable to initialize PIV solution

scopeshot_cal_pulseiv

Description The scopeshot_cal_pulseiv routine is used to display a single Pulse IV scopeshot_pulseiv. This routine is useful to understand the basic source and measure concepts behind the Pulse IV methods for pulse vds-id and vgs-id. Measurements are made with cable compensation values applied to them and load line compensation can be used if desired.

Connection The source and body (well) of the DUT must be shorted together and connected to the common low (outer shield) of the SMA cables on the AC+DC output of the Remote Bias Tees (RBT). The RBT connected to GateSMU (with the Power Divider) should be connected to the gate. The RBT connected to DrainSMU should be connected to the drain.

Set the appropriate values for the scopeshot_cal_pulseiv. Set the appropriate values for the Vds-Id parameters. Inputs, outputs and returned values are provided in Table 4-32, Table 4-33 and Table 4-34.

Table 4-32
Inputs for scopeshot_cal_pulseiv

Input	Type	Description
Vds	double	The DC drain bias, provided by the drainSMU.
Vgs	double	The pulse gate voltage amplitude. This can be set from -5 to +5 V.
VgStart	double	The starting sweep value for Vg, output by channel 1 of the pulse generator card (VPUID).
PulseWidth	double	The Vgs pulse width (PW). The PW can be 40ns to 300ns (10ns resolution).
PulsePeriod	double	The pulse period for the Vgs pulse. The period can be set from 100us to 1s (10ns resolution). The period must be set so that the Duty Cycle (DC) is no more than 0.1%. This period is most easily calculated by multiplying the largest desired pulse width (PW) by 1000. Example: PW = 150ns, so Period = 150us.
AverageNum	int	The number of pulses to average at each step of the sweep. For best low signal performance, set AverageNum = 0 for Adaptive Filtering.
GateRange	double	The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. Note that the range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).
LoadLineCorr	int	Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Drain. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with the 4200-PIV package.
GateSMU	char *	The SMU used for the Gate. This can be SMU1 up to the maximum number of SmUs in the system.
DrainSMU	char *	The SMU used for the Drain. This can be SMU1 up to the maximum number of SmUs in the system. This is the SMU that applies the DC bias to the DUT drain during the sweep.
TimeSize	int	These values must be set to a GatePulseSize value that is at least equal to the DrainPulseSize number of steps in the sweep and all three must be the same value.

Table 4-33
Outputs for scopeshot_cal_pulseiv

Output	Type	Description
Time	double *	Array of time values from the 4200-SCP2 scope (s).
GatePulse	double *	Array of gate pulse voltages from channel 1 of the 4200-SCP2 scope.
DrainPulse	double *	Array of drain voltages from channel 2 of the 4200-SCP2 scope.
GateV	double *	Measured Gate Voltage.
DrainV	double *	Measured Drain Voltage.
GateI	double *	Measured Gate Current.

Table 4-34
Return values for scopeshot_cal_pulseiv

Value	Description
0	Ok
-1	Invalid Gate Voltage (Max 5V).
-2	Invalid Drain Voltage (Max 210V).
-5	Invalid Pulse Width (Min 40nS).
-6	Invalid Pulse Period (Min 40ns).
-7	Invalid Average Num (must be between 1 and 100000).
-8	Invalid LoadLineCorr (must be between 0 and 1).
-9	Time, GatePulse, and Drain Pulse array sizes must be equal.
-11	Invalid VPU. Specified VPU is not in current system configuration.
-12	Invalid GateSMU. Specified SMU is not in current system configuration.
-13	Invalid DrainSMU. Specified SMU is not in current system configuration.
-14	PIV Initialization Failed.

scopeshot_pulseiv

Description The scopeshot_pulseiv routine displays a single Pulse IV scopeshot. This routine is useful to understand the basic source and measure concepts behind the Pulse IV methods for pulse vds-id and vgs-id. The scope waveforms are retrieved and displayed for both channels (no measurements are made). Make sure to set the appropriate values for the scopeshot_pulseiv (Table 4-35). Table 4-36 and Table 4-37 contain outputs and return values, respectively.

Connection The source and body (well) of the DUT must be shorted together and connected to the common low (outer shield) of the SMA cables on the AC+DC output of the 4200 Remote Bias Tees (RBT). The RBT connected to GateSMU (with the Power Divider) should be connected to the gate. The RBT connected to DrainSMU should be connected to the drain. For detailed connection information, refer to the .

Table 4-35
Inputs for scopeshot_pulseiv

Input	Type	Description
RiseTime	double	The gate pulse transition rise time (s). This can be set from 10e-9 to 300e-9 in 10e-9 (10ns) steps. Note that this value programs the full transition time (0–100%), not the 10–90% time.
FallTime	double	The gate pulse transition fall time (s). This can be set from 10e-9 to 300e-9 in 10e-9 (10ns) steps. Note that this value programs the full transition time (0–100%), not the 10–90% time.
PulseWidth	double	The gate pulse width (PW). The PW can be 20ns to 1us (10ns resolution). Pulses wider than 150ns will begin to be attenuated by the capacitor in the 4200 Remote Bias Tee.
PulseBase	double	The pulse gate base voltage. This can be set from -5 to +5 V, inclusive of amplitude.
PulseAmplitude	double	The pulse gate voltage amplitude. This can be set from -5 to +5 V, inclusive of base voltage.
GateLoad	double	The scope card channel 1 input impedance for the gate. Either 50 or 1E6. Use 50 for Pulse IV with RBTs.
GateRange	double	The scope card channel 1 Y scale voltage range for the gate measurement. Typical values are 1, 2, 5 V.
DrainLoad	double	The scope card channel 2 input impedance for the drain. Either 50 or 1E6. Use 50 for Pulse IV with RBTs.
DrainRange	double	The scope card channel 2 Y scale voltage range for the drain measurement. Typical values are 1, 2, 5 V.
PulsePeriod	double	The pulse period for the Vgs pulse. The period can be set from 40ns to 1s (10ns resolution). The period must be set so that the Duty Cycle (DC) is no more than 0.1%. This period is most easily calculated by multiplying the largest desired pulse width (PW) by 1000. Example: PW = 150ns, so Period = 150us.
AverageNum	int	The number of waveforms to average.
GateBias	double	The DC gate bias, provided by the gateSMU.
DrainBias	double	The DC drain bias, provided by the drainSMU.
VPUID	char *	The instrument ID. This should be set to VPU1 for 4200 systems with the 4200-PIV package.
GateSMU	char *	The SMU used for the Gate. This can be SMU1 up to the maximum number of SmUs in the system.
DrainSMU	char *	The SMU used for the Drain. This can be SMU1 up to the maximum number of SMUs in the system. This is the SMU that applies the DC bias to the DUT drain during the sweep.
TimeSize Ch1OutSize Ch2OutSize	int	Set to a value that is at least equal to the number of steps in the sweep and all three must be the same value.

Table 4-36
Outputs for scopeshot_pulseiv

Output	Type	Description
Time	double *	Array of time values from the scope card (s).
Ch1Out	double *	Array of gate voltages from channel 1 of the scope card.
Ch2Out	double *	Array of drain voltages from channel 2 of the scope card.

Table 4-37
Return values for scopeshot_pulseiv

Value	Description
0	Ok
-1	Invalid Pulse Width (Min 40ns)
-2	Invalid Pulse Period (Min 40ns)
-3	Invalid Average Num (1 - 1000)
-4	Array Sizes Do Not Match
-5	Invalid VPU. Specified VPU Is Not In Current System Configuration
-6	Invalid GateSMU. Specified SMU Is Not In Current System Configuration
-7	Invalid DrainSMU. Specified SMU Is Not In Current System Configuration
Negative numbers are errors—refer to LPT and PulseIV documentation for description.	

vdsid_pulseiv_demo

(Also see)

vgsid_pulseiv_demo

(Also see)

scopeshot_pulseiv_demo

(Also see)

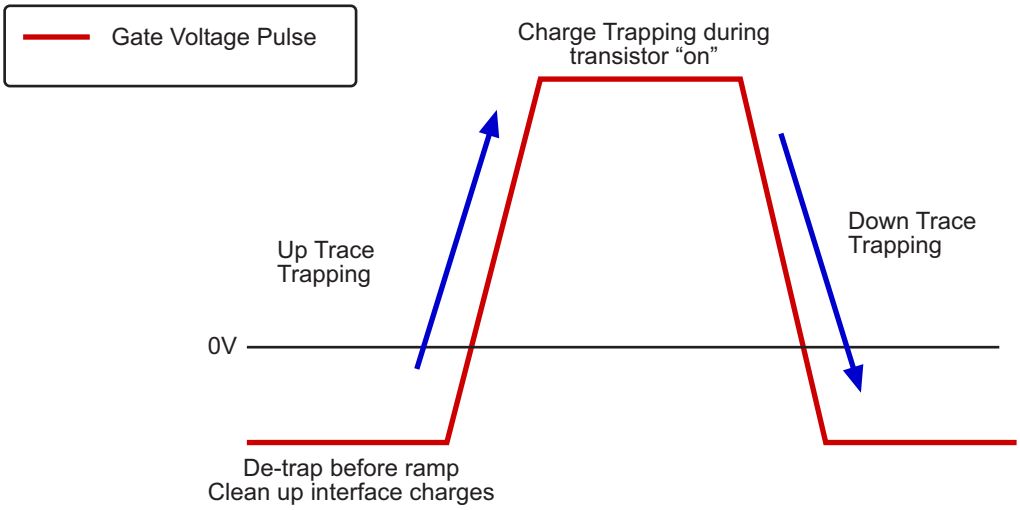
These three UTMs are functionally identical but simpler than their respective routines listed earlier in this section of the manual. The difference being less-used parameters have been eliminated from the parameter list and hard-coded (e.g., SMU channels, ranges, load line).

Slow single pulse charge trapping high K gate stack

The key to using the single pulse method is to look at the charge trapping and de-trapping behavior within a single, well-configured gate pulse (Figure 4-29). The gate pulse usually starts in a position that discharges the gate capacitor before the voltage ramp begins. This is to clean up any residual charges that might be trapped in the gate. Then, during the rise time of the voltage ramp, the corresponding drain current response is captured, allowing a Vgs-Id curve to be formed. Slow single pulse refers to rise and fall transition times of 100ns minimum, with a pulse width of at least 1μs. These relatively slow pulse parameters mean that the RBT are not used and a simple splitter can be used for monitoring the drain current pulse.

For each measurement, a pulse is applied to the gate of the transistor while its drain is biased at a certain voltage. The change in drain current, resulting from the gate pulse, appears on the digital oscilloscope.

Figure 4-29
Trapping and de-trapping in a single gate voltage pulse



Charge trapping procedure

1. Perform cable correction (open and through, if necessary), with calibration substrate. Open and through correction measurements are taken and inputted into correction algorithm to calculate cable losses.
2. Connect DUT (transistor) as shown in Figure 4-30 and Figure 4-31.
3. Input test parameters, refer to key parameters contained in Table 4-38.
4. The UTM will pulse the gate with single pulse (for average >1 use a series of very low duty cycle pulses), bias drain with a PG2, capture drain current response on oscilloscope, then calculate corresponding drain current (Vgs-Id) from the whole waveform.
5. To ensure a determinate number of pulses are applied to the DUT, the period must be set to >10 ms. Wider pulse widths require a longer period. If the period is too short, pulse(s) will not be measured and will cause the UTM to hang, requiring KITE to be manually halted.

Figure 4-30
Slow single pulse—hardware setup block diagram

NOTE This configuration can handle pulse widths ≥ 100ns which is too wide to use Remote Bias Tees.

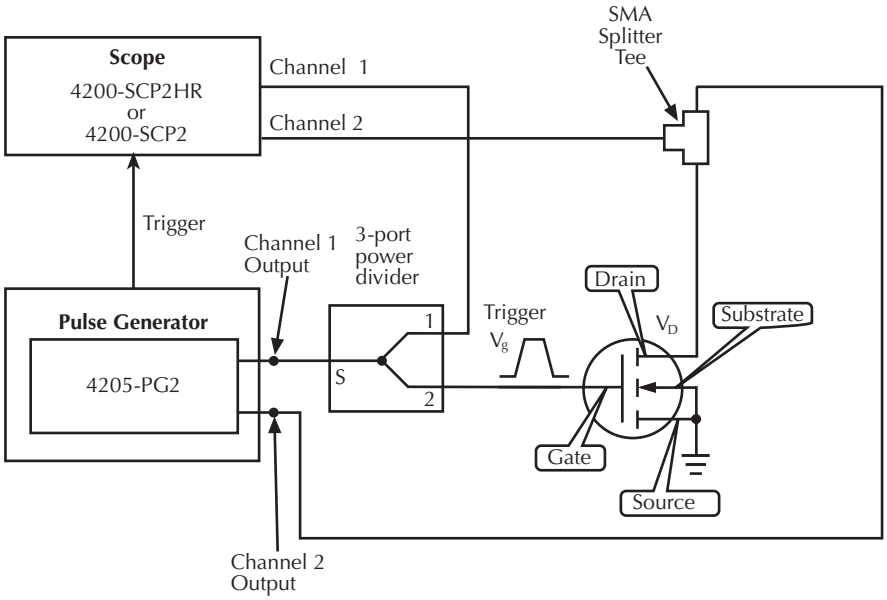


Table 4-38
Key parameters—Slow Single Pulse Charge Trapping

Parameters	Range/Specification
Application	Pulse I-V like application
Rise / Fall time	Variable 10ns–10μs
Pulse width	5μs–1ms single pulse
Pulse amplitude	0-5V
Base voltage	+/- 5V
Load impedance	50Ω or 1MΩ

Figure 4-31
Slow single pulse—hardware connection

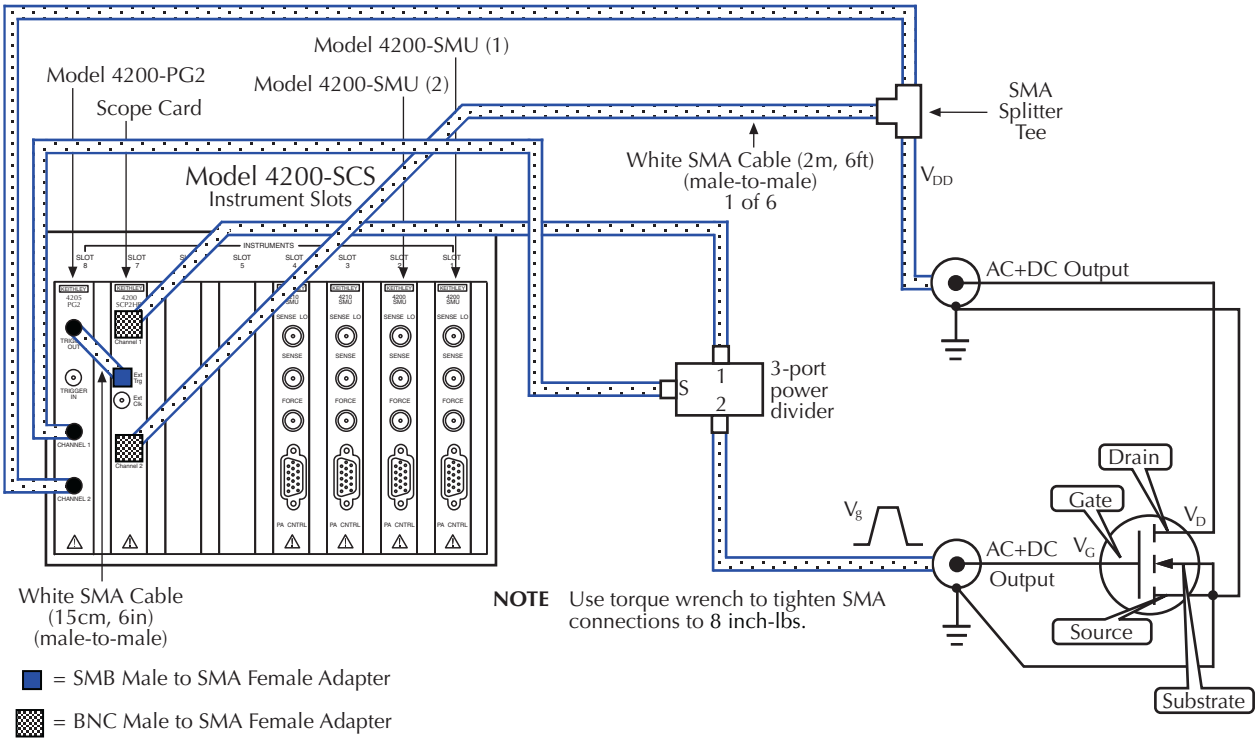


Figure 4-32
Example slow single pulse waveform graph

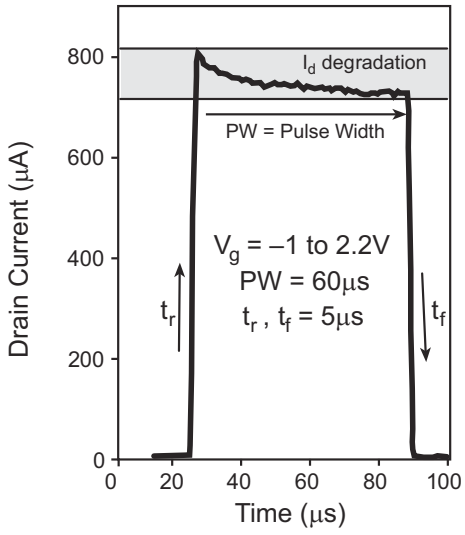
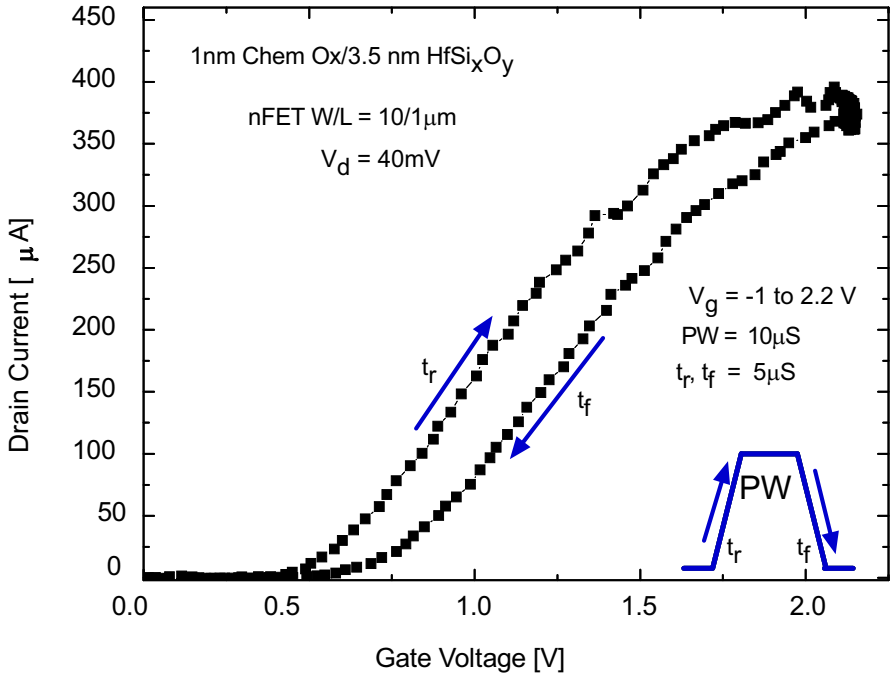


Figure 4-33
Single slow pulse example data plot



Charge Trapping UTM descriptions

The charge trapping user library contains modules required for single pulse measurement method.

The module contained in the charge trapping user library is listed in Table 4-39 with detailed information following the table.

Table 4-39
Charge pumping UTMs

User Module	Description
chargetrapping_single_pulse_slow	Provides a Id degradation vs time plot on high K transistors.

chargetrapping_single_pulse_slow

Description The slow single pulse charge trapping UTM provides a Id degradation vs time plot on high K transistors. This single pulse method is used to investigate the charge trapping and de-trapping behavior of high-k gate structures. This method is called "slow" because it is designed to be used with wider pulse widths (> =100 ns) than the 4200 Remote Bias Tees can transmit. The rise and fall times, along with the pulse base and amplitude, are varied to determine the behavior of the various traps. This method is useful only for the linear region of the transistor (Vd approximately 100 mV).
Set the appropriate values for the single pulse charge trapping test parameters, ensuring that a Vd is used that keeps the transistor in the linear region (Table 4-40). Table 4-41 and Table 4-42 contain outputs and return values, respectively.

Connection The source and body (well) of the device under test (DUT) must be shorted together and connected to the common low (outer shield) of the SMA cables from the instrument.

Gate connection: Channel 1 of the pulse generator card should be connected first to the power divider and then on to the gate/source of the DUT. The other power divider connection goes to channel 1 of the scope card.

Drain connection: Channel 2 from the pulse generator card should be connected first to a splitter tee and then on to the drain/body of the DUT. The other splitter tee connection goes to channel 2 of the scope card. For detailed connection information, refer to the .

Table 4-40
Inputs for chargetrapping_single_pulse_slow

Input	Type	Description
RiseTime	double	Transition rise time for the Vg pulse. This can be set from 10E-9 (10ns) to 1 second with10ns resolution. Note that this is programming the full transition time (0–100%) not 10–90%.
FallTime	double	Transition fall time for the Vg pulse. This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Note that this is programming the full transition time (0–100%) not 10–90%.
PulseWidth	double	The Vgs pulse width (PW). This can be set from 10E-9 (10ns) to 1 second with 10ns resolution. Maximum width: 1/2 rise + 1/2 fall + PW < = 1s
VDischarge	double	The voltage of the base of the Vg pulse. This voltage can be set from -20V to +20 V. This is usually a negative voltage to de-trap or clean out any interface charges.
DischargeTime	double	The time, in seconds, to apply the V_discharge before applying the pulse.
VgPulse	double	Amplitude of Vg pulse (V). This can be set from -40V to +40V (inclusive of v_discharge).
Vd	double	Amplitude of Vd pulse (V). This pulse width is set to be 2x of the Vg PW. This voltage can be set from -80V to +80V
PulsePeriod	double	The pulse period for the pulses (s). The period can be set from 10ns to 1s (10ns resolution).
AverageNum	int	The number of pulses to average. This can be set from 1 to 1000.
GateRange	double	The scope card channel 1 Y scale voltage range for the gate measurement. Typical values are 1, 2, 5, 10Vpp.
DrainRange	double	The scope card channel 2 Y scale voltage range for the drain measurement. Typical values are 1, 2.5, 5, 10, 20, 50, and 100 Vpp.
WindowAve	int	The factor used to perform a moving average to smooth the waveform.
VPUId	char *	The pulse generator card ID. This should be set to VPU1 for 4200 systems with the 4200-PIV package.
GateSMU	char *	The SMU attached to the gate.
DrainSMU	char *	The SMU attached to the drain.
Time_size Vg_size Id_size RawTime_size ch1_out_size ch2_out_size	int	Sizes must be set to the same value. The size of these arrays determines the number of readings retrieved from the scope card as well as the scope card sample rate. Using a larger size will use the fastest sample rate of 1.25 GS/s. For a 500ns PW, a size of 1250 will use the 1.25GS/s sample rate. For smaller ratios, the sample rate will be set to a lower value.

Table 4-41
Outputs for chargetrapping_single_pulse_slow

Output	Type	Description
Time	double *	The array of time values after any smoothing is applied (see Points), in seconds.
Vg	double *	The array of gate voltage values that make up the single pulse waveform.
Id	double *	The array of drain current values that make up the single pulse waveform. This current is determined by measuring the voltage drop across the pulse generator card 50 ohm termination on channel 2, giving Id = Vd/50 ohm.
RawTime	double *	The array of time values returned from the scope card.
ch1_out	double *	The array of Vg voltage readings from the scope card channel 1.
ch2_out	double *	The array of Vd voltage readings from the scope card channel 2.

Table 4-42
Return values for chargetrapping_single_pulse_slow

Value	Description
0	OK
-1	Invalid VPU
-2	Invalid Gate SMU
-3	Invalid Drain SMU
All other negative numbers are also errors – refer to LPT documentation for description.	

AC stress for WLR

AC, or pulsed, stress is a useful addition to the typical stress-measure tests for investigating both semiconductor charge trapping and degradation behaviors. NBTI (negative bias temperature instability) and TDDB (time dependent dielectric breakdown) tests consist of stress/measure cycles. The applied stress voltage is typically a DC signal, that is used because it maps more easily to device models. However, incorporating pulsed stress testing provides additional data that permits a better understanding of device performance in frequency-dependent circuits.

NOTE *Key test parameters are contained in Table 4-43.*

1. Connect pulser to DUT during stress as shown in Figure 4-34, Figure 4-35, and Figure 4-36.
2. The test pulse stresses the device for HCI, NBTI and TDDB test instead of DC bias by outputting a train of pulses for a period of time (stress time). Pulse characteristics are not changed during the stress-measure test.
3. The test then measures device characteristics using SMUs: Vth, Gm, and so on.

Figure 4-34
AC Pulse stress-measure—hardware setup block diagram

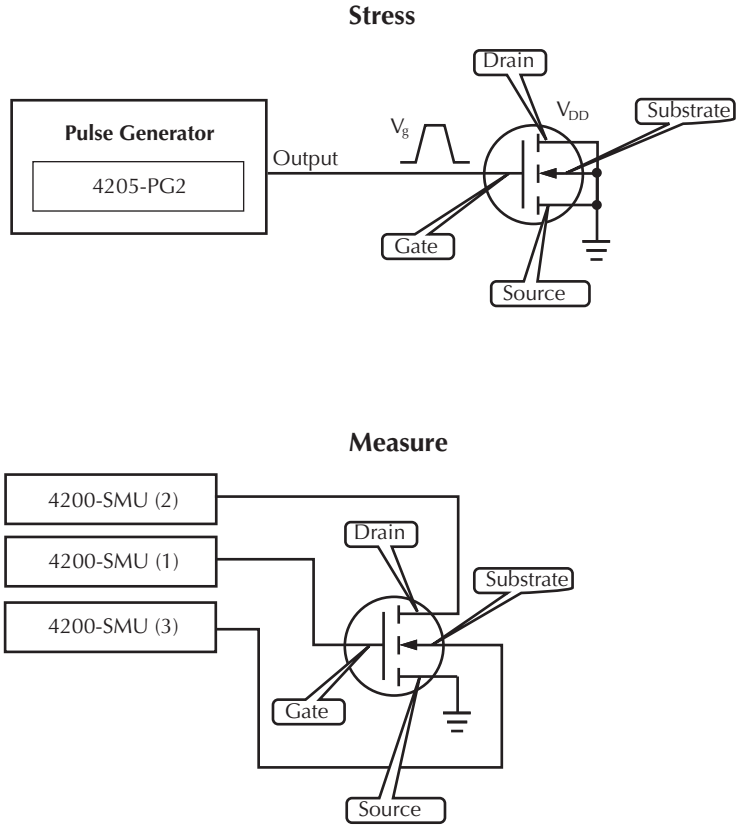


Figure 4-35
AC Pulse stress-measure—hardware matrix card simplified schematic

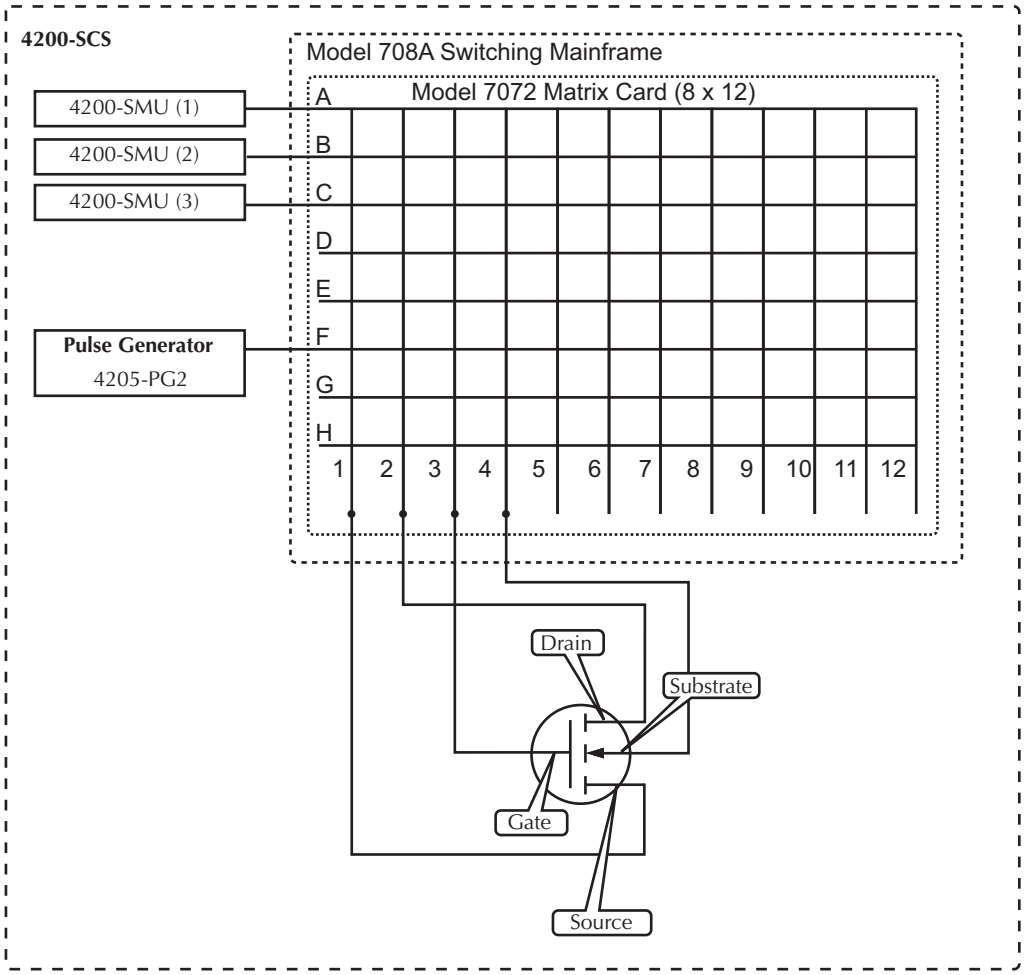


Figure 4-36
AC Pulse stress-measure—hardware connections

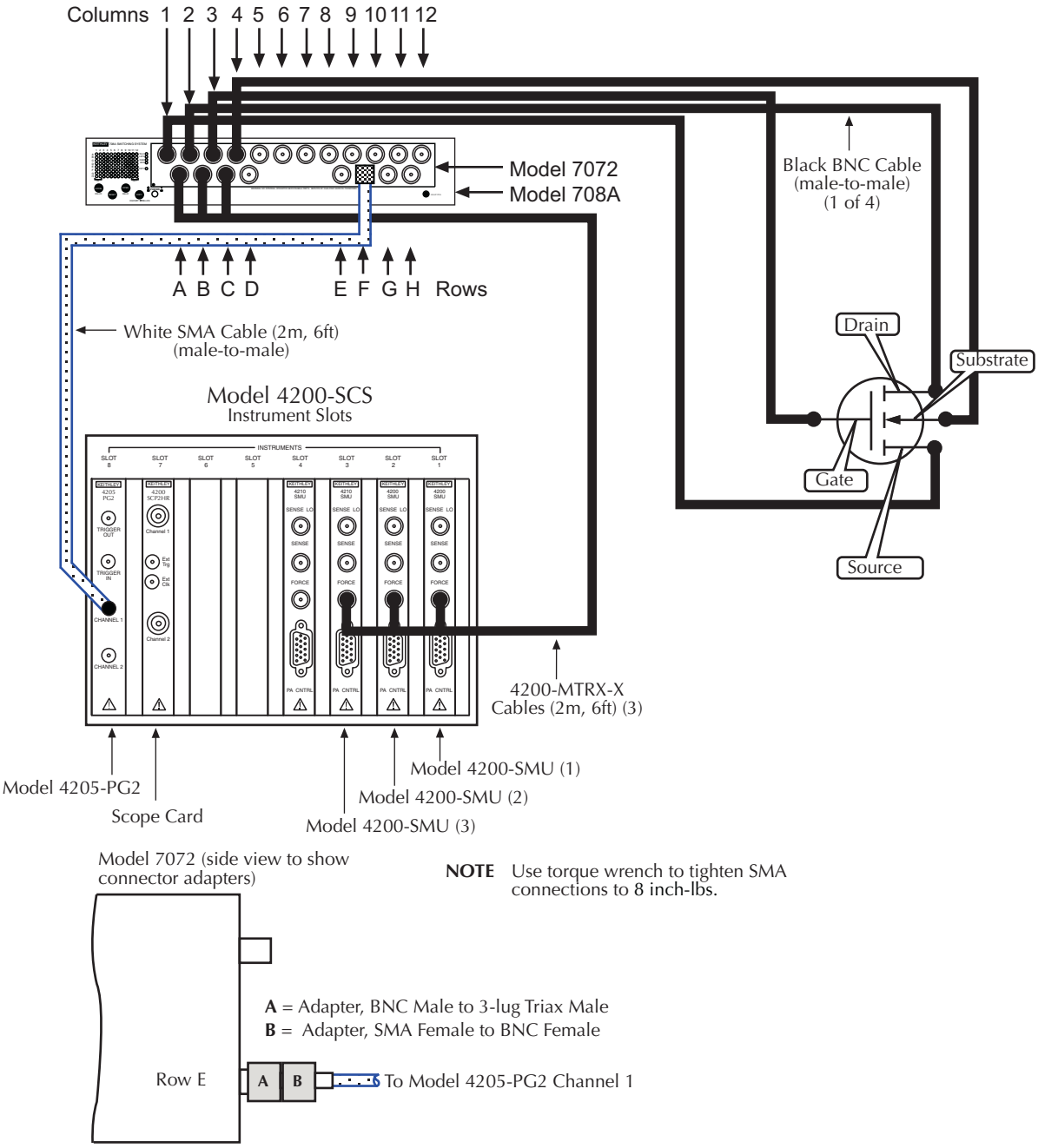


Table 4-43
Key pulse generator parameters—AC stress for WLR

Parameters	Range/Specification
Rise / Fall time	Variable 100ns–500µs
Pulse width	200ns–1ms single pulse
Pulse amplitude	-5 to +5V
Duty cycle	~50%
Base voltage	+/- 5V

Q-Point Pulse IV – Model 4200-PIV-Q

This application provides q-point pulse IV testing for higher power compound semiconductor or LDMOS RF transistors, or any device may benefit from low duty cycle pulse IV testing. This application uses the Model 4200-PIV-Q package, that differs from the Model 4200-PIV-A package by: 1) pulsing both the gate and drain; 2) Providing higher power to the DUT drain; 3) Pulsing from non-zero bias point, or quiescent point, testing. The compound semiconductor transistors typically consist of materials made from the III-V groups in the periodic table of the elements, but other groups or elements may be used.

What is the PIV-Q Package?

The PIV-Q package is an optional factory-installed kit to the Model 4200-SCS. The focus for the PIV package is testing RF FETs that exhibit self-heating or charge trapping effects (also called dispersion).

To accomplish pulse IV testing of LDMOS and compound semiconductor FETs, the PIV-Q package consists of the following:

- Model 4205-PG2 Dual channel voltage pulse generator (Qty: 3)
- Model 4200-SCP2HR Dual channel oscilloscope
- 4205-PCU Pulse Combiner Unit to create one higher power pulse channel for the DUT drain by combining the four pulse channels from two 4205-PG2 cards
- Pulse IV Interconnect adapters and cabling
- Pulse IV software - Projects and test routines for testing of RF FETs, including cable compensation and load-line algorithms to provide DC-like sweep results

Target applications and test projects

The PIV-Q package includes test projects that address the most common parametric transistor tests: Vds-id and Vgs-id. These tests are provided in both DC and Pulse modes, allowing correlation between the two test methods. These tests, as well as initialization steps for scope auto-calibration and cable compensation, are included in a single Model 4200 test project, QPulse-IV-Complete. There are two methods for performing DC IV sweeps. The first methods is to use the SMUs to provide the DC source and measure: Vd-Id-DC, Vg-Id-DC. Because SMU output impedance varies with the source and measure ranges, the Pulse IV instrumentation may also be used to provide DC-like IV sweeps. Use the desired pulse test and set the duty cycle from the typical low values up to 90+% to mimic a DC test by choosing a longer period, up to one second, with appropriate pulse width and transitions.

NOTE *The UTMs used for Q-point Pulse IV tests are described in the following paragraphs. These UTMs control all instrumentation for these applications. The Model 4205-PG2 pulse generator and Model 4200-SCP2HR scope can also be used as stand-alone instruments.*

Section 11 of the Reference Manual explains front panel operation and provides remote programming information for the pulse generator and scope. For remote programming, the Model 4205-PG2 pulse generator uses LPTLib functions (see "Pulse generator settings"), while the Model 4200-SCP2HR scope uses kiscopeulib UTMs (see "kiscopeulib UTM descriptions").

Section 12 of the Reference Manual provides additional information on projects for the PIV-A and PIV-Q packages.

PIV-Q Test Procedure

Perform cable compensation routines (open and through) if necessary, with calibration substrate or structure. Open and through compensation measurements are taken and inputted into an algorithm to account for cable losses. This PulseIVCal procedure should take approximately 15 minutes, and requires the connection change from open to through at the midpoint of the procedure.

Figure 4-37
PIV-Q Block Diagram

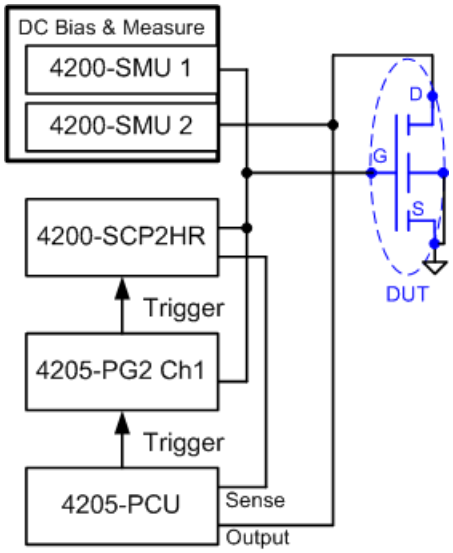
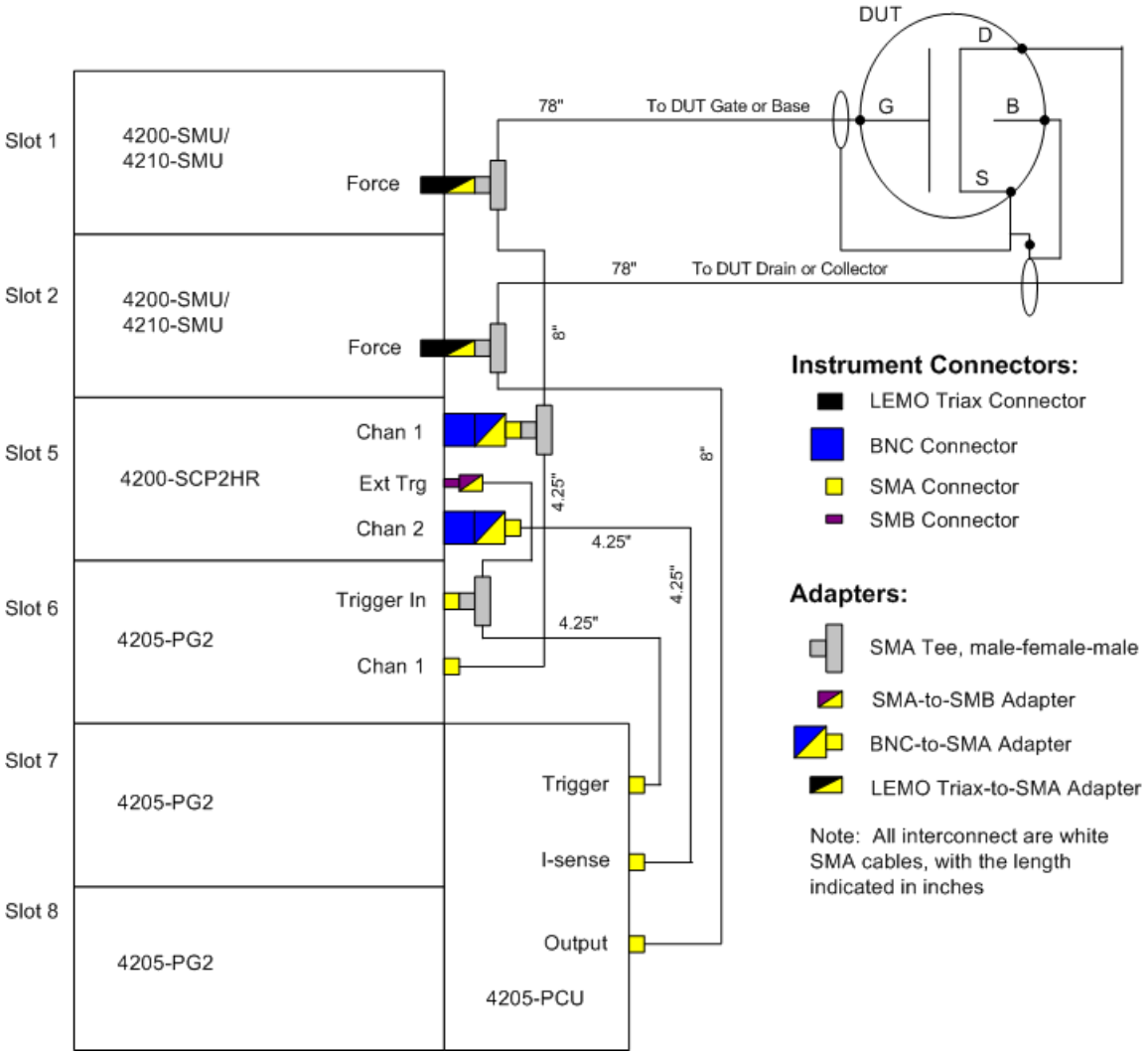


Figure 4-38
PIV-Q connections diagram



Interconnect Assembly Procedure

Table 4-44 lists the interconnect parts included with the PIV-Q package, while Table 4-44 lists the supplied tools.

NOTE *The various adapters, cables and hardware used for the pulse projects are shown in Figure 4-50 (located at the end of this section). The 4205-PCU that is used for the PIV-Q package is also shown.*

Table 4-44
Interconnect parts for PIV-Q

Qty	Description	Comment
4	4.25 in/10.8 cm white SMA cables	Interconnect for triggering and pulse signals
2	8 in/20.3 cm white SMA cables	Interconnect between pulse and DC signals
2	6.6 ft/2 m white SMA cables	Pulse + DC interconnect between instrument and DUT
4	Female-Male-Female SMA Tees	Trigger, Pulse and DC interconnect
2	Female SMA-to-Male BNC	Adapt 4200-SCP2HR BNC to SMA for cables
2	LEMO triax-to-SMA female	Adapt SMU Force output to SMA for signal interconnect
1	Female SMA-to-SMB male	Adapt SMB 4200-SCP2HR Ext Trigger input to SMA
1	4205-PCU, already installed	—

Table 4-45
Tools supplied with the Model 4200-SCS or PIV-Q package

Qty	Description
1	#1 Phillips screwdriver
1	Torque wrench, 8 in/lb, with 5/16 head installed
1	Model 4205-PCU alignment tool

Perform the following steps to connect PIV-Q:

NOTE *Refer to the Figure 4-38 (connection diagram) for the following procedure.*

1. Set up the Model 4200-SCS, referring to the Model 4200-SCS Quick Start Guide and Model 4200-SCS Reference Manual.
2. The two SMUs used for PIV-Q must not have PreAmps installed. To remove PreAmps from SMUs, perform the following steps:
   a. Save all projects, close KITE and all other programs.
   b. Perform a Windows shut down.
   c. Turn off the power supply.
   d. Using the Phillips screwdriver, loosen the screw retaining the two PreAmps.
   e. Remove the two PreAmps.
   f. Turn on the Model 4200-SCS power and wait for Windows to boot.
   g. If KITE is set to Autorun on boot, there will be a beep and an error message concerning the change in configuration. Click OK on this warning dialog.
   h. Double-click the KCON icon on the 4200 desktop
   i. Under Tools, click on "Update PreAmp Configuration"
   j. Once this is finished, close KCON (the new configuration is automatically saved)
   k. Run KITE
   l. When finished using PIV-Q and PreAmps are necessary, simply follow the above steps in reverse: Shutdown > install PreAmps > boot > click OK on warning > run KCON > Tools|Update PreAmp Configuration > quit KCON > run KITE.
3. Optional: It is possible to make all of the connections with the PCU already installed, but removing it first provides increased access to the PG2 card in slot 6, as well as better access to the adjacent SCP2HR scope card in slot 5.
4. To remove the PCU, use the screwdriver to loosen the three screws and the pull the PCU housing straight back from the Model 4200-SCS chassis, and set it aside for later. Make sure to leave the seven SMA-to-MMBX adapters on the two PG2 cards in slots 7 and 8.
5. Connect adapters to the instruments:

a. For the SCP2HR scope card, connect BNC-to-SMA adapters to the Chan 1 and Chan 2 connectors, and connect an SMA-to-SMB adapter to the Ext Trg connector.
b. For the two SMUs, connect a LEMO Triax-to-SMA adapter to each Force connector. Carefully align the Lemo connector with the Force opening, then slide the connector in until it clicks. To remove the Lemo connector, gripe the knurled, or checkered, portion of the sleeve and pull in a direction away from the instrument. There are latches on the Lemo connector. Using excessive force to remove a Lemo connector may cause damage to the connector and/or the SMU.
c. Loosely connect four SMA Tee adapters to the two SMUs, the SCP2HR and the PG2 in slot 6. The photo in Figure 4-39 shows how the SMA Tees should be positioned. Do not tighten the Tee adapters.

6. Cable connections for the two SMUs, SCP2HR and the PG2 in slot 6:
   a. Use eight SMA cables for these instrument connections. Make sure to use the appropriate length cables as indicated in Figure 4-38.
   b. If having trouble making a cable connection to an adapter, the adapter can be temporarily removed. Make sure to re-connect the adapter after connecting the cable.
   c. After verifying that the Tee adapters and cables are positioned as shown in Figure 4-39, tighten all SMA connections using the torque wrench.

NOTE *Perform steps 5 through 7 if the PCU was removed from the Model 4200-SCS chassis. It the PCU is already installed, proceed to step 8.*

7. Ensure that the seven SMA-to-MMBX adapters are still installed on the PG2 cards located in slots 7 and 8. The PCU will be connected to these two PG2 cards.
8. Use the white, plastic alignment tool to pre-align the connectors on the PC board inside of the PCU assembly. Remove the alignment tool.
9. Carefully align the PCU, and push it onto the MMBX adapters; there should be a slight click. Finger tighten the three retained screws on the PCU. Use the screwdriver to finished tightening the screws, but do not use excessive force.
10. Referring to Figure 4-38, connect the three SMA cables to the PCU;
    - Connect the SMA cable from the PG2 in slot 6 to the Trigger connector of the PCU.
    - Connect the SMA cable from the SCP2HR (Ch 2) to the I-sense connector of the PCU.
    - Connect the SMA cable from the SMU in slot 2 to the Output connector of the PCU.
11. Route the two 2yd/2m SMA cables to the prober or DUT fixture. The cable from SMU1 goes to the DUT gate/base and the cable from SMU2 goes to the DUT drain/collector.
12. The SMA cables should be connected to RF G-S-G (ground-signal-ground) probe manipulators and test structures (see Figure 4-40). If using the 8101-PIV test fixture, see Figure 4-41 and Figure 4-42 for connection and proper DUT installation.
13. Finish the setup by verifying connections and running a scope-shot test from the Model 4200 Project QPulse IV-Complete project.

Figure 4-39
Photo showing PIV-Q connections

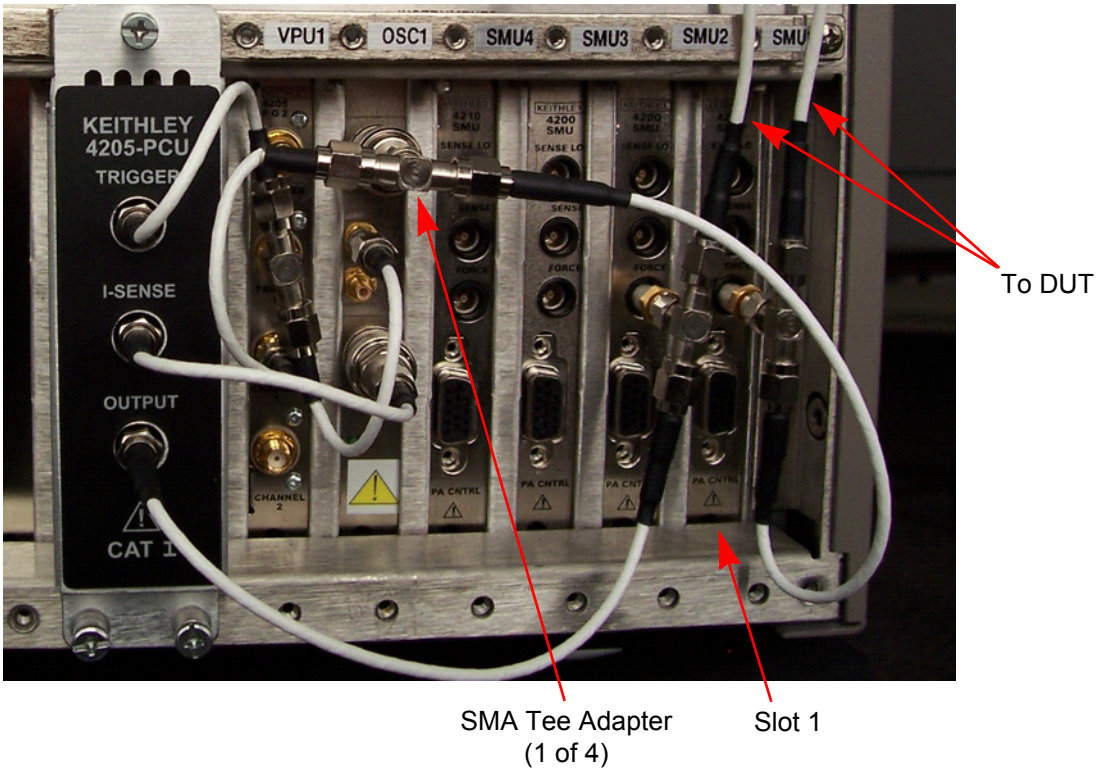


Figure 4-40
PIV-Q connections using RF G-S-G probes

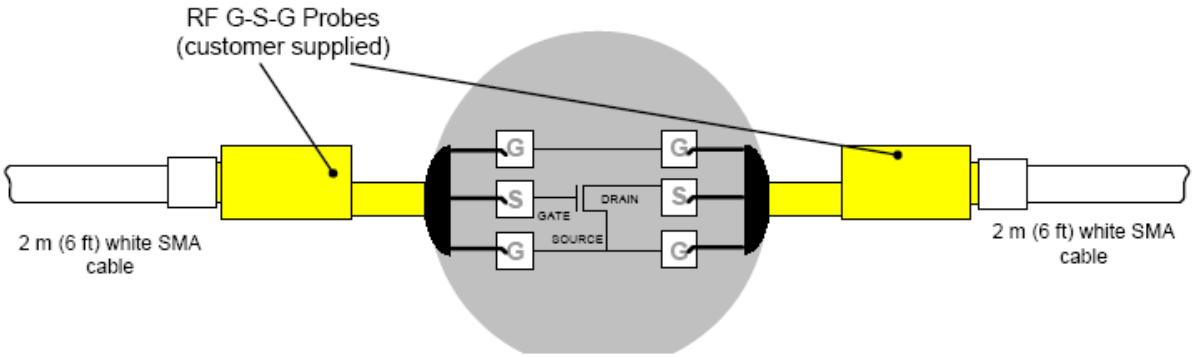


Figure 4-41
PIV-Q connections using the 8101-PIV test fixture

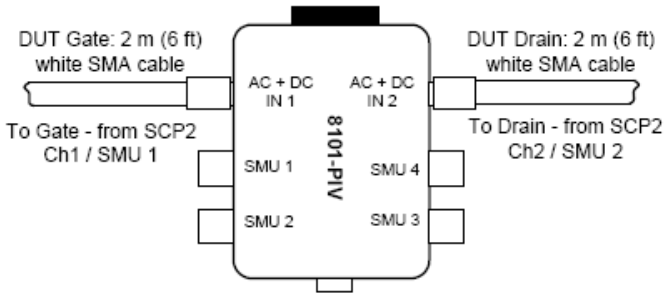
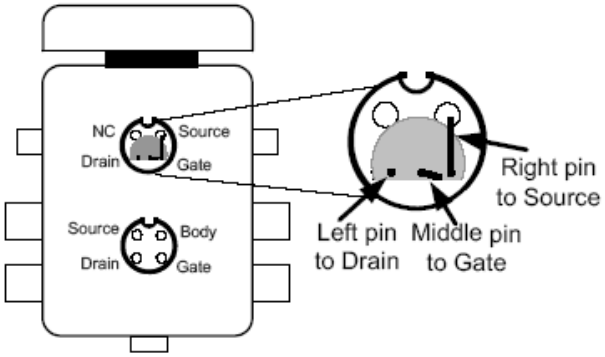


Figure 4-42
Proper orientation for the plastic package TO-92 n-channel enhancement mode FET



Using the Model 4200 Project QPulseIV-Complete for the first time

1. Connect up the Model 4200-SCS and 4200-PIV-Q package as documented above.
2. If KITE is not running, start KITE by double-clicking the KITE icon on the Model 4200-SCS desktop.
3. Open the QPulseIV-Complete project. Or click FILE > Open Project > if necessary, move up one level at the Open Project file window to the display the Projects directory. Double-click _Pulse folder, then double-click QPulseIV-Complete.kpr to open the project.
4. The screen should resemble the screen shot in Figure 4-43. The right portion of the screen may not be the graph shown, but the project navigator should match that shown in Figure 4-43.
5. Connect or touch-down on the chosen device under test (DUT). If using the 8101-PIV test fixture, see figures Figure 4-41 and Figure 4-42 for connection and proper DUT installation. The defaults for each test should provide reasonable results with the sample DUT. The plastic package of the DUT limits the cooling. Take care when running DC tests, as the device can be permanently damaged when running longer tests or higher powers.
6. Verify setup:
   a. Step 1: Follow the instructions for "Running scope-shot" to validate proper setup and operation of the PIV package. Ensure that both the gate and drain waveforms are visible and do not have any significant ringing or overshoot (see Figure 4-25).
   b. Step 2: Try running Vds-id-pulse ("Running vds-id-pulse UTM") and/or Vgs-id-pulse ("Running Vgs-id-pulse") and look for a characteristic response. If desired, DC IV tests may also be run ("Running Vds-id DC ITM"). Once both the scope-shot and a pulse IV test have been verified, pulse system calibration can be performed.
7. Calibration: Follow "Running AutocalScope" and "Running CableCompensation" in this document to perform the necessary pulse calibrations.

8. Use: After successful pulse calibrations, the system is now ready to be used for pulse and DC characterization of transistor devices.

Running AutocalScope

AutocalScope should be run before any pulse calibration is performed. For best Pulse IV results, the AutocalScope should also be run before the first experiments of the day.

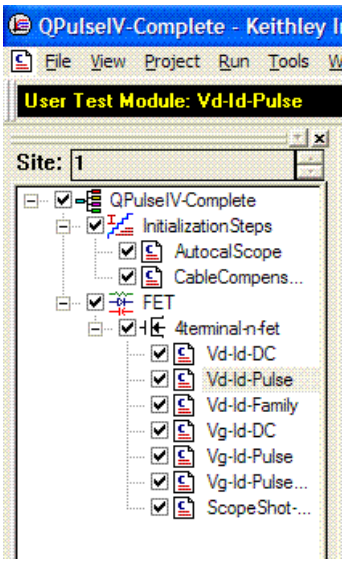
1. The Model 4200-SCS should be turned on at least 30 minutes before performing any calibration or measurements.
2. Double-click AutocalScope in the Project Navigator (Figure 4-43).
3. Click the green Run button.
4. Follow the instructions given on the pop-up dialog box and disconnect all connections to the 4200-SCP2HR scope card.
5. The SCP2HR scope performs an autocal, that takes about 2-3 minutes.
6. The test is complete when the Run button turns green. In the Sheet tab, an AutoCalStatus=0 indicates that there were no errors.
7. Reconnect the cables to the SCP2HR scope card. Use care when installing the cable to the scope card trigger SMB connector.

Running CableCompensation

Verify proper setup by running a scope-shot. For on-wafer testing, have a through, or short, structure available, or ensure that sharing pads for both sets of G-S-G probes provides a good connection. There are two steps to the calibration, open and through/short.

1. If not already performed, run the AutocalScope procedure above.
2. Double-click CableCompensation in the Project Navigator.
3. Click the green Run button to start the CableCompensation.
4. Click OK on the first dialog box to continue the CableCompensation.
5. The second dialog box requests that the probe pins be raised from the wafer, breaking contact. Raise probe pins or lower wafer to create the Open condition.
6. Click OK on the Open dialog box. The Open portion should take about one minute. If using the 8101-PIV test fixture, remove the DUT from the pulse socket. The pulse socket is the one closest to the hinge of the test fixture.
7. The third dialog box requests that the probe pins be connected to each other through another device. Lower the probes onto a through device. If using the 8101-PIV test fixture, install a shorting wire into the bottom two holes of the pulse socket, as shown in Figure 4-18.
8. Click OK on the Through dialog box. The Through portion should take around one minute.
9. The test is complete when the Run button turns green.
10. The system is now ready to test regular devices.

Figure 4-43
QPulse-IV-Complete project



Running Vd-Id-DC

The default settings sweep the drain from 0-15V in 200mV steps while applying a 5V gate voltage. To create a family of curves, change Vgs and use the Append button, or use Vd-Id-Family and supply Vgs start, stop and step voltages. All DC tests in this project use the SMUs. If the variation of the output impedance of the SMU as it ranges is undesirable, it is possible to use the pulse IV instrumentation to provide a DC signal for testing. Adjust the pulse timing parameters (period, rise, fall, width) to create a pulse with a duty cycle of 95+% and append to the low duty cycle results.

1. Double-click Vd-Id-DC in the Project Navigator (Figure 4-43).
2. Click the green Run button. Three Vds-id curves will be generated and displayed in Graph tab.

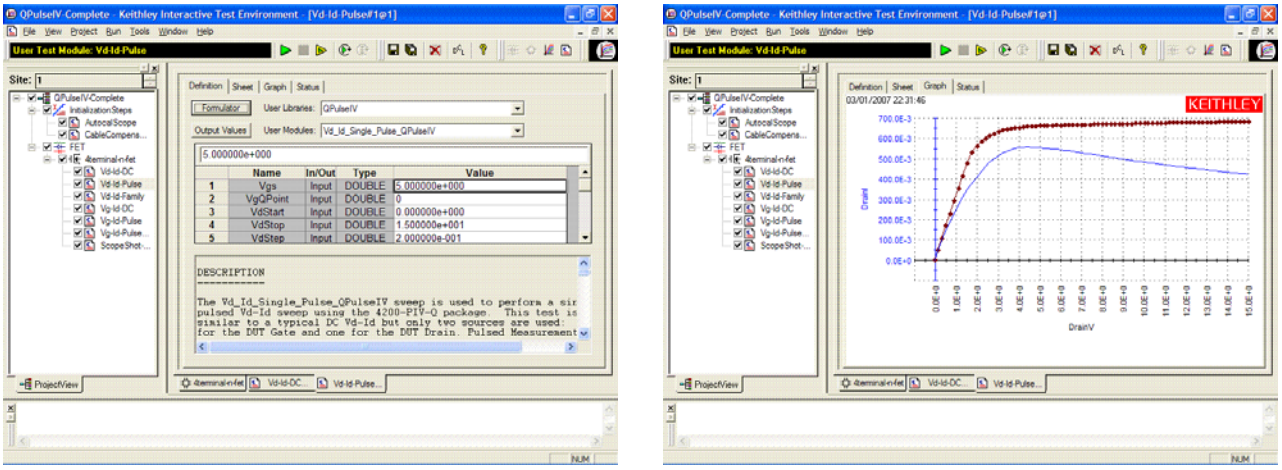
Running Vd-Id-Pulse

The default Vd-Id-Pulse test (Figure 4-44) uses the same drain voltage settings as the Vd-id-DC test. There are two ways to generate a family of pulse IV curves. The easier way is to use the Vd-Id-Family (see “Running Vd-Id-Family” below). To run the Vd-Id-Pulse test using a single gate voltage (5V), perform the following steps:

1. Ensure that the VdStart, VdStop, VdStep values match the values in Vd-Id-DC. To sweep from a high to a low voltage, enter voltages so that VdStop < VdStart and use a negative value for VdStep. If any values need to be modified, remember to press the Enter key after typing in the value.
2. Set Vg to 5V. Make sure to press the Enter key after typing in the value.
3. Click the green Run button.

During the sweep, the PG2 channel 1 is used to source a 5V pulse (500ns pulse-width) to the gate (Vg), while the PCU outputs a sweep of pulses from 0 to 15V on the drain. Figure 4-44 shows the Graph tab for this test. The red trace is the result of the pulse test, while the blue trace is the result of the Vd-Id-DC test (Vgs = 5V). The significant Id degradation in the DC results, due to self heating. These results are from the supplied demo DUT.

Figure 4-44
Definition and Graph tabs for Vd-Id-Pulse (Vgs = 5V)



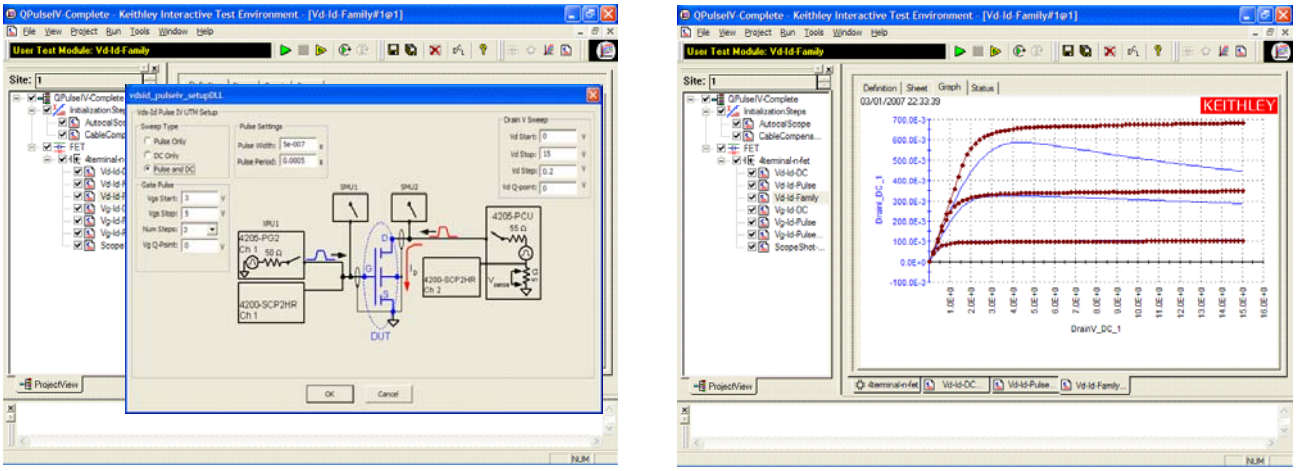
Running Vd-Id-Family

This test incorporates both pulse (Vd-Id-Pulse) and DC (Vd-Id-DC) into a single UTM to generate two families of curves. All DC tests in this project use the SMUs. If the variation of the output impedance of the SMU as it ranges is undesirable, it is possible to use the pulse IV instrumentation to provide a DC signal for testing. Adjust the pulse timing parameters (period, rise, fall, width) to create a pulse with a duty cycle of 95+% and append to the low duty cycle results.

1. If measurement parameters (pulse average, NPLC, measure range) need to be set, use the table in the Definition tab.
2. (Optional) If only source parameters need to be changed, use the UTM GUI by clicking the GUI button on the Definition tab. Modify the source parameters in the GUI, and click OK when finished. Figure 4-45 shows the opened GUI.
3. Click the green Run button. For a test with three curves and 40 points per curve, the test should take about 1.5-2 minutes.

Figure 4-45 shows the Graph tab. This combined test shares the same Vg and Vd values, in addition to all of the DC and pulse measurement parameters, that are not on the GUI, but are available in the Definition tab. The test alternates between the DC IV and pulse IV tests, so each Vg is run first as DC then as pulse before stepping to the next Vg voltage. During the test, neither the Graph tab or Sheet tab is updated.

Figure 4-45
Default definition and typical graph for Vd-Id-Family



Running Vg-Id-DC

The default settings sweep the gate from 0-3V in 100mV steps. The single DC sweep using a SMU is performed using a drain bias voltage (Vd) of 4V.

1. Double-click Vg-Id-DC in the Project Navigator (Figure 4-43).
2. Click the green Run button. One Vg-id curve will be generated and displayed in Graph tab.

Running Vg-Id-Pulse

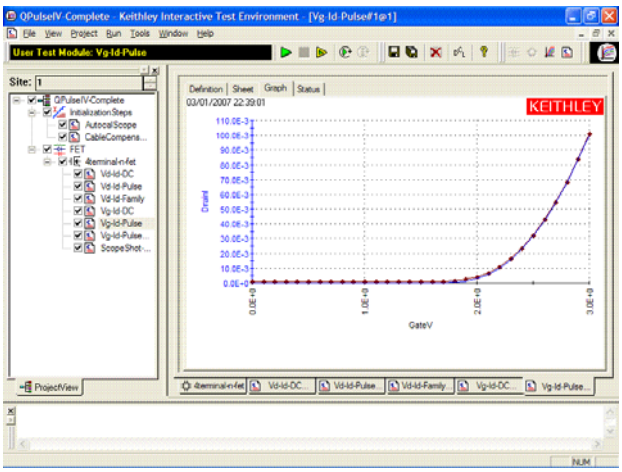
This test is similar to the Vg-Id-DC test but uses pulses. This test uses the PG2 to perform a single pulse Vg sweep from 0 to 3V in 100mV steps. During the sweep, the PCU is used to source a 4V bias pulse (500ns pulse-width) to the drain (Vd). To compare DC and Pulse IV Vg-Id results, follow the procedure below, or use the UTM Vg-Id-Pulse-vs-DC (see "Running Vg-Id-Pulse-vs-DC").

To run the Vg-Id-Pulse test, perform the following steps:

1. Ensure that the VgStart, VgStop, VgStep values match the values in Vd-Id-DC. To sweep from a high to a low voltage, enter voltages so that VgStop < VgStart and use a negative value for VgStep. If any values need to be modified, remember to press the Enter key after typing in the value.
2. Set Vd to 4V. Make sure to press the Enter key after typing in the value.
3. Click the green Run button.

Figure 4-46 shows the Graph tab for this test. The red trace is the result of the pulse test, while the blue trace is the result of the Vg-Id-DC.

Figure 4-46
Vg-Id-Pulse Graph tab



Running Vg-Id-Pulse-vs-DC

This test incorporates both pulse (Vg-Id-Pulse) and DC (Vg-Id-DC) into a single UTM to generate two traces. Figure 4-47 shows Definition tab for this test. Test parameters can also be entered from the GUI, that is displayed by clicking the GUI button. The GUI is also shown in Figure 4-47. All DC tests in this project use the SMUs. If the variation of the output impedance of the SMU as it ranges is undesirable, it is possible to use the pulse IV instrumentation to provide a DC signal for testing. Adjust the pulse timing parameters (period, rise, fall, width) to create a pulse with a duty cycle of 95+% and append to the low duty cycle results.

1. If measurement parameters (pulse average, NPLC, measure range) need to be set, use the table in the Definition tab. For best results when measuring low Id currents, set AverageNum = 0, that will use Adaptive Filtering.

2. (Optional) If only source parameters need to be changed, use the UTM GUI by clicking on the GUI button in the Definition tab. Modify the source parameters and click OK when finished.
3. Click the green Run button.

Figure 4-47 shows the Graph tab. This combined test shares the same Vg and Vd values, in addition to all of the DC and pulse measurement parameters. The test alternates between the DC IV and pulse IV tests to generate the two traces. Neither the Graph tab or Sheet tab is updated.

Figure 4-47
Default GUI definition for Vg-Id-Pulse-vs-DC

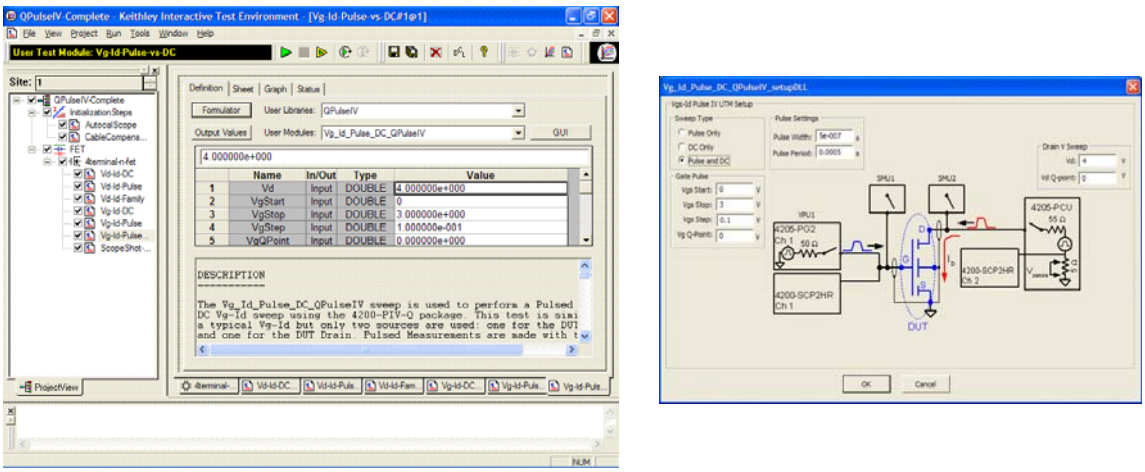
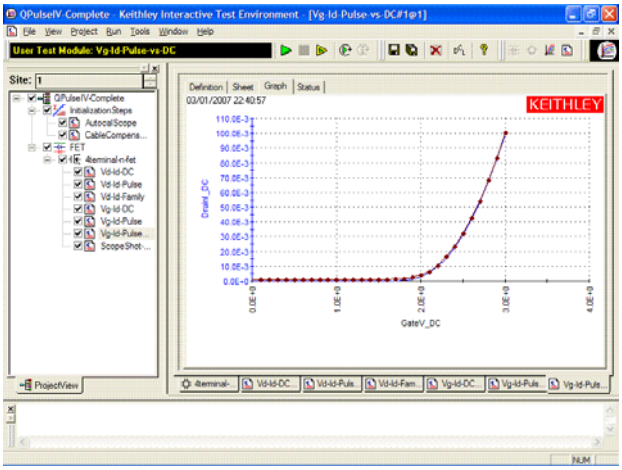


Figure 4-48
Typical graph for Vg-Id-Pulse-vs-DC



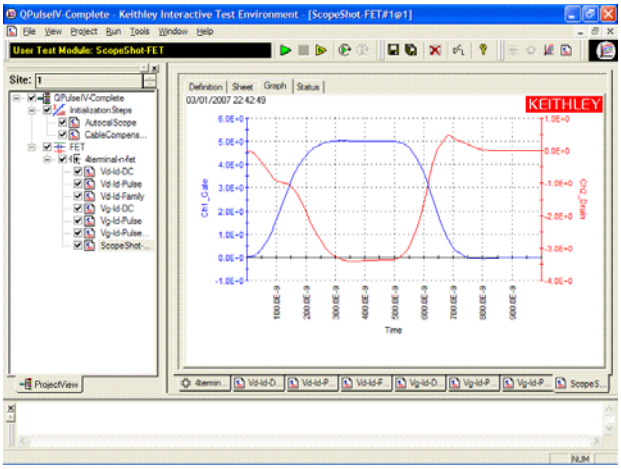
Running ScopeShot-FET

The ScopeShot-FET test is used to verify proper connection and system setup. The waveform shown in Figure 4-47 is a typical result; actual results should be similar. If waveform has significant ringing or overshoot, the pulse IV tests will not provide good results. Check the pulse interconnects to ensure proper cabling and ensure all connections are tight. If using a PRB-C (Y adapter cable for pulsing with DC interconnect and structures), ensure that the two ground lugs are connected together.

The blue pulse curve is the pulse applied to the gate. The displayed waveform data has approximate calibration factors applied, but the calibrated measurement is given in the lower left portion of the Graph tab. The red pulse curve is the drain current, shown with approximate calibration factors applied, with the calibrated Vd and Id measurements listed in the lower left corner of the graph.

See “Adjustable parameters in scopeshot_cal_pulseiv” for information on parameters for ScopeShot-FET.

Figure 4-49
Typical graphical result for ScopeShot -FET



PIV-Q user libraries

The user library UTMs used in the PIV-Q package are documented as follows:

- – The DualPulseulib UTMs are used to control the Dual Card Pulser through the Model 4205-PCU (Pulse Combiner Unit). These UTMs are used in the Model 4200-PIV-Q package.
- – The QPulseIVulib UTMs are used for the Model 4200-PIV-Q package.

DualPulseulib UTM descriptions

The DualPulseulib UTMs are used to control the Dual Card Pulser through the Model 4205-PCU (Pulse Combiner Unit). The Pulse Combiner Unit is supplied with the Model 4205-PIV-Q and provides dual pulse control of two adjacent Model 4205-PG2 pulse generators. For more information, refer to Section 12 of the Reference Manual and the PIVQ application in the Applications Manual.

The modules contained in the dual pulse user library are listed in Table 4-46 with detailed information following the table.

NOTE *Dual pulse control is by a usrlib, not within LPTLib. LPTLib commands such as devclr and devint do not apply to the Dual Card Pulser. To initialize the Dual Card Pulser, use the dpulse_init function.*

Table 4-46
DualPulseulib UTMs

User Module	Description
dpulse_burst_count	Sets the burst count for the Dual Card Pulser.
dpulse_current_limit	Sets current limit for the Dual Card Pulser.
dpulse_delay	Sets the pulse delay for the Dual Card Pulser.
dpulse_fall	Sets the pulse fall time for the Dual Card Pulser.
dpulse_float	Disconnects (floats) the output shield of Dual Card Pulser from ground.
dpulse_halt	Stops all pulse output from the Dual Card Pulser.
dpulse_init	Initializes the Dual Card Pulser.
dpulse_load	Sets the output impedance of the Dual Card Pulser.
dpulse_output	Turns the output of the Dual Card Pulser on or off.
dpulse_output_mode	Sets the output mode for the Dual Card Pulser.
dpulse_period	Sets the period for the Dual Card Pulser.
dpulse_range	Sets the source range for the Dual Card Pulser.
dpulse_rise	Sets the pulse rise time for the Dual Card Pulser.
dpulse_trig	Initiates the start of pulse output, and sets the trigger mode.
dpulse_trig_polarity	Sets the trigger polarity of the Dual Card Pulser.
dpulse_vhigh	Sets the high level value for pulse output of the Dual Card Pulser.
dpulse_vlow	Sets the low level value for pulse output of the Dual Card Pulser.
dpulse_width	Sets the pulse width for the Dual Card Pulser.

dpulse_burst_count

Module Return Type: int
Number of Parameters: 1

Description This function sets the burst count for the Dual Card Pulser. It sets the number of pulses (periods) that the pulser will output when used in the burst mode.

The parameter and return values for this function are listed in Table 4-47 and Table 4-48.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function, and use dpulse_trig to set the trigger mode to burst.

Table 4-47
Parameter for pulse_burst_count

Name	Description
PulseCount	(long) The burst count for Dual Card Pulser.

Table 4-48
Return values for dpulse_burst_count

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_current_limit

Module Return Type: int
Number of Parameters: 1

Description This function sets current limit for the Dual Card Pulser. Maximum limit values are source range dependent:

10V range (high speed): Maximum current limit is 400mA.
40V range (high voltage): Maximum current limit is 800mA.

The current limit is used to protect the Device Under Test (DUT) by using the DUT load impedance setting to calculate the maximum voltage that will be output to the DUT.

The parameter and return values for this function are listed in Table 4-49 and Table 4-50.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function, and use dpulse_load function to set the DUT load impedance.

Table 4-49
Parameter for dpulse_current_limit

Name	Description
PulseCurrentLimit	(double) The current limit of the pulser (amps).

Table 4-50
Return values for dpulse_current_limit

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_delay

Module Return Type: int
Number of Parameters: 1

Description This function sets the delay period from trigger to when the pulse output starts. Delay can be set from 0s to (Period - 10ns).

The parameter and return values for this function are listed in Table 4-51 and Table 4-52.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function.

Table 4-51
Parameter for dpulse_delay

Name	Description
PulseDelay	(double) The pulse delay (seconds).

Table 4-52
Return values for dpulse_delay

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_fall

Module Return Type: int
Number of Parameters: 1

Description This function sets pulse fall time for the Dual Card Pulser. Minimum fall time values are source range dependent:

10V range (high speed): Minimum fall time is 10ns.
40V range (high voltage): Minimum fall time is 100ns.

Based on system and interconnect bandwidth, the actual fall time may need to be greater than the minimum to obtain an acceptable pulse shape.

The parameter and return values for this function are listed in Table 4-53 and Table 4-54.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function, and use the dpulse_range function to set the source range.

Table 4-53
Parameter for dpulse_fall

Name	Description
FallTime	(double) The fall time for the pulse output (seconds).

Table 4-54
Return values for dpulse_fall

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.

Table 4-54
Return values for dpulse_fall

Value	Description
Other	All other error cards match up to LPT errors.

dpulse_float

Module Return Type: int
Number of Parameters: 1

Description This function floats (disconnects) the output shield of the Dual Card Pulser from ground. The Floating setting is intended to be used in conjunction with a scope (sense output located on the Model 4205-PCU) and the PIVQ solution. The No Float setting connects the output shield to ground, bypassing the sense resistor through the Sense connector of the Model 4205-PCU.

The parameter and return values for this function are listed in Table 4-55 and Table 4-56.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function.

Table 4-55
Parameter for dpulse_float

Name	Description
FloatEnable	(long) Sets the float of the pulser: 0 - No Float (pulser is grounded). 1 - Floating (pulser is not grounded and will need a ground through the Model 4200-PCU sense connection.

Table 4-56
Return values for dpulse_float

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_halt

Module Return Type: int
Number of Parameters: 0

Description This function stops all (halts) pulse output from the Dual Card Pulser and turns the output off.

There are no parameters for this function. The return values are listed in Table 4-58.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function, and use the dpulse_range function to set the source range.

Pulse output can be restarted by first turning the outputs back on using dpulse_output, and then starting pulse output using the dpulse_trig function.

Table 4-57
Parameter for dpulse_halt

Name	Description
(none)	N/A

Table 4-58
Return values for dpulse_halt

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_init

Module Return Type: int
Number of Parameters: 2

Description This function initializes the Dual Card Pulser. The Dual Card Pulser consists of two adjacent Model 4205-PG2s that are electrically "stacked" on top of each other through the Model 4205-PCU (Pulse Combiner Unit). Initilization configures each card in the stack using the specified card IDs for the two pulser cards being stacked. Usually, the lower ID PG2 is the HighVpu. In the case where VPU1 and VPU2 are being stacked, VPU1 will be the HighVpu, and VPU2 will be the LowVpu.

The trigger output of the Model 4205-PCU is the same as the trigger output of the HighVpu. In addition to the output setup, this routine also configures the triggers.

The parameters and return values for this function are listed in Table 4-59 and Table 4-60.

Table 4-59
Parameters for dpulse_init

Name	Description
HighVpuId	The PG2 that is at the top of the stack (VPU1)
LowVpuId	The PG2 that is at the bottom of the stack (VPU2)

Table 4-60
Return values for dpulse_init

Value	Description
0	No Errors.
-1	LowVpuId is not in the system configuration.
-2	HighVpuId is not in the system configuration.
-3	HighVpuId and LowVpuId cannot be the same.
Other	All other error cards match up to LPT errors.

dpulse_load

Module Return Type: int

Number of Parameters: 1

Description This function sets the impedance of the load connected to the Dual Card Pulser. This impedance setting determines the actual pulse voltages output by the pulser and is used for calculating the maximum voltage for the dpulse_current_limit function. Valid parameter values range from 1 Ohm to 1M Ohm.

The parameter and return values for this function are listed in Table 4-61 and Table 4-62.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function, and use the dpulse_range function to set the source range.

Table 4-61
Parameter for dpulse_load

Name	Description
PulseLoad	(double) The target (DUT) load impedance (ohms).

Table 4-62
Return values for dpulse_load

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_output

Module Return Type: int
Number of Parameters: 1

Description This function turns the output of the Dual Card Pulser on or off.

The parameter and return values for this function are listed in Table 4-63 and Table 4-64.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function. With the pulser turned on, pulse output will start when a trigger is initiated by the dpulse_init function.

Table 4-63
Parameter for dpulse_output

Name	Description
State	(long) The state of the pulser output: 0 - Output is off. 1 - Output is on.

Table 4-64
Return values for dpulse_output

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_output_mode

Module Return Type: int
Number of Parameters: 1

Description This function sets the pulse output mode of the Dual Card Pulser. There are two modes, NORMAL and COMPLEMENT. When in complement mode, the pulser's VHigh and VLow are swapped.

The parameter and return values for this function are listed in Table 4-65 and Table 4-66.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function.

Table 4-65
Parameter for dpulse_output_mode

Name	Description
Mode	(long) The output mode for the pulser: 0 - Normal mode. 1 - Complement mode (VHigh and VLow are swapped).

Table 4-66
Return values for dpulse_output_mode

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_period

Module Return Type: int
Number of Parameters: 1

Description This function sets the period for the Dual Card Pulser. The maximum period is one second, while the minimum period depends on the rise time, fall time, and pulse width. The minimum period is calculated as follows:

Minimum Period = (Rise Time / 2) + Pulse Width + (Fall Time / 2) + 10ns

The parameter and return values for this function are listed in Table 4-67 and Table 4-68.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function.

Table 4-67
Parameter for dpulse_period

Name	Description
Period	(Double) The period for pulse output (seconds).

Table 4-68
Return values for dpulse_period

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.

Table 4-68
Return values for dpulse_period

Value	Description
Other	All other error cards match up to LPT errors.

dpulse_range

Module Return Type: int
Number of Parameters: 1

Description This function sets the source range for the Dual Card Pulser. There are two source ranges: High Speed (±10V into 50Ω load) and High Voltage (±40V into 50Ω load).

The parameter and return values for this function are listed in Table 4-69 and Table 4-70.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function.

Table 4-69
Parameter for dpulse_range

Name	Description
Range	(double) The range for the pulser: ±10V - High Speed. ±40V - High Voltage.

Table 4-70
Return values for dpulse_range

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_rise

Module Return Type: int
Number of Parameters: 1

Description This function sets pulse rise time for the Dual Card Pulser. Minimum rise time values are source range dependent:

10V range (high speed): Minimum rise time is 10ns.
40V range (high voltage): Minimum rise time is 100ns.

Based on system and interconnect bandwidth, the actual rise time may need to be greater than the minimum to obtain an acceptable pulse shape.

The parameter and return values for this function are listed in Table 4-71 and Table 4-72.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function, and use the dpulse_range function to set the source range.

Table 4-71
Parameter for dpulse_rise

Name	Description
RiseTime	(double) The rise time for the pulse output (seconds).

Table 4-72
Return values for dpulse_rise

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_trig

Module Return Type: int
Number of Parameters: 1

Description This function is used to initiate the start of pulse output on the Dual Card Pulser. It also selects the trigger mode: burst or continuous.

The parameter and return values for this function are listed in Table 4-73 and Table 4-74.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function. If using the Burst Mode, set the burst count with the dpulse_burst_count function.

Table 4-73
Parameter for dpulse_trig

Name	Description
Mode	(long) The pulse output mode: 0 - Burst Mode (using the dpulse_burst_count). 1 - Continuous Mode.

Table 4-74
Return values for dpulse_trig

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_trig_polarity

Module Return Type: int
Number of Parameters: 1

Description This function sets the trigger polarity of the Dual Card Pulser output trigger. The trigger is a TTL level output at the same frequency (period) as the pulse output.

The parameter and return values for this function are listed in Table 4-75 and Table 4-76.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function.

Table 4-75
Parameter for dpulse_trig_polarity

Name	Description
TrigPolarity	(long) The trigger polarity: 0 - Negative trigger polarity. 1 - Positive trigger polarity.

Table 4-76
Return values for dpulse_trig_polarity

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_vhigh

Module Return Type: int
Number of Parameters: 1

Description This function sets the high voltage value of the output pulse of the Dual Card Pulser. The output pulse has two levels, vlow and vhigh. The level is dependent on the source range and pulse load. The difference between pulse low and pulse high is the peak-to-peak amplitude of the output pulse.

50 Ohm Load:

10V Range (High Speed) – Pulse high and pulse low can be set between -10V and +10V.

40V Range (High Voltage) – Pulse high and pulse low can be set between -40V and +40V.

1M Ohm Load:

10V Range (High Speed) – Pulse high and pulse low can be set between -20V and +20V.

40V Range (High Voltage) – Pulse high and pulse low can be set between -80V and +80V.

Related functions: dpulse_current_limit, dpulse_range, dpulse_vlow

The parameter and return values for this function are listed in Table 4-77 and Table 4-78.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function.

Table 4-77
Parameter for dpulse_vhigh

Name	Description
PulseVHigh	(double) The pulse high value (V).

Table 4-78
Return values for dpulse_vhigh

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_vlow

Module Return Type: int
Number of Parameters: 1

Description This function sets the low voltage value of the output pulse of the Dual Card Pulser. The output pulse has two levels, vlow and vhigh. The level is dependent on the source range and pulse load. The difference between pulse low and pulse high is the peak-to-peak amplitude of the output pulse.

50 Ohm Load:

10V Range (High Speed) – Pulse high and pulse low can be set between -10V and +10V.

40V Range (High Voltage) – Pulse high and pulse low can be set between -40V and +40V.

1M Ohm Load:

10V Range (High Speed) – Pulse high and pulse low can be set between -20V and +20V.

40V Range (High Voltage) – Pulse high and pulse low can be set between -80V and +80V.

Related functions: dpulse_current_limit, dpulse_range, dpulse_vhigh

The parameter and return values for this function are listed in Table 4-49 and Table 4-50.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function.

Table 4-79
Parameter for dpulse_vlow

Name	Description
PulseVLow	(double) The pulse low value (V).

Table 4-80
Return values for dpulse_vlow

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

dpulse_width

Module Return Type: int
Number of Parameters: 1

Description This function sets the pulse width of the Dual Card Pulser. The pulse width includes half of the rise time and half of the fall time (FWHM: full width @ half maximum). The maximum width depends on the range:

10V range (high speed): 10ns to (period - 10ns)
40V range (high voltage): 100ns to (period - 100ns)

Based on system and interconnect bandwidth, the actual pulse width may need to be greater than the minimum to obtain an acceptable pulse shape.

The parameter and return values for this function are listed in Table 4-81 and Table 4-82.

Procedure Before using this function, make sure the Dual Card Pulser has been initialized using the dpulse_init function.

Table 4-81
Parameter for dpulse_width

Name	Description
PulseWidth	(double) The pulse width for the Dual Card Pulser.

Table 4-82
Return values for dpulse_width

Value	Description
0	No Errors.
-1	Dual Card Pulser has not been initialized.
Other	All other error cards match up to LPT errors.

QPulseIVulib UTM descriptions

The UTMs used for the Model 4200-PIV-Q package are summarized in Table 4-83. Details for the UTMs follow the table.

Table 4-83
QPulseIVulib UTMs

User Module	Description
CableCompensation_QPulseIV	Used to determine the compensation factors required to account for cable losses incurred during the pulse and DC sweeps for the Model 4200-PIV-Q package.
ScopeShot_FET_QPulseIV	Used to run a single pulse/point measurement on a FET DUT (FET, HEMT, and so on.) using the Model 4200-PIV-Q package.
Vd_Id_Pulse_DC_Family_QPulseIV	Used to perform a Pulsed vs. DC Vd-Id sweep using the Model 4200-PIV-Q package.
Vd_Id_Single_DC_QPulseIV	Used to perform a single DC Vd-Id sweep using the Model 4200-PIV-Q package.
Vd_Id_Single_Pulse_QPulseIV	Used to perform a single pulsed Vd-Id sweep using the Model 4200-PIV-Q package.
Vg_Id_Pulse_DC_QPulseIV	Used to perform a Pulsed vs. DC Vg-Id sweep using the Model 4200-PIV-Q package.
Vg_Id_Single_DC_QPulseIV	Used to perform a DC Vg-Id sweep using the Model 4200-PIV-Q package.

Table 4-83
QPulseIVulib UTMs

User Module	Description
Vg_Id_Single_Pulse_QPulseIV	Used to perform a Pulsed Vg-Id sweep using the Model 4200-PIV-Q package.

Return codes:

All modules share the following return codes:

0	No Errors
-15001	PIV-Q hardware has not been initialized
-15002	Unable to trigger the Gate/Base Pulser
-15003	Unable to trigger the Drain/Collector Pulser
-15004	Unable to obtain a scope reading
-15005	Unable to set scope cursors
-15006	Unable to use load line, as the required voltage is beyond source range
-15007	Invalid Gate/Base PG2 ID Specified
-15008	Invalid Drain/Collector PG2 High ID Specified
-15009	Invalid Drain/Collector PG2 Low ID Specified
-15010	Invalid Gate/Base SMU ID Specified
-15011	Invalid Drain/Collector SMU ID Specified
-15012	Unable to force given voltage on Gate/Base
-15013	Unable to force given voltage on Drain/Collector
-15014	Invalid Cable Compensation Type (not “OPEN” or “THRU”)
-15015	Unable to find Cable Compensation factor in file
-15016	Unable to write Cable Compensation to file
-15017	Unable to allocate enough memory for test
-15018	Unable to close/open the Gate/Base SMU relays
-15019	Unable to close/open the Drain SMU relays
-15020	Unable to close/open the Gate/Base PG2 Solid State Relays (HEOR)
-15021	Unable to close/open the Drain/Collector Solid State Relays (HEOR)
-15022	Unable to initialize the scope
-15023	Unable to set scope impedance
-15024	Unable to set the scope range
-15025	Unable to configure the scope measure
-15026	Unable to set the scope sample rate
-15027	Unable to set the scope offset
-15028	Unable to set the scope to external triggering
-15029	Unable to set the average number desired on the scope
-15030	Unable to set the scope trigger delay
-15031	Unable to read waveform from scope
-15032	Unable to set the scope delay
-15033	Unable to capture waveform on scope
-15034	Unable to find Gate/Base PG2 calibration values
-15035	Unable to find Drain/Collector High PG2 calibration values
-15036	Unable to find Drain/Collector Low PG2 calibration values
-15037	Unable to initialize Gate/Base PG2

-15038 Unable to initialize Drain/Collector High PG2
-15039 Unable to initialize Drain/Collector Low PG2
-15040 Unable to initialize the Model 4205-PCU
-15041 Unable to set the range on the Model 4205-PCU
-15042 Unable to set the period on the Model 4205-PCU
-15043 Unable to set rise time on the Model 4205-PCU
-15044 Unable to set fall time on the Model 4205-PCU
-15045 Unable to set pulse width on the Model 4205-PCU
-15046 Unable to set the Qpoint on the Model 4205-PCU
-15047 Unable to set the load on the Model 4205-PCU
-15048 Unable to set the burst count on the Model 4205-PCU
-15049 Unable to set the delay on the Model 4205-PCU
-15050 Unable to set the current limit on the Model 4205-PCU
-15051 Unable to turn on/off the output on the Model 4205-PCU
-15052 Unable to float the Model 4205-PCU
-15053 Unable to set VHigh on the Model 4205-PCU
-15054 Unable to set VLow on the Model 4205-PCU
-15055 Unable to trigger the Model 4205-PCU
-15056 Unable to halt the Model 4205-PCU
-15057 Unable to initialize the Gate/Base PG2
-15058 Unable to set the range on the Gate/Base PG2
-15059 Unable to set the period on the Gate/Base PG2
-15060 Unable to set the rise time on the Gate/Base PG2
-15061 Unable to set the fall time on the Gate/Base PG2
-15062 Unable to set the pulse width on the Gate/Base PG2
-15063 Unable to set the QPoint on the Gate/Base PG2
-15064 Unable to set the burst count on the Gate/Base PG2
-15065 Unable to set the trigger source on the Gate/Base PG2
-15066 Unable to set the current limit on the Gate/Base PG2
-15067 Unable to set the load on the Gate/Base PG2
-15068 Unable to turn the output of the Gate/Base PG2 on/off
-15069 Unable to halt the Gate/Base PG2

CableCompensation_QPulseIV

Module Return Type: int
Number of Parameters: 5

Arguments:

GateSMU,char *,Input,“SMU1”,,
DrainSMU,char *,Input,“SMU2”,,
Getlevel,char *,Input,"VPU1",,
DrainVPUHigh,char *,Input,"VPU2",,
DrainVPULow,char *,Input,"VPU3",,

Description:

The CableCompensation_QPulseIV function is used to determine the compensation factors required to account for cable losses incurred during the pulse and DC sweeps in the Model 4200-PIV-Q package. This compensation is unique to this package and does not apply to any

other configurations. Please refer to the manual for proper PIV-Q setup and interconnect instructions.

There are two parts to this compensation. The first part is an OPEN test that will take a few seconds to complete. The second part is a THRU/SHORT that will take about two minutes to complete. The THRU/SHORT test requires a THRU structure (on wafer) or some method to connect the Gate/Base and Drain/Collector signals together. Both parts are required for a complete compensation and it is recommended that a compensation be run every time the setup has been moved or has changed in any way. Before running this routine perform an Autocal on the scope card.

The Model 4205-PCU combines two Model 4205-PG2 cards into a single, higher power pulse channel to provide power to the DUT Drain/Collector.

Inputs:

GateSMU	(char*) String representing the ID for the SMU connected to the gate (e.g., SMU1).
DrainSMU	(char*) String representing the ID for the SMU connected to the drain (e.g., SMU2).
GateVPU	(char*) String representing the ID for the PG2 connected to the gate (e.g., VPU1).
DrainVPUHigh	(char*) String representing the top VPU on the Model 4205-PCU. This is the card on the right (or the card with the lower ID number), when facing the back of the system (e.g., VPU2). DrainVPULow(char*) String representing the bottom VPU on the Model 4205-PCU. This is the card on the left (or the card with the higher ID number), when facing the back of the system (e.g., VPU3).

Outputs:

NONE

ScopeShot_FET_QPulseIV

Module Return Type: int
Number of Parameters: 34

Arguments:

Vgs,double,Input,2,,
VgQPoint,double,Input,0,,
Vds,double,Input,4,,
VdQPoint,double,Input,0,,
PulseWidth,double,Input,300e-9,,
PulsePeriod,double,Input,1e-6,,
RiseTime,double,Input,100e-9,,
FallTime,double,Input,100e-9,,
Average,int,Input,100,,
GateVPURange,double,Input,5,,
DrainVPURange,double,Input,5,,
GateScpRange,double,Input,0,,
DrainScpRange,double,Input,0,,
GateCompliance,double,Input,.1,,
DrainCompliance,double,Input,.1,,
GateLoadLine,int,Input,1,,
DrainLoadLine,int,Input,1,,

GateSMU,char *,Input,"SMU1",,
DrainSMU,char *,Input,"SMU2",,
GateVPU,char *,Input,"VPU1",,
DrainVPUHigh,char *,Input,"VPU2",,
DrainVPULow,char *,Input,"VPU3",,
Time,D_ARRAY_T,Output,,,
TimeSize,int,Input,10000,,
Ch1Out,D_ARRAY_T,Output,,,
Ch1OutSize,int,Input,10000,,
Ch2Out,D_ARRAY_T,Output,,,
Ch2OutSize,int,Input,10000,,
GateV,double *,Output,,,
GateI,double *,Output,,,
DrainV,double *,Output,,,
DrainI,double *,Output,,,
MeasurementCursor1,double *,Output,,,
MeasurementCursor2,double *,Output,,,

Description:

The ScopeShot_FET_QPulseIV function is used to run a single pulse/point measurement on a FET DUT (FET, HEMT, and so on.) using the Model 4200-PIV-Q package. This measurement is unique to this package and should not be used with any other setups, as unexpected signals may be provided and/or improper results may be reported. Please refer to the manual for proper PIV-Q setup instructions.

ScopeShot will only perform pulse measurements based on a single set of parameters (no sweeping). Also, this routine will source more than the specified number of pulses in Average if either GateScpRange or DrainScpRange are auto-range, or GateLoadLine or DrainLoadLine are = 1.

All voltage levels specified below assume a 50 ohm DUT load.

Inputs:

Vgs	(double) The pulsed Gate-Source voltage bias. Vgs is range dependent and can be between -5V to +5V or between -20V to +20V.
VgQPoint	(double) The base, or bias point, of the pulsed Gate source. VgQPoint is range dependent and can be between -5V to +5V or between -20V to +20V.
Vds	(double) The pulsed Drain-Source voltage bias. Vds is range dependent and can be between -10V to +10V or between -40V to +40V.
VdQPoint	(double) The base, or bias point, of the pulsed Gate source. VdQPoint is range dependent and can be between -10V to +10V or between -40V to +40V.
PulseWidth	(double) The Vgs and Vds pulse width (PW). The Pulse Width can be 300ns to (1s - 10 ns) in 10ns resolution steps.
PulsePeriod	(double) The pulse period for both Vgs and Vds. Minimum period is (FallTime/2) + (RiseTime/2) + PulseWidth + 10ns. Maximum period is 1s.
RiseTime	(double) The transition time from the Qpoint to the pulse value for both Vgs and Vds.The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and fall time.

FallTime	(double) The transition time from the pulse value to the Qpoint value for both Vgs and Vds. The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and fall time.
Average	(int) The number of pulses to average (recommend 100 for a reasonable result, use larger for lower current measurements).
GateVPURange	(double) The source range for DUT Gate PG2. Valid ranges are 5V (High Speed) and 20V (High Voltage).
DrainVPURange	(double) The source range for the DUT Drain Model 4205-PCU. Valid ranges are 10V (High Speed) and 40V (High Voltage).
GateScpRange	(double) The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).
DrainScpRange	(double) The voltage measure range for the scope channel measuring the Drain. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).
GateCompliance	(double) The current compliance on the gate. This supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load.
DrainCompliance	(double) The current compliance on the drain. This supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load.
GateLoadLine	(int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Gate. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).
DrainLoadLine	(int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Drain. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).
GateSMU	(char*) String representing the ID for the SMU connected to the gate.
DrainSMU	(char*) String representing the ID for the SMU connected to the drain.
GateVPU	(char*) String representing the ID for the PG2 connected to the gate.
DrainVPUHigh	(char*) String representing the top VPU on the Model 4205-PCU. This is the card on the right (or the card with the lower ID number), when facing the back of the system.
DrainVPULow	(char*) String representing the bottom VPU on the Model 4205-PCU. This is the card on the left (or the card with the higher ID number), when facing the back of the system.
TimeSize Ch1OutSize Ch2OutSize	(int) Sizes of the output arrays. All arrays should be the same size and need to be large enough to hold all sweep points.

Outputs:

Time	(double) Array of time values corresponding to the pulsed scope shot.
Ch1Out	(double) Array of Channel 1 (Gate) values as seen on the scope.
Ch2Out	(double) Array of Channel 2 (Drain) values as seenon the scope.
GateV	(double) Measured Gate Voltage. GateI(double) Measured Gate Current.
DrainV	(double) Measured Drain Voltage. DrainI(double) Measured Drain Current.
MeasurementCursor1	(double) Cursor used to measure the voltage and current values on the scope.
MeasurementCursor2	(double) Cursor used to measure the voltage and current values on the scope.

Vd_Id_Pulse_DC_Family_QPulseIV

Module Return Type: int
Number of Parameters: 195

Arguments:

VgStart,double,Input,1.5,,
VgStop,double,Input,2.5,,
VgNumSteps,int,Input,3,,
VdStart,double,Input,0,,
VdStop,double,Input,4,,
VdStep,double,Input,.05,,
VgQPoint,double,Input,0,,
VdQPoint,double,Input,0,,
PulseWidth,double,Input,300e-9,,
PulsePeriod,double,Input,1e-6,,
RiseTime,double,Input,100e-9,,
FallTime,double,Input,100e-9,,
AverageNum,int,Input,100,,
GateVPURange,double,Input,5,,
DrainVPURange,double,Input,5,,
GateSMURange,int,Input,1,,
DrainSMURange,int,Input,1,,
GateScpRange,double,Input,0,,
DrainScpRange,double,Input,0,,
GateLoadLineCorrection,int,Input,1,,
DrainLoadLineCorrection,int,Input,1,,
GateCompliance,double,Input,.1,,
DrainCompliance,double,Input,.1,,
MaxIg,double,Input,1,,
MaxId,double,Input,1,,
MaxPowerGate,double,Input,200,,
MaxPowerDrain,double,Input,200,,
NPLC,double,Input,.01,,
DCSourceDelay,double,Input,0,,
DC_vs_Pulse,int,Input,2,,

```
GateSMU,char *,Input,"SMU1",,
DrainSMU,char *,Input,"SMU2",,
GateVPU,char *,Input,"VPU1",,
DrainVPUHigh,char *,Input,"VPU2",,
DrainVPULow,char *,Input,"VPU3",,
DrainV_DC_1,D_ARRAY_T,Output,,,
DrainV_DC_1_Size,int,Input,10000,,
DrainI_DC_1,D_ARRAY_T,Output,,,
DrainI_DC_1_Size,int,Input,10000,,
GateV_DC_1,D_ARRAY_T,Output,,,
GateV_DC_1_Size,int,Input,10000,,
GateI_DC_1,D_ARRAY_T,Output,,,
GateI_DC_1_Size,int,Input,10000,,
DrainV_DC_2,D_ARRAY_T,Output,,,
DrainV_DC_2_Size,int,Input,10000,,
DrainI_DC_2,D_ARRAY_T,Output,,,
DrainI_DC_2_Size,int,Input,10000,,
GateV_DC_2,D_ARRAY_T,Output,,,
GateV_DC_2_Size,int,Input,10000,,
GateI_DC_2,D_ARRAY_T,Output,,,
GateI_DC_2_Size,int,Input,10000,,
DrainV_DC_3,D_ARRAY_T,Output,,,
DrainV_DC_3_Size,int,Input,10000,,
DrainI_DC_3,D_ARRAY_T,Output,,,
DrainI_DC_3_Size,int,Input,10000,,
GateV_DC_3,D_ARRAY_T,Output,,,
GateV_DC_3_Size,int,Input,10000,,
GateI_DC_3,D_ARRAY_T,Output,,,
GateI_DC_3_Size,int,Input,10000,,
DrainV_DC_4,D_ARRAY_T,Output,,,
DrainV_DC_4_Size,int,Input,10000,,
DrainI_DC_4,D_ARRAY_T,Output,,,
DrainI_DC_4_Size,int,Input,10000,,
GateV_DC_4,D_ARRAY_T,Output,,,
GateV_DC_4_Size,int,Input,10000,,
GateI_DC_4,D_ARRAY_T,Output,,,
GateI_DC_4_Size,int,Input,10000,,
DrainV_DC_5,D_ARRAY_T,Output,,,
DrainV_DC_5_Size,int,Input,10000,,
DrainI_DC_5,D_ARRAY_T,Output,,,
DrainI_DC_5_Size,int,Input,10000,,
GateV_DC_5,D_ARRAY_T,Output,,,
GateV_DC_5_Size,int,Input,10000,,
GateI_DC_5,D_ARRAY_T,Output,,,
GateI_DC_5_Size,int,Input,10000,,
```

```
DrainV_DC_6,D_ARRAY_T,Output,,,
DrainV_DC_6_Size,int,Input,10000,,
DrainI_DC_6,D_ARRAY_T,Output,,,
DrainI_DC_6_Size,int,Input,10000,,
GateV_DC_6,D_ARRAY_T,Output,,,
GateV_DC_6_Size,int,Input,10000,,
GateI_DC_6,D_ARRAY_T,Output,,,
GateI_DC_6_Size,int,Input,10000,,
DrainV_DC_7,D_ARRAY_T,Output,,,
DrainV_DC_7_Size,int,Input,10000,,
DrainI_DC_7,D_ARRAY_T,Output,,,
DrainI_DC_7_Size,int,Input,10000,,
GateV_DC_7,D_ARRAY_T,Output,,,
GateV_DC_7_Size,int,Input,10000,,
GateI_DC_7,D_ARRAY_T,Output,,,
GateI_DC_7_Size,int,Input,10000,,
DrainV_DC_8,D_ARRAY_T,Output,,,
DrainV_DC_8_Size,int,Input,10000,,
DrainI_DC_8,D_ARRAY_T,Output,,,
DrainI_DC_8_Size,int,Input,10000,,
GateV_DC_8,D_ARRAY_T,Output,,,
GateV_DC_8_Size,int,Input,10000,,
GateI_DC_8,D_ARRAY_T,Output,,,
GateI_DC_8_Size,int,Input,10000,,
DrainV_DC_9,D_ARRAY_T,Output,,,
DrainV_DC_9_Size,int,Input,10000,,
DrainI_DC_9,D_ARRAY_T,Output,,,
DrainI_DC_9_Size,int,Input,10000,,
GateV_DC_9,D_ARRAY_T,Output,,,
GateV_DC_9_Size,int,Input,10000,,
GateI_DC_9,D_ARRAY_T,Output,,,
GateI_DC_9_Size,int,Input,10000,,
DrainV_DC_10,D_ARRAY_T,Output,,,
DrainV_DC_10_Size,int,Input,10000,,
DrainI_DC_10,D_ARRAY_T,Output,,,
DrainI_DC_10_Size,int,Input,10000,,
GateV_DC_10,D_ARRAY_T,Output,,,
GateV_DC_10_Size,int,Input,10000,,
GateI_DC_10,D_ARRAY_T,Output,,,
GateI_DC_10_Size,int,Input,10000,,
DrainV_Pulse_1,D_ARRAY_T,Output,,,
DrainV_Pulse_1_Size,int,Input,10000,,
DrainI_Pulse_1,D_ARRAY_T,Output,,,
DrainI_Pulse_1_Size,int,Input,10000,,
GateV_Pulse_1,D_ARRAY_T,Output,,,
```

GateV_Pulse_1_Size,int,Input,10000,,
GateI_Pulse_1,D_ARRAY_T,Output,,,
GateI_Pulse_1_Size,int,Input,10000,,
DrainV_Pulse_2,D_ARRAY_T,Output,,,
DrainV_Pulse_2_Size,int,Input,10000,,
DrainI_Pulse_2,D_ARRAY_T,Output,,,
DrainI_Pulse_2_Size,int,Input,10000,,
GateV_Pulse_2,D_ARRAY_T,Output,,,
GateV_Pulse_2_Size,int,Input,10000,,
GateI_Pulse_2,D_ARRAY_T,Output,,,
GateI_Pulse_2_Size,int,Input,10000,,
DrainV_Pulse_3,D_ARRAY_T,Output,,,
DrainV_Pulse_3_Size,int,Input,10000,,
DrainI_Pulse_3,D_ARRAY_T,Output,,,
DrainI_Pulse_3_Size,int,Input,10000,,
GateV_Pulse_3,D_ARRAY_T,Output,,,
GateV_Pulse_3_Size,int,Input,10000,,
GateI_Pulse_3,D_ARRAY_T,Output,,,
GateI_Pulse_3_Size,int,Input,10000,,
DrainV_Pulse_4,D_ARRAY_T,Output,,,
DrainV_Pulse_4_Size,int,Input,10000,,
DrainI_Pulse_4,D_ARRAY_T,Output,,,
DrainI_Pulse_4_Size,int,Input,10000,,
GateV_Pulse_4,D_ARRAY_T,Output,,,
GateV_Pulse_4_Size,int,Input,10000,,
GateI_Pulse_4,D_ARRAY_T,Output,,,
GateI_Pulse_4_Size,int,Input,10000,,
DrainV_Pulse_5,D_ARRAY_T,Output,,,
DrainV_Pulse_5_Size,int,Input,10000,,
DrainI_Pulse_5,D_ARRAY_T,Output,,,
DrainI_Pulse_5_Size,int,Input,10000,,
GateV_Pulse_5,D_ARRAY_T,Output,,,
GateV_Pulse_5_Size,int,Input,10000,,
GateI_Pulse_5,D_ARRAY_T,Output,,,
GateI_Pulse_5_Size,int,Input,10000,,
DrainV_Pulse_6,D_ARRAY_T,Output,,,
DrainV_Pulse_6_Size,int,Input,10000,,
DrainI_Pulse_6,D_ARRAY_T,Output,,,
DrainI_Pulse_6_Size,int,Input,10000,,
GateV_Pulse_6,D_ARRAY_T,Output,,,
GateV_Pulse_6_Size,int,Input,10000,,
GateI_Pulse_6,D_ARRAY_T,Output,,,
GateI_Pulse_6_Size,int,Input,10000,,
DrainV_Pulse_7,D_ARRAY_T,Output,,,
DrainV_Pulse_7_Size,int,Input,10000,,

DrainI_Pulse_7,D_ARRAY_T,Output,,,
DrainI_Pulse_7_Size,int,Input,10000,,
GateV_Pulse_7,D_ARRAY_T,Output,,,
GateV_Pulse_7_Size,int,Input,10000,,
GateI_Pulse_7,D_ARRAY_T,Output,,,
GateI_Pulse_7_Size,int,Input,10000,,
DrainV_Pulse_8,D_ARRAY_T,Output,,,
DrainV_Pulse_8_Size,int,Input,10000,,
DrainI_Pulse_8,D_ARRAY_T,Output,,,
DrainI_Pulse_8_Size,int,Input,10000,,
GateV_Pulse_8,D_ARRAY_T,Output,,,
GateV_Pulse_8_Size,int,Input,10000,,
GateI_Pulse_8,D_ARRAY_T,Output,,,
GateI_Pulse_8_Size,int,Input,10000,,
DrainV_Pulse_9,D_ARRAY_T,Output,,,
DrainV_Pulse_9_Size,int,Input,10000,,
DrainI_Pulse_9,D_ARRAY_T,Output,,,
DrainI_Pulse_9_Size,int,Input,10000,,
GateV_Pulse_9,D_ARRAY_T,Output,,,
GateV_Pulse_9_Size,int,Input,10000,,
GateI_Pulse_9,D_ARRAY_T,Output,,,
GateI_Pulse_9_Size,int,Input,10000,,
DrainV_Pulse_10,D_ARRAY_T,Output,,,
DrainV_Pulse_10_Size,int,Input,10000,,
DrainI_Pulse_10,D_ARRAY_T,Output,,,
DrainI_Pulse_10_Size,int,Input,10000,,
GateV_Pulse_10,D_ARRAY_T,Output,,,
GateV_Pulse_10_Size,int,Input,10000,,
GateI_Pulse_10,D_ARRAY_T,Output,,,
GateI_Pulse_10_Size,int,Input,10000,,

Description:

The Vd_Id_Pulse_DC_Family_QPulseIV sweep is used to perform a Pulsed vs. DC Vd-Id sweep using the Model 4200-PIV-Q package. This test is similar to a typical Vd-Id but only two sources are used: one for the DUT Gate and one for the DUT Drain. Pulsed Measurements are made with the 2 channel scope, Model 4200-SCP2HR.

To create a family of curves, choose an appropriate start and stop value for Vgs, and a number of steps. This routine can run the sweeps in three different ways: 1) DC only; 2) Pulse only; 3) Pulse and DC curves. This routine supports from 1 to 10 Vd-Id curves based on up to ten different Vgs values.

All voltage levels specified below assume a 50 ohm DUT load.

Inputs:

VgStart	(double) The starting step value for Vg. For DC only sweeps, VgStart must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VgStart must be between -5V to +5V or -20V to +20V depending on the specified source range for the PG2.
VgStop	(double) The final step value for Vg. For DC only sweeps, VgStop must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VgStart must be between -5V to +5V or -20V to +20V depending on the specified source range for the PG2.
VgNumSteps	(double) The number of steps for Vg (Max = 10).
VdStart	(double) The starting sweep value for Vd. For DC only sweeps, VgStart must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VdStart must be between -10V to +10V or -40V to +40V depending on the specified source range for the Model 4205-PCU.
VdStop	(double) The final sweep value for Vd. For DC only sweeps, VdStop must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VgStart must be between -10V to +10V or -40V to +40V depending on the specified source range for the Model 4205-PCU.
VdStep	(double) The number of steps for the Vd sweep. (Max = 10000)
VgQPoint	(double) The base value, or bias point, of the pulsed gate source. VgQPoint is range dependent and must be between -5V to +5V or -20V to +20V.
VdQPoint	(double) The base value, or bias point, of the pulsed drain sweep. VdQPoint is range dependent and must be between -10V to +10V or -40V to +40V.
PulseWidth	(double) The Vgs and Vds pulse width (PW). The Pulse Width range: 300ns to (1s - 10 ns) in 10ns resolution steps.
PulsePeriod	(double) The pulse period for Vgs and Vds. Minimum period is (FallTime/2) + (RiseTime/2) + PulseWidth + 10ns. Maximum period is 1s.
RiseTime	(double) The transition time from the Qpoint to the pulse value for both Vgs and Vds. The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and fall time.
FallTime	(double) The transition time from the pulse to the Qpoint value for both Vgs and Vds. The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and fall time.
Average	(int) The number of pulses to average (recommend 100 for a reasonable result, use larger for lower current measurements).
GateVPURange	(double) The source range for gate side PG2. Valid ranges are 5V (High Speed) and 20V (High Voltage).
DrainVPURange	(double) The source range for the drain side Model 4205-PCU. Valid ranges are 10V (High Speed) and 40V (High Voltage).

GateSMURange (int) The current measurement range to be used for the SMU on the DUT Gate terminal. Values correspond to the table below. Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents.:

1 Full Auto
2 Limited Auto 10pA
3 Limited Auto 100pA
4 Limited Auto 1nA
5 Limited Auto 10nA
6 Limited Auto 100nA
7 Limited Auto 1uA
8 Limited Auto 10uA
9 Limited Auto 100uA
10 Limited Auto 1mA
11 Limited Auto 10mA
12 Limited Auto 100mA

DrainSMURange (int) The current measurement range to be used for the SMU on the DUT Drain terminal. Values correspond to the table below. Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents:

1 Full Auto
2 Limited Auto 10pA
3 Limited Auto 100pA
4 Limited Auto 1nA
5 Limited Auto 10nA
6 Limited Auto 100nA
7 Limited Auto 1uA
8 Limited Auto 10uA
9 Limited Auto 100uA
10 Limited Auto 1mA
11 Limited Auto 10mA
12 Limited Auto 100mA

GateScpRange (double) The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).

DrainScpRange (double) The voltage measure range for the scope channel measuring the Drain. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).

GateLoadLine (int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Gate. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).

DrainLoadLine (int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Drain. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).

GateCompliance	(double) The current compliance for the DUT Gate. This supplied value is used for the maximum source current for both DC and pulse source modes. For pulse current compliance, the supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load.
DrainCompliance	(double) The current compliance for DUT Drain. This supplied value is used for the maximum source current for both DC and pulse source modes. For pulse current compliance, the supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load.
MaxIg	(double) The max current allowed for the DUT Gate, for both DC and pulse sweeps. If the Gate current becomes greater than MaxIg, the present sweep will stop and the test will continue with the next Vg step.
MaxId	(double) The max current allowed on the DUT Drain, for both DC and pulse sweeps. If the Drain current becomes greater than MaxId, the present sweep will stop and the test will continue with the next Vg step.
MaxPowerGate	(double) The max power allowed to the DUT Gate. If the Gate power becomes greater than the MaxPowerGate, the present sweep will stop and the test will continue with the next Vg step.
MaxPowerDrain	(double) The max power allowed to the DUT Drain. If the Drain power becomes greater than the MaxPowerDrain, the present sweep will stop and the test will continue with the next Vg step.
NPLC	(double) The DC measurement integration time in NPLC (Number of Power Line cycles).
DCSweepDelay	(double) Time, in seconds, between the DC source and measure for each sweep point.
DC_vs_Pulse	(int) Determines whether to run a DC and Pulse test or a DC only test or a Pulse only test.0 - Pulse Only, 1 - DC Only, 2 - DC and Pulse.
GateSMU	(char*) String representing the ID for the SMU connected to the DUT Gate.
DrainSMU	(char*) String representing the ID for the SMU connected to the DUT Drain.
GateVPU	(char*) String representing the ID for the PG2 connected to the DUT Gate.
DrainVPUHigh	(char*) String representing the top VPU on the Model 4205-PCU for the DUT Drain. This is the card on the right (or the card with the lower ID number), when facing the back of the system.
DrainVPULow	(char*) String representing the bottom VPU on the Model 4205-PCU for the DUT Drain. This is the card on the left (or the card with the higher ID number), when facing the back of the system.
DrainV_DC_X_Size DrainI_DC_X_Size GateV_DC_X_Size GateI_DC_X_Size DrainV_Pulse_X_Size DrainI_Pulse_X_Size GateV_Pulse_X_Size GateI_Pulse_X_Size	(int) Sizes of the output arrays. All arrays should be the same size, and need to be large enough to hold all sweep points.

Outputs:

DrainV_DC_X DrainV_Pulse_X	(double) Array of programmed drain voltage values.
DrainI_DC_X DrainI_Pulse_X	(double) Array of measured drain currents.
GateV_DC_X GateV_Pulse_X	(double) Array of measured gate voltages.

GateI_DC_X (double) Array of measured gate currents.
GateI_Pulse_X

Vd_Id_Single_DC_QPulseIV

Module Return Type: int
Number of Parameters: 28

Arguments:

Vgs,double,Input,2,,
VdStart,double,Input,0,,
VdStop,double,Input,4,,
VdStep,double,Input,.05,,
GateRange,int,Input,1,,
GateCompliance,double,Input,.1,,
DrainRange,int,Input,1,,
DrainCompliance,double,Input,.1,,
NPLC,double,Input,.01,,
SweepDelay,double,Input,0,,
MaxIg,double,Input,1,,
MaxId,double,Input,1,,
MaxPowerGate,double,Input,200,,
MaxPowerDrain,double,Input,200,,
GateSMU,char *,Input,"SMU1",,
DrainSMU,char *,Input,"SMU2",,
GateVPU,char *,Input,"VPU1",,
DrainVPUHigh,char *,Input,"VPU2",,
DrainVPULow,char *,Input,"VPU3",,
DrainV,D_ARRAY_T,Output,,,
DrainVSize,int,Input,10000,,
DrainI,D_ARRAY_T,Output,,,
DrainISize,int,Input,10000,,
GateV,D_ARRAY_T,Output,,,
GateVSize,int,Input,10000,,
GateI,D_ARRAY_T,Output,,,
GateISize,int,Input,10000,,
PostData,int,Input,1,,

Description:

The Vd_Id_Single_DC_QPulseIV sweep is used to perform a single DC Vd-Id sweep using the Model 4200-PIV-Q package. This test is similar to a typical DC Vd-Id but only two DC sources are used: one for the DUT Gate and one for the DUT Drain.

To create a family of curves, either change Vgs and run the test using append or use the Vd_Id_Pulse_DC_Family_QPulseIV function.

Inputs:

Vgs	(double) The DC gate-source voltage bias. Vgs must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT.
VdStart	(double) The starting sweep value for Vd. VdStart must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT.
VdStop	(double) The final sweep value for Vd. For DC only sweeps, VdStop must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT.
VdStep	(double) The number of steps for the Vd sweep. (Max = 10000)
GateRange	(int) The current measurement range to be used for the SMU on the DUT Gate terminal. Values correspond to the table below. Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents. 1 Full Auto 2 Limited Auto 10pA 3 Limited Auto 100pA 4 Limited Auto 1nA 5 Limited Auto 10nA 6 Limited Auto 100nA 7 Limited Auto 1uA 8 Limited Auto 10uA 9 Limited Auto 100uA 10 Limited Auto 1mA 11 Limited Auto 10mA 12 Limited Auto 100mA
GateCompliance	(double) The current compliance for the DUT Gate.
DrainRange	(int) The current measurement range to be used for the SMU on the DUT Drain terminal. Values correspond to the table below. Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents. 1 Full Auto 2 Limited Auto 10pA 3 Limited Auto 100pA 4 Limited Auto 1nA 5 Limited Auto 10nA 6 Limited Auto 100nA 7 Limited Auto 1uA 8 Limited Auto 10uA 9 Limited Auto 100uA 10 Limited Auto 1mA 11 Limited Auto 10mA 12 Limited Auto 100mA
DrainCompliance	(double) The current compliance for the DUT Drain.
NPLC	(double) The DC measurement integration time in NPLC (Number of Power Line cycles).

SweepDelay	(double) Time, in seconds, between the DC source and measure for each sweep point.
MaxIg	(double) The max current allowed on the DUT Gate. If the Gate current becomes greater than MaxIg, the test will exit.
MaxId	(double) The max current allowed on the DUT Drain. If the Drain current becomes greater than MaxId, the test will exit.
MaxPowerGate	(double) The max power allowed to the DUT Gate. If the gate power becomes greater than the MaxPowerGate, the test will exit.
MaxPowerDrain	(double) The max power allowed on the DUT Drain. If the Drain power becomes greater than the MaxPowerDrain, the test will exit.
GateSMU	(char*) String representing the ID for the SMU connected to the DUT Gate terminal.
DrainSMU	(char*) String representing the ID for the SMU connected to the DUT Drain terminal.
GateVPU	(char*) String representing the ID for the PG2 connected to the DUT Gate terminal.
DrainVPUHigh	(char*) String representing the top VPU on the Model 4205-PCU. This is the card on the right (or the card with the lower ID number), when facing the back of the system.
DrainVPULow	(char*) String representing the bottom VPU on the Model 4205-PCU. This is the card on the left (or the card with the higher ID number), when facing the back of the system.
DrainVSize DrainISize GateVSize GateISize	(int) Sizes of the output arrays. All arrays should be the same size and need to be large enough to hold all sweep points.
PostData	(int) Turns on real time graphing. (1 = Real Time Graphing, 0 = No Real Time Graphing)

Outputs:

DrainV	(double) Array of programmed drain voltage values.
DrainI	(double) Array of measured drain currents.
GateV	(double) Array of measured gate voltages.
GateI	(double) Array of measured gate currents.

Vd_Id_Single_Pulse_QPulseIV

Module Return Type: int
Number of Parameters: 37

Arguments:

Vgs,double,Input,2,,
VgQPoint,double,Input,0,,
VdStart,double,Input,0,,
VdStop,double,Input,4,,
VdStep,double,Input,.05,,
VdQPoint,double,Input,0,,
PulseWidth,double,Input,300e-9,,
PulsePeriod,double,Input,1e-6,,
RiseTime,double,Input,100e-9,,

FallTime,double,Input,100e-9,,
Average,int,Input,100,,
GateVPURange,double,Input,5,,
DrainVPURange,double,Input,5,,
GateScpRange,double,Input,0,,
DrainScpRange,double,Input,0,,
GateCompliance,double,Input,.1,,
DrainCompliance,double,Input,.1,,
GateLoadLine,int,Input,1,,
DrainLoadLine,int,Input,1,,
MaxIg,double,Input,1,,
MaxId,double,Input,1,,
MaxPowerGate,double,Input,200,,
MaxPowerDrain,double,Input,200,,
GateSMU,char *,Input,"SMU1",,
DrainSMU,char *,Input,"SMU2",,
GateVPU,char *,Input,"VPU1",,
DrainVPUHigh,char *,Input,"VPU2",,
DrainVPULow,char *,Input,"VPU3",,
DrainV,D_ARRAY_T,Output,,,
DrainVSize,int,Input,10000,,
DrainI,D_ARRAY_T,Output,,,
DrainISize,int,Input,10000,,
GateV,D_ARRAY_T,Output,,,
GateVSize,int,Input,10000,,
GateI,D_ARRAY_T,Output,,,
GateISize,int,Input,10000,,
PostData,int,Input,1,,

Description:

The Vd_Id_Single_Pulse_QPulseIV sweep is used to perform a single pulsed Vd-Id sweep using the Model 4200-PIV-Q package. This test is similar to a typical DC Vd-Id but only two sources are used: one for the DUT Gate and one for the DUT Drain. Pulsed Measurements are made with the 2 channel scope, Model 4200-SCP2HR.

To create a family of curves, either change Vgs and run the test using append or use the Vd_Id_FET_Family_QPulseIV function.

Inputs:

Vgs	(double) The pulsed Gate-Source voltage bias. Vgs is range dependent and can be between -5V to +5V or between -20V to +20V.
VgQPoint	(double) The base, or bias point, of the pulsed Gate source. VgQPoint is range dependent and can be between -5V to +5V or between -20V to +20V.
VdStart	(double) The starting sweep value for Vd. VdStart is range dependent and must be between -10V to +10V or between -40V to +40V.
VdStop	(double) The final sweep value for Vd. VdStop is range dependent and must be between-10V to +10V or between -40V to +40V.
VdStep	(double) The number of steps for the Vd sweep. (Max = 10000)

VdQPoint	(double) The base value, or bias point, of the pulsed drain sweep. VdQPoint is range dependent and must be between -10V to +10V or -40V to +40V.
PulseWidth	(double) The Vgs and Vds pulse width (PW). The Pulse Width range: 300ns to (1s - 10 ns) in 10ns resolution steps.
PulsePeriod	(double) The pulse period for Vgs and Vds. Minimum period is (FallTime/2) + (RiseTime/2) + PulseWidth + 10ns. Maximum period is 1s.
RiseTime	(double) The transition time from the Qpoint to the pulse value for both Vgs and Vds. The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and fall time.
FallTime	(double) The transition time from the pulse to the Qpoint value for both Vgs and Vds. The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and fall time.
Average	(int) The number of pulses to average (recommend 100 for a reasonable result, use larger for lower current measurements).
GateVPURange	(double) The source range for gate side PG2. Valid ranges are 5V (High Speed) and 20V (High Voltage).
DrainVPURange	(double) The source range for the drain side Model 4205-PCU. Valid ranges are 10V (High Speed) and 40V (High Voltage).
GateScpRange	(double) The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).
DrainScpRange	(double) The voltage measure range for the scope channel measuring the Drain. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).
GateCompliance	(double) The current compliance for the pulse source on the DUT Gate. This supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load. The compliance can be used to protect the DUT.
DrainCompliance	(double) The current compliance for the pulse source on the DUT Drain. This supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load. The compliance can be used to protect the DUT.
GateLoadLine	(int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Gate. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).
DrainLoadLine	(int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Drain. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values.

	When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).
MaxIg	(double) The max current allowed for the DUT Gate. If the Gate current becomes greater than MaxIg, the test will exit.
MaxId	(double) The max current allowed for the DUT Drain. If the Drain current becomes greater than MaxId, the test will exit.
MaxPowerGate	(double) The max power allowed to the DUT Gate. If the Gate power becomes greater than the MaxPowerGate, the test will exit.
MaxPowerDrain	(double) The max power allowed on the DUT Drain. If the Drain power becomes greater than the MaxPowerDrain, the test will exit.
GateSMU	(char*) String representing the ID for the SMU connected to the DUT Gate terminal.
DrainSMU	(char*) String representing the ID for the SMU connected to the DUT Drain terminal.
GateVPU	(char*) String representing the ID for the PG2 connected to the DUT Gate terminal.
DrainVPUHigh	(char*) String representing the top VPU on the Model 4205-PCU. This is the card on the right (or the card with the lower ID number), when facing the back of the system.
DrainVPULow	(char*) String representing the bottom VPU on the Model 4205-PCU. This is the card on the left (or the card with the higher ID number), when facing the back of the system.
DrainVSize DrainISize GateVSize GateISize	(int) Sizes of the output arrays. All arrays should be the same size and need to be large enough to hold all sweep points.
PostData	(int) Turns on real time graphing. (1 = Real Time Graphing, 0 = No Real Time Graphing)

Outputs:

DrainV	(double) Array of programmed drain voltage values.
DrainI	(double) Array of measured drain currents.
GateV	(double) Array of measured gate voltages.
GateI	(double) Array of measured gate currents.

Vg_Id_Pulse_DC_QPulseIV

Module Return Type: int
Number of Parameters: 49

Arguments:

Vd,double,Input,2,,
VgStart,double,Input,0,,
VgStop,double,Input,4,,
VgStep,double,Input,.05,,
VgQPoint,double,Input,0,,
VdQPoint,double,Input,0,,
PulseWidth,double,Input,300e-9,,
PulsePeriod,double,Input,1e-6,,
RiseTime,double,Input,100e-9,,

```
FallTime,double,Input,100e-9,,
AverageNum,int,Input,100,,
GateVPURange,double,Input,5,,
DrainVPURange,double,Input,5,,
GateSMURange,int,Input,1,,
DrainSMURange,int,Input,1,,
GateScpRange,double,Input,0,,
DrainScpRange,double,Input,0,,
GateLoadLineCorrection,int,Input,1,,
DrainLoadLineCorrection,int,Input,1,,
GateCompliance,double,Input,.1,,
DrainCompliance,double,Input,.1,,
MaxIg,double,Input,1,,
MaxId,double,Input,1,,
MaxPowerGate,double,Input,200,,
MaxPowerDrain,double,Input,200,,
NPLC,double,Input,.01,,
DCSourceDelay,double,Input,0,,
DC_vs_Pulse,int,Input,2,,
GateSMU,char *,Input,"SMU1",,
DrainSMU,char *,Input,"SMU2",,
GateVPU,char *,Input,"VPU1",,
DrainVPUHigh,char *,Input,"VPU2",,
DrainVPULow,char *,Input,"VPU3",,
DrainV_DC,D_ARRAY_T,Output,,,
DrainV_DC_Size,int,Input,10000,,
DrainI_DC,D_ARRAY_T,Output,,,
DrainI_DC_Size,int,Input,10000,,
GateV_DC,D_ARRAY_T,Output,,,
GateV_DC_Size,int,Input,10000,,
GateI_DC,D_ARRAY_T,Output,,,
GateI_DC_Size,int,Input,10000,,
DrainV_Pulse,D_ARRAY_T,Output,,,
DrainV_Pulse_Size,int,Input,10000,,
DrainI_Pulse,D_ARRAY_T,Output,,,
DrainI_Pulse_Size,int,Input,10000,,
GateV_Pulse,D_ARRAY_T,Output,,,
GateV_Pulse_Size,int,Input,10000,,
GateI_Pulse,D_ARRAY_T,Output,,,
GateI_Pulse_Size,int,Input,10000,,
```

Description:

The Vg_Id_Pulse_DC_QPulseIV sweep is used to perform a Pulsed vs. DC Vg-Id sweep using the Model 4200-PIV-Q package. This test is similar to a typical Vg-Id but only two sources are used: one for the DUT Gate and one for the DUT Drain. Pulsed Measurements are made with the 2 channel scope, Model 4200-SCP2HR.

This routine can run the sweeps in three different ways: 1) DC only; 2) Pulse only; 3) Pulse and DC curves.

All voltage levels specified below assume a 50 ohm DUT load.

Inputs:

Vds	(double) The voltage value for Vd. For DC only sweeps, Vds must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, Vds must be between -10V to +10V or -40V to +40V depending on the specified source range for the Model 4205-PCU.
VgStart	(double) The starting sweep value for Vg. For DC only sweeps, VgStart must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VgStart must be between -5V to +5V or -20V to +20V depending on the specified source range for the PG2.
VgStop	(double) The final sweep value for Vg. For DC only sweeps, VgStop must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT. For pulse and pulse and DC Sweeps, VgStart must be between -5V to +5V or -20V to +20V depending on the specified source range for the PG2.
VgNumSteps	(double) The number of steps in the Vg sweep (Max = 10 000).
VgQPoint	(double) The base value, or bias point, of the pulsed gate sweep. VgQPoint is range dependent and must be between -5V to +5V or -20V to +20V.
VdQPoint	(double) The base value, or bias point, of the pulsed drain source. VdQPoint is range dependent and must be between -10V to +10V or -40V to +40V.
PulseWidth	(double) The Vgs and Vds pulse width (PW). The Pulse Width range: 300ns to (1s - 10 ns) in 10ns resolution steps.
PulsePeriod	(double) The pulse period for Vgs and Vds. Minimum period is (FallTime/2) + (RiseTime/2) + PulseWidth + 10ns. Maximum period is 1s.
RiseTime	(double) The transition time from the Qpoint to the pulse value for both Vgs and Vds. The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and fall time.
FallTime	(double) The transition time from the pulse to the Qpoint value for both Vgs and Vds. The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and fall time.
Average	(int) The number of pulses to average (recommend 100 for a reasonable result, use larger for lower current measurements).
GateVPURange	(double) The source range for gate side PG2. Valid ranges are 5V (High Speed) and 20V (High Voltage).
DrainVPURange	(double) The source range for the drain side Model 4205-PCU. Valid ranges are 10V (High Speed) and 40V (High Voltage).
GateSMURange	(int) The current measurement range to be used for the SMU on the DUT Gate terminal. Values correspond to the table below. Limited Auto means that

the value given is the minimum measurement range used, with automatic ranging for larger currents:

1 Full Auto
2 Limited Auto 10pA
3 Limited Auto 100pA
4 Limited Auto 1nA
5 Limited Auto 10nA
6 Limited Auto 100nA
7 Limited Auto 1uA
8 Limited Auto 10uA
9 Limited Auto 100uA
10 Limited Auto 1mA
11 Limited Auto 10mA
12 Limited Auto 100mA

DrainSMURange (int) The current measurement range to be used for the SMU on the DUT Drain terminal. Values correspond to the table below. Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents:

1 Full Auto
2 Limited Auto 10pA
3 Limited Auto 100pA
4 Limited Auto 1nA
5 Limited Auto 10nA
6 Limited Auto 100nA
7 Limited Auto 1uA
8 Limited Auto 10uA
9 Limited Auto 100uA
10 Limited Auto 1mA
11 Limited Auto 10mA
12 Limited Auto 100mA

GateScpRange (double) The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).

DrainScpRange (double) The voltage measure range for the scope channel measuring the Drain. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).

GateLoadLine (int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Gate. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).

DrainLoadLine (int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Drain. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).

GateCompliance	(double) The current compliance for the DUT Gate. For pulse current compliance, the supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load.
DrainCompliance	(double) The current compliance for DUT Drain. For pulse current compliance, the supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load.
MaxIg	(double) The max current allowed for the DUT Gate, for both DC and pulse sweeps. If the Gate current becomes greater than MaxIg, the present sweep will stop and the test will continue with the next Vg step.
MaxId	(double) The max current allowed on the DUT Drain, for both DC and pulse sweeps. If the Drain current becomes greater than MaxId, the present sweep will stop and the test will continue with the next Vg step.
MaxPowerGate	(double) The max power allowed to the DUT Gate. If the Gate power becomes greater than the MaxPowerGate, the present sweep will stop and the test will continue with the next Vg step.
MaxPowerDrain	(double) The max power allowed to the DUT Drain. If the Drain power becomes greater than the MaxPowerDrain, the present sweep will stop and the test will continue with the next Vg step.
NPLC	(double) The DC measurement integration time in NPLC (Number of Power Line cycles).
DCSweepDelay	(double) Time, in seconds, between the DC source and measure for each sweep point.
DC_vs_Pulse	(int) Determines whether to run a DC and Pulse test or a DC only test or a Pulse only test.0 - Pulse Only, 1 - DC Only, 2 - DC and Pulse.
GateSMU	(char*) String representing the ID for the SMU connected to the gate.
DrainSMU	(char*) String representing the ID for the SMU connected to the drain.
GateVPU	(char*) String representing the ID for the PG2 connected to the gate.
DrainVPUHigh	(char*) String representing the top VPU on the Model 4205-PCU. This is the card on the right (or the card with the lower ID number), when facing the back of the system.
DrainVPULow	(char*) String representing the bottom VPU on the Model 4205-PCU. This is the card on the left (or the card with the higher ID number), when facing the back of the system.
DrainV_DC_Size DrainI_DC_Size GateV_DC_X_Size GateI_DC_X_Size DrainV_Pulse_Size DrainI_Pulse_Size GateV_Pulse_X_Size GateI_Pulse_X_Size	(int) Sizes of the output arrays. All arrays should be the same size and need to be large enough to hold all sweep points.

Outputs:

DrainV_DC DrainV_Pulse	(double) Array of measured drain voltage values.
DrainI_DC DrainI_Pulse	(double) Array of measured drain currents.
GateV_DC_X GateV_Pulse_X	(double) Array of programmed gate voltages.

GateI_DC_X
GateI_Pulse_X
(double) Array of measured gate currents.

Vg_Id_Single_DC_QPulseIV

Module Return Type: int
Number of Parameters: 28

Arguments:

Vds,double,Input,3,,
VgStart,double,Input,0,,
VgStop,double,Input,2,,
VgStep,double,Input,.05,,
GateRange,int,Input,1,,
GateCompliance,double,Input,.1,,
DrainRange,int,Input,1,,
DrainCompliance,double,Input,.1,,
NPLC,double,Input,.01,,
SweepDelay,double,Input,0,,
MaxIg,double,Input,1,,
MaxId,double,Input,1,,
MaxPowerGate,double,Input,200,,
MaxPowerDrain,double,Input,200,,
GateSMU,char *,Input,"SMU1",,
DrainSMU,char *,Input,"SMU2",,
GateVPU,char *,Input,"VPU1",,
DrainVPUHigh,char *,Input,"VPU2",,
DrainVPULow,char *,Input,"VPU3",,
DrainV,D_ARRAY_T,Output,,,
DrainVSize,int,Input,10000,,
DrainI,D_ARRAY_T,Output,,,
DrainISize,int,Input,10000,,
GateV,D_ARRAY_T,Output,,,
GateVSize,int,Input,10000,,
GateI,D_ARRAY_T,Output,,,
GateISize,int,Input,10000,,
PostData,int,Input,1,,

Description:

The Vg_Id_Single_DC_QPulseIV sweep is used to perform a DC Vg-Id sweep using the Model 4200-PIV-Q package. This test is similar to a typical Vg-Id but only two sources are used: one for the DUT Gate and one for the DUT Drain.

All voltage levels specified below assume a 50 ohm DUT load.

Inputs:

Vds
(double) The voltage value for Vd. Vds must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT.

VgStart	(double) The starting sweep value for Vg. VgStart must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT.
VgStop	(double) The final sweep value for Vg. VgStop must be between -200V to +200V dependent on the type of SMU and the current requirements of the DUT.
VgStep	(double) The number of steps in the Vg sweep (Max = 10 000).
GateRange	(int) The current measurement range to be used for the SMU on the DUT Gate terminal. Values correspond to the table below. Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents: 1 Full Auto 2 Limited Auto 10pA 3 Limited Auto 100pA 4 Limited Auto 1nA 5 Limited Auto 10nA 6 Limited Auto 100nA 7 Limited Auto 1uA 8 Limited Auto 10uA 9 Limited Auto 100uA 10 Limited Auto 1mA 11 Limited Auto 10mA 12 Limited Auto 100mA
GateCompliance	(double) The current compliance for the DUT Gate.
DrainRange	(int) The current measurement range to be used for the SMU on the DUT Drain terminal. Values correspond to the table below. Limited Auto means that the value given is the minimum measurement range used, with automatic ranging for larger currents: 1 Full Auto 2 Limited Auto 10pA 3 Limited Auto 100pA 4 Limited Auto 1nA 5 Limited Auto 10nA 6 Limited Auto 100nA 7 Limited Auto 1uA 8 Limited Auto 10uA 9 Limited Auto 100uA 10 Limited Auto 1mA 11 Limited Auto 10mA 12 Limited Auto 100mA
DrainCompliance	(double) The current compliance for the DUT Drain.
NPLC	(double) The DC measurement integration time in NPLC (Number of Power Line cycles).
DCSweepDelay	(double) Time, in seconds, between the DC source and measure for each sweep point.
MaxIg	(double) The max current allowed on the DUT Gate. If the Gate current becomes greater than MaxIg, the test will exit.
MaxId	(double) The max current allowed on the DUT Drain. If the Drain current becomes greater than MaxId, the test will exit.
MaxPowerGate	(double) The max power allowed to the DUT Gate. If the gate power becomes greater than the MaxPowerGate, the test will exit.

MaxPowerDrain	(double) The max power allowed on the DUT Drain. If the Drain power becomes greater than the MaxPowerDrain, the test will exit.
GateSMU	(char*) String representing the ID for the SMU connected to the gate.
DrainSMU	(char*) String representing the ID for the SMU connected to the drain.
GateVPU	(char*) String representing the ID for the PG2 connected to the gate.
DrainVPUHigh	(char*) String representing the top VPU on the Model 4205-PCU. This is the card on the right (or the card with the lower ID number), when facing the back of the system.
DrainVPULow	(char*) String representing the bottom VPU on the Model 4205-PCU. This is the card on the left (or the card with the higher ID number), when facing the back of the system.
DrainVSize GateISize	(int) Sizes of the output arrays. Note that DrainISize all arrays should be the GateISize same size and need GateVSize to be large enough to hold all sweep points.
PostData	(int) Turns on real time graphing. (1 = Real Time Graphing, 0 = No Real Time Graphing)

Outputs:

DrainV	(double) Array of measured drain voltage values.
DrainI	(double) Array of measured drain currents.
GateV	(double) Array of programmed gate voltages.
GateI	(double) Array of measured gate currents.

Vg_Id_Single_Pulse_QPulseIV

Module Return Type: int
Number of Parameters: 37

Arguments:

Vds,double,Input,3,,
VdQPoint,double,Input,0,,
VgStart,double,Input,0,,
VgStop,double,Input,2,,
VgStep,double,Input,.05,,
VgQPoint,double,Input,0,,
PulseWidth,double,Input,300e-9,,
PulsePeriod,double,Input,1e-6,,
RiseTime,double,Input,100e-9,,
FallTime,double,Input,100e-9,,
Average,int,Input,10,,
GateVPURange,double,Input,5,,
DrainVPURange,double,Input,5,,
GateScpRange,double,Input,0,,
DrainScpRange,double,Input,0,,
GateCompliance,double,Input,.1,,
DrainCompliance,double,Input,.1,,
GateLoadLine,int,Input,1,,
DrainLoadLine,int,Input,1,,
MaxIg,double,Input,1,,

```
MaxId,double,Input,1,,
MaxPowerGate,double,Input,200,,
MaxPowerDrain,double,Input,200,,
GateSMU,char *,Input,"SMU1",,
DrainSMU,char *,Input,"SMU2",,
GateVPU,char *,Input,"VPU1",,
DrainVPUHigh,char *,Input,"VPU2",,
DrainVPULow,char *,Input,"VPU3",,
DrainV,D_ARRAY_T,Output,,,
DrainVSize,int,Input,10000,,
DrainI,D_ARRAY_T,Output,,,
DrainISize,int,Input,10000,,
GateV,D_ARRAY_T,Output,,,
GateVSize,int,Input,10000,,
GateI,D_ARRAY_T,Output,,,
GateISize,int,Input,10000,,
PostData,int,Input,1,,
```

Description:

The Vg_Id_Single_Pulse_QPulseIV sweep is used to perform a Pulsed Vg-Id sweep using the Model 4200-PIV-Q package. This test is similar to a typical Vg-Id but only two sources are used: one for the DUT Gate and one for the DUT Drain. Pulsed Measurements are made with the 2 channel scope, Model 4200-SCP2HR.

All voltage levels specified below assume a 50 ohm DUT load.

Inputs:

Vds	(double) The voltage value for Vd. Vds must be between -10V to +10V or -40V to +40V depending on the specified source range for the Model 4205-PCU.
VdQPoint	(double) The base value, or bias point, of the pulsed drain source. VdQPoint is range dependent and must be between -10V to +10V or -40V to +40V.
VgStart	(double) The starting sweep value for Vg. VgStart is range dependent and must be between -5V to +5V or between -20V to +20V.
VgStop	(double) The final sweep value for Vg. VgStop is range dependent and must be between -5V to +5V or between -20V to +20V.
VgStep	(double) The number of steps in the Vg sweep (Max = 10 000).
VgQPoint	(double) The base value, or bias point, of the pulsed gate sweep. VgQPoint is range dependent and must be between -5V to +5V or -20V to +20V.
PulseWidth	(double) The Vgs and Vds pulse width (PW). The Pulse Width can be 300ns to 1s (10ns resolution).
PulsePeriod	(double) The pulse period for Vgs and Vds. Minimum period is (FallTime/2) + (RiseTime/2) + PulseWidth + 10ns. Maximum period is 1s.
RiseTime	(double) The transition time from the Qpoint to the current pulse value for both Vgs and Vds. The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (although recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and fall time.

FallTime	(double) The transition time from the current value to the QPoint base value for both Vgs and Vds. The transition time is source range dependent. For the 5V range (10V for Model 4205-PCU) the minimum transition time is 10ns (although recommended minimum is 50ns) and for the 20V range (40V for the Model 4205-PCU) the minimum transition time is 100ns. The maximum transition time is dependent on the pulse width, period, and rise time.
Average	(int) The number of pulses to average (recommend 100 for a reasonable result, use larger for lower current measurements).
GateVPURange	(double) The source range for gate side PG2. Valid ranges are 5V (High Speed) and 20V (High Voltage).
DrainVPURange	(double) The source range for the drain side Model 4205-PCU. Valid ranges are 10V (High Speed) and 40V (High Voltage).
GateScpRange	(double) The voltage measure range for the scope channel measuring the Gate. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).
DrainScpRange	(double) The voltage measure range for the scope channel measuring the Drain. Use 0 for scope autoranging, or specify a voltage value for a fixed range. Valid voltages are 0.25, 0.5, 1.25, 2.5, 5, 10, 25, 50. The range is a full range value (for example, 2.5 is -1.25 V to +1.25 V).
GateCompliance	(double) The current compliance for the pulse source on the DUT Gate. This supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load. The compliance can be used to protect the DUT.
DrainCompliance	(double) The current compliance for the pulse source on the DUT Drain. This supplied value is used to calculate the maximum voltage to source to the DUT, based on a default 50 ohm DUT load. The compliance can be used to protect the DUT.
GateLoadLine	(int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Gate. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).
DrainLoadLine	(int) Determines whether to use load line correction to compensate for the voltage drop caused by the DUT impedance on the Drain. When load line correction is on (1), the test will start by assuming a high impedance value for the device and will approach the correct bias and pulse values over a series of pulses, that ensures that the sourced pulses match the requested values. When load line correction is turned off, the specified voltages will be sourced. (1 = Use Load Line, 0 = No Load Line).
MaxIg	(double) The max current allowed for the DUT Gate, for both DC and pulse sweeps. If the Gate current becomes greater than MaxIg, the present sweep will stop and the test will continue with the next Vg step.
MaxId	(double) The max current allowed on the DUT Drain, for both DC and pulse sweeps. If the Drain current becomes greater than MaxId, the present sweep will stop and the test will continue with the next Vg step.
MaxPowerGate	(double) The max power allowed to the DUT Gate. If the Gate power becomes greater than the MaxPowerGate, the test will exit.
MaxPowerDrain	(double) The max power allowed to the DUT Drain. If the Drain power becomes greater than the MaxPowerDrain, the test will exit.

GateSMU	(char*) String representing the ID for the SMU connected to the gate.
DrainSMU	(char*) String representing the ID for the SMU connected to the drain.
GateVPU	(char*) String representing the ID for the PG2 connected to the gate.
DrainVPUHigh	(char*) String representing the top VPU on the Model 4205-PCU. This is the card on the right (or the card with the lower ID number), when facing the back of the system.
DrainVPULow	(char*) String representing the bottom VPU on the Model 4205-PCU. This is the card on the left (or the card with the higher ID number), when facing the back of the system.
DrainVSize DrainISize GateVSize GateISize	(int) Sizes of the output arrays. All arrays should be the same size and need to be large enough to hold all sweep points.
PostData	(int) Turns on real time graphing. (1 = Real Time Graphing, 0 = No Real Time Graphing)

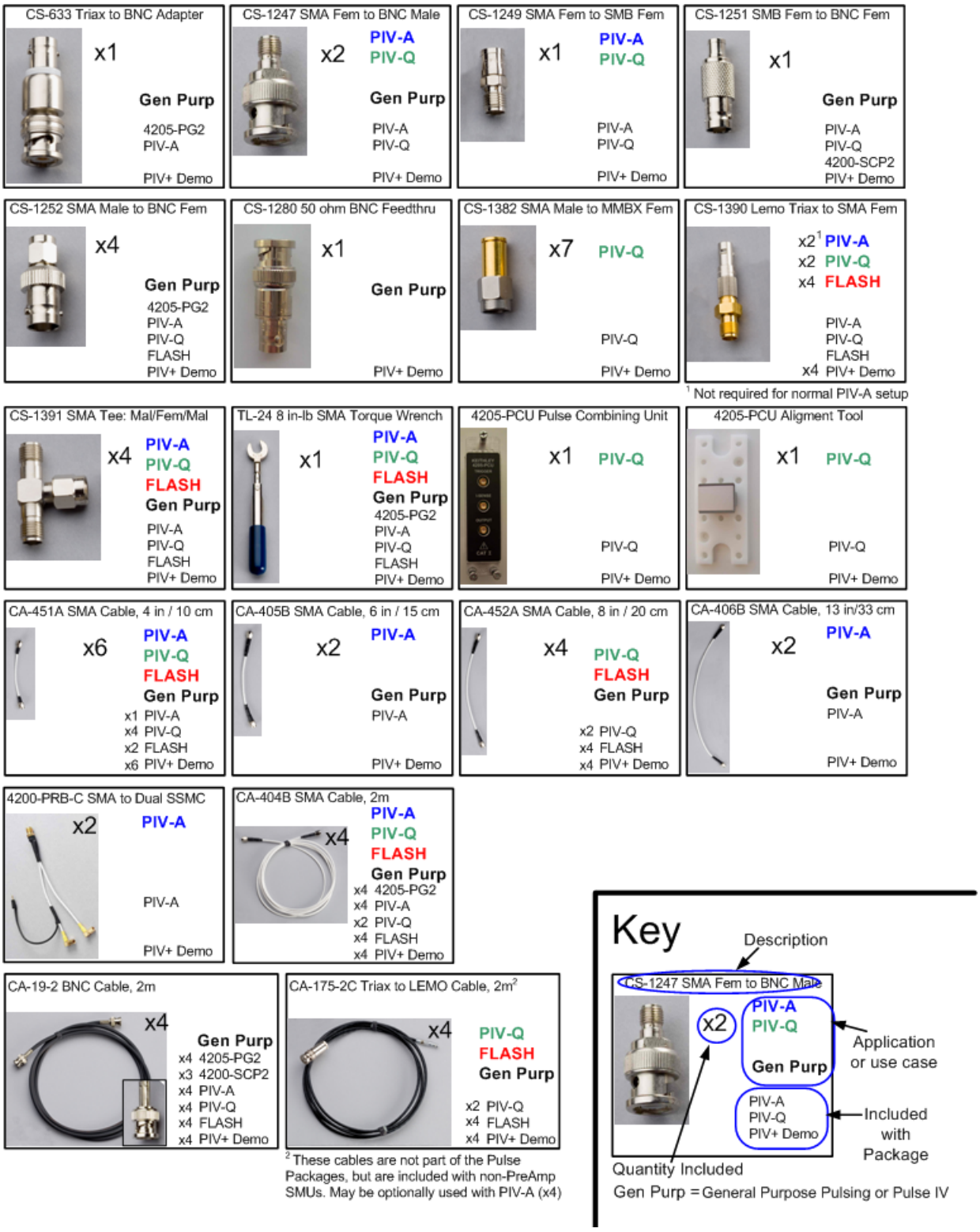
Outputs:

DrainV	(double) Array of measured drain voltage values.
DrainI	(double) Array of measured drain currents.
GateV	(double) Array of programmed gate voltages.
GateI	(double) Array of measured gate currents.

Pulse adapters, cables, hardware and PCU

The various adapters, cables and hardware used for the pulse projects are shown in Figure 4-50. The 4205-PCU that is used for the PIV-Q package is also shown.

Figure 4-50
Pulse adapters, cables and hardware



Flash Memory Testing

Introduction

There are several projects included with the Model 4200-FLASH package that facilitate testing of floating gate transistors (NOR, NAND), as well as other types of Non-volatile Memory (NVM). The package consists of two Model 4205-PG2 cards (four pulse channels), projects described in this section, and all required interconnecting cables and adapters. (see Figure 4-60)

Depending on the desired setup, at least two SMUs are required. To illustrate the flexibility of the Model 4200-FLASH pakage Figure 4-55 and Figure 4-56 depict a typical configuration using four SMUs. This configuration permits independent source and measure for each terminal in a typical four-terminal floating gate transistor.

NOTE *The 4200-FLASH package does not include any pulse measure capability, however, a 4200-SCP2 (2 channel scope) can be added for manual pulse height verification.*

Theory of operation

Programming and erasing flash memory

A floating gate transistor is simply a modified field-effect transistor with an additional floating gate. The floating gate transistor is the basic storage structure for data in non-volatile memory. The floating gate (FG) stores charge, that represents data in memory (see Figure 4-51). The control gate (CG) reads, programs, and erases the FG transistor. The presence of charge on the gate shifts the voltage threshold, V_T, to a higher voltage, as shown in Figure 4-52.

Figure 4-51
Cross section of a floating gate transistor in both the erased and programmed states

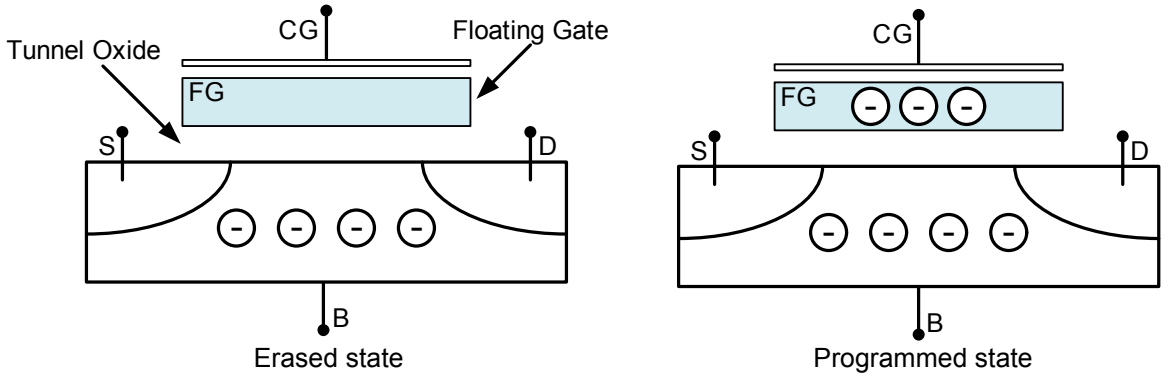
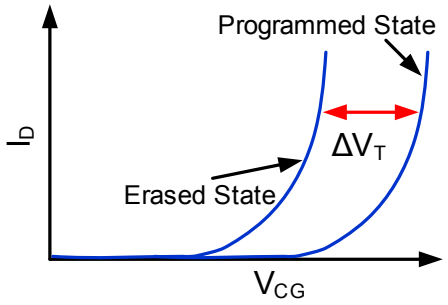


Figure 4-52
Graph of shifted voltage threshold, V_T, due to stored charge on floating gate on a 1 bit (2 level) cell.



The Flash transistors tests consist of two parts:

1. Pulse waveforms to program or erase the DUT
2. DC measurements are taken to determine the state of the device

This implies switching between two conditions:

1. Pulse resources are connected to the DUT
2. Pulse resources are disconnected and the DC resources are connected to the DUT

The pulses are used to move charge to or from the floating gate. There are two different methods to move charge:

1. tunneling
2. hot electron injection (HEI)

The “tunneling” method is commonly known as “Fowler-Nordheim (FN) tunneling,” or “quantum tunneling,” and is a function of the electric potential across the tunneling oxide (Figure 4-53). HEI is considered a damage mechanism in non-floating gate transistors, and is commonly called “hot charge injection (HCI).” HEI/HCI is a method that accelerates charges by applying a drain-source field, and then the charges are directed into the floating gate by a gate voltage.

Figure 4-53 shows examples of tunneling to move charge to and from the FG.

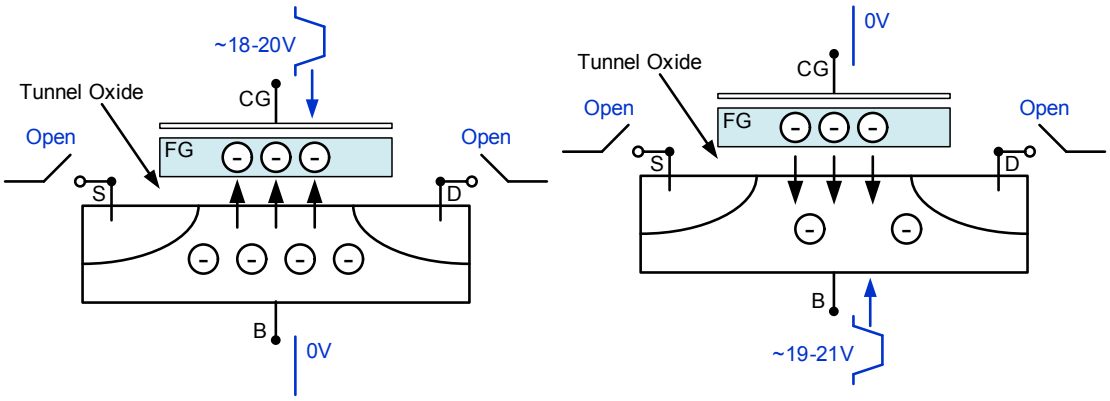
- The electric field and the preferred direction of electron flow are indicated by the black arrows.
- The signal applied to each device terminal are indicated by the blue text and blue features.

NOTE *Both the drain and source are not connected to any test instrumentation.*

This condition may also be called “floating” or “high impedance.” Figure 4-54 shows examples of moving charge using HEI. These conditions are only examples with approximate voltage values, and both pulse width and pulse height will vary depending on device structure and process details.

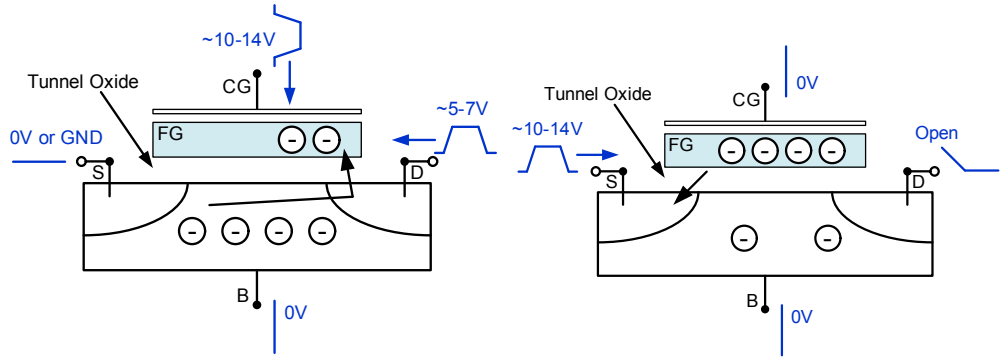
There are many other ways to provide similar electric fields and balance performance across a variety of parameters: program or erase speed, retention longevity, adjacent cell disturbance, endurance, and others.

Figure 4-53
Fowler-Nordheim tunneling Program and Erase.



Program using FN tunneling Erase using FN Tunneling

Figure 4-54
Hot Electron Injection (HEI) Program and Erase.



Program using hot electron injection Erase using HEI

The flash projects support two methods for performing the switching between the pulse and measure phases of the typical flash memory test. The first is the typical method, using a switch matrix to route the pulse or DC signals to the DUT (see Figure 4-55). Using the switch matrix is more complicated, but provides flexibility for certain tests and test structures that use arrays. Because both the SMUs and the Model 4205-PG2s have isolation relays located on the cards, it is possible to configure a simpler setup without the external switch matrix (see Figure 4-56 and Figure 4-60). The advantage of the simpler setup is lower cost, while the switch matrix approach provides lower current measurement performance and flexibility necessary for testing arrays of test structures.

To determine the state of the device, one would typically perform a Vg-Id sweep, then perform a calculation to find the voltage threshold, V_T. The shift in V_T represents a change in the amount of charge stored in the floating gate, that indicates the state of the cell, from fully programmed (1) to fully erased (0). The Model 4200-FLASH package does not include the ability to measure the pulse waveform or pulse response.

Figure 4-55
Block diagram of an example flash test setup using a switch matrix

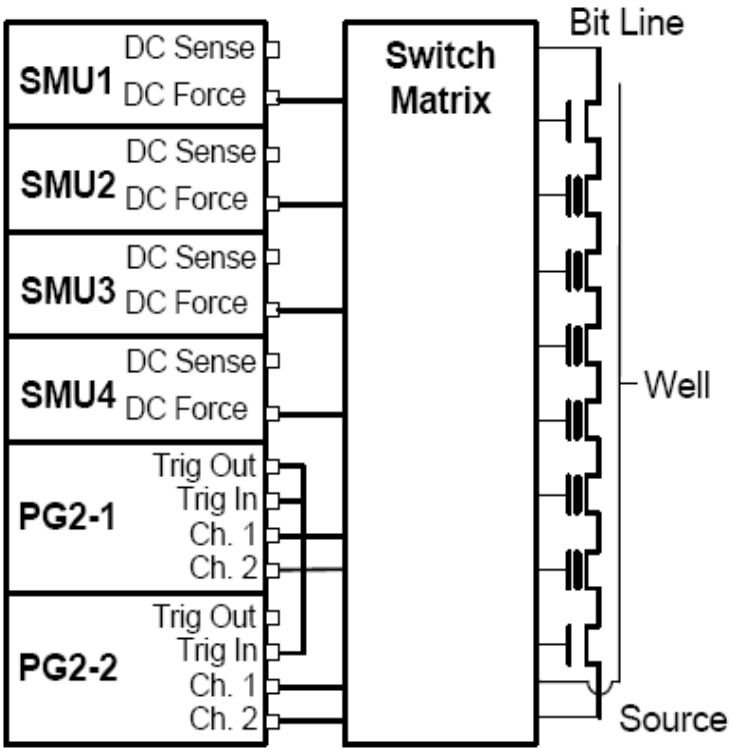
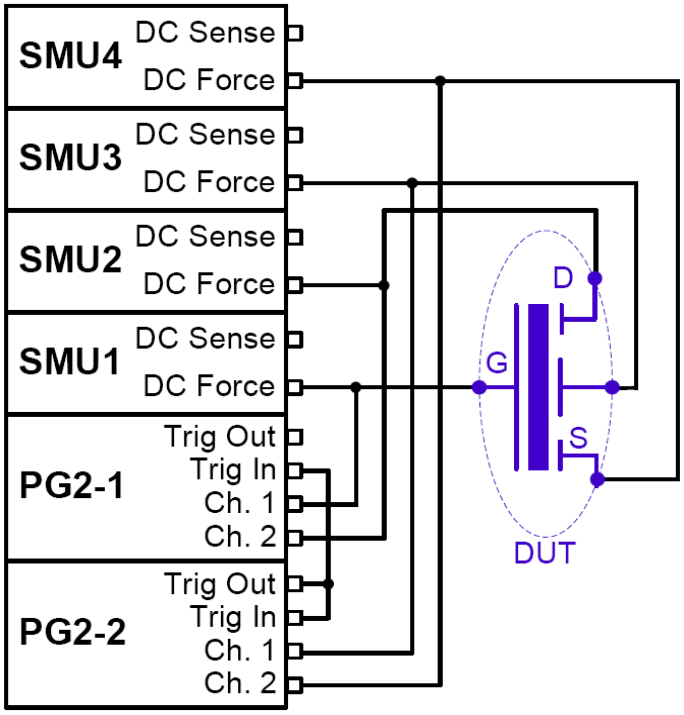


Figure 4-56
Block diagram of a flash test setup without using a switch matrix (direct connect)



The pulse waveforms are typically a program pulse (see Figure 4-57), an erase pulse (see Figure 4-58), or a waveform made up of both program and erase pulses (see Figure 4-59). All of these waveforms are implemented by using the Segment-Arb capability (see Section 11 of the

Model 4200-SCS Reference Manual). There are many different methods and voltage levels for programming and erasing, so these are only examples.

Figure 4-57
Program pulse waveforms for a floating gate DUT, with separate pulse waveforms for the DUT gate, drain, source, and bulk

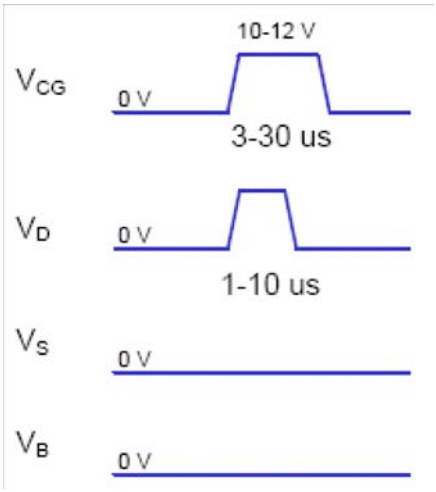


Figure 4-58
Example erase pulse waveforms for a floating gate DUT, with separate pulse waveforms for the DUT gate, drain, source, and bulk

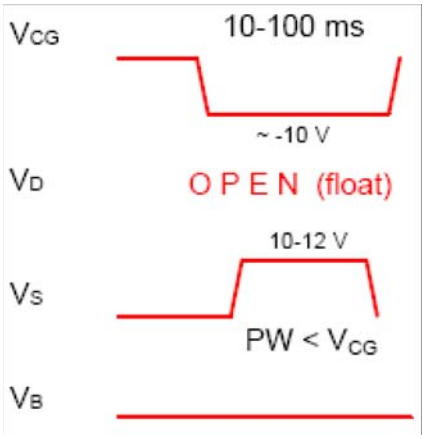
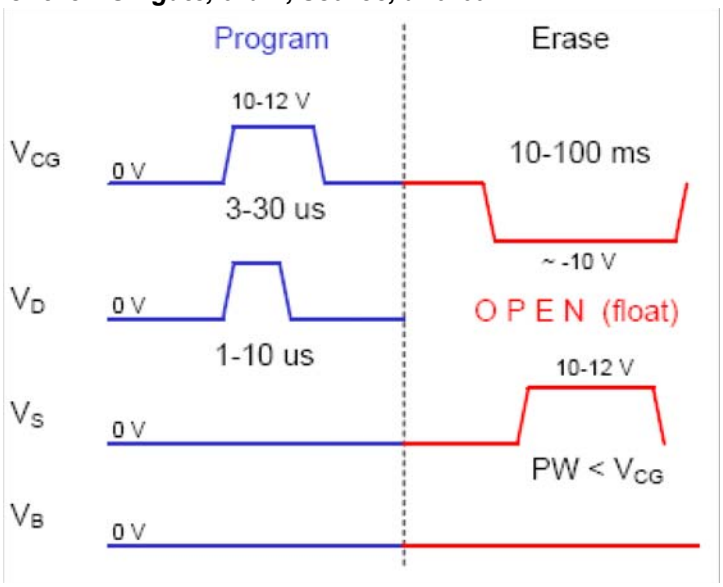
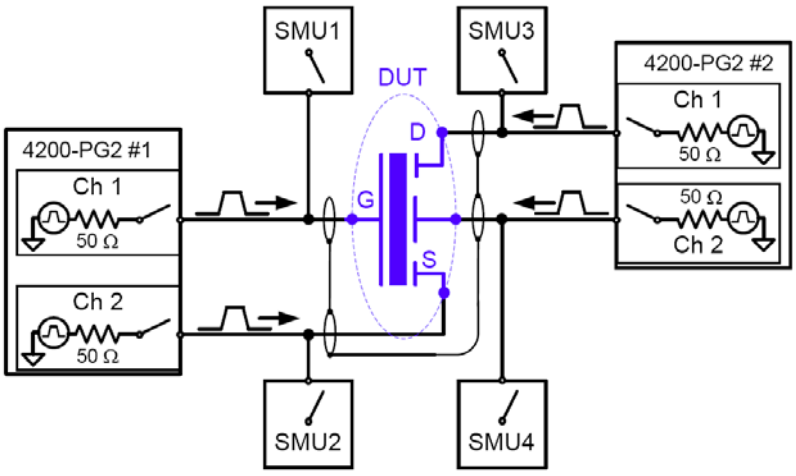


Figure 4-59
Program + Erase pulse waveforms for a floating gate DUT, with separate pulse waveforms for the DUT gate, drain, source, and bulk.



The block diagram for the FLASH setup is shown in Figure 4-60. Reconfiguring from the pulse stress to DC measure phases is done by activating the switches on the SMU and PG2 cards. During the pulse program/erase phase, the relays in the PG2 channels are closed and the relays in the SMUs are open. For the DC measure phase, the opposite is true.

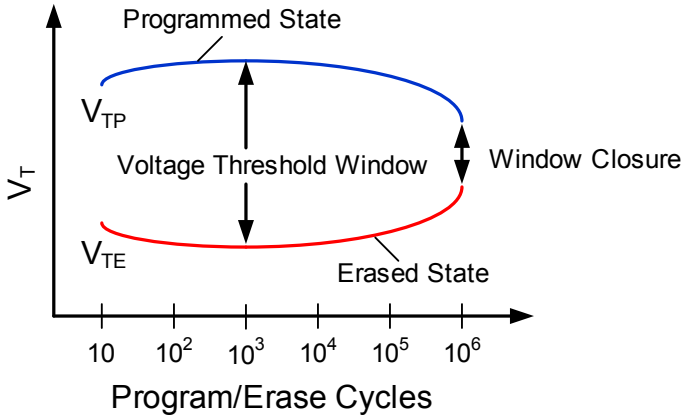
Figure 4-60
Basic schematic of flash testing without a switch matrix



Endurance testing

Endurance testing stresses the DUT with a number of Program+Erase waveform cycles, and then periodically measures both the voltage threshold in the programmed state V_{TP}, as well as the voltage threshold of the erased state, V_{TE}. The purpose of these tests is to determine the lifetime of the DUT, based on the number of Program+Erase cycles withstood by the device before a certain amount of shift, or degradation, in either the V_{TP} or V_{TE}, as shown in Figure 4-61. The endurance test is typically performed a set number of program and erase cycles (Figure 4-59), while periodically measuring V_T for both the programmed and erased state. Figure 4-61 shows typical degradation on a NOR cell for both V_{TP} and V_{TE} as the number of applied program/erase cycles increases.

Figure 4-61
Example results of V_T shift in an Endurance test on a NOR flash cell.



Disturb testing

The purpose of the Disturb test is to pulse stress a device in an array test structure, then perform a measurement, such as VT, on a device adjacent to the pulsed device. The goal is to measure the amount of VT shift in adjacent cells, either in the programmed or erased states, when a nearby device is pulsed with either a Program, Erase, or Program+Erase waveforms. The typical measurement is a VT extraction based on a Vg-Id sweep, but any type of DC test may be configured. This test is similar to the endurance test, but the pulsing and measuring are performed on adjacent devices.

Figure 4-62 shows an example configuration to pulse stress a device (Cell 2) and then test an adjacent device (Cell 1) in an array cell memory structure. The solid-line blue circle indicates the cell to be pulse stressed, and the dotted-line red circles are the adjacent memory cells that will be "disturbed" by the stressing. The stress/measure process is explained as follows.

Initial test conditions – SMU4 outputs a DC voltage to turn on the control devices for the array. This connects instrumentation at the top of array to the flash memory cells. SMU 2 and SMU3 are set to output 0V. This ensures that only the Cell 2 will be turned on during pulse stressing.

Pulse stressing – The output relay for SMU1 is opened, and the gate and drain of Cell 2 are pulse stressed by PG2 #1 (ch 1) and PG2 #2 (ch 1).

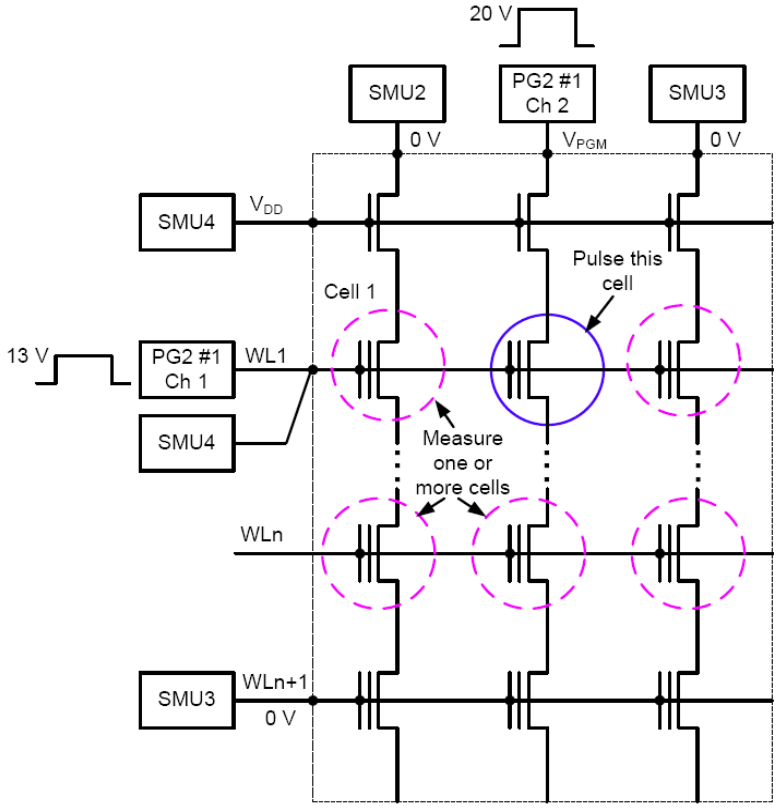
Disturbed cell testing – The outputs for the PG2s are turned off and their output relays are opened. SMU1 and SMU2 are then used to perform a DC Vg-Vd sweep on Cell 1 to determine V_T.

Using a switch matrix

A limitation of the no-switch, direct connect test configuration shown Figure 4-62 is that only three devices can be measured. The test would have to be manually reconfigured or re-cabled to test other devices. Without a switch matrix, the number of adjacent cells that can be measured is limited. Therefore, it is recommended that a switch matrix be used for disturb testing, as shown in Figure 4-69.

Using a switch matrix allows the flexibility of routing pulse and DC signals without having to make connection changes. Also, this type of structure uses a multi-pin probe card, that provides an additional opportunity for mapping test resources to DUT pins. For example, a SMU can be shared across multiple device terminals where the required voltage is the same.

Figure 4-62
Disturb testing – configuration to test a single device



Pulse Waveforms for NVM Testing

The 4205-PG2 has several attributes that support NVM testing. To perform the multi-level pulse waveforms for the typical program/erase waveform (Figure 4-59), each PG2 channel has the Segment Arb™ capability.

See the Segment Arb portion of Section 11 in the 4200-SCS Reference Manual. In addition, the ability to disconnect, or float, a particular device pin, within the Segment Arb waveform requires an inline solid state relay. This solid state relay is called the High Endurance Output Relay (HEOR), and is covered in the Pulse Generator Card portion of Section 11 in the 4200-SCS Reference Manual.

The PG2 output channels each have 50 Ω output impedance. When current flows through the pulse channel, there is a voltage drop across this 50 Ω resistor internal to the pulse card. This means that the voltage at the output may be different from what is expected based on the resistance of the DUT. This effect is called the Load Line Effect and is covered in more detail in DUT Resistance Determines Pulse Voltage across DUT in Section 11 in the 4200-SCS Reference Manual.

Generally, the gate of a flash or NVM device is high impedance, so the voltage at the gate will be double of the programmed voltage. The voltage at the drain will be a function of the resistance of the drain-source, as mentioned above. Adjusting the pulse level to match the desired drain voltage is usually done iteratively with an oscilloscope to measure V_D during the pulse.

The projects in the Flash package use two methods to define the multi-level waveforms typically used in flash memory testing (example: Figure 4-59). For endurance or disturb testing, that uses the subsite stress/measure looping feature of KITE, the Kpulse application is used to define each unique voltage waveform. The details for using Kpulse are provided in Section 13 of the 4200-SCS Reference Manual, in Segment arb waveforms. Here's a brief overview. Use Kpulse to

define and export each unique waveform. The below procedure details how to create and export a hypothetical program/erase waveform.

Using Kpulse to create and export Segment Arb™ waveforms

NOTE *Each segment pulse waveform must have the same total time. . The minimum programmed time for any segment is 20 ns (20E-9), but actual output waveform performance is determined by the channel output capability.*

1. Close KITE and KCON, if open.
2. Open Kpulse.
3. Load the Kpulse setup file "Kpulse_Flash_Example_01.kps." Click on File > Load Setup, then double-click Kpulse_Flash_Example_01.kps. If this file is not available, see Table 4-84 and enter the values into the segment arb definition tables in Kpulse. Kpulse should look similar to Figure 4-63 and Figure 4-64.
   a. The keyboard version of copy (Ctrl-c) and paste (Ctrl-v) can be used to copy the segment arb values between channels. This is useful for ensuring that each waveform has the same period (total waveform time).
   b. To select cells for copying, first move the entry cell to row 1 and Start (V). Then hold down the Shift key while using the cursor arrow keys to highlight all the cells in the segment arb waveform. Then press Ctrl-c to copy.
   c. Place the entry cell into an undefined channel (Row 1, Start (V) column) and press Ctrl-v to copy. Then use the cursor arrow keys to move around and edit the various cells as necessary.
4. The Trigger = 1 values in the 1st and 5th segments. These are the first segments in the program and erase pulses in a typical two pulse program/erase waveform. In the case of multi-card waveform output, the trigger is not used as a typical trigger, but as a synchronizing signal between PG2 cards (see Section 11 "Multi-channel synchronization with the Segment Arb Mode"). It is recommended to use trigger = 1 for the first segment of each pulse in a waveform.
5. For each unique waveform, export each to a file following the steps given in Section 11 "Exporting segment arb waveform files."
6. The exported Segment Arb files cannot be imported back into Kpulse and are saved in the path "c:\S4200\kiuser\KPulse\Sarbfiles" by default.
7. These waveforms will be chosen in the Subsite Setup Device Stress Properties window used in the FlashEndurance projects as well as FlashDisturb. For example, the waveforms are chosen by clicking the browse "…" button on the Device Stress Properties in Figure 4-86. The use of these projects is described below.

Figure 4-63
Kpulse showing the Segment Arb settings for the 4205-PG2 card in the lowest numbered-slot (PG2-1)

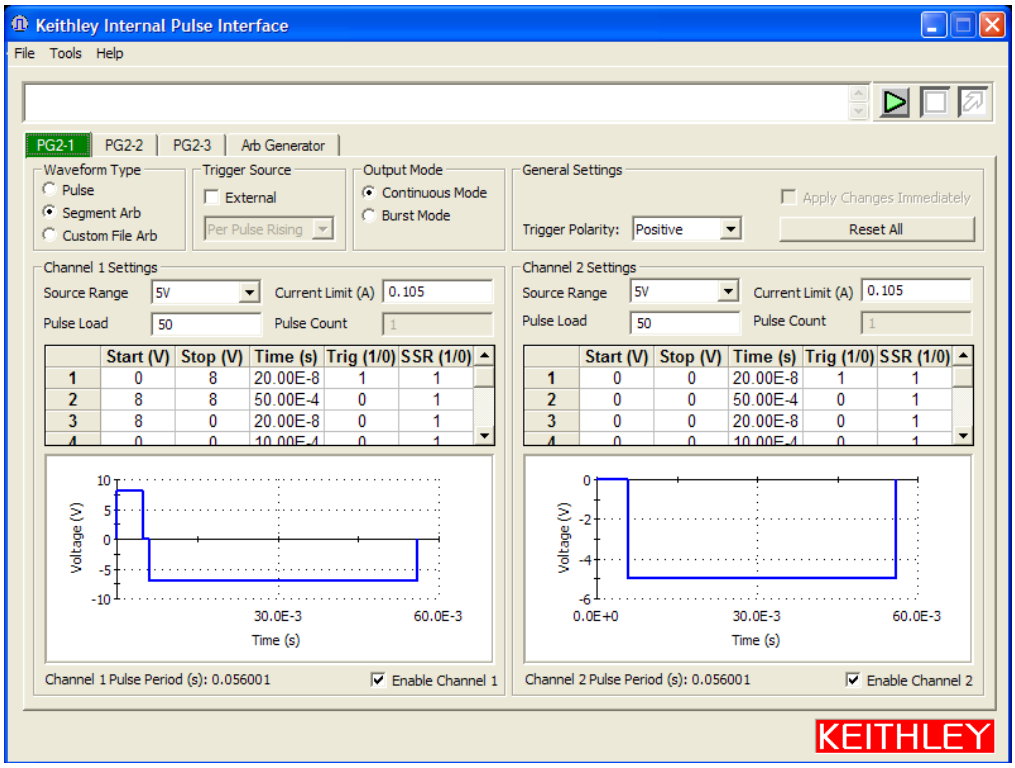


Figure 4-64
Kpulse showing the Segment Arb settings for the 4205-PG2 card in the second lowest-numbered slot (PG2-2)

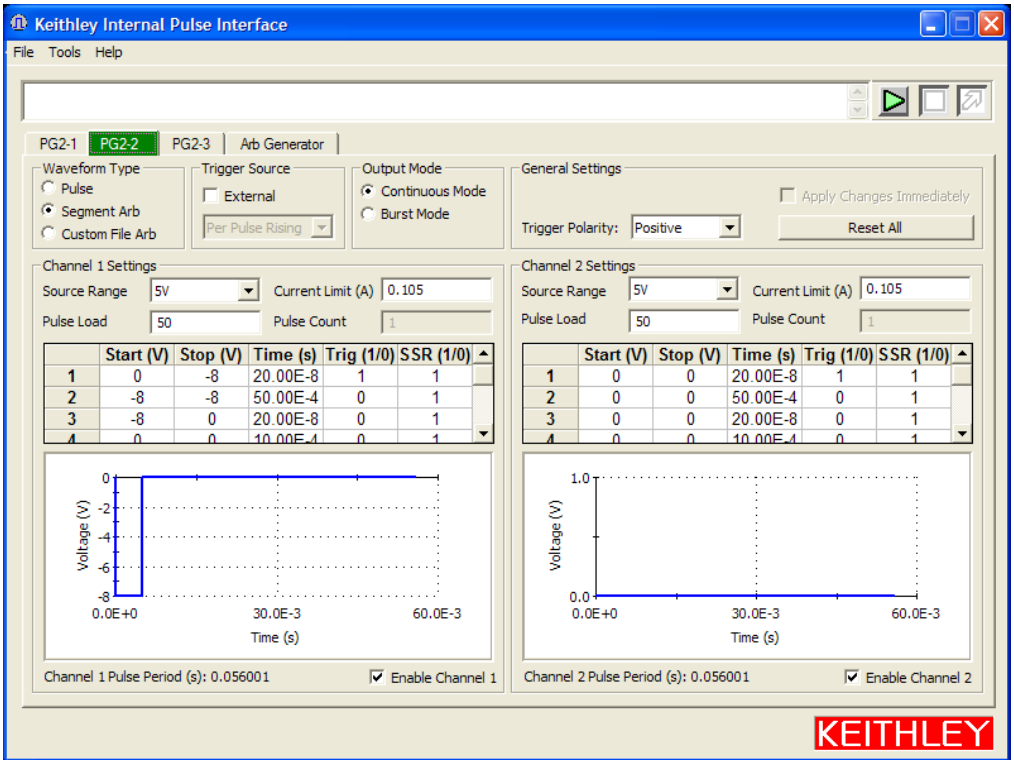


Table 4-84

Segment Arb parameter values for example waveforms

PG2-1 Channel 1					
Segment	Start V	Stop V	Time (s)	Trigger	HEOR*
1	0	8	20.00E-8	1	1
2	8	8	50.00E-4	0	1
3	8	0	20.00E-8	0	1
4	0	0	10.00E-4	0	1
5	0	-7	20.00E-8	1	1
6	-7	-7	5.00E-2	0	1
7	-7	0	20.00E-8	0	1
8	0	0	20.00E-8	0	1
PG2-1 Channel 2					
1	0	0	20.00E-8	1	1
2	0	0	50.00E-4	0	1
3	0	0	20.00E-8	0	1
4	0	0	10.00E-4	0	1
5	0	-5	20.00E-8	1	1
6	-5	-5	5.00E-2	0	1
7	-5	0	20.00E-8	0	1
8	0	0	20.00E-8	0	1
PG2-2 Channel 1					
1	0	-8	20.00E-8	1	1
2	-8	-8	50.00E-4	0	1
3	-8	0	20.00E-8	0	1
4	0	0	10.00E-4	0	1
5	0	0	20.00E-8	1	1
6	0	0	5.00E-2	0	1
7	0	0	20.00E-8	0	1
8	0	0	20.00E-8	0	1
PG2-2 Channel 2					
1	0	0	20.00E-8	1	1
2	0	0	50.00E-4	0	1
3	0	0	20.00E-8	0	1
4	0	0	10.00E-4	0	1
5	0	0	20.00E-8	1	1
6	0	0	5.00E-2	0	1
7	0	0	20.00E-8	0	1
8	0	0	20.00E-8	0	1

NOTE *High Endurance Output Relay – The solid state relay (SSR) on the output of each pulse channel, that provides a high impedance disconnect. (see the 4200-SCS Reference Manual Section 11 Pulse generator card.)*

Entering Segment Arb values into UTM array parameters

The second method for defining a Segment arb waveform is by entering values into arrays for the UTM tests: Program, Erase, Fast-Program-Erase. These UTM-based Segment arb waveforms have been partially pre-defined to reduce the number of parameters required. Figure 4-74 defines the parameters for the single pulse Program and Erase waveforms.

NOTE *The sign of the PulseVoltages array determines whether the pulse is positive (usually for a Program pulse) or negative (usually for an Erase pulse).*

Figure 4-75 defines the parameters for the dual pulse Program and Erase pulse waveform. Each parameter in these figures has a corresponding array, where each entry in the array represents a pulse channel used in the test.

NOTE *The number of parameters and number of pulse channels in the test must be the same. The period of each pulse waveform must be the same.*

This UTM method also includes the triggering settings to synchronize multiple PG2 cards, as described above in the Kpulse method, but they are built-in and do not require user modification.

1. Enter the number of pulse channels required for the test (NumPulseTerminals), up to the maximum number VPU channels in the 4200 chassis (2 channels per pulse card). Figure 4-65 shows 4.
2. Enter the channel names for the number of channels specified above. The names are "VPUnCHm," where n is the number of the VPU card (numbered right to left when viewing the back of the 4200 chassis) and m is the channel number (1 or 2), resulting in "VPU1CH1,VPU1CH2". There is a comma separator, but no spaces used.
3. Now click each array entry and enter the pulse parameter values for each of the 4 channels. There are 5 arrays (red arrows in Figure 4-65) for the 5 pulse parameters shown in Figure 4-74. See Figure 4-66 for two examples of array dialog box displayed after clicking the grey bar on the corresponding UTM parameter (Figure 4-65, red arrows, 1-5).
   NOTE The number of parameters in each array must match the number specified for NumPulseTerminals. If the number of parameters is lower than a previous test, delete the values (blank, not a zero) in the unused cells.
   a. PulseVoltages – Pulse height in volts, assuming a 50 Ω device impedance. The maximum program voltage is 20V, resulting in a nearly 40V pulse on a gate or similar high impedance terminal. To open the solid state relay during the pulse (as shown for V_D in Figure 4-59), use -999.
   b. PrePulseDelays – Pre-pulse delay time in seconds. The minimum time is 20 ns (20E-9), but actual output waveform performance is determined by the channel output capability. Generally, all timing delays are made the same across all channels.
   c. TransitionTimes – Rise and fall times in seconds. The minimum time is 20 ns (20E-9). Generally, all transition times are made the same across all channels.
   d. PulseWidths – Pulse width time in seconds (FWHM – Full width half maximum, as shown in Figure 4-74). The minimum time is 20 ns (20E-9). Generally, all pulse widths are made the same across all channels, but the total waveform time for each channel must be the same.
   e. PostPulseDelays – Post-pulse delay time in seconds. The minimum time is 20 ns (20E-9). Generally, all timing delays are made the same across all channels.
4. Enter the value for NumPulses. For typical characterization, use NumPulses = 1. Setting a higher number is useful for testing multiple pulses before performing a SMU measurement
5. Enter the number of SMUs being used as DC bias terminals, that is useful when DUT terminals require a DC bias to address a particular device. For a direct connection to a single DUT, as shown in Figure 4-67, enter 0. For Figure 4-68, SMU 4 is used to DC bias Bit Line 1, so NumSMUBiasTerminals = 1.
6. Enter the SMU numbers, as a string. For Figure 4-67, leave SMUBiasTerminals blank. For Figure 4-68, use SMUBiasTerminals = SMU4. If more than one bias terminal is required, simply list all SMUs, in ascending order, separated by a comma, but no spaces.
7. Enter the array of bias voltages for the SMUs listed in the previous 2 steps. The number of values in the array must match the value of NumSMUBiasTerminals. If a entry is not needed, delete the value and leave it blank, not 0.
8. Enter the number of SMUs that are sharing a cable with a pulse channel into NumSharedSMUs. Sharing simply means that one pulse and one SMU signal are combined to a single DUT terminal. Figure 4-67 shows 4 SMUs are paired with a pulse channel, with each SMU/pulse pair sharing a cable to a terminal. Figure 4-68 shows that 3 pairs of SMU/pulse channels are shared. The SMA tees on each of the top 3 SMUs that incorporate both a pulse channel and a SMU signal into a single cable to a DUT terminal. Supplying the shared SMU information allows the software to open the SMU relay during the pulse output, that is necessary to permit good pulse fidelity. If a switch matrix is used in the configuration (Figure 4-69), then use NumSharedSMUs = 0.

9. Enter the SMU IDs for the SMU(s) sharing a cable with a pulse channel into SharedSMUs. For the configuration in Figure 4-68, SharedSMUs = SMU1,SMU2,SMU3. There are no spaces allowed in the SharedSMUs string.

Both of the Segment Arb definition methods are required and are test dependent. For all UTMs, the UTM array approach is used. For any stress/measure loop tests, such as endurance or disturb, use Kpulse to define and export the waveform files, then import waveforms into the Device Stress Properties. When the same waveform is required in the stress/measure Device Stress Properties and in a UTM, the same waveform information must be manually entered using both methods. In addition to the waveform definition, the interconnect between cards is necessary to provide synchronized multi-channel Segment Arb™ output.

The interconnection information below is for a typical two card (4 pulse channel) 4200-FLSAH configuration, using 4 Source Measure Units (SMUs). In addition to the cabling there are corresponding parameters in the Segment Arb table that must be set. This is also covered in Section 11, under the heading: Multi-channel synchronization with the Segment Arb Mode.

Figure 4-65
Flash-NAND Project Definition Tab, including arrows for the 6 input arrays

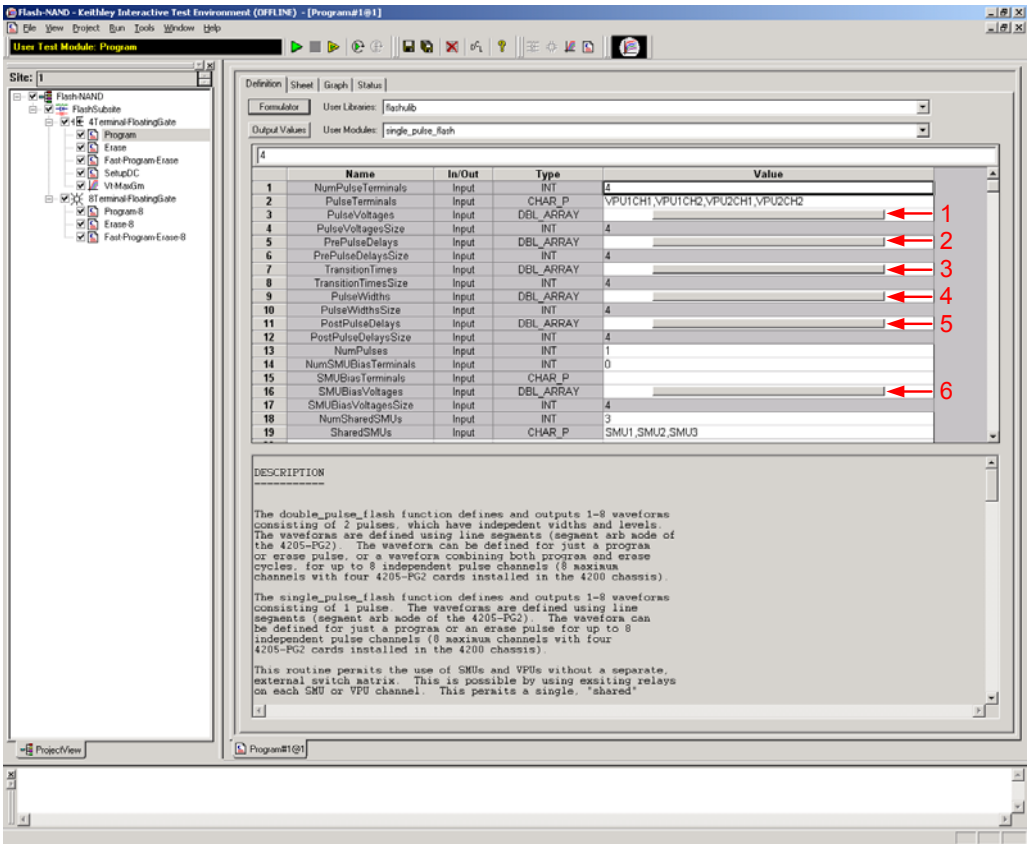
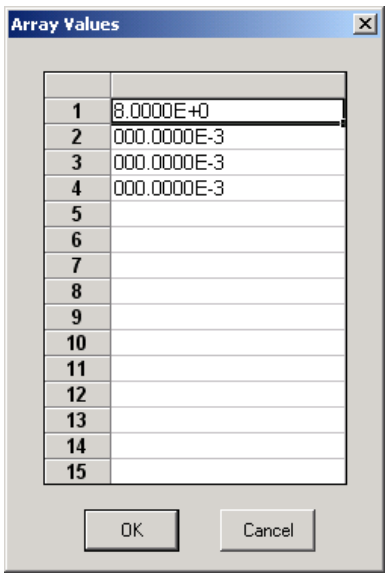
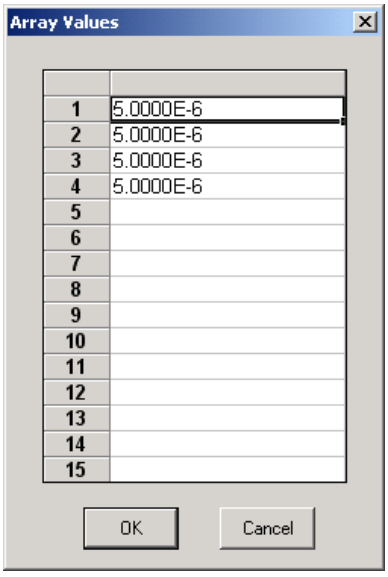


Figure 4-66
Flash-NAND PulseVoltages Array Entry and PrePulseDelays Entry



PulseVoltages Array



PrePulseDelays Array

FLASH Connections

The Flash package includes all the necessary cables and adapters required for the test connections. Also included is an 8 in-lb torque wrench for tightening the SMA connections.

The 4200-FLASH package has 4 channels of multi-level pulse capability. The number of SMUs is configurable. For a system without switching, it is best to have 4 SMUs, to match the number of pulse channels to connect to a 3 or 4 terminal DUT. For a direct connect configuration, the minimum number of pulse channels is equal to the number of DUT terminals that need to be simultaneously pulsed, including terminals that must change from connected to disconnected, or open, states (Figure 4-53 and Figure 4-59), for either the program or erase condition. The minimum number of SMUs is determined by the measurement tests and the number of DUT terminals.

NOTE *The 4205-PG2 card is referred to as a VPU, voltage pulse unit, in the software.*

Figure 4-70 shows the items that are supplied with the Flash package.

Interconnect diagrams for flash testing are shown in Figure 4-67, Figure 4-68 and Figure 4-69. Figure 4-67 shows the connections for test configuration shown in Figure 4-56 and Figure 4-60, that is used for both initial program/erase investigation and endurance testing of a direct connect DUT. This configuration does not require a switch matrix, and provides 4 channels of pulse and well as 4 SMUs, to permit full characterization of single (non-array) NVM DUT.

Figure 4-67
Flash connections – program/erase and endurance testing using direct connection to a single, stand-alone 4 terminal device

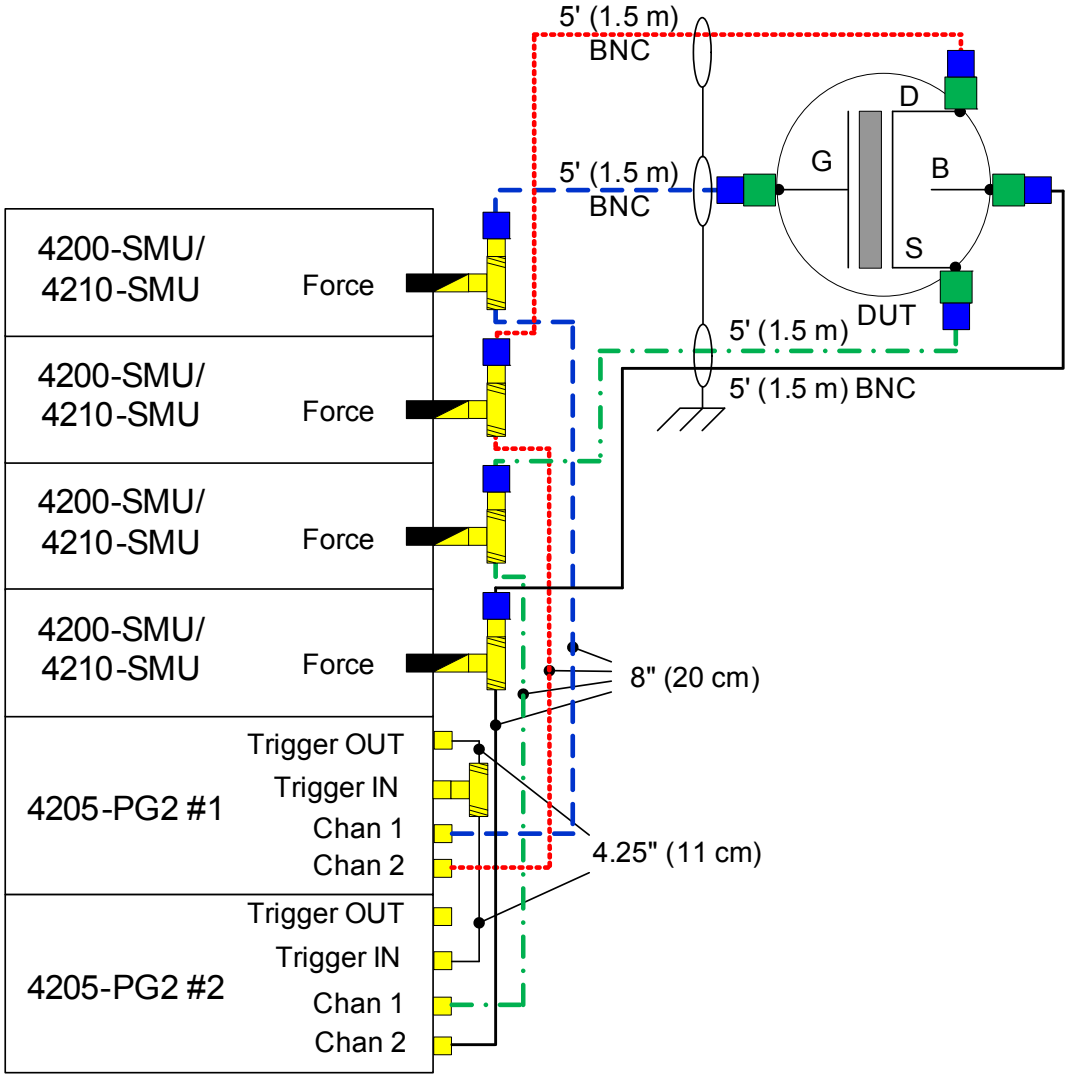


Figure 4-68 shows the connections for test similar to Figure 4-62, that is used for disturb testing. It is strongly recommended to use a switch matrix for testing array test structures, whether for endurance or disturb. However, it is possible to perform a limited test of an array structure without using a switch matrix, as one example is shown in Figure 4-68. Figure 4-68 shows connection to an array test structure, where one of the 4 SMU+PG2 channels was split, to provide a total of 5 test signals to provide the minimum necessary channels for the select pins (Bit Line Select, Bit Lines 1 and 2), to the pulse DUT (circled in blue), and the measure DUTs (circled in dashed purple). Figure 4-68 allows for pulsing one DUT, while performing disturb measurements on the 3

DUTs labeled “Measure.” The preferred connection method for disturb testing, or any testing of a an array DUT, is to add a switch matrix, as shown in Figure 4-69.

Figure 4-68
Flash direct DUT connections – Disturb testing

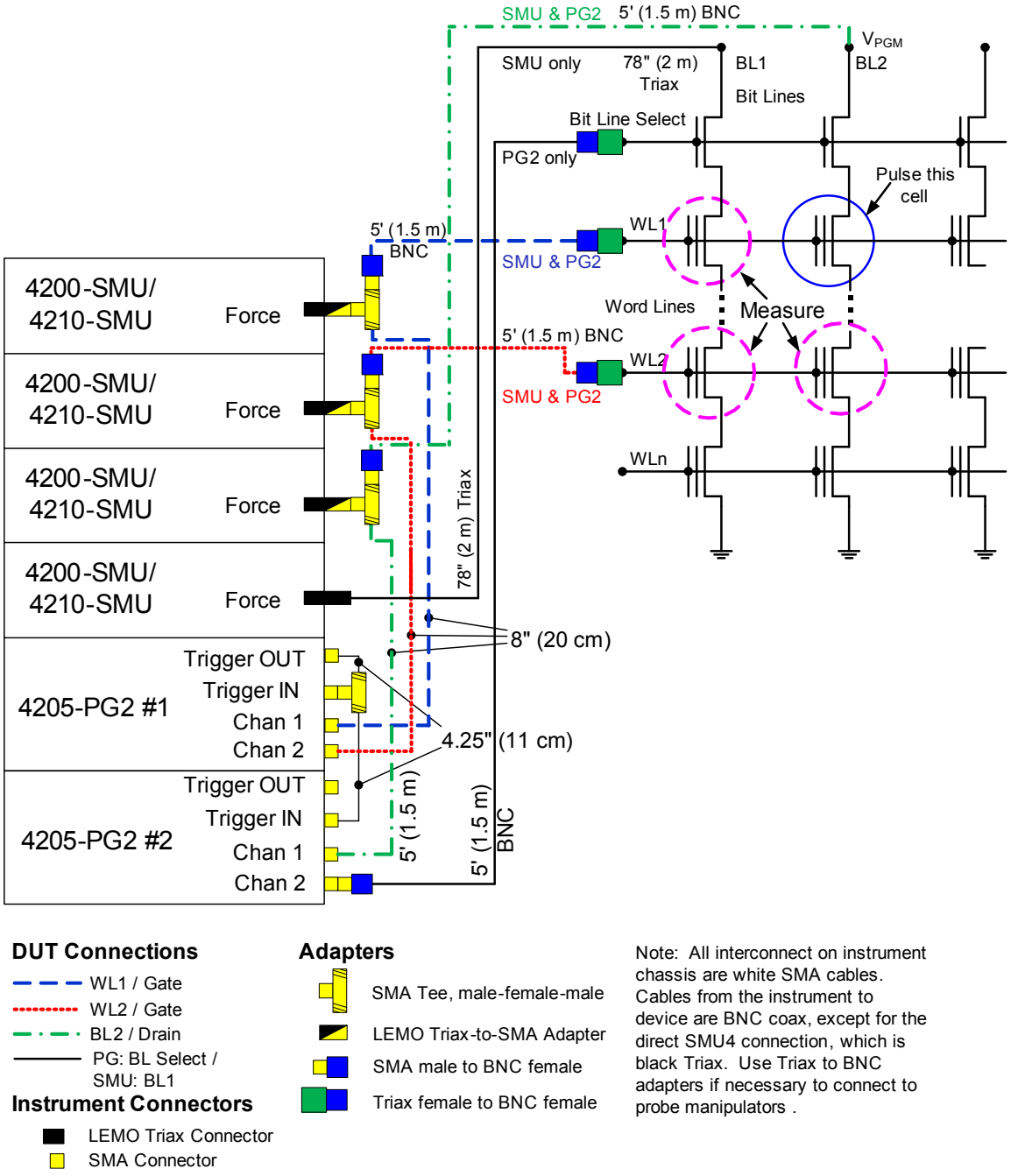


Figure 4-69
Flash Switch connections – Characterization, endurance or disturb testing

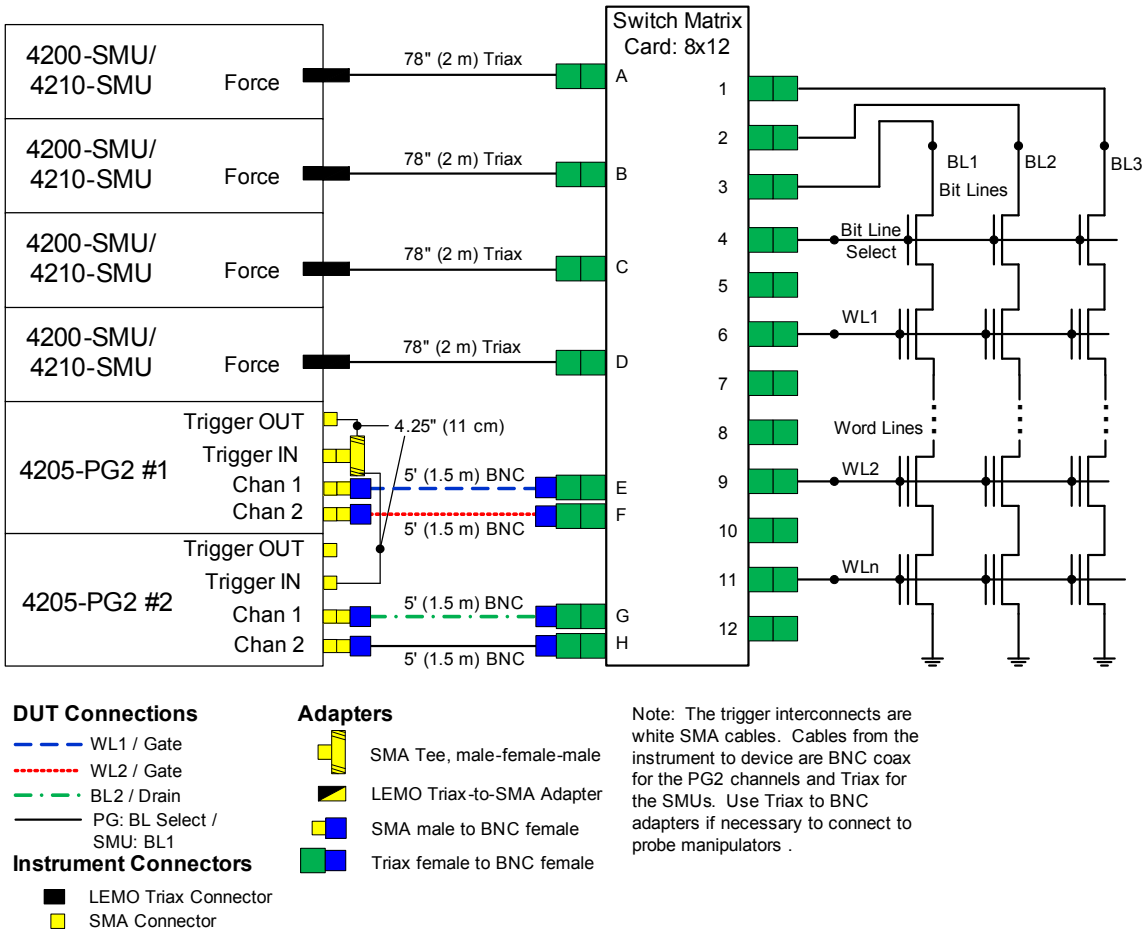


Figure 4-70
Supplied items for 4200-Flash package

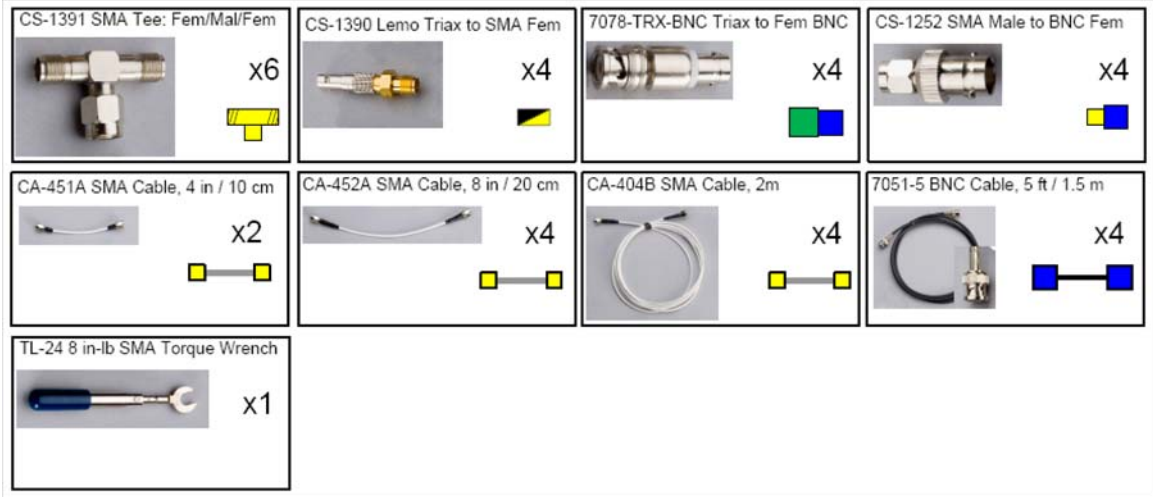


Table 4-85
Interconnect Parts for 4200-FLASH package

Quantity	Description	Comment
6	SMA Tee, female – male – female	Trigger, combine SMU and PG2 channels
4	LEMO triax to SMA adapter	Adapt SMU Force output to SMA for signal interconnect
4	3 slot male triax to female BNC adapter	Convert BNC cabling to Triax for prober or switch matrix connection
4	SMA male to BNC female adapter	Adapt Tee to BNC for cabling from instrument to probe manipulators
2	4.25 in (10.8) cm white SMA cables	Interconnect for triggering
4	8 in (20.3 cm) white SMA cables	Interconnect between PG2 and SMU signals
4	6.6 ft (2 m) white SMA cables	
4	5 foot/ 1.5 m BNC cable	Connect to probe manipulators

Table 4-86
Tools supplied with the 4200-FLASH package

Quantity	Description
1	SMA Torque wrench, 8 in-pound, with 5/16 head installed

NOTE *Use the supplied torque wrench to tighten each connection as it is assembled. Always connect and torque adapter/cable assemblies before attaching the assembly to the instrument cards. Pre-torquing eliminates any non-axial stress on the bulkhead connectors on the SMU or pulse cards, that could possibly cause damage to the cards installed in the 4200 chassis, requiring repair.*

To remove the LEMO triax-to-SMA adapter from a SMU, pull on the knurled silver portion of the connector to release the latches and permit the adapter to separate from the SMU connector. Failure to fully disengage the LEMO adapter latches may result in damage to the adapter and/or the SMU, requiring repair.

The connection instructions below assume a 4 channel 4200-FLASH system, consisting of 2 4205-PG2 cards (4 pulse channels) as well as 4 SMUs, either 4200-SMU or 4210-SMU, with SMU preamps removed (Figure 4-54 or Figure 4-55).

To test on-wafer devices, there are various ways to connect the supplied SMA cables to the probe manipulators. For the direct connect method (Figure 4-67 and Figure 4-68) or switch method (Figure 4-69), adapters convert the BNC cabling to the Triax connector compatible with many types of probe manipulators.

Direct connection to single DUT

Cabling instructions for direct connect to single DUT are below. Refer to Figure 4-67 for the following procedure. These instructions are compatible with the following projects in the "Projects_Memory" folder: Flash-NAND, Flash-NOR, FlashDisturb-NAND, FlashDisturb-NOR, FlashEndurance-NAND, FlashEndurance-NOR.

1. Set up the 4200-SCS, referring to the 4200-SCS Quick Start Guide and 4200-SCS Reference Manual (Sections 2, 4).
2. Take one of the SMA Tees and connect the two shortest (4.25 inch or 10.8 cm) SMA cables to either end. Torque both connections using the wrench.
3. Connect this assembly to the right-most PG2 card, that is, the PG2 card in the lowest numbered slot. First connect one of the SMA cables to TRIGGER OUT and connect the SMA tee to TRIGGER IN. Torque using wrench.

4. Then connect the other SMA cable to TRIGGER IN on the second PG2 card. This second card is the card to the immediate left of the card in step 3. Torque using wrench. *NOTE: If the FLASH package consists of more than 2 PG2 cards, continue to connect the cable and Tees to the adjacent cards. For a system consisting of 4 PG2 cards, there should be 3 SMA tees used to connect the triggering across the 4 cards*.
5. Take an SMA Tee and connect one SMA-to-BNC adapter to one of the female connectors. Torque using wrench.
6. Connect the assembly from step 5 to one Triax-to-SMA Adapter. Torque using wrench.
7. Connect one 8 inch (20.3 cm) SMA cable to the remaining SMA female connector.
8. Connect one 5 foot (1.5 m) black BNC cable to the BNC connection.
9. Perform steps 5-8 three more times.
10. Take one of the cable assemblies from step 9, connect the SMA to CHANNEL 2 of the PG2 in the left most slot (PG2 in the slot with the highest number). Torque using wrench.
11. Carefully insert the LEMO Triax connector from step 10 into the Force connector on the SMU in Slot 4.
12. Route BNC cable from SMU4 to the DUT terminal Bulk connection. Connect Triax-to-BNC adapter, if necessary. Connect cable to probe manipulator.
13. Take one of the cable assemblies from step 9, connect the SMA to CHANNEL 1 of the PG2 in the left most slot (PG2 in the slot with the highest number). Torque using wrench.
14. Carefully insert the LEMO Triax connector from step 13 into the Force connector on the SMU in Slot 3.
15. Route BNC cable from SMU3 to the DUT terminal Source connection. Connect Triax-to-BNC adapter, if necessary. Connect cable to probe manipulator.
16. Take one of the cable assemblies from step 9, connect the SMA to CHANNEL 2 of the PG2 in the right-most slot (PG2 in the slot with the lowest number). Torque using wrench.
17. Carefully insert the LEMO Triax connector from step 16 into the Force connector on the SMU in Slot 2.
18. Route BNC cable from SMU2 to the DUT terminal Drain connection. Connect Triax-to-BNC adapter, if necessary. Connect cable to probe manipulator.
19. Take one of the cable assemblies from step 9, connect the SMA to CHANNEL 1 of the PG2 in the right-most slot (PG2 in the slot with the lowest number). Torque using wrench.
20. Carefully insert the LEMO Triax connector from step 19 into the Force connector on the SMU in Slot 1.
21. Route BNC cable from SMU1 to the DUT terminal Gate connection. Connect Triax-to-BNC adapter, if necessary. Connect cable to probe manipulator.

Direct connection to array DUT for disturb testing

Cabling instructions for direct connect to an array DUT are below. Refer to Figure 4-68 for the following procedure. These instructions are compatible with the following projects in the "Projects_Memory" folder: FlashDisturb-NAND, FlashDisturb-NOR, FlashEndurance-NAND, FlashEndurance-NOR.

1. Set up the 4200-SCS, referring to the 4200-SCS Quick Start Guide and 4200-SCS Reference Manual (Sections 2, 4).
2. Take one of the SMA Tees and connect the two shortest (4.25 inch or 10.8 cm) SMA cables to either end. Torque both connections using the wrench.
3. Connect this assembly to the right-most PG2 card, that is, the PG2 card in the lowest numbered slot. First connect one of the SMA cables to TRIGGER OUT and connect the SMA tee to TRIGGER IN. Torque using wrench.
4. Then connect the other SMA cable to TRIGGER IN on the second PG2 card. This second card is the card to the immediate left of the card in step 3. Torque using wrench. *NOTE: If the FLASH package consists of more than 2 PG2 cards, continue to connect the cable and*

Tees to the adjacent cards. For a system consisting of 4 PG2 cards, there should be 3 SMA tees used to connect the triggering across the 4 cards.

5. Take one SMA-to-BNC adapter and connect one 5 foot (1.5 m) black BNC cable.
6. Take the cable from step 5 and connect the SMA adapter to CHANNEL 2 of the PG2 in the left most slot (PG2 in the slot with the highest number). Torque using wrench.
7. Route BNC cable from step 6 to the DUT array Bit Line Select connection. Use a Triax-to-BNC adapter, if necessary. Connect cable to probe manipulator.
8. Take one black Lemo Triax to 3-slot Triax cable and insert the LEMO end into the Force connection on the left-most SMU in slot 4.
9. Route triax from SMU4 to the DUT array BL1 connection. Connect cable to probe manipulator.
10. Take an SMA Tee and connect one SMA-to-BNC adapter to one of the female connectors. Torque using wrench.
11. Connect the assembly from step 5 to one Triax-to-SMA Adapter. Torque using wrench.
12. Connect one 8 inch (20.3 cm) SMA cable to the remaining SMA female connector.
13. Connect one 5 foot (1.5 m) black BNC cable to the BNC connection.
14. Perform steps 10-13 two more times.
15. Take one of the cable assemblies from step 14, connect the SMA to CHANNEL 2 of the PG2 in the left most slot (PG2 in the slot with the highest number). Torque using wrench.
16. Carefully insert the LEMO Triax connector from step 15 into the Force connector on the SMU in Slot 4.
17. Route BNC cable from SMU4 to the DUT array WL2 terminal. Use a Triax-to-BNC adapter, if necessary. Connect cable to probe manipulator.
18. Take one of the cable assemblies from step 14, connect the SMA to CHANNEL 1 of the PG2 in the left most slot (PG2 in the slot with the highest number). Torque using wrench.
19. Carefully insert the LEMO Triax connector from step 18 into the Force connector on the SMU in Slot 3.
20. Route BNC cable from SMU3 to the DUT array BL2 connection. Connect Triax-to-BNC adapter, if necessary. Connect cable to probe manipulator.
21. Take one of the cable assemblies from step 14, connect the SMA to CHANNEL 2 of the PG2 in the right-most slot (PG2 in the slot with the lowest number). Torque using wrench.
22. Carefully insert the LEMO Triax connector from step 21 into the Force connector on the SMU in Slot 2.
23. Route BNC cable from SMU2 to the DUT array WL2 connection. Connect Triax-to-BNC adapter, if necessary. Connect cable to probe manipulator.
24. Take one of the cable assemblies from step 14, connect the SMA to CHANNEL 1 of the PG2 in the right-most slot (PG2 in the slot with the lowest number). Torque using wrench.
25. Carefully insert the LEMO Triax connector from step 24 into the Force connector on the SMU in Slot 1.
26. Route BNC cable from SMU1 to the DUT array WL1 connection. Connect Triax-to-BNC adapter, if necessary. Connect cable to probe manipulator.

Switch matrix connection to array DUT

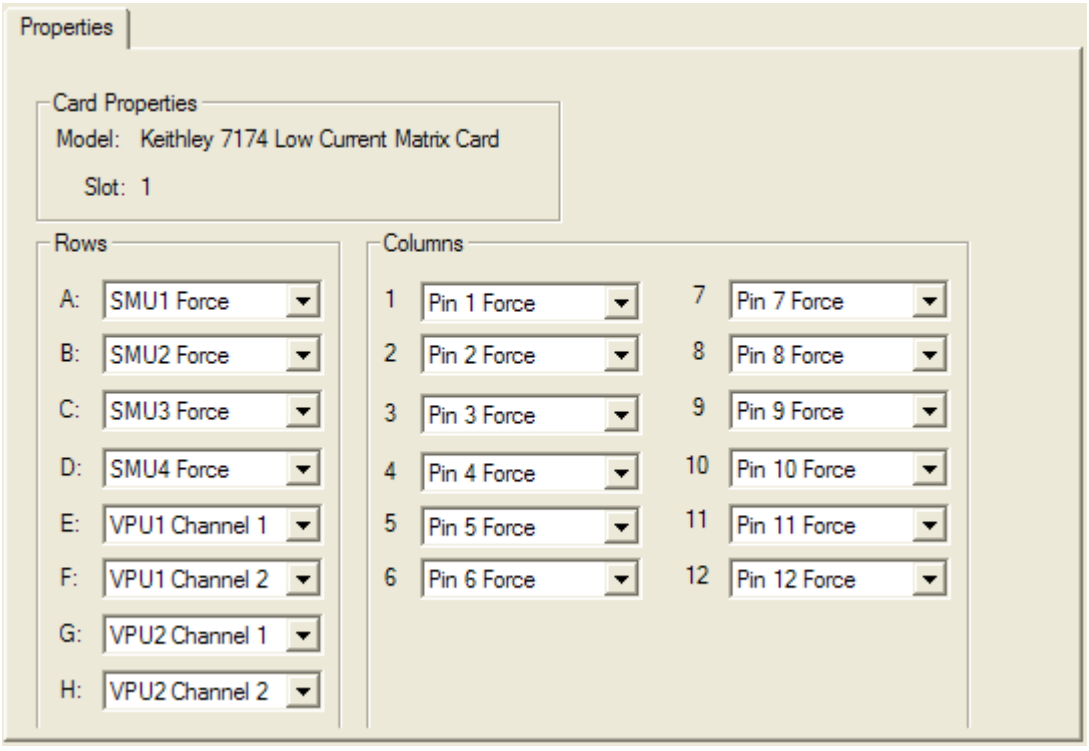
Cabling instructions for switch connect to an array DUT are below. Refer to Figure 4-69 for the following procedure. These instructions are compatible with the following projects in the "Projects_Memory" folder: Flash-Switch, FlashDisturb-Switch, FlashEndurance-Switch. Unlike the direct connect methods described above, the use of a switch matrix permits the use of SMU PreAmps.

1. Set up the 4200-SCS, referring to the 4200-SCS Quick Start Guide and 4200-SCS Reference Manual (Sections 2, 4).
2. Set up the 707 or 708 Switch matrix using Appendix B of the 4200 Complete Reference Manual. Specifically following the steps under "Using KCON to add a switch matrix to the

system." See Figure 4-71 for a suggested configuration for the row-column connection, consistent with Figure 4-69.

3. Take one of the SMA Tees and connect the two shortest (4.25 inch or 10.8 cm) SMA cables to either end. Torque both connections using the wrench.
4. Connect this assembly to the right-most PG2 card, that is, the PG2 card in the lowest numbered slot. First connect one of the SMA cables to TRIGGER OUT and connect the SMA tee to TRIGGER IN. Torque using wrench.
5. Then connect the other SMA cable to TRIGGER IN on the second PG2 card. This second card is the card to the immediate left of the card in step 4. Torque using wrench. *NOTE: If the FLASH package consists of more than 2 PG2 cards, continue to connect the cable and Tees to the adjacent cards. For a system consisting of 4 PG2 cards, there should be 3 SMA tees used to connect the triggering across the 4 cards.*
6. Take one SMA-to-BNC adapter and connect one 5 foot (1.5 m) black BNC cable.
7. Take the cable from step 5 and connect the SMA adapter to CHANNEL 2 of the PG2 in the left most slot (PG2 in the slot with the highest number). Torque using wrench.
8. Route BNC cable from step 7 to the switch matrix card Triax input, using a Triax-to-BNC adapter.
9. Repeat steps 6-8 for the other 3 PG2 pulse channels.
10. Take one black Lemo Triax to 3-slot Triax cable and insert the LEMO end into the Force connection on the left-most SMU in slot 4.
11. Route triax from SMU4 to the switch matrix card Triax input.
12. Repeat steps 10-11 for the remaining 3 SMUs.
13. Use triax cables to route the switch matrix outputs to the array DUT probe manipulators.

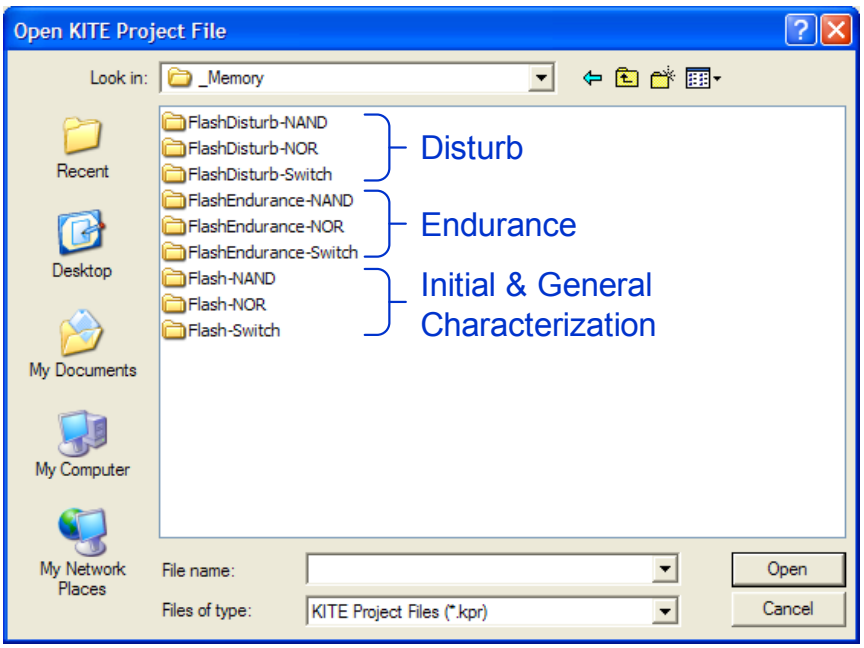
Figure 4-71
KCON Row-Column Card Properties for Flash testing with 4 SMUs and 4 VPU pulse channels



Flash Projects

There are three types of projects included with the 4200-FLASH package: Initial Characterization, Endurance, and Disturb. Each type of project has three different sets of defaults for common setups: NAND device (direct connect), NOR device (direct connect), and Switch (using a Keithley 707A/708A switch matrix and compatible card(s)). This results in the nine different projects in the "Projects_Memory" folder as shown in Figure 4-72.

Figure 4-72
Project listing _Memory folder



Flash-NAND project

Flash-NOR project

Flash-Switch project

These three projects are similar, providing the ability to send ***n*** pulses to the DUT, then perform a V_T sweep. The pulses can be either a single pulse program or erase waveform, or the combined program and erase waveform (see Figure 4-73). These tests allow investigation into program and erase state dependencies on pulse parameters. There are three different waveform types available: Program, Erase, Fast Program and Erase. The Program waveform and Erase waveform output pulses with a single set of parameters for the pulse width, transition (0-100% rise/fall), and level (see Figure 4-74). The Fast Program and Erase test waveform uses two pulses, that can have independent widths and levels (see Figure 4-75). Each test permits programming the pulse width, level and transition (0-100% rise/fall) parameters, as well as the number of pulses. For extended Program Erase cycling, use one of the FlashEndurance projects. Instead of a voltage, the disconnected, or open, state may also be chosen for any pulse segment. The open state is useful when using Fowler-Nordheim tunneling for programming or erasing a floating gate transistor.

These projects support from 1 to 8 pulse channels to support typical 4-terminal devices, as well as higher pin count devices or array test structures.

NOTE *The 8-terminal testing requires four Model 4205-PG2 cards and, for most tests, a compatible external switch matrix.*

The purpose of these projects is to initially characterize a floating gate transistor. For example, determine the appropriate pulse parameters for both the program and erase waveforms to reach a target V_{TE} and V_{TP}. This can be done by setting the Program pulse height to the desired value, but setting the pulse width to a fraction of the expected pulse width (Entering Segment Arb values into UTM array parameters). Set the NumPulses to 1 and uncheck the Erase and Fast-Program-Erase tests. Run the Program, SetupDC and Vt-MaxGm tests, monitoring the shift in the V_T and noting the number of pulses required to reach the target Vt_{TP}. Then the same approach can be used for the erase. If the DUT was initially in an unknown state, the determination of appropriate pulse parameters for the program and erase waveforms may be iterative. The Fast-Program-Erase test may be used to confirm that the chosen pulse parameters are providing an acceptable erase and the V_T after the Fast-Program-Erase is not shifting. Once acceptable pulse parameters are determined, use Kpulse to define and export the waveforms for use in the Endurance and Disturb projects. (Using Kpulse to create and export Segment ArbTM waveforms).

The difference between the Flash-NAND and Flash-NOR are the typical pulse widths and levels specific to the DUT type. The Flash-switch is a generic example of the Flash testing described above, but adds support for an external Keithley switch matrix.

Flash-NAND tests

Program

Erase

Fast-Program-Erase

SetupDC

Vt-MaxGm

Program-8

Erase-8

Fast-Program-Erase-8

The Flash-NAND project navigator is shown in Figure 4-73.

Figure 4-73
Flash-NAND project

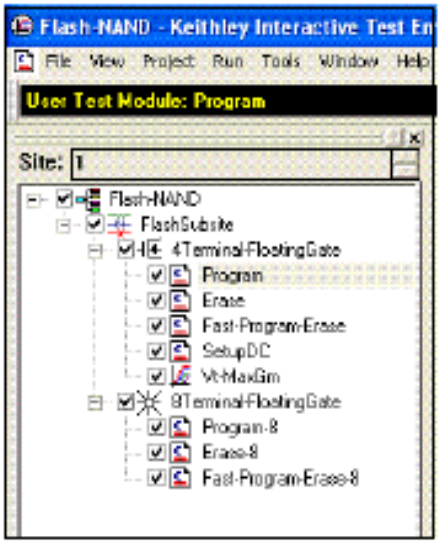


Figure 4-74
Parameters for Program or Erase UTMs (using single_pulse_flash module)

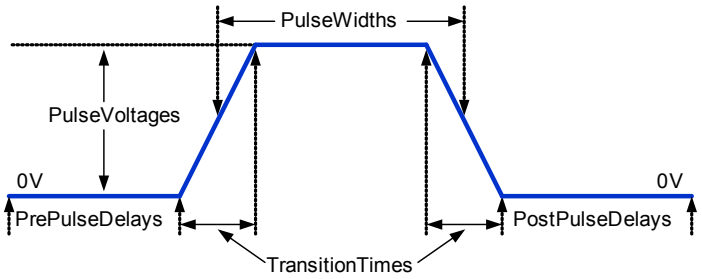
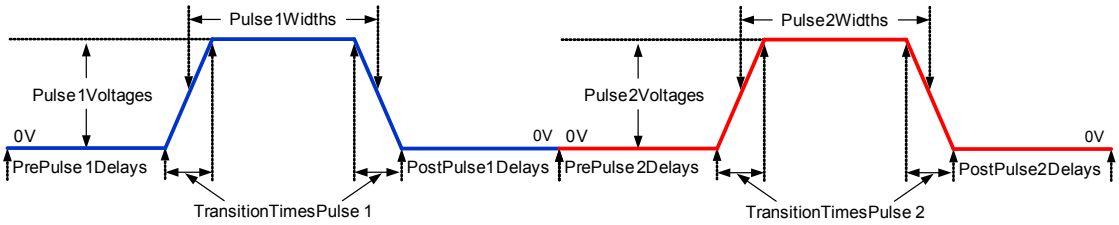
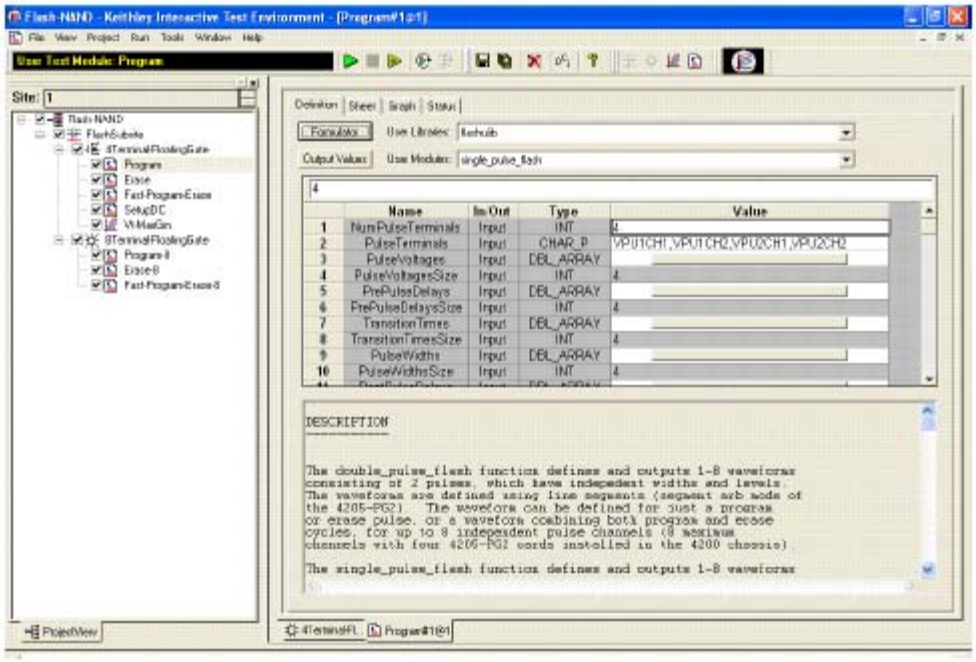


Figure 4-75
Parameters for Fast Program-Erase pulse waveform (using double_pulse_flash module)



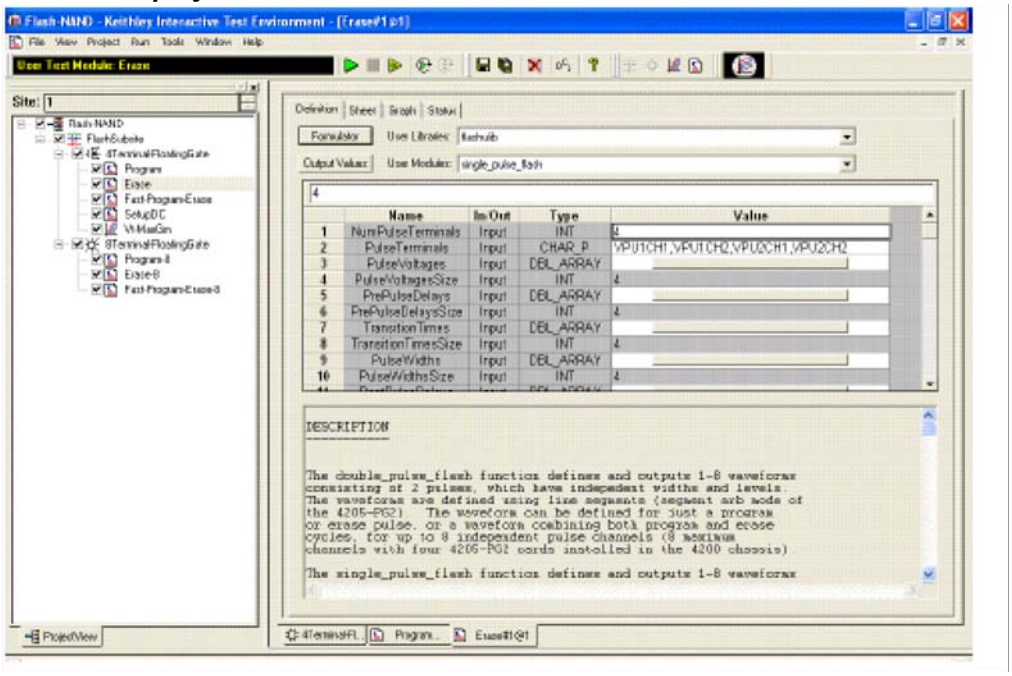
Program test – This test uses a partially pre-defined waveform, see Figure 4-74, to program a flash memory device. The Definition tab for this test is shown in Figure 4-76.

Figure 4-76
Flash-NAND project – Program definition tab



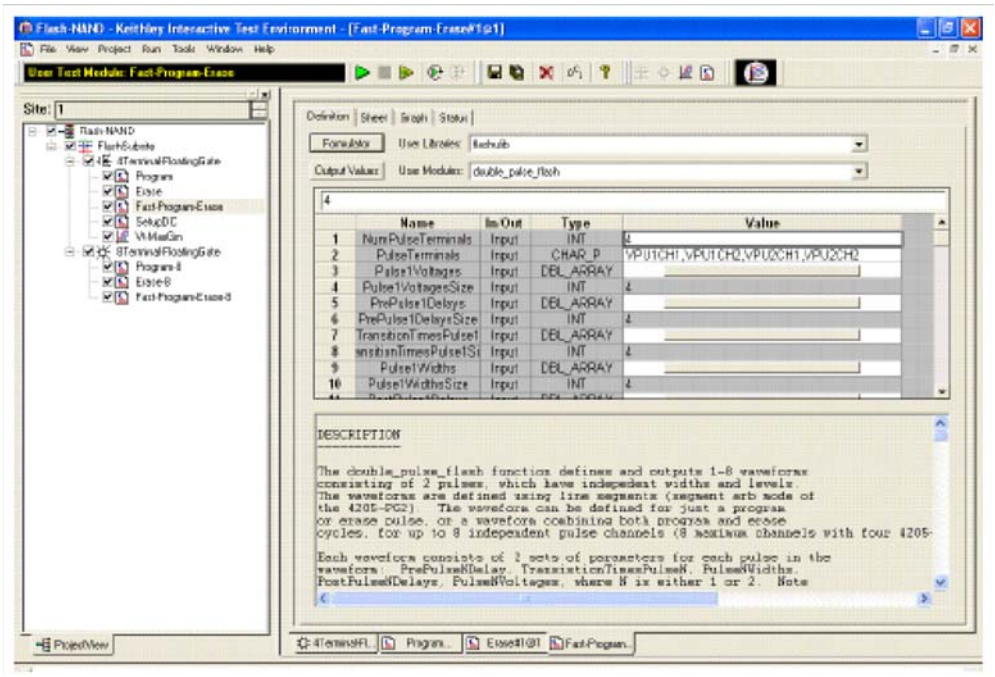
Erase test – This test uses a partially pre-defined waveform, see Figure 4-74, to erase a flash memory device. The Definition tab for this test is shown in Figure 4-77.

Figure 4-77
Flash-NAND project – Erase definition tab



Fast-Program-Erase test – This test uses a partially pre-defined waveform, see Figure 4-75, to program and erase a flash memory device. The Definition tab for this test is shown in Figure 4-78.

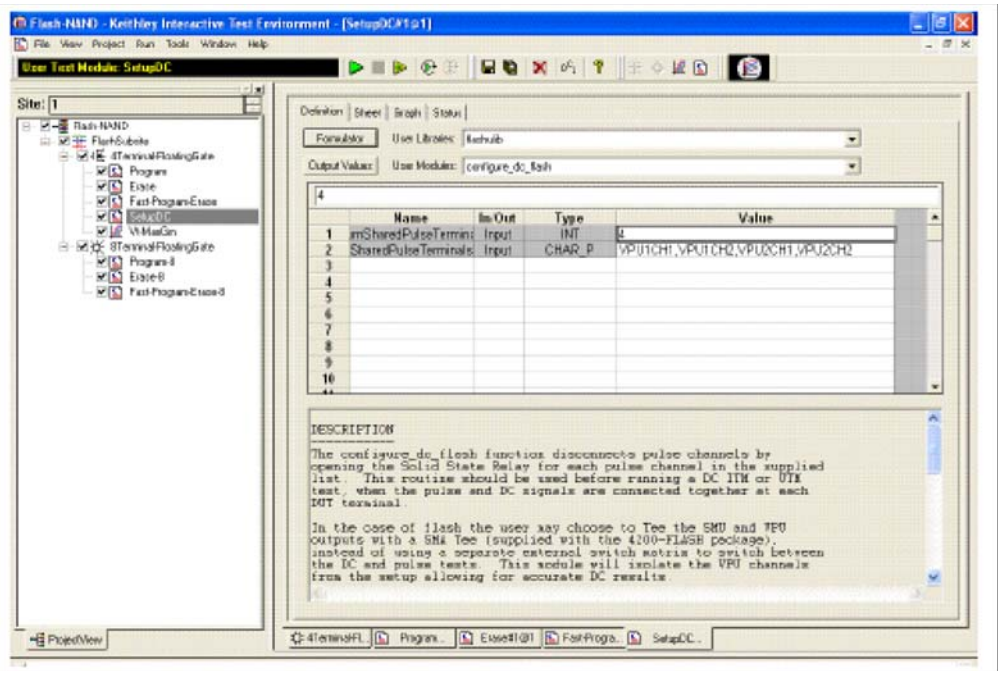
Figure 4-78
Flash-NAND project – Fast-Program-Erase definition tab



SetupDC test – The Definition tab for this test is shown in Figure 4-79. This test isolates the VPU outputs from the DUT, allowing the SMUs to perform a DC without signal interference from the pulse outputs. It does this by opening the High Endurance Output Relay for each VPU channel in the PulseTerminals list. Disconnecting the VPU channels allows for accurate DC results. This isolation step is only necessary when using the direct connect method (Figure 4-67 and Figure 4-68), where the SMU and VPU signals are sharing a single connection to each device terminal (Figure 4-60). The same test step is called Open-VPU-Relay and is optional for switch matrix configurations (Figure 4-69), but is recommended to prevent accidental simultaneous connection of both SMU and PG2 channels to a single terminal.

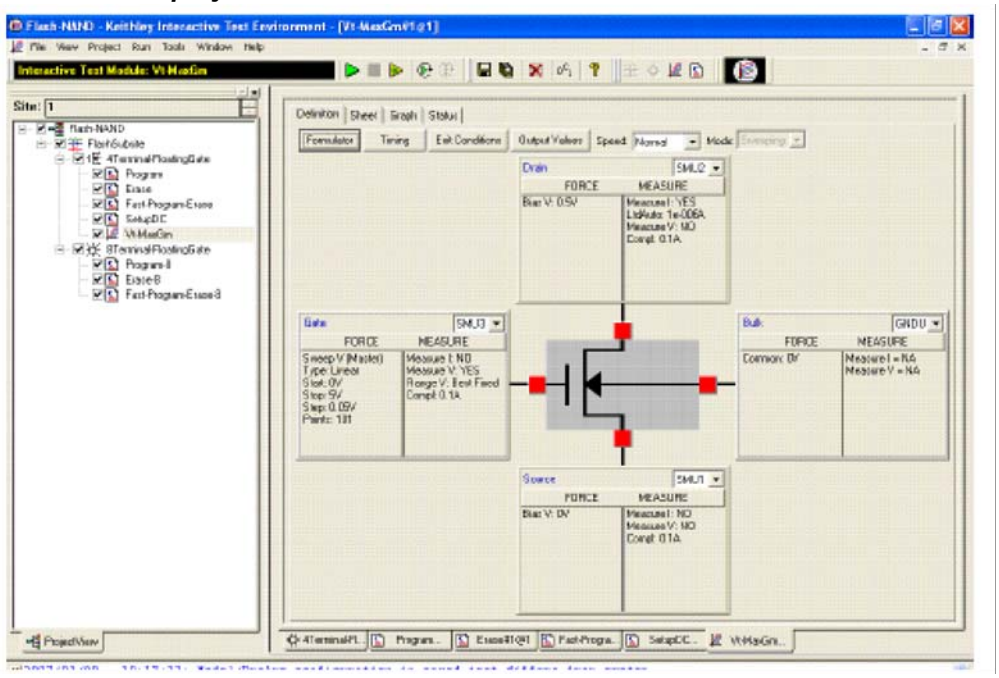
The SetupDC test step is used in the configuration ***without*** a switch matrix and is required before any DC tests. When using a switch matrix, a ConPin test can replace the SetupDC test (see the `config` LPT function in Section 8 of the Reference Manual) to set the appropriate matrix connections prior to any DC tests.

Figure 4-79
Flash-NAND project – SetupDC definition tab



Vt-MaxGm test – This test is used to perform a DC voltage sweep on the gate of the DUT and measure the drain current at each sweep step. The default Definition tab for this test is shown in Figure 4-80. SMU3 is configured to perform a 101 point sweep from 0 to 5V in 50mV steps. SMU1 is configured to DC bias the drain at 0.5V and measure current at each step of the sweep.

Figure 4-80
Flash-NAND project – Vt-MaxGm definition tab



Program-8 test – This test uses Segment Arb waveforms to program an 8-terminal flash memory device.

Erase-8 test – This test uses Segment Arb waveforms to erase an 8-terminal flash memory device.

Fast-Program-Erase-8 test – This test uses Segment Arb waveforms to program and erase an 8-terminal flash memory device.

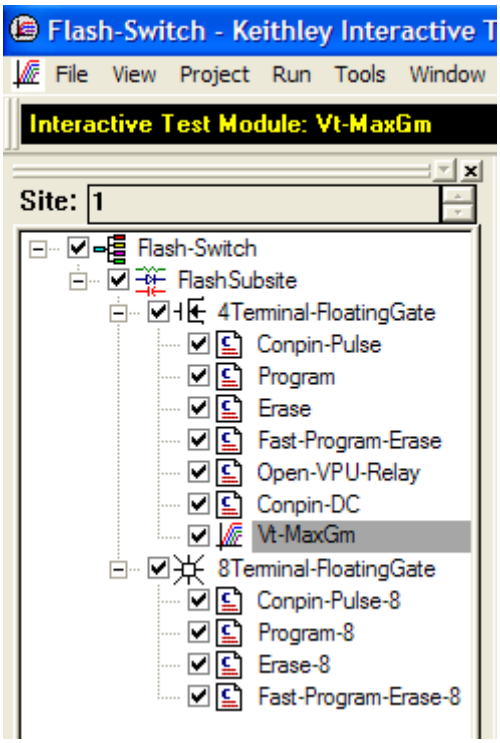
Flash-NOR tests

The Flash-NOR project has tests similar to the Flash-NAND project, with parameter defaults for NOR type floating gate DUTs.

Flash-switch tests

The Flash-Switch project has similar tests to the Flash-NAND, with parameter defaults for using a switch matrix for more complex multi-DUT addressable test structures (see Figure 4-55). Also, SMU and 4205-PG2 pulse channels are connected to the matrix differently, eliminating the SMU+Pulse sharing of cables to the DUT. Using the switch means that ConPin tests (Figure 4-81) are added after the Open-VPU-Relay tests in the direct-connect versions Flash-NAND (Figure 4-73) and Flash-NOR.

Figure 4-81
Flash-Switch project



ConPin-Pulse or **ConPin-DC test** – This test is used to connect pulse or SMUs to the DUT. Figure 4-82 shows the definition tab for ConPin-Pulse. The parameters are typed into the UTM parameter table, with the Pin1, Pin2, etc determining where the instrument (SMU, VPU) signals connect. It is also possible to configure a single switch matrix card using the GUI. Click the GUI button shown in Figure 4-82 to see the dialog in Figure 4-83.

Figure 4-82
ConPin-Pulse test Definition tab

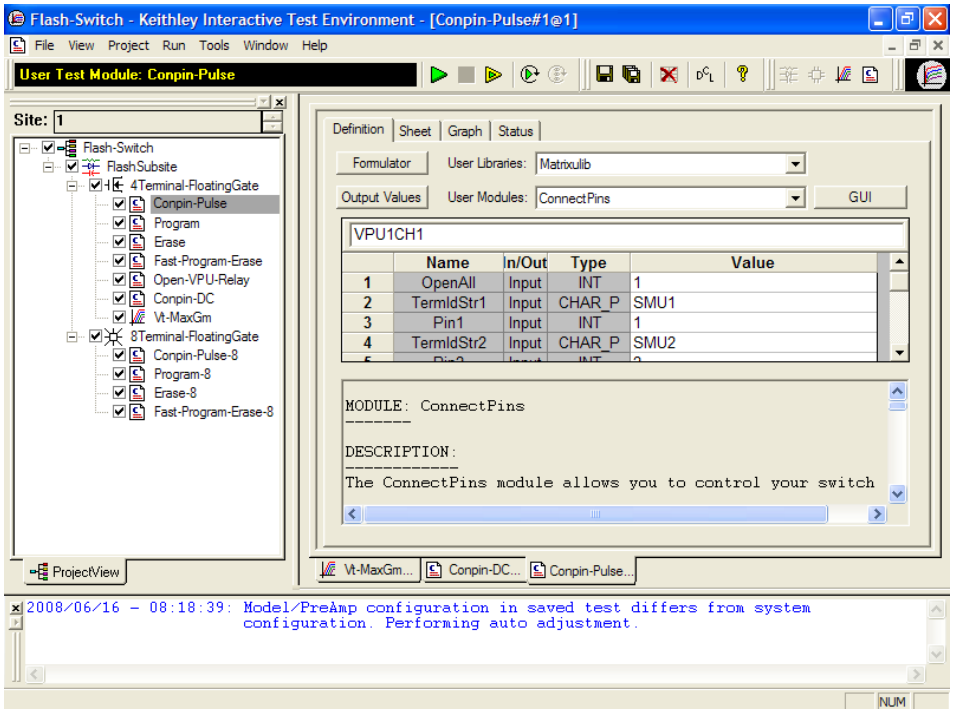
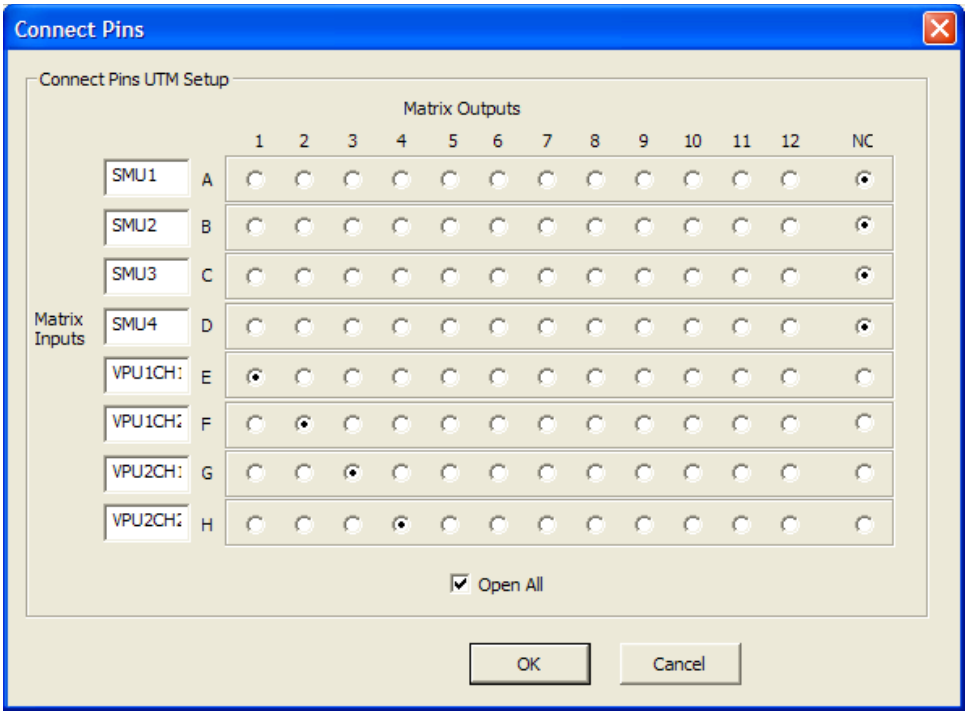


Figure 4-83
ConPin-Pulse test GUI definition dialog



Running any Flash Project for the first time

1. Connect up the 4200-FLASH package, using the Flash Connections instructions for one of the following configurations:
   a. Direct connection (no switch matrix) to single DUT: Use Figure 4-67 and Direct connection to single DUT.
   b. Direct connection (no switch matrix) to array DUT: Figure 4-68 and Direct connection to array DUT for disturb testing.
   c. Switch matrix connection to array or single DUT: Use Figure 4-69 and Switch matrix connection to array DUT.
2. See below procedures for using a specific Flash project.
   a. Initial characterization using the Running the Flash-NAND, Flash-NOR or Flash-Switch Project
   b. Endurance or Disturb testing using the Running a FlashEndurance or FlashDisturb Project by using FlashEndurance-NAND, FlashEndurance-NOR, FlashEndurance-Switch, FlashDisturb-NAND, FlashDisturb-NOR, or FlashDisturb-Switch.

Running the Flash-NAND, Flash-NOR or Flash-Switch Project

The Flash projects use a small number of tests and methods. This section will explain the tests and how to set parameter values.

These projects allow initial characterization of a device, including the determination of the pulse settings (pulse width, height, and transition time) that will provide a target programmed or erased V_T. After the appropriate pulse settings are determined, they can be used to perform Endurance or Disturb testing on the DUT.

1. If system connections have not been made, follow the instruction in Running any Flash Project for the first time.
2. If KITE is not running, start KITE by double-clicking the KITE icon on the 4200 desktop.
3. Open the appropriate KITE Flash project.

a. Within KITE, click FILE > Open Project. If the dialog window is not displaying the _Memory folder, move up one or two levels to the display the Projects directory. Double-click the "_Memory" folder, then double-click the desired Flash test folder (Flash-NAND, Flash-NOR or Flash-Switch), then double-click the Flash-NAND.kpr, Flash-NOR.kpr or Flash-Switch.kpr file to open the desired 4200 Flash project.
b. KITE should resemble Figure 4-76 (or Figure 4-81 for Flash-Switch Project).

4. Touch-down or connect the Device Under Test (DUT).
5. Verify setup and connection by running Vt-MaxGm test
   a. Set appropriate voltages
   b. Run the test by clicking the yellow and green triangle Append button.
   c. Ensure that the V_G-I_D and V_T results are reasonable.
6. Determine the appropriate pulse voltage levels.
   a. Review the section Pulse Waveforms for NVM Testing
   b. Recall that pulse voltage levels on the gate will double
      i. For example, using PulseVoltage = 2 will result in V_G = 4V for a typical high-impedance (1 k Ω) terminal.
   c. Use oscilloscope to determine appropriate PulseVoltage values for the desired V_G and V_D, making sure to use the 1M Ω input setting on the oscilloscope.
      i. Drain: Connect the oscilloscope probe across the drain-source of the DUT.
      ii. Use the Program and Erase UTMs to output pulses, while using the oscilloscope to measure the pulse height. Iterate by modifying the PulseVoltages to reach the target voltage.
7. Once the appropriate voltage levels are determined, the appropriate pulse width may be determined by iteratively outputting pulses while occasionally measuring the V_T.
   a. Start by using a pulse width that is shorter than the expected PW. For example, if 20 μs is the expected PW, try using a 2 μs PW.
   b. Enter the parameter values into the Program UTM, following the procedure in Running the Program or Erase UTM. Initially, set NumPulses = 2, or another small number.
   c. Uncheck the Erase and Fast-Program-Erase tests.
   d. Enter the parameter values for the Vt-MaxGm test, following the procedure in Running the Vt-MaxGm ITM.
   e. Run the test
      i. Double-click on 4Terminal-FloatingGate
      ii. Press the Run button
   f. Check the graph on the Vt-MaxGm test. It will likely be too low on the first few runs, but note the total number of pulses sent to the DUT. Rerun test until the V_T has met the target value, and note the total pulse width to use to program the device, using either the Program or Fast-Program-Erase tests.
   g. Repeat above with the Erase test, feeding final results into the Erase and Fast-Program-Erase tests.
   h. Ensure that the erase parameters are fully erasing the DUT
      i. Set the parameters in the Fast-Program-Erase test. Set NumPulses = 10.
      ii. Uncheck the Program and Erase tests.
      iii. Double-click 4Terminal-FloatingGate. Press run
      iv. Note the V_T.
      v. Change NumPulses = 100 or a larger number
      vi. Double-click 4Terminal-FloatingGate. Press Append.
      vii. Note the V_T. If the V_T value for the tests are similar, then the erase pulse is fully erasing the DUT.

Running the Program or Erase UTM

These tests are located in all of the Flash projects. It outputs a number of pulses (Parameter = NumPulses) with a shape shown similar to Figure 4-74, on a number of pulse channels (Parameters = NumPulseTerminals, PulseTerminals).

1. Enter the number of pulse waveforms required into NumPulseTerminals. This must be a minimum of 1 channel, up to a maximum of the number of channels available. For a typical Flash system with 2 4205-PG2 cards, there are 4 pulse channels available.
2. Enter which pulse channels will be used into PulseTerminals. This is a string of channels, in the form "VPU1CH1,VPU1CH2,VPU2CH1,VPU2CH2". VPU1 is the 4205-PG2 in the lowest-numbered slot (right-most slot when looking at back of 4200 chassis). The characters are all capitalized and each channel is separated by a comma. No spaces are allowed in the PulseTerminals string.
3. Enter the values in the five arrays that define the pulse shape, referring to Figure 4-74 and the instructions in Entering Segment Arb values into UTM array parameters The number of non-blank entries in the array must match NumPulseTerminals, as shown in Figure 4-76, NumPulseTerminals = 4, and the size each array (PulseVoltagesSize, PrePulseDelays, TransitionTimesSize, PulseWidthsSize, PostPulseDelaysSize) are also 4.
   a) PulseVoltages: Use a positive value for a waveform similar to Figure 4-74. If a negative pulse is required, use a negative voltage value. To put a pulse channel into a disconnected, or high impedance, state, use -999.
   b) PrePulseDelays, TransitionTimes, PulseWidths, PostPulseDelays: The minimum time is 20E-9 (20 ns). 0 (zero) is not a valid input value. The maximum time is 1 s.
4. Enter the number of pulses into NumPulses. This parameter determines the number of pulses that will be output each time the test is run.
5. Enter the number of SMUs that are used as Bias Terminals into NumSMUBiasTerminals. An example of using an SMU as a bias terminal is shown in Figure 4-68. The 4th SMU in Figure 4-68 is a dedicated connection to a bit line on the array DUT. During a pulse test, such as Program or Erase, this SMU would output a DC voltage that would provide power to the drain terminal of the first column of the array.
6. Enter the SMU IDs for the SMU(s) used as a bias into SMUBiasTerminals. For the configuration in Figure 4-68, SMUBiasTerminals = SMU4.
7. Enter the voltages in the array SMUBiasVoltages. These are the voltages for the SMUs listed in SMUBiasTerminals. The number of non-blank entries in the array must match NumSMUTerminals.
8. Enter the number of SMUs that are sharing a cable with a pulse channel into NumSharedSMUs. Sharing simply means that one pulse and one SMU signal are combined to a single DUT terminal. Figure 4-68 shows that 3 pairs of SMU/pulse channels are shared. Note the SMA tees on each of the top 3 SMUs that incorporate both a pulse channel and a SMU signal into a single cable to a DUT terminal. Supplying the shared SMU information allows the software to open the SMU relay during the pulse output, that is necessary to permit good pulse fidelity. If a switch matrix is used in the configuration (Figure 4-69), then use NumSharedSMUs = 0.
9. Enter the SMU IDs for the SMU(s) sharing a cable with a pulse channel into SharedSMUs. For the configuration in Figure 4-68, SharedSMUs = SMU1,SMU2,SMU3. There are no spaces allowed in the SharedSMUs string.
10. Press the green triangle Run button to output the pulses.
11. Check the Data tab in the Sheet control. The single_pulse_flash value should be 0, indicating that there were no errors. No measurements are taken in this test, so there is no data to graph.
12. If single_pulse_flash is non-zero, pulses are not being output, or there are error messages in the Project Messages pane, see Troubleshooting section.

Table 4-87
Parameter values for Program or Erase UTM for 4 or 2 channel configurations

Parameter	Value for 4 channel test				Value for 2 channel test	
NumPulseTerminals	4				2	
PulseTerminals	VPU1CH1, VPU1CH2, VPU2CH1, VPU2CH2				VPU1CH1, VPU1CH2	
PulseVoltages	0	7	0	0	0	7
PrePulseDelays	1E-6	1E-6	1E-6	1E-6	1E-6	1E-6
TransitionTimes	3E-7	3E-7	3E-7	3E-7	3E-7	3E-7
PulseWidths	5E-6	5E-6	5E-6	5E-6	5E-6	5E-6
PostPulseDelays	2E-6	2E-6	2E-6	2E-6	2E-6	2E-6
NumPulses	1				1	
NumSMUBiasTerminals	0				0	
SMUBiasTerminals						
SMUBiasVoltages						
NumSharedSMUs	4				2	
SharedSMUs	SMU1,SMU2,SMU3,SMU4				SMU1,SMU2	

NOTE *Channel count refers to the number of pulse+SMU channels with a direct connect setup. A setup with 4 channels of each pulse and SMU is in Figure 4-67. All channels in group must have the same total time.*

Running the Fast-Program-Erase UTM

This test is used in all of the Flash projects. It outputs a number of pulse waveforms (Parameter = NumPulses) with a shape shown similar to Figure 4-75, on a number of pulse channels (Parameters = NumPulseTerminals, PulseTerminals).

1. Enter the number of pulse waveforms required into NumPulseTerminals. This must be a minimum of 1 channel, up to a maximum of the number of channels available. For a typical Flash system with 2 4205-PG2 cards, there are 4 pulse channels available.
2. Enter which pulse channels will be used into PulseTerminals. This is a string of channels, in the form “VPU1CH1,VPU1CH2,VPU2CH1,VPU2CH2”. VPU1 is the 4205-PG2 in the lowest-numbered slot (right-most slot when looking at back of 4200 chassis). The characters are all capitalized and each channel is separated by a comma. No spaces are allowed in the PulseTerminal string.
3. Enter the values in the ten arrays that define the pulse shape, referring to Figure 4-75 and the instructions in Entering Segment Arb values into UTM array parameters. The number of non-blank entries in the array must match NumPulseTerminals, as shown in Figure 4-76, NumPulseTerminals = 4, and the size each array (Pulse1VoltagesSize, PrePulse1Delays, TransitionTimesPulse1Size, Pulse1WidthsSize, PostPulse1DelaysSize, Pulse2VoltagesSize, PrePulse2Delays, TransitionTimesPulse2Size, Pulse2WidthsSize, PostPulse2DelaysSize) are also 4.
   a. PulseVoltages: Use a positive value for a waveform similar to Figure 4-75. If a negative pulse is required, use a negative voltage value. To put a pulse channel in to a disconnected, or high impedance, state, use -999.
   b. PrePulseDelays, TransitionTimes, PulseWidths, PostPulseDelays: The minimum time is 20E-9 (20 ns). 0 (zero) is not a valid input value. The maximum time is 1 s.
4. Enter the number of pulses into NumPulses. This parameter determines the number of program and erase pulse waveforms that will be output each time the test is run.
5. Enter the number of SMUs that are used as Bias Terminals into NumSMUBiasTerminals. An example of using an SMU as a bias terminal is shown in Figure 4-68. The 4th SMU in Figure 4-68 is a dedicated connection to a bit line on the array DUT. During a pulse test, such as Program or Erase, this SMU would output a DC voltage that would provide power to the drain terminal of the first column of the array.

6. Enter the SMU IDs for the SMU(s) used as a bias into SMUBiasTerminals. For the configuration in Figure 4-68, SMUBiasTerminals = SMU4.
7. Enter the voltages in the array SMUBiasVoltages. These are the voltages for the SMUs listed in SMUBiasTerminals. The number of non-blank entries in the array must match NumSMUTerminals.
8. Enter the number of SMUs that are sharing a cable with a pulse channel into NumSharedSMUs. Sharing simply means that one pulse and one SMU signal are combined to a single DUT terminal. Figure 4-68 shows that 3 pairs of SMU/pulse channels are shared. Note the SMA tees on each of the top 3 SMUs that incorporate both a pulse channel and a SMU signal into a single cable to a DUT terminal. Supplying the shared SMU information allows the software to open the SMU relay during the pulse output, that is necessary to permit good pulse fidelity. If a switch matrix is used in the configuration (Figure 4-69), then use NumSharedSMUs = 0.
9. Enter the SMU IDs for the SMU(s) sharing a cable with a pulse channel into SharedSMUs. For the configuration in Figure 4-68, SharedSMUs = SMU1,SMU2,SMU3. There are no spaces allowed in the SharedSMUs string.
10. Press the green triangle Run button to output the pulses.
11. Check the Data tab in the Sheet control. The double_pulse_flash value should be 0, indicating that there were no errors. No measurements are taken in this test, so there is no data to graph.
12. If double_pulse_flash is non-zero, pulses are not being output, or there are error messages in the Project Messages pane, see Troubleshooting section.

Running the SetupDC UTM

This UTM disconnects the PG2 channels by opening solid state relays. This is necessary when using the direct connect method (Figure 4-67 and Figure 4-68), to ensure that a subsequent SMU test is only connected to the DUT terminals.

1. Enter the number of shared terminals into SharedPulseTerminals. Sharing means that a single cable provides either a pulse or a SMU signal to a DUT terminal.
2. Enter the Pulse channel IDs for the VPU channels sharing a cable with a SMU into SharedPulseTerminals. For the configuration in Figure 4-68, SharedPulseTerminals = VPU1CH1,VPU1CH2,VPU2CH1VPU2CH2. There are no spaces allowed in the string.
3. Check the Data tab on the Sheet tab, configure_dc_flash = 0. If the value is non-zero, or there are error messages in the Project Messages pane, see Troubleshooting section.

Running the Vt-MaxGm ITM

This test performs a DC V_G-I_D sweep on the DUT and extracts the threshold voltage (V_T). The VT results for the first run are shown on the graph, in the lower left corner. V_T values for each test (run or append) is shown in each tab, in the right-most column headed “VT.” This test can be run after setting the device in any state, using the Program, Erase and/or Fast-Program-Erase UTMs. This test provides the V_T, but does not determine an appropriate, or target, V_T, that is usually provided by historical performance, a review of the device structure, or the device engineer.

1. Enter the voltage values for each SMU. Defaults have a voltage sweep on the gate, a fixed DC bias on the drain, and 0V or a ground unit (GNDU) signal for the source and bulk.
2. Once the test is run, review the graph or sheet results.

Running the ConPin-Pulse or ConPin-DC UTM (Switch projects only)

This test routes the desired pulse or SMU signals to the DUT by closing switches on a switch matrix card. See Switch matrix connection to array DUT for connection and switch matrix setup instructions. The UTM entries for TermIdStr1-TermIdStr8 Pin1-Pin8 define which instrument (SMU or VPU channel) get connected to which output pin.

1. Enter a value for OpenAll. The default value is 1, that opens all switches. The remaining parameters define which switches to close. If more than 8 closures are required for a test, use 2 ConPin tests, setting the second ConPin test OpenAll = 0, to ensure the first ConPin switch settings are not cleared. See Figure 4-82 for the screenshot of the ConPin parameters.
2. Enter values for TermIdStr1 and Pin1. This first pair of parameters determines which instrument, either SMU or VPU channel, gets output. For a SMU to be output on the 1st output of the switch matrix, TermIdStr1 = SMU1 and Pin1 = 1. If no connection is desired set Pin1 = 0. Another way to set the connections is to use the GUI. To use the GUI, click the GUI button (Figure 4-82) that displays the dialog shown in Figure 4-83. Click the desired crosspoint closures. Notice that the Input strings shown in the GUI must match the labels supplied in the KCON setup for the switch matrix card (Figure 4-71).
3. Continue to enter values for the remaining 7 pairs of TermIdStr and Pins.
4. Running the test will set the switch closures as specified.

FlashEndurance-NAND project

FlashEndurance-NOR project

FlashEndurance-switch project

These three projects are similar and stress the DUT with a number of Program+Erase waveforms, then periodically measures the V_T. The purpose of these projects is to determine the lifetime of the DUT, based on the number of Program+Erase cycles withstood by the device before a certain amount of shift, or degradation, in the V_T or other measurement. The waveforms may be unique for each pulse channel, and are defined in the separate Kpulse program (see Section 13 of the Model 4200-SCS Reference Manual and Using Kpulse to create and export Segment ArbTM waveforms) and saved to files. These files are specified for each pulse channel in the test. The number and interval for the pulse stresses are set, as well as the desired SMU measurements. The typical measurement is a V_T extraction based on a V_G-I_D sweep, but any type of DC test may be configured.

These projects support from 1 to 8 pulse channels to support typical 4-terminal devices, as well as higher pin count devices or array test structures. The 8-terminal testing requires four Model 4205-PG2 cards and, for most tests, a compatible external switch matrix.

The difference between the FlashEndurance-NAND and FlashEndurance-NOR are the difference in the typical pulse widths and levels specific to the DUT type. The FlashEndurance-Switch project is a generic example of the Flash testing described above, but adds support for an external Keithley switch matrix. Example results for the Endurance tests are shown in Figure 4-85 and Figure 4-61.

FlashEndurance-NAND tests

Program

SetupDC-Program

Vt-MaxGm-Program

Erase

SetupDC-Erase

Vt-MaxGm-Erase

The project navigator for FlashEndurance-NAND is shown in Figure 4-84. Stressing for the FlashEndurance-NAND tests are configured from the Subsite Setup tab for the FlashEndurance subsite plan. The default setup (shown in Figure 4-85 and Figure 4-86) uses Segment Arb waveforms to perform log stressing that ranges from 1 to 100,000 counts.

The Segment Arb waveform files (Flash-NAND-Vg-ksf and Flash-NAND-Vd-ksf) used for stressing are loaded into the Device Stress Properties window shown in Figure 4-86. The stress properties window is opened by clicking the **Device Stress Properties** button in Figure 4-85. Example results for the Endurance tests are shown in Subsite Graph tab (see Figure 4-87).

Figure 4-84
FlashEndurance-NAND project plan

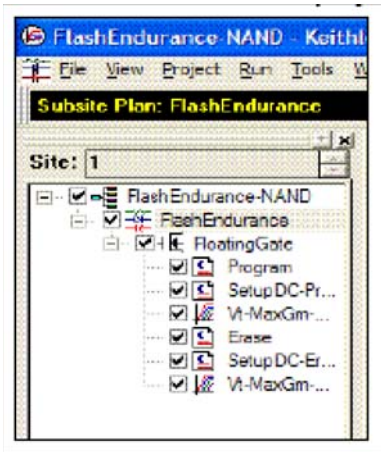


Figure 4-85
FlashEndurance-NAND project – Subsite Plan tab

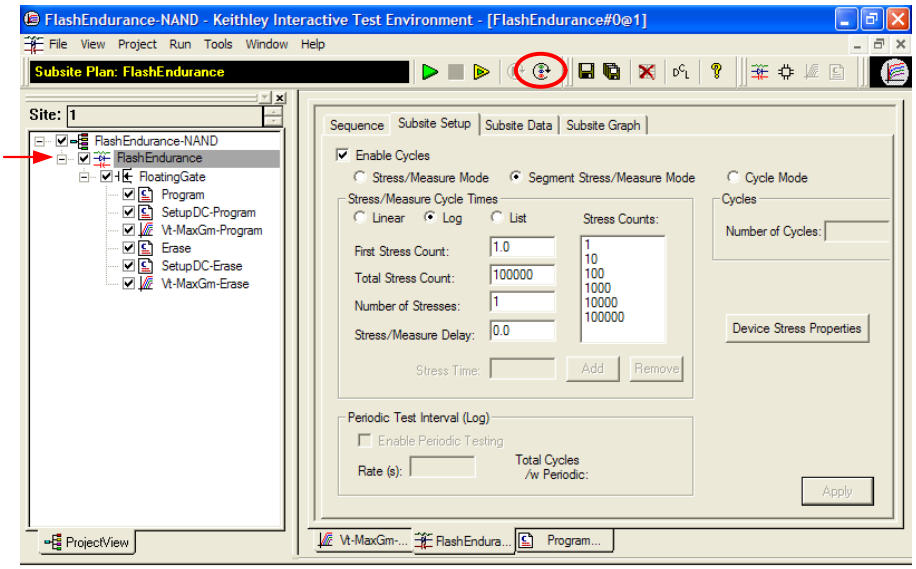


Figure 4-86
FlashEndurance-NAND project – Device Stress Properties

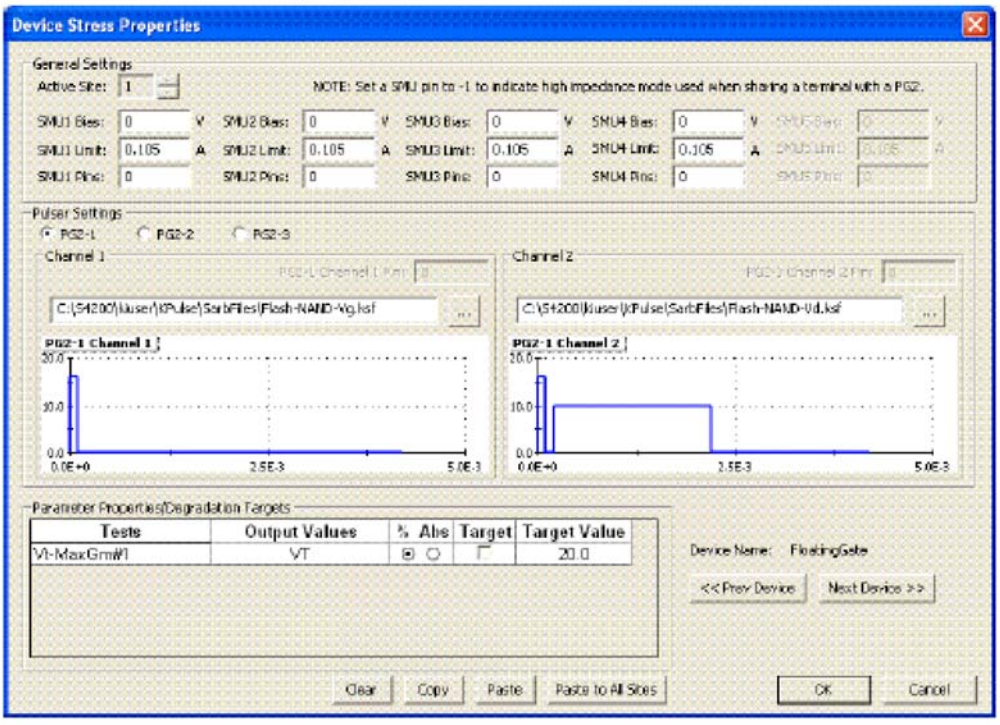
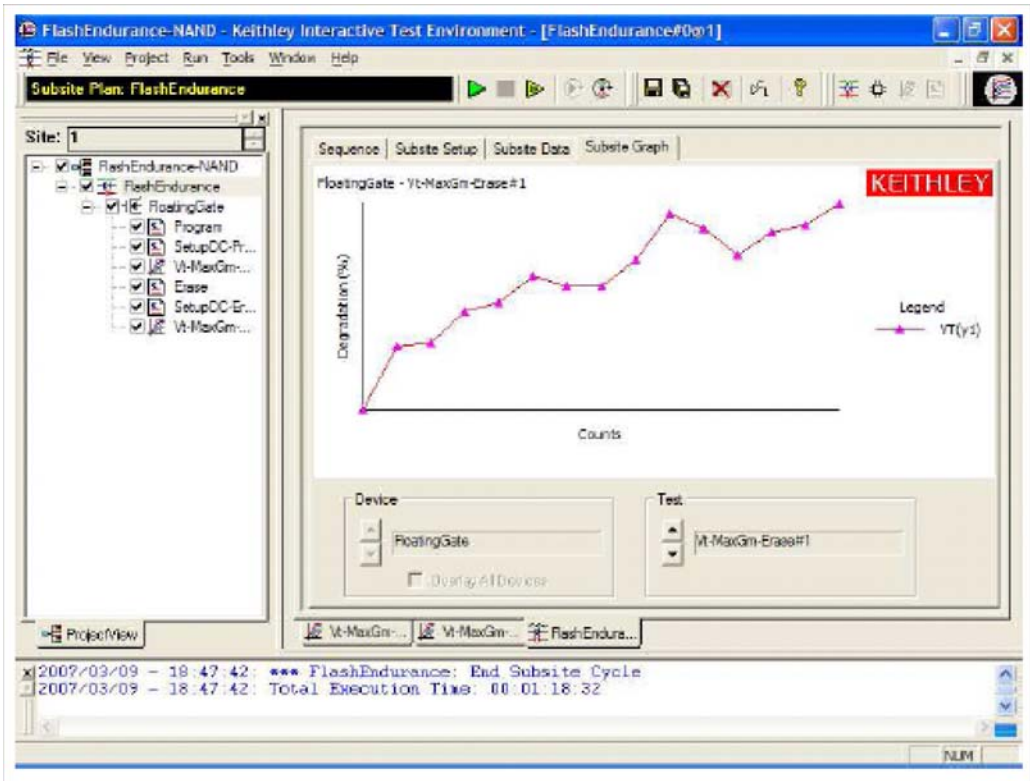
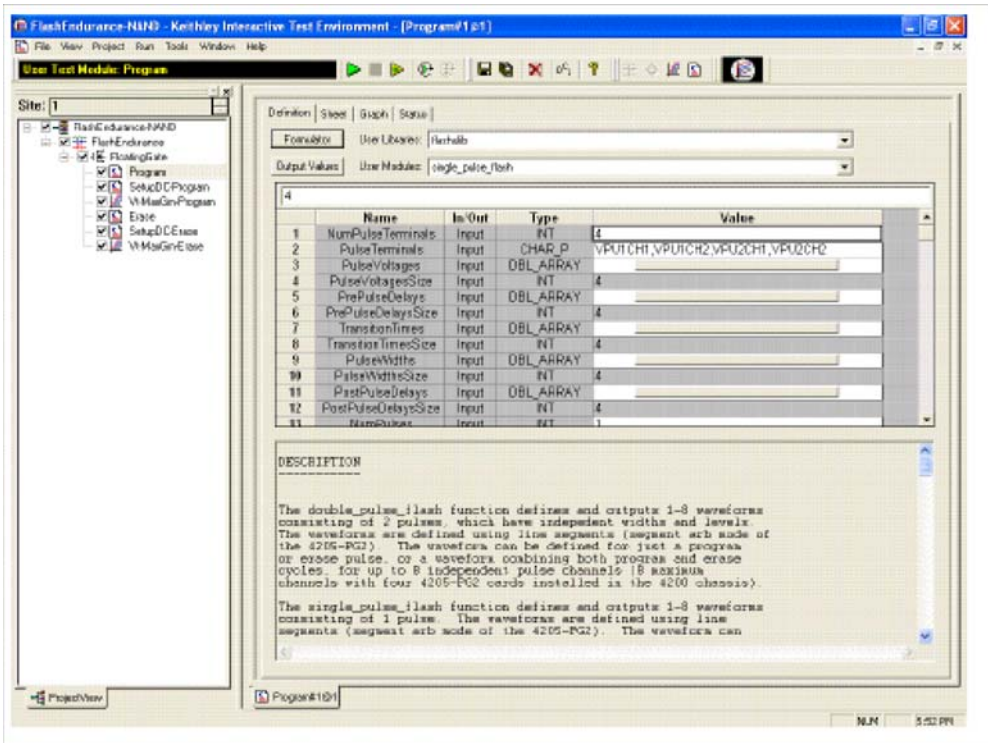


Figure 4-87
FlashEndurance-NAND project – Subsite Graph tab



Program test – The Definition tab for this test is shown in Figure 4-88. This test uses a partially predefined Segment Arb waveforms, see Figure 4-74, to program a flash memory device and identical to the Program UTMs included in the other Flash projects.

Figure 4-88
FlashEndurance-NAND project – Program Definition tab

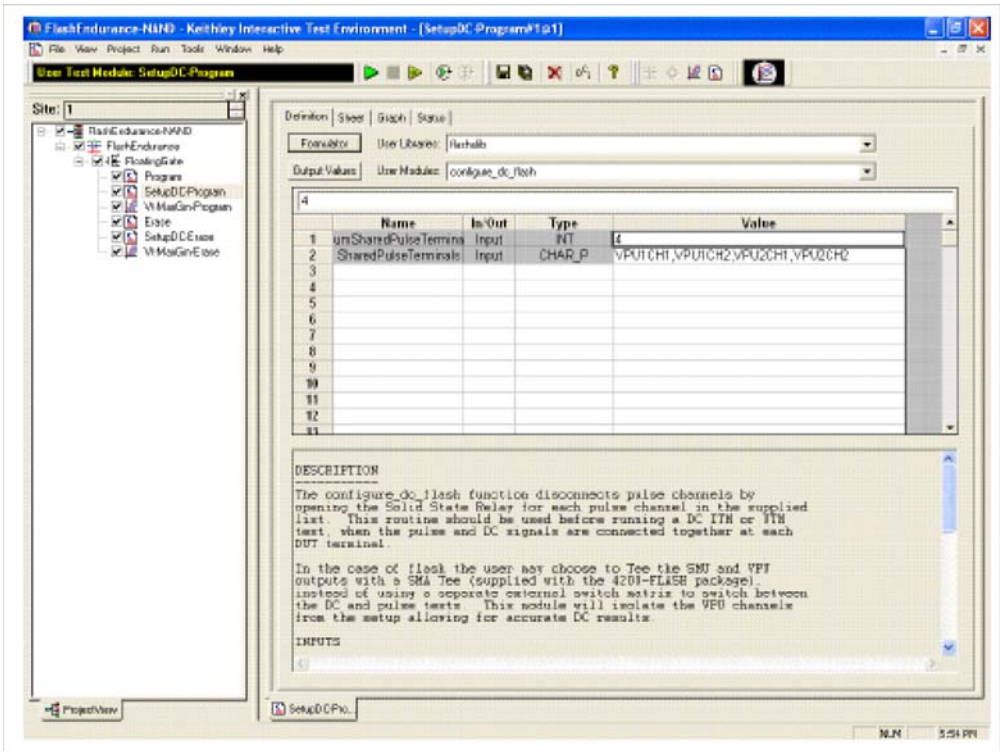


SetupDC-Program test – The Definition tab for this test is shown in Figure 4-89. This test isolates the VPU outputs from the DUT. It does this by opening the High Endurance Output Relay for each VPU channel. Disconnecting the VPU channels allows for accurate DC results.

The SetupDC test is a UTM that should be used when using a directly wired DUT, without an external switch matrix. SetupDC disconnects the PG2 channels from the DUT to permit proper operation of any subsequent DC measurements.

When using a switch matrix, a ConPin test is used (see the `config` LPT function in Section 8 of the Reference Manual) to set the appropriate matrix connections prior to any DC tests.

Figure 4-89
FlashEndurance-NAND project – SetupDC Definition tab



Vt-MaxGm-Program test – This test is used to perform a DC voltage sweep on the gate of the DUT and measure the drain current at each sweep step. The default Definition tab for this test is shown in Figure 4-90. SMU3 is configured to perform a 101 point sweep from 0 to 5V in 50mV steps.

SMU1 is configured to DC bias the drain at 0.5V and measure current at each step of the sweep. The results of the test are shown in the Graph tab (Figure 4-91). The Vt-MaxGm tests may be replaced with another Vt or DC test. Or, additional DC tests may be added after this test.

Figure 4-90
FlashEndurance-NAND project – Vt-MaxGm-Program Definition tab

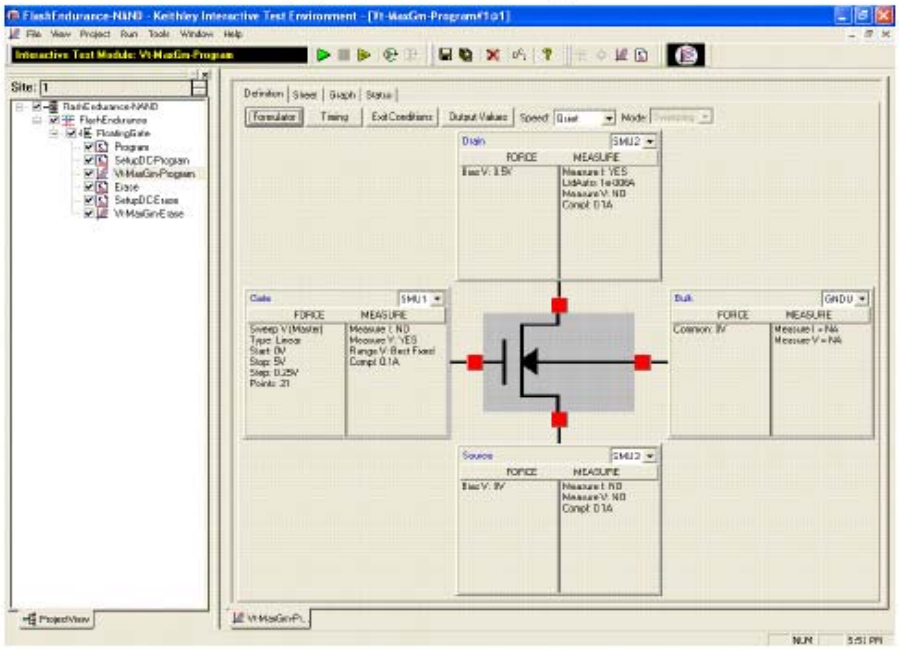
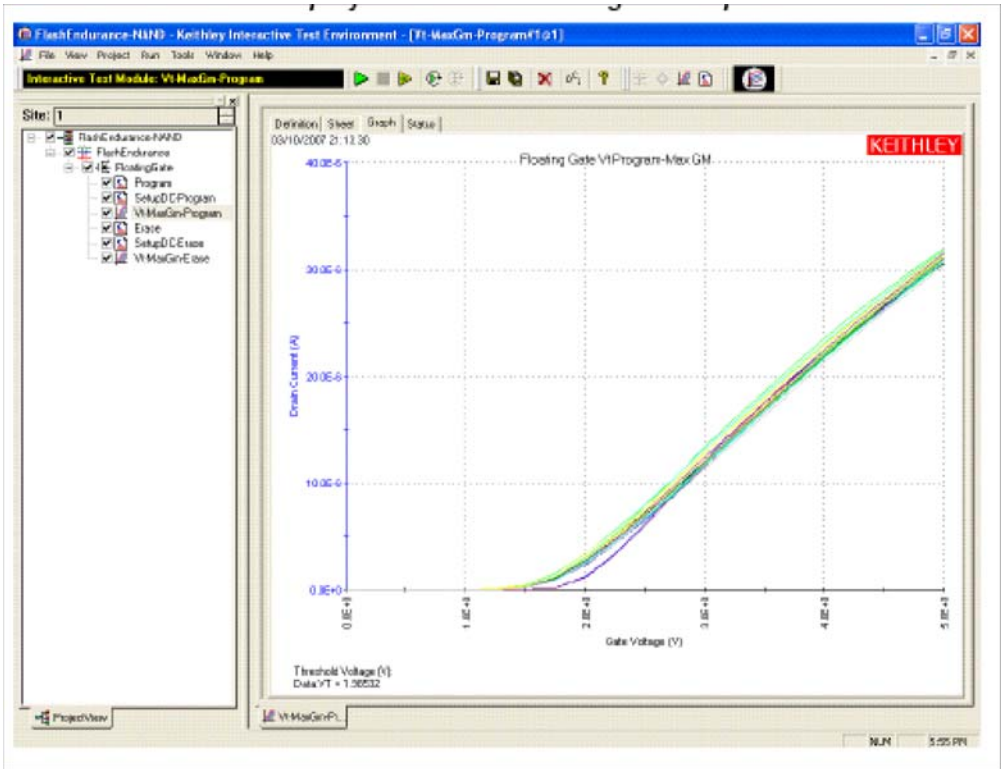
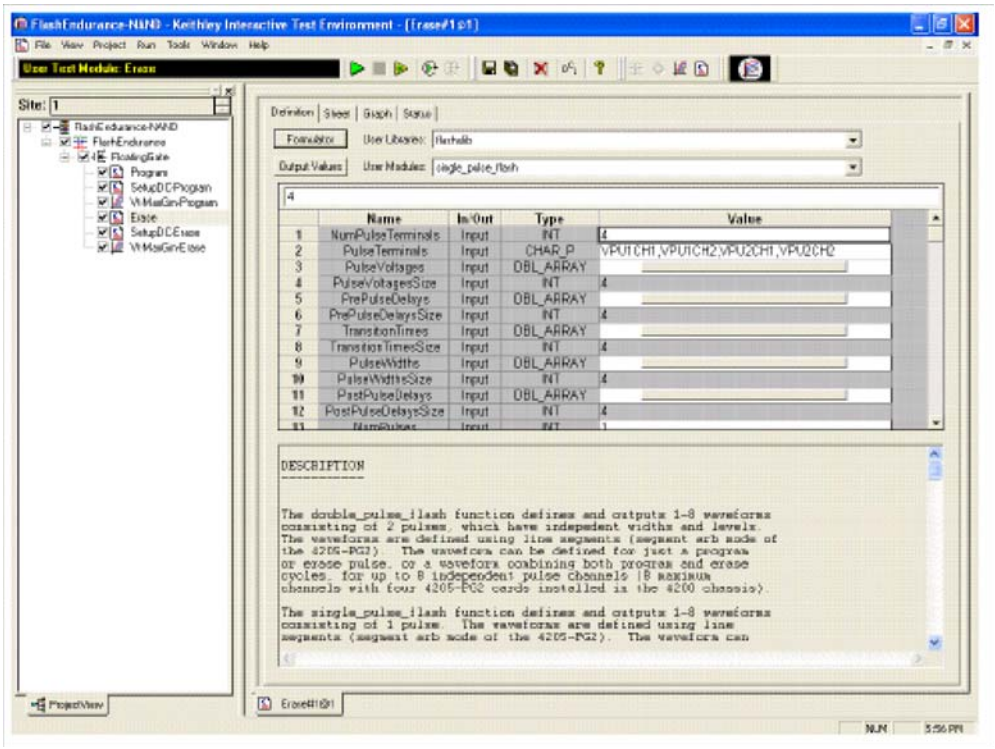


Figure 4-91
FlashEndurance-NAND project – Vt-MaxGm-Program Graph tab



Erase test – This test uses Segment Arb waveforms to program a flash memory device. The default Definition tab for this test is shown in Figure 4-92.

Figure 4-92
FlashEndurance-NAND project – Erase Definition tab



SetupDC-Erase test – This test isolates the VPU outputs from the DUT. It does this by opening the High Endurance Output Relay for each VPU channel. Disconnecting the VPU channels allows for accurate DC results.

Vt-MaxGm-Erase test – This test is used to perform a DC voltage sweep on the gate of the DUT and measure the drain current at each sweep step. SMU3 is configured to perform a 101 point sweep from 0 to 5V in 50mV steps. SMU1 is configured to DC bias the drain at 0.5V and measure current at each step of the sweep.

The Vt-MaxGm tests may be replaced with another Vt or DC test. Or, additional DC tests may be added after this test.

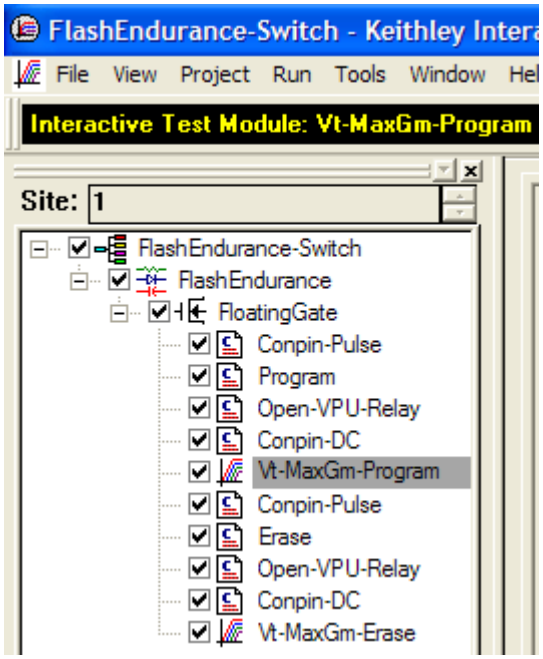
FlashEndurance-NOR tests

The FlashEndurance-NOR project has tests similar to the FlashEndurance-NAND project, with defaults for NOR type floating gate DUTs.

FlashEndurance-switch tests

The FlashEndurance-switch project, Figure 4-93, has similar tests to the FlashEndurance-NAND, with defaults for using a switch matrix for more complex multi-DUT addressable test structures (see Figure 4-69). Also note the additional test, Open-VPU-Relay, added prior to Conpin-DC tests. This step ensures that VPU channels will not be inadvertently connected to a device terminal when the SMU testing is performed.

Figure 4-93
FlashEndurance-Switch project



Running a FlashEndurance or FlashDisturb Project

This section explains how to use the following Flash projects:

FlashEndurance-NAND

FlashEndurance-NOR

FlashEndurance-Switch

FlashDisturb-NAND

FlashDisturb-NOR

FlashDisturb-Switch

These Flash projects use a small number of tests and methods. This section will explain the tests and how to set parameter values for Endurance testing or Disturb testing.

Before using any of these projects, determine the appropriate pulse voltages and widths by first using the appropriate project, using the procedures in Running the Flash-NAND, Flash-NOR or Flash-Switch Project.

The Endurance and Disturb projects include everything from the corresponding Flash-NAND, Flash-NOR, or Flash-Switch projects. To use an Endurance or Disturb test, ensure that each test in the Project Navigator is functioning properly by following the procedures below. After setting up all of the tests, the information below will continue to explain the setup for the stress/measure looping that is the core of any endurance or disturb test.

1. If system connections have not been made, follow the instruction in Running any Flash Project for the first time.
2. If KITE is not running, start KITE by double-clicking the KITE icon on the 4200 desktop.
3. Open the appropriate KITE Flash project.
   a. Within KITE, click FILE > Open Project. If the dialog window is not displaying the _Memory folder, move up one or two levels to the display the Projects directory. Double-

click the "_Memory" folder, then double-click the desired Flash test folder (see list above this procedure), then double-click the appropriate *.kpr file to open the desired 4200 Flash project.

b. KITE should resemble Figure 4-88.

4. Touch-down or connect the Device Under Test (DUT).
5. Verify setup and connection by running Vt-MaxGm test
   a. Set appropriate voltages
   b. Run the test by clicking the yellow and green triangle Append button.
   c. Ensure that the V_G-I_D and V_T results are reasonable.
6. Determine the appropriate pulse voltage levels.
   a. Review the section Pulse Waveforms for NVM Testing [Editor: please add a link to this section in this Apps Manual]
   b. Recall that pulse voltage levels on the gate will double
      i. For example, using PulseVoltage = 2 will result in V_G = 4V for a typical high-impedance (1 k Ω) terminal.
   c. Use oscilloscope to determine appropriate PulseVoltage values for the desired V_G and V_D, making sure to use the 1M Ω input setting on the oscilloscope.
      i. Drain: Connect the oscilloscope probe across the drain-source of the DUT.
      ii. Use the Program and Erase UTMs to output pulses, while using the oscilloscope to measure the pulse height. Iterate by modifying the PulseVoltages to reach the target voltage.
7. Once the appropriate voltage levels is determined, the appropriate pulse width may be determined by iteratively outputting pulses while occasionally measuring the V_T.
   a. Start by using a pulse width that is shorter than the expected PW. For example, if 20 μs is the expected PW, try using a 2 μs PW.
   b. Enter the parameter values into the Program UTM, following the procedure in Running the Program or Erase UTM. Initially, set NumPulses = 2, or another small number.
   c. Uncheck the Erase and Fast-Program-Erase tests.
   d. Enter the parameter values for the Vt-MaxGm test, following the procedure in Running the Vt-MaxGm ITM.
   e. Run the test
      i. Double-click on 4Terminal-FloatingGate
      ii. Press the Run button
   f. Check the graph on the Vt-MaxGm test. It will likely be too low on the first few runs, but note the total number of pulses sent to the DUT. Rerun test by following the previous Run the Test steps. Once the V_T has met the target value, note the total pulse width to use to program the device, using either the Program or Fast-Program-Erase tests.
   g. Repeat above with the Erase test, feeding final results into the Erase and Fast-Program-Erase tests.
   h. Ensure that the erase parameters are fully erasing the DUT
      i. Set the parameters in the Fast-Program-Erase test. Set NumPulses = 10.
      ii. Uncheck the Program and Erase tests.
      iii. Double-click 4Terminal-FloatingGate. Press run
      iv. Note the V_T.
      v. Change NumPulses = 100 or a larger number
      vi. Double-click 4Terminal-FloatingGate. Press Append.
      vii. Note the V_T. If the V_T value for the tests are similar, then the erase pulse is fully erasing the DUT.

The following link to procedures for these UTM and ITM tests:

Running the Program or Erase UTM

Running the Fast-Program-Erase UTM

Running the SetupDC UTM

Running the Vt-MaxGm ITM

Running the ConPin-Pulse or ConPin-DC UTM (Switch projects only)

Running Endurance or Disturb looping

The Endurance or Disturb testing is essentially a stress/measure test. The stress portion applied a number of pulse waveforms to the DUT, then periodically measurements are performed.

1. Double-click on FlashSubsite (red arrow in Figure 4-85), then click the Subsite Setup tab. The screen should resemble Figure 4-85.
2. Ensure that the Segment Stress/Measure Mode is chosen.
3. Determine the stress intervals and how often the measurements are performed. Each entry in the Stress Counts box is the number of waveforms that will be output. After the listed number of waveform counts is output, measurements are peformed. All checked boxes in the Project Navigator will be run after each stress interval.
   a. Choose Linear, Log or List
   b. Enter the First stress count, that must be at least 1.
   c. Enter the Total Stress Count, that is the last stress interval output.
   d. Number of stresses is the number of stress intervals.
      i. For linear, the number of total stress intervals
      ii. For log, the number of stress intervals per decade count of stress counts
   e. Press the Apply button to see the updated Stress Counts and intervals.
4. Click the Device Stress Properties button, that will display something similar to Figure 4-86.
5. The General Settings show the SMU settings during the stress portion of the test. These settings are necessary when using an array DUT structure, either direct connect (Figure 4-68) or using a switch matrix (Figure 4-69).
   a. If a SMU bias is required, then set the voltage and current limit.
   b. If using shared cabling, or SMU/VPU pairings, set all Pins entries = -1, to disconnect the SMUs during the stress, allowing the pulse signals to properly reach the DUT terminals.
   c. If using a switch matrix, set the pin connection. If no connection is required, input 0.
6. Pulser Settings configure the waveforms used during the stress. See Using Kpulse to create and export Segment ArbTM waveforms to create the desired multi-level pulse waveforms.
   a. Click on the "…" button for each available pulse channel. Choose the desired waveform, previously created and exported, from the available list. Each channel must have an associated .ksf waveform and each waveform should have the same duration.
7. Parameter properties show which parameters are graphed in the Subsite graph, and if any test should end early. If a test should end after a certain VT shift, either an absolute voltage shift, or a % shift follow the below:
   a. Choose % or Abs
   b. Check the box
   c. Enter the Target value.
8. Before running the test for the first time, it is recommended to try out the project on a scrap device. Ensure that the project navigator is showing the FlashEndurance entry highlighted, as shown in Figure 4-85. Then click the Run Test/Subsite button (in the red oval on Figure 4-85). Ensure that the test cycles through each test in the Project Navigator, and that data is input into the Subsite Data tab.
9. Move to a fresh device and click the Run Test/Subsite icon.
10. If errors or unexpected operation occurs, see the Error Codes and Troubleshooting sections.

FlashDisturb-NAND project

FlashDisturb-NOR project

FlashDisturb-switch project

These three projects are similar and use the shared stress-measure looping capability of the FlashEndurance projects. The purpose of the Disturb test is to pulse stress a device in an array test structure, then perform a measurement, such as V_T, on a device adjacent to the pulsed device. The goal is to measure the amount of V_T shift in adjacent cells, either in the programmed or erased states, when a nearby device is pulsed with either a Program, Erase, or Program+Erase waveforms. The typical measurement is a V_T extraction based on a Vg-Id sweep, but any type of DC test may be configured.

The difference between the FlashDisturb-NAND and FlashDisturb-NOR are the typical pulse widths and levels specific to the DUT type. The FlashDisturb-switch is a generic example of the Flash testing described above, but adds support for an external Keithley switch matrix.

FlashDisturb tests

Program

SetupDC-Program

Vt-MaxGm-Program

Erase

SetupDC-Erase

Vt-MaxGm-Erase

The six tests listed above are the same ones used for endurance testing (see "FlashEndurance-NAND tests" for details).

Stressing for the disturb tests are configured from the Subsite Setup tab for a disturb project subsite plan. The default subsite setup for FlashDisturb-NAND (shown in Figure 4-94) uses Segment Arb waveforms, defined and saved to file using Kpulse, to perform log stressing that ranges from 100,000 to 1,000,000 counts.

The Segment Arb waveform files (Flash-NAND-Vg.ksf and Flash-NAND-Vd.ksf) used for stressing are loaded into the Device Stress Properties window shown in Figure 4-95. The stress properties window is opened by clicking the **Device Stress Properties** button in Figure 4-94.

Figure 4-94
FlashDisturb-NAND project – Subsite Setup tab

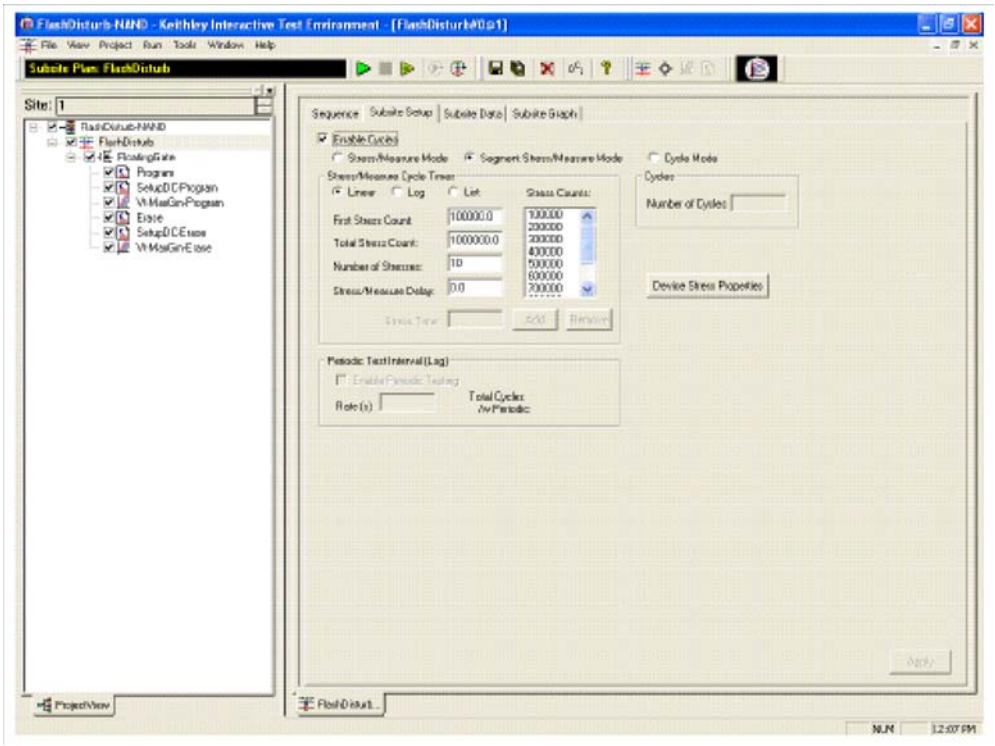
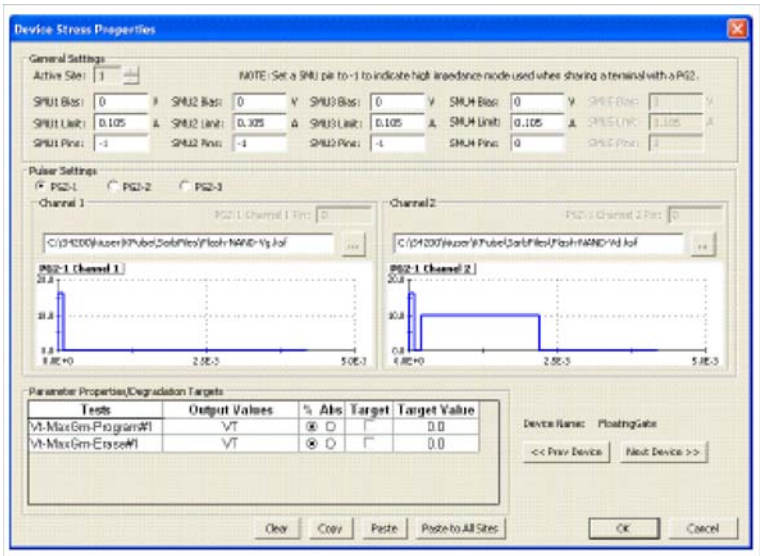


Figure 4-95
FlashDisturb-NAND project – Device Stress Properties



Explanation of Flash UTM Parameters

NumPulseTerminals	(int) The number of pulse terminals, or pulse channels, to use for the test. The number of pulse terminals ranges from 1 to 8.
PulseTerminals	(char *) A string representation of all the VPU channels being used in the test, matching the number given in NumPulse. For example if the setup is such that VPU1 Channel 1 and VPU2 Channel 2 are being used, then PulseTerminals should look like

	this: "VPU1CH1,VPU2CH2". There are no spaces in this list of channels.
Pulse1Voltages	(double) Array of voltage values for the pulse height (0V referenced) of first pulse on each pulse channel. Valid values range from -20V to +20V. Please note that all voltage levels assume a 50 Ohm load. In order to float a channel (disconnect pulse output from a DUT pin), using the Solid State Relay, use -999. Minimum time required for a SSR open or close is 100 us.
PrePulse1Delays	(double) Array of time values used as a delay before the first pulse is output. Valid values range from 20ns to 1s in 10ns increments (s).
TransitionTimesPulse1	(double) The amount of time it will take the first pulse to rise/fall (0-100%/100-0%) from the BaseValue (0V) to the given Pulse Voltage. If the pulse voltage level is from -5 to +5V, then the valid transition times are from 20e-9 to 1s in 10ns increments, else if pulse voltage is within -20 to +20V, then valid values range from 100e-9 to 1s in 10ns increments (s).
Pulse1Widths	(double) Array of values defining the pulse widths for the first pulse of each channel. Minimum values are 20 ns to 1s. Pulse width is defined as FWHM, so it includes half of the fall time and half of the rise time (transition time), in seconds.
PostPulse1Delays	(double) Array of time values used as a delay after the first pulse is output (that is, time at the 0V base voltage). Valid values are 20ns to 1s in 10ns increments (s).
Pulse2Voltages	(double) Array of voltage values for the pulse height (0V referenced) of second pulse on each pulse channel. Valid values range from -20V to +20V. Please note that all tests assume a 50Ohm load. In order to float a channel, or disconnect from a DUT pin, using the Solid State Relay, use -999. Minimum time required for a SSR open or close is 100 us.
PrePulse2Delays	(double) Array of time values used as a delay before the second pulse is output. This delay happens after the PostPulse1Delays. Valid values range from 20ns to 1s in 10ns increments (s).
TransitionTimesPulse2	(double) The amount of time it will take the second pulse to rise/fall (0-100%/100-0%) from the BaseValue (0V) to the given Pulse Voltage. If the pulse voltage level is from -5 to +5V, then the valid transition times are from 20e-9 to 1s in 10ns increments, else if pulse voltage is within -20 to +20V, then valid values range from 100e-9 to 1s in 10ns increments (s).
Pulse2Widths	(double) Array of values defining the pulse widths for the second pulse in each channel. Minimum values are 20 ns to 1s. Pulse width is defined as FWHM, so it includes half of the fall time and half of the rise time (transition time), in seconds.
PostPulse2Delays	(double) Array of time values used as a delay after the second pulse is output (that is, time at the 0V base voltage). Valid values are 20ns to 1s in 10ns increments (s).
NumPulses	(int) The number of pulses to output. Valid range: 1 to (2^31) (about 4.2 billion).
NumSMUBiasTerminals	(int) The number of bias SMUs to include in the test. These are SMUs that are not connected in the SMU+Pulse sharing

	configuration, but additional SMUs that could be used for biasing word or bit lines.
SMUBiasTerminals	(char *) A string representation of all the SMU channels being used in the test. For example if the setup is such that SMU1 and SMU2 are being used for a bias, then SMUBiasTerminals would be: "SMU1,SMU2".
SMUBiasVoltages	(double) Array of SMU bias values used during the test. The values correspond to the number and order in the SMUBiasTerminals string.
NumSharedSMUs	(int) The number of SMUs sharing a connection to the device with a VPU. Sharing a connection means using a Tee to combine a SMU and VPU channel for a DUT terminal, instead of using an external switch matrix. This variable stores the number of shared SMU+Pulse instances.
SharedSMUs	(char *) A string representation of all the shared SMU channels being used in the test. This string is used to disconnect each SMU from the shared cable during pulse output. For example, if SMU1 is connected through a Tee to a pulse channel and SMU2 is also connected to another pulse channel, then the SharedSMUs string would be: "SMU1, SMU2".
SharedPulseTerminals	(int) Number of Pulse channels that are paired with an SMU. This parameter is used in conjunction with SharedPulseTerminals. See Figure 4-60 and Figure 4-67 for examples of a SMU and VPU sharing a cable to a DUT terminal.
SharedPulseTerminals	(char *) A list of pulse channels that each share a cable with a SMU. The list for 2 channels on the lowest numbered VPU would be: VPU1CH1,VPUCH2. There are no spaces or quotation marks in the string. See Figure 4-600 and Figure 4-67 for examples of a SMU and VPU sharing a cable to a DUT terminal.
OpenAll	(int) Value for ConPin test that determines if all matrix switch points are opened, before the desired switch point closures. Using OpenAll = 1 essential resets the switch to an all open state, then the desired switches are closed.

Error Codes

0	No Errors
-16001	Invalid number of pulse terminals
-16002	PulseVoltagesSize has to match the number of pulse terminals
-16003	PrePulseDelaysSize has to match the number of pulse terminals
-16004	TransitionTimesSize has to match the number of pulse terminals
-16005	PulseWidthsSize has to match the number of pulse terminals
-16006	PostPulseDelaysSize has to match the number of pulse terminals
-16007	Invalid number of bias SMU terminals
-16008	Invalid name of shared pulse terminal(s)
-16009	Required bias SMU is not available in current configuration
-16010	Required shared SMU is not available in current configuration
-16011	Required VPU is not available in current configuration

-16012	PrePulseDelay value is out of valid range
-16013	PrePulseDelay value has to be in 10ns increments
-16014	TransitionTime value is out of valid range
-16015	TransitionTime value has to be in 10ns increments
-16016	Pulse level value is out of valid range
-16017	Pulse width value is out of valid range
-16018	PostPulseDelay value is out of valid range
-16019	PostPulseDelay value has to be in 10ns increments

Troubleshooting

Check the Error Codes for additional information.

No pulse output

If pulses are not being output, please check the following:

1. Ensure proper cabling. The trigger interconnections between the pulse cards must match the diagram shown in Figure 4-67, Figure 4-68, or Figure 4-69. The TRIGGER OUT from the pulse card in the lowest numbered slot (right-most slot) must be cabled into TRIGGER IN of the same card, as well as the TRIGGER IN of all adjacent pulse cards.
2. All size values (PulseVoltagesSize, PrePulseDelaysSize, and so on) must match the value of NumPulseTerminals. As shown in Figure 4-65, NumPulseTerminals = 4 and there are 4 entries in: PulseTerminals, PulseVoltages, PrePulseDelays, TransitionTimes, PulseWidths, PostPulseDelays. This rule must be followed for any Program, Erase or Fast-Program-Erase UTM.
3. Ensure that all time-based pulse parameters are not zero. The minimum time interval is 20 ns (20E-9). This applies to the parameters: PrePulseDelays, TransitionTimes, PulseWidths, PostPulseDelays. This rule must be followed for any Program, Erase or Fast-Program-Erase UTM.
4. Ensure that all pulse channel waveforms have the same total time, or period.
   a. To check for a single pulse Program or Erase UTM, add up PrePulseDelays, TransitionsTimes, PulseWidths and PostPulseDelays for each channel. All channels should have the same total. If they do not have the same total time, make them the same by modifying the timing.
   b. To check for a double pulse Fast-Program-Erase UTM, add up PrePulse1Delays, TransitionsTimesPulse1, Pulse1Widths, PostPulse1Delays, PrePulse2Delays, TransitionsTimesPulse2, Pulse2Widths and PostPulse2Delays for each channel. All channels should have the same total. If they do not have the same total time, make them the same by modifying the timing.

Voltage Levels do not match expected values

If the voltage at the DUT terminal is not the expected level, please check the following:

1. The pulse channel is a 50 Ω output and expects a 50 DUT terminal impedance. For a gate, or other high impedance (50 Ω) terminal, the voltage at the terminal will be twice (2x) the value specified. For example, setting PulseVoltage = 2 will result in a 4V level at the DUT gate. See the discussion in 4200 SCS Reference Manual Section 11 - Load Line Effect and Compensation for additional details on the effect of the DUT impedance on the pulse level.
2. If the DUT terminal is the drain, alternate manual methods are appropriate. The most common method for determining the pulse voltage level on the drain is to use an oscilloscope with the scope input impedance set to 1 M. Ensure that the gate voltage level meets the desired value before setting other voltage levels. Modify the PulseVoltage until

the level matched the desired level. The drain voltage level is a function of the drain-source impedance, that is largely determined by the gate voltage.

Index

Symbols

"4terminal-n-fet" selected to run 1-11
"4terminal-n-fet" tests opened in Workspace 1-9
"connect" parameters for "4terminal-n-fet" device 2-10
"connect" test 2-9
"cvsweep" UTM 3-11
"diode" moved to top of Project Navigator 2-12
"diode" moved to top of sequence table 2-11
"id-vg" graphs 3-20
"pgu1-setup" - configure the PGU channel 3-18
"pgu-trigger" test - trigger the burst of stress pulses 3-18
"probesubsites" test descriptions 3-28
"rdson10" added to Project Navigator 2-21
"rdson10" UTM 2-21
"vds-id" graph 1-6

Numerics

4200-PIV-Q 4-64

A

AC stress for WLR 4-61
Add a Device Plan 3-9
Add a new Device Plan to a KITE project 3-8
Add a new Subsite Plan 3-8
Add a new Subsite Plan to a KITE project 3-8
Add a switch card to the system configuration 2-5
Add a switch matrix to the system configuration 2-4
Add a test fixture to the system configuration 2-4
Add new UTM 2-20
Add the "cvsweep" UTM 3-10
Adding a Keithley 590 CV Analyzer to the system configuration 3-6
Adding a probe station 3-14, 3-24
Adding a pulse generator 3-13
Adding a switch matrix 3-13, 3-24
Advanced Applications 2-1
Applications 4-1

B

Buttons to close or reduce size of test documents 3-19

C

Changing the STOPI value 1-4
Charge Pumping 4-2
  Hardware setup 4-3
Charge pumping
  Type of sweeps 4-4
Configuring the switch matrix 3-14, 3-24
Connect SMUs to N-channel MOSFET 3-29
Connect SMUs to NPN transistor 3-29
Connecting the switch matrix 3-15, 3-25
Controlling a CV Analyzer 3-5
  Add a Device Plan 3-8
  Add a Subsite Plan 3-8
  Add a UTM 3-9
  Connections 3-5
  Create a new project 3-7
  Executing the test 3-11
  KCON setup 3-6
  Modifying the "cvsweep" UTM 3-11
Controlling a probe station 3-21
  KCON setup 3-23
  Open the "probesubsites" project 3-26
  Open the project plan window 3-27
  Probe station configuration 3-26
  Prober control overview 3-22
  Running the test sequence 3-31
  Test data 3-32
  Test descriptions 3-27
  Test system connections 3-23
Controlling a pulse generator 3-12
  Compare the test results 3-19
    Overlaying graphs 3-20
  Description of tests 3-16
    "pgu1-init" test 3-17
    "pgu1-setup" test 3-18
    "pgu-trigger" test 3-18

First “connect” test 3-16
First “id-vg” test 3-16
Second “connect” test 3-17
Second “id-vg” test 3-19
Third “connect” test 3-18
KCON setup 3-13
Open the “ivpgswitch” project 3-15
Running the test sequence 3-19
Test system connections 3-12
Controlling a switch matrix 2-2
“connect” test description 2-9
KCON setup 2-4
Open KITE and the “ivswitch” project 2-6
Running test sequences 2-7
Controlling External Equipment 3-1
Controlling external equipment overview 3-2
Copy “Rdson42xx” module as “RdsonAvg” 2-16
Cursors window 1-7
Customizing a user test module (UTM) 2-12
Add a new UTM to the “ivswitch” project 2-20
Add a new UTM 2-20
Connect the “rdson10” UTM to the “RdsonAvg” user module 2-21
Open KITE and load the “ivswitch” project 2-20
Copy “Rdson42XX” to “RdsonAvg” 2-16
Open and modify the “RdsonAvg” user module 2-17
Change a parameter name 2-19
Change the module description 2-19
Modify the user module code 2-18
Open KULT 2-13
Open the “ki42xxulib” user library 2-14
Open the “Rdson42XX” user module 2-15
Save, compile, and build the modified library 2-19
Test description 2-21

D

Default test library folders 3-10
Define new project 3-7
Define the system connections 2-5
Device Plan window for “4terminal-n-fet” 1-10
Devices connected to 707A switching matrix 2-3

E

Execution indicator box 2-12

F

First “connect” test - connects the device to the SMUs 3-16
Flash Memory Testing 4-119
Formulator for “subvt” test 1-3

G

Graph for the “subvt” test 1-2
Graph scale settings 3-20
Graph Settings 1-7
Graph with Cursors 1-8
Graphical analysis 1-5
Display and analyze the “vds-id” graph 1-6
Step 1. Display the graph 1-6
Step 2. Open Graph menu 1-6
Step 3. Enable Cursors 1-7
Step 4. Position Cursor on I-V curve 1-8
Open “default” project and “vds-id” test 1-5
Graphical Data Analysis and Basic Test Sequencing 1-1

I

Initial Cursor position 1-8
Interface Trap Charge 4-4
Interface Trap Density 4-5

K

Keithley Model 590 CV Analyzer DUT connections 3-5
KULT main window 2-13
KULT module window 2-17

M

Model 4200-PIV-A 4-16
Test connections 4-17
Model 8101-PIV schematic 4-19
Model 8101-PIV test fixture 4-19
Modified line fit 1-5
Modified project plan settings 3-27
Module name for Description 2-19

N

NBTI (negative bias temperature instability) 4-61
New order for test sequence table 1-11
New Project menu selection 3-7

O

Open “ki42xxulib” library 2-14
Open “Rdson42XX” module 2-15
Open Formulator window 1-3

P

PGU initialization 3-17
PGU stress pulse specifications 3-18
prbgen user modules 3-22
prober-init 3-28
prober-prompt test and dialog window 3-30
prober-separate 3-30
Program modifications 2-18
Programming and erasing flash memory 4-119
Project Navigator — “4terminal-n-fet” tests 1-9
Project Navigator - “ivpgswitch” project 3-15
Project Navigator - “ivswitch” project 2-6, 2-10
Project Navigator — new execution sequence 1-11
Project Navigator - probesubsites project 3-26
Project Navigator for “ivswitch” project 2-20
Pulse generator configuration 3-13
Pulse IV 4-15
Pulse source-measure UTMs
  DualPulseulib 4-77
    dpulse_burst_count 4-77
    dpulse_current_limit 4-78
    dpulse_delay 4-79
    dpulse_fall 4-79
    dpulse_float 4-80
    dpulse_halt 4-80
    dpulse_init 4-81
    dpulse_load 4-81
    dpulse_output 4-82
    dpulse_output_mode 4-83
    dpulse_period 4-83
    dpulse_range 4-84
    dpulse_rise 4-84
    dpulse_trig 4-85
    dpulse_trig_polarity 4-85
    dpulse_vhigh 4-86
    dpulse_vlow 4-87
    dpulse_width 4-87
  PulseIVulib 4-36
    cal_pulseiv 4-36
    scopeshot_pulseiv 4-49, 4-51
      scopeshot_cal_pulseiv 4-49
    scopeshot_pulseiv_demo 4-53
    Vdid_Pulse_DC_Family_pulseiv 4-39
    vdsid_pulseiv 4-37
    vdsid_pulseiv_demo 4-53
    Vgid_DC_Pulse_pulseiv 4-43
    vgsid_pulseiv 4-47
    vgsid_pulseiv_demo 4-53
  QPulseIVulib 4-88
    CableCompensation_QPulseIV 4-90
    Return codes 4-89
    ScopeShot_FET_QPulseIV 4-91
    Vd_Id_Pulse_DC_Family_QPulseIV 4-94
    Vd_Id_Single_DC_QPulseIV 4-102
    Vd_Id_Single_Pulse_QPulseIV 4-104
    Vg_Id_Pulse_DC_QPulseIV 4-107
    Vg_Id_Single_DC_QPulseIV 4-112
    Vg_Id_Single_Pulse_QPulseIV 4-114

R

Relationships between KULT and KITE and between user libraries, user modules, and UTMs 3-4

S

Sample wafer organization 3-21
Save the system configuration 2-6
Save, compile, and build library 2-19
Saving test data 1-12
Saving the system configuration 3-6, 3-15, 3-25
Second “connect” test - connects the device to the PGU 3-17
Sequencing tests on a single device 1-9
  Change the execution sequence 1-10
    Step 1. Open Device Plan window 1-10
    Step 2. Select and move “ig-vg” 1-11
    Step 3. Apply new execution sequence to the Device Plan 1-11
  Modify tests 1-10
  Open “4terminal-n-fet” tests 1-9
  Open “default” project 1-9
  Run “4terminal-n-fet” test sequence 1-11
    Step 1. Select “4terminal-n-fet” Device Plan 1-11
    Step 2. Click Run button to execute the test sequence 1-12
  Save and export test data 1-12
Sequencing tests on multiple devices 2-10
  Execute the test sequence (Subsite Plan) 2-12
  Open “ivswitch” project 2-10
    Modify test sequence 2-11
Setting the Model 590 GPIB address 3-6
Signal paths for “2-wireresistor” tests 2-8
Signal paths for “3terminal-npn-bjt” tests 2-7
Signal paths for “4terminal-n-fet” tests 2-7
Signal paths for “capacitor” test 2-8
Signal paths for “diode” tests 2-8
Signal paths for the pre and post stress tests 3-16
Signal paths to apply the pulse stress 3-17
Start and stop points for the line-fit 1-4
Subsite Plan window 2-11
SubVt slope 1-2
  Line-fit analysis 1-3
  Modify the line-fit 1-4
  Open “default” project 1-2
  Open “subvt” test and display graph 1-2
Supported external equipment table 3-4

Supported probers 3-22
System configuration for the “probesubsites” project 3-23
System configuration with external instruments 3-3

T

TDDB (time dependent dielectric breakdown 4-61
Test sequence 3-31
Test sequence for “ivpgswitch” project 3-19
Test system for “ivpgswitch” project 3-12
Tools menu access to Graph Settings 1-6
Typical CV curve 3-5

KEITHLEY

Service Form

Model No. _____ **Serial No.** _____ **Date** _____

Name and Telephone No. _____

Company _____

List all control settings, describe problem and check boxes that apply to problem. _____

_____

_____

❑ Intermittent
❑ Analog output follows display
❑ Particular range or function bad; specify _____
❑ IEEE failure
❑ Obvious problem on power-up
❑ Batteries and fuses are OK
❑ Front panel operational
❑ All ranges or functions are bad
❑ Checked all cables

Display or output (check one)

❑ Drifts
❑ Unable to zero
❑ Unstable
❑ Will not read applied input
❑ Overload

❑ Calibration only
❑ Certificate of calibration required
❑ Data required

(attach any additional sheets as necessary)

Show a block diagram of your measurement system including all instruments connected (whether power is turned on or not). Also, describe signal source.

Where is the measurement being performed? (factory, controlled laboratory, out-of-doors, etc.)

_____

_____

What power line voltage is used? _____ Ambient temperature?°F _____

Relative humidity? _____ Other? _____

Any additional information. (If special modifications have been made by the user, please describe.)

_____

_____

Be sure to include your name and phone number on this service form.

KEITHLEY

A GREATER MEASURE OF CONFIDENCE

Keithley Instruments, Inc.

Corporate Headquarters • 28775 Aurora Road • Cleveland, Ohio 44139 • 440-248-0400 • Fax: 440-248-6168 • 1-888-KEITHLEY • www.keithley.com

12/06