Agilent 4155C
Semiconductor Parameter Analyzer
Agilent 4156C
Precision Semiconductor Parameter Analyzer

Programmer’s Guide
Programming Overview

Agilent 4155C/4156C can be fully controlled from an external computer or by using built-in Instrument BASIC (IBASIC) controller. IBASIC is a programming environment that allows full control of the 4155C/4156C without using an external computer.

The 4155C/4156C has three command modes:

• **4155/4156 SCPI command mode**
  SCPI means Standard Commands for Programmable Instruments. This mode is the default mode of the 4155C/4156C, and allows you to control all functions of the 4155C/4156C.

• **4155/4156 FLEX command mode**
  FLEX means Fast Language for EXecution. This mode allows you to control measurement functions of the 4155C/4156C. Command execution is faster than the SCPI command mode.

• **4145 syntax command mode**
  This mode allows you to execute the 4145A/B programs on the 4155C/4156C directly with little or no modification. In this command mode, you cannot control all functions of the 4155C/4156C.
How to Migrate the 4145A/B Programs

The 4145A/B Auto Sequence Program (ASP) programs run on the 4145A/B built-in programming environment and allow basic control of the 4145A/B without using an external computer. To run the ASP programs on the 4155C/4156C, you do one of the following and execute the program in the 4155/4156 SCPI command mode:

- Create a program that performs the same operations as the desired ASP program by using the IBASIC editor typing aid softkeys to enter commands that correspond to each ASP command. This program can run on IBASIC only, not on an external computer. Refer to Chapter 5 for details.

- Create a program using SCPI commands that performs same operations as the desired ASP program. This program can run on IBASIC or on an external computer. Refer to “Programming Example for the 4145 Users” in Chapter 2 for details.

The 4145A/B GPIB programs run on an external computer and allow full control of the 4145A/B. To run these programs on the 4155C/4156C, do one of the following:

- Directly run the 4145A/B program on the 4155C/4156C with little or no modification. You must run this program in the 4145 syntax command mode from IBASIC or an external computer. Refer to Chapter 4 for details.

- Create a program using SCPI commands that performs same operations as the 4145A/B program. You must run this program in the 4155/4156 SCPI command mode from IBASIC or an external computer.

- Create a program using FLEX commands that performs same operations as the 4145A/B program. You must run this program in the 4155/4156 FLEX command mode from IBASIC or an external computer.
In This Manual

This manual describes how to control the 4155C/4156C by using GPIB commands from an external computer or built-in Instrument BASIC.

This manual consists of the following chapters:

- Using Instrument BASIC
- 4155C/4156C SCPI Command Programming
- 4155C/4156C FLEX Command Programming
- Running 4145A/B Program Directly on 4155C/4156C
- ASP-Like IBASIC Programming

Refer to SCPI Command Reference for SCPI commands. And refer to GPIB Command Reference for the FLEX commands and for the 4145 syntax commands.


Text Conventions

The following text conventions are used in this manual:

key Represents a key physically located on the 4155C/4156C or external keyboard.

Screen Text Represents text that appears on screen of the 4155C/4156C.

Italic Refers to a related document, or is used for emphasis.
Contents

1. Using Instrument BASIC
   Before Operating IBASIC ......................................................... 1-3
   To Switch Screen Mode ....................................................... 1-3
   To Use the Help Function ...................................................... 1-4
   Creating and Executing a Simple IBASIC Program ...................... 1-6
   Modifying Program by using Editor Functions .......................... 1-8
   Saving and Getting a Program ............................................... 1-12
   Summary of Softkeys and Keyboard Operations for Editor .......... 1-14
   Other Basic Features of IBASIC .............................................. 1-15
   Branching/Repeating ......................................................... 1-15
   Subprograms and COM Blocks ................................................. 1-16
   IBASIC Basic Operation Tasks .............................................. 1-17
   IBASIC Editor Tasks .......................................................... 1-20
   Controlling IBASIC from External Computer ............................ 1-24
   IBASIC Screen ................................................................. 1-28
      All IBASIC Screen ............................................................ 1-29
      IBASIC Status Screen ....................................................... 1-31
   Keys for IBASIC ............................................................... 1-32
      Front-panel Keys for IBASIC .............................................. 1-32
      Primary Softkeys in Idle, Pause, Run, or Input? execution status . 1-35
      Secondary Softkeys in Idle or Pause execution status .............. 1-36
      Secondary Softkeys in Run or Input? execution status .......... 1-37
      Primary Softkeys in Edit execution status ............................ 1-37
      Secondary Softkeys in Edit execution status ......................... 1-38
      External Keyboard ......................................................... 1-41
   4155C/4156C Specific IBASIC Keywords .................................. 1-45
   Available I/O Resources for IBASIC ....................................... 1-47
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCD</td>
<td>1-48</td>
</tr>
<tr>
<td>GPIB Interfaces</td>
<td>1-49</td>
</tr>
<tr>
<td>Parallel Interface</td>
<td>1-49</td>
</tr>
<tr>
<td>Built-in Flexible Disk Drive</td>
<td>1-50</td>
</tr>
<tr>
<td>Differences from 4155A/4156A Programming</td>
<td>1-51</td>
</tr>
<tr>
<td>Using Text Display and Graphics Display</td>
<td>1-52</td>
</tr>
<tr>
<td>Using :HCOP:DEST Command</td>
<td>1-53</td>
</tr>
<tr>
<td>Using :HCOP:DEV:LANG Command</td>
<td>1-54</td>
</tr>
<tr>
<td>2. SCPI Command Programming</td>
<td></td>
</tr>
<tr>
<td>SCPI Programming Basics</td>
<td>2-3</td>
</tr>
<tr>
<td>Preparation before Controlling 4155C/56C via GPIB</td>
<td>2-3</td>
</tr>
<tr>
<td>SCPI Command Hierarchy</td>
<td>2-5</td>
</tr>
<tr>
<td>To Control 4155C/56C by HP BASIC Programming</td>
<td>2-6</td>
</tr>
<tr>
<td>Getting Started on SCPI Programming</td>
<td>2-8</td>
</tr>
<tr>
<td>Creating a Simple Measurement Program</td>
<td>2-9</td>
</tr>
<tr>
<td>Programming for Data Extraction</td>
<td>2-19</td>
</tr>
<tr>
<td>Complete Example Program for Vth Measurement</td>
<td>2-24</td>
</tr>
<tr>
<td>Example Application Setup for Vth Measurement</td>
<td>2-26</td>
</tr>
<tr>
<td>Programming: Measurement Setup</td>
<td>2-30</td>
</tr>
<tr>
<td>To Set or Change 4155/56 Setup Data Values</td>
<td>2-31</td>
</tr>
<tr>
<td>To Read 4155/56 Setup Data Values</td>
<td>2-33</td>
</tr>
<tr>
<td>Programming: Auto Analysis Setup</td>
<td>2-34</td>
</tr>
<tr>
<td>Programming: Measurement Execution</td>
<td>2-37</td>
</tr>
<tr>
<td>To Execute a Measurement</td>
<td>2-38</td>
</tr>
<tr>
<td>To Force Stress</td>
<td>2-40</td>
</tr>
<tr>
<td>To Start the Knob Sweep Function</td>
<td>2-42</td>
</tr>
<tr>
<td>To Control Standby Units</td>
<td>2-43</td>
</tr>
<tr>
<td>To Control the E5250A Low Leakage Switch Mainframe</td>
<td>2-45</td>
</tr>
</tbody>
</table>
Contents

Programming: File Operation .................................................. 2-47
To Store Setup Data ............................................................. 2-48
To Store Measurement Data ..................................................... 2-49
To Load Setup Data .............................................................. 2-50
To Load Measurement Data .................................................... 2-51
To Store Spreadsheet Data ...................................................... 2-52

Programming: Data Transfer ..................................................... 2-54
To Read 4155/56 Measurement Data ........................................ 2-55
To Transfer Data to 4155C/4156C (Using User Variable) ............. 2-56

Programming: Print/Plot Operation .......................................... 2-60
To Output Setup Data to Printer/Plotter ................................... 2-63
To Output Graphics Result Data to Printer/Plotter ...................... 2-67
To Output List Results Data to Printer/Plotter ............................ 2-70
To Dump Screen Image to Printer/Plotter .................................. 2-73
To Save Print/Plot Data to a File ............................................ 2-76
To Save Hardcopy Image to a File ........................................... 2-78

Other Programming Tips ....................................................... 2-79
Disabling Instrument Screen Update to Improve Speed ................ 2-79
Auto-loading of Files ............................................................ 2-80
Differences From 4155B/4156B SCPI Commands ......................... 2-81
Differences From 4155A/4156A SCPI Commands ......................... 2-83
Programming Example for 4145 Users ..................................... 2-85

3. FLEX Command Programming

Programming Basics .............................................................. 3-4
Controlling 4155C/4156C via GPIB .......................................... 3-4
Controlling 4155C/4156C Using HP BASIC Programming ............. 3-5

High-Speed Spot Measurements .............................................. 3-6
Spot Measurements .............................................................. 3-9
Contents

1 Channel Pulsed Spot Measurements .................................................. 3-12
Staircase Sweep Measurements ......................................................... 3-15
Pulsed Sweep Measurements ............................................................. 3-22
Staircase Sweep with Pulsed Bias Measurements ............................... 3-27
Sampling Measurements ................................................................. 3-32
Quasi-static CV Measurements ......................................................... 3-37
Linear Search Measurements ........................................................... 3-41
Binary Search Measurements ........................................................... 3-45
Stress Force ...................................................................................... 3-49
Controlling PGU .............................................................................. 3-52
Using Program Memory ..................................................................... 3-55
Using Enhanced Sweep Stop Function ............................................. 3-60
Reading Time Stamp Data .................................................................. 3-62
Reading and Writing Data to a File ................................................... 3-67
Printing Data ..................................................................................... 3-76
Reading Binary Output Data ............................................................. 3-79
Using the US42 Control Mode ........................................................... 3-83
Programming Tips ............................................................................ 3-86
Confirming Operation Status ............................................................. 3-86
Improving Measurement Speed ....................................................... 3-87
Disabling Auto Calibration .............................................................. 3-89

4. Running 4145A/B Program Directly on 4155C/4156C

Non-supported Commands ............................................................... 4-3
Considerations about Differences .................................................... 4-4

# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spot Measurement</td>
<td>4-4</td>
</tr>
<tr>
<td>Sweep Steps in Logarithmic Step Mode</td>
<td>4-4</td>
</tr>
<tr>
<td>Terminator</td>
<td>4-5</td>
</tr>
</tbody>
</table>

## 5. ASP-Like IBASIC Programming

- Creating ASP-like IBASIC Programs                                      | 5-3  |
- Programming Tips                                                        | 5-6  |
- Reading 4155/56 Data to IBASIC Variables                                | 5-7  |
- Auto Scaling                                                            | 5-8  |
- 4145 ASP and 4155C/4156C Corresponding Keywords                        | 5-9  |
- ASP-like Commands                                                       | 5-10 |
  - EXECUTE                                                               | 5-10 |
  - GET SETUP Directive                                                   | 5-11 |
  - SINGLE Directive                                                      | 5-11 |
  - STANDBY directive                                                     | 5-12 |
  - STRESS Directive                                                      | 5-12 |
  - AUTO-SCALE Directive                                                  | 5-12 |
  - SAVE DATA Directive                                                   | 5-13 |
  - READ DATA VARIABLE Directive                                          | 5-14 |
  - DEFINE USER VARIABLE Directive                                        | 5-15 |
  - PRINT/ PLOT Directive                                                 | 5-16 |
  - CURVE PLOT Directive                                                  | 5-16 |
1 Using Instrument BASIC
Using Instrument BASIC

The Instrument BASIC (IBASIC) is a system controller built into Agilent 4155C/4156C. By using IBASIC, you can run a program to control the 4155C/4156C and other instruments (connected via interfaces of the 4155C/4156C) without using an external computer.

IBASIC is a subset of HP BASIC. Programs created by IBASIC can run on an HP BASIC controller with little or no modification.

This chapter consists of the following sections.

The following sections provide step-by-step instructions to operate IBASIC by using examples. You can learn the basics of IBASIC programming and operations. If you are not familiar with IBASIC, we recommend to read through these sections first.

• Before Operating IBASIC
• Creating and Executing a Simple IBASIC Program
• Modifying Program by using Editor Functions
• Saving and Getting a Program
• Summary of Softkeys and Keyboard Operations for Editor
• Other Basic Features of IBASIC

The following sections are a task oriented reference for IBASIC. You can quickly find the desired IBASIC task.

• IBASIC Basic Operation Tasks
• IBASIC Editor Tasks
• Controlling IBASIC from External Computer

The following sections provide the reference information of IBASIC.

• IBASIC Screen
• Keys for IBASIC
• 4155C/4156C Specific IBASIC Commands
• Available I/O Resources

The following section provides the differences from the 4155A/4156A IBASIC programming.

• Differences from 4155A/4156A Programming
Before Operating IBASIC

The 4155C/4156C provides the following three screen modes for operating IBASIC.

- **“All IBASIC“ screen**
  
  Entire screen including softkeys is used for IBASIC, so no instrument setup screen is displayed.
  
  You can execute programs, but no instrument setup screen appears in this mode.

- **“IBASIC Status” screen**
  
  Softkeys and bottom two lines are used for IBASIC. Rest of screen is for instrument setup screen.
  
  In this mode, you can start the IBASIC editor. The displayed softkeys are for IBASIC operation. You can execute IBASIC commands interactively. Characters you type are displayed at the bottom of the screen.

- **“All Instrument“ screen**
  
  This is regular instrument screen and the default display mode at power on. Entire screen is for instrument setup screen, and all softkeys are for interactive use of instrument. In this mode, you cannot use the IBASIC editor. Only the front-panel keys of IBASIC key group and Ctrl+U (Run) and Ctrl+P (Pause) on external keyboard are available to execute or pause program for the Instrument BASIC from this screen mode.

For details about the Instrument BASIC screens, refer to “IBASIC Screen” on page 1-28.

To Switch Screen Mode

To switch the screen mode, repeat one of the following instruction until the desired screen is displayed. This operation toggles screen display as shown below:

- From instrument front panel, press Display of IBASIC key group.
- From external keyboard, press Ctrl+G.
Using Instrument BASIC
Before Operating IBASIC

To Use the Help Function

By using the built-in help function of the 4155C/4156C, you can easily get information (name, syntax, and description) about programming commands, and can enter the desired command into the program without typing.

To start the help function for the programming commands, press Help key while you are in the IBASIC editor.

In the help function, the programming commands are divided into the following three categories, which you can access by secondary softkeys.

<table>
<thead>
<tr>
<th>Softkey</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBASIC</td>
<td>IBASIC commands.</td>
</tr>
<tr>
<td>PAGE IMAGE CMD</td>
<td>SCPI commands specific for the 4155C/4156C.</td>
</tr>
<tr>
<td>PAGE IMAGE CMD</td>
<td>These are the help commands associated with</td>
</tr>
<tr>
<td>SCPI CMD</td>
<td>the instrument setup screen that begin with</td>
</tr>
<tr>
<td></td>
<td>:PAGE.</td>
</tr>
<tr>
<td>SCPI CMD</td>
<td>Standard SCPI commands.</td>
</tr>
</tbody>
</table>

The upper part of the help screen displays a list of the command names. The lower part displays a description of the selected (by field pointer) command.

There are no Help function for the 4155/4156 FLEX command mode and the 4145 Syntax command mode.
To move the field pointer

To move the field pointer, refer to the following table:

<table>
<thead>
<tr>
<th>Rotary knob or Arrow keys</th>
<th>Basically, you move the field pointer by using the rotary knob or arrow keys.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGE CONTROL keys</td>
<td>Field pointer moves to first PAGE IMAGE command that is associated with the pressed key. PAGE CONTROL keys are Chan, Meas, Display, Graph/List, Stress, and System.</td>
</tr>
<tr>
<td>MEASUREMENT keys</td>
<td>Field pointer moves to the PAGE IMAGE command that is associated with the pressed key. MEASUREMENT keys are Single, Repeat, and Append.</td>
</tr>
<tr>
<td>Get and Save front-panel keys</td>
<td>Field pointer moves to the associated SCPI command.</td>
</tr>
<tr>
<td>Alphabetical keys</td>
<td>Field pointer moves to next command that has a keyword that begins with same letter as the pressed key. If you are in the PAGE IMAGE command category, search is only within the instrument screen group of the currently selected command.</td>
</tr>
</tbody>
</table>

To search for a command

To search for a command:

1. Press SEARCH secondary softkey.
2. Type in command string that you want to search for, then press Enter

To enter a command into the editor

The command specified by the field pointer is displayed on the entry line. If you press Enter, the command is entered into the editor.

If command specified by the field pointer is a PAGE IMAGE or SCPI command, first select the OUTPUT @Hp415x secondary softkey. The entry line becomes OUTPUT @Hp415x; "command", where command is command specified by field pointer. Then, press Enter.

OUTPUT @Hp415x; "command" is entered into the editor.
Creating and Executing a Simple IBASIC Program

In this section, let's try to create and execute a simple program.

Before creating a program in the IBASIC editor, first change the screen display mode to IBASIC Status screen mode or All IBASIC screen mode by pressing IBASIC Display key as described in “To Switch Screen Mode” on page 1-3. In following sections, the All IBASIC screen display mode is used.

1. Editing
2. Exiting from Editor
3. Executing Program

Step 1 Editing

Select EDIT secondary softkey or type EDIT, then press Enter.

10

The following program prints the numbers from 1 to 10. Type as follows:

10 FOR I=1 TO 10
20 PRINT I
30 NEXT I
40 END

NOTE

To start the editor at a specific program line or label
Type EDIT linenum or EDIT label.
For example, if you type EDIT 30, the cursor appears at line 30. If you do not specify a line number or label, the cursor will appear at line 10.

NOTE

Always insert mode

Editor is always in insert mode, and cannot be changed to overwrite mode. If you typed wrong characters, use Backspace to move back a character, or move cursor using ← key, then use Delete to delete a character. Then type correct characters.

NOTE

Program End

In IBASIC, END must be at end of main program. In above example, line 40 is the last line of the program.
Step 2  Exiting from Editor
Select the End edit primary softkey to exit from the editor.

Step 3  Executing Program
To execute the program, press Run of the IBASIC key group, select RUN primary softkey, or type RUN and press Enter. The following should be displayed on the screen:

1
2
3
4
5
6
7
8
9
10

NOTE  If an error message appears
If an error message appears, you probably typed wrong characters. The error message indicates the line number where the error occurs. You need to modify the line.
Modifying Program by using Editor Functions

In this section, you can learn the following editor functions:

1. Inserting lines
2. Deleting a line
3. Renumbering
4. Inserting characters
5. Recalling deleted line
6. Indenting
**Step 1**

**Inserting Lines**

Type **EDIT 20**, then press **Enter**. Cursor appears at line 20.

```
10 FOR I=1 TO 10
20 PRINT I
30 NEXT I
40 END
```

Select Insert line or press **Insert** to insert a line above line 20.

```
10 FOR I=1 TO 10
11
20 PRINT I
30 NEXT I
40 END
```

Type as follows:

```
10 FOR I=1 TO 10
11 PRINT I^2
20 PRINT I
30 NEXT I
40 END
```

I^2 means the second power of I. The above program increments I from 1 to 10, and displays second power of I and I for each step.

Select End edit to exit editor, then press **Run** to execute the program. The following is displayed:

```
1
1
4
2
9
3:
:
81
9
100
10
```
Modifying Program by using Editor Functions

Step 2

Deleting a Line

Type EDIT 20 to start editor at line 20.

```
10 FOR I=1 TO 10
11 PRINT I^2
20 PRINT I
30 NEXT I
40 END
```

Then, select Delete line or press Shift+Delete to delete line 20. The result is as follows:

```
10 FOR I=1 TO 10
11 PRINT I^2
30 NEXT I
40 END
```

The above program increments I from 1 to 10, and displays the second power of I at each step.

If you exit editor and execute the program, the following is displayed:

```
1
4
9
:
:
81
100
```

Step 3

Renumbering

In above example, line numbers are not in equal increments. To change the line number increment to 10, select Re-number softkey. Line numbers will be changed as follows:

```
10 FOR I=1 TO 10
20 PRINT I^2
30 NEXT I
40 END
```

If you use the Re-number softkey, the renumbering is always as follows: first line is 10 and the increment is 10.

If you desire other numbering, you need to exit the editor, and use the REN command. For example, if you want first line number to be 100 and increment to be 20, type as follows:

```
REN 100, 20 Enter
```
Inserting Characters

Type `EDIT 20`, then press `Enter`.

```
10 FOR I=1 TO 10
20 PRINT I^2
30 NEXT I
40 END
```

Move the cursor by using right key.

```
10 FOR I=1 TO 10
20 PRINT I^2
30 NEXT I
40 END
```

Then type `I`, as follows:

```
10 FOR I=1 TO 10
20 PRINT I,I^2
30 NEXT I
40 END
```

Above program increments `I` from 1 to 10, and displays `I` and the second power of `I` on one line at each step. Exit editor, then execute the program. The following is displayed:

```
1  1
2  4
3  9
  ..
9  81
10 100
```

Recalling Deleted Line

To restore the most recently deleted line, press `Recall` front-panel key.

Indenting

Move to desired line, then select `Indent` to indent the line. Indenting makes the program flow easier to understand.

```
10 FOR I=1 TO 10
20 PRINT I,I^2
30 NEXT I
40 END
```
Using Instrument BASIC
Saving and Getting a Program

Saving and Getting a Program

The created program can be saved to a diskette. So, you can get the saved program from the diskette, then execute it.

In this section, you can learn the following file operation tasks:

1. Saving a Program
2. Listing Contents of Diskette
3. Clearing a Program
4. Getting a Program

Step 1  Saving a Program

Insert a diskette into the built-in flexible disk drive. Then, type `SAVE "filename"`, then press `Enter`. For this example, we will type `SAVE "PROG1"`.

Step 2  Listing Contents of Diskette

Type `CAT` to list contents of the diskette.

If you are using an MS-DOS format diskette, the display is similar to the following example:

```
DIRECTORY : \:INTERNAL,4
LABEL: 4156
FORMAT: DOS
AVAILABLE SPACE : 5692
FILE NAME TYPE RECS LEN DATE TIME PERMISSION
--------------- ----- ----- ----- ---------- ------------
PROG1 DOS65 1 27-Jun-94 14:30 RW-RW-RW-
```

You can also check the contents of the diskette by using the filer (SYSTEM : FILER screen) of the 4155C/4156C. But you cannot save and get the IBASIC programs by using the filer.
Step 3

Clearing a Program

To clear the program, enter the editor, then select Scratch softkey. Then, select Yes secondary softkey.

Existing program will be cleared, and following is displayed:

```
10 COM @Hp415x
20 ASSIGN @Hp415x TO 800
30!
9990 END
```

COM @Hp415x and ASSIGN @Hp415x TO 800 are used to control the 4155C/4156C as follows.

• COM @Hp415x
  Declares COM so that subprograms can access the I/O path (that is assigned in line 20) for controlling the 4155C/56C. Refer to the Instrument BASIC Users Handbook for details.

• ASSIGN @Hp415x TO 800
  Assigns the I/O path for controlling the 4155C/56C. 800 means built-in IBASIC controller.

Refer to “Subprograms and COM Blocks” on page 1-16.

NOTE

To ASSIGN I/O path

• Built-in IBASIC controller
  Specify select code 8. For the GPIB address, you can use any number between 0 to 31. Refer to the following example:

```
10 ASSIGN @Hp4155 TO 800
```

• HP BASIC on an external computer
  Specify the select code of the external computer. And specify the GPIB address that you entered into the GPIB ADDRESS field on the SYSTEM: MISCELLANEOUS screen. In the following example, the select code of the external computer is 7 and GPIB address of the 4155C/4156C is 17:

```
10 ASSIGN @Hp4155 TO 717
```

Step 4

Getting a Program

Type GET "PROG1", then press Enter. When the LED turns off, enter the editor if you want to display the program.
# Summary of Softkeys and Keyboard Operations for Editor

## Front-panel keys

<table>
<thead>
<tr>
<th>Key</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrow keys</td>
<td>Move the cursor</td>
</tr>
<tr>
<td>Delete</td>
<td>Delete character</td>
</tr>
<tr>
<td>Recall</td>
<td>Recall most recently deleted line</td>
</tr>
</tbody>
</table>

## External Keyboard

<table>
<thead>
<tr>
<th>Key</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Esc</td>
<td>Exit editor</td>
</tr>
<tr>
<td>F1 to F8</td>
<td>Primary softkeys</td>
</tr>
<tr>
<td>Shift+F1 to F7</td>
<td>Secondary softkeys</td>
</tr>
<tr>
<td>F9</td>
<td>Toggle screen mode</td>
</tr>
<tr>
<td>F11</td>
<td>Clear to end</td>
</tr>
<tr>
<td>Insert</td>
<td>Insert line</td>
</tr>
<tr>
<td>Delete</td>
<td>Delete character at cursor</td>
</tr>
<tr>
<td>Shift+Delete</td>
<td>Delete line</td>
</tr>
<tr>
<td>Home</td>
<td>Beginning of line</td>
</tr>
<tr>
<td>End</td>
<td>End of line</td>
</tr>
<tr>
<td>Page Up or Page Down</td>
<td>Scroll pages</td>
</tr>
</tbody>
</table>

## Primary Softkeys

<table>
<thead>
<tr>
<th>Key</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Back space</td>
<td>Delete character before cursor</td>
</tr>
<tr>
<td>Insert line</td>
<td>Insert line</td>
</tr>
<tr>
<td>Delete line</td>
<td>Delete line</td>
</tr>
<tr>
<td>Re-number</td>
<td>Renumber the lines</td>
</tr>
<tr>
<td>Indent</td>
<td>Indent the line</td>
</tr>
<tr>
<td>Scratch</td>
<td>Clear program</td>
</tr>
<tr>
<td>End edit</td>
<td>Exit editor</td>
</tr>
</tbody>
</table>
Other Basic Features of IBASIC

This section describes the following:

• Branching/Repeating
• Subprograms and COM Blocks

Branching/Repeating

Branch and Repeat Keywords of IBASIC are shown in the following table:

<table>
<thead>
<tr>
<th>IBASIC Keyword</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>FOR, NEXT</td>
<td>Repeat specified number of times.</td>
</tr>
<tr>
<td>IF THEN, ELSE, END IF</td>
<td>Branch.</td>
</tr>
<tr>
<td>WHILE, END WHILE</td>
<td>Repeat until specified condition is false.</td>
</tr>
<tr>
<td>REPEAT, UNTIL</td>
<td>Repeat until specified condition is true.</td>
</tr>
</tbody>
</table>

Following program tests 1000 devices, and judges them pass or fail.

1  COM Data(1:1000)
2  DIM Id(1:1000)
10 EXECUTE ("GETSETUP 'SAMPL.MES'")
20 FOR I=1 TO 1000
30 EXECUTE ("SINGLE")
40 EXECUTE ("READDATAVAR 'Id'")
50 IF Id(I)<1E-6 THEN
60 PRINT "FAIL LOWER"
70 Data(I)=1
80 ELSE
90 IF Id(I)>1E-4 THEN
100 PRINT "FAIL HIGHER"
110 Data(I)=2
120 ELSE
130 PRINT "PASS"
140 Data(I)=0
150 END IF
160 END IF
170 NEXT I
180 CALL Save_data
190 END
200!
210 SUB Save_data
220 COM Data(*)
230 CREATE "data_file",1
240 ASSIGN @File TO "data_file";FORMAT ON
250 OUTPUT @File;Data(*)
260 ASSIGN @File TO *
270 SUBEND
Using Instrument BASIC
Other Basic Features of IBASIC

Subprograms and COM Blocks

One of the most powerful constructs available is the subprogram. A subprogram has its own "context" or state that is distinct from the main program and all other subprograms. There are several benefits of subprograms.

- The subprogram allows you to take advantage of the "top-down design" method of programming.
- You can remove all subtasks from the overall logic flow of the main program.
- You can debug the program by testing each subprogram independently.
- The subprograms can be used to reduce the overall size of the program.
- Libraries of commonly used subprograms can be assembled for widespread use.

Refer to the example program in the previous section. Line 180 calls a subprogram to store data into a DOS file.

```plaintext
160 END IF
170 NEXT I
180 CALL Save_data
190 END
200!
210 SUB Save_data
220 COM Data(*)
230 CREATE "data_file",1
240 ASSIGN @File TO "data_file";FORMAT ON
250 OUTPUT @File;Data(*)
260 ASSIGN @File TO *
270 SUBEND
```

COM blocks

COM blocks are used by the subprogram to communicate with the main program or with other subprograms.

If you create subprograms and want to use common variables between main program and subprograms, you should use COM blocks.

Refer to the above example.

In the main program, line 1 declares that the Data array will be a COM array. Then, the main program assigns values to this array. Line 220 specifies that the subprogram Save_data will also use the Data array. So, Data array of main program can be operated on in the Save_data subprogram.
IBASIC Basic Operation Tasks

This section describes the following basic operations to use the Instrument BASIC.

1. Executing the Instrument BASIC commands
2. Executing program
3. Listing files
4. Retrieving program
5. Saving program

**Step 1  Executing the Instrument BASIC Commands**

1. Confirm your 4155C/4156C is in the following status:
   - a program is not executing.
   - another command is not executing.
   - Editor is not running.
   - the screen is "All IBASIC" screen or "IBASIC Status" screen. For "All Instrument" screen, Run and Pause front-panel keys and Ctrl+U (Run) and Ctrl+P (Pause) on external keyboard are available.

2. Type in commands by using front-panel keys in the ENTRY key group or external keyboard.

3. Press Enter front-panel key or Enter key on external keyboard.

**Step 2  Executing Program**

To execute the program, perform one of the following instruction:

- From instrument front panel, press Run front-panel key in the IBASIC key group.
- From external keyboard, press Ctrl+U on external keyboard.
Using Instrument BASIC
IBASIC Basic Operation Tasks

Step 3  Listing Files

1. Confirm your 4155C/4156C is in the following status:
   - the screen is "All IBASIC" screen.
   - a program is not executing.
   - another command is not executing.
   - Editor is not running.

2. Insert a 3.5 inch diskette (that contains the files you want to list) into the built-in flexible disk drive.
3. Select CAT secondary softkey, then press Enter front-panel key.
   The file names on diskette are listed on the screen.

Step 4  Retrieving Program

1. Confirm your 4155C/4156C is in the following status:
   - the screen is "All IBASIC" screen or "IBASIC Status" screen.
   - a program is not executing.
   - another command is not executing.
   - Editor is not running.

2. Insert the 3.5 inch diskette (that contains the program you want to retrieve) into the built-in flexible disk drive.
3. Select GET "" secondary softkey.
4. Type in file name to be retrieved. Typed name is inserted after first ".
5. Press Enter front-panel key, or Enter key on external keyboard.

NOTE  External disk drive

An external disk drive cannot be connected to the 4155C/4156C. For using a disk drive connected to external controller, see “Controlling IBASIC from External Computer” on page 1-24.
Step 5 Saving Program

1. Confirm your 4155C/4156C is in the following status:
   - the screen is "All IBASIC" screen or "IBASIC Status" screen.
   - a program is not executing.
   - another command is not executing.
   - Editor is not running.

2. Insert a 3.5 inch diskette into the built-in flexible disk drive.


4. Type in name of file to which you want to save program.
   - If the file already exists on the diskette, SAVE cannot be used. If you want to overwrite an existing file, select RE-SAVE secondary softkey instead of SAVE secondary softkey in the previous step.

5. Press Enter front-panel key or Enter key on the external keyboard.

NOTE External disk drive

An external disk drive cannot be connected to the 4155C/4156C. For using a disk drive connected to external controller, see “Controlling IBASIC from External Computer” on page 1-24.
Using Instrument BASIC
 IBASIC Editor Tasks

IBASIC Editor Tasks

This section describes the following tasks to use built-in editor of the Instrument BASIC.

1. Starting the editor
2. Quitting the editor
3. Moving the cursor
4. Inserting characters
5. Deleting character
6. Inserting line
7. Deleting line
8. Scrolling pages
9. Recalling most recently deleted line

Step 1  Starting the Editor

1. Confirm the screen is "All IBASIC" screen or "IBASIC Status" screen.
2. Select EDIT secondary softkey.
3. Press Enter front-panel key or Enter key on the external keyboard.
   
   If you want to start the editor to edit a specific program line, type in the line number or label of the program line, then press Enter front-panel key. The editor starts, and cursor is displayed on specified line.
4. If a program is loaded into the 4155C/4156C, the program is displayed.

   If no program is loaded, 1 0 is automatically displayed and rest of screen is empty.

   If you start the editor from the "IBASIC Status" screen, the screen switches to "All IBASIC" screen, and the editor starts.

Step 2  Quitting the Editor

•  Select End edit primary softkey.

If you started the editor from the "IBASIC Status" screen, the screen returns from "All IBASIC" screen to the "IBASIC Status" screen after you quit the editor.
Using Instrument BASIC  
IBASIC Editor Tasks

### Step 3  
**Moving the Cursor**

- To move the cursor, use the following keys.

<table>
<thead>
<tr>
<th>Direction</th>
<th>Front-panel</th>
<th>Keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up</td>
<td>⇑ key of MARKER/CURSOR group</td>
<td>⇑ key</td>
</tr>
<tr>
<td></td>
<td>Rotate rotary knob counter-clockwise</td>
<td></td>
</tr>
<tr>
<td>Down</td>
<td>⇓ key of MARKER/CURSOR group</td>
<td>⇓ key</td>
</tr>
<tr>
<td></td>
<td>Rotate rotary knob clockwise</td>
<td></td>
</tr>
<tr>
<td>Right</td>
<td>⇒ key of ENTRY group</td>
<td>⇒ key</td>
</tr>
<tr>
<td></td>
<td>⇒ key of MARKER/CURSOR group</td>
<td>Ctrl+F</td>
</tr>
<tr>
<td>Left</td>
<td>⇐ key of ENTRY group</td>
<td>⇐ key</td>
</tr>
<tr>
<td></td>
<td>⇐ key of MARKER/CURSOR group</td>
<td>Ctrl+B</td>
</tr>
<tr>
<td>Beginning of Line</td>
<td>FAST+ ⇐ of MARKER/CURSOR group</td>
<td>Home</td>
</tr>
<tr>
<td></td>
<td>Shift+ ⇐</td>
<td></td>
</tr>
<tr>
<td>End of Line</td>
<td>FAST+ ⇒ of MARKER/CURSOR group</td>
<td>End</td>
</tr>
<tr>
<td></td>
<td>Shift+ ⇒</td>
<td></td>
</tr>
</tbody>
</table>

### Step 4  
**Inserting Characters**

1. Move the cursor to character you want to insert before.
2. Characters you type will be automatically inserted.
3. After you insert characters, you must select the Enter key to enter the line with inserted characters into the program.

Editor is always in insert mode, and cannot be changed to overwrite mode.
**Using Instrument BASIC**

**IBASIC Editor Tasks**

**Step 5  Deleting Character**

1. Move the cursor to character you want to delete.
2. Press key according to the following table:

<table>
<thead>
<tr>
<th>Front-panel</th>
<th>Keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete of ENTRY group</td>
<td>Delete</td>
</tr>
</tbody>
</table>

3. After you delete characters, you must select the Enter key to enter the line with deleted characters into the program.

**Step 6  Inserting Line**

1. Move the cursor to the line that you want to insert a new line before.
2. Press key or softkey according to following table:

<table>
<thead>
<tr>
<th>Front-panel</th>
<th>Keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert line primary softkey</td>
<td>Shift+Insert</td>
</tr>
<tr>
<td></td>
<td>Alt+I</td>
</tr>
</tbody>
</table>

3. After you type in a new line, you must select the Enter key to enter the new line into the program.

**Step 7  Deleting Line**

1. Move the cursor to line you want to delete.
2. Press key or softkey according to the following table:

<table>
<thead>
<tr>
<th>Front-panel</th>
<th>Keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete line primary softkey</td>
<td>Shift+Delete</td>
</tr>
<tr>
<td></td>
<td>Alt+D</td>
</tr>
</tbody>
</table>
Step 8  Scrolling Pages

- To scroll the editor by one-half screen, use the following keys:

<table>
<thead>
<tr>
<th>Direction</th>
<th>Front-panel</th>
<th>Keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up</td>
<td>n.a.</td>
<td>Page Up</td>
</tr>
<tr>
<td>Down</td>
<td>n.a.</td>
<td>Page Down</td>
</tr>
</tbody>
</table>

Step 9  Recalling Most Recently Deleted Line

- To display the line most recently deleted line, use the following keys.

<table>
<thead>
<tr>
<th>Front-panel</th>
<th>Keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recall↓</td>
<td>Shift+Page Up</td>
</tr>
</tbody>
</table>

If you want to enter the line into the program, you must select the Enter key.
Controlling IBASIC from External Computer

This section describes how to control the IBASIC program on the 4155C/4156C from a program that is running on an external computer:

- Controlling execution of a 4155C/4156C program.
- Downloading a program to the 4155C/4156C.
- Uploading a program from the 4155C/4156C.

Before executing a program on external computer to control the 4155C/4156C, do as follows:

1. Connect an GPIB cable from the external computer to the GPIB connector on rear panel of the 4155C/4156C.
2. Set the "4155C/4156C is" field on the SYSTEM: MISCELLANEOUS screen to NOT SYSTEM CONTROLLER.
3. Enter the GPIB address of your 4155C/4156C into the GPIB ADDRESS field.

Step 1: To Control State of the 4155C/4156C Program

:PROGram[:SELe:STAt] command from external computer can control the Instrument BASIC program in the 4155C/4156C as follows:

- To run the program:
  OUTPUT @Hp4155;" :PROGram[:SELe:STAt] RUN"
- To continue the program:
  OUTPUT @Hp4155;" :PROGram[:SELe:STAt] CONT"
- To stop the program:
  OUTPUT @Hp4155;" :PROGram[:SELe:STAt] STOP"
- To pause the program:
  OUTPUT @Hp4155;" :PROGram[:SELe:STAt] PAUSE"
Step 2  To Download a Program to the 4155C/4156C

To download a program from the external computer to the 4155C/4156C, you need to use the :PROG:SELECT:DEFINE command.

The following is an example of an HP BASIC program (running on external computer) that reads an Instrument BASIC program file (ASCII file stored in a disk drive connected to external computer) and downloads it to the 4156C.

```
10   OPTION BASE 1
20    !
30   DIM Line$(1024)
40    !
50   ASSIGN @Hp4156 TO 717
60    !
70   OUTPUT @Hp4156;" :PROG:DEL:ALL" !Clears program in 4156C
80  File_name$="prog"
90   ASSIGN @File TO File_name$ !Opens file and assigns data path
100  OUTPUT @Hp4156;" :PROG:DEF #0" !Sends header to 4156C
110 ON ERROR GOTO Done
120 LOOP
130  Line$=""
140 ENTER @File;Line$ !Reads one program line
150  OUTPUT @Hp4156;Line$ !Downloads line to 4156C
160 END LOOP
170 Done:!
180 OFF ERROR
190 OUTPUT @Hp4156;Line$
200 OUTPUT @Hp4156;" END
210 ASSIGN @File TO *
220 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>70</td>
<td>deletes existing the Instrument BASIC program in the 4155C/4156C.</td>
</tr>
<tr>
<td>80</td>
<td>name of file (in disk drive of external computer) that contains desired the</td>
</tr>
<tr>
<td></td>
<td>Instrument BASIC program</td>
</tr>
<tr>
<td>90</td>
<td>opens file and assigns data path</td>
</tr>
<tr>
<td>100</td>
<td>#0 indicates that an indefinite length of parameters (program lines) will be</td>
</tr>
<tr>
<td></td>
<td>downloaded</td>
</tr>
<tr>
<td>110 to 160</td>
<td>reads program lines from the file and downloads them until EOF.</td>
</tr>
<tr>
<td>210</td>
<td>closes file.</td>
</tr>
</tbody>
</table>
Using Instrument BASIC
Controlling IBASIC from External Computer

Step 3

To Upload a Program from the 4155C/4156C

To upload a program from the 4155C/4156C to external computer, you need to use the `:PROG:DEF?` command.

The following is an example of an HP BASIC program (running on external computer) that uploads an Instrument BASIC program (ASCII file) from the 4156C and stores the program on a disk drive that is connected to external computer.

```
10 OPTION BASE 1
20 !
30 DIM Num_dig$[2]
40 INTEGER Byte
50 !
60 ASSIGN @Hp4156 TO 717
70 !
80 OUTPUT @Hp4156;";PROG:DEF?"
90 ENTER @Hp4156 USING "%;2A";Num_dig$
100 PRINT Num_dig$
110 !
120 Byte=VAL(Num_dig$[2])
130 !
140 ALLOCATE Data_byt$(Byte)
150 !
160 FOR I=1 TO Byte
170 ENTER @Hp4156 USING ";A";Data_byt$(I;1) !Enter length of program
180 NEXT I
190 !
200 D=VAL(Data_byt$)
210 PRINT D
220 ALLOCATE Prog$(D)
230 PRINTER IS CRT;WIDTH D
240 ENTER @Hp4156 USING ";X";Prog$ !Enter the program into Prog$
250 PRINT Prog$
260 ENTER @Hp4156;B$
270 PRINT B$
280 !
290 CREATE "prog",1
300 ASSIGN @File TO "prog";FORMAT ON
310 OUTPUT @File;Prog$
320 ASSIGN @File TO *
330 !
340 END
```
### Using Instrument BASIC

#### Controlling IBASIC from External Computer

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>80</td>
<td>Sends :PROGram[:SEL ected]:DEFine? query command.</td>
</tr>
<tr>
<td>90</td>
<td>Reads first two characters of response. These two bytes indicate how many bytes are used to specify length of program.</td>
</tr>
<tr>
<td>160 to 180</td>
<td>Reads the bytes that specify length of program.</td>
</tr>
<tr>
<td>200</td>
<td>Calculates length of program.</td>
</tr>
<tr>
<td>220</td>
<td>Allocates string variables for program.</td>
</tr>
<tr>
<td>240</td>
<td>Reads program.</td>
</tr>
<tr>
<td>260</td>
<td>Reads terminator.</td>
</tr>
<tr>
<td>290</td>
<td>Creates file &quot;prog&quot;</td>
</tr>
<tr>
<td>300</td>
<td>Assigns I/O path to &quot;prog&quot;</td>
</tr>
<tr>
<td>310</td>
<td>Stores program into &quot;prog&quot; file.</td>
</tr>
<tr>
<td>320</td>
<td>Closes file.</td>
</tr>
</tbody>
</table>
### IBASIC Screen

The 4155C/4156C provides the following three screen modes for operating IBASIC:

- **"All IBASIC" screen**
  
  Entire screen including softkeys is used for IBASIC, so no instrument setup screen is displayed.
  
  You can execute programs, but no instrument setup screen appears in this mode.

- **"IBASIC Status" screen**
  
  Softkeys and bottom two lines are used for IBASIC. Rest of screen is for instrument setup screen.
  
  In this mode, you can start the IBASIC editor. The displayed softkeys are for IBASIC operation. You can execute IBASIC commands interactively. Characters you type are displayed at the bottom of the screen.

- **"All Instrument" screen**
  
  This is regular instrument screen and the default display mode at power on. Entire screen is for instrument setup screen, and all softkeys are for interactive use of instrument. In this mode, you cannot use the IBASIC editor. Only the front-panel keys of IBASIC key group and Ctrl+U (Run) and Ctrl+P (Pause) on external keyboard are available to execute or pause program for IBASIC from this screen mode.

Display front-panel key or Ctrl+G (or F9) on external keyboard are used to toggle the screen display mode as shown in the following figure:

---

[Diagram showing screen display modes]
For the "All IBASIC" screen, the entire screen including softkeys is used for IBASIC. The following describes each part of this screen:

**IBASIC Output Area**
Screen output commands of IBASIC (such as `PRINT` and `OUTPUT 1;`) display characters in this area. This area has 23 lines and 58 columns (58 characters in a line).

**Command Entry Line**
IBASIC command you type is displayed on this line. The length of this line is 58 characters.

**System Message and Display Line**
For displaying IBASIC error messages and other system messages, and `DISP` and `INPUT` commands of IBASIC.

**Execution Status**
This field displays the execution status of IBASIC:

- **Idle**: IBASIC program is stopped. IBASIC commands can be executed.
- **Run**: IBASIC program or command is being executed.
- **Pause**: IBASIC program is paused.
- **Input?**: IBASIC program is waiting for input from front-panel keys or external keyboard.
- **Edit**: IBASIC editor is running.
Using Instrument BASIC
IBASIC Screen

Blue Key & Green Key Shift Status

This field displays the shift status of ENTRY front-panel keys. The shift status is controlled by using the blue and green front-panel keys:

The following statuses are displayed:

- Non-shift status: B, b, or G is not displayed. You can enter numeric values.
- Uppercase shift status: B is displayed, G is not displayed. You can enter uppercase alphabet characters.
- Lowercase shift status: b is displayed. G is not displayed. You can enter lowercase alphabet characters.
- Non-alphanumeric status: G is displayed. You can enter one non-alphanumeric character. So, you must press green key before entering each alphanumeric character.

Basically, you can change between these states as follows:

- To toggle between non-shift and shift status: press blue key,
- To toggle between uppercase and lowercase shift status: press green key, then blue key.
- To enter one non-alphanumeric character: press green key, then character.
For the "IBASIC Status" screen, the two bottom lines are used to display the status of IBASIC. These two lines are the same as in “All IBASIC” screen. Refer to “All IBASIC Screen” on page 1-29. Also, the softkeys are for IBASIC.

The other part of the screen is the normal 4155C/4156C screen. This is useful if you want to view a graph of the measurement results while executing IBASIC program.
Using Instrument BASIC

Keys for IBASIC

This section provides information about the following keys for IBASIC:

- Front-panel Keys
- Primary Softkeys
- Secondary Softkeys
- External Keyboard Keys

Front-panel Keys for IBASIC

**PAGE CONTROL key group**

- In "IBASIC Status" screen:
  Changes to "All Instrument" screen and displays the specified screen.
- In "All IBASIC" screen:
  All front-panel keys in this group are ignored.

**MARKER/CURSOR key group**

- In "IBASIC Status" screen:
  When you operate MARKER/CURSOR front-panel keys, the screen changes to
  "All Instrument" screen and function of operated key is executed.
- In all IBASIC screen:
  Rotary Knob
  When the editor is running, the rotary knob moves the cursor vertically in the edit area.
  When the editor is not running, the rotary knob scrolls the IBASIC output area.
  ↑, ↓
  When the editor is running, these keys move the cursor vertically.
  When the editor is not running, these keys scroll the IBASIC output area.
  ←, →
  Moves the cursor horizontally on the IBASIC Editor or Command Entry line.

If you hold down the Fast key, the arrow keys move the cursor faster.
**MEASUREMENT key group**

- In "IBASIC Status" screen:
  - **Single**, **Repeat, Append** Changes the display to GRAPH/LIST: GRAPHICS or GRAPH/LIST: LIST screen and executes the measurement.
  - **Standby** Toggles the operation state of the standby channels between the standby state and the idle state.
  - **Stop** Stops the measurement or stress forcing.
  - **Stress** Changes the display to the STRESS: STRESS FORCE screen and starts to force stress.
  - **Short, Medium, Long** Changes the measurement integration time.

- In "All IBASIC" screen:
  - **Standby** Toggles the operation state of the standby channels between the standby state and the idle state.
  - **Stop** Stops the measurement or stress forcing.

Other front-panel keys in this group are ignored.

**IBASIC key group**
The following front-panel keys of IBASIC key group are available to control IBASIC in any display mode.

- **Run** Executes IBASIC program that is loaded into internal memory of the 4155C/4156C.
- **Pause** Pauses program execution until CONT command is executed or Continue primary softkey is pressed. If the program is modified while paused, RUN command must be used to restart program execution.
- **Display** Toggles the display mode in the following sequence.

![Display Mode Sequence Diagram]
Using Instrument BASIC
Keys for IBASIC

**ENTRY key group**

**Recall** ↓
- When the editor is running, this key displays the last deleted line. To enter this displayed line as part of the program, press **Enter** front-panel key.
- When the editor is not running, this key cycles through the 10 commands that were most recently entered on the Command Entry line.

**Recall** ↑
- When the editor is running, this key is the same as **Recall** ↓.
- When the editor is not running, this key is the same as **Recall** ↓, but cycles through commands in opposite order.

**Save, Get**
These keys are ignored.

Other front-panel keys in **ENTRY** group are available to enter characters on the Command Entry line or Editor. For the usage of the blue and green front-panel keys to enter characters, see "All IBASIC Screen" on page 1-29.

**Other Keys**

**Help**
Displays information about IBASIC. And can be used to select and enter SCPI and IBASIC commands into Editor or Command Entry line.

**Plot/Print**
If present screen is "All IBASIC", dumps "All IBASIC" screen image to the printer or plotter.
If present screen is "IBASIC Status", prints/plots instrument part of screen.
Primary Softkeys in Idle, Pause, Run, or Input? execution status

This section describes the primary softkeys that are displayed during the Idle, Pause, Run, or Input? execution status.

Refer to “Primary Softkeys in Edit execution status” on page 1-37 for primary softkeys that are displayed when the editor is running.

**Step**

1. Executes the paused program line of paused program or the first program line of stopped (idle) program.
2. Displays next program line on system message line of the screen.
3. Pauses program again.

**Continue**

Starts execution of paused program from paused program line.

**RUN**

Starts program execution immediately from first program line.

**Pause**

Pauses program execution immediately. And displays line at which execution was paused.

**Stop**

Stops program execution after current line executes.

**Clear I/O**

Stops I/O operation of program.

**Reset**

Stops program execution immediately.
Using Instrument BASIC
Keys for IBASIC

Secondary Softkeys in Idle or Pause execution status

This section describes the secondary softkeys that are displayed during the Idle or Pause execution status.

For the secondary softkeys that are displayed during the Run or Input? status, refer to “Secondary Softkeys in Run or Input? execution status” on page 1-37.

For the secondary softkeys that are displayed when the editor is running, refer to “Secondary Softkeys in Edit execution status” on page 1-38.

**CAT**

Clears the Command Entry line, and types in CAT.

To list file names on the diskette, press Enter.

**SAVE ""**

1. Clears the Command Entry line.
2. Types in SAVE " ".
3. Positions the cursor after first ".

To save program to diskette, type name of file to which you want to save program, then press Enter.

If file already exists on diskette, program will not be saved.

**RE-SAVE ""**

1. Clears the Command Entry line.
2. Types in RE-SAVE " ".
3. Positions the cursor after first ".

To save program to diskette, type name of file to which you want to save program, then press Enter.

If file already exists on diskette, file will be overwritten, so previous data in file is lost.

**GET ""**

1. Clears the Command Entry line.
2. Types in GET " ".
3. Positions the cursor after first ".

To get a program from diskette, type name of file to get, then press Enter.
**PURGE ""**

1. Clears the Command Entry line.
2. Types in `PURGE " "`.
3. Positions the cursor after first `"`.

To delete a file from diskette, type in the file name to be deleted, then press **Enter**.

**EDIT**

Clears the Command Entry line and types in **EDIT**. To start the editor, press **Enter** front-panel key.

**RENumber**

Clears the Command Entry line and types in **RENumber**.

To re-number lines of a program, type in appropriate parameters, then press **Enter**.

For more details about **RENumber** command, refer to the Instrument BASIC User's Handbook.

### Secondary Softkeys in Run or Input? execution status

When the execution status is **Run or Input?**, user-defined softkeys, which are defined by using **ON KEY** command in the program, are displayed in the secondary softkey area.

### Primary Softkeys in Edit execution status

This section describes the primary softkeys that are displayed when the IBASIC editor is running (**Edit** execution status is displayed).

- **Back space**
  Deletes the character before the cursor.

- **Insert line**
  Inserts a line between the cursor line and the previous line.

- **Delete line**
  Deletes the cursor line.

- **Re-number**
  Changes the program line numbers so that first line is 10 and line number increment is 10.

- **Indent**
  Indents so that all program lines begin at the same position.

- **Scratch**
  Clears the program and all variables not in COM. Before clearing, **YES** and **NO** secondary softkeys are displayed for confirmation.

- **End edit**
  Exits the editor.
Using Instrument BASIC

Keys for IBASIC

Secondary Softkeys in Edit execution status

This section describes the secondary softkeys that are displayed when the IBASIC editor is running (Edit execution status is displayed).

These softkeys help you enter program commands. For commands that require you to type in some parameters, these softkeys display the command. You must enter the parameters, then you must press Enter key to enter the command into the program. For commands that do not have parameters, the commands are entered directly into the program. Commands are entered at the cursor line.

For the EXECUTE command, refer to “EXECUTE” in Chapter 5 for details.

For secondary softkeys that are displayed during Idle or Pause execution status, refer to “Secondary Softkeys in Idle or Pause execution status” on page 1-36.

For secondary softkeys that are displayed during Run or Input? execution status, refer to “Secondary Softkeys in Run or Input? execution status” on page 1-37.

In Edit execution status, there are three kinds of secondary softkey menu. To move to next menu, press MORE secondary softkey.

GET SETUP

1. Displays the following program line for loading a setup file:

   \texttt{EXECUTE("GETSETUP ")}

2. Positions cursor at second double quotes. You enter the file name to be loaded, then select Enter key.

SINGLE

Enters the following program line for triggering a single measurement:

\texttt{EXECUTE("SINGLE")}

STANDBY

1. Displays the following program line for changing the operation state of the standby channels:

   \texttt{EXECUTE("STANDBY ")}

2. Positions the cursor at the second double quote. You enter \texttt{ON} or \texttt{OFF}, then select Enter key.

STRESS

Enters the following program line for triggering stress force:

\texttt{EXECUTE("STRESS")}

AUTO SCALE

Enters the following program line for autoscaling:

\texttt{EXECUTE("AUTOSCALE")}

---

### SAVE DATA
1. Displays the following program line for saving measurement data to a file:
   ``` Basic
   EXECUTE("SAVEDATA ")
   ```
2. Positions the cursor at the second double quote. You enter file name to which you want to save measurement data, then select **Enter** key.

### READ DATA VARIABL
1. Displays the following program line for reading the values of an 4155C/4156C data variable, then storing the values into an IBASIC program variable:
   ``` Basic
   EXECUTE("READDATALVAR ,")
   ```
2. Positions the cursor at the comma. You enter names of the 4155C/4156C data variable and IBASIC program variable, then select **Enter** key.

### DEFINE USER VARIABL
1. Displays the following program line for defining a user variable:
   ``` Basic
   EXECUTE("DEFUSERVAR ,,, ")
   ```
2. Positions the cursor at the first comma. You enter the user variable name, number of data, name of IBASIC program variable that contains desired data, and user variable unit, then select **Enter** key.

### PRINT/PLOT
Enters following program line for printing/plotting the instrument window:
``` Basic
EXECUTE("PRINTPLOT")
```

### CURVE PLOT
Enters following program line for printing/plotting a graphics plot of measurement results:
``` Basic
EXECUTE("CURVEPLOT")
```

### OUTPUT @Hp415x
1. Displays the following program line for outputting a command to the 4155C/4156C:
   ``` Basic
   OUTPUT @Hp415x;"
   ```
2. Positions the cursor at the second double quotes. You enter desired command, then select **Enter** key.

### ENTER @Hp415x
1. Displays the following program line for entering data from the 4155C/4156C:
   ``` Basic
   ENTER @Hp415x;
   ```
2. Positions the cursor after the semicolon. You enter desired variable, then select **Enter** key.
Using Instrument BASIC
Keys for IBASIC

**PAUSE**
Enters the following program line for pausing a program:

```
PAUSE
```

**DISP**
1. Displays the following program line for displaying a message:

```
DISP ""
```
2. Positions the cursor at the second double quotes. You enter the message that you want to display, then select Enter key.

**INPUT**
1. Displays the following program line for assigning keyboard input to program variable:

```
INPUT "",
```
2. Positions the cursor at the second double quote. Enter string that you want to be displayed on the screen, and name of variable in which you want to store keyboard input, then select Enter key.

**IF THEN ELSE END IF**
1. Displays the following program lines for conditional branching:

```
IF THEN
ELSE
END IF
```
2. Positions the cursor before THEN. Fill in as desired, then select Enter key.

**WHILE END WHILE**
1. Displays the following program lines for defining a loop:

```
WHILE END WHILE
```
2. Positions the cursor after WHILE. Fill in as desired, then select Enter key.

**FOR NEXT**
1. Displays the following program lines for defining a loop:

```
FOR = TO STEP
NEXT
```
2. Positions the cursor at =. Fill in as desired, then select Enter key.
**External Keyboard**

You can connect an external keyboard to the 4155C/4156C and use to enter text. Also, you can use the keyboard for other tasks as described in this chapter. In this section, the notation "KeyA + KeyB" means to hold down KeyA and press KeyB.

- **Esc**
  Exits the editor.

- **F1 through F8**
  Primary softkeys. Corresponds to the primary softkeys.

- **Shift+ F1 through F7**
  Secondary softkeys. Corresponds to the secondary softkeys.

- **F9**
  Screen mode. Same as Ctrl+G. Toggles the screen mode as follows:

  ![Screen mode diagram](image)

- **F11**
  Clear to end. Deletes characters from cursor to end of line. Same as Ctrl+Delete

- **Shift+F11**
  Clear line.
  When editor is running, same as F11.
  When editor is not running, deletes characters on the Command Entry line.

- **F12**
  Clear display. Clears the display for IBASIC. When the editor is running, exits from the editor, and clears the display for IBASIC.

- **Print Screen**
  Clear I/O. Stops I/O operation of program.

- **Scroll Lock**
  Stop. Stops program execution after executing the current line. Same as Shift+Pause.

- **Shift+Scroll Lock**
  Reset. Stops program execution immediately.

- **Pause**
  Pause. Same as Ctrl+P. Pauses program execution until Cont is executed or Continue primary softkey is pressed. If the program is modified while paused, Run must be used to restart program execution.

- **Shift+Pause**
  Stop. Stops program execution after executing the current line. Same as Scroll Lock.
Using Instrument BASIC
Keys for IBASIC

Insert
Insert. Same as Alt+H.
When the editor is running, opens a new line before the current line.
When the editor is not running, inserts text at the cursor (press Insert again to end insert mode).

Shift+Insert
Insert. Same as Insert.

Delete
Delete. Deletes character at the cursor.

Shift+Delete
Delete line. Same as Alt+D.
When the editor is running, deletes the current line.
When the editor is not running, deletes character at cursor.

Ctrl+Delete
Clear to end. Deletes characters from cursor to end of line. Same as F11.

Home
Beginning of line. Moves the cursor to beginning of the line. Same as Shift+⇐.

Shift+Home
Page move.
When the editor is running, same as Page Up. Also, same as Shift+↑.
When the editor is not running, jumps to the top of the IBASIC output area. Also, same as Shift+↓.

End
End of line. Moves cursor to end of line. Same as Shift+⇒.

Shift+End
Page move.
When the editor is running, same as Page Down. Also, same as Shift+↓.
When the editor is not running, jumps to the bottom of the IBASIC output area. Same as Shift+↓.

Page Up
Page move.
When the editor is running, moves the cursor one-half display page toward the beginning of the program. Same as Shift+Home. Same as Shift+↑.
When the editor is not running, moves display down one page.
Using Instrument BASIC
Keys for IBASIC

**Shift+Page Up**
Recall.
When the editor is running, displays last deleted line. To enter line into program,
press **Enter**.
When the editor is not running, cycles through the 10 commands that were most
recently entered on the Command Entry line.

**Page Down**
Page move.
When the editor is running, moves the cursor one-half display page toward the end
of the program. Same as **Shift+End**. Same as **Shift+⇓**.
When the editor is not running, moves display up one page.

**Shift+Page Down**
Recall backward.
When the editor is running, same as **Shift+Page Up**.
When the editor is not running, cycles through the 10 commands that were most
recently entered on the Command Entry line in reverse order of **Shift+Page Up**.

**⇑**
Previous line.
When the editor is running, moves cursor up one line.
When the editor is not running, display on the I B A S I C output area moves one line
toward the end.

**Shift+⇑**
Page move.
When the editor is running, same as **Page Up**. Also, same as **Shift+Home**.
When the editor is not running, jumps to the bottom of the I B A S I C output area.
Same as **Shift+End**.

**Alt+⇑**
Recall backward.
When the editor is running, same as **Shift+Page Up**.
When the editor is not running, same as **Shift+Page Down**.

**⇓**
Next line.
When the editor is running, cursor moves down one line.
When editor is not running, display on I B A S I C output area moves one line toward
beginning.
Using Instrument BASIC

Keys for IBASIC

**Shift + ↓**
Page move.
When the editor is running, same as Page Down. Also, same as Shift + End.
When the editor is not running, jumps to the top of the IBASIC output area. Same as Shift + Home.

**Alt + ↓**
Recall. Same as Shift + Page Up.

⇐, ⇒
Move cursor. Moves the cursor one character in indicated direction.

**Shift + ⇐**
Beginning of line. Moves the cursor to beginning of line. Same as Home.

**Shift + ⇒**
End of line. Moves the cursor to end of line. Same as End.

**Backspace**
Backspace.
When the editor is running, deletes the character before cursor.
When the editor is not running, deletes the character before cursor (if mode is insert mode). If mode is not insert move, moves cursor to left by one cursor.

**Alt + D**
Delete line. Same as Shift + Delete.
When the editor is running, deletes the current line.
When the editor is not running, deletes the character at the cursor.

**Alt + I**
Insert line. Same as Insert.
When the editor is running, opens a new line before the current line.
When the editor is not running, inserts text at the cursor (press Insert again to end insert mode).

**Ctrl + U**
Run. Executes the program.

**Ctrl + P**
Pause. Same as Pause. Pauses program execution until CONT is executed or Continue primary softkey is pressed. If the program is modified while paused, RUN must be used to restart program execution.

**Ctrl + G**
Screen mode. Same as F9. Toggles the screen mode as follows:

<table>
<thead>
<tr>
<th>All Instrument</th>
<th>I-BASIC status</th>
<th>All I-BASIC</th>
</tr>
</thead>
</table>

4155C/4156C Specific IBASIC Keywords

The following keywords are not standard IBASIC keywords, or are standard keywords, but with a difference. These keywords are specific to the 4155C/4156C.

EXECUTE Not standard IBASIC keyword. Refer to “ASP-like Commands” in Chapter 5 for details.

ON KEY Standard IBASIC keyword, except the range of key selector is 1 to 7. 1 to 7 of key selector corresponds to secondary softkeys 1 to 7, respectively.

PEN Standard IBASIC keyword, except the range of pen selector is 7. The following table shows the corresponding color for each pen selector.

<table>
<thead>
<tr>
<th>pen selector</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>color defined for Foreground on SYSTEM: COLOR SETUP screen.</td>
</tr>
<tr>
<td>2</td>
<td>color defined for Y1 Axis on SYSTEM: COLOR SETUP screen.</td>
</tr>
<tr>
<td>3</td>
<td>color defined for Y2 Axis on SYSTEM: COLOR SETUP screen.</td>
</tr>
<tr>
<td>4</td>
<td>color defined for Marker/Cursor/Line on SYSTEM: COLOR SETUP screen.</td>
</tr>
<tr>
<td>5</td>
<td>color defined for Active Mkr/Csr/Lne on SYSTEM: COLOR SETUP screen.</td>
</tr>
<tr>
<td>6</td>
<td>color defined for Advisory on SYSTEM: COLOR SETUP screen.</td>
</tr>
<tr>
<td>7</td>
<td>color defined for Title on SYSTEM: COLOR SETUP screen.</td>
</tr>
</tbody>
</table>
Using Instrument BASIC
4155C/4156C Specific IBASIC Keywords

The following IBASIC keywords are not implemented in the 4155C/4156C's Instrument BASIC:

- ALPHA ON/OFF
- AREA
- CLIP
- CONTROL
- DUMP
- EDGE
- FILL
- FRAME
- GESCAPE
- GLOAD
- GRAPHICS
- GRID
- GSTORE
- LINE TYPE
- PLOTTER IS
- POLYGON
- POLYLINE
- RATIO
- RECTANGLE
- SET PEN
- SHOW
- STATUS
- VIEWPORT
- WINDOW
Available I/O Resources for IBASIC

This section provides information about available I/O resources for IBASIC of the 4155C/4156C.

The following I/O resources are available for IBASIC.

- LCD Display
- External keyboard and front-panel keyboard
- GPIB Interface on rear panel
- Internal pseudo GPIB Interface (to control the 4155C/4156C itself)
- Parallel Interface
- Built-in Flexible Disk Drive (no select code)

The following table shows available I/O interfaces and their select codes.

<table>
<thead>
<tr>
<th>Select Code</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LCD</td>
</tr>
<tr>
<td>2</td>
<td>External and front-panel keyboard</td>
</tr>
<tr>
<td>7</td>
<td>GPIB Interface on rear panel</td>
</tr>
<tr>
<td>8</td>
<td>Internal pseudo GPIB Interface</td>
</tr>
<tr>
<td>9</td>
<td>Parallel Interface</td>
</tr>
</tbody>
</table>

LAN interface on the 4155C/4156C rear panel is not available for the built-in IBASIC.
Using Instrument BASIC
Available I/O Resources for IBASIC

**LCD**

IBASIC can display text or graphics on the display of the 4155C/4156C. Following figure shows the display area image on the 4155C/4156C screen.

![Display Area Diagram]

**Text display**

Text can be displayed in the IBASIC output area of "All IBASIC" screen. This area has 23 lines and 58 columns (58 characters in a line), and does not covers the softkey display area.

The x and y coordinate values of the area are as follows:

- lower left corner: \((x,y) = (0,88)\)
- upper right corner: \((x,y) = (522,479)\)

The following table shows the area used to display a character. In this table, Reserved column shows the area captured to display a character. This area covers a character, and includes space between characters. Actual column shows the area for a character only.

<table>
<thead>
<tr>
<th>Text Display Area</th>
<th>Softkey Display Area</th>
<th>Graphics Display Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 479)</td>
<td>(522, 479)</td>
<td>(639, 479)</td>
</tr>
<tr>
<td>(0, 88)</td>
<td>(522, 88)</td>
<td>(639, 0)</td>
</tr>
<tr>
<td>(0, 0)</td>
<td>(522, 0)</td>
<td>(639, 0)</td>
</tr>
</tbody>
</table>
Using Instrument BASIC
Available I/O Resources for IBASIC

<table>
<thead>
<tr>
<th>Reserved</th>
<th>Actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size (dots)</td>
<td>$9 \times 17$</td>
</tr>
<tr>
<td>Lower Left Corner</td>
<td>$(9 \times (N-1), 479 - 17 \times M)$</td>
</tr>
<tr>
<td>Upper Right Corner</td>
<td>$(9 \times N, 479 - 17 \times (M-1))$</td>
</tr>
</tbody>
</table>

where, $N$ and $M$ are integer value ($N=1$ to 58, $M=1$ to 23).

**Graphics display**

In “All IBASIC” screen, you can display a graphical plot.

The x and y coordinate values of this area are as follows:

- lower left corner: $(x,y)=(0,0)$
- upper right corner: $(x,y)=(639,479)$

This area covers the softkey display area.

**GPIB Interfaces**

- Internal pseudo GPIB
  
  By using select code 8, you can control the 4155C/4156C via internal pseudo GPIB interface. The GPIB address of the 4155C/4156C has no meaning, so you can use any address (0 through 30).

- GPIB on rear panel
  
  You can access GPIB interface on rear panel by using select code 7.

**Parallel Interface**

You can use parallel interface on the rear panel for the printer interface. Select code is 9.
Using Instrument BASIC
Available I/O Resources for IBASIC

**Built-in Flexible Disk Drive**

If you specify optional volume specifier when accessing the built-in flexible disk drive, the volume specifier must be ":INTERNAL,4".

**Available diskettes**

You can use the following types of 3.5 inch diskettes:

- **2HD 1.44 MB**
- **2DD 720 KB**

Diskette must be formatted as LIF or the following DOS format:

- 80 tracks/side
- 18 sectors/track (2HD), 9 sectors/track (2DD)
- 512 bytes/sector
Differences from 4155A/4156A Programming

Programming differences between the 4155A/4156A and the 4155C/4156C come from the differences of the screen and the SCPI commands supported by the instruments.

For the most case, you can execute the IBASIC program created for the 4155A/4156A on the 4155C/4156C built-in IBASIC controller. But the following programs should be modified to execute on the 4155C/4156C’s IBASIC controller.

- Programs use both text display and graphics display.
- Programs use the :HCOpy:DESTination command.
- Programs use the :HCOpy:DEVice:LANGuage command.

This section provides the information to modify the program.

For the SCPI commands which the 4155A/4156A does not support but the 4155C/4156C supports, refer to “Differences From 4155B/4156B SCPI Commands” and “Differences From 4155A/4156A SCPI Commands” in Chapter 2.

NOTE

Executing the IBASIC program created for the 4155B/4156B

The IBASIC programs for the 4155B/4156B can be executed on the 4155C/4156C. You do not need to modify the programs.

NOTE

Using 4155C/4156C FLEX Command Set

If you use the 4155C/4156C FLEX command set, you cannot reuse the SCPI programs of the 4155A/4156A, and you need to create new program.

Refer to Chapter 3.
Using Instrument BASIC
Differences from 4155A/4156A Programming

Using Text Display and Graphics Display

By the difference of the screen, the 4155C/4156C display area is different from the
4155A/4156A as shown in the following table. So if you execute the IBASIC
program for the 4155A/4156A on the 4155C/4156C built-in IBASIC controller, the
graphics will lie on the text.

Modify the program by using the following information.

<table>
<thead>
<tr>
<th></th>
<th>4155A/4156A</th>
<th>4155C/4156C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text display a</td>
<td>58 characters/line</td>
<td>58 characters/line</td>
</tr>
<tr>
<td></td>
<td>23 lines</td>
<td>23 lines</td>
</tr>
<tr>
<td></td>
<td>814 dots/character</td>
<td>917 dots/character</td>
</tr>
<tr>
<td>Graphics display b</td>
<td>lower left (0,0)</td>
<td>lower left (0,0)</td>
</tr>
<tr>
<td></td>
<td>upper right (546,399)</td>
<td>upper right (639,479)</td>
</tr>
</tbody>
</table>

a. Text display area does not cover the softkey display area.
b. Graphics display area covers the text display area and the softkey dis-
play area.

Following figure shows the display area image on the 4155C/4156C screen.

---

1-52

The following table shows the area used to display a character. In this table, Reserved column shows the area captured to display a character. This area covers a character, and includes space between characters. Actual column shows the area for a character only.

where, \( N \) and \( M \) are integer value (\( N = 1 \) to 58, \( M = 1 \) to 23).

<table>
<thead>
<tr>
<th></th>
<th>Reserved</th>
<th>Actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size (dots)</td>
<td>( 9 \times 17 )</td>
<td>( 5 \times 10 )</td>
</tr>
<tr>
<td>Lower Left Corner</td>
<td>((9 \times (N-1), 479 -17 \times M))</td>
<td>((9 \times (N-1), 483 -17 \times M))</td>
</tr>
<tr>
<td>Upper Right Corner</td>
<td>((9 \times N, 479 -17 \times (M-1)))</td>
<td>((9 \times N-4, 476 -17 \times (M-1)))</td>
</tr>
</tbody>
</table>

**Using :HCOP:DEST Command**

From differences of the supported interfaces, the command parameters are different from the 4155A/4156A as shown below:

<table>
<thead>
<tr>
<th>Model</th>
<th>Serial</th>
<th>Parallel</th>
<th>GPIB</th>
<th>LAN</th>
<th>file</th>
</tr>
</thead>
<tbody>
<tr>
<td>4155A/4156A</td>
<td>SERial</td>
<td></td>
<td>RDEVice</td>
<td></td>
<td>MMEMory</td>
</tr>
<tr>
<td>4155C/4156C</td>
<td>PARallel</td>
<td>RDEVice</td>
<td>NETn</td>
<td></td>
<td>MMEMory</td>
</tr>
</tbody>
</table>

where, \( n \) is 1, 2, 3 or 4.

The meaning of the parameters is as follows:

SERial   Selects serial interface.
PARallel Selects parallel interface.
RDEVice  Selects GPIB interface.
NETn     Selects a remote printer. \( n = 1, 2, 3 \) or 4.
MMEMory  Not make hardcopy. Outputs to a file in the device specified by the :MMEM:DEST command.
Using Instrument BASIC
Differences from 4155A/4156A Programming

Using :HCOP:DEV:LANG Command

Differences of the supported output format for the print/plot function make the
differences on the command parameters as shown below.

<table>
<thead>
<tr>
<th>Model</th>
<th>PCL</th>
<th>HR PCL a</th>
<th>HP-GL</th>
<th>TIFF</th>
<th>HR TIFF b</th>
</tr>
</thead>
<tbody>
<tr>
<td>4155A/4156A</td>
<td>PCL</td>
<td></td>
<td>HPGL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4155C/4156C</td>
<td>PCL</td>
<td>HRPCI</td>
<td>HPGL</td>
<td>TIFF</td>
<td>HRTIFF</td>
</tr>
</tbody>
</table>

  a. high resolution PCL.
  b. high resolution TIFF.

The meaning of the parameters is as follows:

PCL
Selects PCL format.

HRPCI
Selects high resolution PCL format.

HPGL
Selects HP-GL format.

TIFF
Selects TIFF format. For file output only.

HRTIFF
Selects high resolution TIFF format. For file output only.
2 SCPI Command Programming
SCPI Command Programming

Standard Commands for Programmable Instruments (SCPI) is a universal programming language for electronic test and measurement instruments, and based on IEEE 488.1 and IEEE 488.2.

This chapter describes how to create programs that contain SCPI commands to control Agilent 4155C/56C, and consists of the following sections.

- SCPI Programming Basics
- Getting Started on SCPI Programming
- Measurement Setup
- Auto Analysis Setup
- Measurement Execution
- File Operation
- Data Transfer
- Print/Plot Operation
- Other Programming Tips
- Example for 4145 Users

If you are not familiar with the 4155C/4156C programming, “Getting Started on SCPI Programming” on page 2-8 provides step-by-step tutorials for programming and helps you to understand quickly.

In addition to this chapter, Sample Application Programs Guide Book provides some application examples which are helpful to increase your understanding.

Refer to SCPI Command Reference for the command syntax and description of the SCPI commands available for the 4155C/4156C.
SCPI Programming Basics

This section provides the following basic tasks to control and program the 4155C/4156C:

- “Preparation before Controlling 4155C/56C via GPIB”
- “SCPI Command Hierarchy”
- “To Control 4155C/56C by HP BASIC Programming”

Preparation before Controlling 4155C /56C via GPIB

You can use an external computer or the built-in Instrument BASIC (IBASIC) controller to control the 4155C/4156C via GPIB.

NOTE

Device Clear

The 4155C/4156C requires approx. 0.5 seconds for the GPIB device clear. For the HP BASIC or IBASIC, enter CLEAR command.

Controlling from external computer

You must do as follows before controlling the 4155C/4156C from an external computer:

1. Connect the GPIB interface of external computer to GPIB interface on rear panel of the 4155C/4156C.
2. Set the 4155C/56C is field on the SYSTEM:MISCELLANEOUS screen to NOT SYSTEM CONTROLLER.
3. Enter the GPIB address of your 4155C/4156C into the GPIB ADDRESS field.
Controlling from built-in IBASIC controller

If you use built-in IBASIC controller to control only the 4155C/4156C, you do not need to prepare anything before controlling the 4155C/4156C because built-in IBASIC controller is always connected to the 4155C/4156C via internal GPIB.

However, to control external instruments, do following:

1. Connect the GPIB interface of external instruments to GPIB interface on rear panel of the 4155C/4156C.
2. Set "4155C/56C is" field on the SYSTEM : MISCELLANEOUS screen to SYSTEM CONTROLLER.

To use the 4155C/4156C print/plot function, do following:

1. Connect printer/plotter to the 4155C/4156C.
2. If the printer/plotter interface is GPIB:
   a. Set "4155C/56C is" field to SYSTEM CONTROLLER before executing printing/plotting out.
   b. Enter the GPIB address of printer/plotter into "GPIB ADDRESS" "HARD COPY" field on the SYSTEM : MISCELLANEOUS screen.
3. If you use the remote printer connected to the print server:
   a. Connect the 4155C/4156C to your LAN.
   b. Set the "4155C/56C NETWORK SETUP" table and "NETWORK PRINTER SETUP" table on the SYSTEM : MISCELLANEOUS screen.

To use the network file system on the NFS server, do following:

1. Connect the 4155C/4156C to your LAN.
2. Set the "4155C/56C NETWORK SETUP" table and "NETWORK DRIVE SETUP" table on the SYSTEM : MISCELLANEOUS screen.
SCPI Command Hierarchy

SCPI commands use a hierarchical structure for subsystem commands similar to the file system.

For example, in `:PAGE:MEASURE:SWEEP` command, the hierarchy is as follows:

<table>
<thead>
<tr>
<th>Command</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGE</td>
<td>root</td>
</tr>
<tr>
<td>MEASURE</td>
<td>sub-level 1</td>
</tr>
<tr>
<td>SWEEP</td>
<td>sub-level 2</td>
</tr>
</tbody>
</table>

The colon at the beginning of the command means root. The colons between two command keywords means moving down to a lower level.

**NOTE**

Using a Semicolon to Reduce Typing

A semicolon enables two commands to be sent on the same line.

For example, `:PAGE:MEAS:VAR1:START 0;STOP 5` is the same as the following two commands:

`:PAGE:MEAS:VAR1:START 0`
`:PAGE:MEAS:VAR1:STOP 5`

So, using a semicolon reduces typing and simplifies the program.

A command terminator (such as a `<newline>`) resets the path to root.
SCPI Command Programming
SCPI Programming Basics

To Control 4155C/56C by HP BASIC Programming

1. Assign I/O path for controlling the 4155C/4156C.
   Use ASSIGN command to assign I/O path:
   • Built-in IBASIC
     Specify select code 8. For the GPIB address, you can use any number between 0 to 31. Refer to the following example:
     
     10 ASSIGN @Hp415x TO 800
   • HP BASIC on an external computer
     Specify the select code of the external computer. And specify the GPIB address that you entered into the GPIB ADDRESS field on the SYSTEM: MISCELLANEOUS screen. In the following example, the select code of the external computer is 7 and GPIB address of the 4155C/56C is 17:

     10 ASSIGN @Hp415x TO 717

2. Use OUTPUT command to send commands to the 4155C/4156C.
3. Use ENTER command to get query response from the 4155C/4156C.
Example

The following is the example program to control the 4155C/4156C:

```
10   DIM I3(1:501)
20   !
30   ASSIGN @Hp415x TO 717
40   !
50   OUTPUT @Hp415x;"*RST"
60   !
70   OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES','DISK'"
80   !
90   OUTPUT @Hp415x;":PAGE:SCON:SING"
100  OUTPUT @Hp415x;"*OPC?"
110  ENTER @Hp415x;Complete
120  !
130  OUTPUT @Hp415x;":FORM:DATA ASC"
140  OUTPUT @Hp415x;":DATA? 'I3'"
150  ENTER @Hp415x;I3(*)
160  !
170  END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Resets the 4155C/4156C by sending *RST command.</td>
</tr>
<tr>
<td>70</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>90 to 110</td>
<td>Executes measurement</td>
</tr>
<tr>
<td>130 to 150</td>
<td>Gets the measurement data</td>
</tr>
</tbody>
</table>
Getting Started on SCPI Programming

This section provides step-by-step tutorials for programming to control the 4155C/4156C along with programming examples. In this section, you do SCPI programming by using the built-in IBASIC.

This section consists of the following sections:

• “Creating a Simple Measurement Program”
  This section introduces how to create a measurement program.

• “Programming for Data Extraction”
  This section provides the programming tutorials for data extraction.

• “Complete Example Program for Vth Measurement”
  This section shows complete example program based on the parts described in the other sections.

• “Example Application Setup for Vth Measurement”
  This section describes an example application setup that you should save to the file named VTH.MES on diskette before executing program examples (that use VTH.MES) described in this section.
Creating a Simple Measurement Program

This section introduces how to create a measurement program.

A simple measurement program created by using built-in IBASIC controller is provided as an example and you learn step-by-step how to create a measurement program.

This section consists of the following:

1. Getting a setup file from a diskette and making a measurement
2. Changing the sweep setup parameters
3. Changing the display setup parameters
4. Saving the measurement results to a diskette
5. Printing the measurement results

Before Creating a Program

This section assumes that you have already saved a measurement setup file for Vth measurement to diskette.

Prepare the diskette and save the measurement setup (described in “Example Application Setup for Vth Measurement” on page 2-26) to the file named "VTH.MES" on the diskette.

Before starting this section, do following:

1. If the 4155C/4156C has already been turned on, turn the 4155C/4156C off.
2. Connect a printer to the parallel interface or GPIB interface on the rear panel. You will use the printer at Step 5.
3. Turn the 4155C/4156C on.
4. Set the SYSTEM: PRINT/PLOT SETUP screen and SCREEN DUMP dialog as you want. Or set only the "DESTINATION" and "PAPER" fields on the SYSTEM: PRINT/PLOT SETUP screen.
5. If you use the GPIB printer:
   a. Set "4155C/56C is" field on the SYSTEM: MISCELLANEOUS screen to SYSTEM CONTROLLER.
   b. Enter the GPIB address of printer/plotter into "GPIB ADDRESS" "HARD COPY" field.
Step 1

Getting the Setup File and Making a Measurement

In this step, you can create a program to get a setup file from the diskette and execute a measurement.

1. Press IBA SIC Display key until screen display mode is All IBA SIC mode.
2. Select EDIT softkey, then press Enter key to start the IBA SIC editor.
3. If there is an existing program, save it if necessary.
4. Delete existing program and assign I/O path to control the 4155C/56C.
   Type SCRATCH, then Enter. Or select Scratch primary softkey, then YES secondary softkey to delete the program.
   Existing program is deleted and the following program lines are entered automatically. These lines are for assigning the 4155C/56C control I/O path.

```
10 COM @Hp415x
20 ASSIGN @Hp415x TO 800
30 !
9990 END
```

- line 10 Declares COM so that subprograms can access the I/O path (that is assigned in line 20) for controlling the 4155C/56C. Refer to the Instrument BASIC Users Handbook for details.
- line 20 Assigns the I/O path for controlling the 4155C/56C. 800 means built-in IBA SIC controller.

5. Select OUTPUT @Hp415x secondary softkey.

The following characters are automatically entered:

```
30 OUTPUT @Hp415x;""
```

Do not press Enter yet.

6. Use the help function to find the command for getting a setup file:
   a. Press Help.
   b. Press Get.

   The cursor in help window automatically jumps to the command (:MMEM:LOAD:STAT) for getting a setup file.

7. Press Enter to insert the command into the program line.

Now line 30 is as follows:

```
30 OUTPUT @Hp415x;""MMEM:LOAD:STAT"
```
8. Type in the command parameters as in following example:

30 OUTPUT @Hp415x;"::MMEM:LOAD:STAT 0,'VTH.MES','DISK'"

The following table shows the meaning of the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Dummy parameter (no meaning).</td>
</tr>
<tr>
<td>VTH.MES</td>
<td>File name to be loaded.</td>
</tr>
<tr>
<td>DISK</td>
<td>Source mass storage is diskette.</td>
</tr>
</tbody>
</table>

Then press Enter. Then select the Insert line softkey.

**To Specify Mass Storage Device**

When the 4155C/4156C is in the power on state, the mass storage device is set to the built-in flexible disk drive.

If you specify the device, enter the :MMEM:DEST command before the :MMEM:LOAD:STAT command. The following parameters are available for the :MMEM:DEST command:

- **INTERNAL** Selects the built-in flexible disk drive.
- **NET n** Selects the NFS server. n = 1, 2, 3 or 4.

9. Select OUTPUT @Hp415x secondary softkey.

40 OUTPUT @Hp415x;"_

10. Press Help, then press Single, Append, or Repeat to find the command for executing a measurement.

Relation between the execution keys and commands are shown below:

<table>
<thead>
<tr>
<th>Execution Key</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single</td>
<td>:PAGE:SCON:MEAS:SING</td>
</tr>
<tr>
<td>Append</td>
<td>:PAGE:SCON:MEAS:APP</td>
</tr>
<tr>
<td>Repeat</td>
<td>:PAGE:SCON:MEAS:REP</td>
</tr>
</tbody>
</table>
SCPI Command Programming
Getting Started on SCPI Programming

11. Press **Enter** to insert the found command into the program line.

   10 COM @Hp415x
   20 ASSIGN @Hp415x TO 800
   30 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'VTH.MES','DISK'"
   40 OUTPUT @Hp415x;":PAGE:SCON:MEAS:SING"
   9990 END

   Then press **Enter**

12. Press **End edit** softkey to exit from the editor.

Now you have created a measurement program.

To execute the program, do as follows:

1. Press **IBASIC Display** key until screen display mode is All Instrument or IBASIC Status mode. This allows you to monitor the measurement onGRAPH/LIST: GRAPHICS screen.

2. Press **Run** front-panel key. The measurement program is executed.
Step 2  Changing the Sweep Setup Parameters

Modify measurement program created in previous step so that you can enter new sweep start and stop values while program is running:

1. Press I BASIC Display key until the screen display mode is All I BASIC mode.
2. Select EDIT softkey, then press Enter key to start the I BASIC editor.
3. Insert program lines that allow you to enter the sweep start and stop values from the keyboard during program running.
   a. Move the cursor to program line 30.
   b. Select Insert line primary softkey.
   c. Type the following program lines:
   
   21 !
   22 INPUT "Sweep Start (V)?",Start_v
   23 INPUT "Sweep Stop (V)?",Stop_v
   24 !

4. If you do not know the SCPI commands for changing the sweep start and stop parameters, do as follows:
   a. Press I BASIC Display key until screen display mode is All Instrument mode.
   b. Press Meas to change to MEASURE: SWEEP SETUP screen.
   c. Move the field pointer to the parameter that you want to change.
   d. Press Help key.

   The corresponding command is displayed at the bottom of the help window:
   You need to remember the commands, so that you can enter them in the next step.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:PAGE:M E A S:SW E:VAR 1:ST AR</td>
<td>VAR 1 sweep start</td>
</tr>
<tr>
<td>:PAGE:M E A S:SW E:VAR 1:STOP</td>
<td>VAR 1 sweep stop</td>
</tr>
</tbody>
</table>

   e. Select the EXIT HELP softkey.
SCPI Command Programming
Getting Started on SCPI Programming

5. Press IBASIC Display key until screen display mode is All IBASIC mode. Then, do the following to insert the program lines for changing the sweep start and stop values.
   a. Move the cursor to the program line 40.
   b. Select Insert line primary softkey.
   c. Select the OUTPUT @Hp415x softkey.
   d. Type in the SCPI command. Or you can use the help function to enter the command. For the help function, see “To Use the Help Function” in Chapter 1.

After you finish, the program lines should look as follows:

```
31 OUTPUT @Hp415x;";PAGE:MEAS:SWE:VAR1:STAR ";Start_v
32 OUTPUT @Hp415x;";PAGE:MEAS:SWE:VAR1:STOP ";Stop_v
```

Now the program is as follows:

```
10 COM @Hp415x
20 ASSIGN @Hp415x TO 800
21 !
22 INPUT "Sweep Start (V)?",Start_v
23 INPUT "Sweep Stop (V)?",Stop_v
24 !
30 OUTPUT @Hp415x;";MMEM:LOAD:STAT 0,'VTH.MES','DISK'
31 OUTPUT @Hp415x;";PAGE:MEAS:SWE:VAR1:STAR ";Start_v
32 OUTPUT @Hp415x;";PAGE:MEAS:SWE:VAR1:STOP ";Stop_v
40 OUTPUT @Hp415x;";PAGE:SCON:MEAS:SING"
```

6. Select End edit softkey to exit from the editor.

7. Press IBASIC Display key until screen display mode is IBASIC Status mode.

8. Press Run to execute the program.

9. Sweep Start (V)? is displayed on the display line. Enter the desired sweep start voltage.

10. Sweep Stop (V)? is displayed on the display line. Enter the desired sweep stop voltage.
Step 3

Changing the Display Setup Parameters

In this step, change X-axis range of display setup parameters to match the sweep start and stop values.

1. Press IBASIC Display key until screen display mode is All IBASIC mode.
2. Select EDIT softkey, then press Enter key to start the IBASIC editor.
3. If you do not know the SCPI commands for changing the X-axis parameters, do as follows:
   a. Press IBASIC Display key until screen display mode is All Instrument mode.
   b. Press PAGE CONTROL Display key to change to DISPLAY: DISPLAY SETUP screen.
   c. Move the field pointer to the parameter that you want to change.
   d. Press Help key.
   The corresponding commands are displayed at the bottom of the help window. You need to remember the commands, so that you can enter them in the next step.
   e. Select the EXIT HELP softkey.
4. Press IBASIC Display key until screen display mode is All IBASIC mode. Then, do the following to insert the program lines for changing the X-axis display parameters:
   a. Move the cursor to the program line 40.
   b. Select Insert line primary softkey.
   c. Select the OUTPUT @Hp415x softkey.
   d. Type in the SCPI command. Or you can use the help function to enter the command. For the help function, see “To Use the Help Function” in Chapter 1. After you finish, the program lines should look as follows:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>

5. Repeat steps 1 through 4 for changing the Y-axis range.
SCPI Command Programming
Getting Started on SCPI Programming

Now the program is as follows:

10 COM @Hp415x
20 ASSIGN @Hp415x TO 800
21 !
22 INPUT "Sweep Start (V)?",Start_v
23 INPUT "Sweep Stop (V)?",Stop_v
24 !
30 OUTPUT @Hp415x;";MMEM:LOAD:STAT 0,'VTH.MES','DISK"
31 OUTPUT @Hp415x;";PAGE:MEAS:SWE:VAR1:STAR ";Start_v
32 OUTPUT @Hp415x;";PAGE:MEAS:SWE:VAR1:STOP ";Stop_v
33 OUTPUT @Hp415x;";PAGE:DISP:SET:GRAP:X:MIN ";Start_v
34 OUTPUT @Hp415x;";PAGE:DISP:SET:GRAP:X:MAX ";Stop_v
40 OUTPUT @Hp415x;";PAGE:SCON:MEAS:SING"
9990 END

5. Select End edit softkey to exit from the editor.
6. Press IBA SIC Display key until screen mode is IBA SIC Status mode.
7. Press Run to execute the program.
8. Enter the desired sweep start and stop values as prompted. The minimum and maximum X-axis values of the graph will be the same as these entered values.
Step 4  Saving All Measurement Results to a Diskette

In this step, add program lines that save the setup data and measurement results to
the diskette.

1. Press I B A S I C Display key until screen display mode is A ll I B A S I C mode.
2. Select EDIT softkey, then press E nter key to start the I B A S I C editor.
3. Move the cursor to program line 9990.
4. Select Insert line primary softkey.
5. Insert the following program lines, which wait until the measurement is
completed.

50 OUTPUT @Hp415x;"*OPC?"
60 ENTER @Hp415x;Complete

When measurement is completed, the 4155C/56C returns 1 to the Complete
variable.

6. Insert the following program line, which saves the measurement setup and
results to a file named VTH.DAT:

70 OUTPUT @Hp415x;":MMEM:STOR:TRAC DEF,'VTH.DAT'"

Now the program is as follows:

10 COM @Hp415x
20 ASSIGN @Hp415x TO 800
21 !
22 INPUT "Sweep Start (V)?",Start_v
23 INPUT "Sweep Stop (V)?",Stop_v
24 !
30 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'VTH.MES','DISK'"
31 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STAR ";Start_v
32 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STOP ";Stop_v
33 OUTPUT @Hp415x;":PAGE:DISP:SET:GRAP:X:MIN ";Start_v
34 OUTPUT @Hp415x;":PAGE:DISP:SET:GRAP:X:MAX ";Stop_v
40 OUTPUT @Hp415x;":PAGE:SCON:MEAS:SING"
50 OUTPUT @Hp415x;"*OPC?"
60 ENTER @Hp415x;Complete
70 OUTPUT @Hp415x;":MMEM:STOR:TRAC DEF,'VTH.DAT'"
9990 END

7. Select End edit softkey to exit from the editor.

8. Press I B A S I C Display key until screen display mode changes to I B A S I C Status
mode.

9. Press R un to execute the program.

The measurement setup and results are automatically saved to the diskette after
measurement is performed.
SCPI Command Programming
Getting Started on SCPI Programming

Step 5

Printing the Measurement Results

In this step, add program lines that print the measurement results.

1. Press Display key until screen display mode is All IBASIC mode.

2. Select EDIT softkey, then press Enter key to start the IBASIC editor.

3. Move the cursor to the program line 70.

4. Select Insert line primary softkey.

5. Insert the following program lines, which print a screen dump of the results:

   61 OUTPUT @Hp415x;":HCOP:SDUM"
   62 DISP "Printing"
   63 OUTPUT @Hp415x;"*OPC?"
   64 ENTER @Hp415x;Complete
   65 DISP "Done"

:HCOP immediately initiates the plot or print according to the current setup.

After printing is finished, the 4155C/56C returns 1 to the Complete variable, then "Done" is displayed on the screen.

Now the program is as follows:

   10 COM @Hp415x
   20 ASSIGN @Hp415x TO 800
   21 !
   22 INPUT "Sweep Start (V)?",Start_v
   23 INPUT "Sweep Stop (V)?",Stop_v
   24 !
   30 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'VTH.MES','DISK'"
   31 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STAR ";Start_v
   32 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STOP ";Stop_v
   33 OUTPUT @Hp415x;":PAGE:DISP:SET:GRAP:X:MIN ";Start_v
   34 OUTPUT @Hp415x;":PAGE:DISP:SET:GRAP:X:MAX ";Stop_v
   40 OUTPUT @Hp415x;":PAGE:SCON:MEAS:SING"
   50 OUTPUT @Hp415x;"*OPC?"
   60 ENTER @Hp415x;Complete
   61 OUTPUT @Hp415x;":HCOP:SDUM"
   62 DISP "Printing"
   63 OUTPUT @Hp415x;"*OPC?"
   64 ENTER @Hp415x;Complete
   65 DISP "Done"
   70 OUTPUT @Hp415x;":MMEM:STOR:TRAC DEF,'VTH.DAT'"

9990 END
Programming for Data Extraction

This section provides the following programming tutorials for data extraction:

1. Reading the 4155/56 setup data
2. Reading values of data variables (measurement results)
3. Transferring data into a file

Step 1 Reading 4155/56 Setup Data

To read setup data from the 4155/56 into an IBSAC variable, use the query form of the corresponding setting command. To make the query form of a command, simply add a question mark (?) to the end of the command.

Refer to the following program lines of example program:

60 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STAR?"
70 ENTER @Hp415x;Vd_start
80 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STOP?"
90 ENTER @Hp415x;Vd_stop
100 OUTPUT @Hp415x;":DISP:ALL BAS"
110 CLEAR SCREEN
120 PRINT TABXY(1,1);"Vd START=";Vd_start;"(V)"
130 PRINT TABXY(1,2);"Vd STOP =";Vd_stop;"(V)"

Line 60 This query command tells the 4155C/56C to put the VAR1 start value in its output buffer.

:PAGE:MEAS:SWE:VAR1:STAR is the command for setting the VAR1 start value. By adding ?, the command becomes the query command for reading the VAR1 start value.

Line 70 This gets the start value from the output buffer, then enters it in the Vd_start variable.

Line 80 to 90 These lines tell the 4155C/56 to put VAR1 stop value in its output buffer, then the value is entered into the Vd_stop variable.
Step 2

Reading 4155/56 Measurement Data

To read read-out function values or data variable values (output data, measurement data, and user function values) from the 4155/56 to I BASIC variables, use the :DATA? command.

Refer to the following program lines in the example program:

410 OUTPUT @Hp415x;":PAGE:SCON:MEAS:SING"
420 OUTPUT @Hp415x;":OPC?"
430 ENTER @Hp415x;Complete
440 OUTPUT @Hp415x;":DATA? 'VTH'"
450 ENTER @Hp415x;Vth

Line 410 Execute single measurement.
Line 420 to 430 Wait for measurement completion.
Line 440 Send :DATA? query command to read the result value of user function "VTH".
Line 450 Store the result value into Vth variable.

NOTE

Variable Names

Be aware that data variable names, such as user functions and user variables, are case sensitive. For example, if you set up user function name VTH on the CHANNEL: USER FUNCTION DEFINITION screen, then to read it, you must use :DATA? 'VTH', not :DATA? 'Vth'.
Step 3

**Transferring Specific 4155/56 Data to a File**

To transfer data from the 4155C/56C to a file, do as follows:

1. Create a data file.
2. Open an I/O path for transferring data into the file.
3. Store data into the file.
4. Close the I/O path.

**Create a data file**

You can create three types of data files: DOS, LIF ASCII, or BDAT as follows:

```
CREATE "data_file",1  ! Creates a DOS file.
CREATE ASCII "ascii_file",100  ! Creates a LIF ASCII file.
CREATE BDAT "binary_file",100  ! Creates a BDAT file.
```

DOS files are compatible with MS-DOS, which are easy to transfer to PCs and other computers.

LIF ASCII files are compatible with HP computers that support this file type, so this type is best is you are transferring files among HP computers that support this file type.

BDAT (binary data) files provide more flexibility (can specify both number of records and record length) and faster transfer rate. But BDAT files cannot be interchanged with as many other systems.

The first parameter of each statement specifies the file name to create.

The second parameter specifies number of records to allocate for the file as follows:

**DOS**

Second parameter specifies how many records are to be initially allocated for the file. A DOS file system automatically allocates additional space for the file as new data is written to it, so you can always specify 1 for this parameter.

**LIF ASCII**

Second parameter specifies total number of records to allocate for the file, so you must specify a sufficient number of records. The length of one record is 256 bytes.

For example, the following statement would create a file with 100 records (each record is 256 bytes):

```
CREATE ASCII "File",100
```
SCPI Command Programming
Getting Started on SCPI Programming

**BDAT**

Second parameter specifies total number of records to allocate for the file, so you must specify a sufficient number of records. You can specify a record length by using an optional third parameter (default length is 256 bytes).

For example, the following statement creates a file with 7 records (each record is 128 bytes):

```
CREATE BDAT "B_file",7,128
```

The following statement creates a file with 7 records (each record is 256 bytes):

```
CREATE BDAT "B_file",7
```

Open an I/O path for transferring data into the file

To open an I/O path to the file, assign an I/O path name to the file by using an ASSIGN statement as in the following example:

```
340 INPUT "Enter file name to store data",File$
350 CREATE File$,1
360 ASSIGN @File TO File$;FORMAT ON
```

Line 350 creates a DOS file, then line 360 opens an I/O path to the file.

For DOS and BDAT files, ASSIGN statement can also specify the following:

**FORMAT ON**

ASCII data representations are used. Specify this if you need to transport data between IBASIC and other machines.

**FORMAT OFF**

IBASIC internal data representations are used. Specify this if you need a faster transfer rate and space efficiency.

Store data into the file

To store data into a file, use OUTPUT and ENTER statements as in the following examples:

```
340 INPUT "Enter file name to store data",File$
350 CREATE File$,1
360 ASSIGN @File TO File$;FORMAT ON
:
390 REPEAT
:
440 OUTPUT @Hp415x;":DATA? 'VTH'"
450 ENTER @Hp415x;Vth
460 OUTPUT @File;Vth
:
630 UNTIL Stop$="S" OR Stop$="s"
```
The above program repeats appending Vth variable value to a DOS file in ASCII format.

In addition to numeric data, array data and string data can be stored to a file as in following examples:

- **Array data:**

  ```basic
  1 DIM Vth(1:100)
  340 INPUT "Enter file name to store data",File$
  350 CREATE File$,1
  360 ASSIGN @File TO File$;FORMAT ON
  
  390 FOR I=1 TO 100
  
  440 OUTPUT @Hp415x;":DATA? 'VTH'
  450 ENTER @Hp415x;Vth(I)
  460 NEXT I
  470 OUTPUT @File;Vth(*)
  
  480 END
  
  Close the I/O path
  
  To close an I/O path to a file, ASSIGN the path name to an (asterisk) as in the following example:

  ```basic
  340 INPUT "Enter file name to store data",File$
  350 CREATE File$,1
  360 ASSIGN @File TO File$;FORMAT ON
  
  460 OUTPUT @File;Vth
  
  590 ASSIGN @File TO *
  ```

  In this program, line 590 closes the I/O path that was opened by line 360.

- **String data:**

  ```basic
  10 DIM Data$(10)(1:100)
  20 CREATE "DATAFILE",1
  30 ASSIGN @File TO "DATAFILE";FORMAT ON
  
  40 FOR I=1 TO 100
  
  50 Data$(I)="ABC"
  
  60 NEXT I
  
  70 OUTPUT @File;Data$(*)
  
  71 END
  ```

  Close the I/O path
  
  To close an I/O path to a file, ASSIGN the path name to an (asterisk) as in the following example:

  ```basic
  340 INPUT "Enter file name to store data",File$
  350 CREATE File$,1
  360 ASSIGN @File TO File$;FORMAT ON
  
  460 OUTPUT @File;Vth
  
  590 ASSIGN @File TO *
  ```

  In this program, line 590 closes the I/O path that was opened by line 360.
Complete Example Program for Vth Measurement

The example program shown below uses the measurement setup file described in “Example Application Setup for Vth Measurement” on page 2-26. This is a complete example program based on the parts described in the previous sections.

```plaintext
10 COM @Hp415x
20 ASSIGN @Hp415x TO 800
30 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'VTH.MES','DISK''
40 !
50 ! Read and Disp. Measurement Conditions
60 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STAR?"
70 ENTER @Hp415x;Vd_start
80 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STOP?"
90 ENTER @Hp415x;Vd_stop
100 OUTPUT @Hp415x;":DISP:ALL BAS"
110 CLEAR SCREEN
120 PRINT TABXY(1,1);"Vd START=";Vd_start;"(V)"
130 PRINT TABXY(1,2);"Vd STOP =";Vd_stop;"(V)"
140 !
150 ! Parameter Change
160 Change$="n"
170 Change: !
180 INPUT "Change these parameters? (y/n default=n)",Change$
190 SELECT Change$
200 CASE "Y","y"
210 INPUT "New Vd START (V)?",Vd_start
220 INPUT "New Vd STOP (V)?",Vd_stop
230 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STAR ";Vd_start
240 OUTPUT @Hp415x;":PAGE:MEAS:SWE:VAR1:STOP ";Vd_stop
250 PRINT TABXY(1,1);"Vd START=";Vd_start;"(V)"
260 PRINT TABXY(1,2);"Vd STOP =";Vd_stop;"(V)"
270 CASE "N","n"
280 GOTO Store_file
290 CASE ELSE
300 GOTO Change
310 END SELECT
320 !
330 Store_file: !
340 INPUT "Enter file name to store data",File$
350 CREATE File$,1
360 ASSIGN @File TO File$;FORMAT ON
370 !
380 No_of_data=0
390 REPEAT
400 OUTPUT @Hp415x;":DISP:ALL BST"
410 OUTPUT @Hp415x;":PAGE:SCON:MEAS:SING"
420 OUTPUT @Hp415x;"*OPC?"
430 ENTER @Hp415x;Complete
440 OUTPUT @Hp415x;":DATA? 'VTH'
450 ENTER @Hp415x;Vth
460 OUTPUT @File;Vth
470 No_of_data=No_of_data+1
480 OUTPUT @Hp415x;":DISP:ALL BASIC"
490 PRINT TABXY(1,10);"Last measured Vth =";Vth;"(V)"
500 PRINT TABXY(1,11);"Total number of die tested=";No_of_
```
510 Stop_query:INPUT "Continue to next die or Stop test? (c/s)"
,Stop$
520   SELECT Stop$
530   CASE "C","c"
540     DISP "Move to the next die, then press [Continue]"
550     PAUSE
560     DISP ""
570   CASE "S","s"
580     PRINT TABXY(1,24);"Test Stopped!!"
590     ASSIGN @File TO *
600   CASE ELSE
610     GOTO Stop_query
620   END SELECT
630   UNTIL Stop$="S" OR Stop$="s"
640   END
Example Application Setup for Vth Measurement

This section describes an example application setup that you should save to the file named `VTH.MES` on diskette before executing program examples (that use `VTH.MES`) described previously in this section.

A frequently used method of measuring Vth is to synchronously sweep the exact same voltage to gate and drain, and measure the characteristics in the saturation region.

The theoretical value of drain current in the saturation region is calculated as follows:

$$I_d = \beta \times (V_g - V_{th})^2$$

Where $\beta$ is the gain factor, which is $-1/2 \times (\mu \varepsilon_{ox} W/L) \times t_{ox}$. Therefore, if you take the square root of both sides of the equation:

$$\sqrt{I_d} = \sqrt{\beta} \times (V_g - V_{th})$$

$\sqrt{I_d}$ is proportional to $V_g$, and the slope is $\sqrt{\beta}$. At the point where $\sqrt{I_d}$ is equal 0, $V_{th}$ is equal to $V_g$. So, to know $V_{th}$, we need to find that point.

The measurement conditions are explained below:

**CHANNELS: CHANNEL DEFINITION**

- **MEASUREMENT MODE**

  `<Sweep>`

- **CHANNELS**

<table>
<thead>
<tr>
<th>UNIT</th>
<th>VNAME</th>
<th>INAME</th>
<th>MODE</th>
<th>FCTN</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMU1</td>
<td>Vd</td>
<td>Id</td>
<td>V</td>
<td>VAR1'</td>
</tr>
<tr>
<td>SMU2</td>
<td>Vg</td>
<td>Ig</td>
<td>V</td>
<td>VAR1</td>
</tr>
<tr>
<td>SMU3</td>
<td>Vs</td>
<td>Is</td>
<td>COMMON</td>
<td>CONST</td>
</tr>
<tr>
<td>SMU4</td>
<td>Vsb</td>
<td>Isb</td>
<td>V</td>
<td>CONST</td>
</tr>
</tbody>
</table>


MEASURE: SWEEP SETUP
This is the sweep source setup for the Id-Vg characteristics measurement.

• VARIABLE

<table>
<thead>
<tr>
<th></th>
<th>VAR1</th>
<th></th>
<th>VAR1'</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT</td>
<td>SMU2</td>
<td>UNIT</td>
<td>SMU1</td>
</tr>
<tr>
<td>NAME</td>
<td>Vg</td>
<td>NAME</td>
<td>Vd</td>
</tr>
<tr>
<td>SWEEP MODE</td>
<td>SINGLE</td>
<td>OFFSET</td>
<td>0.0000 V</td>
</tr>
<tr>
<td>LIN/LOG</td>
<td>LINEAR</td>
<td>RATIO</td>
<td>1.000</td>
</tr>
<tr>
<td>START</td>
<td>0.0000 V</td>
<td>COMPLIANCE</td>
<td>100.00 mA</td>
</tr>
<tr>
<td>STOP</td>
<td>5.000 V</td>
<td>POWER COMP</td>
<td>OFF</td>
</tr>
<tr>
<td>STEP</td>
<td>100.0 mV</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO OF STEP</td>
<td>51</td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMPLIANCE</td>
<td>100.0 mA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>POWER COMP</td>
<td>OFF</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

• CONSTANT

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT</td>
<td>SMU4</td>
</tr>
<tr>
<td>NAME</td>
<td>Vsb</td>
</tr>
<tr>
<td>MODE</td>
<td>V</td>
</tr>
<tr>
<td>SOURCE</td>
<td>0.0000 V</td>
</tr>
<tr>
<td>COMPLIANCE</td>
<td>100.00 mA</td>
</tr>
</tbody>
</table>
SCPI Command Programming
Getting Started on SCPI Programming

CHANNELS: USER FUNCTION DEFINITION
The following setup is necessary to calculate \( SQRTId \) (square root of \( Id \)), and
\( GRAD \) (differential coefficient of \( SQRTId \)) versus \( Vg \) automatically. \( VTH \) and
\( BETA \) are defined to extract \( Vth \) and \( \beta \) automatically by using the Read Out
Functions and the Auto Analysis Functions. \( VTH \) is \( @L1X \) (X-intercept of line 1)
and \( BETA \) is \( @L1G^2 \) (slope of line 1 to second power). Line 1 is drawn according
to the definitions of the DISPLAY: ANALYSIS SETUP screen.

- USER FUNCTION

<table>
<thead>
<tr>
<th>NAME</th>
<th>UNIT</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQRTId</td>
<td></td>
<td>SQRT(Id)</td>
</tr>
<tr>
<td>GRAD</td>
<td></td>
<td>DIFF(SQRTId,Vg)</td>
</tr>
<tr>
<td>VTH</td>
<td>V</td>
<td>@L1X</td>
</tr>
<tr>
<td>BETA</td>
<td></td>
<td>@L1G^2</td>
</tr>
</tbody>
</table>
DISPLAY: DISPLAY SETUP
The following setup is to plot two curves: SQRTId versus Vg, and GRAD versus Vg. And VTH and BETA will be displayed in the data variables display area.

- GRAPHICS

<table>
<thead>
<tr>
<th></th>
<th>X-axis</th>
<th>Y1-axis</th>
<th>Y2-axis</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>Vg</td>
<td>SQRTId</td>
<td>GRAD</td>
</tr>
<tr>
<td>SCALE</td>
<td>LINEAR</td>
<td>LINEAR</td>
<td>LINEAR</td>
</tr>
<tr>
<td>MIN</td>
<td>0 V</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>MAX</td>
<td>5 V</td>
<td>200 m</td>
<td>80 m</td>
</tr>
</tbody>
</table>

- GRID

  ON

- LINE PARAMETER

  ON

- DATA VARIABLES

  VTH
  BETA

DISPLAY: ANALYSIS SETUP
The Auto Analysis Functions are defined on DISPLAY: ANALYSIS SETUP screen.
A tangent line (line 1) is drawn to "SQRTId versus Vg" curve (Y1) at point where GRAD is maximum. VTH is the X-intercept of this line. Also, the marker is moved to point where GRAD is maximum.

* LINE1   TANGENT  line on  Y1  at a point where
GRAD = MAX(GRAD)

If you execute a single measurement, the two curves are drawn. Right after the measurement, a tangent line is drawn as specified in DISPLAY: ANALYSIS SETUP screen, and resulting VTH and BETA values are displayed.
Programming: Measurement Setup

To set up a measurement, you can use SCPI commands to set the setup screens of the 4155C/4156C the same way that you can by interactive operation.

Basically, there are the following three ways to set up a measurement via SCPI programming:

• Load the measurement setup data from diskette, NFS server or internal memory.
  
  Load the measurement setup data by SCPI programming. The data was previously defined and stored to the mass storage memory interactively or by SCPI programming.

• Load the measurement setup data, then change some of the settings.
  
  Load the measurement setup data from the mass storage memory, then change desired settings by SCPI programming.

• Set all settings.
  
  Set all settings for measurement setup by SCPI programming.

This section describes the following tasks:

• “To Set or Change 4155/56 Setup Data Values”

• “To Read 4155/56 Setup Data Values”

To load previously defined measurement setup data, refer to “Programming: File Operation” on page 2-47.
To Set or Change 4155/56 Setup Data Values

Send :PAGE subsystem commands that correspond to the setup data values that you want to change or set.

There is a command subsystem for each setup screen as shown in the following table. Each command subsystem has commands for setting the setup data of the corresponding setup screen.

<table>
<thead>
<tr>
<th>Setup Screen</th>
<th>Command Subsystem</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANNELS: CHANNEL DEFINITION</td>
<td>:PAGE:CHANnels[:CDEFinition]</td>
</tr>
<tr>
<td>CHANNELS: USER FUNCTION DEFINITION</td>
<td>:PAGE:CHANnels:UFUNction</td>
</tr>
<tr>
<td>CHANNELS: USER VARIABLE DEFINITION</td>
<td>:PAGE:CHANnels:UV ARiable</td>
</tr>
<tr>
<td>MEASURE: SWEEP SETUP</td>
<td>:PAGE:MEASURE[Sweep]</td>
</tr>
<tr>
<td>MEASURE: SAMPLING SETUP</td>
<td>:PAGE:MEASURE[SAMpling]</td>
</tr>
<tr>
<td>MEASURE: PGU SETUP</td>
<td>:PAGE:MEASURE[PGUSetup]</td>
</tr>
<tr>
<td>MEASURE: MEASURE SETUP</td>
<td>:PAGE:MEASURE[MSETup]</td>
</tr>
<tr>
<td>MEASURE: OUTPUT SEQUENCE</td>
<td>:PAGE:MEASURE[OSEQUence]</td>
</tr>
<tr>
<td>DISPLAY: DISPLAY SETUP</td>
<td>:PAGE:DISPLAY[SETup]</td>
</tr>
<tr>
<td>DISPLAY: ANALYSIS SETUP</td>
<td>:PAGE:DISPLAY[ANALysis]</td>
</tr>
<tr>
<td>STRESS: CHANNEL DEFINITION</td>
<td>:PAGE:STRESS[CDEFinition]</td>
</tr>
<tr>
<td>STRESS: STRESS SETUP</td>
<td>:PAGE:STRESS[SETup]</td>
</tr>
</tbody>
</table>
SCPI Command Programming
Programming: Measurement Setup

Example

To load measurement setup data, then change the sweep start and stop values:

10  ASSIGN @Hp415x TO 800
20  !
30  OUTPUT @Hp415x;":MMEM:DEST INT"
40  OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES','DISK'"
50  !
60  Swp_start=1
70  Swp_stop=10
80  !
90  OUTPUT @Hp415x;":PAGE:MEAS:VAR1:STAR ";Swp_start
100 OUTPUT @Hp415x;":PAGE:MEAS:VAR1:STOP ";Swp_stop
110 !
120 END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>40</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>90</td>
<td>Changes start value of VAR1.</td>
</tr>
<tr>
<td>100</td>
<td>Changes stop value of VAR1.</td>
</tr>
</tbody>
</table>
To Read 4155/56 Setup Data Values

To read setup data from the 4155/56 into an IBASIC variable, do as follows:

Send :PAGE subsystem query command that corresponds to setup data that you want to read.

Example

To load measurement setup data, then read the sweep start and stop values:

```
10 ASSIGN @Hp415x TO 717
20 !
30 OUTPUT @Hp415x;":MMEM:DEST INT"
40 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES','DISK'"
50 !
60 OUTPUT @Hp415x;":PAGE:MEAS:VAR1:STAR?"
70 ENTER @Hp415x;Swp_start
80 OUTPUT @Hp415x;":PAGE:MEAS:VAR1:STOP?"
90 ENTER @Hp415x;Swp_stop
100 !
110 PRINT "Sweep-start=";Swp_start,"Sweep-stop=";Swp_stop
120 !
130 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>40</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>60 to 70</td>
<td>Reads start value of VAR1.</td>
</tr>
<tr>
<td>80 to 90</td>
<td>Reads stop value of VAR1.</td>
</tr>
</tbody>
</table>
Programming: Auto Analysis Setup

To set up the auto analysis function, use the :PAGE:DISP:ANAL subsystem commands:

- To draw a normal line:
  use the :PAGE:DISP:ANAL:LINE1|2:POIN commands.

- To draw a gradient line:
  use the :PAGE:DISP:ANAL:LINE1|2:POIN commands and the :

- To draw a tangent line:
  use the :PAGE:DISP:ANAL:LINE1|2:TANG commands.

- To draw a regression line:
  use the :PAGE:DISP:ANAL:LINE1|2:POIN commands.

- To display a marker:
  use the :PAGE:DISP:ANAL:MARK commands.

For example, the following program defines the auto analysis condition to draw
LINE1 between (0,0) and (5,0.001) for Y1 axis:

```
10   ASSIGN @Hp415x TO 717
20
30   OUTPUT @Hp415x;" :PAGE:DISP:ANAL:LINE1 :MODE NORM"
40   OUTPUT @Hp415x;" :PAGE:DISP:ANAL:YAX Y1"
50
60   OUTPUT @Hp415x;" :PAGE:DISP:ANAL:LINE1 :POIN1 :MODE XY"
70   OUTPUT @Hp415x;" :PAGE:DISP:ANAL:LINE1 :POIN1 :X '0'
80   OUTPUT @Hp415x;" :PAGE:DISP:ANAL:LINE1 :POIN1 :Y '0'"
90
100  OUTPUT @Hp415x;" :PAGE:DISP:ANAL:LINE1 :POIN2 :MODE XY"
110  OUTPUT @Hp415x;" :PAGE:DISP:ANAL:LINE1 :POIN2 :X '5'
120  OUTPUT @Hp415x;" :PAGE:DISP:ANAL:LINE1 :POIN2 :Y '1m'"
130
140  END
```

Figure 2-1 and Figure 2-2 shows the commands corresponding to the auto analysis
setup fields. The figures are just example screen image.
**Programming: Auto Analysis Setup**

**Figure 2-1 Normal Line and Regression Line**

Select Line Mode with softkey or rotary knob.

1. :DISP:ANAL:LINE1:POIN1:MODE
2. :DISP:ANAL:LINE1:POIN1:X
3. :DISP:ANAL:LINE1:POIN1:Y
4. :DISP:ANAL:LINE1:POIN2:MODE
5. :DISP:ANAL:LINE1:POIN2:POS
7. :DISP:ANAL:LINE1:POIN2:AFT
8. :DISP:ANAL:LINE2:POIN1:MODE

---

Tangent Line, Gradient Line, and Marker

**Figure 2-2**

Select Line Mode with softkey or rotary knob.

```plaintext
1  :DISP:ANAL:LINE1:MODE
2  :DISP:ANAL:LINE1:YAX
3  :DISP:ANAL:LINE1:TANG:POS
4  :DISP:ANAL:LINE1:TANG:AFT:STAT
5  :DISP:ANAL:LINE1:TANG:AFT
6  :DISP:ANAL:LINE2:MODE
7  :DISP:ANAL:LINE2:YAX
8  :DISP:ANAL:LINE2:POIN1:MODE
10 :DISP:ANAL:LINE2:POIN1:Y
11 :DISP:ANAL:LINE2:GRAD
12 :DISP:ANAL:MARK:POS
14 :DISP:ANAL:MARK:AFT
15 :DISP:ANAL:INT
```

1. LINE1: TANGENT line on [Y1] at a point where
   \( y = y \)
   \( [AFTER] y = y \)
2. LINE2: GRAD line on [Y1] at a point \( AT \)
   \( X: y \)
   \( Y: y \)
   Gradient: \( y \)
3. MARKER: At a point where
   \( y = y \)
   \( [AFTER] y = y \)
4. Interpolate: ON

**SCPI Command Programming**

Programming: Auto Analysis Setup
Programming: Measurement Execution

To execute a measurement, you can use :PAGE:SCON trol subsystem commands. This section describes the following tasks:

- “To Execute a Measurement”
- “To Force Stress”
- “To Start the Knob Sweep Function”
- “To Control Standby Units”
- “To Control the E5250A Low Leakage Switch Mainframe”
To Execute a Measurement

Send :PAGE:SCONtrl[:MEASurement]:SINGLE command to the 4155C/4156C.

- If you would like to repeat measurements, send :PAGE:SCONtrl[:MEASurement]:REPeat command instead of :PAGE:SCONtrl[:MEASurement]:SINGLE command.
- If you would like to append measurement, send :PAGE:SCONtrl[:MEASurement]:APPend command instead of :PAGE:SCONtrl[:MEASurement]:SINGLE command.

Example 1

To execute a measurement after loading the measurement setup data:

```
10 ASSIGN @Hp415x TO 717
20 
30 OUTPUT @Hp415x;"::MMEM:DEST INT"
40 OUTPUT @Hp415x;"::MMEM:LOAD:STAT 0,'SWP.MES'"
50 OUTPUT @Hp415x;"::PAGE:SCON:SING"
60 
70 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>40</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>50</td>
<td>Executes measurement.</td>
</tr>
</tbody>
</table>
Example 2

To load two measurement setups from diskette and store them into internal memory, then execute the measurements sequentially:

```plaintext
10 ASSIGN @Hp415x TO 717
20 !
30 OUTPUT @Hp415x;":MMEM:DEST INT"
40 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'MEAS1.MES','DISK'
50 OUTPUT @Hp415x;":MMEM:STOR:STAT 0,'MEM1.MES','MEMORY'
60 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'MEAS2.MES','DISK'
70 OUTPUT @Hp415x;":MMEM:STOR:STAT 0,'MEM2.MES','MEMORY'
80 !
90 FOR I=1 TO 5
100 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'MEM1.MES','MEMORY'
110 OUTPUT @Hp415x;":PAGE:SCON:SING"
120 OUTPUT @Hp415x;"*OPC?"
130 ENTER @Hp415x;Complete
140 DISP "Analyze manually then press [Continue]"
150 PAUSE
160 !
170 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'MEM2.MES','MEMORY'
180 OUTPUT @Hp415x;":PAGE:SCON:SING"
190 OUTPUT @Hp415x;"*OPC?"
200 ENTER @Hp415x;Complete
210 DISP "Analyze manually and then press [Continue]"
220 PAUSE
230 !
240 IF I<5 THEN
250 DISP "Move to the next TEG and then press [Continue]"
260 PAUSE
270 END IF
280 !
290 NEXT I
300 !
310 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>40 to 70</td>
<td>Loads two measurement setups from diskette, then stores them into internal memory.</td>
</tr>
<tr>
<td>100 to 130</td>
<td>Executes first measurement, then waits for measurement completion.</td>
</tr>
<tr>
<td>170 to 200</td>
<td>Executes second measurement, then waits for measurement completion.</td>
</tr>
</tbody>
</table>
SCPI Command Programming
Programming: Measurement Execution

To Force Stress

Send :PAGE:SCONtrol:STRes[s][:STARt] command to the 4155C/4156C.

Example 1

To force stress after loading the stress setup data:

```
10 ASSIGN @Hp415x TO 717
20 !
30 OUTPUT @Hp415x;":\MMEM:DEST INT"
40 OUTPUT @Hp415x;":\MMEM:LOAD:STAT 0,'STRS.STR'"
50 OUTPUT @Hp415x;":\PAGE:SCON:STR"
60 !
70 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>40</td>
<td>Loads stress setup data from diskette file STRS.STR.</td>
</tr>
<tr>
<td>50</td>
<td>Executes stress forcing.</td>
</tr>
</tbody>
</table>
Example 2

To force stress, then execute sweep measurement:

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>40</td>
<td>Loads stress setup data from diskette file STRS.STR.</td>
</tr>
<tr>
<td>50</td>
<td>Executes stress forcing.</td>
</tr>
<tr>
<td>60 to 70</td>
<td>Waits until stress forcing is completed.</td>
</tr>
<tr>
<td>90</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>100</td>
<td>Executes sweep measurement.</td>
</tr>
</tbody>
</table>
To Start the Knob Sweep Function

Send :PAGE:SCONtrol:KSWeep[:START] command

Example

To start the knob sweep function:

```
10  ASSIGN @Hp415x TO 717
20  
30  OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'MEM1.MES','MEMORY"'
40  OUTPUT @Hp415x;":PAGE:SCON:KSW"
50  
60  END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Loads sweep setup data from internal memory file MEM1.</td>
</tr>
<tr>
<td>40</td>
<td>Starts knob sweep function.</td>
</tr>
</tbody>
</table>
To Control Standby Units

- To change the standby units from the idle state to the standby state:
  
  Send :PAGE:SControl:STANdby ON.

  You cannot change which units are standby units after you execute this command. Standby units are units for which STBY is set to ON in the CHANNELS: CHANNELS DEFINITION screen.

- To change the standby units from the standby state to the idle state.
  
  Send :PAGE:SControl:STANdby OFF to stop standby units.

Example

To set standby units to standby state (so standby value will be output before and after measurements), then after final measurement, change standby units to idle state:

10  ASSGN @Hp415x TO 717
20
30  OUTPUT @Hp415x;"::MMEM:DEST INT"
40  OUTPUT @Hp415x;"::MMEM:LOAD:STAT 0,'SWP1.MES','DISK'"
50  OUTPUT @Hp415x;"::PAGE:SCon:SCon STAN ON"
60  OUTPUT @Hp415x;"::PAGE:SCon:SING"
70  OUTPUT @Hp415x;"::OPC?"
80  ENTER @Hp415x;Complete
90  OUTPUT @Hp415x;"::MMEM:STOR:TRAC DEF,'MEAS1.DAT','DISK'"
100
110  OUTPUT @Hp415x;"::MMEM:LOAD:STAT 0,'SWP2.MES','DISK'"
120  OUTPUT @Hp415x;"::PAGE:SCon:SING"
130  OUTPUT @Hp415x;"::OPC?"
140  ENTER @Hp415x;Complete
150  OUTPUT @Hp415x;"::PAGE:SCon:STAN OFF"
160  OUTPUT @Hp415x;"::MMEM:STOR:TRAC DEF,'MEAS2.DAT','DISK'"
170  END
### SCPI Command Programming

**Programming: Measurement Execution**

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>40</td>
<td>Loads measurement setup data from diskette file <code>SWP1.MES</code>.</td>
</tr>
<tr>
<td>50</td>
<td>The standby units specified in setup data start to output the standby value.</td>
</tr>
<tr>
<td>60</td>
<td>Executes measurement.</td>
</tr>
<tr>
<td>70 to 80</td>
<td>Waits for completion of measurement. After measurement, standby units output the standby value.</td>
</tr>
<tr>
<td>90</td>
<td>Stores measurement data onto a diskette.</td>
</tr>
<tr>
<td>110</td>
<td>Loads another measurement setup data from diskette file <code>SWP2.MES</code>. This setup data cannot change which units are the standby units.</td>
</tr>
<tr>
<td>120</td>
<td>Executes measurement.</td>
</tr>
<tr>
<td>130 to 140</td>
<td>Waits for completion of measurement. After measurement, standby units output the standby value.</td>
</tr>
<tr>
<td>150</td>
<td>Standby units stop standby output and change to idle state.</td>
</tr>
<tr>
<td>160</td>
<td>Stores measurement data onto a diskette.</td>
</tr>
</tbody>
</table>
To Control the E5250A Low Leakage Switch Mainframe

The 4155C/4156C can control the E5250A/E5252A switching matrix using the built-in IBASIC controller. To control the E5250A, connect the E5250A to the 4155C/4156C via GPIB, set the 4155C/4156C to SYSTEM CONTROLLER, and execute the following commands:

- **:PAGE:CHANnels:MATrix:GPIB:ADDRess address**
  
  This command sets the GPIB address of the E5250A.

- **:PAGE:CHANnels:MATrix:CONTROL onoff**
  
  This command enables or disables the remote control of the E5250A.

- **:MMEM:LOAD:STATe 0,’filename.MAT’,’DISK’**
  
  This command loads the E5250A setup file, and sends the setup to the E5250A.

- **:PAGE:CHANnels:MATrix:CONNect**
  
  This command sends the setup to the E5250A.

**Example**

Before executing the following example, set the GPIB address switch of the E5250A to 22.

```
10 ASSIGN @Hp415x TO 800
20 !
30 OUTPUT @Hp415x;":MMEM:DEST INT"
40 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,’SWP1.MES’,’DISK’"
50 !
60 OUTPUT @Hp415x;":PAGE:CHAN:MAT:GPIB:ADDR 22"
70 OUTPUT @Hp415x;":PAGE:CHAN:MAT:CONTROL ON"
80 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,’CONN1.MAT’,’DISK’"
90 DISP "MODIFY E5250A SETUP. THEN PRESS Continue."
100 PAUSE
110 DISP ""
120 OUTPUT @Hp415x;":PAGE:CHAN:MAT:CONN"
130 !
140 OUTPUT @Hp415x;":PAGE:SCON:SING"
150 OUTPUT @Hp415x;""OPC?"
160 ENTER @Hp415x;Complete
170 OUTPUT @Hp415x;":MMEM:STOR:TRAC DEF,’RES1.DAT’,’DISK’"
180 END
```
SCPI Command Programming
Programming: Measurement Execution

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns an I/O path for controlling the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the built-in flexible disk drive as the mass storage device.</td>
</tr>
<tr>
<td>40</td>
<td>Loads the measurement setup data from the diskette file SWP1.MES.</td>
</tr>
<tr>
<td>60</td>
<td>Sets the E5250A GPIB address to 22.</td>
</tr>
<tr>
<td>70</td>
<td>Sets the E5250A control mode to ON.</td>
</tr>
<tr>
<td>80</td>
<td>Loads the E5250A setup data from the diskette file CONN1.MAT, and sends the setup to the E5250A.</td>
</tr>
<tr>
<td>90 to 100</td>
<td>Displays comments and pauses until the Continue key is pressed.</td>
</tr>
<tr>
<td>120</td>
<td>Applies the matrix connection information to the E5250A.</td>
</tr>
<tr>
<td>140</td>
<td>Executes a single measurement.</td>
</tr>
<tr>
<td>150 to 160</td>
<td>Waits for the completion of the measurement.</td>
</tr>
<tr>
<td>170</td>
<td>Stores the measurement data on a diskette as the file name RES1.DAT.</td>
</tr>
</tbody>
</table>
Programming: File Operation

This section describes how to use SCPI commands to store data to or load data from an internal memory, a diskette or the file system on the NFS server.

This section covers the following basic file operations:

• “To Store Setup Data”
• “To Store Measurement Data”
• “To Load Setup Data”
• “To Load Measurement Data”
• “To Store Spreadsheet Data”

Using NFS Server

If you use NFS server, you need to connect the 4155C/4156C to your LAN, and enter the following SCPI commands or set the following entry fields on the SYSTEM: MISCELLANEOUS screen before executing the file operation:

<table>
<thead>
<tr>
<th>SCPI Command a</th>
<th>SYSTEM: MISCELLANEOUS Screen</th>
</tr>
</thead>
<tbody>
<tr>
<td>:SYST:COMM::NET::SELF::NAME</td>
<td>NETWORK SETUP table HOST NAME</td>
</tr>
<tr>
<td>:SYST:COMM::NET::SELF::IPAD</td>
<td>NETWORK SETUP table IP ADDRESS</td>
</tr>
<tr>
<td>:SYST:COMM::NET::SELF::SNET</td>
<td>NETWORK SETUP table SUBNET MASK</td>
</tr>
<tr>
<td>:SYST:COMM::NET::SELF::GATE</td>
<td>NETWORK SETUP table GATEWAY</td>
</tr>
<tr>
<td>:SYST:COMM::NET::SELF::USER</td>
<td>NETWORK SETUP table USER ID</td>
</tr>
<tr>
<td>:SYST:COMM::NET::SELF::GROU</td>
<td>NETWORK SETUP table GROUP ID</td>
</tr>
<tr>
<td>:SYST:COMM::NET::FILE::NET::NAME</td>
<td>NETWORK DRIVE SETUP table LABEL</td>
</tr>
<tr>
<td>:SYST:COMM::NET::FILE::NET::IPAD</td>
<td>NETWORK DRIVE SETUP table IP ADDRESS</td>
</tr>
<tr>
<td>:SYST:COMM::NET::FILE::NET::DIR</td>
<td>NETWORK DRIVE SETUP table DIRECTORY</td>
</tr>
<tr>
<td>:SYST:COMM::NET::FILE::NET::SET</td>
<td>(same as selecting ADD softkey)</td>
</tr>
</tbody>
</table>

a. For details of the SCPI commands, refer to SCPI Command Reference.

To connect the 4155C/4156C to your LAN, refer to User’s Guide: General Information.
To Store Setup Data

1. Send :MMEMory:DESTination command to the 4155C/4156C to specify the mass storage device.

Then specify the command parameter:

**INT** Selects the built-in flexible disk drive.

**NETn** Selects the NFS server. \( n = 1, 2, 3 \) or 4.

2. Send :MMEMory:STORe:STATe command to the 4155C/4156C.

   a. Specify the first parameter to be 0. This parameter has no meaning for the 4155C/4156C, but is necessary for SCPI compatibility.

   b. Specify the second parameter:

      • For diskette or NFS server:
        File name with extension: .MES for measurement setup data or .STR for stress setup data.

      • For internal memory:
        Internal memory name (MEM1, MEM2, MEM3, or MEM4) with extension:
        .MES for measurement setup data or .STR for stress setup data.

   c. Specify the third parameter:

      • For diskette or NFS server: DISK (default)

      • For internal memory: MEMORY

Example

To store measurement setup data to a diskette file:

```
  !
20  ASSIGN @Hp415x TO 717
30  !
40  OUTPUT @Hp415x;":MMEM:DEST INT"
50  OUTPUT @Hp415x;":MMEM:STOR:STAT 0,'SWP.MES','DISK'"
60  !
70  END
```

<table>
<thead>
<tr>
<th>LineNumber</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>50</td>
<td>Stores measurement setup data to diskette file SWP.MES.</td>
</tr>
</tbody>
</table>

To store setup data to a diskette file:
To Store Measurement Data

1. Send :MMEMory:DESTination command to the 4155C/4156C to specify the mass storage device.

   Specify the command parameter:
   - INT       Selects the built-in flexible disk drive.
   - NETn      Selects the NFS server. n = 1, 2, 3 or 4.

2. Send :MMEMory:STORe:TRACe command to the 4155C/4156C.

   a. Specify the first parameter to be DEFAULT. This parameter has no meaning for the 4155C/4156C, but is necessary for SCPI compatibility.

   b. Specify the second parameter:
      - For diskette or NFS server: File name with extension .DAT
      - For internal memory: Internal memory name (MEM1, MEM2, MEM3, or MEM4) with extension .DAT.

   c. Specify the third parameter:
      - For diskette or NFS server: DISK (default)
      - For internal memory: MEMORY

Example

To store measurement data to a diskette file:

```
10 !
20 ASSIGN @Hp415x TO 717
30 !
40 OUTPUT @Hp415x;":MMEM:DEST INT"
50 OUTPUT @Hp415x;":MMEM:STOR:TRAC DEF,'SWP.DAT','DISK'"
60 !
70 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>40</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>50</td>
<td>Stores measurement data to diskette file SWP.DAT.</td>
</tr>
</tbody>
</table>
To Load Setup Data

1. Send :MMEMory:DESTination command to the 4155C/4156C to specify the mass storage device.

   Specify the command parameter:

   **INT**  Selects the built-in flexible disk drive.

   **NETn** Selects the NFS server. n=1, 2, 3 or 4.

2. Send :MMEMory:LOAD:STATe command to the 4155C/4156C.
   a. Specify the first parameter to be 0. This parameter has no meaning for the 4155C/4156C, but is necessary for SCPI compatibility.
   b. Specify the second parameter:
      - From diskette or NFS server:
        File name with extension: .MES for measurement setup data or .STR for stress setup data.
      - From internal memory:
        Internal memory name (MEM1, MEM2, MEM3, or MEM4) with extension: .MES for measurement setup data or .STR for stress setup data.
   c. Specify the third parameter:
      - From diskette or NFS server: DISK (default)
      - From internal memory: MEMORY

Example

To load measurement setup data from a diskette file:

10 !
20 ASSIGN @Hp415x TO 717
30 !
40 OUTPUT @Hp415x;":MMEM:DEST INT"
50 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES','DISK'"
60 !
70 END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>40</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>50</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
</tbody>
</table>
To Load Measurement Data

1. Send :MMEMory:DESTination command to the 4155C/4156C to specify the mass storage device.

   Specify the command parameter:
   
   **INT** Selects the built-in flexible disk drive.
   **NETn** Selects the NFS server. \( n = 1, 2, 3 \) or 4.

2. Send :MMEMory:LOAD:TRACe command to the 4155C/4156C.

   a. Specify the first parameter to be **DEFault**. This file has no meaning for the 4155C/4156C, but is necessary for SCPI compatibility.

   b. Specify the second parameter:

      * From diskette or NFS server:
        File name with extension \( .DAT \)
      * From internal memory:
        Internal memory name \( (\text{MEM1}, \text{MEM2}, \text{MEM3}, \text{MEM4}) \) with extension \( .DAT \).

   c. Specify the third parameter:

      * From diskette or NFS server: **DISK** (default)
      * From internal memory: **MEMORY**

**Example**

To load measurement data from a diskette file:

```
10 !
20 ASSIGN @Hp415x TO 717
30 !
40 OUTPUT @Hp415x;"::MMEM:DEST INT"
50 OUTPUT @Hp415x;"::MMEM:LOAD:TRAC DEF,'SWP.DAT','DISK'"
60 !
70 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>40</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>50</td>
<td>Loads measurement data from diskette file \texttt{SWP.DAT}.</td>
</tr>
</tbody>
</table>
To Store Spreadsheet Data

1. Send :MMEMory:DESTination command to the 4155C/4156C to specify the mass storage device.
   Specify the command parameter:
   - INT: Selects the built-in flexible disk drive.
   - NETn: Selects the NFS server. n=1, 2, 3 or 4.

2. Send :MMEMory:STORe:SSHeet:DELimiter command to the 4155C/4156C to specify the delimiter; SPACE, TAB, or COMma.

3. Send :MMEMory:STORe:SSHeet:LINDex command to the 4155C/4156C to specify the first data index and the last data index for data output. The MINimum and MAXimum are also available for the data index.

4. Send :MMEMory:STORe:SSHeet:SMARK command to the 4155C/4156C to specify the string mark; NONE, DQUote (double quotes), or SQUote (single quotes).

5. Send :MMEMory:STORe:SSHeet:UNIT command to the 4155C/4156C to set the unit ON or OFF. If you set it to ON, the measured data will be saved with the unit.

6. Send :MMEMory:STORe:SSHeet command to the 4155C/4156C to specify the file name and save the spreadsheet data. Then use the single quotes to specify the file name.

Example

The following example stores a spreadsheet data to a diskette file. The example uses a variable to specify the file name.

```
10 !
20 ASSIGN @Hp415x TO 717
30 DIM A$[8] !Max 8 characters for diskette file name
40 A$="DATA1" !File Name
50 !
60 OUTPUT @Hp415x;"":MMEM:DEST INT"
70 OUTPUT @Hp415x;"":MMEM:STOR:SSH:DEL COMM"
80 OUTPUT @Hp415x;"":MMEM:STOR:SSH:LIND 1,MAX"
90 OUTPUT @Hp415x;"":MMEM:STOR:SSH:SMARK DQU"
100 OUTPUT @Hp415x;"":MMEM:STOR:SSH:UNIT ON"
110 OUTPUT @Hp415x;"":MMEM:STOR:SSH ";CHR$(39)&A$&CHR$(39)
120 !CHR$(39) should be used instead of a single quote
130 END
```
### SCPI Command Programming

#### Programming: File Operation

<table>
<thead>
<tr>
<th>LineNumber</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Declares the maximum number of the characters for the file name.</td>
</tr>
<tr>
<td>40</td>
<td>Defines the file name. For example, <code>DATA1</code>.</td>
</tr>
<tr>
<td>60</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>70</td>
<td>Sets the comma as the delimiter.</td>
</tr>
<tr>
<td>80</td>
<td>Sets the minimum data index to 1 and the maximum index to <code>MAX</code> that means the maximum index of the measured data.</td>
</tr>
<tr>
<td>90</td>
<td>Sets the double quotes as the string mark.</td>
</tr>
<tr>
<td>100</td>
<td>Enables to save the data with the unit.</td>
</tr>
<tr>
<td>110</td>
<td>Stores a spreadsheet data to the diskette file <code>DATA1.TXT</code>. The file extension <code>TXT</code> is automatically set.</td>
</tr>
</tbody>
</table>
Programming: Data Transfer

This section describes the data transfer between a program and the 4155C/4156C.

The following programming tasks are described in this section:

- “To Read 4155/56 Measurement Data”
- “To Transfer Data to 4155C/4156C (Using User Variable)”
To Read 4155/56 Measurement Data

Send :DATA? query command to get data variable values (output data, measurement data, user function values) or read-out function values from 4155/56 to IBASIC variables.

Example 1
To get measurement data, then store it in a data array:

```
10 DIM I3(1:501)
20 !
30 ASSIGN @Hp415x TO 717
40 !
50 OUTPUT @Hp415x;":FORM:DATA ASC"
60 !
70 OUTPUT @Hp415x;":DATA? 'I3'"
80 ENTER @Hp415x;I3(*)
90 !
100 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Specifies ASCII data transfer format.</td>
</tr>
<tr>
<td>70 to 80</td>
<td>Gets the values of data variable I3.</td>
</tr>
</tbody>
</table>

Example 2
To get slope of LINE1 for Y2 axis curve on GRAPH/LIST: GRAPHICS screen:

```
10 ASSIGN @Hp415x TO 717
20 !
30 OUTPUT @Hp415x;":DATA? '@L1G2'
40 ENTER @Hp415x;Slope
50 !
60 PRINT Slope
70 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30 to 40</td>
<td>Gets slope of LINE1 for Y2 axis curve on GRAPH/LIST: GRAPHICS screen.</td>
</tr>
</tbody>
</table>
To Transfer Data to 4155C/4156C (Using User Variable)

To transfer a user variable to the 4155C/4156C, use DATA TRACe subsystem commands. A user variable consists of a name, unit, and numeric data.

Transferred user variable data can be used like other data variables in the 4155C/4156C. You can perform calculations between measurement results and transferred data, plot transferred data on GRAPH/LIST: GRAPHICS screen, or list transferred data on GRAPH/LIST: LIST screen.

To transfer numeric data to the 4155C/4156C:

1. Define the data transfer format by using :FORM at[:DATA] command.
   - For ASCII data transfer format, send :FORM ASC.
   - For REAL 64-bit length data transfer format, send :FORM REAL, 64.
   - For REAL 32-bit length data transfer format, send :FORM REAL, 32.

2. For REAL data transfer format, define byte order by using :FORM at:BOARDer command.
   - For normal order, send :FORM:BOARD NORM.
   - For swapped order, send :FORM:BOARD SWAP.

3. Define name of the user variable, unit (optional), and number of numeric data by using the :PAGE:CH ANnels:UV ARiable:DEF ine command.
   You can also define these parameters by using the :DATA:DEF ine and :DATA:UNIT command.
   If user variable is already defined, you do not have to perform this step.

**Example 1**

To transfer data array by using ASCII data transfer format:

```plaintext
10 DIM Uvar1(1:5)
20 !
30 ASSIGN @Hp415x TO 717
40 !
50 Uvar1(1)=1.0
60 Uvar1(2)=1.1
70 Uvar1(3)=1.2
80 Uvar1(4)=1.3
90 Uvar1(5)=1.4
100 !
110 OUTPUT @Hp415x;";FORM:DATA ASC"
120 OUTPUT @Hp415x;";PAGE:CHAN:UVAR:DEF 'UVAR1','V',5"
130 OUTPUT @Hp415x;";TRAC 'UVAR1','';Uvar1(*)
140 !
150 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>110</td>
<td>Specifies ASCII data transfer format.</td>
</tr>
<tr>
<td>120</td>
<td>Defines user variable.</td>
</tr>
<tr>
<td>130</td>
<td>Transfers user variable.</td>
</tr>
</tbody>
</table>
Example 2

To transfer data array by using REAL 64-bit data transfer format:

```
10 DIM Uvar1(1:101)
20 INTEGER I
30!
40 ASSIGN @Hp415x TO 717
50 ASSIGN @Form_off TO 717;FORMAT OFF
60!
70 FOR I=1 TO 101
80    Uvar1(I)=SQRT(I)
90    NEXT I
100!
110 OUTPUT @Hp415x;"":FORM REAL,64"
120 OUTPUT @Hp415x;"":FORM:BORD NORM"
130 OUTPUT @Hp415x;"":PAGE:CHAN:UVAR:DEF 'UVAR1','1',101"
140 OUTPUT @Hp415x;"":TRAC 'UVAR1','#0";
150 OUTPUT @Form_off;Uvar1(*),END
160!
170 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Assigns I/O path to transfer data.</td>
</tr>
<tr>
<td>110 to 120</td>
<td>Specifies REAL 64 bit data transfer format.</td>
</tr>
<tr>
<td>130</td>
<td>Defines a user variable.</td>
</tr>
<tr>
<td>140 to 150</td>
<td>Transfers user variable.</td>
</tr>
</tbody>
</table>
Example 3

To transfer data, then display plot of transferred data and measurement results:

```
10 DIM Uvar1(1:101)
20 !
30 ASSIGN @Hp415x TO 717
40 !
50 FOR I=1 TO 101
60 Uvar1(I)=SQRT(I)
70 NEXT I
80 !
90 OUTPUT @Hp415x;":MMEM:DEST INT"
100 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES'"
110 OUTPUT @Hp415x;":PAGE:SCON:SING"
120 OUTPUT @Hp415x;":OPC?"
130 ENTER @Hp415x;Complete
140 !
150 OUTPUT @Hp415x;":FORM ASC"
160 OUTPUT @Hp415x;":DATA:DEF 'UVAR1',101"
170 OUTPUT @Hp415x;":DATA:UNIT 'UVAR1','V'"
180 OUTPUT @Hp415x;":DATA 'UVAR1','Uvar1(*)"
190 !
200 OUTPUT @Hp415x;":PAGE:DISP:GRAP:Y2:NAME 'UVAR1'"
210 OUTPUT @Hp415x;":PAGE:GLIS"
220 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>90</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>100</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>110</td>
<td>Executes measurement.</td>
</tr>
<tr>
<td>120 to 130</td>
<td>Waits for measurement completion.</td>
</tr>
<tr>
<td>150</td>
<td>Specifies ASCII data transfer format.</td>
</tr>
<tr>
<td>160</td>
<td>Defines user variable.</td>
</tr>
<tr>
<td>170</td>
<td>Defines unit of user variable.</td>
</tr>
<tr>
<td>180</td>
<td>Transfers user variable.</td>
</tr>
<tr>
<td>200</td>
<td>Sets user variable to Y 2 axis of graph.</td>
</tr>
<tr>
<td>210</td>
<td>Displays GRAPH/LIST: GRAPHICS screen.</td>
</tr>
</tbody>
</table>
Programming: Print/Plot Operation

For the print/plot operation, you can use :HCOPy subsystem commands. This section describes the following tasks:

- “To Output Setup Data to Printer/Plotter”
- “To Output Graphics Result Data to Printer/Plotter”
- “To Output List Results Data to Printer/Plotter”
- “To Dump Screen Image to Printer/Plotter”
- “To Save Print/Plot Data to a File”
- “To Save Hardcopy Image to a File”

Before doing print/plot operation

Before performing above tasks, the following print or plot settings must be set interactively or by remote commands. We recommend that you save the following settings in a file, then load it before printing or plotting.

1. Printer information

<table>
<thead>
<tr>
<th>Setting Parameter</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>destination</td>
<td>:HCOP:DEST</td>
</tr>
<tr>
<td>color mode</td>
<td>:HCOP:DEV:CMOD</td>
</tr>
<tr>
<td>control language</td>
<td>:HCOP:DEV:LANG</td>
</tr>
<tr>
<td>resolution (PCL)</td>
<td>:HCOP:DEV:RES</td>
</tr>
</tbody>
</table>
2. Output Items

<table>
<thead>
<tr>
<th>Item</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title of the print or plot out</td>
<td>:HCOP:ITEM:ANN:STAT</td>
</tr>
<tr>
<td>User defined comment for screen group</td>
<td>:HCOP:ITEM:ANN2:STAT</td>
</tr>
<tr>
<td>Present date and time of the built-in clock</td>
<td>:HCOP:ITEM:TDST:STAT</td>
</tr>
<tr>
<td>Page number of the print or plot out</td>
<td>:HCOP:ITEM:PNUM:STAT</td>
</tr>
<tr>
<td>User defined comment for print or plot out</td>
<td>:HCOP:ITEM:LAB:STAT</td>
</tr>
<tr>
<td>Graphics plot curve</td>
<td>:HCOP:ITEM:TRAC:STAT</td>
</tr>
<tr>
<td>Frame and grid</td>
<td>:HCOP:ITEM:TRAC:GRAT:STAT</td>
</tr>
<tr>
<td>Marker, cursor, and data variable coordinate fields, and line parameters (gradients and intercepts)</td>
<td>:HCOP:ITEM:TEXT:STAT</td>
</tr>
<tr>
<td>Names, units, and scale of the graph axis</td>
<td>:HCOP:ITEM:TEXT2:STAT</td>
</tr>
</tbody>
</table>

If you use GPIB printer/plotter

1. Set the GPIB address:

<table>
<thead>
<tr>
<th>Item</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPIB address of printer/plotter</td>
<td>:SYST:COMM:GPIB:RDEV:ADDR</td>
</tr>
</tbody>
</table>

2. To use built-in IBASIC:

Set "4155C/56C is" field on the SYSTEM : MISCELLANEOUS screen to SYSTEM CONTROLLER.
If you use a remote printer

1. If you use a remote printer via your print server, you need to connect the 4155C/4156C to your LAN. To connect the 4155C/4156C to your LAN, refer to User’s Guide: General Information.

2. Enter the following SCPI commands or set the following entry fields on the SYSTEM : MISCELLANEOUS screen before printing out:

<table>
<thead>
<tr>
<th>SCPI Command a</th>
<th>SYSTEM: MISCELLANEOUS Screen</th>
</tr>
</thead>
<tbody>
<tr>
<td>:SYST:COMM:SELF:NAM</td>
<td>NETWORK SETUP table HOST NAME</td>
</tr>
<tr>
<td>:SYST:COMM:SELF:IPAD</td>
<td>NETWORK SETUP table IP ADDRESS</td>
</tr>
<tr>
<td>:SYST:COMM:SELF:SNET</td>
<td>NETWORK SETUP table SUBNET MASK</td>
</tr>
<tr>
<td>:SYST:COMM:SELF:GATE</td>
<td>NETWORK SETUP table GATEWAY</td>
</tr>
<tr>
<td>:SYST:COMM:SELF:USER</td>
<td>NETWORK SETUP table USER ID</td>
</tr>
<tr>
<td>:SYST:COMM:SELF:GROU</td>
<td>NETWORK SETUP table GROUP ID</td>
</tr>
<tr>
<td>:SYST:COMM:PRIN:NET:NAM</td>
<td>NETWORK PRINTER SETUP table PRINTER</td>
</tr>
<tr>
<td>:SYST:COMM:PRIN:NET:IPAD</td>
<td>NETWORK PRINTER SETUP table IP ADDRESS</td>
</tr>
<tr>
<td>:SYST:COMM:PRIN:NET:TEXT</td>
<td>NETWORK PRINTER SETUP table TEXT OUT</td>
</tr>
<tr>
<td>:SYST:COMM:PRIN:NET:GRAP</td>
<td>NETWORK PRINTER SETUP table GRAPH OUT</td>
</tr>
<tr>
<td>:SYST:COMM:PRIN:NET:TYPE</td>
<td>NETWORK PRINTER SETUP table SERVER TYPE</td>
</tr>
<tr>
<td>:SYST:COMM:PRIN:NET:SET</td>
<td>(same as selecting ADD softkey)</td>
</tr>
<tr>
<td>:SYST:NTMO</td>
<td>SYSTEM SETUP table LP TIMEOUT</td>
</tr>
</tbody>
</table>

a. For details of the SCPI commands, refer to SCPI Command Reference.
To Output Setup Data to Printer/Plotter

1. If you want to output print/plot comment, enter comment by using :HCOPy:ITEM:LABel:TEXT command.

2. Specify the range of setup data to print/plot by sending :HCOPy:OPAGE command.
   - To print/plot present screen setup data, send :HCOPy:OPAGE CURRENT
   - To print/plot present screen group setup data, send :HCOPy:OPAGE GROUP
   - To print/plot all setup data, send :HCOPy:OPAGE ALL

3. Display the screen that you want to print/plot by using the appropriate command:

<table>
<thead>
<tr>
<th>Screen</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANNELS: CHANNEL DEFINITION</td>
<td>:PAGE:CHAN</td>
</tr>
<tr>
<td>CHANNELS: USER FUNCTION DEFINITION</td>
<td>:PAGE:CHAN:UFUN</td>
</tr>
<tr>
<td>CHANNELS: USER VARIABLE DEFINITION</td>
<td>:PAGE:CHAN:UVAR</td>
</tr>
<tr>
<td>MEASURE: SWEEP SETUP</td>
<td>:PAGE:MEAS</td>
</tr>
<tr>
<td>MEASURE: SAMPLING SETUP</td>
<td>:PAGE:MEAS:SAMP</td>
</tr>
<tr>
<td>MEASURE: PGU SETUP</td>
<td>:PAGE:MEAS:PGUS</td>
</tr>
<tr>
<td>MEASURE: MEASURE SETUP</td>
<td>:PAGE:MEAS:MSET</td>
</tr>
<tr>
<td>MEASURE: OUTPUT SEQUENCE</td>
<td>:PAGE:MEAS:OSEQ</td>
</tr>
<tr>
<td>DISPLAY: DISPLAY SETUP</td>
<td>:PAGE:DISP</td>
</tr>
<tr>
<td>DISPLAY: ANALYSIS SETUP</td>
<td>:PAGE:DISP:ANAL</td>
</tr>
<tr>
<td>STRESS: CHANNEL DEFINITION</td>
<td>:PAGE:STR</td>
</tr>
<tr>
<td>STRESS: STRESS SETUP</td>
<td>:PAGE:STR:SET</td>
</tr>
<tr>
<td>STRESS: STRESS FORCE</td>
<td>:PAGE:STR:FORC</td>
</tr>
</tbody>
</table>

If you print/plot from built-in IBASIC, change display mode to All Instrument or IBASIC Status by sending :DISPlay[:WINDow]:ALLocation command.
4. Print/plot the setup data by sending :HCOPy command.

If you print/plot from an external computer using a GPIB peripheral, pass Active Controller capability to the 4155C/4156C after sending :HCOPy command because the 4155C/4156C requires Active Controller capability to print.

Refer to the following examples.

Example 1

The example loads a sweep setup file, then prints setup data of the MEASURE: SWEEP SETUP screen. The program is for an external computer.

```
10 ASSIGN @Hp415x TO 717
20 CONTROL 7,3;21
30 !
40 OUTPUT @Hp415x;"*RST"
50 OUTPUT @Hp415x;"*PCB 21"
60 !
70 OUTPUT @Hp415x;":MMEM:DEST INT"
80 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES'"
90 OUTPUT @Hp415x;":HCOP:ITEM:PNUM:STAT OFF"
100 OUTPUT @Hp415x;":HCOP:ITEM:LAB:TEXT 'This is an example'
```

Line number 110 and above, see next page

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C from external computer.</td>
</tr>
<tr>
<td>20</td>
<td>Sets the GPIB address of external computer. This will be necessary to return Active Controller capability from the 4155C/4156C back to the external computer.</td>
</tr>
<tr>
<td>40</td>
<td>Resets the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Specifies to pass Active Controller capability back to external computer after printing is completed.</td>
</tr>
<tr>
<td>70</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>80</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>90</td>
<td>Specifies to not print the page number.</td>
</tr>
<tr>
<td>100</td>
<td>Defines a print/plot comment.</td>
</tr>
</tbody>
</table>
for line number 100 and below, see previous page

110 OUTPUT @Hp415x;"*:HCOP:DEST RDEV"
120 OUTPUT @Hp415x;"*:HCOP:OPAG CURR"
130 !
140 OUTPUT @Hp415x;"*:PAGE:MEAS"
150 !
160 OUTPUT @Hp415x;"*:HCOP"
170 REPEAT
180 OUTPUT @Hp415x;"*:ESR?"
190 ENTER @Hp415x;Event_status
200 UNTIL BIT(Event_status,1)
210 !
220 PASS CONTROL @Hp415x
230 DISP "Printing"
240 REPEAT
250 STATUS 7,6;Hpib_status
260 UNTIL BIT(Hpib_status,6)
270 DISP "Done"
280 END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>110</td>
<td>Selects GPIB interface. If parallel interface, change the parameter to &quot;PAR&quot;.</td>
</tr>
<tr>
<td>120 to 140</td>
<td>Specifies to print/plot the setup data of the MEASURE: SWEEP SETUP screen.</td>
</tr>
<tr>
<td>160 to 200</td>
<td>Sends print command and waits for Active Controller request from the 4155C/4156C.</td>
</tr>
<tr>
<td>220</td>
<td>Passes Active Controller capability to the 4155C/4156C, then the 4155C/4156C starts printing.</td>
</tr>
<tr>
<td>240 to 260</td>
<td>Waits until printing is complete.</td>
</tr>
</tbody>
</table>
Example 2

The example loads a sweep setup file, then prints setup data of the MEASURE: SWEEP SETUP screen. The program is for built-in IBASIC.

```
10 ASSIGN @Hp415x TO 800
20 !
30 OUTPUT @Hp415x;"*RST"
40 !
50 OUTPUT @Hp415x;":MMEM:DEST INT"
60 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES','DISK'"
70 OUTPUT @Hp415x;":HCOP:ITEM:PNUM:STAT OFF"
80 OUTPUT @Hp415x;":HCOP:ITEM:LAB:TEXT 'This is an example'"
90 OUTPUT @Hp415x;":HCOP:DEST RDEV"
100 OUTPUT @Hp415x;":HCOP:OPAG CURR"
110 !
120 OUTPUT @Hp415x;":DISP:ALL INST"
130 OUTPUT @Hp415x;":PAGE:MEAS"
140 !
150 OUTPUT @Hp415x;":HCOP"
160 DISP "Printing"
170 OUTPUT @Hp415x;"*OPC?"
180 ENTER @Hp415x;Complete
190 DISP "Done"
200 !
210 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C from built-in IBASIC.</td>
</tr>
<tr>
<td>30</td>
<td>Resets the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>60</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>70</td>
<td>Specifies to not print the page number.</td>
</tr>
<tr>
<td>80</td>
<td>Defines a print/plot comment.</td>
</tr>
<tr>
<td>90</td>
<td>Selects GPIB interface. If parallel interface, change the parameter to &quot;PAR&quot;.</td>
</tr>
<tr>
<td>100 to 130</td>
<td>Specifies to print/plot the setup data of the MEASURE: SWEEP SETUP screen.</td>
</tr>
<tr>
<td>150</td>
<td>Starts printing.</td>
</tr>
<tr>
<td>170 and 180</td>
<td>Waits until printing is complete.</td>
</tr>
</tbody>
</table>
To Output Graphics Result Data to Printer/Plotter

1. If you want to output print/plot comment, enter comment by using :HCOPY:ITEM:LABEL:TEXT command.

   If you print/plot from built-in IBASIC, change display mode to All Instrument or IBASIC Status display mode by sending :DISPLAY[:WINDOW]:ALLocation command.

3. Execute print/plot by using :HCOPY command.
   If you print/plot from an external computer using a GPIB peripheral, pass Active Controller capability to the 4155C/4156C after sending :HCOPY command because the 4155C/4156C requires Active Controller capability to print.

Refer to the following examples.

**Example 1**

The example loads a sweep setup file, executes measurement, then prints measurement results of GRAPH/LIST: GRAPHICS screen. The program is for an external computer.

10  ASSIGN @Hp415x TO 717
20  CONTROL 7,3;21
30  !
40  OUTPUT @Hp415x;"*RST"
50  OUTPUT @Hp415x;"*PCB 21"
60  !
70  OUTPUT @Hp415x;":MMEM:DEST INT"
80  OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,"SWP.MES"
90  !
100 OUTPUT @Hp415x;":PAGE:SCON:SING"
110 OUTPUT @Hp415x;"*OPC?"
120 ENTER @Hp415x;Complete
130  !
140 OUTPUT @Hp415x;":HCOP:DEST RDEV"
150  !
160 OUTPUT @Hp415x;":PAGE:GLIS"
170  !
180 OUTPUT @Hp415x;":HCOP"
190 REPEAT
200  OUTPUT @Hp415x;"*ESR?"
210 ENTER @Hp415x;Event_status
220 UNTIL BIT(Event_status,1)
230  !
240 PASS CONTROL @Hp415x
250 DISP "Printing"
260 REPEAT
270  STATUS 7,6;HpiB_status
280 UNTIL BIT(HpiB_status,6)
290 DISP "Done"
300 END
<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C from external computer.</td>
</tr>
<tr>
<td>20</td>
<td>Sets the GPIB address of external computer. This will be necessary to return Active Controller capability from the 4155C/4156C back to the external computer.</td>
</tr>
<tr>
<td>40</td>
<td>Resets the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Specifies to pass Active Controller capability back to external computer after printing is completed.</td>
</tr>
<tr>
<td>70</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>80</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>100 to 120</td>
<td>Executes measurement and waits until completed.</td>
</tr>
<tr>
<td>140</td>
<td>Selects GPIB interface. If parallel interface, change the parameter to &quot;PAR&quot;.</td>
</tr>
<tr>
<td>160</td>
<td>Changes screen to GRAPH/LIST: GRAPHICS screen.</td>
</tr>
<tr>
<td>180 to 220</td>
<td>Sends print command and waits for Active Controller request from the 4155C/4156C.</td>
</tr>
<tr>
<td>240</td>
<td>Passes Active Controller capability to the 4155C/4156C, then the 4155C/4156C starts printing.</td>
</tr>
<tr>
<td>260 to 280</td>
<td>Waits until printing is complete.</td>
</tr>
</tbody>
</table>
Example 2

The example loads a sweep setup file, executes measurement, then prints measurement results of GRAPH/LIST: GRAPHICS screen. The program is for built-in IBASIC.

```
10 ASSIGN @Hp415x TO 800
20 !
30 OUTPUT @Hp415x;"*RST"
40 !
50 OUTPUT @Hp415x;";MMEM:DEST INT"
60 OUTPUT @Hp415x;";MMEM:LOAD:STAT 0,'SWP.MES'"
70 !
80 OUTPUT @Hp415x;";PAGE:SCON:SING"
90 OUTPUT @Hp415x;"*OPC?"
100 ENTER @Hp415x;Complete
110 !
120 OUTPUT @Hp415x;";HCOP:DEST RDEV"
130 !
140 OUTPUT @Hp415x;";DISP:ALL INST"
150 OUTPUT @Hp415x;";PAGE:GLIS"
160 !
170 OUTPUT @Hp415x;";HCOP"
180 DISP "Printing"
190 OUTPUT @Hp415x;"*OPC?"
200 ENTER @Hp415x;Complete
210 DISP "Done"
220 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C from built-in IBASIC.</td>
</tr>
<tr>
<td>30</td>
<td>Resets the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>60</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>80 to 100</td>
<td>Executes measurement and waits until complete.</td>
</tr>
<tr>
<td>120</td>
<td>Selects GPIB interface. If parallel interface, change the parameter to &quot;PAR&quot;.</td>
</tr>
<tr>
<td>140 to 150</td>
<td>Changes screen to GRAPH/LIST: GRAPHICS screen.</td>
</tr>
<tr>
<td>170 to 200</td>
<td>Starts printing and waits until completion.</td>
</tr>
</tbody>
</table>
To Output List Results Data to Printer/Plotter

1. Specify the range of measurement results to output by using :HCOPy:LINDex command.

2. If you want to output print/plot comment, enter comment by using :HCOPy:ITEM:LABel:TEXT command.

   If you print/plot from built-in IBASIC, change display mode to All Instrument or IBASIC Status display mode by sending :DISPlay[:WINDow]:ALLocation command.

4. Execute print/plot by using :HCOPy command.
   If you print/plot from an external computer using a GPIB peripheral, pass Active Controller capability to the 4155C/4156C after sending :HCOPy command because the 4155C/4156C requires Active Controller capability to print.

Refer to the following examples.

Example 1

The example loads a sweep setup file, executes measurement, then prints measurement results of GRAPH/LIST: LIST screen. The program is for an external computer.

```bash
10  ASSIGN @Hp415x TO 717
20  CONTROL 7,3,21
30  !
40  OUTPUT @Hp415x;"*RST"
50  OUTPUT @Hp415x;"*PCB 21"
60  !
70  OUTPUT @Hp415x;":MMEM:DEST INT"
80  OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES'"
90  !
100 OUTPUT @Hp415x;":PAGE:SCON:SING"
110 OUTPUT @Hp415x;":OPC?"
120 ENTER @Hp415x;Complete
130 !
140 OUTPUT @Hp415x;":HCOP:DEST RDEV"
150 OUTPUT @Hp415x;":HCOP:LIND MAX"
160 !
170 OUTPUT @Hp415x;":PAGE:GLIS:LIST"
180 !
190 OUTPUT @Hp415x;":HCOP"
200 REPEAT
210 OUTPUT @Hp415x;"*ESR?"
220 ENTER @Hp415x;Event_status
230 UNTIL BIT(Event_status,1)
240 !
250 PASS CONTROL @Hp415x
260 DISP "Printing"
270 REPEAT
280 STATUS 7,6;HpiB_status
```
Programming: Print/Plot Operation

290  UNTIL BIT(Hpib_status,6)  
300  DISP "Done"  
310  END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C from external computer.</td>
</tr>
<tr>
<td>20</td>
<td>Sets the GPIB address of external computer. This will be necessary to return Active Controller capability from the 4155C/4156C back to the external computer.</td>
</tr>
<tr>
<td>40</td>
<td>Resets the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Specifies to pass Active Controller capability back to external computer after printing is completed.</td>
</tr>
<tr>
<td>70</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>80</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>100 to 120</td>
<td>Executes measurement and waits until completed.</td>
</tr>
<tr>
<td>140</td>
<td>Selects GPIB interface. If parallel interface, change the parameter to &quot;PAR&quot;.</td>
</tr>
<tr>
<td>150</td>
<td>Sets the range of list results to be output.</td>
</tr>
<tr>
<td>170</td>
<td>Changes screen to GRAPH/LIST: LIST screen.</td>
</tr>
<tr>
<td>190 to 230</td>
<td>Sends print command and waits for Active Controller request from the 4155C/4156C.</td>
</tr>
<tr>
<td>250</td>
<td>Passes Active Controller capability to the 4155C/4156C, then the 4155C/4156C starts printing.</td>
</tr>
<tr>
<td>270 and 290</td>
<td>Waits until completion of printing.</td>
</tr>
</tbody>
</table>
Example 2

The example loads a sweep setup file, executes measurement, then prints measurement results of GRAPH/LIST: LIST screen. The program is for built-in IBASIC.

```
10 ASSIGN @Hp415x TO 800
20 !
30 OUTPUT @Hp415x;"*RST"
40 !
50 OUTPUT @Hp415x;":MMEM:DEST INT"
60 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0, 'SWP.MES'"
70 !
80 OUTPUT @Hp415x;":PAGE:SCON:SING"
90 OUTPUT @Hp415x;"*OPC?"
100 ENTER @Hp415x;Complete
110 !
120 OUTPUT @Hp415x;":HCOP:DEST RDEV"
130 OUTPUT @Hp415x;":HCOP:LIND MAX"
140 !
150 OUTPUT @Hp415x;":DISP:ALL INST"
160 OUTPUT @Hp415x;":PAGE:GLIS:LIST"
170 !
180 OUTPUT @Hp415x;":HCOP"
190 DISP "Printing"
200 OUTPUT @Hp415x;"*OPC?"
210 ENTER @Hp415x;Complete
220 DISP "Done"
230 END
```

### Line Number | Description
--- | ---
10 | Assigns I/O path to control the 4155C/4156C from built-in IBASIC.
30 | Resets the 4155C/4156C.
50 | Sets the mass storage device to the built-in flexible disk drive.
60 | Loads measurement setup data from diskette file SWP.MES.
80 to 100 | Executes measurement and waits until completion.
120 | Selects GPIB interface. If parallel interface, change the parameter to "PAR".
130 | Sets the range of list results to be output.
150 to 160 | Changes screen to GRAPH/LIST: LIST screen.
180 to 210 | Starts printing and waits until completion.
SCPI Command Programming
Programming: Print/Plot Operation

To Dump Screen Image to Printer/Plotter

1. Display the screen to be dumped.
2. Execute print/plot by using :HCOPy:SDUMP command.

   If you print/plot from an external computer using a GPIB peripheral, pass Active Controller capability to the 4155C/4156C after sending :HCOPy:SDUMP command because the 4155C/4156C requires Active Controller capability to print.

Refer to the following example.

Example 1

The example loads a sweep setup file, executes measurement, displays GRAPH/LIST:GRAPHICS screen, then dumps screen image of GRAPH/LIST:GRAPHICS screen to printer/plotter. The program is for an external computer.

```
10 ASSIGN @Hp415x TO 717
20 CONTROL 7,3;21
30 !
40 OUTPUT @Hp415x;"*RST"
50 OUTPUT @Hp415x;"*PCB 21"
60 !
70 OUTPUT @Hp415x;":MMEM:DEST INT"
80 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,SWP.MES"
90 !
100 OUTPUT @Hp415x;":PAGE:SCON:SING"
110 OUTPUT @Hp415x;"*OPC?"
120 ENTER @Hp415x;Complete
130 !
140 OUTPUT @Hp415x;":HCOP:DEST RDEV"
150 !
160 OUTPUT @Hp415x;":PAGE:GLIS"
170 !
180 OUTPUT @Hp415x;":HCOP:SDUMP"
190 REPEAT
200 OUTPUT @Hp415x;"*ESR?"
210 ENTER @Hp415x;Event_status
220 UNTIL BIT(Event_status,1)
230 !
240 PASS CONTROL @Hp415x
250 DISP "Printing"
260 REPEAT
270 STATUS 7,6;HPIB_status
280 UNTIL BIT(HPIB_status,6)
290 DISP "Done"
300 END
```
## SCPI Command Programming

### Programming: Print/Plot Operation

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C from external computer.</td>
</tr>
<tr>
<td>20</td>
<td>Sets the GPIB address of external computer. This will be necessary to return Active Controller capability from the 4155C/4156C back to the computer.</td>
</tr>
<tr>
<td>40</td>
<td>Resets the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Specifies to pass Active Controller capability back to external computer after printing is completed.</td>
</tr>
<tr>
<td>70</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>80</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>100 to 120</td>
<td>Executes measurement and waits until completed.</td>
</tr>
<tr>
<td>140</td>
<td>Selects GPIB interface. If parallel interface, change the parameter to &quot;PAR&quot;.</td>
</tr>
<tr>
<td>160</td>
<td>Changes screen to GRAPH/LIST: GRAPHICS screen.</td>
</tr>
<tr>
<td>180 to 220</td>
<td>Sends screen dump command and waits for Active Controller request from the 4155C/4156C.</td>
</tr>
<tr>
<td>240</td>
<td>Passes Active Controller capability to the 4155C/4156C, then the 4155C/4156C starts printing.</td>
</tr>
<tr>
<td>260 and 280</td>
<td>Waits until printing is complete.</td>
</tr>
</tbody>
</table>
Example 2

The example loads a sweep setup file, executes measurement, displays GRAPH/LIST: GRAPHICS screen, then dumps screen image of GRAPH/LIST: GRAPHICS screen to printer/plotter. The program is for built-in IBASIC.

```
10  ASSIGN @Hp415x TO 800
20  !
30  OUTPUT @Hp415x;"*RST"
40  !
50  OUTPUT @Hp415x;":MMEM:DEST INT"
60  OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES'"
70  !
80  OUTPUT @Hp415x;":PAGE:SCON:SING"
90  OUTPUT @Hp415x;"*OPC?"
100 ENTER @Hp415x;Complete
110  !
120 OUTPUT @Hp415x;":DISP:ALL INST"
130 OUTPUT @Hp415x;":PAGE:GLIS"
140  !
150 OUTPUT @Hp415x;":HCOP:DEST RDEV"
160 OUTPUT @Hp415x;":HCOP:SDUM"
170 OUTPUT @Hp415x;"*OPC?"
180 ENTER @Hp415x;Complete
190 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C from built-in IBASIC.</td>
</tr>
<tr>
<td>30</td>
<td>Resets the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>60</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>80 to 100</td>
<td>Executes measurement and waits until completion.</td>
</tr>
<tr>
<td>120 to 130</td>
<td>Displays GRAPH/LIST: GRAPHICS screen.</td>
</tr>
<tr>
<td>150</td>
<td>Selects GPIB interface. If parallel interface, change the parameter to &quot;PAR&quot;.</td>
</tr>
<tr>
<td>160 to 180</td>
<td>Starts printing and waits until completion.</td>
</tr>
</tbody>
</table>
SCPI Command Programming
Programming: Print/Plot Operation

To Save Print/Plot Data to a File

1. Use the :HCOP:ITEM subsystem commands to add/select the print/plot output items. See “Before doing print/plot operation” on page 2-60.

2. Use the :PAGE command to specify the screen to print/plot.

3. Use the :HCOP:ITEM:ALL:DATA? command to return the all output data. Or use the :HCOP:DATA? command to return the output data selected by the :HCOP:ITEM subsystem commands.

Example

The following program loads a sweep setup file, executes measurement, and puts the print/plot output data of the GRAPH/LIST: LIST screen into the 4155C/4156C output buffer. Then the computer reads the output data, and saves it to a file named “SAMPLE1.ASC”. The following example program saves the print/plot output data that contains the header lines (11 lines) and the measurement data lines (51 lines) displayed on the GRAPH/LIST: LIST screen.

```
10 INTEGER Nop, Head
20 Head=11 ! Number of header lines
30 Nop=51 ! Number of data lines
40 DIM A$[100]
50 DIM Get_file$(8)
60 Get_file$="SWP.MES" ! Setup file for measurement
70 File$="SAMPLE1.ASC" ! Data file to be saved
80 !
90 ASSIGN @Hp415x TO 717
100 OUTPUT @Hp415x;"*RST"
110 OUTPUT @Hp415x;":MMEM:DEST INT"
120 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,";CHR$(39)&Get_file$&CHR$(39)
130 OUTPUT @Hp415x;":PAGE:SCON:SING"
140 OUTPUT @Hp415x;":OPC?"
150 ENTER @Hp415x;Complete
160 !
170 !OUTPUT @Hp415x;":HCOP:ITEM:LAB:TEXT 'Measurement Data'"
180 OUTPUT @Hp415x;":PAGE:GLIS:LIST" ! LIST screen
190 !OUTPUT @Hp415x;":HCOP:DATA?"
200 OUTPUT @Hp415x;":HCOP:ITEM:ALL:DATA?"
210 !
220 CREATE File$,1
230 ASSIGN @Io TO File$;FORMAT ON
240 FOR I=1 TO Nop+Head
250 ENTER @Hp415x;A$
260 OUTPUT @Io;A$
270 NEXT I
280 ASSIGN @Io TO *
290 !
300 END
```
### Programming: Print/Plot Operation

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>Sets the number of header lines for the print/plot output of the GRAPH/LIST: LIST screen. The lines will be saved to a file.</td>
</tr>
<tr>
<td>30</td>
<td>Sets the number of lines for the measurement data displayed on the LIST screen. The lines will be saved to a file.</td>
</tr>
<tr>
<td>40 to 50</td>
<td>Declares the dimension of the strings A$ and Get_file$.</td>
</tr>
<tr>
<td>60</td>
<td>Specifies the setup data file used for measurement.</td>
</tr>
<tr>
<td>70</td>
<td>Sets the name of the print/plot output data file to be saved.</td>
</tr>
<tr>
<td>90</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>100</td>
<td>Resets the 4155C/4156C.</td>
</tr>
<tr>
<td>110</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>120</td>
<td>Loads a setup data file from a diskette.</td>
</tr>
<tr>
<td>130 to 150</td>
<td>Executes measurement and waits until completion.</td>
</tr>
<tr>
<td>170</td>
<td>This line is not used for this example. This line can be used to add a text to a header line. The :HCOP:ITEM subsystem commands can be used to add/select the output items.</td>
</tr>
<tr>
<td>180</td>
<td>Displays the GRAPH/LIST: LIST screen.</td>
</tr>
<tr>
<td>190</td>
<td>This line is not used for this example. This line can be used to return the items selected by the :HCOP:ITEM commands.</td>
</tr>
<tr>
<td>200</td>
<td>Returns all of the print/plot output data.</td>
</tr>
<tr>
<td>220 to 280</td>
<td>Reads the print/plot output data, and saves it to a file.</td>
</tr>
</tbody>
</table>
To Save Hardcopy Image to a File

1. To set print/plot destination to a mass storage device, then specify the device, send the following commands:
   - :HCOP:DEST MMEM
   - :MMEM:DEST INT or :MMEM:DEST NETn
     where, n is 1, 2, 3 or 4.

2. Specify the file name by using :MMEMory:NAME command.

3. Execute the print/plot operation. Refer to print/plot tasks described previously.

Example

To load sweep setup file, execute measurement, and then saves a hardcopy image of the measurement results of GRAPH/LIST: GRAPHICS screen to a diskette:

```
10 ASSIGN @Hp415x TO 717
20 !
30 OUTPUT @Hp415x;"*RST"
40 !
50 OUTPUT @Hp415x;":MMEM:DEST INT"
60 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'SWP.MES'"
70 !
80 OUTPUT @Hp415x;":PAGE:SCON:SING"
90 OUTPUT @Hp415x;"*OPC?"
100 ENTER @Hp415x;Complete
110 !
120 OUTPUT @Hp415x;":HCOP:DEST MMEM"
130 OUTPUT @Hp415x;":MMEM:NAME 'TEST1'"
140 !
150 OUTPUT @Hp415x;":PAGE:GLIS"
160 !
170 OUTPUT @Hp415x;":HCOP"
180 OUTPUT @Hp415x;"*OPC?"
190 ENTER @Hp415x;Complete
200 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>30</td>
<td>Resets the 4155C/4156C.</td>
</tr>
<tr>
<td>50</td>
<td>Sets the mass storage device to the built-in flexible disk drive.</td>
</tr>
<tr>
<td>60</td>
<td>Loads measurement setup data from diskette file SWP.MES.</td>
</tr>
<tr>
<td>80 to 100</td>
<td>Executes measurement and waits until completion.</td>
</tr>
<tr>
<td>120</td>
<td>Specifies the destination to be diskette.</td>
</tr>
<tr>
<td>130</td>
<td>Specifies the diskette file name.</td>
</tr>
<tr>
<td>150</td>
<td>Displays GRAPH/LIST: GRAPHICS screen.</td>
</tr>
<tr>
<td>170 to 190</td>
<td>Starts printing and waits until completion.</td>
</tr>
</tbody>
</table>
Other Programming Tips

This section provides the advanced programming techniques and useful tips:

- Speed Improvement
- Auto-loading of Files
- Differences from 4155A/4156A/4155B/4156B SCPI Commands

Disabling Instrument Screen Update to Improve Speed

Most of the commands that control and set the 4155C/56C will also update the instrument screen.

For example, :PAGE:CHAN:MODE command changes the measurement mode. This command also changes the instrument screen to the CHANNELS: CHANNEL DEFINITION screen and updates the MEASUREMENT MODE field setting.

This instrument screen update is useful for confirming the settings that were changed by the commands, but it takes time. You can enable or disable this time consuming instrument screen update as follows:

`:DISP OFF` Instrument screen is not updated
`:DISP ON` Instrument screen is updated

where, :DISP OFF command is NOT available when the 4155C/4156C screen displays the following screen:

- SYSTEM: FILER
- SYSTEM: MISCELLANEOUS
- SYSTEM: CONFIGURATION
- SYSTEM: SELF-CALIBRATION/DIAGNOSTICS
- SYSTEM: PRINT/ PLOT SETUP
- SYSTEM: COLOR SETUP
- KNOB SWEEP

Refer to SCPI Command Reference.
Auto-loading of Files

The 4155C/56C can automatically load files when it is turned on.

**INIT files for Initial Settings**

If any setup files named INIT.MES, INIT.STR, INIT.CST, or INIT.DAT are on the diskette (in the built-in drive) when the 4155C/56C is turned on, the 4155C/56C automatically loads these setup files to be the initial settings.

This function saves you the trouble of getting application files every time you turn on the 4155C/56C.

**NOTE**

INIT.MES and INIT.DAT both contain measurement setup data. If both these files exist on the diskette, the 4155C/56C gets INIT.DAT, not INIT.MES.

**MEMno Files**

If any files named MEMno.DAT, MEMno.MES, or MEMno.STR are on the diskette in the drive, the files are automatically loaded from diskette to internal memory when the 4155C/56C is turned on. Where MEMno means MEM1, MEM2, MEM3, or MEM4, which correspond to the four internal memory areas.

If the same internal memory is specified by multiple files (for example, MEM1.MES and MEM1.DAT), the priority is as follows:

1. DAT
2. MES
3. STR

**IBASIC Program File to Auto-execute**

If an IBASIC program is stored in a file named "AUTOST" on the diskette in the built-in drive, the program is automatically loaded and started when you turn on the 4155C/56C.
**Differences From 4155B/4156B SCPI Commands**

The 4155C/4156C SCPI command set differs from the 4155B/4156B SCPI command set in the following ways:

**SCPI commands supported by the 4155C/4156C**

The 4155C/4156C supports the following new commands:

- Quasi-static CV measurement:
  
  :PAGE:MEASURE:QSCV subsystem commands

- E5250A control (available for the built-in IBASIC controller):
  
  :PAGE:CHANNELS:MATRIX subsystem commands

- Network capability:

<table>
<thead>
<tr>
<th>SCPI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:SYST:COMM:NET:SELF:SNET</td>
<td>Sets the subnet mask the 4155C/4156C connects to.</td>
</tr>
<tr>
<td>:SYST:COMM:NET:SELF:GATE</td>
<td>Sets the IP address of the gateway.</td>
</tr>
</tbody>
</table>

- System configuration query:

<table>
<thead>
<tr>
<th>SCPI Command</th>
<th>Description</th>
</tr>
</thead>
</table>

- VMU discharge function:
  
  :PAGE:CHANNELS:CDEF:VMU <n>:DCH command

- Data display resolution:
  
  :PAGE:DISPLAY:SET:DRESOL command
Differences in the Command Parameters

- :PAGE:CHAN:CDEF:MODE command

Command parameters are different as shown below:

<table>
<thead>
<tr>
<th>Model</th>
<th>Parameters available</th>
</tr>
</thead>
<tbody>
<tr>
<td>4155A/4156A</td>
<td>SW eep, SM ping</td>
</tr>
<tr>
<td>4155B/4156B</td>
<td></td>
</tr>
<tr>
<td>4155C/4156C</td>
<td>SW eep, SM ping, QSCV</td>
</tr>
</tbody>
</table>
Differences From 4155A/4156A SCPI Commands

The 4155C/4156C SCPI command set differs from the 4155A/4156A SCPI command set in the following ways:

For information on the built-in IBA SC controller, refer to the “Differences from 4155A/4156A Programming” section in Chapter 1.

SCPI commands supported on 4155B/4156B/4155C/4156C

The following commands are not supported on the 4155A/4156A, but are supported on the 4155B/4156B/4155C/4156C:

- To enable/disable the screen saver:
  :SYST:SSAV command

- To select the mass storage memory:
  :MMEM:DEST command

- To setup the 4155/4156 network:

<table>
<thead>
<tr>
<th>SCPI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:SYST:COMM:NET:SELF:NAME</td>
<td>Sets the host name of the 4155/4156.</td>
</tr>
<tr>
<td>:SYST:COMM:NET:SELF:USER</td>
<td>Sets your user ID.</td>
</tr>
<tr>
<td>:SYST:COMM:NET:SELF:GROU</td>
<td>Sets your group ID.</td>
</tr>
</tbody>
</table>

- To use the 4155/4156 as a NFS client:

<table>
<thead>
<tr>
<th>SCPI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:MMEM:CDIR</td>
<td>Changes the working directory.</td>
</tr>
</tbody>
</table>
SCPI Command Programming
Other Programming Tips

- To use a remote printer:

<table>
<thead>
<tr>
<th>SCPI Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:SYST:COMM:NET:PRIN:NET:NAME</td>
<td>Sets the name of the remote printer.</td>
</tr>
<tr>
<td>:SYST:NTMO</td>
<td>Sets the print server timeout.</td>
</tr>
</tbody>
</table>

Differences in the Command Parameters

- :HCOP:DEST command

The command parameters of the supported interfaces differ from the 4155A/4156A as follows, where NETn, n is 1, 2, 3, or 4.

<table>
<thead>
<tr>
<th>Model</th>
<th>Serial</th>
<th>Parallel</th>
<th>GPIB</th>
<th>LAN</th>
<th>file</th>
</tr>
</thead>
<tbody>
<tr>
<td>4155A/4156A</td>
<td>SERial</td>
<td></td>
<td>RDEVic</td>
<td></td>
<td>MMEMory</td>
</tr>
<tr>
<td>4155B/4156B</td>
<td></td>
<td>PARalle</td>
<td>RDEVic</td>
<td>NETn</td>
<td>MMEMory</td>
</tr>
<tr>
<td>4155C/4156C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- :HCOP:DEV:LANG command

The print/plot function command parameters of the supported output formats differ as shown below. In the table, HR PCL means high resolution PCL, and HR TIFF means high resolution TIFF.

<table>
<thead>
<tr>
<th>Model</th>
<th>PCL</th>
<th>HR PCL</th>
<th>HP-GL</th>
<th>TIFF</th>
<th>HR TIFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>4155A/4156A</td>
<td>PCL</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4155B/4156B</td>
<td>PCL</td>
<td>HRPCI</td>
<td>HPGL</td>
<td>TIFF</td>
<td>HRTiff</td>
</tr>
<tr>
<td>4155C/4156C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
This section shows a programming example with SCPI commands that performs the same operations as the desired 4145 ASP program.

Built-in IBASIC can execute ASP-like commands for controlling the 4155C/4156C. Refer to "Creating ASP-like IBASIC Programs" in Chapter 5 on programming this commands.

Following program is the simplest example of creating an HP BASIC program (with SCPI commands) that performs the same operations as the desired 4145 ASP program. The ASP program gets a setup file named "VTH" from the diskette, makes a single measurement, then saves measurement to a file named "VTH1".

```
10 ASSIGN @Hp415x TO 800
20 OUTPUT @Hp415x;":MMEM:DEST INT"
1 GET P VTH 30 OUTPUT @Hp415x;":MMEM:LOAD:STAT 0,'VTH.PRO'
2 SINGLE 40 OUTPUT @Hp415x;":PAGE:SCON:SING"
50 OUTPUT @Hp415x;"*OPC?"
60 ENTER @Hp415x;Complete
3 SAVE D VT1 70 OUTPUT @Hp415x;":MMEM:STOR:TRAC DEF,'VT1.DAT'
80 END
```

The above HP BASIC program (with SCPI commands) does as follows:

- Line 10 assigns a path named @Hp415x to 800, which is the select code/GPIB address to use if this is an IBASIC program running in the 4155C/56C. If this program will run on an external computer, use the select code of the GPIB interface and the GPIB address of the 4155C/4156C instead.

- Lines 20 to 30 get a measurement setup file named "VTH.MES". So, you need to save setup data to a file named "VTH.MES" on the diskette before executing this program. For an example setup, see "Example Application Setup for Vth Measurement" on page 2-26.

- Line 40 performs a single measurement.

- Line 70 saves measurement setup and result data to a file named "VT1.DAT".

For built-in help function, which makes it easier to enter the desired SCPI command, see “To Use the Help Function” in Chapter 1.
SCPI Command Programming
Programming Example for 4145 Users

Following shows the 4145A/B's ASP keywords and corresponding SCPI commands of the 4155C/4156C:

**Corresponding 4145 ASP and 4155C/56C SCPI Commands**

<table>
<thead>
<tr>
<th>4145A/B</th>
<th>SCPI Commands</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET P</td>
<td>:MEM:LOAD:STAT</td>
<td>Gets setup .MES or .PRO file</td>
</tr>
<tr>
<td>SINGLE</td>
<td>:PAGE:SCON:SING</td>
<td>Initiates single measurement</td>
</tr>
<tr>
<td>SAVE D</td>
<td>:MEM:STOR:TRAC</td>
<td>Saves data to .DAT file</td>
</tr>
<tr>
<td>PLOT</td>
<td>:HCOP</td>
<td>Prints/plots present instrument screen.</td>
</tr>
<tr>
<td>CPLOT</td>
<td>:HCOP:ITEM:TRAC</td>
<td>Prints/plots measurement graph.</td>
</tr>
<tr>
<td>PRINT</td>
<td>:HCOP</td>
<td>Prints/plots present instrument screen.</td>
</tr>
<tr>
<td>PAUSE</td>
<td>(Use BASIC keyword PAUSE)</td>
<td></td>
</tr>
<tr>
<td>WAIT</td>
<td>(Use BASIC keyword WAIT)</td>
<td></td>
</tr>
<tr>
<td>PAGE</td>
<td>(Set in the Print/Plot setup)</td>
<td></td>
</tr>
</tbody>
</table>
3  FLEX Command Programming
FLEX Command Programming

A gilent 4155C/4156C FLEX (Fast Language for EXecution) command set is designed to make automatic measurements via GPIB control. This is the fastest method of measurement for the 4155C/4156C.

This chapter describes how to create measurement programs, and provides program examples. It contains the following sections:

• “Programming Basics”
• “High-Speed Spot Measurements”
• “Spot Measurements”
• “1 Channel Pulsed Spot Measurements”
• “Staircase Sweep Measurements”
• “Pulsed Sweep Measurements”
• “Staircase Sweep with Pulsed Bias Measurements”
• “Sampling Measurements”
• “Quasi-static CV Measurements”
• “Linear Search Measurements”
• “Binary Search Measurements”
• “Stress Force”
• “Controlling PGU”
• “Using Program Memory”
• “Using Enhanced Sweep Stop Function”
• “Reading Time Stamp Data”
• “Reading and Writing Data to a File”
• “Printing Data”
• “Reading Binary Output Data”
• “Using the US42 Control Mode”
• “Programming Tips”
FLEX Command Programming

Refer to Chapter 1 of GPIB Command Reference for the command syntax and descriptions of the 4155C/4156C FLEX commands.

The following command conventions are used in this chapter.

command
Required command for measurement execution.

[command]
Optional command for measurement execution.

parameter
Required command parameter. A value or variable must be specified.

[param]
Optional command parameter. A value may be specified.

---

**CAUTION**

After the Automatic Measurement

After the automatic measurements, open the measurement terminals or disconnect the device under test from the measurement terminals. If you leave the connection with the device, the device may be damaged by unexpected operations.

Do not leave the connection over 30 minutes after the measurement if the auto calibration is set to ON. Then, the 4155C/4156C performs the self-calibration automatically every 30 minutes after the measurement. The calibration requires to open the measurement terminals.

To disable the auto calibration, enter the `CM 0` command.
Programming Basics

This section provides instructions for two methods of controlling and programming the 4155C/4156C.

- Controlling the 4155C/4156C via GPIB
- Controlling the 4155C/4156C using HP BASIC

Controlling 4155C/4156C via GPIB

To control the 4155C/4156C via GPIB, you can use an external computer or the built-in Instrument BASIC (IBASIC) controller.

NOTE

Device Clear

The 4155C/4156C requires approximately 2 seconds for the GPIB device clear. For HP BASIC or IBASIC, enter the CLEAR command.

Controlling from an External Computer

To control the 4155C/4156C using an external computer:

1. Connect the GPIB interface of the external computer to the GPIB connector on the rear panel of the 4155C/4156C.
2. Set the "4155C/56C is" field on the SYSTEM : MISCELLANEOUS screen to NOT SYSTEM CONTROLLER.
3. Enter the GPIB address of the 4155C/4156C in the "GPIB ADDRESS" field.

Controlling from a built-in IBASIC controller

If you use a built-in IBASIC controller, no preparation is required. The built-in IBASIC controller is always connected to the 4155C/4156C via internal GPIB.

To control external instruments:

1. Connect the GPIB interface for the external instruments to the GPIB connector on the rear panel of the 4155C/4156C.
2. Set the "4155C/56C is" field on the SYSTEM : MISCELLANEOUS screen to SYSTEM CONTROLLER.
To use the remote printer connected to the print server:

1. Connect the 4155C/4156C to your LAN.
2. Set the "4155C/56C NETWORK SETUP" table, "NETWORK PRINTER SETUP" table, and "NETWORK DRIVE SETUP" table on the SYSTEM: MISCELLANEOUS screen.

To use the network file system on the NFS server:

1. Connect the 4155C/4156C to your LAN.
2. Set the "4155C/56C NETWORK SETUP" table and "NETWORK DRIVE SETUP" table on the SYSTEM: MISCELLANEOUS screen.

**Controlling 4155C/4156C Using HP BASIC Programming**

1. To assign the I/O path for controlling the 4155C/4156C, use the ASSIGN command.
   - **Built-in IBASIC**
     Specify select code 8. For the GPIB address, you can use any number from 0 to 31. Refer to the following example.
     10 ASSIGN @Hp415x TO 800
   - **HP BASIC on an external computer**
     Specify the select code of the external computer and specify the GPIB address that you entered in the "GPIB ADDRESS" field on the SYSTEM: MISCELLANEOUS screen.
     In the following example, the select code of the external computer is 7 and the GPIB address of the 4155C/4156C is 17.
     10 ASSIGN @Hp415x TO 717

2. To send commands to the 4155C/4156C, use the OUTPUT command, as shown in the following example.
   OUTPUT @Hp415x:"US"
   The 4155C/4156C will only accept a single statement in an OUTPUT command. Do not enter multiple statements.

3. To get a query response or output data from the 4155C/4156C, use the ENTER command.
**High-Speed Spot Measurements**

To make high-speed spot measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables Measurement Units</td>
<td>CN</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Disables Measurement Units</td>
<td>CL</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Filter ON/OFF</td>
<td>[FL]</td>
<td>mode [,chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Averaging Number</td>
<td>[AV]</td>
<td>number [,mode]</td>
</tr>
<tr>
<td>Sets Integration Time</td>
<td>[SIT]</td>
<td>type, time</td>
</tr>
<tr>
<td></td>
<td>[SLI]</td>
<td>type</td>
</tr>
<tr>
<td>Forces constant voltage</td>
<td>DV</td>
<td>chnum, range, output [,Icomp]</td>
</tr>
<tr>
<td>Forces constant current</td>
<td>DI</td>
<td>chnum, range, output [,Vcomp]</td>
</tr>
<tr>
<td>Measures current</td>
<td>TI</td>
<td>chnum, range</td>
</tr>
<tr>
<td>Measures voltage</td>
<td>TV</td>
<td>chnum, range</td>
</tr>
<tr>
<td>Reads measurement data (for TI/TV command)</td>
<td>RMD?</td>
<td>number_of_data</td>
</tr>
<tr>
<td>Measures current and reads data</td>
<td>TI?</td>
<td>chnum, range</td>
</tr>
<tr>
<td>Measures voltage and reads data</td>
<td>TV?</td>
<td>chnum, range</td>
</tr>
</tbody>
</table>

You can use the DV/DI commands, and TI/TV or TI?/TV? commands regardless of the measurement mode (MM command settings).

You cannot use both the TI/TV commands and the TI?/TV? commands in the same measurement program.
A program example of a high-speed spot measurement is shown below. This program executes the current measurement, using the TI? command, and prints the measured data on the screen.

```plaintext
10 ASSIGN @Hp415x TO 800
20 !
30 INTEGER Fmt,Average,Type,Source,Drain,Gate,Sub
40 INTEGER Range_2v,Range_20v,Range_i,B,C
50 DIM B$[50]
60 !
70 Fmt=1           !1:ASCII with Header <LF^EOI>
80 Average=1       !Number of averaging
90 Sinteg=.0005    !Integ Time of Short
100 Linteg=.04     !Integ Time of Long
110 Type=1          !1:Short, 2:Medium, 3:Long
120 Filter=0       !0:Filter off, 1:Filter on
130 Source=1       !1:SMU1
140 Drain=2         !2:SMU2
150 Gate=3          !3:SMU3
160 Sub=4           !4:SMU4
170 Range_2v=11     !11: 2 V Limited Auto Ranging
180 Range_20v=12    !12:20 V Limited Auto Ranging
190 Range_i=15      !15:10 uA Limited Auto Ranging
200 Vs=0            ! Source Voltage
210 Vd=5            ! Drain Voltage
220 Vg=3            ! Gate Voltage
230 Vsub=0          ! Substrate Voltage
240 Icomp_g=.01     ! Current compliance for gate
250 Icomp=.1        ! Current compliance
260 !
270 OUTPUT @Hp415x;"US"
280 OUTPUT @Hp415x;"FMT ",Fmt
290 OUTPUT @Hp415x;"AV ",Average
300 OUTPUT @Hp415x;"SIT 1,",Sinteg !for Short
310 OUTPUT @Hp415x;"SIT 3,",Linteg !for Long
320 OUTPUT @Hp415x;"SLI ",Type
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>70 to 250</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>270</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>280</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>290 to 320</td>
<td>Sets the integration time.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
High-Speed Spot Measurements

330 OUTPUT @Hp415x;"FL ";Filter
340 OUTPUT @Hp415x;"CN ";Source,Drain,Gate,Sub
350 OUTPUT @Hp415x;"DV ";Source,Range_2v,Vs,Icomp
360 OUTPUT @Hp415x;"DV ";Sub,Range_2v,Vsub,Icomp
370 OUTPUT @Hp415x;"DV ";Gate,Range_20v,Vg,Icomp_g
380 OUTPUT @Hp415x;"DV ";Drain,Range_20v,Vd,Icomp
390 !
400 OUTPUT @Hp415x;"*OPC?"
410 ENTER @Hp415x;C
420 !
430 OUTPUT @Hp415x;":SYST:ERR?"
440 ENTER @Hp415x;B,B$
450 IF B=0 THEN
460 OUTPUT @Hp415x;"TI ? ";Drain,Range_i
470 ENTER @Hp415x USING ";#,5X,13D,X";A
480 PRINT "Id(A)=";A
490 ELSE
500 PRINT "ERROR:";B$
510 END IF
520 !
530 OUTPUT @Hp415x;"CL"
540 OUTPUT @Hp415x;":PAGE"
550 END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>330</td>
<td>Sets the filter mode.</td>
</tr>
<tr>
<td>340</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>350 to 380</td>
<td>Forces the dc voltage.</td>
</tr>
<tr>
<td>400 to 410</td>
<td>Waits for the operation complete flag.</td>
</tr>
<tr>
<td>430 to 440</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>460 to 480</td>
<td>Executes a high-speed spot measurement and prints the results on the screen.</td>
</tr>
<tr>
<td>500</td>
<td>Displays an error code if an error has occurred.</td>
</tr>
<tr>
<td>530</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>540</td>
<td>Returns to the 4155C/4156C default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
# Spot Measurements

To make spot measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables Measurement Units</td>
<td>CN</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Disables Measurement Units</td>
<td>CL</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Filter ON/OFF</td>
<td>[FL]</td>
<td>mode[,chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Averaging Number</td>
<td>[AV]</td>
<td>number[,mode]</td>
</tr>
<tr>
<td>Sets Integration Time</td>
<td>[SIT]</td>
<td>type,time</td>
</tr>
<tr>
<td></td>
<td>[SLI]</td>
<td>type</td>
</tr>
<tr>
<td>Forces constant voltage</td>
<td>DV</td>
<td>chnum,range,output[,Icomp]</td>
</tr>
<tr>
<td>Forces constant current</td>
<td>DI</td>
<td>chnum,range,output[,Vcomp]</td>
</tr>
<tr>
<td>Sets VMU measurement mode</td>
<td>[VM]</td>
<td>chnum,mode</td>
</tr>
<tr>
<td>Sets voltage measurement range</td>
<td>[RV]</td>
<td>chnum,range[,Rmode]</td>
</tr>
<tr>
<td>Sets current measurement range</td>
<td>[RI]</td>
<td>chnum,range[,Rmode]</td>
</tr>
<tr>
<td></td>
<td>[RM]</td>
<td>chnum,mode[,rate]</td>
</tr>
<tr>
<td>Selects measurement mode</td>
<td>MM</td>
<td>1,chnum[,chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets SMU measurement mode</td>
<td>[CM M]</td>
<td>chnum,mode</td>
</tr>
<tr>
<td>Executes measurement</td>
<td>XE</td>
<td></td>
</tr>
<tr>
<td>Reads measurement data</td>
<td>RMD?</td>
<td>number_of_data</td>
</tr>
</tbody>
</table>
A program example of a spot measurement is shown below. This program executes the current measurement and prints the measured data on the screen.

```
10   ASSIGN @Hp415x TO 800
20   !
30   INTEGER Fmt,Average,Type,Source,Drain,Gate,Sub
40   INTEGER Range_2v,Range_20v,Range_i,B,C
50   DIM B$[50]
60   !
70   Fmt=1           !1:ASCII with Header <LF^EOI>
80   Average=1      !Number of averaging
90   Sinteg=.0005   !Integ Time of Short
100  Linteg=.04     !Integ Time of Long
110  Type=1         !1:Short, 2:Medium, 3:Long
120  Filter=0       !0:Filter off, 1:Filter on
130  Source=1       !1:SMU1
140  Drain=2        !2:SMU2
150  Gate=3         !3:SMU3
160  Sub=4          !4:SMU4
170  Range_2v=11    !11: 2 V Limited Auto Ranging
180  Range_20v=12   !12: 20 V Limited Auto Ranging
190  Range_i=15     !15: 10 uA Limited Auto Ranging
200  Vs=0           ! Source Voltage
210  Vd=5           ! Drain Voltage
220  Vg=3           ! Gate Voltage
230  Vsub=0         ! Substrate Voltage
240  Icomp_g=.01    ! Current compliance for gate
250  Icomp=.1       ! Current compliance
260  Mmode=1        !1:Spot Measurement
270  Smode=1        !1:Current Measurement
280  Mnum=1        !Number of measurement data
290   !
300  OUTPUT @Hp415x;"US"
310  OUTPUT @Hp415x;"FMT ";Fmt
320  OUTPUT @Hp415x;"AV ";Average
330  OUTPUT @Hp415x;"SIT 1,";Sinteg !for Short
340  OUTPUT @Hp415x;"SIT 3,";Linteg !for Long
350  OUTPUT @Hp415x;"SLI ";Type
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>70 to 280</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>300</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>310</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>320 to 350</td>
<td>Sets the integration time.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Spot Measurements

360   OUTPUT @Hp415x;"FL ";Filter
370   OUTPUT @Hp415x;"CN ";Source,Drain,Gate,Sub
380   OUTPUT @Hp415x;"DV ";Source,Range_2v,Vs,Icomp
390   OUTPUT @Hp415x;"DV ";Sub,Range_2v,Vsub,Icomp
400   OUTPUT @Hp415x;"DV ";Gate,Range_20v,Vg,Icomp_g
410   OUTPUT @Hp415x;"DV ";Drain,Range_20v,Vd,Icomp
420   OUTPUT @Hp415x;"MM ";Mmode,Drain
430   OUTPUT @Hp415x;"CMM ";Drain,Smode
440   OUTPUT @Hp415x;"XE"
450  !
460   OUTPUT @Hp415x;":SYST:ERR?"
470   ENTER @Hp415x;B,B$
480   OUTPUT @Hp415x;"CL"
490   IF B=0 THEN
500     OUTPUT @Hp415x;"RMD? ";Mnum
510     ENTER @Hp415x USING ";#,5X,13D,X";A
520     PRINT "Id(A)=";A
530 ELSE
540     PRINT "ERROR:";B$
550   END IF
560  !
570   OUTPUT @Hp415x;":PAGE"
580   END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>360</td>
<td>Sets the filter mode.</td>
</tr>
<tr>
<td>370</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>380 to 410</td>
<td>Forces the dc voltage.</td>
</tr>
<tr>
<td>420</td>
<td>Sets the measurement mode.</td>
</tr>
<tr>
<td>430</td>
<td>Sets the SMU measurement mode.</td>
</tr>
<tr>
<td>440</td>
<td>Executes a spot measurement.</td>
</tr>
<tr>
<td>460 to 470</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>480</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>500 to 520</td>
<td>Reads the measurement data and prints the results on the screen.</td>
</tr>
<tr>
<td>540</td>
<td>Displays an error code if an error has occurred.</td>
</tr>
<tr>
<td>570</td>
<td>Returns to the 4155C/4156C default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
# 1 Channel Pulsed Spot Measurements

To make 1 channel pulsed spot measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables Measurement Units</td>
<td>CN</td>
<td>[chnum ... ,chnum] ... ]</td>
</tr>
<tr>
<td>Disables Measurement Units</td>
<td>CL</td>
<td>[chnum ... ,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Filter ON/OFF</td>
<td>[FL] b</td>
<td>mode,[chnum ... ,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Averaging Number</td>
<td>[AV] b</td>
<td>number[,mode]</td>
</tr>
<tr>
<td>Sets Integration Time</td>
<td>[SIT]</td>
<td>type,time</td>
</tr>
<tr>
<td></td>
<td>[SLI] b</td>
<td>type</td>
</tr>
<tr>
<td>Forces constant voltage</td>
<td>[DV]</td>
<td>chnum,range,output[,Icomp]</td>
</tr>
<tr>
<td>Forces constant current</td>
<td>[DI]</td>
<td>chnum,range,output[,Vcomp]</td>
</tr>
<tr>
<td>Sets pulse source timing parameters</td>
<td>PT</td>
<td>hold,width[,period [,trigger_delay[,priority]]]</td>
</tr>
<tr>
<td>Forces pulse voltage</td>
<td>PV</td>
<td>chnum,range,base,pulse[,Icomp]</td>
</tr>
<tr>
<td>Forces pulse current</td>
<td>PI</td>
<td>chnum,range,base,pulse[,Vcomp]</td>
</tr>
<tr>
<td>Sets voltage measurement range</td>
<td>[RV]</td>
<td>chnum,range[,Rmode]</td>
</tr>
<tr>
<td>Sets current measurement range</td>
<td>[RI]</td>
<td>chnum,range[,Rmode]</td>
</tr>
<tr>
<td></td>
<td>[RM]</td>
<td>chnum[,mode][,rate]</td>
</tr>
<tr>
<td>Selects measurement mode</td>
<td>MM</td>
<td>3,chnum,[chnum ... ,chnum] ... ] c</td>
</tr>
<tr>
<td>Sets SMU measurement mode</td>
<td>[CM M]</td>
<td>chnum,mode</td>
</tr>
<tr>
<td>Executes measurement</td>
<td>XE</td>
<td></td>
</tr>
<tr>
<td>Reads measurement data</td>
<td>RMD?</td>
<td>number_of_data</td>
</tr>
</tbody>
</table>

- a. For the pulse source, the filter must be set to OFF.
- b. If the PT command priority parameter specifies the "keep pulse width" mode, the AV and SLI commands are ignored. The integration time is automatically set to 80 μsec.
- c. If the PT command priority parameter specifies the "keep pulse width" mode, the available number of chnums is 1.
A program example of a 1 channel pulsed spot measurement is shown below. This program executes the current measurement and prints the measured data on the screen.

```
10    ASSIGN @Hp415x TO 800
20    !
30    INTEGER Emitter,Base,Collector,Mmode,Fmt,Filter
40    INTEGER Range,Mnum,B
50    REAL Vcomp,Icomp,Ve,Ibbase,Ibpulse,Ic,Hold,Width
60    DIM B$(50)
70    !
80    Emitter=1   ! 1: SMU1
90    Base=2      ! 2: SMU2
100   Collector=3 ! 3: SMU3
110  Fmt=1        ! 1: ASCII with Header <LF^EOI>
120   Filter=0    ! Filter mode. 0: OFF, 1: ON
130   Range=0     ! Auto ranging
140   Vcomp=2     ! V compliance (V) for base/collector
150   Icomp=.1    ! I compliance (A) for emitter
160   Ve=0        ! Emitter voltage (V)
170   Ibbase=0    ! Base current base value (A)
180   Ibpulse=.005 ! Base current pulse value (A)
190   Ic=.05      ! Collector current (A)
200   Hold=0      ! Hold time (sec) of Ib
210   Width=.001  ! Pulse width (sec) of Ib
220   Mnum=1      ! Number of measurement points
230   Mmode=3     ! 3: 1CH pulsed spot measurement
240   !
250   OUTPUT @Hp415x;"US"
260   OUTPUT @Hp415x;"FMT ";Fmt
270   OUTPUT @Hp415x;"CN ";Emitter,Base,Collector
280   OUTPUT @Hp415x;"FL ";Filter,Base
```

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>80 to 230</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>250</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>260</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>270</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>280</td>
<td>Sets the filter mode.</td>
</tr>
</tbody>
</table>
FLEX Command Pulsed Spot Measurements

1 Channel Pulsed Spot Measurements

290   OUTPUT @Hp415x;"PT ";Hold,Width
300   OUTPUT @Hp415x;"PI ";Base,Range,Ibbase,Ibpulse,Vcomp
310   OUTPUT @Hp415x;"DV ";Emitter,Range,Ve,Icomp
320   OUTPUT @Hp415x;"DI ";Collector,Range,IC,Vcomp
330   OUTPUT @Hp415x;"MM ";Mmode,Collector
340   OUTPUT @Hp415x;"XE"
350   
360   OUTPUT @Hp415x;":SYST:ERR?"
370   ENTER @Hp415x;B,B$
380   OUTPUT @Hp415x;"CL"
390   IF B=0 THEN
400     GOTO 460
410   ELSE
420     PRINT B,B$
430     GOTO 500
440   END IF
450   
460   OUTPUT @Hp415x;"RMD? ";Mnum
470   ENTER @Hp415x USING ",5X,13D,X";Mdata
480   PRINT "Vce(V)= ";Mdata
490   
500   OUTPUT @Hp415x;":PAGE"
510   END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>290 to 300</td>
<td>Sets the pulse current source (base current).</td>
</tr>
<tr>
<td>310</td>
<td>Forces the dc voltage (Ve).</td>
</tr>
<tr>
<td>320</td>
<td>Forces the dc current (Ic).</td>
</tr>
<tr>
<td>330</td>
<td>Sets the measurement mode.</td>
</tr>
<tr>
<td>340</td>
<td>Executes a 1ch pulsed spot measurement.</td>
</tr>
<tr>
<td>360 to 370</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>380</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>460 to 480</td>
<td>Reads the measurement data and prints the results on the screen.</td>
</tr>
<tr>
<td>500</td>
<td>Returns to the 4155C/4156C default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
Staircase Sweep Measurements

To make staircase sweep measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables Measurement Units</td>
<td>CN</td>
<td>[chnum ..., [,chnum] ...]</td>
</tr>
<tr>
<td>Disables Measurement Units</td>
<td>CL</td>
<td>[chnum ..., [,chnum] ...]</td>
</tr>
<tr>
<td>Sets Filter ON/OFF</td>
<td>[FL]</td>
<td>mode[,chnum ..., [,chnum] ...]</td>
</tr>
<tr>
<td>Sets Averaging Number</td>
<td>[AV]</td>
<td>number[,mode]</td>
</tr>
<tr>
<td>Sets Integration Time</td>
<td>[SIT]</td>
<td>type[,time]</td>
</tr>
<tr>
<td></td>
<td>[SLI]</td>
<td>type</td>
</tr>
<tr>
<td>Sets sweep source timing</td>
<td>[WT]</td>
<td>hold, delay[,step delay]</td>
</tr>
<tr>
<td>parameter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sets staircase sweep source</td>
<td>WV</td>
<td>ch[,mode,range,start,stop,step[,]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Icomp[,Pcomp[,Rmode]]]</td>
</tr>
<tr>
<td></td>
<td>WI</td>
<td>ch[,mode,range,start,stop,step[,]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vcomp[,Pcomp[,Rmode]]]</td>
</tr>
<tr>
<td>Sets sweep abort function</td>
<td>[WM]</td>
<td>abort[,post]</td>
</tr>
<tr>
<td>Sets synchronous sweep source</td>
<td>[WSV]</td>
<td>ch[,range,start,stop[,] Icomp[,] Pcomp[,] Rmode]]</td>
</tr>
<tr>
<td>s^a</td>
<td>[WSI]</td>
<td>ch[,range,start,stop[,] Vcomp[,] Pcomp[,] Rmode]]</td>
</tr>
<tr>
<td>Forces constant voltage</td>
<td>[DV]</td>
<td>chnum, range, output[,] Icomp</td>
</tr>
<tr>
<td>Forces constant current</td>
<td>[DI]</td>
<td>chnum, range, output[,] Vcomp</td>
</tr>
<tr>
<td>Sets VMU measurement mode</td>
<td>[VM]</td>
<td>chnum[,mode]</td>
</tr>
<tr>
<td>Sets voltage measurement range</td>
<td>[RV]</td>
<td>chnum, range[,] Rmode</td>
</tr>
</tbody>
</table>
**FLEX Command Programming**

**Staircase Sweep Measurements**

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sets current measurement range</td>
<td>[RI]</td>
<td>chnum,range[,Rmode]</td>
</tr>
<tr>
<td></td>
<td>[RM]</td>
<td>chnum,mode[,rate]</td>
</tr>
<tr>
<td>Selects measurement mode</td>
<td>MM</td>
<td>2,chnum[,chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets SMU measurement mode</td>
<td>[CM M]</td>
<td>chnum,mode</td>
</tr>
<tr>
<td>Executes measurement</td>
<td>XE</td>
<td></td>
</tr>
<tr>
<td>Reads measurement data</td>
<td>RMD?</td>
<td>number_of_data</td>
</tr>
</tbody>
</table>

a. The WSV/WSI command must be entered after the WV/WI command.
A program example of a staircase sweep measurement is shown below. This program executes the bipolar transistor Ic-Vc characteristics measurement and prints the measured data list on the screen.

```
10   ASSIGN @Hp415x TO 800
20   OPTION BASE 1
30   INTEGER Te,Tb,Tc,Irange,Range,Var1,Var2
40   DIM St$[3],Ch$[1],Md$[1],C$[100]
50   !
60   Te=1           ! 1: SMU1
70   Tb=2           ! 2: SMU2
80   Tc=3           ! 3: SMU3
90   Var1=11        ! Collector voltage number of steps
100  Var2=3         ! Number of Ib steps
110  Vc1=0          ! Collector voltage start value (V)
120  Vc2=1          ! Collector voltage stop value (V)
130  Ic_comp=.003   ! Current compliance (A) for collector
140  Ve=0           ! Emitter voltage (V)
150  Ie_comp=.1     ! Current compliance (A) for emitter
160  Ib1=1.E-5      ! Ib start value (A)
170  Ib2=3.E-5      ! Ib stop value (A)
180  Vb_comp=2      ! Voltage compliance (V) for base
190  Irange=14      ! 14: 1 uA limited auto ranging
200  Range=0        ! 0: Auto ranging
210  !
220  OUTPUT @Hp415x;"US"
230  OUTPUT @Hp415x;"FMT 1,1"     !Data w/Header,Source data
240  OUTPUT @Hp415x;"CN ";Te,Tb,Tc
250  OUTPUT @Hp415x;"MM 2, ";Tc  !Sweep measurement
260  OUTPUT @Hp415x;"WM 2,1"   !Stops any abnormal
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>60 to 200</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>220</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>230</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>240</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>250</td>
<td>Sets the staircase sweep measurement mode.</td>
</tr>
<tr>
<td>260</td>
<td>Enables measurement abort function for abnormal conditions.</td>
</tr>
</tbody>
</table>
### FLEX Command Programming

#### Staircase Sweep Measurements

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>270</td>
<td>Sets the staircase sweep source (Vc).</td>
</tr>
<tr>
<td>280</td>
<td>Sets the measurement range (Ic).</td>
</tr>
<tr>
<td>290</td>
<td>Forces the dc voltage (Ve).</td>
</tr>
<tr>
<td>320</td>
<td>Calculates the base current value.</td>
</tr>
<tr>
<td>330</td>
<td>Forces the dc current (Ib).</td>
</tr>
<tr>
<td>340 to 360</td>
<td>Executes a staircase sweep measurement.</td>
</tr>
<tr>
<td>370 to 450</td>
<td>Checks to see if an error has occurred. If an error is detected, the error code and error message are displayed on the screen.</td>
</tr>
</tbody>
</table>
470   ! Reading data
480     OUTPUT @Hp415x;"NUB?"
490     ENTER @Hp415x;C
500     IF C<>0 THEN
510       PRINT "Ib=";Ib*1.E+6;"(uA)"
520       FOR N=1 TO C
530         OUTPUT @Hp415x;"RMD? 1"
540         ENTER @Hp415x USING ":,3A,A,A,13D,X";St$,Ch$,Md$,Mdata
550         IF Md$="I" THEN
560           PRINT " Ic=";Mdata*1000;"(mA), Status=";St$
570         ELSE
580           IF Md$="v" THEN
590             PRINT " Vc=";Mdata*1000;"(mV), Status=";St$
600           ELSE
610             PRINT " XX=INVALID DATA, Status=";St$
620           END IF
630         END IF
640       NEXT N
650     ELSE
660       PRINT "STATUS: No data returned."
670     END IF
680     NEXT I
690     OUTPUT @Hp415x;"CL"
700     OUTPUT @Hp415x;":PAGE"
710   END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>480 to 490</td>
<td>Confirms the number of data stored in the output buffer.</td>
</tr>
<tr>
<td>530 to 540</td>
<td>Reads the measurement data.</td>
</tr>
<tr>
<td>550 to 590</td>
<td>Displays the measurement data, Ic and Vc.</td>
</tr>
<tr>
<td>610</td>
<td>Displays “XX=INVALID DATA” if the data is not the current measurement data or the voltage output data.</td>
</tr>
<tr>
<td>660</td>
<td>Displays “No data returned” if the number of data is zero.</td>
</tr>
<tr>
<td>690</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>700</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Staircase Sweep Measurements

The following program example executes the synchronous sweep measurement using two sweep sources. This program executes the MOS FET Id-Vg characteristics measurement and prints the measured data list on the screen.

```
10    ASSIGN @Hp415x TO 800
20    INTEGER Source,Drain,Gate,Sub,Range_v,Range_i
30    DIM A$(100)
40    !
50    Drain=1        !1:SMU1
60    Gate=2         !2:SMU2
70    Sub=3          !3:SMU3
80    Source=4       !4:SMU4
90    Range_v=0      !0: Auto Ranging
100   Range_i=15     !15:10 uA Limited Auto Ranging
110   Vs=0           ! Source Voltage
120   Vd=3           ! Drain Voltage
130   Vg=3           ! Gate Voltage
140   Vsub=0         ! Substrate Voltage
150   Icomp_g=.01    ! Current compliance for gate
160   Icomp=.011     ! Current compliance
170   Var1=11        ! Number of Var1 step
180   !
190   OUTPUT @Hp415x;"US"
200   OUTPUT @Hp415x;"FMT 1,0"     !Data w/Header, No Source_data
210   OUTPUT @Hp415x;"MM 2,\";Drain !Sweep measurement
220   OUTPUT @Hp415x;"WM 2,\"      !Stops any abnormal
230   OUTPUT @Hp415x;"CN \";Source,Drain,Gate,Sub
240   OUTPUT @Hp415x;"DV \";Source,Range_v,Vs,Icomp
250   OUTPUT @Hp415x;"DV \";Sub,Range_v,Vsub,Icomp
260   OUTPUT @Hp415x;"WV \";Drain,1,Range_v,0,Vd,Var1,Icomp
270   OUTPUT @Hp415x;"WSV \";Gate,Range_v,0,Vg,Icomp_g
280   OUTPUT @Hp415x;"RI \";Drain,Range_i
290   !
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>50 to 170</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>190</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>200</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>210</td>
<td>Sets the staircase sweep measurement mode.</td>
</tr>
<tr>
<td>220</td>
<td>Enables measurement abort function for abnormal conditions.</td>
</tr>
<tr>
<td>230</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>240 to 270</td>
<td>Forces the dc voltage, and sets the voltage sweep sources.</td>
</tr>
<tr>
<td>280</td>
<td>Sets the current measurement range.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Staircase Sweep Measurements

300    OUTPUT @Hp415x;"XE"
310    OUTPUT @Hp415x;"*OPC?"
320    ENTER @Hp415x;A
330    OUTPUT @Hp415x;":SYST:ERR?"
340    ENTER @Hp415x;A,A$
350    IF A<>0 THEN
360      PRINT "ERROR:";A,A$
370      BEEP
380      DISP "Press press Continue to read data. Or press Stop."
390      PAUSE
400      DISP ""
410    END IF
420    !
430    ! Reading data
440    OUTPUT @Hp415x;"NUB?"
450    ENTER @Hp415x;A
460    IF A<>0 THEN
470      FOR I=1 TO A
480        OUTPUT @Hp415x;"RMD? 1"
490      ENTER @Hp415x USING ",3,A,X,13D,X";A$,B$,Mdata
500      IF B$="I" THEN
510        PRINT "Id=";Mdata*1000;"(mA), Status=";A$
520      ELSE
530        PRINT "XX=INVALID DATA, Status=";A$
540      END IF
550      NEXT I
560    ELSE
570      PRINT "STATUS: No data returned."
580    END IF
590    OUTPUT @Hp415x;"CL"
600    OUTPUT @Hp415x;":PAGE"
610    END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>300 to 320</td>
<td>Executes the sweep measurement.</td>
</tr>
<tr>
<td>330 to 410</td>
<td>Checks to see if an error has occurred. If an error is detected, the error code and error message are displayed on the screen.</td>
</tr>
<tr>
<td>440 to 450</td>
<td>Confirms the number of data stored in the output buffer.</td>
</tr>
<tr>
<td>480 to 510</td>
<td>Reads the measurement data, and displays the measurement data.</td>
</tr>
<tr>
<td>530</td>
<td>Displays &quot;X X =INVALID DATA &quot; if the data is not the current measurement data.</td>
</tr>
<tr>
<td>570</td>
<td>Displays &quot;No data returned&quot; if the number of data is zero.</td>
</tr>
<tr>
<td>590</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>600</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
## Pulsed Sweep Measurements

To make staircase sweep measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables Measurement Units</td>
<td>CN</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Disables Measurement Units</td>
<td>CL</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Filter ON/OFF</td>
<td>[FL] (^a)</td>
<td>mode[,chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Averaging Number</td>
<td>[AV] (^b)</td>
<td>number[,mode]</td>
</tr>
<tr>
<td>Sets Integration Time</td>
<td>[SIT]</td>
<td>type,time</td>
</tr>
<tr>
<td></td>
<td>[SL] (^b)</td>
<td>type</td>
</tr>
<tr>
<td>Sets pulse source timing</td>
<td>PT</td>
<td>hold,width[,period [,trigger_delay [,priority]]]</td>
</tr>
<tr>
<td>parameters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sets pulsed sweep source</td>
<td>PWV</td>
<td>ch,mode,range;base,start,stop,step[,Icomp[,Rmode]]</td>
</tr>
<tr>
<td></td>
<td>PWI</td>
<td>ch,mode,range;base,start,stop,step[,Vcomp[,Rmode]]</td>
</tr>
<tr>
<td>Sets sweep abort function</td>
<td>[WM]</td>
<td>abort[,post]</td>
</tr>
<tr>
<td>Sets synchronous sweep source</td>
<td>[WSV]</td>
<td>ch,range,start,stop[,Icomp [,Pcomp[,Rmode]]]</td>
</tr>
<tr>
<td></td>
<td>[WSI]</td>
<td>ch,range,start,stop[,Vcomp [,Pcomp[,Rmode]]]</td>
</tr>
<tr>
<td>Forces constant voltage</td>
<td>[DV]</td>
<td>chnum,range,output[,Icomp]</td>
</tr>
<tr>
<td>Forces constant current</td>
<td>[DI]</td>
<td>chnum,range,output[,Vcomp]</td>
</tr>
<tr>
<td>Sets voltage measurement</td>
<td>[RV]</td>
<td>chnum,range[,Rmode]</td>
</tr>
<tr>
<td>range</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### FLEX Command Programming

#### Pulsed Sweep Measurements

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sets current measurement range</td>
<td>[RI]</td>
<td>chnum,rangen,R.mode</td>
</tr>
<tr>
<td></td>
<td>[RM]</td>
<td>chnum,mode,rangen</td>
</tr>
<tr>
<td>Selects measurement mode</td>
<td>MM</td>
<td>4,chnum[,chnum ... [,chnum] ..] d</td>
</tr>
<tr>
<td>Sets SMU measurement mode</td>
<td>[CM M]</td>
<td>chnum,mode</td>
</tr>
<tr>
<td>Executes measurement</td>
<td>XE</td>
<td></td>
</tr>
<tr>
<td>Reads measurement data</td>
<td>RMD?</td>
<td>number_of_data</td>
</tr>
</tbody>
</table>

a. For the pulse source, the filter must be set to OFF.
b. If the PT command priority parameter specifies the "keep pulse width" mode, the AV and SLI commands are ignored. The integration time is automatically set to 80 μsec.
c. The WSV/WSI command must be entered after the PWV/PWI command.
d. If the PT command priority parameter specifies the "keep pulse width" mode, the available number of chnums is 1.
FLEX Command Programming
Pulsed Sweep Measurements

A program example of a pulsed sweep measurement is shown below. This program executes the bipolar transistor Ib-Vb and Ic-Vb characteristics measurement and prints the measured data list on the screen.

```
10     ASSIGN @Hp415x TO 800
20     OPTION BASE 1
30     INTEGER Emitter,Base,Collector,Pri,Linlog,Mrange,Range
40     DIM A$(100),Channel$(4)
50     Channel$(1)="A"
60     Channel$(2)="B"
70     Channel$(3)="C"
80     Channel$(4)="D"
90    !
100    Emitter=1         ! 1: SMU1
110    Base=2             ! 2: SMU2
120    Collector=3        ! 3: SMU3
130    Hold=1             ! Hold time
140    Width=.001         ! Pulse width
150    Period=.01         ! Pulse period
160    Delay=0            ! Trigger delay
170    Pri=1              ! 1: Wait meas. 0: Keep pulse width.
180    Linlog=1           ! 1: Linear single sweep mode
190    V0=0               ! Base pulse voltage base value (V)
200    V1=0               ! Base pulse voltage start value (V)
210    V2=1               ! Base pulse voltage stop value (V)
220    Comp=.002          ! Current compliance (A) of base
230    Ve=0               ! Emitter voltage (V)
240    Vc=1               ! Collector voltage (V)
250    Icomp=.1           ! Current compliance (A) of emitter/collector
260    Mrange=11          ! 11: 1 nA limited auto ranging
270    Range=0            ! 0: Auto ranging
280    Var1=11            ! Number of Var1 step
290    !
300    OUTPUT @Hp415x;"US"
310    OUTPUT @Hp415x;"FMT 1,1" ! Data w/Header,Source_data
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>100 to 280</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>300</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>310</td>
<td>Specifies the data output format.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Pulsed Sweep Measurements

320    OUTPUT @Hp415x;"FL 0,";Base   !Filter off
330    OUTPUT @Hp415x;"CN ";Emitter,Base,Collector
340    OUTPUT @Hp415x;"PT ";Hold,Width,Period,Delay,Pri
350    OUTPUT @Hp415x;"MM 4,";Base,Collector !Pulsed sweep
360    OUTPUT @Hp415x;"NM 2,1"  !Stops any abnormal
370    OUTPUT @Hp415x;"PVW ";Base,LineLog,Range,V0,V1,V2,Var1,Comp
380    OUTPUT @Hp415x;"DV ";Emitter,Range,Ve,Comp
390    OUTPUT @Hp415x;"Vv ";Collector,Range,Vc,Comp
400    OUTPUT @Hp415x;"RI ";Base,Mrange
410    OUTPUT @Hp415x;"RI ";Collector,Mrange
420  
430    OUTPUT @Hp415x;"XE"
440    OUTPUT @Hp415x;"*OPC?"
450    ENTER @Hp415x;A
460    OUTPUT @Hp415x;"CL"
470    OUTPUT @Hp415x;":SYST:ERR?"
480    ENTER @Hp415x;A,A$
490    IF A<>0 THEN
500      PRINT "ERROR:";A,A$
510      BEEP
520      DISP "Press press Continue to read data. Or press Stop."
530      PAUSE
540      DISP ""
550    END IF
560  !

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>320</td>
<td>Sets the filter mode.</td>
</tr>
<tr>
<td>330</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>340</td>
<td>Sets the pulse timing parameters.</td>
</tr>
<tr>
<td>350</td>
<td>Sets the pulsed sweep measurement mode.</td>
</tr>
<tr>
<td>360 to 370</td>
<td>Sets the sweep stop function, and sets the pulsed sweep source.</td>
</tr>
<tr>
<td>380 to 390</td>
<td>Forces the dc voltage.</td>
</tr>
<tr>
<td>400 to 410</td>
<td>Sets the measurement range.</td>
</tr>
<tr>
<td>430 to 450</td>
<td>Executes a pulsed sweep measurement.</td>
</tr>
<tr>
<td>460</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>470 to 550</td>
<td>Checks to see if an error has occurred. If an error is detected, the error code and error message are displayed on the screen.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Pulsed Sweep Measurements

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>580 to 590</td>
<td>Confirms the number of data stored in the output buffer.</td>
</tr>
<tr>
<td>620 to 740</td>
<td>Reads the measurement data, and displays the measurement data.</td>
</tr>
<tr>
<td>750 to 790</td>
<td>Displays &quot;XX=INVALID DATA&quot; if the data is not the current measurement data or the voltage output data.</td>
</tr>
<tr>
<td>840</td>
<td>Displays &quot;No data returned&quot; if the number of data is zero.</td>
</tr>
<tr>
<td>860</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
### Staircase Sweep with Pulsed Bias Measurements

To make staircase sweep with pulsed bias measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables Measurement Units</td>
<td>CN</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Disables Measurement Units</td>
<td>CL</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Filter ON/OFF</td>
<td>[FL] a</td>
<td>mode[,chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Averaging Number</td>
<td>[AV] b</td>
<td>number[,mode]</td>
</tr>
<tr>
<td>Sets Integration Time</td>
<td>[SIT]</td>
<td>type,time</td>
</tr>
<tr>
<td></td>
<td>[SLI] b</td>
<td>type</td>
</tr>
<tr>
<td>Sets pulse source timing parameters</td>
<td>PT</td>
<td>hold,width[,period [,trigger_delay[,priority]]]</td>
</tr>
<tr>
<td>Forces pulse voltage</td>
<td>PV</td>
<td>chnum,range,base,pulse [,Icomp]</td>
</tr>
<tr>
<td>Forces pulse current</td>
<td>PI</td>
<td>chnum,range,base,pulse [,Vcomp]</td>
</tr>
<tr>
<td>Sets staircase sweep source</td>
<td>WV</td>
<td>ch,mode,range,start,stop,step [,Icomp[,Pcomp[,Rmode]]]</td>
</tr>
<tr>
<td></td>
<td>WI</td>
<td>ch,mode,range,start,stop,step [,Vcomp[,Pcomp[,Rmode]]]</td>
</tr>
<tr>
<td>Sets sweep abort function</td>
<td>[WM]</td>
<td>abort,[post]</td>
</tr>
<tr>
<td>Forces constant voltage</td>
<td>[DV]</td>
<td>chnum,range,output[,Icomp]</td>
</tr>
<tr>
<td>Forces constant current</td>
<td>[DI]</td>
<td>chnum,range,output[,Vcomp]</td>
</tr>
<tr>
<td>Sets voltage measurement range</td>
<td>[RV]</td>
<td>chnum,range,[Rmode]</td>
</tr>
</tbody>
</table>
**FLEX Command Programming**

**Staircase Sweep with Pulsed Bias Measurements**

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sets current measurement range</td>
<td>[R1]</td>
<td>chnum,range[,Rmode]</td>
</tr>
<tr>
<td></td>
<td>[R M]</td>
<td>chnum,mode[,rate]</td>
</tr>
<tr>
<td>Selects measurement mode</td>
<td>M M</td>
<td>5,chnum[,chnum ... [,chnum] ..] c</td>
</tr>
<tr>
<td>Sets SMU measurement mode</td>
<td>[C M M]</td>
<td>chnum,mode</td>
</tr>
<tr>
<td>Executes measurement</td>
<td>X E</td>
<td></td>
</tr>
<tr>
<td>Reads measurement data</td>
<td>R MD?</td>
<td>number_of_data</td>
</tr>
</tbody>
</table>

a. For the pulse source, the filter must be set to OFF.
b. If the PT command priority parameter specifies the "keep pulse width" mode, the AV and SLI commands are ignored. The integration time is automatically set to 80 μsec.
c. If the PT command priority parameter specifies the "keep pulse width" mode, the available number of chnums is 1.
A program example of a staircase sweep with pulsed bias measurement is shown below. This program executes the bipolar transistor $I_c$-$V_c$ characteristics measurement and prints the measured data list on the screen.

```plaintext
10     ASSIGN @Hp415x TO 800
20     OPTION BASE 1
30     INTEGER Emitter, Base, Col, Pri, Linlog, Mrange, Range
40     DIM C$[100]
50    !
60     Emitter=1      ! 1: SMU1
70     Base=2         ! 2: SMU2
80     Col=3    ! 3: SMU3
90     Hold=1         ! Hold time
100    Width=.001     ! Pulse width
110    Period=.01     ! Pulse period
120    Delay=0        ! Trigger delay
130    Pri=0          ! 1: Wait meas. 0: Keep pulse width.
140    Linlog=1       ! 1: Linear single sweep mode
150    Vc1=0          ! Collector voltage start value (V)
160    Vc2=1          ! Collector voltage stop value (V)
170    Comp=.1        ! Current compliance (A) for collector
180    Ve=0           ! Emitter voltage (V)
190    Ie_comp=.003   ! Current compliance (A) for emitter
200    Ib_base=0      ! Ib pulse base value (A)
210    Ib1=1.E-5      ! Ib start value (A)
220    Ib2=1.E-5      ! Ib step value (A)
230    Vb_comp=1      ! Voltage compliance (V) for base
240    Mrange=14      ! 14: 1 uA limited auto ranging
250    Range=0        ! 0: Auto ranging
260    Var1=11        ! Number of Var1 step
270    Var2=3         ! Number of Var2 step
280    !
290    OUTPUT @Hp415x;"US"
300    OUTPUT @Hp415x;"FMT 1,1"      !Data w/Header, Source_data
310    OUTPUT @Hp415x;"FL 0,";Base   !Filter off
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>60 to 270</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>290</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>300</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>310</td>
<td>Sets the filter mode.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Staircase Sweep with Pulsed Bias Measurements

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>320</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>330</td>
<td>Sets the timing parameters of the pulse source (base current).</td>
</tr>
<tr>
<td>340</td>
<td>Sets the sweep with pulsed bias measurement mode.</td>
</tr>
<tr>
<td>350</td>
<td>Enables measurement abort function for abnormal conditions.</td>
</tr>
<tr>
<td>360</td>
<td>Sets the staircase sweep source (Vc).</td>
</tr>
<tr>
<td>370</td>
<td>Sets the measurement range (Ic).</td>
</tr>
<tr>
<td>380</td>
<td>Forces the dc voltage (Ve).</td>
</tr>
<tr>
<td>400</td>
<td>Calculates the base current peak value.</td>
</tr>
<tr>
<td>410</td>
<td>Sets the pulsed bias (Ib).</td>
</tr>
<tr>
<td>420 to 440</td>
<td>Executes a staircase sweep with pulsed bias measurement.</td>
</tr>
<tr>
<td>450 to 530</td>
<td>Checks to see if an error has occurred. If an error is detected, the error code and error message are displayed on the screen.</td>
</tr>
</tbody>
</table>
### Line Number Description

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>550-570</td>
<td>Confirms the number of data stored in the output buffer.</td>
</tr>
<tr>
<td>560-570</td>
<td>Reads the measurement data, and displays the measurement data.</td>
</tr>
<tr>
<td>590-710</td>
<td>Displays “XX=INVALID DATA” if the data is not the current measurement data or the voltage output data.</td>
</tr>
<tr>
<td>740</td>
<td>Displays “No data returned” if the number of data is zero.</td>
</tr>
<tr>
<td>770</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>780</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
### Sampling Measurements

To make sampling measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables Measurement Units</td>
<td>CN</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Disables Measurement Units</td>
<td>CL</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Filter ON/OFF</td>
<td>[FL]</td>
<td>mode,[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets Averaging Number</td>
<td>[AV]</td>
<td>number[,mode]</td>
</tr>
<tr>
<td>Sets Integration Time</td>
<td>[SIT]</td>
<td>type,time</td>
</tr>
<tr>
<td></td>
<td>[SLI]</td>
<td>type</td>
</tr>
<tr>
<td>Sets the timing parameters</td>
<td>MT</td>
<td>hold,interval,points</td>
</tr>
<tr>
<td>Source setup</td>
<td>MV</td>
<td>ch,range,base,bias[,Icomp]</td>
</tr>
<tr>
<td></td>
<td>MI</td>
<td>ch,range,base,bias[,Vcomp]</td>
</tr>
<tr>
<td></td>
<td>MP</td>
<td>ch,mode,base,bias[,Td,Tw,Tl,Tt,Tp,count] (^a)</td>
</tr>
<tr>
<td>Clears the sampling source</td>
<td>[M CC]</td>
<td>[ch[,ch ... [,ch[ch]] ... ]]</td>
</tr>
<tr>
<td>settings</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sets automatic abort condition</td>
<td>[MSC]</td>
<td>abort</td>
</tr>
<tr>
<td>Forces constant voltage</td>
<td>[DV]</td>
<td>chnum,range,output[,Icomp]</td>
</tr>
<tr>
<td>Forces constant current</td>
<td>[DI]</td>
<td>chnum,range,output[,Vcomp]</td>
</tr>
<tr>
<td>Sets VMU measurement mode</td>
<td>[VM]</td>
<td>chnum,mode</td>
</tr>
<tr>
<td>Sets voltage measurement</td>
<td>[RV]</td>
<td>chnum,range[,Rmode]</td>
</tr>
<tr>
<td>range</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**FLEX Command Programming**

**Sampling Measurements**

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sets current measurement range</td>
<td>[RI] chnum,range[,Rmode]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[RM ] chnum,mode[,rate]</td>
<td></td>
</tr>
<tr>
<td>Selects measurement mode</td>
<td>MM chnum[,chnum ..[,chnum] ..]</td>
<td>10,chnum[,chnum ..[,chnum] ..]</td>
</tr>
<tr>
<td>Sets SMU measurement mode</td>
<td>[CMM] chnum,mode</td>
<td></td>
</tr>
<tr>
<td>Executes measurement</td>
<td>XE</td>
<td></td>
</tr>
<tr>
<td>Reads measurement data</td>
<td>RMD? number_of_data</td>
<td></td>
</tr>
</tbody>
</table>

- a. If the MT command interval parameter is less than 2 msec, the AV and SLI commands are ignored.
- b. The Tp and count settings are effective for both PGU1 and PGU2. The latest value is effective for the output pulse.
- c. If the MT command interval parameter is less than 2 msec, the available number of chnums is 1.
FLEX Command Programming
 Sampling Measurements

A program example of a sampling measurement is shown below. This program executes the resistance measurements and prints the results on the screen.

10     ASSIGN @Hp415x TO 800
20     OPTION BASE 1
30     INTEGER High, Low, Range_v, Range_i, Nop
40     DIM C$[100]
50     High=1         !1:SMU1
60     Low=2          !2:SMU2
70     Hold=1         !Hold time (sec)
80     Interval=.5    !Sampling interval (sec)
90     Nop=11         !Number of sampling points
100    Vbase=0        !Base voltage for High
110    Vbias=10       !Bias voltage for High
120    Icomp=.001     !Current compliance
130    Vl=0           !Voltage for Low
140    Range_v=12    !12:20 V Limited Auto Ranging
150    Range_i=14    !14: 1uA Limited Auto Ranging
160    !
170    OUTPUT @Hp415x;"US"
180    OUTPUT @Hp415x;"FMT 1,1"     !Data w/Header, Index
190    OUTPUT @Hp415x;"CN ";High,Low
200    OUTPUT @Hp415x;"MSC 2"       !Stops for any abort condition
210    OUTPUT @Hp415x;"MCC"
220    OUTPUT @Hp415x;"MT ";Hold,Interval,Nop

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>50 to 150</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>170</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>180</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>190</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>200</td>
<td>Sets the automatic abort condition.</td>
</tr>
<tr>
<td>210</td>
<td>Clears the previous sampling setup.</td>
</tr>
<tr>
<td>220</td>
<td>Sets the sampling measurement timing parameters.</td>
</tr>
</tbody>
</table>
OUTPUT @Hp415x;"MM 10,";High !Sampling measurement
OUTPUT @Hp415x;"MV ";High,Range_v,Vbase,Vbias,Icomp
OUTPUT @Hp415x;"DV ";Low,Range_v,Vl,Icomp
OUTPUT @Hp415x;"RI ";High,Range_i
OUTPUT @Hp415x;"XE"
OUTPUT @Hp415x;"*OPC?"
Enter @Hp415x;C
OUTPUT @Hp415x;"CL"
OUTPUT @Hp415x;";SYST:ERR?"
Enter @Hp415x;C,C$
IF C<>0 THEN
PRINT "ERROR:”;C,C$
BEEP
DISP "Press Continue to read data. Or press Stop."
PAUSE
DISP ""
END IF
!

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>230</td>
<td>Sets the sampling measurement mode.</td>
</tr>
<tr>
<td>240</td>
<td>Sets the voltage source synchronized with the sampling measurements.</td>
</tr>
<tr>
<td>250</td>
<td>Forces the dc voltage to the Low terminal.</td>
</tr>
<tr>
<td>260</td>
<td>Sets the measurement range.</td>
</tr>
<tr>
<td>270 to 290</td>
<td>Executes the sampling measurements.</td>
</tr>
<tr>
<td>300</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>310 to 390</td>
<td>Checks to see if an error has occurred. If an error is detected, the error code and error message are displayed on the screen.</td>
</tr>
</tbody>
</table>
FLEX Command Programming

Sampling Measurements

410 ! Reading data
420 OUTPUT @Hp415x;"NUB?"
430 ENTER @Hp415x;C
440 IF C<>0 THEN
450 FOR N=1 TO C
460 OUTPUT @Hp415x;"RMD? 1"
470 ENTER @Hp415x USING ";,3A,A,A,13D,X";St$,Ch$,Md$,Mdata
480 IF Md$="p" THEN
490 PRINT "INDEX=";Mdata
500 ELSE
510 IF Md$="I" THEN
520 PRINT " R=";Vbias/Mdata;"(ohm), Status=";St$
530 ELSE
540 PRINT " XX=INVALID DATA, Status=";St$
550 END IF
560 END IF
570 NEXT N
580 ELSE
590 PRINT "STATUS: No data returned."
600 END IF
610 OUTPUT @Hp415x;":PAGE"
620 END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>420 to 430</td>
<td>Confirms the number of data stored in the output buffer.</td>
</tr>
<tr>
<td>460 to 530</td>
<td>Reads the measurement data, and displays the measurement data.</td>
</tr>
<tr>
<td>540</td>
<td>Displays &quot;XX=INVALID DATA&quot; if the data is not the index data or the current measurement data.</td>
</tr>
<tr>
<td>590</td>
<td>Displays &quot;No data returned&quot; if the number of data is zero.</td>
</tr>
<tr>
<td>610</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
## Quasi-static CV Measurements

To make quasi-static CV measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables measurement units</td>
<td>CN</td>
<td>[chnum ..., .chnum] ... ]</td>
</tr>
<tr>
<td>Disables measurement units</td>
<td>CL</td>
<td>[chnum ..., .chnum] ... ]</td>
</tr>
<tr>
<td>Turns the filter on/off</td>
<td>[FL]</td>
<td>onoff[,chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets the measurement mode</td>
<td>M M</td>
<td>13[,chnum]</td>
</tr>
<tr>
<td>Sets zero cancel</td>
<td>[QSZ]</td>
<td>onoff</td>
</tr>
<tr>
<td>Sets the leak measurement/leak compensation</td>
<td>[QSL]</td>
<td>data_onoff,compen_onoff</td>
</tr>
<tr>
<td>Sets the abort function</td>
<td>[QSM]</td>
<td>condition[,result]</td>
</tr>
<tr>
<td>Sets the measurement range</td>
<td>QSR</td>
<td>range</td>
</tr>
<tr>
<td>Sets the timing parameters</td>
<td>QST</td>
<td>cinteg,linteg,hold,delay1 [,delay2]</td>
</tr>
<tr>
<td>Sets the sweep source</td>
<td>QSV</td>
<td>chnum,swpmode,range,start,stop ,cvoltage,step[,lcomp]</td>
</tr>
<tr>
<td>Forces the constant source</td>
<td>DV</td>
<td>chnum,range,output[,lcomp]</td>
</tr>
<tr>
<td>Executes a measurement</td>
<td>XE</td>
<td></td>
</tr>
<tr>
<td>Reads the measurement data</td>
<td>RMD?</td>
<td>number_of_data</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Quasi-static CV Measurements

A program example of quasi-static CV measurement is shown below. This program measures the gate capacitance of MOSFET, and displays the data on the screen.

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path for controlling the 4155C/4156C.</td>
</tr>
<tr>
<td>20</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>30</td>
<td>Calculates the number of steps for the voltage sweep.</td>
</tr>
<tr>
<td>40</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>50 to 200</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>210</td>
<td>Sets the quasi-static CV measurement mode and its measurement channel. And closes the SMU input switches.</td>
</tr>
<tr>
<td>220</td>
<td>Enables leakage current measurement and compensation.</td>
</tr>
<tr>
<td>230</td>
<td>Enables measurement abort function for abnormal conditions.</td>
</tr>
<tr>
<td>240</td>
<td>Sets the measurement range.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Quasi-static CV Measurements

Line Number | Description
---|---
300 | Sets the integration time, hold time, and delay time.
310 to 330 | Sets the dc source output for the MOSFET drain, source, and substrate terminals.
340 | Sets the sweep source output for the MOSFET gate terminal.
350 to 470 | Executes the offset measurement, and enables the zero cancel. Before execute the offset measurement, DUT must be removed from the measurement terminals.
480 to 510 | Executes a quasi-static CV measurement.
520 | Opens the SMU input switches.
540 to 620 | Checks to see if an error has occurred. If an error is detected, the error code and error message are displayed on the screen.
### FLEX Command Programming
#### Quasi-static CV Measurements

630   !
640   ! Reading data
650 OUTPUT @Hp415x;"NUB?"
660 ENTER @Hp415x;A
670 IF A<>0 THEN
680 FOR I=1 TO A
690 OUTPUT @Hp415x;"RMD? 1"
700 ENTER @Hp415x USING ",3A,X,A,13D,X";A$;B$;Mdata
710 IF B$="1" THEN
720 PRINT "INDEX=";INT((I+2)/3)
730 PRINT " Leak=";Mdata*1.E+12;"(pA), Status=";A$
740 ELSE
750 IF B$="C" THEN
760 PRINT " Cap=";Mdata*1.E+12;"(pF), Status=";A$
770 ELSE
780 IF B$="v" THEN
790 PRINT " Vg=";Mdata;"(V), Status=";A$
800 ELSE
810 PRINT " XX=INVALID DATA, Status=";A$
820 END IF
830 END IF
840 END IF
850 NEXT I
860 ELSE
870 PRINT "STATUS: No data returned."
880 END IF
890 OUTPUT @Hp415x;":PAGE"
900 END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>650 to 660</td>
<td>Confirms the number of data stored in the output buffer.</td>
</tr>
<tr>
<td>690 to 700</td>
<td>Reads the measurement data.</td>
</tr>
<tr>
<td>710 to 730</td>
<td>Displays the data index and the leakage current measurement data.</td>
</tr>
<tr>
<td>750 to 760</td>
<td>Displays the capacitance measurement data.</td>
</tr>
<tr>
<td>780 to 790</td>
<td>Displays the source output voltage.</td>
</tr>
<tr>
<td>810</td>
<td>Displays “XX=INVALID DATA” if the data is not the current measurement data, capacitance measurement data, or the voltage output data.</td>
</tr>
<tr>
<td>870</td>
<td>Displays “No data returned” if the number of data is zero.</td>
</tr>
<tr>
<td>890</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode) if the exclamation mark of the line is removed.</td>
</tr>
</tbody>
</table>
Linear Search Measurements

To make linear search measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables measurement units</td>
<td>CN</td>
<td>chnum ... [,chnum] ...</td>
</tr>
<tr>
<td>Disables measurement units</td>
<td>CL</td>
<td>chnum ... [,chnum] ...</td>
</tr>
<tr>
<td>Sets filter on/off</td>
<td>[FL]</td>
<td>onoff[,chnum ... [,chnum] ...</td>
</tr>
<tr>
<td>Sets measurement mode</td>
<td>MM</td>
<td>14</td>
</tr>
<tr>
<td>Selects output data</td>
<td>[LSVM]</td>
<td>output_data</td>
</tr>
<tr>
<td>Sets timing parameters</td>
<td>[LSTM]</td>
<td>holdtime, delaytime</td>
</tr>
<tr>
<td>Sets the abort function</td>
<td>[WM]</td>
<td>abort[,post]</td>
</tr>
<tr>
<td>Sets current search or voltage search condition</td>
<td>LGI or LGV</td>
<td>chnum, mode, range, target</td>
</tr>
<tr>
<td>Sets voltage source or current source</td>
<td>LSV or LSI</td>
<td>chnum, range, start, stop, step [,comp]</td>
</tr>
<tr>
<td>Sets synchronous voltage source or current source</td>
<td>[LSSV] or [LSSI]</td>
<td>chnum, polarity, offset[,comp]</td>
</tr>
<tr>
<td>Forces constant source</td>
<td>DV or DI</td>
<td>chnum, range, output[,comp]</td>
</tr>
<tr>
<td>Executes measurement</td>
<td>XE</td>
<td></td>
</tr>
<tr>
<td>Reads measurement data</td>
<td>RM D?</td>
<td>number_of_data</td>
</tr>
</tbody>
</table>

Before sending the LSSI command, the LSI command must be sent. Do not use the LSV command.

Before sending the LSSV command, the LSV command must be sent. Do not use the LSI command.
**FLEX Command Programming**

**Linear Search Measurements**

A program example of a linear search measurement is shown below. This program executes MOSFET threshold voltage measurement, and displays the result data on the screen.

```
10     ASSIGN @Hp415x TO 800
20     OPTION BASE 1
30     INTEGER Gate,Drain,Source,Sub,Rm,Ro,Judge,Posneg
40     DIM A$(100)
50     Drain=1        ! 1: SMU1
60     Gate=2         ! 2: SMU2
70     Sub=3          ! 3: SMU3
80     Source=4       ! 4: SMU4
90     V_b=.5         ! Search start voltage (V)
100    V_e=1          ! Search stop voltage (V)
110    V_s=.01        ! Search step voltage (V)
120    D_comp=.01     ! Drain compliance (A)
130    G_comp=.01     ! Gate compliance (A)
140    Rm=14          ! 14: 1 uA limited auto ranging
150    Ro=11          ! 11: 2 V limited auto ranging
160    Hold=0         ! Hold time (s)
170    Delay=0        ! Delay time (s)
180    Judge=1        ! 1: Meas >= Target
190    Target=1.E-6   ! Target value
200    Posneg=1       ! 1: Positive, 0: Negative
210    Offset=0       ! Offset voltage (V)
220   
230    OUTPUT @Hp415x;"US"
240    OUTPUT @Hp415x;"FMT 1,1"! Data w/Header,Source_data
250    OUTPUT @Hp415x;"MM 14"  ! Linear search
260    OUTPUT @Hp415x;"WM 2,1" ! Stops for any abnormal condition
270    OUTPUT @Hp415x;"LSVM 1" ! 0: Target, 1: w/data
```

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path for controlling the 4155C/4156C.</td>
</tr>
<tr>
<td>50 to 210</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>230</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>240</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>250</td>
<td>Sets the linear search measurement mode.</td>
</tr>
<tr>
<td>260</td>
<td>Enables measurement abort function for abnormal conditions.</td>
</tr>
<tr>
<td>270</td>
<td>Sets all result data output mode.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Linear Search Measurements

280    OUTPUT @Hp415x;"LSTM ";Hold,Delay
290    OUTPUT @Hp415x;"LGI ";Drain,Judge,Rm,Target
300    !
310    OUTPUT @Hp415x;"CN ";Source,Drain,Gate,Sub
320    OUTPUT @Hp415x;"DV ";Source,Ro,0,.01
330    OUTPUT @Hp415x;"DV ";Sub,Ro,0,.01
340    OUTPUT @Hp415x;"LSV ";Drain,Ro,V_b,V_e,V_s,D_comp
350    OUTPUT @Hp415x;"LSSV ";Gate,Posneg,Offset,G_comp
360    OUTPUT @Hp415x;"XE"
370    OUTPUT @Hp415x;"*OPC?"
380    ENTER @Hp415x;A
390    OUTPUT @Hp415x;"CL"
400    OUTPUT @Hp415x;":SYST:ERR?"
410    ENTER @Hp415x;A,A$
420    IF A<>0 THEN
430      PRINT "ERROR:";A,A$
440      BEEP
450      DISP "Press press Continue to read data. Or press Stop."
460      PAUSE
470      DISP ""
480    END IF
490    !

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>280</td>
<td>Sets hold time and delay time.</td>
</tr>
<tr>
<td>290</td>
<td>Sets current search condition.</td>
</tr>
<tr>
<td>310</td>
<td>Closes the SMU input switches.</td>
</tr>
<tr>
<td>320 to 330</td>
<td>Sets the dc source output for the MOSFET source and substrate terminals.</td>
</tr>
<tr>
<td>340</td>
<td>Sets the sweep source output for the MOSFET drain terminal.</td>
</tr>
<tr>
<td>350</td>
<td>Sets the synchronous sweep source output for the MOSFET gate terminal.</td>
</tr>
<tr>
<td>360 to 380</td>
<td>Executes a linear search measurement.</td>
</tr>
<tr>
<td>390</td>
<td>Opens the SMU input switches.</td>
</tr>
<tr>
<td>400 to 480</td>
<td>Checks to see if an error has occurred. If an error is detected, the error code and error message are displayed on the screen.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Linear Search Measurements

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500 to 520</td>
<td>Confirms the number of data stored in the output buffer.</td>
</tr>
<tr>
<td>560 to 570</td>
<td>Reads the measurement data.</td>
</tr>
<tr>
<td>580 to 780</td>
<td>Displays the raw data and the search result data on the screen.</td>
</tr>
<tr>
<td>830</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode) if the exclamation mark of the line is removed.</td>
</tr>
</tbody>
</table>
Binary Search Measurements

To make binary search measurements, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables measurement units</td>
<td>CN</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Disables measurement units</td>
<td>CL</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets filter on/off</td>
<td>[FL]</td>
<td>onoff,[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets measurement mode</td>
<td>MM</td>
<td>15</td>
</tr>
<tr>
<td>Selects output data</td>
<td>[BSVM ]</td>
<td>output_data</td>
</tr>
<tr>
<td>Sets timing parameters</td>
<td>[BST ]</td>
<td>holdtime,delaytime</td>
</tr>
<tr>
<td>Sets source control mode</td>
<td>BSM</td>
<td>mode</td>
</tr>
<tr>
<td>Sets current search or voltage search condition</td>
<td>BGI or BGV</td>
<td>chnum,mode,condition,range, target</td>
</tr>
<tr>
<td>Sets voltage source or current source</td>
<td>BSV or BSI</td>
<td>chnum,range,start,stop[,comp]</td>
</tr>
<tr>
<td>Sets synchronous voltage source or current source</td>
<td>[BSSV ] or [BSSI]</td>
<td>chnum,polarity,offset[,comp]</td>
</tr>
<tr>
<td>Forces constant source</td>
<td>DV or DI</td>
<td>chnum,range,output[,comp]</td>
</tr>
<tr>
<td>Executes measurement</td>
<td>XE</td>
<td></td>
</tr>
<tr>
<td>Reads measurement data</td>
<td>RM D?</td>
<td>number_of_data</td>
</tr>
</tbody>
</table>

Before sending the BSSI command, the BSI command must be sent. Do not use the BSV command.

Before sending the BSSV command, the BSV command must be entered. Do not use the BSI command.
### FLEX Command Programming

#### Binary Search Measurements

A program example of a binary search measurement is shown below. This program executes MOSFET threshold voltage measurement, and prints result data on the screen.

```plaintext
10     ASSIGN @Hp415x TO 800
20     OPTION BASE 1
30     INTEGER Gate,Drain,Source,Sub,Nub,I
40     INTEGER V_n,Rm,Ro,Posneg,Offset
50     DIM C$[50]
60     Source=1       ! 1: SMU1
70     Drain=2        ! 2: SMU2
80     Gate=3         ! 3: SMU3
90     Sub=4          ! 4: SMU4
100    V_b=.5         ! Search start voltage (V)
110    V_e=1          ! Search stop voltage (V)
120    D_comp=.01     ! Drain compliance (A)
130    G_comp=.01     ! Gate compliance (A)
140    Rm=14          ! 14: 1 uA limited auto ranging
150    Ro=11          ! 11: 2 V limited auto ranging
160    Hold=0         ! Hold time (s)
170    Delay=0        ! Delay time (s)
180    Mode=0         ! 0: Limit, 1: Repeat
190    Judge=1.E-8    ! Limit in A. for Limit mode.
200   !Judge=10       ! No of repeats. for Repeat mode.
210    Target=1.E-6   ! Target value
220    Posneg=1       ! 1: Positive, 0: Negative
230    Offset=0       ! Offset voltage (V)
240   !
250   OUTPUT @Hp415x;"US"
260   OUTPUT @Hp415x;"FMT 1"  ! ASCII w/header
270   OUTPUT @Hp415x;"MM 15"  ! Binary search
280   OUTPUT @Hp415x;"BSM 1"  ! 1: SAFE
290   OUTPUT @Hp415x;"BSVM 1" ! 0: Target, 1: w/data
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path for controlling the 4155C/4156C.</td>
</tr>
<tr>
<td>60 to 230</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>270</td>
<td>Sets the binary search measurement mode.</td>
</tr>
<tr>
<td>280</td>
<td>Sets source control mode to cautious mode.</td>
</tr>
<tr>
<td>290</td>
<td>Sets all result data output mode.</td>
</tr>
</tbody>
</table>
### FLEX Command Programming

#### Binary Search Measurements

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>300</td>
<td>OUTPUT @Hp415x;&quot;BST &quot;;Hold,Delay</td>
</tr>
<tr>
<td>310</td>
<td>OUTPUT @Hp415x;&quot;BGI &quot;;Drain,Mode,Judge,Rm,Target</td>
</tr>
<tr>
<td>320</td>
<td>!</td>
</tr>
<tr>
<td>330</td>
<td>OUTPUT @Hp415x;&quot;CN &quot;;Source,Drain,Gate,Sub</td>
</tr>
<tr>
<td>340</td>
<td>OUTPUT @Hp415x;&quot;DV &quot;;Source,Ro,0,.01</td>
</tr>
<tr>
<td>350</td>
<td>OUTPUT @Hp415x;&quot;DV &quot;;Sub,Ro,0,.01</td>
</tr>
<tr>
<td>360</td>
<td>OUTPUT @Hp415x;&quot;BSV &quot;;Drain,Ro,V_b,V_e,D_comp</td>
</tr>
<tr>
<td>370</td>
<td>OUTPUT @Hp415x;&quot;BSSV &quot;;Gate,Posneg,Offset,G_comp</td>
</tr>
<tr>
<td>380</td>
<td>OUTPUT @Hp415x;&quot;XE&quot;</td>
</tr>
<tr>
<td>390</td>
<td>OUTPUT @Hp415x;&quot;*OPC?&quot;</td>
</tr>
<tr>
<td>400</td>
<td>ENTER @Hp415x;B</td>
</tr>
<tr>
<td>410</td>
<td>OUTPUT @Hp415x;&quot;CL&quot;</td>
</tr>
<tr>
<td>420</td>
<td>!</td>
</tr>
<tr>
<td>430</td>
<td>OUTPUT @Hp415x;&quot;:SYST:ERR?&quot;</td>
</tr>
<tr>
<td>440</td>
<td>ENTER @Hp415x;C,C$</td>
</tr>
<tr>
<td>450</td>
<td>IF C=0 THEN</td>
</tr>
<tr>
<td>460</td>
<td>GOTO 520</td>
</tr>
<tr>
<td>470</td>
<td>ELSE</td>
</tr>
<tr>
<td>480</td>
<td>PRINT C,C$</td>
</tr>
<tr>
<td>490</td>
<td>GOTO 730</td>
</tr>
<tr>
<td>500</td>
<td>END IF</td>
</tr>
<tr>
<td>510</td>
<td>!</td>
</tr>
</tbody>
</table>

**Line Number**  **Description**  
300 Sets the hold time and delay time.  
310 Sets the current search condition.  
330 Closes the SMU input switches.  
340 to 350 Sets the dc source output for the MOSFET source and substrate.  
360 Sets the binary search output for the MOSFET drain terminal.  
370 Sets the synchronous binary search output for the gate terminal.  
380 to 400 Executes a binary search measurement.  
410 Opens the SMU input switches.  
430 to 500 Checks to see if an error has occurred. If an error is detected, the error code and error message are displayed on the screen.
## Binary Search Measurements

```plaintext
520 OUTPUT @Hp415x;"NUB?"
530 ENTER @Hp415x;Nub
540 IF Nub<>INT((Nub-3)/2)*2+3 THEN
550 DISP Nub;"data returned. No. of data must be N*2+3."
560 GOTO 730
570 ELSE
580 OUTPUT @Hp415x;"RMD?"
590 PRINT "Raw data **************************"
600 FOR I=1 TO (Nub-3)/2
610 ENTER @Hp415x USING "#,5X,13D,6X,13D,X";A,B
620 PRINT I;"Vg=";A;", Id=";B
630 NEXT I
640 PRINT "Search result ***************"
650 ENTER @Hp415x USING "#,5X,13D,6X,13D,6X,13D,X";St,A,B
660 IF St=0 THEN
670 PRINT "Vth=";A;"V at Id=";B*1.E+6;"uA"
680 PRINT "Search target=";Target*1.E+6;"uA +/";Judge*1.E+6;"uA"
690 ELSE
700 PRINT "No target found."
710 END IF
720 END IF
730 !OUTPUT @Hp415x;":PAGE"
740 END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>520 to 530</td>
<td>Confirms the number of data stored in the output buffer.</td>
</tr>
<tr>
<td>580</td>
<td>Reads the all measurement data.</td>
</tr>
<tr>
<td>590 to 630</td>
<td>Displays the raw data on the screen.</td>
</tr>
<tr>
<td>640 to 680</td>
<td>Displays the search result data on the screen.</td>
</tr>
<tr>
<td>700</td>
<td>Displays “No target found” if the status code of the search status data is not 0.</td>
</tr>
<tr>
<td>730</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode) if the exclamation mark of the line is removed.</td>
</tr>
</tbody>
</table>
**Stress Force**

To utilize the stress force function, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables Measurement Units</td>
<td>CN</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Disables Measurement Units</td>
<td>CL</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets PGU output impedance</td>
<td>POR</td>
<td>chnum,impedance</td>
</tr>
<tr>
<td>Sets stress mode/stress time</td>
<td>STT</td>
<td>hold,mode[,count[,period]] a</td>
</tr>
<tr>
<td>Stress source setup</td>
<td>STV</td>
<td>source,chnum,range,base,stress [,Icomp]</td>
</tr>
<tr>
<td></td>
<td>STI</td>
<td>source,chnum,range,base,stress [,Vcomp]</td>
</tr>
<tr>
<td></td>
<td>STP</td>
<td>source,chnum,mode,base,stress [,Td[,Tw[,Tl[,Tt]]]]</td>
</tr>
<tr>
<td>Clears stress source setup</td>
<td>STC</td>
<td>[source[,source[,source[,source]]]]</td>
</tr>
<tr>
<td>Sets automatic abort condition</td>
<td>STM b</td>
<td>abort</td>
</tr>
<tr>
<td>Forces constant voltage</td>
<td>[DV]</td>
<td>chnum,range,output[,Icomp]</td>
</tr>
<tr>
<td>Forces constant current</td>
<td>[DI]</td>
<td>chnum,range,output[,Vcomp]</td>
</tr>
<tr>
<td>Selects measurement mode</td>
<td>MM</td>
<td>11</td>
</tr>
<tr>
<td>Executes measurement</td>
<td>XE</td>
<td></td>
</tr>
</tbody>
</table>

a. The count and period settings are effective for both PGU1 and PGU2. The latest setting is effective for the output pulse.

b. The automatic abort function is available when the STT command sets the freerun pulse stress mode or the pulse count stress mode. For the pulse count stress mode, the pulse output must be more than 10 sec (count × period > 10 sec) to use the automatic abort function.
FLEX Command Programming
Stress Force

The stress force starts with the XE command and stops when the STT command setting or the STM command setting is satisfied. To stop the stress force immediately, enter the AB command.

A program example of stress force is shown below. This program forces dc stress using SMU and pulse stress using PGU. It does not execute the measurements.

```
10 ASSIGN @Hp415x TO 800
20 !
30 INTEGER Drain,Sub,Source,G1,G2,Mmode,Status
40 INTEGER Impedance,Smode,Pmode,Range_v,B,C
50 DIM B$[50]
60 !
70 Drain=1        !1:SMU1
80 Sub=2          !2:SMU2
90 Source=26      !26:GNDU
100 G1=3           !3:SMU3
110 G2=27          !27:PGU1
120 Impedance=0    !0:Low impedance. 1:50 ohm
130 Abort=2        !2:Selects all abort condition
140 Range_v=12     !12:20 V Limited Auto Ranging
150 Base=0         !Stress base voltage
160 Bias=10        !Dc stress voltage
170 Icomp_g=.01    !Current compliance for gate (G1)
180 Smode=1        !Stress mode of PGU. 0:dc, 1:Pulse
190 Pulse=10       !Pulse stress voltage
200 Td=.03         !Pulse delay (sec)
210 Tw=.05         !Pulse width (sec)
220 Tl=.001        !Pulse leading time (sec)
230 Tt=.001        !Pulse trailing time (sec)
240 Tp=.1          !Pulse period (sec)
250 Hold=0         !Hold time
260 Pmode=1        !Pulse mode. 0:Free, 1:Count, 2:Duration
270 Count=1000     !Pulse count
280 Vd=5           !Drain Voltage
290 Vsub=0         !Substrate Voltage
300 Icomp=.1       !Current compliance
310 Mmode=11       !11:Stress Force
320 !
330 OUTPUT @Hp415x;"US"
340 OUTPUT @Hp415x;"STC"
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>70 to 310</td>
<td>Sets the parameters.</td>
</tr>
<tr>
<td>330</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>340</td>
<td>Clears the previous stress condition.</td>
</tr>
</tbody>
</table>
### Line Number Description

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>350</td>
<td>Enables the stress and bias sources.</td>
</tr>
<tr>
<td>360</td>
<td>Sets the output impedance of the PGU.</td>
</tr>
<tr>
<td>370</td>
<td>Sets the automatic abort condition.</td>
</tr>
<tr>
<td>380</td>
<td>Sets the stress mode and stress time.</td>
</tr>
<tr>
<td>390</td>
<td>Sets the dc voltage stress source.</td>
</tr>
<tr>
<td>400</td>
<td>Sets the pulse stress source.</td>
</tr>
<tr>
<td>410 to 420</td>
<td>Forces the dc voltage to Drain and Sub.</td>
</tr>
<tr>
<td>430</td>
<td>Sets the stress force mode.</td>
</tr>
<tr>
<td>440 to 460</td>
<td>Forces the stress set by the STV and STP commands.</td>
</tr>
<tr>
<td>480 to 500</td>
<td>Checks status of stress force completion</td>
</tr>
<tr>
<td>510</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>520 to 540</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>560</td>
<td>Returns to the 4155C/4156C default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>

---

Controlling PGU

To control PGU, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enables Measurement Units</td>
<td>CN</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Disables Measurement Units</td>
<td>CL</td>
<td>[chnum ... [,chnum] ... ]</td>
</tr>
<tr>
<td>Sets PGU output impedance</td>
<td>POR</td>
<td>chnum,impedance</td>
</tr>
<tr>
<td>Sets output mode</td>
<td>SPG</td>
<td>chnum,mode[,,pulse,Td,Tw,Tl,Tt,Tp,Pc] a</td>
</tr>
<tr>
<td>Starts PGU output force</td>
<td>SRP</td>
<td></td>
</tr>
<tr>
<td>Stops PGU output force</td>
<td>SPP</td>
<td></td>
</tr>
<tr>
<td>Forces constant voltage</td>
<td>[DV ]</td>
<td>chnum,range,output[,Icomp]</td>
</tr>
<tr>
<td>Forces constant current</td>
<td>[DI ]</td>
<td>chnum,range,output[,Vcomp]</td>
</tr>
</tbody>
</table>

a. The Tp and Pc settings are effective for both PGU1 and PGU2. The latest setting is effective for the output pulse.

The PGU output can be controlled by the SPG/SPR/SPP commands. It is not controlled by the XE command. You can use the PGU control commands, regardless of the measurement mode (MM command setting).

To force PGU output, use the SRP command.

To stop PGU output, use the SPP command. The PGU output will then go to the base value. To force 0 V, use the DV command.
A program example of PGU output control is shown below. This program forces constant voltage by using PGU1, forces voltage pulse by using PGU2, and executes the high-speed spot measurements.

```
10 ASSIGN @Hp415x TO 800
20 !
30 INTEGER Fmt,Drain,Source,Gate,Sub,M1,M2,Pc,A
40 INTEGER Range_v,Range_i
50 DIM A$[50]
60 !
70 Fmt=1          !1:ASCII with header <LF^EOI>
80 Drain=1        !1:SMU1
90 Source=26      !26:GNDU
100 Gate=27        !27:PGU1
110 Sub=28         !28:PGU2
120 Impedance=0    !0:Low impedance, 1:50 ohm
130 M1=1           !Output mode for Sub 1:constant
140 M2=2           !Output mode for Gate 2:pulse
150 B1=0           !Constant voltage for Sub
160 B2=2.5         !Pulse base voltage for Gate
170 Out=5          !Pulse voltage for Gate
180 Td=.03         !Pulse delay (sec)
190 Tw=.05         !Pulse width (sec)
200 Ti=.001        !Pulse leading time (sec)
210 Tt=.001        !Pulse trailing time (sec)
220 Tp=.1          !Pulse period (sec)
230 Pc=0           !Pulse count 0: free run
240 Range_v=12     !12: 20 V Limited Auto Ranging
250 Vd=5           !Drain Voltage
260 Vsub=0         !Substrate Voltage
270 Icomp=.1       !Current compliance
280 Range_i=14     !14: 1 uA Limited Auto Ranging
290 !
300 OUTPUT @Hp415x;"US"
310 OUTPUT @Hp415x;"FMT ";Fmt
320 OUTPUT @Hp415x;"CN ";Drain,Gate,Sub
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>70 to 280</td>
<td>Sets the parameters.</td>
</tr>
<tr>
<td>300</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>310</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>320</td>
<td>Enables the PGUs and measurement units.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Controlling PGU

330  OUTPUT @Hp415x;"POR ";Sub,Impedance
340  OUTPUT @Hp415x;"POR ";Gate,Impedance
350  OUTPUT @Hp415x;"SPG ";Sub,M1,B1
360  OUTPUT @Hp415x;"SPG ";Gate,M2,B2,Out,Td,Tl,Tt,Tp,Pc
370  OUTPUT @Hp415x;"DV ";Drain,Range_v,Vd,Icomp
380  OUTPUT @Hp415x;"SRP"
390  !
400  OUTPUT @Hp415x;":SYST:ERR?"
410  ENTER @Hp415x;A,A$
420  IF A=0 THEN
430    FOR I=1 TO 10
440      OUTPUT @Hp415x;"TI? ";Drain,Range_i
450      ENTER @Hp415x USING ";","5X,13D,X";Mdata
460      PRINT "Id(A)=";Mdata
470    WAIT 1
480  NEXT I
490  END IF
500  !
510  OUTPUT @Hp415x;"SPP"
520  OUTPUT @Hp415x;"DV ";Gate,0,0
530  OUTPUT @Hp415x;"DV ";Drain,0,0
540  PRINT A,A$
550  OUTPUT @Hp415x;"CL"
560  OUTPUT @Hp415x;":PAGE"
570  END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>330 to 340</td>
<td>Sets the PGU output impedance.</td>
</tr>
<tr>
<td>350</td>
<td>Sets the PGU setup for Substrate.</td>
</tr>
<tr>
<td>360</td>
<td>Sets the PGU setup for Gate.</td>
</tr>
<tr>
<td>370</td>
<td>Forces the dc voltage to Drain.</td>
</tr>
<tr>
<td>380</td>
<td>Forces the PGU output.</td>
</tr>
<tr>
<td>400 to 410</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>440 to 460</td>
<td>Measures the drain current and prints the results on the screen.</td>
</tr>
<tr>
<td>510</td>
<td>Stops the PGU pulse output.</td>
</tr>
<tr>
<td>520 to 530</td>
<td>Forces 0 V to Gate and Drain.</td>
</tr>
<tr>
<td>540</td>
<td>Prints an error code and error message on the screen.</td>
</tr>
<tr>
<td>550</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>560</td>
<td>Returns to the 4155C/4156C default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
# Using Program Memory

Storing and executing measurement programs from internal memory improves measurement speed. The following commands are available for use in program memory.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stores the command into program</td>
<td>ST, END</td>
<td>ST prog No.;command[ ... ];command].. END</td>
</tr>
</tbody>
</table>
| memory                            |              | ST prog No. [command] :
|                                   |              | [command] END                                                               |
| Scratches the program             | SCR          | [prog No.]                                                                  |
| Gets a list of programs or a      | LST?         | [prog No.]                                                                  |
| specific program listing          |              |                                                                            |
| Executes specified programs       | DO           | prog No. [,prog No. ... [,prog No.] ... ]                                  |
| Executes programs sequentially    | RU           | start, stop                                                                 |
| Pauses command execution or       | PA           | [wait time]                                                                 |
| internal memory program execution |              |                                                                            |

The program memory can store a maximum of 255 programs (a maximum of 100 KB).

The program memory is available when the 4155C/4156C is in the FLEX command control mode. The internal memory programs are deleted when the US or US42 command is executed.
FLEX Command Programming
Using Program Memory

The internal memory does not provide for error checking, so programs must be complete and free of errors before they are stored.

If the program being stored makes changes to the present measurement setup, verify that these changes are correct and compatible with the present setup before storing the FLEX commands in the program memory.

If the program you will be storing executes a measurement, verify the program is free of errors and runs correctly before storing it in the program memory.

Other notes:

1. Invalid commands in the internal memory program:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AB</td>
<td>Ach</td>
</tr>
<tr>
<td>CA</td>
<td>Close</td>
</tr>
<tr>
<td>CM</td>
<td>*CAL?</td>
</tr>
<tr>
<td>*CAL?</td>
<td>*CLS?</td>
</tr>
<tr>
<td>END</td>
<td>Err?</td>
</tr>
<tr>
<td>*ESE?</td>
<td></td>
</tr>
<tr>
<td>*IDN?</td>
<td>Lop?</td>
</tr>
<tr>
<td>LST?</td>
<td>*LRN?</td>
</tr>
<tr>
<td>NUB?</td>
<td>Open</td>
</tr>
<tr>
<td>OPN?</td>
<td>*OPC (?)</td>
</tr>
<tr>
<td>OPT?</td>
<td>Prn</td>
</tr>
<tr>
<td>PRN</td>
<td>Qsz/Qsz?</td>
</tr>
<tr>
<td>Rcv</td>
<td>Rd?</td>
</tr>
<tr>
<td>RMD?</td>
<td>Ru</td>
</tr>
<tr>
<td>*RST</td>
<td></td>
</tr>
<tr>
<td>Scr</td>
<td>Sdk</td>
</tr>
<tr>
<td>Spl</td>
<td>Spr</td>
</tr>
<tr>
<td>ST</td>
<td></td>
</tr>
<tr>
<td>*SRE?</td>
<td>*Stb?</td>
</tr>
<tr>
<td>:Syst:Err?</td>
<td></td>
</tr>
<tr>
<td>*Tst?</td>
<td>Unt?</td>
</tr>
<tr>
<td>Us</td>
<td>Us42</td>
</tr>
<tr>
<td>Wnu?</td>
<td>Wr</td>
</tr>
<tr>
<td>*Wai</td>
<td></td>
</tr>
</tbody>
</table>

2. Command parameters:

When entering FLEX commands in internal memory, some optional parameters are required. You must specify both the necessary command parameters and these optional parameters. For more information regarding necessary parameters, refer to Chapter 1 of the GPIB Command Reference.

3. For 1 channel pulsed spot, pulsed sweep, and staircase sweep with pulsed bias measurements:

Multi-channel measurements are available when the PT command priority parameter is set to "wait for measurement completion". If the parameter is set to "keep pulse width", only one measurement channel is available.

If you change the priority parameter value and do not change the MM command parameter, the returned measurement data is available only for the first channel defined in the MM command. The data for the other channels will not be valid.

4. For sampling measurements:

Multi-channel measurements are available. If the sampling interval is less than 2 msec, only one measurement channel is available.

If you change the sampling interval to a value less than 2 msec, the returned measurement data is available only for the first channel defined in the MM command. The data for the other channels will not be valid.
5. For synchronous sweep measurements:
   The secondary sweep channel must be defined after the primary sweep channel. Enter the WSI/WSV command after the WI/WV or PWI/PWV command.

6. For PGU pulse output:
   If you use two PGUs, set the pulse period and pulse count parameters for the STT, SPG or MP command carefully. If you enter the command to change these parameter values, the previous settings of the pulse timing parameters may become invalid.
   In the internal program memory, the freerun pulse output is not available. Do not set the following command parameters.
   - STT command: mode=0
   - SPG command: P C=0

7. For VMU differential voltage measurements:
   Select the measurement unit (VMU1 or VMU2) carefully. Differential voltage measurements use the measurement range defined for the specified measurement unit.

8. CL command:
   When executing a program from internal memory, the CL command disables the unit in the HIGH VOLTAGE state (forcing more than ±40 V, or the voltage compliance set to more than ±40 V). To prevent the unit output switch from electrical damage, enter the DV command to lower the output voltage to 0 V or less than 40 V, before the CL command.

9. DV command:
   When executing a program from internal memory, the DV command is available for the unit which is in the output disable state by the CL command. This may occur the over current on the SMU. Use the DV command for the unit in the output enable state by the CN command.

10. Interlock circuit:
    The internal memory program cannot be executed if the interlock circuit is open. To execute the internal memory program, close the interlock circuit.
FLEX Command Programming
Using Program Memory

A program example using the internal program memory is shown below. This program does the following:

- enters a high-speed spot measurement program in program memory 1.
- enters a pulsed spot measurement program in program memory 2.
- prints the internal memory program listing on the screen.
- executes the internal memory program 1 and 2.
- prints the measurement results on the screen.

```plaintext
10     ASSIGN @Hp415x TO 800
20     OPTION BASE 1
30     INTEGER Fmt,Source,Gate,Drain,Sub,Mem,Err
40     INTEGER Vrange,Arange,Irange
50     Fmt=1            !ASCII with header <LF^EOI>
60     Source=1         !SMU1
70     Drain=2          !SMU2
80     Gate=3           !SMU3
90     Sub=4            !SMU4
100    Vrange=12        !12:20 V Limited auto ranging
110    Arange=0         ! 0:Auto ranging
120    Irange=14        !14:1 uA Limited auto ranging
130    Vg=3             !Gate voltage (V)
140    Vd=5             !Drain voltage (V)
150    Vaub=0           !Substrate voltage (V)
160    Vs=0             !Source voltage (V)
170    Icomp=.1         !Current compliance (A)
180    Icomp_g=.01      !Current compliance for Gate (A)
190    Vg_b=0           !Gate pulse base voltage (V)
200    Hold=0           !Hold time of Gate pulse (sec)
210    Width=.1         !Pulse width of Gate pulse (sec)
220    Period=.2        !Pulse period of Gate pulse (sec)
230    !
240    OUTPUT @Hp415x;"US"
250    OUTPUT @Hp415x;"FMT ";Fmt
260    !
270    Mem=1            !High-speed spot measurement
280    OUTPUT @Hp415x;"ST ";Mem
290    OUTPUT @Hp415x;"CN ";Gate,Drain,Source,Sub
300    OUTPUT @Hp415x;"DV ";Source,Arange,Vs,Icomp
310    OUTPUT @Hp415x;"DV ";Sub,Arange,Vsub,Icomp
320    OUTPUT @Hp415x;"DV ";Drain,Vrange,Vd,Icomp
330    OUTPUT @Hp415x;"DV ";Gate,Vrange,Vg,Icomp_g
340    OUTPUT @Hp415x;"TI ";Drain,Irange
350    OUTPUT @Hp415x;"CL"
360    OUTPUT @Hp415x;"END"
```

<table>
<thead>
<tr>
<th>Line No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>50 to 220</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>240</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>250</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>270 to 360</td>
<td>Stores the high-speed spot measurement program in the memory 1.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Using Program Memory

<table>
<thead>
<tr>
<th>Line No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>270 to 360</td>
<td>Stores the high-speed spot measurement program in the memory 1.</td>
</tr>
<tr>
<td>390 to 500</td>
<td>Stores the 1 ch pulsed spot measurement program in the memory 2.</td>
</tr>
<tr>
<td>530</td>
<td>Executes the internal program memory.</td>
</tr>
<tr>
<td>540 to 570</td>
<td>Reads the measurement result and prints the data on the screen.</td>
</tr>
<tr>
<td>590</td>
<td>Returns to the 4155C/4156C default control mode (SCPI mode).</td>
</tr>
<tr>
<td>620 to 710</td>
<td>Reads the internal memory program and prints the program listing.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Using Enhanced Sweep Stop Function

Using Enhanced Sweep Stop Function

The following example program uses the ESC command which defines the enhanced stop condition and enables the function. The program executes a MOSFET threshold voltage measurement, and displays the measurement result data on the screen. If the stop condition is true during the measurement, the sweep stops and the data status on the screen displays an “E” (sweep stopped). This function is available only for the sweep measurement.

10 ASSIGN @Hp415x TO 800
20 OPTION BASE 1
30 INTEGER Gate, Source, Drain, Sub
40 INTEGER V_n, Rm, Ro, Swp, Var1, Nub
50 REAL Vg(101)
60 Source=1       ! 1: SMU1
70 Drain=2        ! 2: SMU2
80 Gate=3         ! 3: SMU3
90 Sub=4          ! 4: SMU4
100 V_b=0          ! Sweep start voltage (V)
110 V_e=3          ! Sweep stop voltage (V)
120 V_n=101        ! Number of sweep steps
130 G_comp=.01     ! Current compliance (A) for Gate
140 D_comp=.1      ! Current compliance (A) for Drain
150 Rm=13          ! 13: 100 nA limited auto ranging
160 Ro=0           ! 0: Auto ranging
170 Swp=1          ! Linear single sweep
180 Ta=.1          ! Tgt of \([D(N+1)-D(N)]/D(N)\) value
190 Tb=5.E-7       ! Tgt of measured value
200 !
210 OUTPUT @Hp415x;"US"
220 OUTPUT @Hp415x;"FMT 1,1"  !ASCII w/source data
230 OUTPUT @Hp415x;"MM 2,";Drain !Sweep measurement

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path for controlling the 4155C/4156C.</td>
</tr>
<tr>
<td>60 to 190</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>210</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>220</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>230</td>
<td>Sets the sweep measurement mode and its measurement channel.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Using Enhanced Sweep Stop Function

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>240</td>
<td>OUTPUT @Hp415x;&quot;ESC &quot;;1,Drain,2,Ta,2,Tb</td>
</tr>
<tr>
<td>250</td>
<td>OUTPUT @Hp415x;&quot;RI &quot;;Drain,Rm</td>
</tr>
<tr>
<td>260</td>
<td>OUTPUT @Hp415x;&quot;CN &quot;;Source,Sub,Gate,Drain</td>
</tr>
<tr>
<td>270</td>
<td>OUTPUT @Hp415x;&quot;DV &quot;;Source,Ro,0,.1</td>
</tr>
<tr>
<td>280</td>
<td>OUTPUT @Hp415x;&quot;DV &quot;;Sub,Ro,0,.1</td>
</tr>
<tr>
<td>290</td>
<td>OUTPUT @Hp415x;&quot;MV &quot;;Drain,Swp,Ro,V_b,V_e,V_n,D_comp</td>
</tr>
<tr>
<td>300</td>
<td>OUTPUT @Hp415x;&quot;WSV &quot;;Gate,Ro,V_b,V_e,G_comp</td>
</tr>
<tr>
<td>310</td>
<td>OUTPUT @Hp415x;&quot;XE&quot;</td>
</tr>
<tr>
<td>320</td>
<td>OUTPUT @Hp415x;&quot;*OPC?&quot;</td>
</tr>
<tr>
<td>330</td>
<td>ENTER @Hp415x;C</td>
</tr>
<tr>
<td>340</td>
<td>OUTPUT @Hp415x;&quot;CL&quot;</td>
</tr>
<tr>
<td>350</td>
<td>!</td>
</tr>
<tr>
<td>360</td>
<td>OUTPUT @Hp415x;&quot;:SYST:ERR?&quot;</td>
</tr>
<tr>
<td>370</td>
<td>ENTER @Hp415x;C,C$</td>
</tr>
<tr>
<td>380</td>
<td>IF C=0 THEN</td>
</tr>
<tr>
<td>390</td>
<td>GOTO 450</td>
</tr>
<tr>
<td>400</td>
<td>ELSE</td>
</tr>
<tr>
<td>410</td>
<td>PRINT C,C$</td>
</tr>
<tr>
<td>420</td>
<td>GOTO 570</td>
</tr>
<tr>
<td>430</td>
<td>END IF</td>
</tr>
<tr>
<td>440</td>
<td>!</td>
</tr>
<tr>
<td>450</td>
<td>OUTPUT @Hp415x;&quot;NUB?&quot;</td>
</tr>
<tr>
<td>460</td>
<td>ENTER @Hp415x;Nub</td>
</tr>
<tr>
<td>470</td>
<td>IF Nub=0 OR Nub&lt;&gt;INT(Nub/2)*2 THEN</td>
</tr>
<tr>
<td>480</td>
<td>DISP Nub;&quot;data returned. No. of data must be N*2.&quot;</td>
</tr>
<tr>
<td>490</td>
<td>GOTO 570</td>
</tr>
<tr>
<td>500</td>
<td>ELSE</td>
</tr>
<tr>
<td>510</td>
<td>FOR Var1=1 TO Nub/2</td>
</tr>
<tr>
<td>520</td>
<td>OUTPUT @Hp415x;&quot;RMD? 2&quot;</td>
</tr>
<tr>
<td>530</td>
<td>ENTER @Hp415x USING &quot;;#,X,13D,X,3A,2X,13D,X&quot;;A,St$;B</td>
</tr>
<tr>
<td>540</td>
<td>PRINT St$;Var1;&quot;Vg=&quot;;B;&quot;V, Id=&quot;;A*1.E+6;&quot;uA&quot;</td>
</tr>
<tr>
<td>550</td>
<td>NEXT Var1</td>
</tr>
<tr>
<td>560</td>
<td>END IF</td>
</tr>
<tr>
<td>570</td>
<td>OUTPUT @Hp415x;&quot;:PAGE&quot;</td>
</tr>
<tr>
<td>580</td>
<td>END</td>
</tr>
</tbody>
</table>

Sets the stop condition. The sweep stops if the ratio of changing data is Ta or greater, and the measured data is Tb or greater.

Executes a sweep measurement.

Checks for errors.

Reads the measurement data and displays the data on the screen.

Returns the 4155C/4156C to the default control mode (SCPI command control mode).
Reading Time Stamp Data

The 4155C/4156C returns a time stamp with the measurement data in all measurement modes. Enter the MM command to define the measurement mode and enter the TSC command to set the time stamp output ON.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controls the time stamp output</td>
<td>TSC</td>
<td>onoff</td>
</tr>
<tr>
<td>Resets the time stamp</td>
<td>TSR</td>
<td></td>
</tr>
<tr>
<td>Returns the time data since TSR</td>
<td>TSQ?</td>
<td></td>
</tr>
</tbody>
</table>

The following commands returns the time stamp regardless of the TSC command setting. You can use the commands regardless of the MM command settings.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forces a constant voltage</td>
<td>TDV</td>
<td>chnum,range,output[,Icomp]</td>
</tr>
<tr>
<td>Forces a constant current</td>
<td>TDI</td>
<td>chnum,range,output[,Vcomp]</td>
</tr>
<tr>
<td>Measures the current</td>
<td>TTI</td>
<td>chnum,range</td>
</tr>
<tr>
<td>Measures the voltage</td>
<td>TTV</td>
<td>chnum,range</td>
</tr>
<tr>
<td>Measures the current and reads the data</td>
<td>TTI?</td>
<td>chnum,range</td>
</tr>
<tr>
<td>Measures the voltage and reads the data</td>
<td>TTV?</td>
<td>chnum,range</td>
</tr>
</tbody>
</table>

The TTI/TTV/TTI?/TTV? commands are for high-speed spot measurements. The query executes the measurement and reads the data. When using the TTI/TTV commands use the RMD? command to read the data.

You cannot use both the T/I/T/V/TTI/TTV commands and the T/I?/T/V ?/TTI?/TTV ? commands in the same measurement program. The T/I/T/V/TTI?/TV? commands execute high-speed spot measurements without the time stamp.

The time stamp function does not support QSCV and search measurements.
The following example program executes a high-speed spot measurement using the TTI? command, and displays the measurement result and time stamp on the screen.

```
10 ASSIGN @Hp415x TO 800
20 !
30 INTEGER Source,Drain,Gate,Sub
40 INTEGER Range_s,Range_g,Range_d,Range_i,B,C
50 DIM B$[100]
60 !
70 Source=1       !1:SMU1
80 Drain=2        !2:SMU2
90 Gate=3         !3:SMU3
100 Sub=4          !4:SMU4
110 Range_s=11     !11: 2 V Limited Auto Ranging
120 Range_g=12     !12:20 V Limited Auto Ranging
130 Range_d=12     !12:20 V Limited Auto Ranging
140 Range_i=15     !15:10 uA Limited Auto Ranging
150 Vs=0           ! Source Voltage
160 Vd=5           ! Drain Voltage
170 Vg=3           ! Gate Voltage
180 Vsub=0         ! Substrate Voltage
190 Icomp_g=.01    ! Current compliance for gate
200 Icomp=.1       ! Current compliance
210 !
220 OUTPUT @Hp415x;"US"
230 OUTPUT @Hp415x;"FMT 1"        !ASCII w/header
240 OUTPUT @Hp415x;"AV 1"         !Number of average
250 OUTPUT @Hp415x;"SIT 1,.0005"  !Integ for Short
260 OUTPUT @Hp415x;"SIT 3,.04"    !Integ for Long
270 OUTPUT @Hp415x;"SLI 1"        !Short:1,Long:3
280 OUTPUT @Hp415x;"FL 1"         !Filter on:1, off:0
290 OUTPUT @Hp415x;"CN ";Source,Drain,Gate,Sub
300 OUTPUT @Hp415x;"TSR"
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path for controlling the 4155C/4156C.</td>
</tr>
<tr>
<td>70 to 200</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>220</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>230</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>240 to 270</td>
<td>Sets the integration time.</td>
</tr>
<tr>
<td>280</td>
<td>Sets the filter mode.</td>
</tr>
<tr>
<td>290</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>300</td>
<td>Resets the time stamp (sets the time stamp to 0 sec).</td>
</tr>
</tbody>
</table>
### FLEX Command Programming

#### Reading Time Stamp Data

```
310    OUTPUT @Hp415x;"DV ";Source,Range_s,Vs,Icomp
320    OUTPUT @Hp415x;"DV ";Sub,Range_s,Vsub,Icomp
330    OUTPUT @Hp415x;"DV ";Gate,Range_g,Vg,Icomp_g
340    OUTPUT @Hp415x;"DV ";Drain,Range_d,Vd,Icomp
350  !
360    OUTPUT @Hp415x;"*OPC?"
370    ENTER @Hp415x;C
380  !
390    OUTPUT @Hp415x;":SYST:ERR?"
400    ENTER @Hp415x;B,B$
410    IF B=0 THEN
420      OUTPUT @Hp415x;"TSQ?"
430      ENTER @Hp415x USING ";#,5X,13D,X";T1
440      OUTPUT @Hp415x;"TTI? ";Drain,Range_i
450      ENTER @Hp415x USING ";#,5X,13D,6X,13D,X";T2,Md1
460      PRINT "Id=",Md1*1000;"mA"
470      PRINT "Measurement time=",(T2-T1)*.1;"ms"
480    ELSE
490      PRINT "ERROR: ";B$
500    END IF
510  !
520    OUTPUT @Hp415x;"CL"
530    OUTPUT @Hp415x;"TSQ?"
540    ENTER @Hp415x USING ";#,5X,13D,X";T3
550    PRINT "Source output time=",T3*.1;"ms"
560    OUTPUT @Hp415x;":PAGE"
570    END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>310 to 340</td>
<td>Forces the dc voltage.</td>
</tr>
<tr>
<td>390 to 400</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>420 to 430</td>
<td>Reads the time stamp data before starting the measurement.</td>
</tr>
<tr>
<td>440 to 470</td>
<td>Executes a high-speed spot measurement, and displays the measured data and the time stamp on the screen.</td>
</tr>
<tr>
<td>490</td>
<td>Prints an error code if an error has occurred.</td>
</tr>
<tr>
<td>520</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>530 to 550</td>
<td>Reads the time stamp data after stopping the source output, and displays the source output time.</td>
</tr>
<tr>
<td>560</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>

---

3-64 Agilent 4155C/4156C Programmer's Guide, Edition 4
The following example program executes the sampling measurement, and displays the measurement result and time data on the screen.

```
10  ASSIGN @Hp415x TO 800
20  !
30  INTEGER High,Low,Range_v,Range_i
40  INTEGER Point,Abort,A,C,I
50  REAL B,R
60  DIM C$[50]
70  High=1         !1:SMU1
80  Low=2          !2:SMU2
90  Hold=.1        !Hold time (sec)
100 Interval=.01   !Sampling interval (sec)
110 Point=9        !Number of sampling points
120 Abort=2        !2: Selects all abort condition
130 Range_v=0      !0: Auto range
140 Vbase=0        !Base voltage for High
150 Vbias=.05      !Bias voltage for High
160 Icomp=.1       !Current compliance
170 Vl=0           !Voltage for Low
180 Range_i=-19    !-19: 100 mA fixed range
190  !
200  OUTPUT @Hp415x;"US"
210  OUTPUT @Hp415x;"FMT 1,1"     ! ASCII w/header,index
220  OUTPUT @Hp415x;"TSC 1"       ! Time stamp
230  OUTPUT @Hp415x;"CN ";High,Low
240  OUTPUT @Hp415x;"MCC"
250  OUTPUT @Hp415x;"MT ";Hold,Interval,Point
260  OUTPUT @Hp415x;"MM 10, ";High ! Sampling measurement
270  OUTPUT @Hp415x;"RI ";High,Range_i
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>70 to 180</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>200</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>210</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>220</td>
<td>Enables the time stamp output.</td>
</tr>
<tr>
<td>230</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>240 to 270</td>
<td>Sets the sampling measurement conditions.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Reading Time Stamp Data

280   OUTPUT @Hp415x;"MSC ";Abort
290   OUTPUT @Hp415x;"MV ";High,Range_v,Vbase,Vbias,Icomp
300   OUTPUT @Hp415x;"DV ";Low,Range_v,Vl,Icomp
310   !
320   OUTPUT @Hp415x;"TSR"
330   OUTPUT @Hp415x;"XE"
340   OUTPUT @Hp415x;"*OPC?"
350   ENTER @Hp415x;C
360   OUTPUT @Hp415x;"CL"
370   !
380   OUTPUT @Hp415x;":SYST:ERR?"
390   ENTER @Hp415x;C,C$
400   !
410   IF C=0 THEN
420       PRINT "Hold time=";Hold*1000;"ms"
430       PRINT "Sampling interval=";Interval*1000;"ms"
440       FOR I=1 TO Point
450           OUTPUT @Hp415x;"RMD? 3"
460           ENTER @Hp415x USING ",5X,13D,6X,13D,6X,13D,X";A,T1,B
470           R=Vbias/B
480           PRINT "No.";A;" R=";R;"ohm, Time=";T1*.1;"ms"
490       NEXT I
500   ELSE
510       PRINT "ERROR:";C$
520   END IF
530   !
540   OUTPUT @Hp415x;":PAGE"
550   END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>280</td>
<td>Sets the sampling stop condition.</td>
</tr>
<tr>
<td>290 to 300</td>
<td>Forces the sampling source to the high terminal, and the dc voltage to the low terminal.</td>
</tr>
<tr>
<td>320 to 360</td>
<td>Resets the time stamp, and executes a sampling measurement.</td>
</tr>
<tr>
<td>380 to 390</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>420 to 490</td>
<td>Reads the measurement data and displays the data on the screen.</td>
</tr>
<tr>
<td>510</td>
<td>Prints an error code if an error has occurred.</td>
</tr>
<tr>
<td>540</td>
<td>Returns the 4155C/4156C to the default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
Reading and Writing Data to a File

To read or write ASCII data to a file, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selects the mass storage device</td>
<td>SSDK</td>
<td>0, 1, 2, 3, or 4.</td>
</tr>
<tr>
<td>Opens the specified file</td>
<td>OPEN</td>
<td>file_name,mode</td>
</tr>
<tr>
<td>Closes the file</td>
<td>CLOSE</td>
<td></td>
</tr>
<tr>
<td>Writes data</td>
<td>WR</td>
<td>data (ASCII data, 254 bytes maximum)</td>
</tr>
<tr>
<td>Reads data</td>
<td>RD?</td>
<td>a maximum of 8 KB of ASCII data can be read</td>
</tr>
</tbody>
</table>

To read or write file data on the network file system, the SYSTEM: MISCELLANEOUS tables below must be defined, and the 4155C/4156C must be connected to your LAN.

- 4155C/4156C NETWORK SETUP table
- NETWORK DRIVE SETUP table

The following example writes the data (Data$) to a file (MDATA) on the network file system, defined in the NETWORK DRIVE SETUP table. Data$ must not include a single quotation (').

100 OUTPUT @Hp415x;"SDSK 1" !1:NFS1
110 OUTPUT @Hp415x;"OPEN 'MDATA',1" !1:Over write mode
120 OUTPUT @Hp415x;"WR ";CHR$(39)&Data$&CHR$(39)
130 OUTPUT @Hp415x;"CLOSE"

The following example reads the data from a file (MDATA) on a diskette, and enters the data into Data$.

100 OUTPUT @Hp415x;"SDSK 0" !0:diskette
110 OUTPUT @Hp415x;"OPEN 'MDATA',0" !0:Read only mode
120 OUTPUT @Hp415x;"RD?"
130 ENTER @Hp415x;Data$
140 OUTPUT @Hp415x;"CLOSE"
**Example 1**

The following program example:

1. Executes high-speed spot measurements.
2. Writes the measurement data with a separator (, :comma) into a file on the diskette.
3. Reads the data from the file on the diskette.
4. Prints the data on the screen.

```plaintext
10  ASSIGN @Hp415x TO 800
20  OPTION BASE 1
30  !
40  DIM A$(255)
50  DIM Mdata$(8200)
60  REAL Vout,Mdata
70  INTEGER I,M,N,X,Y
80  !
90  File$="MDATA"
100 Fmt=1
110 Disk=0
120  !
130  OUTPUT @Hp415x;"US"
140  OUTPUT @Hp415x;"FMT ";Fmt
150  "**** HIGH-SPEED SPOT MEASUREMENTS ****
160  OUTPUT @Hp415x;"CN 1"
170  X=1
180  FOR I=1 TO 11
190     Vout=(I-1)*.5
200     OUTPUT @Hp415x;"DV 1,12,";Vout
210     OUTPUT @Hp415x;"TI? 1,15"
220     ENTER @Hp415x USING ",5X,13D,X";Mdata
230     IF I=11 THEN
240         Mdata$(X,X+13)=VAL$(Mdata)
250     GO TO 310
260 ELSE
270     Mdata$(X,X+13)=VAL$(Mdata)&CHR$(44)
280  END IF
290  X=X+14
300 NEXT I
```

<table>
<thead>
<tr>
<th>Line No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>130</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>140</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>160</td>
<td>Enables the measurement unit.</td>
</tr>
<tr>
<td>180 to 300</td>
<td>Forces the dc voltage, and measures the dc current. Measured data is entered into M data$ with a data separator (, ).</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Reading and Writing Data to a File

310    OUTPUT @Hp415x; "CL"
320    !
330    !**** WRITES MEASUREMENT DATA ****
340    N=LEN(Mdata$)
350    M=INT(N/250)+1
360    OUTPUT @Hp415x; "SDSK 0"
370    OUTPUT @Hp415x; "OPEN "&CHR$(39)&File$&CHR$(39);",1"
380    Y=1
390    FOR I=1 TO M
400      A$=Mdata$[Y,Y+249]
410      OUTPUT @Hp415x; "WR "&CHR$(39)&A$&CHR$(39)
420      WAIT .1
430      Y=Y+250
440    NEXT I
450    OUTPUT @Hp415x; "CLOSE"
460    !**** READS MEASUREMENT DATA ****
470    OUTPUT @Hp415x; "OPEN "&CHR$(39)&File$&CHR$(39);",0"
480    OUTPUT @Hp415x; "RD?"
490    ENTER @Hp415x; Mdata$
500    OUTPUT @Hp415x; "CLOSE"
510    !**** PRINTS MEASUREMENT DATA ****
520    PRINT Mdata$
530    !
540    OUTPUT @Hp415x; ":PAGE"
550    END

<table>
<thead>
<tr>
<th>Line No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>310</td>
<td>Enables the measurement unit.</td>
</tr>
<tr>
<td>360</td>
<td>Sets the mass storage device.</td>
</tr>
<tr>
<td>370</td>
<td>Opens the specified file (file name: M DATA ).</td>
</tr>
<tr>
<td>390 to 440</td>
<td>Writes the measurement data (M data$) to M DATA .</td>
</tr>
<tr>
<td>450</td>
<td>Closes the file.</td>
</tr>
<tr>
<td>460 to 500</td>
<td>Reads the measurement data from M DATA .</td>
</tr>
<tr>
<td>520</td>
<td>Prints the measurement data on the screen.</td>
</tr>
<tr>
<td>540</td>
<td>Returns to the 4155C/4156C default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>


**Example 2**

The following program example:

1. executes staircase sweep measurements.
2. writes the measurement data with a separator (, : comma) to a file on the diskette.

```plaintext
10     ASSIGN @Hp415x TO 800
20     OPTION BASE 1
30     INTEGER Fmt, Emitter, Base, Collector, Mmode, Swp, N
40     INTEGER Mrange, Ib_point, Range, Var1, Var2
50     REAL Vc(101)
60     DIM Mdata$(11000)
70     DIM C$(250)
80     Fmt=1          ! 1: ASCII with header <LF^EOI>
90     Emitter=1      ! 1: SMU1
100    Base=2         ! 2: SMU2
110    Collector=3    ! 3: SMU3
120    Swp=1          ! 1: Linear single sweep mode
130    V1=0           ! Collector voltage start value (V)
140    V2=1           ! Collector voltage stop value (V)
150    N=101          ! Collector voltage number of steps
160    Comp=.1        ! Current compliance (A) for collector
170    Mrange=14      ! 14: 1 uA limited auto ranging
180    Range=0        ! 0: Auto ranging
190    Ve=0           ! Emitter voltage (V)
200    Ie_comp=.1     ! Current compliance (A) for emitter
210    Mmode=2        ! 2: Staircase sweep measurement
220    Ib_start=1.E-5 ! Ib start value (A)
230    Ib_step=1.E-5  ! Ib step value (A)
240    Ib_point=3     ! Number of Ib steps
250    Vb_comp=2      ! Voltage compliance (V) for base
260    X=1           ! Mdata$ INDEX
270    Disk=0         ! 0:diskette
280   File$="DATA1"   ! File name for measurement data
290   !
300    OUTPUT @Hp415x;"US"
310    OUTPUT @Hp415x;"FMT ";Fmt
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>80 to 280</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>300</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>310</td>
<td>Specifies the data output format.</td>
</tr>
</tbody>
</table>
FLEX Command Programming

Reading and Writing Data to a File

320
330 \quad Vc\_step=\frac{(V2-V1)}{(N-1)}
340 \quad \text{FOR} \ Var1=1 \ \text{TO} \ N
350 \quad \text{Vc}(Var1)=V1+(Var1-1)\times Vc\_step
360 \quad \text{NEXT} \ Var1
370 \quad \text{OUTPUT} \ @Hp415x;"CN \ Emitter,Base,Collector"
380 \quad \text{OUTPUT} \ @Hp415x;"MV \ Collector,Swp,Range,V1,V2,N,Comp"
390 \quad \text{OUTPUT} \ @Hp415x;"RI \ Collector,Mrange"
400 \quad \text{OUTPUT} \ @Hp415x;"DV \ Emitter,Range,Ve,Ie\_comp"
410 \quad \text{OUTPUT} \ @Hp415x;"MM \ Mmode,Collector"
420 \quad \text{FOR} \ Var2=1 \ \text{TO} \ Ib\_point
430 \quad Ib=Ib\_start+(Var2-1)\times Ib\_step
440 \quad \text{OUTPUT} \ @Hp415x;"DI \ Base,Range,Ib,Vb\_comp"
450 \quad \text{OUTPUT} \ @Hp415x;"XE"
460 \quad \text{IF} \ C=0 \ \text{THEN}
470 \quad \text{ELSE}
480 \quad \text{PRINT} \ C,C$
490 \quad \text{GOTO} \ 560
500 \quad \text{END IF}
510 \quad \text{GOTO} \ 790
520 \quad \text{GOTO} \ 560
530 \quad \text{END IF}
540 \quad \text{GOTO} \ 790
550 \quad \text{GOTO} \ 560

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>330 to 360</td>
<td>Calculates the collector voltage values.</td>
</tr>
<tr>
<td>370</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>380</td>
<td>Sets the staircase sweep source (Vc).</td>
</tr>
<tr>
<td>390</td>
<td>Sets the measurement range (Ic).</td>
</tr>
<tr>
<td>400</td>
<td>Forces the dc voltage (Ve).</td>
</tr>
<tr>
<td>410</td>
<td>Sets the measurement mode.</td>
</tr>
<tr>
<td>440</td>
<td>Forces the dc current (Ib).</td>
</tr>
<tr>
<td>450</td>
<td>Executes a staircase sweep measurement.</td>
</tr>
<tr>
<td>470 to 540</td>
<td>Checks for errors.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Reading and Writing Data to a File

560      FOR Var1=1 TO N
570        OUTPUT @Hp415x;"RMD? 1"
580        ENTER @Hp415x USING ";,5X,13D,X";Ic
590        Mdata$[X,X+9]=VAL$(Ib)&CHR$(44)
600        Mdata$[X+10,X+19]=VAL$(Vc(Var1))&CHR$(44)
610        Mdata$[X+20,X+34]=VAL$(Ic)&CHR$(13)&CHR$(10)
620        X=X+35
630      NEXT Var1
640    NEXT Var2
650    !
660    OUTPUT @Hp415x;"SDSK ";Disk
670    OUTPUT @Hp415x;"OPEN ";CHR$(39)&File$&CHR$(39);",1"
680    C$="Ib(A),Vc(V),Ic(A)"&CHR$(13)&CHR$(10)
690    OUTPUT @Hp415x;"WR ";CHR$(39)&C$&CHR$(39)
700    N=LEN(Mdata$)
710    M=INT(N/250)+1
720    X=1
730    FOR I=1 TO M
740      C$=Mdata$[X,X+249]
750      OUTPUT @Hp415x;"WR ";CHR$(39)&C$&CHR$(39)
760      X=X+250
770    NEXT I
780    OUTPUT @Hp415x;"CLOSE"
790    OUTPUT @Hp415x;"CL"
800    OUTPUT @Hp415x;":PAGE"
810    END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>560 to 630</td>
<td>Reads the measurement data.</td>
</tr>
<tr>
<td>660</td>
<td>Specifies the mass storage device.</td>
</tr>
<tr>
<td>670</td>
<td>Opens the data file (DATA1) to store the measurement data.</td>
</tr>
<tr>
<td>680 to 690</td>
<td>Writes &quot;Ib(A),Vc(V),Ic(A)&quot; (with a return and line feed) into DATA1.</td>
</tr>
<tr>
<td>700 to 770</td>
<td>Writes measured data (with a return and line feed) into DATA1.</td>
</tr>
<tr>
<td>780</td>
<td>Closes the file (DATA1).</td>
</tr>
<tr>
<td>790</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>800</td>
<td>Returns to the 4155C/4156C default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
**Example 3**
The following program example does following a or b:

a. Reads the data from a specified file on the diskette, and writes the data to a specified file on the network file system.

b. Reads the data from a specified file on the network file system, and writes the data to a specified file on the diskette.

Program example limitations:

- The mass storage devices are the disk drive and the network file system, which is first defined in the NETWORK DRIVE SETUP table.
- The data must be a maximum of 8 KB.
- A single quotation (’ ) must not be included in the data.
FLEX Command Programming
Reading and Writing Data to a File

10  ASSIGN @Hp415x TO 800
20  OPTION BASE 1
30  DIM A$(255)
40  DIM B$(8200)
50  DIM C$(100)
60  INTEGER Source,Dest
70  CLEAR SCREEN
80  !
90  PRINT "******************************************"
100 PRINT "* Select Source.*
110 PRINT "* Enter 1 (NFS1) or 0 (diskette)*"
120 INPUT Source
130 IF Source=1 THEN
140 PRINT "* NFS1 ----> DISKETTE"
150 Dest=0
160 ELSE
170 IF Source=0 THEN
180 PRINT "* DISKETTE ----> NFS1"
190 Dest=1
200 ELSE
210 PRINT "* Source selection error. END.*
220 GOTO 630
230 END IF
240 END IF
250 PRINT "******************************************"
260 PRINT "* Enter READ file name.*"
270 INPUT Rname$
280 PRINT "* READ file = ";Rname$
290 PRINT "******************************************"
300 PRINT "* Enter WRITE file name.*"
310 INPUT Wname$
320 PRINT "* WRITE file = ";Wname$
330 PRINT "******************************************"
340!
350 OUTPUT @Hp415x;"US"
360 OUTPUT @Hp415x;"SDSK ";Source

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>90 to 250</td>
<td>Waits for the source device input (0: diskette, 1: NFS1) and stores the value.</td>
</tr>
<tr>
<td>260 to 290</td>
<td>Waits for the file name to be read and stores the value.</td>
</tr>
<tr>
<td>300 to 330</td>
<td>Waits for the file name to be written and stores the value.</td>
</tr>
<tr>
<td>350</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>360</td>
<td>Sets the &quot;source&quot; mass storage device.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Reading and Writing Data to a File

370 OUTPUT @Hp415x;"OPEN ";CHR$(39)&Rname$&CHR$(39);",0"
380 OUTPUT @Hp415x;"RD?"
390 ENTER @Hp415x USING ";K";B$
400 OUTPUT @Hp415x;"CLOSE"
410 OUTPUT @Hp415x;"SDSK ";Dest
420 OUTPUT @Hp415x;"OPEN ";CHR$(39)&Wname$&CHR$(39);","1"
430 N=LEN(B$)
440 M=INT(N/250)+1
450 X=1
460 FOR I=1 TO M
470 A$=B$[X,X+249]
480 X=X+250
490 OUTPUT @Hp415x;"WR ";CHR$(39)&A$&CHR$(39)
500 WAIT .1
510 NEXT I
520 OUTPUT @Hp415x;"CLOSE"
530 !
540 OUTPUT @Hp415x;":SYST:ERR?"
550 ENTER @Hp415x;C,C$
560 IF C=0 THEN
570 PRINT "+ File transfer was completed."
580 GOTO 630
590 ELSE
600 PRINT "+ Code=";C
610 PRINT "+ Message =";C$
620 END IF
630 PRINT "********************************************************************************
640 OUTPUT @Hp415x;":PAGE"
650 END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>370</td>
<td>Opens the file to read data.</td>
</tr>
<tr>
<td>380 to 390</td>
<td>Reads the data (8 KB maximum) from MDATA.</td>
</tr>
<tr>
<td>400</td>
<td>Closes the file.</td>
</tr>
<tr>
<td>410</td>
<td>Sets the &quot;destination&quot; mass storage device.</td>
</tr>
<tr>
<td>420</td>
<td>Opens the file to write data.</td>
</tr>
<tr>
<td>430 to 510</td>
<td>Writes the data to MDATA.</td>
</tr>
<tr>
<td>520</td>
<td>Closes the file.</td>
</tr>
<tr>
<td>540 to 630</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>640</td>
<td>Returns to the 4155C/4156C default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
Printing Data

To print data to a remote printer connected to the print server, use the following commands.

<table>
<thead>
<tr>
<th>Function</th>
<th>FLEX Command</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies network drive</td>
<td>SDSK</td>
<td>1, 2, 3, or 4.</td>
</tr>
<tr>
<td>Specifies remote printer</td>
<td>SPR</td>
<td>1, 2, 3, or 4.</td>
</tr>
<tr>
<td>Spools data</td>
<td>SPL</td>
<td>data (ASCII data)</td>
</tr>
<tr>
<td>Executes print-out</td>
<td>PRN</td>
<td></td>
</tr>
</tbody>
</table>

The 4155C/4156C must be connected to your LAN, and the following setup tables on the SYSTEM: MISCELLANEOUS screen must be defined.

- 4155C/4156C NETWORK SETUP table
- NETWORK PRINTER SETUP table
- NETWORK DRIVE SETUP table

The following example executes the data print-out (Data$) using the remote printer (Printer1), defined in the NETWORK PRINTER SETUP table. Data$ must not include a single quotation(').  

```
100 Disk=1   !1:NFS1, 2:NFS2, 3:NFS3, 4:NFS4
110 Printer=1 !1:Printer1, 2:Printer2, 3:Printer3, 4:Printer4
120 !
130 OUTPUT @Hp415x;"SDSK ";Disk
140 OUTPUT @Hp415x;"SPR ";Printer
150 OUTPUT @Hp415x;"SPL ";CHR$(39)&Data$&CHR$(39)
160 OUTPUT @Hp415x;"PRN"
```
The following program example:

1. executes high-speed spot measurements.
2. prints the data to the remote printer.

```plaintext
10 ASSIGN @Hp415x TO 800
20 OPTION BASE 1
30 DIM A$[25]
40 DIM C$[50]
50 DIM Mdata$[8200]
60 REAL Vout,Mdata
70 INTEGER I,N,X,Y,Fmt,Disk,Prn,No_test
80!
90 Fmt=1       !1: ASCII with header <LF^EOI>
100 Disk=1      !1: NFS1
110 Prn=1       !1: Remote printer 1
120 No_test=10  ! Number of measurement points
130!
140 OUTPUT @Hp415x;"US"
150 OUTPUT @Hp415x;"FMT ";Fmt
160!
170 !HIGH-SPEED SPOT MEASUREMENTS **************
180 OUTPUT @Hp415x;"CN 1"
190 X=1
200 FOR I=1 TO No_test
210 Vout=(I-1)*.5
220 OUTPUT @Hp415x;"DV 1,12,",Vout
230 OUTPUT @Hp415x;"TI? 1,15"
240 ENTER @Hp415x USING ";#,5X,13D,X";Mdata
250 Mdata$[X,X+12]=VAL$(Mdata)
260 X=X+13
270 NEXT I
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>140</td>
<td>Enters the 4155C/4156C FLEX command control mode.</td>
</tr>
<tr>
<td>150</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>180</td>
<td>Enables the measurement unit.</td>
</tr>
<tr>
<td>200 to 270</td>
<td>Forces the dc voltage (0 to 5 V, in 0.5 V steps) and measures the dc current. Measured data is entered into Mdata$.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Printing Data

280    OUTPUT @Hp415x;"CL"
290    !
300    !PRINTS MEASUREMENT DATA **********************
310    OUTPUT @Hp415x;"SDSK ";Disk
320    OUTPUT @Hp415x;"SPR ";Prn
330    !
340    Y=1
350    FOR I=1 TO No_test
360        A$="I("&VAL$(I)&")-"&Mdata$[Y,Y+12]
370        A$=A$&CHR$(13)&CHR$(10)
380    OUTPUT @Hp415x;"SPL ";CHR$(39)&A$&CHR$(39)
390    Y=Y+13
400    NEXT I
410    !
420    OUTPUT @Hp415x;":SYST:ERR?"
430    ENTER @Hp415x;C,C$
440    IF C=0 THEN
450        OUTPUT @Hp415x;"PRN"
460    ELSE
470        PRINT "ERROR:";C$
480    END IF
490    OUTPUT @Hp415x;":PAGE"
500    END

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>280</td>
<td>Disables the measurement unit.</td>
</tr>
<tr>
<td>310</td>
<td>Sets the mass storage device.</td>
</tr>
<tr>
<td>320</td>
<td>Specifies the remote printer.</td>
</tr>
<tr>
<td>350 to 400</td>
<td>Spools the measurement data.</td>
</tr>
<tr>
<td>420 to 430</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>450</td>
<td>Requests a print-out to the remote printer.</td>
</tr>
<tr>
<td>470</td>
<td>If an error has occurred, prints the error message on the screen.</td>
</tr>
<tr>
<td>490</td>
<td>Returns to the 4155C/4156C default control mode (SCPI command control mode).</td>
</tr>
</tbody>
</table>
Reading Binary Output Data

The program examples shown in the previous sections use the ASCII data output format for measurement data.

ASCII data format is easier than binary data format for reading the measurement data, because ASCII data can be read directly, without rearranging the data. The data length is longer in ASCII format than in binary data format, so the data transfer time in ASCII format is longer than in binary format.

To reduce the data transfer time, use binary data output format.

For details of data output formats, refer to Chapter 1 of the GPIB Command Reference.

The following program example:

1. executes high-speed spot measurements
2. reads the measurement data using binary output format
3. rearranges the data and calculates the measured data
4. prints the measured data on the screen

```
10     ASSIGN @Hp415x TO 800
20   !
30     INTEGER Fmt,Average,Type,Source,Drain,Gate,Sub
40     INTEGER Range_2v,Range_20v,Range_i,B,C
50     REAL Value,Status
60     DIM Mdata$[6]
70     DIM B$[50]
80   !
90     Fmt=3          !3:Binary <LF^EOI>
100    Average=1      !Number of averaging
110    Sinteg=.0005   !Integ Time of Short
120    Linteg=.04     !Integ Time of Long
130    Type=1         !1:Short, 2:Medium, 3:Long
140    Filter=0       !0:Filter off, 1:Filter on
150    Source=1       !1:SMU1
160    Drain=2        !2:SMU2
170    Gate=3         !3:SMU3
180    Sub=4          !4:SMU4
190    Range_2v=11    !11: 2 V Limited Auto Ranging
200    Range_20v=12   !12:20 V Limited Auto Ranging
210    Range_i=15     !15:10 uA Limited Auto Ranging
220    Vs=0           ! Source Voltage
230    Vd=5           ! Drain Voltage
240    Vg=3           ! Gate Voltage
250    Vsub=0         ! Substrate Voltage
260    Icomp_g=.01    ! Current compliance for gate
```
**FLEX Command Programming**

**Reading Binary Output Data**

270 Icomp=.1       ! Current compliance
280 !
290 OUTPUT @Hp415x;"US"
300 OUTPUT @Hp415x;"FMT ";Fmt
310 OUTPUT @Hp415x;"AV ";Average
320 OUTPUT @Hp415x;"SIT 1,";Sinteg !for Short
330 OUTPUT @Hp415x;"SIT 3,";Linteg !for Long
340 OUTPUT @Hp415x;"SLI ";Type
350 OUTPUT @Hp415x;"FL ";Filter
360 OUTPUT @Hp415x;"CN ";Source,Drain,Gate,Sub
370 OUTPUT @Hp415x;"DV ";Source,Range_2v,Vs,Icomp
380 OUTPUT @Hp415x;"DV ";Sub,Range_2v,Vsub,Icomp
390 OUTPUT @Hp415x;"DV ";Gate,Range_20v,Vg,Icomp_g
400 OUTPUT @Hp415x;"DV ";Drain,Range_20v,Vd,Icomp
410 !
420 OUTPUT @Hp415x;":SYST:ERR?"
430 ENTER @Hp415x;B,B$
440 IF B=0 THEN
450 OUTPUT @Hp415x;"TI? ";Drain,Range_i
460 OUTPUT @Hp415x;"CL"
470 ENTER @Hp415x USING ";",6A";Mdata$
480 ELSE
490 OUTPUT @Hp415x;"CL"
500 PRINT "ERROR=";B$
510 GOTO 570
520 END IF
530 !
540 CALL Get_data(Mdata$,Value,Status)
550 PRINT ";A=";Value
560 !

<table>
<thead>
<tr>
<th>Line No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4155C/4156C.</td>
</tr>
<tr>
<td>90 to 270</td>
<td>Sets the measurement parameters.</td>
</tr>
<tr>
<td>290 to 400</td>
<td>Sets the measurement condition.</td>
</tr>
<tr>
<td>420 to 430</td>
<td>Checks for errors.</td>
</tr>
<tr>
<td>450 to 470</td>
<td>If no error has occurred, executes the high-speed spot measurement, disables the measurement unit, and reads the output data.</td>
</tr>
<tr>
<td>490 to 510</td>
<td>If an error has occurred, disables the measurement unit, prints the error message on the screen, and ends the program execution.</td>
</tr>
<tr>
<td>540</td>
<td>Calls the Get_data sub-program.</td>
</tr>
<tr>
<td>550</td>
<td>Prints the measured data on the screen.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Reading Binary Output Data

570    OUTPUT @Hp415x;":PAGE"
580    END
590    !
600    SUB Get_data(Mdata$,Value,Status)
610      INTEGER D1,D2,D3,D4,D5,D6,M_s,I_v,X
620    !
630      D1=NUM(Mdata$[1;1])    !Byte 1
640      D2=NUM(Mdata$[2;1])    !Byte 2
650      D3=NUM(Mdata$[3;1])    !Byte 3
660      D4=NUM(Mdata$[4;1])    !Byte 4
670      D5=NUM(Mdata$[5;1])    !Byte 5
680      D6=NUM(Mdata$[6;1])    !Byte 6
690    !
700      M_s=BIT(D1,7)
710    !  M_s : 0: Source data, 1: Measurement data
720    !
730      I_v=SHIFT(BINAND(D1,112),4)  ! 112: 01110000
740    !  I_v :  0:V, 1:I, 6:Sampling index, 7:Status
750    !
760      Range_no=BINAND(D1,15)*2+BIT(D2,7)
770    !  15: 00001111
780    !
790      Status=BINAND(D5,31)*8+SHIFT(D6,5)
800    !
810      Count=SHIFT(D5,5)
820      Count=Count+D4*8
830      Count=Count+D3*8*256.
840      Count=Count+BINAND(D2,63)*8*256.*256.
850    !
860      IF BIT(D2,6)=0 THEN    ! Positive data
870        Count=Count
880      ELSE                   ! Negative data
890        Count=Count-33554432 ! 33554432=2^25
900      END IF

<table>
<thead>
<tr>
<th>Line No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>570</td>
<td>Returns to the 4155C/4156C default control mode (SCPI mode).</td>
</tr>
<tr>
<td>630 to 680</td>
<td>Separates 6 bytes of output data (M data$) to a byte (D1 to D6).</td>
</tr>
<tr>
<td>700</td>
<td>Reads the measurement or source data type.</td>
</tr>
<tr>
<td>730</td>
<td>Reads the data type (voltage, current, sampling index, or status information).</td>
</tr>
<tr>
<td>760</td>
<td>Reads the measurement range. This value is the reference ID of the measurement range.</td>
</tr>
<tr>
<td>790</td>
<td>Reads the status information.</td>
</tr>
<tr>
<td>810 to 900</td>
<td>Reads the Count value used to calculate the measurement data.</td>
</tr>
</tbody>
</table>
FLEX Command Programming
Reading Binary Output Data

910  !
920  SELECT I_v
930   CASE 0       ! V range
940    SELECT Range_no
950    CASE 10
960    Range=.2
970    CASE 11
980    Range=2
990    CASE 12
1000   Range=20
1010   CASE 13
1020   Range=40
1030   CASE 14
1040   Range=100
1050   CASE 15
1060   Range=200
1070  END SELECT
1080  CASE 1       ! I range
1090    Range=10^(Range_no-20)
1100  CASE 6      ! Sampling index
1110    Value=Count
1120    GOTO 1250
1130  CASE 7      ! Status info
1140    Value=0
1150    GOTO 1250
1160  END SELECT
1170  !
1180  SELECT M_s
1190   CASE 0
1200    Value=Count*Range/20000
1210   CASE 1
1220    Value=Count*Range/1.E+6
1230  END SELECT
1240  !
1250  SUBEND

<table>
<thead>
<tr>
<th>Line No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>930 to 1090</td>
<td>For measurement data, finds the measurement range from the ID read by line 760.</td>
</tr>
<tr>
<td>1100 to 1120</td>
<td>For sampling index, enters the sampling index into the Value parameter and returns to the main program.</td>
</tr>
<tr>
<td>1130 to 1150</td>
<td>For status information, enters 0 into the Value parameter and returns to the main program. Only the Status parameter is effected.</td>
</tr>
<tr>
<td>1180 to 1230</td>
<td>Calculates the measurement or source data.</td>
</tr>
</tbody>
</table>
Using the US42 Control Mode

The 4155C/4156C FLEX command set includes some commands which have the same name as the GPIB commands for Agilent 4142B DC source/monitor. The US42 command provides an 4142B-like response for the following items:

- output data format
- query response
- status code (status byte)

To create a measurement program to control the 4155C/4156C, by modifying the program created to control the 4142B:

1. Change the GPIB address, if necessary.
2. Enter the US42 command to enter the FLEX command control mode.
3. Enter the ACH command to translate the measurement unit numbers.
4. Enter a space between the command and the first parameter.
5. If you do not specify the US42 command level parameter 16, enter the RMD? command to read the output data (before executing the ENTER command).
6. Enter the :PAGE command to return to the 4155C/4156C default control mode (SCPI command control mode).

The following program examples show a modified measurement program, which performs a high-speed spot measurement.
FLEX Command Programming
Using the US42 Control Mode

The original 4142B program:

```plaintext
10     ASSIGN @Hp4142 TO 717
20     INTEGER G_ch,D_ch,S_ch
30   !
40   !         !Source:    GNDU
50     G_ch=2  !Gate:      HPSMU (SLOT2)
60     D_ch=3  !Drain:     MPSMU (SLOT3)
70     S_ch=4  !Substrate: MPSMU (SLOT4)
80   !
90     OUTPUT @Hp4142;"FMT1"
100    OUTPUT @Hp4142;"CN";D_ch,G_ch,S_ch
110    OUTPUT @Hp4142;"DV";S_ch;"0,0,0.1"
120    OUTPUT @Hp4142;"DV";G_ch;"0,3,.01"
130    OUTPUT @Hp4142;"DV";D_ch;"0,5,.1"
140    OUTPUT @Hp4142;"TI";D_ch;"0"
150    OUTPUT @Hp4142;"CL"
160    ENTER @Hp4142 USING ":,3X,12D,X";Mdata
170    PRINT "Id(A)=";Mdata
180    END
```

<table>
<thead>
<tr>
<th>Line Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Assigns the I/O path to control the 4142B.</td>
</tr>
<tr>
<td>50 to 70</td>
<td>Defines the measurement channels.</td>
</tr>
<tr>
<td>90</td>
<td>Specifies the data output format.</td>
</tr>
<tr>
<td>100</td>
<td>Enables the measurement units.</td>
</tr>
<tr>
<td>110 to 130</td>
<td>Forces the dc voltage to S_ch, G_ch and D_ch.</td>
</tr>
<tr>
<td>140</td>
<td>Executes the high-speed spot measurement (Id).</td>
</tr>
<tr>
<td>150</td>
<td>Disables the measurement units.</td>
</tr>
<tr>
<td>160 to 170</td>
<td>Reads the measurement data and prints the data on the screen.</td>
</tr>
</tbody>
</table>
The 4142B program modified for use with the 4155C/4156C:

10 ASSIGN @Hp415x TO 800 <-- 1
20 INTEGER G_ch,D_ch,S_ch
30 INTEGER Source,Drain,Gate,Sub,Err <-- 2
40 ! Source: GNDU
50 G_ch=2 !Gate: HPSMU (SLOT2)
60 D_ch=3 !Drain: MPSMU (SLOT3)
70 S_ch=4 !Substrate: MPSMU (SLOT4)
80 Source=1       !1:SMU1                 <-- 2
90 Drain=2        !2:SMU2                 <-- 2
100 Gate=3         !3:SMU3                 <-- 2
110 Sub=4          !4:SMU4                 <-- 2
120 OUTPUT @Hp415x;"US42" <-- 3
130 OUTPUT @Hp415x;"ACH ";Drain,D_ch       <-- 2
140 OUTPUT @Hp415x;"ACH ";Gate,G_ch        <-- 2
150 OUTPUT @Hp415x;"ACH ";Sub,S_ch         <-- 2
160 !
170 OUTPUT @Hp415x;"FMT 1"                 <-- 4
180 OUTPUT @Hp415x;"CN ";D_ch,G_ch,S_ch    <-- 4
190 OUTPUT @Hp415x;"DV ";S_ch":[",0,0,.1"  <-- 4
200 OUTPUT @Hp415x;"DV ";G_ch":[",0,3,.01"  <-- 4
210 OUTPUT @Hp415x;"DV ";D_ch":[",0,5,.1"  <-- 4
220 OUTPUT @Hp415x;"CN ";Source           <-- 5
230 OUTPUT @Hp415x;"DV ";Source;"11,0,.1" <-- 5
240 OUTPUT @Hp415x;"TI ";D_ch":"0"        <-- 4
250 OUTPUT @Hp415x;"CL"                    <-- 4
260 ENTER @Hp415x USING ";,3X,12D,X":Mdata
270 PRINT "Id(A)=";Mdata
280 OUTPUT @Hp415x;":PAGE"                 <-- 3
290 END

<table>
<thead>
<tr>
<th>Ref No.</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GPIB address is changed.</td>
</tr>
<tr>
<td>2</td>
<td>Program lines are added to assign the new channel numbers.</td>
</tr>
<tr>
<td>3</td>
<td>Program lines are added to use the FLEX command control mode.</td>
</tr>
<tr>
<td>4</td>
<td>A space is inserted between the command and the first parameter.</td>
</tr>
<tr>
<td>5</td>
<td>Program lines are added to control the channel for the source (the original 4142B program used GNDU as the source. The 4142B GNDU command does not require the control command).</td>
</tr>
</tbody>
</table>
Programming Tips

This section provides the following additional information on creating measurement programs.

- “Confirming Operation Status”
- “Improving Measurement Speed”
- “Disabling Auto Calibration”

Confirming Operation Status

To complete the measurement program, you can insert statements to check the 4155C/4156C operation status as shown below. This example checks the status caused by the statements before the :SYST:ERR? command, reads and displays the measurement data without errors, or displays an error message when an error occurs.

```plaintext
OUTPUT @Hp415x;":SYST:ERR?"
ENTER @Hp415x;Code,Msg$
IF Code=0 THEN
  OUTPUT @Hp415x;"RMD? 1"
  ENTER @Hp415x USING ",5X,13D,X";Mdata
  PRINT "I(A)=";Mdata
ELSE
  PRINT "ERROR:";Msg$
END IF
END
```

It is important to execute the operation status check before executing the TI?, TV?, or RMD? commands, which wait for the output data and reads the measurement results. If these commands are entered when the 4155C/4156C is in an error state, the 4155C/4156C will not return the measurement data, and will enter the wait state. Enter the device clear command (for example, the CLEAR command in HP BASIC). The 4155C/4156C will recover to normal state in approximately two seconds.
Improving Measurement Speed

To improve measurement speed:

- optimize the measurement range
- optimize the integration time
- use binary output format
- use the internal program memory
- use the TI?/TV? command instead of the TI/TV command

To Optimize the Measurement Range

The most effective way to improve measurement speed is to reduce the number of range changes. The limited autoranging mode is more effective than the autoranging mode. The fixed range mode is the most effective.

Check the typical value of the measurement data, select the optimum range, and use the fixed range mode.

To Optimize the Integration Time

For best repeatability and accuracy of the measurement data, the integration time and the number of averaging samples must be increased. This increases the measurement time.

For low current/voltage measurements, you will not want to decrease the integration time and averaging samples.

For medium or high current/voltage measurements, which do not need long integration time and numerous samples, decrease the parameter values of the following commands:

- **SIT/SLI command** Defines and selects the integration time.
- **AV command** Sets the number of averaging samples.

For more information regarding these commands and changing the parameter values, refer to Chapter 1 of the GPIB Command Reference.

If the measurement speed is given top priority or is more important than the measurement accuracy, disable the ADC zero function by using the AZ command. This reduces the integration time to approximately half, if the integration time is set to 10.24 msec or more.
FLEX Command Programming
Programming Tips

NOTE
The ADC zero function must be enabled to satisfy the measurement accuracy specifications.

To Use the Binary Output Format
To specify the data output format, ASCII or binary, use the FMT command.
If you select ASCII format, you can read the measurement data easily. The data transfer time will be longer than the binary data transfer time because the data length is longer in ASCII format.
If your program executes parameter measurements, sweep measurements, and so on, which outputs various measurement data, select the binary format to reduce the data transfer time. To read binary data, refer to “Reading Binary Output Data” on page 3-79.

To Use the Internal Program Memory
If your program repeats the setup and measurements for numerous devices, use the internal program memory. For these measurements, using the internal program memory reduces the command transfer time.
You can enter a maximum of 255 programs (a maximum of 100 KB) into the internal program memory. Refer to “Using Program Memory” on page 3-55.

To Use the TI?/TV? Command Instead of the TI/TV Command
If your program executes high-speed spot measurements, use the TI?/TV? command. The TI?/TV? command does not require you to enter the RMD? command, and reduces the RMD? command transfer time.
Disabling Auto Calibration

The auto calibration function triggers the self-calibration automatically every 30 minutes after the measurement. And the calibration requires to open the measurement terminals.

If you execute the automatic measurements as the batch job that may leave the connection with the device over 30 minutes after the measurements, disable the auto calibration. Otherwise the calibration may not be performed properly, or unexpected output may appear at the measurement terminals, and it may damage the device.

To disable the auto calibration, enter the `CM 0` command.
FLEX Command Programming
Programming Tips
Running 4145A/B Program Directly on 4155C/4156C
Running 4145A/B Program Directly on 4155C/4156C

This chapter describes how to directly run an 4145A/B GPIB program (non-ASP program) on the 4155C/4156C with little or no modification. To run these programs directly, you need to use the 4145 syntax command mode of the 4155C/4156C.

NOTE

To Enter into 4145 Syntax Command Mode

When the 4155C/4156C is turned on, the 4155C/4156C is always in the 4155C/4156C command mode.

To enter into the 4145 syntax command mode:

- From front-panel
  
  Set COMMAND SET field on the SYSTEM: MISCELLANEOUS screen to 4145.

- From remote control
  
  Send "*:SYST:LANGuage COMPatibility" command to the 4155C/4156C.

Usually, you can run these programs with no modification. But sometimes small modifications are required due to the following, which are described in this chapter:

- Non-supported commands
- Consideration about Differences
Non-supported Commands

The following the 4145A/B commands are not supported in the 4145B syntax command mode:

- **GL0**: Disables HP-GL
- **GL1**: Enables HP-GL overlay graphics
- **GL2**: Enables HP-GL stand-alone graphics
- **MX**: Matrix
- **SH**: Schmoo
- **SV S**: Save ASP file
- **GT S**: Get ASP file
- **DM3**: Display mode Matrix
- **DM4**: Display mode Schmoo
- **AS1**: Auto Sequence Program Start
- **AS2**: Auto Sequence Program Continue
- **AS3**: Auto Sequence Program Stop

If you have the 4145A/B programs that include any of the above commands, they will not work with the 4155C/4156C. Refer to "4145B Syntax Command Set" in GPIB Command Reference for details.
Considerations about Differences

The 4155C/4156C is different from the 4145A/B on the following points:

• Spot Measurement
• Sweep Steps in Logarithmic Step Mode
• Terminator

Spot Measurement

The 4145A/B can execute a spot measurement by setting both start and stop of the sweep to the same value, but the 4155C/4156C executes the measurement twice even if you set both start and stop of the sweep to the same value.

Sweep Steps in Logarithmic Step Mode

Calculation algorithm for primary sweep steps in logarithmic step mode is slightly different between the 4155C/4156C and the 4145A/B, so step values and number of steps may be different between the 4155C/4156C and the 4145A/B.
Terminator

If you run your program on an external controller, use <CR> + <LF> as the command terminator if you execute serial polling to read a status of the 4155C/4156C in your program.

If you use only <CR> or <LF> as command terminator, the 4155C/4156C may respond with incorrect status.

This is due to the differences of reading and parsing commands between the 4145A/B and the 4155C/4156C.

The following example and explanation gives a better understanding of this.

```
10 OUTPUT @Hp415x;"ME1"
20 REPEAT
30 Status=SPOLL(@Hp415x)
40 UNTIL BIT(Status,0)
```

<table>
<thead>
<tr>
<th>line number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>triggers measurement and clears the data ready bit (bit1) of status register.</td>
</tr>
<tr>
<td>20-40</td>
<td>waits until the data ready bit of status register is set to 1.</td>
</tr>
</tbody>
</table>

When the Terminator is only <CR>

- 4145A/B
  - At line 10:
    1. The 4145A/B starts reading data with RFD line set to false (data bus is halted) after each byte.
      In this example:
      
      \[ M \rightarrow \text{bus halted} \rightarrow E \rightarrow \text{bus halted} \rightarrow 1 \rightarrow \text{bus halted} \]
    2. After receiving 1, the 4145A/B recognizes valid command ME1, then executes ME1.
      At this time, the program is paused because the controller is trying to send <CR>, which is a terminator, but the 4145A/B has halted data bus and does not receive <CR>.
    3. After the 4145A/B triggers measurement and clears status bit1, the 4145A/B reads <CR>, then the program proceeds to next step (line 20).

The program reads the correct status at line 30.
Running 4145A/B Program Directly on 4155C/4156C
Considerations about Differences

- 4155C/4156C
  At line 10:
  1. The 4155C/4156C starts and continues reading data until reading a terminator.
     In this example, the 4155C/4156C reads **ME1<CR>**, then halts data bus.
  2. The 4155C/4156C starts executing "ME1". At the same time, the external controller can proceed to the next line, because all data of this line has transferred, then program continues.
  At line 30, controller can read status of the 4155C/4156C even if RFD line is false. RFD holdoff is not effective for serial polling.
  However, the clearing of the status register bit by line 10 may not have been completed yet, so line 30 may get the incorrect status.

When the Terminator is **<CR> + <LF>**
The example program for the 4155C/4156C performs as follows:

1. The 4155C/4156C starts and continues reading data until reading a terminator.
   In this example, the 4155C/4156C reads **ME1<CR>**, then halts data bus.
2. The 4155C/4156C executes "ME1".
   At this time, the program is paused because the controller is trying to send 
   **<LF>**, which is part of the terminator, but the 4155C/4156C has halted data bus and does not receive **<LF>**.
3. After the 4155C/4156C triggers measurement and clears the status bit1, the 4155C/4156C reads **<LF>**, then the program proceeds to next step (line 20).
   The program reads the correct status at line 30.
5  ASP-Like IBASIC Programming
ASP-Like IBASIC Programming

ASP means Auto Sequence Programming environment on the 4145A/B semiconductor parameter analyzer. The 4155C/4156C provides the programming environment like ASP by using the built-in Instrument BASIC.

This chapter describes how to create the ASP-like programs, and provides the reference of the ASP-like commands.

- “Creating A SP-like I B A S I C Programs”
- “A SP-like Commands”
Creating ASP-like IBASIC Programs

This section introduces how to easily create a program by using the typing aid softkeys in the IBASIC editor. This method of creating a program is similar to using the ASP environment on the 4145A/B semiconductor parameter analyzer.

In the IBASIC editor of the 4155C/4156C, there are several typing aid softkeys that allow you to easily create a program.

When you press the softkey, the corresponding IBASIC command is entered into the program, so you do not have to type it, but you may need to type in some parameters.

For typing aid softkeys, refer to "Keys for IBASIC" in Chapter 1, mainly "Secondary Softkeys in Edit execution status" in Chapter 1 for EXECUTE.

For the 4145A/B users, this environment is very familiar because it is similar to the Auto Sequence Program (ASP) programming environment of the 4145A/B. For most of the 4145 ASP commands, the IBASIC editor has a softkey to enter a corresponding IBASIC command.

These programs can run in IBASIC only, not on an external computer.

Step 1

Creating Programs by using the Typing Aid Softkeys

In the IBASIC editor, you can easily create programs that perform the same operations as a desired 4145 ASP program by using the typing aid softkeys. These are secondary softkeys. To display more softkeys, select More softkey.

For the ASP program shown below, let's create the corresponding IBASIC program:

```
! ASP Program:   Corresponding IBASIC Program:
1 GET P ICBVBE  10 EXECUTE (*GETSETUP 'ICBVBE.PRO'*)
2 SINGLE        20 EXECUTE (*SINGLE*)
3 SAVE D BV1    30 EXECUTE (*SAVEDATA 'BV1.DAT'*)
40 END
```

1. Select the GET SETUP secondary softkey. The following appears:

```
10 EXECUTE (*GETSETUP *)
```

You need to specify a filename in this command. At bottom of screen, enter fileName[,msus] is displayed, where msus means the mass storage unit specifier. You can specify DISK or MEMORY. Default is DISK.
ASP-Like IBASIC Programming
Creating ASP-like IBASIC Programs

2. Type a setup file name.
   
   `10 EXECUTE ("GETSETUP 'ICBVBE.PRO'")
   
   File name must be in single quotations (' '). Then press Enter.

3. Select SINGLE secondary softkey.
   
   `10 EXECUTE ("GETSETUP 'ICBVBE.PRO'")
   20 EXECUTE ("SINGLE")
   30 _

4. Select SAVEDATA secondary softkey.
   
   `10 EXECUTE ("GETSETUP 'ICBVBE.PRO'")
   20 EXECUTE ("SINGLE")
   30 EXECUTE ("SAVEDATA ")

5. Specify file name to which you want to save the measurement setup and result data.
   
   `10 EXECUTE ("GETSETUP 'ICBVBE.PRO'")
   20 EXECUTE ("SINGLE")
   30 EXECUTE ("SAVEDATA 'BV1.DAT'")
   40 END

   Finally, type **END** as above.

**NOTE**

**Setup File**

In **EXECUTE ("GETSETUP "**), you can specify a **.PRO** or **.MES** file:

- **.PRO** files are setup files created by the 4145B. The 4155C/4156C can read **.PRO** files.
- **.MES** files are setup files created by the 4155C/4156C.

In **EXECUTE ("SAVEDATA ")**, you specify a **.DAT** file, which is a file for storing the setup and measurement result data.

---

**Step 2**

**Executing the Program**

To execute the program, exit editor, then press **Run**.

The 4155C/4156C gets the setup file from the diskette, performs measurement, then saves setup and results to specified file on the diskette. However, in All IBASIC mode, no graphics results are displayed. To display results graphically, the display mode must be All Instrument mode or IBASIC Status mode.

To execute the program and display the results graphically, change the display mode to All Instrument or IBASIC Status mode, then press **Run**.

---

Creating a Longer Program

In the program below, the left side is an ASP program example from the 4145B manual.

The right side shows a program that was created by using the typing aid softkeys to enter the ASP-like commands (of the 4155C/4156C) that correspond to the original ASP commands. These softkeys allow you to easily create a program that runs on the 4155C/4156C and performs the same operations as the original ASP program.

```
1  GET P ICBVBE 10  EXECUTE ('GETSETUP 'ICBVBE.PRO'')
2  SINGLE 20  EXECUTE ('SINGLE')
3  WAIT 3 30  WAIT 3
4  GET P HFE1 40  EXECUTE ('GETSETUP 'HFE1.PRO'')
5  SINGLE 50  EXECUTE ('SINGLE')
6  WAIT 3 60  WAIT 3
7  GET P VCESAT 70  EXECUTE ('GETSETUP 'VCESAT.PRO'')
8  SINGLE 80  EXECUTE ('SINGLE')
9  WAIT 3 90  WAIT 3
10 GET P COLR 100  EXECUTE ('GETSETUP 'COLR.PRO'')
11 SINGLE 110  EXECUTE ('SINGLE')
12 WAIT 3 120  WAIT 3
13 PAUSE 130  PAUSE
14 GET P NPN1 140  EXECUTE ('GETSETUP 'NPN1.PRO'')
15 SINGLE 150  EXECUTE ('SINGLE')
16 PAUSE 160  PAUSE
17 PLOT 100,100,7000,7000 170  EXECUTE ('PRINTPLOT')
18 GET P BV 180  EXECUTE ('GETSETUP 'BV.PRO'')
19 SINGLE 190  EXECUTE ('SINGLE')
20 PLOT 100,100,7000,7000 200  EXECUTE ('PRINTPLOT')
21 PAUSE 210  PAUSE
22 SINGLE 220  EXECUTE ('SINGLE')
23 CPLOT 100,100,7000,7000 230  EXECUTE ('CURVEPLOT')
240 END
```

**NOTE**

**Print/Plot**

EXECUTE ("PRINTPLOT") prints/plots the information of the present instrument screen, not the IBASIC screen. If present page is GRAPH/LIST: GRAPHICS page, the graph is printed/plotted.

EXECUTE ("CURVEPLOT") changes to the GRAPH/LIST: GRAPHICS page, then prints/plots the graph.

You need to set the desired settings on the SYSTEM: PRINT/PLOT SETUP page and PRINT/PLOT dialog before “PRINTPLOT” or “CURVEPLOT” is performed.

To execute the program that sequentially performs “SINGLE” and “PRINTPLOT” or “SINGLE” and “CURVEPLOT” like above example (see lines 190 to 200 and 220 to 230), set the display mode to All Instrument, and then press Run. If you execute the program in the IBASIC Status mode, the program starts printing/plotting without waiting for the measurement completion and causes error.
ASP-Like IBASIC Programming
Creating ASP-like IBASIC Programs

Programming Tips
This section describes features and tips of IBASIC programs in relation to ASP programs. Some examples use an example measurement setup file named "V TH.M ES". Before executing these examples, you need to save setup data to a file named "V TH.M ES" on the diskette. For an example setup, see “Example Application Setup for Vth Measurement” in Chapter 2.

File Name Variables
You can specify a string variable for the file name in SA VEDATA as follows:

Filename$="DATA1.DAT"
EXECUTE ("SAVEDATA Filename$")

This feature allows you to create a more simple program as follows.

Example ASP Program
Following ASP program gets a setup file, makes measurements, and saves results to following files: VTH1, VTH2, ... VTH10. Program is 21 lines.

```
1 GET P VTH
2 SINGLE
3 SAVE D VTH1
4 SINGLE
5 SAVE D VTH2
6 SINGLE

: 21 SAVE D VTH10
```

Corresponding IBASIC Program.
The following Instrument BASIC (IBASIC) program does the same operation as the above ASP program. The program is simplified by using a filename variable Filename$ and the FOR NEXT keyword.

```
10 EXECUTE ("GETSETUP 'VTH.PRO'")
20 FOR I=1 TO 10
30 EXECUTE ("SINGLE")
40 Filename$="VTH"&VAL$(I)&".DAT"
50 EXECUTE ("SAVEDATA Filename$")
60 NEXT I
```

In line 40, the Filename$ is defined. For example, Filename$="VTH1.DAT" when I=1. So, the 21-line ASP program can be converted to a 6-line IBASIC program.
ASP-Like IBASIC Programming
Creating ASP-like IBASIC Programs

Reading 4155/56 Data to IBASIC Variables
You can transfer read-out function values or data variable values (source data,
measurement data, and user function values) from the 4155/56 to Instrument BASIC
(IBASIC) variables.
Transferring Multiple Data
You can transfer multiple data (such as sweep measurement data) to an array
variable of IBASIC by using EXECUTE ("READDATAVAR ") as follows:
EXECUTE ("READDATAVAR 'ID','Id_data'")
The above example transfers the drain current data ID of a sweep measurement to
the array variable previously defined as Id_data.
Following example program gets VTH.MES setup file, performs measurement, then
transfers ID data to an array. In this example, the array Id_data is defined in line
10, and it has elements 1 to 51.
10
20
30
40
50
60
70
80

DIM Id_data(1:51)
EXECUTE ("GETSETUP 'VTH.MES'")
EXECUTE ("SINGLE")
EXECUTE ("READDATAVAR 'ID','Id_data'")
FOR I=1 TO 51
PRINT "Id(";I;")=";Id_data(I);"A"
NEXT I
END

Result with example measurement data is as follows:
Id(1)=
Id(2)=
Id(3)=
Id(4)=

0.00031 A
0.00282 A
0.00514 A
0.01017 A
:
Id(51)= 0.08274 A


5-7


ASP-Like IBASIC Programming
Creating ASP-like IBASIC Programs

Transferring a Single Data

In the following example, a single data is to be transferred to a variable. For example, \texttt{VTH} is a single data point calculated by a user function that was defined by the user.

\begin{verbatim}
EXECUTE ("READDATAVAR 'VTH','Vthdata'")
\end{verbatim}

In the following example, \texttt{EXECUTE ("READDATAVAR")} is used to transfer the \texttt{VTH} value to the IBASIC variable \texttt{Vthdata}. And for example, \texttt{VTH.MES} is a setup file that includes auto analysis setup to extract a threshold voltage \texttt{VTH}.

\begin{verbatim}
10 EXECUTE ("GETSETUP 'VTH.MES'")
20 EXECUTE ("SINGLE")
30 EXECUTE ("READDATAVAR 'VTH','Vthdata'")
40 PRINT "Vthdata\=\;Vthdata;"V"
50 END
\end{verbatim}

Result will be for example:

\texttt{Vthdata= 1.2345 V}

You can also specify a read out function as the item to be transferred:

\begin{verbatim}
EXECUTE ("READDATAVAR '@MX','Vthdata'")
\end{verbatim}

\texttt{@MX} is the read out function that reads X-axis value of point where marker is located.

Auto Scaling

Auto scaling can be done by using the following:

\begin{verbatim}
EXECUTE ("AUTOSCALE")
\end{verbatim}

In the following example, the image dumps will be scaled for best fit to the printer or plotter even if the measurement results vary greatly.

\begin{verbatim}
10 EXECUTE ("GETSETUP 'VTH.MES'")
11 FOR I=1 TO 100
20 EXECUTE ("SINGLE")
30 EXECUTE ("AUTOSCALE")
40 EXECUTE ("PRINTPLOT")
41 NEXTI
50 END
\end{verbatim}
4145 ASP and 4155C/4156C Corresponding Keywords

Following shows the 4145A/B's ASP keywords and corresponding 4155C/4156C keywords. In IBASIC editor, there are typing aid softkeys to help you quickly enter the related 4155C/4156C keyword, which must be used in the `EXECUTE` directive:

**Corresponding 4145 ASP and 4155C/4156C Keywords**

<table>
<thead>
<tr>
<th>4145A/B</th>
<th>4155C/4156C</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET P</td>
<td>GETSETUP</td>
<td>Gets setup .MES or .PRO file.</td>
</tr>
<tr>
<td>SINGLE</td>
<td>SINGLE</td>
<td>Initiates single measurement.</td>
</tr>
<tr>
<td>SAVE DATA</td>
<td>SAVEDATA</td>
<td>Saves data to .DAT file.</td>
</tr>
<tr>
<td>PLOT</td>
<td>PRINTPLOT</td>
<td>Prints/plots present instrument page.</td>
</tr>
<tr>
<td>C PLOT</td>
<td>CURVEPLOT</td>
<td>Prints/plots measurement graph.</td>
</tr>
<tr>
<td>PRINT</td>
<td>PRINTPLOT</td>
<td>Prints/plots present instrument page.</td>
</tr>
<tr>
<td>PAUSE</td>
<td>Use BASIC keyword PAUSE</td>
<td></td>
</tr>
<tr>
<td>WAIT</td>
<td>Use BASIC keyword WAIT</td>
<td></td>
</tr>
<tr>
<td>PAGE</td>
<td>Set in the Print/Plot setup</td>
<td></td>
</tr>
<tr>
<td>STANDBY</td>
<td></td>
<td>Sets Standby status on or off.</td>
</tr>
<tr>
<td>STRESS</td>
<td></td>
<td>Initiates stress force.</td>
</tr>
<tr>
<td>AUTOSCALE</td>
<td></td>
<td>Scales dump for best fit.</td>
</tr>
<tr>
<td>READDATA</td>
<td>VAR</td>
<td>Gets data variable from 4155C/56C.</td>
</tr>
<tr>
<td>DEFUSERVAR</td>
<td></td>
<td>Defines user variable.</td>
</tr>
</tbody>
</table>

For `WAIT` and `PAUSE` of the 4145's ASP, there are no related typing aid softkeys. You type in the IBASIC keywords (`WAIT` and `PAUSE`).

For more information about IBASIC Keywords, use help functions or refer to Instrument BASIC Users Handbook.
ASP-like IBASIC Programming

ASP-like Commands

EXECUTE is an IBASIC keyword for executing function directives, which allow you to easily create simple programs in a way similar to creating Auto Sequence Programs (ASP) on the 4145A/B Semiconductor Parameter Analyzer.

NOTE

Compatibility Consideration

EXECUTE is not a standard IBASIC or HP BASIC keyword. So, if you use this keyword in your program, it will not execute on another IBASIC or HP BASIC system.

EXECUTE

Keyboard Executable Yes
Programmable Yes
In an IF . . . THEN . . . Yes

This keyword can execute the function directives that are described on the following pages.

Syntax

EXECUTE ("directive_keyword [ parameter]")

The following pages describe the directives that can be used in the EXECUTE command.

Textual Notation

[ ] Square brackets are used to enclose optional information not required for execution of the command sequence.

| The vertical bar can be read as "or" and is used to separate alternative parameter options.
GET SETUP Directive
This directive loads the specified 4155C/4156C setup file.

Directive syntax
GETSETUP file_name [, DISK | MEMORY]

Example
EXECUTE ("GETSETUP 'SWEEP.MES'")
EXECUTE ("GETSETUP 'SWEEP.MES','DISK'")
EXECUTE ("GETSETUP File$,'DISK'")
EXECUTE ("GETSETUP 'MEM1.MES','MEMORY'")

Parameter Type Explanation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>file_name</td>
<td>string</td>
<td>name of setup file with extension (.MES or .STR) to be loaded. You must enclose the name with single quotes or double-double quotes.</td>
</tr>
<tr>
<td>DISK</td>
<td>character</td>
<td>(default) loads setup data from a diskette into the built-in flexible disk drive.</td>
</tr>
<tr>
<td>MEMORY</td>
<td>character</td>
<td>loads setup data from internal memory.</td>
</tr>
</tbody>
</table>

SINGLE Directive
This directive executes measurement.

Directive syntax
SINGLE

Example
EXECUTE ("SINGLE")
ASP-Like IBASIC Programming
ASP-like Commands

STANDBY directive
This directive changes STBY ON channels to standby state or idle state.

Directive syntax
STANDBY ON | OFF

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ON</td>
<td>character</td>
<td>changes STBY ON channels from idle state to standby state.</td>
</tr>
<tr>
<td>OFF</td>
<td>character</td>
<td>changes STBY ON channels from standby state to idle state.</td>
</tr>
</tbody>
</table>

Example
EXECUTE ("STANDBY ON")
EXECUTE ("STANDBY OFF")

STRESS Directive
This directive forces stress.

Directive syntax
STRESS

Example
EXECUTE ("STRESS")

AUTO-SCALE Directive
This directive changes page to GRAPH/LIST: GRAPHICS and executes auto-scaling function.

Directive syntax
AUTOSCALE

Example
EXECUTE ("AUTOSCALE")
SAVE DATA Directive

This directive stores measurement data file to a diskette into the built-in flexible disk drive or internal memory.

**Directive syntax**

```
SAVEDATA file_name [ , DISK | MEMORY ]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>file_name</td>
<td>string</td>
<td>name of measurement data file with extension (.DAT) to be stored. You must enclose the name with single quotes or double-double quotes.</td>
</tr>
<tr>
<td>DISK</td>
<td>character</td>
<td>(default) stores measurement data to a diskette into the built-in flexible disk drive.</td>
</tr>
<tr>
<td>MEMORY</td>
<td>character</td>
<td>stores measurement data to internal memory.</td>
</tr>
</tbody>
</table>

**Example**

```
EXECUTE ("SAVEDATA 'SWEEP.DAT'"")
EXECUTE ("SAVEDATA 'SWEEP.DAT','DISK'"")
EXECUTE ("SAVEDATA File$,'DISK'"")
EXECUTE ("SAVEDATA 'MEM1.DAT','MEMORY'"")
```
ASP-Like IBASIC Programming
ASP-like Commands

**READ DATA VARIABLE Directive**

This directive gets values of specified 4155C/4156C data variable, and stores the values in an IBASIC variable.

**Directive syntax**

```
READDATAVAR data_variable_name, ibasic_variable_name
```

**Directive parameter**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>data_variable_name</td>
<td>string</td>
<td>name of the data variable of the 4155C/4156C. You must enclose the name with single quotes or double-double quotes. Name is case sensitive.</td>
</tr>
<tr>
<td>ibasic_variable_name</td>
<td>string</td>
<td>name of numeric variable or numeric array of IBASIC program. ibasic_variable_name is not case sensitive.</td>
</tr>
</tbody>
</table>

**Example**

```
EXECUTE ("READDATAVAR 'V1', 'V'")
```
**DEFINE USER VARIABLE Directive**

This directive defines an 4155C/4156C user variable, and transfers values from an IBASIC variable to the user variable.

**Directive syntax**

`DEFUSERVAR user_variable_name, no_of_points, ibasic_variable_name [ , unit ]`

**Directive parameter**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>user_variable_name</td>
<td>string</td>
<td>user variable name that you want to define. You must enclose the name with single quotes or double-double quotes.</td>
</tr>
<tr>
<td>no_of_points</td>
<td>numeric</td>
<td>number of data for the user variable.</td>
</tr>
<tr>
<td>ibasic_variable_name</td>
<td>string</td>
<td>name of numeric variable or numeric array of IBASIC program. The data in this variable will be transferred to the user variable.</td>
</tr>
<tr>
<td>unit</td>
<td>string</td>
<td>unit of user variable. You must enclose the unit with single quotes or double-double quotes.</td>
</tr>
</tbody>
</table>

**Example**

`EXECUTE ("DEFUSERVAR 'U_var',101,'Vth','V'")`
ASP-Like IBASIC Programming
ASP-like Commands

PRINT/ PLOT Directive
This directive prints/plots the information of the present instrument page, not the
IBASIC screen. If present page is GRAPH/LIST: GRAPHICS page, the graph is
printed/plotted.

You need to set the desired settings on the SYSTEM: PRINT/PLOT SETUP page
and PRINT/PLOT dialog before executing EXECUTE ("PRINTPLOT").

Directive syntax  PRINTPLOT

Example  EXECUTE ("PRINTPLOT")

CURVE PLOT Directive
This directive changes to GRAPH/LIST: GRAPHICS page, then prints/plots the
graph.

You need to set the desired settings on the SYSTEM: PRINT/PLOT SETUP page
and PRINT/PLOT dialog before executing EXECUTE ("CURVEPLOT").

Directive syntax  CURVEPLOT

Example  EXECUTE ("CURVEPLOT")

NOTE  To Execute “PRINTPLOT” or “CURVEPLOT”

To execute the program that sequentially performs “SINGLE” and “PRINTPLOT”
or “SINGLE” and “CURVEPLOT” as shown in the example below, set the display
mode to All Instrument, and then press Run. If you execute the program in the
IBASIC Status mode, the program starts printing/plotting without waiting for the
measurement completion and causes error.

10 EXECUTE ("GETSETUP 'SWEEP.MES'")
20 EXECUTE ("SINGLE")
30 EXECUTE ("PRINTPLOT")
40 END

10 EXECUTE ("GETSETUP 'SWEEP.MES'")
20 EXECUTE ("SINGLE")
30 EXECUTE ("CURVEPLOT")
40 END