Using HP SPECS

HP 4070 Series Parametric Tester

August 1999

Edition 1

Legal Notice

The information contained in this document is subject to change without notice.

- **Product Warranty**

  Hewlett-Packard warrants Hewlett-Packard hardware, accessories and supplies against defects in materials and workmanship for the period of one year from the warranty start date specified below. If Hewlett-Packard receives notice of such defects during the warranty period, Hewlett-Packard will, at its option, either repair or replace products which prove to be defective. Replacement products may be either new or like-new.

  Warranty service of this product will be performed at Buyer's facility at no charge within Hewlett-Packard service travel areas. Outside Hewlett-Packard service travel areas, warranty service will be performed at Buyer's facility only upon HP's prior agreement and Buyer shall pay HP's round trip travel expenses. In all other cases, products must be returned to a service facility designated by Hewlett-Packard.

  For the products returned to Hewlett-Packard for warranty service, Buyer shall prepay shipping charges to Hewlett-Packard and Hewlett-Packard shall pay shipping charges to return the product to Buyer. However, Buyer shall pay all shipping charges, duties, and taxes for products returned to Hewlett-Packard from another country.

  Hewlett-Packard warrants that Hewlett-Packard software will not fail to execute its programming instructions, for the period of one year from the warranty start date specified below, due to defects in material and workmanship when properly installed and used. If Hewlett-Packard receives notice of such defects during the warranty period, Hewlett-Packard will replace software media which does not execute its programming instructions due to such defects.

  Hewlett-Packard does not warrant that the operation of Hewlett-Packard products will be uninterrupted or error free. If Hewlett-Packard is unable, within a reasonable time, to repair or replace any product to a condition as warranted, customer will be entitled to a refund of the purchase price upon prompt return of the product.

The Hewlett-Packard products may contain remanufactured parts equivalent to new in performance or may have been subject to incidental use.

The warranty period begins on the date of delivery or on the date of installation if installed by Hewlett-Packard. If customer schedules or delays Hewlett-Packard installation more than 30 days after delivery, warranty begins on the 31st day from delivery.

Warranty does not apply to defects resulting from (a) improper or inadequate maintenance or calibration, (b) software, interfacing, parts or supplies not supplied by Hewlett-Packard, (c) unauthorized modification or misuse, (d) operation outside of the published environmental specifications for the product, or (e) improper site preparation or maintenance.

To the extent allowed by local law, the above warranties are exclusive and no other warranty or condition, whether written or oral, is expressed or implied and Hewlett-Packard specifically disclaims any implied warranties or conditions of merchantability, satisfactory quality, and fitness for a particular purpose.

Hewlett-Packard will be liable for damage to tangible property per incident up to the greater of $300,000 or the actual amount paid for the product that is the subject of the claim, and for damages for bodily injury or death, to the extent that all such damages are determined by a court of competent jurisdiction to have been directly caused by a defective Hewlett-Packard product.

To the extent allowed by local law, the remedies in this warranty statement are customer's sole and exclusive remedies. Except as indicated above, in no event will Hewlett-Packard or its suppliers be liable for loss of data or for direct, special, incidental, consequential (including lost profit or data), or other damage, whether based in contract, tort, or otherwise.

For consumer transactions in Australia and New Zealand: the warranty terms contained in this statement, except to the extent lawfully permitted, do not exclude, restrict or modify and are in addition to the mandatory statutory rights applicable to the sale of this product to you.

Printing History

Edition 1: August 1999

In This Manual

The HP 4072A provides an algorithm library, framework, and test plan for use with the HP SPECS software. These are sample programs used to control the HP 4072A using HP SPECS.

This manual contains the following chapters:

1. Introduction

   Explains sample program files for flash memory test.

2. Installation

   Provides the installation instructions for the sample programs.

3. Executing Measurements

   Explains how to execute measurements using the sample programs.

4. Creating Your Test Plan

   Explains how to create the test plan used with the sample framework and algorithm library.

5. Sample Algorithm Library

   A description of the sample algorithm library.

6. Sample Framework

   A description of the sample framework.

7. Sample Test Plan

   A description of the sample test plan.

Contents

Contents

1 Introduction

The HP 4072A provides the following programs for use with the HP SPECS software. They are used to control the HP 4072A using HP SPECS.

- from_sample algorithm library
- FLASH_4072A framework
- FLASH_DEMO test plan

from_sample

This algorithm library contains the following five measurement algorithms. For details on the algorithms, see Chapter 5.

- Che_endurance algorithm

  Executes the write/erase times dependency test of a flash memory cell.

- Che_pwd_erase algorithm

  Executes the accumulated pulse width dependency test for the erase operation of a flash memory cell.

- Che_pwd_write algorithm

  Executes the accumulated pulse width dependency test for the write operation of a flash memory cell.

- Che_vthctrl algorithm

  Applies write/erase pulses to a flash memory cell, and measures the threshold voltage.

- Meas_vth algorithm

  Executes the threshold voltage measurement.

NOTE The write/erase operations of the algorithms are for a typical NOR-type flash memory cell. Refer to Chapter 5 to confirm the write/erase operation. If it is different from the write/erase operation of your test devices, you cannot execute the algorithms without modifying them.

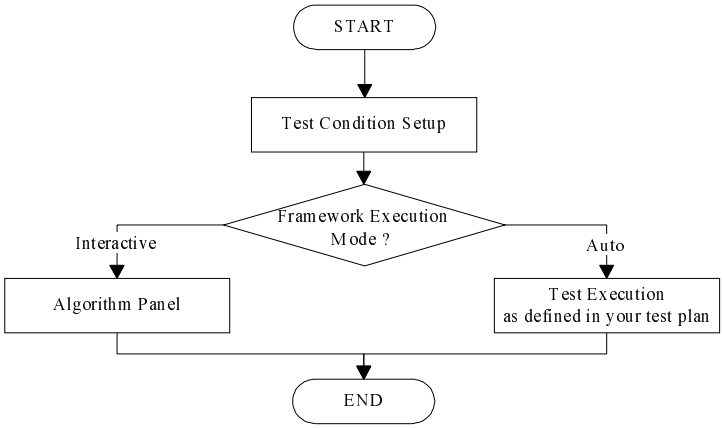
FLASH_4072A

This framework is created based on the HPSTD framework furnished with the HP SPECS software. The FLASH_4072A framework includes an additional operation panel used to select the framework execution mode, either Auto or Interactive.

In Auto mode, the framework executes the algorithms defined in your test plan. This operation is same as the HPSTD framework.

In Interactive mode, the framework calls the Algorithm Panel. You can execute the algorithms interactively using the Algorithm Panel.

For more details, see Chapter 3, “Executing Measurements”.



FLASH_DEMO

This test plan is an example of the FLASH_4072A framework Interactive execution mode. For Auto execution, you will need to create a test plan for the actual measurement. See Chapter 4, “Creating Your Test Plan”, for instructions on creating a test plan.

2 Installation

This procedure will install the sample programs provided. Before starting this procedure, confirm that the HP 4072A software and HP SPECS software are installed on your computer. If they have not been installed, complete the installation before continuing.

1. Install the sample framework as shown below. The FLASH_4072A framework will be installed in the directory */opt/SPECS/usr/fwk*.

   a. If you create the framework based on the HPSTD_NOPROBER framework:

      ```
      $ cd /opt/hp4070/demo/from_sample/fwk
      $ sample_installFWK
      ```

   b. If you create the framework based on the HPSTD_TELP8 framework:

      ```
      $ cd /opt/hp4070/demo/from_sample/fwk
      $ sample_installFWK HPSTD_TELP8
      ```

   c. If you create the framework based on the HPSTD_TSK90A framework:

      ```
      $ cd /opt/hp4070/demo/from_sample/fwk
      $ sample_installFWK HPSTD_TSK90A
      ```

   d. If you create the framework based on the HPSTD_EG4080 framework:

      ```
      $ cd /opt/hp4070/demo/from_sample/fwk
      $ sample_installFWK HPSTD_EG4080
      ```

2. To use the HP BASIC/UX algorithms, install the sample algorithm library as shown below. The from_sample algorithm library will be installed in the directory */opt/SPECS/usr/alg/measure*.

   ```
   $ cd /opt/hp4070/demo/from_sample/rmb
   $ sample_installB
   ```

3. To use the C language algorithms, install the sample algorithm library as shown below. The from_sample algorithm library will be installed in the directory */opt/SPECS/usr/calg/measure*.

   ```
   $ cd /opt/hp4070/demo/from_sample/C
   ```

   ```
   $ sample_installC
   ```

4. To use the sample test plan, copy the following file to your chosen directory. The following example copies the file to the directory */opt/SPECS/demo*.

   ```
   $ cd /opt/hp4070/demo/from_sample/tpl
   ```

   ```
   $ cp FLASH_DEMO.tpl /opt/SPECS/demo
   ```

3 Executing Measurements

This chapter explains how to execute a test using the FLASH_4072A framework.

NOTE

The FLASH_4072A framework is based on the HPSTD framework. For information on using the HPSTD framework, refer to the *"HP SPECS Framework Operation Guide"*.

To select and execute a test:

1. Use the following procedure to launch the FLASH_4072A framework.
   a. Open the HP SPECS NAVIGATOR window.
   b. Click Test Plan **Open**, and select a test plan, for example, FLASH_DEMO.
   c. Click Framework **Change**, and select FLASH_4072A.
   d. Click Algorithm **Change**, and select from_sample.
   e. Select **Execute:Execute**. The RUN PANEL and OPERATION PANEL are displayed on the screen. See Figure 3-1.
   f. Click **Start** on the RUN PANEL to launch the framework. The Initialize Hardware panel is displayed, as shown in Figure 3-1.

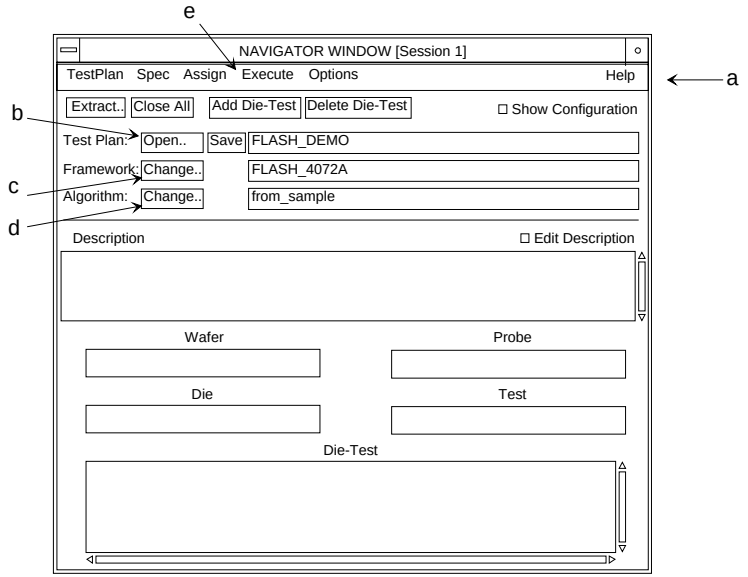
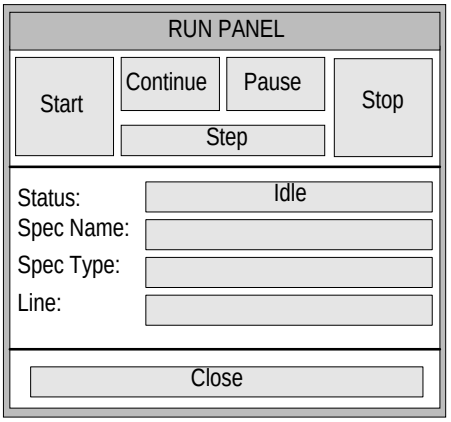
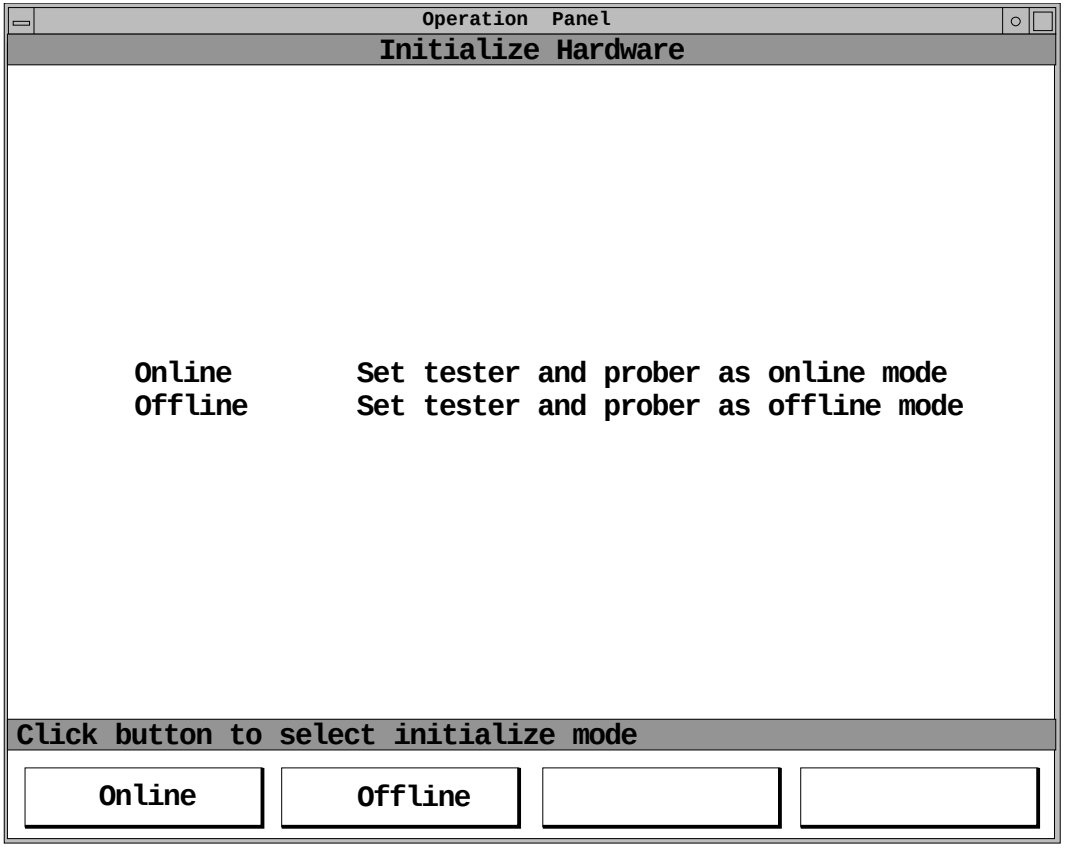


Figure 3-1 Launching the Framework





2. **Initialize Hardware panel**

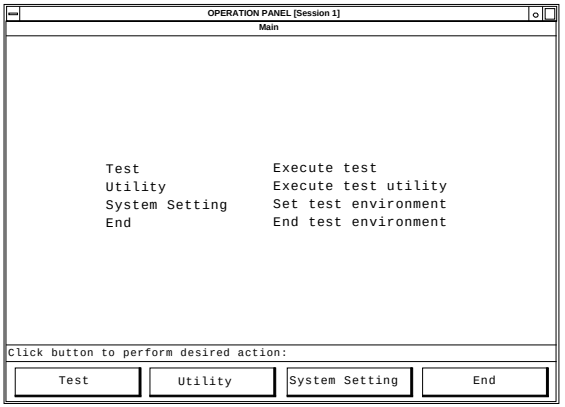
   To select the online execution mode, click **Online**. This executes the hardware initialization.

3. **Hardware Configuration panel**

   After the initialization finishes, click **Next.**

4. **Main panel**

   To continue the test setup, click **Test**.



5. **Register Operator panel**

   Enter your operator name in the Operator Name entry field (optional), and then click **Next**.

6. **Set Test Condition panel**

   Enter the lot ID, cassette ID, and process name in the entry fields (optional).

   Select the Limit File (optional).

   To continue, click **Next**.

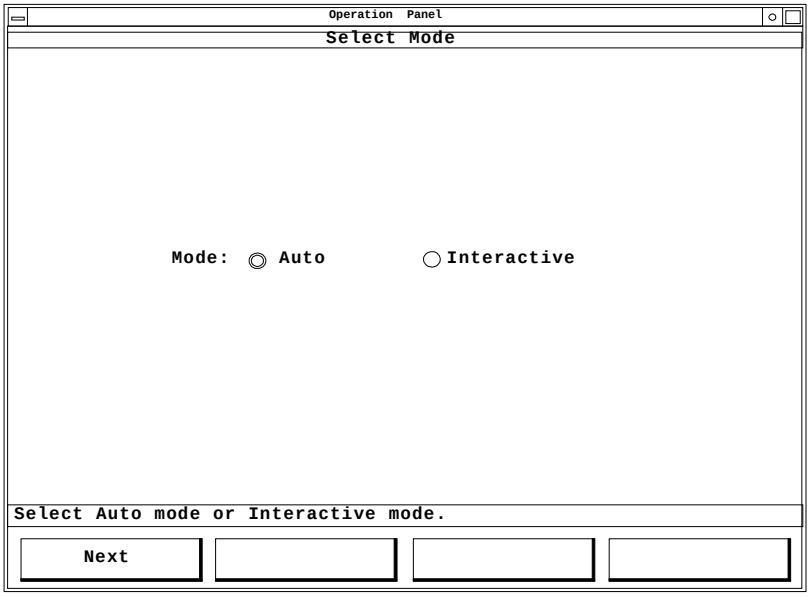
   To return to the Main panel, click **Main Menu**.

7. **Select Wafer panel**

   Select the wafer ID (optional), and click **Start Test**.

   To return to the Set Test Condition panel, click **Abort Test**.

8. **Select Mode panel**



This operation panel selects the framework execution mode, either Auto or Interactive. Select the mode and click **Next**.

Auto The framework automatically executes the algorithms defined in your test plan. This operation is the same as the HPSTD framework.

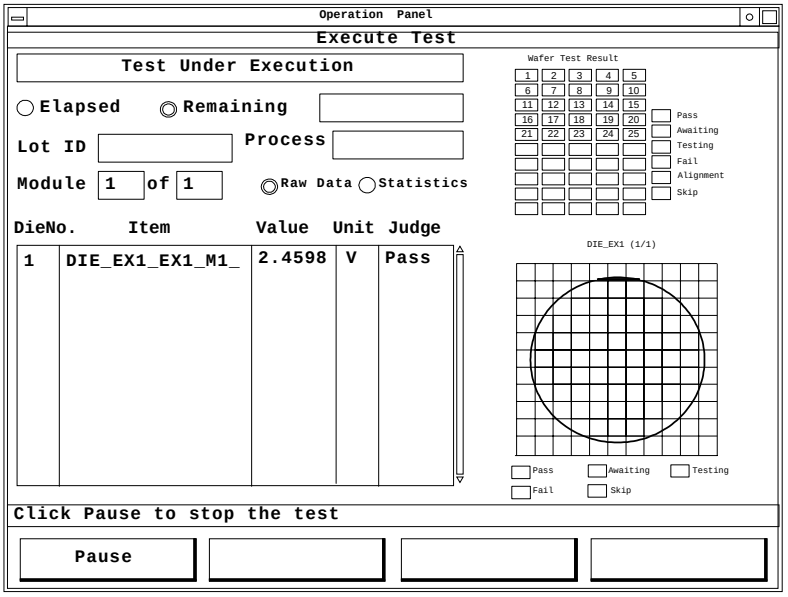
Interactive The framework calls the Algorithm Panel. You can select an algorithm and execute it using the Algorithm Panel.

9. **Select Die/Module panel**

Select the dice and modules to be tested, and click **Next** to start the test.

To apply the default settings to the panel, click **Undo**.

10. **(in Auto mode) Execute Test panel**



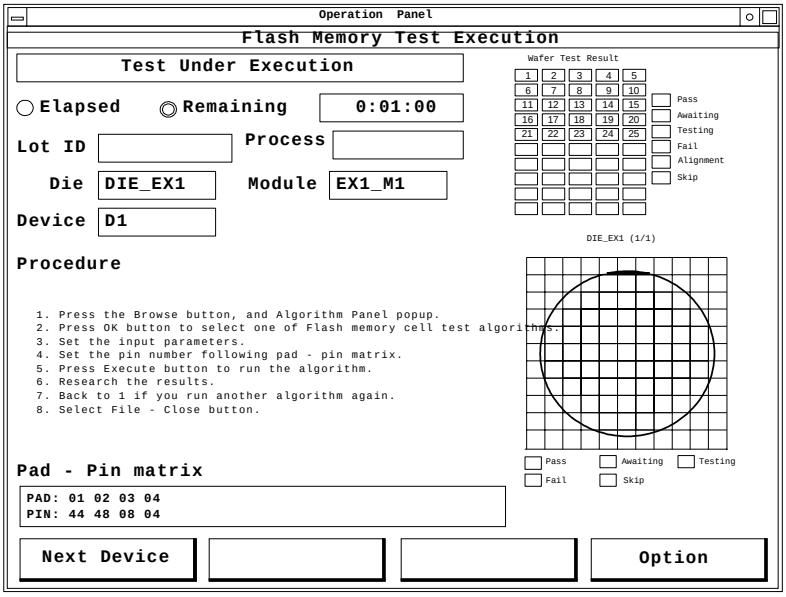
On the Select Mode panel, select Auto mode to display this panel during test execution.

To stop the test, click **Pause**. The test is stopped, and the panel offers the choices **Continue** and **Option**.

To continue the test, click **Continue**.

To reject the test or retest, click **Option**, and click **Retest/Reject Test** on the Option Menu panel. The Retest/Reject Test panel is displayed.

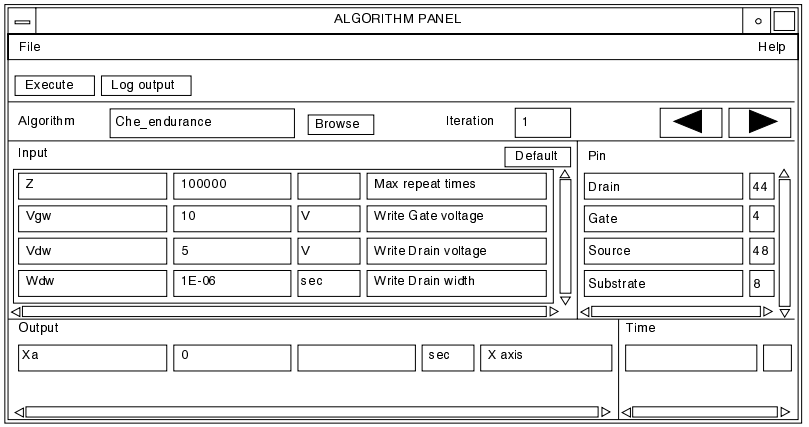
11. **(in Interactive mode) Flash Memory Test Execution panel**



On the Select Mode panel, select Interactive mode to display this panel along with the Algorithm Panel.

To execute an algorithm using the Algorithm Panel:

a. Click **Browse**. A popup dialog box is displayed.



b. Select an algorithm from the list, for example, Che_endurance, and then click **OK**.

c. Set the algorithm input parameter values on the Algorithm Panel. In the Pin area, set the pin number listed in the Pad - Pin matrix section of the operation panel. For example, set 4 for Gate, 8 for Substrate, 44 for Drain, and 48 for Source.

d. To execute the algorithm, click **Execute**.

e. To execute another algorithm, repeat steps a through d.

f. To close the Algorithm Panel, select **File:Close**.

To test the next device, and repeat steps a through f, click **Next Device**.

To reject the test or to retest, click **Option**. The Retest/Reject Test panel will be displayed.

12. **Test Result panel**

This panel is displayed after the test, and shows the test result list.

To return to the Set Test Condition panel, click **Complete Test**.

To run the test again, select a wafer from the list, and select **Start Retest**.

NOTE

After the test, the Lot Statistics panel is displayed. This panel shows the statistical data. To close the panel, enter **q**.

In Interactive mode, the Lot Statistics panel shows only that the test was executed. The test result data is not recorded in the statistical data file.

4 Creating Your Test Plan

This chapter explains how to create a test plan for use with the FLASH_4072 framework and the from_sample algorithm library.

NOTE

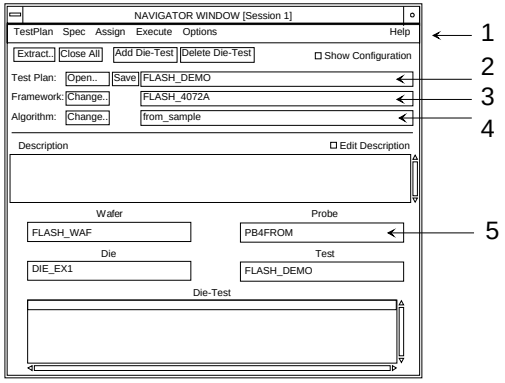
FLASH_4072A Framework Execution Mode

The FLASH_4072 framework provides two execution modes, Interactive and Auto.

In Auto mode, the framework executes the algorithms defined in your test plan. This operation is the same as the HPSTD framework. To execute a specific test, modify the FLASH_DEMO test specification in the sample test plan, or create a new test specification.

In Interactive mode, the framework calls the Algorithm Panel. You can use the Algorithm Panel to select an algorithm and execute the test while the framework is running. Use the FLASH_DEMO test specification without modification.

You can create a test plan by modifying the FLASH_DEMO sample test plan as shown below:



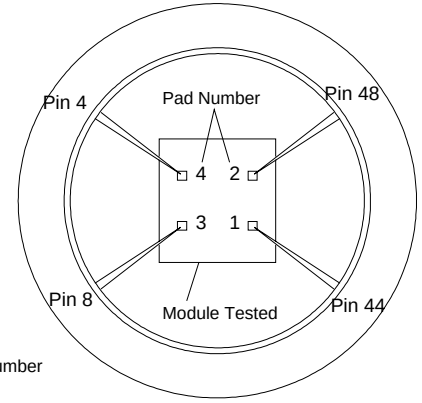
1. Open the HP SPECS NAVIGATOR window.
2. Click Test Plan **Open**, and select FLASH_DEMO.
3. Click Framework **Change**, and select FLASH_4072A.
4. Click Algorithm **Change**, and select from_sample.
5. **(Probe Spec)** Edit the probe specification to define the relationship between the pad number of your test die or module and the matrix pin number of your tester.

   To open the Spec Editor, move the mouse pointer to the Probe field of the NAVIGATOR window, click the right mouse button, and select **Edit**.

Example of Probe Specification

Spec Editor

Pad Number	Pin Number
1	44
2	48
3	8
4	4



Pin N: Matrix Pin number

6. **(Wafer Spec)** Use the Spec Editor Attribute menu to edit the attribute information for the wafer, such as wafer size or die size.

   Use the Spec Editor Body menu to define the dice to be tested, such as Die Name: MY_DIE.

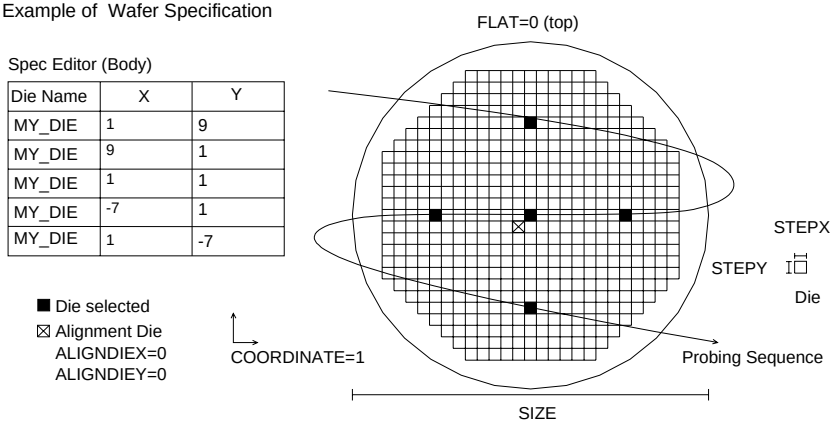
   To open the Spec Editor, move the mouse pointer to the Wafer field on the NAVIGATOR window, click the right mouse button, and select **Edit**.

   To select the Body or Attribute menu, click **Body** or **Attribute** on the Spec Editor.

Example of Wafer Specification

Spec Editor (Body)

Die Name	X	Y
MY_DIE	1	9
MY_DIE	9	1
MY_DIE	1	1
MY_DIE	-7	1
MY_DIE	1	-7



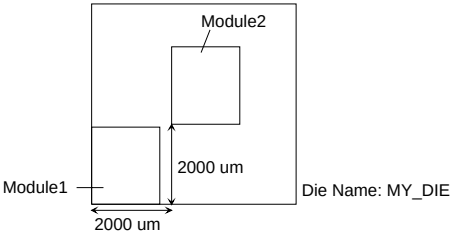
7. **(Die Spec)** Enter the die name, such as MY_DIE, in the Die entry field of the NAVIGATOR window, and press return to open the Spec Editor. Define the new specification.

   Use the same die name defined earlier in the wafer specification.

Example of Die Specification

Spec Editor

Module Name	X[um]	Y[um]
Module1	0	0
Module2	2000	2000



8. **(Test Spec)** If you execute the test in FLASH_4072A framework Interactive mode, skip this step. If you execute the test in Auto mode, do the following:

Enter the test name, such as TEST1, in the Test entry field of the NAVIGATOR window. Press return to open the Spec Editor. Define the new specification as shown below.

Example of Test Specification:

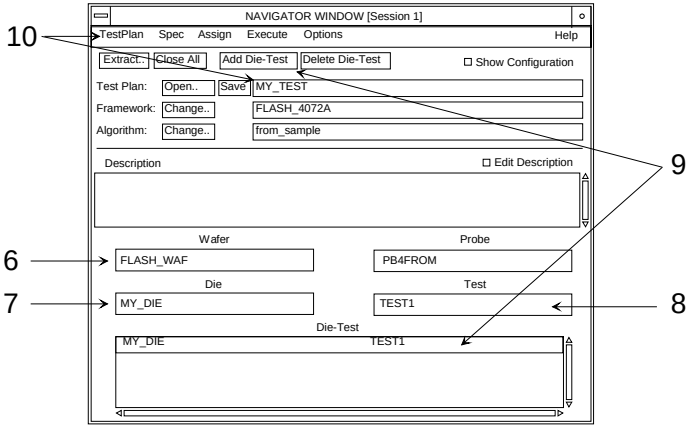
Module Name	Device Name	Algorithm	Input	Pad Number	Out
Module1	Device1	IF	AMODE > 0		
		Che_endurance	Z=10000,Vgw=8,Vdw	D=1,B=2,S=3,	Xa
		ELSE			
		ACTION	“DIRECT_EXEC”		
		END IF			

In this example, only Che_endurance is defined as a measurement algorithm. To add more algorithms, insert them between the IF line and the ELSE line. Do not remove the IF, ELSE, ACTION, or END IF lines. These lines must be defined in the test plan used with the FLASH_4072A framework.

The Module Name is important. Enter the same Module Name, such as Module1, defined in the die specification.

Use the Assist dialog box to enter the algorithm input parameters. To open the Assist dialog box, move the mouse pointer to the Input field, click the right mouse button, and select **Assist**.

9. Click **Add Die-Test**. This registers the die and test specifications pair for the test.
10. Save the test plan with a new name, such as MY_TEST.



5 Sample Algorithm Library

The HP 4072A provides a sample algorithm library for the flash memory cell test. The algorithm library contains these algorithms:

Algorithm Name	Summary
“Che_endurance”	Executes the write/erase times dependency test (endurance test).
“Che_pwd_erase”	Executes the accumulated pulse width dependency test for the erase operation.
“Che_pwd_write”	Executes the accumulated pulse width dependency test for the write operation.
“Che_vthctrl”	Applies write/erase pulses to the memory cell, and measures the threshold voltage.
“Meas_vth”	Measures the threshold voltage.

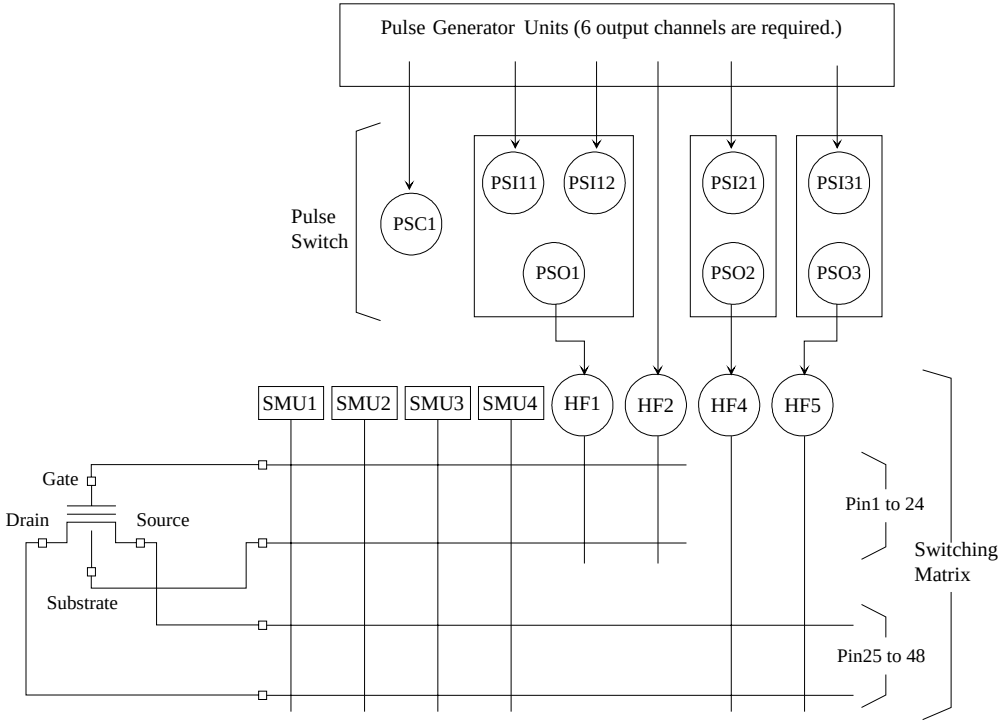
NOTE

The write/erase operations are for a typical NOR type flash memory cell. Refer to “Write/Erase Operation of Algorithms” on page 5-4, and confirm these operations. If they are different from the write/erase operations of your test devices, you will need to modify the algorithms to execute the tests.

Tester Interconnection for Algorithms

These algorithms use six pulse generator (PGU) units and four source monitor (SMU) units. The tester interconnections are shown below. Before executing the algorithms, connect the cables and set the HP 4072A configuration properly.

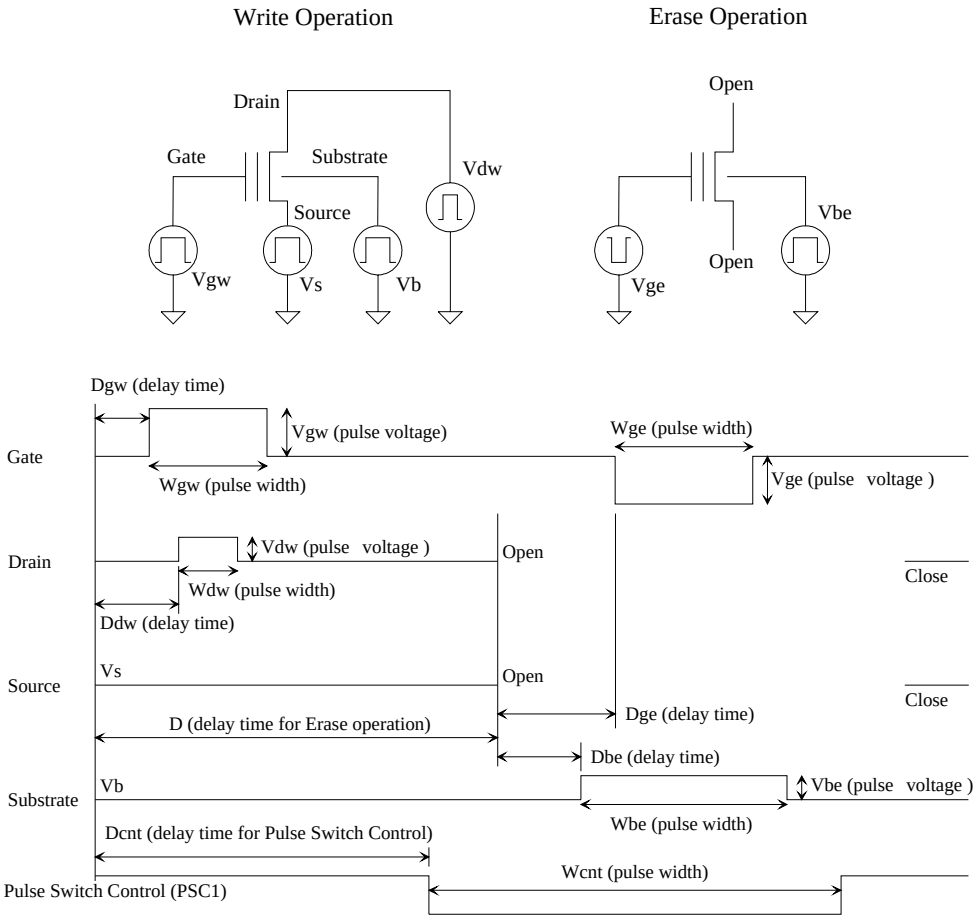
- a pulse generator unit to PSC1
- a pulse generator unit to PSI11, and PSO1 to HF1
- a pulse generator unit to PSI12
- a pulse generator unit to HF2
- a pulse generator unit to PSI21, and PSO2 to HF4
- a pulse generator unit to PSI31, and PSO3 to HF5



Gates and Substrates should be connected to pins 1 through 24 of the switching matrix. Drains and Sources should be connected to pins 25 through 48.

Write/Erase Operation of Algorithms

The write/erase operations are shown below. The pulse generator units are used to force the write/erase pulses, and control the pulse switches used to open the drain and source terminals in the erase operation. To perform a write/erase operation, you must first define the algorithm input parameters, such as the pulse voltage, pulse width, delay time, and pulse edge (pulse leading time and trailing time).



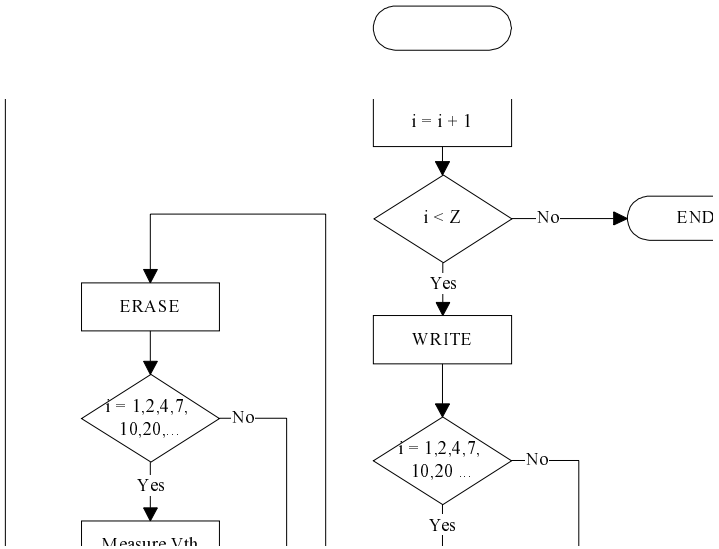
where D=Dcnt+switching time of the pulse switches, Dcnt=Dgw+Wgw+overhead, and Wcnt=Dbe+Wbe+switching time+overhead. These values are important in the endurance test, and calculated in the Che_endurance algorithm automatically.

Che_endurance

This algorithm executes the endurance test for a flash memory cell.

Execution Flow

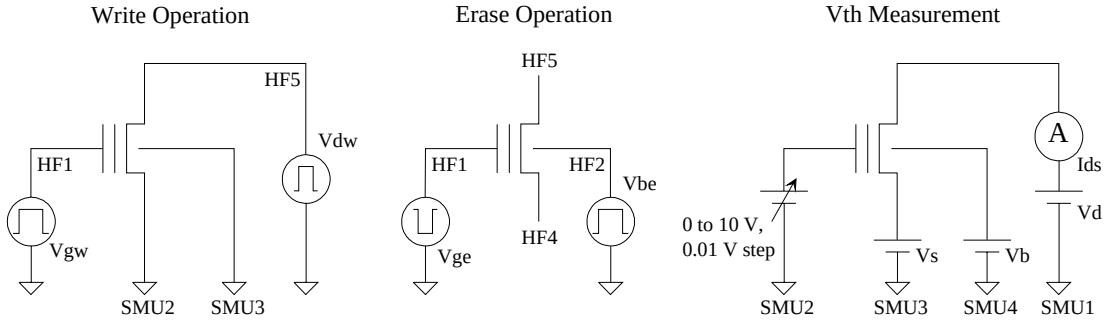
The algorithm repeats the write operation, erase operation and threshold voltage (Vth) measurement. Vth measurement is made after the 1, 2, 4, 7 steps of the pulse count. This means that the algorithm executes the Vth measurement after the 1st pulse, 2nd pulse, 4th pulse, 7th pulse, 10th pulse, 20th pulse, and so on. The maximum pulse count is defined by the algorithm input parameter Z.



Electrical Characteristics

If you set the algorithm input parameter Am to 0, this algorithm displays the test result graph (write/erase counts vs Vth) after the test.

Test Configuration This algorithm connects the device as shown below.



Input Parameters This algorithm requires the following input parameters. You will also need to define the switching matrix pin numbers in the test plan (test specification).

Name	Type	Description	Default Value
Z	REAL	Maximum count of write/erase operation.	100000
Vgw	REAL	Gate pulse voltage [V] for write.	10
Vdw	REAL	Drain pulse voltage [V] for write.	5
Wdw	REAL	Drain pulse width [sec] for write.	1 u
Vge	REAL	Gate pulse voltage [V] for erase.	-10
Vbe	REAL	Substrate pulse voltage [V] for erase.	5
Wge	REAL	Gate pulse width [sec] for erase.	20 m
Vd	REAL	Drain voltage [V] for Vth search.	1.0
Vs	REAL	Source voltage [V] for Vth search.	0
Vb	REAL	Substrate voltage [V] for Vth search.	0
Ids	REAL	Target drain current [A] for Vth search.	1 u
Am	INTEGER	Displays X-Y graph after test. 0: yes or 1: no.	0

Output Parameters This algorithm returns the following values.

Name	Type	Description
Xa	VAR1[100]	X axis data (write/erase counts).
Vthw	REAL[100]	Y axis data (write Vth measured [V]).
Vthe	REAL[100]	Y axis data (erase Vth measured [V]).

PGU Setups

The following PGU setup parameters are defined in the program. If you change the values, edit the program.

Name	Description	Defined Value
Rp_vgw	Gate pulse voltage for write	Vgw
Rp_vdw	Drain pulse voltage for write	Vdw
Rp_vsw	Source voltage for write	0
Rp_vbw	Substrate voltage for write	0
Rp_wgw	Gate pulse width for write	Wdw*2.0 [sec]
Rp_wdw	Drain pulse width for write	Wdw
Rp_dgw	Gate pulse delay time for write	0
Rp_ddw	Drain pulse delay time for write	Wdw*0.5 [sec]
Rp_vge	Gate pulse voltage for erase	Vge
Rp_vbe	Substrate pulse voltage for erase	Vbe
Rp_wge	Gate pulse width for erase	Wge
Rp_wbe	Substrate pulse width for erase	Wge*1.05 [sec]
Rp_dge	Gate pulse delay time for erase	Wge*0.025 [sec]
Rp_dbe	Substrate pulse delay time for erase	0
Rp_per	Pulse period	0
Rp_ew	Pulse edge (leading/trailing time) for write	2.0E-8 [sec]
Rp_ee	Pulse edge (leading/trailing time) for erase	2.0E-8 [sec]

SMU Setups

The following SMU setup parameters are defined in the program. If you change the values, edit the program.

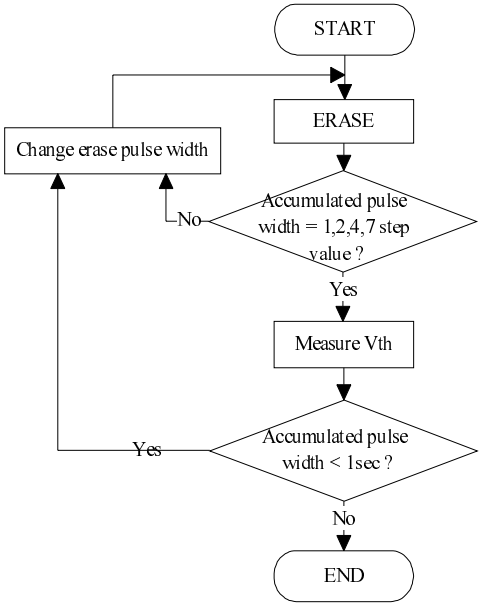
Name	Description	Defined Value
Vth_vgmin	Gate start voltage	0
Vth_vgmax	Gate stop voltage	10.0 [V]
Vth_vgstep	Gate step voltage	0.01 [V]
Vth_vd	Drain voltage	Vd
Vth_vs	Source voltage	Vs
Vth_vb	Substrate voltage	Vb
Vth_ids	Target drain current for Vth search	Ids
Vth_hold	Hold time before search	0
Vth_delay	Delay time for each search point	0
Vth_idrng	Drain current measurement range	0
Vth_idc	Drain current compliance	1.0E-2 [A]
Vth_igc	Gate current compliance	1.0E-3 [A]
Vth_iv	integ value of Set_adc TIS.	1.0
Vth_ss	next force wait of Set_wait_time TIS.	1.0
Vth_sm	measurement wait of Set_wait_time TIS.	1.0
Vth_integ	integ mode of Set_adc TIS.	0
Vth_flt	filter mode of Set_smu_ch TIS.	0
Vth_adc	adc of Set_adc and Set_smu_ch TIS.	0

Che_pwd_erase

This algorithm executes the accumulated pulse width dependency test for the erase operation of a flash memory cell.

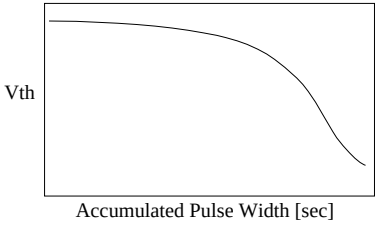
Execution Flow

The algorithm forces the erase pulse to the cell, measures the threshold voltage Vth, and repeats. The pulse width is automatically changed so that the accumulated pulse width will be 100 nsec, 200 nsec, 400 nsec, 700 nsec, 1 usec, 2 usec, and so on. The maximum accumulated pulse width is 1 sec.



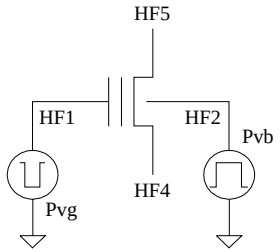
Electrical Characteristics

If you set the algorithm input parameter Am to 0, this algorithm displays the test result graph (accumulated erase pulse width vs Vth) after the test.

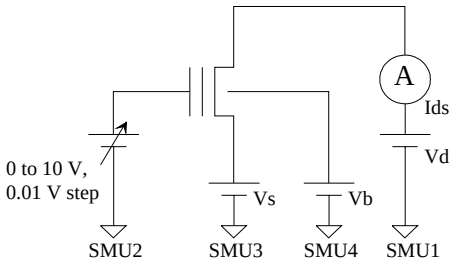


Test Configuration This algorithm connects the device as shown below.

Erase Operation



Vth Measurement



Input Parameters This algorithm needs the following input parameters. You also need to define the switching matrix pin numbers in the test plan (test specification).

Name	Type	Description	Default Value
Pvg	REAL	Gate pulse voltage [V] for erase.	10
Pvb	REAL	Substrate pulse voltage [V] for erase.	5
Vd	REAL	Drain voltage [V] for Vth search.	1.0
Vs	REAL	Source voltage [V] for Vth search.	0
Vb	REAL	Substrate voltage [V] for Vth search.	0
Ids	REAL	Target drain current [A] for Vth search.	1 u
Lvth	REAL	Limit value of Vth [V]. The result graph draws the limit line. -100 disables this function.	-100
Am	INTEGER	Displays X-Y graph after test. 0: yes or 1: no.	0

Output Parameters This algorithm returns the following values.

Name	Type	Description
Xa	VAR1[100]	X axis data (accumulated erase pulse width [sec]).
Vth	REAL[100]	Y axis data (Vth measured [V]).

PGU Setups

The following PGU setup parameters are defined in the program. If you change the values, edit the program. Timing parameters are automatically calculated in the algorithm.

Name	Description	Defined Value
Ep_pvg	Gate pulse voltage for erase	Pvg
Ep_pvb	Substrate pulse voltage for erase	Pvb
Ep_eg	Gate pulse edge (leading/trailing time)	2.0E-8 [sec]
Ep_eb	Substrate pulse edge (leading/trailing time)	2.0E-8 [sec]

SMU Setups

The following SMU setup parameters are defined in the program. If you change the values, edit the program.

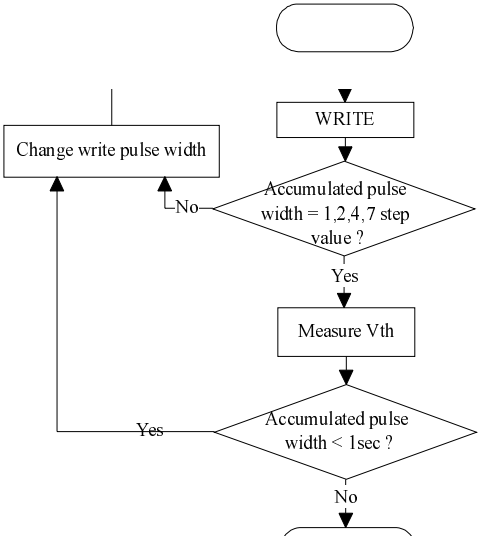
Name	Description	Defined Value
Vth_vgmin	Gate start voltage	0
Vth_vgmax	Gate stop voltage	10.0 [V]
Vth_vgstep	Gate step voltage	0.01 [V]
Vth_vd	Drain voltage	Vd
Vth_vs	Source voltage	Vs
Vth_vb	Substrate voltage	Vb
Vth_ids	Target drain current for Vth search	Ids
Vth_hold	Hold time before search	0
Vth_delay	Delay time for each search point	0
Vth_idrng	Drain current measurement range	0
Vth_idc	Drain current compliance	1.0E-2 [A]
Vth_igc	Gate current compliance	1.0E-3 [A]
Vth_iv	integ value of Set_adc TIS.	1.0
Vth_ss	next force wait of Set_wait_time TIS.	1.0
Vth_sm	measurement wait of Set_wait_time TIS.	1.0
Vth_integ	integ mode of Set_adc TIS.	0
Vth_flt	filter mode of Set_smu_ch TIS.	0
Vth_adc	adc of Set_adc and Set_smu_ch TIS.	0

Che_pwd_write

This algorithm executes the accumulated pulse width dependency test for the write operation of a flash memory cell.

Execution Flow

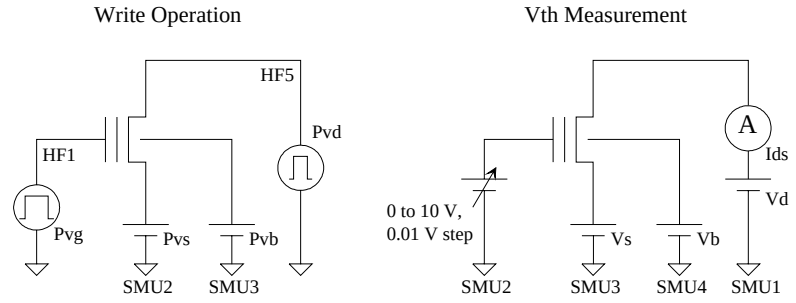
The algorithm forces the write pulse to the cell, measures the threshold voltage Vth, and repeats. The pulse width is automatically changed so that the accumulated pulse width will be 100 nsec, 200 nsec, 400 nsec, 700 nsec, 1 usec, 2 usec, and so on. The maximum accumulated pulse width is 1 sec.



Electrical Characteristics

If you set the algorithm input parameter Am to 0, this algorithm displays the test result graph (accumulated write pulse width vs Vth) after the test.

Test Configuration This algorithm connects the device as shown below.



Input Parameters This algorithm needs the following input parameters. You also need to define the switching matrix pin numbers in the test plan (test specification).

Name	Type	Description	Default Value
Pvg	REAL	Gate pulse voltage [V] for write.	10
Pvd	REAL	Drain pulse voltage [V] for write.	5
Pvs	REAL	Source voltage [V] for write.	0
Pvb	REAL	Substrate voltage [V] for write.	0
Vd	REAL	Drain voltage [V] for Vth search.	1.0
Vs	REAL	Source voltage [V] for Vth search.	0
Vb	REAL	Substrate voltage [V] for Vth search.	0
Ids	REAL	Target drain current [A] for Vth search.	1 u
Lvth	REAL	Limit value of Vth [V]. The result graph draws the limit line. -100 disables this function.	-100
Am	INTEGER	Displays X-Y graph after test. 0: yes or 1: no.	0

Output Parameters This algorithm returns the following values.

Name	Type	Description
Xa	VAR1[100]	X axis data (accumulated write pulse width [sec]).
Vth	REAL[100]	Y axis data (Vth measured [V]).

PGU Setups

The following PGU setup parameters are defined in the program. If you change the values, edit the program. Timing parameters are automatically calculated in the algorithm.

Name	Description	Defined Value
Wp_pvg	Gate pulse voltage for write	Pvg
Wp_pvd	Drain pulse voltage for write	Pvd
Wp_vs	Source voltage for write	Pvs
Wp_vb	Substrate voltage for write	Pvb
Wp_eg	Gate pulse edge (leading/trailing time)	2.0E-8 [sec]
Wp_ed	Drain pulse edge (leading/trailing time)	2.0E-8 [sec]

SMU Setups

The following SMU setup parameters are defined in the program. If you change the values, edit the program.

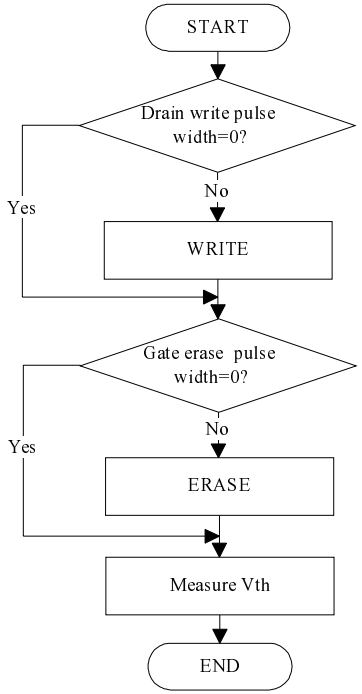
Name	Description	Defined Value
Vth_vgmin	Gate start voltage	0
Vth_vgmax	Gate stop voltage	10.0 [V]
Vth_vgstep	Gate step voltage	0.01 [V]
Vth_vd	Drain voltage	Vd
Vth_vs	Source voltage	Vs
Vth_vb	Substrate voltage	Vb
Vth_ids	Target drain current for Vth search	Ids
Vth_hold	Hold time before search	0
Vth_delay	Delay time for each search point	0
Vth_idrng	Drain current measurement range	0
Vth_idc	Drain current compliance	1.0E-2 [A]
Vth_igc	Gate current compliance	1.0E-3 [A]
Vth_iv	integ value of Set_adc TIS.	1.0
Vth_ss	next force wait of Set_wait_time TIS.	1.0
Vth_sm	measurement wait of Set_wait_time TIS.	1.0
Vth_integ	integ mode of Set_adc TIS.	0
Vth_flt	filter mode of Set_smu_ch TIS.	0
Vth_adc	adc of Set_adc and Set_smu_ch TIS.	0

Che_vthctrl

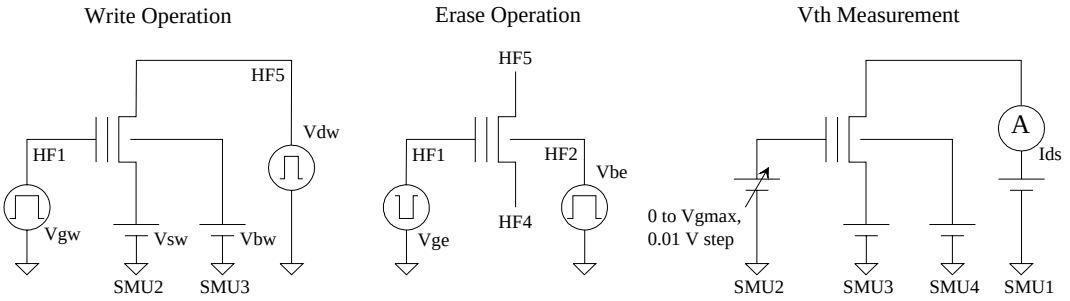
This algorithm forces write and erase pulses to a flash memory cell, and measures the threshold voltage Vth.

Execution Flow

This algorithm forces the specified counts of write pulses and the specified counts of erase pulses to the memory cell, then measures Vth. If you set the drain pulse width for write to 0 sec, the algorithm does not force the write pulses. If you set the gate pulse width for erase to 0 sec, the algorithm does not force the erase pulses.



Test Configuration This algorithm connects the device as shown below.



Input Parameters This algorithm needs the following input parameters. You also need to define the switching matrix pin numbers in the test plan (test specification).

Name	Type	Description	Default Value
Vgw	REAL	Gate pulse voltage [V] for write.	10
Vdw	REAL	Drain pulse voltage [V] for write.	5
Vsw	REAL	Source voltage [V] for write.	0
Vbw	REAL	Substrate voltage [V] for write.	0
Wdw	REAL	Drain pulse width [sec] for write.	1 u
Ew	REAL	Pulse edge [sec] for all write pulses.	20 n
Pctw	REAL	Pulse counts for all write pulses.	1
Vge	REAL	Gate pulse voltage [V] for erase.	-10
Vbe	REAL	Substrate pulse voltage [V] for erase.	5
Wge	REAL	Gate pulse width [sec] for erase.	20 m
Ee	REAL	Pulse edge [sec] for all erase pulses.	20 n
Pcte	REAL	Pulse counts for all erase pulses.	1
Vgmax	REAL	Maximum gate voltage [V] for Vth search.	10
Ids	REAL	Target drain current [A] for Vth search.	1 u

Output Parameters This algorithm returns the following values.

Name	Type	Description
Vth	REAL	Vth measured data [V].
Pst	INTEGER	Measurement status.

PGU Setups

The following PGU setup parameters are defined in the program. If you change the values, edit the program.

Name	Description	Defined Value
Wp_pvg	Gate pulse voltage for write	Vgw
Wp_pvd	Drain pulse voltage for write	Vdw
Wp_vs	Source voltage for write	Vsw
Wp_vb	Substrate voltage for write	Vbw
Wp_per	Pulse period for write	0
Wp_pct	Pulse counts for write	Pctw
Wp_eg	Gate pulse edge (leading/trailing time) for write	Ew
Wp_ed	Drain pulse edge (leading/trailing time) for write	Ew
Wp_wd	Drain pulse width for write	Wdw
Wp_dg	Gate pulse delay time for write	0
Ep_pvg	Gate pulse voltage for erase	Vge
Ep_pvb	Substrate pulse voltage for erase	Vbe
Ep_per	Pulse period for erase	0
Ep_pct	Pulse counts for erase	Pcte
Ep_eg	Gate pulse edge (leading/trailing time) for erase	Ee
Ep_eb	Substrate pulse edge (leading/trailing time) for erase	Ee
Ep_wg	Gate pulse width for erase	Wge
Ep_db	Substrate pulse delay time for erase	0

Gate pulse width and Drain pulse delay time for write are calculated in the algorithm by using the Wp_wd value. And Substrate pulse width and Gate pulse delay time for erase are calculated in the algorithm by using the Ep_wg value.

SMU Setups

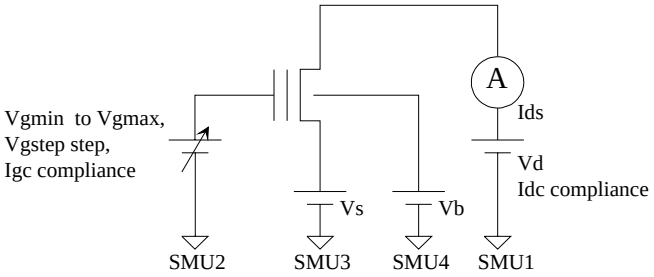
The following SMU setup parameters are defined in the program. If you change the values, edit the program.

Name	Description	Defined Value
Vth_vgmin	Gate start voltage	0
Vth_vgmax	Gate stop voltage	Vgmax
Vth_vgstep	Gate step voltage	0.01 [V]
Vth_vd	Drain voltage	1.0 [V]
Vth_vs	Source voltage	0
Vth_vb	Substrate voltage	0
Vth_ids	Target drain current for Vth search	Ids
Vth_hold	Hold time before search	0
Vth_delay	Delay time for each search point	0
Vth_idrng	Drain current measurement range	0
Vth_idc	Drain current compliance	1.0E-2 [A]
Vth_igc	Gate current compliance	1.0E-3 [A]
Vth_iv	integ value of Set_adc TIS.	1.0
Vth_ss	next force wait of Set_wait_time TIS.	1.0
Vth_sm	measurement wait of Set_wait_time TIS.	1.0
Vth_integ	integ mode of Set_adc TIS.	0
Vth_flt	filter mode of Set_smu_ch TIS.	0
Vth_adc	adc of Set_adc and Set_smu_ch TIS.	0

Meas_vth

This algorithm measures the threshold voltage Vth.

Test Configuration This algorithm connects the device as shown below.



Input Parameters This algorithm needs the following input parameters. You also need to define the switching matrix pin numbers in the test plan (test specification).

Name	Type	Description	Default Value
Vgmin	REAL	Gate start voltage [V].	0.0
Vgmax	REAL	Gate stop voltage [V].	10
Vgstep	REAL	Gate step voltage [V].	0.01
Vd	REAL	Drain voltage [V].	1.0
Vs	REAL	Source voltage [V].	0.0
Vb	REAL	Substrate voltage [V].	0.0
Ids	REAL	Target drain current [A].	1 u
Hold	REAL	Wait time [sec] until starting Vth search.	0
Delay	REAL	Delay time [sec] in every gate voltage steps.	0
Idrng	REAL	Drain current measurement range [A]. For auto ranging, set Idrng=0.	0

Name	Type	Description	Default Value
Idc	REAL	Drain current compliance [A].	10 m
Igc	REAL	Gate current compliance [A].	1 m
Iv	REAL	integ value of Set_adc TIS.	1
Ss	REAL	next force wait of Set_wait_time TIS.	1
Sm	REAL	measurement wait of Set_wait_time TIS.	1
Integ	INTEGER	integ mode of Set_adc TIS. 0:manual, 1:short, 2:medium, or 3:long.	0
Flt	INTEGER	filter mode of Set_smu_ch TIS. 0:filter off, or 1:filter on.	0
Adc	INTEGER	adc of Set_adc and Set_smu_ch TIS. 0:high speed, or 1:high resolution.	0

For Iv, Ss, Sm, Integ, Flt, and Adc parameters, refer to Set_adc, Set_smu_ch and Set_wait_time TIS as described in the *HP 4070 Programming Reference*.

Output Parameters This algorithm returns the following values.

Name	Type	Description
Vth	REAL	Vth measured data [V].
Pst	INTEGER	Meaurement status.

6 Sample Framework

The FLASH_4072A framework is based on the HPSTD framework, and includes some additional and modified specifications.

This chapter describes the additional and modified specifications.

- "Action Specifications"
- "Panel Specifications"

NOTE For information on using the HPSTD framework, refer to the *"HP SPECS Framework Operation Guide"*.

Action Specifications

The following action specifications have been modified or added to the FLASH_4072A framework.

- DIRECT_EXEC

  Displays the Flash Memory Test Execution panel and calls the Algorithm Panel. Data in the Pad - Pin matrix field of the Flash Memory Test Execution panel is also calculated. This action specification must be called from the test plan.

- DISP_FMCT_RUN_CONT

  Action for the Next Device button on the Flash Memory Test Execution panel. Framework redraws the Flash Memory Test Execution panel, and calls the Algorithm Panel.

- DISP_FMCT_RUN_OPTION

  Action for the Option button on the Flash Memory Test Execution panel. Framework displays the Retest/Reject Test panel.

- FMC_SEL_TESTSEQ_PREV

  Action for the Prev button on Retest/Reject Test panel. Returns to the Flash Memory Test Execution panel.

- FMC_SEL_TESTSEQ_REJECTDEV

  Action for the Reject Device menu on the Retest/Reject Test panel. Rejects the present device test.

- FMC_SEL_TESTSEQ_REJECTDIE

  Action for the Reject Die menu on the Retest/Reject Test panel. Rejects the present die test.

- FMC_SEL_TESTSEQ_REJECTLOT

  Action for the Reject Lot menu on the Retest/Reject Test panel. Rejects the present lot test.

- FMC_SEL_TESTSEQ_REJECTMOD

  Action for the Reject Module menu on the Retest/Reject Test panel. Rejects the present module test.

- FMC_SEL_TESTSEQ_REJECTTST

  Action for the Reject Test menu on the Retest/Reject Test panel. Rejects the present test.

- FMC_SEL_TESTSEQ_REJECTWAF

  Action for the Reject Wafer menu on the Retest/Reject Test panel. Rejects the present wafer test.

- FMC_SEL_TESTSEQ_RETESTDEV

  Action for the Retest Device menu on the Retest/Reject Test panel. Repeats the present device test.

- FMC_SEL_TESTSEQ_RETESTDIE

  Action for the Retest Die menu on the Retest/Reject Test panel. Repeats the present die test.

- FMC_SEL_TESTSEQ_RETESTMOD

  Action for the Retest Module menu on the Retest/Reject Test panel. Repeats the present module test.

- FMC_SEL_TESTSEQ_RETESTTST

  Action for the Retest Test menu on the Retest/Reject Test panel. Repeats the present test.

- FMC_SEL_TESTSEQ_RETESTWAF

  Action for the Retest Wafer menu on the Retest/Reject Test panel. Repeats the present wafer test.

- PLG_LOOP_BEGIN

  This specification is called from the LOOP_BEGIN action specification. This specification calls the Select Mode panel.

- PLG_MODULE_END

  This specification is called from the MODULE_END action specification. If the FLASH_4072A framework execution mode is set to Auto, this specification calls the Execute Test panel.

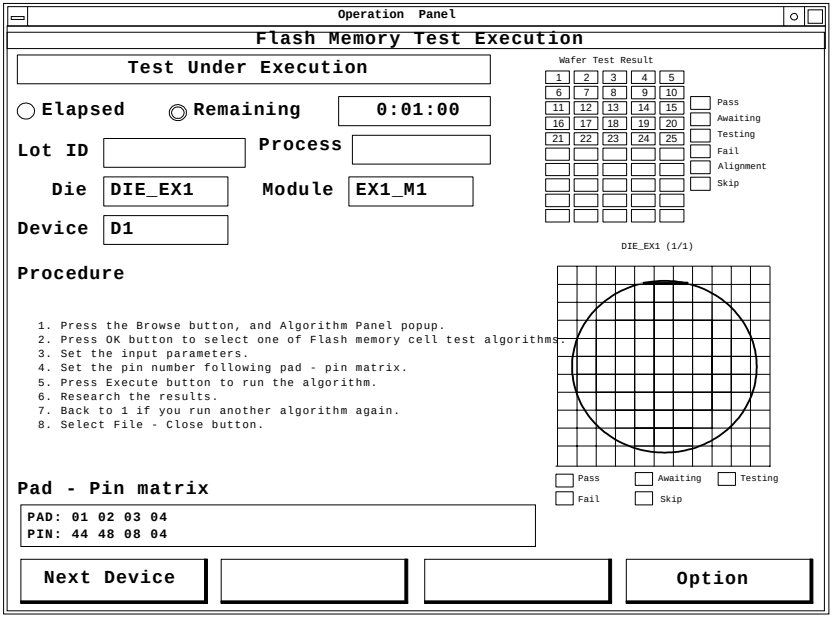
Panel Specifications

The following panel specifications have been added to the FLASH_4072A framework.

- "DISP_FMCT_RUN" (Flash Memory Test Execution panel)
- "FMC_SEL_TESTSEQ" (Retest/Reject Test panel)
- "SEL_FMC_MODE" (Select Mode panel)

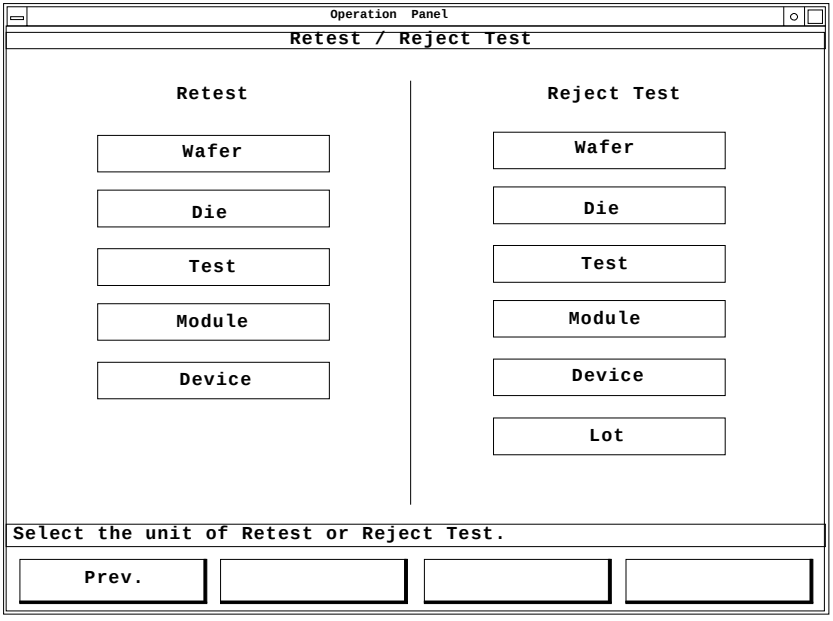
DISP_FMCT_RUN

This specification creates the Flash Memory Test Execution panel. Use this panel to execute a test in the FLASH_4072A framework Interactive mode.



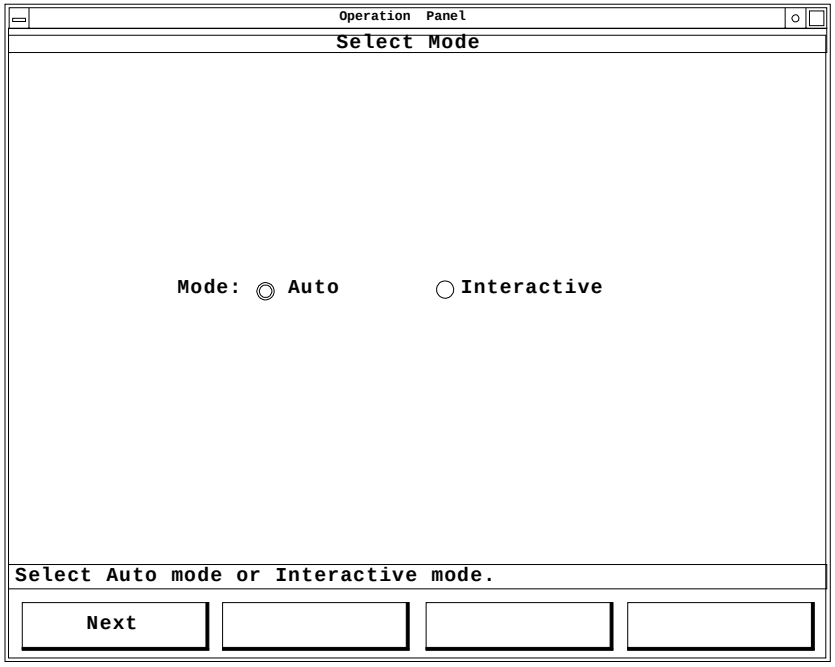
FMC_SEL_TESTSEQ

This specification creates the Retest/Reject Test panel. To display this panel, click the Option button on the Flash Memory Test Execution panel. Use this panel to repeat or reject a test while the framework is running. The buttons on the panel call the FMC_SEL_TESTSEQ_xxxx action specifications. See “Action Specifications”.



SEL_FMC_MODE

This specification creates the Select Mode panel. Use this panel to select the FLASH_4072A framework execution mode, either Auto or Interactive.



7 Sample Test Plan

This chapter explains the contents of the FLASH_DEMO sample test plan.

- PB4FROM Probe Specification
- FLASH_WAF Wafer Specification
- DIE_EX1 Die Specification
- FLASH_DEMO Test Specification

NOTE

FLASH_4072A Framework Execution Mode

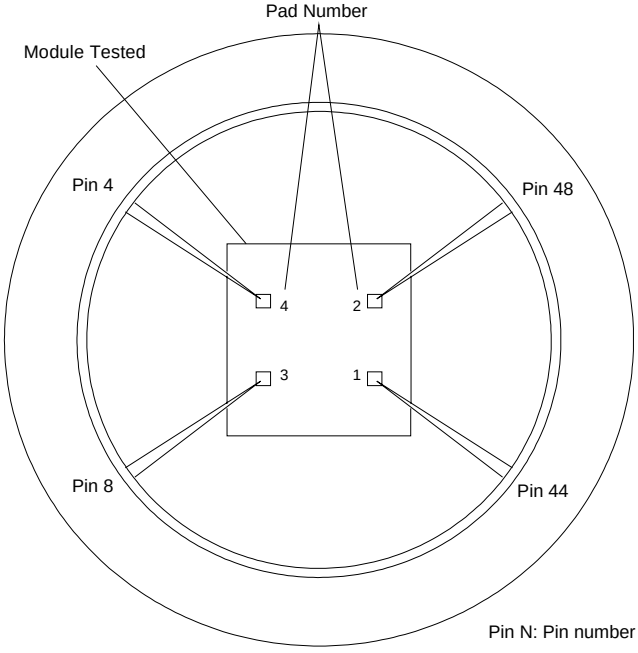
The FLASH_4072 framework provides two execution modes, Interactive and Auto.

In Auto mode, the framework executes the algorithms defined in your test plan. This operation is the same as the HPSTD framework. To execute a specific test, modify the FLASH_DEMO test specification, or create a new test specification.

In Interactive mode, the framework calls the Algorithm Panel. You can use the Algorithm Panel to select an algorithm and execute the test while the framework is running. Use the FLASH_DEMO test specification without modification.

Probe Specification

The probe specification defines the relationship between the pad number of the module tested and the pin number of the switching matrix. The FLASH_DEMO sample test plan defines the PB4FROM probe specification.



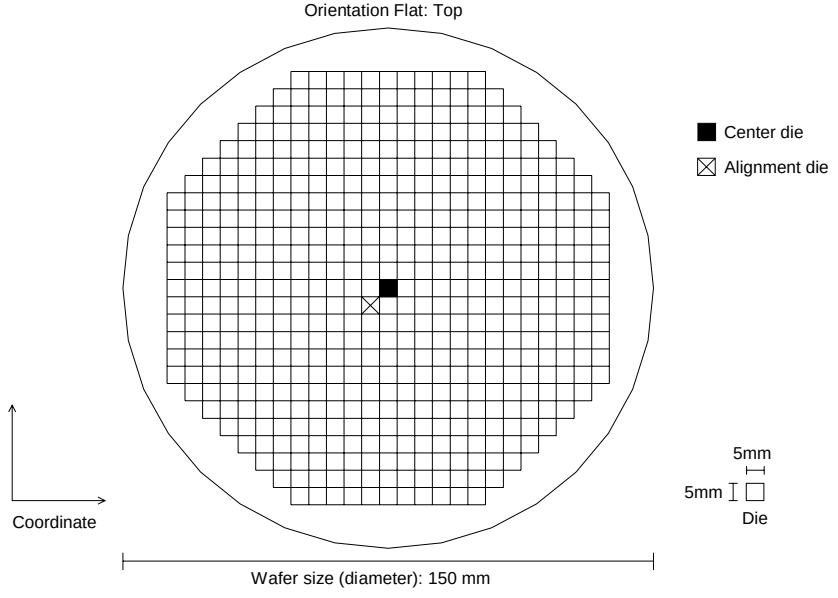
Wafer Specification

The wafer specification registers the attribute information and the dice on the wafer being tested. The FLASH_DEMO sample test plan defines the FLASH_WAF wafer specification.

The wafer specification body menu defines the test die, Die Name: DIE_EX1, Die Index: (0,0).

The wafer specification attribute menu defines the following attribute information. (See figure below.)

- Wafer size (diameter): 150 mm
- Die step index size: 5 mm (X), 5 mm (Y)
- Orientation flat: Bottom (FLAT=0)
- Alignment die index: (0,0)
- Center die index: (1,1)
- Coordinate: see below (COORDINATE=1)



Die Specification

The die specification specifies the module information on the die tested. The names of the die specifications to test must be defined in the wafer specification. The FLASH_DEMO sample test plan defines the DIE_EX1 die specification.

DIE_EX1 Die Specification:

Module Name	X [um]	Y [um]
EX1_M1	0.0	0.0

Test Specification

The test specification specifies the modules and devices to be tested, the test items (algorithms), and the pad numbers of the device terminals. The sample test plan defines the FLASH_DEMO test specification.

FLASH_DEMO Test Specification:

Module Name	Device Name	Algorithm	Input	Pad Number	Output
EX1_M1	D1	IF	AMODE > 0		
		IS_CALG			
		ELSE			
		ACTION	“DIRECT_EXEC”		
		END IF			

The module names defined in the test specifications must also be defined in the die specification.

The FLASH_4072A framework needs the lines IF, ELSE, ACTION, and END IF. These lines must not be removed.

The algorithms used for a test must be inserted between the IF line and the ELSE line. These algorithms are executed in the FLASH_4072A framework Auto execution mode (AMODE=1). For the actual test, remove the IS_CALG line, and insert the measurement algorithms you desire.

In Interactive mode, the “DIRECT_EXEC” action specification is executed, and the Algorithm Panel is opened so algorithms can be selected.