



**LDT 2000**

PRECISION THERMOMETER

MANUAL



INSTRUMENTS  
**leyro**

Leyro Instruments® doesn't accept warranty and liability claims neither upon this publication nor in case of improper treatment of the described products.

The document may contain technical inaccuracies and typographical errors. The content will be revised on a regular basis. These changes will be implemented in later versions. The described products can be improved and changed at any time without prior notice.

**© Copyright Leyro Instruments ®**  
**All rights reserved.**

# **TABLE OF CONTENTS**

## **1. GENERAL**

- 1.1. Symbol assertion*
- 1.2. General safety instructions*

## **2. PRODUCT DESCRIPTION**

- 2.1 Features*

## **3. OPERATION**

- 3.1 Preparation to work*
- 3.2 Power-on sequence*
- 3.3 Stand- alone benchtop operation (without PC)*
- 3.4 Operation with PC and Thermometer Utility software*
- 3.5 Operation with PC and custom software*
- 3.6 Internal battery*
- 3.7 Calibration period*
- 3.8 Calibration period*

## **4 ADVANCED USER INFORMATION**

- 4.1 Block diagram*
- 4.2 Probes*
- 4.3 Extender*
- 4.4 Resistance calibration*
- 4.5 Coefficients*
- 4.6 PC interface*

## **5. TYPICAL PERFORMANCE OF THE LDT 2000 THERMOMETER**

## **6. THERMOMETER UTILITY SOFTWARE**

- 6.1 Program at a glance*
  - 6.1.1 Installation and requirements*
  - 6.1.2 Main window overview*
- 6.2 Viewing and setting up the chart*
- 6.3 Recording measured data*
  - 6.3.1 Storing and using recorded data*
- 6.4 Configuration*
- 6.5 Calibration*
- 6.6 USB Drivers*

## **7. COMMUNICATION PROTOCOL**

### *7.1 Communication interfaces*

### *7.2 SCPI protocol*

#### 7.2.1 Syntax

#### 7.2.2 Commands

#### 7.2.3 Command termination character

#### 7.2.4 Structure of commands

#### 7.2.5 Query

#### 7.2.6 Parameters

#### 7.2.7 Special commands

#### 7.2.8 Special characters

#### 7.2.9 Commands' execution

#### 7.2.10 Default commands

### *7.3 Status registers*

#### 7.3.1 STB status register

#### 7.3.2 Events register (ESR)

#### 7.3.3 Operation register (OPER)

#### 7.3.4 Reliability register (QUES)

#### 7.3.5 Errors queue

#### 7.3.6 Mask registers

### *7.4 Commands' parameters*

#### 7.4.1 Binary <boolean>

#### 7.4.2 Integer number <int>

#### 7.4.3 Floating-point number <double>

#### 7.4.4 String

#### 7.4.5 Channels' list

### *7.5 Special commands*

### *7.6 Thermometer commands catalogue*

#### 7.6.1 Tree of commands

#### 7.6.2 Measurement commands

##### 7.6.2.1 :INITiate

##### 7.6.2.2 :CONFigure

7.6.2.3 :CONFigure?

7.6.2.4 Result readout commands :FETCh? and :READ?

7.6.2.5 :MEASure?

7.6.3 :UNIT – measurement's units handling

7.6.4 :INPut – measurement circuit configuration

7.6.4.1 :FILTer – configuration of line frequency filter

7.6.5 :SENSe – configuration of measurement's parameters

7.6.5.1 :AVERage – measurement averaging

7.6.5.2 :OVERflow –probe's temperature range exceeding flag

7.6.6 :CALibration – calibration commands

7.6.6.1 :ALL – autocalibration

7.6.6.2 :SECure – calibration unlocking

7.6.6.3 :RESistance – resistance measurement calibration

7.6.6.4 : CHannel 1 and :CHannel 2 – probes' calibration

7.6.7 :MEMory – internal EEPROM memory handling

7.6.8 :DISPlay – display operations

7.6.9 :STATus – status registers handling

7.6.9.1 :OPERation – operation register

7.6.9.2 QUEStionable – reliability register

7.6.9.3 :PRESet

7.6.10 :SYSTem

7.6.10.1 :ERRor – reading errors

7.6.10.2 :DATE <year, month, day>

7.6.10.3 :TIME <hour, minute, second>

## 7.7 *Notified errors*

7.7.1 Device errors

7.7.2 SCPI errors

## 1. GENERAL

The manual is a part of the scope of supply and serves to ensure proper handling and optimum functioning of the instrument.

LEYRO Instruments doesn't accept warranty and liability claims neither upon this publication nor in case of improper treatment of the described products.

For this reason, the manual must be read before start-up.

In addition, the manual is for all personnel who require knowledge concerning transport, setup, operation, maintenance and repair.

The manual must not be used for the purpose of competition without a written consent from Leyro Instruments and must also not be forwarded to third parties. Copies for personal use are permitted.

### 1.1 Symbol assertion



This symbol indicates a safety instruction.

These safety instructions should always be followed carefully.

By not following these instructions injuries of persons or material damage could happen.



This symbol indicates a note.

These notes should be observed to achieve optimum functioning of the equipment.

### 1.2 General safety Instructions



The LDT 2000 temperature meter is a device designed for laboratory, indoor use. It is powered from 85 to 250V line, therefore the user should work with respect to all precautions related to operating with mains powered equipment. The mains socket should always have working grounding path!

## 2. PRODUCT DESCRIPTION

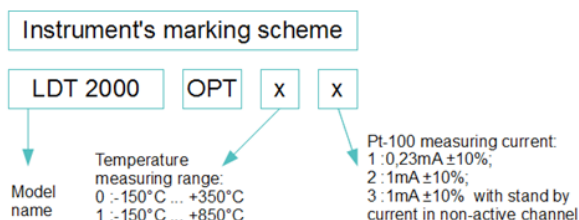
The LDT 2000 is a laboratory- grade precision digital thermometer which uses Pt-100 platinum resistors as temperature sensors. The device provides two measuring channels. The range of measured temperatures is from -150°C to +350°C for option 0x, and from -150°C to +850°C for option 1x (measuring circuit limitation, probe may have its own limitations).

A characteristic feature is high resolution (0,001°C) and high accuracy of temperature measurement ( $\pm 0,007^\circ\text{C}$  with NIST-traceable calibrated probe – on demand). The LDT 2000 digital thermometer uses advanced internal autocalibration circuits that minimize effects of electronic components aging (Vishay ultrastable resistors technology). Thermal EMF induced error is eliminated on the whole path between thermometer and the probe by

the means of Current Reversal technique. The autocalibration sequence is a background task performed continuously. The LDT 2000 has been designed to make operating as simple as possible. An advanced configuration and data acquisition options are available by the means of PC interface and suitable PC software. A large illuminated LCD display is capable of indicating several parameters simultaneously.

## 2.1 Features

- Pt-100 resistors as temperature sensors
- two measurement channels
- high resolution (0,001 °C)
- temperature measurement accuracy up to  $\pm 0.007$  °C (special on-demand version)
- internal auto calibration circuitry
- easy to operate
- enclosure made of durable materials
- Thermometer Utility software in price of the product - configuration, acquisition of measurements
- large display
- calibration data protected by a triple memory write system
- possibility of introduction of user's Callendar- Van Dusen coefficients
- temperature displayed in °C, °F, K
- display of additional parameter: gradient, resistance, temperature difference
- USB and RS-232 interfaces



<b>Temperature measurement range</b>	-150°C ... +350°C (0 ... 230,0000Ω) -OPT0x, 0x, 0x -150°C ... +850°C (0 ... 450,0000Ω) -OPT1x, 1x, 1x
<b>Resolution</b>	0,001°C (1mK), noise $\leq \pm 1$ mK -OPT01, 02, 03 0,001°C (1mK), noise $\leq \pm 3$ mK -OPT11, 12, 13
<b>Accuracy</b>	$\pm 0,007$ °C in the range 0°C ... 100°C ; $\pm 0,00014$ *(T-50°C) beyond (with NIST-traceable calibrated probe on demand) $\pm 0,020$ °C in the range 0°C ... 100°C ; $\pm 0,0004$ *(T-50°C) beyond (with probe calibrated by Leyro Instruments) $\pm 0,080$ °C in the range; $\pm 0,0016$ *(T-50°C) beyond (with uncalibrated probe)

<b>Pt-100 measurement current</b>	LDT 2000 OPT x1, x1: 0,23mA $\pm$ 10%; LDT 2000 OPT x2, x2: 1mA $\pm$ 10%; LDT 2000 OPT x3, x3: 1mA $\pm$ 10% with stand by current in non-active channel
<b>Pt-100 self heating power</b>	LDT 2000 OPT x1, x1 : 5,8 $\mu$ W for 0 °C LDT 2000 OPT x2, x2: 101 $\mu$ W for 0 °C LDT 2000 OPT x3, x3: 101 $\mu$ W for 0 °C
<b>Supply</b>	85-250VAC, 45...65 Hz
<b>Power consumption</b>	8VA max. 4VA typically
<b>Fuses</b>	2x SLOW 1A 250V; 5x20mm
<b>Connectors</b>	<ul style="list-style-type: none"> <li>• Supply 230V – IEC C14</li> <li>• USB B connector – USB interface</li> <li>• DB9 -RS-232 interface</li> </ul>
<b>RS-232 parameters</b>	ABaud rate: 9600; data bits: 8; parity:no; stop bits:1
<b>USB standard</b>	USB 2.0
<b>Communication protocol</b>	SCPI compatible
<b>Temperature range</b>	storage:-10°C...+60 °C; usage: 10°C...40 °C, non-condensing
<b>Dimension</b>	271 x 280 x 102,6 mm
<b>Weight</b>	ca. 1,5 kg

## 3. OPERATION

### 3.1 Safety

The LDT 2000 temperature meter is a device designed for laboratory, indoor use. It is powered from 85 to 250V line, therefore the user should work with respect to all precautions related to operating with mains powered equipment. The mains socket should always have working grounding path!

### 3.2 Preparation to work

1. Connect one or two (as needed) probes to the sockets on the thermometer's front panel. Please keep in mind that each channel has its own calibration coefficients and probe's plug must be placed in proper, signed socket. A mistake will result in wrong temperature readings
2. Connect the instrument to the power line
3. Turn on the instrument by pressing POWER switch
4. Read the messages displayed by the instrument (chapter 3.3)
5. Wait until self calibration sequence reaches 100%
6. The thermometer is now ready to stand- alone operation



Following steps are required only if operation with Thermometer Utility software is required

7. Please install USB driver and Thermometer Utility software prior to the first thermometer connection
8. Connect the thermometer to the PC computer using USB or RS-232 cable
9. Run Thermometer Utility software
10. Press scan button to get serial number of connected instrument
11. Press connect button to communicate the instrument with PC
12. Follow appendix B to take advantage of the Thermometer Utility software

### 3.3 Power-on sequence

After turning on, the instrument displays sequence of three important screens. The whole sequence lasts about 40s to 120s depending on used averaging (N=1 to 10). The screens are presented in order of appearance.

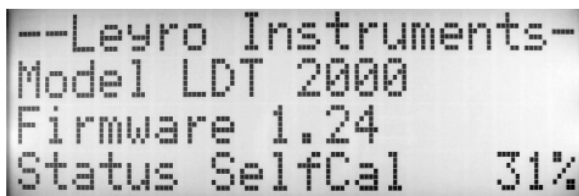
The first screen (fig 1) shows general information about the instrument:

Name of the manufacturer;

Model name;

Firmware version;

Self calibration progress;



--Leyro Instruments--  
Model LDT 2000  
Firmware 1.24  
Status SelfCal 31%

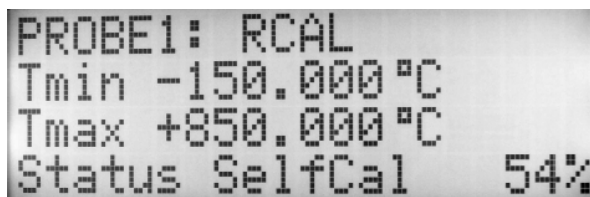
Second screen (fig 2) shows information about probe connected to channel CH1:

Name typed during calibration;

Minimum allowed temperature; Exceeding limit information (not in this example);

Maximum allowed temperature; Exceeding limit information (not in this example);

Self calibration progress;



PROBE1: RCAL  
Tmin -150.000 °C  
Tmax +850.000 °C  
Status SelfCal 54%

Third screen (fig 3) shows information about probe connected to channel CH2:

Name typed during calibration;

Minimum allowed temperature; Exceeding limit information (true in this example);

Maximum allowed temperature; Exceeding limit information (true in this example);

Self calibration progress;

```

PROBE2: 0413
Tmin  -50.000 °C   OVF
Tmax  +150.000 °C  OVF
Status SelfCal     79%

```

When the self calibration counter reaches 100% the thermometer enters its standard measuring mode and is ready to use (fig 4). The blinking “+” sign near T1 means that probe in channel 1 is being measured at the moment.

```

+T1   =   +0.019 °C
R1    =   100.0073 Ω
T2    =   +23.686 °C
R2    =   109.3210 Ω

```

### 3.4 Stand- alone benchtop operation (without PC)

The LDT 2000 has been designed to make operating as simple as possible. Follow steps from chapter 3.2.

### 3.5 Operation with PC and Thermometer Utility software

The Thermometer Utility software was developed for easy intuitive data acquisition from the LDT 2000 thermometer. The software is available for Microsoft Windows Operating Systems.

The software supports:

1. Monitoring measured parameters in time domain using adjustable scale display
2. Data acquisition and recording. Advanced controls allow many options of starting / finishing of recording.
3. Configuring the thermometer for stand-alone operation.
4. Recalibration of resistance measuring circuitry, reading and writing of the coefficient's memory.

The user will find detailed description of all functions and features under in Appendix B Thermometer Utility software.

### 3.6 Operation with PC and custom software

User's custom software or simple Terminal programs can be used to configure/ read data from the thermometer. Appendix C contains detailed description of the communication protocol.

### 3.7 Internal battery

Inside the meter's enclosure there is a battery used for memory backup. The memory stores: internal clock, data, display settings. Estimated lifetime of the battery is 5 years. In case of wrong time or data display, it might be necessary to replace this battery. Lithium battery 3V type CR 2032.

### 3.8 Calibration period

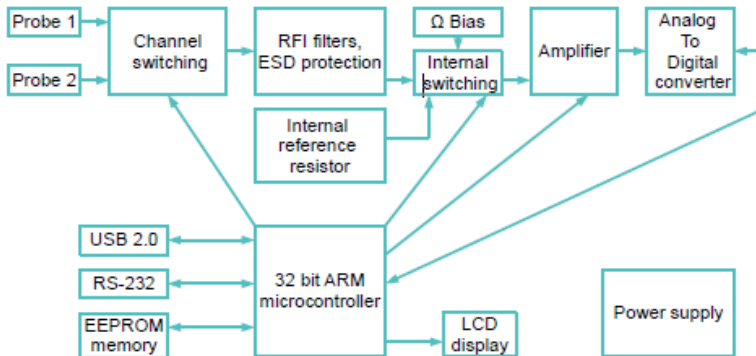
The manufacturer suggest 24 months calibration interval for the LDT 2000 device. The probes should be calibrated:

- every 24 months if the working temperature is  $T < 200\text{ }^{\circ}\text{C}$
- every 12 months if the working temperature is  $T \geq 200\text{ }^{\circ}\text{C}$

## 4. ADVANCED USER INFORMATION

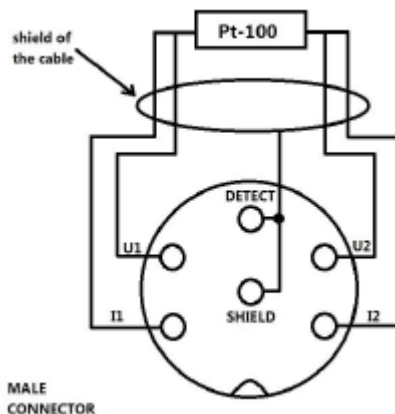
### 4.1 Block diagram

The LDT 2000 thermometer is a microprocessor -based meter which accurately measures probe's resistance and calculates temperature based on the Callendar– Van Dusen equation. Modern high quality components have been used to ensure high precision and reliability. Block diagram of the device is shown on fig 5.



### 4.2 Probes

The thermometer supports two Pt-100 probes. Four terminal Kelvin sensing technique is used to provide highest level of accuracy. The cable of the probe is shielded to reject most of electromagnetic interference that could affect the measurement. Internal wiring of the probe is shown on fig 6.



**Important:** Metal part of the probe is grounded in most types of probes.

The probes are provided with unique Callendar– Van Dusen coefficients.

The device provides two measuring channels. When using two probes the active channel is switched periodically with period of about 1,5s (or longer when averaging of samples is used).

### 4.3 Extender

The probe's cable length may be insufficient in some cases. The user can increase the cable length by using an extender having the maximum length of 8m without losing accuracy. High quality 4 wire, high density shield cable should be used. Teflon (PTFE) insulated cable is preferred. Using long extender in industrial environment with very high electromagnetic interferences level can increase the measurement noise.

### 4.4 Resistance calibration

In case of performing resistance calibration procedure or using the LDT 2000 as a precision ohmmeter the user should follow connection scheme shown on fig 6 and use the Amphenol C091-T3400-001 connector for connecting external measured resistor. It is necessary to short two middle pins of the connector to provide the „probe detect loop“. If the user change accidentally U1 with U2 or I1 with I2 the meter will display negative reading eg. -100.0000  $\Omega$ . This must be corrected prior calibration otherwise an error will occur.



**Important:** During internal resistor calibration procedure, external standard resistor **MUST** be connected to CH1.

## 4.5 Coefficients

The thermometer uses Callendar– Van Dusen coefficients to calculate exact temperature of the probe. Each probe has unique set of coefficients developed while calibration. To achieve proper readings, these unique coefficients should be stored in the thermometer's memory. In order to do so, follow Chapter 5 instructions. If no coefficients are typed, the thermometer uses default values presented in Tab2.

**Tab. 2. Default values of coefficients**

R0	100.0000	
coef:		
coef[0]	3.908e-3	(A)
coef[1]	-5.775e-7	(B)
coef[2]	-4.183e-12	(C)
pcor	0,0,0	
ncor	0,0,0	
tmin	-50.0	
tmax	200.0	

If the resistance value is set to be displayed, the resistance reading is always a raw data – data without any mathematical computing.

## 4.6 PC interface

The meter supports USB 2.0 and RS-232 interfaces. The protocol is compatible with SCPI language. The examples of the most common commands are listed below.

All details connected with SCPI communication are described in appendix B.

A suitable Microsoft Windows USB driver file: „STM\_VCP\_DRIVER.zip” can also be downloaded from the web site listed above.

The most frequently used commands are listed below (this symbol // marks the comment):

*idn?\$00	//identify command, response will be eg. „Leyro Instruments, LDT 2000 OPT02, 0413,1.1”
:syst:date?; time?\$00	//date and hour question
:syst:date 2013,04,04; time 23,22,00\$00	//date and hour setting
:disp:menu:name res\$00	//enabling resistance display
:disp:menu:name grad\$00	//enabling gradient display
:init:cont off; cont?\$00	//enabling sequential results sending
:meas:temp:val? (@1)\$00	//send temperature measured in CH1
:sens:aver:coun 4\$00	//enable averaging of 4 samples
:cal:sec:stat 1,2804\$00;;cal:sec:stat?\$00	//enabling calibration, response: „ON”
:cal:res:val 100.0034\$00	//internal resistor calibration using standard resistor connected to CH1 (value 100.0034 Ω)

## 5. TYPICAL PERFORMANCE OF THE THERMOMETER

**Tab.3. Daily zero drift**

Measurement conditions				
Measurement configuration	Measurement time [h]	Ambient temperature [°C]	Mains voltage [V]	Results
CH1: "four wire zero ohms" Averaging: N=10	24	18...28	218...234	fig 7

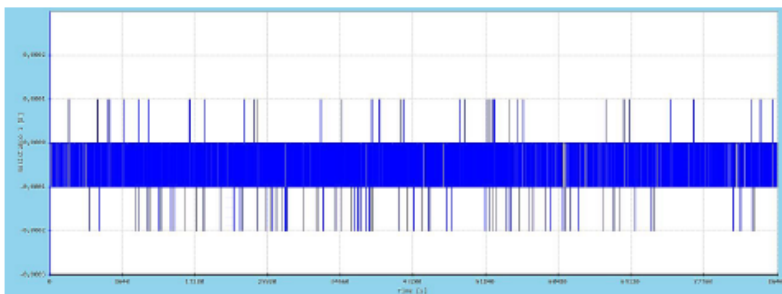


fig 7: Min. = -0,0002Ω; Max. = 0,0001Ω; Full scale reading: 230,0000Ω

**Tab.4. Daily total drift (gain & zero at typical EMI level)**

Measurement conditions				
Measurement configuration	Measurement time [h]	Ambient temperature [°C]	Mains voltage [V]	Results
CH1: ultrastable resistor VISHAY 200Ω 0,01%, 0,05ppm/°C CH2: ultrastable resistor VISHAY 100Ω 0,01%, 0,05ppm/°C Averaging: N=10	24	18...28	218...234	fig 8

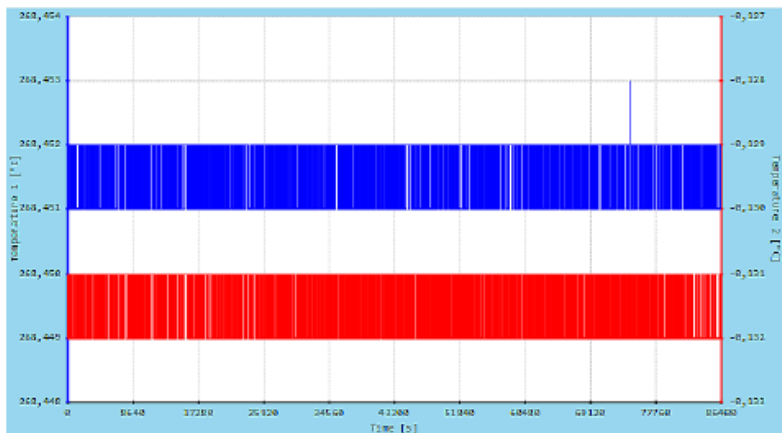
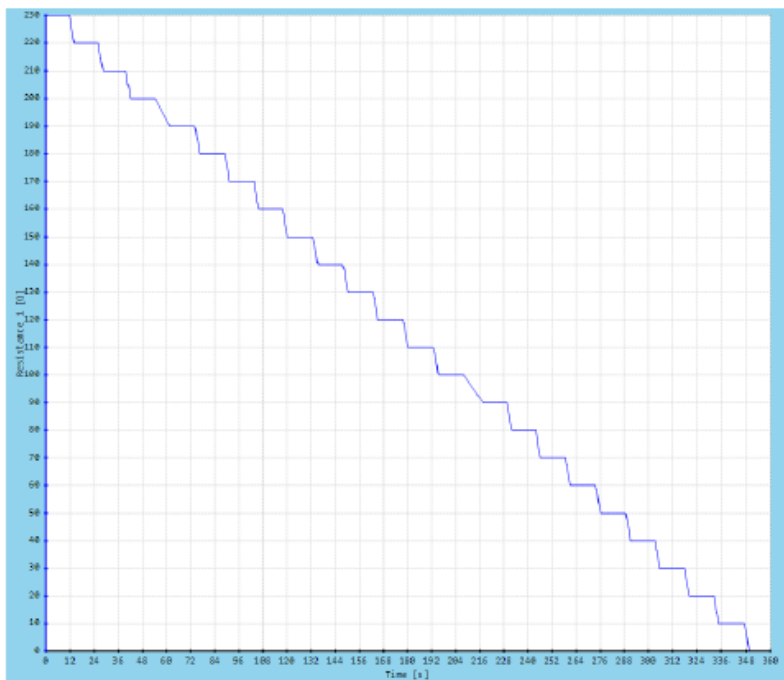


fig 8: Blue trace: Min. = 268,451°C; Max. = 268,453°C; Red trace: Min. = -0,327°C; Max. = -0,351°C

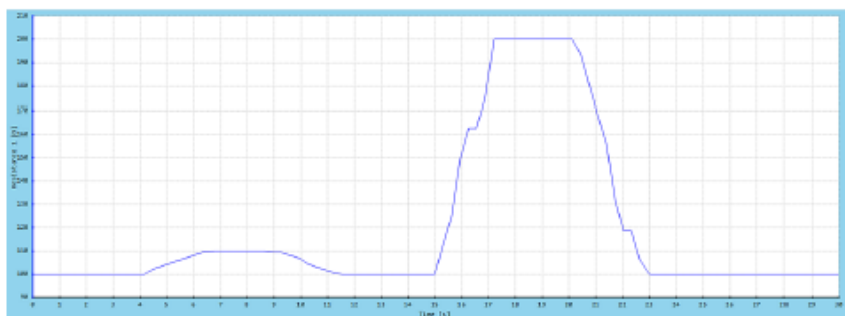
**Tab.5. Check of range coverage 0 ... 230Ω**

Measurement conditions				
Measurement configuration	Measurement time [s]	Ambient temperature [°C]	Mains voltage [V]	Results
CH1: precision decade resistor 0,05%, the resistance was set from 0Ω to 230Ω in 10Ω increments Averaging off (N=1)	360	25,5	231	fig 9

*fig 9: Differences at 90Ω/100Ω and 190Ω/200Ω are effect of decade changing*

**Tab.6. Thermometer's electrical response time**

Measurement conditions				
Measurement configuration	Measurement time [s]	Ambient temperature [°C]	Mains voltage [V]	Results
CH1: precision decade resistor 0,05%, the resistance was changed in sequence: 100Ω-110Ω-100Ω-200Ω-100Ω averaging off (N=1) channel CH2 disconnected	30	25,5	228	fig 10

*fig 10: Measurement time: less than 3s (25°C and 266°C changes were simulated)*



**Tab.7. Effect of measurements averaging**

Measurement conditions				
Measurement configuration	Measurement time [s]	Ambient temperature [°C]	Mains voltage [V]	Results
CH1: "four wire zero ohms" CH2: ultrastable resistor VISHAY 100Ω 0,01%, 0,05ppm/°C Parameter N was changed- number of measurements averaged SCPI command „:sens:aver:coun N\$00"	720	25,5	229	fig 11

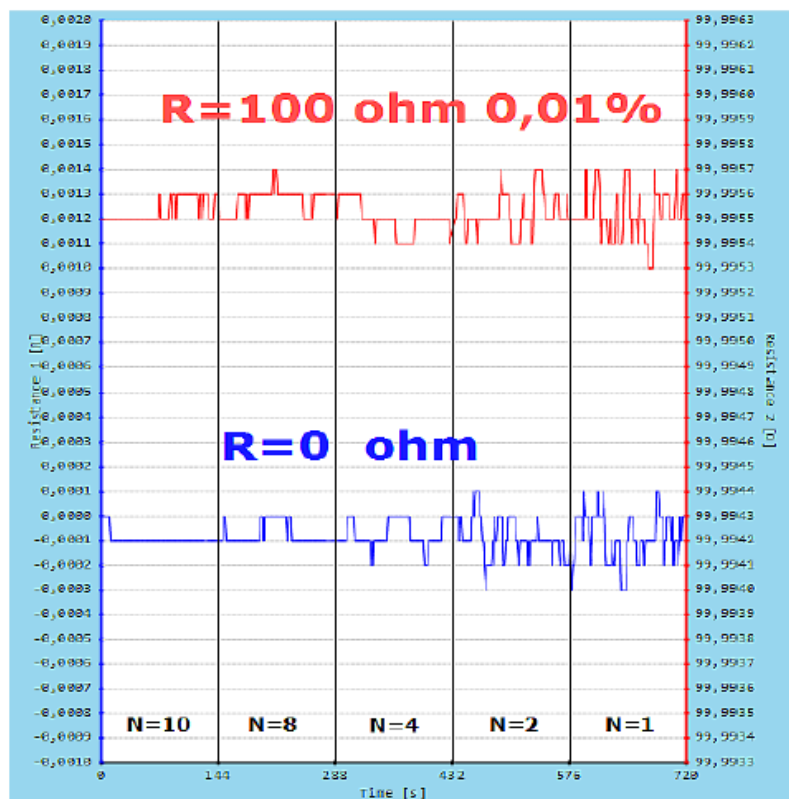


fig 11: Compromise between speed and accuracy must be chosen

Tab.8. Effect of mains voltage change on temperature measurement				
Measurement conditions				
Measurement configuration	Measurement time [s]	Ambient temperature [°C]	Mains voltage [V]	Results
CH1: ultrastable resistor VISHAY 100Ω 0,01%, 0,05ppm/°C averaging off (N=1) channel CH2 disconnected	180	25,7	205 230 255	fig 12

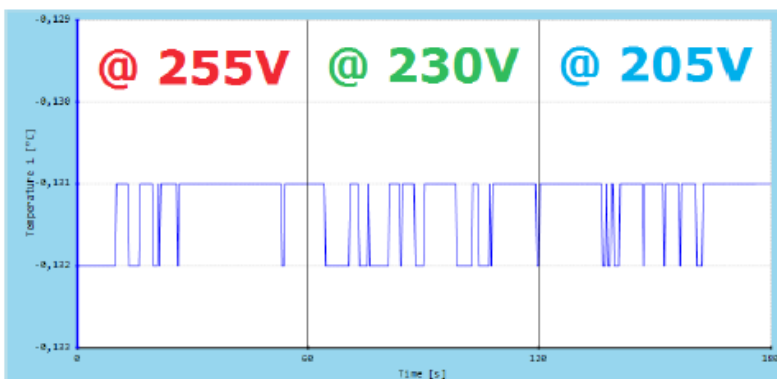


fig 12: Mains voltage changes of  $\pm 11\%$  from nominal 230V does not affect temperature measurement

## 6. THERMOMETER UTILITY SOFTWARE

### 6.1 Program at a glance

Thermometer Utility software allows to use LDT 2000 thermometer with PC computer. It allows to use all benefits of data acquisition and remote configuration of the thermometer. The software can be easily installed on any MS Windows Operating System and requires RS232 (COM) or USB port to connect.

#### 6.1.1 Installation and requirements

To start the installation, run "setup.exe" file and confirm to deploy the program. After quick setup, utility runs for the first time. Later it will be available from desktop shortcut (fig 13).



fig 13: Thermometer Utility icon

Thermometer Utility was developed on .NET Framework and requires at least 2.0 version of this platform to work. Therefore, in case when it is not available on destination computer, it can be installed by running "dotnetfx2.exe" file.

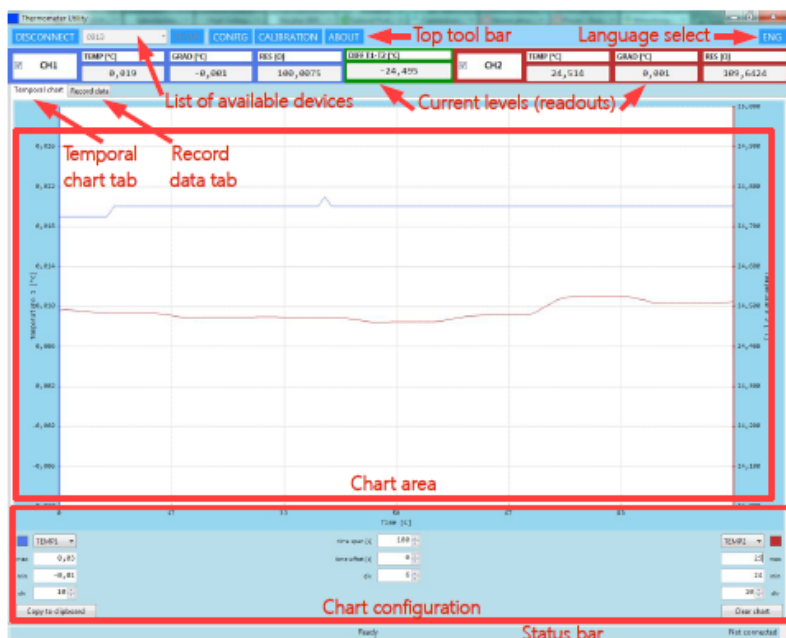
Most up-to-date version of .NET Framework should be downloadable from Microsoft Download Centre on the Internet.

## 6.1.2 Main window overview

Thermometer Utility main window consists of few parts, shown and marked on fig 14. Tool bar on top, allows to select and to connect with chosen thermometer. During startup, the program searches for compatible devices on available serial communication ports. Afterwards, the amount of devices found is displayed on status bar (bottom part of the window). This allows to use several separate thermometers with several Thermometer Utility instances at one PC at one time.

Desired thermometer can be chosen by expanding combo-list next to “CONNECT” button and selecting device’s serial number. After pressing the mentioned button, communication with thermometer will initialize and other functions will activate.

Below the toolbar, resides a panel of currently measured levels: temperature, gradient and resistance (for plugged and used probes: CH1, CH2). These levels can be reviewed in time, on the “Temporal chart” tab, or recorded from “Record data” tab. Both sections of the program are described in separate chapters.

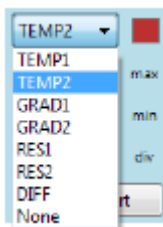


In case when the thermometer was plugged into port during program's work, press “SCAN” to refresh the list of available devices. It is possible however, while the program is disconnected from any device.

## 6.2 Viewing and setting up the chart

Chart component can display function curves of two of possible values at a time:

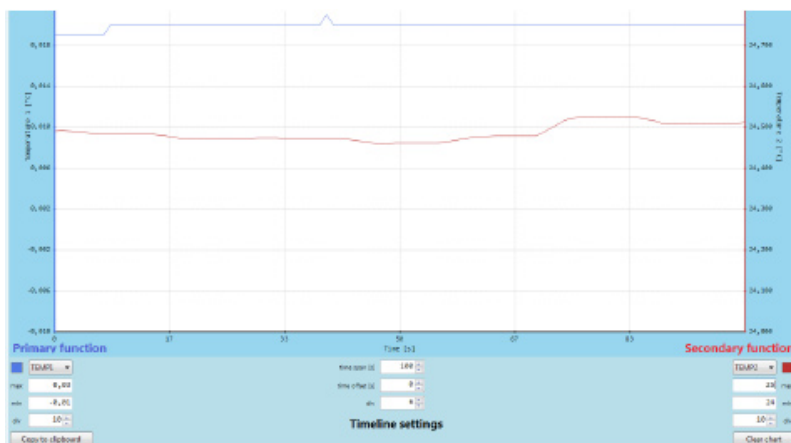
- TEMP1 or TEMP2 – temperature values;
- GRAD1 or GRAD2 – gradients;
- RES1 or RES2 – resistance values.
- DIFF -difference in temperatures T1-T2



As it is shown on fig 15, secondary function can be turned off, by choosing “None” from selection list.

Subsequent measured values are being appended at the end of chart, therefore since the start of collection, it “grows” to the right. The chart can store up to 24 hours of samples, and after that time, the chart stops receiving further data.

The chart however can be reset any time by pressing Clear chart button, located in the lower-right part of the panel.



There are two sets of fields, which relate to primary and secondary function shown on the chart (fig 16). First is an expandable list, where desired value can be selected. Near the list, a square button is available, allowing to change the color of corresponding curve. Clicking this field will show a standard dialog box, where the new color can be chosen.

Below selection combo, there are three fields that determine the scale for each function:

- max – as for maximum value of vertical scale,
- min – as for minimum value of vertical scale,
- div – as for amount of divisions of vertical scale range.

Changing each field takes an immediate effect on chart component. In case of invalid input, a small exclamation icon will appear, showing fault information after holding the mouse cursor over it.

Timeline settings are reduced to specification of time span (maximum of 86400 seconds = 24 hours), and division of time-scale. Besides possibility of changing the width of time window, there is also a chance to scroll its starting position, by modifying the time offset field.

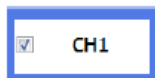
Chart parameters, that have up/down arrows attached, can be edited using two basic ways: mouse-click of an arrow or entering new value from keyboard and confirming with Enter key. But there are also two convenient ways – after selecting such field with mouse, it can be scrolled up/down using respective keyboard up/down keys, or with mouse wheel (if exists).

The chart can be copied to clipboard as a bitmap and used e.g. as a report in a document. This can be done by pressing Copy to clipboard button, located in the lower-left part of the panel.

The thermometer has one measurement circuit that is periodically switched between channel CH1 and CH2. The channel which is not active at the moment returns constantly its last measurement. This has an impact on the presentation of data. The described effect occurs if two probes are connected to the thermometer.

If the user is interested in logging data from only one selected channel and he wants to avoid the described effect, two solutions are proposed:

- 1) physically disconnect the unused probe from the thermometer
- 2) disable the unused channel by de-selecting the check box near the name of unwanted channel as shown on fig 17.



The default condition after running the software is: all attached channels active.

### 6.3 Recording measured data

Following chapter discusses how to record measured samples and store them into a file, which can be used in some external program (datasheet, text editor, etc.). Below, fig 18 shows a partial view of “Record data” tab.

Record		Store samples	Clear
	CH1	CH2	
TEMP [°C]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
GRAD [°C]	<input type="checkbox"/>	<input type="checkbox"/>	
RES [Ω]	<input type="checkbox"/>	<input type="checkbox"/>	
DIFF [°C]	<input type="checkbox"/>		
Schedule			
From	04-29 15:06:45		
To	04-29 15:06:45		
Interval [s]	1		
START			
STOP			

Fecha / hora	TEMP 1 [°C]
04-29 15:07:37	0,020
04-29 15:07:36	0,020
04-29 15:07:35	0,020
04-29 15:07:34	0,020
04-29 15:07:33	0,020
04-29 15:07:32	0,020
04-29 15:07:31	0,020
04-29 15:07:30	0,020
04-29 15:07:29	0,020

As it is shown on the figure, user can select which levels should be recorded, by marking (or deselecting) fields indicating channels (CH1, CH2) and values (TEMP, GRAD, RES). In case when one of probes is not used (not connected), corresponding fields are disabled (grayed).

To have a quick start, user can press START button right away, to view subsequent levels being appended on the right, each one second (by default). Current recording can be stopped, using STOP button.

There are however few more options, which allow scheduling samples collection. First, it is possible to enlarge the time interval between consecutive samples (in seconds, one is a minimum). Second, user can decide to automatically start and/or stop recording process at specified moment of time.

For example, following steps would start recording (each 30 minutes) on 3th of September at 10h30m00s, ending automatically on 3th of September at 22h30m00s:

1. Mark From field.
2. Enter into its corresponding date-time field, a textual value in format MM-dd HH:mm:ss (month-day hours:minutes:seconds), adequate to 3th of September, half past 10 AM. To faster select the date, it is possible to expand a handy calendar, by pressing an icon with arrow, attached to this field.
3. Mark End field.
4. As in p. 2, input date-time, this time adequate to 3th of September, half past 10 PM.
5. Set Interval [s] field to 1800 (30 times 60 seconds).
6. Press START button.

After completing these six steps, above the data view appear an information, that the program is awaiting the time to start. After it passes, samples start appearing on the list.

Until stopping the process, almost all other functions are blocked.

Recording can be started manually or automatically, as well as stopped in one of both ways. During the "automatical stop" mode however, it can be finished instantly, simply by pressing STOP.

Recorded samples can be disposed, using Clear button.

### *6.3.1 Storing and using recorded data*

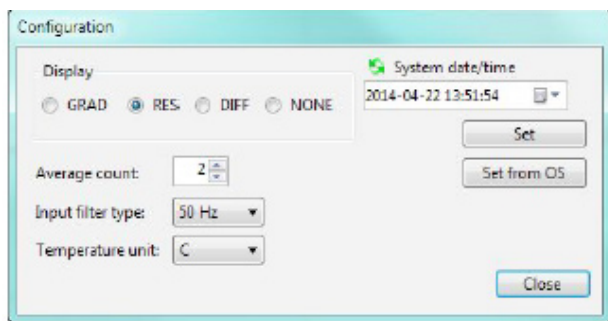
After finishing the process of recording, but also while it lasts, user can store collected samples into a plain-text file, or CSV file. Both types of file save recorded data in similar form, but the second one (CSV) can be opened directly with such programs as OpenOffice Calc or MS Excel. Following steps illustrate how to store and open the recorded samples in Calc program.

1. Press Store samples button, located above the list of recorded samples.
2. Input a name and choose location for newly created file. Select "CSV file" type.
3. If OpenOffice Calc is installed and set as default program for opening CSV files, double-click the icon of stored file (if not, right-click the file and choose Open with →

- OpenOffice Calc from context menu).
4. In "Text import" dialog window of Calc, select "Unicode (UTF-8)" as used character set and mark "Comma" as the field separator only. Press OK to continue.
  5. Format data cells corresponding to their type (numbers as Number type, date-time fields using suitable date-time format).

## 6.4 Configuration

Simple thermometer configuration is available under CONFIG button, located in the top toolbar of the program. After pressing this button however, samples capturing is held until leaving configuration window.



The above fig 19 shows options, explained in subsequent paragraphs:

- Display – here user can switch between several functions, deciding which of these should be presented on thermometer's LCD display below temperature reading. The functions are described below:

GRADient – the temperature gradient per second  
 DIFFerence – the difference between the temperatures of probes in the first and second channel,  
 RESistance – the resistance of the probe's sensor,  
 NONE – additional results are not displayed (for clearance).

- Average count – sets from 1 to 10 the amount of samples that would be averaged to calculate current level). Averaging allows to decrease noise by a factor of  $\sqrt{n}$  however it results in increased time of measurement
- Input filter type - The field changes sampling frequency of internal analog-to-digital converter in order to get better 50Hz or 60Hz line frequency rejection. The user should set this parameter according to local mains frequency
- Temperature unit - The field is used for setting the temperature units used for displaying and calculations. Three basic temperature scales have been implemented into the thermometer: Celsius, Kelvin, Fahrenheit
- System date/time – by pressing the green "refresh" icon, user can retrieve the current system date and time from the thermometer. Make sure that date is correct before attempting to calibrate thermometer or probe.
- Set and Set from OS – these buttons allow to send a new system datetime to the thermometer (first – from the date-time control in window, and the second one – from operating system's clock).

## 6.5 Calibration

More advanced features are provided in Calibration window, accessible through its corresponding button on the top toolbar. After entering, user can see a similar window to this in fig 20.

The Calibration window contains the following elements:

- Analog circuit number:** 091301120
- Standard resistor value [Ω]:** 100,0000
- Buttons:** Unlock..., Calibrate, Clear memory, Calibrate probes, Close
- Probes Table:**

		S/N	Last calibr.	R0	Coeffs.	NInt	NCor	PCor	Tmin	Tmax	Tmin overflow	Tmax overflow					
► #1	Export	Import	RCAL	2014-04-22	100	0,003908	-5,775E-07	-4,183E-12	0	0	0	0	0	-150	850	<input type="checkbox"/>	<input type="checkbox"/>
#2	Export	Import	0413	2014-04-22	100,0845	0,00591211	-6,71229E-07	-1,10175E-09	0	0	0	0	0	-50	150	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

In case if only one probe is attached to the thermometer, program will display a message and present only the parameters of connected probe.

Calibration window presents following information:

- Analogue circuit number – serial number of thermometer's measurement board;
- Standard resistor value (in Ohms) – to use during a resistance calibration;
- Probes – displays the serial number, last calibration date, resistance at 0°C, calibration and correction coefficients, the temperature range and overflow flags. Below each column is described:
  - S/N - probe's name or serial number (10 characters)
  - Last calibr. - indicates last calibration date for probe. This field is filled automatically during calibration.
  - R0 - probe's resistance at 0 °C,
  - Coeffs. <A, B, C> - coefficients for Callendar– Van Dusen equation for calculating resistance of Pt100 sensor,
  - NInt – empty field left for compatibility with older firmware versions
  - NCOR <a0, a1, a2> - (Negative CORrection) – coefficients of 2nd grade polynomial used for correction of negative temperature measurements.  
The polynomial can be used to fit thermometer's reading to the value required by the user in following maneer:  

$$RAC = a2 \cdot (RBC^2) + a1 \cdot RBC + a0$$
 where:  
 RAC =reading after correction  
 RBC =reading before correction
  - PCOR <a0, a1, a2> - (Positive CORrection) – coefficients of 2nd grade polynomial used for correction of positive temperature measurements.  
The polynomial can be used to fit thermometer's reading to the value required by the user in following maneer:  

$$RAC = a2 \cdot (RBC^2) + a1 \cdot RBC + a0$$
 where:  
 RAC =reading after correction  
 RBC =reading before correction
  - TMIN - minimum working temperature for probe,



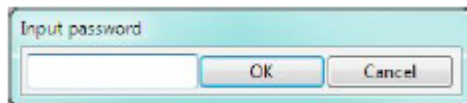
- TMAX - maximum working temperature for probe,
- Tmin overflow , Tmax overflow - the fields are used to confirm that the probe maintains it is metrological properties and allow to check if the probes was used within declared temperature range. If any time during lifespan the particular temperature limit has been exceeded, the thermometer sends back OVF condition is set if within the following 20 measurements of temperature, over 10 measurements exceeds the limit.  
Cancelling overflow flags can only be done by clearing the memory corresponding to the probe. It is good practice to store probe's coefficients to a file before deleting them.

Some columns of Probe list are read-only: Last calibr. ; Nint; Tmin overflow; Tmax Overflow. Other fields can be edited, according to suitable data format (explained in more details in appendix B).

Probe parameters can be exported and imported, using the buttons attached to each row of the table. It is recommended to export certain probe's data to a file for troubleshooting.

The Calibration tab allows to modify probe's memory content and to calibrate internal reference resistor. This features are extremely important for thermometer's accuracy and they should not be changed by non authorized person.

All calibration features are disabled until providing a supervisor password.  
After pressing Unlock button the "input password" window appears as shown on fig 21.



The password is "2804" (only numbers).

The Clear Memory button allows to delete memory content of measuring circuit board, calibration data for channel 1 and calibration data for channel 2. This button is placed only for use during production and should not be used.

## 6.6 USB Drivers

To download drivers supporting USB connection, please type the following phrase into browser's window:

stm\_vcp\_driver.zip

Uncompress downloaded file and install the driver according to manufacturer's instructions.  
The driver should be installed before connecting the thermometer to the PC.

## 7. COMMUNICATION PROTOCOL

for firmware versions 1.23 ... 1.24

### 7.1 Communication interfaces

Communication with the thermometer can be implemented using RS232 (baud rate = 9600, data bits = 8, parity = none, stop bits = one, handshaking = none) or USB 2.0 interface. USB protocol works as a virtual COM port, and uses the STM Virtual COM Port drivers.

### 7.2 SCPI protocol

An IEEE-488.2-based protocol is used to communicate with the thermometer, implementing SCPI commands.

#### 7.2.1 Syntax

Commands consist of ASCII characters, mostly letters and digits, but there are also some special purpose characters. More details further in this document.

#### 7.2.2 Commands

Commands are individual words, which describe the purpose of performed function. It is possible to send full words, as well as their mnemonics (up to 4 characters long, abbreviating its full command, e.g. MEASURE and MEAS). Both, commands and their mnemonics are unique.

The device should properly respond only on mnemonic or its corresponding full word. Therefore MEAS, MEASURE, but also MEASUR or MEASURE1 will execute the same command. Using mnemonics is recommended, since not all full names are properly recognized.

Character case in command is ignored, but this document will tend to use upper-case for mnemonic, and the rest of full word with lower-case (e.g. MEASure).

#### 7.2.3 Command termination character

Standard program message terminator used, is the new line character (<nl> - 0x0A). Thermometer recognizes any character lower than space (0x00 – 0x1F) as a message terminator. End of thermometer's response is signed with 0x0D, 0x0A (carriage return and new line). It is possible to send a sequence of commands in a single message, separating them by semicolon. In such case, thermometer also responds with a sequence, separated with commas, too. Input and output buffers are 250 characters long.

#### 7.2.4 Structure of commands

Set of commands has a tree structure. It composes of main commands' catalogues and their sub-commands. To access specific functions, a full path of commands must be provided, where subsequent tree-levels are separated with colon. Having such exemplary tree of catalogues:

```
:MEASure
      :TEMPerature
            :VALue
            :GRADient
            :RESistance
```

to access the value of temperature, following command must be sent:

:MEAS:TEMP:VAL?

While decoding a sequence of commands, the thermometer stores the catalogue, where it currently points. Therefore in case of the example above, if desiring to access the temperature gradient also, it can be achieved in two ways:

1) :MEAS:TEMP:VAL?; GRAD?

2) :MEAS:TEMP:VAL?; MEAS:TEMP:GRAD?

After running :MEAS:TEMP:VAL? command, we're residing already in TEMP catalogue, therefore we can directly use another command from that catalogue (example 1). However, we can also provide the full path again (example 2).

Returning to main catalogue, requires starting the sequence of commands with colon. The message termination character also causes returning to the main catalogue (and so the colon can be omitted at the beginning of each message).

Calling to a command, which doesn't reside in provided catalogue, or using an incomplete path, will cause an error:

1) MEAS:TEMP:VAL?; TEMP:GRAD?

2) MEAS:TEMP:VAL?; MEAS:TEMP:GRAD?

3) MEAS:TEMP:VAL?; :MEAS:GRAD?

### 7.2.5 Query

Appending a question mark after a command, means the thermometer should provide proper response. For example:

:MEAS:TEMP:VAL?

means the device should send back the temperature value. If no response comes back, it means an error had occurred.

### 7.2.6 Parameters

Both plain commands, as well as queries, can accept parameters. These can be numbers, characters or text. Amount of command's parameters is unlimited and depends on programmer's invention. A parameter should be separated from command with a space character. Each successive parameter should be separated then with commas.

### 7.2.7 Special commands

These type of commands, starts with asterisk '\*'. They have defined purposes, which are common to each SCPI-based device. Their functioning is independent from current position in command-tree.

### 7.2.8 Special characters

There are some special meaning characters, besides letters and digits in commands. Part of them was already described in previous paragraphs.

- 0x00 – 0x1F (white characters) – indicate termination of input/output message;
- ';' (semicolon) – separates subsequent commands in single message;
- ':' (colon) – separates each of mnemonics in provided command; starting a command with this character, means returning to main catalogue;
- ' ' (space) – separates command and its parameters; it is forbidden in command's contents, but is ignored in other places;
- ',' (comma) – splits the list of command's parameters;
- '.' (dot) – decimal separator in rational numbers;
- '\*' (asterisk) – put before, indicates a special command.

### 7.2.9 Commands' execution

There are two possible methods of executing commands in the standard: holding and non-holding. In case of the first one, a command is executed until the previous one finishes. In case of non-holding option, commands are performed in parallel. Thermometer runs commands in holding mode, which means in some cases it is required to wait over a dozen of seconds before achieving a response from device.

### 7.2.10 Default commands

To shorten the contents of messages to send, in some cases there were so called "default commands" defined. These ones are put into square brackets, and their contents can be omitted while calling the command, not causing changes in executed function. Only one command can be default in specific catalogue. As an example, in :MEAS catalogue, TEMP and VAL are default commands:

```
:MEASure  
      [:TEMPerature]  
      [:VALue]  
      :GRADient  
      :RESistance
```

It means that temperature value can be read in three ways:

- 1) :MEAS:TEMP:VAL?
- 2) :MEAS:TEMP?
- 3) :MEAS?

The thermometer doesn't enter command catalogues by default, therefore to enter any catalogue, its name must appear in message. Per example then, to read temperature's value, gradient and resistance, it can be commonly achieved by sending:

```
:MEAS:TEMP:VAL?; GRAD?; RES?
```

To shorten the message however, default commands can be used:

```
:MEAS:TEMP?; GRAD?; RES?  
:MEAS?; TEMP:GRAD?; RES?
```

But following message will result as an error:

```
:MEAS?; GRAD?; RES?
```

since it doesn't include entrance to TEMPerature catalogue. Temperature value will be read properly, and then the error will occur.

## 7.3 Status registers

IEEE-488.2 and SCPI standards implement quite expanded status system. Contents of each of registers is described below. Event's occurrence is indicated by setting specific bit to 1.

### 7.3.1 STB status register

It is the most important status register, containing eight bits, which store information about device's state. While using GPIB main line, setting RQS bit equals to interrupt notification. Reading the value of this register causes it to reset. The meaning of its bits:

- 0 and 1 bit – defined dependently on the type of device. In thermometer, both are 0,
- 2 bit ERROR – indicates the occurrence of errors in error queue; its value is 1 when there are errors, 0 if the queue is empty;

- 3 bit QUES – logical sum of reliability register (QUES);
- 4 bit MAV (Message Available) – value equal to 1 if transmission buffer contains messages; it is always 0 in thermometer, since it responds instantly;
- 5 bit ESB – logical sum of events register (ESR);
- 6 bit RQS – logical sum of STB status register;
- 7 bit OPER – logical sum of operation register (OPER).

### 7.3.2 Events register (ESR)

Eighth-bit register, which stores some of the events occurring in the device. When an event happens, bits are set, but not reset after it passes. Reading contents of the register resets it however. Register's bits have following meaning:

- 0 bit OPC – if the bit is set, it means all tasks were accomplished; is set only after calling \*OPC command;
- 1 bit RQC (Request Control) – means the device tries to take over transmission control, always 0;
- 2 bit QYE (Query Error) – user tries to read empty or overflowed output buffer, always 0;
- 3 bit DDE – bit set means that device error occurred;
- 4 bit EXE – bit set means that command execution error occurred;
- 5 bit CME – bit set means command's syntax was incorrect;
- 6 bit URQ (User Request) – always 0;
- 7 bit PON (Power On) – power failure, always 0.

### 7.3.3 Operation register (OPER)

Sixteen-bit register of currently executed operations. Specific bit is set while an operation continues and is zeroed after it passes. Only three bits are used:

- 0 bit CALibrating – set during the calibration;
- 1 bit SETTling – awaiting signals' stabilization, set during collection of full samples' buffer, needed to calculate temperature level after changing the channel or exiting the calibration process;
- 4 bit MEASuring – set during measurement.

Other bits of this register have always zero value.

### 7.3.4 Reliability register (QUES)

Individual bits of this 16-bit register are set, if for some reasons the measurement result is uncertain. After finding the cause of such uncertainty, its bits are reset. Only two bits are used:

- 4 bit TEMPerature – set while collecting the full buffer of samples, needed for temperature's calculation;
- 8 bit CALibration – set if calibration of the thermometer (or one of probes) could not be done.

Other bits have always zero value.

### 7.3.5 Errors queue

The device implements FIFO queue for storing errors that occur. Reading an error removes it from the queue, which can be up to 10 items long. After the queue is filled, an overflow error is added to it. Description of errors that can be reported by the thermometer, is contained in separate chapter.

### 7.3.6 Mask registers

Each of status registers has a mask register assigned (ENABLE). This register is a multiplier for its relative register's value, before its summation. It means, that if value of a specific bit in status register equals 1, and its related mask's bit is also 1, then this bit enters as 1 to logical summation. In other cases it will have value of 0. After starting the thermometer, all bits of mask registers are 0. Reading a mask register doesn't affect its value.

## 7.4 Commands' parameters

### 7.4.1 Binary <boolean>

Used for switching a function on/off. Allowed values are "0" or "OFF" for switching off, and "1" or "ON" for switching on.

### 7.4.2 Integer number <int>

Signed integer number, with maximum of 9 characters.

### 7.4.3 Floating-point number <double>

Floating-point number, where dot (.) is the decimal separator. Such number can be formatted in exponential form (1.00E+00), accepting small "e", as well as the large one. Maximum amount of characters for both integer and fractional part is 9.

### 7.4.4 String

Text string of any characters, except the special ones. Maximum length is 15 characters.

### 7.4.5 Channels' list

Used for setting the amount and numbers of measured channels. It implements somewhat specific format. It is wrapped inside brackets, and the first character is '@', after which the numbers of channels are listed:

- (@1) – first channel measurement;
- (@2) – second channel measurement.

It is also possible to insert more channels in one list, separating them with comma or colon (the second option means a range of channels). Thermometer implements both options:

- (@1,2) or (@1:2) – double-channel measurement.

The standard also defines, that the device should interpret channels order properly. So (@2,1) should cause the second channel to be received on first position. Thermometer recognizes channels' numbers correctly, but will always respond in 1,2 order. Such inconsistency with SCPI standard will likely be fixed in future releases of the device.

## 7.5 Special commands

LDT 2000 thermometer implements all special commands required by the standard:

- \*CLS – resets all status registers: STB, ESR, OPER, QUES, as well as errors query; it does not however influence registers' masks;
- \*ESE<int> - sets the mask of events register (ESR) to decimal value passed as parameter;

- \*ESE? - reads the value of events register mask (ESR);
- \*ESR? - reads the value of events register (ESR); afterwards, the register is reset;
- \*IDN? - device identification; thermometer sends back an answer containing four fields separated with commas: manufacturer's name, device model, serial number and firmware version;
- \*OPC – termination of current operation, sets OPC bit in ESR register;
- \*OPC? - queries the termination of current operation, sends 1 after finishing all commands before receiving the query;
- \*RST – device reset; thermometer is set to state as it was after switching the power on. It is single channel measurement from first probe (or second, if the first is not present). Erasure of sent results and thermometer's triggers. Turns of result averaging. The command doesn't change values of status registers, registers' masks, nor errors query.
- \*SRE<int> - sets the mask of status register (STB) to decimal value passed as parameter;
- \*SRE? - reads the value of status register mask (STB);
- \*STB? - reads the value of status register (STB); afterwards, the register is reset;
- \*TST? - device auto-test; returns 0 if successful, 1 otherwise;
- \*WAI – awaiting operation completion (no effect occurs).

## 7.6 Thermometer commands catalogue

### 7.6.1 Tree of commands

This chapter contains the full catalogue of thermometer's commands. Individual commands are described in subsequent sections. Commands terminated with question mark are queries only. Others have two forms: parameters setting or requesting their current value.

```

:INITiate
    [:IMMediate]
    :CONTinuous <boolean>
:CONFigure
    [:TEMPerature]
        [:VALue] <channel list>
        :GRADient <channel list>
        :DIFFerence <channel list>
        :RESistance <channel list>
:CONFigure?
:FETCH?
    [:TEMPerature]?
        [:VALue]? <channel list>
        :GRADient? <channel list>
        :RESistance? <channel list>
        :DIFFerence? <channel list>
:READ?
    [:TEMPerature]?
        [:VALue]? <channel list>
        :GRADient? <channel list>
        :DIFFerence? <channel list>
        :RESistance? <channel list>
:MEASure?
    [:TEMPerature]?

```

```

: [VALue]? <channel list>
: GRADient? <channel list>
: DIFFerence? <channel list>
: RESistance? <channel list>

:UNIT
:TEMPerature <unit>

:INPut
:FILTer
:NOTCh <f value>

[:SENSe]
:AVERage
:COUNT <cycles>
:OVERflow
:CH1 (CHannel1)
:TMIN?
:TMAX?
:CH2 (CHannel2)
:TMIN?
:TMAX?

:CALibration
[:ALL]
[:ALL]?
:SECure
[:STATe] <boolean, [password]>
:RESistance
:VALue <R value>
:VALue? <R value>
:DATE?
:CH1 (CHannel1)
:R0 <R0 value>
:COEFFicient <A, B, C>
:PCORrection <a0, a1, a2>
:NCORrection <a0, a1, a2>
:TMIN <Tmin value>
:TMAX <Tmax value>
:DATE?
:IDN?
:SNUMber <string>
:CH2 (CHannel2)
:R0 <R0 value>
:COEFFicient <A, B, C>
:PCORrection <a0, a1, a2>
:NCORrection <a0, a1, a2>
:TMIN <Tmin value>
:TMAX <Tmax value>
:DATE?
:IDN?
:SNUMber <string>

:MEMory
:CLEar

[:NAME] <name>

:DISPlay

```



```

:MENU
    [:NAME] <name>
:STATus
    :OPERation
        [:EVENT]?
        :ENABLE <int>
    :QUESTionable
        [:EVENT]?
        :ENABLE <int>
    :PRESet
:SYSTem
    :ERRor
        [:NEXT]?
    :DATE <year, month, day>
    :TIME <hour, minute, second>

```

### 7.6.2 Measurement commands

These commands are used to perform measurements. They're grouped into following catalogues:

```

:INITiate
:CONFigure
:FETCh?
:READ?
:MEASure?

```

where at the same time, the contents of :CONF, :FETCh?, :READ? and :MEAS? catalogues must be identical.

#### 7.6.2.1 :INITiate

These commands are used to trigger the measurement.

:IMMEDIATE – sets the thermometer internal triggering source and storing the measurements in its memory (implemented for compatibility issues).

:CONTinuous <boolean> - switches on/off the continuous sending of measured values. To initiate sending, certain query must be sent also, to precise what should be sent (FETC?, :READ and :MEAS commands). Results are sent with ¼ second interval.

#### 7.6.2.2 :CONFigure

Command used for configuration of the thermometer – setting the channel measured and requested result. The type of result is chosen with one of the commands - :VALue, :GRA-Dient, :RESistance, DIFFerence. Channel is chosen through <channel list> parameter.

There are three modes of measurement possible: first channel, second channel or double-channel. If no channel is specified, values for first channel are sent by default. In case when the command is unsuccessful, an error is stored into queue. Configuration can be shortened thanks to default commands and parameters. Exemplary setting to measure temperature value on first channel can be done using one of the following commands:

- 1) :CONF:TEMP:VAL (@1)
- 2) :CONF:TEMP:VAL
- 3) :CONF:TEMP
- 4) :CONF

To measure temperature on second channel:

- 1) :CONF:TEMP:VAL (@2)
- 2) :CONF:TEMP (@2)
- 3) :CONF (@2)

Measurement of gradient or resistance, requires providing full path of commands. To measure gradient on first channel, configure using one of following commands:

- 1) :CONF:TEMP:GRAD (@1)
- 2) :CONF:TEMP:GRAD.

Measurement of Pt100 probe's resistance for both channels can be setup using:

:CONF:TEMP:RES (@1,2)

Measurement of temperature difference between channels 2 and 1 can be setup using:

:CONF:TEMP:DIFF (@2,1)

#### 7.6.2.3 :CONFigure?

This command can be used to read thermometer's current configuration. It returns a string, formatted with the same structure, as the one used for configuration. For example, if thermometer is configured to read temperature value of first channel, returned string will be:

TEMP:VAL (@1)

and for resistance measurement on second channel:

TEMP:RES (@2).

Sending this command after connecting to thermometer, allows orientation on which probes are connected to the device.

#### 7.6.2.4 Result readout commands :FETCh? and :READ?

These commands read the result of currently configured measurement. :FETCh? reads the last result stored in thermometer's memory, :READ? starts the measurement and then sends back the result. :READ? can be treated as a sequence of following commands:

- 1) :INIT
- 2) :FETCh?

Both commands need proper configuration of measurement first. It can be set using :CONFigure or :MEASure command. For FETCh? command, exemplary valid execution can consist of following sequence:

- 1) :CONF:TEMP:VAL (@1)
- 2) :INIT
- 3) FETCh?

For :READ? command:

- 1) :CONF:TEMP:VAL (@1)
- 2) :READ?

Commands above must not be called directly one after another. They just need to be called between last reset of the device, and using the read commands. In such way, no error is notified.

Calling read commands without any parameters or sub-commands, results in sending values according to last configuration. It is also possible to read another parameters of the same measurement. While measuring the temperature of certain channel, probe's gradient and resistance can also be read. During double-channel measurement, each parameter's value can be achieved.

Read command must be called in order to do that, including specific sub-commands and list of channels, similarly as in :CONFigure command. While tempting to read an unmeasured value, error is notified.

Measurement result is sent back after filling the sample buffers, which can take up maximum of 25 seconds. Because non-holding mode of command execution is used, during this time the thermometer won't react to any other commands.

#### 7.6.2.5 :MEASure?

This command is used for configuration and execution of measurement, including returning of the result. It can be treated as equal to sending:

1) :CONFigure:...

2) :READ?

commands.

Method of measurement configuration is identical as in case of :CONFigure command.

#### 7.6.3 :UNIT – measurement's units handling

The command is used for setting and checking the temperature units used for displaying and calculations. Using :TEMP sub-command actual temperature unit can be set. Three basic temperature scales have been implemented into the thermometer: Celsius, Kelvin, Fahrenheit. The thermometer can display results and receive data from the user in units that are common with actual chosen scale. To switch between the scales mnemonic of desired scale should be send as a command's parameter:

- C or CEL – Celsius scale,
- K – Kelvin scale,
- F or FAR – Fahrenheit scale.

For example, setting Kelvin scale can go as follows:

:UNIT:TEMP K

Sending a query for this command will result in sending name of a presently used scale.

After switching off, the thermometer stores scale that had been used recently.

#### 7.6.4 :INPut – measurement circuit configuration

The file contains only one command used for changing between 50Hz and 60Hz line frequency rejection.

##### 7.6.4.1 :FILTer – configuration of line frequency filter

The command changes sampling frequency of internal analog-to-digital converter in order to get better 50Hz or 60Hz line frequency rejection. The subcommand :NOTCh is used to type proper frequency. In case of typing values other than 50 or 60 the thermometer communicates an error.

For example, setting rejection frequency of 50 Hz can go as follows:

:INP:FILT:NOTC 50

After switching off the thermometer stores rejection frequency that had been used recently.

#### 7.6.5 :SENSe – configuration of measurement's parameters

This catalogue contains a set of commands used for setting up the measurement. Only one command however is implemented by thermometer.

##### 7.6.5.1 :AVERAge – measurement averaging

Following command is used for setting the amount of averaged measurement cycles, which

helps in gaining more stable result, for longer period of measurement. 1 to 10 cycles can be averaged, where 1 means that no averaging is done at all. :AVERage command contains one sub-command :COUNT, used for specifying the amount of averaged cycles. It was added to provide more compatibility with SCPI standard. Since :SENSe is a default command, setting up the averaging for e.g. 5 cycles, can be done in one of two ways:

1) :SENS:AVER:COUN 5

2) :AVER:COUN 5

Currently set averaging count is stored after switching off the thermometer also.

#### *7.6.5.2 :OVERflow –probe's temperature range exceeding flag*

The command is used to confirm that the probe maintains its metrological properties and allows to check if the probes was used within declared temperature range. Following sub-commands can be sent:

- :CH1:TMIN? – checking exceeding low limit in channel 1
- :CH1:TMAX? – checking exceeding high limit in channel 1
- :CH2:TMIN? – checking exceeding low limit in channel 2
- :CH2:TMAX? – checking exceeding high limit in channel 2

If any time during lifespan the particular temperature limit has been exceeded, the thermometer sends back 1. If the probe was used within specified limit the thermometer sends back 0.

Since :SENSe is a default command, checking exceeding of limit, can be done in one of two ways:

1) :SENS:OVER:CH1:TMIN?

2) :OVER:CH1:TMIN?

Over-range condition is set if within the following 20 measurements of temperature, over 10 measurements exceeds the limit. Over-range condition is immediately stored in the EEPROM of the probe and is never reset. In connection with this command :OVERflow is used to validate the correctness of operation of the probe throughout its lifespan. Cancelling overflow flags can only be done by clearing the memory corresponding to the probe. It is good practice to store probe's coefficients to a file before deleting them.

#### *7.6.6 :CALibration – calibration commands*

This catalogue is used for performing all actions related to thermometer calibration.

##### *7.6.6.1 :ALL – autocalibration*

Following command initializes internal calibration process. Calling it isn't necessary, since the device calibrates on each startup, and during its work, updates calibration data in background.

Sending :ALL? query has the same effect, but additionally it sends the result of calibration, after it passes – zero if it went properly, or error number in case of failure. After sending this query, the device waits for about 20 seconds to finish the procedures, and doesn't react to other commands during that time.

Using :ALL command doesn't hang communication with the thermometer, and procedure's result can be read from OPER register.

#### 7.6.6.2 :SECure – calibration unlocking

This command secures access to calibration functions, to prevent incidental overwriting of settings, by sending one of calibration commands. To unlock calibration procedures, security mode must be turned off, by sending ASCII string "2804" as parameter of the command:

1) :CAL:SEC:STAT ON, 2804

2) :CAL:SEC 1, 2804

After performing calibration, secure mode can be turned off:

1) :CAL:SEC:STAT OFF

2) :CAL:SEC 0

Password isn't required to turn off, but may be used anyway. Secure mode is automatically active after restarting the device.

Entering secure mode is necessary to perform resistance and probes calibration. Performing auto-calibration however, or reading calibration coefficients, is possible at any time, without secure mode.

#### 7.6.6.3 :RESistance – resistance measurement calibration

:RES catalogue is used for resistance measurement calibration. Before performing it, model resistor of known resistance must be plugged into Channel 1 (important!). After that, following command can be sent:

:CAL:RES:VAL <R value>

where "value" is the resistance of connected standard resistor.

Sending the same command as a query:

:CAL:RES:VAL? <R value>

will prompt the device to return calibration's state: zero is successful, or error number in case of failure. Because the calibration itself lasts for about one minute, such time must be taken into account and program should wait. Sending the command as a query, will hang communication for that amount of time.

Last calibration date can be read by using :DATE? query:

:CAL:RES:DATE?

#### 7.6.6.4 :CH1 (CHannel 1) and :CH2 (CHannel 2) – probes' calibration

Commands available in these catalogues allow to store parameters of probes. Following commands include into that catalogue:

- :R0 <value> - probe's resistance at 0 °C,
- COEfficient <A, B, C> - coefficients for Callendar- Van Dusen equation for calculating resistance of Pt100 sensor,
- :NCORrection <a0, a1, a2> - (Negative CORrection) – coefficients of 2nd grade polynomial used for correction of negative temperature measurements,
- :PCORrection <a0, a1, a2> - (Positive CORrection) – coefficients of 2nd grade polynomial used for correction of positive temperature measurements,
- :TMIN <Tmin value> - minimum working temperature for probe,
- :TMAX <Tmax value> - maximum working temperature for probe,
- :DATE? - reads last calibration date for probe,
- :IDN? - reads probe's serial number.
- :SNUMber <string> (Serial Number) – probe's name or serial number (10 characters)

### *7.6.7 :MEMory – internal EEPROM memory handling*

This directory is used to operate the EEPROM memories located on the analogue board and the EEPROMs inside the thermometer probes. In firmware version 1.24 the only active memory function is clearing memory :CLEar. It allows to delete data in the memory associated with the module. The thermometer has three defined memory areas:

- METer – memory located on the analogue (measuring) board: resistance calibration data, board's serial number.
- CH1 (CHannel 1) – the memory associated with the probe in the channel 1 : memory inside the probe , as well as an area dedicated to the memory of the probe on the analogue board. Contains probe's serial number, calibration data etc.
- CH2 (CHannel 2) – the memory associated with the probe in the channel 2 : memory inside the probe , as well as an area dedicated to the memory of the probe on the analogue board. Contains probe's serial number, calibration data etc.

The way of operating memory in CH1 and CH2 is due the fact, that the thermometer that can operate with probes equipped with an internal EEPROM, and with "bare" probes without EEPROM memory. In the second case, data are stored in analogue circuit memory. The command :CLEar includes an additional default sub-command : NAME for compliance with the standard.

Therefore, erasing memory in a first channel of the probe can be done in two ways:

- 1) :MEM:CLEar:NAME CH1
- 2) :CLEar:NAME CH1

To prevent accidental deletion of data , before attempting to erase the memory the user must first unlock access to calibration using the command: CAL : SEC . Deleting data from the EEPROM does not delete them from RAM memory of the thermometer. Therefore, after deleting memory the meter must be restarted.

### *7.6.8 :DISPlay – display operations*

Following catalogue is used to operate the display of thermometer. It contains only one command - :MENU, which switches the values displayed by thermometer.

The temperature is the basic function of the meter and is always displayed. Additionally, as the second parameter, the user can display results:

- GRADient – the temperature gradient per second
- DIFFerence – the difference between the temperatures of probes in the first and second channel,
- RESistance – the resistance of the probe's sensor,
- NONE – additional results are not displayed (for clearance).

To set the display mode the user must call this command with the name of the menu as a parameter. Command :MENU includes an additional default sub-command: NAME for compliance with the standard. Therefore, for example, the display setting of the gradient can be done in two ways:

- 1) :DISP:MENU:NAME GRAD
- 2) :DISP:MENU GRAD

Sending this command without passing any parameter, but as a query, will cause the device to return current name (GRAD, RES etc.). After switching off, the thermometer stores scale that had been used recently.

### 7.6.9 :STATus – status registers handling

Commands grouped in that catalogue, are used for handling additional status registers, which are OPER and QUES. Main status registers (STB and ESR) are handled through special commands.

#### 7.6.9.1 :OPERation – operation register

Handles the operation register (OPER).

- :EVENT? - reads the contents of operation register,
- ENABLE <int> - sets the register mask of OPER status register.

#### 7.6.9.2 QUESTionable – reliability register

Handles the reliability register (QUES).

- :EVENT? - reads the contents of reliability register,
- ENABLE <int> - sets the register mask of QUES status register.

#### 7.6.9.3 :PRESet

This command resets OPER and QUES registers' masks.

### 7.6.10 :SYSTem

Contains set of commands for handling device's settings, not related with performed measurements.

#### 7.6.10.1 :ERRor – reading errors

Reads first error code from queue. After the readout, the error is removed from queue. Full list of notified errors is described in separate chapter.

#### 7.6.10.2 :DATE <year, month, day>

Setting and reading of current date from the device.

#### 7.6.10.3 :TIME <hour, minute, second>

Setting and reading of current date from the device.

## 7.7 Notified errors

The device can notify over a dozen of different errors, which can be divided into few groups. Basic are the thermometer's errors – with positive value, and SCPI errors – with negative value. SCPI errors are defined by standard and are identical for any device which uses such communication method. thermometer errors however, are specific for LDT 2000 device only. Notification of any error during commands' execution, terminates currently performed sequence of procedures (until next prmt – program message terminator).

### 7.7.1 Device errors

- 100 MEASURE ERROR – error in communication with meter's analogue circuit,
- 101 CHANNEL1 ERROR – no probe present in first channel,
- 102 CHANNEL2 ERROR – no probe present in second channel,

- 104 CALIBRATION EXECUTE – can't execute command, since calibration is running,
- 110 CALIBRATION ERROR – resistance calibration error, notified in case when the device can't specify error's cause,
- 111 CALIBRATION START ERROR – can't start resistance calibration,
- 112 RESISTOR MISSING – external standard resistor not connected,
- 113 RESISTOR LOW – external standard resistor has too low resistance,
- 114 RESISTOR HIGH – external standard resistor has too high resistance,
- 120 PROBE CALIBRATION ERROR – probe calibration error, notified in case when it is not possible to give precise error description,
- 121 R0 LOW – too low resistance of sensor for 0 °C temperature,
- 122 R0 HIGH – too high resistance of sensor for 0 °C temperature,
- 123 TEMPERATURE LOW – too small value of temperature, can be notified when entering TMIN and TMAX,
- 124 TEMPERATURE HIGH – too big value of temperature, can be notified when entering TMIN and TMAX,
- 130 CALIBRATION SECURE ERROR – write protection for calibration data is on.
- 140 MEMORY ERROR – both hardware error and triple write or read error,
- 141 CHANNEL1 MEMORY ERROR – no communication with probe in CH1,
- 141 CHANNEL2 MEMORY ERROR – no communication with probe in CH2,
- 143 METER MEMORY ERROR – no communication with analogue board memory
- 150 CALCULATION ERROR – error occurring when using certain coefficients the algorithm is divergent or gives forbidden results,
- 151 CALCULATION ERROR – calculation error for probe in CH1
- 152 CALCULATION ERROR – calculation error for probe in CH2

### 7.7.2 SCPI errors

- 100 COMMAND ERROR – notified in case, when command error cannot be described more precisely,
- 101 INVALID CHARACTER – invalid character in command,
- 102 SYNTAX ERROR – unresolved command, most likely an invalid command type,
- 103 INVALID SEPARATOR – invalid separator in command,
- 104 DATA TYPE ERROR – invalid data type, e.g. number instead of string, passed as parameter,
- 108 PARAMETER NOT ALLOWED – too many parameters provided,
- 109 MISSING PARAMETER – too few parameters provided,
- 110 COMMAND HEADER ERROR – command header error, e.g. invalid catalogue, or no such command in provided catalogue,
- 120 NUMERIC DATA ERROR – error parsing numerical data,
- 200 EXECUTION ERROR – command execution error, notified in case when no precise description is possible,
- 210 TRIGGER ERROR – error triggering the device, e.g. when trying to read measurement without previous configuration,
- 220 PARAMETER ERROR – invalid parameters of execution,
- 221 SETTINGS CONFLICT – proper execution parameters, but no possibility of performing it in current state of device,
- 350 QUEUE OVERFLOW – error queue overflow.



[illegible]

CENTRAL OFFICE:

LEYRO INSTRUMENTS SL  
Avda. Somosierra 24  
28703 San Sebastián de los Reyes  
Madrid  
Tel: +34 912 835 502  
[info@leyro.net](mailto:info@leyro.net)  
[www.leyroinstruments.com](http://www.leyroinstruments.com)





LEYROINSTRUMENTS.COM