HT3530 Communication Interface Operation Manual

Simplified Chinese version

Aug, 2010

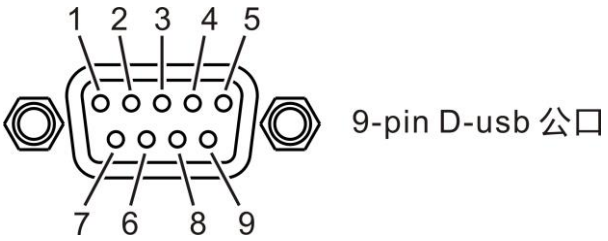
The first version Rev1.0.0

Communication Interface

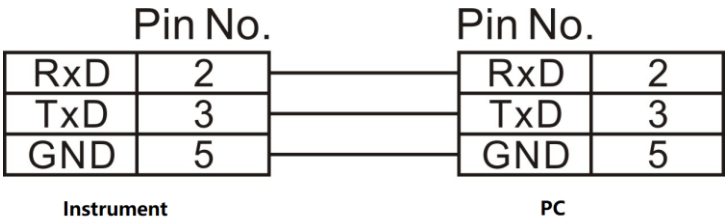
7.1 RS232C communication method

RS232C communication mode adopts 3-wire communication mode.
The baud rate is set on the instrument settings page. 1 stop bit, no parity.

Interfaces and Cables



Connection method



7.3 SCPI Commands

7.3.1 Common commands

Instrument commands are divided into two types: common command and SCPI command (Standard Commands for Programmable Instruments). Common command defined by the IEEE488.2-1987 standard. These commands are to be used with all instruments, but this instrument does not support all common commands. SCPI command is a tree structure.

1. ***IDN?** command

   Function: Query version number

   Example: Send: *IDN? Return: HOPETECH, HT3530, V1.0.0

7.3.2 SCPI Command Structure

The commands at the top of the command tree are called "root command" or simply "root." To access lower-level commands in the tree, you need to specify a specific path.

Command terminator: command input terminator, such as NL (newline character, ASCII code 10)

Colon (:):

A colon is the level of command, means to lower the level of the current command

Semicolon (;)

A semicolon indicates the beginning of multiple commands

Question mark (?)

A question mark (?) indicates query

Comma (,)

A comma is a separator of multiple parameters.

Spaces ()

A space is a separator between command and parameter

The following figure 6.1shows an example of how to use colons and semicolons to efficiently access commands in the command tree.

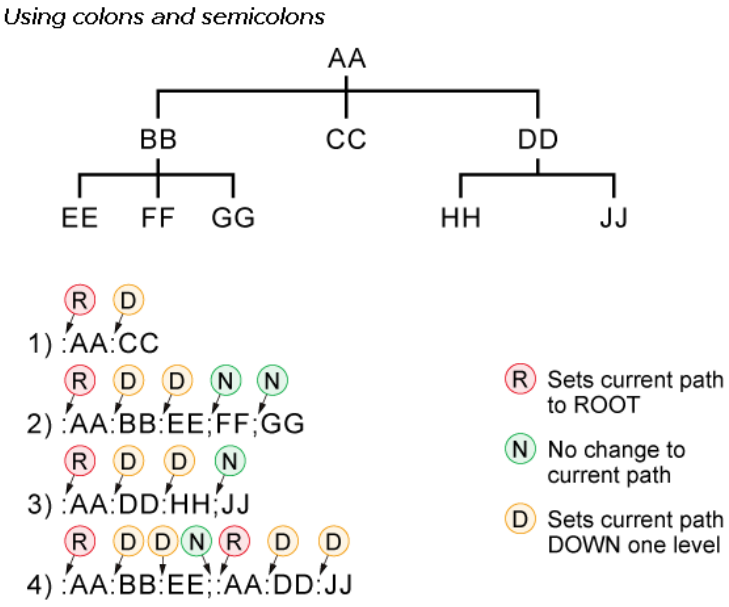


Figure 6.1 SCPI command tree

7.4 SCPI Subsystem

Data Format

NR1: integer (example: +12, -23, 34)

NR2: real numbers (example: +1.23, -23.45, 3.456)

NR3: Floating point scientific notation (example: +1.0E-2, -2.3E+4)

1. **START**

   Function: start test

   Returns: none

2. **STOP**

   Function: stop test

   Returns: none

3. **MEAS:STAT?**

   Function: Query test status

   Returns: 0 to stop, 1 to start

   Example: Querying Test Status

   Send: MEAS:STAT?

   Returns: 1

4. **FETCH?**

   Function: Query test results

   Return: resistance value, voltage value, range, test time, comparison result

   Example: Send: FETCH?

   Returns: 1.0000E+06,100.0000,2,10.0,1, that is, resistance 1M, voltage 100V, range 2, test time 10s, qualified

5. **MEAS: VOLTage**

   Function: Set or query the set voltage

   Returns: <NR1>

   Note: The set value range is: 1-1000

   Example: Query the set voltage

   Send: MEAS:VOLTage?

   Returns: 25

   Example: Setting Voltage

   Send: MEAS:VOLTage 25

6. **MEAS: RANGe {1-7}**

   Function: Set or query the test range

   Returns: 1-7

   Example: querying the test range

Send: MEAS:RANGe?
Returns: 1

Example: setting the test range
Send: MEAS:RANGe 1

7. MEAS:RANGe:AUTO {0|1}

Function: Set or query automatic range
Returns: 0|1

Example: Query test auto range
Send: MEAS:RANGe:AUTO?
returns: 1

Example: Setting the test auto range
Send: MEAS:RANGe:AUTO 1

8. MEAS:TIMe <NR2>

Function: Set or query measurement timing
Return: 0.0-999.999

Example: querying measurement timing
Send: MEAS:TIMe?
Returns: 10.0

Example: Setting the measurement timing
Send: MEAS:TIMe 10.0

9. CHG:TIMe <NR2>

Function: Set or query the charging delay
Return: 0.0-999.999

Example: setting the charging delay
Send: CHG:TIMe 1.0

Example: query charging delay
Send: CHG:TIMe?
Returns: 1.00

10. MEAS: UPLIM

Function: set or query the upper limit of the comparator
Returns: <NR3>,<NR3>
Note: Send a negative number to turn off the comparator

Example: setting the upper limit of the comparator
Send: MEAS:UPLIM 1.0E3

Example: Query the upper limit of the comparator
Send: MEAS:UPLIM?
Returns: 1.0E3

11. MEAS: LOLIM

Function: set or query the lower limit value of the comparator
Returns: <NR3>
Note: send a negative number to turn off the comparator

Example: setting the lower limit of the comparator
Send: MEAS:LOLIM 1.0E3

Example:query the lower limit value of the comparator
Send: MEAS:LOLIM?
Returns: 1.0E3