PicoScope 3000 Series PC Oscilloscopes

User guide

Table of Contents

1 Introduction

1.1 Overview

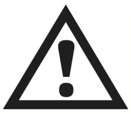
The PicoScope 3000 Series is a range of high speed PC Oscilloscopes, fully USB 2.0-capable and backwards-compatible with USB 1.1. There is no need for an additional power supply as power input is from the USB port, so these oscilloscopes are highly portable.

With the PicoScope software, you can use PicoScope 3000 Series PC Oscilloscopes as oscilloscopes and spectrum analysers; with the PicoLog software, you can use them as data loggers. Alternatively, you might want to use some of the API functions to develop your own programs to collect and analyse data from the oscilloscope.

A typical PicoScope 3000 Series PC Oscilloscope is supplied with the following items:
- USB cable, for use with USB 1.1 and 2.0 ports
- Software CD
- Installation guide
- Quick start guide

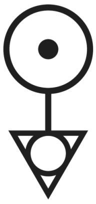
1.2 *Safety symbols*

Symbol 1: Warning Triangle



This symbol indicates that a safety hazard exists on the indicated connections if correct precautions are not taken. Read all safety documentation associated with the product before using it.

Symbol 2: Equipotential



This symbol indicates that the outer shells of the indicated BNC connectors are all at the same potential (shorted together). You must therefore take necessary precautions to avoid applying a potential across the return connections of the indicated BNC terminals as this may result in a large current flow causing damage to the product and/or connected equipment.

1.3 Safety warning

We strongly recommend that you read the general safety information below before using your oscilloscope for the first time. Safety protection built in to equipment may cease to function if the equipment is used incorrectly. This could cause damage to your computer, or lead to injury to yourself and others.

Maximum input range

PicoScope 3000 Series PC Oscilloscopes are designed to measure voltages in the range -20 V to +20 V and are protected to ±50 V (3204/5/6 variants) or ±100 V (3224/3424 variants). Contact with voltages outside the protection range may cause physical damage.

Mains voltages

Pico Technology products are not designed for use with mains voltages. To measure mains, use a differential isolating probe specifically designed for a high source voltage.

Safety grounding

PicoScope 3000 Series PC Oscilloscopes connect direct to the ground of a computer through the interconnecting cable provided. This method minimises interference.

As with most oscilloscopes, avoid connecting the ground input to any source other than ground. If in doubt, use a meter to check that there is no significant AC or DC voltage between the ground input of the oscilloscope and the point to which you intend to connect it. Failure to check may cause damage to your computer, or lead to injury to yourself and others.

You should assume that the product does not have a protective safety earth.

Repairs

The oscilloscope contains no serviceable parts. Repair or calibration of the oscilloscope requires specialised test equipment and must be performed by Pico Technology.

1.4 FCC notice

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to **Part 15 of the FCC Rules**. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his or her own expense.

For safety and maintenance information see the safety warning.

1.5 CE notice

The PicoScope 3000 Series PC Oscilloscopes meet the intent of the **EMC directive 89/336/EEC** and have been designed to **EN61326-1 (1997) Class B Emissions and Immunity** standard.

PicoScope 3000 Series PC Oscilloscopes also meet the intent of the **Low Voltage Directive** and have been designed to meet the **BS EN 61010-1:2001 IEC 61010-1:2001** (safety requirements for electrical equipment, control, and laboratory use) standard.

1.6 Licence conditions

The material contained in this release is licensed, not sold. Pico Technology Limited grants a licence 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
Because 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 licence is that it excludes usage in mission-critical applications; for example, life-support systems.

1.7 Trademarks

Windows and **Excel** are registered trademarks or trademarks of Microsoft Corporation in the USA and other countries.

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

1.8 Warranty

Pico Technology warrants upon delivery, and for a period of 24 months unless otherwise stated from the date of delivery, that the Goods will be free from defects in material and workmanship.

Pico Technology shall not be liable for a breach of the warranty if the defect has been caused by fair wear and tear, wilful damage, negligence, abnormal working conditions or failure to follow Pico Technology's spoken or written advice on the storage, installation, commissioning, use or maintenance of the Goods or (if no advice has been given) good trade practice; or if the Customer alters or repairs such Goods without the written consent of Pico Technology.

1.9 Company details

Address:

Pico Technology Limited
The Mill House
Cambridge Street
St Neots
Cambridgeshire
PE19 1QB
United Kingdom

Phone:	+44 (0) 1480 396 395
Fax:	+44 (0) 1480 396 296

Email:

Technical Support:	tech@picotech.com
Sales:	sales@picotech.com

Web site:

www.picotech.com

2 Product information

2.1 Minimum system requirements

For PicoScope 3000 Series PC Oscilloscopes to operate, you must have a computer with the minimum system requirements to run Windows or the following (whichever is the higher specification):

Processor	Pentium class processor or equivalent minimum.
Memory	32 MB minimum.
Disk space	10 MB minimum.
Operating system	Microsoft Windows 98 SE, ME, 2000, XP or later.
Ports	USB 1.1 compliant port minimum. USB 2.0 compliant port recommended. Must be connected direct to the port or a powered USB hub. Will not work on a passive hub.

2.2 Installation instructions

Important
Do not connect your PicoScope 3000 Series PC Oscilloscope to the PC until you have installed the software.

Procedure

- Follow the instructions in the Quick Start Guide included with your product package.
- Connect your PC Oscilloscope to the PC using the USB cable supplied.
- There is no need for an additional power supply, as the unit obtains its power from the USB port.

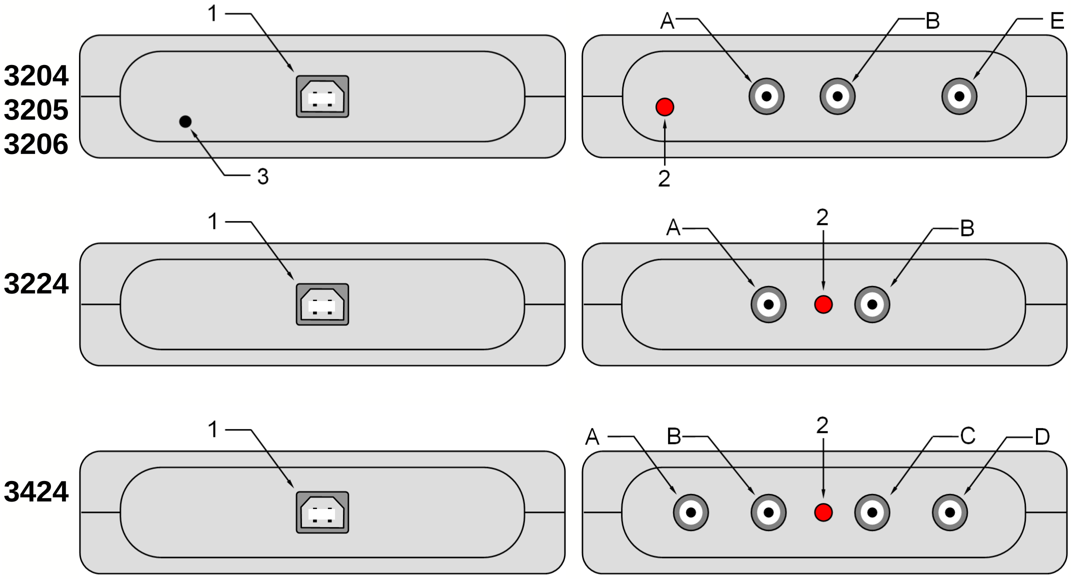
Checking the installation

Once you have installed the software and connected the PC Oscilloscope to the PC, start the PicoScope or PicoLog software. The software should now display the voltage that you have connected. If you are using the PicoScope software and have a probe connected to your oscilloscope, you should see a small 50Hz or 60Hz mains signal in the oscilloscope window when you touch the probe tip with your finger .

Standard oscilloscope connectors

PicoScope 3000 Series PC Oscilloscopes have standard oscilloscope connectors. The input impedance is also standard, so the x10 function on scope probes works correctly.

Connector diagrams



1 USB port.
2 LED. When lit, indicates the oscilloscope is sampling data.
3 Power socket. 12 V DC at 500 mA.
A Input channel A.
B Input channel B.
C Input channel C.
D Input channel D.
E External trigger input / Signal generator output.

The BNC connector labelled '**E**' on the 3204/5/6 models has two functions. In normal use it is the external trigger input with an adjustable threshold. Alternatively, on some oscilloscopes, this connector can also be used to output sine, square and triangle waveforms which can be swept back and forth at a user-defined frequency. The integrated signal generator can be controlled via the PicoScope software or by API calls. The signal generator can also be used to compensate x10 scope probes when set to output a square wave.

Ground loops

If you experience excessive noise or voltage offsets when using the PicoScope 3204/5/6 variants, you may have a ground loop problem. See Powering the 3204/5/6 for advice on overcoming this.

Moving your PicoScope PC Oscilloscope to another USB port

When you install the PicoScope 3000 Series PC Oscilloscope by plugging it into a USB port, Windows associates the Pico driver software with that port. If you later move the oscilloscope to a different USB port, Windows will display the "New Hardware Found Wizard" again. When this occurs, simply follow the steps listed in the Quick Start Guide after the instruction "Connect the PicoScope 3000 Series PC Oscilloscope to the PC..." As all the software you need is already installed on your computer, there is no need to insert the Pico Software CD again.

2.3 Specifications

Variant	3204	3205	3206	3224	3424
Vertical resolution	8 bits	8 bits	8 bits	12 bits	
Analog bandwidth	50 MHz	100 MHz	200 MHz	10 MHz	
Max. sampling rate					
Single channel	50 MS/s	100 MS/s	200 MS/s	20 MS/s	20 MS/s
Dual channel	50 MS/s	100 MS/s	100 MS/s	10 MS/s	10 MS/s
Triple or quad channel	-	-	-	-	5 MS/s
Repetitive signals	2.5 GS/s	5 GS/s	10 GS/s	-	-
Trigger bandwidth	50 MHz	100 MHz	150 MHz	10 MHz	
Buffer size (samples per channel)					
One channel in use	256 K	512 K	1 M	512 K	512 K
Two channels in use	128 K	256 K	512 K	256 K	256 K
3 or 4 channels in use	-	-	-	-	128 K
Inputs	2 BNC channels				4 BNC channels
	1 MΩ impedance AC/DC coupling 20 pF capacitance				
Outputs					
Signal generator	Fixed (Note 1)	Variable (Note 2)		None	
External trigger	1 BNC input shared with signal generator Variable trigger threshold ± 20 V Rising/falling 12.2 mV resolution 1 MΩ impedance			None	
Voltage ranges	± 100 mV to ± 20 V in 1, 2, 5 steps			± 20 mV to ± 20 V	
Accuracy	3% voltage 100 ppm time			1% voltage 100 ppm time	
Operating environment					
Temperature range	0°C to 70°C (25°C for quoted accuracy)			0°C to 70°C (20°C to 30°C for quoted accuracy)	
Humidity	25% to 75% RH			25% to 75% RH	
Overload protection					
Channels	± 50 V			± 100 V	
External trigger	± 30 V			-	
PC connection	USB 2.0 Compatible with USB 1.1				
Power supply	From USB port: 4.6 to 5.25 V 500 mA External power supply is not required			From USB port	
Dimensions	140 mm x 190 mm x 45 mm				
Compliance	CE standard; FCC rules				

(1) 1 BNC shared with external trigger. Fixed frequency 1 kHz. 5 V square wave. 600 Ω output impedance.

(2) 1 BNC shared with external trigger. Variable frequency 100 Hz to 1 MHz. 5 V square wave, 1 V sine wave and triangle functions. Repeat sweep function. Dual slope function. 600 Ω output impedance.

3 Technical reference

3.1 Driver

The Windows 98SE/ME/2000/XP/2003 32-bit driver, picopp.sys, is installed in Windows. It is loaded using an inf file, picopp.inf.

Once you have installed the PicoScope and PicoLog software, and the PicoScope 3000 Series PC Oscilloscope is plugged in for the first time, Windows will automatically install the driver.

3.2 Driver error codes

This section is aimed at those people who intend to write their own programs for use with the driver. A description of the driver error codes is given below. If the PicoScope or PicoLog software reports an error, refer to the FAQ.

Code	Enumeration	Description
0	PS3000_OK	The oscilloscope is functioning correctly.
1	PS3000_MAX_UNITS_OPENED	Attempts have been made to open more than PS3000_MAX_UNITS.
2	PS3000_MEM_FAIL	Not enough memory could be allocated on the host machine.
3	PS3000_NOT_FOUND	An oscilloscope could not be found.
4	PS3000_FW_FAIL	Unable to download firmware.
5	PS3000_NOT_RESPONDING	The oscilloscope is not responding to commands from the PC.
6	PS3000_CONFIG_FAIL	The configuration information in the oscilloscope has become corrupt or is missing.
7	PS3000_OS_NOT_SUPPORTED	Driver supports Windows 98SE, ME, 2000, XP, 2003 or later.

3.3 Functions

3.3.1 ps3000_open_unit

```
short ps3000_open_unit    ( void)
```

This function opens a PicoScope 3000 Series PC Oscilloscope. The API driver can support up to four oscilloscopes.

Arguments	None.
Returns	`-1` if the oscilloscope fails to open, `0` if no oscilloscope is found, `>0` (handle) if the device opened.

3.3.2 ps3000_get_unit_info

```
short ps3000_get_unit_info (   short   handle,
                               char    * string,
                               short   string_length,
                               short   info     )
```

This function writes oscilloscope information to a character string. If the oscilloscope fails to open, only infos 0 and 6 are available to explain why the last open unit call failed.

Arguments	handle, the handle to the device from which info is required. If an invalid handle is passed, the error code from the last unit that failed to open is returned from line 2. * string, a pointer to the character string buffer in the calling function where the unit information string (selected with line) will be stored. If a null pointer is passed, no information will be written. string_length, the length of the character string buffer. If the string is not long enough to accept all of the information, only the first string_length characters are returned. info, is an enumerated type specifying what information is required from the driver.
Returns	The length of the string written to the character string buffer, string, by the function. If one of the parameters are out of range, or a null pointer is passed for string, zero will be returned.

info	Description	Example
PS3000_DRIVER_VERSION (0)	Returns the version number of the DLL used by the oscilloscope driver.	"1, 0, 0, 2"
PS3000_USB_VERSION (1)	Returns the type of USB connection that is being used to connect the oscilloscope to the computer.	"1.1" or "2.0"
PS3000_HARDWARE_VERSION (2)	Returns information about what is the hardware version of the attached oscilloscope.	"1"
PS3000_VARIANT_INFO (3)	Returns information about what model of PicoScope 3000 Series PC Oscilloscope is attached to the computer.	"3206"
PS3000_BATCH_AND_SERIAL (4)	Returns the batch and serial number of the oscilloscope.	"CMY66/052"
PS3000_CAL_DATE (5)	Returns the calibration date of the oscilloscope.	"21Oct03"
PS3000_ERROR_CODE (6)	Returns one of the Error codes.	"4"

3.3.3 ps3000_flash_led

```
short ps3000_flash_led ( short handle )
```

Flashes the LED on the front of the oscilloscope three times and returns within one second.

Arguments	`handle`, the handle of the PicoScope 3000 Series PC Oscilloscope.
Returns	`1` if a valid handle is passed, `0` if not.

3.3.4 ps3000_close_unit

```
short ps3000_close_unit  (short handle)
```

Shuts down a PicoScope 3000 Series PC Oscilloscope.

Arguments	`handle`, the handle, returned by ps3000_open_unit, of the oscilloscope being closed.
Returns	`1` if a valid handle is passed, `0` if not.

3.3.5 ps3000_set_channel

```
short ps3000_set_channel (    short      handle,
                              short      channel,
                              short      enabled,
                              short      dc,
                              short      range )
```

Specifies if a is to be enabled, the position of the AC\DC switch and the input range.

Arguments	handle, the handle to the required device. channel, an enumerated type, use PS3000_CHANNEL_A (0), PS3000_CHANNEL_B (1), PS3000_CHANNEL_C (2) or PS3000_CHANNEL_D (3). Channels C and D are not available on all models. enabled, specify if the channel is active: TRUE=active, FALSE=inactive. dc, specifies the position of the AC/DC switch: TRUE=DC, FALSE=AC. range, a code between 0 and 12. See the table below.
Returns	0 if unsuccessful, or if one or more of the arguments are out of range. 1 if successful.

Code	Enumeration	Range	
0	PS3000_10MV	±10 mV	*Not available on all variants.*
1	PS3000_20MV	±20 mV	*Not available on all variants..*
2	PS3000_50MV	±50 mV	*Not available on all variants.*
3	PS3000_100MV	±100 mV	
4	PS3000_200MV	±200 mV	
5	PS3000_500MV	±500 mV	
6	PS3000_1V	±1 V	
7	PS3000_2V	±2 V	
8	PS3000_5V	±5 V	
9	PS3000_10V	±10 V	
10	PS3000_20V	±20 V	
11	PS3000_50V	±50 V	*Not available on all variants.*

3.3.6 ps3000_get_timebase

```
short ps3000_get_timebase (   short    handle,
                              short    timebase,
                              long     no_of_samples,
                              long     * time_interval_ns,
                              short    * time_units,
                              short    oversample,
                              long     * max_samples)
```

This function discovers which timebases are available on the oscilloscope. This function should be called after channel and ETS options have been set.

Arguments	handle, the handle to the required device. timebase, a code between 0 and the maximum timebase (dependant on variant). Timebase 0 is the fastest timebase, timebase 1 is twice the time per sample as timebase 0, timebase 2 is four times, etc. no_of_samples, the number of samples required. This value is used to calculate the most suitable time unit to use. time_interval_ns, a pointer to the time interval, in ns, between readings at the selected timebase. If a null pointer is passed, nothing will be written here. time_units, a pointer to the most suitable time units to return data in, when calling ps3000_get_times_and_values. If a null pointer is passed, nothing will be written here. oversample, the amount of oversample required. An oversample of 4 would quadruple the time interval and quarter the maximum samples. At the same time it would increase the effective resolution by the amount given by the equation below: Increase in resolution (bits) = (log oversample) / (2 log 2) max_samples, A pointer to the maximum samples available. The maximum samples may vary depending on the number of channels enabled, the timebase chosen and the oversample selected. If this pointer is null, nothing would be written here.
Returns	1 if all parameters are in range, otherwise 0.

3.3.7 ps3000_set_siggen

```
long ps3000_set_siggen (    short    handle,
                            short    wave_type,
                            long     start_frequency,
                            long     stop_frequency,
                            float    increment,
                            short    dwell_time,
                            short    repeat,
                            short    dual_slope)
```

This function is used to enable or disable the signal generator and sweep functions. Sweep functions are not available if the oscilloscope is in streaming mode.

The signal generator is available only on the PicoScope 3204/5/6 PC Oscilloscope variants. See remarks and specifications for more information.

Arguments	handle, the handle of the required device. wave_type, the type of wave. Choose PS3000_SQUARE (0), PS3000_TRIANGLE (1) or PS3000_SINE (2). *This argument has no effect if used with the PicoScope 3204 variant.* start_frequency, the required frequency, in the range 0 < freq < 1 MHz, to start the sweep or the frequency generated in a non-sweep mode. 0 switches the signal generator off. stop_frequency, the required stop frequency of the sweep, in the range 0 < freq < 1 MHz but not necessarily greater than start_frequency. If the start and stop frequencies are the same, the signal generator will be run with a constant frequency. *This argument has no effect if used with the PicoScope 3204 variant..* increment, the size of the steps to increment or decrement the frequency by in a sweep mode. This must always be positive; the start and stop frequencies will determine whether to increment or decrement. This must be a frequency in the range 0.1 Hz < increment < |stop_frequency - start_frequency|. This is not used in a non-sweep mode. *This argument has no effect if used with the PicoScope 3204 variant..* dwell_time, This is the time, in ms, to wait before increasing the frequency by increment in a sweep mode. This is unused in a non-sweep mode. *This argument has no effect if used with the PicoScope 3204 variant..* repeat, TRUE restarts the sweep when the stop_frequency is reached, FALSE continues indefinitely at stop_frequency when it is reached. *This argument has no effect if used with the PicoScope 3204 variant..* dual_slope, if repeat is TRUE this specifies what to do at the stop_frequency. TRUE will sweep back towards the start_frequency, FALSE will restart the sweep from start_frequency. *This argument has no effect if used with the PicoScope 3204 variant..*
Returns	The actual frequency or start frequency, in hertz, that is generated. Zero if one of the parameters are not in range.

Remarks

The PicoScope 3204 variant has a simple 1 kHz square wave signal generator for scope probe calibration. With this variant, therefore, only two arguments of this function have any effect:

To switch the square wave on, use a valid handle and set start_frequency to a non-zero value. To switch the square wave off, use a valid handle and set start_frequency to 0.

3.3.8 ps3000_set_ets

```
long ps3000_set_ets (   short    handle,
                        short    mode,
                        short    ets_cycles,
                        short    ets_interleave)
```

This function is used to enable or disable ETS (equivalent time sampling) and to set the ETS parameters.

Note: The 3224 and 3424 variants do not have an ETS mode.

Arguments	`handle`, the handle to the required device. `mode`, `PS3000_ETS_OFF` (`0`) - disables ETS, `PS3000_ETS_FAST`(`1`) - enable ETS and provides ets_cycles of data, which may contain data from previously returned cycles, `PS3000_ETS_SLOW` (`2`) - enable ETS and provide fresh data every `ets_cycles` cycles. `PS3000_ETS_SLOW` takes longer to provide each data set, but the data sets are more stable and unique. `ets_cycles`, Specifies the number of cycles to store: the computer can then select `ets_interleave` cycles to give the most uniform spread of samples. `ets_cycles` should be between two and five times the value of `ets_interleave`. `ets_interleave`, Specifies the number of ETS interleaves to use. If the sample time is 20 ns and the interleave 10, the approximate time per sample will be 2 ns.
Returns	If ETS is enabled, the effective sample time will be returned. Zero if ETS is disabled or one of the parameters is out of range.

3.3.9 ps3000_set_trigger

```
short ps3000_set_trigger (    short    handle,
                              short    source,
                              short    threshold,
                              short    direction,
                              short    delay,
                              short    auto_trigger_ms)
```

This function is used to enable or disable triggering and its parameters. Triggering is not available in streaming mode.

Arguments	handle, the handle to the required device. source, specifies where to look for a trigger. Use PS3000_CHANNEL_A (0), PS3000_CHANNEL_B (1), PS3000_CHANNEL_C (2), PS3000_CHANNEL_D (3), PS3000_EXTERNAL(4) or PS3000_NONE(5). Channels C, D and External are not available on all models. threshold, the threshold for the trigger event. This is scaled in 16-bit ADC counts at the currently selected range. If an external trigger is enabled the range is fixed at +/-20V. direction, use PS3000_RISING(0) or PS3000_FALLING (1). delay, This specifies the delay, as a percentage of the requested number of data points, between the trigger event and the start of the block. It should be in the range -100% to +100%. Thus, 0% means that the first data value in the block, and -50% means that the trigger event is in the middle of the block. auto_trigger_ms, the delay in ms after which the oscilloscope will collect samples if no trigger event occurs. If this is set to zero the oscilloscope will wait for a trigger indefinitely.
Returns	0 if one of the parameters are out of range, otherwise 1.

3.3.10 ps3000_run_block

```
short ps3000_run_block (    short    handle,
                            long     no_of_samples,
                            short    timebase,
                            short    oversample,
                            long     * time_indisposed_ms)
```

This function tells the oscilloscope to start collecting data in block mode.

Arguments	`handle`, the handle to the required device. `no_of_samples`, the number of samples to return. `timebase`, a code between 0 and the maximum timebase available (consult the driver header file). Timebase 0 gives the maximum sample rate available, timebase 1 selects a sample rate half as fast, timebase 2 is half as fast again and so on. For the maximum sample rate, see the specifications. Note that the number of channels enabled may affect the availability of the fastest timebases. `oversample`, the oversample factor, a number between 1 and 256. `time_indisposed_ms`, a pointer to the `time_indisposed_ms`. This is the approximate time, in ms, over which the ADC will collect data. If a trigger is set, it is the amount of time the ADC takes, in ms, to collect a block of data after a trigger event, calculated as sample interval x number of points required. Note: The actual time may differ from computer to computer, depending on how fast the computer can respond to I/O requests.
Returns	`0` if one of the parameters is out of range, otherwise 1.

3.3.11 *ps3000_run_streaming*

```
short ps3000_run_streaming (  short    handle,
                              short    time_interval_ms,
                              long     max_samples,
                              short    windowed)
```

This function tells the oscilloscope to start collecting data in streaming mode. If this function is called when a trigger has been enabled, the trigger settings will be ignored.

Arguments	handle, the handle to the required device. time_interval_ms, the time interval, in ms, between data points. This can be no shorter than 1 ms. max_samples, the maximum number of samples that the driver is to store. This can be no greater than 60 000. It is the caller's responsibility to retrieve data before the oldest values are overwritten. windowed, if this is 0, only the values taken since the last call to get values are returned. If this is 1, the number of values requested by get_values are returned, even if they have already been read by ps_get_values.
Returns	1 if streaming has been enabled correctly, otherwise 0 if a problem occurred or a value was out of range.

3.3.12 ps3000_ready

```
short ps3000_ready (     short handle)
```

This function checks to see if the oscilloscope has finished the last data collection operation. This function does nothing if the oscilloscope is in streaming mode.

Arguments	handle, the handle to the required device.
Returns	1 (meaning 'ready') is returned when the oscilloscope has collected a complete block of data or the auto trigger timeout has been reached. If an invalid handle is passed or if the oscilloscope is in streaming mode it returns 0 (meaning 'not ready'). -1 (meaning 'device not attached') is returned if the endpoint transfer fails indicating that the unit may well have been unplugged.

3.3.13 ps3000_stop

```
void ps3000_stop (  short handle)
```

Call this function to stop the oscilloscope from sampling data. If this function is called before a trigger event occurs, the oscilloscope may not contain valid data.

Arguments	`handle`, the handle to the required device.
Returns	`0` if an invalid handle is passed, otherwise `1`.

3.3.14 ps3000_get_values

```
long ps3000_get_values(     short    handle
                            short    * buffer_a,
                            short    * buffer_b,
                            short    * buffer_c,
                            short    * buffer_d,
                            short    * overflow,
                            long     no_of_values )
```

This function is used to get values. This function does nothing if ETS triggering is enabled.

Arguments	handle, the handle to the required device. buffer_a, buffer_b, buffer_c, buffer_d, pointers to the buffers that receive data from the specified channels (A, B, C or D). A pointer is unused if the oscilloscope is not collecting data from that channel. If a pointer is NULL, nothing will be written to it. overflow, a bit pattern indicating whether an overflow has occurred on a channel. Bit 0 is the LSB. Bit 0 --> channel A Bit 1 --> channel B Bit 2 --> channel C Bit 3 --> channel D no_of_values. The number of data points to return. In streaming mode, this is the maximum no of values to return.
Returns	The actual number of data values per channel returned, which may be less than no_of_values if streaming. FALSE is returned if one of the parameters is out of range.

3.3.15 ps3000_get_times_and_values

```
long ps3000_get_times_and_values(    short    handle
                                     long     * times,
                                     short    * buffer_a,
                                     short    * buffer_b,
                                     short    * buffer_c,
                                     short    * buffer_d,
                                     short    * overflow,
                                     short    time_units,
                                     long     no_of_values )
```

This function is used to get values and times. It will not return any valid times if the oscilloscope is in streaming mode. It is essential for ETS operation.

Arguments	handle, the handle to the required device. times, a pointer to the buffer for the times. Each time is the interval between the trigger event and the corresponding sample. Times before the trigger event are negative, and times after the trigger event are positive. buffer_a, buffer_b, buffer_c, buffer_d, pointers to the buffers that receive data from the specified channels (A, B, C or D). A pointer is unused if the oscilloscope is not collecting data from that channel. If a pointer is NULL, nothing will be written to it. overflow, a bit pattern indicating whether an overflow has occurred on a channel. Bit 0 is the LSB. Bit 0 --> channel A Bit 1 --> channel B Bit 2 --> channel C Bit 3 --> channel D time_units, which can be one of: PS3000_FS (0), PS3000_PS (1), PS3000_NS (2), PS3000_US (3), PS3000_MS (4) or PS3000_S (5) which are femtoseconds, picoseconds, nanoseconds (default), microseconds, milliseconds and seconds respectively. no_of_values, the number of data points to return. In streaming mode, this is the maximum number of values to return.
Returns	The actual number of data values per channel returned, which may be less than the no_of_values if streaming. 0 is returned if one or more of the parameters are out of range or if the times will overflow with the time_units requested. Use ps3000_get_timebase in order to acquire the most suitable time_units.

3.4 Advanced features

3.4.1 Sampling modes

PicoScope 3000 Series PC Oscilloscopes can run in various sampling modes. At high sampling rates, the oscilloscope collects data much faster than a PC can read it. To compensate for this, the oscilloscope stores a block of data in an internal memory buffer, delaying transfer to the PC until the required number of data points have been sampled. This is called block mode. At very low sampling rates, you may want to switch to streaming mode. This allows accurately timed data to be transferred back to the PC in short blocks, without gaps.

3.4.2 More on block mode

In block mode, the computer prompts a PicoScope 3000 series PC Oscilloscope to collect a block of data into its internal memory. When the oscilloscope has collected the whole block, it will signal it is ready, and transfer the whole block into computer memory via the USB port.

The maximum number of values depends upon the size of the oscilloscope's memory. A PicoScope 3000 Series PC Oscilloscope can sample at a number of different rates. These rates correspond to the maximum clock frequency multiplied by 0.5, 0.25, 0.125, and so on.

There is a separate memory buffer for each channel. When a channel is unused, its memory can be utilised by the enabled channels. On the faster models, one input can be routed to two circuits in the oscilloscope, thus doubling the effective sampling rate of a single channel.

The driver for a PicoScope 3000 Series PC Oscilloscope normally performs a number of setup operations before collecting each block of data. This can take up to 50 milliseconds. If it is necessary to collect data with the minimum time interval between blocks, avoid calling setup functions between calls to ps3000_run_block(), ps3000_ready (), ps3000_stop() (not normally used) and ps3000_get_values ().

3.4.3 More on streaming mode

In streaming mode, the computer prompts the PicoScope 3000 Series PC Oscilloscope to start collecting data. The data is then transferred back to the PC without being stored in oscilloscope memory. Data can be sampled with a period of between 1 ms and 60 s.

Data can be transferred by the oscilloscope's driver to a computer program either in normal or windowed mode. In normal mode, any data collected since the last data transfer operation is returned in its entirety. In windowed mode, a fixed number (n) of samples is returned, where the oldest samples may have already been returned before. Normal mode is useful if the computer program requires fresh data on every transfer. Windowed mode is useful when the program requires a constant time frame of data.

Once the oscilloscope is collecting data in streaming mode, any setup changes (for example, changing a channel range or AC/DC setting in the PicoScope software application) will cause a restart of the data stream. The driver can buffer up to 32K samples of data per channel, but the user must ensure that the ps3000_get_values () function is called frequently enough to avoid buffer overrun.

The ps3000_get_times_and_values () function will always return FALSE (0) in streaming mode.

3.4.4 Triggering

The PicoScope 3000 Series PC Oscilloscope can either start collecting data immediately, or it can be programmed to wait for a trigger event to occur. In either case, you need to use the ps3000_set_trigger () function. A trigger event can occur when the channel A or B input crosses a threshold voltage, or when an external trigger input crosses a threshold voltage. The trigger event can be either a rising or a falling edge.

The external trigger input uses the same physical connection as the signal generator output, so these two functions cannot be used at the same time. It is possible, however, to use the output from the signal generator as a trigger. Triggering is available in block mode only. Any call to the ps3000_set_trigger () function has no effect in streaming mode.

3.4.5 ETS (Equivalent Time Sampling)

ETS is a way of increasing the effective sample rate when working with repetitive signals. It is not possible to use ETS with one-shot signals. ETS is controlled by the ps3000_set_trigger () and ps3000_set_ets() functions. ETS is available in block mode only. Calls to the ps3000_set_trigger () function have no effect in streaming mode. As ETS will return random time intervals, the ps3000_get_times_and_values () function must be used. The ps3000_get_values () function will return FALSE (0).

Only the 3204/5/6 variants provide ETS.

3.4.6 Voltage ranges

It is possible to set the gain for each channel with the ps3000_set_channel() function. This will give an input voltage range between ±20 mV (for the 3224 and 3424) or ±100 mV (for the 3204/5/6) and ±20V. The external trigger has a fixed input range of ±20 V.

3.4.7 AC/DC operation

Using the ps3000_set_channel function, each channel can be set to either AC or DC coupling. When AC coupling is used, any DC component of the signal is filtered out.

3.4.8 Oversampling

When the oscilloscope is operating in block mode at speeds less than the maximum, it is possible to oversample. Oversampling is taking more than one measurement during a time interval and returning an average. This reduces the effects of noise, and increases the effective vertical resolution of the oscilloscope.

3.4.9 Scaling

The PicoScope 3000 Series PC Oscilloscopes have resolutions that range from 8 bits to 12 bits, but the oscilloscope driver normalises all readings to 16 bits. This enables it to take advantage of noise reduction from oversampling, when this is enabled. The following table shows the relationship between the reading from the driver and the voltage of the signal.

Reading	Voltage
-32 767	Minimum
0	Zero volts
32 767	Maximum

3.4.10 Signal generator

The PicoScope 3204/5/6 PC Oscilloscopes have a built-in signal generator which is set using ps3000_set_siggen(). The output of the 3204 is a fixed-frequency square wave, while the 3205 and 3206 can produce a selection of accurate frequencies from 100 Hz to 1 MHz, and the waveform can be set to sine, square or triangle and swept back and forth in frequency. These options are selected under software control.

The signal generator output and external trigger input share the same connector, so these two functions cannot be used at the same time. It is possible, however, to use the output from the signal generator as a trigger.

3.4.11 Combining oscilloscopes

With PicoLog or your own program it is possible to collect data using up to four PicoScope 3000 Series PC Oscilloscopes at the same time. Each oscilloscope must be connected to a separate USB port. If a USB hub is used it must be a powered hub. The ps3000_open_unit () function returns a handle to an oscilloscope. All of the other functions require this handle for oscilloscope identification. For example, to collect data from two oscilloscopes at the same time:

```
handle1 = ps3000_open ()
handle2 = ps3000_open ()

ps3000_set_channel (handle1)
... set up unit 1
ps3000_run(handle1)

ps3000_set_channel (handle2)
... set up unit 2
ps3000_run(handle2)

ready = FALSE
while not ready
    ready = ps3000_ready (handle1)
    ready &= ps3000_ready (handle2)

ps3000_get_values(handle1)
ps3000_get_values(handle2)
```

Note: It is not possible to synchronise the collection of data between oscilloscopes that are being used in combination.

3.5 Using different modes

3.5.1 Introduction

The previous section on advanced features supplied the programmer with extended information on PicoScope 3000 Series PC Oscilloscopes. The C sample program, `ps3000con.c`, demonstrates how to use the functions of the driver software, and includes examples showing how to use each of the modes available.

3.5.2 Using block mode

This is the general procedure for reading and displaying data in block mode:

1. Open the oscilloscope using ps3000_open_unit
2. Select channel ranges and AC/DC switches using ps3000_set_channel
3. Using ps3000_set_trigger, set the trigger if required
4. Using ps3000_get_timebase, select timebases until the required ns per sample is located
5. If required, set the signal generator frequency using ps3000_set_siggen
6. Start the oscilloscope running using ps3000_run_block
7. Wait until the oscilloscope says it is ready using ps3000_ready
8. Transfer the block of data from the oscilloscope using ps3000_get_values or ps3000_get_times_and_values
9. Display the data
10. Repeat steps 6 to 9
11. Stop the oscilloscope using ps3000_stop.

3.5.3 Using streaming mode

This is the general procedure for reading and displaying data in streaming mode:

1. Open the oscilloscope using ps3000_open_unit
2. Select channel ranges and AC/DC switches using ps3000_set_channel
3. Start the oscilloscope running using ps3000_run_streaming
4. Transfer the block of data from the oscilloscope using ps3000_get_values
5. Display the data
6. Repeat steps 3 to 5 as necessary
7. Stop the oscilloscope using ps3000_stop

3.5.4 Using ETS mode

This is the general procedure for reading and displaying data in ETS mode:

1. Open the oscilloscope using ps3000_open_unit
2. Select channel ranges and AC/DC switches using ps3000_set_channel
3. Using ps3000_set_trigger, set the trigger if required
4. Set ETS mode using ps3000_set_ets
5. Start the oscilloscope running using ps3000_run_block
6. Wait until the oscilloscope says it is ready using ps3000_ready
7. Transfer the block of data from the oscilloscope using ps3000_get_times_and_values
8. Display the data
9. Repeat steps 5 to 8 as necessary
10. Stop the oscilloscope using ps3000_stop

Only the 3204/5/6 variants provide this function.

3.6 Programming

3.6.1 C

There are two C example programs: one is a simple GUI application, and the other is a more comprehensive console mode program that demonstrates all of the facilities of the driver.

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

`ps3000.c`;
`ps3000.rc`; and

`ps3000bc.lib`	(Borland 32-bit applications); or
`ps3000.lib`	(Microsoft Visual C 32-bit applications)

The following files must be in the compilation directory:

`ps3000.rch`;
`ps3000.h`;

and the following file must be in the same directory as the executable.

`ps3000.dll`

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

`ps3000con.c`; and

`ps3000bc.lib`	(Borland 32-bit applications); or
`ps3000.lib`	(Microsoft Visual C 32-bit applications).

The following files must be in the compilation directory:

`ps3000.h`;

and the following file must be in the same directory as the executable.

`ps3000.dll`

3.6.2 Visual Basic

The Win32 sub-directory contains the following files:

`ps3000.vbp` - project file
`ps3000.bas` - procedure prototypes
`ps3000.frm` - form and program

Note**:** The functions which return a TRUE/FALSE value, return 0 for FALSE and 1 for TRUE, whereas Visual basic expects 65 535 for TRUE. Check for > 0 rather than =TRUE.

3.6.3 Delphi

The program `ps3000.dpr` demonstrates how to operate PicoScope 3000 Series PC Oscilloscopes. The file `ps3000.inc` contains procedure prototypes that you can include in your own programs. Other required files include `ps300fm.res`, `ps300fm.dfm` and `ps3000fm.pas`. This has been tested with Delphi versions 3.

3.6.4 Excel

1 Load the spreadsheet `ps3000.xls`
2 Select **Tools | Macro**
3 Select **GetData**
4 Select **Run**

Note**:** The Excel Macro language is similar to Visual Basic. The functions which return a TRUE/FALSE value, return 0 for FALSE and 1 for TRUE, whereas Visual Basic expects 65 535 for TRUE. Check for > 0 rather than =TRUE.

3.6.5 *Agilent Vee*

The example function `ps3000.vee` is in the drivers sub-directory. It uses procedures that are defined in `ps3000.vh`. It was tested using Agilent Vee version 5.

4 Troubleshooting

4.1 Software error codes

Consult this section if your are a PicoScope or PicoLog user. If you are writing your own program, refer to the driver error codes section.

PicoLog reports error code 1.
This error is reported when more than 4 oscilloscopes are opened on one machine. It is not possible to use more than 4 oscilloscopes with PicoLog.

PicoScope or PicoLog reports error code 2.
This error is reported when the driver cannot allocate enough of the computer's memory to operate the oscilloscope. Consult the system requirements section for more information.

PicoScope or PicoLog reports error code 3.
This error indicates that a PicoScope 3000 Series PC Oscilloscope could not be found on your machine. Make sure the software is installed before the oscilloscope is plugged into the USB socket and restart your computer.

Ensure that mention of the oscilloscope can be found in the USB section of the Windows Device Manager. If the oscilloscope is not mentioned there, consult Pico Technical Support for further advice.

PicoScope or PicoLog reports error code 4, 5 or 6.
This error is reported when there is a problem with the oscilloscope itself. These problems could arise from configuration settings being corrupted, or a firmware or hardware error.

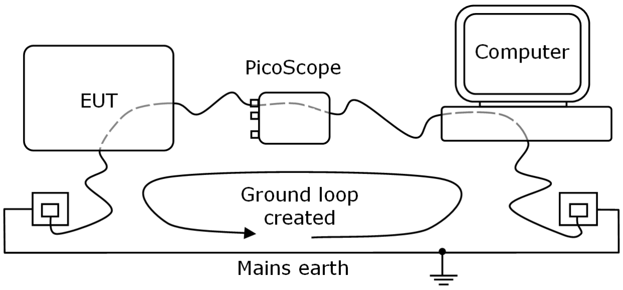
Unplug the oscilloscope, wait a few seconds, and reconnect it to the USB port. If the error is still reported, consult Pico Technical Support for further advice.

PicoScope or PicoLog reports error code 7.
This error is reported if the operating system is not recent enough to support the PicoScope 3000 Series PC Oscilloscope. Consult the system requirements section for more information.

4.2 Powering the 3204/5/6

The PicoScope 3204/5/6 PC Oscilloscopes are normally powered from the USB port of the computer. If the computer and the equipment under test (EUT) are both referenced to the same ground, a "ground loop" may be created. This may degrade the DC accuracy and noise performance when measuring small signals.

Typically, a ground loop is created when the PicoScope is connected to a mains-powered computer and is used to measure a signal on another mains-powered device. In this case the ground loop is created through mains earth, as illustrated below:



The majority of laptop power supplies (chargers) are floating and have no ground reference. If, however, connecting your grounded laptop power supply causes noise/offset problems, you can either use the oscilloscope with the laptop running on its batteries or power the oscilloscope using the supplied mains adaptor.

If necessary, you should plug the mains adaptor into the socket on the back of the oscilloscope (near the USB socket). It can be safely connected and disconnected during operation without risk of damage to the oscilloscope.

5 Glossary

AC/DC switch
To switch from AC coupling to DC coupling, or vice versa, select AC or DC from the control on the oscilloscope toolbar of the PicoScope software application. The AC setting filters out any DC component of the input signal, and is suitable for viewing small AC signals superimposed on a DC or slowly-changing offset. In this mode you can measure the peak-to-peak amplitude of an AC signal but not its absolute value. Use the DC setting for measuring the absolute value of a signal.

Analog bandwidth
The input frequency at which the measured signal amplitude is 3 dB below its true value.

Block mode
A sampling mode in which the computer prompts the oscilloscope to collect a block of data into its internal memory before stopping the oscilloscope and transferring the whole block into computer memory. This mode of operation is effective when the input signal being sampled is high frequency. Note: To avoid aliasing effects, the maximum input frequency must be less than half the sampling rate.

Buffer size
The size of the oscilloscope buffer memory, measured in samples. The buffer memory is used by the oscilloscope to store data temporarily. This allows the oscilloscope to sample data independently of the speed at which it can transfer data to the computer.

Device Manager
Device Manager is a Windows applet that displays the current hardware configuration of your computer. On Windows 98 or Windows ME, right click on 'My Computer' and choose the 'Device Manager' tab. On Windows 2000 or XP, right-click on 'My Computer,' choose 'Properties', then click the 'Hardware' tab and the 'Device Manager' button.

Driver
A software application that controls a piece of hardware. The driver for the PicoScope 3000 Series PC Oscilloscopes is supplied in the form of a 32 bit Windows DLL. This is used by the PicoScope and PicoLog software to control the oscilloscopes.

ETS
Equivalent Time Sampling. ETS constructs a picture of a repetitive signal by accumulating information over many similar wave cycles. This means the oscilloscope can capture fast repeating signals that have a higher frequency than the maximum sampling rate. Note: ETS should not be used for one-shot or non-repetitive signals.

External trigger
This is the BNC socket marked **E** on the PicoScope 3204/5/6 PC Oscilloscopes. It can be used to start a data collection run but cannot be used to record data. As it shares the same connector as the signal generator output, these two functions cannot be used at the same time. It is possible, however, to use the output from the signal generator as a trigger.

Maximum sampling rate
A figure indicating the maximum number of samples the oscilloscope can acquire per second. Maximum sample rates are usually given in MS/s (megasamples per second) or GS/s (gigasamples per second.) The higher the sampling capability of the oscilloscope, the more accurate the representation of the high-frequency details in a fast signal.

PC Oscilloscope
The instrument formed by connecting a PicoScope 3000 Series PC Oscilloscope to a computer running the PicoScope software application.

PicoLog software
This is a software product that accompanies all our oscilloscopes. It turns your PC into a data logger and chart recorder.

PicoScope 3000 Series
An oscilloscope range comprising the PicoScope 3204, 3205, 3206, 3224 and 3424 PC Oscilloscopes.

PicoScope software
This is a software product that accompanies all our oscilloscopes. It turns your PC into an oscilloscope, spectrum analyser, and meter display.

Real time continuous mode
A sampling mode in which the software repeatedly requests single samples from the oscilloscope. This mode is suitable for low sampling rates when you require the latest sample to be displayed as soon as it is captured.

Signal generator
This is a feature on an oscilloscope which allows a signal to be generated without an external input device being present. The signal generator output is the BNC socket marked **E** on the oscilloscope. If you connect a BNC cable between this, and one of the channel inputs, you can send a signal down one of the channels. On some units, the signal generator can generate a simple TTL square wave, while on others it can generate a sine, square or triangle wave that can be swept back and forth. Consult the specifications for further details.

Note: The signal generator output is physically the same as the external trigger input, so these two functions cannot be used at the same time. It is possible, however, to use the output from the signal generator as a trigger.

Streaming mode
A sampling mode in which the oscilloscope samples data and returns it to the computer in an unbroken stream. This mode of operation is suitable when the input signal contains only low frequencies.

Temperature range
The minimum and maximum temperatures between which the oscilloscope is guaranteed to meet its specifications. The 3204/5/6 PC Oscilloscopes are specified at a nominal temperature of 25°C, and the 3224/3424 are specified over the range 20°C to 30°C.

Timebase
The timebase controls the time interval that the width of the scope display represents. If you select **Timebase is time per division** in the **Preferences** dialog box, it works like a traditional bench top scope. There are ten divisions across the screen, so the total time interval is ten times the timebase.

Trigger bandwidth
The maximum frequency at which the trigger circuit will reliably generate a trigger event.

USB 1.1
Universal Serial Bus (Full Speed). This is a standard port that enables you to connect external devices to PCs. A typical USB 1.1 port supports a data transfer rate of 12 Mbps (12 megabits per second), and is much faster than a serial port.

USB 2.0
Universal Serial Bus (High Speed). This is a standard port that enables you to connect external devices to PCs. A typical USB 2.0 port supports a data transfer rate 40 times faster than USB 1.1, and all USB 2.0 ports are backwards-compatible with USB 1.1.

Vertical resolution
A value, in bits, indicating the degree of precision with which the oscilloscope can convert input voltages to digital values. Calculation techniques can improve the effective resolution.

Voltage range
The range of input voltages that the oscilloscope will measure in a given mode.

Index

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