



# **TH-03**

# **Thermistor Converter**

User's Guide

# Contents

1 Introduction .....	1
<b>1 Overview</b> .....	1
<b>2 Legal information</b> .....	1
<b>3 Safety warning</b> .....	2
2 Product information .....	4
<b>1 Specifications</b> .....	4
<b>2 Installing the driver</b> .....	4
<b>3 Connecting the TH-03</b> .....	4
<b>4 Principles of operation</b> .....	7
3 Technical reference .....	8
<b>1 Driver summary</b> .....	8
<b>2 Driver routines</b> .....	9
1 th03_open_unit .....	9
2 th03_close_unit .....	9
3 th03_poll_driver .....	9
4 th03_get_cycle .....	9
5 th03_set_channel .....	10
6 th03_get_temp .....	10
7 th03_set_ref_update .....	11
8 th03_get_version .....	11
<b>3 Windows XP/Vista</b> .....	11
<b>4 Programming</b> .....	12
1 C/C++ .....	12
2 Delphi .....	12
3 Excel .....	12
4 Visual Basic .....	13
5 LabVIEW .....	13
6 Agilent VEE .....	13
7 Linux .....	13
<b>5 Serial port settings</b> .....	13
<b>6 Connections</b> .....	14
<b>7 Protocol</b> .....	14
<b>8 Sensor connector</b> .....	15
<b>9 Modem operation</b> .....	16
Index.....	17

# 1 Introduction

## 1.1 Overview

The Pico TH-03 is a complete temperature measuring device for use with PCs and laptops. Up to three channels can be measured per TH-03 using precision thermistor sensors from Pico Technology.

It can be used with the supplied PicoLog data logging program. Alternatively, you can use the TH-03 driver software to develop your own programs to collect and analyse data from the unit.

The TH-03 software provides all of the calculations necessary for thermistor curve normalisation. The TH-03 is highly accurate, and gives results to a resolution of hundredths of a degree, to an accuracy typically better than 0.1°C.

PicoLog and the drivers support up to nine TH-03 units under Windows.

This manual describes the physical and electrical properties of the TH-03, and explains how to use the software drivers. For information about other facilities in PicoLog, please consult the appropriate manual.

## 1.2 Legal information

The material contained in this release is licensed, not sold. Pico Technology Limited grants a license to the person who installs this software, subject to the conditions listed below.

### Access

The licensee agrees to allow access to this software only to persons who have been informed of these conditions and agree to abide by them.

### Usage

The software in this release is for use only with Pico products or with data collected using Pico products.

### Copyright

Pico Technology Limited claims the copyright of, and retains the rights to, all material (software, documents etc.) contained in this release. You may copy and distribute the entire release in its original state, but must not copy individual items within the release other than for backup purposes.

### Liability

Pico Technology and its agents shall not be liable for any loss, damage or injury, howsoever caused, related to the use of Pico Technology equipment or software, unless excluded by statute.

### Fitness for purpose

No two applications are the same. Pico Technology cannot guarantee that its equipment or software is suitable for a given application. It is your responsibility, therefore, to ensure that the product is suitable for your application.

### Mission-critical applications

This software is intended for use on a computer that may be running other software products. For this reason, one of the conditions of the license is that it excludes usage in mission-critical applications, for example life-support systems.

### Viruses

This software was continuously monitored for viruses during production, but you are responsible for virus-checking the software once it is installed.

### Support

If you are dissatisfied with the performance of this software, please contact our technical support staff, who will try to fix the problem within a reasonable time scale. If you are still dissatisfied, please return the product and software to your supplier within 14 days of purchase for a full refund.

### Upgrades

We provide upgrades, free of charge, from our web site at [www.picotech.com](http://www.picotech.com). We reserve the right to charge for updates or replacements sent out on physical media.

### Trademarks

Pico Technology Limited, PicoScope, PicoLog, DrDAQ and EnviroMon are trademarks of Pico Technology Limited, registered in the United Kingdom and other countries

Pico Technology acknowledges the following product names as trademarks of their respective owners: Windows, Excel, Visual Basic, LabVIEW, Agilent VEE, Delphi.

## 1.3 Safety warning

We strongly recommend that you read the general safety information below before using your product for the first time. If the equipment is not used in the manner specified, then the protection provided may be impaired. This could result in damage to your computer and/or injury to yourself or others.

### Maximum input range

The TH-03 is designed to measure voltages in the range of  $\pm 70$  mV. Any voltages in excess of  $\pm 30$  V may cause permanent damage to the unit.

### Mains voltages

Pico products are not designed for use with mains voltages. To measure mains we recommend the use of a differential isolating probe specifically designed for such measurements.

### Safety grounding

The ground of every product is connected directly to the ground of your computer via the interconnecting cable provided. This is done in order to minimise interference. If the PC (especially laptop) is not grounded, reading stability cannot be guaranteed and it may be necessary to manually ground the equipment.

As with most oscilloscopes and data loggers, you should take care to avoid connecting the inputs of the product to anything which may be at a hazardous voltage. If in doubt, use a meter to check that there is no hazardous AC or DC voltage. Failure to check may cause damage to the product and/or computer and could cause injury to yourself or others.

---

Take great care when measuring temperatures near mains equipment. If a sensor is accidentally connected to mains voltages, you risk damage to the converter or your computer and your computer chassis may become live.

You should assume that the product does not have a protective safety earth. Incorrect configuration or use of the device to measure voltages outside the maximum input range can be hazardous.

### Repairs

The unit contains no user-serviceable parts. Repair or calibration of the unit requires specialised test equipment and must be performed by Pico Technology Limited or their authorised distributors.

## 2 Product information

### 2.1 Specifications

Number of channels	3
Temperature range	-30°C to +70°C (Using Pico EL015 sensors)
Resolution	0.003°C
Accuracy	±0.3°C (Over -40°C to 30°C range)
Conversion time	1 ms per active channel + 1600 ms per reference conversion
Input connectors	FCC68 4/4
Output connector	DE9F to computer serial port
Power connector	No external power supply required
Environmental conditions	to 50°C 0% to 95% humidity NOT water-resistant

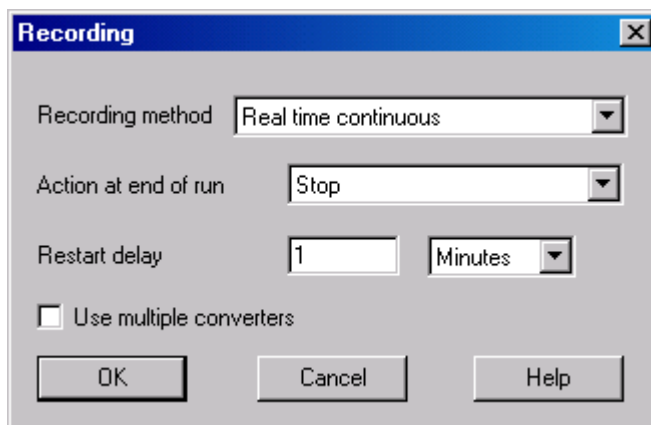
### 2.2 Installing the driver

Installation of the driver is done automatically when you install the PicoLog software. Alternatively, you can download the driver from our website at <http://www.picotech.com>.

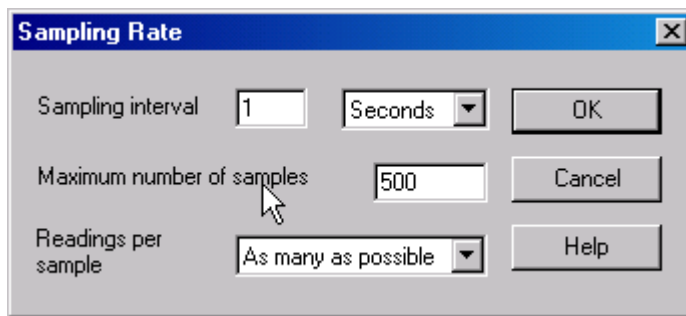
### 2.3 Connecting the TH-03

To use the TH-03, you should connect the its D-connector to the serial port on your computer using the cable provided. If you have a 25-way serial port, use the 9 to 25 way adaptor supplied. Next, connect a EL015 temperature sensor to CH1. Now, to set up the unit with your chosen sensor, do the following:

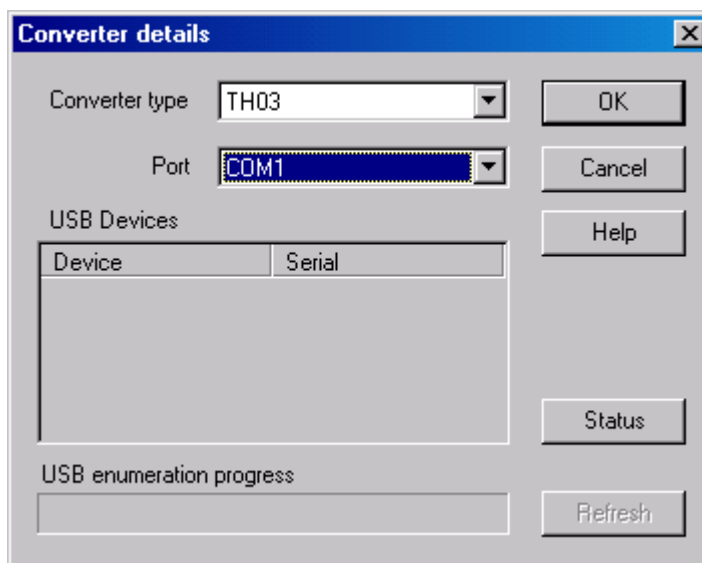
1. Open PicoLog Recorder
2. From the File menu, select New settings. The Recording dialog box appears:



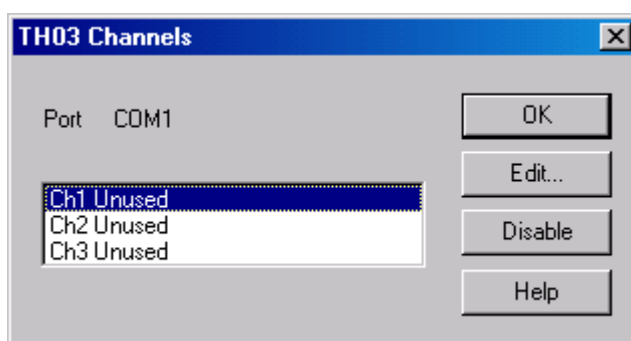
- Click OK. The Sampling Rate dialog box appears:



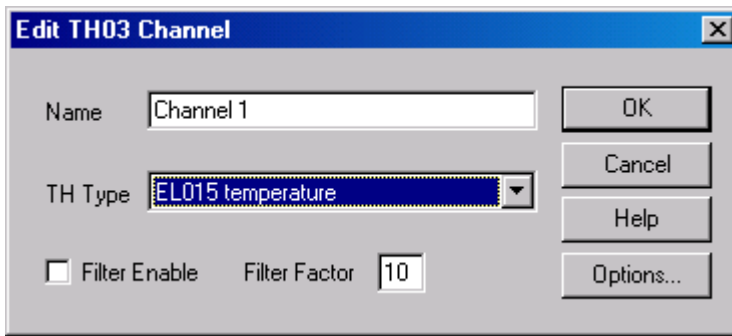
- Click OK. The Converter details dialog box appears:



- From the Converter type drop-down list, select TH03.
- From the Port drop-down list, select the port to which the TH03 is connected and click OK. The TH03 Channels window appears:



7. Double-click on Ch1 Unused. The Edit TH03 Channel dialog box appears:



8. From the TH Type drop-down list, select EL015 temperature, then click OK. You are returned to the TH03 Channels window. Ch1 Unused has become Channel 1.
9. Click OK.

The recorder view should now display the temperature.



## 2.4 Principles of operation

### Thermistors

The TH-03 uses thermistors to measure temperatures. Up to three thermistors can be attached to each TH-03.

A thermistor is a small bead of material whose resistance changes very markedly with temperature. Standard thermistors are in a stainless steel tube about 50 mm long and 8 mm in diameter, but other packages are available.

The resistance normally changes exponentially with temperature. This exponential response is very difficult to deal with in a simple electronic circuit, but it is very easy to use a computer to convert a thermistor resistance short into an accurate temperature measurement.

### Reference resistors

The TH-03 measures the resistance of each thermistor, and also of two internal reference resistors. It uses the readings from the reference resistors to adjust the reading from the thermistor to allow for small errors within the converter.

The software continuously takes readings from the selected thermistor channels and from the reference resistors. As the readings from the reference resistors change very little, it is possible to read the reference resistors less frequently so that the thermistors are read more often. One consequence is that thermistor readings are not at equal short intervals unless reference resistor update is either turned off or set to update every thermistor reading cycle.

### Filtering

The TH-03 software includes a filter, which eliminates errors caused by electrical noise. You can use either the measured or the filtered value: the filtered value is much less prone to electrical noise, but it tends to lag behind if the measured value changes quickly.

The filter time constant is controlled by the filter factor. The filter factor can be adjusted between 1 (no filter) and 100 (very slow filter).

Note: the filter time constant is also affected by the number of channels that are in use. The more channels selected, the slower the filter.

## 3 Technical reference

### 3.1 Driver summary

The TH-03 is supplied with driver routines that you can build into your own programs. Drivers are supplied for use with the following operating systems:

- [Windows XP/Vista](#)
- [Linux](#)

Once you have installed the software, the drivers directory contains the drivers and a selection of examples of how to use the drivers.

The driver routine is supplied as a Windows DLL.

The object files use Pascal linkage conventions and do not require any compiler run-time routines. They can therefore be used with most real-mode and some protected-mode C and Pascal compilers.

The Windows DLL can be used with C, Delphi and Visual Basic programs: it can also be used with programs like Microsoft Excel, where the macro language is a form of Visual Basic. More than one application can access the Windows DLL at the same time, as long as the applications do not change the settings for channels that they are not using.

The following table specifies the function of each of the routines in the driver:

Routine	Description
<a href="#">th03_open_unit</a>	Open the driver to use a specified serial port(s)
<a href="#">th03_close_unit</a>	Close the port (ALWAYS DO THIS!)
<a href="#">th03_poll_driver</a>	Poll the driver (not usually necessary)
<a href="#">th03_get_cycle</a>	Find out when the driver has taken a new set of readings
<a href="#">th03_set_channel</a>	Specify the thermistor type and filtering for a channel
<a href="#">th03_get_temp</a>	Get the most recent temperature reading from a channel
<a href="#">th03_set_ref_update</a>	Set the frequency of update for the reference resistors
<a href="#">th03_get_version</a>	Get the version number of this TH-03

The normal calling sequence for these routines is as follows:

1. Open driver
2. Set channels
3. While you want to measure temperatures,
  4. Get temperature
5. End while
6. Close driver

## 3.2 Driver routines

### 3.2.1 th03\_open\_unit

```
short int th03_open_unit (
    short int port );
```

This routine specifies the serial port number with an TH-03 unit. If you wish to use more than one TH-03, you should call the routine once for each TH-03.

The port must be 1 for COM1, 2 for COM2, etc.

The base address and IRQ information is defined within Windows, so it is not necessary to specify a value.

This routine returns TRUE if the driver successfully opens the TH-03.

### 3.2.2 th03\_close\_unit

```
void th03_close_unit (
    unsigned short int port);
```

This routine disconnects the driver from the specified serial port.

If you successfully open any serial ports, you **MUST** call [th03\\_close\\_unit](#) for each port before you exit from your program. If you do not, your computer may misbehave until you next reboot it.

### 3.2.3 th03\_poll\_driver

```
void th03_poll_driver (void);
```

It is not normally necessary to call this routine, as the driver uses the timer to poll the TH-03. Some programs, like Excel, appear to block the timer and so it is necessary to poll the driver periodically whilst waiting for data.

### 3.2.4 th03\_get\_cycle

```
short int th03_get_cycle(
    long * cycle,
    short int port);
```

This routine returns the number of complete cycles of readings taken from a particular TH-03.

When you call [th03\\_get\\_temp](#), it returns immediately with the most recent reading for the specified channel. If you call it repeatedly, it will return the same reading repeatedly, until the driver takes the next reading from that channel.

If you wish to record values only when the driver has taken a new reading, you can use this routine to find out how many complete cycles of readings the driver has taken, then you can call [th03\\_get\\_temp](#) only when a cycle has completed.

Note: each TH-03 is polled independently, so the cycle numbers for multiple TH-03s may not keep in step.

### 3.2.5 th03\_set\_channel

```
void th03_set_channel (  
    unsigned short int port,  
    unsigned short int channel,  
    unsigned short int th_type,  
    unsigned short int filter_factor);
```

You should call this routine once for each channel that you would like to take readings from. You can do this any time after calling [th03\\_open\\_unit](#).

The fewer channels are selected, the more frequently these channels will be updated: it takes about 2 seconds for reference resistors and 1 second per active channel.

#### Arguments

`channel` - specifies which channel you want to set the details for: it should be between 1 and 3

`th_type` - specifies what type of thermistor is connected to this channel. Set `th_type` to:

- 0 for Off
- 1 for EL015 temperature sensor
- 4 for EL031 light sensor
- 5 for EL029 door switch

See [th03.h](#) for an up-to-date list of sensor types.

The `filter_factor` controls the time constant of the filter. Each time the driver takes a reading from this channel, it updates the filtered value by adding a proportion of the difference between the measured and filtered values. The `filter_factor` sets the proportion that is added. A `filter_factor` of 1 means add all of the difference (effectively no filtering) and 100 means add 1/100 of the difference (very slow filtering). A factor of 10 gives a time constant of about a minute when all channels are selected.

### 3.2.6 th03\_get\_temp

```
short int th03_get_temp (  
    long * temp,  
    short int port,  
    short int channel,  
    short int filtered);
```

Once you open the driver and define some channels, the driver constantly takes readings from the TH-03. When you call this routine, it immediately sets `temp` to the most recent reading for the specified channel, in hundredths of a degree Celsius.

If a reading is available, it returns TRUE, otherwise it returns FALSE. It will normally return FALSE for a few seconds after you open the driver, until the driver has taken a reading from the specified channel.

Channel should be 1 for channel 1, 2 for channel 2, and so on.

If you set `filtered` to `TRUE`, the driver returns a low-pass filtered value of the temperature. The time constant of the filter depends on the value of `filter_factor` for this channel, and on how many channels are active.

### 3.2.7 th03\_set\_ref\_update

```
short int th03_set_ref_update (  
    short int port,  
    short int update_interval);
```

This routine specifies how frequently the computer should take readings from the reference resistors for the specified TH-03. Normally, it is not necessary to call this routine.

The following values are useful:

`0` turns off update - make sure that you leave enough time for at least one reference reading before turning off update

`1` takes one reference reading for every cycle of thermistor readings, so each cycle takes the same amount of time

`10` is the default value: the computer reads the reference resistors every tenth cycle.

### 3.2.8 th03\_get\_version

```
short int th03_get_version (  
    short int * version,  
    short int port);
```

This routine sets `version` to version number of the specified TH-03.

The upper byte of the version is always 8 for a TH-03: the lower byte is the two hex digits of the version and release. It provides a useful check that the link to the TH-03 is working correctly.

## 3.3 Windows XP/Vista

The 32-bit Windows driver is the file `TH0332.dll`, installed in `drivers\win32` directory. If an application is unable to find the DLL, try moving the DLL to `\windows\system`.

## 3.4 Programming

### 3.4.1 C/C++

C

The C example program is a generic Windows application - i.e. it does not use Borland AppExpert or Microsoft AppWizard. To compile the program, create a new project for an Application containing the following files:

```
th03tes.c  
th03tes.rc  
th0332.lib (Borland applications)
```

or

```
th03ms.lib (Microsoft Visual C applications)
```

The following files must be in the same directory:

```
th03.h
```

or

```
th0332.dll (All applications)
```

C++

C++ programs can access all versions of the driver. If `adc11.h` or `adc11w.h` are included in a C++ program, the `PREF1` macro expands to `extern "C";` this disables name-decoration and enables C++ routines to make calls to the driver routines using C headers.

### 3.4.2 Delphi

The `WIN` subdirectory contains a simple program `th03.dpr` which opens the drivers and reads temperatures from the three channels. You will need the following files to build a complete program:

```
● th03fm.dfm  
● th03fm.pas  
● th03.inc
```

The file `TH03.inc` contains procedure prototypes for the driver routines: you can include this file in your application.

This example has been tested with Delphi versions 1, 2 and 3.

### 3.4.3 Excel

The easiest way to transfer data to Excel is to use PicoLog.

If, however, you need to do something that is not possible using PicoLog, you can write an Excel macro which calls the driver to read in a set of data values. The Excel Macro language is similar to Visual Basic.

The example `TH0332.XLS` reads in 20 values of the channel 1 temperature, one per second, and assigns them to cells A1..A20.

### 3.4.4 Visual Basic

The `DRIVERS` subdirectory contains the following files:

- `th0332.VBP`
- `th0332.BAS`
- `th0332.FRM`

### 3.4.5 LabVIEW

The routines described here were tested using LabVIEW for Windows 95 version 4.0.

While it is possible to access all of the driver routines described earlier, it is easier to use the special LabVIEW access routine. The `th03.11b` library in the `DRIVERS` subdirectory shows how to access this routine.

To use this routine, copy `th03.11b` and `th0332.dll` to your LabVIEW `user.lib` directory.

You will then find the `th03` sub-vi, and an example sub-vi which demonstrate how to use them. You can use one of these sub-vis for each of the channels that you wish to measure. The sub-vi accepts the port (1 for COM1), the channel (1 to 3) the thermistor type (1 for standard). The sub-vi returns a temperature.

### 3.4.6 Agilent VEE

The sample program `th03.vee` is in the `drivers` subdirectory. It was tested using Windows 95 and Agilent-VEE version 5.0.

The example demonstrates how to read continuously from the TH-03.

You will need to copy the following file to the program directory:

`th03.vh`

### 3.4.7 Linux

A Linux driver is under development. Please check the drivers section of the Pico Technology web site (<http://www.picotech.com/drivers.html>) for availability.

## 3.5 Serial port settings

The following table shows the standard serial port settings for COM ports.

Port	Base address	Interrupt	Standard?
COM1	3F8	4	Yes
COM2	2F8	3	Yes
COM3	3E8	4	de facto
COM4	2E8	3	de facto
COM5...			No

Note: On most computers, it is not possible to use the same interrupt for two serial ports at the same time. If, for example, you wish to use COM1 and COM3 at the same time, it is necessary to use a serial port card which can be set to an interrupt other than 4. These can be obtained either from Pico Technology or your computer supplier.

### 3.6 Connections

The information presented here is necessary only if you wish to connect the TH-03 to the PC in some unusual way (for example, via a radio modem).

The TH-03 uses the following RS232 data lines (pin connections as on TH-03):

Pin	Name	Usage
3	TX	Data from the PC to the TH-03
2	RX	Data from the TH-03 to the PC
7	RTS	Held at a positive voltage ( $> 7\text{ V}$ ) to power the TH-03
5	GND	0 V line
4	DTR	Held at a negative voltage ( $< -7\text{ V}$ ) to power the TH-03

The driver powers up the TH-03 by enabling RTS and disabling DTR to provide the correct polarity power supply. If these are set incorrectly no damage will occur to either PC or TH-03.

### 3.7 Protocol

The TH-03 operates at 2400 baud with 1 stop bit and no parity.

The driver controls the TH-03 using the following sequence:

1. Switch RTS on and DTR off to provide power.
2. Wait for more than 1 second for the TH-03 to settle
3. Send a single control byte to the TH-03
4. Wait for the 2 byte response from the TH-03

Steps 3 and 4 are repeated for each measurement.

The TH-03 signals the end of conversion by sending two bytes, most significant byte first. No data should be sent to the TH-03 during the conversion, as it may be lost or corrupted.

The commands are:

- 0x01 - get version
- 0x02 - get low reference
- 0x03 - get high reference
- 0x04 - channel 1
- 0x08 - channel 2
- 0x0C - channel 3

The two references should be 20,082 and 43,773 respectively: the actual values may differ by up to 500. The theoretical and actual reference values are used to apply a correction to the measured value:



$$\text{corrected} = 20082 + (\text{measured} - \text{low\_ref}) * (43773 - 20082) / (\text{high\_ref} - \text{low\_ref})$$

The corrected value is then converted to a temperature using a lookup table. The following lookup table is for an EL015 temperature sensor: there is one entry in the table for each degree Celsius.

```
/* 0000 */ 29219U,  
/* 0001 */ 28840U,  
/* 0002 */ 28467U,  
/* 0003 */ 28099U,  
/* 0004 */ 27738U,  
/* 0005 */ 27380U,  
/* 0006 */ 27031U,  
/* 0007 */ 26690U,  
/* 0008 */ 26355U,  
/* 0009 */ 26027U,  
/* 0010 */ 25705U,  
/* 0011 */ 25392U,  
/* 0012 */ 25087U,  
/* 0013 */ 24789U,  
/* 0014 */ 24498U,  
/* 0015 */ 24214U,  
/* 0016 */ 23939U,  
/* 0017 */ 23671U,  
/* 0018 */ 23411U,  
/* 0019 */ 23158U,  
/* 0020 */ 22911U,  
/* 0021 */ 22673U,  
/* 0022 */ 22442U,  
/* 0023 */ 22218U,  
/* 0024 */ 22002U,  
/* 0025 */ 21790U,  
/* 0026 */ 21587U,  
/* 0027 */ 21390U,  
/* 0028 */ 21200U,  
/* 0029 */ 21016U,  
/* 0030 */ 20837U,  
/* 0031 */ 20665U,  
/* 0032 */ 20499U,  
/* 0033 */ 20339U,  
/* 0034 */ 20184U,  
/* 0035 */ 20034U,  
/* 0036 */ 19889U,  
/* 0037 */ 19750U,  
/* 0038 */ 19616U,  
/* 0039 */ 19486U,  
/* 0040 */ 19360U,
```

### 3.8 Sensor connector

The TH-03 sensor input is fitted with an FCC68 type 4/4 connector. Only two pins are used: when facing the socket, these are the left-hand two pins.

Thermistor sensor cables can be extended by up to 100 metres: at normal operating temperatures, this will have a minimal ( $\ll 0.1$  °C) effect on the temperature reading.

### 3.9 Modem operation

The TH-03 is normally connected directly to the computer, but it is also possible to access the TH-03 via a modem using the Windows driver.

It is necessary to provide power to the TH-03, either by instructing the modem to provide power or by connecting a power supply directly to the TH-03. See serial port connections for information.

For some radio modems, there is a delay between sending text to the modem and its arrival at the other end, and a similar delay for the response from the TH-03. If, for example, the maximum possible delay is 150 ms each way, 300 ms total, the following text should be added to `win.ini` so that the driver waits longer for each response.

```
[TH03]  
Turnround=300
```

Note: To comply with current legislation, use only radio modems which comply with the RTTE directive.

## Index

### C

- Connecting the TH-03 4
- Connections 14

### D

- Driver routines
  - th03\_close\_unit 9
  - th03\_get\_cycle 9
  - th03\_get\_temp 10
  - th03\_get\_version 11
  - th03\_open\_unit 9
  - th03\_poll\_driver 9
  - th03\_set\_channel 10
  - th03\_set\_ref\_update 11
- Driver summary 8
- Drivers
  - for Windows NT/2000 11

### I

- Installation 4
- Introduction 1

### L

- Legal information 1

### M

- Modem operation 16

### P

- Principles of operation 7
- Programming
  - C++ 12
  - Delphi 12
  - Excel macros 12
  - HP-Vee 13
  - LabVIEW 13
  - Linux 13
  - Visual Basic 13
- Protocol 14

### S

- Safety warning 2
- Sensor connector 15

- Serial port settings 13
- Specifications 4

### T

- th03\_close\_unit 9
- th03\_get\_cycle 9
- th03\_get\_temp 10
- th03\_get\_version 11
- th03\_open\_unit 9
- th03\_poll\_driver 9
- th03\_set\_channel 10
- th03\_set\_ref\_update 11



## Pico Technology Ltd

The Mill House  
Cambridge Street  
St Neots PE19 1QB  
United Kingdom  
Tel: +44 (0) 1480 396 395  
Fax: +44 (0) 1480 396 296  
Web: [www.picotech.com](http://www.picotech.com)

th03.en 2.10.07

© Copyright 2005-7 Pico Technology Limited. All rights reserved.