

## *DCS, DLM, PRO-T Series*

### **M9 Option Programming Manual**

**Elgar Electronics Corporation**  
9250 Brown Deer Road  
San Diego, CA 92121-2294

**1-800-73ELGAR (1-800-733-5427)**  
**Tel: (858) 450-0085**  
**Fax: (858) 458-0267**  
**Email: [sales@elgar.com](mailto:sales@elgar.com)**



©2007 by Elgar Electronics Corporation

This document contains information proprietary to Elgar Electronics Corporation. The information contained herein is not to be duplicated or transferred in any manner without prior written permission from Elgar Electronics Corporation.



# CONTENTS

SECTION 1 FEATURES, FUNCTIONS, AND SPECIFICATIONS .....	1-1
1.1 Introduction.....	1-1
1.2 Features and Functions.....	1-1
1.3 Specifications.....	1-2
SECTION 2 CONFIGURATION .....	2-1
2.1 Setup Procedure.....	2-1
2.2 Rear Panel Configuration Switch.....	2-2
2.2.1 Remote/Local Selection.....	2-3
2.2.2 Power-On GPIB Service Request (PON SRQ) Selection.....	2-4
2.2.3 Master/Auxiliary Selection .....	2-4
2.2.4 Address Selection.....	2-4
2.3 External User Control Signal Connector.....	2-5
2.4 Remote Programming Via RS-232 .....	2-7
2.5 Extended Interface Bus (EIB) with the M85 Option .....	2-8
SECTION 3 IEEE 488.2 AND SCPI COMMAND OPERATION .....	3-1
3.1 Introduction.....	3-1
3.2 IEEE-488.2 Register Definitions .....	3-1
3.2.1 SCPI Status Byte .....	3-1
3.2.2 Standard Event Status Register (ESR).....	3-2
3.2.3 Protection Condition and Protection Event Status Register .....	3-3
3.2.4 Operation Status and Questionable Status Registers .....	3-3
3.2.5 Error/Event Queue.....	3-3
3.2.6 Serial Poll Operation.....	3-6
3.3 IEEE-488.2 and SCPI Conformance Information .....	3-6
3.3.1 Parameter Definitions .....	3-6

3.3.2	Units .....	3-7
3.3.3	Conventions .....	3-7
3.3.4	Queries.....	3-7
3.4	IEEE-488.2 Common Command Subsystem .....	3-8
3.5	CALIBRATION SCPI Command Subsystem.....	3-10
3.5.1	CALIBRATION SCPI Command Summary .....	3-10
3.5.2	CALIBRATION SCPI Command Reference .....	3-11
3.6	MEASURE SCPI Command Subsystem.....	3-13
3.6.1	MEASURE SCPI Command Summary .....	3-13
3.6.2	MEASURE SCPI Command Reference .....	3-13
3.7	OUTPUT SCPI Command Subsystem.....	3-14
3.7.1	OUTPUT SCPI Command Summary .....	3-14
3.7.2	OUTPUT SCPI Command Reference .....	3-14
3.8	SOURCE SCPI Command Subsystem .....	3-15
3.8.1	SOURCE SCPI Command Summary.....	3-15
3.8.2	SOURCE SCPI Command Reference .....	3-16
3.9	STATUS SCPI Command Subsystem .....	3-20
3.9.1	STATUS SCPI Command Summary.....	3-20
3.9.2	STATUS SCPI Command Reference.....	3-20
3.10	SYSTEM SCPI Command Subsystem.....	3-22
3.10.1	SYSTEM SCPI Command Summary .....	3-22
3.10.2	SYSTEM SCPI Command Reference .....	3-22
3.11	TRIGGER SCPI Command Subsystem .....	3-24
3.11.1	TRIGGER SCPI Command Summary.....	3-24
3.11.2	TRIGGER SCPI Command Reference .....	3-24
3.12	Examples of Using the SCPI Commands .....	3-25
SECTION 4 CALIBRATION .....		4-1
4.1	Introduction .....	4-1
4.2	Setup for Calibration .....	4-2
4.3	Voltage Programming Calibration .....	4-2
4.3.1	Two-Point Voltage Programming Calibration .....	4-3
4.3.2	Five-Point Voltage Programming Calibration .....	4-4
4.4	Voltage Measurement/Readback Calibration.....	4-5
4.5	Overvoltage Protection Programming Calibration.....	4-7
4.6	Current Programming Calibration .....	4-7
4.6.1	Two-Point Current Programming Calibration .....	4-7
4.6.2	Five-Point Calibration Procedure for Current Output .....	4-9
4.7	Current Measurement/Readback Calibration.....	4-10

---

4.8	Voltage Programming and Readback Calibration.....	4-11
4.9	Current Programming and Readback Calibration.....	4-13
SECTION 5 M9A COMMAND OPERATION .....		5-1
5.1	Introduction.....	5-1
5.2	M9A GPIB Register Definitions.....	5-1
5.2.1	Status Byte and the Serial Poll Operation .....	5-1
5.2.2	Status, Accumulated Status, Fault, and Mask Registers.....	5-2
5.2.3	Error Register .....	5-3
5.3	M9A Command Subsystem Reference .....	5-4
5.4	Examples of Using the M9A Commands .....	5-10
SECTION 6 OPERATION WITH M85 OPTION .....		6-1
6.1	Introduction.....	6-1
6.2	Configuration .....	6-1
6.3	System Installation.....	6-3
6.4	RS-485 Interface.....	6-4
6.5	Programming the M85 Unit (Example) .....	6-4
SECTION 7 SCPI STATUS IMPLEMENTATION.....		7-1

LIST OF FIGURES

Figure 2–1 DCS and Pro-T Configuration Switch..... 2-2

Figure 2–2 DLM Series 3kW & 4kW Configuration Switch..... 2-2

Figure 2–3 DLM Series 600W Configuration Switch ..... 2-2

Figure 2–4 External User Connector Designation (8-pin Molex)..... 2-5

Figure 2–5 Example of Open Collector, TTL Input, and Relay Output Circuits..... 2-6

Figure 2–6 RS-232 Rear Panel RJ-11 Connector Pinout..... 2-7

Figure 2–7 M9 to PC RS-232 Connection (RJ-11 to DB-9)..... 2-7

Figure 2–8 EIB(RS485) Rear Panel RJ-11 Connector Pinout..... 2-8

Figure 2–9 M9 to M85 EIB Connection (RJ-11 to RJ-11)..... 2-8

Figure 6–1 DCS and Pro-T Configuration Switch, Auxiliary in Remote Mode..... 6-1

Figure 6–2 DLM Series 3kW & 4kW Configuration Switch, Auxiliary in Remote Mode. 6-2

Figure 6–3 DLM Series 600W Configuration Switch, Auxiliary in Remote Mode ..... 6-2

Figure 6–4 RS-485 System Interconnection with Two Auxiliaries..... 6-4

Figure 6–5 M85 Rear Panel RS-485 Connectors Pinout ..... 6-4

LIST OF TABLES

Table 6–1 Definitions of S1 Switch Settings ..... 6-3

# SECTION 1

## FEATURES, FUNCTIONS, AND SPECIFICATIONS

### 1.1 Introduction

This manual covers the M9 Remote Programming Option for the DCS, PRO-T, and DLM Series power supplies. The M9 interface card enables you to operate your Sorensen power supply from a computer via the IEEE-488.2 GPIB or RS-232 interface, allowing full remote programming control and monitoring of your power supply.

The M9 is compatible with both the M9A (which it replaces) and SCPI languages. The M9 supports the control of up to 30 auxiliary power supplies from a single GPIB/RS232 interface using the Extended Interface Bus (EIB), which uses RS-485, with the M85 option.

### 1.2 Features and Functions

#### **Features**

- 12-bit programming and 12-bit readback of voltage and current
- Programmable overvoltage protection with reset
- IEEE-488.2 and SCPI compliant command set
- User-programmable signals including Local/Remote Sense, External Polarity, and Disconnect Relay Drive
- User selectable Constant-Voltage/Constant-Current or Foldback mode, with reset
- Voltage Ramp and Current Ramp functions
- Field-upgradable firmware via RS-232
- Attachment of up to 30 supplies via EIB interface with the M85 option
- Soft calibration
- Rear panel IEEE-488.2 and RS-232 control interface
- Rear panel User Control Signal interface
- Rear panel configuration switch

**Programmable Functions**

- Output voltage and current
- Soft limits for voltage and current
- Overvoltage protection
- Output enable/disable
- Maskable fault interrupt
- Hold and trigger
- External relay control
- Full calibration

**Readback Functions**

- Actual measured voltage and current
- Voltage and current settings
- Soft voltage and current limits
- Overvoltage protection setting
- Status and Accumulated Status registers
- Programming error codes
- Fault codes
- Manufacturer, power supply model, and firmware version identification

## 1.3 Specifications

**(SUBJECT TO CHANGE WITHOUT NOTICE)**

**Programming Resolution**

Voltage: 0.03% of full scale

Current: 0.03% of full scale

Overvoltage Protection: 0.03% of full scale (full scale is 110% of max output voltage.)

**Programming Accuracy**

Voltage:  $\pm (0.1\% + 0.1\%$  of maximum output voltage)

Current:  $\pm (0.1\% + 0.4\%$  of maximum output current)\*

Overvoltage Protection:  $\pm (0.5\% + 0.5\%$  of max output voltage)

**Readback Resolution**

Voltage:  $\pm 0.03\%$  of full scale

Current:  $\pm 0.03\%$  of full scale

**Readback Accuracy**

Voltage:  $\pm (0.1\% + 0.15\%$  of full scale output voltage)

Current:  $\pm (0.1\% + 0.4\%$  of full scale output current)\*

\* After 30 minutes operation with fixed line, load, and temperature.

**Note:** Refer to the power supply manual for effects of line regulation, load regulation, and temperature on accuracy specifications.



## SECTION 2 CONFIGURATION

The M9 is installed into the supply at the factory. Use the Setup Procedure described below to configure the M9 for your system and application.

### 2.1 Setup Procedure

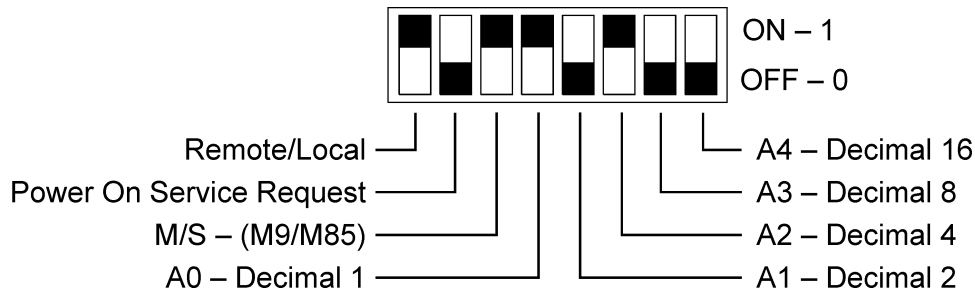
This procedure is a quick reference for the configuration requirements. Refer to Section 2.2 for detailed information on the rear panel switches.

1. Set the rear panel Remote/Local switch to Remote (On or 1).
2. Set the rear panel Power On Service Request switch to No Service Request (Off or 0).
3. Set the rear panel Master/Auxiliary (M9/M85) to Master (M9) (On or 1).  
**Note:** *This switch applies only to the DCS and Pro-T Series of power supplies. It is not used on other models.*
4. Set the GPIB/EIB address switches to the desired address.  
**Note:** *Valid GPIB addresses are 1-30. Valid EIB addresses are 2-31.*
5. Connect the GPIB or RS232 Interface Cable to the supply.
6. Connect power to the unit and turn on the unit. Verify that the green REMOTE LED on the front panel is ON.
7. Configure the controller to match the supply identification and configuration. Use one of the available programs such as IBCONF from National Instruments for GPIB, or set the RS232 baud rate to 19200, 8 data bits, no parity, and 1 stop bit.
8. Test the communication interface by issuing a \*IDN? Command. This returns the supply's model and firmware versions and does not affect the output of the supply.

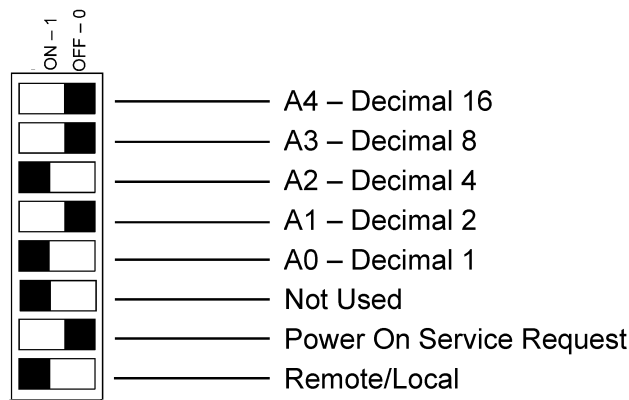
## 2.2 Rear Panel Configuration Switch

The DIP switch is accessible from the rear panel to allow configuration of the supply with the installed M9/M85 for the user's particular system and application. The following figures show the configuration, as set up in Section 2.1, and with GPIB address set to five (5).

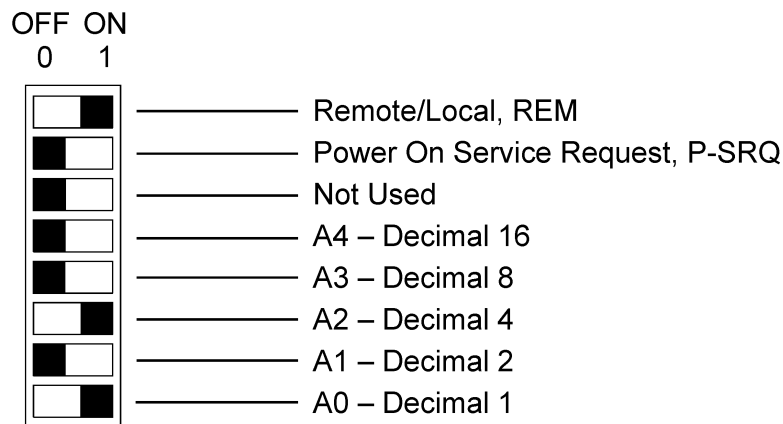
**Note:** Two types of DIP switches are utilized; toggle and rocker. For toggle switches, the shading indicates the position of the toggle switch. For rocker switches, the shading indicates the depressed side.



**Figure 2-1 DCS and Pro-T Configuration Switch**



**Figure 2-2 DLM Series 3kW & 4kW Configuration Switch**



**Figure 2-3 DLM Series 600W Configuration Switch**

### 2.2.1 Remote/Local Selection

Set the rear panel Remote/Local switch to select remote or local operation.

#### REMOTE/LOCAL SWITCH

Switch Position	Description
ON	Remote operation selected. *
OFF	Local operation selected. Front panel control is enabled. Unit will switch to remote operation upon the first GPIB or RS-232 command.

- \* For DCS firmware prior to version 2.07, and DLM firmware prior to version 1.04 – In the ON position, the power hardware and GPIB card initialize to the remote state on power turn on. Initially, control is accessible only through GPIB or RS232. Note that control may still revert to front panel control if the appropriate command is sent, or the REN of the GPIB is disabled.

For DCS firmware version 2.07 and above, and DLM firmware version 1.04 and above – In the ON position, the power hardware and GPIB card initialize to the remote state on power turn on. In addition, front panel control remains disabled regardless of the state of the REN line, or the GTL command. The special command SYST[n]:LOCAL <on/off> is now permitted as a means to revert to front panel control if desired.

Powering up in remote mode will result in the following operating conditions.

#### REMOTE MODE POWER-ON CONDITIONS

Condition	Default
Voltage	0 Volts (initial power-on voltage). Also see CAL:INIT:VOLT
Current	0 Amps (initial power-on current). Also see CAL:INIT:CURR
Soft Voltage Limit	Model maximum voltage
Soft Current Limit	Model maximum current
OVP Trip Voltage	Model maximum voltage +10% (initial power-on OVP). See CAL:INIT:VOLT:PROT
Delay	0.5 seconds
Foldback Protection	OFF
Output	ON
Hold	OFF
Unmask	NONE
Service Request Capability	OFF

### 2.2.2 Power-On GPIB Service Request (PON SRQ) Selection

Set the rear panel PON SRQ switch to ON to cause a GPIB service request to be sent to the computer controller when the supply is turned on. The front panel SRQ LED will also turn on. You may clear the service request and turn off the SRQ LED by issuing a serial poll.

#### POWER-ON GPIB SERVICE REQUEST (PON SRQ) SWITCH

Switch Position	Description
ON	Power-On SRQ selected
OFF	No Power-On SRQ selected

Refer to your specific GPIB controller card manual for further details on serial polling.

### 2.2.3 Master/Auxiliary Selection

[Applies to DCS and PROT series only!]

For the M9, set the rear panel Master/Auxiliary switch to ON to configure the supply to be the master. The address selection for a master is the GPIB address (1-30). For the M85, set the Master/Auxiliary switch to OFF to configure the supply to be an auxiliary unit. The address selection for an auxiliary unit is the channel number of that device (2-31). Refer to Section 2.2.4 for more details regarding address selection.

#### MASTER/AUXILIARY SWITCH

Switch Position	Description
ON	Master unit (M9 only)
OFF	Auxiliary unit (M85 only)

### 2.2.4 Address Selection

The address selection is binary with switch A0 as the LSB and A4 as the MSB. The rear panel switch illustration in Section 2.2 shows the address selection 00101 in binary (5 decimal).

The address selection for a master unit is the GPIB address of that device (1-30). The address selection for an auxiliary unit is the channel number of that device (2-31). Channel selections 0 and 1 are invalid for an auxiliary device because SCPI reserves channel 0 as the global channel to address all channels and channel 1 as the default master channel to address the master unit.

#### ADDRESS SWITCHES

Switch Position	Description
ON	1
OFF	0

## 2.3 External User Control Signal Connector

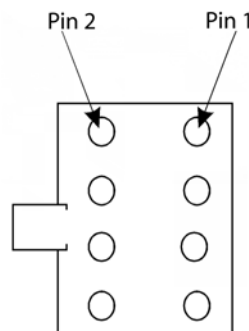
An 8-pin Molex or 9-pin Subminiature-D connector located at the rear panel provides external auxiliary control signals to increase the user's operating control of the supply. The mating receptacle for the 8-pin connector is Molex 43025-0800 with 8 female terminals 43030-0003. The Molex terminals accommodate wire sizes from #20 - #24.

The relay outputs, when active, connect the POLARITY, SENSE, and ISOLATION pins of the connector to the relay COMMON pin. The relays are rated at 120VAC/125VDC @ 1A; for DLM Series 600W, maximum ratings are 60V(PK), 30VA, and 1A.

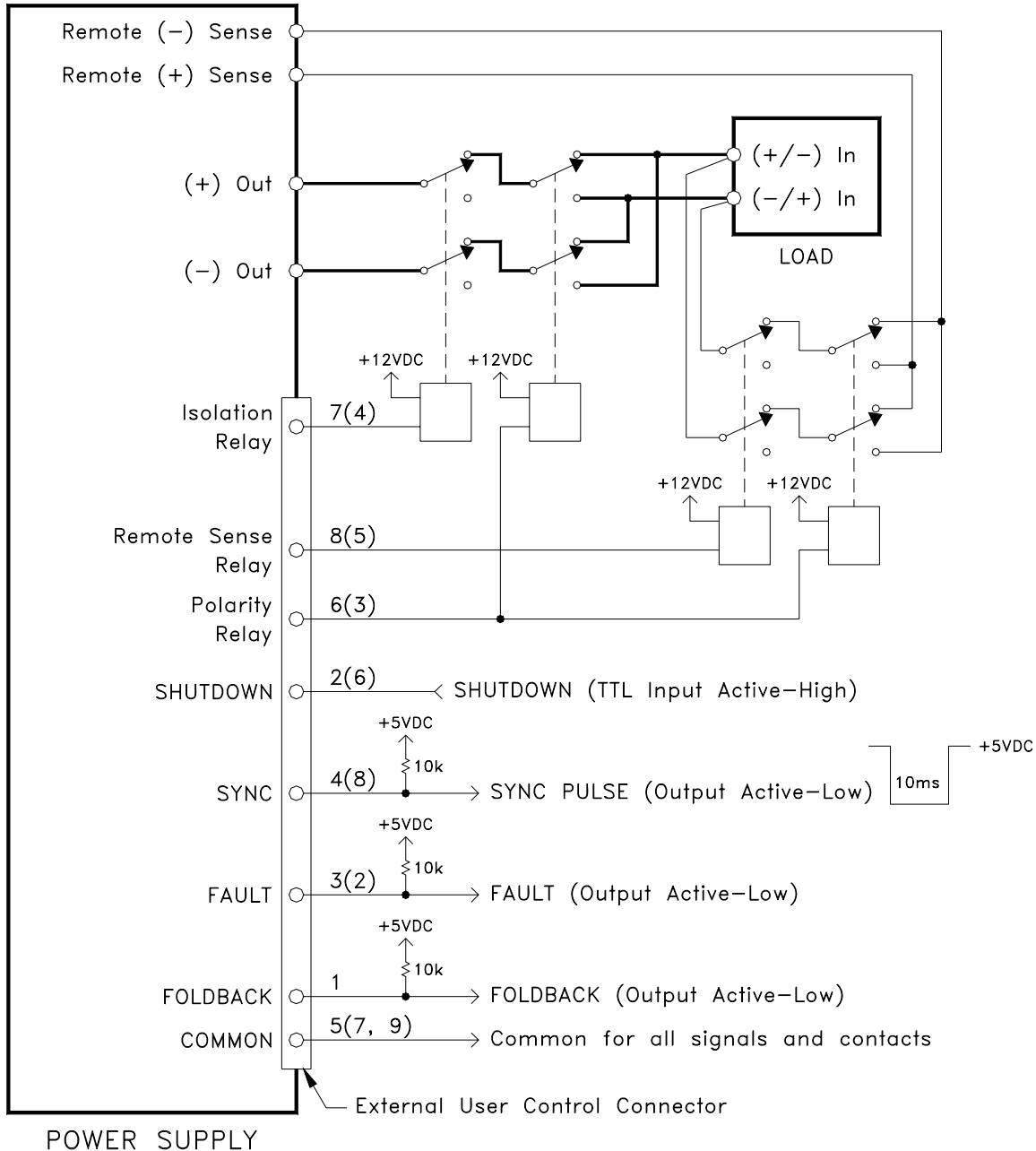
Any change in output (voltage, current, etc.) initiated by the user from the RS-232 or GPIB interface will cause a 10ms synchronization pulse to be generated at the rear panel User Control Signal Connector of the master unit (Sync Signal).

### EXTERNAL USER CONTROL SIGNAL CONNECTOR PINOUT

Sub-D Pin	Molex Pin	Description
1	1	FOLDBACK output signal, open collector, active-low. Asserted when in foldback mode.
2	3	FAULT output signal, open collector, active-low. Asserted when a fault is recorded in the fault register.
3	6	POLARITY output signal, relay contacts. Asserted (contacts close to COMMON) when a negative voltage is programmed. (e.g., SOURce:VOLTage -5.0)
4	7	ISOLATION output signal, relay contacts. Asserted (contacts close to COMMON) when the output relay is programmed OFF. (e.g., OUTPut:ISOLation OFF)
5	8	SENSE output signal, relay contacts. Asserted (contacts close to COMMON) when the sense relay is programmed OFF. (e.g., OUTput:SENse OFF)
6	2	SHUTDOWN TTL input signal, active-high. Allows the user to immediately shutdown the unit with a TTL input signal capable of sourcing a minimum of 1mA, such as the HCTTL Logic family parts.
7	5	COMMON for all signals and relay contacts.
8	4	SYNC output signal, open collector, active-low. Pulsed for 10 ms when a change in the output occurs.
9	—	COMMON for all signals and relay contacts.



**Figure 2–4 External User Connector Designation (8-pin Molex)**



**Figure 2-5 Example of Open Collector, TTL Input, and Relay Output Circuits**

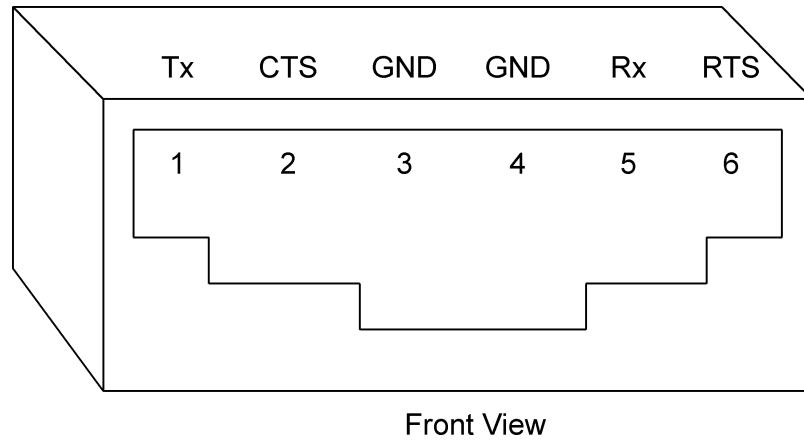


**CAUTION**

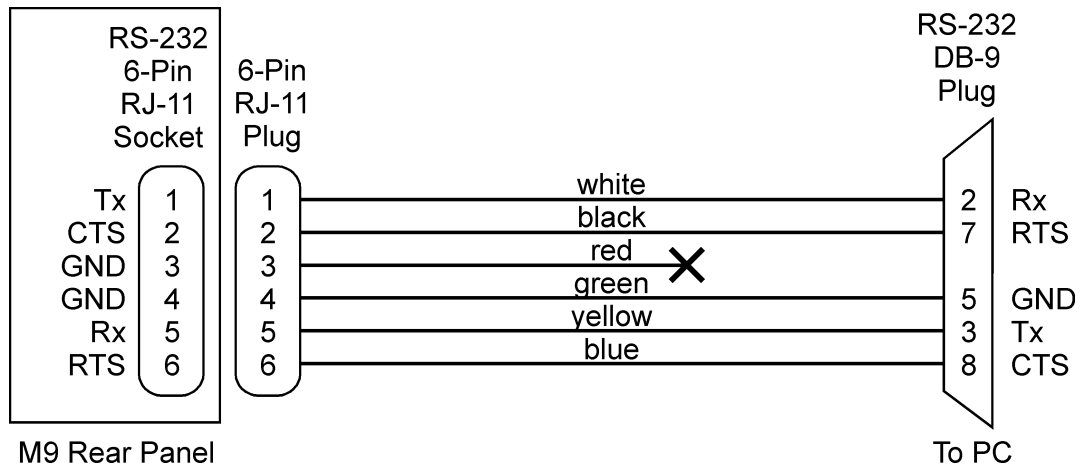
The relays must not be hot-switched; ensure that the voltage across the relay contacts and the current through them is zero prior to changing the relay states.

## 2.4 Remote Programming Via RS-232

The M9's RS-232 interface operates at 19.2Kbaud with 8 data bits, no parity, and 1 stop bit. All M9 commands are supported at the RS-232 interface with the exception of the Service Request (SRQ) function which is a GPIB-specific function requiring the dedicated Service Request line of the IEEE-488.2 interface. In this case, the SRQ function has no effect. The RS-232 interface is accessible through the rear panel 6-pin RJ-11 connector; on DLM Series 600W, the connector is labeled RS232(485).



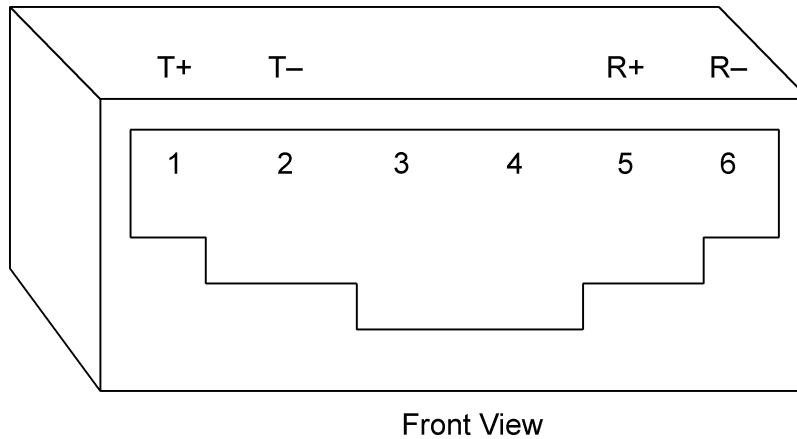
**Figure 2-6 RS-232 Rear Panel RJ-11 Connector Pinout**



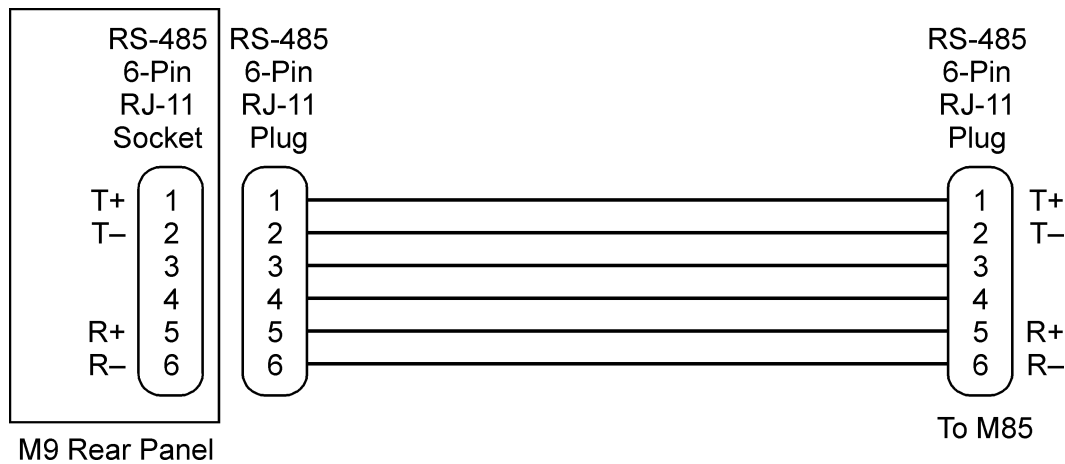
**Figure 2-7 M9 to PC RS-232 Connection (RJ-11 to DB-9)**

## 2.5 Extended Interface Bus (EIB) with the M85 Option

The M85 option is an auxiliary controller used in conjunction with the M9 to provide multiple sources controlled by a single GPIB address or via a single RS-232 connection. Up to 30 power supplies can be attached to a single M9 via the EIB(RS-485) bus connection at the rear panel of the M9. The EIB(RS-485) is accessible through the rear panel 6-pin RJ-11 connector.



**Figure 2-8 EIB(RS485) Rear Panel RJ-11 Connector Pinout**



**Figure 2-9 M9 to M85 EIB Connection (RJ-11 to RJ-11)**



## SECTION 3

# IEEE 488.2 AND SCPI COMMAND OPERATION

### 3.1 Introduction

The following sections describe the operation of the M9 by remote programming using the M9 IEEE-488.2 and SCPI command sets. The M9 IEEE488.2 and SCPI command sets provide programming, query, and status commands that facilitate remote control of the power supply.

### 3.2 IEEE-488.2 Register Definitions

The M9 supports the IEEE-488.2 and SCPI 1995.0 status reporting data structures. These structures are comprised of status registers and status register enable mask pairs. The following sections describe these pairs.

#### 3.2.1 SCPI Status Byte

The SCPI Status Byte status register can be read by the \*STB? command or by issuing a GPIB serial poll. The Status Byte status register can be cleared by the use of the \*CLS command.

The M9 can be configured to request service from the GPIB controller by setting the appropriate bits in the Service Request Enable Register (SRE). The SRE register has the same bit pattern as the Status Byte. It is modified using the \*SRE <mask> command and read with the \*SRE? command. For example, if the SRE register is set to 0x10 (MAV), when the M9 unit has a message available, the Status Byte register will contain 0x50 (RQS and MAV) and the SRQ line of the GPIB will be asserted to indicate a request for service. See table below, and refer to Section 7 for further information.

## SCPI STATUS BYTE

Bit	Hex Value	Description
0	0x01	Not used.
1	0x02	Protection Event Status flag. Indicates the selected protection event occurred.
2	0x04	Error/event queue message available. Set when any error/event is entered in the System Error Queue. It is read using the SYSTem:ERRor? query.
3	0x08	Questionable Status flag. Indicates the quality of the current data being acquired. This bit is not used.
4	0x10	Message available (MAV). Indicates a message is available in the GPIB output queue. Cleared after the GPIB output buffer is read.
5	0x20	Standard Event Status Register (ESR). Summary bit for the ESR. Set when any of the ESR bits are set and cleared when the ESR is read.
6	0x40	Request Service flag (RQS) for serial polling or Master Summary Status (MSS) in response to *STB? If service requests are enabled (with the *SRE command), this bit represents the RQS and will be sent in response to a serial poll, then cleared. If RQS is not enabled, the bit represents the MSS bit and indicates the device has at least one reason to request service. Even though the device sends the MSS bit in response to a status query (*STB?), it is not sent in response to a serial poll. It is not considered part of the IEEE-488.1 Status Byte.
7	0x80	Operation Status flag. Indicates the current operational state of the unit. This bit is not used.

## 3.2.2 Standard Event Status Register (ESR)

The Standard Event Status Register (ESR) can be read by the \*ESR? command. Reading this register or issuing a \*CLS command will clear the ESR. Use the \*ESE (Standard Event Status Enable Register) to enable corresponding ESR bits to be summarized in the summary bit of the SCPI Status byte. To configure the M9 to generate GPIB service requests based on the ESR, both the Standard Event Status Enable Register and the Service Request Enable Register must be programmed. See table below, and Section 7 for further information.

## STANDARD EVENT STATUS REGISTER

Bit	Hex Value	Description
0	0x01	Operation Complete
1	0x02	Request Control - not used
2	0x04	Query Error
3	0x08	Device Dependent Error
4	0x10	Execution Error (e.g., range error)
5	0x20	Command Error (e.g., syntax error)
6	0x40	User Request - not used
7	0x80	Power On

### 3.2.3 Protection Condition and Protection Event Status Register

These two registers have the same bit meanings, but they differ in regards to how they function.

The Protection Condition Register can be read using the `STAT:PROT:COND?` command. This command gives the *present status condition* of the power hardware, so the data is not latched. It is meant to be used as a polling register.

The Protection Event Status Register can be read by the `STATus:PROTection:EVENT?` command. Reading this register, issuing a `*CLS` command, or issuing a `*RST` command will clear the Protection Event Status Register. Bits in the Protection Event Status Register will be set only when the corresponding bit in the Protection Event Status Enable Register is set and the corresponding event occurs. The status is then latched and will remain in that state until it is read or cleared due to some command action. (Use the `STATus:PROTection:ENABLE <mask>` command to set the Enable Register and the `STATus:PROTection:ENABLE?` query to read the Enable Register.) To configure the M9 to generate GPIB service requests based on the Protection Event Status Register, both the Protection Event Status Enable Register and the Service Request Enable Register (`*SRE`) must be programmed. For further information, refer to the table below, and to Section 7.

#### PROTECTION CONDITION AND EVENT STATUS REGISTERS

Bit	Hex Value	Description
0	0x01	Constant voltage operation
1	0x02	Constant current operation
2	0x04	Not used
3	0x08	Overvoltage protection tripped
4	0x10	Overtemperature protection tripped
5	0x20	Supply external shutdown active
6	0x40	Foldback mode operation
7	0x80	Remote programming error

### 3.2.4 Operation Status and Questionable Status Registers

The Operation Status and Questionable Status Registers will always return 0 when queried. The Operation Status Enable and Questionable Status Enable Registers can be programmed and queried to allow SCPI compatibility but have no effect on the Operation Status and Questionable Status Registers.

### 3.2.5 Error/Event Queue

The M9 maintains an Error/Event Queue as defined by SCPI. The queue holds up to 10 error events. It is queried using the `SYSTem:ERRor?` command which reads in a First In/First Out (FIFO) manner. The read operation removes the entry from the queue. The `*CLS` command will clear all entries from the queue.

The following error codes are defined in the SCPI 1995.0 specification and are supported by the M9. Error codes are in the range of [-32768, 32767]. SCPI reserves the negative error codes and 0, while error codes greater than 0 are device specific errors.

## SCPI ERROR CODES

Error Code	Description
208	<p><b>Isolation relay must open first</b></p> <p>This means an attempt to change the state of the polarity relay was made while the isolation relay was closed. First open the isolation relay before changing the state of the polarity relay.</p>
207	<p><b>Voltage sign mismatched polarity relay state</b></p> <p>This means that the algebraic sign on the voltage of a voltage programming command, such as SOUR:VOLT &lt;volt&gt;, did not match the state of the polarity relay, so the voltage command was ignored. For example, if the polarity relay is in the positive voltage position (normal output voltage position), the command SOUR:VOLT -5 will cause this error. Conversely, if the polarity relay is in the negative voltage position (inverted output voltage position), the command SOUR:VOLT 5 will cause this error.</p> <p>Note that programming Over Voltage Protection with a mismatched algebraic sign also can cause this error.</p>
206	<p><b>No channels setup to trigger</b></p> <p>This means that an attempt was made to trigger the M9 using the TRIG:TYPE &lt;1 2 3&gt; command when there are no armed trigger settings. This error is <i>not</i> generated when the GET is received, even when there are no armed trigger settings.</p>
205	<p><b>GPIB GET not allowed during message</b></p> <p>This error means that the GPIB G(roup) E(xecute) T(ripper) multiline command was errantly generated by the system computer while or very shortly after a message is or was sent. Give a few milliseconds after a message was sent before attempting a GET; and never send a GET during the midst of a message transfer over the GPIB.</p>
204	<p><b>GPIB IFC caused warm boot</b></p> <p>This error relates to the GPIB IFC signal, and is available only in association with a proprietary command.</p>
203	<p><b>Hardware watchdog warm boot</b></p> <p>This error is caused by a hardware fault either in the power supply proper, or on the M9. One possible explanation might be that the mains power to the supply was interrupted for a short but sufficient time to cause the M9 processor to reset and re-boot. Also, it might be possible to generate this error by a very momentary off action of the front panel power switch.</p>
202	<p><b>Foreground watchdog warm boot</b></p> <p>This error means that the internal firmware on the M9 found an internal error condition that halted processing; to force resumption of processing, a warm boot was required.</p>
201	<p><b>Unexpected warm boot</b></p> <p>This error means that the M9 GPIB-side processor experienced a warm boot that was unexpected, and it may indicate an internal crash of the M9 processor.</p>
102	<p><b>Incompatible unit type</b></p> <p>This error is not used. It cannot occur.</p>

101	<b>Incompatible unit version</b> This slave does not support user defined initialization values nor ramping; its firmware is too old a version to do so. Slave #n provides its firmware version number as the last number on the response from the *IDN[n]? query command.
100	<b>Incompatibility error</b> This error is not used. It cannot occur.
0	<b>No error</b> The error queue is empty.
-102	<b>Syntax error</b> An unrecognized command or data type was encountered.
-108	<b>Parameter not allowed</b> More arguments than expected were received.
-151	<b>Invalid string data</b> Incorrect password. Manufacturer, model, or serial number string was more than 16 characters. Invalid mnemonic.
-161	<b>Invalid block data</b> The expected number of data values was not received.
-200	<b>Execution error</b> An error/event number in the range [-299,-200] indicates that an error has been detected by the instruments execution control block. The occurrence of any error in this class shall cause the execution error bit (bit 4) in the Event Status Register to be set. An execution error can be the result of: <ul style="list-style-type: none"> <li>• A &lt;program data&gt; element out of range, such as programming 35 volts in a 33 volt device.</li> <li>• A command could not be executed due to the current condition of the device.</li> </ul>
-203	<b>Command protected</b> Attempted to store calibration values to EEPROM without unlocking.
-221	<b>Settings conflict</b> Attempted to set output greater than soft limits or to set soft limits less than output.
-222	<b>Data out of range</b> Parameter exceeded range of valid values.
-225	<b>Out of memory</b> There is not enough memory to perform the requested operation.
-241	<b>Hardware missing</b> A legal command or query could not be executed because the option is not installed.
-284	<b>Program currently running</b> A legal command or query could not be executed because a function is currently running.
-292	<b>Referenced name does not exist</b>
-293	<b>Referenced name already exists</b>
-316	<b>Checksum error</b>
-330	<b>Self-test failed</b> A self-test failure has occurred.

-340	<b>Calibration failed</b> Error during calculation of calibration values occurred.
-350	<b>Queue overflow</b> The error queue can contain up to 10 entries. If more than 10 error/event conditions are logged before the SYSTem:ERRor? query, an overflow will occur; the last queue entry will be overwritten with error -350. When the queue overflows, the least recent error/events remain in the queue and the most recent error/events are discarded.
-360	<b>Communication error</b> Communications to a channel was disrupted.

### 3.2.6 Serial Poll Operation

Refer to Section 7 for further information.

Performing a serial poll will not modify the Status Byte other than to clear the RQS (bit 6) for an M9 requesting service. Queries affecting the Status Registers and subsequent serial poll are described below:

- \*STB? clears the Status Byte
- \*ESR? clears the ESR and bit 5 of the Status Register
- SYSTem:ERRor? clears bit 2 of the Status Register if the queue is empty

## 3.3 IEEE-488.2 and SCPI Conformance Information

The M9 conforms to most of the specifications for devices as defined in IEEE-488.2 and SCPI Version 1995.0. Confirmed Commands are those commands that are approved commands in the SCPI 1995 Specification, Volume 2: Command Reference. They are denoted by a “C” in the “SCPI” column. Any commands that are not Confirmed Commands are labeled as Not Approved denoted by an “N.”

### 3.3.1 Parameter Definitions

PARAMETER DEFINITIONS	
Type	Valid Arguments
<boolean>	“ON” or 1. “OFF” or 0.
<NR1>	The data format <NR1> is defined in IEEE-488.2 for integers. Zero, positive and negative integer numeric values are valid data.
<0+NR1>	Zero and positive integer numeric values.
<-NR1>	Negative integer numeric values.
<NRf>	The data format <NRf> is defined in IEEE-488.2 for flexible Numeric Representation. Zero, positive and negative floating point numeric values are some examples of valid data.
<0+NRf>	Zero and positive floating point numeric values.
<-NRf>	Negative floating point numeric values.
<string>	Characters enclosed by single or double quotes.

### 3.3.2 Units

The Series M9 will accept the following units as suffixes to numeric values:

UNITS	
Type of Unit	Valid Suffix
Voltage	"VOLTS" or "volts", "V" or "v", "MV" or "mv" or "mV"
Current	"AMPS" or "amps", "A" or "a", "MA" or "ma" or "mA"
Time	"SEC" or "sec", "S" or "s", "MS" or "ms", "MIN" or "min"
Frequency	"HZ" or "hz"

The default units are VOLTS, AMPS, SEC, and HZ. For example, "SOUR:VOLT 1" programs 1 volt. To program in units of millivolts, type "SOUR:VOLT 1mV".

### 3.3.3 Conventions

SCPI uses the conventions where optional commands and parameters are enclosed by "[ ]". Additionally the shorthand version of a command is indicated by capital letters. The optional parameter "[n]" selects the auxiliary channel number (power supply) being commanded. If "[n]" is not specified, the default master channel 1 is automatically selected. Channel 1 selects the master unit and channels 2-31 select an auxiliary unit attached with the M85 option. Selecting the global channel 0 is allowed only for the TRIGGERed[n] commands.

For example,

```
SOURce[n]:VOLTage[:LEVel][:IMMEDIATE][:AMPLitude] 120.0
```

can be written as

```
SOURce[n]:VOLTage 120.0 // for any channel
```

or

```
SOUR:VOLT 120.0 // for default master channel 1
```

### 3.3.4 Queries

The query syntax is identical to the command syntax with a "?" appended. For example, to query the programmed voltage, send the string: SOURce:VOLTage?. A subsequent device read will return a value such as "33.000". All queries are terminated with a carriage return and line feed (0x0D 0x0A) for those GPIB controllers that require termination characters. When the M9 has nothing to report, its output buffer will contain two ASCII characters: a carriage return and linefeed (in decimal the values are: <13><10>).

### 3.4 IEEE-488.2 Common Command Subsystem

The following commands are common to all SCPI instruments and declared mandatory by IEEE-488.2. In the following table, the M9 is defined as the “device” on the GPIB bus.

Command	Description
*CLS[n]	Clears all status reporting data structures including the Status Byte, Standard Event Status Register, and Error Queue. The STAT:PROT:ENAB (protection event enable register) is cleared by this command; other enable registers are not cleared by this command.
*ESE[n] <0+NR1>	Sets the value of the Standard Event Status Enable Register that determines which bits can be set in the Standard Event Status Register. See section 3.2.2 for valid values.
*ESE[n]?	Returns the integer value of the Standard Event Status Enable Register. See section 3.2.2 for valid values. <b>Response:</b> <0+NR1>
*ESR[n]?	Returns the integer value of the Standard Event Status Register. The ESR and the Status Byte ESR bit are cleared. See section 3.2.2 for valid values. <b>Response:</b> <0+NR1>
*IDN[n]?	Returns the device identification as an ASCII string. <b>Response:</b> <Manufacturer>, <model>, <serial number>, <DCI firmware version> <AI firmware version> <b>Example:</b> Sorensen, DCS33-33, B90000-0, 1.00,1.00
*OPC[n]	Enables the Operation Complete bit of the Standard Event Status Register to be set when all pending operations are complete. See section 3.2.2.
*OPC[n]?	Returns the integer value “1” when all pending operations are complete. See section 3.2.2. <b>Response:</b> <0+NR1>
*RST[n]	Resets the supply to its Power ON (PON) state. Clears all status reporting data structures including the Status Byte, Standard Event Status Register, and Error Queue. The STAT:PROT:ENAB (protection event enable register) is cleared by this command; other enable registers are not cleared by this command.
*SRE[n] <0+NR1>	Sets the value of the Service Request Enable Register, which determines which bits in the Status Byte will cause a service request from the device. See section on Status Byte for valid values.
*SRE[n]?	Returns the integer value of the Service Request Enable Register. See section on Status Byte for valid values. Values range from 0-63 or 128-191. <b>Response:</b> <0+NR1>



<b>Command</b>	<b>Description</b>
*STB[n]?	<p>Returns the integer value of the Status Byte with bit 6 representing the Master Summary Status (MSS) instead of RQS. The MSS bit acts as a summary bit for the Status Byte and indicates whether the device has at least one reason to request service based on the MAV and the ESR bits. The Status Byte is cleared. See section on Status Byte for valid values. Values range from 0-255.</p> <p><b>Response:</b> &lt;0+NR1&gt;</p>
*TST[n]?	<p>Sets the device to execute an internal self-test and return the integer value of the results. Value of "0" indicates no errors.</p> <p><b>Response:</b> &lt;0+NR1&gt;</p>
*WAI[n]	<p>Sets the device to wait until all previous commands and queries are complete before executing commands following the *WAI command.</p>

## 3.5 CALIBRATION SCPI Command Subsystem

See Section 4 for calibration procedures.



### WARNING

Please refer to the power supply manual for further information before performing calibration procedures. Calibration must be performed by qualified personnel who appropriately deal with attendant hazards. If calibration is not performed properly, functional problems could arise, requiring that the supply be returned to the factory.

### 3.5.1 CALIBRATION SCPI Command Summary

CALibrate[n]

:DATA <NRf><NRf><NRf><NRf><NRf><NRf><NRf><NRf><NRf><NRf>

:INITial

:CURRent <0+NRf>

:VOLTage

[:AMPLitude] <NRf>

:PROTection <NRf>

:LOCK

:MEASure

:CURRent

:ADC?

:CALCulate

:GAIN <NRf>

:OFFSet <NRf>

:POINt <1|2> <0+NRf>

:VOLTage

:ADC?

:CALCulate

:GAIN <NRf>

:OFFSet <NRf>

:POINt <1|2> <NRf>

:OUTPut

:CURRent

:CALCulate

:DAC <0+NR1>

:GAIN <NRf>

:OFFSet <NRf>

:POINt <1|2> <NRf>

:VOLTage

:CALCulate

:DAC <0+NR1>

:GAIN <NRf>

:OFFSet <NRf>

:POINt <1|2> <NRf>

:PROTection

:CALCulate

:DAC <0+NR1>

:GAIN <NRf>

:OFFSet <NRf>

:STORe

:UNLock <string>

## 3.5.2 CALIBRATION SCPI Command Reference

Command	Description	SCPI
CALibrate[n]	Calibration subsystem. n = 1-31. The default channel is 1.	C
:DATA <NRf><NRf><NRf> <NRf><NRf><NRf> <NRf><NRf><NRf> <NRf>	Sets the values of the ten floating point calibration constants: 1) output voltage DAC gain 2) output voltage DAC offset 3) output current DAC gain 4) output current DAC offset 5) output voltage protection DAC gain 6) output voltage protection DAC offset 7) voltage measurement ADC gain 8) voltage measurement ADC offset 9) current measurement ADC gain 10) current measurement ADC offset Values are separated by space or comma.	N
:INITial		N
:CURRent	Sets the power-on default value of current.	N
:VOLTage		N
[:AMPLitude] <NRf>	Sets the power-on default voltage.	N
:PROTection <NRf>	Sets the power-on default value of the overvoltage protection.	N
:LOCK	Disables access to the non-volatile memory. Prevents attempts to store calibration values.	N
:MEASure		N
:CURRent		
:ADC?	Returns the integer value of the A/D for the current measurement.	N
:CALCulate	Calculates the value of the gain and offset for current measurements.	N
:GAIN <NRf>	Sets the value of the gain for current measurements.	N
:OFFSet <NRf>	Sets the value of the offset for current measurements.	N
:POINt <1 2> <0+NRf>	Sets the current measurement calibration point (1 or 2). The actual output current is measured with an external meter.	N
:VOLTage		N
:ADC?	Returns the integer value of the A/D for the voltage measurement.	N
:CALCulate	Calculates the value of the gain and offset for voltage measurements.	N
:GAIN <NRf>	Sets the value of the gain for voltage measurements.	N
:OFFSet <NRf>	Sets the value of the offset for voltage measurements.	N

Command	Description	SCPI
:POINT <1 2> <NRf>	Sets the voltage measurement calibration point (1 or 2). The actual output voltage is measured with an external meter.	N
:OUTPut		N
:CURRent		N
:CALCulate	Calculates the value of the gain and offset for output current.	N
:DAC <0+NR1>	Sets the output of the output current D/A converter. The value range is 0 - 4095.	N
:GAIN <NRf>	Sets the value of the gain for the output current.	N
:OFFSet <NRf>	Sets the value of the offset for the output current.	N
:POINT <1 2> <0+NRf>	Sets the current output calibration point (1 or 2). The actual output current is measured with an external meter.	N
:VOLTage		N
:CALCulate	Calculates the value of the gain and offset for output voltage.	N
:DAC <0+NR1>	Sets the output of the output voltage D/A converter. The value range is 0 - 4095.	N
:GAIN <NRf>	Sets the value of the gain for the output voltage.	N
:OFFSet <NRf>	Sets the value of the offset for the output voltage.	N
:POINT <1 2> <NRf>	Sets the voltage output calibration point (1 or 2). The actual output voltage is measured with an external meter.	N
:PROTection		N
:CALCulate	Calculates the value of the gain and offset for output overvoltage protection. This takes more than 30 seconds to complete. Use *ESE 1 and a serial poll to detect the completed operation.	N
:DAC <0+NR1>	Sets the output of the output overvoltage protection D/A converter. The value range is 0 - 4095.	N
:GAIN <NRf>	Sets the value of the gain for the output overvoltage protection.	N
:OFFSet <NRf>	Sets the value of the offset for the output overvoltage protection.	N
:STORe	Stores the calibration constants in non-volatile memory.	N
:UNLock <string>	Sets the non-volatile memory available to store calibration constants. The access string is "6867".	N

## 3.6 MEASURE SCPI Command Subsystem

### 3.6.1 MEASURE SCPI Command Summary

```

MEASure[n]
  :CURRent?
  :CURRent
    :AVERage <value>
    :AVERage?
  :VOLTage?
  : VOLTage
    :AVERage <value>
    :AVERage?

```

### 3.6.2 MEASURE SCPI Command Reference

Command	Description	SCPI
MEASure[n]	Measure subsystem. n = 1-31. The default channel is 1.	C
:CURRent?	Returns the floating point value of the DC output current in amps.	C
:CURRent		
:AVERage <value>	Enter a value of 1 to 5 to set the number of readings to average together when returning the current value from the MEAS:CURR? command. This function reduces noise in the readback readings. The (default) value of 1 provides the fastest response time, but the noisiest readings. Available for DLM Series firmware versions 1.08 and later.	—
:AVERage?	Returns the number 1 to 5 to indicate the last set number of readings to average together when taking a current reading. Available for DLM Series firmware versions 1.08 and later.	—
:VOLTage?	Returns the floating point value of the DC output voltage in volts.	C
: VOLTage		
:AVERage <value>	Enter a value of 1 to 5 to set the number of readings to average together when returning the voltage value from the MEAS:VOLT? command. This function reduces noise in the readback readings. The (default) value of 1 provides the fastest response time, but the noisiest readings. Available for DLM Series firmware versions 1.08 and later.	—
:AVERage?	Returns the number 1 to 5 to indicate the last set number of readings to average together when taking a voltage reading. Available for DLM Series firmware versions 1.08 and later.	—

## 3.7 OUTPUT SCPI Command Subsystem

### 3.7.1 OUTPUT SCPI Command Summary

OUTPut[n]

```
:ISOLation <boolean>
:POLarity <NORM/0/OFF|INV/1/ON>
:PROTection
  :DELay <0+NRf>
  :FOLD <0|1|2>
:SENSe <boolean>
:STATe <boolean>
:TRIPped?
```

### 3.7.2 OUTPUT SCPI Command Reference

Command	Description	SCPI
OUTPut[n]	Output subsystem. n = 1-31. The default channel is 1.	C
:ISOLation <boolean>	Sets the rear panel isolation relay control signal ON or OFF. Valid arguments are 1/ON or 0/OFF.	N
:POLarity <NORM/0/OFF INV/1/ON>	Changes the state of the polarity relay. This command requires that the isolation relay be open beforehand. If the isolation relay is closed when this command is attempted, the state of the polarity relay will not change, and an error message will be generated.	
:PROTection		N
:DELay <0+NRf>	Sets the programmable time delay executed by the supply before reporting output protection conditions after a new output voltage or current is specified.	N
:FOLD <0 1 2>	Sets the foldback (program down) mode of the supply. Valid arguments are 0 (OFF or do nothing, do not program down to zero), 1 (program down to zero upon entering constant-voltage mode), or 2 (program down to zero upon entering constant-current mode).	N
:SENSe <boolean>	Sets the sense relay signal open or close. Valid arguments are 1/ON or 0/OFF.	N
:STATe <boolean>	Sets the output to zero or the programmed value; opens or closes the isolation relay. Valid arguments are 1/ON or 0/OFF. *RST state value is ON. <b>CAUTION:</b> Ensure that suitable delays are incorporated to preclude hot switching of the isolation relay.	C
:TRIPped?	Returns the integer value 1 (TRIPPED) or 0 (UNTRIPPED) state of the output.	N

## 3.8 SOURCE SCPI Command Subsystem

### 3.8.1 SOURCE SCPI Command Summary

```

SOURCE[n]
  :CURRent
    [:LEVel]
      [:IMMediate]
        [:AMPLitude] <0+NRf>
      :TRIGgered
        :CLEar
          [:AMPLitude] <0+NRf>
      :LIMit
        [:AMPLitude] <0+NRf>
      :RAMP <0+NRf> <0+NRf>
      :ABORt
      :ALL?
      :TRIGgered <0+NRf> <0+NRf>
  :ONLine?
  :STATus
    :BLOCk?
    :REGister?
  :TIMEout?
  :VOLTage
    [:LEVel]
      [:IMMediate]
        [:AMPLitude] <NRf>
      :TRIGgered
        :CLEar
          [:AMPLitude] <NRf>
      :LIMit
        [:AMPLitude] <NRf>
    :PROTection
      [:LEVel] <NRf>
      :TRIPped?
      :STATe?
    :RAMP <NRf> <0+NRf>
    :ABORt
    :ALL?
    :TRIGgered <NRf> <0+NRf>

```

## 3.8.2 SOURCE SCPI Command Reference

Command	Description	SCPI
SOURce[n]	Source subsystem. n = 1-31. The default channel is 1.	C
:CURRent	Sets the output current in amps (default) or in milliamps.	C
[:LEVel]	Sets the output current in amps (default) or in milliamps.	C
[:IMMediate]	Sets the output current in amps (default) or in milliamps.	C
[:AMPLitude] <0+NRf>	Sets the output current in amps (default) or in milliamps.	C
:TRIGgered	Sets the value of the output current to be implemented by the trigger.	C
:CLEar	Clears the trigger mode.	
[:AMPLitude] <0+NRf>	Sets the value of the output current to be implemented by the trigger.	C
:LIMit	Sets an upper soft limit on the programmed output current for the supply.	C
[:AMPLitude] <0+NRf>	Sets an upper soft limit on the programmed output current for the supply.	C
:RAMP <0+NRf> <0+NRf>	Sets the output current to ramp from the present value to the specified value (first argument) in the specified time (second argument). See Ramp Function description below.	N
:ABORt	Aborts ramping and clears trigger mode.	
:ALL?	Returns the ramping status of all channels.	
:TRIGgered <0+NRf> <0+NRf>	Sets the output current to ramp from the present value to the specified value (first argument) in the specified time (second argument) upon the trigger command. See Ramp description below.	N
:ONLine?	Returns the integer value 1 (ONLINE) or 0 (OFFLINE) of the channel online status.	N
:STATus		
:BLOCk?	Returns the block of data critical to the status of the channel:  1) channel number 2) online status 3) status flags register (see table below) 4) status register (see sect. 5.2.2) 5) accumulated status (see sect. 5.2.2) 6) fault mask register (see sect. 5.2.2) 7) fault register (see sect. 5.2.2) 8) error register (see sect. 5.2.3) 9) model serial number 10) model voltage 11) model current 12) model over-voltage	N



Command	Description	SCPI
	13) output voltage DAC gain 14) output voltage DAC offset 15) output current DAC gain 16) output current DAC offset 17) output voltage protection DAC gain 18) output voltage protection DAC offset 19) voltage measurement ADC gain 20) voltage measurement ADC offset 21) current measurement ADC gain 22) current measurement ADC offset 23) model string 24) OVP calibrated  Each value is separated by commas.	
:REGister?	Returns the integer value of the status register. See section 5.2.2.	N
:TIMEout?	Returns the integer value 1 (timeout since last query) or 0 (no timeout) of the timeout status of the channel.	N
:VOLTage	Sets the output voltage of the supply in volts (default) or in millivolts. *See note on page 3-18.	C
[:LEVel]	Sets the output voltage of the supply in volts (default) or in millivolts. *See note on page 3-18.	C
[:IMMEDIATE]	Sets the output voltage of the supply in volts (default) or in millivolts. *See note on page 3-18.	C
[:AMPLitude] <NRf>	Sets the output voltage of the supply in volts (default) or in millivolts. *See note on page 3-18.	C
:TRIGgered	<p>Sets the value of the output voltage to be implemented by the trigger. For DLM Series firmware versions 1.07 and later, the polarity relay must be in the correct position per the algebraic sign of the voltage to be triggered. The state of the polarity relay may be set using the OUP:POL &lt;NORM   INV&gt; command if it is not already in the correct position. <i>Do this before the trigger command is issued.</i></p> <p>For DLM Series firmware versions 1.06 and earlier, the algebraic sign of the trigger voltage determines the state of the polarity relay <i>before</i> the voltage is triggered. Therefore, the user load could be exposed to an unintended reverse polarity voltage if the pre-trigger voltage had an opposite algebraic sign from the trigger voltage. For a pre-trigger voltage of zero, the above consideration is not an issue.</p>	C
:CLEar	Clears the trigger mode.	
[:AMPLitude] <NRf>	Sets the value of the output voltage to be implemented by the trigger. For DLM Series firmware versions 1.07 and later, the polarity relay must be in the correct position per the algebraic sign of the voltage to be triggered. The	C

Command	Description	SCPI
	state of the polarity relay may be set using the OUPP:POL <NORM   INV> command if it is not already in the correct position. <i>Do this before the trigger command is issued.</i>  For DLM Series firmware versions 1.06 and earlier, the algebraic sign of the trigger voltage determines the state of the polarity relay <i>before</i> the voltage is triggered. Therefore, the user load could be exposed to an unintended reverse polarity voltage if the pre-trigger voltage had an opposite algebraic sign from the trigger voltage. For a pre-trigger voltage of zero, the above consideration is not an issue.	
:LIMit	Sets the upper soft limit on the programmed output voltage.	C
[:AMPLitude] <NRf>	Sets the upper soft limit on the programmed output voltage.	C
:PROTection	Sets the overvoltage protection trip point in volts (default) or in millivolts.	C
[:LEVel] <NRf>	Sets the overvoltage protection trip point in volts (default) or in millivolts.	C
:TRIPped?	Returns the integer value 1 (TRIPPED) or 0 (UNTRIPPED) state of the overvoltage protection.	C
:STATe?	Returns the integer value 1 (ON) or 0 (OFF) state of the overvoltage protection.	C
:RAMP <NRf> <0+NRf>	Sets the output voltage to ramp from the present value to the specified value (first argument) in the specified time (second argument). See Ramp Function description below.	N
:ABORT	Aborts ramping and clears trigger mode.	
:ALL?	Returns the ramping status of all channels.	
:TRIGgered <NRf> <0+NRf>	Sets the output voltage to ramp from the present value to the specified value (first argument) in the specified time (second argument) upon the trigger command. See description of the Ramp Function below.	N

**\*Note:** For DLM Series firmware versions 1.07 and later, the algebraic sign of the voltage value must match the state of the polarity relay, otherwise an error message will be generated, and the voltage command will be ignored. If the voltage value entered is positive, then the polarity relay must be in the NORM state (query the state of the polarity relay using the OUPP:POL? query command). If the voltage value entered is negative, then the polarity relay must be in the INV state. For DLM Series firmware versions 1.06 and earlier, the algebraic sign caused the polarity relay to change automatically to the matching state. This automatic change in the polarity relay state is no longer supported in firmware versions 1.07 and later. To change the state of the polarity relay, use the OUPP:POL <INV | norm> command (when the isolation relay is open). For DCS Series supplies, the polarity relay still automatically follows the algebraic sign.

## THE RAMP FUNCTION

The ramp function allows the user to transition from one voltage or current to another linearly in a specified time period (100 ms - 99 sec with 100 ms programming resolution). The ramp-upon-trigger function may be used to program different ramping parameters for several units then trigger them all to begin their ramps at the same time. A unit may ramp only voltage or current, not both at a given time. For example, SOUR:VOLT:RAMP:TRIG 1 1 followed by SOUR:CURR:RAMP:TRIG 2 2 will cause the unit to ramp only the output current to 2 amps in 2 seconds upon the TRIG:RAMP command.

Voltage ramping to a higher voltage requires a programmed current of at least 20% of the full scale value. Settings less than 20% will significantly lengthen the ramp time due to charging of the large capacitance in the output section of the power supply.

Voltage ramping to a lower voltage requires an appropriate resistive load. The discharge rate of the large capacitance in the output section of the power supply, plus other user capacitance, significantly lengthens the ramp time.

Current ramping requires an appropriate resistive load.

**Note:** On DCS and PRO-T models, the Reading of Voltage or Current during Ramping will extend Ramp Time.

## SOURce[n]:STATus:BLOCK? "Status Flags" REGISTER

Bit	Hex Value	Name	Description
0	0x001	remote mode	Remote mode was selected.
1	0x002	polarity signal	Negative voltage programmed. Polarity relay signal is on.
2	0x004	PON SRQ	PON Service Request selected by rear panel switch.
3	0x008	SRQ sent	GPIB Service Request issued and the serial poll has not been received.
4	0x010	EEPROM	The EEPROM is locked.
5	0x020	trip occurred	Either an OV or OT trip occurred.
6	0x040	sense signal	Sense relay signal is on.
7	0x080	isolation signal	Isolation relay signal is on.
8	0x100	hold	Voltage and current output waiting for trigger.
9	0x200	fold	Foldback protection is enabled.
10	0x400	SRQ enabled	Service Request generation is enabled.
11	0x800	output	Output is on.

## 3.9 STATUS SCPI Command Subsystem

**Note:** See Section 7 for further information.

### 3.9.1 STATUS SCPI Command Summary

STATUs[n]

```

:OPERation
  :CONDition?
  :ENABle <0+NR1>
  :EVENT?
:PRESet
:PROTection
  :CONDition?
  :ENABle <0+NR1>
  :EVENT?
  :SELEct <0+NR1>
:QUESTionable
  :CONDition?
  :ENABle <0+NR1>
  :EVENT?

```

### 3.9.2 STATUS SCPI Command Reference

Command	Description	SCPI
STATUs[n]	Status subsystem. n = 1-31. The default channel is 1.	C
:OPERation		C
:CONDition?	Returns the integer value of the Operation Condition Register. The query is supported but will always return "0" indicating operational condition.	C
:ENABle <0+NR1>	Sets the enable mask of the Operation Event Register allowing true conditions to be reported in the summary bit of the Operation Condition Register. Values are written and queried but have no effect on the Operation Condition Register.	C
:EVENT?	Returns the integer value of the Operation Event Register. This query is supported but always returns a value of "0" indicating operational condition.	C
:PRESet	Sets the enable mask of the Operation Event Register and the Questionable Event Register to all 1's.	C

Command	Description	SCPI
:PROTection		C
:CONDition?	Returns the integer value of the Protection Condition Register. Used to read the status of the power hardware. See section 3.2.3 for a detailed table of the various bits that make up this register.	C
:ENABle <0+NR1>	Sets the enable mask of the Protection Event Register which allows true conditions to be reported in the summary bit of the Protection Condition Register.	C
:EVENT?	Returns the integer value of the Protection Event Register.	C
:SELEct <0+NR1>	This command provides a means for selecting which fault bits from the protection event register (also called the fault register and readable using the STAT:PROT:EVENT? command) are able to set the protection event flag bit in the SCPI status byte (readable using the *STB? command). It defaults to value 255 at power-on time, and never changes unless intentionally programmed to a new value. Available in the DLM version of the M9 GPIB option with firmware version 1.05 or later. Not available in the DCS.	N
:SELEct?	Queries the last selection value programmed.	N
:QUESTionable		C
:CONDition?	Returns the integer value of the Questionable Condition Register. The query is supported but will always return "0" indicating operational condition.	C
:ENABle <0+NR1>	Sets the enable mask of the Questionable Event Register allowing true conditions to be reported in the summary bit of the Questionable Condition Register. Values are written and queried but have no effect on the Questionable Condition Register.	C
:EVENT?	Returns the integer value of the Questionable Event Register. This query is supported but always returns a value of "0", indicating operational condition.	C

## 3.10 SYSTEM SCPI Command Subsystem

### 3.10.1 SYSTEM SCPI Command Summary

#### SYSTEM

:ERRor?  
 :FAULt?  
 :VERsion?  
 :LOCAL <boolean>  
 :LOCLOUT <boolean>

### 3.10.2 SYSTEM SCPI Command Reference

Command	Description	SCPI
SYSTEM	System subsystem.	C
:ERRor?	Queries Error Queue for next error/event entry (first in, first out). Entries contain an error number and descriptive text. A 0 return value indicates no error occurred; negative numbers are reserved by SCPI. The maximum return string length is 255 characters. The queue holds up to 10 error/entries. All entries are cleared by the *CLS command.	C
:FAULt?	Returns four numeric values separated by commas for the four system fault registers. See System Fault Registers below (e.g., 1, 1, 2, 4 indicates ch 1, 9, 18, and 27 each have at least one fault).  <b>Response:</b> <Fault1–8>, <Fault9–16>, <Fault17–24>, <Fault25–31>	N
:VERsion?	Returns a numeric value corresponding to the SCPI version number for which the instrument complies. The response is in the format YYYY.V where the Y's represent the year and V represents the approved version number for that year (e.g., 1995.0)	C

Command	Description	SCPI
SYST[N]:LOCAL	<p>SYST[N]:LOCAL &lt;ON OFF&gt; is a special purpose programming command that may be used to cause source [N] to be set to the local state or to the remote state. This command has two noteworthy circumstances under which it may prove necessary. The first case involves using RS232 to communicate with the supply, since the normal GPIB mechanisms for transition between local and remote and back again do not exist when using RS232. The other case is when the REMOTE/LOCAL switch S1-1 is placed in the ON position—thereby disabling the GPIB mechanism for transition from remote to local. The SYST[N]:LOCAL ON command provides a means for the system computer to force source [N] to the local state. Also, the SYST[N]:LOCAL? query command is available to examine the local-remote state of the supply. And the command SYST[N]:LOCAL OFF may be used to force supply [N] to the remote state.</p>	N
SYST[n]:LOCAL:LOCKOUT	<p>The SYST[n]:LOCAL:LOCKOUT &lt;0 1 OFF ON&gt; command provides a means of controlling the local lockout functionality that is an alternative to the low level GPIB LLO command. A source number may be specified with this command, thereby allowing selective disabling of the front panel of a selected source.</p> <p>In contrast, using the GPIB LLO low level command causes the master supply, and all attached online M85 slaves, to be placed into the local lockout state. To place the master and all online M85 slaves into the local lockout state using the SYST[n]:LOCAL:LOCKOUT command, specify the source number of 0.</p> <p>To place only one slave into the local lockout state, specify that slave in the command. For example, if the slave is source 3, use the command: SYST3:LOCAL:LOCKOUT 1.</p>	

## SYSTEM FAULT REGISTERS

Bit Position	Bit Weight	Fault1–8	Fault9–16	Fault17–24	Fault25–31
7	128	Channel 8	Channel 16	Channel 24	not used
6	64	Channel 7	Channel 15	Channel 23	Channel 31
5	32	Channel 6	Channel 14	Channel 22	Channel 30
4	16	Channel 5	Channel 13	Channel 21	Channel 29
3	8	Channel 4	Channel 12	Channel 20	Channel 28
2	4	Channel 3	Channel 11	Channel 19	Channel 27
1	2	Channel 2	Channel 10	Channel 18	Channel 26
0	1	Channel 1	Channel 9	Channel 17	Channel 25

The M9 provides four System Fault Registers to allow the user to monitor the fault status of a multiple-channel system. For example, the user may poll these registers to quickly determine which channel generated an enabled GPIB service request.

The SYStem:FAULT? query returns 4 numeric values separated by commas. Each value is the decimal equivalent of the total bit weights for that System Fault Register as described in the table above.

### 3.11 TRIGGER SCPI Command Subsystem

#### 3.11.1 TRIGGER SCPI Command Summary

TRIGger[n]

:ABORt

:RAMP

:TYPE <1|2|3>

#### 3.11.2 TRIGGER SCPI Command Reference

Command	Description	SCPI
TRIGger[n]	Trigger subsystem. N = 0, 1-31. The default channel is 1. A value of n = 0 will program all channels.	C
:ABORt	Clears all settings of voltage and current upon trigger.	N
:RAMP	Implements current or voltage ramping function previously programmed by the SOURce:CURRent:RAMP or SOURce:VOLTage:RAMP commands.	N
:TYPE<1 2 3>	Implements voltage and current values previously programmed by the SOURce:CURRent:LEVel:TRIGger and SOURce:VOLTage:LEVel:TRIGger commands. Valid arguments are 1 (Voltage), 2 (Current), or 3 (Both).	N



### 3.12 Examples of Using the SCPI Commands

The following examples demonstrate programming a power supply to control and to readback the output using the SCPI commands. The maximum voltage and current output is dependent on the particular model. The examples list only the SCPI commands; the code required to send the commands is dependent on the type of language you are using (e.g., C or BASIC) and GPIB hardware (e.g., National Instruments).

**EXAMPLE:** Program a unit with no load at the output to 5 VDC @ 1A, and verify the output.

```
// Use SYST:ERR? after each command to verify no programming errors.
// turn on the unit.
*CLS // clear the unit to its power-on default settings.
*RST // reset the unit.
SOUR:CURREN 1.0 // program output current to 1.0 A.
SOUR:CURREN? // confirm the output current setting (response: 1.0).
SOUR:VOLT 5.0 // program output voltage to 5.0 VDC.
SOUR:VOLT? // confirm the output voltage setting (response: 5.0).
MEAS:CURREN? // measure the actual output current (response: ~ 0.0 with no load on
output).
MEAS:VOLT? // measure the actual output voltage (response: ~ 5.0).
```

**EXAMPLE:** Program a unit with no load at the output to generate a GPIB service request upon an overvoltage protection trip condition. (Must use GPIB not RS-232.)

```
// Use SYST:ERR? after each command to verify no programming errors.
// assure that PON is not selected on the rear panel switch
// and the front panel SRQ led is OFF. Turn on the unit.
*CLS // clear the unit to its power-on default settings.
*RST // reset the unit.
SOUR:VOLT:PROT 4.0 // program the OVP trip point to 4.0 VDC.
SOUR:VOLT:PROT? // confirm the OVP trip point setting (response: 4.0).
SOUR:CURREN 1.0 // program output current to 1.0 A.
SOUR:VOLT 3.0 // program output voltage to 3.0 VDC.
STAT:PROT:ENABLE 8 // program the unit to report OVP trip.
STAT:PROT:ENABLE? // confirm that OVP fault is enabled (response: 8).
*SRE 2 // enable the GPIB service request upon a fault.
*SRE? // confirm the GPIB service request enabled (response 2).
STAT:PROT:EVENT? // confirm no faults occurred (response: 0).
// confirm that the OVP led and SRQ led is not active.
SOUR:VOLT 7.0 // program output voltage to 7.0 VDC - cause OVP trip!
// confirm that OVP led and SRQ led is active, and unit issued a GPIB service request (use a
serial poll).
```

**EXAMPLE:** Program a unit with no load at the output to change its output voltage and current to 5 VDC @ 1A at the same time.

```
// Use SYST:ERR? after each command to verify no programming errors.
// turn on the unit.
*CLS // clear the unit to its power-on default settings.
*RST // reset the unit.
SOUR:CURREN:TRIG1.0 // program output current to 1.0 A upon trigger.
SOUR:CURREN:TRIG? // confirm output current set to 1.0 A upon trigger.
SOUR:VOLT:TRIG5.0 // program output voltage to 5.0 VDC upon trigger
SOUR:VOLT:TRIG? // confirm output current set to 5.0 VDC upon trigger.
MEAS:CURREN? // measure the actual output current (response: 0.0).
MEAS:VOLT? // measure the actual output voltage (response: 0.0).
TRIG:TYPE 3 // trigger the unit to implement curr and volt programming.
MEAS:CURREN? // measure the actual output current (response: ~ 0.0 with no load
on output).
MEAS:VOLT? // measure the actual output voltage (response: ~ 5.0).
TRIG:ABORT // turn off trigger mode.
```

**EXAMPLE:** Program a unit with no load at the output to ramp its output voltage from 5 VDC to 25 VDC in 30 seconds.

**Note:** The maximum output voltage is dependent upon the power supply rating.

```
// Use SYST:ERR? after each command to verify no programming errors.
// turn on the unit.
*CLS // clear the unit to its power-on default settings.
*RST // reset the unit.
SOUR:CURREN 33.0 // program output current to 33.0 A.
SOUR:VOLT 5.0 // program output voltage to 5.0 VDC.
SOUR:VOLT:RAMP 25.0 30.0 // program voltage to ramp from the present
// value (5.0 VDC) to 25.0 VDC in 30 seconds.
```

**EXAMPLE:** Program a unit with the output shorted to ramp its output current from 5A to 25A in 30 seconds.

```
// Use SYST:ERR? after each command to verify no programming errors.
// turn on with no load at the output.
*CLS // clear the unit to its power-on default settings.
*RST // reset the unit.
// short the output.
SOUR:VOLT 33.0 // program output voltage to 33.0 VDC.
SOUR:CURREN 5.0 // program output current to 5.0 A.
SOUR:CURREN:RAMP 25.0 30.0 // program current to ramp from the present
// value (5.0 A) to 25.0 A in 30 seconds.
```

**EXAMPLE:** Program a unit with no load at the output to ramp its output voltage from 5 VDC to 25 VDC in 30 seconds upon the trigger command.

```
// Use SYST:ERR? after each command to verify no programming errors.
// turn on the unit.
*CLS // clear the unit to its power-on default settings.
*RST // reset the unit.
SOUR:CURR 33.0 // program output current to 33.0 A.
SOUR:VOLT 5.0 // program output voltage to 5.0 VDC.
SOUR:VOLT:RAMP:TRIG 25.0 30.0 // program voltage to ramp from the present
// value (5.0 VDC) to 25.0 VDC in 30 secs.
// upon the trigger command.
TRIG:RAMP // start ramp execution.
TRIG:ABORT // turn off trigger mode.
```

**EXAMPLE:** Program a unit to power-on and initialize to 2 VDC @ 1A with an overvoltage protection level of 3 VDC. Verify proper power-on initialization.

```
// Use SYST:ERR? after each command to verify no programming errors.
// turn on the unit.
*CLS // clear the unit to its power on default settings.
*RST // reset the unit.
CAL:INIT:CURR 1.0 // set power-on initial current to 1.0 A.
CAL:INIT:CURR? // confirm power-on initial current setting.
CAL:INIT:VOLT 2.0 // set power-on initial voltage to 2.0 VDC.
CAL:INIT:VOLT? // confirm power-on initial voltage setting.
CAL:INIT:VOLT:PROT 3.0 // set power-on initial overvoltage protection to 3.0 VDC.
CAL:INIT:VOLT:PROT? // confirm power-on initial overvoltage protection setting.
CAL:UNLOCK "6867" // unlock nonvolatile memory for calibration value storage.
CAL:STORE // store the calibration values in nonvolatile memory.
CAL:LOCK // lock nonvolatile memory for calibration value protection.
// cycle power to the unit.
// note voltage is initialized to 2.0 VDC via front panel.
SOUR:CURR? // confirm power-on initial current setting.
SOUR:VOLT? // confirm power-on initial voltage setting.
SOUR:VOLT:PROT? // confirm power-on initial overvoltage protection setting.
```

This page intentionally left blank.

# SECTION 4 CALIBRATION



## WARNING

Please refer to the power supply manual for further information before performing calibration procedures. Calibration must be performed by qualified personnel who appropriately deal with attendant hazards. If calibration is not performed properly, functional problems could arise, requiring that the supply be returned to the factory.

## 4.1 Introduction

The M9 is calibrated to adjust internal signal levels to correspond to the expected supply output signal levels. You must perform the calibration procedures if the power supply's programming or readback performance falls out of specification due to component aging drifts. Refer to your power supply manual to find the required calibration interval. The M9 is calibrated for output voltage programming, output current programming, output overvoltage protection programming, voltage readback, and current readback. There are 10 calibration factors (four measurement and six output).

The calibration procedures in the following sections are designed to be performed at ambient temperature of  $25^{\circ}\text{C} \pm 5^{\circ}\text{C}$ , after the unit has had a stable output and a stable load for at least 30 minutes.

The following test equipment is required in addition to the computer system to complete the following calibration:

- 6-digit digital voltmeter (DVM)
- current shunt rated for 110% of full output current

## 4.2 Setup for Calibration

### **STEP DESCRIPTION**

1. Disconnect the power supply's AC input power.
2. Disconnect the load from the power supply you want to calibrate.
3. Connect the power supply for sensing at the required load point. Refer to the power supply manual for further information.
4. Connect the DVM to the output for voltage or overvoltage calibration, or connect a current shunt rated for the full output current of the supply and the DVM for current calibration.
5. Assure the correct GPIB primary address has been set by the rear panel switch.
6. Set the power supply to REMOTE mode by the rear panel switch.
7. Connect the GPIB controller to the power supply at the rear panel connector.
8. Reconnect the AC input power. Turn the unit ON and allow the unit to warm up for at least 30 minutes.
9. The unit is ready for all calibration procedures.



#### **WARNING**

Exercise caution when using and servicing power supplies. High energy levels can be stored at the output voltage terminals on all power supplies in normal operation. In addition, potentially lethal voltages exist in the power circuit and the output connector on power supplies that are rated at 60V and over. Filter capacitors store potentially dangerous energy for some time after power is removed.

## 4.3 Voltage Programming Calibration

Two-point voltage programming calibration (see Section 4.3.1) is used in all firmware; however, alternatively, there is an optional five-point voltage programming calibration (see Section 4.3.2) available only for DLM600 M9G, DLM600 M85, and M83 Ethernet if the following requirements are satisfied:

- Version 3.10 for the digital controller side firmware (DLM600 M9G).
- Version 1.14 for the analog controller side firmware (DLM600 M85).
- Version 1.0 for the M83 Ethernet.

### 4.3.1 Two-Point Voltage Programming Calibration

To query whether two-point or five-point calibration is active, issue the following command:

```
CAL[n]:MOD:USETWOPOINT?
```

A return value of 1 means two-point is active; a return value of 0 means five-point is active.

If two-point cal is to be used (i.e., to turn off five-point cal), issue the following commands:

```
CAL[n]:MOD:USETWOPOINT ON  
CAL[n]:UNLOCK "6867"  
CAL[n]:STORE  
CAL[n]:LOCK
```

#### **STEP DESCRIPTION**

1. Program the overvoltage protection to maximum to prevent nuisance trips:  

```
CAL[n]:OUTP:VOLT:PROT:DAC 4095
```
2. Program the output current to full scale to prevent Constant-Current operation:  

```
CAL[n]:OUTP:CURR:DAC 4095
```
3. Program the output of the first calibration point to approximately 15% of full scale voltage by sending the following command string from the computer:  

```
CAL[n]:OUTP:VOLT:DAC 600
```
4. Let the output settle and measure the voltage with the meter.
5. Enter the actual output voltage corresponding to the DAC value 600 of the first calibration point:  

```
CAL[n]:OUTP:VOLT:POINT 1 <voltage>
```
6. Program the output of the second calibration point to approximately 85% of full scale voltage by sending the following command string from the computer:  

```
CAL[n]:OUTP:VOLT:DAC 3400
```
7. Let the output settle and measure the voltage with the meter.
8. Enter the actual output voltage corresponding to the DAC value 3400 of the second calibration point:  

```
CAL[n]:OUTP:VOLT:POINT 2 <voltage>
```
9. Reset the output voltage to 0 volts.  

```
CAL[n]:OUTP:VOLT:DAC 0
```

10. Program the M9 to calculate the output voltage calibration gain and offset values:

```
CAL[n]:OUTP:VOLT:CALC
```

11. Program the M9 to unlock the non-volatile memory for calibration value storage:

```
CAL[n]:UNLOCK "6867"
```

12. Program the M9 to store the calibration values in non-volatile memory:

```
CAL[n]:STORE
```

13. Program the M9 to lock the non-volatile memory for calibration value protection

```
CAL[n]:LOCK
```

14. The output voltage calibration is complete. The unit may be turned OFF or other calibration procedures may be performed.

#### 4.3.2 Five-Point Voltage Programming Calibration

As noted in the introduction of Section 4.3, the following new commands have been now implemented in the DLM600 M9G, DLM600 M85, and M83 Ethernet firmware, set to support an optional five-point calibration of voltage output, provided the following requirements have been satisfied:

Version 3.10 for the digital controller side firmware (DLM600 M9G).

Version 1.14 for the analog controller side firmware (DLM600 M85).

Version 1.0 for the M83 Ethernet.

To query whether two-point or five-point calibration is active, issue the following command:

```
CAL[n]:MOD:USETWOPOINT?
```

A return value of 1 means two-point is active; a return value of 0 means five-point is active.

To ensure that the new five-point data is used, and not the older two-point data, issue the following commands:

```
CAL[n]:MOD:USETWOPOINT OFF
```

```
CAL[n]:UNLOCK "6867"
```

```
CAL[n]:STORE
```

```
CAL[n]:LOCK
```

#### **STEP DESCRIPTION**

1. Attach to the open (unshunted) power supply outputs a high precision volt meter.

```
*CLS[n]
```

```
*RST[n]
```

```
CAL[n]:OUTP:VOLT:PROT:DAC 4095
```

```
CAL[n]:OUTP:CURR:DAC 4095
```



2. Program the first value:

```
CAL[n]:OUTP:VOLT:DAC 600
Measure the output voltage with a high precision volt meter; this is value 1. Enter
it using the following command:
CAL[n]:OUTP:VOLT:FIVEPOINT 1 <value 1>
```

3. Program the second value:

```
CAL[n]:OUTP:VOLT:DAC 1300
Measure the output voltage with a high precision volt meter; this is value 2. Enter
it using the following command:
CAL[n]:OUTP:VOLT:FIVEPOINT 2 <value 2>
```

4. Program the third value:

```
CAL[n]:OUTP:VOLT:DAC 2000
Measure the output voltage with a high precision volt meter; this is value 3. Enter
it using the following command:
CAL[n]:OUTP:VOLT:FIVEPOINT 3 <value 3>
```

5. Program the fourth value:

```
CAL[n]:OUTP:VOLT:DAC 2700
Measure the output voltage with a high precision volt meter; this is value 4. Enter
it using the following command:
CAL[n]:OUTP:VOLT:FIVEPOINT 4 <value 4>
```

6. Program the fifth value:

```
CAL[n]:OUTP:VOLT:DAC 3400
Measure the output voltage with a high precision volt meter; this is value 5. Enter
it using the following command:
CAL[n]:OUTP:VOLT:FIVEPOINT 5 <value 5>
```

7. To review entered data for five point voltage calibration, issue the following query:

```
CAL[n]:OUTP:VOLT:FIVEPOINT?
```

8. Save the results:

```
CAL[n]:UNLOCK "6867"
CAL[n]:STORE
CAL[n]:LOCK
```

## 4.4 Voltage Measurement/Readback Calibration

### **STEP DESCRIPTION**

1. Program the overvoltage protection to maximum to prevent nuisance trips:

```
CAL[n]:OUTP:VOLT:PROT:DAC 4095
```

2. Program the output current to full scale to prevent Constant-Current operation:

```
CAL[n]:OUTP:CURR:DAC 4095
```

3. Program the output of the first calibration point to approximately 15% of full scale voltage by sending the following command string from the computer:

```
CAL[n]:OUTP:VOLT:DAC 600
```

4. Let the output settle and measure the voltage with the meter.
5. Enter the actual voltage readback corresponding to the DAC value 600 of the first calibration point:

```
CAL[n]:MEAS:VOLT:POINT 1 <voltage>
```

6. Program the output of the second calibration point to approximately 85% of full scale voltage by sending the following command string from the computer:

```
CAL[n]:OUTP:VOLT:DAC 3400
```

7. Let the output settle and measure the voltage with the meter.
8. Enter the actual voltage readback corresponding to the DAC value 3400 of the second calibration point:

```
CAL[n]:MEAS:VOLT:POINT 2 <voltage>
```

9. Reset the output voltage to 0 volts.

```
CAL[n]:OUTP:VOLT:DAC 0
```

10. Program the M9 to calculate the voltage readback calibration gain and offset values:

```
CAL[n]:MEAS:VOLT:CALC
```

11. Program the M9 to unlock the non-volatile memory for calibration value storage:

```
CAL[n]:UNLOCK "6867"
```

12. Program the M9 to store the calibration values in non-volatile memory:

```
CAL[n]:STORE
```

13. Program the M9 to lock the non-volatile memory for calibration value protection

```
CAL[n]:LOCK
```

14. The voltage readback calibration is complete. The unit may be turned OFF or other calibration procedures may be performed.

## 4.5 Overvoltage Protection Programming Calibration

The overvoltage protection calibration procedure requires calibrated output voltage programming and voltage readback.

### **STEP DESCRIPTION**

1. Program the M9 to self-calibrate the overvoltage protection:

CAL[n]:OUTP:VOLT:PROT:CALC

The overvoltage protection calibration function requires over 30 seconds. Use \*ESE 1 and serial polls to detect when calibration is done.

2. Program the M9 to unlock the non-volatile memory for calibration value storage:

CAL[n]:UNLOCK "6867"

3. Program the M9 to store the calibration values in non-volatile memory:

CAL[n]:STORE

4. Program the M9 to lock the non-volatile memory for calibration value protection

CAL[n]:LOCK

5. The output overvoltage protection calibration is complete. The unit may be turned OFF or other calibration procedures may be performed.

## 4.6 Current Programming Calibration

Two-point current programming calibration (see Section 4.6.1) is used in all firmware; however, alternatively, there is an optional five-point current programming calibration (see Section 4.6.2) available only for DLM600 M9G, DLM600 M85, and M83 Ethernet if the following requirements are satisfied:

Version 3.10 for the digital controller side firmware (DLM600 M9G).  
Version 1.14 for the analog controller side firmware (DLM600 M85).  
Version 1.0 for the M83 Ethernet.

### 4.6.1 Two-Point Current Programming Calibration

To query whether two-point or five-point calibration is active, issue the following command:

CAL[n]:MOD:USETWOPOINT?

A return value of 1 means two-point is active; a return value of 0 means five-point is active.

If two-point cal is to be used (i.e., to turn off five-point cal), issue the following commands:

```
CAL[n]:MOD:USETWOPOINT ON  
CAL[n]:UNLOCK "6867"  
CAL[n]:STORE  
CAL[n]:LOCK
```

### **STEP DESCRIPTION**

1. Program the overvoltage protection to maximum to prevent nuisance trips:  
CAL[n]:OUTP:VOLT:PROT:DAC 4095
2. Program the output voltage to full scale to prevent Constant-Voltage operation:  
CAL[n]:OUTP:VOLT:DAC 4095
3. Program the output of the first calibration point to approximately 15% of full scale current by sending the following command string from the computer:  
CAL[n]:OUTP:CURR:DAC 600
4. Let the output settle and measure the current with the current shunt and the meter.
5. Enter the actual output current corresponding to the DAC value 600 of the first calibration point:  
CAL[n]:OUTP:CURR:POINT 1 <current>
6. Program the output of the second calibration point to approximately 85% of full scale current by sending the following command string from the computer:  
CAL[n]:OUTP:CURR:DAC 3400
7. Let the output settle and measure the current with the current shunt and the meter.
8. Enter the actual output current corresponding to the DAC value 3400 of the second calibration point:  
CAL[n]:OUTP:CURR:POINT 2 <current>
9. Reset the output current to 0 amps.  
CAL[n]:OUTP:CURR:DAC 0
10. Program the M9 to calculate the output current calibration gain and offset values:  
CAL[n]:OUTP:CURR:CALC
11. Program the M9 to unlock the non-volatile memory for calibration value storage:  
CAL[n]:UNLOCK "6867"

12. Program the M9 to store the calibration values in non-volatile memory:

```
CAL[n]:STORE
```

13. Program the M9 to lock the non-volatile memory for calibration value protection

```
CAL[n]:LOCK
```

14. The output current calibration is complete. The unit may be turned OFF or other calibration procedures may be performed.

#### 4.6.2 Five-Point Calibration Procedure for Current Output

As noted in the introduction of Section 4.6, the following new commands have been now implemented in the DLM600 M9G, DLM600 M85, and M83 Ethernet firmware, set to support an optional five-point calibration of current output, provided the following requirements have been satisfied:

Version 3.10 for the digital controller side firmware (DLM600 M9G).  
Version 1.14 for the analog controller side firmware (DLM600 M85).  
Version 1.0 for the M83 Ethernet.

To query whether two-point or five-point calibration is active, issue the following command:

```
CAL[n]:MOD:USETWOPOINT?
```

A return value of 1 means two-point is active; a return value of 0 means five-point is active.

To ensure that the new five-point data is used, and not the older two-point data, issue the following commands:

```
CAL[n]:MOD:USETWOPOINT OFF
CAL[n]:UNLOCK "6867"
CAL[n]:STORE
CAL[n]:LOCK
```

#### **STEP DESCRIPTION**

1. Attach a high precision shunt between the power supply's output terminals and attach a high precision volt meter across the shunt.

```
*CLS[n]
*RST[n]
CAL[n]:OUTP:VOLT:PROT:DAC 4095
CAL[n]:OUTP:VOLT:DAC 4095
```

2. Program the first value:

```
CAL[n]:OUTP:CURR:DAC 600
Measure the output current to a high precision; this is value 1. Enter it using the
following command:
CAL[n]:OUTP:CURR:FIVEPOINT 1 <value 1>
```

3. Program the second value:

```
CAL[n]:OUTP:CURR:DAC 1300
Measure the output current to a high precision; this is value 2. Enter it using the
following command:
CAL[n]:OUTP:CURR:FIVEPOINT 2 <value 2>
```

4. Program the third value:

```
CAL[n]:OUTP:CURR:DAC 2000
Measure the output current to a high precision; this is value 3. Enter it using the
following command:
CAL[n]:OUTP:CURR:FIVEPOINT 3 <value 3>
```

5. Program the fourth value:

```
CAL[n]:OUTP:CURR:DAC 2700
Measure the output current to a high precision; this is value 4. Enter it using the
following command:
CAL[n]:OUTP:CURR:FIVEPOINT 4 <value 4>
```

6. Program the fifth value:

```
CAL[n]:OUTP:CURR:DAC 3400
Measure the output current to a high precision; this is value 5. Enter it using the
following command:
CAL[n]:OUTP:CURR:FIVEPOINT 5 <value 5>
```

7. To review entered data for five point current calibration, issue the following query:

```
CAL[n]:OUTP:CURR:FIVEPOINT?
```

8. Save the results:

```
CAL[n]:UNLOCK "6867"
CAL[n]:STORE
CAL[n]:LOCK
```

## 4.7 Current Measurement/Readback Calibration

### **STEP DESCRIPTION**

1. Program the overvoltage protection to maximum to prevent nuisance trips:

```
CAL[n]:OUTP:VOLT:PROT:DAC 4095
```

2. Program the output voltage to full scale to prevent Constant-Voltage operation:

```
CAL[n]:OUTP:VOLT:DAC 4095
```

3. Program the output of the first calibration point to approximately 15% of full scale current by sending the following command string from the computer:

```
CAL[n]:OUTP:CURR:DAC 600
```

4. Let the output settle and measure the current with the current shunt and the meter.

5. Enter the actual current readback corresponding to the DAC value 600 of the first calibration point:

```
CAL[n]:MEAS:CURR:POINT 1 <current>
```

6. Program the output of the second calibration point to approximately 85% of full scale current by sending the following command string from the computer:

```
CAL[n]:OUTP:CURR:DAC 3400
```

7. Let the output settle and measure the current with the current shunt and the meter.

8. Enter the actual current readback corresponding to the DAC value 3400 of the second calibration point:

```
CAL[n]:MEAS:CURR:POINT 2 <current>
```

9. Reset the output current to 0 amps.

```
CAL:OUTP:CURR:DAC 0
```

10. Program the M9 to calculate the current readback calibration gain and offset values:

```
CAL[n]:MEAS:CURR:CALC
```

11. Program the M9 to unlock the non-volatile memory for calibration value storage:

```
CAL[n]:UNLOCK "6867"
```

12. Program the M9 to store the calibration values in non-volatile memory:

```
CAL[n]:STORE
```

13. Program the M9 to lock the non-volatile memory for calibration value protection

```
CAL[n]:LOCK
```

14. The current readback calibration is complete. The unit may be turned OFF or other calibration procedures may be performed.

## 4.8 Voltage Programming and Readback Calibration

This procedure may be used to save time if both the output and readback require calibration.

### **STEP DESCRIPTION**

1. Program the overvoltage protection to maximum to prevent nuisance trips:

```
CAL[n]:OUTP:VOLT:PROT:DAC 4095
```

2. Program the output current to full scale to prevent Constant-Current operation:  
CAL[n]:OUTP:CURR:DAC 4095
3. Program the output of the first calibration point to approximately 15% of full scale voltage by sending the following command string from the computer:  
CAL[n]:OUTP:VOLT:DAC 600
4. Let the output settle and measure the voltage with the meter.
5. Enter the actual output voltage corresponding to the DAC value 600 of the first calibration point:  
CAL[n]:OUTP:VOLT:POINT 1 <voltage>  
CAL[n]:MEAS:VOLT:POINT 1 <voltage>
6. Program the output of the second calibration point to approximately 85% of full scale voltage by sending the following command string from the computer:  
CAL[n]:OUTP:VOLT:DAC 3400
7. Let the output settle and measure the voltage with the meter.
8. Enter the actual output voltage corresponding to the DAC value 3400 of the second calibration point:  
CAL[n]:OUTP:VOLT:POINT 2 <voltage>  
CAL[n]:MEAS:VOLT:POINT 2 <voltage>
9. Reset the output voltage to 0 volts.  
CAL[n]:OUTP:VOLT:DAC 0
10. Program the M9 to calculate the output voltage and the voltage measurement calibration gain and offset values:  
CAL[n]:OUTP:VOLT:CALC  
CAL[n]:MEAS:VOLT:CALC
11. Program the M9 to unlock the non-volatile memory for calibration value storage:  
CAL[n]:UNLOCK "6867"
12. Program the M9 to store the calibration values in non-volatile memory:  
CAL[n]:STORE
13. Program the M9 to lock the non-volatile memory for calibration value protection.  
CAL[n]:LOCK
14. The output voltage and voltage measurement calibrations are completed at the same time. The unit may be turned OFF or other calibration procedures may be performed.



## 4.9 Current Programming and Readback Calibration

This procedure may be used to save time if both the output and readback require calibration.

### **STEP DESCRIPTION**

1. Program the overvoltage protection to maximum to prevent nuisance trips:  
CAL[n]:OUTP:VOLT:PROT:DAC 4095
2. Program the output voltage to full scale to prevent Constant-Voltage operation:  
CAL[n]:OUTP:VOLT:DAC 4095
3. Program the output of the first calibration point to approximately 15% of full scale current by sending the following command string from the computer:  
CAL[n]:OUTP:CURR:DAC 600
4. Let the output settle and measure the current with the current shunt and the meter.
5. Enter the actual output current corresponding to the DAC value 600 of the first calibration point:  
CAL[n]:OUTP:CURR:POINT 1 <current>  
CAL[n]:MEAS:CURR:POINT 1 <current>
6. Program the output of the second calibration point to approximately 85% of full scale current by sending the following command string from the computer:  
CAL[n]:OUTP:CURR:DAC 3400
7. Let the output settle and measure the current with the current shunt and the meter.
8. Enter the actual output current corresponding to the DAC value 3400 of the second calibration point:  
CAL[n]:OUTP:CURR:POINT 2 <current>  
CAL[n]:MEAS:CURR:POINT 2 <current>
9. Reset the output current to 0 amps.  
CAL[n]:OUTP:CURR:DAC 0
10. Program the M9 to calculate the output current and current measurement calibration gain and offset values:  
CAL[n]:OUTP:CURR:CALC  
CAL[n]:MEAS:CURR:CALC
11. Program the M9 to unlock the non-volatile memory for calibration value storage:  
CAL[n]:UNLOCK "6867"

12. Program the M9 to store the calibration values in non-volatile memory:

CAL[n]:STORE

13. Program the M9 to lock the non-volatile memory for calibration value protection

CAL[n]:LOCK

14. The output current and current measurement calibrations are completed at the same time. The unit may be turned OFF or other calibration procedures may be performed.

# SECTION 5

## M9A COMMAND OPERATION

### 5.1 Introduction

The following sections describe the operation of the M9 by remote programming using the M9A command set. This command set is included for backward compatibility with older products and is not recommended for new programs. See Section 3 for further information.

The M9A command set of programming, query, and status commands allow control of the power supply. Both GPIB and RS-232 are supported with the exception of Service Request (SRQ), which is a GPIB-specific implementation.

### 5.2 M9A GPIB Register Definitions

#### 5.2.1 Status Byte and the Serial Poll Operation

The supply responds to a GPIB serial poll with the 8-bit Status Byte defined as follows:

#### STATUS BYTE REGISTER

Bit Position	Decimal Weight	Description	Reset By
7 (MSB)	128	Power ON (PON) - Set when the supply initializes at power on.	CLR or Device clear.
6	64	Request Service (SRQ) - Set when the supply requests service.	Serial poll.
5	32	Error (ERR) - Set when the ERR bit is asserted in the Status Register.	ERR? query.
4	16	Ready - Set when the supply is ready to accept commands.	Power supply during command processing period.
3, 2, 1	8, 4, 2	Not used.	—
0 (LSB)	1	Fault - Set when any bit in the fault register is set by a fault condition in the supply.	FAULT? query.

### 5.2.2 Status, Accumulated Status, Fault, and Mask Registers

The Status, Accumulated Status, Fault, and Mask Registers share the same bit map. Each register describes an operating condition of the supply. The Status Register describes the current status of the supply. The Accumulated Status Register records the history of the supply status. The Fault Register records the condition that occurred enabled by the Mask Register.

#### STATUS AND FAULT REGISTERS

Condition	Mnemonic	Bit Position	Bit Weight
Remote programming error	ERR	7	128
Foldback mode operation	FOLD	6	64
Supply external shutdown active (J7-1)	SD	5	32
Overtemperature protection tripped	OT	4	16
Overvoltage protection tripped	OV	3	8
Not used	—	2	4
Constant current operation	CC	1	2
Constant voltage operation	CV	0	1

#### Notes:

1. Only CV,CC,OV,OT,SD,ERR, and FOLD can be MASKed or UNMASKed.
2. The error (ERR) bit is reset with an error query (ERR?).
3. The accumulated status register is cleared with an accumulated status query (ASTS?).
4. A fault is cleared with a fault query (FAULT?).

## 5.2.3 Error Register

## ERROR REGISTER

<b>Error #</b>	<b>Error Identification</b>	<b>Explanation</b>
0	No errors	—
1	Unrecognized character	A character such as @ or \$ was received.
2	Not used	—
3	Not used	—
4	Syntax error	An unrecognized command or data type was encountered. (e.g., AAA)
5	Number out of range	A number received for the command was outside of the allowed range. (e.g., VSET 900)
6	Attempt to exceed soft limits	An attempt was made to program a voltage or current greater than the soft limit. (e.g., VMAX 500V; VSET 550V)
7	Improper soft limit	An attempt was made to program a soft limit less than the output value. (e.g., VSET 5; VMAX 3)
8	Not used	—
9	OVP set below output	An OVSET command was sent with a trip value lower than the output voltage. (e.g., VSET 5; OVSET 3)

## 5.3 M9A Command Subsystem Reference



### CAUTION

Do not intermingle M9A commands with SCPI commands. Doing so will result in unreliable, unmeaningful, and/or incorrect status commands. Use *only* M9A commands, or *only* SCPI commands; do not use both together.

The M9A commands provide control in 4 categories:

- ID control
- initialization control
- output control
- status and fault control

### ID CONTROL

Commands	Description
ID?	Returns the power supply model and the firmware version as an ASCII string.  <b>Response:</b> ID <model name> <version> <b>Example:</b> ID DCS33-33 Version 1.00

The ID? query is a safe command to initially verify communications since it does not affect the unit in the other categories.

### INITIALIZATION CONTROL

Commands	Description
CLR	Initializes the supply to its Power ON (PON) condition. Resets the PON bit (position 7) in the Serial Poll Register.
RST	Resets the supply to present voltage and current settings if the output is disabled by overvoltage or foldback protection. Output values may be changed via VSET, ISET, and OVSET while the unit is disabled but will not take effect until RST is used.

Use the Initialization Control commands to clear an over-voltage fault condition or to set the unit to the known safe power-on default settings state.

### OUTPUT CONTROL

Commands	Description
HOLD <boolean>	Enables or disables voltage and current setting hold mode. When HOLD ON is specified, hold mode is enabled so that all voltage and current settings which normally be implemented by the supply are held until a trigger (TRG) command is received. This allows you to synchronize the operation of several supplies.  <b>Initial value:</b> HOLD 0 (OFF)

Commands	Description
HOLD?	Returns the integer value of the present hold mode setting.  <b>Response:</b> HOLD <mode> where mode is: 0 (OFF or disabled) or 1 (ON or enabled)
IOUT?	Returns the floating point value of the actual current output measured using the built-in current readback A/D converter.  <b>Response:</b> IOUT <0+NRf>
IMAX <0+NRf>	Sets the upper soft limit on the programmed output current. If the soft limit is exceeded or if the soft limit value is lower than the present output current setting, the supply will ignore the command, turn on the ERR LED, and set the ERR bit in the Status and Serial Poll Registers.  <b>Range:</b> 0 to model maximum output current <b>Initial value:</b> model maximum current
IMAX?	Returns the floating point value of the soft current limit setting. <b>Response:</b> IMAX <0+NRf>
ISET <0+NRf>	Sets the output current in amps (default) or in milliamps. This programmed current is the actual output in CC mode or the current limit in Constant-Voltage mode.  <b>Range:</b> 0 to model maximum output current (IMAX) <b>Initial value:</b> 0 amps
ISET?	Returns the floating point value of the present output current setting. Does not apply to current settings which are being held (see HOLD).  <b>Response:</b> ISET <0+NRf>
OUT <boolean>	Enables or disables voltage and current output. The supply will continue to accept new commands while the output is disabled but these will not be implemented until OUT ON or OUT 1 is received. OUT OFF or OUT 0 also sets the isolation signal on the rear panel J7 connector line 3 which may be used to trip external relays to isolate the supply from the load.  <b>Initial value:</b> OUT 1 (ON or output enabled)
OUT?	Returns the integer value of the present enabled/disabled status of the output voltage and current.  <b>Response:</b> OUT <mode> where mode is: 0 (OFF or disabled) or 1 (ON or enabled)
TRG	Implements the programmed voltage and current settings which had been in hold mode. The supply operates with previous values until the trigger (TRG) command is received.

Commands	Description
VMAX <NRf>	<p>Sets the upper soft limit on the programmed output voltage. If the soft limit is exceeded or if the soft limit value is lower than the present output voltage setting, the supply will ignore the command, turn on the ERR LED, and set the ERR bit in the Status and Serial Poll Registers.</p> <p><b>Range:</b> 0 to model maximum output voltage  <b>Initial value:</b> model maximum voltage</p>
VMAX?	<p>Returns the floating point value of the soft voltage limit setting.</p> <p><b>Response:</b> VMAX &lt;NRf&gt;</p>
VOUT?	<p>Returns the floating point value of the actual voltage output measured using the built-in voltage readback A/D converter.</p> <p><b>Response:</b> VOUT &lt;NRf&gt;</p>
VSET <NRf>	<p>Sets the output voltage in volts (default) or in millivolts. This programmed voltage is the actual output in CV mode or the voltage limit in Constant-Current mode.</p> <p><b>Range:</b> 0 to model maximum output voltage (VMAX)  <b>Initial value:</b> 0 volts</p>
VSET?	<p>Returns the floating point value of the present output voltage setting. Does not apply to voltage settings which are being held (see HOLD command).</p> <p><b>Response:</b> VSET &lt;NRf&gt;</p>

Use the Output Control commands to set the output conditions of the unit:

- the output may be enabled or disabled,
- the output current and voltage levels may be set and queried,
- the output current and voltage may be measured,
- the output current and voltage change may be synchronized, and
- the output current and voltage level limits may be set and queried.

#### STATUS AND FAULT CONTROL

Command	Description
ASTS?	<p>Returns the integer value of the Accumulated Status Register. The Accumulated Status Register stores any bit that was entered in the Status Register since the Accumulated Status query command (ASTS?) was last used (if at all) regardless of whether the condition still exists. The Accumulated Status Register has the same bits, weights, and conditions as the Status Register. A bit in the Accumulated Status Register will be set to 1 if the corresponding bit in the Status Register has been 1 (TRUE) at any time since the register was last read.</p> <p><b>Response:</b> ASTS &lt;status mask&gt; where status mask is the decimal equivalent of the total bit weights for the operating conditions as listed in the Status Register.</p>



Command	Description
DLY <0+NRf>	<p>Sets the programmable time delay before reporting fault conditions after a new output voltage or current is specified (VSET, ISET) or RST, TRG, or OUT ON commands are received. During delay time, the power supply disabled CV, CC, and FOLD conditions from generating faults, preventing possible nuisance foldback or service requests if the supply momentarily switches modes while changing an output setting.</p> <p><b>Range:</b> 0 to 32 seconds, with 1 ms resolution.  <b>Initial value:</b> 0.5 seconds</p>
DLY?	<p>Returns the floating point value of the programmable delay time setting before reporting fault conditions.</p> <p><b>Response:</b> DLY &lt;0+NRf&gt;</p>
ERR?	<p>Returns the integer value of the most recent remote programming errors which occurred since the last time the error query (ERR?) was issued. When the supply detects a programming error, it sets the ERR bit in the Status and Fault Registers which can be unmasked (UNMASKed) on the Fault Register to request service. The remaining portion of the command line is discarded. An error query or a new, error-free, command clears the ERR bit in both the Status Register and the Serial Poll Register.</p> <p><b>Response:</b> ERR &lt;0+NRf&gt;</p>
FAULT?	<p>Returns the integer value of the Fault Register for the fault condition status. When a bit is set in the Fault Register, the rear panel J7 connector Fault Line 4 is also asserted. The Fault Line from any supply in the system may be tied to the External Shutdown Line J7-1 of all supplies to provide shutdown of the system, independent of the GPIB, for user-defined faults. The FAULT? query clears bits in the supply's Fault Register and fault lines.</p> <p><b>Response:</b> FAULT &lt;fault mask&gt; where fault mask is the decimal equivalent of the total bit weights for the operating conditions as listed in the Fault Register.</p>
FOLD <OFF CV CC> or FOLD <0 1 2>	<p>Sets the foldback mode. Foldback protection disables the supply output when the output enters the foldback condition. Reset with the RST command. For example, specify FOLD CV or FOLD 1 for the supply to operate in Constant Current mode and have foldback protection disable the output if the supply switches to Constant Voltage mode.</p> <p><b>Initial value:</b> FOLD 0 (OFF).</p>
FOLD?	<p>Returns the integer value of the present foldback mode setting.</p> <p><b>Response:</b> FOLD &lt;mode&gt; where mode is:  0 (OFF), or  1 (CV or Constant Voltage mode), or  2 (CC or Constant Current mode)</p>

Command	Description
MASK <mnemonics>	<p>Disables the specified supply's operating conditions from reporting fault conditions in the Fault and Status Registers. Mnemonics are separated from each other by commas, and may be sent in any order.</p> <p><b>MASK mnemonics:</b> CV (Constant Voltage),            CC (Constant Current)            OV (Overvoltage Protection)            OT (Over Temperature Protection)            SD (Shutdown)            FOLD (Foldback)            NONE (all conditions enabled)            ALL (all conditions disabled)</p> <p><b>Initial value:</b> MASK ALL</p>
OVSET <NRf>	<p>Sets the overvoltage protection trip point in volts (default) or in millivolts.</p> <p><b>Range:</b> 0 to 110% of model maximum output voltage (VMAX)</p> <p><b>Initial value:</b> 110% of model maximum voltage</p>
OVSET?	<p>Returns the floating point value of the present overvoltage protection limit.</p> <p><b>Response:</b> OVSET &lt;0+NRf&gt;</p>
SRQ <boolean>	<p>SRQ ON enables/disables the supply to respond to a variety of fault conditions by generating an IEEE-488 GPIB service request. With SRQ ON, the SRQ line will be asserted true whenever the FAU bit in the Serial Poll Register changes from 0 to 1. Therefore, the mask register, in addition to specifying which conditions set the FAU bit, also determines which conditions can generate service requests. Seven (7) supply conditions are defined as faults: CV, CC, OV, OT, SD, FOLD, and PON. Use the FAULT? query to discover which condition caused the service request except for PON. A request for service at Power ON (PON) is set via the rear panel switch on the supply. SRQ remains disabled until the FAU bit in the Serial Poll Register is cleared by a FAULT? query.</p> <p><b>Initial value:</b> SRQ 0 (OFF)</p>
SRQ?	<p>Returns the integer value of the present enabled/disabled status to generate GPIB service requests.</p> <p><b>Response:</b> SRQ &lt;mode&gt; where mode is:            0 (disabled) or            1 (enabled)</p>

Command	Description
STS?	<p>Returns the integer value of the present Status Register. Status conditions are stored in the 8-bit Status Register. Each bit represents a separate condition. When the condition is true, the corresponding bit is 1. Bits remain set in the Status Register as long as the condition is true.</p> <p><b>Response:</b> STS &lt;status mask&gt; where status mask is the decimal equivalent of the total bit weights for the operating conditions as listed in the Status Register.</p>
UNMASK <mnemonics>	<p>Enables the specified operating conditions for fault occurrence. Mnemonics are separated from each other by commas and may be sent in any order. Specifying one or more mnemonics which describe the conditions (or the decimal equivalent of their total bit weight) enables the selected conditions to set bits in the Fault and Status Registers. A bit is set in the Fault Register when the corresponding bit in the Status Register changes from 0 to 1 and the corresponding bit the Mask Register is 1. Whenever any bit is set in the Fault Register, the FAU bit (position 0) is set in the Serial Poll Register.</p> <p><b>UNMASK mnemonics:</b> CV (Constant Voltage), CC (Constant Current) OV (Overvoltage Protection) OT (Over Temperature Protection) SD (Shutdown) FOLD (Foldback) NONE (all conditions disabled) ALL (all conditions enabled)</p> <p><b>Initial value:</b> UNMASK NONE</p>
UNMASK?	<p>Returns the integer value of the fault conditions currently enabled (UNMASKed).</p> <p><b>Response:</b> UNMASK &lt;fault mask&gt; where fault mask is the decimal equivalent of the total bit weights for the operating conditions as listed in the Status and Fault Registers.</p>

Use Status and Fault Control commands to monitor the operating condition of the unit:

- CV (Constant Voltage operation),
- CC (Constant Current operation),
- OV (Overvoltage Protection tripped),
- OT (Over Temperature Protection tripped),
- SD (Shutdown active),
- FOLD (Foldback mode operation),
- ERR (Remote programming error)

The value of the over-voltage trip point may be set and queried. The foldback condition to monitor may be selected and queried. Enabled fault conditions will generate a GPIB service request if enabled. The most recent programming error may be queried.

## 5.4 Examples of Using the M9A Commands

The following examples demonstrate programming control and to readback the output using the M9A commands. The maximum voltage and current output is dependent on the particular model. The examples list only the M9A commands; the code required to send the commands is dependent on the type of language you are using (e.g., C or BASIC) and GPIB hardware (e.g., National Instruments).

**EXAMPLE:** Program a unit with no load at the output to 5 VDC @ 1 A, and verify the output.

```
// Use ERR? after each command to verify no programming errors.
// turn on the unit.
CLR           // clear the unit to its power-on default settings.
RST          // reset the unit.
ISET 1.0     // program output current to 1.0 A.
ISET?        // confirm the output current setting. (response: ISET 1.000)
VSET 5.0     // program output voltage to 5.0 VDC.
VSET?        // confirm the output voltage setting. (response: VSET 5.000)
IOUT?        // measure the actual output current (response: IOUT ~0.000 with no load
              // on output).
VOUT?        // measure the actual output voltage (response: VOUT ~5.000).
```

**EXAMPLE:** Program a unit with no load at the output to generate a GPIB service request upon an overvoltage protection trip condition. (Must GPIB not RS-232.)

```
// Use ERR? after each command to verify no programming errors.
// assure that PON is not selected in the rear panel switch
// and the front panel SRQ led is OFF. Turn on the unit.
CLR           // clear the unit to its power-on default settings.
RST          // reset the unit.
OVSET 4.0    // program the OVP trip point to 4.0 VDC.
OVSET?       // confirm the OVP trip point setting (response: OVSET 4.000).
ISET 1.0     // program output current to 1.0 A.
VSET 3.0     // program output voltage to 3.0 VDC.
UNMASK OV    // program the unit to report OVP trip.
UNMASK?      // confirm that OVP fault is enabled (response: UNMASK 8).
SRQ 1        // enable the GPIB service request upon a fault.
SRQ?         // confirm the GPIB service request enabled (response: SRQ 1).
FAULT?       // confirm no faults occurred (response: FAULT 0).
              // confirm that the front panel OVP led and SRQ led is not active.
VSET 7.0     // program output voltage to 7.0 VDC - cause OVP trip!
              // confirm that the front panel OVP led and SRQ led is active, and
              // the unit issued a GPIB service request (use a serial poll).
```

**EXAMPLE:** Program a unit with no load at the output to change its output voltage and current to 5 VDC @ 1A at the same time.

```
// Use ERR? after each command to verify no programming errors.
// turn on the unit.
CLR           // clear the unit to its power-on default settings.
RST           // reset the unit.
HOLD 1       // program the unit to hold changes until TRG received.
HOLD?        // confirm that hold is enabled (response: HOLD 1).
ISET 1.0     // program output current to 1.0 A upon TRG.
VSET 5.0     // program output voltage to 5.0 VDC upon TRG.
IOUT?        // measure the actual output current (response: IOUT ~0.000).
VOUT?        // measure the actual output voltage (response: VOUT ~0.000).
TRG          // trigger the unit to implement ISET and VSET.
IOUT?        // measure the actual output current (response: IOUT ~0.000 with no
              // load on output).
VOUT?        // measure the actual output voltage (response: VOUT ~5.000).
HOLD 0       // turn off hold mode.
```

This page intentionally left blank.

# SECTION 6 OPERATION WITH M85 OPTION

## 6.1 Introduction

The following sections describe the operation of the power supply with the M85 option and how to configure the option for your application.

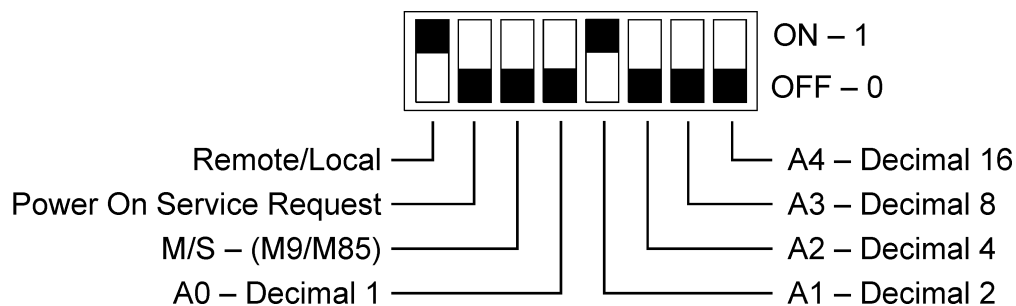
The M85 option enables you to use your power supply as an auxiliary unit. The master unit would be a power supply with the M9 interface option. Both IEEE-488.2 and RS-232 are supported, and the auxiliary unit or units share the GPIB address or RS-232 interface with the master unit. Up to 30 auxiliary units can be controlled by the master unit.

This option can be used with the 1kW and 3kW supplies and supports the SCPI language. Refer to Section 5, above, for programming information.

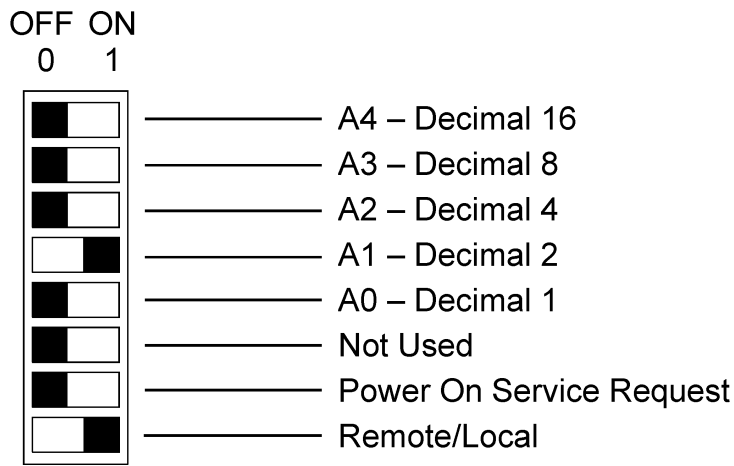
## 6.2 Configuration

The DIP switch S1, represented in Figure 6–1 through Figure 6–3, is accessible from the rear panel. It allows configuration of the M85 for auxiliary operation.

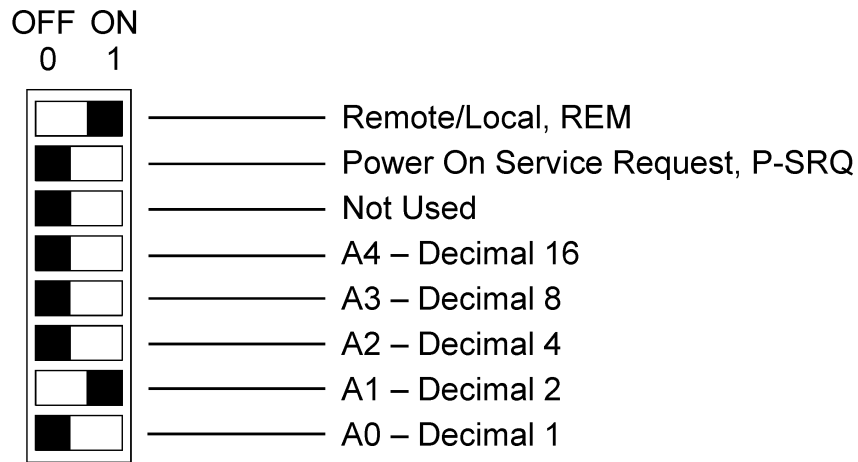
The figures show the settings for the M85 configured as an auxiliary at channel 2, in remote mode, with no SRQ generated upon initialization.



**Figure 6–1 DCS and Pro-T Configuration Switch, Auxiliary in Remote Mode**



**Figure 6–2 DLM Series 3kW & 4kW Configuration Switch, Auxiliary in Remote Mode**



**Figure 6–3 DLM Series 600W Configuration Switch, Auxiliary in Remote Mode**



See Table 6–1 for definitions of all S1 switch settings:

**Table 6–1 Definitions of S1 Switch Settings**

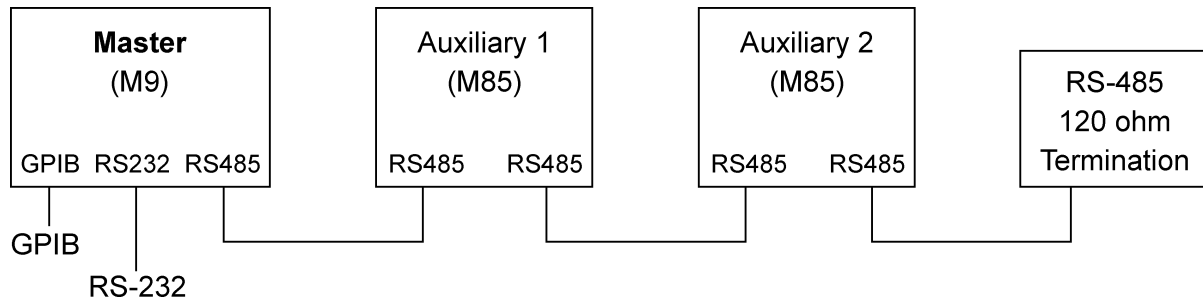
Switch	ON	OFF
S1-1	Remote control	Local control
S1-2	PON SRQ ON. Generates GPIB service request to computer/controller when the power supply initializes upon first communication. The front panel SRQ LED will also turn ON. Issuing a serial poll clears service request and turns OFF LED.	Disables GPIB service request upon initialization.
S1-3	Not applicable. Must be OFF.	Unit operates as an auxiliary unit.
S1-4–8	Set channel number for unit in binary:	(Channels 0 and 1 see note below.)
S1-4	Binary 1 ON	Binary 1 OFF
S1-5	Binary 2 ON	Binary 2 OFF
S1-6	Binary 4 ON	Binary 4 OFF
S1-7	Binary 8 ON	Binary 8 OFF
S1-8	Binary 16 ON	Binary 16 OFF

**Note:** Channel numbers 0 and 1 are invalid for any auxiliary device, because the M9 implementation of the SCPI language reserves channel 0 for the “global” address to address all channels, and it reserves channel number 1 as the default number for the master channel.

### 6.3 System Installation

Follow the steps below, illustrated in Figure 6–4, to install the configured M85 into your system:

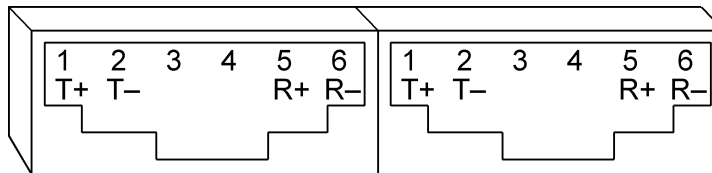
1. Configure the M85 as described in the Configuration section above.
2. Connect the master unit’s RS-485 output connector, J2, to the auxiliary unit’s RS-485 input connector, J1, using the modular cable.
3. If there are additional auxiliary units, connect the installed auxiliary unit’s RS-485 output connector, J2, to the additional auxiliary unit’s RS-485 input connector, J1, using the modular cable.
4. Connect the 120 ohm resistive termination assembly to the last auxiliary unit’s J2 connector.
5. Connect power to the system, power it up, and verify that the green REMOTE LED on the front panel is ON.
6. Test the link by communicating with the auxiliary unit from the master unit, using the **\*IDN[n]?** command. Note that [n] is the channel number—for example, **\*IDN2?** calls channel 2. (In response to this string, the system returns the power supply model number and the firmware version.)



**Figure 6–4 RS-485 System Interconnection with Two Auxiliaries**

### 6.4 RS-485 Interface

The RS-485 interface is accessible through the two rear-panel, 6-pin, RJ-11 connectors, J1 and J2, depicted in Figure 6–5.



**Figure 6–5 M85 Rear Panel RS-485 Connectors Pinout**

### 6.5 Programming the M85 Unit (Example)

The following example programs the M85 unit to:

- Turn on
- Initialize to 2 VDC, at 1A
- Set overvoltage protection level at 3 VDC
- Verify proper power-on initialization
- Save and store changes.

```
// Use SYST:ERR? after each command to verify no programming errors.
// turn on the unit.
*CLS // clear the unit to its power on default settings.
*RST // reset the unit.
CAL[n]:INIT:CURR 1.0 // set power-on initial current to 1.0A.
CAL[n]:INIT:CURR? // confirm power-on initial current setting.
CAL[n]:INIT:VOLT 2.0 // set power-on initial voltage to 2.0V.
CAL[n]:INIT:VOLT? // confirm power-on initial voltage setting.
CAL[n]:INIT:VOLT:PROT 3.0 // set power-on initial overvoltage protection to 3.0V.
CAL[n]:INIT:VOLT:PROT? // confirm power-on initial overvoltage protection setting.
CAL[n]:UNLOCK "6867" // unlock nonvolatile memory for calibration value storage.
CAL[n]:STORE // store the calibration values in nonvolatile memory.
CAL[n]:LOCK // lock nonvolatile memory for calibration value protection.
// cycle power to unit.
// note voltage is initialized to 2.0 VDC via front panel.
SOUR[n]:CURR? // confirm power-on initial current setting.
SOUR[n]:VOLT? // confirm power-on initial voltage setting.
SOUR[n]:VOLT:PROT? // confirm power-on initial overvoltage protection setting.
```

## SECTION 7

# SCPI STATUS IMPLEMENTATION

The M9 product line, as of August 1999, has branched into two families in regards to the SCPI Status implementation. There is a subtle but meaningful difference regarding how the SCPI Standard Event Status Register is treated in the newer of these two SCPI implementations.

If your product has the DCS or PRO-T model name designation, it incorporates the older implementation. If your product has the DLM model designation, it is the newer of the two SCPI implementations.

To ascertain what implementation you have, you may query your supply via SCPI with the \*IDN? command:

- Issue the \*IDN? query command. A response string to the \*IDN? query command should result, such as “Sorensen, **DCS**33-33, B90000-0, 2.05,1.2” or such as “Sorensen, **DLM**33-33, B90000-0, 1.01,1.2”
- The second to last number in the response to the \*IDN? response string represents the *version* number of the respective firmware you have. So, for example, if your response string is something similar to “Sorensen, **DCS**33-33, B90000-0, **2.05**,1.2”, then you have DCS firmware version 2.05.

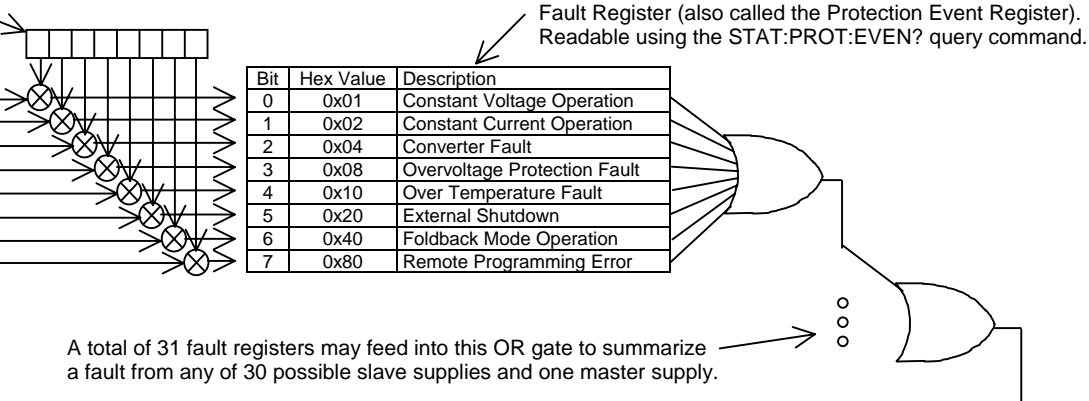
Once you have identified the branch of firmware you have, then you can determine which implementation of the SCPI Standard Event Status Register you have. The SCPI Standard Event Status Register, SESR for short, is read using the \*ESR? command. In both implementations, this command operates as expected. But under what conditions the contents of the SESR is meaningful is different depending upon whether you have the DCS implementation or the DLM implementation.

To understand the subtle but noteworthy difference, find the SESER and the SESR registers on the following two figures. They show that the order of occurrence for the SESER and the SESR in a data flow sense is reversed between these two implementations. Read the text in the two figures that follow to better understand the consequences of this implementation difference.

If you have DCS firmware, certain SCPI compatibility issues may be improved by using a special command at power-on time. DCS firmware versions 2.04 and earlier incorrectly cleared the status byte after serial polling; contrary to the SCPI/IEEE standard. And the persistence of various bits in the status byte in version 2.04 and older is an issue. For DCS firmware versions 2.05 and later, the special command "SYST:ADJUSTAXAA 1" may be issued once at each power-on time to activate a fix for these status problems. This paragraph does not apply to the DLM firmware.

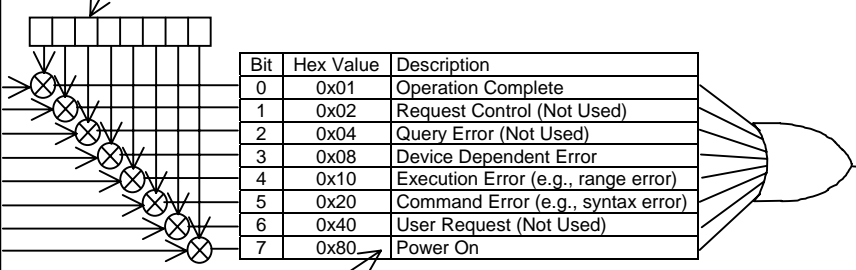
The Protection Enable Register. Readable using the STAT:PROT:ENAB? query command. Write-able using the STAT:PROT:ENAB <value> command. Used to select what fault events may set a bit in the Fault Register. Certain faults can occur even if they are not enabled. This is because the Protection Enable Register merely filters which events are allowed to affect the Fault Register, not whether those events can occur or not. An exception to this rule involves the Constant Voltage Operation, Constant Current Operation, and Foldback Mode Operation bits. If these bits are not enabled, then mode changes shall not cause a shutdown. Read about these bits further in the manual.

Various fault events must be enabled by the protection enable register before they are recorded in the fault register.



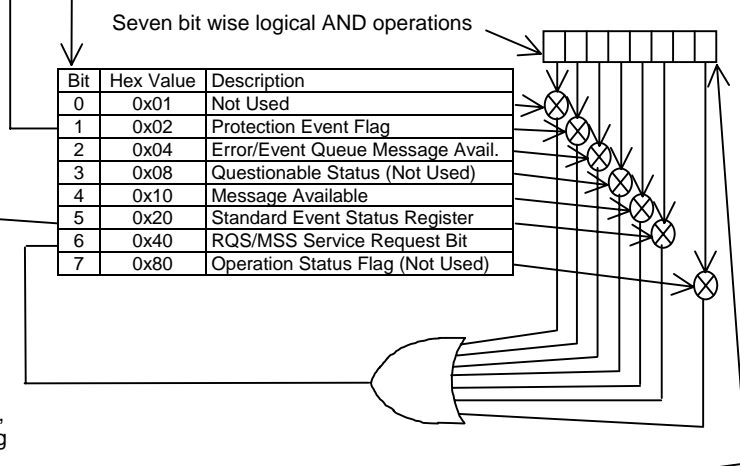
The SESER (Standard Event Status Register). This register is read using the \*ESE? SCPI query command. This register is written to using the \*ESE <value> command. A "1" in the appropriate bit location enables various standard event type events to be recorded in the SESR.

Various events must be enabled by the SESER before they are recorded in the SESR. This makes polling the SESR with the SESER clear unfeasible.



The SESR (Standard Event Service Register). In the DCS implementation, the SESER must have the appropriate bits set to enable even the recording of an event in the SESR. The SESR is read using the \*ESR?.

The SCPI Status Byte. Read using either the \*STB? command, or the GPIB serial poll operation. In the M9E implementation, reading this byte does *not* clear it; this is per SCPI/IEEE requirements.



The Service Request Enable Register (SRER). Used to enable which Status Byte bits can affect the service request bit. \*SRE? reads. \*SRE <value> writes.

## SCPI Status Structure for the DCS Firmware Implementation

The Protection Enable Register. Readable using the STAT:PROT:ENAB? query command. Write-able using the STAT:PROT:ENAB <value> command. Used to select what fault events may set a bit in the Fault Register. Certain faults can occur even if they are not enabled. This is because the Protection Enable Register merely filters which events are allowed to affect the Fault Register, not whether those events can occur or not. An exception to this rule involves the Constant Voltage Operation, Constant Current Operation, and Foldback Mode Operation bits. If these bits are not enabled, then mode changes shall not cause a shutdown. Read about these bits further in the manual.

Various fault events must be enabled by the protection enable register before they are recorded in the fault register.

Bit	Hex Value	Description
0	0x01	Constant Voltage Operation
1	0x02	Constant Current Operation
2	0x04	Converter Fault
3	0x08	Overvoltage Protection Fault
4	0x10	Over Temperature Fault
5	0x20	External Shutdown
6	0x40	Foldback Mode Operation
7	0x80	Remote Programming Error

Fault Register (also called the Protection Event Register). Readable using the STAT:PROT:EVEN? query

A total of 31 fault registers may feed into this OR gate to summarize a fault from any of 30 possible slave supplies and one master supply.

The SESER (Standard Event Status Enable Register). This register is read using the \*ESE? SCPI query command. This register is written to using the \*ESE <value> command. A "1" in the appropriate bit location enables that corresponding bit from the SESR to pass through to the input of the OR gate to be included in the SESR summary bit (bit 5) in the SCPI Status Byte.

Various events set the SESR bits directly.

Bit	Hex Value	Description
0	0x01	Operation Complete
1	0x02	Request Control (Not Used)
2	0x04	Query Error (Not Used)
3	0x08	Device Dependent Error
4	0x10	Execution Error (e.g., range error)
5	0x20	Command Error (e.g., syntax error)
6	0x40	User Request (Not Used)
7	0x80	Power On

The SESR (Standard Event Service Register). In the DLM implementation, masking does not prevent events from setting bits in the SESR. This facilitates polling as one means of detecting these events since the SESR can be polled (read) using the \*ESR? command, irrespective of the bits set or not set in the Standard Event Status Enable Register.

The SCPI Status Byte. Read using either the \*STB? command, or the GPIB serial poll operation. In the M9E implementation, reading this byte does not clear it; this is per SCPI/IEEE requirements.

Seven bit wise logical AND operations

Bit	Hex Value	Description
0	0x01	Not Used
1	0x02	Protection Event Flag
2	0x04	Error/Event Queue Message Avail.
3	0x08	Questionable Status (Not Used)
4	0x10	Message Available
5	0x20	Standard Event Status Register
6	0x40	RQS/MSS Service Request Bit
7	0x80	Operation Status Flag (Not Used)

Eight bit wise logical AND operations

The Service Request Enable Register (SRER). Used to enable which Status Byte bits can affect the service request bit. \*SRE? reads. \*SRE <value> writes.

## SCPI Status Structure for the DLM Firmware Implementation

# INDEX

- A
- Auxiliary Unit (M85 only), 2-4
- C
- CALibrate[n], 3-10
- Calibration
  - Current Measurement/Readback Calibration, 4-10
  - Current Programming and Readback Calibration, 4-13
  - Current Programming Calibration, 4-7
  - Five-Point Current Calibration Procedure, 4-9
  - Five-Point Voltage Programming Calibration, 4-4
  - Overvoltage Protection Programming Calibration, 4-7
  - Setup for Calibration, 4-2
  - Two-point current programming Calibration, 4-8
  - Two-Point Voltage Programming Calibration, 4-3
  - Voltage Measurement/Readback Calibration, 4-6
  - Voltage Programming and Readback Calibration, 4-11
  - Voltage Programming Calibration, 4-3
  - Warning, 4-1, 4-2
- Conventions, 3-7
  - [ ], 3-7
- D
- Default Master Channel 1, 2-4
- E
- Error/Event Queue, 3-3
  - SCPI Error Codes, 3-4
- Examples of Using M9A Commands, 5-10
- Examples of Using SCPI Commands, 3-25
- External User Control Signal Connector
  - FAULT output signal, 2-5
  - FOLDBACK output signal, 2-5
  - ISOLATION output signal, 2-5
  - POLARITY output signal, 2-5
  - SENSE output signal, 2-5
  - SHUTDOWN TTL input signal, 2-5
  - SYNC output signal, 2-5
- External User Control Signal Connector, 2-5
  - Illustration of Open Collector, TTL Input, and Relay Output Circuits, 2-6
  - Warning, 2-6
- F
- Features and Functions, 1-1
- Front Panel REMOTE LED, 2-1
- Front Panel SRQ LED, 2-4
- G
- Global Channel 0, 2-4
- I
- IEEE-488.2 and SCPI Conformance Information, 3-6
- IEEE-488.2 Common Command Subsystem, 3-8
  - \*CLS, 3-8
  - \*ESE, 3-8
  - \*ESR?, 3-8
  - \*IDN?, 3-8
  - \*OPC, 3-8
  - \*RST, 3-8
  - \*SRE, 3-8
  - \*STB?, 3-9
  - \*TST?, 3-9
  - \*WAI, 3-9
- L
- Local Operation, 2-3

- M
- M85, 2-8  
M9, 1-1  
M9A Command Operation  
  ASTS?, 5-6  
  CLR, 5-4  
  DLY, 5-7  
  ERR?, 5-7  
  Error Register, 5-3  
  FAULT?, 5-7  
  FOLD, 5-7  
  HOLD, 5-4  
  ID?, 5-4  
  IMAX, 5-5  
  IOUT?, 5-5  
  ISET, 5-5  
  MASK, 5-8  
  OUT, 5-5  
  OVSET, 5-8  
  RST, 5-4  
  SRQ, 5-8  
  Status and Fault Registers, 5-2  
  Status Byte, 5-1  
  STS?, 5-9  
  TRG, 5-5  
  UNMASK, 5-9  
  VMAX, 5-6  
  VOUT?, 5-6  
  VSET, 5-6  
Master Unit (M9C only), 2-4  
MEASure[n], 3-13  
Multiple Source Control with the M85 Option,  
  2-8  
  RS-485 Rear Panel RJ-11 Connector  
  Pinout, 2-8
- O
- Operation Status and Questionable Status  
  Registers, 3-3  
OUTPut[n], 3-14
- P
- Parameter Definitions, 3-6  
  0+NR1, 3-6  
  0+NRf, 3-6  
  boolean, 3-6  
  NR1, 3-6  
  NRf, 3-6
- string, 3-6  
Power-On Conditions, 2-3  
  Default, 2-3  
Protection Event Status Register, 3-3
- Q
- Queries, 3-7
- R
- Ramping  
  Description, 3-19  
Rear Panel Configuration Switch  
  Address Selection, 2-4  
  Master/Auxiliary Selection, 2-4  
  Power-On GPIB Service Request (PON  
  SRQ) Selection, 2-4  
  Remote/Local Selection, 2-3  
Rear Panel Configuration Switch S1, 2-2  
Remote Operation, 2-3  
Remote Programming via RS-232, 2-7  
  RS-232 Rear Panel RJ-11 Connector  
  Pinout, 2-7
- S
- SCPI Command Operation  
  CALibrate[n], 3-10  
  MEASure[n], 3-13  
  OUTPut[n], 3-14  
  Warning, 3-14  
  SOURce[n], 3-15  
  STATus[n], 3-20  
  TRIGgered[n], 3-24  
Serial Poll Operation, 3-6  
Setup Procedure, 2-1  
SOURce[n], 3-15  
Specifications, 1-2  
Standard Event Status Register (ESR), 3-2  
Status Byte, 3-1  
Status Flags Register, 3-19  
STATus[n], 3-20  
Synchronization Pulse, 2-5
- T
- TRIGgered[n], 3-24
- U
- Units, 3-7