EX7000 SCPI COMMANDS

PROGRAMMER'S MANUAL

P/N: 82-0153-000
Released: 05 May 2015
Rev A

VTI Instruments Corp.

2031 Main Street
Irvine, CA 92614-6509
(949) 955-1894

Table of contents

CERTIFICATION

VTI Instruments Corp. (VTI) certifies that this product met its published specifications at the time of shipment from the factory.

WARRANTY

The product referred to herein is warranted against defects in material and workmanship for a period of one year from the receipt date of the product at customer's facility. The sole and exclusive remedy for breach of any warranty concerning these goods shall be repair or replacement of defective parts, or a refund of the purchase price, to be determined at the option of VTI.

VTI warrants that its software and firmware designated by VTI for use with a product will execute its programming when properly installed on that product. VTI does not however warrant that the operation of the product, or software, or firmware will be uninterrupted or error free.

LIMITATION OF WARRANTY

The warranty shall not apply to defects resulting from improper or inadequate maintenance by the buyer, buyer-supplied products or interfacing, unauthorized modification or misuse, operation outside the environmental specifications for the product, or improper site preparation or maintenance.

VTI Instruments Corp. shall not be liable for injury to property other than the goods themselves. Other than the limited warranty stated above, VTI Instruments Corp. makes no other warranties, express or implied, with respect to the quality of product beyond the description of the goods on the face of the contract. VTI specifically disclaims the implied warranties of merchantability and fitness for a particular purpose.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subdivision (b)(3)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

VTI Instruments Corp.
2031 Main Street

Irvine, CA 92614-6509 U.S.A

SUPPORT RESOURCES

Support resources for this product are available on the Internet and at VTI Instruments customer support centers.

VTI Instruments Corp.
World Headquarters

VTI Instruments Corp.
2031 Main Street
Irvine, CA 92614-6509

Phone: (949) 955-1894
Fax: (949) 955-3041

VTI Instruments
Cleveland Instrument Division

5425 Warner Road
Suite 13
Valley View, OH 44125

Phone: (216) 447-8950
Fax: (216) 447-8951

VTI Instruments
Lake Stevens Instrument Division

3216 Wetmore Avenue, Suite 1
Everett, WA 98201

Phone: (949) 955-1894
Fax: (949) 955-3041

VTI Instruments, Pvt. Ltd.
Bangalore Instrument Division

#75/76, Millers Road
Bangalore – 560 002 India
Phone: +91 80 4040 7900
Fax : +91 80 4170 0200

Asia Support
Phone: +852 9177 6127

Technical Support

Phone: (949) 955-1894
Fax: (949) 955-3041

E-mail: support@vtiinstruments.com

Visit http://www.vtiinstruments.com for worldwide support sites and service plan information.

SECTION 1

Introduction

Background

The intent of the programmer's manual is to describe the EX7000 SCPI commands and to introduce its concepts, structure, and capabilities to software and test application. The reader is expected to be familiar with SCPI commands. Understanding the architecture and concept of SCPI commands will significantly help the reader follow the design of the EX7000 SCPI commands described in this manual.

This guide provides instruction for programming the EX7000 series only.

Glossary

Throughout this document, the following terms will be used:

EX7000	an LXI-based radio frequency (RF) switch system which is part of VTI Instruments' Next Generation line of products
RDB	an acronym for **R**elay **D**river **B**oard; hardware that interfaces between the controller and the actual microwave switches on an EX7000.

Basic Concepts

1) The EX7000 SCPI commands are designed to work with the EX7000 platforms only.

2) An EX7000 platform provides a generic collection of relay drive lines and reset lines intended to drive RF switch coils. The EX7000 SCPI commands allow the user to control coils individually.

SECTION 2

Connection

SCPI RAW

The basic operation of the SCPI socket is:

Connect to TCP port 5025 on the device.

Repeat:

Send a command or query terminated by a newline.

If a query, read query result terminated by a newline.

Close the socket.

There is only one connection at a time allowed on the socket. When a connection is made, the listener is closed. This allows exclusive access to the device and provides an inherent locking mechanism.

To select the SCPI raw interface, use a "TCPIP::<hostname>::5025::SOCKET" resource string, where <host> is the hostname or the IP address of the EX7000 instrument.

VXI-11

The vxi-11 interface acts as a multiplexer to the SCPI socket. Many clients can connect at the same time using vxi-11. All requests will be handled in the order received. If the client uses vxi-11 locking, it will have exclusive access.

When a client is using the SCPI socket, the VXI-11 interface will behave as if a VXI -11 client has obtained the lock.

To select the VXI -11 interface, use a "TCPIP::< hostname >::INSTR" resource string, where <host> is the hostname or the IP address of the EX7000 instrument.

Commands

Syntax

Multiple commands and queries may be given in the same message separated by ';' like SCPI. The leading ':' when using multiple commands is optional.

On the other hand, case does not matter. Both long and short forms of a command are accepted and treated in the same way as in SCPI. Multiple queries in the same message are separated by ';' like SCPI. And any whitespace in the commands is ignored.

Most of the following commands take a list of coils as a parameter. The format of a coil is:

K{*rdbnumber*}_{*coilnumber*} or R{*rdbnumber*}_{*resetnumber*}

with the following rules

- The {rdbnumber} ranges from 1-8 to indicate which RDB is being specified
- The {coilnumber} ranges from 1-72 to indicate which coil is being specified
- The {resetnumber} ranges from 1-12 to indicate which reset line is being specified
- The list is comma separated.
- Ranges are colon separated.
- Ranges must be the same type. (All coils or all resets)
- Ranges can be increasing or decreasing and cross RDB boundaries.

Example: Coils 1,3,4,5 on RDB 2: (@ K1_1, k1_3:k1_5)

The example above would close the first coil and the third to fifth coils in RDB 2.

EX7000 Commands

EX7000 SCPI commands provide a set of commands to close and open as well as verify the state of the drive line and the reset line on the RDB installed in the EX7000 instrument. Please note that the lower case in the command can be omitted. For example, ROUTe:CLOSe can be recognized by ROUT:CLOS and both syntax achieve the same goal.

ROUTe:CLOSe (@<list>)

Close drive lines or reset lines specified in <list>.

When multiple ROUT:CLOS (or ROUT:OPEN) commands are given in the same message, they must have a ROUT:MOD:WAIT command between them.

ROUTe:CLOSe? (@<list>)

Return the open or closed state of the drive lines or reset line specified in <list>. The values 0 indicates open and 1 indicates closed.

The drive value or the confidence value is used depending on the Verify state of each line.

ROUTe:OPEN (@<list>)

Open drive lines or reset lines specified in <list>.

When multiple ROUT:OPEN (or ROUT:CLOS) commands are given in the same message, they must have a ROUT:MOD:WAIT command between them.

ROUTe:OPEN:ALL

Open all drive lines.

ROUTe:CHANnel:VERify bool,(@<list>)

Configure ROUT:CLOS? to return the indicator state instead of the driven state.

ROUTe:CHANnel:VERify? (@<list>)

Returns the Verify mode of the coils in <list>.

ROUTe:CHANnel:VERify:POLarity [INVerted|NORMal],(@<list>)

Configure the polarity confidence values in the <list> to be inverted, or as-is.

ROUTe:CHANnel:VERify:POLarity? (@<list>)

Returns the confidence polarity setting of the coils in <list>. A 1 means the confidence value is inverted, and a 0 means the confidence value as-is.

ROUTe:CHANnel:VERify:POSition:STATe? (@<list>)

If Verify is set for a coil, return the driven state. If Verify is not set, return the confidence value with polarity applied.

.

ROUTe:MODule:BUSY?

Return 1 if relays are settling or 0 if not. May wait an extra 100ms before returning.

.

ROUTe:MODule:WAIT

Waits for relays to debounce. Waits at least 100ms after debouncing.

Common Commands

A subset of SCPI Common commands that are supported by EX7000.

1.1.1 *IDN?

Provides the standard “Vendor,Model,Serial,Revision” response.

1.1.2 SYSTem:ERRor?

Returns the next error in the error queue.

Possible errors:

0, “No error”

-102, “Syntax error; Unknown command: [<erroneous text>]”

-350, “Queue overflow”

-400, “rdb out of range”

-401, “coil out of range”

-402, “Mixed Reset lines and Coil lines in range”

-403, “range crosses rdb boundaries”

1.1.3 *STB?

Always returns “0”.

1.1.4 *RST

Performs a Reset on the device.

SECTION 3

Application Examples

The following section provides some practical examples of how the EX7000 SCPI commands can be used together to meet certain user requirements. Situations will be provided and an appropriate programming example will follow.

Close, open and verify a single relay

```
//Close the 3rd relay on the 2nd RDB.
ROUT:CLOSE(@K2_3);
//Verify that the 3rd relay on the 2nd RDB is closed. A ‘1’ should be returned.
ROUT:CLOSE?(@K2_3);
//Open the 3rd relay on the 2nd RDB.
ROUT:OPEN(@K2_3);
//Verify that the 3rd relay on the 2nd RDB is open. A ‘0’ should be returned.
ROUT:CLOSE?(@K2_3);
```

Close, open and verify couple relays

```
//Close the 3rd relay on the 2nd RDB, the 10th relay on the 1st RDB and the 5th relays on the 3 RDB.
ROUT:CLOSE(@K2_3, K1_10, K3_5);
//Verify that the 3rd relay on the 2nd RDB, the 10th relay on the 1st RDB and the 5th relays on the 3 RDB are all closed. “1,1,1” should be returned.
ROUT:CLOSE? (@K2_3, K1_10, K3_5);
//Open the 3rd relay on the 2nd RDB, the 10th relay on the 1st RDB and the 5th relays on the 3 RDB.
ROUT:OPEN(@K2_3, K1_10, K3_5);
// Verify that the 3rd relay on the 2nd RDB, the 10th relay on the 1st RDB and the 5th relays on the 3 RDB are all open. “0,0,0” should be returned.
ROUT:CLOSE?(@K2_3, K1_10, K3_5);
```

Close, open and verify a series of relays

```
//Close the first 5 relays on the 1st RDB.
ROUT:CLOSE(@K1_1:K1_5);
//Verify the first 5 relays on the 1st RDB are all closed. “1,1,1,1,1” should be returned.
ROUT:CLOSE?(@K1_1:K1_5);
//Open the first 5 relays on the 1st RDB.
ROUT:OPEN(@K1_1:K1_5);
//Verify the first 5 relays on the 1st RDB are all open. “0,0,0,0,0” should be returned.
ROUT:CLOSE?(@K1_1:K1_5);
```

Close multiple relays, open all relays at once and verify the state of the relays

```
//Close the 1st relay on the 1st RDB.
ROUT:CLOSE(@K1_1);
//Wait until all the relays are settle. This command is required between multiple close or open command.
ROUT:MOD:WAIT;
//Close the 2nd, 3rd, 4th, 5th relay on the 1st RDB.
ROUT:CLOSE(@K1_2,K1_3,K1_4,K1_5);
//Wait until all the relays are settle. This command is required between multiple close or open command.
ROUT:MOD:WAIT;
//Close the 6th to 10th relays on the 1st RDB.
ROUT:CLOSE(@K1_6:K1_10);
//Wait until all the relays are settle. This command is required between multiple close or open command.
ROUT:MOD:WAIT;
//Open all the relays on all the RDB.
ROUT:OPEN:ALL;
//Verify all the closed relays are open. “0,0,0,0,0,0,0,0,0,0” should be returned.
ROUT:CLOSE?(@K1_1,K1_2,K1_3,K1_4,K1_5,K1_6:K1_10);
```